Scene Parsing using Multiple Modalities

Sarah Taghavi Namin

A thesis submitted for the degree of
Doctor of Philosophy
The Australian National University

April 2017

Declaration

I hereby declare that this thesis is my original work which has been done in collaboration with other researchers. This document has not been submitted for any other degree or award in any other university or educational institution. Parts of this thesis have been published in collaboration with other researchers in international conferences as listed below:

- **(Chapter 3)** S. Taghavi Namin, L. Petersson, *Classification of materials in natural scenes using multi-spectral images*, IROS 2012.
- **(Chapter 4)** S. Taghavi Namin, M. Najafi, L. Petersson, *Multiview terrain classification using panoramic imagery and Lidar*, IROS 2014.
- **(Chapter 5)** S. Taghavi Namin, M. Najafi, M. Salzmann, L. Petersson, *A Multimodal Graphical Model for Scene Analysis*, WACV 2015.
- **(Chapter 6)** S. Taghavi Namin, M. Najafi, M. Salzmann, L. Petersson, *Cutting Edge: Soft Correspondences in Multimodal Scene Parsing*, ICCV 2015.

Furthermore, I have contributed to the following works, which are related to my PhD topic, though are not reported as the main contributions in this thesis:

- M. Najafi, S. Taghavi Namin, L. Petersson, *Classification of Natural Scene Multispectral Images using a New Enhanced CRF*, IROS 2013.
- M. Najafi, S. Taghavi Namin, M. Salzmann, L. Petersson, *Nonassociative Higher-order Markov Networks for Point Cloud Classification*, ECCV 2014.
- M. Najafi, S. Taghavi Namin, M. Salzmann, L. Petersson, *Sample and Filter: Nonparametric Scene Parsing via Efficient Filtering*, CVPR 2016.

Sarah Taghavi Namin
26 April 2017

To my loving husband
and my beloved parents

Acknowledgments

First of all, I would like to express my sincere gratitude to my supervisors, Dr. Lars Petersson and Dr. Mathieu Salzmann for their incredible support, constructive guidance and continuous encouragement during my candidature at ANU. Without their bright ideas and the efforts they put in, the smooth completion of my PhD could not have been possible. I am so proud to had the opportunity to work under their supervision. I am also grateful to my advisors Prof. Richard Hartley and Dr. Stephen Gould for their kind support and advices at different stages of my PhD.

Many thanks to my friends at NICTA and the Australian National University for creating a friendly and warm working environment. I would especially like to thank my dear friend, Mohammad Najafi, for all the fruitful discussions, his kind support and companionship during our four years of collaboration. He made my PhD so pleasant and memorable. I am also indebted to all my friends at Canberra, especially Hajar Sadeghi, for making the past years so enjoyable.

I would like to thank my various sources of financial support. Firstly, I would like to acknowledge NICTA/Data61 and the Australian National University for providing my PhD scholarship. Moreover, I would like to thank my supervisor, the School of Engineering and NICTA/Data61 for their financial support, which allowed me to attend several conferences.

At last, but at most, I wish to express my deepest gratitude to my husband, Mohammad Esmaeilzadeh, my parents and my brothers for their unconditional love and selfless dedication. This thesis would not be possible without their continuous support and encouragement and I would like to devote all my research achievements to them.

Abstract

Scene parsing is the task of assigning a semantic class label to the elements of a scene. It has many applications in autonomous systems when we need to understand the visual data captured from our environment. Different sensing modalities, such as RGB cameras, multi-spectral cameras and Lidar sensors, can be beneficial when pursuing this goal. Scene analysis using multiple modalities aims at leveraging complementary information captured by multiple sensing modalities. When multiple modalities are used together, the strength of each modality can combat the weaknesses of other modalities. Therefore, working with multiple modalities enables us to use powerful tools for scene analysis. However, possible gains of using multiple modalities come with new challenges such as dealing with misalignments between different modalities. In this thesis, our aim is to take advantage of multiple modalities to improve outdoor scene parsing and address the associated challenges. We initially investigate the potential of multi-spectral imaging for outdoor scene analysis. Our approach is to combine the discriminative strength of the multi-spectral signature in each pixel and the corresponding nature of the surrounding texture. Many materials appearing similar if viewed by a common RGB camera, will show discriminating properties if viewed by a camera capturing a greater number of separated wavelengths. When using imagery data for scene parsing, a number of challenges stem from, e.g., color saturation, shadow and occlusion. To address such challenges, we focus on scene parsing using multiple modalities, panoramic RGB images and 3D Lidar data in particular, and propose a multi-view approach to select the best 2D view that describes each element in the 3D point cloud data. Keeping our focus on using multiple modalities, we then introduce a multi-modal graphical model to address the problems of scene parsing using 2D-3D data exhibiting extensive many-to-one correspondences. Existing methods often impose a hard correspondence between the 2D and 3D data, where the 2D and 3D corresponding regions are forced to receive identical labels. This results in performance degradation due to misalignments, 3D-2D projection errors and occlusions. We address this issue by defining a graph over the entire set of data that models soft correspondences between the two modalities. This graph encourages each region in a modality to leverage the information from its corresponding regions in the other modality to better estimate its class label. Finally, we introduce latent nodes to explicitly model inconsistencies between the modalities. The latent nodes allow us

not only to leverage information from various domains in order to improve the labeling of the modalities, but also to cut the edges between inconsistent regions. To eliminate the need for hand tuning the parameters of our model, we propose to learn potential functions from training data. In addition, to demonstrate the benefits of the proposed approaches on publicly available multi-modality datasets, we introduce a new multi-modal dataset of panoramic images and 3D point cloud data captured from outdoor scenes (NICTA/2D3D Dataset).

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Scene parsing (also known as semantic labeling) consists of assigning a class label to each element of a scene. Labeling our environment (Figure 1.1) is useful when we need to understand the surrounding world in autonomous systems such as robot applications, that, e.g., can help negotiating the environment and assist blind people. Moreover, it is very helpful for other applications like intelligent vehicles (autonomous driving), automatic map generation, defect detection by capturing data periodically, and vegetation management by monitoring their growth. One of the most important components of a scene parsing system is the input data that provides information about our environment and can be obtained using various sensors. The most common sensor in this area of research is RGB cameras that provide color and texture information of the scene. This information is useful in classifying different objects, for example a green region with a specific texture may be classified as vegetation. Other data sensors, such as Lidar to generate 3D point cloud data, multi-spectral and thermal imaging present more information such as shape and temperature about scenes and objects.

The scene parsing task can in general be very challenging due to a number of issues including shadows and data saturation that are often seen in 2D outdoor images, occlusion and variable weather conditions. Even in ideal conditions, scene parsing is still a very complicated task since we face very complicated scenes. While each of the above-mentioned sensors (modalities) can alleviate some limitations of scene parsing to some extent, using multiple modalities concurrently can be more helpful. For example, if Lidar data and panoramic images (360° images) are used together, they can cover weaknesses of each other, and as a result improve scene parsing.

Scene analysis using multiple modalities aims at leveraging complementary information captured by multiple sensing modalities, such as 3D Lidar and 2D imagery (RGB, multi-spectral and thermal). 3D data provide information about structure, shape, size and the real

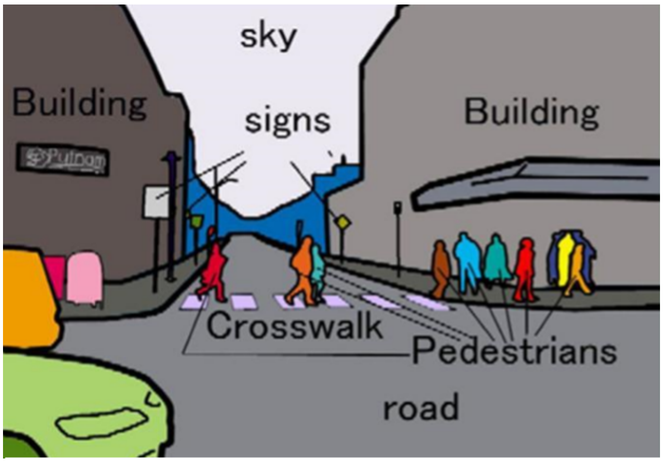


Figure 1.1: Automatic outdoor scene labeling has many applications, such as in robotics, autonomous driving and automatic map generation [87].

distance between objects that are valuable clues for scene understanding. RGB imaging, as mentioned above, is a well-established data capturing tool in scene analysis and is a proper source of visible information of the objects. A multi-spectral camera, which captures multiple wavelength bands in the visible and infrared ranges (compared to RGB), provides the spectral signature of each material that is beneficial when performing material distinction. Existing multi-modal scene parsing approaches consider data from multiple modalities to label the scene. However, they often suffer from an important limitation: they typically assume that corresponding regions in two modalities always have the same label. This assumption is encoded either explicitly by having a single label variable for all modalities [80, 29, 19], or implicitly by penalizing label differences between the domains [118, 72]. While this assumption may seem reasonable, it is often violated in realistic scenarios. Indeed, in practice, the different modalities are typically not perfectly aligned/registered. Figure 2.10 shows three examples of misalignment between 2D and 3D data. Furthermore, in dynamic scenes, moving objects may not be easily captured by some devices, such as 3D Lidar, due to their low acquisition speed. Note that A Lidar system captures 3D data continuously using a rotating sensor, unlike snapshot sensors where the image data are captured instantaneously. To give a concrete example, in the NICTA/2D3D dataset employed in our experiments, 17% of the connections between the two modalities correspond to inconsistent labels. The connections are found by projecting 3D points to 2D images. As a consequence, existing methods will typically produce wrong labels in at least one modality, since they fail to model these inconsistencies. So one of our challenges is to find a proper approach to handle such issues for multi-modal scene analysis.

In a nutshell, the goal of this thesis is to investigate the use of multiple sensing modalities to combine their information. Also we want to address some of the challenges we identified in this regard such as dealing with the misalignments between different modalities.

1.2 Contributions

The contributions of this thesis are on improving scene parsing by developing methods applicable to using multiple sensing modalities. In particular, we make the following contributions:

1.2.1 Multi-spectral Imaging for Material Classification in Scene Analysis

We investigate the potential of multi-spectral imaging for outdoor scene analysis. We propose a method suitable to distinguish between different materials occurring in natural scenes using a multi-spectral camera. Such a capability is useful in autonomous robot applications as well as in applications intended to create large scale inventories of assets in the proximity of roads. The utilized sensor records a seven band multi-spectral image, of which six bands are in the visible range and one in the near infrared (NIR) range. Figure 1.2 shows a sample image of multi-spectral data from our dataset. Many materials appearing similar if viewed by a common RGB camera, will show discriminating properties if viewed by a camera capturing a greater number of separated wavelengths. Our approach consists of combining the discriminating strength of the multi-spectral signature in each pixel and the corresponding nature of the surrounding texture. Texture features are exploited to make the system more robust to different lighting conditions.

1.2.2 Multi-view Terrain Classification using Panoramic Imagery and Lidar

Following our work on multi-spectral imaging, to benefit from using multiple modalities, we focus on addressing the challenges of performing object recognition in real world scenes captured by a commercial surveying vehicle equipped with a 360° panoramic camera in conjunction with a 3D laser scanner. Figure 1.3 shows a sample of point cloud data with the corresponding panoramic image that are captured from one scene. Even with state-of-the-art surveying equipment, there are color saturation and very dark regions in images, as well as some degree of time-varying misalignment between the point cloud data and 2D imagery as we discussed in Section 1.1. Moreover, there are frequent occlusions due to both static and moving objects. These issues are inherently difficult to avoid and therefore need to be dealt

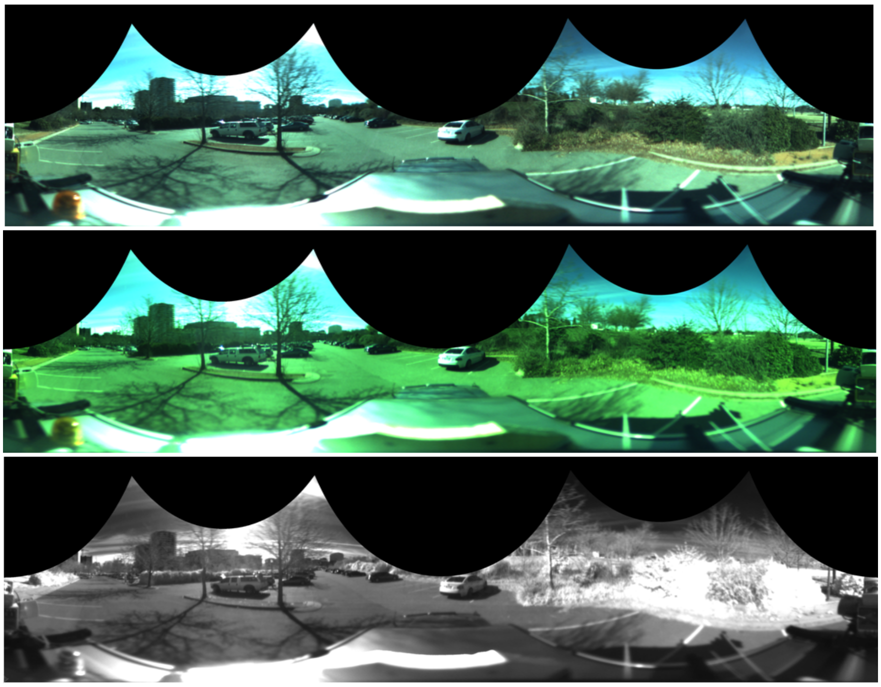


Figure 1.2: Sample multi-spectral data covered 7 wavelength bands (RGB, shifted RGB and NIR).

with in a more robust fashion. This is where the contribution of our work is; that is, the development of a consensus method that can intelligently incorporate feature responses from multiple 2D views and reject those that are not very descriptive. The 3D point cloud data are then labeled using their local information as well as the information of their corresponding 2D view. Subsequently, a conditional random field (CRF) which is equipped with the probabilities of the adjacent points and confusion matrix from local classifier, is applied to the system. The experiments are performed on a challenging dataset captured both in summer and winter.

1.2.3 A Multi-modal Graphical Model for Scene Analysis

To improve our previous work and provide the possibility of labeling both 2D and 3D domains simultaneously using other domain information, we introduce a multi-modal graphical model using 2D-3D data exhibiting extensive many-to-one correspondences. Existing methods often force corresponding regions in different modalities to receive identical labels. This results in performance degradation due to misalignments, 3D-2D projection errors and occlusions. We address this issue by defining a graph over the entire set of data that models soft correspondences between the two modalities. This graph encourages each region in a modality to

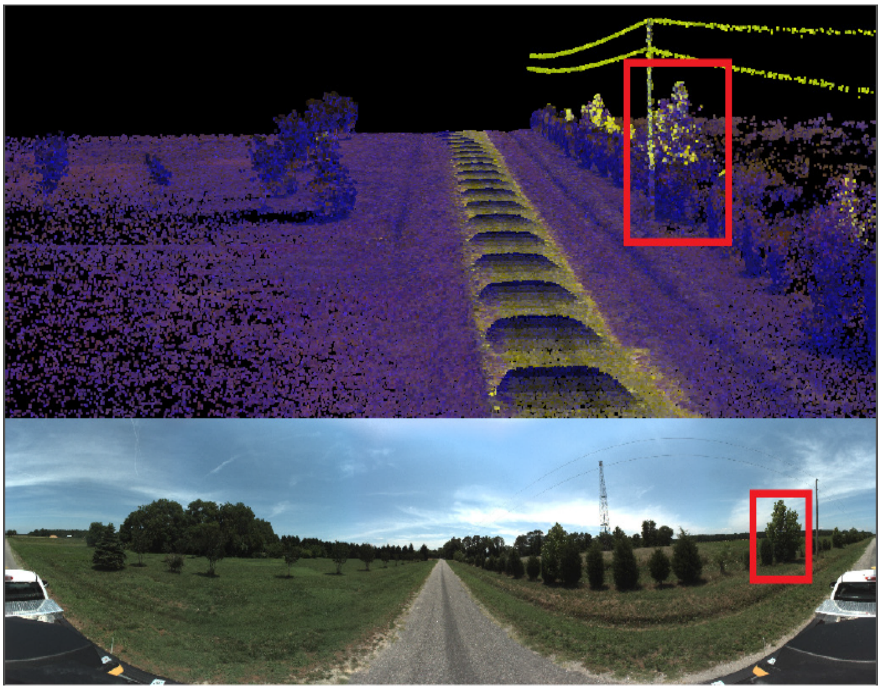


Figure 1.3: Sample point cloud data with corresponding panoramic image that cover same area.

leverage the information from its corresponding regions in the other modality to better estimate its class label. We evaluate our method on a publicly available dataset. Additionally, to demonstrate the ability of our model to support multiple correspondences for objects in 3D and 2D domains, we introduce a new multi-modal dataset. This dataset consists of panoramic images and 3D point cloud data captured from outdoor scenes (NICTA/2D3D Dataset). The data includes the entire set of 3D points which provides naturally occurring many-to-one relationships. That is, each 3D point is seen from a number of 2D images. The images have both a large vertical and horizontal Field of View (FOV) of the associated point cloud data, providing an opportunity to establish correspondences between 3D points and imagery from a large number of view points. We have made this dataset publicly available [1]. This enables research on methods necessary to resolve issues with ambiguity, occlusions that are spurious or due to parallax, and missing 2D-3D correspondences.

1.2.4 Soft Correspondences in Multi-modal Scene Parsing

We improved our multi-modal graphical model to better address the problems of data misalignment and label inconsistencies in semantic labeling by introducing latent nodes to explic-

[1] Publicly available at http://www.nicta.com.au/computer_vision_datasets.

itly model inconsistencies between two modalities. These latent nodes allow us not only to leverage information from both domains to improve their labeling, but also to cut the edges between inconsistent regions. To eliminate the need for hand tuning the parameters of our model, we propose to learn intra-domain and inter-domain potential functions from training data. We demonstrate the benefits of our approach on CMU/VMR dataset and NICTA/2D3D dataset containing 2D imagery and 3D point clouds. Thanks to our latent nodes and our learning strategy, our method outperforms the state-of-the-art in both cases. Moreover, in order to highlight the benefits of the geometric information and the potential of our method in simultaneous 2D/3D semantic and 2D/3D geometric inference, we perform simultaneous inference of semantic and geometric classes both in 2D and 3D that leads to satisfactory improvements of the labeling results in both datasets. Note that geometric classes are specified based on the geometric shapes of the scene elements, which include vertical planes, horizontal planes or cylindrical objects.

1.3 Thesis Outline

The remainder of this thesis is organized into five chapters as follows: Chapter 2 reviews different modalities that are useful for scene understanding, as well as the related works in multi-modal scene parsing. In Chapter 3 we study the potential of one specific such modality, multi-spectral imaging in classifying materials in outdoor natural scenes. Furthermore, to take advantage of using multiple modalities for scene analysis, Chapter 4 provides a method for 3D point cloud data classification using multi-view 2D information (panoramic imagery. Since some modalities may not describe the scene properly, for example moving objects are rarely adequately captured in Lidar data), labeling both 2D and 3D data simultaneously is favorable. Therefore, in Chapter 5, we propose a multi-modal graphical model for scene analysis. This model provides simultaneous inference of modalities, using other modalities' information. In Chapter 6, we introduce latent nodes for the multi-modal graphical model, which addresses the problem of data misalignment and label inconsistencies. Also, we propose using learned potentials to eliminate the need for hand tuning the parameters of our model. Chapter 7 concludes the thesis with a summary.

Chapter 2

Background and Related Work

In this chapter, we briefly introduce 2D and 3D modalities for scene understanding. Then, we review the literature and present the previous works on outdoor scene understanding using multiple modalities.

2.1 Sensory Modalities

There are different sensing modalities to capture information from our environments, e.g., visual and auditory modalities. In this work, we concentrate on 2D modalities, such as RGB and multi-spectral imaging, and 3D modalities such as depth and Lidar sensors for scene understanding.

2.1.1 2D Modalities

2D imaging has been widely used for decades and its use for scene analysis has been demonstrated convincingly [64, 54, 63]. For example, common cameras that provide color measurements (RGB) can be used for object classification by extracting texture information for the observed object. Multi-spectral imaging and panoramic imagery constitute other examples. The former captures additional wavelength bands compared to RGB imaging, and the latter provides wide horizontal view images. We further discuss these 2D imaging modalities in the following sub-sections.

2.1.1.1 Multi-spectral Imaging and RGB

Multi-spectral imaging typically facilitates capturing 2D information of objects in several particular wavelength bands including bands in the visible and invisible light ranges. RGB imaging can then be considered a simple type of multi-spectral imaging with three wide bands corresponding to the wavelengths of Red,Green and Blue light. Compared to RGB imaging,

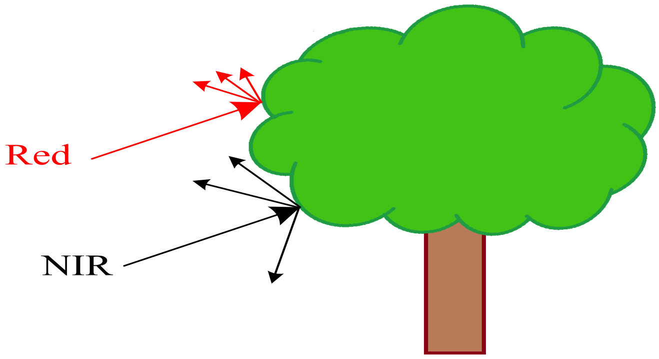


Figure 2.1: chlorophyll is a strong absorbent of light in the red spectral band (and scatters only a small portion of light in this band) and heavily scatters the other parts of the spectrum, especially the NIR band. This property can be used for vegetation detection.

multi-spectral imaging extracts additional information that the human eyes are often incapable of capturing. Various objects may show different spectral responses in different wavelength bands depending on their materials. This property that gives a unique spectral signature to each material, has been used for object classification [103, 102, 20]. One of the applications of multi-spectral imaging is in vegetation detection. Vegetation is chlorophyll-rich [23] and chlorophyll is a strong absorbent of light in the red spectral band and heavily scatters the other parts of the spectrum, especially the Near Infrared (NIR) band (Figure 2.1).

Figure 2.2 shows the seven filters used for the FluxData camera and a sample seven-band multi-spectral image composed of RGB, RGB-shifted (An image with three channels similar to RGB, with the difference that the wavelength bands are shifted in the spectrum, compared to the standard RGB wavelength bands) and NIR images. As another example, Figure 2.3 presents sample images from a sixteen-band Xiema camera [2] and the corresponding filters.

2.1.1.2 Panoramic Imagery

Panoramic imagery captures images with wide horizontal fields of view. The most common method for producing panoramic images is to take a series of pictures and stitch them together. These series of pictures can be captured by a single camera or several cameras. Using a single camera, which then needs to be a rotating one to capture a wide view field, is suitable for stationary scenes. However, in cases where the camera is mounted on a moving platform, e.g., a surveying vehicle in our experiments, using one rotating camera to capture panoramic images is not appropriate. In such scenarios, Ladybug cameras [3] are a good option. Therefore, in our experiments we employed a Ladybug3 camera (Figure 2.4) that has six 2 MP cameras. These

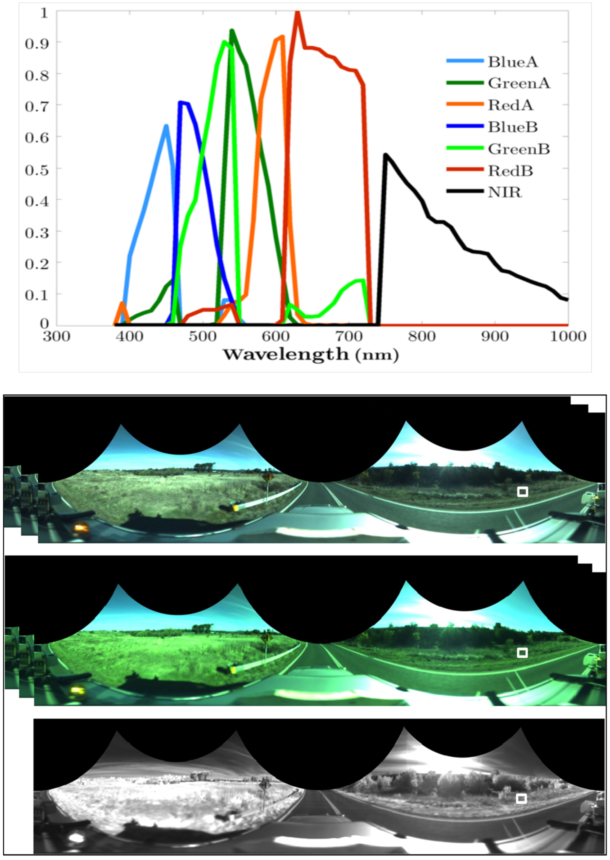


Figure 2.2: **Top:** Seven filters in visible and NIR range for the FluxData camera [1]. **Bottom:** A sample 7-band image composed of two RGB and RGB-shifted images and one NIR image from our terrestrial multi-spectral dataset.

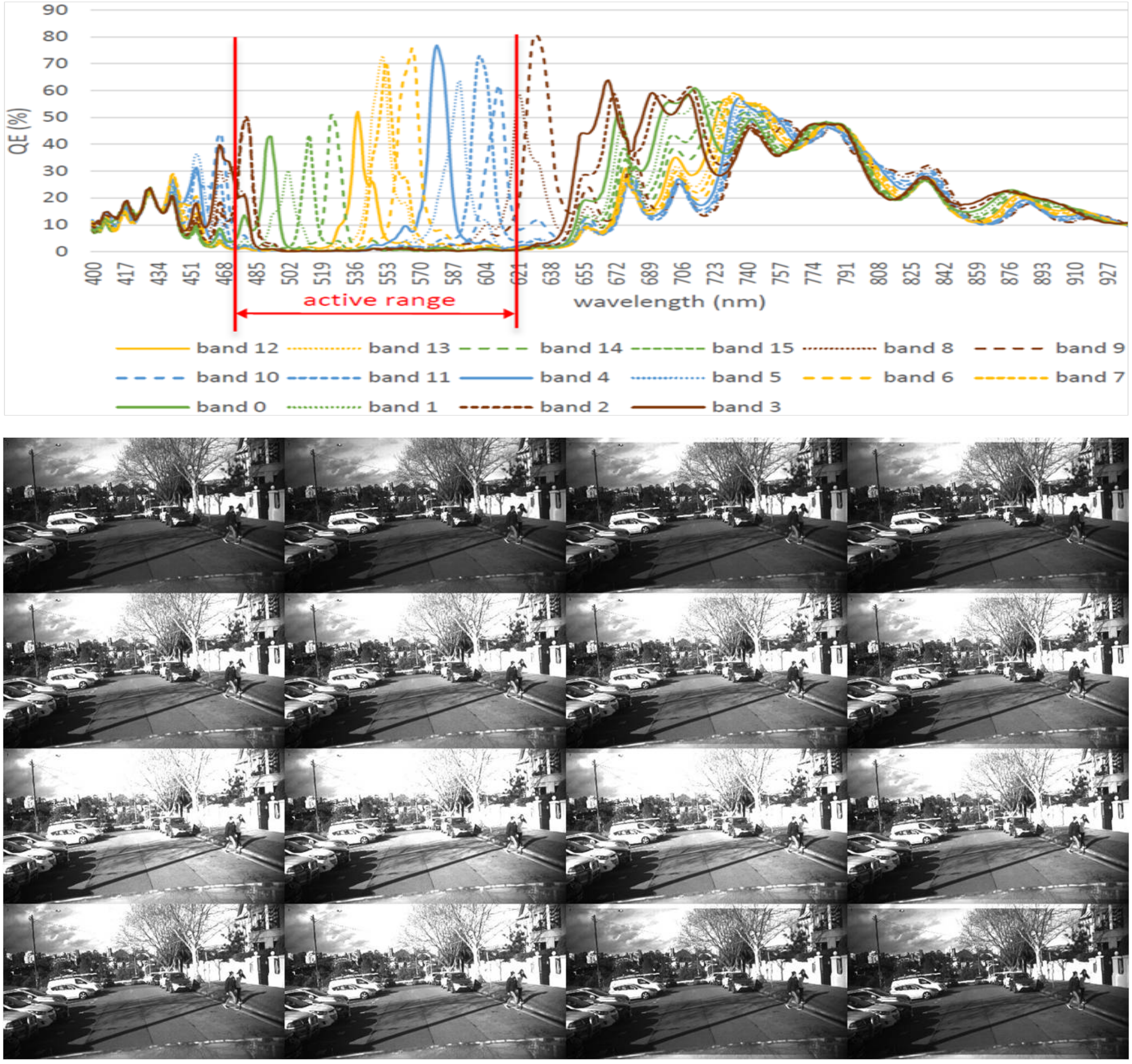


Figure 2.3: **Top:** 16 filters in the visible range for Ximea multi-spectral camera [2]. **Bottom:** A sample 16-band image from our Sydney multi-spectral dataset. Filters are designed for active range in the visible spectrum.

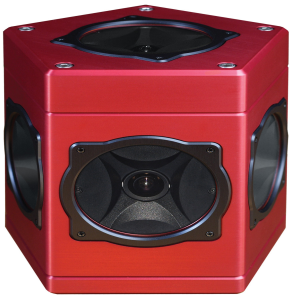


Figure 2.4: Ladybug 3 camera with six 2MP cameras [3].

cameras enable the system to collect video from more than 80 % of the full 360° sphere.

To obtain panoramic multi-spectral images for the purpose of the work in this thesis, instead of using several multi-spectral cameras that can be very costly, a multi-spectral camera is fitted with a panoramic mirror (*GoPano+*) to enable a 360° view. Panoramic imaging, in addition to providing a full view of our environment, is suitable to fuse with 3D Lidar point cloud data, as they both typically provide 360° coverage. Figure 2.5 shows two sample panoramic images. The top image was captured by the Ladybug camera and the bottom one is a multi-spectral image that was captured using a panoramic mirror. The panoramic images captured by ladybug camera typically have distortions in areas where sub-images are stitched together. The black regions in the panoramic images produced by the panoramic mirrors are the results of dewarping process.

2.1.2 3D Modalities

3D sensors provide 3-dimensional information about our environment by measuring distance between objects and the sensor. Having access to 3D data can be very beneficial for scene analysis due to its potential in providing shape, size and distance information. Two widely used 3D modalities are RGB-D cameras and Lidar sensors, which we describe below.

2.1.2.1 RGB-D Imaging

Today RGB-D imaging is very common to obtain 3D information from an environment. A well-known example of systems featuring this capability is the Microsoft Kinect sensor that has been widely used in the research community. RGB-D cameras typically provide both color and dense depth images. A dense depth image is a 2D image showing the distance to the points in a scene from the camera. This image is produced via a depth sensor often consisting of an infrared laser projector and a CMOS sensor that captures 3D data. In Kinect, the depth

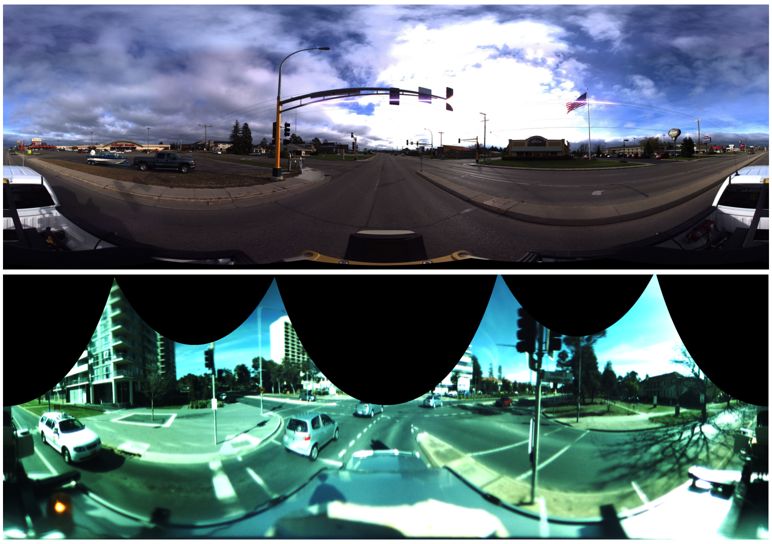


Figure 2.5: **Top:** 360° view Ladybug Panoramic image. **Bottom:** Panoramic multi-spectral image captured by a panoramic mirror (*GoPano+* [4])
(In this image just three bands in the RGB range are presented).

map is constructed by analyzing a speckle pattern of infrared laser light. The technique of analyzing a known pattern is called structured light [70]. The general principle of structured light is to project a known pattern onto the scene and infer depth from the deformation of that pattern. The Kinect takes advantage of structured light using two computer vision techniques, depth from focus, and depth from stereo [70]. Depth from focus exploits the fact that blur increases with distance and depth from stereo relies on the fact that the horizontal shift of a point observed from two different view points is inversely proportional to its distance to the camera. Figure 2.6 shows a sample RGB-D image. RGB-D cameras have been extensively used for indoor scene analysis due to their ease of access and use. However, their limitations in maximum distance coverage (around 4-5 meters), small field of view (57.8°) [6] and also low resolution make them inapplicable for our purpose of outdoor scene understanding. Note that, since RGB-D images are 2D images that contain distance information, they are usually called 2.5D data, where the 3D environment of the observer is projected onto the 2D planes of the retina.

2.1.2.2 Lidar Sensor

Lidar is a technology that measures distance from targets by illuminating them with light pulses. More specifically, objects are exposed to light beams with known speeds. Then the

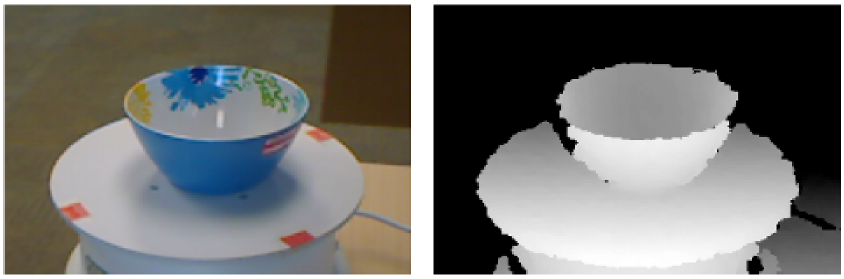


Figure 2.6: RGB-D sample images [5].

distance to objects are computed, given the time-of-flight of the signal travelled between the sensor and the object (Figure 2.7). Lidar uses near infrared light for this measurement (Ultraviolet and visible lights are also used for specific applications). It can target a wide range of materials, including non-metallic objects, rocks and trees. For terrestrial outdoor mapping, the Velodyne [7] sensor is a common choice. The Velodyne sensor scans the area using a rotating beam with individual 32 or 64 laser rays and has been widely applied in the applications such as the autonomously driving Google car [8]. In the Velodyne 64E [9], 64 lasers are mounted on the sensor and the entire unit spins. This allows for 64 separate lasers, each firing thousands of times per second, thus providing a rich point cloud. The unit inherently delivers a 360° horizontal field of view (FOV) and 26.8° vertical FOV [9]. This sensor is able to provide returns from surfaces up to 120 meters away. Note that since the point cloud is built by a rotating sensor, it may miss fast moving objects. Figure 2.8 shows the Velodyne 64E sensor that has been used for our dataset and Figure 2.9 presents sample point cloud data from our dataset. Note that a new technology (called Solid-State-Lidar) is coming, making 3D Lidar viable in consumer products. It has no moving part and uses an optical phased array as a transmitter which can steer pulses of light by shifting the phase of a laser pulse as it is projected through the array[1].

2.2 Multiple Sensory Modalities

Different modalities with their specific properties can help to capture certain properties of the environment. For example, 2D imaging provides information such as color and texture, 3D data supplies distance information, which in turn can be used to infer shape and size cues. Capturing data simultaneously with several modalities provides a rich source of information

[1] http://spectrum.ieee.org/cars-that-think/transportation/sensors/quanergy-solid-state-lidar

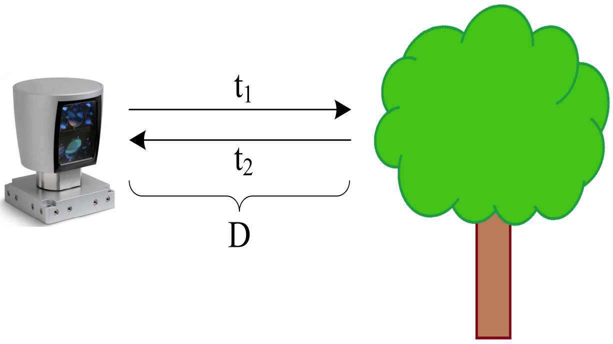


Figure 2.7: Time-of-flight: $t = 2.t_1 = 2.t_2$, measuring distance using a known speed light signal between the sensor and the object. Distance is measured by $D = C.t/2$.

Figure 2.8: Velodyne 64E with rotating beam and 64 laser rays

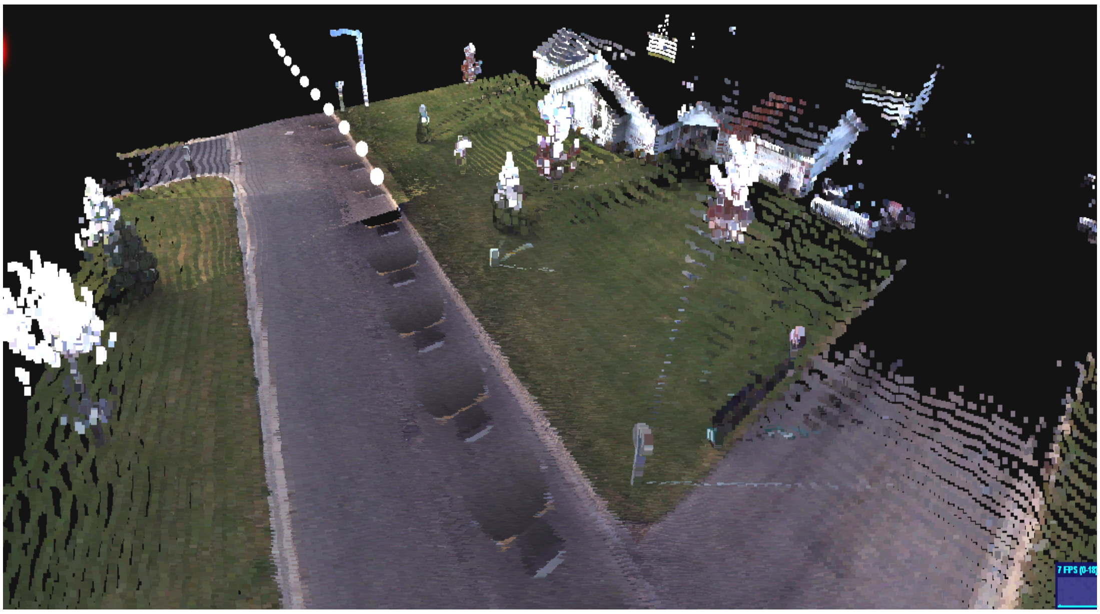


Figure 2.9: Sample point cloud data from our NICTA/2D3D dataset

about our environment that can improve classification results [30, 72]. The systems producing these data modalities are often mounted on a platform on a surveying vehicle to record the multi-modal data. Although employing several modalities is helpful for scene analysis, their different properties, data capturing methods and locations create new challenges. The multi-modal data should be aligned/registered with each other to enable their joint analysis. Alignment/registration is the process of putting all the various modalities data into the same coordinate system. It is important to note that even with a good registration/alignment between modalities, their corresponding elements may point to different items in the scene due to the different properties of the modalities. For example fast moving objects are often not captured correctly in Lidar data due to its rotating capture system.

2.2.1 Registration

The process of aligning various data modalities and putting them in the same coordinate system is called registration. Data can be from different sensors, viewpoints and times. There are single-modality registration to register the data from the same modality and multi-modality registration to align data from different modalities. Different registration methods have been proposed with applications in various fields, such as remote sensing multi-spectral classification, environmental monitoring, change detection, medicine combining data from different modalities, e.g., computer tomography (CT) and magnetic resonance imaging (MRI) [111]. In particular existing methods include curve methods [12, 109], surface methods [28, 71], correlation methods [86], wavelength based methods [41] and soft computing based methods [81]. Note that the focus of this work is not on registration methods. However, registration is very important since our goal is to address the problem of misalignment between modalities. In multi-modality registration, for example for 2D image data and 3D point cloud data, 2D-3D projection can be applied. Access to the estimated point cloud coordinates as well as the pose of the surveying vehicle in that coordinate system with a known relationship enables us to achieve proper 2D-3D projection results. However, due to the different properties of 2D and 3D modalities and the fact that panoramic images are obtained by stitching several images, registration error at the borders of objects especially for narrow objects are inevitable. Figure 2.10 depicts three examples of mis-registration between 2D image and 3D point cloud data.

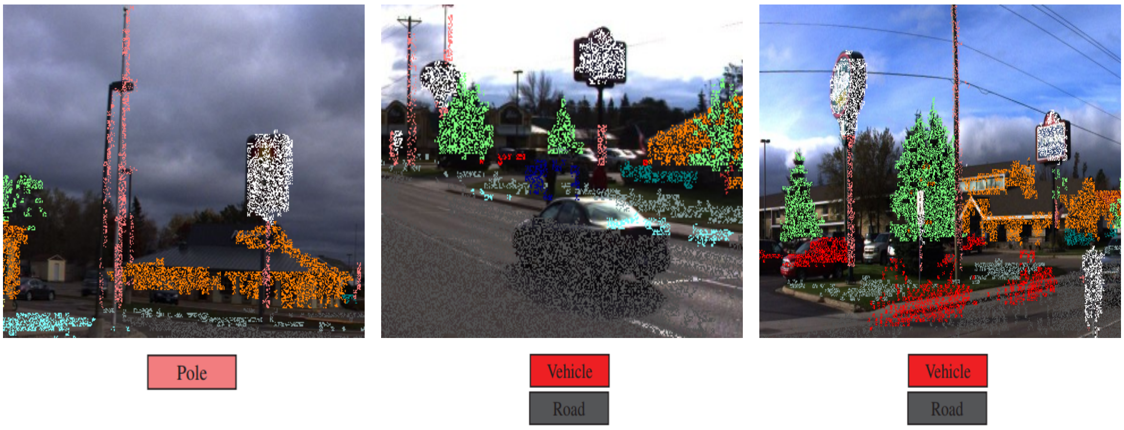


Figure 2.10: Three examples of misalignment between 2D and 3D data. Left: The projection of pole from 3D to 2D covers some regions of sky. Middle: A vehicle can be observed in 2D, but was not present when the 3D laser sensor covered this area. Right: This represents the opposite scenario where the image depicts an empty road, while the 3D points were acquired when a vehicle was passing.

2.3 Approaches to Scene Understanding

2.3.1 2D Scene Understanding

Scene understanding from 2D imagery has been intensely studied, yielding increasingly accurate results [92, 112, 57, 35, 114, 49]. Scene analysis using 2D images alone is ,however, not the subject of this thesis and while there is a large body of work on this topic, in this section, I focus the discussion on the most related works, especially the ones that utilize Conditional Random Field (CRF) to leverage the contextual information of the scene. The related work on multi-spectral imaging is reviewed in Section 2.3.1.1.

Terrain classification based on RGB images has, for example, been the topic of [94, 95] where typically color information of different objects along with their inherent textures are used for classification. Since the information of the individual pixels are very prone often to noise and *superpixels* convey information about neighboring regions, superpixels have been widely used in the past [36, 40, 56]. Superpixels are a group of pixels that have similar features (color or texture) and are obtained in an unsupervised segmentation process [24]. Chetan et al. [22] devised a method to segment terrain into road, muddy-road, rough-terrain, grass and obstacles. They utilized RGB and Local Binary Pattern (LBP) histograms and compared K-Nearest Neighbors (KNN) [93], Support Vector Machines (SVM) [104] and Random Forests [17] as classifiers in their system. Kim et al. [47] investigated terrain classification under different environmental conditions. They extracted color and wavelet features from the luma and chroma color space (YUV) and also spatial coordinates of the objects. They then classified the data using Neural Networks [93], SVM and a Maximum Likelihood classifier with Gaussian Mixture Models (GMM-ML) [47].

The classifiers that just consider the local information of the scene are called *unary* classifiers in which the labeling of each element is done independently of the other elements in the scene. These classifiers in most cases are not strong enough to classify the elements well and are heavily vulnerable to noise. In order to improve the results attained by the unary classifiers, higher-level knowledge, such as contextual information, can be leveraged by using graphical models and CRF in particular.

A CRF [58] (Figure 2.11) models a labeling problem with a graph where each node is one of the data elements, Pixels or superpixels. In this graph, the label of each node is dependent of its local evidence (result of the unary classifier) and the status of its neighbors. A set of nodes that have similar features is called a *clique*. A CRF formulation may consist of unary potentials, pairwise potentials and higher-order potentials. The unary potential indicates the cost of assigning a label to a single node and can be computed using the result of the unary classifier. Pairwise potentials and higher-order potentials however determine the cost of assigning a label combination to two nodes and a clique of more than two nodes, respectively. The nodes of a clique are conditionally dependent of each other. Let $\mathbf{x} = \{\mathbf{x}_i\}\ ,\ 1 \leq i \leq N$, be the set of features extracted from N elements of the data and $\mathbf{y} = \{y_i\}\ ,\ 1 \leq i \leq N$, be the set of variables encoding the labels of the nodes, where each variable can take a label in the set $\mathcal{L} = \{1, \cdots, L\}$. Then, the joint distribution of all modalities conditioned on the features can be expressed as

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z} \cdot \exp\Big(-(\sum_{i=1}^{N} \Phi_i + \sum_{(i,j)\in\mathcal{E}} \Psi_{ij} + \sum_{c\in\mathcal{C}} \Psi_c)\Big), \tag{2.1}$$

where Z is the partition function and Φ denotes the unary potentials. Ψ_{ij} denotes pairwise potentials defined over the set of edges $\mathcal{E}$ and Ψ_c denotes higher-order potentials defined over the set of cliques $\mathcal{C}$ with more than two nodes. [90, 117, 115, 110] used a CRF to model the contextual information of the scene. In [90] several local features (color, location and texture information) were employed and contrast sensitive *Potts* model were used as pairwise potential in CRF. The Potts model enforces two adjacent nodes to take identical class labels. More complex context information has been encoded in the pairwise term in the recent years. For example, the fact that airplanes are mostly observed with sky rather than with sea has been

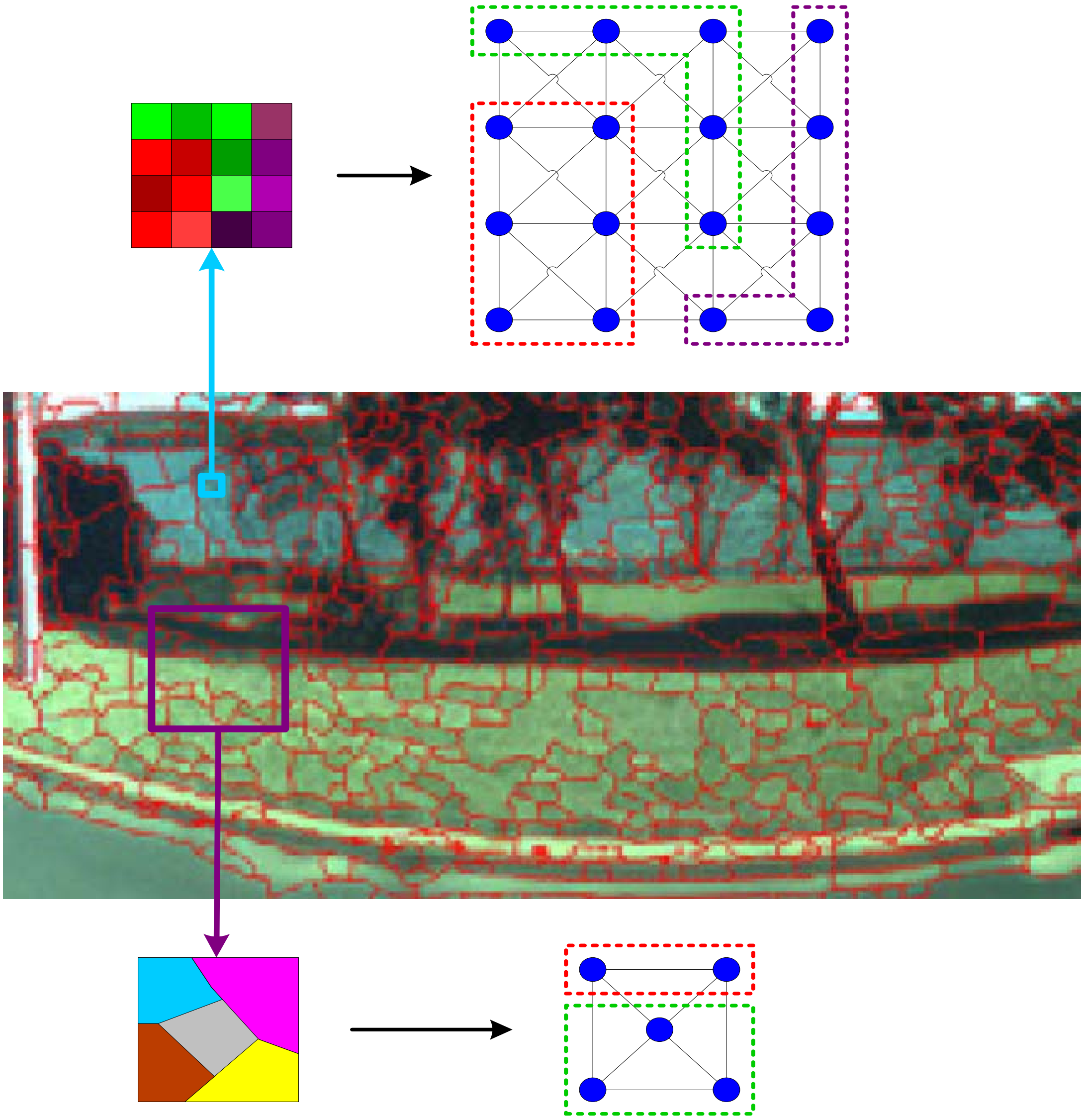


Figure 2.11: Middle: A sample of a segmented scene where the superpixels are presented by red lines. This image is zoomed in, so it appears blurry. Top: The pixel-based graphical model. Each pixel is denoted by a node in the graph (blue circles), all the neighboring pixels are connected in the graph via black lines that represent pairwise connections. The dashed lines show the scope of the cliques. Cliques are the set of nodes that are co-dependent. Bottom: The superpixel-based graphical model. In this model each superpixel is represented by a node in the graph (blue circles). Similar to the pixel-based model, each superpixel is connected to its neighbors via black lines, and cliques are the set of correlated superpixels shown via dashed lines.

modeled using pairwise potentials. [53] proposed a method for learning the label compatibility knowledge that can be applied to a fully connected CRF. In addition, higher-order models [48, 49, 51, 55, 105, 107] were used to capture higher-order relationships between the elements in the scene. [48] introduced the P^n Potts model to model higher-order relationships that encourage all the pixels within one image patch to take the same class label. [51] presented Pattern-based potentials for higher-order models, where higher-order cliques are encouraged to follow one of the library patterns that are learned previously.

2.3.1.1 Multi-spectral Imaging

Despite the useful information provided by RGB cameras, the information from a wide range of the light spectrum is not recorded using this data modality. With the prospect of low-cost multi-spectral imaging around the corner, a vast array of potential applications has opened up. By considering reflections of a material in different spectral sub-bands, material identification will be more tractable. As a result, multi-spectral imaging can facilitate object recognition tasks based on which material they are made. For example, objects that look the same to the naked eye may in fact look very different if the reflected light is sampled in more bands than the typical three bands that an RGB camera provides.

In the last decade, multi-spectral imagery have been a great asset to a variety of imaging fields. It is very useful for picturing some of the characteristics of objects and materials which can not be revealed in visible imaging. The number of sampled bands depends on the imaging system and imaging speed. Salamati et al. [84] performed a classification task on four types of materials including textile, wood, linoleum and tile using multi-spectral images (including visible and NIR). They based their approach on the intensities of NIR and color images and also textural features of the materials. Their result showed that augmenting the conventional visible data with NIR data improves the classification performance substantially. Brown et al. [18] extracted Scale-Invariant Feature Transform (SIFT) features in multi-spectral images and proposed multi-spectral SIFT (MSIFT) descriptors for scene category recognition. They took advantage of Principal Component Analysis (PCA) to reduce the dimensionality of the features. Salamati et al. [85] exploited these MSIFT features for an image categorization task and showed that adding a NIR band to the system improves recognition accuracy. Huynh et al. [43] exploited multi-spectral imaging for a skin recognition and material classification task. They applied a pre-processing algorithm to recover material reflectance and eliminate shadow effects.

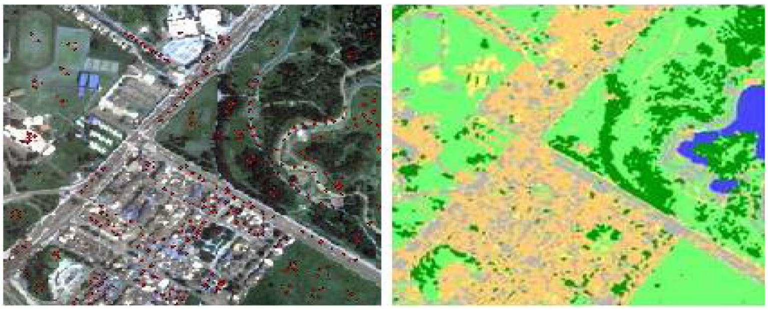


Figure 2.12: Left: A sample aerial image; Right: Final land-cover classification. Classified images show Road in gray, Roof in orange and cyan, Grass in light green, Trees in dark green, Water in blue and Bare Soil in yellow. [67]

In [84], material classification using multi-spectral images was performed indoors, for a few classes, and without the challenges of outdoor lighting. The authors in [18] and [85] exploited multi-spectral imaging for scene recognition and image categorization, respectively. While scene recognition and image categorization are important tasks, the problem of material classification is not being addressed. Outdoor terrain classification aided by multi-spectral images is addressed by [97, 16], however, only vegetation detection is considered. Additionally, aerial spectral imaging has been utilized in some terrain and urban environment classification tasks. Lizarazo et al. [67] classified urban land-cover into roads, rooftops, grass, trees, water body and soil, using RGB and NIR aerial images, by applying a fuzzy segmentation method. A sample of their aerial dataset and the classification results are shown in Figure 2.12. Fauvel et al. [32] classified urban area to different categories such as asphalt, metal sheet, brick and other terrain parts. They employed 115-band hyper-spectral aerial data, exploited spatial information and learned an SVM based classifier.

In the literature, there is some research on vegetation detection using terrestrial spectral imaging. Tarrant et al. [97] used a FluxDataTM camera to capture multi-spectral images in six visible bands and one NIR band to discriminate vegetation from real obstacles in routes. Benefiting from these image bands, a Normalized Difference Vegetation Index (NDVI) [101] was calculated to achieve a high accuracy in vegetation detection (Figure 2.13). Bradley et al. [16] worked on a similar approach and showed that NDVI is very informative for vegetation detection in outdoor environments. The above works are, however, limited to vegetation detection applications.

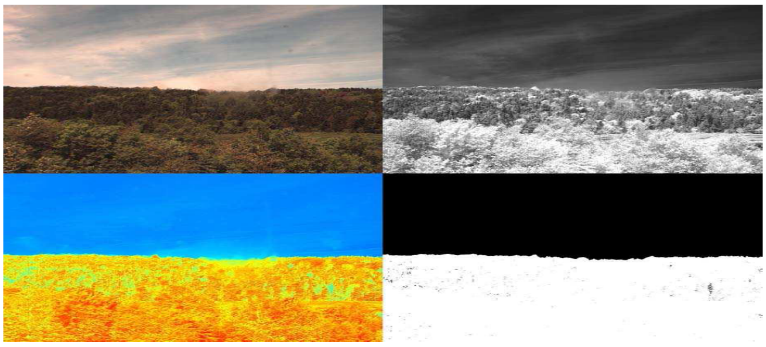


Figure 2.13: Left Up: RGB image; Right Up: NIR image; Left Down: NDVI; and Right Down: Discrimination image for vegetation [97].

2.3.2 3D Scene Understanding

2D features are often insufficient to describe objects and 3D shape information can help disambiguate between objects that would appear very similar in 2D. For example, a concrete wall can easily be distinguished from a concrete road using 3D shape features. Unfortunately, acquiring accurate and dense 3D shape features from a sequence of sparsely collected images is difficult or even impossible. In order to capture the 3D information of the scenes, Lidar systems have been widely used to create so called point clouds. Lalonde [60] used a Bayesian classifier operating on saliency features measuring the local point cloud distribution to classify natural terrain. They identified three general classes of objects which had scattered (like grass and bushes), linear (like tree branches) and planar (ground surface and big rocks) point cloud distributions. They measured the spatial distribution of the points in a local neighborhood by computing the eigenvalues of a 3D covariance matrix of each region. Saliency features were then constructed from these eigenvalues by fitting a Gaussian Mixture Model (GMM) [14] to the training dataset using Expectation Maximization [26]. Jutzi and Gross [45] classified the point clouds of urban buildings into general structural classes, such as edge, corner and plane, by assigning 3D spherical neighborhood volumes to each point and computing the eigenvalues of point distributions within this volume. They also estimated 3D contours of the objects by considering consecutive points with a similar eigenvector. These works attempted to classify the point cloud data into some generic point distribution categories and did not assign semantic class labels to the points.

In [31, 119, 98] different object-level and point-level features were extracted from the point cloud data to describe the scenes. Himmelsbach [39] proposed a system for vehicle detection, classification and tracking, using object-level and point-level features from the 3D object candidates. Their method for finding car candidates was to generate 2.5D occupancy grids via a segmentation process. Subsequently, they extracted the object-level and point-level features from the 3D object candidates and classified them into two classes of vehicle or non-vehicle with an SVM. Douillard [31] devised a feature-less classification system based on 3D template matching and the Iterative Closest Point (ICP) algorithm [13] to detect cars, poles, trees and walls in the environment. These works are quite capable of detecting some fundamental objects and point distributions, but a common problem in most of them and other related works is the limited number of classes. Moreover, despite the rich information that 3D point clouds give, they are relatively sparse and carry noise from, for example, scan misalignment and error in distance estimates. This leads to difficulties such as accurately segmenting and classifying complex outdoor scenes where the magnitude of the noise is sometimes similar to the size of the objects of interest. Furthermore, if two objects are located too close to each other, the 3D points may not be able to distinguish them appropriately. [11, 65, 69, 77] have worked on pairwise graphical models on point cloud data and improved the classification accuracy significantly. Munoz et al. [73] used a higher-order model to perform contextual classification of a 3D point cloud in an outdoor environment. The Markov Random Field [65] is used as their model to consider contextual information and the parameters of this model was defined by a functional gradient approach. Also these authors [74] used higher-order Associative Markov Networks on 3D outdoor point clouds. They defined their high-order cliques in the 3D point cloud as a set of locally similar points obtained by k-means clustering [46] over the points' features and locations. For their clique potentials, they considered similar potentials to the work in [48] called P^n Potts model. This associative model favors all variables in the clique taking on the same label. Xiong et al. [113] used a sequence of hierarchical classifiers at different scales (region-wise and point-wise) where the class predictions of each classifier was given as a feature-set to the classifier in the next stage. Then, through an iterative process of going back and forth between the stages, they predicted the final labeling of the 3D point cloud data with some promising results. However, the convergence of their method should be investigated when dealing with a large number of class labels. In Figure 2.14, Figure 2.15 and Figure 2.16 we show some results of [73], [74] and [113], respectively.

2.3.3 Multiple Modalities

2D and 3D data are very helpful for outdoor scene understanding and provide different types of information about our environment, but they both have their own weaknesses depending on the contex. These problems, however, can in some cases be handled by leveraging the comple-

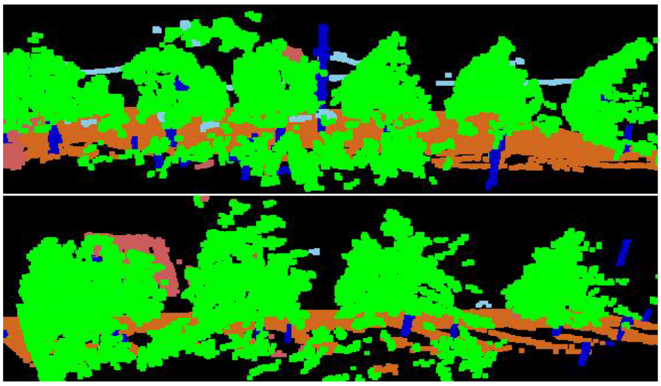


Figure 2.14: A sample results of terrain classification [73] using a higher-order model to perform contextual classification of a 3D point cloud in an outdoor environment, orange = ground, green = vegetation, dark-blue = tree-trunks/poles, sky-blue = wire, red = facade.

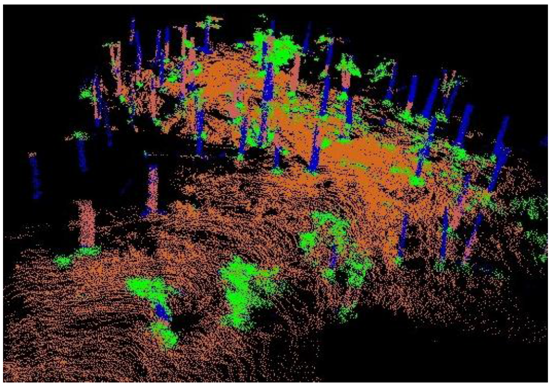


Figure 2.15: A sample results of 3D point cloud classification [74] using higher-order Associative Markov Networks, vegetation (green), large (red) and small (blue) tree trunks, and ground (orange).

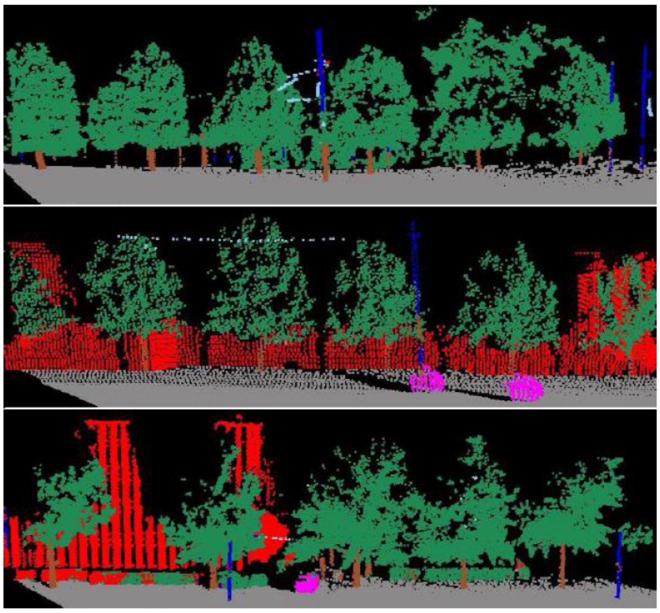


Figure 2.16: Example results of 3D point cloud classification [113] from VMR-Oakland-v2 dataset using a sequence of hierarchical classifiers at different scales (region-wise and point-wise), grey = ground, light-red = building, brown = tree-trunk, dark-green = vegetation, pink = vehicle, dark-blue = pole, lightblue = wire [113].

mentary information coming from other data modalities. Combining 2D imagery and 3D point clouds for semantic labeling has been the focus of several recent works [80, 29, 118, 19, 72]. They utilize both image and Lidar data in order to extend the number of classified objects and improve the labeling accuracy. In particular, [80, 29] defined models on the variables corresponding to the elements of only one modality and augmented them with information extracted from the other modality. In [80], the authors proposed a probabilistic approach for labeling objects in an urban environment using both laser scanner and image data. 3D surface normal features were used in conjunction with 2D color, texture and geometric features were used to first segment the objects and then classify them into pavement, dirt path, smooth wall, textured wall, vehicle, foliage and grass. Douillard [29] designed a rule based system using 3D Velodyne Lidar data and monocular color imagery to classify the urban environment into 16 different classes. These approaches, however, assume that the same portions of the scene are observed in both modalities, which is virtually never the case in practice due to misalignments, 3D-2D projection errors and occlusions. By contrast, the model of [19] incorporates variables for the two domains, but still relies on a single variable for the corresponding regions in both modalities. Therefore, this model still assumes that the modalities are perfectly aligned. This, unfortunately, can typically not be achieved in practice, and the above-mentioned techniques

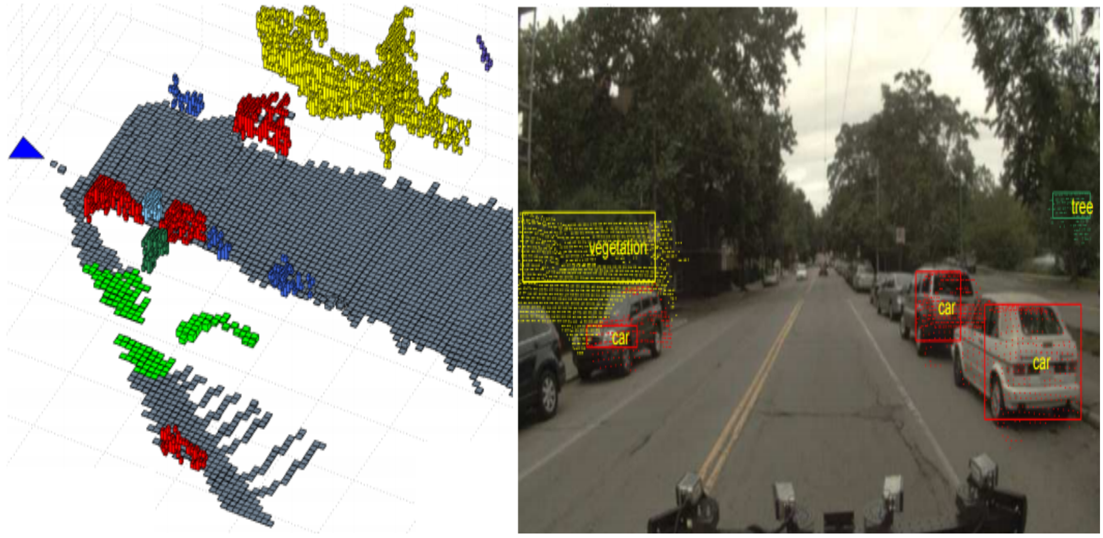


Figure 2.17: Semantic representation of one of the labeled scenes. Left image: 3D view of the inferred class labels. The blue triangle indicates the vehicle's position. Right image: The inferred labels as well as the ROIs and the projected laser returns. The color of each ROI matches the color of the associated object in the 3D plot [29].

will thus mis-classify some regions in at least one of the domains.

Some approaches have, nonetheless, proposed to relax this assumption by having separate variables for the two modalities, even for corresponding regions. In this context, [72] designed a hierarchical labeling approach that alternatively performs classification in each domain. They presented a new co-inference technique, where an integrated inference process was performed for 3D and 2D data types, simultaneously. They classified the urban environment into 21 categories and achieved a better performance compared to a simple integration of 3D and 2D features. However, since the classification result of one modality is then transferred to help labeling in the other domain, depending on the overlapping area of the projection of the 3D segment onto the 2D region, this method implicitly encodes the assumption that corresponding regions should have the same label. In [118], a framework to train a joint 2D-3D graph from unlabeled data was proposed. As in [72], this framework transfers the labels from one modality to the other, thus implicitly assuming that corresponding 2D and 3D nodes belong to the same class. While this assumption may seem reasonable, it is often violated in realistic scenarios due to misalignments, 3D-2D projection errors and occlusions. In Figure 2.17 and Figure 2.18 we provide some results of [29] and [72], respectively. Indeed, in practice, the different modalities are typically not perfectly aligned/registered. Furthermore, in dynamic scenes, moving objects may not easily be captured by some sensors, such as 3D Lidar, due to their lower acquisition

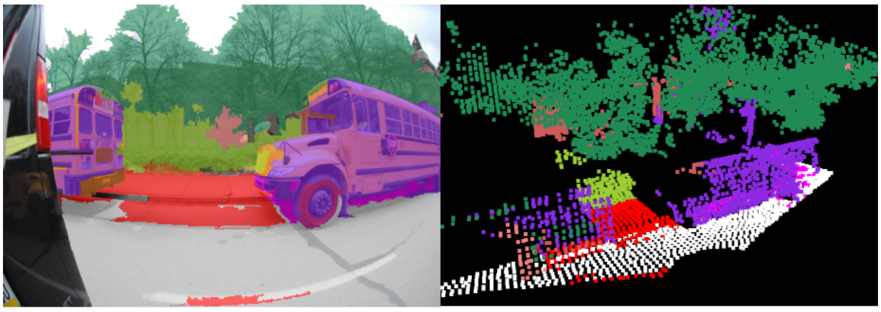


Figure 2.18: The co-inference approach results in 2D and 3D data [72] using a hierarchical labeling approach that alternatively performs classification in each domain. Color code: purple=big-vehicle, dark-red=sidewalk, white=road, light-green=shrub, darkgreen=tree-top, brown=tree-trunk, light-red=building, pink=small-vehicle.

speed. To give a concrete example, in the NICTA/2D3D dataset, 17% of the connections between the two modalities correspond to inconsistent labels. As a consequence, since existing methods fail to model these inconsistencies, they will typically produce wrong labels in at least one modality.

In this thesis, our aim is to benefit from multiple modalities to improve outdoor scene understanding and address the challenges relevant to this problem.

Chapter 3

Multi-spectral Imaging for Materials Classification in Scene Analysis

In this chapter, we investigate the potential of multi-spectral imaging for outdoor scene understanding. A method suitable for distinguishing between different materials appearing in natural scenes using such a multi-spectral camera is devised. The application we have in mind is a system capturing natural outdoor imagery in road scenes, and we are interested in classifying the objects in the environment based on their material. This kind of information is useful to, e.g., create large scale inventories of materials for road asset management and vegetation management.

Considering the RGB based approaches [22, 47] and their inherent limitations in material identification, we propose an automatic system for material classification in natural environments by using multi-spectral images alone. Multi-spectral cameras are still expensive but will be commonplace and cheap in the next few years thanks to the technology evolution. The rest of this chapter is organized as follows. Section 3.1 describes the capture platform and the associated data which have been used. Next, we introduce our approach in Section 3.2 and our experimental results in Section 3.3. The last section provides a summary.

3.1 Data Capture System

The multi-spectral imagery studied in this chapter was recorded from the roads around Canberra, using a *FluxData*^{TM} [1] camera configured to capture seven frequency bands. Six of the bands are in the visual range and one band is in the NIR range. Figure 3.1-a gives an overview of the bands and respective photon efficiency. The camera uses three individual 2M pixel sensors which capture 3, 3 and 1 bands, respectively.

The *FluxData*^{TM} camera was fitted with a panoramic mirror (*GoPano+*) to enable a 360°

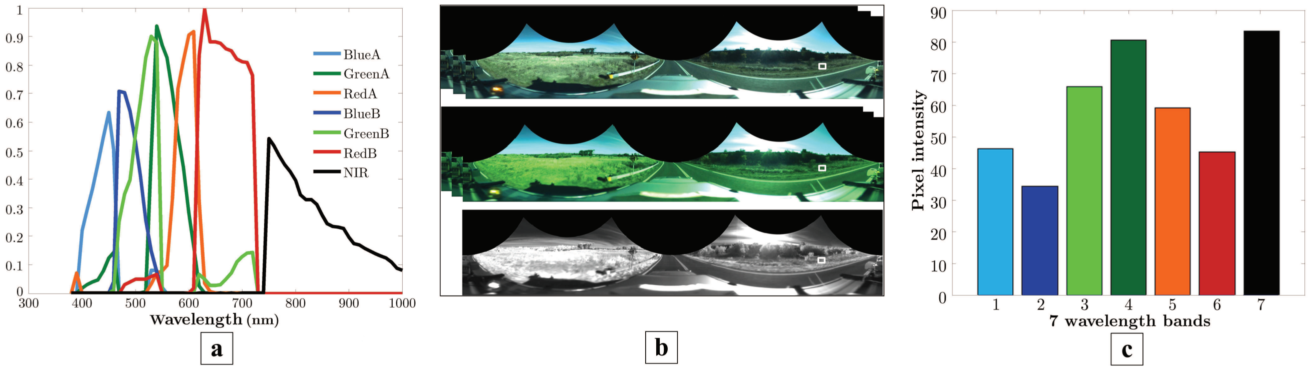


Figure 3.1: a) Seven Filters devised to achieve multi-spectral intensities. b) A sample 7-band image composed of two RGB and RGB-shifted images and one NIR image. c) Intensities of seven bands for a pixel within the specified white box inside the images in (b).

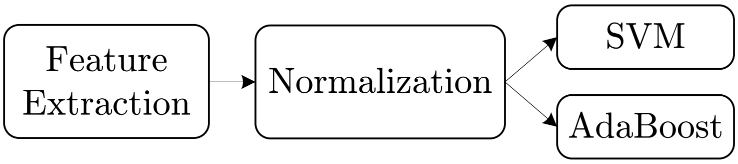


Figure 3.2: A general overview of our method: feature extraction, normalization and then classification using both SVM and AdaBoost.

view, and subsequently attached to the surveying vehicle. In a post processing step, the spherical images were unwarped by a software package associated with *GoPano+*, creating panoramic images of 1241×4176 pixels. Images were taken approximately every 2.7 meters along the road. Figure 3.1-b shows a sample image from this setup, in which, a pixel is selected and its multi-spectral signature is illustrated in Figure 3.1-c.

3.2 Method

The method is comprised of three stages; feature extraction, normalization and then classification. In the classification stage, both SVM and AdaBoost [33] are presented. Figure 3.2 shows an overview of our method.

3.2.1 Feature Extraction

Pixel-level and region-level features were extracted in a pixel-wise manner. The pixel-level features are directly computed from the intensities in the seven bands of each individual pixel, whereas the region-level features are extracted from the local neighborhood of each pixel.

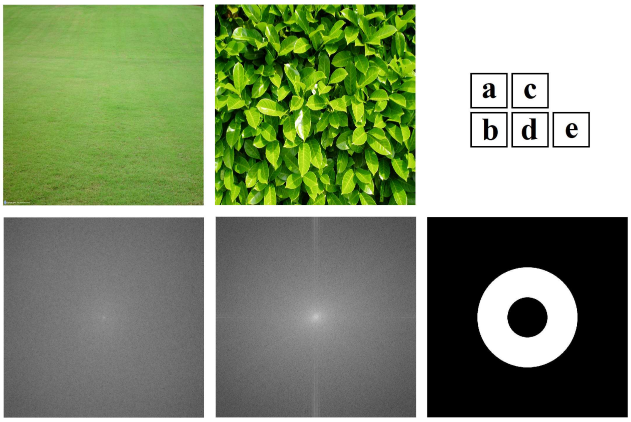


Figure 3.3: Utilizing Fourier spectrum to estimate the fineness of the image. a) A homogeneous image (grass). b) Fourier transform of grass. c) A detailed image of leaves. d) Fourier transform of leaves. e) The mask which is used to extract fineness feature from the Fourier spectrum.

3.2.1.1 Pixel-level features

These features are extracted from the information at a single pixel location. They are:

-Intensity features:

For each pixel, we store the seven intensities obtained from the seven bands of the sensor.

-Normalized Difference Vegetation Index (NDVI):

One of the most important characteristics of vegetation is that it is chlorophyll-rich [23]. Chlorophyll is a strong absorbent of light in the red spectral band. On the other hand, it heavily scatters the other parts of the spectrum, especially the NIR band. From these observations, vegetations can be detected using [101]

$$NDVI = \frac{\rho_{NIR} - \rho_{RED}}{\rho_{NIR} + \rho_{RED}}, \tag{3.1}$$

where ρ_{NIR} and ρ_{RED} represent the spectral reflectance in the NIR and Red bands, respectively. It can be demonstrated that, for vegetation regions, NDVI is approximately $+1$.

3.2.1.2 Region-level features

The following features are extracted from a block of radius R around the pixel of interest.

-Mean and Standard Deviation:

These two features are obtained by computing the mean and standard deviation of multispectral intensities in the neighborhood block. In total, 14 features from 7 bands are computed for each pixel.

-Gray Level Co-occurrence Matrix (GLCM):

This matrix represents the distribution of gray-level values in the image [38]. In other words, it reveals some specific neighborhood structure that exist among the gray-level values. Here, the GLCM is calculated for the block surrounding the pixel of interest. Several properties can be computed from this matrix, among which, Contrast, Homogeneity and Energy are utilized [38]. We compute two different GLCMs: one to encode the influence of pixels in the horizontal direction and one for vertical direction. These texture features are independent of intensity scaling of the image. This comes from the fact that the GLCM matrix is computed from the number of intensity levels in the image, and not from the intensity values. As a result, multiplying the whole image by a coefficient does not affect the co-occurrence matrix. It makes this matrix suitable to classify similar objects in different lighting conditions. In total, 42 features from 7 bands are computed for each pixel.

-Fourier Spectrum:

The Fourier transform [44] of the image can be used to indicate some properties of the image as well [61]. For example, the fineness of the picture can be estimated by examining how much spectrum of Fourier transform is focused around the center. A smooth image has an almost compact Fourier spectrum. However, the presence of details in the image results in a more scattered Fourier transform. Figure 3.3 shows that the more detailed image has a sparser Fourier spectrum. A measure of sparseness of the Fourier spectrum can be obtained by applying a mask to keep the information in the mid-frequency bands. This mask is presented in Figure 3.3-e.

Furthermore, the Fourier spectrum also defines directional texture very well. In Figure 3.4, we can see that vertical texture in the image of wood gives rise to a horizontal pattern in its Fourier transform. The masks that are used to extract directional features from the image blocks are shown in Figure 3.4-c,d.

In the above, the masks are embedded in neighborhood blocks of radius R. The mean of the intensity of the pixels inside these masks are extracted as Fourier features. To make these features independent of the intensity scaling, they are divided by the mean value of their corresponding block. In total, 28 features from 7 bands are computed for each pixel.

A normalization process is applied to the features to bring them into a similar range of values, aiding the subsequent classification. After the normalization, they are zero mean with a standard deviation of one.

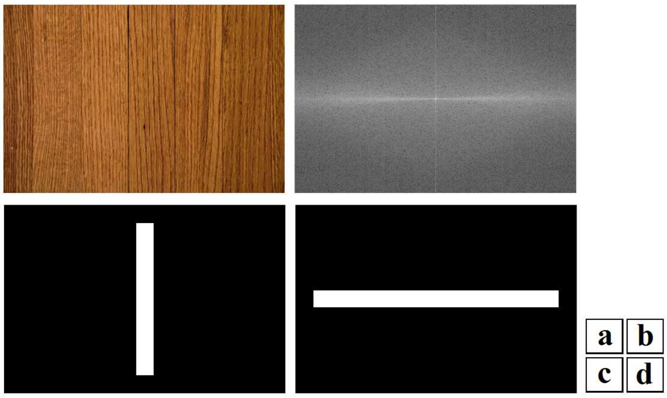


Figure 3.4: Utilizing Fourier spectrum to find directional patterns in the image. a) Image of wood surface with a vertical pattern. b) Fourier transform of wood surface. c,d) The masks which are used to extract directional features from the Fourier spectrum.

3.2.2 Classification

Although SVM and AdaBoost are well-known classification methods, in this section, we briefly describe these classifiers that were used in this chapter.

3.2.2.1 Support Vector Machine (SVM)

The SVM classifier attempts to find the hyperplane that divides the data points into their correct classes with the maximum possible margin:

$$\min_{w,b,\xi} \quad \frac{1}{2}w^T w + C\sum_{i=1}^{T} \xi_i \tag{3.2}$$

$$\text{s.t.:} \quad y_i(w^T x_i + b) \geq 1 - \xi_i, \; \xi_i \geq 0, \; \forall i = 1, 2, \ldots, N \; , \tag{3.3}$$

where N is the number of training data. Here we make use of kernel SVM with a Radial Basis Function (RBF) kernel:

$$\mathcal{K}(x_i, x_j) = exp\left(- \frac{\|x_i - x_j\|^2}{2\sigma^2} \right) \tag{3.4}$$

This kernel is applied to the 85-dimensional features discussed in Section 3.2.1. Parameters γ and C are optimized using validation data. We make use of the *LIBSVM MATLAB*^{TM} toolbox [21].

3.2.2.2 AdaBoost

To handle non-linearities in the classification problem, and when the features are high-dimensional, AdaBoost has proven as a highly effective approach. AdaBoost combines multiple weak learners, where each one has a specific weight, into a strong classifier [33]. The final classifier can be expressed as

$$h(x) = sign\big(\sum_{t=1}^{T} \alpha_t h_t(x)\big). \tag{3.5}$$

Here, each weak learner is built by employing a discriminant feature f_t and a threshold θ_t for classification:

$$h_t(x) = \begin{cases} 1 & f_t(x) < \theta_t \\ 0 & \text{Otherwise} \end{cases} \tag{3.6}$$

Note that a feature may be used more than once as a weak learner, with different thresholds.

3.3 Experimental Results

The system was implemented in *MATLAB*^{TM}. The dataset includes 169 images. We select ten classes as the targets for classification. Table 3.1 shows the assigned classes with their corresponding colors used for labeling the images. This label assignment can be observed in Figure 3.5. Note that, in addition to *grass* and *road*, four more classes, i.e., *leaves*, *tree trunks*, *white lines*, and *light poles and road guards* suffer from shadows. However, no separate class was assigned to them due to the difficulties of accurate manual labeling.

All 169 images underwent a labeling process based on the color codes in Table 3.1 and some regions from each class were labeled. 10 images were selected as the validation data for tuning the parameters of the models and features. In the next step, from the training data, 15,000 pixels from each class (150,000 in total) were randomly picked for the feature extraction part. 85 features, as explained above, were obtained for each pixel and for different block sizes, from $R = 4$ to $R = 15$. The best block size was chosen later, in the validation process for each classifier. Afterwards, we conduct 5-fold cross-validation in order to reduce the overfitting.

SVM: We apply an SVM classifier with an RBF kernel for training with different block sizes and different model parameters. These parameters were optimized using the validation data. The best validation accuracy[1] was achieved for the neighborhood block with a pixel radius of $R = 8$ and also for $C = 4$ and $\sigma = 0.04$. Choosing $R = 8$ seems reasonable because a very large block size leads to a smoothly labeled image which might be wrongly classified at

[1] Accuracy: The number of correctly classified samples divided by the total number of samples

Table 3.1: The targets which were classified using the system

Class number	Class name and color
1	Tree trunks:dark brown
2	Light poles and road guards: blue
3	Shadow on the grass: dark blue
4	Grass: dark green
5	Road: brown
6	White lines on the road: red
7	Shadow on the road: yellow
8	Leaves: green
9	Sky:light blue
10	Clouds and white regions in the sky: Purple

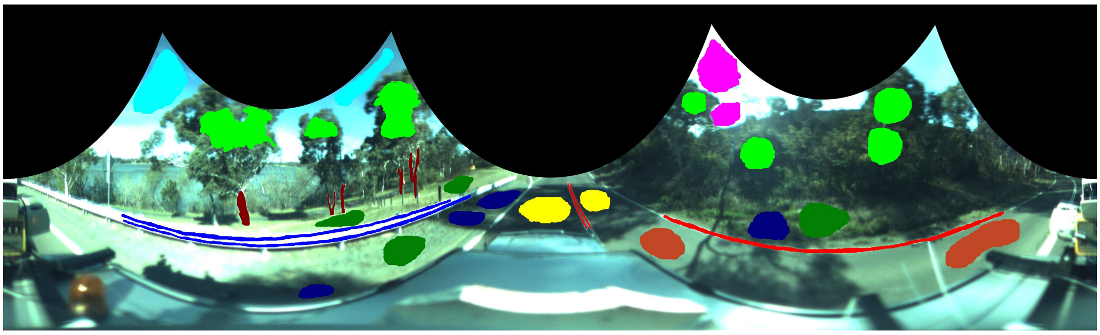


Figure 3.5: Sample labeled multi-spectral image

the edges, while selecting a very small neighborhood block results in a noisy labeled image.

The computed SVM model was employed to classify the test data and the accuracy was computed to be 92.9%. Table 3.2 shows the resulting confusion matrix of the SVM classification. Note that almost all classes were considerably distinguished from each other, despite the presence of shadows in most of the classes and also the similarity of pixel intensities between some of them. For example, although there is a high similarity between the pixel intensities of *grass* and *leaves*, these two classes have been discriminated down to an error of less than 2%. Furthermore, the system has been able to classify *shadow on road* and *shadow on grass* regions relatively well, using the features of Section 3.2.1. This clearly demonstrates the usefulness of texture features.

In addition, the average accuracy achieved by 5-fold cross validation using SVM was

Table 3.2: The confusion matrix computed using an SVM with an RBF kernel applied to the test data (Results are in percent and rounded)

	Tree trunks	Poles	Shadow-grass	Grass	Road	White lines	Shadow-road	Leaves	Sky	Clouds
Tree trunks	84	3	5	1	0	1	3	3	0	0
poles	15	76	1	0	7	1	0	0	0	0
Shadow-grass	9	1	63	0	0	0	25	2	0	0
Grass	0	0	0	98	0	0	0	2	0	0
Road	0	0	1	1	98	0	0	0	0	0
White lines	1	0	0	0	0	99	0	0	0	0
Shadow-road	1	1	4	0	2	0	92	0	0	0
Leaves	2	0	4	1	0	0	0	93	0	0
Sky	0	0	0	0	0	0	0	0	100	0
Clouds	0	1	0	0	0	3	0	0	0	96

91.9%. The difference between the maximum and minimum of classification accuracies for each class in cross validation was negligible.

AdaBoost: The number of weak learners selected by AdaBoost can often be significantly fewer than what is available in the pool of features. Finding an appropriate number of weak learners of the resulting strong classifier is done by measuring the performance of the strong classifier as a function of its size using the validation data. The feature pool consisted of a total of 85 features. The optimal number of weak learners turned out to be 80. Note that $R = 8$ was also found to be the best block size, as in the case of SVM.

Classification of the test data using the AdaBoost strong classifier with 80 weak learners resulted in an accuracy of 89.1%. A confusion matrix for this classifier is shown in Table 3.3. Furthermore, the average accuracy of 5-fold cross-validation using the AdaBoost classifier was 89.1%.

Full image labeling: Figure 3.6 shows a fully labeled image using the SVM classifier. The saturated pixels in the middle of the road are removed from the classification and indicated by black pixels. Clearly, our automatic approach shows great potential to discriminate between different classes in natural scenes.

Importance of sub-bands: Further experiments were made to establish the relative importance of the different sub-bands. Eliminating each of the six visible sub-bands resulted in less than 4% decrease in the overall accuracy of the system, while removing the NIR sub-band reduced the accuracy by about 10%. As shown in Table 3.4, the correlation coefficients between

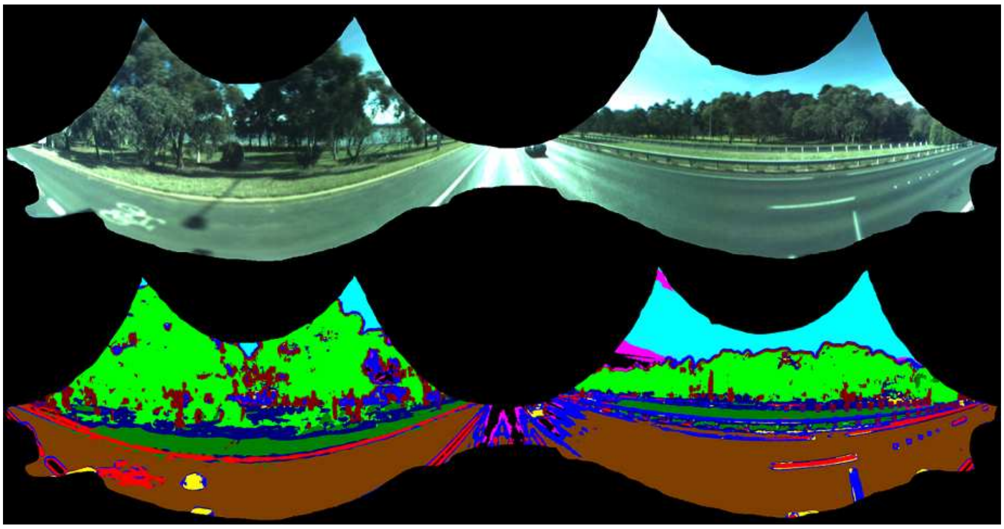


Figure 3.6: A sample of a fully labeled multi-spectral image

the intensities of the NIR image and the 6 visible range images are very low, compared to the correlation coefficients among the intensities in the visible spectrum images.

It shows that the image extracted from the NIR band can bring a considerable amount of information to the system and improve its accuracy. This result supports the previous studies [84, 18, 85, 96, 16] in which augmenting the NIR data led to a more accurate system.

3.4 Summary

In this chapter, as a first investigation into the utility of multi-spectral camera for material identification, we have proposed a method for discriminating between various materials in natural scenes based on imagery captured using a multi-spectral camera. In addition to the regular RGB spectrum, three visible sub-bands and one NIR sub-band were captured. This extra information and specifically the NIR image offer an improved classification system for distinguishing a variety of outdoor materials and objects. We extract the local texture features from 7 spectral bands for each pixel. We choose SVM and AdaBoost for classification, thanks to their great potential in to generalize. As a result, the test data were classified into ten pre-defined classes using SVM and AdaBoost with the average cross-validation accuracies of 91.9% and 89.1%, respectively. The results in ***Importance of sub-bands*** demonstrates the significance of multi-spectral imaging compared to using traditional RGB cameras.

Different lighting conditions for each pixel, such as shadow effects, influence the intensities of different spectral bands. Among the ten classes that have been assigned, six are dealing with shadows, completely or partly. Shadows can be found in *grass*, *light poles and road guards* and *light poles*, *leaves*, *white lines*, *road* and *tree trunk*. Tackling this issue, we added two extra classes for *shadow on road* and *shadow on grass* as they were readily marked up. Shadows on other materials were not added due to difficulties in consistent manual labeling. We used texture features independent of intensity scaling of the image to achieve a system which is more robust to varying lighting conditions. The results in Tables 3.2 and 3.3 shows that the proposed approach has been quite successful in separating these classes.

Note that this work was performed as an initial study and introduction to the general topic in this thesis. However, while there is no great novelty, the material lends itself to explain the general area of interest.

After studying multi-spectral imaging, we want to investigate the benefits of using multiple modalities in road scene understanding. Therefore, in the next chapter we utilize panoramic images and 3D Lidar data jointly to take advantage of multi-view 2D information for 3D classification.

Table 3.3: Confusion matrix computed using AdaBoost with 80 weak learners, applied to the test data. (The values are in percent and rounded)

	Tree trunks	Poles	Shadow-grass	Grass	Road	White lines	Shadow-road	Leaves	Sky	Clouds
Tree trunks	82	7	5	1	0	1	0	4	0	0
poles	13	78	1	0	3	5	0	0	0	0
Shadow-grass	10	3	63	0	0	0	21	3	0	0
Grass	0	0	0	97	0	0	0	3	0	0
Road	0	0	3	2	94	1	0	0	0	0
White lines	0	4	0	0	1	95	0	0	0	0
Shadow-road	3	4	7	0	0	0	86	0	0	0
Leaves	3	0	4	2	0	0	0	91	0	0
Sky	0	0	0	0	0	0	0	0	100	0
Clouds	0	1	0	0	0	3	0	0	0	96

Table 3.4: Correlation coefficients for pixel intensities of seven bands (RGB, shifted RGB and NIR bands) in training data. (The values are in percent and rounded)

	Red	Green	Blue	Shifted-Red	Shifted-Green	Shifted-Blue	NIR
Red	100	-	-	-	-	-	-
Green	92	100	-	-	-	-	-
Blue	87	95	100	-	-	-	-
Shifted-Red	98	89	86	100	-	-	-
Shifted-Green	88	98	90	87	100	-	-
Shifted-Blue	80	88	96	81	83	100	-
NIR	56	45	28	56	49	19	100

Chapter 4

Multi-view Terrain Classification using Panoramic Imagery and Lidar

After investigating the potential of multi-spectral imaging for outdoor scene understanding, in this chapter, we study 3D point cloud data for terrain classification and more specifically, how to improve 3D data classification using multi-view 2D information. In particular, we make use of terrestrial Lidar data and panoramic RGB images. Unlike indoor object classification, a system which is designed for the classification of outdoor environments should deal with many unavoidable issues such as occlusion, very complex scenes, variable weather conditions and also misalignment between the different data modalities. We devise a new framework to improve the classification and make it more robust against these issues that has many applications in natural scene classification tasks and robotics. In Section 4.1, we discuss the challenges of outdoor scene understanding especially using multiple modalities. The datasets and capture platform used in this work are introduced in Section 4.2 and then we describe the proposed approach. Finally, the performance of the proposed method is presented in Section 4.4.

4.1 Multi-view Outdoor Scene Understanding

It can indeed be useful to combine image data with Lidar. However, a common problem with the current state-of-the-art approaches is that the imagery is single-view only. If an object is partly or completely occluded in that view, the detection and classification of that object will be difficult, if not impossible.

Furthermore, a natural effect that is often seen in 2D outdoor images are dark shadows in parts of the image, which make some objects difficult to recognize. In addition, the excessive amount of light in some image areas gives rise to color saturation, which in turn, results in loss of information in those image regions.

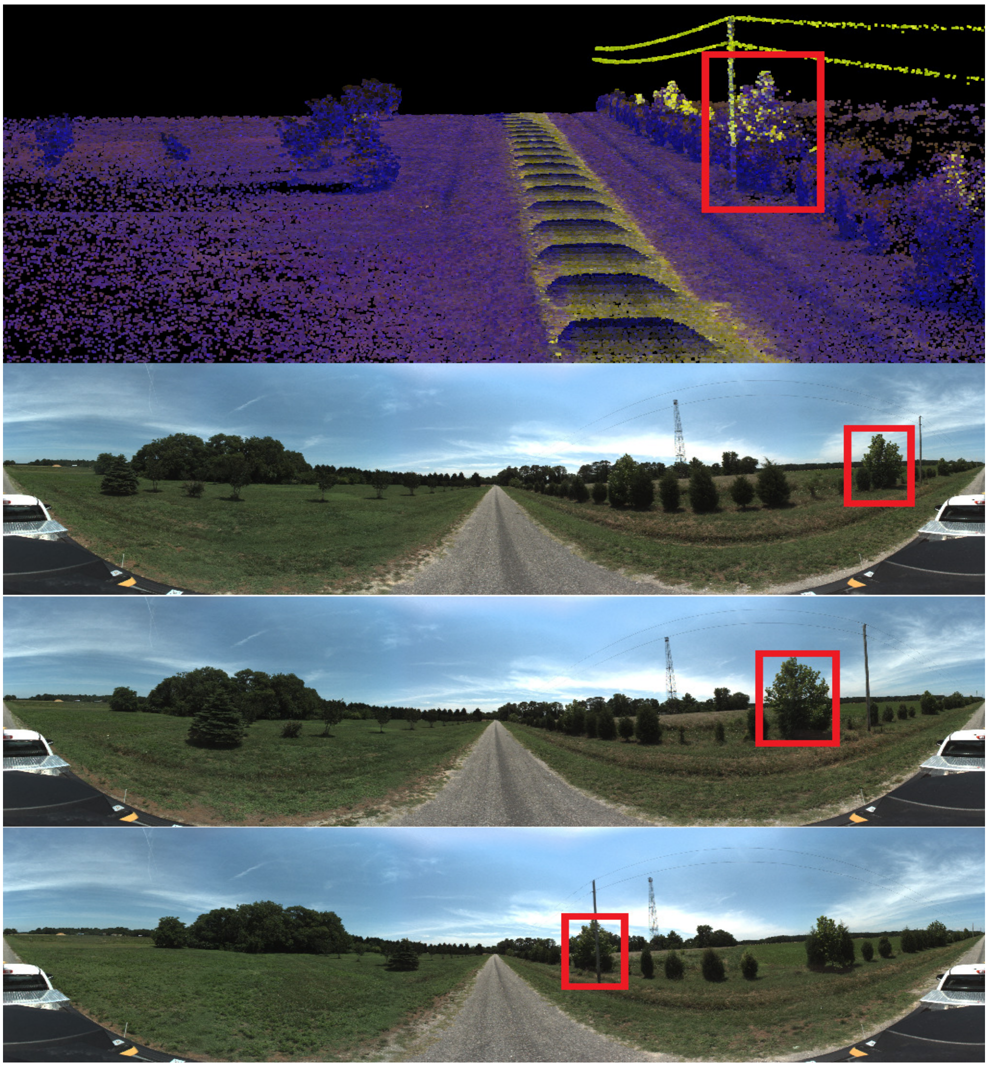


Figure 4.1: A tree in the 3D point cloud data that corresponds with three different views in the panoramic images.

Another challenging problem which exists in outdoor classification tasks is that the registration between the two sensor modalities (video imagery and Lidar) can never be perfect, and it changes slightly over time because of, for instance, vibration of the non rigid capture platform. This means that, due to varying amounts of misalignment between image and Lidar data, the 2D image features may be wrongly mapped to the corresponding 3D points. Also, in dynamic scenes, moving objects may not be easily captured by some devices, such as 3D Lidar, due to their low acquisition speed. Note that A Lidar system captures 3D data continuously using a rotating sensor, unlike snapshot sensors where the image data are captured instantaneously.

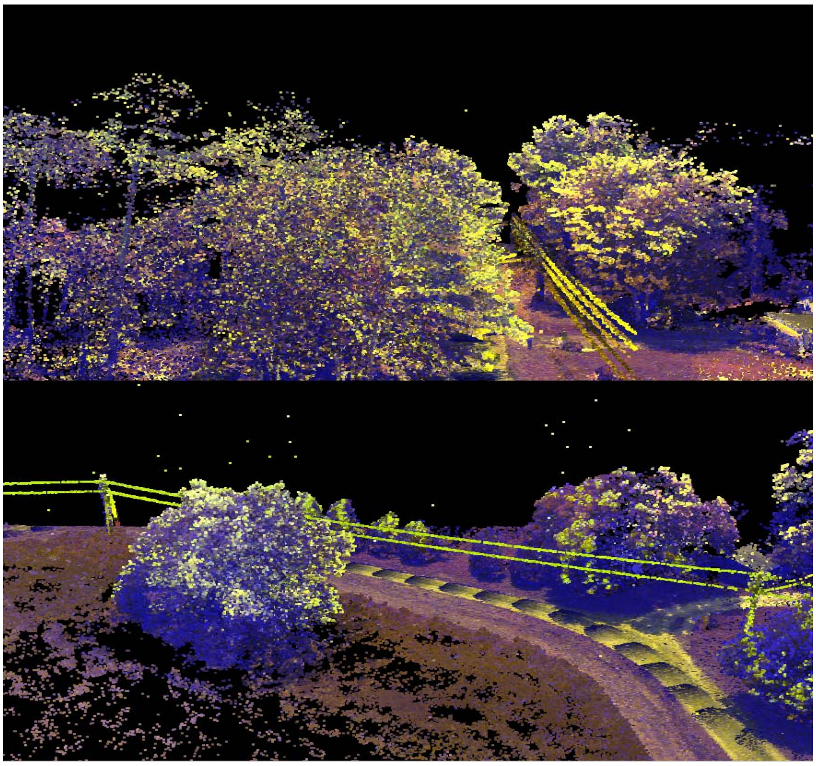


Figure 4.2: Two sample point cloud data in our dataset.

In this chapter, we propose a multi-view terrain classification system in which each scene is observed from several viewpoints at different points in time. Our approach exploits this property to address some problems that exist in outdoor object classification. The dataset includes 360° panoramic images that are captured every 3 meters along the road. This means that an object that corresponds to a set of 3D points is often viewed multiple times in the panoramic imagery. Figure 4.1 shows a tree in the 3D point cloud data that corresponds to three different views in the panoramic images.

It might be hard, or impossible, to obtain useful information from the corresponding pixel of a 3D point in one image, due to color intensity saturation, dark shadows or occlusion. In such cases, a different view of the same region might be more informative.

The proposed multi-view approach is able to seek consensus between the observations and can pick the image view that gives us the 2D features that most likely correspond to a particular region in the point cloud. As will be demonstrated, the difference in classification performance is significant.

Due to the nature of point cloud data, individual points are very vulnerable to noise and measurement errors. In order to improve the accuracy of the point cloud classification, a CRF

(Conditional Random Field) is added to the system [58]. We have modified the 2D CRF framework proposed in [75] to use in our 3D point classification system.

This CRF formulation is very adaptive against unary mis-classifications of neighbors and can also learn from mistakes of the unary classifier through a confusion matrix.

It should be noted that in this work, multi-view refers to multiple panoramic images of the same part of an object captured from different viewpoints and the goal is to find the most descriptive viewpoint. Therefore, our goal is different from indoor multi-view object recognition tasks such as [52, 15]. In addition, data from indoor settings usually do not suffer as much from the issues encountered in outdoor data discussed above.

4.2 Dataset

The data used in this work is composed of two synchronous sets of point cloud data and 2D panoramic images which are taken from road side objects in two seasons, summer and winter, and under different weather conditions. Both Lidar datasets were captured using a Velodyne-64E laser scanning system on top of a surveying vehicle and they were later partitioned into vertical blocks of 75m×75m, with unrestricted heights. Simultaneously, 360° panoramic images were captured using a Ladybug3 camera which was installed on the vehicle, with a spatial separation of 3 meters. The images are of size 2700×5400 pixels.

The first dataset was captured in the winter time with 845 panoramic images and 173 3D point blocks. The second dataset consists of 4307 panoramic images taken in the summer[1], along with 444 point blocks. Two samples of point cloud data are illustrated in Figure 4.2. To the best of our knowledge there was no publicly available dataset which could be used for implementing our panoramic multi-view approach and the most similar dataset to ours was provided by Munoz *et al.* (CMU/VMR dataset) which does not include images that captured objects from several views. KITTI [34] is the publicly available multi-modal dataset. The main problem with KITTI is the small vertical and horizontal Field of View (FOV). Therefore, similar to CMU/VMR dataset, it does not include images that captured objects from several views. Also, as shown in [19], a large portion of the images in this dataset does not correspond to any laser scanning data. This enables research on methods necessary to resolve issues such as correspondence ambiguities, occlusions (either spurious or due to parallax) and missing 2D-3D correspondences.

[1] The summer dataset contains bright images with a significant contrast and also more extensive vegetation coverage (Figure 4.1), but the images in the winter dataset are mostly captured in cloudy weather and are rather dark and low contrast (Figure 4.9)

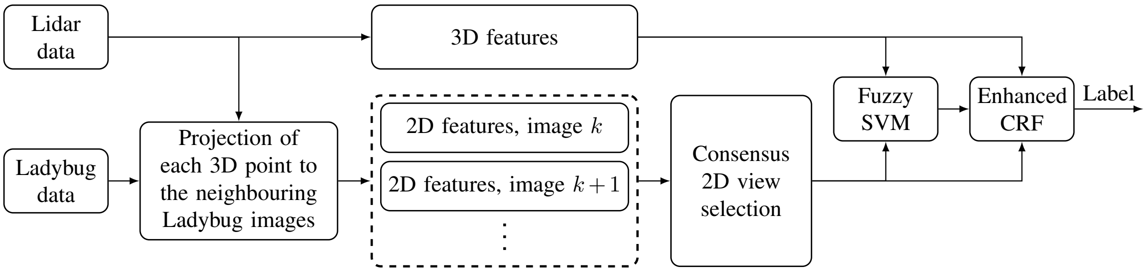


Figure 4.3: An overview of the approach presented in this chapter. 3D points are projected onto the panoramic Ladybug images for finding their corresponding image pixels. The 2D features are extracted from the corresponding pixels in different images and then in a 2D view selection process, the most representative 2D view is acquired. The total feature vector (2D+3D) is used in probabilistic SVM classification and subsequently in the CRF.

4.3 2D-3D Terrain Classification

The proposed terrain classification system is comprised of a number of steps. The point cloud is projected onto the nearby Ladybug image frames, from where 2D features are obtained for the corresponding regions. A consensus mechanism is used to select the most representative image source for each 3D point. Similarly, 3D features are extracted for those points in the point cloud. The 3D points are classified using an SVM classifier based on these 2D and 3D features. Finally, a CRF is incorporated into the classification scheme to improve the discriminative power of the system by considering local context. The overall approach is illustrated in Figure 4.3.

4.3.1 2D-3D projection

The data capture platform used here estimates the coordinates of th 3D points as well as the pose of the surveying vehicle in coordinate systems with a known relationship. This enables us not to get involved in the registration problem in great detail. Hence, the point cloud data is first transformed to the coordinate system of the camera. Assuming the origin of the camera is at $\mathbf{O_{cam}} = [\mathbf{O_x}, \mathbf{O_y}, \mathbf{O_z}]$, the new coordinates of each 3D point in the camera coordinate system are computed using the transformation

$$\mathbf{M} = \mathbf{T}[\mathbf{I}| - \mathbf{O_{cam}}], \tag{4.1}$$

where $\mathbf{T}$ is the rotation matrix of the camera and $\mathbf{I}$ is the identity matrix. In order to project the points in 3D space to the image plane, a frustrum model (perspective projection) is used.

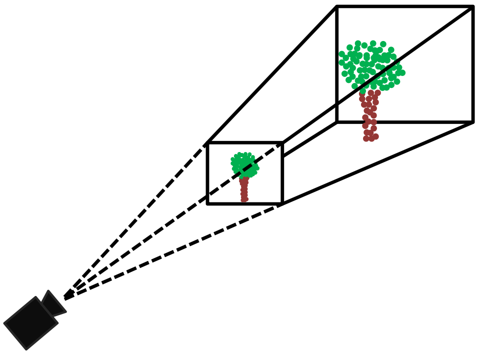


Figure 4.4: A frustrum that is used for modeling the camera.

The frustrum is a truncated pyramid that encompasses some of the 3D points which should be projected to the image plane (Figure 4.4). The maximum range of the laser scanning device is regarded as the far plane of this frustrum. Since our image dataset is 360° panoramic, we cannot find a single correct projection on a plane for the whole image. Instead, we divide the image into small sub-images which produces small frustra. Then the 3D points within each of the resulting frustra are projected to their corresponding sub-images on the image.

Figure 4.5-a depicts the projection of the point cloud that was previously shown in Figure 4.1 onto the panoramic image plane in Figure 4.5-b. Note that the correspondence between the two images is acceptable. The probable mismatches in some areas can be corrected using the proposed consensus system.

4.3.2 Consensus 2D View Selection

As mentioned above, each 3D point might be projected to the camera plane from several views along the road (Figure 4.6). Therefore, several 2D feature vectors are recorded for each point from different views, among which some might be noisy and less informative and some others might be irrelevant to the 3D point due to misalignment or occlusion.

In order to find the image view with the feature vector that best represents a point, all the recorded features from different views are investigated in a feature space. This space is built based on these features as its dimensions. Figure 4.7 illustrates a number of successive panoramic images taken from the road and roadside objects, including a power pole. A 3D Lidar point might have a correspondence with a region on the power pole in all of these image views. The 2D features from all of them are extracted and plotted in the feature space. It can be seen in Figure 4.7 that some views ($\mathbf{x}_1$ and $\mathbf{x}_2$) are separated from the others. This separation is due to color saturation, which is visible in the highlighted images. Since these images do

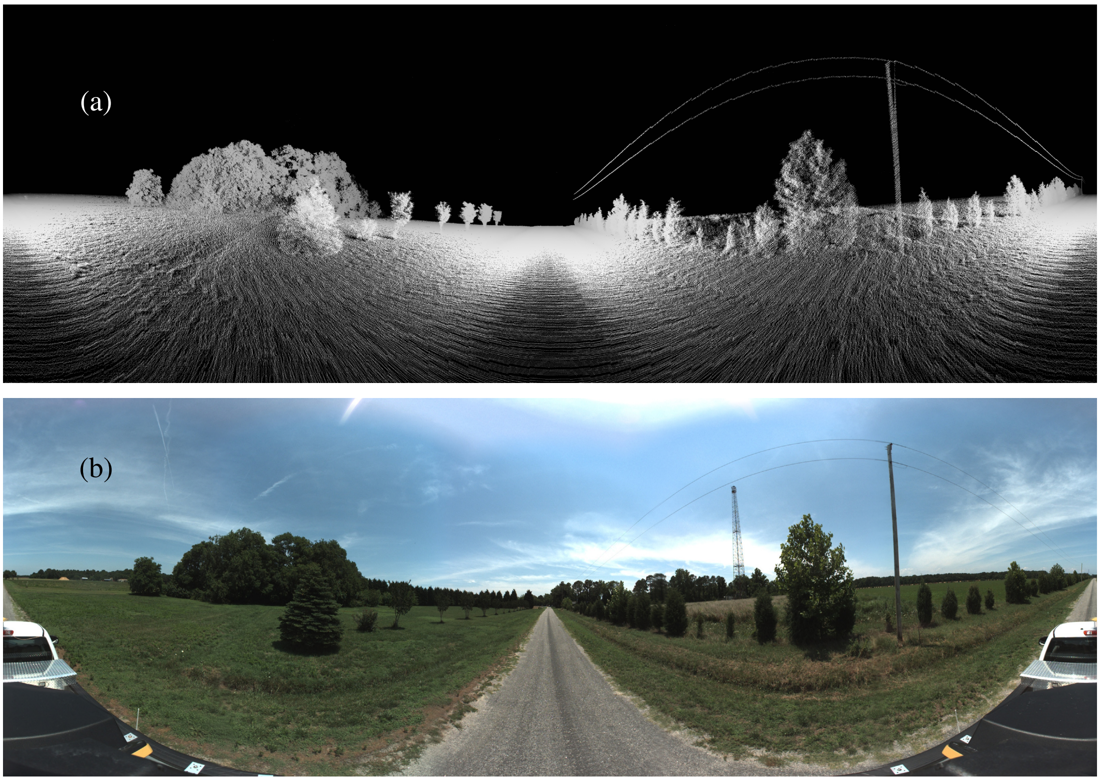


Figure 4.5: a) The projection of the 3D point cloud data onto a sample panoramic image plane in (b).

not convey any useful information about the power pole, they are not used to describe its 2D properties, and a good 2D representative should be selected from the rest of the image views. We formulate this process as the optimization problem

$$\mathbf{IDX} = \arg\min_{\mathbf{idx}} \sum_{i=1}^{N} \|\mathbf{x_i} - \mathbf{x_{idx}}\|, \tag{4.2}$$

where $\mathbf{x_i} = [\mathbf{f}_{i1}, \mathbf{f}_{i2}, \ldots]$ is the feature vector obtained from the i-th image view, N is the number of 2D views. Based on this formulation, the image view which has the least total distance from the others in the feature space is chosen to describe the region of interest. The selected feature vector is then augmented with the local 3D features and given to the classifier.

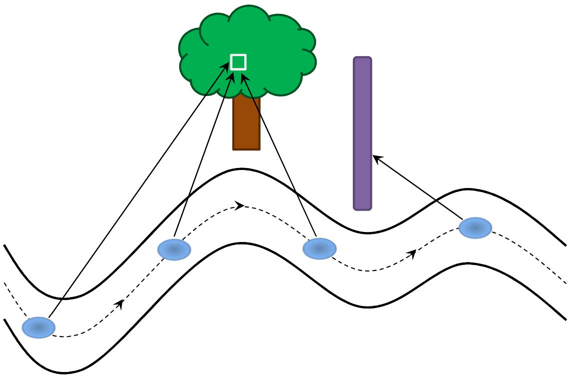


Figure 4.6: Panoramic Ladybug images capture the 2D information of an object from different views along the road. This enables a better understanding of the object properties.

4.3.3 CRF

We adapt the 2D CRF formulation presented in [75] to utilize it in our 3D classification framework. The unary part represents the certainty of the pointwise classifier (the SVM) while the pairwise function determines the amount of dependency on neighboring points.

We consider a 3D neighborhood graph in our CRF which looks like the one Figure 4.8. All points that are inside a sphere of radius **R** and centered at point i are considered as neighbors. To obtain the optimal labeling, the goal is to maximize the joint which is given by

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z} \cdot \exp\Big(- \big(\sum_{i=1}^{M} (\Phi_i + \sum_{j \in N_i} \Psi_{ij}) \big) \Big), \tag{4.3}$$

In this equation, $\mathbf{y} = \{y_i\}$, $1 \leq i \leq N$, is the set of class labels, $\mathbf{x} = \{\mathbf{x}_i\}$, $1 \leq i \leq M$, is the feature set extracted from the data and Z is the partition function. Additionally, M is the number of data items, N_i is the neighborhood space of data and Φ and Ψ are the unary and pairwise potentials, respectively. The negative logarithm of the probabilistic output of the SVM [21] is used to produce the unary term. That is,

$$\Phi_{(}y_i, x_i) = -\log(P(y_i|x_i) + \epsilon), \tag{4.4}$$

where, ϵ avoids having a zero argument in the logarithm. The SVM probabilistic output, $P(y_i|x_i)$, is generated using the approach of [78].

The authors in [75] proposed a novel pairwise function for a graph of 2D superpixels. Here,

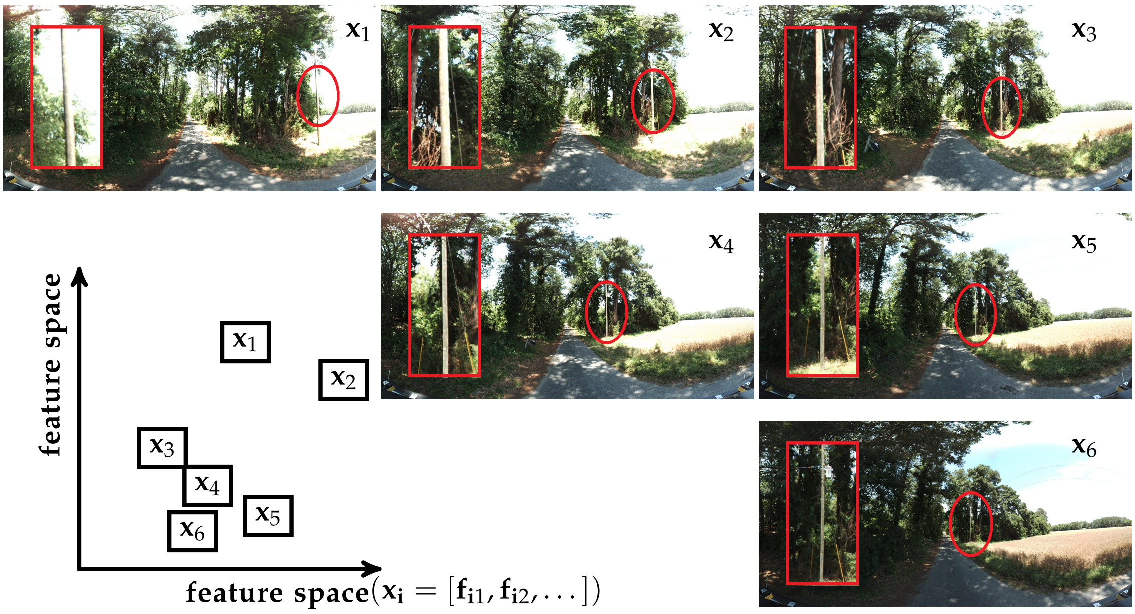


Figure 4.7: A number of a group of successive images containing a power pole (magnified) is shown. The distribution of the feature vectors that are extracted from a specific region of the power pole and from different image views is shown in a hypothetical multi-dimensional feature space. Despite the multi-dimensionality of the feature vectors, the feature space is shown here in two dimensions to simplify the illustration. In this example, the first two views are separated from the other views due to the apparent color saturation on the highlighted power poles in these views. The algorithm can easily eliminate these outliers using Equation 4.2 and select a satisfying view of the power pole from $\{\mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5, \mathbf{x}_6\}$.

we adapt this pairwise function to the case of 3D point cloud data. this yields

$$\Psi(y_i, y_j) = \Big(\frac{\omega D_{ij}}{1 + \|(x_i - x_j)\|}\Big)\delta(y_i \neq y_j) + \alpha\Big((1 - P_i(y_i))\Big)\Big((1 - P_j(y_j))\Big). \tag{4.5}$$

Here, the matrix D is responsible for penalizing the neighborhood of pairs of class labels that have had a large number of mis-classifications between them during the unary classification. Due to the amount of mix-up between these two categories, a high cost should be set for their adjacency to penalize the existing errors between them. On the other hand, a pair of object categories that are discriminated very well from each other using the unary classifier should not be strongly smoothed by the CRF. We use the confusion matrix to address this issue and, consequently, avoid over smoothing.

In general, if the classes i and j have a high misclassification rate in the unary classifier, their corresponding component in the confusion matrix will be large. Based on this fact, the matrix D can be computed by considering the confusion matrix of the unary classifier and

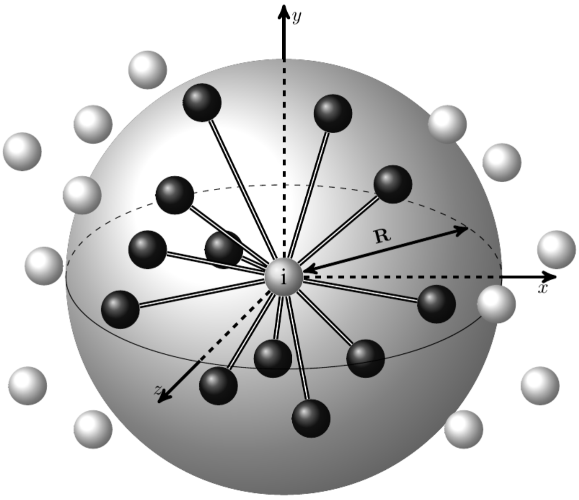


Figure 4.8: The graph of neighborhood which is used in our CRF framework. Each node (3D point) i interacts with its adjacent nodes that are within a radius of **R** (illustrated with black color).

setting its diagonal components to zero. Therefore, according to Equation 4.5, the pairwise cost for the neighborhood of these class labels will be increased.

In addition, $P_j(\mathbf{y_j})$ is the probability of the neighboring point j to get the label $\mathbf{y_j}$, which is computed using the probabilistic SVM classifier. If a neighboring point has a low class probability, it indicates that the SVM is not very confident about the class label of that point. Consequently, the influence of its pairwise interaction should not be as strong as the influence of other neighbors. To avoid spreading its, probably wrong, classification to other points, a large cost should be set for its pairwise connection with its neighbors. Equation 4.5 therefore assigns a higher pairwise cost to the interaction with a neighbor that has a more unreliable (lower probability) class label.

This CRF formulation is very adaptive against the mis-classifications of the neighboring points in the graph and can learn from the mistakes of the unary classifier, and also avoids over-smoothing, via the matrix D.

In Equation 4.5, α and ω are the CRF parameters which should be computed in the training process. CRF training is performed using a maximum pseudo-likelihood approach, where each point interacts only with other points in its proximity [89]. The CRF parameters are optimized by applying this technique to Equation 4.3. For inference, we choose the Iterated Conditional Modes (ICM) approach due to its simple implementation and the non-submodularity of the pairwise potential.

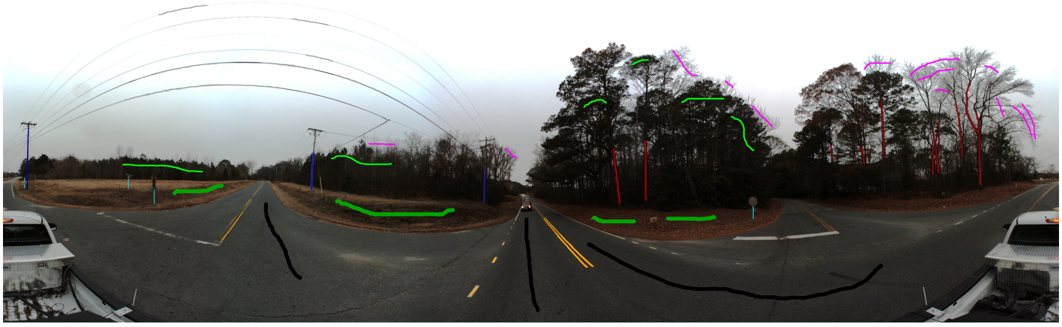


Figure 4.9: Illustration of the selected classes. The color codes are described in Table 4.1.

4.3.4 Noise Removal

While the Velodyne-64E laser scanner produces excellent 3D point clouds, there are occasional noisy data such as free floating points not related to any structure. Since these points probably do not represent any real object, they are removed. This operation is performed by computing the distance from a point to every other point in a neighborhood, and if the average is larger than a pre-defined threshold, the point will be eliminated from the data [83].

4.3.5 Features

The features that are extracted from the 3D point cloud data and the 2D imagery are described in this section.

4.3.5.1 3D Features

1) Ground:

The ground feature determines if a point is part of the ground plane. Each point block, which typically is 75m×75m (with no boundary along the **z** direction), is split into thin horizontal layers with equal thickness of 25cm. Hence, the layer which encompasses the largest number of points, is taken as the main part of the ground structure. Subsequently, a region growing algorithm based on the consistency of directions of the normal vectors is applied to the whole point block where the points in the selected layer are regarded as the starting seed points. The result gives us the ground plane in the point block.

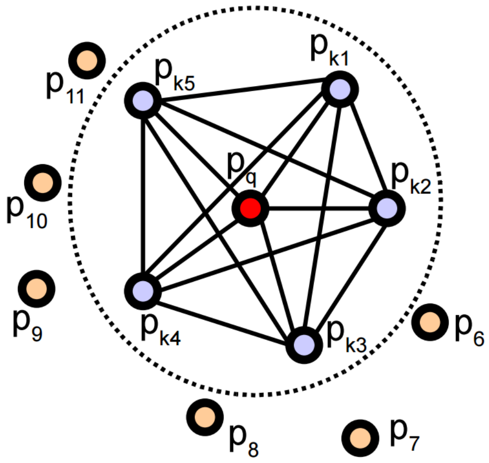


Figure 4.10: Influence region diagram of the PFH computation for a query point $\mathbf{p_q}$ (illustrated with red colour) [83].

2) Height:

The height of each 3D point is the difference between its z component and the z component of the closest ground point.

3) Curvature:

The curvature is a measure of the extent of the bending of the surface on which the point lies [83].

4-5) Linearity and Planarity:

Linearity and planarity features are computed as $\mathbf{f_L} = \lambda_1/\lambda_2$ and $\mathbf{f_P} = \lambda_2/\lambda_3$, respectively, where $\{\lambda_1, \lambda_2, \lambda_3\}$ are the eigenvalues of the point distribution around each point [52].

6) Point Feature Histogram (PFH):

The goal of the PFH formulation is to encode a point's k-neighborhood geometrical properties by generalizing the mean curvature around the point using a multi-dimensional histogram of values (Figure 4.10). The point feature histogram encodes the geometrical relationship between the points locations and their normals for each pair of points. This relationship is described by three angles, where each angle is binned into 5 intervals [83]. As a result, there will be 5×5×5 states for the relationships between each two points. This gives, a histogram with 125 bins which can be used to describe geometrical relationships in a neighborhood. This histogram is then subdivided into 5 groups of bins and, subsequently, the mean value of each

Table 4.1: The list of the classes in our classification system.

1 - Sign Poles: Light Blue	**6** - Grass and Soil: Dark Green
2 - Power Poles: Blue	**7** - Tree Trunks: Brown
3 - Guards: Yellow	**8** - Tree Branches: Pink
4 - Asphalt: Black	**9** - Leaves: Light Green
5 - Road Lines: Orange	**10** - Wires: Gray

partition is computed, which results in a feature vector with 5 features.

4.3.5.2 2D Features

All calculations related to the 2D features consider a 2D neighborhood block around the pixel of interest with a radius of **R** pixels.

1) Pixel Intensity, Mean and Standard Deviation:

The RGB intensities are regarded as three individual features. Mean and Standard Deviation of the RGB intensities of the pixels inside the block make up six more features.

2) GLCM (Gray-Level Co-Occurrence Matrix):

As explained in Section 3.2.1.2 the GLCM provides us with several measures of texture for the neighborhood block [38]. Energy, Contrast and Homogeneity features were extracted from this matrix.

3) Fourier Transform:

Fourier features are computed by applying masks to the Fourier Transform of the image blocks. These features reveal how much spatial detail is embedded in the block. Details are presented in Section 3.2.1.2.

4.4 Experimental Results

The whole system was implemented in C++ and MATLAB. 3D points were classified into 10 different classes listed in Table 4.1.

Table 4.2: The confusion matrix computed using a CRF applied to the test data. (Results are in percent and rounded).

	Sign Poles	Power Poles	Guards	Asphalt	Road Lines	Grass and Soil	Tree Trunks	Tree Branches	Leaves	Wires
Sign Poles	68	6	6	0	3	14	3	0	0	0
Power Poles	8	69	6	3	0	6	8	0	0	0
Guards	3	3	89	0	0	5	0	0	0	0
Asphalt	0	0	0	89	6	5	0	0	0	0
Road Lines	0	0	5	6	86	3	0	0	0	0
Grass and Soil	2	6	3	0	8	81	0	0	0	0
Tree Trunks	0	3	7	0	0	7	70	3	10	0
Tree Branches	0	0	0	0	0	0	0	100	0	0
Leaves	0	3	0	0	0	0	15	3	79	0
Wires	0	0	0	2	0	0	0	0	0	98

First of all, the raw point cloud data underwent an outlier removal step in which the sparse and noisy points were eliminated from the data. To evaluate our approach, 100 Ladybug images were randomly selected from both datasets and manually labeled. The corresponding 3D points were drawn out of the point cloud data by projecting them on the labeled Ladybug images and their labels were checked. Then $20,000$ points were chosen randomly (2,000 points from each class) as the data required for a cross-validation test. In addition, $3,000$ points were picked for training the CRF parameters.

Multi-view matched pixels of each point are found by projecting the 3D point to the nearby Ladybug images. Then, the features introduced in Section 4.3.5 were computed with radios of 8 for each of these pixels, resulting in several 2D feature vectors from different image views. The best 2D feature vector for representing each point was then identified by applying the method presented in Section 4.3.2. In total, each point had a set of 10 3D features which were directly obtained from the point cloud data and also 36 2D features that were extracted from the corresponding pixel of the 3D point on the proper image view . All these features were normalized to make them zero mean with unit variance.

Subsequently, the 20000 points were partitioned into three parts to perform a three-fold cross validation, which led to an average accuracy of 77.5%.

We also performed a single-view experiment in which only the closest image was considered for 2D feature extraction. The accuracy for the single-view experiment was 70.5%, which is considerably lower than the accuracy of the multi-view system.

Furthermore, we conducted experiments using different combinations of datasets and methods and compared them to our framework based on the multi-view data. As is illustrated in Figure 4.11, this new approach can significantly boost the performance of the classification system. This improvement is especially apparent for the classes of thin objects such as *traffic signs*, *power poles*, *road guards* and *power wires*. These categories are very vulnerable to misclassification due to, for instance, mis-registration between the 2D and 3D data. The mismatch between these datasets is further aggravated due to parallax effect of Lidar sensors or vibrations of moving surveying vehicle. Using the multi-view approach provides us with more reliable 2D-3D information about the objects, as confirmed by our results.

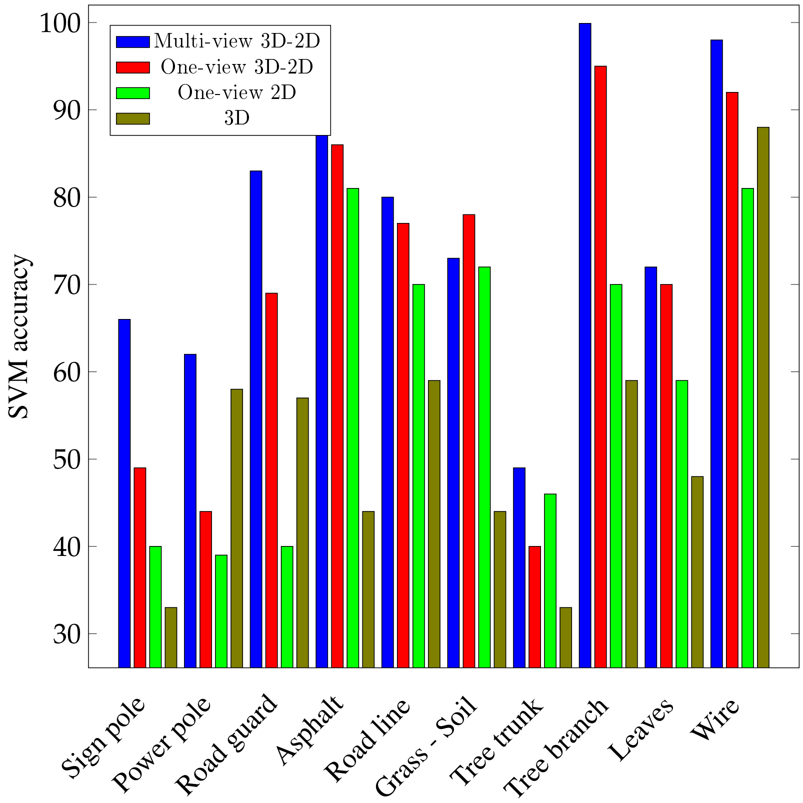


Figure 4.11: Comparison of the per-class SVM accuracies for the investigated methods. It can be seen that our multi-view approach can significantly enhance the performance of the classification system, especially for the category of thin objects like *sign pole*, *power pole* or *road guard.*

In the next step, the CRF parameters were trained using the 3000 points set aside before. For each point, the neighbors that had a maximum distance of **R**= 20cm were determined and the class labels of the points and their neighbors were predicted using the trained SVM classifier.

Subsequently, the trained CRF model was applied to the SVM results through the inference process, which resulted in an average accuracy of 82.9%. The confusion matrix for the CRF validation is shown in Table 4.2.

Table 4.3 illustrates a detailed comparison of the methods investigated in this work in terms of the accuracy and *F1*-score for each class. The *F1*-score, which is calculated by $\frac{2\times recall\times precision}{recall+precision}$, is informative if there is a large imbalance between the number of data in different classes.

Table 4.3: The accuracy and F1-Scores of the 3D classification using single-view, multi-view and CRF.

Class Acronyms		*SP*	*PP*	*RG*	*As*	*RL*	*G-S*	*TT*	*TB*	*Le*	*Wi*
Accuracy (%)	*single-view SVM (2D+3D)*	49	46	69	86	77	78	42	95	71	92
	multi-view SVM (2D+3D)	66	62	83	92	80	73	49	100	72	98
	multi-view SVM (2D+3D) + CRF	68	69	89	89	86	81	70	100	79	98
F1-Score	*single-view SVM (2D+3D)*	56	53	71	82	72	64	43	90	74	92
	multi-view SVM (2D+3D)	70	68	81	87	82	65	51	93	77	99
	multi-view SVM (2D+3D) + CRF	75	72	82	89	85	73	71	97	83	99

4.5 Summary

In this chapter, we devised a multi-view point cloud classification system using synchronous Lidar data and 360° panoramic images for the classification of outdoor scenes and objects. In contrast to a single-view image classification system, where there is only one corresponding 2D image for each 3D Lidar point, the multi-view system provides us with more information from different views. This brings many benefits into the 3D-2D object classification system. For example, in the single-view system, a misalignment between the 3D Lidar point and corresponding 2D image makes the 2D feature unrelated and useless. Besides, unlike indoor images, outdoor images typically suffers from undesired effects, such as color saturation or dark shadows which make some parts of the image barely informative. Employing a multi-view approach, all the captured images that convey some information about one Lidar point is considered, and the image which best represents the object is identified. It was shown in Section 4.4 that using our consensus multi-view technique can increase the classification power of the system from 70.5% to 77.5%.

Furthermore, the classification accuracy of the system improved from 77.5% to 82.9% using a CRF framework. The confusion matrix of the main classifier was given to the CRF as a clue to enhance the system based on its weaknesses. In this method, more constraints are put on the neighborhood of the cases that have large misclassification rates in the unary classifier. Moreover, the pairwise function was equipped with the class probabilities of the neighboring points and this added more cost to the cases where the neighboring class labels have a lower degree of certainty.

The data is a mixture of two datasets which are collected in winter and summer. Although vegetation coverage in the winter dataset is a bit sparse compared to the summer dataset, the system shows good robustness against this issue. Additionally, the dark scenes in the winter dataset has made this problem much more challenging. Despite that we in this chapter addressed some of the challenges of using multiple modalities, classification has been done just in 3D. Due to the inherent limitations of Lidar data, it is possible that many objects, especially moving ones are not captured adequately in this data. Therefore, classification of data in both 2D and 3D domains is desirable. In the next chapter we propose a multi-modal graphical model to label both 2D and 3D data simultaneously while taking advantage of information from other domains .

Chapter 5

A Multi-modal Graphical Model for Scene Analysis

In the previous chapter, we focused on 3D labeling using multi-view 2D information. In this chapter, we present a new classification framework which jointly leverages the information of 2D imagery and 3D Lidar data to label both 2D and 3D data, simultaneously. The ultimate goal is to obtain a semantic labeling of the entire set of image pixels and 3D points in multi-modal datasets. Existing methods often impose a hard correspondence between the 2D and 3D data, where the corresponding 2D and 3D regions are forced to receive identical labels. This results in performance degradation due to misalignments, 3D-2D projection errors and occlusions. We address this issue by defining a graph over the entire set of data that models soft correspondences between the two modalities. This graph encourages each region in a modality to leverage the information from its corresponding regions in the other modality to better estimate its class label. To demonstrate the ability of our model to support multiple correspondences for objects in the 3D and 2D domains, we introduce a new multi-modal dataset. This dataset which is publicly available contains panoramic images and 3D point cloud data captured from outdoor scenes (NICTA/2D3D Dataset). Below, we discuss the overview of our method followed by an introduction to our multi-modal graphical model in Section 5.2. Finally, we present our new dataset and experimental results.

5.1 Overview of Our Approach

Similarly to our work in the previous chapter, in most of the classification frameworks that utilize both 2D and 3D information [112, 116, 59], appearance and shape features are extracted from corresponding objects and elements (e.g., regions) in the two domains and then inference is performed in only one of the domains. This makes these approaches only suitable to scenarios where the objects and regions are seen simultaneously in both modalities. In other words, objects that are not captured in one of the modalities are eliminated from the classification process. Furthermore, finding valid and accurate correspondences between the elements of these modalities is a challenging task, due to the difference in the nature of the captured data and the inevitable misalignment between the modalities. This may lead to an association of 2D and 3D features that truly belong to different objects.

There are only a few works that come closer to treating both modalities separately. In particular, Cadena *et al.* [19] employed a Conditional Random Field (CRF) to perform 2D-3D semantic labeling on 2D RGB images and 3D Lidar data. They initially designed a graph based on 2D nodes (representing superpixels). Then, for each 3D segment, they found its corresponding 2D superpixel and represented them jointly with one single node, as in the aforementioned works. As a result, their design still suffers from similar issues as described above. In contrast, Munoz *et al.* [72] explicitly introduced separate nodes for the 2D and 3D regions and tackled the correspondence problem using an inter-domain overlap function. To be able to handle the uncommon nature of their model, they had to design a specialized co-inference technique, which relies on hierarchical segmentations in both domains and alternates between 2D and 3D labeling. Importantly, they only evaluated their approach on a dataset where one-to-one correspondences between 2D and 3D were available.

In this chapter, we address the problem of joint 2D-3D outdoor scene analysis by proposing a graphical model in which each region in 2D (superpixel) or 3D (3D segment) is assigned a separate node. The strength of the pairwise link that connects a 2D and a 3D node is adjusted according to the amount of overlap between the 3D segment projected onto the image plane and the 2D image superpixel. The benefits of this representation are threefold. First, it allows us to account for 2D or 3D regions that have no correspondence in the other modality. Second, when a correspondence between a 2D and a 3D region exists, specifying separate nodes addresses the problems that arise because of inaccurate 2D-3D registration and projection. Finally, our representation lets us model the fact that several superpixels (e.g., from different images) may correspond to a single 3D segment. This yields richer appearance information for the segment and makes inference more reliable. We evaluate our approach on the CMU/VMR urban image + Lidar dataset [72] and show the superiority of our method over the state-of-the-art [72].

Furthermore, we release a dataset of panoramic images and 3D point cloud data captured from outdoor scenes (the NICTA/2D3D Dataset[1]). In contrast to the dataset in [72] where all the 3D points have a one-to-one correspondence with 2D image pixels and other points

[1] Publicly available at http://www.nicta.com.au/computer_vision_datasets.

are removed, our data includes the entire set of 3D points which provides naturally occurring many-to-one relationships. Furthermore, for our purpose of multi-modal semantic segmentation, the NICTA/2D3D dataset has the advantage over the KITTI dataset [34] that the point cloud data has both a large vertical and horizontal Field of View (FOV) and is seen from multiple images, thus providing an opportunity to establish correspondences between 3D points and imagery from a large number of view points. This enables research on methods necessary to resolve issues such as correspondence ambiguities, occlusions (either spurious or due to parallax) and missing 2D-3D correspondences. We therefore make use of the NICTA 2D/3D dataset to demonstrate the effectiveness of our method at handling these issues.

5.1.1 Limitations of Previous Works on using Multiple Modalities

Semantic scene analysis has been an important problem in computer vision for the past decade. In particular, scene parsing from 2D imagery has been intensely studied, yielding increasingly accurate results [92, 112, 57, 35, 114, 49]. With the advent of 3D depth sensors, such as laser range sensors (Lidar) [42, 100] and RGB-D cameras (e.g., Kinect) [25, 91, 37, 15], it seems natural to leverage these additional sources of information to reach even better levels of scene understanding [76, 10, 88]. This information can be encoded using graphical models. In particular, CRFs have often been used to model the contextual information of the scene in the form of pairwise and higher order potentials between pixels or superpixels in images, and 3D points (voxels) or segments in 3D data. Pairwise graphical models have been studied in many semantic segmentation problems on 2D or 3D datasets [79, 30, 112, 88, 10, 66]. Due to the limitation of the pairwise models at describing complex contexts in the scene, higher-order graphical models have been used in some works to account for more complex relationships among the objects and elements [49, 55]. However, as the size of the data and number of nodes in the graph grow, the inference process of higher-order models becomes much more time-consuming.

Only a few methods have proposed to jointly exploit 2D and 3D in graphical models for outdoor scene analysis [79, 30, 19]. In [79], the authors proposed a probabilistic approach to label the objects in an urban environment using both laser data and imagery. 3D surface normal features in conjunction with 2D color, texture and geometric features were used to first segment the objects and then classify them into *pavement*, *dirt path*, *smooth wall*, *textured wall*, *vehicle*, *foliage* and *grass*. Douillard *et al.* [30] designed a rule-based system using 3D Velodyne Lidar data and monocular color imagery to classify the urban environment into 16 different classes.

The major limitation of these works is that the graph is defined over either the 2D domain or the 3D domain and there is no connection between the 2D and 3D nodes. Cadena *et al.* [19] designed a graph where the 2D superpixels and 3D segments which correspond to each other (according to the 3D-2D projection map), were jointly assigned one single node. Then the 2D features were augmented with the 3D features to represent the feature vector of this node in the graph. However, since perfect correspondences between 2D and 3D are assumed, this approach cannot handle many-to-one correspondences, or account for misalignments between the two modalities. Note also that only the 2D results are provided in [19].

Munoz *et al.* [72] assigned separate nodes to 2D superpixels and 3D segments and presented a correspondence function to find the degree of overlap between 2D and 3D nodes and to determine how much they influence each other in the inference process. They also presented a new co-inference technique based on hierarchical segmentations in both the 2D and 3D domains. In their framework, classification was performed in each level of the hierarchy for each domain and the results were transferred to the next hierarchy levels in both domains as a set of features. This approach was repeated through an iterative back-and-forth process over both modalities. Unfortunately, this non-standard model and inference technique make it difficult to generalize the approach to other problems. Furthermore, their method was only demonstrated on a dataset where every 3D point had only one corresponding image pixel.

Our formulation addresses the above-mentioned issues regarding 2D-3D correspondence and is based on a standard inference method, which makes it easy to apply to other similar problems.

5.2 A Multi-modal Graphical Model

In this section, we introduce our approach to joint semantic segmentation of 2D panoramic images and 3D point cloud data captured using a Lidar system. In particular, we consider the scenario where the visual information of an outdoor scene is recorded into F panoramic frames and one 3D point cloud. The ultimate goal is to find the most probable class label for the pixels in the images and 3D points in the point cloud data. In this section, we explain our model which is defined jointly over the 2D and 3D domains.

Given the large size of the point cloud (around $1,000,000$ points) and panoramic images (2000×4000 pixels), it is computationally very demanding to perform inference in a graphical model defined over the entire set of points and pixels. Instead, we build our model using image superpixels and 3D segments as the nodes of the graph.

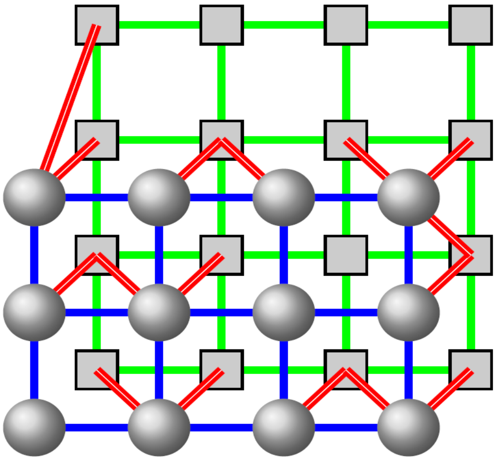


Figure 5.1: The graphical model in our approach. 2D superpixels are represented by squares and 3D segments are represented by spheres. The blue edges connect 3D segments, green edges link 2D superpixels and double lines (in red) associate the corresponding 2D and 3D nodes. 2D and 3D nodes can be connected to each other, depending on their neighborhood condition and also the 2D-3D projection.

Here, we propose a full model in which the entire set of 2D superpixels and 3D segments are accounted for in one graph. Figure 5.1 illustrates our graphical model. In this figure, squares represent superpixels and spheres represent 3D segments and various types of connections between 2D and 3D nodes are illustrated. Note that some 3D nodes are connected to more than one 2D node, whereas others have no connection with the 2D domain at all.

Let $\mathbf{y}^{2D} = \{y_{ij}^{2D}\}$, $1 \leq i \leq F$, $1 \leq j \leq N_i$, be the set of variables encoding the labels of the 2D nodes in F frames, where frame i contains N_i 2D regions. Similarly, let $\mathbf{y}^{3D} = \{y_i^{3D}\}$, $1 \leq i \leq M$, be the set of variables encoding the label of M 3D nodes. Each of these variables, either 2D or 3D, can take a label in the set $\mathcal{L} = \{1, \cdots, L\}$. We define the joint distribution over the labels given the features $\mathbf{x}^{2D} = \{\mathbf{x}_{ij}^{2D}\}$ and $\mathbf{x}^{3D} = \{\mathbf{x}_i^{3D}\}$ as

$$P(\mathbf{y}^{2D}, \mathbf{y}^{3D} | \mathbf{x}^{2D}, \mathbf{x}^{3D}) = \frac{1}{Z} \cdot \tag{5.1}$$
$$\exp\Big(- \sum_{i=1}^{F} \sum_{j=1}^{N_i} \Phi_{ij}^{2D} - \sum_{i=1}^{M} \Phi_i^{3D} - \sum_{i=1}^{F} \sum_{(j,k)\in\mathcal{E}_i^{2D}} \Psi_{ijk}^{2D}$$
$$- \sum_{(i,j)\in\mathcal{E}^{3D}} \Psi_{ij}^{3D} - \sum_{i=1}^{F} \sum_{(ij,k)\in\mathcal{E}^{2D-3D}} \Psi_{ijk}^{2D-3D} \Big),$$

where Z is the partition function. Φ^{2D} and Φ^{3D} denote the unary potentials of the 2D and 3D nodes, respectively. Ψ^{2D}, Ψ^{3D} and Ψ^{2D-3D} denote pairwise potentials defined over the sets of edges $\mathcal{E}^{2D}$, $\mathcal{E}^{3D}$ and $\mathcal{E}^{2D-3D}$, respectively. This probability distribution consists of different potentials detailed below. There are weighting parameters for these potentials indicating their contribution in inference. These parameters are adjusted via a validation process. Since, in the next chapter, we present *Learned potentials* (whose parameters are learned during the training process), the potentials in this chapter are called *Handcrafted potentials*.

5.2.1 Handcrafted Potentials

We build the potential functions in Equation 5.1 such that they could intuitively model the correlation between the class probabilities and local information of each node (unary potentials), as well as the contextual relationships between the pairs of adjacent nodes in the graph (pairwise potentials).

5.2.1.1 2D Unary Potential

This potential indicates the cost of assigning label y to the j^{th} superpixel in the i^{th} image, given its features, x_{ij}. The $2D$ notation clarifies that this function operates in the 2D domain. The potential function Ψ^{2D} is computed as the negative logarithm of the class probabilities obtained by an SVM classifier.

5.2.1.2 3D Unary Potential

As for the 2D superpixels, the negative logarithm of the SVM class probabilities are taken as the potential function Ψ^{3D}.

5.2.1.3 2D Pairwise Potential

This potential function is defined over all pairwise edges between adjacent superpixels. $\mathcal{E}^{2D}$ is the collection of all 2D pairwise edges in frame i, which is generated using the method introduced in [75]. The potential function Ψ^{2D} is defined in a way that penalizes dissimilar class labels for two adjacent superpixels if their RGB values are very close. It can be expressed as

$$\Psi^{2D}\Big(y_1, y_2, x_1, x_2\Big) = \frac{\delta(y_1 \neq y_2)}{1 + a\|\mathrm{RGB}_{\mathrm{x}_1} - \mathrm{RGB}_{\mathrm{x}_2}\|_1} \,. \tag{5.2}$$

This potential equals zero (via the delta indicator function) if the pair of superpixels have identical class labels. a is the weight of the RGB contrast which is determined using cross-validation ($a = 0.05$).

5.2.1.4 3D Pairwise Potential

$\mathcal{E}^{3D}$ denotes the collection of pairwise edges between 3D segments. We consider every pair of segments whose minimum inter-point distance is lower than a threshold ($D_t = 1$m) to be a neighbor and constitute a pairwise connection. The potential function Ψ^{3D} is computed as

$$\Psi^{3D}\Big(y_1, y_2, x_1, x_2\Big) = \frac{\delta(y_1 \neq y_2)}{1 + b|\theta_{x_1} - \theta_{x_2}|} \,. \tag{5.3}$$

where θ is the angle between the direction of the average normal vector of the 3D segment and the vertical axis. The weight b is optimized by cross-validation ($b = 1/90$).

5.2.1.5 2D-3D Pairwise Potential

We applied this potential to the edges that connect 2D nodes to 3D nodes. Since the outdoor scene is captured using a panoramic RGB camera and a 360° laser scanning system, many objects and regions can be observed in both data modalities. The pairwise potential Ψ^{2D-3D} takes the relationships between the 2D objects and their 3D counterparts into account by considering pairwise links between them. To find all the pairwise links between the 2D and 3D domains, the point cloud is projected onto the image planes. As a result, the projection of each 3D segment may intersect with zero, one or more superpixels in the images. Some segments may also be observed in more than one panoramic image. The list of the entire set of 2D-3D pairwise links is recorded into $\mathcal{E}^{2D-3D}$ and the potential function Ψ^{2D-3D}, which is computed as

$$\Psi^{2D-3D} = w_{ij,k}\delta(y_1 \neq y_2) \,, \tag{5.4}$$

is applied to all of them. In this function, $w_{ij,k}$ is the 2D-3D overlap weight and is calculated as follows. The size of the overlap between the projected 3D segment and each of its corresponding superpixels is computed and normalized with respect to the size of the superpixels. This yields an overlap weight vector $\mathbf{w} = [w_1, w_2, \ldots, w_t]$ for each 3D segment, where t is the number of overlapping superpixels. Figure 5.2 illustrates how this overlap weight vector is computed where a 3D segment in the point cloud data has some projection overlap with the superpixels in two different images. In this figure, suppose that we have $\frac{P_1 \cap A}{A} = 0.2$, $\frac{P_1 \cap B}{B} = 0.8$, $\frac{P_2 \cap C}{C} = 0.6$ and $\frac{P_2 \cap D}{D} = 0.6$. This yields the overlap weight vector $\mathbf{w} = [0.2, 0.8, 0.6, 0.6]$. As a result, the superpixel with maximum size of overlap impacts the 3D segment more than others (due to a larger weight for its 2D-3D pairwise link).

Note that, due to the imperfection of the 2D segmentation, there might be some cases

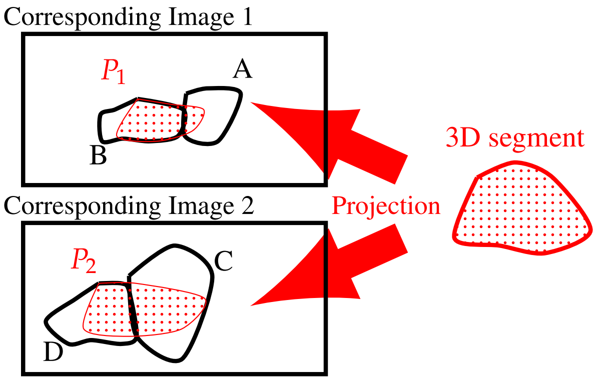


Figure 5.2: An example that illustrates how the corresponding superpixels of each 3D segment and their 2D-3D pairwise weights are determined. 3D segment is projected onto its nearby image planes and the superpixels that have a significant overlap with its projection are considered and the weight of pairwise links are determined according to the degree of overlap and size of the superpixels.

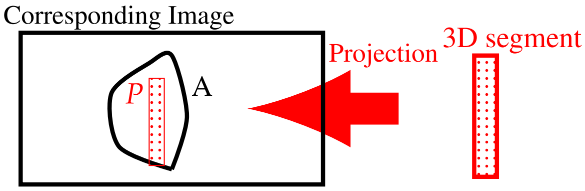


Figure 5.3: An example that justifies the need for a second step normalization on 2D-3D pairwise weight vector. Since the object is very thin and the ratio of $\frac{P \cap A}{A}$ weakens the strength of the pairwise link between the 3D segment and its only counterpart superpixel in the image, the ratio should be normalized w.r.t. the size of the overlap.

where the projection of thin or small objects (like *poles* or *wires*) is surrounded by only one large superpixel (Figure 5.3). As a consequence, the normalized overlap weight for these cases becomes very small which makes the impact of the 2D-3D pairwise potential for them negligible. To overcome this issue, the overlap weights are normalized again by dividing them by the maximum weight among all corresponding superpixels of each 3D segment. For instance, the weight vector $\mathbf{w}$ which was computed for Figure 5.2 is normalized by dividing all its components by their maximum value of 0.8 which results in $\mathbf{w} = [0.25, 1, 0.75, 0.75]$.

The potentials introduced in Section 5.2.1 form the total probability distribution in Equation 5.1 by taking five weighting parameters indicating their contribution in inference. These parameters are then adjusted through a validation step to produce the lowest total error rate on the validation data.

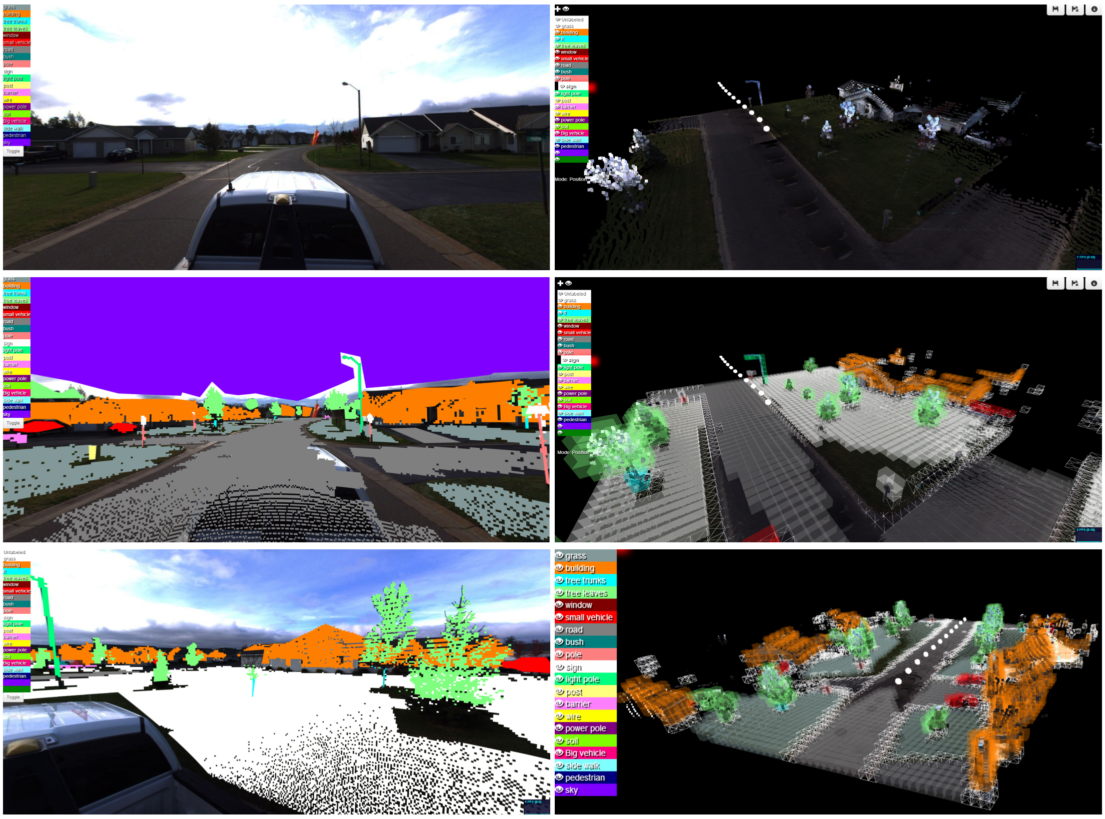


Figure 5.4: Manual annotation of the 3D point clouds and 2D images. 1st column: Some screenshots from the 2D labeller program. 2nd column: Some screenshots from the 3D annotator program.

5.3 NICTA/2D3D Dataset

In this section, we first review two existing multi-modal 2D+3D datasets and discuss their problems as benchmarks for semantic segmentation tasks. Then we present a new multi-modal dataset in which those problems have been addressed.

KITTI [34] is probably the largest publicly available multi-modal dataset which is mainly used for object detection tasks. The main problem with KITTI is the small vertical FOV of the point cloud data. As shown in [19], a large portion of the images in this dataset does not correspond to any laser scanning data. As a consequence, the result of 2D labeling relies mostly on image data rather than on multi-modal 2D+3D data.

CMU/VMR [72] is another multi-modal dataset which is composed of wide RGB images in conjunction with 3D point cloud data, collected from 372 urban scenes. One issue with this dataset is that the 3D point cloud data is not complete and that points with no corresponding image pixel are removed. The 3D points are annotated by back-projecting the labeling of the annotated images. However, due to the inaccurate 2D-3D back-projection, the ground truth of some of the 3D points is incorrect. Figure 6.10 shows two samples from these datasets that show incorrectly a building and tree leaves are labeled as a pedestrian and a big vehicle.

In this work, we present a new multi-modal dataset (NICTA/2D3D dataset) for outdoor scene understanding which consists of a synchronous series of 2D panoramic imagery and 3D Lidar point cloud. It contains 12 outdoor scenes and each scene includes an extended block of 3D point cloud along with several panoramic images. The number of 3D points in the scenes varies from 1 to 2 millions and depending on the size of the point cloud block, each scene contains between 10 and 20 panoramic images.

The dataset was manually annotated in the 3D domain using a 3D annotator (Figure 5.4) and the ground truth labeling of the panoramic images were obtained via the 3D-2D projection of the 3D labels. The 2D ground truth images were later checked and retouched to produce a more precise 2D ground truth (Figure 5.4). This step accounts for projection errors due to misalignments or parallax and also deals with the moving and reflective objects whose point cloud data is very sparse. Additionally, *Sky*, which does not exist in the 3D data was added as a new label in the 2D images. The 360° panoramic images cover the whole FOV of the point cloud data and not just a portion of it as in the KITTI dataset [34]. In contrast to the dataset in [72], where all the 3D points have a correspondence with 2D images and other points are removed, NICTA/2D3D data includes the entire set of 3D points.

The second advantage of NICTA/2D3D dataset over the aforementioned datasets is that the panoramic images provide the chance of capturing each object several times in different frames and view-angles. Therefore it not only provides the corresponding 2D information for each 3D segment, but does it several times from different views. We benefited from this property in our graph and connected each 3D node to all of its corresponding 2D nodes in different frames. As a result, each 3D segment can leverage the appearance features of the scenes from several views and each 2D region can utilize the 3D information as well as the 2D information of other superpixels (in other panoramic image frames), which are indirectly connected to it in the graph via its corresponding 3D node.

5.4 Experimental Results

In this section, we first describe the steps required to compute the unary potentials (3D and 2D), and then discuss our experiments on two datasets in details.

5.4.1 3D Features and Unary Potentials

We extracted the following 3D shape features from the point cloud data: fast point feature histograms that describe the local point distributions based on the point distances and orientations of their surface normal vectors w.r.t. each other, eigenvalue features that model the shape of the spatial distribution of the points, deviation of the surface normal vectors from the vertical axis, and also the height of the points. The 3D segments were obtained from these features by first classifying them using an SVM classifier, partitioning the points into different groups given their class labels, and then performing k-means clustering on each group of the points based on their spatial coordinates. We then further leveraged the SVM results and used the negative logarithm of the multi-class SVM probabilities as features in our unary potentials. The probabilities for a segment were obtained by averaging over the points belonging to the segment. We also used three eigenvalue descriptors and the vertical-axis deviation as additional features for the segments.

We used a probabilistic SVM classifier for training using the extracted features. We classified the 3D points into one of the L pre-defined class labels and calculated the probability of belonging to each class for each point. We then separated the points into L different groups according to their class labels and performed a k-means segmentation to each group to further divide them into our final 3D segments. The class probabilities of each 3D segment was calculated by averaging over the class probabilities of its points. Finally, the unary potentials of the 3D segments were computed by taking the negative logarithm of their class probabilities, as discussed in Section 5.2.1.2.

5.4.2 2D Features and Unary Potentials

As 2D regions, we used superpixels extracted by the mean-shift algorithm [24]. We utilized histogram of SIFT features [68], GLCM features (entropy, homogeneity and contrast, each computed in both horizontal and vertical directions), and RGB values to train an SVM classifier, and used the negative logarithm of the SVM probabilities as features in our unary potentials. We augmented these features with six GLCM features and three RGB features. These features were used to train a probabilistic SVM classifier and predict the class probabilities of the superpixels in the test images. The 2D unary potentials for each superpixel was then computed as the negative logarithm of the class probabilities (see Section 5.2.1.1).

Table 5.1: The F1-scores of the 2D classification for the CMU/VMR dataset ([72]) using our model with handcrafted potentials, compared to the method of [72]. The results of the 2D-only model with handcrafted potentials are provided as well for comparison.

	Road	Sidewalk	Ground	Building	Barrier	Bus stop	Stairs	Shrub	Tree trunk	Tree top	Small Vehicle	Big vehicle	Person	Tall light	Post	Sign	Utility pole	Wire	Traffic Signal	avg
Unary	95	81	75	56	29	17	32	50	31	53	32	49	29	16	15	**16**	33	41	29	41
Pairwise 2D	90	80	76	65	31	18	25	53	33	61	57	54	32	**17**	**16**	14	34	**44**	23	43
Munoz [72]	**96**	**90**	70	83	50	16	33	62	30	86	**84**	50	47	2	9	**16**	14	2	17	45
Ours	89	77	79	76	44	**23**	41	56	29	86	72	23	41	12	2	11	**36**	40	**30**	46

Table 5.2: The F1-scores of the 3D classification for the CMU/VMR dataset ([72]) using our model with handcrafted potentials, compared to the method in [72]. The results of the 3D-only model with handcrafted potentials are provided as well for comparison.

	Road	Sidewalk	Ground	Building	Barrier	Bus stop	Stairs	Shrub	Tree trunk	Tree top	Small Vehicle	Big vehicle	Person	Tall light	Post	Sign	Utility pole	Wire	Traffic Signal	avg
Unary	70	49	62	67	34	2	19	26	11	67	34	4	13	2	0	1	2	0	0	24
Pairwise 3D	70	48	63	67	34	2	23	26	11	67	36	6	16	2	0	1	2	0	0	25
Munoz [72]	82	73	68	87	46	11	38	63	28	**88**	73	**56**	26	**10**	0	0	0	0	0	39
Ours	89	77	79	87	56	**18**	57	59	**31**	84	70	24	46	3	**3**	**8**	**22**	**10**	**15**	44

5.4.3 Experimental Results on CMU/VMR

The CMU/VMR dataset consists of 372 urban scenes, each of which was surveyed using an RGB camera and a laser scanner. We divided the dataset into 4 non-overlapping folds, one of them used for validation and the others used for 3-fold cross-validation. The CRF parameters were obtained from the validation data, and TRW [50] was chosen as inference method. We use the F1-score as the performance measure in order to facilitate the comparisons with [72]. On average, we achieve $F1^{2D} = 0.46$ for the semantic segmentation of the images and $F1^{3D} = 0.43$ for the semantic segmentation of the 3D point cloud, which are higher than the results of [72] who reports $F1^{2D} = 0.45$ and $F1^{3D} = 0.39$. Table 5.1 and Table 5.2 show our quantitative 2D and 3D classification results per class, compared to the results of [72].

Since the ground truth of the 3D data in this dataset is not as reliable as the 2D ground truth (as described in Section 5.3), and also because the 3D point cloud is not as dense as it could be (due to the removal of the 3D points with no pixel correspondence), the results of the 3D semantic labeling is lower than that of 2D semantic segmentation. This issue is even more critical for the last five categories in Table 5.2, which are all thin and small objects and thus influenced the most by projection errors. Nevertheless, as indicated in Table 5.2, our model considerably improves the performance of the 3D labeling for these classes. The average F1-score of the 3D labels has also been improved from 25% to 44%, which demonstrates the benefits of our model. Figure 5.5 depicts the qualitative results for a sample scene in this dataset, compared to the ground truth.

5.4.4 Experimental Results on NICTA/2D3D

For our new dataset, we used the same experimental setup (number of folds, parameter selection, inference method, performance measure) as in the previous experiment on the CMU/VMR dataset. The average F1-scores of our model are $F1^{2D} = 0.45$ and $F1^{3D} = 0.52$, which outperform the results of the unary classifier ($F1^{2D} = 0.38$ and $F1^{3D} = 0.43$).

Table 5.3 and Table 5.4 evidence that the performance of the system in each domain has been improved by incorporating information from the other domain via 2D-3D pairwise links. In particular, our model has increased the classification rate of the classes which had a small number of training samples, by exploiting the 2D-3D multi-correspondence. Note that 2D-3D pairwise edges can add inter-domain semantic information while they do not yield over-smoothing. As evidenced by Table 5.3, the 2D pairwise edges were unable to recover any mis-classified objects in the *Post* and *Barrier* categories. Nonetheless, our model has improved the classification rate for these classes by 8% and 5%, respectively.

Table 5.3 and Table 5.4 show that, except for two classes (*Road* and *Sidewalk*), the other classes have had a significant improvement in their F1-score in at least one of the 2D or 3D datasets. This is mainly because the 3D information of *Road* and *Sidewalk* are very similar, and the 2D-3D pairwise edges are not effective enough to correct the mis-classification between them in the 3D data. As can be seen in Table 5.3, this mis-classification has been transferred to the 2D domain via 2D-3D pairwise edges and has deteriorated the F1-scores of *Sidewalk* and *Road* in the 2D dataset. The qualitative results of the 2D and 3D semantic labeling in two different scenes of the NICTA/2D3D dataset are illustrated in Figure 5.6.

Table 5.3: The F1-scores of the 2D classification for NICTA/2D3D dataset using our model with handcrafted potentials. The results of the 2D-only model with handcrafted potentials are provided as well for comparison.

	Grass	**Building**	**Tree trunk**	**Tree leaves**	**Vehicle**	**Road**	**Bush**	**Pole**	**Sign**	**Post**	**Barrier**	**Wire**	**Sidewalk**	**Sky**	**avg**
Unary	80	33	14	80	49	95	16	28	3	0	0	29	15	98	38
Pairwise 2D	82	38	15	78	53	90	16	30	5	0	0	31	49	97	42
Ours	84	50	**18**	83	64	93	24	31	**10**	**5**	9	33	30	**99**	45

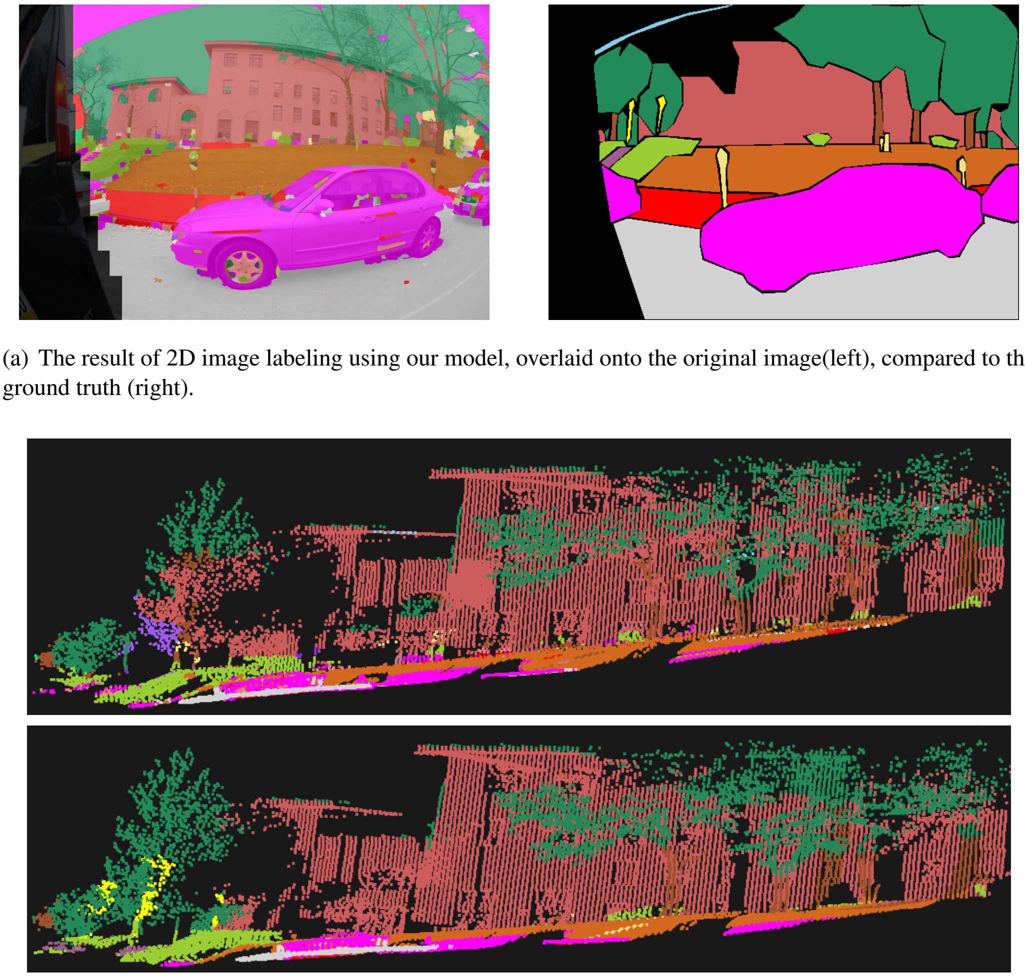
Table 5.4: The F1-scores of the 3D classification for NICTA/2D3D dataset using our model with handcrafted potentials. The results of the 3D-only model with handcrafted potentials are provided as well for comparison.

	Grass	Building	Tree trunk	Tree leaves	Vehicle	Road	Bush	Pole	Sign	Post	Barrier	Wire	Sidewalk	Sky	avg
Unary	52	61	27	87	58	**82**	10	24	19	43	19	74	0	#	43
Pairwise 3D	54	81	35	94	60	61	20	40	33	36	22	88	0	#	48
Ours	58	81	36	96	66	74	24	38	30	**45**	24	**88**	0	#	51

5.5 Summary

In this chapter, we have proposed a graphical model that enabled us to perform joint inference on two different modalities of data (2D imagery and 3D point cloud) and improve the semantic labeling in both modalities. We have incorporated 2D-3D pairwise edges in the graph (in addition to 2D-2D and 3D-3D edges) that connect corresponding 2D nodes and 3D nodes and transfer information from one modality to the other. Although these pairwise connections utilize information from both the 2D and 3D domains to enhance the labeling of corresponding superpixels and 3D segments, they do not force these corresponding nodes to be assigned identical class labels, which is beneficial in the presence of projection errors. As a result, such pairwise connections do not cause over-smoothing and improve the performance of the system, especially for small and narrow classes. Our experiments have evidenced that we outperform the state-of-the-art on a publicly available dataset. Furthermore, we have introduced a new publicly available multi-modal dataset, which addresses the problems of the existing datasets in terms of correspondence between the 2D and 3D domains. Our model can be applied to data with other modalities, given that connections between the data modalities are well-defined.

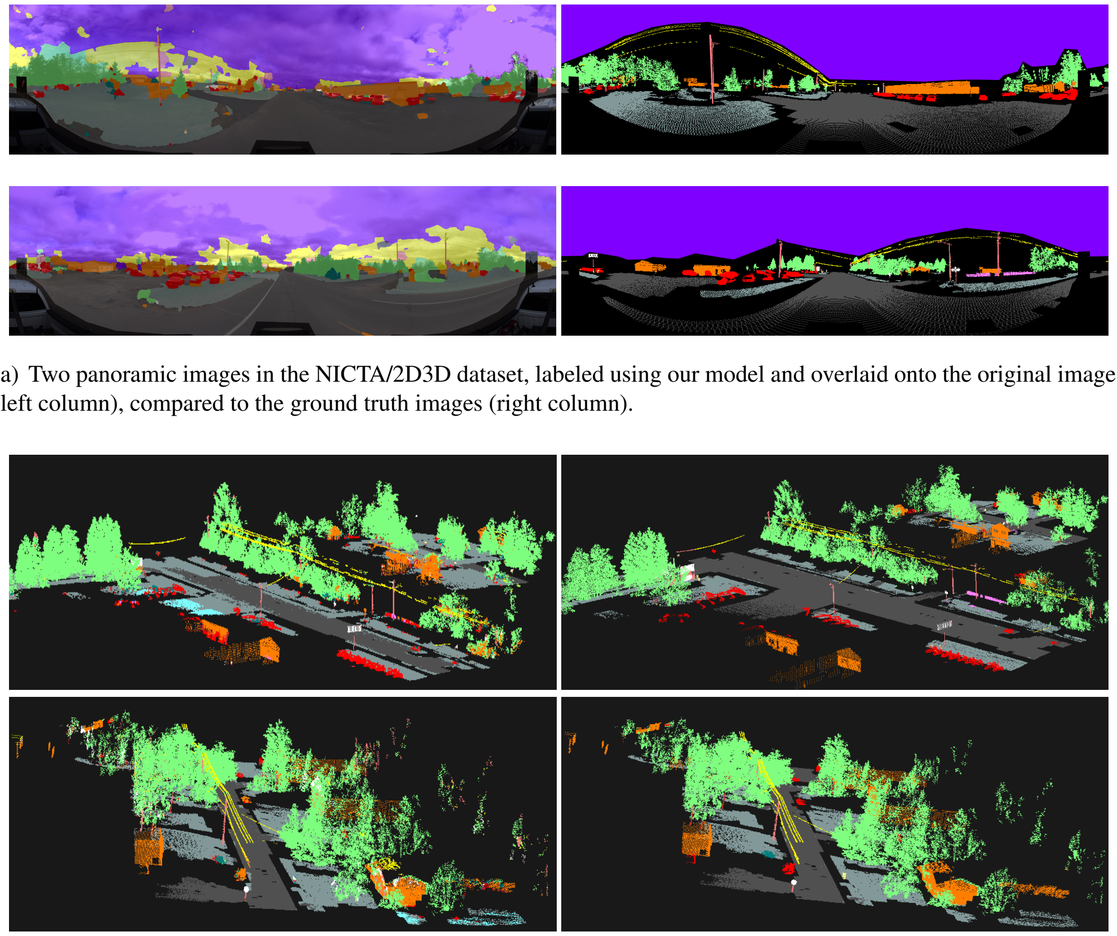
In the next chapter, we introduce latent nodes to address the problems of data misalignment and label inconsistencies in multi-modal data. To eliminate the need for hand tuning the parameters of our model, we propose to instead learn potential functions from training data.



(a) The result of 2D image labeling using our model, overlaid onto the original image(left), compared to the ground truth (right).

(b) The result of the 3D point cloud semantic labeling (above), compared to the ground truth (below).

Figure 5.5: The qualitative results of our proposed model for semantic segmentation of (a) 2D image and (b) 3D point cloud, captured from a scene in CMU/VMR dataset. The color codes for this figure are: White=Road, Brown=TreeTrunk, Light-Red=Building, Green=TreeTop, Light-Green=Shrub, Pink=Vehicle, Red=Sidewalk, Orange=Ground, Yellow=Utility pole



(a) Two panoramic images in the NICTA/2D3D dataset, labeled using our model and overlaid onto the original images (left column), compared to the ground truth images (right column).

(b) Two point cloud blocks in the NICTA/2D3D dataset, labeled using our model (left column), compared to their ground truth (right column).

Figure 5.6: The qualitative results of our proposed model for semantic segmentation of (a) 2D images and (b) 3D point cloud, captured from two different scenes in the NICTA/2D3D dataset. The color codes for this figure are: Dark-Gray=Road, Orange=Building, Green=Leaves, Red=Vehicle, Blue=Sidewalk, Gray=Grass, Light Pink=Pole, Purple=Sky, Yellow=Wire.

Chapter 6

Soft Correspondences in Multi-modal Scene Parsing

In the previous chapter, we proposed a multi-modal graphical model that can leverage the potential of multiple modalities simultaneously, and the labeling of each modality can be enhanced using the information of other sensing modalities. In this chapter, to better address the problems of data misalignment and label inconsistencies between modalities, we introduce latent nodes to explicitly model inconsistencies between modalities. These latent nodes allow us not only to leverage information from both domains to improve the labeling of the modalities, but also to cut the edges between inconsistent regions. To eliminate the need for hand tuning the parameters of our model, we propose to learn intra-domain and inter-domain potential functions from training data. We expand the model to handle more modalities (Figure 6.1). In order to highlight the benefits of the geometric information and the potential of our method in simultaneous 2D/3D semantic and 2D/3D geometric inference, we performed simultaneous inference of semantic and geometric classes both in 2D and 3D that led to satisfactory improvements of the labeling results in both datasets. First, an overview of our proposed method and some related works are given. Then our multi-modal with learned potentials is presented. After that, we explain our proposed latent nodes to address the misalignment between modalities. Finally we present two especial cases and experimental results.

6.1 Overview of Our Approach

As discussed in the previous chapter, due to the inherent misalignments between the domains and also the dissimilarities in the classes of different modalities, these modalities should be either studied separately, or connected such that each one of them could simultaneously utilize the incoming information of other modalities correctly. To this end, as illustrated in Figure 6.2,

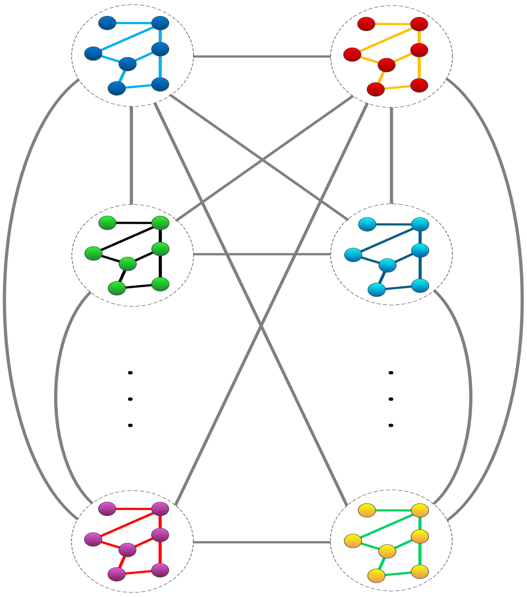


Figure 6.1: The proposed multi-modal graphical model. The dots represent the nodes of more modalities, the intra-domain connections are represented by colored lines and the inter-domain connections are denoted by gray lines. The latent nodes exist between each inter-modality connection, though they have not been illustrated in this figure to avoid any confusion.

we introduce latent nodes to handle conflicting evidence between the different domains. The benefit of these latent nodes is twofold: First, they can leverage information from both domains to improve their respective labeling. Second, and maybe more importantly, these nodes allow us to cut the edges between regions in different modalities when the local evidence of the domains is inconsistent. As a result, our approach lets us correctly assign different labels to the modalities. In our formulation, different modalities can cover different sets of class labels and still leverage the information of other modalities to enhance the performance of the scene parsing system.

More specifically, each connection between two domains is encoded by a latent node, which can take either a label from the same set as the regular nodes, or an additional label that explicitly represents a broken link. We then model the connections between the latent nodes and the different modalities with potential functions that allow us to handle inconsistencies. While many such connections exist, they come at little cost, because the only cases of interest are when the latent node and the regular node have the same label, and when the latent node indicates a broken edge. By contrast, having direct links between two modalities would require to consider potential functions for each combination of two labels (i.e., for L labels, L^2 vs $2L$ in our model). The connections between the modalities that do not have identical label spaces

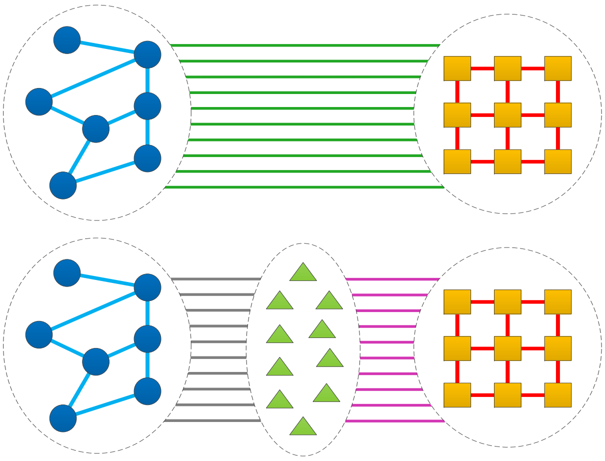


Figure 6.2: **Top:** Existing approaches typically directly connect corresponding regions in different modalities and penalize these regions for taking different labels, thus producing wrong labeling in the presence of data misalignment, or other causes of label disagreement. **Bottom:** Here, we introduce latent nodes that are placed between each connected pair of 2D and 3D nodes in the graph. They explicitly let us account for such inconsistencies, and potentially cut edges between the different domains. Circles denote the nodes in one domain (e.g., 3D) and squares denote the nodes in another domain (e.g., 2D). The latent nodes are depicted by triangles.

are also governed by the latent nodes which have access to the features of both modalities. If these features match, the latent nodes then take the class labels that are consistent with the labels of the nodes at two ends of their respective connections. For example, the class Grass for a latent node is consistent with both a horizontal plane in one modality and Grass in another one. However, in case of a mis-match between the features of two modalities, the latent node breaks the link between them.

In the previous chapter, our multi-modal graphical model relied on a Pott's model as pairwise potentials for both intra-domain and inter-domain edges. As a consequence, it also implicitly attempts to assign the same label to the corresponding nodes in each modality. Here, by contrast, we propose to learn the intra-domain and inter-domain relationships from training data. Learning the parameters of CRFs for semantic labeling has been tackled by a number of works, such as [108, 53] with mean-field inference, [62] with TRW, and [82] with loopy belief propagation. Of more specific interest to us is the problem of learning label compatibility, as studied by [53] for 2D images and by [52] for 3D data. Here, we consider label compatibility within and across domains. We make use of the truncated tree-reweighted (TRW) learning algorithm of [27]. To the best of our knowledge, this is the first time such a learning approach is employed for multi-modal scene parsing. The resulting method therefore incorporates local

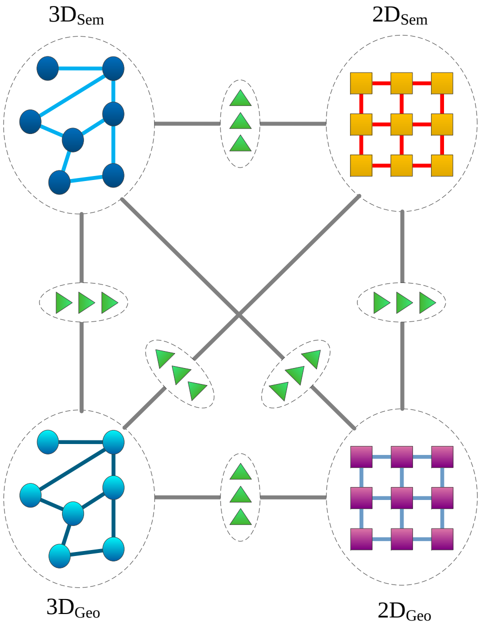


Figure 6.3: **Top:** Our model which considers 2D semantic, 3D semantic, 2D geometric and 3D geometric nodes that are connected to each other via latent nodes. This model enables us to do inference on all the nodes using the semantic and geometric information simultaneously. Different colors represent different modalities. The latent nodes are represented by triangles.

evidence from each domain, intra-domain relationships and inter-domain compatibility via our latent nodes.

Gould et al [36] integrated the semantic and geometric clues into their 2D scene understanding system and decomposed the scene into semantically and geometrically meaningful regions. Following [36], Tighe and Lazebnik [99] incorporated the geometric information into their region-wise scene parsing system (*Superparsing*) where they enforced coherence between the semantic labels (*building, car, person, etc.*) and geometric labels (*vertical surfaces, sky and ground*). Inspired by the above, we propose to use the semantic and geometric information of both 2D and 3D data simultaneously. To this end, we build our model upon different nodes which represent the *semantic* and *geometric* labels of each modality separately. These nodes are then linked together as seen in Figure 6.3 for a simultaneous inference procedure. Note that our method enables us to apply more modalities with their own set of categories. This in turn improves the performance of the system, with negligible impact on its run-time.

We demonstrate the effectiveness of our approach on two publicly available 2D-3D scene analysis datasets: Our NICTA/2D3D dataset and the CMU/VMR dataset [72]. Our experiments evidence the benefits of the latent nodes and augmentation of the multiple modalities with their semantic and geometric annotations. It also indicates the advantage of learning the potentials for multi-modal scene parsing. In particular, our approach outperforms the state-of-the-art on both datasets.

6.2 A General Multi-modal CRF

In this section, we present our multi-modal graphical model. Let $\mathbf{x}^{Mod_P} = \{\mathbf{x}_i^{Mod_P}\}$, $1 \leq i \leq N_m$, be the set of features extracted from the elements of the p^{th} modality and $\mathbf{y}^{Mod_P} = \{y_i^{Mod_P}\}$, $1 \leq i \leq N_m$, be the set of variables encoding the labels of the nodes in that modality, where each variable can take a label in the set $\mathcal{L} = \{1, \cdots, L\}$. Then the joint distribution of all modalities conditioned on the features can be expressed as

$$P(\mathbf{y}^{Mod_1}, \mathbf{y}^{Mod_2}, ..., \mathbf{y}^{Mod_P} | \mathbf{x}^{Mod_1}, \mathbf{x}^{Mod_2}, ..., \mathbf{x}^{Mod_P}) = \tag{6.1}$$
$$\frac{1}{Z} \cdot \exp\Big(- \sum_{m=1}^{P} \Big(\sum_{i=1}^{N_m} \Phi_i^{Mod_m} + \sum_{(i,j) \in \mathcal{E}^{Mod_m}} \Psi_{ij}^{Mod_m} + \sum_{i=1}^{m-1} \sum_{(j,t) \in \mathcal{E}^{Mod_m, Mod_i}} \Psi_{jt}^{Mod_m - Mod_i} \Big) \Big),$$

where Z is the partition function, and Φ^{Mod_P} denotes the unary potentials of modality p. Ψ^{Mod_P} and $\Psi^{Mod_m - Mod_i}$ denote pairwise potentials defined over the set of edges $\mathcal{E}^{Mod_P}$ (intra-domain) and $\mathcal{E}^{Mod_m - Mod_i}$ (inter-domain), respectively. The potential functions in Equation 6.1 are built such that they could intuitively model the correlation between the class probabilities and local information of each node, as well as the contextual relationships between the pairs of adjacent nodes in the graph (pairwise potentials).

In the previous chapter, we introduced a multi-modal graphical model relying on handcrafted potentials, where the pairwise potential function was defined in a way that penalizes dissimilar class labels for two adjacent regions if their feature vectors are very similar. The contributions of the handcrafted potentials in the inference process were determined via a set of weighting parameters. These parameters were then adjusted through a validation step, so as to produce the lowest error on the validation data.

A drawback of the handcrafted potentials that are based on a Pott's model is that they do not convey any information on the compatibility of different objects and class labels. As an example, take the scenario where a superpixel in the 2D domain is classified as *Grass* and it has connections with two different 3D segments, one labeled as a flat object, e.g., *Road* or *Grass*, and the other one predicted to be a cylindrical object such as *Powerpole*. Assigning the same weight to these pairwise links, even if they have the same amount of 2D-3D overlap, might not be the right decision because, in the former case, the predicted 2D class is compatible with the predicted class in the 3D domain. However, in the latter, the difference in shape of the predicted classes demands a more tuned and class-specific pairwise weight. This problem can be addressed by considering different weights for different class combinations of the nodes in a pairwise edge, e.g., *2D:Grass-3D:Grass*, *2D:Grass-3D:Road*, or *2D:Grass-3D:Tree Trunk*. Therefore we assign a set of label compatibility parameters for all possible class combinations and learn them from data.

Moreover, assigning a fixed set of weights to the unary potentials of different modalities overlooks the fact that some of the classes are recognized better using one data modality and some other object classes can be described more precisely using the other modality. For instance, when it is deduced from the 3D data that the object of interest has a flat shape, the labeling algorithm should trust the 3D information more to put the object in one of the flat categories. If, in this case, the 2D data describes the object as a green entity, e.g., *Grass*, *Bushes*, *Tree top*, the classifier should ideally pick *Grass* as class label.

Our goal is to construct and train our graphical model based on a set of learned potentials that describe: **I)** the reliability of the local information of each domain per class, and **II)** the cost of various intra-domain and inter-domain class neighborhoods (a.k.a, the label compatibility). To obtain a labeling, we perform inference in our CRF by making use of the truncated TRW algorithm in [27].

6.2.1 Potential Definition

The CRF formulation in Equation 6.1 includes several unary and pairwise potentials that are redefined here. The unary potential of a node is generally computed via its local information and indicates the cost of assigning a class label to the node. We define the cost of assigning label l to the corresponding variables as

$$\Phi_i^{Mod_P}(y_i^{Mod_P} = l) = \mathbf{A}_l^{Mod_P}\mathbf{x}_i^{Mod_P} \ , \tag{6.2}$$

where $\mathbf{A}^{Mod_P} \in \mathbb{R}^{L \times D_{Mod_P}}$ is the parameter matrix for the unary potential in modality p, with $\mathbf{A}_l^{Mod_P}$ the row of $\mathbf{A}^{Mod_P}$ corresponding to label l. Since they directly act on the local features $\mathbf{x}_i^{Mod_P}$, this matrix encodes how much each feature dimension should be relied on to predict a specific label. Note that D_{Mod_p} refer to the dimensions of the feature vector in modality p.

Pairwise potentials express the cost of all possible joint label assignments for two adjacent nodes in the graph. The handcrafted potentials are limited to simply encouraging the nodes to share the same labels. By contrast, here, we define general pairwise potentials that let us encode sophisticated label compatibilities. For the intra-domain edges, these potentials are defined as

$$\Psi_{jk}^{Mod_P}(y_j^{Mod_P} = l, y_k^{Mod_P} = s) = \mathbf{B}_{ls}^{Mod_P} \mathbf{v}_{jk}^{Mod_P} \,, \tag{6.3}$$

where $\mathbf{B}^{Mod_P}$ is a parameter matrix with L^2 rows representing all possible combinations of two labels, and $\mathbf{B}_{ls}^{Mod_P}$ is the row of $\mathbf{B}^{Mod_P}$ corresponding to the combination of label l with label s. In this case, we set the edge features $\mathbf{v}_{jk}^{Mod_P}$ to be the ℓ_2-norm of the difference of a subset of the original node features x_j and x_k, which will be discussed in Section 6.5.1.1.

Similarly, the inter-domain pairwise potential between modality i and modality m is defined as

$$\Psi_{jt}^{Mod_i - Mod_m}(y_j^{Mod_i} = l, y_t^{Mod_m} = s) = \mathbf{B}_{ls}^{Mod_i - Mod_m} \mathbf{v}_{jt}^{Mod_i - Mod_m} \,, \tag{6.4}$$

where $\mathbf{v}_{jt}^{Mod_i - Mod_m}$ is the concatenation of a subset of the original node features in modality i and modality m.

6.3 General Multi-modal CRF with Latent Nodes

We now address the problem of inconsistencies across the modalities by introducing latent nodes to our model. The latent nodes are placed between the pairs of corresponding nodes in two modalities. This breaks down the between-modality edges into two edges that link the node in modality i and the latent node, and also the node in modality m and the latent node. In other words no edge directly connects modality i to modality m. Our latent nodes can either take a label from the same space as the label space of the modality i or modality m nodes[1], or another label indicating that the link between the two modalities should be cut.

Formally, let $\mathbf{y}^{Mod_P} = \{y_i^{Mod_P}\}$, $1 \le i \le N_m$ be the set of variables encoding the node label in modality p. Each of these variables, can take a label in the set $\mathcal{L} = \{1, \cdots, L\}$. Furthermore, let $T_{m,i}$ be the number of pairs of corresponding nodes in modality m and modality i, found in the manner described in Section 6.5.1.1. We then denote by $\mathbf{y}^{\Delta} = \{y_t^{\Delta_{Mod_i, Mod_m}}\}$, $1 \le t \le T_{m,i}$ the latent nodes associated with these correspondences. These variables can be assigned a label from the space $\mathcal{L}' = \{0, 1, \cdots, L\}$, where label 0 represents a broken link, which means the nodes do not impact each other.

Given $\mathbf{x}^{Mod_P} = \{\mathbf{x}_i^{Mod_P}\}$ as the features extracted from the elements in modality p, the

[1] When modality i and modality m have different label spaces, the latent node can take a label from one of them.

joint probability distribution of all data nodes and latent nodes conditioned on the features can be expressed as

$$\begin{aligned} P(&\mathbf{y}^{Mod_1}, \mathbf{y}^{Mod_2}, ..., \mathbf{y}^{Mod_P}, \mathbf{y}^{\Delta_{Mod_1,Mod_2}}, \mathbf{y}^{\Delta_{Mod_1,Mod_3}}, \mathbf{y}^{\Delta_{Mod_2,Mod_3}}, ..., \\ &\mathbf{y}^{\Delta_{Mod_{P-1},Mod_P}} | \mathbf{x}^{Mod_1}, \mathbf{x}^{Mod_2}, ..., \mathbf{x}^{Mod_P}) = \frac{1}{Z} \cdot \\ &\exp\Big(- \sum_{m=1}^{P} \Big(\sum_{i=1}^{N_m} \Phi_i^{Mod_m} + \sum_{(i,j)\in\mathcal{E}^{Mod_m}} \Psi_{ij}^{Mod_m} \\ &+ \sum_{i=1}^{m-1} \Big(\sum_{t=1}^{T_{m,i}} \Phi_t^{\Delta_{Mod_m,Mod_i}} + \sum_{(j,t)\in\mathcal{E}^{Mod_m,\Delta_{Mod_m,Mod_i}}} \Psi_{jt}^{Mod_m-\Delta_{Mod_m,Mod_i}} \\ &+ \sum_{(j,t)\in\mathcal{E}^{Mod_i,\Delta_{Mod_m,Mod_i}}} \Psi_{jt}^{Mod_i-\Delta_{Mod_m,Mod_i}} \Big) \Big) \Big), \end{aligned} \tag{6.5}$$

Where $\Phi^{\Delta_{Mod_m,Mod_i}}$ denotes the unary potential of the latent nodes and $\Psi^{Mod_m-\Delta_{Mod_m,Mod_i}}$ denotes the pairwise potentials defined over the set of edges $\mathcal{E}^{Mod_m-\Delta_{Mod_m,Mod_i}}$. To obtain a labeling, as in Section 6.2 we use the TRW method to perform inference in our CRF. In the remainder of this section, the latent potentials in Equation 6.5 are described.

6.3.1 Unary Potentials of Latent Nodes

Similar to data modality nodes, the unary potential for the latent nodes is defined as

$$\Phi_t^{\Delta_{Mod_m,Mod_i}}(y_t^{\Delta_{Mod_m,Mod_i}} = l) = \mathbf{A}_l^{\Delta_{Mod_m,Mod_i}} \mathbf{x}_t^{\Delta_{Mod_m,Mod_i}} , \tag{6.6}$$

where $\mathbf{A}^{\Delta_{Mod_m,Mod_i}}$ is again a parameter matrix, which this time contains $L + 1$ rows to represent the fact that a latent node can take an additional label to cut the connection between two modalities. The feature vector of a latent node is constructed by concatenating the features of the corresponding Mod_m and Mod_i nodes, i.e., $\mathbf{x}_t^{\Delta_{Mod_m,Mod_i}} = [(\mathbf{x}_j^{Mod_m})^T, (\mathbf{x}_k^{Mod_i})^T]^T$. Having access to both Mod_m and Mod_i features allows this unary to detect mis-matches in the Mod_m and Mod_i observations, and in that event, favor cutting the corresponding edge.

6.3.2 Inter-domain Pairwise Potentials with Latent Nodes

The inter-domain pairwise potentials associated with the latent nodes that connect two modalities are defined as

$$\Psi_{jt}^{Mod_m-\Delta_{Mod_m,Mod_i}}(y_j^{Mod_m}=l, y_t^{\Delta_{Mod_m,Mod_i}}=s) = \mathbf{B}_{ls}^{Mod_m-\Delta_{Mod_m,Mod_i}} \mathbf{v}_{jt}^{Mod_m-\Delta_{Mod_m,Mod_i}}, \tag{6.7}$$

and

$$\Psi_{kt}^{Mod_i-\Delta_{Mod_m,Mod_i}}(y_k^{Mod_i}=l, y_t^{\Delta_{Mod_m,Mod_i}}=s) = \mathbf{B}_{ls}^{Mod_i-\Delta_{Mod_m,Mod_i}} \mathbf{v}_{kt}^{Mod_i-\Delta_{Mod_m,Mod_i}}, \tag{6.8}$$

where the parameter matrices now have $L \times (L+1)$ rows to account for the extra label of the latent nodes. In practice, we set $\mathbf{v}_{jt}^{Mod_m-\Delta_{Mod_m,Mod_i}}$ and $\mathbf{v}_{kt}^{Mod_i-\Delta_{Mod_m,Mod_i}}$ to 1, thus resulting in $L \times (L+1)$ parameters. Note, however, that the effective number of parameters corresponding to these potentials is much smaller. The reason is that the only cases of interest are when the latent node and the regular node take the same label, and when the latent node indicates a broken link. The cost of the other label combinations should be heavily penalized since they never occur in practice. This therefore truly results in $2L$ parameters for each of these potentials.

6.4 Training our Multi-modal Latent CRF

Our multi-modal CRF contains many parameters, which thus cannot be tuned manually. Here, we propose to learn these parameters from training data. To this end, we make use of the direct loss minimization method of [27].

More specifically, let $\{\mathbf{z}_i\}$, $1 \leq i \leq N$ be a set of N labeled training examples, such that $\mathbf{z}_i = \left(\mathbf{x}_i^{Mod_1}, ..., \mathbf{x}_i^{Mod_P}, \tilde{\mathbf{y}}_i^{Mod_1}, ..., \tilde{\mathbf{y}}_i^{Mod_P}, \tilde{\mathbf{y}}_i^{\Delta_{Mod_1-Mod_2}}, ..., \tilde{\mathbf{y}}_i^{\Delta_{Mod_{P-1}-Mod_P}}\right)$, where, with a slight abuse of notation compared to Section 6.2 and Section 6.3, $\mathbf{x}_i^{Mod_P}$, resp. $\tilde{\mathbf{y}}_i^{Mod_P}$, englobes the features, resp. ground-truth labels, of all the nodes in the i^{th} training sample for modality P, and similarly for the other terms in $\mathbf{z}_i$. In practice, to obtain the ground-truth labels of the latent nodes $\tilde{\mathbf{y}}_i^{\Delta_{Mod_i-Mod_m}}$, we simply check if the ground-truth labels of the corresponding modality i and modality m nodes agree, and set the label of the latent node to the same label if they do, and to 0 otherwise[2].

[2] Note that our nodes are latent in the sense that they do not correspond to physical entities, not in the sense that we do not have access to their ground-truth during training. They are imagined nodes in our model between each two modalities that have access to the information of both modalities.

Learning the parameters of our model is then achieved by minimizing the empirical risk

$$r(\Theta) = \sum_{i=1}^{N} l(\Theta, \mathbf{z}_i) \tag{6.9}$$

w.r.t. $\Theta = \{\mathbf{A}^{Mod_1}, ..., \mathbf{A}^{Mod_P}, \mathbf{A}^{\Delta_{Mod_1-Mod_2}}, ..., \mathbf{A}^{\Delta_{Mod_{P-1}-Mod_P}}, \mathbf{B}^{Mod_1}, ..., \mathbf{B}^{Mod_P}, \mathbf{B}^{Mod_1-\Delta_{Mod_1-Mod_2}}, ..., \mathbf{B}^{Mod_P-\Delta_{Mod_{P-1}-Mod_P}}\}$, where $l(\Theta, \mathbf{z}_i)$ is a loss function.

Here, we use a marginal-based loss function, which measures how well the marginals obtained via inference in the model match the ground-truth labels. In particular, we rely on a loss function defined on the clique marginals [106]. This can be expressed as $l(\Theta, \mathbf{z}_i) = -\sum_c \log \mu(\mathbf{z}_{i,c}; \Theta)$ where c sums over all the cliques in the CRF, i.e., all the inter-domain and intra-domain pairwise cliques in our case, $\mathbf{z}_{i,c}$ denotes the variables of $\mathbf{z}_i$ involved in a particular clique c, and $\mu(\mathbf{z}_{i,c}; \Theta)$ indicates the marginals of clique c obtained by performing inference with parameters Θ.

We use the publicly available implementation of [27] with truncated TRW as inference method. This method was shown to converge to stable parameters in only a few iterations. In practice, we run a maximum of 5 iterations of this algorithm.

6.5 Especial Cases

In this section, we demonstrate how our general multi-modal model can be used for modeling two especial cases of **I)** 2D-3D multi-modal data, and **II)** 2D-3D semantic and geometric multi-modal data, both accompanied with latent nodes.

6.5.1 2D-3D CRF with Latent Nodes

Here, we focus the discussion on two modalities, 2D imagery and 3D point clouds, which are typically the most common ones for scene parsing. Note, however, that our approach generalizes to other modalities, such as multi-spectral or infrared data.

Our model contains separate nodes for 2D regions (i.e., superpixels) and 3D regions (i.e., 3D segments). More details about these regions are provided in Section 6.5.1.1. We also consider latent nodes that allow us to account for inconsistencies between the different domains. To this end, and as illustrated in Figure 6.2, we incorporate one such latent node between each pair of corresponding 2D and 3D nodes. This results in edges between either a 2D node and a latent node, or a 3D node and a latent node, but no edges directly connecting a 2D node to a 3D node. Our latent nodes can then either take a label from the same space as the 2D and 3D nodes, or take another label indicating that the link between the two modalities should be cut (label 0).

Figure 6.4 illustrates through an example how latent nodes operate in case of a misalignment between 2D and 3D data for narrow objects. In Figure 6.5, we show that multi-modal data is prone to errors due to moving objects like a vehicle. In each case, latent nodes utilize the 2D and 3D information and either assist the linked 2D-3D regions to find their class label or cut off the link between them.

Formally, let $\mathbf{y}^{2D} = \{y^{2D}_{ij}\}\ ,\ 1 \leq i \leq F\ ,\ 1 \leq j \leq N_i$, be the set of variables encoding the labels of the 2D nodes in F frames, with frame i containing N_i 2D regions. Similarly, let $\mathbf{y}^{3D} = \{y^{3D}_i\}\ ,\ 1 \leq i \leq M$ be the set of variables encoding the label of M 3D nodes. Each of these variables, either 2D or 3D, can take a label in the set $\mathcal{L} = \{1, \cdots, L\}$. Furthermore, let T be the number of pairs of corresponding 2D and 3D nodes, found in the manner described in Section 6.5.1.1. We then denote by $\mathbf{y}^{\Delta} = \{y^{\Delta}_t\}\ ,\ 1 \leq t \leq T$ the latent nodes associated with these correspondences. These variables can be assigned a label from the space $\mathcal{L}' = \{0, 1, \cdots, L\}$.

Given features extracted from the 2D and 3D regions, $\mathbf{x}^{2D} = \{\mathbf{x}^{2D}_{ij}\}$ and $\mathbf{x}^{3D} = \{\mathbf{x}^{3D}_i\}$, respectively, the joint distribution of the 2D, 3D and latent nodes conditioned on the features can be expressed as

$$P(\mathbf{y}^{2D}, \mathbf{y}^{3D}, \mathbf{y}^{\Delta} | \mathbf{x}^{2D}, \mathbf{x}^{3D}) = \frac{1}{Z} \cdot \tag{6.10}$$
$$\exp\Big(- \sum_{i=1}^{F} \sum_{j=1}^{N_i} \Phi^{2D}_{ij} - \sum_{i=1}^{M} \Phi^{3D}_i - \sum_{t=1}^{T} \Phi^{\Delta}_t - \sum_{i=1}^{F} \sum_{(j,k) \in \mathcal{E}^{2D}_i} \Psi^{2D}_{ijk}$$
$$- \sum_{(i,j) \in \mathcal{E}^{3D}} \Psi^{3D}_{ij} - \sum_{i=1}^{F} \sum_{(j,t) \in \mathcal{E}^{2D-\Delta}} \Psi^{2D-\Delta}_{ijt} - \sum_{(i,t) \in \mathcal{E}^{3D-\Delta}} \Psi^{3D-\Delta}_{it} \Big),$$

where Φ^{2D}, Φ^{3D}, and Φ^{Δ} denote the unary potentials of the 2D, 3D and latent nodes, respectively. Ψ^{2D}, Ψ^{3D}, $\Psi^{2D-\Delta}$ and $\Psi^{3D-\Delta}$ denote pairwise potentials defined over the set of edges $\mathcal{E}^{2D}$, $\mathcal{E}^{3D}$, $\mathcal{E}^{2D-\Delta}$ and $\mathcal{E}^{3D-\Delta}$, respectively. All the unary and pairwise potentials are calculated based on the formulations in Section 6.2 and Section 6.3. Below, we provide some details regarding our features and potentials.

6.5.1.1 Features and Potentials

3D Nodes and 2D Nodes These features are the same as the ones used in the previous chapter.

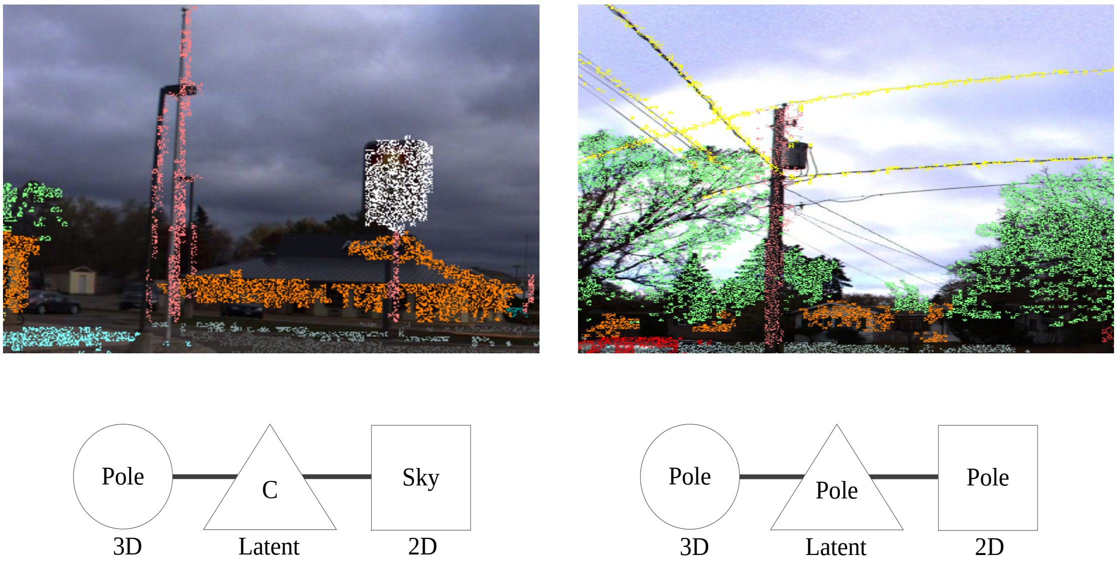


Figure 6.4: **Latent nodes for data misalignment. Left:** The projection of *pole* from 3D to 2D covers some regions of *sky*, which creates a connection between the corresponding 3D and 2D nodes. Having access to both 3D and 2D features, the latent node should detect the mis-match and cut this connection thus allowing the nodes to take different labels. **Right:** In this case, the projection is accurate. Therefore, the 2D and 3D features are both coherent with the class label *pole*, and thus the latent node should keep the edge active and predict the same class.

Latent Nodes The features of the latent nodes were obtained by concatenating the features of their respective 2D and 3D nodes, described above. Furthermore, we augmented these features with the normalized overlap area of the projection of the 3D segment onto the 2D superpixel.

Edges For the intra-domain potentials, we employed the ℓ_2-norm of the difference of a subset of the local feature vectors (RGB for 2D-2D edges and vertical-axis deviation for 3D-3D edges) as pairwise features. The feature vectors of the 2D-Δ and 3D-Δ edges were set to a single value of 1. In the case of the 2D-3D CRF with no latent nodes, however, the feature vector of the 2D-3D edges was constructed by concatenating the RGB values of the 2D node with the eigenvalue features and the vertical-axis deviation of the 3D node, as well as with the same normalized overlap area used for the unary of the latent nodes. These features were selected via an ablation study on a validation set. As evidenced by our results, they yield better accuracies than employing all of them, which causes overfitting. Note that the 2D-3D edges were obtained by projecting the 3D segments onto the 2D superpixels and connecting the pairs of nodes that have a significant projection overlap, i.e., intersection over union more than 0.2.

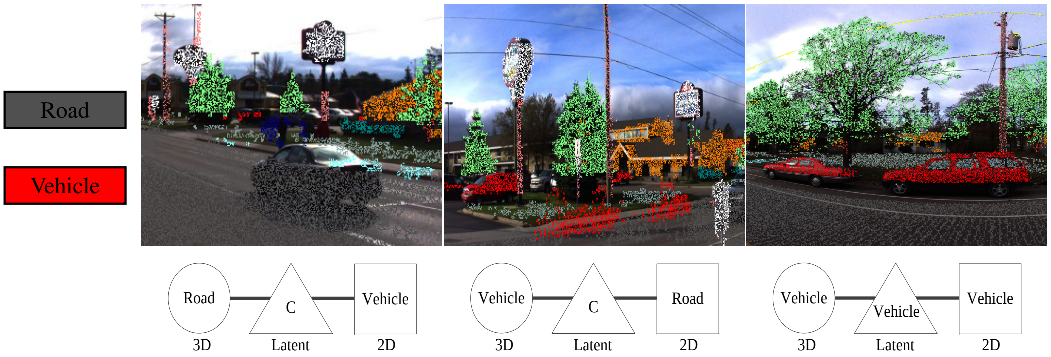


Figure 6.5: **Latent nodes for moving objects. Left:** A *vehicle* can be observed in 2D, but was not present when the 3D laser sensor covered this area. Therefore, the label of the 3D point is *road* instead of *vehicle* for 2D. By relying on both 2D and 3D features, the latent node should predict that this connection must be cut. **Middle:** This represents the opposite scenario where the image depicts an empty *road*, while the 3D points were acquired when a *vehicle* was passing. Here again, the latent node should cut the edge, thus allowing the nodes to take different labels. **Right:** In contrast, here, the 2D and 3D regions belong to the same class and thus have coherent features. The latent node should therefore leverage this information to help predicting the correct class *vehicle*.

6.5.2 Simultaneous Inference of Semantic and Geometric Classes in 2D and 3D

Fusing geometric and semantic cues has shown some potential in enhancing scene parsing results [36], [99]. This procedure can be further improved by using 3D data geometric labeling, counter to relying on 2D data for computing geometric labels [36], [99]. In Figure 6.6 the results of semantic and geometric labeling of wire are shown. In semantic labeling they were wrongly labeled as tree leaves, but in geometric labeling, they were distinct from tree leaves and correctly labeled as wire and scattered categories. This preference can help us improve the semantic labeling. In this work, we use the 2D and 3D semantic labelings as well as the 2D and 3D geometric labelings collaboratively and leverage their information through a concurrent inference process to improve the labeling results in each one of them. [36], [99] picked three categories, *horizontal*, *vertical* and *sky*, as geometric classes in their methods. Having access to 3D point cloud data enabled us to expand this list by taking into account the *cylindrical* and *scattered* categories in both 2D and 3D data, which is explained in more detail in Section 6.5.2.1. In our semantic-geometric mapping, each semantic class belongs only to one of the geometric classes, e.g., all the roads are assigned a horizontal label and all the vehicles are given a vertical label.

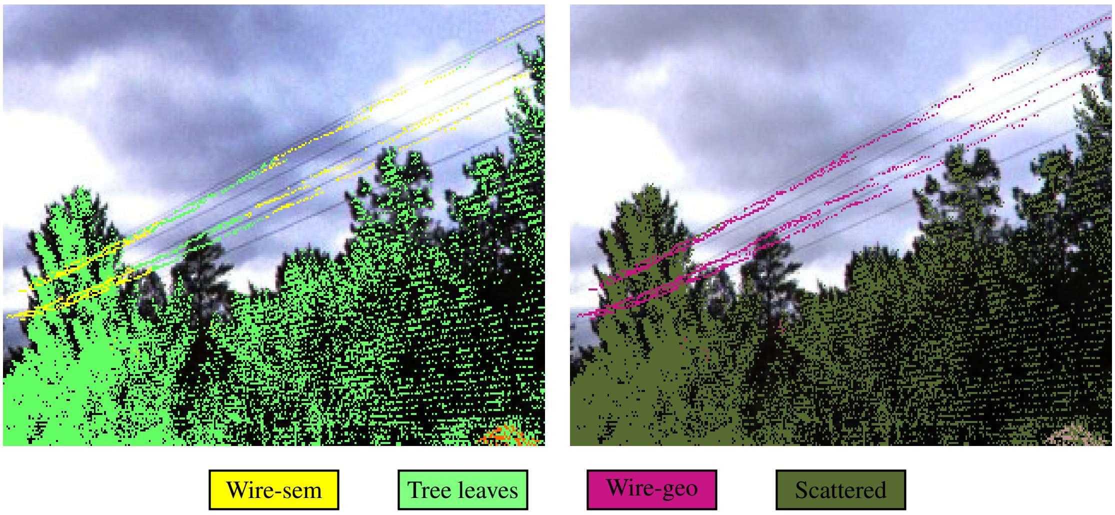


Figure 6.6: **Semantic labeling vs. geometric labeling. Left:** Semantic labeling **Right:** Geometric labeling. This sample image shows geometric labeling in compare with semantic labeling could distinct between wire and tree leaves.

Let $y^{2D_{Sem}}$, $y^{3D_{Sem}}$, $y^{2D_{Geo}}$ and $y^{3D_{Geo}}$ be the variables encoding the 2D semantic, 3D semantic, 2D geometric and 3D geometric class labels, respectively. We can then define the joint distribution of the 2D semantic, 2D geometric, 3D semantic, 3D geometric and the latent nodes (that rule over the connecting edges between these nodes), conditioned on the node features $P(\mathbf{y}^{2D_{Sem}},\mathbf{y}^{3D_{Sem}},\mathbf{y}^{2D_{Geo}},\mathbf{y}^{3D_{Geo}},\mathbf{y}^{\Delta_{2D_{Sem},2D_{Geo}}},\mathbf{y}^{\Delta_{3D_{Sem},3D_{Geo}}},\mathbf{y}^{\Delta_{2D_{Sem},3D_{Sem}}},\mathbf{y}^{\Delta_{2D_{Geo},3D_{Geo}}},$ $\mathbf{y}^{\Delta_{2D_{Sem},3D_{Geo}}},\mathbf{y}^{\Delta_{3D_{Sem},2D_{Geo}}}|\mathbf{x}^{2D_{Sem}},\mathbf{x}^{3D_{Sem}},\mathbf{x}^{2D_{Geo}},\mathbf{x}^{3D_{Geo}})$, similarly to the definition in Equation 6.5. Note that the label set in geometric nodes and semantic nodes are different.

Given that the geometric nodes represent the same set of 2D and 3D regions that were previously produced for semantic labeling, the 2D-3D geometric edges are similar to the 2D-3D semantic edges. Furthermore, note that the latent nodes which link the semantic and geometric nodes both representing one 2D region (or 3D segment), cannot cut their corresponding edges although their class labels are different. The reason behind this is that they connect two visually identical regions (segments). Instead, they try to find a coherent pair of semantic and geometric class labels that sufficiently fit the 2D and 3D features of the region (segment). The truncated TRW inference method is used for this purpose, similar to what is described in Section 6.4. The inference time however does not change significantly, despite the considerable increase in the size of the graph (number of nodes and edges). Table 6.1 presents the training and inference time for the NICTA/2D3D and CMU/VMR datasets. Our method trains all the compatibility parameters between the semantic and geometric class labels, which contrasts with the Superparsing method [99], where only one parameter is embedded in the cost function to enforce consistency between these two groups of classes. Note that we used the same features and re-trained classifier for the new classes as in Section 6.5.1.1 for the geometric nodes.

6.5.2.1 Semantic and Geometric Classes

In order to best exploit the geometric cues, particularly given the 3D point cloud data, the data is clustered into different structural classes including *horizontal plane*, *vertical plane*, *scattered* and *cylindrical* (in addition to three other groups for specifically representing *sky*, *person* and *wire*). Table 6.2 provides the mapping between the geometric and semantic classes.

6.6 Experiments

We evaluate our method on two publicly available 2D-3D multi-modal datasets (NICTA/2D3D and CMU/VMR [72]). We provide the results of 2D-3D CRF with and without latent nodes and also simultaneous inference of semantic and geometric classes both in 2D and 3D. We also compare the results to the state-of-the-art algorithms of [72]. The experiment on 2D-3D CRF without latent nodes is an especial case study of the general multi-modal CRF (Section 6.2). In addition, we provide the results of the pairwise models with learned potentials acting on a single domain, either 2D or 3D. We will refer to these models as *Pairwise 2D (learned)* and *Pairwise 3D (learned)*. We followed the evaluation protocol of our previous chapter and partitioned the data into 4 non-overlapping folds. We then used three of the folds for training and the remaining fold as test set.

6.6.1 Results on NICTA/2D3D

The NICTA/2D3D dataset is comprised of 14 classes (13 for 3D where *sky* was removed), which yields the following sizes for the parameter matrices for 2D-3D CRF with latent nodes: $\mathbf{A}^{2D}_{[14\times23]}$, $\mathbf{A}^{3D}_{[13\times17]}$, $\mathbf{A}^{\Delta}_{[15\times41]}$, $\mathbf{B}^{2D}_{[196\times1]}$, $\mathbf{B}^{3D}_{[169\times1]}$, $\mathbf{B}^{2D-\Delta}_{[210\times1]}$ and $\mathbf{B}^{3D-\Delta}_{[195\times1]}$. The 2D-3D CRF with no latent nodes involves a different parameter matrix of the form $\mathbf{B}^{2D-3D}_{[182\times1]}$, $\mathbf{B}^{2D-3D}_{[182\times8]}$ and $\mathbf{B}^{2D-3D}_{[182\times41]}$.

Table 6.3 and Table 6.4 compare the results, as F1-scores, of the 2D-3D CRF model with handcrafted and learned potentials, and also with latent nodes and no latent nodes. Note that no results for [72] are available on this dataset. The results in these tables evidence the benefit of using latent nodes, especially on the narrow classes that suffer more from misalignment. On average, our approach with latent nodes clearly outperforms the model with no latent nodes,

Table 6.1: Training and inference time for NICTA/2D3D and CMU/VMR datasets.

	Training time (NICTA/2D3D)	**Inference time (NICTA/2D3D)**	**Training time (CMU/VMR)**	**Inference time (CMU/VMR)**
2D-3D CRF with latent nodes	6hr45min	0.85s	4hr40min	0.47s
Simultaneous Inference of Semantic and Geometric Classes both in 2D and 3D	19hr20min	2.3s	24hr15min	1.2s

Table 6.2: Mapping table between the geometric and semantic classes for NICTA/2D3D dataset and CMU/VMR dataset.

Geometric Classes	**Semantic Classes (NICTA/2D3D dataset)**	**Semantic Classes (CMU/VMR dataset)**
Horizontal Plane	Grass - Road - Sidewalk	Road - Sidewalk - Ground - Stairs
Vertical Plane	Building - Vehicle	Building - Small Vehicle - Big Vehicle
Cylindrical	Tree Trunk - Pole- Sign - Post-Barrier	Barrier - Bus Stop - Tree Trunk- Tall Light Post - Sign - Utility Pole- Traffic Signal
Scattered	Tree Leaves - Bush	Shrub - Tree Top
Sky	Sky	—
Person	—	Person
Wire	Wire	Wire

and thus achieves state-of-the-art results on this dataset. Furthermore, note that the 2D-3D CRF with no latent nodes that utilizes fewer features (selected features) for the 2D-3D edges is less likely to face overfitting and yields better results. Also the results of the 2D-3D CRF with no latent nodes are presented in Table 6.3 and Table 6.4 for comparison. For this experiment the feature vector of the 2D-3D edges was set to a single value of 1. In Figure 6.7, we illustrate the influence of our latent nodes by two examples. As shown in the figure, cutting the edge between the non-matching 2D and 3D nodes (which have been connected because of misalignment) helps predicting the correct class labels. Figure 6.8 shows the results of our approach in one of the scenes in this dataset, compared to the results in the previous chapter.

Table 6.3: Per class F1-scores for the 2D domain in the NICTA/2D3D dataset. We present the results for unary, pairwise model learned on the 2D domain only, with handcrafted potentials, the 2D-3D learned potentials, the 2D-3D learned potentials with latent nodes, semantic results with semantic - geometric model with and without latent nodes.

	Grass	**Building**	**Tree trunk**	**Tree leaves**	**Vehicle**	**Road**	**Bush**	**Pole**	**Sign**	**Post**	**Barrier**	**Wire**	**Sidewalk**	**Sky**	**avg**
Unary	80	33	14	80	49	95	16	28	3	0	0	29	15	98	38
Pairwise 2D (learned)	85	57	17	85	55	95	18	30	0	0	3	34	20	**99**	43
2D-3D handcrafted potentials	74	56	21	82	58	92	23	33	19	8	5	32	29	97	45
2D-3D learned potentials (no feature)	94	58	12	83	72	64	31	34	6	0	13	37	48	97	46
2D-3D learned potentials (full features)	90	63	10	91	68	96	31	43	1	0	0	44	53	**99**	49
2D-3D learned potentials (selected features)	92	64	18	92	69	98	36	34	3	0	**28**	40	60	**99**	52
2D-3D learned potentials with latent nodes	**95**	71	28	93	76	97	44	44	10	5	21	38	68	**99**	56
Semantic results with semantic - geometric model (selected features)	92	70	26	93	72	97	32	49	17	0	0	**63**	65	**99**	55
Semantic results with semantic - geometric model and latent nodes	93	79	45	**95**	77	98	34	55	22	0	0	**63**	83	**99**	60
Semantic results with semantic - geometric model (Connected 2D frames)	**95**	**82**	**52**	90	**78**	99	**78**	**99**	**33**	**60**	20	61	**92**	**99**	**62**

Our results on NICTA/2D3D indicate that, while our latent nodes are in general beneficial, thanks to their ability to cut incorrect connections, they still occasionally yield lower performance than a model without such nodes. We observed that this is mainly due to the inaccurate ground-truth (which is inevitable because of the imperfect 3D-2D projection of the ground-truth labels particularly at the boundaries of the narrow objects), or to the fact that, sometimes, while the 2D and 3D features appear to be incompatible, e.g., due to challenging viewing conditions, they still belong to the same class. The stronger smoothness imposed by the model without latent nodes is then able to address this issue.

2D-3D multi-modal scene parsing on semantic and geometric classes can be seen as an

Table 6.4: Per class F1-scores for the 3D domain in the NICTA/2D3D dataset. We present the results for unary, pairwise model learned on the 2D domain only, with handcrafted potentials, the 2D-3D learned potentials, the 2D-3D learned potentials with latent nodes, semantic results with semantic - geometric model with and without latent nodes.

	Grass	**Building**	**Tree trunk**	**Tree leaves**	**Vehicle**	**Road**	**Bush**	**Pole**	**Sign**	**Post**	**Barrier**	**Wire**	**Sidewalk**	**Sky**	**avg**
Unary	52	61	27	87	58	82	10	24	19	43	19	74	0	#	43
Pairwise 3D (learned)	58	80	50	97	56	76	16	**62**	32	40	0	89	0	#	50
2D-3D handcrafted potentials	63	81	41	96	70	76	21	38	28	47	23	87	0	#	52
2D-3D learned potentials (no feature)	68	81	31	92	67	83	**69**	43	37	25	16	75	10	#	54
2D-3D learned potentials (full features)	72	75	27	95	77	90	42	**62**	31	9	0	89	0	#	52
2D-3D learned potentials (selected features)	60	92	45	97	75	79	61	58	49	29	**27**	82	0	#	58
2D-3D learned potentials with latent nodes	66	**94**	49	95	**79**	83	51	**62**	**54**	43	25	89	8	#	61
Semantic results with semantic - geometric model (selected features)	71	88	51	97	76	84	56	60	51	49	6	92	21	#	62
Semantic results with semantic - geometric model and latent nodes	79	91	64	**99**	77	**93**	60	61	50	58	0	**96**	**34**	#	**66**
Semantic results with semantic - geometric model (Connected 2D frames)	**80**	92	**65**	98	75	**93**	65	59	49	**62**	0	93	32	#	**66**

especial case of our multi-modal model with four modalities. We considered six geometric classes in NICTA/2D3D dataset (Table 6.2) and conducted similar procedures as in the semantic labeling for finding their regions and node features. The 2D and 3D geometric data are augmented to the semantic model as two separate data modalities and their simultaneous inference is carried out given the semantic and geometric cues of the 2D and 3D data. Tables 6.3 and 6.4 demonstrate the results of the 2D and 3D semantic scene parsing using the proposed semantic and geometric 2D/3D multi-modal model. As reported in this table, leveraging the geometric cues has led to 4% and 5% improvement in F1-scores of the 2D and 3D data, respectively. The results of the geometric labeling of the 2D and 3D data are shown in Table 6.5. Furthermore, the panoramic images in NICTA 2D/3D provide the chance of observing an object in successive image frames and as a result, multiple 2D features for each object can be recorded. We linked these corresponding 2D nodes together with considering latent nodes in each connection to enhance their labeling and gained a 2% improvement on the 2D performance, as shown in Table 6.3. Figure 6.9 shows some sample results of our semantic and geometric labeling on the NICTA/2D3D dataset.

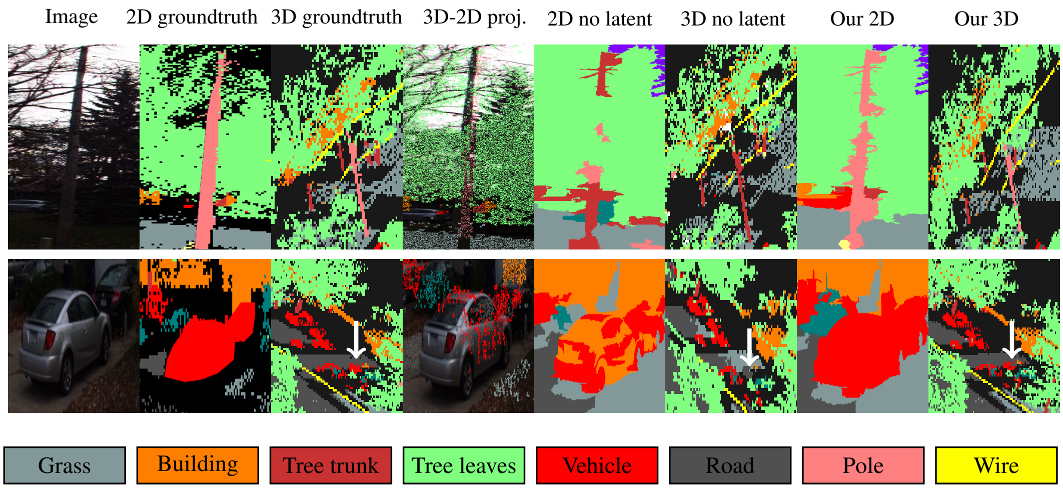


Figure 6.7: **Examples of how our latent nodes improve the labeling in practice.** As shown in the 3D-2D projection, the data misalignment and object motions have caused 3D points labeled as *leaves* to cover the *pole* (top) and 3D points labeled as *road* to project onto the *vehicles* (bottom). As a consequence, with the method in our previous chapter which encourages the modalities to have the same label, the pole was labeled as *leaves* in 2D and the vehicle as *road* in 3D (indicated by a white arrow). By contrast, thanks to our latent nodes that can cut inconsistent edges, our method produces the correct labels.

Table 6.5: Per class F1-scores for geometric results with semantic - geometric model and latent nodes in the NICTA/2D3D dataset.

	Horizontal plane	**Vertical plane**	**Cylindrical**	**Scattered**	**Wire**	**Sky**	**avg**
2D geometric results with semantic - geometric model	98	76	25	94	43	99	72
3D geometric results with semantic - geometric model	99	91	62	99	95	#	89

6.6.2 Results on CMU/VMR

The ground-truth of the CMU/VMR dataset data is such that the labels of corresponding 2D and 3D nodes are always the same[3]. In other words, this dataset is not particularly well-suited to our approach. However, it remains a standard benchmark, and no other dataset explicitly evidencing the misalignment problem is available. The CMU/VMR dataset contains 19 classes, which yields the following sizes for the parameter matrices for 2D-3D CRF with latent nodes: $\mathbf{A}^{2D}_{[19\times28]}$, $\mathbf{A}^{3D}_{[19\times23]}$, $\mathbf{A}^{\Delta}_{[20\times52]}$, $\mathbf{B}^{2D}_{[361\times1]}$, $\mathbf{B}^{3D}_{[361\times1]}$, $\mathbf{B}^{2D-\Delta}_{[380\times1]}$ and $\mathbf{B}^{3D-\Delta}_{[380\times1]}$, with alternative matrices for the 2D-3D CRF with no latent nodes of the form $\mathbf{B}^{2D-3D}_{[361\times1]}$, $\mathbf{B}^{2D-3D}_{[361\times8]}$ and $\mathbf{B}^{2D-3D}_{[361\times52]}$.

[3] Note that by looking at the data, one can observe that this ground-truth is often wrong, because of the misalignment problem.

Table 6.6: Per class F1-scores for the 2D domain in the CMU/VMR dataset. We present the results for unary and pairwise models learned on the 2D domain only, the method of [72], with handcrafted potentials, the 2D-3D learned potentials, the 2D-3D learned potentials with latent nodes, semantic results with a semantic - geometric model with and without latent nodes.

	Road	Sidewalk	Ground	Building	Barrier	Bus stop	Stairs	Shrub	Tree trunk	Tree top	Small Vehicle	Big vehicle	Person	Tall light	Post	Sign	Utility pole	Wire	Traffic Signal	avg
Unary	95	81	75	56	29	17	32	50	31	53	32	49	29	**16**	**15**	**16**	33	**41**	29	41
Pairwise 2D (learned)	89	77	74	84	25	17	40	62	37	89	78	57	38	1	5	3	16	12	9	43
Munoz [72]	**96**	**90**	70	83	50	16	33	62	30	86	**84**	50	47	2	9	**16**	14	2	17	45
2D-3D handcrafted potentials	94	87	79	74	45	22	40	54	27	84	67	24	38	13	2	10	37	35	**40**	46
2D-3D learned potentials (no feature)	95	84	78	70	58	18	57	68	43	84	81	52	55	9	3	2	15	5	8	47
2D-3D learned potentials (full features)	93	85	83	**88**	60	4	61	**67**	41	87	79	61	45	0	3	2	12	9	2	46
2D-3D learned potentials (selected features)	93	80	80	87	60	1	70	**67**	37	**90**	**84**	**67**	54	7	4	4	21	15	3	49
2D-3D learned potentials with latent nodes	94	84	**84**	84	**65**	4	**75**	64	43	89	**84**	58	52	11	6	2	25	18	3	**50**
Semantic results with semantic - geometric model (selected features)	94	87	82	82	61	26	59	68	43	89	74	60	55	0	4	4	27	15	8	49
Semantic results with semantic - geometric model and latent nodes	94	87	**84**	81	58	**28**	63	66	**47**	87	78	64	**56**	0	6	5	**38**	17	10	**51**

We compare the results of the 2D-3D CRF model with handcrafted and learned potentials and also with latent nodes and no latent nodes in Table 6.6 and Table 6.7 for the 2D and 3D domains, respectively. In this case, while our approach still yields the best F1-scores on average, there is less difference between our results with latent nodes and the no latent method. This can easily be explained by the fact that, as mentioned above, the ground-truth labels of corresponding nodes in 2D and 3D are always the same. Furthermore, we can also see that our approach yields low accuracy on classes where few training samples were available, such as the last 5 categories in the tables. This should come at no surprise, since our learning strategy strongly relies on training data. A qualitative comparison is provided in Figure 6.10.

Six geometric classes are considered in CMU/VMR dataset (Table 6.2). Similarly to the NICTA/2D3D dataset, the 2D and 3D geometric data are augmented to the semantic model as two separate data modalities and their simultaneous inference is carried out given the semantic and geometric cues of the 2D and 3D data. Tables 6.6 and 6.7 demonstrate the results of the 2D and 3D semantic scene parsing using the proposed semantic and geometric 2D/3D multi-modal model. It improves the F1-scores of the 2D and 3D data. The results of the geometric labeling of the 2D and 3D data are shown in Table 6.8. Figure 6.11 shows some sample results of our semantic and geometric labeling on the CMU/VMR dataset.

Table 6.7: Per class F1-scores for the 3D domain in the CMU/VMR dataset. We present the results for unary and pairwise models learned on the 2D domain only, the method of [72], with handcrafted potentials, the 2D-3D learned potentials, the 2D-3D learned potentials with latent nodes, semantic results with a semantic - geometric model with and without latent nodes.

	Road	**Sidewalk**	**Ground**	**Building**	**Barrier**	**Bus stop**	**Stairs**	**Shrub**	**Tree trunk**	**Tree top**	**Small Vehicle**	**Big vehicle**	**Person**	**Tall light**	**Post**	**Sign**	**Utility pole**	**Wire**	**Traffic Signal**	**avg**
Unary	70	49	62	67	34	2	19	26	11	67	34	4	13	2	0	1	2	0	0	24
Pairwise 3D (learned)	78	52	67	78	15	1	32	31	1	73	44	14	9	1	0	0	0	0	0	26
Munoz [72]	82	73	68	87	46	11	38	63	28	**88**	73	**56**	26	**10**	0	0	0	0	0	39
2D-3D handcrafted potentials	92	85	81	85	50	16	42	55	29	82	70	16	43	6	2	**7**	**29**	9	**23**	43
2D-3D learned potentials (no feature)	92	84	85	87	64	3	59	64	32	77	70	19	42	5	2	3	7	3	9	42
2D-3D learned potentials (full features)	90	86	**87**	90	59	2	64	69	31	79	70	29	47	1	1	0	5	0	0	43
2D-3D learned potentials (selected features)	90	85	85	89	62	2	63	68	29	86	78	46	53	3	1	0	15	0	0	45
2D-3D learned potentials with latent nodes	92	**88**	84	88	64	7	**66**	66	31	86	75	42	53	8	**7**	0	17	10	0	47
Semantic results with semantic - geometric model (selected features)	93	86	85	**92**	66	12	62	68	39	86	**80**	47	56	0	2	2	21	10	0	**48**
Semantic results with semantic - geometric model and latent nodes	**94**	86	**87**	90	**71**	**18**	60	**70**	**44**	87	78	43	**58**	0	2	2	28	**13**	0	**50**

Table 6.8: Per class F1-scores for geometric results with semantic-geometric model and latent nodes in the CMU/VMR dataset.

	Horizontal plane	**Vertical plane**	**Cylindrical**	**Scattered**	**Person**	**Wire**	**avg**
2D geometric results with semantic - geometric model	97	85	44	88	56	52	70
3D geometric results with semantic - geometric model	96	91	60	87	56	19	68

6.7 Summary

In this chapter, we have presented a general multi-modal model that could simultaneously accommodate multiple modalities. We have also addressed the problem of domain inconsistencies in multi-modal semantic labeling, which is an important issue when multi-modal data is concerned. Such inconsistencies typically cause undesirable connections between regions in different modalities, which in turn lead to poor labeling performance. We have therefore proposed a latent CRF model, in which latent nodes supervise the pairwise edges between domains. Having access to the information of both modalities, these nodes can either improve the labeling in both domains or cut the links between inconsistent regions. Furthermore, we presented a new set of data-driven learned potentials, which can model complex relationships between the latent nodes and the modalities. In addition, our general model enables us to consider the geometric classes together with the semantic categories for both 2D and 3D data and perform a concurrent inference on them to enhance the 2D and 3D semantic labeling results even further. Thanks to our general model, latent nodes and our learned potentials, our model achieved state-of-the-art results on two publicly available datasets.

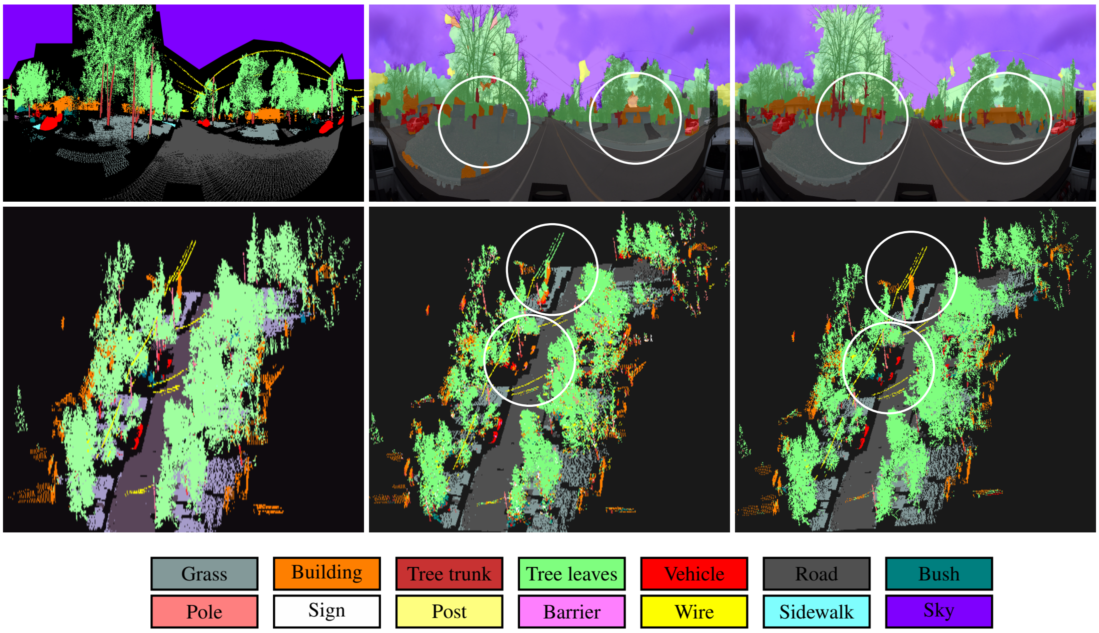


Figure 6.8: Sample results on the NICTA/2D3D dataset. **1st row: Left:** 2D ground-truth; **Middle:** 2D results with handcrafted potentials; **Right:** 2D results with learned potentials. **2nd row: Left:** 3D ground-truth; **Middle:** 3D results with handcrafted potentials; **Right:** 3D results with learned potentials. This method has been able to fix some of the mis-labelings present in our previous results with handcrafted potentials, such as the *tree trunks* and *poles* in 2D images, and *wires* and *vehicles* in 3D data. Note that these are the object classes that are most likely to be affected by misalignments.

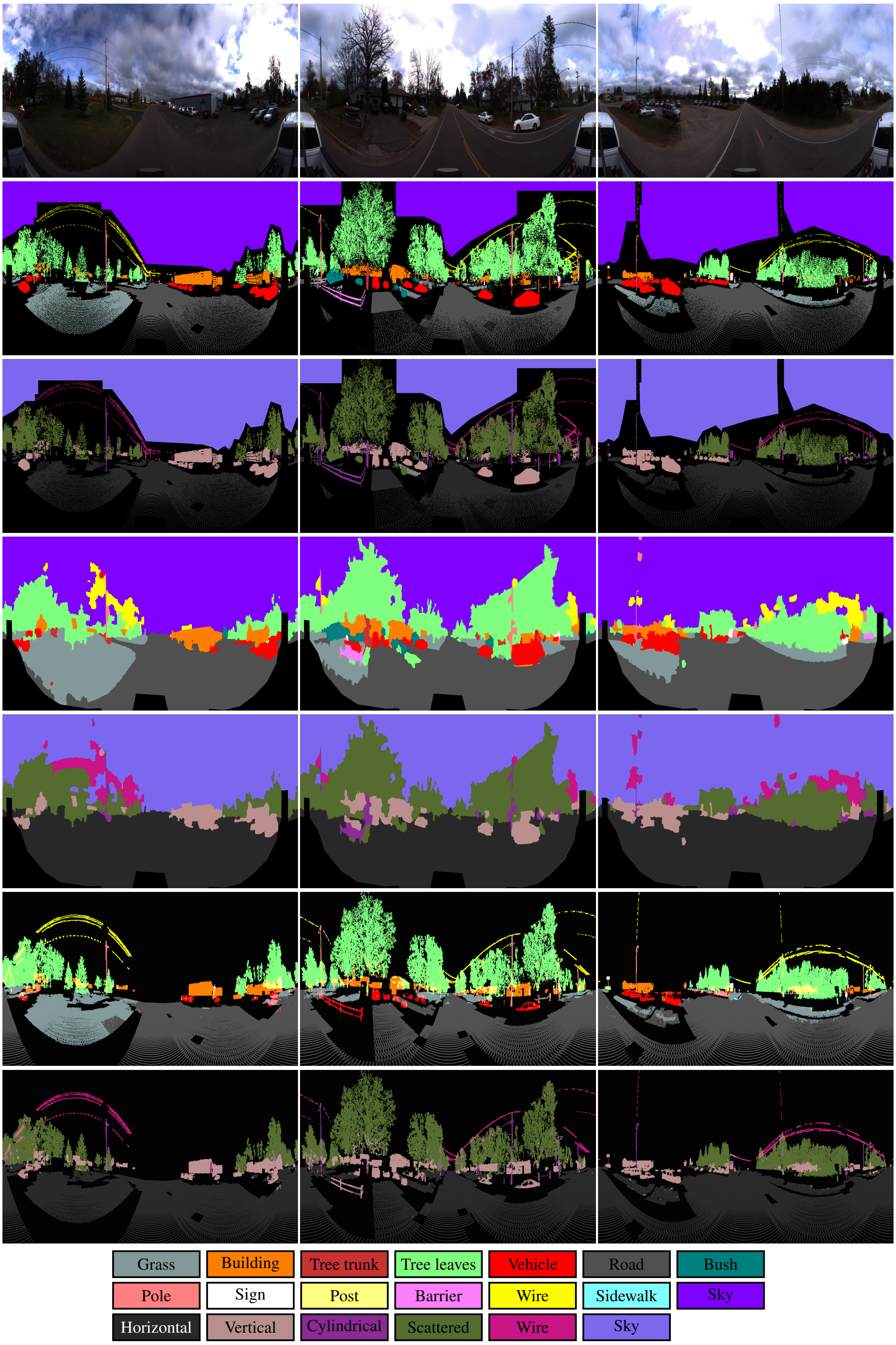


Figure 6.9: Sample results of semantic and geometric labeling in the NICTA/2D3D dataset. **1st row:** image, **2nd row:** 2D semantic ground-truth, **3rd row:** 2D geometric ground-truth, **4th row:** 2D semantic results, **5th row:** 2D geometric results, **6th row:** 3D semantic results, **7th row:** 3D geometric results.

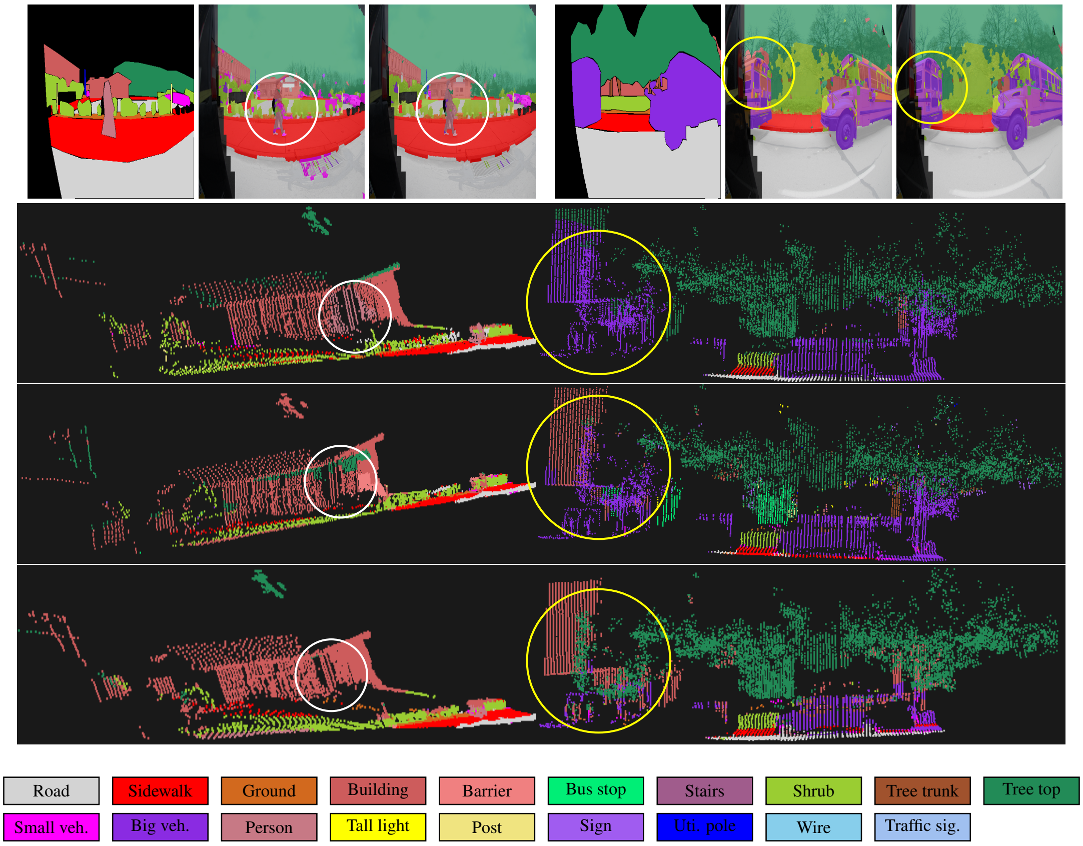


Figure 6.10: Sample results of two scenes in the CMU/VMR dataset. **1st row in each scene: Left:** 2D ground-truth; **Middle:** the results with handcrafted potentials; **Right:** 2D results with learned potentials. **2nd row:** ground-truth of the 3D data; **3rd row:** the results with handcrafted potentials; **4th row:** 3D results with learned potentials. The circles highlight mislabeling in the 3D ground-truth of this dataset, which occurred due to misalignments between 2D and 3D data, and illustrate how our method has improved the results in those regions compared to results with handcrafted potentials.

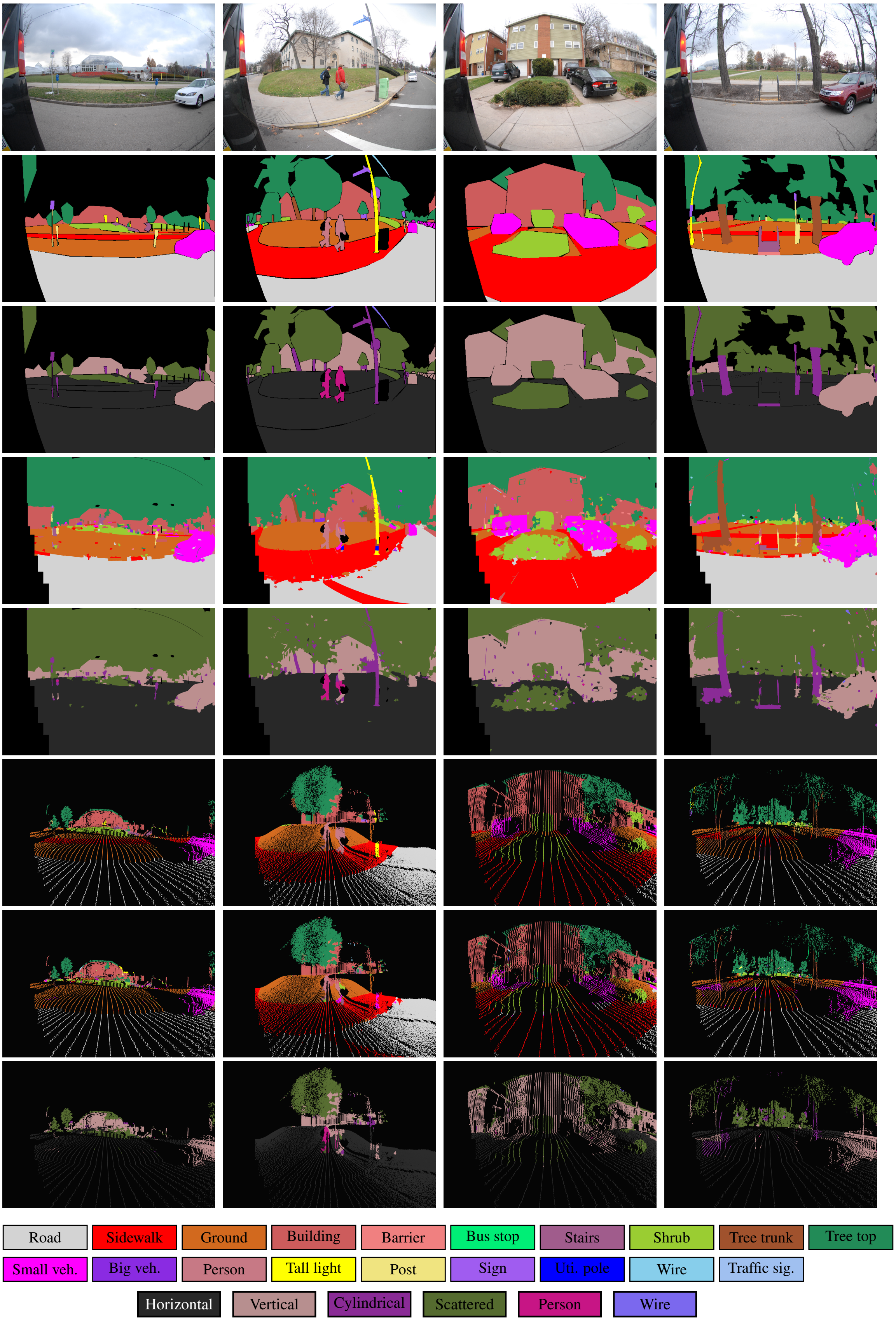


Figure 6.11: Sample results of semantic and geometric labeling in the CMU/VMR dataset. **1st row:** image, **2nd row:** 2D semantic ground-truth, **3rd row:** 2D geometric ground-truth, **4th row:** 2D semantic results, **5th row:** 2D geometric results, **6th row:** 3D semantic ground-truth, **7th row:** 3D semantic results, **8th row:** 3D geometric results.

Chapter 7

Conclusion

The overall goal of this work was to investigate methods for the purpose of using multiple modalities for scene understanding. Although various sensing modalities can be concurrently used to improve the performance of scene understanding systems, as we mentioned in Chapter 1, taking into account multiple modalities that contain different types of information and cover different sorts of object categories is a challenging task. We identified some of these challenges and addressed them in this thesis. In Section 7.1, we summarize our contributions and then we propose some possible extensions and research directions.

7.1 Summary of Contributions

In Chapter 3, we investigated the utility of multi-spectral imaging for outdoor scene understanding. We studied the benefit of the additional spectral bands in improving the system accuracy. Our approach combines the discriminating strength of the multi-spectral signature in each pixel and the corresponding nature of the surrounding texture. We exploited local features and texture features to make the system more robust to different lighting conditions. Then, classifiers built on these features were evaluated with promising results for a ten class problem. In Chapter 4, we focused on working with multiple modalities to take advantage of using 2D information for 3D point cloud labeling. We used panoramic images in conjunction with 3D Lidar data. To address issues such as occlusions, 3D-2D projection errors and misalignment between the point cloud data and 2D imagery, we proposed a consensus method that can intelligently incorporate feature responses from multiple views and reject those that are not very descriptive and select the best 2D features. These selected 2D features are used for 3D classification. The experiments are performed on a challenging dataset captured both in summer and winter. We showed that our multi-view approach improve the 3D classification in comparison using the closest 2D view only. In Chapter 5, we introduced a multi-modal graphical model that performs simultaneous inference of semantic classes both in 2D and 3D data, taking advantage of the information in both domains. We defined a graph over the entire set of data that encourages each region in a modality to leverage the information from its corresponding regions in the other modality to better estimate its class label. We evaluated our method on a publicly available dataset and beat the state-of-the-art. Additionally, to demonstrate the ability of our model to support multiple correspondences for objects in 3D and 2D, we introduced and released a new multi-modal dataset of panoramic images and 3D point cloud data captured from outdoor scenes (NICTA/2D3D Dataset). In Chapter 6, we address the problems of data misalignment and label inconsistencies in semantic labeling by introducing latent nodes to explicitly model inconsistencies between two modalities. These latent nodes allow us not only to leverage information from different modalities to improve the labeling of the modalities, but also to cut the edges between inconsistent regions. To eliminate the need for hand tuning the parameters of our model, we proposed to learn potential functions from training data. Moreover, in order to highlight the benefits of the geometric information and the potential of our method in simultaneous 2D/3D semantic and 2D/3D geometric inference, we performed simultaneous inference of semantic and geometric classes both in 2D and 3D that led to satisfactory improvements of the labeling results.

7.2 Future Work

We investigated the benefit of different modalities such as multi-spectral imaging, panoramic imaging and Lidar data in outdoor scene understanding. We proposed methods to take advantage of multiple modalities to improve the system accuracy in outdoor labeling while addressing the challenges of using multiple modalities, e.g., misalignment between modalities. However, there is still a long way to go to accomplish this goal and obtain a reliable outdoor scene understanding system. Some possible extensions and research directions are summarized below:

- Our multiple modality datasets were limited only to RGB and Lidar data for outdoor scenes. Preparing datasets that include, not only RGB images and 3D Lidar data, but also other modalities such as multi-spectral imaging and thermal imaging, that is a potential future work.

- The unsupervised segmentation process for producing 2D superpixels and 3D clusters has been done solely based on the information of their respective modality. The outcome of this step can be improved by utilizing the information of other modalities. To this end, pixel level and point level connections between modalities can be considered, which means corresponding pixels and 3D points in different modalities should be connected for the segmentation step.

- Our learning process relies heavily on training data. Therefore our approach may not work well for rare classes that do not occur very often in training data. Finding a solution to address this issue can be very helpful.
- In our model, the corresponding regions in different modalities are connected via pairwise links. Extending our model to consider higher-order potentials for corresponding regions can possibly improve the labeling results. However, defining purposeful cliques across modalities is non-trivial.

Bibliography

1. http://www.fluxdata.com/multispectral-cameras. (cited on pages xv, 9, and 27)

2. https://www.ximea.com/en/products/hyperspectral-cameras-based-on-usb3-xispec/mq022hg-im-sm4x4-vis. (cited on pages xv, 8, and 10)

3. https://www.ptgrey.com/ladybug3-360-degree-firewire-spherical-camera-systems. (cited on pages xv, 8, and 11)

4. http://www.gopano.com/. (cited on pages xv and 12)

5. . http://rgbd-dataset.cs.washington.edu/. (cited on pages xv and 13)

6. . http://www.ros.org/wiki/kinect_calibration/technical. (cited on page 12)

7. . http://velodynelidar.com. (cited on page 13)

8. https://www.google.com/selfdrivingcar/. (cited on page 13)

9. . http://velodynelidar.com/hdl-64e.html. (cited on page 13)

10. ANAND, A.; KOPPULA, H.; JOACHIMS, T.; AND SAXENA, A. Contextually guided semantic labeling and search for 3d point clouds. *IJRR*. (cited on page 59)

11. ANGUELOV, D.; TASKAR, B.; CHATALBASHEV, V.; KOLLER, D.; GUPTA, D.; HEITZ, G.; AND NG, A., 2005. Discriminative learning of markov random fields for segmentation of 3d scan data. In *CVPR*. (cited on page 22)

12. BATLER, J.; PELIZARRI, C.; AND CHEN, G., 1992. Correlation of projection radiographs in radiation therapy using open curve segments and points. *Medical Physics*, 19, 2 (1992), 329 – 334. (cited on page 15)

13. BESL, P. AND MCKAY, H., 1992. A method for registration of 3-d shapes. *PAMI*, (1992). (cited on page 22)

14. BISHOP, C. M., 2006. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York. (cited on page 21)

15. BO, L.; LAI, K.; REN, X.; AND FOX, D., 2011. Object recognition with hierarchical kernel descriptors. In *CVPR*. (cited on pages 42 and 59)

16. BRADLEY, D.; UNNIKRISHNAN, R.; AND BAGNELL, J., 2007. Vegetation detection for driving in complex environments. In *ICRA*. (cited on pages 20 and 35)

17. BREIMAN, L. Random forests. *Machine Learning*. (cited on page 16)

18. BROWN, M. AND SUSSTRUNK, S., 2011. Multi-spectral sift for scene category recognition. In *CVPR*, 177–184. (cited on pages 19, 20, and 35)

19. CADENA, C. AND KOSECK, J., 2014. Semantic Segmentation with Heterogeneous Sensor Coverages. In *ICRA*. (cited on pages 2, 24, 42, 58, 59, 60, and 65)

20. CHAN, A., 2014. An assessment of normalized difference skin index robustness in aquatic environments. In *Air Force Institute of Technology, WPAFB*. (cited on page 8)

21. CHANG, C. AND LIN, C., 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2 (2011), 27:1–27:27. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm. (cited on pages 31 and 46)

22. CHETAN, J.; KRISHNA, M.; AND JAWAHAR, C., 2010. Fast and spatially-smooth terrain classification using monocular camera. In *ICPR*. (cited on pages 16 and 27)

23. CLARK, R.; SWAYZE, G.; LIVO, K.; KOKALY, R.; SUTLEY, S.; DALTON, J.; MCDOUGAL, R.; AND GENT, C., 2002. Imaging spectroscopy: Earth and planetary remote sensing with the usgs tetracorder and expert systems. *Journal of Geophysical Research*, 108, E12 (2002). (cited on pages 8 and 29)

24. COMANICIU, D. AND MEER, P., 2002. Mean shift: A robust approach toward feature space analysis. *PAMI*, (2002). (cited on pages 16 and 67)

25. COUPRIE, C.; FARABET, C.; NAJMAN, L.; AND LECUN, Y., 2013. Indoor semantic segmentation using depth information. CoRR, abs/1301.3572. (cited on page 59)

26. DO, C. AND BATZOGLOU, S., 2008. What is the expectation maximization algorithm? *Nature Biotechnology*, (2008). (cited on page 21)

27. DOMKE, J., 2013. Learning graphical model parameters with approximate marginal inference. *PAMI*, 35, 10 (2013), 2454–2467. (cited on pages 75, 78, 81, and 82)

28. DORAI, C.; WANG, G.; JAIN, A.; AND MERCER, C., 1998. Registration and integration of multiple object views for 3d model construction. *PAMI*, 20, 1 (1998), 83–89. (cited on page 15)

29. DOUILLARD, B.; BROOKS, A.; AND RAMOS, F., 2009. A 3d laser and vision based classifier. In *RSS*. (cited on pages xvii, 2, 24, and 25)

30. DOUILLARD, B.; BROOKS, A.; AND RAMOS, F., 2009. A 3d laser and vision based classifier. In *ISSNIPC*. (cited on pages 15 and 59)

31. DOUILLARD, B.; UNDERWOOD, J.; VLASKINE, V.; QUADROS, A.; AND SINGH, S., 2010. A pipeline for the segmentation and classification of 3d point clouds. In *ISER*. (cited on pages 21 and 22)

32. FAUVEL, M.; BENEDIKTSSON, J.; CHANUSSOT, J.; AND SVEINSSON, J., 2008. Spectral and spatial classification of hyperspectral data using svms and morphological profiles. *GRS letters*, 46, 11 (2008), 3804–3814. (cited on page 20)

33. FREUND, Y. AND SCHAPIRE, R., 1999. A short introduction to boosting. *Japanese Society for Artificial Intelligence*, 14, 5 (1999), 771–780. (cited on pages 28 and 32)

34. GEIGER, A.; LENZ, P.; STILLER, C.; AND URTASUN, R., 2013. Vision meets robotics: The kitti dataset. *IJRR*, 32, 31 (2013), 1231 –1237. (cited on pages 42, 59, 65, and 66)

35. GIRSHICK, R.; DONAHUE, J.; DARRELL, T.; AND MALIK, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*. (cited on pages 16 and 59)

36. GOULD, S.; FULTON, R.; AND KOLLER, D., 2009. Decomposing a scene into geometric and semantically consistent regions. In *ICCV*. (cited on pages 16, 76, and 85)

37. GUPTA, S.; ARBELAEZ, P.; AND MALIK, J., 2013. Perceptual organization and recognition of indoor scenes from rgb-d images. In *CVPR*. (cited on page 59)

38. HARALICK, R., 1979. Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67, 5 (1979), 786 – 804. (cited on pages 30 and 51)

39. HIMMELSBACH, M.; MULLER, A.; LUTTEL, T.; AND WUNSCHE, H., 2008. Lidar-based 3d object perception. In *Cognition for Technical Systems Workshop*. (cited on page 21)

40. HOIEM, D.; EFROS, A.; AND HEBERT, M., 2007. Recovering surface layout from an image. *IJCV*, 75, 1 (2007), 151–172. (cited on page 16)

41. HONG, G. AND HANG, Y., 2007. Combination of feature-based and area-based image registration technique for high resolution remote sensing image. In *IGARSS*. (cited on page 15)

42. HU, H.; MUNOZ, D.; BAGNELL, J.; AND HEBERT, M., 2013. Efficient 3-d scene analysis from streaming data. In *ICRA*. (cited on page 59)

43. HUYNH, C. AND ROBLES-KELLY, A., 2010. A solution of the dichromatic model for multispectral photometric invariance. *IJCV*, 1, 6 (2010), 1–27. (cited on page 19)

44. JAMES, J., 2012. *A Student's Guide to Fourier Transforms*. Cambridge University Press. (cited on page 30)

45. JUTZI, B. AND GROSS, H., 2009. Nearest neighbour classification on laser point clouds to gain object structures from buildings. In *Remote Sensing and Spatial Information Sciences*, vol. 38. (cited on page 21)

46. KANUNGO, T.; MOUNT, D.; NETANYAHU, N.; PIATKO, C.; SILVERMAN, R.; AND WU, A., 2002. An efficient k-means clustering algorithm: Analysis and implementation. *PAMI*, 24, 2 (2002), 881–892. (cited on page 22)

47. KIM, T.; SUNG, G.; AND LYOU, J., 2010. Robust terrain classification by introducing environmental sensors. In *SSRR Workshop*. (cited on pages 16, 17, and 27)

48. KOHLI, P.; KUMAR, M.; AND TORR, P., 2009. P3 and beyond: Move making algorithms for solving higher order functions. *PAMI*, 31, 9 (2009), 1645 – 1656. (cited on pages 19 and 22)

49. KOHLI, P.; LADICKY, L.; AND TORR, P., 2009. Robust higher order potentials for enforcing label consistency. *IJCV*, 82, 3 (2009), 302–324. (cited on pages 16, 19, and 59)

50. KOLMOGOROV, V., 2006. Convergent tree-reweighted message passing for energy minimization. *PAMI*, 28 (2006), 1568–1583. (cited on page 68)

51. KOMODAKIS, N. AND PARAGIOS, N., 2009. Beyond pairwise energies: Efficient optimization for higher-order mrfs. In *CVPR*. (cited on page 19)

52. KOPPULA, H.; ANAND, A.; JOACHIMS, T.; AND SAXENA, A., 2011. Semantic labeling of 3D point clouds for indoor scenes. In *NIPS*. (cited on pages 42, 50, and 75)

53. KRÄHENBÜHL, P. AND KOLTUN, V., 2013. Parameter learning and convergent inference for dense random fields. In *ICML*, vol. 28, 513–521. (cited on pages 19 and 75)

54. KUMAR, M. P.; TORR, P.; AND ZISSERMAN, A., 2005. Obj cut. In *CVPR*. (cited on page 7)

55. LADICKY, L.; RUSSELL, C.; KOHLI, P.; AND TORR, P., 2013. Inference methods for crfs with co-occurrence statistics. *IJCV*, 103, 2 (2013), 213–225. (cited on pages 19 and 59)

56. LADICKY, L.; RUSSELL, C.; KOHLI, P.; AND TORR, P., 2014. Associative hierarchical random fields. *PAMI*, 36, 6 (2014), 1056–1077. (cited on page 16)

57. LADICKY, L.; STURGESS, P.; ALAHARI, K.; RUSSELL, C.; AND TORR, P., 2010. What, where and how many? combining object detectors and crfs. ECCV. (cited on pages 16 and 59)

58. LAFFERTY, J.; A.MCCALLUM; AND PEREIRA, F., 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, 282–289. (cited on pages 17 and 42)

59. LAI, K.; BO, L.; REN, X.; AND FOX, D., 2012. Detection-based object labeling in 3d scenes. In *ICRA*. (cited on page 57)

60. LALONDE, J.; VANDAPEL, N.; HUBER, D.; AND HEBERT, M., 2006. Natural terrain classification using three-dimensional ladar data for ground robot mobility. *JFR*, 23 (2006), 839–861. (cited on page 21)

61. LEE, K. AND CHEN, L., 2005. An efficient computation method for the texture browsing descriptor of mpeg-7. *Image Vision Computing*, 23, 5 (2005), 479–489. (cited on page 30)

62. LEVIN, A. AND WEISS, Y., 2009. Learning to combine bottom-up and top-down segmentation. *IJCV*, 81, 1 (2009), 105–118. (cited on page 75)

63. LI, J. AND WANG, J., 2003. Automatic linguistic indexing of pictures by a statistical modeling approach. *PAMI*, 25, 9 (2003), 1075–1088. (cited on page 7)

64. LI, L.; SOCHER, R.; AND FEI-FEI, L., 2009. Towards total scene understanding:classification, annotation and segmentation in an automatic framework. In *CVPR*. (cited on page 7)

65. LI, S., 2009. *Markov Random Field Modeling in Image Analysis*. Springer Publishing Company. (cited on page 22)

66. LIN, D.; FIDLER, S.; AND URTASUN, R., 2013. Holistic scene understanding for 3d object detection with rgbd cameras. In *ICCV*. (cited on page 59)

67. LIZARAZO, I. AND BARROS, J., 2010. boostifuzzy image segmentation for urban landcover classification. *hotogrammetric Engineering and Remote Sensing*, 76, 2 (2010), 151–162. (cited on pages xvi and 20)

68. LOWE, D., 2004. Distinctive image features from scale-invariant keypoints. *IJCV*, 60, 2 (2004), 91–110. (cited on page 67)

69. LU, Y. AND RASMUSSEN, C., 2012. Simplified markov random fields for efficient semantic labeling of 3d point clouds. In *IROS*. (cited on page 22)

70. MACCORM, J., 2013. How does the kinect work? Available at: http://pages.cs.wisc.edu/ ahmad/kinect.pdf. (cited on page 12)

71. MASUDA, T. AND YOKOYA, N., 1995. A robust method for registration and segmentation of multiple range images. *CVIU*, 61, 3 (1995), 295–307. (cited on page 15)

72. MUNOZ, D.; BAGNELL, J. A.; AND HEBERT, M., 2012. Co-inference for multi-modal scene analysis. In *ECCV*. (cited on pages xvii, xxiii, xxiv, 2, 15, 24, 25, 26, 58, 60, 65, 66, 68, 76, 87, 92, and 93)

73. MUNOZ, D.; BAGNELL, J. A.; VANDAPEL, N.; AND HEBERT, M., 2009. Contextual classification with functional max-margin markov networks. In *CVPR*. (cited on pages xvi, 22, and 23)

74. MUNOZ, D.; VANDAPEL, N.; AND HEBERT, M., 2009. Onboard contextual classification of 3-d point clouds with learned high-order markov random fields. In *ICRA*. (cited on pages xvi, 22, and 23)

75. NAJAFI, M.; NAMIN, S. T.; AND PETERSSON, L., 2013. Classification of natural scene multi spectral images using a new enhanced crf. In *IROS*. (cited on pages 42, 46, and 62)

76. NAJAFI, M.; NAMIN, S. T.; SALZMANN, M.; AND PETERSSON, L., 2014. Non-associative higher-order markov networks for point cloud classification. In *ECCV*. (cited on page 59)

77. NIEMEYER, J.; ROTTENSTEINER, F.; AND SOERGEL, U., 2014. Contextual classification of lidar data and building object detection in urban areas. In *ISPRS JPRS*. (cited on page 22)

78. PLATT, J., 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*, 61–74. MIT Press. (cited on page 46)

79. POSNER, I.; CUMMINS, M.; AND NEWMAN, P., 2008. Fast probabilistic labeling of city maps. In *RSS*. (cited on page 59)

80. POSNER, I.; CUMMINS, M.; AND NEWMAN, P., 2009. A generative framework for fast urban labeling using spatial and temporal context. *Autonomous Robots*, 26 (2009), 153–170. (cited on pages 2 and 24)

81. RAMIREZ, L.; DURDLE, N.; AND RASO, V., 2003. Medical image registration in computational intelligence framework: A review. In *CCECE*. (cited on page 15)

82. REN, X.; FOWLKES, C.; AND MALIK, J., 2008. Learning probabilistic models for contour completion in natural images. *IJCV*, 77 (2008), 47–63. (cited on page 75)

83. RUSU, R. AND COUSINS, S., 2011. 3D is here: Point Cloud Library (PCL). In *ICRA*. (cited on pages xviii, 49, and 50)

84. SALAMATI, N.; FREDEMBACH, C.; AND SSSTRUNK, S., 2009. Material classification using color and nir images. In *CIC*. (cited on pages 19, 20, and 35)

85. SALAMATI, N.; LARLUS, D.; AND CSURKA, G., 2011. Combining visible and near-infrared cues for image categorisation. In *BMVC*. (cited on pages 19, 20, and 35)

86. SAMRITJIARAPON, O. AND CHITSOBHUK, O., 2008. An fft-based technique and best-first search for image registration. In *ISCIT*. (cited on page 15)

87. SERMANET, P., 2014. Deep learning and computer vision. In *CVPR workshops*. (cited on pages xv and 2)

88. SHAPOVALOV, R.; VELIZHEV, A.; AND BARINOVA, O., 2010. Non-associative markov networks for 3d point cloud classification. In *PCV*. (cited on page 59)

89. SHETTY, S.; SRINIVASAN, H.; BEAL, M.; AND SRIHARI, S., 2007. Segmentation and labeling of documents using conditional random fields. In *SPIE*. (cited on page 48)

90. SHOTTON, J.; WINN, J.; ROTHER, C.; AND CRIMINISI, A., 2006. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *ECCV*. (cited on page 17)

91. SILBERMAN, N.; HOIEM, D.; KOHLI, P.; AND FERGUS, R., 2012. Indoor segmentation and support inference from rgbd images. In *ECCV*. (cited on page 59)

92. SINGH, G. AND KOSECKA, J., 2013. Nonparametric scene parsing with adaptive feature relevance and semantic context. In *CVPR*. (cited on pages 16 and 59)

93. SMOLA, A. AND VISHWANATHAN, S., 2008. *Introduction to Machine Learning*. Cambridge University Press. (cited on pages 16 and 17)

94. SUNG, G.; KWAK, D.; AND LYOU, J., 2010. Neural network based terrain classification using wavelet features. *Intelligent and Robotic Systems*, 59, 3-4 (2010), 269–281. (cited on page 16)

95. TANG, I. AND BRECKON, T., 2011. Automatic Road Environment Classification. *ITS*, 12 (2011), 476–484. (cited on page 16)

96. TARRANT, S.; HART, G. P. D.; AND MCGUIRE, P., 2009. Automatic road extraction from multispectral high resolution satellite images. *Defence R&D Canada, Suffield, Ralston ALTA (CAN), C-Core*, (2009). (cited on page 35)

97. TARRANT, S.; PIERCEY, G.; HART, D.; AND MCGUIRE, P., 2009. Real-time vegetation discrimination. Defence R&D Canada, Suffield, Ralston ALTA (CAN), C-Core. (cited on pages xvi, 20, and 21)

98. TEICHMAN, A.; LEVINSON, J.; AND THRUN, S., 2011. Towards 3d object recognition via classification of arbitrary object tracks. In *ICRA*. (cited on page 21)

99. TIGHE, J. AND LAZEBNIK, S., 2013. Superparsing - scalable nonparametric image parsing with superpixels. *IJCV*, 101 (2013), 329–349. (cited on pages 76, 85, and 87)

100. TRIEBEL, R.; PAUL, R.; RUS, D.; AND NEWMAN, P., 2012. Parsing outdoor scenes from streamed 3d laser data using online clustering and incremental belief updates. In *AAAI*. (cited on page 59)

101. ÜNSALAN, C. AND BOYER, K., 2004. Linearized vegetation indices based on a formal statistical framework. *GRS*, 42 (2004), 1575–1585. (cited on pages 20 and 29)

102. UTO, K.; SEKI, H.; KOSUGI, Y.; MURASE, T.; AND TAKAGISHI, S., 2012. Human detection based on active infrared illumination. In *Global Humanitarian Technology Conference (GHTC)*. (cited on page 8)

103. UTO, K.; SEKI, H.; MURASE, T.; TAKAGISHI, S.; AND KOSUGI, Y., 2012. Development of a portable human skin detector based on active infrared illumination. In *Contemporary Materials*. (cited on page 8)

104. VAPNIK, V. N., 1995. *The Nature of Statistical Learning Theory*. Springer-Verlag New York. (cited on page 16)

105. VINEET, V.; WARRELL, J.; AND TORR, P., 2012. Filter-based mean-field inference for random fields with higher-order terms and product label-spaces. In *ECCV*. (cited on page 19)

106. WAINWRIGHT, M. AND JORDAN, M., 2008. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1, 1-2 (2008), 1–305. (cited on page 82)

107. WEGNER, J.; MONTOYA-ZEGARRA, J.; AND SCHINDLER, K., 2013. A higher-order crf model for road network extraction. In *CVPR*. (cited on page 19)

108. WEINMAN, J.; TRAN, L.; AND PAL, C., 2008. Efficiently learning random fields for stereo vision with sparse message passing. In *ECCV*. (cited on page 75)

109. WEN, P., 2008. Medical image registration based-on points, contour and curves. In *ICBEI*. (cited on page 15)

110. WOJEK, C. AND SCHIELE, B., 2008. A dynamic conditional random field model for joint labeling of object and scene classes. In *ECCV*. (cited on page 17)

111. WYAWAHARE, M. AND ABHYANKAR, P. P. H., 2009. Image registration techniques: An overview. In *International Journal of Signal Processing, Image Processing and Pattern Recognition*. (cited on page 15)

112. Xiao, J. and Quan, L., 2009. Multiple view semantic segmentation for street view images. In *ICCV*. (cited on pages 16, 57, and 59)

113. Xiong, X.; Munoz, D.; and Bagnell, J., 2011. 3-d scene analysis via sequenced predictions over points and regions. In *ICRA*. (cited on pages xvi, 22, and 24)

114. Yang, J.; Price, B.; Cohen, S.; and Yang, M., 2014. Context driven scene parsing with attention to rare classes. In *CVPR*. (cited on pages 16 and 59)

115. Yang, M. and Forstner, W., 2011. Regionwise classification of building facade images. In *ISPRS*. (cited on page 17)

116. Zhang, C.; Wang, L.; and Yang, R., 2010. Semantic segmentation of urban scenes using dense depth maps. In *ECCV*. (cited on page 57)

117. Zhang, G. and Jia, X., 2012. Simplified conditional random fields with class boundary constraint for spectral-spatial based remote sensing image classification. In *GRS*. (cited on page 17)

118. Zhang, H.; ; Wang, J.; Fang, T.; and Quan, L., 2013. Joint segmentation of images and scanned point cloud in large-scale street scenes with low annotation cost. *TIP*, (2013). (cited on pages 2, 24, and 25)

119. Zhu, X.; Zhao, H.; Liu, Y.; Zhao, Y.; and Zha, H., 2010. Segmentation and classification of range image from an intelligent vehicle in urban environment. In *IROS*. (cited on page 21)