Scene verification using an imaging model in 3-D computer vision

Document status and date:
Published: 01/01/1995

Document Version:
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

Download date: 04. Oct. 2023

SCENE VERIFICATION USING AN IMAGING MODEL IN 3-D COMPUTER VISION

Milan Hanajík

SCENE VERIFICATION USING AN IMAGING MODEL IN 3-D COMPUTER VISION

SCENE VERIFICATION USING AN IMAGING MODEL IN 3-D COMPUTER VISION

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Technische Universiteit Eindhoven, op gezag
van de Rector Magnificus, prof.dr. J.H. van Lint,
voor een commissie aangewezen door het College
van Dekanen in het openbaar te verdedigen op
donderdag 7 december 1995 om 14.00 uur

door

Milan Hanajík

geboren te Bratislava

Dit proefschrift is goedgekeurd door de promotoren:

prof.dr.ir. P.P.J. van den Bosch
en
prof.ir. F.J. Kylstra

Preface

De laatste loodjes wegen het zwaarst. This Dutch saying which can be translated as "it is the last straw that breaks the camel's back" perfectly expresses the feelings of every doctoral student when completing the last paragraphs of his/her thesis. It is the very same when I am writing these lines. There are still some small details that have to be done, but then the thesis is finally finished.

However, a willingness to help is not an implied endorsement. The aid of others was invaluable, but I alone am responsible for the opinions, technical details, and faults of this thesis.

<div align="right">

M. Hanajík
Eindhoven, October 22, 1995.

</div>

Abstract

A technique for the verification of scene descriptions is presented which employs an imaging model, i.e. a model describing the scene illumination and the image formation processes. It is assumed that the scene description to be verified has been created by a computer vision system. This verification problem involves the estimation of unknown scene parameters, and a decision-making problem based on the original acquired image and the synthetic image created with the imaging model.

The image of a scene projected onto an image plane is the result of an interaction of light with objects in the scene and a camera image plane. Discussed are surface reflectance models, a camera model, and three techniques for the computation of the global illumination, ray tracing, radiosity and stochastic ray tracing. An imaging model consisting of stochastic ray-tracing and the Torrance-Sparrow surface reflectance model is considered for the purpose of the scene description verification.

When using the imaging model, knowledge of surface reflectance parameters and parameters of the light sources is essential. These parameters are also the parameters of the imaging model. A maximum-likelihood technique for the estimation of the unknown parameters is proposed and elaborated on.

For the verification of a sceue description, a decision procedure consisting of the difference between the camera image and the synthetic image (an image created using the imaging model), the linear filtering of the result of the difference by a North filter, and the subsequent thresholding of the filtered image is proposed. It is shown that for the special case of a simple hypothesis and a simple alternative this procedure is a most powerful test in the Neyman-Pearson sense. The verification of the scene description is a decision problem with a simple hypothesis and a composite alternative. A method for the determination of suitable filter coefficients in the decision procedure is given.

Finally, some methods for the improvement of the scene description are given, which can be used if a discrepancy between the syuthetic image and the acquired images is detected. The techniques employing the imaging model are discussed.

The application of the technique is demonstrated in a number of experiments reported throughout the thesis, showing the feasibility of the proposed verification method.

Contents

Chapter 1

Introduction

1.1 3-D computer vision

Three-dimensional (3-D) computer vision is a challenging problem, and a field of intense research during the last 30 years. Haralick and Shapiro [33] specified computer vision as follows:

> *Computer vision, image understanding,* or *scene analysis* is that combination of image processing, pattern recognition, and artificial intelligence technologies which focusses on the computer analysis of one or more images, taken with a single/multiband sensor, or taken in a time sequence. The analysis recognises, locates the position and orientation, and provides sufficiently detailed symbolic description or recognition of those imaged objects deemed to be of interest in the three-dimensional environment.

The essential problem of the perception of 3-D scenes using 2-D images is the loss of one dimension by imaging process, and the resulting many-to-one mapping between the scene and the image. The subject of computer vision is the investigation of clues leading to the recovery of the missing information, as well as the development of the appropriate models, techniques, and data representations. Despite the progress in research and a huge increase of computers performance, the problem of the computer vision of generic scenes has not yet been solved. On the other hand, a number of efficient computer vision systems exists for a specific class of scenes, or a field of applications, such as 3-D MOSAIC (Herman 1986 [34]), SCERPO (Lowe 1987, [55]), and BONSAI (Flynn

and Jain, 1991, [17]). These systems are applicable for man-made scenes, like urban areas, indoor and robotic scenes *etc.* ACRONYM (Brooks 1981, [8]) is an example of a system capable of understanding generic scenes. The performance of ACRONYM decreases considerably in the presence of noise and artifacts, which can not be avoided in real images.

By comparison, the generic 3-D vision problem is routinely solved by human beings and animals. This is proved also by the fact, that humans are able to interpret 3-D scenes correctly from 2-D photographs of the scenes. Anyhow, the ability to recognise some scenes depends on the cultural context of humans subjected to the interpretation task. Even human beings require training and experience to interpret 2-D images correctly.

A typical set-up for 3-D computer vision is depicted in figure 1.1. The scene
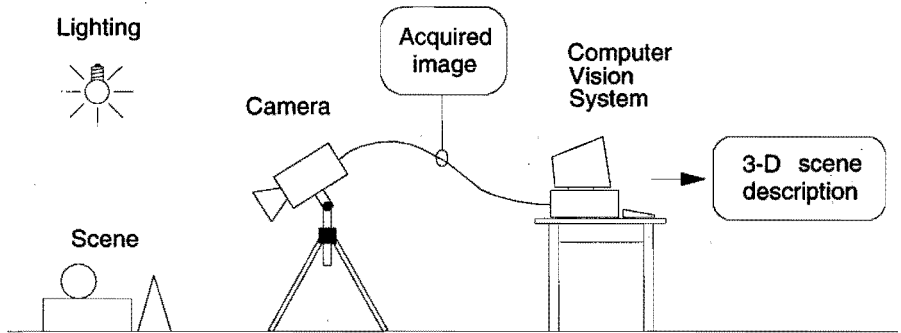
Figure 1.1: A computer vision set-up.

is a part of the real world observed by a camera. The image is a projection of the 3-D scene onto a two-dimensional (2-D) image plane, e.g. onto a surface of an imaging sensor inside a camera. The information contained in the image is processed by a computer vision system and yields a 3-D description of the sensed scene. A more precise definition of notions encountered in the computer vision set-up is given in section 1.3.

A number of clues can be employed to recover the missing third dimension. The most salient ones are summarized below:

Model-based techniques. When information about the dimensions of an object in the scene is available, the position of the object (the location and orientation with respect to a camera) can be derived by matching the 3-D object model to the 2-D image. This is possible even if some of the dimensions are not known and the model is parametrical [55, 56, 28, 30].

The prerequisite for the application of the object model is that the object recognition has already been solved.

Stereo vision. Two (or more) images of the scene are acquired from different view points. In the images, pairs of points can be found that are the projections of the same point in the 3-D scene. This is called a *correspondence problem*. Once this problem has been solved it is not difficult to compute the depth of the projected scene points by triangulation. The most critical part of stereo vision is the correspondence problem.

Shape from shading. Variations of the image brightness can be related to variations of the orientation of the viewed surface, while constant surface properties and homogeneous lighting are usually assumed. Intuitively, the brighter element of the investigated surface is tilted more toward the light source than the darker one. Under certain conditions the shape of the surface can be recovered from a single image of the surface, which is known as shape from shading [36, 39, 37].

Structure from motion. The essential idea is similar to that of stereo vision. With a moving camera a number of images of the scene from different viewpoints can be acquired and processed [2, 35, 62]. The spatial–temporal context in the image sequence can be employed for the solution of the correspondence problem. In addition, other quantities computed from the image series, for instance the *optical flow* [65, 40], can be employed.

Other clues. *Geometrical constraints* can be employed to compute the 3-D coordinates of scene points [73, 20]. These constraints are often combined with model-based techniques. The shape of objects can be recovered from the texture of the surfaces by *shape from texture* [3]. The shades cast by objects can be employed to locate objects and determine their heights in aerial photographs. By *photometric stereo*, the shape of observed objects can be determined by similar techniques as used by shape from shading from multiple images obtained under varying lighting conditions from the same viewpoint. An illumination by a light pattern, structured light, can be used to obtain the depth of viewed points by triangulation. The geometrical interpretation of lines of a linedrawing can be found by artificial intelligence techniques, like Waltz filtering [75].

1.2 Motivation for the new approach

As stated earlier, computer vision is an ill-posed problem due to the loss of depth information during the imaging process, resulting in many-to-one mapping between the 3-D scene and the 2-D image. On the other hand, a huge amount of 'redundant' information is present in the acquired images, although

only a fraction of this information is utilised for an inference by most computer vision algorithms. In model-based vision and feature-based stereo vision, only salient features, like detected edges and segmented regions that have been extracted from the image, are matched with the model features. One good reason why only these features are employed is that they are invariant to changes of the lighting conditions, the object surface quality, and other factors. Another reason is that data reduction is achieved by the extraction of features from an image, and consequently improves tractability of the problem by a computer.

To better exploit the data contained in the acquired camera image, and to use the information carried by each picture element of the digital image, we must have a model of how the value of the picture element is assigned. Shape from shading and photometric stereo use such a model, a *reflectance map*, which under a given illumination specifies the relation between the orientation of a surface, and the projected image brightness. However, the reflectance map is only valid, and shape from shading or photometric stereo only lead to correct results, under the following conditions:

- The reconstructed surface is uniformly lit, which in practice means that the light source is far from the surface. The surface can not be reconstructed at places in a shadow (possibly cast by the surface itself).

- A parallel projection onto the image plane is assumed. Central projection may be approximated by parallel projection, when the surface is sufficiently far from the camera, when compared with the distance between optical centre and the image plane.

- Often, a diffuse reflection (Lambertian reflection) of light on the reconstructed surface is assumed (see section 2.2.3). This excludes the reconstruction of glossy surfaces.

Under these conditions, the brightness of the surface projected onto the image plane is proportional to the term

$$E = I_0 \cos \theta_i, \tag{1.1}$$

where I_0 is the "intensity" of the light source, and θ_i is the angle between the surface normal and the incident direction of the light (see figure 1.2). The conditions given above are rarely simultaneously satisfied in real scenes. Actually, the surface is not only illuminated by a light source (or light sources), but also indirectly by the other surfaces in the environment. This is called *mutual illumination*. Experiments showing the effects of mutual illumination, and their consequences for computer vision algorithms have been reported by Forsyth and Zisserman [18, 19].
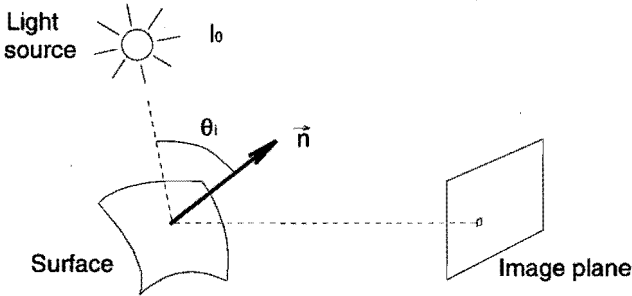
Figure 1.2: For a Lambertian surface and a single distant light source, an image brightness is proportional to $I_0 \cos \theta_i$.

Within the computer graphics community, in an effort to achieve photorealistic synthetic imagery, physically valid and computationally tractable models of the surface reflection, scene illumination and image formation processes have been developed. Given an appropriate scene description, a description of scene lighting, and a description of camera position, orientation and intrinsic properties, a synthetic image of the scene can be computed. We will call the algorithm used for the computation of the synthetic image a *model of the imaging process*, or an *imaging model*. The formula (1.1) for computation of an image brightness can be regarded as the imaging model that provides a coarse approximation to an actual imaging process. It takes into account only a single interaction of the projected surface with the light source. In general, an imaging model of a higher complexity, i.e. a model that takes into account more interactions, provides a better approximation to the actual imaging process. Thus accuracy is paid for by computation time.

In this thesis we investigate how to apply the concept of an imaging model to improve the performance of present computer vision algorithms, and make steps toward a new constructive computer vision algorithm.

1.2.1 Verification of scene descriptions

Consider a situation, when an image of the 3-D scene is acquired by a camera (or more cameras), and a computer vision system, using some of the clues mentioned earlier, infers a description of the scene (see figure 1.1). This is a scenario of many computer vision applications in robotics and production automation, navigation and exploration of dangerous and/or remote environments, design automation, monitoring, and elsewhere. Due to the ill-posedness of the vision problem, limitations of particular techniques, influence of random disturbances

and other factors, the resulting scene description is prone to errors. Still, the computer vision system has the ability to predict a (possibly erroneous) description of the scene observed by a camera. It is a *predictive system*.

A computational model of the imaging process, while being too complex to be employed for the inference, can be used for the verification of the scene description obtained by the predictive computer vision system. In this thesis we will discuss aspects of the 3-D scene description verification using the model of the imaging process—the imaging model. The essential idea is elucidated in figure 1.3:
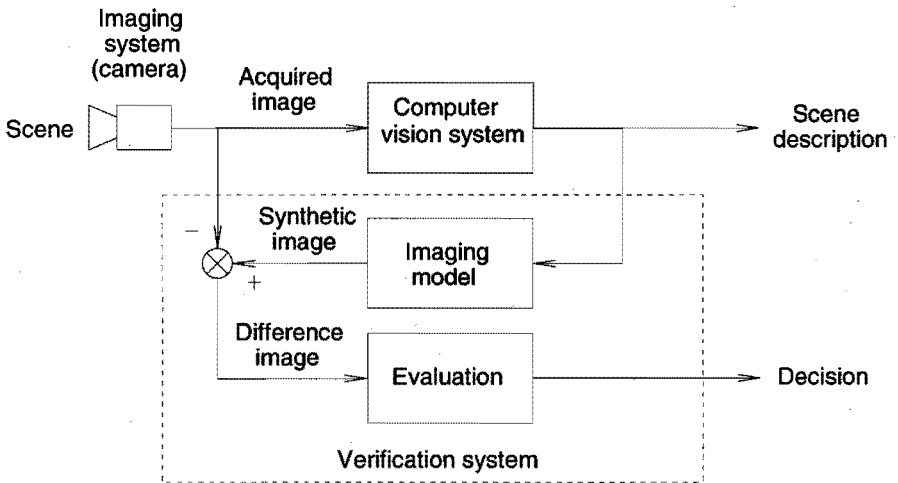
Figure 1.3: Diagram of the verification process.

The scene description which has been generated by a computer vision system is used as an input of the imaging model, and a synthetic image of the scene is generated. This synthetic image is compared with the actually acquired camera image (or images), and the difference between the images is evaluated. Based on the difference a decision is made about the correctness of the scene description. In figure 1.4 examples of images are shown, an actual, a synthetic and a difference image, which can be obtained by such a system. The differences between the acquired camera image and the synthetic image have different reasons: incorrect scene description, small inaccuracies, inaccuracy of the imaging model, noise, *etc.* The decision about the correctness of the scene description created by the computer vision system is going to be based on the difference image. Moreover, the location of inconsistencies between the images becomes visible.
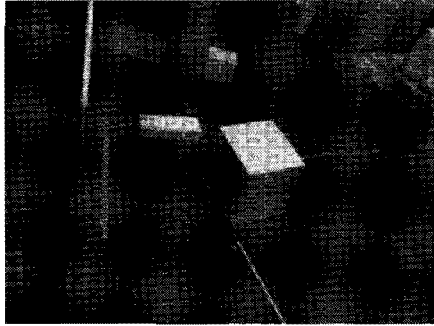
(a)
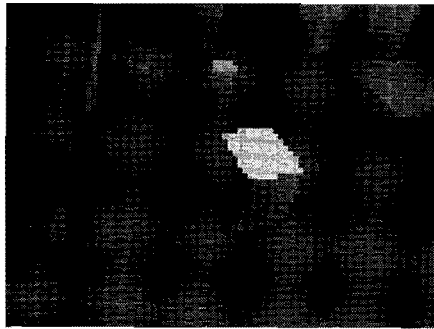
(b)
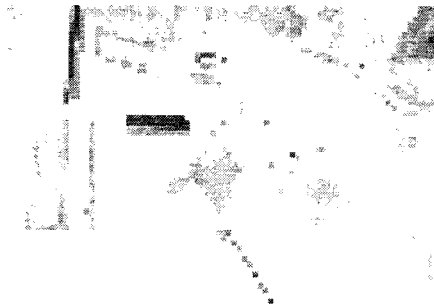
(c)

Figure 1.4: (a) An actual image acquired by a camera, (b) a synthetic image generated from a scene description using an imaging model, (c) the difference between images (a) and (b). In the difference image white colour represents no difference in intensities. The differences resulting in dark regions in the image (c) are caused by an incorrect scene description (a missing object, as well as small inaccuracies), and by an inaccuracy of the imaging model.

The suggested verification scheme needs some refinement for the following reasons:

- The scene description obtained from the computer vision system might possibly miss a part of information necessary for the modelling of the imaging process and the generation of the synthetic image. Particularly, the reflectance properties of object surfaces are often not available. Also the lighting conditions might be partly unknown, since they depend on the position of the sun and weather conditions. In an indoor environment this might be due to changes of a part of the illumination coming via windows, depending on the outside conditions.

- The scene description usually does not contain the description of all details in the scene.

- The imaging model is not perfect. It is only an approximation of the actual scene illumination and imaging processes.

- The camera image is influenced by random disturbances (noise).
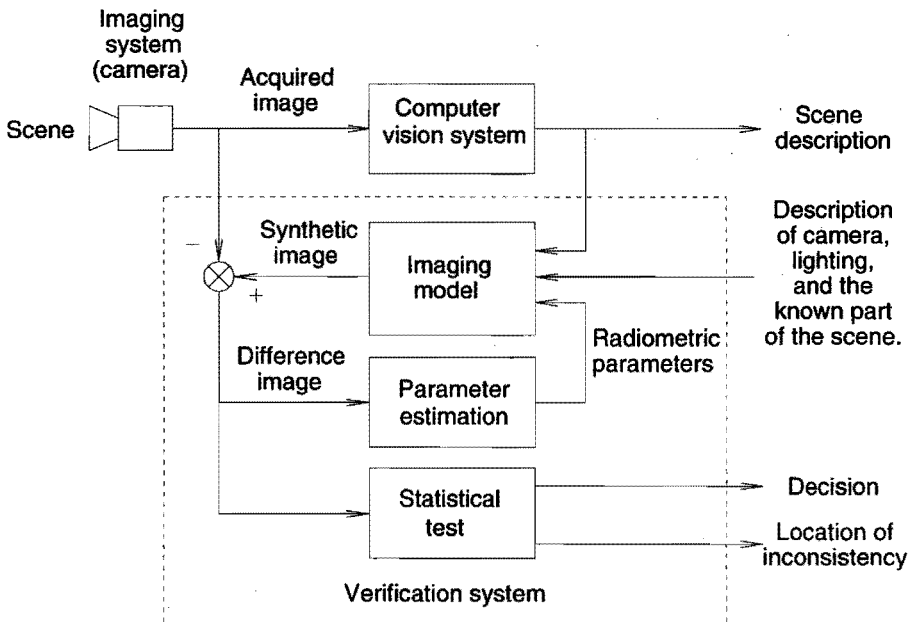
Figure 1.5: Refined diagram of the verification process.

The verification procedure must include a mechanism for the restoration of unknown surface reflectance properties and of the unknown part of the illumination from the available images. This is done in the proposed diagram by the

block entitled "Parameter estimation". The difference between actual and synthetically generated image is tested by statistical techniques, to reveal whether the difference is a consequence of random influences, or whether it is due to an incorrect description of the scene. This is done by the block "Statistical test".

A considerable part of the thesis is concerned with the development of the proposed verification scheme and an assessment of its performance. An implementation of the scene description verification using the imaging model should result in an improved reliability of the complete system (the combination of the predictive computer vision system with scene description verification).

1.2.2 An iterative computer vision algorithm

Some currently existing techniques can be adopted to exploit an accurate model of the imaging process. Particulary, the result of the verification procedure and the information obtained during the verification can be used by some algorithms for the improvement of the existing scene description. The application of the verification procedure followed, in the case of detected scene description errors, by the use of such a technique, results in an iterative improvement of the available scene description. The iterative improvement of the scene description is also discussed in the thesis.

1.3 Definitions of terminology

Scene is a part of the real world observed by a camera.

Scene description is an abstract description of the scene, consisting of appropriate descriptions of objects contained in the scene. A scene description can be generated by a computer vision system. Alternatively, it might be obtained by conversion from a CAD system, from a database, or created manually.

Image is a recorded projection of the 3-D scene onto a 2-D image plane. Usually it is obtained by a central projection via the optical centre of the system (see figure 1.6). Sometimes, parallel projection along the optical axis of the system is considered. Then we will explicitly refer to parallel projection. An image can be described by a function I of two coordinates u and v defined on a rectangular region of the image plane, $I(u, v)$. The value I is either a scalar (e.g. an image brightness of a grey scale image), or a vector for multiband images (e.g. colour images).

(x,y,z)

Image plane

Optical axis

Optical centre   I(u,v)

(u,v)
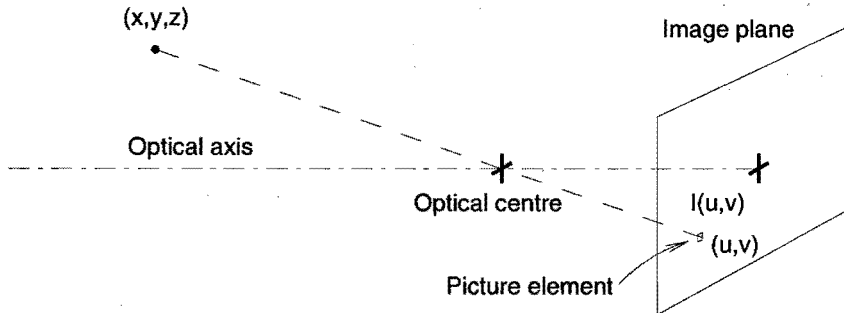
Picture element

Figure 1.6: Central projection of the 3-D scene onto the image plane.

Picture element, PEL. The image can be partitioned by a rectangular grid into small rectangular elements of the same size. These elements are called picture elements, or PELs.

Digital image is a discretized and quantized version of the image $I(u,v)$. It may be thought as a two-dimensional array (matrix) $I_{i,j}$, where each value $I_{i,j}$ is obtained from a continuous image $I(u,v)$ by sampling in the centre of the respective PEL, or by an integration over the PEL. We will often refer to a digital image as just an image, where the difference is obvious from the context, or is not significant for a problem under discussion.

Computer vision system is a system by which the analysis of one or more images acquired with an imaging sensor (a camera) creates a scene description.

Imaging model is the model of processes of light propagation, interaction of light with surfaces, image formation and other processes that determine an image projected onto an image plane, given the scene and lighting description.

Synthetic image is an image generated using an imaging model.

1.4   Outline of the thesis

In chapter 2, the computational model of the imaging process is developed. The model originates from geometrical optics and from computer graphics research, and consists of three components: a local interaction of light with surfaces (section 2.2), an image formation model (section 2.3), and a global illumination computation (sections 2.4–2.7). A summary of an imaging model using

stochastic ray tracing, as implemented and used in the experiments reported in the thesis, is given in Appendix A.

Chapter 3 is devoted to the estimation of missing scene description parameters, from the acquired images and the geometrical description of the scene. The unknown parameters include the parameters of the reflectance model of surfaces of objects in the scene and also the intensity of the light sources.

The verification of the consistency between the acquired camera images and the hypothetic scene description is discussed in chapter 4. The verification is based on the results of decision theory.

The methods of improving an incorrect scene description are discussed in chapter 5. The correction of the scene depends on the kind of discrepancy detected. The use of shape from shading and photometric stereo are discussed in more detail.

Finally, concluding remarks are given in chapter 6. The original contributions of the thesis are concentrated in chapters 3, 4, and 5. The use of an illumination model taking into account global light interactions is novel, and enables the computer vision algorithms to exploit the information contained in the image brightness.

Chapter 2

Computational model of the imaging process

An image of a scene projected onto a camera image plane is the result of an interaction of light with objects in the scene and with the camera. Light emitted by light sources is reflected, refracted and scattered by objects in the scene and a part of the light passing through the lens and falling onto the image plane creates an image. This image is converted by an imaging sensor into an electric signal and processed further by the computer vision system.

In this chapter a computational model of the imaging process based on geometrical optics is developed. The topic is treated in a bottom-up way. First, in section 2.1 the basic notions and principles are described. In section 2.2 the local interactions of light with object surfaces are treated, in section 2.3 the model of the image formation on an image plane is given, and in sections 2.4 through 2.7 methods for the computation of global illumination are discussed. Finally, we propose a computational model of the imaging process. The model is applicable for verification purposes.

2.1 Basic principles and notions

Light is an electromagnetic phenomenon. In physical optics the processes related to the propagation of light are analyzed using the electromagnetic wave theory. In situations where the wave length of light is very small when compared with the dimensions of illuminated objects, the electromagnetic character of the light can be ignored and geometrical optics is applicable. The basic observation

of geometrical optics is that in a homogeneous transparent medium light prop-
agates along straight lines: rays. Another observation is that light rays having
different directions and intersecting each other have no influence on each other,
i.e. they do not interfere. When light ray is incident on a boundary of two
media, one ray is reflected from the boundary, while another ray penetrates
through the boundary and is refracted (figure 2.1). The angles of incident,
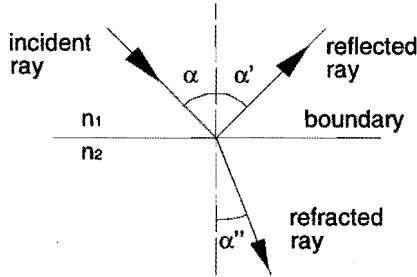
Figure 2.1: Reflection and refraction of light on a boundary of two media.

reflected and refracted rays α, α' and α'' satisfy

$$\alpha = \alpha', \tag{2.1}$$

$$n_1 \sin \alpha = n_2 \sin \alpha'', \tag{2.2}$$

where n_1 and n_2 are the refraction indices of the two media. The relative
amounts of reflected and refracted light are determined by the Fresnel coefficient[1]
F,

$$F = \frac{1}{2} \left(\frac{\sin^2(\alpha - \alpha'')}{\sin^2(\alpha + \alpha'')} + \frac{\tan^2(\alpha - \alpha'')}{\tan^2(\alpha + \alpha'')} \right) \qquad \text{for } \alpha \neq 0, \alpha'' \neq 0, \tag{2.3}$$

and

$$F = \left(\frac{n_2 - n_1}{n_2 + n_1} \right)^2 \qquad \text{for } \alpha = \alpha'' = 0. \tag{2.4}$$

While F is the fraction of incident light that is reflected, $1 - F$ is the fraction
of incident light that is refracted.

Two photometric quantities are introduced to measure the amount of light in-
cident on a surface and radiated by the surface.

Irradiance refers to the amount of light falling on a surface. It is the power
of radiant energy incident on a surface per unit area and is measured in
$[\text{W}\cdot\text{m}^{-2}]$.

---

[1] The Fresnel coefficient depends on the light polarization. The one given above is for
non-polarized light.

Radiance refers to the amount of light leaving a surface in a certain direction. It is the power of radiated or reflected energy per unit solid angle in the specified direction, per unit foreshortened[2] surface area. Radiance is measured in $[\text{W} \cdot \text{m}^{-2} \cdot \text{sr}^{-1}]$.

The necessity for the introduction of radiance stems from the fact, that a surface can radiate into a whole hemisphere of possible directions with different amounts of energy in different directions.

2.2   Interaction of light with a rough surface

2.2.1   Light reflection geometry and the bidirectional reflectance distribution function

Consider an elementary surface patch, and a light reflection geometry depicted in figure 2.2. Let $\vec{n}$ be a normal vector of the surface patch, $\vec{l}$ be a unit length
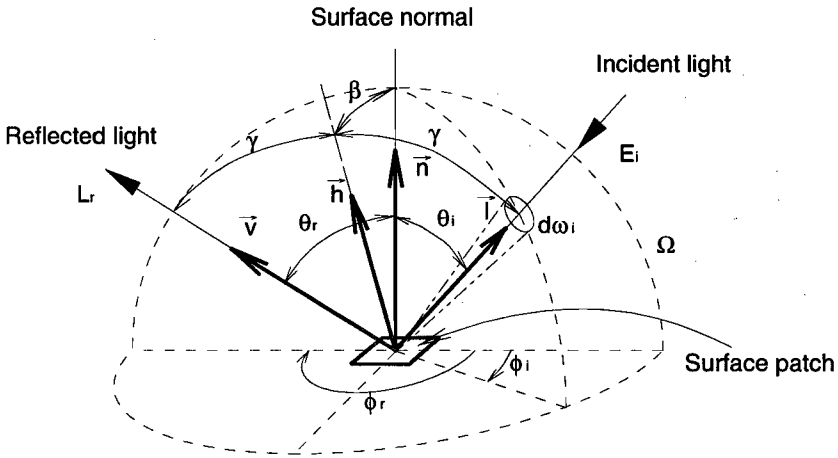
Figure 2.2: Definition of angles and directions.

vector in the direction toward a light source, and $\vec{v}$ be a unit length vector in the direction toward a viewer. Let us define vector $\vec{h}$ as a unit length vector that bisects the angle between vectors $\vec{l}$ and $\vec{v}$, i.e. $\vec{h}$ is given by $\vec{h} = (\vec{v}+\vec{l})/|\vec{v}+\vec{l}|$. We

---

[2] Foreshortened surface area is the area of the surface projected in the specified direction, thus it is a product of the original surface area, and the cosine of the angle between the surface normal and the specified direction.

will refer to the directions pointed to by the vectors $\vec{l}, \vec{v}$ and $\vec{h}$, as the directions $\vec{l}, \vec{v}$ and $\vec{h}$ respectively. The direction of incident light $\vec{l}$ is specified by two angles, zenith θ_i and azimuth ϕ_i. Similarly, the direction of reflected light $\vec{v}$ is specified by the angles zenith θ_r and azimuth ϕ_r. The angle β between the normal $\vec{n}$ and the direction $\vec{h}$ is called the *off-specular angle*, since for the direction $\vec{h}$ identical with $\vec{n}$, i.e. for $\beta = 0$, the configuration is the geometry of a specular reflection on an ideal surface, as in figure 2.1.

On a rough surface, one part of the incident light is absorbed by the surface, and another part is scattered into the whole hemisphere of directions above the surface. A unified way of specification of the reflectance, in terms of the reflection geometry, was introduced by Nicodemus [63]. The *bidirectional reflectance* ρ_{bd} of the surface is defined as the relationship between the irradiance dE_i arriving from the direction θ_i, ϕ_i, and the radiance dL_r reflected in the direction θ_r, ϕ_r

$$\rho_{bd} = \frac{dL_r(\theta_r, \phi_r)}{dE_i(\theta_i, \phi_i)}. \tag{2.5}$$

Bidirectional reflectance is a function of four angles, $\rho_{bd}(\theta_i, \phi_i; \theta_r, \phi_r)$, or of the two directions, and is called the *bidirectional reflectance distribution function* (BRDF). It depends on physical, texture and finish properties of the surface. Furthermore, it is a function of the wave length and polarization of incident light. However, we disregard the dependence on the wave length and the polarization for the sake of simplicity.

2.2.2   The total surface radiance

The contribution of the light coming from the direction θ_i, ϕ_i to the radiance of light reflected in the direction θ_r, ϕ_r is

$$dL_r(\theta_r, \psi_r) = \rho_{bd}(\theta_i, \psi_i; \theta_r, \psi_r) \, dE_i(\theta_i, \phi_i)$$
$$= \rho_{bd}(\theta_i, \psi_i; \theta_r, \psi_r) \, L_i(\theta_i, \phi_i) \, \cos\theta_i \, d\omega_i, \tag{2.6}$$

where $L_i(\theta_i, \phi_i)$ is the radiance of another surface found in the direction θ_i, ϕ_i from the reflecting surface, $d\omega_i$ is an infinitely small solid angle of the hemisphere about direction θ_i, ϕ_i, and the term $\cos\theta_i$ accounts for the foreshortening of the reflecting surface as seen from direction θ_i, ϕ_i. Hence, the total radiance L_r emanating from the surface in the direction θ_r, ϕ_r can be computed as the integral of dL_r over the entire hemisphere Ω of incident direction θ_i, ϕ_i, plus the radiance L_e emitted by the reflecting surface itself

$$L_r(\theta_r, \phi_r) = L_e(\theta_r, \phi_r) + \int_\Omega \rho_{bd}(\theta_i, \phi_i; \theta_r, \phi_r) L_i(\theta_i, \phi_i) \cos\theta_i \, d\omega_i. \tag{2.7}$$

Obviously, only for light sources is the term $L_e(\cdot)$ not equal to zero.

The knowledge of the BRDFs of surfaces is essential for the computation of the global scene illumination. The BRDF function values can be obtained by exhaustive measurements over the whole range of angles of incidence and reflection. For several reasons the use of the BRDF is impractical. The storage of the BRDF requires a great amount of memory, depending on a chosen discretization. Available measurements seldom cover the entire domain of the BRDF. Numerous reflectance models, approximations of the BRDF, have been developed to avoid the use of BRDF in computations.

2.2.3 Lambertian reflectance model

An ideal diffuse surface, or a Lambertian surface [49], reflects light uniformly in all directions in the sense, that the Lambertian surface appears equally bright from all viewing directions[3]. The mechanism that produces Lambertian reflection is *internal scattering*. Light rays penetrate the surface and encounter microscopic inhomogeneities inside the medium. They are repeatedly reflected and refracted inside the medium, and a part of them leaves the medium in an arbitrary direction, which results in a uniform radiance of the surface. The BRDF of the Lambertian surface is constant, $\rho_{bd}(\theta_i, \phi_i; \theta_r, \phi_r) = k_{dif}$.

Often, the Lambertian surface is characterized by a *surface albedo* κ, defined as the fraction of energy incident on the surface that is reflected. The relation between k_{dif} and κ can be obtained from an energy balance. Assuming the irradiance of the surface (illuminated from an arbitrary direction) is E_i, the surface radiance is $L_r = k_{dif}E_i$. An integration of reflected radiance over the entire hemisphere should lead to the value κE_i,

$$\kappa E_i \stackrel{\triangle}{=} \int_\Omega k_{dif}E_i \cos\theta_r \, d\omega_r = k_{dif}E_i \int_0^{2\pi}\int_0^{\pi/2} \cos\theta_r \sin\theta_r \, d\theta_r \, d\phi_r = k_{dif}E_i\pi. \quad (2.8)$$

From (2.8) it follows that the bidirectional reflectance of the Lambertian surface is $k_{dif} = \kappa/\pi$.

2.2.4 Torrance-Sparrow reflectance model

A more realistic reflectance model has been introduced by Torrance and Sparrow [72]. The Torrance-Sparrow reflectance model has many times been used

---

[3]This does not imply, that equal amounts of radiant energy are reflected into all directions. The amount of energy reflected by the Lambertian surface is proportional to the cosine of the angle between the reflected ray direction and the surface normal $\cos\theta_r$, so that the radiance is constant for all directions.

in computer graphics [6], [11], and recently also in computer vision [59] due to
its acceptable complexity and a high degree of match with the measured data.

It is assumed that the surface consists of small, randomly disposed, mirror-like
facets, as it is illustrated in figure 2.3.   The distribution of surface normal
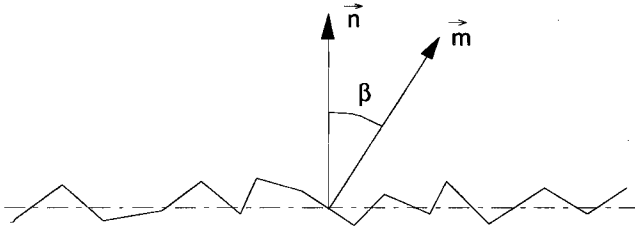
*Figure 2.3: Surface consisting of planar micro facets. $\vec{n}$ is the normal of the
macroscopic surface, and $\vec{m}$ is the normal of a micro facet.*

orientations of the micro facets is assumed to be known.

Reflected light is composed of two components: a *diffuse component* produced
by internal scattering of light under the surface, and a *specular component*
caused by a specular reflection from the mirror-like facets. The contribution to
the reflected radiance dL_r given by (2.6) can be written as the sum

$$dL_r(\theta_r, \phi_r) = dL_{r,d}(\theta_r, \phi_r) + dL_{r,s}(\theta_r, \phi_r), \tag{2.9}$$

where $dL_{r,d}(\theta_r, \phi_r)$ and $dL_{r,s}(\theta_r, \phi_r)$ are respectively the diffuse and specular
components of the surface radiance.

The diffuse component can be described by a Lambertian reflectance model,
hence

$$dL_{r,d}(\theta_r, \phi_r) = k_{\text{dif}} \, dE_i(\theta_i, \phi_i) = k_{\text{dif}} \, dL_i(\theta_i, \phi_i) \cos\theta_i \, d\omega_i. \tag{2.10}$$

The specular component of the radiance is reflected directly by the micro facets
according to laws of geometrical optics, and depends on more factors. As the
angles of the incident and the reflected ray must obey equation (2.1), only the
micro facets with the orientation of surface normals identical to $\vec{h}$ (see figure 2.2)
reflect light from the direction $\vec{l}$ into the direction $\vec{v}$. The orientation of micro
facets obeys a probability distribution, which can be characterized by a density
function D, which is for an anisotropic surface a function of a single variable β.
Torrance and Sparrow have assumed a Gaussian distribution:

$$D(\beta) = C \exp\left\{-\frac{\log 2}{\beta_h^2}\beta^2\right\} = C \, 2^{-\frac{\beta^2}{\beta_h^2}}, \tag{2.11}$$

where parameter β_h is the value of the angle β, for which the density of facets $D(\beta)$ drops to $1/2$ of the maximal value, and C is a constant factor normalizing the density function so that it is a probability density function. The reflected radiance $dL_{r,s}(\theta_r, \phi_r)$ is proportional to the value of $D(\beta)$ for the angle β determined by the reflection geometry.

Another factor affecting the intensity of specularly reflected light is the Fresnel coefficient F, given by (2.3) and (2.4), which is a function of refraction indices of the reflecting media, and the reflection geometry.

Furthermore, the reflected radiance is affected by the effects called *shadowing* and *masking*. Shadowing and masking are illustrated in figure 2.4. Shadowing
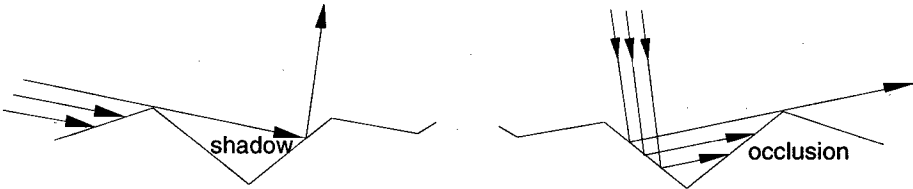
Figure 2.4: Shadowing and masking of surface micro facets.

takes place when the incident ray is almost parallel with the rough surface. Then a considerable part of the micro facets is in the shadow of other ones, reducing the reflectance which would be obtained by taking into account all micro facets with the normal orientation $\vec{h}$. In a similar way, masking takes place when the reflected ray is almost parallel with the surface. Then, a considerable part of micro facets is not seen by an observer, as they are occluded by other facets. The reflectance is corrected for shadowing and masking effects by multiplication with a geometrical attenuation factor G, defined as

$$G = \min \left\{ 1, \ 2\frac{\cos\beta\cos\theta_r}{\cos\gamma}, \ 2\frac{\cos\beta\cos\theta_i}{\cos\gamma} \right\} \tag{2.12}$$

where the angles β, γ, θ_i and θ_r were defined in figure 2.2. For the derivation of the geometrical attenuation factor we refer to [72], or [6]. The factor has a value between zero and one, the value one is attained when neither shadowing, nor masking, takes place.

The complete Torrance-Sparrow reflectance model is given by

$$dL_r = \left(k_{dif}\cos\theta_i + k_{spec}\frac{2^{-\beta^2/\beta_h^2} \ F(n) \ G}{\cos\theta_r} \right) L_i d\omega_i \tag{2.13}$$

or, the BRDF of the surface is approximated by

$$\rho_{\mathrm{bd}} = k_{\mathrm{dif}} + k_{\mathrm{spec}} \frac{2^{-\beta^2/\beta_{\mathrm{h}}^2} \, F(n) \, G}{\cos\theta_{\mathrm{i}} \cos\theta_{\mathrm{r}}}, \qquad (2.14)$$

where k_{dif} is the coefficient of diffuse reflection, k_{spec} is the coefficient of specular reflection, β_{h} is the tilt angle of micro facets, for which their density drops to 1/2 of the maximal value, n is the index of refraction, $F(n)$ is the Fresnel coefficient given by (2.3), G is the geometrical masking/shadowing factor given by (2.12), and $\theta_{\mathrm{i}}, \theta_{\mathrm{r}}$ and β were defined in figure 2.2.

In practice, the values of trigonometric functions in (2.12)–(2.14) are computed as vector inproducts: $\cos\theta_{\mathrm{i}} = \vec{n}{\cdot}\vec{l}$, $\cos\theta_{\mathrm{r}} = \vec{n}{\cdot}\vec{v}$, $\cos\beta = \vec{n}{\cdot}\vec{h}$ and $\cos\gamma = \vec{v}{\cdot}\vec{h} = \vec{l}\cdot\vec{h}$. This computation is faster then the evaluation of trigonometric functions. The Torrance-Sparrow model is specified by four parameters $k_{\mathrm{dif}}, k_{\mathrm{spec}}, \beta_{\mathrm{h}}$ and n. Once these parameters are specified, the bidirectional reflectance depends only on the reflection geometry.

2.2.5   Other reflectance models

Some authors use a simplified Torrance-Sparrow reflectance model, where the Fresnel factor is approximated by a constant, which is then absorbed in the parameter k_{spec} [59].

A number of reflectance models resulting in naturally looking synthetic images, but having nothing or little in common with the underlying light reflection mechanisms, has been developed in computer graphics. These so-called black-box models are fast and simple and they were rarely confronted with measured data. According to Phong [64], the reflected radiance is computed by

$$dL_{\mathrm{r}} = [k_{\mathrm{dif}} \cos\theta_{\mathrm{i}} + k_{\mathrm{spec}}(\cos\beta)^{n_{\mathrm{sp}}}] \, dE_{\mathrm{i}}, \qquad (2.15)$$

where n_{sp} is the so called specular power and the rest of the symbols is defined as in the Torrance-Sparrow model. The specular power n_{sp} determines the width of the BRDF specular lobe. The higher n_{sp}, the narrower the lobe is, and the reflection looks more like the specular one. We avoid the use of black-box models for illumination computation because of their insufficient accuracy.

Reflectance models based on physical optics use the electromagnetic wave theory and reflectance is obtained by solving Maxwell's equations using the boundary conditions imposed by the reflecting surface. A detailed treatment of the physical optics reflectance models can be found in Beckmann and Spizzochino [4]. Physical optics models are more complex, and therefore also less popular in visualization and computer vision. A recent comparison of the Torrance-Sparrow and Backmann-Spizzochino reflectance models can be found in [59].

There seems to be no significant difference between these two models for visible light and usual surfaces, except in the case of almost specular surfaces with a very fine roughness. These are better described by a Backmann-Spizzochino model.

2.3   Irradiance of the image plane

Consider a lens with diameter d at a distance f (the focal distance[4]) from the image plane (see figure 2.5). Furthermore, consider an object surface patch dO having a radiance L in the direction towards the lens. The patch dO is projected onto a picture element of area dA on the image plane. Let φ be the angle between the optical axis of the lens, and the ray projecting the patch dO onto the image plane via the optical centre of the lens. The foreshortened area
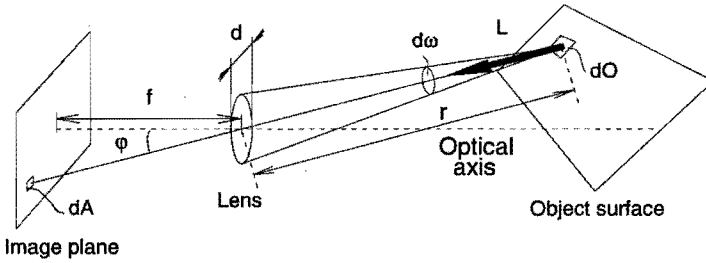
Figure 2.5: Surface patch projected onto an image plane.

dO_{fs} of the patch dO when viewed from the lens is

$$dO_{fs} = dA \left(\frac{r \cos \varphi}{f} \right)^2 \cos \varphi,$$

and the solid angle $d\omega$ subtended by the lens when viewed from patch dO is

$$d\omega = \frac{\pi d^2}{4} \cos \varphi \, \frac{1}{r^2}.$$

In both expressions above r denotes the distance from the lens to the patch dO, which cancels when computing the radiant power collected by the lens. The irradiance E of the picture element dA is the radiance L radiated by patch dO and collected by the lens with the diameter d, divided by the picture element area dA,

$$E = \frac{L \, dO_{fs} \, d\omega}{dA} = L \frac{\pi}{4} \left(\frac{d}{f} \right)^2 \cos^4 \varphi. \qquad (2.16)$$

---

[4]The image plane is placed at such distance f from the optical centre, that the projected image is in focus. For distant objects this distance is almost equal to the focal distance.

From equation (2.16) it follows that the image plane irradiation, and hence the imaging sensor output, is proportional to the radiance L of the surface projected onto the image plane. Notice, that neither the orientation of the object surface patch dO, nor the distance r of the object surface from the lens, does affect the image irradiance. Therefore, the surface radiance is subjectively perceived as a 'brightness' of the surface, and it is meaningful to compute this quantity for scene objects. The consequence of the factor $\cos^4 \varphi$ is that equally bright objects appear darker, when they are projected outside the image plane centre. The acquired image can be corrected for this effect.

2.4   Computation of the global illumination

Let S be the union of all surfaces in the scene which contribute to the scene illumination. We will define a new quantity, the *two-point transport intensity* $I(x_1, x_2)$, as the power of radiant energy passing from an infinitely small surface patch around point $x_2 \in S$ to an infinitely small surface patch around point $x_1 \in S$, per unit area of patch around x_2 per unit area of patch around x_1. The units of I are [W·m^{-4}]. By *global illumination* of the scene we mean a mapping $I : S \times S \to R$, which assigns to each pair of scene points (x_1, x_2) a value of the two-point transport intensity $I(x_1, x_2)$ from point x_2 to point x_1.

The domain of integration in equation (2.7) can be changed from the hemisphere Ω above the surface patch, to the set S containing all surface patches in the scene. Furthermore, the integral in equation (2.7) can be rewritten in terms of two-point transport intensities between scene points

$$I(x_1, x_2) = g(x_1, x_2) \left[\epsilon(x_1, x_2) + \int_S \rho(x_1, x_2, x_3)\, I(x_2, x_3)\, dx_3 \right], \qquad (2.17)$$

where $g(x_1, x_2)$ is a geometric factor, $\epsilon(x_1, x_2)$ is the two-point transport intensity emitted from x_2 to x_1, $\rho(x_1, x_2, x_3)$ is related to the intensity of light scattered from the point x_3 to the point x_1 by the surface patch around x_2, and dx_3 denotes an elementary surface patch around point x_3. The value of the geometric factor $g(x_1, x_2)$ depends on the distance r_{12} of the points x_1 and x_2, and the mutual visibility of these two points. The value of $g(x_1, x_2)$ is $1/r_{12}^2$ in the case when point x_1 is visible from point x_2, i.e. there is no occlusion between x_1 and x_2, and is zero in the case of an occlusion.

The equation (2.17) was introduced by Kajiya [45], and is called the *rendering equation*. It simply states, that the transport intensity of light $I(x_1, x_2)$ from the point x_2 to the point x_1 is the sum of light emitted by the elementary surface patch around point x_2 toward the point x_1, and the integral of the intensity of

light reflected by the patch around x_2 from point x_3 to point x_1, integrated over all points $x_3 \in \mathcal{S}$ in the scene.

The relations between the introduced quantities I, g and ρ, and the quantities defined in sections 2.1 and 2.2 can be easily derived. Consider two surface patches dx_1 and dx_2 around two points x_1 and x_2 respectively, as it is shown in figure 2.6. The solid angle subtended by dx_1 when viewed from point x_2
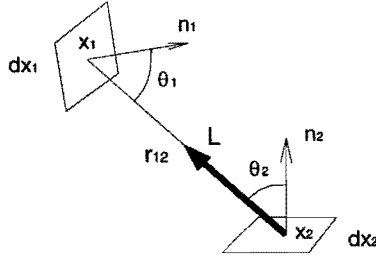
Figure 2.6: Radiant energy transport between two surface patches.

is $dx_1 \cos\theta_1 / r_{12}^2$, and the area of patch dx_2 projected into direction toward x_1 is $dx_2 \cos\theta_2$. From definitions of radiance and two-point transport intensity it follows that

$$I(x_1, x_2)\, dx_1\, dx_2 = L\, \frac{\cos\theta_1 \cos\theta_2}{r_{12}^2} dx_1\, dx_2,$$

hence the relation between radiance L and the transport intensity I is given by

$$I(x_1, x_2) = L\, \frac{1}{r_{12}^2}\, \cos\theta_1 \cos\theta_2. \tag{2.18}$$

Similarly, $\rho(x_1, x_2, x_3)$ is related to bidirectional reflectance ρ_{bd} of the surface at point x_2 (see figure 2.7). From (2.17) it can be seen that in the case of not occluded points x_1 and x_2, i.e. $g(x_1, x_2) = 1/r_{12}^2 \neq 0$, $\rho(x_1, x_2, x_3)$ is given by

$$\rho(x_1, x_2, x_3) = \frac{dI(x_1, x_2)}{g(x_1, x_2) I(x_2, x_3)\, dx_3}.$$

After substitution for $I(x_1, x_2)$ and $I(x_2, x_3)$ from (2.18), with the observation that the solid angle subtended by patch dx_3 when viewed from point x_2 is $d\omega = (\cos\theta_3\, dx_3)/r_{23}^2$, and with the use of definition (2.5), we obtain the relation between ρ and the bidirectional reflectance ρ_{bd}

$$\rho(x_1, x_2, x_3) = \frac{dL_r \cos\theta_1 \cos\theta_2}{L_i \cos\psi_2\, d\omega} = \rho_{bd}\, \cos\theta_1 \cos\theta_2. \tag{2.19}$$

In the case that x_1 and x_2 are occluded and $g(x_1, x_2) = 0$, the value of $\rho(x_1, x_2, x_3)$ has no influence on $I(x_1, x_2)$ in (2.17), thus it may also be defined by (2.19).
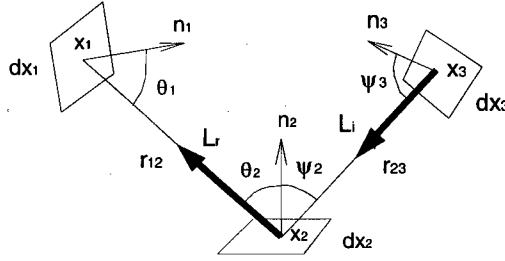
Figure 2.7: Scattering of radiant energy from point x_3 to point x_1 by patch dx_2.

2.4.1   Solution of the rendering equation

The global illumination $I : \mathcal{S} \times \mathcal{S} \mapsto R$ must obey the rendering equation (2.17).
This follows the lines of geometrical optics, as the rays intersecting each other in
space are regarded to be independent while interactions occur only at surfaces.
Equation (2.17) suggests the solution for I by recurrent substitution for $I(x_1, x_2)$
(cf. Courant and Hilbert [12], pages 112–163),

$$I(x_1, x_2) = g(x_1, x_2)\, \epsilon(x_1, x_2) +$$

$$+ g(x_1, x_2) \int_{\mathcal{S}} \rho(x_1, x_2, x_3)\, g(x_2, x_3)\, \epsilon(x_2, x_3)\, dx_3 +$$

$$+ g(x_1, x_2) \int_{\mathcal{S}} \rho(x_1, x_2, x_3)\, g(x_2, x_3) \left[\int_{\mathcal{S}} \rho(x_2, x_3, x_4)\, I(x_3, x_4)\, dx_4 \right] dx_3$$

$$= \cdots \tag{2.20}$$

If we denote the integral operator in (2.17) by $\mathcal{M}$, the rendering equation can
be rewritten as

$$I = g\epsilon + g\mathcal{M}I, \tag{2.21}$$

and the solution by recurrent substitutions is

$$I = g\epsilon + g\mathcal{M}g\epsilon + g\mathcal{M}g\mathcal{M}g\epsilon + (g\mathcal{M})^3 g\epsilon + \cdots. \tag{2.22}$$

The series in (2.22) is known as a *Neumann's series*. It converges under the
condition that the spectral norm of the operator $g\mathcal{M}$ is less than one. It turns
out that this condition is met in our case and the series converges, which is also
justified by a fact, that we are dealing with a model a physical phenomenon with
an obvious solution, the global scene illumination. The physical interpretation
of (2.22) is that the radiant energy transfer from point x_2 to point x_1 is the
sum of the direct term $g\epsilon$, the once scattered term $g\mathcal{M}g\epsilon$, the twice scattered
term $g\mathcal{M}g\mathcal{M}g\epsilon$, etc.

A finite dimensional operator M can be obtained by partitioning the scene S into a number of small patches $S_i \subset S, i = 1, \ldots, n$, such that $S = \cup_{i=1}^{n} S_i$, and $S_i \cap S_j = \emptyset$ for $i \neq j$. We make the approximation that for all pairs of patches $(S_i, S_j), i, j = 1, \ldots, n$, and for all pairs of points selected from these patches (x_i, x_j), $x_i \in S_i$ and $x_j \in S_j$, the two-point transport intensity $I(x_i, x_j)$ is constant, and can be denoted as I_{ij}:

$$I(x_i, x_j) = I_{ij} \qquad \text{for all } x_i \in S_i, x_j \in S_j. \tag{2.23}$$

Similarly, for all pairs of points (x_i, x_j), $x_i \in S_i$ and $x_j \in S_j$, the emitted two-point transport intensity $\epsilon(x_i, x_j)$ is approximated by a constant ϵ_{ij}, and for all triples of points (x_i, x_j, x_k), $x_i \in S_i$, $x_j \in S_j$ and $x_k \in S_k$, the scattered light intensity $\rho(x_i, x_j, x_k)$ is approximated by a constant ρ_{ijk}.

We can select n points in the scene $x_i \in S_i$, $i = 1, \ldots, n$, so that from each patch one point is selected. The integral in the rendering equation (2.17) then turns into a summation over the patches $S_i, i = 1, \ldots, n$.

$$I(x_i, x_j) = g(x_i, x_j)\epsilon(x_i, x_j) + g(x_i, x_j) \sum_{k=1}^{n} \rho(x_i, x_j, x_k)I(x_j, x_k)|S_k|, \tag{2.24}$$

where $|S_k|$ denotes an area of the patch S_k. By using matrix algebra, the same might be written as

$$I = G\epsilon + GMI, \tag{2.25}$$

where I and ϵ are column vectors of the size n^2 and G, M are matrices of the size $n^2 \times n^2$, where n is a number of surface patches in the scene. From (2.25) we may write

$$(\mathbf{1} - GM)I = G\epsilon, \tag{2.26}$$

where $\mathbf{1}$ is an identity matrix. Then we can write:

$$I = (\mathbf{1} - GM)^{-1} G\epsilon \tag{2.27}$$

$$= G\epsilon + GMG\epsilon + GMGMG\epsilon + (GM)^3 G\epsilon + \cdots. \tag{2.28}$$

The Neumann series on the right hand side of (2.28) converges under the condition that the spectral norm of the matrix GM is less then 1. The matrix GM represents the attenuation of light intensity due to an interaction with a surface, since a part of the radiant energy is absorbed.

Remark: It must be stressed, that (2.24)–(2.28) are only approximations of the infinite dimensional case, with the accuracy depending on the partitioning of the set S into patches S_i. The equations (2.24)–(2.28) are satisfied only approximately, due to the fact that the transport intensity $I(x_i, x_j)$ is not exactly equal to a constant I_{ij} for all $x_i \in S_i$ and $x_j \in S_j$. Also, $\rho(x_i, x_j, x_k)$ is not constant for all $x_i \in S_i$, $x_j \in S_j$ and $x_k \in S_k$. $\qquad \square$

The finite dimensional case is useful for the understanding of the problem. Due to the presence of an infinitely large number of interactions between the surface patches in an actual scene, the global illumination computation is a complex problem. Any computational framework can take into account only a finite number of them.

There are essentially two different ways to compute the illumination. The first one is to evaluate some terms of the Neumann series on the right hand side of (2.22) or (2.28), as it is done by *ray tracing* techniques. The second possibility is to solve the linear system of equations (2.26) with respect to vector I. Basically this idea is exploited by the *radiosity technique* for the computation of illumination of diffuse environments.

2.5   Ray-tracing

Ray tracing has been proposed by Whitted [79] as a technique for a realistic rendering of synthetic images of three-dimensional scenes.

The irradiance of a particular picture element (PEL) at the image plane is computed by tracing interactions in the scene significantly contributing to the illumination of the PEL. The interactions are evaluated starting from the image plane. From the PEL a *primary ray* is shot via the optical centre of the lens to the scene (see figure 2.8).   The first intersected object face is the one that is projected onto the PEL and determines its irradiance. The radiance of a surface patch S at the point of intersection in the direction towards the PEL (i.e. against the primary ray) is given by equation (2.7), and depends on the BRDF of the patch S and the radiance incident on this patch. Significant incidence directions are pursued by shooting rays into these directions. *Secondary rays* are shot in the direction to all point light sources illuminating the scene. If there is no occlusion between point S and the light source, then the irradiance of S caused by the light source and the resulting contribution reflected radiance are evaluated.

Next to this, the so called *recursive rays* are shot in the directions of reflected and refracted rays (see also figure 2.1, the refracted ray is shot only into a transparent medium). Each recursive ray is treated as if it were a primary ray, so when it intersects some other object face, the radiance of this intersection is evaluated by shooting other secondary rays to the light sources, and other recursive rays, until a certain recursion level limit is reached. All these rays recursively contribute to the irradiance of the PEL where the first primary ray was shot from.

The ray tracing as proposed by Whitted in [79] is an approximation of the
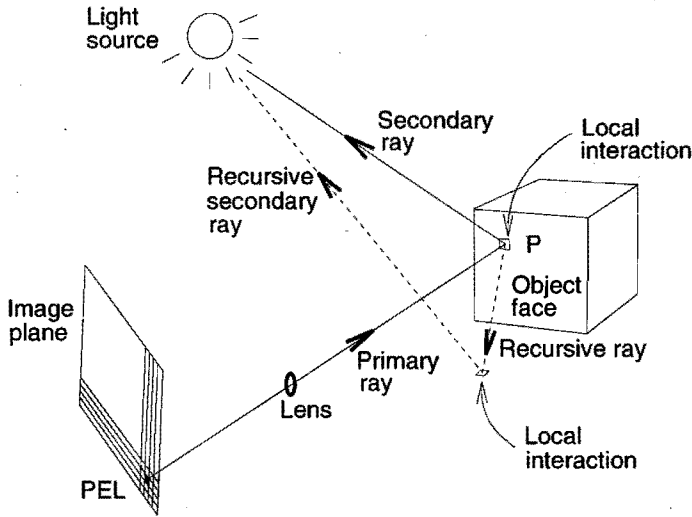
Figure 2.8: Ray-tracing technique.

identity (2.28) by

$$I \approx G\epsilon_0 + GM_0G\epsilon_0 + GM_1GM_0G\epsilon_0 + \cdots + (GM_1)^k GM_0G\epsilon_0, \qquad (2.29)$$

where M_0 is the approximation of the operator $\mathcal{M}$ by a sum over the light sources, M_1 is the approximation of the operator $\mathcal{M}$ by a sum over intersected patches in the reflected and refracted directions, G is the geometric factor which takes care of the proper casting of shadows, ϵ_0 is the intensity of light emitted by the point light sources, and k is the limit for the recursion level.

Obviously, only two-point transport intensities with incidence points at the image plane have to be computed. Therefore, only certain paths of rows and columns of matrices $M0, M1$ and G have to be evaluated. This makes ray-tracing view dependent, and the entire computation has to be repeated when the view point is changed.

The major drawback is that no directions other than reflected and refracted ones are taken into account when evaluating interactions. This makes ray tracing less suitable for a scene with diffuse surfaces where interreflections between object faces are significant.

2.6   Radiosity

Radiosity is a technique introduced by Goral, Torrance, Greenberg and Battaile [22] that solves the energy balance equations for the surfaces of the scene. Object surfaces in the scene are divided into sufficiently small but still macroscopic patches, so that the radiance in each single patch can be approximated by a constant value. A diffuse reflection of all surfaces is assumed with uniform radiance in all reflected directions. This allows to reduce the number of energy balance equations from n^2 to n, when compared with $(2.26)^5$.

Consider two surface patches as S_j, S_k as shown in figure 2.9. The radiosity of a
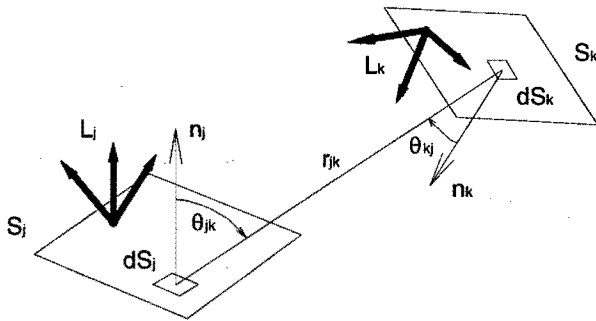
Figure 2.9: Radiosity.

surface is the total energy leaving the surface per second. We obtain an identity between radiosities by integrating equation (2.7) (p. 16) over the hemisphere of reflecting directions over the patch S_j:

$$L_j \int\limits_{S_j} \int\limits_{\Omega} \cos\theta_r \, d\omega_r dS_j = \int\limits_{S_j} \int\limits_{\Omega} \left(L_{ej} + \int\limits_{\Omega} \rho_{bd} L_i \cos\theta_i \, d\omega_i \right) \cos\theta_r \, d\omega_r \, dS_j.$$

$$(2.30)$$

From that we obtain

$$\pi S_j L_j = \pi S_j L_{ej} + \rho_{bd} \int\limits_{S_j} \int\limits_{\Omega} \int\limits_{\Omega} L_i \cos\theta_i \cos\theta_r \, d\omega_i \, d\omega_r \, dS_j. \qquad (2.31)$$

The inner integral over Ω in the right hand side can be replaced by a sum of integrals of all surfaces $S_k, k = 1, \ldots, n$ in the scene. Angle $d\omega_i$ for k-th path is

---

[5]Such reduction can be achieved only when the BRDF can be written as a product of two factors, one depending only on the incident direction, and another depending only on the reflected direction. This is possible for the Lambertian reflectance model, however impossible for the Torrance-Sparrow reflectance model.

written as $\cos\theta_{kj}\,dS_k/r_{jk}^2$, where r_{jk} is a distance. After changing the order of summations and integration, we obtain

$$\pi S_j L_j = \pi S_j L_{ej} + \pi\rho_{bd}\sum_{k=1}^{n}L_k\int\limits_{S_j}\int\limits_{S_k}\frac{\cos\theta_{jk}\cos\theta_{kj}}{r_{jk}^2}dS_k\,dS_j. \qquad (2.32)$$

The surface patch radiances L_j are the solution of the linear system

$$L_j = L_{ej} + \rho_j\sum_{k=1}^{n}L_k K_{kj} \quad \text{for } j = 1,\ldots,n, \qquad (2.33)$$

where K_{kj} are the so called *form factors* given by

$$K_{kj} = \frac{1}{S_j}\int\limits_{S_j}\int\limits_{S_k}\frac{\cos\theta_{jk}\cos\theta_{kj}}{r_{jk}^2}dS_k\,dS_j. \qquad (2.34)$$

Once the system (2.33) is solved, it is not difficult to obtain different views of the scene: radiosity is a view-independent technique. A serious drawback of the radiosity is that it handles only diffuse environments. Immel, Cohen and Greenberg [44] applied the radiosity technique for non-diffuse environments by directly solving equation (2.26). However, the computations are tractable only for the most simple scenes with a low resolution.

Hybrid techniques combining ray tracing and radiosity have been proposed to remove their shortcomings ([10], [74]). Better results are obtained at the expense of a higher computational burden.

2.7 Stochastic ray tracing

Both radiosity and ray tracing are examples of an approximation to the solution of the rendering equation (2.21). They neglect various optical phenomena to yield an tractable solution. An alternative to these approximations is to evaluate the solution given by the right hand side of (2.22) directly by a method of statistical experiment, by a *Monte Carlo technique*. This so called *stochastic ray tracing*, or *distributed ray tracing*, receives much attention [10, 45, 66, 7, 77], as it seems to be superior to conventional ray tracing, radiosity, and their hybrids. It puts no restrictions on the scene reflectances and on the lighting, all ways of energy transport are taken into account, and attention is paid to an appropriate spending of computational effort.

2.7.1   Monte Carlo technique

For an introduction to the Monte Carlo technique we refer to [24]. The Monte Carlo technique is generally defined as representing the solution of a problem as a parameter of a hypothetical population, and using a random sequence of numbers to construct a sample of the population, from which statistical estimates of the parameter can be obtained.

The *Riemann integral*

$$I = \int_a^b g(x)f(x)\,dx \qquad (2.35)$$

can be interpreted as the expected value of a random variable $g(X)$, where X is a random variable with the probability density function (p.d.f.) $f(x)$. Therefore, $f(x)$ must have the obvious properties of the p.d.f., i.e. $f(x) \geq 0$ for $x \in R$, and

$$\int_{-\infty}^{\infty} f(x)\,dx = 1.$$

If k independent samples $x_1, \ldots, x_k$ of variable X are taken, the arithmetic mean

$$\hat{I}_k = \frac{1}{k} \sum_{i=1}^{k} g(x_i) \qquad (2.36)$$

is an estimate of the expected value of $g(X)$, and thus of the integral I. The variance of this estimate is [24]

$$\mathrm{var}\,\hat{I}_k = \frac{1}{k} \left[\int_a^b g(x)^2 f(x)\,dx - I^2 \right], \qquad (2.37)$$

and if $\mathrm{var}\,\hat{I}_1 < \infty$, then the sequence $\{\hat{I}_k\}_{k=1}^{\infty}$ converges in quadratic mean, in probability, and with probability one, to I, as $k \to \infty$. In this way integral I can be evaluated to a desired precision by collecting a sufficient number of samples of $g(X)$.

In a similar way, sums of finite or infinite series can be computed. The method is useful when $g(x)$ is not known analytically, and $g(x_i)$ is a result of a complex computation or of an experiment, and standard analytical or numerical techniques can not be used. The convergence of the estimate $\hat{I}_k$ can be improved by *variance reduction techniques*. We will mention *stratified sampling*.

2.7.1.1   Stratified sampling

Stratified sampling, or systematic sampling [24] is a method of sampling in which the interval $[a, b]$ is partitioned into m disjoint subintervals S_j, $j = 1, \ldots, m$,

such that $[a, b] = \cup_{i=1}^{n} S_i$, and $S_i \cap S_j = \emptyset$ for $i \neq j$. The integral I is the sum of integrals over the subintervals,

$$I = \sum_{j=1}^{m} \int_{S_j} g(x)f(x)\, dx = \sum_{j=1}^{m} I_j,$$

and each subintegral is estimated by the Monte Carlo technique separately. Under the original density function $f(x)$, the probability that the sample is coming from the subinterval S_j is

$$P_j = \int_{S_j} f(x)\, dx.$$

Particularly, subdivision can be done so that these probabilities are the same for all subintervals, $P_j = 1/m$, $j = 1, \ldots, m$, and only one sample is taken from each subinterval. Consequently, from each subinterval S_j a sample x_j is drawn using the probability density function $f_j(x)$:

$$f_j(x) = \begin{cases} mf(x) & \text{for } x \in S_j, \\ 0 & \text{for } x \notin S_j, \end{cases}$$

and the value $g(x_j)/m$ is used as the estimate of I_j. Hence, the estimate of I is

$$\hat{I}_m = \frac{1}{m} \sum_{j=1}^{m} g(x_j).$$

The variance of this estimate is [24]

$$\text{var } \hat{I}_m = \sum_{j=1}^{m} \left[\frac{1}{m^2} \int_{S_j} g(x)^2 mf(x)\, dx - I_j^2 \right] = \frac{1}{m} \int_a^b g(x)^2 f(x)\, dx - \sum_{j=1}^{m} I_j^2,$$

which is always less then or equal to the variance (2.37) for the same number of samples, $k = m$, because from the Schwarz inequality we have

$$\sum_{j=1}^{m} I_j^2 \geq \frac{I^2}{m}.$$

From that it follows, that it is always more efficient to use a finer partition of the interval $[a, b]$ and to take one sample from each subinterval, than to increase the number of samples in the intervals.

2.7.2   Application to the rendering equation

Stochastic ray tracing essentially consists of a multi-stage Monte Carlo integration. The integral

$$I = \int_S \rho(x_1, x_2, x_3)\, g(x_2, x_3)\, [\epsilon(x_2, x_3) + I(x_2, x_3)]\, dx_3, \qquad (2.38)$$

which repeatedly occurs in the Neumann's series in (2.22), can be split into three separate integrals I_1, I_2 and I_3, by using the identity

$$\rho(x_1, x_2, x_3) = \cos\theta_1 \cos\theta_2 [\rho_{\text{dif}} + \rho_{\text{spec}}(\theta_r, \phi_r, \theta_i, \phi_i)],$$

where ρ_{dif} is a direction independent part of the BRDF, and $\rho_{\text{spec}}()$ is a direction dependent part of the BRDF (cf. (2.19),(2.14)). Then

$$I = I_1 + I_2 + I_3,$$

where

$$I_1 = \int_{\mathcal{S}} \rho(x_1, x_2, x_3)\, g(x_2, x_3)\, \epsilon(x_2, x_3)\, dx_3 \qquad (2.39)$$

$$I_2 = \cos\theta_1 \cos\theta_2\, \rho_{\text{dif}} \int_{\mathcal{S}} g(x_2, x_3)\, I(x_2, x_3)\, dx_3 \qquad (2.40)$$

$$I_3 = \cos\theta_1 \cos\theta_2 \int_{\mathcal{S}} \rho_{\text{spec}}(\theta_r, \phi_r, \theta_i, \phi_i)\, g(x_2, x_3)\, I(x_2, x_3)\, dx_3. \qquad (2.41)$$

I_1 represents the part of the two-point transport intensity $I(x_1, x_2)$, that is a reflection of light coming from the light sources directly, I_2 is the part of light coming from other surfaces (an indirect illumination) reflected by a diffuse reflection mechanism, and I_3 is the part of light coming from other surfaces reflected by a mechanism of specular reflection. Integrals I_1, I_2 and I_3 are estimated by the Monte Carlo technique.

2.7.2.1   Sampling light sources

Small light sources can be considered as point light sources. Assume that N point light sources, each with a radiant flux Φ_i, are placed at positions Z_i, $i = 1, \ldots, N$, in the scene. Also assume, that area light sources (extended light sources) occupy the surface area A_L, that is

$$A_L = \{x \in \mathcal{S} \mid \epsilon(\cdot, x) \neq 0\}.$$

Integral I_1 can be rewritten as

$$I_1 = \sum_{i=1}^{N} g(x_2, Z_i)\frac{\Phi_i}{4\pi} \cos\theta_i + \int_{A_L} \rho(x_1, x_2, x_3)\, g(x_2, x_3)\, \epsilon(x_2, x_3)\, dx_3. \qquad (2.42)$$

The contribution of the point light sources can be exactly evaluated, while the area sources are sampled using stratification. The area light sources might be partly occluded by other objects in the scene, and evaluation of the integral on

the right hand side of (2.42) amounts to the random sampling of points x_3 on the surface area A_L using stratification, with the probability density proportional to

$$\frac{\rho(x_1, x_2, x_3)\, \epsilon(x_2, x_3)}{|x_2 - x_3|^2}.$$ (2.43)

Often, (2.43) is approximated by a constant. For the sampled points, the visibility factor $g(x_2, x_3)|x_2 - x_3|^2$ is evaluated. For adaptive techniques of sampling of light sources we refer to [48].

2.7.2.2 Sampling the illuminating hemisphere

The domain of integration of integrals I_2 and I_3 can be changed into an illuminating hemisphere Ω above the surface patch x_2. Hence

$$I_2 = \cos\theta_1 \cos\theta_2\, \rho_{\text{dif}} \int_\Omega L_i(\theta_i, \phi_i) \cos\theta_i\, d\omega$$ (2.44)

$$I_3 = \cos\theta_1 \cos\theta_2 \int_\Omega \rho_{\text{spec}}(\theta_r, \phi_r, \theta_i, \phi_i)\, L_i(\theta_i, \phi_i) \cos\theta_i\, d\omega,$$ (2.45)

Integrals I_2 and I_3 can be estimated using stratified sampling [77], [7]. By sampling the illumination hemisphere we mean a random generation of a direction into the hemisphere Ω, shooting a ray in that direction and evaluation of the surface radiance $L(\theta_i, \phi_i)$ at the point of intersection with another object. In both I_2 and I_3 the sampled function $L_i(\theta_i, \phi_i)$ corresponds to $f(x)$ in (2.35), and the terms $\cos\theta_i$, and $\rho_{\text{spec}}(\cdot, \theta_i, \phi_i)\cos\theta_i$ respectively, correspond after normalization to a density function $f(x)$. For subdivision of the integration domain and normalization of the density function we refer to [7], [77].

Ward, Rubinstein and Clear [77] claim that the amounts of indirect illumination computed by (2.44) do not change rapidly, and they can be reused for neighbouring points and for different views (thanks to the directional independence) by the algorithm they propose.

2.7.2.3 Termination of rays

At each surface patch sampled by the ray during Monte Carlo integration, new rays are shot in directions to the light sources and to other surfaces surrounding the patch. In this way a tree of rays is created. The rays are terminated at the light sources, or if the ray does not hit any other surface and leaves the system. The illumination computation algorithm must provide a way of early termination of the rays, to prevent tracing rays bringing only a negligible contribution to the total illumination.

This can be done either by terminating rays after a fixed number of bounces to surfaces, or by setting a threshold for the contribution of the ray to the total illumination of the picture element [23]. The ray is terminated if its contribution is below this threshold. In this way the remaining terms of the evaluated series are neglected, and the illumination is underestimated, since the neglected terms are always positive. This bias can be made as small as possible at the expense of increased computational effort.

Arvo and Kirk [1] proposed a technique, that removes the bias by slightly rising the variance, known as *Russian roulette*. Suppose the weight of the ray contribution is w. According to this technique, if w is smaller than some threshold w_t, then with the probability P the ray is terminated, or alternatively with the probability $1 - P$ it is continued with the increased weight $w/(1 - P)$. It is easy to show, that the expected weight of the ray is not changed

$$E[w] = P \cdot 0 + (1 - P)\frac{w}{1 - P} = w,$$

where E denotes a random variable expectation operator.

2.8   The choice of the imaging model

After considering several aspects of the imaging models which are listed below, we have selected stochastic ray tracing as an appropriate imaging model:

- Stochastic ray tracing is a view dependent technique. Only the illumination of that part of the scene which contributes significantly to the irradiation of the image plane is explored. This is achieved by tracing the rays starting from the camera optical centre. Only a selected image area, or a single PEL can be computed within an amount of time proportional to the number of PELs. The user has a control over the tradeof between the accuracy and complexity of the model by changing the numbers of randomly generated rays used to sample the illumination hemisphere. Plain ray tracing can then be obtained as a special case of stochastic ray tracing. However, when the view point (a camera position) is changed, the global illumination computation must be repeated from scratch.

- Stochastic ray tracing can be combined with an arbitrary surface reflectance model. The ray tracer and the surface reflection model are to a great extent independent from each other, which simplifies the analysis and the implementation of the imaging model.

- In stochastic ray tracing the computed image irradiance is an explicit function of the reflectance model parameters, and a linear function of the light

sources intensities. An influence of the parameters can be easily investigated, (computation of partial derivative with respect to the parameters). This simplifies the parameter estimation.

- The application of radiosity, an alternative to the stochastic ray tracing, is limitation Lambertian surfaces. This is possibly the major drawback when compared with stochastic ray tracing. Radiosity is also less suitable for th estimation of the parameters. The complexity of hybrid models combining ray tracing and radiosity is high, making application, parameter estimation, *etc.* difficult.

Some aspects of stochastic ray tracing are demonstrated in figure 2.10. In figure 2.10a is an actual image of the scene acquired by a camera. In figures 2.10b, 2.10c and 2.10b, are images of the same scene obtained respectively ba a (plain) ray tracing, stochastic ray tracing, and once more stochastic ray tracing with an increased number of sampling rays. The quality of image in figures 2.10d is better than that of figure 2.10c. It can be seen, that the plain ray tracing fails to model the left side of the vertical plate of T-shaped steel profile correctly. It underestimates the illumination of this vertical plate as it does not take into account an indirect illumination from other objects in the environment (e.g. from a wall which is located left to the scene and is not seen in the figure).

The summary of the imaging model using stochastic ray tracing with the Torrance-Sparrow surface reflectance model is given in Appendix A.

(a)   (b)

(c)   (d)
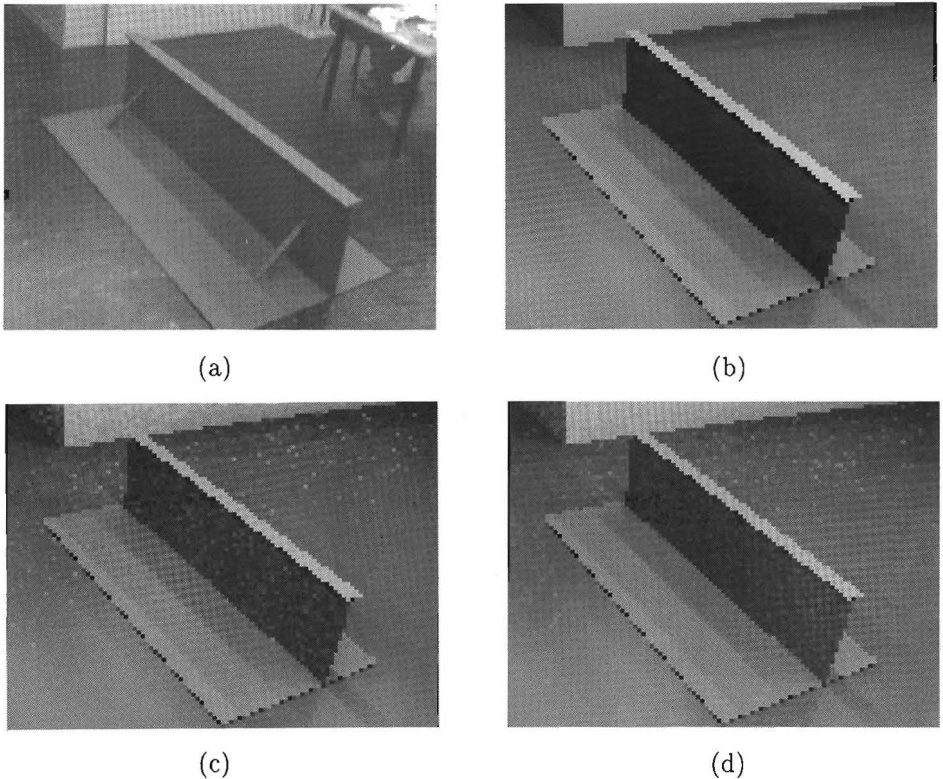
Figure 2.10: (a) An image of a steel construction. (b) A synthetic image obtained by ray tracing. (c), (d) Synthetic images obtained by stochastic ray tracing using different numbers of rays to sample the illuminating hemisphere. In (c) 2 rays, and in (d) 8 rays were used. Notice the left side of the vertical plate. This plate is indirectly illuminated by the wall left to the scene (not contained in the image), and also by other objects in the scene. Ray tracing underestimates the illumination of the vertical plate, while stochastic ray tracing does not. The noise which can be observed in images obtained by stochastic ray tracing diminishes with the increasing number of sampling rays.

Chapter 3

Parameter estimation

In the previous chapter, the model of the imaging process has been developed. This model allows us to compute the irradiance of the image plane and so to construct an image, if scene, lighting and camera properties are completely specified. Next to the scene and lighting topology, the reflectance parameters of the scene surfaces, as well as the radiance of the light sources are essential when using the imaging model. In this chapter, an estimation of these parameters from the image plane irradiance of the available camera images is investigated.

3.1 Estimation of the surface reflectance in computer vision

The bidirectional reflectance distribution function (BRDF), which characterizes the reflectance properties of a surface, is a function of two directions (the directions of incident and reflected light), and so a function of four angles (the zenith and azimuth angles of the directions). In case of isotropic surfaces, the BRDF is a function of three angels, two zenith angles and the difference between the azimuth angles. The BRDF can be measured by a *gonioreflectometer* (see figure 3.1), where all angles, the arguments of BRDF, can be adjusted separately. The arrangement shown in figure 3.1 has been designed by Murray-Coleman and Smith [57]. The zenith angle of incident light can be set by moving the light source along the source driver hoop, the zenith angle of reflected light with the use of the detector hoop, and both azimuth angles can be adjusted by rotating the sample and the detector hoop. The system can be controlled by a computer. The measurement over the entire domain of the BRDF by the gonioreflectometer takes hours and is neither practical, nor necessary for our
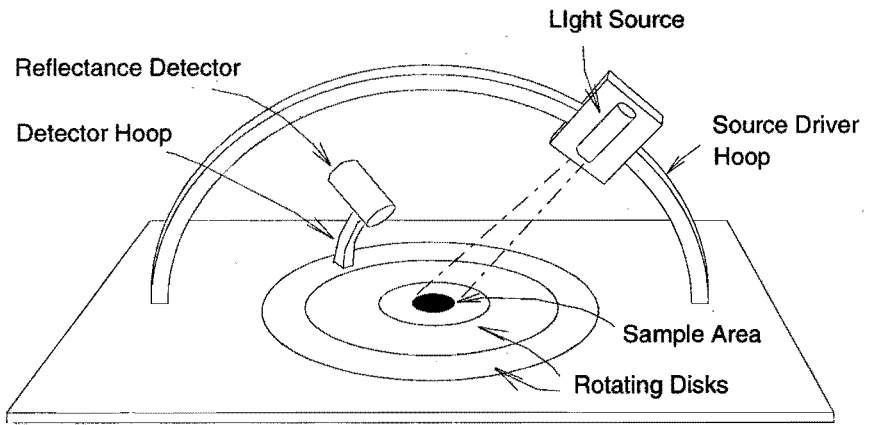
Figure 3.1: Gonioreflectometer.

computer vision application. However, the measurement results may serve as a reference for testing computer vision algorithms. Other arrangements can be found in the literature as well [76].

In computer vision, a surface reflectance model is required when the relation between the 3-D scene and the image plane irradiance is exploited. Then the problem of estimating the BRDF has to be solved. By using a reflectance model, the problem of the BRDF estimation is reduced to the estimation of the reflectance model parameters. The surface reflectance, and the closely related *reflectance map* is exploited by the *shape from shading* and *photometric stereo* techniques. The concept of reflectance maps gives a good insight into the problem of reflectance parameter estimation in photometric stereo. Consider the imaging geometry in figure 3.2. In contrast to the camera model from section 2.3, a parallel projection is assumed. Parallel projection is a good approximation of central projection for a large distance between the optical centre and the image plane, and as a consequence we may assume that the viewing direction $\vec{v}$ is constant over the entire scene as projected on the image plane. We will also assume, that all light sources are far enough, so that the incident directions $\vec{l}$ are constant for the entire scene. In the camera centred coordinate system as shown in figure 3.2, i.e. with z-axis parallel to the projecting rays, the surface projected onto the image plane can be represented by a function $f(x,y)$. For a given uniform isotropic material, and a given distribution of distant light sources, the brightness of a surface element and the irradiance of the corresponding image plane point depend on the orientation of the surface normal $\vec{n}$ only, and is described by a function called the *reflectance map*. For the
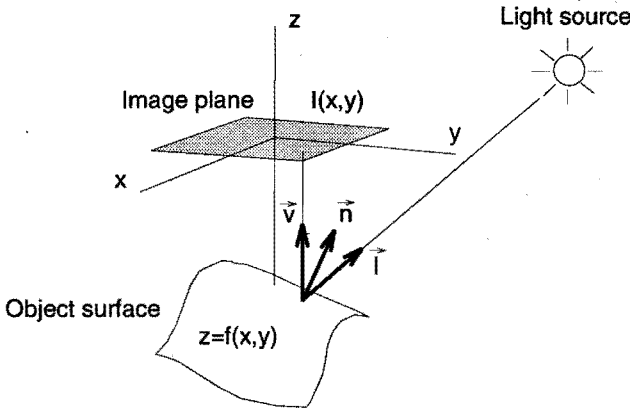
Figure 3.2: The imaging geometry for shape from shading.

image plane irradiance $E(x, y)$ at the position(x, y) we obtain the equation

$$E(x, y) = R(\vec{n}(x, y)), \tag{3.1}$$

that constrains the orientation of the surface normal $\vec{n}$ at point $(x, y, f(x, y))$. It is not difficult to obtain the reflectance map R from the BRDF of the surface. Alternatively it can be determined experimentally from an image of an object with a suitable and known shape. Compared to the BRDF, the reflectance map captures the entire imaging situation including the lighting. That simplifies further reasoning and calculations.

The aim of the shape from shading technique introduced by Horn [36] is to compute the shape of the continuous, smooth surface $z = f(x, y)$ from a single image acquired under known conditions (the reflectance map is known). As the surface orientation $\vec{n}$ has two degrees of freedom, and (3.1) provides only one constraint, it is not possible to compute $\vec{n}$ locally at each point, and global constraints must be used. Horn determines the change of the surface gradient (the second derivative of the function f), and uses the fact, that for a smooth surface its integral over any closed path is zero. A precise mathematical treatment and other global constraints can be found in [80].

In photometric stereo, more images of the surface obtained under different distributions of lights are processed. For all images, the position of the camera with respect to the object remains unchanged, and a specific point on the image plane corresponds to the same surface patch in all images. For instance, two images provide two independent equations at each image point:

$$E_1(x, y) = R_1(\vec{n}(x, y)), \tag{3.2}$$

$$E_2(x,y) = R_2(\vec{n}(x,y)).$$

Here, subscripts identify the image, and the corresponding reflectance map. With two equations at each image point we are able to solve for the surface orientation locally. In general, equations (3.2) are nonlinear and might lead to more solutions at each point. The ambiguities can be resolved by using three or more images (than a least squares solution must be computed, since the system is overdetermined), or by employing global constraints.

With a sufficient number of images acquired under different lighting conditions, the redundancy can be used to estimate the surface reflection properties, and the photometric stereo method can deal with nonuniform surfaces. In this case the reflectance map will vary across the surface. Equation (3.1) can be modified by adding one or more parameters of the reflectance map that are functions of the image coordinates, and in this way model the surface nonuniformity. Hence, for each image point we obtain equations

$$E_j(x,y) = R_j(\vec{n}(x,y), \theta(x,y)) \qquad j = 1, \ldots, k, \qquad (3.3)$$

where θ is a vector of parameters and k is the number of acquired images. Silver [68] studied the problem of simultaneous estimation of surface shape and the reflectance properties for reflectance maps in a form $R(\vec{n}, \theta_1) = \theta_1 R'(\vec{n})$, and $R(\vec{n}, \theta_1, \theta_2) = \theta_1 [\theta_2 R'(\vec{n}) + (1 - \theta_2) R''(\vec{n})]$. The later model suits well for a number of surfaces, where R' represents the Lambertian part of a reflection, and R'' the specular part. For a more general reflectance model, the problem has been treated by Tagare and deFigueiredo [71]. They observed that the problem is of an ill posed nature and they gave an indication for its regularization.

When the shape of an object is known, surface reflectance properties can be determined from a single image (unlike the photometric stereo method). Ikeuchi and Sato [43] proposed a technique for the reflectance estimation from an intensity image and a range image. From the range image the surface normals are determined at each point, and the unknown reflectance model parameters are estimated such that for all image points (x, y)

$$E(x,y) \approx R(\vec{n}(x,y), \theta). \qquad (3.4)$$

The strict equality can not usually be achieved due to random perturbations of the observed images, and a least-squares fitting is employed. In [43] the Torrance-Sparrow reflectance model and a simple imaging geometry as shown in figure 3.2 have been used. The technique has been refined by Kay and Caelli [46]. This technique is the one most related to the technique we propose in the remaining part of this chapter. However, the parameterized reflectance map $R(\vec{n}(x,y), \theta)$ is in our case replaced by the parameterized imaging model developed in the previous chapter.

3.2 Estimation of the scene parameters

We will discuss the problem of the estimation of surface reflectance parameters of objects in the scene and of the radiances of light sources illuminating the scene, when the topology of the scene and its illumination are known, and a camera image (camera images) of the scene is (are) available. We make assumptions that

- surfaces of objects in the scene are homogeneous, so that a single reflectance must be estimated for each object, and

- extended light sources (area light sources, i.e. not point light sources) emit an uniform diffuse light over the whole surface, so that a single parameter, the light source radiance, is to be estimated for each extended light source.

Using the model of the image formation process developed in chapter 2, an estimate of the image plane irradiance can be computed. This estimate is a function of the parameters to be estimated, and due to the character of the model, it can be considered to be a random variable. The parameters can be estimated by the *maximum likelihood* method known from statistics.

Let a random vector Y depend on an unknown parameter vector θ. Let $f_Y(y|\theta)$ be the conditional probability density function of Y given the parameter vector θ, also called a *likelihood function*. Based on an actual observation y, the realization of the random vector Y, we want to obtain an estimate of θ. A maximum likelihood estimate is defined as the value of θ for which $f_Y(y|\theta)$ attains a maximum. Thus $\hat{\theta}_{\mathrm{ML}}$ is a maximum likelihood estimate if it satisfies

$$f_Y(y|\hat{\theta}_{\mathrm{ML}}) \geq f_Y(y|\theta) \qquad \text{for all } \theta \in \Theta, \qquad (3.5)$$

where Θ is the feasible set of θ, the parameter space. Statistical properties of the estimate $\hat{\theta}_{\mathrm{ML}}$ can be investigated, thus providing an accuracy analysis of the estimate. For the sake of conciseness we will drop the index ML in $\hat{\theta}_{\mathrm{ML}}$ in the sequel.

The conditional probability density function for the problem of the scene parameter estimation might be a complicated function. However it can sufficiently well be approximated by a density of the normal distribution. The mean and variance of the distribution can be estimated using the imaging model. The likelihood function has the form

$$f_Y(y|\theta) = \frac{1}{(2\pi)^{n/2}|Q|^{1/2}} \exp\left\{-\frac{1}{2}[h(\theta) - y]^t Q^{-1}[h(\theta) - y]\right\}, \qquad (3.6)$$

where y is the vector of observed image irradiances, $h(\theta)$ are the irradiance values computed by the model, Q is the covariance matrix of the imaging model

error $h(\theta) - Y$, and n is the length of vector y. The maximum likelihood estimate of parameter vector θ is then obtained by maximization of the likelihood function,

$$\hat{\theta} = \arg \max_{\theta \in \Theta} f_Y(y|\theta). \tag{3.7}$$

Since a logarithm is an increasing function, (3.7) is equivalent to

$$\hat{\theta} = \arg \max_{\theta \in \Theta} \{\log f_Y(y|\theta)\}, \tag{3.8}$$

or to the minimization

$$\hat{\theta} = \arg \min_{\theta \in \Theta} \left\{ [h(\theta) - y]^t Q^{-1} [h(\theta) - y] \right\}. \tag{3.9}$$

This is the non-linear weighted least-squares estimation. The feasible set Θ is determined by physical limits for parameter values, and is given by linear inequality constraints. For instance, a coefficient of diffuse reflection is constrained by the inequalities

$$0 \leq k_{\text{dif}} \leq \frac{1}{\pi}. \tag{3.10}$$

In each inequality only one parameter is involved, and this particular case is called a problem with *simple bounds*.

Solving (3.9) is a well-defined problem which always has a solution. Numerical techniques have to be used to find this solution, since $h(\theta)$ is a nonlinear function of the unknown parameters. Due to the large size of the vector y which is equivalent to the number of picture elements of the processed image, techniques that efficiently utilize the resources (computing time and memory) are used. Despite the clear formulation, the solution of problem (3.9) might be ill-conditioned, or it might have an infinite number of solutions. This depends on e.g., the scene, available image(s), a priori knowledge *etc*. The possibilities for detecting ill-conditionedness are discussed later.

3.2.1   Numerical solution of the minimization problem

For the essential principles, and a good reference of numerical optimization techniques we refer to [21]. We will concentrate only on aspects specific for the problem at hand. The technique chosen for the solution is known as the Gauss-Newton method. The Gauss-Newton method uses the second derivative of the objective function, and exhibits an asymptotically fast, in fact quadratic, convergence to the solution.

The problem is to find the parameter vector $\hat{\theta} \in \Theta$ that minimizes an objective function

$$\hat{\theta} = \arg\min_{\theta \in \Theta} \left\{ [h(\theta) - y]^t Q^{-1} [h(\theta) - y] \right\}. \tag{3.11}$$

The covariance matrix Q is positive definite, and therefore there always exists a real matrix S such that $S'S = Q^{-1}$. The matrix S can often be directly computed from the model, instead of the matrix Q. Hence, the minimization problem is

$$\hat{\theta} = \operatorname*{argmin}_{\theta \in \Theta} \left\{ [h(\theta) - y]^t S'S [h(\theta) - y] \right\}, \qquad (3.12)$$

or equivalently

$$\hat{\theta} = \operatorname*{argmin}_{\theta \in \Theta} \| S\, h(\theta) - S\, y \|_2, \qquad (3.13)$$

where $\|\cdot\|_2$ denotes an Euclidean vector norm. Suppose that before the $(n+1)$-th iteration step, the parameter vector estimate θ_n is available (the initial estimate is denoted by θ_0). In the vicinity of point θ_n the nonlinear function $h(\theta)$ can be approximated by the first two elements of the Taylor series:

$$h(\theta) \approx h(\theta_n) + \nabla_\theta h(\theta_n)(\theta - \theta_n), \qquad (3.14)$$

where $\nabla_\theta h(\theta)$ is the Jacobian, the matrix of derivatives, i.e. $(\nabla_\theta h(\theta_n))_{ij} = \left. \frac{\partial h_i(\theta)}{\partial \theta_j} \right|_{\theta=\theta_n}$.

When this approximation is substituted into (3.13), we obtain an equation for the updated parameter vector estimate,

$$\theta_{n+1} = \operatorname*{argmin}_{\theta \in \Theta} \| S\, h(\theta_n) + S\, \nabla_\theta h(\theta_n)(\theta_{n+1} - \theta_n) - S\, y \|_2, \qquad (3.15)$$

hence, the parameter vector update $\Delta\theta_{n+1} = \theta_{n+1} - \theta_n$ is the solution of the overdetermined[1] linear system

$$S\, \nabla_\theta h(\theta_n)\, \Delta\theta_{n+1} = S\, (y - h(\theta_n)), \qquad (3.16)$$

in the least-squares sense. Taking into account a large length of the vectors y and $h(\theta)$ (the number of their components is equal to the number of processed image pixels), system (3.16) can be effectively solved by QR-decomposition using Givens rotations. Thereby, only the coefficients of a relatively small system

$$R\, \Delta\theta_{n+1} = b, \qquad (3.17)$$

with R an upper triangular matrix, are kept in the memory. Rows of the vector equation (3.16) are sequentially processed, and the system (3.17) is updated. The description of the technique at an expository level can be found in [13].

In addition to the parameter update computation, the validity of the constraints must be examined. When the value $\theta_{n+1} = \theta_n + \Delta\theta_{n+1}$, computed from (3.15) is outside of the feasible set Θ, the largest $a < 1$ is found, such that

$$\theta_n + a\Delta\theta_{n+1} \in \Theta, \qquad (3.18)$$

---

[1]The system is overdetermined under the assumption, that the number of observations is greater than the number of parameters. In our case, the number of observations is equal to the number of picture elements of the image used for the estimation, and it is mostly large enough to overdetermine the system.

and the value above becomes the new updated parameter vector value. In this way at least one constraint becomes *active*, i.e. a strict equality in the constraint is satisfied. This is taken into account when the next update is computed. In each new update, the parameters either stay inside the subspace determined by all active constraints (i.e. the computed updated is projected onto this subspace), or some of the active constraints become inactive. A constraint becomes inactive, if the component of the update corresponding to the constraint is in the direction inside the feasible set Θ. Then the strict equality in the constraint does not hold anymore, however, the constraint still holds. Care must be paid to the correct handling of constraints, however this does not present any extensive computational effort as all constraints are simple bounds.

3.2.2   Accuracy analysis

The scene parameter estimates $\hat{\theta}$ are computed from the observation vector y using a maximum likelihood estimator (3.7) through (3.9) The observation vector y is a realization of a random vector, and so is the estimate $\hat{\theta}$. We will analyze the statistical properties of this estimate. Besides the effect of observation noise, the inaccuracy of the obtained result can also be caused by the numerical procedure used to solve the problem (3.9). This is due to the limited accuracy of the numbers represented by the computer and the limited accuracy of computations. We refer to [21] for the discussion of the inaccuracy of the numerical solution to (3.9). It turns out that the inaccuracy of the numerical procedure is orders of magnitude smaller than the error caused by observation noise, and we will neglect it in the sequel.

Assume for a while, that the computed estimate $\hat{\theta}$ is an interior point of the feasible set Θ, and none of the parameters reaches its bounds. $F(y, \theta)$ denotes the function that is optimized in (3.7), (3.8), or (3.9). Without loss of generality, any of (3.7), (3.8), and (3.9) can be investigated, since they all yield the same result. $F(y, \theta)$ (the likelihood $f_Y(y, \theta)$, or its logarithm, *etc.*) can be approximated in the vicinity of the point $(y_{\text{obs}}, \hat{\theta})$ by its Taylor series expansion. If it is not otherwise explicitly stated, partial derivatives[2] of function F are taken at the point $(y_{\text{obs}}, \hat{\theta})$, therefore we will use only an abbreviated notation, i.e.

---

[2]We will use the convention that the partial derivative of a scalar function $F(y) = F(y_1, \ldots, y_n)$ is a row vector $\frac{\partial F}{\partial y} = \left[\frac{\partial F}{\partial y_1}, \ldots, \frac{\partial F}{\partial y_1} \right]$, and the mixed partial derivate of a scalar function $F(y_1, \ldots y_n, \theta_1, \ldots, \theta_m)$ is a matrix defined as

$$
\frac{\partial^2 F(y, \theta)}{\partial \theta \partial y} =
\begin{bmatrix}
\frac{\partial^2 F}{\partial \theta_1 \partial y_1} & \cdots & \frac{\partial^2 F}{\partial \theta_1 \partial y_n} \\
\vdots & & \vdots \\
\frac{\partial^2 F}{\partial \theta_m \partial y_1} & \cdots & \frac{\partial^2 F}{\partial \theta_m \partial y_n}
\end{bmatrix}
$$

$\frac{\partial^2 F}{\partial \theta^2} = \frac{\partial^2 F(y_{\text{obs}}, \theta)}{\partial \theta^2}\Big|_{\theta=\hat{\theta}}$. The Taylor series expansion of function F is:

$$F(y, \theta) = F(y_{\text{obs}}, \hat{\theta}) + \frac{\partial F}{\partial y}(y - y_{\text{obs}}) + \frac{1}{2}(\theta - \hat{\theta})'\frac{\partial^2 F}{\partial \theta^2}(\theta - \hat{\theta}) +$$

$$+ \frac{1}{2}(y - y_{\text{obs}})'\frac{\partial^2 F}{\partial y^2}(y - y_{\text{obs}}) + (y - y_{\text{obs}})'\frac{\partial^2 F}{\partial y \partial \theta}(\theta - \hat{\theta}) +$$

$$+ O_3, \tag{3.19}$$

where the term O_3 consists of the third and higher-order powers (and mixed powers) of $(y - y_{\text{obs}})$ and $(\theta - \hat{\theta})$. In the expansion (3.19) the prime denotes the vector transpose. Notice, that in the expansion the term $\frac{\partial F}{\partial \theta}(\theta - \hat{\theta})$ is missing. That is because for $\theta = \hat{\theta}$ function $F(y_{\text{obs}}, \theta)$ attains its extreme with respect to θ, and the partial derivative $\frac{\partial F}{\partial \theta}$ vanishes.

Let us denote the actual value of the parameter vector by θ^*, and consider a noiseless observation y_0, which is obtained for an unbiased model as $y_0 = h(\theta^*)$. Obviously, for $\theta = \theta^*$ function $F(y_0, \theta)$ attains its extreme, which gives us the necessary condition

$$\frac{\partial F(y_0, \theta)}{\partial \theta}\Big|_{\theta=\theta^*} = 0. \tag{3.20}$$

By substituting the expansion (3.19) into the condition (3.20), we obtain the relation between the observed vector y_{obs} and the estimate $\hat{\theta}$:

$$\frac{\partial^2 F}{\partial \theta \partial y}(y_0 - y_{\text{obs}}) + \frac{\partial^2 F}{\partial \theta^2}(\theta^* - \hat{\theta}) + O_2 = 0. \tag{3.21}$$

Term O_2 consists of the second and higher order powers of $(y - y_{\text{obs}})$ and $(\theta - \hat{\theta})$. In the vicinity of the point $(y_{\text{obs}}, \hat{\theta})$ we neglect the term O_2, hence if $\frac{\partial^2 F}{\partial \theta^2}$ is invertible, we have

$$\hat{\theta} \approx \theta^* - \left[\frac{\partial^2 F}{\partial \theta^2}\right]^{-1}\frac{\partial^2 F}{\partial \theta \partial y}(y_{\text{obs}} - y_0). \tag{3.22}$$

For a function $F(y, \theta) = \frac{1}{2}[h(\theta) - y]^t Q^{-1}[h(\theta) - y]$ we have

$$\frac{\partial^2 F}{\partial \theta^2} = -J'Q^{-1}, \tag{3.23}$$

and

$$\frac{\partial^2 F}{\partial \theta \partial y} = G + J'Q^{-1}J, \tag{3.24}$$

where J is a Jacobian of the function $h(\theta)$ at $\theta = \hat{\theta}$, defined as $J_{ij} = \frac{\partial h_i(\theta)}{\partial \theta_j}\Big|_{\theta=\hat{\theta}}$. The matrix G is defined as

$$G = \sum_{k=1}^{n}{}^{k}H\, r_k'\, (h(\hat{\theta}) - y_{\text{obs}}), \tag{3.25}$$

where ${}^kH$ is the Hessian matrix of the k-th component of the function $h(\theta)$ at point $\theta = \hat{\theta}$, i.e. ${}^kH_{ij} = \frac{\partial^2 h_k(\hat{\theta})}{\partial \theta_i \partial \theta_j}$, and the row vector r'_k is the k-th row of the matrix Q^{-1}. For a small residual $h(\hat{\theta}) - y_{\text{obs}}$, G is mostly much smaller than $J'Q^{-1}J$, and can be neglected in (3.24). Then, the relation between y_{obs} and $\hat{\theta}$ has a particularly simple form

$$\hat{\theta} \approx \theta^* + \left(J'Q^{-1}J\right)^{-1} J'Q^{-1}(y_{\text{obs}} - y_0), \tag{3.26}$$

which is a linear transformation of the random variable Y. The mean $E(\hat{\theta})$ and the covariance $\text{cov}(\hat{\theta})$ of the estimate $\hat{\theta}$ are

$$E(\hat{\theta}) = \theta^*, \tag{3.27}$$

$$\text{cov}(\hat{\theta}) = \left(J'Q^{-1}J\right)^{-1}. \tag{3.28}$$

We recapitulate the three assumptions we have made:

- $\hat{\theta}$ is an interior point of the feasible set Θ,

- observation noise $y_{\text{obs}} - y_0$ is small, and

- residual $h(\hat{\theta}) - y_{\text{obs}}$ is also small.

Under these assumptions and in the case of normally distributed y, the distribution of $\hat{\theta}$ is also approximately normal, and it is completely specified by the mean and covariance given above.

The situation changes only slightly when some of the constraints are active and the computed parameter estimate lies on the boundary of the set Θ. Basically, those components of the vector $\hat{\theta}$ that reach the bounds are fixed, and the analysis of the remaining components of the vector has to be performed. Let $\hat{\theta}_{free}$ and θ^*_{free} be vectors containing the free (without active constraint) components of the vectors $\hat{\theta}$ and θ^* respectively. Similarly, let J_{free} be the Jacobian of the function $h(\theta)$ with respect to the free components of θ at point $\theta = \hat{\theta}$, defined as $J_{ij} = \frac{\partial h_i(\theta)}{\partial \theta_{free,j}}$. Then, by following the same line of reasoning as before, we obtain that

$$\hat{\theta}_{free} \approx \theta^*_{free} + \left(J'_{free}Q^{-1}J_{free}\right)^{-1} J'_{free}Q^{-1}(y_{\text{obs}} - y_0). \tag{3.29}$$

Values of fixed components $\hat{\theta}_{fixed}$ of the vector $\hat{\theta}$, i.e. the values of components with active constraints, are given by their respective bounds.

From the results (3.26) and (3.29) the following can be concluded. As in the relations there is an inverse of the matrix $J'Q^{-1}J$ (or $J'_{free}Q^{-1}J_{free}$), when

this matrix looses rank the error of the result $\hat{\theta}$ is unbounded. Matrix Q is a covariance matrix, that is usually a well-behaving positive definite matrix. For uncorrelated observation noise it is a diagonal matrix. Therefore, the problems of an unbounded error may be caused only by the deficiency of the column rank of the jacobian J (or J_{free}).

3.2.3   Coping with ill-conditionedness

Despite the clear formulation of the parameter estimation as an optimization problem, the solution of this problem might be ill-conditioned, and there might be an infinite number of solutions. Whether the problem is ill-conditioned or not depends on scene, distribution of light sources, available images, choice of parameters to be estimated, a priori knowledge, *etc.* Essentially, if the effect of the variations of the parameters can be observed in the image(s), a different effect for different parameters, then the parameters can be estimated and the problem is *well-conditioned.* We will call these parameters *excited parameters*, and we will call the configuration of the scene, light sources and cameras where all parameters are excited the *persistently exciting* configuration. When one or more parameters are not excited, i.e. they have no, or only a small (comparable with an effect of noise) effect on the acquired images, or two or more parameters have the same effect, then the parameter estimation problem is ill-conditioned. This leads to a rank deficient (or an ill-conditioned) Jacobian J from the previous section, and consequently to an unbounded, or unreasonably large solution. This problem can be solved by elimination of the not-excited parameters, as proposed in the sequel.

An example of an ill-conditioned problem is the estimation of the surface reflectance parameters of the Torrance-Sparrow reflectance model (section 2.2.4) under certain circumstances. The Torrance-Sparrow reflectance model is specified by four parameters, $k_{\text{dif}}, k_{\text{spec}}, \beta_{\text{h}}$ and n. The latter three parameters $k_{\text{spec}}, \beta_{\text{h}}$ and n specify the specular part of the reflection, and consequently these parameters have substantial effect only at and in the vicinity of places with highlights caused by a specular reflection. In the case that there are no specular highlights caused by the reflection of light sources on the surface, the problem becomes ill-conditioned. In this case it is reasonable to apply the Lambertian reflectance model with a single parameter k_{dif}. Similarly, the effect of the Fresnel term in the Torrance-Sparrow reflectance model, and thereby the effect of the refraction index n, often can not be observed. The Fresnel coefficient varies with the angle of incidence of the light. Since the refraction index and the Fresnel coefficient have significant effect only at, or in the vicinity of places with specular highlights, there is often not enough data for their estimation. Then, the Fresnel coefficient can be considered to be a constant and can be absorbed in the parameter k_{spec}, i.e. a simplified Torrance-Sparrow reflectance

model can be used.

The not-excited parameters are detected based on their value. On all parameters which can potentially be eliminated bounds are applied specifiying a sufficiently large interval for the actual values of the parameters. When some of these bounds are reached, i.e. the bound becomes an active bound, it is concluded that the parameter that reached the bound is not excited, and it is eliminated from the model.

3.3   Experimental results

The estimation of the missing parameters is demonstrated in the following example:

From an image shown in figure 3.3a, the parameters of the Torrance-Sparrow reflectance model of the floor, and of the T-shaped steel profile placed on the floor, have been estimated. The scene was illuminated by TL-tubes placed on the ceiling. The reflection of TL-tubes can be seen on the floor. The scene description including the positions of the TL-tubes was available. The intensity of the TL-tubes was estimated in advance using a white diffuse surface as a reference (a white paper, $k_{dif} = 0.318$). The values of parameters of a simplified Torrance-Sparrow reflectance model (a model with a constant Fresnel factor absorbed in the parameter k_{spec}) have been computed. For the T-shaped profile only the coefficient of diffuse reflection has been computed, and the rest of the parameters has been eliminated due to a fact that there were no data (no specular highlights) available for the estimation of these parameters. The computed parameter values are listed below:

<div style="text-align:center">

floor   T-stiffener

k_{dif} = 0.081 k_{dif} = 0.166

k_{spec} = 0.818

β_h = 5.3°

</div>

A synthetic image generated with the computed parameter values, and the difference image between the original image and this synthetic image are shown in figures 3.3b and 3.3c.

Other examples of the estimation of the missing scene parameters, the original and the resulting synthetic images, are given in section 4.3, where the parameter estimation is routinely applied as a part of the scene description verification algorithm.

(a)   (b)

(c)
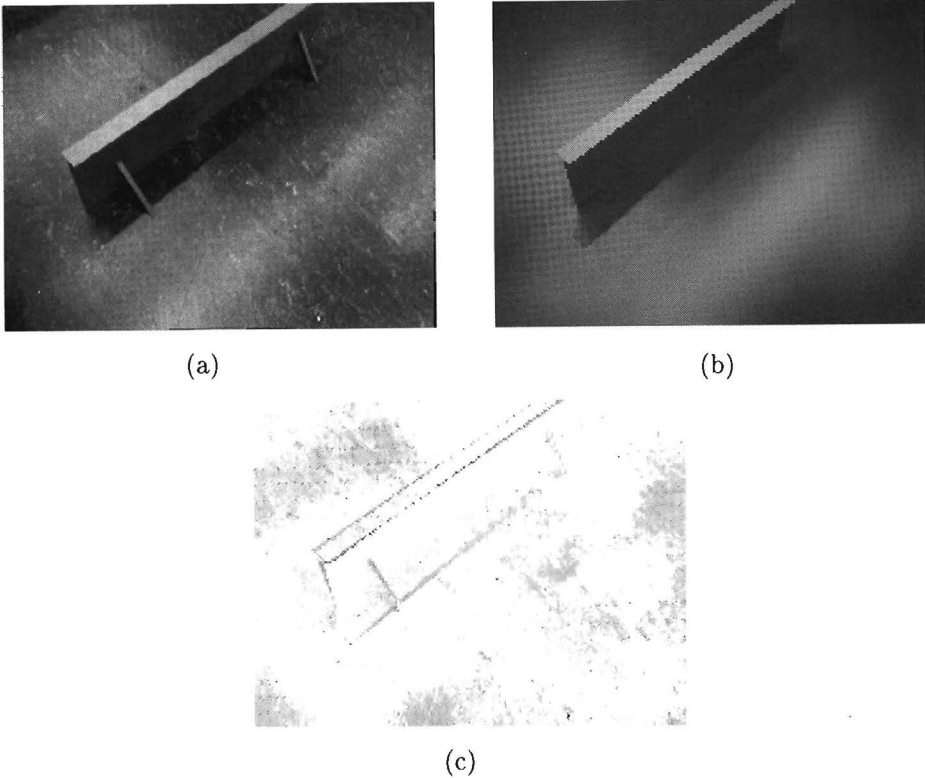
Figure 3.3: An example of the parameter estimation: (a) acquired camera image of the scene, (b) synthetic image computed using the estimated values, (c) difference between images (a) and (b). White colour in the difference image represents no difference in intensities between the images (a) and (b).

Chapter 4

Verification of scene descriptions

Verification of the 3-D scene description can be regarded as a decision problem, where a decision has to be made about the truth of some given 3-D scene description. The decision is to be based on the available camera images, and on data computed by the imaging model. What we seek is a *decision procedure*, or a *decision rule*, a prescription for the selection of a decision from the set of possible decisions based on the available data. Because of an influence of noise, and the stochastic nature of the imaging model, the acquired images as well as the output of the imaging model are considered to be random (realizations of random variables). As a consequence, no test can by itself provide an evidence of truth or falsehood of the hypothesis, and there always remains a possibility of making an incorrect decision. Our aim then might be to look for a decision procedure which, in a long run experience, will not be too often wrong, without hoping to know whether each particular hypothesis is true or false. The aim is to find and use the optimal decision procedure with respect to some statistical optimality criterion. Sometimes the aim is to find an optimal decision procedure having some given structure, which is easy to compute, or to realize in hardware. This topic is studied by *statistical decision theory* [14].

In this chapter two formulations of the decision problem are discussed, Neyman-Pearson and Bayesian. For the verification of scene descriptions we have adapted the decision procedure which has an attractive physical implementation, a two-dimensional (2-D) North filter followed by a thresholding operation. Originally, North filter is used for the detection of a known continues signal in a one-dimensional gaussian process, and the adapted version can detect the known pattern in the 2-D gaussian signal. The problem of the selection of the decision

procedure is thus reduced into a problem of the choice of the filter coefficients, and the choice of a threshold level.

Related to this chapter is Appendix B, where the theory of Karhunen-Loeve expansion is given, and Appendix C, where the derivation of filter coefficients for the test with a composite alternative is given.

4.1   Elements of a decision problem

Parameter, parameter space, events.   Suppose that the unknown reality about which it must be decided can be represented by a number of variables, a vector, which is traditionally called *a parameter*. N.B., an analogy between the parameter estimation and decision theory can not be denied. The parameter value not known to us is a point in a *parameter space*. The way in which the parameter represents the reality might be complicated (e.g. in the case of a description of a scene), however this does not alter the concepts of the decision theory. Thereby, we may think of some facts as of points from the parameter space, or subsets of this space.

Formally, let w be a parameter which belongs to a *parameter space* Ω. Any subset A of the parameter space Ω, belonging to an appropriate set of subsets of Ω, is called an *event*. A set of disjoint events constitutes a set of hypotheses of a decision problem. A hypothesis comprising a single element of the parameter space Ω is called a *simple* hypothesis, while the hypothesis comprising a proper subset of the parameter space is called a *composite* hypothesis.

Sample, Sample space.   Before choosing a decision about the truth of the events, we have the opportunity to observe an outcome (a sample) of a random variable, or a random vector X, taking a value from a *sample space* $\mathcal{X}$. Random variable X is related to the parameter w to the effect that a probability distribution of X is given for each $w \in \Omega$. In case of a continuous X, a family of density functions $\{f_w(x), w \in \Omega\}$ specifies the relation. The density function $f_w(x)$ is also called a *likelihood*.

Decision space, decision procedure.   A *decision space* $\mathcal{D}$ is the set of all possible decisions $d \in \mathcal{D}$. In the most simple case, set $\mathcal{D}$ contains only two elements, e.g. $\mathcal{D} = \{d_1, d_2\}$, where the interpretation of the decisions might be $d_1 =$'accept the hypothesis H_1' and $d_2 =$'accept the hypothesis H_2'.

A *decision procedure*, or a *decision rule* δ, is a mapping $\delta : \ \mathcal{X} \mapsto \mathcal{D}$ from the sample space to the decision space, that assigns to each possible outcome $x \in \mathcal{X}$

the decision $\delta(x) \in \mathcal{D}$. For a finite decision set $\mathcal{D} = \{d_1, d_2, \ldots d_n\}$, the decision procedure is a partitioning of the sample space $\mathcal{X}$ into subsets $\mathcal{X}_i, i = 1, \ldots n$, each corresponding to a particular decision $d_i \in \mathcal{D}$, so that $x \in X_i$ implies $\delta(x) = d_i$.

4.1.1 Neyman–Pearson problem formulation

Neyman and Pearson [60, 61] studied the problem of deciding between two hypotheses, a hypothesis H_0, usually called a *null hypothesis*, and an *alternative hypothesis* H_1. The decision space of the problem consists of two decisions $\mathcal{D} = \{d_0, d_1\}$ with interpretations $d_0 = $ 'accept the null hypothesis, reject the alternative', and $d_1 = $ 'reject the null hypothesis, accept the alternative'. The decision rule is given by partitioning of the sample space $\mathcal{X}$ into two disjoint subsets $(\mathcal{X}_0, \mathcal{X}_1)$, where $\mathcal{X}_1 = \mathcal{X} \setminus \mathcal{X}_0 = \mathcal{X}_0^c$ ($\mathcal{X}_0^c$ denotes the complement of $\mathcal{X}_0$). The subset $\mathcal{X}_1$ is sometimes called a *critical set*, since for $x \in \mathcal{X}_1$ the null hypothesis is rejected.

Consider first the test of the simple hypothesis $H_0 = \{w_0\}$ against the simple alternative $H_1 = \{w_1\}$. Making a decision d_1, when H_0 is actually true is called a *type I error*, and the probability of this event is denoted by α. Making a decision d_0, when H_1 is actually true is called a *type II error*, and the probability of this event is denoted by β. Obviously

$$\alpha_\delta = \int_{\mathcal{X}_1} f_{w_0}(x)\, dx, \qquad \beta_\delta = \int_{\mathcal{X}_0} f_{w_1}(x)\, dx, \qquad (4.1)$$

where $f_{w_0}(x)$ and $f_{w_1}(x)$ are probability densities of X under hypotheses H_0 and H_1 respectively. The subscript δ indicate that α and β are characteristics of the decision rule δ. The problem is to find $\mathcal{X}_0$ and $\mathcal{X}_1$ such that α and β are as small as possible. However, in general, making α smaller will increase β, and the other way around. Neyman and Pearson proposed to confine the attention to a class of *size-α* decision rules, with the type I error probability less then or equal to α, and find among them the one with the smallest type II error probability. Formally, the decision rule δ^* is optimal in Neyman-Pearson sense, or it is the *most powerful* size-α decision rule, if it satisfies

$$\beta_{\delta^*} = \inf \beta_\delta \qquad (4.2)$$

subject to

$$\alpha_{\delta^*} \leq \alpha.$$

The way of searching for the most powerful decision rule is given by the following theorem:

Lemma 1 (Neyman-Pearson [61]) *Let $f_{w_0}(x)$ and $f_{w_1}(x)$ be the densities of the random variable X under the hypotheses $H_0 : w = w_0$ and $H_1 : w = w_1$ respectively. Let $(\mathcal{X}_0, \mathcal{X}_1)$ be the partition of the sample space $\mathcal{X}$, specifying a decision rule δ of size-α. If there exists a $\lambda \in \mathcal{R}$ such that*

$$\frac{f_{w_1}(x)}{f_{w_0}(x)} \leq \lambda \qquad \text{for all } x \in \mathcal{X}_0,$$

and

$$\frac{f_{w_1}(x)}{f_{w_0}(x)} \geq \lambda \qquad \text{for all } x \in \mathcal{X}_1,$$

then the decision rule δ is the most powerful decision rule of size-α in Neyman-Pearson sense.

The problem gets more complicated when either H_1, or both H_0 and H_1 are composite hypotheses, and the parameter w belongs to an infinite subset under hypothesis H_1 or H_0. Consider for instance a test of the simple hypothesis $H_0 : w = w_0$ against the *one-sided* composite alternative $H_1 : w > w_0$. Instead of a single type II error probability β, the decision procedure is characterized by the so called *power function* of the test, $\beta_\delta : (w_0; \infty) \mapsto [0; 1]$, giving a type II error probability for different values of parameter w. The power function will be defined in the sequel.

The Neyman-Pearson point of view for the most general case of the composite hypothesis $H_0 : w \in A_0$, and the composite alternative $H_1 : w \in A_1$, is summarized in the following definitions:

Definition 1 (size-α decision procedure) *The decision procedure δ, given by the partition of the sample space $(\mathcal{X}_0, \mathcal{X}_1)$, is a size-$\alpha$ decision procedure if*

$$\sup_{w' \in A_0} P(x \in \mathcal{X}_1 \mid w = w') = \alpha$$

Definition 2 (power function) *The power function P_δ of the decision procedure δ is the function $P_\delta : A_0 \cup A_1 \mapsto [0; 1]$ given by*

$$P_\delta(w') = P(x \in \mathcal{X}_1 \mid w = w')$$

Definition 3 *Let δ and δ' are two size-α decision procedures. The decision procedure δ is said to be better than the decision procedure δ', if $P_\delta(w) \leq P'_\delta(w)$ for all $w \in A_1$, and $P_\delta(w) < P'_\delta(w)$ for at least one $w \in A_1$. The decision procedure δ' is said to be not better than the decision procedure δ, if $P_\delta(w) \leq P'_\delta(w)$ for all $w \in A_1$.*

Definition 4 (uniformly most powerful decision procedure) *The decision procedure δ is said to be the* uniformly most powerful decision procedure, *if each other size-α decision procedure is not better than δ in the sense of definition 3.*

Sometimes the uniformly most powerful decision procedure might not exist. Then one should use at least an *admissible decision procedure* defined as follows:

Definition 5 (admissible decision procedure) *The decision procedure δ is called an admissible procedure if there does not exist any better procedure in the sense of definition 3.*

Some additional ordering principles might be used to choose a decision procedure from the admissible decision procedures. However, this lies outside of the scope of the thesis.

4.1.2 Bayesian decision procedure

From the *Bayesian* point of view, the unknown parameter w can be regarded as an outcome of the random variable W with some known *a priori* probability distribution. Even in cases when the parameter is not the result of a chance mechanism, it might be considered to be obtained so, since it is unknown to us. For the sake of simplicity, we identify here the parameter value w with the hypothesis itself. This implies no loss of generality, since the appropriate transformation of the random variable W can always be done. We will also confine the discussion to problems where both the parameter space Ω and the decision space D are finite, with k ($k \geq 2$) and m ($m \geq 2$) elements respectively[1]. Hence, $\Omega = ega = ega = ega = ega = \{\omega_1, \ldots, \omega_k\}$ and $D = \{d_1, \ldots, d_m\}$.

Once the outcome of X, which is related to the actual value w, has been observed, the probability distribution of W is updated. According to the Bayes rule

$$\pi(w) = \frac{f(x|w)\pi_0(w)}{\sum\limits_{w_i \in \Omega} f(x|w_i)\pi_0(w_i)} \qquad \text{for all } w \in \Omega, \qquad (4.3)$$

where $\pi_0(w)$ is the a priori probability of an event w, and $\pi(w)$ is the updated *a posteriori* probability of this event.

The updated distribution is the basis for the decision making.

---

[1] A special case when $k = m$, and the decision d_i can be interpreted as 'accept hypothesis $w = w_i$, is known as *classification*. Each hypothesis is then called a *class*.

Suppose that the consequences of decision making are expressed by a *loss function* $L : \Omega \times D \mapsto R$, with the value $L(w, d)$ being a loss incurred by making a decision d when event w occurred. Alternatively a utility function $U : \Omega \times D \mapsto R$, with the values $U(w, d)$ being utilities (rewards), might be used. The choice of a loss (utility) function is often subjective, and is evident only in some situations, like in the case of monetary losses or rewards. From now on we will assume that the loss function is given.

For the parameter with a probability distribution given by a mass function $\pi : \Omega \mapsto R^+$, the *risk* R of making a decision d is defined as an expected loss incurred by making this decision

$$R(\pi, d) = \sum_{w_i \in \Omega} L(w_i, d) \pi(w_i). \qquad (4.4)$$

The aim is to find a *decision procedure* $\delta : \mathcal{X} \mapsto D$, which specifies for each value $x \in \mathcal{X}$ a decision $\delta(x) \in D$.

The Bayesian decision procedure $\delta^* : \mathcal{X} \mapsto \mathcal{D}$ is a procedure which selects the decision d^*, that minimizes the expected loss (the risk), hence

$$R(\pi, d^*) = \inf_{d \in D} R(\pi, d). \qquad (4.5)$$

The decision d^* is called the *Bayesian decision* and the risk $R(\pi, d^*)$ is called the *Bayesian risk*. There might be more than one Bayesian decision (and therefore also more Bayesian decision procedures), if they all attain the infimum in (4.5).

In the case of finite spaces Ω and D, it is straightforward to find the Bayesian decision d^* by exhaustive evaluation of risks for all decisions, or by the technique given in [14], page 132. The main concern remains to compute the probability distribution π.

4.1.3   Discussion of the problem formulation

First it will be shown, that both the Neyman-Pearson and the Bayesian formulations may lead to the same decision procedures. Suppose that under the hypothesis H_0 the probability density of the sample X is $f_0(x)$, and under the alternative H_1 the density is $f_1(x)$. The decision space is $\mathcal{D} = \{d_0, d_1\}$ with $d_0 =$ 'H_0 is true' and $d_1 =$ 'H_1 is true'.

From Lemma 1 it follows that the most powerful decision procedure in Neyman-Pearson sense has the form

$$\delta(x) = \begin{cases} d_1 & \text{if } \frac{f_1(x)}{f_0(x)} > T, \\ d_0 & \text{otherwise.} \end{cases} \qquad (4.6)$$

Either decision d_0, or d_1 can be made in the case $\frac{f_1(x)}{f_0(x)} = T$, without affecting the performance. To obtain a size-α decision procedure, the threshold T must be adjusted so that the type I error probability computed by (4.1) is α.

To find a Bayesian decision procedure, assume that the costs of incorrect decisions are $L(H_0, d_1) = l_0$, $L(H_1, d_0) = l_1$, and there are no costs associated with the correct decisions. The risk incurred by making decisions d_0 and d_1 is

$$R(\pi, d_0) = l_1 \pi(H_1), \qquad R(\pi, d_1) = l_0 \pi(H_0), \qquad (4.7)$$

where from (4.3) we have

$$\pi(H_i) = \frac{f_i(x)\pi_0(H_i)}{f_0(x)\pi_0(H_0) + f_1(x)\pi_0(H_1)}, \qquad i = 0, 1, \qquad (4.8)$$

By comparing $R(\pi, d_0)$ and $R(\pi, d_1)$, the Bayesian decision procedure is

$$\delta(x) = \begin{cases} d_1 & \text{if } \frac{f_1(x)}{f_0(x)} > \frac{l_0 \pi_0(H_0)}{l_1 \pi_0(H_1)}, \\ d_0 & \text{otherwise.} \end{cases} \qquad (4.9)$$

Obviously, it is not difficult to find an α, l_0 and l_1, such that the most powerful decision procedure and the Bayes decision procedure are identical.

The main difference between the Bayesian and Neyman-Pearson approach is that with the Bayesian approach the parameter is treated as a random variable and its a priori probability distribution is applied, while with the Neyman-Pearson approach the parameter is considered to have some fixed unknown value. Notice, that the Bayesian decision procedure (4.3) depends on the a priori probabilities $\pi_0(\cdot)$. It is generally accepted, that in well described, or well identified situations, the Bayesian approach leads to better overall performance due to the fact, that it uses in a well-defined way more knowledge about the parameter. On the other hand, the Neyman-Pearson formulation leads to simpler solutions, when in the ultimate case only one threshold value has to be determined to achieve the required performance, without the necessity to possess a priori probabilities, costs, etc.

It may be concluded, that the choice of the performance criterion (α-size for Neyman-Pearson formulation, cost function for Bayes procedures), and also the proper knowledge of the underlying statistical quantities (likelihood functions, a priori probabilities) have a major impact on the decision procedure performance. This is not so for the choice of the problem formulation itself.

It can also be seen, that the likelihood ratio $\frac{f_1(x)}{f_0(x)}$ plays an important role in this decision problem. The logarithm of the likelihood ratio (called the loglikelihood ratio), or another monotonic function of the likelihood ratio can be used by the decision procedure instead of the likelihood ratio, and only the threshold value has to be adjusted appropriately.

4.2   The decision procedure for scene verification

The **parameter space** of the scene description verification problem can be identified with the set of all possible scene descriptions. Let $\mathcal{G}$ be the set of all possible scene descriptions, and let $G_0 \in \mathcal{G}$ be the candidate scene description, which has to be verified based on acquired camera images using an imaging model. The verification of a scene description might be seen as the statistical test of the hypotheses $H_0 : \{G_0\}$, $G_0 \in \mathcal{G}$ against the alternative $H_1 : \mathcal{G} \setminus \{G_0\}$.

The **decision space** of the verification problem consists of two elements, $\mathcal{D} = \{d_0, d_1\}$, with the decisions $d_0 =$'accept the candidate scene description', and $d_1 =$'reject the candidate scene description'.

The **sample space** of the scene description verification can be identified with the set of possible images. Consider a digital image with PELs placed in a rectangular grid. The set of indices of all PELs on the image plane is $\mathcal{I} = \{(i,j);\, i = 1,\ldots,n_{\mathrm{col}};\, j = 1,\ldots,n_{\mathrm{row}}\}$, with n_{col} denoting the number of columns, and n_{row} denoting the number of rows of PELs in the image. Let $\mathcal{Q}$ be the set of signal values of each PEL. For the grey scale images with the resolution k-bits per PEL, set $\mathcal{Q}$ is the subset of the set of natural numbers, $\mathcal{Q} = \{0, 1, \ldots, 2^k - 1\}$, and the values of the PELs are treated as image plane irradiances. Although it seems to be more natural to treat images as two-dimensional arrays (matrices), for the sake of notational convenience we will represent the images by vectors. Thereby, the sample vector x is a column vector of the size $n_{\mathrm{col}} n_{\mathrm{row}}$, with its components being the image irradiance values of PELs ordered in some specific way, e.g. by image rows. The sample space of the scene description verification problem is the set $\mathcal{X} = \mathcal{Q}^{n_{\mathrm{col}} n_{\mathrm{row}}}$.

To be able to select a decision from the set $\mathcal{D}$, we must be in a possession of a family of likelihood functions $\{f_G(x), G \in \mathcal{G}\}$. Likelihood function $f_G(x)$ for a given scene description G can be obtained with the use of the imaging model. Use of the imaging model is a computationally intensive process, and apparently it is not feasible to apply this model for all $G \in \mathcal{G}$. Instead, the hypothesis and the alternative must be formulated directly on the set of all possible images, the sample space $\mathcal{X}$, using only a likelihood function for the candidate scene description $f_{G_0}(x)$. In this way the imaging model has to be addressed only once. Such formulation is only an approximation of an actual problem, however it is feasible to obtain the solution.

Another observation is that even for a modest image size the dimension of the sample vector is rather high, as it grows with a product of the vertical and horizontal image dimensions. The decision procedure is a partitioning of the sample space into two partitions, corresponding to the two decisions d_0 and

d_1. The high dimension of the sample space necessitates in a search for a decision procedure which is efficient in computations, or which can be realized in hardware.

In the subsequent sections, a decision procedure which is both mathematically rigid and convenient for the implementation, is proposed. It is a discrete two-dimensional extension of the *North filter*, which is used for detection of a known signal.

4.2.1 The North filter

A well studied problem of hypothesis testing and detection theory, which occurs e.g. in all radar systems and many other signal processing systems, is the detection of a known, finite energy continuous signal corrupted with additive Gaussian noise. The two hypotheses concerning the received signal $v(t)$ are:

$$
\begin{aligned}
H_0 : & \quad v(t) = n(t) & \text{(noise only),} \\
H_1 : & \quad v(t) = s(t) + n(t) & \text{(a known signal plus noise).}
\end{aligned}
\tag{4.10}
$$

where $n(t)$ is a zero-mean Gaussian noise, and $s(t)$ is a signal that would be received in an ideal, noiseless case. It is assumed, that the autocorrelation function of the noise $n(t)$ is known. The solution of this problem, a most powerful decision procedure in Neyman-Pearson sense, has a feasible physical realization displayed in figure 4.1. It consists of a *North filter*, also called a
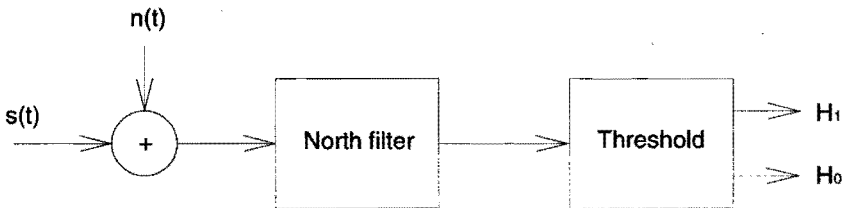
Figure 4.1: Detection of a known signal in a Gaussian noise.

matched filter, followed by a threshold. The North filter is a linear filter with the transfer function $S^*(f)/N(f)$, where $S^*(f)$ is the complex conjugate of the Fourier transform of the signal $s(t)$ to be detected, and $N(f)$ is the power spectral density[2] of the noise $n(t)$. The derivation of the filter and the proof of its optimality can be found e.g. in [5].

An analog problem in image processing is the detection of the known pattern in the zero-mean Gaussian noise on the two-dimensional image plane. The two

---

[2]Power spectral density is the Fourier transform of the autocorrelation function.

hypotheses concerning this problem are

$$
\begin{aligned}
H_0 : \quad & Z = N & \text{(noise only)} \\
H_1 : \quad & Z = p + N & \text{(a known pattern plus noise)}
\end{aligned}
\tag{4.11}
$$

where Z is the random vector representing the acquired image, p is a constant vector specifying the given pattern, and N is a Gaussian random vector with zero mean and a given covariance. The vectors Z, p and N are column vectors of the size $n_{col}n_{row}$. For this problem, a decision procedure depicted in figure 4.2 can be applied: the image is filtered by a finite impulse response (FIR) filter
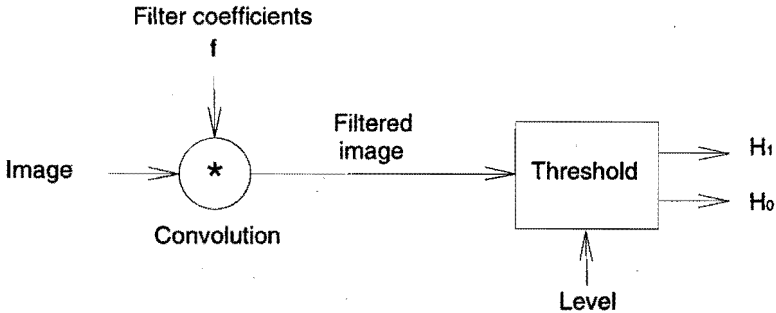
Figure 4.2: Detection of a known pattern in the noise on the image plane.

with coefficients f, and the result is thresholded. When the threshold level is exceeded, H_0 is rejected and H_1 is accepted, otherwise H_0 is accepted and H_1 is rejected. The questions of the choice of the filter coefficients f, and of the optimality of such a statistical test are pursued in the next section.

Suppose a hypothetic scene description G_0 (the null hypothesis H_0) leads to the distribution of the image data $X = X_0 \sim \mathcal{N}(x_0, Q)$, where x_0 and Q are the mean and the covariance matrix of a Gaussian distribution. The decision procedure of the figure 4.2 can be applied for the verification against the simple alternative $H_1 : X = X_1 \sim \mathcal{N}(x_1, Q)$, by using the substitution $Z = X - x_0$ and $p = x_1 - x_0$. Pattern p to be detected is the difference between the means x_0 and x_1.

The decision procedure can be applied for the verification against a composite alternative. The issue of the filter selection and the properties of the decision procedure in such case are pursued in section 4.2.3.

4.2.2 Optimality of the decision procedure

When developing decision procedures on a sample space $\mathcal{X}$ of a high dimension, it is useful to investigate the expansions of a random vector X in terms of a set

of orthonormal vectors. Hence, for the m-dimensional[3] random vector X we may write

$$X = \sum_{k=1}^{m} V_k \psi_k \qquad (4.12)$$

where V_k, $k = 1, \ldots, m$, are scalar coefficients obtained by

$$V_k = \psi_k' X, \qquad (4.13)$$

and ψ_k, $k = 1, \ldots, m$, are column vectors forming an orthonormal base, so that

$$\psi_k' \psi_l = \begin{cases} 1 & \text{for } k = l, \\ 0 & \text{for } k \neq l. \end{cases} \qquad (4.14)$$

In the above expressions primes denote the transpose. Coefficients of the expansion V_k, $k = 1, \ldots, m$ are random variables. We seek an orthonormal base ψ_k, $k = 1, \ldots, m$, that yields un-correlated expansion coefficients:

$$E[V_k V_l] = \begin{cases} \lambda_k & \text{for } k = l, \\ 0 & \text{for } k \neq l. \end{cases} \qquad (4.15)$$

Furthermore, if X has a jointly Gaussian distribution, the coefficients V_k are independent Gaussian random variables. Consequently, it is easier to find appropriate decision procedures based on V_k, $k = 1, \ldots, m$, than the procedure based on $x \in \mathcal{X}$.

The expansion having the required properties is known as the Karhunen-Loeve expansion. Its theory is summarized in Appendix B. The orthogonal basis of the Karhunen-Loeve expansion is formed by the eigenvectors of the covariance matrix of X, and the associated eigenvalues are the correlation coefficients λ_k in (4.15).

Using the properties of the Karhunen-Loeve expansion, we can prove the following theorem concerning the optimality of a decision procedure:

Theorem 1 *Consider the decision procedure for the detection of a known pattern p corrupted by an additive zero-mean Gaussian vector with covariance matrix Q, proposed in figure 4.2. If the filter coefficients f solve the equation*

$$p = Qf,$$

then the decision procedure is the most powerful decision procedure in Neyman-Pearson sense.

---

[3]In the case of a decision procedure for the detection of a known pattern, $m = n_{col} n_{row}$.

We can prove the theorem following the lines of reasoning of the proof for the one-dimensional continues-time case [5, pg. 123].

Proof: An observed image x is the outcome of the random vector N under the hypothesis H_0, and the outcome of the random vector $p + N$ under the hypothesis H_1, where $N \sim \mathcal{N}(0, Q)$. To find a likelihood ratio, the observed image can be expanded into a Karhunen-Loeve expansion. The orthonormal base vectors of the Karhunen-Loeve expansion are the eigenvectors of the matrix Q, and the coefficients of the expansion under both hypotheses H_0 and H_1 are independent Gaussian random variables (see Appendix B). Let ψ_k, $k = 1, \ldots, m$ be the eigenvectors of matrix Q, and let λ_k be the corresponding eigenvalues. Furthermore, let the expansion coefficients of the observed image x be

$$u_k = \psi'_k x \qquad\qquad k = 1, \ldots, m,$$

and the expansion coefficients of the pattern p be

$$p_k = \psi'_k p \qquad\qquad k = 1, \ldots, m.$$

From the theory of the Karhunen-Loeve expansion, of the coefficients u_k are the realizations of random variables U_k with the distribution $\mathcal{N}(0, \lambda_k)$ under H_0, and $\mathcal{N}(p_k, \lambda_k)$ under H_1. Hence, the logarithm of the likelihood ratio of the test is

$$\log \frac{f_1(x)}{f_0(x)} = \sum_{k=1}^{m} \left(-\frac{(u_k - p_k)^2}{2\lambda_k} + \frac{u_k^2}{2\lambda_k} \right)$$

$$= \sum_{k=1}^{m} \frac{u_k p_k}{\lambda_k} - \frac{1}{2} \sum_{k=1}^{m} \frac{p_k^2}{\lambda_k}$$

According to Neyman-Pearson lemma, the most powerful test has the form: Decide in favour of H_1 if

$$\sum_{k=1}^{m} \frac{u_k p_k}{\lambda_k} - \frac{1}{2} \sum_{k=1}^{m} \frac{p_k^2}{\lambda_k} \geq T,$$

and otherwise decide in favour of H_0. The second term on the left hand side is a fixed constant and can be included into the threshold. Hence the decision rule for the decision in favour of H_1 is

$$\sum_{k=1}^{m} u_k \frac{p_k}{\lambda_k} \geq T_2, \qquad\qquad (4.16)$$

where T_2 is the redefined threshold T,

$$T_2 = T + \frac{1}{2} \sum_{k=1}^{m} \frac{p_k^2}{\lambda_k}. \qquad\qquad (4.17)$$

Let

$$w_k = \frac{p_k}{\lambda_k} \qquad k = 1, \ldots, m.$$

From theorem 6 in appendix B it follows, that w_k are coefficients of the Karhunen-Loeve expansion of f, such that

$$p = Qf.$$

From theorem 3 in appendix B it follows that the entity compared with the threshold in inequality (4.16) is equal to the result of the image convolution with the filter pattern f,

$$\sum_{k=1}^{m} u_k w_k = f'x = (x * f)(0). \tag{4.18}$$

This concludes the proof. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ $\square$

We will evaluate the probabilities of the type I and type II errors of the proposed decision procedure, as functions of the threshold value. The response of the filter

$$f'X = \sum_{k=1}^{m} U_k \frac{p_k}{\lambda_k} \tag{4.19}$$

is a random variable with distribution $\mathcal{N}(0, b)$ under the hypothesis H_0 , and $\mathcal{N}(b, b)$ under the hypothesis H_1, where b is defined as

$$b = \sum_{k=1}^{m} \frac{p_k^2}{\lambda_k}. \tag{4.20}$$

Type I error probability, that is the probability that the filter response exceeds the threshold value given H_0 is true, or the *probability of false alarm* is

$$\alpha = P\left\{ \sum_{k=1}^{m} U_k \frac{p_k}{\lambda_k} \geq T_2 \,\middle|\, H_0 \right\} = 1 - \Phi\left(\frac{T_2}{\sqrt{b}}\right) \tag{4.21}$$

and the type II error probability, that is the probability that the filter response does not exceed the threshold value given H_1 is true, or the *probability of miss* is

$$\beta = P\left\{ \sum_{k=1}^{m} U_k \frac{p_k}{\lambda_k} < T_2 \,\middle|\, H_1 \right\} = \Phi\left(\frac{T_2 - b}{\sqrt{b}}\right). \tag{4.22}$$

In (4.21) and (4.22) Φ denotes the cumulative distribution function of $\mathcal{N}(0, 1)$. For the size-α test, the threshold must be set to the value

$$T_2 = \sqrt{b}\,\Phi^{-1}(1 - \alpha) \tag{4.23}$$

Substituting this threshold into (4.22), we obtain

$$\beta = \Phi\left(\Phi^{-1}(1-\alpha) - \sqrt{b}\right), \tag{4.24}$$

which is a monotonically decreasing function of b. As a consequence, the discriminatory power of the test procedure increases with the increasing value of b, which is actually the response of the North filter to a noise-free pattern p.

Let us re-derive the North filter from another point of view. Let $f = (f_1, \ldots, f_m)'$ be the vector of the filter coefficients. Under the null hypothesis $H_0 : X \sim \mathcal{N}(0, Q)$, the output of the filter of figure 4.2 is a random variable O with the distribution

$$H_0 : \quad O \sim \mathcal{N}(0, f'Qf). \tag{4.25}$$

When a random vector $X \sim \mathcal{N}(p, Q)$ (the hypothesis H_1) is applied to the input of the filter, the distribution of the output is

$$H_1 : \quad O \sim \mathcal{N}(f'p, f'Qf). \tag{4.26}$$

To achieve a good discrimination against the null hypothesis, the ratio of the mean and the standard deviation $f'p/\sqrt{f'Qf}$ (or its square, as it follows below) should be maximized. The filter coefficients are the solution of the maximization problem

$$f = \operatorname*{argmax}_{f \in \mathcal{R}^m} \left\{ \frac{f'pp'f}{f'Qf} \right\}. \tag{4.27}$$

In the domain of Karhunen-Loeve coefficients an equivalent problem can be formulated. Let $\Psi = (\psi_1, \ldots, \psi_m)$ be the $m \times m$ matrix with columns that are the normalized eigenvectors of Q, and let Λ be an m by m diagonal matrix with the respective eigenvectors λ_k, $k = 1, \ldots, m$ on the diagonal. Let v, w and V be the expansion coefficients of p, f and X, respectively. We may write that $v = \Psi'p$, $w = \Psi'f$ and $V = \Psi'X$. The random vector V has the distribution

$$V \sim \mathcal{N}(\Psi'p, \Psi'Q\Psi) = \mathcal{N}(v, \Lambda),$$

and consequently, the distribution of the filter output O is

$$O \sim \mathcal{N}(w'v, w'\Lambda w).$$

Similarly to (4.27), the goal is to find filter expansion coefficients

$$w = \operatorname*{argmax}_{w \in \mathcal{R}^m} \left\{ \frac{w'v\,v'w}{w'\Lambda w} \right\}. \tag{4.28}$$

The problem (4.28) is easier to solve than (4.27) due to a fact that Λ is a diagonal matrix. After making substitutions

$$s = (\sqrt{\lambda_1}\, w_1, \ldots, \sqrt{\lambda_m}\, w_m) \tag{4.29}$$

$$t = \left(\frac{v_1}{\sqrt{\lambda_1}}, \ldots, \frac{v_k}{\sqrt{\lambda_m}} \right) \tag{4.30}$$

we arrive to a problem

$$s = \underset{s \in \mathcal{R}^m}{\arg\max} \left\{ \frac{s't\, t's}{s's} \right\}, \tag{4.31}$$

which has the solution $s = t/|t|$. This maximizes the dot product $t's$ in the numerator, under the constraint $s's = 1$. After back substitution, this solution yields the North filter.

4.2.3 Choice of the filter for a composite alternative

As opposed to the assumptions made in the previous section, the scene verification is a problem with a composite alternative. In this section we propose the application of the decision procedure of figure 4.2 for the decision problem with a composite alternative:

$$\begin{aligned} H_0 : & \quad X \sim \mathcal{N}(0, Q) \\ H_1 : & \quad X \sim \mathcal{N}(p, Q); \quad p \in \mathcal{P}. \end{aligned} \tag{4.32}$$

where $\mathcal{P}$ is a set of vectors representing possible difference patterns resulting from hypothetic and actual scene descriptions. We propose a method for the selection of filter coefficients for the case of a countable set $\mathcal{P}$.

The performance of the decision procedure of figure 4.2 can be characterized by the probability of the type I error α (the size of the decision procedure), and by the power function $\beta(p)$. We search for a filter f, that yields favourable detection properties, i.e. a small $\beta(p)$, for all elements from the set $\mathcal{P}$. It is not possible anymore to find a Neyman-Pearson decision procedure with the structure of figure 4.2, since for each different pattern a different optimal filter is required. Additional selection criteria must be used. We may seek a filter f, which under the fixed value α minimizes the probability β for the worst case from the set $\mathcal{P}$, i.e. minimizes the supremum of the power function $\beta(p)$. The size of the decision procedure α depends only on the hypothesis H_0, and can be adjusted by the proper setting of the threshold level. The required filter does not depend on the threshold level, therefore the decision procedure having the required properties can be obtained by first the filter selection, and then the threshold adjustment. This is worked out in the sequel.

The above stated criterion is formally expressed by:

$$f = \underset{f \in \mathcal{R}^m}{\arg\min} \left\{ \max_{p \in \mathcal{P}} \beta(p) \right\}. \tag{4.33}$$

Following the same line of reasoning as in the case with a simple alternative explained in the last part of the previous section, the required filter can be

obtained as the solution of the problem

$$f = \underset{f \in \mathcal{R}^m}{\arg\max} \left\{ \min_{p \in \mathcal{P}} \left\{ \frac{f'pp'f}{f'Qf} \right\} \right\}. \tag{4.34}$$

In the domain of the Karhunen-Loeve expansion coefficients the equivalent problem is

$$s = \underset{s \in \mathcal{R}^m}{\arg\max} \left\{ \min_{t \in \mathcal{T}} \left\{ \frac{s'tt's}{s's} \right\} \right\}, \tag{4.35}$$

where set $\mathcal{T}$ is defined as

$$\mathcal{T} = \left\{ t \in \mathcal{R}^m \,\middle|\, \sum_{k=1}^{m} \psi_k \sqrt{\lambda_k}\, t_k \in \mathcal{P} \right\}. \tag{4.36}$$

Loosely speaking, the set $\mathcal{T}$ is obtained by the Karhunen-Loeve transformation of the elements of the set $\mathcal{P}$ and their substitution according to (4.30). Under the constraint $s's = 1$, problem (4.35) is equivalent to the problem

$$s = \underset{s \in \mathcal{R}^m}{\arg\max} \left\{ \min_{t \in \mathcal{T}} \{t's\} \right\}. \tag{4.37}$$

The geometrical interpretation of this problem and its solution are given in Appendix C. Once this problem is solved, expansion coefficients $w = (w_1, \ldots, w_m)'$ of the required filter are obtained from the found vector s by a back substitution via the inverse of (4.29). Then the required filter coefficients f satisfying the criterion (4.33) are obtained by the inverse Karhunen-Loeve transformation of the vector w.

Notice, that in the case of a diagonal covariance matrix Q, the Karhunen-Loeve transform is an identity, and we obtain $\Lambda = Q$. In this case the filter design problem can be directly solved in the spatial domain.

We complete the analysis by computing the error probabilities for the threshold level T_2. Notice, that under the constraint $s's = 1$ also $w'\Lambda w = f'Qf = 1$, and the filter output O will always have a normal distribution with the standard deviation one. The mean value depends on p and is $f'p = w'v = s't$. The type I error probability, or the *probability of false alarm* is

$$\alpha = P\{O \geq T_2 | H_0\} = 1 - \Phi(T_2), \tag{4.38}$$

and the power function, i.e. the type II error probability, or the *probability of miss* as a function of p is

$$\beta(p) = P\{O < T_2 | H_1\} = \Phi(T_2 - f'p), \tag{4.39}$$

where Φ denotes the cumulative distribution function of $\mathcal{N}(0,1)$.

To obtain the size-α test, the threshold must be set to the value

$$T_2 = \Phi^{-1}(1 - \alpha).$$

(4.40)

The power function for this threshold is

$$\beta(p) = \Phi\left(\Phi^{-1}(1 - \alpha) - f'p\right).$$

(4.41)

4.2.4   Actual performance of the decision procedure

In this section we will revise the error probabilities α and β computed in sections 4.2.1 and 4.2.3, and the validity of theorem 1 for the decision procedure in figure 4.2.

In the development of the theory an implicit assumption has been made that the position of the pattern to be detected on the image plane is known. For instance, the likelihood ratio is computed in (4.18) as the convolution of the image and the filter pattern at the given location on the image plane. By application of the convolution filter and thresholding the resulting image, the decision process has actually been performed many times for each PEL. The thresholded image can be seen as the result of a grid of separate decision makers, each making a local decision about the presence of the pattern at the particular place. To each of these decision makers appart applies the theory and the error probabilities derived in preceding sections.

What one actually needs is to make a joint decision about the presence of the known pattern $p \in \mathcal{P}$ anywhere in the image. This can be done by a slightly modified decision procedure: the final decision will be in favour of hypothesis H_1 (the pattern is present), when at least one of the PELs decides in favour of H_1, otherwise the final decision will be in favour of H_0. In other words, if anywhere in the filtered image the intensity exceeds the threshold level, hypothesis H_1 is accepted. However, we possess no theorem concerning the (Neyman-Pearson) optimality of this modified decision procedure. Still, we may use the Norton filter, or the filter developed in the previous section, to obtain very satisfactory results.

It is not difficult to assess the error probabilities of the modified decision procedure for the case of a diagonal covariance matrix Q. In this case the decisions are independent under the hypothesis H_0, and since the type I error occurs if any PEL makes a type I error, the probability of the type I error of the modified procedure α_m is

$$\alpha_m = 1 - (1 - \alpha)^n,$$

(4.42)

where n is the total number of decision makers (PELs). The effect of the number of PELs on the distribution of the maximum of filter outputs over the entire
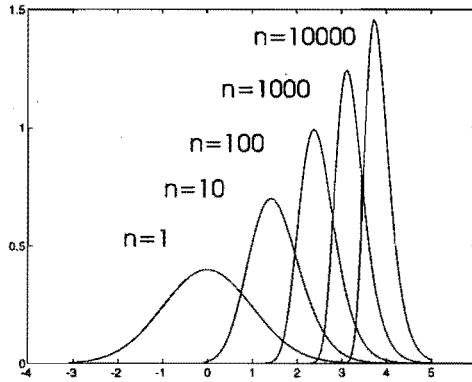
Figure 4.3: The probability density function of a random variable $\max\{X_i; i = 1, \ldots, n\}$, *where* $X_i \sim \mathcal{N}(0,1)$, *and* $E(X_i, X_j) = 0$ *for* $i \neq j$. *Plotted for* $n = 1, 10, 100, 1000$ *and* 10000 *(from left to right). For* $n = 1$ *the probability density function is the density function of normal distribution* $\mathcal{N}(0,1)$.

image is depicted in figure 4.3. Obviously, this leads to a larger type I error probability $\alpha_m > \alpha$, or one must set the threshold T_2 higher to attain the required test size. This deteriorates the type II error probability β_m.

It is still very likely, that the maximum of the filtered image under the hypothesis H_1 which is decisive for the final decision, will occur at a place where the pattern is localized. The total number of decision makers actually does not play a role and the type II error probability of the modified procedure β_m is approximately the same as the type II error probability of the single decision maker,

$$\beta_m \approx \beta. \tag{4.43}$$

We will omit the analysis of the case when Q is not diagonal.

4.2.5   The choice of a filter and a threshold

The design of the decision procedure for the scene description verification consists of two steps: the selection of a filter (the computation of the filter coefficients), and the choice of the threshold value.

The vector of coefficients of the North filter f is the solution of the equation

$$p = Qf, \tag{4.44}$$

where p is the pattern (the difference of the two images x_0 and x_1) to be detected, and Q is a covariance matrix of the images. The computation of filter coefficients is in general a time consuming process due to a large length of the vectors p and f, and a large size of the square matrix Q. The size is equal to the number of image PELs. For the composite alternative, the filter can be designed in the space of Karhunen-Loeve expansion coefficients of the difference patterns from the set $\mathcal{P}$.

A diagonal covariance matrix Q, that means independent components of the sample vector X, may be assumed. In this case the filter coefficients can be computed by the scalar divisions

$$f_i = \frac{p_i}{Q_{ii}}, \quad i = 1, \ldots, m, \tag{4.45}$$

and if furthermore all diagonal elements of the matrix Q are equal, the filter coefficients are proportional to the pattern to be detected. In the case of a composite alternative and a diagonal matrix Q, the filter design can be done directly in the spatial domain.

Another question to be answered is which difference patterns should be used to design the filter. The appropriate patterns depend on the domain of application and the likely mistakes of the computer vision system and from that resulting scene descriptions mistakes. The filter coefficients can be obtained successively from the data accumulated during the usage of the system. At the beginning an appropriate difference pattern able to detect most of the significant inconsistencies in the scene description must be applied. We have used for this purpose the filter with a circular shape (see figure 4.4) with a diameter adjusted to the size of objects in the scene. The (typical) difference images obtained from
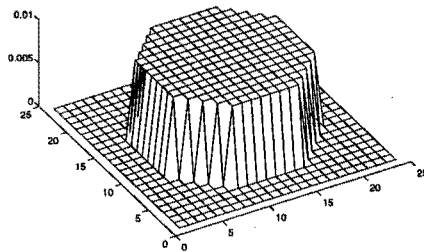

Figure 4.4: Filter shape for the verification procedure.

the experimental run can be used to design the filter by the technique for the composite alternative.

Once the filter has been designed, the threshold level must be set appropriately to achieve the required test parameters, namely the probability of the type I error α. The level can be determined based on the records of the maximum values of the filter output for a number of experiments performed with a correct scene description. This kind of learning is called a *supervised learning*. From the records, the parameters of the distribution are determined, and the threshold level is set so that the required α-size of the test is achieved.

Since the filter coefficients and the threshold level are decisive for the performance of the verification system, great care must be paid to their adjustment for the particular application.

4.3   Experimental results

The application of the scene description verification procedure is demonstrated in two examples.

Example 4.1

In the first example an artificial scene consisting of paper boxes placed on a table is verified. The correct description and two different incorrect descriptions of the scene have been processed. Intermediate and final results of the verification are shown in figures 4.5 and 4.6.

For each scene description, the surface reflectance parameters of the boxes and of the table have been estimated. Then the difference between the synthetic image obtained by stochastic ray tracing, and an original camera image has been computed. The difference image has been filtered by a convolution by a filter with the coefficients shown in the figure 4.4. The maximum of the filtered image has been determined, and when this maximum was beyond the threshold level, the discrepancy between the camera image and the scene description has been signalled. This is shown by a cross in the filtered image at the location where the maximum has been detected which exceeded the threshold.

The camera image of the scene is shown in figure 4.5a. In figures 4.5b, 4.5c and 4.5d the synthetic, difference and filtered images, respectively, are shown for the correct scene description.

In figures 4.6a, 4.6b and 4.6c the synthetic, difference and filtered images, respectively, are shown for the incorrect correct scene description. The small box present in the left hand side of the acquired image of the scene is not present

in the scene description. The maximum intensity of the filtered image is at the location of the inconsistency between the scene and the description.

In figures 4.6d, 4.6e and 4.6f the synthetic, difference and filtered images, respectively, are shown for the incorrect scene description. The small box placed on a larger box is missing in the scene description. The inconsistency has been detected at the location where the box cast a shadow.

Example 4.2

In this second example the scene containing an industrial object, a steel plate construction, is verified. The correct description, and one incorrect descriptions where a narrow top plate of the T-shaped profile is missed, have been processed. Intermediate and final results of the verification are shown in figures 4.7 and 4.8.

(a)   (b)

(c)

(d)

Figure 4.5: An example of the scene description verification in the case of a correct scene description. (a) The acquired camera image, (b) the synthetic image (c) the difference image, and (d) the filtered image.

(a)   (b)

(c)   (d)

(e)   (f)

Figure 4.6: Examples of the scene description verification in the case of incorrect scene descriptions. (a), (b) The synthetic images, (c), (d) the respective difference images, and (e), (f) the respective filtered images. The location of the detected inconsistency is marked with a cross.

(a)

(b)
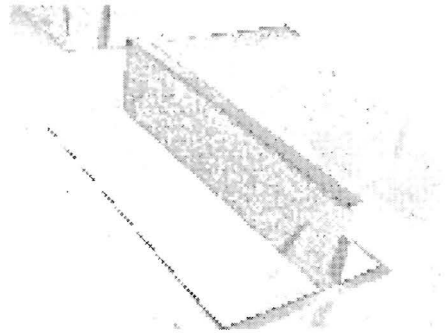
(c)
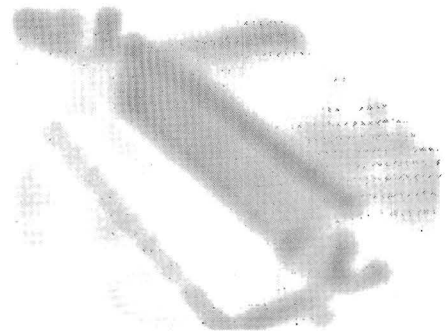
(d)

Figure 4.7: Example of the scene description verification in the case of a correct scene description. (a) The acquired camera image, (b) the synthetic image (c) the difference image, and (d) the filtered image.
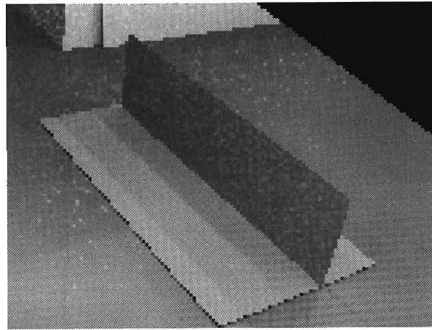
(a)
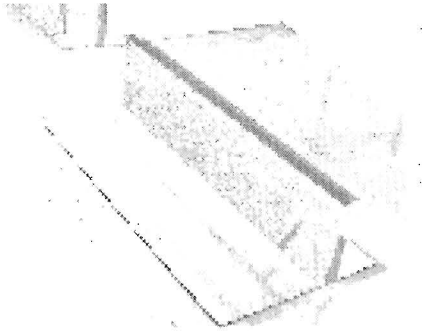
(b)
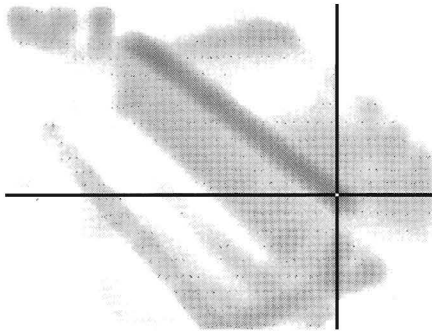
(c)
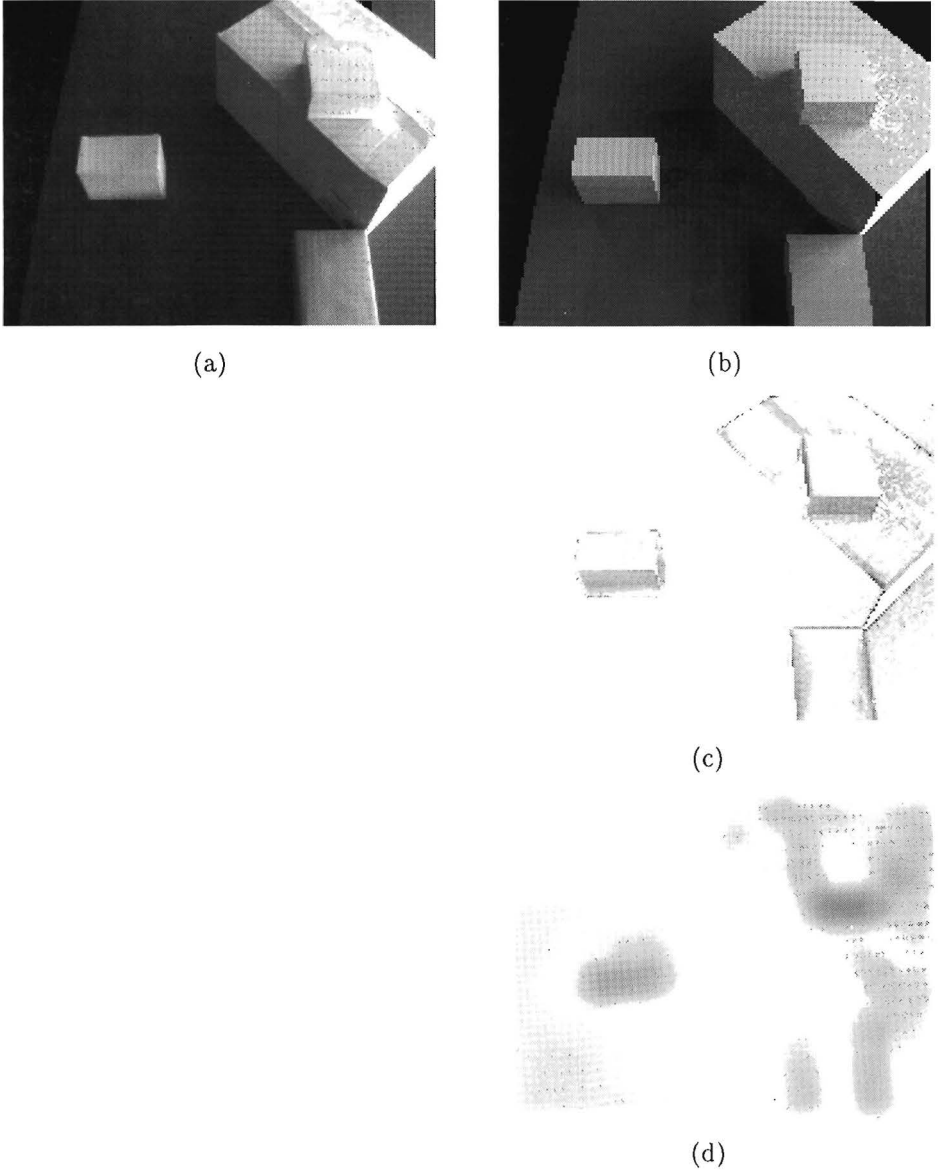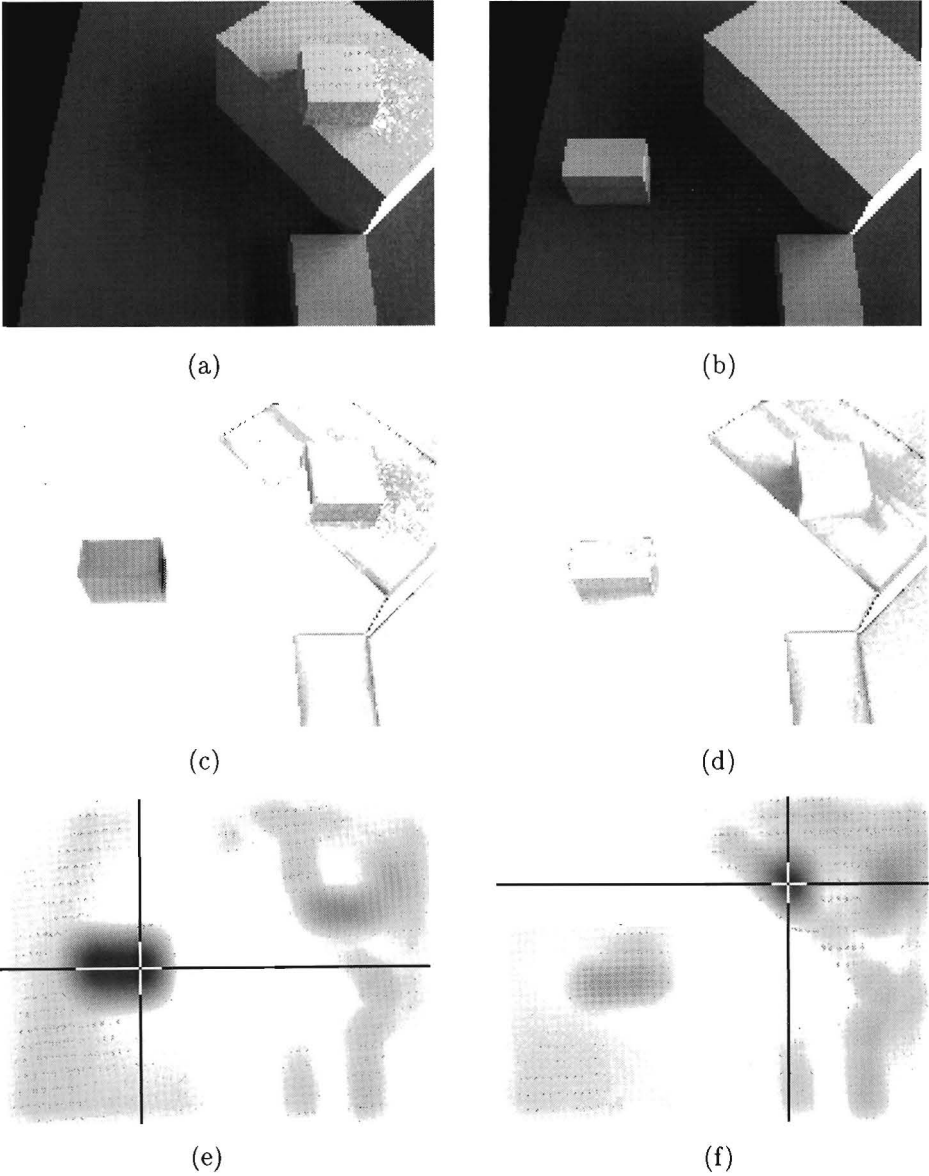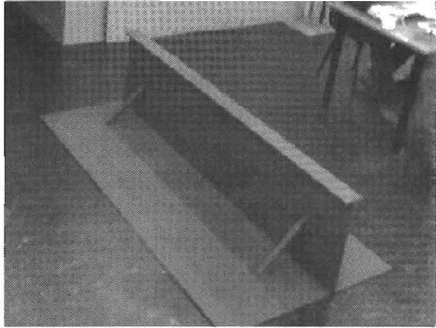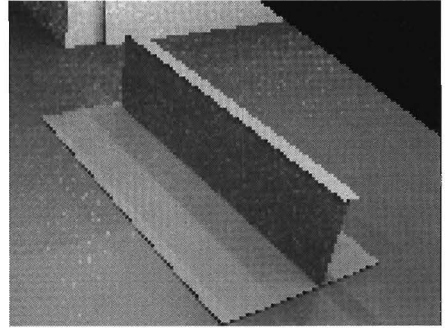
Figure 4.8: Example of the scene description verification in the case of an incorrect scene description. (a) The synthetic image, (b) the difference image, and (c) the filtered image. The location of the detected inconsistency is marked with a cross.

Chapter 5

Iterative improvement of the scene description

In the previous sections, the model of the imaging process has been described. Based on this model, techniques for the estimation of unknown parameters and for the scene verification have been proposed. To apply these techniques an initial (hypothetic) scene description has to be available. The obtained result is either the confirmation of the hypothetic scene description, or the detection of a discrepancy between the description and the acquired images. In the latter case, the discrepancy is localized on the image plane in the acquired image. In this chapter we will discuss ways of correction or improvement of the initial scene description, that remove or diminish the discrepancy. Repetition of the processes of scene verification and scene correction/improvement should after a number of iterations lead to an improved description or even to the correct scene description, provided that the actual scene satisfies requirements and assumptions made in the previous sections.

In section 5.1 the classification of discrepancies between the scene description and the acquired images is discussed. Depending on the kind of discrepancy, scene description corrections are proposed. Known computer vision techniques like photometric stereo and shape from shading can be utilized. The issues concerning the interconnection (interchange of information) between the applied technique and our specific setup are discussed in sections 5.4.2 and 5.4.1. The application of the techniques is demonstrated with examples.

5.1 Classification of the scene description discrepancy

Let us assume that the result of the verification procedure is correct, and that the eventual discrepancy between the hypothetic scene description and the acquired images is always caused by the difference between the actual scene and the scene description. That means that neither a type I, nor a type II error has occurred, This difference can be roughly classified into the following classes:

1. **Missing object** - an object actually present in the scene is missing in the scene description.

2. **False object** - the scene description contains an extra (false) object, which is actually not part of the scene.

3. **Misplaced object** - an object present in both the actual scene and the scene description is misplaced, i.e. its position, orientation or dimensions are not specified correctly.

4. **Misclassified object** - at the approximately same location there is a different kind of object in the actual scene and the description. The object has been misclassified.

5. **Combination** - there are more differences between the scene and the scene description. These can be treated as a combination of several single differences of the kind as given in 1 to 4.

Notice, that the distinction between classes 3, 4 and 5 is not strict. A "Misplaced object" can equally well be seen as a combination of "False object" and "Missing object". Ultimately, all differences can be reduced to the combination of "Missing objects" and "False objects".

Heuristic rules are developed for the classification into the above mentioned classes from the location of the discrepancies on the image plane. First, it must be determined, which part of the hypothetic scene description is projected onto the image plane on places, where the discrepancy has been detected. The rules can be summarized as follows:

- If there is no hypothetic object projected onto the image plane of the synthetic image at the location where the discrepancy occurs, it is concluded that there is a "Missing object", an object present in the scene, but not in the description.

- If the discrepancy occupies the place in the image plane where an object contained in the scene description is projected, it is concluded that this object is a "False object".

Based on this classification, a scene description correction is performed.

5.2   Correction of the scene description

In the case that a "False object" has been detected in the scene description, this object is merely excluded from the description, and the verification phase is repeated.

In the case that a "Missing object" has been detected in the scene, some predictive computer vision algorithm must be invoked to create a corrected description of the part of the scene, where the discrepancy has been detected. The algorithm can employ information about the location of the discrepancy. The knowledge of light conditions at this location may improve the performance of the predictive algorithm.

5.3   Feature-based techniques

Feature-based techniques use features extracted from the image to infer an interpretation of the image. The appropriate features are e.g. edges, contours and corners of objects. The knowledge of the scene illumination constrains the appearance of the 3-D features projected onto the image. This might be used to improve the performance of the conventional feature-based techniques. Camps, Shapiro and Haralick [9] have proposed such an object recognition/localization system, called PREMIO (PREdiction in Matching Images to Objects). To make predictions, they build a model called *Vision Model*, consisting of a topological model of objects, a surface physical model of objects, a model of light source and a sensor (a camera), and a model of detectors describing the performance of the feature detectors (i.e. early vision algorithms for the extraction of features) available to the system. The analysis of feature based techniques for the scene description correction in the case of known illumination falls beyond the scope of the thesis.

5.4   Techniques employing image shading

Another class of techniques to be considered are techniques that employ variations of image irradiance, or *shading*, exhibited in the region of an image, for the recovery of the shape of a part of a surface. We will discuss the use of *shape from shading* (SFS) and *photometric stereo*. The variations of image irradiance are interpreted by the variations of the orientation of the surface normal, while the reflectance properties of the surface are usually assumed to be constant over the investigated part of the surface. Shape from shading is a technique where the surface shape is recovered from a single image of the surface. In photometric stereo, more images of the surface, taken from the same position but under different lighting conditions, are processed. Each additional image presents an additional constraint for the surface orientation at each point. That is why photometric stereo is conceptually much simpler then shape from shading.

We have referred to SFS and photometric stereo already in section 3.1, where the problem of the reflectance parameters estimation was discussed. SFS and photometric stereo utilize the concept of the reflectance map, (section 3.1). The reflectance map specifies the relation between surface orientation and the image irradiance, and it is a property of the surface (it depends on the surface BRDF), and of the configuration of lighting. The knowledge of the *reflectance map* is a pre-requisite for the use of SFS, or photometric stereo. The reflectance map can either be obtained experimentally, or it can be computed from the surface BRDF and the distribution of light sources [41]. For a Lambertian surface, the reflectance map can be computed from the distribution of light sources up to a proportional factor. The fact, that reflectance maps can be computed by the scene illumination model already implemented in the system makes photometric stereo and SFS particularly attractive for the scene description correction.

For a surface $z = f(x, y)$, illumination by distant light sources, and parallel projection (see figure 3.2), image plane irradiance provides at every surface point a single constraint for the reconstructed surface orientation: see equation (3.1), p. 39. Since the normal of the surface $z = f(x, y)$ can be written as

$$\vec{n}(x, y) = \frac{1}{\sqrt{p^2 + q^2 + 1}} (-p, -q, 1)', \tag{5.1}$$

where

$$p = \frac{\partial f(x, y)}{\partial x} \quad \text{and} \quad q = \frac{\partial f(x, y)}{\partial y}, \tag{5.2}$$

the surface is the solution of the first order partial differential equation

$$E(x, y) - R(\vec{n}(x, y)) = 0. \tag{5.3}$$

Often the reflectance map is written directly in terms of the partial derivatives p, q

$$E(x, y) - R(p(x, y), q(x, y)) = 0, \tag{5.4}$$

and the pair (p, q) is called a *surface gradient*.

SFS solution by a characteristic strip expansion.   The solution of (5.4) can be found using a characteristic strip expansion as Horn has shown in [36]. Consider a four-dimensional space of points (x, y, p, q) and a parametric curve in this space $(x(t), y(t), p(t), q(t))$ for $t \in [t_0, t_1]$. If the fourtuple $(x(t_0), y(t_0), p(t_0), q(t_0))$ satisfies equation (5.4), and the condition

$$\frac{d}{dt}\left[E(x(t), y(t)) - R(p(t), q(t))\right] = 0$$

is met for $t \in [t_0, t_1]$, then obviously also the other points on the curve solve (5.4). Equivalently, we have

$$\frac{\partial E}{\partial x}\frac{dx}{dt} + \frac{\partial E}{\partial y}\frac{dy}{dt} - \frac{\partial R}{\partial p}\frac{dp}{dt} - \frac{\partial R}{\partial q}\frac{dq}{dt} = 0, \qquad (5.5)$$

where arguments of all functions have been omitted for conciseness. This condition is met by the choice:

$$\frac{dx}{dt} = \frac{\partial R}{\partial p}, \qquad \frac{dy}{dt} = \frac{\partial R}{\partial q},$$

$$\frac{dp}{dt} = \frac{\partial E}{\partial x}, \qquad \frac{dq}{dt} = \frac{\partial E}{\partial y}. \qquad (5.6)$$

An appropriate starting point for the expansion of solutions (characteristic strips) according to (5.6) is a point on the surface where the orientation is known, for instance the brightest point of the smooth surface, where the surface normal points to the source of illumination. The drawback of the method is that the solution is prone to accumulation of errors in the presence of noise.

SFS solution by minimization of an objective function.   Due to the presence of observation noise, variations of the surface albedo, *etc.*, the partial differential equation (5.3) will not hold even for the true surface shape. Instead of attempting to find an exact solution of (5.3), the surface orientations $\vec{n}(x, y)$ (or $p(x, y)$ and $q(x, y)$) can be found by minimizing some objective function using variational techniques [38, 51, 37, 69]. In the next step surface elevation is computed from the orientations. Also techniques that solve directly for the surface elevation $f(x, y)$ have been developed [50, 52, 53].

An appropriate objective function is for instance the functional

$$C_{\text{irr}} = \int\int_A \left[E(x, y) - R(p, q)\right]^2 dy\, dx, \qquad (5.7)$$

which leads under certain assumptions about irradiance observation error to a maximum likelihood reconstruction of the surface. The set A is the *area of interest*, the subset of the image plane where the reconstructed surface is projected.

It has been shown by many authors, that minimization of the objective function C_{irr} is not a well-posed problem. It is not sufficiently constrained, and consequently, it does not have a unique solution. There are two ways of constraining the problem further:

- A regularization of the problem by adding a regularization term to the objective function. A commonly used regularization term is a surface smoothness term involving second partial derivatives of the surface elevation:

$$C_{reg} = \int\int_A \left[f_{xx}(x,y)^2 + 2f_{xy}(x,y)^2 + f_{yy}(x,y)^2 \right] dy\, dx. \qquad (5.8)$$

  This term does not vanish for the actual surface shape, except for planar surfaces. The result obtained by minimization of a weighted sum $C_{tot} = w\, C_{irr} + (1-w)\, C_{reg}$, for some $0 < w < 1$, is biased. The surface is flattened (see also the discussion in [38]). The inclusion of the regularization term in the objective function regularizes the problem and allows to find a solution. However it should be avoided if other means to constrain the problem are available.

- The boundary conditions on the surface shape known to us can be applied to constrain the solution. Such a constraint is the orientation of a smooth surface at the brightest place where the surface normal points to the light source (see figure 5.1). At the places where specular highlights on the surface appear, the surface normal is known (it bisects the angle between the viewer and the source into two equal angles). At the occluding boundary of a smooth surface, the tangent plane is parallel to the optical axis, so the surface normal is perpendicular to the optical axis and the projection of the occluding boundary on the image plane.

Other methods for the SFS problem solution   Recently the solution of the SFS problem by means of the technique of dynamic programming [16, 67, 54] has been proposed.

5.4.1   Shape from shading with nonuniform illumination

For the sake of simplicity, most SFS techniques assume parallel projection and distant light sources. However, the SFS problem can be solved also for the
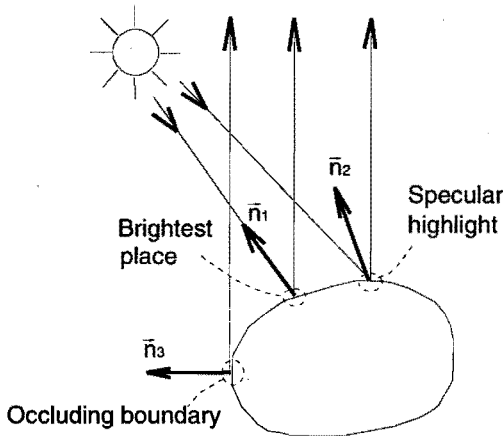
Figure 5.1: Constraints on the surface orientation.

case of perspective projection [36, 53]. For SFS with perspective projection and the direct computation of the surface elevation, the assumption of a uniform illumination by distant light sources can be relaxed. The reflectance map can be evaluated at an arbitrary point in space if we do have the scene illumination model. This is novel in the SFS algorithm described in this section. The algorithm can be used for the recovery of a surface at a specific location in the scene. The surface elevation is computed directly in a camera centred coordinate system by the minimization of an objective function.

5.4.1.1   Reconstructed surface representation

Let the camera centred coordinate system be defined in the following way: the origin O of the system coincides with the optical centre of the camera C, the z-axis is parallel to the optical axis of the camera, and x- and y-axes are parallel to the u- and v-axes of the two-dimensional image plane coordinate system (see figure 5.2). The distance of the image plane from the origin O will be denoted as f.

The surface to be reconstructed can be represented by a number of discrete points on the surface, that are projected onto a grid of points on the image plane. The grid of points on the image plane coincides with the corners of rectangular picture elements (PELs), so that the centre of each PEL will be in the middle between four grid points. Since a point in 3-D with coordinates $(x, y, z)'$, which is projected onto the image plane at location (u, v), can be
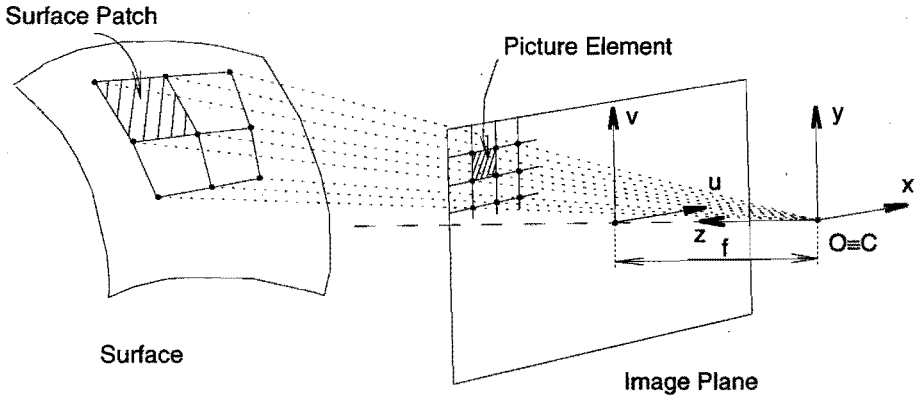
Figure 5.2: Representation of the surface in the camera centred coordinate system.

specified by a single parameter t,

$$(x, y, z)' = (tu,\ tv,\ tf)' = t\,(u, v, f)',\qquad\qquad (5.9)$$

the surface points are specified by a number of parameters equal to the number of grid points, each parameter corresponding to one node of the grid. If (i, j) are the indices of some particular PEL, then the grid point at the lower left corner of this PEL can be denoted by indices (i, j), lower right by $(i + 1, j)$, upper left by indices $(i, j + 1)$ and upper right by $(i + 1, j + 1)$ (see figure 5.3). Let (u_i, v_j) be the image plane coordinates of the node (i, j), let $(x_{i,j}, y_{i,j}, z_{i,j})$
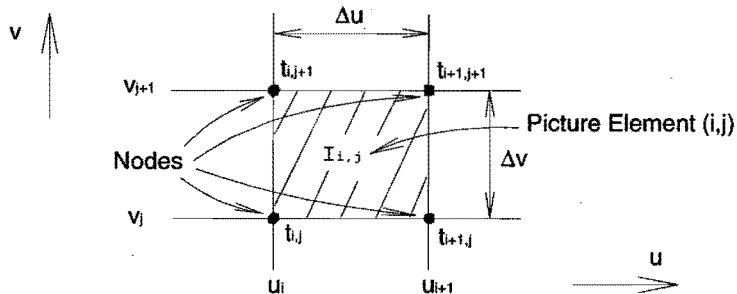
Figure 5.3: Location of nodes and picture elements.

be the coordinates of the surface point projected on (u_i, v_j), let $t_{i,j}$ be the parameter assigned to the node (i, j), and let $E_{i,j}$ be the irradiance of the PEL

(i, j). Let Δu and Δv be the spacing between PELs, i.e. $\Delta u = u_{i+1} - u_i$ and $\Delta v = v_{j+1} - v_j$.

The surface in between the grid points will be approximated by a bilinear interpolation of the parameter t, so that for $(u, v) \in [u_i, u_{i+1}] \times [v_j, v_{j+1}]$

$$t(u, v) = \frac{1}{\Delta u \, \Delta v} \left[(u_{i+1} - u)(v_{j+1} - v) \, t_{i,j} + (u - u_i)(v_{j+1} - v) \, t_{i+1,j} + \right.$$
$$\left. + (u_{i+1} - u)(v - v_j) \, t_{i,j+1} + (u - u_i)(v - v_j) \, t_{i+1,j+1} \right] \quad (5.10)$$

Then it is not difficult to show, that the gradient $(p_{i,j}, q_{i,j})$ of the interpolated surface at the point which is projected onto the centre of the PEL (i, j) with coordinates $u_c = (u_i + u_{i+1})/2$ and $v_c = (v_j + v_{j+1})/2$ is given by

$$p_{i,j} = \frac{f \, t_{u,c}}{t_c + t_{u,c} u_c + t_{v,c} v_c}, \quad (5.11)$$

$$q_{i,j} = \frac{f \, t_{v,c}}{t_c + t_{u,c} u_c + t_{v,c} v_c}, \quad (5.12)$$

where

$$t_c = t(u_c, v_c) = \frac{1}{4} \left(t_{i,j} + t_{i+1,j} + t_{i,j+1} + t_{i+1,j+1} \right) \quad (5.13)$$

$$t_{u,c} = \left. \frac{\partial t(u, v)}{\partial u} \right|_{(u_c, v_c)} = \frac{t_{i+1,j} + t_{i+1,j+1} - t_{i,j} - t_{i,j+1}}{2\Delta u} \quad (5.14)$$

$$(5.15)$$

$$t_{v,c} = \left. \frac{\partial t(u, v)}{\partial v} \right|_{(u_c, v_c)} = \frac{t_{i,j+1} + t_{i+1,j+1} - t_{i,j} - t_{i+1,j}}{2\Delta v} \quad (5.16)$$

5.4.1.2   Objective function

As has already been mentioned, SFS exploits for the reconstruction only the irradiance of the area of interest A. Let $\mathcal{A}$ accordingly be the index set containing indices of all PELs that are entirely inside the area of interest A, and let $\mathcal{B}$ be the index set of all nodes in the area of interest A.

For the discrete surface representation from the previous section, one term of a objective function will be

$$C_{\mathrm{irr}} = \Delta u \Delta v \sum_{(i,j) \in \mathcal{A}} \left[E_{i,j} - R(p_{i,j}, q_{i,j}) \right]^2. \quad (5.17)$$

In the limiting case of a PEL size approaching zero, this term of the objective function is equivalent to (5.7).

Next to this, the following constraints on the surface shape are applied:

- It is assumed that straight boundaries of the area of interest on the image plane correspond to edges of planar faces of an object (see figure 5.4). The surface gradient of the PELs neighbouring such a boundary have all
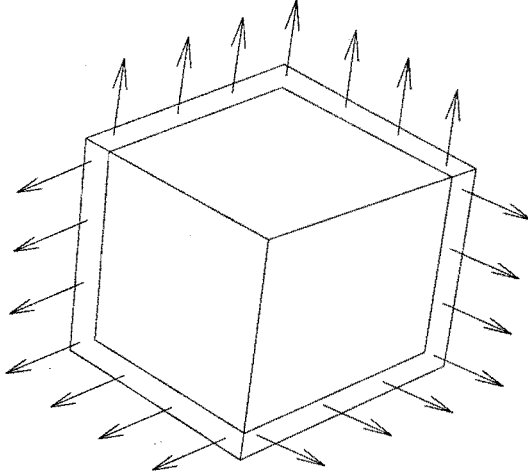
Figure 5.4: Constraints on the shape of a planar face bounded object.

the same gradient. This is captured by the following term of the objective function

$$C_{\text{edg}} = \sum_{(i,j) \in \mathcal{E}} (p_{i,j} - p_{\text{fce}})^2 + (q_{i,j} - q_{\text{fce}})^2 \qquad (5.18)$$

where $\mathcal{E}$ is the index set of indices of PELs that share the boundary $\partial \mathcal{A}$ of the set $\mathcal{A}$ and form a straight *digital line* (a straight line approximated by the discrete PELs). Application of the term C_{edg} for each straight boundary edge of the area of interest constrains the orientation of involved surface patches, and introduces two new variables p_{fce} and q_{fce} for each edge.

- The next constraint is the smoothness of the planar faces. It is assumed, that the regions where the image irradiance variation is negligible (possibly caused by observation noise only) are the projections of planar faces. There, a smoothness term resembling the term (5.8) is applied:

$$C_{\text{smt}} = \sum_{\{(i,j); |E_{i+1,j} - E_{i,j}| < T\}} \left[(p_{i+1,j} - p_{i,j})^2 + (q_{i+1,j} - q_{i,j})^2 \right]$$

$$+ \sum_{\{(i,j); |E_{i,j+1} - E_{i,j}| < T\}} \left[(p_{i,j+1} - p_{i,j})^2 + (q_{i,j+1} - q_{i,j})^2 \right] \,(5.19)$$

where T is the maximum value of the difference of irradiances of neighbouring PELs for the inclusion of the PELs into the smoothness term.

The total objective function has the form

$$C_{\text{tot}} = C_{\text{irr}} + a_1 \sum_{k=1}^{n_e} C_{\text{edg}}, + a_2 C_{\text{smt}}, \tag{5.20}$$

where n_e is the number of boundary edges used as constraints, $C_{\text{irr}}, C_{\text{smt}}$ and $C_{edg,k}$ are the objective function terms, and a_1 and a_2 are appropriate coefficients. After the substitution for p and q into all terms involved in C_{tot} from (5.11) and (5.12), the objective function is the function of parameters $t_{i,j}$, $\forall (i,j) \in \mathcal{B}$, specifying the surface.

5.4.1.3 Objective function minimization

For the minimization of the objective function the Gauss-Newton numerical optimization technique described in section 3.2.1 can be applied. In comparison with the problem of the scene reflectance and illumination parameters estimation from chapter 3, the number of arguments of the objective function is considerably larger, what makes the problem harder.

The objective function (5.20) can be rewritten in the form

$$C_{\text{tot}} = \|c(t)\|_2^2, = \sum_{l=1}^{n_c} c_l(t)^2, \tag{5.21}$$

where $t = (t_1, \dots, t_{n_n})'$ is a vector of parameters of all nodes from the set $\mathcal{B}$ (the nodes specifying the surface shape), $c(t) = (c_1(t), \dots, c_{n_c})'$ is a vector function of parameters t, and n_c is the total number of constraints coming from all terms of the objective function.

Following the same line of reasoning as in section 3.2.1, the problem of finding the vector $\hat{t}$ that minimizes the objective function,

$$\hat{t} = \arg \min_{t \in \mathcal{R}^{n_n}} C_{\text{tot}}, \tag{5.22}$$

leads to the following iterative scheme:

1. Choose an appropriate initial value t_0, the initial estimate of t.

2. For the k-th iteration, $k = 1, 2, \dots$, compute an update Δt_k by solving the system of linear equations

   $$\nabla(c(t_{k-1}))\Delta t_k = -c(t_{k-1}),$$

   $\nabla(c(t_{k-1}))$ is the matrix of partial derivatives, the Jacobian of the function $c()$.

3. Make an update of the vector t

$$t_k = t_{k-1} + \Delta t_k.$$

4. Check the condition for quitting the iteration loop. If this condition is not met, repeat from step 2, and if the condition is met, quit the iterations with the result

$$\hat{t} = t_k.$$

The form of the components of $c(t)$ depends on the kind of the constraint. The PEL irradiance delivers the component

$$c_l(t) = E_{i,j} - R(p_{i,j}, q_{i,j}),$$

each boundary edge pixel delivers two components

$$c_l(t) = p_{i,j} - p_{\text{fce}},$$

$$c_{l+1}(t) = q_{i,j} - q_{\text{fce}},$$

and finally, from the smoothness condition there are the pairs of components of the form

$$c_l(t) = p_{i+1,j} - p_{i,j},$$

$$c_{l+1}(t) = q_{i+1,j} - q_{i,j},$$

or

$$c_l(t) = p_{i,j+1} - p_{i,j},$$

$$c_{l+1}(t) = q_{i,j+1} - q_{i,j}.$$

5.4.1.4   Evaluation of the reflectance map

The reflectance map relates surface orientations to values of the image irradiance (or a surface radiance, which is proportional to the irradiance) under specified lighting conditions. Under the assumption of parallel projection and illumination by distant light sources, the same map is valid for the entire surface. With perspective projection and a non-homogeneous illumination a different map applies at different places of the scene. Fortunately, the SFS algorithm needs only samples from the map depending on the current estimate of the surface during the iterations. Furthermore, the partial derivatives with respect to the values of p and q at the same point are required.

The value of the reflectance map for a particular surface orientation can be obtained by placing (in a scene model) a sample surface patch at the required position with the required orientation, and evaluating the image irradiance caused

by this surface patch. Computationally this is equivalent to the evaluation of a single PEL of the synthetic image. The position of the surface element projected onto the centre of the particular PEL is obtained from the values $t_{i,j}, t_{i,j+1}, t_{i+1,j}$ and $t_{i+1,j+1}$ using formulas (5.13) and (5.9), while the surface normal is obtained using (5.11) through (5.16) and (5.1).

The reflection map depends on the BRDF of the surface. The BRDF is in general unknown, and a Lambertian reflection is usually assumed. Since the Lambertian reflection model is specified by a single parameter k_{dif}, the reflectance map can be obtained up to the one remaining unknown constant of proportionality, the coefficient k_{dif}. The coefficient can be estimated from at least one surface element with a known surface normal orientation. Heuristics can be used to locate such a surface element.

5.4.2 Photometric stereo

In photometric stereo, more images of the surface to be recovered obtained from the same camera position under different lighting conditions are processed. It was the first time proposed by Woodham [80, 81], and since then successfully used by many authors [68, 42, 58, 47, 70]. Photometric stereo is a significantly simpler technique than SFS, because there are more constraints on the surface orientation. The number of available constraints is equivalent to the number of acquired images. Two images, providing two constraints (3.2), together with global constraints for resolving the ambiguities, are sufficient for the surface reconstruction. Another advantage is that if enough constraints are available, besides surface orientation, additional unknown parameters can be recovered, e.g. the reflectance coefficient k_{dif}.

The implementation of photometric stereo for the scene description correction is straightforward. The surface representation, reflectance map evaluation, and the technique for objective function minimization are shared with SFS. The objective function differs from the one for a SFS problem, and has the form

$$C_{\text{PS}} = \sum_{k=1}^{n_v} C_{irr,k}, \qquad (5.23)$$

where n_v is the number of camera views, and $C_{irr,k}, k = 1, \ldots, n_v$ are the objective function terms obtained from the respective images in the same way as in (5.17).

Chapter 6

Conclusions and remarks

In this thesis a method has been described for the verification of scene descriptions using an imaging model: a model of the scene illumination and image formation processes. The difference between the verification algorithm described in the thesis, and conventional computer vision algorithms is the accuracy of the applied imaging model, and the completeness of the representation of the scene, illumination, and the sensor (camera). Some techniques that are able to detect and to correct inconsistencies between the actual scene and the scene description are also given.

To apply the verification method, the user must be in possession of the description of the scene, which is generated by any computer vision algorithm. For the application of the verification method, the following assumptions must be valid:

- the light reflectance of the surfaces in the scene can be described by a parametric model (e.g. the Torrance-Sparrow reflectance model, the Lambertian model),

- it is assumed that the scene consists of objects, with each object having constant reflectance properties over its faces, or surface parts,

- the unknown part of the scene lighting can be specified by a small number of parameters, so that it can be obtained by parameter estimation techniques.

There are several new results originating from this research:

- Both the luminance and reflectance parameters can be estimated from the acquired images using the imaging model. The ill-conditionedness of this approach is dealt with.

- The difference between two images is evaluated by filtering the difference image using an adapted North filter, and used to make decisions about the candidate scene descriptions. A new method for the selection of the filter for the special case of a composite alternative has been proposed.

- The reflectance maps for the shape from shading and photometric stereo techniques are evaluated using the imaging model.

- The use of an accurate imaging model originating in computer graphics for a computer vision application is novel, and potentially brings a considerable improvement of the performance of the computer vision algorithms. The field of applications of this technique is in robotics, and man-made environments. Demand is also growing in virtual reality applications, where the aim is to create the descriptions of existing 3-D scenes automatically, without the need for human interaction.

By using the imaging model, the system is able to exploit more information contained in the image than by using the conventional clues only. Experiments have shown that the system is feasible, and is able to make appropriate decisions about man-made scenes. The reliability of the complete system (a computer vision system and the scene description verification system) is improved when compared with the system without the verification. The time performance (the computation time) of the system is satisfactory for static scenes. The decisive power of the verification system depends on the kind of inconsistency between the scene and its description. The inconsistencies might be missed under certain circumstances, e.g. in poorly illuminated places, or when the reflectance of objects does not differ much from the background. Great care must be taken with the choice of the threshold level in the scene verification algorithm, affecting the probability of miss, and the probability of a false alarm.

There are several ways to further improve the decisive power of the system without having to make a compromise between the above-mentioned probabilities.

- In the thesis the processing of gray-scale intensity images is discussed. The system can be extended in a straightforward way for the processing of multiband (colour) images. The dependence of the surface reflectance on the wavelength, as well as the spectral distribution of light sources must then be taken into account. We expect, that this will improve the decisive power of the verification system.

- It is a well-known fact that lighting plays a crucial role in computer vision applications. The ability to control the scene lighting (to switch on and off the light sources) is expected to increase the decisive power of the system. With the ability to process more images acquired under different illumination, the possibility of missing an inconsistency between the scene

and the description will decrease. Moreover, photometric stereo using the imaging model can be used for the correction of the inconsistent scene description. The questions about the placement of the light sources and cameras in such a way that an optimal performance of the system (a predictive computer vision system plus the verification system) is achieved are also interesting. The imaging model might provide a good insight into this problem.

- There would be an interest for fast verification techniques that use only a small number of appropriately chosen picture elements. Such techniques would allow cooperation with techniques using motion of the camera (structure from motion), or techniques for dynamic scenes. The verification would increase the reliability of such systems. Stochastic ray-tracing (because of the independent computation for particular picture elements), and the estimation and verification techniques developed in this thesis can be efficiently used for this purpose.

- The system can be extended to handle a broader class of scenes, e.g. for the scenes with textured surfaces. Such extensions involve the issues of representation, parameterization, parameter estimation, and evaluation of the difference between actual and synthesized images. An ultimate long-term goal might be a system for generic scenes.

Bibliography

1 J. Arvo and D. Kirk. Particle transport and image synthesis. *Computer Graphics,* 24(4):63–66, Aug. 1990.

2 N. Ayache and O. D. Faugeras. Building, registrating, and fusing noisy visual maps. *The International Journal of Robotics Research,* 7(6):45–65, 1988.

3 R. Bajcsy and L. Lieberman. Texture gradient as a depth cue. *Computer Graphics & Image Processing,* 5(1):52–67, 1976.

4 P. Beckmann and A. Spizzochino. *The Scattering of Electromagnetic Waves from Rough Surfaces.* Pergamon Press, 1963.

5 R. E. Blahut. *Principles and Practice of Information Theory.* Addison-Wesley, 1987.

6 J. F. Blinn. Models of light reflection for computer synthesized pictures. *Computer Graphics,* 11(2):192–198, 1977.

7 C. Bouville et al. Monte-Carlo integration applied to an illumination model. In D. A. Duce and P. Jancene, editors, *EUROGRAPHICS'88,* pages 483–497. Elsecier Science Publishers B. V., 1988.

8 R. A. Brooks. Symbolic reasoning among 3-D models and 2-D images. *Artificial Intelligence,* 17:285–348, 1981.

9 O. I. Camps, L. G. Shapiro, and R. M. Haralick. PREMIO: An overview. In *IEEE Workshop on Directions in Automated CAD-Based Vision,* pages 11–21, Maui, Hawaii, June 1991.

10 R. L. Cook, T. Porter, and L. Carpenter. Distributed ray tracing. *Computer Graphics,* 18(3):137–145, July 1984.

11 R. L. Cook and K. E. Torrance. A reflectance model for computer graphics. *Computer Graphics,* 15(3):307–316, Aug. 1981.

12 R. Courant and D. Hilbert. *Methods of Mathematical Physics*, volume 1. John Wiley & sons, 1953.

13 M. G. Cox. The least squares solution of overdetermined linear equations having band or augmented band structure. *IMA Journal of Numerical Analysis*, 1:3–22, 1981.

14 M. H. DeGroot. *Optimal Statistical Decisions*. McGraw-Hill Inc., 1970.

15 P. Dunias, M. Hanajík, and N. G. M. Kouwenberg. Knowledge-based machine vision. In *ISPRS Commission III Symposium on Spatial Information from Digital Photogrammetry and Computer Vision*, pages 201–205, Munich, Germany, Sept. 1994.

16 P. Dupuis and J. Oliensis. Direct method for reconstructing Shape from Shading. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 453–458, Champaign, Illinois, June 1992.

17 P. J. Flynn and A. K. Jain. Bonsai: 3-D object recognition using constrained search. *IEEE Trans. Pattern Anal. Machine Intell.*, 13(10):1066–1075, 1991.

18 D. Forsyth and A. Zisserman. Mutual illumination. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 466–473, 1989.

19 D. Forsyth and A. Zisserman. Reflections on shading. *IEEE Trans. Pattern Anal. Machine Intell.*, 13(7):671–679, 1991.

20 P. Fua and Y. G. Leclerc. Using 3-dimensional meshes to combine image-based and geometry-based constraints. In *Proc. of the 5th European Conference on Computer Vision*, volume 2, pages 281–291, 1994.

21 P. E. Gill, W. Murray, and M. H. Wright. *Practical Optimization*. Academic Press, 1981.

22 C. M. Goral, K. E. Torrance, D. P. Greenberg, and B. Battaile. Modeling the interaction of light between diffuse surfaces. *Computer Graphics*, 18(3):213–222, July 1984.

23 R. Hall and D. Greenberg. A testbed for realistic image synthesis. *Computer Graphics and Applications*, 3(8):10–20, 1983.

24 J. H. Halton. A retrospective and prospective survey of the monte carlo method. *SIAM Review*, 12(1):1–63, Jan. 1970.

25 M. Hanajík. Interpretation of line drawings of surveyed work pieces. In *11th Benelux Meeting on Systems and Control, Abstracts, Lecture Notes*, page 36, Veldhoven, The Netherlands, March 4-6, 1992.

26 M. Hanajík. An analytical approach to inexact matching of labelled graphs. In *12th Benelux Meeting on Systems and Control, Abstracts, Lecture notes*, page 26, Houffalize, Belgium, March 3-5, 1993.

27 M. Hanajík. Accuracy analysis of the model based 3-D computer vision using 2-D imaging. In *13th Benelux Meeting on Systems and Control, Abstracts, Lecture notes*, page 123, Veldhoven, The Netherlands, March 4-6, 1994.

28 M. Hanajík. Efficient data driven algorithm for the model based 3-D scene reconstruction from perspective images. In *ISPRS Commission III Symposium on Spatial Information from Digital Photogrammetry and Computer Vision*, pages 347-352, Munich, Germany, Sept. 1994.

29 M. Hanajík. The verification of 3-D scene interpretations. In *14th Benelux Meeting on Systems and Control, Final Program Abstracts, Course Notes*, page 139, Houthalen, Belgium, March 29-31, 1995.

30 M. Hanajík and N. G. M. Kouwenberg. Model driven measurement of industrial object dimensions from camera images. In *1st International Symposium on Advances in Intelligent Computer Integrated Manufacturing System*, pages 304-309, Seoul, Korea, Nov. 1994.

31 M. Hanajík, F. J. Kylstra, and R. G. van Vliet. An analytical approach to the matching of attributed graphs. In *Proceedings of the 8th Scandinavian Conference on Image Analysis*, pages 419-425, Tromsø, Norway, May 25-28, 1993.

32 M. Hanajík and R. G. van Vliet. Real time image processing for fast seam tracking. In D. Chetverikov and W. G. Kropatsch, editors, *Computer Analysis of Images and Patterns*, Lecture Notes in Computer Science 719, pages 698-705. Springer-Verlag, Budapest, Hungary, 1993.

33 R. M. Haralick and L. G. Shapiro. Glossary of computer vision terms. *Pattern Recognition*, 24:69-93, 1991.

34 M. Herman. Representation and incremental construction of a three-dimensional scene model. In A. Rosenfeld, editor, *Techniques for 3-D Machine Perception*, pages 149-183. North-Holland, 1986.

35 L. Hong and D. Brzakovic. 3D scene reconstruction from noisy image sequences using data fusion. *Control Engineering Practice*, 2(5):825-831, 1994.

36 B. K. P. Horn. Obtaining shape from shading information. In P. H. Winston, editor, *The Psychology of Computer Vision*, pages 115-155. McGraw-Hill, New York, 1975.

37 B. K. P. Horn. Height and gradient from shading. *International Journal of Computer Vision*, 5:584-595, 1990.

38 B. K. P. Horn and M. J. Brooks. The variational approach to shape from shading. *Computer Vision, Graphics & Image Processing*, 33(2):174–208, 1986.

39 B. K. P. Horn and M. J. Brooks. *Shape from Shading*. The MIT Press, 1989.

40 B. K. P. Horn and B. G. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

41 B. K. P. Horn and R. W. Sjoberg. Calculating the reflectance map. *Applied Optics*, 18(11):1770–1779, 1979.

42 K. Ikeuchi. Determining surface orientations of specular surfaces by using the photometric stereo method. *IEEE Trans. Pattern Anal. Machine Intell.*, 3(6):661–669, 1981.

43 K. Ikeuchi and K. Sato. Determining reflectance properties of an object using range and brightness images. *IEEE Trans. Pattern Anal. Machine Intell.*, 13(11):1139–1153, 1991.

44 D. S. Immel, M. F. Cohen, and D. P. Greenberg. A radiosity method for non-diffuse environments. *Computer Graphics*, 20(4):133–142, Aug. 1986.

45 J. T. Kajiya. The rendering equation. *Computer Graphics*, 20(4):143–150, Aug. 1986.

46 G. Kay and T. Caelli. Inverting an illumination model from range and intensity maps. *CVGIP: Image Understanding*, 59(2):183–201, 1994.

47 B. Kim and P. Burger. Depth and shape from shading using the photometric stereo method. *CVGIP: Image Understanding*, 54(3):416–427, 1991.

48 A. J. F. Kok and F. W. Jansen. Adaptive sampling of area light sources in ray tracing including diffuse interreflection. In A. Kilgour and L. Kjelldahl, editors, *EUROGRAPHICS'92*, volume 11, pages C–289–C–298, 1992.

49 J. H. Lambert. Photometria sive de mensura et gratibus luminus, colorum et umbrae, eberhard klett, augsburg, 1760. In W. Engelman, editor, *Translated in: Lambert's Photometrie*. Ostwald's Klassiker der Exacten Wissenschaften, 1892.

50 Y. G. Leclerc and A. F. Bobick. The direct computation of height from shading. In *Proc. of the IEEE Conference on Computer Vision*, pages 552–558, 1991.

51 D. Lee. A provably convergent algorithm for shape from shading. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 478–485, Ann Arbor, MI, June 1988.

52 K. M. Lee and C.-C. J. Kuo. Shape from shading with linear triangular element surface model. *IEEE Trans. Pattern Anal. Machine Intell.*, 15(8):815–822, 1993.

53 K. M. Lee and C.-C. J. Kuo. Shape from shading with perspective projection. *CVGIP: Image Understanding*, 59(2):202–212, 1994.

54 P. L. Lions, E. Rouy, and A. Tourin. Shape-from-Shading, viscosity solutions and edges. *Numerische Mathematik*, 64:323–353, 1993.

55 D. G. Lowe. Three-dimensional object recognition from single two-dimensional images. *Artificial Intelligence*, 31:355–395, 1987.

56 D. G. Lowe. Fitting parameterized three-dimensional models to images. *IEEE Trans. Pattern Anal. Machine Intell.*, 13:441–450, 1991.

57 J. F. Murray-Coleman and A. M. Smith. The automated measurement of BRDFs and their application to luminaire modeling. *Journal of the Illumination Engineering Society*, Winter 1990.

58 S. K. Nayar, K. Ikeuchi, and T. Kanade. Determining shape and reflectance of hybrid surfaces by photometric sampling. *IEEE Transactions on Robotics and Automation*, 6(4):418–431, Aug. 1990.

59 S. K. Nayar, K. Ikeuchi, and T. Kanade. Surface reflection: Physical and geometrical perspectives. *IEEE Trans. Pattern Anal. Machine Intell.*, 13(7):611–634, 1991.

60 J. Neyman and E. S. Pearson. On the use and interpretation of certain test criteria for purpose of statistical inference. *Biometrika*, 20A:175–240, 263–294, 1928.

61 J. Neyman and E. S. Pearson. On the problem of the most efficient test of statistical hypotheseses. *Phil. Trans. Roy. Soc. London, Series A*, 231:289–337, 1933.

62 H. V. Nguyen and M. Hanajík. 3-D scene reconstruction from image sequences. In V. Hlaváč and R. Šára, editors, *Computer Analysis of Images and Patterns*, Lecture Notes in Computer Science 970, pages 182–189, Prague, Czech Rep., Sept. 1995. Springer-Verlag.

63 F. E. Nicodemus et al. *Geometrical considerations and nomenclatyre for reflectance*. NBS Monograph 160, National Bureau of Standards, Oct. 1977.

64 B. Phong. Illumination for computer generated pictures. *Commun. ACM*, 18:311–317, 1975.

65 K. Prazdny. Egomotion and relative depth map from optical flow. Technical report, Computer Science Dept., Univ. Essex, Mar. 1979.

66 W. Purgathofer. A statistical method for adaptive stochastic sampling. In A. A. G. Requicha, editor, *EUROGRAPHICS'86*, pages 145–152. Elsecier Science Publishers B. V., 1986.

67 E. Rouy and A. Tourin. A viscosity solutions approach to Shape-from-Shading. *SIAM J. Numer. Anal.*, 29(3):867–884, 1992.

68 W. M. Silver. Determining shape and reflectance using multiple images. Master's thesis, Massachusetts Institute of Technology, June 1980.

69 R. Szeliski. Fast Shape from Shading. *CVGIP: Image Understanding*, 53(2):129–153, 1991.

70 H. D. Tagare and R. J. P. deFigueiredo. A theory of Photometric Stereo for a class of diffuse non-Lambertian surfaces. *IEEE Trans. Pattern Anal. Machine Intell.*, 13(2):133–152, 1991.

71 H. D. Tagare and R. J. P. deFigueiredo. Simultaneous estimation of shape and reflectance map from Photometric Stereo. *CVGIP: Image Understanding*, 55(3):275–286, 1992.

72 K. E. Torrance and E. M. Sparrow. Theory for off-specular reflection from roughened surfaces. *Journal of the Optical Society of America*, 57:1105–1114, Sept. 1967.

73 F. Ulupinar and R. Nevatia. Constraints for interpretation of line drawings under perspective projection. *CVGIP: Image Understanding*, 53:88–96, 1991.

74 J. R. Wallace, M. F. Cohen, and D. P. Greenberg. A two-pass solution to the rendering equation: A synthesis of ray tracing and radiosity methods. *Computer Graphics*, 21(4):311–320, July 1987.

75 D. Waltz. Understanding line drawings of scenes with shadows. In P. H. Winston, editor, *The Psychology of Computer Vision*, pages 19–91. McGraw-Hill, New York, 1975.

76 G. J. Ward. Measuring and modeling anisotropic reflection. *Computer Graphics*, 26(2):265–272, July 1992.

77 G. J. Ward, F. M. Rubinstein, and R. D. Clear. A ray tracing solution for diffuse interreflection. *Computer Graphics*, 22(4):85–92, Aug. 1988.

78 J. Weng, P. Cohen, and M. Herniou. Camera calibration with distortion models and accuracy evaluation. *IEEE Trans. Pattern Anal. Machine Intell.*, 14:965–980, 1992.

79 T. Whitted. An improved illumination model for shaded display. *Communications of the ACM*, 23(6):343–349, June 1980.

80 R. J. Woodham. Reflectance map techniques for analyzing surface defects in metal castings. Technical Report AI-TR-457, AI Laboratory, M.I.T., June 1978.

81 R. J. Woodham. Photometric method for determining surface orientation from multiple images. *Optical Engineering*, 19(1):139–144, 1980.

Appendix A

Imaging model summary

This appendix contains information concerning the implementation of the imaging model which probably does not belong to chapter 2.

A.1 Stochastic ray tracing

The implementation of stochastic ray tracing is simply an evaluation of integral (2.38), page 31, at places where the traced ray interacts with a surface. This is done in the following way:

- The rays are traced to all point light sources. In this way the first term on the right hand side of equation (2.42) is evaluated. The area light sources are approximated by arrays of point light sources, therefore the second term is currently not evaluated. The diffuse as well as the specular reflection of the light coming directly from light sources are modelled in this way.

- The term I_2 (2.40) that models the diffuse reflection caused by an indirect illumination is evaluated using stratified sampling of the illumination hemisphere. We use the distribution of orientations of sampling rays proposed by Ward, Rubinstein and Clear [77]. We decrease the number of generated rays at each recursion level. The numbers of generated rays were in the most of experiments 8 rays at the first level (the first interaction), and 2 rays at the second level. The generation of third level rays had no effect on the generated images.

- The term I_3 (2.41) that models the specular part of reflection of an indirect illumination is neglected.

When evaluating the integral, the reflectance model of the interacting surface is addressed.

A.2   Surface reflectance model

We have obtained the most satisfactory results with a simplified Torrance-Sparrow reflectance model. The Fresnel factor was approximated by a constant, which was then absorbed in the parameter k_{spec}. This reflectance model is described by only three parameters: the coefficients k_{dif}, k_{spec}, and the angle β_{h}.

A.3   Collaboration between imaging model and parameter estimation

An imaging model and a parameter estimation procedures were implemented as a single program unit. The reason therefor was to maintain the efficiency of computations. The parameter estimation procedure addresses the imaging model repeatedly when computing successive iterations.

Stochastic ray tracing functions in two stages: In the first stage rays used for the computation of an image irradiance are traced. This involves the generation of random ray directions and solution of the hidden surface problem when checking for the intersections with scene objects. The results are stored in operating memory, or a disk. The computation of the irradiance is deferred to the second stage.

In the second stage the image irradiance, and if required also partial derivatives of the irradiance with respect to the parameters are computed using data structures from the first stage. Only the second stage must be re-invoked when the parameters change.

A.4   Scene and lighting description.

One of the inputs of the imaging model is the description of the scene and the lighting. In the current implementation of the verification system, the description is a text file with a specific format, which contains the specification

of objects in the scene, light sources, and the declaration of the radiometric parameters to be estimated.

Scene objects. Each object of the scene is represented by its surface, consisting of a number of planar faces. Each face has a polygonal shape, and is represented by its vertices. Optical properties of the faces are specified by parameters of the surface reflectance model. It is assumed that these parameters are constant over the whole face, i.e. a one set of parameters is assigned to the face. If it is appropriate, a single set of parameters describes the surface properties of the entire object, or more objects. The values of the reflectance parameters might be given, or they might be estimated by the estimation procedure. The following is an example of the specification of the object CUBE.

```
OBJECT{ CUBE,
        REFLEC(0, 0.21, 0.7, 12.3),
        POINT(0,0,0), POINT(500,0,0), POINT(500,500,0),
        POINT(0,500,0), POINT(0,0,500), POINT(500,0,500),
        POINT(500,500,500), POINT(0,500,500),
        FACE(1,2,6,5), FACE(4,8,7,3), FACE(2,3,7,6),
        FACE(1,5,8,4), FACE(5,6,7,8) }
```

It contains the specification of the reflectance model parameters (REFLEC( )), the specification of coordinates of eight vertices(POINT( )), and the specification of five faces (FACE( )).

Curved surfaces can not be modelled using a chosen representation. They must be approximated by a number of planar facets.

Light sources In the current implementation only point light sources can be represented. The point light source is specified by its position, and its radiant flux. The following is an example of the point light source specification:

```
LIGHT{ INTENSITY(LT_INTS), POSITION(3010,-2080,3000) }
```

Area light sources can be approximated by a array of point light sources with the radiant flux given by a single parameter.

Radiometric parameters The surface reflection parameters and the light source intensity can be estimated by the parameter estimation procedure. The parameters must be declared, and the information required by a parameter estimation procedure must be provided. This information includes an identifier, an initial value of the parameter, lower and upper bounds, and a tolerance for the estimated results. The following is an example of the parameter declaration:

```
PARAMETER{LT_INTS, INITVAL=300, LOBOUND=0, HIBOUND=2000, TOLERANCE=0.5}
```

A.5   Camera specification

The camera is represented by a pinhole camera model, extended by a lens distortion model proposed by Weng, Cohen and Herniou [78]. The camera model is specified by 16 parameters: 5 parameters specify intrinsic camera properties, 5 parameters specify the camera and lens distortion, and 6 extrinsic parameters specify the camera position and orientation. The camera model parameters are obtained by a camera calibration procedure, and passed to the imaging model.

A.6   Input conversion

The scene description is generated automatically by the computer vision system. Next to this, an additional information about the known part of environment is mostly available in some form. A conversion program has been written which converts the output of the computer vision system from its own specific format to the format accepted by the imaging model. At the same time the additional information is merged. This prevents a certain level of portability of the system.

Appendix B

Karhunen-Loeve Expansion

The Karhunen-Loeve expansion, similarly to the Fourier expansion, is an expansion of a finite sequence of numbers (a vector) into a series of orthogonal sequences (vectors). The Karhunen-Loeve expansion is used for multivariate statistical analysis.

An m-dimensional random vector $X = (X_1, \ldots, X_m)'$ can be expressed as the sum

$$X = \sum_{k=1}^{m} V_k \psi_k \tag{B.1}$$

where column vectors ψ_k, $k = 1, \ldots, m$ form an orthonormal base:

$$\psi_k' \psi_l = \begin{cases} 1 & \text{for } k = l, \\ 0 & \text{for } k \neq l. \end{cases} \tag{B.2}$$

The prime in the formula above denotes the transpose. The expansion coefficients are calculated as the inproduct of the vector X with the base vectors,

$$V_k = \psi_k' X, \qquad k = 1, \ldots, m. \tag{B.3}$$

We seek such an orthonormal base ψ_k, $k = 1, \ldots, m$, that the expansion coefficients are uncorrelated:

$$E[V_k V_l] = \begin{cases} \lambda_k & \text{for } k = l, \\ 0 & \text{for } k \neq l. \end{cases} \tag{B.4}$$

Such an expansion is called the Karhunen-Loeve expansion. The following theorem tells us how to obtain such orthonormal base.

Theorem 2 *Let Q be the covariance matrix of the random vector X,*

$$Q = cov(X) = E\{[X - E(X)][X - E(X)]'\}.$$

Let ψ_k and λ_k for $k = 1, \ldots, m$ be the eigenvectors and eigenvalues of the covariance matrix Q. The expansion coefficients

$$V_k = \psi_k' X$$

are uncorrelated random variables of variance λ_k. If X is jointly gaussian, then V_k, $k = 1, \ldots, m$ are independent gaussian random variables.

Proof: Normalized eigenvectors of the symmetric positive definite matrix Q always form an orthonormal basis. Therefore

$$E[V_k, V_l] = E[\psi_k' X X' \psi_l] = \psi_k' E[X X'] \psi_l$$

$$= \lambda_k \psi_k' \psi_l = \lambda_k \delta_{kl}.$$

Hence, the random variables V_k are uncorrelated, with variance λ_k.

If X is jointly gaussian, coefficients V_k obtained as a linear combination of components of X are also gaussian, and consequently also independent.   □

We may treat the expansion coefficients as a vector $V = (V_1, \ldots, V_m)'$ and benefit from a more efficient notation. Furthemore, we may denote by Ψ the m by m matrix composed of the column vectors ψ_k, the eigenvectors of the covariance matrix Q, $\Psi = (\psi_1 \ldots \psi_m)$. We may write, equivalently to (B.1) and (B.3), that $X = \Psi V$, and $V = \Psi' X$.

In the next theorem, a formula which is analog to Parseval's formula for Fourier transform coefficients is given.

Theorem 3 *Let $u = (u_1, \ldots, u_m)'$ and $v = (v_1, \ldots, v_m)'$ be the Karhunen-Loeve expansion coefficients of two m-dimensional vectors x and y (two realizations of the random vector X), respectively. Then*

$$x'y = \sum_{k=1}^{m} u_k v_k = u'v$$

Proof:

$$x'y = \left(\sum_{k=1}^{m} u_k \psi_k \right)' \sum_{l=1}^{m} v_l \psi_l = \sum_{k=1}^{m} \sum_{l=1}^{m} u_k v_l \psi_k' \psi_l$$

$$= \sum_{k=1}^{m} \sum_{l=1}^{m} u_k v_l \delta_{kl} = \sum_{k=1}^{m} u_k v_k$$

$\square$

Another property useful for the derivation of the North filter is the following:

Theorem 4 *Let us denote $\psi_k^{(i)}$ the i-th component of the vector ψ_k. The Karhunen-Loeve orthonormal base vectors satisfy*

$$\sum \psi_k^{(i)} \psi_k^{(j)} = \delta_{ij},$$

Proof: Columns of matrix Ψ are normalized eigenvectors of the symmetric matrix Q. Consequently, the matrix Ψ is orthonormal. The proposition of the theorem follows from the fact, that an orthonormal matrix Ψ has the property

$$\Psi\Psi' = \Psi'\Psi = I,$$

where I is an identity matrix. $\square$

Theorem 5 *The Karhunen-Loeve orthonormal base vectors satisfy*

$$\sum_{k=1}^{m} \lambda_k \psi_k^{(i)} \psi_k^{(j)} = s_{ij},$$

where s_{ij} is the covariance between the i-th and j-th component of vector X (the (i,j)-th element of matrix Q).

Proof: Matrix Ψ is composed of the eigenvectors ψ_k of the symetric matrix Q. Let Λ be the diagonal matrix with the respective eigenvalues $\lambda_k, k = 1, \ldots, m$ on the diagonal. From the definition of eigenvectors and eigenvalues

$$Q\Psi = \Psi\Lambda.$$

After right multiplication by Ψ', we obtain

$$Q = \Psi\Lambda\Psi'$$

The proposition of the theorem is equivalent to the last formula written elementwise. $\square$

Theorem 6 *Let λ_k be the eigenvalue of the covariance matrix Q, and let $u_k, v_k, k = 1, \ldots, m$ be the Karhunen-Loeve expansion coefficients of x and y, respectively. Then*

$$x = Qy$$

if and only if

$$v_k = \frac{u_k}{\lambda_k} \qquad for\ k = 1, \ldots, m$$

Proof:

$$x = \sum_{k=1}^{m} u_k \psi_k \qquad y = \sum_{k=1}^{m} v_k \psi_k \tag{B.5}$$

Therefore

$$\sum_{k=1}^{m} u_k \psi_k = Q \sum_{k=1}^{m} v_k \psi_k$$

$$= \sum_{k=1}^{m} v_k \lambda_k \psi_k$$

the theorem follows from the uniqueness of the expansion.   □

Appendix C

Matched filter for a composite alternative

Our goal is to design a decision procedure with the structure given in figure 4.2, page 60, for a decision problem with the composite alternative:

$$
\begin{array}{ll}
H_0: & X \sim \mathcal{N}(0, Q) \\
H_1: & X \sim \mathcal{N}(p, Q); \quad p \in \mathcal{P},
\end{array}
\tag{C.1}
$$

where set $\mathcal{P}$ is a countable set of vectors representing the patterns to be detected in a noisy image. We propose to use the filter that satisfies the criterion (4.33), page 65. As it is shown in section 4.2.3, this leads to the solution of the problem

$$
s = \operatorname*{argmax}_{s \in \mathcal{R}^m} \left\{ \min_{t \in \mathcal{T}} t's \right\}
\tag{C.2}
$$

subject to the constraint $s's = 1$. The set $\mathcal{T}$ is obtained from the set $\mathcal{P}$ by the Karhunen-Loeve transformation of the elements of the set $\mathcal{P}$ and their substitution according to (4.30). We will derive the solution of the problem for the case of a finite set $\mathcal{T} = \{t_1, \ldots, t_n\}$, with linearly independent elements, that means $\sum_{i=1}^{n} c_i t_i \neq 0$ for all $(c_1, \ldots, c_n) \neq (0, \ldots, 0)$. This also implies $n \leq m$.

The problem has the following geometrical interpretation. The set $\mathcal{T}$ contains n points t_i, $i = 1, \ldots, n$ in the m-dimensional space. Let σ be a hyperplane passing through the origin O with the normal direction given by the unit length vector s. The dot products $t_i's$ is an oriented distance of the point t_i from the hyperplane σ (i.e. a positive distance for points in one half space, and negative in the other half space defined by the hyperplane σ). The goal is thus to find such surface normal orientation s that the distance of the point $t_i \in \mathcal{T}$ which is the closest to the plane σ is maximized.

The following proposition restricts the solution set:

Proposition 1 *Let $\mathcal{A}_T$ be the linear subspace spanned by vectors t_i, $i = 1,\ldots,n$ contained in the set T. Then the solution of the problem (C.2), the vector s, belongs to a subspace $\mathcal{A}_T$ and all its coordinated with respect to the base $t_1,\ldots,t_n$ are non-negative. Hence, for the vector s we can write:*

$$s = \sum_{i=1}^{n} c_i t_i, \qquad \text{with } c_i \geq 0 \text{ for } i = 1,\ldots,n \tag{C.3}$$

It can be shown that there exists a unique hyperplane π with the normal vector $p = (p_1,\ldots,p_m)'$ from the subspace $\mathcal{A}_T$, which contains all points $t_i, i = 1,\ldots,n$. For the normal vector we can write

$$p = \sum_{i=1}^{n} a_i t_i. \tag{C.4}$$

The coordinates a_i, $i = 1,\ldots,n$ of the normal vector p with respect to the base $t_1,\ldots,t_n$ are the solution of a system

$$t_i'p = t_i'(a_1 t_1 + \cdots + a_n t_n) = 1, \quad \text{for } i = 1,\ldots,n. \tag{C.5}$$

On the right hand side of equations (C.5) there might be an arbitrary constant, since the length of the normal vector p can be scaled. Let us denote T the $m \times n$ matrix with the columns $t_i, i = 1,\ldots,n$. The above equations constitute a system $T'Ta = (1,\ldots,1)'$, where $a = (a_1,\ldots,a_n)'$ specifies the normal vector p. Matrix T, as well as the matrix $T'T$, have full column rank, which confirms the uniqueness of the solution.

The normal vector p of the plane π after scaling to a unit length is a candidate for the solution s. The case when the vector p is the solution of the problem (C.2) is shown on a two-dimensional example in figure C.1a. In this case the planes σ and π are parallel. In some cases the normal vector p does not satisfy the requirements of the proposition 1, and consequently is not a solution. Such case is shown in figure C.1b. Then, the solution can be obtained by the following algorithm:

Algorithm C.1

1. The coordinates $(a_1,\ldots,a_n)$ of the vector p with respect to the base $t_1,\ldots,t_n$ are computed by solving the system C.5.

2. The coordinates of the solution s with respect to the base $t_1,\ldots,t_n$ are determined as

$$c_i = \max\{a_i, 0\}, \qquad \text{for } i = 1,\ldots,n, \tag{C.6}$$

   so that they satisfy the requirements of the proposition 1.
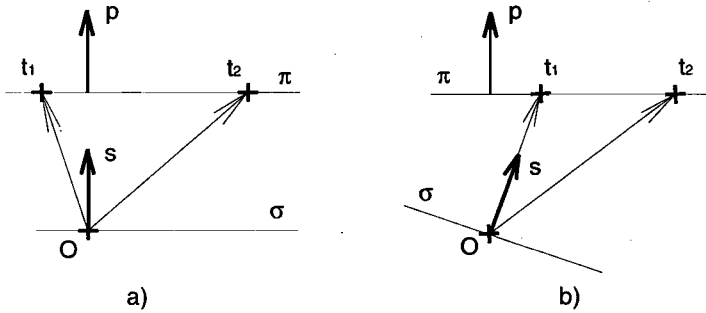
Figure C.1: Hyperplane π containing the points $t_i \in T$ and the solution vector s.

3. The solution vector is computed and scaled to the unit length:

$$ s = \frac{\sum\limits_{i=1}^{n} c_i t_i}{\left| \sum\limits_{i=1}^{n} c_i t_i \right|} \tag{C.7} $$

Proposition 2 *The vector s computed by the algorithm C.1 is a solution of a problem (C.2).*

Once this problem is solved, expansion coefficients $w = (w_1, \ldots, w_m)'$ of the required filter are obtained from the found vector s by a back substitution via the inverse of (4.29), page 64. The coefficients f of the filter for the decision procedure are obtained by the inverse Karhunen-Loeve transformation of the vector w.

When computing the detection filter coefficients, the Karhunen-Loeve transform and inverse transform must be used in general. In the case of a diagonal covariance matrix Q, the Karhunen-Loeve transform is an identity, and in this case the filter design problem can be directly solved in the spatial domain.

List of symbols

$\mathbb{1}$	identity matrix
$\nabla_\theta f(\theta)$	Jacobian of a function $f(\theta)$
A	area of interest on the image plane in $[\mathrm{m}^2]$
A_0, A_1	events, subsets of a parameter space of a decision problem
A_L	surface of extended (area) light sources in $[\mathrm{m}^2]$
$\mathcal{A}$	set of indices of picture elements inside an area of interest A
$\mathcal{B}$	set of indices of nodes inside an area of interest A
C_{tot}	objective function for shape from shading
$C_{\mathrm{irr}}, C_{\mathrm{smt}},$	
$C_{\mathrm{reg}}, C_{\mathrm{edg}}$	terms of an objective function for shape from shading
C_{PS}	objective function for photometric stereo
$c(t)$	vector function, components are constraints on the shape from shading solution
$D(\cdot)$	distribution of micro-facets of a rough surface
$\mathcal{D}$	decision space
d	lens diameter in $[\mathrm{m}]$
d, d_i	decision, an element of $\mathcal{D}$
dO	elementary surface patch in $[\mathrm{m}^2]$
$d\omega$	solid angle subtended by the lens in $[\mathrm{sr}]$
E, E_{i}	irradiance, incident irradiance in $[\mathrm{W}\cdot\mathrm{m}^{-2}]$
E	operator of an expectation of a random variable
$F, F(n)$	Fresnel coefficient, Fresnel term
f	distance of the image plane from the optical centre in $[\mathrm{m}]$
f	column vector containing filter coefficients
$f(x)$	probability density function in the Monte Carlo method
$f(x, y)$	function specifying a surface shape in $[\mathrm{m}]$
G	geometrical attenuation (masking/shadowing) factor of the Torrance-Sparrow reflectance model
$\mathcal{G}$	set of possible scene descriptions
$g(x_1, x_2), g$	geometrical factor, operator of application of a geometrical factor in $[\mathrm{m}^{-2}]$
$g(x)$	sampled function in the Monte Carlo method

115

H_0	null hypothesis (a hypothesis) of a decision problem
H_1	alternative hypothesis (an alternative) of a decision problem
$h(\theta)$	image plane irradiance computed by the imaging model in $[\text{W·m}^{-2}]$
$\vec{h}$	unit length vector bisecting the angle between vectors $\vec{v}$ and $\vec{l}$
$I(x_1, x_2)$	two point transport intensity from x_2 to x_1 in $[\text{W·m}^{-4}]$
$\mathcal{I}$	set of indices of all picture elements on the image plane
J	Jacobian matrix
K_{ij}	form factor for the radiosity method
k_{dif}	coefficient of diffuse reflection in $[\text{sr}^{-1}]$
k_{spec}	coefficient of specular reflection in $[\text{sr}^{-1}]$
$L, L_{\text{r}}, L_{\text{e}}$	radiance, reflected radiance, emitted radiance in $[\text{W·m}^{-2}\text{·sr}^{-1}]$
L_{i}	incident radiance in $[\text{W·m}^{-2}\text{·sr}^{-1}]$
$L(w, d)$	loss function of a decision problem
$\vec{l}$	unit length vector in the direction to the light source
M	finite dimensional operator corresponding to $\mathcal{M}$
N	noise, a random vector
$\mathcal{N}(m, Q)$	normal distribution with the mean m and the covariance Q
n, n_1, n_2	refraction indices of media
$\vec{n}$	surface normal
n_{sp}	specular power, a coefficient of the Phong reflectance model
$P(A)$	probability of an event A
$\mathcal{P}$	set of patterns p forming a composite alternative hypothesis
p	pattern to be detected by the North filter, a column vector
p, q	gradient of a surface $z = f(x, y)$, $p = \partial f/\partial x$ and $q = \partial f/\partial y$
Q	covariance matrix of the random sample vector
$R(\vec{n}), R(p, q)$	reflectance map in $[\text{W·m}^{-2}]$
$R(\pi, d)$	risk, an expected loss by making a decision d
r	distance from the optical centre to a projected surface in $[\text{m}]$
r_{jk}	distance between surface patches in the radiosity method in $[\text{m}]$
S	matrix square root of the inverse of the covariance matrix Q
$\mathcal{S}$	scene, a set of scene surfaces in $[\text{m}^2]$
S_i	surface patches in the scene in $[\text{m}^2]$
T, T_2	threshold value for a likelihood ratio test
t	vector of parameters $t_{i,j}$ specifying a surface projected onto an area of interest
$t_{i,j}$	parameter specifying the distance of the surface projected onto the node (i, j)
u, v	image plane coordinates in $[\text{m}]$
V_k	expansion coefficients of the Karhunen-Loeve expansion
$\vec{v}$	unit length vector in a direction to a viewer
w	parameter of a decision problem
X	random variable or a random vector
$\mathcal{X}$	sample space of a decision problem

x_1, x_2, x_3	surface points, surface patches in the scene
x, y, z	cartesian system coordinates
Y	random vector
y	realization of the random vector Y, an observation
Z_i	position of a point light source
α, α_δ	probability of a type I error, probability of false alarm
β, β_δ	probability of a type II error, probability of miss
β	off-specular angle, an angle between vectors $\vec{n}$ and $\vec{h}$ in [rad]
β_h	off-specular angle for which the density of micro-facets drops to 1/2 of its maximal value in [rad]
γ	angle between vectors $\vec{v}$ and $\vec{h}$, and between $\vec{l}$ and $\vec{h}$ in [rad]
$\delta(x)$	decision procedure, decision rule
$\epsilon(x_1, x_2)$	emitted two point transport intensity in [W·m^{-2}]
Θ	parameter space, a feasible set of parameters
θ	vector of parameters
$\hat{\theta}_{ML}, \hat{\theta}$	maximum likelihood estimate of θ
θ_i	azimuth angle of an incident direction in [rad]
θ_r	azimuth angle of a reflected direction in [rad]
κ	surface albedo
Λ	diagonal matrix with correlation coefficients λ_k on a diagonal
λ	thereshold of a likelihood ratio
λ_k	correlation coefficients of the Karhunen-Loeve expansion coefficients V_k
$\mathcal{M}$	integral operator in the rendering equation
$\pi(w)$	probability of an event w
$\pi_0(w)$	a priori probability of an event w
ρ_{bd}	bidirectional reflectance, BRDF in [sr^{-1}]
ρ_{dif}	diffuse part of the bidirectional reflectance in [sr^{-1}]
ρ_{spec}	specular part of the bidirectional reflectance in [sr^{-1}]
$\rho(x_1, x_2, x_3)$	intensity of light scattered from x_3 to x_1 by x_2
Φ_i	radiant flux of a point light source in [W]
$\Phi(\cdot)$	cummulative distribution function for the normal distribution $\mathcal{N}(0,1)$
ϕ_i	zenith angle of an incident direction in [rad]
ϕ_r	zenith angle of a reflected direction in [rad]
φ	angle between a projecting ray and the optical axis in [rad]
Ψ	orthonormal matrix with the orthonormal base vectors ψ_k as its columns
ψ_k	vectors of the orthonormal base
Ω	hemisphere above a surface element in [sr]
Ω	parameter space of a decision problem
ω	solid angle in [sr]

Samenvatting

Er wordt een techniek voor het verifiëren van scène beschrijvingen gepresenteerd die gebruik maakt van een afbeeldingsmodel. Dit model beschrijft de scènebelichting en het beeldvormingsproces. Er wordt verondersteld dat de geverifieerde scènebeschrijving is verkregen met een computer vision systeem. Het verificatiemethode bevat een schatting van de onbekende scèneparameters en een beslissingsmethode gebaseerd op het oorspronkelijke beeld en het afbeeldingsmodel.

Het beeld van een scène dat geprojecteerd wordt op een beeldscherm is het resultaat van een wisselwerking van licht met de voorwerpen in de scène en een camera beeldvlak. Besproken worden oppervlakte-reflectiemodellen, een cameramodel en een drietal technieken voor de berekening van de globale verlichting, 'ray tracing', 'radiosity' en stochastische ray tracing. Als onderwerp voor de verificatie van een scènebeschrijving wordt een afbeeldingsmodel genomen dat bestaat uit stochastische ray-tracing en het Torrance-Sparrow oppervlaktereflectiemodel.

Bij het gebruik van een afbeeldingsmodel is kennis van oppervlaktereflectieparameters en parameters van de lichtbronnen onontbeerlijk. Deze parameters zijn ook de parameters van het afbeeldingsmodel. Voor de schatting van de onbekende parameters wordt een 'maximum-likelihood' techniek voorgesteld.

Voor de verificatie van scènebeschrijvingen wordt een beslissing methode voorgesteld die bestaat uit de aftrekking van het camera beeld van het synthetisch beeld (een beeld gemaakt met het afbeeldingsmodel), voorts een lineaire filtering van de aftrekking d.m.v. een North filter, en een drempeling van het gefiltreerde beeld. Het blijkt dat voor het bijzondere geval van een eenvoudige hypothese en een eenvoudig alternatief deze methode een optimale beslissingsmethode is in de Neyman-Pearson betekenis. De verificatie van de scènebeschrijving is een beslissingsprobleem met een eenvoudige hypothese en een samengesteld alternatief. Voor het bepalen van geschikte filter coëfficiënten in de bovengenoemde beslissingswijze wordt een methode gegeven.

Ten slotte worden er enkele methoden voor de verbetering van de scènebeschrijving gegeven, die gebruikt kunnen worden indien een discrepantie tussen de scènebeschrijving en de verworven beelden wordt geconstateerd. De technieken die bij het afbeeldingsproces worden gehanteerd worden besproken.

De toepassing van de techniek wordt in een aantal beproefde experimenten onderbouwd.

Curriculum vitae

Milan Hanajík was born on December 28, 1962 in Bratislava, Czechoslovakia.

In 1986 he obtained a M.Sc. degree in Electrical Engineering from the Slovak University of Technology (SUT) in Bratislava. He graduated with honours for being among the top two percent of the students.

From 1987 until 1991 he was an Assistant Professor at the Measurement Group of the SUT.

Since July 15, 1991, he was a doctoral student at the Measurement and Control Group of the Eindhoven University of Technology in the Netherlands.

He is a member of the Dutch Society for Pattern Recognition and Image Processing (Nederlandse vereniging voor patroonherkenning and beeldverwerking), a Dutch section of the IAPR.

Stellingen behorende bij het proefschrift:

SCENE VERIFICATION USING AN IMAGING MODEL
IN 3-D COMPUTER VISION

van

Milan Hanajík

Eindhoven, 7 december 1995

1. The absence of performance characterization in many computer vision algorithms published in the past suggests that there has been a sort of scientific deficiency in the computer vision society. The mere fact that a specific task can be done was important. Nowadays, this gap is being filled as it becomes increasingly necessary to quickly design computer vision systems which work efficiently, and which meet specified requirements.
   (Haralick, In D. Chetverikov and W. Kropatch, editors, Computer analysis of images and patterns, Lecture notes in computer science, Springer-Verlag, 1993, pp. 1–9)

2. As an approximation to Maxwell's equation for electromagnetics, the equation

$$I(x_1, x_2) = g(x_1, x_2) \left[\epsilon(x_1, x_2) + \int_S \rho(x_1, x_2, x_3) \, I(x_2, x_3) \, dx_3 \right]$$

   does not attempt to model all optical phenomena. It is essentially a geometrical optics approximation.
   (Kajiya, Computer Graphics, vol. 20, No. 4, 1986, pp. 143–150, and this thesis, chapter 2)

3. In stochastic ray tracing, the imaging model output is obtained as a result of a chance mechanism. The aim of using this model is to obtain the least random results possible.
   (This thesis, chapter 2, chapter 4)

4. It is unrealistic to expect that a computer vision system is able to interpret images which cannot be clearly interpreted by a trained human observer.

5. Presentation of results and advertising oneself, in short 'public relations', is a necessary and highly-valued, but time-demanding activity. For those who can not maintain a delicate balance between research and PR, an ultimate result might be a situation where the presentation and publication take all available time and there is no more time to do the work to be presented.

6. Every political system is the rule of an initiative-taking minority acting in favour of its own interests, over the majority of the people. Both these groups co-determine the political system: the ruling minority by implementing the system, and the majority by accepting it.

7. In contrast to the USA and the former communist countries, the democratic system and the market economy have brought a high living standard and a high degree of social security simultaneously to West European countries. The question is whether both can be preserved at the current level in the future.

8. It is difficult to learn the Dutch language. Therefore it is misleading to confuse the capability of coherent thought with the capability of coherent expression.