B-Spline Modeling of Road Surfaces With an Application to Free-Space Estimation

Andreas Wedel, Hernán Badino, Clemens Rabe, Heidi Loose, Uwe Franke, and Daniel Cremers, *Member, IEEE*

***Abstract*—We propose a general technique for modeling the visible road surface in front of a vehicle. The common assumption of a planar road surface is often violated in reality. A workaround proposed in the literature is the use of a piecewise linear or quadratic function to approximate the road surface. Our approach is based on representing the road surface as a general parametric B-spline curve. The surface parameters are tracked over time using a Kalman filter. The surface parameters are estimated from stereo measurements in the free space. To this end, we adopt a recently proposed road-obstacle segmentation algorithm to include disparity measurements and the B-spline road-surface representation. Experimental results in planar and undulating terrain verify the increase in free-space availability and accuracy using a flexible B-spline for road-surface modeling.**

***Index Terms*—B-spline, free space, road surface, v-disparity.**

I. Introduction

MODELING a vehicle's environment is challenging but absolutely essential in maneuvering autonomous vehicles. It includes the localization of moving objects, as well as the modeling of the stationary infrastructure. In an ideal world, all the necessary information is available on demand from an omniscient oracle. In reality, only a small portion of the information is available on demand by making use of a database. This may include the location of traffic signs, the information about road curvature, or even a 3-D model of the complete infrastructure.

In most environments with other traffic participants, such information has to be generated online using environment-perception techniques. The ideal environment-perception sensor generates a 3-D model of the vehicle environment. In this paper, we use rectified stereo camera images and focus on modeling the free space in front of the vehicle. The free space is the available space to maneuver a road vehicle to avoid collision with any object. It is described by the ground surface and is limited by other obstacles.

However, what defines an obstacle? In general, obstacle refers to something that stands in the way. In vehicle environments, it refers to a structure that blocks the path by sticking out of the ground surface. Common obstacle-detection algorithms detect obstacles by evaluating the height above ground, where the ground is modeled as a planar surface. In situations such as that shown in Fig. 1, the assumption of a planar ground surface is violated, and such a procedure fails.

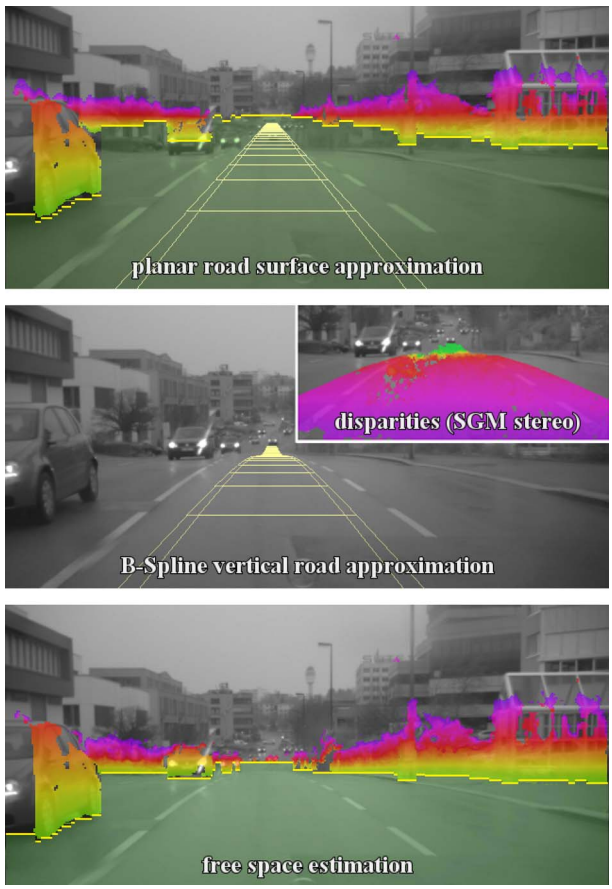


Fig. 1. Contributions in this paper. An example of a scene with undulating terrain in a city environment is shown. The road course ahead is planar in the vehicle vicinity; then, it drops down before it starts rising. The color encodes the relative height of obstacles for the free space and the distance for the disparities. (Top image) A planar ground assumption is invalid in the depicted scene and yields errors in the free-space estimation. The better vertical road approximation using the flexible spline representation and the correct free-space estimation is demonstrated in the lower images.

Manuscript received August 29, 2008; revised January 28, 2009 and June 3, 2009. First published August 11, 2009; current version published December 3, 2009.

A. Wedel is with the Environment Perception Group, Daimler Research, 71059 Sindelfingen, Germany, and also with the Computer Vision Group, University of Bonn, 53105 Bonn, Germany (e-mail: andreas.wedel@daimler.com).

H. Badino is with Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213-3890 USA (e-mail: hbadino@cs.cmu.edu).

C. Rabe is with the University of Kiel, 24118 Kiel, Germany (e-mail: uni-kiel.rabe@daimler.com).

H. Loose and U. Franke is with the Environment Perception Group, Daimler Research, 71059 Sindelfingen, Germany.

D. Cremers is with the Research Group for Computer Vision, Image Processing, and Pattern Recognition, University of Bonn, 53105 Bonn, Germany (e-mail: dcremers@cs.uni-bonn.de).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2009.2027223

A robust free-space estimation approach requires modeling the road surface to distinguish between obstacles and a free driving corridor. Assuming a planar road surface, slope changes in the road course ahead due to approaching a hill or a dip are not modeled and cannot be used to restrict the free space. In this paper, we develop an algorithm to model nonplanar road surfaces, which we represent as B-splines. The approximation via B-spline techniques, which are widely used in surface modeling, yields accurate results for the vertical road profile, even in large distances up to 100 m. Section II introduces B-splines and describes the estimation of the spline parameters from stereo disparity measurements. We describe how to track the spline parameters over time using a Kalman filter to improve accuracy and gain robustness in Section III.

In Section IV, we adopt a lately published free-space approach to use the obtained B-spline representation of the road surface. The original algorithm uses image edges to calculate a boundary between the road and the obstacle. In this paper, we extend this algorithm to use the disparity values of a stereo method as a second driving force for free-space estimation. We fuse both approaches, i.e., edge directions and disparities, into a single framework for free-space estimation, yielding better results in the novel combined approach. An evaluation section proving accurate free-space estimation in situations where the planar ground estimation fails demonstrates the practicability of ground surface modeling via B-splines.

II. Road Surface Modeling With B-Splines

In this section, we acquire the modeling of the road surface based on B-splines. We first replicate the common v-disparity approach for modeling planar road surfaces and give an overview of existing approaches to extend this approach toward modeling nonplanar surfaces. We will then discuss how a subset of these approaches can be modeled using B-splines, which are more general. The embedding into a Kalman filter framework is discussed in Section III.

A. Review of the v-Disparity Approach

The v-disparity approach was first introduced as Helmholtz shear in [1] and [2]. Under the assumption of a planar road without bank angle, the key idea is to fit a plane through 3-D measurements obtained by triangulating corresponding image points in the left and right camera images. An image point (u, v) in the left image corresponds to an image point $(u - d, v)$ in the right image. Disparity offset d is zero at the horizon and linearly increases in v, yielding $d(v) = a \cdot v + b$. v is the image row, and a and b are the parameters that depend on the camera height and tilt angle. A linear fit is computed in the image space using the Hough transform on a row-disparity image, which is commonly known as *v-disparity* space (see Fig. 2). Labayrade *et al.* were the first to introduce the name *v-disparity approach* and proposed a real-time accumulation strategy in [3].

We will review the idea of this robust v-disparity approach (from [3]) in the simple case of fixed camera height h. Each point on the ground plane is then described by $Y(Z) = \tan\alpha \cdot Z - h$ with camera tilt angle α and height Y and distance Z

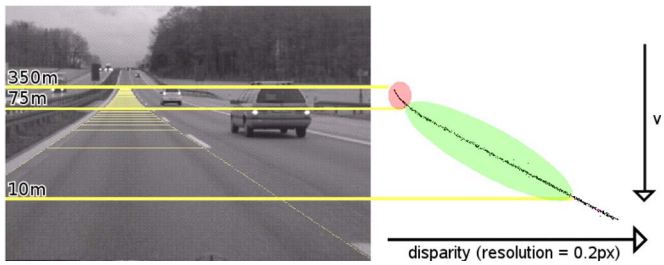


Fig. 2. v-disparity space of measurements on the road course ahead. The linear dependence between disparity and image row for the planar part of the road is visible. The plot also shows that the resolution of the v-disparity space (both v and d resolution) decreases with increasing distance.

of world points in the world coordinate system. Using the projection formulas for the pinhole camera and solving for tilt angle α, we get

$$\tan\alpha = \frac{h}{c_b f_x} \cdot d + \frac{1}{f_y}(c_y - v) \tag{1}$$

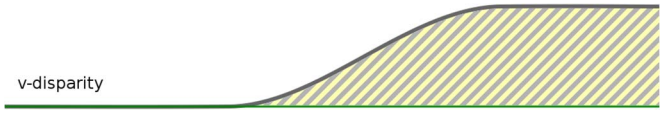
for each v-disparity point (v, d). Here, f_x and f_y are the focal lengths (in pixels), c_b is the baseline of the stereo camera system, and c_y is the y-coordinate of the principal point in the image.

For a robust estimate of the tilt angle from a set of v-disparity points, the histogram of the $\tan\alpha$ values calculated by (1) is analyzed, and the tilt angle is found as the maximum in the histogram. In addition, the variance and the number of v-disparity points supporting the found tilt angle are used as a quality measure.

B. Extending the v-Disparity Approach

In the literature on intelligent vehicles, some approaches have been proposed to yield an approximation of nonplanar road surfaces. We will describe the basic ideas of these approaches and exemplary point out their differences using an artificial ramp in the road course ahead (shown in gray). The camera is assumed on the left, and measurements are assumed to be noise free. The hatched regions show the approximation error for the different methods. As the focus is set on vertical road modeling, approaches that model the lateral surface change (such as [4] and [5]) are excluded from this review.

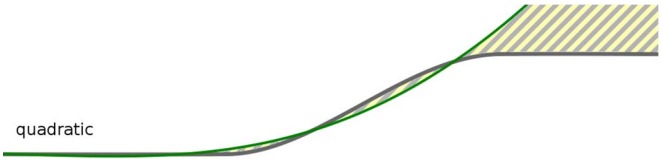
The Helmholtz shear approach, as described in [2], approximates the ground as a planar surface. It uses the Hough transform to fit a planar surface (*green*) in the disparity space, yielding accurate approximation if the planar vehicle vicinity but failing if the surface is rising or falling:



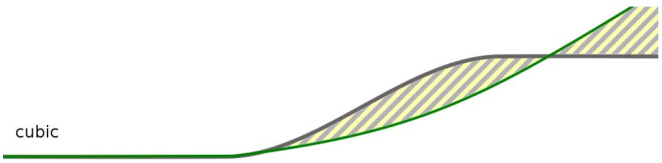
The envelope of the surface approach described in [3] computes the k main v-disparity surfaces in the region, where k has to be chosen (3 in the example, shown in *red*). The resulting surface (*green*) is represented as an inner or outer envelope (depending on the slope direction of the road). Surfaces are

modeled as piecewise planar; hence, slope changes are abrupt and not continuous. The approach incorporates the robust Hough transform techniques for the main surfaces but allows only for slope changes in one direction:

A quadratic approximation of the ground surface is proposed in [6]. It allows only slope changes in one direction and is not as stable in the vehicle vicinity as a Hough-transform-based approach. Using a B-spline of order two with one segment would yield the same result:

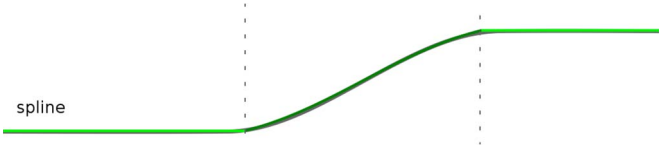


In [7], the authors propose a clothoid approximation of the ground surface. A clothoid is a higher order polynomial of degree 3 (cubic) that is widely used for lane recognition in the computer vision community. In their implementation, however, the cubic parametrization is not evaluated. The authors use a Hough transform for the vehicle vicinity and a quadratic fit in large distances, yielding a better approximation than a quadratic fit in total:



Using a B-spline of order two with two segments, where the first segment is constrained by the camera height and pitch, would yield similar results.

Due to its restricted parametrization, all aforementioned techniques can only model slope changes into one direction; hence, these approaches may fail to approximate the road surface if the road is undulating. The last example shows a surface approximation using a B-spline curve with three segments. Note the piecewise definition of the spline shown in light and dark colors and the good approximation of the surface ramp:



An example of undulating terrain in a real sequence with a B-spline surface reconstruction can be seen in Fig. 15. As we are interested in modeling the road course ahead (in large distances), fitting the approximated ground surface into the disparities, as done in the original a v-disparity approach, is not necessarily the best solution. If the modeling is done in image space, the resolution decreases with increasing distance (shown in Fig. 2). To overcome this effect, we propose to fit measurements in world coordinates, instead of image space, to get a well-defined depth resolution. The drawback of such an approach is that one has to account for the nonlinear error propagation of the (noisy) disparity measurements.

In the next section, we will review B-splines as a function vector space that embeds all polynomials, among which are the piecewise linear and clothoid functions. Hence, our approach is a generalization of known surface approximations in the literature and bridges the gap between these different approaches.

C. B-Splines

B-splines are a basis for the vector space of piecewise polynomials of degree γ [8]. A B-spline curve $B(Z)$ of degree γ is defined by an $n+1$-dimensional coefficient vector $\mathbf{c}$, i.e.,

$$B(Z) = \sum_{i=0}^{n} c_i N_{i,d}(Z) = \mathbf{N}_\gamma(Z)^\top \mathbf{c} \tag{2}$$

with

$$\mathbf{N}_\gamma(Z) = [N_{0,\gamma}(Z) \quad \cdots \quad N_{n,\gamma}(Z)]^\top$$

and

$$\mathbf{c} = [c_0 \quad \cdots \quad c_n]^\top.$$

Polynomial basis functions $\mathbf{N}_\gamma(Z) = \{N_{i,\gamma}(Z)\}_i$ with local support are described by

$$N_{i,j}(Z) = \frac{Z - T_i}{T_{i+j} - T_i} N_{i,j-1}(Z) + \frac{T_{i+j+1} - Z}{T_{i+j+1} - T_{i+1}} N_{i+1,j-1}(Z)$$

with

$$N_{i,0}(Z) = \begin{cases} 1, & T_i \le Z < T_{i+1} \\ 0, & \text{otherwise.} \end{cases}$$

The number of nodes n defines the number of piecewise intervals where $1 \le \gamma \le n$. We use a node vector

$$\mathbf{T} = \{T_0, \ldots, T_{n+d+1}\}$$

where T_i denotes distances in world coordinates and is in ascending order. Note that the following conditions have to be fulfilled:

$$T_0 = T_1 = \cdots = T_\gamma, \quad T_\gamma < T_{\gamma+1} < \cdots < T_n$$

and

$$T_n = T_{n+1} = \cdots = T_{n+\gamma+1}.$$

Fig. 3 shows basis functions $\mathbf{N}_1(Z)$ for piecewise linear splines and $\mathbf{N}_2(Z)$ for piecewise quadratic splines.

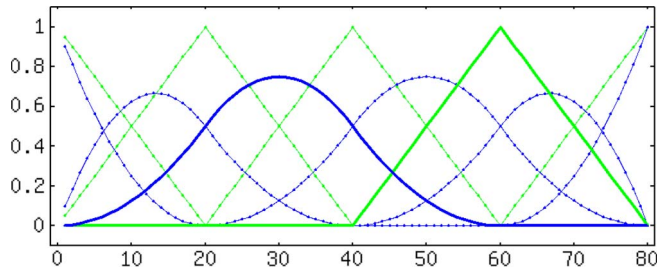


Fig. 3. Basis functions for an equidistant node vector. The linear basis functions are plotted in green, and quadratic basis functions are plotted in blue.

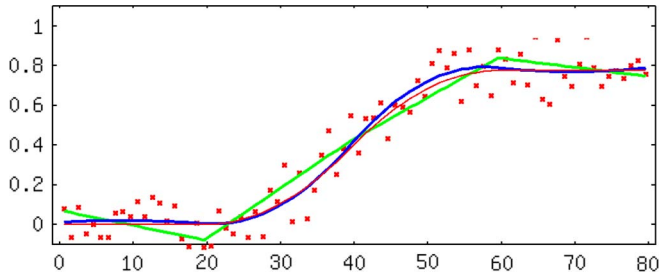


Fig. 4. (Red) Surface fit through measurements with additive Gaussian noise using (green) piecewise linear splines and (blue) piecewise quadratic splines.

To get the same parametrization as in the original v-disparity approach, both d and n must be set to 1. Note that only the parametrization is equivalent; the estimation technique via Hough transform is different from the least-square approach that we describe in the following. The quadratic and cubic ground surface approximation techniques are represented by setting $\gamma = 2$ and 3, respectively.

In our implementation, we use equidistant nodes within the observed distance interval and cubic splines. Further details on B-spline construction and evaluation can be found in [8].

Fig. 4 shows a surface fit with linear and quadratic splines using the same node vector. Note the better approximation with quadratic splines.

If a fixed node vector is used, the basis functions remain constant, and the spline function is altered only by coefficient vector $\mathbf{c}$. This yields the common name *control vector* for the vector of spline coefficients. Due to the fact that the basis functions for the B-spline fit do not change, they can be calculated in a precomputing step yielding real-time efficiency for the surface approximation via B-splines.

For the road course ahead, the B-spline $B(Z)$ encodes the relative height of the ground surface. If we are given M independent measurements

$$\{\text{distance, height}\}_{m=0}^{M} = \{Z_m, Y_m\}_{m=0}^{M}$$

and associated standard deviations σ_m (see [6] for the stereo triangulation error propagation), the goal is to find an optimal control vector $\mathbf{c}^*$ such that $B(Z)$ best fits to the measurements. The goodness of the fit can be expressed by a cost function evaluating the sum of deviations from the measurements

$$\mathbf{c}^* = \min_{\mathbf{c}} \left\{ \sum_m \frac{1}{\sigma_m^2} \left(B(Z_m) - Y_m \right)^2 \right\}. \tag{3}$$

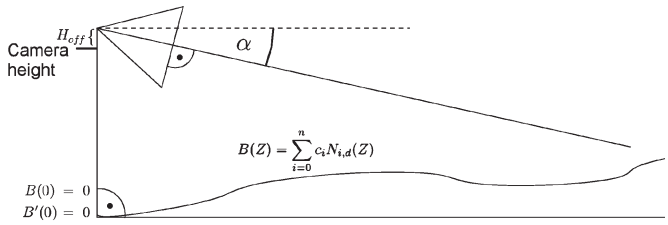


Fig. 5. Camera parameters pitch angle α and height offset H_{off}. The touch and gradient constraints imposed on the B-spline can be seen as boundary conditions, ensuring that the spline surface has height 0 and no slope at the camera footprint.

This boils down to finding the coefficient vector $\mathbf{c}^*$, which minimizes the weighted sum, i.e.,

$$\mathbf{c}^* = \min_{\mathbf{c}} \left\{ \sum_m \frac{1}{\sigma_m^2} \left(\mathbf{N}_\gamma(Z_m)^\top \mathbf{c} - Y_m \right)^2 \right\} \tag{4}$$

$$\Leftrightarrow \mathbf{c}^* = \min_{\mathbf{c}} \left\{ \underbrace{\begin{bmatrix} \frac{1}{\sigma_0^2} \mathbf{N}_\gamma(Z_0)^\top \\ \vdots \\ \frac{1}{\sigma_m^2} \mathbf{N}_\gamma(Z_m)^\top \end{bmatrix}}_{\mathbf{A}} \mathbf{c} - \underbrace{\begin{bmatrix} Y_0 \\ \vdots \\ Y_m \end{bmatrix}}_{\mathbf{h}} \right\} \tag{5}$$

and yields the familiar form of the least-squares problem

$$\mathbf{A}^\top \mathbf{A} \mathbf{c}^* = \mathbf{A}^\top \mathbf{h}. \tag{6}$$

Equation (6) can be solved either directly by matrix inversion or iteratively using any matrix-vector solver. Due to the embedding in a Kalman filter framework, these equations are fed as measurements to the Kalman filter and are not directly solved. The state vector of the Kalman filter is the coefficient vector $\mathbf{c}$. Each row of $\{\mathbf{A}\mathbf{c} - \mathbf{h}\}$ is a Kalman filter measurement equation.

D. Camera Parameters

Until now, the outer orientation of the camera is assumed to remain unchanged; therefore, only changes in surface topology are accounted for. In vehicle applications, because of vehicle motion, one has to model changes in the camera pitch angle α and an offset in the camera height H_{off} (see Fig. 5). To account for these parameters, the surface equation is extended by these two parameters, i.e.,

$$Height(Z) = \cos(\alpha) \mathbf{N}_\gamma(Z)^\top \mathbf{c} + \sin(\alpha) Z + H_{\text{off}}. \tag{7}$$

Recall that distance Z and the height are given in the world coordinate system of the moving observer. Because the camera height and the camera pitch angle could be modeled by translating and rotating the ground surface, additional boundary conditions have to be imposed. For moving platforms, these boundary conditions are straightforward: The vehicle has to touch the ground surface, and the surface gradient, where the vehicle touches the ground, has to vanish. Mathematically, this can be formulated as

$$\text{Touch constraint:} \quad B(0) = 0 \tag{8}$$

$$\text{Gradient constraint:} \quad B'(0) = 0. \tag{9}$$

We include both equations in the Kalman filter framework as measurements $\{m_t = B(Z_0) - 0\}$ and $\{m_g = B'(Z_0) - 0\}$. Because the B-spline equation is linear in the coefficient vector $\mathbf{c}$, the measurement equation m_t for the Kalman filter is a linear measurement equation. The same linearity of the coefficients is true for the derivative measurement m_g; the derivative of a B-spline $B'(Z)$ computes as

$$B'(Z) = \sum_{i=0}^{n} c_i N'_{i,d}(Z)$$

with (see [8])

$$N'_{i,d}(Z) = \frac{d}{Z_{i+d} - Z_i} N_{i,d-1}(Z) - \frac{d}{Z_{i+d+1} - Z_{i+1}} N_{i+1,d-1}(Z). \tag{10}$$

E. Surface Smoothness

For measurements corrupted by noise, a best fitted curve approximation is not necessarily the best continuous approximation. Two problems may arise: The number of measurements is too small to estimate all parameters, and/or outliers may influence the result. One way to solve for such problems is prior knowledge in terms of smoothness. The curvature and the inclination of road surfaces are usually small. This knowledge can be introduced by penalizing high curvature and derivatives of the resulting B-spline. Since integration and differentiation are linear operators on the vector space of B-splines [compare (10)], this can be formulated by additionally penalizing the quantities

$$\int \left(B'(Z)\right)^2 = \int \left(\mathbf{N}'_\gamma(Z)^\top \mathbf{c}\right)^\top \left(\mathbf{N}'_\gamma(Z)^\top \mathbf{c}\right) \tag{11}$$

$$\int \left(B''(Z)\right)^2 = \int \left(\mathbf{N}''_\gamma(Z)^\top \mathbf{c}\right)^\top \left(\mathbf{N}''_\gamma(Z)^\top \mathbf{c}\right). \tag{12}$$

The integrals over the basis functions are efficiently computed using Gaussian quadrature. This step is computed offline, because the basis functions remain constant.

The Gaussian quadrature, with n_g being the number of (control) weights, is exact for polynomials with degree $2n_g - 1$. For $n_g = 2$, the corresponding weights become $\omega_{1,2} = 1$ at positions $x_1 = -\sqrt{1/3}$ and $x_2 = \sqrt{1/3}$, and the approximation is exact for polynomials of degree ≤ 3. The Gaussian quadrature is defined on the interval $[a, b]$ as

$$\int_a^b f(x)dx \approx \frac{b-a}{2} \sum_{i=1}^{n_g} \omega_i f\left(\frac{b-a}{2} x_i + \frac{a+b}{2}\right). \tag{13}$$

For computing the smoothness constraints in (11) and (12), the function $f(x)$ becomes the multiplication of two basis functions, i.e.,

$$f(x) = N'_{j,d}(x) N'_{j,d}(x) \quad \text{and} \quad f(x) = N''_{j,d}(x) N''_{j,d}(x).$$

Because each of the first-derivative basis function has degree $d - 1$, the Gaussian quadrature for the integral of derivatives is exact for B-splines of degree $d = n_g$. The Gaussian quadrature for the integral of the second derivatives is exact for B-splines of degree $d = 2n_g$. Note that the local support of the basis polynomials can be used to gain real-time processing.

III. Road Surface Kalman Filter

Since we work on image sequences, we impose regularity over time by applying a Kalman filter. This section describes the filter steps for the surface fit based on B-splines. Measurement equations for 3-D world points are derived. A Kalman update is formulated, and the Kalman prediction step is derived for the B-spline coefficients, assuming a moving platform. For an introduction on Kalman filters, see [9].

The Kalman filter state vector $\mathbf{x}$ consists of the current parameter vector $\mathbf{c}$ describing the road profile, the camera pitch angle α, and the camera height offset H_{off}, i.e.,

$$\mathbf{x} = \begin{bmatrix} \mathbf{c} & \alpha & H_{\text{off}} \end{bmatrix}^\top. \tag{14}$$

This includes the physical actual movement of the vehicle in terms of height and pitch angle change and implies optimization of the road surface in the coefficient space. In practice, this is still acceptable, because a one-to-one mapping from the coefficient space into world coordinates (height and distance) exists.

A. Kalman Update Step

Given an initial estimate $\mathbf{x}' = [\mathbf{c}', \alpha', H'_{\text{off}}]^\top$, the goal is to derive an equation for the update $\Delta\mathbf{x} = [\Delta\mathbf{c}, \Delta\alpha, \Delta H_{\text{off}}]^\top$ to get a better (updated) solution

$$\mathbf{x}^* = \mathbf{x}' + \Delta\mathbf{x} = \left[\mathbf{c}^*, \alpha^*, H^*_{\text{off}}\right]^\top.$$

Recall that B-splines form a function vector space and are linear in their coefficients. Assuming a small pitch angle, we can set $\cos(\alpha) \approx 1$ and $\sin(\alpha) \approx \alpha$ and get

$$\begin{bmatrix} \mathbf{N}_\gamma(Z_0)^\top & Z_0 & 1 \\ \vdots & & \\ \mathbf{N}_\gamma(Z_m)^\top & Z_m & 1 \end{bmatrix}^\top \begin{bmatrix} \mathbf{c}' + \Delta\mathbf{c} \\ \alpha' + \Delta\alpha \\ H'_{\text{off}} + \Delta H_{\text{off}} \end{bmatrix} = \begin{bmatrix} Y_0 \\ \vdots \\ Y_m \end{bmatrix}.$$

The standard deviations for the single measurements are calculated via error propagation from stereo triangulation [10]. We assume a disparity standard deviation of 0.4 px. Our cameras used in the experiments have a baseline of 35 cm and a focal length of 840 px.

The touch and gradient constraint boundary conditions in (8) and (9) need to be formulated as measurements for the Kalman filter. This is done by introducing the deviation from zero for the B-spline and its derivative as additional measurements, i.e.,

$$\underbrace{\begin{bmatrix} \mathbf{N}_\gamma(0)^\top \\ \mathbf{N}'_\gamma(0)^\top \end{bmatrix}}_{\mathbf{P}_C} [\mathbf{c}' + \Delta\mathbf{c}] = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{15}$$

We choose the standard deviations for these measurements on the order of 10^{-5}, allowing only small deviations to the constraints.

The smoothness constraints in (11) and (12) are quadratic in the coefficients $\mathbf{c}$. We linearize the equations around the given

estimate $\mathbf{c}'$ to solve for the update. Hence, we used lagged feedback for the smoothness constraints, i.e.,

$$\underbrace{\begin{bmatrix}\mathbf{c}'^{\top}\int \mathbf{N}'_{\gamma}(Z)\mathbf{N}'_{\gamma}(Z)^{\top}dZ \\ \mathbf{c}'^{\top}\int \mathbf{N}''_{\gamma}(Z)\mathbf{N}''_{\gamma}(Z)^{\top}dZ\end{bmatrix}}_{\mathbf{P}_S}[\mathbf{c}'+\Delta\mathbf{c}] = \begin{bmatrix}0\\0\end{bmatrix}. \tag{16}$$

The summarized Kalman filter update equations for the measurement equations, the constraints, and the smoothness equations read

$$\begin{bmatrix}\mathbf{N}_{\gamma}(Z_0)^{\top} & Z_0 & 1\\ \vdots & & \\ \mathbf{N}_{\gamma}(Z_m)^{\top} & Z_m & 1\\ \mathbf{P_C} & \mathbf{0} & \mathbf{0}\\ \mathbf{P_S} & \mathbf{0} & \mathbf{0}\end{bmatrix}\begin{bmatrix}c'+\Delta c\\ \alpha'+\Delta\alpha\\ H'_{\text{off}}+\Delta H_{\text{off}}\end{bmatrix} = \begin{bmatrix}Y_0\\ \vdots\\ Y_m\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix}.$$

Both matrices P_C and P_S can be precomputed for a fixed node vector to save online computation time. Most of the measurement equations can also be precomputed using a simple trick (at the cost of some negligible inaccuracies): For each measurement equation, the basis functions N_γ need to be evaluated at the given distance Z_m. In our experiments, we sum up all measurements within discrete equidistant distance intervals

$$\{I_{Z_i}, I_{Z_{i+}}\} \quad \text{with} \quad Z_{i+1} = Z_i + \Delta Z.$$

The basis functions can be precomputed for the middle point of every discrete interval, and the corresponding row of the Kalman filter update equations is multiplied with the square root of the number of measurements per interval (because the solution is computed via least squares). For the sum of variances, we use error propagation to weigh the measurements of the Kalman filter with the correct variances [10].

B. Kalman Prediction Step

The Kalman filter prediction step models the dynamics of the system. For a moving vehicle, the translation and rotation of the vehicle have to be modeled.

We model the road surface in the coordinate system of a moving observer. The node vector is kept fixed in predefined distances. Keeping the node vector fixed, old B-spline coefficients $\mathbf{c}'$ have to be projected onto the current coefficients $\mathbf{c}$. Minimizing the quadratic difference between the last and current surface parametrization under the translation T yields

$$\min_{\mathbf{c}} \int \left(\mathbf{N}_{\gamma}(Z)^{\top}\mathbf{c} - \mathbf{N}_{\gamma}(Z+T)^{\top}\mathbf{c}'\right)^2 dZ. \tag{17}$$

This can directly be formulated into

$$\int \mathbf{N}_{\gamma}(Z)dZ \int \mathbf{N}_{\gamma}(Z)^{\top}dZ\mathbf{c} = \tag{18}$$

$$\int \mathbf{N}_{\gamma}(Z)dZ \int \mathbf{N}_{\gamma}(Z+T)^{\top}dZ\mathbf{c}'. \tag{19}$$

Again, the integrals can be computed using Gaussian quadrature. The Kalman filter hence acts as a low-pass filter on the coefficients. As the ground is assumed to be static, we set a low variance to the state coefficients and increase the variance with increasing distance.

Instead of projecting onto a static node vector, one can also shift the node vector with the ground plane. Then, however, one has to deal with inserting new nodes and removing nodes at the endpoints.

Prior knowledge about change in camera pitch and camera height is also modeled in the prediction step. Such knowledge can either be estimated using ego-motion or applying the robust v-disparity approach in the vehicle vicinity. In our experiments, we use the second approach and estimate the camera height and the camera pitch angle in the vehicle vicinity using the v-disparity approach. We update the state vector with the calculated height and tilt angle and allow only low variance as we assume these parameters to be accurate in the vehicle vicinity (up to 15 m).

IV. Free-Space Computation

In this section, we describe our approach for computing the free space in front of a vehicle. The computation of the free-space computation has two main goals.

1) Find the distances to the closest objects.
2) Find the road-surface segmentation.

While finding the distance to objects aims at navigating the car or triggering safety systems, the second objective is probably of the same importance. It is crucial for the road surface estimation task described in the first part of this paper. The reason for this is that measurements on vehicles and other objects in crowded scenarios influence the B-spline curve and that the resulting curve estimation may become unstable under such scenarios. Therefore, only 3-D measurements in the free space are used for the spline estimation, neglecting all stereo measurements on objects.

First, we present a literature overview describing different approaches to free-space computation and motivate our choice of free-space computation in Section IV-A. We describe the ideas presented in [11], on which our algorithm is based, in more detail in Section IV-B. Simultaneously, we describe how the nonplanar road-surface representation can be used for the free-space computation and subsequently focus on our proposed changes and extensions.

A. Review of Free-Space Algorithms

The computation of free space is an important issue in the autonomous robot domain. The motion-planning problem implies the autonomous displacement of a robot from one place to another while avoiding collisions with obstacles on its way. For this purpose, occupancy grids are built [12], and the free space is obtained by analyzing the occupancy likelihood of the grid cells (e.g., [13]). The literature on robot motion planning and occupancy grids is quite extensive, and an overview of the state of the art is given in [14].

Occupancy grids are also used in the automotive environment. The main difference is that robots usually maintain a global grid, whereas vehicles only build a local grid from the

current ego-position. In [15], stereo measurement is used to build an occupancy grid. Free space is obtained by applying a threshold to the occupancy likelihood of the cells.

In [16], free space is computed independently of evidence grids by applying inverse perspective mapping.

In [17], stochastic occupancy grids are computed based on stereo information. Stereo is integrated over time to reduce the disparity uncertainty and improve the accuracy of the occupancy grids. Free space is obtained by applying dynamic programming to a polarlike representation of the occupancy grid.

The algorithm presented in [11] computes a globally optimal solution to the road/obstacle boundary. The authors proposed energy-combining measurements on objects, as well as measurements on the road surface. The maximum of this energy is found by dynamic programming. To our knowledge, this is the first time that measurements on the road surface in the free space directly contribute to the road/obstacle boundary estimation. We use the basic idea of this approach and extend it to use a B-spline representation of the road surface, instead of the planar ground assumption. We also show how to integrate direct disparity measurements next to edge directions, as originally proposed in [11].

B. Free Space Using Dynamic Programming

The basic procedure to find the road-obstacle (free space) boundary via dynamic programming (see [11] and [17]) is given as follows:

1) Estimate the road surface orientation parameters.
2) Calculate a disparity-matching score.
3) Find a consistent road-obstacle boundary.

In our case, the road-surface orientation parameters are given by the B-spline $B(Z)$, where Z encodes distance, and $B(Z)$ encodes the height of the road surface in the coordinate system of the moving observer. To distinguish between image pixels on the road surface and pixels on obstacles, the expected disparity of the road surface for an image pixel has to be known. Under the common assumption of no road bank angle, the disparity of the road surface for an image row v is constant and depends on the relative height of the surface. We denote the disparity of the road surface for an image row v by $d(v)$, which is given by

$$d(v) = \text{disparity of mapped road at image row } v. \tag{20}$$

This equation is linear if and only if the ground plane is planar, yielding the v-disparity equation.

The first goal of a free-space algorithm is to find the distance to the bounding obstacles. In image space, this results in finding the disparity value d of the obstacles that bound the free space. The disparity of the obstacle becomes the same as the disparity value of the road surface in the image row, where the footprint of an obstacle touches the ground surface. A free-space algorithm needs to know in which image row obstacles of disparity d touch the ground plane. We denote this image row $v(d)$ as

$$v(d) = \text{v-coordinate of the footprint of objects with disparity } d. \tag{21}$$

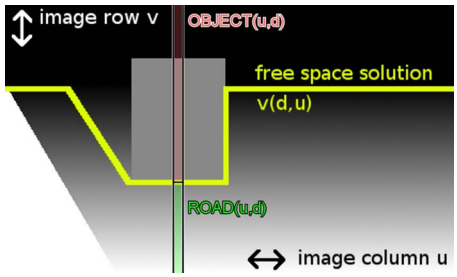


Fig. 6. Example of free-space segmentation on the planar ground. The gray value encodes disparity (white = near, black = far), and the free-space border is shown in yellow. The linear dependence (v-disparity) between disparity d and column v for the ground plane is depicted by the white triangle at the left.

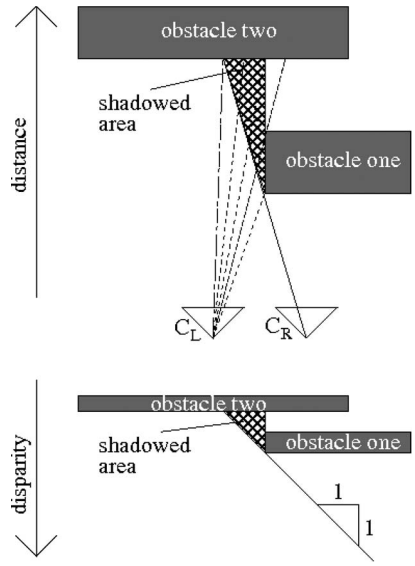


Fig. 7. Shadowed area seen only by one camera. The upper draft shows the distance encoded in world coordinates (X-axis versus Z-axis). The lower draft shows the same top view with distances encoded in disparities (X versus image disparity).

The free space can now be described by its boundary $v(d,u)$, the distance $Z(u)$, or the disparity $d(u)$ of bounding obstacles for every image column u (see Fig. 6).

However, how is the *correct free-space boundary* found? The key idea is to inspect every individual image column u (see Fig. 6 for an example). A matching score is obtained, summing up a score that evaluates the likelihood of pixels belonging to the road surface from the bottom of the image up to the free-space boundary $v(d,u)$. A second matching score evaluates the fit of pixels belonging to objects with disparity d from the free-space boundary in Fig. 6. The total score for an image row u and an obstacle at disparity d becomes

$$\text{SCORE}(u,d) = \text{ROAD}(u,d) + \text{OBJECT}(u,d). \tag{22}$$

The best boundary match is given as the maximal score, i.e.,

$$v(d,u) = \max_d \{\text{SCORE}(u,d)\}.$$

If the maximal score for every image column is independently calculated (winner takes all strategy), the results will be noisy (differ from column to column). This is mainly due to low texture and low information content in some image regions.

Fig. 8. Idea of *plane sweep stereo*. The right image is translated according to a given disparity value, and the consistency of gray values in the left image and the translated right image corresponds to the likelihood of a stereo match. The superposition of the right and left images for different disparity values results in sharp structures in the background (for small disparities), midplane, or foreground (for large disparities).

(a) (b) (c)

Fig. 9. Comparison of free-space computation using the different cost functions described in this paper. The edge-based approach has problems in low-contrast areas on the concrete wall. The disparity-based approach does not pick up enough of the measurements at large distances, yielding to smoothing between the signposts. The combined approach looks most convincing. (a) Free space with edge-based score. (b) Free space with disparity-based score. (c) Free space using the combined approach.

Another reason is stereo occlusion, as shown in Fig. 7: *Obstacle one* is visible in both the left and right cameras; parts of *obstacle two* are only visible in the left camera and occluded by *obstacle one* in the right camera. A disparity measurement for this region is not possible. Hence, for the image columns within the occluded region, the free-space boundary will be undefined.

This is where the idea of dynamic programming comes in [11] and [17]: Deviations in the results between neighboring image columns are penalized to reduce the influence of outliers and to smooth the resulting free-space boundary.

Fig. 7 also shows why disparities are used, instead of world distances, to describe the free-space boundary. If world distances were used, the shape of the *shadowed area* depends on object distances and camera setting. If, on the other hand, the distance is encoded in disparities, the incline always has an angle of 45° (change in u: change in $d = 1:1$). Hence, penalizing disparity differences is preferred over penalizing world distances due to the simple handling of shadowed regions and the direct inaccuracy treatment.

Algorithms that introduce smoothness in a global optimum manner via dynamic programming make use of a d–u (disparity–column) matching score table (see Fig. 10 in the results section). It has the dimension image width by disparity range and encodes, for every image column u and disparity d, the likelihood that d is the disparity of the road-obstacle boundary $v(d)$. For details on finding the optimal boundary, given the d–u table, see [11] or [17].

We will now describe two matching scores, i.e., image edges and disparity values, to construct a d–u table. We will discuss the advantages and disadvantages of both matching scores and

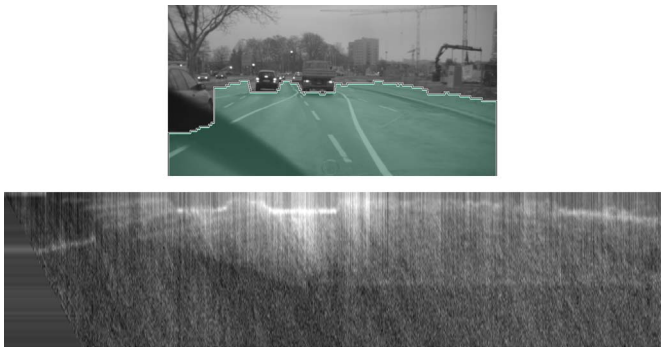


Fig. 10. (Top) Typical free-space segmentation in a challenging environment due to the partly visible windshield wiper and bad illumination in rainy weather. (Bottom) In the d–u table, brightness depicts the likelihood of the free-space boundary; dynamic programming was used to find a smooth path from the left to the right with the largest likelihood.

combine both scores by adding the table entries for both approaches. The better accuracy in free-space computation using the combined approach is verified in the experiments.

C. Image-Based Matching Score Proposed in [11]

In [11], a direct-image-based disparity score is proposed to find the free-space boundary using stereo vision. It is based on the plane sweep idea [18]: For every disparity value d, the pixel (u, v) in the left image is compared with the pixel $(u+d, v)$ in the right image. This corresponds to shifting the right image over the left image or, equivalently, projecting the left and right images onto a plane, which is swept along the Z-axis of the camera coordinate system. The principle is shown in Fig. 8

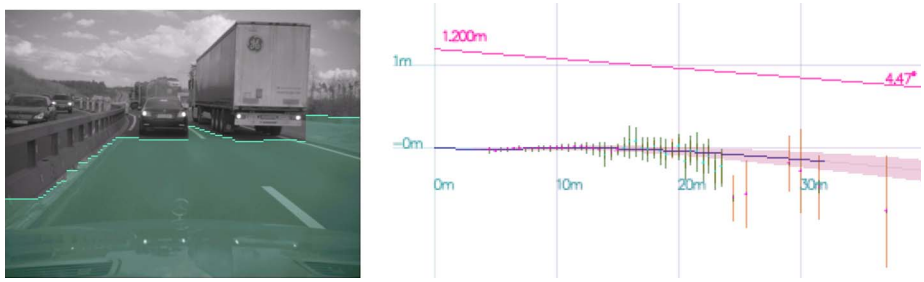


Fig. 11. Due to (left) the free-space segmentation, (right) the height estimate is not influenced by the vehicle that is 16 m ahead. The free space between the vehicle in front and the truck on the right allows for a B-spline estimate up to 30 m, but reliable height measurements are found only up to 15 m.

using plain gray values. Only obstacles with the correct disparity value are in focus, and the gray values in the right and left images coincide. All other regions of the image seem to be out of focus. Collins [18] took this principle and applied an edge filter on the input images to provide a geometric reconstruction of the scene.

The same idea of matching image edges is adopted by [11] to construct the d–u table. Let $E_{L,R}(u,v)$ be the edge direction in the left and right images, respectively, at image position (u,v). The image-based disparity score is then computed as

$$\text{ROAD}(u,d) = \sum_{v=v(d)}^{v_{\max}} w\left(E_L(u,v) - E_R\left(u, v+d(v)\right)\right) \quad (23)$$

$$\text{OBJECT}(u,d) = \sum_{v=v_{\min}}^{v(d)} w\left(E_L(u,v) - E_R(u, v+d)\right) \quad (24)$$

with $w(\arg) = 1$ if $|\arg| <$ threshold and 0 otherwise. $v_{\min}$ and $v_{\max}$ are the upper and lower bounds of the region of interest in the images, respectively. The threshold for computing w is set to an angle of 10°. A too-large threshold yields an oversmooth free-space boundary, whereas a too-small threshold does not accumulate enough edges. Equation (23) counts the number of matched edge directions on the road between the obstacle and the camera. Considering the homography of the road surface between the left and right images improves the results (see [19]). Equation (24), on the other hand, counts the number of matches for the image column u on any potential obstacle with the disparity d.

D. Disparity-Based Matching Score

If disparity maps are computed for pixels in the left image (we use the dense semiglobal matching method (SGM) [20]), the plane sweep approach can be replaced by direct disparity measurements. This has the advantage of implicit robustness, because edge directions may be matched for more than one disparity value. Furthermore, this speeds up the calculation of the disparity score table, because no sweep step is necessary. (Here, we assume that the disparity estimates are already available.) Let (u,v) be an image position and $d_{u,v}$ be the corresponding disparity value. The height $Y(v,d)$ and distance $Z(d)$ of the corresponding world point are computed by stereo triangulation. We define the disparity-based score (with w as before and a threshold of 20 cm and 3 px, respectively) as

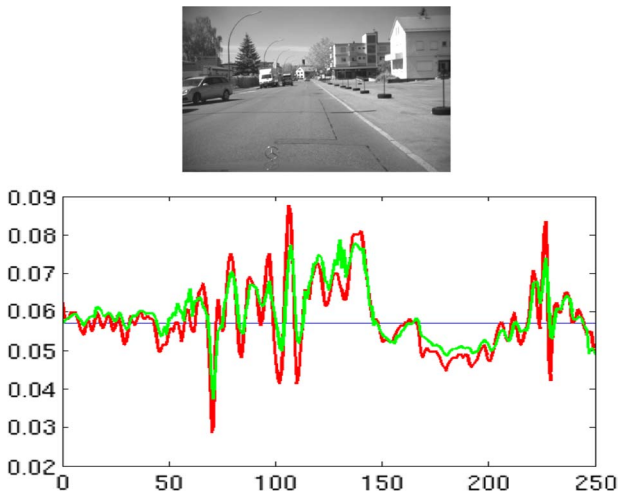


Fig. 12. Comparison of the pitch rate (in radians per frame) obtained using (green) the algorithm described in this paper and (red) the v-disparity approach on the scene shown at the top. Both plots show only small differences.

$$\text{ROAD}(u,d) = \sum_{v=v(d)}^{v_{\max}} w\left(\underbrace{Y(v,d_{u,v}) - Y(v,d(v))}_{\text{relative height}}\right) \quad (25)$$

$$\text{OBJECT}(u,d) = \sum_{v=v_{\min}}^{v(d)} w(d_{u,v} - d). \quad (26)$$

Note that (25) is equivalent to thresholding the height of measurements. Only measurements on the ground surface contribute to the ROAD score. In (26), the distance to possible objects is thresholded. Again, measurements are summed up in the road area on obstacles where the boundary is defined by the footprint of obstacles $v(d)$.

E. Discussion and Combined Approach

The image-based approach yields dense measurements and accurate boundaries. However, measurements may vote for many different disparity values, which may yield unstable results. An edge in the right image matches every edge with the same direction in the corresponding row of the left image.

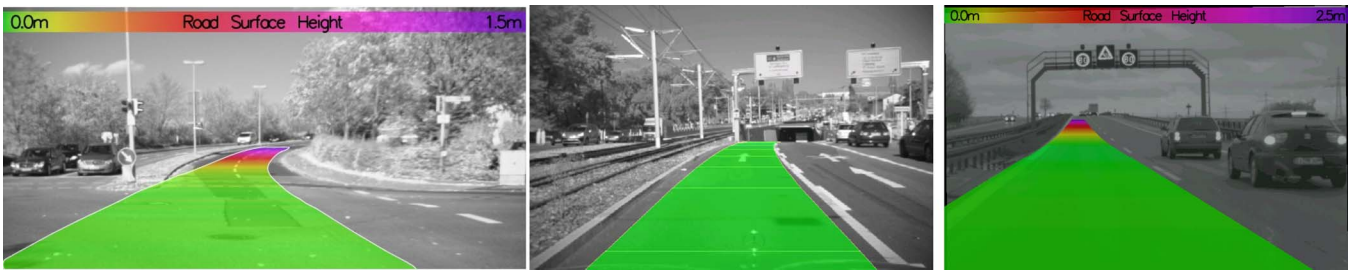


Fig. 13. Examples of the vertical road surface estimation for different scenarios. The proposed approach is able to estimate the height of the road surface up to 75 m in uphill and downhill scenarios. (Right) Autobahn scene with road surface estimation up to 250 m.

Hence, one can be sure that every possible match is found at the cost of many mismatches. One way to reduce the number of mismatches is a coarse-to-fine approach (see [11]). Such a procedure, however, implies that one loses the global optimality of the solution.

The disparity-based score relies on the disparity algorithm to dissolve multiple hypotheses. This implies that the number of mismatches is reduced. However, if a false disparity measurement is present, the correct match is not in the set of disparity solutions. A disparity score is more robust (in terms that matches are unique) but less dense (no disparity available at some pixel positions) than the edge score.

We propose to combine both approaches, i.e., the edge-based matching score and the disparity-based matching score, by adding the single scores. This combines the robustness of the direct disparity measurements and the density of edge information. Experimental results showing the improved free-space calculation are found in the next section.

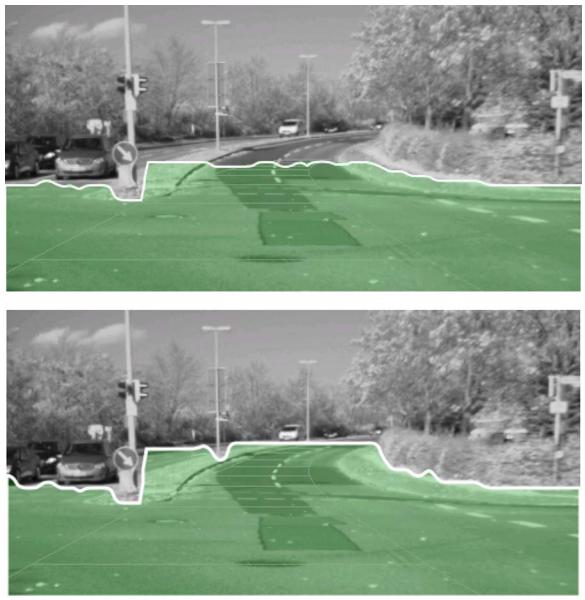


Fig. 14. Comparison of free-space calculation using (top) a planar road surface and (bottom) a B-spline representation of the road surface. The improvement compared with the flat ground assumption, where the uprising road shows up as an obstacle becomes visible.

V. Experimental Results

In our experiments, we use stereo cameras with a baseline of 35 cm and a focal length of 840 px. The image resolution is 640×480 px. Overall, the computation time (Intel CPU) is below 25 ms, using the hardware version of SGM stereo [20].

Fig. 9 compares the proposed cost functions. The edge-based approach has some problems in low-contrast areas of the image; the high contrast on the coming van, for example, influences the free-space estimation, and the free space is too large. The disparity-based approach has not enough disparity measurements at large distances, which yields to a smoothing of the free-space boundary between objects (the signposts). In the combined approach, the result looks most convincing.

In Fig. 10, the free-space segmentation for a challenging scene under rainy conditions is shown. Note that, although only part of the vehicle on the left is visible, the free space is correctly determined using the combination of edges and disparities. A smooth path from the left to the right with a large energy (depicted in gray value) is found in the d–u score table and describes the free space.

The following experiments evaluate the B-spline modeling of the road surface. In our experiments, we use B-splines with degree 3 and five control points (which are equally distributed in the observed interval). The computational time is below 15 ms/frame on standard consumer hardware. Note that, for the ground approximation, only the measurements within the free driving corridor are used. Using all stereo measurements would result in an unstable estimation. This can be seen in Fig. 11, where traffic blocks the view onto the road surface, such that a reliable estimate is only possible up to 15 m.

Fig. 12 compares the pitch angle estimation using the proposed B-spline fit and the original v-disparity approach. In the scene, the road is mainly planar, and we may assume that the v-disparity approach yields trustful results. The plot demonstrates similar results obtained by our algorithm when solving for the road surface parameters and the pitch angle within one Kalman filter (for the experiment, we disabled the v-disparity prediction in the Kalman filter). It can also be seen that the Kalman filter acts as a low-pass filter as long as no dynamic for the pitch angle is modeled. Experimentally, this demonstrates that the road surface and vehicle dynamics can be modeled in a Kalman filter.

Road surface approximations using the B-spline fit are shown in Fig. 13. In the first two scenes, the road surface is modeled up to 75 m in the Autobahn scene (the camera focal length is 1400 px) up to 250 m. The qualitative experimental results demonstrate that modeling the road surface with B-splines is suitable and yields accurate results.

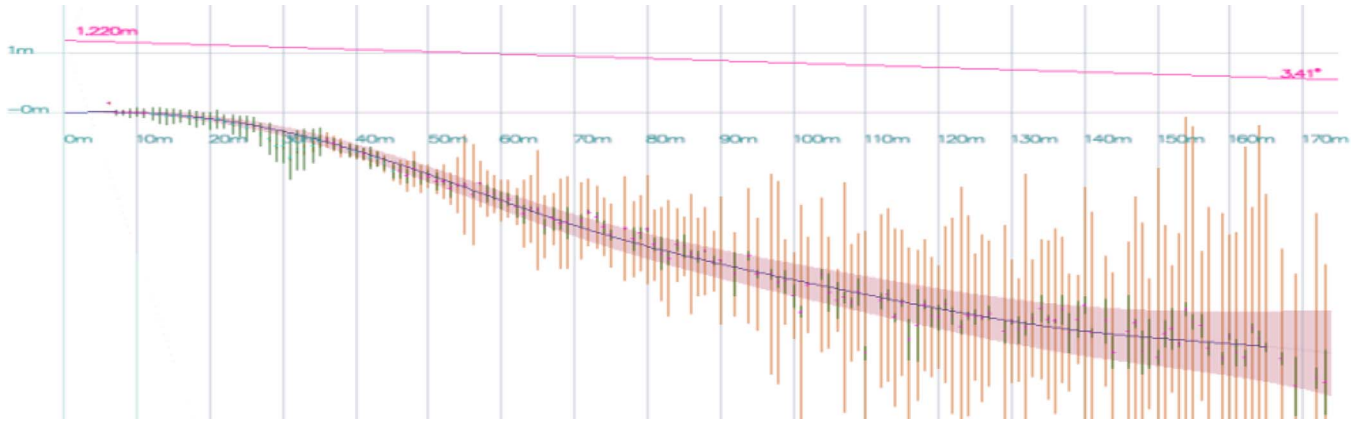


Fig. 15. Height measurements and variances used in the Kalman filter estimation of the B-spline road model for the example in Fig. 1. The camera is at the left. As one would expect, with increasing distance, the measurement variance and the variance of the spline increase.

Fig. 14 now shows the result for the combination of B-spline road modeling and free-space estimation. It also shows the result when modeling the road with a planar surface. This assumption holds for the nearby environment. Then, the road rises to above 1 m within the next 70 m. Using a planar ground assumption, the image-based free-space computation fails because the assumed displacement of the road surface beyond 50 m has an offset of several pixels; the disparity-based approach fails because the height of the road surface beyond 50 m is above any appropriate height threshold. Using the B-spline representation of the road surface, the free space is correctly determined.

An example of measurements and their approximating spline surfaces for the example in Fig. 1 is shown in Fig. 15. The road profile is correctly estimated, and as one expects, with increasing distance, the measurement variance and the variance of the spline increase. The road course ahead falls before it rises again. In contrast to Fig. 14, the free space is estimated to be too large if the ground plane is assumed to be planar.

VI. Conclusion and Outlook

We have introduced an algorithm to robustly track smooth nonplanar road surfaces. In contrast to existing approaches, which are based on a piecewise planar or quadratic ground assumption, we allow for nonplanar ground planes represented by a flexible B-spline curve. Thus, our approach is a generalization of known surface approximations in the literature and bridges the gap between these different approaches.

We have experimentally demonstrated the accuracy of the B-spline representation for the application of free-space estimation. To this end, we have modified a recently published free-space algorithm to make use of the road surface approximation technique and directly use disparity values. Experimental results in planar and undulating terrain have verified the gained availability of free space in everyday traffic.

Some open issues to be addressed in future work are given as follows:

1) evaluating the visibility of surfaces to more intelligently accumulate stereo measurements;
2) integrating prior knowledge in terms of map data in the B-spline estimation;
3) robust M-estimator techniques to reduce the influence of outliers in the B-spline estimation;
4) quantitative evaluation of the B-spline modeling for the road profile.

The presented generalization of the v-disparity approach does not only offer more flexibility in road modeling from image sequences but also enables road modeling using range sensors, such as the Velodyne 3-D laser scanner. The topic of sensor fusion is of broad interest. A possible research topic may be the fusion of stereo and laser scanner distance values within the presented approach for the road surface.

References

[1] H. Helmholtz, *Treatise on Physiological Optics*, (translated by J. P. C. Southall). New York: Opt. Soc. Amer., 1925.
[2] J. Weber, D. Koller, Q.-T. Luong, and J. Malik, "An integrated stereo-based approach to automatic vehicle guidance," in *Proc. 5th Int. Conf. Comput. Vis.*, 1995, pp. 52–57.
[3] R. Labayrade, D. Aubert, and J.-P. Tarel, "Real time obstacle detection on non flat road geometry through 'v-disparity' representation," in *Proc. IEEE Intell. Veh. Symp.*, Versailles, France, 2002, pp. 646–651. [Online]. Available: http://perso.lcpc.fr/tarel.jean-philippe/iv02.html
[4] R. Labayrade and D. Aubert, "A single framework for vehicle roll, pitch, yaw estimation and obstacles detection by stereovision," in *Proc. IEEE Intell. Veh. Symp.*, Columbus, OH, Jun. 2003, pp. 31–36.
[5] N. Suganuma and N. Fujiwara, "An obstacle extraction method using virtual disparity image," in *Proc. IEEE Intell. Veh. Symp.*, Istanbul, Turkey, Jun. 2007, pp. 456–461.
[6] F. Oniga, S. Nedevschi, M. Meinecke, and T. Binh, "Road surface and obstacle detection based on elevation maps from dense stereo," in *Proc. IEEE Intell. Trans. Syst.*, Seattle, WA, 2007, pp. 859–865.
[7] S. Nedevschi, R. Danescu, D. Frentiu, T. Marita, F. Oniga, C. Pocol, T. Graf, and R. Schmidt, "High accuracy stereovision approach for obstacle detection on non-planar roads," in *Proc. IEEE INES*, Cluj Napoca, Romania, 2004, pp. 211–216.
[8] M. Unser, A. Aldroubi, and M. Eden, "B-spline signal processing—Part I: Theory," *IEEE Trans. Signal Process.*, vol. 41, no. 2, pp. 821–833, Feb. 1993.
[9] G. Welch and G. Bishop, "An introduction to the Kalman filter," Univ. North Carolina, Chapel Hill, Chapel Hill, NC, 1995. Tech. Rep.
[10] L. Matthies and S. A. Shafer, "Error modeling in stereo navigation," *IEEE J. Robot. Autom.*, vol. RA-3, no. 3, pp. 239–248, Jun. 1987.
[11] S. Kubota, T. Nakano, and Y. Okamoto, "A global optimization algorithm for real-time on-board stereo obstacle detection systems," in *Proc. IEEE Intell. Veh. Symp.*, Istanbul, Turkey, Jun. 2007, pp. 7–12.
[12] A. Elfes, "Sonar-based real-world mapping and navigation," *IEEE J. Robot. Autom.*, vol. RA-3, no. 3, pp. 249–265, Jun. 1987.
[13] J. Miura, Y. Negishi, and Y. Shirai, "Mobile robot map generation by integrating omnidirectional stereo and laser range finder," in *Proc. IROS*, 2002, pp. 250–255.

[14] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*, ser. Intelligent Robotics and Autonomous Agents. Cambridge, MA: MIT Press, 2005.
[15] D. Murray and J. J. Little, "Using real-time stereo vision for mobile robot navigation," *Auton. Robots*, vol. 8, no. 2, pp. 161–171, Apr. 2000.
[16] P. Cerri and P. Grisleri, "Free space detection on highways using time correlation between stabilized sub-pixel precision IPM images," in *Proc. ICRA*, 2005, pp. 2223–2228.
[17] H. Badino, U. Franke, and R. Mester, "Free space computation using stochastic occupancy grids and dynamic programming," in *Proc. Workshop Dyn. Vis. ICCV*, Rio de Janeiro, Brazil, Oct. 2007.
[18] R. T. Collins, "A space-sweep approach to true multi-image matching," in *Proc. CVPR*, Jun. 1996, pp. 358–363.
[19] T. Williamson and C. Thorpe, "Detection of small obstacles at long range using multibaseline stereo," in *Proc. IEEE Int. Conf. Intell. Veh.*, 1998, pp. 311–316.
[20] S. Gehrig and U. Franke, "Improving stereo sub-pixel accuracy for long range stereo," in *Proc. IEEE 11th ICCV*, Rio de Janeiro, Brazil, Oct. 2007, pp. 1–7.

Andreas Wedel was born in Siegburg, Germany, in 1981. He received the B.Sc. degrees in mathematics and computer science and the M.Sc. degree in computer vision from the University of Bonn, Bonn, Germany, in 2003 and 2006, respectively. He is currently working toward the Ph.D. degree in computer science with the Computer Vision Group, University of Bonn.

Since October 2005, he has been with the Environment Perception Group, Daimler Research, Sindelfingen, Germany, and the Computer Vision Group, University of Bonn. His research interests include motion analysis, segmentation, and modeling of the vehicle environment.

Hernán Badino received the M.Sc. degree in software engineering from the National Technological University, Córdoba, Argentina, in 2002 and the Ph.D. degree from the Goethe University Frankfurt, Frankfurt, Germany, in 2008.

He was with the Environment Perception Group, Daimler Research, Sindelfingen, Germany. He is currently with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA. His research interests include ego-motion from stereo imagery and the development of stereo-vision algorithms for real-time detection and tracking of objects.

Clemens Rabe was born in Giessen, Germany, in 1979. He received the Diploma degree in computer engineering from the University of Applied Sciences Wuerzburg, Wuerzburg, Germany, in 2005. He is currently working toward the Ph.D. degree in computer science with the University of Kiel, Kiel, Germany, in cooperation with the Environment Perception Group, Daimler Research, Sindelfingen, Germany.

His research interests include stereo vision, motion analysis, and traffic scene understanding.

Heidi Loose received the Diploma degree in computer science from the University of Tübingen, Tübingen, Germany, in 2007. She is currently working toward the Ph.D. degree in engineering with the University of Karlsruhe, Karlsruhe, Germany.

Since April 2007, she has been with the Environment Perception Group, Daimler Research, Sindelfingen, Germany. Her research interests include lane recognition and road modeling.

Uwe Franke received the Ph.D. degree in electrical engineering from the Technical University of Aachen, Aachen, Germany, in 1988.

Since 1989, he has been with Daimler Research, Sindelfingen, Germany, working on vision-based driver-assistance systems. He developed lane-keeping and Daimler's lane-departure warning systems and has been working on sensor fusion and truck platooning. Since 2000, he has been the Head of the Environment Perception Group, Daimler Research, where he concentrates on vision for increased traffic safety. Continuously working in the field of intelligent vehicles for 19 years, he is one of the most experienced experts in the world. His recent work is on optimal fusion of stereo and motion, which is called 6D-Vision and scene flow. His special interest is image understanding in complex city scenarios and on real-time stereo analysis.

Dr. Franke was a Program Chair of the IEEE Intelligent Vehicles Conference in Versailles, France, in 2002.

Daniel Cremers (M'09) received the B.S. degrees in mathematics and physics and the M.S. degree in theoretical physics from the University of Heidelberg, Heidelberg, Germany, in 1994 and 1997, respectively, and the Ph.D. degree in computer science from the University of Mannheim, Mannheim, Germany, in 2002.

He was a Postdoctoral Researcher with the University of California, Los Angeles, for two years and a Permanent Researcher with Siemens Corporate Research, Princeton, NJ. Since October 2005, he has been the Head of the Research Group for Computer Vision, Image Processing, and Pattern Recognition, University of Bonn, Bonn, Germany.

Prof. Cremers served on the program committees of the European Conference on Computer Vision, the IEEE Conference on Computer Vision and Pattern Recognition, and numerous other conferences and workshops. He is a regular reviewer for many major computer vision journals. He serves as a proposal reviewer for the German Science Foundation, the U.S. National Science Foundation, the Israel Science Foundation, the Austrian Science Foundation, and the Hungarian Science Foundation. He has received several awards for his publications.