Accurate Real-Time Disparity Estimation with Variational Methods

Sergey Kosov, Thorsten Thormählen, Hans-Peter Seidel

Max-Planck-Institut Informatik (MPII), Saarbrücken, Germany

Abstract. Estimating the disparity field between two stereo images is a common task in computer vision, e.g., to determine a dense depth map. Variational methods currently are among the most accurate techniques for dense disparity map reconstruction. In this paper a multi-level adaptive technique is combined with a multigrid approach that allows the variational method to achieve real-time performance (on a CPU). The multi-level adaptive technique refines the grid only at peculiarities in the solution. Thereby it reduces the computational effort and ensures that the reconstruction quality is kept almost the same. Further, we introduce a technique that adapts the regularizer, used in the variational approach, dependend on the the current state of the optimization. This improves the reconstruction quality. Our real-time approach is evaluated on standard datasets and it is shown to perform better than other real-time disparity estimation approaches.

1 Introduction

A classical correspondence problem in computer vision is the estimation of a disparity field between a stereo image pair. During disparity estimation, for each pixel in one image the corresponding pixel in the other image is sought, so that the corresponding pixels are the projections of the same 3D position. Afterwards, if the camera calibration is known, a depth map can be calculated from the disparity field. If a standard stereo setup is used, the corresponding pixels are constrained to lie on the same row. Thus, the search range for the disparity is 1-dimensional.

Estimation of a 1D disparity field is related to the estimation of a 2D displacement field. A displacement field of corresponding pixels arises, e.g, between consecutive frames in an image sequences. Such a displacement field, represented as a vector field, is called *optic flow*. Variational methods allow to compute a precise and dense estimation of an optic flow field. Moreover, the research by Mémin and Pérez [1] and Brox et al. [2] has proven the variational methods to be among the best techniques for optic flow reconstruction. These techniques minimize an energy functional by solving the corresponing Euler-Lagrange equation. Numerically, the Euler-Lagrange equation is represented as a system of differential equations with finite differences. To optimize the energy functional, iterative solvers, like the Jacobi and Gauss-Seidel methods, are used. The speed of convergence of these methods is quite slow. As a result, processing a single image pair takes several minutes or even up to half an hour on todays CPUs.

As a remedy against this slow convergence, multigrid methods were developed [3], which allow to overcome the rigidity of the single grid approach by using multiple discretization levels. With a single fixed sampling grid, multiple solution components that have different scales may produce conflicting solutions and, thereby, cause slower convergence. For example, the smooth components, which are effectively approximated on coarse grids but slowly converge on fine grids, are often in conflict with high-frequency components, which should be taken into account only on fine grids.

In 1961, Fedorenko [4] formulated the multigrid algorithm for a standard 5-point discretization of the Poisson equation, which allowed to gain a numerical solution in $O(N)$ arithmetical operations (where N is the number of grid nodes). During the 1980s, Brandt [5], Stuben and Trottenberg [6], and Hackbush [7] made important contributions by transfering the multigrid ideas to the area of non-linear problems, by introducing *multi-level adaptation techniques* (MLAT), and by developing the *full multigrid* (FMG) method. In 2006, Bruhn et al. [8] have demonstrated a real-time variational solver for optic flow reconstruction with discontinuity-preserving techniques. The solver uses a coarse-to-fine strategy in combination with a *full approximation scheme* (FAS).

Because the first-order Taylor expansion to linearize the energy functional is only valid for small disparities, for large disparities, multigrid methods are often combined with so-called *warping steps*. With warping steps [9,2] the original problem is compensated by the already computed solution from all coarser levels before the remaining residual is minimized on the finer level. In this paper, we use linear interpolation to linearize the energy functional. This approach can handle large displacements directly and was shown to be faster and more accurate [10].

In this paper a current variational approach with multigrids is extended by a MLAT in combination with a FAS. In contrast to the current multigrid methods, a grid adaptation technique refines the sampling grid not for the whole image, but locally in regions where interesting structures are located [11]. A similar adaptive mesh algorithm, which is based on a Galerkin finite element method on a triangular mesh for object flow computation [12], is difficult to use with the FAS. It will be shown that with multigrids in combination with the MLAT, heterogeneous adaptive structures can be used with a variational solver for real-time disparity estimation. Thereby, the MLAT allows to quickly perform local and precise adjustment. Furthermore, improved reconstruction quality is achieved by adapting the applied regularizer locally during optimization. A comparison on standard data sets with other real-time disparity estimators shows that our real-time variational approach outperforms the current state-of-the-art.

The paper is organized as follows. The next section gives an introduction to disparity estimation with variational methods. In Section 3 first state-of-the-art multigrid techniques are described and afterwards a different multigrid techniques to improve computation time is suggested. This technique is base on what we call *null-cycles* (O-cycles). Section 4 describes our multi-level adaptive technique for variational solvers. In Section 6 the approach is evaluated and the paper ends with a conclusion.

2 Variational Methods

Let us suppose that we are given a stereo image pair. Each scalar-valued image $I(x, y)$ is stored in a pixel matrix and $(x, y)^\top$ is the coordinate of the pixel within the rectangular image domain Ω. Having two images of a stereo pair $I_1(x, y)$ and $I_2(x, y)$, we try to estimate the position to which every pixel from the first image has moved in the second image. In order to do that, we have to assume that certain image features are still the same in both images. Such features may include the grey value, higher image derivatives (such as the gradient or the Hessian), or scalar-valued expressions (such as the norm of the gradient, the Laplacian, or the determinant of the Hessian) [13]. For simplicity, we will only consider grey value constancy assumption in the remainder of this paper.

If $(x, y)^\top$ is the coordinate of a pixel in the first image and $u(x, y)$ is the disparity, then $(x + u(x, y), y)^\top$ is the new position of the pixel in the second image. By formulating the problem like that, we can state that the computation of the disparity field is actually the computation of the vector field $(u(x, y), 0)^\top$. Now we can write the grey value constancy assumption:

$$I_1(x, y) - I_2(x + u(x, y), y) = 0 \quad . \tag{1}$$

As we are working with continuous real-world data, which is not discrete like the pixel locations in the pixel matrices, the disparities are not necessarily integer values. To perform the linearization, we use a linear interpolation technique [14]. We express the disparity $u(x, y)$ as the sum of two components: integer $A(x, y)$ and floating point $b(x, y)$, such that:

$$u(x, y) = A(x, y) + b(x, y), \quad \text{with} \quad |b(x, y)| < 1 \quad . \tag{2}$$

The linearized form of Eq. (1) is given by:

$$\begin{gathered} I_1(x, y) - |b(x, y)| \cdot I_2(x + A(x, y), y) - \\ (1 - |b(x, y)|) \cdot I_2\left(x + A(x, y) + \operatorname{sign}(b(x, y)), y\right) = 0 \end{gathered} \tag{3}$$

We construct an energy functional, that consists of two terms: a data term that imposes constancy on the grey values, and a smoothness term that regularizes the often non-unique (local) solution of the data term by an additional smoothness assumption.

2.1 Data term

Due to possible occlusions or unpredictable reflection properties of the object's surfaces, the equality from Eq. (1) can usually not be satisfied perfectly in reality. However, we can fulfill the demand: $\|I_1(x, y) - I_2(x + u(x, y), y)\|^2 \to \min$. The energy functional $E(u(x, y))$, based on the grey value constancy assumptions, can be written as:

$$E(u(x, y)) = \iint_\Omega \|I_1(x, y) - I_2(x + u(x, y), y)\|^2 \, dS \quad . \tag{4}$$

2.2 Smoothness term

The smoothness term is derived from the assumption that the neighboring regions belong to the same object and thus these regions have similar disparity. The main role of the smoothness term is the redistribution of the computed information and the elimination of local disparity outliers. In case that reliable information from the data term is not available, the smoothness term helps to fill the problematic region with disparities calculated from neighboring regions.

In this paper, we use 3 different regularizers: Tichonovm, Charbonnier, and Perona-Malik regularization. Tichonov regularization assumes overall smoothness and does not adapt to semantically important image or flow structures (Horn and Schunck [15]). Charbonnier's and Perona-Malik's flow-driven regularization assumes piecewise smoothness and respects discontinuities in the flow field (see, e.g., [16,17,18,19]). For all three regularizers, the smoothness term in general form is given by

$$\Psi(|\nabla u(x,y)|^2) \quad . \tag{5}$$

Thus, we can rewrite the energy functional (4) as follows:

$$E(u(x,y)) = \iint_{\Omega} \|I_1(x,y) - I_2(x + u(x,y), y)\|^2 + \varphi \cdot \Psi(|\nabla u(x,y)|^2) \, dS, \tag{6}$$

where φ is a weighting factor for the smoothness term. In case of the Tichonov regularizer the smoothness term is given by

$$\Psi(s^2) = s^2 \quad , \tag{7}$$

for the Charbonnier regularizer by

$$\Psi(s^2) = 2\lambda^2 \sqrt{1 + \frac{s^2}{\lambda^2}} - 2\lambda^2 \quad , \tag{8}$$

and for the Perona-Malik regularizer by

$$\Psi(s^2) = \lambda^2 ln(\lambda^2 + s^2) - \lambda^2 ln(\lambda^2) \quad . \tag{9}$$

2.3 Euler-Lagrange equation

The goal of the variational method is to find a function $u(x,y)$, which minimizes the energy functional $E(u(x,y))$. In other words, having constructed the energy functional, we should minimize it in order to find the best solution for the disparity field. Moreover, if the constructed functional is strictly convex, it will have a unique solution that minimizes it.

The Euler-Lagrange equation is an equation satisfied by the unknown function $u(x,y)$ that minimizes the functional $E(u(x,y)) = \iint_{\Omega} F(x,y,u,u_x,u_y) \, dS$, where $u_x = \frac{\partial u}{\partial x}$, $u_y = \frac{\partial u}{\partial y}$ and F is a given function which has continuous first order partial derivatives. The Euler-Lagrange equation then is the partial differential equation:

$$F_u - \frac{\partial}{\partial x} F_{u_x} - \frac{\partial}{\partial y} F_{u_y} = 0 \quad . \tag{10}$$

For the energy functional (6) the Euler-Lagrange equation for each pixel $(x, y)^\top$ is given by

$$I_{2x}(x+u, y)(I_1(x, y) - I_2(x+u, y)) + \varphi \cdot \operatorname{div}(\Psi'(|\nabla u|^2) \cdot \nabla u) = 0 \quad . \tag{11}$$

In order to minimize our energy functional, we solve the resulting system of differential equations with homogeneous Neumann boundary conditions [20]. This step is done via discrete numerical schemes. The Euler-Lagrange equations are discretized, linearized with the Eq. (3), and approximated via finite-differences schemes. In the end, we arrive at a linear (in case of Tichonov regularizer) or non-linear (in case of Charbonnier or Perona-Malik regularizers) system of equations.

3 Multigrid

In general, large equation systems arising from finite difference approximations of elliptic boundary problems are solved with iterations methods, like the Jacobi or Gauss-Seidel method [21,22].

However, such methods converge very slowly for equation systems that are only coupled via a small local neighborhood because numerous iterations are needed to exchange data between unknowns that are coupled indirectly. This leads to efficient computation of high-frequency components, while the lower-frequency components remain almost unchanged. Multigrid methods effectively handle this problem by starting from a fine grid but then perform correction steps that compute the error on a coarser grid and propagate this information back to the finer grid. Thus, lower frequency components of the error reappear as higher ones on the coarser grid and allow an efficient attenuation with basic iterative methods.

To employ this multigrid approach to non-linear problems, the Full Approximation Scheme (FAS) is used. For completeness, in the next subsection a short introduction to the FAS is given.

3.1 Full Approximation Scheme (FAS)

In the following equations the indices H and h indicate entities from a coarser grid and a finer grid, respectively. For the sake of clarity, let us reformulate Eq. (11) as

$$L_h \, u_h = -f_h \quad . \tag{12}$$

Here L_h is a non-linear operator and f_h stands for the right hand side, which in our particular case is equal to zero. Let u_h^i denotes the approximate solution after i iterations. Then the error is given by

$$e_h^i = u_h - u_h^i \quad . \tag{13}$$

Substituting u_h in equation (12) with u_h from Eq. (13), we obtain

$$L_h(u_h^i + e_h^i) = -f_h \quad . \tag{14}$$

Now we substract $L_h u_h^i$ from the left and right parts of the Eq. (14):

$$L_h(u_h^i + e_h^i) - L_h u_h^i \;=\; -f_h - L_h u_h^i \quad . \tag{15}$$

Then we restrict the solution to the coarser grid. Here we have to introduce two operators I_H^h and I_h^H. Let I_h^H denote the *restriction* operator from a fine grid h to a coarse grid H and I_H^h the *interpolation* operator from a coarse grid H to a fine grid h. For the coarser grid we get:

$$L_H(I_h^H u_h^i + e_H^i) - L_H I_h^H u_h^i \;=\; -I_h^H f_h - I_h^H L_h u_h^i \quad . \tag{16}$$

If we denote

$$f_H = I_h^H f_h + I_h^H L_h u_h^i - L_H I_h^H u_h^i \quad , \tag{17}$$

then we can rewrite Eq. (16) in a short form: $L_H(I_h^H u_h^i + e_H^i) = -f_H$ (note the similarity with Eq. (12)). Let u_H^i denote the new approximation of the solution on the coarse grid with $u_H^i = I_h^H u_h^i + \tilde{e}_H$, where $\tilde{e}_H$ is the new error approximation after i iterations. From that follows:

$$\tilde{e}_H = u_H^i - I_h^H u_h^i \quad . \tag{18}$$

Now we interpolate the error to the finer grid $\tilde{e}_h = I_H^h \tilde{e}_H$ and after that correct the solution on the finer grid: $\tilde{u}_h = u_h^i + \tilde{e}_h$. The steps of the FAS are summarized in Fig. 1. Note that only the error and the residual are transfered to the finer grid, but not the solution, since only the error and the residual are smooth functions.

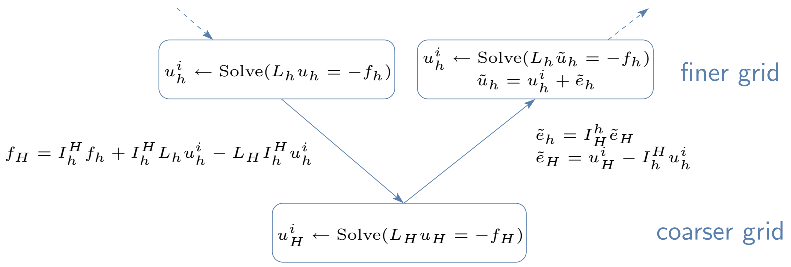
$u_h^i \leftarrow \text{Solve}(L_h u_h = -f_h)$
$u_h^i \leftarrow \text{Solve}(L_h \tilde{u}_h = -f_h)$
$\tilde{u}_h = u_h^i + \tilde{e}_h$
finer grid
$f_H = I_h^H f_h + I_h^H L_h u_h^i - L_H I_h^H u_h^i$
$\tilde{e}_h = I_H^h \tilde{e}_H$
$\tilde{e}_H = u_H^i - I_h^H u_h^i$
$u_H^i \leftarrow \text{Solve}(L_H u_H = -f_H)$
coarser grid

Fig. 1. The steps of the full approximation scheme.

3.2 V-cycle and W-cycle

The main idea of the multigrid method is that on a coarse grid we are not obliged to solve $L_H u_H = -f_H$ precisely. It is enough to perform a few iterations and achieve an approximate solution u_H^i. On each multigrid level, only a few iterations must be performed to compute the high-frequency components, because the lower-frequency components can be more efficiently computed on the coarser grids. Therefore, in order to increase the computational efficiency, two types of grid cycle are commonly used: V- and W-cycles. While the V-cycles make one recursive call of a two-grid cycle per level, the more reliable W-cycles perform two. The cycles are applied in a hierarchical way as depicted in Fig. 2.

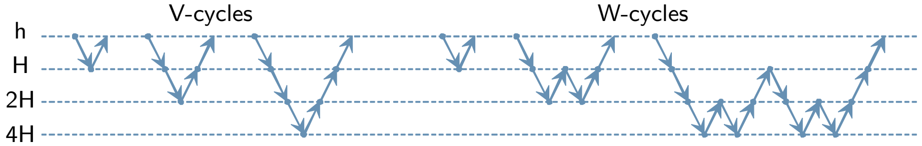

Fig. 2. V-cycles and W-cycles with two, tree and four levels.

3.3 Full Multigrid (FMG)

It is possible to significantly improve the convergence of the multigrid methods to the correct solution by applying the full multigrid method (FMG), also known as method of nested iterations. In contrast to the simple multigrid approach, the FMG approach starts from a coarse grid and not from a fine grid. The schematic view of the FMG method with W-cycles is shown in Fig. 3. The full multigrid method combines the solution from coarser grids as initial approximations and than applies V- and W-cycles for calculating the solution at finer grids. Details about the FMG can be found in [5,6].

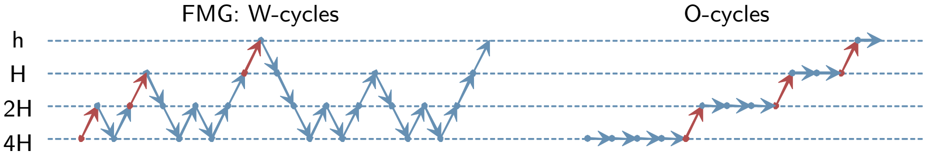

Fig. 3. Left:Full Multigrid (FMG) implementation with W-cycles per resolution level, Right: O-cycles. Refinement steps are marked with red color. Each W- and O-cycle is marked with a blue color.

3.4 O-cycle

The FMG method assumes that the information from a finer grid is necessary to guide the solver on the coarser grid to the correct solution. This comes at the cost of additional V- and W-cycles. In our experiments we found that these additional V- and W-cycles are redundant for most input images and a similar reconstruction quality can be obtain with less computational effort, which is desirable for real-time applications. To achieve a similar reconstruction quality compared to FMG constructed with V- or W-cycles, we introduce a new approach, based on what we call O-cycle (or null-cycle). The idea is to perform significantly more iterations on the coarse grids, which have a low computational cost since the image resolution is smaller, and thereby increase the likelihood of convergence on the coarse grid. Once convergence on a coarse grid is obtained, the algorithm processes the next finer grid and never returns to the coarser one.

As illustrated in Fig. 3, the iteration number m now is dependend on the current multigrid level. In particular, we have obtained good results for $m = m_1 \cdot (n^p)$, where $p = \{1, 2\}$ and $n \in [1, N]$ with $n = 1$ for the finest grid and $n = N$ for the coarsest grid. The user parameter m_1 defines the number of iterations on the finest grid.

4 Multi-Level Adaptive Technique (MLAT)

In this section, a multi-level adaptive technique is described that reduces the computational effort of the variational solver and at the same time retains a high reconstruction quality. From a coarser to a finer grid, we usually only need to refine the reconstruction at some regions of interest in the image. This means that with a static grid we waste resources on grid nodes that do not improve the resulting solution. Therefore, we employ a non-static grid structure.

The whole process starts on the coarsest grid. After finding a solution on the coarsest grid using FAS and O-cycles, we use this solution to identify the grid nodes that need refinement. Therefore, we evaluate the residual error of the energy functional in Eq. (4) as well as the spatial gradient of the solution. These two criteria are used to detect peculiarities of the solution. If either the residual $E(u(x, y)) > \epsilon_E$ or the spatial gradient $||\nabla u(x, y)||^2 > \epsilon_\nabla$, the grid cell is refined. Thereby, the thresholds ϵ_E and ϵ_∇ are user-defined parameters.

As shown in Fig. 4, the number of nodes is always upsampled by to factor of 2 when going from a coarser to a finer grid. Thereby, the area covered by a finer grid cell, can only be a part of the area covered with a coarser grid cell. For example, let us assume that the red arrow in Fig. 4 marks a local image peculiarity. Then the algorithm will employ finer grids at this location.

Despite the irregular grid structure, the differential equations can still be solved with a multigrid solver. This is done in a similar way as with regular grids. As shown in Fig. 4, there are different types of nodes:

1. The nodes belonging only to the grid G_H, which will not be refined and no further calculation must be performed for them.
2. The nodes of G_H belonging at the same time to G_h, which will be used for gaining the solution correction with the FAS at nodes on the finer grid G_h.
3. The boundary nodes of the finer grid G_h. In order to connect the solutions from both grids, these nodes are initialized from the coarser grid but are not altered during optimization.

The multi-level grid adaptation can not only be used with O-cycles (which we use for our real-time results), it can also be used with a classical FMG approach. In the areas on the coarse grid G_H, which are not covered with the fine grid G_h, we need not only the error, but the whole solution. Therefore, even if we have a linear problem (e.g., when using Tichonov regularizer in the smoothness term) the FAS must be employed.

Let us consider the error of restriction, when coming from a finer grid to a coarser one. Using Eq. (17), we can write

$$f_H = I_h^H f_h + \tau_h^H \quad , \tag{19}$$

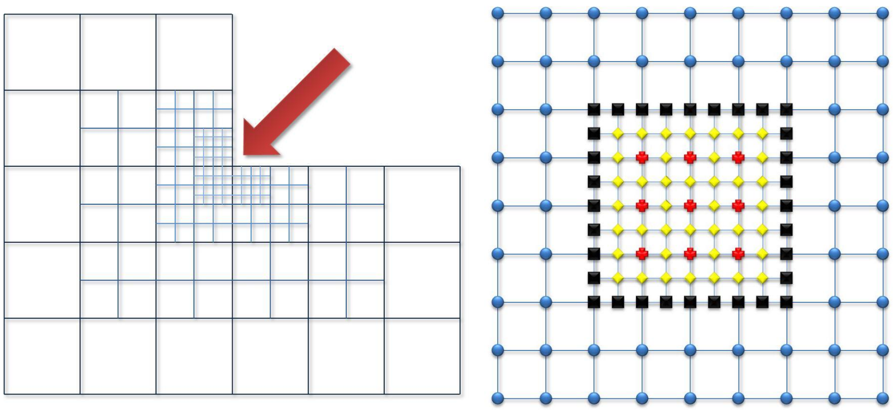

Fig. 4. Irregular grids: **left**: nodes concentration near to the local image peculiarity; **right**: structure of two grid levels G_h and G_H: common nodes of G_h and G_H (red crosses); nodes which belong only to G_h (yellow rhombi); nodes which belong only to G_H (blue circles); boundary nodes of G_h (black squares).

where $\tau_h^H = I_h^H L_h u_h^i - L_H I_h^H u_h^i$ is the error of restriction or the transfer correction from the finer to a coarser grid. This transfer correction for the equation on the coarser level provides a measure for the co-occurrence between the solutions on the coarse and fine grid. When solving the equation $L_H u_H = -F_H$ on the coarse grid G_H, the term F_H is given by Eq. (19), but in the areas where the corresponding node on the fine grid G_h is not available, we assume that τ_h^H is equal to zero.

5 Regularizer adaptation

We found in our experiments that the reconstruction results can be further improved, if the applied regularizer for each pixel is changed locally within the iteration process. For each pixel we always start with the Tichonov regularizer. Based on thresholds for $E(u(x,y))$ and $||\nabla u(x,y)||^2$ the algorithm decides when to switch to the Charbonnier or Perona-Malik regularizer. Furthermore, we adapt the parameters λ and φ (see Eqs. (6), (8), and (9)). The choice for these parameters depends on the regularizer, the multi-grid level, as well as $||\nabla u(x,y)||^2$ and $E(u(x,y))$. The best parameters for the actual situation are trained off-line by Monte-Carlo simulation on stereo pictures which have a similar characteristic as the test images, e.g., similar range of disparities, comparable image resolution, and equivalent scene illumination conditions. Once these parameters are found they are kept fixed during real-time experiments.

6 Evaluation

To evaluate our novel multigrid variational solver with MLAT, we use the standard stereo data sets provided by the Middlebury Stereo evaluation website [23]. These datasets include stereo images as well as ground-truth disparity maps. The resolution of the provided stereo images is approximately 450 × 375 pixels and the disparities are in the range [0; 22] pixels.

The Middlebury stereo evaluation website provides a convenient and objective way to evaluate the accuracy of the reconstruction by the percentage of bad pixels (see more details in [24]). As can be seen in Tab. 1, our approach is among the most accurate methods. Furthermore, those methods, which have a higher accuracy, are not marked to be real-time approaches like ours.

	Tsukuba			Venus			Teddy			Cones			Average Percent
Algorithm	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc	of bad pixels
DoubleBP	0.83	1.24	4.49	0.10	0.35	1.46	1.41	4.13	4.73	1.71	7.02	5.16	2.72
CoopRegion	0.77	1.00	4.14	0.11	0.18	1.53	2.14	3.41	6.61	2.10	5.95	6.24	2.80
...													
our method	1.33	3.13	6.94	0.27	1.07	3.16	1.30	2.30	3.87	2.31	3.43	6.90	**3.00**
...													
MultiResGC	0.67	1.05	3.64	0.22	0.46	2.97	4.20	7.13	11.6	3.22	8.80	8.07	4.30
RealtimeBP	1.25	3.04	6.66	0.63	1.53	7.68	5.68	8.27	10.2	2.90	9.11	8.27	5.43
RealTimeGPU	1.34	3.27	7.17	1.02	1.90	12.4	3.90	8.65	10.4	4.37	10.8	12.3	6.46
Infection	6.34	7.81	22.8	2.70	3.66	26.0	12.8	18.3	33.5	10.7	16.6	30.1	15.9

Table 1. Excerpt from the evaluation table generated by the Middlebury stereo evaluation webpage (error threshold = 2 pixels).

In Fig. 5 the corresponding results for the Tsukuba, Venus, Teddy, and Cones scene are shown. The timings for our approach are given in Tab. 2, where we compare the performance of FMG with V- and W-cycles or O-cycles. All speed measurements are carried out using a standard desktop PC with a 2.83 GHz Intel Pentium CPU executing C++ code.

	Speed [fps]		
Scene	FMG:V-cycles	FMG:W-cycles	O-cycles
Tsukuba	0,6	1,3	2,15
Venus	1,15	1,85	3,75
Teddy	1	1,3	3,3
Cones	0,75	2,1	3,5

Table 2. Computational effort of our algorithm using FMG with V- and W-cycles or O-cycles at the same reconstruction quality, as used in Tab. 1.

7 Future Work & Conclusion

In this paper, we have introduced a combination of multigrids and a multi-level adaptive technique for the variational approach to reconstruct a disparity field. Furthermore, a regularizer adaptation technique was proposed. This allows the

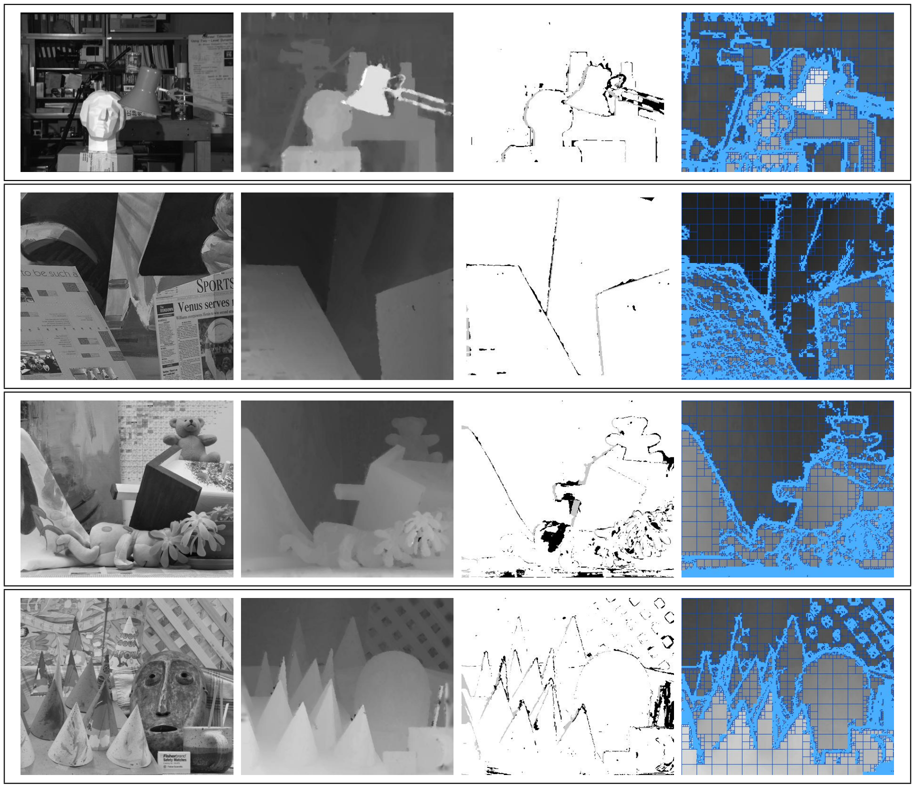

Fig. 5. Top to bottom: Tsukuba, Venus, Teddy, and Cones scene; Left to right: the right image of the stereo pair, solution disparity map, bad pixels (absolute disparity error > 1.0 pixel), the finest MLAT grid.

variational solver to achieve a real-time performance even on a CPU. The gained reconstruction quality is competitive to other state-of-the-art approaches that require more computation time.

In future, we are going to develop and implement a parallel version of our algorithm, which could be capable to run on multiple CPUs or a GPU with stream processing technology. Another direction of future work is to extend our method of 1-dimensional disparity estimation to the problem of 2-dimensional optic flow reconstruction.

References

1. Mémin, E., Pérez, P.: Dense estimation and object-based segmentation of the optical flow with robust techniques. IEEE Transactions on Image Processing **7** (1998) 703–719
2. Brox, T., Bruhn, A., Papenberg, N., Weickert, J.: High accuracy optical flow estimation based on a theory for warping. In: ECCV. Volume 3024 of LNCS., Springer (2004) 25–36

3. Brandt, A.: Multi-level adaptive solutions to boundary-value problems. Mathematics of Computation **31** (1977) 333–390
4. Fedorenko, R.: Relaxation method for solving elliptic differential equations. Journal of computational mathematics and mathematical phisics. **1** (1961) 922–927
5. Brandt, A.: Multilevel adaptive computations in fluid dynamics. AIAA Journal **18** (1980) 1165–1172
6. Stüben, K., Trottenberg, U.: Multigrid methods: Fundamental algorithms, model problem analysis and applications. Lecture Notes in Mathematics **960** (1982) 1–176
7. Hackbusch, W.: Multigrid Methods and Applications. Springer, New York (1985)
8. Bruhn, A., Weickert, J., Kohlberger, T., Schnörr, C.: A multigrid platform for real-time motion computation with discontinuity-preserving variational methods. Int. J. Comput. Vision **70** (2006) 257–277
9. Mémin, E., Pérez, P.: A multigrid approach for hierarchical motion estimation. IEEE International Conference on Computer Vision (1998) 933–938
10. Kosov, S.: 3D map reconstruction with variational methods. Master thesis, Saarland University (2008)
11. Brandt, A.: Multi-level adaptive technique (MLAT) for fast numerical solution to boundary value problems. Lecture Notes in Physics **18** (1973) 82–89
12. Condell, J., Scotney, B., Morrow, P.: Adaptive grid refinement procedures for efficient optical flow computation. Int. J. Comput. Vision **61** (2005) 31–54
13. Papenberg, N., Bruhn, A., Brox, T., Didas, S., Weickert, J.: Highly accurate optic flow computation with theoretically justified warping. Int. J. Comput. Vision **67** (2006) 141–158
14. Meijering, E.: A chronology of interpolation: From ancient astronomy to modern signal and image processing. In: Proceedings of the IEEE. (2002) 319–342
15. Horn, B.K.P., Schunck, B.G.: Determining optical flow. Artificial Intelligence **17** (1981) 185–203
16. Charbonnier, P., Aubert, G., Blanc-Ferraud, M., Barlaud, M.: Two deterministic half-quadratic regularization algorithms for computed imaging. International Conference on Image Processing **2** (1994) 168–172
17. Cohen, I.: Nonlinear variational method for optical flow computation. In: Eighth Scandinavian Conference on Image Analysis. Volume 1. (1993) 523–530
18. Schnörr, C.: Segmentation of visual motion by minimizing convex non-quadratic functionals. In: Twelfth International Conference on Pattern Recognition. Volume A., Jerusalem, Israel, IEEE Computer Society Press (1994) 661–663
19. Weickert, J., Schnörr, C.: A theoretical framework for convex regularizers in pde-based computation of image motion. Int. J. Comput. Vision **45** (2001) 245–264
20. Cheng, A., Cheng, D.T.: Heritage and early history of the boundary element method. Engineering Analysis with Boundary Elements **29** (2005) 268–302
21. Ortega, J.M., Rheinboldt, W.C.: Iterative solution of nonlinear equations in several variables. Society for Industrial and Applied Mathematics, USA (2000)
22. Young, D.: Iterative Solution of Large Linear Systems. Academic Press, New York (1971)
23. Scharstein, D., Szeliski, R.: High-accuracy stereo depth maps using structured light. In: Proc. Computer Vision and Pattern Recognition. (2003) I: 195–202, datasets available on http://vision.middlebury.edu/stereo/
24. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. International Journal of Computer Vision **47** (2001) 7–42