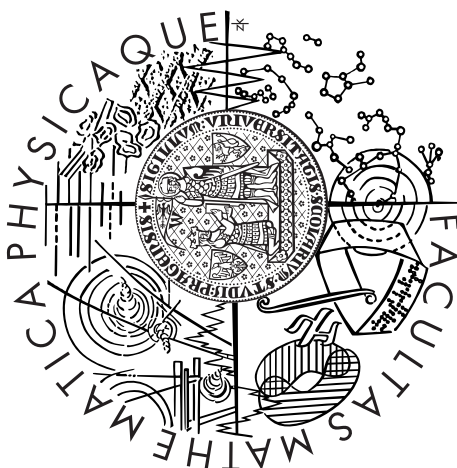


Univerzita Karlova v Praze  
Matematicko-fyzikální fakulta

## BAKALÁŘSKÁ PRÁCE



David Krška

## Správa a analýza družicových dat

Katedra softwarového inženýrství (204. • 32-KSI)

Vedoucí bakalářské práce: RNDr. Jakub Lokoč, Ph.D.

Studijní program: Informatika (B1801)

Studijní obor: Obecná informatika

Praha 2014

Děkuji panu RNDr. Jakubu Lokočovi, Ph.D. za odborné vedení, rady a čas strávený při zpracování této práce. Mé poděkování patří též všem, kteří si tuto práci přečetli a pomohli odstranit její nedostatky.

Prohlašuji, že jsem tuto bakalářskou práci vypracoval(a) samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova v Praze má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle §60 odst. 1 autorského zákona.

V ..... dne .....

Podpis autora

Název práce: Správa a analýza družicových dat

Autor: David Krška

Katedra: Katedra softwarového inženýrství (204. • 32-KSI)

Vedoucí bakalářské práce: RNDr. Jakub Lokoč, Ph.D., Katedra softwarového inženýrství

Abstrakt: Předmětem této bakalářské práce je navrhnout a otestovat interaktivní nástroj pro analýzu, klasifikaci a barevnou vizualizaci družicových dat. Data jsou předzpracována a automaticky načítána a skládána do barevných snímků. Aplikace umožňuje dále vytvářet a porovnávat klasifikační algoritmy, které pracují jak na samostatných snímcích, tak na kombinaci snímků pořízených v různém časovém období. K testování algoritmů je vytvořeno rozhraní pro tvorbu trénovacích dat a dále je v aplikaci implementováno několik běžně používaných klasifikačních algoritmů. Úspěšnost těchto algoritmů byla porovnána v experimentu na datech z projektu Landsat. Výsledkem této práce je tak aplikace primárně zaměřená na klasifikaci dat, která by po vhodném rozšíření mohla sloužit i jako alternativa k dnes běžným GIS systémům.

Klíčová slova: Družicová data; Indexování; Podobnostní vyhledávání; Klasifikace; GIS

Title: Satellite data management and analysis

Author: David Krška

Department: Department of Software Engineering (204. • 32-KSI)

Supervisor: RNDr. Jakub Lokoč, Ph.D., Department of Software Engineering

Abstract: The aim of this bachelor thesis is to implement and test an interactive application for the analysis, visualization and classification of satellite data. The satellite data are preprocessed and automatically composed into colored images. The application allows to create and compare two types of classification algorithms. The first type uses single images and the second type uses multiple images of the same place, but at different times. We also created an interface for selecting and managing the training data. A few well-known classification algorithms of both types were implemented and their success rates were compared in an experiment. All the satellite data used in the experiment are from the Landsat program. The result of this bachelor thesis is an application primarily focused on classification. But the application could also be extended into a complex GIS system.

Keywords: Satellite data; Indexing; Similarity search; Classification; GIS

# Obsah

Úvod	3
<b>1 Data projektu Landsat</b>	<b>5</b>
1.1 Historie satelitů projektu Landsat	5
1.1.1 Landsat 1-6	5
1.1.2 Landsat 7	5
1.1.3 Landsat 8	7
1.2 Předzpracování dat	8
1.3 Formát dat a jejich distribuce	8
1.4 Metadata soubor	9
<b>2 Související práce</b>	<b>11</b>
2.1 Obecné dělení GIS	11
2.2 Oficiální nástroje USGS	11
2.2.1 GloVis	12
2.2.2 LandsatLook Viewer	13
2.2.3 Earth Explorer	13
2.3 Google Maps	13
2.4 GIS systémy	14
2.4.1 GRASS GIS	14
2.4.2 Alternativa k současným GIS	16
2.4.3 QGIS	16
2.4.4 gvSIG	17
2.4.5 SAGA	17
<b>3 Motivace a analýza problému</b>	<b>18</b>
3.1 Statistika využití Landsat dat	18
3.2 Mraky a další negativní vlivy	18
3.3 Klasifikace zemského povrchu	19
3.4 Klasifikace druhů rostlin	19
3.5 Další možnosti detekce	20
<b>4 Struktura aplikace</b>	<b>22</b>
4.1 Obecný logický model aplikace	22
4.1.1 Vlákna	22
4.2 Databázový modul	23
4.2.1 Formát dat na disku	24
4.2.2 Třístupňové předzpracování vizualizovaných dat	24
4.3 Prohlížeč	25
4.3.1 Načítání dat	26
4.3.2 Další vlastnosti prohlížeče	26
4.3.3 Komunikace mezi moduly	27
4.4 Klasifikační modul	27
4.5 Grafické uživatelské rozhraní	28
4.5.1 Vynášení průměrných hodnot třídy do grafu	29

4.5.2	Testování úspěšnosti klasifikátoru . . . . .	30
<b>5</b>	<b>Použité algoritmy a datové struktury</b>	<b>31</b>
5.1	Použité technologie . . . . .	31
5.2	Struktura tříd aplikace . . . . .	31
5.2.1	Index čtverce . . . . .	32
5.3	Asynchronní načítání čtverců . . . . .	33
5.3.1	Signály a sloty v Qt . . . . .	33
5.3.2	Načítání dat . . . . .	33
5.4	Cachování . . . . .	33
5.4.1	Datové struktury uvnitř cache . . . . .	34
5.4.2	Správa a uvolňování paměti . . . . .	34
5.5	Klasifikační algoritmy . . . . .	35
5.5.1	Detekce mraků . . . . .	35
5.5.2	Algoritmy strojového učení . . . . .	35
5.5.3	Algoritmy využívající více scén . . . . .	38
5.5.4	Programátorské rozhraní pro tvorbu klasifikačních algoritmů	38
<b>6</b>	<b>Srovnání úspěšnosti vybraných algoritmů</b>	<b>39</b>
6.1	Vytvoření trénovacích a referenčních dat . . . . .	39
6.2	Struktura prezentovaných dat . . . . .	40
6.3	Vyhodnocení experimentu . . . . .	40
	<b>Závěr</b>	<b>43</b>
	<b>Seznam použité literatury</b>	<b>45</b>
	<b>Seznam tabulek</b>	<b>48</b>
	<b>Seznam použitých pojmů a zkratek</b>	<b>51</b>
	<b>Přílohy</b>	<b>52</b>

# Úvod

S vyvíjejícími se technologiemi pořizují satelity stále kvalitnější snímky. Zlepšuje se rozlišení snímků a zvyšuje se množství spekter, ve kterých jsou senzory schopny tyto snímky pořizovat. Satelity tak už lidem neslouží pouze pro tvorbu map nebo předpovídání počasí, ale lze s nimi provádět velké množství různých analýz zemského povrchu. S přibývajícím množstvím satelitních dat již přestává být možné zpracovávat tato data pouze lidskou silou a je tak také kladen velký důraz na využívání automatického zpracování.

Jednou z častých úloh analýzy satelitních dat je klasifikace zemského povrchu, kdy se uživatel snaží rozpoznat, co se v dané části povrchu právě nachází. Uživatel vybere několik oblastí, kterým přiřadí třídu, do které patří. Na základě těchto dat klasifikační algoritmus rozdělí do tříd i zbylé oblasti. Tomuto procesu se říká klasifikace s učitelem. Zemský povrch lze dělit na základní třídy, jako jsou voda, sníh, les, pole a další. Ke klasifikaci lze ale také využít mnoho spektrálních pásem, která jsou lidským okem neviditelná. Především jsou využívána spektra infračerveného záření. Díky nim lze například rozlišovat i různé druhy rostlin, které v dané oblasti rostou. Tyto a další možnosti klasifikace jsou důkladně popsány v kapitole 3.

Vizualizaci, analýzu a klasifikaci satelitních dat zvládá mnoho aplikací. Především se na tuto problematiku zaměřují Geografické informační systémy (dále GIS). Jedná se o velice komplexní systémy, které kromě zpracování a analýzy dat slouží také k úpravě, anotaci či vytváření zcela nových mapových podkladů. GIS systémy často obsahují pouze jeden odladěný klasifikační algoritmus. Pro různá data a různé trénovací množiny může být ale výhodnější využívat rozdílné algoritmy. Jsou-li vyžadovány velice přesné výsledky, není výjimkou vytváření specifických algoritmů na konkrétní data. Vytváření a porovnávání klasifikačních algoritmů je právě jedním z cílů zde popisované aplikace.

Částečně vznikla tato aplikace jako alternativa k současným GIS systémům. Umožňuje načítat, analyzovat a vizualizovat satelitní data do člověku co nejpřirozenější podoby. Snímky ze satelitů se automaticky skládají do barevných obrázků na základě uživatelem volených parametrů. Kromě toho umožňuje sledovat změnu vybraných oblastí v čase a tento vývoj vynášet do grafu. Výsledky této analýzy je pak možné využít pro návrh nových a efektivnějších algoritmů.

Klasifikaci dat je pak možné provádět ve dvou režimech. Jeden je určený pro klasifikaci samostatných scén a v druhém režimu mohou klasifikační algoritmy využívat více scén současně pořízených v různém čase. Součástí aplikace je i několik vzorových algoritmů, které využívají obecně známé algoritmy strojového učení. Pro snadnější tvorbu algoritmů je pak k dispozici jednotné programátorské rozhraní. Obecnou strukturu aplikace, včetně tohoto rozhraní a popisu grafického uživatelského rozhraní, se může čtenář dozvědět v kapitole 4. Kapitola 5 pak obsahuje informace o vnitřním fungování aplikace. Popisuje technologie využitě při tvorbě a další implementační detaily, jako je komunikace jednotlivých komponent nebo využitě algoritmy. Součástí je i popis algoritmů strojového učení využitých pro zde implementované klasifikační algoritmy.

K obecnému GIS systému má ale tato aplikace poměrně daleko. Není účelem implementovat veškeré funkcionality, které mají současné GIS systémy. Cílem je

naopak co nejefektivněji a uživatelsky nejvýhodněji řešit některé vybrané úlohy, jako je právě vizualizace, analýza dat v čase a následná klasifikace.

K tomu aplikace využívá data z projektu Landsat [1]. Tento projekt funguje již od doby prvního dobývání vesmíru a jeho data jsou volně dostupná ke stažení. V současné době má nejnovější satelit tohoto projektu k dispozici 11 různých senzorů a pokrývá vlnové délky od ultrafialového záření až po tepelné záření. Všechny důležité vlastnosti těchto dat se čtenář může dozvědět v kapitole 1, která se věnuje pouze tomu.

V poslední kapitole 6 je pak provedeno srovnání zde implementovaných klasifikačních algoritmů. V něm je naměřena jejich úspěšnost ve třech samostatných experimentech na různých velikostech dat. Součástí experimentů je pak i porovnání klasifikačních algoritmů pracujících na samostatných scénách a algoritmů upravených pro klasifikaci několika scén dohromady.

# 1. Data projektu Landsat

## 1.1 Historie satelitů projektu Landsat

Landsat je nejdéle fungující projekt satelitního snímkování Země a je provozovaný Americkým geologickým ústavem (dále USGS) [1]. Pod hlavičkou organizace NASA bylo do této chvíle (2014) postupně vypuštěno 8 satelitů. Data jsou ale dostupná pouze ze 7 satelitů, poněvadž satelitu Landsat 6 se nepodařilo dosáhnout oběžné dráhy. Z těchto 7 satelitů jsou v současné době funkční pouze dva nejnovější.

### 1.1.1 Landsat 1-6

První satelit byl vypuštěn 23.července 1972 a svoji misi skončil 6.ledna 1978. S sebou nesl dvě zařízení Return-Beam Vidicon (RBV) a Multispectral Scanner (MSS). RBV byla kamera snímající viditelné světlo a blízké infračervené záření. Měla tři senzory snímající vlnové délky v rozmezí 480 až 570 nm. MSS se čtyřmi senzory snímala vlnové délky v rozmezí 500 až 1100 nm a oba senzory fungovaly s rozlišením 80 m. V jednom bodu snímku je tak zaznamenána oblast  $80 \times 80$  metrů. Dále je každý bod snímku reprezentován jako 8 bitové číslo a může nabývat hodnot 1 až 255. Hodnota 0 je vyhrazena pro body, které se nachází mimo scénu.

Velice podobní byli i jeho dva následníci Landsat 2 a Landsat 3, kteří fungovali postupně v rozmezí 22.1.1975 - 25.2.1982 a 5.3.1978 - 7.9.1983. Vybavení satelitu Landsat 2 bylo zcela totožné. V Landsat 3 bylo vylepšeno RBV zařízení na rozlišení 40 m a k MSS byl přidán pátý senzor snímající tepelné záření s rozlišením 240 m. Tento senzor ale brzy po spuštění selhal.

Landsat 4 fungující mezi 16.7.1982 a 15.6.2001 měl také MSS a k tomu byl vybaven novým zařízením Thematic Mapper (TM). TM měl sedm senzorů, 3 ve viditelném spektru, 3 v infračerveném spektru a jeden využívající tepelné záření. Všechny senzory snímají v rozlišení 30 m, kromě tepelného, který má rozlišení 120 m. Snímky těchto tepelných senzorů jsou ale následně převzorkovány na rozlišení 60 nebo 30 metrů. Stejně vybavení měl i následující satelit Landsat 5. Ten dokázal nepřetržitě posílat snímky po dobu 27 let, čímž se stal nejdéle fungujícím satelitem, který kdy snímkoval Zemi. Odstartován byl 1.3.1984 a data posílal až do 5.6.2013. Šestý satelit byl vypuštěn 5.10.1993 a měl nést vylepšené zařízení ETM, ale jak již bylo zmíněno, neměl možnost poslat žádný snímek. Vybavení satelitů 1-7 je shrnuté na obrázku 1.1.

### 1.1.2 Landsat 7

Sedmý satelit byl vypuštěn 15.4.1999 se zařízením Enhanced Thematic Mapper Plus (ETM+), které mělo nahradit TM. Prvních sedm senzorů snímá ve stejných vlnových délkách, jako TM. Navíc má ještě panchromatický senzor s rozlišením 15 m. Vlnové délky panchromatického senzoru jsou z celého viditelného a části infračerveného spektra. Díky širokému rozsahu vlnových délek, ve kterých pracuje, pak může dosáhnout mnohem přesnějšího rozlišení.

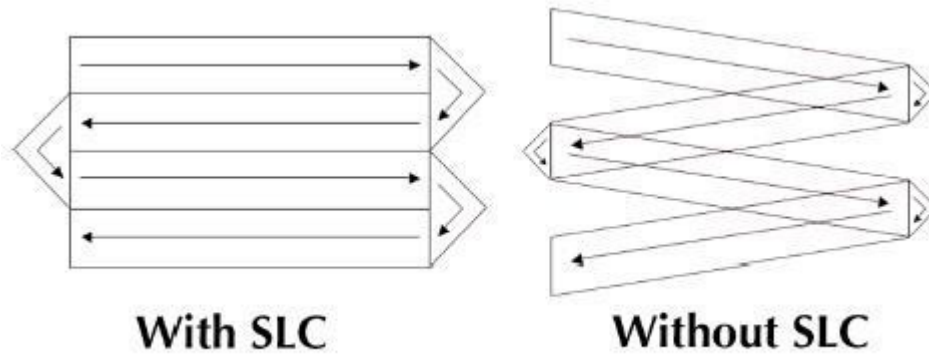
Satellite	Sensor	Bandwidths	Resolution	Satellite	Sensor	Bandwidths	Resolution
LANDSATs 1-2	RBV	(1) 0.48 to 0.57	80	LANDSATs 4-5	MSS	(4) 0.5 to 0.6	82
		(2) 0.58 to 0.68	80			(5) 0.6 to 0.7	82
		(3) 0.70 to 0.83	80			(6) 0.7 to 0.8	82
	MSS	(4) 0.5 to 0.6	79		TM	(7) 0.8 to 1.1	82
		(5) 0.6 to 0.7	79			(1) 0.45 to 0.52	30
		(6) 0.7 to 0.8	79			(2) 0.52 to 0.60	30
		(7) 0.8 to 1.1	79			(3) 0.63 to 0.69	30
LANDSAT 3	RBV	(1) 0.505 to 0.75	40	LANDSAT 7	ETM <sup>+</sup>	(4) 0.76 to 0.90	30
		(5) 0.6 to 0.7	79			(5) 1.55 to 1.75	30
	MSS	(4) 0.5 to 0.6	79			(6) 10.4 to 12.5	120
		(5) 0.6 to 0.7	79			(7) 2.08 to 2.35	30
		(6) 0.7 to 0.8	79				
	(7) 0.8 to 1.1	79					
	(8) 10.4 to 12.6	240					
						PAN 0.50 to 0.90	15

Obrázek 1.1: Přehled Landsat satelitů a jejich senzorů [2]

Senzory jsou očíslované od 1 do 8. Přesto, že satelit má pouze 8 senzorů, obsahuje každá scéna celkem 9 snímků. První tři jsou senzory běžného viditelného záření odpovídající složkám modrá, zelená a červená. Poté následují dva snímky blízkého infračerveného záření. Šestý senzor je tepelný a je rozdělený na dva snímky podle citlivosti senzorů. První snímek je snímán s menší citlivostí a má příponu VCID\_1. Druhý snímek s větší citlivostí má příponu VCID\_2. Následuje senzor krátkovlnného infračerveného záření a poslední senzor je již zmíněný panchromatický. Skutečné rozlišení tepelného senzoru je zde 60 m a je opět převzorkován na 30 m.

Dvě možné hladiny citlivosti byly nově představeny právě v satelitu Landsat 7. Snímek každého senzoru satelitu je pořízen buď s nízkou nebo vysokou citlivostí. Obě citlivosti jsou použity pouze u tepelného senzoru. Cílem tohoto nastavení je co nejefektivněji využít 8 bitový rozsah senzorů. Více osvětlené oblasti je snaha snímat s nízkou citlivostí, naopak méně osvětlené oblasti jsou snímány s vysokou. Míra osvětlení je určena úhlem, pod kterým dopadají sluneční paprsky. Dále byly oblasti zemského povrchu rozděleny do kategorií podle předpokládané odrazivosti světla. Kategorie jsou země, poušť, sníh, voda, zamrzlé moře a vulkány. Přiřazená kategorie a osvětlení scény pak určují zda u vybraného senzoru bude nastavena nízká nebo vysoká citlivost. Detailnější informace lze získat například z internetové příručky organizace NASA [4].

Satelit je stále funkční, ale z důvodu neopravitelné poruchy neposílá data kompletní. K poruše došlo v květnu roku 2003 na korektoru řádek obrazu. Za normálních okolností se senzory pohybují tak, aby souvisle pokryly celou snímanou oblast. Senzory se natáčejí po celé šířce scény a po řádcích snímají zemský povrch. Jejich složení dá následně vzniknout výsledné scéně. Protože samotný proces snímání vyžaduje určitý čas, je nutné do pohybu senzorů zohlednit také rychlost satelitu. A právě o vykompenzování rychlosti satelitu se stará korektor řádek obrazu (dále SLC). Pohyb senzoru v případech kdy je korektor zapnutý (SLC on) a když je vypnutý (SLC off) je dobře vidět na obrázku 1.2. Při vypnutém SLC nedochází ke správnému posunování senzorů na okrajích scén, a tudíž



Obrázek 1.2: Vlevo je vidět pohyb senzoru s funkčním SLC a vpravo pohyb senzoru s vypnutým SLC [3]

část těchto okrajů je zaznamenána dvakrát a druhá část ani jednou. Na každé scéně je tím ztraceno přibližně 22 procent dat. Zbylá data jsou ale stále použitelná i pro klasifikaci v tomto programu.

### 1.1.3 Landsat 8

Zatím poslední satelit Landsat 8 byl vypuštěn 11.2.2013. Nese zařízení OLI s 9 senzory a 2 TIRS senzory. Důležitou změnou je, že každý pixel senzoru již není reprezentovaný 8 bity, jako to bylo u všech předešlých satelitů, ale je reprezentovaný 16 bity. Může tak nabývat hodnot od 1 do 65535, hodnota 0 je vyhrazena pro pixely mimo snímanou oblast.

Další změnou jsou 2 nové senzory. Sedm sensorů zařízení OLI jsou opět shodné se zařízením ETM+ předchozího satelitu. K nim přibyl senzor s názvem Coastal/Aerosol. Detekuje blízké ultrafialové záření a je především určen k měření kvality vody. Druhý přidaný senzor se nazývá Cirrus a měří infračervené záření. Jak jeho název napovídá, je přidaný za účelem detekce mraků ve vysokých nadmořských výškách nazývaných jako Cirrus. Poslední dva senzory jsou opět v zařízení TIRS a snímají tepelné záření. Tentokrát jsou data zaznamenána s rozlišením 100 m s následným převzorkováním na rozlišení 30 m.

Kromě těchto 11 snímků je součástí stažených dat ještě jeden snímek s označením QAB (Quality Assessment Band). Nejedná se o snímek z žádného senzoru, ale o výsledky analýzy na zbylých snímcích. Jsou v něm zaznamenány různé informace o povrchu, atmosféře a stavu senzoru, které mohou mít vliv na použitelnost daného pixelu. 16 bitů každého pixelu je rozděleno na skupiny po jednom nebo dvou bitech, kde každá skupina reprezentuje určitou vlastnost. Pokud se jedná o skupinu dvou bitů, je v ní zaznamenána míra jistoty detekce dané vlastnosti. Přitom 4 možné hodnoty pixelu reprezentují: žádnou detekci, 0-33 %, 34-66 % a 67-100 % jistotu detekce. Vlastnosti, které skupiny detekují jsou například mraky, sníh, voda, vegetace a další. Další informace jsou dostupné ze stránek USGS [5].

V tomto článku se především zaměřím na data z posledních dvou satelitů Landsat 7 a 8. Jedná se o satelity, které stále produkují data a svým vybavením umožňují mnohem širší možnosti než starší typy satelitů. Cílem tohoto programu není vytvořit obecný systém, který bude univerzálně zpracovávat veškerá možná

data, ale systém fungující s homogenními daty, ve kterých lze sledovat a analyzovat časový vývoj. Návrh programu, který bude popsán později, ale umožňuje v případě potřeby další rozšíření i pro jiná data z jiných programů, případně pro zobecnění stávajícího načítání dat. U starších satelitů, než je Landsat 7 a 8, není zcela zaručena funkčnost.

## 1.2 Předzpracování dat

Velikost každé scény není vždy stejná, přibližně se ale pohybuje okolo 200 x 200 km. K identifikaci každé scény využívá NASA Celosvětový referenční systém (WRS). Jsou dvě verze tohoto systému, WRS 1 pro data ze satelitů Landsat 1,2 a 3 a WRS 2 pro satelity novější. V principu fungují oba systémy stejně, pouze se liší v konkrétním umístění scén. Každá scéna je jednoznačně určena dvojicí čísel Path a Row. Číslo Path označuje cestu, po které se satelit právě pohybuje, a číslo Row určuje pozici satelitu na dané cestě.

Veškerá pořízená data jsou před zveřejněním zpracovávána systémem Level 1 Product Generation System (LPGS) především do podoby Level 1T. Na některých odlehlejších místech, kde nejsou k dispozici kontrolní stanoviště, se můžou data zpracovávat do méně přesné podoby Level 1Gt nebo Level 1G. Datům jsou nejprve pomoci kontrolních stanovišť přiřazené přesné zeměpisné souřadnice. Následně je provedena mapová projekce dat na plochu, známá jako Polární Stereografická projekce. Obrazem projekce je souřadnicový systém, kde jedna jednotka odpovídá jednomu metru reálné vzdálenosti. Orientace snímku je vždy ve směru severního pólu. Součástí projekce je dále převzorkování a naškálování tak, aby jeden pixel odpovídal reálnému čtverci o hraně, která je daná rozlišením senzoru (především 30 m pro TM, ETM+ a OLI). Každý pixel je navíc přesně zarovnán tak, aby roh pixelu byl ve výsledných projektivních souřadnicích násobek třiceti [6].

Výsledkem tohoto předzpracování je scéna obdélníkového tvaru. Satelit se ale nepohybuje rovnoběžně s žádnou rovnoběžkou ani poledníkem. Čtverec pořízený satelitem je proto otočený vždy ve směru pohybu satelitu. Aby se dosáhlo orientace směrem k severnímu pólu, je snímku opsán obdélník orientovaný na sever a zbylá data jsou doplněna hodnotou nula, odpovídající černé barvě. V této podobě jsou data i distribuována. Nevýhodou tohoto postupu je fakt, že může vznikat mnoho nadbytečných černých pixelů. V případě 45 stupňového natočení satelitu vzhledem k zeměpisným souřadnicím dochází až ke zdvojnásobení velikosti snímku. K dispozici jsou tak ale veškerá data, získaná senzorem.

## 1.3 Formát dat a jejich distribuce

Scény lze stáhnout z oficiálních stránek USGS a mají příponu „.tar.gz“. Tato přípona značí, že data jsou zabalená do tar formátu a následně zazipovaná do gzip formátu. Archiv obsahuje složku se snímky jednotlivých senzorů a několik dalších příložených souborů. Snímek jednoho senzoru je publikován ve formátu GeoTIFF. Jedná se o tiff soubor, který má navíc několik tiff tagů určující jeho geografické zařazení. Přidání těchto tagů umožňuje spojit samotná data s geografickým zařazením do jediného souboru. Obsahují informace o času pořízení,

zeměpisné souřadnice, či souřadnice v projektivní rovině. Můžou ale obsahovat spoustu dalších informací, jako jsou například výsledky geografické analýzy. Stejně informace ale obsahuje i přiložený metadata soubor popsany níže. K získávání informací o datech využívá zde popisovaná aplikace tento soubor, tudíž je možné zacházet s každým snímkem jako s běžným tiff souborem.

Metadata soubor je vždy přiložený ke scéně a má koncovku MTL.txt. Další přiložené soubory se liší podle toho, z jakých satelitů pochází. Scény ze satelitů 1-7 mají například README soubor s obecnými informacemi o datech, dále u některých satelitů může být přiložen soubor s příponou GCP.txt obsahující informace o využitých kontrolních bodech a další soubory. Pojmenování souboru se scénou má svá pevná pravidla a jsou v něm obsaženy informace o jejím zařazení. Příkladem může být název scény „LE71920262014162ASN00“. Z názvu lze získat následující informace:

- L značí, že se jedná o projekt Landsat.
- E je senzor, v tomto případě ETM+.
- 7 označuje pořadové číslo satelitu.
- 192, 026 je dvojice path, row a definuje globální WRS souřadnice scény. Tato dvojice konkrétně značí oblast jižních a středních Čech.
- 2014, 162 určuje datum pořízení scény. 2014 je rok a 162 je den v Juliánském kalendáři
- Následující tři znaky ASN označující stanici, která data přijala a zpracovala.
- Poslední číslo 00 je číslo verze archivu.

Toto označení se bere jako jednoznačný identifikátor a v názvu ho mají i veškeré soubory uvnitř, které obsahují nějaká data. Jednotlivé snímky mají ještě na konci přidanou a podtržítkem oddělenou informaci identifikující konkrétní senzor. Většinou je v podobě jednoho čísla označující pořadové číslo senzoru nebo se může jednat o zkratku názvu daného snímku. Zbylé soubory ve scéně, převážně textové, mohou mít speciální příponu, jako je právě MTL či GCP. Dále se ve scéně může vyskytnout například README.GTF soubor obsahující informace o daném produktu.

## 1.4 Metadata soubor

Metadata soubor je rozdělen na několik skupin. Každá skupina začíná označením „GROUP = NÁZEV SKUPINY“ a je ukončena „END\_GROUP = NÁZEV SKUPINY“. Celý dokument je pak uvnitř jedné velké skupiny, která má název „L1\_METADATA\_FILE“. Každá skupina obsahuje různé vlastnosti dat a další informace. Jednotlivé položky jsou ve tvaru „NÁZEV POLOŽKY = HODNOTA“. Celkově tak lze skupiny a položky reprezentovat ve stromové struktuře. Díky tomu je možné převést tento dokument i do xml formátu pomocí aplikace Metadata Service dostupné ze stránek USGS.

První skupinou v souboru je vždy „METADATA\_FILE\_INFO“ obsahující informace o tvorbě samotných souborů scény. Obsahuje například datum vytvoření či identifikátor stanice zpracovávající scénu či verzi nástroje, který data zpracovával. Následuje skupina „PRODUCT\_METADATA“, kde je geografické zařazení snímku. Zde jsou geografické a projektivní souřadnice rohů jednotlivých snímků, path, row, datum pořízení snímku nebo názvy souborů obsažených ve složce scény. Jak bylo zmíněno výše, některé tyto informace lze najít i přímo v jednotlivých tiff souborech.

Následují další skupiny, ve kterých se nachází velké množství dalších informací, ale které se u jednotlivých satelitů mohou lišit. Tento soubor se postupně vyvíjel a často tam byly přidávány či odebírány různé položky. Z těch, které mají nějaký význam i pro zde popisovanou aplikaci, lze zmínit například rozlišení sensorů, rozměry jednotlivých snímků, rozsah hodnot, které mohou pixely snímků nabývat, či třeba podíl mraků v dané scéně.

## 2. Související práce

K práci se satelitními daty jsou nejčastěji využívány Geografické informační systémy (GIS). Většinou se jedná o velice komplexní systémy, které slouží k získávání, ukládání, správě, vizualizaci a analýze dat, která mají prostorový vztah k povrchu Země. Několik nejznámějších GIS systémů bude představeno v této kapitole.

### 2.1 Obecné dělení GIS

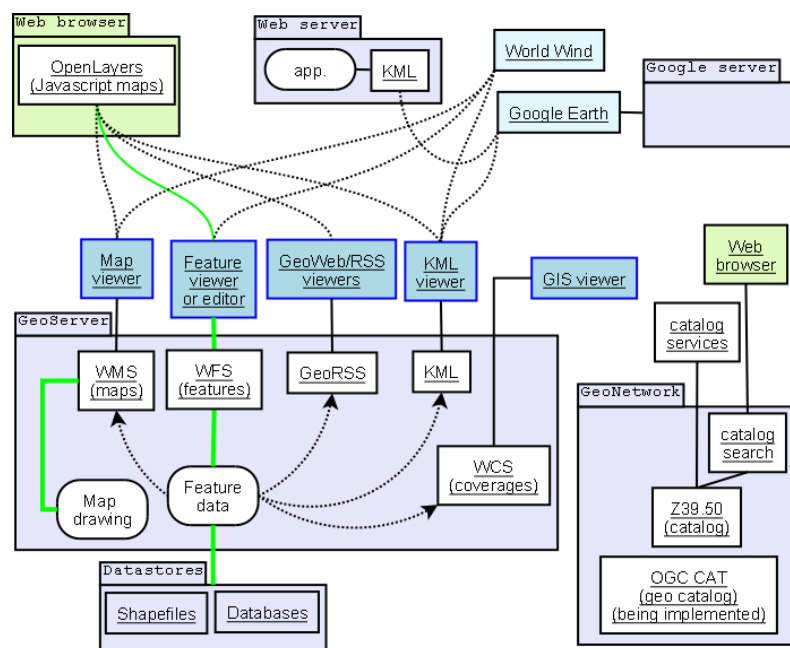
V zásadě jsou dvě možnosti, jak pracovat s daty v GIS systémech. První možnost je využívat data uložená na stroji s GIS aplikací. Ta je často využívána zejména pro účely analýzy nebo klasifikace snímků. Množství zpracovávaných dat je pak omezené paměťovými limity aktuálního stroje. Uživatel se ale může vyhnout zbytečnému přenášení velkého množství dat po síti. Toto je vhodné zejména v situacích, kdy předpokládáme opětovné používání stejných dat. Druhou možností je využít služeb takzvaných Geo Serverů a získávat tak data po síti až v momentě, kdy jsou skutečně potřeba. Geo Servery bývají napojené na databázové servery a díky tomu umožňují spravovat velké množství dat. Často je tento přístup uplatněn u různých vizualizačních nástrojů. Většinou tyto nástroje umožňují přístup k mnoha mapám z celého světa, zároveň se ale nepředpokládá, že by na nich uživatel prováděl nějaké výpočty. Mohou být buď ve formě webové aplikace, příkladem je třeba Google Maps, nebo ve formě desktopové aplikace, mezi které patří NASA World Wind či Google Earth.

Samotné mapy se pak dělí na dvě skupiny – rastrové a vektorové. Snímky pořízené senzory bývají téměř výhradně rastrové. Vektorové pak mohou být výsledkem anotace dat, různých druhů analýzy nebo mohou být zcela lidským tvorem. Poněvadž GIS nástroje mohou mít opravdu rozličné funkce, byla snaha vše co nejvíce standardizovat. Pro tyto účely vznikla organizace Open Geospatial Consortium (OGC) [7], která vytvořila desítky specifikací pro poskytování dat přes web, formát ukládání anotovaných dat a další specifikace. Cílem této organizace je umožnit přenositelnost dat a další spolupráci jednotlivých GIS systémů. Obrázek 2.1 ukazuje, jak vidí OGC strukturu klientů a serverů, jejichž komunikaci se pokouší standardizovat.

Zde popisovaný software využívá lokálně uložená data. Nemá za cíl implementovat jednotlivé standardy, ale spíše navrhnout alternativní architekturu geografického systému pro analýzu a klasifikaci dat. Komplexní GIS systém, splňující navíc tyto standardy, by daleko přesáhl možnosti bakalářské práce. Pokud by ale měl být v budoucnu tento software dopracován do plnohodnotného GIS systému, určitě by bylo nutné tyto standardy implementovat.

### 2.2 Oficiální nástroje USGS

Samo USGS vytvořilo několik nástrojů pro analytické účely a k získávání některých dalších informací z dat z projektu Landsat [8]. Jedním z nich je nástroj Earth Now, pomocí kterého lze sledovat aktuálně snímaná data z projektu Landsat 7. Dále pomocí nástroje Spectral Viewer je možné vizualizovat jak jednotlivé



Obrázek 2.1: Síť serverů, klientů a webových aplikací standardizovaná organizací OGC [9]

senzory satelitů měří intenzitu světla v různých vlnových délkách. To umožňuje vybrat senzory, které se například nejvíce hodí pro rozlišení námi vybraných oblastí. A dalšími nástroji jsou třeba Metadata Service, který je určen pro převod metadata souboru do formátu xml či L-LDOPE Toolbelt, pomocí něhož lze získat některé vlastnosti, charakteristiky a informace o kvalitě konkrétních snímků ze satelitu Landsat 8.

Veškeré výše popsané nástroje jsou celkem jednoduché aplikace, které mohou pomoci s interpretací a analýzou dat. Kromě nich USGS ještě spravuje tři nástroje pro vyhledávání, procházení a stahování scén. Všechny tři jsou webové aplikace, které navíc většinou umožňují stahovat data i z jiných projektů nebo stahovat výsledky jejich analýzy. Ve své podstatě jsou všechny tři nástroje velice podobné, pouze se liší ve způsobu jak si data může uživatel vybírat, procházet a zobrazovat. Ke stažení jsou pak k dispozici sestavené RGB snímky i původní data v Level 1 formátu. Veškerá data jsou k dispozici zdarma, pouze je vyžadována registrace. Velká část snímků, především ty s dobrým počasím, jsou k dispozici k okamžitému stažení. Zbylé snímky je nutné objednat. Jejich příprava pak může zabrat i několik dnů.

### 2.2.1 GloVis

Jedním z těchto tří nástrojů je GloVis [10]. Jedná se o Java applet, který zobrazuje vždy vybranou scénu a k ní osm sousedních scén v jejím okolí. Poněvadž každá scéna je pořízená v jiném čase, jsou sousední scény vybírány jako časově nejbližší. Vybráním sousedních scén nebo vybráním WRS souřadnic se uživatel může pohybovat v prostoru, stejně tak může vybírat ze snímků z různého časového období. Dále samozřejmě může o vybrané scéně získat detailní informace, které lze najít i v metadata souboru či přidat ji do listu ke stažení.

## 2.2.2 LandsatLook Viewer

Další nástroj je LandsatLook Viewer [11]. Ten se liší od předchozího nástroje tím, že uživatel prochází digitální mapy a zde si vybírá oblast, o kterou má zájem. Po vybrání oblasti si může nechat zobrazit všechny scény, které tuto oblast pokrývají, měnit časový rozměr a stejně jako u GloVis data stahovat. Tentokrát má ale k dispozici data pouze z projektu Landsat.

## 2.2.3 Earth Explorer

Poslední nástroj se jmenuje Earth Explorer [12] a funguje podobně jako LandsatLook Viewer, akorát umožňuje stahovat data i z jiných projektů než je projekt Landsat. Je postavený na satelitních a leteckých snímcích od společnosti Google, kde si uživatel vybere body na Zemi, o které má zájem. Po zvolení satelitů, ze kterých mají data být, může procházet všechny scény, na kterých je alespoň jeden z vybraných bodů.

Výše popsané nástroje jsou webové aplikace a nepatří do skupiny GIS systémů, zároveň jsou ale nezbytnou součástí při výběru dat. Výběr vhodných dat je jednou z důležitých součástí analýzy povrchu Země a může mít velký vliv na výsledek klasifikace.

## 2.3 Google Maps

Pro vizualizaci satelitních a i leteckých dat slouží také webové aplikace kterými jsou Google Maps či Mapy.cz. Tyto aplikace nejsou určené k analyzování dat. Jejich účel je zobrazování satelitních snímků, mapových podkladů a následná navigace a orientace v nich. Jsou výrazně anotované a veškerá data jsou přenášena po síti. Je proto velký důraz na to přenášet co nejmenší množství dat, i přesto se ale průměrný uživatel nevyhne časové prodlevě způsobené čekáním na data. V případě výhradně vizualizačních nástrojů lze komprimovat a snižovat jejich kvalitu, poněvadž jde především o vizuální dojem. Jsou-li ale součástí aplikace také nástroje pro analýzu, je nutné data přenášet v plné kvalitě.

Pro tyto účely je proto používán způsob zobrazování, kde souřadnicový systém je rozdělený na čtverce, po kterých jsou data přenášena a zobrazována. Data jsou pak po částech průběžně přidávána, což vede k lepší a interaktivnější práci s mapami. Veškerá data se navíc musí na straně klienta co nejvíce cachovat, což rozřezání na čtverce výrazně usnadňuje.

Toto rozřezání na čtverce jsem také využil ve své aplikaci. V běžných GIS systémech se ale tento přístup téměř nepoužívá. GIS systémy zpracovávají snímky vždy jako jeden celek. Díky tomu naopak nemusí data rozřezávat ani jinak předzpracovávat. V případě, že jsou ale data získávána po síti, může být rozdělení dat na menší bloky velkou výhodou. U těchto webových aplikací je podobný přístup téměř nutností.

V případě zde popisované aplikace naráží tento způsob na fakt, že téměř neexistuje formát, který by zmíněný princip využíval. V GIS systémech pro to není podpora, tudíž takto formátovaná data budou téměř nepřenositelná na další systémy. Tím zároveň nelze využívat již připravené GIS knihovny pro správu dat, které ostatní systémy využívají.

## 2.4 GIS systémy

Následující systémy již patří do skupiny GIS. Těchto systémů existuje velké množství a není v možnostech této práce všechny zmínit. Zaměřím se proto na ty nejznámější a volně dostupné pod licencí Open Source. Především se zaměřím na Grass GIS, o kterém by se dalo říci, že určuje směr současných GIS systémů a zároveň z něj tato práce částečně vychází.

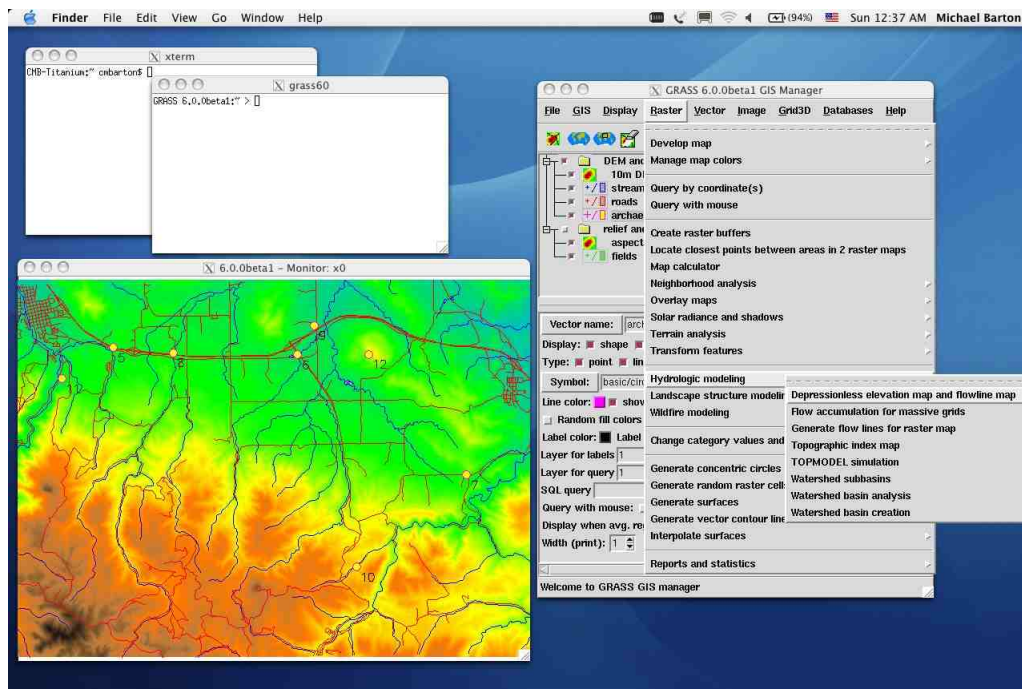
### 2.4.1 GRASS GIS

GRASS GIS [13] je jedním z nejznámějších a zároveň nejstarších GIS systémů. Od roku 1982 byl vyvíjen americkou armádou pro vojenské účely. Na konci 80. let byl uvolněn veřejnosti a později jeho licence změněna na GNU GPL. Díky velkým investicím do vývoje ze strany americké vlády a s dalším vývojem na univerzitách se postupně prosadil jak do veřejné, tak do soukromé sféry. V současné době ho stále využívá například organizace NASA, NOAA, USDA, USGS a další. Hojně je také využíván v akademické sféře pro vědecké účely.

Původně byl tento software vyvíjen pro UNIX a od toho se také odvíjí jeho architektura. Tato aplikace není založená na žádném grafickém rozhraní, ale spíše na zpracovávání dat na základě příkazů. Svým konceptem se tak výrazně liší od ostatních systémů podobného zaměření. Základem aplikace je běžná příkazová řádka s UNIXovým shellem, rozšířená o příkazy vztažené ke zpracování geografických dat, které se nazývají moduly. Tyto moduly pracují nezávisle na sobě a mohou provádět téměř libovolné operace. Později bylo nad nimi vytvořeno uživatelské rozhraní, umožňující dané příkazy vykonávat kliknutím myši, nebo kombinovat příkazovou řádku s různými grafickými prvky.

Samotný GRASS je tedy spíše prostředí, do kterého je možné přidat takové moduly, které budou odpovídat našim potřebám a pomocí nich provádět třeba klasifikaci nebo nějakou analýzu. Tato modularita je velice výhodná pro vědecké účely a proto je GRASS také často používán v univerzitním prostředí. Spoustu modulů lze stáhnout z internetu a snadno nainstalovat, případně si naprogramovat vlastní, aniž by člověk musel rozumět jiným částem aplikace. Momentálně existuje asi 300 modulů, které jsou součástí jádra aplikace a dále okolo 100 modulů vytvořených uživateli a dostupných z oficiálních stránek projektu [13]. Díky UNIXové filozofii, kterou tento program využívá, mohou být jednotlivé moduly vytvořeny téměř v libovolném jazyce. Moduly jádra jsou většinou napsané v programovacím jazyce C. Další jsou potom psané v C++, Pythonu či v samotném UNIXovém shellu. Často pak i kombinací různých dalších skriptovacích jazyků. Nespornou výhodou je pak i možnost kombinovat moduly, vytvářet skripty a automaticky tak data zpracovávat či analyzovat.

Data, se kterými uživatel pracuje, jsou rozdělena do datových sad. Každá sada má své jméno a lze s ní pracovat jako s jedním celkem. Zadávané příkazy jsou pak ve tvaru *pkaz.nzev\_moduluparametry*. Příkaz je většinou jednopísmenný a označuje o jaký typ operace půjde a jaký druh dat bude zpracováván. Základní příkazy jsou *g*, *r* a *v*, kde *g* je pro práci s obecnými soubory, *r* pracuje s 2D rastrovými daty a *v* pracuje s vektorovými daty. Následuje název modulu a parametry vztažené k danému modulu. Příkladem může být třeba příkaz *g.listtype = rastmapset = sada1*. Zde *list* je modul pro zobrazení všech map



Obrázek 2.2: GRASS GIS [14]

v dané mapové sadě, parametr *type = rast* značí pouze rastrové mapy a *mapset* označuje zvolenou mapovou sadu. Více o příkazové řádce se čtenář dočte v manuálu k systému [15].

Výstup každého příkazu může být jak konzolový, tak i v dialogu v rámci grafického uživatelského rozhraní. Každý příkaz je tak převoditelný do předpřipraveného dialogu se čtyřmi záložkami. V první jsou povinné parametry, ve druhé volitelné, ve třetí se zobrazuje výstup a v poslední je klasický UNIXový manuál o daném příkazu. Moduly se nemusí omezovat pouze na tento dialog, lze vytvořit i vlastní, ale pro účely většiny aplikací je tento dialog dostačující. Pro dnešního uživatele je pak významný modul *g.gui*, který zobrazí komplexní grafické prostředí, ve kterém lze spravovat data a pouštět příkazy výběrem z menu. Člověk tak pro běžnou práci nemusí přijít s příkazovou řádkou do kontaktu. Pro komplikovanější výpočty či pro vědeckou práci může být ale její využití nezbytné.

Grafické rozhraní se skládá ze 2 oken, jedno pro zobrazování map a druhé pro správu snímků a nastavení. V okně pro zobrazování lze data procházet, přibližovat a anotovat. Ve druhém okně pak může uživatel spravovat načtené snímky a zpracovávat je pomocí různých nástrojů. Tento přístup se dvěma okny byl zvolen i ve zde popisovaném programu, poněvadž to umožňuje zobrazovat data na větší ploše. Díky zmíněné modularitě může člověk využít ale i jiná grafická rozhraní. Jeden z takových je QGIS. Jedná se o samostatný, také velice používaný GIS systém, který může být spuštěn v GRASS prostředí jako GUI modul. Může ale samozřejmě fungovat i samostatně. Detailněji bude popsán v jednom z dalších odstavců.

## 2.4.2 Alternativa k současným GIS

V důsledku UNIXové filozofie je ale stále vše řízeno pouze jako posloupnost příkazů. Vybranou část dat zpracuje každý příkaz vždy jako celek. Chce-li uživatel při zpracovávání dat měnit nějaký parametr a přitom sledovat dopady této změny, je nucen při každém spuštění příkazu zpracovávat data znovu. Přitom musí generovat nové snímky s patřičnou změnou, aby je mohl následně porovnat. Vzhledem k tomu, že kupříkladu snímky ze satelitu Landsat 8 mají 120 MB, některé dokonce 450 MB, mohou být podobné operace dosti časově a paměťově náročné. Cílem této práce je navrhnout a implementovat systém, který bude klást důraz na rychlejší a efektivnější interakci s uživatelem. Program nebude řízen jako posloupnost příkazů, ale kontinuálně bude zobrazovat a zpracovávat data na základě uživatelských instrukcí. Tento přístup nevede k takové univerzalitě, jakou můžeme vidět v GRASS systému. Zato by mohl řešit efektivněji a uživatelsky přívětivěji některé vybrané úlohy, jako je právě problém klasifikace. A to nejen běžné klasifikace na jednom snímku, ale i klasifikace, kde trénovací data pochází z více snímků. Konkrétně je ve zde popisovaném programu implementováno a srovnáno několik klasifikačních algoritmů, které provádějí klasifikaci na základě vývoje oblastí v určitém časovém rozmezí (například jednoho roku).

Další filozofií, kterou se zde popisovaná aplikace snaží řídit, je takzvané „on demand“ vyhodnocování. Jedná se o princip, kdy jsou načítána a zpracovávána jen ta data, která jsou nutná pro zobrazení výsledku uživateli. Toto je v systému Grass, stejně tak ve většině ostatních GIS systémů, téměř neproveditelné. Součástí aplikace je i automatické skládání vybraných snímků do RGB a rozdělení aplikace do několika logických celků, kde každý běží v samostatném vlákne.

Výše zmíněné vlastnosti mohou ušetřit paměť i výpočetní výkon. Na druhou stranu je důsledkem větší provázanost kódu, kdy jednotlivé moduly musí spolu úzce spolupracovat, aby mohly odpovídajícím způsobem reagovat. To může vést k horší udržitelnosti kódu a tím i případně k horší modularitě.

Logicky se z časových důvodů daných možnostmi bakalářské práce nemůže tato aplikace rovnat svou velikostí a množstvím funkcí moderním GIS systémům. Podporovaná data jsou pouze z projektu Landsat a možnosti anotace dat také nejsou dostatečně univerzální a přenositelné na ostatní platformy. Aplikaci je třeba brát spíše jako demonstrativní, za účelem otestování a ověření funkčnosti zde navrhovaného systému. Způsob ovládání a zpracování dat je ale navržen tak, aby v případě dalšího rozvoje, mohla být tato aplikace rozšířena a dovedena do komplexního GIS systémů.

## 2.4.3 QGIS

Kromě Grass GIS je také oblíbený například systém QGIS [16]. Jedná se o mnohem novější systém, který je napsaný v jazyce C++ s využitím knihovny Qt. Knihovnu Qt využívá i aplikace popisovaná v této práci a její využití, výhody a nevýhody jsou popsány v kapitole 5. Na rozdíl od Grass GIS nemá takové možnosti rozšiřitelnosti a modularity. Zaměřený je spíše na přívětivé uživatelské rozhraní a jednoduchost ovládání. Jinak je ale QGIS zcela plnohodnotný GIS systém umožňující komplexní zpracování, analýzu i vizualizaci satelitních dat.

Aplikace se skládá pouze z jednoho okna, které slouží k veškeré práci s daty. V principu funguje ale velice podobně jako grafické uživatelské rozhraní Grass

systému. Dokonce může být spuštěn buď s Grass pluginem, který emuluje jeho příkazovou řádku, nebo přímo jako GUI modul v systému Grass. Ovládání je v tom případě stejné, pouze používá pro vykonávání jednotlivých funkcí Grass moduly.

#### 2.4.4 gvSIG

Jako Grass modul lze spustit i další systém gvSIG [17], který je napsaný v Javě. Opět jako QGIS je gvSIG zcela plnohodnotný GIS systém. Od QGIS se liší v některých grafických prvcích, jako je třeba navigační tabulka, a dále v různých rozšiřujících funkcích, jako je ku příkladu podpora 3D dat či některé algoritmy. Tím, že je gvSIG napsaný v Javě, má i jistou výhodu v přenositelnosti na jiná zařízení, jako jsou například mobilní zařízení.

Oba dva systémy QGIS i gvSIG vznikly do jisté míry jako alternativa ke Grass GIS. Jejich smysl je spíše usnadnit práci s GIS daty běžným lidem, kteří potřebují data především vizualizovat, nikoli provádět na nich klasifikaci. I přesto lze vždy přidat některé moduly, které klasifikování dat umožní, nemají ale takové možnosti jako jsou právě u zmiňovaného Grass GIS systému.

#### 2.4.5 SAGA

Poměrně velkou podporu klasifikace má například nástroj SAGA GIS [18]. Jedná se o německý projekt zahájený v roce 2004, který je používán i v České republice. Dostupný je ve třech jazycích – angličtině, němčině a češtině. Název SAGA je anglická zkratka, která v češtině znamená Systém pro automatickou geovědeckou analýzu. Jak název napovídá, je tento systém určen zejména pro geovědecké výpočty a vytváření automatických skriptů. Při tom si ale snaží zachovat co nej-jednodušší grafické rozhraní vhodné i pro neobornou veřejnost.

Aplikace je silně modulární a pro téměř každou operaci je nezbytné přidat nějaký modul. Základ aplikace má pouze 10 MB. Bez přidaných modulů ale nelze data ani načítat. Doinstalovat lze například tyto moduly: Práce se soubory, Projekce, Vektorové nástroje, Analýza terénu, Klasifikace, Geo statistika a další. Aplikace má navíc rozhraní jazyka R, ve kterém lze pouštět některé moduly a z výsledků provádět různé statistické výpočty.

Grafické uživatelské rozhraní je koncipováno jako jedno okno, kde střední část okna slouží jako pracovní plocha pro vizualizaci dat. V ní je pro každý zobrazovaný snímek spuštěné samostatné vnitřní okno. Na okrajích se pak nachází panely pro nastavení, práci s daty a adresáři a různé druhy analýzy.

## 3. Motivace a analýza problému

### 3.1 Statistika využití Landsat dat

Výběr dat z projektu Landsat nebyl zdaleka náhodný. Svým vybavením, pokrytím a nepřetržitým fungováním dávají spolehlivý a velice cenný zdroj informací o povrchu Země. Satelitní snímky obecně zasahují do spousty lidských oborů. Často se člověk s nimi setká ve formě různých map nebo při předpovědi počasí. Využití mají ale mnohem širší a to jak v komerčních oborech, tak v oblastech jako je ochrana životního prostředí, plánování rozvoje měst a obcí, ochrana zdraví, v otázkách národní bezpečnosti, vzdělání, při hledání hornin a přírodních zdrojů a v mnoha dalších.

Projekt Landsat původně vznikl pro vědecké a vojenské účely a toto zaměření si drží dodnes. V roce 2011 provedla organizace USGS rozsáhlý průzkum [19], kde zkoumala nejčastější využití dat mezi lidmi pracující s Landsat daty. Průzkum byl zaměřený především na obyvatele USA, obsahuje ale data z celého světa. Suverénně nejčastějším využitím s více než 40 % bylo vědecké zkoumání spojené s životním prostředím, jako je biodiverzita, změna klimatu, ekologie, geologie, vodní zdroje a další. Další pak byly analýza a využití zemského povrchu se 17 %, plánování rozvoje měst a obcí s 11 % a poté následovalo vzdělání a zemědělství oba s 8 %.

Vědecké uplatnění dokládá i srovnání organizací, ze kterých tito uživatelé přichází. Celých 33 % uživatelů jsou lidé z akademických institucí, zatímco soukromá sféra se na využití podílí jen v 18 %. Dalším častým uživatelem těchto dat jsou vládní organizace, kde na federální úrovni se organizace podílí na využití v 17 %, na státní v 16 % a lokální vlády mají 10 % využití. Současně je třeba zmínit, že naprostá většina uživatelů pochází ze Spojených států a údaje v různých částech světa jsou tak velice rozdílné.

### 3.2 Mraky a další negativní vlivy

V důsledku relativně dlouhého času, který satelitu trvá, než se vrátí na původní pozici, nejsou Landsat data vhodná například pro předpovídání počasí. Detekce a analýza mraků má ale také velké uplatnění. Není to ani tak proto, že by byly sami něčím zajímavé, ale spíše proto, že nám vadí v detekci a analýze toho, co je pod nimi. Vzniká tak potřeba automaticky detekovat mraky a následně určit o jaký druh mraku se jedná. Některé druhy jsou zcela neprostupné a nezbývá, než se jim vyhnout. Jiné mohou umožňovat částečnou průhlednost, využití těchto dat je ale stejně výrazně omezeno. Velká pozornost je také věnována mrakům ve vysokých nadmořských výškách nazývaných jako Cirry. Na běžných senzorech jsou tyto mraky téměř nedetekovatelné, ovlivňují ale hodnotu naměřených dat pod nimi. V nejnovějším satelitu Landsat 8 byl proto přidán nový senzor s názvem Cirrus speciálně určený pro detekci těchto mraků.

Potlačení vlivu mraků na sledovanou oblast je pak celkem komplikovaná úloha. Výška mrakové vrstvy je velice proměnlivá a je těžké určit, jak moc v daném místě ovlivňují hodnoty pixelů. Kromě samotných mraků způsobují problémy také jejich stíny. Ty jsou často kruhového tvaru a jsou mnohem chladnější, než

okolní část snímku. Mohou tak být snadno zaměněny například s vodní hladinou rybníka. Jejich naměřená hodnota z různých senzorů je v některých případech zcela nerozlišitelná od vodní hladiny a pro jejich detekci je tak nutné využívat spíše příznakové metody, než metody založené na intenzitě.

Podobné problémy, jaké způsobují mraky, vytváří také sněhová pokrývka. Na rozdíl od mraků se ale nachází přímo na zemském povrchu a jejich detekce je tak o poznání snažší. Jejich způsob odražení slunečních paprsků je v různých spektrech charakteristický a nehrozí tak záměna s jiným povrchem. Kromě těchto vlivů může kvalitu snímku také ovlivnit čas pořízení či intenzita a úhel slunečních paprsků. Díky předzpracování jsou ale tyto vlivy částečně potlačeny.

### 3.3 Klasifikace zemského povrchu

Základní klasifikací povrchu bývá odlišení vodní hladiny, lesů, polí a luk a zastavěných oblastí. Lze tak sledovat využití půdy, měřit množství zalesněné a zastavěné půdy a na základě toho plánovat budoucí rozvoj oblasti. Tímto způsobem lze sledovat i vlastníky nemovitostí, zda nevyužívají pozemek k činnostem, které v daném území nejsou povolené.

Sledování satelitních snímků má velký význam také při řešení přírodních či jiných katastrof. Satelitní snímky umožňují prozkoumat rozsáhlou oblast mnohem rychleji, než by to bylo možné s využitím jiných prostředků. Určení rozsahu požárů a jeho postupu může pomoci při evakuaci lidí a koordinaci hasičských složek. Stejně tak při záplavách může být významné znát rozsah zaplavených oblastí. Dále to může být například zmapování oblastí zasažených hurikány, vlnou tsunami či sledování zničených oblastí po válečném konfliktu.

Využití snímků nemusí být samozřejmě jenom mírové. Jedno z největších uplatnění najdou právě ve vojenských a zpravodajských službách. Od detekce tanků a vojenské techniky, až po určování strategických objektů a plánování útoků. Pro tyto účely je ale významné mít data s co největším rozlišením. Rozlišení 30 m projektu Landsat je v tomto případě zcela nedostatečné. Ačkoli satelity Landsat umožňují snímat zemi s mnohem přesnějším rozlišením, veřejnosti jsou z bezpečnostních důvodů dostupná pouze v této kvalitě.

### 3.4 Klasifikace druhů rostlin

S využitím infračerveného a tepelného záření se ale nemusí klasifikační algoritmy omezovat pouze na toto základní dělení. Je známo, že různé plodiny odrážejí světlo v těchto spektrech odlišně. Teoreticky by tak mohlo být možné sledovat, jaké rostliny se na daném území nachází. Dobře je tento jev vidět ku příkladu na jehličnatých a listnatých lesích. V běžném viditelném spektru jsou velice podobné, podíváme-li se ale na les v blízkém infračerveném záření, rozdílů jsou snadno viditelné i pouhým okem. Obzvláště v jarních měsících jsou nové listy mnohem světlejší, než staré jehlice. V zimních měsících jsou listnaté lesy samozřejmě opadané a klasifikace je též snadná.

I různé druhy listnatých lesů se ale od sebe liší a to jak například vyzařováním v různých spektrech, tak například dobou kdy opadávají. Často je nutné důkladně vybírat časové období, ve kterém se začnou jednotlivé druhy lišit. K tomu je pak

také nutné mít k dispozici dostatečné množství vzorových dat, nebo mít možnost porovnat údaje z jiných zdrojů. Velikou výhodou lesního porostu je dále to, že se na jednom místě vyskytují stejné rostliny po mnoho let. Lze tak využít snímky z delšího časového období.

Čím jemnější dělení klasifikačních tříd budeme používat, tím bude ale logicky docházet k větším nepřesnostem a chybám. Existuje mnoho vlivů, které mohou způsobit naprosté zmatení klasifikátoru. Jsou to například stáří lesa, hustota porostu nebo třeba lesní cesty a další zásahy člověka. Lesy jsou navíc často složené z více druhů a konkrétní skladbu tak nelze vůbec určit. Při současném rozlišení, kdy jeden pixel odpovídá oblasti  $30 \times 30$  m, by tak tato detailní klasifikace mohla být dosti chybová.

O něco přívětivější prostředí nabízí pole a louky. Opět je pro klasifikaci podstatné období, ze kterého jsou snímky pořízené. Odlišit pole od louky je snadné v době, kdy je zorané, v jiných měsících může ale naopak být rozlišení téměř nemožné. Jistou výhodou je, že na polích se téměř vždy vysévá pouze jeden druh rostlin. Každé rostliny mají navíc své charakteristiky, které je jednoznačně odlišují. Pro určování rostlin, které se na daném území nachází, tak může být velice výhodné sledovat danou oblast po určitý časový interval, třeba jednoho roku. Vývoj naměřených hodnot pak může jednoznačně odlišit různé rostlinné druhy. Kromě odrážení slunečních paprsků v různých spektrech, může být charakteristická i barva květu, čas výsadby, doba od vysazení po sklizeň a mnoho dalších vlastností, které je možné s výhodou využít. Ke klasifikaci by tak teoreticky mohly stačit pouze biologické znalosti jednotlivých druhů.

### 3.5 Další možnosti detekce

Senzory snímající různé druhy záření lze ale využít i k mnoha dalším účelům. Někteří zemědělci již v současné době využívají satelitního snímkování k diagnostice zda plodina má dostatek vláhy. Opět jsou v tomto případě výhodná především infračervená spektra. Množství vláhy je možné sledovat obecně v rostlinách a najde uplatnění i z hlediska prevence případných požárů. Dále na čerstvě zoraných polích je možné získávat informace o samotné půdě. Barva půdy v různých spektrech může mnohé napovědět o jejím složení a aktuální kvalitě.

Široké možnosti přináší i zkoumání zastavěné plochy. Mnoho budov lze automaticky detekovat díky jejich jedinečným vlastnostem, jako jsou například letiště, nádraží, výrobní haly či třeba parkoviště. Význam pak má také dlouhodobé sledování rozvoje měst pro účely plánování nové zástavby. Jednou z častých potřeb států je sledování, jestli vlastníci zachází s pozemky dle platných právních předpisů. Mohou tak odhalovat černé stavby, nelegální těžení přírodních zdrojů, černé skládky či kontrolovat hranice pozemků.

Komplexnější analýzou by teoreticky mohlo být možné ze satelitních dat odvodit i minerály nacházející se v půdě. Barva povrchu země může mnohé napovědět o stáří a původu hornin, které se zde nachází. Na základě toho lze odvozovat, jaké nerostné suroviny by se v dané lokalitě mohly nacházet. Tyto možnosti detekce dávají i data z projektu Landsat, jak například píše ve své práci i Floyd F Sabins [20]. Vhodným studijním materiálem by mohlo být například čerstvě zorané pole. Ale i země pokrytá vegetací může nést jisté znaky toho, co se nachází pod povrchem. Rostliny jsou ovlivněny chemickým složením půdy a jejich skladba

pak může odrážet hladiny minerálů v půdě. Podobné detekce ale budou pravděpodobně vyžadovat spolupráci s dalšími vědeckými obory, jako jsou chemické a biologické.

## 4. Struktura aplikace

Jak již bylo zmíněno, cílem této práce je navrhnout a otestovat nástroj pro zpracování družicových dat. Aplikace je navržena jako alternativa k současným GIS systémům, není ale účelem implementovat veškeré funkcionality těchto systémů. Je zaměřená pouze na některé oblasti zpracování geografických dat, jako je vizualizace, analýza a klasifikace. A to jak jednotlivých snímků, tak skupiny snímků jedné oblasti, ve které lze sledovat časový vývoj. Dále aplikace pracuje pouze s daty z projektu Landsat. Návrh aplikace ale nebrání v případném rozšíření o další data či funkcionality z běžných GIS systémů.

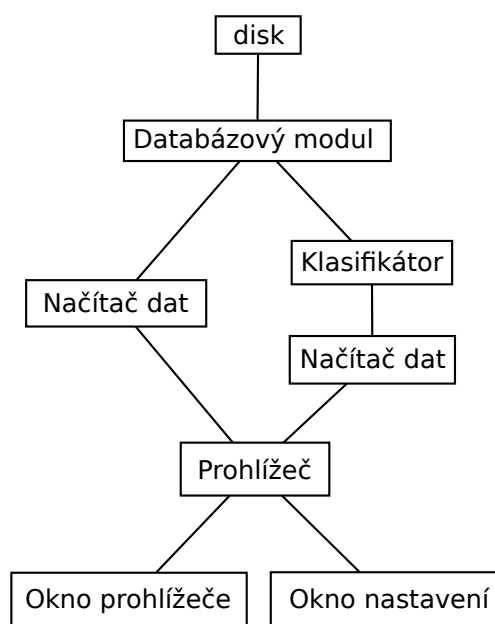
### 4.1 Obecný logický model aplikace

Aplikace je rozdělena na tři hlavní logické celky. První část slouží k načítání, předzpracování a správě satelitních dat a následně jejich poskytování ostatním modulům aplikace. V principu funguje jako databázový systém. Snímky rozřezá na menší oblasti, indexuje a ukládá na disk do adresářové struktury. Dále poskytuje rozhraní pro poskytování jak běžných černobílých snímků jednotlivých senzorů, tak barevných snímků složených do RGB. Druhou částí je prohlížeč. Skládá se ze dvou oken, jedno pro vizualizaci dat a druhé pro nastavení a navigaci v datech. Umožňuje procházet, spravovat snímky a provádět základní analýzu. Dále zobrazuje výsledky klasifikace a umožňuje vybírat a spravovat trénovací data. Poslední částí je klasifikační modul. Ten provádí samotnou klasifikaci a výsledky dává k dispozici prohlížeči. Přípravených je několik základních klasifikačních algoritmů a pomocí programátorského rozhraní lze přidávat další.

Z prohlížeče lze ještě oddělit modul, který se stará o získávání a poskytování dat a správu paměti. Tato část je oddělená především proto, že načítání dat probíhá asynchronně v jiném vlákne než samotná vizualizace. Z toho důvodu je nutný modul, který bude zajišťovat rozhraní mezi databází a prohlížečem a bude se starat o aktualizaci dat, jejich načítání, a posílání načtených dat prohlížeči k vizualizaci. Kromě toho se stará o cachování dat a uvolňování paměti. Podobně vypadá i rozhraní mezi prohlížečem a klasifikátorem. Má na starosti aktualizaci a správu dat, která jsou výsledkem klasifikace. Má stejný účel jako modul mezi databází a prohlížečem a i v principu funguje stejným způsobem. Celkový logický model aplikace lze vidět na obrázku 4.1.

#### 4.1.1 Vlákna

Rozvláknění se ukázalo jako zcela nezbytné pro efektivní fungování aplikace. Velikost zobrazovaných dat se pohybuje ve stovkách MB. Před vizualizací dochází ještě k jejich skládání do RGB a zároveň s tím může probíhat klasifikace snímků. V případě většího množství dat mohou tak tyto operace trvat i desítky vteřin. Zároveň je podstatné, aby se zpětně nenačítala data, která zatím načtená nebyla a uživatel již o jejich vizualizaci ztratil zájem. Obzvláště je to podstatné v situacích, kdy uživatel jen v rychlosti prochází data nebo hledá konkrétní části snímků. V těchto případech by se pak snímky hromadily ve frontě a výrazně by se snižovala odezva aplikace.



Obrázek 4.1: Model logických částí aplikace a jejich propojení

Celkově tak tedy aplikace běží ve třech vláknech. V prvním vlákne je prohlížeč s dialogem pro nastavení. Obstarává veškerou interakci s uživatelem, která tak může probíhat bez jakýchkoli prodlev. V druhém vlákne běží načítač dat, jak byl již popsán dříve. Poslední vlákno má na starosti klasifikaci a posílání jeho výsledků prohlížeči. Samotný databázový modul je pak sdílený a mohou ho využívat všechna vlákna.

## 4.2 Databázový modul

Databázový modul implementuje vlastní formát pro ukládání snímků na disk. Je navržen speciálně pro účely vizualizace a klasifikace. Data jsou ukládána do adresářové struktury a takto vytvořená struktura je dále nazývána jako databáze. Snímky jsou rozřezány na čtverce o pevné velikosti a následně každý čtverec je zvlášť ukládán jako samostatný soubor. Práce s celými snímky by byla neefektivní poněvadž uživatel většinou zobrazuje jen malou část z celkového snímku. Zpracování po čtvercích řeší tento problém a navíc umožňuje snadnější indexaci a správu dat. Pro účely přibližování a oddalování dat jsou navíc předpočítány naškálované čtverce ve třech oddáleních - dvakrát, čtyřikrát a osmkrát zmenšené.

Alternativně by šlo čtverce načítat využitím některých formátů, které by umožňovaly načítat pouze vybrané podoblasti. Formát tiff je právě jedním z nich. Režie spojená s vyřezáváním a indexováním dat by byla poměrně malá, tudíž by to jistě bylo zcela použitelné řešení. Jednou z výhod rozřezání snímků na spoustu menších souborů je ale například to, že lze bez problémů smazat některé nepotřebné čtverce, aniž by bylo narušeno čtení zbylých dat. V případě dat z projektu

Landsat tak mohou být například odstraněny černé oblasti, které nejsou v rozsahu senzoru. Současně odpadá jakákoli režie s vyřezáváním a indexováním čtverců. Nevýhodou je naopak vyšší fragmentace. Každé načtení čtverce vyžaduje samostatné systémové volání, proto nemohou být čtverce příliš malé. Při zkoušení různých rozměrů čtverce se ukázalo, že pokud je velikost hrany menší než přibližně 50 pixelů, začíná se to výrazně projevovat v odezvě aplikace. Velikost hrany čtverce byla nakonec zvolena na 400 pixelů.

### 4.2.1 Formát dat na disku

Každá samostatná databáze obsahuje jeden hlavní soubor s příponou „.ldb“. Ten je zde zaprvé pro informaci, že se v daném adresáři nachází databáze s Landsat daty. Zadruhé obsahuje například zvolenou velikost čtverce v metrech a o další vlastnosti může být případně rozšířen. U tohoto souboru se pak nachází složky s jednotlivými scénami. Každá scéna obsahuje složky se senzory a každý senzor pak obsahuje samotné čtverce. Scéna ještě obsahuje soubor s příponou „.scn“. Jedná se ale pouze o kopii „\_MTL.txt“ souboru, který byl popsán v kapitole 1. Přípona je změněná, aby nemohlo dojít k záměně s původní, ještě nerozřezanou, scénou.

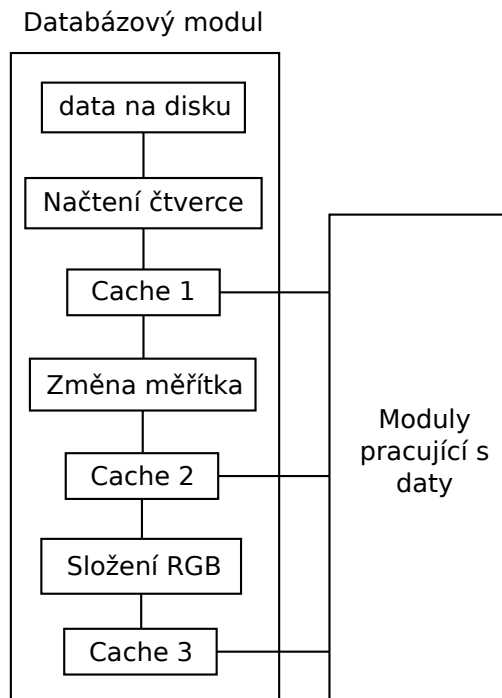
Scény jsou pojmenované stejným způsobem, jaké používá samo USGS. Složky s daty jednotlivých senzorů mají akorát přidanou příponu „\_BAND“. Samotné čtverce jsou v tiff formátu a jejich název je sestaven ze tří čísel oddělených dvojtečkou. První číslo charakterizuje oddálení. 0 je žádné, 1 odpovídá dvojnásobnému zmenšení, 2 čtyřnásobnému, a 3 značí 8 krát zmenšený čtverec. Poté následují projekтивní souřadnice levého horního rohu čtverce.

### 4.2.2 Třístupňové předzpracování vizualizovaných dat

Pro zobrazování dat pak databázový modul předzpracovává data ve třech krocích. Tento proces je zobrazen na obrázku 4.2. Nejprve je vybrán a načten odpovídající čtverec z databáze, který je následně naškálován na zobrazované přiblížení. Tři takovéto snímky pak dají vzniknout výslednému RGB obrázku. Zobrazuje-li uživatel oddálené snímky, ke škálování nedochází a jsou rovnou načteny již naškálované čtverce z disku. Data z každého kroku jsou ukládány do samostatné cache. Ostatní moduly tak mohou k datům přistupovat ve všech třech fázích tohoto předzpracování. Prohlížeč využívá především RGB čtverce, naopak klasifikátor potřebuje používat původní černobílé a nenaškálované čtverce.

V zásadě šlo toto předzpracování provést ještě jiným způsobem. Nejprve načíst čtverec, složit ho do RGB a až poté naškálovat na správné přiblížení. V případě zvětšování čtverce by byl tento postup výhodnější, poněvadž skládání do RGB by probíhalo na menších čtvercích. Dále by se škáloval pouze jeden RGB obrázek místo čtyř černobílých. V principu by ale obě operace měly zabrat srovnatelný čas, poněvadž RGB obrázek má tři krát větší velikost než ten černobílý. Z analogických důvodů je naopak při zmenšování čtverce tento postup o něco pomalejší. Navíc by v tomto případě nebylo možné využít již předpočítaných zmenšených čtverců. Z toho důvodu byl tedy zvolen postup nejprve provést naškálování a poté až skládání do RGB.

Ideálním řešením by bylo měnit pořadí zpracování v závislosti na tom, jestli se jedná o zvětšování či zmenšování. Přinášelo by to ale několik dalších problémů.



Obrázek 4.2: Třístupňový model předzpracování, cachování jednotlivých vrstev a rozhraní pro ostatní části aplikace.

Přístup k naškálovaným černobílým čtvercům by byl jen u oddálení, navíc by se tomu musely nějak přizpůsobit cache. Proveditelné by to ale být mělo a mohl by to tak být případný námět pro další vylepšení aplikace.

Kromě rozhraní pro získávání samotných čtverců je k dispozici i rozhraní pro ovládání cache. Lze pustit jejich promazání, kdy jsou odstraňována nejstarší data dokud je velikost cache větší než zvolená hodnota, případně je lze vyčistit úplně. Dále lze ještě získávat další informace o scénách, jako je třeba datum pořízení a další.

### 4.3 Prohlížeč

Dalším modulem je prohlížeč. Ovládání a navigace v datech bude detailněji popsáno v kapitole o grafickém uživatelském rozhraní. Zde se zaměřím spíše na jeho strukturu, rozhraní a komunikaci s ostatními částmi aplikace.

Prohlížeči je neustále přiřazena pozice na Zemi, kterou zrovna vizualizuje. Tato pozice jsou souřadnice v metrech v projektivní rovině, která již byla popsána v kapitole 1. Této pozici odpovídá levý horní roh prohlížeče. Pro zobrazování je ale zaokrouhlena na 30 metrů, což standardně odpovídá velikosti jednoho pixelu. Kromě toho si aplikace pamatuje path a row, které určují pozici scény dle WRS 2 systému. Všechny scény v tomto umístění jsou seřazené podle data vzniku a aplikace si udržuje pořadové číslo scény, která je aktuálně zobrazovaná.

### 4.3.1 Načítání dat

K aktualizaci prohlížeče dochází při libovolném pohybu v datech. Může to být například změna globálních souřadnic, změna zobrazované scény nebo jiný výběr senzorů, ze kterých jsou skládány RGB snímky. Dále je to změna rozměrů okna a jeho minimalizace a maximalizace.

Při každé aktualizaci prohlížeče jsou spočítány souřadnice všech čtverců, které zasahují do zobrazované oblasti. Jsou z nich vytvořeny indexy jednoznačně definující konkrétní čtverce a ty jsou pak poslány modulu pro načítání dat. Tento modul postupně prochází indexy a načítá čtverce z databáze. Ve chvíli, kdy je nějaký čtverec načtený, je okamžitě poslán zpět prohlížeči, který jej zobrazí. Současně je před každým načtením čtverce znovu zkontrolováno, zda je jeho načtení stále vyžadováno. Nehromadí se tak ve frontě požadavky, které již přestaly být aktuální.

Po každém načtení jedné skupiny čtverců volá načítač funkci na promazání cache, aby data nepřekračovala stanovený limit paměti. Poněvadž je tato rutina až za načítáním dat, může v průběhu načítání dojít k překročení tohoto limitu. Velikost takového překročení se ale pohybuje maximálně v řádech několika desítek megabytů, což by u dnešních počítačů nemělo hrát významnou roli. Správa dat je koncipovaná tak, že paměť je uvolněna až ve chvíli, kdy na daný čtverec zanikne poslední ukazatel. Nezbytná data pro zobrazení aktuálně vizualizovaných dat jsou tak vždy držena v paměti, bez ohledu na paměťová omezení.

Naprosto stejně funguje i modul pro načítání klasifikovaných dat. Ta jsou zobrazována nad vizualovaná data. Stejně jako originální data jsou tak klasifikované čtverce počítány až ve chvíli, kdy jsou nezbytné pro potřeby zobrazení.

### 4.3.2 Další vlastnosti prohlížeče

Další funkcí prohlížeče je možnost anotovat data. Uživatel může vybírat oblasti obdélníkového tvaru patřící do stejných tříd a vytvářet tak trénovací data pro klasifikační algoritmy. O těch se čtenář více dozví v následující kapitole. Prohlížeč dále umožňuje tato trénovací data pojmenovávat, spravovat a nastavovat barvu, kterou bude tato třída reprezentována při klasifikaci. Trénovací data lze i exportovat do speciálního formátu s příponou „.smp“. Tato funkcionalita je zatím omezená pouze pro účely klasifikace. Do budoucna se ale počítá s případným rozšířením o další druhy anotace.

Ve zmíněném „.smp“ formátu se zaznamenává název třídy, její barva a poté vybrané obdélníky v projektivních souřadnicích. Jednotlivé třídy tak nejsou vázány na data, ze kterých byly vytvořené, ale pouze na geografické souřadnice označující jejich pozici na zemi. Tento způsob byl zvolen zcela záměrně, poněvadž jedním ze smyslů této aplikace je umožnit srovnávání klasifikačních algoritmů pracujících s daty, u kterých lze pozorovat vývoj v čase. A nás tedy ani tak nezajímají objekty, které se nacházely na jednom konkrétním snímku, ale spíše se zajímáme o to, k čemu byla daná oblast v průběhu daného časového úseku využívána. Příkladem tak může být rozpoznávání polí, zástavby či různých druhů lesů. Zároveň to ale nijak nebrání v tom použít stejným způsobem klasifikační algoritmy i pro samostatné scény.

K dispozici je dále možnost vynášet časový vývoj daných trénovacích dat do grafu. Vynášení do grafu probíhá opět jen s vybranými scénami a zobrazování

vypadá tak, že pro každý senzor je vytvořena samostatná křivka charakterizující vývoj hodnoty daného senzoru v čase. Jedna hodnota grafu je počítána jako průměrný odstín dané třídy v jednom snímku senzoru. Uplatnění najde takovýto druh analýzy například v případě, že tvoříme účelově klasifikační algoritmus se samostatným kódem pro vybrané třídy.

### 4.3.3 Komunikace mezi moduly

Prohlížeč de facto také slouží jako centrální modul, který zajišťuje propojení a komunikaci mezi ostatními částmi aplikace. Především zajišťuje aktualizaci jednotlivých částí, dojde-li ke změně v části jiné. Kromě aktualizace prohlížených dat se stará o účinky veškerého nastavení, načítání nových dat či upozornění klasifikátoru, že je nutné algoritmy znovu natrénovat. Konkrétní řešení komunikace bude popsán v následující kapitole.

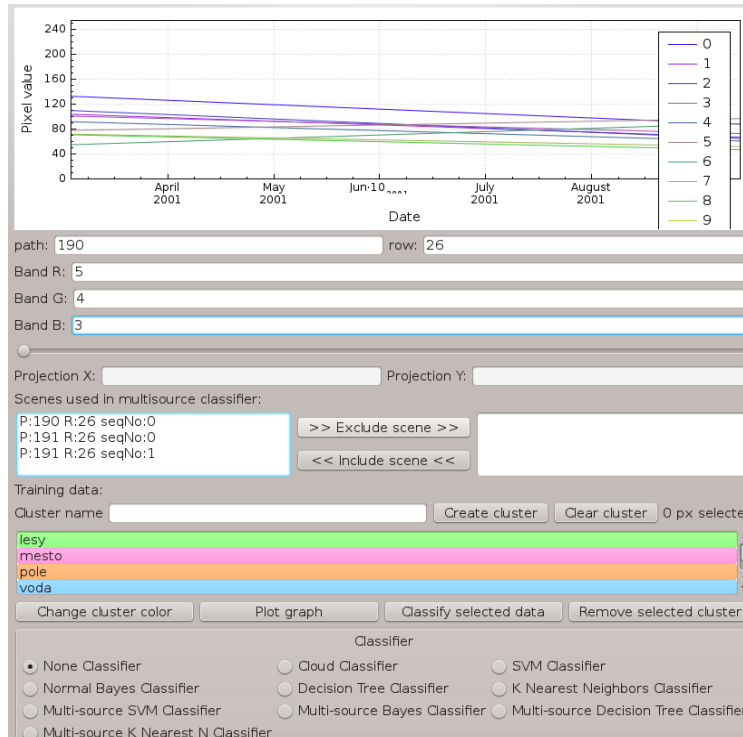
Celkové propojení do jisté míry snižuje nezávislost jednotlivých částí. Snižuje to tak modularitu, což může komplikovat případná zobecnění systému. Na druhou stranu to umožňuje vytvořit zcela interaktivní aplikaci, která pro vybrané úlohy zefektivňuje práci. Uživatel navíc může pracovat s částečnými výsledky již v průběhu načítání. Tyto vlastností jsou v ostatních GIS systémech jen velice obtížně dosažitelné.

## 4.4 Klasifikační modul

Poslední modul je klasifikátor, který na základě trénovacích dat provádí klasifikaci. Samotný modul žádné algoritmy neimplementuje, ale vytváří pouze prostředí pro využívání těchto algoritmů. Klasifikátor se stará o výběr správného algoritmu a následné vytvoření barevného obrázku, který jednotlivým pixelům přiřadí odpovídající barvu třídy, do které patří. Klasifikační algoritmy je možné přidávat pomocí jednotného programátorského rozhraní popsaného v kapitole 5.

Při prvním zavolání nebo při změně vstupních dat probíhá natrénování klasifikačního algoritmu. Klasifikace pak probíhá po jednotlivých čtvercích. Pro každý pixel čtverce vybere algoritmus třídu, do které patří. Následně mu přiřadí barvu odpovídající jeho třídě, čímž vznikne nový barevný čtverec. Kromě tříd zvolených uživatelem je v aplikaci vyhrazena ještě speciální třída pro pixely, které jsou mimo scénu, nebo které jsou pokryty mraky. Té je vždy přiřazena bílá barva. Další speciální třída je využita v případech, že klasifikační algoritmus nebyl schopen daná data klasifikovat, například z důvodu nedostatečného množství trénovacích dat. Pixely patřící do této třídy se pak vůbec nezobrazují.

Po vytvoření čtverce dochází k jeho naškálování a následnému zobrazení v nové vrstvě nad RGB snímky senzorů. Ke cachování dochází tentokrát pouze u samotných klasifikovaných čtverců, nikoli u čtverců naškálovaných. Důvodem je fakt, že doba škálování je celkem zanedbatelná v porovnání s dobou samotné klasifikace. Případné zrychlení by tak uživatel téměř nepocítil a zbytečně by to spotřebovávalo paměť.



Obrázek 4.3: Dialog pro nastavení a ovládání zobrazovaných dat

## 4.5 Grafické uživatelské rozhraní

Aplikace se skládá ze dvou oken, hlavní okno pro zobrazování dat a navigaci v nich a druhé okno pro nastavení, analýzu a klasifikaci dat. V hlavním okně je menu, které obsahuje několik položek převážně týkající se práce s databází a s trénovacími daty. První dvě položky umožňují vytvořit novou databázi nebo načíst dříve vytvořenou. Další položkou lze přidat novou scénu. To lze udělat pouze tehdy, je-li nějaká databáze načtená. Po nich následují další dvě pro ukládání a načítání trénovacích dat. Vnitřek hlavního okna slouží už výhradně pro procházení dat. To lze provádět pomocí následujících příkazů:

- Stisknutí a tažení myši slouží pro pohyb globálně v prostoru v rámci projekčních souřadnic.
- Pohyb kolečkem myši lze použít pro přibližování a oddalování. K dispozici jsou tři stupně oddálení i přiblížení, 2 krát, 4 krát a 8 krát.
- Během stisknuté klávesy Ctrl je možný výběr oblastí pomocí stisknutí tlačítka myši a následným tažením.
- Během stisknuté klávesy Ctrl pohyb kolečka myši slouží ke změně scény na časově následující či předchozí scénu

Dialog pro nastavení dává některé další možnosti ovládání prohlížeče. Umožňuje pracovat s trénovacími daty, zobrazovat vývoj hodnot třídy v grafu či vybírat klasifikační algoritmy. Popisovaný dialog je vidět na obrázku 4.3. Prvky, které v něm lze najít jsou následující:

- Graf pro vykreslování vývoje hodnot dané třídy. Jeho fungování je popsáno v podkapitole níže.
- Dvě textová pole pro nastavení path a row. Při změně je automaticky vybrána první scéna v časové řadě.
- Tři textová pole slouží k výběru spekter. Ty určují jaká spektra budou použita pro složení zobrazovaného RGB obrázku.
- Jezdec umožňuje pohyb v časové ose. Jedná se o alternativu k stisknuté klávese Ctrl a pohybu kolečka myši.
- Pod jezdcem jsou souřadnice v projektní rovině. Jejich funkce jsou pouze informativní a zobrazují souřadnice, na kterých bylo naposledy stisknuté tlačítko myši.

Zbývající prvky slouží ke klasifikaci:

- Prvek pro výběr scén. Vybrané scény používají klasifikační algoritmy, které pracují na více scénách současně. V levé části jsou scény, které budou zahrnuty do klasifikace a v pravé části jsou scény, které zahrnuty nebudou. Scény lze přidávat a odebírat pomocí dvou tlačítek uprostřed.
- Textové pole pro zadání názvu trénovací třídy. Vedle něj se nachází tlačítko, které uloží všechny vybrané oblasti do třídy s tímto názvem. Současně je k ní vygenerována náhodná barva a třída je zařazena do listu ostatních tříd. Touto barvou budou po klasifikaci zvýrazněny všechny body, které budou patřit do této třídy. Barvu je možné změnit tlačítkem popsaným níže. Poslední tlačítko je určeno pro smazání všech neuložených, aktuálně vybraných oblastí.
- Další prvek je list všech klasifikačních tříd se čtyřmi tlačítky. Každá třída je zvýrazněná barvou, kterou má přiřazenou. Označené třídě lze změnit barvu kliknutím na první tlačítko. Druhé tlačítko vezme vybranou třídu a vykreslí graf zobrazující vývoj vybrané třídy v čase. Třetí tlačítko souží k vyhodnocení oblastí načtených ze souboru právě vybraným klasifikátorem a poslední tlačítko odstraní třídu ze seznamu.
- Prvky ve spodní části dialogu slouží pro výběr klasifikačního algoritmu. Při výběru se data automaticky začnou klasifikovat a výsledek zobrazovat v Prohlížeči.

#### 4.5.1 Vynášení průměrných hodnot třídy do grafu

Graf slouží pro vykreslování vývoje vybrané trénovací množiny v čase. Pro vykreslení musí být vybrána jedna třída z listu trénovacích tříd a stisknuto tlačítko „Plot graph“. Po jeho stisknutí jsou vybrány všechny body, které patří do zvolené třídy. Jednotlivým scénám je následně pro každý snímek zvlášť spočítána průměrná hodnota pixelů v těchto bodech. Výsledkem je tak pro každou scénu jeden vektor, kde jedna hodnota vektoru odpovídá průměru hodnot zvolené třídy v jednom snímku.

$k$ -tá křivka grafu je pak sestavena z průměrů, které byly vytvořené z  $k$ -tých snímků všech scén. Osu  $X$  tvoří den pořízení jednotlivých scén a osu  $Y$  hodnoty průměrů, kde každý je v intervalu od 0 do 255. Celkově tak pro každý senzor satelitu vznikne jedna křivka, která vyjadřuje vývoj průměrných hodnot dané třídy pořízených jedním konkrétním senzorem. Jednotlivé křivky jsou barevně odlišeny a jejich maximální počet je 11, poněvadž poslední dvanáctý snímek projektu Landsat 8 nebyl vytvořen žádným senzorem.

#### 4.5.2 Testování úspěšnosti klasifikátoru

Tlačítko s názvem „Classify samples from file“ je určeno pro otestování klasifikačního algoritmu na referenčních datech. Referenční data slouží pro porovnání výsledků klasifikačního algoritmu se skutečností a jsou vytvářena stejným způsobem, jakým jsou tvořena i trénovací data. Stejně jako v trénovacích datech i zde každé třídě odpovídají oblasti bodů, které uživatel ručně označil a o kterých si je jist, že do dané třídy patří. Referenční data musí být uložena v souboru s příponou „.smp“ a názvy tříd musí být shodné s názvy, které jsou použité pro trénovací data. Při stisknutí tlačítka musí být také vybrán jeden algoritmus, na kterém bude prováděno testování.

Po stisknutí je uživatel vyzván k vybrání referenčních dat. Po jejich vybrání vybere uživatel soubor, do kterého bude uloženo vyhodnocení testu. Tento soubor má příponu „.out“. Následně je na všech bodech z referenčních dat provedena standardní klasifikace. Každá třída referenčních dat je vyhodnocena zvlášť a výsledek je zapsán do souboru v následujícím formátu:

- Název algoritmu, kterým se sám prezentuje v aplikaci. Položka je zapsána jako hodnota atributu „ALGORITHM“.
- Druhý řádek nese označení „SCENES“ a obsahuje identifikátory scén, na kterých byla prováděna klasifikace. Jednotlivé scény jsou oddělené čárkou.
- Následuje vyhodnocení jednotlivých tříd. Toto vyhodnocení začíná vždy názvem „CLUSTERS\_BEGIN“ a končí „CLUSTERS\_END“. V každé řádce je vypsána úspěšnost jedné třídy ve tvaru: „ $N : S : C$ “, kde  $N$  značí název trénovací třídy,  $S$  je počet správně klasifikovaných bodů a  $C$  je počet všech bodů této třídy referenčních dat.

# 5. Použité algoritmy a datové struktury

## 5.1 Použité technologie

Aplikace je vytvořena v jazyce C++ s využitím standardních knihoven STL [21]. Tento jazyk byl zvolen především proto, že aplikace bude pracovat s velkým objemem dat a C++ umožňuje zcela kontrolovat a spravovat množství využívané paměti. Pro tvorbu grafického uživatelského rozhraní byla zvolena knihovna Qt [22]. U této knihovny byla výhodou možnost využívat její nativní způsob komunikace mezi objekty pomocí signálů a slotů. Byly využity zejména pro komunikaci a synchronizaci mezi vlákny. Signály a sloty navíc umožňují snadno řešit některé základní vícevláknové konstrukce, jako je kupříkladu producent-konzument. Detailněji bude tato technologie a její využití popsána v jedné z následujících podkapitol.

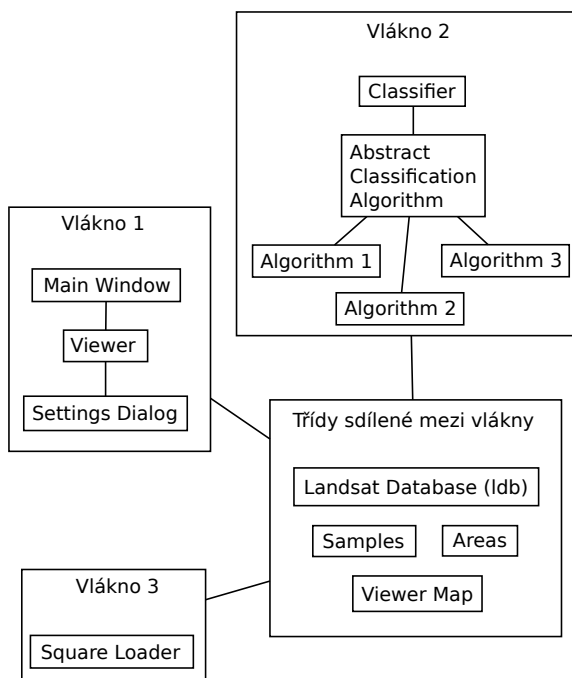
Pro načítání snímků byla dále využita knihovna libtiff [23]. Knihovna Qt si ce zvládá pracovat se soubory ve formátu tiff, neexistuje ale v ní žádný formát, kterým by mohla reprezentovat snímky se vzorkováním 16-ti bitů na pixel. Toto vzorkování mají snímky ze satelitu Landsat 8, proto bylo nutné pro jejich načítání využít externí knihovnu. Pro reprezentaci dat v Qt je pak využito pouze 8 nejvýznamějších bitů pixelu.

Pro zobrazování grafů byla zvolena knihovna QCustomPlot [24]. Poslední externí knihovnou, použitou v této aplikaci, je knihovna OpenCV [25]. Z ní využívám modul Machine Learning pro účely klasifikace. Veškeré zde uvedené klasifikační algoritmy využívají tuto knihovnu.

## 5.2 Struktura tříd aplikace

Logickému modelu, který byl představen v minulé kapitole, odpovídá i struktura tříd v jazyce C++. Jednotlivé třídy a jejich závislosti jsou zobrazeny na obrázku 5.1. Od logického modelu se částečně liší klasifikátor, kde je do jedné třídy sloučená jak samotná klasifikace, tak procedura pro aktualizaci dat a jejich posílání prohlížeči. Třídy běžící v samostatných vláknech jsou pak shodné s jednotlivými částmi logického modulu. Pouze je zde navíc několik sdílených tříd, které jsou následující:

- *LandsatDatabase* odpovídá databázovému modulu.
- *Areas* jsou veškeré, zatím neuložené, oblasti.
- *Samples* obsahuje všechny trénovací třídy. Pro každou třídu si udržuje množinu uživatelem vybraných oblastí.
- *ViewerMap* je třída pro posílání dat prohlížeči. Je ve dvou instancích, jednu sdílí *SquareLoader* a *Viewer* a ukládají se zde načtené čtverce z databáze, druhá je pro dvojici *Classifier* a *Viewer* a nese čtverce, které jsou výsledkem klasifikace.



Obrázek 5.1: Struktura tříd a jejich propojení

Kromě těchto tříd je zde ještě několik pomocných tříd. Jsou to třídy *SquareImage*, která reprezentuje jeden obecný čtverec, *SquareCache* pro cachování čtverců a *MapKey* jako obecný klíč pro indexování čtverců. A to jak barevných čtverců načítaných z disku, tak těch, které jsou výsledkem klasifikace.

### 5.2.1 Index čtverce

Index čtverce je obecně  $k$ -tice čísel, která ho jednoznačně definuje. Čtverec vyříznutý z jednoho snímku obsahuje následující informace: Path, row, pořadové číslo scény v uspořádání scén podle času, projektivní souřadnice  $x$  a  $y$  a pořadové číslo senzoru, který daný snímek pořídil. Naškálovaný čtverec má kromě těchto informací ještě navíc číslo definující stupeň přiblížení. RGB čtverec pak obsahuje navíc čísla tří senzorů, ze kterých je složen.

Čtverec, který je výsledkem klasifikačního algoritmu pracujícího s pouze jednou scénou, je identifikován podobně jako čtverec z jednoho snímku. Pouze je zde pořadové číslo senzoru nahrazeno pořadovým číslem klasifikačního algoritmu. Čtverec klasifikačního algoritmu využívající více scén identifikují pouze souřadnice  $x$ ,  $y$  a pořadové číslo algoritmu.

## 5.3 Asynchronní načítání čtverců

### 5.3.1 Signály a sloty v Qt

Veškerá komunikace mezi vlákny je vyřešena pomocí Qt signálů a slotů. Qt je využívá pro notifikaci ostatních objektů o tom, že nastala nějaká událost, jako je třeba stisknutí tlačítka či zadání nějakého textu. Signály a sloty nemusí být implementované pouze pro grafické prvky knihovny, ale lze vytvořit i vlastní pro řešení komunikace a synchronizace mezi vlákny. Aby třída byla schopná emitovat a zpracovávat signály, stačí aby dědila z třídy `QObject` – obecného předka všech tříd knihovny Qt. V těchto třídách lze vytvořit speciální funkce označené jako sloty a signály. Odpovídající signály a sloty mohou být propojeny a zavoláním signálu se pak vyvolá vykonání slotu. Při tom nezáleží na tom, zda třídy běží ve společném vlákně nebo každá ve svém.

K vykonání slotu ale nedochází okamžitě. Je-li emitován signál, je zařazen příslušnému vlákně do fronty pro zpracování. Vlákna běží ve smyčce a postupně zpracovávají signály v pořadí, ve kterém přišly. Při volání signálů lze navíc jako parametr předávat proměnné. Při posílání dat tímto způsobem ale dochází k celkové kopii dat, tudíž je výhodnější využít pro přenos velkého množství dat sdílených datových struktur.

### 5.3.2 Načítání dat

Výhodně tak lze signály a sloty využít právě k asynchronnímu načítání dat. Sdílenou datovou strukturou, ve které jsou data přenášena je výše zmíněná třída *ViewerMap*. Ta obsahuje mapu, kde klíčem je index čtverce a hodnotou ukazatel na čtverec. Aktualizaci mapy provádí jak prohlížeč tak modul pro načítání dat. Prohlížeč mění indexy čtverců a načítač má na starosti doplňování dat.

Během aktualizace jsou spočítány souřadnice všech čtverců, které zasahují do zobrazované oblasti. Z nich jsou vytvořeny klíče, kterými je mapa doplněna. Novým klíčům, které se v mapě zatím nenachází, je přiřazen nulový ukazatel. Klíče, které jsou v mapě, ale nejsou v nové množině klíčů jsou odstraněny. Zbylé položky v mapě zůstávají zachovány i se svými daty.

Následně je poslán signál o tom, že došlo k aktualizaci dat, který zachytí *Square Loader*. Ten postupně prochází položky v mapě a pokud má klíč přiřazený nulový ukazatel, tak data načte a doplní. Položky odstraněné ještě před doplněním tak nebudou zbytečně zpětně načítána. Při každém načtení čtverce je pak okamžitě odeslán signál prohlížeči na překreslení. Prohlížeč pak zobrazí veškerá zatím dostupná data ze společné mapy. Je-li zapnutý některý z klasifikačních algoritmů, dochází ke stejnému procesu i mezi prohlížečem a klasifikátorem.

## 5.4 Cachování

Pro efektivní práci s daty je nezbytné maximálně využívat dříve spočítaných dat. Veškerá data, která jsou jednou spočítána, by měla být cachována pro účely opětovného použití. Tvorba RGB snímků se skládá ze tří stupňů předzpracování, kde každým projde jiné množství dat. Jednotlivé operace jsou navíc různě výpočetně náročné. Velikosti cache by tak měly jednotlivým vrstvám odpovídat. Dále by

v ideálním případě mělo promazávání cache probíhat od nejstarších dat po ty nejnovější. Tohoto všeho je docíleno pomocí dvou datových struktur, které tvoří každou cache.

### 5.4.1 Datové struktury uvnitř cache

První datovou strukturou je hash mapa, která slouží pro vyhledávání a ukládání dat. Klíčem je zde opět index definující čtverec a hodnotou je ukazatel na čtverec. Hashování klíče probíhá ve dvou krocích. První krok zahashuje k-tici čísel z indexu na jedno celé číslo. To je provedeno pomocí hashovací funkce známé pod jménem `djb2` [26]. Nejčastěji je tato funkce využívána k hashování řetězců a právě uspořádaná k-tice celých čísel je pouze jiná reprezentace řetězce. Výsledné číslo je pak hashováno standardní hashovací funkcí knihovny STL.

Druhou strukturou je list čtverců. V něm jsou čtverce uloženy v pořadí, ve kterém k nim bylo v minulosti přistoupeno. Každá instance čtverce je uložena v obou strukturách, jak v tomto listu tak v hash mapě. Nedochozí tím k duplikaci dat a zároveň je díky tomu možné k nalezení daného čtverce využít pouze jednu z těchto dvou datových struktur. S takto nalezeným čtvercem je pak možné pracovat v obou strukturách současně.

Aby bylo možné v konstantním čase odebírat z listu čtverec, který byl nalezený pomocí indexu v hash mapě, nelze k implementaci tohoto listu využít standardní knihovnu STL. Odkazy na následníka a předchůdce je nutné implementovat přímo ve třídě čtverce. Každý čtverec má tedy kromě ukazatele na samotný obrázek ještě dva ukazatele, jeden na svého předchůdce a druhý na následníka. Aby šlo i obráceně odebírat čtverec z hash mapy, který byl dříve nalezený pomocí listu, má v sobě čtverec uložený i samotný index, kterým ho lze v mapě vyhledat.

Operace v cache jsou pak prováděné na obou strukturách. Je-li načten nový čtverec a není v cache, je přidán nejprve do hash mapy a následně je připojen na konec do listu jako nejnovější. V případě, že se v cache už nachází, je v mapě vyhledán, odstraněn z listu a následně je opět vložen na konec. V listu je tak stále dodržováno pořadí podle času přistoupení. Obrácený směr je využitý v případě odstraňování starých dat. Každá cache obsahuje limit dat, které může obsahovat. Přesáhne-li ho, jsou data odebírána, dokud se nedostane pod tuto limitní hodnotu. Odebírání probíhá ze předu listu, kde se nachází nejstarší čtverce, a následně s využitím indexu jsou tato data mazána i v hash mapě.

### 5.4.2 Správa a uvolňování paměti

V obou strukturách jsou tak udržovaná stejná data. Navíc všechny zmíněné operace probíhají v konstantním čase. Veškeré ukazatele na data jsou řešené pomocí `shared_ptr` z knihovny STL. A to jak ukazatele na třídu čtverce, tak ukazatel na surová data uvnitř čtverce.

Jedná se o speciální ukazatel, který si udržuje počet odkazů na daný objekt. Jakmile zanikne poslední odkaz, je automaticky zavolán destruktorek objektu. Data tak zůstávají validní i v případě, že jsou odstraněny z cache ale zůstávají v mapě sdílené prohlížečem a načítačem dat. Díky tomu tento přístup umožňuje i to, aby se stejná data nacházela ve dvou různých cache najednou. Může se tím ušetřit paměť například ve zmiňovaném vícefázovém načítání dat, kde v některých fázích

se mohou data pouze předávat dál. Data se nemění například ve fázi škálování, jsou-li data v prohlížeči oddálena. Pouze je vytvořena nová třída čtverce obalující samotná data.

Zvolit správné velikosti cache není nijak jednoduché, poněvadž úspěšnost cachování záleží také částečně na chování uživatele. Pokud uživatel například stále přibližuje a oddaluje data, bude vytvořeno mnohem více naškálovaných čtverců, než původních čtverců načtených z disku. Škálování je ale levnější operace, než načítání z disku, tudíž by zde případný cache miss byl bolestivější. Vzhledem k tomu, že nejvíce budou využívány první a třetí fáze předzpracování a to klasifikátorem a prohlížečem, navíc u oddalování dat ke škálování ani nedochází, byly nakonec zvoleny velikosti cache jednotlivých fází v poměru 3:1:3. V praxi se tento poměr ukazuje jako zcela použitelný, pro dosažení optimálních výsledků by pravděpodobně bylo nutné provést komplexní experiment.

## 5.5 Klasifikační algoritmy

Současně s aplikací bylo vytvořeno několik základních klasifikačních algoritmů. Veškeré zde popisované algoritmy využívají metodu strojového učení, při které jsou k dispozici trénovací data. Aplikace ale samozřejmě umožňuje testovat algoritmy i bez nich. Jak bylo zmíněno na začátku kapitoly, pro účely strojového učení využívá tato aplikace knihovnu OpenCV. Výjimkou je pouze detekce mraků, která byla též implementována jako samostatný algoritmus.

### 5.5.1 Detekce mraků

V datech ze satelitu Landsat 8 je detekce výrazně usnadněna přítomností dvanáctého snímku, který výsledky detekce mraků přímo obsahuje. K jejich detekci ho využívá i tento algoritmus. Úspěšnost této klasifikace ale bohužel není sto-procentní. K falešnému vyhodnocení mraku dochází u některých světlejších částí scén, jako jsou zejména města a zoraná pole.

U satelitu Landsat 7 není možné podobný snímek využít, je proto k detekci mraků vytvořený jednoduchý test. Ten vyhodnotí bod scény jako mrak, je-li součet hodnot z několika vybraných senzorů vyšší, než stanovený limit. Tento test také není zcela ideální, poněvadž se jednotlivé scény mohou lišit například v míře osvětlení, což způsobí na různých scénách různou citlivost detektoru. Občas se tak objevují falešné detekce v místech, jako u dat ze satelitu Landsat 8. Jedná se ale o přijatelný kompromis mezi kvalitou a časovými nároky detekce.

Oba testy trpí především na falešné detekce mraků, než na jejich neodhalení. Takto byl ale tento algoritmus navržen zcela záměrně. Důvod je ten, že falešná detekce je pro účely klasifikace mnohem přijatelnější, než používání dat s mraky pro účely trénování klasifikačních algoritmů.

### 5.5.2 Algoritmy strojového učení

Zbylé klasifikační algoritmy se pak dělí na dvě skupiny. První skupinou jsou algoritmy pracující se samostatnými scénami a druhou skupinou jsou algoritmy využívající ke klasifikaci více scén najednou. Jako vzorové byly v této aplikaci zvoleny čtyři algoritmy strojového učení. Každý je využitý k implementaci

dvou klasifikačních algoritmů, jeden z nich patří do první skupiny a druhý patří do skupiny druhé. Jedná se o algoritmy SVM (Support vector machines), Naivní Bayesův klasifikátor, učení rozhodovacím stromem a k-NN (Nearest neighbour classifier). Informace o jejich programátorském rozhraní a správném použití lze nalézt v dokumentaci knihovny OpenCV [27]

Základní informace o fungování algoritmů pracujících na jedné scéně jsou popsány níže. Pro snadnější popis zde zavádím několik pojmů a značek. Počet snímků v jedné scéně je označen proměnnou  $n$ . Většinou je každý snímek pořízen jedním senzorem satelitu. Na každý bod scény s pevnými geografickými souřadnicemi pak lze nahlížet jako na vektor  $x = (v_1, v_2, \dots, v_n)$ , kde každému číslu  $v_k$  odpovídá intenzita  $k$ -tého snímku v daném bodě. Množinu klasifikačních tříd dále označíme jako  $C$ . Trénovací data pak můžeme vyjádřit uspořádanou dvojicí  $D = (x_i, y_i)$ , kde  $x_i$  je výše zmíněný vektor a  $y_i \in C$  je třída přiřazená tomuto vektoru. Algoritmy se snaží na základě množiny  $D$  přiřadit novému vektoru odpovídající klasifikační třídu.

## SVM

Algoritmus SVM využívá ke strojovému učení  $n$ -dimenzionální příznakový prostor a každému vektoru přiřadí odpovídající bod v tomto prostoru. Pokud bychom uvažovali pouze dvě trénovací třídy, snaží se algoritmus sestrojít  $(n-1)$ -dimenzionální nadrovinu, která je bude co nejlépe oddělovat. Jsou-li navíc třídy separovatelné, je za optimální považována ta nadrovina, která množiny separuje a zároveň maximalizuje vzdálenost nadroviny od nejbližšího bodu trénovacích dat. Tato vzdálenost je v angličtině označována jako *margin*.

V případě, že množiny nejsou separovatelné, k separaci se ještě využívá jádrová transformace. Tato transformace zobrazí body z jednoho příznakového prostoru do jiného, ve kterém už jsou tyto množiny separovatelné. Většinou je nový příznakový prostor vyšší dimenze, než ten původní. Přesnější podobu této transformace je většinou možné nastavit v parametrech algoritmu. Na volbě těchto parametrů je často závislá úspěšnost celého algoritmu. Pro různé problémy jsou optimální jiné hodnoty parametrů a často tak nezbyvá, než tyto hodnoty vyzkoušet experimentálně.

Algoritmus je primárně vytvořen pouze pro dvě klasifikační třídy. Máme-li více tříd, je nejprve úloha převedena na více problémů, ve kterých můžeme separovat pouze dvě množiny. Může to být například opakované seskupování tříd do dvou skupin. Možných postupů je ale samozřejmě mnohem více.

## Naivní Bayesův klasifikátor

Bayesův klasifikátor využívá ke klasifikaci pravděpodobnostní model založený na Bayesově větě. Ta v obecné podobě vypadá takto

$$p(c|v_1, \dots, v_n) = \frac{P(c) \cdot P(v_1, \dots, v_n|c)}{P(v_1, \dots, v_n)} \quad (5.1)$$

Zde v interpretaci strojového učení je  $c \in C$  klasifikační třída a  $x = (v_1, \dots, v_n)$  vektor příznaků odpovídající jednomu bodu ve scéně. Levá strana rovnice nám jinými slovy říká, s jakou pravděpodobností patří vektor  $x$  do třídy  $c$ . To je také

to, co se snažíme spočítat. Třída, které vyjde největší pravděpodobnost, bude přiřazena danému bodu.

Pravá strana rovnice pak obsahuje  $P(c)$  jako pravděpodobnost samotné třídy  $c$ . V našem případě předpokládáme, že výskyt každé třídy je stejně pravděpodobný, můžeme proto tuto pravděpodobnost zcela zanedbat. Lze ji ale také využít třeba jako parametr k nastavení klasifikátoru. Ze stejných důvodů můžeme zanedbat i pravděpodobnost výskytu vektoru  $x$ . Úlohu pak lze zredukovat na hledání maxima z množiny  $\{P(x|c)|c \in C\}$ .

Existuje více matematických modelů, jak určit  $P(x|c)$ . V případě OpenCV se předpokládá, že výskyt bodů jednotlivých tříd se řídí normálním rozdělením. Každé třídě odpovídá jedna Gaussova funkce v  $n$ -dimenzionálním prostoru. Bod je pak zařazen to třídy, jehož Gaussova funkce má v tomto bodě největší hodnotu. Proces trénování algoritmu tedy obnáší pouze spočítání parametrů těchto funkcí. V případě  $n$ -dimenzionálního prostoru je pro každý algoritmus nutné spočítat jednu pozitivně-definitní matici o rozměrech  $n \times n$ .

## Rozhodovací strom

Další algoritmus využívá ke klasifikaci rozhodovací strom. Rozhodovací strom je obecně strom, kde listy odpovídají klasifikačním třídám a vnitřní uzly jsou testy. Algoritmus začíná v kořeni stromu a na základě těchto testů se rozhoduje, jakou větví bude dále pokračovat. To provádí do té doby, dokud se nedostane do listu, jehož třídu přiřadí klasifikovanému bodu.

Při trénování je tento strom stavěn rekurzivně od kořene. V případě knihovny OpenCV je stavěn pouze binární strom. V každém vrcholu je spočítáno rozhodovací pravidlo, pomocí něhož je množina trénovacích dat v daném vrcholu rozdělena na dvě skupiny. Zároveň s tím dojde k rozdělení vrcholu a rozhodovací pravidlo pak tvoří test, který určuje, jakou větví se má algoritmus vydat.

Rozhodovací pravidlo je vytvářeno tak, aby „co nejlépe“ rozdělovalo danou množinu. K tomu se dají využít různé metriky, v tomto případě se zvolená metrika nazývá Gini impurity. K zastavení tohoto rekurzivního procesu pak má algoritmus vnitřní schémata, která mají zabránit jejímu přetrénování.

## Klasifikátor k-NN

k-NN patří mezi velice jednoduché klasifikační algoritmy. Stejně jako SVM pracuje v  $n$ -dimenzionálním příznakovém prostoru. Pro každý klasifikovaný bod vybírá postupně nejbližší body z trénovací množiny, dokud nenapočítá  $k$  bodů z jedné trénovací třídy. Jakmile algoritmus získá těchto  $k$  bodů, jejich třídu přiřadí i klasifikovanému bodu.

Je zřejmé, že volba parametru  $k$  bude klíčová pro výsledek klasifikace. Tento algoritmus je velice citlivý na chyby v trénovacích datech. Čím větší množství chyb obsahuje, tím větší  $k$  je nutné volit, aby byly tyto chyby potlačeny. Je-li použito velké množství trénovacích dat, může být hledání nejbližších bodů ve vícerozměrných prostorech časově náročné. Obecně se tak tento algoritmus považuje za vhodný zejména pro malé množství trénovacích dat.

### 5.5.3 Algoritmy využívající více scén

Výše zmíněné algoritmy strojového učení jsou pak použity i pro vytvoření klasifikačních algoritmů využívající více scén ze stejné oblasti. Algoritmus je v principu velice jednoduchý. Pro každou scénu je zvlášť spuštěn jeden klasifikační algoritmus. Z výsledků klasifikace na jednotlivých scénách je pak pro zvolený bod vybrána ta třída, která pro tento bod byla nejčastějším výsledkem klasifikace. Zvýšení úspěšnosti tohoto přístupu v porovnání s klasifikačními algoritmy pracujícími na samostatných scénách bylo porovnáno v experimentu v následující kapitole.

Možnosti těchto algoritmů jsou ale zajisté mnohem širší. Některé klasifikační algoritmy lze rozšířit pro klasifikaci z více scén. Přímo na Landsat datech se této problematice věnovali například Solberg, Jain a Taxt [28]. Dále není v tomto algoritmu například žádným způsobem využitý čas, kdy byly snímky pořízeny. Ke klasifikaci by se také mohlo využít i některé charakteristiky, jako je rozptýl hodnot pixelů nebo tvar křivky, kterou tyto hodnoty vytvářejí v čase. Výsledkem by mohly být algoritmy vytvořené „na míru“ jednotlivým klasifikačním třídám.

### 5.5.4 Programátorské rozhraní pro tvorbu klasifikačních algoritmů

Pro tvorbu a porovnávání klasifikačních algoritmů je v rámci aplikace navrženo jednotné programátorské rozhraní. Každý algoritmus dědí z abstraktní třídy, ze které implementuje vybrané metody. K jeho zařazení do aplikace stačí instanci přidat do listu s ostatními třídami. Pro samotnou klasifikaci jsou významné především dvě metody. *Train* pro natrénování dat a *predict* pro určení třídy, do které má daný pixel patřit.

Obecně jsou zde klasifikační algoritmy ve dvou variantách. První je klasifikace samotných scénách a druhou je klasifikace využívající časový vývoj v dané oblasti. Pro každou skupinu je vytvořeno samostatné rozhraní. Množinu scén, které budou do klasifikace zahrnuty, si uživatel může zvolit v dialogu s nastavením. Detailní fungování jednotlivých algoritmů je popsáno v následující kapitole.

Kromě dvojice *train* a *predict* je zde ještě několik funkcí, které zajišťují další související věci s klasifikací. Jedna, která vrací název algoritmu, jakým se bude identifikovat v aplikaci. Další funkce slouží k informování algoritmu, že došlo ke změně klasifikačních tříd a je tedy nutné nové natrénování dat. Dále jsou k dispozici ještě dvě dodatečné funkce pro detekci mraků a oblastí mimo rozsah senzorů. Trénovací data, která budou kladně vyhodnocena jednou z těchto dvou funkcí, nebudou do trénování klasifikátoru zahrnuta. Obě funkce si samozřejmě může uživatel implementovat sám nebo je zcela potlačit.

Vybraný algoritmus se nemusí vždy podařit úspěšně použít. Jak bylo zmíněno výše, pixely vyhodnocené jako mraky nejsou do trénování zahrnuty. Je-li tedy scéna výrazně pokrytá mraky, nemusí být dostupné dostatečné množství trénovacích dat. To může vést například k nepřítomnosti některých tříd nebo nemusí dojít k nenatrénování vůbec. Některé algoritmy navíc mohou k úspěšnému natrénování vyžadovat jisté minimální množství trénovacích dat.

# 6. Srovnání úspěšnosti vybraných algoritmů

Algoritmy implementované v této aplikaci byly otestovány v experimentu. Cílů tohoto testu bylo hned několik. Zaprvé posloužil pro vyzkoušení a ověření funkčnosti aplikace. Experimentem byla ověřena použitelnost aplikace pro účely vytvoření trénovacích dat, analýzu a vyhodnocení výsledků klasifikačních algoritmů.

Zadruhé bylo cílem porovnat zde implementované klasifikační algoritmy. Zhodnotit jejich úspěšnost na datech z projektu Landsat a porovnat s obecně známými vlastnosti těchto algoritmů. Dále bylo cílem vyzkoušet, zda a případně jakého zlepšení lze dosáhnout, použijeme-li algoritmus, který bude využívat více scén současně.

K experimentu byly použity tři scény ze satelitu Landsat 8. Jsou označeny jako scéna 1, 2 a 3. Jejich identifikátory jsou ve stejném pořadí LC81910262013208LGN00, LC81910262013272LGN00 a LC81910262013304LGN00. Celý experiment pak byl rozdělen na tři nezávislé menší experimenty. Jednotlivé experimenty se liší především v množství trénovacích dat.

Tyto scény byly speciálně vybrány pro jejich malé pokrytí mraky. Klasifikace pak probíhala na těch částech snímků, kde se v ani jedné ze scén nevyskytovaly žádné mraky. Při experimentu bylo vypnuté automatické filtrování mraků. Pro zopakování experimentu je tento zásah nutný z důvodu několika falešných detekcí v oblastech zastavěné plochy.

## 6.1 Vytvoření trénovacích a referenčních dat

Celkem byly vytvořeny tři skupiny trénovacích dat. Jedna skupina obsahovala přibližně 10 bodů od každé trénovací třídy, druhá skupina měla okolo 100 bodů na třídu a třetí okolo 1000 bodů na jednu třídu. Pro jednoduchost jsou dále označeny jako malá, střední a velká trénovací data. Pro účely vyhodnocení algoritmů byly navíc ručně vytvořeny referenční data, podle kterých byla vyhodnocena úspěšnost algoritmů. Počet bodů v těchto datech je řádově několik tisíc od každé třídy. Body referenčních dat jsou stejné pro všechny tři experimenty. Samotné třídy byly zvoleny jen 4 základní. Jednalo se o lesy, pole a louky, vodní hladinu a zastavěné oblasti.

Vytvoření těchto dat probíhalo standardním vybíráním obdélníkových oblastí v naší aplikaci. Nejprve byla vybrána první oblast bez mraků, kde byly následně označeny oblasti pro trénovací třídy. V jiné části scény pak byla vybrána relativně malá oblast, kde byla vytvořena referenční data. Do referenčních dat byla snaha začlenit co nejvíce souvislou oblast. Do každé třídy byly přidávány všechny její výskyty v dané oblasti, které bylo možné lidským okem odlišit a vyznačit pomocí výběrů obdélníkového tvaru.

Algoritmy pak byly postupně natrénované na jednotlivých scénách, případně na kombinaci všech tří scén. Následně byl pro každý z nich pomocí tlačítka „Classify samples from file“ vygenerován výsledek klasifikace.

## 6.2 Struktura prezentovaných dat

Z vygenerovaných dat byly sestaveny tabulky informující o úspěšnosti jednotlivých algoritmů. Každá tabulka obsahuje počty správně klasifikovaných bodů jednoho algoritmu s využitím vybrané množiny trénovacích dat. Příkladem je tabulka 6.1, která ukazuje výsledky SVM algoritmu využívajícího střední trénovací data. Množinu všech referenčních bodů lze rozdělit do čtyř skupin podle toho, do jaké třídy skutečně patří. První množině odpovídá v tabulce řádek s názvem *Lesy* a jsou to ručně vybrané body, na kterých se nacházel pouze lesní porost. Další tři řádky značí množiny zbylých tří tříd, které se algoritmy pokoušely klasifikovat. Jsou to *Město* označující zastavěnou plochu, *Pole* jako pole a zatravněné plochy a *Voda* odpovídající vodní hladině.

Tabulka 6.1: Výsledky algoritmu SVM na středních trénovacích datech

	Scéna 1	Scéna 2	Scéna 3	Kombinace	Bodů celkem
Lesy	8642	8555	8046	8583	8655
Město	1779	1765	1890	1865	2025
Pole	3516	3667	3775	3845	3859
Voda	1311	1290	1192	1290	1315
Celkem	15248	15277	14903	15583	15854
Úspěšnost	0.9618	0.9636	0.9400	0.9829	

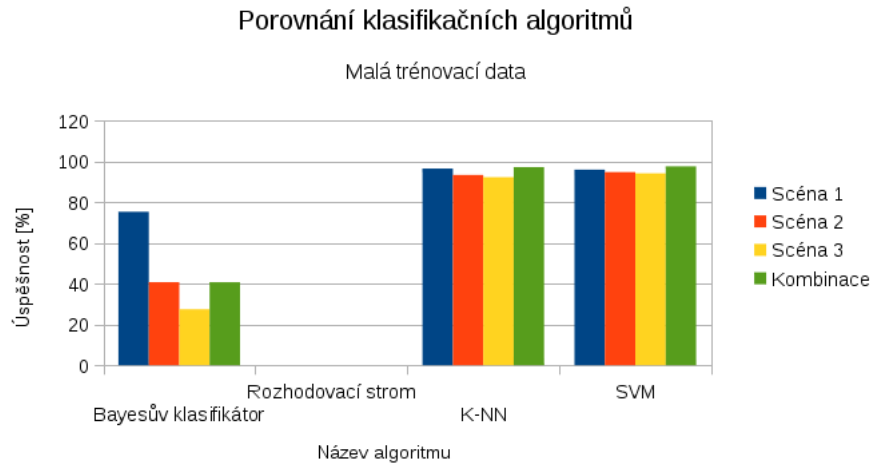
Každý z těchto čtyř řádků nese informace o počtech bodů, které algoritmus přiřadil do stejné třídy, do jaké byly vloženy uživatelem při výběru referenčních dat. Tři sloupce označené jako *Scéna 1*, *Scéna 2* a *Scéna 3* označují jak byl algoritmus úspěšný u jednotlivých scén, na kterých bylo prováděno testování. Ve sloupci *Kombinace* je pak počet bodů, které správně vyhodnotil algoritmus využívající všechny tři scény současně. Poslední sloupec informuje o celkovém množství bodů, které byly použity u jednotlivých tříd.

Předposlední řádek *Celkem* pak obsahuje počty správně vyhodnocených bodů ze všech čtyř tříd dohromady a poslední řádek s názvem *Úspěšnost* ukazuje podíl počtu správně vybraných bodů a počtu všech klasifikovaných bodů referenčních dat. Z těchto konkrétních dat je vidět, že algoritmu SVM na scéně 1 úspěšně klasifikoval 96.18 % referenčních dat, na scéně 2 měl úspěšnost 96.36 % a na scéně 3 měl 94 %. Při využití všech tří scén současně byl úspěšný z 98.29 %.

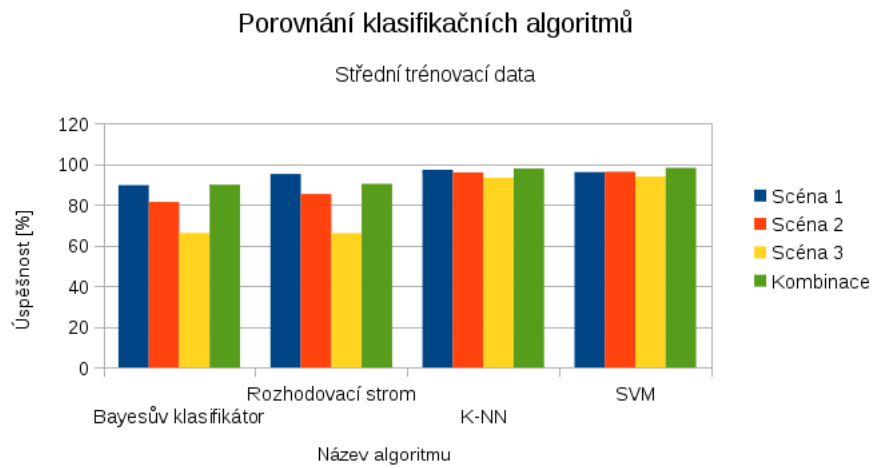
Úplné výsledky všech algoritmů a na všech třech velikostech trénovacích dat jsou shrnuté v příložených tabulkách 6.2 6.3 6.4. Celkové úspěšnosti všech tříd dohromady jsou pak znázorněné ve třech grafech 6.1 6.2 6.3. Každý sloupec grafu ukazuje procentuální úspěšnost algoritmu na vybrané velikosti trénovacích dat a na dané scéně nebo kombinaci scén.

## 6.3 Vyhodnocení experimentu

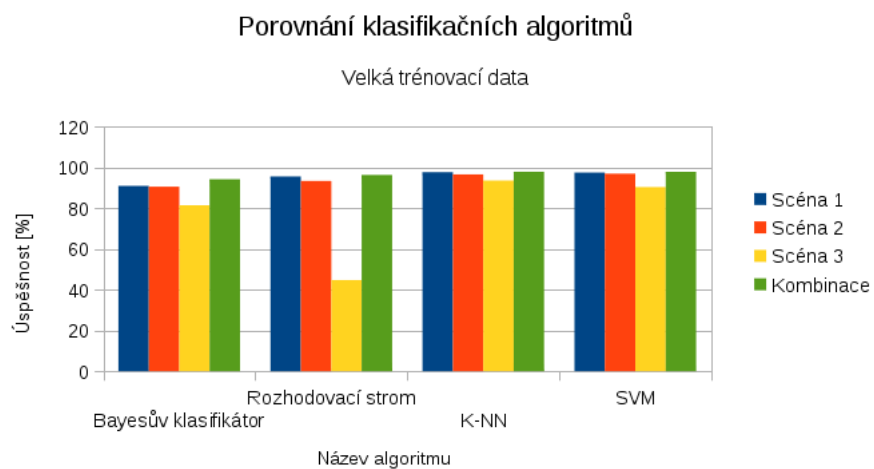
Na grafu, kde byla použita malá trénovací data, si lze všimnout, že zcela chybí výsledky klasifikace rozhodovacího stromu. To je způsobeno tím, algoritmus vyžaduje určité minimální množství trénovacích dat, aby mohl být vůbec natrénován. To nebylo s množstvím 10 bodů na třídu splněno. Nejúspěšnější na těchto datech



Obrázek 6.1: Výsledky experimentu na malých trénovacích datech



Obrázek 6.2: Výsledky experimentu na středních trénovacích datech



Obrázek 6.3: Výsledky experimentu na velkých trénovacích datech

byly očividně k-NN a SVM algoritmy, které si drží úspěšnost okolo 95 %. Výkon Bayesova algoritmu byl naopak dosti nevyrovnaný. Některé třídy byly dokonce při klasifikaci téměř zcela vynechány.

Na střední velikosti trénovacích dat již byl Bayesův algoritmus i rozhodovací strom o něco úspěšnější. Úspěšnost byla nízká zejména u scény 3, kde se u obou pohybovala okolo 66 %. k-NN a SVM algoritmy si naopak stále drží velice vysokou úspěšnost nad 50 %.

V posledním experimentu již Bayesův algoritmus i rozhodovací strom dosahovaly 90 % úspěšnosti. Výjimkou ale byla opět scéna 3, ve které dokonce rozhodovací strom dosáhl necelých 45 %. Vzhledem k výsledkům na ostatních scénách je tento výsledek poměrně překvapující. Na něm lze ale poměrně dobře demonstrovat fakt, že algoritmus úspěšný na jedné scéně, může být s trénovacími i referenčními daty ze stejné oblasti zcela neúspěšný na scéně jiné. V kontextu všech tří experimentů je dále možné sledovat trend, že úspěšnost obou algoritmů se zvyšuje s přibývajícím množstvím trénovacích dat.

Úspěšnost k-NN a SVM algoritmů je naopak stále velice vysoká a stabilní. S přibývajícím množstvím trénovacích dat se ale začínají u algoritmu k-NN výrazně projevovat jeho časové nároky. V případě velkých trénovacích dat se tento algoritmus ukazuje prakticky téměř nepoužitelný.

V této souvislosti stojí za povšimnutí, že algoritmy byly vždy méně úspěšné na scéně 3, než na ostatních scénách. Stejně tak na scéně 2 byly až na jednu výjimku méně úspěšné než na scéně 1. To může souviset s faktem, že osvětlení scény 3 je mnohem menší, než scény 2 a to je menší než scény 1. Při sníženém osvětlení tak může dojít ke ztrátě některých detailů spojených s klasifikací. Dalším vlivem může být samozřejmě také to, že jsou scény pořízeny v rozdílném období v roce.

Z naměřených dat lze také téměř s jistotou říci, že i tak jednoduchý způsob klasifikace více scén, který byl zvolen v této aplikaci, výrazně zvyšuje na těchto třech scénách úspěšnost klasifikace. Z 11 zde změřených výsledků na různých algoritmech a trénovacích datech byl v 9 případech algoritmus využívající všechny tři scény úspěšnější než nejlepší výsledek na samostatné scéně. Navíc dojde-li k selhání algoritmu pouze na jedné scéně ze tří, je tento způsob schopný velice dobře potlačit toto selhání, jako to bylo vidět například u algoritmu využívající rozhodovací strom ve třetím experimentu.

# Závěr

V této práci byla navržena a implementována aplikace pro vizualizaci, analýzu a klasifikaci dat. Aplikace byla vytvořena s důrazem na interaktivitu a automatické zpracování a zobrazování dat. Vícevláknový návrh umožňuje souběžně načítat data z disku, provádět klasifikaci i měnit parametry vizualizovaných snímků.

Pro ukládání dat byl vytvořen nový formát, který je založený na rozřezání snímků na čtverce pevné velikosti. Přínosem tohoto přístupu je snadnější načítání, zpracování a cachování dat. Dostatečně rychlá odezva a efektivní využívání paměti je dále umožněna díky principu „on demand“. Na základě tohoto principu jsou veškerá data načítána až ve chvíli, kdy jsou skutečně potřeba.

Data jsou ukládána do adresářové struktury a každý čtverec je v ní samostatným souborem. Další výhodou je tak možnost odstranit čtverce nacházející se v oblastech, které nejsou pro senzory dostupné, a ušítřit tak paměť. Rozsah těchto oblastí může u dat z projektu Landsat dosahovat až 50 % z celkové plochy snímku.

Na druhou stranu je nevýhodou velké množství souborů, které v této souvislosti vzniknou. Načtení každého souboru je spojené se systémovým voláním, což zbytečně zpomaluje aplikaci. Vzhledem k volbě velikosti čtverce, není toto zpomalení nijak významné, je zde ale prostor pro případné vylepšení. Efektivnějším řešením by mohlo být ukládání všech čtverců do jednoho souboru či využití externího databázového serveru.

Dále bylo implementováno několik klasifikačních algoritmů a k nim i jejich rozšířené verze, které využívají více scén současně. Úspěšnost těchto algoritmů byla srovnána v experimentu na snímcích ze satelitu Landsat 8. Experiment posloužil jako demonstrace použitelnosti této aplikace pro účely vytváření, analyzování a porovnávání klasifikačních algoritmů.

Kromě toho ukázal i na vlastnosti a chování několika klasifikačních algoritmů na datech z projektu Landsat. Bayesův algoritmus a algoritmus využívající rozhodovací strom měly poměrně malou úspěšnost, pokud využívaly malé množství trénovacích dat. Se zvětšujícím množstvím trénovacích dat rostla i jejich úspěšnost. Naopak algoritmy k-NN a SVM byly úspěšné na všech trénovacích datech, na kterých byli testovány.

Tyto výsledky je nutné vztahovat k podmínkám vytvořeným touto aplikací, typem použitých dat a lidským faktorem, který vytvářel trénovací a referenční data. Za těchto podmínek se ale pro malé množství trénovacích dat ukázaly algoritmy k-NN a SVM jednoznačně nejvhodnější. Současně s tím ukázal experiment i to, že jednoduchý algoritmus kombinující více scén, který byl v experimentu použit, může výrazně zvyšovat úspěšnost klasifikace.

Možnosti dalších experimentů v této oblasti nebyly ani zdaleka vyčerpány. Mnoho algoritmů lze rozšířit, aby umožňovaly využívat pro klasifikaci více scén současně. Implementace a porovnávání těchto algoritmů by tak mohlo být pokračováním této práce.

Pro účely trénování klasifikačních algoritmů obsahuje aplikace ještě nástroj pro výběr trénovacích dat. Jistým omezením byla nemožnost využívat k výběru trénovacích dat oblasti jiných než obdélníkových tvarů. Každá oblast pixelů lze sice pokrýt pomocí obdélníků, dodržování hranic, které nejsou vodorovné či svislé,

může být ale časově náročné. Rozšíření této aplikace o další možnosti výběru oblastí by bylo do budoucna velice vhodné.

Další možnosti rozšíření aplikace se nachází v množství satelitů, jejichž data jsou aplikací podporována. Pro testování aplikace byla vytvořena podpora pouze pro data ze satelitů Landsat 7 a 8.

Smyslem aplikace bylo především prakticky ověřit, že zde navržený a implementovaný koncept je pro účely analýzy, vizualizace a klasifikace satelitních dat použitelný. To se také z velké části podařilo prokázat. Pro dovedení aplikace do plnohodnotného GIS systému by ale bylo potřeba ještě velkých úprav a rozšíření.

# Seznam použité literatury

- [1] U. S. GEOLOGICAL SURVEY. *Landsat Missions* [online]. 7.5.2014 [cit. 2014-7-15]. Dostupné z: <http://landsat.usgs.gov/>
- [2] Landsat Satellites and Sensors. NATIONAL AERONAUTICS AND SPACE ADMINISTRATION. *Landsat 7 Science Data User's Handbook* [online]. 11.3.2011 [cit. 2014-07-17]. Dostupné z: <http://landsathandbook.gsfc.nasa.gov/program/>
- [3] SLC Failure. U. S. GEOLOGICAL SURVEY. *Landsat Missions* [online]. 16.1.2014 [cit. 2014-07-17]. Dostupné z: [http://landsat.usgs.gov/products\\_slciffbackground.php](http://landsat.usgs.gov/products_slciffbackground.php)
- [4] Radiometric Characteristics. NATIONAL AERONAUTICS AND SPACE ADMINISTRATION. *Landsat 7 Science Data User's Handbook* [online]. 11.3.2011 [cit. 2014-07-17]. Dostupné z: [http://landsathandbook.gsfc.nasa.gov/data\\_properties/prog\\_sect6\\_4.html](http://landsathandbook.gsfc.nasa.gov/data_properties/prog_sect6_4.html)
- [5] Landsat 8 Quality Assessment Band. U. S. GEOLOGICAL SURVEY. *Landsat Missions* [online]. 2.4.2014 [cit. 2014-07-17]. Dostupné z: <http://landsat.usgs.gov/L8QualityAssessmentBand.php>
- [6] Landsat Processing Details. U. S. GEOLOGICAL SURVEY. *Landsat Missions* [online]. 5.3.2014 [cit. 2014-07-17]. Dostupné z: [http://landsat.usgs.gov/Landsat\\_Processing\\_Details.php](http://landsat.usgs.gov/Landsat_Processing_Details.php)
- [7] OPEN GEOSPATIAL CONSORTIUM. *Geospatial and location standards* [online]. © 1994-2014 [cit. 2014-7-17]. Dostupné z: <http://www.opengeospatial.org/>
- [8] Featured Tools. U. S. GEOLOGICAL SURVEY. *Landsat Missions* [online]. 16.7.2013 [cit. 2014-07-17]. Dostupné z: [http://landsat.usgs.gov/tools\\_featured.php](http://landsat.usgs.gov/tools_featured.php)
- [9] GeoServer GeoNetwork with web app. SEWILCO. *Wikimedia Commons* [online]. 7.12.2007 [cit. 2014-07-17]. Dostupné z: [http://commons.wikimedia.org/wiki/File:GeoServer\\_GeoNetwork\\_with\\_web\\_app.png](http://commons.wikimedia.org/wiki/File:GeoServer_GeoNetwork_with_web_app.png)
- [10] U. S. GEOLOGICAL SURVEY. *Global Visualization Viewer* [online]. 24.6.2014 [cit. 2014-7-17]. Dostupné z: <http://glovis.usgs.gov/>
- [11] U. S. GEOLOGICAL SURVEY. *LandsatLook Viewer* [online]. 31.5.2013 [cit. 2014-7-17]. Dostupné z: <http://landsatlook.usgs.gov/>
- [12] U. S. GEOLOGICAL SURVEY. *Earth Explorer* [online]. 1.7.2014 [cit. 2014-7-17]. Dostupné z: <http://earthexplorer.usgs.gov/>
- [13] GRASS DEVELOPMENT TEAM. *GRASS GIS* [online]. © 1998-2014 [cit. 2014-7-17]. Dostupné z: [grass.osgeo.org/](http://grass.osgeo.org/)

- [14] Early GRASS 6.3-CVS running natively on MS-Windows (Tcl/Tk GUI). GRASS DEVELOPMENT TEAM. *GRASS GIS* [online]. © 1998-2014 [cit. 2014-7-17]. Dostupné z: <http://grass.osgeo.org/screenshots/platforms/>
- [15] GRASS GIS 6.4.5svn Reference Manual. GRASS DEVELOPMENT TEAM. *GRASS GIS* [online]. © 2003-2014 [cit. 2014-7-17]. Dostupné z: <http://grass.osgeo.org/grass64/manuals/index.html>
- [16] QGIS COMMUNITY. *QGIS project* [online]. 19.7.2014 [cit. 2014-7-19]. Dostupné z: <http://www.qgis.org/>
- [17] ASOCIACIÓN GVSIG. *gvSIG Portal* [online]. 2011 [cit. 2014-7-19]. Dostupné z: <http://www.gvsig.org/web>
- [18] SAGA USER GROUP ASSOCIATION. *System for Automated Geoscientific Analyses* [online]. 19.7.2014 [cit. 2014-7-19]. Dostupné z: <http://www.saga-gis.org/>
- [19] MILLER, H.M., SEXTON, N.R., KOONTZ, Lynne, LOOMIS, John, KOONTZ, S.R., and HERMANS, Caroline. The users, uses, and value of Landsat and other moderate-resolution satellite imagery in the United States. *Executive report: U.S. Geological Survey Open-File Report 2011-1031*. 2011 [cit. 2014-7-19]. Dostupné z: <http://pubs.usgs.gov/of/2011/1031/pdf/OF11-1031.pdf>
- [20] SABINS, Floyd F. Remote sensing for mineral exploration. *Ore Geology Reviews*. 1999, vol. 14, 3-4, s. 157-183. DOI: 10.1016/S0169-1368(99)00007-4. Dostupné z: <http://linkinghub.elsevier.com/retrieve/pii/S0169136899000074>
- [21] The GNU C++ Library. FREE SOFTWARE FOUNDATION, INC. *GCC, the GNU Compiler Collection* [online]. 16.7.2014 [cit. 2014-07-19]. Dostupné z: <https://gcc.gnu.org/onlinedocs/libstdc++/>
- [22] QT DEVELOPMENT FRAMEWORKS. *Qt Project* [online]. © 2014 [cit. 2014-7-19]. Dostupné z: <http://qt-project.org/>
- [23] SILICON GRAPHICS. *LibTIFF - TIFF Library and Utilities* [online]. 23.12.2003 [cit. 2014-7-19]. Dostupné z: <http://www.libtiff.org/>
- [24] EICHHAMMER, Emanuel. EMANUEL EICHHAMMER. *QCustomPlot* [online]. © 2013-2014 [cit. 2014-7-19]. Dostupné z: <http://www.qcustomplot.com/>
- [25] ITSEEZ. *Open Source Computer Vision Library* [online]. © 2014 [cit. 2014-7-19]. Dostupné z: <http://opencv.org/>
- [26] Hash Functions. YIGIT, Ozan. *Department of Computer Science and Engineering*, York University. [cit. 2014-7-19]. Dostupné z: <http://www.cse.yorku.ca/~oz/hash.html>

- [27] ml. Machine Learning. ITSEEZ. *Open Source Computer Vision Library* [online]. 21.4.2014 [cit. 2014-7-19]. Dostupné z: <http://docs.opencv.org/modules/ml/doc/ml.html>
- [28] SOLBERG, A.H.S., A.K. JAIN a T. TAXT. Multisource classification of remotely sensed data: fusion of Landsat TM and SAR images. *IEEE Transactions on Geoscience and Remote Sensing*. vol. 32, issue 4, s. 768-778. DOI: 10.1109/36.298006. Dostupné z: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=298006>

# Seznam tabulek

Tabulka 6.2: Výsledky experimentu na malých trénovacích datech

(a) Bayesův klasifikátor

	Scéna 1	Scéna 2	Scéna 3	Kombinace	Bodů celkem
Lesy	7811	2188	37	2044	8655
Mesto	1109	619	464	590	2025
Pole	3028	3653	3859	3823	3859
Voda	4	1	0	0	1315
Celkem	11952	6461	4360	6457	15854
Úspěšnost	0.7539	0.4075	0.2750	0.4073	

(b) k-NN

	Scéna 1	Scéna 2	Scéna 3	Kombinace	Bodů celkem
Lesy	8461	8407	8352	8468	8655
Mesto	1986	1388	1438	1840	2025
Pole	3557	3741	3624	3832	3859
Voda	1308	1285	1230	1291	1315
Celkem	15312	14821	14644	15431	15854
Úspěšnost	0.9658	0.9348	0.9237	0.9733	

(c) SVM

	Scéna 1	Scéna 2	Scéna 3	Kombinace	Bodů celkem
Lesy	8542	8392	8124	8444	8655
Mesto	1527	1934	1787	1907	2025
Pole	3856	3430	3843	3859	3859
Voda	1307	1281	1187	1281	1315
Celkem	15232	15037	14941	15491	15854
Úspěšnost	0.9608	0.9485	0.9424	0.9771	

Tabulka 6.3: Výsledky experimentu na středních trénovacích datech

(a) Bayesův klasifikátor

	Scéna 1	Scéna 2	Scéna 3	Kombinace	Bodů celkem
Lesy	8285	6274	4998	8166	8655
Mesto	1951	1882	1631	1959	2025
Pole	3818	3840	3564	3847	3859
Voda	167	920	299	291	1315
Celkem	14221	12916	10492	14263	15854
Úspěšnost	0.8970	0.8147	0.6618	0.8996	

(b) Rozhodovací strom

	Scéna 1	Scéna 2	Scéna 3	Kombinace	Bodů celkem
Lesy	8558	8635	5740	8643	8655
Mesto	1961	1208	1491	1520	2025
Pole	3281	2397	2034	2874	3859
Voda	1298	1296	1211	1295	1315
Celkem	15098	13536	10476	14332	15854
Úspěšnost	0.9523	0.8538	0.6608	0.9040	

(c) K-NN

	Scéna 1	Scéna 2	Scéna 3	Kombinace	Bodů celkem
Lesy	8583	8539	8371	8564	8655
Mesto	1734	1744	1895	1867	2025
Pole	3789	3650	3309	3798	3859
Voda	1312	1290	1223	1292	1315
Celkem	15418	15223	14798	15521	15854
Úspěšnost	0.9725	0.9602	0.9334	0.9790	

(d) SVM

	Scéna 1	Scéna 2	Scéna 3	Kombinace	Bodů celkem
Lesy	8642	8555	8046	8583	8655
Mesto	1779	1765	1890	1865	2025
Pole	3516	3667	3775	3845	3859
Voda	1311	1290	1192	1290	1315
Celkem	15248	15277	14903	15583	15854
Úspěšnost	0.9618	0.9636	0.9400	0.9829	

Tabulka 6.4: Výsledky experimentu na velkých trénovacích datech

(a) Bayesův klasifikátor

	Scéna 1	Scéna 2	Scéna 3	Kombinace	Bodů celkem
Lesy	7754	7612	7431	8214	8655
Mesto	1858	1883	1371	1875	2025
Pole	3841	3821	3849	3849	3859
Voda	981	1053	272	1012	1315
Celkem	14434	14369	12923	14950	15854
Úspěšnost	0.9104	0.9063	0.8151	0.9430	

(b) Rozhodovací strom

	Scéna 1	Scéna 2	Scéna 3	Kombinace	Bodů celkem
Lesy	8559	8559	427	8544	8655
Mesto	1941	1247	1612	1624	2025
Pole	3366	3744	3846	3845	3859
Voda	1307	1271	1212	1275	1315
Celkem	15173	14821	7097	15288	15854
Úspěšnost	0.9570	0.9348	0.4476	0.9643	

(c) K-NN

	Scéna 1	Scéna 2	Scéna 3	Kombinace	Bodů celkem
Lesy	8450	8465	8293	8486	8655
Mesto	1956	1769	1750	1918	2025
Pole	3785	3820	3601	3855	3859
Voda	1313	1279	1204	1281	1315
Celkem	15504	15333	14848	15540	15854
Úspěšnost	0.9779	0.9671	0.9365	0.9802	

(d) SVM

	Scéna 1	Scéna 2	Scéna 3	Kombinace	Bodů celkem
Lesy	8583	8528	8098	8559	8655
Mesto	1966	1728	1534	1825	2025
Pole	3604	3845	3505	3857	3859
Voda	1313	1286	1207	1290	1315
Celkem	15466	15387	14344	15531	15854
Úspěšnost	0.9755	0.9705	0.9048	0.9796	

# Seznam použitých pojmů a zkratek

**GIS (Geografický informační systém)** = Počítačový systém pro správu, vizualizaci a analýzu dat, která mají prostorový vztah k povrchu Země.

**WRS (Worldwide Reference System)** = Celosvětový referenční systém definující oblasti na zemském povrchu, kterou využívá i projekt Landsat pro zařazení scén do geografického souřadnicového systému. Každá oblast je jednoznačně dána dvojicí čísel *path* a *row*.

**Scéna** = Soubor snímků pořízených satelitem projektu Landsat, vytvořených nad oblastí danou souřadnicemi WRS v jednom konkrétním čase.

**Snímek** = Fotografie ve stupních šedi pořízená jedním senzorem satelitu.

**Čtverec** = Oblast čtvercového tvaru vyříznutá z jednoho snímku.

**Bod** = Pozice na Zemi jednoznačně určená geografickými souřadnicemi.

**Pixel** = Jeden bod na jednom snímku scény. Hodnotou pixelu je senzorem naměřená intenzita v tomto bodě.

**Databáze** = Uskupení předzpracovaných scén, které je schopná zde popisovaná aplikace načítat a vizualizovat. Scény jsou uloženy na disk do adresářové struktury.

# Přílohy

V příloženém CD jsou následující data:

- Složka `experiment/` s trénovacími daty, které byly použity při experimentu. Společně s referenčními daty a se scénami, které jsou volně dostupné ke stažení z internetu, umožňují zopakování experimentu. Soubory mají názvy `data_mala.smp`, `data_stredni.smp`, `data_velka.smp` a `data_ref.smp`. První tři postupně odpovídají malým, středním a velkým trénovacím datům. Posledním souborem jsou data referenční.
- Ve složce `src/landsat` jsou obsaženy zdrojové kódy aplikace. Ve stejné složce se nachází i soubor `COMPILE`, který obsahuje informace o tom, jak přeložit kódy na různých platformách.
- Další je složka `build/`, která obsahuje soubor `landsat.exe`. Jedná se o 32-bitový spustitelný soubor pro platformu Microsoft Windows. Zbytek složky tvoří dynamické knihovny potřebné pro spuštění.
- Poslední složkou je `doc/`, ve které se nachází programátorská dokumentace vygenerovaná programem `Doxygen`.