A Comparative Study of Two Vertical Road Modelling Techniques

Konstantin Schauwecker and Reinhard Klette

Computer Science Department, The University of Auckland
Private Bag 92019, Auckland 1142, New Zealand

Abstract. Binocular vision combined with stereo matching algorithms can be used in vehicles to gather data of the spatial proximity. To utilize this data we propose a new method for modeling the vertical road profile from a disparity map. This method is based on a region-growing technique, which iteratively performs a least-squares fit of a B-spline curve to a region of selected points. We compare this technique to two variants of the v-disparity method using either an envelope function or a planarity assumption. Our findings are that the proposed road-modeling technique outperforms both variants of the v-disparity technique, for which the planarity assumption is slightly better than the envelope version.

1 Introduction

Vehicles have become more and more intelligent over the past few years. Nowadays, drivers are supported by a range of helpful *driver assistant systems* (DAS). Some DAS, such as advanced automatic cruise control or parking assistants, only work well if information about the spatial environment is available. This data is gathered, for example, by using radar sensors. The problem with radar is that it only provides a measurement in the vehicle surroundings along a given direction.

It can be expected that future DAS will be more intelligent and thus require a more detailed model of the spatial proximity. This data can be gathered in principle using binocular vision (the human visual system may be cited as a proof). Top-performing *stereo matching* algorithms provide a dense measure for the disparity of most visible pixels. From the resulting *disparity map* we can reconstruct the 3D origin of each pixel and thus receive a detailed representation of the vehicle environment. An intelligent car not only has to gather this data but is also required to "understand" it. In particular, it is important that it recognizes properties of the road such as its geometry in 3D space, surface properties, speed bumps, obstacles on the road, and so forth. Our study is focussing on the road profile, modeled by a geometric manifold.

An accurate road profile helps to identify other vehicles and objects on the road by comparing the height of matched points with the road model: Points that are significantly above the road must belong to obstacles; of course, a road may also be elevated with visible objects next to the road that are below road level, but those objects would not be *on the road*.

Different manifold models may be selected to be fitted to the disparity data obtained for the road profile. Ideally, a road model should precisely match the vertical road profile. The common planarity assumption does not support that.

In this paper we present a new technique for creating a vertical road model, which is based on *B-spline curves* and a *region-growing* process. We evaluate the performance of this method and compare it to the widely used v-disparity approach. We use a common *belief propagation* stereo algorithm for the stereo matching part, but this is not crucial for the processing pipeline or the comparison (because we use it uniformly for both techniques), and it could be replaced by another stereo algorithm.

2 Related Work

The simplest way to model the vertical road profile is to assume that the road is planar and its normal perpendicular to the horizon, as done by Weber et al. [1]. This assumption is known to be inaccurate, and more advanced methods haven been proposed.

The method introduced by Labayrade et al. [2] is based on *v-disparity images*. In this method, the disparity map is first transformed into a new virtual image, by counting the occurrences of each disparity value in each image row and plotting the result. The disparities corresponding to the road surface are likely to be incident with a curve. In [2] this curve is modeled as a a piecewise linear curve, and its segments are detected through a standard *Hough transform*, which delivers a set of best matching straight lines. Those lines are mapped into one polygonal chain by either calculating the upper or lower envelope.

The approach proposed in [3] is based on approximating the road by a three-dimensional quadratic model. The first step in this procedure is to convert the disparity map into a digital elevation map. The quadratic model is then fitted using a region-growing method. First, a small region close to the *ego-vehicle* (i.e., the car where the system is operating in) is selected and used for fitting the first version of the model. This region is then iteratively extended by including matching adjacent pixels; the model is continuously refitted.

In [4], Nedevschi et al. perform an approximation of the road surface by fitting a *clothoid*, which is a polynomial of degree three. The approximation process works by first reconstructing a *lateral view* of the scene from the disparity map. In a next step, a polar histogram is created that counts the number of points near a range of selected polar lines. The angle of the polar line with the maximum of surrounding points will be selected as being the pitch angle for the clothoid curve. The curvature of the clothoid is then detected using a similar histogram.

Wedel et al. introduce an approximation of the road surface in [5] that uses a B-spline curve with equidistant nodes. The control points of this curve are found with a least-squares method, which is not solved directly but embedded into a special *Kalman filter*. The curve is fitted to a region that is believed to match the road and was detected before with a *free-space estimation* algorithm. To improve accuracy, constraints are introduced which require that the height

and the gradient of the curve equals zero at the camera position. Furthermore, solutions with high gradients and curvatures are penalized.

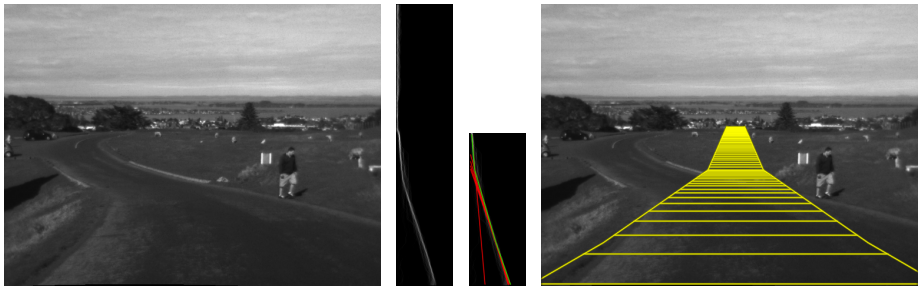
3 *V*-Disparity Images

The previously mentioned v-disparity method appears to be the most popular approach within the set of introduced vertical road-modeling techniques. This might be due to its simplicity. However, [3] criticizes the v-disparity approach by stating that it requires the road to occupy most of the image, and that it is sensitive to changes in roll-angle. Furthermore, the usage of v-disparity images for performing a piecewise linear approximation of the road surface, as done in [2], is not as accurate as other modeling approaches that rely on higher order curves [5].

Nevertheless, the v-disparity method can provide good results on predominantly straight roads without large curvatures [2, 5]. We therefore chose to implement it as a reference system for evaluating our own technique.

Figure 1b shows the v-disparity image we obtained with our implementation for an example of a stereo pair; one image of the pair is shown in Fig. 1a. The v-disparity image has the same height as the input image and its width is equal to the number of possible disparity values. The intensity of a pixel (u, v) in this image represents how often the disparity u occurs in image row v.

Figure 1c shows the best matching lines (red) found with the Hough transform, and their upper envelope (green). We clipped the image to avoid false matches from sections above the road. The decision on whether to use the upper or lower envelope is done with the same method as in [2], which is by comparing the intensity sum of all pixels along both possible envelopes. In Fig. 1d a perspective projection of the resulting envelope function has been overlaid on one image of the input stereo pair.



(a) (b) (c) (d)

Fig. 1: (a) Image of a stereo pair. (b) Corresponding v-disparity image. (c) Lines found by the Hough transform. (d) Perspective projection of the road profile.

4 B-Spline Road-Modeling

Using B-spline curves to model the vertical road profile, as done in [5], allows to model road profiles whose curvature changes its sign. None of the other approaches we discussed is capable of doing this. Thus, if we encounter such a road, those techniques will become largely inaccurate. We have developed a system for approximating the road by a B-spline curve, which uses a different approach from the one presented in [5].

A B-spline curve is fitted in [5] to a set of points extracted by a free-space estimation algorithm. This means that the accuracy of the created road model strongly depends on the used free-space algorithm; its results may be inaccurate if there are difficulties in detecting road boundaries. We propose an alternative approach that is based on a region-growing technique. In general, our method may potentially be more accurate if clear road boundaries are missing.

4.1 Region-Growing

Our method works on the set of 3D-points we obtain, when we reconstruct the 3D-location for all pixels of the disparity map. For initializing the region growing process, we select a small region of points close to the ego-vehicle, for which we have a high confidence that they are part of the road. We increase our confidence by only selecting points that do not deviate much from the model of the previous frame. A B-spline curve is then fitted to those points and used for finding further road points. This will increase the set of points we "understand" to be part of the road, and we use the enlarged set to fit a new and presumably more accurate curve. The selection of points and fitting of a new curve is repeated, either for a predefined number of iterations, or until a termination criterion is met.

Our region-growing method differs from the one proposed in [3], in that we do not require an elevation map. Furthermore, we allow the selection of new points that are not adjacent to already selected ones in the disparity map. This drastically reduces the number of iterations required, as more points can be selected in a single step compared to [3]. For the tested stereo sequences, less than twenty iterations were necessary for all stereo pairs.

4.2 Least-Squares Fitting

Once a region of identified road points has been selected, we use these points for fitting a uniform B-spline curve. For this task we use the method of least-squares, which has also been used in [3, 5] for model fitting. This means that we try to minimize the error

$$E = \sum_{k=0}^{m} \left(B(z) - P_k\right)^2 = \sum_{k=0}^{m} \left(\sum_{j=0}^{n} N_j(t_k) Q_j - P_k \right)^2 \tag{1}$$

where P_k is in the set of selected road points, $B(z)$ is the wanted B-spline curve, N_j a B-spline basis function, and Q_j is in the set of B-spline control points.

In [5], a solution is found by feeding the least-square equations as a measurement into a special Kalman filter. We cannot use this method because we develop our solution by an iterative process. Applying a Kalman filter to inaccurate intermediate estimates would disrupt the filter state.

We thus determine our solution the ordinary way, using the method of linear least-squares. The subject of fitting B-spline curves is discussed in [6]. To find a solution, we need to formulate our problem as a system of linear equations. In the ideal case, all measurement points lie exactly on the B-spline curve and can thus be expressed in terms of the control points and a matrix A, containing the B-spline basis functions, with

$$P = AQ \quad \text{and} \quad A = \begin{bmatrix} N_0(t_0) & N_1(t_0) & \cdots & N_n(t_0) \\ N_0(t_1) & N_1(t_1) & \cdots & N_n(t_1) \\ \vdots & \vdots & \ddots & \vdots \\ N_0(t_m) & N_1(t_m) & \cdots & N_n(t_m) \end{bmatrix} \tag{2}$$

In the case of linear least-squares, we can transpose the above equation to obtain the *normal equation*

$$A^T AQ = A^T P \tag{3}$$

of our linear system. The above linear system is invertible and can thus be solved by matrix inversion; we have the solution

$$Q = (A^T A)^{-1} A^T P \tag{4}$$

4.3 Error Model

Whether a point should be included in the enlarged region or not, is decided by its vertical distance to the previously fitted curve. For evaluating a new candidate point we need an error model that tells us the maximum distance that is still acceptable. In [3], the error in z- and y-direction is calculated by the formulas [1]

$$z_{err} = \left| \frac{z^2 \cdot d_{err}}{b \cdot f - z \cdot d_{err}} \right| \quad \text{and} \quad y_{err} = \left| \frac{y \cdot z_{err}}{z} \right| \tag{5}$$

where d_{err} is the disparity error, b is the baseline and f is the focal length.

With those two equations we can determine the maximum error from triangulation in y-direction. Experiments with this error model showed that it is not sufficient for our region-growing approach. It is missing the error introduced by the curve fitting, which can cause a displacement of the curve along the z-axis. This displacement can be as large as the the error in z-direction z_{err}.

We have to take this error into account when we decide whether a given point with the z-coordinate z should be considered to be part of the road or not. To do this, we not only examine the curve at position z but also at $z + z_{err}$ and $z - z_{err}$. If the vertical distance of a candidate point to any of the three selected curve points is less than y_{err} plus a tolerance threshold s, then the point will be considered to be part of the road and added to the selected region.

[1] The original equations contain the camera height, as the origin is assumed to be on the road. We use the camera position as origin and thus do not require this variable.

4.4 Region-Reduction

Because we allow a large number of pixels to be selected in one iteration, it is "very likely" that pixels outside the road are falsely selected; then those pixel distort the region-growing process. To cope with this problem we exclude some particular pixels from the selected region, even though they meet the criteria of our error model. We call this step *region-reduction* as it counteracts to the region-growing step. For performing region-reduction, we use a set of independent techniques:

Z-Distance Limit. The fitted B-spline curve is accurate for points that are not far from the selected region. The accuracy greatly declines the farther the curve is extrapolated. We thus limit the evaluation of a curve to be not continued beyond a distance d to the farthest point in the current region. This also limits the disconnectivity in z-direction: The distance between a new point and its closest selected neighbor can never exceed d.

Connectivity Constraint. We enforce connectivity in the xy-plane by treating our selection masks as a binary image, and use a flood-fill algorithm to extract a connected subset. The seed is selected as any point from the initial region. Points that are not in the extracted subset will be removed.

Density Constraint. The flood-fill algorithm does not remove erroneous regions if they share just a single pixel with the current road region. Therefore, we eliminate such connecting pixels by evaluating the number of selected pixels in the neighborhood of a pivot pixel. Pixels that have less than the required number of neighbors, will be removed. A rectangular neighborhood is evaluated in constant time based on the use of *integral images* [7].

SSR Threshold. If some erroneous points are selected, those points will have a small disruptive impact on the curve fitting. This may cause a selection of more and more erroneous points in subsequent iterations. In such cases we need to stop the iteration earlier. We perform this decision by calculating the *sum of squared residuals* (SSR). If the SSR per selected pixel exceeds a given threshold value, the iteration will be discontinued.

5 Smoothness Constraint

A smoothness constraint is used in [5] that penalizes high gradients and curvatures because we expect the road to be never extremely steep or curved. For this purpose, two penalizing quantities are introduced in [5], which are based on the squared first and second order derivative of the B-spline curve. The penalizing quantities are embedded in the update equation of the employed Kalman filter.

As we do not make use of a Kalman filter, we cannot apply the same method. We therefore suggest an alternative approach, which embeds the constraint equations into the least-squares linear system. To achieve this, we have to introduce two different penalizing quantities based on the absolute derivatives

$$w_1 \int |B'(z)|dz \quad \text{and} \quad w_2 \int |B''(z)|dz \tag{6}$$

where w_1 and w_2 are two factors controlling the influence of the constraint.

If we insert the B-spline equation into those quantities we obtain

$$w_1 \int |B'(z)|dz = w_1 \int \sum_{j=0}^{n} |N_j'(t_k) \cdot Q_j| dt_k = w_1 \sum_{j=0}^{n} |Q_j| \underbrace{\int |N_j'(t_k)| dt_k}_{I_1} \quad (7)$$

$$w_2 \int |B''(z)|dz = w_2 \int \sum_{j=0}^{n} |N_j''(t_k) \cdot Q_j| dt_k = w_2 \sum_{j=0}^{n} |Q_j| \underbrace{\int |N_j''(t_k)| dt_k}_{I_2} \quad (8)$$

Because we are using uniform B-splines, all basis functions N_j are shifted copies of each other. This means that the integrals over their absolute gradient and curvature I_1 and I_2 will both be constant. If we replace the products of the integrals and weighting factors $I_1 w_1$ and $I_2 w_2$ with a new weighting factor w_s, we can unify both equations in one single formula. Further, we can eliminate taking the absolute values if we shift the curve along the positive y-axis, such that all points are positive. We thus receive a simplified penalizer

$$w_s \sum_{j=0}^{n} Q_j \quad (9)$$

We want to find a solution where this sum becomes a small value. This is equivalent to finding a small value for the squared entity

$$\left(\sum_{j=0}^{n} w_s Q_j - 0 \right)^2 \quad (10)$$

If we compare this expression with Eq. (1), we realize that it has the same structure as the inner term. This inner term calculates the contribution of one control point to the overall error. If we interpret Eq. (10) in this context then 0 would be a measurement point and w_s the value of all B-spline basis functions. We can thus incorporate the smoothness constraint into the least-squares system by using a new matrix $\widehat{A}$ and point vector $\widehat{P}$, which both contain one additional row, where

$$\widehat{A}_{m+1} = \begin{bmatrix} w_s & w_s & \cdots & w_s \end{bmatrix} \quad \text{and} \quad \widehat{P}_{m+1} = 0$$

6 Gradient Constraint

We introduce another constraint, which corresponds to the gradient constraint used in [5], and penalizes solutions for which the first derivative at the origin is nonzero. As the ego-vehicle is standing flat on the road surface, we assume that the gradient of the road equals 0 at the camera position.

The derivative of a B-spline curve can be calculated by replacing the individual basis functions with their derivatives. We can thus implement the new

constraint by adding another row to the system matrix $\widehat{A}$ that contains the value of the derived basis functions at position 0, multiplied with a weighting factor w_g to control the influence of the constraint. Furthermore, we need to add a new point with a value of 0 (the desired gradient) to the point vector. The new row of the resulting matrix $\widetilde{A}$, and the new point of the point vector $\widetilde{P}$, are thus as follows:

$$\widetilde{A}_{m+2} = \begin{bmatrix} w_g N_0'(0) & w_g N_1'(0) & \cdots & w_g N_n'(0) \end{bmatrix} \quad \text{and} \quad \widetilde{P}_{m+2} = 0 \tag{11}$$

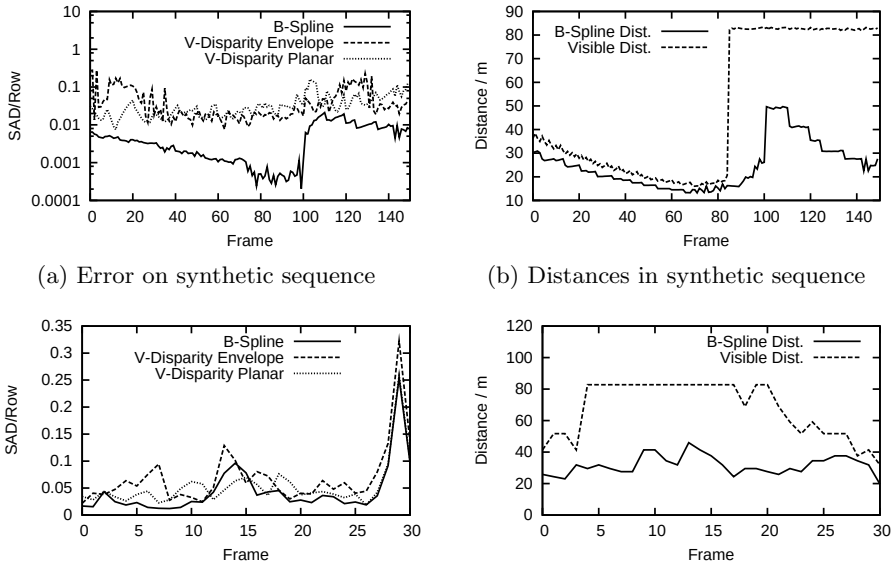
7 Results

To judge the performance of our new road-modeling algorithm we performed a comparative evaluation with two versions of the v-disparity method. The first version matches the method discussed in [2] and creates a polygonal chain, while the second one only selects the best matching line and thus creates a planar model. We tested both variants and our algorithm on the second synthetic driving sequence in Set 2 of EISATS [8]. The used disparities are the provided ground-truth values rounded to the nearest integer, which is the best result we could expect from a stereo matching algorithm which is not aiming at subpixel precision.

We were able to extract the road pixels from the provided ground-truth for the first 150 frames, which gives us a precise measure of the road profile at each image row. For a quantitative evaluation we calculate the *sum of absolute differences* (SAD) per image row between estimated and ground-truth road profile. We perform this evaluation only up to the last image row selected by our region-growing algorithm. This is in favor of the v-disparity approach, which does not set a distance boundary and is more inaccurate with increase in distance.

Figure 2a shows the results we receive for the three cases on a logarithmic scale. Our approach clearly outperforms both variants of the v-disparity method. The envelope-based v-disparity version does not perform any better than the simpler planar version. Figure 2b compares the fitted distance to the maximal distance at which the road is still observable with a minimum disparity. The sudden jump in visible distance is caused by driving over a hill that occludes the road in the beginning of the sequence. This is in favor of the v-disparity approach, which does not set a distance boundary and would otherwise be evaluated until the maximum visible distance.

In a second evaluation we compared the performance of all methods on a real world sequence. The used stereo sequence has been recorded on a hilly and windy road without prominent road boundaries, and should thus present a difficult challenge for any algorithm. We manually extracted the road from 30 frames and used the median disparity for the road pixels in each image row to obtain an estimate of the road profile. Figure 2c shows the comparison of this road profile to the results of the tested road-modeling techniques. Our approach still performs predominantly better than both v-disparity variants, but with a much smaller margin and not for all frames. It appears that the planar v-disparity



(a) Error on synthetic sequence

(b) Distances in synthetic sequence

(c) Error on real sequence

(d) Distances in real sequence

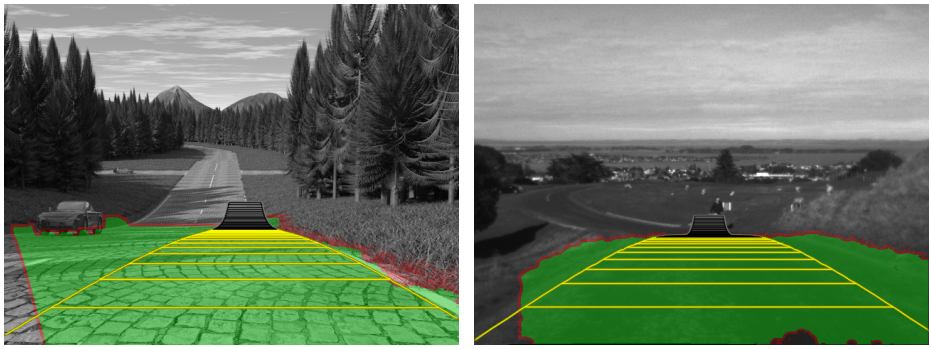
Fig. 2: Quantitative comparison of B-spline and v-disparity road-modeling.

variant performs better than the envelope version for most of the tested frames. The sudden increase in the modeling error at the end of the sequence can be explained by the much worse performance of the used stereo matching algorithm during this section. Figure 2d compares the visible and fitted distances for the tested sequence.

8 Conclusions

In this research we have proposed a new method for modeling the vertical road profile using B-spline curves. The method does not require a free-space estimation and has proven to work on scenes where the road is not constrained by any prominent boundaries. Examples for the performance of this method on a synthetic and real world scene are shown in Figs. 3a and 3b. In our experiments, the new method performed better than both tested versions of the popular v-disparity technique. The advance was major on the tested synthetic sequence but minor on the real world sequence. We suspect that this gap is caused by the lower accuracy of the disparity map for the real world sequence. Using better stereo matching algorithms could thus improve the results of our road-modeling technique.

Furthermore, we have found that using the envelope of best matching straight lines for the v-disparity method does not produce any better results. On the real world sequence, the performance of the envelope based v-disparity implementation produced the worst results for most of the frames, while performing



(a) Model created for synthetic sequence. (b) Model created for real sequence

Fig. 3: Examples for created B-spline road model curves.

roughly equal to the planar v-disparity version on the synthetic sequence. Our road-modeling method has proven to be competitive to both tested v-disparity approaches. Nevertheless, more research is required to further improve the accuracy. The method could particularly benefit from taking features of the intensity image into account and introducing a temporal filter. This could, however, distort the comparison if the v-disparity results are not filtered as well.

References

1. Weber, J., Koller, D., Luong, Q.T., Malik, J.: Integrated stereo-based approach to automatic vehicle guidance. In: IEEE International Conference on Computer Vision. (1995) 52–57
2. Labayrade, R., Aubert, D., Tarel, J.P.: Real time obstacle detection in stereovsion on non flat road geometry through "v-disparity" representation. In: IEEE Intelligent Vehicle Symposium. (2002) 646–651
3. Oniga, F., Nedevschi, S., Marc, M., Thanh, B.: Road surface and obstacle detection based on elevation maps from dense stereo. In: IEEE Conference on Intelligent Transportation Systems (ITSC). (2007) 859–865
4. Nedevschi, S., Danescu, R., Frentiu, D.: High accuracy stereovision approach for obstacle detection on non-planar roads. In: IEEE Intelligent Engineering Systems (INES). (2004) 292–297
5. Wedel, A., Badino, H., Rabe, C., Loose, H., Franke, W., Cremers, D.: B-spline modeling of road surfaces with an application to free-space estimation. IEEE Transactions on Intelligent Transportation Systems **10** (2009) 572–583
6. Eberly, D.: Least-Square Fitting of Data with B-Spline Curves. Geometric Tools, LLC. (2008)
7. Viola, P., Jones, M.: Robust real-time object detection. International Journal of Computer Vision **57** (2002) 137–154
8. The University of Auckland: Multimedia Imaging Technology Portal – EISATS. (`http://www.mi.auckland.ac.nz/EISATS`) Viewed 19.04.2010.