



ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

**Fakulta stavební
Katedra geomatiky**

**Mobilní laserové skenování v dopravě
Mobile Laser Scanning in Transportation**

DISERTAČNÍ PRÁCE

Ing. Martina Hůlková

Doktorský studijní program: Geodézie a kartografie

Studijní obor: Geodézie a kartografie

Školitel: prof. Dr. Ing. Karel Pavelka

Praha, 2017

Prohlášení

Prohlašuji, že jsem tuto doktorskou disertační práci vypracovala samostatně pod vedením školitele prof. Dr. Ing. Karla Pavelky.

Použitou literaturu a další materiály uvádím v seznamu použité literatury.

V Praze dne

Ing. Martina Hůlková

Poděkování

Ráda bych poděkovala svému školiteli a kolegům za poskytnuté rady a zpětnou vazbu, Evě Matouškové a Vojtovi Hronovi za korektury a v neposlední řadě také celé rodině (zejména manželovi) za podporu a trpělivost.

„Když jste schopni se něco naučit, dokážete přijít i na to, jak.“

Terry Pratchett

Abstrakt

Vývoj technologie laserového skenování v průběhu posledních dvou dekad umožnil tvorbu skenovacích systémů adaptovatelných na množství nejrozličnějších nosičů (automobilů, lodí, bezpilotních leteckých prostředků a dalších). Rozmanitost těchto nosičů společně s možností propojit laserový skener s dalšími sensory přispívá k širokému uplatnění skenovacích systémů - od mapování říčních koryt až po dokumentaci měst. Mobilní skenovací systémy využívající jako nosič automobil se osvědčují v dokumentaci pozemních komunikací a uličních sítí. Současné možnosti a aplikace mobilního laserového skenování, s důrazem na aplikace v dopravní infrastruktuře, shrnuje řešeršní část práce.

Dokumentace pozemních komunikací a jejich nejbližšího okolí je aktuálně řešeným problémem a tématem mnoha odborných článků. Stávající systémy umožňují pořizování enormního množství prostorových dat. Vysoké nároky jsou poté kladeny na jejich rychlé a efektivní zpracování, tzn. na vysokou míru automatizace tohoto procesu. Většina vědeckých publikací na téma klasifikace mračen bodů z MLS se zabývá pouze extrakcí jednoho konkrétního prvku zájmu pro potřeby specifické analýzy (vodorovného, nebo svislého dopravního značení, lamp pouličního osvětlení a pod). Přitom efektivní a ekonomické by bylo širší využití získaného datového souboru - pro porovnávání stavu v různých časových epochách, analýzy bezpečnosti, viditelnosti, stavu vozovky, stavu vegetace apod. Toto širší využití vyžaduje komplexní nástroj, který umožní rozdělit mračno do několika obecných tříd. Takový nástroj provede první hrubou klasifikaci mračna a připraví tak základ pro provádění dalších analýz dle konkrétní okamžité potřeby. Specifičtější a sofistikovanější nástroje pak již mohou pracovat pouze s body vybrané třídy, čímž se sníží množství testovaných dat a náročnost analýz.

Předkládaná práce si dala za cíl takový efektivní nástroj pro využití na dálnicích a rychlostních komunikacích navrhnout. Byl vytvořen a otestován algoritmus provádějící klasifikaci mračna bodů prostřednictvím rozdělení mračna do pravidelných voxelů a vytvořením čtyřrozměrné matice, jejíž první tři rozměry odkazují na polohu voxelu a čtvrtý rozměr na parametry reprezentující body spadající do daného voxelu. Z vybraných hodnot této matice jsou vytvářeny rastrové obrazy, na které jsou aplikovány postupy ke zpracování obrazových dat s cílem vybrat pixely obsahující body hledané třídy. Výsledky klasifikace jsou pak přeneseny z rastrových dat zpět na mračno bodů. Základní princip metody – tedy převod klasifikace do rastrových dat a řešení úloh na úrovni pixel-pixel místo bod-bod přináší výrazné snížení náročnosti úlohy a vysokou efektivitu, kterou je možno vyvážit mírou chybovosti algoritmu danou principem metody.

Funkčnost metody byla ověřena na vybraném vzorku dat (úseku dálnice D11 o délce 1,8km obsahujícím všechny typy objektů, jejichž klasifikace byla v metodě řešena) s celkovou správností klasifikace 94,5% (správně klasifikovaných bodů z celkového počtu bodů).

Klíčová slova

mračno bodů, mobilní laserové skenování, pozemní komunikace, klasifikace, rastrový obraz

Abstract

Last two decades of technology development brought to reality laser scanning systems adaptable to various platforms (cars, boats, unmanned aerial vehicles and others). The platform variability together with possibility of linking to other sensors lead to wide system application – from fluvial mapping to city modelling. Laser scanning systems mounted on car are used for road documentation and city street mapping. The technique and mobile laser scanning applications state of art, with special regard to road documentation, are summarized in retrieval section of presented work.

Many studies dealing with road corridors mapping have been presented. Mobile laser scanning systems can perform detailed road documentation. Hand in hand with enormous datasets acquired by these systems, demands on fast and effective dataset processing increases, especially on the point cloud processing automation. Most of presented work still focuses on sole object type detection for specific analysis (road markings, vertical traffic signs, trees etc.). However, the acquired datasets can be used for change detection, safety analysis, and analysis of visibility, roadway condition, vegetation and many other purposes. An extraction tool for all objects in the scene is needed for efficient and wide use of datasets; a tool for rough classification into several basic classes, which serves as a background for further sophisticated analysis.

The aim of presented work was to develop a method of highway scene classification. Proposed method transforms the point cloud classification problem into image classification problem, which enables application of image processing techniques for the classification and reduces the computational demands of spatial queries. The point cloud data are divided into voxels and a matrix summarizing information about individual voxels is created to serve as an intermediate between point cloud and raster image. The voxel matrix is a four dimensional matrix – the first three dimensions refer to the position of individual voxels and the last dimension refers to parameters describing points which belong to corresponding voxel. This voxel matrix serves as a source for raster image creation. Selection of pixels representing points of each class is done by applying image processing techniques on raster images. In the last step, the results of image classification are applied to the point cloud.

Presented method was tested on 1.8 km long part of a D11 highway located in central and eastern Bohemia, Czech Republic. The automatic classification achieved the overall accuracy of 94.5%.

Key words

point cloud, mobile laser scanning, roadway, classification, raster image

Obsah

Seznam použitých zkratk	4
1 Úvod	5
1.1 Hypotéza	5
1.2 Cíle práce	6
1.3 Struktura práce	6
2 Rešerše	7
2.1 Specifika dat z mobilního laserového skenování	7
2.2 Příklady systémů	8
2.3 Příklady aplikací	10
2.4 Zpracování dat mobilního laserového skenování	11
2.4.1 Obecný postup zpracování	11
2.4.2 Filtrace terénu	11
2.4.3 Extrakce a rozpoznávání objektů	12
2.5 Klasifikace mračna bodů pomocí komerčního řešení	16
2.5.1 Obecný postup	17
2.5.2 Čištění mračna	17
2.5.3 Klasifikace terénu	18
2.5.4 Funkce ke klasifikaci	18
2.5.5 Použití kombinace funkcí	19
2.6 Shrnutí současného stavu a motivace	20
2.6.1 Význam práce	20
3 Data	21
3.1 Datový soubor	21
3.1.1 Projekt CG912-105-520	21
3.1.2 Sběr dat a předzpracování	21
3.1.3 LYNX Mobile Mapper	22
3.2 Výběr a příprava dat pro testování	24
3.2.1 Rozdělení do bloků	24
3.2.2 Výběr bloků pro experiment	24
3.2.3 Klasifikace terénních bodů a přepočítání výšky	25

4	Experiment.....	27
4.1	Koncept.....	27
4.1.1	Třídy.....	28
4.1.2	Nástroje.....	30
4.2	Tvorba rastru.....	30
4.2.1	Vstupní data.....	30
4.2.2	Postup.....	31
4.3	Klasifikace rastrového obrazu.....	33
4.3.1	Roadway - vozovka	33
4.3.2	Crash barriers – svodidla	39
4.3.3	Vehicles – vozidla	44
4.3.4	Gates – objekty překlenující komunikaci	46
4.3.5	Signs – svislé dopravní značky a jiné tabule	51
4.3.6	Walls – zdi.....	57
4.3.7	Poles – tyčové objekty	62
4.3.8	Vegetation - vegetace	65
4.4	Přenos výsledků klasifikace z rastru do mračna bodů.....	66
5	Ověření metody	70
5.1	Postup.....	70
5.2	Výsledky.....	71
5.3	Příklady chyb v klasifikaci	73
6	Shrnutí	80
6.1	Výsledky.....	80
6.2	Závěry	81
6.3	Diskuse	81
	Literatura	84
	Přílohy.....	88
1	Seznam obrázků	88
2	Seznam tabulek	91
3	Seznam příloh na CD	92
4	Kód “twoZcoord.m“	93

5	Kód "pcloud2image.m"	94
6	Kód "image2pcloud.m"	105
7	Kód "ClassQualCheck.m"	107

Seznam použitých zkratk

DIA	<i>Direct Inertial Aiding</i>
DMP	digitální model povrchu
DMT	digitální model terénu
DPZ	dálkový průzkum Země
ETRS89	Evropský terestrický referenční systém 89
FWF	<i>Full-waveform</i>
GIS	geografický informační systém
ICP	<i>Iterative Closest Point</i>
GNSS	<i>Global Navigation Satellite Systems</i>
GPS	<i>Global Positioning System</i>
IMU	<i>Inertial Measurement Unit</i>
LIDAR	<i>Light Detection and Ranging</i>
LLS	letecké laserové skenování
MLS	mobilní laserové skenování
RANSAC	<i>Random Sample Consensus Algorithm</i>
RPAS	<i>Remotely Piloted Aircraft System</i>
ROI	<i>Region of Interest</i>
RTK	<i>Real-time Kinematic</i>
S-JTSK	Souřadnicový systém Jednotné trigonometrické sítě katastrální
TIN	<i>Triangulated Irregular Network</i>
TLS	terestrické laserové skenování
WGS 84	<i>World Geodetic System 84</i>

1 Úvod

Obrovský pokrok techniky v druhé polovině 20. století vytvořil podmínky k sestrojení laserových skenerů – přístrojů umožňujících pořizování enormního množství prostorových dat ve formě hustých mračen bodů. Tyto přístroje se během let staly běžnou součástí geodetické praxe. Propojení laserových skenerů s dalšími sensory jako jsou GNSS přijímače, akcelerometry, tachometry a dalšími poté umožnilo vytvoření skenovacích systémů, které je dnes možno umístit na množství různorodých nosičů pohybujících se po zemi, na vodě i ve vzduchu. Možnosti dokumentace objektů a mapování krajiny pomocí laserového skenování se tak zdají téměř neomezené. Milionová mračna bodů je dnes možné pořizovat během několika okamžiků, přičemž ukládání a zpracování takového množství dat klade velké nároky na hardwarové a softwarové vybavení. Důležitou roli hraje rychlost a ekonomická efektivita, a protože nejvíc času při laserovém skenování zabere právě zpracování dat, klade se velký důraz na automatizaci procesů zpracování. Stávající softwarová řešení jsou schopna extrahovat z mračen bodů potřebné prostorové informace s různou mírou úspěšnosti pro různé aplikace, manuální kontrola a opravy zpracovatelem však zůstávají nezbytností. Cílem lidí podílejících se na vývoji algoritmů pro extrakci požadovaných prostorových informací a jejich prezentaci je omezit manuální práce na datových souborech na minimum.

Předkládaná práce se zabývá aplikací mobilního laserového skenování (MLS) v dopravě, resp. ve správě dopravní infrastruktury. Mobilní laserové skenovací systémy využívající jako nosiče automobil se již stávají standardem pro dokumentaci stavu pozemních komunikací a jejich bezprostředního okolí (Olsen, 2013). Jako u jiných aplikací je i v dokumentaci pozemních komunikací esenciální extrakce jednotlivých zájmových objektů z mračen bodů, přičemž komplexní řešení dosud neexistuje. Zpracováním mračen bodů z pozemních komunikací za použití dostupného softwaru se autorka zabývala již ve své diplomové práci (Faltýnová, 2010). Cílem této práce je navrhnout a otestovat vlastní přístup ke klasifikaci mračen bodů pro potřeby dokumentace pozemních komunikací a přispět tím k nalezení optimálního řešení.

1.1 Hypotéza

Práce je založená na následující hypotéze:

- Metody využívané k segmentaci a klasifikaci obrazových dat lze aplikovat na data mobilního laserového skenování.

Přístupy ke klasifikaci mračen bodů pracující přímo s mračnem bodů (ať už organizovaným nebo neorganizovaným) jsou často výpočetně náročné. Převedení mračen bodů do rastrového obrazu v určité fázi zpracování redukuje množství testovaných dat a umožňuje využití výpočetně méně náročných metod pro segmentaci a klasifikaci obrazových dat, čímž se sníží celková výpočetní náročnost klasifikace těchto dat. Tento postup byl s úspěchem využit např. pro segmentaci dat z pozemního laserového skeneru (Barnea a Filin, 2008).

1.2 Cíle práce

Cílem práce je navrhnout metodu pro využití obrazové segmentace a klasifikace ke klasifikaci mračen bodů z mobilních skenovacích systémů s ohledem na využití metody při správě pozemních komunikací. Dílčí cíle lze charakterizovat následovně:

- stanovení objektů/prvků zájmu, které je třeba klasifikovat
- stanovení charakteristik jednotlivých tříd, na základě kterých je možné oddělit je od ostatních objektů a využití těchto charakteristik pro tvorbu rastrových vrstev
- výběr a testování vhodných metod zpracování obrazových dat vedoucích ke klasifikaci objektů stanovených tříd
- převod výsledků klasifikace rastrů na mračno bodů
- ověření metody na vybraných datových sadách
- zhodnocení kvality výsledků a využitelnost v praxi

1.3 Struktura práce

Předkládaná práce je členěna do následujících kapitol:

Kapitola Rešerše představuje specifika dat MLS, popisuje aplikace, ve kterých nachází MLS uplatnění, popisuje současné možnosti zpracování těchto dat a na základě shrnutí současného stavu řešené problematiky vysvětluje vznik hypotézy, jejímž ověřením se tato práce zabývá.

Kapitola Data obsahuje popis pořízení a předzpracování datového souboru využitého pro vývoj a ověření představované metody.

Kapitola Experiment se zabývá podstatou této práce, tedy vlastní tvorbou metody klasifikace.

Kapitola Ověření metody popisuje způsob ověření úspěšnosti metody na vybraném vzorku dat a jeho výsledek.

Kapitola Shrnutí hodnotí dosažené výsledky a navrhuje možný směr dalšího výzkumu.

2 Rešerše

Vzhledem k problematice, kterou se předkládaná práce zabývá, tedy zpracování dat z MLS a jejich využití při dokumentaci pozemních komunikací, se rešerše snaží postihnout, jak se lze se zpracováním dat vyrovnat při různých aplikacích. Podrobné informace o principu skeneru, složení systémů a dalším lze nalézt např. ve *Vosselman a Maas (2010)* nebo *Shan a Toth (2009)*.

2.1 Specifika dat z mobilního laserového skenování

Na trhu již existuje řada softwarů určených ke zpracování dat laserového skenování zejména pro jeho terestrickou a leteckou formu. S masivnějším rozvojem mobilních systémů se objevují nástroje zaměřené i na data z mobilních laserových skenovacích systémů, ale výzkum v oblasti zpracování těchto dat ukazuje, že tu zůstává prostor pro zlepšování.

Data z různých typů skenovacích systémů mají různé charakteristiky (viz níže) a očekáváme z nich různé výstupy, což znemožňuje použití jednoho univerzálního nástroje pro jejich zpracování. Data z leteckých systémů jsou zpravidla rovnoměrně rozložena v ploše, a proto se svým charakterem hodí k převodu na tzv. 2,5D data (výškové a jiné rastrové obrazy). Za cenu ztráty části dat je umožněno využití efektivních nástrojů ke zpracování obrazu. U dat terestrických skenerů je opět v určité fázi využíván převod na 2,5D data – tentokrát na *range image* – hloubkový obraz (obraz ve kterém barva pixelu odpovídá vzdálenosti od skeneru), toho využívají např. *Barnea a Filin, (2008)*.

Data mobilního laserového skenování mají na rozdíl od dat leteckého laserového skenování (LLS) výrazně různou hustotu bodů v závislosti na vzdálenosti od skeneru, nejsou také tak rovnoměrně rozložena. Oproti datům z terestrických systémů jsou data MLS rozložena podél trajektorie nosiče, ne soustředěná kolem jednoho bodu. Právě jediné společné stanovisko, odkud jsou měřeny všechny body mračna, umožňuje snadné promítnutí bodů na plochu válce a její rozvinutí do roviny při tvorbě hloubkového obrazu. Použití hloubkového obrazu v obdobné podobě je možné jen u mobilních skenerů pořizujících data v řádcích kolmých ke směru jízdy (válec, na který jsou body promítnuty je v takovém případě v horizontální poloze s osou v trajektorii referenčního bodu skeneru), takový typ mobilního skenovacího systému je popsán např. v *Rao a kol. (2006)*. Mobilní systémy jsou využívány k mapování mnohem rozsáhlejších a komplexnějších oblastí než terestrické systémy a zároveň s mnohem vyšší hustotou bodů než data leteckých systémů.

Mimo zdroje dat je pro způsob zpracování zásadní také to, jakou formu má mít výsledný výstup. Terestrické skenery se používají např. k mapování nepravidelných povrchů, jako jsou sochy nebo jiné menší objekty, jejichž výstupem je trojrozměrný model ve formě trojúhelníkové sítě. V případě použití terestrických skenerů pro reverzní inženýrství nebo mapování průmyslových areálů je očekáván výstup ve formě modelu z geometrických primitiv, v případě mapování staveb může být požadován výstup ve formě drátěného modelu (vektORIZOVANÝCH HRAN). Výstupem dat LLS může být digitální model terénu - DMT (ve formě nepravidelné trojúhelníkové sítě - TIN, pravidelného gridu...) doplněný např. o vektorizované hrany střešních plášťů budov a hranicemi ploch lesa. V případě dat MLS může být požadován DMT pozemní komunikace, nebo pouze klasifikované mračno bodů.

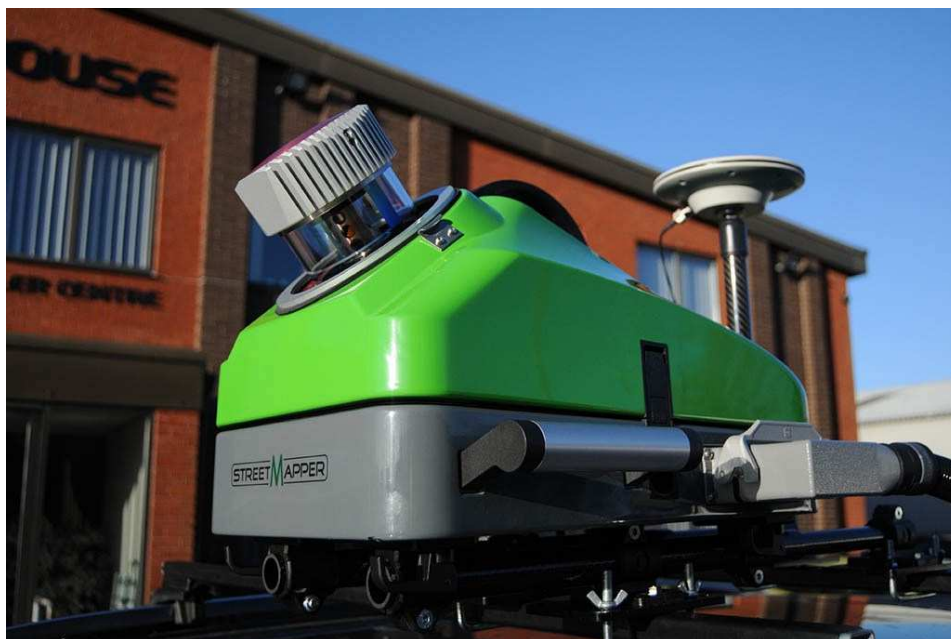
Z výše popsaných charakteristik dat MLS je zřejmé, že algoritmy vyvinuté pro zpracování dat terestrických ani leteckých systémů, nejsou pro aplikace MLS ideální. Přesto je možné pro některé účely tyto softwary použít. Znalost postupů běžně používaných pro práci s daty leteckého a terestrického skenování je při vývoji nových metod pro MLS nezbytná, protože tyto osvědčené postupy lze v modifikované podobě často úspěšně aplikovat.

2.2 Příklady systémů

Skenovací systémy je možné za pomoci vhodné platformy umístit téměř na jakýkoli pohybující se objekt. Jako nosiče se běžně používají auta, lodě, lokomotivy nebo letadla. Systémy pro letecké laserové skenování bývají i přes množství podobných charakteristik s ostatními pohyblivými skenovacími systémy vyčleněny do zvláštní skupiny, proto se jimi tato práce podrobněji nezabývá. Následuje několik příkladů skenovacích systémů využívajících různé nosiče.

Street Mapper IV (IGI;3D Laser Mapping, Německo/Velká Británie)

- systém nesený automobilem pro skenování liniových staveb a uliční sítě měst, obsahuje GNSS přijímač, IMU a volitelně DIA+ (*Direct Inertial Aiding* – zařízení umožňující zvýšit rychlost nosiče při skenování a vypořádat se s krátkodobou ztrátou signálu GNSS). Mimo laserového skeneru obsahuje systém také panoramatickou kamerou. *Street Mapper* (ve své starší verzi) byl v roce 2004 prvním dostupným komerčním systémem pro MLS.



Obr. 1 Street Mapper IV (<http://www.3dlasermapping.com>)

Lynx HS-600 (Teledyne Optech, Kanada)

- systém existující v jedno nebo dvou senzorové verzi, registrace až 4 ech, přídavné kamery, rychlost snímání skeneru – až 600 řádků/s

Merlin (Renishaw, dříve MDL, Velká Británie)

- mobilní skenovací systém pro umístění na plavidle, určen pro mapování pobřeží, přístavů a vodních toků nad i pod hladinou, kompatibilní se systémy pro mapování dna pobřežních vod, dosah až 250m



Obr. 2 Merlin (<http://www.renishaw.com>)

GRP 5000 (Amberg Technologies, Švýcarsko)

- systém určený k pojezdu na kolejích, tlačení měřičem, obsahuje skener Amberg Profiler 5002, Leica GPS, odometr a další senzory
- dosah cca 80m, rychlost skenování - 500 000 bodů/s

ZEB CAM (3D Laser Mapping, Velká Británie)

- systém obsahující ruční skener o váze 1kg doplněný kamerou a příslušenstvím v batohu (cca další 3kg), nosičem skeneru je samotný měřič
- dosah do 30m, rychlost skenování - 43 200 bodů/s, přesnost relativní 2-3cm, absolutní 3-30cm (závisí na přesnosti skeneru a členitosti skenovaného okolí - např. skenování dlouhých chodeb je problematické)



Obr. 3 ZEB CAM (<http://www.3dlasermapping.com>)

2.3 Příklady aplikací

Při práci s daty MLS je třeba mít na zřeteli, jakým účelům mají data sloužit a jaká forma výstupu je očekávána. Množství aplikací laserového skenování obecně je popsáno ve *Vosselman, Maas a kol. (2010)* a *Shan a Toth (2009)*. Následují příklady aplikací pro různé nosiče mobilních skenovacích systémů.

Platforma nesená silničním vozidlem je nerozšířenější formou mobilního skenování, používá se k mapování uliční sítě, viz *Golovinskij a kol. (2009)*, pozemních komunikací a jejich okolí obecně, dále k mapování vedení vysokého napětí a jiných koridorů. Výstupem mohou být 3D modely měst (často vytvářené kombinací mobilního a leteckého skenování), modely terénu ve formě TIN doplněné o drátěné modely objektů kolem komunikace. Cílem může být určení parametrů pozemní komunikace, jako je šířka, sklon, výška obrubníků a dalších, viz *Denis a kol. (2010)*, případně segmentace a klasifikace objektů v okolí komunikace, např. svislého dopravního značení (*Pu a kol., 2011*), dále detekce poškození mostů a tunelů (*González-Jorge a kol., 2013*) a detekce osvětlení v tunelech (*Puente a kol., 2014*). MLS a LLS může být použito při zjišťování škod při přírodních katastrofách. V těchto situacích je důležitá zejména rychlost a proto představili *Hebel a Stilla (2008)* postup pro *on-the-fly* klasifikaci dat LLS, která pracuje s daty přímo ve formě skenovaných řádků a umožňuje provádět klasifikaci terénních bodů na místě.

Kolejové systémy se používají zejména k určení průjezdného profilu – tomu odpovídá i technologie skenování profilů kolmých ke směru jízdy (např. *Z+F Profiler 9012*) a výstupy ve 2D. Problematiku segmentace objektů z dat MLS železnic řeší např. *Luo, Jwa a Sohn (2014)*.

Systémy umístěné na lodi jsou využívány např. k dokumentaci eroze pobřeží, nebo monitorování vegetace v okolí řek (*Saarinen a kol., 2013*). Obvyklé senzory mohou být na lodi doplněny o sonar k určení průběhu terénu i pod vodní hladinou. Výstupem bývá detailní DMT.

Indoorové systémy jsou určeny k dokumentaci interiérů rozlehlých objektů, jako jsou letištní haly a objekty průmyslové výroby. Výstupem mohou být 3D modely objektů nebo výkresy půdorysu a řezů.

2.4 Zpracování dat mobilního laserového skenování

2.4.1 Obecný postup zpracování

Algoritmy pro zpracování dat laserového skenování popsal už *Axelsson (1999)*, v dalších letech bylo vyzkoušeno množství různých přístupů pro data z různých typů skenovacích systémů a v *Pu a kol. (2011)* byl navržen následující obecný postup zpracování datových souborů MLS: v první řadě se datové soubory dělí na menší části (kvůli objemu a rychlosti zpracování dat). Za nejvhodnější označují autoři dělení bloků zvolené délky a šířky podél trajektorie skeneru. Dělení podél os x , y se nedoporučuje, protože nese vyšší riziko dělení objektů zájmu na těžko klasifikovatelné části. Následuje segmentace dat v jednotlivých blocích a klasifikace do tří základních tříd (*ground*, *on-ground*, *off-ground*). Body třídy *ground* odpovídají odrazům od terénu, třída *on-ground* zahrnuje všechny objekty spojené s terénem a *off-ground* v sobě zahrnuje odlehlé body, zpravidla hrubé chyby v měření. Dalším krokem je rozlišení objektů do jednotlivých tříd našeho zájmu (např. vegetace, budovy, dopravní značení).

2.4.2 Filtrace terénu

Jedním z prvních a zpravidla zásadních kroků při zpracování dat MLS je klasifikace terénu. *Axelsson (1999)* označuje klasifikaci terénních bodů pojmem *filtering* (filtrování). Množství metod k výběru terénních bodů je shrnuto také v *Vosselman, Maas a kol. (2010)*, *Shan a Toth (2009)*. V *Sithole a Vosselman (2004)* je provedeno testování osmi různých algoritmů na filtrování dat LLS. Další konkrétní algoritmy jsou popsány např. v *Chang a kol. (2008)*, *Filin a Pfeifer (2006)* a *Hebel a Stilla (2008)*. Existuje již i několik algoritmů vyvíjených čistě pro data MLS, např. *Jaakkola a kol. (2008)*, *Rao a kol (2006)*.

Sithole a Vosselman (2004) charakterizují filtry sedmi základními prvky:

- strukturou dat (surové mračno, rastrový obraz...),
- způsobem testování okolí a množstvím testovaných bodů v jednom kroku (funkce porovnávající bod-bod, bod-body, body-body),
- způsobem měření diskontinuit (výškový rozdíl, sklon, vzdálenost k plošce TIN, vzdálenost k jinak definovanému povrchu),
- konceptem filtru (*slope-based*, *block-minimum*, *surface-based*, *clustering/segmentation*),
- tím, zda je výpočet prováděn v jednom kroku, nebo iteračně,
- tím, zda jsou body označené jako neterénní vyřazeny (zpravidla algoritmy pracující s mračnem), nebo je jejich hodnota přepsána (zpravidla u dat ve formě rastrového obrazu),
- dle použití dodatečných informací (intenzita, barva, použití i prvních ech).

Úspěšnost filtrovacích algoritmů snižují podmínky, jako jsou:

- nízko položené odlehlé body v mračnu (méně jednotlivé, více skupiny),
- objekty, které jsou velmi malé, velmi velké, velmi nízké, složité (bloky budov),
- nesouvislý terén (dvory uvnitř bloků budov),
- objekty vnořené do terénu (budovy ve svahu, mosty),
- vegetace ve svahu, nízká vegetace,
- diskontinuity v terénu (strmé svahy, ostré hrany...).

Jak uvádí *Sithole a Vosselman (2004)* hodnocení úspěšnosti algoritmu lze provádět na základě množství chyb typu I. (terénní body chybně odfiltrované), chyb typu II. (body objektů označené za terénní) a celkového množství chybně klasifikovaných bodů. Automatická filtrace se porovnává se souborem filtrovaným manuálně, který je považován za správný. Při nastavování různých parametrů filtrů se ukázalo, že snaha o snížení počtu chyb typu I. vedla ke zvýšení chyb typu II. a obráceně. Cílem nastavení je vhodné vyvážení obou typů chyb.

Následuje popis základních metod filtrování terénu tak, jak je uvádí *Vosselman, Maas a kol. (2010)*:

Morfologické filtrování

Jednou z metod je morfologické filtrování založené na konceptu matematické morfologie. Matematická morfologie pracuje se dvěma základními operátory - erozí a dilatací. Kombinací těchto dvou operací, tzv. "zavírání" (eroze - dilatace) a "otvírání" (dilatace - eroze), může být použito pro určení minima a maxima v rámci určitého prvku. Morfologické filtrování může být vyjádřeno funkcí, která popisuje maximální přípustný výškový rozdíl Δh rámci prvku (povrchu) v závislosti na vzdálenosti d . Pro testovaný bod se vypočte vzdálenost a výškový rozdíl ke každému bodu v daném okruhu a porovná se s přípustným maximem.

Progresivní zhušťování

Metoda progresivního zhušťování začíná s malou podmnožinou předklasifikovaných terénních bodů a iterativně zvyšuje množství informací, díky kterým klasifikuje postupně celý datový soubor. Například při progresivním zhušťování nepravidelné trojúhelníkové sítě se vypočtou tři úhly mezi rovinou trojúhelníka a přímkou spojující každý ze tří vrcholů trojúhelníka s testovaným bodem a tyto úhly se porovnají s předdefinovanou hraniční hodnotou. Princip metody je uveden v *Axelsson (2000)*.

Metody založené na povrchu

Filtrovací metody založené na povrchu obvykle začínají s předpokladem, že všechny uvedené body patří k povrchu terénu a pak iterativně odstraňují nebo snižují vliv bodů, které neodpovídají modelu povrchu za postupného vylepšování tohoto modelu.

Segmentace

Metody, které patří do kategorie filtrování na základě segmentů, klasifikují celé segmenty, tj. soubory sousedních bodů s podobnými vlastnostmi. Pro segmentaci se obvykle používají metody narůstání plochy - *region growing*.

Některé postupy pro filtraci terénu nelze jednoduše zařadit, protože jde o kombinaci výše uvedených metod. Existuje tedy množství rozmanitých přístupů k řešení problému, a již *Sithole a Vosselman (2004)* ověřili poměrně velkou úspěšnost většiny z testovaných algoritmů.

2.4.3 Extrakce a rozpoznávání objektů

Mimo zjištění průběhu terénu, které se obvykle provádí nejdříve, bývá objektem zájmu při mobilním skenování množství dalších prvků. Jde o budovy, stromy, lampy, vodorovné a svislé dopravní značení a další. Extrahovat z mračna tyto jednotlivé prvky a vytvořit jejich model je cílem mnoha vyvíjených

algoritmů.

Zejména člověkem vytvořené objekty se často skládají z pravidelných ploch (rovinných, válcových...), což umožňuje jejich odlišení od vegetace. Rozpoznání těchto základních geometrických struktur bývá zásadní pro extrakci jednotlivých objektů. Shrnutí metod pro rozpoznávání geometrických struktur v mračnu bodů provádí *Vosselman a kol. (2004)*. Metody rozdělují do tří skupin – extrakce hladkých povrchů, iterativní extrakci rovin a přímou extrakci parametrizovaných tvarů. Extrakce hladkých povrchů je založena na shlukování sousedících bodů sdílejících nějakou vlastnost, jako lokálně odhadnutou normálu. Body jsou rozdělovány do skupin bodů reprezentujících jednotlivé povrchy. Proces může být realizován např. segmentací *scan lines*, nebo *surface growing* algoritmem. Iterativní extrakce rovin se od extrakce hladkých povrchů liší v zásadě tím, že požadavek na to, aby body odpovídaly rovině lokálně, změní na globální (pro celý segment). Přímá extrakce parametrizovaných tvarů je prováděna např. pomocí 3D Houghovy transformace (roviny, válců).

Dále jsou popsány příklady toho, jak se s extrakcí různých druhů objektů vyrovnávají různí autoři:

Wang a Tseng (2004) popisují segmentaci dat založenou na *octree* struktuře. Datový prostor je rozdělen do osmi stejně velkých podprostorů, data v podprostoru jsou testována, a pokud překročí daná kritéria (data nejsou rozložena podél roviny, nebo jsou v podprostoru víc jak tři body), prostor je dále dělen a znovu testován. Body je prokládána rovina, jejíž parametry jsou určovány pomocí metody nejmenších čtverců, pokud je suma odchylek bodů od roviny větší než přípustná hodnota, podprostor se dále dělí. Pro každé mračno v subprostoru, jemuž je přiřazena rovina, se vytvoří TIN a definují hranice rovinné plochy. V dalším kroku se provede spojení sousedních rovin, které mají dostatečnou shodu. Klasifikace se provádí na základě parametrů jednotlivých rovin – plocha, průměrná výška, gradient, průměrná intenzita, tvar apod.

Barnea a Filin (2008) využívají k segmentaci dat terestrického skeneru *range image* a obrazová data z přídavné kamery. *Range image* vzniká rozvinutím válcové plochy, na kterou jsou měřená data promítnuta. Poloha bodu v obraze a barva pixelu je dána polárními souřadnicemi bodu (poloha – horizontální a vertikální úhel, barva pixelu – vzdálenost bodu od skeneru). K segmentaci využívají tři obrazy – klasický *range image* (popsaný výše), obraz normál – místo vzdálenosti jsou vypočteny normály ze sousedních bodů (na ty je ještě aplikován filtr ke snížení šumu) a obraz s barvami odvozenými z doplňujících snímků. Jednotlivé obrazy jsou segmentovány na základě podobnosti a kritéria min. velikosti. Kombinace jednotlivých kanálů se provádí následovně: z provedené segmentace na jednotlivých kanálech se vybere největší segmentovaný objekt, ten je odstraněn ze všech kanálů, provede se přepočítání kalibrace na jednotlivých kanálech, je vybrán druhý největší segment atd. Proces se opakuje, dokud velikost vybraného segmentu neklesne pod definovanou hranici, nebo počet iterací nepřekročí daný limit. Dojde k tomu, že různé segmenty jsou vybrány na základě informací z kanálů, kde jsou nejkompaktnější (zemský povrch na základě normál, fasády na základě barvy apod.). Tento postup by bylo možné snadno aplikovat na mobilní skenovací systémy pořizující data v profilech kolmých ke směru jízdy.

Golovinskiy a kol. (2009) navrhli systém pro rozpoznávání objektů v prostorovém mračnu bodů v městských lokalitách. Nejprve vytvoří shluky bodů (potenciální místa objektů) hierarchickým

shlukováním, poté rozdělí body v těchto místech do segmentů v popředí a pozadí. Pro každý segment se vytvoří příznakový vektor a provede se řízená klasifikace. Metoda byla vytvářena a testována na vzorku dat města Ottawa (100 mil. bodů), tento vzorek pak byl použit jako trénovací objekty pro klasifikaci cca 1 bilionu bodů zbývající části města.

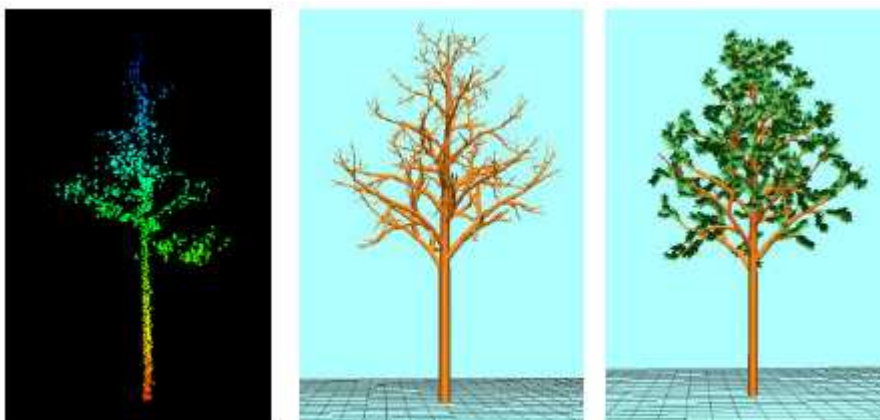
Vosselman a Liang (2009) se věnují detekci okrajů vozovky (obrubníků) vyznačujících se malými terénními skoky v datech LLS. K detekci využívají výškový obraz - jsou vybrány body do dané vzdálenosti od terénu, mezi kterými se nachází malý terénní skok, průměr mezi dvěma body (nízkým a vysokým) je označen za bod obrubníku. Body jsou prokládány liniemi a ty jsou spojovány do delších segmentů. Stejnou problematikou se zabývá i *Díaz-Vilarino a kol. (2015)*.

Denis a kol. (2010) představují postup pro extrakci hran obrubníků, určení jejich výšky a šířky vozovky z dat MLS. Vstupní data jsou mračna bodů získaná ze dvou skenerů umístěných na vozidle (mračna vpravo a vlevo od vozidla se zpracovávají zvlášť) a prostorový polygon definující osu vozovky (data pro GIS aplikace – s přesností 1m). Nejprve je provedena registrace osy do mračna bodů. Pro každý skenovaný řádek je vytvořen výškový histogram, body odpovídající nejnižšímu vrcholu histogramu jsou základem třídy terén. Tyto body jsou v řádce spojovány do krátkých řad a ty jsou dále shlukovány. V každém kroku iterace je lokálně proložena body rovina a všechny body příslušející této rovině jsou přidány do třídy terén, nově přidané body se použijí v dalším výpočtu. Protože je výpočet prováděn jen na malé ploše, může se snadno přizpůsobit malým nerovnostem terénu, ale bohužel i špatně klasifikovat nízké objekty – dolní části budov, nohy chodců apod. Detekce hran je založena na analýze výškového gradientu v mračnu terénních bodů. Je vytvořen obraz, ve kterém hodnota pixelu odpovídá min. výšce bodu klasifikovaného jako terén v dané buňce. Na obraz je použit Gaussův filtr ke snížení šumu. Poté jsou vypočteny gradienty Sobelovým filtrem a hrany jsou určeny hysterezním filtrem. Body podél hrany jsou rozděleny na nízké a vysoké a jsou mezi nimi vytvořeny páry, střední hodnota z každého páru je uložena jako obrubník.

Na principu vyhledávání malých terénních skoků pracuje i algoritmus, který představují *Yang a Fang (2014)*. Algoritmus pro automatickou extrakci kolejí pracuje s mračnem bodů rozřezaným na profily kolmé k trajektorii MLS. Profil prohledává pomocí pohyblivého se okna, v okamžiku, kdy je v okně detekován malý výškový rozptyl (terén železničního svršku), začne se profil prohledávat pomocí tří po sobě jdoucích oknech a hledá se výškový rozdíl v následujících oknech (rozdíl železniční svršek x koleje).

Rutzinger a kol. (2010) navrhuje postup pro detekci, extrakci a modelování stromů např. pro účely vizualizace ve 3D modelech měst z dat MLS. Dosavadní metody extrakce stromů byly vyvinuty pro data LLS (absence kmenů), kde bylo k vizualizaci přistupováno voxelově, nebo pomocí jednoduchých tvarů. V případě použití TLS jsou kmeny a větve v hustých mračcích prokládány válcovými plochami. Pro klasifikaci kmene a listů se využívá intenzity. Rozdíl mezi lesnickými aplikacemi a modely měst je v tom, že stromy je třeba odlišit od jiných podobných objektů, zejména lamp. Zpracování probíhá ve třech krocích – detekce stromů, zjednodušení tvaru a modelování. Nejprve je mračno segmentováno na homogenní plochy, velké rovinné plochy jsou odstraněny. Zbývající body jsou spojeny do skupin na základě prostorové blízkosti. Pro tyto segmenty je vypočtena směrodatná výšková odchylka a poměr

mezi celkovou hustotou bodů v segmentu a hustotou cca 0,5m nad zemí. Díky tomu lze odlišit stromy od ostatních objektů (segmenty s vysokou směrodatnou odchylkou ve výšce a malým poměrem hustot). V dalším kroku je sníženo množství bodů stromu pomocí alfa algoritmu. Alfa tvary pomáhají zachovat vnější tvar neorganizovaných mračen bodů při obrovském snížení bodů – v tomto případě o 95%. Z redukovaného mračen jsou určeny základní parametry stromu – výška, šířka a tvar koruny, výška a šířka kmene. Modely byly vytvořeny prostřednictvím open-source nástroje *OpenAles* pro 3D modelování stromů. Nástroj umožňuje rozlišení korun stromů na čtyři základní tvary (koule, válec a kužel rozšiřující se dolů, nebo nahoru), provádí detailní modelování větví (sklon větví je fixně nastaven pro jednotlivé typy korun) a modely doplní o listy. Výsledkem je realistický model stromu, který dostatečně odpovídá skenovanému objektu (obr. 4).



Obr. 4 Detekce a modelování stromů – Rutzinger a kol. (2010), str.523

Lehtomäki a kol. (2010) se zabývají detekcí tyčových objektů z mračen bodů z MLS. Objekty typu lamp, dopravních značek a stromů jsou detekovány z jednotlivých skenovaných linií. Jsou vyhledávány izolované malé skupiny bodů s malými vzdálenostmi mezi sebou a tyto skupinky jsou shlukovány napříč skenovanými liniemi. *Yu a kol. (2014)* řeší stejný problém konstrukcí ROI (*region of interest*) a následným popisem bodů v rámci ROI prostřednictvím histogramu.

Jaakkola a kol. (2008) popisují postup k automatické extrakci vozovky, dělicích čar a přechodů z dat MLS založený na rozdílné intenzitě odraženého signálu. Výstupem je model ve formě TINu s vyznačeným vodorovným dopravním značením. *Kumar a kol. (2014)* navrhuje vylepšený postup, při kterém jsou hodnoty intenzity odraženého signálu opravovány o vliv vzdálenosti místa odrazu a úhlu dopadu.

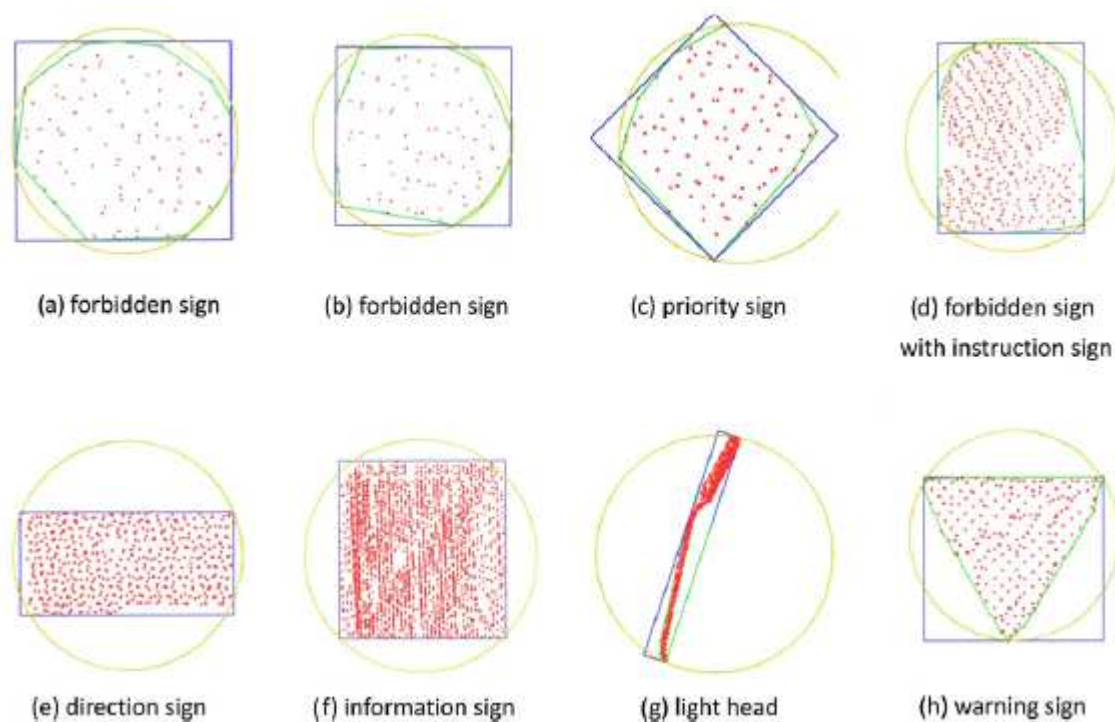
Pu a kol. (2011) shrnuje současný stav řešení problému klasifikace dat MLS – to, že neexistuje univerzální automatický nástroj pro klasifikaci všech prvků zájmu, pouze množství nástrojů řešících různě úspěšně klasifikaci jednotlivých prvků (vodících pruhů na vozovce, pouličního osvětlení...). Navrhuje postup ke zpracování dat MLS (viz úvod kapitoly 3.2). Používá k segmentaci dat algoritmus *surface growing* založený na detekci rovin prostřednictvím 3D Houghovy transformace (rovinné plochy jsou základní segment většiny člověkem vytvořených objektů). Klasifikace třídy *ground* se provádí na základě předpokladu, že terén se sestává z velkých ploch v konstantní výšce pod trajektorií skeneru, segmenty spojené s body terénu jsou zařazeny do třídy *on-ground* a zbývající body (chyby, oříznuté

části, které by bylo komplikované přesněji určit) jsou zařazeny pod *off-ground*.

Postup kombinuje metody řízené daty (tedy segmentace) a metody řízené modelem (rozpoznávání objektů na základě znalostí – znalostní metody). Jednotlivé hledané objekty jsou charakterizovány svými vlastnostmi, segmenty jsou poté určovány na základě porovnání jejich vlastností s očekávanými vlastnostmi různých typů objektů. Mezi porovnávané vlastnosti patří: velikost (výška, šířka, délka), poloha, orientace (zemský povrch – horizontální, lampy – vertikální), barva, materiál (odrazivost – vodorovné a svislé dopravní značky).

Svislé dopravní značky jsou rozpoznávány na základě vytvoření ohraničujícího konvexního polygonu kolem svislých segmentů, dále se pro daný segment určí nejmenší ohraničující kruh a čtverec a porovnává se obsah ohraničujícího polygonu s obsahy obou geometrických primitiv, pokud není shoda ani u jednoho prvku dostačující, provede se testování na shodu s trojúhelníkem.

Rozpoznávání tyčových objektů probíhá na základě porovnávání třetí čtvrtiny segmentu (počítáno odspodu). Pouze konkrétní část svisle orientovaného segmentu byla vybrána z důvodu, že části lamp a sloupů jsou často zarostlé vegetací, což ztěžuje jejich identifikaci. Vybraná část segmentu je dále rozdělena na vrstvy, bodům v jednotlivých vrstvách jsou přiřazeny ohraničující čtverce a parametry těchto čtverců jsou porovnávány s nastavenými limity.



Obr. 5 Detekce dopravních značek – *Pu a kol. (2011), str.532*

2.5 Klasifikace mračna bodů pomocí komerčního řešení

Dnes dostupné komerční softwary nabízí řešení pro proces klasifikace mračen bodů s různou mírou automatizace. V následujících odstavcích je popsán příklad možného řešení klasifikace mračna bodů za použití softwaru TerraScan finské společnosti Terrasolid, který měla autorka k dispozici pro

testování. Software byl původně vyvinut pro data leteckého laserového skenování, ale dnes již v nadstavbách nabízí množství nástrojů pro analýzu dat MLS pozemních komunikací (zejména v pozdějších fázích zpracování dat, jako je vyhodnocování podélného a příčného sklonu, tvorba podélných a příčných řezů apod.). Přesto, že se text věnuje popisu funkcí konkrétního softwaru, principy, na kterých tyto funkce pracují, mají obecnou platnost.

2.5.1 Obecný postup

Popisovaný postup předpokládá, že jsou již k dispozici georeferencovaná data ve formě mračen bodů. Princip georeferencování dat laserového skenování není tématem této práce, a proto zde není uveden. Více informací lze uvádět např. *Shan a Toth (2009)*.

V případě dat MLS se pracuje často s mračny v řádu stovek milionů bodů a není v možnostech běžně dostupného hardwaru tato data zpracovávat najednou. Z tohoto důvodu jsou data dělena do bloků. Jak již bylo uvedeno dříve mračna bodů z MLS je vhodné dělit podél trajektorie z důvodu minimalizace množství těžko klasifikovatelných částí. Testovaný software umožňuje práci s projektem, ve kterém je možné data rozdělit do bloků o určité délce (případně i šířce) podél dané trajektorie. Tyto bloky je pak možné zpracovávat postupně manuálně nebo automaticky pomocí předem připravených maker (předem sestavených sledů funkcí, které jsou postupně aplikovány na všechny bloky v projektu).

V první fázi klasifikace je třeba mračno očistit od hrubých chyb - od bodů "plovoucích" ve vzduchu, bodů zjevně pod terénem apod. Zejména body pod terénem značně komplikují klasifikaci terénních bodů, která vychází často právě z těchto chybných nízko položených bodů a neumožní správně klasifikovat terénní body v bezprostřední blízkosti bodů chybných.

Po očištění se přistupuje k samotné klasifikaci. Body se dělí do skupin, zpravidla podle příslušnosti k terénu, vegetaci, budovám a dalším. Správně provedená klasifikace může výrazně zjednodušit další zpracování dat, na druhou stranu, chyby v klasifikaci si data nesou sebou a komplikují další práci.

Při zpracování kompletního projektu se zpravidla provede testování nastavení funkcí na vzorku dat a poté se vytvoří makro, které provede automatickou část zpracování s otestovaným nastavením jednotlivých funkcí postupně na celém souboru dat. Pro správný efekt použitých automatických a poloautomatických funkcí je třeba vhodné nastavení parametrů těchto funkcí vzhledem k vlastnostem zpracovávaných dat, což vyžaduje zkušenosti a dlouhodobější testování.

Při zpracování dat MLS následují po klasifikaci zpravidla další kroky vedoucí k požadovaným finálním výstupům. K těmto krokům patří např. tvorba DMT, manuální kresba, případně nahrazování částí mračna geometrickými primitivami, tvorba půdorysů a řezů.

2.5.2 Čištění mračna

Užitečné poloautomatické funkce k očištění mračna lze nalézt v hlavní nabídce softwaru *Classify/Routine*:

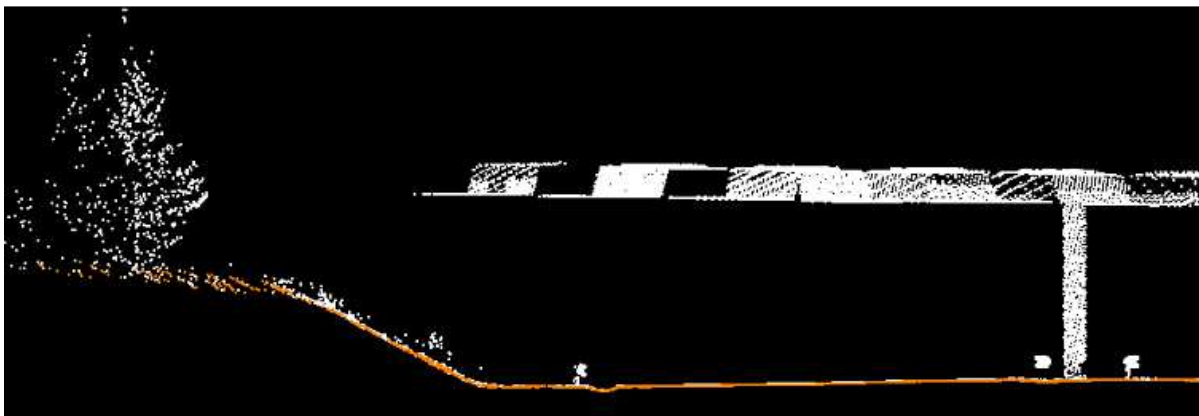
- *Isolated points* - funkce klasifikuje body, které v daném okolí nemají daný minimální počet sousedů. Funkci lze provést i několikrát s různým nastavením – pro nalezení odlehklých bodů i skupin bodů.

- *Air points* - funkce klasifikuje body, jejichž výška je v daném okruhu větší o víc, než je daný násobek směrodatné odchylky výšky v daném okruhu od průměrné výšky v tomto okruhu.
- *Bellow surface* - funkce klasifikuje body z vybrané třídy, jejichž výška je menší o víc, než je daný násobek směrodatné odchylky výšky od průměrné výšky 25 nejbližších sousedů – pokud tento rozdíl není větší než daný absolutní limit. Tato funkce pomůže předcházet chybám v klasifikaci terénu.

2.5.3 Klasifikace terénu

Klasifikaci terénu lze provádět pomocí následujících funkcí:

- *Classify/Routine/Ground* - automatická funkce, která na základě předem klasifikovaných bodů terénu a lokálně nízko položených bodů vytvoří jednoduchý primární DMT, který iteračně doplňuje body z mračna, čímž klasifikuje terénní body. Funkce může selhávat místy v příkopech podél komunikace a to zejména při defaultním nastavení redukce iteračního úhlu pro strany kratší, než je stanovený limit.
- *Classify/Add point to ground* - poloautomatická funkce určená pro doplnění klasifikace v místech, kde automatická funkce selhala, po manuální klasifikaci bodu v postižené oblasti zopakuje klasifikaci terénu v daném okruhu.



Obr. 6 Výsledek klasifikace terénních bodů (oranžové)

2.5.4 Funkce ke klasifikaci

Mimo výše zmíněných funkcí pro klasifikaci odlehlých bodů a klasifikaci terénních bodů nabízí software TerraScan množství dalších funkcí pro klasifikaci (záložka *Classify*), pro přehlednost jsou rozděleny do několika kategorií:

- Funkce pro klasifikaci na základě obecných vlastností

Mezi tyto funkce patří *By intensity* (vybere body o intenzitě v daném intervalu), *By color* (body dané barvy), *By absolute elevation* (body v intervalu daných nadmořských výšek) a *By class* (body vybrané třídy do jiné třídy).

- Funkce vyžadující inicializační prvek

Tyto funkce vyžadují zpravidla předem vytvořený prvek a body klasifikují na základě prostorových

vztahů k tomuto prvku, např. *By center line* (vyžaduje předem vykreslený a označený liniový prvek/prvky, klasifikuje na základě požadované min. a max. vzdálenosti od prvku - možné i výškově), *Inside shapes* (body uvnitř/vně vybraného uzavřeného prvku), mohla by sem patřit i výše zmíněná *Below surface*.

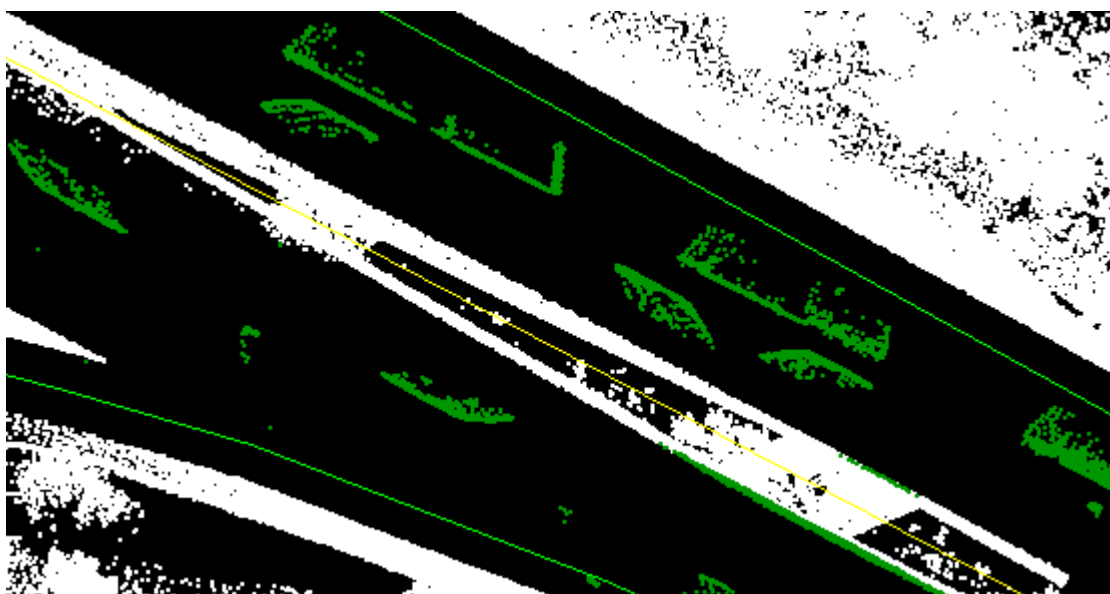
- Specifické funkce

Do této kategorie spadají funkce určené pro klasifikaci zcela specifických objektů, z nichž u některých je třeba tyto objekty blíže definovat. Jsou to: *By tunnel section* (předem se vykreslí typový řez tunelu, na základě tohoto řezu a vykreslené trasy se klasifikují body tunelu), *Railroad* (vybere body odpovídající svým výškovým uspořádáním kolejím - opět podél vykreslené trasy), *Buildings* (klasifikuje střechy budov - pouze pro data ALS), *Wire danger points* (klasifikuje body, které jsou do dané max. vzdálenosti od vykresleného elektrického vedení).

2.5.5 Použití kombinace funkcí

Nabízené funkce pro klasifikaci nejsou natolik univerzální, aby bylo možno jimi klasifikovat jednoduše všechny požadované třídy, vizuální kontrola a případná manuální oprava jsou nezbytné. Manuální klasifikaci bodů lze provádět např. pomocí funkcí *Classify/Inside fence* (klasifikuje body uvnitř ohrady) a *Assign points* (umožňuje klasifikaci jednotlivých bodů). Při manuálním výběru bodů pomáhá zobrazit si mračno v řezech s různým nastavením barev (dle intensity, třídy, apod.). Obecně platí, že třídu nelze klasifikovat za použití pouze jedné funkce. Následuje popis příkladu použití kombinace klasifikačních funkcí pro extrakci hledaných objektů.

Např. klasifikaci bodů získaných odrazem od jedoucích aut lze provést pomocí několika poloautomatických funkcí (obr. 7). Předpokládá se již provedená klasifikace terénních bodů. Po vykreslení osy komunikace se klasifikují body v šíři komunikace (funkcí *Classify/Routin/By centerline* s nastavením vhodné vzdálenosti od osy komunikace, s rozdílným nastavením pro připojovací pruhy a na ně navazující komunikace). Z této skupiny se poté klasifikují body projíždějících vozidel pomocí výšky nad terénem (*Classify/Routin/By height from ground*).



Obr. 7 Příklad využití poloautomatických funkcí ke klasifikaci projíždějících vozidel

2.6 Shrnutí současného stavu a motivace

Využití MLS pro dokumentaci pozemních komunikací a jejich nejbližšího okolí je aktuálně řešeným problémem a tématem mnoha odborných článků. Význam využití MLS v dopravě dokládá i v USA vydaná publikace (*Olsen, 2013*) shrnující popis principů MLS, mobilních skenovacích systémů, jejich parametrů a návodů, jak tuto technologii standardně využívat pro aplikace v dopravě. Většina vědeckých publikací na toto téma se bohužel zabývá pouze extrakcí jednoho konkrétního prvku zájmu (vodorovného, nebo svislého dopravního značení, lamp pouličního osvětlení a pod). I přes existenci obecného návrhu pro zpracování těchto dat (*Pu a kol., 2011*) dosud neexistuje nástroj pro automatickou klasifikaci mračen bodů z pozemních komunikací do několika základních tříd odpovídajících obecným potřebám.

2.6.1 Význam práce

Zásadní otázkou je, zda je takový komplexní nástroj pro automatickou klasifikaci dat z pozemních komunikací potřebný. Aby bylo možné si tuto otázku zodpovědět, je třeba zjistit, za jakým účelem se dokumentace pozemní komunikace a jejího okolí provádí.

Z doposud řešených projektů je zřejmé, že dokumentace pozemních komunikací mobilními skenovacími systémy má široké uplatnění:

- detekce změn (*Lindenbergh, Pietrzyk, (2015), Xiao a kol. (2016)*)
- detekce dopravního značení (pro analýzy bezpečnosti provozu na pozemních komunikacích - viditelnost dopravního značení, *Soilan a kol. (2016), Guan a kol., (2015)*)
- dokumentace stavu a poškození vozovky (vyjeté koleje, výtluky, průsaky vody v tunelech apod.) nebo jejího okolí (sesuvy půdy); např. *González-Jorge a kol. (2013), Lantieri a kol. (2015)*
- dokumentace stavu zařízení (např. kontrola správné funkce osvětlení v tunelech - *Puente a kol. (2014)*)
- tvorba podélných a příčných řezů (*Holgado-Barco a kol., 2014*)
- dokumentace bezprostředního okolí jako podklad pro další práce (např. pro odhad zemních prací při rozšiřování vozovky, *Wang a kol., 2013*)

Obecně se touto metodou provádí dokumentace součástí pozemní komunikace (vodorovného a svislého dopravního značení, zábradlí, odrazníků, svodidel, směrových sloupků, zpomalovacích prahů a dalších) a jejího příslušenství (veřejného osvětlení, světelných signalizačních zařízení sloužících k řízení provozu, silniční vegetace, zásobníků a skládek údržbových hmot, objektů a prostranství bezprostředně sloužících výkonu údržby pozemní komunikace, zařízení zabraňujících vniknutí volně žijících živočichů) pro potřeby správce komunikace.

Snaha je o širší využití získaného datového souboru - pro porovnávání stavu v různých časových epochách, analýzy bezpečnosti, viditelnosti, apod. Toto širší využití vyžaduje právě komplexní nástroj, který umožní rozdělit mračno do několika obecných tříd. Takový nástroj provede první hrubou klasifikaci mračna a připraví tak základ pro provádění dalších analýz dle konkrétní okamžité potřeby. Sofistikovanější nástroje pak již mohou pracovat pouze s body vybrané třídy, čímž se sníží množství testovaných dat a náročnost analýz.

3 Data

3.1 Datový soubor

3.1.1 Projekt CG912-105-520

Data použitá pro vývoj a ověřování metody klasifikace prezentované v této práci byla pořízena v rámci projektu CG912-105-520. V rámci téhož projektu byla také vytvořena diplomová práce zabývající se aplikacemi MLS a postupem zpracování dat MLS pomocí komerčního softwaru (*Faltýnová, 2010*), na kterou tato disertační práce volně navazuje.

Fakulta stavební ČVUT v Praze zastoupená prof. Karlem Pavelkou byla jedním z řešitelů projektu poskytovaného Ministerstvem dopravy pro roky 2009 a 2010. Celý název projektu byl: „Možnosti monitorování stavu a změn v okolí hlavních komunikací metodami dálkového průzkumu Země a laserového skenování a jejich využití pro realizaci udržitelného rozvoje dopravy“. Projekt byl součástí programu: CG - Podpora realizace udržitelného rozvoje dopravy (2007-2011). Cílem projektu bylo vyvinout a ověřit technologii zjišťování změn a vývoje prostředí zejména monitorováním stavu vegetace v okolí komunikací, navrhnout, vyvinout a otestovat možnosti prostorové dokumentace okolí komunikací pomocí mobilního 3D skeneru.

V rámci projektu byla metodou mobilního laserového skenování zmapována dálnice D11, z Prahy do Hradce Králové. Mapování a předzpracování dat provedly firmy GEOVAP spol. s r.o. (Pardubice) a Teccon bvba (Zwijndrecht, Belgie). K mapování byl použit mobilní mapovací systém LYNX Mobile Mapper.

3.1.2 Sběr dat a předzpracování

Základní údaje o měření

Datum: 25. 2. 2009

Čas: 8:00-16:15

Počasí: oblačno až polojasno, slabé sněhové přeháňky; teplota 0-4°C; vítr zpočátku Z, během dne až JV do 5m/s; tlak 1005 hPa; vlhkost 51-60%

Měřicí vozidlo: obsluha mapovacího systému - Matthijs Vanommeslaeghe/Bruno Van Bastelaere (Teccon bvba)

řidič - Bruno Van Bastelaere/Matthijs Vanommeslaeghe

navigátor - Josef Pazdera (GEOVAP spol. s r.o.)

Doprovodné vozidlo: Bohumil Kouřim (GEOVAP spol. s r.o.)

Na počátku a konci každého měřeného úseku se provádělo cca pětiminutové měření statickou GPS. Měření statickou GPS (cca 1 min) se opakovalo několikrát za měřený úsek. Rychlost jízdy při měření v odstavném pruhu byla 60 - 70 km/h a v rychlém pruhu cca 100 km/h.

Předzpracování dat

Pro výpočet trajektorie systému byla použita data z POS Applanix 420 LV (poziční a orientační systém), zpracování bylo provedeno v softwaru Applanix POSPac, do výpočtu byla zahrnuta i data z referenčních stanic GPS. Použita byla denní 1Hz data stanic CZEPOS Praha, Pecný, Mladá Boleslav, Trutnov a Pardubice a referenčních GPS postavených na vytipovaných místech u dálnice D11 (tab. 1).

Tab. 1 Referenční GPS

č.b.	ETRS 89		
	N	E	H _{el}
05150240	50° 8' 21.68"	15° 20' 39.69"	252.720m
14090400	50° 9' 52.95"	14° 49' 01.91"	218.250m

Na základě znalosti přesné trajektorie a kalibrace celého mobilního mapovacího systému byly spočteny v softwaru DASHMap souřadnice jednotlivých bodů mračna bodů. Výstupem bylo mračno bodů ve formátu LAS v souřadnicích ETRS 89.

Pro transformaci mračna bodů do S-JTSK byl využit software TRANSFORM+ a globální transformační klíč.

Formát dat

Výsledné mračno bylo rozděleno do cca 800 souborů (250x250m) s množstvím bodů do cca 15 mil. Jednotlivé části mračen jsou k dispozici v souborech formátu LAS a souřadnicovém systému UTM - zóna 33N, dále pak v textových formátech opět v souřadnicovém systému UTM 33 a v S-JTSK. Mimo informaci o poloze bodů obsahují soubory i intenzitu navráceného signálu v rozmezí 1-4096.

Informace o trajektorii systému je k dispozici v textovém souboru v systému UTM.

Data jsou doplněna fotografiemi pořízenými během skenování kamerami, které jsou součástí systému. Pro obarvení mračna se ale snímky příliš nehodí, protože zobrazují jen malou část mračna a při kombinování snímků by vadila jejich různá světlost způsobená různým natočením kamer a pozice slunce.

3.1.3 LYNX Mobile Mapper

LYNX Mobile Mapper je produktem kanadské společnosti Optech Inc. Senzorové pole mobilního systému je pevná konstrukce, která je použitelná se standardními střešními nosiči a lze ji přizpůsobit vlastní instalaci. Standardní aparatura obsahuje úchyty s nastavitelnou orientací pro dva lidarové senzory a dvě kamery, stejně jako držák na systém IMU a GPS antény. POSPAC Land od Applanix/Trimble používaný LYNX Mobile Mapperem představuje softwarové řešení pro zpracování úloh dynamické navigace. Softwarové řešení pro LYNX Mobile Mapper se sestává z LYNX-Survey a LYNX-Process.

Specifikace systému

Prostorové rozlišení bodů až 1 cm při 100 km/h (závisí na nastavení lidarů)

Maximální dosah	>100 m (při 20% návratnosti signálu)
Přesnost měření vzdálenosti	±7 mm
Absolutní přesnost	±5 cm (závisí na datech z GPS)
Pozice/Orientace systému	Applanix POS LV 420
Zorné pole skeneru	plných 360°
Rychlost skeneru	9000 otáček za minutu
Rychlost měření	100,000 pulsů/s na senzor
Měření na puls	až 4 (1., 2., 3. a poslední echo na senzor)
Počet senzorů	1 až 4 lidary
Kamery	až 2
Nosné vozidlo	systém plně adaptabilní
Operační teplota	-20°C to 40°C (vnější prostředí)
Skladovací teplota	-40°C to 80°C
Klasifikace laseru	IEC/CDRH 1. třída

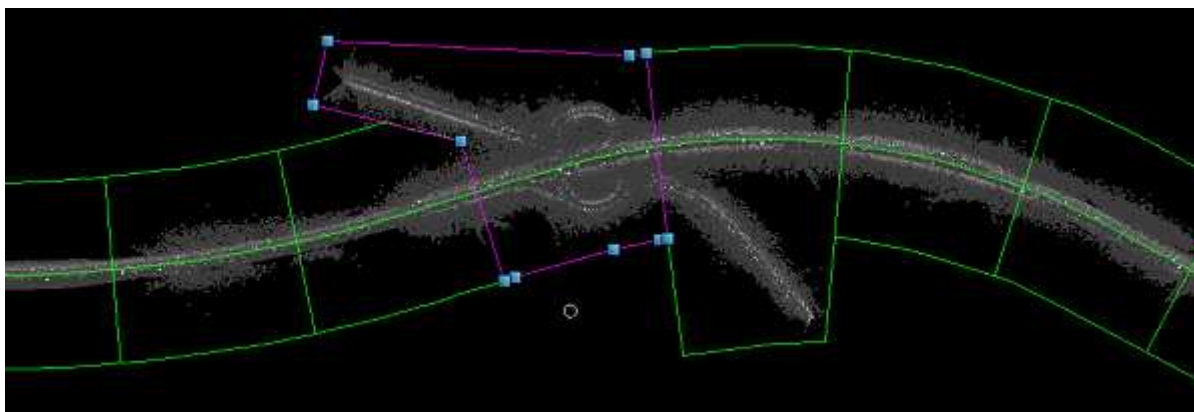


Obr. 8 Lynx Mobile Mapper (skenování dálnice D11, 2009, foto Ing. Pazdera)

3.2 Výběr a příprava dat pro testování

3.2.1 Rozdělení do bloků

Metoda laserového skenování umožňuje pořizování enormních datových souborů. Jak bylo zmíněno dříve, bývá nezbytné data v rámci jednoho projektu rozdělit do menších bloků. Data ze skenování dálnice D11 byla dodána v souborech rozdělených do čtvercových bloků rovnoběžných s osami x a y . Tento způsob dělení není pro data MLS mající liniový charakter vhodný, vzniklé bloky obsahují často malé nesouvislé části mračka a počet bodů v jednotlivých blocích je značně nevyvážený. Pro potřeby projektu byla vybraná data přerozdělena do bloků vytvořených podél trajektorie. Cílem předkládané práce bylo nalézt metodu klasifikace výpočetně nepřítliš náročnou a hardware použitý pro testování odpovídal svými parametry současným kancelářským počítačům (procesor Intel(R) Core™2 Quad CPU 2,83GHz, RAM 8GB, 64bitový operační systém Windows 7). Na základě prvních testů byly zvoleny bloky o délce a šířce 200m, takto vzniklá mračka obsahovala cca 3-4 miliony bodů. Limitujícím faktorem při volbě velikosti bloku byla maximální povolená velikost matice v softwaru MatLab. Velikost matice byla závislá na rozměrech bloku ve směru os x , y , z a stanoveném rozlišení rastrových dat. Software MatLab vychází při stanovování maximální povolené velikosti matice z velikosti dostupné operační paměti.



Obr. 9 Automatická generace hranic bloků podél trajektorie a jejich manuální úprava

3.2.2 Výběr bloků pro experiment

Bloky pro testování byly vybrány tak, aby v co nejširší míře postihovaly rozmanitost prostředí a zobrazovaly většinu objektů, které se na pozemní komunikaci a jejím bezprostředním okolí mohou vyskytovat. Dále byl vybrán souvislý pás dálnice v délce 1,8km rozdělený do devíti bloků a obsahující všechny zájmové prvky. Data z tohoto pásu nebyla využita v průběhu vývoje metody, ale pro ověření její funkčnosti.

Z výběru bloků pro testování byl vynechán první úsek dálnice od Prahy (cca 15km), z důvodu špatně georeferencovaných dat pořizovaných v opačných směrech jízdy. Na datech z vozovky je zřejmý výškový rozdíl mezi body pořizovanými z jednoho a druhého směru jízdy (obr. 10Obr. 10). Soubory testovacích dat byly pojmenovány podle kilometráže dálnice, na kterém se nachází. Popis jednotlivých bloků následuje.



Obr. 10 Chyba v georeferencování mračen - příčný řez mračnem, 8. km (rozdíl ve výšce 0,6m)

Data pro testování v průběhu vývoje

- 18. km – svislá dopravní značka, elektrické vedení, svodidla, směrové sloupky, vegetace
- 22. km – svislá dopravní značka, billboard, svodidla, vegetace
- 24. km – billboardy, svodidla, hustá vegetace
- 26. km – billboardy, svodidla, SOS hlásky, vegetace, svodidla
- 31. km – svodidla, směrové sloupky, mýtná brána, vegetace
- 36. km – svislá dopravní značka, svodidla, směrové sloupky, mýtná brána, vegetace
- 58. km – svodidla, směrové sloupky, mýtná brána, oplocení
- 62. km – svislé dopravní značky, svodidla, směrové sloupky, mýtná brána, oplocení
- 62. km (B) – svodidla, most
- 72. km – přechod pro zvěř, svodidla, SOS hlásky, oplocení, vegetace

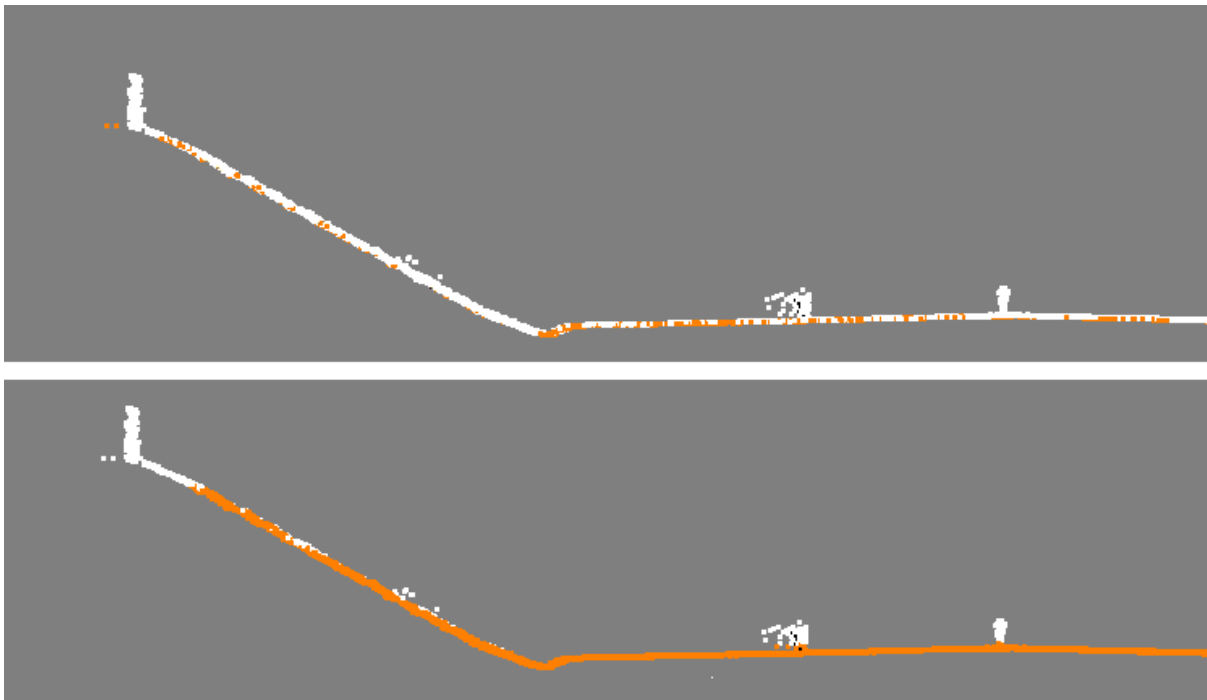
Data pro ověření metody

- 73. - 75. km – úsek 1,8km, rozdělen podél trajektorie na 9 bloků o délce 200m, obsahuje svislé dopravní značky, billboardy, SOS hlásky, směrové sloupky, svodidla, oplocení, elektrické vedení, mýtnou bránu, most, tunel, oplocení i vegetaci

3.2.3 Klasifikace terénních bodů a přepočítání výšky

Rešerše ukázala, že existuje množství nástrojů řešících klasifikaci terénních bodů s vysokou úspěšností. Představovaná metoda klasifikace mračna bodů se tedy klasifikací terénních bodů nezabývá, vychází z mračna, u kterého byla již třída terén klasifikována, a využívá výhod, které provedená klasifikace přináší.

Ke klasifikaci terénních bodů pro potřeby experimentu byly testovány dva softwary – Lastools a TerraScan. U obou byla vyzkoušena základní automatická funkce pro klasifikaci terénu, ponecháno bylo defaultní nastavení. Lastools zaměřený především na data LLS klasifikoval vyšší počet bodů – v případě dat D11 i spodní části projíždějících vozidel a neklasifikoval horní okraje svahů (případ, kdy byla komunikace v zářezu). Oproti tomu TerraScan klasifikoval výrazně méně bodů, ale minimalizoval množství bodů chybně klasifikovaných jako terénní a vypořádal se dobře s klasifikací bodů na okrajích mračna. Pro klasifikaci testovacích dat byl zvolen jako vhodnější nástroj TerraScan.



Obr. 11 Klasifikace terénu – Terrascan (nahore), Lastools (dole); příčný řez komunikací a přilehlým svahem (na spodním obrázku neklasifikován konec svahu – vlevo, klasifikována spodní část auta a svodidel - vpravo)

Klasifikace terénních bodů v softwaru TerraScan probíhala následovně:

Nejprve bylo třeba očistit mračno od hrubých chyb. Pro zvýšení úspěšnosti klasifikace terénních bodů bylo třeba zbavit se bodů pod terénem. K nalezení těchto nízko položených bodů byla použita funkce *Classify/Routine/Low points* – s nastavením: max. počet bodů ve skupině – 10, min. výškový rozdíl 0,20m oproti bodům v okruhu 5,00m.

Poté byla provedena automatická klasifikace funkcí *Classify/Routine/Ground* s defaultním nastavením.

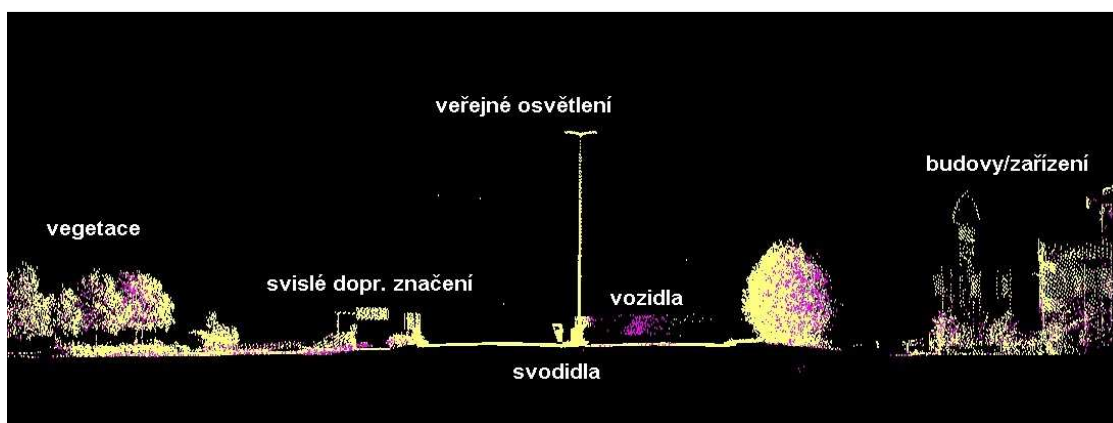
Vzhledem k tomu, že parametry objektů podél pozemní komunikace jsou často spojeny s určitou výškou nad terénem (např. minimální výška spodní hrany svislé dopravní značky nad jízdním pruhem), bylo třeba tuto výšku získat. Po klasifikaci terénních bodů byla v programu TerraScan přepočtena výška bodů na výšku bodů nad terénem (stejný postup jako při tvorbě normalizovaného digitálního povrchu – DMP). Použita byla automatická funkce *Tools/Transform loaded points* s nastavením *Transform: Height from ground*.

4 Experiment

4.1 Koncept

Koncept navrhované metody klasifikace mračen bodů vychází z následujících předpokladů:

- úlohu klasifikace mračna lze převést na úlohu klasifikace rastru, což vede k redukci zpracovávaných dat a snížení výpočetní náročnosti algoritmu,
- vlastnosti hledaných objektů jsou známy a lze je využít k jejich klasifikaci, stejně jako u metod řízených modelem (rozpoznávání objektů na základě znalostí),
- klasifikace terénních bodů je již dostatečně úspěšně řešená – není tedy předmětem představované práce a lze využít jejich výsledků
- cílem práce je základní klasifikace mračna bodů, která je nutným předpokladem pro aplikaci dalších již úzce zaměřených klasifikačních postupů a algoritmů



Obr. 12 Řez mračnem bodů

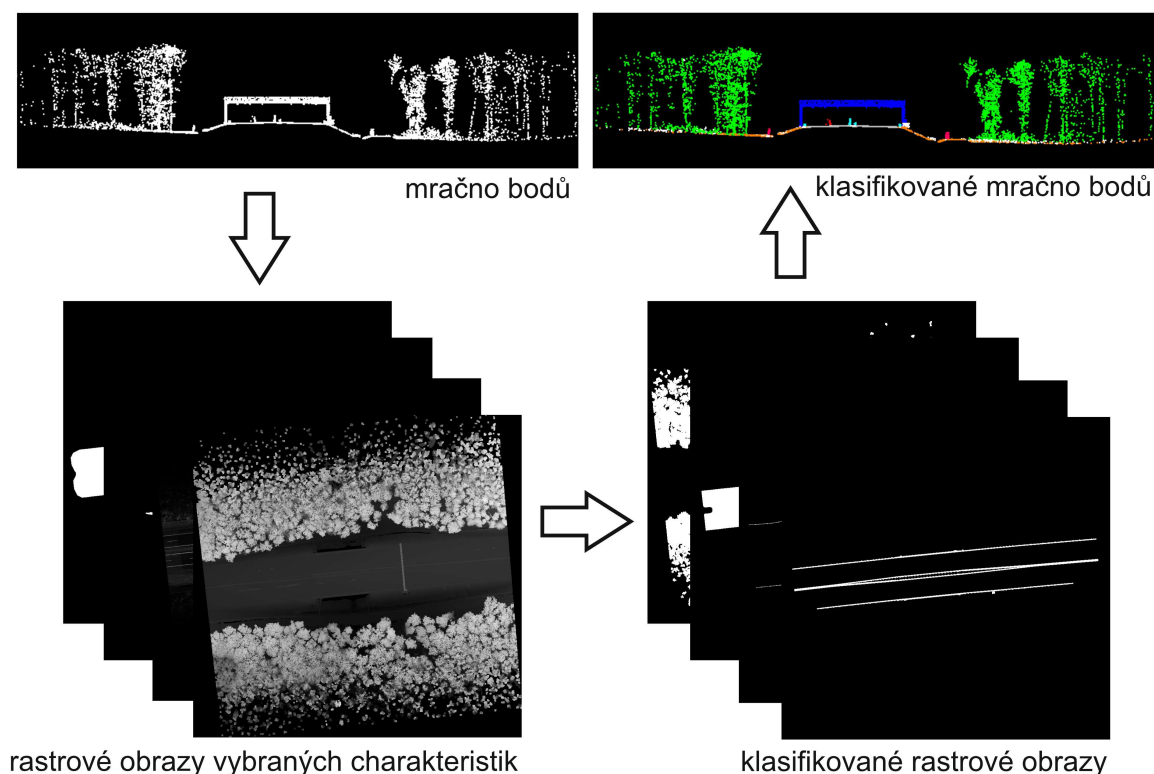
Navrhovaný postup zahrnuje převod mračna bodů do rastrového obrazu. Rastrový obraz vzniká jako průmět dat do vodorovné roviny, nebo jako řez daty vodorovnou rovinou, přičemž hodnota jednotlivého pixelu je odvozena z bodů spadajících do daného pixelu, nebo bodů v určitém výškovém rozmezí spadajících do daného pixelu. Hodnoty pixelů reprezentují pixelu příslušnou část mračna bodů (např. počet bodů, maximální výšku, intenzitu). Takto se vytvoří několik rastrových vrstev, které popisují data pomocí vybraných charakteristik.

Při výběru sledovaných charakteristik/parametrů se vychází ze znalosti, tj. hledají se takové parametry, ve kterých se jednotlivé typy objektů v mračnu od sebe odlišují, např.:

- vozovka – se vyznačuje malým výškovým rozptylem bodů a malým sklonem, dále velkou hustotou bodů (při pořizování dat se nachází blízko skeneru)
- vegetace – se vyznačuje větším rozptylem bodů ve výšce a tvoří kompaktnější objekty (blíží se kruhu) než většina umělých objektů
- svodidla – jsou liniové prvky, složené z bodů určité výšky nad terénem a v blízkosti vozovky

Na vytvořené rastry se aplikují algoritmy používané pro segmentaci a klasifikaci obrazových dat.

Výsledky klasifikace rastrových obrazů jsou poté přeneseny na mračna bodů. Schéma postupu je znázorněné na obr. 13.



Obr. 13 Schéma navrhovaného postupu

Při realizaci navrženého postupu je třeba zvážit možná úskalí, se kterými je třeba se vypořádat, zejména:

- přítomnost bodů více tříd v rámci jednoho pixelu – využití rastrového obrazu přináší tu nevýhodu, že jednomu pixelu přísluší jedna hodnota, zatímco body v různých výškách odpovídající danému pixelu mohou náležet do více tříd,
- volba velikosti pixelu – velikost pixelu je třeba volit s ohledem na hustotu skenování a zároveň na velikost a vzájemnou vzdálenost klasifikovaných objektů.

Přítomnost bodů více tříd v rámci jednoho pixelu algoritmus řeší tak, že klasifikace jednotlivých tříd probíhá samostatně a výsledkem je binární rastrový obraz pro každou třídu. Jeden pixel může tedy v různých rastrových obrazech odpovídat několika třídám. Při převodu klasifikace rastrových obrazů na mračno se příslušnost k třídě řeší nejen na základě klasifikovaného rastrového obrazu ale i výšky bodu nad terénem (např. u vozovky a mostů).

Velikost pixelu je zvolena na základě výsledků testování různých hodnot.

4.1.1 Třídy

Nezbytným krokem při tvorbě nástroje ke klasifikaci je stanovení tříd. Při stanovení tříd se vychází

z toho, jaké prvky se na pozemní komunikaci a jejím nejbližším okolí vyskytují. Názvy jednotlivých tříd jsou zvoleny v anglickém jazyce, aby bylo usnadněno případné využití nástroje odbornou veřejností.

Seznam tříd:

- 01 *Unclass*
- 02 *Ground*
- 03 *Roadway*
- 04 *Vehicles*
- 05 *Gates*
- 06 *Crash Barriers*
- 07 *Poles*
- 08 *Signs*
- 09 *Walls*
- 10 *Vegetation*

Třída *Unclass* je ponechána pro body neklasifikované, tj. body které není možné zařadit do žádné jiné třídy. Sem patří hrubé chyby, těžko klasifikovatelné části a odlehlé body nad terénem.

Třída *Ground* obsahuje terénní body s výjimkou bodů zobrazujících povrch komunikace.

Třída *Roadway*, tedy vozovka, obsahuje body povrchu vozovky a zpevněné krajnice. V podstatě jde o terénní body, ale protože v rámci výzkumu využití MLS je vozovce věnována zvláštní pozornost, jsou tyto body vyčleněny do zvláštní třídy. Tato třída tvoří základ pro analýzu stavu vozovky nebo extrakci vodorovného dopravního značení.

Třída *Vehicles* obsahuje body nad vozovkou představující dočasné objekty, tedy projíždějící vozidla.

Třída *Gates* zahrnuje jakékoli objekty překlenující pozemní komunikaci – elektrické vedení, mýtné brány, mosty a ekodukty (přechody pro zvěř).

Třída *Crash Barriers* obsahuje svodidla.

Třída *Poles* obsahuje samostatně stojící tyčové objekty v bezprostředním okolí komunikace, zejména směrové sloupky, SOS hlásky a sloupy elektrického vedení. Do této třídy nepatří tyče nesoucí svislé dopravní značky a stromy.

Třída *Signs* zahrnuje svislé dopravní značky, informační a reklamní tabule (nejčastěji billboardy).

Třída *Walls* obsahuje stěny (např. protihluková opatření) a ploty.

Třída *Vegetation* obsahuje vegetaci o daných minimálních parametrech.

Výše uvedený výčet tříd záměrně neobsahuje samostatnou třídu pro budovy. Mobilní skenovací systémy oproti leteckému provádí skenování tak nízko, že zpravidla snímají jen stěny budov bez střech. V bezprostřední blízkosti části naskenované dálnice D11 (mimo intravilán), která byla pro testování k dispozici, nebyly nalezeny vhodné objekty k testování. U částečně nasnímaných budov se

předpokládá zařazení do třídy *Walls*.

4.1.2 Nástroje

Příprava datových souborů, tj. dělení mračen do bloků, klasifikace terénu, výpočet výšky bodů nad terénem a manuální klasifikace souborů pro ověření kvality automatické klasifikace, se provádí v softwaru TerraScan.

Samotná realizace algoritmu automatické klasifikace probíhá v prostředí MatLab. Tento nástroj obsahuje širokou knihovnu funkcí pro práci nejen s obrazovými daty, je akademické obci dostupný a dobře známý. Při vývoji algoritmu se pracuje s verzí MatLab R2017a bez potřeby instalace zvláštních knihoven (z důvodu snazší dostupnosti a aplikovatelnosti výsledku). Pro snadnější využití dalšími subjekty je kód opatřen poznámkami v anglickém jazyce.

Algoritmus je realizován prostřednictvím čtyř kódů v programu MATLAB.

- **“twoZcoord.m”** – zajišťuje spojení dvou seznamů souřadnic do jednoho obsahujícího nadmořskou výšku i výšku nad terénem,
- **“pcloud2image.m”** – provádí import seznamu souřadnic, převod mračna bodů do rastrových obrazů a klasifikaci rastrových obrazů,
- **“image2pcloud.m”** – převádí výsledky klasifikace z rastrových obrazů na mračno,
- **“ClassQualCheck.m”** – porovnává mračna klasifikovaná manuálně a automaticky, ověřuje úspěšnost klasifikace.

4.2 Tvorba rastru

4.2.1 Vstupní data

Mračno bodů vstupuje do algoritmu ve formě neorganizovaného seznamu souřadnic uloženého v textovém souboru; jako oddělovač slouží mezera. Prvotní klasifikace terénních bodů za účelem zjištění průběhu terénu již byla provedena, proto obsahuje seznam souřadnic informaci o třídě. Body neklasifikované se označují kódem “1” a body klasifikované jako terén kódem “2” (standardní značení používané napříč softwary zabývajícími se klasifikací). Vstupní data obsahují dále rovinné souřadnice x a y , výšku z a výšku bodů nad terénem $Z_{\text{over ground}}$. Vzhledem k tomu, že software TerraScan neumožňuje export obou výšek najednou, byla v prostředí MATLAB vytvořena jednoduchá funkce, která spojuje informace ze dvou datových souborů:

$$[data_ZZ] = twoZcoord (data_1, data_2)$$

Formáty vstupních souborů ($data_1$, $data_2$) a výstupního souboru ($data_ZZ$) jsou uvedeny v tab. 2, kód funkce je uveden v příloze 4.

Tab. 2 Formát seznamů souřadnic

Soubor	Formát	Příklad
data_1	x y z	-655760.365 -1048527.401 266.990
data_2	class x y $Z_{\text{over ground}}$ intensity	1 -655760.365 -1048527.401 1.021 426
data_ZZ	class x y $Z_{\text{over ground}}$ intensity z	1 -655760.365 -1048527.401 1.021 426 266.990

4.2.2 Postup

Představovaná metoda klasifikace dat pozemní komunikace je založena na klasifikaci rastrových obrazů popisujících vstupní data pomocí různých charakteristik – hustoty bodů, maximální výšce bodů nad terénem apod. Aby bylo možno tyto rastrové obrazy jednoduše vygenerovat, mračno bodů je rozděleno do voxelů o velikosti odpovídající rozlišení budoucího rastrového obrazu (voxel označuje částici objemu v pravidelné mřížce třídimenzionálního prostoru; jde o analogii k pixelu, který reprezentuje oblast ve 2D mřížce). Pro každý voxel se uloží informace charakterizující body ve voxelu obsažené, na jejichž základě se určí hodnota pixelu rastrového obrazu.

Informace o jednotlivých voxelích jsou uloženy ve čtyřrozměrné matici (dále nazývána voxelová matice) – první tři rozměry odpovídají prostorovým souřadnicím x , y , $Z_{over\ ground}$ a označují umístění voxelu, čtvrtý rozměr obsahuje informace o bodech spadajících do daného voxelu. Jednotlivé hodnoty ve čtvrtém rozměru udávají:

- 1) počet bodů v dané buňce
- 2) minimální výšku bodu nad terénem (relativní)
- 3) maximální intenzitu bodu
- 4) minimální výšku bodu (absolutní)

Základním parametrem určujícím velikost voxelové matice jsou rozměry voxelů (rozlišení polohové a výškové). Čím větší rozměr voxelů, tj. nižší rozlišení, tím menší matice a obráceně. Po úvodních testech bylo zvoleno rozlišení polohové 0,20m a výškové 0,10m. Při vyšším polohovém rozlišení (testováno 0,10m) obsahuje matice velké množství prázdných voxelů a bylo by nutné hodnoty v prázdných voxelích ve velké míře interpolovat, při nižším rozlišení (testováno 0,30m a 0,50m) už se stává, že jeden voxel obsahuje body více blízkých objektů nebo dochází k tomu, že menší objekty jsou zastoupeny malým množstvím voxelů (resp. v obraze pixelů) a těžko se klasifikují. Výškové rozlišení bylo nastaveno na nižší hodnotu 0,10m z toho důvodu, že jsou v rámci klasifikace vyhledávány i poměrně nízké objekty a objekty s malým výškovým rozdílem a je snaha vyhnout se spojení bodů těchto objektů do jednoho voxelu.

Po importu mračna bodů jsou nalezeny krajní hodnoty – nejnižší a nejvyšší hodnoty souřadnic x , y a $Z_{over\ ground}$. Z rozdílu nejnižších a nejvyšších hodnot vydělených stanoveným rozlišením se po jejich zaokrouhlení na celá čísla získají rozměry voxelové matice. Vytvoří se prázdná čtyřrozměrná matice o daných rozměrech. Před plněním této voxelové matice je ještě třeba převést geodetické souřadnice bodů na souřadnice voxelové – udávající do jakého voxelu bod spadá. Proměnné v následujících rovnicích jsou označeny stejně jako v kódu.

Transformace souřadnic probíhá následovně:

$$dataIm = Mt2Im * (Mt2Or * data') \quad (1)$$

kde:

- vstupní geometrické souřadnice

$$data = \begin{bmatrix} Class & x & y & z_{over\ ground} & Intensity & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \end{bmatrix} \quad (2)$$

- výsledné voxelové souřadnice

$$dataIm = \begin{bmatrix} Class & x_{Im} & y_{Im} & z_{Im} & Intensity & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \end{bmatrix} \quad (3)$$

- matice redukující souřadnice geometrické o souřadnice počátku voxelové soustavy souřadnic

$$Mt2Or = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -x_0 \\ 0 & 0 & 1 & 0 & 0 & -y_0 \\ 0 & 0 & 0 & 1 & 0 & -z_0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

- matice rotace

$$Mt2Im = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{-1}{par.res} & 0 & 0 & 0 \\ 0 & \frac{-1}{par.res} & 0 & 0 & 0 & 1,5 \\ 0 & 0 & 0 & \frac{1}{par.Hres} & 0 & 0,5 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

Při převodu souřadnic geodetických na voxelové se osy x a y prohodí a změní orientaci, tak aby x_{Im} odpovídalo řádku matice a y_{Im} sloupci. Hodnoty v posledním sloupci matice rotace $Mt2Im$ mají význam kvůli budoucímu zaokrouhlení souřadnic (0,5 - voxelové souřadnice označující příslušnost bodu k voxelu začínají 1 a ne 0). Hodnoty v druhém a třetím řádku (1,5) jsou o jedna vyšší, tak aby byl vytvořen nárazník kolem dat, tj. aby žádný voxel/pixel obsahující body nezůstal na okraji rastrového obrazu (z důvodu pozdějšího filtrování obrazu).

Po provedení transformace souřadnic bodů na voxelové souřadnice následuje plnění voxelové matice. Algoritmus projde postupně celý seznam bodů, parametry každého bodu se porovnají s odpovídajícími prvky matice a případně se na základě těchto parametrů hodnoty prvků matice upraví. Např. v případě prvku matice reprezentujícího počet bodů obsažených v příslušném voxelu se k původní hodnotě prvku přičte jedna. V případě prvku matice charakterizujícího maximální intenzitu se hodnota intenzity testovaného bodu porovná s dosavadní hodnotou prvku a v případě, že je nová hodnota vyšší, nahradí tu původní.

Výsledná voxelová matice je základem pro tvorbu rastrových obrazů využitých pro klasifikaci. Seznam souřadnic bodů v systému voxelové matice je uložen ve formě textového souboru a později využit při přenosu výsledků klasifikace rastrových obrazů na mračno bodů.

4.3 Klasifikace rastrového obrazu

Vzhledem k odlišnému charakteru dat MLS oproti datům dálkového průzkumu Země (DPZ) není možné využít metodu příznakové klasifikace, při které by byl jeden pixel přiřazen k jedné třídě. Objekty jednotlivých klasifikovaných tříd se nachází v různých výškách nad terénem a mohou se ve svislém průmětu překrývat, z toho důvodu je třeba vytvářet více rastrových obrazů zobrazujících každou třídu jednotlivě (tedy spíše v řezu než průmětu) a takto jednotlivě třídy klasifikovat.

Klasifikace jednotlivých tříd začíná tvorbou primárního rastrového obrazu, který je vytvářen z určité vrstvy nebo více vrstev voxelové matice (podle výšky objektů dané třídy nad terénem), případně je využita kombinace těchto rastrů. Hodnota pixelu odpovídá vybrané charakteristice (např. množství bodů, maximální intenzitě apod.) bodů ve vybrané vrstvě.

Na vytvořený primární obraz jsou aplikovány následující metody pro zpracování obrazových dat vedoucí ke klasifikaci pixelů dané třídy:

- segmentace obrazu prahováním (rozdělení obrazu do oblastí se společnými vlastnostmi)
- filtrování obrazu mající za cíl zvýraznění určité informace (filtry s nízkou propustností k potlačení detailů, např. linií; filtry s vysokou propustností ke zdůraznění detailů)
- objektově orientovaná klasifikace (filtrace objektů na základě jejich vlastností – rozměrů, orientace)

Podrobně o digitálním zpracování obrazových dat pojednávají např. *Halounová (2009)* a *Liu a Mason (2016)*.

Popis klasifikace jednotlivých tříd je uveden v pořadí tak, jak je jejich klasifikace postupně řešena v algoritmu. Algoritmus klasifikace dat byl vyvíjen na základě testování na vybraných datových souborech uvedených v oddílu 3.2. Při stanovování parametrů pro klasifikaci různých prvků se, pokud to bylo možné, vycházelo z platných předpisů z oblasti projektování a provozu na pozemních komunikacích, pokud to možné nebylo (např. proto že takové parametry nejsou předpisy upraveny), byly hodnoty parametrů určeny na základě pozorování. Zdrojový kód klasifikace je uveden v příloze 5.

4.3.1 Roadway - vozovka

Každý algoritmus pro klasifikaci pracuje s určitou mírou nejistoty a jeho úspěšnost lze hodnotit na základě množství chyb typu I (body do dané třídy patří, ale klasifikovány nebyly) a chyb typu II (body do třídy nepatří, ale jsou do ní chybně klasifikovány). Vzhledem k tomu, že smyslem klasifikace této třídy, je připravit podklad pro analýzu stavu vozovky, je kladen důraz na minimalizaci chyb typu I i za rizika zvýšení množství chyb typu II, tj. je nezbytné klasifikovat kompletně povrch vozovky, případná klasifikace části nezpevněné krajnice, nebo středního dělicího pásu není zásadním problémem.

Kvalita klasifikace vozovky má vliv na kvalitu klasifikace dalších tříd, protože klasifikace některých

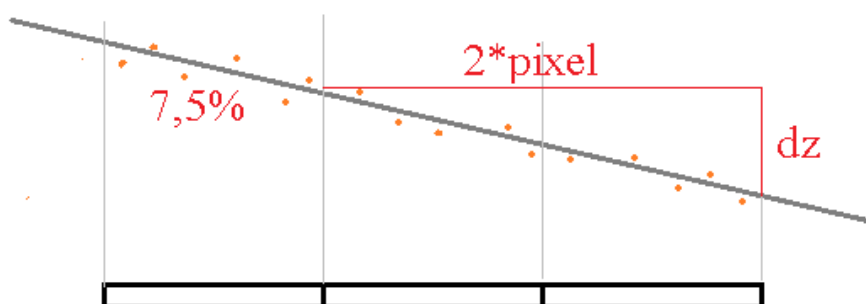
tříd vychází z prostorového vztahu objektů dané třídy vůči vozovce.

Klasifikace vozovky vychází z předpokladu, že voxely/pixely zobrazující vozovku by měly obsahovat pouze terénní body, případně šum, tedy body s nízkou výškou nad terénem. Dále předpokládá, že jde o povrch s pevně stanoveným maximálním sklonem. Klasifikace pracuje s oběma charakteristikami – max. výškou bodů nad terénem i maximálním sklonem, žádná z nich není samostatně dostačující. Např. mostní opěry mohou být klasifikovány jako terén a obsahovat pouze body s nízkou výškou nad terénem, ale překročí limit pro sklon vozovky; oproti tomu travnaté plochy v bezprostředním okolí komunikace, byť poměrně rovinné, obsahují body výš nad terénem. Dále lze z klasifikace vynechat všechny pixely neobsahující žádné body, nebo malé množství bodů. Vzhledem k tomu, že sběr dat probíhá z vozidla projíždějícího po pozemní komunikaci a hustota bodů v mračnu se snižuje se vzdáleností od zdroje, platí, že pixely zobrazující vozovku se nacházejí blízko skenujícího vozidla a mají vyšší hustotu bodů než body dále od vozovky. V nejpomalejším jízdním pruhu, kde se pohybovalo skenující vozidlo při skenování D11 dosahuje hustota až 30 bodů na pixel (0,20x0,20m), zatímco v nejrychlejším pruhu nejčastěji 3-5 bodů na pixel.

V první fázi jsou vytvořeny tři binární rastrové obrazy (obsahující pouze černou a bílou barvu, tj. hodnoty "0" a "1") zobrazující pixely potenciálně nepříslušející třídě *roadway*. Následuje popis vzniku jednotlivých obrazů.

Rastrový obraz maximální výšky bodů nad terénem v buňce je převeden na binární prahováním, bílou barvou jsou označeny pixely obsahující body s výškou nad terénem vyšší, než je nastavená prahová hodnota odpovídající šumu dat (nastaveno 0,05m). Do testování vstupují pouze body s hodnotou výšky do 0,75m nad terénem z toho důvodu, aby výsledek nebyl narušen objekty překlenujícími vozovku (tj. mýtnými bránami, přechody pro zvěř apod.) a případnou vegetací naklánějící se přes svodidla (hodnota vychází z dané minimální výšky svodidla na pozemních komunikacích – 0,75m (TP 114, 2015).

Rastrový obraz maximálního sklonu vzniká tak, že se pro každý pixel nalezne výška nejnižšího bodu do něj spadajícího, poté se pro každý pixel určí maximální rozdíl těchto výšek v okolí 3x3 pixely. Výsledný binární obraz zobrazuje pixely, u nichž vypočtená hodnota rozdílu výšek přesahuje prahovou hodnotu. Prahová hodnota vychází z maximálního možného výškového rozdílu při neměnném sklonu ve směru úhlopříčky pixelu (obr. 14). K výpočtu prahové hodnoty (na obrázku *dz*) je použit největší povolený výsledný sklon pro dálnice 7,5% (ČSN 73 6101, 2004).



Obr. 14 Schéma výpočtu rozdílu výšek pro maximální daný sklon

Rastrový obraz minimální hustoty bodů zobrazuje všechny pixely, do nichž spadá menší množství bodů než je stanovená prahová hodnota (2 body/pixel). Pokud je množství bodů na pixel menší než 2 nelze hovořit o žádném porovnávání hodnot v rámci pixelu (v předchozích obrazech).

Další postup vychází z kombinace tří předchozích obrazů. Každý pixel, který alespoň v jednom z těchto obrazů překračuje prahovou hodnotu je v novém obraze označen bílou barvou – tedy jako bod mimo třídu *roadway* (obr. 15). Je zřejmé, že rozložení barev na obrázku (bílá – mimo *roadway*, černá - *roadway*) odpovídá ve větší míře skutečnosti, ale zároveň obsahuje množství chyb. V černé ploše jsou rozeznatelné plochy obsahující body projíždějících vozidel a střední dělicí pás, na druhou stranu v bílé ploše jsou zřetelné menší plošky splňující nastavená kritéria pro vozovku, ale k vozovce zjevně nepatřící. Další postup zpracování obrazu má za cíl tyto chyby odstranit.



Obr. 15 Výsledek klasifikace na základě sklonu terénu, výšky bodů nad terénem a hustoty bodů; 31. km

Algoritmus pro úpravu obrazu využívá kombinace následujících funkcí určených pro filtrování binárních obrazových dat. Funkce, z nichž některé jsou v algoritmu aplikovány opakovaně, jsou uvedeny v pořadí, v jakém jsou prvně využity.

Funkce *bwmorph* provádí morfologické operace na binárním obraze:

- Operace *clean* odstraní izolované pixely – izolovaný pixel hodnoty "1" obklopený pixely hodnoty "0".
- Operace *bridge* vytváří „most“ mezi nepropojenými pixely – pokud má nulový pixel dva nenulové nespojené sousedy, změní se hodnota pixelu na "1" – tato operace pomáhá oddělit úzké výběžky pixelů mimo souvislé plochy.

např. $\begin{matrix} 1 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \end{matrix}$ upraví na $\begin{matrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{matrix}$

- Operace *diag* – propojí pixely umístěné na diagonále – má obdobný cíl, jako *bridge*.

např. $\begin{matrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{matrix}$ upraví na $\begin{matrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{matrix}$

- Operace *hbreak* přeruší spojení bloků ve tvaru písmene H – operace pomůže oddělit výběžky pixelů mimo *roadway* zasahující do souvislého povrchu vozovky.

např. $\begin{matrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \end{matrix}$ upraví na $\begin{matrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{matrix}$

- Operace *majority* nastaví novou hodnotu jednotlivých pixelů podle hodnoty, která v nejbližším okolí daného pixelu (3x3) převažuje.

- Operace *spur* – odstraní koncové pixely výběžků

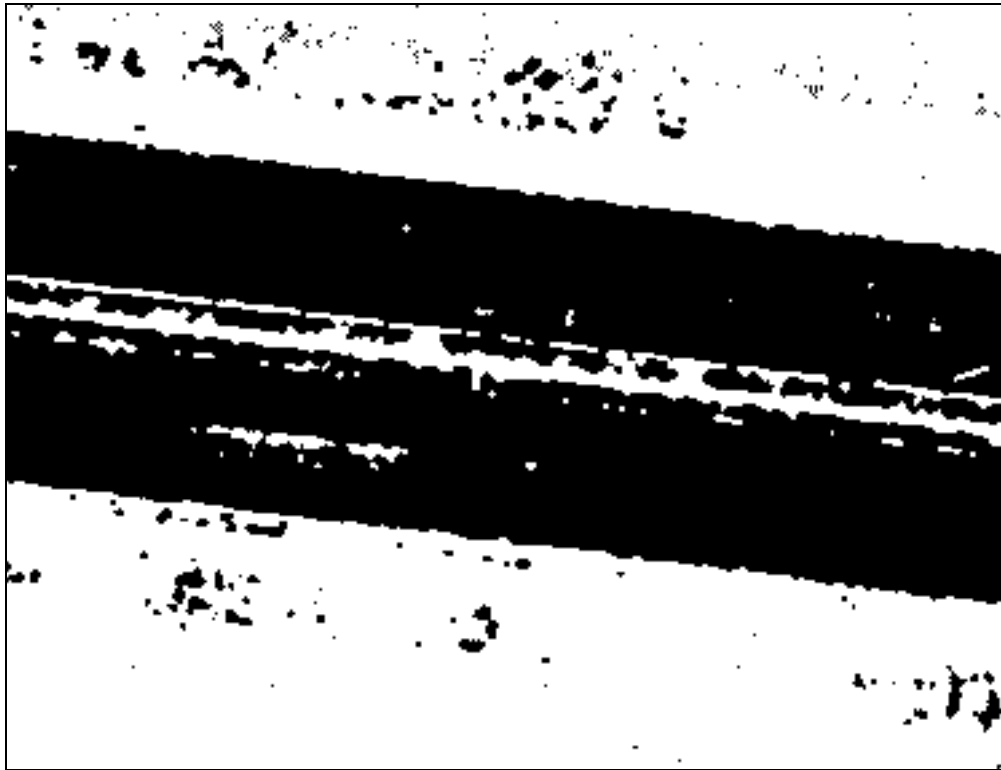
např. $\begin{matrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{matrix}$ upraví na $\begin{matrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{matrix}$

Funkce *bwpropfilt* – ponechává v obraze jen objekty, jejichž vybraný parametr je v daném rozsahu, v tomto případě je vybrán parametr *MajorAxisLength* (délka hlavní poloosy elipsy, jejíž druhý centrální moment – tj. rozptyl je stejný, jako rozptyl objektu) funkce odstraní objekty typu vozidla na vozovce, zbytky šumu apod. (v algoritmu se osvědčilo nastavení minimální hodnoty na 1/3 délky menšího z rozměrů obrazu).

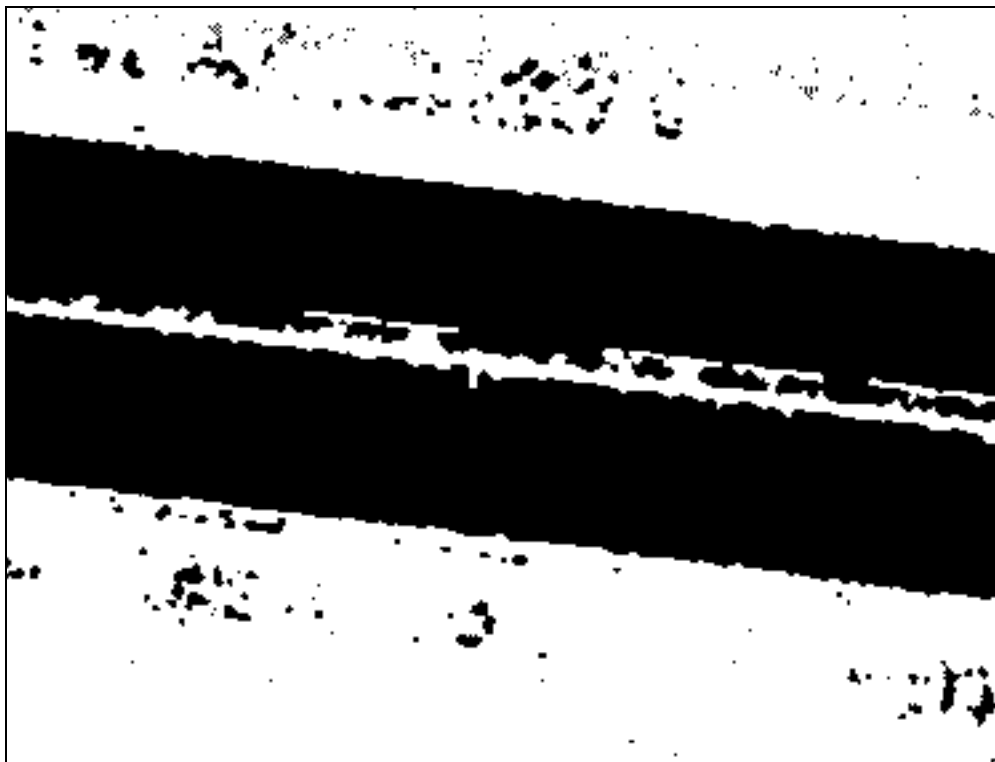
Funkce *imclose* provádí morfologickou operaci uzavření za použití definovaného elementu – funkce spojí objekty, které jsou blízko u sebe, zaplní malé díry a úzké zálivy (v algoritmu je použit kruhový element o průměru 2,00m – zvolen byl kruh pro vytvoření hladkých okrajů objektu, průměr nastaven tak, aby došlo ke smazání děr po středním dělicím pásu).

Funkce *imfill* – vyplní díry v objektech – např. odstraní vozidla z vozovky.

To, jak se aplikace výše jmenovaných funkcí projevuje ve zpracovávaném obraze, je možné sledovat na detailu ze 72km dálnice na obr. 15 – obr. 19. V průběhu zpracování je obraz invertován, aby bylo možné pomocí objektivě orientovaného filtrování (např. dle velikosti) odstranit chybné objekty jak z třídy *roadway* (např. vozidla) tak mimo třídu (rovinné plochy mimo vozovku). Algoritmus vnímá jako objekty vždy shluky bílých pixelů.



Obr. 16 Obraz po aplikaci prvních filtrů (funkce *bwmorph*, operace *clean*, *bridge*, *majority*); 31. km



Obr. 17 Obraz po aplikaci funkce *bwpropfilt* – filtrace objektů na základě délky; 31. km



Obr. 18 Obráz po inverzi a aplikaci funkce *bwmorph*, operace *hbreak* a funkce *bwpropfilt*; 31. km



Obr. 19 Výsledný obráz po finálním shlazení (funkce *bwmorph*, operace *spur*, funkce *imclose* a *imfill*); 31. km

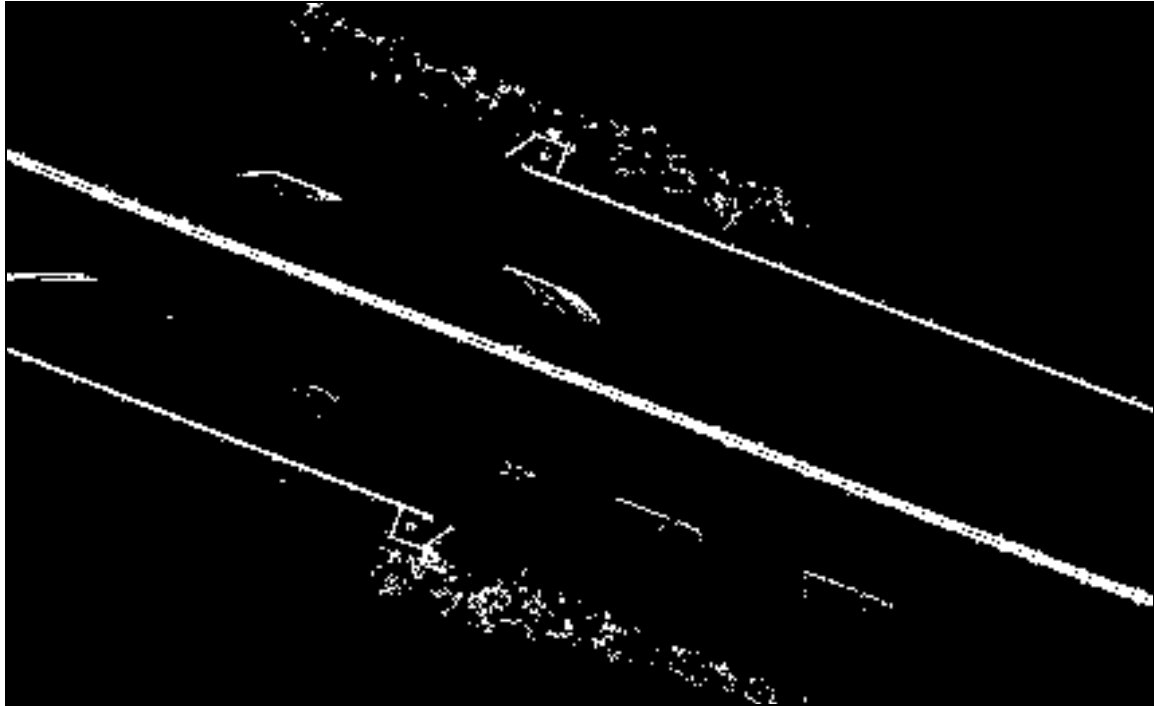
4.3.2 Crash barriers – svodidla

Klasifikace vychází z předpokladu, že svodidla se nachází v bezprostřední blízkosti vozovky – tj. třídy *roadway* a jde o liniové prvky s určitými danými výškovými parametry. Dle technických podmínek pro svodidla stanovených Ministerstvem dopravy je minimální výška svodidla na pozemních komunikacích stanovena na 0,75m (*TP 114, 2015*).

Algoritmus vyhledává pixely obsahující body ve výšce 0,50-1,00m nad terénem. Minimální limit je stanoven tak, aby do výběru nezasahoval šum a traviny, maximální limit tak, aby do výběru nezasahovaly dopravní značky. Spodní okraj nejnižší umístěné standardní stálé značky vedle vozovky (včetně dodatkové tabulky) je dle předpisů nejméně 1,20 m nad úrovní vozovky (*TP 65, 2013*). Z těchto pixelů jsou vybrány pouze objekty, které se alespoň z části nachází v klasifikované oblasti *roadway* a jejím bezprostředním okolí (limit vzdálenosti od třídy *roadway* je na základě zkušenosti z testovacích souborů stanoven na 3,00m). Většina svodidel se nachází do této vzdálenosti, mimo částí ohraničujících bezpečnostní zálivy a SOS hlásky, ale ty jsou s bližšími částmi svodidel spojeny. Vzniklý obraz (obr. 20) může obsahovat oříznuté pruhy souvislé vegetace poblíž svodidel a vozidla. Oba typy těchto objektů mohou tvořit liniové prvky podobné svodidlům a je třeba je z obrazu odstranit.

Odstranění souvislé vegetace se provede následujícím postupem. Vytvoří se rastr zobrazující pixely s body výše než 0,50m (obr. 21) a odstraní se ty, spadající do oblasti *roadway* a jejího nejbližšího okolí – primární obraz vegetace. Na obraz pixelů obsahujících alespoň jeden bod ve výšce svodidel (0,50-1,00m nad terénem) se aplikuje funkce *ordfilt*, k odstranění svodidel. Funkce *ordfilt* hodnotu každého pixelu obrazu určí na základě hodnot pixelů v okolí (hodnoty pixelu v okolí 3x3 jsou seřazeny od nejmenší po největší a nová hodnota pixelu je určena jako n-tá hodnota z okolních pixelů). V tomto případě je filtrace obrazu provedena funkcí *ordfilt* s nastavením nové hodnoty jako nejnižší okolí - filtr odstraní všechny liniové prvky do tloušťky dvou pixelů a drobné objekty. Z výsledného obrazu se vyberou všechny objekty, které mají společnou část s primárním obrazem vegetace mimo *roadway* a okolí (výsledek na obr. 22). K tomuto účelu slouží funkce *imreconstruct*, která ze dvou binárních obrazů (*marker* a *mask*) vytvoří jeden tak, že výsledný obraz obsahuje všechny objekty z obrazu *mask*, které mají společný nebo sousední pixel z nějakého objektu v obraze *marker*.

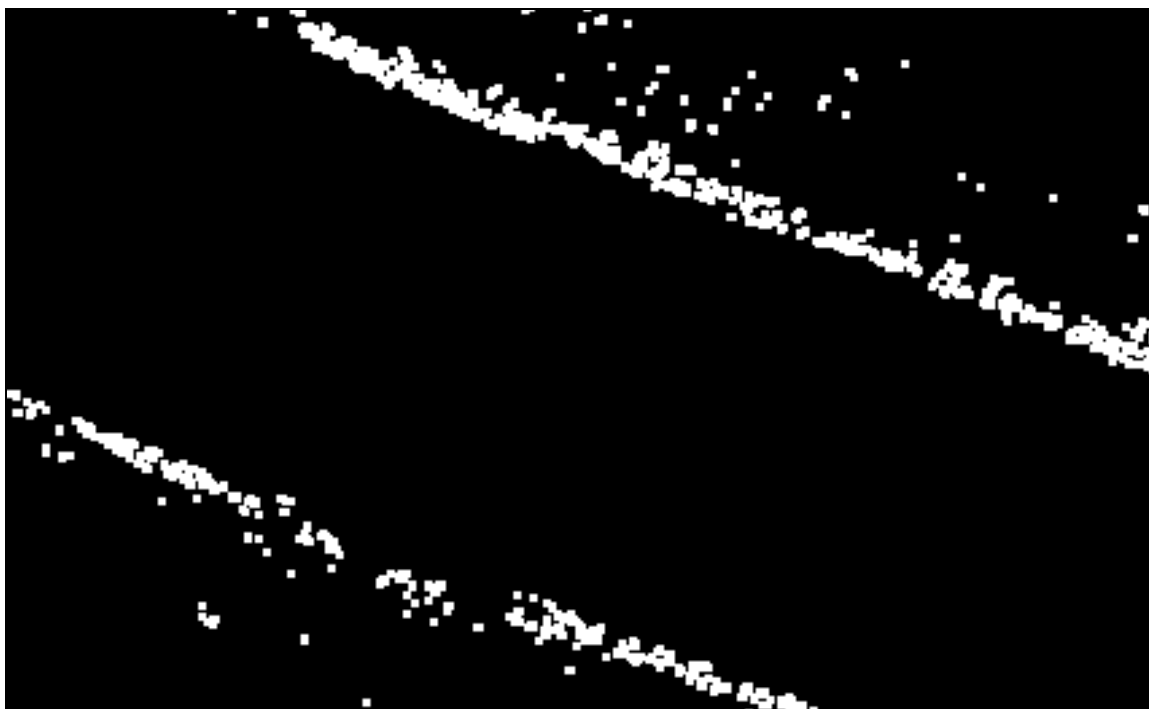
Výsledný obraz se odečte od obrazů bodů potenciálně příslušejících k třídě *crash barriers*. Tento sofistikovaný postup, dokáže odstranit vegetaci bez toho, aby byly zároveň odstraněny části svodidel, přes které vegetace přesahuje a dokáže alespoň částečně oddělit nízkou vegetaci přiléhající ke svodidlům (obr. 21).



Obr. 20 Svodidla – pixely vybrané na základě výšky nad terénem nad *roadway* a okolím; 26. km



Obr. 21 Pixely obsahující body 0,5m nad terénem a výš; 26. km

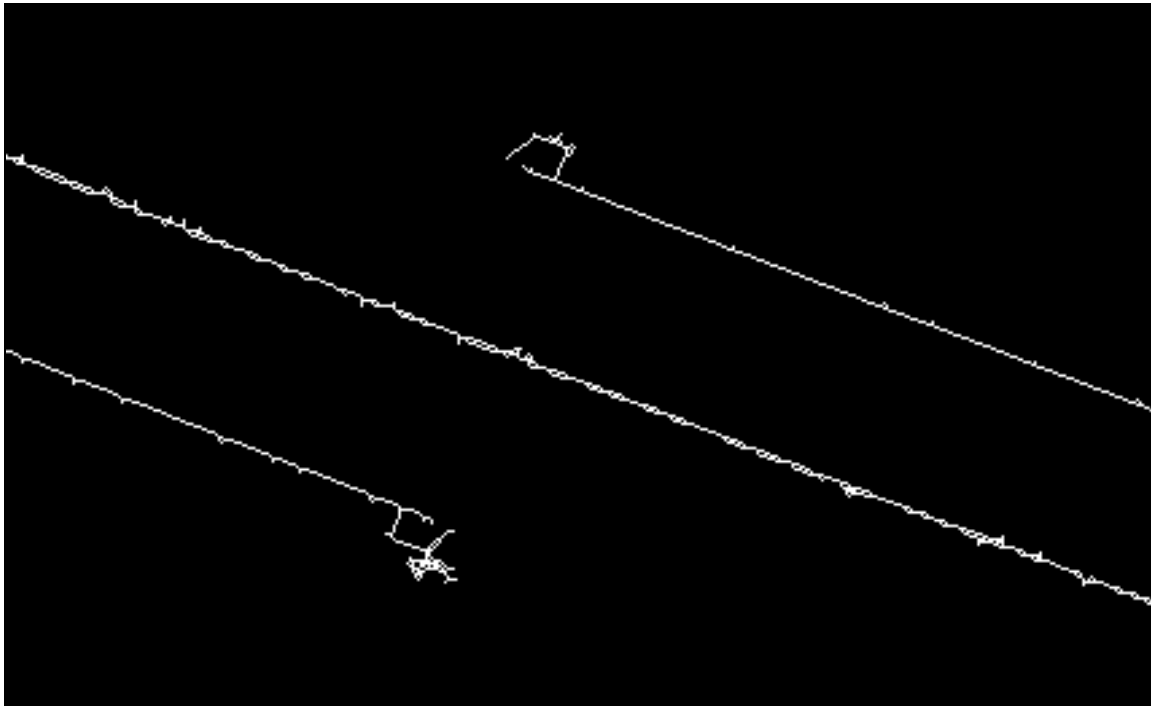


Obr. 22 Hrubý obraz vegetace v úrovni svodidel – pro odstranění; 26. km

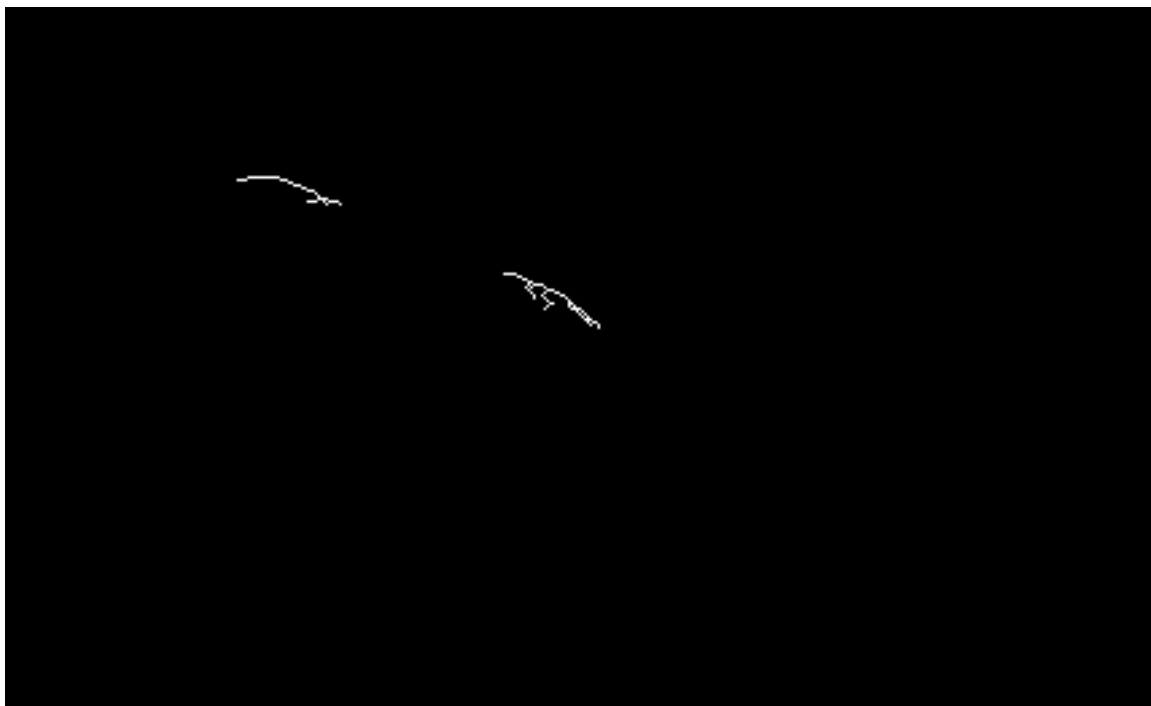
Z vozidel jsou největší komplikací dlouhé a relativně pomalu se pohybující nákladní vozy, tato vozidla mohou předjíždět skenovací vůz i několik desítek metrů a zanechají v mračnu dlouhou liniovou stopu. Nákladní vozidla bývají vyšší než svodidla (pro maximální výšku svodidel se nepodařilo najít limitní hodnotu, vychází se z pozorování). Vyhledání těchto vozidel v obraze probíhá následovně: z pixelů klasifikovaných jako *roadway* se vyberou pixely s body ve výšce 3,00-4,50m nad terénem. Dolní hranice je zvolena vyšší než výška lehkých užitkových vozidel (2,80m) a výška svodidel na testovaných souborech, horní hranice vyšší než výška soupravy tahače s návěsem (4,20m) a nižší než nejnižší objekt vyskytující se nad dálnicí (spodní okraj značky 5,20 m). Při stanovení výšky vozidel se vychází z orientačních rozměrů vozidel definovaných jako podklad pro rozměry parkovacích stání a garáží (ČSN 73 6058, 2011) a výška spodního okraje značky nad vozovkou vychází z TP 65 (2013). Takto získaný obraz se odečte od obrazu znázorňujícího svodidla.

V dalším kroku jsou zbylé objekty vybírány na základě délky a šířky (resp. největšího a nejmenšího rozptylu pixelů. Pro každý objekt se vypočte elipsa, jejíž druhý centrální moment, tj. rozptyl odpovídá rozptylu objektu (funkce *bwpropfilt*, s parametry – *MajorAxisLength* a *MinorAxisLength*). Filtrace na základě rozptylu ve směru menší poloosy má odstranit zejména objekty zobrazující vozidla – vozidlo bývá skenováno z boku a zepředu/zezadu, v obraze se projevuje ve tvaru rovné linie zakončené obloukem (obr. 24). Aby byla velikost rozptylu ve směru menší poloosy dána tvarem a ne šířkou objektu (např. svodidla středního dělicího pásu bývají spojena v jeden širší objekt), redukuje se jednotlivé objekty na kostru za použití funkce *bwmorph* operace *skel* (postupně odstraňuje pixely na okraji objektu, až na šířku jednoho pixelu, ale nedovolí rozdělení objektu). Kostry objektů jsou filtrovány na základě rozptylu ve směru hlavní poloosy (u úsečky odpovídá rozptyl její délce) – objekty delší než 40,00m jsou rovnou považovány za svodidla (obr. 23). U objektů s rozptylem ve směru hlavní poloosy mezi 10,00-40,00m

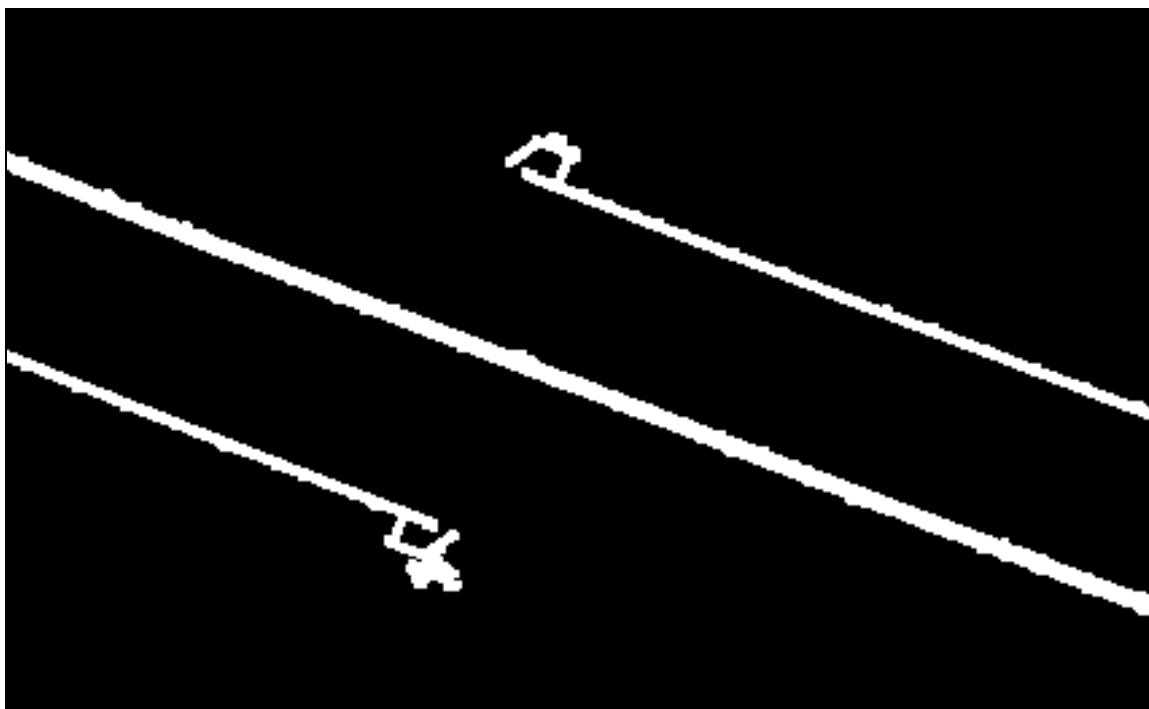
(obr. 24) se testuje ještě rozptyl ve směru vedlejší poloosy. U vozidel a zbytků vegetace byla testováním určena hranice maximálního rozptylu ve směru malé poloosy 0,50m. Výslednou třídu *crash barriers* tvoří objekty, jejichž kostra splnila výše zmíněná kritéria.



Obr. 23 Objekty s maximálním rozptylem nad 40m; 26. km



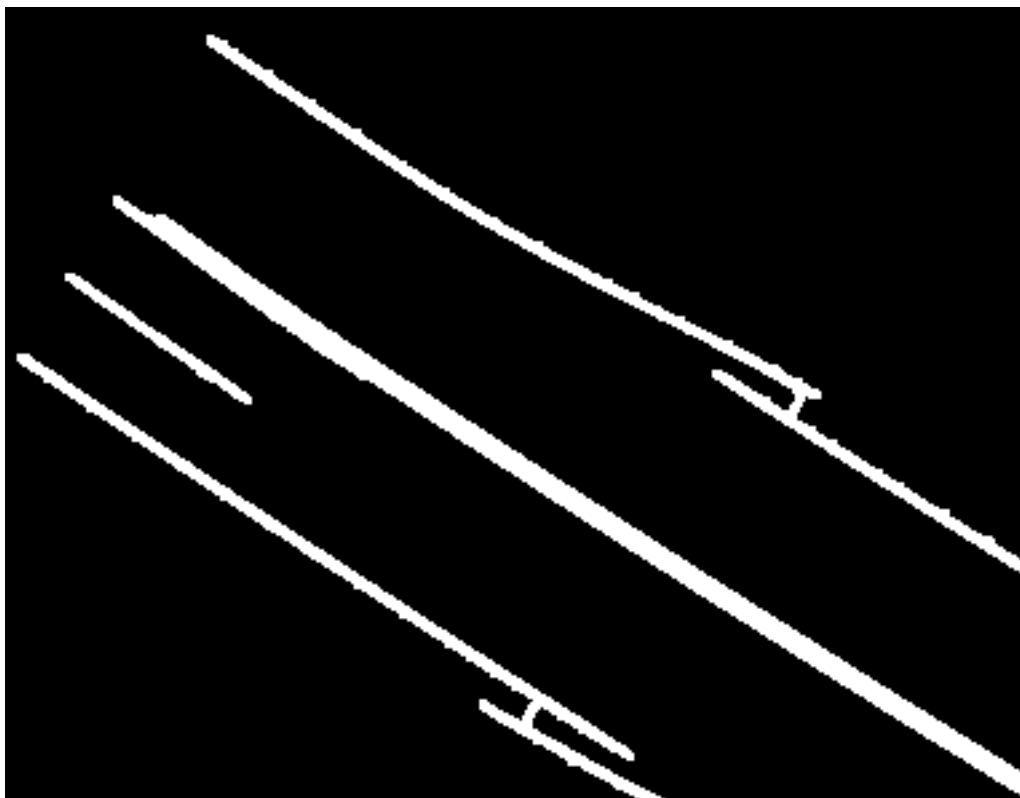
Obr. 24 Objekty s maximálním rozptylem mezi 10-40m; 26. km



Obr. 25 *Crash barriers* – svodila; 26. km

Hranice 40m byla stanovena na základě zkušenosti. Odstranění souprav tahačů a souvislých pásů vegetace snížilo množství takto dlouhých objektů v obraze, ale neeliminovalo je zcela. Objekty delší než 40m není vhodné filtrovat na základě rozptylu v menším rozměru/šířce, protože u delších částí svodidel může být tento parametr ovlivněn tvarem svodidel kopírujícím zakřivení trasy pozemní komunikace. U objektů kratších než 5m nestačí ani omezení šířkou a jsou proto z klasifikace zcela vynechány.

Z předchozího vyplývá, že klasifikace selhává pro úseky kratší 40-ti metrů, pokud jsou výrazným způsobem zakřivené, nebo pro úseky kratší 5-ti metrů. Minimální délka svodidla není obecně právními předpisy stanovena, přesto zpravidla přesahuje hranici 40m. Problém nastává u krátkých úseků vytvořených rozdělením datového souboru do bloků. Takto se může stát, že nebude v ojedinělých případech klasifikována část svodidla. Zároveň nastavení algoritmu nezabrání tomu, aby byly v rámci třídy *crash barriers* klasifikovány v některých případech i části projíždějících vozidel (viz obr. 26). Přísnější nastavení kritérií vedlo k navýšení chyb I. typu, přičemž chyby v klasifikaci II. typu jsou v tomto případě hodnoceny jako přijatelnější, než chyby I. typu. Při analýze bezpečnosti provozu je snazší odstranit ze třídy *crash barriers* body zobrazující např. vozidlo než dohledávat body svodidel v jiných třídách.



Obr. 26 Svodidla – chybně klasifikované vozidlo; 72. km

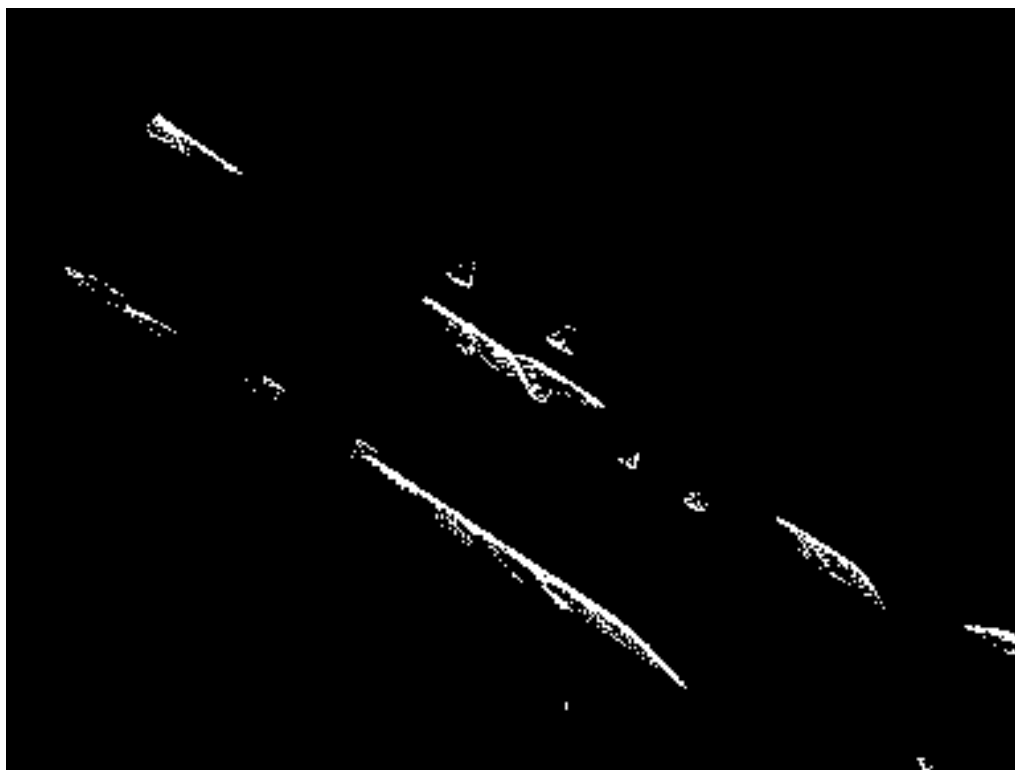
4.3.3 Vehicles – vozidla

Třída *vehicles* obsahuje dočasné objekty nad vozovkou, které nemají být předmětem dokumentace, tj. projíždějící/stojící vozidla. Tato vozidla je třeba z mračna odstranit např. pro provádění analýzy viditelnosti dopravního značení.

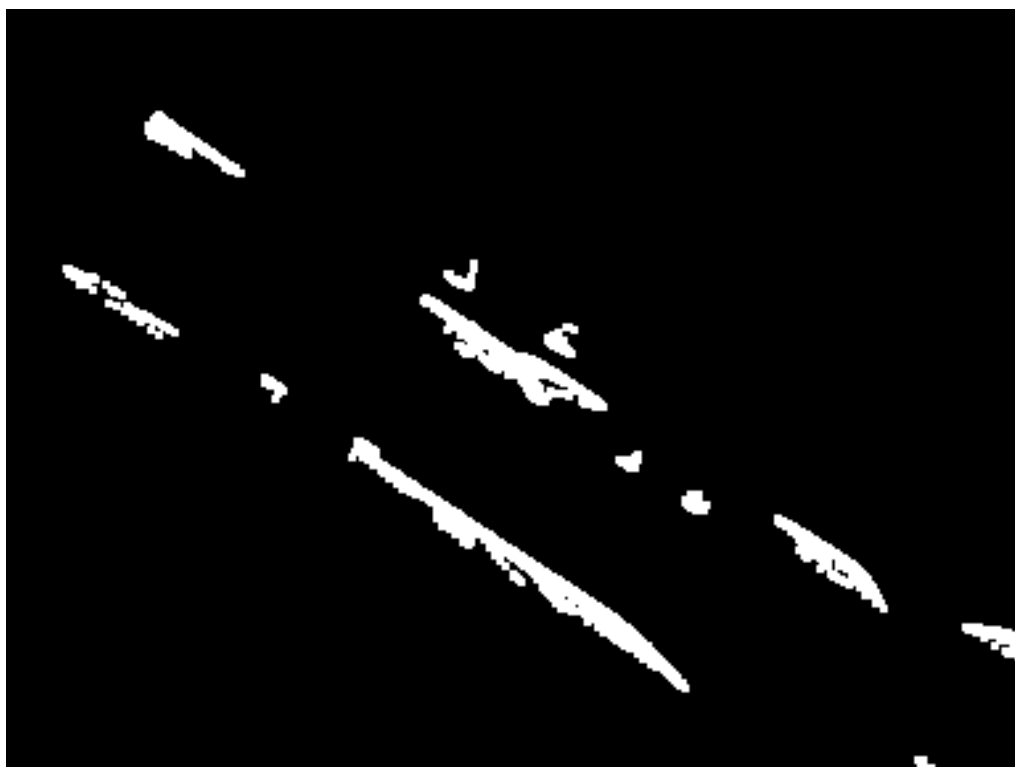
Algoritmus vychází z toho, že hledá pixely obsahující body ve výšce 0,50-4,50m nad vozovkou (body výše než traviny v nejbližším okolí vozovky a níž než objekty překlenující vozovku – mýtné brány apod.), které nebyly klasifikovány jako svodidla.

Výše zmíněné podmínky mohou mimo pixelů zobrazujících vozidla splňovat i pixely představující tyčové objekty na zpevněném povrchu navazujícím na vozovku – SOS hlásky a směrové sloupky (obr. 27Obr. 27, dole uprostřed). Tyto předměty patří do třídy *poles* a je třeba je z obrazu odstranit. Eliminace těchto objektů je možná na základě jejich rozměru v půdorysu. Dolní limit pro maximální rozptyl bodů objektu ponechávaného ve třídě *vehicles* je stanoven na 0,80m, tato hodnota přesahuje maximální úhlopříčku standardních typů SOS hlásek (na základě pozorování testovacích dat a dostupného přehledu typů SOS hlásek společnosti Spel – Spel (2017)). Pixely zobrazující vozidla při daném rozlišení 0,20m netvoří vždy souvislé celky (obr. 27Obr. 27). Pro spojení nesouvislých částí je na obraz aplikován filtr *ordfilt2*, který nahradí hodnotu pixelu n-tou hodnotou (seřazeny od nejmenší po největší) v okolí o velikosti 3x3 pixely. V tomto případě je nejvyšší hodnota, tj. každý pixel, v jehož okolí 3x3 pixely se vyskytuje alespoň jeden bod třídy *vehicles*, je k této třídě přidán. Teprve po této proceduře je provedena

filtrace objektů na základě velikosti, tj. rozptylu bodů (stanovený limit maximálního rozptylu je zvětšen o vliv filtru *ordfilt2* na velikost objektů) – viz obr. 28.



Obr. 27 Třída *vehicles* před aplikací filtrů; 72. km



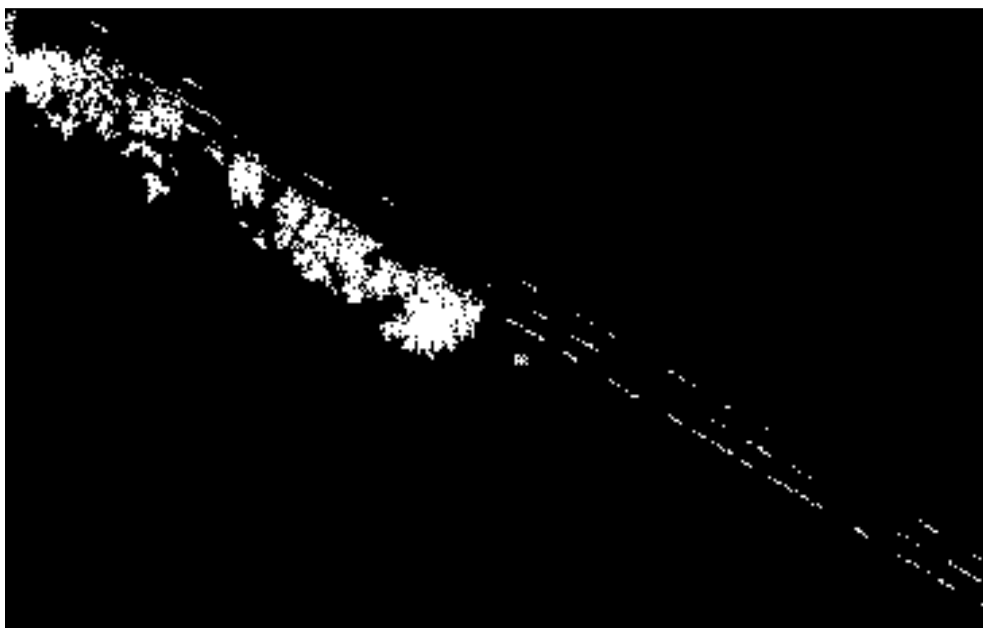
Obr. 28 Třída *vehicles* – 72. km

4.3.4 Gates – objekty překlenující komunikaci

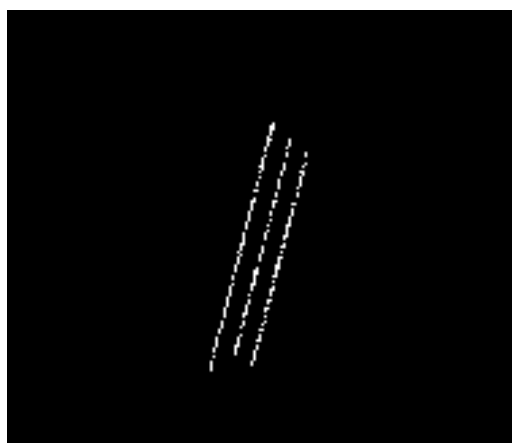
Třída *gates* obsahuje jakékoli objekty překlenující vozovku – elektrické vedení, mýtné brány, nosiče svislého dopravního značení, mosty a tunely, kterými je komunikace vedena (např. z důvodu stavby ekoduktu). Z uvedených příkladů je zřejmé, že jde o množství poměrně různorodých objektů. Objekty jsou v rámci třídy *gates* rozděleny do tří podskupin, k jejichž klasifikaci je třeba přistupovat odlišně – první tvoří elektrická vedení, druhou mýtné brány a nosiče svislého dopravního značení a poslední mosty a tunely.

Elektrické nadzemní vedení je vzhledem ke své malé tloušťce v mračnu reprezentováno jen velmi malým množstvím bodů, což se promítá i do obrazu (obr. 29 – zobrazené vedení se nachází ve vzdálenosti 8-10m od vnějšího okraje vozovky). V obraze o zvoleném rozlišení (0,20m) netvoří pixely zobrazující elektrické vedení mimo vozovku souvislé objekty a odlišení pixelů zobrazujících vedení a části vegetace by bylo značně problematické. Další postup se soustředí na nalezení elektrického vedení pouze nad vozovkou a jejím nejbližším okolím.

Pozice elektrického nadzemního vedení je upravena předpisy, přičemž minimální výška elektrického nadzemního vedení nad terénem (nad dálnicí) je stanovena na 7,00m (ČSN EN 50 341 -1 ed. 2, 2013). Obraz použitý pro klasifikaci vybírá pixely s body nacházejícími se nad vozovkou a jejím nejbližším okolím (okolí definováno pásem o stejné šířce jako u svodidel a značek – 7,00m) ve výšce 7,00m a víc nad terénem. Obraz může mimo vedení obsahovat části vegetace a větší objekty překlenující vozovku. Odstranění případné vegetace a dalších objektů proběhne tak, že na primární obraz se aplikuje mediánový filtr (*medfilt2* – nahradí hodnotu pixelu mediánem v okolí 3x3), tím se z obrazu odstraní veškeré tenké liniové a izolované prvky, takto vzniklý obraz se od původního odečte. Výsledný obraz obsahuje body elektrického vedení, případně okrajů vegetace. Aby bylo možné oddělit okraje vegetace od elektrického vedení, aplikuje se na obraz filtr *ordfilt* ($n = 9$), čímž se kolem všech objektů v obraze vytvoří nárazníková zóna (*buffer*) o šířce jednoho pixelu a pixely zobrazující elektrické vedení se spojí do větších celků. Vzniklé objekty jsou poté filtrovány na základě parametrů elipsy rozptylu hodnot – minimální délky velké poloosy (stanoveno na 2m) a excentricitě elipsy (poměr malé k velké poloose je menší než 1/3), tím se odstraní zbytky vegetace. Na závěr se z primárního obrazu vyberou objekty, které jsou alespoň částečně obsaženy v obraze filtrovaném (návrat k originálním datům) a objekty ve výsledném obraze jsou filtrovány na základě maximální hodnoty malé poloosy elipsy rozptylu hodnot (filtrace odstraní případnou vegetaci, která by se mohla nacházet v bezprostřední blízkosti vedení) – obr. 30.



Obr. 29 Pixely obsahující body výše než 7,00m nad terénem; 18. km



Obr. 30 Elektrické vedení nad vozovkou; 18. km

Klasifikace ostatních podtříd pracuje s body 4,50m nad terénem a výš, hranice byla stanovena tak, aby se nacházela nad nákladními vozidly a pod nejnižší částí objektů nad vozovkou. Pro značku umístěnou nad vozovkou platí, že spodní okraj značky včetně dodatkové tabulky je nad nejvyšším bodem vozovky nejméně: 5,20 m u dálnic (*TP 65, 2013*). Výška soupravy tahače s návěsem je (pro potřeby projektování garáží) stanovena na 4,20m (*ČSN 73 6058, 2011*).

Mýtné brány a značky nad vozovkou se od objektů typu mostů a tunelů liší zejména svou velikostí, mýtné brány ani značky umístěné na rameni nad vozovkou nemusí překlenovat celou šířku pozemní komunikace. Na druhou stranu jsou oproti mostům umístěné vždy kolmo na osu komunikace. Při klasifikaci mýtných bran a značek nad vozovkou se vychází z pixelů odpovídajících třídě *roadway* obsahujících zároveň body 4,50m nad terénem a výš. Zobrazené objekty se filtrují na základě minimální délky 2,00m (velikosti hlavní poloosy elipsy rozptylu pixelů v objektu) a na základě orientace (objekt je orientován kolmo na objekt třídy *roadway* s tolerancí 10°, funkce *bwpropfilt*, parametr *Orientation*),

filtrace na základě těchto podmínek zajistí odstranění případného elektrického vedení a částí vegetace. V posledním kroku je třeba připojit k nalezeným objektům i části mimo oblast *roadway*. Z obrazu bodů nad danou výškovou hranici (4,50m, v celém obraze) se vyberou všechny objekty, které mají průnik s objekty vybranými nad třídou *roadway* (obr. 31).

Klasifikace mostů a tunelů začíná stejně jako u předchozí podtřídy z pixelů odpovídajících třídě *roadway* obsahujících zároveň body 4,50m nad terénem a výš. Liší se další postup filtrace, v obraze jsou ponechány objekty, u nichž délka – resp. hlavní poloosa elipsy rozptylu pixelů dosahuje alespoň poloviny šířky – resp. vedlejší poloosy elipsy rozptylu pixelů objektu třídy *roadway*. Z principu musí délka části mostu nad vozovkou odpovídat minimálně šířce vozovky (kterou má překlenout), limit je nastaven na polovinu této hodnoty z důvodu možného rozšíření vozovky v blízkosti mostu (např. o připojovací pruhy). Připojení částí objektů mimo plochu *roadway* probíhá stejně jako v předchozím případě (obr. 33).

V následujícím kroku jsou všechny podtřídy spojeny do jedné třídy *gates*.

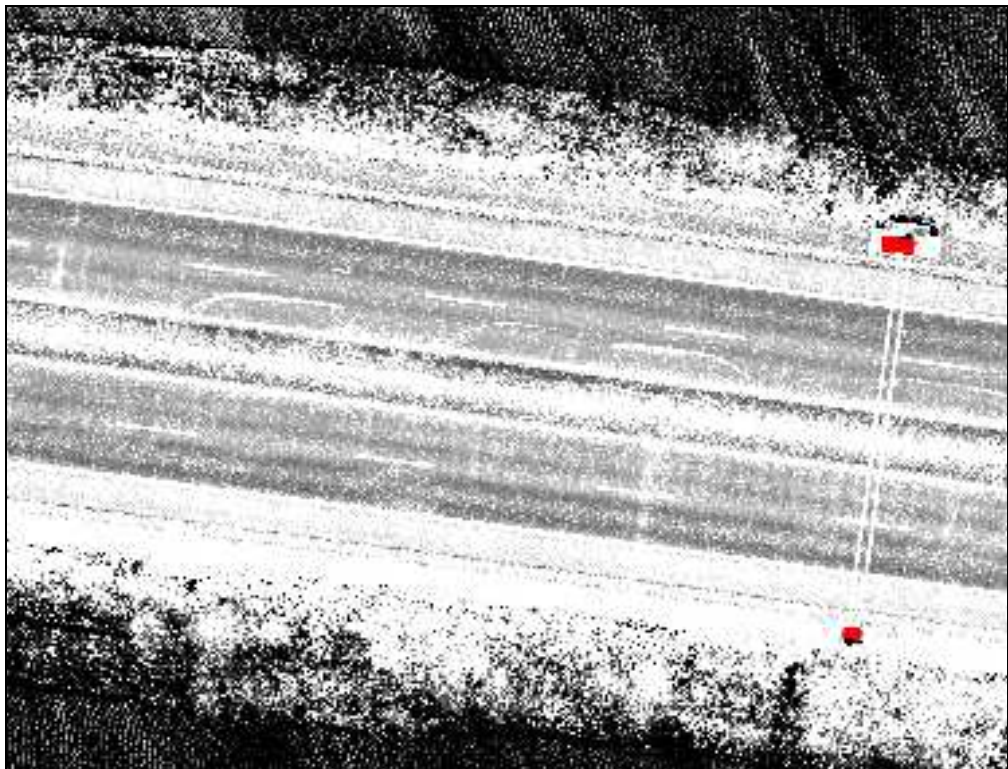
Dosud se pracovalo s předpokladem, že body třídy *gates* se nachází pouze nad úrovní 4,50m nad terénem a podle této výškové hranice lze rozřadit při klasifikaci mračna body, které jsou v obraze klasifikované do třídy *gates* a zároveň např. do třídy *crash barriers*, nebo *vehicles*. Tato premisa neplatí pro konce mostů, kde se body mostu přibližují terénu, respektive terén se zvedá do úrovně mostu. Zároveň je třeba nalézt v obraze sloupy mostů, které je třeba klasifikovat také ve výšce pod 4,50m nad terénem.

Klasifikace okrajů mostů a bočních zdí tunelů začíná výběrem pixelů obsahujících body ve výšce 1,00-4,50m nad terénem (spodní hranice je nastavena tak, aby nebyla ovlivněna nízkou vegetací, horní hranice odpovídá výšce, nad kterou se nachází body řešené v předchozích krocích). Z tohoto obrazu jsou odstraněny pixely tříd *vehicles* a *crash barriers* a objekty s maximálním rozptylem pixelů menším než 1,00m (odstraní zbývající šum z vozidel nad vozovkou, nastavená hodnota je menší než minimální volná šířka trvalé lávky – 2,00m dle (ČSN 73 6201, 2008)). Z objektů splňujících výše zmíněná kritéria jsou vybrány ty, které mají průnik s objekty třídy *gates* (obr. 34).

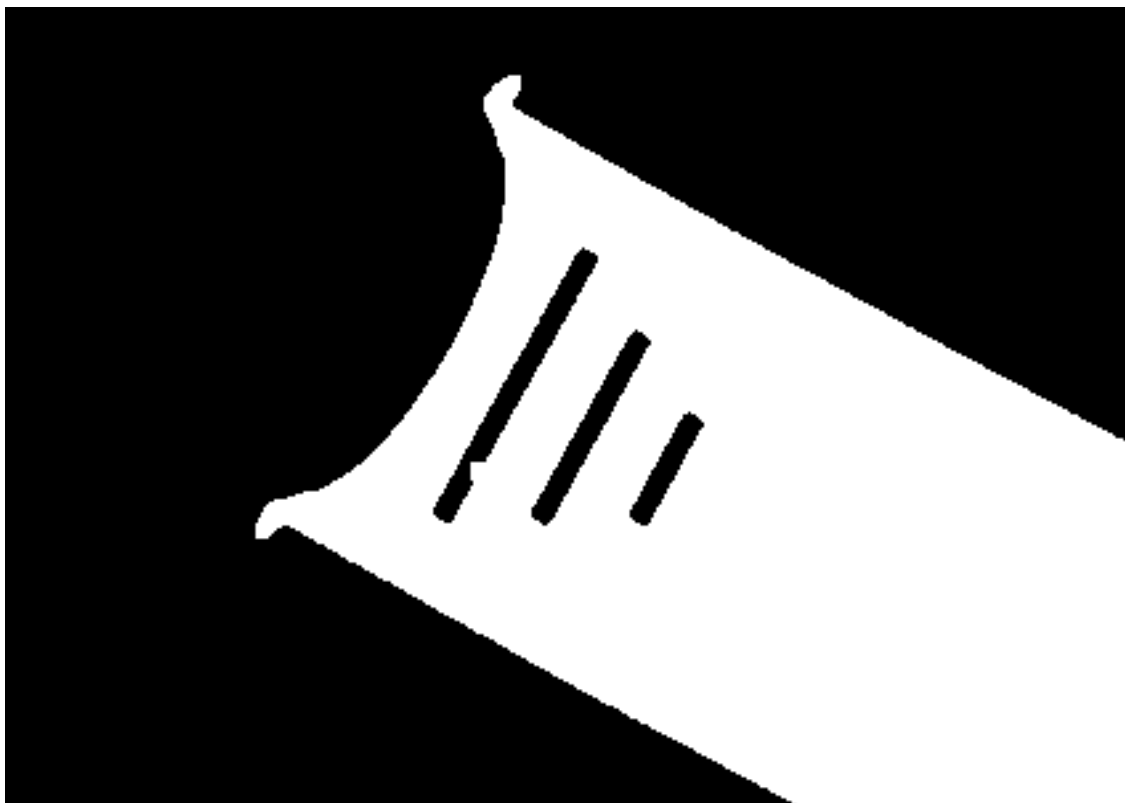
Sloupy mostů a mýtných bran jsou řešeny zvlášť, protože menší sloupy se v obraze jeví podobně jako menší vozidla (mohou být stejně klasifikovány) a není možné oddělit je na základě rozměru, tvaru nebo polohy (okolí sloupu může být pokryto rovným hladkým povrchem a klasifikováno v třídě *roadway*, jako u *vehicles*). Klasifikační algoritmus vybírá pixely obsahující body ve výšce 2,00-4,50m (spodní hranice nastavena vyšší než výška osobních vozidel – 1,80m dle kategorií stanovených ČSN 73 6058 (2011), horní hranice odpovídá výšce, nad kterou se nachází body řešené v předchozích krocích). Z obrazu jsou odstraněny pixely tříd *vehicles* (pouze objekty s maximálním rozptylem větším než 6.00m – k odstranění větších vozů) a *crash barriers*. Z objektů splňujících výše zmíněná kritéria jsou vybrány ty, které mají průnik s objekty třídy *gates* (obr. 32). Zmíněný postup může způsobit, že budou ve třídě *gates* klasifikovány i objekty zobrazující vozidla, ale minimalizuje se tak riziko, že sloupy klasifikovány nebudou. Problém nastává v případě šikmých sloupů, u kterých je možné, že spodní část sloupu zůstane neklasifikovaná (obr. 35, obr. 36).



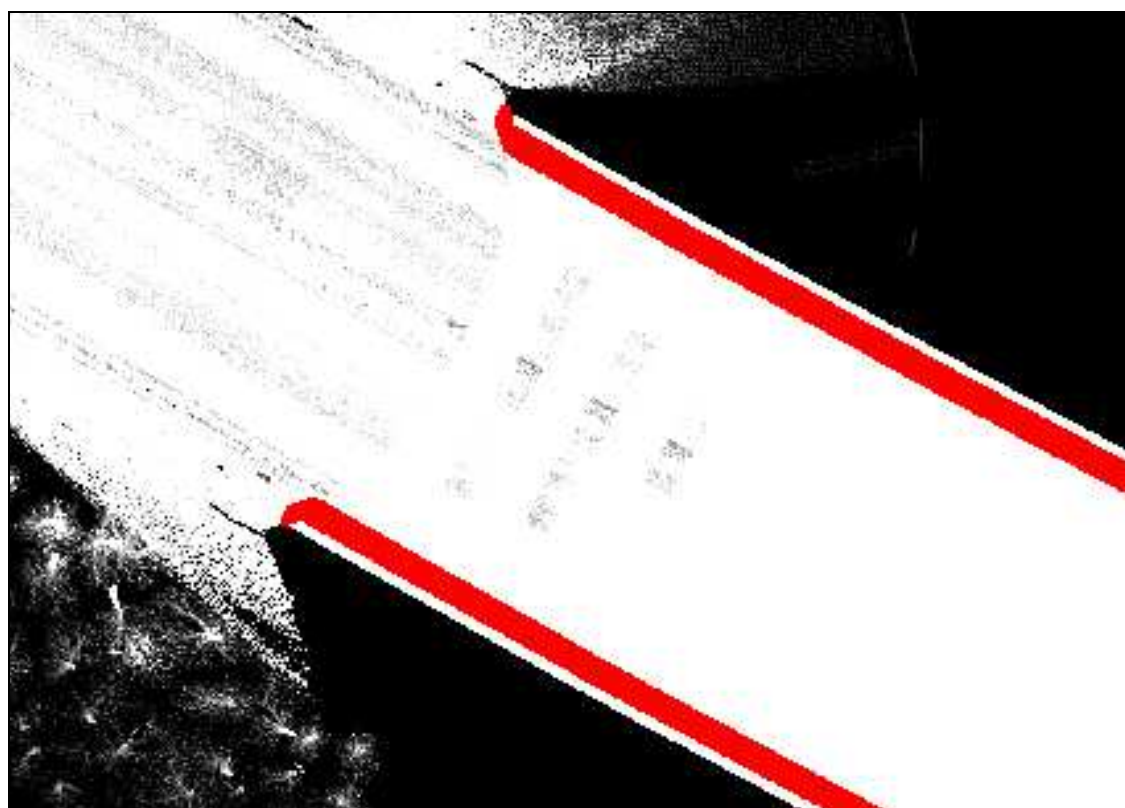
Obr. 31 Mýtná brána; 31. km



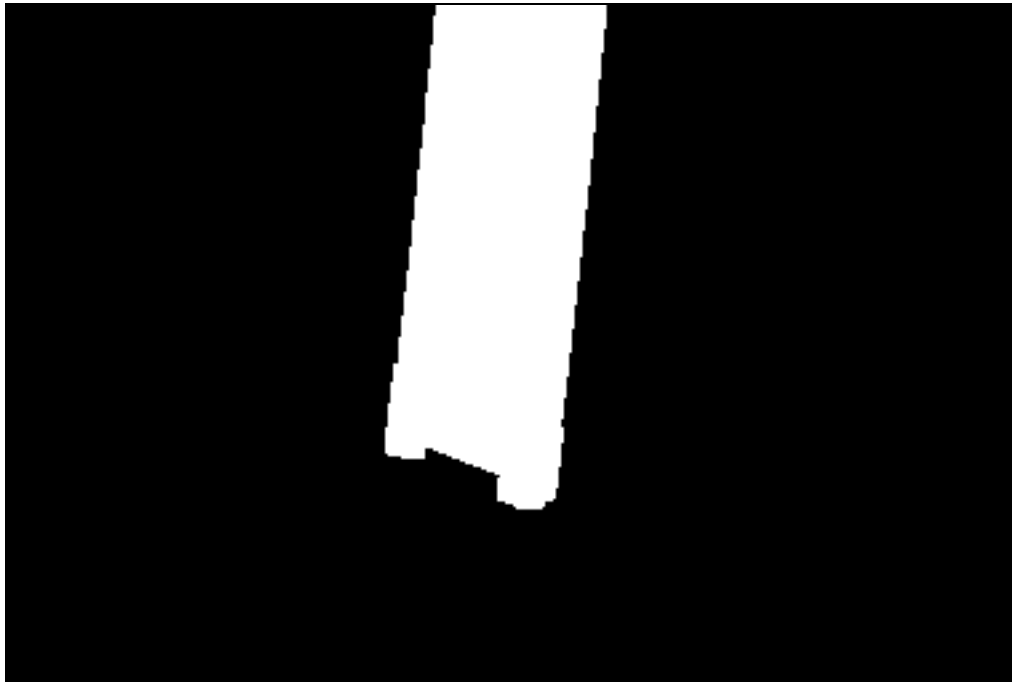
Obr. 32 Sloupy mýtné brány; 31. km



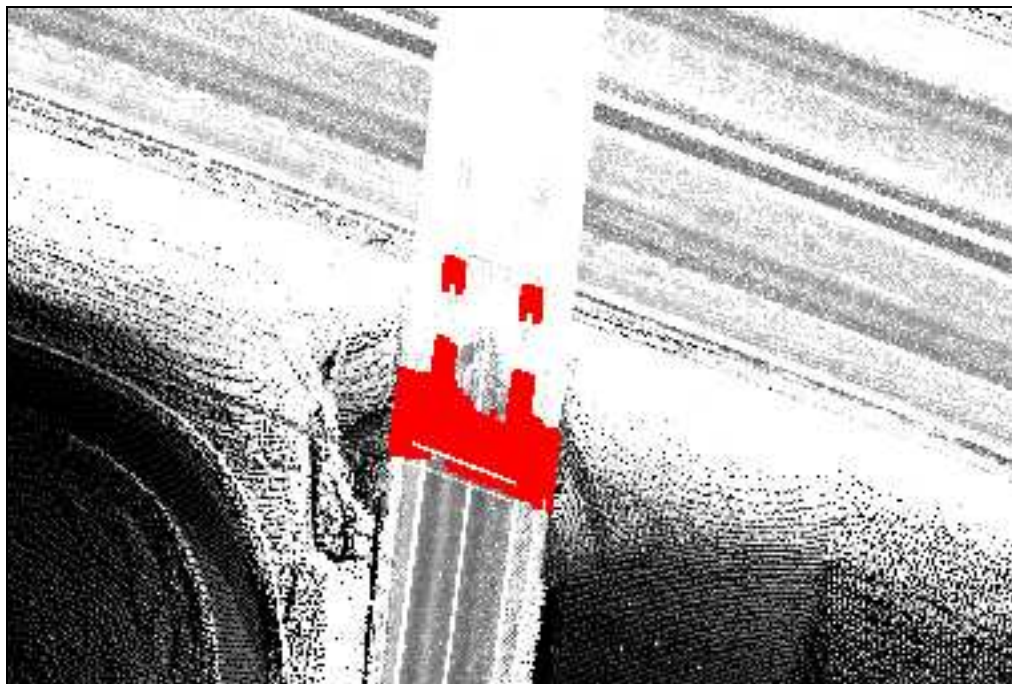
Obr. 33 Ekodukt - třída gates; 72. km



Obr. 34 Ekodukt - boky tunelu na pozadí *intensity image*; 72. km



Obr. 35 Most; 62. km (b)



Obr. 36 Konce mostu, sloupy; 62. km (b)

4.3.5 Signs – svislé dopravní značky a jiné tabule

Třída *signs* zahrnuje samostatně stojící svislé dopravní značky, reklamní a jiné velkoformátové tabule. Svislé dopravní značky a tabule umístěné nad vozovkou jsou součástí objektů překlenujících vozovku a vzhledem k základnímu principu klasifikace (klasifikace rastrů vzniklých průmětem do vodorovné roviny) by byly těžko oddělitelné, jsou tedy ponechány ve třídě *gates*. Objekty třídy *signs* jsou

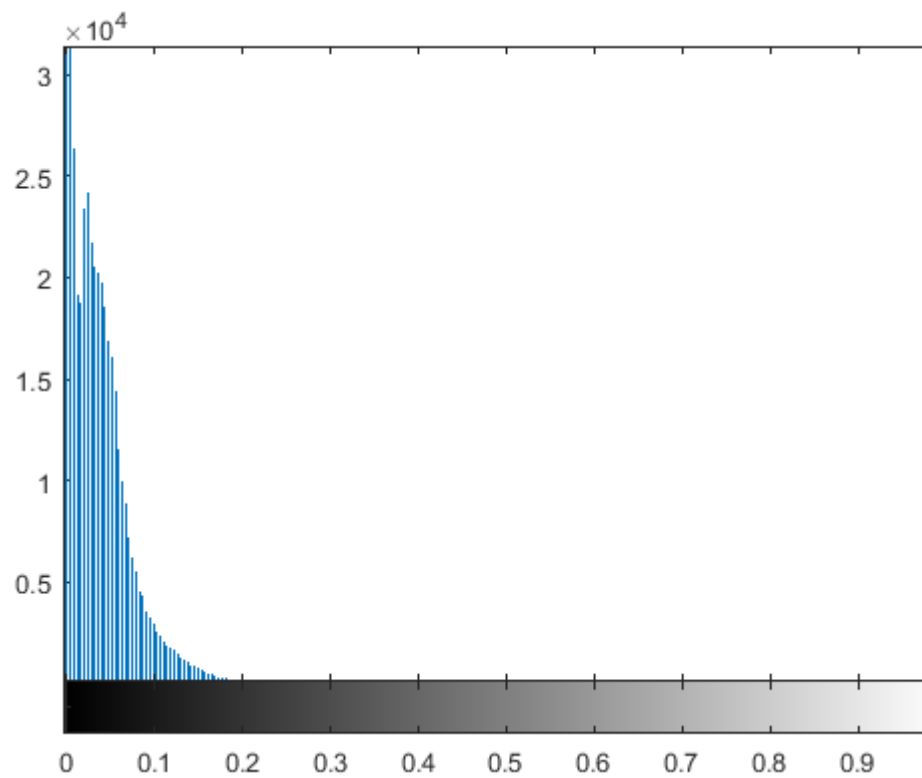
charakterizovány vyšší odrazivostí, obsahují body v určitém rozmezí nad terénem a zobrazují se jako krátké liniové objekty přibližně kolmé na osu pozemní komunikace, z těchto informací lze vycházet při jejich klasifikaci.

Umístění svislých dopravních značek vzhledem k pozemní komunikaci upravuje *TP65 (2013)* polohově: „Nejmenší vodorovná vzdálenost bližšího okraje svislé značky, dopravního zařízení včetně jejich nosné konstrukce od vnějšího okraje zpevněné části krajnice, případně od vozovky (u pozemní komunikace bez zpevněné části krajnice), je 0,50 m; největší vzdálenost je 2,00 m,“ i výškově: „Spodní okraj nejnižše umístěné standardní stálé značky – vedle vozovky (včetně dodatkové tabulky) je nejméně 1,20 m nad úrovní vozovky.“

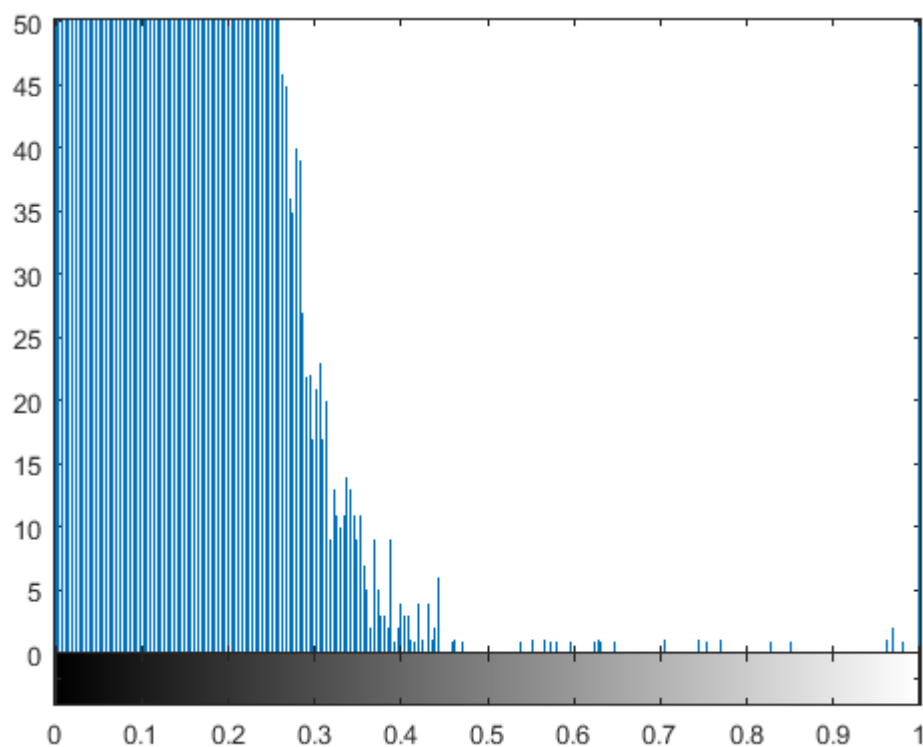
Algoritmus v prvním kroku vybírá pixely obsahující body s výškou nad terénem v daném rozmezí 1,20-4,50m. Spodní hranice odpovídá minimální výšce spodní hrany značky nad vozovkou – předpokládá se, že terén bezprostředně za krajnicí nestoupá, ale naopak klesá z důvodu nutnosti odvodnění vozovky, tj. pokud je bod 1,20m nad terénem je zároveň minimálně 1,20m nad vozovkou. Horní hranice je stanovena tak, aby byla nižší než výška objektů překlenujících vozovku).

Další krok vychází z předpokladu, že svislé dopravní značky a reklamní tabule mají vyšší míru odrazivosti než okolí. Pozorování na testovacích datech ukázalo, že intenzita u bodů získaných odrazem od reflexních dopravních značek převyšuje intenzitu okolních bodů mnohonásobně, u reklamních tabulí není rozdíl intenzity oproti okolí tak výrazný jako u značek, ale přesto je významný. Z pixelů obsahujících body o vybrané výšce proto algoritmus vybere ty, které obsahují body s intenzitou vyšší než je daná hranice (postup zvaný prahování). Otázkou je, jak stanovit tuto hranici – práh. Nejjednodušší možností je manuální nastavení konkrétní hodnoty. Druhou variantou je automatické určení na základě statistického testování obrazu. Do takové kategorie patří např. metoda Otsu (práh je stanoven tak, aby rozdělil soubor na dvě třídy s co nejmenším rozptylem). Třetí možností je adaptivní prahování - jde o dynamické lokální prahování, při kterém je obraz rozdělen na několik oblastí a v nich je následně spočten práh, tj. pro různé pixely obrazu nemusí být hodnota prahu stejná.

V případě manuálního nastavení absolutní hodnoty prahu, by bylo třeba tuto hodnotu určovat pro každý datový soubor zvlášť (různé systémy mohou používat různý rozsah hodnot intenzity a mimo to mohou mít na odrazivost materiálů vliv vnější podmínky – např. vlhkost). Knihovna funkcí nástroje MatLab sice nabízí automatické prahování metodou Otsu a adaptivní prahování, ale ani jedna z metod se neumí dobře vypořádat s případem, kdy je jedna klasifikovaná třída výrazně menší než ta zbývající (jak je tomu v řešené situaci). Rozložení hodnot intenzity ukazuje histogram na obr. 37 a detail na obr. 38 (intenzita převedena na hodnoty 0-1 lineární úpravou histogramu, 0 a 1 odpovídají extrémním hodnotám intenzity v daném souboru). Na základě pozorování testovacích souborů bylo zjištěno, že body odpovídající reflexním dopravním značkám mají hodnoty intenzity po úpravě histogramu vyšší než 0,5, ostatní tabule vyšší jak 0,1). Vhodným řešením se ukázalo být nastavení fixní hodnoty prahu až po úpravě histogramu. Tato hodnota je relativní ve vztahu k absolutním hodnotám intenzity (práh se nachází v polovině/desetině vzdálenosti mezi minimální a maximální hodnotou intenzity v souboru).



Obr. 37 Histogram intenzitního obrazu; 24. km



Obr. 38 Histogram intenzitního obrazu - detail; 24. km

Původním záměrem bylo mít jeden práh pro reflexní značky i ostatní tabule (tj. použít práh nižší). Problém nastává u vegetace, která má v některých případech odrazivost obdobnou odrazivosti tabulí

(obr. 40). Souvislejší části vegetace je možné odfiltrvat na základě velikosti, ale menší části vegetace zůstanou. Nabízí se možnost filtrace objektů ještě na základě vzdálenosti od vozovky, tvaru a orientace. Podmínka minimální vzdálenosti může část vegetace odstranit, ale ne veškerou. Snaha filtrovat na základě podmínky přibližné orientace značky k ose vozovky selhává v případě sjezdů (funkce *regionprops* totiž umožňuje určení orientace osy vozovky pro celý objekt *roadway*, taková orientace pak neodpovídá realitě na odbočující části komunikace). S touto podmínkou mají problém i malé značky představované jen několika pixely – vypočtená orientace objektu v obraze neodpovídá skutečné orientaci značky.

Výsledný algoritmus pracuje se dvěma prahy a vyhledává svislé dopravní značky a ostatní tabule zvlášť.

Pro klasifikaci značek je intenzitní obraz po úpravě histogramu převeden na binární prahováním s prahem 0,5 a odstraněny jsou pixely spadající do tříd *vehicles*, *crash barriers* a *gates* (obr. 39). Na příkladu dat z 62. km je zřejmé, že v obraze může zůstat šum ve formě menšího "stínu" bodů za značkou. Tento šum má výrazně menší hustotu bodů než značky a jeho odstranění se provede na základě požadavku minimální hustoty bodů na pixel ve výšce značek (nastavení 10 bodů/pixel ověřeno na testovacích souborech).



Obr. 39 *Signs* - primární obraz (vlevo), po aplikaci podmínky minimálního počtu bodů na pixel (vpravo); 62. km

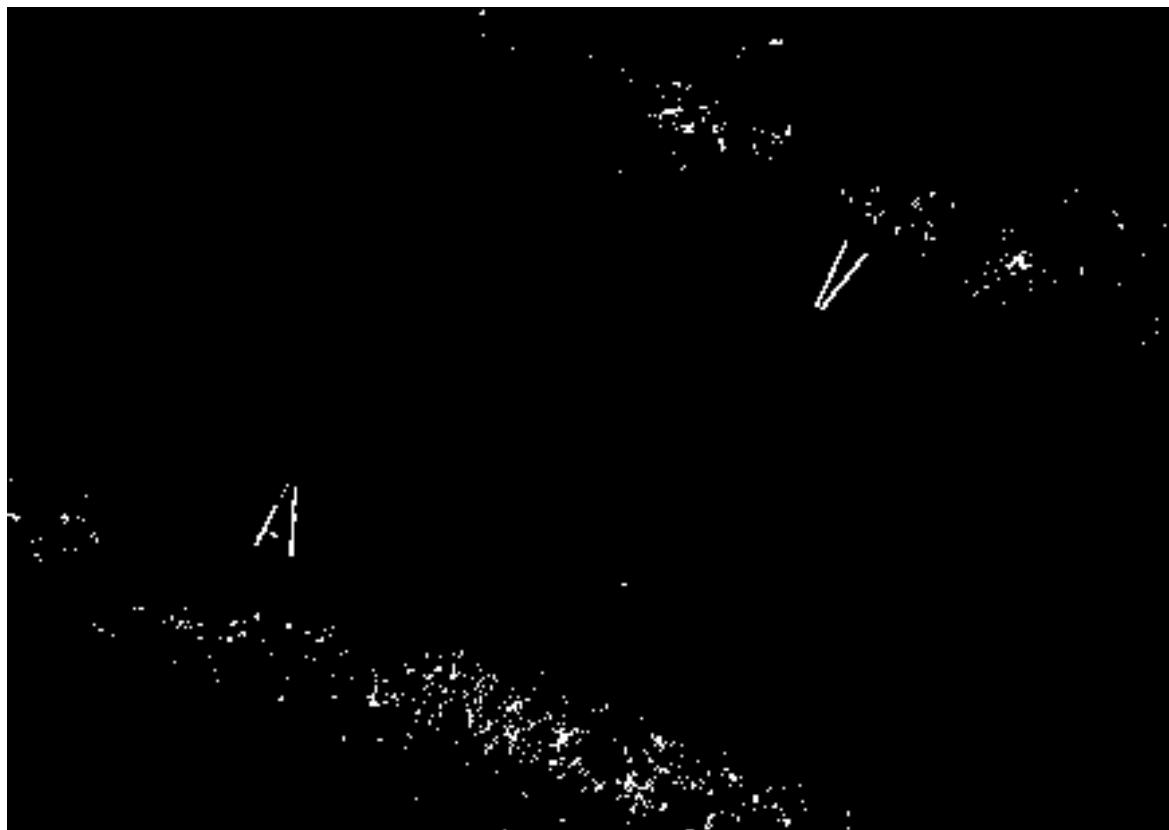
V dalším kroku jsou v obraze ponechány pouze objekty, které se alespoň svou částí nachází do vzdálenosti 7,00m od *roadway* (dle *TP65 (2013)* je maximální vzdálenost od vnějšího okraje zpevněné části krajnice k bližšímu okraji značky 2,00m, hodnota stanovena s větší rezervou kvůli závlivům u SOS hlásek, vychází z pozorování testovacích souborů).

Následuje filtrace na základě velikosti. V obraze zůstávají i objekty s minimálními rozměry vybavené reflexními prvky – např. SOS hlásky. Filtrace se provádí funkcí *bwpropfilt* s volbou minimální hodnoty velké poloosy elipsy rozptylu - *MajorAxisLength* 0,60m (typický rozměr SOS hlásek nepřesahuje 0,50m, minimální rozměry značky na dálnici udává *TP65 (2013)* pro čtvercovou značku - 750mm).

V posledním kroku je kontrolován tvar značky – na základě excentricity elipsy, jejíž rozptyl je totožný s rozptylem pixelů objektu (funkce *bwpropfilt*, parametr *Eccentricity*). Minimální excentricita je stanovena na 0,745, což odpovídá případu, kdy rozptyl ve směru malé poloosy dosahuje maximálně 2/3 rozptylu ve směru velké poloosy. Hodnota byla stanovena tak, aby ji splňovaly všechny malé značky v testovacích souborech (u malých značek je poměr šířky k délce v obraze největší).

Pro klasifikaci tabulí bez reflexní povrchové úpravy (zejména reklamních billboardů) je intenzitní obraz po úpravě histogramu převeden na binární prahováním s prahem 0,1 (obr. 40), dále jsou odstraněny pixely spadající do tříd *vehicles*, *crash barriers* a *gates*. V obraze mimo dopravních značek a tabulí zůstává i část vegetace. Tato vegetace je stejně jako šum u dopravních značek částečně odstraněna pomocí kritéria minimální hustoty bodů ve výšce značek na pixel. Z větší části je pak odstraněna obdobně jako vegetace při klasifikaci *crash barriers*. U tabulí se často stává, že jsou v obraze spojené s blízkou vegetací. Je třeba tedy odstranit shluky bodů o vyšší intenzitě uvnitř bloků vegetace, zároveň ale zamezit odstranění s vegetací spojených tabulí. Vegetace v tomto případě odpovídá pixelům obsahujícím body s výškou nad minimální hranici (1,00m, nižší vegetace na klasifikaci tabulí nemá vliv), mimo pixelů zařazených do třídy *crash barriers* a obrazu představujícího tabule (vzniklý prahováním intenzitního obrazu). V obraze jsou ponechány jen objekty s maximálním rozptylem větším než 5m (větší plochy vegetace). Obraz vegetace je následně upraven funkcí *imfill* operací *holes* (zaplní otvory v objektech – uvnitř bloků vegetace) a funkcí *bwmorph* operací *spur* (odstraní úzké výběžky objektů – např. pozůstatky částí tabulí). V dalším kroku jsou všechny okrajové pixely objektů (mající alespoň jednoho souseda hodnoty 0) odstraněny z objektu, čímž se zajistí přerušení možného spojení objektů vegetace a tabulí (obr. 41). Z obrazu zobrazujícího tabule (obr. 40) jsou poté odstraněny všechny objekty, které mají spojení s objekty z obrazu vegetace.

Zbytky vegetace a dalších objektů jsou od tabulí odděleny na základě minimální a maximální velikosti (ve směru největšího rozptylu bodů – funkce *bwpropfilt*, operace *MajorAxisLength*) stanovené na 3,00-30,00m (menší a větší než rozměr standardních reklamních tabulí umístovaných v blízkosti pozemních komunikací - billboardů, bigboardů, megaboardů (*QUO-reklama.cz, 2017*)). Posledním kritériem filtru je tvar tabule – excentricita elipsy rozptylu (totožný postup a nastavení jako u dopravních značek). Výsledný obraz tabulí (obr. 42) je spojen s obrazem značek do společné třídy *signs*.



Obr. 40 Billboardy – primární obraz před filtrací; 24. km



Obr. 41 Odstranění vegetace; 24. km



Obr. 42 Billboardy - po aplikaci filtrů; 24. km

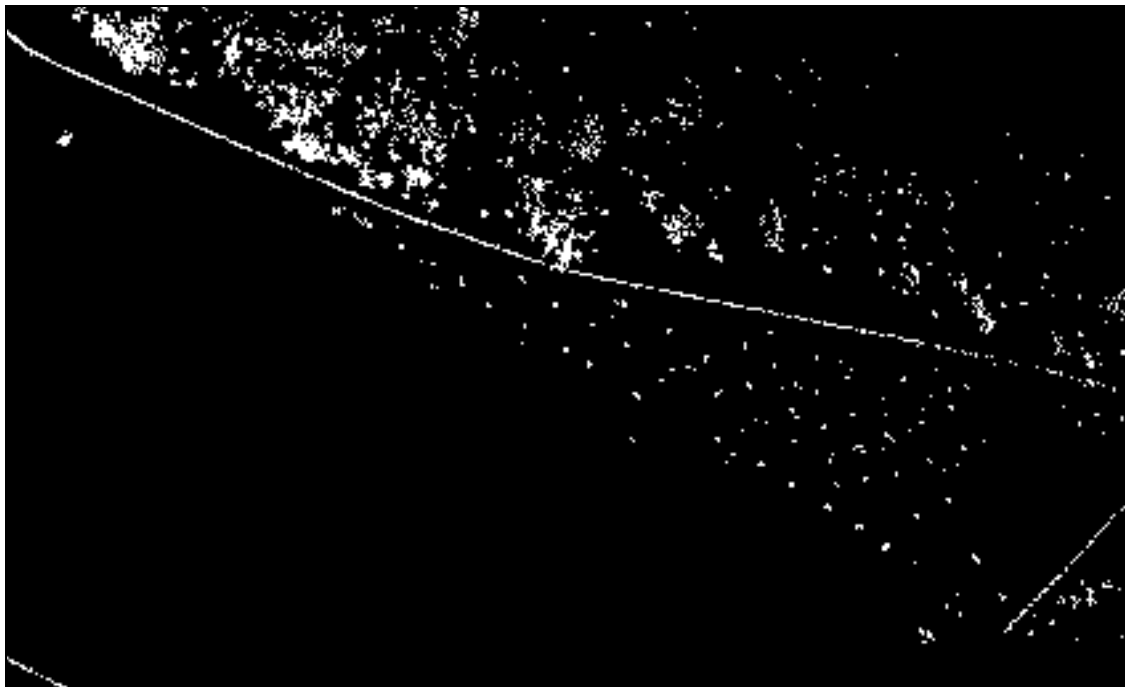
4.3.6 Walls – zdi

Třída *walls* zahrnuje stěny (např. součásti protihlukových opatření) a ploty. Kompletní budovy, např. čerpací stanice, nebyly v rámci souboru dostatečně zdokumentovány, proto nejsou předmětem klasifikace. Objekty třídy *walls* se po průmětu do horizontální roviny jeví jako linie. U plotů (podél dálnic zpravidla drátěných) může být problém s malým množstvím registrovaných bodů; při rozlišení 0,20m tvoří v obraze nesouvislé linie.

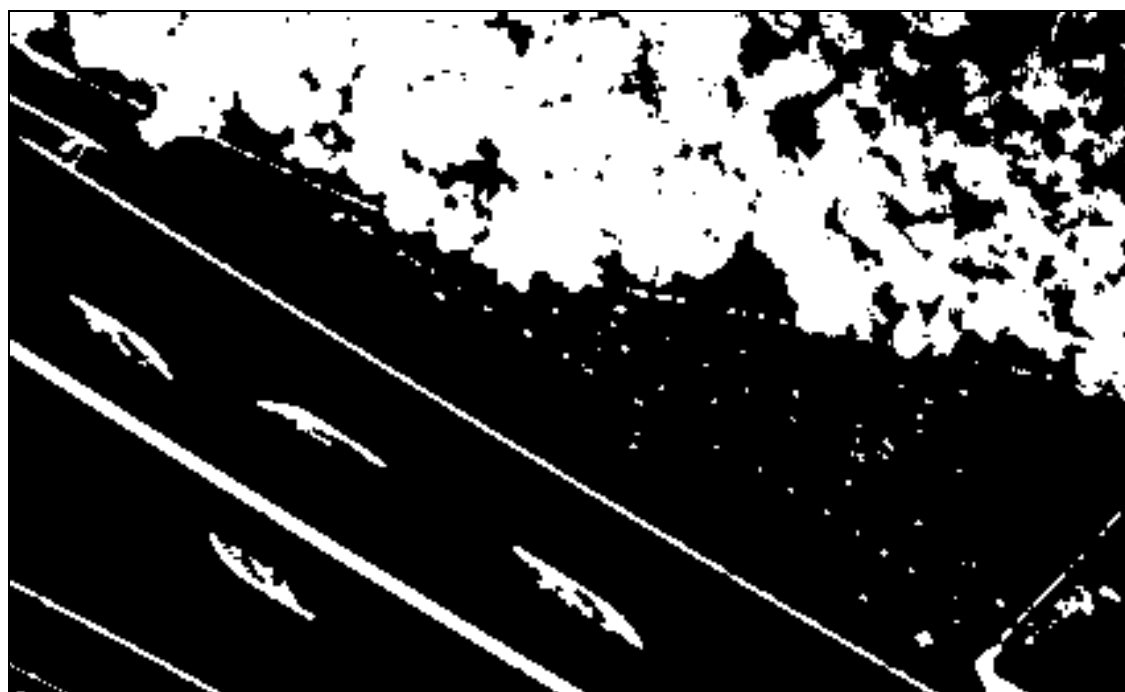
Algoritmus pracuje s body ve výšce 0,80-2,00m nad terénem. Prahy jsou zvoleny tak, aby byl v co největší míře omezen vliv nízké vegetace bezprostředně navazující na ploty (spodní hranice) a vliv vegetace přesahující přes plot (horní hranice), výška plotů v testovacích souborech se pohybovala v rozmezí 1,5-2,0m. Z obrazu bodů v daném rozmezí výšek jsou odstraněny pixely tříd *roadway*, *signs*, *crash barriers* a *gates* (obr. 43). V takto vytvořeném obraze zbývá mimo objektů třídy *walls* i množství vegetace, sloupů a jinak nezařazených objektů.

V dalších krocích je třeba odstranit menší i větší vegetaci, k tomu účelu se vytvoří přibližný obraz vegetace. Vybrány jsou pixely obsahující body s výškou vyšší než 0,5m nad terénem. Součástí takového obrazu jsou samozřejmě i ploty/stěny, které je třeba odstranit. Na tento obraz se aplikuje mediánový filtr – funkce *medfilt2* (částečně vyplní otvory v souvislé vegetaci a odfiltruje linie (obr. 44)), poté je dvakrát aplikována funkce *ordfilt2* – nahradí hodnotu bodu n-tou hodnotou z množiny hodnot pixelů v okolí 3x3. Při první filtraci se hodnota pixelu nahrazuje 1. hodnotou z okolí, tj. pokud je v okolí pixelu pixel s hodnotou 0, bude hodnota pixelu 0 (odstraní zbytky stěn a drobnou vegetaci), při druhé filtraci se

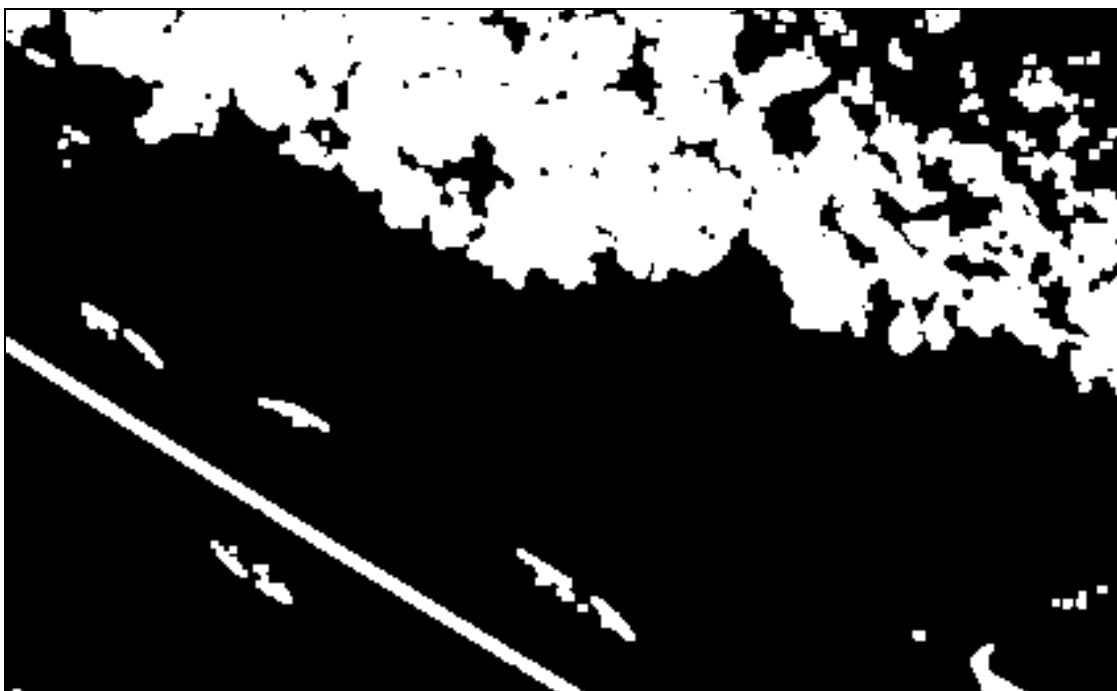
použije 9. hodnota – tj. pokud je v okolí pixelu pixel s hodnotou 1 bude hodnota pixelu 1 (větší objekty vegetace, které se při první aplikaci filtru zmenšily, se vrací k původní velikosti, s výjimkou tenkých výběžků, které mohly představovat části vegetace, nebo plotu). Výsledný obraz (obr. 45) se odečte od primárního obrazu *walls* (obr. 43).



Obr. 43 Primární obraz třídy *walls* – ploty a vegetace; 72. km

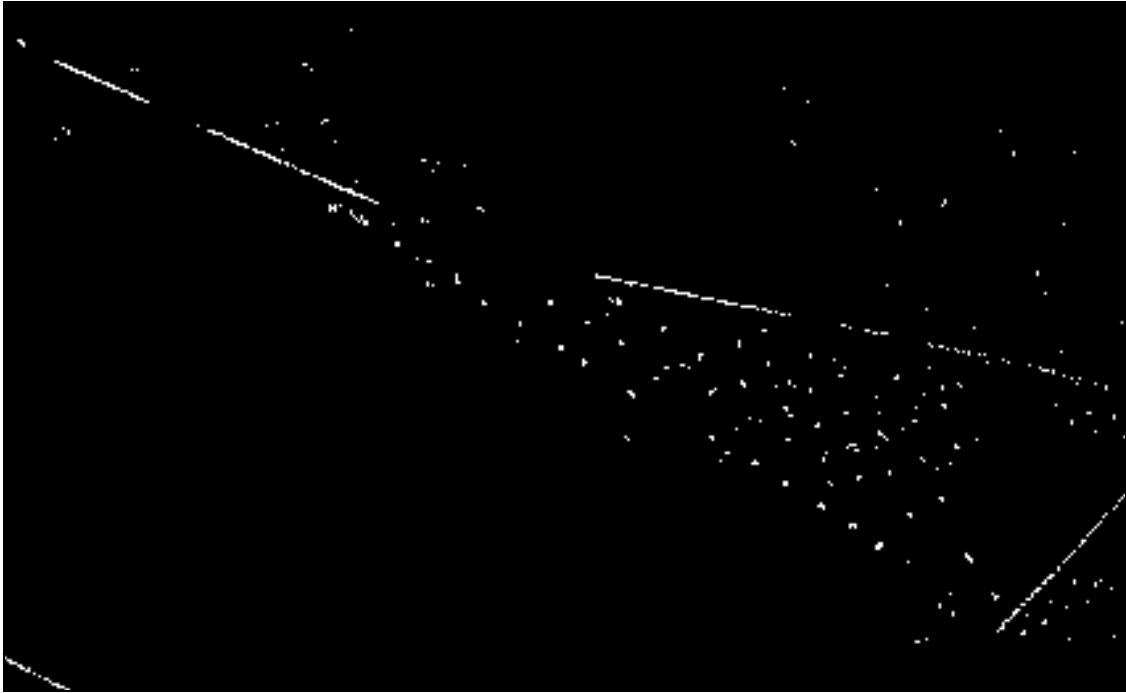


Obr. 44 Body s výškou 0,50m nad terénem a výš, po aplikaci mediánového filtru; 72. km

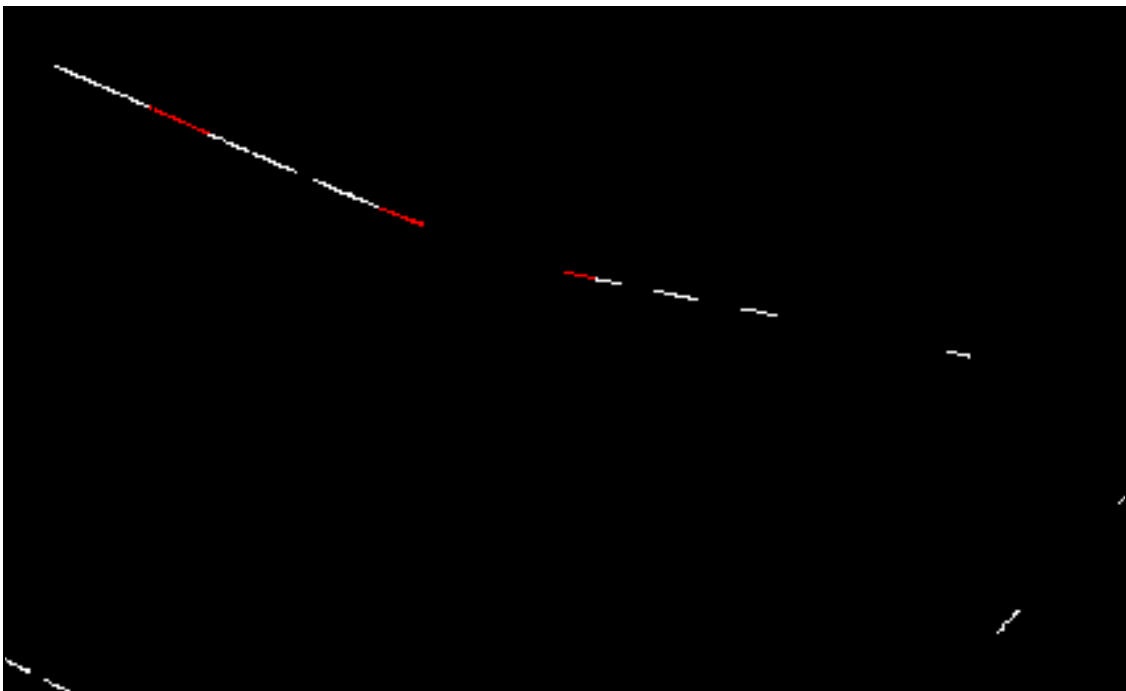


Obr. 45 Vegetace k odstranění; 72. km

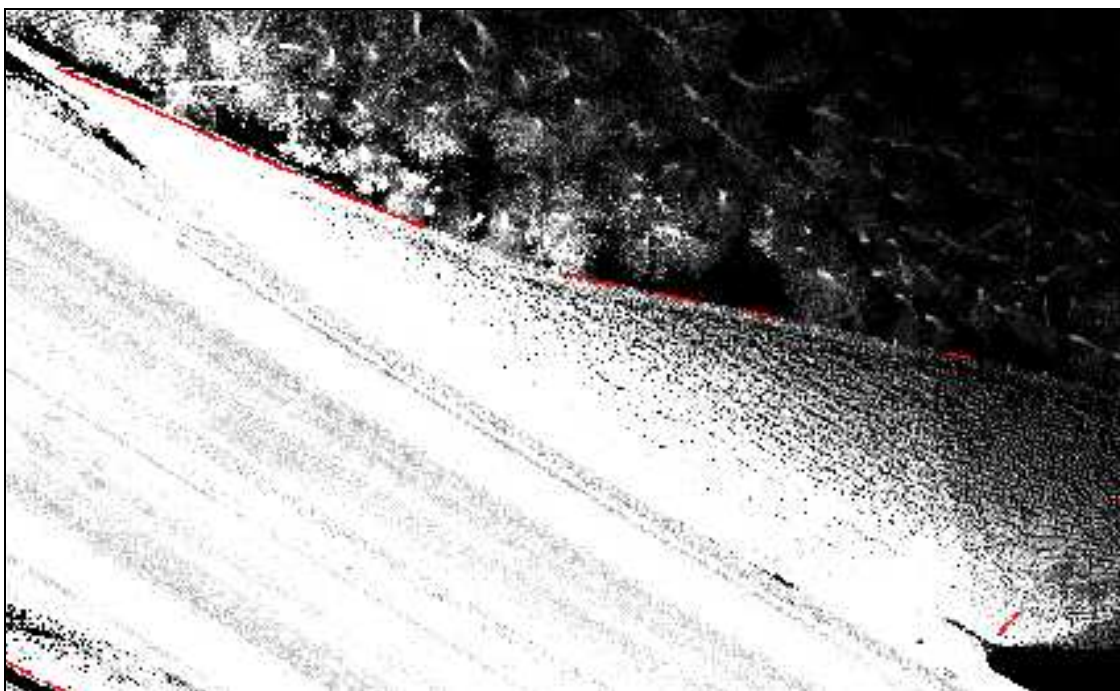
Tímto postupem je odstraněna větší vegetace, ale v obraze *walls* zůstanou mimo stěn/plotů ještě další drobné objekty (obr. 46). Na vzniklý obraz se aplikuje znovu dvakrát filtr *ordfilt2*, tentokrát s nastavením v opačném pořadí (první filtrace s nastavením nové hodnoty na 9., druhá s nastavením na 1.), tento postup částečně kompenzuje nízkou hustotu bodů u drátěných plotů. Dojde ke spojení blízkých objektů – částí plotů, do delších linií. Menší objekty (především drobná vegetace) se z obrazu odstraní na základě parametrů elipsy rozptylu hodnot – minimální délky velké poloosy (stanoveno na 2,0m) – funkce *bwproffilt*, parametr *MajorAxisLength*. V případě, že přes plot přesahuje vegetace, může být dotčená část plotu odstraněna spolu s vegetací v předchozím kroku a je třeba ji k obrazu plotů připojit. Alespoň částečně je to možné následujícím postupem: z primárního obrazu *walls* se vyberou objekty, které mají společný průnik s objekty prošlými filtrací. Z těchto objektů se vyberou pouze ty, u nichž malá poloosa elipsy rozptylu hodnot nepřesahuje 1,00m (odfiltruje případnou vegetaci). Nalezené části jsou poté připojeny k obrazu (obr. 47). V případě plotů s přesahující nebo do plotu vrostlou vegetací je mračno často natolik řídké, že uvedený postup neumožní klasifikaci kompletního plotu. Jako možné řešení se nabízí snížení rozlišení obrazu, které by ovšem vedlo k těsnějšímu propojení plotu a blízké vegetace a zhoršilo by možnosti jejich odlišení.



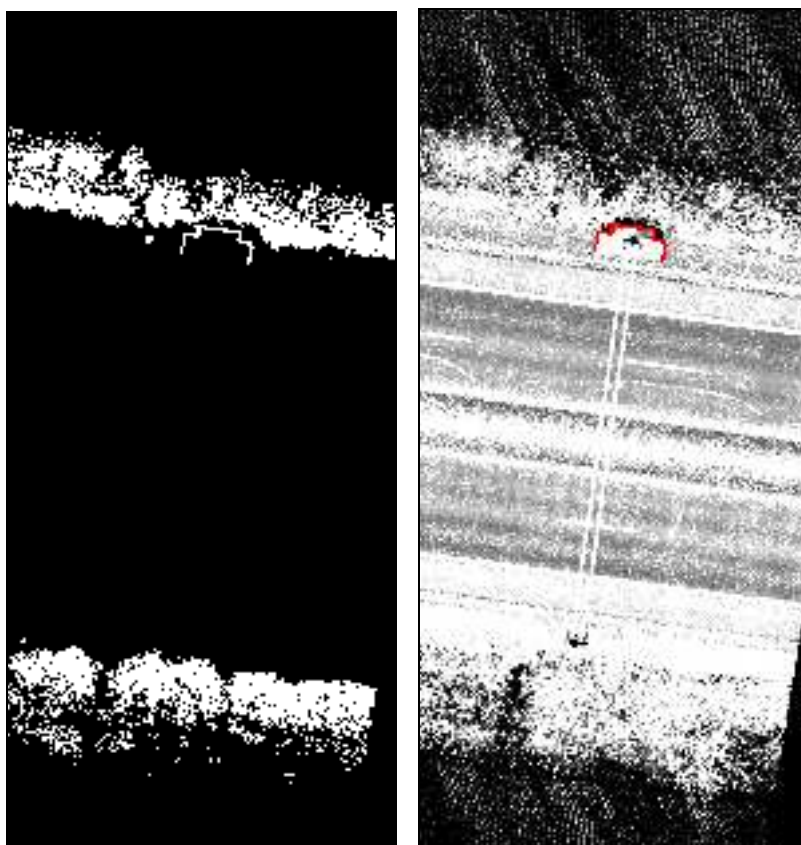
Obr. 46 Obráz plotů po odstranění větší vegetace; 72. km



Obr. 47 Třída *walls*, z toho červeně označeny části schované pod vegetací a přidané dodatečně; 72. km



Obr. 48 Třída *walls* (červeně), na pozadí intenzitní obraz; 72. km



Obr. 49 Zábradlí u mýtné brány, vlevo: primární obraz *walls*, vpravo: výsledná třída *walls* (červeně) na pozadí intenzitního obrazu; 31. km

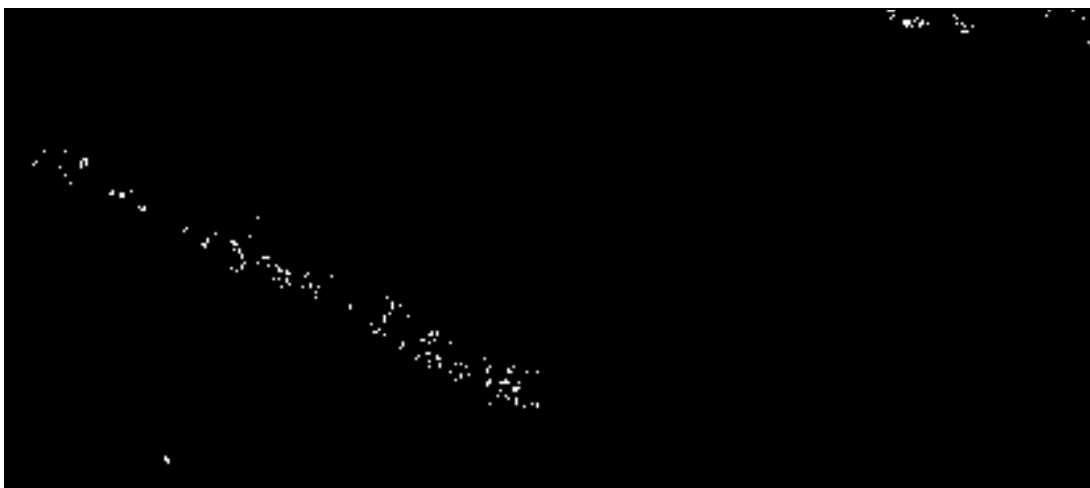
Funkce mimo zdí a plotů klasifikuje i objekty s podobnými charakteristikami jako má plot, např. zábradlí kolem betonového základu mýtné brány (obr. 49).

4.3.7 Poles – tyčové objekty

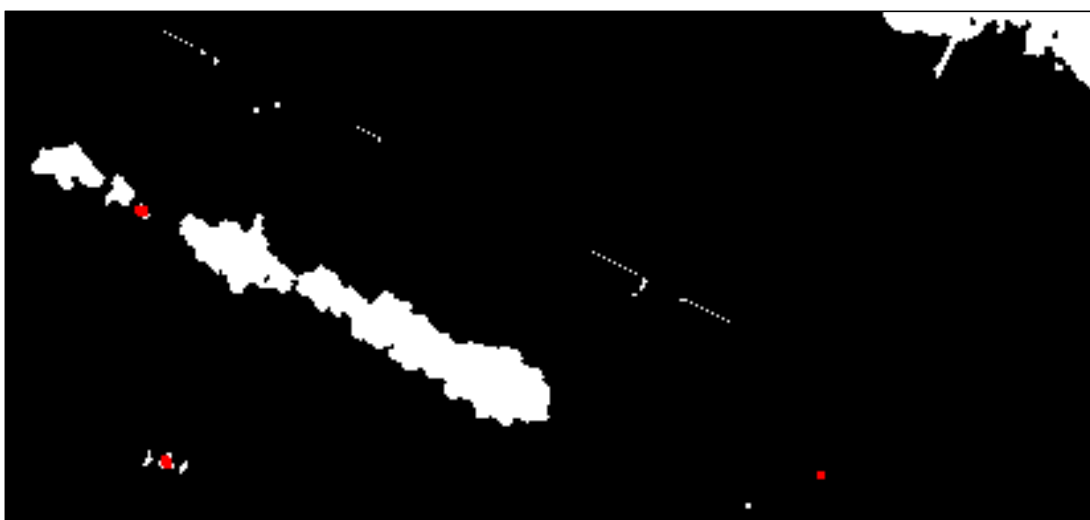
Třída *poles* obsahuje samostatně stojící tyčové objekty (nepatří sem nosiče svislých dopravních značek ani kmeny stromů, či jiné tyčové objekty, které jsou součástí dříve definovaných tříd). Z hlediska velikosti je možné tyto objekty rozdělit na dvě skupiny. Do první spadají objekty většího rozměru, zejména sloupy elektrického vedení. Sem by patřily i sloupy veřejného osvětlení, které se ovšem v testovaném datovém souboru (zobrazujícím dálnici mimo město) nevyskytují. Do druhé spadají objekty menšího rozměru, které jsou součástí příslušenství komunikace - směrové sloupky a SOS hlásky.

Sloupy elektrického vedení mohou být v některých případech zaměnitelné s vegetací – ramena nesoucí dráty elektrického vedení a části přilehlého vedení se v půdoryse jeví jako koruna stromu. Je tedy třeba nalézt tyčové objekty a odlišit sloupy elektrického vedení od kmenů stromů. Algoritmus vybírá pixely s body ve výšce 3,00-5,00m (spodní hranice nad možnou vegetací obklopující sloup, horní pod úrovní nesených vodičů – minimální vzdálenost vodičů nízkého napětí od země je u volně přístupné zemědělské plochy stanovena na 5,00m). Z těchto pixelů jsou vybrány pouze ty, obsahující v daném výškovém rozmezí více jak 10 bodů – takto se odstraní množství řidší vegetace a dalších objektů. Zbývající objekty jsou filtrovány na základě maximální hodnoty velké poloosy elipsy rozptylu (1,00m), tím se odstraní rozměrnější objekty např. billboardy. V obraze zůstávají pouze sloupy, kmeny stromů a jiné části husté vegetace (obr. 50). V dalším kroku je třeba odstranit objekty související s vegetací. Výběr vegetace v tomto případě probíhá následovně: algoritmus vybere všechny pixely s body výše než 3m (totožná hranice jako u výběru sloupů, nízká vegetace je mimo oblast zájmu). Ramena sloupy elektrického vedení a přilehlé vedení se mohou svými rozměry blížit rozměrům koruny, jsou ovšem zpravidla složeny z několika liniových objektů a tvoří v obraze tak souvislý objekt jako koruna stromu. Na výsledný obraz se proto aplikuje mediánový filtr (*medfilt2*), díky němuž se objekt ramen sloupu rozdělí na menší objekty, ale kompaktnější koruny výrazně neovlivní (obr. 51). Z filtrovaného obrazu se následně vyberou objekty s délkou malé poloosy elipsy rozptylu do 2,00m (odstraní se tím vegetace s průměrem koruny 2,00m a víc a ponechají se části ramen sloupů). Z dosavadního obrazu sloupů se vyberou pouze objekty, které mají společný prvek s objekty zobrazujícími ramena sloupů).

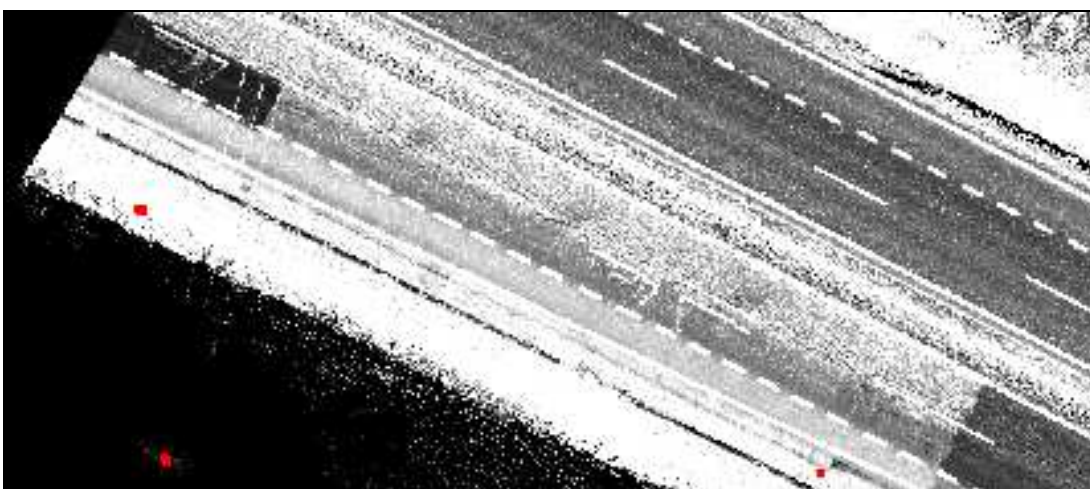
Uvedený postup klasifikuje mimo sloupů elektrického vedení i stromy s podrobně naskenovaným kmenem a malou korunou, tyto objekty jsou si značně podobné a nepodařilo se je jednoznačně odlišit (obr. 52). Další snížení hranice maximálního rozměru objektu představujícího ramena sloupu (pod 2,00m) vedlo k vyřazení části sloupů. Inspiraci při řešení tohoto problému poskytl *Pytel (2006)*. Z důvodu minimalizace množství takto chybně klasifikovaných stromů, nepřistupuje algoritmus k vyhledávání sloupů, pokud nejsou ve zpracovávaném souboru nalezeny dráty nadzemního vedení.



Obr. 50 Body potenciálně příslušející k třídě *poles*; 18. km



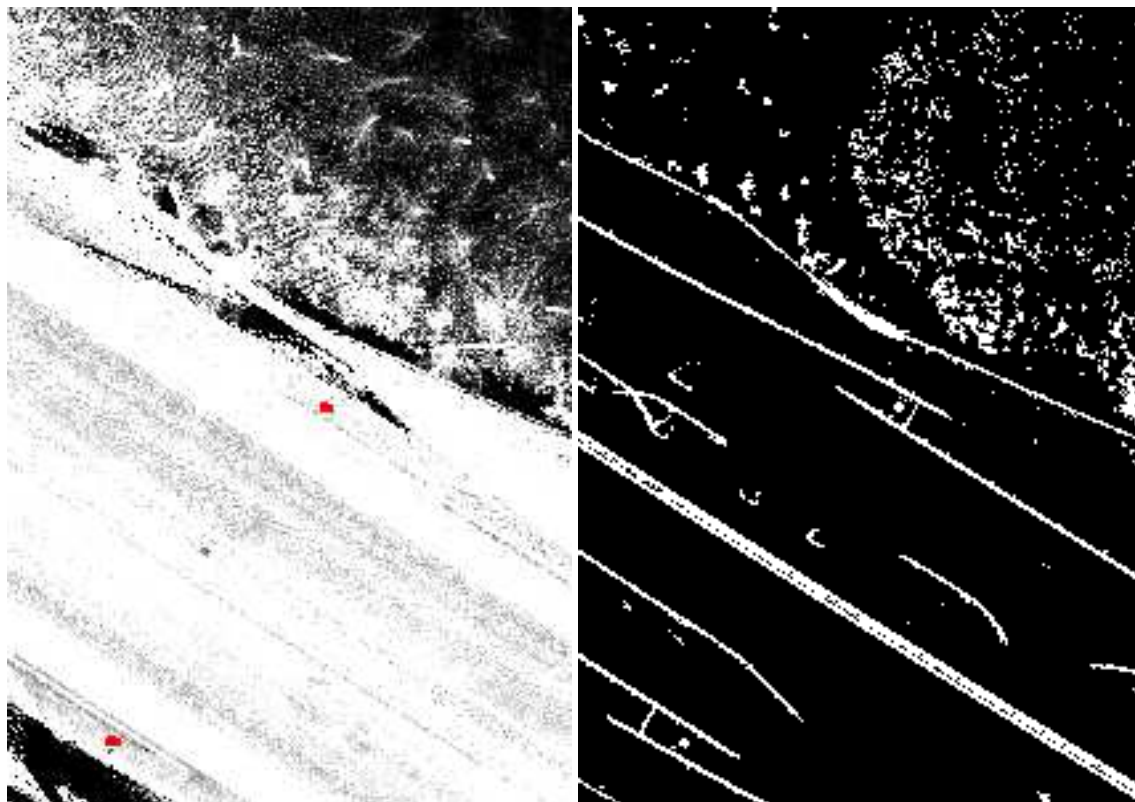
Obr. 51 Třída *poles* (červeně), na pozadí obraz bodů 3m nad terénem a výš po mediánovém filtru; 18. km



Obr. 52 Třída *poles* (červeně), na pozadí obraz intenzity; vlevo nahoře - chybně klasifikovaný kmen stromu, vlevo dole - sloup elektrického vedení, vpravo – směrový sloupek; 18. km

Při klasifikaci směrových sloupků a SOS hlásek se vychází z jejich rozměrů, přičemž limitujícími faktory jsou: minimální výška směrových sloupků a maximální rozměr SOS hlásky v půdorysu. Směrové sloupky u dálnic se musí řidiči jevit jako objekt o výšce 1050 ± 50 mm (TP 58, 2008). Půdorysné rozměry SOS hlásek nepřekračují hodnoty $0,5 \times 0,5$ m (Spel, 2017), což odpovídá úhlopříčce 0,7 m.

Směrové sloupky a SOS hlásky se po průmětu do horizontální roviny jeví jako malé objekty a měly by se takto projevovat i v horizontálních řezech vedených v různé výšce (až do výšky objektu). Algoritmus využívá tři vrstvy voxelové matice, jejíž tvorba je popsána v oddílu 4.2. Každá vrstva obsahuje informace o bodech nacházejících se v daném výškovém rozmezí (definované polohou vrstvy a její tloušťkou/výškovým rozlišením - 0,1 m). Vybrány jsou vrstvy odpovídající svou polohou výškám 0,40 m, 0,60 m a 0,80 m nad terénem (výš než nejnižší vegetace a níž než horní hrana směrového sloupku). Jako sloupky jsou vybrány všechny pixely obsahující alespoň dva body v každé z vybraných vrstev. Tuto podmínku mohou splnit i jiné objekty, např. hustá vegetace. Z obrazu je proto třeba odstranit pixely, které mohou být součástí větších objektů (např. právě vegetace). Z toho důvodu je vytvořen obraz pixelů obsahujících jakýkoli bod v rozmezí výšek stanovených pro vyhledávání sloupků (0,40-0,80 m). Z výsledného obrazu jsou vybrány všechny pixely tvořící objekty, jejichž velká poloosa elipsy rozptylu přesahuje 1,00 m (je větší než maximální úhlopříčka hledaných SOS hlásek). Z obrazu sloupků jsou odstraněny všechny objekty vybrané v předchozím kroku, dále objekty tříd *vehicles*, *gates*, *crash barriers* a *signs*. V takto vytvořeném obraze stále zůstává část vegetace – např. malé stromky a jejich opěry. Z důvodu snížení množství chybně klasifikované vegetace je omezeno vyhledávání sloupků na oblast okraje vozovky – třídy *roadway* – v rozsahu 2,00 m do vozovky a 3,00 m mimo ni (obr. 53).



Obr. 53 Vlevo - SOS hlásky (červeně), na pozadí intenzitního obrazu, vpravo - obraz bodů ve výšce 0,4-0,8 m nad terénem; 72. km

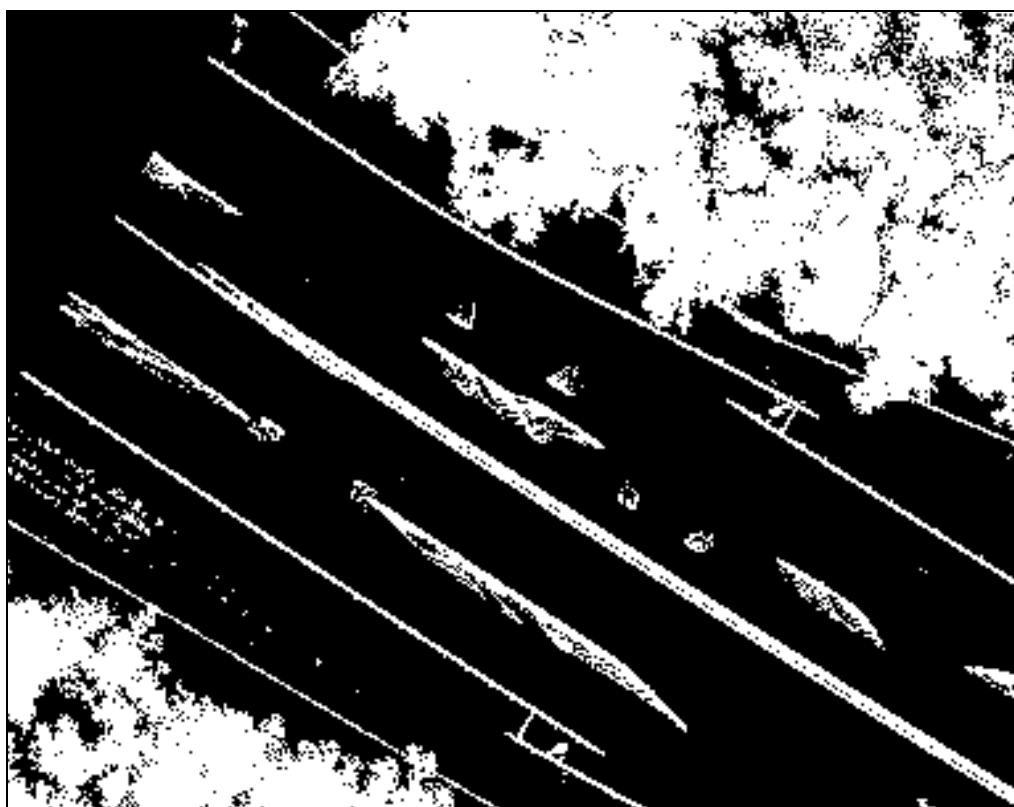
4.3.8 Vegetation - vegetace

Vegetace je chápána jako rostlinná pokrývka daného území. Pro účely klasifikace mračen bodů je často dělena na nízkou (do 1m), střední (1-5 m) a vysokou (vyšší než 5m), např. dle *Miura, Jones (2010)*. Dělení vegetace dle výšky by mohlo mít v oblasti dokumentace pozemních komunikací a jejich okolí význam z hlediska bezpečnosti – např. při sledování vztahu výška vegetace – vzdálenost od vozovky, ale pro potřeby základní klasifikace význam nemá.

Drobná vegetace (např. vyšší travní porost, menší keře, apod.), může být z hlediska klasifikace těžko oddělitelná od dalších částí mračna (např. neklasifikovaných částí plotů). V zásadě existují dvě možnosti jak se s tímto problémem vypořádat: klasifikovat vegetaci včetně drobné i s rizikem vysokého množství chyb typu II (ve třídě *vegetation* budou klasifikovány objekty, které do třídy nepatří), nebo specifikovat minimální parametry vegetace, nad které bude vegetace klasifikována. Při tvorbě algoritmu se vychází z druhé možnosti, tedy omezení klasifikace na větší vegetaci, která má význam pro bezpečnost provozu na pozemních komunikacích.

Pro účely klasifikace jsou stanoveny následující parametry vegetace:

- minimální výška vegetace 1,00m
- ve výšce nad 0,50m pokrývá vegetace souvisle plochu o průměru minimálně 1,00m



Obr. 54 Body s výškou 0,5m nad terénem a výš; 72. km

Algoritmus vytvoří obraz bodů s výškou 0,50m nad terénem (obr. 54) a výš a 1,00m nad terénem a výš. Na první obraz (body s výškou 0,50m nad terénem) je aplikován mediánový filtr (*medfilt2*), což zajistí oddělení liniových prvků od vegetace a odstranění izolovaných bodů/skupinek bodů. Z tohoto

obrazu jsou vybrány objekty, které mají společný prvek s obrazem bodů s výškou nad 1,00m nad terénem (tj. objekty přesahující výšku 1,00m). Z výsledného obrazu jsou odstraněny všechny dříve klasifikované třídy. Ze zbývajících objektů jsou ponechány jen ty, jejichž malá poloosa elipsy rozptylu přesahuje 1,00m. Nakonec jsou k objektům přiřazeny i nejbližší sousední pixely (funkce *ordfilt*), což alespoň z části eliminuje vliv aplikace mediánového filtru na počátku procesu, tj. přiřadí k vegetaci i její odfiltrované okraje (obr. 55).



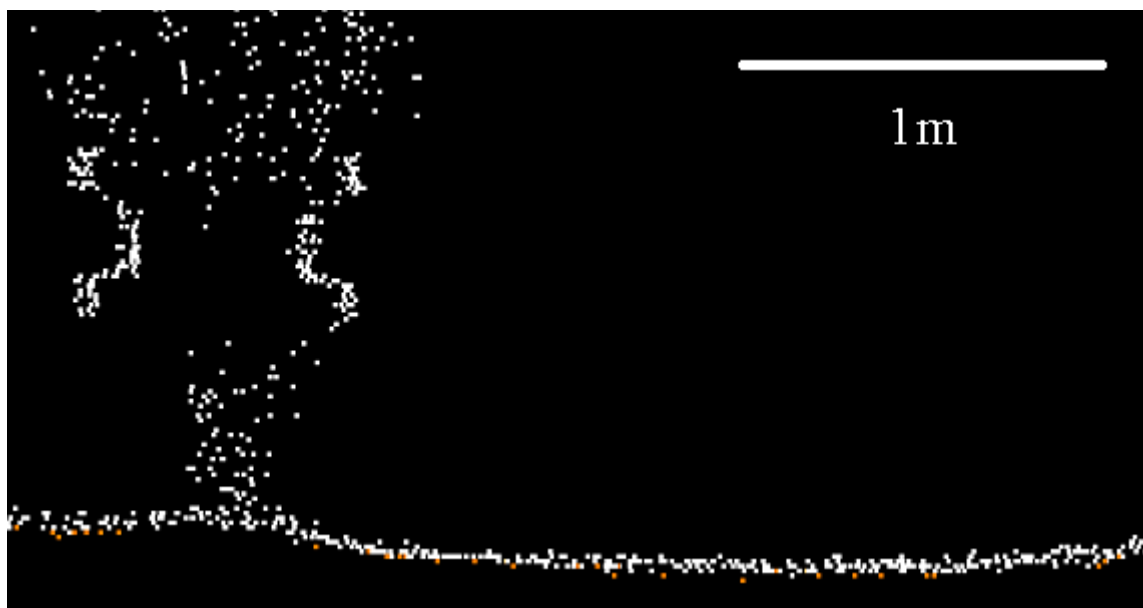
Obr. 55 Třída *vegetation*; 72. km

4.4 Přenos výsledků klasifikace z rastru do mračna bodů

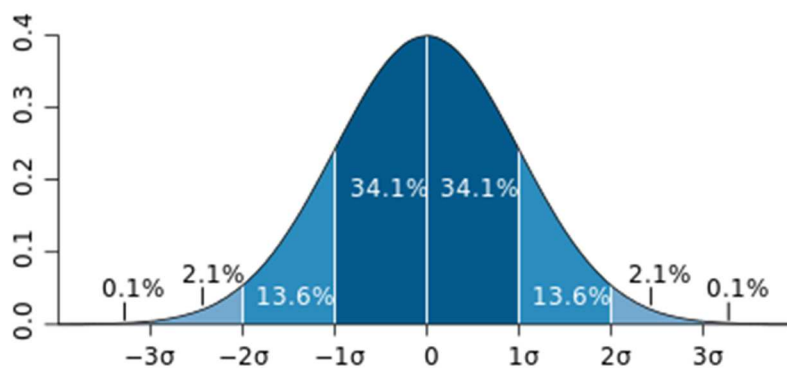
Výsledkem předchozích kroků je množství rastrových obrazů (ve formátu TIFF), které představují jednotlivé třídy, nebo jejich části (v případě třídy *gates* jsou okraje mostů – *gates end* a sloupy – *gates column* uloženy zvlášť) a seznam souřadnic bodů v systému voxelové matice. Tato data společně s původním seznamem souřadnic (formát - *class x y Z_{overground} intensity z*, popsán v oddíle 4.2.1) vstupují do procesu přenosu výsledků klasifikace rastrových dat na mračno bodů. Proces je realizován prostřednictvím kódu – *image2pcloud*.

Zvolená metoda klasifikace s využitím převodu mračen bodů do rastrových dat přináší zásadní komplikaci v tom, že jeden pixel může obsahovat body více tříd. V horizontální rovině je výskyt tohoto jevu alespoň z části eliminován volbou velikosti pixelu. V různých výškách tomu ale zabránit nelze. Jeden pixel může zároveň obsahovat např. body tříd *roadway*, *vehicles* a *gates*, nebo *ground*, *wall* a *vegetation*. Částečně to lze řešit u tříd *ground*, *roadway* a *gates*, u kterých je možné jasně definovat výškové hranice.

Jak bylo uvedeno již v konceptu (oddíl 4.1), metoda pracuje na vstupu s mračnem bodů s klasifikovanou třídou *ground*. Vzhledem k široké nabídce nástrojů pro automatickou klasifikaci terénních bodů se představovaná metoda touto problematikou nezabývá a využívá existujícího řešení. Jak je zřejmé z obr. 56, automatická klasifikace provedená v softwaru TerraScan neklasifikuje všechny body představující terén. Cílem této klasifikace je totiž výběr bodů pro tvorbu DMT a proces klasifikace se po dosažení určité hustoty klasifikovaného mračna zastaví. Zbývající body odpovídající terénu je možné klasifikovat na základě minimální vzdálenosti nad terénem. Z důvodu zatížení měření chybami nejsou ani body odpovídající jedné rovině zcela v rovině, hovoří se o „šumu“. Za předpokladu, že je rozložení bodů (šum) náhodnou veličinou, mělo by toto rozložení odpovídat normálnímu rozdělení. Pravděpodobnost, že se hodnota náhodné veličiny bude od střední hodnoty lišit nejvýše o dvě směrodatné odchylky, je cca 95 % (obr. 57). Z toho vyplývá, že na každém konci Gaussovy křivky se nachází cca 2,5% hodnot vzdálených od střední hodnoty o více než dvojnásobek směrodatné odchylky. Bez znalosti střední hodnoty a směrodatné odchylky lze odstranit tyto extrémní hodnoty tak, že se odstraní odpovídající procento krajních hodnot. Ke stanovení maximální vzdálenosti bodů od původně klasifikovaného terénu - hranici odpovídající šumu dat byl proveden pokus. U vybrané části mračna zobrazujícího vozovku (úsek o šířce dvou jízdních pruhů a délce cca 22m, obsahující 93 352 bodů; body zobrazující projíždějící vozidlo byly odstraněny) byla provedena klasifikace bodů na základě různé výšky nad terénem, hraniční výška se měnila po 0,010m, v jednom případě byl interval rozdělen ještě na polovinu. Cílem bylo nalézt výšku nad terénem, do které bude spadat více jak 97,5% bodů (hodnoty zahrnují spodní extrémní hodnoty). Na základě testu byla zvolena hodnota 0,025m (tab. 3).



Obr. 56 Řez mračnem bodů - středový dělicí pás se svodidly, oranžové body - třída *ground*



Obr. 57 Graf normálního (Gausova) rozdělení, Autor: M. W. Toews – Vlastní dílo, based (in concept) on figure by Jeremy Kemp, on 2005-02-09, CC BY 2.5, <https://commons.wikimedia.org/w/index.php?curid=1903871>

Tab. 3 Rozdělení bodů na základě vzdálenosti od třídy *ground*

Kategorie	počet bodů	ze všech bodů
<i>ground</i>	34852	37,3%
do 0,010m	76496	81,9%
do 0,020m	90686	97,1%
do 0,025m	91989	98,5%
do 0,030m	92852	99,5%

Body do této vzdálenosti nad terénem jsou tedy považovány za terénní (*ground*), nebo *roadway*, pokud přísluší pixelu klasifikovanému jako *roadway*.

Do třídy *gates* jsou přiřazeny pouze body, které dosahují minimální výšky bodů nad terénem jako body využitě při klasifikaci rastrového obrazu (4,50m), stěny a sloupy mostů jsou proto řešeny zvlášť.

Body ostatních tříd nelze rozdělit na základě žádné konstantní výškové hranice a v případě příslušnosti k pixelu klasifikovaného ve více třídách má přednost třída vyššího významu. Třídy byly uspořádány tak, aby měly přednost ty, které mají při dokumentaci pozemních komunikací větší význam, tj. nejprve objekty související z řízením provozu a bezpečností, poté ostatní objekty trvalého rázu, vegetace a na závěr dočasné objekty, tedy vozidla. Třídy jsou tedy seřazeny následovně: *signs* a *poles* (dopravní značky a drobnější zařízení), dále *crash barriers* a *walls* (větší zařízení) a na konec *vegetation* a *vehicles*.

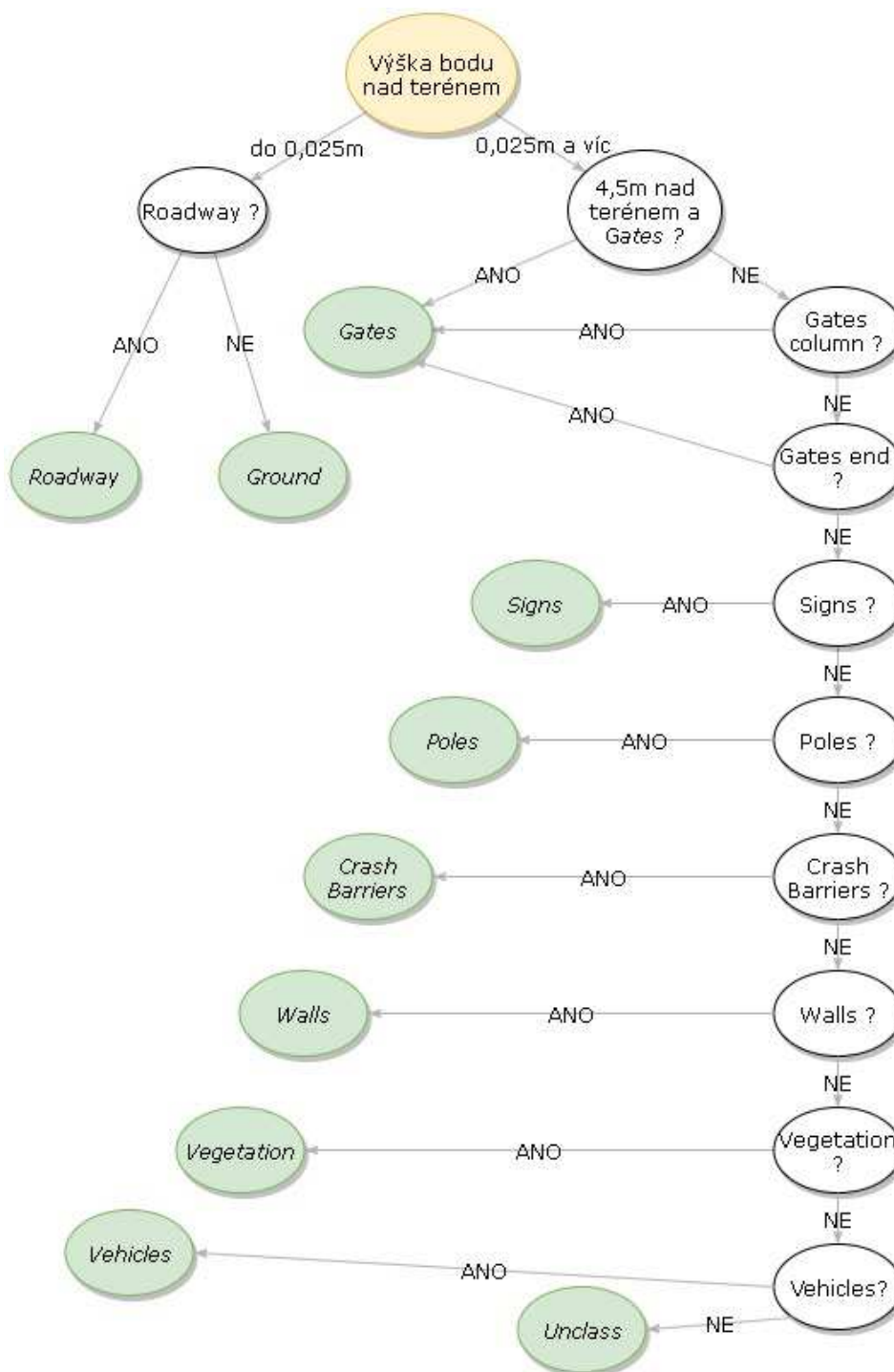
Rozhodování o příslušnosti bodu do té které třídy popisuje rozhodovací strom na obr. 58.

Bodům je přidělen kód třídy, do které jsou přiřazeny:

- 1 *Unclass*
- 2 *Ground*
- 3 *Roadway*
- 4 *Vehicles*
- 5 *Gates*
- 6 *Crash Barriers*

- 7 Poles
- 8 Signes
- 9 Walls
- 10 Vegetation

Klasifikované mračno bodů je uloženo ve formě seznamu souřadnic doplněného o informaci o třídě a intenzitě: *class x y z intensity*. Zdrojový kód přenášející výsledky klasifikace na mračno bodů je uveden v příloze 6.



Obr. 58 Klasifikace bodu - rozhodovací strom

5 Ověření metody

5.1 Postup

Pro hodnocení úspěšnosti algoritmu byla vybrána stejná metoda, jakou použili *Sithole a Vosselman (2004)* při hodnocení úspěšnosti algoritmů pro klasifikaci terénních bodů. Úspěšnost algoritmu se hodnotí na základě množství chyb typu I. (body do dané třídy patří, ale klasifikovány nebyly), chyb typu II. (body do třídy nepatří, ale jsou v ní chybně klasifikovány) a celkového množství chybně klasifikovaných bodů.

Protože množství autorů (např. *Yu a kol. (2015)*, *Rodríguez-Cuenca a kol. (2015)*, *Zhu a kol. (2017)*) vyjadřují úspěšnost algoritmu také dalšími parametry – úplností, přesností a F-mírou, jsou pro jednotlivé třídy vypočteny i tyto:

$$\text{Úplnost} = \frac{TP}{TP + FN} \quad (6)$$

$$\text{Přesnost} = \frac{TP}{TP + FP} \quad (7)$$

$$F - \text{míra} = \frac{2 \times \text{Přesnost} \times \text{Úplnost}}{\text{Přesnost} + \text{Úplnost}} \quad (8)$$

kde *TP* je *true positive* (počet správně klasifikovaných bodů ve třídě), *FN* je *false negative* (počet bodů chybně do třídy nezařazených, tj. počet chyb I. typu) a *FP* je *false positive* (počet bodů do třídy chybně zařazených, tj. počet chyb II. typu).

Klasifikace provedená automaticky (na základě vytvořené metody) se porovná s klasifikací manuální (provedenou vyhodnocovatelem), která je považována za správnou. Toto porovnání se provede na úseku dálnice D11 vybraném tak, aby obsahoval všechny typy objektů, na které se metoda klasifikace zaměřuje.

Data na vybraném úseku na 73. – 75. km o délce 1,8km byla klasifikována za použití dostupných funkcí softwaru TerraScan a prošla kompletní manuální kontrolou.

Porovnání klasifikovaných souborů je realizováno prostřednictvím funkce *ClassQualCheck* (příloha 7):

```
function [Qual] = ClassQualCheck(autoclass,reference)
```

Parametry *autoclass* (automaticky klasifikované) a *reference* (referenční - manuálně klasifikované) reprezentují porovnávaná mračna bodů ve formě textového souboru (formát: *Class x y z Intensity*). Výstupem funkce je tabulka (resp. matice záměn, uložená ve formátu TXT) porovnávací výsledky klasifikace. U každého bodu je zjištěna třída dle manuální a automatické klasifikace a je přiřazen do odpovídající kolonky v tabulce.

5.2 Výsledky

Porovnávaný datový soubor obsahoval celkem 32,6 milionu bodů a pro potřeby klasifikace byl rozdělen do devíti bloků. Manuální klasifikace souboru trvala 19hodin (bez úvodního rozdělení souboru do bloků a automatické klasifikace terénních bodů, která byla totožná pro oba typy klasifikace). Automatická klasifikace souboru trvala 54 minut.

Výsledky porovnání jsou shrnuty v následujících tabulkách. Tab. 4 zobrazuje počty klasifikovaných bodů. Sloupečky znázorňují, do jaké třídy měly být body klasifikovány (výsledky manuální klasifikace), a řádky, do jaké třídy byly klasifikovány (výsledky automatické klasifikace). Na diagonále se nachází počty správně klasifikovaných bodů. Veškeré počty bodů jsou uvedeny v tisících. Pro lepší představu o poměrech správně a chybně klasifikovaných bodů uvádí další tabulka hodnoty v procentech (tab. 5). Celkem bylo klasifikováno 32 645 520 bodů z toho 1 809 089 chybně, což odpovídá 5,5%.

Tab. 4 Porovnání výsledků manuální a automatické klasifikace - počty bodů v tisících

		Manuální klasifikace									
		třída	1	2	3	4	5	6	7	8	9
Automatická klasifikace	1	3343,1	6,4	2,5	1,6	1,7	3,3	0,4	0,2	18,9	107,6
	2	411,0	3993,6	185,7	0,0	0,4	0,6	0,0	0,0	1,1	6,6
	3	44,9	373,7	18133,4	0,0	0,0	0,0	0,0	0,0	0,0	0,0
	4	14,2	0,0	0,1	65,2	0,0	0,0	0,7	0,5	0,0	0,0
	5	10,7	0,0	0,0	0,0	443,8	0,0	0,0	0,0	0,0	0,0
	6	416,9	0,1	0,4	27,3	3,7	1165,3	0,0	0,3	0,0	0,0
	7	3,5	0,0	0,0	0,0	0,0	0,0	5,4	0,1	0,0	0,5
	8	3,3	0,0	0,0	0,0	0,0	0,0	0,0	17,8	0,0	0,0
	9	31,6	0,0	0,0	0,0	0,0	0,0	0,0	0,0	157,0	3,1
	10	122,3	0,0	0,0	0,0	0,3	0,0	0,0	0,0	2,8	3511,8

Tab. 5 Porovnání výsledků manuální a automatické klasifikace - v % z manuálně klasifikovaných bodů

		Manuální klasifikace									
		třída	1	2	3	4	5	6	7	8	9
Automatická klasifikace	1	76,0	0,1	0,0	1,7	0,4	0,3	5,5	1,0	10,5	3,0
	2	9,3	91,3	1,0	0,0	0,1	0,1	0,0	0,0	0,6	0,2
	3	1,0	8,5	99,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0
	4	0,3	0,0	0,0	69,3	0,0	0,0	11,1	2,7	0,0	0,0
	5	0,2	0,0	0,0	0,0	98,6	0,0	0,0	0,0	0,0	0,0
	6	9,5	0,0	0,0	29,0	0,8	99,7	0,0	1,4	0,0	0,0
	7	0,1	0,0	0,0	0,0	0,0	0,0	83,4	0,7	0,0	0,0
	8	0,1	0,0	0,0	0,0	0,0	0,0	0,0	94,1	0,0	0,0
	9	0,7	0,0	0,0	0,0	0,0	0,0	0,0	0,0	87,4	0,1
	10	2,8	0,0	0,0	0,0	0,1	0,0	0,0	0,0	1,5	96,8

Kódy tříd: 1 - *uncalss*, 2 – *ground*, 3 – *roadway*, 4 – *vehicles*, 5 – *gates*, 6 – *crash barriers*, 7 – *poles*, 8 – *signes*, 9 – *walls*, 10 – *vegetation*.

Tab. 5 na diagonále zobrazuje množství správně klasifikovaných bodů ze všech bodů ve třídě. Nejnižší úspěšnost (méně než 80%) klasifikace je u tříd 1 - *unclass* (body nezařaditelné do ostatních tříd) a 4 - *vehicles*. Nejvyšší úspěšnost (nad 95%) je u tříd 3 - *roadway*, 5 - *gates*, 6 - *crash barriers*, 10 - *vegetation*.

Další tabulka (tab. 6) popisuje úspěšnost algoritmu z hlediska počtu chyb I. a II. typu. Nejnižší procento chyb obou typů obsahují třídy 3 - *roadway* (1,0% chyb I. typu; 2,3% chyb II. typu), 5 - *gates* (1,4% chyb I. typu; 2,4% chyb II. typu) a 10 - *vegetation* (3,3% chyb I. typu; 3,5% chyb II. typu). Nevyšších hodnot v chybách I. nebo II. typu dosahují třídy 4 - *vehicles* (30,7% chyb I. typu; 19,2% chyb II. typu), 6 - *crash barriers* (jen 0,3% chyb I. typu, ale 27,8% chyb II. typu) a 7 - *poles* (16,6% chyb I. typu, 43,8% chyb II. typu).

Tab. 7 zobrazuje další používané kvantitativní hodnocení úspěšnosti algoritmu. Vzhledem k tomu, že jsou míry přesnosti a úplnosti odvozeny od počtu chyb I. a II. typu, je zřejmá spojitost mezi hodnotami v tab. 6 a tab. 7.

Tab. 6 Chyby I. a II. typu

Třída	1	2	3	4	5	6	7	8	9	10
I. typu [tis.]	1058,5	380,3	188,6	28,8	6,1	4,0	1,1	1,1	22,7	117,9
II. typu [tis.]	142,5	605,5	418,7	15,5	10,7	448,6	4,2	3,3	34,7	125,3
I. typu [%]	24,1	8,7	1,0	30,7	1,4	0,3	16,6	5,9	12,6	3,3
II. typu [%]	4,1	13,2	2,3	19,2	2,4	27,8	43,8	15,6	18,1	3,5

Tab. 7 Úplnost a přesnost klasifikace

Třída	1	2	3	4	5	6	7	8	9	10
Úplnost	0,760	0,913	0,990	0,693	0,986	0,997	0,833	0,941	0,874	0,968
Přesnost	0,959	0,868	0,977	0,808	0,976	0,722	0,562	0,844	0,819	0,966
F - míra	0,848	0,890	0,984	0,746	0,981	0,837	0,671	0,890	0,845	0,967

Kódy tříd: 1 - *unclass*, 2 - *ground*, 3 - *roadway*, 4 - *vehicles*, 5 - *gates*, 6 - *crash barriers*, 7 - *poles*, 8 - *signes*, 9 - *walls*, 10 - *vegetation*.

Podrobnější pohled na výsledky klasifikace poskytuje Tab. 8 Porovnání výsledků manuální a automatické klasifikace - počty objektů (pro třídy 4-5 a 7-8) tab. 8 **Chyba! Nenalezen zdroj odkazů.**, obsahující počty klasifikovaných objektů. Tato statistika se týká pouze tříd, které obsahují objekty bodového (ne liniového) charakteru a u kterých bylo možné určit počty objektů v souboru: 4 - *Vehicles*, 5 - *Gates*, 7 - *Poles*, 8 - *Signes*.

Za správně klasifikovaný byl považován objekt, který byl klasifikován alespoň z části. Ze 126 vozidel nebo jejich částí bylo ve třídě *vehicles* správně zařazeno 120, zbývajících 5 bylo chybně zařazeno do třídy *crash barriers* a jedno zůstalo neklasifikováno. Objekty třídy *gates* (4) byly správně klasifikovány ve všech případech. Pět vozidel a tři značky (šlo o informativní tabulky o rozměru do 0,5m nacházející

se v bezprostřední blízkosti svodidel) byly chybně zařazeny do třídy *crash barriers*.

Z 59 objektů třídy *poles* bylo správně klasifikováno 54, zbývající 4 (tři směrové sloupky a jeden vzdálený sloup vysokého napětí) zůstaly neklasifikovány, jedna SOS hláska byla chybně zařazena do třídy *vehicles* – tato hláska představuje 2/3 všech chybně klasifikovaných bodů dané třídy, tedy 2/3 z hodnoty chyb I. typu u této třídy - 16,6%). Do třídy *poles* jsou chybně zařazeny dvě značky/tabulky do velikosti 0,5m a především 6 kmenů stromů, které mají za následek enormní míru chyb II. typu (43,8%). Těchto šest objektů je sice v menšině oproti 54 správně klasifikovaným objektům, ovšem jsou představovány výrazně vyšším počtem bodů, než většina objektů ve třídě (směrové sloupky).

Třída *signs* obsahuje 26 objektů, z nichž správně zařazeno bylo 18. Z osmi nenalezených je 7 kilometrovníků nebo informativních značek s rozměrem maximálně 0,5m (nesplňují podmínky pro trvalé standardní svislé dopravní značky - umístění 1,2m nad výškou terénu, nebo minimální rozměr). Pouze jedna dopravní značka splňující tyto podmínky byla chybně zařazena do třídy *vehicles*. Jediný objekt, který byl chybně přiřazen do třídy *signs* je závora na připojující se komunikaci vybavená reflexními prvky.

Tab. 8 Porovnání výsledků manuální a automatické klasifikace - počty objektů (pro třídy 4-5 a 7-8)

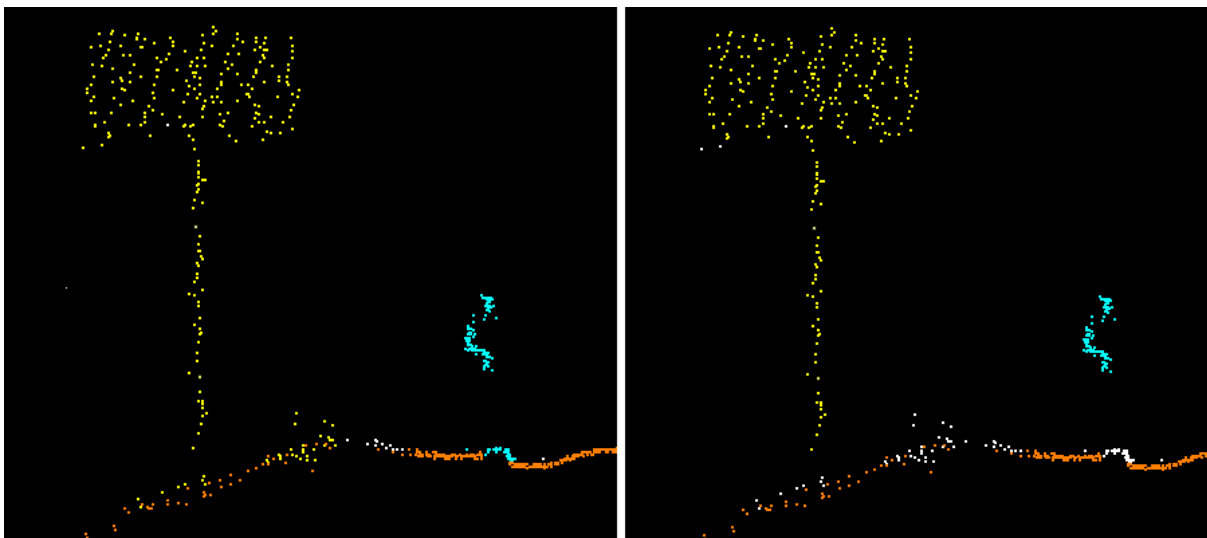
		Manuální klasifikace									
		1	2	3	4	5	6	7	8	9	10
Automatická klasifikace	třída										
	1	0	0	0	1	0	0	4	2	0	0
	2	0	0	0	0	0	0	0	0	0	0
	3	0	0	0	0	0	0	0	0	0	0
	4	0	0	0	120	0	0	1	1	0	0
	5	0	0	0	0	4	0	0	0	0	0
	6	0	0	0	5	0	0	0	3	0	0
	7	1	0	0	0	0	0	54	2	0	6
	8	1	0	0	0	0	0	0	18	0	0
	9	0	0	0	0	0	0	0	0	0	0
10	1	0	0	0	0	0	0	0	0	0	
	Celkem				126	4		59	26		
	úspěšnost [%]				95	100		92	69		

5.3 Příklady chyb v klasifikaci

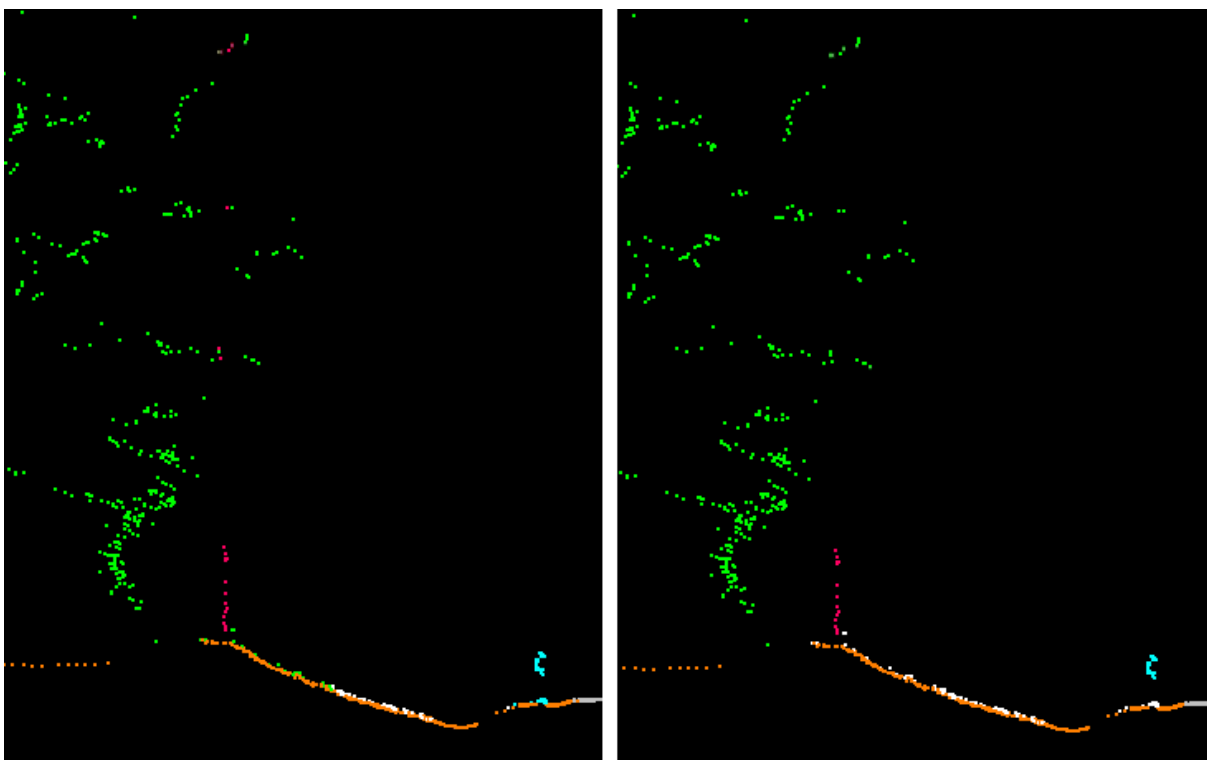
Následující odstavce podrobně popisují situace, ve kterých algoritmus selhává. Všechny uvedené příklady vychází z porovnání automaticky a manuálně klasifikovaného souboru popsáno v předchozí části textu.

Použitá metoda automatické klasifikace již z principu počítá s jistou mírou generalizace, která vnáší do procesu odpovídající míru chybovosti. Mírou generalizace je v tomto případě nahrazení klasifikace bodu klasifikací pixelu a chyba je dána tím, že body odpovídající jednomu pixelu (až na výjimky *ground*, *roadway* a *gates*) jsou přiřazeny stejné třídě. Tyto chyby jsou tedy metodě vlastní a počítá se s nimi. Algoritmus tím, že klasifikuje všechny body příslušející klasifikovanému pixelu (nad hranicí terénních

bodů), zařazuje do třídy i body, které do ní nenáležejí. Tyto body jdou nejčastěji na vrub třídy *unclass* (body nízko nad terémem) – obr. 59 zobrazuje body nízké vegetace a body obrubníku automatickou klasifikací zařazené do tříd *signs* (žlutá) a *crash barriers* (tyrkysová). Problém nastává také u vegetace přesahující přes plot - současně s plotem může být do třídy *walls* (růžová) přiřazena i část vegetace (obr. 60).



Obr. 59 Porovnání automatické (vlevo) a manuální (vpravo) klasifikace; *unclass* – bílá, *ground* – oranžová, *crash barriers* – tyrkysová, *signs* – žlutá (blok pt09)

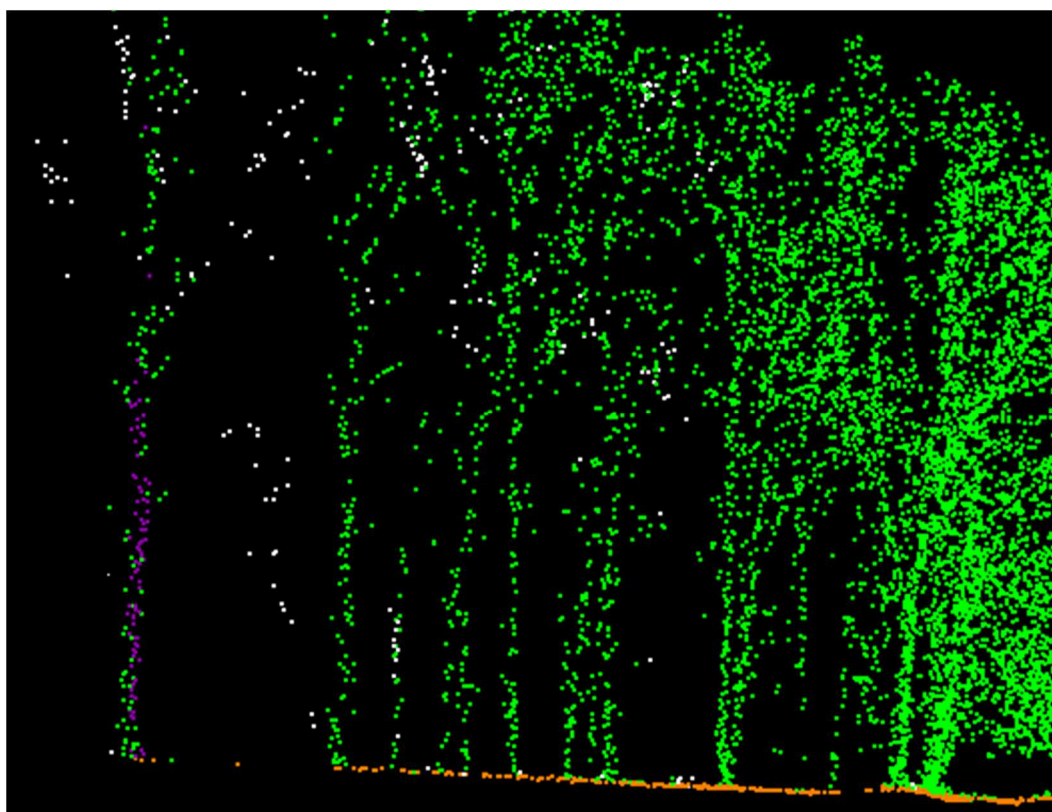


Obr. 60 Porovnání automatické (vlevo) a manuální (vpravo) klasifikace; *unclass* – bílá, *ground* – oranžová, *crash barriers* – tyrkysová, *walls* – růžová, *vegetation* – zelená (blok pt09)

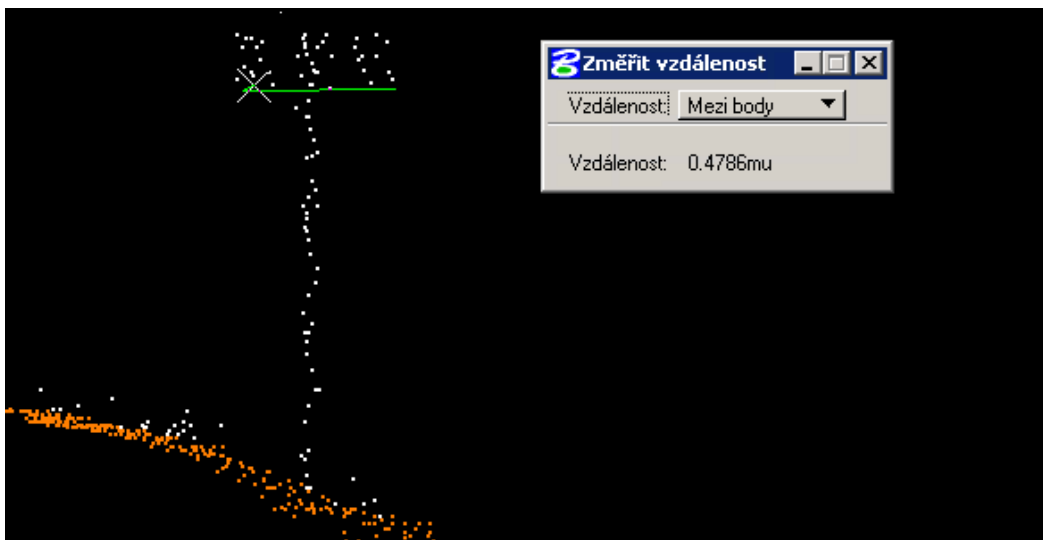
Obr. 61 ukazuje příklad chyb II. typu při klasifikaci třídy *poles*. Zařazení kmenu stromu do třídy

poles (fialová) nastává zejména u stromů s vyšší vzdáleností od pozemní komunikace a to v případě souvislého porostu. Koruna je naskenována pouze sporadicky (přes bližší koruny nemá paprsek laseru potřebný dosah) a netvoří souvislý objekt, zatímco kmen je nasnímaný podrobně.

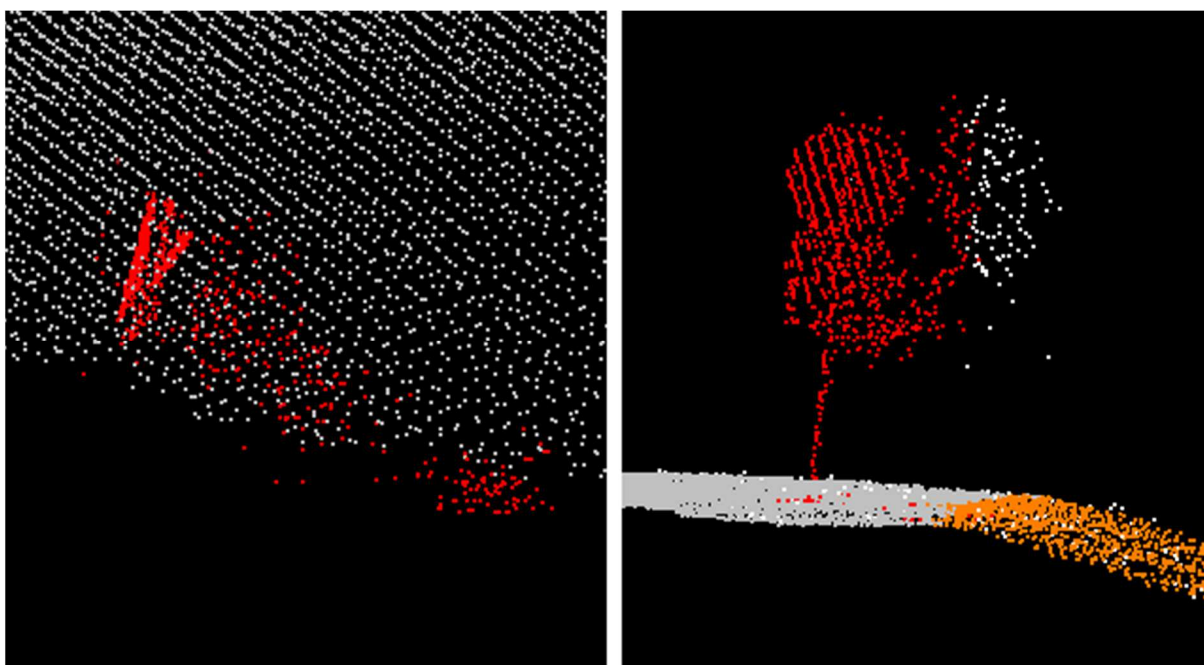
Dalším problémem je klasifikace drobných informativních tabulek nespĺňujících kritéria stanovená pro svislé dopravní značky (výška nad terénem a velikost). Tyto informativní tabulky by bylo vhodné zařadit do třídy *signs*, ale nemívají dostatečně reflexní povrch a je problematické oddělit je od vegetace a jiných objektů (ať už na základě intenzity, rozměru či polohy). Zůstávají neklasifikované (obr. 62), případně zařazené do třídy *poles* nebo třídy *crash barriers* (pokud stojí v bezprostřední blízkosti svodidel).



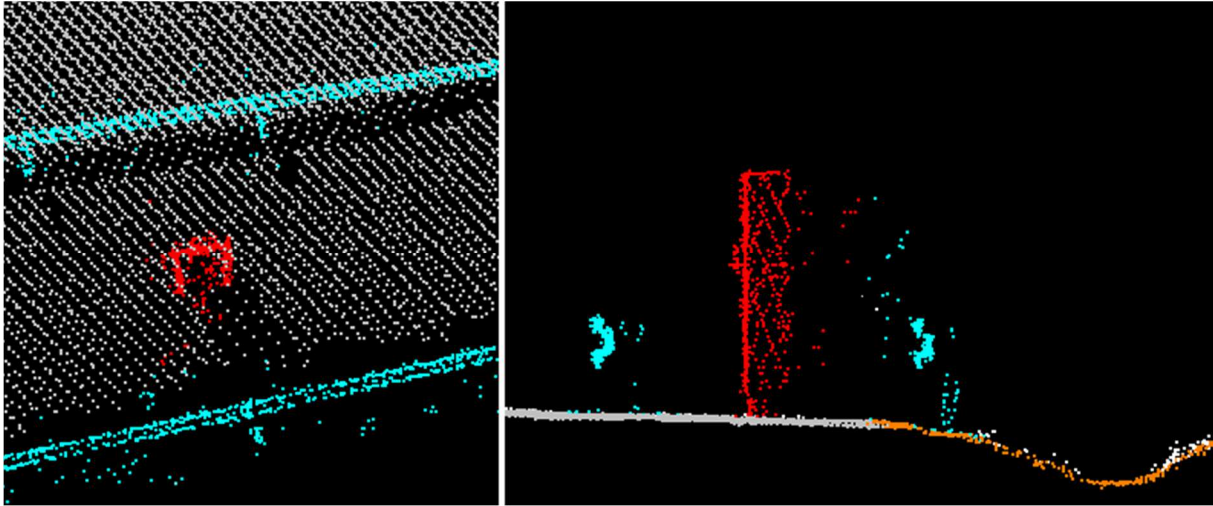
Obr. 61 Kmen stromu přiřazený do třídy *poles*; *unclass* – bílá, *ground* – oranžová, *vegetation* – zelená, *poles* – fialová (blok pt01)



Obr. 62 Neklasifikována informativní tabulka; *unclass* – bílá, *ground* – oranžová (blok pt01)



Obr. 63 Dopravní značka přiřazená do třídy *vehicles*; *unclass* – bílá, *ground* – oranžová, *roadway* – šedá, *vehicles* – červená (blok pt02)

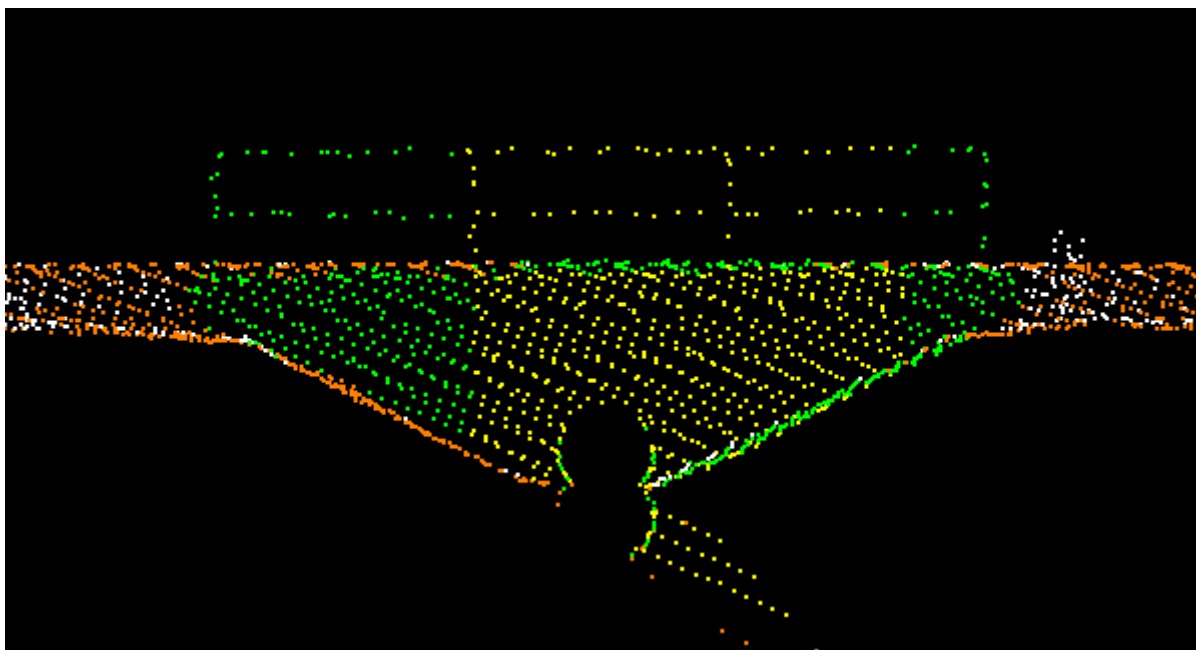


Obr. 64 SOS hláska přiřazená do třídy *vehicles*; *unclass* – bílá, *ground* – oranžová, *roadway* – šedá, *vehicles* – červená, *crash barriers* - tyrkysová (blok pt07)

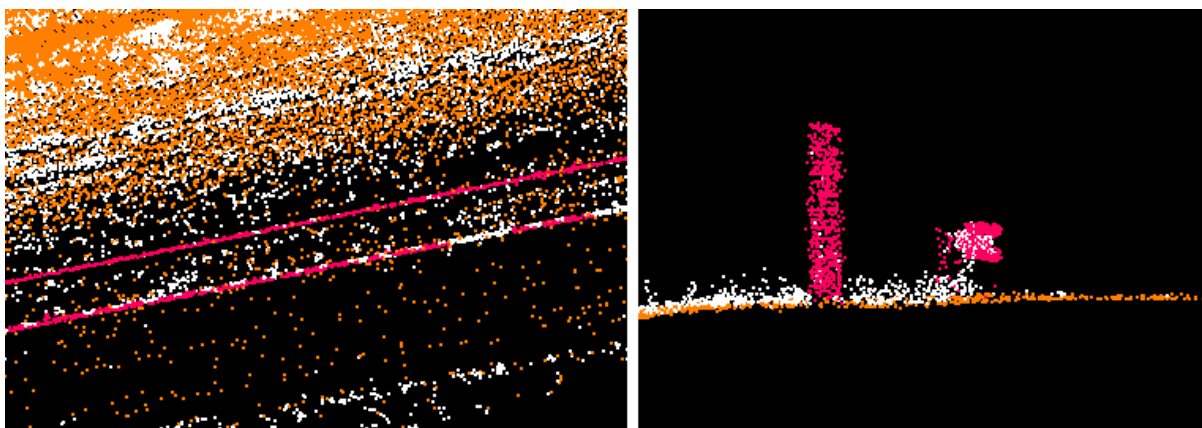
Jediný případ, kdy nebyla v ověřovacím souboru správně klasifikována standardní svislá dopravní značka, zobrazuje obr. 63. Dopravní značka na odbočující komunikaci nacházející se nad povrchem klasifikovaným jako *roadway* a s početnými vícenásobnými odrazy za sebou byla zařazena do třídy *vehicles* (červená). Vícenásobné odrazy vytvořily za svislou dopravní značkou shluk bodů, díky němuž se celý objekt jeví jako větší, než ve skutečnosti je, a byl proto chybně zařazen mezi vozidla. Ze stejného důvodu byla do třídy *vehicles* chybně zařazena SOS hláska (obr. 64).

Problém nastává i při klasifikaci stěny s propustkem, přes který prochází komunikace souběžná s dokumentovanou dálnicí. Automatická klasifikace terénních bodů si obecně neumí poradit se skokovými změnami terénu a je proto klasifikováno dno příkopu vedoucího do propustku, ale už ne povrch blízké komunikace. Povrch komunikace neklasifikovaný jako terén se po přepočtu na výšku nad terénem může jevit jako jiný objekt (obr. 65) – v tomto případě billboard (třída *signs*, žlutá barva) nebo vegetace (zelená barva).

V případě, kdy podél dokumentované komunikace probíhá ještě jedna souběžná, je tato komunikace naskenována pouze řídce a není možné její povrch zařadit do třídy *roadway* (obr. 66). Z toho důvodu nejsou v jejím okolí vyhledávána svodidla a tato mohou být přiřazena do třídy *walls*.



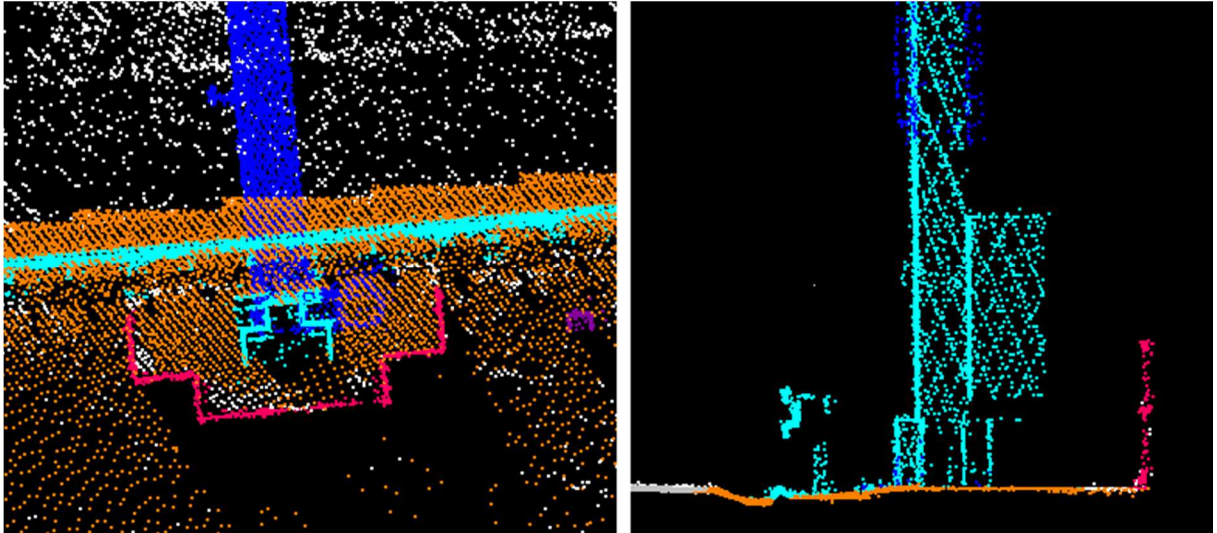
Obr. 65 Stěna s propustkem a částí komunikace přiřazená do tříd *signs* a *vegetation*; *unclass* – bílá, *ground* – oranžová, *vegetation* – zelená, *signs* – žlutá (blok pt03)



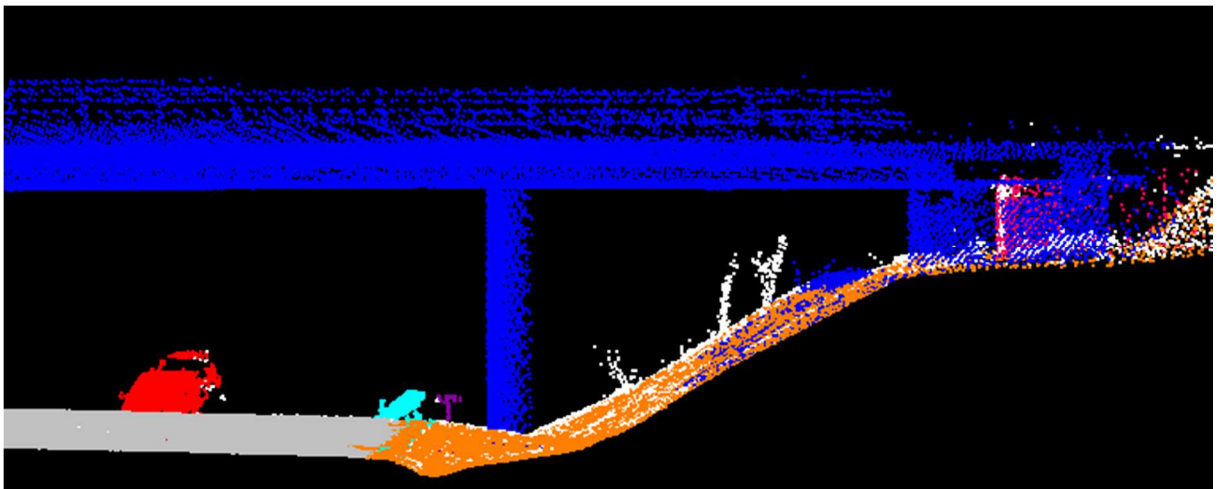
Obr. 66 Svodidla komunikace souběžné s dálnicí přiřazená do třídy *walls*, *unclass* – bílá, *ground* – oranžová, *walls* – růžová (blok pt06)

Problematická je i klasifikace objektů bezprostředně si blízkých (v obraze tvořících jeden objekt). Jak již bylo zmíněno výše, informativní tabulky v blízkosti svodidel jsou nezřídka zařazeny do třídy *crash barriers*. Jak ukazuje obr. 67, stejná situace může nastat i u sloupu mýtné brány.

Poměrně komplikovaná je úloha klasifikace mostů, resp. jejich částí nacházejících se pod stanovenou úrovní nad terénem – tj. konců mostu a sloupů. U těchto částí jsou klasifikovány objekty ve všech výškách nad terénem, např. i nízká vegetace na svahu pod mostem (obr. 68; třída *gates* - modrá), zároveň se u mostů může stát, že některé jeho části na koncích mostu mohou být klasifikovány ve třídě *walls* (růžová). Při automatické klasifikaci testovacích ani ověřovacích dat, se nestalo, že by objekt mostu nebo mýtné brány nebyl klasifikován vůbec.



Obr. 67 Sloup mýtné brány přiřazený do třídy *crash barriers*; *roadway* – šedá, *ground* – oranžová, *crash barriers* – tyrkysová, *gates* – modrá, *walls* – růžová (blok pt09)



Obr. 68 Klasifikace mostu - třída *gates*; *unclass* – bílá, *roadway* – šedá, *ground* – oranžová, *crash barriers* – tyrkysová, *gates* – modrá, *walls* – růžová, *vehicles* – červená (blok pt05)

6 Shrnutí

6.1 Výsledky

Cílem práce bylo potvrdit následující hypotézu:

- Metody využívané k segmentaci a klasifikaci obrazových dat lze aplikovat na data mobilního laserového skenování.

V rámci práce byla formulována metoda klasifikace mračna bodů z dat MLS pro účely dokumentace pozemních komunikací. Dále byl vytvořen algoritmus, který tuto klasifikaci realizuje - převádí mračno bodů na obrazová data, tato klasifikuje do předem definovaných tříd a výsledky klasifikace přenáší na originální mračno bodů.

Funkčnost metody byla ověřena na vybraném vzorku dat (úseku dálnice D11 o délce 1,8km obsahujícím všechny typy objektů, jejichž klasifikace byla v metodě řešena) s celkovou správností klasifikace 94,5% (správně klasifikovaných bodů z celkového počtu bodů). Kvalitu klasifikace jednotlivých tříd udává prostřednictvím ukazatelů přesnosti, úplnosti a F-míry tab. 9 Tab. 9. Z tabulky je zřejmé, že nejvyšší úspěšnosti dosahoval algoritmus v klasifikaci tříd 3 – *roadway*, 5 – *gates* a 10 – *vegetation* (hodnoty přesnosti, úplnosti a F-míry 0,97 a víc). Pouze u tříd 4 – *vehicles*, 6 – *crash barriers*, 7 – *poles* klesl některý ze sledovaných parametrů pod 0,80. Podrobnější rozbor klasifikace jednotlivých tříd je uveden v oddíle 5.2.

Tab. 9 Úplnost a přesnost klasifikace

Třída	1	2	3	4	5	6	7	8	9	10
Úplnost	0,76	0,91	0,99	0,69	0,99	1,00	0,83	0,94	0,87	0,97
Přesnost	0,96	0,87	0,98	0,81	0,98	0,72	0,56	0,84	0,82	0,97
F - míra	0,85	0,89	0,98	0,75	0,98	0,84	0,67	0,89	0,85	0,97

Kódy tříd: 1 - *uncalss*, 2 – *ground*, 3 – *roadway*, 4 – *vehicles*, 5 – *gates*, 6 – *crash barriers*, 7 – *poles*, 8 – *signes*, 9 – *walls*, 10 – *vegetation*.

V odborných publikacích prezentované algoritmy pro klasifikaci dat mobilního laserového skenování se zpravidla věnují pouze klasifikaci specifické třídy (*Riveiro a kol. (2016)*, *Rodríguez-Cuenca (2015)*, *Yu a kol. (2015)*), např. svislého/vodorovného dopravního značení, a ne komplexní klasifikaci mračna. Pokud algoritmy řeší klasifikaci mračna komplexně, zaměřují se spíše na městské prostředí (*Aijazi a kol., 2013*). Nebyl nalezen nástroj řešící komplexní klasifikaci mračna bodů zobrazujících pozemní komunikaci a její okolí mimo město, se kterým by bylo možné vytvořený algoritmus v detailu porovnat. Tab. 10 ukazuje úspěšnost několika vybraných algoritmů zabývajících se detekcí bodů určité třídy nebo klasifikací obecně na datech MLS. Z uvedených hodnot je zřejmé, že úspěšnost klasifikace představovaného algoritmu alespoň u některých tříd za úspěšností těchto algoritmů nijak nezaostává.

Vytvořením metody a ověřením její funkčnosti byla hypotéza potvrzena.

Tab. 10 Přehled úspěšnosti klasifikačních algoritmů

Klasifikace	Úplnost	Přesnost	F-míra	Zdroj
Svislé dopravní značky, mimo město	0,94	0,98	-	<i>Riveiro a kol. (2016)</i>
Tyčové objekty, město	0,95	0,99	-	<i>Rodríguez-Cuenca (2015)</i>
Třídy – <i>building, road, pole, car, tree, unclassified</i> , město			0,83-0,94*	<i>Aijazi a kol. (2013)</i>
Vodorovné dopravní značení	0,93	0,92	0,93	<i>Yu a kol. (2015)</i>

*výsledky z pěti datových souborů

6.2 Závěry

Záměrem bylo vytvořit metodu jednoduchou, výpočetně nenáročnou (oproti metodám provádějícím výpočty přímo na mračnu bodů – ať už organizovaném či neorganizovaném) a vycházející ze znalosti klasifikovaných objektů. Třídy byly zvoleny na základě znalosti aktuálně řešených úloh souvisejících s dokumentací pozemních komunikací a jejich okolí. Rozdělení mračna do zvolených tříd vytváří podklady pro sofistikovanější metody analýzy prostorových dat – např. body třídy *roadway* mohou sloužit jako vstupní data při analýze stavu vozovky nebo extrakci vodorovného dopravního značení. K analýze viditelnosti svislého dopravního značení lze využít objekty třídy *signs* a potenciální překážky hledat ve třídě *vegetation*. Další třídy mohou být využity při evidenci příslušenství pozemní komunikace, analýze bezpečnosti, sledování stavu vegetace, detekci změn a v mnoha dalších aplikacích.

Použití algoritmu vzhledem k množství parametrů vycházejících z legislativního rámce ČR je poměrně úzce vázáno na daný typ pozemní komunikace (dálnice a rychlostní komunikace) a danou lokalitu (ČR). Tyto parametry je ovšem možné upravit konkrétním podmínkám a nebrání aplikaci algoritmu v odlišném prostředí. Ze zkušenosti při vývoji a testování metody lze navrhnout následující obecná doporučení:

- při přípravě dat je vhodné vytvářet bloky s překryvem (o velikosti odpovídající minimálně dvěma pixelům) a tento překryv po klasifikaci odstranit (oříznout), hodnoty pixelů na okraji datového souboru jsou při filtraci obrazu ovlivněny absencí hodnot sousedních (prázdných) pixelů
- při aplikaci algoritmu na nový datový soubor je vhodné otestovat algoritmus na malém souboru, případně upravit některé parametry a teprve poté aplikovat algoritmus na celý datový soubor

6.3 Diskuse

Klasifikace mračna bodů je prováděna prostřednictvím rozdělení mračna do pravidelných voxelů a vytvořením čtyřrozměrné matice, jejíž první tři rozměry odkazují na polohu voxelu a čtvrtý rozměr na parametry reprezentující body spadající do daného voxelu. Z vybraných hodnot této matice jsou vytvářeny rastrové obrazy, na které jsou aplikovány postupy ke zpracování obrazových dat s cílem vybrat pixely obsahující body hledané třídy. Cíle je dosaženo kombinací funkcí dostupných v použitém nástroji, která je vytvořena na základě testování na několika datových souborech. Není samozřejmě vyloučeno, že by bylo možné dosáhnout podobných, ne-li lepších výsledků jinou kombinací podobných

funkcí, to by mohl ukázat další výzkum. Základní princip metody – tedy převod klasifikace do rastrových dat a řešení úloh na úrovni pixel-pixel místo bod-bod přináší výrazné snížení náročnosti úlohy a vysokou efektivitu, kterou je možno vyvážit míru chybovosti algoritmu danou principem metody.

Přesto, že úspěšnost algoritmu předčila očekávání, zůstává tu prostor pro zlepšení. Testování ukázalo, že při nastavování různých parametrů filtrů vedla snaha o snížení počtu chyb typu I ke zvýšení chyb typu II a obráceně. Cílem nastavení bylo vhodné vyvážení obou typů chyb (s ohledem na konkrétní třídu). Přes veškerou snahu, zůstává poměrně vysoké množství chyb typu II (*FP – false positive*) u klasifikace svodidel a sloupů elektrického vedení. Naskenovaná vozidla algoritmus v některých případech chybně zařazuje do třídy svodidel. Problémem je široká variabilita svodidel (betonová, ocelová, lanová, často doplněná prorůstající vegetací). Další omezení při klasifikaci na základě velikosti, tvaru nebo orientace svodidel vedlo při testování k navýšení množství chyb typu I (*FN – false negative*). Omezení na základě polohy vůči třídě *roadway* (pouze na okrajích) neřešilo situaci v místech, kde vozovka pokrývá i oblast středního dělicího pásu. Dalším problémem zůstávají kmeny stromů zařazené mezi sloupy (třída *poles*). Ve vzrostlých lesních porostech poblíž komunikace se stává, že u stromů dále od vozovky je poměrně podrobně naskenovaný kmen, ale velice řídké koruna a takové stromy jsou od sloupů těžko odlišitelné. Možným řešením je propojení s dalším typem dat, např. s daty z UAV nebo leteckého snímkování, a využití barev případně vegetačních indexů (v případě nasnímání lokality ve viditelném i v blízkém infračerveném pásmu) pro odlišení vegetace.

Otázkou také zůstává do jaké vzdálenosti od skeneru (vozovky) klasifikaci bodů řešit. Obecně lze říci, že čím dále od skeneru objekt je, s tím menší hustotou bodů je nasnímán. To, jak komplexně bude objekt nasnímán, ovšem závisí nejen na vzdálenosti od skeneru, ale také na konkrétní situaci (zda je komunikace v zářezu nebo v náspu, kolik dalších objektů je blíže skeneru a do jaké míry brání laserovým paprskům v dalším průchodu), nelze tedy stanovit žádnou konstantní vzdálenost, za kterou by mělo význam mračno oříznout a jeho klasifikaci neřešit.

Nevýhodou algoritmu je také závislost klasifikace jednotlivých tříd. Prvky některých tříd (např. *signs, crash barriers*) jsou vyhledávány na základě dané vzdálenosti od třídy *roadway*, jiné vychází z dat, ze kterých jsou prvky zobrazující dříve klasifikované třídy odstraněny (např. do třídy *vegetation*, jsou zařazeny jen pixely dříve nezařazené do tříd *signs, vehicles...*). Chyby v klasifikaci určitých tříd se tedy mohou přenášet na další třídy. Tuto závislost nebylo možné odstranit, protože použité omezené množství informací o bodech spadajících do jednotlivých voxelů nestačilo k jednoznačnému odlišení některých tříd. Bylo tedy nutné stanovit ještě podmínky vzájemných vztahů (např. vzdáleností) k jiným třídám. Pro další výzkum by bylo vhodné zvážit využití dalších parametrů, kterými by bylo možné popsat body spadající do jednotlivých pixelů – např. výškový rozptyl bodů v pixelu, a zbavit se tak vzájemné závislosti jednotlivých tříd.

Dále stojí za úvahu přenos klasifikace rastrů na voxely a z nich teprve na mračno. Klasifikace jednotlivých tříd probíhá zpravidla na základě dat jedné nebo více vrstev voxelové matice a rastr představující klasifikovanou třídu je platný pro tyto vrstvy. Voxely v ostatních vrstvách (výškách) by mohly být přiřazovány do dané třídy pouze v případě, že tvoří souvislé objekty s již klasifikovanými voxely. To by mohlo rozšířit možnosti klasifikace více objektů vyskytujících se v různých výškách ale

spadajících do jednoho pixelu.

Na závěr lze říci, že výsledky testování vytvořené metody potvrzují platnost předkládané hypotézy a nabízí možný směr dalšího vývoje.

Literatura

- Aijazi, A.K.; Checchin, P.; Trassoudaine, L., 2013. *Segmentation Based Classification of 3D Urban Point Clouds: A Super-Voxel Based Approach with Evaluation*. Remote Sens., 5, 1624-1650.
- Axelsson, P., 1999. *Processing of Laser Scanner Data – Algorithms and Applications*. ISPRS Journal of Photogrammetry & Remote Sensing, 54 (1999), str. 138-147.
- Axelsson, P., 2000. *DEM Generation from Laser Scanner Data Using Adaptive TIN Models*. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. 33 (B4/1), str.110-117.
- Barnea, S.; Filin, S., 2008. *Segmentation of Terrestrial Laser Scanning Data by Integrating Range and Image Content*. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. 37 (B3), str. 745-750, Beijing.
- ČSN 73 6101, 2004. *Projektování silnic a dálnic*. Český normalizační institut, Praha.
- ČSN 73 6058, 2011. *Jednotlivé, řadové a hromadné garáže*. Český normalizační institut, Praha.
- ČSN EN 50 341 -1 ed. 2, 2013. *Elektrická venkovní vedení s napětím nad AC 1 kV - Část 1: Obecné požadavky - Společné specifikace*. Český normalizační institut, Praha.
- ČSN 73 6201, 2008. *Projektování mostních objektů*. Český normalizační institut, Praha.
- Denis, E.; Burck, R.; Baillard, C., 2010. *Towards Road Modelling from Terrestrial Laser Points*. The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 38 (3A), str. 293-298.
- Díaz-Vilariño, L. a kol., 2015. *Automatic classification of urban pavements using mobile LiDAR data and roughness descriptors*, Construction and Building Materials, Volume 102, Part 1, 15 January 2016, Pages 208-215, ISSN 0950-0618, <https://doi.org/10.1016/j.conbuildmat.2015.10.199>.
- Faltýnová, M., 2010. *Možnosti zpracování laserových dat – dokumentace liniové stavby*. Diplomová práce (Ing.), ČVUT v Praze, Fakulta stavební, Katedra mapování a kartografie, Praha.
- Filin, S.; Pfeifer, N., 2006. *Segmentation of Airborne Laser Scanning Data Using a Slope Adaptive Neighborhood*. ISPRS Journal of Photogrammetry & Remote Sensing, 60 (2006), str. 71-80.
- Golovinskiy, A.; Kim, V.G.; Funkhouser, T., 2009. *Shape-based Recognition of 3D Point Clouds in Urban Environments*. 2009 IEEE 12th International Conference on Computer Vision, 29.zář-2.října 2009, Kyoto, str. 2154-2161, E-ISBN 978-1-4244-4419-9.
- González-Jorge, H. a kol., 2013. *Automatic segmentation of road overpasses and detection of mortar efflorescence using mobile LiDAR data*. Optics & Laser Technology 54 (2013), str.353-361, doi:10.1016/j.optlastec.2013.06.023.
- GRP 500 [online], [cit. 2017-08-15]. Dostupné z: <http://www.ambergtechnologies.ch>
- Guan, H. a kol., 2015. *Using Mobile LiDAR Data for Rapidly Updating Road Markings*. IEEE Trans. Intell. Transp. Syst. 2015, 16, 2457–2466.

- Halounová, L., 2009. *Zpracování obrazových dat*, Praha: Česká technika – nakladatelství ČVUT.
- Hebel, S.; Stilla, U., 2008. *Pre-classification of Points and Segmentation of Urban Objects by Scan Line Analysis of Airborne Lidar Data*. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 37 (B3a), str. 105-110, Beijing.
- Holgado-Barco, A. a kol., 2014. *An automated approach to vertical road characterisation using mobile LiDAR systems: Longitudinal profiles and cross-sections*. ISPRS Journal of Photogrammetry and Remote Sensing 2014, 96, 28–37.
- Chang, Y.-CH. a kol., 2008. *Automatic Classification of Lidar Data into Ground and Non-ground Points*. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 37 (B4), str. 457-462.
- Jaakkola, A. a kol., 2008. *Retrieval Algorithms for Road Surface Modelling Using Laser-based Mobile Mapping*. Sensors, r. 8, sv. 9, str. 5238-5249.
- Lantieri, C. a kol., 2015. *Mobile laser scanning system for assessment of the rainwater runoff and drainage conditions on road pavements*. International Journal of Pavement Research and Technology, 8(1), str.1-9.
- Lehtomaki, M. a kol., 2010. *Detection of Vertical Pole-like Objects in a Road Environment Using Vehicle-based Laser Scanning Data*. Remote Sensing, MDPI, r.2, sv.3, str.641-664.
- Lindenbergh, R.; Pietrzyk, P., 2015. *Change detection and deformation analysis using static and mobile laser scanning*. Applied Geomatics (2015)7, str. 65-74.
- Liu, J.-G.; Mason, P. J., 2016. *Image processing and GIS for remote sensing: techniques and applications*. Second edition., Chichester: John Wiley.
- Luo, C.; Jwa, Y.; Sohn, G., 2014. *Context based multiple railway object recognition from mobile laser scanning data*. IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2014), Québec, Canada (2014), str. 3602-3605.
- LYNX HS-600 [online], [cit. 2017-08-15]. Dostupné z: <http://www.teledyneoptech.com>.
- Merlin [online], [cit. 2017-08-15]. Dostupné z: <http://www.renishaw.com>.
- Miura, N.; Jones, S.D., 2010. *Characterizing forest ecological structure using pulse types and heights of airborne laser scanning*. Remote Sensing of Environment, r. 114, sv. 5, str. 1069-1076.
- Olsen, M. J., 2013. *Guidelines for the Use of Mobile LiDAR in Transportation Applications*. National Academy of Sciences, Washington, DC. ISSN 0077-5614, ISBN 978-0-309-25914-9.
- Pavelka, K., a kol., 2013. Výzkumná zpráva k projektu CG 912-105-520 MDČR 2009-2010. náb.gen.Svobody 3: MDČR. 2010, 3. Dostupné z: <http://lfgm.fsv.cvut.cz>.
- Pu, S. a kol., 2011. *Recognizing Basic Structures from Mobile Laser Scanning Data for Road Inventory Studies*. ISPRS Journal of Photogrammetry and Remote Sensing 66 (2011) 528-539.
- Puente, I. a kol., 2014. *Automatic detection of road tunnel luminaires using a mobile LiDAR system*.

Measurement, 47 (2014), str. 569-575, doi:10.1016/j.measurement.2013.09.044.

Projekt CG912-105-520: Možnosti monitorování stavu a změn v okolí hlavních komunikací metodami dálkového průzkumu Země a laserového skenování a jejich využití pro realizaci udržitelného rozvoje dopravy [online], [cit. 16_08_2017]. Dostupné z: <http://lfgm.fsv.cvut.cz/mdcr/index.html>.

Pytel, J., 2006. *NOP*. Geoinformatics FCE, ČVUT v Praze, Vol. 2, ISSN 1802-2669.

Rao, R. a kol., 2006. *Ground Surface Extraction from Side-scan (Vehicular) Lidar*. Proceedings of the MAPPS/ASPRS Fall Conference, San Antonio, Texas, USA.

Riveiro, B. a kol., 2016. *Automatic Segmentation and Shape-Based Classification of Retro-Reflective Traffic Signs from Mobile LiDAR Data*. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, sv. 9, č. 1, str. 295-303.

Rodríguez-Cuenca, B. a kol., 2015. *Automatic Detection and Classification of Pole-Like Objects in Urban Point Cloud Data Using an Anomaly Detection Algorithm*. Remote Sens. 2015, 7, 12680-12703.

Rutzinger, M. a kol., 2010. *Detection and Modelling of 3D Trees from Mobile Laser Scanning*. The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 38 (5), str. 520-525, Newcastle upon Tyne.

Saarinen, N. a kol., 2013. *Area-based approach for mapping and monitoring riverine vegetation using mobile laser scanning*. Remote Sensing, Vol. 5, str. 5285-5303, doi: 10.3390/rs5105285.

Shan, J.; Toth, CH., 2009. *Topographic Laser Ranging and Scanning: Principles and Processing*. Boca Raton, Florida, USA: CRC Press, 2009, str. 590, ISBN-13:978-1-4200-5142-1.

Sithole, G.; Vosselman, G., 2004. *Experimental Comparison of Filter Algorithms for Bare-Earth Extraction from Airborne Laser Scanning Point Clouds*. ISPRS Journal of Photogrammetry and Remote Sensing 59 (2004), str. 85-101.

Soilan, M. a kol. 2016. *Traffic sign detection in MLS acquired point clouds for geometric and image-based semantic inventory*. ISPRS Journal of Photogrammetry and Remote Sensing 2016, 114, 92–101.

Spel, 2017 [online], [cit. 2017-09-1]. Dostupné z: https://www.spel.cz/download/eng/Prezentace_SOS_hlasky.pdf

Street Mapper IV [online], [cit. 2017-08-15]. Dostupné z: <http://www.3dlasermapping.com>.

TP 58 - Směrové sloupky a odrazky - Zásady pro používání, 2008. Technické podmínky, Ministerstvo dopravy. Dostupné z: <http://www.pjpk.cz/technicke-podminky-tp/>.

TP 65 - Zásady pro dopravní značení na pozemních komunikacích, 2013. Technické podmínky, Ministerstvo dopravy. Dostupné z: <http://www.pjpk.cz/technicke-podminky-tp/>.

TP 114 - Svodidla na pozemních komunikacích, 2015. Technické podmínky, Ministerstvo dopravy. Dostupné z: <http://www.pjpk.cz/technicke-podminky-tp/>.

Vosselman, G.; Liang, Z., 2009. *Detection of Curbstones in Airborne Laser Scanning Data* The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 38

(3/W8), str. 111-116, Paris.

Vosselman, G.; Maas, H.-G.; et al., 2010. *Airborne and Terrestrial Laser Scanning*. Dunbeath, Scotland, UK: Whittles Publishing, 2010, str.318, ISBN 978-1904445-87-6.

Wang, M.; Tseng, Y., 2004. *Lidar Data Segmentation and Classification Based on Octree Structure*. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. 35 (3), str. 308-313, Istanbul.

Wang, J., González-Jorge, H., Lindenbergh, R., Arias-Sánchez, P. and Menenti, M., 2013. *Automatic estimation of excavation volume from laser mobile mapping data for mountain road widening*. Remote Sensing, 5(9), str.4629-4651.

QUO-reklama.cz [online]. [cit. 3. 9. 2017].

Xiao, W. a kol., 2016. *Street-side vehicle detection, classification and change detection using mobile laser scanning data*. ISPRS Journal of Photogrammetry and Remote Sensing 2016, 114, 166–178.

Yang, B., Fang, L., 2014. Automatic extraction of 3-D railway tracks from mobile laser scanning point clouds. IEEE Journal of Selected Topics In Applied Earth Observation and Remote Sensing, r. 7, sv. 12, str. 4750-4761, doi: 10.1109/JSTARS.2014.2312378.

Yu, Y. a kol., 2014. *Pairwise three-dimensional shape context for partial object matching and retrieval on mobile laser scanning data*. IEEE Geoscience and Remote Sensing Letters. sv. 11, č.5, str. 1019-1023, doi: 10.1109/LGRS.2013.2285237.

Yu, Y. a kol., 2015. *Learning Hierarchical Features for Automated Extraction of Road Markings From 3-D Mobile LiDAR Point Clouds*. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 2015;8:709–726. doi: 10.1109/JSTARS.2014.2347276.

Z+F Profiler 9012 [online], [cit. 2017-08-14]. Dostupné z: <http://www.zf-laser.com/>.

ZEB-CAM [online], [cit. 2017-08-15]. Dostupné z: <http://www.3dlasermapping.com>.

Zhu, Q. a kol., 2017. *Robust point cloud classification based on multi-level semantic relationships for urban scenes*, ISPRS Journal of Photogrammetry and Remote Sensing, sv. 129, 2017, str. 86-102, ISSN 0924-2716.

Přílohy

1 Seznam obrázků

Obr. 1 Street Mapper IV (http://www.3dlasermapping.com)	8
Obr. 2 Merlin (http://www.renishaw.com)	9
Obr. 3 ZEB CAM (http://www.3dlasermapping.com)	10
Obr. 4 Detekce a modelování stromů – <i>Rutzinger a kol. (2010)</i> , str.523	15
Obr. 5 Detekce dopravních značek – <i>Pu a kol. (2011)</i> , str.532	16
Obr. 6 Výsledek klasifikace terénních bodů (oranžové)	18
Obr. 7 Příklad využití poloautomatických funkcí ke klasifikaci projíždějících vozidel	19
Obr. 8 Lynx Mobile Mapper (skenování dálnice D11, 2009, foto Ing. Pazdera)	23
Obr. 9 Automatická generace hranic bloků podél trajektorie a jejich manuální úprava	24
Obr. 10 Chyba v georeferencování mračen - příčný řez mračnem, 8. km (rozdíl ve výšce 0,6m)	25
Obr. 11 Klasifikace terénu – Terrascan (nahore), Lastools (dole); příčný řez komunikací a přilehlým svahem (na spodním obrázku neklasifikován konec svahu – vlevo, klasifikována spodní část auta a svodidel - vpravo)	26
Obr. 12 Řez mračnem bodů	27
Obr. 13 Schéma navrhovaného postupu	28
Obr. 14 Schéma výpočtu rozdílu výšek pro maximální daný sklon	34
Obr. 15 Výsledek klasifikace na základě sklonu terénu, výšky bodů nad terénem a hustoty bodů; 31. km	35
Obr. 16 Obraz po aplikaci prvních filtrů (funkce <i>bwmorph</i> , operace <i>clean</i> , <i>bridge</i> , <i>majority</i>); 31. km	37
Obr. 17 Obraz po aplikaci funkce <i>bwpropfilt</i> – filtrace objektů na základě délky; 31. km	37
Obr. 18 Obraz po inverzi a aplikaci funkce <i>bwmorph</i> , operace <i>hbreak</i> a funkce <i>bwpropfilt</i> ; 31. km	38
Obr. 19 Výsledný obraz po finálním shlazení (funkce <i>bwmorph</i> , operace <i>spur</i> , funkce <i>imclose</i> a <i>imfill</i>); 31. km	38
Obr. 20 Svodidla – pixely vybrané na základě výšky nad terénem nad <i>roadway</i> a okolím; 26. km	40
Obr. 21 Pixely obsahující body 0,5m nad terénem a výš; 26. km	40
Obr. 22 Hrubý obraz vegetace v úrovni svodidel – pro odstranění; 26. km	41
Obr. 23 Objekty s maximálním rozptylem nad 40m; 26. km	42
Obr. 24 Objekty s maximálním rozptylem mezi 10-40m; 26. km	42
Obr. 25 <i>Crash barriers</i> – svodila; 26. km	43
Obr. 26 Svodidla – chybně klasifikované vozidlo; 72. km	44
Obr. 27 Třída <i>vehicles</i> před aplikací filtrů; 72. km	45
Obr. 28 Třída <i>vehicles</i> – 72. km	45
Obr. 29 Pixely obsahující body výše než 7,00m nad terénem; 18. km	47
Obr. 30 Elektrické vedení nad vozovkou; 18. km	47
Obr. 31 Mýtná brána; 31. km	49
Obr. 32 Sloupy mýtné brány; 31. km	49
Obr. 33 Ekodukt - třída <i>gates</i> ; 72. km	50

Obr. 34 Ekodukt - boky tunelu na pozadí <i>intensity image</i> ; 72. km.....	50
Obr. 35 Most; 62. km (b).....	51
Obr. 36 Konce mostu, sloupy; 62. km (b).....	51
Obr. 37 Histogram intenzitního obrazu; 24. km.....	53
Obr. 38 Histogram intenzitního obrazu - detail; 24. km.....	53
Obr. 39 <i>Signs</i> - primární obraz (vlevo), po aplikace podmínky minimálního počtu bodů na pixel (vpravo); 62. km.....	54
Obr. 40 Billboardy – primární obraz před filtrací; 24. km.....	56
Obr. 41 Odstranění vegetace; 24. km.....	56
Obr. 42 Billboardy - po aplikaci filtrů; 24. km.....	57
Obr. 43 Primární obraz třídy <i>walls</i> – ploty a vegetace; 72. km.....	58
Obr. 44 Body s výškou 0,50m nad terénem a výš, po aplikaci mediánového filtru; 72. km.....	58
Obr. 45 Vegetace k odstranění; 72. km.....	59
Obr. 46 Obraz plotů po odstranění větší vegetace; 72. km.....	60
Obr. 47 Třída <i>walls</i> , z toho červeně označeny části schované pod vegetací a přidané dodatečně; 72. km.....	60
Obr. 48 Třída <i>walls</i> (červeně), na pozadí intenzitní obraz; 72. km.....	61
Obr. 49 Zábradlí u mýtné brány, vlevo: primární obraz <i>walls</i> , vpravo: výsledná třída <i>walls</i> (červeně) na pozadí intenzitního obrazu; 31. km.....	61
Obr. 50 Body potenciálně příslušející k třídě <i>poles</i> ; 18. km.....	63
Obr. 51 Třída <i>poles</i> (červeně), na pozadí obraz bodů 3m nad terénem a výš po mediánovém filtru; 18. km.....	63
Obr. 52 Třída <i>poles</i> (červeně), na pozadí obraz intensity; vlevo nahoře - chybně klasifikovaný kmen stromu, vlevo dole - sloup elektrického vedení, vpravo – směrový sloupek; 18. km.....	63
Obr. 53 Vlevo - SOS hlásky (červeně), na pozadí intenzitního obrazu, vpravo - obraz bodů ve výšce 0,4-0,8m nad terénem; 72. km.....	64
Obr. 54 Body s výškou 0,5m nad terénem a výš; 72. km.....	65
Obr. 55 Třída <i>vegetation</i> ; 72. km.....	66
Obr. 56 Řez mračnem bodů - středový dělicí pás se svodidly, oranžové body - třída <i>ground</i>	67
Obr. 57 Graf normálního (Gausova) rozdělení, Autor: M. W. Toews – Vlastní dílo, based (in concept) on figure by Jeremy Kemp, on 2005-02-09, CC BY 2.5, https://commons.wikimedia.org/w/index.php?curid=1903871	68
Obr. 58 Klasifikace bodu - rozhodovací strom.....	69
Obr. 59 Porovnání automatické (vlevo) a manuální (vpravo) klasifikace; <i>unclass</i> – bílá, <i>ground</i> – oranžová, <i>crash barriers</i> – tyrkysová, <i>signs</i> – žlutá (blok pt09).....	74
Obr. 60 Porovnání automatické (vlevo) a manuální (vpravo) klasifikace; <i>unclass</i> – bílá, <i>ground</i> – oranžová, <i>crash barriers</i> – tyrkysová, <i>walls</i> – růžová, <i>vegetation</i> – zelená (blok pt09).....	74
Obr. 61 Kmen stromu přiřazený do třídy <i>poles</i> ; <i>unclass</i> – bílá, <i>ground</i> – oranžová, <i>vegetation</i> – zelená, <i>poles</i> - fialová (blok pt01).....	75
Obr. 62 Neklasifikována informativní tabulka; <i>unclass</i> – bílá, <i>ground</i> – oranžová (blok pt01).....	76

Obr. 63 Dopravní značka přiřazená do třídy <i>vehicles</i> ; <i>unclass</i> – bílá, <i>ground</i> – oranžová, <i>roadway</i> – šedá, <i>vehicles</i> – červená (blok pt02)	76
Obr. 64 SOS hláska přiřazená do třídy <i>vehicles</i> ; <i>unclass</i> – bílá, <i>ground</i> – oranžová, <i>roadway</i> – šedá, <i>vehicles</i> – červená, <i>crash barriers</i> - tyrkysová (blok pt07)	77
Obr. 65 Stěna s propustkem a částí komunikace přiřazená do tříd <i>signs</i> a <i>vegetation</i> ; <i>unclass</i> – bílá, <i>ground</i> – oranžová, <i>vegetation</i> – zelená, <i>signs</i> – žlutá (blok pt03)	78
Obr. 66 Svodidla komunikace souběžné s dálnicí přiřazená do třídy <i>walls</i> , <i>unclass</i> – bílá, <i>ground</i> – oranžová, <i>walls</i> – růžová (blok pt06)	78
Obr. 67 Sloup mýtné brány přiřazený do třídy <i>crash barriers</i> ; <i>roadway</i> – šedá, <i>ground</i> – oranžová, <i>crash barriers</i> – tyrkysová, <i>gates</i> – modrá, <i>walls</i> – růžová (blok pt09)	79
Obr. 68 Klasifikace mostu - třída <i>gates</i> ; <i>unclass</i> – bílá, <i>roadway</i> – šedá, <i>ground</i> – oranžová, <i>crash barriers</i> – tyrkysová, <i>gates</i> – modrá, <i>walls</i> – růžová, <i>vehicles</i> – červená (blok pt05)	79

2 Seznam tabulek

Tab. 1 Referenční GPS	22
Tab. 2 Formát seznamů souřadnic	30
Tab. 3 Rozdělení bodů na základě vzdálenosti od třídy <i>ground</i>	68
Tab. 4 Porovnání výsledků manuální a automatické klasifikace - počty bodů v tisících	71
Tab. 5 Porovnání výsledků manuální a automatické klasifikace - v % z manuálně klasifikovaných bodů	71
Tab. 6 Chyby I. a II. typu.....	72
Tab. 7 Úplnost a přesnost klasifikace.....	72
Tab. 8 Porovnání výsledků manuální a automatické klasifikace - počty objektů (pro třídy 4-5 a 7-8) ..	73
Tab. 9 Úplnost a přesnost klasifikace.....	80
Tab. 10 Přehled úspěšnosti klasifikačních algoritmů.....	81

3 Seznam příloh na CD

1. Disertační práce (PDF)
2. Zdrojové kódy
3. Příklady vstupních souborů

4 Kód “twoZcoord.m”

```
function [data_ZZ] = twoZcoord(data_1,data_2)
tic;
% Example of use:
% twoZcoord ('72km.xyz', '72km_gheight.txt');

% Unite 2 same pointclouds (1. Z, 2. Z over ground) into one
% output (Cl X Y Zoverg Int Z)

% PARAMETERS
% 1. (X Y Z)
inp_x1 = 1;
inp_y1 = 2;
inp_z1 = 3;

% 2. (Cl X Y Zoverground Int)
inp_cl = 1;
inp_x2 = 2;
inp_y2 = 3;
inp_z2 = 4;
inp_in = 5;

data1 = importdata(data_1);
data2 = importdata(data_2);

format longG

if size(data1,1) == size(data2,1)

    if data1(1,inp_x1) == data2(1,inp_x2)

        data_ZZ = [data2(:,inp_cl),data2(:,inp_x2), data2(:,inp_y2), data2(:,inp_z2),
        data2(:,inp_in), data1(:,inp_z1)];
        dlmwrite('data_zz.txt', data_ZZ,'delimiter','\t', 'precision',10);

    else
        disp('Error - points in input files are not in same order');
    end
else
    disp('Error - input files do not have same number of points');
end
toc;
end
```

5 Kód "pcloud2image.m"

```
clear all;
fclose('all');
clc;
tic;

%%% PCLLOUD2IMAGE %%%%
% TRANSFORM POINT CLOUD DATA INTO RASTER IMAGES AND PROVIDE CLASSIFICATION
% ON THAT IMAGES
version = '201709013';

% input data file
par.input = 'pt01_data_zz.txt';

% IMAGE CLASSIFICATION PARAMETERS
% resolution
par.res = 0.20; % planar[m]
par.Hres = 0.10; % height[m]
Z_limit = 8; % [m] set to reduce the size of image matrixs - due to MatLab
% array size limit, all points heigher than 8m over terrain are merged
% into one layer
par.dens_thres = 2; % threshold for density mask/roadway classification,
% min number of points per voxel
par.GP_thres = 0.05; % [m] threshold for roadway points - max height from ground points
par.slope = 7.5; % percent - for highways in CzR - max. slope 7.5%

par.buff1 = 3.0; % [m], size of buffer around roadway
par.buff2 = 7.0; % [m], size of buffer around roadway
par.buff3 = 2.0; % [m], size of buffer into roadway (due to emergency phones
% and traffic delineators)

par.hveg = 1; % [m] min height of vegetation
par.minCrown = 1; % [m] min diameter of crown higher than 1m over ground

% IMPORT
% Import data from ASCII file
% (Class X Y Z_over_ground Intensity Z)
dataIn = importdata(par.input);
sz = size (dataIn);

% Number format
format longG

% DATA LIMITS -> IMAGE MATRIX SIZE
% minx maxx,
% miny maxy,
% minz maxz
data_limits = [min(dataIn(:,2:4)), max(dataIn(:,2:4))];

if par.res <= 1 && par.Hres <= 1

    rasterlimits = round (data_limits,0);
    for r = 1:3
        pixsize = par.res;
        if r == 3
            pixsize = par.Hres;
        end
    end
end
```

```

while rasterlimits(r,1) > data_limits(r,1)
rasterlimits(r,1) = rasterlimits(r,1) - pixsize;
end

while rasterlimits(r,1) < data_limits(r,1) && abs(rasterlimits(r,1)-data_limits(r,1)) > pixsize
rasterlimits(r,1) = rasterlimits(r,1) + pixsize;
end

while rasterlimits(r,2) < data_limits(r,2)
rasterlimits(r,2) = rasterlimits(r,2) + pixsize;
end

while rasterlimits(r,2) > data_limits(r,2) && abs(rasterlimits(r,2)-data_limits(r,2)) > pixsize
rasterlimits(r,2) = rasterlimits(r,2) - pixsize;
end
end

elseif par.res > 1 || par.Hres > 1
disp('Error - set par.res or par.Hres lower than 1m');
end

% TRANSFORMATION
% Transformation matrixes - math system -> image system

origin = zeros (3,1,'double');

origin(1) = rasterlimits(1,1);
origin(2) = rasterlimits(2,2);
origin(3) = rasterlimits(3,1);

% Class, X, Y, Zg, Intensity, ones - for translation
% from dataIn(Class X Y Zg Intensity Z)
% data = [dataIn(:,1:5), ones(size(dataIn,1),1)];
data = [dataIn(:,1:5), ones(size(dataIn,1),1)];

Mt2Or = [[eye(5);zeros(1,5)], [0; -origin;0;1]];

Mt2Im = [
1 0 0 0 0 0
0 0 -1/par.res 0 0 1.5
0 1/par.res 0 0 0 1.5
0 0 0 1/par.Hres 0 0.5
0 0 0 0 1 0
0 0 0 0 0 1];

% Transformation - math system -> image system
dataIm = round(Mt2Im*(Mt2Or*data));

% dataImandOr - class, row(Y), collumn (X), layer (Z), Intensity,
% ones, Zg, Z
dataImandOr = [dataIm', data(:,4), dataIn(:,6)]';
DataIm_max = max(dataIm')+ [0 1 1 0 0 0];

% height limit for image matrix size
if DataIm_max(4) > (Z_limit/par.Hres)
DataIm_max(4) = Z_limit/par.Hres;
end

% IMAGE MATRIX

```

```

% 4 matrix dimensions
% 3 dimensions show position of the voxel - row(Y), column (X), layer (Z)
% 4th dimension include description of point in voxel - density (number of
% points in voxel, maxZg, maxInt, maxZ)
ImMatrix = zeros (DataIm_max(2),DataIm_max(3),DataIm_max(4), 4);
ImMSize = size (ImMatrix);

% Image Matrix Filling
for s = 1:size(dataIm,2)
    if dataIm(4,s) > Z_limit/par.Hres
        dataIm(4, s) = Z_limit/par.Hres;
    end
    % density (1)
    ImMatrix (dataIm(2,s),dataIm(3,s),dataIm(4,s),1) = ImMatrix (dataIm(2,s),dataIm(3,s),dataIm(4,s),1)
+ 1;

    % maxZg (2)
    if ImMatrix (dataIm(2,s),dataIm(3,s),dataIm(4,s),2) == 0 || ImMatrix
(dataIm(2,s),dataIm(3,s),dataIm(4,s),2) < dataImandOr(7,s)
        ImMatrix (dataIm(2,s),dataIm(3,s),dataIm(4,s),2) = dataImandOr(7,s);
    end

    % maxInt (3)
    if ImMatrix (dataIm(2,s),dataIm(3,s),dataIm(4,s),3) == 0 || ImMatrix
(dataIm(2,s),dataIm(3,s),dataIm(4,s),3) < dataImandOr(5,s)
        ImMatrix (dataIm(2,s),dataIm(3,s),dataIm(4,s),3) = dataImandOr(5,s);
    end

    % minZ (4)
    if ImMatrix (dataIm(2,s),dataIm(3,s),1,4) == 0 || ImMatrix (dataIm(2,s),dataIm(3,s),1,4) >
dataImandOr (8,s)
        ImMatrix (dataIm(2,s),dataIm(3,s),1,4) = dataImandOr(8,s);
    end
end

%%%%% MAX Zover_ground %%%%%
ImMatrix_Z_max = max (ImMatrix(:, :, 2), [], 3);

% Number of layers between origin and zero (height)
% - below zero, + over zero
zero_pos = floor(origin(3)/par.Hres);

%%%%%%%%%%%%% SUM OF DENSITY ALONG Z %%%%%%%%%%%%%%
% Crash Barriers - points 0.5-1.0m over terrain
% low limit - to avoid noise and low vegetation
% height limit - to avoid higher vegetation and vertical traffic signs
minz_crash = - zero_pos + ceil(0.5/par.Hres);
maxz_crash = - zero_pos + ceil(1/par.Hres);

if minz_crash > ImMSize(3)
    minz_crash = 0;
end

if maxz_crash > ImMSize(3)
    maxz_crash = ImMSize(3);
end

if minz_crash == 0
    points_crash = zeros (ImMSize(1:2));
elseif maxz_crash == minz_crash

```



```

    points_crash = ImMatrix(:, :, maxz_crash, 1);
else
    points_crash = sum (ImMatrix(:, :, minz_crash:maxz_crash, 1), 3);
end

%%% Long vehicles - over 3.0m and lower 4.5m
minz_veh = - zero_pos + ceil(3/par.Hres);
maxz_veh = - zero_pos + ceil(4.5/par.Hres);

if minz_veh > ImMSize(3)
    minz_veh = 0;
end

if maxz_veh > ImMSize(3)
    maxz_veh = ImMSize(3);
end

if minz_veh == 0
    points_veh = zeros (ImMSize(1:2));
elseif maxz_veh == minz_veh
    points_veh = ImMatrix(:, :, veh, 1);
else
    points_veh = sum (ImMatrix(:, :, minz_veh:maxz_veh, 1), 3);
end

% Traffic Signes - points 1.2-4.5m over terrain
minz_signs = - zero_pos + ceil(1.2/par.Hres);
maxz_signs = - zero_pos + ceil(4.5/par.Hres);

if minz_signs > ImMSize(3)
    minz_signs = 0;
end

if maxz_signs > ImMSize(3)
    maxz_signs = ImMSize(3);
end

if minz_signs == 0
    points_signs = zeros (ImMSize(1:2));
elseif maxz_signs == minz_signs
    points_signs = ImMatrix(:, :, maxz_signs, 1);
else
    points_signs = sum (ImMatrix(:, :, minz_signs:maxz_signs, 1), 3);
end

% Gates - points higher than 4.5m over terrain
minz_gates = - zero_pos + ceil(4.5/par.Hres);

if minz_gates > ImMSize(3)
    minz_gates = 0;
end

if minz_gates == 0
    points_gates = zeros (ImMSize(1:2));
elseif ImMSize(3) == minz_gates
    points_gates = ImMatrix(:, :, ImMSize(3), 1);
else
    points_gates = sum (ImMatrix(:, :, minz_gates:ImMSize(3), 1), 3);
end

```

```

% Complete density image - NO DATA MASK (INVERSE)
points_ground = sum (ImMatrix(:,:,1:(- zero_pos + 1),1), 3);
points_density = sum (ImMatrix(:,:,1), 3);
no_data_mask = imbinarize(-1*points_density, -0.5);

% Maximal Intensity Image
intensity_max = max(ImMatrix(:,:,3),[],3);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Image CLASSIFICATION part %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%% thresholding - maximal heigh of points over ground lower than 0.75m
% (to avoid gates and vegetation over crash barriers)
z_lower = - zero_pos + floor(0.75/par.Hres);
if z_lower > ImMSize(3)
    z_lower = ImMSize(3);
end
GP_image = max (ImMatrix(:,:,1:z_lower,2), [], 3);
GP_image = imbinarize(GP_image,par.GP_thres);

%%% thresholding - minimal points density
pts_dens_mask = imbinarize (-1*points_ground, -1*par.dens_thres);

%%% thresholding - maximal range between minimal Z in 3x3 neighborhood
GP_Z = ImMatrix(:,:,1,4);
range_limit = 2*realsqrt(2*par.res^2)*par.slope/100 + 0.02;
GP_range = imbinarize(rangefilt(GP_Z),range_limit);

%%% masks merging - based on max.Z over terrain + density
mask_con = imbinarize(GP_image + GP_range + pts_dens_mask, 0.5);

%%% mask filtering to reduce the noise and connect close objects
mask_con = bwmorph(bwmorph(bwmorph(mask_con,'clean'),'bridge'),'majority');

%%% clear mask from small objects (cars, central crash barriers..)
mask_con = bwpropfilt(mask_con,'MajorAxisLength',[min(ImMSize(1:2))/3, ImMSize(1)+ImMSize(2)]);

%%% invert mask to - pixels to keep
mask_inv = imbinarize (-1*(mask_con -1));

%%% disconnect small objects
mask_inv = bwmorph(mask_inv,'hbreak');

%%% clear mask from small objects - usually parts of low vegetation close to
mask_fin = bwpropfilt(mask_inv,'MajorAxisLength',[min(ImMSize(1:2))/3, ImMSize(1)+ImMSize(2)]);
mask_fin = bwmorph(mask_fin,'spur',10);

%%% invert mask to - pixels to throw away, connect close parts
mask_inv = bwmorph(bwmorph(bwmorph(imbinarize(-1*(mask_fin -1)), 'bridge'), 'diag'),'spur', 5);
roadway_for_crash = mask_inv;
%%% invert mask to - pixels to keep - deleting small disconnected parts and
%%% smoothing
mask_fin = bwpropfilt(imbinarize(-1*(mask_inv -1)), 'MajorAxisLength',[min(ImMSize(1:2))/3,
ImMSize(1)+ImMSize(2)]);
mask_fin = bwmorph (mask_fin, 'spur', 5);

```

```

se = strel('disk',ceil(2/par.res));
mask_fin = imclose(mask_fin, se);
roadway = imfill(mask_fin,'holes');
imwrite(roadway, 'roadway.tiff')

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% ROADWAY - BUFFER %%%%%%%%%

```

```

%%% buffer size - 3.00m (buff1), 6.00m (buff2) and buffer on both sides of roadway border (buff3)
num_it1 = ceil(par.buff1/par.res);
num_it2 = ceil((par.buff2-par.buff1)/par.res);
num_it3 = ceil((par.buff1+par.buff3)/par.res); %inverse
roadway_buff1 = double(roadway);

```

```

for i = 1:num_it1
roadway_buff1 = ordfilt2(roadway_buff1,7,ones(3,3));
end

```

```

roadway_buff2 = roadway_buff1;

```

```

for i = 1:num_it2
roadway_buff2 = ordfilt2(roadway_buff2,7,ones(3,3));
end
roadway_buff1 = imbinarize (roadway_buff1, 0.5);
roadway_buff2 = imbinarize (roadway_buff2, 0.5);
roadway_buff1 = imfill(roadway_buff1,'holes');
roadway_buff2 = imfill(roadway_buff2,'holes');

```

```

%%% morphological closing - for case of unclassified pixels of median strip

```

```

se = strel('disk',ceil(2/par.res));
roadway_buff1 = imclose(roadway_buff1, se);
roadway_buff2 = imclose(roadway_buff2, se);

```

```

roadway_buff3 = imbinarize(-1*roadway_buff1, -0.5);
for i = 1:num_it3
roadway_buff3 = ordfilt2(roadway_buff3,9,ones(3,3));
end

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CRASH BARRIERS %%%%%%%%%

```

```

%%% points in a specific height over terrain - on or close to roadway

```

```

crash_barr = imbinarize(points_crash, 1) + roadway_buff1;
crash_barr = imbinarize(crash_barr, 1.5);
crash_barr = imreconstruct(crash_barr, imbinarize(points_crash, 1));

```

```

%%% deleting of continuous strips of vegetation

```

```

vegetation = imbinarize(ImMatrix_Z_max, 0.5);
part_veg = ordfilt2(imreconstruct(imbinarize(vegetation - roadway_buff1, 0.5),
ordfilt2(imbinarize(points_crash, 0.5),1,ones(3,3))),9,ones(3,3));
crash_barr = imbinarize(crash_barr - part_veg, 0.5);

```

```

%%% deleting of long vehicles

```

```

crash_barr = crash_barr - imreconstruct(bwpropfilt(imbinarize(imbinarize(points_veh,0.5) +
roadway,1.5), 'MajorAxisLength', [2/par.res, ImMSize(1)*ImMSize(2)]), crash_barr);
crash_barr = imbinarize(crash_barr, 0.5);
crash_barr = bwmorph (crash_barr, 'bridge');

```

```

%%% object to skeleton

```

```

crash_barr_skel = bwmorph (crash_barr, 'skel', Inf);

%%% filtering of skeletons
crash_barr1 = bwpropfilt(crash_barr_skel, 'MajorAxisLength', [40/par.res, ImMSize(1)*ImMSize(2)]);
crash_barr2 = bwpropfilt(crash_barr_skel, 'MajorAxisLength', [5/par.res,40/par.res]);

crash_barr2 = bwpropfilt(crash_barr2, 'MinorAxisLength', [0 0.5/par.res]);
crash_barr_skel = imbinarize(crash_barr1 + crash_barr2, 0.5);

%%% selected skeleton is used as a seed to find matching object
crash_barr = imreconstruct(crash_barr_skel, crash_barr);
crash_barr = ordfilt2(crash_barr,9,ones(3,3));
imwrite (crash_barr, 'crash_barr.tiff');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%% VEHICLES %%%%%%%%%
% points over roadway (lower than gates) without crash barriers
% min z setting - 0.5m to avoid noise and grass close to roadway
minz_vehicles = - zero_pos + ceil(0.5/par.Hres);
vehicles = imbinarize (sum (ImMatrix(:, :, minz_vehicles:minz_gates, 1), 3), 1);
vehicles = imbinarize((vehicles + roadway - crash_barr), 1.5);
vehicles = ordfilt2(vehicles,9,ones(3,3));

% min limit for offround group of points to avoid elimination of pole-like
% objects
vehicles = bwpropfilt(vehicles, 'MajorAxisLength', [ceil(0.8/par.res)+4 ImMSize(1)+ImMSize(2)]);
imwrite (vehicles, 'vehicles.tiff');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%% GATES %%%%%%%%%
%%% subclass - POWERLINES %%%
%%% min 7m over ground (over highways)
minz_powerlines = - zero_pos + ceil(7/par.Hres);
if minz_powerlines <= ImMSize(3)

    powerlines_0 = imbinarize (sum (ImMatrix(:, :, minz_powerlines:ImMSize(3), 1), 3), 0.5);
    powerlines = imbinarize(powerlines_0 + roadway_buff2, 1.5);
    powerlines = imbinarize ((powerlines - medfilt2(powerlines_0)), 0.5);

    %%% min length - 2m
    powerlines = ordfilt2(powerlines,9,ones(3,3));
    powerlines = bwmorph(powerlines, 'bridge');
    powerlines = bwpropfilt(powerlines, 'MajorAxisLength', [2/par.res (ImMSize(1)+ImMSize(2))]);
    %%% max. eccentricity, 0,942 corespond to b=1/3a
    powerlines = bwpropfilt(powerlines, 'Eccentricity', [0.942 1]);
    powerlines = imreconstruct (powerlines, powerlines_0);
    powerlines = bwpropfilt(powerlines, 'MinorAxisLength', [0 1/par.res]);

else
    powerlines = zeros (ImMSize(1),ImMSize(2));
end

%%% subclass - toll-gates and traffic signs over roadway
%%% points higher than 4.5m over roadway
gates_points = imbinarize(points_gates, 0.5);

```

```

gates0 = imbinarize((gates_points + roadway - powerlines), 1.5);
gates = bwpropfilt(gates0, 'MajorAxisLength',[2/par.res 100/par.res]);
gates = imreconstruct(gates, gates_points);

%%% orientation filter - delete vegetation parts
prop = regionprops(bwareafilt(roadway, 1),'orientation');
if abs(prop.Orientation) < 10
gates1 = bwpropfilt(gates, 'Orientation',[-90 -70]);
gates2 = bwpropfilt(gates, 'Orientation',[70 90]);
gates = imbinarize(gates1 + gates2, 0.5);

else
if prop.Orientation > 0
gates = bwpropfilt(gates, 'Orientation',[prop.Orientation-100 prop.Orientation-80]);

else
gates = bwpropfilt(gates, 'Orientation',[prop.Orientation+80 prop.Orientation+100]);
end
end

%%% subclass - BRIDGES, TUNNELS %%%
%%% roadway width
roadway_max = bwareafilt(roadway,1);
roadway_width = regionprops(roadway_max, 'Minoraxislength');

bridges = bwpropfilt(gates0, 'MajorAxisLength',[roadway_width.MinorAxisLength/2
(ImMSize(1)+ImMSize(2))]);
bridges = imreconstruct(bridges, gates_points);

%%% merging
gates = imbinarize ((gates + bridges), 0.5);

powerlines = imbinarize (powerlines - gates, 0.5);
powerlines = ordfilt2(powerlines,9,ones(3,3));
% imwrite (powerlines, 'powerlines.tiff');

% extrapolation of gates/bridges to lower parts
gate_low = - zero_pos + floor(1/par.Hres);
gate_high = - zero_pos + floor(4.5/par.Hres);
if gate_high > ImMSize (3)
gate_high = ImMSize (3);
end
gate_end = imbinarize(sum(ImMatrix(:, :, gate_low:gate_high, 1), 3), 0.5);
gate_end = imbinarize ((gate_end - vehicles - crash_barr), 0.5);
gate_end = medfilt2 (gate_end);
gate_end = bwpropfilt(gate_end, 'MajorAxisLength', [1/par.res (ImMSize(1)+ImMSize(2))]);
gate_end = imreconstruct(gates, gate_end);
gate_end = ordfilt2(gate_end,9,ones(3,3));
imwrite (gate_end, 'gate_end.tiff');

%%% gates and lower parts
gates = ordfilt2(gates,9,ones(3,3));
gates = imbinarize((gates + gate_end), 0.5);

%%% columns of gates
%%% points with height 2.0-4.5m
gate_low = - zero_pos + floor(2.0/par.Hres);
gate_high = - zero_pos + floor(4.5/par.Hres);
if gate_high > ImMSize (3)

```

```

gate_high = ImMSize (3);
end
gate_column = imbinarize(sum(ImMatrix(:, :, gate_low:gate_high, 1), 3), 1);
gate_column = imbinarize ((gate_column - bwpropfilt(vehicles, 'MajorAxisLength', [6/par.res
200/par.res]) - crash_barr), 0.5);
gate_column_0 = gate_column;
gate_column = imreconstruct(gates, gate_column);
gate_column = ordfilt2(gate_column, 9, ones(3, 3));
imwrite (gate_column, 'gate_column.tiff');

gates = imbinarize ((gates + powerlines), 0.5);
imwrite (gates, 'gates.tiff');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% VERTICAL TRAFFIC SIGNS %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% intensity over threshold in set height without vehicles and crash_barr
intensity_signs = max(ImMatrix(:, :, minz_signs : maxz_signs, 3), [], 3);
intensity_signs0 = imbinarize(mat2gray(intensity_signs), 0.5);
intensity_billb0 = imbinarize(mat2gray(intensity_signs), 0.1);
intensity_signs0 = imbinarize((intensity_signs0 - vehicles - crash_barr - gates), 0.5);
intensity_billb0 = imbinarize((intensity_billb0 - vehicles - crash_barr - gates), 0.5);

%%% min 10 points/pixel in sign height - delete noise near signs, partly
%%% vegetation
intensity_signs = imbinarize(intensity_signs0 + imbinarize (sum (ImMatrix(:, :, minz_signs :
maxz_signs, 1), 3), 10), 1.5);
intensity_billb = imbinarize(intensity_billb0 + imbinarize (sum (ImMatrix(:, :, minz_signs : maxz_signs, 1),
3), 10), 1.5);

%%% SIGNS
%%% only objects near to roadway
ver_signs = imreconstruct (roadway_buff2, intensity_signs);
ver_signs = bwpropfilt(ver_signs, 'MajorAxisLength', [0.6/par.res 10/par.res]);

%%% eccentricity 0,745 corespond to b=2/3a
ver_signs = bwpropfilt(ver_signs, 'Eccentricity', [0.745 1]);

%%% BILLBOARDS
%%% delete big part of vegetation (pixels of high intensity hidden
%%% in the vegetation strips), disconnect billboards from vegetation
part_veg = bwpropfilt(imbinarize(vegetation - crash_barr - intensity_billb0, 0.5), 'MajorAxisLength',
[5/par.res 300/par.res]);
part_veg = ordfilt2(bwmorph(imfill (part_veg, 'holes'), 'spur'), 1, ones(3, 3));

intensity_billb = imbinarize (intensity_billb - imreconstruct(part_veg, intensity_billb), 0.5);
intensity_billb = bwmorph(intensity_billb, 'bridge');

%%% min. size and eccentricity
intensity_billb = bwpropfilt (intensity_billb, 'MajorAxisLength', [3/par.res 30/par.res]);
intensity_billb = bwpropfilt(intensity_billb, 'Eccentricity', [0.745 1]);

%%% combination SIGNS and BILLBOARDS
ver_signs = imbinarize (ver_signs + intensity_billb, 0.5);
ver_signs = ordfilt2(ver_signs, 9, ones(3, 3));
imwrite (ver_signs, 'ver_signs.tiff');

```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% WALLS AND BUILDINGS %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
% outside roadway, height 0.8-2.0m over terrain
wall_low = - zero_pos + floor(0.8/par.Hres);
wall_high = - zero_pos + floor(2/par.Hres);
if wall_high > ImMSize (3)
    wall_high = ImMSize (3);
end
wall_0 = imbinarize(sum(ImMatrix(:, :, wall_low:wall_high, 1), 3), 1);
wall_0 = wall_0 - roadway - ver_signs - gates - gate_column - crash_barr;
wall_0 = imbinarize(wall_0, 0.5);
```

```
% vegetation to erase
vegetation_0 = imbinarize (ImMatrix_Z_max, 0.5);
vegetation_0 = medfilt2(vegetation_0);
vegetation_0 = ordfilt2(ordfilt2(vegetation_0, 1, ones(3,3)), 9, ones(3,3));
```

```
%%% erase vegetation without connected part of walls
walls = imbinarize ((wall_0 - vegetation_0), 0.5);
%%% small parts of vegetation - min length of object
walls = ordfilt2(ordfilt2(walls, 9, ones(3,3)), 1, ones(3,3));
walls = bwpropfilt(walls, 'MajorAxisLength', [2/par.res (ImMSize(1)+ImMSize(2))]);
```

```
%%% assign deleted parts from wall_0
wall_add = bwpropfilt(imbinarize((imreconstruct(walls, wall_0) - walls), 0.5), 'MinorAxisLength', [0 1/par.res]);
```

```
%%% merge
walls = imbinarize ((walls + wall_add), 0.5);
imwrite (walls, 'walls.tiff');
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% POLES %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%%% high poles - power line poles
vegetation_high = medfilt2(imbinarize (ImMatrix_Z_max, 3.0));
```

```
if powerlines == zeros (ImMSize(1), ImMSize(2))
    pole_big = powerlines;
else
    pol_pos_low = - zero_pos + floor(3/par.Hres);
    pol_pos_high = - zero_pos + floor(5/par.Hres);

    pole_big = imbinarize(sum(ImMatrix(:, :, pol_pos_low:pol_pos_high, 1), 3), 10);

    pole_big = bwpropfilt(pole_big, 'MajorAxisLength', [0 1/par.res]);

    pole_big = imreconstruct (bwpropfilt(vegetation_high, 'MinorAxisLength', [2 2/par.res]), pole_big);
end
```

```
%%% traffic delineator and emergency telephones
pol_pos_low = - zero_pos + floor(0.4/par.Hres);
pol_pos_med = - zero_pos + floor(0.6/par.Hres);
pol_pos_high = - zero_pos + floor(0.8/par.Hres);

pole_1 = imbinarize(ImMatrix(:, :, pol_pos_low, 1), 1);
pole_2 = imbinarize(ImMatrix(:, :, pol_pos_med, 1), 1);
pole_3 = imbinarize(ImMatrix(:, :, pol_pos_high, 1), 1);
```

```

poles = pole_1 + pole_2 + pole_3;
poles = imbinarize (poles, 2.5);

%%% larger objects to erase
vegetation_low = imbinarize(sum(ImMatrix(:, :, pol_pos_low:pol_pos_high, 1), 3), 0.5);
vegetation_low = bwpropfilt (vegetation_low, 'MajorAxisLength', [1/par.res (ImMSize(1)+ ImMSize(2))]);

%%% limit position and erase other objects
poles = poles + roadway_buff3 + roadway_buff2 - vehicles - gates - crash_barr - ver_signs -
vegetation_low;
poles = imbinarize (poles, 2.5);

%%% merging poles
poles = imbinarize (poles + pole_big, 0.5);
poles = ordfilt2(poles, 9, ones(3,3));
imwrite (poles, 'poles.tiff');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% VEGETATION %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%% vegetation - higher than 1.0m,
%%% crown diameter - at least 1.0m in 0.5m over terrain or higher
%%% medfilt2 - delete rest of linear objects and very sparse vegetation
vegetation = imbinarize (ImMatrix_Z_max, par.hveg);
vegetation_0 = imbinarize (ImMatrix_Z_max, 0.5);
vegetation = imreconstruct(vegetation, medfilt2(vegetation_0));

%%% erase other objects
vegetation = vegetation - roadway - vehicles - ver_signs - gates - crash_barr - walls - poles;
vegetation = imbinarize(vegetation, 0.5);

%%% delete small objects
vegetation = bwpropfilt(vegetation, 'MinorAxisLength', [(par.minCrown/par.res), (ImMSize(1)+
ImMSize(2))]);

%%% buffer
vegetation = ordfilt2(vegetation, 9, ones(3,3));
imwrite(vegetation, 'vegetation.tiff');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% export image coordinates for image2pcloud
%%% use when image classification results are satisfying
dlmwrite('dataIm.txt', (dataIm(2:4,:)), 'delimiter', '\t')
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

toc;

```


6 Kód "image2pcloud.m"

```
%%% License: Attribution – Non-Commercial – ShareAlike: CC BY-NC-SA
%%% This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0
%%% International License.
%%% https://creativecommons.org/licenses/by-nc-sa/4.0/
```

```
clear all;
fclose('all');
clc;
tic;
format longG

version = '20170913';

% input data file and res
par.input1 = 'pt01_data_zz.txt';
par.input2 = 'datalm.txt';

% Import data from ASCII file (organized - Class X Y Zg Intensity Z)
dataln = importdata(par.input1);
% Import data from ASCII file (X Y Z) / image coordinates
datalm = importdata(par.input2);
sz = size (dataln);

par.dHG = 0.025; %[m](limit dH over Ground for ground class)

%%% CLASS CODES %%%
cl_unclass = 1;
cl_ground = 2;
cl_roadway = 3;
cl_vehicles = 4;
cl_gates = 5;
cl_crashbar = 6;
cl_poles = 7;
cl_signs = 8;
cl_walls = 9;
cl_veget = 10;

if size(dataln) ~= size(datalm)

    disp('Error - input files do not have same number of points');
else

roadway = imread('roadway.tiff');
gates = imread('gates.tiff');
gate_column = imread('gate_column.tiff');
gate_end = imread('gate_end.tiff');
crash_barr = imread('crash_barr.tiff');
ver_signs = imread('ver_signs.tiff');
vehicles = imread('vehicles.tiff');
poles = imread('poles.tiff');
walls = imread('walls.tiff');
vegetation = imread('vegetation.tiff');

for i=1:size(dataln)
    %low points - ground or roadway
    if dataln(i,4) <= par.dHG
```

```

if roadway (dataIm(i,1),dataIm(i,2)) == 0
    dataIn (i,1) = cl_ground;
else
    dataIn (i,1) = cl_roadway;
end
% higher points
else
if gates (dataIm(i,1),dataIm(i,2)) == 1 && dataIn(i,4) >= 4.5
    dataIn (i,1) = cl_gates;
elseif gate_column (dataIm(i,1),dataIm(i,2)) == 1
    dataIn (i,1) = cl_gates;
elseif gate_end (dataIm(i,1),dataIm(i,2)) == 1
    dataIn (i,1) = cl_gates;
elseif ver_signs (dataIm(i,1),dataIm(i,2)) == 1
    dataIn (i,1) = cl_signs;
elseif poles (dataIm(i,1),dataIm(i,2)) == 1
    dataIn (i,1) = cl_poles;
elseif crash_barr (dataIm(i,1),dataIm(i,2)) == 1
    dataIn (i,1) = cl_crashbar;
elseif walls (dataIm(i,1),dataIm(i,2)) == 1
    dataIn (i,1) = cl_walls;
elseif vegetation (dataIm(i,1),dataIm(i,2)) == 1
    dataIn (i,1) = cl_veget;
elseif vehicles (dataIm(i,1),dataIm(i,2)) == 1
    dataIn (i,1) = cl_vehicles;
else
    dataIn (i,1) = cl_unclass;
end
end
end
end

%%%%%%%%%%
%%% export classified point cloud
%%% Cl X Y Z Int
data_class = [dataIn(:,1:3), dataIn(:,6), dataIn(:,5)];
dlmwrite('data_class.txt', data_class, 'delimiter','\t', 'precision',10);
%%%%%%%%%%

toc;

```

7 Kód "ClassQualCheck.m"

```
%%% License: Attribution – Non-Commercial – ShareAlike: CC BY-NC-SA
%%% This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0
%%% International License.
%%% https://creativecommons.org/licenses/by-nc-sa/4.0/
```

```
function [Qual] = ClassQualCheck(autoclass,reference)
tic;

% ClassQualCheck('pt01_data_class.txt', 'pt01_man_clas.txt');
% Input files - *.txt (Cl X Y Z Int)

%%% CLASS CODES %%%
% cl_unclass = 1;
% cl_ground = 2;
% cl_roadway = 3;
% cl_vehicles = 4;
% cl_gates = 5;
% cl_crashbar = 6;
% cl_poles = 7;
% cl_signs = 8;
% cl_walls = 9;
% cl_veget = 10;

autoclass = importdata(autoclass);
reference = importdata(reference);

% code for missclassified points - higher code + 1
class_num = max(max(autoclass(:,1), reference(:,1)));
class_wrong = class_num + 1;

class_table = zeros (class_num, class_num);

if size(autoclass) ~= size(reference)
    disp('Error - input files do not have same number of points');
elseif autoclass(1,2:4) ~= reference(1,2:4)
    disp('Error - points in input files are not in same order');
else
    for i=1:size(autoclass)
        % fill in the statistic table
        class_table (autoclass(i,1),reference(i,1)) = class_table (autoclass(i,1),reference(i,1)) + 1;
        % label wrong classified points
        if autoclass(i,1) ~= reference(i,1)
            autoclass (i,1) = class_wrong;
        end
    end
end

sum_class = sum(class_table);
class_percent = class_table;

for j = 1:class_num
    class_percent (:,j) = (100*class_table(:,j))/sum_class(j);
end

class_codes = 1:1:class_num;
class_table = [[0; class_codes], [class_codes; class_table]];
class_percent = [[0; class_codes], [class_codes; class_percent]];
```

```

% Safe classification statistic table as a file
fid = fopen('Class_statistic.txt', 'w');

fprintf(fid, 'Classification Quality');
fprintf(fid, '\n');
fprintf(fid, '\n');
fprintf(fid, 'Rows - automatic classification');
fprintf(fid, '\n');
fprintf(fid, 'Columns - reference manual classification');
fprintf(fid, '\n');
fprintf(fid, 'First row and column - class code');
fprintf(fid, '\n');

fclose(fid);

dlmwrite('Class_statistic.txt', class_table, 'delimiter', '\t', 'precision', '%7.0f', '-append')
dlmwrite('Class_statistic.txt', '\n', '-append')

fid = fopen('Class_statistic.txt', 'a');
fprintf(fid, '\n');
fprintf(fid, 'Classification Quality in percent');
fprintf(fid, '\n');
fclose(fid);

dlmwrite('Class_statistic.txt', class_percent, 'delimiter', '\t', 'precision', '%.2f', '-append')

%%% export classified point with assigned wrong classified points
%%% CI X Y Z Int
%dlmwrite('data_class_check.txt', auticlass, 'delimiter', '\t', 'precision', 10);

toc;
end

```