



ZADÁNÍ DIPLOMOVÉ PRÁCE

Název:	System pro sběr a zobrazení sférických panoramat
Student:	Bc. Josef Rypáček
Vedoucí:	Ing. Petr Pulc
Studijní program:	Informatika
Studijní obor:	Webové a softwarové inženýrství
Katedra:	Katedra softwarového inženýrství
Platnost zadání:	Do konce letního semestru 2017/18

Pokyny pro vypracování

Realizujte systém pro automatický sběr, indexaci a následné prohlížení sférických panoramat:

1. Seznamte se se sférickými panoramaty a metodami jejich zobrazení.
2. Prozkoumejte alespoň dvě webové služby pro sdílení fotografií a jejich API.
3. Navrhněte, implementujte a otestujte komunikaci s těmito vybranými službami. Především za účelem získání nových fotografií.
4. Navrhněte, implementujte a otestujte detekci sférické fotografie. Zvažte použití klasifikátoru i vlastních pravidel.
5. Navrhněte index pro sférická panoramata. I s ohledem na možnost následného použití indexu pro shlukovou analýzu.
6. Implementujte webovou aplikaci pro prohlížení sférických panoramat a informací o nich.

Seznam odborné literatury

Dodá vedoucí práce.

L.S.

Ing. Michal Valenta, Ph.D.
vedoucí katedry

prof. Ing. Pavel Tvrdík, CSc.
děkan

V Praze dne 21. prosince 2016

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
KATEDRA SOFTWAREVÉHO INŽENÝRSTVÍ



Diplomová práce

Systém pro sběr a zobrazení sférických panoramat

Bc. Josef Rypáček

Vedoucí práce: Ing. Petr Pulc

7. května 2017

Poděkování

Děkuji svému vedoucímu Ing. Petru Pulcovi za poskytnutí zajímavého tématu, za rady ohledně způsobu zpracování a náplně práce.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval(a) samostatně a že jsem uvedl(a) veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 46 odst. 6 tohoto zákona tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou, a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla, a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené. Každá osoba, která využije výše uvedenou licenci, se však zavazuje udělit ke každému dílu, které vznikne (byť jen zčásti) na základě Díla, úpravou Díla, spojením Díla s jiným dílem, zařazením Díla do díla souborného či zpracováním Díla (včetně překladu), licenci alespoň ve výše uvedeném rozsahu a zároveň zpřístupnit zdrojový kód takového díla alespoň srovnatelným způsobem a ve srovnatelném rozsahu, jako je zpřístupněn zdrojový kód Díla.

V Praze dne 7. května 2017

.....

České vysoké učení technické v Praze
Fakulta informačních technologií

© 2017 Josef Rypáček. Všechna práva vyhrazena.

Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí, je nezbytný souhlas autora.

Odkaz na tuto práci

Rypáček, Josef. *Systém pro sběr a zobrazení sférických panoramat*. Diplomová práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2017.

Abstrakt

Tato diplomová práce se zabývá implementací systému pro sběr, indexaci a zobrazování sférických panoramat. Pojednává o metodách získávání, ověřování a analyzování fotografií z internetu. V práci jsou použity metody pro text mining a zpracování obrazu k získání atributů vhodných k vyhledávání a shlukování panoramat.

Klíčová slova sférické panorama, fotografie, text mining, zpracování obrazu, webová aplikace, klasifikace, shlukování

Abstract

This master thesis deals with an implementation of a system for retrieval, indexing and display of spherical panoramas. It describes methods of retrieval, verification and analysis of photographs from Internet. In the thesis we use methods of text mining and image processing to obtain attributes suitable for searching and clustering of panoramas.

Keywords spherical panorama, photography, text mining, image processing, web application, classification, clustering

Obsah

Úvod	1
1 Analýza	3
1.1 Základní pojmy	3
1.2 Panoramata a jejich projekce	5
1.3 Projekce	7
1.4 Systém pro sběr, indexaci a zobrazení sférických panoramat	11
1.5 Cílová platforma	11
1.6 Zdroje dat	12
1.7 Ověřování panoramat	12
1.8 Analýza fotografií	13
1.9 Procházení a shlukování panoramat	14
1.10 UI a panoramatický prohlížeč	14
2 Návrh	17
2.1 Cílová platforma	17
2.2 Zdroje dat	18
2.3 Ověřování panoramat	20
2.4 Analýza textu	21
2.5 Analýza obrazu	22
2.6 Návrh databázového indexu	23
2.7 Procházení a shlukování panoramat	25
2.8 UI a panoramatický prohlížeč	26
2.9 Wireframe prototyp	27
3 Implementace	31
3.1 Cílová platforma	31

3.2	Zdroje dat	33
3.3	Ověřování panoramat	35
3.4	Analýza textu	40
3.5	Analýza obrazu	41
3.6	Uživatelské rozhraní	52
	Závěr	59
	Literatura	61
	A Seznam použitých zkratk	65
	B Obsah přiloženého CD	67

Seznam obrázků

1.1	Cylindrická projekce s úhlem záběru $360 \times 110^\circ$	7
1.2	Rectilineární projekce s úhlem záběru $150 \times 110^\circ$	8
1.3	Kubická projekce s úhlem záběru $360 \times 180^\circ$	9
1.4	Sférická projekce s úhlem záběru $360 \times 180^\circ$	9
1.5	Stereografická projekce s úhlem záběru $360 \times 180^\circ$	10
2.1	Sférická projekce krajních řádků obrazu	20
2.2	Databázové schéma	24
2.3	Hlavní stránka	28
2.4	Zobrazení mapy	28
2.5	Filtrování na základě obsahu	29
2.6	Detail panoramatu	29
3.1	Analýza barevné vzdálenosti pixelů v rámci řádky	37
3.2	Analýza barevné vzdálenosti zvolených atributů	38
3.3	Matice záměn	39
3.4	Matice záměn pro zvolené hraniční hodnoty	39
3.5	Histogramy nočních (červené) a denních (modré) fotografií	42
3.6	3D RGB histogram [1].	46
3.7	Prostor HSL a rozložení barev	47
3.8	Interaktivní mapa	53
3.9	Filtrování na základě obsahu	53
3.10	Detail zobrazeného panoramatu	56

Seznam tabulek

3.1	Porovnání typů histogramů	43
3.2	Porovnání jemnosti histogramů	44
3.3	Porovnání rozlišení fotografií	44
3.4	Porovnání největší přípustné chyby	45
3.5	Úspěšnost klasifikace v závislosti na skryté vrstvě neuronů	50
3.6	Úspěšnost klasifikace v závislosti na požadované chybě	51
3.7	Úspěšnost klasifikace v závislosti na typu histogramu	51

Úvod

Sférická panoramata jsou v posledních letech velmi často vytvářena a používána v nejrůznějších případech. Na rozdíl od běžné fotografie dokáží zobrazit prostor ve všech směrech a divákovi nabízí možnost se rozhlédnout tak, jako by na daném místě byl přítomen. Této vlastnosti se nejčastěji využívá u mapových služeb, virtuálních prohlídek, pohledů z turistických míst a u brýlí pro virtuální realitu. Na internetu lze nalézt velké množství těchto snímků, služeb které by je vhodně seskupovaly a analyzovaly však mnoho není.

Cíl práce

Tato práce si klade za cíl vytvoření systému pro automatický sběr, indexaci a následné prohlížení sférických panoramat. Pro splnění cíle je nutné navrhnout a realizovat několik dílčích částí celého systému. V první řadě je potřeba zajistit automatické získávání existujících fotografií z internetu. Následně je nutné ověřit, zdali se jedná o sférické panoramatické snímky. Jen ty jsou dále zpracovány a zařazeny do databázového indexu spolu se získanými metadaty. Jednotlivé fotografie a jejich metadata jsou dále zpracovány pomocí metod pro analýzu textu a zpracování obrazu k získání dalších atributů použitelných k rozšíření funkcionality systému. Vytvořená webová aplikace umožňující prohlížení panoramatických fotografií využije získaná data pro filtrování, vyhledávání a shlukování podobných snímků.

Struktura práce

V první kapitole jsou vysvětleny základní pojmy nutné pro porozumění této práci. Nachází se zde informace o panoramatech, jednotlivých projekcích a jejich použití. Dále jsou zde uvedeny dílčí části výsledné aplikace a analýza možných řešení.

Druhá kapitola pojednává o návrhu jednotlivých částí systému a výběru konkrétních metod použitých k implementaci. Jsou zde uvedeny použité zdroje fotografií, získávaná metadata a analyzované vlastnosti sférických panoramat.

Samotné implementaci navržených řešení je věnována třetí kapitola, ve které jsou uvedeny jednotlivé kroky realizace pro všechny dílčí části systému včetně případných výsledků z testování použitých metod.

Analýza

1.1 Základní pojmy

Panorama

Označení panorama se používá pro široký výhled nebo fotografii zachycující více prostoru, než je obvyklé. Nejčastěji vzniká použitím velmi širokoúhlých objektivů nebo spojením několika běžných fotografií vedle sebe. V některých případech se objevují i panorama vertikální, tedy fotografie spojené nad sebou.

Panorama je často využíváno na fotografie pohoří, krajiny nebo interiéru, kde může být spojení několika fotografií jedinou možností, jak zachytit vše potřebné. Dalším důvodem, proč pořizovat a skládat panoramata je zvýšení kvality a rozlišení výsledné fotografie. Tímto způsobem lze dosáhnout rozlišení v řádu stovek megapixelů, což následně umožní přiblížení nebo tisk na velmi velké formáty. Skládání jednotlivých snímků lze tak v praxi využít při pořizování vysoce profesionálních fotografií produktů.

Pro zachycení panoramatické fotografie lze dále použít speciální 360° kamery. Ty mají obvykle nejméně dva přesně seřízené současně snímající objektivy namířené do opačných stran. Pořízení snímku je v tomto případě velmi snadné včetně automatického spojení snímku díky předem vypočítaným parametrům prolnutí. Daní za jednoduchost pořízení panoramatu je nižší kvalita i rozlišení výsledné fotografie. Velkou výhodou většiny těchto zařízení je možnost pořizování 360° videa opět včetně implicitního spojení obrazu z obou objektivů.

API

Application Programming Interface definuje rozhraní pro programování aplikací. V souvislosti s webovým API se jedná o definici metod a protokolů, kterými lze metody volat. Jde tedy o rozhraní služby třetí strany, které výsledný systém používá pro komunikaci se službou k získání potřebných informací.

Typickým příkladem takové komunikace jsou metody volané pomocí protokolu HTTP. Metoda a její parametry jsou určeny adresou, na kterou se požadavek zasílá a odpověď je vrácena v libovolném požadovaném formátu podobně jako běžná webová stránka. Ke komunikaci s takovým API je možné použít například knihovnu cURL, nebo libovolný jiný prostředek podporující HTTP komunikaci.

GPS

Global Positioning System je celosvětový lokační systém přiřazující souřadnice každému místu na Zemi. Vzhledem k tomu, že fotografie mohou být označeny GPS souřadnicemi, je možné zobrazit snímky na mapě a nalézt nejbližší sousední fotografie.

EXIF

Exchangeable Image File Format je způsob, jak jsou do obrazových souborů fotografií vkládána metadata. Nejčastěji je používán pro zápis informací o fotoaparátu a expozici konkrétního snímku. Dále lze vložit doplňující údaje, jako jsou název, popis, klíčová slova a GPS souřadnice. Právě tyto údaje jsou pro shlukování a vyhledávání fotografií velmi důležité.

JSON

JavaScript Object Notation je styl zápisu objektů a dalších struktur v jazyce JavaScript [2]. Tato metoda uložení datových struktur je velmi rozšířená i v ostatních programovacích jazycích a oproti XML je díky stručnějšímu zápisu vhodná pro přenos po síti.

Webové prohlížeče z důvodů bezpečnosti zajišťují tzv. *Same-origin policy*, čímž zamezují skriptu na jedné stránce přistupovat k datům jiné stránky. Pro přístup k datům je nutné povolení na straně cílového serveru (*Cross-origin resource sharing*), nebo použití datového formátu JSONP. *JSON with padding* je metoda použitelná pouze pro čtení dat z jiné stránky. Jedná se

o standardní JSON obalený voláním funkce. V okamžiku načtení dat se zavolá uvedená funkce s daty ve formátu JSON jako parametrem.

Zranitelnost aplikace

Webové aplikace jsou vystavené několika hlavním druhům zranitelností [3]. Ve většině případů lze napadení aplikace předejít správným ošetřením vstupů tak, aby útočník nenarušil běžný chod programu.

Cross-site scripting, metoda známá jako XSS umožňuje útočníkovi do aplikace pomocí neošetřených vstupů formuláře vložit škodlivý kód, který je následně načten běžnými uživateli. Nebezpečný kód může sledovat uživatele nebo napodobit jeho chování. Jako škodlivý kód je obvykle použit JavaScript.

Další zranitelností využívající neošetřené vstupy je *SQL injection*. Tou je možné přímo změnit databázový požadavek a způsobit tak závažné škody v datech aplikace.

Velmi rozšířenou bezpečnostní chybou, která často není ošetřena je *Cross Site Request Forgery*. Při útoku CSRF je přihlášeným uživatelem načtena podvodná stránka, která automaticky zašle falešný požadavek na neošetřený web pod identitou uživatele. Tímto postupem lze simulovat chování přihlášeného uživatele a provádět změny v aplikaci. Útok je snadný především pro aplikace, kde útočník zná metody zpracování formulářů, zejména v administrátorské části aplikace.

1.2 Panoramata a jejich projekce

Samotný pojem panorama nedefinuje přesné požadavky na výslednou fotografii a její formát. K rozlišení jednotlivých typů panoramat a dostupných projekcí je třeba určit, jaký zorný úhel fotografie pokrývá. Panoramata lze na základě tohoto aspektu rozdělit na částečná a úplná.

Částečné panorama

Panoramata, která nepokrývají celý prostor okolo diváka ve všech směrech se označují jako částečná. Běžné je spojení dvou až čtyř sousedních fotografií krajiny do širokého formátu. Tímto způsobem vznikne částečné panorama, které pokrývá pouze část horizontálního i vertikálního zorného pole. Jeho vyfotografování i spojení je jednoduché a tento úkon automaticky zvládá většina moderních mobilních telefonů i kompaktních fotoaparátů.

360° panorama

Pokud je částečné panorama vytvořeno tak, aby pokrývalo úhel 360° v horizontálním směru, lze ho nazvat 360° panoramatem. U tohoto typu je spojování jednotlivých snímků jedinou cestou, neboť neexistuje objektiv s tak širokým úhlem záběru. Při spojování je nutné správně upravit i boční strany výsledné fotografie tak, aby při případném spojení přesně navazovaly. S použitím vhodného panoramatického prohlížeče lze výslednou fotografii prohlížet cyklicky dokola bez viditelného spojení nebo přerušení.

Spojení fotografií tak, aby správně navazovaly i boční strany při cyklickém prohlížení je možné realizovat pomocí specializovaných programů určených pro spojování panoramatických fotografií. Jedním z nich je i volně dostupný *Hugin*. Cyklické panorama lze však ve dvou krocích složit také v libovolném programu umožňujícím spojení fotografií. V prvním kroku je N fotografií spojeno pomocí $N - 1$ spojů. V druhém kroku je spojená fotografie rozdělena v polovině, pořadí těchto polovin je prohozeno a následně jsou tyto 2 fotografie spojeny pomocí 1 spoje. Tímto způsobem vznikne 360° fotografie s navazujícími bočními stranami.

Další výhodou 360° panoramatu je možná změna kompozice pro zobrazení bez panoramatického prohlížeče. Pomocí cyklického posunu lze fotografii a tím i požadovaný objekt posunout do požadované pozice. Toho lze jednoduše docílit opět rozdělením panoramatické fotografie v místě, kde má vzniknout nová strana fotografie a zaměnit pořadí vzniklých polovin.

Úplné panorama

Oproti částečnému 360° panoramatu úplné navíc zachycuje i pohled nahoru a dolů. Výsledný úhel záběru je tedy $360 \times 180^\circ$. Při skládání fotografií jsou ve většině případů spojovány snímky nejen vedle sebe, ale i nad sebou. S využitím speciálních širokoúhlých objektivů je možné složit úplné panorama už ze 2 nebo 3 snímků. Stejného výsledku lze dosáhnout i s použitím běžnějších objektivů s delší ohniskovou vzdáleností, počet potřebných snímků však rychle roste. Běžně jsou úplná panoramata vytvořena ze 20 až 30 snímků. Pro úspěšné spojení je nutnost použití programů pro spojování úplných panoramat nebo speciální aplikace pro mobilní zařízení. Jelikož jsou úplná panoramata často pořizována ze stativu a zabírají i pohled dolů, je vhodné pořídít jednu fotografii navíc bez použití stativu pro jeho následné odstranění při úpravě fotografií v počítači.

1.3 Projekce

Zachycená panoramata lze zobrazit více způsoby. Každá z projekcí má své výhody i nevýhody a je vhodná pro odlišný typ panoramat [4]. Částečná panoramata je možné zobrazit přímo bez prohlížeče, pokud kratší strana pokrývá úhel menší než 120° . Pak již je zkrácení výrazné a pro diváka ve většině případů rušivé. Pro přirozené zobrazení úplných panoramat je tedy obvykle nutné použít specializovaný interaktivní prohlížeč.

Pro názornější ilustraci základních typů projekcí uvedených v této kapitole bylo vytvořeno úplné panorama složené z 18 samostatných snímků. Pomocí programu Hugin [5] byly tyto fotografie spojeny do panoramatu a vygenerovány ukázky pro každou z následujících projekcí.

Cylindrická

Cylindrické panorama tvoří fotografie natažená na stěnu imaginárního válce, kde pozorovatel je uvnitř. Tento typ projekce je vhodný pro 360° panoramata, ve kterých chybí nadir a zenit, tedy pohled nahoru a dolů. Cylindrickou projekci je samozřejmě možné použít i pro snímky pokrývající úhel menší než 360° . Tato projekce je nejčastější a hodí se například pro snímky krajiny. U fotografií architektury může docházet k rušivému prohýbání linií.



Obrázek 1.1: Cylindrická projekce s úhlem záběru $360 \times 110^\circ$

Rectilineární

Planární neboli rectilineární projekce zachovává linie rovné, ale úhel jejího zobrazení je omezený přibližně na 120° . Poté dochází k výrazné deformaci rohů obrazu. Hodí se na částečná panoramata s menším úhlem záběru a především architekturu.



Obrázek 1.2: Rectilineární projekce s úhlem záběru $150 \times 110^\circ$

Kubická

Jednou z prvních úplných projekcí je kubická. Jedná se o 6 snímků, které jsou následně namapovány na stěny virtuální krychle. Pro tuto projekci je panoramatický prohlížeč nutností. Ve zdrojové podobě panoramatu mají jednotlivé snímky minimální zkreslení a lze je snadno upravovat i retušovat.

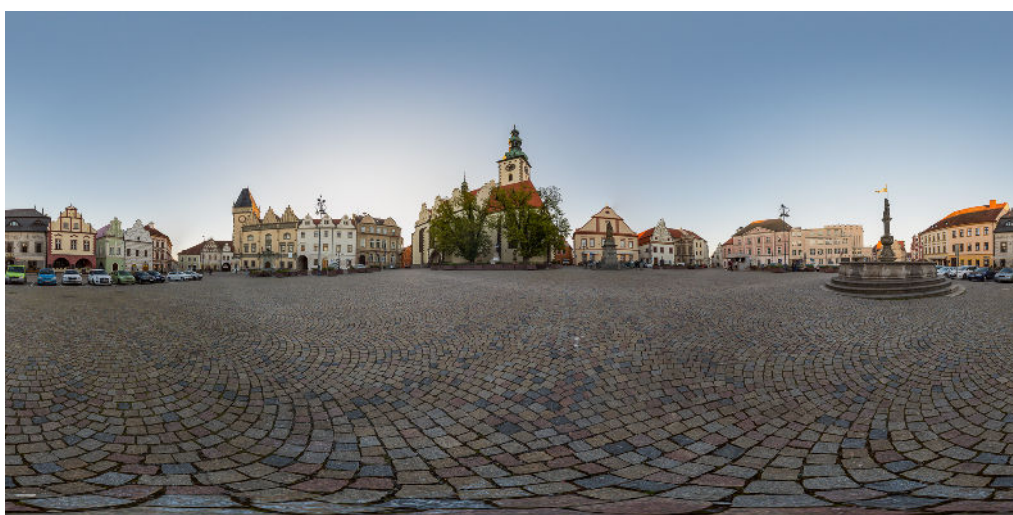
Tento typ zobrazení byl v minulosti používán jako výchozí formát pro panoramata QuickTime VR na platformě Apple [6]. Kubická projekce je poměrně jednoduchá a výkonově nenáročná. Dnes se ve spojitosti s projekcí panoramat používá především pro zařízení s nízkým výkonem a pro starší verze prohlížečů, které nepodporují modernější typy projekcí.

Sférická

Sférická neboli equirectangulární projekce představuje virtuální kouli, na kterou je natažena fotografie pokrývající celý její plášť. Výslednou fotografii lze kromě specializovaného prohlížeče zobrazit i přímo, pouze horní a spodní část je výrazně zdeformovaná. V případě úplného panoramatu je poměr stran výsledné fotografie vždy 2:1. Horní a spodní část výsledné fotografie lze jen obtížně upravovat a retušovat, neboť je zkreslení výrazné a při zobrazení v panoramatickém prohlížeči by výsledek nevypadal dobře.



Obrázek 1.3: Kubická projekce s úhlem záběru $360 \times 180^\circ$



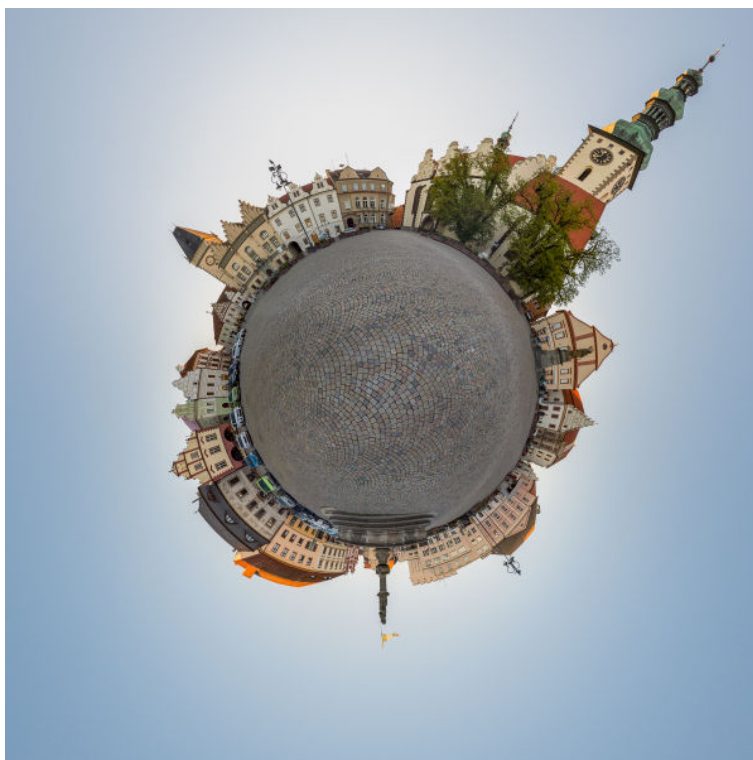
Obrázek 1.4: Sférická projekce s úhlem záběru $360 \times 180^\circ$

1. ANALÝZA

Pro potřeby úprav se sférická panoramata dočasně převádí do projekce jiného typu (např. kubické). Při zobrazení sférické fotografie bez prohlížeče představuje prostřední část obrazu o šířce 25 % zorné pole před divákem. Čtvrtiny vlevo a vpravo od střední části jsou zobrazením po levé a pravé straně diváka. Pohled za divákem je viditelný na levém a pravém kraji vždy o šířce 12,5 % z celkové šířky fotografie.

Stereografická

Jednou z dalších metod projekce je stereografická, která mapuje neobvyklým způsobem úplné panorama do plochého snímku. Často je označována jako Little Planet a pro vytvoření lze využít i částečná panoramata. Poměr stran výsledné fotografie je obvykle 1:1. Vzhledem k tomu, že tento typ projekce nepotřebuje prohlížeč, může se poměr stran lišit v závislosti na autorově záměru. Střed fotografie připomínající planetu je tvořený vodou nebo zemským povrchem. Naopak kraje jsou tvořeny oblohou. Dobré výsledky lze získat z panoramat náměstí, jejichž výsledek připomíná právě malou planetu.



Obrázek 1.5: Stereografická projekce s úhlem záběru $360 \times 180^\circ$

1.4 Systém pro sběr, indexaci a zobrazení sférických panoramat

Existuje několik webových aplikací, které nabízí částečně podobnou funkcionalitu, jakou si klade za cíl tato práce. Jednou z nich je stránka `2vr.in`, která umožňuje pouze jednoduché sdílení a zobrazení uživatelských panoramat bez možnosti prohledávání. Další aplikace `airpano.com` se zabývá sférickými panoramaty pořízenými ze vzduchu úzkou skupinou autorů. Panoramata lze vyhledávat na mapě a lze přecházet mezi sousedními panoramaty. Třetí a nejpropracovanější systém nabízí `360cities.net`. Zde se nachází sférická panorama libovolného typu od mnoha autorů. Mezi panoramaty je možné vyhledávat textově, nebo je vybírat na mapě podle uvedené GPS lokace. Stránka je komerčně zaměřená na prodej obrazových dat i na prémiové uživatelské účty nabízející pokročilé možnosti. Panoramata jsou ručně nahrávána do systému se všemi potřebnými údaji (název, popis, pozice), které slouží pro vyhledávání.

Žádná z uvedených aplikací nezískává panoramatické fotografie ze zdrojů třetích stran. Metadata jsou získána přímo od autora fotografie a systém nenabízí pokročilé možnosti vyhledávání a filtrování. Výsledná aplikace této práce si klade za cíl získávání fotografií i metadat z internetu a využití metadat získaných z API i analýzy textu a obrazu k vyhledávání, filtrování a shlukování souvisejících panoramat.

1.5 Cílová platforma

Požadovaný výstup této práce je multiplatformní webová aplikace přístupná přes libovolný webový prohlížeč. Nejrozšířenější a volně dostupnou technologií pro tvorbu webových aplikací je programovací jazyk PHP, který lze provozovat na širokém spektru webhostingových služeb. Zároveň existuje velké množství volně dostupných knihoven od samotných uživatelů pro řešení častých problémů. Uložení metadat a dalších informací nutných k provozu systému je vhodné realizovat pomocí databáze. Mezi ty nejčastěji používané nejen pro webové aplikace patří MySQL a PostgreSQL.

Jazyk PHP má zabudovanou podporu pro volání knihovny ImageMagick a GD, které lze využít pro práci s obrázky. Dále je možné pro realizaci jednotlivých částí aplikace využít také další programovací jazyky a tyto moduly následně volat z jádra aplikace. Moduly lze naprogramovat v libovolném programovacím jazyce (například C, Java, Python).

1.6 Zdroje dat

Jednou z primárních funkcí systému je získání fotografií z internetu. Nejúčinnější metodou je prohledávání sociálních sítí přímo určených pro sdílení fotografií. Tyto sítě často poskytují API, podporují filtrování, obsahují velké množství obrazových dat, a to včetně souvisejících metadat. Ve většině případů je u fotografie uvedena také informace o autorovi a použité licenci. Další možností je využití obrázkových vyhledávačů, mezi které patří picsearch.com a většina předních fulltextových vyhledávačů. Tento způsob však postrádá výhody API u specializovaných sítí a především nenabízí informaci o autorovi, zvolené licenci a možnost získat obrazová data. Tato práce se zaměřuje na získávání dat ze sítí pro sdílení fotografií s přístupem přes API.

Mezi nejvíce používané sítě s API patří Flickr, 500px, Instagram, DevianArt, Imgur, SmugMug a Photobucket. Flickr obsahuje největší množství fotografií a informací o nich, má rozsáhlé API a umožňuje pokročilé filtrování fotografií. Sít 500px také nabízí propracované API vhodné pro tuto práci, množství fotografií je ale menší. API sítí DevianArt a Instagram nejsou na podrobné vyhledávání a získávání požadovaných fotografií připraveny. Obsahují minimum sférických panoramat a není zde kladen důraz na přidružená metadata.

Některé sítě pro sdílení fotografií (např. Flickr) samy podporují zobrazení sférických panoramat pomocí jednoduchého prohlížeče. Samotná panoramata ale není možné vyfiltrovat, nelze zobrazit tématicky související snímky a sférický prohlížeč poskytuje pouze základní funkcionalitu. Zobrazený snímek nelze přibližovat, nelze ho přepnout do módu brýlí pro virtuální realitu, ani ho zobrazit přes celou obrazovku.

1.7 Ověřování panoramat

Úplné panorama ve sférické projekci má poměr stran 2:1. Toto je základní vlastnost, podle které lze fotografie ověřit. Na většině exteriérových fotografií se horizont vyskytuje v polovině obrazu a je rovný. Horní část zabírá obloha (typicky modrá barva), spodní část zem (zelená nebo šedá barva). Horní a dolní části obrazu jsou výrazně zdeformované a obsahují velké množství redundantních informací čím blíže jsou ke hraně fotografie. Je zde tedy předpoklad, že první a poslední řádka pixelů má velmi podobnou barvu. Pokud je panorama správně vytvořeno, pak pravá a levá část na sebe přesně navazují a krajní sloupce obrazových bodů mají podobný obsah. Fotografie mohou mít informaci o použité projekci uvedenou v EXIF údajích. Na pa-

noramatické fotografii dále mohou být viditelné spoje a to vlivem několika faktorů.

Jednotlivé snímky, ze kterých je panorama složeno lze odlišit, pokud nebyla správně odstraněna vinětace objektivu. V tom případě se přechody mezi snímky vyznačují tmavšími barvami. Samotné spoje nemusí správně navazovat vlivem nepřesného spojení, posunu fotoaparátu během snímání, pohybu objektů na fotografované scéně nebo mírnou změnou zaostření či ohniskové vzdálenosti objektivu. Výraznější nepřesnosti spojů na objektech umístěných blíže fotoaparátu ukazují na paralaxní chybu, která se objevuje, pokud při snímání panoramatu fotoaparát nerotuje okolo nodálního bodu.

Nodální bod je střed optické soustavy, který zaručuje následnou možnost snadného a přesného spojení fotografií. Pro tento účel se především u sférických panoramat interiéru používá panoramatická hlava, která umožňuje správné nastavení nodálního bodu a uchycení fotoaparátu.

1.8 Analýza fotografií

Získané fotografie, které jsou sférickými panoramaty je vhodné analyzovat pro vytěžení maxima informací a tyto údaje následně použít ke shlukování výsledků.

Prvním způsobem je analýza metadat získaných z API přidružených k fotografii. Mezi ty nejzajímavější patří název, popis, klíčová slova a GPS souřadnice. Štítky neboli klíčová slova jsou přiřazeny u většiny fotografií a i bez další analýzy je možné podle nich fotografie shlukovat. Zeměpisné souřadnice mohou být použity pro zobrazení na mapě nebo převedeny na skutečnou adresu. Z té jsou pro aplikaci zajímavé informace o zemi a městu, ve kterém se fotografie nachází. Pokud není uvedena zeměpisná poloha, lze ji v některých případech částečně odvodit z textových dat. V popisu fotografie bývá často uvedeno i prostředí fotografie a lze tak určit, kde bylo panorama pořízeno, případně jaký objekt je na něm zachycený. Pro analýzu textu i následnou možnost vyhledávání je vhodné nejprve přeložit textová data všech získaných fotografií do jednoho společného jazyka.

Samotné fotografie lze dále analyzovat graficky. Pro účely shlukování postačí i jednoduché metody pro zpracování obrazu, které určí převažující barvu nebo světlost snímku. Další možností analýzy je detekce základních tvarů, osob nebo analýza podobnosti jednotlivých fotografií. Pro pokročilejší analýzu je možné využít prostřednictvím API specializované služby pro rozpoznávání obrazu založené typicky na hlubokých neuronových sítích.

1.9 Procházení a shlukování panoramat

Shluková analýza slouží k seskupení podobných fotografií do skupin. Shlukování může probíhat na základě libovolných atributů zařazených do indexu získaných při analýze. Atributy lze použít také pro filtrování a vyhledávání. Běžné je seskupování fotografií podle jejich zeměpisné polohy. Dalšími metodami, které nejsou tolik obvyklé je shlukování podle typu snímku. Panorama lze rozdělit na denní a noční, interiérová a exteriérová nebo z města a z přírody. Panoramata lze dále seskupovat dle typu fotografie získaného z textových dat. Je možné seskupit zámky, náměstí, kostely a další typy fotografovaných míst.

Získané atributy a jejich shlukování je vhodné použít jak při procházení panoramatickým indexem dle vybraných vlastností, tak při zobrazení konkrétní fotografie k zobrazení návrhů na další snímky s podobnou tematikou.

1.10 UI a panoramatický prohlížeč

Vzhledem k cílové platformě webové aplikace je potřeba uživatelské rozhraní postavit na technologiích pro tvorbu webových stránek. V dnešní době, kdy jsou stále častěji pro prohlížení internetu používané mobilní telefony a tablety je důležité navrhnout design responzivní a veškeré technologie volit tak, aby správně fungovaly i na mobilních zařízeních. K vytvoření responzivního designu lze použít HTML, CSS a JavaScript.

Značkovací jazyk HTML definuje obsah stránky a jeho základní uspořádání. Kaskádové styly CSS určují výsledný vzhled a pomocí technologie Media Queries se starají se o rozvržení responzivního designu na jednotlivé typy zařízení podle velikosti jejich obrazovky. JavaScript je skriptovací jazyk spouštěný na straně klienta umožňující vytvoření dynamických stránek, animací a modifikaci obsahu stránky za běhu.

Uvedených technologií využívají i frameworky specializované na responzivní design a návrh uživatelského rozhraní webových stránek obecně. Kromě úspory času nabízejí kompatibilitu s širokou škálou webových prohlížečů a předpřipravené styly běžně používaných elementů, jejichž použití vede k návrhu čistého, konzistentního a použitelného webového uživatelského rozhraní. Mezi nejznámější frameworky tohoto typu patří Bootstrap, Skeleton a Foundation.

Důležitou součástí systému je prohlížeč panoramatických fotografií. Pro zachování maximální kompatibility je nutné zvolit technologie dostupné na všech platformách, včetně mobilních telefonů. Z tohoto důvodu není vhodné využití oblíbené technologie Adobe Flash, neboť na mnoha platformách již

není podporována nebo její podpora bude brzy zrušena. Jedinou možností je využití technologie HTML5 doplněné JavaScriptem. Jedním z kandidátů na panoramatický prohlížeč je Pannellum, který je open source, podporuje několik druhů projekcí, panoramata ve vysokém rozlišení a to ve většině moderních prohlížečích. Mezi další panoramatické prohlížeče patří Marzipano, Leanorama a VR5 Pano Viewer. Existuje mnoho dalších výborných prohlížečů jako PTViewer, Pano2VR, Panoweaver nebo KR pano. Ty jsou ale ve většině případů placené a využívají primárně technologii Flash nebo Java Applet.

Návrh

V kapitole 1 je uvedeno mnoho směrů a možností ověřování i analýzy panoramat, které je možné realizovat. Jedná se především o metody založené na pokročilém rozpoznávání textu a obrazu. V samotném návrhu a realizaci této práce je počítáno s využitím existujících knihoven i implementací vlastních metod v závislosti na jejich účinnosti. Při návrhu aplikace je počítáno s možnostmi budoucího rozšíření systému o další zdroje, metody ověřování a analýzy fotografií.

2.1 Cílová platforma

Primární platformou a základním programovacím jazykem pro webovou aplikaci je zvoleno PHP v aktuální verzi 7.1. S tím úzce souvisí i výběr frameworku, jehož úkolem je vedení programátora k využití správných návrhových vzorů, zjednodušení nejčastěji implementovaných funkcionalit a zajištění vysoké bezpečnosti aplikace proti zranitelnostem jako jsou XSS, CSRF a SQL injection. V neposlední řadě framework často poskytuje databázovou vrstvu pro snadný a bezpečný přístup do databáze a šablonovací systém pro kompletní oddělení uživatelského rozhraní od logiky aplikace. Český framework Nette ve verzi 2.4 splňuje všechny výše uvedené požadavky a jeho zvolení bylo provedeno na základě předchozích zkušeností a rozsáhlé komunity.

Ačkoli PHP nebývá považováno za nejvýkonnější programovací jazyk, jeho rychlost i paměťová náročnost se výrazně zlepšila s příchodem přepracované verze 7 [7]. V případě, že by PHP nebylo pro některé účely dostatečující nebo vhodné lze z aplikace volat moduly napsané v libovolném jiném programovacím jazyce.

2.2 Zdroje dat

Na sítích určených pro sdílení fotografií je umístěno velké množství pořízených snímků a není je technicky možné všechny jednotlivě procházet a ověřovat, zdali jsou sférickými panoramaty. Z toho důvodu je vhodné požadované fotografie filtrovat už na úrovni API konkrétní služby. Parametry filtrování, které by umožnily získat pouze panoramatické snímky s poměrem stran 2:1 nenabízí API žádné služby. Jedinou možností zbývá filtrace fotografií pomocí textu v názvu, popisku nebo klíčových slovech fotografie.

Vhodná slova pro vyhledávání panoramatických fotografií jsou například *equiangular*, *spherical* nebo *panorama*. V textu se dále často vyskytuje *360* nebo *hugin*, což je oblíbený nástroj na spojování panoramat všech typů. Na vyhledávací dotazy s tímto omezením jsou z větší části vrácena sférická panoramata, dále jde o panoramata částečná a běžné fotografie. Tyto případy má za úkol vytřídit následný algoritmus pro ověřování sférických panoramat.

Dalšími parametry pro vyhledávání je vhodné specifikovat pouze fotografie s volnou licencí, aby nedošlo k porušení autorského práva. V současném stavu jsou získávány pouze fotografie s licencí Creative Commons [8] libovolného typu a pro splnění licenčních podmínek nejsou fotografie upravovány ani použity ke komerčním účelům.

2.2.1 Flickr

Jako první a primární zdroj fotografií je vybrána síť Flickr, která obsahuje velké množství snímků včetně těch panoramatických. Služba poskytuje bohaté API, podporu několika formátů požadavků i odpovědí a ukázky pro více než 15 programovacích jazyků. Pro komunikaci se službou je potřeba získat API klíč, který slouží k ověření aplikace, sledování dotazů na API a zajištění limitů. API klíč lze získat po registraci, celý proces je rychlý a nepotřebuje schválení aplikace. Limity počtu přístupů jsou nastaveny na 3 600 dotazů za hodinu [9]. Při vkládání fotografií na jiné webové stránky je povoleno zobrazit maximálně 30 snímků na jednu stránku.

API služby Flickr dovoluje při požadavku na vyhledávání fotografií definovat vlastnosti, které mají být spolu s výsledkem vráceny. To pomáhá silně redukovat počet potřebných dotazů na API i čas běhu aplikace. Pro získání informací o nejvýše 500 fotografiích stačí jediný dotaz na API.

Vyhledávání lze otestovat pomocí klíčového slova *equiangular*. Počet všech nalezených výsledků je 38 tisíc. Dle předpokladu se ne ve všech případech jedná o sférická panoramata a mezi výsledky jsou i panoramata částečná s jinými typy projekce, v menším počtu pak běžné fotografie. S po-

žadavkem na licence Creative Commons se počet nalezených fotografií snížil na třetinu.

Flickr API poskytuje možnost prohledávat i EXIF fotografie. Pokusy o nalezení výsledků pomocí vlastnosti *ProjectionType* i jmenných prostorů *XMP* a *GPano* skončily neúspěšně. Jediným parametrem, který vedl k nalezení nějakých výsledků je jmenný prostor *panorama* a vlastnost *projection*. Z celé databáze fotografií bylo nalezeno pouhých 11 výsledků. Z toho důvodu bylo filtrování podle EXIF údajů zavrženo a výsledná aplikace používá filtrování pouze na základě textu. Zde není jasné, zdali se jedná o chybu v API, nebo fotografie neobsahují uvedené EXIF vlastnosti.

Nejdůležitější metodou API je *flickr.photos.search*, která slouží k vyhledávání fotografií na základě zvolených parametrů. Údaje vrácené spolu s výsledky lze přizpůsobit, nelze však vybírat ze všech dostupných vlastností. Pro podrobnější informace lze použít metodu *flickr.photos.getInfo*, která vrátí větší množství informací o jedné fotografii. Výsledek stále neobsahuje například informace o počtu lidí, kterým se konkrétní fotografie líbí. K získání dalších informací je tedy nutné použití dalších metod. Volání několika metod pro každou fotografii zvláště značně zvyšuje nároky nejen na výsledný systém, ale také na využití limitů API. Z toho důvodu je v této práci počítáno pouze s využitím informací z metody *flickr.photos.search*, která nabízí všechny důležité údaje potřebné k zařazení do indexu. Doplňující informace, budou-li potřeba, lze doplnit kdykoli později a rozšířit tak funkčnost aplikace.

2.2.2 500px

Druhou ze sítí, která je použita k získávání fotografií je 500px. Její API je velmi dobře zdokumentované, dodržuje principy REST API, a odpovědi jsou vráceny ve formátu JSON. Pro komunikaci se službou je nutná registrace a následné získání tokenu pro vytvořenou aplikaci. Dokumentace API stanovuje limity na 1 000 000 dotazů za měsíc [10]. Počet fotografií zobrazených na jedné stránce je omezen na 20.

Funkcionalita poskytovaná přes API je znatelně menší než u konkurenční služby Flickr. Hlavní metodou *photos/search* je opět umožněno vyhledávání fotografií na základě zvolených parametrů. Vracené výsledky obsahují téměř všechny informace o nalezených fotografiích a použití dalších metod není potřeba. Ve vyhledání fotografií je možné specifikovat požadované licence, práce s EXIF údaji chybí úplně.

Nevýhodou API je absence možnosti stanovení datumu při vyhledávání fotografií. Nelze se tak omezit pouze na nové fotografie nahrané od poslední kontroly, ale je nutné znovu projít všechny vrácené výsledky.

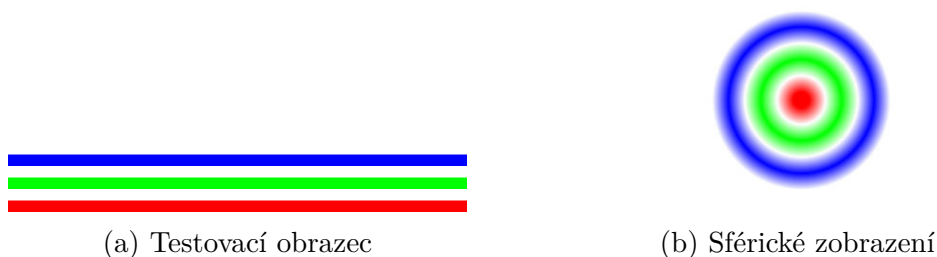
2.3 Ověřování panoramat

Fotografie získané prostřednictvím API nemusí být nutně sférickými panoramaty. Na základě konkrétního vyhledávacího dotazu je určité množství fotografií jiného typu, než aplikace požaduje. Každou získanou fotografií je nutné ověřit a rozhodnout, zdali se jedná o panorama. Samotné ověření lze provádět několika způsoby podle společných znaků sférických panoramat.

Nejpřímochařejší řešení je kontrola návaznosti bočních stran panoramatu. Obrazové body ležící na opačných stranách jedné řádky jsou při sférické projekci zobrazeny vedle sebe. Je zde tedy velká pravděpodobnost, že barevná hodnota těchto pixelů bude podobná. Kromě podobnosti obrazových bodů je možné kontrolovat navazující linie a tvary. Aplikace pro spojování panoramatických fotografií nelze k ověření použít, neboť jejich algoritmy hledají společné body a potřebují fotografie s překryvem, který u výsledného panoramatu vždy chybí.

Ověření podle bočních stran nemusí vždy fungovat podle očekávání. Algoritmus může za panorama označit fotografii předmětu s jednobarevným pozadím, kde se boční sloupce pixelů velmi dobře shodují. Taktéž fotografie krajiny, kde převládá jednoduchá plocha jako je obloha má za následek vysokou pravděpodobnost panoramatického ověření. Naopak zavržení panoramatu může být způsobeno výrazným šumem, špatně navazujícími stranami vlivem vertikální či horizontálního posunu fotografie nebo rozdělením panoramatu ve výrazné hraně, jako je například roh budovy.

Další metodou pro ověření sférických panoramat je analýza horní a spodní řádky pixelů. Krajiní řádky obsahují značně zdeformované a redundantní obrazové informace a lze očekávat, že barevná hodnota pixelů bude podobná. Pro ověření předpokladů bylo vytvořeno několik testovacích obrazců, jejichž cílem je prozkoumat princip projekce ve vrchní a spodní části panoramatu. Na obrázku 2.1a lze vidět zvětšený výřez spodní části testovacího obrazce s červenou linkou na posledním řádku. Sférická projekce spodní části téhož testovacího obrazce je viditelná na obrázku 2.1b.



Obrázek 2.1: Sférická projekce krajních řádků obrazu

Obrázek 2.1 znázorňuje předpoklad podobné barevné hodnoty pixelů v krajních řádcích sférického panoramatu. Nejedná se o jediný bod, ale o nejmenší možnou kružnici zobrazenou při sférické projekci.

Pokud ověření touto metodou skončí úspěšně, může jít o sférické panorama, nebo fotografii s jednobarevným pozadím alespoň v oblasti krajů. Neúspěšné ověření může nastat z důvodu špatně spojeného panoramatu, výrazného vzoru na horní nebo spodní části nebo pokud je vstupem běžná fotografie. Problémový výrazný vzor ve středu horní a spodní části nejčastěji vzniká větvemi stromů nebo šachovnicovou podlahou. Taková panoramata systém pravděpodobně označí za běžné fotografie.

Při analýze podobnosti pixelů je porovnávána jejich barevná hodnota. U běžných fotografií s 8 bitovou barevnou hloubkou se každý pixel skládá ze tří barevných kanálů RGB s rozsahem hodnot 0 až 255. Kontrola podobnosti může kromě naivní metody porovnávající jednotlivé kanály zvláště, používat také euklidovskou vzdálenost v trojrozměrném prostoru. Některé zdroje zmiňují také pokročilejší metody aproximující citlivost vnímání lidského oka na jednotlivé barvy [11].

2.4 Analýza textu

Text mining je metoda, která umožňuje z textu získávat zajímavé a cenné informace. Při získávání panoramat jsou kromě samotných fotografií indexovány také související metadata. Jedná se o název a popis fotografie, zeměpisné souřadnice a štítky. Název a popis jsou pro text mining velmi vhodné. Z těchto údajů je pro výslednou aplikaci přínosné získání informací o místu pořízení fotografie a o zachyceném objektu, které se mohou v textu vyskytovat.

Štítky už danou fotografii reprezentují a mohou být přímo využity pro vyhledávání, filtrování nebo shlukování. Zeměpisné souřadnice udávají přesnou polohu a lze je zobrazit na mapě, nebo použít k nalezení nejbližších panoramat. Pro vyhledávání a filtrování uživatelem je vhodné GPS souřadnice dále analyzovat.

Geocoding je metoda přidávání metadat obsahujících zeměpisné souřadnice k elementům, u kterých známe jejich adresu. Reverzní geocoding naopak umožňuje GPS pozici převést na přibližnou adresu. Z této získané adresy jsou zajímavé především informace o státech a městech, ve kterých byla fotografie pořízena. Takto získané údaje jsou velmi dobře použitelné k vyhledávání i filtrování. Ve spojení se štítky a analýzou textu vytváří komplexní index umístění pořízených fotografií. Mezi služby poskytující geo-

coding patří Google Geocoding API, OpenCage Geocoder API, Mapquest Geocoding API a Yahoo! Geocoding API.

Název a popis fotografie může být uveden v libovolném jazyce. Více než polovina všech fotografií má údaje vyplněné v angličtině. Text zbývajících fotografií je tedy vhodné přeložit také do angličtiny, aby mohl být použit při dalším zpracování. Pro přeložení do anglického jazyka lze použít API služeb SDL, Microsoft Translator, Yandex Translate nebo Google Translation. U získaných údajů není znám jazyk obsahu, proto je nutné, aby překladač podporoval automatické rozpoznání jazyka.

Název a popis fotografie v anglickém jazyce je použit k dalšímu zpracování textu pomocí metody *named entity recognition* rozpoznávající pojmenované entity v textu. Pro účely této aplikace jsou zajímavé entity reprezentující lokaci. Získané entity lze použít k podrobnějšímu vyhledávání, filtrování a shlukování.

Pro rozpoznávání entit existují knihovny i API služeb třetích stran. Lze použít volně dostupnou a oblíbenou knihovnu Stanford Named Entity Recognizer nebo API služeb Thomson Reuters Open CalaisTM a Google Natural Language. Během testování bylo zjištěno, že nejlepší výsledky pro vstupní data podává API Google Natural Language, neboť entity typu lokace rozděluje na dvě samostatné skupiny. První z nich odpovídá skutečnému geopolitickému rozdělení, kam patří státy, města a další oblasti. Druhá skupina sdružuje obecná místa, která mohou být lokacemi. Příkladem je kino, zámek nebo letiště. Právě tyto místa určitým způsobem popisují objekt zachycený na fotografii a shlukování podle nich je zajímavou alternativou k obvyklým typům shlukování.

2.5 Analýza obrazu

K získání dalších atributů do indexu lze použít metody image miningu. Oproti analýze textových údajů atributy vytěžené z obrazu vždy přímo vypovídají o vlastnostech zobrazeného panoramatu. U fotografií bez uvedených textových metadat je obrazová analýza jedinou možností k zajištění shlukování podobných fotografií. Pro analýzu lze použít pokročilých služeb třetích stran i vlastních algoritmů pro získání základních parametrů způsobených na míru požadavkům aplikace. Základní technikou vlastních algoritmů je použití histogramů různých typů při analýze.

Histogram je běžně používaným nástrojem při úpravě fotografií a zobrazuje distribuci barev, respektive jejich intenzit v obraze. Základní histogram se vytváří pro jednotlivé barevné kanály RGB. Jejich zprůměrováním lze získat histogram černobílé verze fotografie. Jednou z informací, kterou

je možné z histogramu vyčíst je světlost snímku. Toho lze využít pro analýzu denní doby zachycené na fotografii. Z barevného histogramu je dále možné určit podíl červené, zelené a modré barvy. Pro pokročilou detekci četnosti barev to ale není vhodné řešení. Histogram lze vytvořit takovým způsobem, aby z něj bylo možné přímo určit četnost jednotlivých barevných odstínů. Toho lze kromě analýzy nejčastějších barev využít také k rozpoznávání prostředí fotografie. V případě převažující modré barvy na horní polovině snímku jde pravděpodobně o oblohu a tedy fotografii z exteriéru.

2.6 Návrh databázového indexu

Každá nalezená fotografie musí projít ověřením a případným vložením do indexu. Vzhledem k možné složitosti celého ověřovacího procesu jsou do databáze ukládány i fotografie, které splňují pouze základní podmínky, ale zatím neprošly kompletním ověřením. Tento přístup poskytuje možnost odložit ověřovací proces na později a odděluje ho od získávání dat. Dále dovoluje uchovávat informace o fotografiích, které ověřením neprošly a lze je při příští úplné aktualizaci indexu rychle přeskočit.

Fotografie, které byly nalezeny a jejich poměr stran odpovídá sférické projekci, jsou vloženy se všemi získanými informacemi do databázové tabulky a čekají na následné zpracování. Prvním krokem je ověření. Je-li neúspěšné, pak lze identifikátory jednotlivých fotografií smazat, nebo ponechat pro urychlení v budoucnu. Na úspěšně ověřená panoramata je dále aplikována analýza textu a obrazu. Během analýzy jsou do indexu doplňovány další vlastnosti vytěžené z dat. Pouze ověřené a analyzované fotografie jsou zobrazeny ve webové aplikaci uživatelům a počítají se z nich celkové statistiky.

2.6.1 Index

Databázový index obsahuje unikátní ID záznamu, název služby, datum nalezení fotografie a příznak, jestli byla daná fotografie ověřena nebo analyzována.

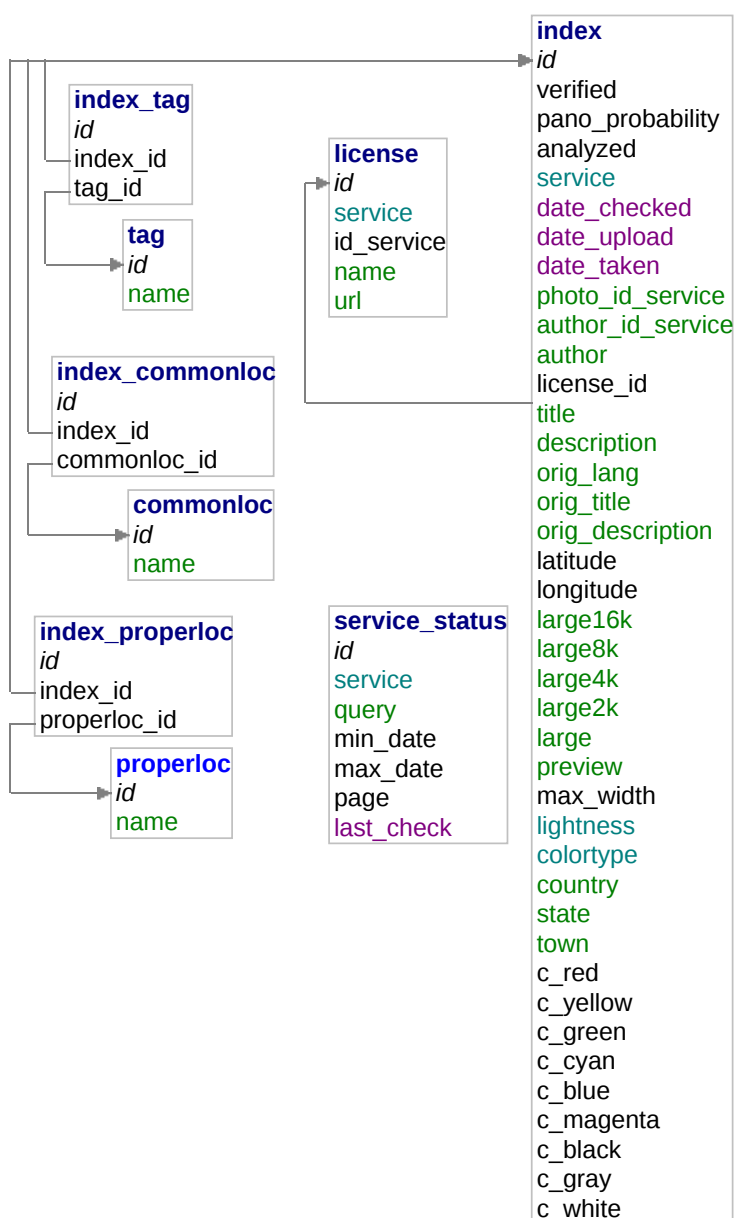
Z API jsou doplněny datумы pořizení a nahrání fotografie, jméno autora, ID autora ve zdrojové službě, informace o licenci, název, popis a GPS souřadnice fotografie. Metodou M:N dekompozice jsou uloženy štítky jednotlivých fotografií. Index dále obsahuje URL adresy fotografie v několika rozměrech a šířku maximálního dostupného rozlišení.

Během textové analýzy probíhá doplnění názvu a popisu přeloženého do anglického jazyka a adresy získané ze zeměpisných souřadnic. Běžná a

2. NÁVRH

přesná umístění získaná z textových metadat jsou uložena pomocí M:N dekompozice do zvláštních tabulek. Analýza obrazu doplní index údaji o světlosti, prostředí a poměru barev přítomných v obraze získaných z vlastních algoritmů pracujících s barevnými histogramy.

Separátní tabulka *service_status* uchovává informace o aktuálně zpracovávaných úlohách získávání dat ze sítě pro sdílení fotografií.



Obrázek 2.2: Databázové schéma

2.7 Procházení a shlukování panoramat

Hlavní podstatou aplikace je využití získaných dat pro shlukování, filtrování a vyhledávání v indexu panoramat. Jednotlivé fotografie je možné procházet kombinací výběru ze všech dostupných parametrů získaných analýzou textu a obrazu. V textových atributech je umožněno vyhledávání a zeměpisné souřadnice jsou využity k zobrazení pozice panoramat na mapě. Pro každé zobrazené panorama jsou nabízeny podobné fotografie v závislosti na společných atributech.

Procházení panoramat je možné započít na stránce s atributy získanými z textu nebo obrazu. Pro každou skupinu vlastností jsou zobrazeny nejčastější atributy. Výběrem ze seznamu dojde k zobrazení stránky připravené pro podrobné filtrování dle všech dostupných atributů a zobrazí se pouze fotografie odpovídající výběru. Seznamy pro výběr atributů se také upraví tak, aby odpovídaly aktuálnímu výběru a jejich zvolením se uživatel nemohl dostat do stavu, ve kterém by systém nezobrazil žádné výsledky.

Po zadání hledaného výrazu a potvrzení formuláře je zobrazena stejná stránka jako v případě filtrování. Zobrazené výsledky i nabízené atributy odpovídají vyhledávacímu dotazu a panoramata lze dále filtrovat dle dostupných vlastností získaných z analýzy textu a obrazu. Vyhledávání je také jedinou výjimkou, kdy může dojít k nenalezení žádných výsledků.

Panoramata s přesně definovanou zeměpisnou pozicí jsou zobrazena na interaktivní mapě vytvořené pomocí Google Maps JavaScript API. Tato služba poskytuje rozhraní k vytvoření plnohodnotné mapy s možností umístění vlastních značek na požadovaná místa. Pro velké množství zobrazených panoramat je vhodné značky shlukovat pomocí rozšíření MarkerClusterer.

Hlavní stránka webové aplikace náhodně zobrazuje doporučená panoramata pomocí předem definovaných podmínek. Vyloučeny jsou například fotografie s nízkým rozlišením nebo snímky bez zadaných metadat, u kterých není dostatek zajímavých informací.

Zvolení konkrétní fotografie na libovolné stránce vede k zobrazení detailu s interaktivním panoramatickým prohlížečem. Současně jsou vypsány také všechny parametry získané z API a pomocí metod pro zpracování textu i obrazu. Zobrazení návrhů na podobné snímky je realizováno na základě podobnosti jednotlivých atributů zařazených do indexu. Pro fotografie se zadanými souřadnicemi jsou dále zobrazeny nejbližší sousední snímky a jejich vzdušná vzdálenost.

2.8 UI a panoramatický prohlížeč

K vytvoření uživatelského rozhraní postaveného na technologiích HTML, CSS a JavaScript je použitý Bootstrap framework. Ten zajišťuje především responzivní design pro maximální uživatelský komfort zejména při použití mobilních zařízení s menším displejem. Dále je vhodný k zachování jednotného a přehledného vzhledu v celé aplikaci. O zobrazování samotných panoramatických fotografií se stará interaktivní prohlížeč pannellum.

Sítě pro sdílení fotografií, ze kterých jsou panoramata získána, ve svých podmínkách omezují možnost ukládání fotografií na vlastní server na pouhou možnost cachování po dobu nezbytně nutnou, nejvíce však na několik dní. To prakticky vylučuje možnost fotografie stahovat, trvale uchovávat a zobrazovat ze svého serveru v přesně definovaném rozlišení. Veškerá panoramata jsou tedy zobrazena přímo ze serverů konkrétních služeb v nejvhodnějším rozlišení pro daný účel. Fotografie zařazené do indexu mají uloženy URL adresy na několik velikostí rozlišení obrazu. Tyto adresy jsou použity k zobrazení náhledů i fotografií ve vyšším rozlišení pro panoramatický prohlížeč.

Příliš vysoké rozlišení fotografií může způsobit kromě pomalého načítání stránek také úplnou nefunkčnost panoramatického prohlížeče. K zobrazení je tedy zvoleno nejvyšší dostupné rozlišení, které je menší nebo rovno limitu panoramatického i webového prohlížeče.

Toto omezení neplyne přímo ze samotného HTML5 elementu Canvas, ale z parametru `MAX_TEXTURE_SIZE` technologie WebGL, která se stará o vykreslování grafiky právě pro HTML5 Canvas [12]. Tento parametr omezuje velikost jedné textury, v tomto případě panoramatické fotografie, kterou je prohlížeč schopen zobrazit. Dle [13] podporují všechny dnešní prohlížeče textury o velikosti 4096 x 4096 pixelů. 75 % uživatelů je ve svém prohlížeči schopno zobrazit textury o rozlišení 8192 x 8192 px a jen polovina uživatelů má webový prohlížeč s podporou textur o rozlišení 16384 x 16384 obrazových bodů.

Maximální podporované rozlišení textur technologií WebGL je možné získat pomocí volání metody `getParameter()` s uvedením požadovaného parametru `MAX_TEXTURE_SIZE`, nebo na diagnostické stránce [14]. Prohlížeč Chromium 57 (Linux) podporuje textury o rozlišení 8192 pixelů. Prohlížeče Firefox 53 (Linux) a Google Chrome 57 (Android) jsou schopny zobrazit textury s rozlišením nejvýše 4096 px. Ve výsledné aplikaci je možné zvolit rozlišení fotografie automaticky v závislosti na podpoře aktuálního prohlížeče získané výše uvedenou funkcí `getParameter()`.

Pro zobrazení panoramat ve velmi vysokém rozlišení je běžně využíváno techniky, kdy je fotografie rozdělena na několik dílčích částí o nejvýše tako-

vém rozlišení, které podporují všechny webové prohlížeče. Jednotlivé části jsou považovány za samostatné textury a výše uvedené omezení se na ně nevztahuje. Jednotlivé kousky je možné lépe cachovat a implementovat načítání pouze částí potřebných k vykreslení aktuálního pohledu panoramatického prohlížeče. Pro využití této techniky je nutné fotografie předzpracovat a bez cachování výstupu, které je podmínkami služeb pro sdílení fotografií silně omezeno, by docházelo k prodlevám před zobrazením panoramatických fotografií uživatelům. Z toho důvodu nejsou dělená panoramata v této práci použita.

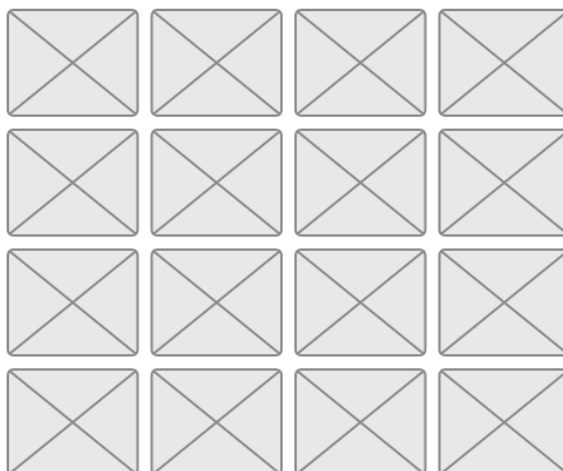
2.9 Wireframe prototyp

Uživatelské rozhraní navržené na základě uvedených technologií počítá se čtyřmi základními stránkami zobrazenými na následujících obrázcích vytvořených v rámci prototypu wireframe. Hlavní stránka webové aplikace zobrazená na obrázku 2.3 je velmi jednoduchá a uživatelům nabízí výběr náhodných panoramatických fotografií. Pomocí navigace lze přejít na mapové zobrazení a stránku určenou k filtrování pomocí obsahu. Mapa zobrazuje panoramata se známou zeměpisnou polohou a umožňuje zobrazit jejich detail. Návrh stránky s mapou panoramat je vyobrazen na obrázku 2.4. Rozložení prvků na stránce určené k filtrování panoramat na základě obsahu znázorňuje obrázek 2.5. Jednotlivé fotografie se zobrazují pomocí stránek a jejich filtrování je možné na základě výběru z dostupných atributů v levém bočním sloupci. Po výběru konkrétní fotografie na libovolné stránce dojde k přechodu na stránku s detailem panoramatu. Ta spolu se sférickou projekcí fotografie zobrazuje veškeré dostupné informace o panoramatu. Prototyp stránky s detailem zobrazuje obrázek 2.6.

PanoBrowser

[Home](#) [Map](#) [Content](#)

Recommended panoramas

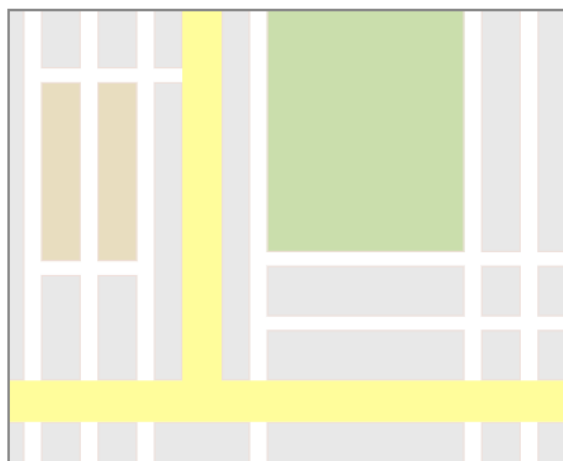


Obrázek 2.3: Hlavní stránka

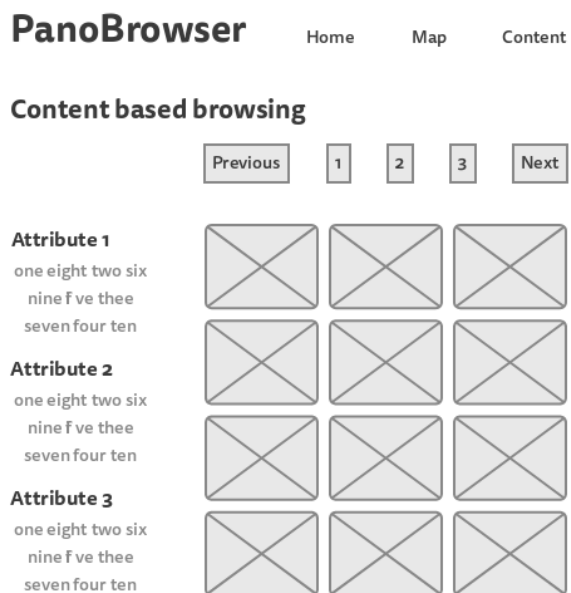
PanoBrowser

[Home](#) [Map](#) [Content](#)

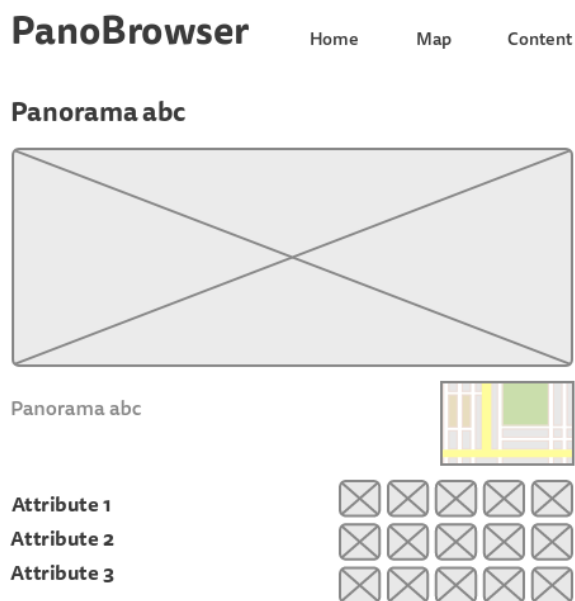
Map



Obrázek 2.4: Zobrazení mapy



Obrázek 2.5: Filtrování na základě obsahu



Obrázek 2.6: Detail panoramatu

Implementace

Systém pro sběr, analýzu a zobrazení panoramat je implementován na základě předchozích kapitol věnovaných analýze a návrhu možných řešení. Během implementace získávání dat ze sítí pro sdílení fotografií se naskytlo několik situací, které vyžadovaly mírné upravení logiky aplikace oproti navrženému řešení. Ke změnám docházelo především při komunikaci s API služby Flickr.

Implementace jednotlivých metod pro získávání a analýzu fotografií je provedena modulárně tak, aby bylo možné metody zpracování vyměnit nebo přidat další a rozšířit tak funkčnost aplikace. Systém nepoužívá cachování panoramatických fotografií a ke své funkci potřebuje pouze minimální diskový prostor. Na serverové straně aplikace je kromě jazyka PHP použita pouze jedna snadno nainstalovatelná knihovna pro podporu neuronových sítí v jazyce C.

3.1 Cílová platforma

Vývoj a testování aplikace probíhá na lokálním počítači s linuxovým operačním systémem, webovým serverem Apache, PHP ve verzi 7.1 a databázovým serverem MariaDB, který je open source odnoží známé MySQL. Pro přípravu vývojového prostředí, které se podobá běžným webhostingovým službám byly použity následující příkazy.

```
apt-get install apache2 libapache2-mod-php7.1 php7.1  
php7.1-mysql php7.1-gd php7.1-intl php7.1-curl  
php7.1-xml php7.1-mbstring mariadb-server
```

Pro klasifikaci panoramatických fotografií do skupin dle vybraných atributů je určena knihovna FANN implementující neuronové síť. Instalaci

3. IMPLEMENTACE

knihovny i rozšíření umožňující volat její funkce z PHP skriptů lze provést pomocí následujících příkazů.

```
apt-get install libfann-dev
pecl install fann
```

Pro definici a stahování závislostí webových aplikací se často používá Composer [15]. Jedná se o nástroj umožňující spravovat a stahovat závislosti potřebné pro chod webových aplikací. Composer také zajišťuje automatické načítání všech požadovaných balíčků do aplikace, odpadá tak nutnost manuálně připojovat velké množství knihoven zvlášť.

Na klientské závislosti v podobě CSS a JavaScriptu je vhodné použít nástroj Bower [16], který funguje na podobném principu. Jím získané knihovny je ale nutné připojit k aplikaci manuálně v hlavičce HTML stránky. Oba správce závislostí použité při implementaci aplikace lze jednoduše nainstalovat pomocí níže uvedených příkazů.

```
curl -s http://getcomposer.org/installer | php
npm install -g bower
```

V tuto chvíli je vývojové prostředí připravené a je možné začít s vytvářením systému pro sběr, indexaci a zobrazování panoramat. Jako první krok je vhodné stáhnout skeleton frameworku Nette [17] do kořenového adresáře dokumentů webového serveru. Tato ukázková struktura včetně frameworku samotného je určena pro okamžitý vývoj aplikace. Staženou strukturu je možné pozměnit a přizpůsobit potřebám vyvíjeného systému a aplikačního prostředí.

```
cd /var/www
composer create-project nette/sandbox panorama-browser
```

Správnost provedených kroků pro přípravu vývojového prostředí lze ověřit zadáním níže uvedené adresy do webového prohlížeče. Zobrazení úvodní stránky indikuje správou instalaci a nastavení požadovaných komponent.

```
http://localhost/panorama-browser/www/
```

Nette framework používá třívrstvou aplikační architekturu Model-View-Controller. Model představuje hlavní logiku aplikace a pracuje s daty. View je vrstva určená k zobrazování informací uživateli a tvoří ji šablony pro generování HTML obsahu. Controller neboli presenter je ústřední část aplikace zpracovávající požadavky uživatelů, která komunikuje s modelem a zajišťuje vykreslení šablon.

3.2 Zdroje dat

Výsledná aplikace používá k získávání sférických panoramat dvě sítě pro sdílení fotografií s přístupem přes API. Jedná se o sítě Flickr a 500px, které jsou pro vytvořenou aplikaci nejvhodnější. Ačkoli jsou obě služby i jejich API odlišné, poskytují všechny potřebné údaje pro vytvoření indexu a následnou analýzu textových údajů.

3.2.1 Flickr

První službou použitou k získávání sférických panoramat je Flickr, síť obsahující největší množství fotografií. Implementace při komunikaci s API používá pro dotazy výchozí formát založený na architektuře REST [18]. Jedná se o bezstavovou komunikaci pomocí protokolu HTTP. Na endpoint API je zaslán požadavek, jehož metoda a parametry jsou určeny pomocí parametrů HTTP GET. Po jeho zpracování je klientovi zaslána odpověď v požadovaném formátu. Pro webovou aplikaci jsou vhodné dva z nabízených typů formátů odpovědí. Výchozí PHP serializovaný formát a JSON. Přestože Flickr dokumentace v ukázkových kódech předpokládá využití PHP serializace, tato práce používá z bezpečnostních důvodů JSON. V případě použití PHP serializovaného formátu by mohla být škodlivá odpověď převedena na objekt, který bude načten a spuštěn. Použití formátu JSON je v tomto ohledu bezpečnější a obsahem je pouze datová struktura bez metod a funkcí.

Data vrácená z API jsou ve skutečnosti ve formátu JSONP, což je JSON obalený funkcí. Ta umožňuje načtení dat v klientském JavaScriptu nehlédě na omezení same-origin policy. Pro zpracování dat v serverové aplikaci je nejprve tato funkce ze získané odpovědi odstraněna a její parametr, pole obsahující požadovaná data, převeden na PHP objekt obsahující informace o všech vrácených fotografiích.

Metoda *flickr.photos.search* umožňuje vyhledávat fotografie podle mnoha zadaných parametrů a dle dokumentace vrací maximálně 4000 výsledků ve formě stránek o maximální velikosti 500 položek. Právě omezení celkového počtu výsledků vede k nutnosti implementace postupného získávání informací o fotografiích. Toho lze docílit uvedením parametrů *min_upload_date* a *max_upload_date* specifikujících velikost pohyblivého časového okna, které se dynamicky přizpůsobí maximálnímu počtu výsledků vrácených z API. Implementace dále obsahuje procházení jednotlivých stránek získaného výsledku. Po zpracování všech stránek algoritmus upraví hodnoty časového okna a pokračuje v získávání další sady fotografií.

3. IMPLEMENTACE

Takto popsaná implementace by měla správně fungovat a získávat všechny nalezené fotografie. Při opakovaných testech bylo zjištěno, že množství získaných fotografií se liší a nedosahuje předem zjištěného celkového počtu výsledků. Počet získaných fotografií je závislý na počtu nalezených výsledků a parametru počet výsledků na stránku *per_page*. Při procházení stránek jedné sady fotografií opakovaně dochází ke změně celkového počtu fotografií a tím i stránek samotných. V některých situacích vrácená stránka neobsahuje žádné výsledky. Během dalších testů bylo zjištěno, že ani parametry pro určení časového okna nejsou striktně dodržovány a API vrací i fotografie za uvedeným intervalem, ačkoli se dokumentace o tomto omezení nezmiňuje. Problém se vyskytuje především u kratších intervalů v řádech hodin a minut.

Uvedené problémy nejsou způsobeny nevhodnou implementací ani dočasnou nedostupností služby. Dle internetových diskuzí a odpovědí od pracovníků služby Flickr [19, 20, 21] se jedná o známý problém, který není vyřešen. Vzhledem k těmto skutečnostem byla implementace upravena za účelem co nejspolehlivějšího získávání dat.

Problém s chybějícími daty se podařilo minimalizovat omezením maximálního počtu výsledků a stránek na minimum. Upravená verze algoritmu prochází fotografie po menších časových intervalech maximálně v řádu měsíců tak, aby odpověď obsahovala nejvíce 100 výsledků a jedinou stránku. V případech kde již nelze časový úsek dále zmenšovat jsou výsledky procházeny po jednotlivých stránkách o velikosti 100 položek jako v původní verzi. Takto upravená implementace vrací v průměru o 10 % více fotografií než původní verze pracující s větším počtem nalezených výsledků.

3.2.2 500px

Druhou sítí využitou k získávání fotografií je služba 500px. Pro komunikaci s jejím API je také využíváno protokolu HTTP a architektury REST. Volání metod i jejich odpovědi je velmi podobné komunikaci se službou Flickr. Podstatným rozdílem je jediný dostupný formát odpovědi JSON, který je na rozdíl od JSONP poskytovaného službou Flickr možné ihned použít ke zpracování dat.

Pro vyhledávání fotografií je použita metoda *photos/search*, který na zadaný výraz vrátí všechny nalezené fotografie pomocí stránek o maximálním počtu 100 položek. Absence limitu maximálního počtu nalezených výsledků umožňuje implementaci bez postupného získávání fotografií pomocí časového okna jako v případě služby Flickr. Přestože je získávání fotografií pouze v uvedeném časovém okně výhodné, nelze jej u API služby 500px

implementovat, neboť metoda *photos/search* nepodporuje uvedení datumu jako omezující podmínky při vyhledávání.

Metoda implementovaná k získávání fotografií projde všechny výsledky na zadaný výraz a nové fotografie přidá do indexu spolu s potřebnými metadaty. Kvůli nemožnosti specifikovat časový interval při vyhledávání fotografií je nutné při každé aktualizaci indexu projít seznam všech fotografií odpovídajících zadanému vyhledávacímu výrazu a ty dosud neexistující do indexu vložit.

3.3 Ověřování panoramat

První analýzou aplikovanou na vstupní fotografie je metoda pro ověřování sférických panoramat. Pro rychlejší vývoj a testování části systému zajišťující ověřování panoramat bylo staženo 2 000 fotografií, z nichž polovina jsou sférická panoramata a druhá polovina běžné fotografie. Tento vzorek byl použit pro analýzu vlastností dat a testování algoritmů pro ověření panoramatických snímků.

Implementovaný klasifikátor má za úkol přijmout do systému pouze sférická panoramata. Jelikož se jejich konkrétní podoba může lišit v závislosti na tématu fotografie, jsou při jejich ověřování kontrolovány společné vlastnosti, kterými jsou navazující boční strany a podobné obrazové body v prvním a posledním řádku obrazu. Nutnou podmínkou k úspěšnému ověření je poměr stran fotografie odpovídající formátu 2:1, který je pro sférická panoramata typický.

Při ověřování panoramat založeném na obrazové analýze je použit výpočet barevné vzdálenosti dvou pixelů, která měří jejich odlišnost. Tato barevná vzdálenost je vypočítána jako euklidovská vzdálenost jednotlivých barevných kanálů RGB podle následujícího vzorečku.

$$\text{barevná vzdálenost} = \sqrt{(r_1 - r_2)^2 + (g_1 - g_2)^2 + (b_1 - b_2)^2}$$

Správná návaznost bočních stran fotografie je klíčová pro jakákoli panoramata. Kontrola návaznosti je v tomto případě implementována algoritmem, který kontroluje barevnou informaci pixelů. Pro každý řádek je vypočítána barevná vzdálenost pixelů ležících na opačných stranách panoramatu podle barevných hodnot kanálů RGB. Získaná vzdálenost je sčítána přes všechny řádky a následně vydělena jejich počtem z důvodu normalizace tak, aby metoda byla nezávislá na rozlišení vstupní fotografie. Nižší

3. IMPLEMENTACE

vypočítané hodnoty značí vyšší pravděpodobnost, že se jedná o sférické panorama.

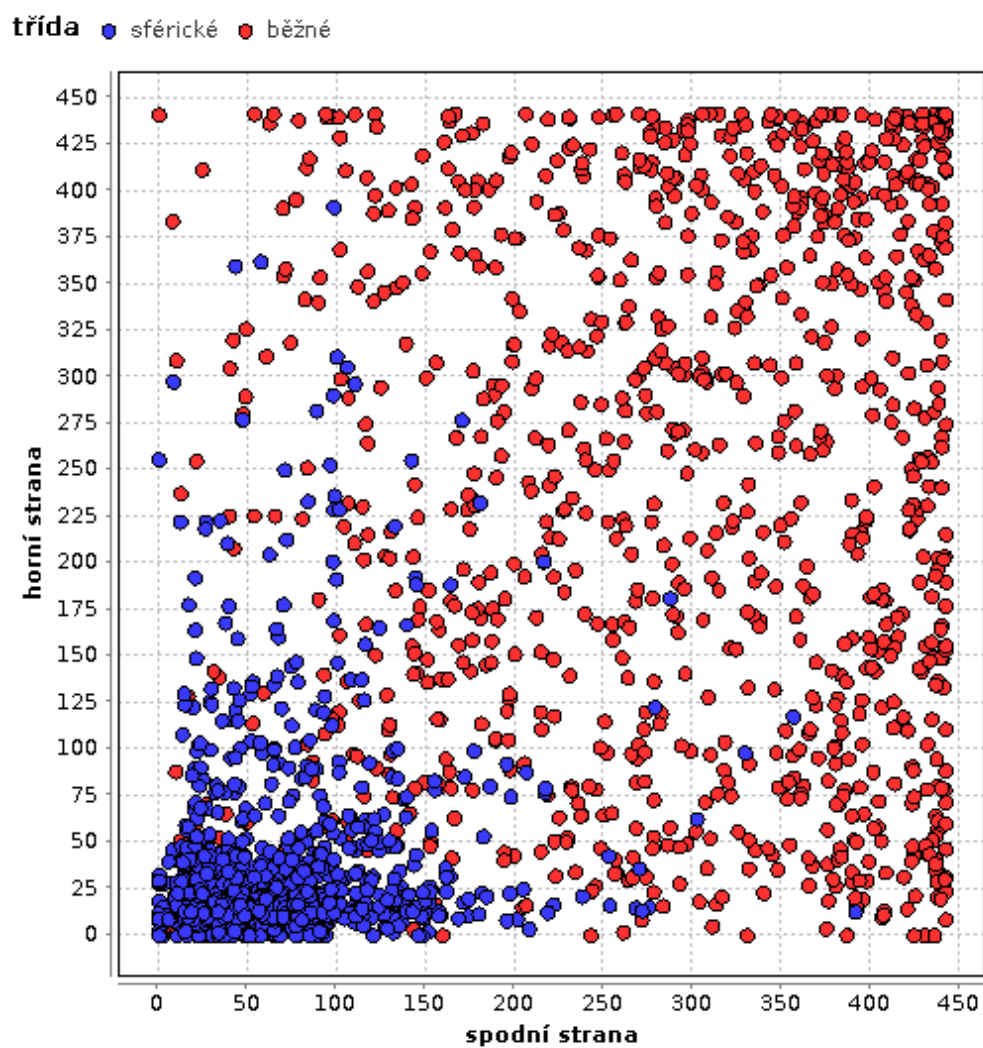
Doplňkovou metodou pomáhající odlišit sférická panoramata od běžných fotografií je podobnost obrazových bodů v rámci celého řádku. Jak je uvedeno v sekci 2.3, první a poslední řádek pixelů obsahuje nejvíce redundantních informací, neboť v projekci zobrazuje velmi malou část obrazu. Nejde o jediný bod, ale o nejmenší možnou kružnici v rámci sférické projekce. Tato metoda vrací největší nalezenou barevnou vzdálenost mezi pixely v rámci požadovaného řádku.

Obě metody používají grafickou knihovnu GD a její funkce pro zjištění barevné hodnoty jednotlivých pixelů. Testovací fotografie byly použity k vytvoření datasetu pro analýzu jejich atributů získaných z ověřovacích metod. Jedná se o barevnou vzdálenost stran fotografie, barevnou vzdálenost v rámci horní řádky, barevnou vzdálenost v rámci spodní řádky a zprůměrovanou barevnou vzdálenost horní a spodní řádky.

Cílem analýzy datasetu je seznámit se s četností a rozložením jednotlivých atributů, vybrání vhodné klasifikační metody a případně i správných hraničních hodnot. K analýze vytvořeného datasetu byl použit program RapidMiner Studio, který umožňuje nejen rychlé statistiky a grafy jednotlivých atributů, ale také testování klasifikačních metod jako jsou k-NN, rozhodovací stromy, neuronové sítě a další.

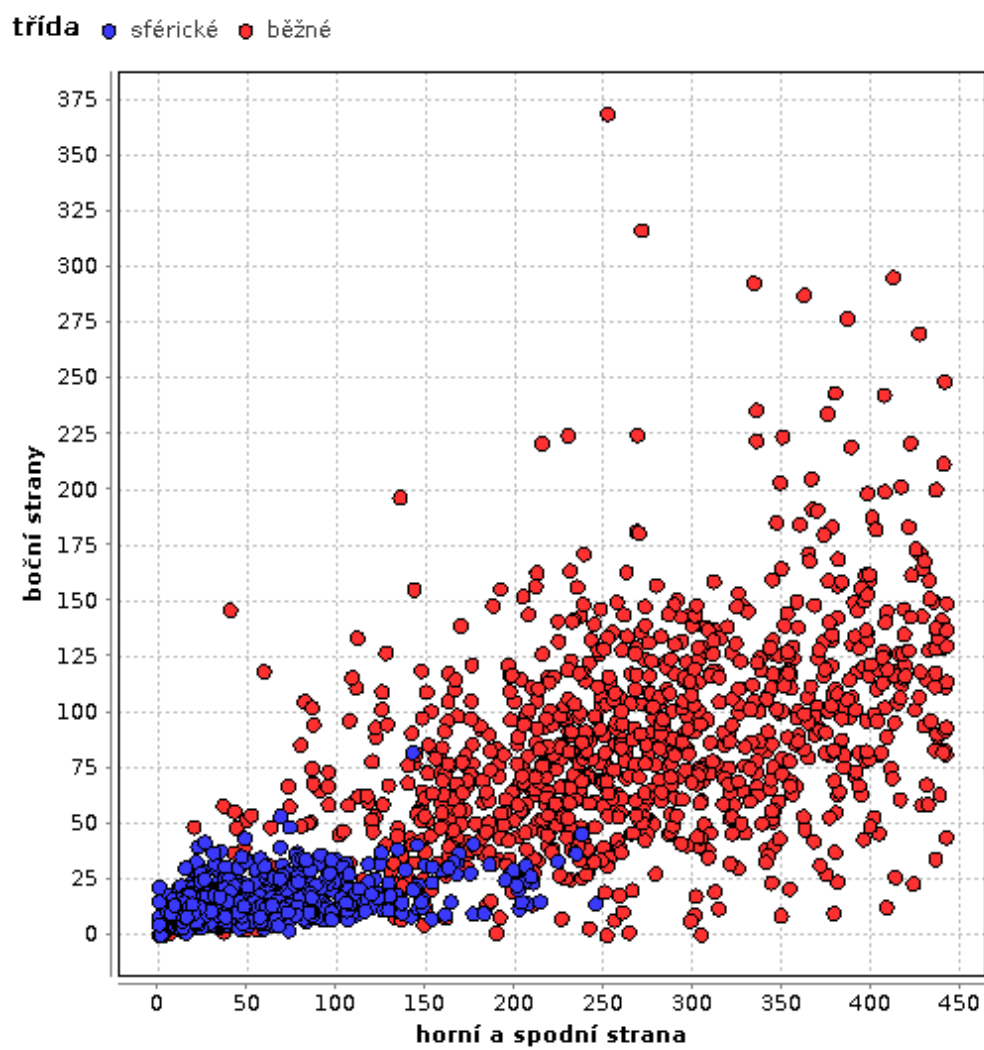
Na obrázku 3.1 je zobrazeno rozložení panoramat a běžných fotografií v závislosti na barevné vzdálenosti pixelů v horní a spodní řádce panoramatu. Fotografie nelze na základě těchto atributů separovat bez chyb, přesto se barevná vzdálenost pixelů v rámci řádky ukázala jako výhodná a analyzované údaje potvrzují domněnku o podobnosti uvedenou v sekci 2.3. Obrázek 3.2 zobrazuje graf kombinující barevnou vzdálenost horní a spodní strany s barevnou vzdáleností bočních stran. Barevná vzdálenost bočních stran pomáhá separovat další skupinu fotografií, které by jinak byly klasifikovány chybně. Na základě výsledků této analýzy byly pro finální ověřovací metodu zvoleny atributy barevná vzdálenost pixelů na stranách fotografie a zprůměrovaná největší barevná vzdálenost v rámci první a poslední řádky panoramatu.

Vzhledem k nízkému počtu atributů a lineární separovatelnosti byl ke klasifikaci použit jednoduchý rozhodovací strom se dvěma podmínkami. Pro nastavení rozhodovacích hranic byla použita matice záměn vypovídající o vlastnostech klasifikace. Definice matice záměn uvedená na obrázku 3.3 specifikuje čtyři skupiny výsledků v závislosti na skutečné a klasifikované hodnotě atributu. Pro výslednou aplikaci je důležité především, aby nebyla běžná fotografie označena za sférické panorama. Takový výsledek je nazýván falešně pozitivní a jeho četnost lze měřit jako *false positive rate*.



Obrázek 3.1: Analýza barevné vzdálenosti pixelů v rámci řádky

3. IMPLEMENTACE



Obrázek 3.2: Analýza barevné vzdálenosti zvolených atributů

		Klasifikace	
		positive	negative
Skutečnost	positive	TP	FN
	negative	FP	TN

Obrázek 3.3: Matice záměn

		Klasifikace	
		sférické	běžné
Skutečnost	sférické	932	68
	běžné	35	965

Obrázek 3.4: Matice záměn pro zvolené hraniční hodnoty

Opačný problém klasifikující sférické panoramata jako běžné fotografie se označuje *false negative rate*.

$$FPR = \frac{FP}{FP + TN}$$

$$FNR = \frac{FN}{TP + FN}$$

Hraniční hodnota pro separaci pomocí barevné vzdálenosti stran byla nastavena na hodnotu 40. Hranice pro rozhodování dle atributu barevná

vzdálenost v rámci horní a spodní řádky byla nastavena na hodnotu 120. Pro uvedené hodnoty dosahuje FPR pouze 3,5 %. Nerozpoznaných sférických panoramat je méně než 7 %, jak je uvedeno v matici záměn na obrázku 3.4.

3.4 Analýza textu

Pomocí metod text miningu jsou zpracována textová data poskytovaná prostřednictvím API službami pro sdílení fotografií. Jedná se nejen o název a popis fotografie, ale také o klíčová slova a zeměpisné souřadnice.

První textová data, která jsou již z API připravena k okamžitému použití jsou štítky. Ty jsou uloženy do zvláštní tabulky a s fotografií spojeny pomocí vazby M:N, čímž je docíleno úplného odstranění redundantních dat. Štítky jsou zachovány v původním formátu, bez mezer a nejsou překládány do společného jazyka.

Další atributy, které je možné ihned použít jsou zeměpisné souřadnice. Ty slouží k zobrazení pozice jednotlivých panoramat na mapě a vyhledání nejbližších panoramat k právě zobrazené fotografii. Ze zeměpisných souřadnic jsou dále pomocí reverzního geocodingu získány další údaje po shlukovou analýzu. Pro převod souřadnic na adresu je použito API služby Google Geocoding. Ta na požadavek s uvedenými GPS souřadnicemi odpoví adresou rozloženou na několik dílčích částí podle geopolitického členění. Do databázového indexu jsou uloženy informace o zemích, státech a městech, ve kterých se fotografie nachází. API je voláno jen v případě, že souřadnice u získaného panoramatu jsou nenulové. Adresní údaje nemusí existovat pro libovolné souřadnice, případně mohou být dostupné pouze některé z uvedených částí.

Další textová data je nutné před jejich analýzou nejprve předzpracovat. Jejich úprava spočívá v překladu do společného jazyka, který je zároveň podporován metodami následné analýzy. Zvoleným jazykem je angličtina, která je přítomna u největší části fotografií a zároveň je podporována nejvíce službami a knihovnamy určených pro text mining. Pro realizaci detekce jazyka a překladu do angličtiny je zvoleno API Google Translation, které aktuálně podporuje 104 jazyků [22], což je nejvíce z dostupných služeb. Pokud je v názvu a popisu fotografie detekován podporovaný jazyk jiný než angličtina, pak je proveden překlad a označení původního jazyka je uloženo do databáze. Jakmile jsou údaje dostupné v cílovém jazyce, je možné spustit analýzu na základě těchto textových údajů.

Pro rozpoznávání umístění z textových metadat fotografie pomocí analýzy pojmenovaných entit se používá API Google Natural Language, které poskytuje přesnější výsledky rozdělené do podrobnějších kategorií. Jednot-

livé lokace jsou rozděleny na běžné a přesné [23]. Každá z těchto typů lokací je uložena v oddělené tabulce podobně jako štítky. K získaným lokacím se lze také chovat podobně jako ke štítkům a v aplikaci slouží k vyhledávání, filtrování i shlukování.

3.5 Analýza obrazu

Analýza fotografií na základě obrazových dat používá vlastní detekční algoritmy založené na analýze histogramů různých podob. Tyto metody mají za cíl získat informace, které je vhodně charakterizují a na základě kterých lze fotografie seskupit nebo vyhledávat. Analýza obrazu je vhodná také pro fotografie bez podrobných textových údajů, které by jinak zůstaly bez atributů vhodných ke shlukování.

Níže uvedené a implementované metody poskytují abstraktní klasifikaci do malého počtu skupin s širokým rozsahem. Jejich úkolem je vystihnout celkové pojetí fotografie bez zaměření na detail. Díky tomu lze při nabídce podobných panoramat vždy zobrazit dostatek snímků s maximální mírou celkové podobnosti fotografie.

3.5.1 Rozpoznávání dne a noci

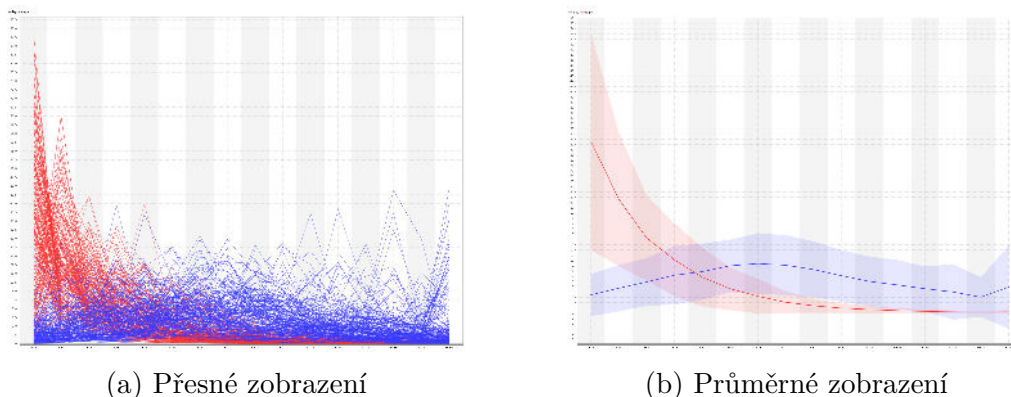
Denní a noční expozice jsou ve většině případů lidským vnímáním velmi snadno odlišitelné. O podobnou klasifikaci se snaží metoda pro analýzu světlosti fotografie. Ta je implementována analýzou histogramu, který je schopen vyjádřit poměr světla a stínů v obraze. Rozpoznávání mírně komplikuje fakt, že fotoaparáty se snaží vždy exponovat fotografii tím způsobem, aby byl poměr světla a stínů vyvážený. Naopak mnohé povedené noční fotografie jsou vlivem manuálního nastavení expozičních parametrů relativně světlé s modrou oblohou a výraznými světly v podobně umělého osvětlení, což se liší od běžné představy tmavé noční fotografie. Rozlišení denní doby nemusí být v tomto případě jednoznačné a při implementaci je potřeba počítat s tím, že některé fotografie mohou zůstat neklasifikované.

Pro analýzu histogramů během vývoje a testování bylo staženo a vybráno 240 fotografií s denním záběrem a 240 fotografií s noční tematikou. Z těchto fotografií bylo vytvořeno několik typů histogramů za účelem jejich prozkoumání. Z jednotlivých histogramů byly vygenerovány datasey pro zpracování v nástroji RapidMiner. Ten byl použit pro analýzu a grafické znázornění histogramů.

Z obrázků černobílých histogramů 3.5 je dobře patrná základní rozlišitelnost fotografií podle četnosti nejtmařejších částí obrazu. Klasifikace pomocí

3. IMPLEMENTACE

pouze této podmínky by nepracovala spolehlivě například na světlejších nočních fotografiích a je proto nutné využít sofistikovanějších klasifikačních metod, které dobře pracují v multidimenzionálním prostoru vstupních atributů. Jednou z takových metod jsou neuronové sítě, které fungují pro lineárně separovatelné i složité problémy, jsou odolné proti odlehlým bodům a velmi přizpůsobivé širokému spektru klasifikačních úloh.



Obrázek 3.5: Histogramy nočních (červené) a denních (modré) fotografií

Pro klasifikaci použita knihovna FANN napsaná v programovacím jazyce C implementuje umělé vícevrstvé neuronové sítě a umožňuje široké nastavení jejich parametrů. Pro tuto knihovnu dále existuje rozšíření, které zajišťuje volání jejích funkcí přímo z jazyka PHP a to i v aktuální verzi 7.1.

Jako vybavovací funkce výstupní vrstvy byla zvolena omezená lineární funkce. Oproti nejpoužívanější sigmoidní má výhodu v lineárním rozdělení hodnot klasifikace a rychlejším zpracování [24]. Lineární rozdělení výstupních hodnot umožňuje lépe odlišit pravděpodobnost příslušnosti do konkrétní třídy. Omezený obor hodnot z intervalu $< 0; 1 >$ značí nutnost upravit hodnoty, které by určovaly příslušnost do třídy s pravděpodobností větší než 1 právě na maximální hodnotou 1. Pro libovolnou klasifikovanou fotografii jsou vráceny dvě hodnoty určující příslušnost do každé z tříd. Panoramata s pravděpodobností příslušnosti větší než 95 % jsou označena klasifikovanou třídou. Ostatní fotografie zůstanou neklasifikované.

Z každé sady připravených fotografií bylo zvoleno 40 jako trénovacích a 200 jako testovacích sloužící k ověření úspěšnosti klasifikace. Trénování probíhalo na neuronové síti s omezenou lineární aktivační funkcí a maximální požadovanou chybou 0,001.

První testy měly za úkol vybrat nejlepší z implementovaných algoritmů pro výpočet histogramu. Jeden histogram pro tři oddělené barevné kanály a dva histogramy pro černobílou verzi fotografie. Převod do odstínů šedi

input : red, green, blue
output: weighted average

```

1 begin
2    $average = 0.21 * R + 0.72 * G + 0.07 * B$ 
3   return  $average$ 
4 end

```

Algoritmus 1: Vážený průměr barevných kanálů RGB

byl proveden jak pomocí aritmetického průměru, tak dle algoritmu 1, který pomocí váženého průměru aproximuje vnímání lidského oka [11]. Jemnost histogramu určuje počet stejně širokých intervalů jasu, do kterých byly hodnoty při výpočtu seskupeny. Počet intervalů histogramu také odpovídá požadovanému počtu vstupních neuronů klasifikační neuronové sítě a jejich nárůstem se zvyšuje doba potřebná pro naučení sítě z trénovacích dat.

Histogram	Jemnost	Správně denních	Správně nočních
ČB aritmetický	64	93,0 %	88,5 %
ČB vážený	64	92,0 %	84,5 %
RGB	3 * 64	94,0 %	87,0 %

Tabulka 3.1: Porovnání typů histogramů

Dle výsledků uvedených v tabulce 3.1 podávají jednotlivé typy histogramů podobné výsledky. Do dalších testů byl vybrán černobílý histogram počítaný aritmetickým průměrem z jednotlivých barevných kanálů, který poskytuje uspokojivé výsledky a oproti histogramu vytvořeného z barevné fotografie je jeho výstup třikrát menší a tedy rychlejší při následném zpracování.

Další test je zaměřený na přesnost rozpoznávání vzhledem k jemnosti histogramu. Barevná hloubka zpracovávaných fotografií je 8 bitů, tedy 256 možných úrovní jasu. Tolik úrovní není pro klasifikaci denní doby potřeba a seskupením sousedních hodnot do intervalů lze zjednodušit následné zpracování klasifikátorem.

Z testů jemnosti histogramů dle tabulky 3.2 nejlépe vplynuly histogramy s rozlišením 16 jasových hodnot. Tyto histogramy jsou stále dostatečně vypovídající a oproti plné bitové hloubce černobílé fotografie poskytují výraznou úsporu prostředků. U histogramů s jemnějšími intervaly bylo dokonce zaznamenáno zhoršení přesnosti klasifikace. Tento stav může být způsobem příliš velkým počtem intervalů, na které se není klasifikátor schopen naučit na zvoleném počtu trénovacích fotografií. V případě potřeby

3. IMPLEMENTACE

Histogram	Jemnost	Správně denních	Správně nočních
ČB aritmetický	2	83,5 %	84,5 %
ČB aritmetický	4	86,5 %	89,0 %
ČB aritmetický	8	93,5 %	91,5 %
ČB aritmetický	16	94,0 %	92,0 %
ČB aritmetický	32	92,5 %	84,5 %
ČB aritmetický	64	93,0 %	88,5 %
ČB aritmetický	128	92,0 %	85,0 %

Tabulka 3.2: Porovnání jemnosti histogramů

použití jemnějších intervalů by bylo vhodné zvýšit velikost trénovací sady, nebo zvolit jinou klasifikační metodu.

Dalším testovaným parametrem je vliv rozlišení panoramatu na úspěšnost klasifikace. Pro analýzu světlosti nejsou detaily fotografie důležité a použitím náhledu s menším rozlišením lze analýzu výrazně urychlit. Zrychlení nastává jak při samotném vytváření histogramu, tak při stahování fotografie pro jeho vytvoření. API použitých služeb pro sdílení fotografií poskytují náhledy s horizontálním rozlišením 240 nebo 256 pixelů.

Histogram	Šířka obrázku	Správně denních	Správně nočních
ČB aritmetický	30	83,0 %	84,5 %
ČB aritmetický	60	93,0 %	86,5 %
ČB aritmetický	120	94,5 %	85,5 %
ČB aritmetický	240	94,0 %	92,0 %
ČB aritmetický	500	93,5 %	91,0 %

Tabulka 3.3: Porovnání rozlišení fotografií

Výsledky uvedené v tabulce 3.3 ukazují na minimální závislost vlivu rozlišení vstupní fotografie na účinnost klasifikace pro šířku fotografie 60 a více pixelů. Z důvodů efektivity jsou pro klasifikaci použity panoramata o malém rozlišení, které API konkrétní služby nabízí. Pro službu Flickr je to rozlišení s šířkou 240 obrazových bodů.

Testování maximální přípustné chyby klasifikátoru umožňuje správné nastavení neuronové sítě, aby nedocházelo k nedoučení nebo přeučení klasifikátoru. Oba problémy mají za následek sníženou schopnost klasifikace fotografií do stanovených tříd na reálných datech.

Z tabulky 3.4 vychází nejlépe hodnota maximální přípustné chyby 0,01. Tato hodnota dosahuje nejlepších výsledků klasifikace panoramat a nehrozí

Histogram	Max. chyba	Správně denních	Správně nočních
ČB aritmetický	0.1	70,0 %	87,5 %
ČB aritmetický	0.01	94,0 %	92,5 %
ČB aritmetický	0.001	93,5 %	92,0 %
ČB aritmetický	0.0001	93,0 %	92,5 %

Tabulka 3.4: Porovnání největší přípustné chyby

u ní přečtení na trénovací data jako v případě použití dalších, nižších hodnot maximální požadované chyby.

Dle výsledků testů byly zvoleny nejlepší parametry pro klasifikaci panoramat. Následně byla neuronová síť naučena na všech 480 testovacích fotografiích a otestována na reálných datech, aby byla ověřena její správná funkčnost.

Vzhledem k tomu, že rozlišení denní a noční fotografie je z histogramu poměrně přímočaré, funguje zvolená neuronová síť velmi dobře i bez skryté vrstvy. Přidáním skryté vrstvy do neuronové sítě již nedojde ke zlepšení přesnosti klasifikace.

Pro aplikaci je důležitá především přesnost (precision), tedy aby se mezi klasifikovanými výsledky nezobrazovaly fotografie, které do konkrétní třídy nepatří. Naopak chybějící neklasifikované fotografie (recall) v malém množství nejsou problém. Z toho důvodu mohou některé fotografie zůstat neklasifikované, není-li vysoká pravděpodobnost, že do dané třídy opravdu patří. Tato podmínka je ošetřena zařazením pouze fotografií s přesností klasifikace větší než 95 %. Klasifikace denních panoramat je spolehlivá a noční fotografie se mezi nimi objevují jen minimálně. Správné zařazení nočních fotografií je obtížnější a mezi výsledky se v menší míře vyskytují i panoramata pořízená během dne, především ve stínu. Přesná klasifikace i určení její úspěšnosti je obtížné, neboť mnoho fotografií se pohybuje na rozhraní dne a noci, jejich správné zařazení může být subjektivní a v některých případech tedy nelze rozhodnout o jejich správné kategorii.

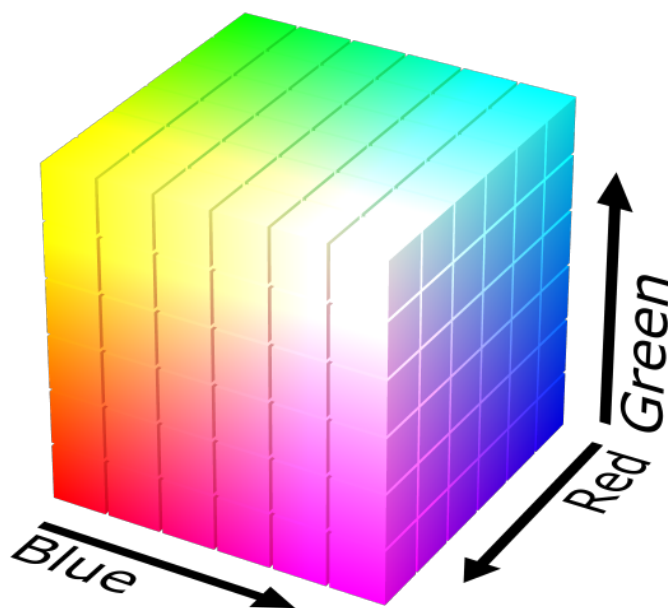
3.5.2 Rozpoznávání barev

Analýza barev zachycených na fotografii a jejich relativní četnost může být použita pro odlišné účely. Hlavním cílem této metody je vložení podílu nalezených barev do indexu pro možnost následného filtrování a shlukování barevně podobných snímků. Další možností využití analýzy barev je naučení klasifikátoru určujícího typ panoramatu v závislosti na podílu jednotlivých barev a jejich umístění ve fotografii.

3. IMPLEMENTACE

Výše uvedená analýza barev může probíhat dvěma odlišnými způsoby. Jednou z variant je nalezení barev, které se na fotografii vyskytují nejvíce a uložení jejich barevného kódu do databáze. Druhá varianta seskupuje libovolné barvy přítomné v obraze do několika předem definovaných barevných kategorií. Výhodou této metody je pevně daný počet možných barev, který je vhodný k možnosti přímého shlukování podobných obrázků, ale také využití v dalších analytických metodách.

Prvním krokem k vytvoření kvalitní metody pro analýzu barev je implementace vhodného algoritmu pro barevnou analýzu fotografie. Využití barevných histogramů pro jednotlivé kanály RGB není pro tento účel vhodné. Výstup takové metody je zbytečně velký a rozlišení barev náročné a nepřesné. Vhodnější variantou je sestavení 3D histogramu, kde je každá osa vyhrazena pro jeden z barevných kanálů. V závislosti na pozici v třídídimenzionálním prostoru lze konkrétní barvy zařadit do předem definované skupiny rozsahu barev.

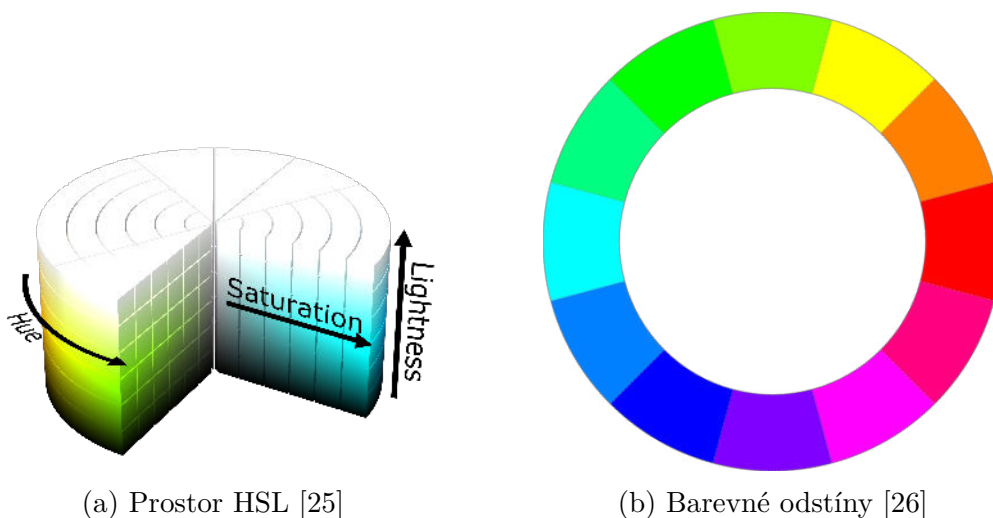


Obrázek 3.6: 3D RGB histogram [1].

Tato metoda se zná být vhodná a jednoduchá na zpracování. Při vytváření každého histogramu je nutné zvolit počet intervalů, do kterých budou hodnoty zařazeny. Při zvolení pouhých tří intervalů na jeden barevný kanál

vznikne kostka složená z 27 částí, což je příliš vysoký počet. Každá barva je v tomto histogramu zastoupena v několika odlišných odstínech, což ztěžuje volbu barvy uživatelem a zároveň rozděluje barevné plochy (obloha, tráva) do několika odlišných skupin. Takové rozdělení zvyšuje nároky na následnou klasifikaci a snižuje její přesnost. Při pokusu o zjednodušení výstupu a vytvoření histogramu za pomoci dvou intervalů, vznikne kostka o 8 částech. Tento počet je pro následné zpracování vhodný. Jeho nevýhodou je, že v takovém případě jsou všechny tmavé barvy zařazeny do jediné části, a to černé.

K vytvoření kvalitní a spolehlivé analýzy barev se zachováním nízkého počtu skupin je tedy vhodné použít jinou metodu. Za tímto účelem je možné převést barevnou informaci do prostoru HSL, kde jsou barvy identifikovány pomocí odstínu, saturace a světlosti. Rozdělením odstínu na N stejných intervalů lze dosáhnout přirozeného seskupení všech barev do N skupin. Zde nutno podotknout, že N je vhodné volit jako $3 * 2^n$, aby počet intervalů korespondoval s počtem namíchaných barevných odstínů. V opačném případě by došlo ke ztrátě některých z barev. Tento přístup je velmi vhodný pro uživatelské filtrování barev i následné zpracování pomocí klasifikátoru.



Obrázek 3.7: Prostor HSL a rozložení barev

Odstín je běžně udáván ve stupních v rozsahu 0° až 360° . Barevný prostor tvoří tři primární barvy. Červená barva má úhel 0° , zelená 120° a modrá 240° . Jejich smícháním vznikají sekundární barvy žlutá, azurová a purpurová, které mají úhel 60° , 180° a 300° . Dalším mícháním lze získat terciální barvy, jak je zobrazeno na obrázku 3.7b. Takto jemné dělení již není pro účely výsledné aplikace potřebné ani vhodné, neboť by opět

docházelo k rozdělení barevných ploch do více skupin a pro uživatele by mohlo být obtížnější přesně zvolit požadovanou barvu.

Požadovaných šesti barevných skupin je dosaženo pomocí intervalů o šířce 60° . Kategorie odstínů je nutné posunout o 30° tak, aby se primární a sekundární barvy nacházely uprostřed požadovaného intervalu. Skupina pro červené barvy tedy sahá od 330° do 30° , pro žluté od 30° do 90° a podobně pro zbytek odstínů. V závislosti na výsledcích testování byl odstín posunut o dalších 10° tak, aby odstín žlutozelené trávy spadl do zelené kategorie a světle modrá obloha byla zařazena do kategorie modré.

Saturace prostoru HSL udává sytost zvoleného odstínu. Pokud je saturace 0 %, jedná se o barvy ze šedé škály. Naopak 100% saturace ukazuje na maximální rozdíl mezi barevnými kanály v reprezentaci RGB prostoru. Světlost dokáže barvu dále měnit. 0% světlost značí černou barvu, základní barva má světlost 50 % a 100% světlost znamená barvu bílou.

Z výše uvedených informací je patrné, že v případě minimální nebo maximální světlosti je barva černá nebo bílá nezávisle na uvedeném odstínu a saturaci.

Převod barev do prostoru HSL je implementován algoritmem 2 dle [27]. Jeho vstupem jsou hodnoty barevných kanálů RGB v rozmezí 0 až 255. Algoritmus vrací pole obsahující odstín, saturaci a světlost požadované barvy. Takto získané hodnoty jsou následně rozděleny do šesti kategorií: červená, žlutá, zelená, azurová, modrá a purpurová. Z implementace vyplývá, že barvám z šedé škály je přiřazen odstín i saturace 0 a bez dalšího zpracování by tyto barvy byly zařazeny do červené skupiny. To je ošetřeno spolu s dalšími hraničními případy, kdy je barva velmi slabě saturovaná. Vzhledem k toleranci nastavení vyvážení bílé barvy jsou za odstín šedé škály považovány všechny pixely se saturací menší než 15 %. Ty jsou následně rozděleny do třech samostatných skupin (černá, šedá a bílá) pomocí světlosti pixelu. Celkem tak systém indexuje podíl devíti barev na každé fotografii.

3.5.3 Rozpoznávání města a přírody

Z barevného histogramu lze dále získat informace o prostředí fotografie. Tento typ analýzy je specifitější než dříve uvedené metody a z důvodu rozmanitosti jednotlivých fotografií nelze očekávat přesnou klasifikaci každé z nich. K analýze prostředí je použit barevný histogram uvedený v kapitole 3.5.2. Ten na základě předložené panoramatické fotografie vypočítá poměr devíti předem určených barevných skupin přítomných v obraze. Podle tohoto poměru lze rozhodnout o pravděpodobném prostředí fotografie na základě podobnosti s fotografiemi použitými pro naučení klasifikátoru.


```
input : red, green, blue
output: array of hue, saturation and luminance

1 begin
2   red = red / 255
3   green = green / 255
4   blue = blue / 255
5   min = MIN(red, green, blue)
6   max = MAX(red, green, blue)
7   lightness = (max + min) / 2
8   if max == min then
9     | hue = 0
10    | saturation = 0
11  else
12    | if max == red then
13      | hue = (green - blue) / (max - min)
14    | else if max == green then
15      | hue = 2 + (blue - red) / (max - min)
16    | else
17      | hue = 4 + (red - green) / (max - min)
18    | endif
19    | hue = hue * 60
20    | if hue < 0 then
21      | hue = hue + 360
22    | endif
23    | if lightness < 0.5 then
24      | saturation = (max - min) / (max + min)
25    | else
26      | saturation = (max - min) / (2 - max - min)
27    | endif
28  endif
29  return [hue, saturation, lightness]
30 end
```

Algoritmus 2: Převod barev z prostoru RGB do HSL [27]

3. IMPLEMENTACE

K vytváření barevného histogramu pro následnou klasifikaci lze postupovat dvěma způsoby. První z nich vytvoří histogram z celé předložené fotografie, jak je obvyklé a jeho výstupem je poměr 9 barev. Druhá metoda využívá specifika sférické panoramatické fotografie k vytvoření dvou oddělených histogramů z horní a dolní poloviny obrazu. Na horní polovině fotografie je v exteriéru zobrazena obloha, spodní polovinu tvoří ve většině případů tráva, silnice a chodníky. Výstupy obou histogramů jsou spojeny do jednoho pole o velikosti 18 prvků. Při vytváření půlených histogramů není analyzována celá fotografie, ale je vynechána malá část okolo horizontu o velikosti 5 % výšky z každé strany a dalších 5 % na spodní části fotografie. Tato úprava vede k minimálnímu ovlivňování dvou odlišných polovin a zanedbání spodní části fotografie k odstranění fotografických stativů a jiných pomůcek viditelných na některých panoramatech.

Analýzou dostupných fotografií bylo rozhodnuto o vytvoření čtyř kategorií pro klasifikaci prostředí fotografie, kterými jsou město, příroda, zimní období a interiérové fotografie. Fotografie z exteriéru mají často modrou oblohu, interiérové nikoli. Panoramata zachycující město obsahují ve spodní části šedou barvu a u fotografií přírody převládá barva zelená. Zimní fotografie mají spodní část převážně bílou.

Pro klasifikaci je využita neuronová síť FANN představená již v kapitole 3.5.1. Pro celkový histogram obsahuje 9 vstupních neuronů, pro půlené histogramy má neuronů 18. V obou případech je zde jeden výstupní neuron pro každou klasifikovatelnou třídu, celkem tedy čtyři. Testovány byly neuronové sítě s žádnou nebo jednou skrytou vrstvou. Aktivační funkce případných neuronů ve skryté vrstvě jsou sigmoidní. Vybavovací funkce výstupní vrstvy je opět omezená funkce s lineárním průběhem. Výchozí maximální chyba je nastavena na 0.01.

Pro účely trénování a testování klasifikátoru bylo staženo 408 unikátních fotografií pro jednotlivé klasifikované třídy. Data byla rozdělena v poměru 1:1 na trénovací a testovací.

Neuronů ve s.v.	Město	Příroda	Zima	Interiér
0	95 %	87 %	89 %	85 %
6	95 %	85 %	94 %	85 %
8	95 %	86 %	92 %	92 %

Tabulka 3.5: Úspěšnost klasifikace v závislosti na skryté vrstvě neuronů

První sada testů měla za úkol určit nutnost použití skryté vrstvy. Dle zdroje [28] je vhodné začít volit počet neuronů ve skryté vrstvě tak, aby jich nebylo více než vstupních neuronů a zároveň méně než neuronů ve vý-

stupní vrstvě. Při zvyšování jejich počtu nemívá smysl použít více neuronů ve skryté vrstvě než dvojnásobek neuronů vstupních. Z výsledků uvedených v tabulce 3.5 není viditelná výrazná změna v úspěšnosti klasifikace při využití skryté vrstvy neuronů. Další testy klasifikační metody používají neuronovou síť bez skryté vrstvy.

Maximální chyba	Město	Příroda	Zima	Interiér
0.1	92 %	86 %	43 %	46 %
0.01	95 %	87 %	89 %	85 %
0.001	95 %	87 %	89 %	85 %
0.0001	93 %	86 %	89 %	85 %

Tabulka 3.6: Úspěšnost klasifikace v závislosti na požadované chybě

Tabulka 3.6 vypovídá o testech provedených k nalezení vhodné maximální přípustné chyby. Výsledky prvního testu s požadovanou chybou 0.1 ukazují na nedoučenou síť, především pro skupiny s menším počtem trénovacích fotografií. Zvolena byla chyba 0.01, která poskytuje dobré výsledky a nevede k možnému přeučení klasifikátoru.

Histogram	Město	Příroda	Zima	Interiér
Celkový	95 %	87 %	89 %	85 %
Půlený	97 %	88 %	82 %	85 %

Tabulka 3.7: Úspěšnost klasifikace v závislosti na typu histogramu

Poslední testy, jejichž výsledky jsou zobrazeny v tabulce 3.7 nepotvrdily předpoklad o výhodnosti použití dvou spojených histogramů z jednotlivých polovin obrazu. Oba typy vytváření histogramů podávají podobné výsledky a pro analýzu byl zvolen běžný celkový histogram podílu barev.

Výsledná neuronová síť bez skryté vrstvy s devíti vstupy a čtyřmi výstupy s maximální požadovanou chybou 0.01 je natrénována na všech ukázkových fotografiích a slouží ke klasifikaci analyzovaných panoramat získaných z API sítí pro sdílení fotografií. Klasifikace na reálných datech je nejspolehlivější pro fotografie zachycující přírodu díky přítomnosti zelené barvy. Klasifikace panoramat města, interiéru a zimního období nemusí v některých případech pracovat správně, neboť všechny uvedené kategorie obsahují fotografie s velkým podílem bílé a šedé barvy.

3.6 Uživatelské rozhraní

Výsledná webová aplikace je rozdělena do minimálního počtu samostatných stránek tak, aby byla zachována jednoduchost systému. Zároveň jsou přidány informativní stránky zobrazující přehled uchovávaných informací o získaných fotografiích a přidružených metadatech včetně statistik jednotlivých atributů. Fotografie nejsou na serveru uchovávány, ale jsou zobrazovány přímo ze serverů služeb pro sdílení fotografií. Při procházení panoramat jsou zobrazeny miniatury s rozlišením o šířce 240 nebo 256 obrazových bodů. Náhledy fotografií jsou zobrazeny v mřížce s nejvíce čtyřmi sloupci v závislosti na velikosti displeje použitého zařízení. Pro rovnoměrné zobrazení mřížky na všech zařízeních jsou fotografie zobrazovány po násobcích dvanácti položek na stránku.

3.6.1 Hlavní stránka

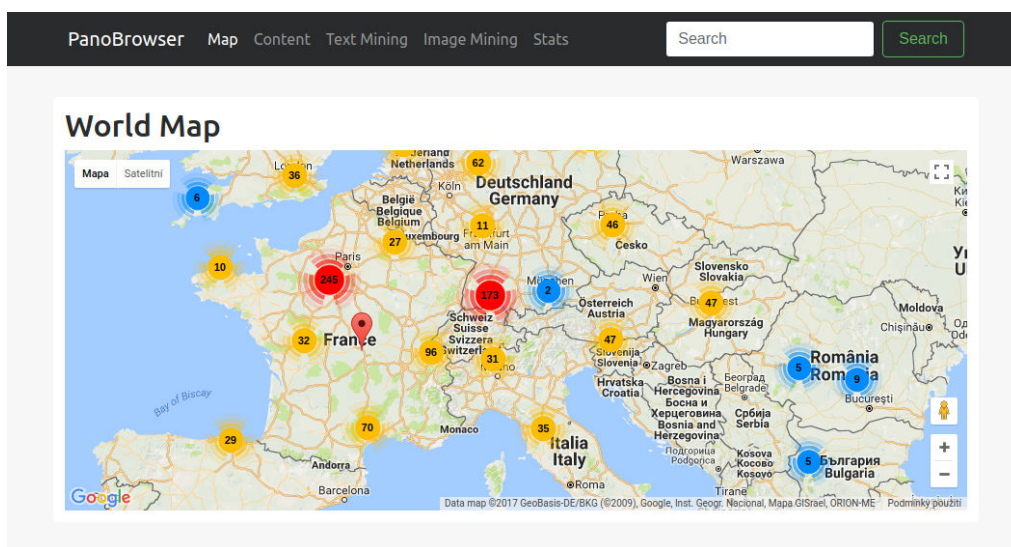
Hlavní stránka zobrazuje náhodný výběr panoramat, která jsou dostupná v rozlišení o šířce 4096 pixelů a zároveň obsahují dostatek metadat pro zobrazení. Právě rozlišení 4096 x 2048 px je nejvyšší rozlišení, které je podporované všemi hlavními webovými prohlížeči. Pomocí metadat jsou vybírány fotografie, které mají vyplněný název, popis a mají určenou zeměpisnou polohu. Tato vstupní stránka slouží k prozkoumávání těch nejzajímavějších fotografií uložených v indexu, které se zároveň zobrazí ve vysoké kvalitě ve všech hlavních webových prohlížečích.

3.6.2 Interaktivní mapa

S podrobným prozkoumáním panoramatického indexu lze začít na stránce s interaktivní mapou zobrazující všechny fotografie se zadanou GPS pozicí. Mapa je vytvořena za použití knihovny Google Maps JavaScript API a rozšíření MarkerClusterer, které vytváří shluky z mnoha značek s podobnou polohou na mapě. Po kliknutí na libovolnou značku umístěnou na mapě je zobrazeno informační okno s názvem a náhledem fotografie.

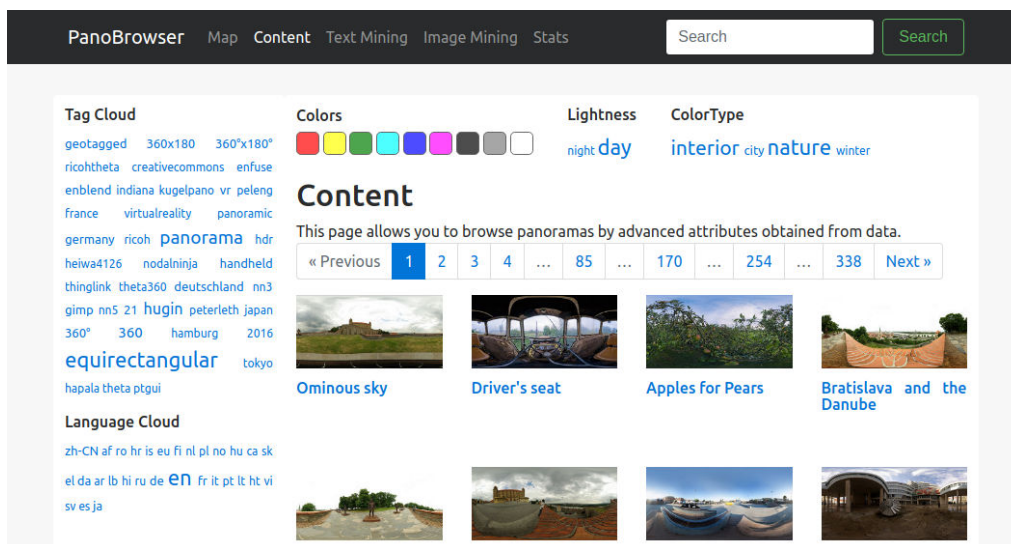
3.6.3 Filtrování na základě obsahu

K prohledávání pomocí dalších parametrů je určena stránka umožňující filtrování pomocí obsahu textu nebo obrazu. Lze zvolit požadovaná klíčová slova, jazyk textových metadat, běžné i přesné lokace nebo zemi, stát a město získané ze zeměpisných souřadnic. Dle obrazových dat lze filtrovat na základě výběru nejčtetnějších barev, denní doby založené na světlosti



Obrázek 3.8: Interaktivní mapa

scény a typu panoramatu. Výsledky odpovídající kombinaci všech zadaných atributů jsou zobrazeny v mřížce za pomoci stránkování po 24 položkách. Během výběru požadovaných metadat jsou kromě fotografií samotných filtrovány také ostatní atributy, aby nemohlo dojít k neplatnému výběru, pro který neexistují žádné výsledky.



Obrázek 3.9: Filtrování na základě obsahu

3.6.4 Textové vyhledávání

V celém indexu je dále umožněno textové vyhledávání pomocí vyhledávacího pole umístěného v hlavičce stránek. Nalezené výsledky jsou zobrazeny na stránce pro filtrování pomocí obsahu a mohou být dále procházeny pomocí výběru z indexovaných metadat. V MySQL databázi lze velmi jednoduché vyhledávání realizovat pomocí operátoru *LIKE %%*. Ten hledá shodu kdekoli v určeném textu. Pro reálné vyhledávání se však nehodí, neboť vyhledává zadaný řetězec a nebere v potaz jednotlivá slova zadaná uživatelem. Za tímto účelem existuje podpora fulltextového vyhledávání se syntaxí *MATCH() AGAINST()*. Jednotlivá slova zadaná uživatelem jsou jednotlivě otestována na shodu. Pro vrácení výsledku stačí shoda jediného ze zadaných slov.

Pro podporu fulltextového vyhledávání je potřeba nejprve vytvořit index typu *FULLTEXT* pro sloupec, nad kterým má být prováděno vyhledávání. Fulltextový index v MySQL automaticky vynechává více než 500 anglických stop slov definovaných v souboru distribuovaném spolu s databázovým serverem. Index pro dva zadané sloupce nad současnou tabulkou *index* lze vytvořit pomocí níže uvedeného SQL příkazu.

```
ALTER TABLE 'index '  
ADD FULLTEXT INDEX 'ft_all' ('title ', 'description ')
```

3.6.5 Přehled atributů

Aplikace dále nabízí dvě stránky s přehledem nalezených textových i obrazových atributů. Na rozdíl od stránky určené k filtrování je zde zobrazeno více možností k výběru. Vlastnosti jsou zde opět zobrazeny pomocí *Word Clouds*. Atributy zastoupené ve větším množství jsou zobrazeny větším písmem než ty, které se vyskytují v méně případech. Atributy jsou spolu s jejich četností získány pomocí SQL dotazu a pomocí normalizační funkce jsou vypočítány velikosti písma, jak je uvedeno v algoritmu 3.

Na řádku 8 algoritmu 3 je pomocí logaritmické normalizace vypočítána velikost písma pro konkrétní vlastnost. Aditivní konstanta s hodnotou 4 určuje rozdíl mezi největší a nejmenší velikostí písma. Při snížení hodnoty bude rozdíl ve velikosti mezi nejčetnějším a nejméně častým slovem větší. Rozdíly ve velikost mezi jednotlivými atributy jsou pouze mírné, aby byly všechny vlastnosti dobře čitelné a zároveň prostorově nenáročné.

Barvy zastoupené v obrazu jsou jako jediné vybírány pomocí barevných tlačítek. Tlačítka barev, které nelze aktuálně zvolit nejsou skryta jako v pří-

input : tags (sorted array of tags and their counts)
output: array of tags and their font sizes

```

1 begin
2   output = []
3   arrayLength = count(tags)
4   max = tags[0]['count']
5   min = tags[arrayLength - 1]['count']
6   for i = 0; i < arrayLength; i++ do
7     count = tags[i]['count']
8     size = log10(4 + ((count - min) / (max - min)) * 10)
9     output[i] = ['name' => tags[i]['name'], 'size' => size]
10  end
11  shuffle(output)
12  return output
13 end

```

Algoritmus 3: Výpočet velikosti písma pro word cloud

padě ostatních atributů, ale z důvodu přehlednosti pouze viditelně odlišena a deaktivována.

3.6.6 Statistika

Stránka se statistikami zobrazuje nejrůznější informace vzhledem k počtu fotografií a jednotlivých atributů. Jsou zde viditelné celkové počty panoramatických fotografií, zadaných souřadnic, štítků, metadat získaných z textu i atributů z obrazových dat.

3.6.7 Detail zobrazeného panoramatu


Stránka zobrazená po výběru konkrétního panoramatu pomocí mapy, filtrování dle obsahu nebo doporučených fotografií z hlavní stránky zobrazuje detailně zvolené panorama a podrobné informace o něm. Stránka dále obsahuje návrhy na další podobné panoramatické fotografie.

Nejdůležitějším elementem je interaktivní panoramatický prohlížeč využívající HTML5 a JavaScript zajištěný knihovnou Pannellum. Prohlížeč automaticky zahájí stahování fotografie ve vysokém rozlišení, během kterého dočasně zobrazí náhled v menší velikosti. Po načtení panoramatu je automaticky spuštěna sférická projekce s rotací do strany. Výběr vysokého rozlišení panoramatické fotografie je závislý na podpoře webového prohlí-

3. IMPLEMENTACE

PanoBrowser Map Content Text Mining Image Mining Stats Search Search

Charles Bridge








Charles Bridge
DRI Processing - 5 shot 1EV brackets, fused and blended with Hugin. Overlaid the enfusion with a Fattal layer from Luminance in Lighten mode at 20% and overlay mode at 50% opacity. Immersion View

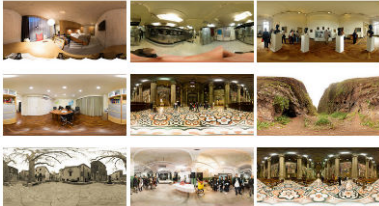
Tags: [zeiss](#), [prague](#), [pano](#), [gimp](#), [praha](#), [charlesbridge](#), [hugin](#), [equirectangular](#), [zf2](#), [nikond3](#), [distagont2821](#), [distagon2128zf2](#)
Proper locations: [Charles Bridge](#)
Lightness: [day](#)
Color analysis: [yellow: 85.9%](#) [red: 4.5%](#) [gray: 3.6%](#) [blue: 2.5%](#) white: 1.8%

Fetches from [Flickr](#), taken on 14. 8. 2011 20:14 by [ruei_ke](#) under [Attribution-NonCommercial-NoDerivs License](#).

Nearest panoramas

- 0.17 km  [Charles Bridge, Prague](#)
- 0.34 km  [Prague](#)
- 0.45 km  [Kafka Square](#)
- 0.51 km  [Astronomical clock, Prague](#)
- 0.55 km  [Old Town Square, Prague](#)

Similar panoramas



Charles Bridge
GPS: 50.086132, 14.413654
[Prague, Czechia](#)

Obrázek 3.10: Detail zobrazeného panoramatu

žeché a technologii WebGL. Za tímto účelem je do HTML kódu stránky vložen element Canvas, který umožní získat maximální rozlišení podporované technologií WebGL, jak je uvedeno v algoritmu 4. Na základě zjištěné hodnoty je načtena fotografie s šířkou 16384, 8192 nebo 4096 pixelů.

```

1 var canvas = $('#measuring')[0];
2 var gl = canvas.getContext('webgl');
3 var maxWidth = gl.getParameter(gl.MAX_TEXTURE_SIZE);

```

Algoritmus 4: Zjištění maximálního podporovaného rozlišení

Kromě panoramatu samotného jsou zobrazeny také veškeré informace získané z API nebo analýzy textu a obrazu. Jedná se především o název, popis a klíčová slova fotografie. Pokud je název a popis v jiném než anglickém jazyce, je uveden také jeho překlad. Z textu jsou pomocí analýzy pojmenovaných entit vypsány běžné a přesné lokace. Na základě rozpoznávání obrazu je zobrazena detekovaná denní doba, prostředí fotografie a poměry barev zastoupených v obraze. Pokud má fotografie uvedeny zeměpisné souřadnice, je zobrazena malá mapa s aktuálním a okolními panoramaty, samotné GPS souřadnice a adresa odvozená prostřednictvím reverzního geocodingu. Na stránce je dále uvedena informace o zdroji fotografie, jejím autorovi, datumu pořízení a použité licenci.

Fotografie zobrazené jako související s aktuálním panoramatem jsou dvou typů. První návrhy tvoří deset fotografií se zadanou zeměpisnou polohou vyfotografované nejbližší k právě zobrazenému panoramatu. U každé navržené fotografie je zobrazena vzdálenost od aktuální pozice v kilometrech a malý náhled společně s názvem v angličtině. Pro nalezení nejbližších fotografií je použit algoritmus dle [29], který počítá vzdálenost mezi dvěma body na kouli s poloměrem 6371 km, což je aproximace planety Země. Vzhledem k efektivnosti algoritmu je výpočet vzdálenosti prováděn v rámci SQL dotazu.

```

SELECT 'id', (6371 * ACOS(
    COS(RADIANS(?lat)) * COS(RADIANS('latitude')) *
    COS(RADIANS('longitude') - RADIANS(?lon)) +
    SIN(RADIANS(?lat)) * SIN(RADIANS('latitude'))
) AS 'distance'
FROM 'index'
ORDER BY 'distance' ASC
LIMIT 10

```

Druhou skupinu doporučených fotografií tvoří výběr dvaceti nejpodobnějších panoramat založený na společných attributech. Podobná panoramata

3. IMPLEMENTACE

jsou vyhledána dle denní doby, typu panoramatu a podobnosti poměru všech barev uložených v indexu. Barevná podobnost je vypočítána na základě euklidovské vzdálenosti pomocí níže uvedeného SQL dotazu, kde za otazníky jsou doplněny aktuální hodnoty porovnávané fotografie.

```
SELECT 'id', (
    POW('c_red' - ?red, 2) +
    POW('c_yellow' - ?yellow, 2) +
    ... +
    POW('c_white' - ?white, 2)
) AS 'distance'
FROM 'index'
WHERE lightness = ?lightness AND colortype = ?colortype
ORDER BY 'distance' ASC
LIMIT 20
```

Právě podobnost založená na poměru jednotlivých barev obsažených ve fotografii dosahuje vizuálně velmi dobrých výsledků a v návržích se objevují pouze fotografie s podobným prostředím, ve kterém byla fotografie pořízena. V případě série podobných fotografií pořízených jedním autorem jsou v návržích často nabízeny právě tyto podobné fotografie.

Závěr

V rámci této práce byly popsány způsoby vytváření a zobrazování sférických panoramat včetně jejich možných projekcí. Byly prozkoumány sítě pro sdílení fotografií a jejich API za účelem získávání fotografií a informací o nich. Dále byly analyzovány a navrženy metody určené pro vytěžení informací z textových i obrazových dat. Při implementaci byl největší důraz kladen na metody s výstupem vhodným k vyhledávání, filtrování a shlukování nalezených panoramat.

Požadavky definované v zadání práce byly splněny a systém pro automatický sběr, indexaci a následné prohlížení sférických panoramat je plně funkční. Přesto zde existuje prostor pro zlepšení a rozšíření webové aplikace. Zlepšovat je možné přesnost ověřování sférických panoramat a jejich klasifikaci do definovaných tříd pomocí dalších metod.

Jednou z možností rozšíření této práce je implementace detekce a zobrazování panoramat se stereografickou projekcí, kterých je v sítích pro sdílení fotografií nezanedbatelné množství. Kromě možného rozšíření zdrojů fotografií by bylo vhodné umožnit manuální přidání panoramatu pomocí URL adresy. Další vlastnosti vypovídající o fotografiích je možné získat prostřednictvím API služeb specializovaných na rozpoznávání obrazu. Mezi takové služby lze zařadit Google Cloud Vision API, Microsoft Computer Vision API nebo Amazon Rekognition. Tyto služby používají hluboké neuronové sítě naučené na velkém množství obrazových dat k identifikaci objektů zachycených na fotografii. Tento způsob by mohl částečně nahradit klíčová slova pro fotografie bez dostatečného množství textových údajů a umožnit přesnější shlukování těchto fotografií.

Literatura

- [1] Horvath, M.: *RGB color solid cube [online]*. [cit. 2017-02-10]. Dostupné z: https://commons.wikimedia.org/wiki/File:RGB_color_solid_cube.png
- [2] Ecma International: *The JSON Data Interchange Format [online]*. 2013, [cit. 2017-02-18]. Dostupné z: <http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf>
- [3] Ćosić, J.: *Web 2.0 services (vulnerability, threats and protection measures)*. 2009, [cit. 2017-01-26]. Dostupné z: <https://www.researchgate.net/publication/290192195>
- [4] Cambridge in Colour: *Panoramic Image Projections [online]*. [cit. 2017-01-23]. Dostupné z: <http://www.cambridgeincolour.com/tutorials/image-projections.htm>
- [5] Pablo d'Angelo, H. B., György Balló: *Hugin - Panorama photo stitcher [online]*. 2016, [cit. 2017-05-01]. Dostupné z: <http://hugin.sourceforge.net/>
- [6] Apple Inc.: *QuickTime VR [online]*. [cit. 2017-01-25]. Dostupné z: https://developer.apple.com/legacy/library/documentation/QuickTime/InsideQT_QTVR/0Preface/QTVR-preface.html
- [7] Zend Technologies Ltd: *PHP7 Infographic [online]*. 2015, [cit. 2017-02-02]. Dostupné z: <https://pages.zend.com/rs/zendtechnologies/images/PHP7-Performance%20Infographic.pdf>
- [8] Creative Commons Česká republika: *Licence CC [online]*. [cit. 2017-02-13]. Dostupné z: <http://www.creativecommons.cz/licence-cc/>

- [9] Yahoo company: *Flickr Services [online]*. [cit. 2017-01-28]. Dostupné z: <https://www.flickr.com/services/api/>
- [10] 500px Inc.: *500px API documentation [online]*. 2017, [cit. 2017-02-05]. Dostupné z: <https://github.com/500px/api-documentation>
- [11] Riemersma, T.: *Colour metric [online]*. [cit. 2017-02-27]. Dostupné z: <https://www.compuphase.com/cmetric.htm>
- [12] Khronos WebGL Working Group: *WebGL Specification [online]*. 2017, [cit. 2017-04-10]. Dostupné z: <https://www.khronos.org/registry/webgl/specs/latest/1.0/>
- [13] Bösch, F.: *WebGL Stats [online]*. 2017, [cit. 2017-04-10]. Dostupné z: https://webglstats.com/webgl/parameter/MAX_TEXTURE_SIZE
- [14] BrowserLeaks: *WebGL Browser Report [online]*. 2017, [cit. 2017-04-11]. Dostupné z: <https://browserleaks.com/webgl>
- [15] Nils Adermann, J. B.: *Introduction to Composer [online]*. [cit. 2017-02-13]. Dostupné z: <https://getcomposer.org/doc/00-intro.md>
- [16] Twitter: *Bower: A package manager for the web [online]*. [cit. 2017-02-13]. Dostupné z: <https://bower.io>
- [17] Nette Foundation: *Nette Documentation [online]*. [cit. 2017-02-13]. Dostupné z: <https://doc.nette.org/en/2.4/>
- [18] Fielding, R. T.: *Architectural Styles and the Design of Network-based Software Architectures [online]*. 2000, [cit. 2017-04-10]. Dostupné z: https://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm
- [19] Berzi, C.: *Search API pagination bug [online]*. 2016, [cit. 2017-03-22]. Dostupné z: <https://www.flickr.com/groups/51035612836@N01/discuss/72157666364892360/>
- [20] peterpolack2: *Photos array empty for certain pages [online]*. 2016, [cit. 2017-03-22]. Dostupné z: <https://www.flickr.com/groups/51035612836@N01/discuss/72157674867579785/>
- [21] Stavric, M.: *Advanced Search- Search by Date Taken - Not working [online]*. 2016, [cit. 2017-03-26]. Dostupné z: <https://www.flickr.com/help/forum/en-us/72157633772894957/>

-
- [22] Google Inc.: *Language Support [online]*. 2017, [cit. 2017-04-15]. Dostupné z: <https://cloud.google.com/translate/docs/languages>
- [23] Google Inc.: *Natural Language API Basics [online]*. 2017, [cit. 2017-04-20]. Dostupné z: <https://cloud.google.com/natural-language/docs/basics>
- [24] H. Yonaba, V. F., F. Anctil: Comparing Sigmoid Transfer Functions for Neural Network Multistep Ahead Streamow Forecasting. *Journal of Hydrologic Engineering [online]*, duben 2010, [cit. 2017-03-10]. Dostupné z: <https://www.researchgate.net/publication/245287610>
- [25] Horvath, M.: *HSL color solid cylinder [online]*. [cit. 2017-02-10]. Dostupné z: https://commons.wikimedia.org/wiki/File:HSL_color_solid_cylinder.png
- [26] Horvath, M.: *RGB Wheel [online]*. [cit. 2017-02-10]. Dostupné z: <https://commons.wikimedia.org/wiki/File:RGBWheel.svg>
- [27] Kriko, L. Z.; Baba, S. E.; Krikor, M. Z.: Palette-based image segmentation using HSL space. *Journal of Digital Information Management*, ročník 5, č. 1, 2007: str. 8. Dostupné z: <http://www.dirf.org/jdim/v5i10a2.pdf>
- [28] Sarle, W. S.: *How many hidden units should I use? [online]*. 2002, [cit. 2017-04-20]. Dostupné z: <ftp://ftp.sas.com/pub/neural/FAQ3.html>
- [29] Veness, C.: *Calculate distance, bearing and more between Latitude/Longitude points [online]*. 2017, [cit. 2017-04-14]. Dostupné z: <http://www.movable-type.co.uk/scripts/latlong.html>

Seznam použitých zkratk

API Application Programming Interface

CORS Cross Origin Resource Sharing

CSRF Cross Site Request Forgery

CSS Cascading Style Sheets

EXIF Exchangeable Image File Format

FPR False Positive Rate

GPS Global Positioning System

HSL Hue Saturation Lightness

HTML Hypertext Markup Language

HTTP Hypertext Transfer Protocol

JSON JavaScript Object Notation

JSONP JavaScript Object Notation with Padding

PHP PHP Hypertext Preprocessor

REST Representational State Transfer

RGB Red Green Blue

UI User Interface

URL Uniform Resource Locator

A. SEZNAM POUŽITÝCH ZKRATEK

WebGL Web Graphics Library

XML Extensible Markup Language

XSS Cross Site Scripting

Obsah přiloženého CD

readme.txt	stručný popis obsahu CD
src	
├─ impl	zdrojové kódy implementace
├─ thesis	zdrojová forma práce ve formátu \LaTeX
text	text práce
├─ thesis.pdf	text práce ve formátu PDF
├─ thesis.ps	text práce ve formátu PS