STATISTICS OF VISUAL AND PARTIAL RANGE DATA FOR MOBILE ROBOT ENVIRONMENT MODELING

Luz Abril Torres Méndez

Department of Computer Science
McGill University, Montreal

14 December 2005

A thesis submitted to the Faculty of Science
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

ABSTRACT

This thesis presents a statistical learning framework for inferring geometric structures from images. Specifically, the proposed framework computes dense range maps of locations in the environment using only intensity images and very limited amount of range data as an input. This is achieved by integrating and analyzing the statistical relationships between the visual data and the available depth on terms of small patches. The scientific issue is to represent this correlation such that it can be used to recover range data where missing. Markov Random Fields are used as a model to capture the local statistics of the intensity and range.

Experiments on real-world data are conducted under different configurations to demonstrate the feasibility of the method. In particular, our application is in mobile robotics, where inferring the 3D layout of indoor environments is a critical problem for achieving exploration and navigation tasks. The modeling of a large-scale environment involves the acquisition of a huge amount of range data to extract the geometry of the scene, and is often performed using sophisticated but costly hardware solutions. This task is physically demanding and time consuming for many real systems. By using the proposed framework, it is demonstrated that we can learn the geometric characteristics of the environment from the incomplete sensory data to build a 3D model of it.

The contributions of this thesis are mainly three: First, it demonstrates the viability of the use of very limited range data together with intensity to recover complete dense range maps. Second, it presents a complete framework for building a

3D model of an indoor environment using a mobile robot. And third, it analyses and outlines the advantages and limitations encountered when dealing with large indoor environments.

An additional contribution is the use of the method we propose for range estimation to an alternative problem: color correction and augmentation with the specific application to underwater images.

RÉSUMÉ

Cette thèse présente une structure d'apprentissage statistique pour déduir les structures géométriques des images. Spécifiquement, la structure proposée calcule des cartes de distance denses des emplacements dans l'environnement en utilisant seulement des images d'intensité et une quantité très limitée de données de distance comme entrée. Ceci est réalisé en intégrant et en analysant les rapports statistiques entre les données visuelles et de distance disponibles aux conditions de petites pièces. Le point scientifique est de représenter cette corrélation de telle fa¸con qu'elle puisse ètre employée pour récupérer des données de distance où ce type d'information manque. Des champs aléatoires de Markov sont employés comme un modèle pour capturer les statistiques locales d'intensité et de distance.

Des expériences sur des données réelles sont effectués sous différentes configurations pour démontrer la praticalité de la méthode. En particulier, notre application est en robotique mobile, où impliquer la disposition 3D des environnements d'intérieur est un problème critique pour la réalisation des tâches d'exploration et de navigation. La modélisation d'un environnement à grande échelle comporte l'acquisition d'une quantité énorme de données de distance pour obtenir la géométrie de la scène, et ceci est souvent effectué en utilisant des plateformes sophistiquées mais coûteuses. Cette tâche est d'une demande physique exigeante et de longue durée pour beaucoup de systèmes réels. En employant la structure proposée, il est démontré que nous pouvons apprendre les caractéristiques géométriques de l'environnement en employant des données incomplètes provenant des capteurs afin the construire une représentation fidéle de l'environnement.

Les contributions de cette thèse sont les suivanates : D'abord, elle démontre la viabilité de l'utilisation des données très limitées de distance ainsi que d'intensité pour récupérer des cartes denses de distance complètes. Deuxièmement, elle présente un cadre complet pour établir un modèle 3D d'un environnement de bureau à l'aide d'un robot mobile. Et troisièmement, elle analyse et décrit les avantages et les limitations rencontrées en traitant des grands environnements de bureau.

Une contribution additionnelle est l'utilisation de la méthode que nous proposons pour l'estimation de distance dans un problème alternatif: la correction et l'augmentation de la couleur, particulièrement son application spécifique aux images sous-marines.

ACKNOWLEDGEMENTS

First and foremost, I want to thank Gregory Dudek, for being such a wonderful supervisor and for being a great inspiration in my life. He encouraged creative thinking and to have confidence in my abilities while offering his advice and critical comments at every step. It has been a privilege to know such a great man.

I would also like to thank to my thesis committee and all other faculty members at McGill who I have the opportunity to either take courses with or get feedback about my work. In particular, I would like to thank Doina Precup, Joelle Pineau, Tal Arbel, Frank Ferrie, Mike Langer and Kaleem Siddiqi for interesting discussion, helpful suggestions and improvements to this work.

Thanks to Saul Simhon, Robert Sim, Yiannis Rekleitis, Eric Bourque, Christine Georgiades, Paul Di Marco, Matt Garden, Dimitri Marinakis, Diana Hernández, Isabelle Bégin, Junaed Sattar, Sandra Skaff, for being both my friends and lab-mates.

This thesis would not have been possible without the financial support from CONACyT (the National Council for Science and Technology), IRIS/Precarn and NSERC.

I would like to thank all my family for their love and support. I have learned so much from my parents and brother, I am forever grateful. I especially thank my husband Javier for being a pillar of strength and comfort and other wonderful things that he is to me.

for Andréa, mi pollito

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

Introduction

Surface depth recovery is a critical problem in robotics and computer vision. In the vision community, solutions to such *"shape-from-X"* problems are often based on strong prior assumptions regarding the physical properties of the objects in the scene (such as matte or Lambertian reflectance properties). In the robotics community, such depth inference is often performed using sophisticated but costly hardware solutions.

While several elegant algorithms for depth recovery have been developed, the use of laser range data in many applications has become commonplace due to their simplicity and reliability (but not their elegance, cost or physical robustness). However, it is often hampered by the fact that range sensors that provide complete ($2\frac{1}{2}$D) depth maps with a resolution akin to that of a camera, are prohibitely costly or otherwise impractical. Stereo cameras can produce volumetric scans that are economical, but they often require calibration or produce range maps that are either incomplete or of limited resolution. Thus, when building 3D models or map representations of large scenes, a desired characteristic is to simplify the way range sensor data is acquired so that time and energy consumption can be minimized. This can be achieved by acquiring only partial, but reliable, depth information.

In this thesis, we address the 3D scene recovery problem. Specifically, we integrate visual information with *very* sparse depth information and demonstrate how the underlying geometry of the scene can be characterized by the visual information,

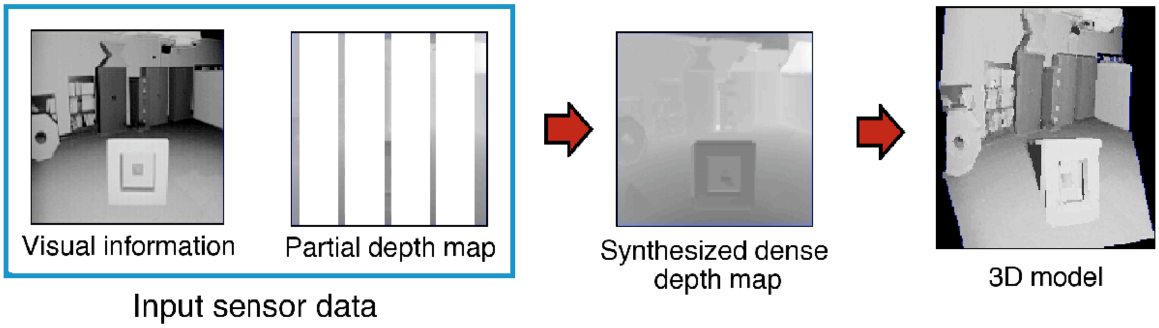


FIGURE 1.1. Building a 3D model by integrating visual and partial geometric information.

and the interaction of this visual information with the environment together with its inter-relationships with the available depth. Figure 1.1 illustrates this idea by giving an example of estimating a complete and dense range map to build a 3D model of a scene from its intensity image and associated partial range map.

We explore and analyze the statistical relationships between intensity and range data in terms of small image patches. Our goal is to demonstrate that the surround (context) statistics on both the intensity and range image patches can provide information to infer the complete 3D layout of space. It has been shown by Lee *et al.* [**83**] that although there are clear differences between optical and range images, they do have similar second-order statistics and scaling properties (i.e., they both have similar structure when viewed as random variables). Our motivation in this thesis is to exploit this fact and also that both video imaging and *limited* range sensing are ubiquitous readily-available technologies while complete volume scanning is prohibitive on most mobile platforms. Section 2 gives an overview of our method for range synthesis which is covered in more detail in Chapter 5.

Our application is in mobile robotics. In particular, we investigate the autonomous integration of incomplete sensory data to build a 3D model of an unknown large-scale [1] indoor environment. In addition, we believe this work has applications in other domains, as we will demonstrate.

[1] Large-scale space is defined as a physical space that cannot be entirely perceived from a single vantage point [**79**].

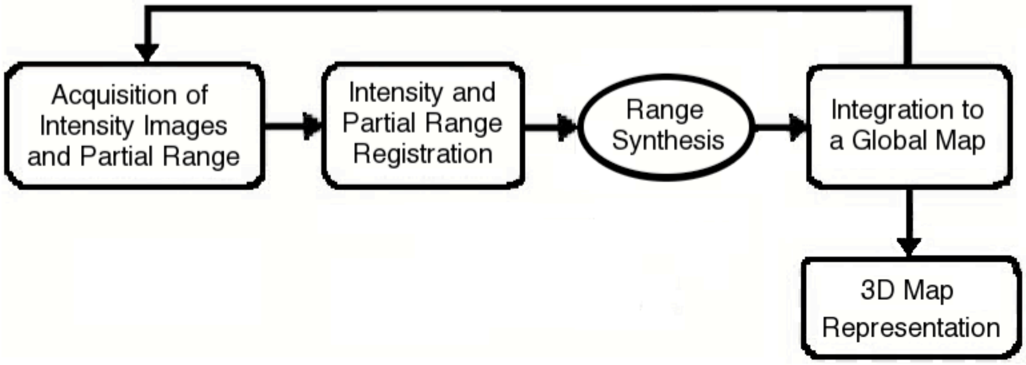

Acquisition of Intensity Images and Partial Range
Intensity and Partial Range Registration
Range Synthesis
Integration to a Global Map
3D Map Representation

Figure 1.2. Diagram showing the procedures to be followed for the 3D environment modeling problem.

Despite the fact that robot environments are three-dimensional, much of the prior work in robot mapping deals only with 2D map reconstruction due to its utility for robot navigation, fast visibility computation and the fact that it is much easier to construct. Only more recently many researches have been attracted to the idea of having a 3D model of the environment. 3D models are much richer than 2D models and facilitate the disambiguation of different places. Having a mobile robot able to build a 3D map of the environment is particularly appealing as it can be used for several important applications. For example, virtual exploration of remote locations, either for safety, automatic rescue and inspection of hazardous or inhospitable environments (e.g. the reactor of a nuclear power plant) or for efficiency reasons (museums' tours). All these applications depend on the transmission of meaningful visual and geometric information.

Figure 1.2 shows our approach for the mobile robot environment modeling problem. A brief introduction to the stages involved in our approach is given in Section 3 and Chapter 7 covers them in detail.

1. Depth Recovery in Computer Vision

The idea of extracting shape or depth information from an image has been studied in the field of computer vision since the late 1960s. Computer vision scientists were mainly interested in techniques that were supposed to reflect the way the human eye works. These techniques, known as *shape-from-X* techniques, extract depth information from intensity images by using cues such as shading, texture, retinal disparity and motion. This group of techniques are *passive*, i.e., they obtain image data without emitting energy, and are typically performed by designing mathematical models of image formation and inverting these models. These models are traditionally based on physical principles of light interaction. However, due to the highly underconstrained characteristic of the inverse problem of these principles, many assumptions about the type of surface and albedo need to be made, which may not be all suitable for the complex real scenes. A greater understanding of how real images are formed could lead to substantial insight into how depth information may be inferred from single images.

Depth information from an object or scene can also be acquired directly by using active range sensors. These techniques are known as *active sensing* techniques. This type of range sensors provide directly precise and accurate 3D points. In addition, they are independent from external lighting conditions and do not need any texture to perform well. Range sensors are commonly used to perform scene understanding for indoor mobile robots. However, these sensors tend to be expensive, the data acquisition process slow, and normally of limited spatial resolution. A picture of a big expensive 3D laser scanner is given in Figure 1.3. On the other hand, intensity images have high resolution that permits very accurate results on well-defined targets. They are easy to acquire and provide texture-maps based on real color-photographs.

1.1. Fusing passive and active sensing techniques

The fundamental problem of obtaining accurate depth from traditionally approaches such as shading based shape analysis or binocular stereo, remains a difficult task. Using 2D imagery alone will only provide sparse and unreliable geometric measures unless some underlying simple scene geometry is assumed. Consequently, the field of active rangefinding has grown parallel to the area of computer vision and

Figure 1.3. Commercial 3D Laser scanner, Trimble GS200. Dimensions: 34cm (D) × 27cm (W) × 42cm (H). Weight: 13.6 kg.

robotics, in an endeavor to find complementary techniques that overcome the limitations of current camera technology. By fusing 2D vision with rangefinding sensors, as first demonstrated in [**74**], a solution to 3D vision is realized -circumventing the problem of inferring 3D from 2D.

A vast body of research on 3D modeling has been focused on the fusion of intensity and range data. These works all consider the complete acquisition of 3D points from the object or scene to be modeled, focusing mainly on the registration and integration problems. One of the main problems in building (3D) map representations, is to simplify the way sensor data is acquired, to minimize time and energy consumption, yet accurately and reliably recover the 3D layout of an scene.

1.2. Image Statistics on Intensity and Range Images

Real scenes are constrained by many regularities in the environment, such as the natural geometry of objects and their arrangements in space, natural distributions of light, and regularities in the observer's position. Statistical studies may be helpful for understanding these regularities, which are not obvious from physical models.

Statistical methods have enjoyed a great deal of recent success in their applications to problems in computer vision. However, few studies have been made into the statistical relationship between images and range images (depth images). Those few studies have uncovered meaningful and exploitable statistical trends in real scenes which may be useful for designing new algorithms in surface inference, and also for understanding how humans perceive depth in the real scenes. An overview of this work is presented in Chapter 3.

2. Statistics on Visual and Partial Range Data

Our approach for range estimation is based on the assumption that the pixels constituting both the range and intensity images acquired in an environment can be regarded as the results of pseudo-random processes, but that these random processes exhibit useful structure. In particular, the assumption that range and intensity images are correlated, albeit in potentially complicated ways, is exploited. A second assumption made is that the variations of pixels in the range and intensity images are related to the values elsewhere in the image(s) and that these variations can be efficiently captured by the neighborhood system of a Markov Random Field. Both these assumptions have been considered before [**63, 45, 164, 44, 68**], but they have never been exploited in tandem. Texture synthesis [**45, 164, 44**] and digital inpainting [**13, 14, 31**] (see Section 5 of Chapter 4), are quite similar to the range estimation problem, although the domain and approach are quite different. In [**4**], a learned representation of pixel variation for perform resolution enhancement of face images is presented. The processes employed to interpolate new high-resolution pixel data are quite similar in spirit to what is described in this research, although the application and technical details differ significantly. The work in [**153, 60**] on learning the relationships between intrinsic images is also related.

Markov Random Fields (MRF) provide a methodological framework which allows the images from different sensors to be merged in a consistent way. In this thesis, we demonstrate that a natural way to incorporating spatial correlations into the range synthesis process is to use Markov random fields as *a priori* models.

The appeal of Markov random fields for range estimation comes from their explicit attempt to model interactions and relationships between neighboring parts of the data space. By knowing the pixel intensity values of the scene, and only partial range data coming from part of the objects and background, we can make an intelligent guess to recover the rest of the object shape. Markov random fields let us model the relationships between intensity and observed range directly and explicitly. These relationships are made similar to or, when appropriate, different from, relationships among other observed data. This feature also makes Markov random fields well suited for modeling spatial data, i.e., data that come from different spatial locations.

3. Our application: Mobile Robot Environment Modeling

In order to successfully achieve its tasks, a mobile robot depends on the environmental information gathered by its exteroceptive sensors, such as laser rangefinders, ultrasonic sensors, CCD cameras, etc., either separately or in combination. However, one of the greatest challenges to conventional mobile robotics is perception: mobile robots can travel across much of earth's man-made surfaces, but they cannot perceive the world nearly as well as humans and other animals. Moreover, perception does not just involve sensing, but also the *interpretation* of the sensed data in meaningful ways.

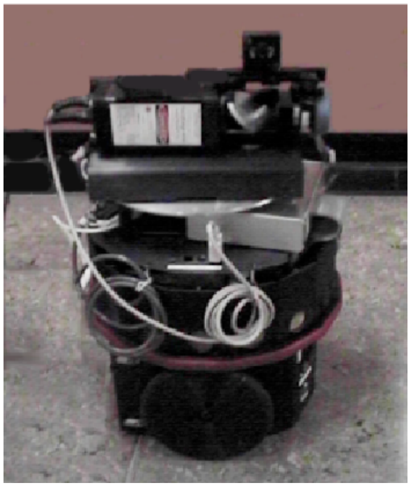


FIGURE 1.4. A picture of our mobile robot with the 2D laser range finder and the camera mounted on it.

Data Acquisition and Registration

While there has been enormous progress on the automated acquisition of environmental models, the process of acquiring information about the robot's environment, in particular 3D data, is often costly or labor intensive in practice. In our approach, a mobile robot navigates its environment. At each pose, it collects a set of intensity images and a small amount of 3D data. The intensity images are combined into a panoramic mosaic. Since the intensity and range data are coming from different types of sensors, they must be aligned or registered in order to be able to estimate a complete and dense range map. In this thesis, we present a robust image-based registration technique that takes advantage of the way sensors are arranged and how sensor data is acquired.

Range synthesis

After registering the intensity and partial range information at every robot pose, we apply our range synthesis method. A crucial aspect is the sampling strategy used when the robot is at particular pose. The denser the sampling the easier the range map estimation. However, since our goal is to minimize time and energy consumption in the data acquisition process, we want to be strategic in sampling the 3D data. We use a heuristic that is based on the distance the robot is from the objects in the scene. Basically, the closer the robot is to the objects, the denser the sampling is, and the further the robot is from the objects, the sparser the sampling of 3D data. The advantages and limitations of this heuristic are analyzed.

Data Integration

In general, only partial and imperfect information can be obtained from real sensors, thus integration of multiple observations of the same entity is required to compute an estimate of the photometric and geometric characteristics of the scene. The output from the previous stage is a set of panoramic mosaics with depth or 3D panoramas. These 3D panoramas are integrated in a common reference frame. We take

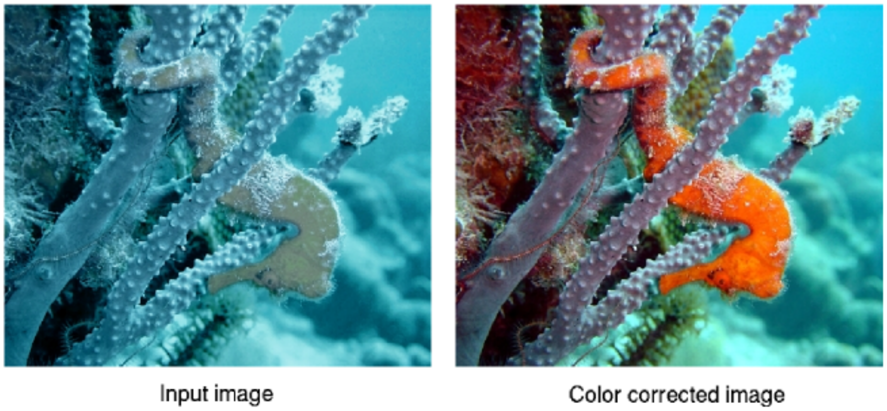


FIGURE 1.5. Color correction of underwater images. Left image depicts the input color depleted image. Right image is the color corrected image.

a hybrid approach for data integration that is based on matching range scans while simultaneously matching intensity features on the panorama mosaics.

4. Alternative Application: Color Correction

We apply the proposed method for range estimation to an alternative application: color correction. In particular, we apply this to the color correction of underwater images such as those collected by our swimming robot AQUA [**64, 43**]. For example, images acquired in deep water tend to be almost achromatic. Figure 1.5 shows to the left, an example of a color depleted image, and to the right, the color corrected image after applying our method. For many aquatic robot tasks, the quality of the images is crucial. Our method corrects the color of images by using a Markov Random Field (MRF) to represent the relationship between color depleted and color images. The parameters of the MRF model are learned from the training data and then the most probable color assignment for each pixel in the given color depleted image is inferred by using belief propagation (BP). This allows the system to adapt the color restoration algorithm to the current environmental conditions and also to the task requirements. Chapter 8 describes in detail our approach for color correction and presents experimental results on a variety of underwater scenes.

5. Problem Statement

This thesis answers three questions. First, how can the statistical nature of visual (photometric) surround or context provide information about its geometric properties? In particular, how can the statistical relationships between intensity and range data be modeled reliably such that the inference of unknown range be as accurate as possible? Second, how can the data collected at different robot poses in a large-scale scene be fused and modeled such that the models we build are useful for robotic navigation and localization? Third, based on the previous question, how can such models be constructed automatically, particularly when very limited range information is available with respect to intensity information and no prior information concerning the position of the robot is known as it collects training data?

The research presented in this thesis focuses primarily on wheeled mobile robots that navigate in indoor environments. The goals of this thesis are to answer the above questions by exploiting assumptions about the world and about the robot that are as general as possible.

A framework for computing solutions to the questions at hand will be developed, with a discussion of the motivation for each component. The solutions will in turn be validated experimentally.

6. Contributions

The following is a list of key contributions made in this dissertation.

- The first statistical model for the spatial structure of the surround of intensity and partial range data using Markov Random Fields;
- The first two algorithms for estimating depth information where only intensity is known, based on the above mentioned statistical model, that does not rely on strong assumptions concerning specific surface geometries or surface reflectance functions;

- The application of this model to the mobile robot environment modeling problem. A complete framework of data acquisition, registration and integration is described, and the viability of using a limited amount range data together with intensity to build a 3D model or map of a large-scale indoor environment is demonstrated.

Additional contributions are:

- The design of a physical framework to rapidly acquire visual and partial geometric information is presented. This framework is composed of a camera and a laser rangefinder mounted on a mobile robot. We demonstrate that our framework to acquire the sensor data also facilitates the registration process between the input data.
- A robust image-based algorithm to register intensity and partial range data that does not require previous sensor calibration is developed. Since the input data comes from different types of sensors, the registration algorithm must account for the different resolutions and projections of the input data. Also, the registration must be carried on a variety of data subsampling, going from dense to very sparse.
- A demonstration and experimental validation of the feasibility of using the proposed model for the spatial structure to a computer vision problem: color correction of underwater images. To this end, we define an algorithm for enhancing underwater images that suffer for degradation due to optical scattering resulting in an image that is bluish, blurry and out of focus.

7. Statement of Originality

Portions of the results presented in this thesis work have appeared previously in, or are currently in submission [**158, 156, 160, 159, 157, 64, 43, 123**]. In some cases, I have co-authored papers with the team of the AQUA project, that employs the proposed method for color correction of underwater images. In those works, the distinction of our individual contributions is explicit.

My published work also contains results that are not reported in this thesis. In most cases, new results are presented here duplicating the original experiments (Chapter 6), but using improvements to the original algorithms presented in the published work.

8. Organization of the thesis

This thesis presents a statistical learning method that combines visual and partial range information for the depth recovery problem. The previous sections of the introduction provided the gist of what is to come in the thesis. In short, the thesis is organized as follows:

Chapter 2 provides an overview of the state of the art on the *depth* or *shape recovery* problem. Brief descriptions of relevant work to date, their main contributions and limitations are given.

Chapter 3 is a review of statistical approaches that use images for depth recovery. This includes a description of prior studies exploring the statistical distribution and correlations between intensity and range images are given, in order to provide a solid foundation for our image-based statistical approach for 3D inference.

Chapter 4 presents principles and concepts needed to adapt Markov Random Field models to the method proposed for range synthesis in this dissertation. It also presents some relevant applications of Markov Random Fields to computer vision problems that are related to or serve as an inspiration of our proposed method.

Chapter 5 introduces the proposed statistical model to infer dense range maps of man-made scenes using only intensity images and sparse partial depth information. This chapter represents the main contribution of this thesis. In the first part of the chapter, the Markov Random Field (MRF) model is described for the spatial distribution of intensity and range data. The second part of the chapter presents the MRF-MAP approach using Belief propagation that greatly improves the range synthesis results when using only the deterministic MRF-MAP approach.

Chapter 6 presents experimental results on real data of indoor scenes. A variety of sampling strategies of the range data are experimented and analysis on the results according to these samplings are given.

Chapter 7 presents an application of our range synthesis method. Specifically, the 3D modeling of an indoor environment under the context of mobile robotics. It describes a complete framework of our approach for environment modeling: the data acquisition setup used to gather the visual and partial geometric information and the algorithm to register the intensity and partial range data. It also describes the depth recovery process by using our range synthesis method. Finally, the process of integrating the sensor data from multiple viewpoints to a common coordinate frame is given. In other words, it describes the integration of each of the dense maps at each robot pose to a global 3D dense map of the environment. Along with each stage, experimental results on real data collected in a building from our own mobile robot are shown.

Chapter 8 presents an alternative application of our proposed statistical learning method, for the problem of color correction of underwater images. This chapter can be read independently of the other chapters. It shows a formulation for color recovery and more general enhancement as an energy minimization problem using learned constraints.

Chapter 9 gives a summary and some conclusions on the research contained in this disssertation. In the thesis, choices have been made regarding the topics of research. Necessarily many other interesting topics have been left aside. This chapter provides a list of topics for possible future research.

CHAPTER 2

Passive and Active Sensing Methods for Depth and Shape Recovery

As the main objective of this dissertation is to infer depth information by combining passive and active sensing techniques, this chapter reviews several depth and shape recovery methods that use these sensing techniques either individually or by fusing them, highlighting their advantages and disadvantages.

1. Introduction

Surface depth recovery is one of the central problems of computer vision and robotics research. Techniques for inferring the 3D layout of space are divided by the type of sensor they use. In general, two types of sensors are commonly used to infer the geometry of an object/scene: *passive* and *active* sensors. Passive sensors do not emit energy for the purpose of sensing, only receive it. A passive sensor relies on the environment to provide information. Most robot vision is done using CCD cameras which detect visible light from the surrounding world. Methods that infer depth from images include all the monocular shape-from-X methods, structure from motion, hazing and focus/defocus. On the other hand, active sensors emit a special source of energy such as laser or ultrasonic waves, and then receive once it has interacted with the object to digitize. Such active sensors include laser, sonar, radar and structured light.

The techniques described in the following sections are not intended to be exhaustive; we will mention briefly only the prominent approaches.

2. Passive Depth Estimation: Using Visual Cues

Depth estimation can be relative or absolute. Examples of techniques that recover relative depth information are: shape from shading [**70**], from texture [**147**], from edges and junctions [**7**], from symmetric patterns [**137**] and from other pictorial cues such as occlusions, relative size, and elevation with respect to the horizon line [**110**]. Most of these techniques apply only to a limited set of scenes. Literature on absolute depth estimation is also large but the proposed methods rely on a limited number of sources of information (e.g., binocular vision, motion parallax, and defocus). Human vision, under normal viewing conditions, can provide a rough estimate of the absolute depth of a scene even in the absence of all these sources of information (e.g., when looking at a photograph). We use one additional source of information for absolute depth estimation: the size of known objects like faces, bodies, cars, etc. In computer vision, however, this strategy requires segmenting the image. In general, image segmentation and object recognition processes remain difficult and unreliable. In the following, a subset of these techniques are described in more detail in order to highlight their advantages and disadvantages compared to those of the active sensing techniques.

2.1. Shape-from-Shading

Shape-from-shading (also known as photoclinometry) has attracted substantial attention from the research community since the 1970s. The problem of estimating the shape of an object from its shading was first introduced by Horn [**70**]. He defined the mapping between the shading and surface shape in terms of the reflectance function (surface albedo) $I_{x,y} = R(p, q)$, where $I_{x,y}$ denotes image intensity, $p = z_x$ and $q = z_y$,

z are the depth of the object, and (x, y) are projected spatial coordinates of the 3D object. The standard approach to the problem involves modeling the behavior of light as it travels through space and interacts with surfaces and then attempts to invert the image formation processes. Unfortunately, inverting this process is highly underconstrained, and various assumptions about image formation models and their parameters such as Lambertian surface reflectance, uniform albedo, and shadow-free, single-point-source illumination, have to be made for this approach to work. In the classical formulation, it is assumed that surface radiance is determined entirely by the surface orientation relative to a point light source at infinity. For a Lambertian surface, in a scene with the same reflectivity and composed of a single smooth surface, the surface shape can be recovered by inverting the following equation

$$I_{x,y} = \mathbf{N}(x, y) \cdot \mathbf{L}$$

where $\mathbf{L}$ is the light source direction, and $\mathbf{N}(x, y)$ is the normal at the surface point corresponding to an image point (x, y) (see Fig 2.1). A variety of methods have been developed for inverting the above model to obtain the depth map. However, this model is not valid when the light source is diffuse [**82**]. The first algorithm for solving the shape-from-shading problem under diffuse lighting was proposed by Langer and Zucker [**82**]. In general, surface recovery from a single image is an ill-posed problem, requiring strong assumptions (like those mentioned above) and sophisticated

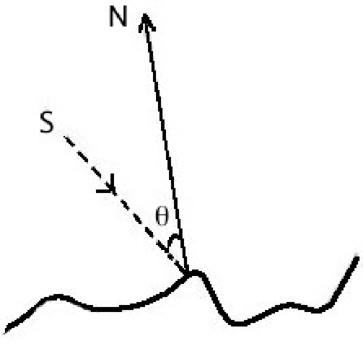


FIGURE 2.1. Lambertian reflection geometry.

mathematical methods. Moreover, the results are usually not accurate and reliable for real scenes.

2.1.1. Statistical Shape from Shading

Some exceptions to the classical shape-from-shading were presented by Lehky and Sejnowski [**84**], Atick *et al.* [**2**] and more recently by Freeman *et al.* [**59**]. In their work, shape-from-shading algorithms make use of a direct association between luminance images and the 3D models used to generate them. The image statistics were derived from computer-generated images, that make the same assumptions made by traditional shape-from-shading algorithms. These approaches suggest that a deeper understanding of the joint statistics of natural images and their associated 3D structures might be important for the successful development of a statistical approach for 3D inference. Other methods that use similar insight include those of Torralba an Oliva [**154**] and Freeman and Torralba [**60**]. In this thesis, we consider a similar approach, thus a review of the field of image statistics with focus on these methods for shape recovery is given in Chapter 3.

2.2. Shape-from-Texture and Shape-from-Specularity

Texture is another cue that provides important source of information about the 3D structure of visible surfaces, particularly for stationary monocular views. Shape-from-texture approaches have also been the focus of attention [**169, 147**]. When a texture surface is projected onto an image plane, the texture pattern is distorted systematically following laws of projective geometry. This provides a cue for 3D shape: if the visual system can measure the compression of the texture at each image location, it can recover local orientation and thus shape. This requires that assumptions must be made about the isotropy properties of the texture, however, the unpredictability of natural textures precludes the use of highly restrictive assumptions. Various methods proposed in the literature intent to make minimal assumptions, yet it is difficult that one model can cover all surface types and in particular, the huge variety of textures.

Fleming *et al.* [**56**] proposed a new visual cue that allows the visual system to estimate the shape of an object from monocular images. In their approach, specular reflections are treated a bit like textures; the authors argue that this is because specular reflections also lead to stochastic image patterns with well-conserved statistics. When the world is reflected in a specular surface, the reflection is distorted by the shape of the object. The pattern of distortion is a function of the 3D shape, just as it is with textures. Crucially, however, for specularities the compression is a function of surface curvature as well as orientation. Hence, the mapping from images to 3D shape follows different rules for specular versus textured surfaces. *Texture trajectories* [1] can thus allow the visual system to distinguish specular reflections from textures, and to estimate 3D shape for both textured and specular objects. The texture trajectory cue is weakest for spheres and planes, and strongest for objects with small Gaussian curvature but large mean curvature.

2.3. Stereo vision

One of the most remarkable properties of the human visual system is the ability to reconstruct the 3D depth of a scene from the essentially 2D retinal fields. The small positional differences, or disparities, of the images of objects on the two spatially separated retinas is used to accurately reconstruct the depth of objects.

Formally, a point $\mathbf{P}$ in the scene, at a distance $\mathbf{Z}$ from the focal plane, projects onto left and right retinas at positions x_L and x_R. The difference between these positions, or *disparity* and is related to the depth $\mathbf{Z}$, of the point $\mathbf{P}$ as follows:

$$disparity = x_L - x_R = \frac{\mathbf{B}f}{\mathbf{Z}} \tag{2.1}$$

where $\mathbf{B}$ is the baseline distance and f, the focal length.

Dense stereo vision gained popularity in the early 1990's due to the large amount of range data that it could provide [**89, 61**]. In mobile robotics, a common setup is the use of one or two cameras mounted on the robot to acquire depth information as the robot moves through the environment [**99**]. The cameras must be precisely

[1] A texture trajectory is defined in [**56**] as the pattern of compressions across an image.

calibrated for reasonably accurate results. The depth maps generated by stereo under normal scene conditions (i.e., no special textures or structured lighting) suffer from problems inherent in window-based correlation. These problems manifest as imprecisely localized surfaces in 3D space and as hallucinated surfaces that in fact do not exist.

Other works have attempted to model 3D objects from image sequences [**152, 55, 115**], with the effort of reducing the amount of calibration and avoiding restriction on the camera motion. Fitzgibbon and Zisserman [**55**] proposed a method that sequentially retrieves the projective calibration of a complete image sequence based on tracking corner and/or line features over two or more images, and reconstructs each feature independently in 3D. Their method solves the feature correspondence problem using methods based on the fundamental matrix and tri-focal tensor, which encode precisely the geometric constraints available from two or more images of the same scene from different viewpoints. A similar work is that of Pollefeys et. al. [**115**], in which a 3D model of an object is obtained from image sequences acquired from a freely moving camera. The camera motion and its settings are unknown. Their method is based on a combination of the projective reconstruction, self calibration and dense depth estimation techniques. In general, these methods derive the epipolar geometry and the trifocal tensor from point correspondences. However, they assume that it is possible to run an interest operator such as a corner detector to extract from one of the images a sufficiently large number of points that can then be reliably matched in the other images. It appears that if one uses information of only one type, the reconstruction task becomes very difficult and works well only under narrow constraints.

2.4. Structure from Motion

Structure from motion (SFM) is another widely recognized approach to the retrieval of 3D structure of a scene from optical flow resulting from unknown camera motions. Ullman [**161**] summarizes earlier work in SFM, and Oliensis [**105**] gives a critique of recent techniques. Ideally, SFM allows arbitrary camera motion and only requires that the scene is static. However, a serious drawback of this approach is that it is often unstable and hard to exploit in practice. As a generalization of the method, *structure and motion* is presented in [**40, 114**]; uncalibrated image sequences acquired with a hand-held camera are mapped into visual models of 3D scenes, illustrated by examples showing a single building or a statue. Dick *et al.*'s method [**40**] uses building primitives such as doors or windows within a recovery process. Zhu *et al.* in [**174**], discuss the use of an uncalibrated sequence for panoramic images for 3D scene reconstruction.

3. Active Sensing Techniques

The fundamental problem of obtaining depth from traditionally approaches such as shading based shape analysis or binocular stereo remains a difficult task. Using 2D imagery alone will only provide sparse and unreliable geometric measures unless some underlying simple geometry is assumed. By fusing 2D vision with active rangefinding sensors, as demonstrated in [**74**], a solution to 3D vision is realised, circumventing the problem of inferring 3D from 2D.

The active sensing methods can be divided into two categories: contact and non-contact methods. Coordinate Measure Machine (CMM) is a prime example of the contact methods. CMMs consist of probe sensors which provide 3D measurements by touching the surface of an object. Although CMMs generate very accurate and fine measurements, they are very expensive and slow. Also, the types of objects that can be used by CMMs are limited since physical contact is required. Furthermore, this method is impractical when trying to model large environments containing numerous objects. The non-contact methods project their own energy source onto an object or scene, then observe either the transmitted or the reflected energy. Active stereo uses the same idea of passive sensing stereo method, but a light pattern is projected onto an object to solve the difficulty of finding corresponding points between two (or more) camera images. Scanning the object with the light constructs 3-D information about the shape of the object. This is the basic principle behind depth perception for machines, or 3D machine vision. In this case, structured lighting is sometimes described as active triangulation.

Structured-light methods citeBesl89, Chen97 project a light pattern onto a scene, then use a camera to observe how the pattern is illuminated on the object surface. Broadly speaking, the structured-light methods can be divided into scanning and non-scanning methods. The scanning methods consist of a moving stage and a laser plane, so either the laser plane scans the object or the object moves through the laser plane. A sequence of images is taking while scanning. Then, by detecting illuminated points in the images, 3D positions of corresponding object points are computed by the equations of camera calibration. The non-scanning methods project a spatially or temporally varying light pattern onto an object or scene. An appropriate decoding of the reflected pattern is then used to compute the 3D coordinates of an object.

With sonar sensors, sound is emitted, bounces off objects, and is reflected back to the sensor. The difference from when the sound was emitted and when it returns, and the speed of sound in that environment are used to calculate the distance to the object. Relevant work using sonar sensing in robotic applications include those developed by Elfes [**48**] and Crowley [**32**]. However, sonar sensors are less used nowadays due to the fact that data measurements are often inaccurate and inconsistent. This is because readings depend on the speed of sound, which varies according to atmospheric conditions such as temperature and humidity. Another problem is that the ultrasonic echoes might cause the sensor to measure totally incorrect values.

LIDAR-based laser radar systems are more accurate than sonar sensors and use the information of emitted and received laser beam to compute the depth. There are mainly two methods that are widely used: (1) using amplitude modulated continuous wave (AM-CW) laser, and (2) using laser pulses.

In particular, for man-made environments the choice of the sensor type from which precise raw range information from the scene is obtained is crucial. The system that acquired the 3D data presented in this dissertation is a LIDAR-based laser rangefinder. They are specially suitable for applications in 3D environment modeling because they acquire dense and accurate 3D data compared to passive sensing methods.

There is a vast body of research work using laser rangefinders for different applications, particularly, in the 3D reconstruction problem [**29, 26, 76, 94, 167, 104, 47, 134, 144**]. In what follows, we will describe the basic concept of laser rangefinders.

3.1. LIDAR-based Laser Rangefinders

Laser rangefinders measures distance (depth) as a direct consequence of the propagation delay of an electromagnetic wave, i.e., the laser rangefinder emits a modulated, collimated beam of laser light, and detects the reflected echo beam. The time taken for a laser beam to leave the sensor, strike a surface, and return is measured. This technology is known as *time of flight* (TOF), and has been used for decades in industrial and military applications, and more recently, for research purposes. There are basically two techniques for measuring the TOF –the *pulsed* and the *phase-shift* methods. Figure 2.2 shows how this technique can be used for range estimation by measuring the phase shift between transmitted and received signals.

Laser rangefinders provide good distance precision with the possibility of increasing accuracy by means of longer measurement integration times. The integration time is related with the number of samples in each measurement. The final measurement is normally an average of sample measures, decreasing therefore the noise associated to each single measure. Spatial resolution (i.e., the ability to distinguish two targets at different distances when placed side by side) is guaranteed by the small aperture and low divergence of the laser beam. Laser rangefinders can also provide a measure of the reflectance (amplitude) of the object being sensed. The amplitude corresponds, however, to the reflectance of the target at the wavelength of the laser beam (monochromatic, normally in the infrared), thus it hardly contains a textural description of the scene.

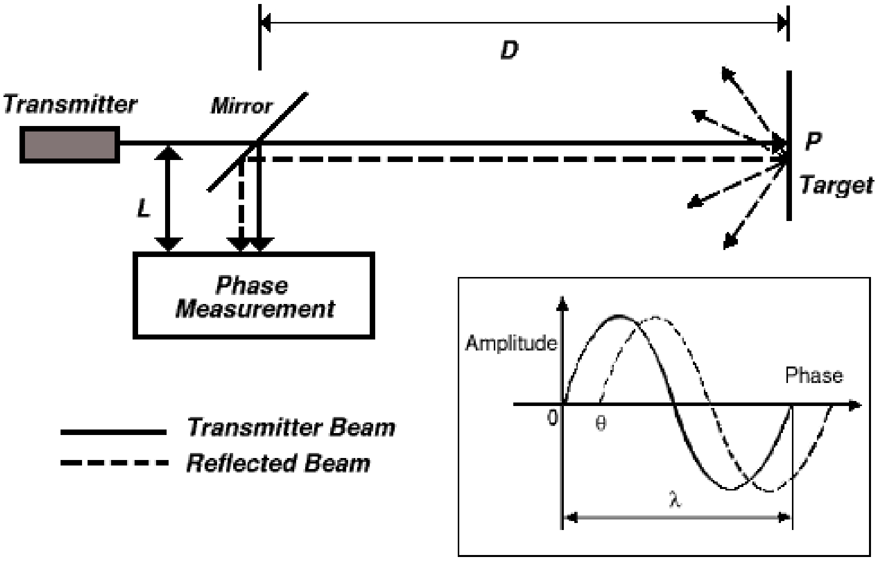


Figure 2.2. Schematic of laser rangefinding by phase-shift measurement. *Adapted from* [**140**]

Laser range finders are not free from sources of error, distance measurements can be affected by errors inherent to the sensor or to the scanning system. As a result, the measured data is not certain, and this uncertainty must be incorporated in the sensor fusion model. However, the completeness of coverage of these sensors is a more important factor that have to be taking into account when dealing with large and complex environments.

In general the following attributes must be considered carefully when choosing an active sensor:

- *Field of view and range*: How much of the world it can *reliably* measure –FOV refers to the degrees covering the horizontal and vertical axis, range is how far into the distance the sensor can measure.
- *Accuracy, repeatability and resolution*: How correct is the measurement, how often it reaches the same reading, how precisely we can measure.
- *Speed* of the device: How long will it take to get those measurements? Are the measurements averaged? How many points are captured per second?

There is a tradeoff between the number of points that are captured and averaged, and the expected standard deviation in the measurement.

- *Easy to use*: How heavy is the device? How big is it? Can it be transported as carry-on luggage on an airplane? The sensor size can affect the design/mobility of the robot and conversely, it can be difficult to install due to the robot's design/size. How straightforward is it to calibrate the instrument? How often do calibration procedures need to be performed? What facilities are required for this? How often does it need to be factory-serviced?
- *Responsiveness in the target domain*: How well does the sensor work in the environment in which the robot is supposed to navigate?
- *Physical robustness*: How will performance be affected by temperature, humidity, dust, vibration? (e.g., on a moving shaking vehicle). Scanning in environments where toxic chemicals, nuclear radiation or other agents can degrade scanner performance is another level of consideration.
- *Power consumption*: Sensors are a major drain of power, generally passive sensors use less power than active. There is a trade off between locomotion ability and sensor capability.
- *Hardware reliability*: Sensors often work best (or with reasonable reliability) within a certain range of physical conditions, such as temperature and moisture.
- *Eye safety*: Buyers need to consider the potential for workers or the public to be exposed to the laser beam. The setting in which the laser scanner will be used is a guide to what level of eye safety is appropriate.
- *Cost*: Of course no one makes a purchase decision without considering this. Scanner list prices start at about $30,000 and can be as much as $200,000 fully loaded with software, maintenance, training and support.

3.1.1. Laser scanners

A large number of possible 3D scanners are available on the market. However, cost is still the major concern. An overview of different systems available to 3D shape of objects is presented by Blais [**18**], highlighting some of the advantages and disadvantages of the different methods.

Laser Range Finders directly map the acquired data into a 3D volumetric model thus having the ability to partly avoid the correspondence problem associated with visual passive techniques. Indeed, scenes with no textural details can be easily modelled. Moreover, laser range measurements do not depend on scene illumination.

3.1.2. Acquiring Data: Range Image

The sequence of images taken by the laser rangefinder during a scan can be stored in a compact data structure called *range image*, also known as *range map*, *range data*, *depth map*, or *depth image*. In most scanners, a range image is a set of distance measurements arranged in a $m \times n$ grid. Typically, for the case of laser scanners, m is the number of horizontal scan lines (rows) in the sequence, and n is the total number of vertical scan lines (i.e., stripes). We can also represent a range image in a parametric form $r(i, j)$ where r is the column coordinate of the measured point at the ith row in the jth stripe. Sometimes, the computed 3D coordinate (x, y, z) is stored instead of the column coordinate of the measured point.

4. Fusing Active and Passive Sensing for Depth Recovery

Passive capture compared with active sensing exhibits some significant advantages in terms of cost, practicality, complexity of use and integration to a larger system. These are the characteristics that make cameras the more attractive sensor in the market. For the case of stereo cameras, unless accurate calibration and precise information on their pose are known, 3D measurements can only be recovered for a sparse set of features. As these features correspond to discontinuities in the intensity of the images and are usually associated with ”significant” structure in the scene they can be reliably matched and tracked along successive frames.

In the 90s, most existing approaches were based on a single sensor [**35, 76, 141, 55**]. In robotics, inter-robot communication [**5, 125**] is being used to overcome the limitations of using single robot systems (which obtain sensor readings from one position at a time). Depending on the application and the complexity of the object or scene, achieving geometric correctness and realism may require data collection from different sensors (passive and/or active) as well as the correct fusion of all these observations. The idea of using more than one sensor to complement the data of one sensor with that of another is not new [**103**]. However, methods for data fusion are not as well developed as those for the design and analysis of individual sensors.

Independent of the level of representation used, a variety of popular mathematical techniques for sensor fusion appear in the literature: for example, probability and Bayesian inference [**34**], Dempster-Shafer theory of evidence [**135**]. These established techniques are then incorporated into a framework for transforming data into a common coordinate system, producing verdicts on the correctness of the various sources, and allowing stable estimation of the parameters of a problem. An important aspect of each technique is the way in which it models uncertainty in sensor information.

There has been substantial interest in fusing intensity and range information for 3D model building and virtual reality applications [**121, 47, 134, 86, 144**] with promising results. Specifically, the use of dense intensity images to provide photometric detail which can be registered and fused with range data to provide geometric detail. However, there is one notable difference of our work in this dissertation with those in the literature: the amount of range data acquired is *very* small compared to the intensity data. There is no prior work that uses partial depth measurements together with the intensity images, for *complete* and *dense* scene recovery.

CHAPTER 3

Image Statistics for Shape Recovery

Natural images (including both natural landscapes and man-made environments [1]) exhibit strong statistical regularities that are exploitable by biological and machine vision systems. Only an infinitely small fraction of all possible pixel combinations have interpretations as natural (realistic) images –the structure and content of a realistic image deviates from those of random images in specific ways to form a sparse subset of all randomly formed images. The fact that images exhibit great variability on the patterns they represent has made it clear that exact mathematical models may not be practical, and a statistical approach needs to be adopted.

Despite the growing success of statistical methods in vision, there has been relatively little investigation into joint statistics of range and intensity images. This chapter presents a brief background on the statistics of natural images, and describes prior work that explore the statistical distribution and correlations between 2D images and 3D structures in order to provide a solid foundation for our image-based statistical approach for 3D inference.

[1] The fact that images containing man-made structures are considered natural images may be a bit confusing. Natural images are those that are taken from a natural environment, i.e., an environment that is commonly encountered by a particular organism [**130**].

1. Introduction

The statistics of natural images have been studied to understand how their properties influence the human visual system. Images' statistical properties are usually studied by collecting a number of such images in an ensemble and computing first or, often, power spectrum statistics on them. One advantage of statistical approaches is that they can provide a unified view of learning, classification and generation. The visual world offers a tremendous amount of data that the human visual system process quickly and continuously. People can quickly infer low-level scene properties such as shape, motion, lightness and occlusion boundaries. However, it has been shown that images that do not behave statistically as natural images are harder for the human visual system to interpret.

The computation of statistics of image space has a long history. The efforts of finding explanations for the patterns exhibited by natural images through observing, gave birth to the field of natural image statistics [**51, 143**].

Most images where objects lie are expected to be of relatively large-scale in practice. For that reason, many statistical approaches will need probability models that capture *essential* image variability and yet are computationally tractable. Filtering methods based on local neighborhood statistics such as *median filtering* can be found throughout the literature. Image contrast enhancement techniques based on histogram equalization have also been explored and are in use in medical as well as other production environments [**112**]. There exist numerous methods for performing segmentation and classification of images based on statistical pattern recognition [**42**]. Statistically based relaxation filters founded on the theory of Markov processes (Markov Random Fields) [**63, 173**], and expectation-minimization methods [**36**] also have a long history.

2. Background

To quantitatively analyze the aspects of natural images that make them predictable (i.e., scene properties such as texture, shape and surface structure), a statistical description needs to be adopted. An important property of a statistical description is that it does not apply to individual images, but only to a set of sample images. Once having a well chosen and large sample set, one need to decide which image statistics to calculate. Natural image statistics can be characterized by their order, which describes the distribution of image values at a single position (first order), at two separate positions (second order), or at three or more positions in images (high order) [**130**]. A brief description of these order statistics is given next:

First order statistics treat each pixel in the image independently, so that, for example, the distribution of intensities encountered in natural images can be estimated. The average intensity in an image is an example of the first-order statistics.
Second-order statistics measures the correlation between pairs of pixels. Natural images have the property that the intensity at separate positions is not independent. The most popular second order statistics are the autocorrelation function and the power spectrum (described later).
Higher order statistics are used to extract properties of natural scenes which can not be modeled using first and second order statistics. These properties include lines and edges.

The power spectrum is one of the most well studied natural image statistics. It is obtained by computing the Fourier transform of an image and multiplying each element of the transform by its complex conjugate. Averaging over all directions gives power as function of frequency. For most natural images, plots of this quantity can be easily fitted by a straight line with a slope of around $1/f^2$, where f is the spatial frequency (usually measured in cycles per image) [**21, 50, 81**]. This special property of natural images can not generally be obtained from random images (e.g., random noise images would produce a flat power spectrum). The $1/f^2$ spectral slope of natural images means that equal power is encoded in each frequency band. It also implies that natural images are statistically scale-invariant.

Torralba and Oliva [**155**] studied the statistics of natural images and concluded that these statistics strongly vary as a function of the interaction between the observer and the world. In particular, they show that second order statistics of images are correlated with scene scale and scene category and provide information to perform fast and reliable scene and object categorization. Statistical regularities might be a relevant source for top-down and contextual priming, very early in the visual processing chain. Their results show how visual categorization based directly on low-level features, without grouping or segmentation stages, can benefit object localization and identification and can be used to predict the presence and absence of objects in the scene before exploring the image.

The global image statistics are correlated with the objects present in the scene. Some studies show that the spectral slope for scenes containing man-made objects is slightly different from that of images representing natural environments [**155**]. Figure 3.1 shows the average power spectra of natural landscapes and man-made environments. It can be noted that vertical and horizontal orientations are more frequent than obliques [**3, 148, 130, 106**]. The anisotropic distribution of orientations is also compatible with neurophysiological data showing that the number of cells in early cortical stages varies in regard to the spatial scale tuning and the orientation (e.g. more vertical and horizontal tuned cells than oblique in the fovea [**39**]. Besides the fact that object shapes have an impact on the global image statistics, there exists strong correlation between the objects present and their context [**155**].

2.1. Scene scale and image scale

It has also been shown in [**155**], that image statistics also vary when considering scenes at different scales. Close-up views on man-made objects tend to produce images that are composed of flat and smooth surfaces. Consequently, the energy of the power spectra for close-up views is concentrated mainly in low spatial frequencies.

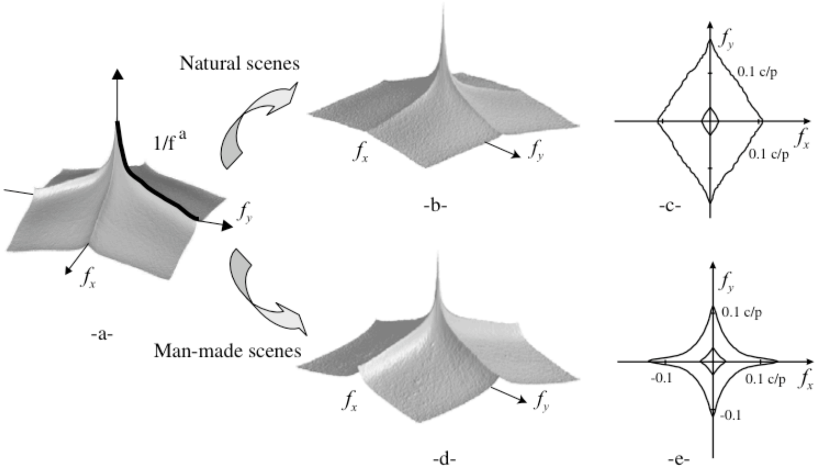


FIGURE 3.1. (a) Mean power spectrum averaged from 12000 images (vertical axis is in logarithmic units). Mean power spectra computed with 6000 pictures of man-made scenes (b) and 6000 pictures of natural scenes (d);(c) and (e) are their respective spectral signatures. The contour plots represent 50 and 80% of the energy of the spectral signature. The contour is selected so that the sum of the components inside the section represents 50% (and 80%) of the total. Units are in cycles per pixel. *From* [**155**].

As distance between the observer and the scene background increases, the visual field comprehends a large space, that is likely to encompass more objects. The images of man-made scenes appear as a collection of surfaces that break down into smaller pieces (objects, walls, windows etc). Thus, the spectral energy corresponding to high spatial frequencies increases as the scene becomes more cluttered due to the increase, with distance, in the area covered by the visual field. In contrast, spectral signatures of natural environments behave differently while increasing depth. Figure 3.2 shows that when the distance between the observer and the background grows, natural structures become a single entity and smoother (small grain disappears due to the spatial sampling of the image). Therefore, on average, with an increment of distance, the level of clutter decreases, as does the energy in the high spatial frequencies. In addition, the pattern of orientation varies with the scale. Close-up views on natural structures have a tendency to be isotropic in orientations (and the point of view of the observer is unconstrained). As distance grows, there is an increased bias towards vertical and horizontal orientations, together with the point of view of the observer

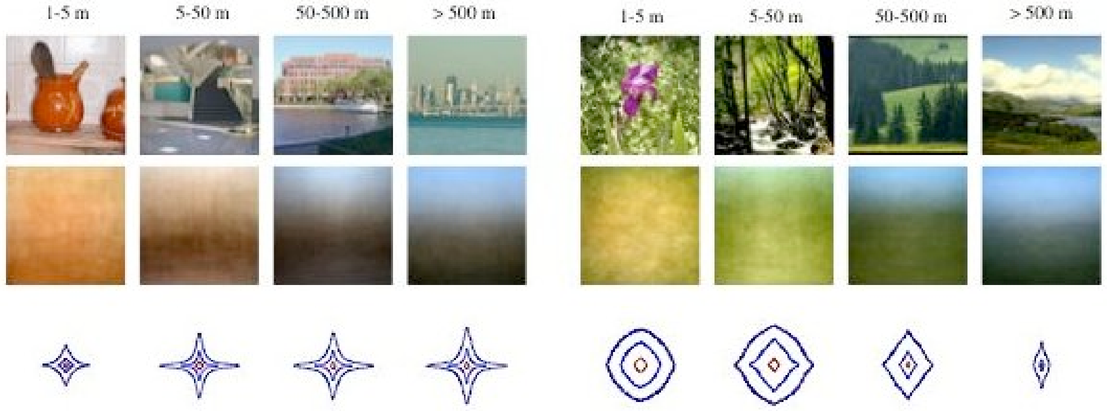


FIGURE 3.2. The effect of distance in natural image statistics. The first row shows examples of the intensity images of each category and the averaged spatial images are shown in the second row. The third row are the spectral signatures as a function of scene scale. Scene scale refers to the mean distance between the observer and the principal elements that compose the scene. Each average image and spectral signature was calculated with 300 – 400 images. From [**155**]

.

becoming more constrained. As distance continues to increase, energy is concentrated mainly in vertical spatial frequencies, as very large environmental scenes are organized along horizontal layers. In order to recognize the scene or to navigate such panoramic environments, faced with point of view limitations, an observer might consider looking towards the horizon to visually embrace the whole scene.

2.2. Non-stationary statistics

Another important characteristic of natural images is how the image statistics change with spatial location. When considering all the possible directions of the eye or camera, statistics of natural images are usually scale invariant and stationary (the features are equally distributed in regard to locations [**52, 53**]). This is the case indeed with the statistics of images of close-up views of objects that are, on average, stationary, as there is no preferred point of view for the camera. However, for images of scenes that embrace a large volume, the probable points of view that a human observer will adopt become much more constrained, because of his height

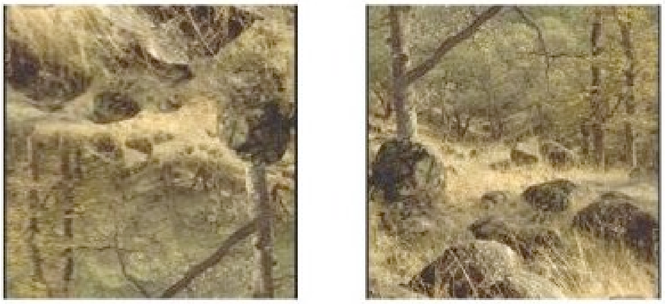


FIGURE 3.3. Top-down effect on depth judgments. The left image is generally recognized as close-up view on bushes and maybe a spider's web on top, but it is actually a rotated version of the right image. The right image is recognized as the inside of a forest, corresponding to a larger distance than the image on the left (see text for explanation). From [**155**].

and his probable location (on the floor). If the task of the observer is to recognize the identity of a large scale scene, most of the useful information will be given while looking towards the horizon.

Therefore, different image statistics will characterize the top and bottom half of the image (e.g., smooth texture of the sky, long vertical contours of skylines at the top, cluttered forms at the bottom). Figure 3.3 shows an example of how image inversion affects the perception of the absolute depth of a scene. The upside-down inversion affects the perception of concavities and convexities due to the assumption of light from above, and, therefore, modifies the perceived relative 3D structure of the scene. But, moreover, the incorrect recognition affects the absolute scale of the perceived space. The image on the left appears as a closer structure than the image on the right for most observers.

3. Image-based Statistical Approaches for 3D Inference

Real scenes are complex and constrained by many regularities in the environment, such as the natural geometry of objects and their arrangements in space, natural distributions of light, and regularities in the observer's position. A greater understanding of how real images are formed could lead to substantial insight into how depth information may be inferred from single images. Statistical studies may be helpful for understanding these regularities, which are not obvious from physical models.

3.1. Shape Recipes

Depth information obtained from other cues may be used to refine the shape-from-shading inference process. Recently, a method that takes advantage of this idea has been proposed by Freeman and Torralba [**60**]. In their work, new low-dimensional representations, called *scene recipes* are used to learn the relationships between a low resolution range image and a high resolution intensity image to infer a more accurate high resolution range image. Such a problem may arise when range data is derived through stereo algorithms. Stereo algorithms suffer from the lack of local surface texture due to smoothness of depth constraint, or local missmatches in disparity estimates. Thus, the stereo methods only provide a coarse depth map, which is often accurate in the low frequency bands, but less accurate in the high frequency bands. In order to compute a better depth map, their method integrates both the high resolution image and the low resolution range image, which are decomposed into a steerable wavelet filter pyramid. This filter breaks the image down according to scale and orientation, with minimal aliasing between subbands [**58**]. The learning phase involves the search for the best high range band from the image band, and it is done using linear regression between the highest frequency band of the available low-resolution range image and the corresponding band of the intensity image. The hypothesis of the model is that this filter can then be used to predict high frequency range bands from the high frequency image bands. Specifically, let $z_{n,o}$ and $i_{n,o}$ be the n^{th} highest resolution subband of the range and intensity images, respectively, at orientation o ($n = 0$ is the highest resolution band of the intensity image). Then the filter $k_{n,o}$ is learned at the highest available resolution range band to give us $z_{n,o} \approx k_{n,o} \star i_{n,o}$, where $\star$ denotes convolution. Higher resolution subbands of the range image are inferred by the equation

$$\hat{z}_{n-m}, o = c^{-m}(k_{n,o} \star i_{n-m,o})$$

where $c = 2$. This model is motivated by the linear Lambertian shading model.

Since linear regression and convolution are both linear operations, it is expected that the underlying model behind shape recipes has a natural consequence on the second order statistical structure of the relation between images and range images. A model of image/range statistics assumed by the shape recipe technique was derived in [**118**] and their result is stated here. Let h_{ii} be the autocorrelation of the image, and let h_{zi} be the cross-correlation between the image and the range image. Then, $h_{zi}(\vec{\Delta}x)$ is defined by $(1/N)\sum i(\vec{x})z(\vec{x}+\vec{\Delta}x)$, where $z(x, y)$ is the full resolution range image, which is assumed to be centered (zeromean). Let II and ZI be the Fourier transforms of h_{ii} and h_{zi}. Then the shape recipe hypothesis implies that

$$ZI(r, \theta) = \frac{B(\theta)}{r^{\gamma}} II(r, \theta) \tag{3.1}$$

where $\gamma = log_2 c = 1$, and $B(\theta)$ is some function that does not depend on r. Equation 3.1 shows that shape recipes assume that a particular form holds for the second order statistics of images and range images. If the power spectrum of the image is $1/r^2$, then the cross-correlation must fall off with $1/r^3$ in the Fourier domain. Figure 3.4, an example of using shape recipes in improving stereo is depicted.

3.2. Depth Estimation from Global Image Structure

A method presented by Torralba and Oliva [**154**] emphasizes the drawbacks of estimating depth from monocular information. Figure 3.5 illustrates the ambiguity problem, in the absence of cues for absolute depth measurement, such as binocular disparity, motion, or defocus. The same retinal image is produced by the three cubes and, therefore, the absolute distance between the observer and each cube cannot be measured. Relative depth between parts of the cube can be obtained by interpreting the shading, edges and junctions, but it will not inform about its actual size. However,

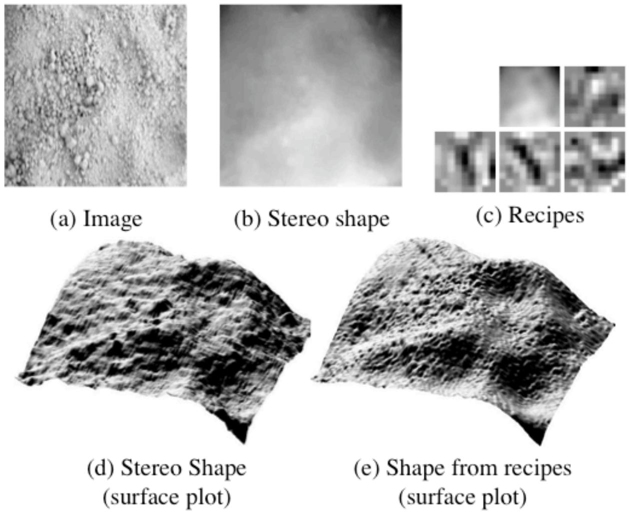

(a) Image
(b) Stereo shape
(c) Recipes
(d) Stereo Shape (surface plot)
(e) Shape from recipes (surface plot)

FIGURE 3.4. Example of real image and the improvement in the shape estimate using the shape recipes. (a) shows one image of the stereo pair and (b) shows the shape obtained by the stereo algorithm. (c) shows the recipes learned at the lowest resolution pyramids. *Taken from* [**60**].

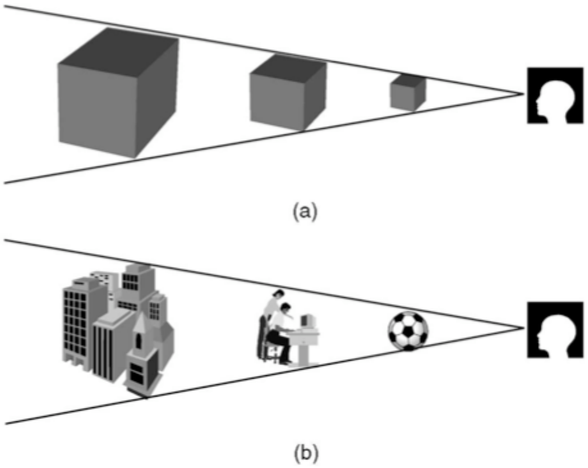


FIGURE 3.5. (a) Artificial stimulus: The monocular information cannot provide an absolute depth percept. (b) Real-world stimulus: The recognition of image structures provides unambiguous monocular information about the absolute depth between the observer and the scene. Adapted from [**154**].

when dealing with real-world stimuli (Fig. 3.5b), there is no ambiguity in depth. The reason for this is that physical processes that shape natural structures are different at each scale. Humans also build different types of structures at different scales, mainly due to functional constraints in relation with human size (e.g., chair, building, city). As a result, different laws with respect to the building blocks, the way they are organized in space and their shape, govern each spatial scale [**75**]. At each spatial scale, image content is therefore what constraints the structure on a 3D scene. As the observed scale directly depends on the depth of view, by *recognizing* the properties of the image structure, the scale of the scene can be inferred, and therefore, the absolute depth [**154**]. The global image structure approach is based on the claim that recognition of the scene as a whole is a simpler problem that the one of general object detection and recognition.

3.3. Studying the Correlations between Intensity and Range Images

One of the few investigations into the joint statistics of range and intensity images was carried on by Howe and Purves [**71**]. In their study, they examine range and coregistered intensity images to find how the length of a line segment on a blank background depends on the orientation of the line. They found that this bias closely matches the 3D length of the line segments when projected into a range image. This result may help to explain how the brain computes depth.

In the analysis of the natural statistics of images it is common to work with the logarithm of the light-intensity [**52, 162**]. One advantage of this is that image contrast, rather than being a multiplicative factor, becomes an additive factor under log intensity. This means that linear filters respond to contrast rather than raw differences in amplitude, and therefore zero-sum linear filters are insensitive to the total contrast of each patch. For the case of range data, as described by Huang *et al.* [**72**], a large object and a small object of the same shape appear identical to the eye when the large object is positioned appropriately far away and the small object is close. However, the raw range measurements of the large, distant object will differ from those of the small object by a constant multiplicative factor. In the *log* range data, however, the two objects will differ by an additive constant. Therefore, a zero-sum linear filter will respond identically to the two objects.

In a more recent study, Potetz and Lee [**117**] measure the correlations between linear properties of range images (e.g., curvature) and linear properties of image intensity data, to explore the structure of the correlations that could usefully underlie 3D judgments from intensity data in images of natural scenes (e.g., shape from shading). By using linear regression, ridge regression, and canonical correlation techniques, they extract simple but interesting statistical trends between both image domains. In their study, they use a sophisticated long-range scanner (a Riegl LMS-Z360) that collects coregistered range and color data by using an integrated color photosensor and a time-of-flight laser scanner with a rotating mirror. The content of their images include scans of trees and wooded areas, rocky areas, building exteriors, and sculptures. The logarithm of the light-intensity values were used rather than intensity itself. Range images were also transformed by applying a logarithmic transform, as was done in previous studies of pure range data.

In their study, images patches of size 25×25 pixels of coregistered light intensity and range data were used. A total of $15,577,472$ patches of rural and urban images. There were in total 15.6 million observations of 1250 random variables: 625 luminance variables and 625 range variables. Correlational relationships between intensity and range data were investigated. The covariance between each pair of pixels in the set of image patches. Two important observations were concluded from the results: i) neighboring range pixels are much more highly correlated than neighboring luminance pixels and, ii) the luminance and range values are negatively correlated. This is shown in Figure 3.6, where the correlation between a specific pixel and other pixels in the patch are plotted. The correlation is obtained using the following equation

$$\rho = \mathrm{cor}[X,Y] = \frac{\mathrm{cov}[X,Y]}{\sqrt{\mathrm{var}[X]\mathrm{var}[Y]}}.$$

The first observation suggests that the low-frequency components of range data contain much more power than in luminance images and that the spatial Fourier spectra for range images drops off more quickly than for luminance images, which are known to have roughly $1/f$ spatial Fourier amplitude spectra [**128**]. Because the

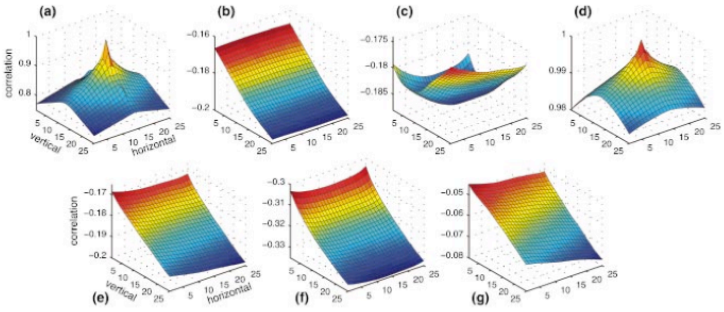


FIGURE 3.6. (a) Correlation between intensity at pixel $(13, 13)$ and all the pixels of the intensity patch. (b) Correlation between intensity at pixel $(13, 13)$ and the pixels of the range patch. (c) Correlation between range at pixel $(13, 13)$ and the pixels of the intensity patch. (d) Correlation between range at pixel $(13, 13)$ and the pixels of the range patch. (e) Correlation between intensity and range at pixel (i, i). (f) Same figure as (e), except measured over rural images only. (g) Same figure as (e), except measured over urban images only. *From* [**117**].

physical measurement of each point or pixel is independent of the others, any regular distortion of the power spectrum of range images caused by the scanner must be caused by the divergence of the laser beam measuring the distance. This finding is reasonable because factors that cause high-frequency variation in range images, such as texture or occlusion contours, tend also to cause variation in the luminance image. However, much of the high-frequency variation found in luminance images, such as shadow and surface markings, are not observed in range images. The second observation suggests that brighter pixels in natural images tend to be closer to the observer. This work is the first evidence that this relationship actually holds in nature. Leonardo da Vinci had this intuition that humans perceive brighter objects as closer [**95**], and in general, artists have made use of this fact to help create compelling illusions of depth. Psychologists subsequently confirmed this fact in controlled experiments [**151, 46, 30**]. In psychology literature, this effect is known as relative brightness [**100**].

In general, the authors conclude that the relationship between the shape of objects and their images depends on the statistical trends that cannot be inferred from physical models of lights. They found that this relationship is intimately related to the statistics of lighting directions, the statistics of camera or head orientation, and the statistics of surface shapes in natural scenes. Some of these statistical trends have already been suspected by psychologists and artists in the past.

There may be many factors that affect the statistics of natural images and that cannot be inferred from simple physical models. For example, the statistical relationship between images and their surfaces will be affected by the natural statistics of illumination direction, a factor that is known to heavily influence human performance on shape-from-shading problems [**122**]. Other factors that influence this relationship may include the statistics of object size and shape in natural scenes as well as the natural statistics of the surface properties of those objects. This opens up the possibility that there may be simple, exploitable statistical relationships between real images and surface shapes that have been overlooked. Discovering these relationships might further the development of vision algorithms that utilize shape-from-shading information. It may also provide insight into how the human visual system is so adept at solving these problems [**146**].

Reinhard *et al.* [**124**] applied second order image statistics in computer graphics implications. They use the power spectrum and demonstrated its theoretical importance with reconstruction filters. They also mention their will to extend the concept of image statistics to 3D geometry, that they would call *3D geometry statistics*.

The concepts and ideas presented so far give the basis for our depth inference approach for man-made indoor scenes based on the inherent statistics observed on the intensity and partial range images, which is described in detail in Chapter 5.

4. Summary

We presented an overview of statistical modeling that relates to the topic of this dissertation. Specifically, we looked at the literature related to the statistics methods for depth inference. We noted, however, that there are few prior work on the study of statistical relationships or correlations between intensity and range data. It is now clear that when missing one part of the puzzle, i.e., intensity information when having only range or viceversa, the completion of missing information is not that easy. However when considering both, at least as partial amounts, some fruitful results may be obtained.

Evidence in prior work have shown that the correlational structure between images and range images may come from several sources. Lambertian shading represents a significant portion of these correlations, but not all. Correlations caused by shadows also contribute to this structure, which complicates potential models of the cross correlation. Also, scenes that are dominated by a few objects may exhibit correlational structures particular to their specific arrangement: a bright object in the foreground, for example. These complications make it difficult to construct a simple model of the image/range image relationship that can be exploited by vision algorithms. However, the assumption that the cross-correlation function varies smoothly has proven to be robust and effective for inference applications.

CHAPTER 4

Probabilistic Models for Images and Markov Random Fields

Recent emphasis on explicit probability models for images may be due to the growing appreciation for the variability exhibited by the images and the realization that exact mathematical models may not be feasible. The key to a statistical approach is to have probability models that capture the essential variability and yet are tractable. In this chapter, we list existing probabilistic models for images and focus our attention to Markov Random Field Models, which are used in this dissertation to solve the depth inference problem. We present a brief review of the history of MRF models, give the basic notions, fundamentals and background on the MRF theory. Also, existing techniques for learning MRFs and the approaches for computing the maximum *a posteriori* (MAP) are presented. Specifically, we describe the Belief Propagation (BP) algorithm. In the last part of the chapter, relevant applications related to our problem are presented. These are texture synthesis and image inpainting.

1. Introduction

Classical statistics provide tools for model building and density estimation but their application to image modeling is difficult due to the large dimensionality [1] of the

[1] For example, an image of 128×128 pixels implies estimation of a density on the space $\mathbb{R}^{16K}$.

observation space. Thus, in addition to the model building, a dimension reduction needs to be solved. A popular idea in the vision literature has been to first reduce dimensions using purely numerical considerations and then impose probability models on the reduced data. By not involving any physical consideration on the imaged objects, or any contextual knowledge, the images are treated as elements of a vector space and one seeks a low-dimensional vector subspace (or its basis) that best represents those numbers (under some chosen criterion). Principal component [**78**], independent components [**28, 9**], sparse coding [**107**], Fishers discriminant [**8**], Fourier transforms, wavelet transforms [**96**], and many other representations are all instances of this idea. The main advantage of such linear projections is that they are computationally cheap. However, a lack of physical or contextual information leads to a limited performance, particularly in recognition of objects from their cluttered images. Furthermore, it seems likely that the space generated by real images is a curved manifold and cannot be simply *approximated* globally by a low-dimensional vector space.

Earliest, and still widely used, probabilistic models for images were based on Markov Random Field (MRF) models [**168**]. MRFs, in particular, define a class of statistical models which enable to describe both the local and global properties of structures in images.

2. Marvov Random Fields

Markov random fields are a type of stochastic processes. Originally motivated from statistical physics, such processes are clearly a natural generalization of the well known concept of Markov chain, in which a time index in 1D is replaced by a space index in 2D.

A Markov chain can be seen as a chain graph of stochastic variables, where each variable depends only on its two neighbors, and it is independent of all the other nodes of the graph. Similarly, a Markov random field is a graph that can be of any structure (as oppose to only a chain graph) to define the relationship between variables, so that each stochastic variable (node) only depends on its immediate neighbors. Figure 4.1 shows at the top, the structure of a Markov chain as a graph and at the bottom, some examples of Markov random field structures as graphs. One can imagine Markov random fields having the same (or even more) wide variety of applications that Markov chains have. The foundations of the theory of Markov random fields may be found in the early 1970's, by Preston [**120**] and Spitzer [**142**] . Although, an example of an early attempt of developing a satisfactory methodology for the analysis of spatial data can be found in [**145**].

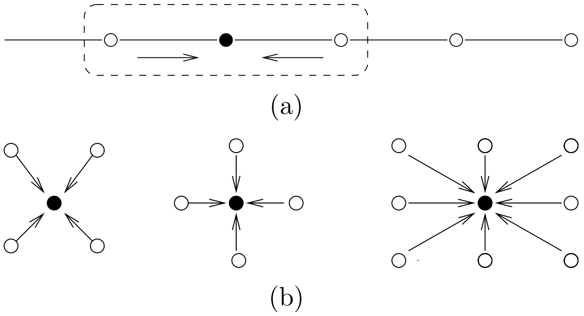


FIGURE 4.1. (a) The structure of a Markov chain as a graph. (b) Examples of graphs on which a Markov random field could be defined.

MRF theory provides an efficient and powerful framework for modeling spatial dependence by combining and organizing spatial information according to feature interactions. These interactions are nonlinear and the features can be of the same nature or of a different one.

MRF models have been successfully used to solve many fundamental problems of image analysis and computer vision [**87**]. Most of these models are for low level processing. These include image restoration, edge detection, image segmentation, texture synthesis and analysis, image inpainting, surface reconstruction, stereovision, motion analysis, and recently for high level vision, such as scene interpretation and object matching and recognition. In this context, the fundamental idea of MRF models relies on the fact that an image pixel cannot be based only on its own value, but it must be influenced by the values and properties of its neighboring pixels.

Moreover, it is very improbable that a pixel has a value different than all of its neighbors, in fact we can affirm that certain configurations are indeed impossible.

MRFs are chosen to solve this type of problems since they can represent efficiently the *a priori* information about an image, specifically, its *spatial context* such that Bayesian decision theory can then be applied. A particular MRF model favors its own class of patterns (e.g., textures or object features) by associating them with larger probabilities than other pattern classes. Then objective functions are formulated in terms of established optimality principles. One of the most popular statistical criteria for optimality, that has been widely used for MRF vision modeling, is the maximum *a posteriori* (MAP) probability. In this research, we adopt the MAP-MRF framework. This framework, advocated by Geman and Geman [**63**] and others, enables us to develop algorithms for a variety of vision problems systematically using rational principles rather than relying on *ad hoc* heuristics. Section 4 gives a complete description of this framework.

An unfortunate by product of this flexibility is that these models are usually analytically intractable. Having constrained the problem and defined the "best" solution (according to some performance measure), there is still the problem of computing it, not to mention the issue of estimating model parameters. As we have seen, these estimates are usually defined in terms of the conditional process given the data, which is another Markov field whose joint distribution is too complex for direct sampling or direct computation of global quantities such as means and modes. A partial answer is provided by the equivalence between MRFs and Gibbs distributions established by Hammersley and Clifford [**66**] and further developed by Besag [**16**]. The MRF-Gibbs equivalence gives an explicit formula for the joint distributions for MRFs. This enables us to model vision problems by a *mathematically* sound yet tractable means for the image analysis in the Bayesian framework [**63**].

3. MRF Theory

In this section, we give the basic notions, fundamentals and background on the MRF theory in the context of computer vision.

3.1. Random field preliminaries

In the following definitions, we use the same notation as in Geman [**62**].

Definition 1. *Let $\mathcal{S} = \{s_1, s_2, ..., s_N\}$ be a set of N sites of a lattice, with a variable X_s at each site $s \in \mathcal{S}$, and let Λ be a finite set called the* state space*. A* **random field** *on $\mathcal{S}$ is a collection of random variables X_s with values in Λ, where the complete set of variables for the whole lattice is denoted by $\mathbf{X} = \{X_s, s \in \mathcal{S}\}$.*

A particular configuration for the lattice is given as $\{X_1 = x_1, X_2 = x_2, ..., X_N = x_N\}$, which is abbreviated to $\{\mathbf{X} = x\}$, where $\mathbf{x} = (x_1, x_2, ..., x_N)$ or $\mathbf{x} = \{x_s\}$ for convenience. The *configuration space* for the variable x is denoted by Ω, whereby,

$$\Omega = \prod_{s \in \mathcal{S}} \Lambda_s, \ \Lambda_s \subset \mathbb{R} \tag{4.1}$$

A dependence structure can be imposed on a random field in two ways. One is by defining a *joint* distribution on X, treating a configuration $\mathbf{x}$ as a realization from that field, and directly model the correlations using the variance-covariance matrix [**16**]. The second way, is to build Markovian dependence structures on the set S. Our interest is on the latter, for which, a neighborhood system (also called a *topology*) on S is introduced by defining a symmetric relation $\sim$ on S and defining neighborhoods as the relational sets:

$$N = \{(i, j) \in \mathcal{S} \times \mathcal{S} : i \sim j\}. \tag{4.2}$$

This generates the graph $\mathcal{G} = (\mathcal{S}, N)$ as a topology on $\mathcal{S}$, with neighborhoods, say $\delta s = N_s$ associated with each element s as:

(i) $s \notin N_s$

(ii) $s \in N_t \iff t \in N_s$

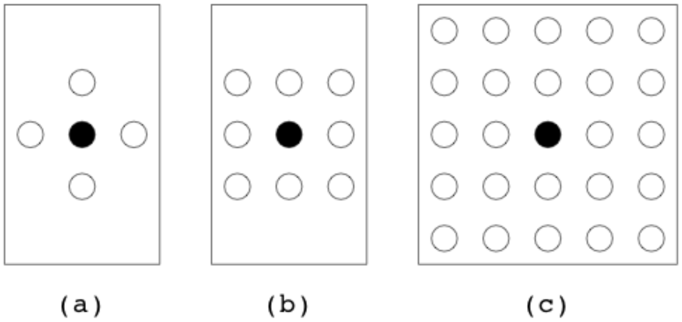


FIGURE 4.2. Neighborhoods: (a) The first order neighborhood $o = 1$ or *nearest-neighbor* neighborhod for the site $s =' \bullet'$ and $r =' \circ' \in \mathcal{N}_s$; (b) second order neighborhood $o = 2$; (c) eight order neighborhood $o = 8$.

This implies that the neighborhoods must be symmetrical and self similar for homogeneous MRFs. The symmetrical neighborhoods systems employed in this thesis are the same as used in [**63**], for which the neighborhood system $\mathcal{N}^o = \{\mathcal{N}^o_s, s \in \mathcal{S}\}$ is defined as

$$\mathcal{N}^o_s = \{r \in \mathcal{S} : 0 < |s - r|^2 \leq o, \tag{4.3}$$

where $|s-r|$ is the Euclidean distance between two points $s, r \in \mathcal{S}$. The neighborhood $\mathcal{N}^o_s$ is defined by the neighborhood order o (but this does not refer to the statistical order of the neighborhood). A first order neighborhood system $o = 1$ (also called the *nearest neighbor* neighborhood system) is shown in Figure 4.2(a), which consists of the four nearest adjacent pixels. The second and eight order neighborhood systems for $o = 2$ and $o = 8$ are shown in Figures 4.2(b) and (c), respectively.

An MRF [**87**] is a set of N random variables indexed over the vertices, or sites, in an ordered lattice. The typical example is a 2D image, where the random variables are the labels (e.g. color) associated with the pixels. The MRF variables are not independent, but are mutually coupled; the key property of MRFs is that the distribution of the random variable associated with a site s, given the values associated with the sites in a (typically small) *neighborhood* of s, is independent of the rest of the sites in the MRF. This is formalized in the following definition.

Definition 2. *An MRF with respect to a neighborhood system is a* **discrete MRF** *if its probability density function satisfies the following conditions:*

$$P(\mathbf{x}) > 0 \;\; \forall \mathbf{x}, \qquad \text{(positivity).} \tag{4.4}$$

$$P(\mathbf{x}_i \mid \mathbf{x}_{\mathcal{S}-\{s\}}) = P(\mathbf{x}_s \mid \mathbf{x}_r, r \in \mathcal{N}_s), \qquad \text{(Markovianity).} \tag{4.5}$$

where $\mathbf{x_s}$*, denotes the random variable of site* s *and* $\mathcal{N}_s$ *is the set of random variables associated with the sites that are in the neighborhood of site* s*.*

When the positivity condition is satisfied, the joint probability $P(\mathbf{x})$ of any random field is uniquely determined by its local conditional probabilities [**16**]. An MRF can be homogeneus if the Markovianity property of Equation 4.5 holds regardless of the relative position of site s in $\mathcal{S}$. The Markovianity depicts the local characteristics of the random variables $\mathbf{X}$.

3.2. Neighborhood System and Cliques

Given a neighbourhood system $\mathcal{N}$, a clique is a set $C \subseteq S$ if every pair of distinct sites in C are neighbours. That is, given $s, r \in C$, $s \neq r$ implies $s \in \mathcal{N}_r$. The single site subset is also a clique. Let $\mathcal{C}$ denote the set of cliques defined on S with respect to $\mathcal{N}$, and let $\mathcal{C}_s$ denote the *local clique set* for a neighbourhood $\mathcal{N}_s$ such that $\mathcal{C}_s = \{C \in \mathcal{C}, s \in C\}$. Cliques are important when considering the equivalence between MRFs and the Gibbs distribution (see [**16**] for a proof of this equivalence).

Figures 4.3(a), (b), and (c) show the neighbourhood configurations for o = 1, 2 and 8 respectively. If we represent the lattice S on a rectangular grid $Z_m = \{(i,j) : 1 \leq i,j \leq m\}$ where $S = Z_m$, $N = m^2$, then the first-order or nearest neighbor system $N^1_{i,j} = \{(i,j-1),(i,j+1),(i-1,j),(i+1,j)\}$. The cliques associated with this neighborhood system are then those subsets of S whereby $\{(i,j)\}$, $\{(i,j),(i,j+1)\}$ and $\{(i,j),(i+1,j)\} \subset Z_m$, as shown in Figure 4.3(e). The cliques contained in the local clique set $\mathcal{C}_s$ of $N^1_{i,j}$ are then those cliques $\{(i,j)\}$, $\{(i,j),(i,j+1)\}$, $\{(i,j),(i+1,j)\}$, $\{(i,j),(i,j-1)\}$ and $\{(i,j),(i-1,j)\}$, as shown in Figure 4.3(d). For the second-order neighbourhood $\mathcal{N}^2_s$, the set of cliques $\mathcal{C}$ are those of type shown

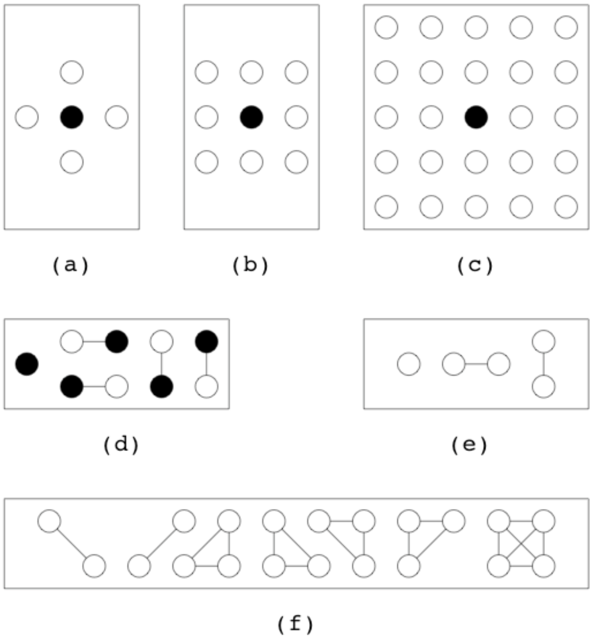


FIGURE 4.3. Neighborhoods and cliques. (a) The first order neighborhood $o = 1$ or *nearest-neighbor* neighborhod; (b) second order neighborhood $o = 2$; (c) eight order neighborhood $o = 8$. (d) Local clique set for nearest-neighborhood; (e) clique types for nearest-neighbor neighborhood; (f) additional clique types for second-order neighborhood.

in Figures 4.3(e) and (f). The number of clique types grows almost exponentially with increasing order o.

3.3. Gibbs Random Fields

The distribution over the MRF variables, can be written as a Gibbs distribution,

$$P(\mathbf{x}) = \frac{\exp(-U(\mathbf{x}))}{Z} \tag{4.6}$$

where $\mathbf{x}$ is a $NK-$dimensional vector formed by concatenating the vectors $\mathbf{x}_s(s = 1, ..., N)$, U is an *energy function* and Z is a normalizer given by,

$$Z = \sum_{\mathbf{x}} \exp(-U(\mathbf{x})). \tag{4.7}$$

where the sum runs over all possible values of $\mathbf{x}$. Note that, computing Z, which is known as the *partition function*, is generally tractable only for very small MRFs, since the number of terms in the sum of Equation 4.7, increases exponentially with the size of the MRF. This is due to the mutual coupling between the MRF variables. Same problem emerges if we want o compute the marginal posterior distribution over any of the individual MRF variables e.g., for the purpose of parameter fitting since this requires summing over all remaining variables. The energy function U defines the properties of the MRF model and can generally be written

$$U(\mathbf{x}, \mathbf{y}, \mathbf{\Theta}, \beta) = U^{ext}(\mathbf{x}, y, \Theta) + U^{int}(\mathbf{x}, \beta). \tag{4.8}$$

where U^{ext} denotes the energy (or potential) arising from external influence; in the context of probabilistic image modeling, this typically comes from observed data $\mathbf{y}$ via a model determined by parameters $\mathbf{\Theta}$, and corresponds to a log-likelihood term. U^{int} denotes the internal energy which, as suggested by the notation, only depends on the MRF variables $\mathbf{x}$ and parameter β, and corresponds to a prior distribution over $\mathbf{x}$.

3.4. Markov-Gibss Equivalence

An MRF is characterized by its local property (the Markovianity), whereas a GRF is characterized by its global property (the Gibbs distribution). The Hammersley-Cliffor theorem [**66**] establishes the equivalence of these two types of properties. The following

Theorem 1. *X is an MRF on $\mathcal{S}$ with respect to $\mathcal{N}$ if and only if X is a GRF on $\mathcal{S}$ with respect to N.*

Many proofs of the theorem exist (e.g. Besag [**16**], Kindermann and Snell [**77**]). The practical value of the theorem is that it provides a simple way of specifying the joint probability $P(X = x)$ by defining the clique potential functions $V_c(x)$ and choosing appropriate potential functions for desired system behavior. In this way, the *a priori* knowledge or preference about interactions between labels are encoded. However, a major topic in MRF modeling is how to choose the forms and parameters of the potential functions for a proper encoding of constraints.

To calculate the joint probability of an MRF, which is a Gibbs distribution, the partition function (Equation 4.7) needs to be evaluated. The reason for this is that the sum over a combinatorial number of configuration in Λ is computational intractable. The explicit evaluation can be avoided in maximum-probability based MRF vision models when $U(x)$ contains no unknown parameters. However, this is not true when the parameter estimation is also a part of the problem. In the latter case, the energy function $U(x) = U(x \mid \theta)$ is also a function of parameters θ and so is the partition function $Z = Z(\theta)$. Thus, one needs to evaluate $Z(\theta)$.

4. MAP-MRF Framework

To address the computational difficulties associated with the computation of the joint probability in MRF models, a number of approximate methods have been proposed [**87**].

In the MAP-MRF framework, the posterior distribution of an MRF is defined by $P(\mathbf{x}|\mathbf{d})$, where $\mathbf{d}$ is the observation. Its form and parameters are determined, in turn, according to the Bayes formula, by those of the joint prior distribution of the labels and the conditional probability of the observed data. It is this probability function specified by the functional form and the parameters that defines the particular type of a MRF model. Two major parts of the MAP-MRF modeling is to derive the form of the posterior distribution and to determine the parameters in it, so as to completely define the posterior probability. Another important part is to design optimization algorithms for finding the maximum of the posterior distribution.

4.1. Pairwise MRF Model

A pairwise MRF model (shown in Figure 4.4, also known as *Markov network*, is defined as a set of hidden nodes x_i (white circles in the graph) representing local patches in the output image C, and the observable nodes y_i (shaded circles in the graph) representing local patches in the input image B. Each local patch is centered to pixel location i of the respective images.

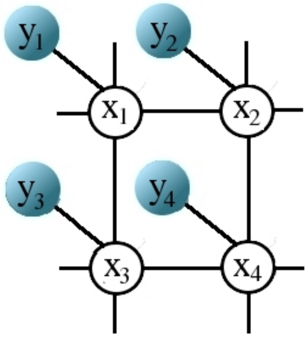


FIGURE 4.4. The pairwise Markov Random Field used to model the joint probability distribution of the system. Observation nodes, y, represent an image patch input image, and hidden nodes x, an image patch in the output image to be inferred.

Denoting the pairwise potentials between variables x_i and x_j by ψ_{ij} and the local evidence potentials associated with variables x_i and y_i by ϕ_i (see Figure 4.5), the joint probability of the MRF model under variable instantiation $\mathrm{x} = (x_1, ..., x_N)$ and $\mathrm{y} = (y_1, ..., y_N)$, can be written [**16, 63**] as:

$$P(\mathrm{x}, \mathrm{y}) = \frac{1}{Z} \prod_{ij} \psi_{ij}(x_i, x_j) \prod_{i} \phi_i(x_i, y_i), \tag{4.9}$$

where Z is the normalization constant. We wish to maximize $P(\mathrm{x}, \mathrm{y})$, that is, we want to find the most likely state for all hidden nodes x_i, given all the evidence nodes y_i. The compatibility functions allows to set high (or low) compatibilities to neighboring pixels according to the particular application. These potentials are used in messages that are propagated between the pixels to indicate what combination of values each image pixel should have.

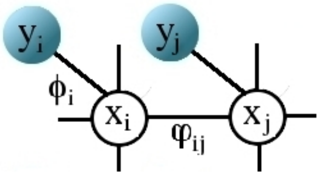


FIGURE 4.5. The potential functions ϕ and ψ define the compatibilities between nodes in the Markov network.

A pixel value in C is synthesized by estimating the maximum *a posteriori* (MAP) solution of the MRF model using the training set. The MAP solution of the MRF model is:

$$\mathbf{x}_{MAP} = \arg\max_{\mathbf{x}} P(\mathbf{x} \mid \mathbf{y}), \tag{4.10}$$

where

$$P(\mathbf{x} \mid \mathbf{y}) \propto P(\mathbf{y} \mid \mathbf{x}) P(\mathbf{x}) \propto \prod_i \phi_i(x_i, y_i) \prod_{(i,j)} \psi_{ij}(x_i, x_j) \tag{4.11}$$

Calculating the conditional probabilities in an explicit form to infer the exact MAP in MRF models is intractable. We cannot efficiently represent or determine all the possible combinations between pixels with its associated neighborhoods. Various techniques exist for approximating the MAP estimate, such as Markov Chain Monte Carlo (MCMC), iterated conditional modes (ICM), maximizer of posterior marginals (MPM), etc. See [**41**] for a comparison. In this research, we compute a MAP estimate, by using a learning-based framework on pairwise MRFs, as proposed by [**59**], using belief propagation (BP).

4.2. MRF-MAP inference using Belief Propagation

Belief propagation (BP) is a machine-learning method for sharing probabilistic information from multiple connected sources. Seminal work by Freeman *et al.* [**59**] demonstrates the usefulness of Bayesian belief propagation to quickly find approximate solutions to various two-dimensional vision problems modeled by Markov networks with loops. Using a common machinery, but with a representation and training sets appropriate to each problem, they showed exceptional results for super-resolution, image enhancement and motion estimation, which have inspired the use of belief propagation in the computer vision community. In later work, Weiss and Freeman [**166**] gives a theoretical understanding of the previously good performance of the BP algorithm in networks with loops, by showing the formal connections between BP and some well-understood approximations in statistical physics [2]. They show how the posterior probabilities calculated by belief propagation related to the true marginal

[2] Like those due to Bethe and Kikuchi [**109**]

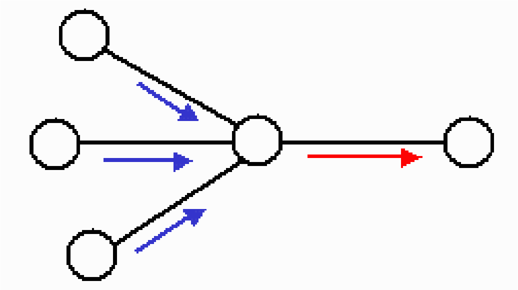


FIGURE 4.6. Computing messages in the belief propagation algorithm.

probabilities. This has led to the development of better inference algorithms (e.g., the Generalized Belief Propagation (GBP) [**172**]), that give a new possibility for solving vision problems that were previously computationally intractable.

The BP algorithm solves graphical inference problems via a series of local message-passing operations. For example (see Figure 4.6), to compute the outgoing (red) message at left, the central node must combine all incoming messages (blue) with its local observation. In tree structured graphs, BP is exact, and messages can be interpreted as sufficient statistics. In graphs with cycles, BP is approximate, but has reasonable theoretical justifications and produces excellent empirical results in many applications. BP may be applied to graphs with discrete (messages are vectors) or Gaussian (messages are means and covariances) variables.

For MRFs, BP is an inference method to efficiently estimate Bayesian beliefs in the network by the way of iteratively passing messages between neighboring nodes. The Markov assumption follows a "message-passing" rule that involves only local computations, resulting in a maximum *a posteriori* estimate [**63, 111, 59**]. Formally, the message send from node i to any of its adjacent nodes $j \in \mathrm{N}(i)$ is

$$m_{ij}(x_j) = Z \sum_{x_i} \psi(x_i, x_j)\phi(x_i, y_i) \prod_{k \in \mathrm{N}(i)\backslash\{j\}} m_{ki}(x_i) \tag{4.12}$$

where Z is the normalization constant (also known as the partition function). The maximum *a posteriori* scene patch for node i is:

$$x_{iMAP} = \arg \max_{\mathbf{x}_i} \phi(x_i, y_i) \prod_{j \in \mathrm{N}(i)} m_{ji}(x_i). \tag{4.13}$$

The BP algorithm is not guaranteed to converge, but if it does so, then it converges to a local stationary point of the Bethe approximation to the free energy [**171**]. In our experiments, the BP algorithm usually converges in less than 10 iterations. And it is also notable that BP is faster than many traditional inference methods.

5. Computer Vision Applications using MRFs

Many computer vision applications have been extensively use MRFs. In his book, Li [**87**] gives several examples in the context of computer vision where MRFs are used. Here we describe three applications that are relevant to our research.

5.1. Texture Synthesis

The field of texture synthesis is concerned with synthesizing, from an input texture sample, an arbitrary amount of *perceptually similar* output texture in 2D image space or on surfaces of 3D models. The term *perceptually similar* in this context means that the user should recognize the result as the *same* texture, yet it must also contain sufficient variation of the input so it is not perceived as *identical.*

Textures can be synthesized by different techniques such as fractals, random fields, reaction-diffusion, morphology, Gabor filters, Eigen-patterns, steerable pyramids, wavelets, tiling and co-occurrence methods. Each method can generate only a particular subset of texture patterns and have their own advantages and disadvantages. Texture synthesis techniques can be broadly categorized into local region-growing methods and global optimization-based methods. Local methods grow the texture one pixel or patch at a time with the goal of maintaining coherence of the grown region with nearby pixels [**45, 164, 44**]. In such approaches, small errors can accumulate over large distances leading to inconsistencies in the synthesized texture. On the other hands, global methods evolve the entire texture as a whole, based on some criteria for evaluating similarity with the input. Most existing global approaches either model only pixel-to-pixel interactions that may be insufficient to capture large scale structures of the texture [**108**], or lead to complex formulations that are difficult to optimize [**116, 60**].

Markov Random Fields for image completion has also proven successful in texture synthesis [**108, 45, 164**]. In this context, MRF methods model a texture based on its local and stationary properties. A new texture is generated pixel by pixel in such a way that these two properties are preserved in a small set of spatially neighboring pixels which characterizes every pixel on the texture image. The MRF property of textures requires that *locality* and *stationarity* be satisfied. Locality implies that the color at a pixel's location is dependent only on a neighborhood of pixels around it, while stationarity means that this dependency is independent of the actual location of the pixel.

Global synthesis methods have usually employed matching of statistical properties like histograms and wavelet coefficients between input and output textures [**116**]. There has also been previous work that makes use of optimization over MRFs for synthesis. Paget and Longstaff [**108**] use local annealing over a multi-scale MRF for texture synthesis. They consider only pixel-to-pixel interactions. Recently, Kwatra et al. [**80**] propose an approach that is based on optimization of texture quality with respect to a similarity metric. This similarity metric is motivated by the MRF-based similarity criterion used in most local pixel-based synthesis techniques. The authors merge these locally defined similarity measures into a global metric that can be used to jointly optimize the entire texture. This global metric allows modeling of interactions between large neighborhoods; nevertheless, it can be optimized using a simple iterative algorithm with reasonable computational cost. Figure 4.7 shows comparisons of their technique with other existing techniques.

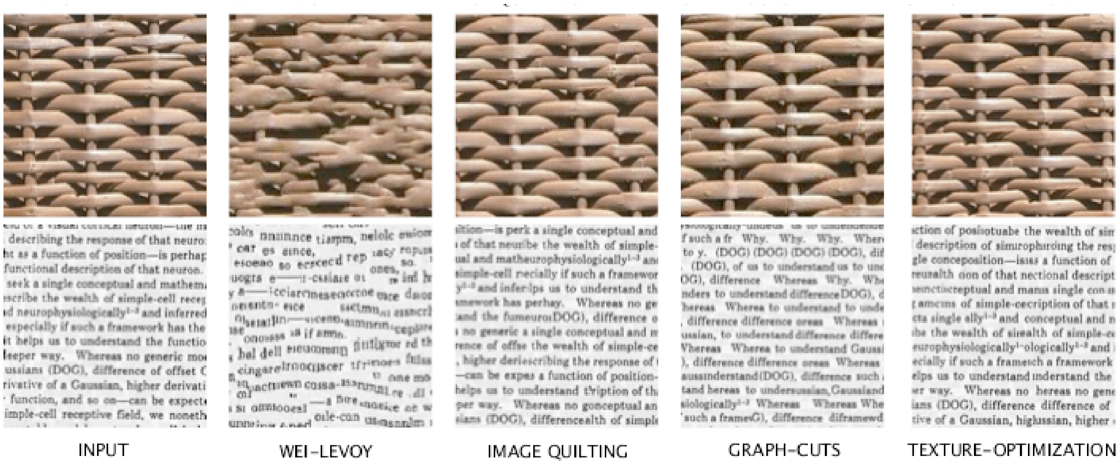


FIGURE 4.7. Comparison of various other texture synthesis from [**80**]

.

5.2. Image Inpainting

Inpainting, also referred as dis-occlusion and retouching (artistic synonym for ”image interpolation”), is a well-known technique in the context of image and art restoration, where paint losses are filled up to the level of the surrounding paint and then colored to match in an undetectable way. Figure 4.8 shows an example of manual photo restoration. Bertalmio *et al.* [**13**], were the pioneers in digital inpainting. Since then, a wide number of applications emerge, including the scratch removal in

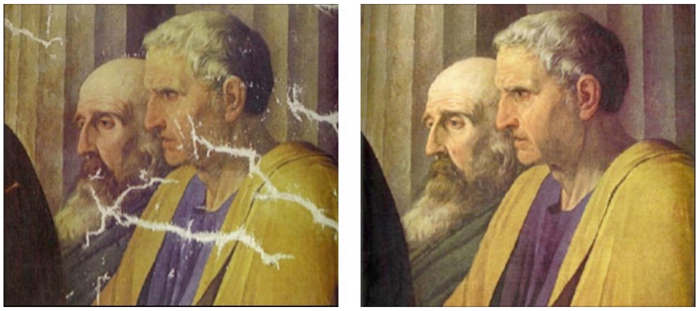
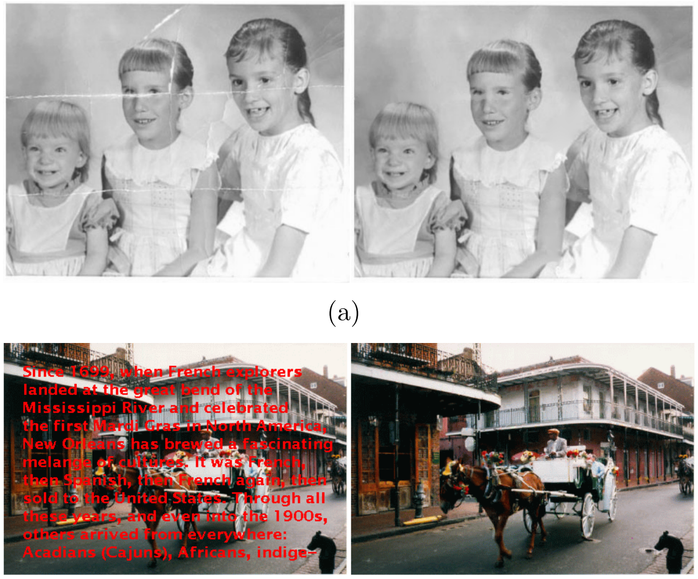


FIGURE 4.8. Example of manual inpainting. Detail of ”Cornelia, Mothe of the Gracchi” by J. Suvee (Louvre). Taken from Emile-Male ”The Restorer’s Handbook of easel painting.”

digital photos and old films, object removal (superimposed text, logos, occluding objects), zooming and super-resolution, etc. Some examples of digital inpainting taken from [**13**], are shown in Figure 4.9.



(a)

Since 1699, when French explorers
landed at the great bend of the
Mississippi River and celebrated
the first Mardi Gras in North America,
New Orleans has brewed a fascinating
melange of cultures. It was French,
then Spanish, then French again, then
sold to the United States. Through all
these years, and even into the 1900s,
others arrived from everywhere:
Acadians (Cajuns), Africans, indige-

(b)

FIGURE 4.9. Examples of digital inpainting from [**13**].

The inpainting problem is clearly ill-posed. Any method must therefore use some prior assumptions about the unknown missing values and their relations with the known values near the hole.

Most existing approaches (see e.g. [**136**] for a review) use a generic prior on images (e.g. high smoothness, low total variation or low curvature) and use an optimization to find the most probable completion given the prior model and the immediate boundary of the hole. Thus a hole is estimated as the most ”smooth” continuation of the local structure of the image, where smoothness can be defined in different ways. Bertalmio *et al.* [**13**], inspired by professionals art restorators, propagate gradient direction and gray values from surrounding neighborhood into the hole. They formulate the process elegantly in a PDE framework, and solve it using fast iterative solvers.

Chan and Shen [**24**] presented another way to define a smooth filling-in. Their method minimized the total variation in the result image. As mentioned in [**6**], such approach handles noise very well, but tends to complete straight lines. Filling-in of holes is also performed by texture synthesis algorithms where it is assumed that the missing data is part of a (usually homogeneous) texture. The region can be filled-in by a texture synthesis engine, e.g., [**19, 45, 116, 44**]. The texture-synthesis approach can process large holes, and fill them with rich structures learned from similar regions in the image. Two applications of texture synthesis to image inpainting are shown in [**69**], there Hirani and Toksuka fill in a selected texture by combining spectral and spatial information, achieving impressive results. Criminisi *et al.* [**31**] used an exampler-based approach, adapting the synthesis method of Efros and Leung [**45**] to image inpainting. Recent works combine texture synthesis with inpainting of structure [**14**].

As indicated in [**85**] , the above approaches can be seen as *local* inpainting algorithms and despite their impressive results they must by definition give identical completions when the immediate boundary of the hole is identical. This issue is depicted in Figure 4.10, the first row shows two images, a square and a circle, each with a missing square region on the bottom-right part. Since the small neighborhoods around the holes are identical [3], local inpainting algorithms give similar results. However, we know that the two images are different. Figures 4.10(c),(d) show the results of the local algorithm in [**13**]. While the completions are very reasonable given the local information, they do not appear *perceptually correct.* This gives evidence that our visual system is taking more global information into account.

5.3. Surface Inpainting

In 3D geometry processing, we can have an analogous inpainting tasks, as digital representations of real-world objects often contain holes, due to problems during data acquisition or as a consequence of interactive modeling operations. This is an area of ongoing research. For example, Verdera *et al.* [**163**] present an algorithm for filling-in surfaces holes based on geometric partial differential equations, derived from

[3] Up to numerical error, the gradients and gray levels in the immediate boundary of the hole, are identical.

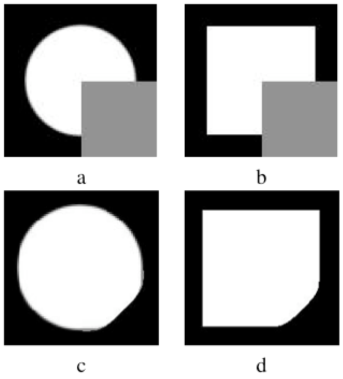


FIGURE 4.10. An example of the local inpainting problem from [**85**]. a-b) Two images with holes. In both cases the boundaries of the holes are identical thus local inpainting algorithms would complete them identically. c-d) The results of the algorithm in [**13**] run with a single resolution. As can be expected from a local algorithm, the completion is identical.

inpainting algorithms, that smoothly continue the surface into the hole. Recently, Bendels et al. [**11**] propose a fragment-based surface inpainting method to fill holes in point set surfaces by extrapolating or restoring the basic and detail geometry. This is motivated by the fact that real life objects often exhibit a high degree of coherence in the sense that for missing parts one can find similar regions on the object. The method analyzes the neighborhood of a hole, and identifies and copies into the hole region appropriate local neighborhood patches represented in local frames (the 3D analogue to what is called a *fragment* in image processing). By finding best matches hierarchically on several scales, the hole is filled in conformance with the context with respect to all considered scales. Figure 4.11 shows an example.

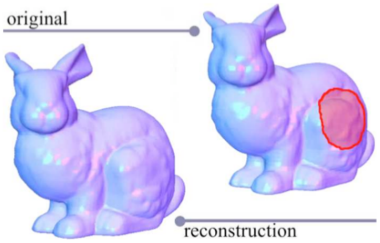


FIGURE 4.11. An example of a reconstruction from [**11**]. The hole (indicated in red on the right) is filled hierarchically, leading to the visually plausible reconstruction (left).

CHAPTER 5

Statistics on Visual and Partial Range Data for Scene Recovery

When building a 3D model of a real environment, suitable sensors to densely cover the environment are required. Visual sensors are the preferred alternative for capturing the photometric characteristics of the scene and they are fast. Conversely, range sensors capture geometric data directly, but acquiring dense range maps is often impractical.

This chapter details our statistical learning method for depth recovery. Specifically, we estimate dense or high resolution range maps of indoor environments using only intensity images and sparse partial depth information. Markov Random Field (MRF) models are proposed as a viable stochastic model for the spatial distribution of intensity and range data. This model is trained using the (local) relationships between the observed range data and the variations in the intensity images and then used to compute unknown depth values. Two techniques for the MAP-MRF estimation are described, the first one based on a non-parametric sampling strategy and the second one by using the belief propagation (BP) algorithm. Their advantages as well as their limitations are highlighted.

1. Introduction

The appeal of Markov random field models for range estimation comes from their explicit ability to model interactions and relationships between neighboring parts of the data space (a background on MRF models was given in Chapter 4).

For the particular problem of estimating dense range maps from intensity images and partial range data, we first need to consider how the relationships between the intensity information and the partial range can give us knowledge about the total underlying geometry of the scene. To this end, we introduce a definition of *augmented voxels* [1] which contain intensity (either from grayscale or color images) and range information (where the range can initially be unknown). We are interested only in the set of such augmented voxels in which one augmented voxel lies on each ray that intersects each intensity pixel of the input intensity image, thus giving us registered range and intensity images (see Figure 5.1).

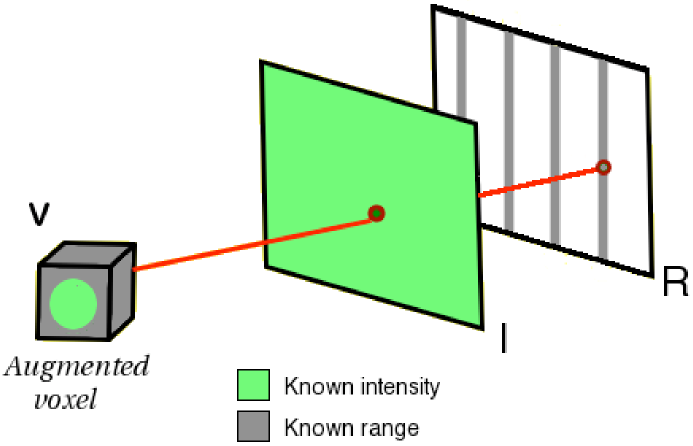
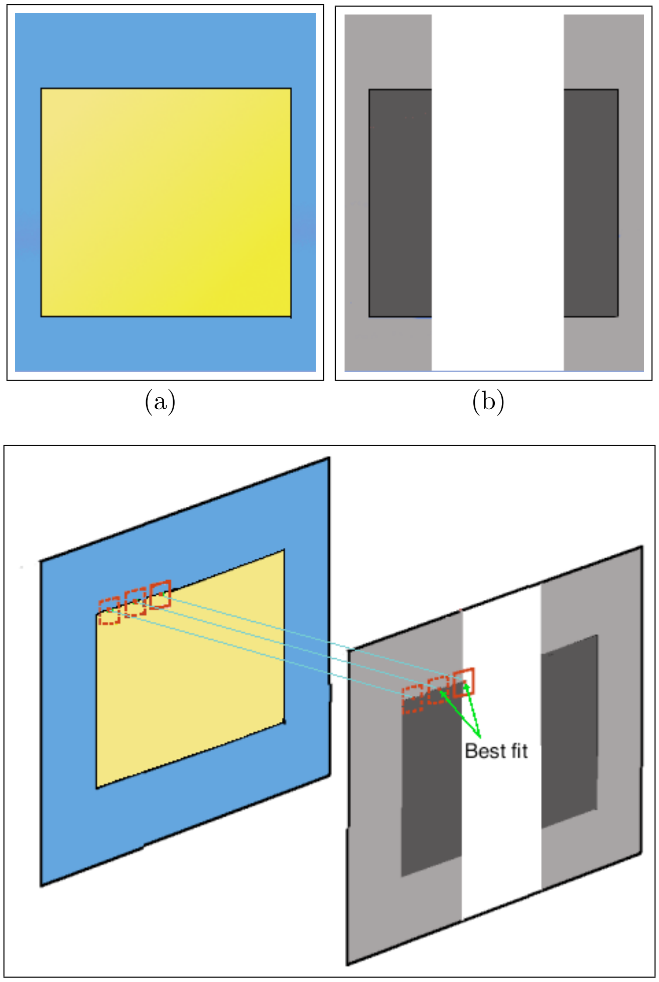


Figure 5.1. Definition of an augmented voxel.

Let us now illustrate how the modeling of the relationships between neighboring augmented voxels can help in the reconstruction process by using a very simple scene composed of only one object and a uniform background (see Figure 5.2.a). The associated (partial) range image is shown in Figure 5.2.b. By considering the known

[1] In this thesis, we define a voxel as a location containing only intensity information, whereas an augmented voxel is a location that may contain both intensity and range information.



(a) (b)

(c)

FIGURE 5.2. A very simple (synthetic) example of the range estimation process. (a) The intensity image showing an object and a background, (b) the associated (incomplete) range map, and (c) how local relationships are helpful for range inference by considering the neighborhoods with the most similar content in range and intensity (squares).

augmented voxels (i.e., intensity pixel locations with already assigned range) coming from part of the object and background, we can make an intelligent guess regarding the content of the rest of the object and background shapes. Markov random fields let us model the relationship between intensity and observed range explicitly, and make it similar to or, when appropriate, different from, relationships among other observed data (see Figure 5.2.c). This feature also makes Markov random fields well suited for modeling spatial data, i.e. data that come from different spatial locations. A key issue in modeling spatial data is determining how best to describe the relationship between several data points taken in close proximity to each other, and how it differs from the case in which data points are taken at a distance. In our example, Markov random fields could model the similarity between the intensity and range data at different positions and select those with a high probability (shown by the little squares in Figure 5.2.c) of being a good fit for the missing region to fill.

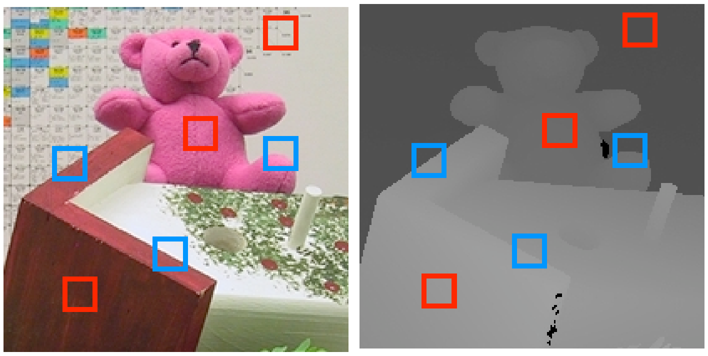
1.1. On the statistics of image structure

The complexity of natural images suggests the development of complex methods in vision science. However, it has been demonstrated (see related work on Chapter 3) that the visual environment is constrained by a large number of statistical regularities in addition to the valuable information available in the images about the scene (e.g., shading, specularities, contour shape, color gradients, texture gradients, binocular disparity, optical flow, etc.). In this thesis, we go beyond image statistics to understand how the statistical relationships between image properties (intensity data) and range (3D data) can be used to make accurate inferences about the scene geometry.

One of the most important constraints for recovering surface properties is that the physical processes underlying image formation are typically smooth: depth and orientation of surfaces are *mostly* continuous and so are reflectance and illumination. The smoothness property is captured well by standard regularization [**15**]. Surfaces and their properties, however, are not always smooth: they are smooth *almost* everywhere, but not at discontinuities. Lines of discontinuity are themselves usually continuous, relatively smooth, nonintersecting curves.

Thus the detection of discontinuities is a critical issue as they usually represent the most important locations in a scene; for instance, depth discontinuities often correspond to boundaries of an object or of a part. In this context, intensity values of pixel locations contained in a neighborhood, provide constraints on surface shape that can be used as statistical relationships between the available range and intensity images in the range estimation process.

The two main constraints provided by intensity values in an image are that: (1) the surface is smooth along the cluster of pixel locations that have intensity values that are smooth, and (2) a cluster containing pixel locations with different intensity values indicates a depth discontinuity. Figure 5.3 shows examples of these two types of constraints. The left image is the intensity image and the right, its corresponding range map. Areas indicated by the red or dark squares in the intensity image show no change in intensity. This contributes knowledge about surface smoothness in the range domain. On the other hand, when variations in intensity exist, as indicated by the areas in the blue or light squares, there is a high probability that there exist depth discontinuities. The change in depth within the objects in a scene is usually gradual, and hence, depth can be said to exhibit a local dependency. However, detecting discontinuities directly using the intensity values of an image is often not sufficient. Fortunately, line (or edge) features, also known as *intensity edges*, which typically correspond to boundaries of homogeneous regions in an image, also provide important geometric information about the 3-D structure of objects in the scene. In the following



(a) Intensity image (b) Range image

FIGURE 5.3. Two examples of knowledge that intensity data provides about surfaces. The areas indicated by the red squares provides knowledge about surface smoothness and the areas in blue squares give knowledge about variations in depth.

section we will give a short taxonomy of the types of geometric phenomena that cause intensity edges.

1.2. Central role of intensity edges

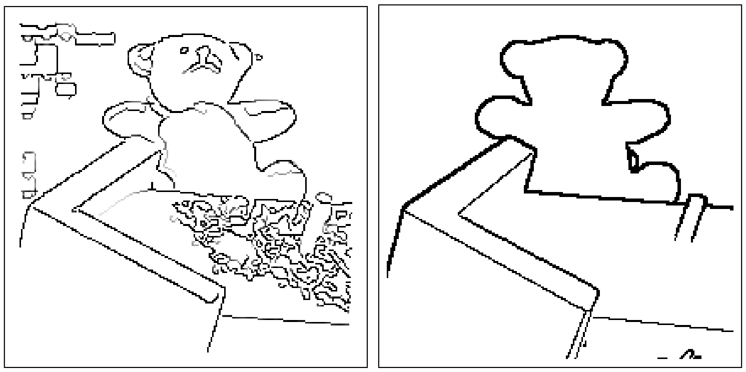
Intensity edges can be used as the primary cue in guiding the search for discontinuities in other physical processes (for example surface depth, surface orientation, texture, shadows, color, specularities) [**113**]. The critical role of intensity edges in artificial -and probably also biological - vision is intuitively clear: changes in surface properties (depth, orientation, material, texture) usually produce large gradients in the image intensity.

Assuming a simple imaging model (e.g., Lambertian), large intensity gradients in the image can be caused by six physical phenomena [**113**]:

- Occluding edges (extremal edges and blades);
- folds;
- shadow edges;
- surface markings and
- specular edges.

Intensity edges are detected quite reliably by the Canny edge detector [**22**]. Figure 5.4(a) shows the edges detected from the intensity image of Figure 5.3. In (b) the edges from the range image are depicted. It can be observed that the edges detected from the range image, which reflect depth discontinuities, are contained in the intensity edges. Because of the constraints of image formation discussed earlier, the *correct* depth discontinuities will, in almost all cases, correspond precisely to the locations of intensity edges. Our range synthesis method exploits this by restricting the range estimation process according to intensity edges, thus assuring the smoothness and continuity of discontinuities. This process will be described in detail later.

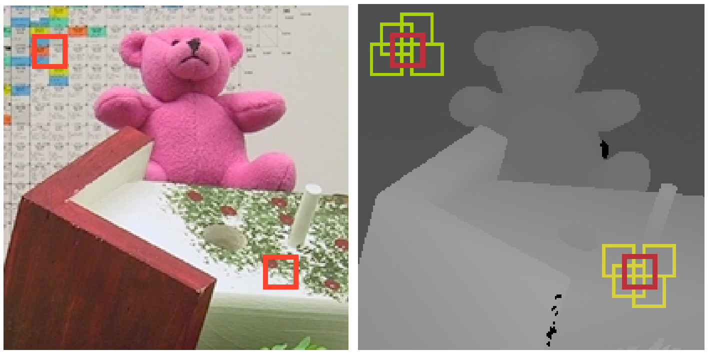
There are some cases in which discontinuities will not occur at intensity edges, for instance, objects that *blend in* with their background. Although this situation occurs rarely in natural scenes, it is usually present due to camera underexposure or



(a) Intensity edges (b) Range edges

Figure 5.4. Edges detected from intensity (a) and range (b) images. Edges help in detecting depth discontinuities, which is a critical step in the reconstruction process.

saturation, where some locations of the objects may blend in with the background. Moreover, there may be *false* edges (i.e., edges that do not represent a discontinuity), such as those coming from texture-like regions or even shadows. However, as will be demonstrated in our experiments, this is not critical for reconstruction since range data inside the neighborhoods that are very close to these texture-like edges represent the same type of *smooth* surface (see Figure 5.5).



(a) Intensity image (b) Range image

Figure 5.5. Figure shows that changes in intensity coming from texture-like regions do not present a problem in the reconstruction process since very close neighborhoods represent the same type of *smooth* surface.

Also, edges coming from shadows play an important role in the perception of 3D surface geometry [**23, 101, 33**]. Our intuition tell us that it is the shapes of the intrinsic shadow boundaries that directly provide information about surface shape and illumination. But the mathematical modeling of realistic shadows or any other shape-related clues is very complex. Work in the literature has to make assumptions regarding the type of surfaces in the scene and lighting conditions, so that simple mathematical models can be defined. When learning relationships between intensity and shape, special attention has to be paid to occlusions and boundaries. Not all intensity values in an image are directly related to shape variations. For example, an image can be decomposed into paint (reflectance) and shading variations (surface normal) [**60**]. Shading variations are directly produced by the shape, however, the paint will be related to changes in the reflectance function of the material and not directly related to the shape.

However, the method we propose, learns the relationships from the intensity and range images directly without having to hypothesize surface smoothness or reflectance properties, that may be inappropriate to a particular environment. Thus, our approach does not intent to implicitly obtain a general algorithm, but to use the local statistical relationships between the intensity and the input range directly from the observed data.

2. The MRF Model for Range Synthesis

We want to solve the following problem: How to infer a dense range map from an intensity image and a limited amount of initial range data. In the rest of this dissertation we refer to this problem as the *range estimation* or *range synthesis* problem. We made the following assumptions: (1) the initial range data and intensity data is already registered [2], and (2) the range data is clumped into at least some sets of mutually-adjacent voxels as opposed to scattered measurements far from one-another (the range data sampling strategies with experimental results are discussed in Chapter 6.).

[2] The registration process depends on the application, in our case, we are interested in the mobile robot environment modeling and the registration process is described in detail in Section 3.4 of Chapter 7.

The solution of the range estimation problem can be defined as the minimum of an energy function. The first idea on which our approach is based, is that an image can be modeled as a sample function of a stochastic process based on the Gibbs distribution, that is, as a Markov Random Field (MRF) [**63**]. We consider range estimation a task of assigning a depth value to each pixel of the input image that best describes its surrounding structure using the already available intensity and range data. The MRF model has the ability to capture the characteristics of this input data and then use them to learn a marginal probability distribution that is to be used to infer data in regions with missing range values. This model uses multi-scale representations of the intensity and range images to construct a probabilistic algorithm that efficiently estimate the missing range. Statistical relationships are learned directly from the input data, without having to make any assumptions regarding lighting conditions of specific nature, location or environment type that would be inappropiate to a particular scene.

As was previously mentioned, we focus on our development of a set of **augmented voxels V** that contain intensity and range information. Thus, $\mathbf{V} = (\mathrm{I}, \mathrm{R})$, where I is the matrix of known pixel intensities, and R denotes the matrix of pixel depths (see Figure 5.1). Let $Z_m = (x, y) : 1 \leq x, y \leq m$ denote the m integer lattice (over which the images are described); then $\mathrm{I} = \{I_{x,y}\}$, $(x, y) \in Z_m$, denotes the gray levels of the input image, and $\mathrm{R} = \{R_{x,y}\}$, $(x, y) \in Z_m$ denotes the depth values.

2.1. The MRF Model

The range estimation problem can be posed as a labeling problem. A labeling is specified in terms of a set of *sites* and a set of *labels*. In our case, sites represent the pixel intensities in the matrix I and the labels represent the depth values in R. Let $\mathcal{S}$ index a discrete set of M sites $\mathcal{S} = \{s_1, s_2, ..., s_M\}$, and $\mathcal{L}$ be the set of corresponding labels $\mathrm{L} = \{l_1, l_2, ..., l_M\}$, where each l_i takes a depth value. The inter-relationship between sites and labels define the *neighborhood system* $\mathcal{N} = \{N_s \mid \forall s \in \mathcal{S}\}$, where N_s is the set of *neighbors* of s (i.e., the neighborhood of s), such that (1) $s \not\in N_s$, and

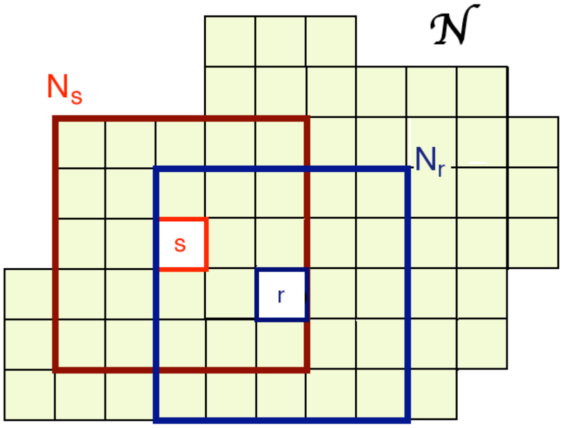


FIGURE 5.6. A neighborhood system definition.

(2) $s \in N_r \iff r \in N_s$ (see Figure 5.6). Each site s_i is associated with a random variable F_i. Formally, let F $= \{F_1, ..., F_M\}$ be a random field defined on $\mathcal{S}$, in which a random variable F_i takes a value f_i in $\mathcal{L}$. A realization $f = f_1, ..., f_M$, is called a *configuration* of F, corresponding to a realization of the field. The random variables F defined on $\mathcal{S}$ are related to one another via the neighborhood system $\mathcal{N}$.

F is said to be an MRF on $\mathcal{S}$ with respect to $\mathcal{N}$ if and only if the following two conditions are satisfied [**66**]:

$$P(f) > 0 \text{ (positivity), and}$$
$$P(f_i \mid f_{\mathcal{S}-\{i\}}) = P(f_i \mid f_{N_i}) \quad \text{(Markovianity).}$$

where $\mathcal{S} - \{i\}$ is the set difference, $f_{\mathcal{S}-\{i\}}$ denotes the set of labels at the sites in $\mathcal{S} - \{i\}$ and $f_{Ni} = \{f'_i \mid i' \in N_i\}$ stands for the set of labels at the sites neighboring i. The Markovianity condition describes the local characteristics of F. The depth value (label) at a site is dependent only on the augmented voxels (containing intensity and/or range) at the neighboring sites. In other words, only neighboring augmented voxels have direct interactions on each other.

The choice of N together with the conditional probability distribution of $P(f_i \mid f_{\mathcal{S}-\{i\}})$, provides a powerful mechanism for modeling spatial continuity and other scene features. On one hand, we choose to model a neighborhood N_i as a square mask of size $n \times n$ centered at pixel location i, where only those augmented voxels with already assigned intensity and range values are considered in the synthesis process. Thus, the neighborhood is in fact, of an arbitrary shape depending on the current available information on each of its augmented voxels. On the other hand, as it was already mentioned in Chapter 4, calculating the conditional probabilities in an explicit form to infer the exact maximum *a posteriori* (MAP) in MRF models is intractable. We cannot efficiently represent or determine all the possible combinations between pixels with its associated neighborhoods. Various techniques exist for approximating the MAP estimate, such as Markov Chain Monte Carlo (MCMC), iterated conditional modes (ICM), maximizer of posterior marginals (MPM), etc. Refer to Chapter 4 for a more complete list of the existing methods and their descriptions.

In our research, we avoid the computational expense of sampling from a probability distribution and take two different approaches for the MAP-MRF estimation. The first approach is based on a non-parametric sampling strategy that is easy to implement, generates good results and is fast to execute. The second approach, which is an extension of the first, uses the belief propagation (BP) algorithm to compute marginal probabilities. This approach improves the synthesized results of those of the first approach, at the expense of additional computation.

In the following sections, we describe each of these approaches and compare their results in Chapter 6, where our experimental results are shown.

3. Non-parametric Sampling and MAP-MRF estimation

To compute the MAP estimate for a depth value R_i of the augmented voxel V_i, one first needs to construct an approximation to the conditional probability distribution $P(f_i \mid f_{N_i})$ and then sample from it. We could use the parameters themselves, estimated from the given input data or samples (neighborhoods from input intensity and range data), as an approximation to maximum likelihood sampling, however this is hindered by the need to compute the partition function Z which is usually computationally intractable (refer to Chapter 4, Section 3.3). Instead, for each new depth value $R_i \in \mathrm{R}$ to estimate, the samples, which correspond to the neighborhood system for the voxel location $i, \mathcal{N}_i$, are queried and the distribution of R_i is constructed as a histogram of all possible values that occurred in the samples. $\mathcal{N}_i$ is a subset of the real infinite set of augmented voxels, denoted by $\mathcal{N}_{real}$.

Based on our MRF model, we assume that the depth value R_i depends only of its immediate neighbors in intensity and range, i.e. of N_i. If we define a set

$$\Gamma(R_i) = \{N^\star \subset \mathcal{N}_{real} : \parallel N_i - N^\star \parallel = 0\} \tag{5.1}$$

containing all occurrences of N_i in $\mathcal{N}_{real}$, then the conditional probability distribution of R_i can be estimated with a histogram based on the depth values of voxels representing each $N^\star$ in $\Gamma(R_i)$.

Unfortunately, we are only given $\mathbf{V}$, i.e., a finite sample from $\mathcal{N}_{real}$. Thus, there might not be any neigbhorhood containing exactly the same characteristics in intensity and range as N_i in $\mathbf{V}$. Thus, we must use a heuristic which let us find a plausible $\Gamma'(R_i) \approx \Gamma(R_i)$ to sample from.

Let $\mathcal{A}_p$ be a *local* neighborhood system for the augmented voxel p, that comprises nearby neighborhoods within a radius r,

$$\mathcal{A}_p = \{A_q \in \mathcal{N} \mid distance(p, q) \leq r\} \tag{5.2}$$

where R_q is a known depth value. In other words, the local neighborhood system contains all the neighborhoods which center voxels have already assigned labels (depth values), located at a maximum distance r from the location of voxel p. This set will conform the training/sample data for that particular voxel p.

In the non-parametric approach, a depth value R_p from the augmented voxel V_p with neighborhood N_p, is synthesized by first selecting the most similar neighborhood (N_{best}) to N_p, i.e., the closest match to the region being filled in,

$$N_{best} = \underset{A_q \in \mathcal{A}_p}{\mathbf{argmin}} \parallel N_p - A_q \parallel, \tag{5.3}$$

Second, all the neighborhoods A_q in $\mathcal{A}_p$ that are similar (within a threshold ϵ) to this N_{best} are included in $\Gamma'(R_p)$, as follows

$$\parallel N_p - A_q \parallel < (1+\epsilon) \parallel N_p - N_{best} \parallel \tag{5.4}$$

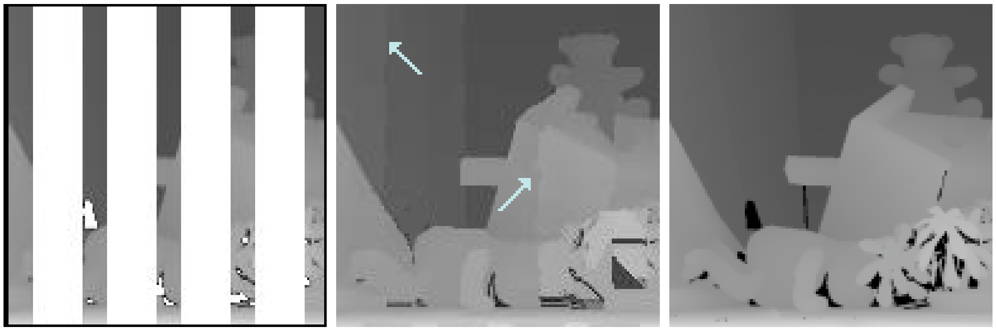
The similarity measure $\parallel . \parallel$ between two generic neighborhoods N_a and N_b is described over the partial data in the two neighborhoods and is calculated as follows,

$$\parallel N_a - N_b \parallel = \sum_{\vec{v} \in N_a, N_b} G(\sigma, \vec{v} - \vec{v}_0) \sqrt{(I^a_{\vec{v}} - I^b_{\vec{v}})^2 + (R^a_{\vec{v}} - R^b_{\vec{v}})^2}, \tag{5.5}$$

where $\vec{v}_0$ represents the augmented voxel located at the center of the neighborhoods N_a and N_b, $\vec{v}$ is a neighboring voxel of $\vec{v}_0$. I^a and R^a are the intensity and range values of the neighboring augmented voxels of the depth value $R_p \in \vec{v}_0$ to synthesize, and I^b and R^b are the intensity and range values to be compared with and in which, the center voxel $\vec{v}_0$ has already assigned a depth value. G is a 2-D Gaussian kernel applied to each neighborhood, such that those voxels near the center are given more weight than those at the edge of the window.

We can now construct a histogram from the depth values R_p in the center of each neighborhood in $\Gamma'(R_p)$, and randomly sample from it. R_q is then used to specify R_p. For each successive augmented voxel this approximates the maximum *a posteriori* estimate.

Experimental results using this non-parametric sampling approach are shown in Chapter 6. While it seems sufficient to infer depth values by learning the local relationships among the nodes in a *local* neighborhood system around the voxel whose depth value is to be inferred, yet, there are some issues. In particular, when the local information does not capture correctly the global, smooth variations in depth, the reconstructions may lead to piecewise constant (i.e., fronto-parallel) surfaces along surfaces like walls. Consider imaging a wall in which thick stripes of range data are missing. If we only observe the local neighborhoods on each side of the missing range, we will tend to get estimates of constant depth, because there were no observations of the wall sloping away at the desired depths. An example of this is shown in



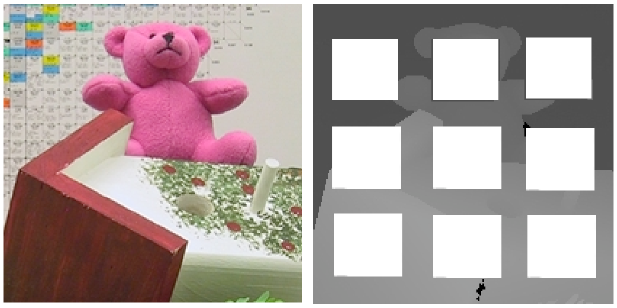
(a) Input range image (b) Synthesized range (c) Ground thruth range

Figure 5.7. An example illustrating the problem of using the non-parametric sampling method. The synthesized range image (b) shows an artifact of sudden depth variation (indicated by the arrows).

Figure 5.7: in (a) is the input range image, (b) is the synthesized range after using the non-parametric sampling approach, and (c) the ground truth range. In this prototypical case, we can see a difficulty: An artifact of sudden depth variation (the regions indicated by the arrows in (b)) is apparent (marked with an arrow). As the synthesis process advances from the borders to the interior, only local information is considered at each time, in consequence, when they get to the meeting point (the middle in this case), their depth values are not showing a smooth transition, instead a sudden change in their depth values is obtained. Therefore, a mechanism to propagate information from all sides of the area to be filled in is needed. Thus, we do not just need to model the relationships between local regions of images and scenes, but also between neighboring local scene regions. This can be done by propagating the current initial estimates by means of the belief propagation algorithm (introduced in Section 4 of Chapter 4).

4. MAP-MRF using Belief Propagation (BP)

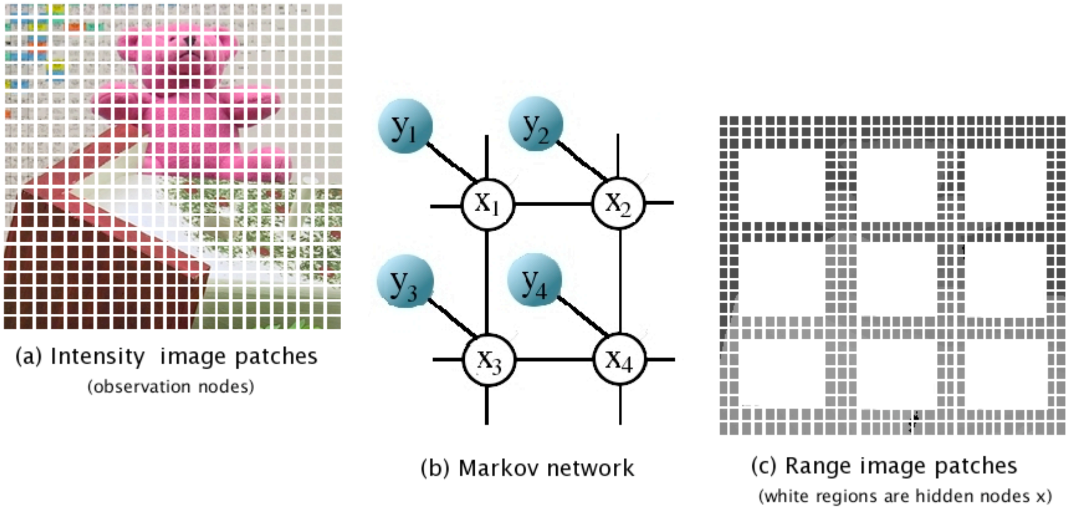
In order to propagate evidence, we use a pairwise Markov network (see Section 4.1 of Chapter 4). In our case, observation nodes represent image patches where some of the voxels may contain both intensity and range information, or only intensity information; the hidden nodes are the depth values to be estimated of the center voxels



(a) Input intensity (b) Input range

Figure 5.8. Input data used to illustrate the MAP-MRF using BP.

of the image patches of the observable nodes. BP efficiently estimates Bayesian beliefs in the MRF network by iteratively passing messages between neighboring nodes. We illustrate this by using an example. Figure 5.8 shows the input intensity and range images, where white areas represent unknown range values. The pairwise Markov network for the range estimation problem is depicted in Figure 5.9(b), where the observation node y_i is a neighborhood in intensity centered at voxel location i (see (a)), and the hidden nodes x_i represent the depth values to be estimated (white areas in (c)), but also hidden nodes contain the already available range data (as image patches), whose beliefs remain fixed at all times; a subset of these available range is



(a) Intensity image patches

(observation nodes)

(b) Markov network

(c) Range image patches

(white regions are hidden nodes x)

Figure 5.9. Pairwise Markov network for the range estimation problem.

used to locally train the network and compute compatibilities between observation and hidden nodes and the rest is used for propagating the beliefs *in* and *out* globally.

4.1. Learning the Compatibility Functions

The training pairs of intensity image patches with its corresponding range image patches are used to learn the compatibility functions. However, not all available (observed) pairs of intensity and range image patches are used as training pairs. We choose only a *local* set of image pairs that are located up to a distance d from the voxel location of the depth value to be estimated. This reflects our heuristics about how the intensity values locally provide knowledge about the type of surface that intensity value belongs to.

As in the method described elegantly in [**59**], we use the overlapping information from the intensity image patches themselves, to estimate the compatibilities $\Psi(x_j, x_k)$ between neighbors. Let k and j be two neighboring intensity image patches. Let d^l_{jk} be a vector of pixels of the lth possible candidate for image patch x_k which lie in the overlap region with patch j. Likewise, let d^m_{kj} be the values of the pixels (in correspondence with those of d^l_{jk}) of mth candidate for patch x_j which overlap patch k (see Figure 5.10). We say that image candidates x^l_k (candidate l at node k) and x^m_j are compatible with each other if the pixels in their region of overlap agree. We assume that the image and training samples differ from the "ideal" training samples by Gaussian noise of covariance σ_i and σ_s, respectively. Those covariance values are

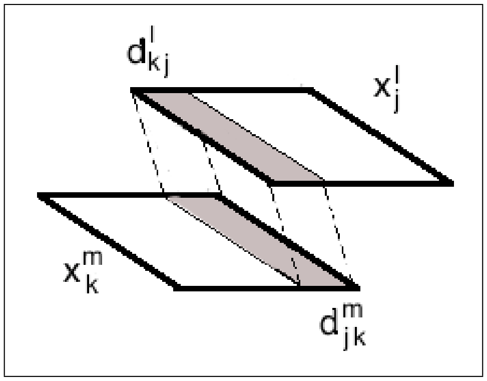


FIGURE 5.10. The compatibility between range image patch candidates.

parameters of the algorithm. Then, the compatibility matrix between range nodes k and j are defined as follows:

$$\Psi(x_k^l, x_j^m) = \exp^{-|d_{jk}^l - d_{kl}^m|^2/2\sigma_s^2}. \tag{5.6}$$

The rows and columns of the compatibility matrix $\Phi(x_k^l, x_j^m)$ are indexed by l and m, the range image candidates at each node, at nodes j and k.

We say that a range image patch candidate x_k^l is compatible with an observed intensity image patch y_0 if the intensity image patch y_k^l, associated with the range image patch candidate x_k^l in the training database matches y_0. Since it will not exactly match, we must again assume "noisy" training data and define the compatibility

$$\Phi(x_k^l, y_k) = \exp^{-|y_k^l - y_o|^2/2\sigma_s^2}. \tag{5.7}$$

4.2. The MAP estimate

The maximum a posteriori (MAP) range image patch for node i is:

$$x_{iMAP} = \arg \underset{\mathbf{x}_i}{\max}\, \Phi(x_i, y_i) \prod_{j \in \mathrm{N}(i)} M_{ji}(x_i). \tag{5.8}$$

where $N(i)$ are all node neighbors of node i, and M_{ji} is the message from node j to node i and is computed as follows:

$$M_{ij}(x_j) = Z \sum_{x_i} \Psi(x_i, x_j) \Phi(x_i, y_i) \prod_{k \in \mathrm{N}(i) \backslash \{j\}} M_{ki}(x_i) \tag{5.9}$$

where Z is the normalization constant.

Resuming, the algorithm for the range estimation problem using the MRF-BP framework is as follows:

Currently, the main drawback of using the BP-based method is the computational time. The BP-based algorithm runs in $O(nk^2T)$ time, where n is the number of pixels to be synthesized, k is the number of possible labels for each pixel and T is the number of iterations. It takes $O(k^2)$ time to compute each message and there are $O(n)$ messages per iteration. The NP sampling method runs in $O(Mpn)$ time,

i) Divide the input registered images, i.e., the intensity and the (incomplete) range images, into small patches, which form the sets of x_i's and y_i's. The areas of unknown depth values, i.e., $\mathbf{\Omega}$, is divided in such a way that each voxel with missing depth has its own associated intensity image patch and available range patch.

ii) For each intensity image patch y_i in $\mathbf{\Omega}$, find the k closest y_{x_i}'s from the local training set located up to distance d from that voxel location. The corresponding x_i's are the candidates for that patch.

iii) Calculate the compatibility function $\Phi(x_i, y_i)$ according to Eq. 5.7.

iv) For each pair of neighboring input patches, calculate the $k \times k$ compatibility function $\Psi(x_i, x_j)$ according to Eq. 5.6.

v) Estimate the MRF-MAP solution using BP.

vi) Assign the depth value of the center pixel of each estimated maximum probability patch x_{iMAP} to the corresponding pixel in the range image patch.

where M is the number of neighborhoods to be compared with, p is the number of pixels that conform the neighborhood of each pixel (e.g., $5 \times 5 = 25$ pixels). It usually takes around 30 seconds for an image of size 128×128 in a 3 GHz Pentium 4 with 1 Gigabyte of RAM.

Aq critical issue on the range estimation process is the order in which the depth values are synthesized, we described our approach for the range synthesis ordering in the next section. Here, we just mention that as messages are propagated in the Markov network, we have set a message-passing rule that are essentially constrained on the edge information from the intensity images.

5. Range Synthesis Ordering

In standard MRF methods, the assumption is that the field is updated in either stochastically or in parallel according to an iterative schedule. In practice, several authors have considered more limited update schedules. In the presented work, a single update is done at each unknown measurement. Thus, a depth value $R(x, y)$ is synthesized sequentially (although this does not preclude parallel implementations).

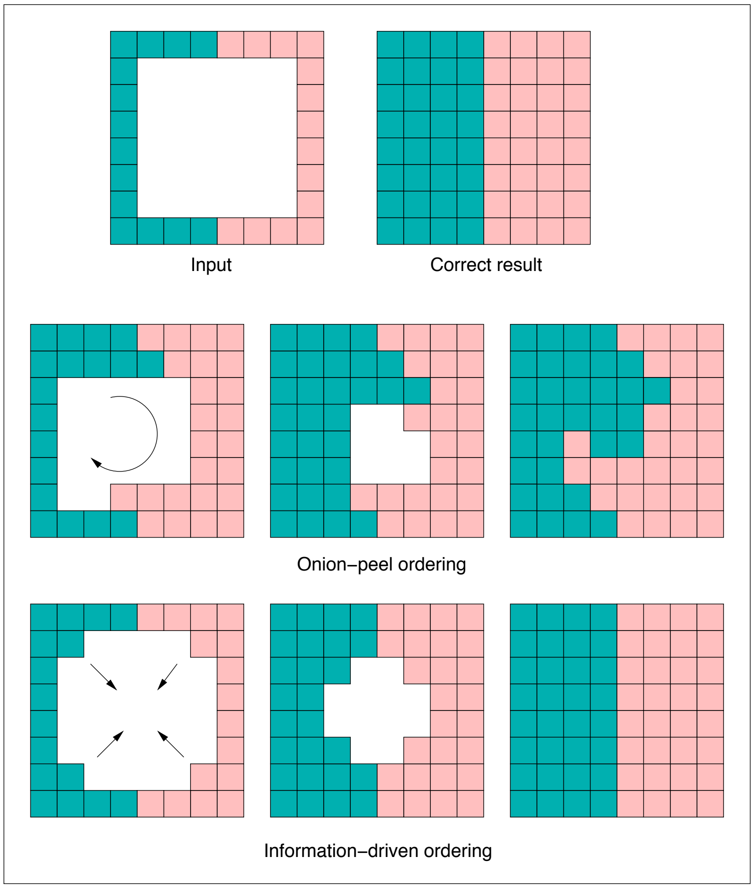


FIGURE 5.11. Comparing the onion-peel and information-based orderings.

Critical to the quality of the reconstruction is the order in which a voxel, whose range value is to be synthesized, is selected. An ordering that is commonly used, is the well-known onion-peel ordering. This ordering uses a predetermined schedule over space, essentially walking a spiral from the boundary of a region towards the center. The main problem with this ordering is the strong dependence from the previous assigned voxel. A more suitable ordering is based on the amount of available information in the voxel's neighborhood, such that voxels containing the maximum number of neighboring augmented voxels are synthesized first. Figure 5.11 shows a simulated example that compares the onion-peel ordering with the information-driven ordering proposed in this thesis.

It was also observed that the reconstruction across depth discontinuities is often problematic as there is comparatively little constraint for probabilistic inference at these locations. Such regions are often identified with edges in the intensity images and with linear structures in the range maps. These linear structures are called *isophotes*, which in the range domain are defined as all normals forming same angle with direction to eye (see Figure 5.12). Thus, in the reconstruction sequence, synthesis of voxels close to intensity discontinuities (indicated by edges) and/or depth discontinuities (indicated by the isophotes) are deferred as much as possible.

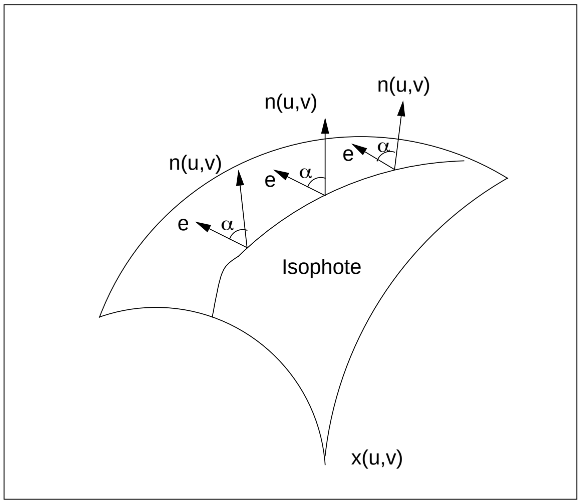


FIGURE 5.12. Isophote on a surface. This linear structure is formed such that all normals on it, form the same angle with direction to the eye.

Summarizing, the reconstruction sequence synthesize first the depth values of those voxels for which we can make the most reliable inferences based on essentially two factors: 1) the number of neighboring augmented voxels (i.e. locations with already assigned range and intensity) and, 2) the existence of intensity and/or depth discontinuities (i.e. if an edge or a linear structure exists). Priority values are computed based on these two factors and are assigned to each voxel for reconstruction, such that as we reconstruct, the voxel with the maximum priority value is selected. If more than one voxel shares the same priority value, then the selection is done randomly.

5.1. Computing the Priority Values

The reconstruction sequence depends entirely on the priority values that are assigned to each voxel on the boundary of the region to be synthesized. The priority computation is biased toward those voxels that are surrounded by high-confidence voxels, that are not on an isophote line, and whose neighborhood does not represent an intensity discontinuity, in other words, whose neighborhood does not have any edges on it. Furthermore, edge information is used to defer the synthesis of those voxels that are on an edge to the very end. When using the BP-based method, we essentially use the following message-passing rule:

Edge-constraint rule: *A node can send a message to a neighboring node once it has received messages from all its other neighbors except those edge-labeled nodes.*

We start by giving some basic notations to explain how the methods we present reconstruct the unknown depth values (notation similar to that used in the inpainting literature [**31**]).

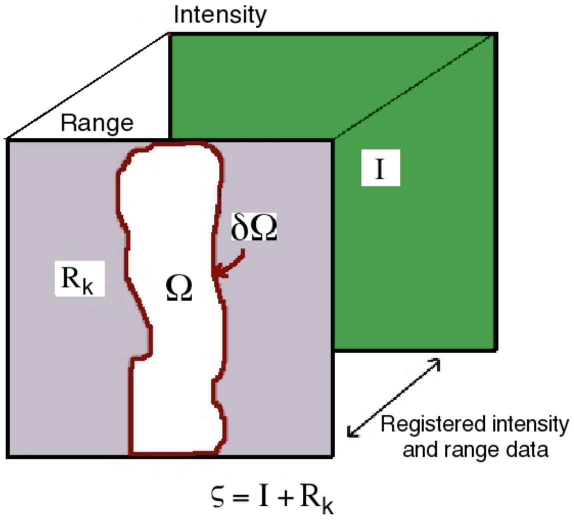


FIGURE 5.13. The notation diagram.

Figure 5.13 shows the notation diagram. The region to be synthesized, i.e., the *target* region is indicated by $\mathbf{\Omega} = \{\omega_i \mid i \in \mathcal{A}\}$, where $\omega_i = R(x_i, y_i)$ is the unknown depth value at location (x_i, y_i) and $\mathcal{A} \subset Z_m$ is the set of subscripts for the unknown range data. The contour of Ω is denoted by $\delta\Omega$. The input intensity (I) and the known range values (R_k) together form the *source* region, and is indicated by ζ. This region ζ is used to calculate the statistics between the intensity and the input range for reconstruction, as it was described in Section 3. Let V_p be an augmented voxel with unknown range located at the boundary $\delta\Omega$ and N_p be its neighborhood, which is a $n \times n$ square window centered at V_p.

For all voxels $V_p \in \delta\Omega$, their priority value is computed (which is going to determine the order in which they are filled) as follows:

$$P(V_p) = C(V_p)D(V_p) + 1/(1 + E). \tag{5.10}$$

where E is the number of edges found in the neighborhood or image patch N_p; $C(V_p)$ is the *confidence* term, and $D(V_p)$ the *data* term. The confidence term is defined as follows:

$$C(V_p) = \frac{\sum_{i \in N_p \cap \zeta} C(V_p)}{|N_p|}, \tag{5.11}$$

where $|\mathcal{N}_p|$ is the total number of voxels (augmented or not) in $\mathcal{N}_p$. At the beginning, the confidence of each voxel is assigned 1 if its intensity and range values are filled and 0 if the range value is unknown. This confidence term $C(V_p)$ may be thought of as a measurement of the amount of reliable information surrounding the voxel V_p. Thus, as we reconstruct, those voxels whose neighborhood has more of their voxels already filled, are synthesized first, with additional preference given to voxels that were synthesized early on.

The data term $D(V_p)$ is computed using the available range data in the neighborhood $\mathcal{N}_p$, as follows (see Figure 5.14):

$$D(V_p) = \frac{\alpha}{|\nabla_{I_p}^{\perp} \cdot n_p|}. \tag{5.12}$$

where α is a normalization factor (e.g. $\alpha = 255$, in a typical gray-level image), n_p is a unit vector orthogonal to the boundary $\delta\Omega$ at voxel V_p. This term reduces the priority of a voxel in whose neighborhood an isophote "flows" into, thus altering the

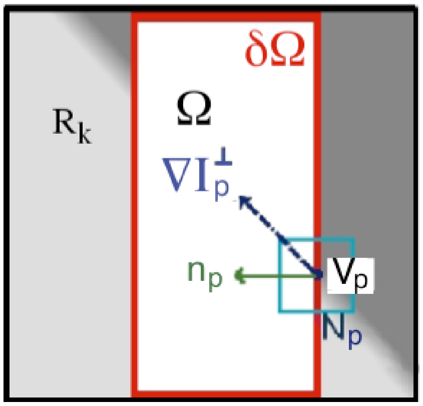


FIGURE 5.14. The diagram shows how priority values are computed for each voxel V_p on $\delta\Omega$. Given the neighboorhood of V_p, n_p is the normal to the contour $\delta\Omega$ of the target region Ω and $\nabla_{I_p}^{\perp}$ is the isophote (direction and range) at voxel location p.

sequencing of the extrapolation process. This term plays an important role in the algorithm because it prevents the synthesis of voxels lying near a depth discontinuity. Note, however, that it does not explicitly alter the probability distribution associated the voxel (except by deferring its evaluation), and thus has only limited risk for the theoretical correctness of the algorithm.

Once all priority values of each voxel on $\delta\Omega$ have been computed, we find the voxel with the highest priority. We then use our MRF model to synthesize its depth value. After a voxel has been augmented (i.e. it has intensity and range data), the confidence of the $C(V_p) = C(V_q)$, i.e. it is assigned the confidence of the augmented voxel which most resemble the neighborhood of V_p (see Eq. 5.11).

6. Implementation Issues

6.1. The searching space

The searching space depends on the parameter d, i.e., the distance from the voxel to be synthesized p to the farthest neighboring voxel, which neighborhood is going to be compared to. Thus, if d is equal to the size of the image, then N_p is compared with all possible neighborhoods in the image, which would be a time consuming process. The exhaustive search takes $O(n^2)$ for each voxel to be synthesized, where

n is the total number of known augmented voxels whose neighborhoods are being compared to. Therefore, d should be of a reasonable size, not to big not to small, so that appropriate candidates, which are similar enough in their neighborhoods, can be found. However, if the searching space must be large for whatever reason, the searching time can be reduced by indexing the n augmented voxels using a kd-tree structure [**12**], so that the new computational time is $O(\text{logn})$.

6.2. Neighborhood size

Neighborhoods should be as small as possible to minimize processing but should be large enough that features are not missed. Additionally, if neighborhoods are too large, small features are more easily missed in the processing phase. In other words, ideal neighborhoods would include enough voxels to easily distinguish variations in surface structure but be no larger.

6.3. Similarity between neighborhoods

There is also the question of how ”similarity” between two neighborhoods is measured. This is partly determined by the model through neighborhood size. However, it is also determined by how we measure the classification probability. We determine a good fit probabilistically by comparing the set of statistics obtainable from the available data (see Section 3). The consequence of this, however, is that when the MRF model is under trained, the statistics will not change rapidly as the similarity between the neighborhood of the voxel to synthesize and the neighborhood from the training data diminishes. Therefore neighborhoods which are not all that similar could be given a high similarity measure. On the other hand, if the MRF model is overtrained, then the statistics obtainable from the training data would have a high entropy, and again dissimilar neighborhoods could be given a high similarity measure.

CHAPTER 6

Experimental Results on Alternative Sampling Strategies

In this chapter we show experimental results conducted on data acquired in a real environment to evaluate both the non-parametric (NP) sampling method and the belief propagation (BP) technique described in Chapter 5. We use ground truth data from two widely available databases containing sets of registered intensity and range data. Different sampling strategies on the range data were tested to evaluate our methods. We demonstrate the versatility of the methods as well as its limitations.

1. Experimental Setup

In our experimental setup, intensity images are input directly to our algorithm, while the complete range data provides ground truth and a limited subset, selected with different sampling strategies, is used to simulate sparse readings and to provide input to the algorithm. Having ground truth data allow us to compare the quality of the reconstruction with what is actually in the scene.

Experiments are conducted on data acquired on a number of real environments. Both intensity (achromatic and color) images and real range data files, already registered, are obtained from two databases available on the web. The first database, the

USF range image database [1] from the CESAR lab at Oak Ridge National Laboratory, provides real intensity (reflectance) and range images of indoor scenes acquired by an Odetics laser range finder mounted on a mobile platform. The second database, the Middlebury stereo database [2] [**131**], provides color images with complex geometry and pixel-accurate ground-truth disparity data.

The first dabase contains images of size 128 × 128 pixels represented by 256 grey levels and the second database contains color images of size 450 × 375 pixels. The neighborhood systems used in all experiments were windows of 5 × 5 pixels, unless otherwise indicated. The Canny edge detector[**22**] is used to extract the edges of the input intensity images and the smoothing parameter is set to 0.8. The parameter d, i.e., the maximum distance between two neighborhoods' center voxels, which are to be compared to find the best matches, is set to 10 pixels. For the case of the BP-based method, the number of iterations were set to 50. We have conducted many experiments on the number of iterations, and observed that below 50 iterations the results are poor, while iterations above this number do not reflect much of improvement.

We consider several strategies for sampling the range data. In general, the sampling of the range data is assumed to be clumped into at least some sets of mutually-adjacent voxels as opposed to scattered measurements far from one-another. The sampling strategies presented in the next sections are not necessarily identical to those provided by any particular sensor (except for those that are indicated). However, they are useful for analyzing and testing our methods for range synthesis so we can determine suitable samplings on the range data to have a good reconstruction. Some of these results have been already published [**158, 156, 159, 157**].

1.1. Evaluation Methodology

Since we have ground truth range in all of the experiments, we can evaluate the performance of our method. There are different techniques to compute the errors in

[1] http://marathon.csee.usf.edu/range/DataBase.html
[2] http://www.middlebury.edu/stereo

the reconstruction. We choose to compute the mean absolute residual (MAR) error. The absolute value of each error is taken and the mean of those values is computed to arrive at the MAR error. The MAR error of the two range data matrices R_1 and R_2 is defined as:

$$\text{MAR error} = \frac{\Sigma_{i,j}|R_1(i,j) - R_2(i,j)|}{\#\text{ of unknown range pixels}} \tag{6.1}$$

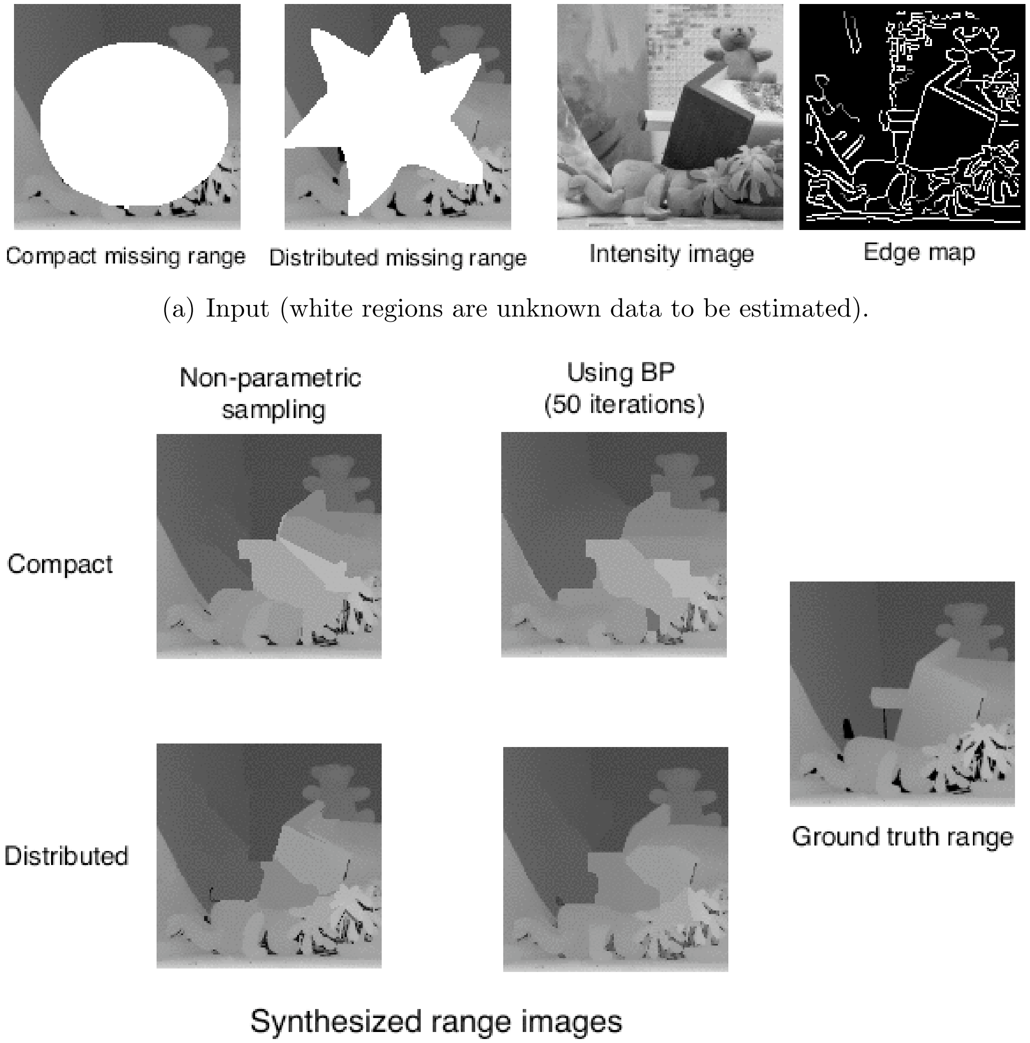
In general, however, just computing MAR errors are not a good way of assessing the success of our methods. For example, when there exists a few number of outliers with high MAR errors, the average MAR error is slightly high. We can visually appreciate a good or bad reconstruction by looking at a *normalized* difference/residual image Γ, using the following equation:

$$\Gamma_(i,j) = \frac{|R_1(i,j) - R_2(i,j)| * 255}{k} \tag{6.2}$$

where the value k is the maximum difference error value between the two range data images. For some of the experiments, we show the normalized residual images.

2. Areas with unknown range data of arbitrary shapes

The first type of experiment involves the range synthesis when the region spanning the missing range data is of irregular shape. In particular, we show how the shape that contains the unknown range influences their estimation. In Fig. 6.1a, two input range images (the two left images) are given. The percentage of the missing area (shown in white) of both range images is 47.6% (7800 pixels). Their perimeters and shape however are different. The first range image shows a compact shape whereas the second image represents a more distributed missing range. In the same Figure, the two right images depict the input intensity and its corresponding edge map, respectively. These data are given as an input to our algorithm. Fig. 6.1b shows in the left two columns, the synthesized range images. The first row are the results for the compact shape in Figure 6.1a and the second row for the distributed shape.



(a) Input (white regions are unknown data to be estimated).

(b) Synthesized range images.

FIGURE 6.1. Results on two different shapes of unknown range with same area: 7800 pixels (47.6% of total image).

The first column indicates the synthesized results when using the non-parametric (NP) sampling method described in Section 3 of Chapter 5. The second column gives the synthesized results when using the belief propagation-based (BP) method (see Section 4 of Chapter 5) after 50 iterations. The ground truth range image (the right image) is also displayed for comparison purposes.

Although the synthesized results for both type of shapes reflect a poor approximation of the real range measurements in the depth map, it can be seen that when synthesizing big areas of unknown range data, our algorithm performs better if the area is not compact, since combinations of already known range and intensity give more information about the geometry of the scene. In other words, the sample spans a broader distribution of range-intensity combinations. This experiment gives us a good indication of what kind of sampling of the input range data we need to have a good reconstruction. In the same trend, we carried out a simple experiment on 5 different scenes, where the percentage of the total area with unknown range varies from 10% to 90%. In Figure 6.2, the input intensity and associate edge map together with the ground truth range images for two of the scenes are shown. Figures 6.3(a) and (b) show in the first column the input range for some of the cases of study. The second and third columns are the synthesized range images using the NP-sampling and the BP-based methods, respectively. Figure 6.4 is a plot of the MAR errors averaged over the 5 scenes versus the size of the unknown range data.

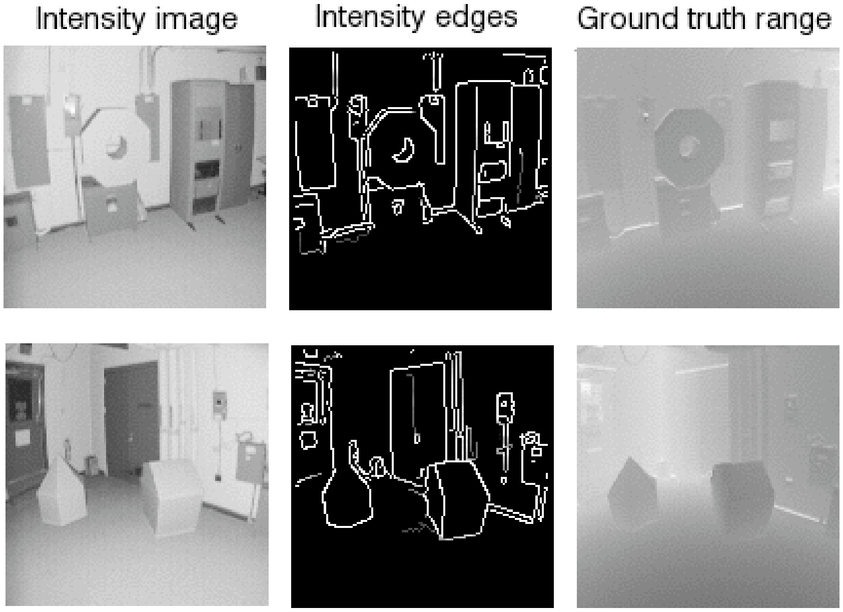


FIGURE 6.2. The input intensity, edge map and ground truth range for two of the scenes to evaluate.

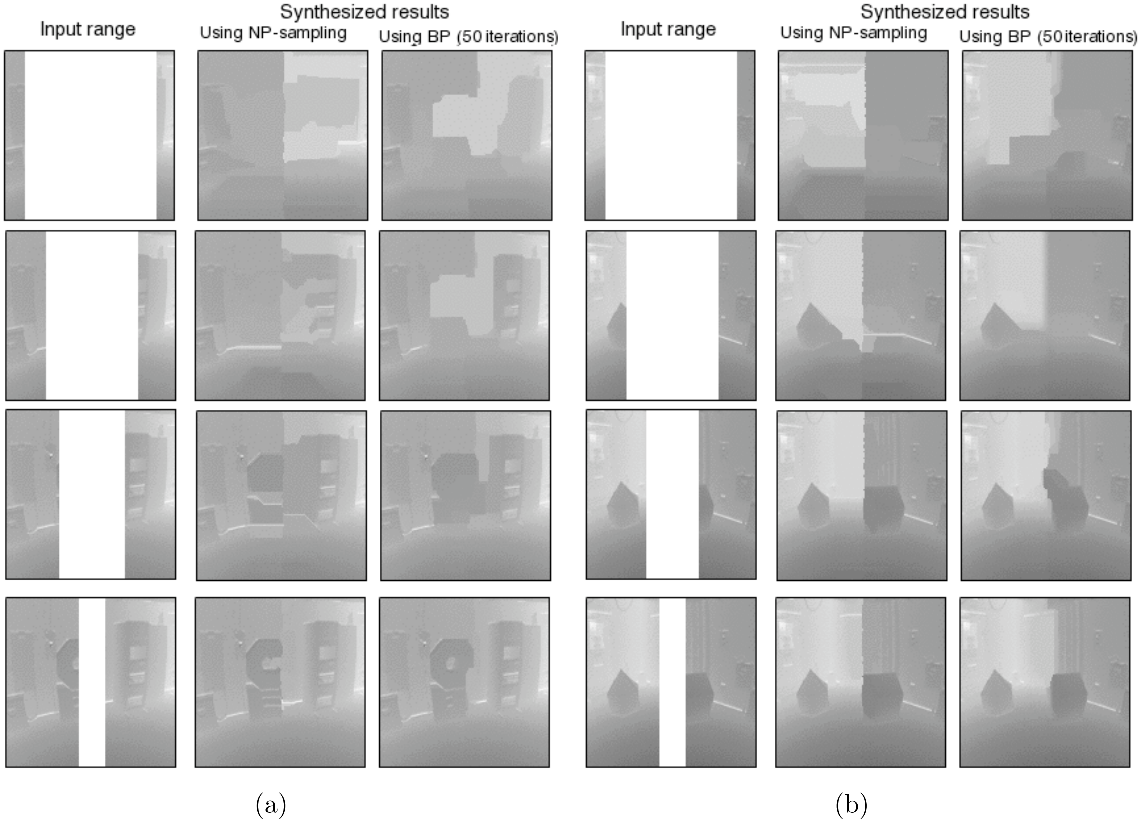


FIGURE 6.3. Results on two different scenes with varying input range.

3. Range stripes along the $x-$axis and $y-$axis

This type of experiment involves the range synthesis when the initial range data is a set of stripes with variable width along the $x-$ and $y-$axis of the intensity image. In the following cases, we tested our methods with the same intensity image in order to compare the results. Figure 6.5a shows the input intensity image (left) of size 128×128 pixels, the edge map (middle), and for purpose of comparison we show the ground truth range image (right) from which we omit using much of the data for input to simulate real sensing conditions.

Four cases of sampling are shown in Figure 6.5b. The initial range data, shown in the left column, goes from dense to very sparse. The percentage of missing range data is indicated below each image. The parameters r_w and x_w indicate the width of the stripes of known and uknown range, respectively. For example, the first case has

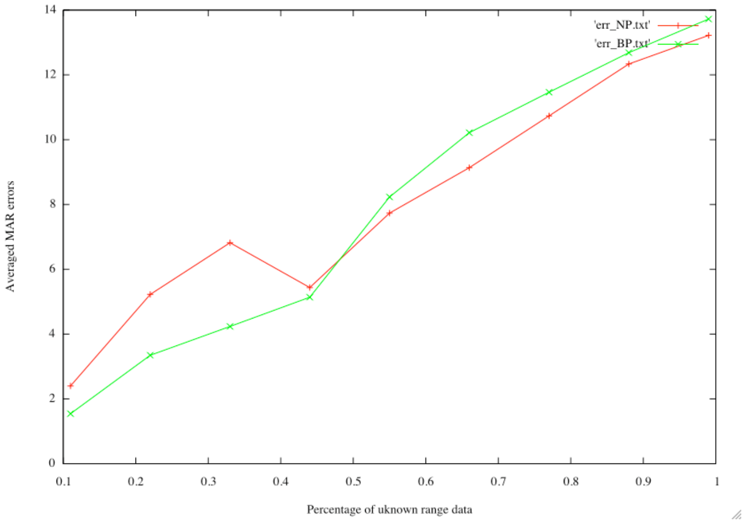


FIGURE 6.4. Plot of the averaged MAR errors over 5 different scenes.

$r_w = 24$ and $x_w = 80$. For the first three cases the size of the neighborhood is set to be 5 × 5 pixels and for the last case 3 × 3 pixels. The second and third columns show the synthesized range data obtained after running our algorithms using the NP sampling and when using the BP-based (after 50 iterations) methods, respectively. We also show the residual images for the BP-based synthesized results.

The first two cases have the same amount of missing range (39%), however the synthesized range for the second case is much better. Intuitively and based on the results from the previous section, this is because the sample spans a broader distribution of range-intensity combinations.

Table 6.1 shows the MAR errors (calculated only on the unknown areas) of the examples shown in Figure 6.5b for each of the two methods.

		% of area with	**MAR Errors (in pixels)**	
r$_w$	**x**$_w$	**missing range**	**NP sampling**	**BP-based**
24	80	39	36.36	4.23
10	20	39	5.76	1.33
5	25	61	8.86	1.85
3	28	76.5	9.99	2.76

TABLE 6.1. MAR errors for the cases shown in Figure 6.5b.

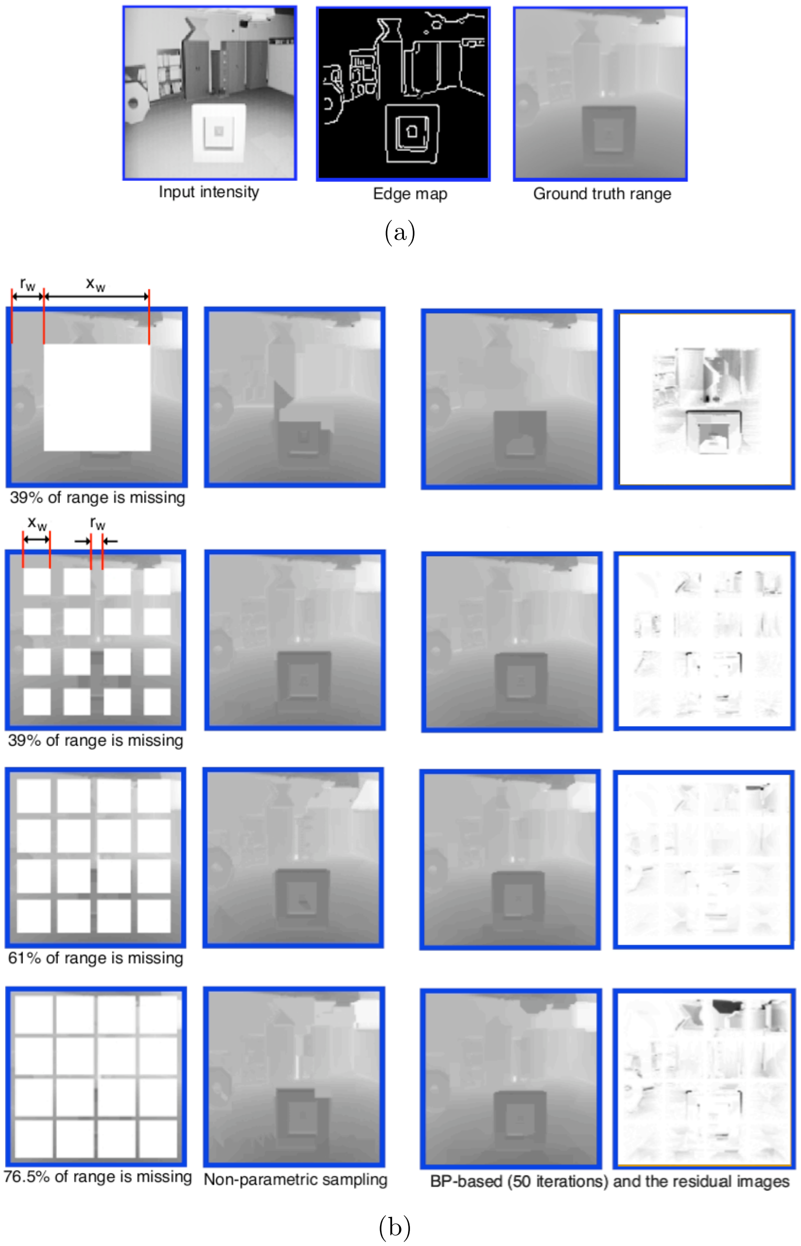


FIGURE 6.5. Results on real data. (a) From left to right, the input intensity image, the edge map and the associated ground truth range. (b) The left column shows the initial range data (the white squares represent unknown data to be estimated). The second and third columns show the synthesized results when using the NP sampling and BP-based methods, respectively. The last column are the residual images for the BP-based synthesized results.

From the data in the table, we can see that the BP-based method produces smaller errors than the NP sampling method. The reason for this is because the belief at each pixel is propagated through all its neighbors (except when edges are detected), so that it covers all the area of missing range. This allows for smooth transitions on the range values and the removal of inconsistencies to find the minimum energy of the overall system. Whereas the NP sampling method only consider one iteration per assignment of all the best matches of the already existing range values, which also induces order-dependant artefacts.

We conducted experiments on 32 images from the USF database of common scenes found in a general indoor man-made environment. We choose the sampling case with $r_w = 5$ and $x_w = 25$, which corresponds to images with 61% of the range points unknown. The average MAR errors for the ensemble of images using both the NP sampling and BP methods are shown in Table 6.2.

Mean MAR	**Errors (in pixels)**
NP sampling	BP-based
4.27	2.65

Table 6.2. Average MAR errors for 32 images in the database.

Four examples of these experiments are shown in Figures 6.6 and 6.7, which corresponds to the input and the synthesized range images, respectively. In order to confirm the importance of using edge information in the reconstruction process, we display the synthesized results in the first column of Figure 6.7 when edge information is not considered. The second and third columns show the synthesized range images when considering edge information and using the NP sampling and belief propagation-based methods, respectively. The last column displays the ground truth range images for comparison purposes.

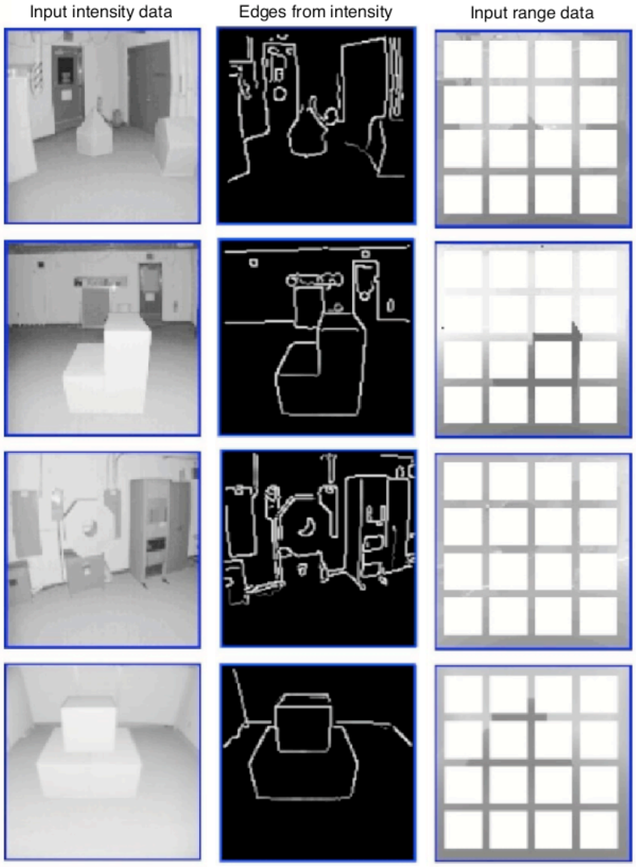


FIGURE 6.6. Input data to our range synthesis algorithms. The first two columns are the input intensity and the associated edge map; the third column shows the input range images (61% of the range is unknown).

The MAR errors in the grey-level range (i.e. 0 for no error and 255 for maximum error), from top to bottom are shown in Table 6.3.

MAR	Errors (in	pixels)
No edge-info	NP sampling	BP-based
8.58	3.77	2.58
13.48	3.03	1.63
11.39	2.99	2.03
7.12	2.20	1.36

TABLE 6.3. MAR errors for the cases shown in Figure 6.7.

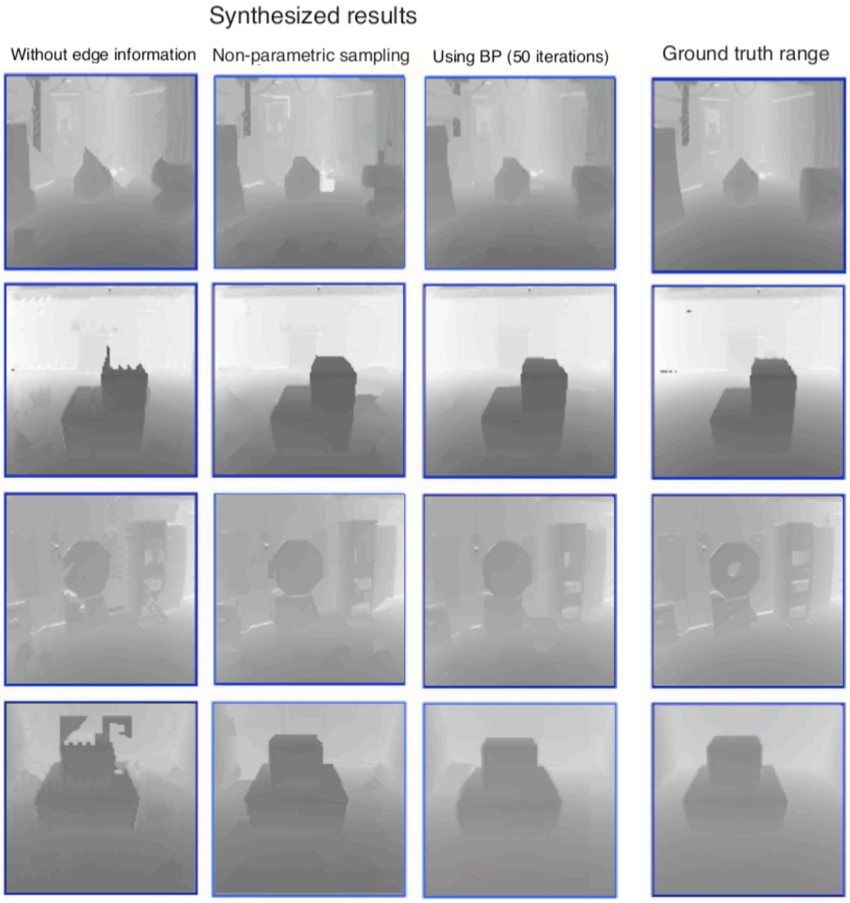


FIGURE 6.7. The first three columns are the synthesized results for the input data shown in Figure 6.6 when no edge info is used and when using the NP sampling and BP-based methods, both with edge information. The last column shows the ground truth range for visual comparison.

3.1. Using color images

We now show how color information may improve the range synthesis. Figure 6.8 displays in the first row, the grayscale and color images of the same scene, and to their right the input range data. The synthesized results after running our algorithm is shown below together with the ground truth data for comparison purposes.

It can be seen that there are some regions where color information may help in the synthesis process. For example, the chimney in the center of the image is separated from the background since they have different colors. This is hardly noticeable in the grayscale image.

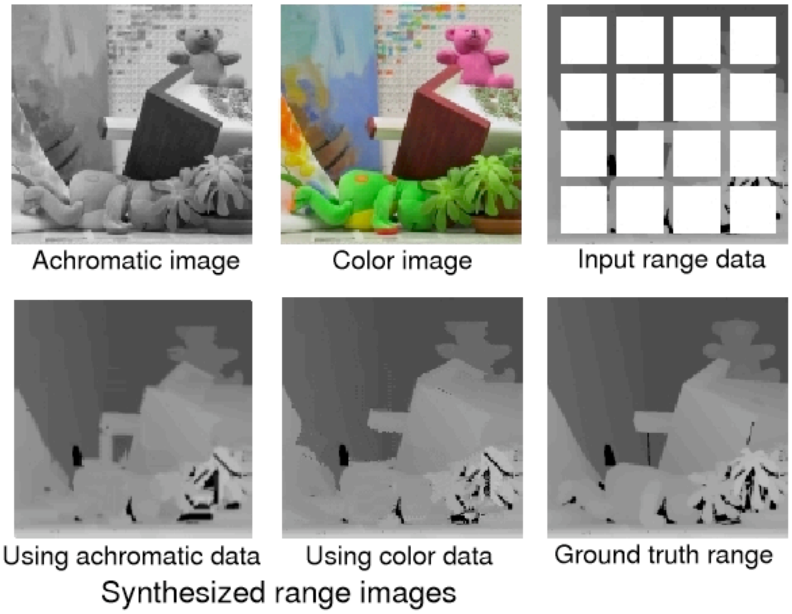


Figure 6.8. Results on achromatic and color images.

In general, when we use color images in the reconstruction process, it appears that the fidelity of the reconstruction is somewhat improved over achromatic images. This appears to be due to the fact that the color data provides tighter constraint over where and how the interpolation process should be applied. At the same time, the higher dimensionality of the Markov Random Field model for color images may make the reconstruction problem more difficult in some cases as the RGB channels have to be compared individually.

4. Stripes along the $x-$axis

Another type of experiments is where the initial range data is a set of stripes only along the $x-$axis. This set of experiments is interesting because the subsampling strategy resembles what is obtained by sweeping a two-dimensional LIDAR sensor (see Chapter 7, Section 3.3.1). This sampling, however, presents a challenge for the range synthesis process, since we have very limited range data available (just along the $y-$axis). Thus, it is in this type of sampling where the isophote information (see Chapter 5 Section 5) from the available range data, together with the edge map from the input intensity image, greatly help in the reconstruction process. As we will show in the following experiments.

Figure 6.9 displays from left to right, the input color intensity image with its corresponding edge map, and the ground truth range image from where we hold back the data to simulate the samples. Two samplings on the same range image are shown in order to compare the results. In Figure 6.10, the first column displays the initial range data. The percentage of unknown range are 65% and 62%, respectively.

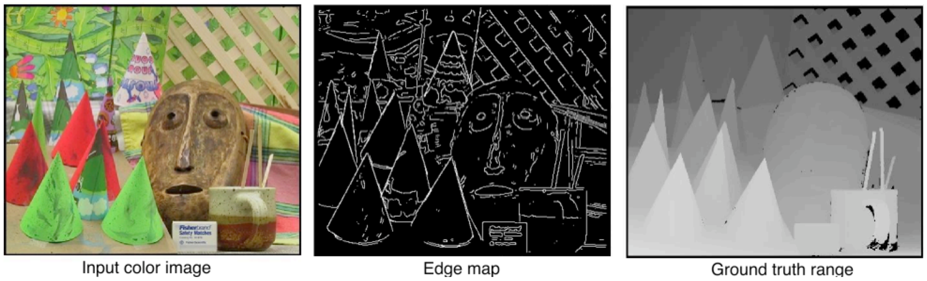


FIGURE 6.9. The input intensity image, the edge map and the associated ground truth range.

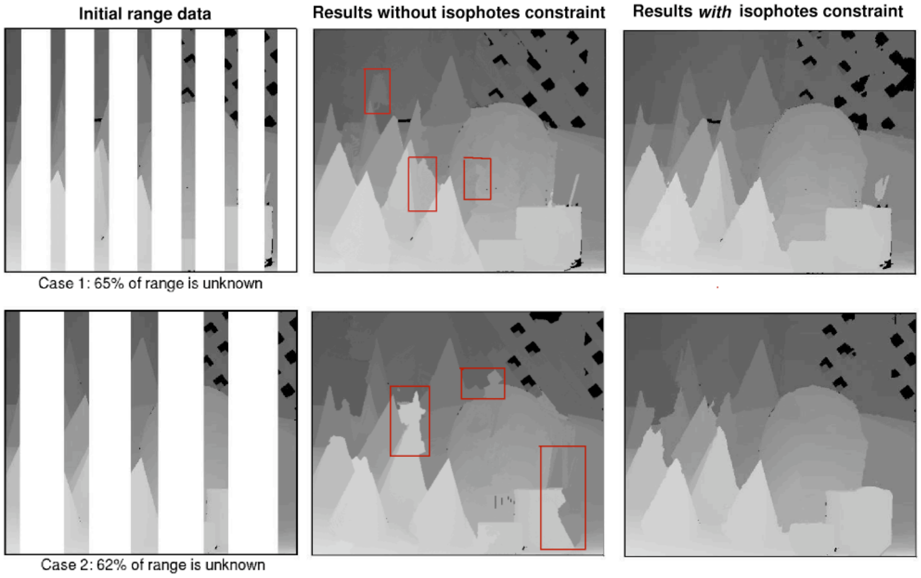

Initial range data
Results without isophotes constraint
Results *with* isophotes constraint
Case 1: 65% of range is unknown
Case 2: 62% of range is unknown

FIGURE 6.10. The initial range images are in the first column with their percentage of unknown range indicated below each. To compare results, the middle column shows the synthesized range images without using isophote information and the last column show the improved synthesized results with the incorporation of isophote constraints.

The two right columns show the synthesized range images without using isophote constraints and when they are incorporated to our algorithm. The regions enclosed by the red rectangles show where our algorithm performed poorly.

The mean absolute residual errors (MAR) are from top to bottom, 10.5 and 12.2, when no using isophote constraints compared to 6.5 and 7.3, when using isophote constraints. The algorithm was able to capture the underlying structure of the scene by being able to reconstruct object boundaries efficiently, even with the small amount of range data given as an input.

More experimental results are shown in Figure 6.11. The first row show the input color intensity image, the edge map and its associated ground truth range (for comparison purposes). Three cases of sampling are shown in the subsequent rows. The first column is the input range. The percentages of unknown range (indicated below each image) are 63%, 78% and 85%, respectively. The last two columns depicts the synthesized results when using isophotes constraints in both the NP sampling and BP-based methods, respectively. The MAR errors from top to bottom, are given in Table 6.4.

Percentage of	**MAR**	**Errors**
unknown range	NP sampling	BP-based
63%	6.42	3.90
78%	6.45	3.90
85%	7.98	4.80

TABLE 6.4. MAR errors for the cases shown in Figure 6.11.

In general, the use of isophote information improves the reconstructions because it helps to detect the continuation of linear structures from the available range. This was clearly illustrated in the examples above.

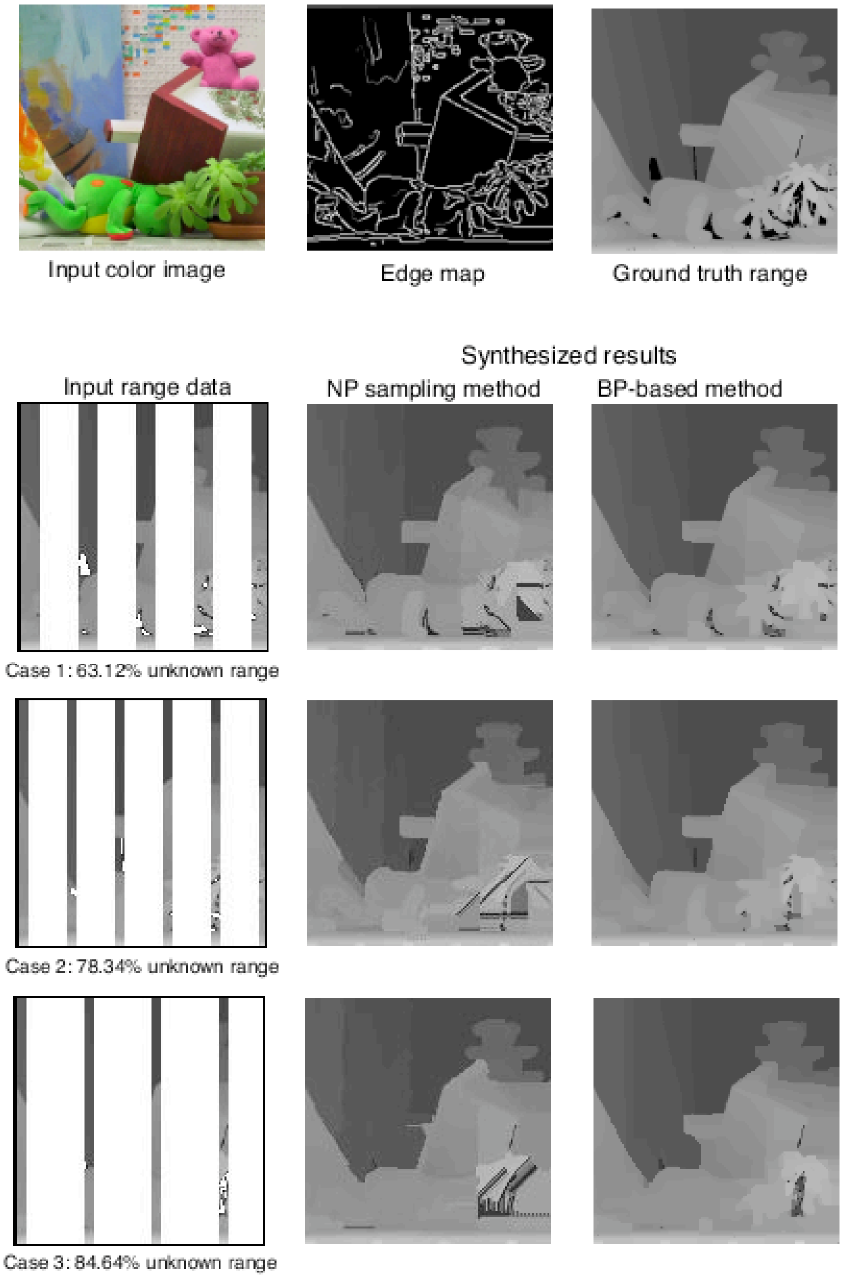


FIGURE 6.11. Results on real data. In the first row are the input color image and ground truth range. The subsequent rows show three cases, the initial range images are in the left column (percentage of unknown range is indicated below each). The two right columns show the synthesized results using the NP sampling and BP-based (50 iterations) methods, respectively.

5. Discussion

The initial range data given as an input plays a very important role in the quality of the synthesis. As the percentage of known range data increases, and if this known range data is distributed about the scene to be synthesized, our methods can produce very good results in inferring the missing range data. There are however a notable difference in the performance of both methods we presented. The synthesized result from the NP sampling method are obtained faster than those when running our BP-based algorithm. However, the results from the BP-based method are much more better than those from the NP sampling method.

The synthesized results when using the belief propagation show that our method can accomplish the propagation of geometric structure from the available neighborhood information. However, there are some regions where this propagation was not effective, again, due to the amount of initial range and on the types of surfaces captured by this initial range.

CHAPTER 7

Mobile Robot Environment Modeling

This chapter presents an application of the range synthesis method described in Chapter 5, specifically in modeling a large-scale indoor environment using a mobile robot. When modeling large environments, sensor data from different viewpoints need to be acquired. The acquisition of images is easy, however to acquire a complete and dense range map of the environment is often done by using sophisticated but costly hardware solutions. This task is time consuming for many real systems. Our approach overcomes this problem by allowing a robot to rapidly collect a set of intensity images and a small amount of range information. Then, our range synthesis method performs scene recovery, i.e., it estimates a dense range map at each location the robot takes measurements while navigating the environment. We give a complete description of the stages involved in our approach for the mobile robot environment modeling. These stages are: data acquisition and registration of the intensity and partial range data; the integration of these data at different views; and the estimation of dense range maps locally and globally. Experimental results on data acquired on our lab and building are given to illustrate the suitability of our approach.

1. Introduction

One of the major goals of mobile robot research is the creation of a 3D model from local sensor data collected as the robot moves in an unknown environment. This

3D model must contain geometric and photometric details as well as knowledge about empty spaces. The final representation of this knowledge will reflect the ability of the autonomous system to interact with its environment and accomplish its tasks.

In mobile robotics, it is common to combine information from many sensors, even using the same sensors repeatedly, over time, with the goal of building a model of the environment. Since all sensors are imperfect, sensor inputs must be used in a way that enables the robot to interact with its environment successfully in spite of measurement uncertainty. One way to cope with the accumulation of uncertainty is through *sensor fusion*, as different types of sensors can have their data correlated appropriately, strengthening the confidence of the resulting percepts well beyond that of any individual sensor's readings.

A typical 3D model acquisition pipeline is composed by a specialized 3D scanner to acquire precise geometry, and a digital camera to capture appearance information. Acquiring images is fast and easy, however, to acquire a complete and dense range map is a time and energy consuming process, unless costly and/or sophisticated hardware is used. Moreover, it is often difficult to achieve in practice, especially when dealing with dynamic environments (such as offices and indoors).

We avoid the need to obtain dense distance measurements, and rely instead on partial knowledge of the geometry of the scene. Our range synthesis method (described in Chapter 5) can then be used to estimate a dense range image from the sparse measurements at each robot pose. Thus, in modeling large environments, the challenge becomes one of trying to extract, from the sparse sensory data, an overall concept of shape and size of the structures within the environment.

2. Stages in the Mobile Robot Environment Modeling

3D environment modeling typically involves two main goals: 1) the detailed *shape* of specific objects or targets within the scene; and 2) the spatial *layout* of objects within the scene. The first goal involves the accurate reconstruction of the sensed objects, perhaps for reprojection from novel viewpoints, or for higher level shape

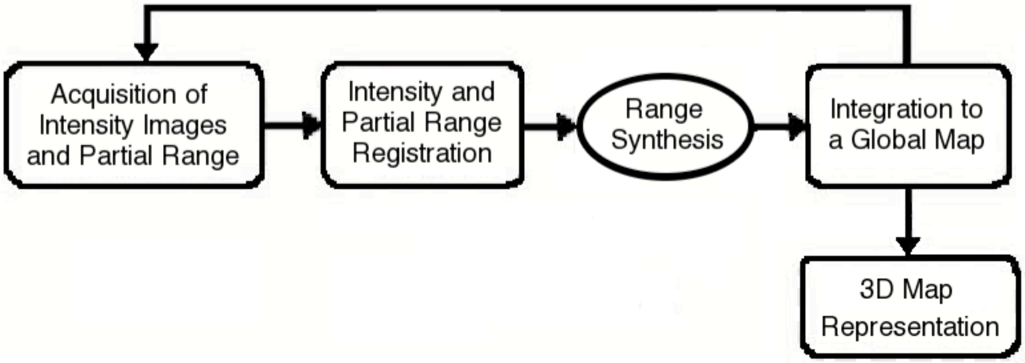

Acquisition of Intensity Images and Partial Range
Intensity and Partial Range Registration
Range Synthesis
Integration to a Global Map
3D Map Representation

FIGURE 7.1. Diagram showing the procedures to be followed for the 3D environment modeling problem.

analysis for a cultural exposition, for example. The second goal is important for the purpose of navigation and localization.

If the model is required only for navigation purposes, which is one of the most important problems in mobile robotics, then the second goal, i.e., the spatial relationships between objects is important. If the model will be used for virtual walkthroughs, example in a museum, then both goals must be fulfilled. In other words, the geometric and photometric details from specific viewpoints must be captured in the final representation.

Since we are dealing with incomplete range data, the above mentioned goals involve an additional process: the synthesis of complete and dense range maps from sparse partial distance measurements. Therefore, we divide the 3D environment modeling in the following stages (see Figure 7.1):

- data acquisition of the intensity and partial range and their registration to a common reference frame;
- range synthesis, which refers to the estimation of dense range maps locally at each robot pose and globally when integrating data from different views;
- data integration to a global map; and
- 3D model representation.

The following sections describe each of these stages in detail, except for the range synthesis stage, which was covered in Chapter 5.

3. Data Acquisition and Registration

This section describes the data acquisition system used to capture the appearance and geometry of *large-scale* indoor environments. It consists of a CCD camera and a 2D laser rangefinder mounted on a mobile robot. The main aspect of our data acquisition system relies on *how* the data is acquired, which provides two important benefits: *i*) it allows the robot to rapidly collect sparse range data and intensity images while navigating the environment to be modeled, and *ii*) it facilitates the sensor-to-sensor registration. The first benefit is essential when dealing with large environments, where the acquisition of huge amount of range data is a time consuming and impractical task. The second benefit is related to the complexity of registering different types of sensor data, which have different projections, resolutions and scaling properties. To this end, an image-based technique is presented for registering the range and intensity data that takes advantage of the way data is acquired.

3.1. Environment Analysis

Prior to the design of the system configuration, is the analysis of the environment on which the system will be used. This analysis is based on the features, surfaces, and general characteristics we expect to encounter in the areas comprising the environment to be modeled. This is of particular importance since the features that can be obtained from the sensors depend totally on the characteristics of the environment where they are acquired.

This research work focuses on modeling man-made indoor environments. Man-made indoor environments have inherent geometric and photometric characteristics that can be exploited to help in the reconstruction.

3.1.1. Geometric Characteristics

Man-made indoor environments are structured environments, i.e., they are full of structural regularities, such as the presence of sets of parallel and orthogonal lines and planes aligned with the principal orthogonal directions of the world coordinate frame. The two dominant orthogonal directions are: vertical (walls) and horizontal (floors).

Due to gravitational forces, objects are attached to the floor and/or walls. The geometric properties that are often present are: coplanarity, parallelism (alignment) and orthogonality. Some of these properties may be shared by many points, and those points can be put together into surfaces, reducing the complexity of the final representation. We assume that objects are solid volumes bounded by colored surfaces and they do not have holes. Another assumption made is that the surfaces of the objects are composed of relative smooth planar surfaces with continuity among them. This characteristic helps to assign belief values to decide if surfaces should be joined.

3.1.2. Photometric Characteristics

Different types of materials are used in man made scenes, including wood, concrete, plastics, metal, glass, etc. For environment modeling, the particular interest is in how they reflect light. All materials absorb light, some are translucent allowing light to penetrate, and all materials also reflect light. The surface texture also contributes to the image, in that smoother surfaces produce specular reflections. Moreover, lighting conditions in indoor environments may vary at different locations, and almost any large-scale environment will have some areas that are much brighter than others.

In general, the assumptions we made about the environment are:

- The environment is static.
- The environment is composed of 'smooth' planar surfaces (e.g. floors, doors, walls and furniture present large planar surfaces).
- Most of the regions contain low texture.
- The illumination of the environment is assumed to be constant, that is the brightness or darkness in some places does not dependent on the time of day that images are acquired.

In large-scale indoor environments (e.g., offices, labs, museums, etc.), all surfaces will rarely be visible from a single viewpoint, nor will data be acquired at sufficient resolution to encompass the entire layout. We use a robot to navigate the

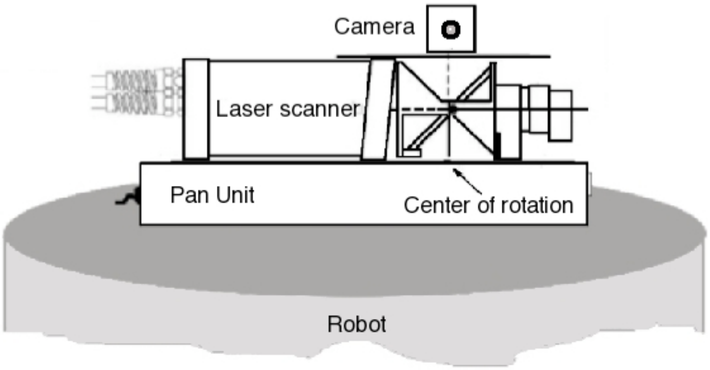


FIGURE 7.2. A diagram showing how the sensors are arranged on the pan unit on the mobile robot.

environment, and together with is sensors, captures the geometry and appearance of the environment in order to build a complete 3D model. The presented work does not consider the problem of planning the robot's trajectory, and we assume that a planning strategy is given (see for example [**65**]).

3.2. The Data Acquisition System

The aim of our system configuration is to reduce the data acquisition time and facilitate the registration process. On top of the robot, we have assembled a system consisting of a 2D *laser rangefinder* (laser scanner) and a high-resolution digital *camera*, both mounted on a *pan unit*. Figure 7.2 shows the arrangement of the sensors on the pan unit. The (CCD) camera is attached to the laser rangefinder in such a way that their center of projections (optical center for the camera and mirror center for the laser) are aligned to the center of projection of the pan unit. This alignment facilitates the sensor-to-sensor registration, that is, the registration between the intensity and range data, as we only need to know their projection types in order to do image mapping. The image-based registration algorithm is described in detail in Section 3.4.

3.2.1. The 2D Laser Rangefinder

A 2D laser range finder or laser scanner, is composed by a laser rangefinder and spinning mirror (please refer to Section 3.1.1 of Chapter 2 for definition and mechanisms details). The spinning mirror and panning motor combine to allow the laser to sweep out a longitude-latitude sphere to acquire complete range information [1] (see Figure 7.3). The data returned for each rangefinder sample comprises a fixed number of points, each point includes range r, a value representing the amplitude a of the laser light reflected back to the scanner, and the angular position of the rotating mirror θ of the scanner. Since the pan angle ϕ of the pan unit is also known, each sample is taken in polar coordinates, expressed as a quadruple (r, a, θ, ϕ), and it can be considered as two images sampled on a spherical surface: a range image $\mathbf{R}$ and an amplitude image $\mathbf{A}$.

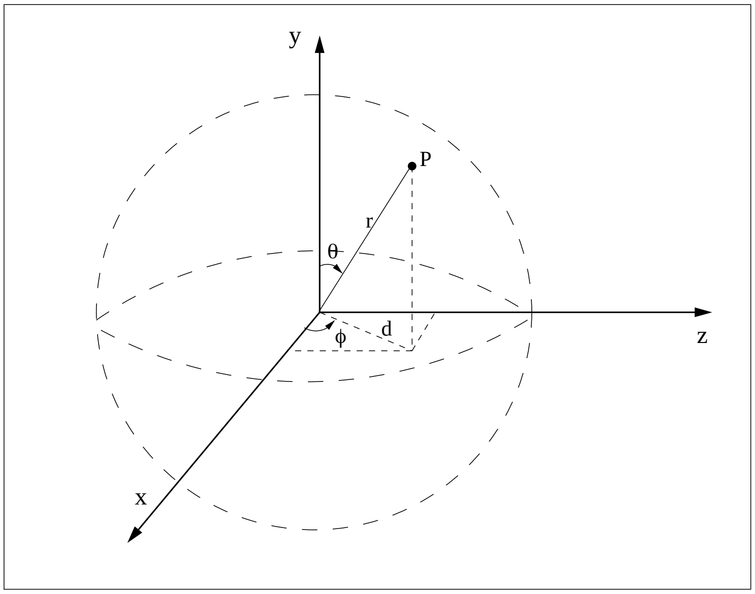


FIGURE 7.3. Laser coordinate system.

It is important to mention that the samples are not regular in longitude and latitude, since the rangefinder and scanning mirror are not synchronized. As with

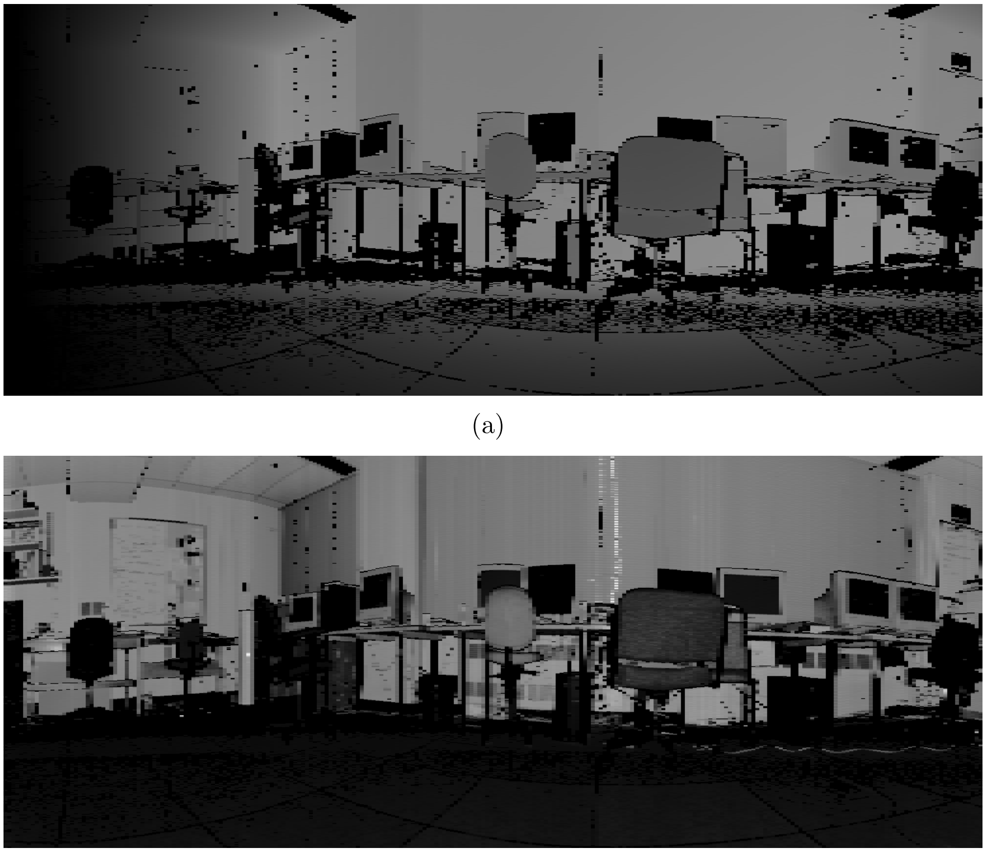
[1] Note that acquired range data (for one viewing position) provide (limited) 2.5D surface data only, and full 3D surface acquisitions can only be obtained by merging of data from several viewpoints.

any rangefinder, system calibration has taken considerable effort. Also, the shaft encoder position of the horizon and the rotation of the pan unit caused by the weight of the rangefinder required some calibration. Experiments were carried to calibrate these factors and then correction were applied in the software.

With respect to the quality of the measurements, there are many factors that influence laser reflectance, such as: target range, albedo, angle of incidence, surface roughness, and specularity. It is unlikely that all of these parameters can be determined from a limited number of reflectance measurements from an unknown obstacle. Our approach for range synthesis do not assume a particular surface type or scene illumination, thus making it more appealing and robust. The key aspect of our approach is that a reference image (sampled under the same illumination conditions as those at the time of scanning) can be used to capture the intrinsic relationships between the photometric characteristics of objects in the scene with their corresponding geometric characteristics, and use these relationships to filter out outliers and more importantly to infer missing range values.

When acquiring data from the laser rangefinder, raw data is streamed to disk and then transformed into 3D points registered into a common coordinate system. Since the mirror center of rotation and the pan unit center are aligned, we can directly project these points into a spherical grid and filter the data to eliminate outliers and produce a spherical amplitude and range image. In order to eliminate the outliers, we look at the eight nearest neighbors of the range value r_i on the spherical grid and, if at least four are within a tolerance γ from r_i, the value is considered valid, otherwise it is zeroed and then stored, to maintain proper order within the array.

Finally, a 3×3 median filter is applied over the neighboring samples to fill the missing values and create a uniform grid. One way to improve the range data quality, is to use a slower sampling rate and shorter maximum measurable range, since it can find the proper interference frequency more quickly. The payoff is, however, an additional acquisition time. Figure 7.4a shows, the 2D projection of a scaled spherical range image $\mathbf{R}$, and in the bottom the corresponding spherical amplitude image $\mathbf{A}$,



(a)

(b)

FIGURE 7.4. A 90^o-view range and amplitude image output. (a) Spherical range image. (b) Spherical amplitude image.

representing 90^o scan of our navigation environment. In our scheme for representing depth values, the close objects appear in darker shades than far objects. The total time needed to acquire this complete range data was 20 minutes. Details about this timing is given in the experimental results section (Section 5.1). From the range data, we can note that there exists areas (shown in black) with missing range values, which the laser scanner was not able to capture due to material properties (color, shininess, transparency, etc.), lighting conditions and object dimensions. In some of our experiments, a pre-processing step was needed in order to obtain dense range data as our reference of ground truth. This pre-processing step was done either by using our range synthesis algorithm or, manually, for those cases where no information at all about depth was captured. From this *new* ground truth range data we perform our sampling strategies to obtain partial range data (see Section 3.3.1).

3.2.2. The Camera

Digital cameras return photometric information (intensity and color) of a particular scene. Their projective model can be approximated using a perspective model, where the size of objects varies as a function of distance from the center of our perceived world, such that close objects are encoded at a larger representational scale that objects in the distance. The resolution of most cameras is higher than that of the laser. Furthermore, taking pictures require far less effort and they are a convincing rendition of the scene.

For the camera, we use a pinhole camera model which consists of a focal point and a view plane. Each point in the scene is mapped onto the image plane (also called view plane) by a ray passing from the focal point through the image plane and contacting an imaged point in the scene (see Figure 7.5).

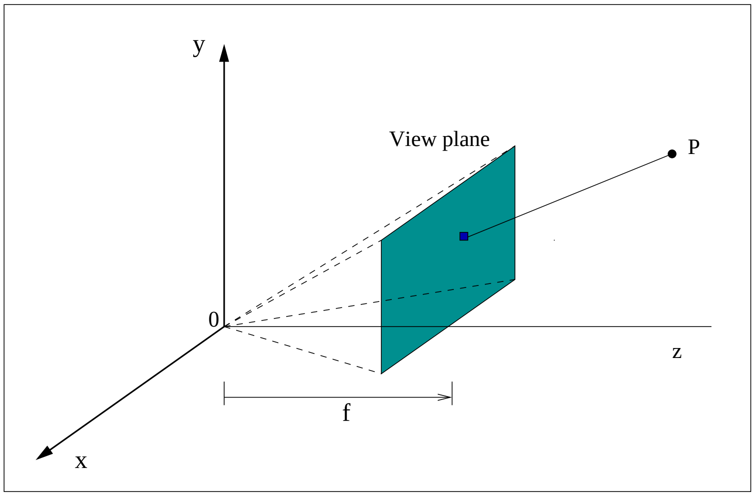


FIGURE 7.5. Camera coordinate system.

A single photograph gives us a large amount of information about the scene's appearance and structure. However, since it is a 2D image, we have lost the ability to look in different directions, to move about in the scene, to collide with its surfaces,

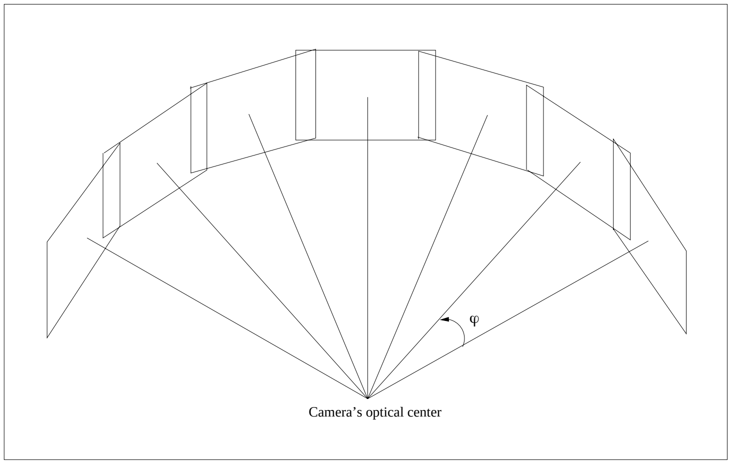


FIGURE 7.6. Mosaicking images taken from the same viewpoint but different viewing angles.

to change the illumination, and to modify the scene itself. All the above would have been possible if having a 3D model of the scene.

In attempting to build a 3D map of the environment, a mobile robot needs to gather data from different viewpoints. Since a laser rangefinder usually captures data in a wider field of view of that of a camera (from a single robot pose), by combining or ”stitching” two or more images, we yield a larger image of the same scene (see Figure 7.6). This technique is called *image mosaicking* [**97, 149, 150, 139**], and was developed long before digital computers. Image mosaics can be used for many different applications like the construction of large aerial and satellite photographs, and more recently for approximation of 3D scenes [**149, 150, 138**], video compression [**49**], architectural walkthroughs, virtual museums, and telepresence. For a complete description of image mosaics and techniques, see [**150**].

As an alternative, we could choose to use wide field of view lenses or imaging devices, such as Columbia’s OmniCam [**102, 133**], as the whole scene can be captured in a single image. However, the images acquired have substantial distortions, and the image quality is low because of mapping an entire scene into a fixed resolution video camera. Image mosaicking does not need special imaging devices and does not compromise image quality. It basically involves following three major steps:

(i) Project the input images onto a cylindrical coordinate system.

(ii) Register the warped images into a common coordinate system.

(iii) Correct the resulting small errors from the registration process.

Before projecting the images, is necessary to correct the geometric deformations caused by different types of lenses. It was already mentioned that the center of projection of the pan unit is aligned to the optical center of the camera. Therefore, images are taken from the same point of view (i.e., the camera undergoes a pure rotation around the optical center) and there is no misregistration caused by motion parallax. This mosaic is geometrically correct because the input images are related by a 2D projective transformation (homography). Camera parameters are generally not needed for image mosaics. But for cylindrical and spherical panoramas, in order to warp the images correctly, we need to know the camera's focal length. Section 3.3.2 describes the general procedure for constructing a cylindrical panorama.

Some advantages and disadvantages of the data acquisition setup need to be considered at this point. On one hand, data acquired from the sensors will correspond to scenes taken at the same height from the floor; we may take advantage of this fact when estimating the relative viewpositions of the sensors. On the other hand, even with a panoramic mosaic, some views can never be acquired, for example, the superior views of objects located at elevated positions (1.5m from the floor).

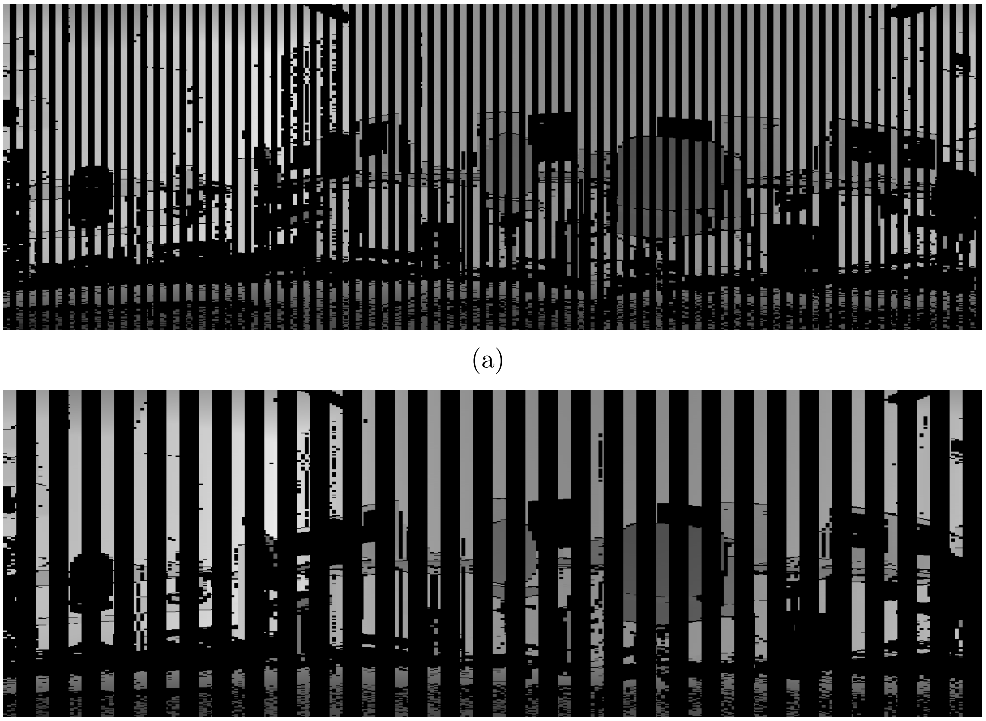
3.3. Space and Time Data Sampling

In comparing measurements from different sensors, the comparison process itself introduces uncertainty because of space and time sampling issues. Some of the sources of these errors are small variations in the sensor position during data acquisition and the type of the data taken.

We assume dense and uniformly sampled intensity images, and sparse but uniformly sampled range images. Since taking images from the camera is an effortless task, sampling of intensity images occurs more often than that of range images. The total sampling area of both intensity and range images remains fixed all the time, i.e., the area covered by the sampling data is equal at each robot pose. This area covers approximately a view of 90^o. However, the amount of range data may vary depending essentially on the sampling strategy.

3.3.1. Acquiring Partial Range Data

The configuration of the sensors also plays an important role in the sampling scheme. In our acquisition framework, the spinning mirror (y-axis) and panning motor (x-axis) combine to allow the laser to sweep out a longitude-latitude sphere. Since each step taken by the pan unit can be programmed (i.e., the stepping angle can be different at each step) we can have different sampling strategies to acquire sparse range data. Figure 7.7 shows a number of sampling strategies for the range map shown in Figure 7.4a. Black regions indicate unknown range data.


(a)

(b)

FIGURE 7.7. Examples of sampling range data. The complete range map is shown in Figure 7.4a.

In obtaining partial range data, we need to consider two key factors which will reflect the quality of the whole range map. These are: the sampling strategy used and the amount of total range acquired. The sampling strategy will depend on how far the objects/walls in the scene are from the robot. Thus, as the robot gets closer to walls/objects, the subsampling can be sparser since no much details are lost, compared to when the robot is located far away. Figure 7.8 shows this simple heuristic, which is used in collecting our experimental data. It is essentially based on the relationship between the size of the objects and the distance from where those objects are seen:

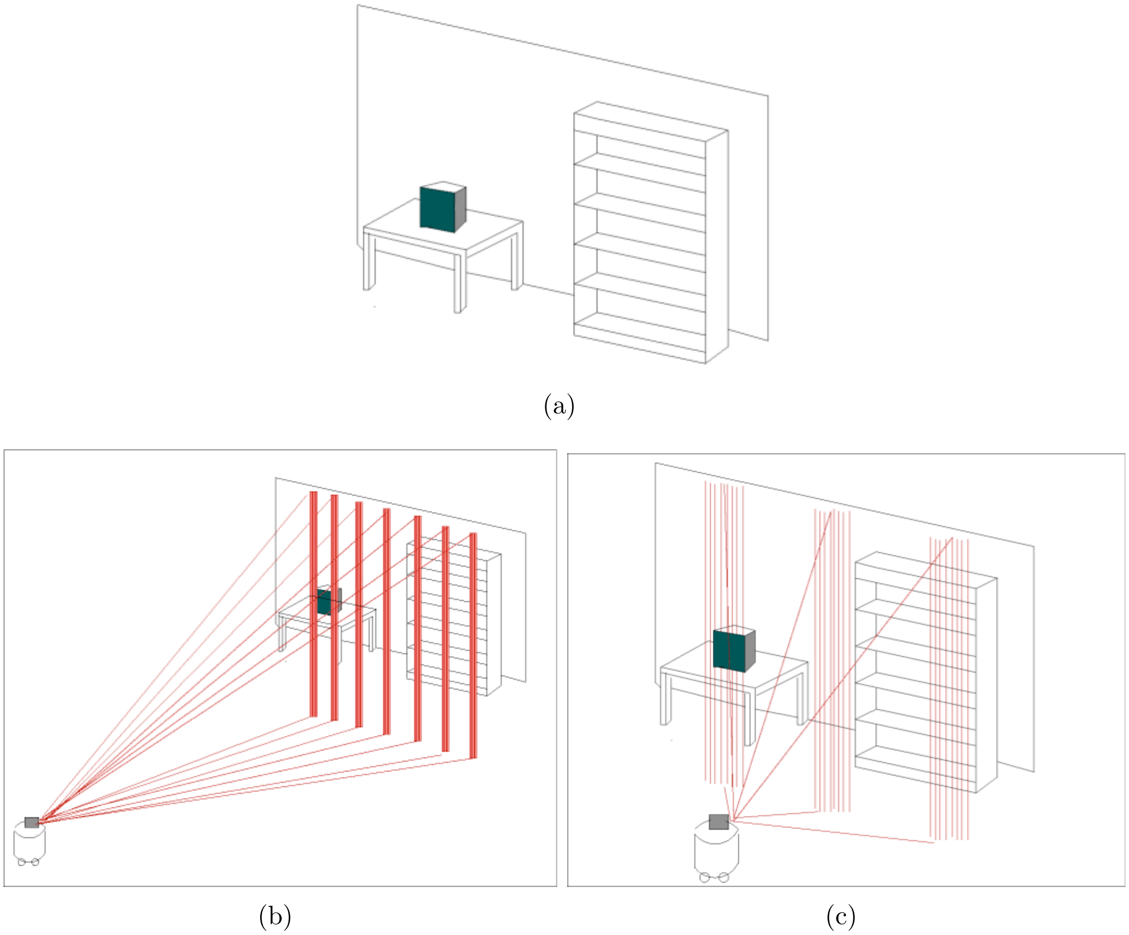


FIGURE 7.8. Sampling strategy based on distance of the scene from the robot. (a) The scene. (b) Robot is far from the objects on the scene, the subsampling is dense. (c) Robot is close to the objects, thus the subsampling is sparser.

the farther we are from an object, the smaller it looks, so dense sampling is needed to accurately represent its shape. On the other hand, the closer we are to an object, the larger it looks, so sparse subsampling is enough.

There are other strategies for sampling the range data. However, answering the question of which sampling strategy is more suitable, is not the subject of this thesis. A brief discussion on the subject and which are the possible trends are given in Chapter 9, in the section for future work.

3.3.2. Acquiring the Cylindrical Panorama Mosaic

A cylindrical panorama is created by projecting images taken from the same viewpoint, but with different viewing angles onto a cylindrical surface. Each scene point $\mathbf{P} = (x, y, z)^T$ is mapped to the cylindrical coordinate system (ψ, v) by

$$\begin{aligned} \psi &= \arctan(\frac{x}{z}), \\ v &= f \frac{y}{\sqrt{x^2 + z^2}}. \end{aligned} \tag{7.1}$$

where f is the camera's focal length. Figure 7.9 shows two warped images used in constructing a panorama.

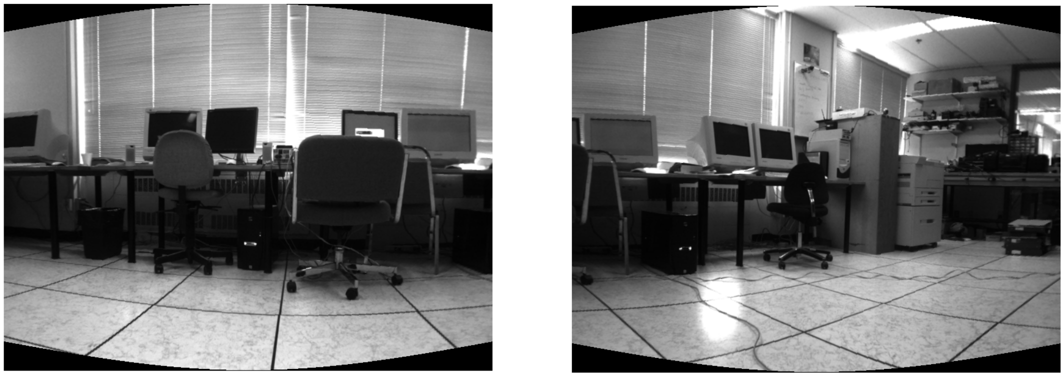


FIGURE 7.9. Two warped images used to construct the panorama mosaic.

The projected images are ”stitched” and correlated in order to precisely determine the amount of rotation between two consecutive images, this process is called *image spatial aligment* or *image registration*. In the cylindrical space, a translation becomes a rotation, so we can easily build the cylindrical image by translating each component image with respect to the previous one. In practice, there is also small vertical misalignment between images due to vertical jitter and optical twist. Therefore, both a horizontal translation t_x and a vertical translation t_y are estimated for each input image. To recover the translational motion, we estimate the incremental translation $\delta\mathbf{t} = (\delta t_x, \delta t_y)$ by minimizing the intensity error between two images,

$$E(\delta\mathbf{t}) = \sum_{\mathbf{i}} [\mathbf{I_1}(\mathbf{x'_i} + \delta\mathbf{t}) - \mathbf{I_0}(\mathbf{x_i})]^{\mathbf{2}}, \tag{7.2}$$

where $\mathbf{x}_i = (x_i, y_i)$ and $\mathbf{x}'_i = (x'_i, y'_i) = (x_i + t_x, y_i + t_y)$ are corresponding points in the two images, and $\mathbf{t} = (\mathbf{t_x}, \mathbf{t_y})$ is the global translational motion field which is the same for all pixels.

After a first order Taylor series expansion, the above equation becomes

$$E(\delta\mathbf{t}) \approx \sum_{\mathbf{i}} [\mathbf{g_i^T}\delta\mathbf{t} + \mathbf{e_i}]^{\mathbf{2}}, \tag{7.3}$$

where $e_i = I_1(x'_i) - I_0(x_i)$ is the current intensity or color error, and $g_i^T = \nabla I_1(x'_i)$ is the image gradient of I_1 at x'_i. This minimization problem has a simple least-squares solution,

$$(\sum_i g_i g_i^T)\delta\mathbf{t} = -(\sum_{\mathbf{i}} [\mathbf{e_i g_i}]). \tag{7.4}$$

The complexity of the registration lies on the amount of overlap between the images to be aligned. In our experimental apparatus, we will typically have about 20 to 30 percentage of overlap between adjacent images. However, as the panning angles at which images are taken is known, the overlap can be as small as 10 percent and still be able to align the images.

When aligning the images, corresponding points often have different intensity values for various reasons, including change in view angle of the camera, vignetting (intensity decreases towards the edge of the image), parallax effects due to unwanted motion of the optical centre, and change in scene lighting. To reduce discontinuities in intensity between images, we weight each pixel in every image proportionally to their distance to the edge of the image (i.e., it varies linearly from 1 at the centre of the image to 0 at the edge), so that intensities in the overlap area show a smooth transition between intensities in one image to intensities of the other image. A natural weighting function is the *hat function*,

$$\mathbf{w}(x,y) = \|\frac{h/2 - x}{h/2}\| - \|\frac{w/2 - y}{w/2}\| \tag{7.5}$$

where h and w are the height and the width of the image. Intuitively, this function gives less weight to the pixels along the image edges.

3.4. Camera-Laser Data Registration: Panorama with depth

To effectively use the panoramic image mosaic and the incomplete spherical range data for range synthesis, model building and rendering, both sensor inputs need to be registered with respect to a common reference frame. Note that the problem is eliminated when the data is acquired using the same sensor, as in the case of range from stereo, structured light or range and radiance images acquired by a laser rangefinder, but it becomes complex when data come from two or more different types of sensors. Their projections, resolutions and scaling properties are different due to the fact that they capture different types of data representing different physical aspects of the environment (i.e., appearance and distance). Therefore, before any meaningful high-level interpretation of the acquired images can be made, the projective model for both sensing technologies must be understood. The laser scanner and the CCD camera work with different coordinate systems and they must be adjusted one to each other. The software of most laser scanner delivers spherical coordinates, whereas the camera puts out data in a typical image projection. Further, the limited range information do not permit the use of typical methods for registration, such as feature extraction. In this work, an image-based technique was devised for registering the range and intensity data, gathered from the scans described in Sections 3.3.1 and 3.3.2, respectively. This image-based registration algorithm is similar to that presented in [**27**].

3.4.1. The Image-based Registration Algorithm

Determining the relationship between the projective models of the camera and laser rangefinder can be done through calibration, and is the basis of many geometry-based techniquesIt requires that both sensors, together with the pan unit, be fixed on the mobile robot once the relationship is calibrated. In our method, even though we also have a fixed sensor configuration, there is no calibration involved. Instead, an image-based technique is used that recovers the projective model transformation by computing a direct mapping between the points in the data sets. The accuracy of this method depends on the characteristics of the physical system and the assumptions made (e.g. affine camera or planar scene), but in general their performance is good for locally recovering a mapping between data sets.

The approach we take assumes that the optical center of the camera and the mirror center of the laser scanner are vertically aligned and that the orientation of both rotation axes coincide (see Figure 7.10). Thus, we only need to transform the panoramic camera data into the laser coordinate system. The following section describes the geometric relationship between the range and intensity data acquired

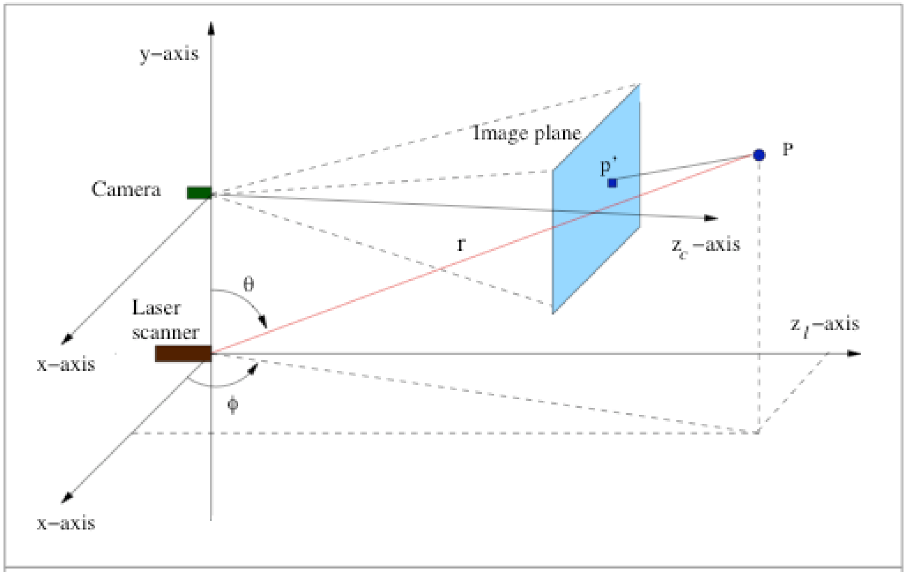


FIGURE 7.10. Camera and laser scanner orientation and world coordinate system.

from the sensors, in other words, the relation between laser points and image pixels with respect to their coordinate systems.

3.4.2. Coordinate systems

Range and intensity data are in different independent coordinate systems. To fuse both systems it is necessary to transform the data into one common reference system. We choose to use a world coordinate system. Figure 7.10 shows the mapping of a 3D point P in the scene to a 2D image point $p'(x, y)$, when observed by the camera, and onto spherical coordinates (r, θ, ϕ), when observed by the laser scanner. Calibration of the laser with respect to the camera involves determining the transformation that will map 3D points in the laser coordinate system (LCS) to 2D image points in the camera coordinate system (CCS) given a set of corresponding features in the two sensors.

The first step in the registration process is to map both type of data to a cylindrical coordinate system. After that, the two data sets are registered under a common coordinate system. In the setup presented here, the two sensors are very close to each other, and an image-to-image warp is suitable for aligning the two data sets.

Section 3.3.2 already described how the acquired intensity images are stitched together to form a cylindrical panorama. To convert the spherical range image (Section 3.2.1) to a cylindrical representation similar to that of the panoramic image mosaic, the radius of the cylindrical range image must be equal to the camera's focal length. This mapping is given by

$$\mathbf{P}(r, \theta, \phi) \mapsto \mathbf{P}(r, \phi, \frac{f}{\tan\theta}) \mapsto \mathbf{P}(r, \phi, h) \tag{7.6}$$

where r represents the distance from the center of the cylinder to the point, h is the height of the point projected on the cylinder, ϕ is the azimuth angle and f the focal length of the camera (see Fig. 7.11). Again, this data is sampled on a cylindrical grid (ϕ, h) and represented as a cylindrical image. The same procedure is applied to the amplitude data to obtain the cylindrical amplitude image.

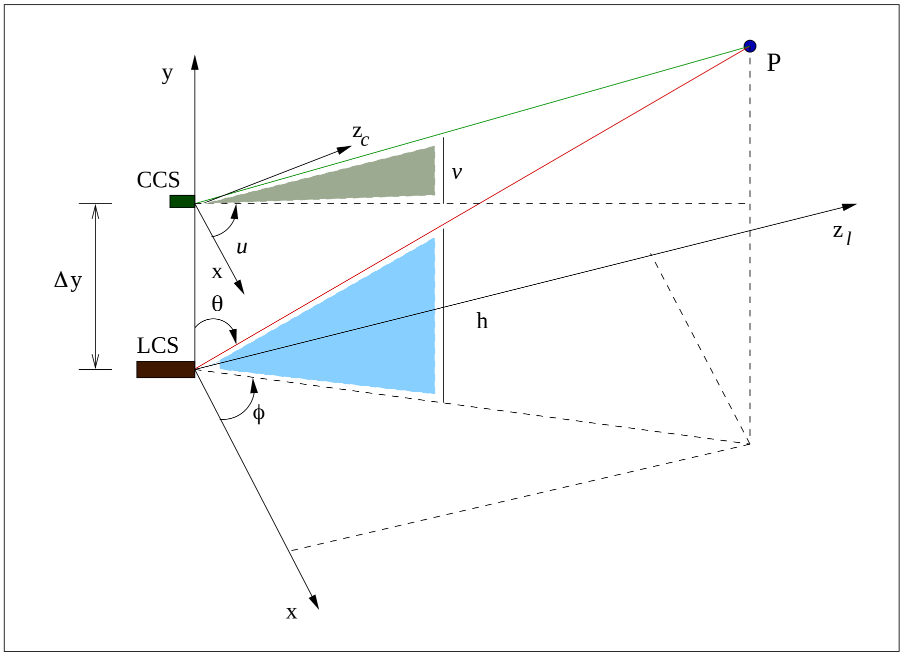


FIGURE 7.11. Projection of the 3D point P onto cylindrical coordinates: (ϕ, h) for the range data and (u, v) for the panoramic mosaic.

Once having the intensity and range data in similar cylindrical image representations, a global mapping between them is computed. The physical configuration of the sensors is approximated, as in Figure 7.11, assuming only a vertical translation ΔY and a pan rotation between the two reference coordinate systems LCS (laser coordinate system) and CCS (camera coordinate system). For a point $x_l(\phi, h)$ in the cylindrical laser image, its corresponding point in the panoramic mosaic $x_c(u, v)$ is

$$\begin{aligned} u &= a\phi + \alpha, \\ v &= f\frac{Y - \Delta Y}{r} \;=\; f\frac{Y}{r} - f\frac{\Delta Y}{r} \;=\; bh - f\frac{\Delta Y}{r} \end{aligned} \tag{7.7}$$

where a and b are two warp parameters that will account for difference in resolution between the two images, α aligns the pan rotation, and $Y = rh/\sqrt{f^2 + h^2}$ is the height of the 3D point $X(r, \phi, h)$. Since f, ΔY, and the r remain fixed through the experimental setup, the term $f\frac{\Delta Y}{r}$ can be approximated to a constant β. Thus, the general warp equations are:

$$u = a\phi + \alpha, \quad v = bh + \beta \tag{7.8}$$

The warp parameters (a, b, α, β) are computed by minimizing the sum of the squared error of two or more corresponding points [2] in the two images. In the case of a good (i.e., what we hope to be normal) initial estimate, a reasonable convergence to a nearly correct [3] solution is obtained. The initial estimate places the panorama mosaic nearly aligned with the range data, with a moderate translation or misalignment typically of about 5 to 7 pixels. To correct this, a *local alignment* is performed using the set of corresponding control points. Since these control points are typically obtained by matching the 2D edges on the intensity image to the 3D edges on the range image, after calculating the misalignment with the warp parameters, we can locally "stretch" the range data to fit the intensity data by using cubic interpolation. After that, each range data point acquired by the laser rangefinder has a corresponding pixel value in the panoramic image. Since the resolution of the depth data is sparser than the resolution of the image data, the inverse mapping cannot be computed because the function is not one-to-one.

Summarizing, the image-based registration algorithm consists of the following steps:

(i) Manually find γ corresponding points from the complete augmented panorama

(ii) Compute the global mapping parameter by using a least squares linear fit.

(iii) For each pixel (ϕ, h) in the cylindrical depth map, compute its corresponding pixel in the cylindrical panorama by using Equation 7.8.

(iv) Form a composite (augmented) image from the above warped images.

(v) Locally align the images by matching 2D edges with 3D edges.

Steps *i* and *ii* are carried for only once, thus subsequent registrations are fully automatic and there is no need for user intervention.

[2] Since the computation of the warp parameters is done only once, the actual number of corresponding points must be a large number, ideally 20, to obtain a good linear fit.

[3] The term "correct" simply means that the final solution "looks good." There is not recorded ground truth for this purpose.

4. Data Integration

In order to produce a complete 3D model or representation of a large environment, we need to integrate *dense* panoramas with depth from multiple viewpoints. Our approach is based on a hybrid method, similar to that proposed in [**10**]. This hybrid method combines two different techniques, one is for matching two 3D range scans and the other for matching intensity features on the panorama mosaic. Since dense panoramas with depth are given as an input, their integration to a common reference frame is easier than having only intensity or range data separately. To this end a simple method that combines the two previous techniques is presented.

4.1. Matching of Two Range Data Sets

The common approach to register two range data sets from different viewpoints is to derive an initial transformation by aligning a small set of corresponding feature points in the range images. These feature points are either found as local geometric features on the surface of the objects or by placing additional markers in the scene. In the former case, robustness of the feature detection is of vital importance, whereas in the latter, besides the inconvenience of taking special care in the placement of the markers, it is often infeasible to do when the environment to be digitalized must not be touched at all, e.g., cultural heritage in museums [**10**].

State-of-the-art systems often require manual specification of initial pose estimates or rely on external pose measurement systems [4], making the pre-registration a tedious and time-consuming task. Recently, methods for automatically registrating multiple range images have been proposed. One of the most popular registration methods in the literature, is the *Iterative Closest Point* (ICP) algorithm. As we use a variant of the ICP algorithm, the next section describe the ICP algorithm briefly.

[4] For instance, the relative viewpoint position might be known, e.g., from tracking the scanner position. Although direct and convenient, this is not always feasible due to the characteristics of the environment and inherent errors in the robot's odometry.

4.1.1. The ICP algorithm

The name ICP was introduced in the seminal paper by Besl and McKay [**17**], but similar ideas were developed independently at the same time by Chen and Medioni [**25**]. The ICP algorithm iteratively searches for closest point pairs in two surface patches and optimizes the transformation to minimize the distances between these points. Although the role of the two point pairs to be matched is exchangeable, we speak of data points which are to be registered with a model or model points.

Let x_i be a data point and assume that an initial estimate of the parameters is given. Then map each x_i according to these parameters into the coordinate frame of the model.
The ICP algorithm now iterates two subsequent steps:

(i) *Correspondence*: For each x_i, find the closest point $\mathbf{m}_i$ of the model.
(ii) *Estimation of new parameters*, such as the sum of squared distances between each x_i and $\mathbf{m}_i$ pair is minimized. Update the x_i according to the new parameters.

Several researchers noted that the convergence properties of this point-to-point approach is poor. The algorithm only converges toward a reasonable solution if the patches are roughly prealigned. To overcome this problem, variants of the original ICP were proposed basically by considering additional attributes, like color or surface normal, in the verification of closest point pairs. In the field of mobile robotics, Lu and Milios [**92**] estimated normals from the 2D range scans and incorporated a point-to-line metric. More sophisticated optimization schemes were proposed, as for example simulated annealing or evolutionary algorithms. Good surveys about these ICP variants can be found in [**126, 129**]. Although these measures improve the convergence properties of the original ICP algorithms and achieve high registration accuracy, they still do not allow for a registration of several completely unaligned surface patches in reasonable time.

Detecting special surface features points on the surface patches, and constraining the search for correspondences to these features, can accelarate and automatize the registration process. Feature-based approaches primarily differ in their definition of feature points and in the way they are matched. An approach presented by DePiero [**37**], detects KLT features [**93**] in range images and maintains these features together with a graph structures in a database. A common drawback of these approaches is that they rely on a sufficient number of prominent or salient features in the geometry. Especially in the presence of noise or missing values, this is often problematic.

The fact that we capture not only range data but also intensity at each robot pose, can help to alleviate the problems feature-based range matching approaches have. Intensity images are far less subject to noise. As a consequence, features points extracted from these images are more robust than those extracted from range images, making them more suitable for correspondence computation.

The idea of exploiting 2D features for 3D registration problem is not new. For instance, Roth [**127**] uses the popular Harris feature detector to extract features from an intensity image that is aligned with a range image.

4.2. Feature Detection and Matching

Extraction of geometric features from real-life range images is a difficult task as only parts of the objects' surface are visible due to occlusion and limited field-of-view. In the particular (common) situation, when the surfaces of neighboring objects are geometrically similar, distinguishing them as surfaces of independent objects is practically impossible.

On the other hand, feature detection and matching in 2D images is a well-researched topic, and algorithms robustly detecting features that are insensitive to brightness changes, scaling or local occlusions exist. In this research, we select to use the Scale Invariant Feature Transform (SIFT), developed by Lowe [**90**] (see also [**91**]) based on earlier work by Lindeberg [**88**]). This algorithm was found to perform best in a recent comparison study among several local feature descriptors [**98**]. The study compared their robustness with respect to noise, lighting and viewpoint changes up to 60 degrees.

The SIFT algorithm detects features with a scale parameter that reflects the spatial extension of its defining image neighborhood. This scale property is of vital importance for our method since it allows to robustly estimate a 3D position for each detected image feature. Figure 7.12 shows the SIFT features and their association from two panoramic mosaics to be registered.

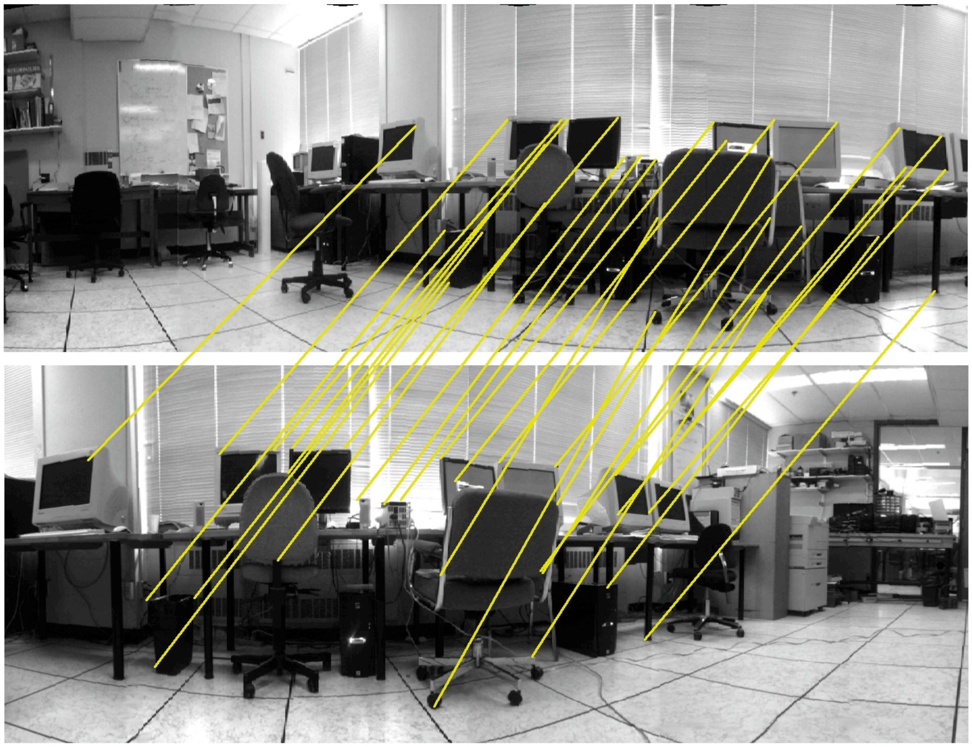


FIGURE 7.12. The SIFT features and their associations from two panoramic mosaics acquired in our lab.

We can directly derive a 3D feature position from a 2D feature using the one-to-one correspondence between pixels in the intensity image and the depth values in the range image. However, as pointed out in [**10**], this is not advisable, as the resulting 3D point is sensitive to noise and small feature deviations. Therefore, instead of using a single 3D point (the direct corresponding point to the 2D feature point) as feature, we use a set of 2D-image features with intrinsic scale information for finding corresponding points on the 3D views. These derived features are called *feature surface elements* [**10**] to accent that they are indeed a surface realization of the scale-equipped feature points [5]. Thus, a *feature point* can be defined as the center of gravity of the respective feature surface element. The set of corresponding feature points is denoted by $\mathcal{C}_{lk}$ for any pair (l, k) of range images (see Figure 7.13).

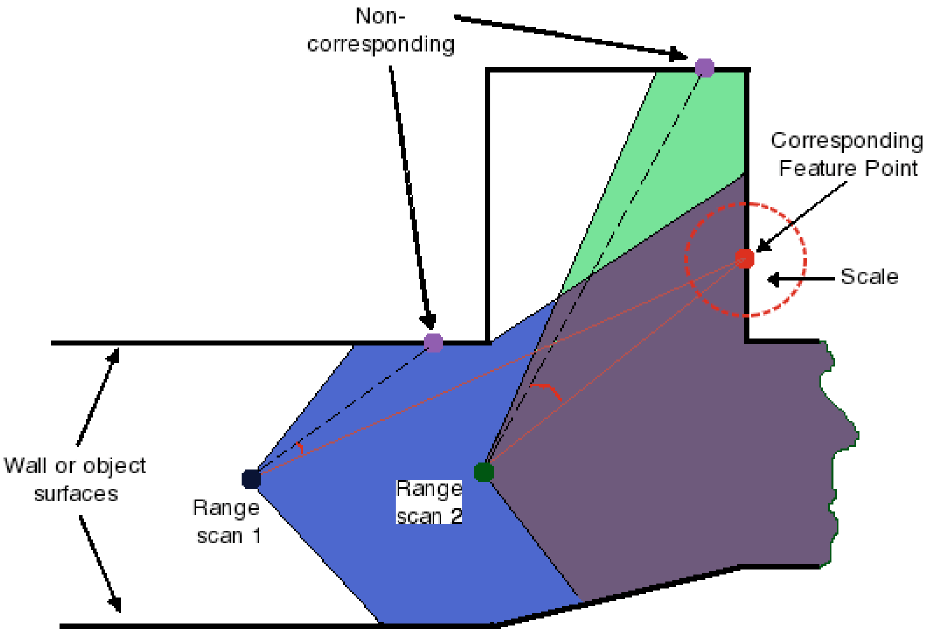


FIGURE 7.13. Two range images with matching feature point and scale. For the given corresponding feature point, only inside the scale-induced feature surface element (the dotted circle) the two range images can robustly be expected to contain corresponding parts of the scene.

The SIFT algorithm provides good matching results, however false positive matches are possible. Since the subsequent registration steps are sensitive to such false correspondences, an additional filtering to the matches based on the RANSAC method [**54**] is applied. First, the 3D positions of the features are determined by checking their

[5] The definition of these feature surface elements is taken from the notion of *surfels*, i.e., surface points equipped with normals. Surfels implicitly store a local first-order approximation of the neighboring surface. Analogously, feature surface elements represent a sampling of the neighborhood. Unlike surfels though, the feature surface elements represent a region on the surface with a well-defined size known from the 2D image features [**10**].

conformity with respect to rigid transformations. Then, the set of matching features in pair of images are validated. Since it is computationally expensive to actually compute the largest conformal set of matching features (maximum clique), the RANSAC method selects a set of three features pairs randomly and computes its support, i.e. the set of all features pairs conforming to the implied transformation. A support set is rejected if it is below a certain size α. This allows us to remove unreliable correspondences, since large sets of false, yet conforming matches are extremely improbable.

At this point, slight deviations in the 3D positions may occur because the sampling of a feature surface element in different images is usually not consistent. While such deviated features can be filtered out using the RANSAC approach to improve the registration accuracy, we tolerate these deviations to a certain extent to increase the number of conformal matches.

4.3. Hybrid 2D and 3D Feature Matching Approach

Our integration algorithm is incremental in the sense that additional 3D panoramas (i.e., panoramas with depth) can be incorporated into a set of already integrated 3D panoramas very efficiently. The feature detection is performed unilaterally (constant time), whereas the feature matching has to be done with respect to each of the intensity information in the panoramas in the given set (linear). Results from previous 3D panorama integration can nonetheless be exploited.

4.4. Pairwise Registration

Let $P^l = \{\mathbf{p}_i^l \mid i = 1, ..., n_l\}$ be a set of scale-equipped feature points of the range image l. For any pair (l, k) of range images we have a (possible empty) set of correspondences

$$C_{lk} = \{(i, j) \mid \mathbf{p}_i^l \in P^l \text{ and } \mathbf{p}_j^k \in P^k \text{ corresponding}\}. \tag{7.9}$$

In the following sections, we describe the two-stage registration procedure for a pair (l, k) with non-empty correspondence set C_{lk}.

4.4.1. Coarse registration

The coarse registration step consists of aligning the point sets P^l and P^k in a least square sense, i.e., among the set of all rigid transformations we are looking for the solution to the local minimization problem

$$T_{lk} = \underset{T}{\text{arg min}}\ \varepsilon(T \cdot l, k), \tag{7.10}$$

where the registration error ε is defined as

$$\varepsilon(l, k) = \sum_{(i,j)\in\mathcal{C}_{lk}} d^2(\mathbf{p}_i^l, \mathbf{p}_j^k). \tag{7.11}$$

This is a non-iterative procedure since correspondences are known and fixed. The result is a fast and efficient initial registration for l and k. However, the alignment based solely on the feature points accounts only for a fraction of the information available in the range images [6]. To compensate for the errors induced in the feature point computation as described in the previous section a second registration step is performed.

4.4.2. Fine registration

After the coarse registration, the pre-aligned pair of range images can be registered by applying one of the many variants of the ICP algorithm. Although, they have proven to lead to excellent registration results for good starting positions, they are, unfortunately, computationally non-trivial and susceptible to false correspondences, which might lead to slow convergence and, more importantly, to run into a local minima. Therefore, we solve these problems by restricting the domain for the correspondence computation to regions of the object that are known to correspond (from the feature detection in the 2D images, we know that the feature surface elements constitute corresponding parts of the surface).

[6] Typically, the number of feature points is in the order of dozens compared to the several hundred thousands of data points.

Thus, for all pairs $(i, j) \in \mathcal{C}_{lk}$, we find new correspondences at closest point pairs in the according sets of 3D points. These enhanced correspondence sets are then aligned using standard ICP techniques.

It is important to note, however, that the 2D feature matching procedure does not take into account the distribution of the feature points over the range images. Thus, substantial registration errors may occur in regions far from the feature surface elements for cases where the bounding box of the feature surface elements is very small compared to the bounding box of the range image itself. One solution to resolve the remaining inconsistency could be to perform a final ICP stage on the full data. However, in our experiments, the two-step registration process by feature surface elements alignment proved to be sufficient.

5. Experimental Results

In order to validate our approach for 3D environment modeling, we carried out experiments into two environments. The environments are different in size and the type of objects they contain. The first environment is a medium-size room (approximately 9.5m × 6m × 3m, which corresponds to our Mobile Robotics laboratory. It contains the usual objects in offices and labs (e.g., chairs, tables, computers, tools, windows, etc.) The second environment is larger (approximately 2m × 20m × 3m) and corresponds to the corridors of the CIM floor [7]. This environment is mostly composed of walls, doors, windows.

In the following sections, we will give details about the hardware setup, implementation and show the experimental results for the environments described above for each of the stages involved in the 3D environment modeling.

5.1. Acquiring Intensity and Range Data

The mobile robot used in our experiments is a Nomad Super Scout II, manufactured by Nomadics, Inc., retrofitted and customized for this work. It has a mobile

[7] The Center for Intelligent Machines (CIM) floor is the 4^{th} floor of the McConnell Engineering Building at McGill University.

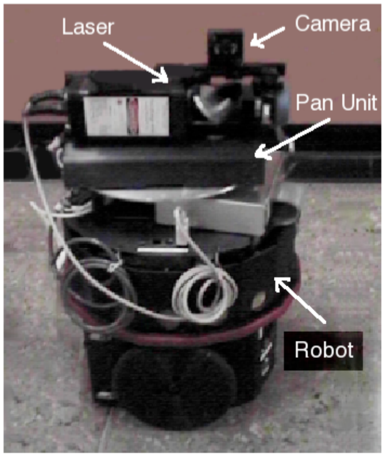


FIGURE 7.14. Our mobile robot. Nomad Super Scout II with the 2D laser range finder and the camera mounted on it.

base equipped with two driven by servo motors. Mounted on the robot are the 2D laser range finder and the CCD camera (Figure 7.14). At each robot pose, the intensity and range information is acquired and then the robot moves approximately 1m from the previous pose (this rough odometry is consider to integrate the multiple views, as explained in Section 5.5). Intensity images are acquired by using a Dragonfly camera from Point Grey Research, with a resolution of 1024 × 768 at 15 frames per second. The camera, attached to the laser, is mounted on the pan unit, allowing for panoramic image acquisition. In Figure 7.15, two intensity images acquired from the two environments are depicted. Figure 7.15(a) is an image from our lab and (b) an image from the corridors of the CIM floor.

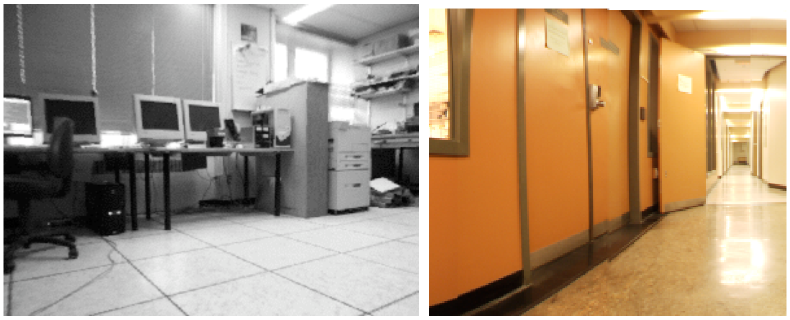
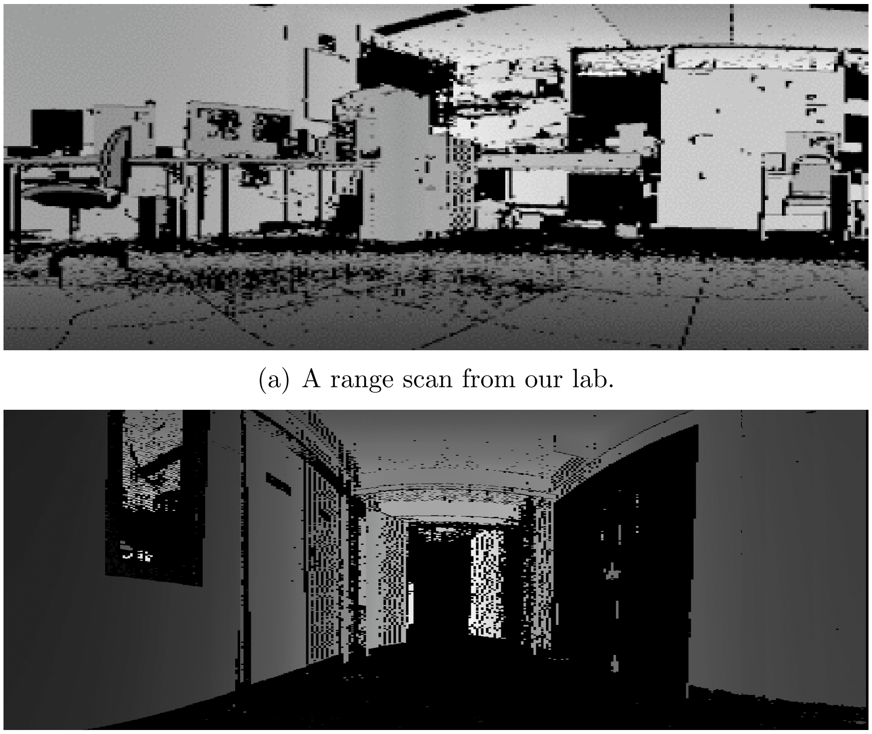

(a) An image from our lab. (b) Image from CIM floor.

FIGURE 7.15. Intensity images from the environments to model.

To acquire the range images, we use an infrared 8mW AccuRange 4000-LIR, from Accuity Research, Inc. The scanning range of the laser allows distance measurements between 0 to 15 meters at up to 50,000 range readings per second, with precision better than 1cm. A mirror, 45^o off-axis, rotates about a shaft with a 2000-position encoder to sweep out a plane. The mirror center of projection of the laser rangefinder is attached to the center of rotation of the pan unit. The pan unit was built in our lab, and has a very clean interface, making programming very simple. The panning angle (the horizontal angle) covers an area of 180^o with one step of 0.36^o. Since we are looking at having quality range values, each (2D) slice is an average of 10 samples, taking around 0.5 sec. per sample with a slower sampling resolution. The acquisition time of complete range scans covering a 180^o area (500 slices) is about 20 minutes. Figure 7.16 shows two examples of the complete range scans acquired from the two environments (black areas represent missing range values).



(a) A range scan from our lab.

(b) A range scan from the CIM floor.

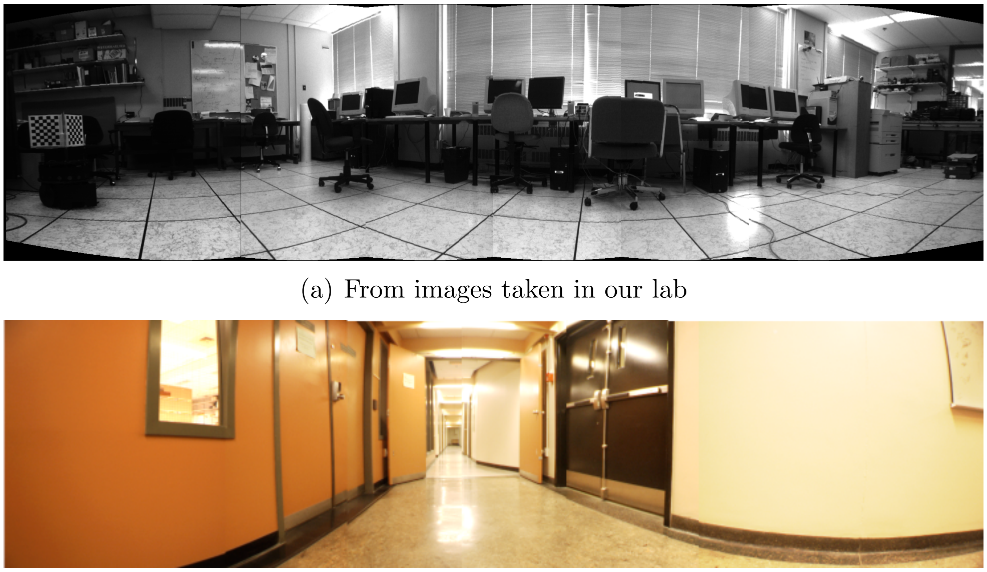
FIGURE 7.16. Complete range scans from the environments to be modeled.

5.1.1. Acquiring Partial Range

In practice, we will acquire partial range data by sampling a small amount of range data (approximately 30 to 50% of total range) at each robot pose, thus reducing significantly the acquisition time. However, in order to be able to estimate the performance of our method for estimating dense range maps, complete range scans were acquired through all the experiments presented in this chapter.

5.2. Acquiring the Panorama

As the pan unit rotates, at every 18 degrees our CCD camera takes a picture. All the images are then projected onto a cylindrical representation to obtain a cylindrical panorama mosaic. Figure 7.17 presents two 180^o cylindrical panoramas (from the respective environments) constructed using the technique described in Section 3.3.2.



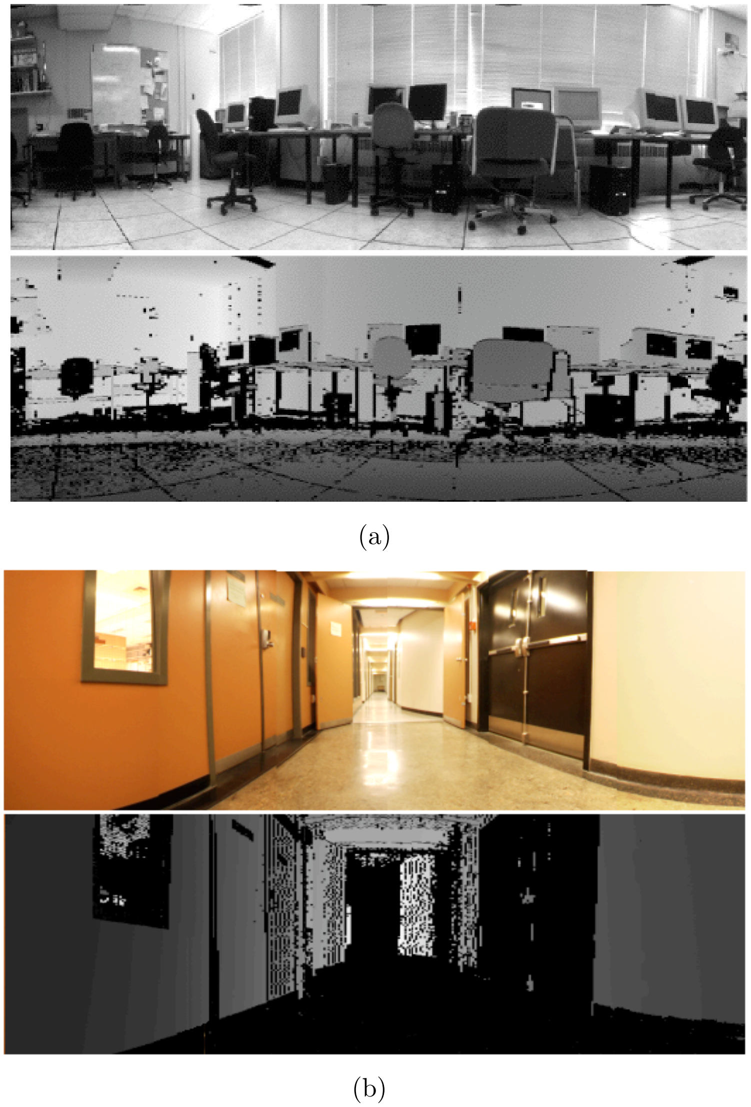
(a) From images taken in our lab

(b) From images taken in the hall

FIGURE 7.17. Two cylindrical panoramas.

5.3. Camera-Laser Data Registration

As described in Section 3.4, both the range and intensity data must be in similar cylindrical representations. For the arrangement used in these experiments, $f = 300$ pixels, $\Delta Y = 5$ cm and the range of the points is $r = 5{-}8$ meters, and β is between 6 to 10 pixel units. Figures 7.18 shows samples of the registered panorama mosaic (top) and range image (bottom). It is important to note, that the registration was



(a)

(b)

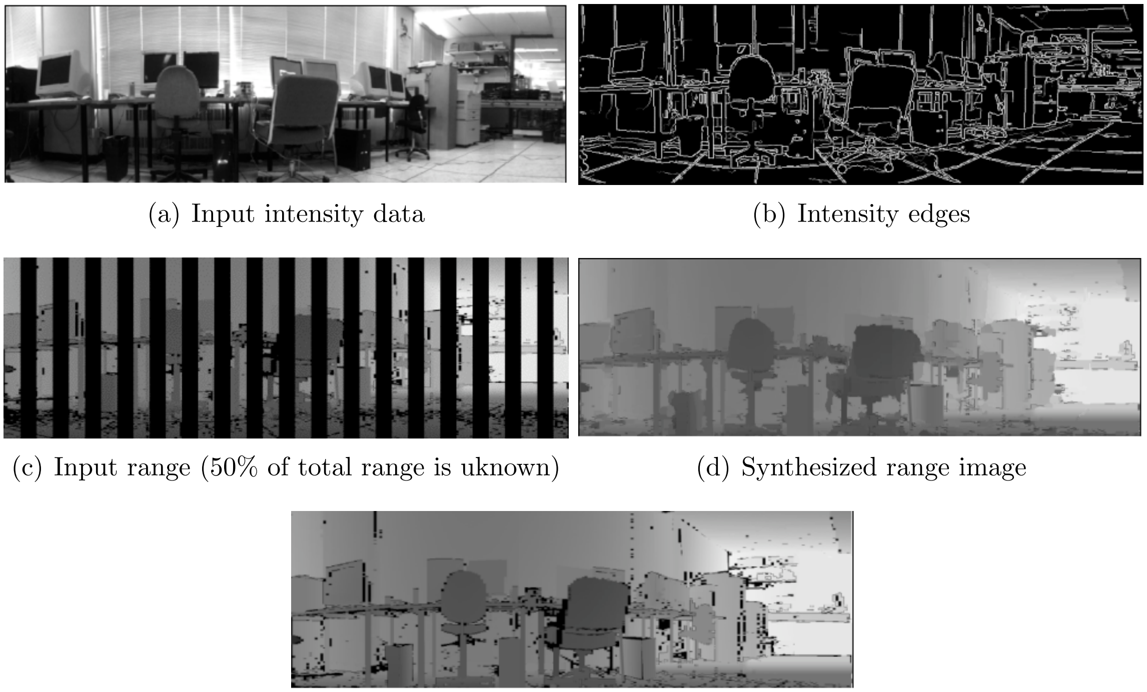
Figure 7.18. Samples of the registered intensity (top) and range data (bottom) collected (a) from our lab and (b) from the CIM floor.

computed using only the partial range data as an input. Since we are using a mapping between intensity and range image coordinates, the quality of the registration in the coarse registration step, does not depend on the amount of range data given as an input. In the fine registration step, i.e., the local alignment of 2D and 3D edges, the features captured by the input range data are crucial.

5.4. Estimating Dense Range Maps

In this section, we use our BP-based method for range synthesis (described in Section 5 of Chapter 5), so we can have *dense* panorama mosaics with depth. We present experiments with different samplings on the range data and compute our performance by comparing the synthesized results with the ground truth range data.

Figure 7.19 shows an example of data collected in our lab. (a) is the input intensity, (b) the intensity edges and (c) the input partial range data, where 50% of



(a) Input intensity data

(b) Intensity edges

(c) Input range (50% of total range is uknown)

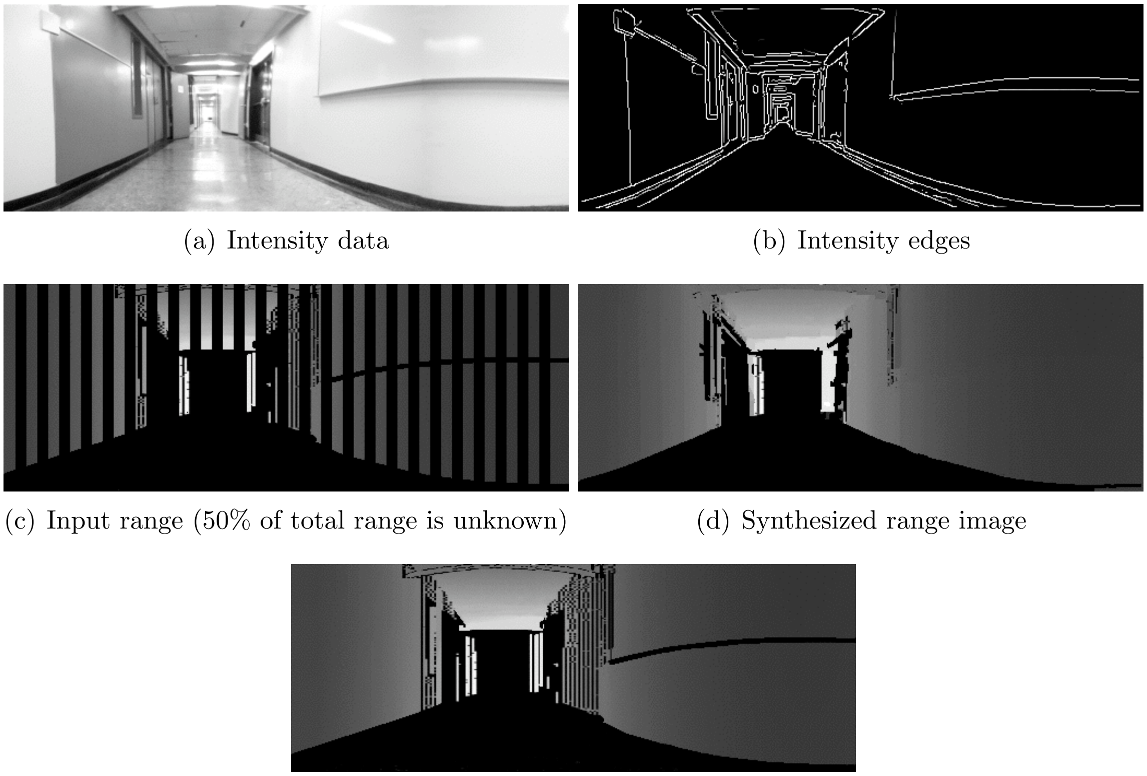
(d) Synthesized range image

(e) Ground truth range

Figure 7.19. Results on dense range map estimation. (a)-(c) Input data to our range synthesis algorithm. (b) The synthesized range image and (e) the ground truth range.

the total range is unknown. The resulted synthesized range image is shown in (d), and the ground truth range image in (e), for comparison purposes. The MAR error for this example is 7.85 centimeters.

A second example, from the data acquired on the CIM floor, is shown in Figure 7.20. (a)-(c) is the input data: the intensity image, intensity edges and the partial range, respectively. The synthesized range image after running our algorithm is shown in (d), and (e) shows the ground truth range for comparison purposes. The MAR error is 5.76 centimeters.



(a) Intensity data

(b) Intensity edges

(c) Input range (50% of total range is unknown)

(d) Synthesized range image

(e) Ground truth range

FIGURE 7.20. Results on dense range map estimation. (a)-(c) Input data to our range synthesis algorithm. (b) The synthesized range image and (e) the ground truth range.

5.5. Integration of Multiple Panoramas with Depth

In this stage, we integrate all the panoramas with depth taken at different robot poses. We first show results for the environment that corresponds to our lab. The first step is to find the corresponding 2D feature points among the intensity panoramas. Figure 7.21 shows the SIFT features and their association from two panoramas.

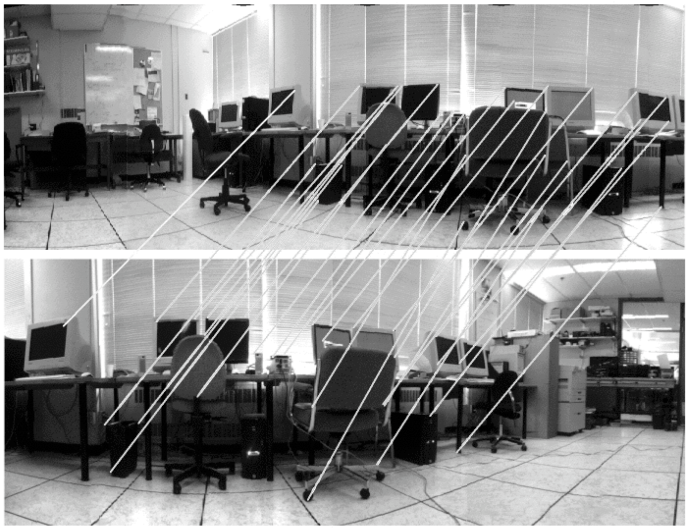
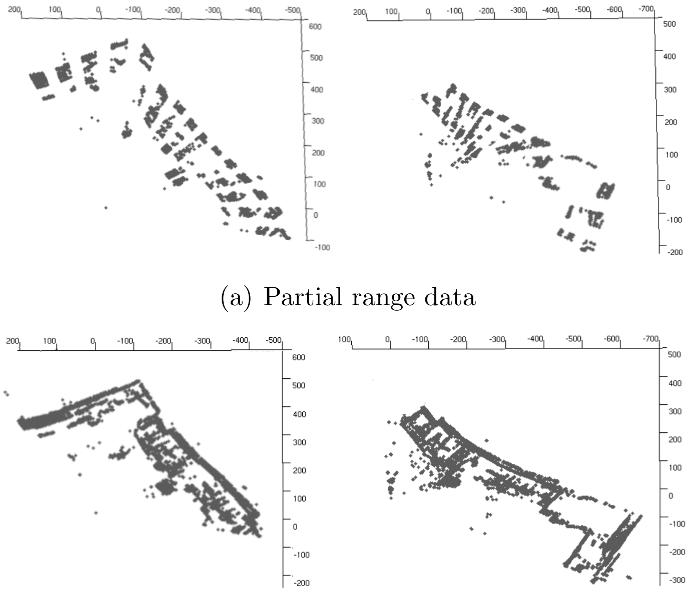


FIGURE 7.21. The SIFT features and their associations from two panoramic mosaics acquired in our lab.

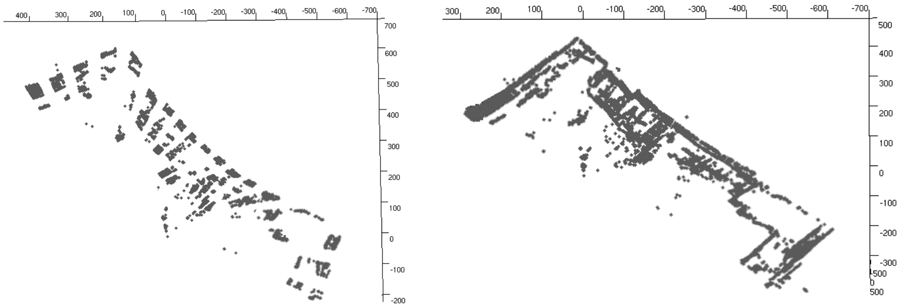
The second step is to establish a set of 3D points from these 2D feature points to define the surface feature elements, and apply the registration steps described in Section 4.4. We set a window of 5 × 5 pixels to the corresponding 3D points in each view. It is important however, to mention at this point, that using only the partial range maps to perform a global alignment may not generate good results, especially for highly unstructured environments and the lack of distinctive features, like the one we are dealing with here. The reason for this is that there may not be enough surface feature elements to perform the alignment if the 2D feature points correspond to areas with missing range. To demonstrate this, we show in Figure 7.23, the alignment of the two range scans (displayed in Figure 7.22) using only the incomplete range scans (a) and when using the synthesized dense range scans (b).



(a) Partial range data

(b) Synthesized (dense) range data

FIGURE 7.22. Top views of the two 3D range scans to be registered. (a) The input (partial) range scans and (b) the synthesized (dense) range scans.



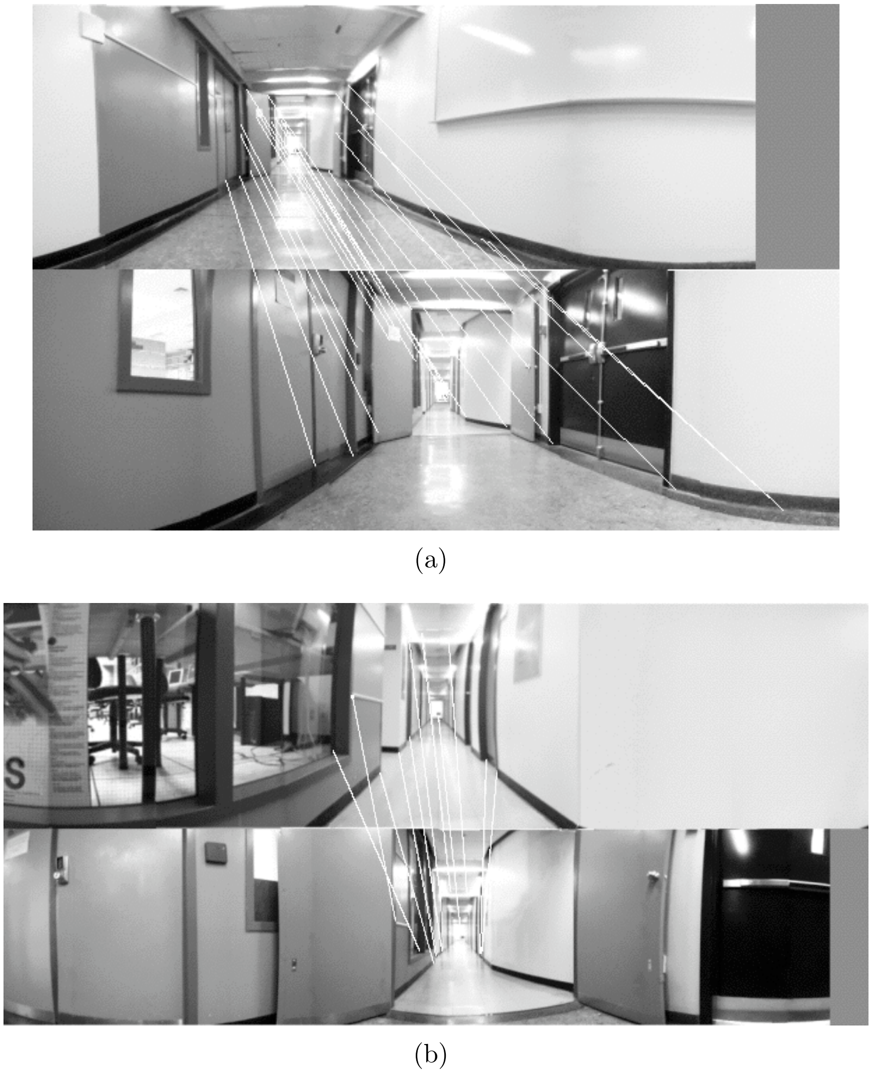
(a)

(b)

FIGURE 7.23. Top views of the alignments of the range scans of Figure 7.22, (a) when using only the partial range data and (b) using the synthesized dense range.

From the images, we observe that the alignment using only partial range data is not accurate. The total length obtained is approximately 1 meter bigger than the actual size of the room, with an averaged missalignment of individual readings of about 12 centimeters. On the other hand, when the synthesized dense range data is used together with the 2D feature points there is an error of 12 centimeters in the actual length and an average missalignment of 7 centimeters.

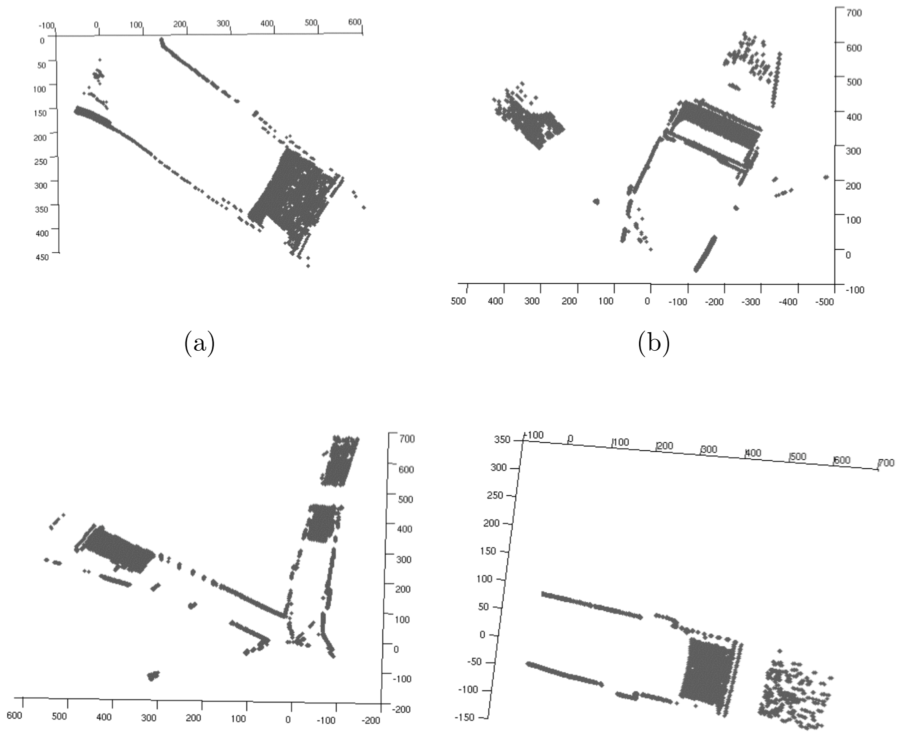
We now show results for the environment corresponding to the data acquired in the corridors of CIM. As in the previous environment, we first find the SIFT features. We show the SIFT features and their associations for two pairs of panoramas in Figure 7.24.


(a)

(b)

FIGURE 7.24. The SIFT features and their associations from two pairs of panoramic mosaics acquired in the CIM floor.

After defining the surface feature elements, we register each of the synthesized range scans. In this experiment, we set a window of 9×9 pixels to the corresponding 3D points in each view. Figure 7.25 shows four synthesized dense range scans (some of them also show parts of the ceiling). We then perform the global alignment of all the synthesized range scans incrementally, i.e., every time we integrate two range scans, we run our range synthesis method to fill in the missing range data, and continue with the next range scan. We also update real readings when available and discard the synthesized ones if they differ in more than 3 centimeters. A top view of the final aligment is displayed in Figure 7.26.



(a)

(b)

(c)

(d)

FIGURE 7.25. Top views of the four synthesized (dense) range scans taken in the CIM floor.

As we can see, the use of intensity information dramatically improves the integration of range scans. In general, the accuracy of the results depends on having a good intensity feature detector.

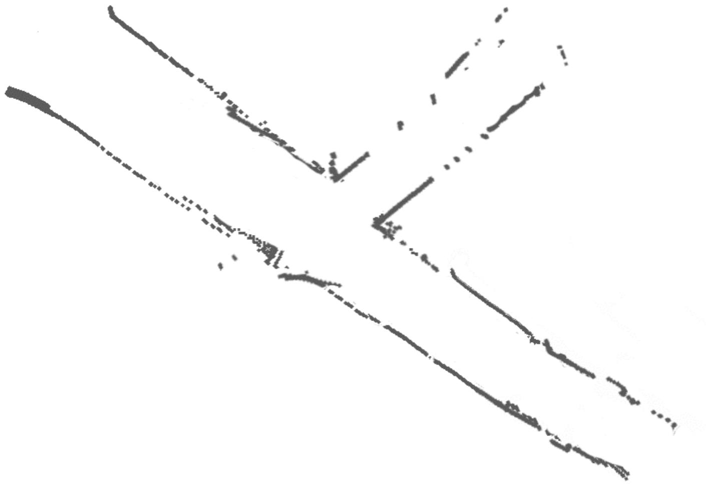


FIGURE 7.26. Top view of the global alignment of the synthesized range scans.

6. Summary

The ability to reconstruct a 3D model of an object or scene greatly depends on the type, quality and amount of information available. The data acquisition framework described in this chapter was designed to speed up the acquisition of range data by obtaining a relatively small amount of range information from the scene to be modeled. By doing so, we compromise the accuracy of our final representation. However, since we are dealing with man-made environments, we can take into account the coherence of surfaces and their causal inter-relationships with the photometric information, to estimate complete range maps from the partial range data. The experimental results shown in this chapter for two different indoor scenes demonstrate the feasibility of our method.

Since the statistical relationship between the two types of data reveal which values are to be assigned to the missing regions, the quality of the registration process, which aligns the range with the visual data, is a crucial factor in the range synthesis process.

CHAPTER 8

Color Correction of Underwater Images

In this chapter we apply the proposed method for range estimation to a different problem: color correction and augmentation, with the specific application to underwater images. Underwater images present a challenge when trying to correct the blue-green monochrome shift to bring out the color visible under full spectrum illumination in a transparent medium. For aquatic robot tasks, the quality of the images is crucial and may even be needed in real-time. Our method enhances the color of the images by a Markov Random Field (MRF) to represent the relationship between color depleted and color images. The parameters of the MRF model are learned from the training data and then the most probable color assignment for each pixel in the given color depleted image is inferred by using belief propagation (BP). This allows the system to adapt the color restoration algorithm to the current environmental conditions and also to the task requirements. Experimental results on a variety of underwater scenes demonstrate the feasibility of our method.

1. Introduction

High quality image data is desirable for many underwater inspection and observation tasks. Particularly, vision systems for aquatic robots [**20, 57, 64**] must cope with a host of geometrical distortions: colour distortions, dynamic lighting conditions and suspended particles (known as 'marine snow') that are due to inherent physical properties of the marine environment. All these distortions cause poor visibility and hinder computer vision tasks, e.g., those based on stereo triangulation or on structure from motion.

Image restoration in general, involves the correction of several types of degradation in an image. Traditionally, the most common sources of degradation are due to imperfections of the sensors, or in transmission. Underwater vision is plagued by poor visibility [**73, 67**] (even in the cleanest water). Additional factors are the ambient light, and frequency-dependent scattering and absorption, both between the camera and the environment, and also between the light source (the sun) and the local environment (i.e. this varies with both depth and local water conditions). The light undergoes scattering along the line of sight. The result is an image that is color depleted (typically appearing bluish), blurry and out of focus. In this paper, we focus on the specific problem of restoring/enhancing the color of underwater images. The term *color* refers to the red, green and blue values (often called the color channels) for each pixel in an image. Prominent blue color of clear ocean water, apart from sky reflection, is due to selective absorption by water molecules. The quality of the water determines its filtering properties. The greater the dissolved and suspended matter, the greener (or browner) the water becomes. The time of day and cloudiness of the sky also have a great effect on the nature of the light available. Another factor is depth, once at sufficient depth, no amount of simple linear filtration can effectively restore color loss. Due to the nature of underwater optics, red light diminishes when the depth increases, thus producing blue to grey like images. By 3m in depth there is almost no red light left from the sun. By 5m, orange light is gone, by 10m most yellow is also gone. By the time one reaches 25m only blue light remains [**38**]. Since many (if not all) of the above factors are constantly changing, we cannot really know all the effects of water.

Color recovery is not a simple linear transform since it depends on distance and it is also affected by sensor quantization and even light source variations. We propose a learning based Markov Random Field model for color correction based on training from examples. This allows the system to adapt the algorithm to the current environmental conditions and also to the task requirements. As proposed in [**59**], our approach is based on learning the statistics from training image pairs. Specifically, our MRF model learns the relationships between each of the color training images with its corresponding color depleted image. This model uses multi-scale representations of the color corrected (enhanced) and original images to construct a probabilistic enhancement algorithm that improves the observed video. This improvement is based on a combination of color matching correspondences from the training data, and local context via belief propagation (BP), all embodied in the Markov Random Field. Training images are small patches of regions of interest that capture the maximum of the intensity variations from the image to be restored.

2. Related Work

There are numerous image retouching programs available that have easy-to-use, semi-automated image enhancement features. But since they are directed at land-based photography, these features do not always work with underwater images. Learning to manipulate the colors in underwater images with computer editing programs requires patience. Automated methods are essential, specially for real-time applications (such as aquatic inspection). Most prior work on image enhancement tend to approximate the lighting and color processes by idealized mathematical models. Such approaches are often elegant, but may not be well suited to the particular phenomena in any specific real environment. Color restoration is an ill-posed problem since there is not enough information in the poor colored image alone to determine the original image without ambiguity. In their work, Ahlen *et al.* [**1**] estimate a diffuse attenuation coefficient for three wavelengths using known reflectance values of a reference gray target that is present on all tested images. To calculate new intensity values they use Beer's Law, where the depth parameter is derived from images that are taken at different depths. Additional parameters needed are the image enhancements functions built into the camera. In general, their results are good, but the method's efficiency depends highly on the previously noted parameters. In [**132**] a method that eliminates the backscatter effect and improves the acquisition of underwater images with very good results is presented. Their method combines a mathematical formula with a physical filter normally used for land photography. Although the method does not perform color correction, the clarity achieved on the underwater images may allow for color correction.

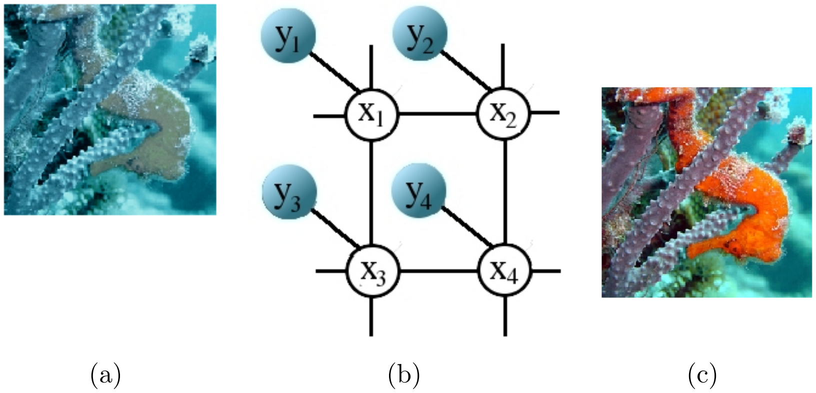
3. Our MRF-BP Approach for Color Correction

The solution of the color correction problem can be defined as the minimum of an energy function. The first idea on which our approach is based, is that an image can be modeled as a sample function of a stochastic process based on the Gibbs distribution, that is, as a Markov Random Field (MRF) [**63**]. We consider the color correction a task of assigning a color value to each pixel of the input image that best describes its surrounding structure using the training image patches. The MRF model has the ability to capture the characteristics between the training sets and then used them to learn a marginal probability distribution that is to be used on the input images. This model uses multi-scale representations of the color corrected and color depleted (bluish) images to construct a probabilistic algorithm that improves the color of underwater images. The power of our technique is evident in that only a small set of training patches is required to color correct representative examples of color depleted underwater images, even when the image contains literally no color information. Each pair of the training set is composed by a color-corrected image patch with its corresponding color-depleted image patch. Statistical relationships are learned directly from the training data, without having to consider any lighting conditions of specific nature, location or environment type that would be inappropiate to a particular underwater scene. We use a pairwise MRF model, which is of particular interest in many low-level vision problems.

3.1. The Pairwise MRF Model

Denote the input color depleted image by B $= \{b_i\}, i = 1, ..., N$, where $N \in \mathbf{Z}$ is the total number of pixels in the image and b_i is a triplet containing the RGB channels of pixel location i. We wish to estimate the color-corrected image C $=$ $\{c_i\}, i = 1, ..., N$, where c_i replaces the value of pixel b_i with a color value.

A pairwise MRF model (also known as *Markov network*) is defined as a set of hidden nodes x_i (white circles in the graph) representing local patches in the output image C, and the observable nodes y_i (shaded circles in the graph) representing local patches in the input bluish image B. Each local patch is centered to pixel location i of the respective images. Figure 8.1 shows the MRF model for color correction.



(a) (b) (c)

FIGURE 8.1. (b) Pairwise Markov Random Field used to model the joint probability distribution of the system. Observation nodes, y, represent an image patch in the bluish image (a), and hidden nodes x, an image patch in the color image (b) to be inferred.

Denoting the pairwise potentials between variables x_i and x_j by ψ_{ij} and the local evidence potentials associated with variables x_i and y_i by ϕ_i (see Figure 8.2), the joint probability of the MRF model under variable instantiation x $= (x_1, ..., x_N)$ and y $= (y_1, ..., y_N)$, can be written [**16, 63**] as:

$$P(\mathrm{x}, \mathrm{y}) = \frac{1}{Z} \prod_{ij} \psi_{ij}(x_i, x_j) \prod_{i} \phi_i(x_i, y_i), \tag{8.1}$$

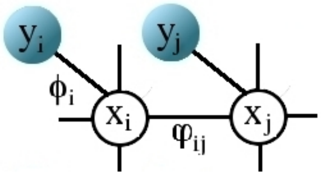


FIGURE 8.2. The potential functions ϕ and ψ define the compatibilities between nodes in the Markov network.

where Z is the normalization constant. We wish to maximize $P(\mathrm{x}, \mathrm{y})$, that is, we want to find the most likely state for all hidden nodes x_i, given all the evidence nodes y_i.

The compatibility functions allows to set high (or low) compatibilities to neighboring pixels according to the particular application. In our case, we wish to preserve discontinuities (edges) in the input (color depleted) image to avoid over smoothing the color corrected image. Thus, we set high compatibility between neighboring pixels that have similar colors, and low compatibility between neighboring pixels with abrupt change in color values. These potentials are used in messages that are propagated between the pixels to indicate what color or combination of intensities each image pixel should have.

A color pixel value in C is synthesized by estimating the maximum a posteriori (MAP) solution of the MRF model using the training set. The MAP solution of the MRF model is:

$$\mathbf{x}_{MAP} = \arg \max_{\mathbf{x}} P(\mathbf{x} \mid \mathbf{y}), \tag{8.2}$$

where

$$P(\mathbf{x} \mid \mathbf{y}) \propto P(\mathbf{y} \mid \mathbf{x}) P(\mathbf{x}) \propto \prod_i \phi_i(x_i, y_i) \prod_{(i,j)} \psi_{ij}(x_i, x_j) \tag{8.3}$$

Calculating the conditional probabilities in an explicit form to infer the exact MAP in MRF models is intractable. We cannot efficiently represent or determine all the possible combinations between pixels with its associated neighborhoods. Various techniques exist for approximating the MAP estimate, such as Markov Chain Monte Carlo (MCMC), iterated conditional modes (ICM), maximizer of posterior marginals

(MPM), etc. (refer to Chapter 4). In this work, we compute a MAP estimate, by using a learning-based framework on pairwise MRFs, as proposed by [**59**], using belief propagation (BP). The compatibility functions $\phi(x_i, y_i)$ and $\psi(x_i, x_j)$ are learned from the training set using the patch-based method in [**59**]. They are usually assumed to obey a Gaussian distribution to model Gaussian noise. The $\phi_i(x_i, y_i)$ compatibility function is defined as follows

$$\phi_i(x_i, y_i) = \mathrm{e}^{-|y_i - y_{x_i}|^2/2\sigma_i^2} \tag{8.4}$$

where x_i is a color-corrected patch candidate, y_{x_i} is the corresponding bluish patch of x_i, and y_i is the bluish patch in the input image.

The image is divided so that the corresponding color-corrected patches overlap. If the overlapping pixels of two node states match, the compatibility between those states is high. We define $\psi(x_i, x_j)$ as:

$$\psi_{ij}(x_i, x_j) = \mathrm{e}^{-d_{ij}(x_i,x_j)/2\sigma_i^2} \tag{8.5}$$

where d_{ij} is the difference between neighborhoods i and j (Section 3.3 defines the precise similarity measure we use). Images in the training set are pairs of small image regions of the bluish image with its corresponding color-corrected image, thus the compatibility functions depend on each particular input image.

3.2. MRF-MAP inference using BP

Belief propagation (BP) was originally introduced as an exact algorithm for tree-structured models [**111**], but it can also be applied for graphs with loops, in which case it becomes an approximate algorithm, leading often to good approximate and tractable solutions [**165**]. For MRFs, BP is an inference method to efficiently estimate Bayesian beliefs in the network by the way of iteratively passing messages between neighboring nodes.

The message send from node i to any of its adjacent nodes $j \in \mathrm{N}(i)$ is

$$m_{ij}(x_j) = Z \sum_{x_i} \psi(x_i, x_j) \phi(x_i, y_i) \prod_{k \in \mathrm{N}(i) \setminus \{j\}} m_{ki}(x_i) \tag{8.6}$$

where Z is the normalization constant. The maximum a posteriori scene patch for node i is:

$$x_{iMAP} = \arg \max_{\mathbf{x}_i} \phi(x_i, y_i) \prod_{j \in \mathrm{N}(i)} m_{ji}(x_i). \tag{8.7}$$

The BP algorithm is not guaranteed to converge, but if it does so, then it converges to a local stationary point of the Bethe approximation to the free energy [**171**]. In our experiments, the BP algorithm usually converges in less than 10 iterations. And it is also notable that BP is faster than many traditional inference methods.

Candidate states for each patch are taken from the training set. Fore each bluish patch in the image, we search the training set for patches that best resemble the input. The color-corrected patches corresponding the best k patches are used as possible states for the hidden nodes.

The algorithm for color correction can be summarized as follows:

(i) Divide the training images (both the bluish and color images) into small overlapping patches, which form the sets of x_i's and y_i's.

(ii) For each input patch y_i, find the k closest y_{x_i}'s. The corresponding x_i's are the candidates for that patch. Calculate the compatibility function $\phi(x_i, y_i)$ according to Eq. 8.4.

(iii) For each pair of neighboring input patches, calculate the $k \times k$ compatibility function $\psi(x_i, x_j)$ according to Eq. 8.5.

(iv) Estimate the MRF-MAP solution using BP.

(v) Assign the color value of the center pixel of each estimated maximum probability patch x_{iMAP} to the corresponding pixel in output image C.

3.3. Implementation issues

Measuring the dissimilarity between image patches is of crucial for obtaining quality results, especially when there is a prominent color (blue or green) as in underwater images. Color information can be specified, created and visualized by different color spaces (see [**170**] for more information about color spaces). For example, the *RGB* color space, can be visualized as a cube with red, green and blue axes.

Color distance is a metric of proximity between colors (e.g., Euclidean distance) measured in a color space. However, color distance does not necessarily correlate with *perceived* color similarity. Different applications have different needs which can be handled better with certain types of color spaces. For our needs it is important to be able to measure differences between colors in a way that matches perceptual similarity as good as possible. This task is simplified by the use of *perceptually uniform* color spaces.

A color space is perceptually uniform if a small change of a color will produce roughly the same change in perception anywhere in the color space. Neither RGB, HLS or CIE XYZ color spaces are perceptually uniform. We use the CIE *Lab* metric to measure the dissimilarity between image patches. Instead of red, green and blue, the Lab channels are L (luminosity or lightness), which carries the information about the darkness or lightness of each pixel, it's basically a black-and-white version of the image. All color information is carried in the other two channels. The a channel values represents the relative redness or greenness of each pixel. Shifting the curve upwards builds up reds and weakens greens. The b channel does the same for yellow versus blue. Altering the slope of these curves changes color contrast, while adjusting parts of the curve selectively changes different ranges of colors.

The (nonlinear) conversions from RGB to CIE Lab are given by: [1]

[1] Following ITU-R Recommendation BT.709, we use D_{65} as the reference white point so that $[X_n, Y_n, Z_n] = [0.9504511.088754]$ (see [**119**])

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

$$L^* = \begin{cases} 116(Y/Y_n)^{1/3} - 16 & \text{if } Y/Y_n > 0.008856 \\ 903.3(Y/Y_n) & \text{otherwise} \end{cases}$$

$$a^* = 500[f(X/X_n)^{1/3} - f(Y/Y_n)^{1/3}]$$

$$b^* = 200[f(Y/Y_n)^{1/3} - f(Z/Z_n)^{1/3}]$$

where

$$f(t) = \begin{cases} t^{1/3} & \text{if } Y/Y_n > 0.008856 \\ 7.787t + 16/116 & \text{otherwise} \end{cases}$$

We use the CIE *Lab* space which was designed such that the equal distances in the color space represent equal perceived differences in appearance. Color difference is defined as the Euclidean distance between two colors in this color space:

$$\Delta E^*_{ab} = \sqrt{(\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2} \tag{8.8}$$

where ΔL^*, Δa^*, and Δb^* are the differences between two color pixel values.

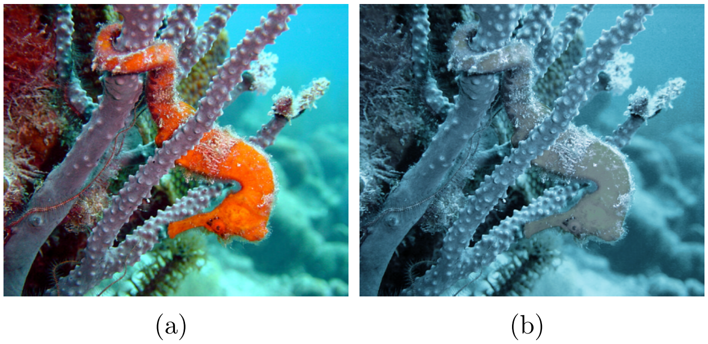
This is the similarity measure used to select possible candidates to define the compatibility functions and also to evaluate the performance of our method. Our algorithm uses a pixel-based synthesis, i.e.. one pixel (color) value c_i is estimated at a time.

4. Experimental results

We test the proposed approach in two different scenarios. In the first scenario, we use color underwater images available on the web [2] as our ground truth data. These images were taken with a professional camera and in most of the cases they were also enhanced by using a commercial software. The second scenario, involves the acquisition of underwater video by our aquatic robot. Sections 4.1 and 4.2 describe these scenarios with the experimental results.

4.1. Scenario 1

In order to simulate the effects of water, an attenuation filter were applied to each of the color underwater image. Figure 8.3a shows the ground truth (color) image and Figure 8.3b, the simulated (color depleted) image after applying the attenuation filter. Since we have ground truth information, we can compute the performance of our algorithm. The images in the training set correspond to small image regions extracted from the ground truth image and the color depleted image (see Figure 8.4). These images correspond to regions of interest in terms of the variations in pixel color values , thus the intention is that they capture the intrinsic statistical dependencies between the color depleted and ground truth pixel values. The size of the neighborhoods in all experiments were 5 × 5 pixels, and the number of possible


(a)

(b)

FIGURE 8.3. (a) The ground truth (color) image. (b) The simulated bluish image (this is the test image to be color corrected by our algorithm).

[2] http://www.pbase.com/imagine

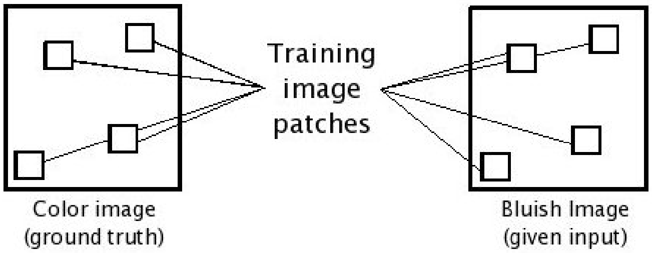


FIGURE 8.4. Diagram showing how the training image pairs are acquired for the Scenario 1.

candidates k, was fixed to be 10. Figure 8.5a shows the training image patches from where our algorithm learns the compatibility functions and Figure 8.5b shows the resulted image after running our learning-based method. The color-corrected image

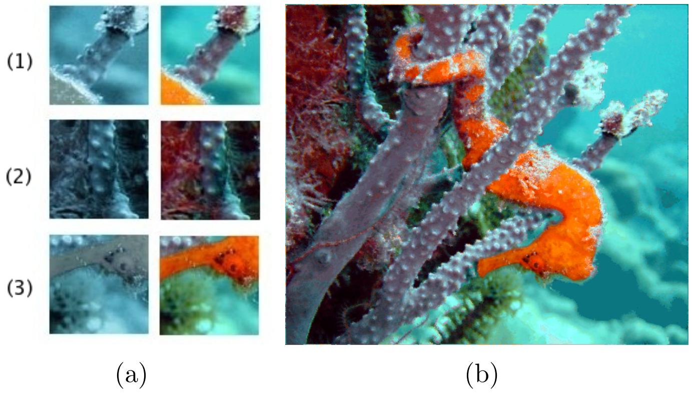


FIGURE 8.5. (a) The training image patches (labeled from (1) to (3)) used to learn the compatibility functions. (b) The color corrected image.

looks good, the discontinuities and edges are preserved since our method assign colors pixel by pixel, thus avoiding over-smoothing. Also, there are no sudden changes in color which are typically both unrealistic and perceptually unappealing. To evaluate the performance of our algorithm, we compute the mean absolute residual (MAR) error between the ground truth and the color corrected images. As mentioned in Section 3.3, the CIE Lab metric was used to calculate the similarities between pixels in the images. For this case, the MAR error is 6.5. Note that while our objective is perceptual similarity, this is difficult to evaluate and we use this objective measure to obtain quantitative performance data. For comparison purposes, we calculate the MAR error between the input (color depleted) image and the ground truth image, this is 22.03.

Using the same input image (Figure 8.5b), we now show how the final result varies depending on the training data. In Figure 8.6, we show 4 examples with alternative training sets. For example, Figure 8.6a shows a color-corrected image when using training pairs (1) and (3) (see Figure 8.5a). The MAR errors are 9.43, 9.65, 9.82, and 12.20, respectively. It can be seen that the resulting images are limited to the statistical dependencies captured by the training pairs.

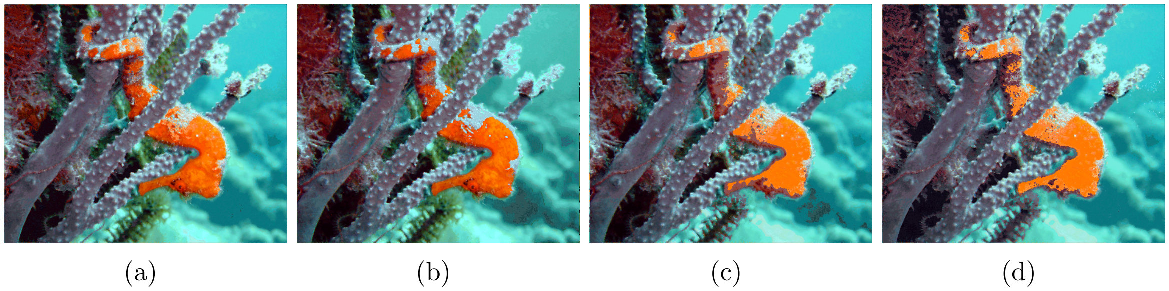


Figure 8.6. Color correction results using different training sets. The input image is shown in Figure 8.3b. The labeled training pairs are shown in Figure 8.5a. (a) Results using the training pairs (1) and (3); (b) using (2) and (3); (c) using (1) and (2), and (d) using training pair (1).

Three more examples of underwater scenes are shown in Figure 8.7. Each row shows from left to right, the ground truth color image, the input bluish image and the color corrected image after running our algorithm. The training image regions are shown by squares in the corresponding color and bluish images. In general the results look very good. For the last two examples, the size of the image patches in the training set is very small and enough to capture all the statistical dependencies between bluish and color information, as a result, the number of total comparisons in our algorithm is reduced and speed is achieved.

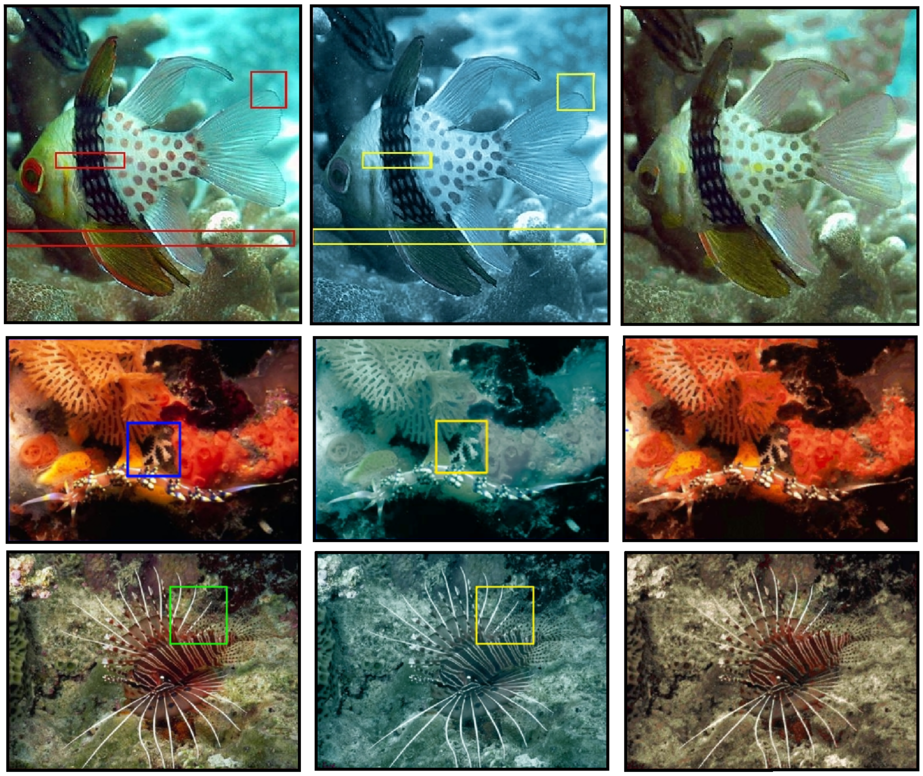
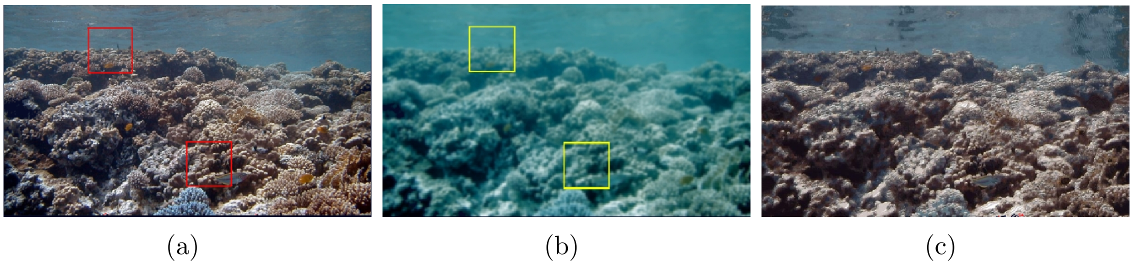


FIGURE 8.7. The training pairs are indicated by the squares in the original and input images respectively.

It was previously mentioned, that underwater images also contain some blurriness. In Figure 8.8, we show an example of applying our algorithm to a blurry and color depleted image at the same time. From left to right are, the ground truth image, the input image given to our algorithm and the color-corrected and deblurred image after running our algorithm.



(a)

(b)

(c)

FIGURE 8.8. An example of color correcting and deblurring at the same time. The training pairs are indicated by the boxes in the original (a) and input images (b) respectively. (c) is the color-corrected and deblurred image.

4.2. Scenario 2: The aquatic robot in action

As our aquatic robot [**64**] swims through the ocean, it takes video images. Figure 8.9 shows a picture of our aquatic robot in action.

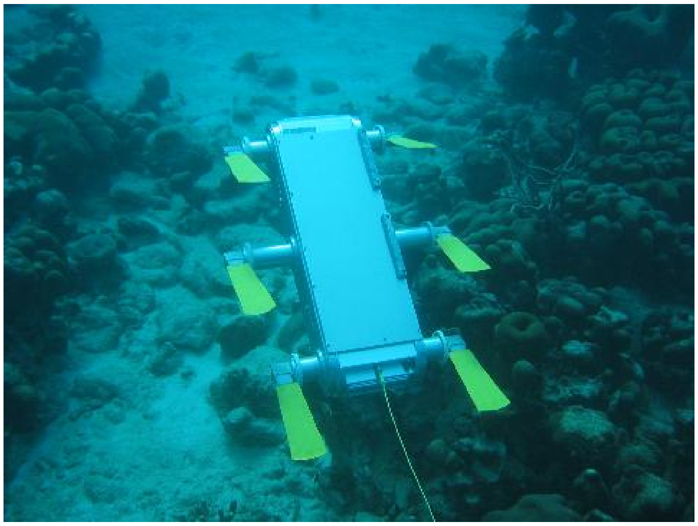


FIGURE 8.9. The aquatic robot.

In order to be able to correct the color of the images, training data from the environment that the robot is currently seeing needs to be gathered. How can better images be acquired? As light is absorbed selectively by water, not only does it get darker as you go deeper, but there is a marked shift in the light source color. In addition, there are non-uniformities in the source amplitude. Therefore, the aquatic robot needs to bring its own source of white light on it. However, due to power consumption, the light cannot be left turned on. Therefore, only at certain time intervals, the robot stops, turns its light on and take an image. These images are certainly much better, in terms of color and clarity, than the previous ones, and they can be used to train our algorithm to color correct neighboring frames (under the assumption that neighboring frames are similar). Figure 8.10 shows this scenario, here frame t_3 represents the image pair to be used to train our model for color correction.

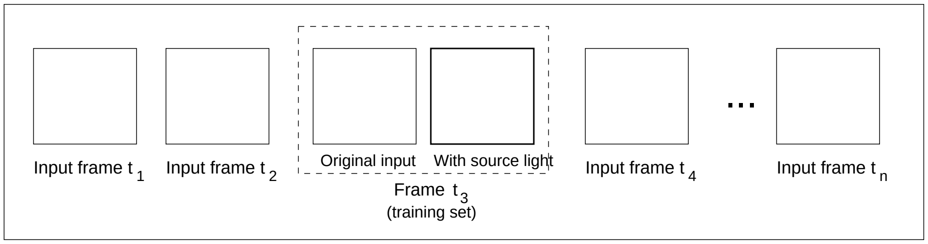
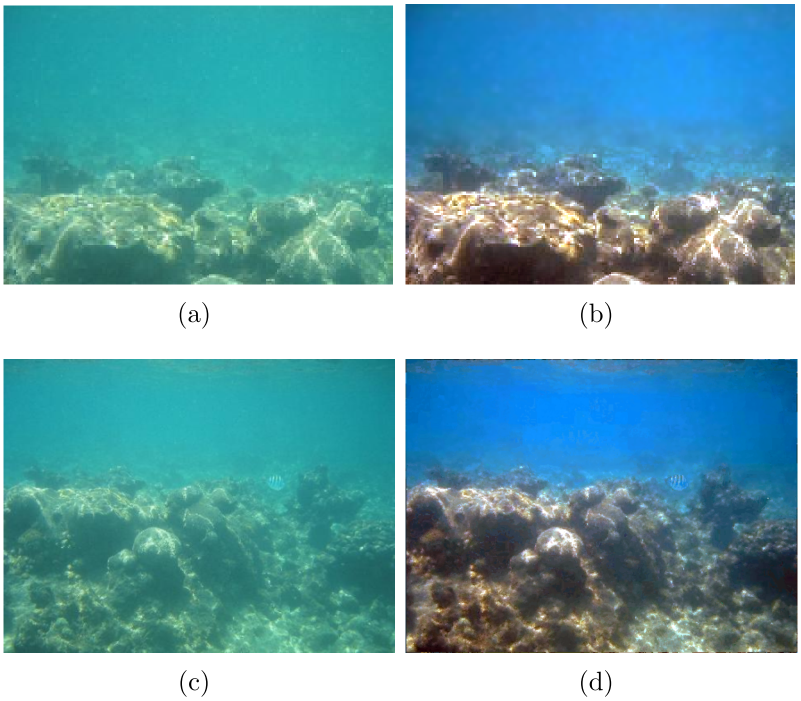


FIGURE 8.10. The scenario 2.

Now we show an example. Figures 8.11a,b show the training image pair captured at time t. The robot moves around and then at time $t + \delta$ takes an image (Figure 8.11c), which is input to our algorithm. The resulting color-corrected image is shown in Figure 8.11d. Since we do not have ground truth data for this scenario, we cannot objectively measure the performance of our algorithm, however it can be seen that the resulting image look good.


(a) (b)

(c) (d)

FIGURE 8.11. (a)-(b) The training image pair captured at frame t. (c) Image taken at frame $t + \delta$ and input to our algorithm. (d) The color corrected image.

5. Summary

Color restoration and image enhancement are ubiquitous problems. In particular, underwater images contain distortions that arise from multiple factors making them difficult to correct using simple methods. In this chapter, we show how to formulate color recovery and more general enhancement as an energy minimization problem using learned constraints. This approach's novelty lies in using a pair of images to constrain the reconstruction. There are some factors that influence the quality of the results, such as the adequate amount of reliable information as an input and the statistical consistency of the images in the training set.

Acknowledgments

We would like to thank photographer Ellen Muller for allowing the use of her underwater images for research purposes.

CHAPTER 9

Conclusions and Future Work

This thesis has considered the problem of automatically recovering the 3D structure of man-made scenes from incomplete sensor data under the context of mobile robotics. Specifically, we integrate visual information with very limited depth information. The central idea is to learn the inter-relationships between the visual information and the available depth to probabilistically infer the geometry in the missing areas. To this end, we have presented two novel statistical learning-based techniques for range synthesis. The general methodology is related to extrapolation and interpolation methods, and is based on the use of learned Markov models. Both of the presented techniques analyze the statistical relationships between intensity and range data on terms of small image patches. They differ in the way the *maximum a posteriori* (MAP) estimate is computed. The first technique, the non-parametric (NP) sampling, computes the MAP estimate by obtaining the samples directly from the near neighborhoods to the voxel to be synthesized its range value, while the second technique is an increment of the first technique that is based on using the belief propagation (BP) algorithm. For general cases, both of these techniques allow for good quality scene reconstruction in real environments. Although the second technique gives much better results when dealing with more complex environments.

In Chapter 6 we examined a variety of sampling strategies on the input range data, demonstrating the versatility of our method. From the experiments we conclude that the input range measurements are most effective if they are provided in the form of clusters of measurements scattered about the image. This form of sampling is best since it allows local statistics to be computed, but also provides boundary conditions at various locations in the image. While clumps *per se* are not available from most laser range scanners, swaths of data can be readily and efficiently extracted using standard laser scanners.

We demonstrated that when applied to more complex indoor environments, i.e., containing a variety of objects with varying surfaces and slopes, the BP-based method outperforms the NP-sampling method. The reason for this is that the the BP-based method propagates the beliefs at every location with unknown range and constraining only on those locations where a boundary is reached. This allows for smoothness in surfaces that change its slope between the sparse known range data. This approach offers a method for capturing a wide range of image structure with a sparse range data while making minimal *a priori* assumptions.

Critical to the performance of our method is the statistical similarity of the regions being reconstructed and the portions of the image used to define the Markov model. An open question is how to validate this statistical similarity which could be useful both to control and to validate the reconstruction process. A more detailed description of this problem and possible solutions are given later

We have also evaluated the performance of our reconstruction method in a mobile robotics application. In Chapter 7, we consider the mobile robot enviroment modeling problem. Specifically, the autonomous integration of incomplete sensory data to build a 3D model of an unknown large-scale environment. The main objective was to speed up the data acquisition process by obtaining a very small amount of range information from the scene to be modeled. To this end, a complete description of the physical setup and the stages involved along with their techniques was given. The experimental results on data obtained from our own building demonstrated the feasibility of our approach.

Finally, we also apply our method for range synthesis to an alternative problem. In Chapter 8, the color correction and augmentation, with the specific application to underwater images was addressed.

1. Future Work

While a broad variety of problems have been covered with respect to the automatic 3D reconstruction of unknown environments, there remain several open problems and unanswered questions. These are divided primarily between the problem of data collection and the final 3D model representation and visualization.

1.1. Optimal Range Data Collection

With respect to the data collection, a key issue in our method is the quality of the observable range data. In particular, the type of the geometric characteristics that can be extracted in relation to the objects or scene that the range data represent. If the range data do not capture the inherent geometry of the scene to be modeled, then the range synthesis process on the missing range values will be poor.

The experiments presented in this thesis were based on sampling strategies of the range data that were determined beforehand and remained fixed during the data acquisition process. Future directions to solve this problem may be to add an initial stage that optimally select the regions where the range data reflect important changes in the geometry as well as the selection among these regions of the minimum number of range data to be acquired, thus avoiding the measure of redundant range data.

The use of visual techniques and methodologies that automatically extract the most informative features related to geometry from intensity images must be investigated, as well as development of hardware tools for the acquisition of optimal range data.

The principal scientific questions posed by this research are how features that represent changes in geometry can be extracted from a small set of intensity images (the training set) and how the information from those features can be integrated into a learning model to select regions of interest in future intensity images taken from the environment. One would require that, once trained, such a model would be able to rapidly extract highly stable geometric features of the scene.

1.2. 3D Model Representation and Visualization

With respect to the final 3D model representation, which was not covered in this dissertation, an open question is that of generating a realistic, visually convincing representation of very large environments with large amounts of visual and range data. For example, it is possible to represent the environment only by the acquired raw data, but this is not efficient. First of all, not all 3D points are representative due to noise during the acquisition and the synthesis phase. Secondly, there is a great amount of redundancy in the data acquired, considering geometric structure and consistency of most environment. The last fact will result in an overload of the memory resources.

In general, the final representation must be a low-complexity 3D structural and texture model. Most of the existing approaches for 3D modeling are limited to flat surfaces, as they are mainly directed to man-made indoor environments. The geometry of those 3D models consists essentially of single planes that model the floor and walls. Visualization of these models (using a virtual reality package o language) can be used as walk-throughs. However, improving the quality of these 3D models so that they include additional details with respect to the objects in the environment is a crucial aspect. By using dense depth maps, as proposed in this thesis, these representations can be easily enriched.

The main problem addressed in this thesis (3D Environment Modeling) is central in the field of Mobile Robotics. Our approach takes advantage of acquiring only visual and very small amount of range data in order to reconstruct a 3D model of an unknown large-scale environment. The results produced by this thesis clearly demonstrate the advantage of using such approach.

REFERENCES

[1] J. Åhlén and D. Sundgren, *Bottom reflectance influence on a color correction algorithm for underwater images*, Proc. of the Scandinavian Conference on Image Analysis (SCIA), 2003, pp. 922–926.

[2] J.J. Atick, P.A. Griffin, and A.N. Redlich, *Statistical approach to shape from shading: Reconstruction of 3d face surfaces from single 2d images*, Neural Computation **8** (1996), no. 6, 1321–1340.

[3] R. Baddeley, *The correlational structure of natural images and the calibration of spatial representations*, Cognitive Science **21** (1997), 351–372.

[4] S. Baker and T. Kanade, *Limits on super-resolution and how to break them*, IEEE Trans. on Pattern Analysis and Machine Intelligence **24** (2002), no. 9, 1167–1183.

[5] T. Balch and R.C. Arkin, *Communication in reactive multiagent robotic systems*, Autonomous Robots **1** (1994), no. 1, 27–52.

[6] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, and J. Verdera, *Filling-in by joint interpolation of vector fields and gray levels*, IEEE Trans. Image Processing **10** (2001), no. 1, 8.

[7] H.G. Barrow and J.M. Tenenbaum, *Interpreting line drawings as tree-dimensional surfaces*, Artificial Intelligence **17** (1981), 75–116.

[8] P.N. Belhumeur, J.P. Hepanha, and D.J. Kriegman, *Eigenfaces vs. fisherfaces: Recognition using class specific linear projection*, IEEE Trans. Pattern Anal. Machine Intell. **19** (1997), no. 7, 711–720.

[9] A.J. Bell and T.J. Sejnowski, *The independent components of natural scenes are edge filters*, Vision Research **37** (1997), no. 23, 3327–3338.

[10] G. H. Bendels, P. Degener, R. Wahl, M. Kortgen, and R. Klein, *Image-based registration of 3d-range data using feature surface elements*, 5th International Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST) (Y. Chrysanthou, K. Cain, N. Silberman, and F. Niccolucci, eds.), December 2004, pp. 115–124.

[11] G.H. Bendels, R. Schnabel, and R. Klein, *Fragment-based surface inpainting*, Eurographics Symposium on Geometry Processing, M. Desbrun and H. Pottmann (Editors), 2005, p. 2.

[12] J.L. Bentley, *Multidimensional binary search trees used for associative searching*, Communication of the ACM **18** (1975), no. 9.

[13] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, *Image inpainting*, Proc. ACM Conf. Computer Graphics (SIGGRAPH), July 2000, pp. 417–424.

[14] M. Bertalmio, L. Vese, G. Sapiro, and S. Osher, *Simultaneous structure and texture image inpainting*, IEEE Computer Vision and Pattern Recognition (CVPR), 2003, pp. 707–712.

[15] M. Bertero, T. Poggio, and V. Torre, *Ill-posed problems in early vision*, A.I. Memo No. 924, 1986.

[16] J.E. Besag, *Spatial interaction and the statistical analysis of lattice systems*, Journal of the Royal Statistical Society, series B **36** (1974), 192–326.

[17] P.J. Besl and N.D. McKay, *A method for registration of 3-d shapes*, IEEE Trans. Pattern Anal. Mach. Intell. **14** (1992), no. 2, 239–256.

[18] F. Blais, *A review of 20 years of range sensor development*, Journal of Electronic Imaging **13** (2004), no. 1, 231–240.

[19] J. De Bonet, *Multiresolution sampling procedure for analysis and synthesis of texture images*, Proceedings of SIGGRAPH, August 1997, pp. 361–368.

[20] T. Boult, *DOVE: Dolphin omni-directional video equipment*, Proc.Int. Conf. Robotics and Automation, 2000, pp. 214–220.

[21] G.J. Burton and I.R. Moorhead, *Color and spatial structure in natural scenes*, Applied Optics **26** (1987), no. 1, 157–170.

[22] J. Canny, *A computational approach to edge detection*, IEEE Trans. Pattern Analysis and Machine Intelligence **8** (1986), no. 6, 679–698.

[23] P. Cavanagh and Y.G. Leclerc, *Shape from shadows*, Journal of Exp. Psychol. **15** (1989), no. 1, 3–27.

[24] T. Chan and J. Shen, *Mathematical models for local non-texture inpaintings*, SIAM Journal on Applied Mathematics **62** (2002), no. 3, 1019–1043.

[25] Y. Chen and G. Medioni, *Object modelling by registration of multiple range images*, Image Vision Computation, vol. 10(3), 1992, pp. 145–155.

[26] C.H. Chien, Y.B. Sim, and J.K. Aggarwal, *Generation of volume/surface octree from range data*, In IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR) (Michigan), IEEE Computer Society Press, Ann Arbor, June 1988, pp. 254–260.

[27] D. Cobzas, *Image-based models with applications in robot navigation*, Ph.D. thesis, University of Alberta, Canada, 2003.

[28] P. Comon, *Independent component analysis, a new concept?*, IEEE Signal Processing Mag. **36** (1994), no. 3.

[29] C.I. Connolly, *Cumulative generation of octree models from range data*, In Proceedings Int. Conference Robotics, March 1984, pp. 25–32.

[30] J. Coules, *effect of photometric brightness on judgments of distance*, Journal of Exp. Psychol. **50** (1955), 1925.

[31] A. Criminisi, P. Perez, and K. Toyama, *Object removal by examplar-based inpainting*, IEEE Computer Vision and Pattern Recognition (CVPR), 2003.

[32] J.L. Crowley, *World modeling and position estimation for a mobile robot using ultrasonic ranging*, In IEEE International Conference on Robotics and Automation, 1989, pp. 674–680.

[33] M. Daum and G. Dudek, *On 3-d surface reconstruction using shape from shadows*, IEEE Computer Vision and Pattern Recognition (CVPR), 1998, pp. 461–468.

[34] D. Davies, *Uncertainty and how to treat it: Modeling under uncertainty*, In Proceedings of the First International Conference on Modeling Under Uncertainty, April 1986, pp. 16–18.

[35] P.E. Debevec, C.J. Taylor, and J. Malik, *Modeling and rendering architecture from photographs: A hybrid geometry and image-based approach*, Computer Graphics **30** (1996), no. Annual Conference Series, 11–20.

[36] A.P. Dempster, N.M. Laird, and D.B. Rubin, *Maximum likelihood from incomplete data via the em algorithm*, J. Royal Statistical Society **1** (1977), 1–38.

[37] F. W. DePiero, *Deterministic surface registration at 10hz based on landmark graphs with prediction*, In 14th British Machine Vision Conf. (BMVC), 2003.

[38] J. Dera, *Marine physics*, Elsevier, 1992.

[39] R.L. DeValois and K.K. DeValois, *Spatial vision*, Oxford, 1988.

[40] A.R. Dick, P.H.S. Torr, and R. Cipolla, *Modelling and interpretation of architecture from several images*, International Journal of Computer Vision **60** (2004), 111–134.

[41] R.C. Dubes, A.K. Jain, S.G. Nadabar, and C.C. Chen, *Mrf model-based algorithms for image segmentation*, Proc. of IEEE Intl. Conf. on Pattern Recognition (ICPR), 1990, pp. 808–814.

[42] R.O. Duda and P.E Hart, *Pattern classification and scene analysis*, John Wiley & Sons, 1973.

[43] G. Dudek, M. Jenkin, C. Prahacs, A. Hogue, J. Sattar, P. Giguerre, A. German, H. Liu, S. Saunderson, A. Ripsman, S. Simhon, L. A. Torres, Milios, Zhang P. E., and I. Rekletis, *A visually guided swimming robot*, Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (Edmonton, Alberta, Canada), IEEE/RSJ, IEEE Press, August 2005.

[44] A. Efros and W.T. Freeman, *Image quilting for texture synthesis and transfer*, SIGGRAPH, 2001, pp. 1033–1038.

[45] A. Efros and T.K. Leung, *Texture synthesis by non-parametric sampling*, ICCV (2), September 1999, pp. 1033–1038.

[46] H. Egusa, *Effects of brightness, hue, and saturation on perceived depth between adjacent regions in the visual field*, Perception **12** (1983).

[47] S.F. El-Hakim, *A multi-sensor approach to creating accurate virtual environments*, Journal of Photogrammetry and Remote Sensing **53** (1998), no. 6, 379–391.

[48] A. Elfes, *Sonar-based real-world mapping and navigation*, IEEE Journal of Robotics and Automation **3** (1987), no. 3, 249–265.

[49] M.C. Lee et al., *A layered video object coding system using sprite and affine motion model.*, IEEE Transactions on Circuits and Systems for Video Technology **7** (1997), no. 1, 130–145.

[50] D. J. Field, *Scale-invariance and self-similar wavelet transforms: An analysis of natural scenes and mammalian visual systems*, Wavelets, Fractals, and Fourier Transforms (1993), 151–193.

[51] D.J. Field, *Relations between the statistics of natural images and the response properties of cortical cells*, Journal of the Optical Society of America A **4** (1987), 2379–2394.

[52] D.J. Field, *What is the goal of sensory coding?*, Neural Computation **6** (1994), 559–601.

[53] D.J. Field, *Wavelets, vision and the statistics of natural scenes*, Phil. Trans. R. Soc. A., vol. 357, 1999, pp. 2527–2542.

[54] M. Fischler and R Bolles, *Random sample consesus: A paradigm for model fitting with applications to image analysis and automated cartography*, Communication of the ACM **24** (1981), 381–395.

[55] A.W. Fitzgibbon and A. Zisserman, *Automatic 3d model acquisition and generation of new images from video sequences*, Proceedings of European Signal Processing Conference, 1998, pp. 1261–1269.

[56] R.W. Fleming, A. Torralba, R.O. Dror, and E. Adelson, *How image statistics drive shape-from-texture and shape-from-specularity*, Journal of Vision **3** (2003), no. 9, 79.

[57] G.L. Foresti, *Visual inspection of sea bottom structures by an autonomous underwater vehicle*, IEEE Trans. Syst. Man and Cyber., Part B **31** (2001), 691–795.

[58] W. T. Freeman and E. H. Adelson, *The design and use of steerable filters*, IEEE Trans. on Pattern Anal. and Mach. Intel. **13** (1991), 891–906.

[59] W.T. Freeman, E.C. Pasztor, and O.T. Carmichael, *Learning low-level vision*, International Journal of Computer Vision **20** (2000), no. 1, 25–47.

[60] W.T. Freeman and A. Torralba, *Shape recipes: scene representations that refer to the image*, Adv. in Neural Information Processing Systems 15 (NIPS) (2002), 1335–1342.

[61] P. Fua, *A parallel stereo algorithm that produces dense depth maps and preserves image feature*, Machine Vision and Applications **6** (1993), 35–49.

[62] D. Geman, *Random fields and inverse problems in imaging*, Lecture Notes in Mathematics, Springer-Verlag **1427** (1991), 113–193.

[63] S. Geman and D. Geman, *Stochastic relaxation, gibbs distributions, and the bayesian restoration of images*, IEEE Trans. on Pattern Analysis and Machine Intelligence **6** (1984), 721–741.

[64] C. Georgiades, A. German, A. Hogue, H. Liu, C. Prahacs, A. Ripsman, R. Sim, L. A. Torres-Méndez, P. Zhang, M. Buehler, G. Dudek, M. Jenkin, and E. Milios, *AQUA: an aquatic walking robot*, Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (Sendai, Japan), vol. 3, IEEE/RSJ, IEEE Press, September 2004, pp. 3525–3531.

[65] H. Gonzalez-Banos and J.C. Latombe, *Navigation strategies for exploring indoor environments*, The International Journal of Robotics Research **21** (2002), no. 10-11, 829–848.

[66] J.M. Hammersley and P. Clifford, *Markov field on finite graphs and lattices*, Unpublished, 1971.

[67] S. Harsdorf, R. Reuter, and S. Tonebon, *Contrast-enhanced optical imaging of submersible targets*, Proc. of the International Society for Optical Engineering (SPIE), vol. 3821, 1999, pp. 378–383.

[68] A. Hertzmann, C.E. Jacobs, N. Oliver, B. Curless, and D.H. Salesin, *Images analogies*, SIGGRAPH, August 2001, pp. 327–340.

[69] A. Hirani and T. Totsuka, *Combining frequency and spatial domain information for fast interactive image noise removal*, Proceedings of SIGGRAPH, 1996, pp. 269–276.

[70] B.K.P. Horn and M.J. Brooks, *Shape from shading*, MIT Press, Cambridge Mass., 1989.

[71] C.Q. Howe and D. Purves, *Range image statistics can explain the anomalous perception of length,*, Proc. Natl. Acad. Sci. U.S.A, vol. 99, 2002, pp. 13184–13188.

[72] J. Huang and D. Mumford, *Statistics of natural images and models*, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1999, pp. I: 541–547.

[73] J.S. Jaffe, *Computer modeling and the design of optimal underwater imaging systems*, IEEE J. Oceanic Engineering **15** (1990), 101–111.

[74] R.A. Jarvis, *3d shape and surface colour sensor fusion for robot vision*, Robotica **10** (1992), 389–396.

[75] A. Jepson, W. Richards, and D. Knill, *Modal structures and reliable inference*, D. Knill and W. Richards, Cambridge University Press, 1996.

[76] A.E. Johnson, R. Hoffman, J. Osborn, and M. Hebert, *A system for semi-automatic modeling of complex environments*, In Proc. of the Int. Conference on Recent Advances in 3-D Digital Imaging and Modeling (Ottawa, ON), 1997, pp. 213–220.

[77] R. Kindermann and J. L. Snell, *Markov random fields and their applications*, Series of the Contemporary Mathematics, American Mathematical Society, 1980.

[78] M. Kirby and L. Sirovich, *Application of the karhunen-loeve procedure for the characterization of human faces*, IEEE Trans. Pattern Anal. Machine Intell. **12** (1990), no. 1, 103–108.

[79] B. Kuipers., *Modelling spatial knowledge*, Cognitive Science **2** (1978), 1291–153.

[80] V. Kwatra, I. Essa, A. Bobick, and N. Kwatra, *Texture optimization for example-based synthesis*, ACM Transactions on Graphics, SIGGRAPH (2005).

[81] M.S. Langer, *Large scale failures of $f^{-\alpha}$ scaling in natural image spectra*, Journal of the Optical Society of America A. **17** (2000), no. 1, 28–33.

[82] M.S. Langer and S.W. Zucker, *Diffuse shading, visibility fields, and the geometry of ambient light*, In Proceedings of the 4th International Conference on Computer Vision (Berlin, Germany), 1993, pp. 138–147.

[83] A. Lee, K. Pedersen, and D. Mumford, *The complex statistics of high-contrast patches in natural images*, 2001, private correspondence.

[84] S.R. Lehky and T.J. Sejnowski, *Network model for shape-from-shading: neural function arises from both receptive and projective fields*, Nature **333** (1988), 452–454.

[85] A. Levin, A. Zomet, and Y. Weiss, *Learning how to inpaint from global image statistics*, Proc. of the International Conference on Computer Vision (ICCV), 2003, pp. 305–312.

[86] M. Levoy, K. Pulli, B. Curless, S. Rusinkiewicz, D. Koller, L. Pereira, M. Ginzton, S. Anderson, J. Davis, J. Ginsberg, J. Shade, and D. Fulk, *The digital michelangelo project: 3d scanning of large statues*, SIGGRAPH, July 2000.

[87] S. Z. Li, *Markov random field modeling in computer vision*, Computer Science Workbench, Editor: Tosiyasu L. Kunii, Springer, 1995.

[88] T. Lindeberg, *Detecting salient blob-like image structures and their scales with a scale-space primal sketch: a method for focus -of-attention*, Intl. Journal of Computational Vision **11** (1993), no. 3, 283–318.

[89] J.J. Little and W.E. Gillett, *Direct evidence for oclussion in stereo and motion*, Image and Vision Computing **8** (1990), 328–340.

[90] D.G. Lowe, *Object recognition from local scale-invariant features*, Proceedings of the International Conference on Computer Vision ICCV, 1999, pp. 1150–1157.

[91] D.G. Lowe, *Distinctive image features from scale-invariant keypoints*, International Journal of Computer Vision **60** (2004), no. 2, 91–110.

[92] F. Lu and E. Milios, *Robot pose estimation in unknown environments by matching 2d range scans*, Proceedings of Computer Vision and Pattern Recognition (CVPR), 1994, pp. 935–938.

[93] B. D. Lucas and T. Kanade, *An iterative image registration technique with an application to stereo vision*, In Proceedings of the DARPA Image Understanding Workshop, April 1981, pp. 121–130.

[94] P. K. Allen M. Reed and I. Stamos, *3d modeling from range imagery: An incremental method with a planning component*, Int. Conf. on Recent Advances in 3-D Digital Imaging and Modeling (Ottawa, ON), May 1997.

[95] E. MacCurdy, *The notebooks of leonardo da vinci, volume ii*, Reynal & Hitchcock, 1938.

[96] S. Mallat, *Multiresolution approximations and wavelet orthonormal basis of $l^2(r)$*, Trans. Amer. Math. Soc. **315** (1989), 69–87.

[97] L. McMillan and G. Bishop, *Plenoptic modeling: An image-based rendering system*, Computer Graphics **29** (1995), no. Annual Conference Series, 39–46.

[98] K. Mikolajczyk and C. Schmid, *A performance evaluation of local descriptors*, In Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, June 2003, pp. 257–263.

[99] D. Murray and J. J. Little, *Using real-time stereo vision for mobile robot navigation*, Autonomous Robots **8** (2000), no. 2, 161–171.

[100] D.G. Myers, *Psychology*, Worth, 1995.

[101] M.Hatzitheodorou, *The derivation of 3-d surface shape from shadows*, In Proc. Image Understanding Workshop, 1989, pp. 1012–1020.

[102] S.K. Nayar and A. Karmarkar, *360 x 360 mosaics*, Proc. of IEEE Conference on Computer Vision and Pattern Recognition (South Carolina,USA), June 2000, p. 8.

[103] W. Nelson and I. Cox, *Local path control for an autonomous vehicle*, In IEEE International Conference on Robotics and Automation (New York, NY), 1988, pp. 1504–1510.

[104] L. Nyland, D. McAllister, V. Popescu, C. McCue, A. Lastra, P. Rademacher, M. Oliveira, G. Bishop, G. Meenakshisundaram, M. Cutts, and H. Fuchs, *The*

impact of dense range data on computer graphics, Proceedings of Multi-View Modeling and Analysis Workshop (MVIEW part of CVPR), June 1999, p. 8.

[105] J. Oliensis, *A critique of structure from motion algorithms*, Tech. report, NEC Research Institute, 1987.

[106] A. Oliva and A. Torralba, *Modeling the shape of the scene: a holistic representation of the spatial envelope*, International Journal of Computer Vision **42** (2001), 145–175.

[107] B.A. Olshausen and D.J. Field, *Sparse coding on the statistical analysis of dirty picture with an overcomplete basis set: a strategy employed by v1?*, Vision Res. **37** (1997), 3311–3325.

[108] R. Paget and D. Longstaff, *Texture synthesis via noncausal nonparametric multiscale markov random field*, IEEE Trans. on Pattern Anal. and Mach. Intell. **7** (1998), 925–931.

[109] P. Pakzad and V. Anantharam, *Belief propagation and statistical physics*, Proc. Conf. Inform. Sciences and Systems, March 2002.

[110] S.E. Palmer, *Vision science*, MIT Press, Cambridge Mass., 1999.

[111] J. Pearl, *Probabilistic reasoning in intelligent systems: Networks of plausible inference*, Morgan Kaufmann Publishers, Inc., 1988.

[112] S.M. Pizer, E.P. Amburn, J.D. Austin, R. Cromartie, A. Geselowitz, B.M. ter Haar Romeny, and J.B. Zimmerman, *Adaptive histogram equalization and its variations*, Comp. Vision, Graphics, and Image Processing. CVGIP-35, 1987, pp. 355–368.

[113] T. A. Poggio, E.B. Gamble, and J.J. Little, *Parallel integration of vision modules*, Science **242** (1988), 436–439.

[114] M. Pollefeys, L. Van Gool, M. Vergauwen, F. Verbies, K. Cornelis, and J. Tops, *Visual modeling with a hand-held camera*, International Journal of Computer Vision **59** (2004), 207–232.

[115] M. Pollefeys, R. Koch, M. Vergauwen, and L. Van Gool, *Metric 3d surface reconstruction from uncalibrated images sequences*, Proceedings of SMILE Workshop (post-ECCV), 1998, pp. 138–153.

[116] J. Portilla and E. P. Simoncelli, *A parametric texture model based on joint statistics of complex wavelet coefficients*, International Journal of Computer Vision **40** (2000), no. 1, 49–71.

[117] B. Potetz and T. S. Lee, *Statistical correlations between 2d images and 3d structures in natural scenes*, Journal of Optical Society of America **20** (2003), no. 7, 1292–1303.

[118] B. Potetz and T.S. Lee, *Scaling in natural scenes and the inference on 3d shape*, In Advances in Neural Information Processing Systems (NIPS), 2004.

[119] C. Poynton, *A technical introduction to digital video*, Wiley, NewYork, 1996.

[120] C.J. Preston, *Gibbs states on countable sets*, Cambridge Tracks in Mathematics, Cambridge University Press., 1974.

[121] K. Pulli, M. Cohen, T. Duchamp, H. Hoppe, J. McDonald, L. Shapiro, and W. Stuetzle, *Surface modeling and display from range and color data*, Lecture Notes in Computer Science 1310 (1997), 385–397.

[122] V.S. Ramachandran, *Perception of shape from shading*, Nature **331** (1988), 163–166.

[123] Anand Rangarajan, Baba Vemuri, and Alan Yuille (eds.), *Color correction of underwater images for aquatic robot inspection*, Lecture Notes in Computer Science, Springer, november 2005.

[124] E. Reinhard, P. Shirley, M. Ashikhmin, and T. Troscianko, *Second order image statistics in computer graphics*, ACM Symposium on Applied Perception in Computer Graphics and Visualization, 2004, pp. 99–106.

[125] Ioannis M. Rekleitis, Gregory Dudek, and Evangelos Milios, *Multi-robot cooperative localization: A study of trade-offs between efficiency and accuracy*, IEEE/RSJ/ International Conference on Intelligent Robots and Systems (Lausanne, Switzerland), IEEE/RSJ, October 2002, pp. 2690–2695.

[126] M. Rodrigues, R. Fisher, and Y. Liu, *Special issue on registration and fusion of range images*, Computational Vision and Image Understanding, vol. 87, 1/2/3, 2002, pp. 1–7.

[127] G. Roth, *Registering two overlapping range images*, Proceedings of the Second International Conference on 3-D Digital Imaging and Modeling, 1999, pp. 191–200.

[128] D.L. Ruderman and W. Bialek, *Statistics of natural images: scaling in the woods*, Phys. Rev. Letters **73** (1994), 814–817.

[129] S. Rusinkiewicz and M. Levoy, *Efficient variants of the icp algorithm*, Proceedings of the Third International Conference on 3D Digital Imaging and Modeling, 2001, pp. 145–152.

[130] A. Van Der Schaaf, *Natural image statistics and visual processing*, Ph.D. thesis, Rijksuniversiteit Groningen, 1998.

[131] D. Scharstein and R. Szeliski, *High-accuracy stereo depth maps using structured light*, IEEE CVPR, vol. 1, 2003, pp. I: 195–202.

[132] Y.Y. Schechner and N. Karpel, *Clear underwater vision*, Proc. of Intt. Conference of Vision and Pattern Recognition, vol. 1, 2004, pp. 536–543.

[133] Y.Y. Schechner and S. K. Nayar, *Generalized mosaicing*, Proc. of IEEE Intl. Conference on Computer Vision (Vancouver, Canada), July 2001.

[134] V. Sequeira, K. Ng, E. Wolfart, J.G.M. Goncalves, and D.C. Hogg, *Automated reconstruction of 3d models from real environments*, ISPRS Journal of Photogrammetry and Remote Sensing **54** (1999), 1–22.

[135] G. Shafer, *Mathematical theory of evidence*, Princeton University Press, Princeton, NJ, 1976.

[136] J. Shen, *Inpainting and the fundamental problem of image processing*, SIAM news, 2003.

[137] I. Shimshoni, Y. Moses, and M Lindenbaum, *Shape reconstruction of 3d bilaterally symmetric surfaces*, International Journal of Computer Vision **2** (2000), 1–15.

[138] H. Shum and R. Szeliski, *Panoramic image mosaics*, Tech. Report MSR-TR-97-23, Microsoft Research, 1997.

[139] H. Shum and R. Szeliski, *Systems and experiment paper: Construction of panoramic image mosaics with global and local alignment*, International Journal of Computer Vision **36** (2000), no. 2, 101–130.

[140] R. Siegwart and I.R. Nourbakhsh, *Introduction to autonomous mobile robots*, MIT Press, 2004.

[141] G. Soucy and F. P. Ferrie, *Motion and surface recovery using curvature and motion consistency*, Tech. report, Center for Intelligent Machines, McGill University, Montréal, Québec, Canada, April 1995.

[142] F. Spitzer, *Random fields and interacting particle systems*, M.A.A. Summer Seminar Notes, 1971.

[143] A. Srivastava, A.B. Lee, E.P. Simoncelli, and S.C. Zhu, *On advances in statistical modeling of natural images*, Journal of the Optical Society of America **53** (2003), no. 3, 375–385.

[144] I. Stamos and P.K. Allen, *3d model construction using range and image data*, CVPR, June 2000.

[145] Student, *The elimination of spurious correlation due to position in time*, Biometrika **10** (1914), 179–180.

[146] J. Sun and P. Perona, *Preattentive perception of elementary three dimensional shapes*, Vision Res. **36** (1996), 2515–2529.

[147] B.J. Super and A.C. Bovik, *Shape from texture using local spectral moments*, IEEE Trans. Pattern Analysis and Machine Intelligence **17** (1995), no. 4, 333–343.

[148] E. Switkes, M.J. Mayer, and J.A. Sloan, *Spatial frequency analysis of the visual environment: anisotropy and the carpentered environment hypothesis*, Vis. Res., vol. 18, 1978, pp. 1393–1339.

[149] R. Szeliski, *Video mosaics for virtual environments*, IEEE Computer Graphics and Applications, March 1996, pp. 22–30.

[150] R. Szeliski and H. Shum, *Creating full view panoramic image mosaics and environment maps*, Computer Graphics **31** (1997), no. Annual Conference Series, 251–258.

[151] I.L. Taylor and F.C. Sumner, *Actual brightness and distance of individual colors when their apparent distance is held constant*, Journal of Psychol. **19** (1945), 7985.

[152] C. Tomasi and T. Kanade, *Shape and motion from image streams under orthography: A factorization approach*, International Journal of Computer Vision **9** (1992), no. 2, 137–154.

[153] A. Torralba and W.T. Freeman, *Properties and applications of shape recipes*, IEEE Conference of Vision and Pattern Recognition (CVPR), 2003, pp. 383–390.

[154] A. Torralba and A. Oliva, *Depth estimation from image structure*, IEEE Trans. Pattern Analysis and Machine Intelligence **24** (2002), no. 9, 1226–1238.

[155] A. Torralba and A. Oliva, *Statistics of natural images categories*, Computation in Neural Systems, 2003, pp. 391–412.

[156] L. A. Torres-Méndez and G. Dudek, *A statistical learning method for mobile robot environment modeling*, Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI) Workshop on Reasoning with Uncertainty in Robotics (RUR) (Acapulco, Mexico), Morgan Kaufmann, August 2003, pp. 85–92.

[157] L. A. Torres-Méndez and G. Dudek, *Statistical inference and synthesis in the image domain for mobile robot environment modeling*, Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems (IROS), IEEE Press, September 2004.

[158] L.A. Torres-Méndez and G. Dudek, *Range synthesis for 3d environment modeling*, IEEE Workshop on Applications of Computer Vision (Orlando, FL), 2002, pp. 231–236.

[159] L.A. Torres-Méndez and G. Dudek, *Statistics in the image domain for mobile robot environment modeling*, Proceedings of 4th the International Symposium of Robotics and Automation (ISRA) (Queretaro, Mexico), August 2004, pp. 699–706.

[160] L.A. Torres-Méndez, G. Dudek, and P. Di Marco, *Inter-image statistics for scene reconstruction*, Proceedings of the 1st. Canadian Conference on Computer and Robot Vision (CRV) (London, Ontario), May 2004, pp. 432–439.

[161] S. Ullman, *Computational studies on the interpretation of structure and motion: Summary and extension*, Artificial Intelligence Memo, 1983.

[162] H. van Hateren and A. van der Schaaf, *Independent component filters of natural images compared with simple cells in primary visual cortex*, Proc. R. Soc. London Ser. B, vol. 265, 1998, p. 359366.

[163] J. Verdera, V. Caselles, M. Bertalmo, and G. Sapiro, *Inpainting surface holes*, IEEE International Conference on Image Processing (ICIP), September 2003.

[164] L. Wei and M. Levoy, *Fast texture synthesis using tree-structured vector quantization*, SIGGRAPH, July 2000, pp. 479–488.

[165] Y. Weiss, *Belief propagation and revision in networks with loops*, Tech. report, Berkeley Computer Science Dept., 1998.

[166] Y. Weiss and W. T. Freeman, *On the optimality of solutions of the max-product belief propagation algorithm in arbitrary graphs*, IEEE Trans. Information Theory, Special Issue on Codes on Graphs and Iterative Algorithms **47** (2001), no. 2, 736–744.

[167] P. Whaite and F. P. Ferrie, *Autonomous exploration: driven by uncertainty*, IEEE Trans. on Pattern Analysis and Machine Intelligence **19** (1997), no. 3, 193–205.

[168] G. Winkler, *Image analysis, random fields, and dynamic monte carlo methods: a mathematical introduction*, Springer-Verlag, Berlin, 1995.

[169] A.P. Witkin, *Recovering surface shape and orientation from texture*, AI **17** (1981), no. 1-3, 17–45.

[170] G. Wyszecki and W.S. Stiles, *Color science: Concepts and methods, quantitative data and formulae*, Wiley, NewYork, 1982.

[171] J. Yedidia, W. Freeman, and Y. Weiss, *Constructing free energy approximations and generalized belief propagation algorithms*, Tech. report, Mitsubishi Electrical Research Laboratories, Inc., 2004.

[172] J. Yedidia, W. T. Freeman, and Y. Weiss, *Generalized belief propagation*, In Advances in Neural Information Processing Systems (NIPS), vol. 13, 2001, pp. 689–695.

[173] S. Zhu, Y. Wu, and D. Mumford, *Filters, random fields and maximum entropy (frame): towards the unified theory for texture modeling*, (1998), no. 2, 107–126.

[174] Z. Zhu, A.R. Hanson, and E.M. Riseman, *Generalized parallel-perspective stereo mosaics from airbone video*, IEEE Trans. on Pattern Analysis and Machine Intelligence **26** (2004), 226–237.

Document Log:

Manuscript Version Final

Typeset by $\mathcal{AMS}$-LaTeX — 14 December 2005

Luz Abril Torres Méndez

Centre for Intelligent Machines, McGill University, 3480 University St.,
Montréal (Québec) H3A 2A7, Canada, *Tel.* : (514) 398-6319
E-mail address: `latorres@cim.mcgill.ca`

Typeset by $\mathcal{AMS}$-LaTeX