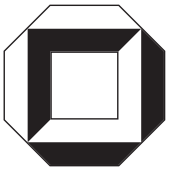


UNIVERSITÄT KARLSRUHE

Shape from Specular Reflection and Optical Flow

J. Lellmann
J. Balzer
A. Rieder
J. Beyerer

Preprint Nr. 07/02

Institut für Wissenschaftliches Rechnen
und Mathematische Modellbildung

76128 Karlsruhe

Anschriften der Verfasser:

Dipl.-Inform. Jan Lellmann
Institut für Technische Informatik
Lehrstuhl für Interaktive Echtzeitsysteme
Universität Karlsruhe (TH)
D-76128 Karlsruhe

Dipl.-Ing. Jonathan Balzer
Institut für Technische Informatik
Lehrstuhl für Interaktive Echtzeitsysteme
Universität Karlsruhe (TH)
D-76128 Karlsruhe

Prof. Dr. Andreas Rieder
Institut für Angewandte und Numerische Mathematik
Universität Karlsruhe (TH)
D-76128 Karlsruhe

Prof. Dr. Jürgen Beyerer
Institut für Technische Informatik
Lehrstuhl für Interaktive Echtzeitsysteme
Universität Karlsruhe (TH)
D-76128 Karlsruhe

Shape from Specular Reflection and Optical Flow

J.Lellmann[*], J.Balzer[*], A.Rieder[†] and J.Beyerer[*]

jan.lellmann@web.de, jonathan.balzer@ies.uni-karlsruhe.de,
andreas.rieder@math.uni-karlsruhe.de, juergen.beyerer@iitb.fraunhofer.de

March 15, 2007

Abstract

Inferring scene geometry from a sequence of camera images is one of the central problems in computer vision. While the overwhelming majority of related research focuses on diffuse surface models, there are cases when this is not a viable assumption: in many industrial applications, one has to deal with metal or coated surfaces exhibiting a strong specular behavior. We propose a novel and generalized constrained gradient descent method to determine the shape of a purely specular object from the reflection of a calibrated scene and additional data required to find a unique solution. This data is exemplarily provided by optical flow measurements obtained by small scale motion of the specular object. We present a general forward model to predict the optical flow of specular surfaces, covering rigid body motion as well as elastic deformation, and allowing a characterization of problematic points. We demonstrate the applicability of our method by numerical experiments.

Keywords: Specular surfaces, deflectometry, Shape from Shading, level sets, optical flow, constrained gradient descent.

1 Introduction and previous work

We consider the problem of reconstructing a free-form mirror surface by observing the reflected image of a calibrated scene. This method is known as Shape from Specular Reflection or deflectometry, see [Bon06] for a comprehensive introduction. Deflectometry is widely used because it allows to detect small irregularities of the surface under inspection. Real-world applications include high resolution scanning of optical lenses and industrial quality control of coated components [Kam04]. However, as with classical camera-based scene reconstruction, a single image generally does not suffice to fully determine the surface structure.

Current literature contains some suggestions how to overcome the ambiguity. Solem et al. investigate the problem in a variational setting, cf. [SAH04]. Assuming a set of surface points is known, they propose an iterative algorithm to minimize an energy functional consisting of surface normal and point constraints. Kickingereder et al. [KD04] and Bonfort [BS03] use stereo vision to uniquely recover the surface. Some approaches based on a local surface representation have also been discussed: Savarese, Chen, and Perona [SCP05] use a special pattern to find a set of surface patches by locally neglecting higher-order surface properties. In [BWB06], the Lambertian behavior of a certain class of surfaces is employed. The latter approach is closely connected to a research area known as Shape from Shading [Hor70, PCF].

[*]Lehrstuhl für Interaktive Echtzeitsysteme, Institut für Technische Informatik (ITEC), Department of Computer Science, University of Karlsruhe, 76128 Karlsruhe, Germany

[†]Institut für Angewandte und Numerische Mathematik, Department of Mathematics, University of Karlsruhe, 76128 Karlsruhe, Germany

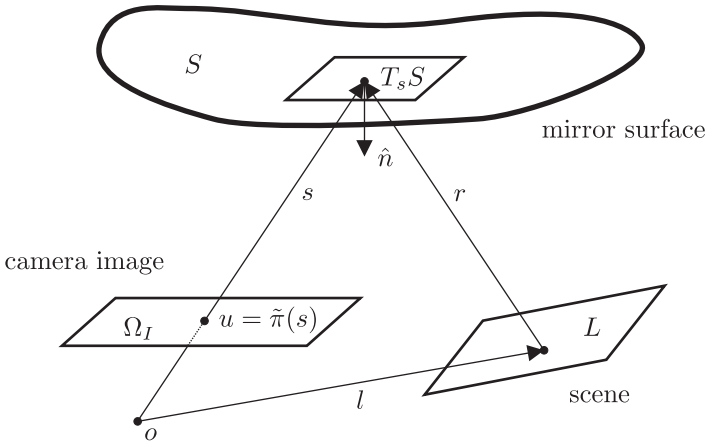


Figure 1: Basic setup. The scene L is reflected in the specular surface S and observed by the camera in the origin.

In an early predecessor of our work, Oren and Nayar [ON95] investigate an abstract setting where only the camera is moving. They do not rely on the classical monocular stereo approach but use a differential multiview method to derive a family of possible surface curves. To our knowledge, the first authors incorporating optical flow measurements were Roth and Black [RB06]. Their model requires a large distance between mirror and reflected scene, leaving room for a more realistic small-scale generalization. This will be one of our main contributions.

This paper is outlined as follows: in the following section, we will state the problem, provide some necessary background in level set theory, and finally show how to formulate the reconstruction problem as a constrained optimization problem. In Section 3, we will then derive the governing equations of specular flow. We will see that an flow vector field fails to be well-defined under certain circumstances. Our model provides for the characterization of such special points. In Section 4, we propose a gradient descent method to solve the general optimization problem of reconstructing the shape of an unknown specular surface in combination with optical flow measurements. Our numerical procedure is sketched and experimental results are presented to underline the practicability of our method.

2 Problem statement

2.1 Notation and basic problem

We frequently use the Euclidean scalar product $\langle \cdot, \cdot \rangle$ which induces the norm $\| \cdot \|$ on $\mathbb{R}^3$. Vectors are distinguished by bold letters $\boldsymbol{x} = \begin{pmatrix} x_1 & x_2 & x_3 \end{pmatrix}^\top$, matrices by capitalization $\mathbf{M}$. Normalization to unit length is denoted by an additional hat: $\hat{\boldsymbol{x}} = \pi(\boldsymbol{x})$, where $\pi : \mathbb{R}^3 \backslash \{0\} \to S^2, \boldsymbol{x} \mapsto \pi(\boldsymbol{x}) := \frac{\hat{\boldsymbol{x}}}{\|\hat{\boldsymbol{x}}\|}$, is the projection to the unit sphere S^2. The tangent map of π can be described by the Jacobian of its continuation on $\mathbb{R}^3$: $\nabla\pi(\boldsymbol{x}) = \frac{1}{\|\boldsymbol{x}\|}\mathbf{P}_{\boldsymbol{x}}$, where $\mathbf{P}_{\boldsymbol{x}} = \mathbf{I} - \hat{\boldsymbol{x}}\hat{\boldsymbol{x}}^\top$ denotes the orthogonal projection onto $\hat{\boldsymbol{x}}^\perp$ in matrix form.

Figure 1 illustrates the basic setup. We assume a single-viewpoint camera with optical center in origin $\boldsymbol{o}$ of the coordinate system. For simplicity, we will generally use a standard pinhole camera model with focal length $f = 1$. The pinhole camera is modelled by the function $\Pi : \Omega \to \Omega_I$, projecting each point $\boldsymbol{x}$ in a subset $\Omega \subseteq \mathbb{R}^3$ of the field of view[1] onto some $\boldsymbol{u}$ in the image plane $x_3 = 1$, i.e. $\boldsymbol{u} = \Pi(\boldsymbol{x}) = \frac{\boldsymbol{x}}{x_3}$.

Intersecting the ray emanating from the origin through an image point $\boldsymbol{u} \in \Omega_I$ with S, we get the *reflection point* $\boldsymbol{s}(\boldsymbol{u})$ on S. We generally require that the *surface map* $\boldsymbol{s}$ is well-

[1]The field of view is the set of all points $\boldsymbol{x} \in \mathbb{R}^3, x_3 > 0$, which can be connected to the origin by a straight line intersecting the image plane Ω_I.

defined on all of Ω_I, i.e. the surface occupies the whole image. If S was diffuse, the physical image intensity in $\boldsymbol{u}$ would be completely determined by the texture in $\boldsymbol{s}(\boldsymbol{u})$. But since S is specular, the ray is reflected and advances until it hits the scene surface L in $\boldsymbol{l}(\boldsymbol{u})$. We denote the reflected ray by $\boldsymbol{r} := \boldsymbol{s} - \boldsymbol{l}$. We also assume that there are no multiple reflections and the *light map* $\boldsymbol{l} : \Omega_I \to L$ is defined on all of Ω_I. We require S to be sufficiently smooth and Ω to be a bounded domain with piecewise smooth boundary $\partial\Omega$. Now, we can formulate the basic problem in an abstract way:

Problem 1. *Given the light map $\boldsymbol{l}(\boldsymbol{u})$, find a corresponding surface map $\boldsymbol{s}(\boldsymbol{u})$.*

This means for each image point $\boldsymbol{u}$ in the distorted image, we must be able to estimate the exact scene point $\boldsymbol{l}(\boldsymbol{u})$ *without knowing* S. In practice, this is realized with the help of time-coded radiance patterns in the scene, see [Kam04] for a discussion. Here, we assume this has already been accomplished.

The law of reflection connects measurement and surface: for $\boldsymbol{s} = \boldsymbol{s}(\boldsymbol{u}), \boldsymbol{l} = \boldsymbol{l}(\boldsymbol{u})$, and the normal $\hat{\boldsymbol{n}}$ of S in $\boldsymbol{s}$, we have

$$\hat{\boldsymbol{n}}(\boldsymbol{s}) = -\pi(\pi(\boldsymbol{s} - \boldsymbol{o}) + \pi(\boldsymbol{s} - \boldsymbol{l})) = -\frac{\hat{\boldsymbol{s}} + \hat{\boldsymbol{r}}}{\|\hat{\boldsymbol{s}} + \hat{\boldsymbol{r}}\|}. \tag{2.1}$$

The normals are forced downward, i.e. $\langle \hat{\boldsymbol{n}}, \boldsymbol{s} \rangle < 0$. Thus, if we set

$$\hat{\boldsymbol{m}}(\boldsymbol{x}) = -\pi(\hat{\boldsymbol{x}} + \pi(\boldsymbol{x} - \boldsymbol{l}(\Pi(\boldsymbol{x})))), \tag{2.2}$$

$\boldsymbol{x} \in \Omega$, we have a vector field $\hat{\boldsymbol{m}}$ restricting the normals of S: if $\boldsymbol{s} \in S$, then

$$\hat{\boldsymbol{n}}(\boldsymbol{s}) = \hat{\boldsymbol{m}}(\boldsymbol{s}). \tag{2.3}$$

If this condition holds for all $\boldsymbol{s}$, the surface is said to satisfy the *normal condition*. This leads to an equivalent formulation:

Problem 2. *Given the normal field $\hat{\boldsymbol{m}}$, find a surface map $\boldsymbol{s}$ satisfying $\hat{\boldsymbol{n}}(\boldsymbol{s}) = \hat{\boldsymbol{m}}(\boldsymbol{s})$.*

Despite its elegance, this way of viewing the problem has not been widely considered. Notable exceptions are the works of Bonfort and Sturm [BS03] as well as Hicks and Perline [HP04], who also provide a generalized formulation using so-called m-distributions.

The problem in its current form does not necessarily admit a unique solution. In [KD04] for example, an adept parametrization transforms (2.3) into a total differential equation exhibiting an infinite number of solutions if no suitable boundary conditions are given.

2.2 Optical flow

The objective of this paper is to incorporate *optical flow* measurements to overcome the ambiguity addressed in the last section. Consider a scene surface containing a fixed point $\boldsymbol{x}$ which is observed at image coordinates $\boldsymbol{u}(t)$ for $t \in T$ with T a time interval. The optical flow for $\boldsymbol{u} = \boldsymbol{u}(t_0)$ is then defined as the temporal derivative $\dot{\boldsymbol{u}}(t_0)$. Its prominent advantage is that it can be estimated solely from the image sequence if the scene provides distinctive texture and the temporal resolution is sufficiently high [BFB94]. Being a local property, the optical flow effectively adds information to a *single* image, which allows to pointwise infer scene information in a much easier way than with genuine stereo [Hor86, ZGB89].

Consider a fixed camera observing a dynamical scene, and let $\boldsymbol{w}(\boldsymbol{x})$ denote the velocity vector for each scene point $\boldsymbol{x}$ at time t_0. Note that this includes uniform Euclidean motion, where

$$\boldsymbol{w}(\boldsymbol{x}) = \dot{\boldsymbol{x}}(t_0) = \boldsymbol{\Omega}\boldsymbol{x} + \boldsymbol{v}, \tag{2.4}$$

and $\boldsymbol{\Omega} = \boldsymbol{\Omega}(t_0)$ is a skew-symmetric matrix s.th. $\boldsymbol{x}(t) = \mathbf{R}(t)\boldsymbol{x} + \boldsymbol{y}(t)$, $\mathbf{R}(t) = \mathrm{e}^{\boldsymbol{\Omega}(t)} \in \mathrm{SO}(3)$ and $\boldsymbol{v} = \dot{\boldsymbol{y}}(t_0)$. However, this *velocity field* formulation allows far more complex and possibly non-rigid motion, e.g. waves on a water surface. For a point $\boldsymbol{s}$ on a *diffuse* surface S, the

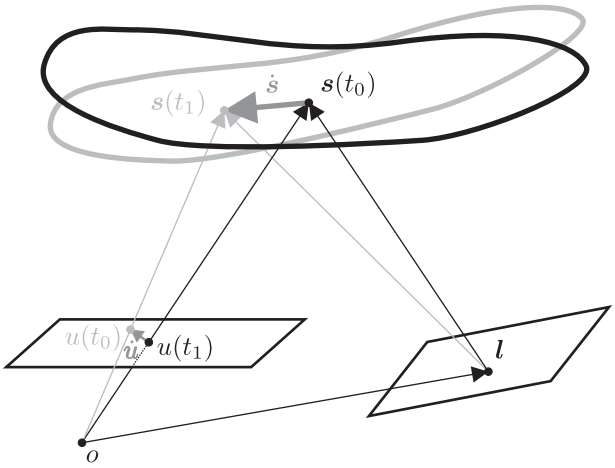


Figure 2: Specular optical flow. The reflection point $\boldsymbol{s}$ varies as the specular surface moves relative to camera and diffuse scene.

optical flow is given by [Hor86]

$$\begin{aligned}\dot{\boldsymbol{u}} &= \frac{d}{dt}\Pi(\boldsymbol{s}(t)) = \nabla\Pi(\boldsymbol{s})\dot{\boldsymbol{s}} \\ &= \nabla\Pi(\boldsymbol{s})\boldsymbol{w}(\boldsymbol{s}) = \frac{1}{s_3}\begin{pmatrix} 1 & 0 & -u_1 \\ 0 & 1 & -u_2 \\ 0 & 0 & 0 \end{pmatrix}\boldsymbol{w}(\boldsymbol{s}).\end{aligned} \tag{2.5}$$

In the simple case of purely translational motion, i.e. $\boldsymbol{\Omega} = 0$, we have $\boldsymbol{w}(\boldsymbol{s}) \equiv \boldsymbol{u}$ and can easily deduce s_3 – and thus $\boldsymbol{s}$ – from known $\boldsymbol{u}$, $\boldsymbol{v}$, and $\dot{\boldsymbol{u}}$ (see [HZ03] for a survey). If S is *specular*, the process is far more involved since $\boldsymbol{u}$ is now the projection of the *reflection point* $\boldsymbol{s}$ of a fixed scene point $\boldsymbol{l}$. Here, we will consider the case where the reflected scene L remains stationary, i.e. camera and diffuse target are rigidly mounted on a sensor head which moves relatively to the unknown specular object. Generally, $\boldsymbol{s}$ will move on the surface in time: one does not always "see" the same *surface* point although the observed *scene* point does not change (see Figure 2).

Specularities have been widely regarded as a source of error in diffuse surface reconstruction, with many authors proposing methods to detect and discard the respective flow vectors. To take advantage of the extra information, Roth and Black [RB06] recently suggested a model based on a first-order Taylor expansion of the path function introduced by Chen and Arvo [CA00]. Their model requires a large distance between $\boldsymbol{s}$ and $\boldsymbol{l}$, leaving the reflected ray $\boldsymbol{r}$ in (2.1) approximately constant. For our setup, this assumption is too restrictive. We propose a remedy in Section 4.2.

2.3 Level sets and differential geometry

The surfaces under concern are assumed to be regular for the rest of the paper, i.e. they can be described by a set $S \subset \mathbb{R}^3$ and an atlas of C^k-diffeomorphisms $h_{\boldsymbol{s}}$, $k \geq 1$, such that for every point $\boldsymbol{s} \in S$, there exists a neighborhood $\tilde{U} \subset \mathbb{R}^3$ of $\boldsymbol{s}$ with $h_{\boldsymbol{s}}$ mapping $\tilde{U} \cap S$ to an open set $U \subset \mathbb{R}^2$ [dC76]. Some inherent drawbacks of this representation, namely the possible need for reparametrization and the strong dependence on the dimension of the underlying space, can be circumvented by using an implicit – or *level set* – representation made popular by Osher [OF02] and Sethian [Set05]. Here, we express S as the zero set of a function $\varphi \in C^2(V, \mathbb{R})$ s.th. $S \cap V = \varphi^{-1}(\{0\})$. The level set function φ defines a regular surface if $\nabla\varphi \neq 0$ on $\varphi^{-1}(\{0\})$, which we assume throughout.

While a regular surface locally allows such a representation (for any local graph representation $f : U \subset \mathbb{R}^2 \to \Omega$, set $\varphi(x, y, z) = f(x, y) - z$), no global level set function $\varphi \in C^2(\Omega, \mathbb{R})$

exists in general. The continuity of φ enforces orientability, which is why one only allows regular surfaces of the form $S = \partial W$ with some open $W \subset \mathbb{R}^3$, s.th. $\varphi < 0$ in W and $\varphi > 0$ in $\Omega \backslash W$. We will assume S to satisfy the above requirements.

The following inner geometric quantities are needed frequently:

$$\hat{\boldsymbol{n}} = \frac{\nabla\varphi}{\|\nabla\varphi\|}, \quad \mathbf{K} = -\frac{1}{\|\nabla\varphi\|}\mathbf{P}_{\hat{\boldsymbol{n}}}\nabla^2\varphi\mathbf{P}_{\hat{\boldsymbol{n}}},$$

where $\hat{\boldsymbol{n}}$ is the normal in the direction of $\varphi > 0$ and $\mathbf{K}$ is the symmetric matrix defining the second fundamental form satisfying $\mathbf{K}\hat{\boldsymbol{n}} = 0$. By $\nabla^2\varphi$, we denote the Hessian of φ. For the mean curvature κ, we get $\kappa = -\operatorname{Tr}\mathbf{K}$ (see [Jin03]).

The *surface gradient* $D_S f(\boldsymbol{s})$ of a function $f : S \to \mathbb{R}$ is defined as the unique vector $\boldsymbol{y} \in T_{\boldsymbol{s}}S$ with $\frac{d}{d\tau}\{f(\boldsymbol{x}(\tau))\}|_{\tau=0} = \langle \boldsymbol{y}, \frac{d}{d\tau}\boldsymbol{x}(\tau)|_{\tau=0}\rangle$ for all curves $\boldsymbol{x} \subseteq S$ with $\boldsymbol{x}(0) = \boldsymbol{s}$. In case one has access to the gradient ∇f of a continuation of f on a neighborhood of $\boldsymbol{s}$, we have $D_S f(\boldsymbol{s}) = \mathbf{P}_{\hat{\boldsymbol{n}}}\nabla f(\boldsymbol{s})$.

Moving surfaces are naturally written as $S(t) = \varphi^{-1}[\{0\}, t]$, where $\varphi \in C^2(\Omega \times T, \mathbb{R})$, and each $\varphi[\cdot, t]$ itself is a level set function. We will use brackets to distinguish these level set *evolutions* from ordinary level set functions φ. For such an evolution, let $\nabla\varphi := \partial_x\varphi$ and $\dot{\varphi} := \partial_t\varphi$.

Consider a differentiable curve $\boldsymbol{s} : T \to \Omega$ lying on the moving surface, i.e. $\varphi[\boldsymbol{s}(t), t] \equiv 0$. Derivation w.r.t. t results in

$$\dot{\varphi}[\boldsymbol{s}(t), t] = -\nabla\varphi[\boldsymbol{s}(t), t]\dot{\boldsymbol{s}}(t) = -\langle \dot{\boldsymbol{s}}(t), \hat{\boldsymbol{n}}[\boldsymbol{s}(t), t]\rangle\|\nabla\varphi[\boldsymbol{s}(t), t]\|.$$

This means that the way the surface evolves only depends on the *normal velocity* field

$$v := \langle \dot{\boldsymbol{s}}, \hat{\boldsymbol{n}}\rangle \tag{2.6}$$

of the parametrization [SO05b]. By substituting v, we get the *level set equation*

$$\dot{\varphi} = -v\|\nabla\varphi\|. \tag{2.7}$$

As an important example, consider the special level set function

$$\varphi(\boldsymbol{x}) = e_{\boldsymbol{l},\boldsymbol{x}_0}(\boldsymbol{x}) := c(\boldsymbol{x}_0) - (\|\boldsymbol{x}\| + \|\boldsymbol{x} - \boldsymbol{l}\|) \tag{2.8}$$

with $\boldsymbol{s}, \boldsymbol{l} \in \mathbb{R}^3$ and $c = c(\boldsymbol{x}_0) := \|\boldsymbol{x}_0\| + \|\boldsymbol{x}_0 - \boldsymbol{l}\|$. The zero set $E_{\boldsymbol{l},\boldsymbol{x}_0}$ of $e_{\boldsymbol{l},\boldsymbol{x}_0}$ consists of all points $\boldsymbol{x}$ for which the sum of the distances to $\boldsymbol{o}$ and $\boldsymbol{l}$ is the same as for $\boldsymbol{x}_0$. In the two-dimensional case, it coincides with an ellipse with focal points $\boldsymbol{o}$ and $\boldsymbol{l}$. In three dimensions, a *prolate spheroid* $E_{\boldsymbol{l},\boldsymbol{x}_0}$ is generated by rotating such an ellipse around its major axis through $\boldsymbol{o}$ and $\boldsymbol{l}$.

For $\boldsymbol{x} \in E_{\boldsymbol{l},\boldsymbol{x}_0}$, the normal computes as

$$\hat{\boldsymbol{n}}_{\boldsymbol{l},\boldsymbol{x}_0}(\boldsymbol{x}) = \pi(\nabla e_{\boldsymbol{l},\boldsymbol{x}_0}(\boldsymbol{x})) = -\pi\left(\hat{\boldsymbol{x}} - \pi(\boldsymbol{x} - \boldsymbol{l})\right).$$

Comparing this to (2.1) confirms the fact that such a spheroid will reflect *all* rays emerging from the origin into its second focal point $\boldsymbol{l}$ and vice versa. The law of reflection then reduces to the fact that the surface normal $\hat{\boldsymbol{n}}(\boldsymbol{s})$ for a point $\boldsymbol{s} \in S$ and the normal $\hat{\boldsymbol{n}}_{\boldsymbol{l},\boldsymbol{s}}(\boldsymbol{s})$ of the (unique) spheroid through $\boldsymbol{s}$ with foci $\boldsymbol{o}$ and $\boldsymbol{l}$ coincide. In the case that S is exactly shaped like $E_{\boldsymbol{l},\boldsymbol{s}}$ in a neighborhood of $\boldsymbol{s}$ (and assuming there are no occlusions), the scene point $\boldsymbol{l}$ will be visible in a neighborhood of the image point $\boldsymbol{u} = \Pi(\boldsymbol{s})$. This means we cannot generally assume the injectivity of $\boldsymbol{l}^{-1} : L \to \Omega_I$, not even locally: features on L might be "infinitely" magnified when observed indirectly through the mirror.

2.4 Stating the problem

We will now see how the extra information provided by the optical flow can be integrated into the model to help finding the original surface. First, observe that solving (2.3) for S can be expressed as minimizing the surface functional

$$J(S) := \frac{1}{2}\int_S \|\hat{\boldsymbol{n}} - \hat{\boldsymbol{m}}\|^2 d\sigma,$$

where $d\sigma$ is a suitable surface measure. In level set notation, this can formally be written as

$$J(\varphi) = \frac{1}{2} \int_{\Omega} \|\hat{\boldsymbol{n}} - \hat{\boldsymbol{m}}\|^2 \|\nabla\varphi\| \delta(\varphi(\boldsymbol{x})) d\boldsymbol{x},$$

where δ is the Dirac delta distribution. Our extra information shall be contained in a similar functional H. While we will illustrate the procedure only for the optical flow case, H might as well include any additional measurements, e.g. from range sensors or analysis of diffuse surface parts. Here, we will use

$$H(S) = \frac{1}{2} \int_{S} \|\dot{\boldsymbol{u}} - \dot{\boldsymbol{u}}_m\| d\sigma, \tag{2.9}$$

where $\dot{\boldsymbol{u}}$ is the expected optical flow for the surface hypothesis S and $\dot{\boldsymbol{u}}_m(\boldsymbol{s})$ is the observed flow field at image coordinates $\Pi(\boldsymbol{s})$, interpolated from measurements. In practice, problems may arise if optical flow data are only partially available, a case we will ignore for the scope of this paper.

Traditionally (cf. [SAH04, KD04]), one would blend both functionals into a third one, e.g. $E(S) = \alpha J(S) + (1-\alpha)H(S)$, and then minimize E. We will adopt a different viewpoint inspired by [SO05a]: let $\mathcal{R}$ be the set of regular surfaces (or even level set surfaces if necessary), and denote by $\mathcal{R}_{\hat{\boldsymbol{m}}} \subset \mathcal{R}$ the subset of all surfaces satisfying the normal condition, i.e. $S \in \mathcal{R}_{\hat{\boldsymbol{m}}} \Leftrightarrow J(S) = 0$. We may now solve the *constrained minimization problem*

$$S = \operatorname*{argmin}_{S \in \mathcal{R}_{\hat{\boldsymbol{m}}}} H(S).$$

This concept provides some benefits: $\mathcal{R}_{\hat{\boldsymbol{m}}}$ has fewer degrees of freedom than $\mathcal{R}$. In fact, it can be shown that $\mathcal{R}_{\hat{\boldsymbol{m}}}$ is a one-parameter family, parametrizeable by the intersection of S with a ray through the origin. On the other hand, one can assume $S \in \mathcal{R}_{\hat{\boldsymbol{m}}}$ for the construction of H. This proves useful in the specular flow case, where one has to assume the law of reflection holds in order to correctly calculate the flow vectors. Before we will see how to realize the minimization process, we will derive an expression for the expected optical flow $\dot{\boldsymbol{u}}$.

3 Optical flow on specular surfaces

3.1 Calculating optical flow

We now solve the forward problem: computing the optical flow $\dot{\boldsymbol{u}}$ for a known surface S moving according to the velocity field $\boldsymbol{w}$. In Section 2.2, we saw that instead, we may calculate the reflection point velocity $\dot{\boldsymbol{s}}$ like in (2.5). First, we need the following lemma to determine how a given normal velocity relates to changes in the normal field.

Lemma 1. *Let φ be a sufficiently differentiable level set evolution and $\boldsymbol{s} : T \to \mathbb{R}^3$ a differentiable curve on $S(t) := \varphi^{-1}[\{0\}, t]$, i.e. $\varphi[\boldsymbol{s}(t), t] \equiv 0$. Then,*

$$\frac{d}{dt}\{\hat{\boldsymbol{n}}[\boldsymbol{s}(t), t]\} = -D_S v + d_{\mathbf{P}_{\hat{\boldsymbol{n}}}\dot{\boldsymbol{s}}}\hat{\boldsymbol{n}},$$

where $D_S v$ is the surface gradient of the normal velocity (2.6) *and $d_{\mathbf{P}_{\hat{\boldsymbol{n}}}\dot{\boldsymbol{s}}}\hat{\boldsymbol{n}} \in T_{\boldsymbol{s}}S$ the derivative of $\hat{\boldsymbol{n}}$ in the direction of the projection $\mathbf{P}_{\hat{\boldsymbol{n}}}\dot{\boldsymbol{s}}$.*

Proof. From the implicit representation, we get a natural extension of $\hat{\boldsymbol{n}}$ and v on a small neighborhood of a point $\boldsymbol{s}(t_0)$. Thus by the chain rule we may write

$$\partial_t\{\hat{\boldsymbol{n}}[\boldsymbol{s}(t), t]\} = \nabla\hat{\boldsymbol{n}}[\boldsymbol{s}(t), t]\dot{\boldsymbol{s}}(t) + \partial_t\hat{\boldsymbol{n}}[\boldsymbol{s}(t), t].$$

Decomposing $\dot{\boldsymbol{s}}$ orthogonally into $\dot{\boldsymbol{s}} = \mathbf{P}_{\hat{\boldsymbol{n}}}\dot{\boldsymbol{s}} + \langle\dot{\boldsymbol{s}}, \hat{\boldsymbol{n}}\rangle\hat{\boldsymbol{n}} = \mathbf{P}_{\hat{\boldsymbol{n}}}\dot{\boldsymbol{s}} + v\hat{\boldsymbol{n}}$, we get (in short-hand notation)

$$\partial_t\hat{\boldsymbol{n}}[\boldsymbol{s}(t), t] = d_{\mathbf{P}_{\hat{\boldsymbol{n}}}\dot{\boldsymbol{s}}}\hat{\boldsymbol{n}} + (\nabla\hat{\boldsymbol{n}})v\hat{\boldsymbol{n}} + \partial_t\hat{\boldsymbol{n}}. \tag{3.1}$$

By $\partial_t \hat{\boldsymbol{n}} = \nabla\pi\partial_t\nabla\varphi$, changing the order of differentiation, and substituting $\dot{\varphi} = -v\|\nabla\varphi\|$ from the level set equation (2.7), it follows that

$$\partial_t \hat{\boldsymbol{n}} = \partial_t\{\pi(\nabla\varphi)\} = \nabla\pi\nabla\dot{\varphi} = \nabla\pi\nabla\{-v\|\nabla\varphi\|\}.$$

By the product rule, $\nabla\{v\|\nabla\varphi\|\} = v\nabla\|\nabla\varphi\| + \|\nabla\varphi\|\nabla v$. With $\nabla\|\boldsymbol{x}\| = \hat{\boldsymbol{x}}$, it follows that

$$\nabla\pi\nabla\{v\|\nabla\varphi\|\} = \frac{1}{\|\nabla\varphi\|}\mathbf{P}_{\hat{\boldsymbol{n}}}\left(v\nabla^2\varphi\frac{\nabla\varphi}{\|\nabla\varphi\|} + \|\nabla\varphi\|\nabla v\right).$$

Now, we use $\frac{1}{\|\nabla\varphi\|}\mathbf{P}_{\hat{\boldsymbol{n}}}\nabla^2\varphi = \nabla\hat{\boldsymbol{n}}$, and thus

$$\partial_t\hat{\boldsymbol{n}} = -(\nabla\hat{\boldsymbol{n}})v\hat{\boldsymbol{n}} - \mathbf{P}_{\hat{\boldsymbol{n}}}\nabla v$$

which yields the assertion in view of (3.1).

□

The central result of this section is

Theorem 2. *Let φ be a level set evolution and $\boldsymbol{s}$ a curve as in Lemma 1 which is additionally a* reflection point curve*: all points on $\boldsymbol{s}$ satisfy the law of reflection* (2.3) *with respect to the origin and a fixed $\boldsymbol{l} \in \mathbb{R}^3$. Then, $\dot{\boldsymbol{s}}$ solves the linear system*

$$(\hat{\boldsymbol{n}}\hat{\boldsymbol{n}}^\top + \mathbf{M} - \mathbf{K})\dot{\boldsymbol{s}} = (\mathbf{I} + \nabla\hat{\boldsymbol{m}})(v\hat{\boldsymbol{n}}) + D_S v.$$

Here, $\mathbf{M} = -\nabla\hat{\boldsymbol{m}} = -\frac{1}{\|\nabla e_{\boldsymbol{l},\boldsymbol{x}}\|}\mathbf{P}_{\hat{\boldsymbol{n}}}\nabla^2 e_{\boldsymbol{l},\boldsymbol{s}}\mathbf{P}_{\hat{\boldsymbol{n}}}$ with $e_{\boldsymbol{l},\boldsymbol{s}}$ from Equation (2.8).

Proof. By definition, $\boldsymbol{s}$ satisfies two constraints:

1. $\boldsymbol{s}(t) \in S(t)$ for all $t \in T$ and, by the law of reflection,
2. $\hat{\boldsymbol{n}}[\boldsymbol{s}(t), t] = \hat{\boldsymbol{m}}(\boldsymbol{s}(t))$.

As shown in Section 2.3, the first of the above properties immediately implies

$$\langle\dot{\boldsymbol{s}}, \hat{\boldsymbol{n}}\rangle = v. \tag{3.2}$$

Now by Lemma 1,

$$\frac{d}{dt}\{\hat{\boldsymbol{n}}(\boldsymbol{s}(t), t)\} = -D_S v + d_{\mathbf{P}_{\hat{\boldsymbol{n}}}\dot{\boldsymbol{s}}}\hat{\boldsymbol{n}}.$$

But the second constraint forces $\hat{\boldsymbol{n}}(\boldsymbol{s}(t), t) = \hat{\boldsymbol{m}}(\boldsymbol{s}(t))$, thus

$$-\nabla\hat{\boldsymbol{m}}\dot{\boldsymbol{s}} = D_S v - d_{\mathbf{P}_{\hat{\boldsymbol{n}}}\dot{\boldsymbol{s}}}\hat{\boldsymbol{n}}.$$

As $d_{\mathbf{P}_{\hat{\boldsymbol{n}}}\dot{\boldsymbol{s}}}\hat{\boldsymbol{n}} = -\mathbf{K}\mathbf{P}_{\hat{\boldsymbol{n}}}\dot{\boldsymbol{s}} = -\mathbf{K}\dot{\boldsymbol{s}}$, we have

$$(-\nabla\hat{\boldsymbol{m}} - \mathbf{K})\dot{\boldsymbol{s}} = D_S v.$$

Again, decomposing $\dot{\boldsymbol{s}} = \mathbf{P}_{\hat{\boldsymbol{n}}}\dot{\boldsymbol{s}} + \hat{\boldsymbol{n}}\hat{\boldsymbol{n}}^\top\dot{\boldsymbol{s}} = \mathbf{P}_{\hat{\boldsymbol{n}}}\dot{\boldsymbol{s}} + v\hat{\boldsymbol{n}}$, it follows that

$$(-\nabla\hat{\boldsymbol{m}}\mathbf{P}_{\hat{\boldsymbol{n}}} - \mathbf{K})\dot{\boldsymbol{s}} = \nabla\hat{\boldsymbol{m}}(v\hat{\boldsymbol{n}}) + D_S v.$$

For $\mathbf{M}$ defined as above, we have $-\nabla\hat{\boldsymbol{m}}\mathbf{P}_{\hat{\boldsymbol{n}}} = \mathbf{M}$, and thus

$$(\mathbf{M} - \mathbf{K})\dot{\boldsymbol{s}} = \nabla\hat{\boldsymbol{m}}(v\hat{\boldsymbol{n}}) + D_S v. \tag{3.3}$$

Equations (3.2) and (3.3) form a linear system in $\dot{\boldsymbol{s}}$. Moreover, since $\mathrm{Im}(\mathbf{M} - \mathbf{K}) \subseteq T_{\boldsymbol{s}}S$ and $\nabla\hat{\boldsymbol{m}}(v\hat{\boldsymbol{n}}), D_S v \in T_{\boldsymbol{s}}S$, we may multiply (3.2) by $\hat{\boldsymbol{n}}$ and add both equations, thereby transforming the system into the aggregate form

$$(\hat{\boldsymbol{n}}\hat{\boldsymbol{n}}^\top + \mathbf{M} - \mathbf{K})\dot{\boldsymbol{s}} = (\mathbf{I} + \nabla\hat{\boldsymbol{m}})(v\hat{\boldsymbol{n}}) + D_S v. \tag{3.4}$$

□

Assuming the matrix on the left side of (3.4) is invertible, Theorem 2 allows us to compute, for each point $\boldsymbol{s}$ on the surface and the corresponding *fixed* scene point $\boldsymbol{l}$, the velocity vector $\dot{\boldsymbol{s}}$ of the reflection point on the surface. Projecting this into the image plane finally gives – similar to (2.5) – the optical flow vector

$$\dot{\boldsymbol{u}} = \frac{1}{\langle \boldsymbol{s}, \boldsymbol{e}_3 \rangle} (\mathbf{I} - \boldsymbol{u}\boldsymbol{e}_3^\top) \dot{\boldsymbol{s}}$$

with $\mathbf{I}$ the identity matrix and $\boldsymbol{e}_3$ the third standard basis vector. A striking consequence of the theorem is that the optical flow is only influenced by surface properties up to second order. This is in close analogy to a result of Savarese and Perona [SCP05] for a related problem.

Note that the optical flow computation only makes sense if one can presume that the law of reflection holds. This is one of the reasons why our constrained minimization approach is superior: we only need to evaluate flow vectors for surfaces $S \in \mathcal{R}_{\hat{\boldsymbol{m}}}$, which by construction of $\mathcal{R}_{\hat{\boldsymbol{m}}}$ satisfy the law of reflection in every point.

Also note that our model allows arbitrary surface motion, as it depends only on the normal velocity field. In practice, we may want to restrict ourselves to rigid body motion:

Corollary 3. *Under the conditions of Theorem 2, let S move relatively to the camera according to an external velocity field $\boldsymbol{w}$ as in Section 2.2. Then $\dot{\boldsymbol{s}}$ satisfies*

$$(\hat{\boldsymbol{n}}\hat{\boldsymbol{n}}^\top + \mathbf{M} - \mathbf{K})(\dot{\boldsymbol{s}} - \boldsymbol{w}) = \nabla\hat{\boldsymbol{m}}\boldsymbol{w} + \mathbf{P}_{\hat{\boldsymbol{n}}}(\nabla\boldsymbol{w})^\top\hat{\boldsymbol{n}}.$$

Proof. As the motion is externally controlled, we may consider the curve that a point $\boldsymbol{x} \in S(t_0)$ follows over time. Let $\boldsymbol{c}$ be such a curve, so $\boldsymbol{c}(t_0) = \boldsymbol{x}$. The external velocity field then dictates $\dot{\boldsymbol{c}}(t_0) = \boldsymbol{w}(\boldsymbol{x})$. Since $\boldsymbol{c}$ tracks one fixed point on S, $\boldsymbol{c}$ is a surface curve, i.e. $\boldsymbol{c}(t) \in S(t)$. This in turn forces the normal velocity in $\boldsymbol{x}$ to be $v = \langle \dot{\boldsymbol{c}}(t_0), \hat{\boldsymbol{n}} \rangle = \langle \boldsymbol{w}, \hat{\boldsymbol{n}} \rangle$. Substituting this in the result of Theorem 2, applying the product rule, and using $\mathbf{M} = -\nabla\hat{\boldsymbol{m}}\mathbf{P}_{\hat{\boldsymbol{m}}}$ yields the assertion. □

In particular, for an Euclidean rigid body motion we have $\boldsymbol{w} = \boldsymbol{\Omega}\boldsymbol{x} + \boldsymbol{v}$, so $\nabla\boldsymbol{w} = \boldsymbol{\Omega}$. Also note that by assumption $\hat{\boldsymbol{n}} = \hat{\boldsymbol{m}}$ on S, so any occurrences of $\hat{\boldsymbol{n}}$ may be replaced by the known $\hat{\boldsymbol{m}}$. If the matrix $(\hat{\boldsymbol{n}}\hat{\boldsymbol{n}}^\top + \mathbf{M} - \mathbf{K})$ is regular, $\dot{\boldsymbol{s}}$ is uniquely determined. By definition, we have $\hat{\boldsymbol{n}} \in \mathrm{Ker}(\mathbf{M} - \mathbf{K})$, so this is the case iff $\mathrm{Rank}(\mathbf{M} - \mathbf{K}) = 2$. If this rank condition is satisfied, we will call $\boldsymbol{s}$ a *flow regular* point, otherwise *flow singular*.

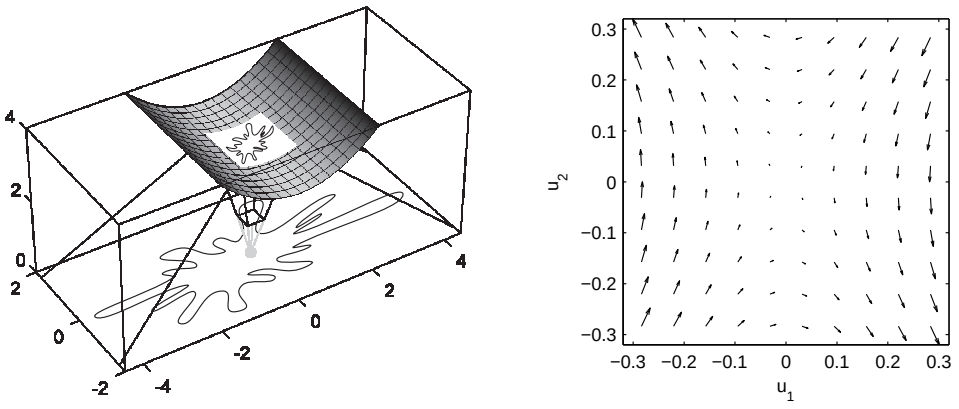
For illustration, consider a spheroid $E_{\boldsymbol{l}_0,\boldsymbol{s}}$ with $\boldsymbol{l}_0 := \boldsymbol{l}(\Pi(\boldsymbol{s})) \in L$ as defined by (2.8). Each point on E reflects $\boldsymbol{l}_0$ into the origin, meaning that the light map $\boldsymbol{l} : \Omega_I \to L$ cannot be inverted locally. When calculating optical flow, we are essentially tracking the projection $\boldsymbol{l}^{-1}(\boldsymbol{l}_0)$ of the *diffuse scene feature* at $\boldsymbol{l}_0$. This must fail if $\boldsymbol{l}$ is not at least locally invertible in $\boldsymbol{l}_0$.

So it comes out naturally that the optical flow vector is not well-defined in points $\boldsymbol{u} = \Pi(\boldsymbol{s})$ for surfaces that look like the spheroid $E_{\boldsymbol{l}_0,\boldsymbol{s}}$ in a neighborhood of $\boldsymbol{s}$. This is a perfectly reasonable constraint since such a surface would locally exhibit "infinite" magnification. But the rank constraint is actually weaker: it also forbids reflection points $\boldsymbol{s}$ where the second order properties of S and $E_{\boldsymbol{l}_0,\boldsymbol{s}}$ match in a single tangential direction $\boldsymbol{y} \in T_{\boldsymbol{s}}S$. This accounts for the fact that the image of $\boldsymbol{l}_0$ may split and merge, or even abruptly disappear from the camera image. We conclude that flow singular points are not only a byproduct of our model but have a real physical counterpart.

Figure 3 shows an exemplary optical flow for a stretched paraboloid rotating around the principal axis of the sensor as well as the "raw" reflection point velocities $\dot{\boldsymbol{s}}$. The result differs drastically from its counterpart for diffuse surfaces, which is rotation symmetric and – in this special case – even independent of the surface shape.

4 Reconstruction of specular surfaces

Having derived the intrinsic properties of real-world specular reflection, we shall now present a numerical method to solve the constrained minimization problem. We will use a gradient



(a) (b)

Figure 3: (a) Exemplary vector field of reflection point motion $\dot{\boldsymbol{s}}$ for a paraboloid rotating around the z axis and (b) corresponding optical flow field $\dot{\boldsymbol{u}}$.

descent method for the functional H carrying the "extra" information together with a gradient projection step to force $S \in \mathcal{R}_{\hat{\boldsymbol{m}}}$.

We face three problems: finding a starting point $S \in \mathcal{R}_{\hat{\boldsymbol{m}}}$, the gradient of H, and finally, the subspace on which to project. We will look for a starting point by unconstrained minimization of J as presented in the following section. For an outline of the basic concepts in abstract form, refer to [SO05b]. The derivation of such gradient descents has also been extensively discussed by the active contours community [ABFJB03, JBHBA06].

4.1 Unconstrained minimization

Consider the minimization problem $S = \operatorname{argmin}_{S \in \mathcal{R}} J(S)$. In level set notation, J is of the form

$$J(\varphi) = \int_\Omega (1 - \langle \hat{\boldsymbol{m}}, \hat{\boldsymbol{n}} \rangle) \|\nabla \varphi\| \delta(\varphi) dx. \tag{4.1}$$

Note $\hat{\boldsymbol{m}}$ and $\hat{\boldsymbol{n}}$ are both normalized so that $\frac{1}{2}\|\hat{\boldsymbol{n}} - \hat{\boldsymbol{m}}\|^2 = 1 - \langle \hat{\boldsymbol{m}}, \hat{\boldsymbol{n}} \rangle$. In order to minimize J, we seek a critical point: find φ^* s.th. $dJ(\varphi^*)\eta := \frac{d}{dt}\{J(\varphi^* + t\eta)\}|_{t=0} = 0$ for all test functions $\eta \in C^\infty(\Omega)$. Formally differentiating $J(\varphi^* + t\psi)$ w.r.t. t shows this is equivalent to

$$\operatorname{div}(\hat{\boldsymbol{n}}^* - \hat{\boldsymbol{m}}) \delta(\varphi^*) = 0, \tag{4.2}$$

where div is the divergence operator. As will be shown in the next section, this induces a time-dependent process:

$$\dot{\varphi} = \operatorname{div}(\hat{\boldsymbol{n}} - \hat{\boldsymbol{m}}) \|\nabla \varphi\|. \tag{4.3}$$

Any steady state solution of this equation satisfies (4.2) and is thus a stationary point of J. Using the level set equation (2.7), one can formulate the result in terms of normal velocities:

$$v = -\operatorname{div}(\hat{\boldsymbol{n}} - \hat{\boldsymbol{m}}). \tag{4.4}$$

This proves to be a very useful tool for deriving properties of evolving surfaces. Under the premise that we can find a corresponding level set evolution φ, we may use the level set representation as a tool to simplify calculations. Afterwards, results are converted back to a representation-independent form – which must be possible if the surface evolution is well defined, i.e. only depends on the *zero set* set of the level set function. In the following sections we will take advantage of this approach.

4.2 Deriving natural boundary conditions

In current vision related level set literature, it is common practice [GM04, SO05b] to postulate that S is closed and fully contained in the computational domain Ω, i.e. $S \cap \partial\Omega = \emptyset$. The surface is then represented by a discretization of the actual level set function, and standard Neumann boundary conditions $\frac{d\varphi}{d\hat{\boldsymbol{o}}} = 0$ are applied ($\hat{\boldsymbol{o}}$ refers to the outside normal of $\partial\Omega$). Using these homogeneous boundary conditions is justified, provided that S has a sufficiently large distance to $\partial\Omega$.

However, this case requires normal measurements for every point in Ω – even outside the actual surface – which in practice are rarely available. In our setup, the camera generally sees only a small part of the surface, thus forcing S and $\partial\Omega$ to intersect. This renders the above approach useless: the Neumann condition would then force $\langle \hat{\boldsymbol{n}}, \hat{\boldsymbol{o}} \rangle = 0$, which will generally be violated by the sought-after surface. We shall therefore extend the proof of the normal speed equation (4.4), as shown in [SO05b], by formally deriving the natural boundary condition.

Let E be the level set version of a surface error functional of the form

$$E(\varphi) = \int_\Omega g(\boldsymbol{x}, \hat{\boldsymbol{n}}) \|\nabla\varphi\| \delta(\varphi) d\boldsymbol{x}$$

with $g : \mathbb{R}^3 \times S^2 \to \mathbb{R}_0^+$ integrable over S. For $g(\boldsymbol{x}, \hat{\boldsymbol{n}}) = \frac{1}{2}\|\hat{\boldsymbol{n}} - \hat{\boldsymbol{m}}(\boldsymbol{x})\|$, we have $E(\varphi) = J(\varphi)$ as in (4.1). We denote by $\boldsymbol{g}_{\hat{\boldsymbol{n}}} := D_S \hat{\boldsymbol{n}}$ the surface gradient of $\hat{\boldsymbol{n}}$.

Theorem 4. *Let the level set surface φ be a critical point of the functional E. Then for sufficiently smooth $\partial\Omega$ the natural boundary condition*

$$\langle \boldsymbol{g}_{\hat{\boldsymbol{n}}} + g\hat{\boldsymbol{n}}, \hat{\boldsymbol{o}} \rangle = 0 \tag{4.5}$$

on $\varphi^{-1}[\{0\}, t] \cap \partial\Omega$ follows. Generally, if φ fulfills this condition, we have the following expression for the Gâteaux derivative $dE(\varphi)\eta$ of E w.r.t. η:

$$dE(\varphi)\eta = -\int_\Omega \eta \operatorname{div}(\boldsymbol{g}_{\hat{\boldsymbol{n}}} + g\hat{\boldsymbol{n}}) \delta(\varphi) dx \tag{4.6}$$

for all $\eta \in C^\infty(\Omega)$.

Proof. For a normal variation $\varphi^\varepsilon := \varphi + \varepsilon\eta$, we have

$$dE(\varphi)\eta = \frac{d}{d\varepsilon} E(\varphi^\varepsilon)|_{\varepsilon=0} = \int_\Omega \frac{d}{d\varepsilon} \{g(\boldsymbol{x}, \hat{\boldsymbol{n}}) \|\nabla\varphi^\varepsilon\| \delta(\varphi^\varepsilon)\}|_{\varepsilon=0} d\sigma.$$

As shown in [SO05b], this leads to

$$dE(\varphi)\eta = \int_\Omega \langle \nabla\eta, \boldsymbol{g}_{\hat{\boldsymbol{n}}} + g\hat{\boldsymbol{n}} \rangle \delta(\varphi) d\boldsymbol{x} + \int_\Omega \eta g \|\nabla\varphi\| \delta'(\varphi) d\boldsymbol{x}.$$

Applying Gauss' theorem to the first integral yields

$$\begin{aligned} dE(\varphi)\eta &= \int_{\partial\Omega} \eta \langle \boldsymbol{g}_{\hat{\boldsymbol{n}}} + g\hat{\boldsymbol{n}}, \hat{\boldsymbol{o}} \rangle \delta(\varphi) d\sigma - \int_\Omega \eta \operatorname{div}\{(\boldsymbol{g}_{\hat{\boldsymbol{n}}} + g\hat{\boldsymbol{n}}) \delta(\varphi)\} d\boldsymbol{x} \\ &\quad + \int_\Omega \eta g \|\nabla\varphi\| \delta'(\varphi) d\boldsymbol{x} \\ &= \int_{\partial\Omega} \eta \langle \boldsymbol{g}_{\hat{\boldsymbol{n}}} + g\hat{\boldsymbol{n}}, \hat{\boldsymbol{o}} \rangle \delta(\varphi) d\sigma - \int_\Omega \eta \operatorname{div}(\boldsymbol{g}_{\hat{\boldsymbol{n}}} + g\hat{\boldsymbol{n}}) \delta(\varphi) d\boldsymbol{x} \\ &\quad - \int_\Omega \eta \langle \boldsymbol{g}_{\hat{\boldsymbol{n}}} + g\hat{\boldsymbol{n}}, \nabla\varphi \rangle \delta'(\varphi) d\boldsymbol{x} + \int_\Omega \eta g \|\nabla\varphi\| \delta'(\varphi) d\boldsymbol{x}. \end{aligned}$$

Observe that $\langle \boldsymbol{g}_{\hat{\boldsymbol{n}}}, \nabla\varphi \rangle = \langle \boldsymbol{g}_{\hat{\boldsymbol{n}}}, \hat{\boldsymbol{n}} \rangle \|\nabla\varphi\| = 0$, since the surface gradient $\boldsymbol{g}_{\hat{\boldsymbol{n}}}$ by definition lies in the tangent plane of S. Also $\langle \hat{\boldsymbol{n}}, \nabla\varphi \rangle = \|\nabla\varphi\|$, so the remaining terms in the last line cancel

each other leaving

$$dE(\varphi)\eta = \int_{\partial\Omega} \eta\langle \boldsymbol{g}_{\hat{\boldsymbol{n}}} + g\hat{\boldsymbol{n}}, \hat{\boldsymbol{o}}\rangle\delta(\varphi)d\sigma - \int_{\Omega} \eta \operatorname{div}(\boldsymbol{g}_{\hat{\boldsymbol{n}}} + g\hat{\boldsymbol{n}})\delta(\varphi)d\boldsymbol{x}.$$

Now, let φ be a critical point of E. Constraining the choice of η to functions vanishing outside a compact domain, we see that the right side vanishes for all such η. Since this applies to all η, it forces $\operatorname{div}(\boldsymbol{g}_{\hat{\boldsymbol{n}}} + g\hat{\boldsymbol{n}})\delta(\varphi) \equiv 0$. But this means the left integral must vanish for all η by itself. Again, this applies to all $\eta \in C^\infty(\Omega)$, resulting in the natural boundary condition $\langle \boldsymbol{g}_{\hat{\boldsymbol{n}}} + g\hat{\boldsymbol{n}}, \hat{\boldsymbol{o}}\rangle\delta(\varphi) \equiv 0$ on $\partial\Omega$.

On the other hand, if φ satisfies this boundary restriction, we have

$$dE(\varphi)\eta = -\int_{\Omega} \eta \operatorname{div}(\boldsymbol{g}_{\hat{\boldsymbol{n}}} + g\hat{\boldsymbol{n}})\delta(\varphi)dx.$$

□

The term $\operatorname{div}(\boldsymbol{g}_{\hat{\boldsymbol{n}}} + g\hat{\boldsymbol{n}})$ is the so-called shape gradient of E so in view of (4.4), our minimization scheme is of steepest descent type. For a thorough introduction from this perspective, see [DZ01, Bur03, CKPF05].

4.3 The space of admissible gradients

In the case of the specular flow error functional H from (2.9), we face two difficulties:

1. H contains derivatives of φ up to second order (cf. Theorem 2), while the above derivation only permits first order derivatives.
2. We must respect the normal condition, i.e. we may not leave $\mathcal{R}_{\hat{\boldsymbol{m}}}$ during the time advancement step.

Let us consider the first point: while it is possible to derive a gradient descent process similar to (4.3), this requires computing surface derivatives of up to fourth order. Following the common Lagrangian approach, the gradient must then be projected onto a – yet to be determined – space $\mathcal{A}_{\hat{\boldsymbol{m}}}$. This space consists of all gradients v of surface evolutions through the current surface that obey the normal condition.

We will instead project on a vector space $\tilde{\mathcal{A}}_{\hat{\boldsymbol{m}}}$ containing $\mathcal{A}_{\hat{\boldsymbol{m}}}$ that in practice turns out to be one-dimensional. $\mathcal{A}_{\hat{\boldsymbol{m}}}$ is closed under scalar multiplication, as can be seen by scaling the time parameter of the underlying surface evolution of $v \in \mathcal{A}_{\hat{\boldsymbol{m}}}$. So we may as well project onto $\tilde{\mathcal{A}}_{\hat{\boldsymbol{m}}}$ whenever $\mathcal{A}_{\hat{\boldsymbol{m}}} \neq \{0\}$, i.e. whenever noise in the measurements of normals is small enough. The gradient projection then reduces to the computation of a directional derivative of H, which can be numerically approximated, sparing the cumbersome calculation of the complete gradient.

Proposition 5. *Let $\varphi : \Omega \times T \to \mathbb{R}$ be a sufficiently differentiable level set evolution respecting the normal condition at each point in time, i.e. $\hat{\boldsymbol{n}}[\boldsymbol{s}, t] = \hat{\boldsymbol{m}}(\boldsymbol{s}), t \in T$, $\boldsymbol{s} \in S(t)$. Then for all $t \in T$ and $\boldsymbol{s} \in S(t)$, the following relationship holds:*

$$D_S v + (\nabla\hat{\boldsymbol{m}})\hat{\boldsymbol{m}}v = 0. \tag{4.7}$$

Here $D_S v$ denotes the surface gradient of v on S.

For a fixed point in time, let $\tilde{\mathcal{A}}_{\hat{\boldsymbol{m}}}$ consist of all v satisfying (4.7). The linear structure confirms that $\tilde{\mathcal{A}}_{\hat{\boldsymbol{m}}}$ is a vector space. Forcing a Lipschitz property for the measured normals $\hat{\boldsymbol{m}}$, we can make sure the solution of (4.7) is unique up to a scalar.

Proof of Proposition 5. For a sufficiently differentiable φ and $t_0 \in T$, $\boldsymbol{x}_0 \in S(t_0)$, we can locally find a curve $\boldsymbol{s} : T \to \mathbb{R}^3$ on S with $\boldsymbol{s}(t_0) = \boldsymbol{x}_0$. This can be seen making the ansatz $\boldsymbol{s}(t) = \boldsymbol{x} + \hat{\boldsymbol{n}}(\boldsymbol{x}_0, t_0)\lambda(t), \lambda(t_0) = 0$, and using an implicit function argument to find $\lambda \in C^1(T_\varepsilon, \mathbb{R})$ in a neighbourhood $T_\varepsilon := (t_0 - \varepsilon, t_0 + \varepsilon)$ of t_0.

From Lemma 1, we get $D_S v + \frac{d}{dt}\{\hat{\boldsymbol{n}}[\boldsymbol{s}(t),t]\} - d_{\mathbf{P}_{\hat{\boldsymbol{n}}}\dot{\boldsymbol{s}}}\hat{\boldsymbol{n}} = 0$. Using the normal condition $\hat{\boldsymbol{n}}[\boldsymbol{s}(t),t] = \hat{\boldsymbol{m}}(s(t))$, it follows that

$$D_S v + (\nabla\hat{\boldsymbol{m}})\dot{\boldsymbol{s}} - d_{\mathbf{P}_{\hat{\boldsymbol{n}}}\dot{\boldsymbol{s}}}\hat{\boldsymbol{m}} = 0.$$

Decompose $\boldsymbol{s}$ orthogonally, then $(\nabla\hat{\boldsymbol{m}})\dot{\boldsymbol{s}} = d_{\mathbf{P}_{\hat{\boldsymbol{n}}}\dot{\boldsymbol{s}}}\hat{\boldsymbol{m}} + d_{\hat{\boldsymbol{n}}\hat{\boldsymbol{n}}^\top\dot{\boldsymbol{s}}}\hat{\boldsymbol{m}}$ and

$$D_S v + d_{\hat{\boldsymbol{n}}\hat{\boldsymbol{n}}^\top\dot{\boldsymbol{s}}}\hat{\boldsymbol{m}} = 0.$$

Substituting $d_{\hat{\boldsymbol{n}}\hat{\boldsymbol{n}}^\top\dot{\boldsymbol{s}}}\hat{\boldsymbol{m}} = (\nabla\hat{\boldsymbol{m}})\hat{\boldsymbol{n}}\hat{\boldsymbol{n}}^\top\dot{\boldsymbol{s}} = (\nabla\hat{\boldsymbol{m}})\hat{\boldsymbol{m}}\hat{\boldsymbol{m}}^\top\hat{\boldsymbol{m}}v = (\nabla\hat{\boldsymbol{m}})\hat{\boldsymbol{m}}v$ yields the assertion. □

4.4 The minimization algorithm

Let us now outline the complete gradient descent procedure for solving the constrained minimization problem for general H.

1. Choose a start value $S(t_0 = 0)$ for the surface iteration.
2. Advance S in time according to (4.4) until $J(S(t)) < \varepsilon_n$.
3. While $H(S(t)) \geqslant \varepsilon_H$, repeat:
   (a) For $S(t)$, find a basis vector b of the discrete version of $\tilde{\mathcal{A}}_{\hat{\boldsymbol{m}}}$.
   (b) Numerically, approximate the directional derivative $dH(t)b$. The gradient of H projected onto $\tilde{\mathcal{A}}_{\hat{\boldsymbol{m}}}$ is then $\tilde{v} = b\{dH(S(t))b\}$.
   (c) Advance $t \to t+\delta_t$ and $S(t) \to S(t+\delta_t)$ according to the normal speed given by $\tilde{v}$.

Depending on the specific method used, the time discretization step may cause an increase in the normal error J during phase 3. In our prototype, we use a simple forward Euler step with some additional repetitions of step 2 if the normal error exceeds a certain upper bound. For more implementation details, see [Lel06].

Note that the formulation works for any H, not only for the optical flow case. Also note that though the derivation of the proposed method is based on the level set calculus, we are free to use any other representation for the implementation. As an example, we shall briefly discuss the case of an explicit parametrization in two dimensions. Here we have $f : X \times T \to \mathbb{R}^+$ for an interval $X = [x_l, x_r]$ and $\Omega = X \times \mathbb{R}^+$. The surface is then simply the graph of f, $S(t) = \{(x, f[x,t])^\top \mid x \in X\}$. Some important normal related properties are

$$\hat{\boldsymbol{n}} = \frac{1}{\sqrt{(f')^2+1}}\begin{pmatrix} f' \\ -1 \end{pmatrix}, \quad \nabla\hat{\boldsymbol{n}} = \frac{1}{[(f')^2+1]^{\frac{3}{2}}}\begin{pmatrix} f'' & 0 \\ f'f'' & 0 \end{pmatrix}, \quad \operatorname{div}\hat{\boldsymbol{n}} = \frac{f''}{[(f')^2+1]^{\frac{3}{2}}}.$$

Consider a curve $\boldsymbol{s}$ on S, i.e. $\boldsymbol{s}(t) = (x(t), f[x(t),t])^\top$. By $v = \langle\dot{\boldsymbol{s}}, \hat{\boldsymbol{n}}\rangle$, we get

$$v = \frac{1}{\sqrt{(f')^2+1}}\left\langle\begin{pmatrix} \dot{x} \\ f'\dot{x} + \dot{f} \end{pmatrix}, \begin{pmatrix} f' \\ -1 \end{pmatrix}\right\rangle = \frac{-\dot{f}}{\sqrt{(f')^2+1}}.$$

The normal velocity field v translates into the temporal derivative of f by $\dot{f} = -v\sqrt{f'^2+1}$, in direct analogy to the level set equation $\dot{\varphi} = -v\|\nabla\varphi\|$:

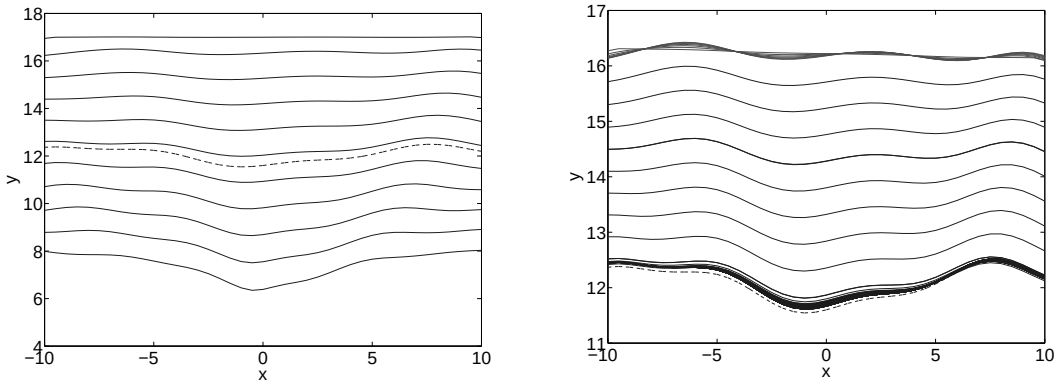
$$\left\langle\frac{1}{\sqrt{f'^2+1}}\begin{pmatrix} f' \\ -1 \end{pmatrix} - \hat{\boldsymbol{m}}, \begin{pmatrix} -1 \\ 0 \end{pmatrix}\right\rangle = 0,$$

where $\hat{\boldsymbol{m}} = (\hat{\boldsymbol{m}}_1, \hat{\boldsymbol{m}}_2)^T$. This scalar equation is equivalent to

$$f' = \hat{\boldsymbol{m}}_1\sqrt{f'^2+1}. \tag{4.8}$$

Assuming that the normal field $\hat{\boldsymbol{m}}$ is compatible with the surface model in the sense that through every point $\boldsymbol{x}$, one can actually find a surface in this parametrization s.th. $\hat{\boldsymbol{m}}(\boldsymbol{x}) = \hat{\boldsymbol{n}}(\boldsymbol{x})$, the case $\hat{m}_2 = 0$ can be excluded. We finally obtain

$$f'(x_l) = \frac{\hat{m}_1(x_l)}{|\hat{m}_2(x_l)|}.$$



(a) (b)

Figure 4: (a) The space $\mathcal{R}_{\hat{\boldsymbol{m}}}$ visualized by a set of surfaces satisfying the normal condition for a curved surface (dashed). Here, Ω_I is the $z = 1$ plane, $\Omega = (-10, 10) \times \mathbb{R}^+$, and the scene L is the $z = 0$ plane. (b) Reconstruction process for a curved surface starting with $f^0 \equiv 16$. Curves represent snapshots of f^m every 5 steps for a total of 500 iterations. The normal adjustment takes place in the first few iterations at the top of the figure. The constrained minimization (black) converges to the correct solution (dashed).

as natural boundary condition. The same result applies to the right boundary. Note that, in this special case, this is equivalent to forcing $\hat{\boldsymbol{n}} = \hat{\boldsymbol{m}}$ on $S \cap \partial\Omega$, i.e. on the border the minimizing surface normals coincide exactly with those of the measured normal field.

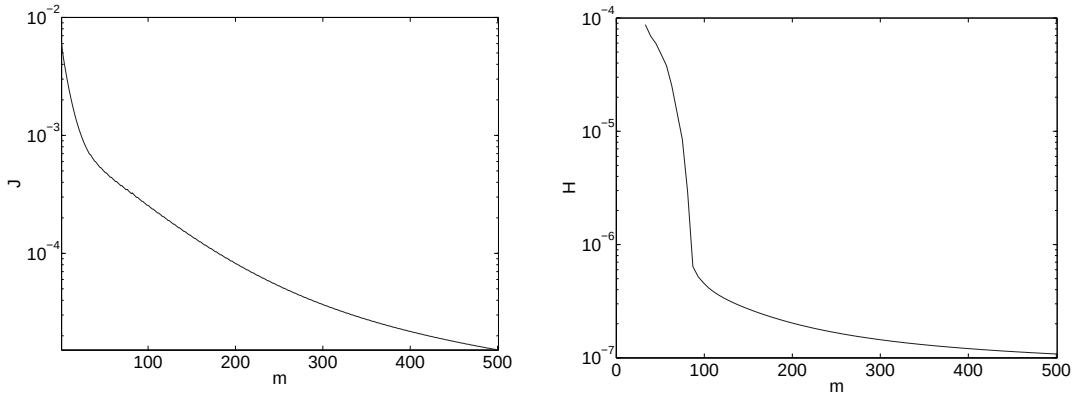
5 Experimental results

To verify the feasibility of the presented minimization process, we created a MATLAB implementation of the algorithm. For the evaluation of our method, we used the above explicit parametrization for two-dimensional Ω combined with a simple finite difference scheme and forward Euler step method for solving the PDE. At the m-th iteration we have $f^m = (f_1^m, \ldots, f_N^m)$ and $\dot{f}^m = (\dot{f}_1^m, \ldots, \dot{f}_N^m)$. Using central differences for $\dot{f}^m$, the discrete equivalent of the space $\tilde{\mathcal{A}}_{\hat{\boldsymbol{m}}}$ is the kernel of a matrix of the form

$$\mathbf{A} = \begin{pmatrix} \boldsymbol{a}_{0,1} & \boldsymbol{a}_{0,2} & & & \\ & \boldsymbol{a}_{1,1} & \boldsymbol{a}_{1,2} & & \\ & & \ddots & \ddots & \\ & & & \boldsymbol{a}_{N-1,1} & \boldsymbol{a}_{N-1,2} \end{pmatrix} \in \mathbb{R}^{(2N-2)\times N},$$

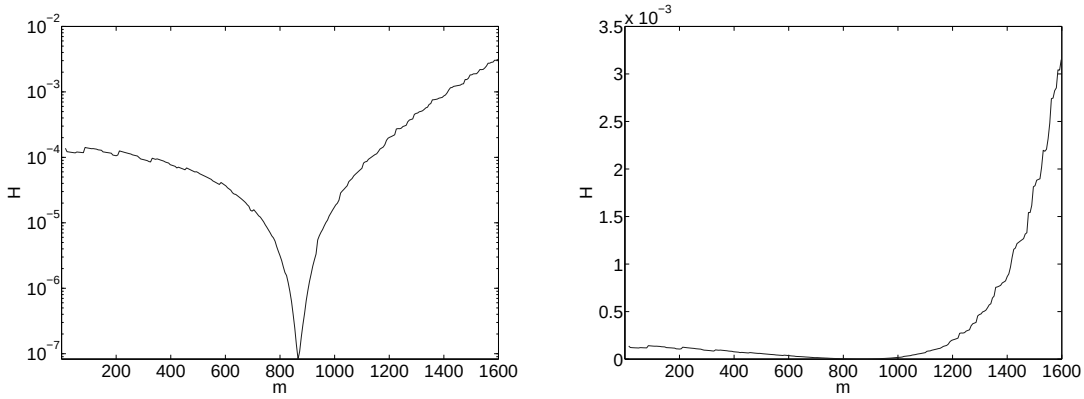
where each $\boldsymbol{a}_{i,j}$ represents an $\mathbb{R}^2$ vector. Looking at the defining equation (4.7), we expect rows $2k$ and $2k+1$ to constitute one scalar equation thus limiting the rank of $\mathbf{A}$ to $N-1$. In this special two-dimensional case, one could actually parameterize v by arc length of the surface curve $S(t)$. This transforms (4.7) into a homogenous linear ODE for v, so all solutions are scalar multiples of an arbitrary non-zero solution v_0. Thus, the kernel of $\mathbf{A}$ should be one-dimensional, which could be confirmed in all test cases. This supports the assumption about the simple structure of $\mathcal{R}_{\hat{\boldsymbol{m}}}$. Figure 4(a) shows a visualization of $\mathcal{R}_{\hat{\boldsymbol{m}}}$ for a sample curved surface. This surface is common for all following illustrations.

In order to obtain good data for the normal field, the discrete simulation values of the light map $\boldsymbol{l}$ are interpolated using cubic B-splines. For the optical flow data, we use a piecewise linear interpolation, which simplifies the exclusion of singular points. To estimate the error, we recorded the values of the error functionals J and H. Figures 4 and 5 show the steps taken as well as the error evolution for the reconstruction of a sample surface. As can be seen, the surface converges within expectations for the simple discretization.



(a) (b)

Figure 5: Error progression for (a) normal and (b) specular flow error functionals J and H, respectively.



(a) (b)

Figure 6: The (a) logarithmic and (b) linear specular flow error functional evaluated for several surfaces, all obeying the normal constraint, see also Figure 4. The minimum uniquely identifies the desired surface reconstruction.

Notwithstandig these encouraging results, one might ask if the optical flow is actually a good candidate to select a unique solution from $\mathcal{R}_{\hat{\boldsymbol{m}}}$. To investigate this, we evaluated the optical flow functional H for a number of solution candidates in $\mathcal{R}_{\hat{\boldsymbol{m}}}$ Figure 6. While there are some – possibly discretization caused – local minima, the specular flow error exhibits a distinct global minimum at the reference surface.

Extending the method to three dimensions is straightforward. Figure 7 shows the result of a simulation for a curved three-dimensional surface. The corresponding error evolution is shown in Figure 8.

6 Conclusion

Based on the theory of level set evolutions, we contributed a very general model for describing optical flow fields that originate from the imaging of a scene over a time-variant specular object. We could additionally derive necessary and sufficient conditions for these modelling equations to be invertible.

A novel view of the reconstruction problem was put forward henceforth: light map mea-

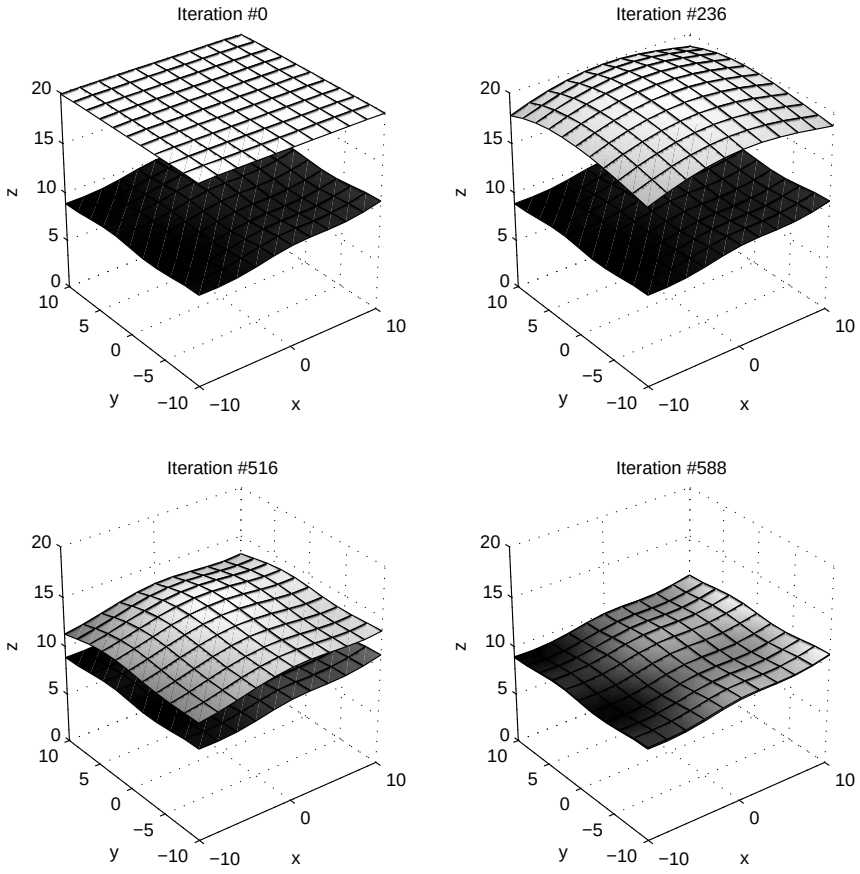
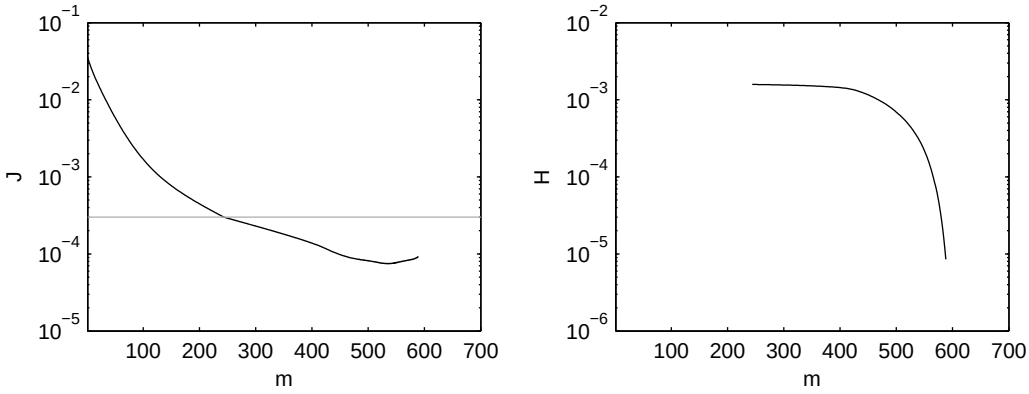


Figure 7: Steps during the reconstruction of a two-dimensional curved surface (dark), starting at $f^0 \equiv 20$ (light). At iteration 236, the normal error has become small enough to start the constrained gradient descent. The image domain Ω_I lies in the $z = 1$ plane, $\Omega = (-10, 10) \times (-10, 10) \times \mathbb{R}^+$, and the scene L is the $z = 0$ plane.



(a) (b)

Figure 8: Error evolution for (a) normal and (b) optical flow error functionals J and H. The normal error threshold ε_n is marked with a gray line.

surements were treated as a constraint for the minimization of an additional functional which – in our case – represented the optical flow error. This could then be solved by finding the steady-state solution of a partial differential equation under the normal constraint. To cope with partial measurements, a way of deriving natural boundary conditions was presented. To solve the PDE, a gradient projection type algorithm was proposed, which made it necessary to have a closer look at the space of admissible gradients. Finally, the results of a reference implementation were presented, suggesting good convergence. A side experiment helped to justify the choice of an optical flow-based method to pinpoint the unique solution to the reconstruction problem.

In the future, more precise statements about the structure of the candidate solution space should be possible. Adopting a more sophisticated integration process will certainly improve accuracy as well as convergence speed. Finally, an answer to the question of well-posedness completed with necessary and sufficient preconditions has yet to be found.

References

[ABFJB03] G. Aubert, M. Barlaud, O. Faugeras, and S. Jehan-Besson. Image segmentation using active contours: Calculus of variations or shape gradients? *SIAM J. Appl. Math*, 63(6):2128–2154, 2003.

[BFB94] J.L. Barron, D.J. Fleet, and S.S. Beauchemin. Performance of optical flow techniques. *IJCV*, 12(1):43–77, 1994.

[Bon06] T. Bonfort. *Reconstruction de surfaces reflechissantes a partier d'images*. PhD thesis, Institut National Polytechnique de Grenoble, 2006.

[BS03] T. Bonfort and P. Sturm. Voxel carving for specular surfaces. In *ICCV*, volume 1, pages 591–596, 2003.

[Bur03] M. Burger. A framework for the construction of level set methods for shape optimization and reconstruction. *Interfaces and Free Boundaries*, 5:301–329, 2003.

[BWB06] J. Balzer, S. Werling, and J. Beyerer. Regularization of the deflectometry problem using shading data. In *Proceedings of the SPIE Optics East*, 2006.

[CA00] M. Chen and J. Arvo. Theory and application of specular path pertubation. *ACM Transactions on Graphics*, 19(1):246–278, 2000.

[CKPF05] G. Charpiat, R. Keriven, J.-P. Pons, and O. Faugeras. Designing spatially coherent minimizing flows for variational problems based on active contours. In *ICCV*, volume 2, pages 1403–1408, 2005.

[dC76] M. do Carmo. *Differential geometry of curves and surfaces*. Prentice-Hall, 1976.

[DZ01] M. Delfour and J.-P. Zolesio. *Shapes and Geometries*. SIAM, 2001.

[GM04] M. Goldlücke and M. Magnor. Weighted minimal hypersurfaces and their applications in computer vision. In *ECCV*, volume 2, pages 366–378, 2004.

[Hor70] B. Horn. *Shape from Shading: A Method for Obtaining the Shape of a Smooth Opaque Object from One View*. PhD thesis, Department of Electrical Engineering, MIT, 1970.

[Hor86] B. Horn. *Robot vision*. MIT Press, 1986.

[HP04] R. Andrew Hicks and Ronald K. Perline. The method of vector fields for catadioptric sensor design with applications to panoramic imaging. In *CVPR*, volume 2, pages 143–150, Jun 2004.

[HZ03] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2nd edition, 2003.

[JBHBA06] S. Jehan-Besson, A. Herbulot, M. Barlaud, and G. Aubert. *Handbook of Mathematical Models in Computer Vision*, chapter 19, pages 309–323. Springer, 2006.

[Jin03] H. Jin. Estimation of 3d surface shape and smooth radiance from 2d images: A level set approach. *Journal of Scientific Computing*, 19(1-3):267–292, 2003.

[Kam04] S. Kammel. *Deflektometrische Untersuchung spiegelnd reflektierender Freiformflächen*. PhD thesis, University of Karlsruhe, 2004.

[KD04] R. Kickingereder and K. Donner. Stereo vision on specular surfaces. In *Proceedings of IASTED Conference on Visualization, Imaging, and Image Processing*, pages 335–339, 2004.

[Lel06] J. Lellmann. Mathematical Modelling and Analysis of the Deflectometry Problem: Level Set based Reconstruction of Specular Free-Form Surfaces from Optical Flow. German Title: Mathematische Modellierung und Analyse des Deflektometrieproblems: Rekonstruktion reflektierender Freiformflächen anhand des optischen Flusses auf der Grundlage von Level Sets. Diploma thesis, University of Karlsruhe, 2006.

[OF02] S. Osher and R. Fedkiw. *Level set methods and dynamic implicit surfaces*. Springer, November 2002.

[ON95] M. Oren and S.K. Nayar. A theory of specular surface geometry. In *ICCV*, pages 740–747, 1995.

[PCF] E. Prados, F. Camilli, and O. Faugeras. A viscosity solution method for shape-from-shading without image boundary data. To appear in Mathematical Modelling and Numerical Analysis.

[RB06] S. Roth and M. Black. Specular flow and the recovery of surface structure. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2:1869–1876, 2006.

[SAH04] J. Solem, H. Aanaes, and A. Heyden. A variational analysis of shape from specularities using sparse data. In *3DPVT*, pages 26–33, 2004.

[SCP05] S. Savarese, M. Chen, and P. Perona. Local shape from mirror reflections. *International Journal of Computer Vision*, 64 (1):31–67, 2005.

[Set05] J. Sethian. *Level set methods and fast marching methods*. Cambridge University Press, 2nd edition, 2005.

[SO05a] J. Solem and N. Overgaard. A geometric formulation of gradient descent for variational problems with moving surfaces. In *Scale-Space*, pages 419–430, 2005.

[SO05b] J. Solem and N. Overgaard. A gradient descent procedure for variational dynamic surface problems with constraints. In *IEEE Proc. Workshop on Variational, Geometric and Level Set Methods in Computer Vision*, pages 332–343, 2005.

[ZGB89] A. Zissermann, P. Giblin, and Andrew Blake. The information available to a moving observer from specularities. *Image and Vision Computing*, 7:38–42, 1989.

IWRMM-Preprints seit 2004

Nr. 04/01 Andreas Rieder: Inexact Newton Regularization Using Conjugate Gradients as Inner IteractionMichael

Nr. 04/02 Jan Mayer: The ILUCP preconditioner

Nr. 04/03 Andreas Rieder: Runge-Kutta Integrators Yield Optimal Regularization Schemes

Nr. 04/04 Vincent Heuveline: Adaptive Finite Elements for the Steady Free Fall of a Body in a Newtonian Fluid

Nr. 05/01 Götz Alefeld, Zhengyu Wang: Verification of Solutions for Almost Linear Complementarity Problems

Nr. 05/02 Vincent Heuveline, Friedhelm Schieweck: Constrained H^1-interpolation on quadrilateral and hexahedral meshes with hanging nodes

Nr. 05/03 Michael Plum, Christian Wieners: Enclosures for variational inequalities

Nr. 05/04 Jan Mayer: ILUCDP: A Crout ILU Preconditioner with Pivoting and Row Permutation

Nr. 05/05 Reinhard Kirchner, Ulrich Kulisch: Hardware Support for Interval Arithmetic

Nr. 05/06 Jan Mayer: ILUCDP: A Multilevel Crout ILU Preconditioner with Pivoting and Row Permutation

Nr. 06/01 Willy Dörfler, Vincent Heuveline: Convergence of an adaptive *hp* finite element strategy in one dimension

Nr. 06/02 Vincent Heuveline, Hoang Nam-Dung: On two Numerical Approaches for the Boundary Control Stabilization of Semi-linear Parabolic Systems: A Comparison

Nr. 06/03 Andreas Rieder, Armin Lechleiter: Newton Regularizations for Impedance Tomography: A Numerical Study

Nr. 06/04 Götz Alefeld, Xiaojun Chen: A Regularized Projection Method for Complementarity Problems with Non-Lipschitzian Functions

Nr. 06/05 Ulrich Kulisch: Letters to the IEEE Computer Arithmetic Standards Revision Group

Nr. 06/06 Frank Strauss, Vincent Heuveline, Ben Schweizer: Existence and approximation results for shape optimization problems in rotordynamics

Nr. 06/07 Kai Sandfort, Joachim Ohser: Labeling of n-dimensional images with choosable adjacency of the pixels

Nr. 06/08 Jan Mayer: Symmetric Permutations for I-matrices to Delay and Avoid Small Pivots During Factorization

Nr. 06/09 Andreas Rieder, Arne Schneck: Optimality of the fully discrete filtered Backprojection Algorithm for Tomographic Inversion

Nr. 06/10 Patrizio Neff, Krzysztof Chelminski, Wolfgang Müller, Christian Wieners: A numerical solution method for an infinitesimal elasto-plastic Cosserat model

Nr. 06/11 Christian Wieners: Nonlinear solution methods for infinitesimal perfect plasticity

Nr. 07/01 Armin Lechleiter, Andreas Rieder: A Convergenze Analysis of the Newton-Type Regularization CG-Reginn with Application to Impedance Tomography

Nr. 07/02 Jan Lellmann, Jonathan Balzer, Andreas Rieder, Jürgen Beyerer: Shape from Specular Reflection Optical Flow

Eine aktuelle Liste aller IWRMM-Preprints finden Sie auf:

www.mathematik.uni-karlsruhe.de/iwrmm/seite/preprints