Depth from Perspective Transformations

Nils Einecke, Sven Rebhan, Julian Eggert

2008

Depth from Perspective Transformations

Nils Einecke, Sven Rebhan, Julian Eggert

***Abstract*— All binocular depth estimation algorithms need to apply some kind of matching process for correspondence search. Unfortunately, this search process is difficult because of ambiguities. One possibility to reduce ambiguities is to use 3-D models of the scene geometry. In this paper we interpret the scene as a composition of basic parameterizable surfaces. We present a general way of deriving formulas for perspectively mapping such surfaces from one stereo camera image to the second one. For estimating the model parameters we perform a search in the parameter space, which is guided by the error between the mapped and the original view. By means of the found model parameters depth values can be extracted. For searching the model parameter space we use the Hooke-Jeeves optimization which does not need an explicit gradient formulation and hence constitutes an easy way of circumventing the complex gradient formulas.**

I. Introduction

A dense and accurate depth map is a prerequisite for scene analysis, autonomous system navigation and object manipulation. Depth estimation and related approaches constitute a vast field of algorithms: ”Shape-from-X” approaches (like Shape from Texture [6] or Shape from Shading [19]), depth from motion [4], depth feature learning [13], 3-D model generation [16] and stereoscopic depth [15], to name just the most prominent ones.

There are two main approaches to stereoscopic depth: variational and correlation based methods. Variational methods [2], [17] minimize a certain energy function to calculate the depth of a scene. The energy function describes the error by means of comparing the gray values of the pixels in one image with their counterparts in a second image which depends on the depth. The main advantage of variational approaches is the possibility to directly introduce additional constraints. For example a smoothness constraint is often used as this leads to a ”fill-in-effect” in areas where no correspondence information is available. High computational effort and weaknesses with large displacements (for rectified stereo cameras, this is equivalent to large disparities) are two drawbacks of the variational approaches. Large displacements are difficult to handle, because the number of suboptimal local solutions increases with the displacement range. To some extent this effect can be alleviated using coarse-to-fine strategies, but this introduces additional parameters that have to be tuned in order to arrive at reasonable results.

Correlation based depth algorithms are the standard approach for extracting depth information. The main idea dates back to 1976 [12]. Inspired by their research of the human visual system Marr and colleagues proposed that depth information can be extracted from stereo cameras by finding corresponding points. The difference in the position of the corresponding points is directly coupled to the depth. To find these correspondences, patchwise correlations of the two camera images are calculated. The main advantages of correlation based depth algorithms are simplicity and speed. The drawback is the break down in homogeneous regions or in repetitive structures as correlations here lead to multiple or uncertain correspondences, known as the aperture problem. One way of circumventing this is to use a resolution pyramid [1]. By searching correspondences in the images at different scales, the aperture problem can be reduced. Unfortunately, correspondences found at a coarser scale are less accurate.

Another way of tackling the aperture problem is to incorporate model knowledge. This is done either in a postprocess to improve the generated depth maps or recently also in the correlation process itself. Algorithms that incorporate models into a postprocess try to fit surfaces into the depth maps. By this outliers are removed and unlabeled areas are filled. Most commonly planes are fit into the depth maps, sometimes also coupled with an image segmentation step [5]. The problem with this kind of approaches is that its accuracy crucially depends on the quality of the underlying depth algorithm. Here the algorithms incorporating models at the correlation level, i.e. in the matching process that determines the correspondences, have the advantage over algorithms incorporating models in the postprocessing.

Incorporating models in the matching process is rather new in the field of stereo depth estimation. Currently, the prevailing approach is that of extending the standard matching with a homography transformation constraint [8]. Instead of determining the translation between corresponding patches the parameters of the homography transformation are estimated, such that expected patch correspondences from one to another camera can be estimated. Actually, the homography transformation is used for mapping views of planar 3-D surfaces between arbitrarily oriented cameras, meaning that the underlying model is planar. The correspondence search with planarity constraints enhances the accuracy and allows to map larger parts of a scene at once, which in turn reduces the aperture problem. Unfortunately, the homography transformation is only defined for planar surfaces. As far as the authors know, there are currently no attempts to find a way to be able to use more complex surface models.

In this paper, we present a general way for deriving

N. Einecke is with the Honda Research Institute Europe, email: nils.einecke@honda-ri.de

S. Rebhan is with the Honda Research Institute Europe, email: sven.rebhan@honda-ri.de

J. Eggert is with the Honda Research Institute Europe, email: julian.eggert@honda-ri.de

formulas for transforming the views of a surface between two camera images based on parameterizable surface models. Furthermore we present an algorithm that can estimate the model parameters and calculate depth maps from these model parameters. Note that the transformation formulas have to be derived for each model. Here we show this at the example of a planar and a spherical model. With this framework we are able to map various parameterizable surfaces between camera views and by this overcome the limitations of the commonly used homography transformation which is based on a planar model. In fact the planar model (homography) is just a special case of our approach. With the derived formulas we are able to replace the standard correlation matching for finding correspondences with a surface parameter estimation process. This is a straightforward extension from point based correlation, over edge matching and area matching to real 3-D patch (surface) matching. Our algorithm determines the surface model parameters for a given transformation formula and uses them to generate depth maps. For parameter estimation we use the Hooke-Jeeves [9] optimization method. As this optimization does not need an explicit formulation of the gradient, it is very easy to replace the transformation formulas. It removes the necessity of deriving the complex gradient formulas or to rely on approximations of the gradient. This is an advantage over direct approaches [3], [7] that use a Taylor approximation of the image gradient in a KLT-like [11] fashion which is necessary for a gradient descent in the surface's parameter space.

Our algorithm for surface parameter estimation and depth construction consists of three main steps:

1. Segment one image into regions that belong to one parameterizable surface model.
2. Estimate the model parameters using the Hooke-Jeeves optimization.
3. Generate the depth map by means of the estimated model parameters.

First one camera image has to be partitioned into multiple regions, each obeying one surface model. Currently, we use a simple segmentation algorithm to extract homogeneous regions as these are likely to contain continuous surfaces. In the second step the surface model parameters have to be estimated for each region by performing a searching in the respective model's parameter space. As described above, we've decided to use the Hooke-Jeeves optimization for parameter estimation. This has several advantages which will be discussed later. The last step constitutes the calculation of the depth maps from the estimated surface model parameters. Recently, a similar approach [7] using a planar assumption was shown to be very accurate and efficient in constructing 3-D models from multiple camera images. Here we concentrate on depth map estimation, the extension from planar to other surface models and give a detailed algorithmic explanation and analysis. To show the applicability of our framework, we introduce a special instance of our algorithm that tackles the aperture problem in homogeneous regions by assuming that homogeneous regions are most likely to be planes.

The paper is structured as follows: In section II we discuss the derivation of the surface mapping formulas at the example of planes and spheres. The equations derived describe the transformation of the left camera image into the right camera image given a set of surface parameters. In section III we describe our algorithm that exploits the derived mapping formulas for generating depth maps. We describe the three sub-steps of our algorithm: segmentation, estimation of the unknown surface parameters and depth map generation. In the result section IV we analyze our approach by first evaluating the accuracy of the parameter estimation of the planar model under ideal as well as cluttered conditions. Then we show the performance of the algorithm on three example scenes of increasing difficulty. Furthermore, we show some results using the spherical model which overcomes the limitations of the homography. A summary and an outlook concludes the paper in section V.

II. Mathematical Basics

In the following, we derive formulas for transforming surface patches from one camera view to another based on an arbitrary parametric description of the surface patches. In case of planes such a transformation is known as homography. We derive the formulas starting from a different background to motivate the research and usage of other surface models than planes, which the homography is restricted to. In order to make the formulation easier we derive the formulas for a parallel camera setting. However, the approach itself is not constrained to such a setting.

A. Stereo Perspective Projection

In this paper, we consider a parallel stereo camera setting with two cameras, left(L) and right(R), which have the same focal length (just for convenience). Furthermore we have two coordinate systems with the origins in the foci of the two cameras. The perspective projections for 3-D points into these two coordinate systems lying on the CCD chips are

$$\bar{\mathbf{u}}_L = \frac{f}{z'_L}\begin{pmatrix} x'_L \\ y'_L \end{pmatrix} \quad (1)$$

$$\bar{\mathbf{u}}_R = \frac{f}{z'_R}\begin{pmatrix} x'_R \\ y'_R \end{pmatrix} , \quad (2)$$

where $\bar{\mathbf{u}}_L$ and $\bar{\mathbf{u}}_R$ are the perspective projections of $\bar{\mathbf{x}}'_L$ and $\bar{\mathbf{x}}'_R$, respectively. Note that $\bar{\mathbf{u}}_L$ and $\bar{\mathbf{u}}_R$ are two-dimensional chip coordinates, while $\bar{\mathbf{x}}'_L$ and $\bar{\mathbf{x}}'_R$ are three-dimensional world coordinates. Because of the special geometry of a parallel stereo system coordinates from the left coordinate system can be easily transformed into coordinates in the right coordinate system by subtracting the base line b. Hence Eqn. (2) can be rewritten as

$$\bar{\mathbf{u}}_R = \frac{f}{z'_L}\begin{pmatrix} x'_L - b \\ y'_L \end{pmatrix} . \quad (3)$$

For a correspondence pair of $\bar{\mathbf{u}}_L$ and $\bar{\mathbf{u}}_R$ the 3-D coordinates $\bar{\mathbf{x}}'_L$ of the corresponding 3-D world point could be calculated. The other way around, if the depth of a point is known, it can

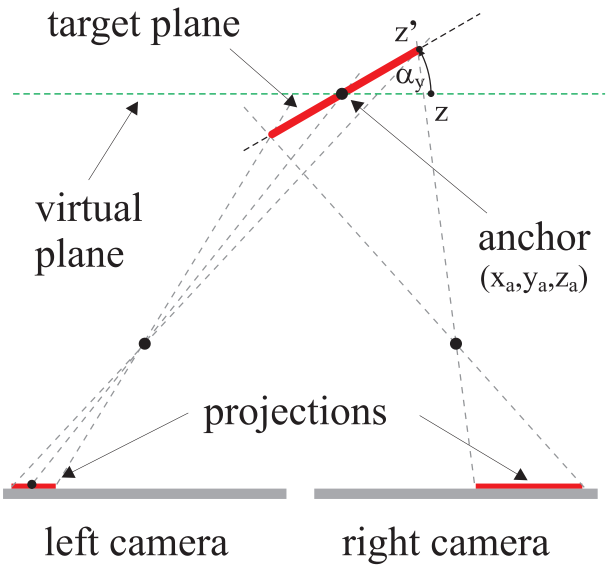

Fig. 1. This image shows the schematic build up of a parallel stereo camera setting and a planar surface, 2-D top view only.

be transformed from one view to the other. By rearranging Eqn. (1) we get

$$x'_L = \frac{u_{Lx} \cdot z'_L}{f} \quad (4)$$

$$y'_L = \frac{u_{Ly} \cdot z'_L}{f} \ . \quad (5)$$

Substituting x'_L and y'_L in Eqn. (3) and simplifying leads to the fundamental equation for mapping parameterizable surface views

$$\bar{\mathbf{u}}_R = \bar{\mathbf{u}}_L - b\frac{f}{z'_L}\begin{pmatrix} 1 \\ 0 \end{pmatrix} \ . \quad (6)$$

By means of the above equation a pixel from the left camera can be transformed to a pixel in the right camera using the known depth z'_L. What is necessary now is a way to describe z'_L by means of a parametric description. In the following we will sketch the derivations for planes and spheres. However, the method is applicable in an analogous way to other parametric surfaces.

B. Plane formulation

In order to derive a formula z'_L that depends on planar parameters we assume that a planar image region (*target plane*) originates from a *virtual plane* parallel to the CCD-chip, which has been rotated at a certain anchor point about the x- and y-axis. Figure 1 shows a schematic top view. The anchor point is specified in world coordinates and denoted with $\bar{\mathbf{x}}_a$. The orientation is specified via rotation angles about the x-axis(α_x) and y-axis(α_y). Note that these two rotations suffice to describe any possible plane orientation. The points $\bar{\mathbf{x}}$ on the original frontoparallel plane are transformed into points $\bar{\mathbf{x}}'$ on the rotated plane by applying the transformation matrix

$$\mathbf{T} = \begin{pmatrix} \cos\alpha_y & \sin\alpha_x\sin\alpha_y & \cos\alpha_x\sin\alpha_y \\ 0 & \cos\alpha_x & -\sin\alpha_x \\ -\sin\alpha_y & \sin\alpha_x\cos\alpha_y & \cos\alpha_x\cos\alpha_y \end{pmatrix} \quad (7)$$

leading to the following formula

$$\bar{\mathbf{x}}' = \mathbf{T}\left[\bar{\mathbf{x}} - \bar{\mathbf{x}}_a\right] + \bar{\mathbf{x}}_a \ . \quad (8)$$

Because the unrotated plane is parallel to the CCD-chip of the camera, the z-coordinate for points on the unrotated plane is always equal to the z-coordinate of the anchor $z = z_a$. Using this, we can rewrite the equation above as

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \mathbf{T}\begin{pmatrix} x - x_a \\ y - y_a \\ 0 \end{pmatrix} + \begin{pmatrix} x_a \\ y_a \\ z_a \end{pmatrix} \ . \quad (9)$$

The depth z' of a point $\bar{\mathbf{x}}$, given the anchor point and rotation angles, then is

$$z' = -(x - x_a)\sin\alpha_y + (y - y_a)\sin\alpha_x\cos\alpha_y + z_a, \quad (10)$$

where $(x - x_a)$ and $(y - y_a)$ can also be expressed with their counterparts on the rotated plane using Eqn. (9):

$$x - x_a = \frac{x' - x_a - (y - y_a)\sin\alpha_x\sin\alpha_y}{\cos\alpha_y} \quad (11)$$

$$y - y_a = \frac{y' - y_a}{\cos\alpha_x} \ . \quad (12)$$

Applying Eqn. (11) and (12) to Eqn. (10) and replacing world coordinates with their projections on the CCD-chips (Eqn. (4) and Eqn. (5)) finally leads to

$$z'_L = z_a\frac{u_{ax}\sin\alpha_y - u_{ay}\tan\alpha_x + f\cos\alpha_y}{u_{Lx}\sin\alpha_y - u_{Ly}\tan\alpha_x + f\cos\alpha_y} \ . \quad (13)$$

With this we have an equation that describes z'_L by the parameters of a planar model. Substituting z'_L in the base Eqn. (6) leads to

$$u_{Rx} = u_{Lx} - \frac{bf\left(u_{Lx}\sin\alpha_y - u_{Ly}\tan\alpha_x + f\cos\alpha_y\right)}{z_a\left(u_{ax}\sin\alpha_y - u_{ay}\tan\alpha_x + f\cos\alpha_y\right)} \quad (14)$$

$$u_{Ry} = u_{Ly} \ . \quad (15)$$

These equations allow for a mapping of the view of a plane from the left camera to the right camera by means of the planar parameters (z_a, α_x and α_y).

C. Sphere formulation

Mapping planes is already known as the homography transformation. Now we will show that it is possible to also map other parametric surfaces using the sphere as an example. We need to formulate z'_L as a function of a parametric model. A sphere in the three dimensional space can be formulated as

$$r^2 = (x - x_a)^2 + (y - y_a)^2 + (z - z_a)^2 \ , \quad (16)$$

where (x_a, y_a, z_a) is the anchor point (center) of the sphere. For more clearness see Figure 2. Again we replace the world

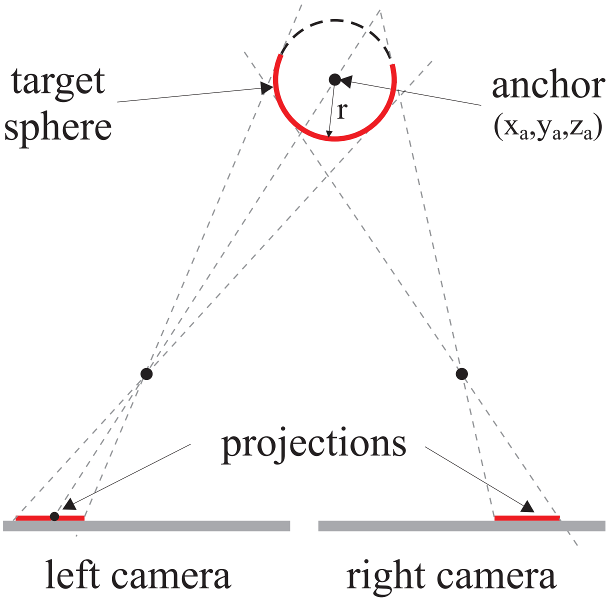

Fig. 2. This image shows the schematic build up of a parallel stereo camera setting and a spherical surface, 2-D top view only.

coordinates with their projections on the CCD-chips and rearrange the formula for z'_L. We get

$$z_{L1,2} = \frac{\mu \pm \sqrt{\mu^2 - \nu\lambda}}{\lambda} , \tag{17}$$

with

$$\begin{aligned} \lambda &= 1 + \frac{u_{Lx}^2 + u_{Ly}^2}{f^2} \qquad (18)\\ \mu &= z_a + \frac{u_{Lx}x_a + u_{Ly}y_L}{f} \qquad (19)\\ \nu &= x_a^2 + y_a^2 + z_a^2 - r^2 . \qquad (20)\end{aligned}$$

At a first glance having two solutions in Eqn. (17) looks disappointing. In fact, a closer look at Figure 2 reveals that using the minus in Eqn. (17) means mapping a sphere (convex structure) and using the plus means mapping a bowl (concave structure). As we are looking for spheres we use

$$z_L = \frac{\mu - \sqrt{\mu^2 - \nu\lambda}}{\lambda} . \tag{21}$$

Inserting this into the base Eqn. (6) leads to

$$\bar{\mathbf{u}}_R = \bar{\mathbf{u}}_L - \frac{bf\lambda}{\mu - \sqrt{\mu^2 - \nu\lambda}} \begin{pmatrix} 1 \\ 0 \end{pmatrix} . \tag{22}$$

This equation allows for a mapping of the view of a sphere from the left camera to the right camera by means of the sphere parameters (z_a, x_a, y_a and r).

III. Depth from Perspective Transformations (DfPT)

In this section we present an algorithm that exploits the derived formulas for mapping parameterizable surfaces for computing disparity maps. As a first step one stereo camera image has to be partitioned into regions (patches) containing continuous surfaces. The main idea of the algorithm is to determine the parameters for each surface patch in the stereo camera images and use these parameters to generate a disparity map.

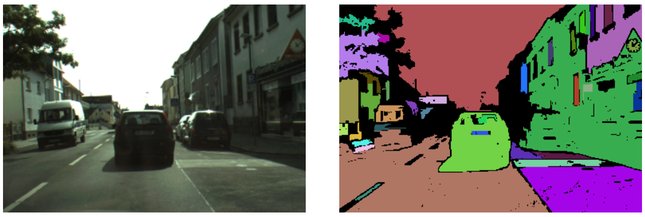

Fig. 3. The image on the right shows a false colored result of a simple Region Growing procedure applied on the left image. This preprocessing can be improved as there are some regions which cluster pixels from different planes, e.g. the car and its shadow are merged into one region. However, Region Growing already provides a sufficiently reliable starting point for the mapping of parameterizable surfaces.

A. Segmentation

In order to be able to apply the Depth-from-Perspective-Transformation (DfPT) algorithm to a whole scene it is necessary to provide masks that specify regions within the scene that contain parameterizable surfaces. It is yet an open question how to do so in a general way. However, for our purposes a segmentation approach based on Region Growing [18] suffices. Here the underlying idea is that large isochromatic image patches are likely to belong to single surfaces.

Figure 3 displays a sample image and its region map in false colors. Note that the patches are not perfectly isochromatic as choosing a color distance threshold of zero would lead to a vast number of small regions due to CCD-chip noise and the illumination and reflection characteristics in real world scenes. The quality of this preprocessing step, i.e. the quality of the masks provided for the DfPT algorithm, can have a strong impact on the quality of the computed surface parameters and hence on the computed disparity values. Region Growing is a quite simple method which can be improved but it is sufficient for the analysis discussed in this work.

B. Parameter estimation

Having a set of masks we can determine the parameters for each surface utilizing the formulas derived in section II. For finding the parameters of a parametric surface the work flow is as follows.

Assume a mask is provided for each surface of the left stereo camera image. First the mask is applied to the left image in order to exclude all other parts of the image except the surface to process. Now by means of Hooke-Jeeves optimization [9] the parameters of the surface model are determined.

Hooke-Jeeves is an iterative optimization algorithm for (fitness) functions based on sampling the landscape defined by the function. Starting from an initial parameter set, an iterative refinement is conducted by sampling parameter sets that are certain step sizes away and taking the best set. If no better solution is found, the step size is reduced. This is repeated until a minimal step size has been reached. Here we use the Euclidean distance between the masked region of the left image and the transformed right image as the fitness function for the Hooke-Jeeves algorithm. This means that the search algorithm tries to find those parameters of a parametric surface that minimize the Euclidean distance between the mapped and the actual view of a surface. The whole procedure is repeated for each mask.

For planes we start the Hooke-Jeeves optimization with α_x and α_y set to zero and z either set to a scene typical value if the kind of scene is known, e.g. for indoor scenes we use generally 2m as starting value, or we use standard correlation techniques for an initial z value. Nevertheless, the Hooke-Jeeves optimization has proven to be quite robust against the starting conditions, even when no resolution pyramid is used.

For spheres we calculate starting parameters by first doing a standard correlation of the patch to get a rough value for z_a. Then we initialize the position of the center (x_a, y_a) by mapping the center of the patch to world coordinates using the guessed depth z_a. Finally the radius r is initialized by mapping the left and the right border of the region mask to world coordinates.

The Hooke-Jeeves optimization has several advantages over the Taylor approximation of the image gradient introduced in [11]. First, it is easy to replace one fitness function with another, i.e. it is straightforward to exchange the mapping functions. Second, there is no need to derive equations that describe the gradient of the fitness function as the Hooke-Jeeves optimization searches the parameter space by means of sampling. Third, the Hooke-Jeeves optimization is numerically very stable, since only simple arithmetic functions are used for the mapping function. Last but not least, an advantage we've discovered is that there is no need to use a resolution pyramid. This is in contrast to Taylor approximation approaches which usually need to use a resolution pyramid because they only incorporate the first derivatives.

C. Calculating disparity or depth maps

After the parameters for the surfaces have been estimated, depth or disparity values can be calculated from them. This is easily done by iterating over all pixels within a mask and using the parametric description for depth z, i.e. Eqn. (13) for planar surfaces or Eqn. (21) for spherical surfaces.

Furthermore additional steps could be taken to improve the disparity maps. In principle most of the standard stereo postprocessing methods are applicable, for example a left-right check could be done by calculating disparity maps for both views and merging them using the residual errors of the Hooke-Jeeves optimization. Furthermore the missing parts in structured parts of the scene can be filled by standard correlation based stereo.

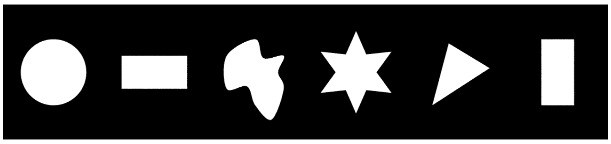

Fig. 4. The six objects used to evaluate our DfPT approach in rendered scenes.

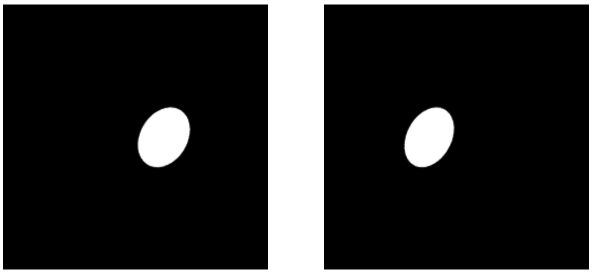

Fig. 5. The left and the right image show the left and the right view of the *disk* object, respectively. In this case the object has a distance of 100 virtual units and is rotated 20° about the x-axis and 40° about the y-axis.

IV. Main Results

In this section we evaluate our approach. For this purpose we've applied our approach to rendered and real world scenes.

A. POVRay generated stereo images

We used POVRay generated stereo images to make a proof-of-concept evaluation of our approach. The idealized character of such images allows to judge the properties of the approach neglecting additional difficulties that arise with real world images, e.g. noise, camera calibration or challenging lighting conditions.

We have chosen six planar objects of different shape for the following experiment. These six objects are depicted in Figure 4. The objects were placed in an empty POVRay scene and rendered for two camera viewpoints in order to simulate a stereo camera system. All objects were rotated about the x- and y-axis in $10°$ steps in the range of $\pm 60°$. Rotating the planar objects leads to projective deformations of the image of the objects in the two views from which our approach draws information about the orientation of planar objects. We determined how accurate our approach estimates the parameters of the planar objects, i.e. α_x and α_y and depth z. The planar objects had a distance of 100 virtual units which corresponds to a disparity of about 65 pixels. The image resolution was 256×256 whereas the objects had a size of roughly 64×64 pixels. Figure 5 shows exemplarily the left and the right view of the *disk* object rotated $20°$ about the x-axis and $40°$ about the y-axis.

For each object and orientation our approach was run 100 times with different random initializations of the depth z. The initial depth value was varied randomly between 60 and 140 whereas the initial values of α_x and α_y were always zero. Figure 6 shows histograms of the errors of α_x, α_y and z over all objects and orientations.

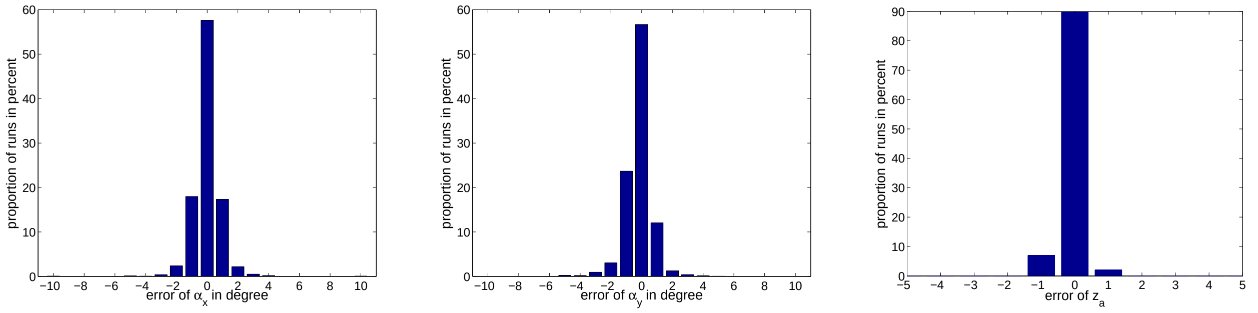

Fig. 6. Histograms showing the error for the three plane parameters α_x, α_y and z applied to different orientations of the six planar objects seen in figure 4. For each orientation 100 runs with random initial conditions for the surface parameters were conducted.

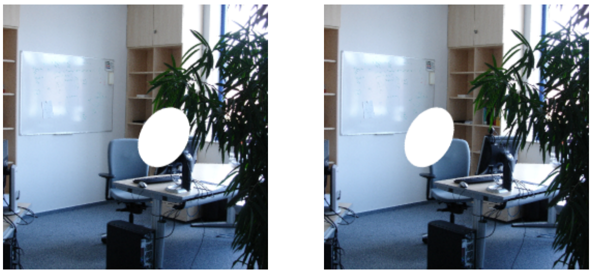

Fig. 7. These two images show the left and the right view of the *disk* object in front of a cluttered background. Parameters correspond to those in figure 5.

This preliminary test shows the validity of our approach. It is the case that under these artificial conditions the orientations of the planes can be estimated quite accurately. In most cases the estimated values do not differ more than one degree from the actual value, even though the objects are small in size and untextured.

In a second series of experiments we tested how well our approach can cope with clutter in the background. For this purpose we introduced a plane at a large distance and mapped an image taken from an office environment onto it (see Figure 7). Note that the clutter background is far away, so that there is almost no disparity for objects in the background. This is quite challenging, as the surrounding of the white test objects changes dramatically, making life harder for the Hooke-Jeeves optimization.

The results for the second test are depicted in figure 8. As expected the performance drops due to the cluttered background. There are several errors in estimating the plane parameters. Disregarding some rare cases the depth z is still estimated very accurately, but the accuracy of the two angles α_x and α_y is lower compared to the uncluttered case. For 78% of α_x and 82% of α_y the error was not larger than $10°$ (and for 60% of α_x and 62% of α_y the error was not larger than $5°$).

Altogether these preliminary tests show that our approach is able to estimate plane orientation and distance quite well. Clutter in the background can cause errors in the estimated values, but the results are very promising.

B. *Standard stereo test images*

The stereo community provides a vast set of stereo images with ground truth, which allow to compare the accuracy of different stereo algorithms. Here we've used the cones scene from the Middlebury data set [14] for further experiments.

In order to evaluate the performance of our algorithm the plane parameters for each homogeneous region have to be transformed into disparity values. This can easily be done using Eqn. (6). It has to be mentioned that the cones data set does not contain demanding aperture problems as the whole scene is well structured. Nevertheless reasonable regions can be extracted by Region Growing due to the colorful objects shown. Figure 9 shows the results of our approach on the cones scene. The first row shows the left camera view and the left ground truth disparity map. In the second row the result of Region Growing is shown at the left side, and at the right side the disparity map generated by our algorithm is shown. The disparity ranges from zero (black) to 55 pixels (white). Each color in the Region Growing map represents one region processed by the algorithm, i.e. a planar surface. This map also shows which parts of the scene are estimated at all. For all black pixels no homogeneous region was found, i.e. for these pixels no plane parameters are estimated and hence no disparity can be calculated. As a consequence the disparity map in Figure 9 shows some gaps. Nevertheless, for the remaining parts of the images, the disparity map is very smooth due to the planar assumption. Especially large regions, like the planks of the fence, are well estimated. If the planar assumption does not hold, e.g. on the cones, the planes are fitted as well as possible, but of course can not represent perfectly the non-linear shape properties of the region. However, the results show that violations of the planar assumption lead to only minor problems and the disparity maps are quite accurate.

A more severe problem is that small regions tend to be less accurately estimated. The main reason for this is that the smaller a region, the less accurate the plane parameters can be estimated due to the smaller resolution. Another reason is that we use a fixed dilatation of the region masks provided by Region Growing. Unfortunately, this means that for smaller regions proportionally more background is incorporated into the parameter estimation. This can be improved and will be

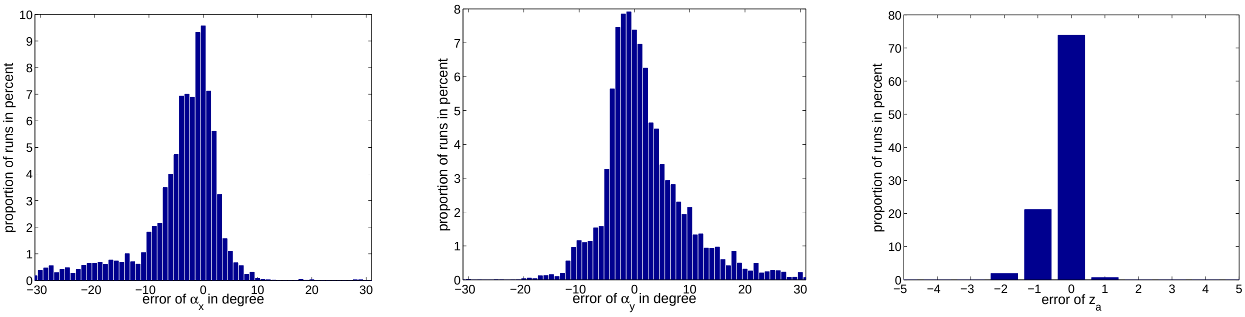

Fig. 8. Histograms showing the error for the three plane parameters α_x, α_y and z applied to different orientations of the six planar objects seen in figure 4, positioned in front of the cluttered background seen in figure 7. For each orientation 100 runs with random initial conditions for the surface parameters were conducted.

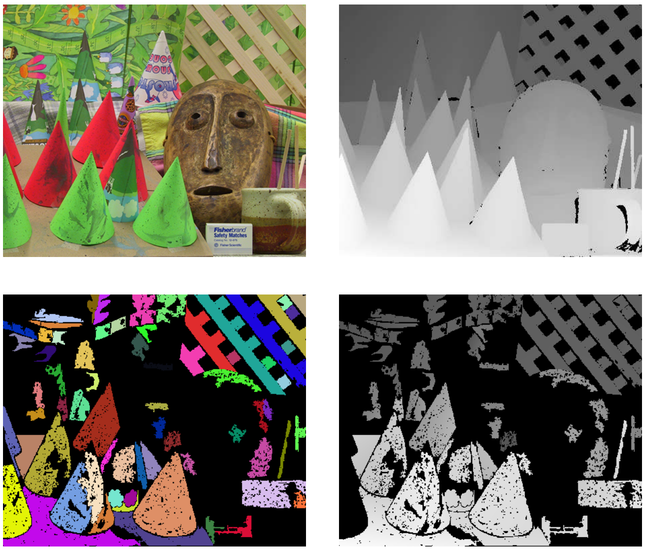

Fig. 9. This figure shows the results of our approach on the Middlebury cones data. In the first row the left and the right image show the left camera view and the left ground truth disparity maps, respectively. The left image in the second row shows a false color image of the segmentation, i.e. all regions for which parameters are estimated. The right image in the second row shows the disparity map generated by our approach. Gray values encode disparity from black (zero disparity) to white (55 pixel disparity).

one focus of future work.

It should be mentioned here that standard correlation based approaches generate denser disparity maps for this scene, because all objects are well structured and the images are taken under optimal lighting conditions. The results in the cones scene show that, considering the accuracy of the disparity of the image patches found by Region Growing, our approach is comparable to state-of-the-art stereo algorithms in case of medium structured ”homogeneous regions”.

In order to show the real advantages of our approach over standard correlation based stereo algorithms the corridor scene was used. This is a rendered scene that consists mainly of homogeneous planes. Here correlation based approaches heavily suffer from the aperture problem, i.e. they are not able to calculate reliable disparity or depth within homogeneous regions.

Figure 10 shows the original image, ground truth disparity and results of our approach in the same arrangement as in the cones scene. The rendered images and the ground truth are from [10]. Here disparity ranges from zero (black) to 25 pixels (white). The bottom right image shows that our algorithm is well suited for the large homogeneous surfaces that this scene is composed of. The disparity map generated by our approach is close to the ground truth and the processed surfaces show hardly any error. The only exception is the right wall which is closest to the virtual camera. Due to a lack of a proper border the algorithm fails to generate accurate disparity values. The problem here is that there is only a left border in both images. Because of this the problem of finding a corresponding plane is underdetermined. Another problem is the ball in the foreground. As the plane assumption does not hold for its surface, our approach is unable to find accurate disparity values for the whole surface. However, the calculated disparity is still quite close to the ground truth. For a better comparison, in figure 11 the left disparity map of a correlation based stereo algorithm and our approach are shown. It can be seen that our algorithm produces more accurate and ”sharper” results, i.e. our algorithm reflects the depth discontinuities very well. In contrast the results of standard stereo look smeared. However, the correlation based algorithm calculates a more dense disparity map, especially regions of high structure like the pictures on the wall or the tiles in the distance. Here our approach fails due to a lack of proper masks. It is very important to note that these gaps in our approach do not appear because of the detailed image structure. If one crafts a mask by hand for the picture on the wall, our algorithm has no problems estimating the plane parameters and calculating the disparities. The problem is just that our rudimentary preprocessing step (Region Growing) is not able to provide masks for planes with high structure.

C. *Real world stereo images*

Although standard stereo test images, like the cones stereo image shown in the last section, are taken from the real world, they are commonly gained under idealized conditions. Especially the homogeneous illumination does not prevail in the real world. Hence we took some pictures from a stereo camera in use on a car for scene analysis.

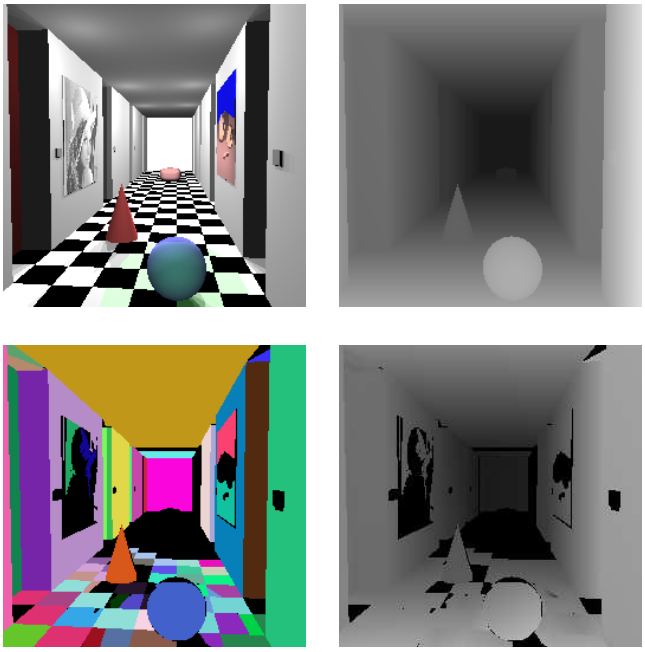

Fig. 10. This figure shows the results of our approach on the rendered corridor data. In the top row the left and right image show the left camera view and the left camera view ground truth disparity, respectively. The left image in the bottom row depicts the result of segmentation in false colors, i.e. each color represents one region. The resulting disparities of our algorithm are shown in the bottom right image. For the disparity images the gray values range from black (zero disparity) to white (25 pixel disparity).

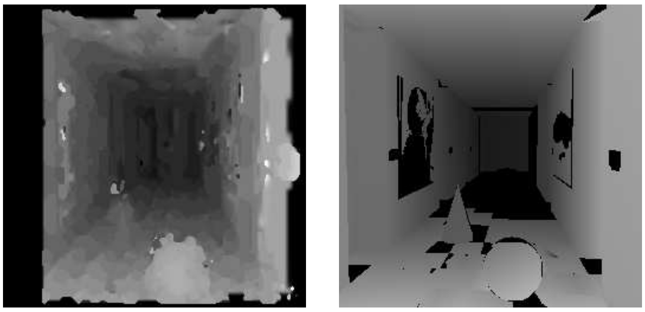

Fig. 11. Comparison of a standard correlation based stereo algorithm (left) with our approach (right).

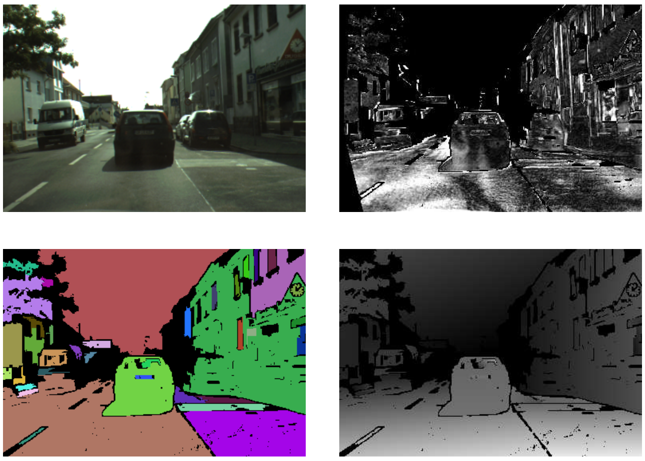

Fig. 12. This figure shows the results of our approach on real world car data. The top left image shows the left camera image from within the car. Below is the region segmentation of that image. At the bottom right the disparity map generated by our algorithm is shown. Due to a lack of ground truth disparity we calculated a pixelwise absolute error between the original left image and the transformed right image (using the disparity map). This error map is shown on the top right. Gray values encode disparity from black (zero disparity) to white (55 pixel disparity) for the bottom right image and error from black (zero error) to white for the top right image. In both the disparity and the error map, pixels not assigned to a region are masked out (value set to zero).

In the top left of figure 12 the left camera view of the stereo system of the car is shown. Note that the camera struggles with overexposure in real world conditions (the sky is completely overexposed). In general, real world scene can exhibit complex lighting conditions like dramatic lighting changes within the scene (overexposure problem), specular reflections and cast shadows. Furthermore, surfaces are not looking exactly the same in the left and the right camera image, e.g. the street in the front has a different appearance in the left and the right image which is partly due to reflections of the car interior on the window and partly due to the reflection properties of the street. Here our algorithm struggles a bit with the border effect, i.e. as the street is quite near, there is some part missing in the other camera's view. This leads to small errors in the parameters and hence to errors in the disparity estimation of the left part of the street. As no ground truth depth estimation is available for the car scene, we can only judge the correctness by means of transforming the right view into the left view by means of the disparity maps. The top right of figure 12 shows the absolute distance between the transformed and the original left view. The error is calculated pixel wise as follows:

$$E_{x,y} = (|I^R_{x,y} - T^R_{x,y}| + |I^G_{x,y} - T^G_{x,y}| + |I^B_{x,y} - T^B_{x,y}|)/3 \quad (23)$$

I and T are the original and the transformed image, respectively. At each (x, y) position the average absolute distance between the RGB values of the original and the transformed image are calculated. This means that the error at (x, y) position is per pixel and channel. Furthermore, regions without mask are set to zero error. The errors are coded from black (zero error / non-masked region) to white (pixel channel error 20 or higher). This means that transformed pixels that differ more than an average of 20 units per RGB channel are displayed in white. These error maps highlight the problem of different illumination under different viewing angles. There are parts within one street region that match badly while some regions match very well. For example the right street region matches well in its left half but badly in its right half. A close look has revealed this cannot be accounted for by a wrong parameter estimation because the disparity gradient has the correct direction. In fact, this shows that our algorithm can cope well with local ambiguities if parameters and depth map for large regions are estimated.

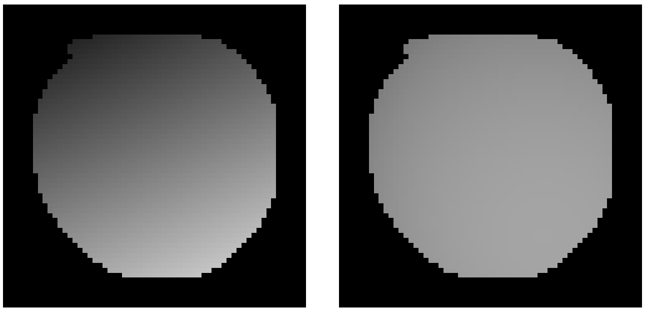

Fig. 13. These two images are depth map close-ups of the ball in the corridor scene. The estimated disparity map using the planar model is shown on the left and the estimated disparity map using the spherical model is shown on the right. The planar model has some problems in fitting, which can be seen in the gradient of the depth values in the left image. Note that gray values code from zero disparity (black) to 25 pixel disparity (white).

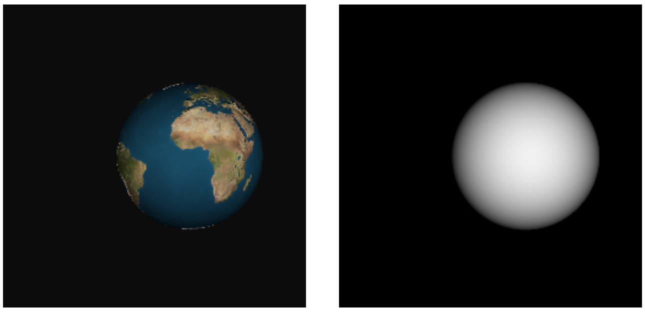

Fig. 14. The left image shows the left camera view of the earth rendered in POVRay. On the right the disparity map generated by our algorithm using the spherical model is shown. Gray values range from black (50 pixel disparity) to white (70 pixel disparity).

D. Using a spherical model

Up to now we have only presented results for the planar model. In this section we will show some results for spheres. Let us first have a look at the sphere of the corridor scene. Figure 13 shows the result of our algorithm using the planar and the spherical model in a close-up. Compared to the results of the planar model, the spherical model leads to much better disparity values.

Furthermore we rendered an image of the earth in POVRay in order to judge how well parameters are guessed for the sphere. Figure 14 shows the left image of the earth and the result of our algorithm using the spherical model. The actual parameters of the earth are 3000, 0, 40000, 6366 for x_a, y_a, z_a and r, respectively. Our algorithm estimates 2949, 31, 40101, 6448 for x_a, y_a, z_a and r, respectively. This shows that it is possible to also estimate sphere parameters quite accurately. Unfortunately, we have no automatic mask generation for spheres yet. One idea is to also use Region Growing and let different surface models compete for each region, selecting the best in the end. This will be part of future work.

V. Conclusions

In this work we've presented a new approach to stereo disparity calculation based on perspective transformations. First, we've showed that by assuming a parameterizable surface model it is possible to derive formulas that enable us to perspectively transform image regions between the two cameras of a parallel stereo camera setting. This general approach has been shown at the example of a planar and a spherical surface model. Hereby we overcome the limitations of the homography transformation, which is restricted to planes.

Based on this we've introduced an algorithm that exploits the newly derived formulas of parametric surface transformation for disparity calculation. The algorithm consists of three main steps. First, the image has to be segmented into regions that belong to one parameterizable surface model. Second, the parameters of the model are estimated using the Hooke-Jeeves optimization method. Third, by means of the estimated parameters a disparity map is generated. By doing several experiments we've shown the high accuracy of our algorithm in terms of parameter estimation and disparity map generation.

For the planar model we've proposed to use Region Growing for the segmentation step. This is based on the idea that isochromatic image regions are likely to belong to one surface. Indeed we've shown that this is well suited for a large variety of scenes. The advantage of this application of our algorithm is that it is complementary to standard stereo because standard stereo is bad in estimating disparities for large homogenous regions.

For future work we plan to extend our approach in order to be able to cope with various types of parameterizable surfaces. One key point that has to be solved here is the problem of segmenting the image into regions of parameterizable surfaces. As this is quite difficult, we also think about implementing a multi-hypothesis system. By applying the optimization for a certain region with different surface models, the best model can be selected by means of the residual errors produced by the single models. Furthermore, future work will concentrate on fusing our approach with standard correlation based stereo algorithms in order to combine the advantages of both approaches.

References

[1] E. H. Adelson, C. H. Anderson, J. R. Bergen, P. J. Burt, and J. M. Ogden. Pyramid methods in image processing. *RCA Engineer*, 29(6), 1984.

[2] L. Alvarez, R. Deriche, J. Sanchez, and J. Weickert. Dense disparity map estimation respecting image discontinuities: a pde and scalespace based approach. *Journal of Visual Communication and Image Representation*, 13(1-2):3–21, 2002.

[3] S. Baker, R. Szeliski, and P. Anandan. A layered approach to stereo reconstruction. In *CVPR '98: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 434–441, Washington, DC, USA, 1998. EEE Computer Society.

[4] J. L. Barron, D. J. Fleet, and S. S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–77, 1994.

[5] M. Bleyer and M. Gelautz. Graph-based surface reconstruction from stereo pairs using image segmentation. In *Videometrics VIII*, volume 5665, pages 288–299, January 2005.

[6] M. Clerc and S. Mallat. The texture gradient equation for recovering shape from texture. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):536–549, 2002.

[7] M. Habbecke and L. Kobbelt. Iterative multi-view plane fitting. In *Vision, Modeling, Visualization VMV'06*, pages 73–80, 2005.

[8] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004.

[9] R. Hooke and T. A. Jeeves. ”Direct Search” Solution of Numerical and Statistical Problems. *J. ACM*, 8(2):212–229, 1961.

[10] P. LeClercq and J. Morris. Assessing stereo algorithm accuracy. In *Image Vision Computing New Zealand (IVCNZ)*, pages 110–115, 2002.

[11] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *International Joint Conference on Artificial Intelligence*, pages 674–679, 1981.

[12] D. Marr and T. A. Poggio. Cooperative computation of stereo disparity. *Science*, 194(4262):283–287, October 1976.

[13] A. Saxena, S. H. Chung, and A. Y. Ng. Learning depth from single monocular images. In *NIPS 18*, 2005.

[14] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1-3):7–42, 2002.

[15] D. Scharstein and R. Szeliski. High-accuracy stereo depth maps using structured light. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2003)*, volume 1, pages 195–202, Madison, WI,, June 2003.

[16] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *CVPR '06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 519–528, Washington, DC, USA, 2006. IEEE Computer Society.

[17] N. Slesareva, A. Bruhn, and J. Weickert. Optic flow goes stereo: A variational method for estimating discontinuity-preserving dense disparity maps. In *DAGM-Symposium*, number 2, pages 33–40, 2005.

[18] Milan Sonka, Vaclav Hlavac, and Roger Boyle. *Image Processing, Analysis, and Machine Vision*. Thomson-Engineering, 2 edition, 1998.

[19] R. Zhang, P.-S. Tsai, J. E. Cryer, and M. Shah. Shape from shading: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(8):690–706, August 1999.