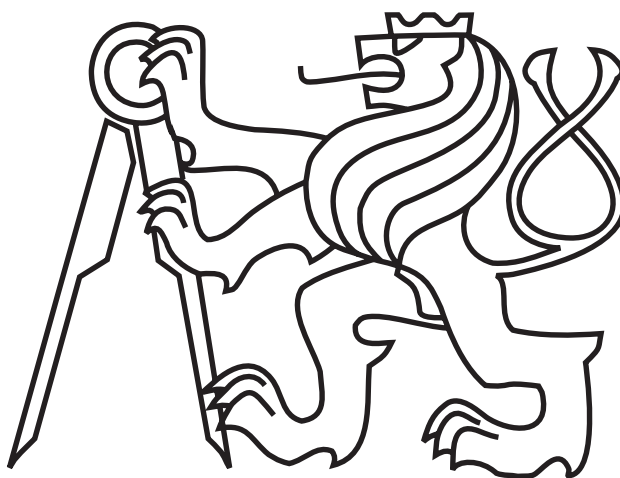


CZECH TECHNICAL UNIVERSITY IN PRAGUE
FACULTY OF ELECTRICAL ENGINEERING

MASTER'S THESIS



Josef Štrunc

Fusing Shape-from-Shading with Stereo

Department of Cybernetics

Supervisor: Ing. Tomáš Pajdla, Ph.D.

Prague, 2011

Czech Technical University in Prague
Faculty of Electrical Engineering

Department of Control Engineering

DIPLOMA THESIS ASSIGNMENT

Student: **Bc. Josef Štrunc**

Study programme: Cybernetics and Robotics
Specialisation: Systems and Control

Title of Diploma Thesis: **Fusing Shape-from-Shading with Stereo**

Guidelines:

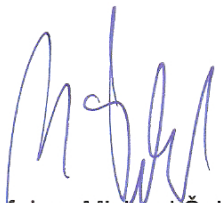
1. Review the literature about shape from shading, stereo vision and the state of the art in combining them.
2. Propose, design and implement a method for 3D reconstruction from stereo vision and shape from shading.
3. Validate the reconstruction method in experiments carried out in a controlled environment with known 3D shape. Demonstrate the method on real Mars data.

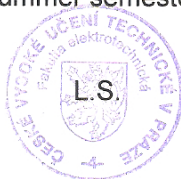
Bibliography/Sources:


- [1] R. Hartley, A. Zisserman. Multiple View Geometry in Computer Vision. Cambridge University Press, Cambridge, UK, 2003.
- [2] D. A. Forsyth, J. Ponce. Computer Vision: A Modern Approach. Prentice Hall. 2003.
- [3] M. Oren, S. K. Nayar. Generalization of Lambert's Reflectance Model. SIGGRAPH ". ACM. DOI 10.1145/192161.192213.
- [4] R. Oren, D. Barnes. A new shape from shading technique with application to Mars Express HRSC images. Aberystwyth University. 24 March 2010. Preprint.
- [5] C.X.Yan. Planet Photo-Topography Using Shading and Stereo. MSc Thesis. MIT 1993.
- [6] C. K. Chow, S.Y.Yuen. Recovering Shape by Shading and Stereo Under Lambertian Shading Model. IJCV (85):58-100, 2009.
- [7] L. Zhang, B. Curless, A. Hertzmann, S. M. Seitz. Shape and Motion under Varying Illumination: Unifying Structure from Motion, Photometric Stereo, and Multi-view Stereo. ICCV 2003.

Diploma Thesis Supervisor: Ing. Tomáš Pajdla, Ph.D.

Valid until summer semester 2011/2012


prof. Ing. Michael Šebek, DrSc.
Head of Department




prof. Ing. Boris Šimák, CSc.
Dean

Prague, April 8, 2011

Prohlášení

Prohlašuji, že jsem svou diplomovou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, SW, projekty atd.) uvedené v příloženém seznamu.

V Praze dne 31.5.2011...

Josef Štrunc
.....
podpis

Acknowledgements

First of all I would like to thank my supervisor Ing. Tomáš Pajdla, Ph.D. who devoted a lot of his time, effort and patience to help me with the thesis. Further my acknowledgements goes to RNDr. Michal Jančošek for helping me with stereo reconstruction and providing the 3D models and to Ing. Vladimír Smutný and Ing. Pavel Krsek, Ph.D., who helped me with the laboratory experiment. I would also like to thank my family for their constant support during my whole studies.

Abstrakt

Tato diplomová práce se zabývá problémem spojení techniky *shape-from-shading* (tvar ze stínování) s *multi-view stereo (MVS) rekonstrukcí* při použití Oren-Nayar modelu odrazivosti pro hrubé přírodní materiály. Byly navrženy dvě metody, které mohou být použity k vylepšení multi-view stereo rekonstrukce použitím nové foto-konzistenční metriky. Byly provedeny experimenty s obrázky z laboratoře i s obrázky povrchu Marsu, které ukazují, že navržená *plane-sweeping* metoda využívající informace o stínování, která je vhodná pro spojení s algoritmem MVS, dokáže správně určit polohu povrchu ve 3D scéně. Experimenty také ukázaly, že Oren-Nayar model odrazivosti je velmi přesný pro některé reálné materiály a může být úspěšně použit v *plane-sweeping* metodě k dosažení lepších výsledků než při použití Lambertovského modelu odrazivosti. Pokud jsou vlastnosti materiálu a pozice a směry zdroje světla a všech kamer dostatečně přesně známy, je možné dosáhnout přesnosti v řádu několika centimetrů při určování polohy povrchu ve 3D scéně.

Abstract

The thesis deals with incorporating the shape-from-shading technique into the multi-view stereo (MVS) reconstruction framework using the Oren-Nayar reflectance model for rough natural materials. Two methods for enhancing the MVS algorithm with new photo-consistency measure are proposed. Experiments with the laboratory images as well as with images of Mars's surface were conducted, proving that the proposed plane-sweeping method using shading information suitable for combining with MVS can find the correct position of surface in 3D scene. The experiments also showed, that the Oren-Nayar reflectance model is very accurate for some real-world materials and it can be successfully used in the plane-sweeping method to accomplish better results than the Lambert's reflectance model. With precisely estimated material parameters and the light source and camera directions, it is possible to achieve the accuracy of few centimeters in estimating the position of real surface in the scene.

Contents

1	Introduction	1
1.1	Related Works	2
1.2	Background	4
1.2.1	Epipolar Geometry	4
1.2.2	Stereo Reconstruction	5
1.2.3	Projective Spaces	6
1.2.4	Homography	8
1.2.5	Camera Projection Matrix	9
2	Multi-View Stereo Reconstruction	11
2.1	Components of Multi-View Stereo Algorithms	12
2.2	Stereo Ambiguity	14
2.3	Plane-Sweeping Based Reconstruction	15
3	Reflectance Model	19
3.1	Oren-Nayar Reflectance Model	20
3.2	Derivation of Oren-Nayar Reflectance Model	22
3.2.1	Model for Gaussian Slope-Area Distribution	23
3.2.2	Qualitative Model	24
3.3	Behaviour of Oren-Nayar Reflectance Model	25
4	Method	31
4.1	Modified Normalized Cross-Correlation	31
4.1.1	Rectification	33
4.1.2	Properties of the Modified Normalized Cross-Correlation	33
4.2	Plane-Sweeping Using Shading Information	36
5	Experiments	40
5.1	Verifying Oren-Nayar Reflectance Model	41
5.2	Cylinder Experiment	43
5.2.1	Fitting the Reflectance Model	45
5.2.2	Plane-Sweeping	47
5.3	Real Mars Data Experiment	50

5.3.1	Set A Simulations	50
5.3.2	Set B Simulations	53
6	Conclusion	60
6.1	Future Work	60
Appendix A: CD content		64

List of Tables

5.1	Summary of known parameters for the cylinder experiment.	47
5.2	Parameters and results of the plane-sweeping algorithm run #1 with the cylinder data.	49
5.3	Parameters and results of the plane-sweeping algorithm run #2 with the cylinder data.	49
5.4	Parameters and results of the plane-sweeping algorithm run #1 with the Mars image set <i>A</i>	57
5.5	Parameters and results of the plane-sweeping algorithm run #2 with the Mars image set <i>A</i>	57
5.6	Parameters and results of the plane-sweeping algorithm run #1 with the Mars image set <i>B</i>	59
1	Directory structure on the CD	64

List of Figures

1.1	Epipolar geometry.	5
1.2	Projective plane.	7
1.3	Pinhole camera model.	9
2.1	Illustration of the CMP's MVS reconstruction algorithm performance. . . .	18
3.1	Coordinate system	21
3.2	Facet foreshortening effect.	22
3.3	Magnified profile of rough surface	23
3.4	Reflectance models comparison	26
3.5	Reflectance models comparison	26
3.6	Model scene arrangement	27
3.7	Radiance reflected from cylinder	27
3.8	Effect of changing angle θ_i on Lambert's and Oren-Nayar model	28
3.9	Difference between Lambert's and Oren-Nayar model	29
3.10	Effect of changing angle θ_i on Oren-Nayar model	29
3.11	Effect of changing $\phi_r - \phi_i$	30
3.12	Effect of changing σ	30
4.1	Illustration of NCC in 3D space.	32
4.2	Illustration of NCC and the newly proposed normalized cross-correlation NCC_m measures on synthetic example with patches having inexpressive repetitive texture, but diverging in intensity by a constant.	35
4.3	Illustration of maximal and average values of scaled NCC and the newly proposed normalized cross-correlation NCC_m measures for the wall example.	36
4.4	Illustration of SFS plane-sweeping method.	38
5.1	Photo of Mars's surface	40
5.2	Experiment set-up for measuring the radiance functions of real-world materials.	41
5.3	Real-world material samples	42
5.4	Rough sand sample picture set	43
5.5	Reflectance functions of real-world surfaces	44
5.6	Oren-Nayar radiance fitted to rough sand sample radiance function	44

5.7	The cylinder model	45
5.8	The cylinder experiment.	47
5.9	Cylinder with projected facet corners.	48
5.10	Fitted Oren-Nayar and Lambertian radiances for cylinder surface.	48
5.11	Plane-sweeping algorithm run#1. The plot is in the same scale as the laboratory experiment.	50
5.12	Plane-sweeping - brightness difference function.	51
5.13	Plane-sweeping - brightness difference function, run#2.	51
5.14	Images and 3D reconstruction of Mars's surface.	52
5.15	Chosen sets of images sharing the same feature. The same patch is selected in the 3D point cloud and it is chosen as the fitting facet, projection of its corners into each image is marked with red dots. Under the images are details of the rectified facet projection.	53
5.16	Fitted Oren-Nayar and Lambertian radiances for Mars's surface set A . . .	54
5.17	Illustration of plane-sweeping with Mars image set A	55
5.18	Plane-sweeping result for Mars image set A , run#1	56
5.19	Plane-sweeping result for Mars image set A , run#2	56
5.20	Fitted Oren-Nayar and Lambertian radiances for Mars's surface set B . . .	58
5.21	Plane-sweeping result for Mars image set B , run#1	59

Symbols and Abbreviations

A^2	two-dimensional affine space
A^3	three-dimensional affine space
$\vec{b}_1, \vec{b}_2, \vec{b}_3$	basis vector of coordinate system
C	principal point of the camera image plane
E	irradiance
Φ	light flux
\mathbf{H}	homography matrix
L	radiance
\mathbf{M}	camera projection matrix
O	camera center
ω	solid angle
\mathbf{R}	camera rotation matrix
ρ_{BD}	bidirectional reflectance distribution function
ρ	albedo
σ	roughness of surface
\mathbf{t}	camera position vector
2D	two-dimensional
3D	three-dimensional
BRDF	bidirectional reflectance distribution function
CMP	Center for Machine Perception
CTU	Czech Technical University in Prague
MVS	multi-view stereo
NASA	National Aeronautics and Space Administration
NCC	normalized cross-correlation
SFS	shape-from-shading

Chapter 1

Introduction

For planetary rovers, the time needed for signal to reach a celestial body from the Earth is considerably high to teleoperate the rover. Therefore it is highly desirable to develop rover with high level of autonomy to lower the frequency of time consuming communication. For any mobile autonomous robot it is essential that it is able to create model of its unknown environment, process these data and make intelligent decisions, e.g. to which position navigate itself. The navigation task is demanding for accurate and high-quality environmental models, especially in expensive applications like planetary robotics. It is unacceptable for the rover to stuck on some rock or hole. For this, 3D models are necessary, which can be obtained from visual data. That's one of reasons why the European Framework Programme 7 - Space projects PProVisG (Planetary Robotics Vision Ground Processing) [5] and PProViScout (Planetary Robotics Vision Scout) focus on developing the comprehensive framework for planetary visual processing for future planetary rover missions.

This thesis ¹ concerns with the topic of incorporating shape-from-shading technique into stereo reconstruction technique. Both of them use different cues from the visual data to obtain 3D shape of the scene and fusing them together, using strong capabilities of each technique, promises more accurate and robust results. Stereo reconstruction is in most cases accurate method for reconstructing the 3D scene. There are situations, though, where it could be inaccurate or erroneous, such as on flat surfaces without (or with repetitive) texture, where the corresponding points in the scene cannot be assigned correctly. It is also inaccurate for distant points, as the angle between the left camera, scene point and the right camera is very small. In those cases the cues which are using the shape-from-shading methods can be helpful.

The work is organized in the following way. At first an overview of recent and relevant works which concern with fusion of different techniques for obtaining a 3D model of the scene is given in the Section 1.1 Related Works. Then in the Section 1.2 Background several basic concepts used later in the work are explained. Chapter 2

¹This work was supported by grant for SpaceMaster students from the Directorate of Human Spaceflight, ESA.

Multi-View Stereo Reconstruction copes in more detail with current state-of-the art multi-view stereo reconstruction algorithms and searches for the appropriate place in this technique, where the use of the shading information can be incorporated. Chapter 3 Reflectance Model introduces the Oren-Nayar reflectance model above all. Its behavior is investigated in detail and demonstrated on numerous plots. In Chapter 4 Method are proposed the main ideas how to use the shading information towards developing better multi-view photo-consistency measure. Chapter 5 Experiments describes verifying the Oren-Nayar reflectance model on several real-world materials and tests of the proposed plane-sweeping method on images taken in the lab and images of the Mars's surface.

The main contribution of this work is verifying that some real world materials obey Oren-Nayar reflectance model and that this fact can be used to develop new photo-consistency measure which can be easily incorporated into existing stereo reconstruction algorithm using the plane sweeping method.

1.1 Related Works

In computer vision there have been developed several techniques for obtaining 3D shape of the scene from images in last 40 years. Shape from shading introduced by Berthold K. Horn in 1975 [12], photometric stereo and binocular stereo are the basic techniques for obtaining 3D information from vision data. Other used methods are multi-view stereo [10], shape-from-motion/silhouettes/texture. Each of them using different clues to estimate the 3D shape of the scene. In recent works, researchers are trying to combine those basic techniques to get use of each method's strength or/and incorporate better models of image formation (realistic camera models, more accurate reflectance models). The goal is to achieve better robustness, accuracy, faster execution and ability to process larger amounts of data, e.g. high-resolution images or large data sets. In this section an overview of several recent works dealing with those problems is stated.

Yuen and Chow in [7] describe integration of shape-from-shading and stereo for Lambertian objects. Their method solves recovering depth at occlusion, matching at places with similar shading and matching at smooth silhouettes complementing the shape from stereo method by SFS. The SFS method they use, on the other hand, takes use of the prior knowledge of the boundary information provided by the sparse stereo algorithm in the form of depths of singular points. The method has the advantage that it is based on a model of the physics of image formation. Their SFS works with perspective projection and oblique lighting. They show, that oblique lighting can be regarded as orthographic after using transformation same as the rectification transformation frequently used in stereo. However, they assume Lambertian shading model and single light source in infinity. Their stereo algorithm generates only sparse matches, which ensures that only reliable, easy to identify points, are matched and overcomes the ambiguity problem of dense stereo algorithm.

They propose new depth matching method for stereo, which uses SFS to match a neighborhood centered on a candidate point to match depth directly instead of intensity. This property overcomes the foreshortening problem caused by distorted features from different view angles which are hard to match. The SFS uses the Fast Marching Method [22], which starts in the singular points (the local brightness maxima with known depth determined by sparse stereo) and propagates towards the edges of segments attached to other singular points recovering the depth in the singular point neighborhood. This is done for each input image and then the depths are compared instead of intensities to verify the match. They show in experimental results, that the rectification transformation and the SFS method work with different lighting directions, also with highly oblique angles, and that the combination of methods is able to accurately recover visible and occluded parts.

The work of Yan [23] solves a similar problem with proposed technique called Depth From Shading and Stereo. The difference here is in the focus on images obtained by planet orbiting probes. Those images show the same area of surface from different view-points, so that the stereo and SFS can be applied. Moreover, the images are taken under different illumination, because they are not captured in the same time and the position of the Sun changes. Therefore the cues about the 3D shape from varying illumination can also be used by photometric stereo method. *Z-only* algorithm for solving photo-topography closely coupling the solution of SFS and stereo in a variational approach by formulating a combined cost function is used. It is tested on realistic Mars images and can deal with noise, geometry error, calibration errors and reflectance errors. The limitation is, that it does not handle varying albedo or very weak shading, it has also problems when the light is coming from the similar direction as is the view direction.

Other approach attempting to fuse different shape recovering techniques is the work of Zhang *et al.* [24]. They define a framework relating optical flow, motion, lighting and albedo. The general form of optical flow (which is the trajectory of a scene point in an image sequence) under varying illumination is developed. They use the key observation that surface positions, normals, motion, and illumination are all coupled together and can be obtained by one minimization problem solution. Surface positions are constrained by optical flow trajectories (geometric cue) and surface normals are constrained by intensity variations along these trajectories (photometric cue). Their formulation of photometric constraint on point intensity subsumes as special cases the structure from motion, photometric stereo and multi-view stereo problems, which all use assumptions that some parameters are known and allow others to vary. The solution how to get all of the parameters together is shown and analyzed. Their algorithm iteratively estimates affine camera parameters, illumination, shape and albedo from a monocular image sequence. It computes dense shape that is accurate even in uniform untextured regions. However this approach is not robust to occlusions, shadows, inter-reflections and specularities.

O'Hara and Barnes present in [18] an optimization based SFS algorithm suited for reconstructions of planetary surfaces whose results overcome the stereo recon-

structions details. Their algorithm does not require any 3D data for initialization and provides flexibility to use various camera and reflectance models. It uses the steepest descent approach to iteratively minimize the cost function

$$f = \sum_{i,j} (I(i,j) - R(p(i,j), q(i,j), V(i,j)))^2, \quad (1.1)$$

where $R(p, q, V)$ is a reflectance function given by the reflectance model, $p(x, y) = \frac{\delta U}{\delta x}$, $q(x, y) = \frac{\delta U}{\delta y}$ are the height gradients of the surface in a discrete height grid $U(x, y)$, $V(i, j)$ are the unit viewing directions of every pixel (i, j) from the camera center given by the camera model and $I(i, j)$ is the image. In their implementation they use Lambert's and simplified Oren-Nayar (see Chapter 3) reflectance models. To achieve fine precision in affordable time they exploit two-stage multiresolution approach, which scales down the image resolution and generates low resolution depth map in the first stage and use it as initial surface for the second stage with high resolution image.

1.2 Background

1.2.1 Epipolar Geometry

Epipolar geometry is the basic structure used by stereo reconstruction algorithms to calculate the position of feature seen by different cameras in 3D space. Considering two projections p_1 and p_2 of a 3D point P into the image planes Π_1 and Π_2 of two cameras with optical centers O_1 and O_2 , all of the points P , p_1 , p_2 , O_1 and O_2 are laying in the *epipolar plane* as described in [9]. This is visualized in Figure 1.1. The *epipolar plane* is defined by the two intersecting lines O_1P and O_2P ; The *epipoles* e_1 and e_2 of the two cameras are defined by intersection of the line connecting camera centers O_1 and O_2 (*baseline*) with image planes Π_1 and Π_2 , respectively. The epipole of one camera is the projection of the optical center of another camera into the first camera's image plane and vice versa. The line l_1 connecting the epipole e_1 with the projection of point p_1 is called the *epipolar line* associated with the point p_2 . The *epipolar constraint* is the fact, that if p_1 and p_2 are projections of the same point P from 3D space, then p_1 must lie on the epipolar line associated with p_2 . This greatly constraints the search for correspondences done by most stereo algorithms.

Using the arrangement from the Figure 1.1, where the line segments a_1, b_1, c_1 and a_2, b_2, c_2 are known, the distances d_1 and d_2 of the point P from the camera centers O_1 and O_2 can be expressed using for instance the *law of sines* ($\frac{a}{\sin(\alpha)} = \frac{b}{\sin(\beta)} = \frac{c}{\sin(\gamma)}$) and the *law of cosines* ($a^2 = b^2 + c^2 - 2bc \cdot \cos(\alpha)$) as follows:

$$\cos(\alpha) = \frac{b_1^2 + c_1^2 - a_1^2}{2b_1c_1} \rightarrow \alpha$$

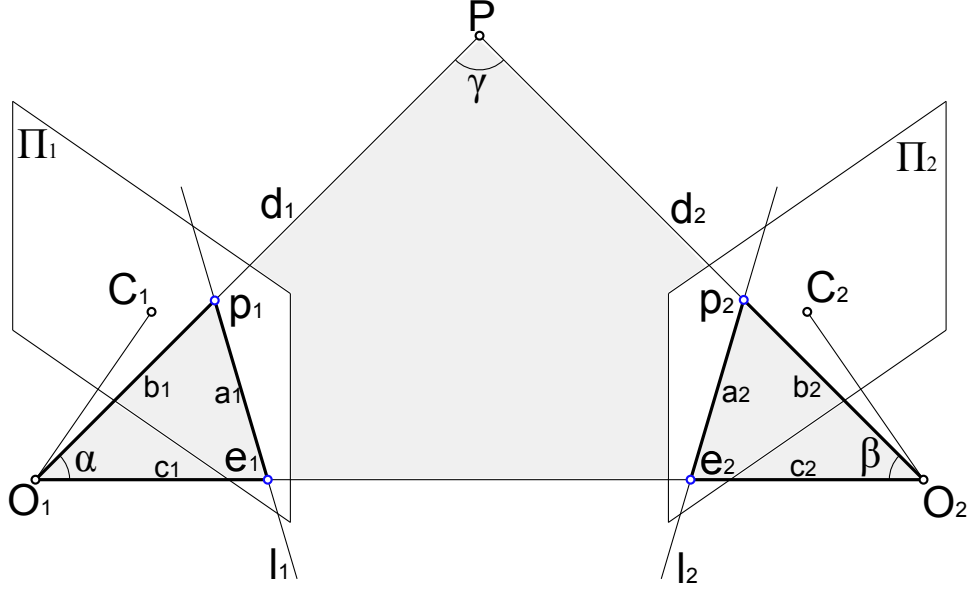


Figure 1.1: Epipolar geometry.

$$\begin{aligned}
 \cos(\beta) &= \frac{b_2^2 + c_2^2 - a_2^2}{2b_2c_2} \rightarrow \beta \\
 \gamma &= 180^\circ - \alpha - \beta \\
 d_1 &= \frac{\sin(\beta)}{\sin(\gamma)} \cdot |O_1 - O_2| \\
 d_2 &= \frac{\sin(\alpha)}{\sin(\gamma)} \cdot |O_1 - O_2|
 \end{aligned} \tag{1.2}$$

1.2.2 Stereo Reconstruction

The word *stereo* originally from greek means "solid" or "three-dimensional". In computer vision the stereo reconstruction denotes a class of techniques for creating the three-dimensional models of the environment. The goal of binocular or multi-view stereo is to reconstruct a complete 3D model from the set of images taken from known camera positions and directions using the epipolar constraint described in the previous section. The principle of this technique is also the main source of 3D information perceived visually by humans and animals which have eyes placed in front of the head (needed so that both eyes can see the same scene). Let's write now for simplicity only about binocular stereo which uses left and right camera (eye). It is based on the fact, that given two cameras with known parameters, positions and orientations and a visual feature located in both camera images, one can calculate the relative position of the object corresponding to the visual feature straightforwardly using epipolar geometry. The main problem here is finding the

corresponding feature in both images. Since the left image is taken from different position than the right one, it is slightly different than the right one. Moreover, to obtain a dense depth map there has to be found a lot of corresponding points in every image. Therefore only very small windows are compared for correspondence, so that the image features are still not distorted too much on this scale, and a lot of corresponding points can be found.

The two-dimensional normalized cross-correlation (NCC) function is used for this purpose with relatively small correlation windows (several pixels). For images with frequent change of brightness (with expressive texture) it works well, difficulties appear, when trying to find correspondence in low textured places of image. In areas with uniform color the NCC of two points equals to zero. Discrete version of 2D NCC function of image I ($k \times l$ pixels) with template T ($m \times m$ pixels) is described by following equation

$$NCC(x, y) = \frac{\sum_{i=0}^{k-m} \sum_{j=0}^{l-m} (I(x+i-\frac{n}{2}-1, y+j-\frac{m}{2}-1) - \text{mean}(I)) \cdot (T(i, j) - \text{mean}(T))}{|I - \text{mean}(I)| \cdot |T - \text{mean}(T)|} \quad (1.3)$$

It assigns values for each position of the template above the image in the range $\langle -1, 1 \rangle$. 1 is for perfect correspondence of intensity variations, -1 for completely contrary intensity variations. One property of NCC is, that it cannot express any difference for template and image which brightness differ by a constant value. That is it assigns value 1 to two images related by affine intensity transform

$$I = a \cdot T + b. \quad (1.4)$$

This is mostly desirable in stereo reconstruction, where the scene is viewed from different angles and the brightness of the same spot can change due to reflectivity function of given material (which is usually not perfectly Lambertian), but the shape of the image feature can remain more or less the same and therefore we want to confirm the correspondence.

1.2.3 Projective Spaces

In projective geometry the projective spaces are used instead of affine spaces because of certain advantages for modelling the perspective projection. Projective space is generalization of the affine space, it is able to distinguish perspective projections of all points in the space. The most important projective space we will use is the projective plane. It is given by a 2D affine plane and projection center not lying in this plane, which in our case correspond to the camera image plane and the camera center respectively. On the Figure 1.2 is the affine geometrical model of the projective plane. It is composed of an affine plane A^2 (subset of an affine space A^3) and point C not lying in A^2 . Each point x in A^2 can be represented by exactly one ray r going through the points x and C in A^3 . The rays passing through C and parallel to the plane A^2 does not intersect with A^2 and correspond to special set of

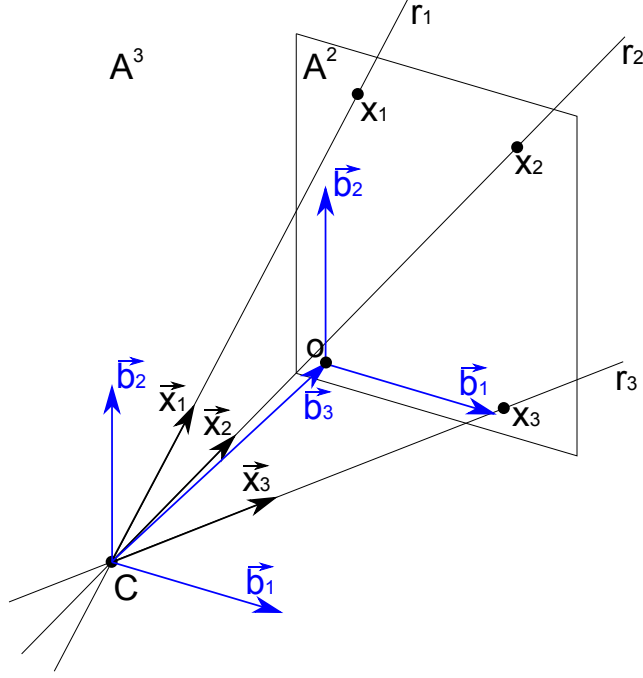


Figure 1.2: Projective plane.

points - the *ideal points*. The set of all rays in A^3 passing through C is called the projective plane and the rays are called points of the projective plane [20].

The points in the plane A^2 can be expressed using the coordinate system $(o, \vec{b}_1, \vec{b}_2)$ as $x = [x, y]^T$ and in the space A^3 using the coordinate system $(C, \vec{b}_1, \vec{b}_2, \vec{b}_3)$ as $x = [u, v, 1]^T$. The latter coordinates are called the *homogeneous coordinates* of a point. They describe the basic vector of one-dimensional subspace of \Re^3 (\vec{x} on the Figure 1.2), that means the ray r corresponding to point x , therefore they are not unique for one particular point and can be scaled by any number $\alpha \in \Re$, $x = \alpha \cdot [x, y, 1]^T$.

Using homogeneous coordinates $X = [x, y, z, 1]^T$ of a point in 3D in the coordinate system $(C, \vec{b}_1, \vec{b}_2, \vec{b}_3, \vec{b}_4)$ we can express the projection of this point into projective plane to coordinates $x = [u, v, 1]^T$ in coordinate system $(C, \vec{b}_1, \vec{b}_2, \vec{b}_3)$ as

$$\alpha \cdot \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{M} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}, \quad (1.5)$$

where $\mathbf{M} \in \Re^{3 \times 4}$ is the camera projection matrix.

1.2.4 Homography

Homography is a projective transformation from one projective space to another. In computer vision those spaces would usually be the image planes of models of projective cameras. Using homography, points with known coordinates in one image plane with known position and orientation in the 3D space can be projected into another image plane with different position and orientation, i.e. expressing the coordinates of the 2D points in the new image plane. This is useful when comparing the appearance of a scene from different view points. There applies, however, the assumption that the points in the scene lie in a single plane. Fulfilling the assumption and when the camera centers doesn't lie in the plane where the points are, it is possible to express point $x_\beta = [u, v, 1]^T$ in coordinate system β of one camera in relation to point $x'_{\beta'} = [u', v', 1]^T$ in coordinate system β' of another camera as

$$\lambda \cdot \begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \mathbf{H} \cdot \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}, \quad (1.6)$$

where $\mathbf{H} \in \mathbb{R}^{3 \times 3}$ is the homography matrix and $\lambda \in \mathbb{R} \setminus \{0\}$. This equation holds for all points lying in the same plane in 3D space.

Homography is used in the multi-view stereo algorithms for *rectification* of the images seen from different positions so they can be searched for correspondences. The homography matrix can be computed from coordinates of 4 points in one image corresponding to 4 points in the second image. In practice we need to assess those corresponding point coordinates by locating corresponding features in both images. Reshaping the homography matrix rows into one column in the following manner

$$\mathbf{H} = \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} \quad (1.7)$$

$$\mathbf{h} = \begin{bmatrix} h_1 & h_2 & h_3 \end{bmatrix}^T \quad (1.8)$$

we can express the dependency on two corresponding points (u, v) and (u', v') as

$$\begin{bmatrix} u & v & 1 & 0 & 0 & 0 & -u' \cdot u & -u' \cdot v & -u' \\ 0 & 0 & 0 & u & v & 1 & -v' \cdot u & -v' \cdot v & -v' \end{bmatrix} \cdot \mathbf{h} = 0 \quad (1.9)$$

Because the homography matrix can be scaled by a real multiple λ to obtain different matrix but still representing the same homography transformation, we are looking for one-dimensional subspaces of $\mathbb{R}^{3 \times 3}$ matrices of rank 3. Vector \mathbf{h} has 9 columns, therefore we need to get at least 8 rows into the matrix on the left-hand side in equation 1.9 by adding 4 correspondence points (so the solution will have one undetermined variable).

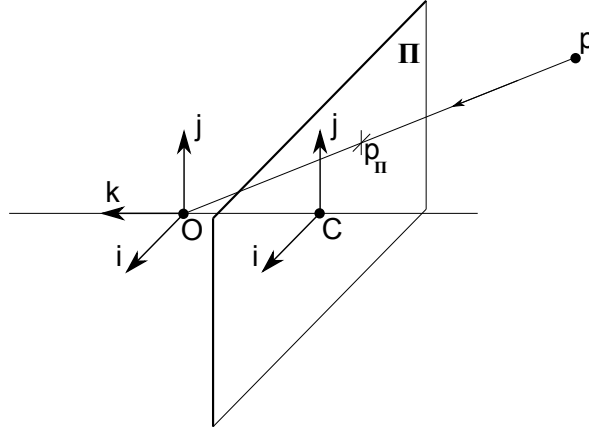


Figure 1.3: Perspective projection into virtual image plane Π with pinhole camera model.

1.2.5 Camera Projection Matrix

To describe the image formation process when taking a picture with physical camera, i.e. the projection of points from 3D space into the image, geometrical camera models are used. Using the camera model, the projection matrix can be found and the equation 1.5 can be used to project the 3D points into the image plane. For perspective projection a model of a pinhole camera is used, where only one light ray from every direction goes through the pinhole and projects inversly to the image plane behind the pinhole. For the model purposes the virtual image plane is considered which is located in a unit distance in front of the pinhole (in the object space) so the projected images are not upside down. Then the point p on position $[x, y, z]$ in 3D coordinate system $(O, \vec{i}, \vec{j}, \vec{k})$ defined as on the Figure 1.3 is projected into the point p_{Π} on position $[x', y']$ in the image plane with coordinate system (C, \vec{i}, \vec{j}) using following equations:

$$\begin{aligned} x' &= f \frac{x}{z} \\ y' &= f \frac{y}{z}, \end{aligned}$$

where f is the distance of the pinhole O from the image plane Π .

In reality cameras with lenses are used to gather more light from the scene, which mainly causes that the image is not sharply focused in the whole area of the image (depth of field), but also some other discrepancies from the ideal pinhole model. The physical retina is not in the unit distance form the optical center, its coordinate system doesn't have the origin in the point of intersection of optical axis with retina, it may be skewed, so the angle between the two image axes is not 90° , and the position is usually expressed in pixel units instead of meters in retina coordinate system. The projection matrix relates the world and camera coordinate

systems and contains all parameters to map it into the physical retina coordinates. The physical parameters can be divided into two groups - the *intrinsic parameters*, such as the focal length of the lens, the size of the pixels, the position of the principal point, and the *extrinsic parameters* such as the position and orientation of the camera. Incorporating those into the camera projection matrix yields [9]

$$\mathbf{M} = \begin{bmatrix} \alpha & -\alpha \cot \theta & u_0 \\ 0 & \frac{\beta}{\sin \theta} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix}, \quad (1.10)$$

where α and β are horizontal and vertical magnifications in pixel units, θ is the skewness angle of the physical retina, (u_0, v_0) is the position in pixel units of the point where the optical axis pierces the physical retina, \mathbf{R} is the camera rotation matrix and \mathbf{t} is the position of camera in the world coordinate system.

Chapter 2

Multi-View Stereo Reconstruction

Multi-view stereo reconstruction is a technique for obtaining 3D shape of scene from the set of images of the scene taken from different view-angles, and the associated camera projection matrices. Current state-of-the-art methods are very powerful and precise tools for reconstructing 3D model of the scene. They exploit the epipolar constraint described in Section 1.2.1. The more cameras see a particular scene point, the stronger this constraint is. That is why this approach is more accurate and robust than binocular stereo, where each scene point can be seen at most from 2 cameras (if not occluded).

Since the Middlebury multi-view data sets with laser scanned ground truth models were introduced together with Seitz's *et al.* article [21] and direct quantitative comparison of different methods got easier, the development of algorithms have accelerated and they are quickly increasing their performance. Evaluation of the main multi-view reconstruction algorithms is available online on accompanying web page [2]. The best methods are approaching the precision of laser-based reconstructions [11]. However, this applies only for small scenes with Lambertian objects and precisely controlled imaging conditions. There are situations where the input images are not precise enough (moving objects, moving camera, unfavourable light conditions), number of images or camera positions are deficient or the scene itself is problematic, containing for example shiny objects causing specular reflections or unsufficiently textured objects, where the correspondence points are ambiguous. Problematic are also large-scale outdoor scenes mainly due to the difficulties with visibility and the need to process large amounts of data. In those cases there still is need for improvement. Lot of works trying to solve those problems have appeared recently, e.g. the method for reconstruction of large-scale scenes [11] which is used as a base for the method described in section 2.3.

2.1 Components of Multi-View Stereo Algorithms

Existing algorithms differ in several attributes or constituents which are: the *scene representation*, *photoconsistency measure*, *visibility model*, *shape prior*, *reconstruction algorithm*, and *initialization requirements* as stated in [21].

Scene Representation

Scene representation is the way of describing the 3D geometry. Popular representation is regularly sampled 3D grid, which divides the space into voxels (volumetric picture elements) which state if the piece of volume on certain position is occupied by an object. It can also be defined as a 2D function giving for every pixel on image plane the distance to the closest surface. 3D grids are popular for their simplicity and ability to approximate any surface, but they are not suited for scaling up to higher 3D resolutions because the computational demands of algorithms using them grow cubically. Therefore they cannot be used for reconstruction of large scenes. Another frequently used representation is the polygon mesh, which represents surface as a set of adjacent planar facets. Polygon meshes are well-suited for visibility computations and they are efficient to store and render. Some methods does not describe the scene centrally with one 3D model, but use depth map given for every input image separately instead.

Photo-Consistency Measure

Photo-consistency measure is the part of algorithm evaluating how much the (temporal) scene model corresponds to the input images. There are two categories - *scene space* and *image space* measures. The first one takes a point or patch in 3D, projects it into the input images and calculate the agreement between projection and image using eg. sum of squared differences or normalized cross-correlation. Image space measures use the estimated scene geometry to map one input image on it and predict image appearance from the viewpoint of another input image. The predicted and input images are compared which results in the prediction error. The main difference between those two types of measure is the domain of integration. Scene space measure error is integrated over a surface and therefore prefers smaller surfaces, but prediction error is integrated over the set of input images and therefore penalize the parts with large area or which appear in lot of images.

In stereo algorithms there is mostly assumed, that the brightness of a surface is view-independent - it has Lambertian reflectance properties. Lately some photo-consistency measures appear that use more accurate reflectance models. In this work new photo-consistency measure is proposed, using the Oren-Nayar reflectance model for rough surfaces. This measure uses clues from shading of the surface assuming certain properties of surface material, like albedo and roughness. It needs the knowledge about position of the light source in addition, but using together with some

standard stereo photo-consistency measure can improve the results of reconstruction. The condition of known light source should be fulfilled in environments of expected use of this method. For instance on Mars the only light source is the Sun and its position on the sky can be determined in any time given the position on Mars's surface. Development of this measure and a way of incorporating it into existing stereo algorithm, fusion of stereo and shape-from-shading in a way, will be described in following chapters.

Visibility Model

Visibility models specify which views to consider when evaluating photo-consistency, they should be utilized by every algorithm because visibility of certain points often change with the viewpoint. *Geometric visibility techniques* try to model the image formation from the estimate of 3D geometry of the scene to determine which points are visible in which images. Simplification can be made by constraining the allowable viewpoints distribution which make the algorithms more efficient. *Quasi-geometric techniques* use heuristics about the geometry to solve visibility. For example limiting the photo-consistency analysis to groups of nearby cameras minimize the effect of occlusions. Third type of solving visibility is to treat occlusions as outliers and then use outlier rejection techniques.

Shape Prior

Photo-consistency measures alone are not sufficient to recover precise geometry in some cases, particularly in low-textured scene regions. Shape priors are assumptions about certain properties of the scene surface. They are necessary for binocular stereo, which is less constrained than the multi-view stereo. Some algorithms assume small overall surface areas or smooth surfaces, some build the scene from reference shapes such as a sphere or a plane. Some algorithms trying to find the largest photo-consistent scene reconstruction ("photo hull" [16]; it is the shape which encapsulates all possible photo-consistent shapes) prefer maximal surfaces instead, they do well in reconstructing high curvature or thin structures, but tend to bulge out in low textured regions.

Reconstruction Algorithm

Multi-view stereo reconstruction algorithms can be divided into four categories. The first calculate a cost function on 3D volume at the beginning, and then extract surface from this volume. Example is the voxel coloring algorithm. A lot of algorithms work with volumetric Markov random fields and use max-flow or multi-way graph cut to obtain the optimal surface. Second class is iteratively evolving the surface represented by voxels, level sets or meshes to minimize a cost function. Methods based on space carving start from a large initial volume and proceed by

progressively removing inconsistent voxels. Level-set methods in distinction minimize a set of partial differential equations defined on a volume and they can also locally expand the evolved volume. In third class are image-space methods that compute a set of depth maps for every image and ensure consistency between them or merge the depth maps into single 3D model as a post process. The last group of algorithms extract and match feature points and fit a surface to the matched features.

Initialization Requirements

Beside the set of input images multi-view stereo algorithms usually require some input information about the geometric extent or constraints of the scene. Some need only a rough bounding box, some require segmentation on foreground and background for each input image. Image-space algorithms typically use constraints on the allowable range of depth, that is specifying if part of scene lies near the camera or in background.

2.2 Stereo Ambiguity

The fundamental problem in stereo reconstruction is the *correspondence problem* - assuming Lambertian radiance function (surface radiance is constant over all outgoing directions), there may be some distinct points in the scene which have the same radiance and therefore their projection into every image have the same brightness. Even when the radiance is not Lambertian, there may be pixels in the image having the same brightness and not corresponding to each other in the scene. Moreover there can be pixels in input images with different brightness which do correspond to the same 3D scene point. It is therefore hard to match points from one image with points from another image which correspond to the same scene point.

More generally this is investigated in [6], where the maximal theoretical amount of information that can be obtained from input images using any passive 3D vision algorithm wheter it is stereo reconstruction, depth-from-defocus, shape-from-shading or shape-from-sillhouette is studied theoretically. They call the complete information that can be obtained from input images the *light-field*. It is in fact the radiance as a function of 4 variables, where two of them are coordinates of a point on a surface in the scene and two are the polar angles of outgoing light ray. They prove that stereo ambiguities for Lambertian scenes can be caused only by constant intensity regions in any input image or uncomplete light-field, i.e. insufficient number of camera views. The first reason is inherent and there is nothing that could resolve the related ambiguity. The points with the same intensity (or color) match all with each other and correct correspondence cannot be identified. The shape cannot be recovered using visual measurements alone, prior information must be used in addition . The second is only problem in practice, when not enough camera views

can be provided, and could be removed by investing more effort or money to acquire more views.

Let's mention some practical consequence for some reconstruction algorithms. Most of the state-of-the-art multi-view reconstruction methods use a 3D point cloud in one of stages, which is further processed in various ways. The 3D point cloud can be obtained using stereo based, growing based or plane-sweeping [8] based methods. For some situations or surfaces those methods doesn't produce dense support in the 3D point cloud. It is the case mainly for surfaces without expressive texture, eg. surface areas with constant or slowly varying intensity (color), where no strong correspondences between points can be found. This is a consequence of the facts stated in previous two paragraphs. The 3D point cloud is sparse also for transparent or highly reflective surfaces or when using photos as input images, where certain areas are not focused, which is in a way similar to the textureless surfaces. Those surfaces cannot be reconstructed using standard stereo matching techniques based on photo-consistency measures and they will be usually missing or they will produce incorrect surfaces in the 3D model. However, there are some recent works dealing with those problems, for instance [15] which copes with the weakly supported surfaces in the 3D point cloud.

2.3 Plane-Sweeping Based Reconstruction

The SFS photo-consistency measure defined further in section 4.2 was developed in order to be easily incorporated into existing MVS reconstruction algorithm developed at CTU in the Center for Machine Perception (CMP) [13, 14, 15]. The result of the Dino data set reconstruction taken from the Middlebury comparison web-page [2] created with CMP's algorithm in the year 2009 compared with the ground truth model and the Dragon dataset reconstruction is on the Figure 2.1 to illustrate the performance of this algorithm. The accuracy of the Dino data set reconstruction is 95.9% according to the measure on the Middlebury comparison web page. The algorithm is also capable of accurate reconstruction of large-scale scenes from large sets (hundreds of pictures) of hi-resolution (3072×2048) images, without the prior knowledge about the camera matrices. It is continuously under development and the abilities to cope with weakly-supported surfaces, hallucinated surfaces and other capabilities were added recently. In this section its basic parts are described in order to see where the possible improvement investigated in this work can fit in.

The CMP's stereo reconstruction algorithm processing pipeline is based on the method of Keriven and Pons [11], a state-of-the-art multi-view stereo method for reconstructing large-scale environments. At first the feasible camera pairs are computed based on the *epipolar geometry* 1.2.1. Then features are matched in the feasible camera pairs and seeds (a 3D point with a set of cameras it was triangulated from) are created. For each camera the minimal and maximal depth based on the related seeds is determined and the dense 3D point cloud is generated using the

plane-sweeping method originally proposed in [8].

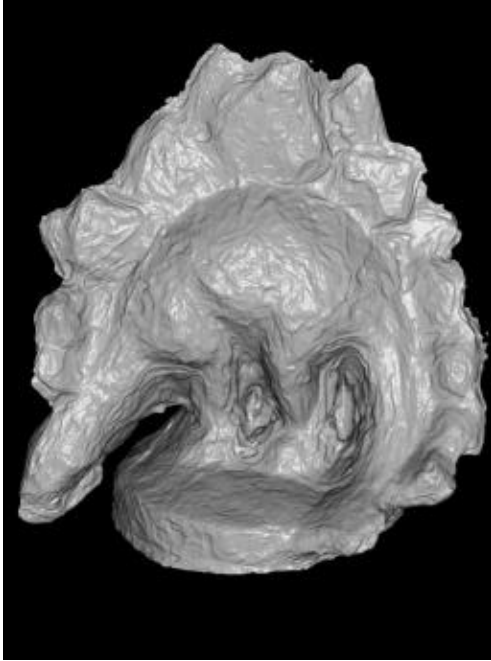
The plane-sweeping method is based on an assumption that the feature in the image is created by a set of points laying in single plane in 3D. Limiting the size of features to be small enough, this assumption is close to the truth. Then the position of the real plane in the scene is searched by shifting some hypothetical plane in 3D space and looking how would the reprojection of one image into other images look like if the scene bit generating them had the hypothesized position. The discrepancy of the reprojected and input images is evaluated and the position of the hypothesized plane which generated the reprojection with the lowest discrepancy is estimated to be the position of real surface in the 3D scene. For this all the camera matrices have to be known.

Plane-sweeping used in CMP’s algorithm can be concisely described as follows. One camera is taken and marked as reference r . Then a plane parallel to the image plane of r is hypothesized and it is shifted in the direction of its normal. For each pixel p and distance d of the hypothesized plane from the image plane of r , the photo-consistency $f(p, d)$ is computed. α nearest feasible cameras to r are chosen and marked as target t successively. A window of size 5×5 pixels centered in the pixel p is reprojected from the image plane of r into the image plane of t using homography transformation (taking assumption that all points from both r and t lay in 3D space in the hypothesized plane) and $c(p, d, r, t) = \text{NCC}$ of those two windows is computed. The homography matrix can be evaluated because the projection matrices of cameras are known. This is made for every pixel p , distance d , reference camera r and target camera t . The photo-consistency $f(p, d)$ is set to the maximum of $c(p, d, r, t)$ over all target cameras t and fixed r for every pixel and every distance of hypothesized plane. The reconstructed depth of the pixel p is chosen as the depth d for which $f(p, d)$ is maximal and the position in 3D space of corresponding pixels is esimated. This method produces a lot of true 3D points, but also a lot of false 3D points. A filtering is done on resulting 3D point cloud such that for each 3D point its small neighborhood is searched for other 3D points and if there is more than β points from more than β different cameras, the point is accepted. Other points are considered as unknown.

After obtaining the 3D point cloud a minimum s-t cut based optimization is made, which outputs a visibility consistent mesh close to the final reconstruction. The mesh is finally refined using a variational optimization. It is assumed that for every point in the cloud the list of cameras that can see it is available so further specialized improvements can be done (eg. removing halucinated surfaces). Describing this parts of algorithm is out of scope of this work and details can be found for example in [11].

The weak point of this method is in using the NCC in the plane-sweeping part when evaluating the amount of resemblance between one input image and reprojection of another input image. That is one of the reasons for large numbers of generated false 3D points in the point cloud. NCC does not take into account the affine transform of the correlation windows as described in section 1.2.2. This affine

transform can contain informations about shading which can be used to refine the correlation function. Particularly, the shading is more informative, when the scene is non-Lambertian and when the view-points are very different (but still observing the same part of scene). This is the point where some SFS technique can be used. Two concrete methods are described further in Chapter 4.



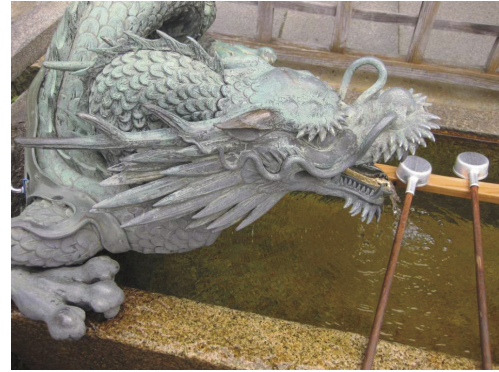
(a) Rendered 3D model of the Middlebury Dino data set, reconstructed by 2009 version of the algorithm.



(b) Ground truth laser-scanned model.



(c) Rendered textured 3D model of the Dragon data set made by the latest version of the algorithm [15].



(d) One of the input images from the Dragon data set.

Figure 2.1: Illustration of the CMP's MVS reconstruction algorithm performance.

Chapter 3

Reflectance Model

The behaviour of light being reflected from the surface of objects can be described by the *reflectance model*. There are two main types of light reflectance,

- surface reflectance caused by reflection on the interface of two surfaces with different refractive indices and
- body reflectance caused by subsurface scattering.

Surface reflectance is usually modeled as ideal specular reflectance for flat mirror-like surfaces or specular reflectance with wider directional lobes for rougher and matt surfaces. Subsurface scattering was in the field of computer graphics and computer vision most often modeled as *Lambertian reflectance*. Lambertian matt surface is perfectly diffusive and appears equally bright from all view directions. The overall amount of reflected light can be described by single parameter called *albedo* which completely characterises a Lambertian surface. However this is just an approximation which is valid for only few real-world materials, like matt paper or smooth wall plaster. There is a lot more materials in the real-world for which the Lambertian reflectance model is incorrect, for example shiny materials like glass and polished metals or rough materials like cloth, sand and concrete. Surface of those materials changes its brightness while the observer changes its view-angle.

Computer graphics is one area, where the reflectance models are used for creation of realistic images and videos. Its applications ranges from computer rendered films or video games to medical visualizations, CAD programs or advertisement graphics. The quality of rendered image depends on the accuracy of used reflectance model. Another need of reflectance models is in understanding the process of image creation studied in the field of computer vision. Here the approach is reversed to the one in computer graphics, where object's parameters, material, position and viewer position and direction are given and the image appearance is in demand. In computer vision, researchers are trying to obtain parameters of real-world objects from visual data, for example image from digital camera, microscope or optical telescope. This is obviously a hard problem because of the large number of scene parameters having

influence on its visual appearance, e.g. material, distance and orientation of objects, number, type, position and orientation of light sources or used camera parameters. Other complicating factors are some complex light behaviour effects like specular glow, luminence, interreflections and shadows. It would be impossible to get all unknown information only from images, some parameters of the scene usually have to be given beforehand or assumed. For example, when we assume only one light source, we know its direction wrt. the viewer and we assume certain type of surface material, we can use appropriate reflectance model for this material and information about brightness from the image to obtain surface orientation or distance. This is done by shape-from-shading and photometric stereo algorithms. Therefore it is important to have accurate reflectance model. As will be seen in Chapter 5, the real-world materials are often strongly non-Lambertian and Lambert's reflectance model is inaccurate for obtaining information from images. Therefore the main motivation of investigation in following sections is understanding and testing of appropriate reflectance model which can be used for rough natural surfaces occurring among others in planetary exploration.

3.1 Oren-Nayar Reflectance Model

The Oren-Nayar reflectance model is a body reflectance model for diffuse reflection from rough surfaces introduced by M. Oren and S. K. Nayar in 1994 [19]. It is successfully used to predict the appearance of natural surfaces with macroscopic roughness such as concrete, sand, plaster, etc. It describes the radiance in given direction going from surface element dA into solid angle $d\omega$ in dependence on:

- polar angles θ_i, ϕ_i of incident light ray,
- polar angles θ_r, ϕ_r of reflected light ray,
- variance σ^2 for Gaussian distribution of elementary surface facets with different slopes (higher σ means rougher surface) and
- albedo ρ of elementary Lambertian surface facets

Radiance in direction θ_r, ϕ_r caused by flux Φ incoming from direction θ_i, ϕ_i to area dA in general is given by:

$$L_r(\theta_r, \phi_r; \theta_i, \phi_i) = \frac{d\Phi_r(\theta_r, \phi_r; \theta_i, \phi_i)}{dA \cdot \cos(\theta_r) \cdot d\omega_r} [W \cdot m^{-2} \cdot sr^{-1}] \quad (3.1)$$

The θ and ϕ angles are polar angles related to the normal \hat{n} of surface dA and are defined as on the Figure 3.1. The term $dA \cos(\theta_r)$ in radiance definition represents foreshortened area dA projected into direction of reflected ray. Φ_r is reflected light

flux (power intensity in wats $[W]$) bounded to incident flux Φ_i by bidirectional reflectance distribution function (BRDF) ρ_{BD} by equation

$$\Phi_r(\theta_r, \phi_r) = \int_{\Omega} \rho_{BD}(\theta_r, \phi_r; \theta_i, \phi_i) \cdot \Phi_i(\theta_i, \phi_i) \cos(\theta_r) d\omega_r, \quad (3.2)$$

where Ω is the outcoming solid angle into which is the flux Φ_r going. Radiance is defined so that the corresponding light flux is propagating through area perpendicular to the flux's direction. It is appropriate for measuring the distribution of light anywhere in space.

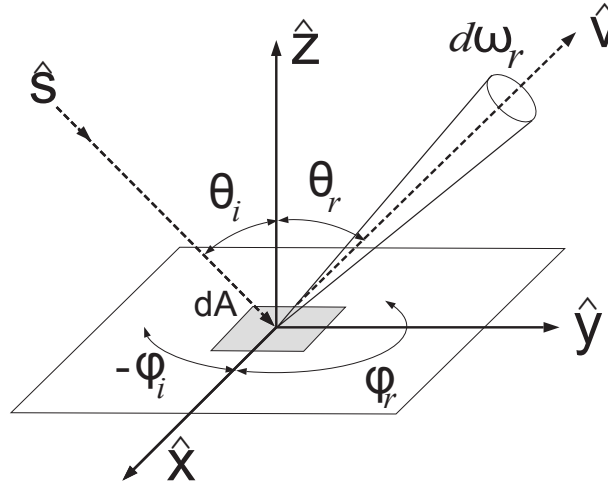


Figure 3.1: Coordinate system. \hat{S} is the light source direction vector, \hat{V} is the viewer direction vector, θ and ϕ are the corresponding polar angles and $\hat{X}, \hat{Y}, \hat{Z}$ are the coordinate system basis vectors. Taken from [19].

The advantage of radiance definition is obvious when we calculate power incoming to some patch with area dA which is tilted to the light flux direction. It is clear, that it will absorb less power than if it was perpendicular, we have to project area dA to the direction of light flux, so the projected area is $dA \cdot \cos(\theta)$. We just take the radiance in certain point of space coming from certain direction and multiply it with solid angle subtracted by the light source from the surface patch's view and with foreshortened surface patch area $dA \cdot \cos(\theta)$ and we have the power absorbed by the patch. When we integrate the radiance L_r over certain surface and over all outgoing directions, we get the power this surface is radiating. In our case we handle light sources and other surfaces separately. We suppose that all radiated power from surface is given only by reflection of power incoming from sources.

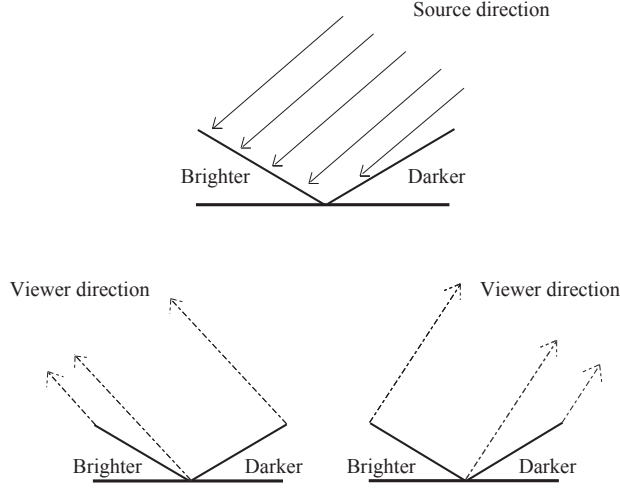


Figure 3.2: Facet foreshortening effect. The overall brightness vary when looking at V-cavity from different angles. Picture taken from [19].

3.2 Derivation of Oren-Nayar Reflectance Model

The rough surface in Oren-Nayar model is assumed to be composed of a large number of symmetric V-cavities which are formed by two small facets with Lambertian surface inclined one to another. Further it is assumed that the size of V-cavities is much smaller than the area of observed surface, i.e. a large amount of V-cavities is projected into one pixel of the camera, and much bigger than the wavelength of incident light. This surface structure originates more complex way of reflecting light. The basic effect is foreshortening of facets when watching from different angles, as illustrated on Figure 3.2. It can be easily visualized, that when looking at one single V-cavity from different view angles, one can see various fractions of illuminated and not illuminated facet and therefore the overall brightness of surface can vary. Small disparities on the rough surface cause shadowing when the light source angle θ_i is large and masking when the view angle θ_r is large. These phenomena together with interreflection between opposite facets makes the reflectivity function obviously non-Lambertian. On Figure 3.3 is depicted magnified profile of such a surface, where surface normal \hat{n} , facet normal \hat{a} , source vector \hat{s} , view vector \hat{v} and corresponding angles are noted.

Derivation of the Oren-Nayar model begins with defining the projected radiance $L_{rp}(\theta_a, \phi_a)$ of single Lambertian facet into the direction $\hat{a} = (\theta_a, \phi_a)$ of the surface plane normal \hat{n} :

$$L_{rp}(\theta_a, \phi_a) = \frac{\rho}{\pi} E_0 \frac{\langle \hat{s}, \hat{a} \rangle \langle \hat{v}, \hat{a} \rangle}{\langle \hat{a}, \hat{n} \rangle \langle \hat{v}, \hat{n} \rangle}, \quad (3.3)$$

where \hat{s} is the direction of source, \hat{v} is the direction of viewer, \hat{a} is the direction of the facet normal, ρ is albedo of the Lambertian facet, E_0 is the irradiance when

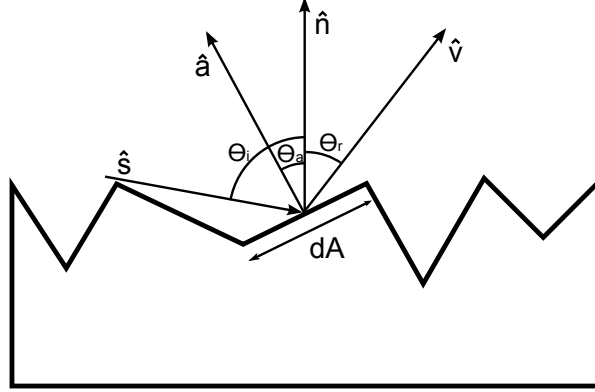


Figure 3.3: Magnified profile of rough surface modeled by symmetric V-cavities which are composed of 2 similar facets with area dA .

$\hat{s} = \hat{a}$ and $\langle a, b \rangle$ is dot product of vectors. If a, b are unit vectors and θ the angle between them, $\langle a, b \rangle = |a| \cdot |b| \cdot \cos(\theta)$.

The total radiance of the surface can then be obtained as the average of $L_{rp}(\theta_a, \phi_a)$ by integrating over all possible facet normals \hat{a} :

$$L_r^1(\theta_r, \phi_r; \theta_i, \phi_i) = \frac{1}{2\pi} \int_{\theta_a=0}^{\frac{\pi}{2}} \int_{\phi_a=0}^{2\pi} P(\theta_a, \phi_a) L_{rp}(\theta_a, \phi_a) \sin(\theta_a) d\phi_a d\theta_a, \quad (3.4)$$

where $P(\theta_a, \phi_a)$ is the slope-area distribution of facets, it gives the fraction of total surface area which is occupied by facets with normal $\hat{a} = (\theta_a, \phi_a)$ and $\sin(\theta_a) d\phi_a d\theta_a = d\omega$, which means integrating over all elemental solid angles in directions of facet normals.

3.2.1 Model for Gaussian Slope-Area Distribution

The slope-area distribution $P(\theta_a, \phi_a)$ can be given by uniform distribution of facet orientations ϕ_a (isotropic surface) and Gaussian distribution of facet slopes θ_a with mean $\mu = 0$ and standard deviation of facet normals σ^2 . Bigger facet normal angle is bigger slope of facets composing the V-cavity and therefore bigger roughness. Then the factors expressing the shadowing and masking are included and reflectance of rough surface due to direct illumination evaluated by functional approximation of integral 3.4 is

$$L_r^1(\theta_r, \theta_i, \phi_r - \phi_i; \sigma) = \frac{\rho}{\pi} E_0 \cos \theta_i \left[C_1(\sigma) + \cos(\phi_r - \phi_i) C_2(\alpha; \beta; \phi_r - \phi_i; \sigma) \tan \beta + (1 - |\cos(\phi_r - \phi_i)|) C_3(\alpha; \beta; \sigma) \tan \left(\frac{\alpha + \beta}{2} \right) \right], \quad (3.5)$$

where the coefficients are:

$$C_1 = 1 - 0.5 \frac{\sigma^2}{\sigma^2 + 0.33}$$

$$C_2 = \begin{cases} 0.45 \frac{\sigma^2}{\sigma^2 + 0.09} \sin \alpha & \text{if } \cos(\phi_r - \phi_i) \geq 0 \\ 0.45 \frac{\sigma^2}{\sigma^2 + 0.09} \left(\sin \alpha - \left(\frac{2\beta}{\pi} \right)^3 \right) & \text{otherwise} \end{cases}$$

$$C_3 = 0.125 \left(\frac{\sigma^2}{\sigma^2 + 0.09} \right) \left(\frac{4\alpha\beta}{\pi^2} \right)^2 \quad \alpha = \text{Max}[\theta_r, \theta_i] \quad \beta = \text{Min}[\theta_r, \theta_i]$$

In this functional approximation all angular values should be given in radians except for σ^2 which is in degrees. The radiance is expressed in dependence on $\phi_r - \phi_i$ only, because the relative angle between incident ray plane and reflected ray plane perpendicular to the surface plane matters for isotropic surfaces. There is also derived the interreflection component of the radiance, ie. part of radiance which is due to the light incoming from the neighboring facet, not from the light source in infinity. Its approximation for surface with roughness σ is

$$L_r^2(\theta_r, \theta_i, \phi_r - \phi_i; \sigma) = 0.17 \frac{\rho^2}{\pi} E_0 \cos(\theta_i) \frac{\sigma^2}{\sigma^2 + 0.13} \left[1 - \cos(\phi_r - \phi_i) \left(\frac{2\beta}{\pi} \right)^2 \right]. \quad (3.6)$$

Total surface radiance is combination of direct illumination component and interreflection component

$$L_r(\theta_r, \theta_i, \phi_r - \phi_i; \sigma) = L_r^1(\theta_r, \theta_i, \phi_r - \phi_i; \sigma) + L_r^2(\theta_r, \theta_i, \phi_r - \phi_i; \sigma). \quad (3.7)$$

The model reduces to the Lambertian model when $\sigma = 0^\circ$. By taking albedo as a function of the wavelength $\rho(\lambda)$, the dependency on the wavelength comes out explicitly.

3.2.2 Qualitative Model

The relative significance of various terms in equation (4) was determined through numerous simulations by Oren and Nayar. The coefficient C_3 makes small contribution to the total radiance. The following qualitative model is obtained by discarding C_3 and interreflection component

$$L_r(\theta_r, \theta_i, \phi_r - \phi_i; \sigma) = \frac{\rho}{\pi} E_0 \cos \theta_i \left(A + B \text{Max} \left[0, \cos(\phi_r - \phi_i) \right] \sin \alpha \tan \beta \right). \quad (3.8)$$

$$A = 1.0 - 0.5 \frac{\sigma^2}{\sigma^2 + 0.33}$$

$$B = 0.45 \frac{\sigma^2}{\sigma^2 + 0.09}$$

3.3 Behaviour of Oren-Nayar Reflectance Model

Figures 3.4 and 3.5 compare radiances as functions of the view angle θ_r given by Lambertian model, Oren-Nayar accurate functional approximation and Oren-Nayar less accurate qualitative approximation, both compared with the functions from original Oren's and Nayar's article. In Figure 3.4 the incident ray is in the same plane as all reflected rays, in Figure 3.5 the incident ray is shifted by 45° to this plane. It can be seen on the first of these figures, that the qualitative model was implemented identically and the curve coincides to the one from original paper, only between implemented functional approximation and numerical evaluation taken from the original paper are minor differences. In the later figure qualitative models coincide again, but there is larger disparity between the functional approximation and the numerical evaluation, especially near $|\theta_r| = 70^\circ$ and larger. The arrangement for obtaining this radiance is on the Figure 3.6 and includes:

- elementary facet with rough surface positioned in the center of global coordinate system with normal $\hat{n}(\theta, \phi) = (0, 0)$,
- point light source in infinity with direction $\hat{s}(\theta_i, \phi_i) = (75^\circ, 0^\circ)$
- view directions $\theta_r = \langle -90^\circ, 90^\circ \rangle$ in the same plane with source vector, $\phi_i - \phi_r = 0$.

Near to the light source direction the Oren-Nayar model radiance approaches its maximum. With moving the view direction away from the source direction the radiance drops off, mainly because of the foreshortening and masking phenomena. In distinction, the Lambertian radiance is constant over all viewer directions and it differs from Oren-Nayar radiance significantly. There is also obvious difference between accurate functional approximation and less accurate qualitative approximation, the later one is constant for viewer directions from -90° to 0° , opposite side to the light source. On Figure 3.7 is plotted radiance function (corresponding to brightness) of a cut of look on a rough cylinder which is illuminated from the same direction as is the view direction. As moving to the edge of cylinder, the Lambertian radiance is decreasing much more than Oren-Nayar radiance which makes the cylinder look more flat for Oren-Nayar model.

On Figure 3.8 is shown the effect of changing the light source direction θ_i on the radiance function for both Lambert and Oren-Nayar model (accurate functional approximation). With increasing θ_i , the radiance is generally decreasing, because the surface patch with area dA receives smaller portion of light (projected area $dA \cos(\theta_i)$ is smaller). Also it can be seen, that for light direction perpendicular to the surface patch $\theta_i = 0^\circ$, Oren-Nayar model radiance is flat¹, but lower than the Lambertian. As θ_i is increasing, the shape of Oren-Nayar model radiance differs more from the Lambertian. This graph also shows the explanation for the brightness values of

¹for the accurate functional approximation; For numerical evaluation shown in Oren and Nayar's article [19] it is close to flat, but slightly decreasing with larger θ_i .

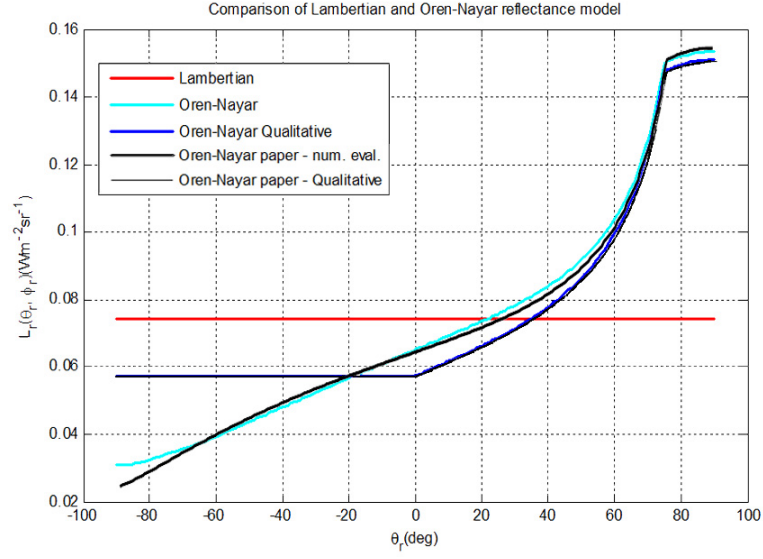


Figure 3.4: Radiance as a function of view direction - comparison of Lamberts and Oren-Nayar (accurate functional aproximation) reflectance models. Black lines are taken from the original Oren and Nayar's paper for comparison. $\phi_i - \phi_r = 0^\circ, \sigma = 30^\circ$ (for Oren-Nayar), $\rho = 0.9, \Phi_i = 1W$

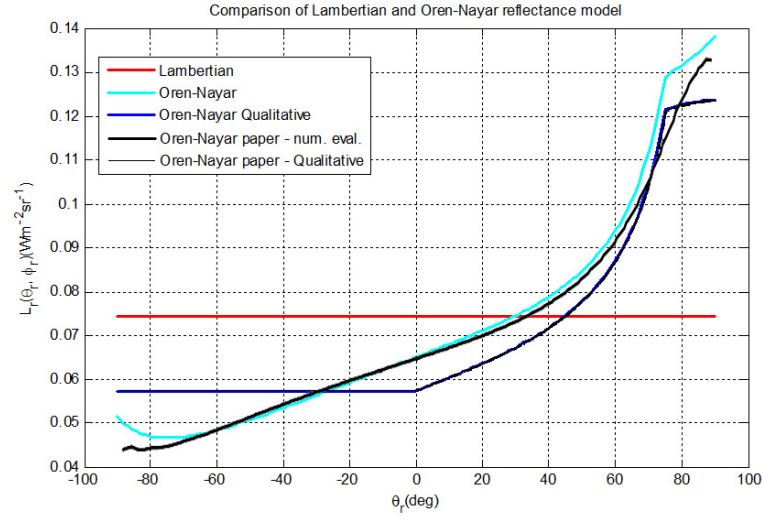


Figure 3.5: Radiance as a function of view direction - comparison of Lamberts and Oren-Nayar (accurate functional aproximation) reflectance models. . Black lines are taken from the original Oren and Nayar's paper for comparison. $\phi_i - \phi_r = 45^\circ, \sigma = 30^\circ$ (for Oren-Nayar), $\rho = 0.9, \Phi_i = 1W$

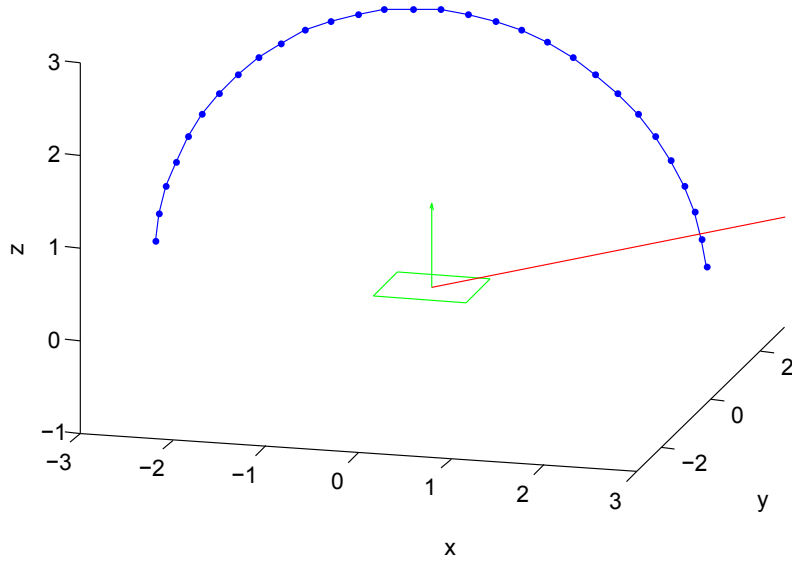


Figure 3.6: Model scene arrangement for obtaining radiance function on the Figures 3.4 and 3.5. Green lines mark the test surface patch and its normal vector, red line is the light source direction and blue half-circle are positions of viewer, view angles are in the range $\langle -90^\circ, 90^\circ \rangle$ wrt. the patch's center. Light source direction vector and viewers direction vectors are in the same plane.

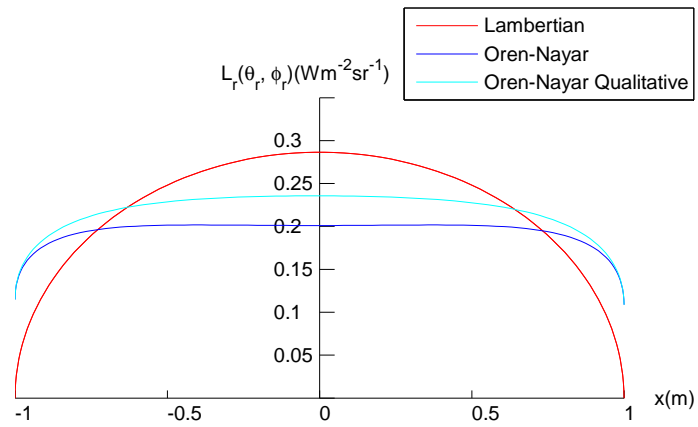


Figure 3.7: Radiance reflected from cylinder obtained using Oren-Nayar and Lambertian models. The Oren-Nayar model makes the cylinder look more flat.

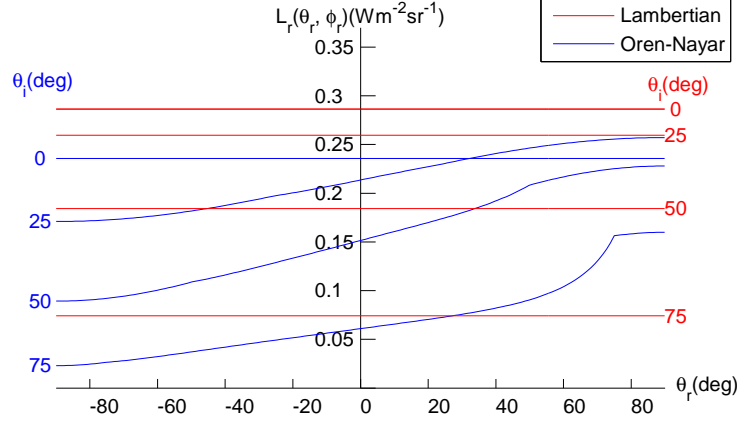


Figure 3.8: Effect of changing light source angle θ_i , comparison of Lambertian and Oren-Nayar model. $\phi_i - \phi_r = 0^\circ$, $\sigma = 40^\circ$, $\rho = 0.9$, $\Phi_i = 1W$

cylinder from Figure 3.7. For equal light source and camera angles θ_i and θ_r the Oren-Nayar radiance is approaching its maximum, especially for larger angles. Compared to Lambertian radiance, Oren-Nayar radiance is bigger for larger angle values, it means it is bigger in regions of the cylinder which are close to edges, where the θ_i and θ_r angles are large. This partially compensates decline of brightness due to the surface foreshortening in the near-edge regions and if the cylinder was rendered using Oren-Nayar model, it would look more flat in comparison with cylinder rendered using Lambert's model, which can be also seen from the Figure 3.7. On Figure 3.9 is plotted the difference of Oren-Nayar and Lambertian radiances for several different light source angles.

On following three Figures 3.10, 3.11 and 3.12 is plotted changing shape for Oren-Nayar radiance when changing light source direction θ_i and ϕ_i and surface roughness σ , respectively, while keeping other parameters constant. On the last one of these figures there is shown that Oren-Nayar model reduces to Lambertian for $\sigma = 0^\circ$ (Lambertian radiance coincide with the line for $\sigma = 0^\circ$).

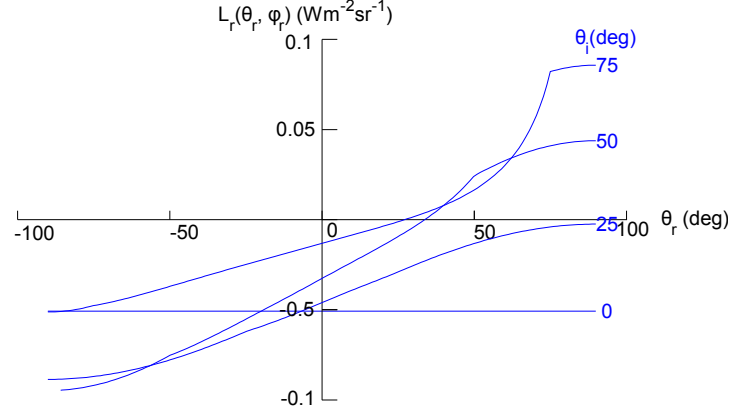


Figure 3.9: Difference between Lambertian and Oren-Nayar model for several different light source angles θ_i . $\phi_i - \phi_r = 0^\circ$, $\sigma = 40^\circ$, $\rho = 0.9$, $\Phi_i = 1W$

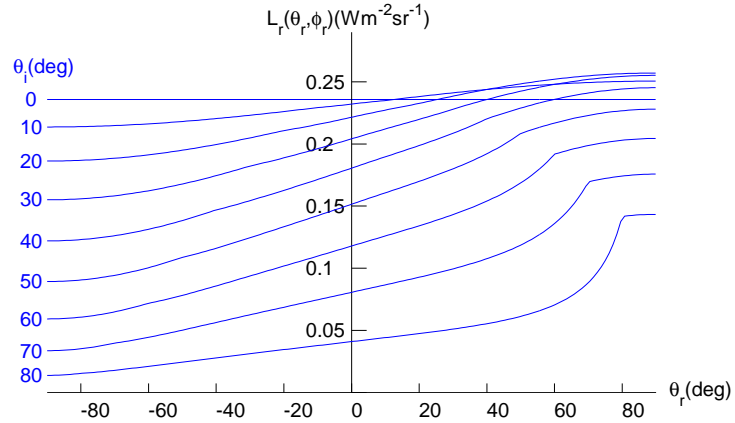


Figure 3.10: Effect of changing light source angle θ_i on Oren-Nayar radiance function. $\phi_i - \phi_r = 0^\circ$, $\sigma = 40^\circ$, $\rho = 0.9$, $\Phi_i = 1W$

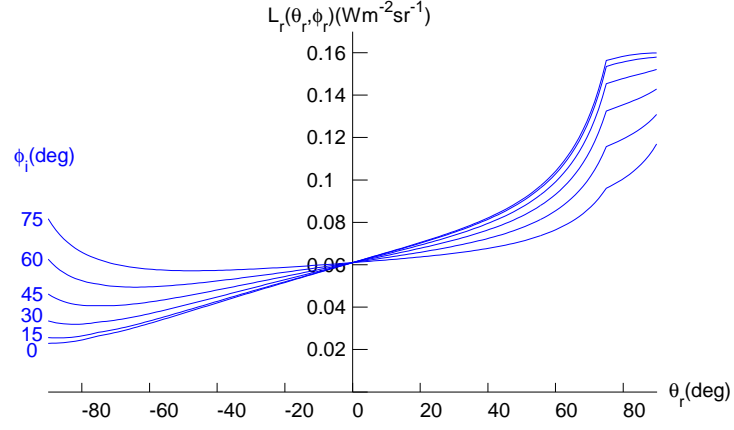


Figure 3.11: Effect of changing $\phi_r - \phi_i$ on Oren-Nayar radiance function. $\theta_i = 75^\circ, \sigma = 40^\circ, \rho = 0.9, \Phi_i = 1W$

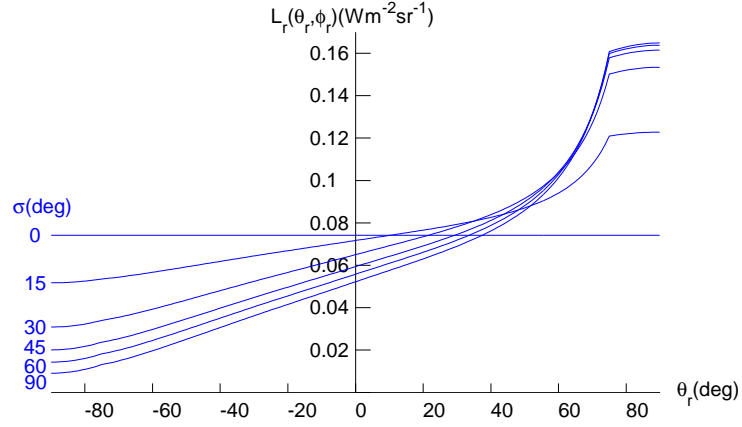


Figure 3.12: Effect of changing σ on Oren-Nayar radiance function. $\phi_i - \phi_r = 0^\circ, \rho = 0.9, \Phi_i = 1W$

Chapter 4

Method

In this chapter two methods exploiting the shading information and Oren-Nayar reflectance model towards better MVS consistency measure are proposed. This leads to the fusion of the stereo reconstruction and shape-from-shading. This chapter reassumes on the Section 2.3, where one part of the CMP's MVS algorithm - the plane sweeping method and its weak-points is described. Following two methods can substitute or complement the NCC photo-consistency measure to enhance the feature matching and to generate 3D point cloud with less false 3D points.

4.1 Modified Normalized Cross-Correlation

In MVS algorithms the normalized cross-correlation function (Equation 1.3) is used to find matching correspondences in images and to generate 3D points afterwards. However, there are situations, where, using the NCC, a lot of false correspondences is found. Some of those can be disregarded using information about the brightness difference or shading. As mentioned in Section 1.2, NCC, as it is defined, ignores constant brightness differences between two images. That means, it has the same value for two images I_1 and I_2 and also for two images I_1 and $a \cdot I_2 + b$, where every pixel of the second image is changed by an affine transform. In this section a modified normalized cross-correlation function denoted as NCC_m is proposed, which conserves some usefull properties of NCC, but which can distinguish between the two cases stated above. The not normalized cross-correlation function cannot be taken in consideration, its properties are completely unsuitable for stereo matching purposes, because it gives high values where the overlapping images have high intensities, not where their intensities vary in the same manner.

It is usefull to transfer the image vectors into 3D space for visualization. We can imagine the image with n pixels as a n -dimensional vector of pixel intensities. The correspondent representation in 3D space would be a 3D vector, though no image has only 3 pixels in practice, it is suitable for picturing the situation. Now, having the template \vec{X} and a fraction of image \vec{Y} under the template, their means

are subtracted to obtain $\vec{x} = \vec{X} - \text{mean}(\vec{X})$ and $\vec{y} = \vec{Y} - \text{mean}(\vec{Y})$, and they can be visualized in 3D like on the Figure 4.1. What the NCC does is not calculating $\vec{x} - \vec{y}$, the actual difference between vectors \vec{x} and \vec{y} , but at first it transforms \vec{x} into \vec{z} with an affine transformation $\vec{z} = a \cdot \vec{x} + b \cdot \vec{1}$, so that it is as close to \vec{y} as possible. The vector \vec{z} is a linear combination of unitary vector multiplied by some constant \vec{b} and vector \vec{x} multiplied by some constant a and therefore it lies in the green plane in the Figure 4.1. This represents the disregarding of the constant difference in intensity. Now the difference of vectors $\vec{y} - \vec{z}$ is smaller than original difference $\vec{y} - \vec{x}$.

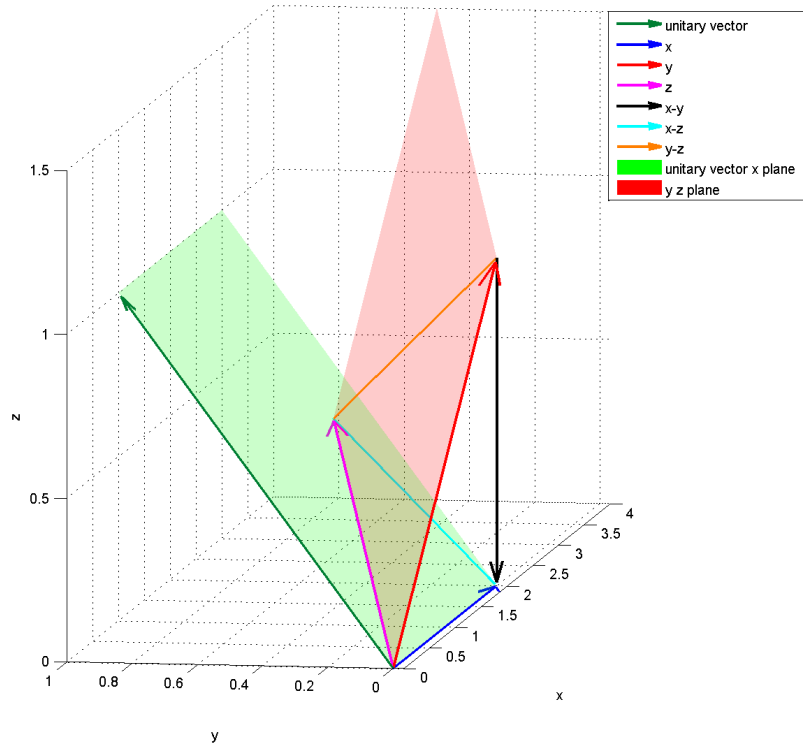


Figure 4.1: Illustration of NCC in 3D space.

The newly proposed correlation measure in the 3D analogy is defined as follows

$$\text{NCC}_m(\vec{X}, \vec{Y}) = 1 - \left(\frac{|\vec{x} - \vec{z}|}{c} + \frac{|\vec{y} - \vec{z}|}{c} \right) / 2, \quad (4.1)$$

where $\vec{z} = a \cdot \vec{x} + b \cdot \vec{1}$, $x = \vec{X} - \text{mean}(\vec{X})$, $y = \vec{Y} - \text{mean}(\vec{Y})$, $a, b, c \in \mathbb{R}$, and $\vec{x}, \vec{y}, \vec{z} \in \mathbb{R}^3$. The variable c is the normalization constant representing maximal possible value of $|\vec{x} - \vec{y}|$, then the expression for NCC_m is in the interval $\langle 0, 1 \rangle$, the

value 1 represents complete correspondence. It directly transfers from 3D space into the image space, when the vectors $\vec{x}, \vec{y}, \vec{z} \in N^{m \times n}$ represents discrete 2D images.

4.1.1 Rectification

Before two images obtained by two different cameras can be directly compared using NCC or NCC_m they has to be transformed as they were captured from the same camera position and with the same camera direction. This transformation is called rectification. It is used in stereo matching algorithms because of one extra beneficial attribute. Considering only binocular stereo vision, the matching features in two images are searched along the epipolar lines, which is given by the epipolar constraint (see Section 1.2.1). After rectification, the epipolar lines change into horizontal lines in both images, therefore a patch from one image can be only shifted in horizontal direction in the other image to search for match. Here the rectification is used only in order to be able to directly compare images taken from distinct view-points with NCC and NCC_m . The pseudocode of procedure of rectification of images from the set of images I and camera matrices C is described in the Algorithm 1. The function *xyToHomography* uses two corresponding sets of 4 points and Equation 1.9 to calculate the homography matrix.

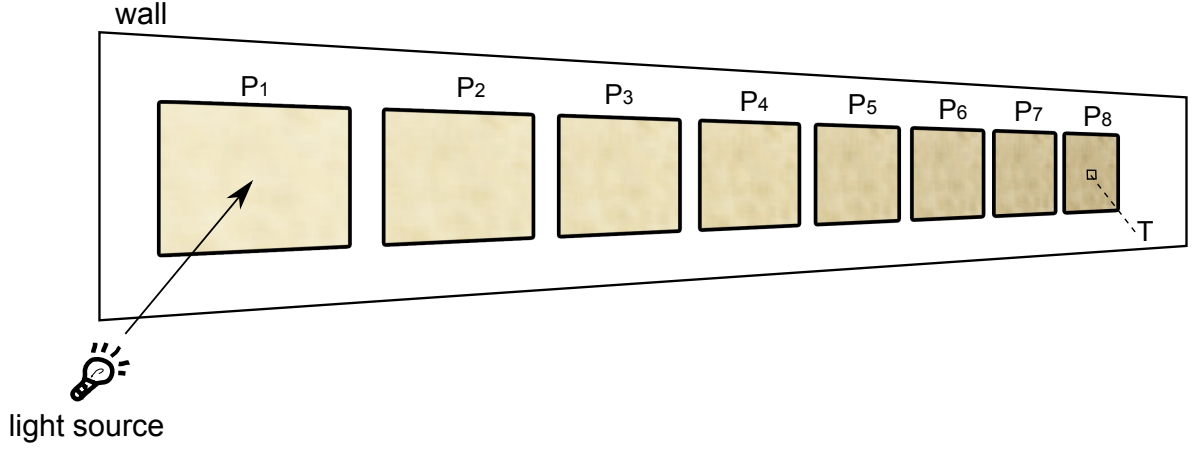
4.1.2 Properties of the Modified Normalized Cross-Correlation

Useful properties of NCC_m are demonstrated on following synthetic example. Lets have a scene with flat wall which has a noninformative repetitive texture and which is lit from the left side, so it is brighter on the left than the right side. The wall is observed by two cameras from different view-points, therefore its projection into the image planes is differently distorted because of perspective projection. The view from one camera can look like on the Figure 4.2(a). Although the emphasized patches are of the same size in the 3D scene, they have different size and shape in the image. They are of course differently distorted in the other camera image. In order to use NCC_m , the images are rectified using the Algorithm 1. After this transformation, all patches look the same from the view of both cameras, e.g. like on the Figure 4.2(b) (although here is chosen the view direction perpendicular to the wall as the common view of both cameras and without perspective projection, and not the view of one of the cameras). Now, when searching for the match of the template T for instance, it is moved all over the wall and in every position the normalized correlation is computed. For the purpose of this synthetic example, this process can be imagined as splitted into computing NCC for each patch separately, this is depicted on the Figure 4.2(c). Because the texture is unexpressive and repetitive, the NCC looks more or less the same for every patch, there is a lot of places on the wall with high correlation value, even on the other side of the wall, where the overall brightness is obviously higher. On the contrary, using NCC_m for calculating the correlation for each patch with the template T , which is shown on the Figure 4.2(d), there

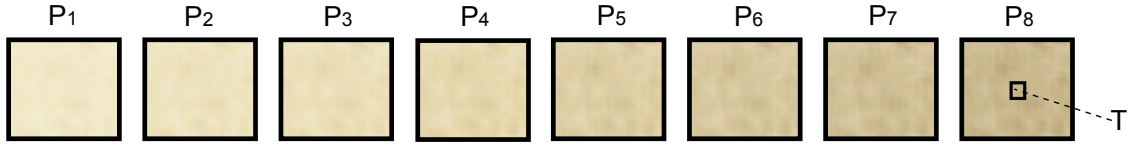
Algorithm 1: Rectification transformation

```
1 begin
  input : set  $I$  of  $n$  images of size  $(k \times l)$ , set  $C$  of  $n$  camera projection matrices
  output: set  $R$  of  $n$  rectified images
2 for  $i = 1$  to  $n$  do
3   |  $corrPoints[i] = 4$  user defined corresponding points from  $I[i]$ ;
4 end
5  $ref =$  user defined index of reference image in  $I$ ;
6  $cpRefHom = \text{AffineToHomogeneous}(corrPoints[ref])$ ;
  /* compute homography matrices  $H$  for every image in  $I$  from 4
    corresponding points. */
7 for  $i = 1$  to  $n$  do
8   |  $cpHom = \text{AffineToHomogeneous}(corrPoints[i])$ ;
9   |  $H[i] = \text{xyToHomography}(cpRefHom, cpHom)$ ;
10 end
11  $coorRef = 2 \times (k \cdot l)$  array of coordinates of the reference image;
12  $coorRefHom = \text{AffineToHomogeneous}(coorRef)$ ;
13 for  $i = 1$  to  $n$  do
14   |  $coorRectified = \text{HomogeneousToAffine}(H[i] \cdot coorRefHom)$ ;
15   | for  $j = 1$  to  $(k \times l)$  do
16     | if  $\text{Round}(coorRectified)$  lies inside  $I[i]$  then
17       |  $R[i].image(coorRef[j]) = I[i].image(\text{Round}(coorRectified))$ ;
18     | end
19     | else  $R[i].image(coorRef[j]) = 0$ ;
20   | end
21 end
22 end
```

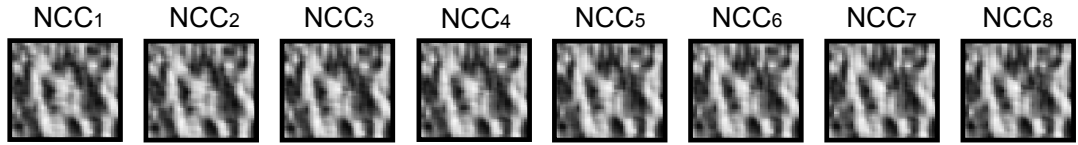
is difference in its overall value, it is increasing when the template is approaching the patch P_8 . The average and maximal value of NCC_m and NCC scaled into the interval $\langle 0, 1 \rangle$ for every patch P_i is shown on Figures 4.3(a) and 4.3(b), respectively, which illustrates this trend more accurately. Both, average and maximal value of NCC_m are increasing (almost) monotonically as approaching the patch P_8 , but the behavior of the same values off standard NCC is not that clear. This reflects the ability to incorporate also the brightness intensity differences. There is still a lot of places on the wall with high correlation values, but only near to the actual value of the template. The conclusion is, that using this correlation measure, a lot of false matches can be disregarded.



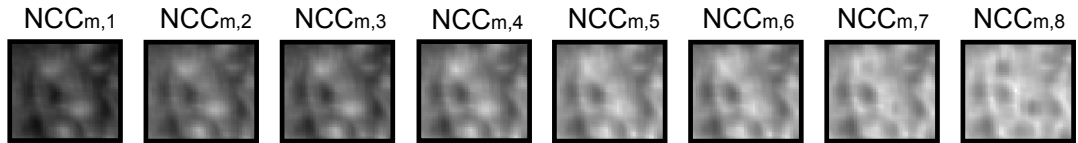
(a) Wall having surface with inexpressive repetitive texture viewed with perspective projection. Close to its left side is the light source causing the left part of the wall look brighter than the right part. P_i are patches of the wall texture cutted out in distinct places. T is the template with side length $\frac{1}{10}$ of the side length of each patch P_i cutted out from the patch P_8 .



(b) Patches P_i rectified to the same proportions using homography transformation (see the text) with template T cutted out of the patch P_8



(c) NCC of each patch P_i and template T respectively, plotted in gray color scale (-1 is black, 1 is white). There is a lot of correlating pieces with T in every patch P_i and the overall correlation confidence does not change much over patches P_i



(d) NCC_m of each patch P_i and template T respectively, plotted in gray color scale (0 is black, 1 is white). There is also a lot of correlating pieces with T within every patch P_i , but the overall correlation confidence changes over patches when approaching P_8 (from which is the template T cutted out).

Figure 4.2: Illustration of NCC and the newly proposed normalized cross-correlation NCC_m measures on synthetic example with patches having inexpressive repetitive texture, but diverging in intensity by a constant.

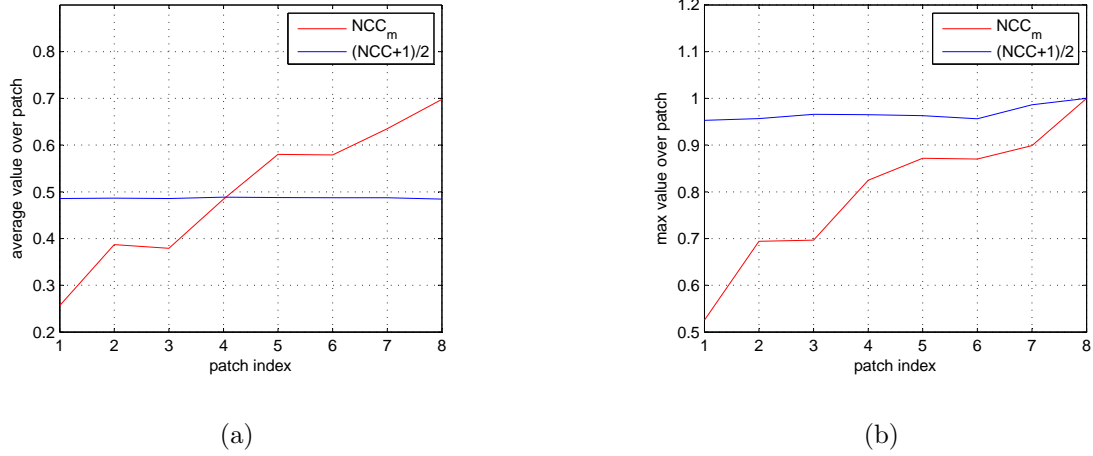


Figure 4.3: Illustration of maximal and average values of scaled NCC and the newly proposed normalized cross-correlation NCC_m measures for the wall example.

4.2 Plane-Sweeping Using Shading Information

The method described in the previous section can use some information about shading, but in some specific cases there can be more information to employ. That is case e.g. when the light source position and surface material reflectance properties are known. It is proved in the Chapter 5, that for some rough real world materials the reflectance function obeys the analytically given Oren-Nayar model. When the sample of the material is available, an experiment can be conducted to obtain the radiance function versus the view angle and the Oren-Nayar model can be fitted to this function using some optimization technique, which yields values of parameters of the surface - roughness and albedo and the value of light flux. Having those values, the radiance can be predicted using the model in general case, for arbitrary source and viewer directions and for arbitrary surface normal.

The plane-sweeping method used in CMP's MVS algorithm can be used as the framework for plane-sweeping method, which uses the position and normal of hypothesized facet, light source and camera to predict the brightness of facet projected into images. The main principle is the expectation, that the facet projection into most of the images will have the correct brightness similar to the one predicted by the suitable reflectance model (Oren-Nayar can be substituted by any other useful model), when the hypothesized facet position corresponds to the position of real surface. Of course, the projection into some images can be misleading, because the surface seen from certain angle from one camera can be hidden by another surface when observed by another camera, especially when the view angles of cameras differ a lot. But the assumption is made, that the same surface can be seen by most of the cameras. This is true for most cases, where the cameras are not very far from

each other.

The principle of plane-sweeping method is illustrated in the Figure 4.4 on a simplified 2D scene. Here the hypothesized facet f is initially placed below the real surface and it has normal with the same direction as the reference camera C_{ref} . Projection of f into cameras C_{ref} , N_1 and N_2 cuts out patches of the real surface f_1 , f_2 and f_3 confined by points where the black dashed lines connecting facet corners with camera centers intersect the real surface. These patches have different normals \vec{n}_{f1} , \vec{n}_{f2} and \vec{n}_{f3} and different light source angles which are not known, however. Even if the light source was modelled as the light source in infinity (with parallel light rays), the light source angles (which are relative to the surface normal) would be different for every patch.

Lets at first assume the Lambert's reflectance model. The patches will have different brightness because of different light source angles. Their brightness is known (because we can measure it from the patches projected into images). The brightness of the hypothesized facet is also different, because, in general, it has different normal then the projected patches. The brightness of f is also known, because it can be predicted using the Lambert's model. When shifting f along the solid blue line, the normals of projected facets on the real surface are approaching until they are (almost) the same when f coincides with the real surface. They are not exactly the same because of the arbitrary shape of the real surface. It should be also noted, that the projected patches change with the normal of f . But when the facet is small enough and its normal does not differ much from the normal of the real surface in the place of intersection of blue line with the real surface, normals of projected patches are close to \vec{n}_{f1} . In the ideal case, the predicted and measured brightnesses by all cameras are the same in this position of f and it can be regarded as the true position of real surface in the scene. In practice there will always be some difference between the measured and predicted brightness, but it should be the smallest when f is near the real surface.

Assuming the Oren-Nayar reflectance model and knowing the positions of all cameras, we can bring more information into this procedure. Now, the view angles wrt. the center of f can also be calculated and used for the brightness prediction. When the real surface has the Oren-Nayar properties, this model predicts better its brightness and therefore the prediction will also match better the measured brightness of projected patches.

The algorithm of this method is listed in Algorithm 2. One camera marked as reference r and a set N of k neighboring cameras of r are chosen. A rectangular facet f with side size l is hypothesized in 3D space, its position is changed along a direction vector of r . In every position it is projected into images of cameras in N and the average brightness of patch which is bordered by the facet's projection is computed. In every position of the facet, the direction vectors of the light source and each camera in N relative to the facet center and the brightness prediction given by Oren-Nayar model are also calculated. The brightness prediction and average brightness of projected facet are subtracted to get the value of *differenceON* for

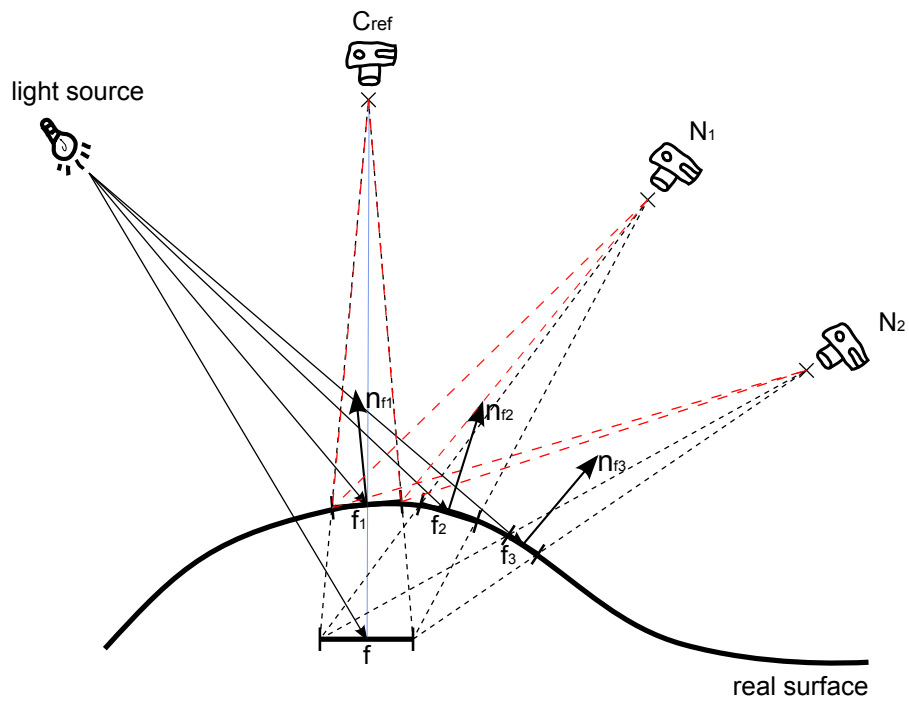


Figure 4.4: Illustration of the plane-sweeping method using the information about shading. For description see the text.

every position of f and every camera in N . The squares of $differenceON$ are then summed over all cameras in N , square root of the sum is calculated and the overall difference $differenceONtotal$ for corresponding facet position is obtained. Finding the minimum of $differenceONtotal$ over all f to r distances, the corresponding distance of f from r is determined, which is the estimated distance of rectangular patch with side length l in the scene and also the output of the Plane-Sweeping algorithm. The procedure can be repeated with different normal of f , reference camera r , set of neighboring cameras N or with different shifting direction so that more estimated distances of real surface in distinct places of scene are obtained. Those results can be averaged or combined in various ways and offer support for MVS NCC based plane-sweeping. It would be interesting to combine those two methods also with the NCC_m measure to get even more robust results. Since this approach was not yet experimentally tested yet, it is proposed for future investigation.

Algorithm 2: Plane-Sweeping

```

1 begin
  input : set  $I$  of  $n$  images of size  $(k \times l)$ , set  $C$  of  $n$  camera projection matrices,
          light source  $L$ , index  $i_r$  of the reference camera, the length  $l$  of side of
          hypothesized facet
  output: estimated distance  $d$  of the real surface from the reference camera
2   $r = C[i_r]$ ;
3   $N \leftarrow k$  neighboring cameras of  $r$  from  $C$ ;
4   $f =$  rectangular facet with side length  $l$ ;
5   $f.normal = r.normal$ ;
6   $f.position = r.position$ ;
7  for  $i = 1$  to  $NumberOfShiftingSteps$  do
8     $f.position += i \cdot step \cdot r.normal$ ;
9     $distance[i] = \text{GetDistance}(f, r)$ ;
10    $lightNormal = \text{GetLightDirection}(L, f)$ ;
11   for  $j = 1$  to  $k$  do
12      $camNormal = \text{GetCameraDirection}(N(j), f)$ ;
13      $radianceON[j] = \text{RadianceOrenNayar}(f, camNormal, lightNormal)$ ;
14      $brightnessProj[j] = \text{GetAvgBrightnessOfProjectedFacet}(f, N[j] \text{ and }$ 
         $\text{corresponding } I)$ ;
15      $differenceON[j] = radianceON[j] - brightnessProj[j]$ ;
16   end
17    $differenceONtotal[i] = \text{Sqrt}(\text{Sum}(differenceON)^2)$ ;
18 end
19  $min_i = \text{GetIndexOfMin}(differenceON)$ ;
20  $d = distance[min_i]$ ;
21 end

```

Chapter 5

Experiments

In first part of this chapter a similar experiment to the one conducted by Oren and Nayar in their paper [19] is described. The effort was made to find a real-world material which truly has different reflectance properties than the widely used Lambert's model predicts and which obeys Oren-Nayar model in the same time. Moreover we were trying to find such material that is similar to those which can be found on Mars. Mars's surface is mostly covered with fine dust, sand and rocks of different sizes as illustrated on photo of typical Mars environment taken by NASA rover Spirit from the mission Mars Exploration Rover (MER) [3] on Figure 5.1. These materials are expected to have Oren-Nayar model behaviour. When verified that Oren-Nayar model works for some real-world material, it can be then used in the shape-from-shading and alike algorithms for more accurate 3D reconstruction of objects made of this material.

In the next section of this chapter another experiment is described examining the shape of a cylinder covered with one of materials with Oren-Nayar behaviour. A plane-sweeping based photo-consistency measure using informations about shading and predictions of Oren-Nayar model is us used with the cylinder images and images of the Mars's surface.

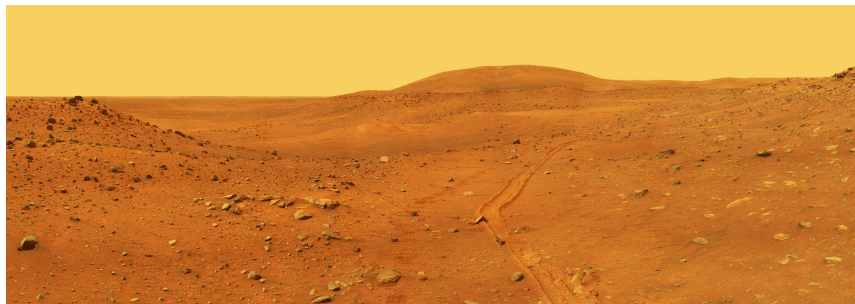


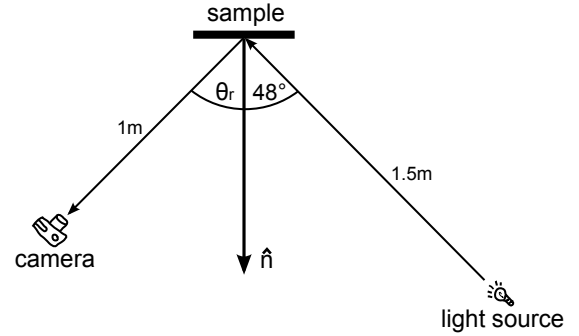
Figure 5.1: Photo of a typical Mars's surface. It is a fraction of panoramic photo taken by NASA Spirit rover Pancam in June 2009. The picture is in approximate true-color, it is reconstruction composed of many photos.

5.1 Verifying Oren-Nayar Reflectance Model

For verifying the Oren-Nayar reflectance model, an experiment was conducted to measure the radiance of real-world surfaces. On Figure 5.2(a) is photograph of the experiment set-up for measuring the radiance function, Figure 5.2(b) depicts the same on a scheme. 11Mpixel digital camera Canon EOS-1D was mounted on 1m long rotary beam pointing on the sample material fixed in the axis of beam rotation. An intense white light lamp (in upper right corner of Figure 5.2(a)) was posted behind the camera, 1.5m far from the sample, 24° above the beam rotation plane, so that no shadowing from the camera could appear. The light source angles with respect to the sample material surface plane normal and beam rotation plane was $\theta_i = 48^\circ$, $\phi_i = 24^\circ$, respectively. Then the beam with camera was rotated using accurate positioning device so that the view angle was in interval $\theta_r = \langle -85^\circ, 85^\circ \rangle$ with step 5° and the plane of rotation was defined to have $\phi_r = 0^\circ$. In every position the picture was taken under the same illumination with camera in manual mode with parameters f-stop = $f/16$ and shutter speed = 0.5s. During the measurement, all lights except of the directional lamp were shutted down and windows were dimmed-out. On Figure 5.3 is the set of chosen sample materials, size of each of these rectangular samples is approximately $1.5\text{cm} \times 1.5\text{cm}$ except for the white and black painting samples which are $0.5\text{cm} \times 0.5\text{cm}$.



(a) Photo of experiment set-up.



(b) Scheme of experiment set-up for measuring reflectance of real surfaces. \hat{n} is the sample surface normal.

Figure 5.2: Experiment set-up for measuring the radiance functions of real-world materials.

Series of images of sample materials viewed from different θ_r angles were obtained, such as the one for rough sand sample on Figure 5.4. Sample images were then rectified wrt. the central image I_c using homography (explained in section 1.2.4) matrices \mathbf{H}_i computed from 4 manually selected correspondence points for every image. The correspondence points were chosen so that each one belongs to a feature which can be seen on every image from all the investigated angles. This was a problem for the images taken from large θ_r angles, where the features are distorted too



Figure 5.3: Set of investigated real-world material samples. From the top left: red clay, light rough sand, light fine sand, modeller material, cotton pad, white matt painting, abrasive paper, abrasive rock (carborundum), black matt painting.

much to assign a correctly corresponding points, therefore those images were omitted from rectification. The transformation is computed so that all points $\mathbf{I}_i(x, y)$ in every image \mathbf{I}_i hold following equation:

$$\mathbf{I}_i(x, y) = \mathbf{H}_i \cdot \mathbf{I}_c(x, y), \quad (5.1)$$

where \mathbf{H} is different for every image. After this coordinate transformation the images can be displayed as they were viewed from the same point, with the same view angle and can be directly compared to each other pixel by pixel.

The radiance functions of real surfaces were obtained for each color channel r, g, b separately. The overall brightness is calculated as an average over the three color channels and whole area of the sample. The averaging is important, because Oren-Nayar model properties appear only when the size of surface elements is small enough so that lot of these elements projects into one pixel of camera. And since in the laboratory experiment it was feasible to have only 1m long beam, i. e. sample to camera distance, the surface was too close and surface elements appear sharply in the images (see clay and sand samples on the Figure 5.3. By averaging we acquire similar effect as moving the camera farther.

Experimentally obtained radiance functions of the 9 real-world sample materials are plotted on Figure 5.5. Some of these functions show behaviour similar to the Oren-Nayar model, most similar are rough and fine sand samples and red clay sample, they are plotted with a thick line. These materials have relatively rough and granular surface, where the particles casts small shadows when the light source

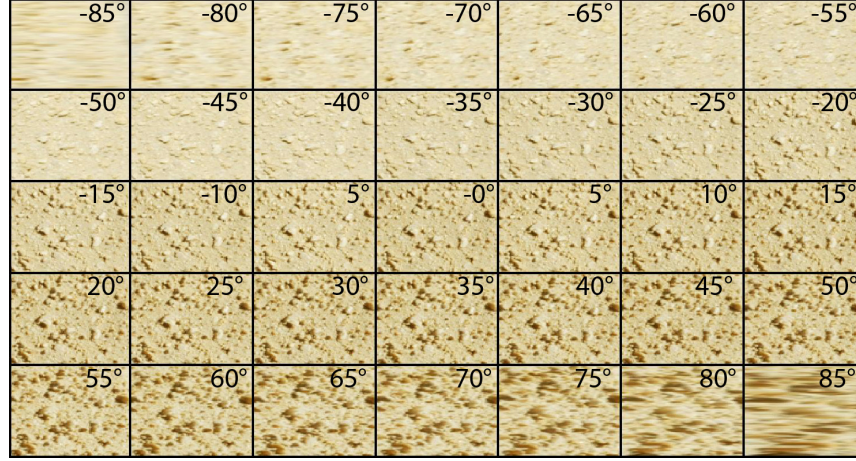


Figure 5.4: Rough sand sample picture set. Sample patch is viewed from 35 different angles. It can be seen, that the overall brightness is changing from top left $\theta_r = 85^\circ$ to bottom right $\theta_r = -85^\circ$

angle is large. This property seems to be the main pre-requisite for Oren-Nayar model behaviour. Some other materials like the modeller material, abrasive rock and black painting show also supposition of increasing brightness when the view angle θ_r approaches the light source angle θ_i , but all of them have also large specular component in opposite direction to the light source direction. The abrasive paper, which has quite rough surface with noticeable granulas was expected to have Oren-Nayar model behavior, but it has very large specular reflectance component instead. That is caused by small glass granulas which are glued on the surface, which alone have specular properties. Oren-Nayar model assumes, that the small granulas forging the surface are Lambertian. On the other hand, none of these materials is truly Lambertian, the only one which is very close to it, though, is the cotton sample.

On Figure 5.6 is compared measured radiance function of the rough sand sample with modeled Oren-Nayar radiance function, whose unknown parameters Φ_i , ρ and σ were found using the Matlab's least squares fitting function *lsqcurvefit* and they are $\Phi_i = 2470.3W$, $\rho = 1$, $\sigma = 11.4^\circ$. The viewer and light source directions are setted to the same values as during the experiment: $\theta_i = -48^\circ$, $\phi_i = 24^\circ$, $\phi_r = 0^\circ$. It can be seen, that reflectance properties of this sample are in strong agreement with the Oren-Nayar model.

5.2 Cylinder Experiment

In this experiment the reflectance of surface of a regular cylinder of proportions 10cm in diameter and 10cm in height was investigated. The cylinder model was made out of polyvinyl chloride pipe on whose surface was glued cca 2mm thick layer of rough sand, the same which was proved to have Oren-Nayar properties

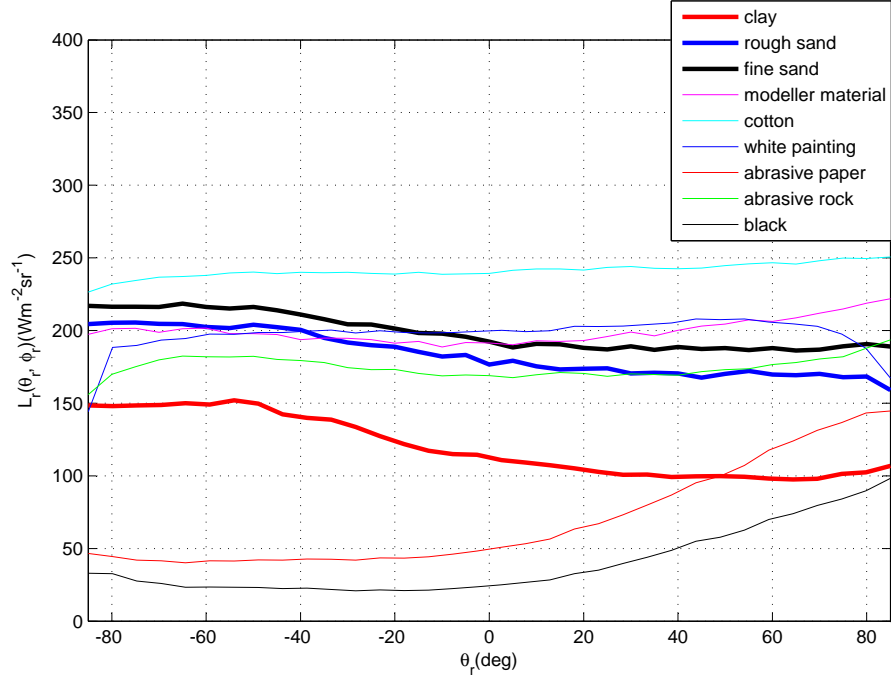


Figure 5.5: Reflectance functions of real-world surfaces. The experimental set-up for obtaining these functions is depicted on the scheme in Figure 5.2(b)

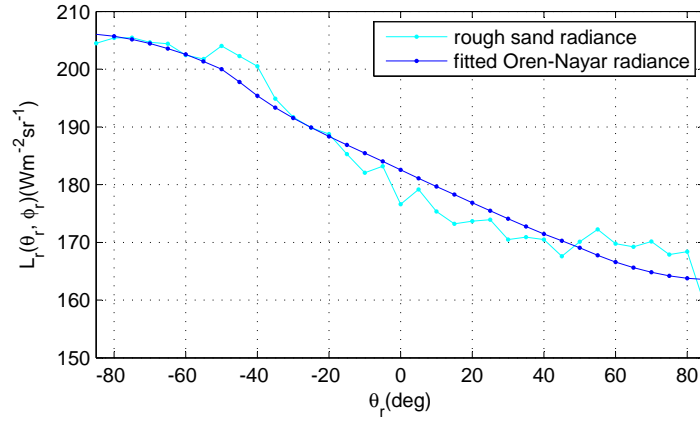
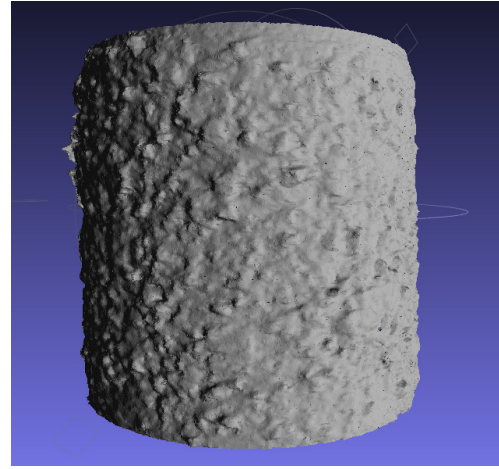


Figure 5.6: Oren-Nayar radiance fitted to rough sand sample radiance function. The parameters of the model are $\Phi_i = 868.6W$, $\rho = 1$, $\sigma = 11.2^\circ$

in the the previous section. The cylinder model is shown on the Figure 5.7(a). The experiment was conducted in the same way as in the previous section with camera moving in one plane, changing only the θ_r angle with the step 5° in interval $\langle -85^\circ, 85^\circ \rangle$ and the light coming from the angles $\theta_i = -48^\circ$ and $\phi_i = 24^\circ$. Again, the set of 35 images of the cylinder viewed from different angles under constant illumination was obtained. Using the CMP's multi-view stereo algorithm (described in section 2.3), very accurate 3D model of the cylinder was reconstructed, which is used in following processing as the ground truth. The reconstruction viewed in the MeshLab software [1] is depicted in the Figure 5.7(b).



(a) The cylinder model with rough sand on surface. Its size is approximately 10cm in diameter and 10cm in height.



(b) The cylinder model reconstructed with CMP's multi-view stereo algorithm viewed in the MeshLab software

Figure 5.7: The cylinder model

A plane sweeping algorithm, similar to the one proposed in section 4.2 was implemented in Matlab(R2010b) [17]. The goal was to verify that it can be used to find a 3D position of surface with known parameters in a scene with known lighting conditions. Particularly then, if Oren-Nayar reflectance model can accomplish better results than the less informed Lambert's model.

5.2.1 Fitting the Reflectance Model

Using this method, it is always needed to obtain the parameters of the surface material at first. When the parameters roughness σ and albedo ρ of the material are not known beforehand, a controlled experiment has to be conducted. In this experiment, the exact position and normal of the sample surface, direction of the light source and camera positions have to be known and after processing also the brightness versus view angle. Having all those informations, we have all necessary parameters which the Oren-Nayar model needs, except for the roughness σ , albedo

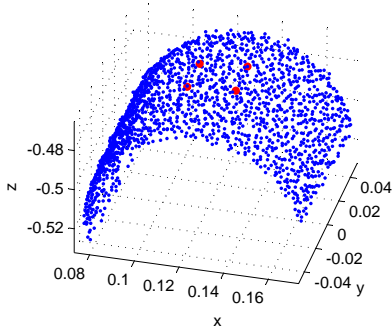
ρ and the light flux Φ . Those parameters can be found using some optimization technique to fit the analytical expression of Oren-Nayar onto the course of measured brightnesses versus view angle. In this experiment most parameters were measured in the lab, but to have precise position of the surface and camera directions the CMP's MVS reconstruction algorithm is used here. It processes the set of images of the scene and gives very accurate 3D reconstruction in the form of the 3D point cloud and exact positions and directions of cameras.

At first a 3D point cloud *p3d* from the stereo reconstruction was loaded into Matlab to have the ground truth information for comparison with the position of facet determined from the shading information. 4 points from the front of the point cloud were selected, representing the corners $c_{f,j}$ of the fitting facet f_f , they are plotted in Figure 5.8(a) in red color together with blue points of the 3D point cloud. Then the normal n_{ff} of f_f is calculated. The point cloud together with the complete set of 35 cameras with matrices P_i , their relative positions and orientations is shown in Figure ??, the plot is in the same scale as the real experiment, all distances are in meters. Having those informations, the facet corners $c_{f,j}$ are transformed into homogeneous coordinates $c_{f,hom,j}$ and projected into every camera image using

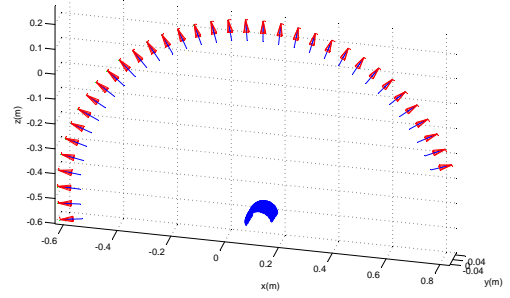
$$cp_{f,hom,j,i} = P_i \cdot c_{f,hom,j}. \quad (5.2)$$

The projections into input images transformed back to affine coordinates $cp_{f,j,i}$ are depicted as the red points in Figure 5.9 for each view angle. The average brightness of the polygonal areas circumscribed by $cp_{f,j,i}$ in input images is calculated.

Now the Matlab's function *lsqcurvefit* is used to find yet unknown parameters of the Oren-Nayar model - roughness σ , albedo ρ_{ON} and light flux Φ_{ON} and also parameters of Lambert's model albedo ρ_L and light flux Φ_L . The function *lsqcurvefit* is provided with the reflectance model function (Oren-Nayar and Lambert's, respectively), fitting facet normal n_{ff} , light source and camera directions wrt. the facet's center and measured brightness B_i for every view position. The unknown parameters of the reflectance model function are stated explicitly and *lsqcurvefit* finds values of these parameters, which minimize the difference between the measured brightness B_i and the brightness calculated by the reflectance model function. Obtained values of these parameters together with the list of all variable's symbols used in this experiment are stated in Table 5.1. The measured brightness together with predicted radiances (revise, that here radiance is equivalent to brightness) are plotted in Figure 5.10. Again, the modeled Oren-Nayar radiance function fits quite precisely the rough sand radiance function, eventhough the facet is laying now on a curved surface of the cylinder. The size of this facet is chosen small enough (approximately $\frac{1}{25}$ of the area of frontal projection of the cylinder) and its position in the center of cylinder, that the Oren-Nayar behavior clearly appears. The values for the few left-most cameras are not plotted, because their θ view angles wrt. the center of the fitting facet are larger than 90° and therefore they cannot see the surface of the fitting facet. This fact can be observed also from the Figures 5.8(b) and 5.9.



(a) The cylinder point cloud loaded into Matlab, with fitting facet corners plotted in red color.



(b) The cylinder point cloud with a complete set of cameras.

Figure 5.8: The cylinder experiment.

$p3d$	3D point cloud from MVS reconstruction
$c_{f,j}$	3D position of 4 fitting facet corners
$cp_{f,j,i}$	fitting facet corners projected into 2D coordinates of input images
n_{ff}	fitting facet normal
x_l, y_l, z_l	position of the light source
P_i	set of 35 camera matrices
I_i	set of 35 input images
$\sigma = 11.7^\circ$	roughness for Oren-Nayar model
$\rho_{ON} = 0.074$	albedo for Oren-Nayar model
$\Phi_{ON} = 306.5W$	light flux for Oren-Nayar model
$\rho_L = 0.146$	albedo for Lambert's model
$\Phi_L = 146.4W$	light flux for Lambert's model

Table 5.1: Summary of known parameters for the cylinder experiment.

5.2.2 Plane-Sweeping

A plane sweeping algorithm was programmed in Matlab to find if the Oren-Nayar model can predict correct position of hypothesized facet and if it can bring some more knowledge into the plane sweeping used by MVS algorithms. In the Figure 5.11 is visualized the plane sweeping method for one reference camera *ref* and one camera from the set of neighboring N , denoted as *proj* (for projection camera). The blue point cloud is the reconstructed surface points of the cylinder and the green rectangle is the facet on different positions during the plane sweeping, it is moved in the direction of view of the reference camera. In the top right corner is the position of the light source. The brightness difference function versus distance of the facet from the reference camera obtained by summing the square of brightness difference over 10 cameras from N is on the Figure 5.12. The difference functions



Figure 5.9: Cylinder with projected facet corners. The camera views are -85° in the top left image, increasing when moving right and down to 85° in the bottom right corner.

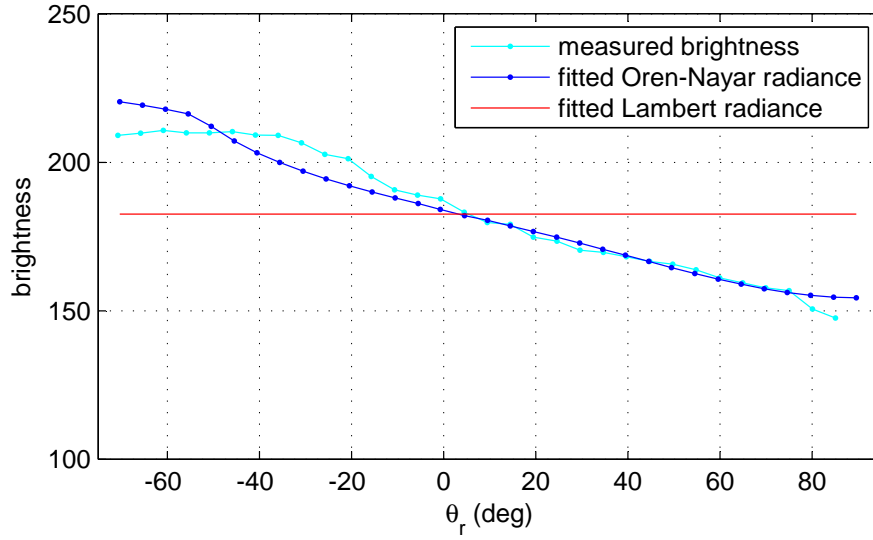


Figure 5.10: Fitted Oren-Nayar and Lambertian radiances for cylinder surface compared with the measured brightness of fitting facet projected into each input image. The parameters obtained by fitting are light flux $\Phi_{ON} = 306.5W$, albedo $\rho_{ON} = 0.074$, roughness $\sigma = 11.7^\circ$ for the Oren-Nayar model and $\Phi_L = 146.4W$, $\rho_L = 0.146$ for the Lambert's model.

using both Oren-Nayar and Lambert's models are plotted there for comparison. In the minimum of this curves (the black dot) is the estimated distance of the real surface from the reference camera. It can be seen that it is different for both reflectance models. The ground truth position of the surface is in the distance of 0.743m from the reference camera and it is indicated by the solid black line. Oren-Nayar model estimates the position exactly in the distance of 0.743m (with chosen shifting step) and Lambert's model in the distance of 0.724m from the reference camera, the errors are 0cm and 1.9cm respectively, the Oren-Nayar model is more accurate. Although, here, the normal of the hypothesized facet was set to the normal of the fitting facet, i.e. the normal of the real surface, which is unknown in general case. Parameters of this plane-sweeping run are summarized in Table 5.2.

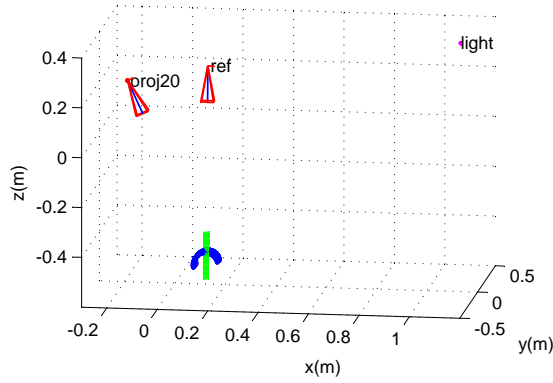
cameras in N	P_{10} to P_{20}
reference camera	P_{15}
facet side size	2 cm
facet normal	same as the fitting facet normal
shift step	0.47cm
Oren-Nayar model estimation error	0cm
Lambert's model estimation error	1.9cm

Table 5.2: Parameters and results of the plane-sweeping algorithm run #1 with the cylinder data.

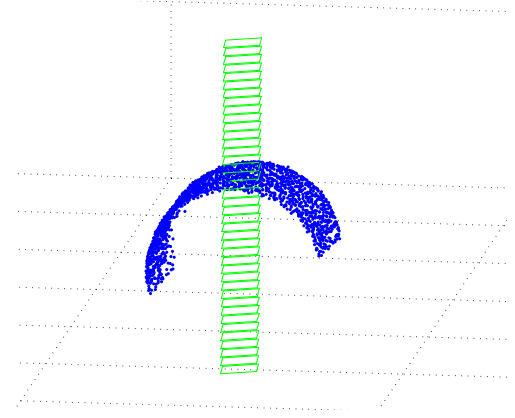
The result of another run of plane-sweeping algorithm with different parameters listed in Table 5.3 is in Figure 5.13. The main difference is in setting the hypothesized facet normal parallel to the reference camera view direction, which is different from the real surface normal direction and will mostly be the case when using the algorithm. Here the facet position estimated using the Oren-Nayar model is 2.37cm closer to the reference camera than is the position of the real surface. But it is still more accurate than estimation using the Lambert's model, which is 3.32cm far from the ground truth.

cameras in N	P_{10} to P_{20}
reference camera	P_{15}
facet side size	1 cm
facet normal	parallel with the reference camera direction
shift step	0.47cm

Table 5.3: Parameters and results of the plane-sweeping algorithm run #2 with the cylinder data.



(a) Illustration of the plane-sweeping method. *proj* and *ref* are cameras figuring in one step of the algorithm, blue points form the surface of cylinder and green rectangles is the facet in different positions throughout the whole algorithm. In the top right corner is the position of the light source.



(b) Detail of the facet positions and cylinder surface.

Figure 5.11: Plane-sweeping algorithm run#1. The plot is in the same scale as the laboratory experiment.

5.3 Real Mars Data Experiment

Next simulations were done using the gray scale images of the real Mars's surface captured by the NASA's Phoenix Mars Mission landing module [4], which are depicted in Figure 5.14(a). Again, the 3D model reconstructed by the CMP's MVS algorithm shown on Figure 5.14(b) was used as the ground truth. From the 9 pictures provided in the dataset, two subsets of images which contain the same feature were picked up. Only few images fulfil this requirement. The first subset *A* is composed of the images 1, 2, 3, 4 and with chosen feature (selected from the 3D point cloud provided by MVS reconstruction) projected into the images using corresponding camera matrices is in Figure 5.15(a). The second subset containing images 3, 4, 5 is in Figure 5.15(b).

5.3.1 Set *A* Simulations

The model fitting with image set *A* was done in the similar manner as in the previous section in the cylinder experiment. The fitted Oren-Nayar and Lambertian radiances compared with measured brightness are plotted in Figure 5.16. One difference is, that the light source direction is not known, so it has to be guessed. There are only the measured brightness data near to $\theta_r = 0$, so we don't know how

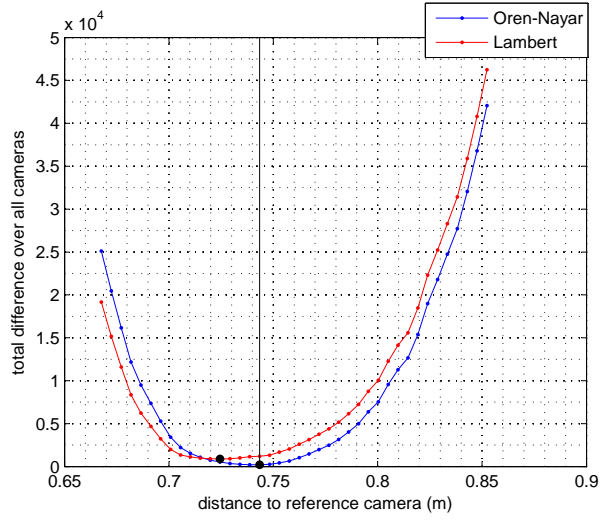


Figure 5.12: Total brightness difference over all cameras in the set N versus the distance of hypothesized facet to the reference camera. The result of plane-sweeping algorithm run #1 with parameters summarized in Table 5.2.

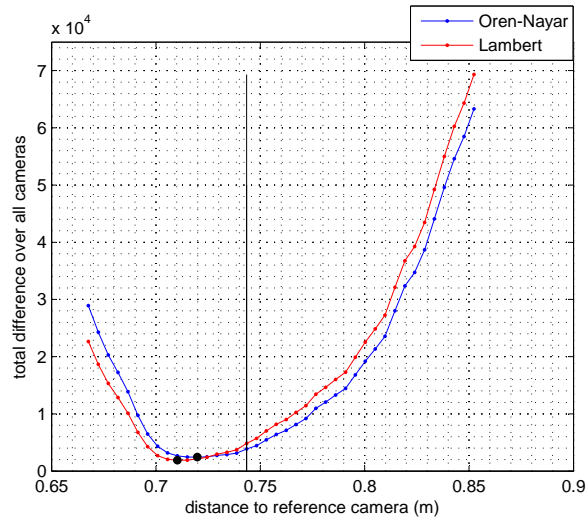
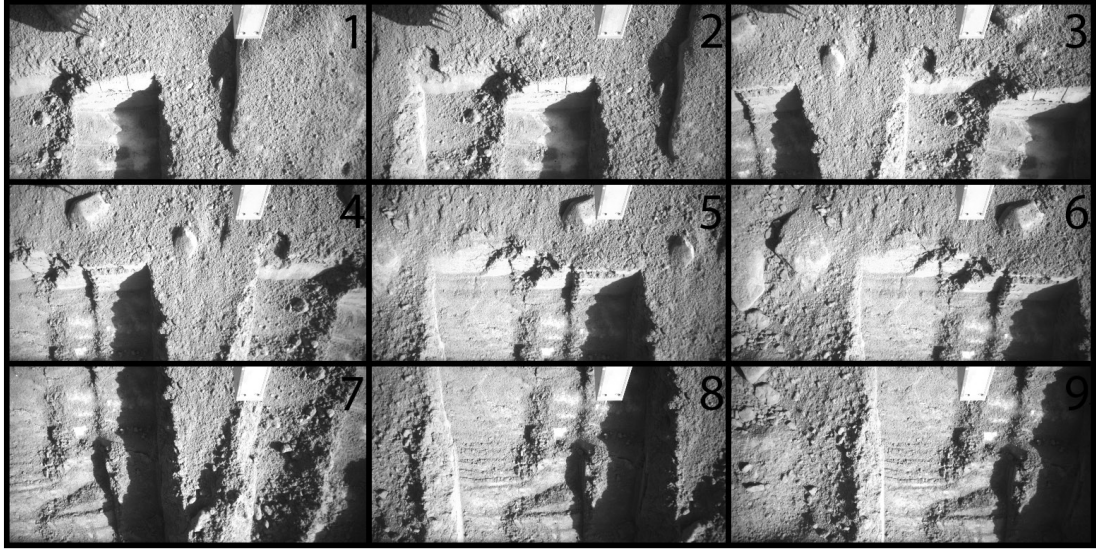
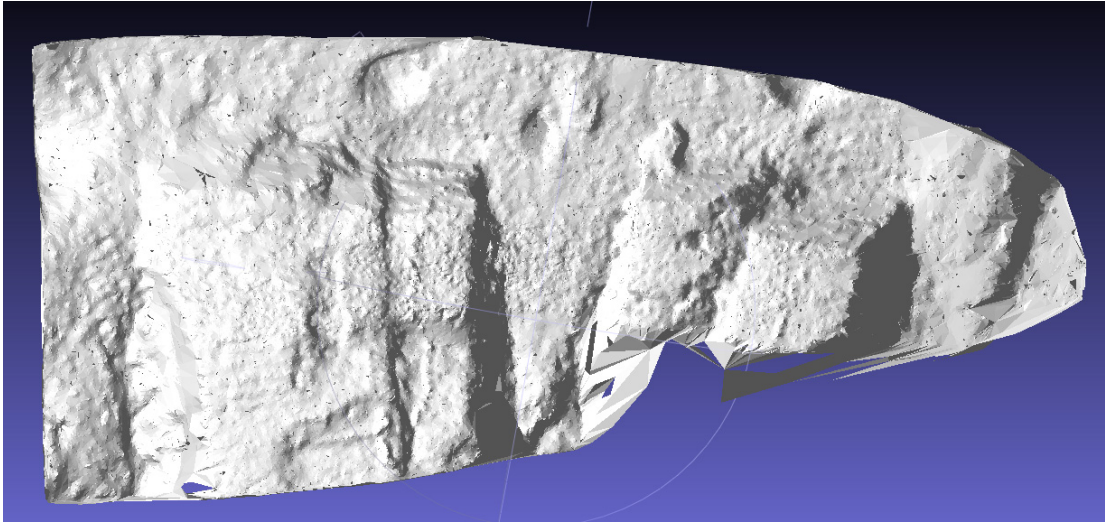


Figure 5.13: Total brightness difference over all cameras in the set N versus the distance of hypothesized facet to the reference camera. The result of plane-sweeping algorithm run #2 with parameters summarized in Table 5.3.



(a) Images of the real Mars's surface captured by the NASA's Phoenix Mars Mission landing module [4].



(b) 3D model reconstruction of the Mars's surface created by CMP's MVS algorithm.

Figure 5.14: Images and 3D reconstruction of Mars's surface.

the right and left sides of Oren-Nayar radiance should look like. Also for the data available, the radiance cannot be fitted very precisely, Oren-Nayar radiance differs in particular data points even with the value around 20 brightness units. However, as will be seen later, such precision is enough to outperform the Lambert's model.

The plane-sweeping algorithm with parameters listed in Table 5.4 was used for the image set A . The normal of the hypothesized facet was set to be parallel with the reference camera view direction. The algorithm was executed several times to find

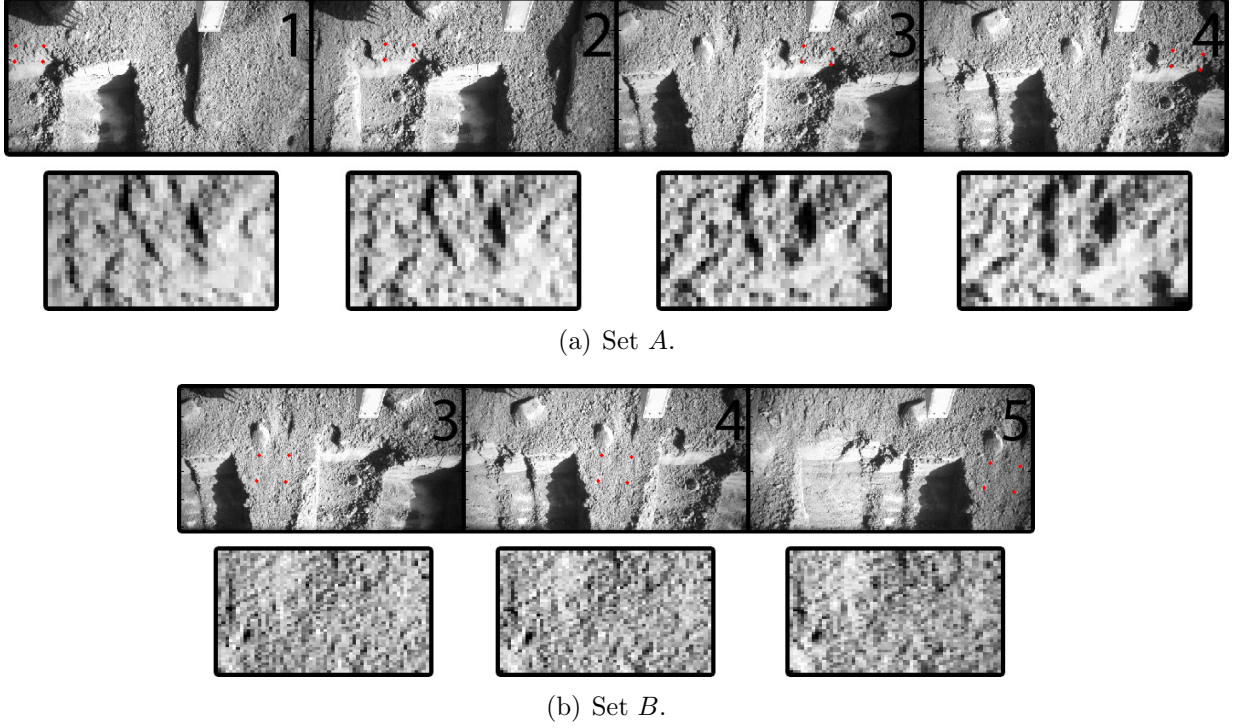


Figure 5.15: Chosen sets of images sharing the same feature. The same patch is selected in the 3D point cloud and it is chosen as the fitting facet, projection of its corners into each image is marked with red dots. Under the images are details of the rectified facet projection.

the optimal light source direction. Illustration of the setting of the scene with camera positions and positions of the hypothesized facet is in the Figure 5.17. It should be noted, that in this experiment the exact scale is not known, we don't know the units of length. The result of the algorithm is on Figure 5.18. Here the prediction of the position of the real surface with Oren-Nayar model is exactly at the real surface position (which is now set to be in zero). The predicted surface position using the Lambert's model is 0.4469 units far from the ground truth. These precise results are obtained because of the manual fine tuning of the guessed light direction, but even when the light direction is little bit different, the results are still quite satisfactory as proved on Figure 5.19 by another run of the algorithm with slightly different parameters listed in Table 5.6.

5.3.2 Set B Simulations

The reflectance models fitted on image set *B* compared with the measured brightness are shown in Figure 5.20. Here even less data points are available, the feature occurs only in 3 images. But the fitted Oren-Nayar model matches little bit better the measured brightness than with the image set *A*. The plane-sweeping algorithm

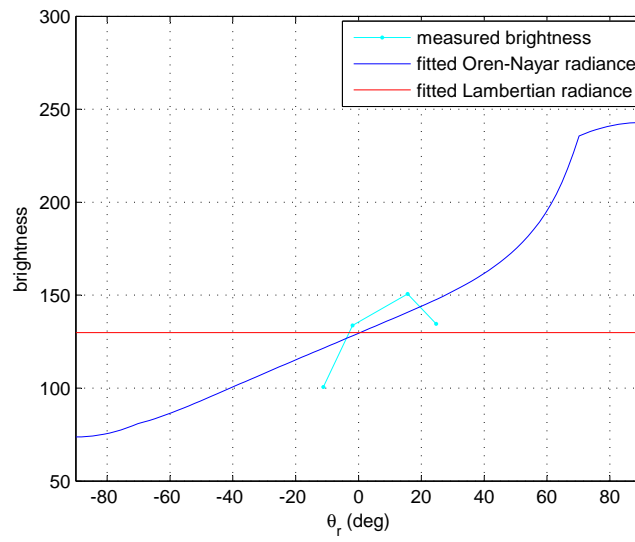


Figure 5.16: Fitted Oren-Nayar and Lambertian radiances for Mars's surface image set A , compared with the measured brightness of fitting facet projected into each input image. The parameters obtained by fitting are light flux $\Phi_{ON} = 652.3W$, albedo $\rho_{ON} = 1$, roughness $\sigma = 26.0^\circ$ for the Oren-Nayar model and $\Phi_L = 1150.6W$, $\rho_L = 0.534$ for the Lambert's model. The 4 data points correspond from the right to the left side to images 1, 2, 3, 4, respectively.

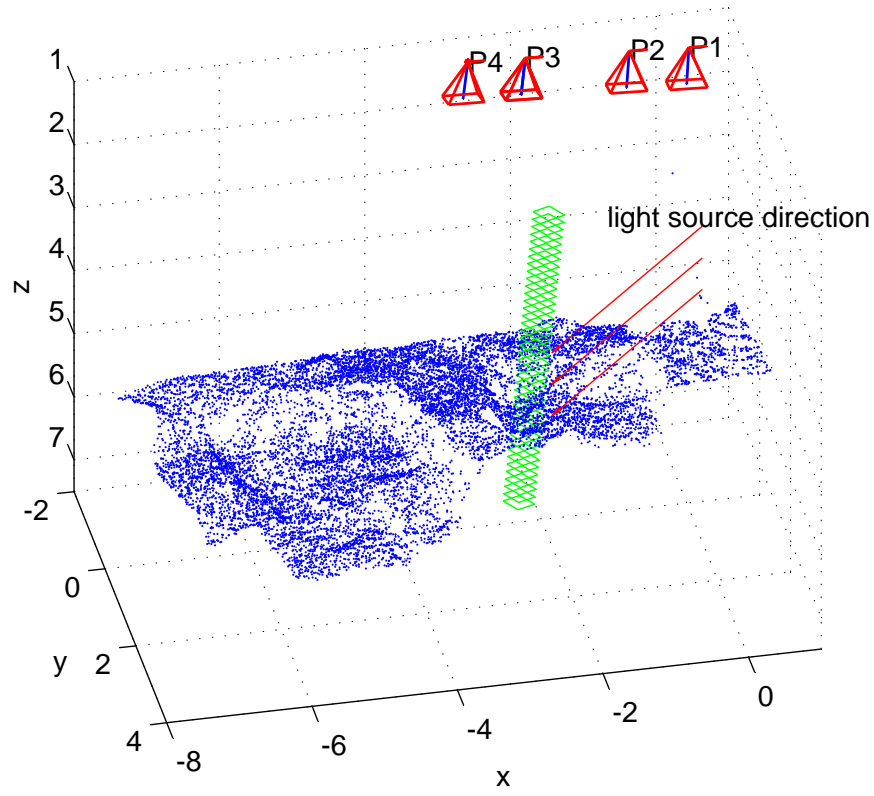


Figure 5.17: Illustration of plane-sweeping with Mars image set A . P_1, P_2, P_3, P_4 denote cameras corresponding to images in set A , green rectangles are the positions of the hypothesized facet throughout the algorithm and blue points is the 3D point cloud generated by CMP's MVS algorithm. The light source is assumed to be in infinity, it has parallel rays with guessed direction $\theta_i = 57^\circ, \phi_i = 0^\circ$, which are visualized as the red arrows.

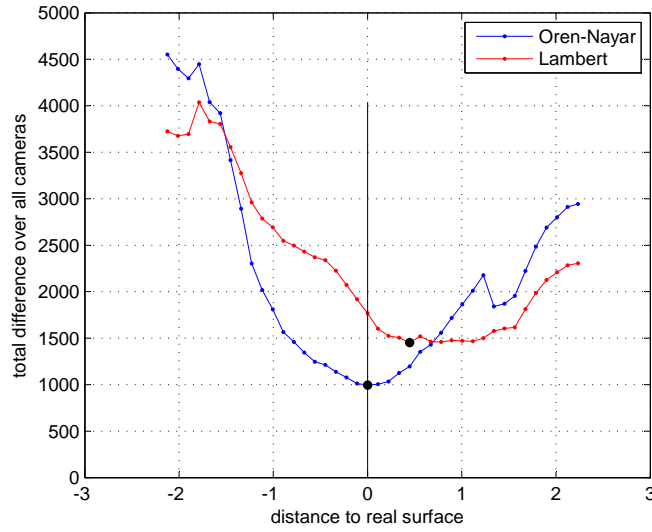


Figure 5.18: Total brightness difference over all cameras in the set N versus the distance of hypothesized facet to the real surface. The result of plane-sweeping algorithm run #1 with the Mars image set A and parameters summarized in Table 5.4.

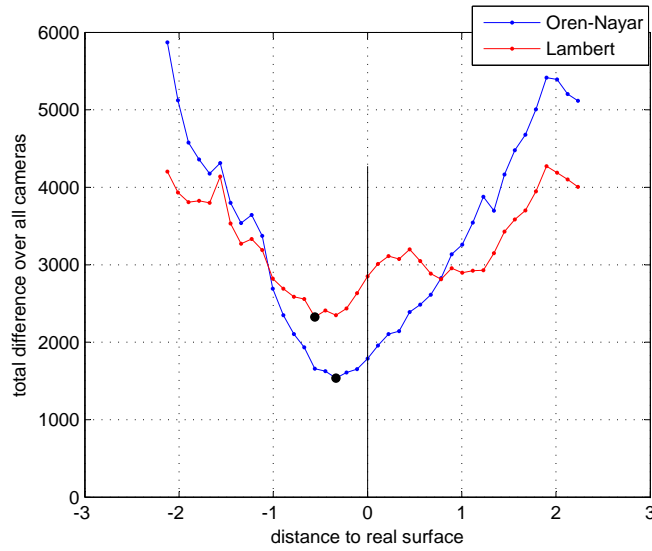


Figure 5.19: Total brightness difference over all cameras in the set N versus the distance of hypothesized facet to the real surface. The result of plane-sweeping algorithm run #2 with the Mars image set A and parameters summarized in Table 5.6.

guessed light source direction	$\theta_i = 57^\circ, \phi_i = 0^\circ$
cameras in N	P_1, P_2, P_3, P_4
reference camera	P_3
facet side size	0.5 (unknown units)
facet normal	parallel with the reference camera direction
shift step	0.112 (unknown units)
Oren-Nayar model estimation error	0 (unknown units)
Lambert's model estimation error	0.4469 (unknown units)

Table 5.4: Parameters and results of the plane-sweeping algorithm run #1 with the Mars image set A .

guessed light source direction	$\theta_i = 50^\circ, \phi_i = 0^\circ$
cameras in N	P_1, P_2, P_3, P_4
reference camera	P_3
facet side size	0.3 (unknown units)
facet normal	parallel with the reference camera direction
shift step	0.112 (unknown units)
Oren-Nayar model estimation error	0.335 (unknown units)
Lambert's model estimation error	0.559 (unknown units)

Table 5.5: Parameters and results of the plane-sweeping algorithm run #2 with the Mars image set A .

executed with parameters in Table 5.6 results into plot on Figure 5.21. The estimated position of the surface using both, Oren-Nayar and Lambert's models, is in distance 1.229 units from the ground truth surface. Here the Oren-Nayar model does not acquire better results than the simpler Lambert's model even with the best estimated light direction. It is probably caused by the lack of different camera views, which is the main difference between the image sets A and B . It can be observed, that in some cases the total brightness difference curve can be approximated with a polynomial of the 2nd order to comprise the overall trend in the area near the minimum to get more accurate result. These approximations are shown in the Figure 5.21 as the solid lines.

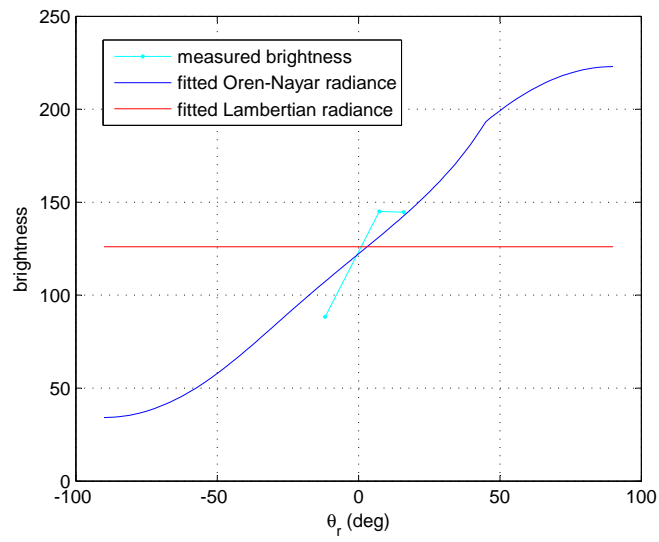


Figure 5.20: Fitted Oren-Nayar and Lambertian radiances for Mars's surface image set B , compared with the measured brightness of fitting facet projected into each input image from B . The parameters obtained by fitting are light flux $\Phi_{ON} = 10000W$, albedo $\rho_{ON} = 0.05178$, roughness $\sigma = 90.0^\circ$ for the Oren-Nayar model and $\Phi_L = 623.89W$, $\rho_L = 0.51$ for the Lambert's model. The 3 data points correspond from the right to the left side to images 3, 4, 5, respectively.

guessed light source direction	$\theta_i = 57^\circ, \phi_i = 0^\circ$
cameras in N	P_3, P_4, P_5
reference camera	P_3
facet side size	0.5 (unknown units)
facet normal	parallel with the reference camera direction
shift step	0.112 (unknown units)
Oren-Nayar model estimation error	1.229 (unknown units)
Lambert's model estimation error	1.229 (unknown units)

Table 5.6: Parameters and results of the plane-sweeping algorithm run #1 with the Mars image set B .

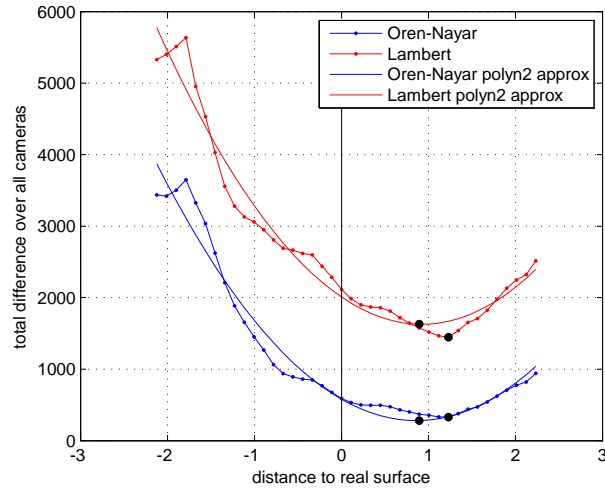


Figure 5.21: Total brightness difference over all cameras in the set N versus the distance of hypothesized facet to the real surface. The result of plane-sweeping algorithm run #1 with the Mars image set B and parameters summarized in Table 5.6.

Chapter 6

Conclusion

This work attempts to enhance the MVS reconstruction algorithm by incorporating clues from shading information. The properties of the Oren-Nayar reflectance model are investigated in detail so it can be fully understood and used to extract as much information from the brightness of objects in the images as possible. Reflectance properties of different materials were examined in the laboratory experiment, finding that behaviour of some of them, and most importantly of those of high relevance to planetary exploration (), can be very accurately described by the Oren-Nayar model.

The main contribution of this work are then two methods proposed to improve the MVS photo-consistency measure. First of them, NCC_m modifies the NCC function so it preserves its beneficial ability to distinguish places in two images with correlating intensity variations and adds the ability to distinguish between places, which differ in intensity by a constant over whole area of correlation window. Second method is the adapted plane-sweeping method using the Oren-Nayar and Lambert's models to employ shading predictions for determining the position of the real surface in 3D scene. Experiments with images from the laboratory as well as with images of Mars's surface showed, that the results of plane-sweeping method using the Oren-Nayar model surpass the results obtained with the use of Lambert's model when the surface material parameters are known with sufficient accuracy. With precisely estimated material parameters and the light source and camera directions, it is possible to achieve the accuracy of centimeters, as showed the experiment with the cylinder with sand on its surface.

6.1 Future Work

The plane-sweeping method should be tested in the future on more various surfaces and under different illumination conditions to examine more properly in which scenarios it is useful. One way of improvement would be to equip the future imaging system with an automated tool to obtain the parameters of the material.

It could for example use its own light source and measuring instrument to conduct an in situ experiment to obtain the radiance function of the unknown material.

In the future, each of the two proposed methods should be implemented into the CMP's MVS pipeline to see how much they can improve the accuracy of the used NCC based plane-sweeping method. Next direction will be to investigate the exact way how to combine the two methods and NCC plane-sweeping all together to get the best possible results.

Bibliography

- [1] Meshlab, open source software, 2011. Available from: <http://meshlab.sourceforge.net/>.
- [2] Middlebury multi-view dataset and comparison of multi-view stereo algorithms[online], 2011. Available from: <http://vision.middlebury.edu/mview/eval>.
- [3] Nasa exploration rovers mission [online], 2011. Available from: http://www.nasa.gov/mission_pages/mer/index.html.
- [4] Phoenix mars mission, 2011. Available from: <http://phoenix.lpl.arizona.edu/index.php>.
- [5] Provisg - planetary robotics vision ground processing project, 2011. Available from: <http://www.provisgt.eu/>.
- [6] S. Baker, T. Sim, and T. Kanade. When is the shape of a scene unique given its light-field: A fundamental theorem of 3d vision? *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 25(1):100 – 109, January 2003.
- [7] C. K. Chow and S. Y. Yuen. Recovering shape by shading and stereo under lambertian shading model. *Int. J. Comput. Vision*, 85:58–100, October 2009.
- [8] R. T. Collins and R. T. Collins. A space-sweep approach to true multi-image matching, 1996.
- [9] D. A. Forsyth and J. Ponce. *Computer Vision: A Modern Approach*. Prentice Hall Professional Technical Reference, 2002.
- [10] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.
- [11] V. H. Hiep, R. Keriven, P. Labatut, and J.-P. Pons. Towards high-resolution large-scale multi-view stereo. In *CVPR*, pages 1430–1437. IEEE, 2009.
- [12] B. K. P. Horn. Obtaining shape from shading information. *The Psychology of Computer Vision*, 1975.

- [13] M. Jancosek and T. Pajdla. Segmentation based multi-view stereo. *Computer Vision Winter Workshop 2009*, 2009.
- [14] M. Jancosek and T. Pajdla. Hallucination-free multi-view stereo. 2010.
- [15] M. Jancosek and T. Pajdla. Segmentation based multi-view stereo. 2011.
- [16] K. N. Kutulakos and S. M. Seitz. A theory of shape by space carving. *Int. J. Comput. Vision*, 38:199–218, July 2000.
- [17] MathWorks. Matlab, the language of technical computing, 2011. Available from: <http://www.mathworks.com/products/matlab/>.
- [18] R. O’Hara and D. Barnes. A new shape from shading technique with application to mars express hrsc images. 2010.
- [19] M. Oren and S. K. Nayar. Generalization of lambert’s reflectance model. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, SIGGRAPH ’94, pages 239–246, New York, NY, USA, 1994. ACM.
- [20] T. Pajdla. Theory of computer vision, graphics and interaction - lecture notes, 2011.
- [21] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1*, pages 519–528, Washington, DC, USA, 2006. IEEE Computer Society. doi:10.1109/CVPR.2006.19.
- [22] J. A. Sethian. Advancing interfaces: Level set and fast marching methods, 1999.
- [23] C. X. Yan. Planet photo-topography using shading and stereo, master’s thesis. 1993.
- [24] L. Zhang, B. Curless, A. Hertzmann, and S. M. Seitz. Shape and motion under varying illumination: Unifying structure from motion, photometric stereo, and multi-view stereo. 1993.

CD content

The CD is attached to the printed version of this work containing the text of the Thesis in a PDF file and Matlab source codes of the plane-sweeping algorithm and for testing the Oren-Nayar reflectance models. The data used in the cylinder and Mars experiments are also provided. In following table the directory structure on the CD is described.

Directory/File	Description
<code>thesis_strunc.pdf</code>	the Master's Thesis report in pdf format
<code>src</code>	Matlab source codes
<code>src/data</code>	experiment data

Table 1: Directory structure on the CD