Probabilistic Disparity Fusion for Real-Time Motion-Stereo

Christian Unger, Eric Wahl, Peter Sturm, Slobodan Ilic

Technical Report

Abstract. Advanced driver assistance using cameras is a first important step towards autonomous driving tasks. However, the computational power in automobiles is highly limited and hardware platforms with enormous processing resources such as GPUs are not available in serial production vehicles. In our paper we address the need for a highly efficient fusion method that is well suited for standard CPUs. We assume that a number of pair-wise disparity maps are available, which we project to a reference view pair and fuse them efficiently to improve the accuracy of the reference disparity map. We estimate a probability density function of disparities in the reference image using projection uncertainties. In the end the most probable disparity map is selected from the probability distribution. We carried out extensive quantitative evaluations on challenging stereo data sets and real world images. These results clearly show that our method is able to recover very accurate disparity maps in real-time.

1 Introduction

Dense real-time multi-view stereo allows for a wide spectrum of useful applications including automotive driver assistance or robotic navigation. Although a large amount of research has been devoted to the stereo problem using image pairs [1–8] and using multiple cameras [9–14], obtaining dense high-quality disparity maps in real-time is still a challenging problem. Traditional real-time stereo methods [5, 8] still lack accuracy compared to methods which do not impose time constraints. A few multi-view stereo methods [15, 16] may achieve real-time performance, but only by using the enormous processing power of graphics

Fig. 1. Real-Time Motion-Stereo for automotive driver assistance: a camera on the vehicle observes the lateral space. If the vehicle moves, depth is inferred via motion-stereo. We use our proposed fusion method to remove outliers and to improve quality.
cards. However, such hardware is not feasible for mobile platforms and therefore it is absolutely necessary that all calculations can be performed in real-time on a standard mobile CPU at video frame rate.

Another important application based on traditional pair-wise stereo methods is motion stereo for automotive driver assistance (see Fig. 1): a camera is mounted laterally on a vehicle, so depth can be computed over time if the vehicle moves. From these disparity maps, we build a model of the environment, in order to mitigate collisions or to find lateral parking space. Even at higher velocities, the disparity maps may exhibit a large overlap and thus depth information is highly redundant. At the same time, due to the real-time stereo method used, disparities are very error prone. The question is how to fuse all those disparity maps to improve the accuracy of the disparity map defined by a reference image pair (for example, the last two images in case of motion stereo). The problem of fusing disparity maps was addressed by [15–17] in order to produce either dense disparity maps of the scene from video [17], or to do a dense surface reconstruction [15], or to recover a volumetric grid [16]. These methods can either globally fuse disparity maps obtained from different views or locally fuse them from the overlapping views. We restrict ourselves to the local fusion of the disparities among overlapping views, because it is more typical for our application. Since the methods of Merrell et al. [15] and Zhang et al. [17] currently provide impressive results, we extensively compare our method to them. However, both methods are not real-time with our hardware: [15] requires a GPU to be real-time and [17] performs expensive energy minimization with belief propagation.

In our paper we assume that a set of disparity maps is available and that they were computed using any available short baseline stereo technique. Then, given any other reference view pair we propose a novel probabilistic disparity fusion method to produce an accurate disparity map of the given reference view pair by fusing all available disparity maps. In our approach, we first project all disparity maps to the reference view pair. After maintaining visibility constraints, we estimate a probability density function over all valid disparities in the reference view using uncertainties of these reprojections and their photo-consistencies. Finally, this allows us to select the most probable disparity map from this distribution. In addition we model occlusions which produce holes in the reprojected disparity maps and define reliable areas by checking visibility in the reference and input views. This contributes to the overall statistics of the disparity and provides a better pdf estimation.

We tested our method on the challenging datasets of Middlebury [7] and compared it to the fusion methods of [15] and [17]. The experiments show that our technique is very robust and that the quality is significantly improved, especially in occluded regions and at discontinuities. We also show results on real-world sequences acquired from a camera attached to a vehicle. A very important fact is that our method allows real-time operation on CPU without dedicated hardware.

In the remainder of the paper we will first review related work, then present our method and finally show an exhaustive experimental evaluation.
1.1 Related Work

In recent years, traditional stereo and multi-view stereo methods have been extensively studied and tested using the available Middlebury datasets [13]. While resulting in a large amount of excellent results, little attention has been spent on computational performance. However, when that was the case and real-time stereo methods were proposed [5, 8], the reconstruction quality was significantly decreasing. While multi-view stereo approaches introduce assumptions on shape priors and use robust photo-consistency measures, there are others which aim to produce consistent disparity maps [18, 10, 15, 14, 17, 19]. In many cases, disparity maps that are produced locally using a number of overlapping views, are later fused into either a global disparity video [17], or a full 3D model [15, 16]. Again, the vast majority of works aim at high quality reconstructions of single objects and only very few try to minimize the computational overhead.

Since the main motivation of our work comes from motion stereo we tend to fuse locally overlapping disparity maps and do not aim to produce full 3D models. Works of Merrell et al. [15] and Zhang et al. [17], which explicitly deal with fusion of the disparity maps are thus directly related to our approach.

Merrell et al. [15] compute depth maps between neighboring views and fuse this information based on the stability of every depth. In order to keep track of occlusions, the stability is determined for every depth hypothesis and is defined by counting occlusions in the reference and other views. A valid depth is defined as the first depth hypothesis which is stable. However outliers affect the stability and such hard decisions may produce incorrect depth estimates. Further, the computational complexity grows quadratically with the number of disparity maps and in practice real-time operation is only possible with GPU hardware. In our paper, we overcome these problems. Our probabilistic approach employs reprojection uncertainties, handles outliers robustly and depth-accuracy gets improved compared to this approach.

Zhang et al. [17] impressively generalized the fusion problem by formulating it as an energy minimization problem. In their bundle optimization framework all disparity maps are optimized iteratively using belief propagation. In contrast to Merrell et al. [15] they do not model occlusions or visibility constraints explicitly. To our understanding, these constraints are handled by the simultaneous use of geometric coherence and color-similarity as well as the regularization of belief propagation. The minimization of the energy functional is in practice very time consuming and thus, this method is not an option for mobile real-time applications.

Koch et al. [20] introduced the efficient correspondence linking algorithm: by chaining correspondences across many views outliers are rejected and accuracy is improved. However, no solution was provided for multiple disparity maps per view and disparities in occluded regions or outliers near the beginning of the chain are problematic.

Finally the method of Zach [16], which fuses multiple depth maps to obtain a full volumetric 3D reconstruction, was formulated as a relatively efficient method that uses the GPU and produces very good results. However, the hardware re-
quirements are too high and the volumetric representation is problematic for our application.

Compared to other fusion methods, our work focuses mainly on real-time performance, but also offers quality depth maps. This is demonstrated through exhaustive experimentation and comparison to prior art.

2 Method

The major problem in motion and multi-view stereo are occlusions and discontinuities. Here we consider a reference view pair (RVP) in which we want to improve disparities, especially in occluded and discontinuity areas by bringing the information from other view pairs to this view. For this we propose to compute a probability density function defining the probability of the disparities in the reference image. It is done from the re-projection of all disparity maps of all available view pairs to this RVP. This allows us to select the most probable disparity at certain pixel locations of the RVP. Since the probability density function (pdf) is sampled from a relatively large number of measurements coming from other view pairs reprojected to the RVP, we demonstrate in the results section, that our approach significantly improves disparities at occlusions and areas near discontinuities.

Left-Right and Right-Left Distinction. We assume that there is a set of $n$ views and a set of $m$ input disparity maps between pairs of those views. Disparity maps between two, pairwise rectified views $\mathcal{I}_A$ and $\mathcal{I}_B$ are denoted by $D_{A,B}$ (using left-right stereo-matching) and $D_{B,A}$ (using right-left matching). In the rest of the section we will discuss the computation of only one pdf for left-right disparity maps $D_{A,B}$. The computation of the pdf for right-left disparity maps is identical. The intuition behind this is that in most stereo methods left object boundaries are usually very stable when performing right to left matching (because no occluded pixels are present there in the right image). This implies that we do not check for left-right consistency. Instead we directly use unfiltered disparity maps to compute two separate pdfs and combine the information later.

Goal. Our goal is to compute an improved disparity map $\hat{D} = \hat{D}_{R_1,R_2}$ for a specific RVP $(\mathcal{I}_{R_1}, \mathcal{I}_{R_2})$. To do this we transfer disparities from all input disparity maps (e.g. $D_{0,1}$, $D_{2,3}$) to the RVP $(\mathcal{I}_{R_1}, \mathcal{I}_{R_2})$. A simple triangulation and projection is sufficient [17] to perform this transfer. Independent from the transfer method used, we refer to it using the transfer function $\Theta_{k}^{A,B} : (x_{A}, d_{A,B}) \mapsto x_{k}$, which transfers the point $x_{A}$ using input disparity $d_{A,B} = D_{A,B}(x_{A})$ into view $\mathcal{I}_k$. So, we use functions $\Theta_{R_1}^{A,B}$ and $\Theta_{R_2}^{A,B}$ to compute a reprojected disparity map $\hat{D}_{A,B}$ by applying the transfer to every disparity in $D_{A,B}$: $\hat{D}_{A,B}(x_{R_1}) = \Theta_{R_1}^{A,B}(x_{A}, D_{A,B}(x_{A})) - \Theta_{R_2}^{A,B}(x_{A}, D_{A,B}(x_{A})) = x_{R_1} - x_{R_2}$.

In practice, all available disparity maps are transferred to the RVP. Later, all these disparity maps are used to compute the pdf of the disparities in the RVP. From this pdf, the most probable values are extracted and define the improved disparity map $\hat{D}_{R_1,R_2}$. 
2.1 Handling Occlusions

When performing the reprojection, depending on the occlusions and discontinuities in \( D_{A,B} \), there are in general zero, one or even multiple disparity estimates for every pixel of the reprojected disparity map. In an ideal world, the case with only one disparity occurs when cameras of the reference and input views observe only non-occluded scene points. Multiple disparities occur due to depth discontinuities where several input disparities reproject to the same location in the reference view with different disparities. However, in our method we must make sure that there is only one disparity per pixel and thus we choose the closest depth estimate (i.e. maximal disparity) from these values to maintain correct visibility. Zero disparities can occur due to disparity quantization in the input view or due to occlusions on surface discontinuities. It means that at a particular pixel location in the reprojected disparity map there is no information about the disparity, resulting in holes in the disparity map. We eliminate those holes and fill them with approximated values taking into account surrounding disparity values. This is important to be done because it improves the estimation of the probability density function in many cases. Below we discuss proposed solutions for the hole filling. In addition it is very important to check whether close parts of the scene are visible in the input views: we have to ensure that background disparities are used for the pdf, if the input disparity map contains information about the presence of foreground disparities. We perform this check (the reliable area) using the maximum disparity (the closest possible depth) and invalidate areas which are not visible in both, reference and input view pairs. If we did not perform this check, background disparities would receive too much support and chances would be high that background is visible in spite of the presence of foreground objects.

Holes from disparity quantization. In the first row of Fig. 2 we show an example where missing disparities in the reprojected disparity map are artifacts of the
disparity quantization. This usually happens on slanted surfaces or on discontinuities. In Fig. 2 we used an example of a slanted surface. Discontinuity of the input disparity map shown in the first row of Fig. 2 is an artifact of disparity quantization. When reprojected to the reference view, holes are created whose size varies depending on the camera motion between the reference and input frames. We detect and interpolate those holes by checking if the difference of the left and right neighbouring disparities is less than a small threshold. In practice, we set this value to two and interpolate holes smaller than five pixels.

Holes at Occlusions. The remaining holes occur at depth discontinuities. In the second row of Fig. 2 we show an example of occlusion where part of the surface is visible in the RVP, but is not visible in the input view pair. The reprojected disparities near the discontinuity (i.e. the occluded area) will create a hole at that place. To fill this hole we chose to extrapolate the left and right neighboring surfaces. To avoid using all left and right disparities, which can belong to multiple objects in the scene, we segment those disparities and take only those that potentially create the same surface. The interpolation is done by linear regression, i.e. we fit the line to segmented left and right disparities as shown in Fig. 2. At a specific point \( x \), where the disparity is missing, the extrapolation gives two estimated disparities \( d_l \) and \( d_r \) and we use the background (the occluded surface) as the final disparity, i.e. \( \min(d_l, d_r) \).

Reliable Area. We must ensure for every reprojected disparity that it comes from the surface visible in both, reference and input camera pair. If that is not the case in means that the point corresponding to this disparity is occluded or not visible in the reference view. To check this we verify if a point on the surface defined by the maximum disparity is outside the frustum of \( I_A \) and \( I_B \). In practice, for every point \( x_{R_1} \in I_{R_1} \), we compute \( x_k = \Theta_{R_1,R_2}(x_{R_1}, d_{\text{max}}) \) and check if \( x_k \in I_k \) for \( k \in \{A,B\} \). If \( x_k \notin I_k \), then the disparity at \( x_{R_1} \) is invalidated, meaning that either it is occluded in the reference view or it is not visible in the input view. Here, \( d_{\text{max}} \) is the maximum disparity of view pair \( I_{R_1} \) and \( I_{R_2} \).

2.2 Probability Density Function of Disparity

We reproject all input disparities to the RVP and use them as measurements to compute a probability density function of the disparity in the reference image. Later we draw from this pdf the most probable disparity at every pixel location of the reference view as illustrated in Fig. 3.

First we build the set of reprojected disparity maps \( \tilde{S} \) by reprojecting every input disparity map to the RVP. Using \( \tilde{S} \) we obtain a set of disparity estimates for every pixel: \( C(x) = \{ d \mid d = \tilde{D}(x), \tilde{D} \in \tilde{S} \} \).

Now we use these disparities as measurements to sample the pdf of disparity \( d \) at any pixel location \( x \) in the reference image. If we assume that \( x \) and \( d \) are random variables that take all possible image locations and all possible reprojected disparities, and that they represent independently and identically
Fig. 3. The pdf estimation: (a) The 2D-geometry is observed from three different stereo cameras. (b) Disparity maps from input stereo pairs are determined. (c) The reprojected disparity maps to the reference view pair lead to (d) three pdfs for each reprojected disparity map which are finally (e) combined in one pdf.

distributed samples, we can approximate the probability density function with:

\[
p(x, d) = \frac{1}{m} \sum_{\tilde{x} \in \mathbb{R}} \sum_{\tilde{d} \in \mathbb{C}} P(x, d | \tilde{x}, \tilde{d})
\]

(1)

where \(P\) is a probability of the disparity \(d\) at pixel location \(x\) computed from the measurements \(\tilde{d} \in \mathbb{C}\) at location \(\tilde{x}\).

PDF estimation. The probability of disparity \(d\) at pixel location \(x\) given a disparity measurement \(\tilde{d} = \tilde{D}_{A,B}(\tilde{x})\) at location \(\tilde{x}\) of a reprojected disparity map is given by \(P(x, d | \tilde{x}, \tilde{d})\). It depends on the probability \(P_L(x, d | \tilde{x}, \tilde{d})\) that the scene point \(X(\tilde{x}, \tilde{d})\) (computed from the uncertain correspondence \(x_A \leftrightarrow x_B\) before reprojection) projects to the location \(x\) in the image \(I_B\). It further depends also on the probability \(P_D(x, d | \tilde{x}, \tilde{d})\) that the disparity of \(X\) is \(d\) in the RVP. \(P_C\) is the probability that measures color-similarity between \(x_A\) and \(x_B\). So we write it as:

\[
P(x, d | \tilde{x}, \tilde{d}) = P_L(x, d | \tilde{x}, \tilde{d}) \cdot P_D(x, d | \tilde{x}, \tilde{d}) \cdot P_C(\tilde{x}, \tilde{d})
\]

(2)

We assume that \(P_L\) has its maximum probability at the position \(\tilde{x}\) and that it decreases with increasing distance:

\[
P_L(x, d | \tilde{x}, \tilde{d}) \sim \exp \left( -\frac{\|x_A - \tilde{x}_A\|^2}{2\sigma_x^2} \right)
\]

(3)

Similarly, the probability \(P_D\) should be maximal at \(\tilde{d}\) and should be small for differing depths:

\[
P_D(x, d | \tilde{x}, \tilde{d}) \sim \exp \left( -\frac{\|\tilde{x}_B - \tilde{x}_A\| - (x_B - x_A)^2}{2\sigma_d^2} \right)
\]

(4)
with \( x_A = \Theta_{R_1,R_2}^A(x,d) \) and \( \tilde{x}_A = \Theta_{R_1,R_2}^A(\tilde{x},\tilde{d}) \) (\( x_B \) and \( \tilde{x}_B \) analogously). Here \( \Theta_{R_1,R_2}^A \) and \( \Theta_{R_1,R_2}^B \) are transfer functions which back-project point \( x \) with the disparity \( d \) from the RVP to the input view pair. \( \sigma_x \) is the location uncertainty and \( \sigma_d \) is the accuracy of the disparity estimation. Note that \( \tilde{x} \) and \( \tilde{d} \) are taken from the set of reprojected disparity maps. If the disparities \( d \) and \( \tilde{d} \) are the same, the point defined by \( (x,d) \) will project to exactly the same input locations \( \tilde{x}_A \) and \( \tilde{x}_B \) and define the same disparity in the input view, which will result in the maximum probability. Otherwise, points with different disparities or locations will back-project to locations away from the measurement \( (\tilde{x},\tilde{d}) \) and get much lower probability. The slope of the functions \( P_L \) and \( P_D \) is mainly determined by the configuration of camera positions and the baselines of the view pairs.

The color-similarity measure between \( x_A \) and \( x_B \) is given as in [17]:

\[
P_C(\tilde{x},\tilde{d}) = \frac{\sigma_C}{\sigma_C + |I_A(\tilde{x}_A) - I_B(\tilde{x}_B)|}
\]  \( (5) \)

where \( \sigma_C \) is the color variance which we obtained experimentally. The value of \( P_C \) will be 1 for points with identical image intensities and will decrease as a function of the color dissimilarity.

### 2.3 Disparity Selection

From the probability distribution \( p(x,d) \) we select for every pixel \( x \) the disparity with highest probability: \( \hat{d} = \arg\max_d p(x,d) \). We compute two different probability density functions \( p_l \) and \( p_r \) each corresponding to the reprojection of left-right \( D_{A,B} \) and right-left disparity maps \( D_{B,A} \). In our experiments we found out, that the final disparity should be defined as \( \hat{d} = \min(\hat{d}_l,\hat{d}_r) \), where \( \hat{d}_l \) and \( \hat{d}_r \) are obtained from \( p_l \) and \( p_r \). When determining \( \hat{d} \), using the max-function instead of the min-function appears to significantly degrade the final disparity map (see Fig. 5). This is due to the bad performance of most stereo methods at object boundaries (regions near discontinuities). Distinguishing between left-right and right-left matching will separate good from bad left-boundaries (and bad from good right-boundaries). In ambiguous situations, the min-function will favour the background (which is usually occluded).

### 3 Results

We evaluate our method using classical stereo datasets with ground truth [7] and demonstrate the applicability to real world data. In our experiments we used \( \sigma_d = 1, \sigma_x = 1 \) and \( \sigma_C = 5 \). The standard two-frame stereo datasets from Middlebury [7] contain up to 9 images from which we computed 72 (\( Venus, Teddy, Cones \)) or 42 (\( Art, Moebius, Aloe \)) disparity maps from all possible image combinations. After that, we fused these disparity maps to the standard reference view pair (e.g. (2,6) for \( Teddy \)) and computed the percentage of erroneous pixels (disparities that differ by more than 1). For stereo processing we used Belief Propagation [2] (BP), Semi-Global Matching [3] (SGM), Geodesic Support
Fig. 4. The performance of our fusion method with different stereo methods (GSW, SGM, BP and RT) or when using GSW and no hole filling. Error bars show percentages of disparities that differ by more than 1 from the ground truth in the whole image (all), non-occluded (nocc) or occluded pixels (occl) and regions near discontinuities (disc). We fused up to 72 disparity maps. The overall improvement of hole filling is below 2% and replacing the stereo method makes only a slight difference.

Fig. 5. The impact of the different steps of our method: the use of projection uncertainties is important and the distinction between left-right and right-left matching (LR/RL) using the min-function helps at discontinuities. See Fig. 4 for a description of the bars.

Weights [6] (GSW) and a local Real-Time method [8] (RT). We used constant parameters for the stereo methods among all baselines and datasets.

We found out that fusion results are relatively independent of the actual stereo method used (see Fig. 4), but particular systematic errors of each stereo method are still visible after fusion (e.g. bad object boundaries for RT). We also measured that the overall improvement of hole filling is below 2% – it helps mainly in occluded regions. The use of projection uncertainties is relatively important (see Fig. 5): even for very well calibrated sequences, there is always an uncertainty in matching which influences reprojection. The distinction between left-right and right-left matching (see also section 2.3) is important in regions near discontinuities (see Fig. 5).

3.1 Comparison to other Fusion Methods

We compare our method to other fusion algorithms, in particular the stability-based algorithm of Merrell et al. [15] using our own implementation running on CPU and the bundle optimization of Zhang et al. [17] using their own implementation (without their stereo-matching and without final bundle adjustment). We used the same input data (i.e. disparity maps) for all fusion methods. We also perform hole-filling when using the method of [15] (which was not described in [15]), because it leads to slightly better results. During our benchmark, we got
Fig. 6. The performance of different fusion methods. Disparity maps were computed using GSW [6]. See Fig. 4 for a description of the bars.

<table>
<thead>
<tr>
<th>Art</th>
<th>True Disparities</th>
<th>Our Method</th>
<th>Zhang et al.</th>
<th>Merrell et al.</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Art" /></td>
<td><img src="image2" alt="True Disparities" /></td>
<td><img src="image3" alt="Our Method" /></td>
<td><img src="image4" alt="Zhang et al." /></td>
<td><img src="image5" alt="Merrell et al." /></td>
</tr>
</tbody>
</table>

Fig. 7. The disparities and bad pixels of different fusion methods for the dataset Art.

the feeling that the method of [17] is optimized for short baselines (the video sequences of [17] have much smaller baselines than the datasets of [7]). Our method works better with larger baselines, which is our target application.

For our comparisons we focused on recent, challenging datasets. For example, Art [21] is a dataset from 2005 with several difficult occlusions and makes it an ideal test case for algorithms that fuse disparity maps: from a fusion algorithm we expect that it performs very well in occluded areas, because there may be other views contributing information. The performance in regions near discontinuities is also very interesting, because in these areas the fusion algorithm usually has to select the correct value from a mixture of the nearby fore- and background disparities.

**Regions Near Discontinuities.** Our method preserved sharp object boundaries and thin structures: obvious when looking at the error bars in Fig. 6 (e.g. at Art) and disparity maps in Fig. 7 (e.g. the house in Teddy, or Moebius). Hole-filling helps only slightly, but the distinction between left-right and right-left matching is quite important: left object boundaries are stable in right to left matching
(and vice versa). The approach of Merrell et al. appears to perform better at discontinuities than the method of Zhang et al.\(^1\), but behaves less robust if the set \(C\) contains many outliers.

**Occluded and not Occluded Regions.** In these regions, we benchmarked our method better than the other methods which is mostly due to our probabilistic model (which is robust to outliers), the explicit visibility computation (using reprojection and the reliable area; it strongly narrows the number of hypotheses) and also hole-filling (it reduces artifacts from reprojection and helps in occluded regions). In general, the method of Zhang et al. seems to excel at planar surfaces and the fused disparity maps tend to be slightly oversmoothed (visually very obvious in Fig. 8 at the Moebius dataset), which might be optimized by parameter tuning. There is also another interesting artifact, which we were not able to explain: thin objects in Fig. 7 are extracted, but with an incorrect disparity value. The robustness of Merrell et al. seems to be impacted by the number of outliers in the set \(C\).

**How can this proposal be better than prior art in these experiments?** In Merrell, visibility-constraints are enforced using their expensive definition of stability (having a complexity of \(O(m^2)\)). However, visibility can be maintained more efficiently using reprojection and the reliable area (having \(O(m)\)). This also has the big advantage that projection uncertainties can be used later, whereas in Merrell it is not possible. Moreover, for optimal stability calculation it is important that the number of outliers having a negative stability is equal to the number of outliers with positive stability. This assumption seems to behave suboptimally in occluded regions, when many outliers are present.

In Zhang’s method, the correct disparity is supported by the simultaneous combination of geometric coherence and color similarity. Geometric coherence alone supports also background disparities of surfaces occluded by foreground objects in the reference view, because visibility is not determined and this is problematic in cases where fore- and background objects are of similar color. The optimization using belief propagation ensures smoothness in these ambiguous situations but seems to perform suboptimally in regions near discontinuities. Due to the results we obtained during our experimental evaluation (our method does not use any kind of energy minimization), we believe that our pdf will also bring a huge advantage to the method of Zhang, especially near discontinuities and when using wide-baseline sequences.

Further, we would like to stress that in our method from every single input disparity a global pdf is computed (parameterized using projection uncertainties and color-similarity). The final pdf of the reference disparity map results from summing up all those single pdfs. Efficiency is preserved by computing the final disparity map “bottom up” and the probabilistic model ensures robustness. Hole-

\(^1\) On the first sight, this may look odd, when taking the performance in non-occluded and occluded regions into account, where Zhang et al. is better in most cases. However, the disc-value is computed within a smaller, predefined portion of the image.
filling helps slightly in occluded regions and the distinction between left-right and right-left matching is important for sharp object boundaries.

**Execution Times.** At the dataset *Teddy* (72 disparity maps) our method took 8.7 s (not optimized), the method of [15] took 40.7 s and the method of [17] 175 minutes (i.e. 146 s/disparity map). These times do not include stereo matching and were measured on a Intel E8200 dual-core with 2.66 GHz (for our method and [15]) or a Intel E5405 quad-core Xeon CPU with 2.00 GHz (for [17]).

For our real-time implementation, we use SIMD-instructions of the SSE2 instruction set and simplified the reprojection for motion-stereo (in this case, we assume that the y-coordinate of projected scene points is constant over time). Using pre-computed kernels, we are able to fuse 16 disparity maps in just 20 ms on a mobile CPU (2 GHz).

We also implemented an incremental variant that needs 3 ms per frame. We use the pdf of the previous frame and reproject it to the current vehicle position. This pdf is then updated using the disparity maps computed using the current camera image and the fused disparity map is determined.

### 3.2 Real World Scenes

We tested our method on real world sequences from a moving vehicle. Fig. 9 shows a rectified camera frame, one input disparity map (computed using a real-time stereo method [8]) and one fused disparity map. For fusion we used a highly optimized implementation (using SIMD instructions) to fuse 16 adjacent input disparity maps.
Fig. 9. Our method applied to sequences from our vehicle using real-time stereo [8].

Fig. 10. Our method applied to the scene “Road” provided by [17].

Fig. 10 shows fused disparity maps of a sequence provided by [17], along with the camera frame and their fused depth map. For stereo matching we used GSW [6] and ensured a minimal baseline of 5 and a maximal baseline of 7 frames (the baseline of adjacent frames was too small for robust matching with GSW). We fused disparity maps of 20 adjacent frames and this explains why some disparities which are outside of the field of view are missing (black regions at the left and right). Please have a look at the supplemental material for a complete video sequence.

4 Conclusion

In this paper, we propose a novel probabilistic method for fusing disparity maps in classical stereo or motion-stereo setups. We achieve this by computing a probability density function from all provided disparity maps. From this distribution, we determine the most probable disparity map for a given reference view pair.

We introduce several novel concepts: reprojection using the reliable area (for efficient visibility determination), a generic probabilistic model that uses projection uncertainties (for robustness against outliers), a distinction between left-
right and right-left matching (for sharp object boundaries) and hole-filling (for improved quality in occluded regions).

We compare our method to the current art using real scenes and disparity maps generated from datasets with ground truth. Our paper shows clearly that our proposal appeals through simplicity, good results and efficiency.

References