VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

SÉMANTICKÁ SEGMENTACE V HORSKÉM PROSTŘEDÍ

SEMANTIC SEGMENTATION IN MOUNTAINOUS ENVIRONMENT

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE — **Bc. JAKUB PELIKÁN**
AUTHOR

VEDOUCÍ PRÁCE — **Ing. JAN BREJCHA**
SUPERVISOR

BRNO 2017

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav počítačové grafiky a multimédií Akademický rok 2016/2017

Zadání diplomové práce

Řešitel: **Pelikán Jakub, Bc.**
Obor: Informační systémy
Téma: **Sémantická segmentace v horském prostředí**
Semantic Segmentation in Mountainous Environment
Kategorie: Zpracování obrazu

Pokyny:

1. Proveďte rešerši existujících metod sémantické segmentace. Zaměřte se zejména na existující metody pro sémantickou segmentaci venkovních scén.
2. Vyberte několik metod vhodných pro sémantickou segmentaci scén v horském prostředí. Adaptujte výsledky metod pro použití s datasetem GeoPose3K (dodá vedoucí).
3. S vybranými metodami experimentujte, změřte jejich úspěšnost na datasetu GeoPose3K.
4. Metody, které to umožňují, dotrénujte pomocí dat z datasetu GeoPose3K.
5. Diskutujte dosažené výsledky, možnosti použití a možnosti dalšího vývoje.
6. Vytvořte krátké video nebo plakát demonstrující výsledky práce.

Literatura:

- Fully Convolutional Networks for Semantic Segmentation;Jonathan Long, Evan Shelhamer, Trevor Darrell; The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 3431-3440

Při obhajobě semestrální části projektu je požadováno:

- Body 1 až 3.

Podrobné závazné pokyny pro vypracování diplomové práce naleznete na adrese http://www.fit.vutbr.cz/info/szz/

Technická zpráva diplomové práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap, které byly vyřešeny v rámci dřívějších projektů (30 až 40% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Brejcha Jan, Ing.**, UPGM FIT VUT
Datum zadání: 1. listopadu 2016
Datum odevzdání: 24. května 2017

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačové grafiky a multimédií
612 66 Brno, Božetěchova 2

doc. Dr. Ing. Jan Černocký
vedoucí ústavu

Abstrakt

Sémantická segmentace je jedním z klasických problémů počítačového vidění a silným nástrojem pro strojové zpracování a pochopení scény. V této práci nasazujeme sémantickou segmentaci v čistě horském prostředí. Hlavní motivací naší práce je možnost použití sémantické segmentace pro automatické zjištění geografické pozice, kde byla fotografie pořízena. V této práci jsme zhodnotili aktuální metody sémantické segmentace a vybrali z nich tři, které jsou vhodné pro adaptování do horského prostředí. Vhodně jsme rozdělili datovou sadu obsahující horské fotografie na validační, trénovací a testovací množinu tak, aby je bylo možné použít pro dotrénování vybraných metod sémantické segmentace. Na horských datech jsme dotrénovali modely z vybraných metod. Segmenty z nejlepších dotrénovaných modelů jsme nechali vyhodnotit respondenty pomocí elektronického dotazníku a také jsme je použili v procesu odhadu orientace kamery. Ukázali jsme, že vybrané metody sémantické segmentace lze úspěšně použít v horském prostředí. Naše modely jsou dotrénovány na 11, 5 nebo 4 horských třídách a nejlepší z nich dosahují na 4 třídách hodnocení mean IU 57.4%. Modely jsou použitelné i prakticky, což jsme ukázali jejich nasazením jako součást procesu odhadu orientace kamery.

Abstract

Semantic segmentation is one of classic computer vision problems and strong tool for machine processing and understanding of the scene. In this thesis we use semantic segmentation in mountainous environment. The main motivation of this work is to use semantic segmentation for automatic location of geographic position, where the picture was taken. In this thesis we evaluated actual methods of semantic segmentation and we chose three of them that are appropriate for adapting to mountainous environment. We split the dataset with mountainous environment into validation, train and test sets to use for training of chosen semantic segmentation methods. We trained models from chosen methods on mountainous data. We let segments from the best trained models get evaluated in electronic survey by respondents and we evaluated these segments in process of camera orientation estimation. We showed that chosen methods of semantic segmentation are possible to use in mountainous environment. Our models are trained on 11, 5 or 4 mountainous classes and the best of them achieve on 4 class mean IU 57.4%. Models are usable in practise. We show it by their deployment as a part of camera orientation estimation process.

Klíčová slova

Sémantická segmentace, sémantická segmentace v horském prostředí, Caffe, ALE, FCNs, Deeplab, GeoPose3K

Keywords

Semantic segmentation, semantic segmentation in mountainous environment, Caffe, ALE, FCNs, Deeplab, GeoPose3K

Citace

PELIKÁN, Jakub. *Sémantická segmentace v horském prostředí.* Brno, 2017. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Brejcha Jan.

Sémantická segmentace v horském prostředí

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením pana Ing. Jana Brejchy. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Jakub Pelikán
17. května 2017

Poděkování

Rád bych poděkoval panu Ing. Janu Brejchovi za poskytnuté konzultace a odborné vedení práce.

Obsah

Kapitola 1

Úvod

V dnešní době je snaha nahrazovat lidské činnosti umělou inteligencí, což se promítá v jejím masivním rozvoji. Jedná se o celou řadu činností, některé byly strojem nahrazeny již před delší dobou, například automatická digitalizace poštovních směrovacích čísel na poštovních zásilkách. Jiné činnosti jsou nahrazeny nově, ale již jsou běžně používané, jako je rozpoznávání obličejů a anotace fotografií. Další činnosti jsou aktuálně vyvíjené a jsou tedy spíše testované než běžně používané, například inteligentní virtuální společníci pro konverzaci či autonomní vozidla. Všechny tyto činnosti mají společnou vlastnost, kterou je zpracovávání nějakého vstupu například zvuku nebo obrazu, kdy se snaží rozpoznat, co obsahuje. Tato činnost je pro člověka naprosto přirozená. Člověk pozná na fotce cyklistu na kole nebo štěkání psa v audio nahrávce. Lidé se již od narození přirozeně učí, jak vypadá cyklista nebo štěká pes. Umělá inteligence se snaží o podobný přístup, například naučit se, jak vypadá cyklista. Využívá k tomu data, která mohou být typicky člověkem doplněna o úplnou nebo částečnou informaci o tom, co reprezentují. Snaží se nalézt podobnosti mezi vzorovými daty stejného typu a na základě takto naučených pravidel poznat třeba psa, který mezi zdrojovými daty nebyl. Stejně tak, jako když člověk vidí psa, kterého nikdy předtím neviděl, ví, že je to pes nikoli třeba žirafa. Pokud takto naučíme stroj rozpoznávat přírodní oblasti, jako je les, voda, skály a obloha, tak lze s jednotlivými segmenty dále pracovat. Můžeme je namapovat na 3D model zemského povrchu a podle toho určit orientaci kamery a zeměpisnou pozici, ze které byla fotografie pořízena. Geo-lokalizace, kterou se zabývá projekt LOCATE [22], je také hlavní motivací této práce. Po úspěšné lokalizaci je možné fotografii lépe porozumět, což umožňuje například automaticky popsat obsah obrázku, nebo fotografii automaticky upravit díky znalosti vzdálenosti jednotlivých objektů od kamery. Pro dosažení zvoleného cíle učení je ovšem klíčové zvolit vhodnou metodu a vzorová data. U vzorových dat je také důležité použít vhodnou míru abstrakce. Pokud chceme naučit stroj, aby rozpoznával les od řeky, pak není důležité, zdali se jedná o jehličnatý, nebo listnatý les, a lze tedy typ lesa abstrahovat. Pokud naopak chceme, aby rozpoznával typy lesů, pak je tedy abstrahovat nelze.

Vzhledem k tomu, že sémantická segmentace je klasický problém v počítačovém vidění, existuje celá řada metod, které ovšem nejsou trénovány na sémantickou segmentaci v horském prostředí. Cílem této práce je natrénovat a vyhodnotit modely právě pro sémantickou segmentaci horských scén. Nejprve musíme tedy z řady metod pro sémantickou segmentaci vybrat vhodné metody pro trénování. Následně je potřeba upravit trénovací data pro použití s vybranými metodami a řadou experimentů dotrénovat na trénovacích datech vybrané modely. Nakonec musíme dotrénované modely vyhodnotit a to nejen pomocí naměřených metrik při trénovaní, ale také praktickým použitím rozsegmentovaných fotografií.

Nejprve se práce v kapitole 2 zaměří na obecné principy sémantické segmentace, poté v kapitole 3 na hluboké učení a strojové učení obecně. Následně jsou v kapitole 4 zhodnoceny aktuální metody pro sémantickou segmentaci a vybrány vhodné metody pro experimentování. V kapitole 5 se trénovací data upraví tak, aby je bylo možné použít pro trénování a vyhodnocování vybraných modelů. V předposlední kapitole 6 jsou provedeny a zhodnoceny experimenty s vybranými metodami. V závěrečné kapitole 7 jsou shrnuty dosažené výsledky a navržena další možná rozšíření.

Kapitola 2

Sémantická segmentace

Sémantická segmentace [42] je zvláštní případ segmentace (kap. 2.1), kdy jejím úkolem je rozdělení obrazu do segmentů, které patří do stejné třídy objektů. Tedy části obrazu nejsou shlukovány pouze podle základních charakteristik jednotlivých objektů v konkrétním obraze, ale podle obecných charakteristik tříd, které jsou naučeny ze vzorových dat. Většina algoritmů pracuje s pevnou množinou tříd a některé jenom s binárními třídami rozlišujícími např. pozadí a popředí, nebo cesty a zbytek obrázku. Tento přístup k segmentaci se uplatňuje například ve zdravotnictví pro detekci nádorů, nebo pro detekci dopravního značení.

Řada moderních metod sémantické segmentace, které jsou založené na detekci pixelů plně konvolučními přístupy[1], je inspirována konvolučními neuronovými sítěmi[2] (CNN) (kap. 3.4.5) a používá hluboké neuronové sítě[3] (DNN) [12]. Hlavní myšlenkou těchto metod je učit se přímo mapovat pomocí CNN pixely z obrazu do tříd.

2.1 Segmentace

Kniha Segmentace obrazu [18] popisuje segmentaci jako proces rozdělení obrazu do různých oblastí za pomoci seskupování sousedních pixelů dohromady podle předem definovaných kriterií. Kritéria můžou být stanovena pomocí specifických vlastností, nebo rysů pixelů reprezentujících objekt v obrazu. Jinými slovy je segmentace technika klasifikace pixelů, která umožňuje detekovat podobné oblasti v obrazu.

Výsledkem segmentace má být soubor vzájemně se nepřekrývajících oblastí, které buď přesně odpovídají objektům vstupního obrazu, pak se jedná o kompletní segmentaci, nebo oblasti úplně neodpovídají, pak se jedná o částečnou segmentaci [21].

Metody segmentace obrazu dle publikací [21, 18] lze klasifikovat do tří základních kategorií, a to na metody vycházející z detekce hran - Edge-based (kap. 2.1.1), metody vycházející z detekce pixelů - Pixel-based (kap. 2.1.2) a na metody orientované na regiony v obraze - Region-based (kap. 2.1.3). Některé moderní metody (kap. 2.1.4) nelze zařadit přímo do jedné ze základních kategorií.

[1] z angl. fully convolutional approaches
[2] z angl. convolutional neural network
[3] z angl. deep neural network

2.1.1 Segmentace na základě detekce hran

Segmentace na základě detekce hran [21] je jeden z prvních přístupů k segmentaci, který vychází ze skutečností, že hranice oblastí obrazu jsou tvořeny hranami, které lze nalézt pomocí aplikace hranového operátoru. Tyto hrany označují místa, kde dochází k jisté nespojitosti např. hodnoty jasu, barvy nebo textury. Poté se v dalším kroku spojují hrany do řetězu s cílem vytvořit obraz, ve kterém se vyskytují jen takové řetězce hran, které odpovídají v původním obrazu hranici oblasti.

2.1.2 Segmentace na základě detekce pixelů

Metody segmentace na základě detekce pixelů [18] nejčastěji používají histogram statistik pro určení jednoduchého, nebo mnohonásobného prahu pro klasifikování obrazu pixel po pixelu. Prahy pro klasifikaci pixelů jsou získány z analýzy histogramu obrazu, který obsahuje počty pixelů s určitou hodnotou odstínu barvy.

2.1.3 Segmentace na základě detekce regionů

Základní myšlenka [21] je rozdělit obraz na maximální souvislé homogenní úseky. Kritérium homogenity může využívat jasové vlastnosti, nebo komplexnější způsob popisu, jako je například textura. Tyto metody se výrazně uplatňují v obrazech se šumem, kde je detekce hran obtížná.

Při sémantické segmentaci založené na detekci regionů [12] je nejprve obraz rozdělen do jednotlivých oblastí, které jsou poté klasifikovány do určité třídy. Jednotlivé oblasti mají typicky různá měřítka a zachycují kompletní objekt a jeho části. Tedy nejprve metody extrahují jednotlivé oblasti a poté trénují, a tím optimalizují znalosti pro klasifikaci oblasti. Nakonec provedou finální klasifikaci oblastí do tříd.

2.1.4 Moderní segmentační metody

Některé používané algoritmy nelze přímo začlenit do jedné z kategorií, tyto algoritmy obsahují principy z více kategorií. Lze zde zařadit přístupy inspirované v přírodě, jedná se například o genetické algoritmy, neuronové sítě nebo algoritmy založené na DNA (DNA-based molecular computing) [15].

Kapitola 3

Hluboké učení

Hluboké učení[1] je nová oblast výzkumu strojového učení (kap 3.1), jejíž hlavním cílem je rozvoj umělé inteligence [20]. Na principu hlubokého učení je postavena celá řada moderních frameworků (kap. 3.4.4), mezi které patří například framework Caffe (kap. 3.5). Hluboké učení [33] umožňuje vytvářet výpočetní modely, které jsou složeny z více zpracovávajících vrstev, které slouží pro učení dat s různou úrovní abstrakce. Tato metoda dramaticky zlepšila rozpoznávání obrazu, řeči, ale i detekci objektů v dalších oblastech, např. při vývoji léků nebo v genomice. Hluboké učení nalezne složitou strukturu ve velkých datových sadách, použije algoritmy zpětné propagace (kap. 3.4.2) pro výpočty změn vnitřních parametrů, které se používají při výpočtech v jednotlivých vrstvách. Vnitřní parametry typicky nazývané jako váhy se upravují pomocí výpočtu gradientu (kap. 3.4.1). Klíčovým aspektem hlubokého učení jsou typicky první vrstvy, jejíž parametry nejsou navrhovány lidmi, ale jsou naučeny ze vstupních dat. V principu tedy máme obrázek na vstupu např. ve formě pole pixelů a v první vrstvě naučené vlastnosti[2], které typicky reprezentují přítomnost, nebo absenci hran v určité orientaci a umístění v obraze. Druhá vrstva typicky detekuje motivy podle části uspořádání hran bez ohledu na drobné změny v krajních polohách. Třetí vrstva shromáždí motivy do velkých kombinací, které korespondují s částmi známých objektů, a následné vrstvy detekují objekty jako kombinaci těchto částí. Při hlubokém učení se využívají principy reprezentačního učení (kap. 3.2) a učení s učitelem (kap. 3.3). Metody hlubokého učení využívají neuronové sítě (kap. 3.4), které mají přímo zakomponovány v algoritmech pro učení a v architektuře.

3.1 Strojové učení

Technologie strojového učení [33] se využívají v mnoha oblastech od webového vyhledávání přes filtrování obsahu na sociálních sítích až po doporučení internetových obchodů. Jsou obsaženy v zákaznických produktech, jako jsou například chytré mobilní telefony. Strojové učení se využívá pro identifikaci objektů na obrázku, převod řeči na text, hledání nových položek, ve zprávách a produktech, ve kterých je zainteresován člověk, a ve vybírání relevantních výsledků vyhledávání. Běžné techniky strojového učení jsou limitované schopností zpracovávat data, která neprošla procesem předzpracování, kde byla upravena pro použití s konkrétní metodou strojového učení. Po desetiletí bylo zapotřebí vývoje a značných od-

[1] z angl. Deep learning
[2] z angl. learned features

borných znalostí pro převod nepředzpracovaných dat do vnitřní reprezentace, ve které byl systém strojového učení schopen ve vstupu detekovat nebo klasifikovat určité vzory.

3.2 Reprezentační učení

Reprezentační učení[3] [33] je soubor metod, které umožňují, aby stroj načítal čistá nezpracovaná data, která tedy nebyla upravena pro použití s konkrétní metodou a automaticky vytvořil reprezentaci, která je potřeba pro detekci nebo klasifikaci. Metody hlubokého učení jsou metody, které využívají reprezentativní učení s více úrovněmi reprezentace. Úrovně jsou skládány z vrstev, kde každá vrstva transformuje reprezentaci na jedné úrovni do vyšší reprezentace, kdy na začátku je čistý nezpracovaný vstup a s každou vyšší úrovní je reprezentace na více abstraktní úrovni. Klíčovým aspektem hlubokého učení je, že vlastnosti vrstev nejsou navrženy člověkem, ale jsou naučeny z dat pomocí obecných učebních metod. Tedy metody hlubokého učení mají velmi slibné výsledky pro různé úlohy v porozumění přirozenému jazyku.

3.3 Učení s učitelem

Učení s učitelem[4] [33] je nejvíce používanou formou strojového učení. Například pokud chceme vybudovat systém, který umožňuje určit, zda na obrázku je dům, auto, nebo člověk, tak je nejprve potřeba shromáždit velkou datovou sadu, která obsahuje obrázky domů, aut a lidí, kde každý obrázek obsahuje označení kategorie, do které zapadá. Ovšem vytvoření této datové sady a popisu dat, která se v ní nachází, se často dělá ručně, nebo jen částečně automaticky, což je hlavní nevýhodou učení s učitelem. Během trénování stroj zpracuje obrázky a vytvoří výstup ve formě vektoru hodnocení pro každou kategorii. Chceme, aby všechny výsledné kategorie měly co nejvyšší skóre, ale tento výsledek je bez tréninku nepravděpodobný. Během učení se počítá funkce, která měří chybu mezi hodnocením výstupu a vzorovým hodnocením. Stroj poté modifikuje své vnitřní nastavitelné parametry za účelem snížení této chyby. Tyto nastavitelné parametry se často nazývají váhy a jsou to reálná čísla. Typické systémy hlubokého učení obsahují stovky milionů těchto nastavitelných vah a stovky milionů značených příkladů, pomocí kterých cvičí stroj.

Po trénování se výkon nastaveného systému měří na jiné datové sadě, která se nazývá testovací sada. To slouží k otestování obecných schopností nastaveného systému a určení výkonu na datech, která nebyla nikdy během tréninku zpracovávána.

3.4 Neuronové sítě

Lidský mozek může být popsán jako biologická neuronová síť [16, 41], což je propojená síť neuronů přenášející komplikované vzory složené z elektrických signálů. Dendridy přijmou vstupní signál a na základě těchto signálů vyšlou výstupní signál prostřednictvím axonů. Mozkem respektive obecně nervovým systémem jsou inspirovány umělé neuronové sítě. Klíčovým elementem umělých neuronových sítí je struktura systému pro zpracování informací. Struktura je složena z velkého počtu propojených elementů (neutronů), které společně pracují na řešení specifického problému. Stejně jako lidé se učí z příkladů, tak

[3] z angl. representation learning
[4] z angl. supervised learning

umělé neuronové sítě se během učení adaptivně upravují a nastavují váhy pro specifické použití, jako je rozpoznání vzorů nebo klasifikace dat.

Jedná se nejčastěji o aplikace, které jsou "jednoduché pro člověka", ale "těžké pro stroj" [41]. Běžné výpočetní systémy jsou psány procedurálně, program začíná prvním řádkem kódu, ten vykoná a pokračuje následujícím příkazem, ale pravé neuronové sítě neběží lineárně, ale paralelně ve všech uzlech. Nejjednodušším modelem je Perceptron (kap. 3.4.3).

3.4.1 Nastavení vah a výpočet gradientu

Pro správné nastavení váhového vektoru [33] algoritmus učení vypočítává vektor gradientu, který pro každou váhu indikuje, zda velikost chyby vzroste, nebo klesne v případě, že se hodnota váhy mírně zvýší. Váhový vektor se poté nastaví v opačném směru k vektoru gradientu. Negativní směr vektoru indikuje, kterým směrem se budou kroky přibližovat k minimu a kde bude průměrný výsledek velikosti chyby nejmenší.

3.4.2 Zpětná propagace při trénování na vícevrstvých architekturách

Procedura zpětné propagace [33] pro výpočet gradientu, který zohledňuje váhy více vrstev, je pouze aplikace řetízkového pravidla derivace funkce. Řetízkové pravidlo je jednoznačný vzorec pro složenou derivaci dvou funkcí[5]. Rovnice zpětné propagace muže být aplikována opakovaně pro propagaci gradientu přes všechny vrstvy. Zpětná propagace začíná od výstupu, tedy na nejvyšší vrstvě a pokračuje všemi možnými cestami dolů k vrstvě, kde je zadáván vstup. Jakmile je tento gradient vypočítán, tak je jednoduché vypočítat gradienty tak, aby respektovaly váhy všech vrstev.

3.4.3 Perceptron

Perceptron [41, 16] je nejjednodušší model neuronové sítě, který představuje výpočetní model jednoho neutronu. Perceptron se skládá z více vstupů, jednoho procesoru a jednoho výstupu. Perceptron má dva módy, trénovací mód, ve kterém je trénován pomocí vstupních vzorů, zda vyslat signál, a používaný mód, ve kterém, když je na vstupu detekován naučený vstupní vzor, je vygenerován signál na výstup neutronu. Pokud není vstup rozpoznán v naučených vstupních vzorech, je použito vysílací pravidlo[6] k rozhodnutí, zda vyslat signál. Činnost perceptronu

1. Přijmutí vstupu ($in_1 = 6$, $in_2 = 5$)
2. Vážení vstupu ($w_1 = 0.5$, $w2 = -1$)
3. Součet vážených vstupů ($(0.5 * 6) + (-1 * 5)$)
4. Vygenerování výstupu (Používá se aktivační funkce - například pokud je suma kladné číslo, pak výsledek je 1, jinak výsledek je -1)

3.4.4 Frameworky pro hluboké učení

Frameworků pro hluboké učení je poměrně mnoho, firma NVidia mezi hlavní řadí frameworky [3]:

[5] Řetízkové pravidlo [6]
[6] z angl. firing rule

- Caffe
  - Vyvíjen: Berkeley Vision and Learning Center (BVLC)
  - Podporované interface: C, C++, Python, MATLAB, příkazová řádka
- Computational Network Toolkit (CNTK)
  - Vyvíjen: Microsoft
  - Podporované interface: C++, příkazová řádka
- Tensor Flow
  - Vyvíjen: Google's Machine Intelligence research organization
  - Podporované interface: C++, Python
- Theano
  - Vyvíjen: Theano Development Team
  - Podporované interface: Python
- Torch, MXnet, Chainer

Mezi další dostupné frameworky patří například Deeplearning4j (DL4J) s knihovnami psanými pro Javu a Scalu [2].

3.4.5 Konvoluční neuronové sítě

Konvoluční neuronové sítě[7] (CNN) [33, 1] jsou navrženy ke zpracování dat zadávaných ve formě tenzorů, například obrázek se skládá ze tří 2D polí reprezentující barevné složky pro každý pixel. Klíčovými myšlenkami CNN jsou místní propojení, sdílené váhy, sdružování a použití mnoha vrstev. Architektura konvolučních sítí je strukturována jako série stupňů, kde první stupně jsou složeny z konvolučních a sdružovacích[8] vrstev. Vstupem jednotek na úrovni *k* je výstup z podmnožiny jednotek na úrovni *k-1*. Jednotky v konvolučních vrstvách jsou organizovány v tzv. *feature maps*, v nichž všechny jednotky jsou spojené a propojené s předchozí vrstvou prostřednictvím řady vah nazývaných filtrační blok. Každý filtr se replikuje v celém vizuálním poli, tyto replikované jednotky sdílí stejné parametry. Gradient sdílených vah se získá jako suma gradientů ze sdílených parametrů. Použití této architektury má dva důvody, v poli dat, jako například v obrázcích, jsou často vzájemně propojené skupiny hodnot a lokální statistiky obrázků a ostatní signály jsou nezávislé na umístění.

3.5 Caffe framework

Framework Caffe [28, 24] je určený pro hluboké učení s důrazem na expresivnost, rychlost a modularitu. Je vyvíjen Barkeley Vision and Learning Center (BVLC) a komunitou přispěvatelů, která obsahuje přes tisíc vývojářů. Umožňuje trénovat na GPU nebo CPU a přepínat mezi nimi. Lze používat například na výpočetních klastrech nebo mobilních zařízeních. Základní struktura Caffe modelu, která vychází z hlubokých sítí, je popsána v kapitole 3.5.1 a následně v kapitole 3.5.2 je popsán samotný běh modelu, při kterém dochází k jeho optimalizaci.

[7] z angl. convolutional neural networks
[8] z angl. pooling layer

3.5.1 Základní struktura Caffe modelu

Základem Caffe modelů [23] jsou hluboké sítě[9], což jsou složené modely, které jsou reprezentovány jako kolekce vnitřně propojených vrstev pracujících s bloky dat. Vrstvy jsou základem pro modely a výpočty. Vyvíjí se z filtrů, používají vnitřní funkce, aplikují různé transformace, normalizují, načítají data a vypočítávají ztrátovou funkci. Caffe pomocí vlastního schématického modelu definuje síť jako sérii za sebou jdoucích vrstev, které jsou propojeny v orientovaném acyklickém výpočetním grafu. Typická síť začíná vrstvou, která načítá data z disku, a končí vrstvou, která počítá cíl úlohy, což může být například klasifikace, nebo rekonstrukce. Řešení je nakonfigurováno samostatně od modelu kvůli oddělení modelování a optimalizace modelu. Síť je popsána jejím modelem, který je definován pomocí *plaintext protocol buffer schema (prototxt)*, zatímco naučené modely jsou serializované pomocí *binary protocol buffer (binaryproto)* a ukládány jako *.caffemodel* soubory. Formát modelu je definovaný pomocí *protobuf* schématu v souboru *caffe.proto*.

Učení v Caffé [26], stejně jako u většiny nástrojů strojového učení, je řízeno pomocí ztrátové funkce. Ztrátová funkce specifikuje cíl učení mapováním parametrů nastavení na skalární hodnoty, které stanovují, do jaké míry je nastavení těchto parametrů "špatné". Cílem učení je tedy najít takové nastavení vah, které minimalizuje ztrátovou funkci.

3.5.2 Řešitel

Řešitel[10] [25, 27] optimalizuje model tak, že nejprve provede dopředný průchod, čímž získá výstup a ztráty, a poté provede zpětný průchod, pomocí kterého vygeneruje gradient modelu, na základě kterého upraví váhy, čímž se pokusí minimalizovat ztráty. Řešitel dohlíží na optimalizaci a generovaní aktualizací parametrů a síť získává ztráty a gradient.
Běh řešitele:

1. vytvoření tréninkové sítě pro učení a testovací sítě pro vyhodnocovaní
2. iterativní optimalizace pomocí dopředného a zpětného průchodu a úprava parametrů
3. pravidelné vyhodnocování testovací sítě
4. vytváření snímků modelu a stavu řešitele v průběhu optimalizace

Kde v každé iteraci:

1. síť provede dopředný průchod pro výpočet výstupů a ztrátové funkce
2. síť provede zpětný průchod pro výpočet gradientu
3. zahrnutí gradientu do aktualizovaných parametrů v závislosti na zvolené metodě řešitele
4. upravení stavu řešitele v závislosti na hodnocení učení, historii a zvolené metodě

[9] z angl. deep networks
[10] z angl. Solver

Kapitola 4

Metody pro sémantickou segmentaci

Aktuální metody zabývající se rozpoznáváním objektů v obrázcích se zaměřují na klasifikaci (na obrázku je nějaký objekt X), detekci (kde přesně se objekt X na obrázku nachází) a segmentaci (které pixely náleží objektu X). V této práci se věnujeme problému sémantické segmentace v horském prostředí, kdy na rozdíl od jiných prací [8] nechceme segmentovat obrázky pouze na oblohu a pozadí, a tím získat horizont, ale chceme přímo nalézt jednotlivé horské oblasti jako je ledovec, voda, les a další. V rámci projektu LOCATE [22] je pro segmentaci v horském prostředí k dispozici upravená verze frameworku ALE, který vychází z metody *Robust Higher Order Potentials for Enforcing Label Consistency* [30], ovšem tato upravená verze je optimalizována pouze pro segmentaci oblohy a pozadí. Framework ALE patří sice mezi úspěšné, ale již starší metody, proto chceme pro segmentaci v horském prostředí vyzkoušet moderní segmentační metody založené na hlubokém učení. Mezi novější úspěšné metody pro sémantickou segmentaci patří metoda DeepLab [14]. Tato metoda je aktuálně dostupná ve dvou verzích, ke kterým je k dipozici řada modelů obsahující různé vylepšení, ale všechny se zaměřují především na segmentaci běžných objektů jak ve vnitřních, tak venkovních scénách. Modely optimalizované pro segmentaci objektů ovšem nejsou pro horské oblasti úplně ideální a lepší by bylo použití modelů, které jsou před-trénovány na horských oblastech, takovéto modely jsou k dipozici k metodě FCNs - plně konvoluční sítě pro sémantickou segmentaci [40]. Tato metoda efektivně pracuje se vstupy libovolné velikosti a má velké množství variant, které jsou vhodné od segmentace venkovních scén včetně horských přes vnitřní scény až po obrázky z hloubkové kamery Microsoft Kinect. Podle porovnání VOC2012[1], které slouží k objektivnímu srovnávání metod pomocí datové sady PASCAL VOC (kap. 4.4.1), dosahuje metoda FCN podobného hodnocení (tabulka 4.1) jako řada dalších metod, které se snaží různými způsoby vylepšit či zefektivnit sémantickou segmentaci. Tyto metody ovšem neobsahují modely před-trénované na horských datech, ale zaměřují se na segmentaci objektů, vnitřních scén, nebo venkovních scén, ale spíše městských, tedy segmentaci budov, aut, silnic, dopravního značení a dalších. Mezi tyto metody patří například metoda *Feedforward semantic segmentation with zoom-out features* [36], která mapuje malé elementy obrázku do zanořující se hierarchie regionů, tedy specifičtější elementy mapuje do obecnějších. Metoda SegNet [10], která podobně jako metoda DeepLab vychází ze sítě VGG16, ale inovativně mění způsob, jakým dekodér nad-vzorkuje svůj vstup s nižším rozlišením, a tím se snaží zefektivnit práci s pamětí a celkovou dobu výpočtu. Tuto

[1] Visual Object Classes Challenge 2012 - hodnocení dostupné online na [9]

metodu se dále snaží vylepšit metoda Bayesian SegNet [29], která je schopna predikovat přiřazení pixelu do třídy s jistou mírou nejistoty, čehož dosáhla pomocí vzorkování metodou Monte Carlo, a tím dosáhla vylepšení o 2-3%. Toto vylepšení se projevilo i při použití toho přístupu na metodách FCN a Dilation Network. Ovšem toto vylepšení bylo u FCN použito pouze na základní model, a tedy nejsou k dispozici varianty modelů segmentující horské oblasti.

Metoda	hodnocení VOC2012 (AP)
DeepLab v1	73.9%
DeepLab v2	79.7%
SegNet	59.1%
Bayesian SegNet	60.5%
Bayesian DN	73.1 %
Bayesian FCN-8	65.4%
FCN-8	62.2%
Feedforward sem. segm. with zoom-out features	64.4%

Tabulka 4.1: Výsledky VOC2012 hodnocení metod pro sémantickou segmentaci podle metriky Averege Precision (AP).

Při výběru metod pro experimentování jsme se snažili zvolit nejen metody s nejlepším hodnocením, ale také zohlednit vhodnost použití metod na horských datech. V experimentech se tedy zaměříme na metodu DeepLab(kap. 4.1), která je vzhledem k tomu, že k ní není dostupný model před-trénovaný na horských datech, z vybraných metod nejméně vhodná, ale v hodnocení segmentačních metod dosahuje jednoho z nejlepších výsledků, a tedy, pokud se jí povede adaptovat na horská data, tak by i na nich mohla dosahovat dobrých výsledků. Dále jsme zvolili metodu FCN(kap. 4.2), ke které jsou k dispozici modely před-trénované na datových sadách obsahující i horské oblasti, a tedy by mělo být možné tyto modely dotrénovat i na čistě horských datech. Metody DeepLab i FCN jsou postaveny nad frameworkem Caffe(kap. 3.5). Na poslední vybrané metodě je založen framework ALE(kap. 4.3), který již v čistě horském prostředí byl úspěšně použit a my se jej pokusíme natrénovat tak, aby kromě oblohy a pozadí segmentoval i zbytek horských oblastí.

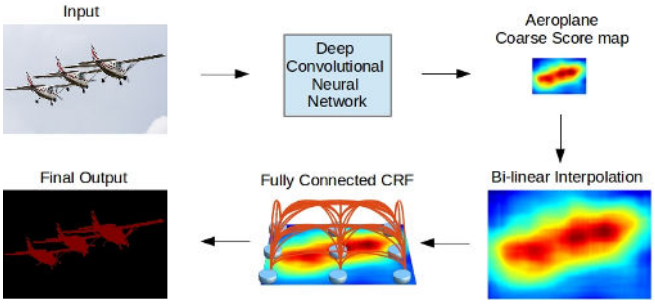
4.1 DeepLab

Metoda sémantické DeepLab [14] je postavena nad frameworkem Caffe s využitím hlubokých konvolučních sítí [2], Atrous konvolucí a plně propojených podmíněných náhodných polích[3] (CRF) [31].

Základní princip metody [34], který je vyobrazen na obr. 4.1, rozšiřuje sémantickou segmentaci pomocí hlubokého učení o tři vylepšení a bylo experimentálně dokázáno, že mají na výsledek podstatný pozitivní vliv. Prvním vylepšením bylo zvýraznění konvoluce s nadvzorkovacím[4] filtrem, nebo atrous konvolucí, čímž vznikl mocný nástroj pro predikci každého pixelu. Atrous konvoluce umožňují explicitně kontrolovat rozlišení, ve kterém jsou

[2] z angl. Deep Convolution Nets
[3] z angl. fully connected Conditional Random Field
[4] z angl. upsampling

odpovědi vypočítávány pomocí hlubokých konvolučních sítí. Také umožňují efektivně zvětšit zorné pole filtrů, což slouží k začlenění většího kontextu bez zvýšení počtu parametrů nebo množství výpočtů. Druhým vylepšením je návrh *atrous spatial pyramid pooling* (ASPP) pro segmenty objektů, které jsou v obrázku v různém měřítku. ASPP sondy jsou na vstupních vrstvách konvoluční funkce spolu s filtry na různých vzorkovacích frekvencích, tedy ASPP sondy zachycují objekty a kontext v různých měřítkách. Třetím vylepšením je zlepšení lokalizace hranic pomocí kombinace DCNN vrstev s plně propojenými CRF, kdy se tato kombinace projevuje jak kvalitativně, tak kvantitativně na zlepšení výkonu lokalizace. CRF je pravděpodobnostní grafický model používaný pro segmentaci do více tříd, který v plně propojené variantě zjišťuje podobnost všech dvojic pixelů v obraze [31].



Obrázek 4.1: Základní princip DeepLab. Převzato z [34].

4.1.1 Modely a sítě

Metoda DeepLab [14, 13] je založena na dvou hlubokých konvolučních neuronových sítích, které jsou získány upravením klasifikačních sítí VGG a ResNet. Nejprve byla založena pouze na síti VGG-16 a až poté vznikla residualní verze sítě DeepLab pomocí upravení moderní ResNet obrazové klasifikace DCNN. Tato upravená verze dosahuje lepších výsledků než původní metoda založená na VGG-16.

Nejlepší výsledek mean IU 79.7% na testovací množině datové sady PASCAL VOC 2012 byl dosažen s modelem DeepLab založeným na síti ResNet-101. Model byl dále porovnáván na datových sadách PASCAL-Context, PASCAL-PersonPart a Cityscapes s dalšími segmentačními metodami. V tabulce 4.2 je porovnání nejlepších modelů DeepLab s modelem FCN-8 na datové sadě PASCAL-Context, která kromě dalších tříd obsahuje i některé horské třídy. Výsledky ukazují, že model DeepLab založený na síti ResNet-101 dosahuje na této datové sadě lepších výsledků než ostatní modely, a tedy i dotrénování na čistě horských datech by mohlo být úspěšné.

Metoda	založena na	mean IU
DeepLab	VGG-16	39.6
DeepLab	ResNet-101	45.7
FCN-8s	-	37.8

Tabulka 4.2: Porovnáná DeepLab a FCNs na datové sadě PASCAL-Context. Převzato z [14].

4.2 Fully Convolutional Networks for Semantic Segmentation (FCNs)

Metoda FCN [40] trénuje konvoluční neuronové sítě (kap. 3.4.5) s end-to-end a pixels-to-pixels přístupem, tedy metoda pro každý pixel na vstupu produkuje pixel na výstupu, čímž se jí povedlo překonat aktuální úroveň vývoje sémantické segmentace bez dalších mechanizmů. Klíčovou myšlenkou je použití "plně konvoluční" sítě, která přijímá vstup o libovolné velikosti a produkuje stejně velký výstup s efektivním odvozováním a učením. Plně konvoluční sítě pro sémantickou segmentaci[5] je první práce na trénování plně konvolučních sítí end-to-end, která k predikci zařazení jednotlivých pixelů do tříd používá před-trénovaný model za pomocí trénování s učitelem. FCNs verze existujících sítí (VGG net, GoogLeNet, AlexNet - kap. 4.2.1) předpovídají třídu každého pixelu výstupu z libovolně velkých vstupů. Učení a závěr je prováděn v celém obrázku zároveň. V nad-vzorkovávajících vrstvách je umožněna predikce jednotlivých pixelů a učení v sítích se provádí pomocí podvzorkování. Tato metoda je účinná jak asymptoticky, tak absolutně a odstraňuje komplikace z jiných prací. Nevyužívá trénování založené na výřezech, které je běžné v různých metodách, ale které postrádá účinnost tohoto plně konvolučního trénování. Metoda neobsahuje problémy s před-zpracováním a následným zpracováním zahrnujícím například superpixely, nebo problémy se zjemňováním pomocí CRF.

4.2.1 Modely a sítě

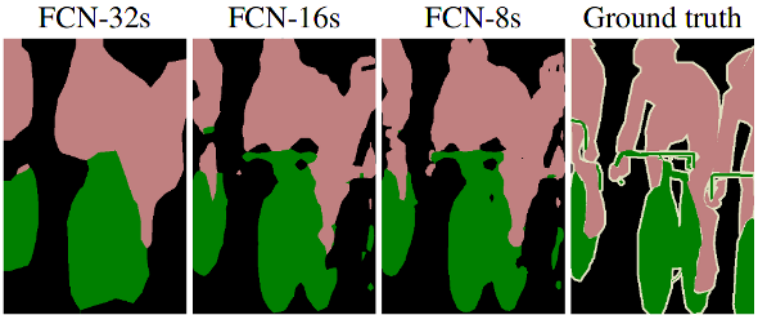
V metodě byly upraveny do plně konvoluční sítě současné klasifikační sítě AlexNet, který vyhrál ILSVRC12[6], VGG net a GoogLeNet, které dopadly velice dobře na ILSVRC14[7], a převedla jejich naučenou reprezentaci na otázku segmentace. Prvotní výsledky mean IU pro testovací datovou sadu PASCAL VOC 2011 pro sítě FCN-AlexNet, FCN-VGG16, FCN-GoogLeNet byly 39.8%, 56.0% a 42.5%. Nejlépe hodnocená síť FCN-VGG16 vycházející z VGG16 se stala pro metodu základní sítí.

Metoda definovala novou plně konvoluční síť pro sémantickou segmentaci, která kombinuje vrstvy hierarchie vlastností a zjemňuje prostorovou přesnost výstupu. Metoda obsahuje tři varianty (8, 16, 32), které se liší velikostí kroku ve finální vrstvě predikce, kde velikost kroku 8, 16 nebo 32 pixelů limituje rozsah detailů v nad-vzorkovaném výstupu. Změna rozsahu detailů je zobrazena na obrázku 4.2 a porovnání výsledků na validační podmnožině PASCAL VOC 2011 je v tabulce 4.3.

	pixel acc.	mean acc.	mean IU	f.w IU
FCN-32s	90.5	76.5	63.6	83.5
FCN-16s	91.0	78.1	65.0	84.3
FCN-8s at-once	91.1	78.5	65.4	84.4
FCN-8s staged	91.2	77.6	65.5	84.5

Tabulka 4.3: Porovnání kroků FCN na segval podmnožině PASCAL VOC 2011 [40].

[5] z angl. Fully Convolutional Networks for Semantic Segmentation
[6] Large Scale Visual Recognition Challenge 2012
[7] Large Scale Visual Recognition Challenge 2014



Obrázek 4.2: Porovnání výsledků kroků 8, 16 a 32 pixelů a originálního ground truth. Převzato z [40].

4.3 ALE - The Automatic Labelling Environment

The Automatic Labelling Environment(ALE) [32] je software pro sémantickou segmentaci, který vznikl na univerzitě Oxford Brookes a je volně dostupný pro akademické účely. Tento software je založen na jádru práce [30] zabývající se porozuměním a rozpoznáváním scén. Tato práce začíná s metodou založenou na jejich P^N modelu, který pro každý pixel určí, který objekt ve scéně reprezentuje. Kód umožňuje vytvořit vlastní klasifikátor pomocí vlastních trénovacích dat. Práce byla oceněna jako nejlepší práce na ECCV[8] 2010.

Metoda *Robust Higher Order Potentials for Enforcing Label Consistency* [30], podle které je vytvořen framework ALE, je založena na podmíněných náhodných polích (CRF) vyššího řadu a využívá potenciál definovaný na množině pixelů (obrazových segmentů) generovaných pomocí segmentačních algoritmů učení bez učitele. Tento potenciál využívá konzistenci popisu regionů v obrazu. Potenciálová funkce vyššího řádu, která je použita ve frameworku, má formu Robust[9] P^n modelu. Metoda využívá výpočtu optimálního prohození a expanze pohybů pro energickou funkci[10] složenou z těchto potenciálů, které jsou počítány pomocí řešení problému st-mincut[11]. Metoda byla testována na problému segmentace objektů do více tříd pomocí rozšíření konvenčního CRF použitého pro segmentaci objektů s potenciálem vyššího řádů definovaným v obrazových regionech. Experimenty na datových sadách ukázaly, že integrace potenciálu vyššího řádů kvantitativně a kvalitativně vylepšuje výsledky vedoucí k definici hranic objektu.

4.4 Datové sady před-trénovaných modelů

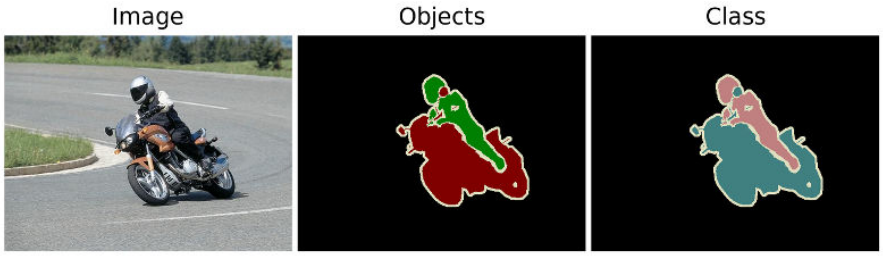
Jednotlivé datové sady jsou určeny k trénování modelů s různými cíli. Základní referenční sada pro rozpoznávání a porovnávání metod je PASCAL VOC (kap. 4.4.1), pro tuto sadu existuje rozšiřující anotace pro detekci hran (kap. 4.4.2) a anotace rozšiřující počet tříd PASCAL-Context (kap. 4.4.3) a jí podobná datová sada SIFT Flow (kap. 4.4.5), která obsahuje anotace jak pro sémantickou, tak pro geometrickou segmentaci. Další datovou sadou je NYU Depth Dataset 4.4.4, který slouží pro segmentaci vnitřních scén, a datová sada COCO určená pro segmentaci jednotlivých instancí objektů (kap. 4.4.6).

[8] European Conference on Computer Vision
[9] Model, který bude fungovat i v případě nejasných cílů, pravidel a poškozených dat
[10] Funkce, jejíž hodnotu se snažíme minimalizovat při trénování modelu
[11] Definice je dostupná na [17]

4.4.1 PASCAL Visual Object Classes (VOC)

PASCAL Visual Object Classes (VOC) [19] je standardní datová sada pro rozpoznávání a benchmark pro detekci a sémantickou segmentaci. Hlavním cílem je rozpoznání objektů z velkého počtu vizuálních objektů v realistických scénách. Při sémantické segmentaci rozpoznává 20 objektových tříd a pozadí (obrázek 4.3). Třídy jsou rozděleny do čtyř hlavních kategorií:

- Člověk
- Zvíře - pták, kočka, kráva, pes, kůň, ovce
- Dopravní prostředek - letadlo, kolo, loď, autobus, auto, motorka, vlak
- Interiér - láhev, židle, jídelní stůl, pokojové rostliny, pohovka, televize nebo monitor



Obrázek 4.3: Ukázka přiřazení každého pixelu reprezentující objekt do třídy objektu nebo přiřazení do třídy pozadí pro ostatní pixely. Převzato z [19].

4.4.2 Semantic Boundary Dataset (SBD)

Semantic Boundary Dataset (SBD) [5] je další rozšířená anotace pro data z PASCAL VOC. Na rozdíl od sémantické segmentace, která si klade za cíl predikovat pixely, které leží uvnitř objektu, SBD se zaměřuje na predikci pixelů ležících na hranici objektu (obrázek 4.6). Tento úkol je pravděpodobně těžší, neboť metriky pro vyhodnocování jsou přísnější. SBD momentálně obsahuje anotace k 11355 obrázkům z datové sady PASCAL VOC 2011.

Obrázek 4.4: Ukázka rozšířené anotace pro predikci pixelů ležících na hranici objektu. Převzato z [5].

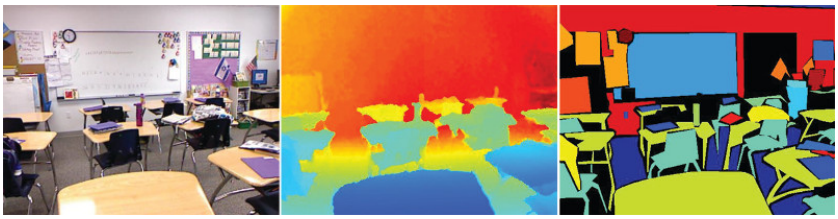
4.4.3 PASCAL-Context

Pascal-Context dataset [37] je sada dodatečné anotace pro PASCAL VOC 2010, tato sada obsahuje více jak 400 různých tříd, mezi které patří jak třídy objektů (například letadlo, kůň, sedačka, hrníček,...), tak třídy povrchů (hora, obloha, les, voda, led, kámen, tráva,...). Příklad ukázky zdrojového obrázku a jeho anotace obsahující jak objekty (člověk, motorka,...), tak povrch (silnice, tráva,...) jsou vyobrazeny na obrázku 4.5.

Obrázek 4.5: Ukázka zdrojového obrázku a dodatečné anotace. Převzato z [37].

4.4.4 NYU Depth Dataset V2 (NYUDv2)

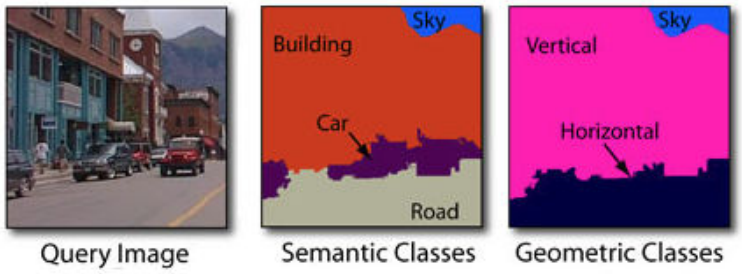
NYU Depth Dataset V2 [38] slouží k segmentaci vnitřních scén. NYU-Depth V2 datová sada je složena z videosekvencí z různých vnitřních scén, které jsou nahrány pomocí RGB a Depth kamery z Microsoft Kinectu (obrázek 4.6).



Obrázek 4.6: Výstup z RGB kamery(vlevo), předzpracování (uprostřed), množina popisků objektů (vpravo). Převzato z [38].

4.4.5 SIFT Flow

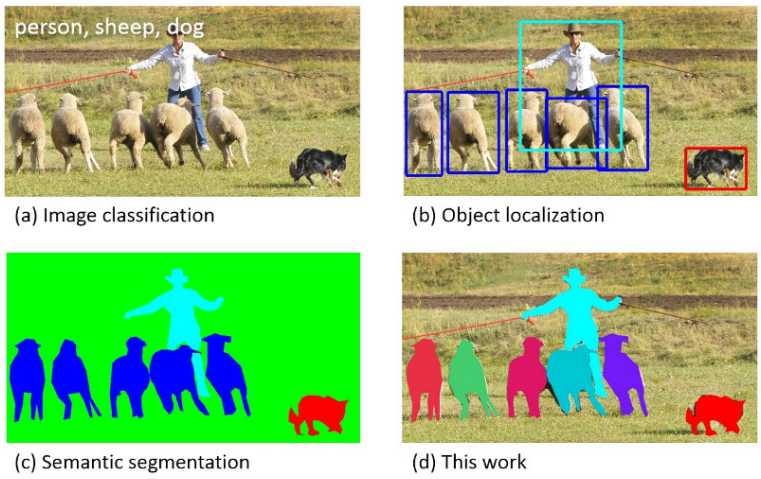
SIFT Flow [43] je datová sada pro sémantickou segmentaci, která obsahuje dva různé typy tříd, geometrické a sémantické (obrázek 4.7). Sémantické třídy jsou podobné třídám z datové sady PASCAL-Context, ale je jich pouze 33. Geometrické třídy jsou triviálnější ve srovnání se sémantickými třídami [44]. Dá se použít rozumný předpoklad, že každá sémantická třída je spojena s unikátní geometrickou třídou a lze je tedy ručně namapovat. Například třída dům by se namapovala na vertikální třídu a třída řeka na horizontální třídu.



Obrázek 4.7: Ukázka originálního obrázku a jeho příslušné geometrické a sémentické anotace. Převzato z [43].

4.4.6 Common Objects in Contex (COCO)

Datová sada COCO [35] obsahuje kolem 328 000 obrázků, na kterých jsou každodenní scény obsahující běžné předměty v jejich přirozeném kontextu. Na obrázcích je anotováno kolem 2.5 milionu objektů, které jsou jedním z 91 rozpoznávaných typů objektů. Datová sada se zaměřuje na segmentaci individuálních objektů, čímž se liší od většiny datových sad (obrázek 4.8), které obsahují anotace buď komplexní pro celý obrázek, nebo pro lokalizaci objektů či pro sémantickou segmentaci.



Obrázek 4.8: Ukázka z datových sad zaměřených na (a) klasifikaci obrázků, (b) ohraničující rám pro lokalizaci, (c) sémantickou segmentaci na úrovni pixelů, (d) segmentaci jednotlivých instancí objektů. Převzato z [35].

Kapitola 5

Příprava dat

Před samotným trénováním modelů je potřeba předzpracovat data z datové sady GeoPose3K [11] (kap. 5.1) a pro každý trénovaný model vytvořit datovou sadu v odpovídajícím formátu. Z ground truth je potřeba odstranit barevné přechody mezi oblastmi a pročistit data (kap. 5.2). Poté jsou ze statistik získaných z upravených ground truth vypočítány globální procentuální statistiky obsažení jednotlivých tříd ve vstupních datech. Následně jsou z globálních statistik a GPS souřadnic vypočítány vhodné množiny obrázků, které reprezentují trénovací, testovací a validační množinu (kap. 5.3). Nakonec jsou transformovány obrázky do požadovaných formátů a je vytvořena datová sada pro trénování konkrétního modelu (kap. 5.4).

5.1 Datová sada GeoPose3K

V rámci výzkumu projektu LOCATE je k dispozici téměř 3200 fotografií, u kterých je známa informace o přesné pozici, kde byly vytvořeny. Informace je reprezentovaná pomocí GPS souřadnic v souřadném systému WGS84 a orientace kamery. Na základě těchto informací byla fotografie přesně lokalizována a umístěna do panoramatu. Díky OpenStreetMap, který obsahuje informace o povrchu, bylo možné ke každé fotce vytvořit ground truth segmentů (kap. 5.1.2) se základními třídami 5.1.1.

5.1.1 Typy tříd ve vstupních datech

Ve vstupních datech rozlišujeme 12 specifických tříd definovaných v OpenStreetMap[1] (lesní porost, ledovec, voda, útes, holá skála, kamenná suť, prales, horské oblasti s nízkou vegetací, horské oblasti pokryté trávou a mechy, pastviny, oblázky, přírodní prohlubně v terénu), tři obecné typy třídy reprezentující oblohu, neznámý typ povrchu a oblast obsahující vše, co je blíže než 30 metrů od kamery, která vyfotila snímek. Mezi objekty, které jsou bližší než 30 metrů, se typicky nachází lidé, auta, cedule a různá stavení. Tyto objekty jsou pro lokalizaci nevhodné, a proto, aby se jejich výskyt omezil, se vše do 30 metrů ignoruje. Procentuální obsažení tříd po odstranění barevných přechodů mezi oblastmi (kap. 5.2) je vyobrazeno v tabulce 5.1.

[1] Dostupné online [4]

Třída	průměrný výskyt
neznámý typ povrchu	37.595%
obloha	34.073%
< 30m	2.854%
holá skála	2.478%
lesní porost, prales	16.616%
voda	3.449%
ledovec	0.977%
kamenná suť	1.152%
pastviny	0.803%
útes	0.0003%
oblázky	0.00002%
přírodní prohlubně v terénu	0.001 %
horské oblasti s nízkou vegetací	0.0%
horské oblasti pokryté trávou a mechy	0.0%

Tabulka 5.1: Procentuální obsažení jednotlivých tříd v datové sadě GeoPose3K

5.1.2 Ground truth

Pojem "Ground truth" původně pochází z vědního oboru zabývajícího se povrchem země a označuje fakta, která jsou potvrzena pomocí skutečné kontroly informací o půdě probíhající přímo v terénu, přičemž tyto informace jsou získány například z leteckých snímků [7]. V kontextu počítačového vidění ground truth nepopisuje pouze fakta o půdě, ale obecně fakta o objektech na fotografii, ke které přísluší. Ground truth k určitému obrázku přidává popis, ve kterém s jistou mírou abstrakce přiřazuje každému pixelu z obrázku třídu objektu, jehož je součástí. Tedy například pro každý pixel v obrázku, na kterém je obloha, existuje v ground truth informace, že tento pixel obsahuje oblohu. Tato informace může být reprezentována například formou druhého obrázku, který obsahuje jen předem specifikované barvy reprezentující jednotlivé třídy, kdy pixel na souřadnici x,y značí, který typ objektu je na pixelu x,y na popisovaném obrázku. Jednou z dalších forem ground truth můžou být například binární soubory *MAT* používané programem Matlab. V popisovaném obrázku nás typicky zajímá jen jistý typ objektů a jen s jistou mírou abstrakce. U geo-lokalize je důležitá například obloha, les, skály atd., ale už není důležité, zdali jsou na obloze mraky či letadlo, ty chceme abstrahovat. Ground truth v počítačovém vidění je často vytvářeno ručně nebo poloautomaticky a nemusí vždy přesně reflektovat skutečnost. Tedy ani ground truth v datové sadě GeoPose3K, které je generováno z OpenStreetMap nemusí být přesné. Nepřesné a případně i neaktuální můžou být samotné mapy, nebo i přes manuální korekci v nich nemusí být fotografie správně zarovnané. Ale i samotné fotografie mohou a často i mají horské oblasti částečně překryté jak řadou nehorských objektů (silnice, auta, budovy, lidi,...), tak i přírodními vlivy (mlha, opar, duha,...).

5.2 Odstranění barevných přechodů a čištění dat

Každá třída je v ground truth reprezentována oblastmi se specifickou barvou, ovšem problém nastává u přechodu mezi oblastmi, které se mírně překrývají, a vzniká tam barevný přechod. Tento barevný přechod obsahuje barvy, které nepatří mezi základních 15 používa-

ných odstínů, a tedy není možno k nim přímo a jednoznačně přiřadit třídu. Nejprve tedy musíme tyto pozvolné přechody eliminovat a nahradit je ostrým přechodem, který obsahuje pouze používané barvy. Souběžně s odstraňováním pozvolných přechodů provádíme i čištění dat. Při zpracování jednotlivých ground truth nejprve provedeme fázi čištění a to tak, že provedeme analýzu barev obsažených v ground truth, jejíž výsledkem je podmnožina seznamu základních barev obsahující pouze barvy z daného ground truth. Pokud barvy v podmnožině seznamu barev reprezentují pouze třídy obloha a neznámý povrch, nebo pouze jednu třídu, pak originální obrázek i ground truth odebereme ze seznamu zpracovávaných obrázků a pokračujeme se zpracováním dalšího ground truth. Jinak pokračujeme procesem odstranění pozvolných přechodů, který pracuje pouze s podmnožinou seznamu barev, tudíž při dalším zpracování nepracujeme se všemi barvami reprezentujícími oblasti, ale pouze s těmi, které jsou reálně v obrázku obsaženy. Při práci s podmnožinou barev tedy nemůže z přechodu vzniknout oblast reprezentující třídu, která v daném ground truth není původně vůbec obsažena.

V prvním kroku odstraňování barevných přechodů ke každé barvě v obrázku, která není v podmnožině obsažena, vyhledáme, zdali barevné složky některé barvy z této podmnožiny nejsou podobné barevným složkám neznámé barvy. Pokud ano, tak neznámou barvu nahradíme barvou s podobnými barevnými složkami z podmnožiny. Za podobné barevné složky barvy považujeme ty, které mají hodnotu odlišnou o $\pm 4\%$. Konstantu 4% jsme nalezli experimentálně tak, že jsme u vybraných obrázků vizuálně hodnotili nově vzniklé přechody. Při větší hodnotě již v přechodech vznikalo velké množství pixelů, jejichž barva reprezentovala jiné oblasti, než mezi kterými jsme odstraňovali přechody.

Poté jsme v druhém kroku odstranili zbylé neznámé barvy, jejichž složky nejsou podobné složkám v podmnožině barev. U každé neznámé barvy jsme vyhledávali ve všech směrech nejbližší barvu z podmnožiny známých barev a tou jsme neznámou barvu nahradili. Při experimentech jsme zkoušeli první krok hledání podobných barev vynechat, ovšem docházelo k tomu, že oblasti, které jsou relativně velké, ale pouze jejich střed má barvu z podmnožiny barev a široké okolí má pouze podobné barvy jako střed, byly výrazně zmenšeny.

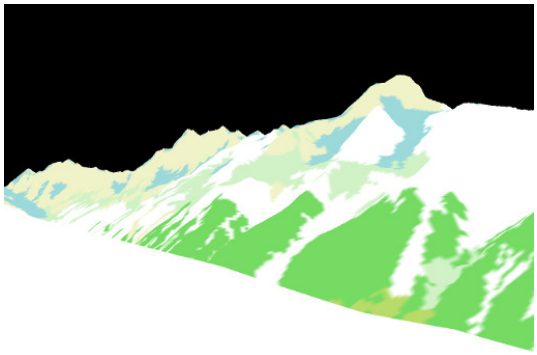
Následně v závislosti na čištění dat pro konkrétní experiment, který pracuje pouze s určitými třídami, jsme některé typy oblastí nahradili a označili jako oblast reprezentující třídu neznámý povrch. Ukázka originálního ground truth a jeho nové verze s odstraněnými barevnými přechody a nahrazením typů oblastí je vyobrazena na obrázcích 5.1 a 5.2.

Posledním krokem je výpočet statistik, které obsahují velikosti jednotlivých typů oblastí v daném ground truth v pixelech.

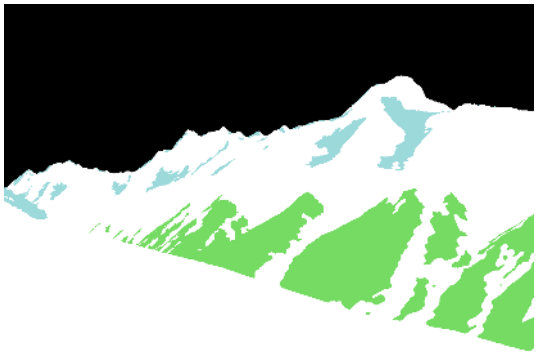
5.3 Rozdělení GeoPose3K na validační, testovací a trénovací množinu

Datovou sadu GeoPose3K je potřeba rozdělit na tři množiny, trénovací pro učení, validační pro vyhodnocení trénování, které probíhá jednou za x-tisíc kroků, kde hodnota x záleží na konkrétním modelu, a testovací pro vyhodnocování modelů. Datovou sadu dělíme tak, aby trénovací množina obsahovala 4/6 obrázků, testovací množina 1/6 obrázků a validační množina 1/6 obrázků.

Při vytváření množin je potřeba zohlednit globální procentuální statistiky velikosti všech tříd tak, aby množiny měly mezi sebou stejný poměr obsažených oblastí reprezentujících jednotlivé třídy. Dále se musí zohlednit i GPS souřadnice jednotlivých obrázků tak, aby všechny obrázky v rámci validační/testovací množiny byly geograficky blízko sebe a nevy-



Obrázek 5.1: Původní ground truth s barevnými přechody a neredukovaným počtem tříd.



Obrázek 5.2: Ukázka nového ground truth s odstraněnými barevnými přechody a nahrazením oblastí pro trénování na 5 třídách (bílé oblasti - neznámý povrch, zelené - lesní porost, černé - obloha, modré - ledovec).

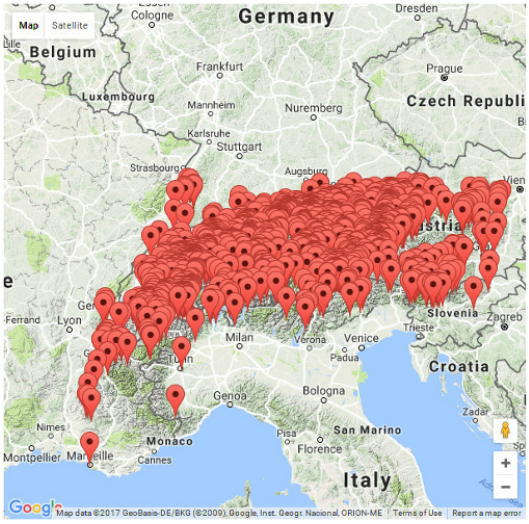
skytovaly se mezi nimi obrázky z jiné množiny a zároveň, aby vzdálenost mezi validační a testovací množinou byla co největší. Tyto požadavky jsou důležité pro objektivní hodnocení metod. Pokud by více obrázků obsahovalo stejné místo, pak by model byl testován na podobném obrázku, který již zpracovával u trénování, což by mohlo nadhodnotit skutečný výsledek.

Prvním krokem je načtení a zpracování souboru obsahujícího GPS souřadnice jednotlivých obrázků. Z těchto dat odebereme ty obrázky, které byly odstraněny v rámci čištění. V druhém kroku nad GPS souřadnicemi vytvoříme nejmenší obalující obdélník, který rozdělíme čtvercovou sítí. Následně vypočteme počet obrázků v jednotlivých segmentech sítě. Pokud je počet obrázků v nejpočetnějším segmentu příliš vysoký, pak zjemníme velikost čtvercové sítě a opakujeme znovu druhý krok. Čím více je obrázků v jednom segmentu, tím je následující výpočet rychlejší, ale méně přesný vzhledem k dodržení požadavků na množiny. Po dosažení optimálního počtu obrázků v nejpočetnějším segmentu ve třetím kroku vytváříme všechny možné obdélníkové podsítě sítě tak, aby obsahovaly požadovaný počet obrázků ve validační/trénovací množině. Z takto vzniklých podsítí vybereme ty, jejichž průměrná procentuální statistika typů oblastí v obrázcích, jenž jsou v dané podsíti obsaženy, je maximálně o $\pm 5\%$ rozdílná od celkové statistiky všech obrázků. Mezi takto vzniklými podsítěmi hledáme dvě takové, které neobsahují stejné obrázky a zároveň neexistují žádné dvě podsítě, kde minimální vzdálenost mezi obalujícími obdélníky obsažených obrázků je větší. Zbytek obrázků, které nejsou obsaženy ve validační a testovací množině, tvoří trénovací množinu.

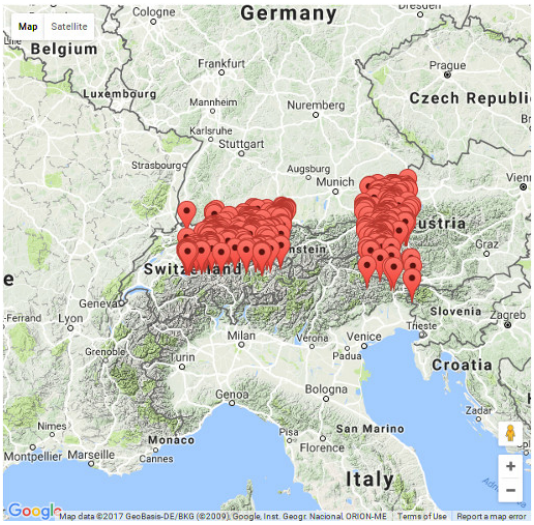
Ukázka rozložení vstupních obrázků v mapě podle jejich GPS souřadnic a příklad vypočtené testovací a validační množiny jsou vyobrazeny na obrázcích 5.3 a 5.4.

5.4 Převod GeoPose3K do formátu pro jednotlivé metody

Jednotlivé testované metody vyžadují individuální formát vstupních dat a maximální velikost dat, tedy rozlišení obrázků je omezeno buď metodou, nebo dostupným hardwarem. Prvním krokem je tedy změna velikosti obrázků, čímž ale přicházíme o detaily, tudíž některé



Obrázek 5.3: Ukázka rozložení obrázků v mapě před dělením na jednotlivé množiny.



Obrázek 5.4: Ukázka rozložení obrázků z testovací(vpravo) a validační množiny(vlevo) v mapě. Minimální vzdálenost obalujucích obdelníků množin je přibližně 150km.

experimenty jsou prováděny i s upravenou datovou sadou, která kromě zmenšených obrázků obsahuje navíc výřezy z původních nezmenšených obrázků. Výřezy vytváříme z původního obrázku pomocí náhodného vertikálního a horizontálního posunu okna o požadované velikosti. Po každém posunu okna náhodně výřez zahodíme, nebo spustíme analýzu, ve které vypočítáme statistiku obsažení typů oblastí v daném výřezu. Pokud obsahuje pouze jeden typ oblasti, nebo výrazně převažují oblasti reprezentující třídu neznámý povrch, pak výřez zahodíme. Rozsah náhodného posunu okna a atribut udávající maximální hranici, kdy může převažovat třída neznámý povrch, jsme nastavili experimentálně s ohledem na redukci výsledného počtu obrázků. Vzhledem k nutnosti porovnání s ostatními metodami a experimenty výřezy přidáváme pouze do trénovací množiny a v testovací a validační množině jsou pouze zmenšené obrázky.

V druhém kroku následuje samotné převedení do požadovaného tvaru. U modelů DeepLab a FCN převádíme ground truth do podoby, ve které barevné pixely reprezentující jednotlivé třídy nahradíme hodnotou klasifikátoru sítě pro danou třídu. Pro framework ALE další úprava obrázků není potřeba.

V posledním kroku pro každou metodu příslušně upravené obrázky a ground truth vytvoříme odpovídající adresářovou strukturu a uložíme textové soubory obsahující rozdělení obrázků do jednotlivých množin.

Kapitola 6

Experimenty s vybranými metodami

Cílem experimentů je pomocí metod FCN, DeepLab a frameworku ALE natrénovat na testovací množině datové sady GeoPose3K modely, které budou vhodné pro sémantickou segmentaci v horském prostředí. Při experimentech pro porovnání jednotlivých modelů primárně používáme metriku mean IU a pro porovnání dosažených výsledků jednotlivých tříd metriku IU. Metriky jsou definovány v kapitole 6.1. Nejprve se v kapitole 6.2 zaměříme na metodu FCN, ke které jsou k dispozici před-trénované modely na datových sadách obsahující horské třídy, a je tedy možné testovací množinu vyhodnotit již před dotrénováním modelů. Poté v následující kapitole 6.3 experimentujeme s metodou DeepLab, přesněji tedy s modely pro obě její verze DeepLab v1 a DeepLab v2. Při posledním experimentu v kapitole 6.4 natrénujeme model pomocí frameworku ALE. Následně v kapitole 6.5 z dotrénovaných modelů vybereme ty, které dosáhly podle metriky mean IU nejlepších výsledků. Rozsegmentovanou testovací množinu datové sady pomocí vybraných modelů použijeme nejprve pro dotazník, ve kterém respondenti vyhodnotí výsledky vizuálně. Následně segmenty použijeme pro lokalizaci, kdy výsledky vyhodnotíme z hlediska přesnosti určení orientace kamery. V poslední kapitole 6.6 shrneme a porovnáme dosažené výsledky jak z hlediska použitých metrik, tak z hlediska lidského vyhodnocení dotazníků a použití segmentů pro určení orientace kamery.

Pro lepší orientaci v experimentech je v příloze B navigační diagram, který obsahuje jak tok odpovídající pořadí provádění experimentů a jejich kroků (červené šipky), tak i návaznosti použití modelů mezi jednotlivými kroky (černé přímky). V průběhu popisu experimentů budeme uvádět čísla příslušných přechodů, kterými se v grafu dostaneme k popisovanému experimentu, například (nav. diagram 1.1.2).

Vzhledem k výpočetní náročnosti trénování vybraných metod lze pro trénování a testování na dostupných výpočetních zdrojích použít pouze obrázky s omezenou maximální velikostí. Maximální velikost obrázků je rozdílná pro trénování a testování, ale liší se také pro jednotlivé metody a jejich konkrétní konfigurace. Aby bylo možné výsledky trénování objektivně porovnávat, je potřeba, aby všechny metody byly trénovány a poté testovány na stejných datech, tedy i stejně velkých datech. Pro trénování většiny metod jsme zvolili obrázky s maximálním rozlišením do 400x400 pixelů, které je vzhledem k výpočetním zdrojům maximum pro většinu metod. Testování má u řady metod menší nároky na výpočetní zdroje, tudíž kromě segmentace relativně malých obrázků do 400x400 pixelů lze segmento-

vat i oproti malým obrázkům velké obrázky[1] s maximálním rozlišením do 800x800 případně 1000x1000 pixelů[2]. Natrénované modely budou testovány jak na malých, tak na velkých obrázcích, kdy velikost malých obrázků bude odpovídat velikosti obrázků, na kterých byl model trénován, a tedy pro tyto obrázky by měl být nejvíce přizpůsoben a mít nejlepší výsledky. Velké obrázky obsahují oproti malým obrázkům více detailů, a tedy i po segmentaci bude vyžadována větší přesnost, což může způsobit, že metody budou na velkých obrázcích dosahovat horších výsledků. Při dosažení horších výsledků tedy bude potřeba rozšířit prováděné experimenty s cílem odstranit, nebo alespoň minimalizovat rozdíl mezi výsledky na malých a velkých obrázcích. Trénování bude vždy probíhat na trénovací množině, během trénování může být postup učení průběžně kontrolován na přetrénování a vyhodnocován na validační množině. Dotrénované modely budou testovány na testovací množině.

Originální obrázky k segmentům, které se v rámci této kapitoly vyskytují v ukázkách segmentace, jsou převzaty z datové sady GeoPose3K [11].

6.1 Použité metriky

Jako základní metriku pro hodnocení dosažených výsledků v jednotlivých experimentech používáme mean Intersection over Union (mean IU). Tuto metriku jsme zvolili, protože je běžně používána pro hodnocení metod sémantické segmentace a je použita například i v pracích DeepLab a FCN. Pro hodnocení jednotlivých tříd používáme metriku IU, ze které jako průměr IU jednotlivých tříd získáme metriku mean IU. Metrika IU pro každou třídu se vypočte jako true positive / (true positive + false positive + false negative), kde true positive je počet pixelů, které byly správně klasifikovány, false positive je počet pixelů, které byly špatně klasifikovány, a false negative je počet pixelů, které měly být, ale nebyly klasifikovány. Při experimentech jsou dále měřeny metriky accuracy, pixel accuracy, mean accuracy a frequency weighted accuracy, kdy naměřené hodnoty jsou uvedeny v celkových výsledcích v příloze C. Dle FCN [40] jsou metriky matematicky definovány následovně, nechť n_{ij} je počet pixelu třídy i predikovaných do třídy j, n_{cl} je počet různých tříd a $t_i = \sum_j n_{ij}$ je celkový počet pixelů třídy i. Metriky jsou počítány:

- IU: $n_{ii}/(t_i + \sum_j n_{ji} - n_{ii})$
- mean IU: $(1/n_{cl}) \sum_i n_{ii}/(t_i + \sum_j n_{ji} - n_{ii})$
- accuracy: $n_{ii}/\sum_i t_i$
- pixel accuracy: $\sum_i n_{ii}/\sum_i t_i$
- mean accuracy: $(1/n_{cl}) \sum_i n_{ii}/\sum_i t_i$
- frequency weighted accuracy: $(\sum_k t_k)^{-1} \sum_i t_i n_{ii}/\sum_i t_i$

6.2 Experimenty s FCN

K metodě FCN jsou k dispozici modely vycházející ze sítě VGG16, které jsou před-trénovány na datových sadách VOC, NYUD, PASCAL-Context a SIFT Flow. Nejprve jsme z těchto

[1] Pojmy malé a velké obrázky budou v uvedeném smyslu používány dále v textu.

[2] Dále v textu může být zápis maximální velikosti zkracován z např. 400x400 pixelů pouze na 400 pixelů.

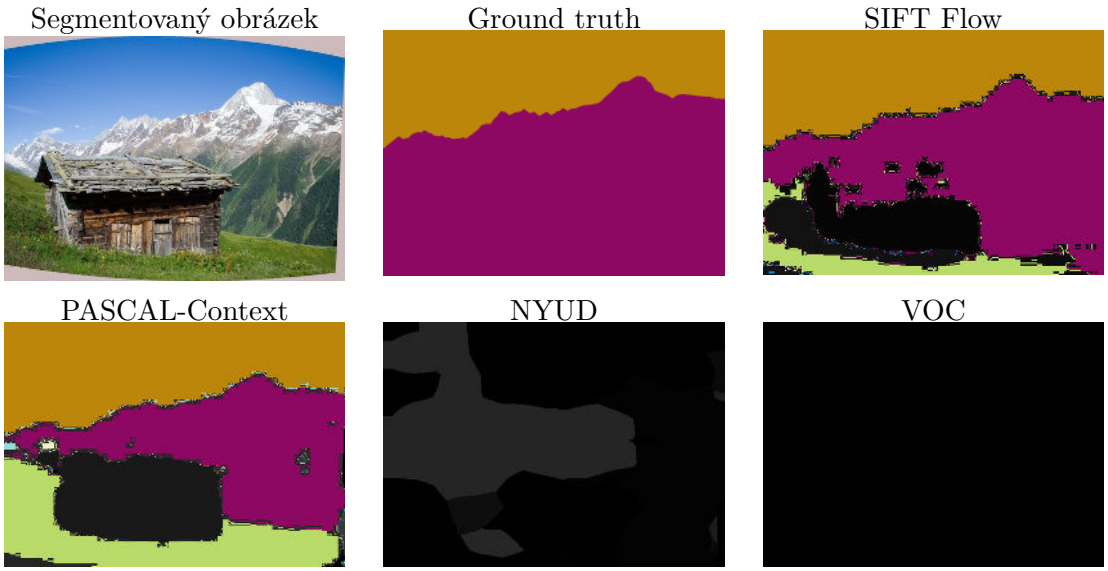
před-trénovaných modelů vybrali ty, které mají nejlepší předpoklad dosáhnout dobrých výsledků segmentace horského prostředí.

Datová sada PASCAL-Context obsahuje přes 400 tříd, ze kterých 7 tříd přibližně odpovídá horským oblastem, které chceme segmentovat. Sémantická část datové sady SIFT Flow obsahuje sice pouze 6 tříd, které přibližně odpovídají horským oblastem, ale zato celkový počet rozpoznávaných tříd je pouze 33, tudíž by rozpoznávání těchto 6 tříd mohlo být v modelu natrénováno lépe než rozpoznávání 7 tříd v modelu natrénovaném na datové sadě PASCAL-Context. Tuto úvahu ověříme v kapitole 6.2.1 při dotrénování těchto modelů s původním klasifikátorem, který klasifikuje oblasti do všech tříd z původní datové sady, na které byl model natrénován. Následně, aby bylo možné modely objektivně porovnat a dotrénovat na různých libovolných podmnožinách horských tříd, tak v kapitole 6.2.2 modely dotrénujeme s vlastním klasifikátorem obsahujícím pouze určité horské třídy. Při změně klasifikátoru pouze v modelu nahradíme poslední vrstvy, které nalezené oblasti přiřazují do určité třídy, ale samotné váhy v modelu zůstanou nezměněny. Nakonec výsledky experimentů s původním a vlastním klasifikátorem shrneme v kapitole 6.2.3, kdy předpokládáme, že experimenty budou dosahovat podobných výsledků, nebo modely s původním klasifikátorem, které obsahují kromě horských tříd i další, budou klasifikovat i do nehorských tříd, a tedy budou dosahovat horších výsledků.

Datové sady VOC, NYUD neobsahují žádné třídy, které odpovídají horskému prostředí, tedy modely se při před-trénování nesetkaly s horským prostředím, respektive pokud se setkaly, tak o tom neměly potřebnou informaci a zpracovaly ho například jako pozadí, které se ignoruje. Je tedy možné předpokládat, že dotrénování těchto modelů, které v před-trénované podobě mají výrazně horší výsledky, by nepřineslo lepší výsledky než dotrénovaní modelů před-trénovaných na datových sadách PASCAL-Context a SIFT Flow. Tento předpoklad jsme ověřili pomocí segmentace horského obrázku z datové sady GeoPose3K na nemodifikovaných modelech před-trénovaných na datových sadách VOC, NYUD, PASCAL-Context a SIFT Flow. Výsledky jsou vyobrazeny na obrázku 6.1, kde jsou barevně vyznačeny horské oblasti reprezentující horské třídy a ostatní rozpoznané oblasti jsou v odstínech šedi. Ze vzniklých segmentů je patrné, že modely před-trénované na datových sadách PASCAL-Context a SIFT Flow nejenže rozpoznaly horské třídy, ale navíc příslušné rozpoznané oblasti z poměrně velké části přibližně odpovídají oblastem v ground truth. Naopak modely před-trénované na datových sadách VOC a NYUD nerozpoznaly žádnou horskou třídu, ale ani v odstínech šedi nejsou oblasti, které by měly podobnost s objekty v segmentovaném obrázku nebo s oblastmi v ground truth.

6.2.1 Dotrénování modelů s původním klasifikátorem

Při experimentech s původním klasifikátorem (nav. diagram 1) používáme modely z FCN v nezměněné podobě tak, jak byly před-trénovány na datových sadách PASCAL-Context a SIFT Flow. Klasifikátory tedy obsahují všechny třídy, které byly v příslušných datových sadách a při trénování na datové sadě GeoPose3K jsme se pokusili modely dotrénovat tak, aby jednotlivé oblasti byly klasifikovány pouze do horských tříd. Nejprve jsme provedli experimenty se všemi horskými třídami, které jsme vhodně namapovali na třídy z původních datových sad, čímž jsme ověřili schopnost modelů učit se na horských datech, a získali jsme první výsledky pro porovnání modelů. Poté jsme zkoumali vliv odebrání tříd s malým procentuálním zastoupením v datové sadě GeoPose3K na dosažené výsledky segmentace. U tříd s malým procentuálním zastoupením je riziko, že se model nenaučí do těchto tříd správně klasifikovat, protože pro ně bude mít příliš málo charakteristických dat, což se ne-



Obrázek 6.1: Ukázka segmentace horské fotografie z datové sady GeoPose3K pomocí nemodifikovaných modelů z FCN. Oranžové oblasti reprezentují třídu obloha, fialové hory, zelené trávu a odstíny šedi ostatní třídy, které nejsou z horského prostředí.

gativně projeví na celkových výsledcích segmentace. Použití původního klasifikátoru nám umožňuje vyhodnotit testovací množinu datové sady GeoPose3K a segmentovat obrázky pomocí původního před-trénovaného modelu. Následně lze tyto výsledky porovnat s dotrénovanými modely jak pomocí statistik získaných při vyhodnocení testovací množiny, tak vizuálně pomocí porovnání segmentovaných obrázků.

Experimenty se všemi třídami

V prvním experimentu (nav. diagram 1.1) jsme při trénování použili všechny typy horských tříd, které jsou obsaženy v datové sadě GeoPose3K. Protože původní klasifikátor obsahuje jen některé z těchto tříd, bylo potřeba horské třídy namapovat na vizuálně podobné třídy, které klasifikátor podporuje. U modelu pascalcontext-fcn8s před-trénovaném na datové sadě PASCAL-Context bylo možné přímo přiřadit třídy obloha, lesní porost, voda, ledovec a tráva. Zbytek tříd skála, kamenná suť, útes, oblázky a přírodní prohlubně jsme shrnuli pod vizuálně podobnou třídu kámen. Poslední nepřiřazenou třídou je neznámý typ povrchu, který vzhledem k tomu, že se jedná o neznámý povrch v horském prostředí a případné nehorské oblasti chceme abstrahovat, tak lze považovat obecně za hory, a tedy i namapovat na třídu hory. Obdobně jsme namapovali třídy i pro model siftflow-fcn8s předtrénovaný na datové sadě SIFT Flow, u kterého pouze pro třídu ledovec neexistuje žádná přímá ani vizuálně podobná třída, a tudíž jsme ji označili jako třídu, která se má při učení ignorovat.

První krok (nav. diagram 1.1.1)

Při do-trénování obou modelů jsme zachovali přednastavené parametry sítě a u modelu pascalcontext-fcn8s jsme provedli 400 000 iterací a u modelu siftflow-fcn8s 100 000 iterací trénování. Trénování v prvním kroku probíhalo na obrázcích, které byly zmenšeny na maximální velikost 256 pixelů, na kterých také probíhalo vyhodnocení modelů. Při trénování jsme na validační množině dosáhli nejlepších výsledků (tabulka 6.1) mean IU 40% v iteraci 96000 pro model siftflow-fcn8s a mean IU 23.6% v iteraci 320000 pro model pascalcontext-

fcn8s. Po do-trénování jsme pomocí modelů, které byly vytvořeny v iteraci s nejlepším mean IU, vyhodnotili obrázky z testovací množiny. U těchto obrázků jsme dosáhli u modelu pascalcontext-fcn8s zvýšení mean IU z původních 2.9% na 17.1% a u modelu siftflow-fcn8s, který měl již bez trénování mírně lepší hodnocení, se mean IU zvýšilo z 4.6% na 37%, což je výrazně lepší výsledek. Poté jsme do-trénovaný model siftflow-fcn8s vyhodnotili na obrázcích z testovací množiny zmenšených na maximální velikost 1000 pixelů, kde již výsledky dosáhly pouze mean IU 17.2%.

Druhý krok (nav. diagram 1.1.2)

Model siftflow-fcn8s, který dopadl lépe, jsme se v druhém kroku pokusili ještě vylepšit dalším dotrénováním na datové sadě, která je rozšířena o výřezy z obrázků v jejich původním rozlišení. Tyto výřezy obsahují detaily, které byly zmenšením obrázků ztraceny a mohly by tedy vylepšit výsledky segmentace jak pro malé, tak hlavně pro velké obrázky. Dotrénování pozitivně ovlivnilo nejen výsledky na malých obrázcích, ale také výsledky velkých obrázků, které se zlepšily z mean IU 17.2% na 33.3%.

V jednotlivých krocích jsme dotrénováním docílili vylepšení modelů, které je patrné na výsledcích metriky mean IU, ale také vizuálně při porovnání segmentovaných obrázků (obrázek 6.2), které byly segmentovány pomocí modelů z jednotlivých kroků. Při segmentaci nedotrénovanými modely byla spousta oblastí klasifikovaných do nehorských tříd, ale v dalších krocích dotrénování takovýchto oblastí ubylo a rozsegmentové obrázky obsahují především oblasti klasifikované do horských tříd.

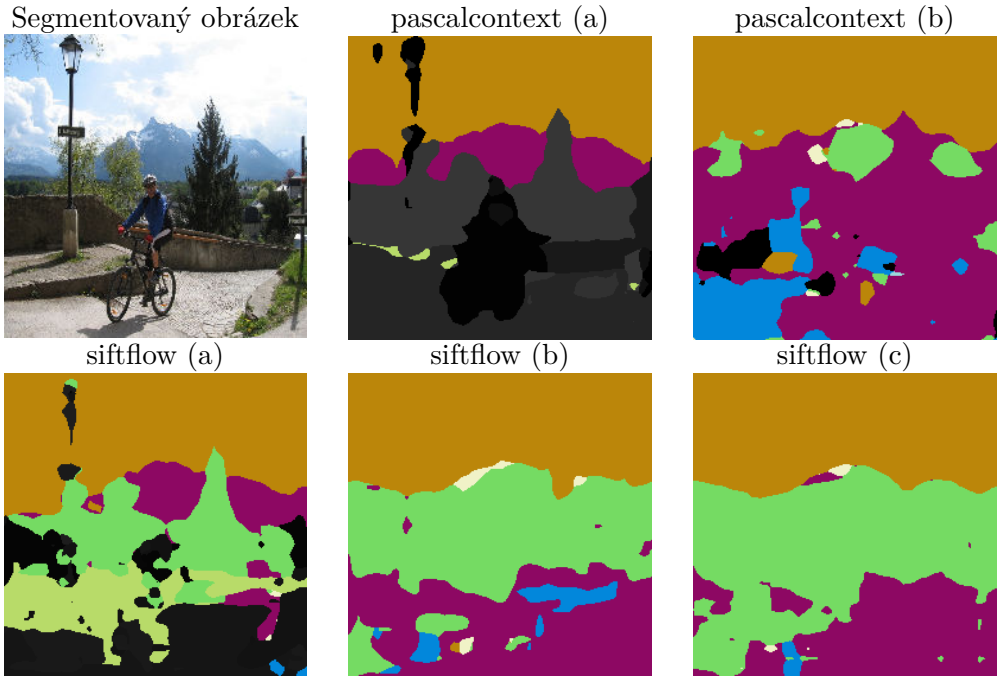
Všechny třídy	pascal.	sift.	pascal.	sift.	sift.
Krok	0.	0.	1.	1.	2.
Trénování					
Dat. sada s výřezy	-	-	NE	NE	ANO
Nejlepší iterace	-	-	320 000	96 000	88 000
Nejlepší iter. - mean IU (%)	-	-	23.6	39.8	43.9
Testování - mean IU (%)					
256px	2.9	4.6	17.1	37	48.8
1000px	-	-	-	17.2	33.3

Tabulka 6.1: Výsledky jednotlivých kroků dotrénování modelů pascalcontext-fcn8s a siftflow-fcn8s na datové sadě GeoPose3K a jejich vyhodnocení na testovací množině.

Experimenty s redukovaným počtem tříd

Experimenty s redukovaným počtem tříd (nav. diagram 1.2) jsme opět provedli s oběma modely, i když z předchozího experimentu vyplývá, že model pascalcontext-fcn8s dosahuje výrazně horších výsledků než model siftflow-fcn8s, ovšem oproti němu klasifikuje do třídy ledovec, která má sice v datové sadě zastoupení pouze 1%, ale například z hlediska geo-lokalizace horského prostředí na fotografii by jakožto část hor, která je trvale pokrytá sněhem, a tudíž se v průběhu roku nemění, mohla být významná. Ostatní třídy, které mají procentuální zastoupení v datové sadě menší než 3%, jsme namapovali na obecnou třídu hory. Takto upravená datová sada obsahuje třídy hory, obloha, lesní porost, voda a pro model pascalcontext-fcn8s navíc ledovec.

Pro dotrénování jsme použili datovou sadu s obrázky do maximální velikosti 400 pixelů pro model siftflow-fcn8s a 350 pixelů pro model pascalcontext-fcn8s. Vyhodnocení probíhalo



Obrázek 6.2: Porovnání výsledků segmentace obrázku z testovací množiny datové sady GeoPose3K pomocí modelů pascalcontext-fcn8s a siftflow-fcn8s. Obrázky (a) jsou segmentovány nedotrénovanými modely, obrázky (b) jsou segmentovány modely po 1 kroku dotrénování a obrázek (c) je segmentován pomocí modelu siftflow-fcn8s, který vznikl ve druhém kroku dotrénováním na datové sadě rozšířené o výřezy.

na obrázcích do maximální velikosti 350, 400, 800 pixelů a pro model siftflow-fcn8s i na obrázcích do 1000 pixelů. Velikost obrázků pro trénování a maximální velikosti obrázků pro testování jsme zvolili s ohledem na nároky jednotlivých metod na dostupné výpočetní zdroje.

První krok (nav. diagram 1.2.1)

Nejprve jsme vyhodnotili testovací množinu z upravené datové sady na před-trénovaném modelu. Následně jsme v prvním kroku základní modely dotrénovali na datové sadě GeoPose3K, což na testovací množině do velikosti 350 pixelů přineslo u modelu siftflow-fcn8s zlepšení mean IU z 5.2% na 53.8% a u modelu pascalcontext-fcn8s zlepšení z 3% na 15.7%. Metoda siftflow-fcn8s dosáhla dotrénováním na datové sadě s redukovaným počtem tříd výrazného zlepšení jak u malých, tak částečně i u velkých obrázků, které sice oproti počátečnímu modelu jsou výrazně lepší, ale oproti malým obrázkům mají téměř poloviční úspěšnost.

Druhý krok (nav. diagram 1.2.2)

V druhém kroku jsme modely z nejlepší iterace předchozího kroku dotrénovali na datové sadě rozšířené o výřezy. Tímto dotrénováním se výsledky na malých obrázcích vylepšily pouze mírně, ale u velkých obrázků došlo k významnému zlepšení. U modelu siftflow-fcn8s pro velké obrázky do 800 pixelů došlo k vylepšení mean IU z 29.8% na 56.1%, což překonalo i výsledek 54.8% dosažený na malých obrázcích do 350 pixelů. Také se zmenšil rozdíl mezi výsledky vyhodnocení mean IU pro velké obrázky do 800 a do 1000 pixelů, kdy původní rozdíl byl 4.5% a nyní je pouze 0.6%. U metody pascalcontext-fcn8s došlo sice ke zlepšení na velkých obrázcích do 800 pixelů z 4.9% na 8.5%, ale stále jsou výsledky (tabulka 6.2) výrazně

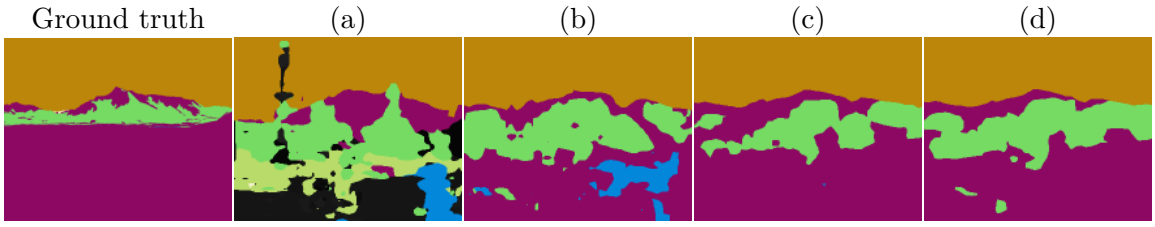
horší než 20.9% získaných na obrázcích do 350 pixelů. Metoda pascalcontext-fcn8s dosáhla nejlepších výsledků trénování v iteraci 64 000 a dalších 336 000 iterací neměla metoda tendenci se zlepšovat, a tedy její další dotrénování již nemá smysl, a dále dotrénujeme pouze metodu siftflow-fcn8s.

Třetí krok (nav. diagram 1.2.3)

V posledním kroku jsme upravili u metody siftflow-fcn8s parametry trénovaní s cílem zmenšit velikosti kroku, kterým se upravují váhy, čímž chceme dosáhnou jemnějšího doladění dotrénovaného modelu, a tím dále vylepšit dosažené výsledky. Původní nastavení, které má fixní *lerning rate*, jsme nahradili za krokový *lerning rate*, kdy se každých 100 000 iterací sníží *lerning rate* 10x, a celkový počet iterací jsme zvýšili na 200 000. Tato úprava přinesla drobné vylepšení výsledků (tabulka 6.2) v řádu jednotek procent oproti modelu z předchozího kroku. Nově natrénovaný model dosahuje u malých obrázků do 350 pixelů mean IU 57.4% a u velkých obrázků do 800 pixelů 57.3%. Experimenty s dalším snižováním *lerning rate* nebo s použitím jiné politiky změny *lerning rate*, již nepřinesly žádné další vylepšení. Rozdíly mezi jednotlivými kroky trénování jsou patrné i vizuálně na rozsegmentovaných obrázcích (obrázek 6.3), kdy dotrénováním ubývá oblastí klasifikovaných do nehorských tříd, ale mění se i samotné horské oblasti.

Redukované třídy	pascal.	sift.	pascal.	sift.	pascal.	sift.	sift.
Krok	0.	0.	1.	1.	2.	2.	3.
Trénování							
Dat. sada s výřezy	-	-	NE	NE	ANO	ANO	ANO
Nejlepší iterace	-	-	364 000	92 000	64 000	84 000	156 000
Nejlepší it. - mIU (%)	-	-	14.9	49	22.5	51.5	53
Testování - mean IU (%)							
350px	3	5.2	15.7	53.8	20.9	54.8	**57.4**
800px	2.5	4.7	4.9	25.3	8.5	55.4	**56.7**
1000px	-	4.7	-	25.3	-	55.4	56.7

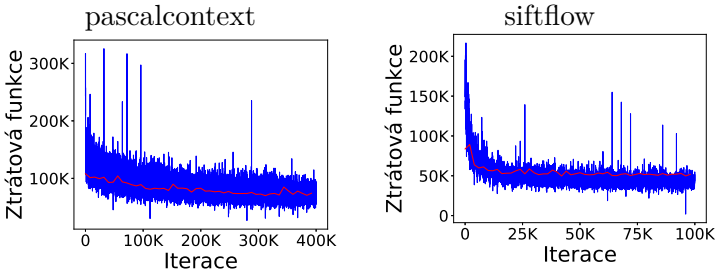
Tabulka 6.2: Výsledky jednotlivých kroků trénování modelů pascalcontext-fcn8s a siftflow-fcn8 na datové sadě GeoPose3K s redukovaným počtem tříd a jejich vyhodnocení na testovací množině.



Obrázek 6.3: Porovnání výsledků segmentace obrázku z testovací množiny datové sady GeoPose3K, která byla redukována pouze na 4 třídy, pomocí modelu siftflow-fcn8s. Obrázek (a) je segmentován nedotrénovaným modelem, obrázek (b) je segmentován modelem z prvního kroku trénovaným na základní datové sadě, obrázek (c) je segmentován modelem z druhého kroku trénovaným na datové sadě rozšířené o výřezy a obrázek (d) je segmentován modelem dotrénovaným se zjemněným *lerning rate*.

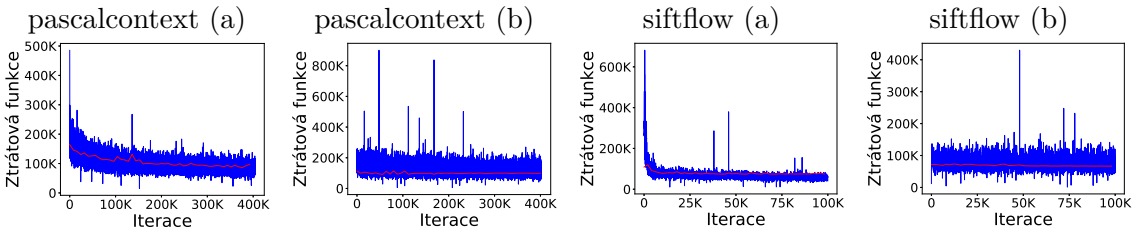
Zhodnocení experimentů s původním klasifikátorem

Při experimentech s původním klasifikátorem jsme nejprve prokázali schopnost modelů učit se na horských datech obsažených v datové sadě GeoPose3K, tato schopnost je patrná jak z výsledků mean IU, tak z grafu ztrátové funkce 6.4, ve kterém hodnota ztrátové funkce vyjadřuje průběh učení. Model před-trénovaný na datové sadě SIFT Flow se velice rychle v prvních 20 000 iteracích adaptoval na datovou sadu a poté se pouze mírně vylepšoval. Druhý model před-trénovaný na datové sadě PASCAL Context se adaptoval výrazně pomaleji přibližně prvních 150 000 iterací a také s horšími výsledky než první model, poté se sice ještě mírně vylepšil, ale celkově dosáhl horších výsledků.



Obrázek 6.4: Graf ztrátové funkce prvního kroku dotrénování modelů pascalcontext-fcn8s a siftflow-fcn8s na datové sadě s neredukovaným počtem tříd. Graf obsahuje hodnoty ztrátové funkce na trénovací množině (modrá barva) a na validační množině (červená barva).

Následnou redukcí tříd ze 6 na 4 u modelu siftflow-fcn8s jsme ukázali, že u této metody nelze segmentaci tříd, které mají malé procentuální zastoupení v datové sadě, dostatečně dotrénovat a jejich zobecnění se výrazně projevilo na vylepšení výsledků mean IU po prvním kroku z 37% dosažených u neredukovaných tříd na 53.8%. Vylepšení je také vidět na grafu ztrátové funkce (obrázek 6.5), kde rychlost adaptace na datovou sadu ještě vzrostla a následné mírné vylepšování je výraznější. Redukce ze 7 na 5 tříd u modelu pascalcontext-fcn8s nepřinesla v prvním kroku trénování žádné zlepšení, spíše naopak se výsledek mean IU mírně zhoršil z 17.1% 15.7% a průběh učení zůstal také téměř stejný.



Obrázek 6.5: Graf ztrátové funkce při dotrénování modelů pascalcontext-fcn8s a siftflow-fcn8s na datové sadě GeoPose3K s redukovaným počtem tříd. Na obrázcích (a) jsou grafy ztrátové funkce při trénování v prním kroku z před-trénovaných modelů, na obrázcích (b) jsou grafy ztrátové funkce při dotrénování modelů z prvního kroku na datové sadě rozšířené o výřezy. Graf obsahuje hodnoty ztrátové funkce na trénovací množině (modrá barva) a na validační množině (červená barva).

Experimenty jak s redukovanými, tak s neredukovanými třídami ukázaly významný vliv trénování s datovou sadou rozšířenou o výřezy na segmentaci velkých obrázků. Toto zlepšení je patrné na hodnotách mean IU pro velké obrázky, kdy zejména metoda siftflow-

fcn8s u datové sady s redukovanými třídami dosáhla stejných výsledků segmentace jak pro malé, tak pro velké obrázky.

Oba modely se nám podařilo dotrénovat na datové sadě GeoPose3K tak, že rozpoznávají převážně horské segmenty, a tedy rozsegmentované obrázky již neobsahují pouliční lampy, člověka či kolo. Model siftflow-fcn8s, který ve všech experimentech dosahoval lepších výsledků, se nám nakonec pomocí doladění s nižším *lerning rate* podařilo z původního mean IU 5% dotrénovat na 57.4%, které dosahuje i na velkých obrázcích. Model pascalcontext-fcn8s se nám podařilo dotrénovat z mean IU 3% na 20.9% na malých obrázcích a 8.5% na velkých obrázcích. Kompletní naměřené výsledky obou modelů jsou v příloze C.1.

6.2.2 Dotrénování modelů s vlastním klasifikátorem

Po experimentech s původním klasifikátorem následují experimenty s vlastním klasifikátorem (nav. diagram 2), ve kterých již není potřeba namapovat třídy z datové sady GeoPose3K na třídy z datové sady, na které byl model před-trénován, což nám umožňuje libovolně volit, které a kolik tříd se bude model učit klasifikovat. Při trénování tedy neupravujeme pouze datovou sadu, ale také samotné parametry modelu, kdy nahrazujeme původní klasifikátor za vlastní, který je určený pro specifický počet tříd, se kterými má model pracovat.

Nejprve jsme dotrénovali modely na datové sadě obsahující všechny třídy a trénování vyhodnotili nejen obecně pro celou testovací množinu, ale také zvlášť pro jednotlivé třídy. Ze získaných výsledků jsme ověřili schopnost dotrénování modelů na jednotlivých třídách, kdy jsme předpokládali, že třídy s velkým procentem výskytu v datové sadě dosáhnou lepších výsledků než třídy s nízkým výskytem. Poté jsme na základě získaných výsledků provedli experimenty s redukovaným počtem tříd, při kterých jsme zkoumali, jak se projeví vliv redukce třídy se špatným hodnocením na hodnocení ostatních tříd a na celkovém hodnocení.

Experimenty se všemi třídami

Hlavním cílem experimentů se všemi třídami (nav. diagram 2.1) je zjistit schopnost modelů učit se jednotlivé třídy, které mají různé procentuální obsažení v datové sadě a zjištění celkových výsledků při použití všech tříd. Pro dotrénování modelů siftflow-fcn8s a pascalcontext-fcn8s jsme použili datovou sadu GeoPose3K s 11 třídami a vzhledem k použití vlastního klasifikátoru tedy nedochází k mapování na třídy z jiných datových sad jako u předchozích experimentů s původním klasifikátorem. V datové sadě jsou pouze spojeny třídy neznámý typ povrchu a vše do 30 metrů pod obecnou třídu hory.

První krok (nav. diagram 2.1.1)
Na upravené datové sadě jsme dotrénovali základní před-trénované modely pascalcontext-fcn8s a siftflow-fcn8s. Model pascalcontext-fcn8s jsme dotrénovali standardních 400 000 iterací a na malých obrázcích do 400 pixelů dosáhl výsledku mean IU 17.1%. Model dosáhl velice podobných výsledků také pro velké obrázky do 800 pixelů, kde hodnota mean IU dosáhla 16%. Při porovnání výsledků IU jednotlivých tříd na malých obrázcích do 400 pixelů nejlepší hodnocení dosáhla třída obloha s IU 81.9% následována horami s 42.9% a lesním porostem s 31.5%. Nejhorší výsledky měly třídy přírodní prohlubně v terénu, oblázky, útes, pastviny a kamenná suť, které měly výsledky IU pod 1%.

Ztrátová funkce při dotrénování modelu siftflow-fcn8s na standardních 100 000 iteracích vykazovala neustále mírné zlepšování, tak jsme dotrénování modelu prodloužili na dvojnásobek, tedy 200 000 iterací, kde model s nejlepšími výsledky byl dotrénován v iteraci 136 000. Dotrénovaný model dosáhl na malých obrázcích do 400 pixelů výsledků mean IU 23.7% a na

velkých obrázcích do 800 pixelů 22.9%. Výsledné pořadí hodnocení tříd odpovídá výsledkům dosažených u metody pascalcontext-fcn8s, pouze hodnoty IU jsou u jednotlivých tříd vyšší. Nejlépe hodnocené třídy dosáhly IU 88.0% u oblohy, 53.0% u hor a 37.3% u lesního porostu.

Oba modely se nám tedy podařilo dotrénovat na datové sadě obsahující všechny třídy. Při seřazení dosažených výsledků IU podle procentuálního obsažení tříd v datové sadě (tabulka 6.3) je patrné, že třídy, které mají velké zastoupení v datové sadě, mohou být modelem lépe natrénovány, a tedy dosahují výrazně lepších výsledků než třídy s malým procentuálním zastoupením. Dobrých výsledků dosahuje i třída voda, která má v datové sadě zastoupení pouze 3.5%, a třída ledovec, která má zastoupení necelé 1%. Ostatní třídy s malým zastoupením nebyly při segmentaci buď vůbec klasifikovány, nebo jejich výsledky dosáhly hodnot nižších než 1%.

IU (%)	pascalcontext	siftflow
hory	**42.9**	**53.0**
obloha	**81.9**	**88.0**
lesní porost, prales	**31.5**	**37.3**
voda	**19.8**	**33.1**
holá skála	4.5	5
kamenná suť	0	0
ledovec	5.9	**20.7**
pastviny	<1	0
útes	0	0
oblázky	0	0
přírodní prohlubně v terénu	0	0

Tabulka 6.3: Naměřené hodnoty IU na testovací množině datové sady GeoPose3K rozsegmentované pomocí modelů pascalcontext-fcn8s a siftflow-fcn8s dotrénovaných na datové sadě obsahující všechny horské třídy. Výsledky IU jsou seřazeny podle procentuálního zastoupení tříd v datové sadě od nejčetnějších po nejméně obsažené.

Experimenty s redukovaným počtem tříd

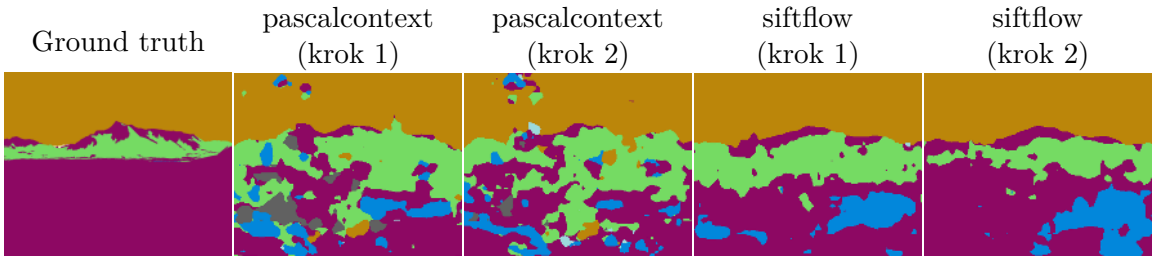
Při experimentech s redukovaným počtem tříd vycházíme z hodnocení IU pro jednotlivé třídy, které jsme získali v minulém experimentu, a ověříme, zdali zobecněním tříd se špatným hodnocením dosáhneme lepších výsledků učení. Předchozí experiment také ukázal, že při použití vlastního klasifikátoru rozdíl mezi výsledky testování na malých a velkých obrázcích je pouze malý, tedy se v experimentech zaměříme i na důležitost kroku dotrénování modelu na datové sadě rozšířené o výřezy. Při všech experimentech jsme oba modely trénovali na malých obrázcích do velikosti 400 pixelů. Model pascalcontext-fcn8s jsme trénovali standardních 400 000 iterací a u modelu siftflow-fcn8s jsme stejně jako v předchozím experimentu zdvojnásobili počet iterací na 200 000.

Experiment 1 (nav. diagram 2.2) - První krok (nav. diagram 2.2.1)
V prvním experimentu jsme v datové sadě nechali pouze pět tříd, které dosáhly nejlepšího hodnocení, tedy třídy obloha, hory, lesní porost, voda a ledovec. Základní před-trénované modely pascalcontext-fcn8s a siftflow-fcn8s jsme v prvním kroku dotrénovali na datové sadě bez výřezů. U modelu pascalcontext-fcn8s jsme dotrénováním dosáhli hodnocení mean IU

na malých obrázcích do 400 pixelů 31.6% a na velkých obrázcích do 800 pixelů 29.6%, což ukazuje, že redukcí tříd jsme dosáhli téměř dvojnásobného zlepšení. Pořadí v hodnocení úspěšnosti tříd se oproti experimentu se všemi třídami nezměnilo a odpovídá tedy prvním pěti třídám. Samotné hodnoty IU jsou velice podobné, třídy hory, lesní porost a ledovec dosáhly lepšího hodnocení, ale hodnocení ostatních tříd se mírně zhoršilo. Model se podařilo úspěšně naučit, aby zredukované třídy klasifikoval jako obecnou oblast hory. Následným dotrénováním modelu na datové sadě rozšířené o výřezy nedošlo ke zlepšení hodnocení, spíše k drobnému poklesu, kdy výsledky mean IU na malých obrázcích klesly na 30% a na velkých obrázcích na 29.4%. Zhoršení nastalo u všech tříd, kromě třídy ledovec u které došlo k mírnému zlepšení. Dotrénováním modelu siftflow-fcn8s jsme dosáhli u malých obrázků do 400 pixelů výsledků mean IU 46.6% a u velkých obrázků do 800 pixelů 45.1%. Redukcí tříd jsme podobně jako u modelu pascalcontext-fcn8s docílili přibližně dvojnásobného zlepšení. Výsledky IU jednotlivých tříd nám ukázali, že oproti nezredukovaným třídám dochází ke zlepšení u tříd hory, obloha a ledovec.

Experiment 1 - Druhý krok (nav. diagram 2.2.2)

Při následném dotrénování modelu v druhém kroku na datové sadě rozšířené o výřezy došlo u malých obrázků k poklesu mean IU z 46.6% na 44.8% a u velkých obrázků do 800 pixelů zůstala hodnota nezměněna. Naopak u velkých obrázků do 1000 pixelů došlo k mírnému zlepšení z 44.1% na 44.7%. Dotrénováním se ovšem mírně upravily výsledky IU jednotlivých tříd, kdy hodnocení tříd hory a voda mírně vzrostlo na úkor ostatních tříd, kdy zejména třída ledovec klesla z 21.3% na 15%. Výsledky mean IU prvního a druhého kroku se u obou metod téměř neliší, ovšem výsledky IU ukazují, že dochází ke změnám v hodnocení jednotlivých tříd. Změny jsou patrné i z vizuálního porovnání segmentů (obrázek 6.6), které ukazuje výrazné změny mezi segmenty získanými pomocí modelů dotrénovaných v prvním a druhém kroku experimentu.



Obrázek 6.6: Ukázka rozsegmentování obrázků pomocí modelů pascalcontext-fcn8s a siftflow-fcn8s získaných po prvním a druhém kroku experimentu, kdy dotrénování probíhalo na datové sadě GeoPose3K, který byla zredukována na 5 tříd.

Experiment 2 - První krok (nav. diagram 2.3)

V druhém experimentu při trénování základních modelů pascalcontext-fcn8s a siftflow-fcn8s jsme použili stejnou datovou sadu jako v předchozím experimentu, ovšem hned v prvním kroku je rozšířená o výřezy. Model pascalcontext-fcn8s po dotrénování dosahoval výsledků mean IU 30.5% a model siftflow-fcn8s 42.5%. U obou modelů dotrénování přineslo horší celkové výsledky mean IU, než trénování bez výřezů, a to jak na malých, tak velkých obrázcích. Horší výsledky ukazuje i porovnání výsledků IU jednotlivých tříd, kde pouze třída hory dosáhla v obou modelech mírného zlepšení, ale ostatní třídy dopadly stejně, nebo hůře. Dosažené výsledky jsou spolu s výsledky předchozího experimentu zobrazeny v tabulce 6.4.

5 tříd	pascal.	sift.	pascal.	sift.	pascal.	sift.
Krok	1.	1.	2.	2.	1.	1.
Trénování						
Dat. sada s výřezy	NE	NE	ANO	ANO	ANO	ANO
Nejlepší iterace	392 000	120 000	312 000	144 000	312 000	144 000
Nejlepší iter. - mean IU	30.9	41.3	30.6	41.2	30.2	40.2
Testování - mean IU						
350px	31.7	46.4	30	44.8	30.5	42.5
400px	31.6	**46.6**	30	45.3	30.6	43
800px	29.6	**45.1**	29.4	**45.2**	30	43.2
1000px	-	44.2	-	44.7	-	42.7
Testování - IU						
hory	43.7	**57.5**	41.7	**57.8**	45.1	**58.0**
obloha	81.8	**88.6**	77.8	87.7	79.6	87.7
lesní porost, prales	32.0	34.3	30.3	31.7	30.4	28.6
voda	16.5	31.2	15.1	34.5	15.9	30.5
ledovec	14.0	21.3	14.8	15.0	12.4	10.0

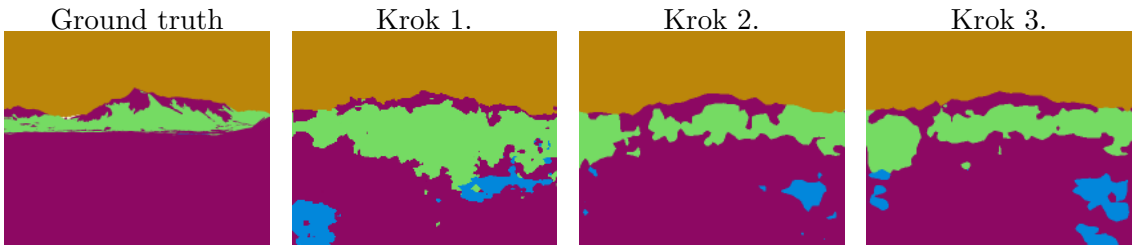
Tabulka 6.4: Výsledky trénování, validace a testování modelů pascalcontext-fcn8s a siftflow-fcn8s na datové sadě GeoPose3K s redukovaným počtem tříd na 5.

Experiment 3 (nav. diagram 2.4) - První krok (nav. diagram 2.4.1)
Ve třetím experimentu jsme vyzkoušeli zredukovat datovou sadu o třídu ledovec, která má v hodnocení u předchozích experimentů s redukcí na 5 tříd nejhorší výsledky a zároveň její výskyt je v datové sadě GeoPose3K menší než 1%. Základní před-trénované modely pascalcontext-fcn8s a siftflow-fcn8s jsme tedy v prvním kroku dotrénovali na datové sadě obsahující pouze třídy hory, obloha, lesní porost a voda. Po dotrénování model pascalcontext-fcn8s dosahuje na malých obrázcích do 400 pixelů výsledky mean IU 35.6% a na velkých obrázcích 33.9%, což přináší pouze 4% zlepšení. Výsledky IU jednotlivých tříd ukazují, že ke zlepšení došlo hlavně u tříd hory a voda.

Experiment 3 - Druhý krok (nav. diagram 2.4.2)
Následně jsme v druhém kroku dále dotrénovali model na datové sadě rozšířené o výřezy, což stejně jako při experimentu s 5 segmenty přineslo mírné zhoršení na malých obrázcích a zlepšení na velkých obrázcích. Dotrénování se také projevilo na zlepšení výsledků IU tříd hory a voda, ale hodnocení ostatních tříd se zhoršilo. U modelu siftflow-fcn8s jsme dotrénováním dosáhli na malých obrázcích do 400 pixelů výsledků mean IU 55.1%, což je vylepšení oproti výsledkům experimentu s 5 segmenty, kde mean IU po prvním kroku dosáhlo pouze 46.6%. V druhém kroku po dotrénování modelu na datové sadě rozšířené o výřezy výsledky stejně jako u modelu pascalcontext-fcn8s ukázaly zlepšení pouze na velkých obrázcích, ale na malých obrázcích došlo k zhoršení, i když pouze o 0.1%. Výsledky IU pro jednotlivé třídy ukázaly, že opět došlo ke změnám, ovšem nyní se vylepšily výsledky pro třídy lesní porost a voda, naopak výsledky hor, které se u modelu pascalcontext-fcn8s zvýšily, se zde snížily.

Experiment 3 - Třetí krok (nav. diagram 2.4.3)
Model jsme se poté ve třetím kroku pokusili dále vylepšit pomocí jemného doladění změnou *lerning rate*, které se po 100 000 iteracích 10x snížilo. Model se nám úspěšně povedlo vylepšit a po dotrénování dosahuje výsledků mean IU na malých obrázcích do 400 pixelů 56.8% a na velkých obrázcích do 800 pixelů 55.9%. Ovšem nejlepší model z trénování,

který tohoto výsledku dosáhl, vznikl v iteraci 66 000, kdy ještě nedošlo ke zjemnění *lerning rate*, takže změna na vylepšení neměla vliv a k vylepšení tedy přispěla pouze delší doba trénování. Dotrénováním se opět změnily i poměry výsledků IU mezi jednotlivými třídami. Změny v jednotlivých krocích jsou vidět i při vizuálním porovnání segmentů získaných pomocí modelů z jednotlivých kroků (obrázek 6.7), kde například na vrcholu pohoří je patrný pokles úspěšnosti třídy hory, který u metody siftflow-fcn8s nastal v druhém kroku experimentu. Celkové výsledky z trénování obou modelů na datové sadě se 4 třídami jsou shrnuty v tabulce 6.5.



Obrázek 6.7: Ukázka segmentů získaných v jednotlivých krocích dotrénování modelu siftflow-fcn8s, na kterých je patrné zhoršení výsledků třídy hory, které nastalo v druhém kroku, jako zakulacení vrcholu pohoří.

4 třídy	pascal.	sift.	pascal.	sift.	sift.
Krok	1.	1.	2.	2.	3.
Trénování					
Dat. sada s výřezy	NE	NE	ANO	ANO	ANO
Nejlepší iterace	392 000	188 000	264 000	176 000	60 000
Nejlepší iter. - mean IU	35.7	50.1	35.1	50.8	51.5
Testování - mean IU					
350px	35.6	54.7	33.7	54.6	**56.4**
400px	35.6	55.1	34.1	54.9	**56.8**
800px	33.9	53.4	34.4	54.3	**55.9**
1000px		52		53.4	**54.8**
Testování - IU					
hory	46.1	**62.1**	48.0	59.1	61.4
obloha	81.8	**88.7**	75.4	88.1	88.2
lesní porost, prales	32.0	34.4	25.2	36.9	**36.2**
voda	16.8	35.3	21.0	35.6	**41.3**

Tabulka 6.5: Výsledky trénování, validace a testování modelů pascalcontext-fcn8s a siftflow-fcn8s na datové sadě GeoPose3K s redukovaným počtem tříd na 4.

Shrnutí

Experimenty se snižováním počtu tříd jsme ukázali, že třídy se špatným hodnocením lze spojit a přidat pod obecnou třídu hory, což se pozitivně projeví jak na celkovém hodnocení, tak i na hodnocení samotné třídy hory. Spojování tříd pod obecnou třídu hory je možné vzhledem k tomu, že cílem experimentů je co nejvíce se přiblížit k horským datům z OpenStreetMap a ostatní nehorské objekty abstrahovat. Nejprve jsme tedy redukovali z 11 na 5 tříd a poté na 4 třídy, což u modelu pascalcontext-fcn8s přineslo zlepšení mean IU ze

17% na 31.6 u 5 tříd a 35.6% u 4 tříd a pro model siftflow-fcn8s z 23.7% na 46.6% u 5 tříd a 56.8% u 4 tříd.

Výsledky experimentů dále ukázaly, že základní modely dotrénované na datové sadě rozšířené o výřezy dosahují horších výsledků, než základní modely dotrénované na datové sadě bez výřezů, a to na malých i na velkých obrázcích. Dotrénování modelu, který byl v prvním kroku natrénovaný bez výřezů, na malých obrázcích nepřineslo zlepšení a na velkých pouze mírné. Došlo ke zlepšení některých tříd za cenu zhoršení jiných, kdy v obou modelech se zlepšovaly a zhoršovaly jiné třídy, což vyvrací možnost, že by výřezy obsahovaly výrazně více některých tříd, a tedy model se na nich oproti ostatním třídám lépe dotrénoval.

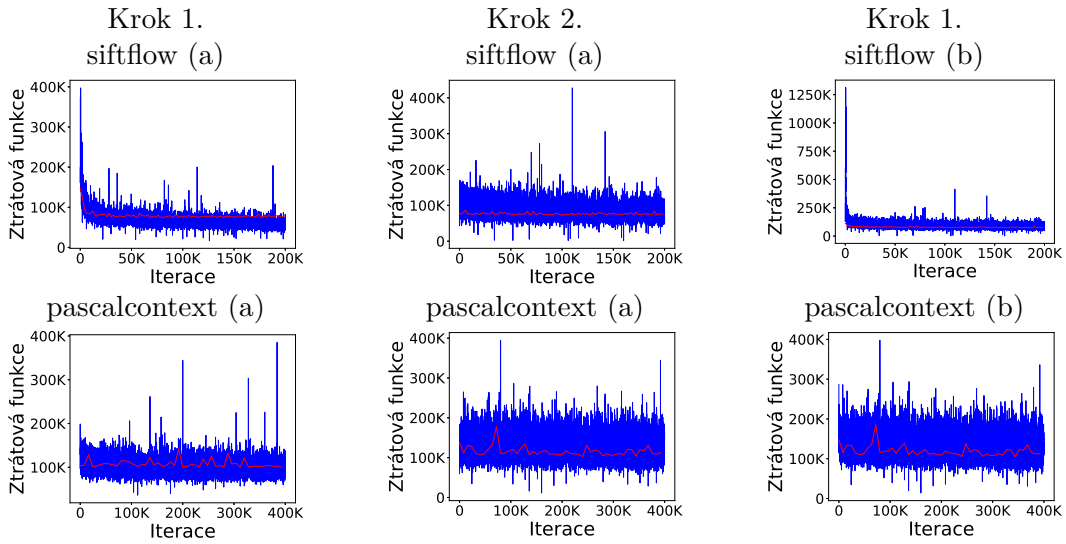
Zhodnocení experimentů s vlastním klasifikátorem

V experimentech s vlastním klasifikátorem jsme úspěšně dotrénovali modely pascalcontext-fcn8s a siftflow-fcn8s na datové sadě obsahující všechny horské třídy z datové sady GeoPose3K. Získaný model jsme vyhodnotili na testovací množině a z výsledků jsme vypočetli metriku IU, pomocí které jsme porovnali jednotlivé horské třídy. Nejlépe dopadly dvě třídy s nejvyšším procentuálním obsažením v datové sadě a poté s klesajícím obsažením tříd v datové sadě klesají i dosažené hodnoty IU, kdy pouze třída ledovec překonala některé četnější třídy. Na základě těchto výsledků jsme provedli sérii experimentů, které ukázaly, že snižováním počtu tříd se u obou metod zvyšuje celková úspěšnost klasifikace a i úspěšnost samotné třídy hory. I po redukci pouze na 4 třídy, kdy jsme v experimentech dosáhli na malých obrázcích do 400 pixelů nejlepších výsledků mean IU 56.8% u modelu siftflow-fcn8s, byly stále v datové sadě zachovány všechny významné třídy, přičemž jsme zobecnili pouze necelých 6% dat. Experimenty dosahovaly dobrých výsledků hned po prvním kroku trénování, a to jak na malých, tak na velkých obrázcích, mezi kterými byly pouze minimální rozdíly. Další trénování modelů v druhém kroku s datovou sadou rozšířenou o výřezy nepřineslo žádné výrazné zlepšení a i dotrénování základního modelu pouze na datové sadě rozšířené o výřezy nedosáhlo takových výsledků jako po prvním kroku trénování s nerozšířenou datovou sadou. Nepřínosnost výřezů je viditelná i ze ztrátové funkce při učení (obrázek 6.8), kde zejména u lepšího modelu siftflow-fcn8s je patrné, že v druhém kroku se trénovaný model na výřezech nezlepšuje. Rozdíl ve vlivu použití výřezů mezi trénováním s původním a vlastním klasifikátorem mohl být způsobem tím, že při použití původního klasifikátoru metoda stále klasifikovala i do nehorských tříd, a tedy potřebovala větší množství a detailnější data k tomu, aby se adaptovala.

Celkově dopadlo lépe dotrénování modelu siftflow-fcn8s, který dosáhl při testech mean IU 23.7% u všech tříd, 46.6% u redukce na 5 tříd a 56.8% u redukce na 4 třídy. Dotrénování modelu pascalcontext-fcn8s dopadlo hůře, kdy dosáhl mean IU 17% pro všechny třídy, 31.6% u redukce na 5 tříd a 35.6% u redukce na 4 typy třídy. Kompletní naměřené výsledky všech experimentů jsou v příloze C.2.

6.2.3 Zhodnocení trénování FCN

Při experimentech s metodou FCN jsme vybrali dva před-trénované modely pascalcontext-fcn8s a siftflow-fcn8s, které by mohly být vhodné k dotrénování na horských datech a jejich následné segmentaci. Před-trénované modely jsme dotrénovali s původním klasifikátorem, kdy bylo potřeba namapovat třídy z datové sady GeoPose3K na podobné třídy, které se vyskytují v datových sadách, na kterých byly modely před-trénovány. Dotrénováním jsme úspěšně ověřili schopnost obou modelů učit se na horských datech, kdy model před-trénovaný na datové sadě SIFT Flow dosahoval lepších výsledků než druhý model

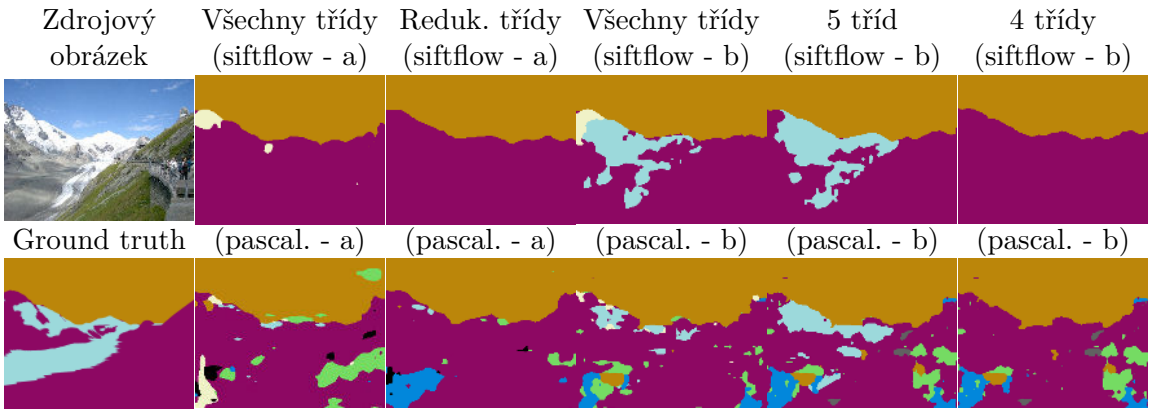


Obrázek 6.8: Grafy ztrátových funkcí trénování modelů na datové sadě s 5 třídami. Grafy (a) ukazují trénování modelu v prvním kroku bez výřezů a poté v druhém kroku na výřezech. Graf (b) ukazuje trénování přímo v prvním kroku datové sadě rozšířené o výřezy. V prvním řádku jsou grafy dotrénování modelu siftflow-fcn8s a v druhém řádku modelu pascalcontext-fcn8s. Grafy obsahují hodnoty ztrátové funkce na trénovací množině (modrá barva) a na validační množině (červená barva).

před-trénovaný na datové sadě PASCAL-Context. Následně jsme při experimentech detekovali tendenci zvyšování se celkových výsledků segmentace při redukci tříd, které mají malé procentuální obsažení v datové sadě. Tuto tendenci jsme poté blíže prozkoumali a potvrdili v experimentech s vlastním klasifikátorem, kdy jsme upravili a dotrénovali modely tak, aby přímo pracovaly s třídami obsaženými v datové sadě GeoPose3K. Nejlepších výsledků dosahovala třída obloha, poté hory a dále pořadí tříd seřazených podle procentuálního zastoupení v datové sadě odpovídalo pořadí úspěšnosti jednotlivých tříd dle metriky IU. Pouze jedinou třídou, která i přes svoje méně jak 1% obsažení v datové sadě dosahovala relativně vysokých výsledků, byla třída ledovec.

V hodnocení dosažených výsledků u experimentů s původním klasifikátorem docházelo u obou modelů k podstatným rozdílům mezi segmentací malých obrázků do 400 pixelů, na kterých byl model dotrénován, a segmentací velkých obrázků do 800 pixelů. Tento rozdíl jsme připisovali ztrátě detailů v malých obrázcích, ke které došlo při jejich zmenšení, a na základě tohoto předpokladu, jsme modely dotrénovali na datové sadě rozšířené o výřezy, což přineslo zlepšení, a to jak na malých, tak na velkých obrázcích, a rozdíly mezi velkými a malými obrázky se u metody siftflow-fcn8s úplně odstranily a u metody pascalcontext-fcn8s alespoň částečně minimalizovaly. Ovšem experimenty s vlastním klasifikátorem ukázaly, že pokud model nepracuje s původním klasifikátorem, který klasifikuje oblasti do řady nehorských tříd, tak není třeba model dotrénovat na detailech a rozdíly mezi malými a velkými obrázky se neprojevují. Použití vlastního klasifikátoru trénování na výřezech nemá vliv na celkové zlepšování výsledků, ale ukázalo se, že mění výsledky jednotlivých tříd, tedy vylepšuje některé třídy na úkor jiných. Při použití vlastního klasifikátoru se výhoda trénování s výřezy neprojevila ani u experimentů, kdy základní před-trénované modely byly hned v prvním kroku trénovány na datové sadě rozšířené o výřezy a modely dosahovaly horších výsledků než při trénování základních modelů na datové sadě bez výřezů.

U modelu pascalcontext-fcn8s lépe dopadlo trénování s vlastním klasifikátorem než s původním klasifikátorem, který klasifikuje 59 tříd, a ani trénováním na výřezech, nejspíše díky velkému počtu tříd, se nám nepodařilo docílit, aby model klasifikoval pouze do horských tříd. Naopak u modelu siftflow-fcn8s, kde původní klasifikátor používá pouze 33 tříd, se trénováním podařilo docílit klasifikace téměř všech pixelů do horských tříd, a tedy výsledky pro oba klasifikátory jsou téměř stejné. Z hlediska metriky mean IU dopadl o 1% lépe model s původním klasifikátorem. Rozdíly mezi jednotlivými natrénovanými modely jsou vidět i vizuálně na rozsegmentovaných obrázcích (obrázek 6.9), na kterých jsou patrné i rozdíly mezi segmentací pomocí modelu siftflow-fcn8s s původním a s vlastním klasifikátorem, které dosahují podobných výsledků.



Obrázek 6.9: Ukázka porovnání rozsegmentovaných obrázků pomocí nejlepších dotrénovaných modelů siftflow-fcn8s (první řádek) a pascalcontext-fcn8s (druhý řádek) na datových sadách s neredukovaným a redukovaným počtem tříd. Obrázky (a) jsou rozsegmentované modely s původním klasifikátorem a obrázky (b) modely s vlastním klasifikátorem.

Celkově ve všech experimentech dopadlo lépe dotrénování modelu siftflow-fcn8s, který dosáhl nejlepších výsledků u malých obrázků do velikosti 350 pixelů na testovací množině datové sady GeoPose3k redukované na 4 třídy mean IU 57.4% s původním a 56.4% s vlastním klasifikátorem. U malých obrázků do 400 pixelů dosáhl v experimentech s vlastním klasifikátorem a redukcí na 4 třídy mean IU 56.8% a při redukci na 5 tříd 46.6%. Druhý model pascalcontext-fcn8s dosáhl u malých obrázků do 350 pixelů na datové sadě s 5 třídami mean IU 20.9% s původním, 30% s vlastním klasifikátorem a na datové sadě se 4 třídami 35.6% s vlastním klasifikátorem.

6.3 Experimenty s DeepLab

Metoda DeepLab obsahuje řadu modifikací, kterými se snaží vylepšit výsledky sítí při použití s různými daty, a tyto modifikace jsou kombinovány v dostupných modelech. Metoda je aktuálně dostupná v první verzi (v1) a ve druhé verzi (v2). Dostupné modely k oběma verzím buď nejsou před-trénované nebo jsou před-trénované na datových sadách MS-COCO nebo VOC2012. Obě datové sady jsou určeny pro rozpoznávání objektů, tedy pro sémantickou segmentaci v horském prostředí nejsou příliš vhodné pro před-trénování modelů, a proto budeme experimentovat jak s verzí DeepLab v1, která obsahuje více před-chystaných modelů s různými modifikacemi, tak s novější verzí v2, která dosahuje v hodnocení na datové sadě VOC2012 lepších výsledků, ale jsou k ní k dispozici pouze dva před-chystané modely.

Nejprve se v kapitole 6.3.1 zaměříme na ty před-chystané modely pro verzi metody v1, které jsou určené pro plně anotovaná data, tedy ta, kdy pro každý pixel v obrázku máme informaci o tom, kterou třídu reprezentuje, což odpovídá informacím, které jsou v datové sadě GeoPose3K. Experimenty ověříme schopnost modelů učit se na horských datech a u modelů, které se podaří úspěšně dotrénovat, provedeme sérii experimentů, kterými zjistíme vliv tříd s malým procentuálním zastoupením v datové sadě GeoPose3K na trénování modelů, a tedy i na jejich dosažené výsledky. Následně se v kapitole 6.3.2 zaměříme na dva dostupné modely pro verzi v2, kde první model je založen na síti VGG16, ze které vycházejí všechny modely ve verzi v1, a druhý model, který dosahuje na datové sadě VOC2012 nejlepších výsledků, je založen na síti RESNET101. Při experimentech budeme vycházet z poznatků získaných v experimentech s první verzí a otestujeme, zda novější modely ve verzi v2 přinesou na horských datech lepší výsledky. Obě verze metody DeepLab nabízejí vylepšení dosažených výsledků pomocí podmíněných náhodných polí (CRF), kdy se pomocí vstupních obrázků a jejich segmentů získaných z modelu vypočtou nové segmenty. Pro jednotlivé metody DeepLab jsou specifikovány přesné parametry pro CRF. Ve všech experimentech budeme vylepšení pomocí CRF používat a sledovat jeho vliv na dosažené výsledky. Nakonec v kapitole 6.3.3 porovnáme dosažené výsledky v experimentech s oběma verzemi metody DeepLab, zhodnotíme přínos CRF a vhledem k podstatně delšímu trénování modelů z metody DeepLab v2 tak zhodnotíme i přínos délky trénování na dosažené výsledky oproti první verzi metody.

6.3.1 DeepLab v1

Pro metodu DeepLab ve verzi v1 (nav. diagram 3) je k dispozici řada modelů založených na síti VGG16, ze kterých je část určena pro slabě nebo částečně anotovanou datovou sadu a část pro silně anotovanou datovou sadu. Právě s modely pro silně anotovanou datovou sadu budeme experimentovat. Konkrétně budeme využívat základní model DeepLab a rozšířené modely DeepLab-COCO-LargeFOV, DeepLab-LargeFOV, DeepLab-MSC, DeepLab-MSc-COCO-LargeFOV, DeepLab-MSc-LargeFOV. Označení COCO indikuje, že model byl před-trénován na datové sadě MS-COCO, označení MSC indikuje, že model používá multi-scale funkci, kdy se na vstup vkládá obrázek v různých velikostech a poté se na úrovni pixelů výsledky pomocí multi-scale funkce spojí, a označení LargeFOV (Large Field-Of-View) indikuje, že model používá velké zorné pole. Zorné pole při klasifikaci pixelu udává, kolik okolních pixelů se zohlední při klasifikaci. V prvních experimentech otestujeme schopnost modelů učit se na horských datech obsažených v datové sadě GeoPose3K. Prozkoumáme, jaký vliv mají na výsledek jednotlivá rozšíření modelů, a ověříme, zda na trénování má vliv datová sada, na které byl model před-trénován, kdy předpokládáme, vzhledem k tomu, že datové sady neobsahují horské oblasti, tak nebudou mít výrazný vliv na dosažený výsledek. Z výsledků experimentů také získáme úspěšnosti segmentace jednotlivých tříd, kde předpokládáme existenci vztahu mezi procentuálním obsažením třídy v datové sadě GeoPose3K a jejím hodnocením. Na základě těchto informací provedeme experimenty, kdy zobecníme třídy s malým výskytem v datové sadě, které mají zároveň nízké dosažené výsledky, pod obecnou třídu hory, čímž se pokusíme zvýšit úspěšnost klasifikace ostatních typů tříd. Nakonec v poslední části této kapitoly vyhodnotíme dosažené výsledky.

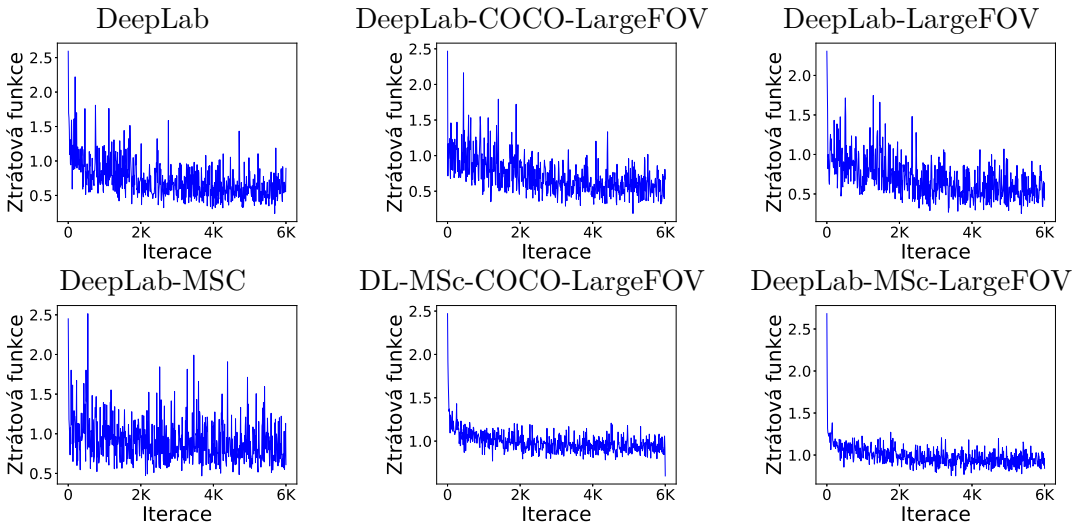
Při experimentech probíhá trénování modelů vždy na trénovací množině datové sady GeoPose3K s malými obrázky do maximální velikosti 400 pixelů, které byly interně modelem ořízнuty na velikost 317x317 pixelů. Vyhodnocení modelů probíhalo na testovací množině datové sady GeoPose3K jak s malými obrázky do 400 pixelů, tak s velkými obrázky do

800 pixelů, pokud to nároky metod na dostupné výpočetní zdroje umožnily. Jestliže u testování byla interně v modelu změněna velikost, tak to bylo vždy na větší rozměr a před následným vyhodnocením byly vzniklé segmenty oříznuty zpět na původní velikost.

Experimenty se všemi třídami

Cílem prvních experimentů (nav. diagram 3.1) s metodou DeepLab v1 je zjistit, které modely, nebo zdali je vůbec možné některé z modelů dotrénovat na datové sadě GeoPose3K pro použití k segmentaci horských scén. Z výsledků je třeba určit, která rozšíření metody DeepLab mají pozitivní vliv na výsledky trénovaní, a ověřit či vyvrátit předpoklad, že předtrénování modelů na datových sadách VOC2012 a MS-COCO, které neobsahují horské třídy, nemá vliv na dosažené výsledky.

Experiment 1 - První krok (nav. diagram 3.1.1)
Při experimentu jsme upravili všech 6 modelů tak, aby je bylo možné dotrénovat na trénovací množině datové sady GeoPose3K s 11 horskými třídami. U základní metody DeepLab jsme dotrénovali počáteční model a u ostatních modelů jsme dotrénovali před-trénované modely. Trénování probíhalo u všech modelů přednastavených 6000 iterací. Z průběhu ztrátové funkce (obrázek 6.10) je patrné, že všechny modely jsou schopny učit se na horských datech a modely, které neobsahují vylepšení MSC lépe než modely s tímto vylepšením.



Obrázek 6.10: Graf průběhu ztrátové funkce, při učení jednotlivých modelů z metody DeepLab v1 na datové sadě GeoPose3K, která obsahuje 11 horských tříd.

Závěry získané z průběhu ztrátové funkce potvrzují i výsledky testování, kdy nejlepší výsledky mean IU 26.7% a 26.5% s CRF na malých obrázcích do 400 pixelů dosáhl model dotrénovaný z DeepLab-LargeFOV, následován dotrénovaným modelem DeepLab-COCO-LargeFOV, který obsahuje stejné rozšíření, ale navíc byl před-trénován na datové sadě MS-COCO, s výsledky mean IU 25.5% a 24.8% s CRF. Na třetím místě skončil s velice podobnými výsledky dotrénovaný základní model DeepLab a poté se umístily modely DeepLab-MSC, DeepLab-MSc-COCO-LargeFOV a DeepLab-MSc-LargeFOV, které ukázaly, že modely s rozšířením MSC dosahují horších výsledků než ostatní modely. Nejlepší tři dotrénované modely jsme otestovali i na velkých obrázcích do 800 pixelů na kterých se modely umístily ve stejném pořadí, tedy nejlépe dopadl model DeepLab-LargeFOV s výsledky mean IU 22.7% a 22.6% s CRF, poté DeepLab-COCO-LargeFOV s 22.5% a 22%

s CRF a nakonec model DeepLab s 21.8% a 22.3% s CRF, kde tento model je jediný v tomto experimentu, u kterého použití CRF přineslo zlepšení. Výsledky mean IU všech modelů (tabulka 6.6) ukazují pouze malé rozdíly mezi modely před-trénovanými na datové sadě MS-COCO, VOC2012 a modely, které nebyly před-trénovány. U modelů bez rozšíření MSC byl nejlepší výsledek na modelu před-trénovaném na datové sadě VOC2012, poté na MS-COCO a nakonec u modelu bez před-trénování, kde ale rozdíl mezi nejlepším a nejhorším mean IU je pouze 1.5%.

Pořadí nejlepších modelů je také částečně patrné z výsledků IU pro jednotlivé třídy (tabulka 6.7), kde před použitím CRF je stejné, až na třídu hory. U modelů s MSC se pořadí obrátilo a lépe dopadl model před-trénovaný na datové sadě MS-COCO, ovšem tato tendence již není patrná u výsledků IU pro jednotlivé třídy, kdy některé jsou lépe hodnoceny v modelech před-trénovaných na datové sadě MS-COCO a některé na VOC2012.

11 tříd - mean IU (%)	400px	400px - CRF	800px	800px - CRF
DeepLab	24.8	25.1	21.9	22.3
DeepLab-COCO-LargeFOV	25.5	24.8	22.5	22
DeepLab-LargeFOV	**26.7**	**26.5**	**22.7**	**22.6**
DeepLab-MSC	21.9	21.2	-	-
DeepLab-MSc-COCO-LargeFOV	19.7	19.6	-	-
DeepLab-MSc-LargeFOV	19.5	19.3	-	-

Tabulka 6.6: Výsledky testování dotrénování modelů z metody DeepLab v1, na datové sadě GeoPose3K. Výsledky obsahují hodnocení mean IU před a po použití CRF.

Výsledky IU pro jednotlivé třídy (tabulka 6.7) ukazují souvislost mezi procentuálním obsažením tříd v datové sadě a dosaženými výsledky pro danou třídu. Četně obsažené třídy reprezentující hory, oblohu a lesní porost mají nejlepší výsledky. Třída voda, která má obsažení v datové sadě 3.5%, dosáhla také poměrně vysokých výsledků. Mezi třídami s nižším zastoupením než 3.5% pouze třída ledovec dosahuje vyššího hodnocení, ale zbylé třídy mají výsledky buď velice nízké, nebo model do dané třídy žádná data vůbec neklasifikuje. Mezi výsledky segmentů před a po použití CRF jsou pouze drobné rozdíly, kdy u většiny modelů použitím CRF dochází ke zlepšení hodnocení tříd hory, obloha a voda na úkor ostatních tříd. U modelů DeepLab-MSc-COCO-LargeFOV a DeepLab-MSc-LargeFOV došlo použitím CRF k úplnému odstranění všech tříd reprezentující ledovec a holou skálu.

Experimenty s redukcí tříd

Předchozí experiment ukázal, že některé třídy s malým procentuálním zastoupením dosahují nízkého hodnocení výsledků. Tyto třídy jsme spojili a přidali pod obecnou třídu hory, čímž jsme zmenšili počet tříd, které se model při trénování učí, a experimenty jsme ověřili, zda se díky tomu model naučí lépe rozpoznávat zbývající třídy. Následně jsme u modelů s nejlepšími dosaženými výsledky ověřili, jaký vliv na model má jeho dotrénování na datové sadě rozšířené o výřezy.

Experiment 1 (nav. diagram 3.2) - První krok (nav. diagram 3.2.1)
V prvním experimentu jsme zredukovali třídy holou skálu, kamennou suť, pastviny, útes, oblázky a přírodní prohlubně v terénu, které dosáhly nejnižších výsledků IU. Při experimentu jsme dotrénovali základní model DeepLab a pět rozšířených modelů, kdy trénování probíhalo autory metody přednastavených 6000 iterací. Z dotrénovaných modelů nejlépe

IU (%) bez/s CRF	DeepLab	DeepLab-COCO-LargeFOV	DeepLab-LargeFOV	DeepLab-MSC	DeepLab-MSc-COCO LargeFOV	DeepLab-MSc-LargeFOV
hory	54.4/55.2	53.5/53.8	53.5/54.4	**55.4**/55.9	**55.3**/**56.3**	55.1/56.1
obloha	88.9/**89.5**	**89.2**/89.0	88.9/89.1	87.1/87.3	85.8/86.5	85.7/86.3
lesní por.	38.8/**39.5**	**39.8**/**39.6**	**39.9**/**39.6**	35.9/36.0	31.8/31.6	31.2/30.8
voda	29.7/30.3	34.8/32.0	**38.0**/**38.7**	27.9/28.3	22.0/21.7	20.3/19.7
holá skála	4.4/3.7	5.9/4.7	7.5/6.2	2.4/0.0	0.4/0.0	0.3/0.0
kam. suť	0/0	0/0	0/0	0/0	0/0	0/0
ledovec	32.1/33.1	31.9/29.0	**38.9**/**36.9**	10.3/4.9	2.3/0.0	2.0/0.0
pastviny	0/0	0/0	0/0	0/0	0/0	0/0
útes	0/0	0/0	0/0	0/0	0/0	0/0
oblázky	0/0	0/0	0/0	0/0	0/0	0/0
prohlubně	0/0	0/0	0/0	0/0	0/0	0/0

Tabulka 6.7: Naměřené hodnoty IU na testovací množině datové sady GepPose3K, rozsegmentované pomocí dotrénovaných modelů z metody DeepLab v1. Výsledky IU jsou seřazeny podle procentuálního zastoupení tříd v datové sadě od nejčetnějších po nejméně obsažené

dopadl model DeepLab-LargeFOV, který na malých obrázcích do 400 pixelů dosáhl hodnocení mean IU 50.5% a 47.6% s CRF. Druhého nejlepšího hodnocení dosáhl model DeepLab-COCO-LargeFOV s 49% a 47.4% s CRF, následován základním modelem DeepLab s 46.4% a 46.5% s CRF. Modely DeepLab-MSC, DeepLab-MSc-COCO-LargeFOV, DeepLab-MSc-LargeFOV dosáhly horších výsledků 43.2%, 39.2% a 38.5%, ani použitím CRF nedošlo k jejich vylepšení.

Nejlépe tedy dopadl model DeepLab-LargeFOV, u kterého jsme redukcí z 11 na 5 tříd docílili zvýšení výsledků mean IU z 26% na 50.5% u malých obrázků a z 22.7% na 45.6% u velkých obrázků. Pořadí úspěšnosti jednotlivých tříd se nezměnilo (tabulka 6.8), došlo k vylepšení výsledků třídy hory z 54.4% na 59.2%, což ukazuje, že se model úspěšně naučil klasifikovat spojené třídy.

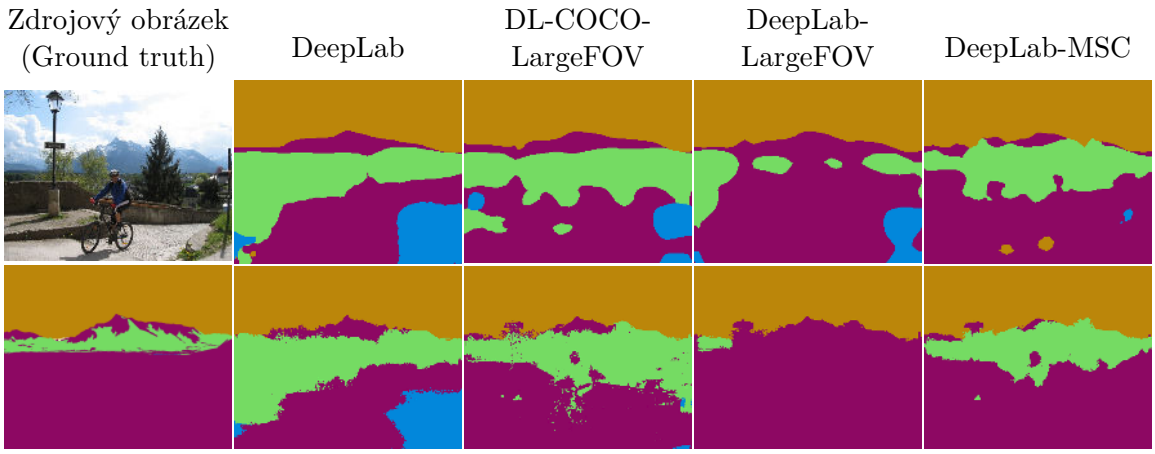
Použití CRF se pozitivně projevilo u tříd hory, obloha a lesní porost, což částečně odpovídá i dosaženým výsledkům použití CRF u všech 11 tříd. Naopak u třídy ledovec použitím CRF došlo k výraznému zhoršení, které se nejvíce projevilo u dotrénovaného modelu DeepLab-LargeFOV, kde IU kleslo z 31.5% na 18.9%.

Dle výsledků mean IU (tabulka 6.8) použití CRF u většiny modelů přináší zhoršení. Změny výsledných segmentů před a po použití CRF jsou patrné i vizuálně (obrázek 6.11). Při porovnání výsledných segmentů lze říci, že použití CRF může vylepšit například vrchol hory, ale zároveň do výsledných segmentů zanese další objekty např. stromy v popředí, cyklisty, nebo pouliční lampy, což jsou prvky, které se model naučil ignorovat a dané pixely klasifikovat jako horskou třídu, která se vyskytuje v pozadí daného objektu.

Experiment 2 (nav. diagram 3.3) - První krok (nav. diagram 3.3.1)
V druhém experimentu jsme navíc redukovali další třídu, která reprezentuje ledovec. Tato třída má v datové sadě zastoupení méně než 1%, přestože dosahuje výrazně vyšších výsledků než ostatní třídy s takto malým obsažením v datové sadě, by její redukce, kterou dále snížíme počet tříd, které se model učí rozpoznávat, mohla zlepšit nejen celkové výsledky, ale i výsledky jednotlivých tříd. V prvním kroku dotrénování jsme zvolili stejných 6 před-trénovaných modelů jako v předchozích experimentech. Při vyhodnocení na ma-

5 tříd bez/s CRF	DeepLab	DeepLab-COCO-LargeFOV	DeepLab-LargeFOV
mean IU (%)	46.4/46.5	49/47.4	**50.5/47.6**
IU (%)			
hory	**59.2/59.8**	55.6/55.8	58.1/58.4
obloha	89.0/89.5	**89.3/89.2**	89.0/89.1
les. porost	37.4/38.0	**39.8/40.2**	38.4/37.8
voda	31.8/32.5	**35.5/33.3**	**35.4/34.0**
ledovec	14.5/12.8	24.6/18.2	**31.5**/18.9

Tabulka 6.8: Výsledky mean IU a IU nejlépe dotrénovaných modelů z metody DeepLab v1 na datové sadě GeoPose3K, která je zredukována na 5 tříd. Výsledky IU jsou seřazeny podle procentuálního zastoupení tříd v datové sadě od nejčetnějších po nejméně obsažené.



Obrázek 6.11: Porovnání výsledků segmentace nejlepšími modely DeepLab dotrénovanými na datové sadě GeoPose3K s 5 třídami před a po použití CRF. Segmenty v prvním řádku jsou před použím CRF a v druhém řádku jsou segmenty po použití CRF.

lých obrázcích došlo ke změně pořadí, kdy se prohodilo pořadí dvou nejlepších modelů. Nejlepší výsledky mean IU 56.8% a 56.7% s CRF u malých, 51.6% a 52.3% s CRF u velkých obrázků dosáhl model DeepLab-COCO-LargeFOV následován 55.5% a 54.9% s CRF u malých, 52.1% a 52.2% s CRF u velkých obrázků modelem DeepLab-LargeFOV. Poté se na třetím místě umístil základní model DeepLab s mean IU 55.2% a 55.2% s CRF. Tato změna pořadí také znamená, že při odebrání třídy ledovec dosáhl lepších výsledků model před-trénovaný na datové sadě MS-COCO. Nejhorších výsledků dosáhly modely s vylepšením MSC, u kterých se pořadí jejich umístění nezměnilo. Výsledky IU pro jednotlivé třídy ukázaly u nejlepšího modelu DeepLab-COCO-LargeFOV zlepšení třídy hory z 55.6% na 61.9%, což ukazuje, že model se úspěšně naučil třídu, která původně reprezentovala ledovec, klasifikovat jako obecnou třídu hory. K mírnému zhoršení výsledků IU došlo pouze u třídy reprezentující lesní porost, a tedy všechny ostatní třídy se díky redukci rozpoznávaných typů tříd podařilo lépe natrénovat.

Experiment 2 - Druhý krok (nav. diagram 3.3.2)
Následně ve druhém kroku experimentu jsme na datové sadě rozšířené o výřezy dotrénovali

nejlepší tři modely z prvního kroku, čímž chceme zmenšit, či úplně odstranit rozdíl mezi výsledky na malých a velkých obrázcích a ověřit, zda trénování na větší datové sadě obsahující více detailů přinese celkově lepší výsledky i na malých obrázcích. Výsledky vyhodnocení dotrénování nejlepších modelů z prvního kroku experimentu (tabulka 6.9) ukázaly, že trénování s výřezy přineslo na malých obrázcích vylepšení pouze u základního modelu DeepLab, který s mean IU 56.4% a 56.9% s CRF dosáhl nejlepšího hodnocení. Druhý se umístil model DeepLab-COCO-LargeFOV, který oproti prvnímu kroku dosáhl horších výsledků 56.8% a 56.7% s CRF. Nejhůře dopadl model DeepLab-LargeFOV. U všech tří modelů došlo ke zlepšení na velkých obrázcích do 800 pixelů, kdy nejlepší výsledky dosáhl model DeepLab, kde se mean IU zlepšilo z 52% a 52.5% s CRF na 55.1% a 55.5% s CRF, což je pouze mírně horší výsledek než který model dosahuje na malých obrázcích.

4 třídy - mean IU (%) bez / s CRF	Krok 1. 400px	Krok 2. 400px	Krok 1. 800px	Krok 2. 800px
DeepLab	55.2/55.2	56.4/**56.9**	52/52.5	**55.1**/**55.5**
DeepLab-COCO-LargeFOV	**56.8**/**56.7**	56/55	51.6/52.3	54.1/53.8
DeepLab-LargeFOV	55.5/54.9	55.2/53.7	52.1/52.2	54.6/54.7

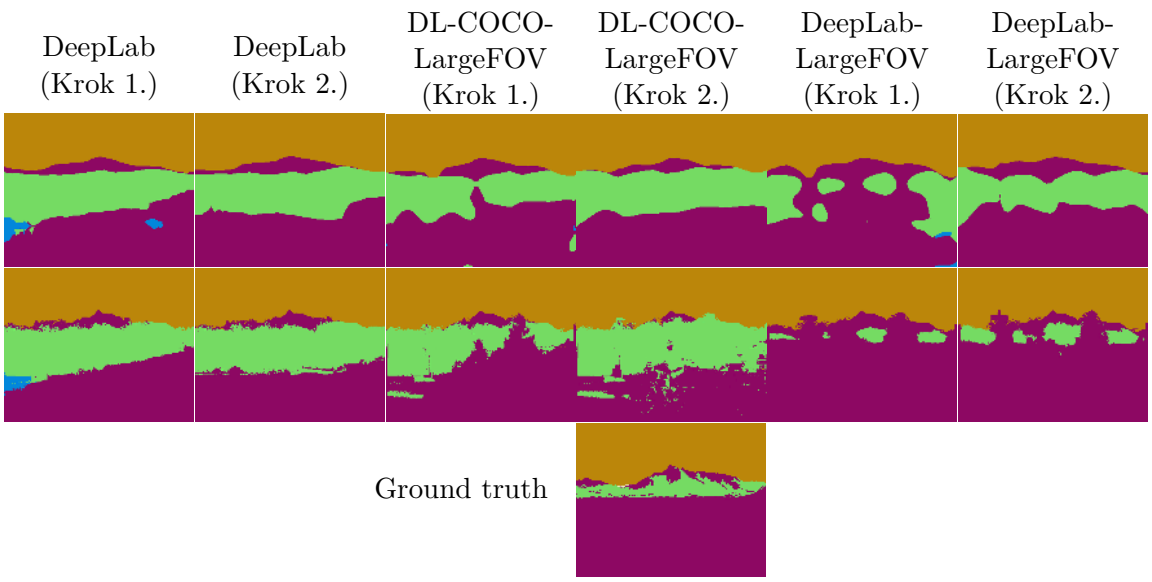
Tabulka 6.9: Výsledky mean IU z pvního a druhého kroku dotrénování tří nejlepších modelů DeepLab na datové sadě GeoPose3K, který obsahuje pouze 4 třídy. V 1. kroku byly základní modely dotrénovány na datové sadě bez výřezů a v 2. kroku byly dotrénovány na datové sadě rozšířené o výřezy. Výsledky obsahují hodnocení mean IU před a po použití CRF.

Experiment 2 - Porovnání kroků

Porovnání výsledků IU (tabulka 6.10) prvního a druhého kroku na segmentech bez použití CRF u všech modelů ukazuje stejné změny, kdy dotrénováním na datové sadě rozšířené o výřezy se výsledky u třídy hory zlepšily, naopak u třídy lesní porost se zhoršily a výsledky třídy obloha zůstaly přibližně stejné. U modelu DeepLab došlo na rozdíl od ostatních modelů ke zlepšení výsledků třídy voda, což vedlo i k celkovému zlepšení tohoto modelu. Rozdíly, které vznikly mezi prvním a druhým krokem dotrénování modelů, jsou patrné i vizuálně na obrázcích, které byly pomocí těchto modelů rozsegmentovány (obrázek 6.12).

4 třídy IU (%) bez/s CRF	DeepLab Krok 1.	DeepLab Krok 2.	DeepLab-COCO-LargeFOV Krok 1.	DeepLab-COCO-LargeFOV Krok 2.	DeepLab-LargeFOV Krok 1.	DeepLab-LargeFOV Krok 2.
hory	61.7/62.1	64.4/64.9	61.9/62.2	63.7/63.4	61.3/61.8	63.4/63.6
obloha	88.6/89.1	89.1/89.6	89.4/89.3	89.5/89.3	88.9/89.0	88.8/88.8
les. porost	36.6/36.9	36.2/36.8	38.4/37.7	38.4/37.5	37.8/36.7	35.5/33.1
voda	33.9/32.8	36.0/36.4	37.6/37.8	32.4/29.7	33.9/32.2	33.2/29.4

Tabulka 6.10: Výsledky IU nejlépe dotrénovaných modelů z metody DeepLab v1 na datové sadě GeoPose3K, která je zredukována na 4 třídy. Výsledky IU jsou seřazeny podle procentuálního zastoupení tříd v datové sadě od nejčetnějších po nejméně obsažené



Obrázek 6.12: Ukázka vizuálních rozdílů mezi segmenty získanými pomocí modelů z prvního a druhého kroku experimentu s datovou sadou GeoPose3K, která obsahuje pouze 4 třídy. V prvním řádku jsou výsledné segmenty bez použití CRF a v druhém řádku segmenty s použitím CRF.

Zhodnocení experimentů s verzí metody DeepLab v1

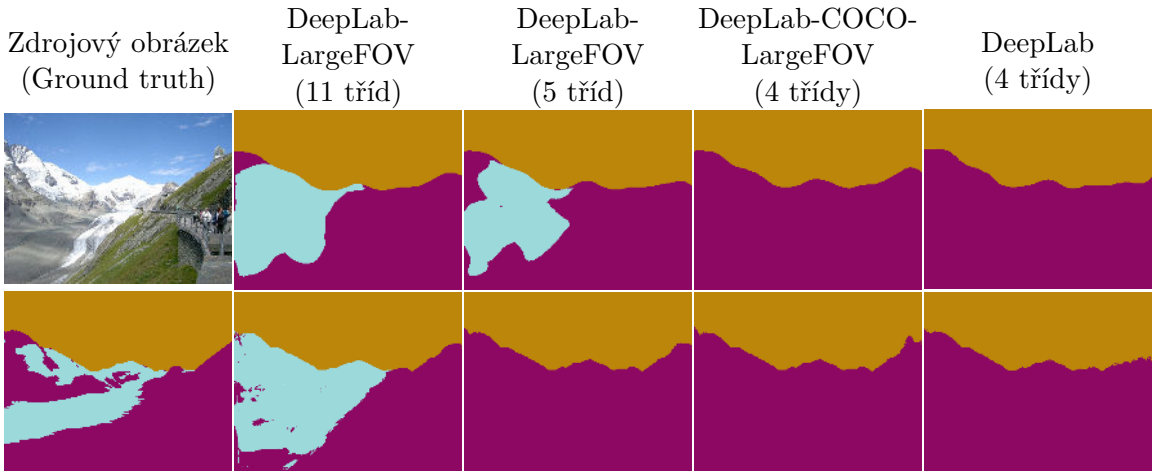
Experimenty se nám nejprve podařilo ověřit jak z průběhu ztrátové funkce při trénování, tak následně z dosažených výsledků testování, že varianty modelu DeepLab v1 jsou schopné dotrénování na horských datech obsažených v datové sadě GeoPose3K. Z hlediska hodnocení pomocí metriky mean IU nejlépe dopadly modely DeepLab-LargeFOV, DeepLab-COCO-LargeFOV a základní model DeepLab. Modely využívající multi-scale funkci (MSC) dopadly nejhůře. Pořadí umístění tří nejlepších modelů se měnilo v závislosti na prováděném experimentu, ale výsledky těchto modelů byly vždy velice podobné.

Výsledky prvních experimentů ukazují, že před-trénované modely dosahují lepších výsledků, kdy nejlépe dopadl model před-trénovaný na datové sadě VOC2012, což potvrzují i první experimenty s redukováním počtu tříd, ale při následné redukci třídy ledovec, která má v datové sadě zastoupení nižší než 1%, již nejlepší výsledky dosahoval před-trénovaný model na datové sadě MS-COCO a výsledky základního modelu a modelu před-trénovaného na datové sadě VOC2012 byly rozdílné pouze o 0.3%. Při dalším dotrénování těchto modelů na výřezech se výsledky na malých obrázcích před-trénovaných modelů zhoršily a základní model DeepLab se jako jediný model dále zlepšoval a tedy v druhém kroku dotrénování dosáhl nejlepších výsledků. Výsledky nám tedy ukazují, že před-trénování i na nehorských datech má pozitivní vliv při trénování na datové sadě obsahující třídy s malým procentuálním zastoupením. Jak ale ukázalo odebrání třídy reprezentující ledovec, stačí pouze jedna taková třída a při jejím odebrání se rozdíly mezi před-trénovanými modely a základním modelem minimalizují. Předpoklad, že před-trénování modelů na datových sadách MS-COCO a VOC2012, které neobsahují horské třídy, nemá významný pozitivní vliv na dosažené výsledky, nebyl tedy potvrzen.

Výsledky IU pro jednotlivé třídy v prvním experimentu potvrdily předpoklad existence vztahu mezi procentuálním obsažením oblastí reprezentujících danou třídu v datové sadě

GeoPose3K a schopností modelu učit se klasifikovat do této třídy. Třídy s vysokým výskytem v datové sadě dosahují nejlepších výsledků a třídy s nízkým zastoupení dosahují horších výsledků, nebo do těchto tříd model neklasifikuje vůbec. Z výsledků ale nelze přesně stanovit například hranice, od které model nemá v datové sadě dostatek informací, aby se naučil danou třídu klasifikovat, nebo určit přímá úměrnost mezi obsažením třídy v datové sadě a jejím hodnocením.

Celkově se nám podařilo dotrénovat na horských datech různě varianty metody DeepLab, kdy při trénování na 11 třídách dopadl model DeepLab-LargeFOV s mean IU 26% a 25.9% s CRF, který s 50.5% a 47.6% s CRF dopadl nejlépe i při trénování na 5 třídách. Při trénování na 4 třídách v prvním kroku dopadl nejlépe model DeepLab-COCO-LargeFOV s 56.8% a 56.7% s CRF a po dotrénování na výřezech v druhém kroku pak základní model DeepLab s hodnocení 56.4% a 56.9% s CRF, kdy hodnota mean IU 56.9% je nejvyšší dosaženou hodnotou při experimentech s verzí metody DeepLab v1. Kompletní naměřené výsledky jsou v příloze C.3. Experimenty s použitím CRF pro vylepšení získaných segmentů v horském prostředí z hlediska metriky mean IU, kromě základního modelu DeepLab, nepřináší zlepšení dosažených výsledků. Vizuální porovnání segmentů před a po použití CRF ukazuje (obrázek 6.13), že sice dochází k jistému zpřesnění segmentů, ovšem dochází také ke zvýraznění nehorských objektů, které se model na trénovacích datech naučil ignorovat a klasifikovat je jako horské třídy, které se nachází v pozadí daného objektu.



Obrázek 6.13: Ukázka segmentů získaných pomocí nejlepších modelů dotrénovaných v jednotlivých experimentech. První sloupec obsahuje segmentovaný obrázek a ground truth, druhý sloupec obsahuje segmenty získané pomocí modelu s nejvyššími výsledky trénování na datové sadě s 11 třídami, třetí sloupec s 5 třídami, čtvrtý sloupec první krok experimentu se 4 třídami a páty sloupec druhý krok experimentu se 4 třídami. V prvním řádku jsou výsledné segmenty bez použití CRF a v druhém řádku segmenty s použitím CRF.

6.3.2 DeepLab v2

Pro metodu DeepLab ve verzi v2 (nav. diagram 4) jsou k dispozici pouze dva modely, respektive dva typy modelů DeepLabv2-VGG16 a DeepLabv2-ResNet101. První typ modelu je založený na síti VGG16 stejně jako všechny modely z verze v1 a druhý typ modelu je založen na síti RESNET101. K oběma typům modelů jsou k dispozici dva modely, jeden je počáteční a druhý je již před-trénovaný na datové sadě VOC2012. U modelu DeepLabv2-

ResNet101 je i počáteční model před-trénován a to na datové sadě MS-COCO. Při experimentech budeme vycházet z poznatků získaných u verze DeepLab v1 a experimenty tedy omezíme pouze na ty s redukovaným počtem tříd. Nejprve provedeme experimenty s 5 třídami, kde ověříme schopnost modelů z metody DeepLab v2 učit se na horských datech a zjistíme vliv použití datové sady rozšířené o výřezy na další dotrénování. Poté dotrénujeme modely před-trénované na datové sadě VOC2012 a porovnáme je s výsledky získanými z dotrénovaných základních modelů.

Následně provedeme další redukci tříd a v prvním experimentu dotrénujeme základní modely na datové sadě se 4 třídami a stejně jako u experimentů s 5 třídami v druhém kroku ověříme vliv výřezů na další dotrénovaní modelů. V druhém experimentu dotrénujeme základní modely přímo v prvním kroku na datové sadě rozšířené o výřezy a výsledky porovnáme s prvním experimentem. Nakonec zhodnotíme přínos redukce tříd, trénování s výřezy a vliv použití před-trénovaných modelů na dosažené výsledky.

Při experimentech trénujme modely stejně jako u verze v1 na malých obrázcích do 400 pixelů, které byly interně modelem ořízntuty. Vyhodnocování modelů ovšem probíhá na obrázcích jiné velikosti, neboť model založený na síti RESNET101 má vyšší požadavky na výpočetní zdroje a největší obrázky, které je možné vyhodnotit jsou o maximální velikosti 385 pixelů. Tedy výsledky dotrénovaného modelu DeepLabv2-ResNet101 jsou vyhodnoceny pouze na malých obrázcích do 385 pixelů. Výsledky dotrénovaného modelu DeepLabv2-VGG16 jsou vyhodnoceny jak na malých obrázcích do 385 pixelů, aby je bylo možné porovnat s výsledky modelu DeepLabv2-ResNet101, tak na malých obrázcích do 400 pixelů a na velkých obrázcích do 800 pixelů, aby výsledky bylo možné porovnat s předchozími experimenty s verzí v1.

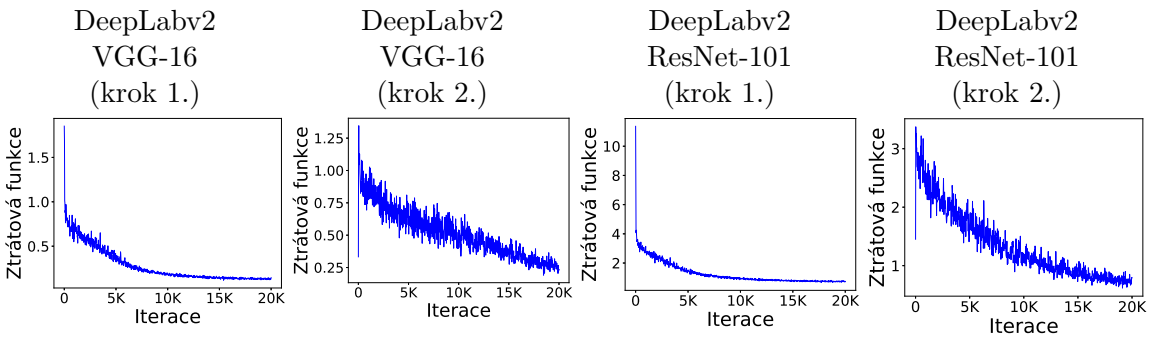
Experimenty s redukcí na 5 tříd

Výsledky experimentů s verzí metody DeepLab v1 ukázaly, že třídy s nízkým procentuálním zastoupením v datové sadě až na třídu ledovec dosahují velice nízkých výsledků. Z těchto výsledků jsme vycházeli v experimentech s verzí metody DeepLab v2.

Experiment 1 (nav. diagram 4.1) - První krok (nav. diagram 4.1.1)
V prvním experimentu jsme základní modely DeepLabv2-ResNet-101 a DeepLabv2-VGG-16 v prvním kroku dotrénovali na datové sadě GeoPose3K, která byla redukována na 5 tříd. Trénování probíhalo u obou modelů standardních 20 000 iterací a průběh ztrátové funkce během trénování (obrázek 6.14) ukazuje schopnost modelů učit se na horských datech, což potvrzují i výsledky testování. Vyhodnocení na testovací sadě ukázalo (tabulka 6.11), že lepších výsledků dosáhl dotrénovaný model DeepLabv2-VGG-16 s mean IU 50% a dotrénovaný model DeepLabv2-ResNet-101 dosáhl horších výsledků 48.7%. Použití CRF na výsledné segmenty ani u jednoho z modelů nepřineslo zlepšení.

Experiment 1 - Druhý krok (nav. diagram 4.1.2)
Vzhledem k průběhu ztrátové funkce ze které je patrná tendence mírného zlepšování modelu i na konci trénování, jsme se v druhém kroku experimentu pokusili dále vylepšit modely a dotrénovali jsme je na datové sadě rozšířené o výřezy. Graf ztrátové funkce při dotrénování modelů vykazuje oproti prvnímu kroku dotrénování změnu průběhu učení, kdy větší množství dat a detailů způsobilo zpomalení rychlosti učení, ovšem jak ukazují výsledky mean IU, tak dotrénování přineslo vylepšení výsledků obou modelů. Model DeepLabv2-VGG-16 dosáhl mean IU 53.1% a také u velkých obrázků do 800 pixelů se výsledky vylepšily ze 46.2% na 50.1% a následným použitím CRF u velkých obrázků došlo k dalšímu mírnému zlepšení na 50.5%. Druhý model DeepLabv2-ResNet-101 dosáhl dotrénováním také zlepšení

dosažených výsledků mean IU na 50.5%. Použití CRF na malých obrázcích ani u jednoho z dotrénovaných modelů nepřineslo další zlepšení výsledků.



Obrázek 6.14: Graf ztrátové funkce při trénování modelů DeepLabv2-VGG-16 a DeepLabv2-ResNet-101 na datové sadě GeoPose3K, která je redukovaná na 5 tříd. Ztrátová funkce v prvním kroku ukazuje trénování na datové sadě bez výřezů a v druhém kroku následné dotrénování modelů na datové sadě rozšířené o výřezy.

5 tříd - mean IU (%) bez / s CRF	Krok 1. 385px	Krok 2. 385px	Krok 1. 800px	Krok 2. 800px
DeepLabv2-VGG-16	50/48.9	**53.1/52.2**	46.2/46.6	**50.1/50.5**
DeepLabv2-ResNet-101	48.7/48	50.5/49.8	-/-	-/-
DeepLabv2-VGG-16 (VOC2012)	49.4/48.5	-/-	43.2/42.8	-/-
DeepLabv2-ResNet-101 (VOC2012)	46.8/46	-/-	-/-	-/-

Tabulka 6.11: Výsledky mean IU z pvního a druhého kroku dotrénování modelů z metody DeepLab v2 na datové sadě GeoPose3K, která je redukovaná na 5 tříd. Výsledky jsou pro základní modely, a pro modely před-trénované na datové sadě VOC2012. V 1. kroku byly základní modely dotrénovány na datové sadě bez výřezů a dotrénované modely byly následně dotrénovány v 2. kroku na datové sadě rozšířené o výřezy. Výsledky obsahují hodnocení mean IU před a po použití CRF.

Experiment 1 - Porovnání tříd

Výsledky IU pro jednotlivé třídy (tabulka 6.12) ukazují, že nejlepších výsledků modely dosahují u třídy obloha, která má v obou modelech podobné výsledky kolem 89%, ovšem v druhém kroku se dotrénováním na datové sadě rozšířené o výřezy již dále nevylepšila. Naopak hodnocení tříd hory, lesní porost a zejména vody se dotrénováním podařilo vylepšit. Třída ledovec dosáhla zlepšení pouze dotrénováním modelu DeepLabv2-VGG-16.

Při vizuálním porovnání výsledků segmentace pomocí dotrénovaných modelů (obrázek 6.15) je na jednotlivých typech segmentů vidět odlišnost modelů, která se projevila v rozdílném hodnocení a to nejen mezi modely DeepLabv2-VGG-16 a DeepLabv2-ResNet-101, ale také mezi prvním a druhým krokem dotrénování. Výrazný vizuální rozdíl je patrný u třídy lesní porost, přestože u obou modelů dosáhla dotrénováním v druhém kroku pouze mírného zlepšení, nebo u třídy hory, kde kromě hranic s lesním porostem se mění i horizont. Vizuálně jsou také patrné rozdíly mezi segmenty před a po použití CRF, které většinou přineslo zlepšení pouze u tříd hory a obloha, ale celkově se na výsledcích projevilo spíše negativně.

IU (%) bez/s CRF 5 tříd	DeepLabv2 VGG-16 Krok 1.	DeepLabv2 ResNet-101 Krok 1.	DeepLabv2 VGG-16 Krok 2.	DeepLabv2 ResNet-101 Krok 2.
hory	59.1/59.7	57.7/58.2	**59.8**/**60.1**	58.9/59.5
obloha	**89.2**/**89.7**	89.1/89.5	88.9/89.2	89.1/89.5
lesní porost	37.2/36.9	33.7/33.3	**37.6**/**37.2**	34.8/34.5
voda	36.3/36.0	36.1/35.4	43.6/43.8	**44.4**/**43.9**
ledovec	28.0/22.4	26.9/23.6	**35.7**/**30.8**	25.3/21.4

Tabulka 6.12: Naměřené hodnoty IU na testovací množině datové sady GepPose3K rozsegmentované pomocí dotrénovaných modelů z metody DeepLab v2. Výsledky IU jsou seřazeny podle procentuálního zastoupení tříd v datové sadě od nejčetnějších po nejméně obsažené.

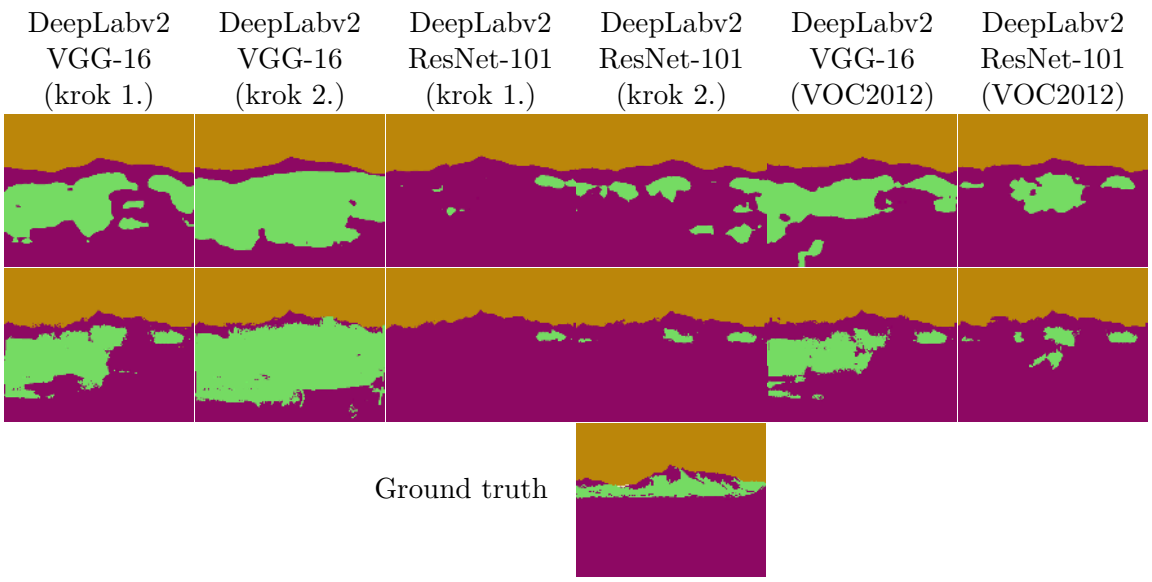
Experiment 2 (nav. diagram 4.2) - První krok (nav. diagram 4.2.1)
Ve druhém experimentu jsme dotrénovali modely DeepLabv2-VGG-16 a DeepLabv2-ResNet-101, které jsou 20000 iterací před-trénované na datové sadě VOC2012. Před-trénované modely jsme dotrénovali na datové sadě GeoPose3K, která je redukovaná pouze na 5 tříd. Dotrénované modely dosáhly horších výsledků mean IU než modely dotrénované v prvním experimentu ze základních modelů (tabulka 6.11). Lépe dopadl model DeepLabv2-VGG-16, který dosáhl mean IU 49.4% a 48.5% s CRF, což je oproti výsledkům dotrénování základního modelu pouze drobné zhoršení, ale při porovnání výsledků u velkých obrázků do 800 pixelů bez CRF došlo k výraznému zhoršení mean IU z 46.2% na 43.2%. Druhý model DeepLabv2-ResNet-101 si oproti základnímu dotrénovanému modelu pohoršil z mean IU 48.7% na 46.8% bez použití CRF. Při porovnání výsledků IU došlo ke zhoršení téměř všech tříd, kdy pouze u dotrénovaného modelu DeepLabv2-VGG-16 třída voda a u druhého modelu DeepLabv2-ResNet-101 třídy hory a lesní porost dosáhly stejných nebo lepších výsledků oproti dotrénovanému základnímu modelu. Rozdíly v dotrénovaných modelech jsou patrné i vizuálně na segmentech (obrázek 6.15).

Experimenty s redukcí na 4 třídy

Při experimentech jsme provedli další redukci tříd, kdy jsme v datové sadě odebrali třídu ledovec, která dosahovala u experimentů s redukcí na 5 tříd nejhorších výsledků a její procentuální zastoupení v datové sadě je menší než 1%.

Experiment 1 (nav. diagram 4.3) - První krok (nav. diagram 4.3.1)
V prvním experimentu jsme na datové sadě se zbylými čtyřmi třídami dotrénovali základní modely DeepLabv2-ResNet-101 a DeepLabv2-VGG-16. Výsledky mean IU (tabulka 6.13) dopadly pro oba modely velice podobně. Lepší výsledek dosáhl dotrénovaný model DeepLabv2-ResNet-101 s 56.1% a 56.3% s CRF, druhý model DeepLabv2-VGG-16 dosáhl výsledků 56% a 56.1% s CRF.

Experiment 1 - Druhý krok (nav. diagram 4.3.2)
Následně jsme modely v druhém kroku dále dotrénovali na datové sadě rozšířené o výřezy, kdy se lépe dotrénoval horší model z prvního kroku DeepLabv2-VGG-16 a dosáhl hodnocení mean IU 57.1% a 57.3% s CRF. Na rozdíl od experimentu s 5 třídami použití CRF přineslo u obou modelů mírné vylepšení získaných výsledků. I přes velice podobné výsledky obou modelů, jsou segmenty získané pomocí dotrénovaných modelů vizuálně odlišné (obrázek 6.16) a to jak před, tak i po použití CRF.



Obrázek 6.15: Ukázka rozdílu mezi segmenty získanými pomocí modelů z metody DeepLab v2 dotrénovaných na datové sadě GeoPose3K, která je redukovaná na 5 tříd. V 1. kroku byly základní modely dotrénovány na datové sadě bez výřezů a v 2. kroku na datové sadě rozšířené o výřezy. Segmenty označeny VOC2012 jsou získany z dotrénování modelů předtrénovaných na datové sadě VOC2012. Segmenty v prvním řádku jsou před použitím CRF a v druhém řádku po použití CRF.

4 třídy - mean IU (%) bez / s CRF	Krok 1. 385px	Krok 2. 385px	Krok 1. 800px	Krok 2. 800px
DeepLabv2-VGG-16	56/56.1	**57.1/57.3**	52.1/52.8	55.7/56.5
DeepLabv2-ResNet-101	56.1/56.3	56.5/57.1	-/-	-/-

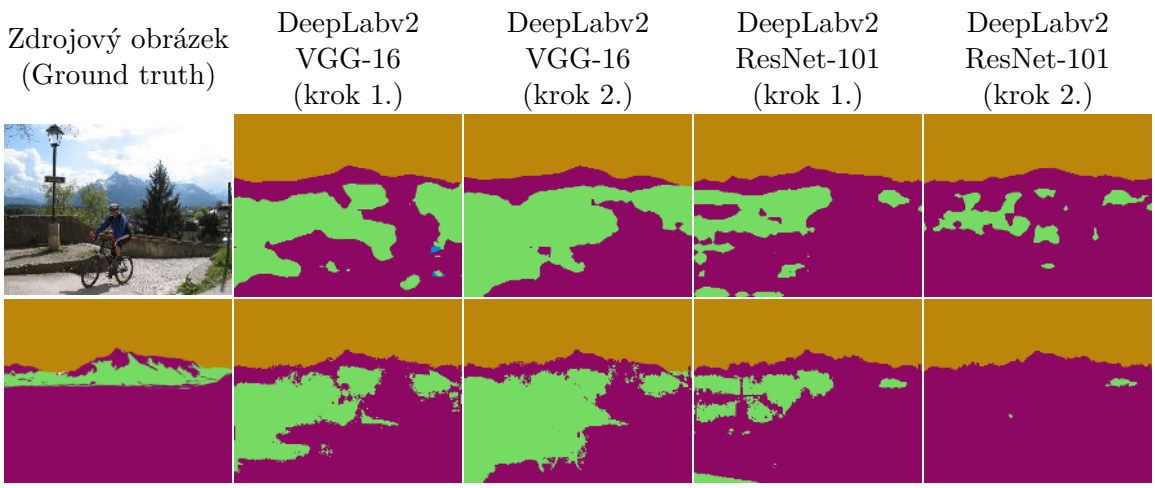
Tabulka 6.13: Výsledky mean IU z pvního a druhého kroku dotrénování modelů z metody DeepLab v2 na datové sadě GeoPose3K, která je redukovaná na 4 třídy. V 1. kroku byly základní modely dotrénovány na datové sadě bez výřezů a dotrénované modely byly následně dotrénovány v 2. kroku na datové sadě rozšířené o výřezy. Výsledky obsahují hodnocení mean IU před a po použití CRF.

Experiment 1 - Porovnání tříd

Při porovnání dosažených výsledků IU pro jednotlivé třídy (tabulka 6.14) se použití CRF pozitivně projevilo u tříd hory a obloha, naopak se negativně projevilo na třídě lesní porost. Porovnání dosažených hodnot IU před a po dotrénování v druhém kroku na datové sadě rozšířené o výřezy ukazuje, že zlepšení modelu DeepLabv2-VGG-16 nastalo v třídách hory, lesní porost a voda. U druhého modelu DeepLabv2-ResNet-101 došlo k mírnému zlepšení u všech tříd.

Experiment 2 - První krok (nav. diagram 4.4)

Druhý krok prvního experimentu ukázal, že při trénování modelů na datové sadě rozšířené o výřezy dle metriky mean IU dochází ke zlepšení. Ve druhém experimentu jsme tedy vyzkoušeli vynechat první krok a přímo jsme základní modely dotrénovali na datové sadě GeoPose3K redukované na 4 třídy, která je rozšířena o výřezy. Průběh ztrátové funkce (obrázek 6.17) ukazuje, že po celou dobu trénování dochází k vylepšování trénovaných



Obrázek 6.16: Ukázka rozdílu mezi segmenty získanými pomocí modelů z metody DeepLab v2 dotrénovaných na datové sadě GeoPose3K, která je redukovaná na 4 třídy. V 1. kroku byly základní modely dotrénovány na datové sadě bez výřezů a dotrénované modely byly následně dotrénovány v 2. kroku na datové sadě rozšířené o výřezy. Segmenty v prvním řádku jsou před použitím CRF a v druhém řádku po použití CRF.

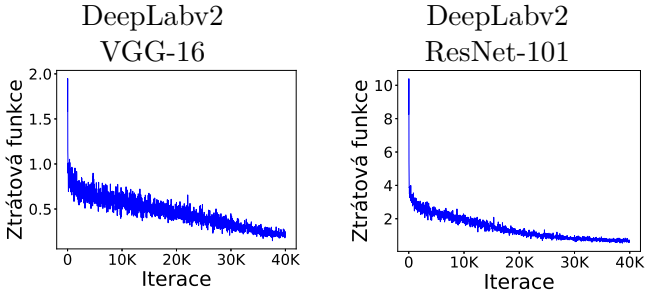
IU (%) bez/s CRF 4 třídy	DeepLabv2 VGG-16 Krok 1.	DeepLabv2 ResNet-101 Krok 1.	DeepLabv2 VGG-16 Krok 2.	DeepLabv2 ResNet-101 Krok 2.
hory	61.9/62.7	61.8/62.3	62.0/62.6	**62.5/63.3**
obloha	**89.0/89.6**	88.9/89.2	88.6/88.8	**89.0/89.4**
lesní porost	35.6/35.3	34.8/34.8	**37.1/36.5**	35.6/35.5
voda	37.5/36.6	38.8/38.8	**40.7/41.1**	38.9/40.0

Tabulka 6.14: Naměřené hodnoty IU na testovací množině datové sady GepPose3K, rozsegmentované pomocí dotrénovaných modelů z metody DeepLab v2. Výsledky IU jsou seřazeny podle procentuálního zastoupení tříd v datové sadě od nejčetnějších po nejméně obsažené.

modelů a proto jsme trénování prodloužili ze standardních 20 000 iterací na 40 000 iterací, kdy ale máme výsledné modely jak pro 20 000 iterací, tak pro konečných 40 000 iterací. Výsledky testování (tabulka 6.15) u obou modelů ukázaly, že i když po celou dobu učení docházelo ke zlepšování, tak na testovací množině lepších výsledků dosáhly modely v půlce učení při 20000 iteracích než na konci učení. Lépe dopadl dotrénovaný model DeepLabv2-VGG-16, který dosáhl mean IU 57.4% a 56.8% s CRF. Druhý model dosáhl mean IU 56.9%, kterého dosáhl i s použitím CRF.

Experiment 2 - Porovnání tříd

V předchozím experimentu výsledky IU jednotlivých tříd byly v prvním kroku u obou modelů přibližně stejné, výsledky IU (tabulka 6.16) druhého experimentu ukazují, že při trénování s výřezy se jednotlivé třídy učí rozdílně v obou modelech. Po 20 000 iteracích, což odpovídá jednomu kroku v předchozím experimentu, je zejména u třídy hory mezi modely velký rozdíl. U modelu DeepLabv2-VGG-16 třídy hory, obloha a voda dosáhly v polovině trénování lepších výsledků něž na konci trénování a naopak třída lesní porost se na konci



Obrázek 6.17: Graf ztrátové funkce při dotrénování základních modelů DeepLabv2-VGG-16 a DeepLabv2-ResNet-101 na datové sadě rozšířené o výřezy.

4 třídy - mean IU (%) bez / s CRF	20 000 385px	40 000 385px	20 000 800px	40 000 800px
DeepLabv2-VGG-16	**57.4/56.8**	56.7/56.5	54.2/54.2	53.7/54.5
DeepLabv2-ResNet-101	56.9/**56.9**	55.9/55.9	-/-	-/-

Tabulka 6.15: Výsledky mean IU dotrénovaných modelů z metody DeepLab v2 na datové sadě GeoPose3K rozšířené o výřezy, která je redukovaná na 4 třídy. Výsledky obsahují hodnocení mean IU před a po použití CRF.

trénování vylepšila. U druhého modelu se třída hory v průběhu trénování vylepšila, ale ostatní třídy dosáhly na konci horších výsledků než v polovině trénování.

IU (%) bez/s CRF 4 třídy	DeepLabv2 VGG-16 20 000	DeepLabv2 ResNet-101 20 000	DeepLabv2 VGG-16 40 000	DeepLabv2 ResNet-101 40 000
hory	**64.3/64.0**	58.6/58.7	61.0/60.8	61.6/61.8
obloha	88.4/88.8	**88.9/89.3**	87.4/87.6	88.6/88.9
lesní porost	36.0/34.9	37.3/37.2	**38.5/37.3**	34.9/34.3
voda	40.9/39.4	**42.8/42.3**	39.9/40.2	38.5/38.7

Tabulka 6.16: Naměřené hodnoty IU na testovací množiny modely DeepLab dotrénovanými pouze na datové sadě rozšířené o výřezy. Výsledky IU jsou seřazeny podle procentuálního zastoupení tříd v datové sadě od nejčetnějších po nejméně obsažené

Zhodnocení experimentů s verzí metody DeepLab v2

Při experimentech jsme nejprve úspěšně ověřili schopnost modelů DeepLabv2-VGG-16 a DeepLabv2-ResNet-101 učit se na horských datech obsažených v datové sadě GeoPose3K. V prvních experimentech s 5 třídami dopadl lépe model DeepLabv2-VGG-16, který po dotrénování v druhém kroku na datové sadě rozšířené o výřezy dosáhl mean IU 53.1% a 52.2% s CRF. Z hlediska jednotlivých tříd dle metriky IU u obou modelů nejlépe dopadla třída obloha, která jak u experimentů se 4, tak s 5 třídami dosahovala hodnoty kolem 88-90%, a poté třída hory, která dosahovala 58-64%. Následovaly třídy lesní porost a voda, které se u obou redukcí tříd dotrénováním na výřezech vždy podařilo dále vylepšit. Nejhůře dopadla třída ledovec, kterou jsme následně při dalších experimentech redukovali, čímž jsme

snížili počet tříd, které se model učí rozpoznávat. Redukcí této třídy, která má méně jak 1% zastoupení v datové sadě se nám podařilo vylepšit výsledky zbylých tříd, a tedy i celkové dosažené výsledky mean IU na 57.4% a 56.8% (57.3%[3]) s CRF na malých obrázcích, které byly dosaženy dotrénováním modelu DeepLabv2-VGG-16. Celkově dopadlo lépe dotrénování modelu DeepLabv2-VGG-16, který u 5 tříd dosahuje výrazně lepších výsledků, ale po redukci na 4 třídy sice dosahuje stále lepších výsledků, ale rozdíl oproti druhému modelu DeepLabv2-ResNet-101 je pouze malý. Kompletní výsledky všech experimentů s druhou verzí metody DeepLab jsou v příloze C.4.

Experimenty jsme ukázali, že modely před-trénované na datové sadě VOC2012, která neobsahuje horské třídy, po dotrénování dosahují horších výsledků než dotrénované základní modely, a tedy jejich použití nepřináší žádné výhody.

Nakonec jsme prokázali výhodu trénování na datové sadě rozšířené o výřezy, kdy u obou modelů jak dotrénování na výřezech v druhém kroku, tak přímé dotrénování základních modelů na výřezech přineslo lepší výsledky, než trénování bez výřezů a to jak na malých, tak na velkých obrázcích.

6.3.3 Zhodnocení trénování DeepLab

Při experimentech s oběma verzemi metody DeepLab jsme u vybraných modelů ukázali, že je lze dotrénovat na horských datech. U obou verzí metody jsme potvrdili vztah mezi procentuálním obsažením tříd v datové sadě a jejími dosaženými výsledky, které jsme porovnávali pomocí metriky IU. Redukcí tříd s nejmenším procentuálním zastoupením a také nejnižšími dosaženými výsledky jsme u obou verzí metody dosáhli lepšího dotrénovaní modelů, které se projevilo jak v celkových výsledcích mean IU, tak ve výsledcích IU pro jednotlivé třídy.

Předpoklad, že před-trénování modelů na datových sadách MS-COCO a VOC2012, které neobsahují horské třídy, nebude mít vliv na dosažené výsledky platil pouze u druhé verze metody DeepLab, kde před-trénované modely dosáhly mírně horšího hodnocení. U první verze metody se ovšem při experimentech s 5 třídami před-trénování pozitivně projevilo ve výsledcích nejlepších metod, ale po redukci na 4 třídy před-trénované modely dosahovaly podobných výsledků jako základní model.

Vliv výřezů na trénování se u první verze metody DeepLab projevil spíše negativně a dotrénováním docházelo kromě základního modelu ke zhoršení výsledků. Nejlepší výsledky dosahovaly modely, které byly dotrénovány pouze na datové sadě bez výřezů. Ovšem u druhé verze metody DeepLab bylo naopak trénování na datové sadě rozšířené o výřezy klíčové při dotrénování nejlepších modelů a přímo dotrénováním základního modelu na datové sadě rozšířené o výřezy jsme dosáhli nejlepších výsledků.

Celkově dopadl na obrázcích do 400 pixelů nejlépe jak u redukce na 5, tak na 4 třídy z hlediska metriky mean IU dotrénovaný model DeepLabv2-VGG-16 s výsledky 52.0% a 51.6% s CRF na 5 třídách a s výsledky 57.2% a 56.6% (57.1% a 57.2% s CRF[4]) na 4 třídách. Druhý nejlepší model na 5 třídách byl s hodnocení mean IU 50.5% a 47.6% s CRF dotrénovaný model DeepLab-LargeFOV. Na 4 třídách skončil druhý model DeepLab-COCO-LargeFOV s hodnocení mean IU 56.8% a 56.7% s CRF, zároveň se základním modelem DeepLab, který dosáhl 56.4% a 56.9% s CRF. Model DeepLabv2-ResNet-101 dosahoval na 4 i 5 třídách velmi podobných výsledků jako modely, které se umístily druhé, ovšem tento model bylo možné vyhodnotit pouze na obrázcích do 385 pixelů, které, jak ukazují výsledky u modelu DeepLabv2-VGG-16, jsou mírně lepší než výsledky na obrázcích do 400 pixelů. Na 11 tří-

[3] Nejlepší hodnota s CRF dosažena u modelu DeepLabv2-VGG-16 s horšími výsledky bez CRF
[4] Druhý dotrénovaný model DeepLabv2-VGG-16 s lepšími výsledky s CRF

dách byly trénovány pouze modely z první verze metody DeepLab. Nejlepšího hodnocení mean IU 26% a 25.9% s CRF dosáhl dotrénovaný model DeepLab-LargeFOV.

Z hlediska mean IU dosahuje 5 nejlepších modelů velice podobných výsledků. Rozdíly mezi modely jsou ale patrné ve výsledcích IU jednotlivých tříd, kde pouze třída obloha dosahuje u redukce jak na 5, tak na 4 třídy výsledků IU přibližně 87-90%. Výsledky ostatních tříd se u jednotlivých modelů a jednotlivých redukcí počtu tříd měnily. Po nejlépe hodnocené třídě obloha následovala třída hory, která dosahovala u redukce na 4 i 5 tříd IU 56-65%. Třídy voda a lesní porost dosahovaly výsledků IU mezi 32-44% a nejhorší hodnocení měla třída ledovec, u které hodnota IU nepřesáhla 36%.

U modelů dotrénovaných na horských datech se z hlediska použitých metrik výrazně neprojevilo ani použití CRF, ani použití novější verze metody DeepLab. Použití CRF se pozitivně projevilo na zpřesnění obrysů jednotlivých oblastí, čímž ale také zvýraznilo nehorské objekty, které se síť úspěšně naučila ignorovat, a to u většiny modelů převážilo a tedy dosáhly s CRF horších výsledků. Metoda DeepLab v2, která přináší model založený na síti RESNET-101, který na referenční datové sadě dosahoval oproti ostatním modelům založeným na síti VGG-16 lepších výsledků, tak na horských datech žádné vylepšení nepřinesl. Novější verze metody DeepLab v2 přinesla o 1.5% lepší výsledky na 5 třídách, ale po redukci na 4 třídy, i když dotrénovaný model DeepLabv2-VGG-16 dosáhl celkově nejlepších výsledků mean IU, tak rozdíl oproti nejlepšímu výsledku metody DeepLab v1 je pouze 0.3%. Ovšem z pohledu délky trénování toto vylepšení o 0.3% je získáno za cenu více jak trojnásobné délky učení, kdy modely z verze DeepLab v1 se dotrénují pouze 6 000 iterací oproti 20 000 iterací u verze DeepLab v2.

6.4 Experimenty s ALE

Framework ALE je již starší nástroj pro sémantickou segmentaci, ale v rámci projektu LOCATE jsou k dispozici parametry pro nastavení frameworku ALE pro sémantickou segmentaci oblohy. Právě obloha je druhou třídou s nejčetnějším procentuálním zastoupením v datové sadě GeoPose3K a nejčastěji má společnou hranici s nejčetnější třídou reprezentující hory, tak předpokládáme, že právě toto nastavení by mohlo u těchto nejčetnějších tříd přinést dobré výsledky. Tento předpoklad ověříme experimenty v kapitole 6.4.1, kdy využijeme poznatků z předchozích experimentů a provedeme trénování na datové sadě s redukovaným počtem tříd. Na závěr získané výsledky shrneme v kapitole 6.4.2.

6.4.1 Experimenty s redukcí tříd

Pro trénování jsme upravili vstupní a výstupní metody ve zdrojovém kódu frameworku tak, aby jej bylo možné trénovat na horských datech obsažených v datové sadě GeoPose3K a výsledné segmenty získané při testování bylo možné ukládat. Při nastavení trénování frameworku ALE jsme použili parametry, které byly přímo s autory frameworku konzultovány pro segmentaci oblohy. Datovou sadu jsme redukovali na 5 tříd reprezentující oblohu, hory, lesní porost, vodu a ledovec.

Experiment 1 (nav. diagram 5.1) - První krok (nav. diagram 5.1.1)
Pomocí takto upraveného frameworku ALE jsme na trénovací množině datové sady GeoPose3K s maximální velikostí obrázku do 400 pixelů natrénovali nový model, který na testovací množině dosáhl hodnocení mean IU 46% na malých obrázcích do 400 pixelů a 41.6% na velkých obrázcích do 800 pixelů. Rozdíl mezi malými a velkými obrázky je 4.4%, což ukazuje, že metoda i přes trénování na malých obrázcích, u kterých došlo zmenšením ke

ztrátě detailů, dosahuje u velkých obrázků, které obsahují větší množství detailů, pouze mírně horších výsledků než na malých obrázcích.

Experiment 1 - Porovnání tříd

Dosažené výsledky IU pro jednotlivé třídy (tabulka 6.17) ukazují, že třída obloha, na kterou je model optimalizován, dosahuje nejlepších výsledků 78.9% a zároveň výsledek 48.8% dosažený u třídy hory potvrzuje náš předpoklad, že tyto dvě třídy dosáhnou nejlepších výsledků. Třetí nejlepší výsledky měla třída ledovec, která má méně jak 1% obsažení v datové sadě GeoPose3K a výsledkem IU 39.5% překonala četnější třídy lesní porost a voda.

5 tříd	400px	800px
mean IU (%)	**46**	41.6
IU (%)		
hory	**48.8**	-
obloha	**78.9**	-
lesní porost	33.1	-
voda	29.7	-
ledovec	**39.5**	-

Tabulka 6.17: Naměřené hodnoty mean IU a IU na testovací množině datové sady GeoPose3K rozsegmentované pomocí natrénovaného modelu ALE. Výsledky IU jsou seřazeny podle procentuálního zastoupení tříd v datové sadě od nejčetnějších po nejméně obsažené.

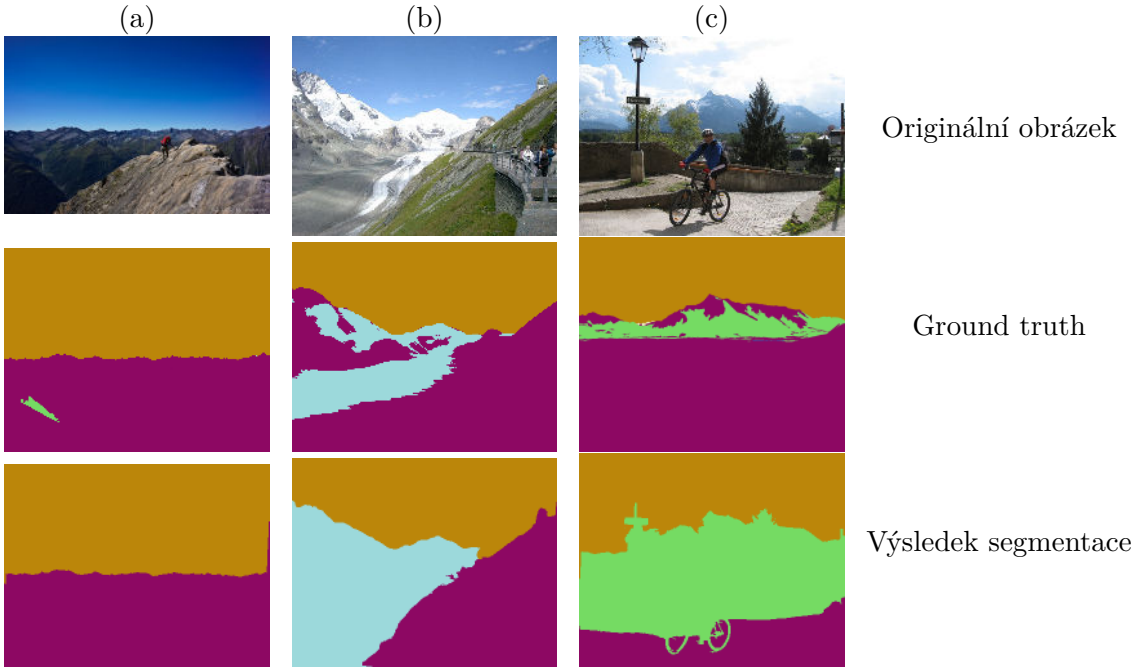
Výsledky jednotlivých tříd jsou patrné i vizuálně ve vzniklých segmentech (obrázek 6.18), kde segmenty reprezentující oblohu jsou zejména v čistě horských fotografiích velice podobné obloze v ground truth, ovšem pokud do oblohy zasahují například sloupy, budovy nebo stromy, pak jsou patrné i v samotných segmentech. Výsledné oblasti reprezentující jednotlivé třídy jsou spíše komplexní, tedy místo více malých oblastí reprezentujících různé třídy je pouze jeden velký segment, který nejčastěji reprezentuje třídu, která v daném místě převažuje.

6.4.2 Zhodnocení trénování ALE

Při experimentu jsme úspěšně pomocí frameworku ALE natrénovali model pro segmentaci do 5 horských tříd, který dosahuje hodnocení mean IU 46%. Zvolené nastavení optimalizované pro segmentaci oblohy se pozitivně projevilo nejen na třídě obloha, která dosáhla nejlepšího výsledku IU 78.9%, ale i na druhé nejlepší třídě hory s výsledkem IU 48.8%. Experiment ukázal velmi dobré výsledky pro třídu ledovec, která má z trénovaných horských tříd nejmenší procentuální zastoupení v datové sadě. Vzhledem k dosaženým výsledkům třídy reprezentující ledovec nejsou další vhodné třídy k redukci, které by měly malé procentuální zastoupení v datové sadě a zároveň dosahovaly špatných výsledků, a tedy jsme neprovedli další experimenty s redukcí tříd. Kompletní naměřené výsledky jsou v příloze C.5.

6.5 Vyhodnocení nejlepších modelů

Při vyhodnocení experimentů nejprve v kapitole 6.5.1 shrneme nejlepší výsledky, kterých jsme dotrénováním dosáhli u jednotlivých metod, a vybereme z nich množinu vhodných modelů. Testovací množinu datové sady GeoPose3K rozsegmentovanou pomocí vybraných



Obrázek 6.18: Výsledky segmentace natrénovaným modelem pomocí frameworku ALE. První řádek obsahuje originální obrázeky, druhý řádek ground truth a třetí řádek výsledek segmentace. Obrázek (a) - ukázka čistě přirodní scény s nejčetnějšími třídami obloha a hory. Obrázek (b) - ukázka spojení více prolínajících se oblastí. Obrázek (c) - ukázka promítnutí nehorských objektů (kolo, lampa,...) do hranic segmentů.

modelů nejprve vyhodnotíme vizuálně a následně použijeme pro určení orientace kamery. Pro vizuální vyhodnocení segmentů sestavíme v kapitole 6.5.2 dotazník, který poskytneme respondentům k vyplnění a zhodnotíme dosažené výsledky. K určení orientace kamery v kapitole 6.5.3 využijeme metody z projektu LOCATE, pomocí kterých určíme orientaci fotoaparátu, kterým byla fotografie pořízena a porovnáme ji se skutečnou orientací fotoaparátu.

6.5.1 Výběr modelů pro další vyhodnocení

Při výběru vhodných modelů pro další vyhodnocení nelze vybírat jen podle nejlepších výsledků (tabulka 6.18), ale musíme zohlednit i přínos informací, které nám další vyhodnocení přinese. Při vyhodnocení chceme nejen porovnat nejlepší modely a modifikace v rámci metod, ale také vzájemně metody, vliv redukce třídy reprezentující ledovec a také trénování na datové sadě rozšířené o výřezy. Původně jsme vybrali 12 dotrénovaných modelů, ale poté jsme jejich počet na základě zpětné vazby k prvním dotazníkům, kdy respondenti měli problém s výběrem i zobrazením velkého počtu možností, zmenšili pouze na 8.

Z metody FCN jsme vybrali dotrénovaný model siftflow-fcn8s na 4 třídách jak s původním, tak s novým klasifikátorem, které dosahují v rámci metody FCN nejlepšího hodnocení. Oba modely byly dotrénovány na výřezech a jejich porovnání nám navíc poskytne informace o vlivu změny klasifikátoru. Dále jsme zvolili z metody DeepLab v1 model DeepLab-COCO-LargeFOV dotrénovaný na 4 třídách, který na rozdíl od nejlepších modelů z FCN a DeepLab v2 nebyl trénován na datové sadě rozšířené o výřezy, přesto dosahuje v rámci první verze metody nejlepších výsledků. U tohoto modelu navíc získáme informaci o vlivu použití CRF,

které z hlediska mean IU nepřineslo zlepšení. Z druhé verze metody DeepLab jsme vybrali dva modely DeepLabv2-VGG16 dotrénované na 4 třídách, které nám poskytnou informaci jak o vlivu CRF, tak o rozdílu mezi přímým dotrénováním na výřezech a dotrénováním na výřezech až v druhém kroku. Z této metody jsme také vybrali model DeepLabv2-VGG16 dotrénovaný na 5 třídách, který na nich zároveň dosahuje nejlepších výsledků a kromě porovnání s dotrénovaným modelem na 4 třídách přinese srovnání s posledním vybraným modelem dotrénovaným na 5 třídách pomocí frameworku ALE.

mean IU (%)	11 tříd	5 tříd	4 třídy
ALE	-	46	-
pascalcontext-fcn8s (původní klas.)	-	20.9	-
siftflow-fcn8s (původní klas.)	-	-	**57.4**
pascalcontext-fcn8s (vlastní klas.)	17.1	31.6	35.6
siftflow-fcn8s (vlastní klas.)	23.7	46.6	56.8 (56.4)
DeepLab	24.8/25.1	46.4/46.5	56.4/56.9
DeepLab-COCO-LargeFOV	25.5/24.8	49/47.4	56.8/56.7
DeepLab-LargeFOV	**26.7**/**26.5**	**50.5**/**47.6**	55.5/54.9
DeepLabv2-VGG16	-	**52**/**51.6**	**57.1**/**57.2**, **57.2**/55.6
	-	(53.1/52.2)	(57.1/57.3), (57.4/56.8)
DeepLab-v2-ResNet101	-	(50.5/49.8)	(56.5/57.1), (56.9/56.9)

Tabulka 6.18: Nejlepší dosažené výsledky mean IU na modelech, které jsou dotrénovány na datové sadě GeoPose3K redukované na 11, 5 a 4 třídy. Naměřené hodnoty jsou pro obrázky do maximalní velikosti 400 pixelů a hodnoty uvedené v závorkách jsou pro obrázky do 385 pixelů. U modelů DeepLab jsou uvedeny hodnoty bez/s CRF.

6.5.2 Vizuální vyhodnocení modelů

Při vizuálním hodnocení (nav. diagram 6) prováděném člověkem respondenti subjektivně hodnotí modely na základě porovnání rozsegmentovaného a původního obrázku. Předpokládáme, že všechny modely dosahují při vizuálním porovnání stejně dobrých výsledků a rozdíly mezi metodami a jednotlivými modely nebudou mít na celkové hodnocení vliv. Pro ověření této hypotézy jsme sestavili elektronický dotazník v podobě webových stránek. Dotazník obsahuje horskou fotografii a k ní sérii obrázků se segmenty rozsegmentovaných pomocí 8 vybraných modelů, ze kterých respondenti vybírají ty, které nejlépe odpovídají původní fotografii. Každý respondent dostane k ohodnocení 15 horských fotografií, ke kterým vybírá podle něj jeden nejlépe odpovídající rozsegmentovaný obrázek. Horské fotografie se náhodně vybírají ze 490 obrázků obsažených v testovací množině datové sady GeoPose3K. Kromě samotného výběru dotazník obsahuje i úvodní stranu s informacemi o autorovi a cíli dotazníku spolu s návodem k jeho vyplnění, který je doplněn o grafickou ilustraci ukazující proces segmentace spolu s ukázkou příkladu odpovídajících a neodpovídajících segmentů. Úvodní stránka spolu s ukázkou výběrové otázky z dotazníku je vyobrazena v příloze D.

Celkové výsledky

Dotazník vyplnilo 120 respondentů, kteří k 15 obrázkům vybrali nejlépe odpovídající segmenty. Celkové výsledky vyhodnocení dotazníku (tabulka 6.19) ukazují, že respondenti nejčastěji u 21.2% obrázků zvolili segmenty získané pomocí modelu DeepLab-COCO-LargeFOV

dotrénovaného na 4 třídách s následným vylepšením pomocí CRF. Poté volili z 18.3% segmenty získané pomocí frameworku ALE, které oproti nejčetněji voleným segmentům z modelu DeepLab-COCO-LargeFOV obsahují navíc oblasti reprezentující ledovec. Třetí nejčastěji volené segmenty z 14.5%, které také obsahují oblasti reprezentující ledovec, jsou získané pomocí modelu DeepLabv2-VGG16 a následně vylepšené pomocí CRF. Segmenty získané verzí tohoto modelu dotrénovaného pouze na 4 třídách byly následujícími nejčetněji volenými. Následovaly segmenty z obou verzí modelu siftflow-fcn8s, kdy častěji byly voleny segmenty získané pomocí modelu s vlastním klasifikátorem a nejméně často byly voleny segmenty získané z modelů metody DeepLab, které nebyly vylepšeny pomocí CRF.

Výsledky ukazují významný vliv použití CRF na vizuální hodnocení, kdy nejčetněji vybírané segmenty byly buď vylepšeny pomocí CRF dodatečně (modely z DeepLab), nebo CRF bylo použito interně přímo u segmentace (framework ALE). Rozdíl je vidět zejména u modelu DeepLab-COCO-LargeFOV, kde segmenty bez použití CRF byly druhé nejméně často vybírané, ale po aplikaci CRF se staly vůbec nejčastěji volenými. Při porovnání modelů DeepLab-COCO-LargeFOV a DeepLabv2-VGG16 tak první model, který nebyl trénován na datové sadě rozšířené o výřezy, dopadl jak s CRF, tak bez CRF lépe jak druhý model, který, aby dosáhl nejlepších výsledků mean IU, byl vždy dotrénován na datové sadě rozšířené o výřezy.

Vyhodnocení zvlášť obrázků, které podle ground truth obsahují ledovec, a zbytku obrázků (tabulka 6.19) ukazuje, že modely, jejíž segmenty obsahují oblasti reprezentující třídu ledovec, byly častěji voleny u obrázků, které skutečně obsahují ledovec než u ostatních obrázků. Nejčastěji z 24.3% byly voleny segmenty vzniklé pomocí frameworku ALE, kdy ovšem ledovec průměrně pokrýval pouze 3.5% z plochy obrázku. U verze modelu DeepLabv2-VGG16, která segmentuje ledovec, byly obrázky, které skutečně obsahují ledovec, vybírány ve 14.2% a ledovec na těchto obrázcích průměrně pokrývá 5% jejich plochy.

Obdobně i modely, které nesegmentují třídu ledovec, byly častěji voleny u obrázků bez ledovce než u obrázků s ledovcem. Obrázky bez ledovce byly z 22.9% nejčastěji rozsegmentovány pomocí modelu DeepLab-COCO-LargeFOV a následně vylepšeny pomocí CRF, ovšem i obrázky rozsegmentované tímto modelem, kde originální obrázek obsahuje ledovec, byly voleny z 18.3%, tedy častěji než u horšího z modelů, jehož segmenty třídu ledovec skutečně obsahují.

Vizuální vyhodnocení (%)	bez ledovce	s ledovcem	celkové
ALE	14.4	**24.3**(3.5)	18.3
siftflow-fcn8s (původní klas.)	9.9	8.5(2.3)	9.2
siftflow-fcn8s (vlastní klas.)	12.2	8.4(1.7)	10.5
DeepLab-COCO-LargeFOV	8.2	6.2(2.2)	7.6
DeepLab-COCO-LargeFOV (CRF)	**22.9**	18.3(1.9)	**21.2**
DeepLabv2-VGG16 (5 tříd)	9.4	14.2(5)	11.5
DeepLabv2-VGG16	8.1	5.7(3.7)	7.2
DeepLabv2-VGG16 (CRF)	15.0	14.2(2.7)	14.5

Tabulka 6.19: Výsledky vizuálního hodnocení modelů pomocí dotazníků. První sloupec obsahuje výsledky pro obrázky, které neobsahují ledovec, druhý sloupec pro obrázky, které obsahují ledovec, kde v závorkách je uvedena průměrná plocha, kterou zabírá ledovec na obrázcích, které byly vybírány u daného modelu. Poslední sloupec obsahuje celkové výsledky na všech obrázcích.

Výsledky podle tříd v hodnocených obrázcích

Kromě celkového hodnocení jsme zvlášť vyhodnotili výsledky pro různé kombinace tříd (tabulka 6.20), kdy jsme rozdělení obrázků do kombinací provedli podle typů oblastí v ground truth. Vyhodnocení neobsahuje všechny kombinace tříd, ale pouze 7 nejčastějších kombinací. Do těchto kombinací nespadá z celkových 490 pouze 15 obrázků, které obsahují specifické kombinace tříd a vzhledem k malému počtu obrázků v těchto specifických kombinacích nepřináší objektivní výsledky. Výsledky pro jednotlivé kombinace ukazují, že u všech kombinací byly nejčastěji vybírány segmenty získané stejně jako u celkových výsledků z modelu DeepLab-COCO-LargeFOV klasifikujícího do 4 tříd s následným vylepšením pomocí CRF a z frameworku ALE. U modelu DeepLab-COCO-LargeFOV byly nejčastěji vybírány segmenty obsahující oblasti reprezentující třídu lesní porost, a to ve všech kombinacích s ostatními třídami včetně třídy ledovec, do které model oblasti neklasifikuje. Druhé nejčastěji vybírané segmenty u těchto kombinací tříd byly získané frameworkem ALE a modelem DeepLabv2-VGG16 dotrénovaným na 4 třídách s následným použitím CRF, kdy četnost výběru segmentů pro DeepLab i ALE byla přibližně stejná. Naopak segmenty, které neobsahují třídu lesní porost, tedy různé kombinace tříd hory, obloha, voda a ledovec, byly nejčastěji vybírány z frameworku ALE.

Při porovnání obou modelů siftflow-fcn8 byly u většiny kombinací vybírány segmenty vzniklé z modelu s vlastním klasifikátorem, kdy pouze u kombinace všech pěti tříd byly častěji vybírány segmenty z modelu s původním klasifikátorem.

Nejméně byly vybírány segmenty z variant modelů DeepLab bez CRF, kde pouze segmenty obsahující oblasti reprezentující třídy hory, obloha, voda a hory, obloha, ledovec z modelu DeepLabv2-VGG16, který segmentuje do 5 tříd, byly vybírány četněji, ovšem v dalších kombinacích segmentace třídy ledovec nepřinesla oproti ostatním metodám výrazné zlepšení.

Vizuální vyhodnocení (%)	HOL	HOV	HOD	HOLV	HOLD	HOVD	HOLVD
Počet obrázků	152	9	71	104	60	14	65
ALE	14.0	**29.4**	**35.5**	14.0	18.8	**30.6**	14.9
siftflow-fcn8s (pův. klas.)	10.3	5.9	3.5	9.6	9.4	8.2	13.9
siftflow-fcn8s (vl. klas.)	11.4	8.8	4.7	13.2	10.3	14.3	9.6
DL-COCO-LargeFOV	7.7	5.9	3.5	7.5	4.5	4.1	12.0
DL-COCO-LFOV (CRF)	**25.6**	17.6	16.4	**20.7**	**22.3**	12.2	**17.8**
DL-v2-VGG16 (5 tříd)	8.7	14.7	23.4	9.6	9.4	10.2	9.1
DL-v2-VGG16	8.1	2.9	2.7	8.8	8.0	6.1	6.7
DL-v2-VGG16 (CRF)	14.0	14.7	10.2	16.6	17.4	14.3	15.9

Tabulka 6.20: Výsledky vizuálního hodnocení pro skupiny obrázků rozdělených podle typů oblastí, které dle ground truth obsahují. Kombinace tříd je popsána pomocí zkratek názvů tříd (H - **H**ory, O - **O**bloha, V - **V**oda, D - le**D**ovec, L - **L**esní porost).

Zhodnocení vizuálního vyhodnocení modelů

Výsledky vizuálního vyhodnocení vyvrátily stanovený předpoklad, že všechny modely dosáhnou při vizuálním porovnání stejně dobrých výsledků. Segmenty vzniklé s použitím CRF byly vybírány výrazně častěji než segmenty bez CRF, kdy celkově nejčastěji byly vybírány segmenty získané modelem DeepLab-COCO-LargeFOV dotrénovaným na 4 třídách a poté

z frameworku ALE, ze kterého byly celkově vybírány méně často, ale obsahují i třídu ledovec. Na obrázcích, ve kterých se ledovec skutečně vyskytuje byly segmenty z této metody vybírány nejčastěji. Zároveň oba tyto modely byly trénovány pouze na datové sadě bez výřezů. Při porovnání metod bez CRF byly celkově nejčastěji vybírány segmenty z modelu DeepLabv2-VGG16, který byl dotrénován na 5 třídách, kdy segmenty bez oblastí reprezentujících třídu ledovec byly vybírány přibližně stejně často jako u ostatních modelů bez CRF, ale segmenty s oblastmi, které reprezentují třídu ledovec, byly vybírány výrazně častěji.

6.5.3 Vyhodnocení modelů pomocí orientace kamery

Při vyhodnocení pomocí určení orientace kamery (nav. diagram 7) použijeme vybrané modely v procesu, kdy se pomocí nich nejprve rozsegmentuje fotografie, u které známe GPS souřadnice. Následně se použije 3D model povrchu země obsahující horské segmenty, kdy se panoramatický pohled z bodu určeného GPS pozicí namapuje na povrch koule a poté se na ní pomocí sférické vzájemné korelace hledají segmenty odpovídající segmentům z fotografie a pomocí nejlepší shody se určí orientace kamery. Pro určení orientace provádíme dva experimenty, které se liší v rozlišení sférické funkce, kdy jeden experiment pracuje s nízkým a druhý s vysokým rozlišením. Získanou orientaci porovnáme se skutečnou orientací kamery a vypočteme chybu orientace. Tuto chybu vypočteme pro všechny fotografie z testovací množiny datové sady GeoPose3K rozsegmentované pomocí vybraných dotrénovaných modelů a použijeme ji pro porovnání těchto modelů.

Chyba orientace kamery [39] je definována následovně, nechť R_{gt} je matice rotace ground truth a R_c je odhadnutá matice rotace, pak je chyba orientace definována jako:

- $e(R_{gt}, R_c) = arccos(\frac{tr[R_{gt}^T R_c]-1}{2})$

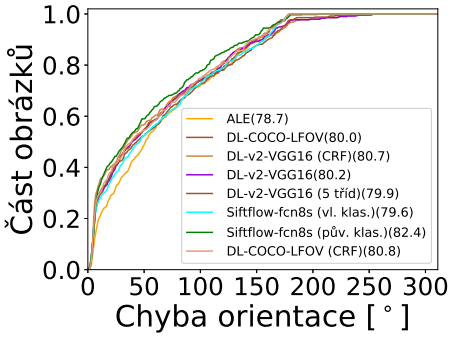
Celkové výsledky

Vyhodnocením rozsegmentovaných obrázků pomocí vybraných modelů jsme získali chybu orientace kamery pro nízké rozlišení a vysoké rozlišení. Výsledky pro obě rozlišení jsou vizualizovány pomocí grafů v obrázku 6.19. Z grafů je patrné, že rozdíly mezi segmenty z jednotlivých modelů jak u malého, tak u velkého rozlišení jsou pouze malé. U malého rozlišení nejlépe dopadly segmenty z metody siftflow-fcn8s s původním klasifikátorem, kde plocha pod křivkou grafu (AUC[5]), který vizualizuje výsledky, dosahuje 82.4% z celkové plochy. U velkého rozlišení nejlépe dopadly s AUC 83.3% segmenty z modelu DeepLabv2-VGG16 s použitím CRF a dotrénovaném na 4 třídách. Porovnání výsledků malého a velkého rozlišení ukazuje, že při použití většího rozlišení se vylepšují dosažené výsledky jednotlivých modelů, kdy pouze nejlepší segmenty u malého rozlišení z modelu siftflow-fcn8s dosáhly u velkého rozlišení horší výsledky.

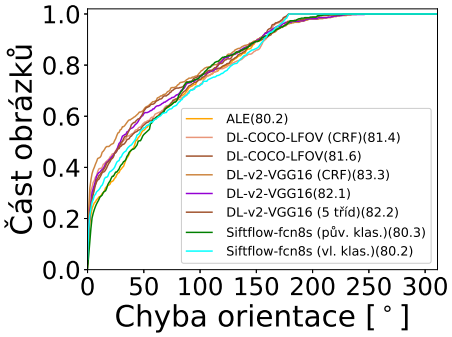
Kromě použití čistě segmentů pro odhad orientace jsme testovali i rozšíření, které kombinovalo segmenty s hranami, což se pozitivně projevilo na výsledcích. Při rozšíření dochází k vylepšení získaného odhadu orientace pomocí porovnání hran v terénním modelu s hranami detekovanými ve vstupní fotografii. Použití rozšíření se projevilo zejména u výsledků velkého rozlišení (tab. 6.21), kdy segmenty z modelu DeepLabv2-VGG16 dotrénovaného na 5 třídách dosáhly nejvyšší hodnocení AUC, které vzrostlo z 82.2% na 86.9%. Ovšem u segmentů z modelu DeepLabv2-VGG16 dotrénovaném pouze na 4 segmentech, které dosahují bez vylepšení téměř stejný výsledek AUC 82.1%, došlo použitím rozšíření k výrazně menšímu zlepšení pouze na 84.6%. Rozdíly ve výsledcích bez a s rozšířením ukazují, že segmenty

[5] z angl. Area under the curve

(a) Nízké rozlišení



(b) Vysoké rozlišení



Obrázek 6.19: Porovnání chyby odhadu orientace kamery pro rozsegmentovanou testovací množinu datové sady GeoPose3K pomocí jednotlivých modelů. Levý graf zobrazuje výsledky pro nízké rozlišení a pravý graf pro vysoké rozlišení. U jednotlivých modelů je v závorce uvedena plocha pod křivkou (AUC) v %.

získané pomocí různých modelů, přestože bez vylepšení dosahují podobných výsledků, se v kombinaci s určitým rozšířením mohou chovat rozdílně, a tedy dosahovat různých výsledků. Nárůst AUC se kombinací s hranami projevil i u malého rozlišení, došlo ale pouze k drobným zlepšením, kdy se nejvíce zlepšily segmenty z modelů klasifikujících do třídy ledovec. Kompletní výsledky pro malé i pro velké rozlišení jsou v příloze E.

U výsledků s vysokým rozlišením jsme vypočetli průměrnou chybu orientace jak pro všechny segmenty, tak zvlášť pro rozsegmentované obrázky, jejichž ground truth obsahuje třídu ledovec, a také naopak pro rozsegmentované obrázky bez ledovce (tabulka 6.21). Všechny verze modelů DeepLabv2-VGG16 a to i včetně verze, která do třídy ledovec klasifikuje, měly menší průměrnou chybu orientace na rozsegmentovaných obrázcích, které ledovec neobsahují, kdy nejlépe dopadly segmenty z modelu dotrénovaném na 4 třídách a vylepšené pomocí CRF. Ostatní modely dosáhly menší průměrně chyby orientace na rozsegmentovaných obrázcích obsahujících ledovec, kdy nejlépe dopadly segmenty z verze modelu DeepLab-COCO-LargeFOV s CRF, který do třídy ledovec neklasifikuje.

Chyba orientace (°) (Vysoké rozlišení)	bez ledovce	s ledovcem	celkové	AUC (%)	AUC-EDG (%)
ALE	67.4	54.1	61.7	80.2	82.7
siftflow-fcn8s (pův. klas.)	61.8	60.8	61.4	80.3	84.6
siftflow-fcn8s (vl. klas.)	67.6	53.6	61.6	80.2	82.0
DL-COCO-LargeFOV	64.2	48.4	57.4	81.6	-
DL-COCO-LFOV (CRF)	67.7	**45.1**	58.0	81.4	-
DL-v2-VGG16 (5 tříd)	55.1	55.7	55.4	82.2	**86.9**
DL-v2-VGG16	54.2	58.1	55.9	82.1	84.6
DL-v2-VGG16 (CRF)	**50.6**	53.8	**51.9**	**83.3**	85.1

Tabulka 6.21: Průměrná chyba orientace kamery a hodnoty AUC. První sloupec obsahuje výsledky pro segmenty, které dle ground truth neobsahují ledovec, druhý pro segmenty obsahující ledovec, třetí sloupec obsahuje celkové výsledky pro segmenty s i bez ledovce, čtvrtý sloupec obsahuje hodnoty AUC a poslední sloupec hodnoty AUC při kombinaci segmentů s hranami.

Výsledky podle tříd v hodnocených obrázcích

Obdobně jako u vyhodnocení dotazníku jsme výsledky chyby orientace kamery s vysokým rozlišením vyhodnotili zvlášť pro různé kombinace tříd (tabulka 6.22), kdy jsme datovou sadu rozdělili dle kombinací uvedených v tabulce 6.22. Výsledky jednotlivých kombinací se liší jak pro různé kombinace tříd v rámci jednotlivých metod, tak i v rámci jednotlivých kombinací tříd mezi metodami. Segmenty z modelů metody DeepLab, které mají nejmenší celkovou průměrnou chybu orientace, dosahovaly nejlepších výsledků u kombinace tříd hory, obloha, voda a ledovec, na kterých nejlépe dopadly segmenty z modelu DeepLab-COCO-LargeFOV s CRF, u kterých byla průměrná chyba orientace pouze 10.3°. Největší průměrná chyba orientace 90.3° byla u kombinace tříd hory, obloha a voda u segmentů z modelu siftflow-fcn8s s původním klasifikátorem.

Při porovnání segmentů z modelů siftflow-fcn8s s původním a s vlastním klasifikátorem segmenty z modelu s vlastním klasifikátorem dopadly lépe v kombinacích obsahujících vodu a/nebo ledovec, ale v kombinacích obsahujících lesní porost měly segmenty většinou horší průměrnou chybu orientace. U segmentů z modelu DeepLab-COCO-LargeFOV s použitím CRF docházelo k menší chybě orientace u kombinace tříd hory, obloha, voda a/nebo ledovec. U ostatních kombinací tříd měly menší chybu orientace segmenty z verze modelu bez CRF. Naopak u segmentů z modelu DeepLabv2-VGG16 přineslo použití CRF oproti segmentů bez CRF zlepšení u všech kombinací tříd které obsahují lesní porost. Při porovnání segmentů z modelů DeepLabv2-VGG16 dotrénovaných na 4 a na 5 třídách se u kombinace tříd hory, obloha a ledovec na chybu orientace vliv klasifikace do třídy ledovec téměř neprojevil. Segmenty z verze modelu klasifikujícího ledovec měly průměrnou chybu orientace pouze o 0.2° menší než segmenty z druhého modelu a u ostatních kombinací tříd, ve kterých se vyskytuje ledovec, byla průměrná chyba orientace pouze mírně menší. U segmentů z metody ALE byla nejmenší chyba orientace u kombinace tříd hory, obloha a ledovec. Výhoda klasifikace do třídy ledovec se ale u ostatních kombinací tříd na zmenšení chyby orientace neprojevila.

Chyba orientace (°)	HOL	HOV	HOD	HOLV	HOLD	HOVD	HOLVD
Počet obrázků	152	9	71	104	60	14	65
ALE	70.4	45.6	35.8	60.2	60.5	62.0	66.6
siftflow-fcn8s (pův. klas.)	57.7	90.3	73.6	63.2	51.7	59.5	55.5
siftflow-fcn8s (vl. klas.)	75.9	60.6	43.5	56.1	55.3	44.4	65.2
DL-COCO-LargeFOV	70.6	40.9	41.6	**55.4**	**46.3**	26.1	62.6
DL-COCO-LFOV (CRF)	71.0	39.9	**32.6**	65.0	47.2	**10.3**	64.1
DL-v2-VGG16 (5 tříd)	54.7	54.8	50.0	56.9	61.6	20.4	63.9
DL-v2-VGG16	51.2	**27.6**	50.2	63.6	65.3	22.9	67.7
DL-v2-VGG16 (CRF)	**47.7**	42.4	53.1	57.6	60.2	23.0	**55.2**
Průměr	62.4	50.3	47.5	59.7	56.0	33.6	62.6

Tabulka 6.22: Průměrná chyba orientace kamery pro skupiny segmentů rozdělených podle typů oblastí, které rozsegmentované obrázky dle ground truth obsahují. Kombinace tříd je popsána pomocí zkratek názvů tříd (H - **H**ory, O - **O**bloha, V - **V**oda, D - le**D**ovec, L - **L**esní porost).

Zhodnocení výsledků chyby orientace kamery

Výsledky chyby odhadu orientace kamery ukazují, že segmenty ze všech vybraných modelů dosahují velice podobné průměrné celkové výsledky, ale při porovnání jednotlivých kombinací tříd se vzájemně liší. Nejlépe dopadly segmenty z verze modelu DeepLabv2-VGG16 s použitím CRF, poté z verze dotrénované na 5 třídách a následně z verze bez CRF. Ovšem u segmentů z modelu DeepLab-COCO-LargeFOV se naopak použití CRF na odhadu orientace kamery celkově projevilo negativně a také u těchto segmentů docházelo ke změnám hodnocení u jiných kombinací tříd než u segmentů z modelů DeepLabv2-VGG16. Nelze tedy obecně říct, že použitím CRF dochází k celkovému zlepšení odhadu orientace nebo zlepšení odhadu u segmentů obsahující určité kombinace tříd. Na rozsegmentovaných obrázcích, které dle ground truth obsahují ledovec, dopadl nejlépe model DeepLab-COCO-LargeFOV s CRF, který do třídy ledovec neklasifikuje, a tedy klasifikace do třídy ledovec nepřinesla z hlediska odhadu orientace kamery zlepšení. Ovšem výsledky získané pomocí rozšíření, které kromě segmentů využívá i hrany, přineslo u segmentů z modelu DeepLabv2-VGG16 dotrénovaného na 5 třídách oproti segmentům získaných z ostatních modelů výrazně vyšší zlepšení odhadu orientace kamery.

6.6 Zhodnocení experimentů

Při experimentech jsme nejprve dotrénovali modely z metody FCN, u kterých jsme experimentovali s původním klasifikátorem, který již některé horské oblasti obsahoval a umožnil nám vyhodnotit model na datové sadě GeoPose3K ještě před samotným trénováním. Poté jsme experimentovali s vlastním klasifikátorem, který nám umožnil libovolně kombinovat horské oblasti. Ve všech experimentech dosahoval lepších výsledků dotrénovaný model siftflow-fcn8, kdy s původním klasifikátorem trénování na výřezech přineslo významné zlepšení a naopak u vlastního klasifikátoru zlepšení nepřineslo. Druhý model pascalcontext-fcn8, přestože se dotrénováním na horských datech zlepšoval, tak proti modelu siftflow-fcn8 dosahoval výrazně horších výsledků. Po experimentech s FCN jsme dotrénovali modely z metody DeepLab a to jak z první verze v1, tak z novější verzi v2, která oproti první verzi přináší model založený na síti RESNET-101, který ovšem na horských datech nepřinesl oproti ostatním modelům z metody DeepLab založeným na síti VGG-16 zlepšení. Experimenty také ukázaly, že dotrénované modely z verze v1 dosahují lepších výsledků při trénování na datové sadě, která neobsahuje výřezy, kdežto modely z druhé verze naopak dosáhly nejlepších výsledků dotrénováním na datové sadě rozšířené o výřezy. Nakonec jsme experimentovali s frameworkem ALE, který sice dosahuje na 5 třídách horších výsledků jak modely z metod FCN a DeepLab, ale u třídy ledovec dosáhl ze všech metod nejlepších výsledků.

Po experimentech s trénováním jsme z nejlépe hodnocených dotrénovaných modelů vybrali 8, jimiž rozsegmentovanou testovací množinu datové sady GeoPose3K jsme nejprve použili pro vizuální vyhodnocení výsledků pomocí dotazníků a poté v procesu odhadu orientace kamery.

Porovnání dosažených výsledků mean IU vizuálního vyhodnocení a průměrné chyby orientace kamery u vybraných modelů (tabulka 6.23) ukazuje, že mezi jednotlivými modely jsou rozdíly, které se projeví při určitém způsobu použití segmentů. Nelze tedy říct, že některý model by byl komplexně lepší než ostatní modely. Segmenty z metody ALE z hlediska mean IU na 5 třídách i odhadu chyby orientace dosahovaly nejhorších výsledků, ovšem u vizuálního hodnocení byly druhými nejčastěji vybíranými. Naopak segmenty z modelů siftflow-fcn8s dosahovaly dobrého hodnocení mean IU i nízké chyby orientace u vysokého

rozlišení, ovšem při vizuálním hodnocení byly vybírány výrazně méně často. Vyhodnocení modelů z metody DeepLab ukázalo významný vliv použití CRF na vizuální hodnocení, kdy segmenty z metody DeepLab-COCO-LargeFOV s použitím CRF byly nejčastěji vybírané, ale bez použití CRF byly naopak druhé nejméně často vybírané. Z hlediska mean IU a průměrné chyby orientace obě verze segmentů dosahovaly velice podobných výsledků, které byly spíše průměrné. Segmenty z modelu DeepLabv2-VGG16 na 4 třídách dopadly z hlediska mean IU i chyby orientace velice dobře, ale při vizuálním hodnocení byly bez CRF nejméně vybíranými a s použitím CRF třetími nejčastěji vybíranými. Segmenty z modelu dotrénovaném na 5 třídách dosahovaly v porovnání s ostatními modely dotrénovanými na 5 třídách nejlepší hodnocení mean IU, ale u vizuálního hodnocení i u odhadu orientace dosahovaly pouze průměrných hodnot. Ovšem u odhadu orientace při použití rozšíření, které kombinuje segmenty s hranami dosáhly, segmenty výrazného zlepšení oproti ostatním segmentům a dosáhly nejmenší průměrnou chybu odhadu orientace kamery.

Při trénovaní na 5 třídách z hlediska metriky mean IU dopadaly nejlépe modely z metody DeepLab, kdy nejlepší model dosáhl hodnocení 52%. Při porovnání výsledků trénování na 4 třídách jak model siftflow-fcn8s, tak většina modelů z metody DeepLab dosáhly hodnocení mean IU kolem 57%. Dosažení velice podobného hodnocení by mohlo signalizovat potenciální limit, kterého jsou metody na aktuálně dostupných datech z datové sady GeoPose3K schopny dosáhnout. Vzhledem k tomu, že ground truth je generováno z OpenStreetMap a nemusí tedy vždy přesně reflektovat oblasti na fotografii, je možné, že pro další vylepšení metod, a tedy i dosažených výsledků by bylo potřeba zpřesnit ground truth. Jednou z možností zpřesnění by mohla být ruční korekce jak hranic, tak i samotných oblastí v ground truth.

Kompletní výsledky	mean IU (%)	Viz. vyhodnocení (%)	Chyba orientace (°)
ALE	46.0/41.6	18.3	66.4/61.7
siftflow-fcn8s (původní klas.)	**57.4**/**56.7**	9.2	**54.8**/61.4
siftflow-fcn8s (vlastní klas.)	56.8/55.9	10.5	63.6/61.6
DeepLab-COCO-LargeFOV	56.8/51.6	7.6	62.3/57.4
DeepLab-COCO-LargeFOV (CRF)	56.7/52.3	**21.2**	59.8/58.0
DeepLabv2-VGG16 (5 tříd)	52.0/50.1	11.5	62.8/55.4
DeepLabv2-VGG16	57.2/54.2	7.2	61.8/55.9
DeepLabv2-VGG16 (CRF)	57.2/56.5	14.5	60.2/**51.9**

Tabulka 6.23: Porovnání výsledků vybraných metod dosažených u vyhodnocení trénování (1. sloupec), vizuálního vyhodnocení (2. sloupec) a průměrné chyby orientace kamery (3. sloupec). Hodnoty v prvním sloupci jsou pro obrázky do maximální velkosti 400/800 pixelů a ve třetím sloupci u průměrné chyby orientace kamery jsou výsledky pro nízké/vysoké rozlišení.

Kapitola 7

Závěr

Cílem této práce bylo vybrat a následně dotrénovat na horských datech metody vhodné pro sémantickou segmentaci v horském prostředí.

Nejprve jsme se seznámili s podstatou sémantické segmentace a obecnými metodami a principy, na kterých jsou postaveny moderní segmentační metody. Poté jsme zhodnotili dostupné segmentační metody, ze kterých jsme vybrali metody FCN, DeepLab a ALE, které by mohly být vhodné pro sémantickou segmentaci v horském prostředí. Následně jsme upravili ground truth a rozdělili datovou sadu GeoPose3K tak, abychom získali objektivní výsledky. S vybranými metodami jsme postupně provedli sérii experimentů, ve kterých jsme testovali nejen, zda je možné metodu adaptovat na čistě horské prostředí, ale také vliv redukce rozpoznávaných horských tříd, trénování na výřezech, dostupných vylepšení a způsobu před-trénování na výsledky dotrénování jednotlivých modelů dostupných k vybraným metodám.

Experimenty se nám na horských datech podařilo úspěšně dotrénovat sérii modelů. Z dotrénovaných modelů jsme následně vybrali ty, které dosahovaly nejlepších výsledků, a segmenty z těchto modelů jsme nechali respondenty vizuálně vyhodnotit pomocí elektronického dotazník. Nakonec jsme vybrané modely úspěšně použili v procesu odhadu orientace kamery.

Výsledkem práce je tedy série 56 dotrénovaných modelů vhodných pro sémantickou segmentaci v horském prostředí, které jsou dotrénovány na 11, 5 nebo 4 horských třídách. Segmenty z jednotlivých dotrénovaných modelů se i přes podobné celkové hodnocení pomocí metriky mean IU vzájemně liší, což se projevilo jak u vizuálního hodnocení, tak u odhadu orientace kamery. Modely, které v některém z vyhodnocení dosáhly nejlepších výsledků, byly v ostatních vyhodnoceních spíše průměrné. Žádný z modelů tedy není komplexně lepší než všechny ostatní, ale do budoucna pro určité použití mohou být některé modely vhodnější než jiné.

Vzhledem k neustálému vývoji segmentačních metod by do budoucna mohla být tato práce rozšířena o další inovativní segmentační metody, které by mohly dosáhnout lepších výsledků trénování nebo být případně vhodnější pro použití v určitém procesu využívajícím sémantickou segmentaci v horském prostředí. Ovšem důležitým faktorem pro trénování modelů jsou také vstupní data, ze kterých se model učí, kdy kvalita výstupních dat reflektuje kvalitu vstupních dat. Práce by se tedy do budoucna dále mohla zabývat také zpřesněním ground truth, které jsou generovány z OpenStreetMap a nemusí vždy přesně odpovídat příslušné fotografii.

Literatura

[1] *Convolutional Neural Networks (LeNet)*. [Online; navštíveno 10.08.2016].
URL http://deeplearning.net/tutorial/lenet.html

[2] *Deep Learning for Java*. [Online; navštíveno 10.08.2016].
URL http://deeplearning4j.org/

[3] *Deep Learning Frameworks*. [Online; navštíveno 22.07.2016].
URL https://developer.nvidia.com/deep-learning-frameworks

[4] *Map Features*. [Online; navštíveno 25.12.2016].
URL http://wiki.openstreetmap.org/wiki/Map_Features#Natural

[5] *Semantic Boundary Dataset*. [Online; navštíveno 21.07.2016].
URL http://home.bharathh.info/home/sbd

[6] *Řetízkové pravidlo*. [Online; navštíveno 25.12.2016].
URL https://cs.wikipedia.org/wiki/%C5%98et%C3%ADzkov%C3%A9_pravidlo

[7] *Dictionary.com's 21st Century Lexicon*. Dec 2016, [Online; navštíveno 14.8.2016].
URL http://www.dictionary.com/browse/ground-truth

[8] Ahmad, T.; Campr, P.; Čadík, M.; aj.: *Comparison of Semantic Segmentation Approaches for Horizon/Sky Line Detection*. [Online; navštíveno 2.3.2017].
URL http://cadik.posvete.cz/papers/ahmad17comparison.pdf

[9] Aytar, Y.: *Segmentation Results: VOC2012*. [Online; navštíveno 25.12.2016].
URL http://host.robots.ox.ac.uk:8080/leaderboard/displaylb.php?challengeid=11&compid=6

[10] Badrinarayanan, V.; Kendall, A.; Cipolla, R.: SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. 2015, arXiv:1511.00561.

[11] Brejcha, J.; Čadík, M.: GeoPose3K: Mountain Landscape Dataset for Camera Pose Estimation in Outdoor Environments. Submitted to: *Image Vision and Computing*, 2017.

[12] Caesar, H.; Uijlings, J.; Ferrari, V.: Region-based semantic segmentation with end-to-end training. 2016, arXiv:1607.07671.

[13] Chen, L.-C.; Papandreou, G.; Kokkinos, I.; aj.: Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs. In *ICLR*, 2015, arXiv:1412.7062.

[14] Chen, L.-C.; Papandreou, G.; Kokkinos, I.; aj.: DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. 2016, arXiv:1606.00915.

[15] Christinal, H. A.; Díaz-Pernil, D.; Real, P.: *Region-based segmentation of 2D and 3D images with tissue-like P systems*. 2011, doi:http://dx.doi.org/10.1016/j.patrec.2011.05.004.
URL http://www.sciencedirect.com/science/article/pii/S0167865511001395

[16] Christos Stergiou, D. S.: *NEURAL NETWORKS*. [Online; navštíveno 10.08.2016].
URL
https://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html

[17] Csardi, G.: *R igraph manual pages*. [Online; navštíveno 26.12.2016].
URL http://www-test.igraph.org/r/doc/all.st.mincuts.html

[18] Dhawan, A. P.: *Image Segmentation*. John Wiley & Sons, Inc., 2011, ISBN 9780470918548, s. 229–264, doi:10.1002/9780470918548.ch10.
URL http://dx.doi.org/10.1002/9780470918548.ch10

[19] Everingham, M.; Van Gool, L.; Williams, C. K. I.; aj.: The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results.
http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

[20] Gulcehre, C.: *Welcome to Deep Learning*. [Online; navštíveno 22.07.2016].
URL http://deeplearning.net/

[21] HLAVÁČ, V. a. M. S.: *Počítačové vidění*. 1992, ISBN 80-85424-67-3, 97–121 s.

[22] Ing. Martin Čadík, P.: *LOCATE-vizuální lokalizace v přírodě*. [Online; navštíveno 22.07.2016].
URL http://cadik.posvete.cz/locate/locate_cz.pdf

[23] Jia, Y.: *Blobs, Layers, and Nets: anatomy of a Caffe model*. [Online; navštíveno 14.08.2016].
URL http://caffe.berkeleyvision.org/tutorial/net_layer_blob.html

[24] Jia, Y.: *Caffe*. [Online; navštíveno 14.08.2016].
URL http://caffe.berkeleyvision.org/

[25] Jia, Y.: *Forward and Backward*. [Online; navštíveno 14.08.2016].
URL http://caffe.berkeleyvision.org/tutorial/forward_backward.html

[26] Jia, Y.: *Loss*. [Online; navštíveno 14.08.2016].
URL http://caffe.berkeleyvision.org/tutorial/loss.html

[27] Jia, Y.: *Solver*. [Online; navštíveno 14.08.2016].
URL http://caffe.berkeleyvision.org/tutorial/solver.html

[28] Jia, Y.; Shelhamer, E.; Donahue, J.; aj.: Caffe: Convolutional Architecture for Fast Feature Embedding. *arXiv preprint arXiv:1408.5093*, 2014.

[29] Kendall, A.; Badrinarayanan, V.; Cipolla, R.: Bayesian SegNet: Model Uncertainty in Deep Convolutional Encoder-Decoder Architectures for Scene Understanding. 2015, arXiv:1511.02680.

[30] Kohli, P.; Ladický, L.; Torr, P. H. S.: Robust Higher Order Potentials for Enforcing Label Consistency. 2009, doi:10.1007/s11263-008-0202-0.
URL http://dx.doi.org/10.1007/s11263-008-0202-0

[31] Krähenbühl, P.; Koltun, V.: Efficient Inference in Fully Connected CRFs with Gaussian Edge Potentials. 2012, arXiv:1210.5644.

[32] Lubor Ladický, P. H. T.: *The Automatic Labelling Environment*. [Online; navštíveno 14.08.2016].
URL http://www.robots.ox.ac.uk/~phst/ale.htm

[33] LeCun, Y.; Bengio, Y.; Hinton, G.: Deep learning. May 28 2015.
URL http://search.proquest.com/docview/1685003444?accountid=17115

[34] Liang-Chieh Chen, I. K. a. s., George Papandreou: *DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs*. [Online; navštíveno 22.12.2016].
URL http://liangchiehchen.com/projects/DeepLab.html

[35] Lin, T.-Y.; Maire, M.; Belongie, S.; aj.: Microsoft COCO: Common Objects in Context. 2014, arXiv:1405.0312.

[36] Mostajabi, M.; Yadollahpour, P.; Shakhnarovich, G.: Feedforward semantic segmentation with zoom-out features. 2014, arXiv:1412.0774.

[37] Mottaghi, R.; Chen, X.; Liu, X.; aj.: *PASCAL-Context Dataset*. [Online; navštíveno 14.08.2016].
URL http://www.cs.stanford.edu/~roozbeh/pascal-context/

[38] Nathan Silberman, P. K., Derek Hoiem; Fergus, R.: *NYU Depth Dataset V2*. [Online; navštíveno 14.08.2016].
URL http://cs.nyu.edu/%7Esilberman/datasets/nyu_depth_v2.html

[39] Porzi, L.; Bulò, S. R.; Lanz, O.; aj.: An automatic image-to-DEM alignment approach for annotating mountains pictures on a smartphone. *Machine Vision and Applications*, ročník 28, č. 1, 2017: s. 101–115, ISSN 1432-1769.
URL http://dx.doi.org/10.1007/s00138-016-0808-0

[40] Shelhamer, E.; Long, J.; Darrell, T.: Fully Convolutional Networks for Semantic Segmentation. 2016, arXiv:1605.06211.

[41] Shiffman, D.: *The Nature of Code*. D. Shiffman, 2012, ISBN 9780985930806.

[42] Thoma, M.: A Survey of Semantic Segmentation. 2016, arXiv:1602.06541.

[43] Tighe, J.; Lazebnik, S.: SuperParsing: Scalable Nonparametric Image Parsing with Superpixels. 2010, european Conference on Computer Vision.

[44] Wang, C.: Scene parsing on SIFT Flow sub-dataset. 2015.
URL http://cs231n.stanford.edu/reports/canw_final+report.pdf

Přílohy

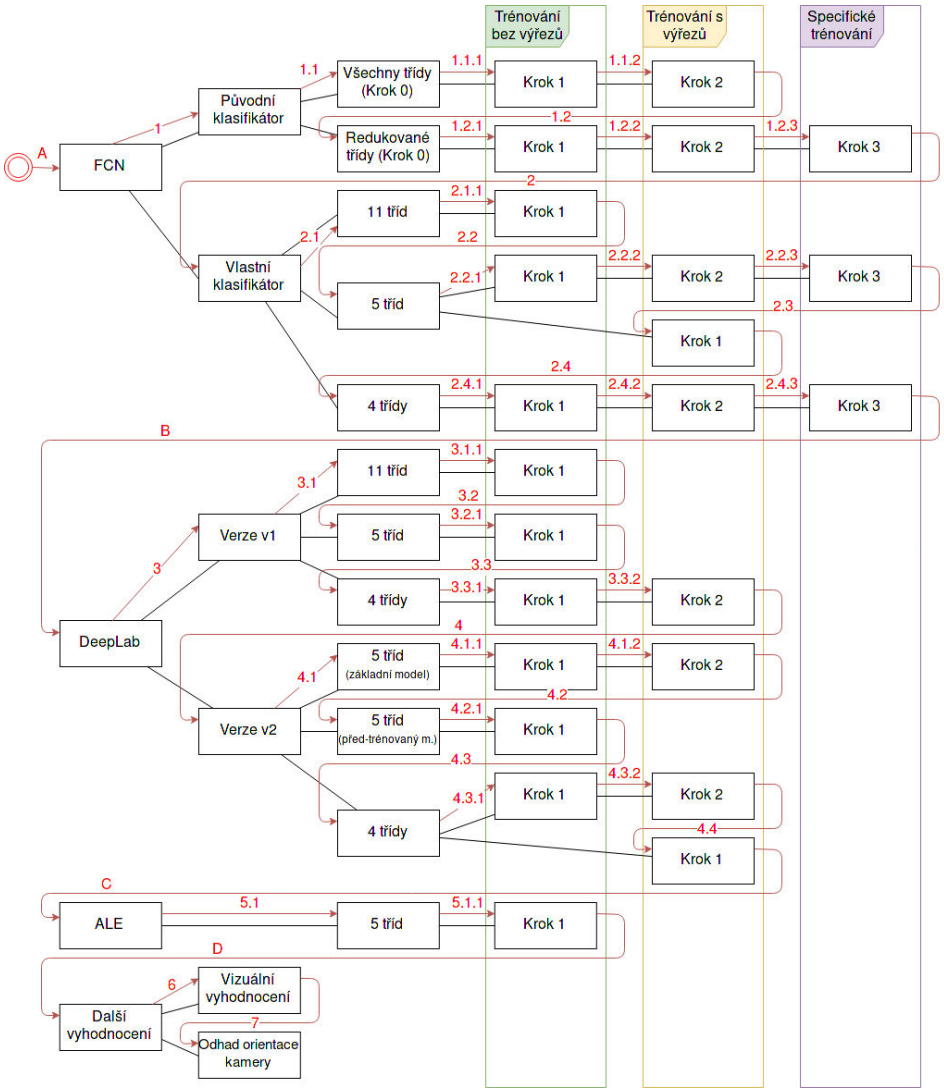
Příloha A

Obsah přiloženého paměťového média

- ALE/ - upravené zdrojové kódy frameworku ALE
- caffe/ - zdrojové kódy verzí frameworku Caffe používaných při experimentech
- e-verze_DP.pdf - textová část diplomové práce
- orientation_evaluation/ - zdrojové kódy pro vyhodnocení výsledků odhadu orientace kamery
- other_scripts/ - pomocné zdrojové kódy pro vyhodnocení a transformaci výsledků
- plakat.png - plakát demonstrující výsledek práce
- readme.txt - informace o diplomové práci a obsahu paměťového média
- segmentation/ - dotrénované modely, konfigurační soubory a zdrojové kódy včetně jednotného rozhraní pro úpravu datové sady, segmentaci a trénování modelů
- sets/ - testovací, trénovací a validační množiny použité při experimentech
- tex/ - zdrojové kódy textové části diplomové práce pro LaTeX
- www_research/ - zdrojové kódy pro vyhodnocení vizuálních výsledků, včetně zdrojových kódů webové stránky s dotazníkem

Příloha B

Diagram návaznosti experimentů



Obrázek B.1: Diagram zobrazuje pořadí popisu experimentů v rámci této práce (červené šipky) a hierarchii návaznosti experimentů (černé úsečky).

Příloha C

Kompletní naměřené výsledky

C.1 FCN - experimenty s původním klasifikátorem

C.1.1 Výsledky experimentů se všemi třídami

	Krok 0	Krok 0	Krok 1	Krok 1	Krok 2
Trénovaný model	pascalcontext fcn8s	siftflow fcn8s	pascalcontext fcn8s	siftflow fcn8s	siftflow fcn8s
Maximální velikost	bez trénování	bez trénování	256px	256px	256px
Dat. sada s výřezy	-	-	ne	ne	ano
Počet it. trénování	0	0	400 000	100 000	100 000
nejlepší iterace	-	-	23.63	39.79	43.93
mean IU nejlepší it.	-	-	320 000	96000	88 000
Testovací množina					
Maximální velikost	256px	256px	256px	256px	256px
pixel accuracy	47.31	59.04	62.72	72.01	72.78
mean accuracy	27.93	37.50	35.21	55.84	62.15
mean IU	2.94	4.63	17.12	36.96	48.81
f.w. accuracy	38.79	47.97	47.73	58.74	61.09
Maximální velikost	-	-	-	1000px	1000px
pixel accuracy	-	-	-	68.43	52.11
mean accuracy	-	-	-	49.71	57.14
mean IU	-	-	-	17.24	33.31
f.w. accuracy	-	-	-	53.17	41.94

Tabulka C.1: Výsledky experimentů metody FCN s původním klasifikátorem na datové sadě s neredukovaným počtem tříd v % (nav. diagram 1.1).

C.1.2 Výsledky experimentů s redukovaným počtem tříd

5/4 třídy	Krok 0	Krok 0	Krok 1	Krok 1	Krok 2	Krok 2	Krok 3
Trénovaný model	pascalcontext fcn8s	siftflow fcn8s	pascalcontext fcn8s	siftflow fcn8s	pascalcontext fcn8s	siftflow fcn8s	siftflow fcn8s
Maximální velikost	bez trénování	bez trénování	350px	400px	400px	400px	400px
Dat. sada s výřezy	ne	ne	ne	ne	ano	ano	ano
Learning rate	1.00E-14	1.00E-12	1.00E-14	1.00E-12	1.00E-14	1.00E-12	1E-12 krok 0.1 po 100K iter.
Počet it. trénování	0	0	400 000	100 000	400 000	100 000	200 000
nejlepší iterace	-	-	364 000	92 000	64 000	84 000	156 000
mean IU nejlepší it.	-	-	14.91	49.05	22.50	51.50	53.07
Testovací množina							
Maximální velikost	350px	350px	350px	350px	350px	350px	350px
pixel accuracy	49.05	63.91	66.79	77.49	65.79	78.39	78.94
mean accuracy	35.46	53.49	41.50	66.69	40.99	65.87	69.07
mean IU	3.00	5.16	15.73	53.84	20.94	54.78	57.40
f.w. accuracy	41.97	54.08	52.78	65.39	52.14	66.43	67.27
Maximální velikost	800px	800px	800px	800px	800px	800px	800px
pixel accuracy	44.39	61.02	64.10	76.33	65.91	78.88	79.37
mean accuracy	32.71	49.67	41.41	66.14	41.26	68.29	69.66
mean IU	2.50	4.76	4.93	29.81	8.48	56.06	57.32
f.w. accuracy	39.00	51.21	50.85	64.48	52.08	67.24	67.87
Maximální velikost	800px	1000px	800px	1000px	800px	1000px	1000px
pixel accuracy	44.39	60.43	64.10	75.26	65.91	78.67	79.22
mean accuracy	32.71	49.09	41.41	65.05	41.26	67.85	69.08
mean IU	2.50	4.68	4.93	25.33	8.48	55.43	56.68
f.w. accuracy	39.00	50.44	50.85	63.27	52.08	66.98	67.67

Tabulka C.2: Výsledky experimentů metody FCN s původním klasifikátorem na redukované datové sadě v % (nav. diagram 1.2).

C.2 FCN - experimenty s vlastním klasifikátorem

C.2.1 Výsledky experimentů s 11 třídami

11 tříd	Krok 1		Krok 1	
Trénovaný model	pascalcontext-fcn8s		siftflow-fcn8s	
Maximální velikost	400px		400px	
Dat. sada s výřezy	ne		ne	
Learning rate	1.00E-14		1.00E-12	
Počet it. trénování	400 000		200 000	
nejlepší iterace	224000		136000	
mean IU nejlepší iterace	16.74		20.51	
Testovací množina				
Maximální velikost	350px		350px	
pixel accuracy	63.49		73.34	
mean accuracy	24.06		30.03	
mean IU	17.05		23.71	
f.w. accuracy	49.44		60.33	
Maximální velikost	400px		400px	
pixel accuracy	63.44		73.41	
mean accuracy	23.92		30.29	
mean IU	16.97		23.71	
f.w. accuracy	49.33		60.52	
Maximální velikost	800px		800px	
pixel accuracy	62.43		72.31	
mean accuracy	22.58		29.96	
mean IU	16.01		22.87	
f.w. accuracy	47.80		59.43	
Maximální velikost	-		1000px	
pixel accuracy	-		71.68	
mean accuracy	-		29.35	
mean IU	-		22.26	
f.w. accuracy	-		58.66	
Statistiky jednotlivých tříd	Accuracy	IU	Accuracy	IU
Hory	65.60	42.94	71.14	52.96
Obloha	87.10	81.85	93.87	88.04
Lesní porost	54.43	31.45	56.17	37.26
Voda	38.14	19.84	51.31	33.14
Ledovec	6.26	5.93	22.42	20.73
Holá skála	11.48	4.53	8.04	4.97
Pastviny	0.02	0.01	0.00	0.00
Ostatní třídy	0.00	0.00	0.00	0.00

Tabulka C.3: Výsledky experimentů metody FCN s vlastním klas. na datové sadě s 11 třídami v % (nav. diagram 2.1).

C.2.2 Výsledky experimentů s 5 třídami

5 tříd	Krok 1	Krok 1	Krok 2	Krok 2	Krok 1	Krok 1
Trénovaný model	pascal. fcn8s	siftflow fcn8s	pascal. fcn8s	siftflow fcn8s	pascal. fcn8s	siftflow fcn8s
Maximální velikost	400px	400px	400px	400px	400px	400px
Dat. sada s výřezy	NE	NE	ANO	ANO	ANO	ANO
Learning rate	1.00E-14	1.00E-12	1.00E-14	1.00E-12	1.00E-14	1.00E-12
Počet it. trénování	400 000	200 000	400 000	200 000	400 000	200 000
nejlepší iterace	392000	120000	312000	144000	312000	144000
mIU nejlepší it.	30.88	41.35	30.59	41.18	30.26	40.18
Testovací množina						
Maximální velikost	350px	350px	350px	350px	350px	350px
pixel accuracy	64.66	75.42	62.26	75.09	64.56	74.69
mean accuracy	45.29	58.08	44.27	56.28	43.62	53.85
mean IU	31.69	46.37	29.91	44.81	30.52	42.47
f.w. accuracy	51.09	62.89	48.29	62.04	50.17	61.39
Maximální velikost	400px	400px	400px	400px	400px	400px
pixel accuracy	64.55	75.65	62.45	75.40	64.68	74.94
mean accuracy	45.06	58.70	44.16	57.04	43.57	54.57
mean IU	31.57	46.58	30.01	45.33	30.59	42.97
f.w. accuracy	50.90	63.28	48.40	62.45	50.23	61.79
Maximální velikost	800px	800px	800px	800px	800px	800px
pixel accuracy	62.78	74.85	62.27	76.01	64.47	75.63
mean accuracy	42.37	58.19	42.42	57.39	42.20	55.36
mean IU	29.56	45.14	29.36	45.23	29.99	43.23
f.w. accuracy	48.63	62.73	47.87	63.43	49.62	62.90
Maximální velikost	-	1000px	-	1000px	-	1000px
pixel accuracy	-	74.31	-	76.01	-	75.58
mean accuracy	-	57.44	-	56.74	-	54.72
mean IU	-	44.15	-	44.67	-	42.73
f.w. accuracy	-	62.18	-	63.43	-	62.86
Stat. jednotl. tříd	400px	400px	400px	400px	400px	400px
Accuracy						
Hory	60.50	76.70	57.85	78.57	64.99	80.65
Obloha	88.63	93.13	85.99	93.65	88.37	93.38
Lesní porost	57.40	48.14	56.34	42.56	50.15	37.30
Voda	43.22	51.24	42.14	54.17	41.14	50.89
Ledovec	19.11	24.26	22.36	16.26	16.68	10.63
IU						
Hory	43.67	57.48	41.73	57.76	45.05	58.02
Obloha	81.84	88.64	77.80	87.65	79.61	87.69
Lesní porost	32.03	34.27	30.31	31.70	30.41	28.62
Voda	16.51	31.19	15.14	34.55	15.94	30.52
Ledovec	14.04	21.34	14.76	15.00	12.44	10.02

Tabulka C.4: Výsledky experimentů metody FCN s vlastním klasifikátorem na datové sadě s 5 třídami v % (nav. diagram 2.2).

C.2.3 Výsledky experimentů se 4 třídami

4 třídy	Krok 1	Krok 1	Krok 2	Krok 2	Krok 3
Trénovaný model	pascalcontext fcn8s	siftflow fcn8s	pascalcontext fcn8s	siftflow fcn8s	siftflow fcn8s
Maximální velikost	400px	400px	400px	400px	400px
Dat. sada s výřezy	NE	NE	ANO	ANO	ANO
Learning rate	1.00E-14	1.00E-12	1.00E-14	1.00E-12	1.00E-12 krok 0.1 po 100K it.
Počet it. trénování	400 000	200 000	400 000	200 000	200 000
nejlepší iterace	392000	188000	264000	176000	60000
mIU nejlepší it.	35.72	50.05	35.10	50.84	51.50
Testovací množina					
Maximální velikost	350px	350px	350px	350px	350px
pixel accuracy	66.13	77.69	64.01	76.59	77.46
mean accuracy	50.52	66.94	45.67	69.24	70.19
mean IU	35.61	54.67	33.69	54.64	56.41
f.w. accuracy	52.67	65.73	49.56	64.77	65.73
Maximální velikost	400px	400px	400px	400px	400px
pixel accuracy	66.08	78.08	64.25	77.01	77.90
mean accuracy	50.51	67.80	46.07	69.91	70.67
mean IU	35.64	55.12	34.06	54.93	56.76
f.w. accuracy	52.55	66.26	49.92	65.25	66.22
Maximální velikost	800px	800px	800px	800px	800px
pixel accuracy	64.04	76.84	64.53	76.98	77.90
mean accuracy	48.44	67.40	46.18	69.18	69.41
mean IU	33.92	53.43	34.38	54.30	55.91
f.w. accuracy	50.03	65.12	50.48	65.31	66.27
Maximální velikost	-	1000px	-	1000px	1000px
pixel accuracy	-	76.11	-	76.76	77.59
mean accuracy	-	66.29	-	68.01	68.18
mean IU	-	51.96	-	53.35	54.82
f.w. accuracy	-	64.30	-	65.08	65.91
Stat. jednotl. tříd	400px	400px	400px	400px	400px
Accuracy					
Hory	61.77	80.11	71.28	72.51	77.54
Obloha	88.49	92.60	79.93	93.48	91.97
Lesní porost	58.05	47.53	44.25	56.59	52.59
Voda	42.67	50.94	34.26	57.04	60.57
IU					
Hory	46.10	62.10	47.98	59.07	61.36
Obloha	81.80	88.69	75.43	88.12	88.21
Lesní porost	32.05	34.36	25.22	36.94	36.19
Voda	16.80	35.31	21.03	35.60	41.28

Tabulka C.5: Výsledky experimentů metody FCN s vlastním klasifikátorem na datové sadě se 4 třídami v % (nav. diagram 2.4).

C.3 DeepLab verze v1

C.3.1 Výsledky experimentů s 11 třídami (část 1)

Trénovaný model	DeepLab		DeepLab COCO-LargeFOV		DeepLab LargeFOV	
Maximální velikost	400px		400px		400px	
Počet it. trénování	6 000		6 000		6 000	
Dat. sada s výřezy	NE		NE		NE	
CRF	NE	ANO	NE	ANO	NE	ANO
Maximální velikost	400px	400px	400px	400px	400px	400px
pixel accuracy	74.40	74.99	74.42	74.47	74.51	74.95
mean accuracy	30.60	30.64	31.58	30.20	33.54	32.33
mean IU	24.82	25.12	25.52	24.82	26.67	26.50
f.w. accuracy	61.88	62.54	61.94	61.82	62.12	62.42
Maximální velikost	800px	800px	800px	800px	800px	800px
pixel accuracy	72.86	73.49	73.42	73.61	72.51	73.22
mean accuracy	27.75	27.97	28.21	27.46	29.19	28.58
mean IU	21.87	22.25	22.45	21.96	22.69	22.62
f.w. accuracy	59.51	60.22	60.33	60.52	59.38	60.27
Stat. jednotl. tříd	400px	400px	400px	400px	400px	400px
Accuracy						
Hory	73.67	74.61	71.09	72.13	70.40	72.77
Obloha	93.25	93.56	93.73	93.52	93.49	93.40
Lesní porost	57.32	58.04	60.92	60.97	61.03	60.09
Voda	38.12	38.08	45.62	39.00	52.34	50.29
Ledovec	35.11	35.61	33.73	29.34	43.38	38.14
Holá skála	8.52	6.53	10.75	7.06	14.80	8.61
Ostatní třídy	0.00	0.00	0.00	0.00	0.00	0.00
IU						
Hory	54.45	55.19	53.46	53.82	53.52	54.43
Obloha	88.88	89.46	89.20	88.97	88.90	89.11
Lesní porost	38.76	39.48	39.82	39.65	39.95	39.56
Voda	29.68	30.33	34.78	32.00	37.98	38.71
Ledovec	32.11	33.12	31.95	29.00	38.90	36.90
Holá skála	4.36	3.66	5.95	4.73	7.47	6.24
Ostatní třídy	0.00	0.00	0.00	0.00	0.00	0.00

Tabulka C.6: Výsledky experimentů metody DeepLab v1 na datové sadě s 11 třídami v % - první část (nav. diagram 3.1).

C.3.2 Výsledky experimentů s 11 třídami (část 2)

Trénovaný model	DeepLab-MSC		DeepLab-MSC COCO-LargeFOV		DeepLab-MSC LargeFOV	
Maximální velikost	400px		400px		400px	
Počet it. trénování	6 000		6 000		6 000	
Dat. sada s výřezy	NE		NE		NE	
CRF	NE	ANO	NE	ANO	NE	ANO
Maximální velikost	400px	400px	400px	400px	400px	400px
pixel accuracy	73.88	74.15	72.91	73.55	72.73	73.35
mean accuracy	27.35	26.43	24.69	24.24	24.34	23.87
mean IU	21.89	21.22	19.76	19.61	19.47	19.29
f.w. accuracy	60.43	60.54	58.82	59.33	58.58	59.00
Maximální velikost	800px	800px	800px	800px	800px	800px
pixel accuracy	-	-	-	-	-	-
mean accuracy	-	-	-	-	-	-
mean IU	-	-	-	-	-	-
f.w. accuracy	-	-	-	-	-	-
Stat. jednotl. tříd	400px	400px	400px	400px	400px	400px
Accuracy						
Hory	78.47	80.06	81.75	84.01	82.45	84.88
Obloha	92.62	92.70	91.69	92.35	91.25	91.96
Lesní porost	49.69	48.93	42.26	40.51	41.00	38.73
Voda	39.39	37.77	28.46	25.52	26.39	23.14
Ledovec	10.39	4.86	2.29	0.00	2.04	0.00
Holá skála	2.98	0.00	0.42	0.00	0.26	0.00
Ostatní třídy	0.00	0.00	0.00	0.00	0.00	0.00
IU						
Hory	55.42	55.89	55.29	56.26	55.14	56.14
Obloha	87.07	87.27	85.84	86.51	85.72	86.27
Lesní porost	35.88	35.96	31.76	31.62	31.22	30.77
Voda	27.90	28.26	22.04	21.68	20.35	19.71
Ledovec	10.26	4.85	2.28	0.00	2.04	0.00
Holá skála	2.35	0.00	0.42	0.00	0.26	0.00
Ostatní třídy	0.00	0.00	0.00	0.00	0.00	0.00

Tabulka C.7: Výsledky experimentů metody DeepLab v1 na datové sadě s 11 třídami v % - druhá část (nav. diagram 3.1).

C.3.3 Výsledky experimentů s 5 třídami

Trénovaný model	DeepLab		DeepLab COCO-LargeFOV		DeepLab LargeFOV		DeepLab-MSC		DeepLab-MSC COCO-LargeFOV		DeepLab-MSC LargeFOV	
Dat. sada s výřezy	NE		NE		NE		NE		NE		NE	
CRF	NE	ANO	NE	ANO	NE	ANO	NE	ANO	NE	ANO	NE	ANO
Maximální velikost	400px	400px	400px	400px	400px	400px	400px	400px	400px	400px	400px	400px
pixel accuracy	76.88	77.33	76.03	76.09	76.79	76.71	76.14	76.42	74.97	75.85	74.67	75.19
mean accuracy	56.29	56.15	60.03	57.34	61.20	57.28	53.07	51.71	48.18	47.94	47.26	46.01
mean IU	46.38	46.53	48.97	47.36	50.47	47.65	43.23	42.37	39.21	39.55	38.48	37.81
fwavacc	64.53	65.03	63.97	63.92	64.74	64.47	63.19	63.35	61.34	62.15	60.73	60.92
Maximální velikost	800px	800px	800px	800px	800px	800px	800px	800px	800px	800px	800px	800px
pixel accuracy	75.11	75.69	74.34	74.59	74.90	75.52	-	-	-	-	-	-
mean accuracy	53.23	53.27	54.79	52.94	56.70	54.46	-	-	-	-	-	-
mean IU	42.78	43.10	43.30	42.17	45.59	44.16	-	-	-	-	-	-
fwavacc	62.33	63.01	61.85	62.18	62.54	63.28	-	-	-	-	-	-
Stat. jednotl. tříd	400px	400px	400px	400px	400px	400px	400px	400px	400px	400px	400px	400px
Accuracy												
Hory	78.79	79.48	70.71	71.61	74.98	76.99	81.19	82.74	84.80	86.86	86.24	89.14
Obloha	93.18	93.57	93.42	93.31	93.14	93.10	92.27	92.49	90.89	92.02	90.70	91.66
Lesní porost	52.37	52.83	62.56	62.89	57.12	54.92	46.59	45.26	37.42	35.77	33.32	28.55
Voda	41.96	41.67	47.76	40.23	48.34	42.43	38.13	35.99	27.70	25.04	26.03	20.68
Ledovec	15.16	13.21	25.72	18.65	32.44	18.96	7.17	2.06	0.11	0.00	0.00	0.00
IU												
Hory	59.21	59.82	55.61	55.84	58.06	58.41	59.44	59.92	59.12	60.22	59.04	60.01
Obloha	89.01	89.46	89.27	89.23	88.96	89.11	87.23	87.52	86.05	87.10	85.71	86.61
Lesní porost	37.37	38.03	39.83	40.24	38.43	37.76	34.70	34.62	29.41	29.34	27.06	24.57
Voda	31.79	32.50	35.52	33.29	35.42	34.05	27.65	27.71	21.38	21.11	20.60	17.84
Ledovec	14.52	12.83	24.60	18.20	31.47	18.91	7.13	2.06	0.11	0.00	0.00	0.00

Tabulka C.8: Výsledky experimentů metody DeepLab v1 na datové sadě s 5 třídami v % (nav. diagram 3.2).

C.3.4 Výsledky experimentů se 4 třídami (část 1)

Trénovaný model	DeepLab				DeepLab-COCO-LargeFOV				DeepLab-LargeFOV			
	Krok 1		Krok 2		Krok 1		Krok 2		Krok 1		Krok 2	
Maximální velikost	400px		400px		400px		400px		400px		400px	
Počet it. trénování	6 000		6 000		6 000		6 000		6 000		6 000	
Dat. sada s výřezy	NE		ANO		NE		ANO		NE		ANO	
CRF	NE	ANO	NE	ANO	NE	ANO	NE	ANO	NE	ANO	NE	ANO
Maximální velikost	400px	400px	400px	400px	400px	400px	400px	400px	400px	400px	400px	400px
pixel accuracy	78.20	78.49	79.43	79.83	78.69	78.72	79.40	79.17	78.23	78.33	78.84	78.64
mean accuracy	66.33	66.07	67.10	67.42	69.56	68.24	66.63	65.33	67.58	66.04	66.03	63.82
mean IU	55.19	55.22	56.44	56.93	56.85	56.74	55.98	55.00	55.50	54.91	55.23	53.74
fwavacc	66.44	66.82	67.76	68.25	67.25	67.18	67.90	67.53	66.64	66.60	67.03	66.58
Maximální velikost	800px	800px	800px	800px	800px	800px	800px	800px	800px	800px	800px	800px
pixel accuracy	76.59	77.03	79.00	79.29	75.90	76.60	79.01	79.10	76.58	77.25	79.03	79.30
mean accuracy	64.10	64.31	65.90	66.14	65.72	65.46	64.44	63.84	64.55	63.85	65.18	64.90
mean IU	52.02	52.50	55.09	55.46	51.58	52.31	54.10	53.76	52.09	52.24	54.56	54.74
fwavacc	64.31	64.84	67.16	67.51	64.15	64.96	67.31	67.36	64.82	65.62	67.26	67.49
Stat. jednotl. tříd	400px	400px	400px	400px	400px	400px	400px	400px	400px	400px	400px	400px
Accuracy												
Hory	78.13	78.39	83.23	83.43	77.23	78.27	81.10	81.20	76.90	78.87	82.98	84.33
Obloha	93.00	93.43	92.53	93.15	93.18	93.22	92.79	93.03	92.92	92.97	91.82	92.24
Lesní porost	52.76	53.17	48.57	49.00	56.07	54.29	53.22	51.68	55.51	52.06	47.61	43.31
Voda	41.42	39.31	44.07	44.08	51.74	47.17	39.43	35.43	44.98	40.31	41.72	35.39
IU												
Hory	61.69	62.07	64.41	64.88	61.95	62.17	63.66	63.44	61.29	61.82	63.43	63.56
Obloha	88.62	89.09	89.13	89.63	89.37	89.30	89.47	89.35	88.94	88.98	88.80	88.82
Lesní porost	36.57	36.94	36.23	36.77	38.45	37.71	38.39	37.45	37.85	36.65	35.50	33.15
Voda	33.85	32.76	36.01	36.42	37.62	37.78	32.38	29.74	33.90	32.21	33.17	29.43

Tabulka C.9: Výsledky experimentů metody DeepLab v1 na datové sadě se 4 třídami v % - první část (nav. diagram 3.3).

C.3.5 Výsledky experimentů se 4 třídami (část 2)

Trénovaný model	DeepLab-MSC		DeepLab-MSC COCO-LargeFOV		DeepLab-MSC LargeFOV	
Maximální velikost	400px		400px		400px	
Počet it. trénování	6 000		6 000		6 000	
Dat. sada s výřezy	NE		NE		NE	
Testovací množina						
CRF	NE	ANO	NE	ANO	NE	ANO
Maximální velikost	400px	400px	400px	400px	400px	400px
pixel accuracy	78.39	78.79	77.27	78.03	76.84	77.17
mean accuracy	64.83	64.29	60.35	59.76	58.85	56.67
mean IU	53.38	53.60	50.03	50.21	48.65	47.25
fwavacc	66.28	66.61	64.24	64.88	63.54	63.35
Stat. jednotl. tříd	400px	400px	400px	400px	400px	400px
Accuracy						
Hory	82.28	63.53	86.06	63.10	87.29	62.85
Obloha	92.00	87.29	91.08	85.92	90.79	85.86
Lesní porost	46.82	34.87	35.19	28.29	30.95	25.49
Voda	38.22	27.82	29.12	22.93	26.40	26.40
IU						
Hory	83.67	64.12	88.40	64.21	90.38	63.64
Obloha	92.34	87.60	92.09	86.99	91.77	86.59
Lesní porost	45.37	34.65	32.27	27.16	24.50	21.61
Voda	35.80	28.03	26.34	22.59	20.06	17.26

Tabulka C.10: Výsledky experimentů metody DeepLab v1 na datové sadě se 4 třídami v % - druhá část (nav. diagram 3.3).

C.4 DeepLab verze v2

C.4.1 Výsledky experimentů s 5 třídami

5 tříd	Krok 1		Krok 1		Krok 2		Krok 2	
Trénovaný model	DeepLab-v2 VGG16		DeepLab-v2 ResNet101		DeepLab-v2 VGG16		DeepLab-v2 ResNet101	
Maximální velikost	400px		400px		400px		400px	
Dat. sada s výřezy	NE		NE		ANO		ANO	
Počet it. trénování	20 000		20 000		20 000		20 000	
Testovací množina								
CRF	NE	ANO	NE	ANO	NE	ANO	NE	ANO
Maximální velikost	385px	385px	385px	385px	385px	385px	385px	385px
pixel accuracy	77.07	77.35	75.97	76.17	77.56	77.68	76.76	77.02
mean accuracy	60.20	58.69	58.84	57.80	63.42	62.11	60.72	59.54
mean IU	49.95	48.94	48.71	48.02	53.12	52.25	50.51	49.77
fwavacc	64.92	65.16	63.66	63.85	65.49	65.58	64.47	64.69
Maximální velikost	400px	400px	-	-	400px	400px	-	-
pixel accuracy	78.02	78.46	-	-	78.05	78.34	-	-
mean accuracy	60.77	59.35	-	-	63.39	62.24	-	-
mean IU	50.05	49.47	-	-	52.05	51.60	-	-
fwavacc	66.01	66.45	-	-	66.05	66.31	-	-
Maximální velikost	800px	800px	-	-	800px	800px	-	-
pixel accuracy	76.09	76.85	-	-	77.93	78.53	-	-
mean accuracy	57.27	57.20	-	-	61.93	61.82	-	-
mean IU	46.19	46.62	-	-	50.10	50.53	-	-
fwavacc	63.79	64.71	-	-	65.93	66.67	-	-
Stat. jednotl. tříd	385px	385px	385px	385px	385px	385px	385px	385px
Accuracy								
Hory	78.74	80.23	77.64	78.78	78.33	79.48	79.10	80.58
Obloha	92.65	93.12	93.05	93.31	92.90	93.11	93.02	93.17
Lesní porost	52.10	50.49	48.02	46.81	53.26	51.91	48.50	47.20
Voda	47.71	45.96	47.67	45.83	55.42	54.35	57.19	55.16
Ledovec	29.78	23.63	27.81	24.25	37.20	31.69	25.81	21.59
IU								
Hory	59.13	59.70	57.71	58.23	59.76	60.13	58.88	59.52
Obloha	89.16	89.71	89.13	89.51	88.86	89.23	89.15	89.50
Lesní porost	37.16	36.90	33.69	33.30	37.63	37.23	34.80	34.51
Voda	36.30	36.00	36.09	35.41	43.62	43.83	44.38	43.91
Ledovec	28.02	22.40	26.93	23.62	35.72	30.82	25.33	21.44

Tabulka C.11: Výsledky experimentů metody DeepLab v2 na datové sadě s 5 třídami v % (nav. diagram 4.1).

C.4.2 Výsledky experimentů s 5 třídami - dotrénování před-trénovaného modelu

5 tříd	Krok 1		Krok 1	
Trénovaný model	DeepLab-v2 VGG16		DeepLab-v2 ResNet101	
Maximální velikost	400px		400px	
Dat. sada s výřezy	NE		NE	
Počet it. trénování	20 000		20 000	
Testovací množina				
CRF	NE	ANO	NE	ANO
Maximální velikost	385px	385px	385px	385px
pixel accuracy	76.68	76.83	76.18	76.35
mean accuracy	59.65	58.14	57.35	56.30
mean IU	49.36	48.48	46.83	46.01
fwavacc	64.43	64.53	63.80	63.93
Maximální velikost	400px	400px	400px	400px
pixel accuracy	77.65	77.86	-	-
mean accuracy	60.28	58.79	-	-
mean IU	49.29	48.72	-	-
fwavacc	65.51	65.68	-	-
Maximální velikost	800px	800px	800px	800px
pixel accuracy	75.44	76.13	-	-
mean accuracy	54.20	53.10	-	-
mean IU	43.18	42.76	-	-
fwavacc	62.68	63.49	-	-
Stat. jednotl. tříd	385px	385px	385px	385px
Accuracy				
Hory	78.74	79.74	77.85	78.92
Obloha	92.84	93.29	93.28	93.51
Lesní porost	49.70	48.53	49.54	48.40
Voda	48.55	47.88	48.99	48.30
Ledovec	28.44	21.24	17.09	12.38
IU				
Hory	58.73	59.00	58.29	58.73
Obloha	89.13	89.67	89.00	89.35
Lesní porost	35.56	35.16	34.87	34.54
Voda	36.31	37.92	35.29	35.16
Ledovec	27.05	20.64	16.72	12.28

Tabulka C.12: Výsledky experimentů metody DeepLab v2 na datové sadě s 5 třídami v % (nav. diagram 4.2).

C.4.3 Výsledky experimentů se 4 třídami

4 třídy	Krok 1		Krok 1		Krok 2		Krok 2	
Trénovaný model	DeepLab-v2 VGG16		DeepLab-v2 ResNet101		DeepLab-v2 VGG16		DeepLab-v2 ResNet101	
Maximální velikost	400px		400px		400px		400px	
Dat. sada s výřezy	NE		NE		ANO		ANO	
Počet it. trénování	20 000		20 000		20 000		20 000	
Testovací množina								
CRF	NE	ANO	NE	ANO	NE	ANO	NE	ANO
Maximální velikost	385px	385px	385px	385px	385px	385px	385px	385px
pixel accuracy	78.20	78.65	78.14	78.44	78.35	78.62	78.58	79.04
mean accuracy	67.70	67.21	68.08	68.01	68.76	68.34	68.76	69.01
mean IU	56.01	56.06	56.09	56.29	57.08	57.25	56.50	57.05
wavacc	66.54	67.04	66.35	66.66	66.75	66.99	66.83	67.31
Maximální velikost	400px	400px	400px	400px	400px	400px	400px	400px
pixel accuracy	78.17	78.48	-	-	78.54	78.83	-	-
mean accuracy	67.61	66.98	-	-	68.70	68.21	-	-
mean IU	55.87	55.71	-	-	57.07	57.15	-	-
fwavacc	66.54	66.88	-	-	66.99	67.28	-	-
Maximální velikost	800px	800px	800px	800px	800px	800px	800px	800px
pixel accuracy	76.29	77.05	-	-	78.50	79.09	-	-
mean accuracy	64.24	64.62	-	-	66.84	67.37	-	-
mean IU	52.09	52.78	-	-	55.67	56.49	-	-
fwavacc	64.13	65.11	-	-	66.92	67.65	-	-
Stat. jednotl. tříd	385px	385px	385px	385px	385px	385px	385px	385px
Accuracy								
Hory	78.95	80.34	79.16	80.12	77.76	79.09	80.18	81.51
Obloha	92.69	93.26	93.12	93.33	92.68	92.97	93.04	93.32
Lesní porost	50.98	49.39	49.04	48.15	54.28	52.30	49.10	47.91
Voda	48.18	45.85	50.98	50.46	50.31	49.01	52.71	53.31
IU								
Hory	61.91	62.69	61.82	62.33	61.95	62.55	62.52	63.35
Obloha	88.99	89.64	88.92	89.19	88.58	88.84	89.03	89.36
Lesní porost	35.62	35.31	34.82	34.78	37.08	36.50	35.57	35.49
Voda	37.54	36.60	38.78	38.84	40.71	41.13	38.89	40.01

Tabulka C.13: Výsledky experimentů metody DeepLab v2 na datové sadě se 4 třídami v % (nav. diagram 4.3).

C.4.4 Výsledky experimentů se 4 třídami - trénování na výřezech v 1. kroku

4 třídy	Krok 1		Krok 1		Krok 1		Krok 1	
Trénovaný model	DeepLab-v2 VGG16		DeepLab-v2 ResNet101		DeepLab-v2 VGG16		DeepLab-v2 ResNet101	
Maximální velikost	400px		400px		400px		400px	
Dat. sada s výřezy	ANO		ANO		ANO		ANO	
Počet it. trénování	20 000		20 000		40 000		40 000	
Testovací množina								
CRF	NE	ANO	NE	ANO	NE	ANO	NE	ANO
Maximální velikost	385px	385px	385px	385px	385px	385px	385px	385px
pixel accuracy	79.21	79.05	77.01	77.11	78.01	77.80	77.96	78.05
mean accuracy	68.37	67.55	68.74	68.52	68.87	68.26	67.74	67.52
mean IU	57.41	56.78	56.92	56.88	56.70	56.48	55.90	55.92
fwavacc	67.49	67.27	65.57	65.68	66.16	65.93	66.15	66.23
Maximální velikost	400px	400px	400px	400px	400px	400px	400px	400px
pixel accuracy	79.24	79.06	-	-	77.96	77.86	-	-
mean accuracy	68.15	67.37	-	-	68.35	67.97	-	-
mean IU	57.22	56.58	-	-	56.19	56.20	-	-
fwavacc	67.56	67.34	-	-	66.10	66.00	-	-
Maximální velikost	800px	800px	800px	800px	800px	800px	800px	800px
pixel accuracy	78.78	78.83	-	-	77.31	77.86	-	-
mean accuracy	64.50	64.55	-	-	65.61	66.22	-	-
mean IU	54.16	54.18	-	-	53.71	54.52	-	-
fwavacc	67.28	67.43	-	-	65.23	65.97	-	-
Stat. jednotl. tříd	385px	385px	385px	385px	385px	385px	385px	385px
Accuracy								
Hory	84.06	83.69	73.11	73.07	75.56	75.71	78.68	79.25
Obloha	90.89	91.81	91.86	92.41	92.75	93.05	92.91	93.20
Lesní porost	48.21	46.61	59.18	58.81	57.31	55.40	49.73	48.39
Voda	50.32	48.08	50.83	49.80	49.84	48.90	49.64	49.23
IU								
Hory	64.31	64.02	58.65	58.69	61.05	60.80	61.58	61.76
Obloha	88.44	88.83	88.94	89.27	87.36	87.65	88.60	88.88
Lesní porost	36.02	34.89	37.31	37.20	38.47	37.25	34.93	34.35
Voda	40.86	39.38	42.77	42.35	39.91	40.21	38.50	38.70

Tabulka C.14: Výsledky experimentů metody DeepLab v2 na datové sadě se 4 třídami, která byla již v prvním kroku rozšířena o výřezy (nav. diagram 4.4). Výsledky jsou uvedeny v %.

C.5 ALE

C.5.1 Výsledky experimentů s 5 třídami

5 tříd	ALE	
Maximální velikost	400px	
Dat. sada s výřezy	NE	
Testovací množina		
Maximální velikost	400px	800px
pixel accuracy	70.28	70.32
mean accuracy	60.67	56.23
mean IU	46.01	41.59
f.w. accuracy	56.44	56.13
Stat. jednotl. tříd	400px	-
Accuracy		
Hory	60.24	-
Obloha	90.69	-
Lesní porost	56.44	-
Voda	46.74	-
Ledovec	49.26	-
IU		
Hory	48.81	-
Obloha	78.93	-
Lesní porost	33.14	-
Voda	29.65	-
Ledovec	39.54	-

Tabulka C.15: Výsledky experimentů metody ALE na datové sadě s 5 třídami v % (nav. diagram 5.1).

C.6 Venturi

Model	DeepLab-v2-VGG16						FCN-siftflow	
Varianta	Krok 3		Krok 1 - výřezy		Krok 2		Krok 3	
Počet tříd	5	5	4	4	4	4	5	4
CRF	NE	ANO	NE	ANO	NE	ANO	-	-
overall accuracy	83.07	83.62	86.34	87.41	82.61	84.36	84.09	86.99
mean accuracy	48.93	49.25	63.90	64.68	60.49	61.98	49.26	64.41
mean IU	41.77	42.21	55.99	57.16	51.77	53.62	43.10	56.80
fwavacc	73.20	73.92	77.46	78.98	72.80	74.99	75.33	78.42
Accuracy								
Hory	86.76	87.28	85.00	86.26	77.84	81.39	74.69	85.36
Obloha	95.37	95.97	94.22	95.59	95.38	96.00	95.87	94.61
Lesní porost	62.53	63.00	76.38	76.86	68.74	70.50	75.69	77.66
Voda	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Ledovec	0.00	0.00	-	-	-	-	0.05	-
IU								
Hory	58.89	59.80	64.78	66.50	55.62	59.38	56.73	65.57
Obloha	94.92	95.55	93.92	95.22	94.90	95.56	95.14	93.95
Lesní porost	55.02	55.68	65.25	66.90	56.57	59.56	63.59	67.69
Voda	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Ledovec	0.00	0.00	-	-	-	-	0.05	-

Tabulka C.16: Výsledky segmentace datové sady Venturi pomocí dotrénovaných modelů v % - první část.

Model	DeepLab				DeepLab-LargeFOV				DeepLab-COCO-LargeFOV			
Varianta	Krok 1		Krok 1		Krok 1		Krok 1		Krok 1		Krok 1	
Počet tříd	5	5	4	4	5	5	4	4	5	5	4	4
CRF	NE	ANO	NE	ANO	NE	ANO	NE	ANO	NE	ANO	NE	ANO
overall accuracy	84.73	86.48	81.26	83.38	84.22	84.23	85.30	85.02	87.83	88.13	83.83	82.39
mean accuracy	50.01	51.22	59.44	61.05	49.63	49.61	62.96	62.72	51.95	52.21	61.72	60.64
mean IU	43.24	44.81	50.15	52.42	42.82	42.78	54.68	54.31	46.10	46.40	52.96	51.27
fwavacc	75.39	77.62	70.44	73.31	74.72	74.74	76.09	75.71	79.55	79.97	74.14	72.13
Accuracy												
Hory	87.42	90.43	79.78	77.59	83.52	85.08	86.67	87.80	85.45	87.57	87.63	90.81
Obloha	95.71	95.92	95.35	96.03	95.10	95.78	95.62	96.05	96.26	96.13	96.16	96.12
Lesní porost	66.92	69.74	62.64	70.57	69.52	67.20	69.54	67.01	78.04	77.35	63.10	55.63
Voda	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Ledovec	0.00	0.00	-	-	0.00	0.00	-	-	0.00	0.00	-	-
IU												
Hory	61.34	65.55	55.96	58.26	59.77	60.41	62.34	62.39	67.16	68.31	60.26	59.19
Obloha	95.27	95.34	91.98	93.50	94.51	95.35	95.02	95.50	95.54	95.63	95.43	95.48
Lesní porost	59.58	63.13	52.64	57.92	59.84	58.13	61.35	59.38	67.79	68.03	56.16	50.43
Voda	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Ledovec	0.00	0.00	-	-	0.00	0.00	-	-	0.00	0.00	-	-

Tabulka C.17: Výsledky segmentace datové sady Venturi pomocí dotrénovaných modelů v % - druhá část.

Model	DeepLab-MSC				DeepLab-MSc-COCO-LargeFOV				DeepLab-MSc-LargeFOV			
Varianta	Krok 1		Krok 1		Krok 1		Krok 1		Krok 1		Krok 1	
Počet tříd	5	5	4	4	5	5	4	4	5	5	4	4
CRF	NE	ANO	NE	ANO	NE	ANO	NE	ANO	NE	ANO	NE	ANO
overall accuracy	79.91	79.71	79.88	80.25	78.43	78.91	78.40	78.93	79.48	81.26	78.80	79.63
mean accuracy	46.84	46.63	58.55	58.76	45.92	46.17	57.31	57.69	46.62	47.70	57.81	58.35
mean IU	39.04	38.91	48.76	49.17	37.82	38.17	47.26	47.77	38.70	40.17	47.59	48.44
fwavacc	69.24	69.12	69.19	69.75	67.37	67.98	67.36	68.06	68.62	70.91	67.67	68.81
Accuracy												
Hory	86.79	84.83	87.50	85.86	87.48	87.28	86.30	86.23	87.05	86.33	89.56	88.34
Obloha	95.47	95.82	95.36	95.79	95.14	95.82	95.26	95.83	94.82	95.69	94.59	95.61
Lesní porost	51.95	52.52	51.35	53.40	46.96	47.74	47.68	48.69	51.24	56.48	47.09	49.46
Voda	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Ledovec	0.00	0.00	-	-	0.00	0.00	-	-	0.00	0.00	-	-
IU												
Hory	54.47	53.61	54.54	54.62	53.06	53.37	52.76	53.11	54.28	56.21	53.97	54.56
Obloha	94.99	95.37	94.99	95.42	94.48	95.25	94.39	95.20	94.13	95.13	94.00	95.08
Lesní porost	45.76	45.58	45.51	46.66	41.56	42.23	41.87	42.76	45.12	49.54	42.40	44.11
Voda	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Ledovec	0.00	0.00	-	-	0.00	0.00	-	-	0.00	0.00	-	-

Tabulka C.18: Výsledky segmentace datové sady Venturi pomocí dotrénovaných modelů v % - třetí část.

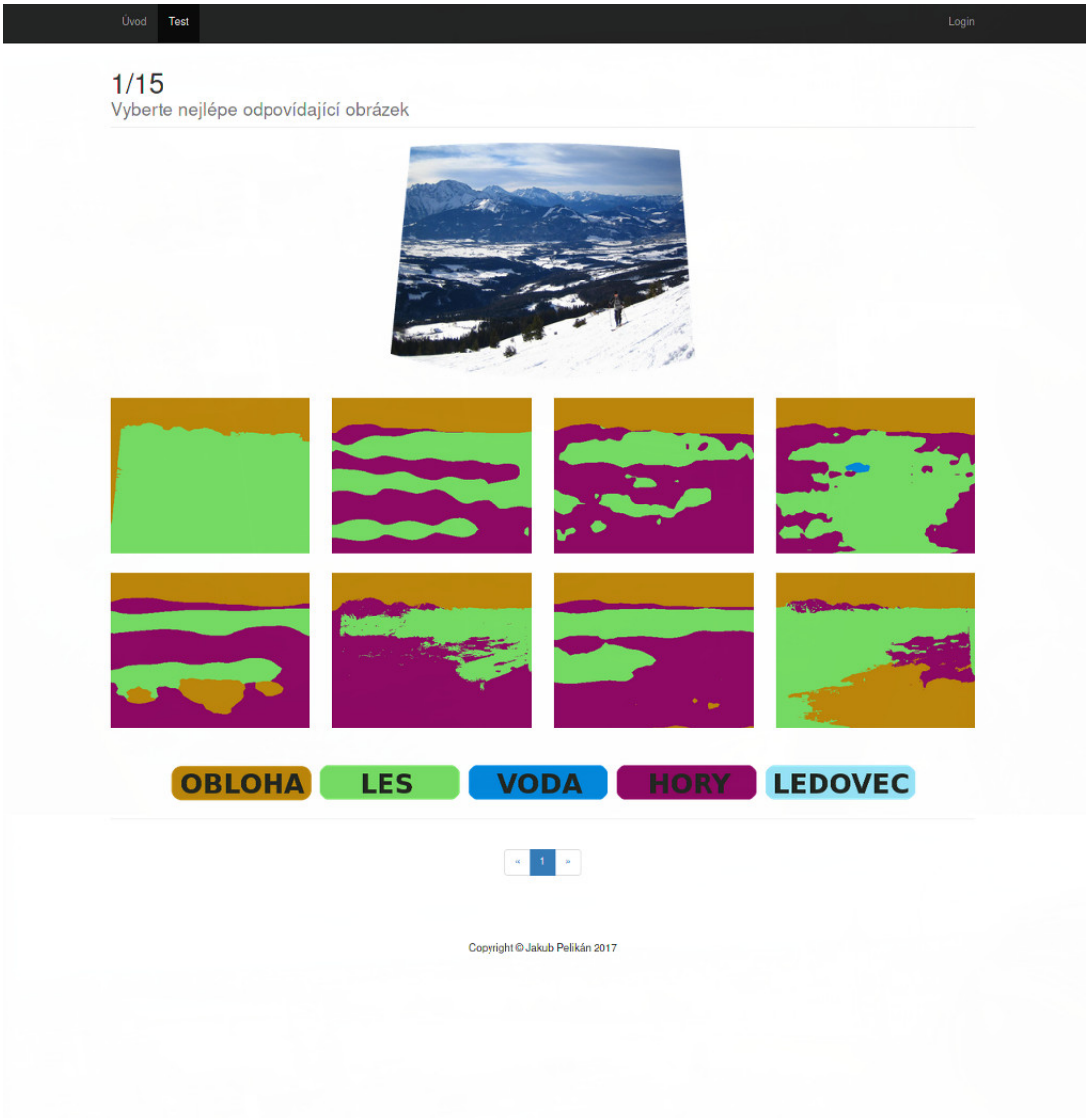
Příloha D

Ukázka dotazníku

D.1 Úvodní strana

Obrázek D.1: Úvodní strana dotazníku obsahující základní informace, cíl dotazníku a grafickou ilustraci segmegmentace s příklady odpovídajících a neodpovídajících segmentů (nav. diagram 6).

D.2 Ukázka výběru segmentů



Obrázek D.2: Ukázka otázky dotazníku, kde respondenti vybírají jeden z 8 segmentů, který podle nich nejlépe odpovídá originálnímu obrázku (nav. diagram 6).

Příloha E

Chyba odhadu orientace

E.1 AUC

AUC (%)	Nízké rozlišení		Vysoké rozlišení	
Kombinace s hranami	NE	ANO	NE	ANO
ALE	78.7	82.6	80.2	82.7
siftflow-fcn8s (pův. klas.)	82.4	83.0	80.3	84.6
siftflow-fcn8s (vl. klas. - 4 třídy)	79.6	80.9	80.2	82.0
siftflow-fcn8s (vl. klas. - 5 tříd)	80.1	82.9	80.8	84.1
DL-COCO-LargeFOV	80.0	-	81.6	-
DL-COCO-LFOV (CRF)	80.8	-	81.4	-
DL-v2-VGG16	80.2	80.9	82.1	84.6
DL-v2-VGG16 (CRF)	80.7	81.1	83.3	85.1
DL-v2-VGG16 (5 tříd)	79.9	82.9	82.2	86.9
DL-v2-VGG16 (5 tříd - CRF)	80.1	82.1	82.9	86.3

Tabulka E.1: Výsledky použití dotrénovaných modelů v procesu odhadu orientace kamery vyjádřené pomocí plochy pod křivkou grafu (AUC), který reprezentuje chybu orientace v datové sadě (nav. diagram 7).

E.2 Průměrná chyba orientace kamery

Chyba orientace (°)	bez ledovce	s ledovcem	celkové	bez ledovce	s ledovcem	celkové
Kombinace s hranami	NE	NE	NE	ANO	ANO	ANO
Nízké rozlišení						
ALE	70.0	61.6	66.4	58.8	48.3	54.3
siftflow-fcn8s (pův. klas.)	57.5	51.3	54.8	50.2	56.6	52.9
siftflow-fcn8s (vl. klas. - 4 třídy)	65.0	61.8	63.6	56.7	63.0	59.4
siftflow-fcn8s (vl. klas. - 5 tříd)	69.0	52.6	61.9	60.8	43.6	53.4
DL-COCO-LargeFOV	68.3	54.2	62.3	-	-	-
DL-COCO-LFOV (CRF)	66.6	50.8	59.8	-	-	-
DL-v2-VGG16	57.2	68.0	61.8	58.1	61.4	59.5
DL-v2-VGG16 (CRF)	63.8	55.3	60.2	58.1	60.2	59.0
DL-v2-VGG16 (5 tříd)	59.3	67.4	62.8	54.1	52.5	53.4
DL-v2-VGG16 (5 tříd - CRF)	67.9	53.9	61.9	59.9	50.3	55.8
Vysoké rozlišení						
ALE	67.4	54.1	61.7	59.2	46.9	53.9
siftflow-fcn8s (pův. klas.)	61.8	60.8	61.4	46.0	50.5	47.9
siftflow-fcn8s (vl. klas. - 4 třídy)	67.6	53.6	61.6	60.7	49.9	56.0
siftflow-fcn8s (vl. klas. - 5 tříd)	67.8	49.0	59.8	59.3	36.7	49.6
DL-COCO-LargeFOV	64.2	48.4	57.4	-	-	-
DL-COCO-LFOV (CRF)	67.7	45.1	58.0	-	-	-
DL-v2-VGG16	54.2	58.1	55.9	47.7	48.1	47.8
DL-v2-VGG16 (CRF)	50.6	53.8	51.9	45.6	47.4	46.4
DL-v2-VGG16 (5 tříd)	55.1	55.7	55.4	41.8	39.7	40.9
DL-v2-VGG16 (5 tříd - CRF)	56.6	48.7	53.2	44.4	40.6	42.8

Tabulka E.2: Výsledky použití dotrénovaných modelů v procesu odhadu orientace kamery vyjádřené pomocí průměrné chyby orientace (nav. diagram 7).

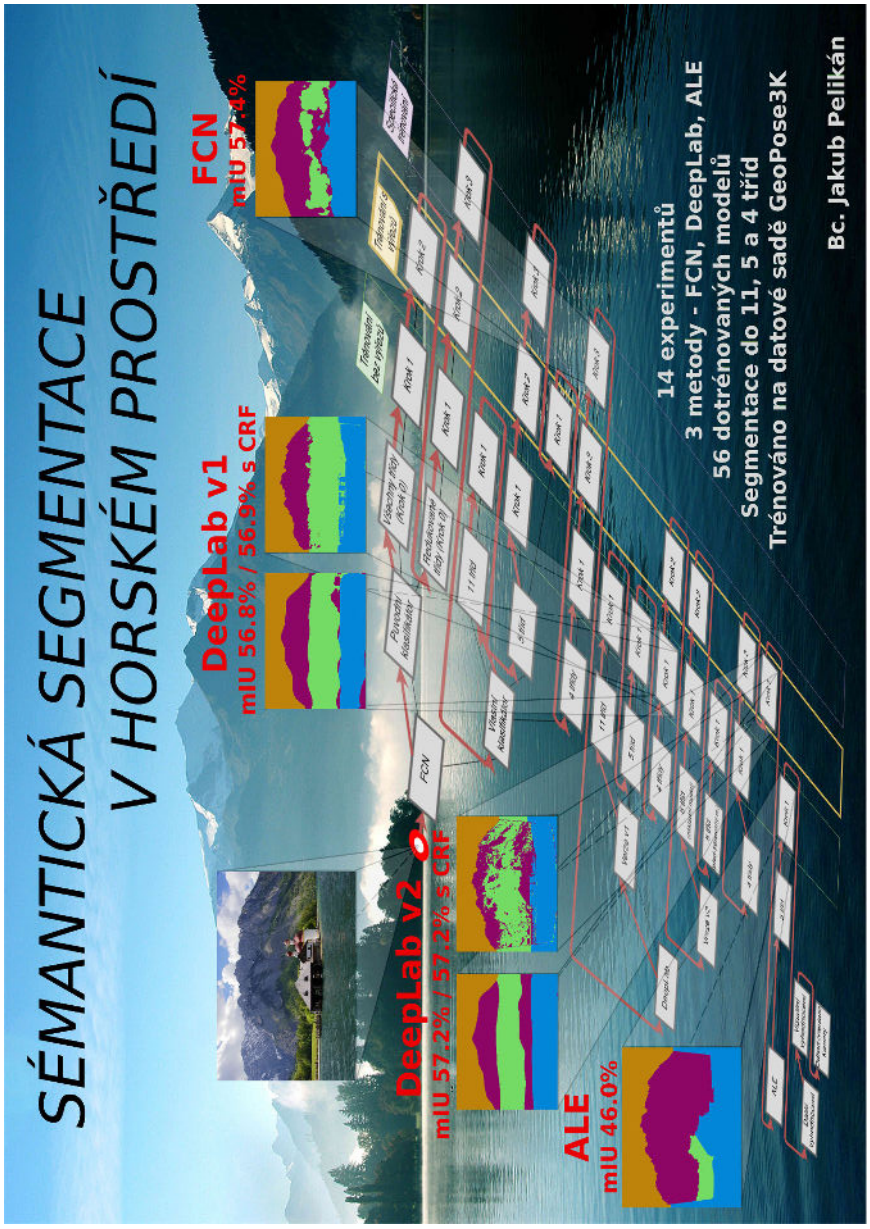
E.3 Průměrná chyba orientace kamery pro nejčastější kombinace tříd

Chyba orientace (°)	HOL		HOV		HOD		HOLV		HOLD		HOVD		HOLVD	
Kombinace s hranami	NE	ANO	NE	ANO	NE	ANO	NE	ANO	NE	ANO	NE	ANO	NE	ANO
Nízké rozlišení														
ALE	71.0	58.1	19.1	79.7	44.3	39.5	71.3	52.2	70.8	51.3	53.4	54.1	73.8	53.9
siftflow-fcn8s (pův. klas.)	57.0	47.4	59.3	86.3	56.6	77.0	55.6	44.2	45.3	49.3	50.1	74.3	51.3	37.2
siftflow-fcn8s (vl. klas. - 4 tř.)	70.5	59.9	40.7	68.0	60.0	82.9	58.8	44.3	54.1	38.3	69.0	85.5	69.2	59.4
siftflow-fcn8s (vl. klas. - 5 tř.)	72.1	61.2	58.4	81.6	47.4	48.3	62.3	50.6	50.5	38.2	56.7	50.9	59.3	42.0
DL-COCO-LargeFOV	72.0	-	54.1	-	56.9	-	62.8	-	44.6	-	61.5	-	58.7	-
DL-COCO-LFOV (CRF)	67.5	-	69.7	-	53.7	-	65.3	-	42.4	-	35.8	-	58.6	-
DL-v2-VGG16	54.6	59.7	41.9	65.4	63.9	59.0	65.1	58.0	69.2	55.2	49.7	62.7	75.2	69.5
DL-v2-VGG16 (CRF)	63.9	56.9	56.7	73.3	58.3	80.8	62.2	49.7	49.7	40.7	50.9	96.6	58.3	47.8
DL-v2-VGG16 (5 tříd)	61.4	57.1	59.4	75.2	63.6	46.2	59.2	49.0	71.8	51.2	64.5	66.6	68.0	57.4
DL-v2-VGG16 (5 tříd - CRF)	65.4	61.3	71.1	80.5	55.0	54.6	69.9	47.4	49.5	39.5	54.1	84.4	56.9	48.4
Vysoké rozlišení														
ALE	70.4	61.2	45.6	60.7	35.8	45.3	60.2	47.7	60.5	42.7	62.0	60.6	66.6	49.7
siftflow-fcn8s (pův. klas.)	57.7	46.7	90.3	80.2	73.6	64.7	63.2	35.2	51.7	49.2	59.5	67.7	55.5	32.4
siftflow-fcn8s (vl. klas. - 4 tř.)	75.9	68.0	60.6	73.9	43.5	52.9	56.1	45.2	55.3	45.0	44.4	68.3	65.2	47.1
siftflow-fcn8s (vl. klas. - 5 tř.)	71.3	58.9	38.0	75.6	40.4	37.8	61.9	52.1	52.2	38.3	42.0	42.2	57.0	32.8
DL-COCO-LargeFOV	70.6	-	40.9	-	41.6	-	55.4	-	46.3	-	26.1	-	62.6	-
DL-COCO-LFOV (CRF)	71.0	-	39.9	-	32.6	-	65.0	-	47.2	-	10.3	-	64.1	-
DL-v2-VGG16	51.2	48.5	27.6	75.7	50.2	44.0	63.6	44.8	65.3	47.8	22.9	50.1	67.7	52.4
DL-v2-VGG16 (CRF)	47.7	45.3	42.4	69.3	53.1	46.7	57.6	44.8	60.2	42.1	23.0	46.1	55.2	53.4
DL-v2-VGG16 (5 tříd)	54.7	45.8	54.8	54.1	50.0	34.4	56.9	37.0	61.6	40.0	20.4	50.7	63.9	42.9
DL-v2-VGG16 (5 tříd - CRF)	56.6	46.6	53.0	62.7	43.8	34.3	58.7	40.6	44.6	38.9	10.0	38.1	66.1	49.6

Tabulka E.3: Výsledky použití dotrénovaných modelů v procesu odhadu orientace kamery vyjádřené pro jednotlivé kombinace tříd ve zdrojových obrázcích pomocí průměrné chyby orientace. Kombinace tříd je popsána pomocí zkratek názvů tříd(H - **H**ory, O - **O**bloha, V - **V**oda, D - le**D**ovec, L - **L**esní porost) (nav. diagram 7).

Příloha F

Plakát



Obrázek F.1: Plakát demonstrující výsledek práce. Pro vytvoření plakátu byly použity fotografie z datové sady GeoPose3K [11].