

A New Triangulation-Based Method for Disparity Estimation in Image Sequences

Dimitri Bulatov, Peter Wernerus, and Stefan Lang

Research Institute for Optronics and Pattern Recognition,
Gutleuthausstr. 1, 76275 Ettlingen, Germany
{bulatov,wernerus,lang}@fom.fgan.de

Abstract. We give a simple and efficient algorithm for approximating computation of disparities in a pair of rectified frames of an image sequence. The algorithm consists of rendering a sparse set of correspondences, which are triangulated, expanded and corrected in the areas of occlusions and homogeneous texture by a color distribution algorithm. The obtained approximations of the disparity maps are refined by a semi-global algorithm. The algorithm was tested for three data sets with rather different data quality. The results of the performance of our method are presented and areas of applications and future research are outlined.

Keywords: Color, dense, depth map, disparity map, histogram, matching, reconstruction, semi-global, surface, triangulation.

1 Introduction

Retrieving dense three-dimensional point clouds from monocular images is the key-issue in a large number of computer vision applications. In the areas of navigation, civilian emergency and military missions, the need for fast, accurate and robust retrieving of disparity maps from small and inexpensive cameras is rapidly growing. However, the matching process is usually complicated by low resolution, occlusion, weakly textured regions and image noise. In order to compensate these negative effects, robust state-of-the-art methods such as [2], [10], [13], [20], are usually global or semi-global, i.e. the computation of matches is transformed into a global optimization problem. Therefore all these methods require high computational costs. On the other hand, the local methods, such as [3], [12], are able to obtain dense sets of correspondences, but the quality of the disparity maps obtained by these methods is usually below the quality achieved by global methods.

In our applications, image sequences are recorded with handheld or airborne cameras. Characteristic points are found by means of [8] or [15] and the fundamental matrices are computed from the point correspondences by robust algorithms (such as a modification of RANSAC [16]). As a further step, the structure and motion can be reconstructed using tools described in [9]. If the cameras are not calibrated, the reconstruction can be carried out in a projective coordinate system and afterwards upgraded to a metric reconstruction using methods

of auto-calibration ([9], Chapter 19). The point clouds thus obtained have extremely irregular density: Areas with a sparse density of points arising from homogeneous regions in the images are usually quite close to areas with high density resulting from highly textured areas. In order to reconstruct the surface of the unknown terrain, it is extremely important to obtain a homogeneous density of points. In this paper, we want to enrich the sparse set of points by a dense set, i.e. to predict the position in space of (almost) every pixel in every image. It is always useful to consider *all* available information in order to facilitate the computation of such dense sets. Beside methods cited above and those which were tested in the survey due to Scharstein and Szeliski [21], there are several methods which combine the approaches of disparity estimation and surface reconstruction. In [1], for example, the authors propose to initialize layers in the images which correspond to (almost) planar surfaces in space. The correspondences of layers in different images are thus given by homographies induced by these surfaces. Since the surface is not really piecewise planar, the authors introduce the distances between the point on the surface and its planar approximation at each pixel as additional parameters. However, it is difficult to initialize the layers without prior knowledge. In addition, the algorithm could have problems in the regions which belong to the same segment but have depth discontinuities. In [19], the Delaunay triangulation of points already determined is obtained; [18] proposes using edge-flip algorithms in order to obtain a better triangulation since the edges of Delaunay-triangles in the images are not likely to correspond to the object edges. Unfortunately, the sparse set of points usually produces a rather coarse estimation of disparity maps; also, this method can not detect occlusions. In this paper, we will investigate to what extent disparity maps can be initialized by triangular meshes in the images.

In the method proposed here, we will use the set of sparse point correspondences $\mathbf{x} = \mathbf{x}_1 \leftrightarrow \mathbf{x}_2$ to create initial disparity maps from the support planes for the triangles with vertices in \mathbf{x} . The set \mathbf{x} will then be iteratively enriched. Furthermore, in the areas of weak texture and gradient discontinuities, we will investigate to what extent the color distribution algorithms can detect the outliers and occlusions among the triangle vertices and edges. Finally, we will use the result of the previous steps as an initial value for the global method [10], which uses a random disparity map as input. The necessary theoretical background will be described in Sec. 2.1 and the three steps mentioned above in Sec. 2.2, 2.3, and 2.4. The performance of our method is compared with semi-global algorithms without initial estimation of disparities in Sec. 3. Finally, Sec. 4 provides the conclusions and the research fields of the future work.

2 Our Method

2.1 Preliminaries

Suppose that we have obtained the set of sparse point correspondences and the set of camera matrices in a projective coordinate system, for several images of an airborne or handheld image sequence. The fundamental matrix can be

extracted from any pair of cameras according to the formula (9.1) of [9]. In order to facilitate the search for correspondences in a pair of images, we perform *image rectification*, i.e. we transform the images and points by two homographies to make the corresponding points (denoted by $\mathbf{x}_1, \mathbf{x}_2$) have the same y -coordinates. In the rectification method we chose, [14], the epipoles $\mathbf{e}_1, \mathbf{e}_2$ must be transformed to the point at infinity $(1, 0, 0)^T$, therefore $\mathbf{e}_1, \mathbf{e}_2$ must be bounded away from the image domain in order to avoid significant distortion of the images. We can assume that such a pair of images with enough overlap can be chosen from the entire sequence. We also assume that the percentage of outliers among the points in $\mathbf{x} = \mathbf{x}_1$ is low because most of the outliers are supposed to be eliminated by robust methods. Finally, we remark that we are not interested to compute correspondences of all points inside of the overlap of both rectified images (which will be denoted by I_1 respectively I_2) but restrict ourselves to the convex hull of the points in \mathbf{x} . Computing point correspondences of pixels outside of the convex hulls does not make much sense since they often do not lie in the overlap area and, especially in the case of uncalibrated cameras, suffer more from the lens distortion effects. One should better use another pair of images to compute disparities for these points.

Now suppose we have a partition of \mathbf{x} into triangles. Hereafter, $\check{\mathbf{p}}$ denotes the homogeneous representation of a point \mathbf{p} ; T represents a triple of integer numbers; thus, $\mathbf{x}_{1,T}$ are the columns of \mathbf{x}_1 specified by T . By $\mathbf{p}_1 \in T$, we will denote that the pixel \mathbf{p}_1 in the first rectified image lies in triangle $\mathbf{x}_{1,T}$. Given such a partition, every triangle can be associated with its support plane which induces a triangle-to-triangle homography. This homography only possesses three degrees of freedom which are stored in its first row since the displacement of a point in a rectified image only concerns its x -coordinate.

Result 1: Let $\mathbf{p}_1 \in T$ and let $\mathbf{x}_{1,T}, \mathbf{x}_{2,T}$ be the coordinates of the triangle vertices in the rectified images. The homography induced by T maps \mathbf{x}_1 onto the point $\mathbf{p}_2 = (X_2, Y)$, where $X_2 = \mathbf{v}\check{\mathbf{p}}_1$, $\mathbf{v} = x_{2,T}(\check{\mathbf{x}}_{1,T})^{-1}$, and $x_{2,T}$ is the row vector consisting of x -coordinates of $\mathbf{x}_{2,T}$.

Proof: Since triangle vertices $\mathbf{x}_{1,T}, \mathbf{x}_{2,T}$ are corresponding points, their correct locations are on the corresponding epipolar lines. Therefore they have pairwise the same y -coordinates. Moreover, the epipole is given by $\mathbf{e}_2 = (1, 0, 0)^T$ and the fundamental matrix is $F = [\mathbf{e}_2]_{\times}$. Inserting this information into Result 13.6 of [9], p.331 proves, after some simplifications, the statement of **Result 1**.

Determining and storing the entries of $\mathbf{v} = \mathbf{v}_T$ for each triangle, optionally refining \mathbf{v} for the triangles in the big planar regions by error minimization and calculating disparities according to **Result 1** provide, in many cases, a coarse approximation for the disparity map in the areas where the surface is approximately piecewise planar and does not have many self-occlusions.

2.2 Initialization of Disparity Maps Given from Triangulations

Starting from the Delaunay-Triangulation obtained from several points in the image, we want to expand, because the first approximation is too coarse, the quantity of points. Since the fundamental matrix obtained from structure-from-motion algorithms is noisy, it is necessary to search for correspondences not only in the direction along the epipolar lines but also in the vertical direction. We suppose that the distance of a pair of corresponding points to the corresponding epipolar lines to be bounded by 1 pel. Therefore, given a point $\mathbf{p}_1 = (X_1, Y_1) \in T$, we consider the search window in the second image given by:

$$\begin{aligned} W_s &= [X_1 + X_{\min}; X_1 + X_{\max}] \times [Y - 1; Y + 1], \\ X_{\min} &= \max(d_{\min} - \varepsilon, \min(s_T)), X_{\max} = \min(d_{\max} + \varepsilon, \max(s_T)) \end{aligned} \quad (1)$$

where $\varepsilon = 3$ is a fixed scalar, s_T are the x -coordinates of at most six intersection points between the epipolar lines at $Y, Y - 1, Y + 1$ and the edges of $\mathbf{x}_{1,T}$ and d_{\min}, d_{\max} are the estimates of smallest and biggest possible disparities which can be obtained from the point coordinates.

The search for correspondent points succeeds by means of the *normalized cross correlation* (NCC) algorithm between the quadratic window $I_1(W(\mathbf{p}_1))$ of size between 5 and 10 pixels and $I_2(W_s)$. However, in order to avoid including mismatches into the set of correspondences, we impose three filters on the result of the correlation. A pair of points $\mathbf{p}_1 = (X_1, Y)$ and $\mathbf{p}_2 = (X_2, Y)$ is added to the set of correspondences if and only if: 1. the correlation coefficient c_0 of the winner exceeds a user-specified value c_{\min} ($= 0.7-0.9$ in our experiments), 2. the windows have approximately the same luminance, i. e. $\|I_1(W(\mathbf{p}_1)) - I_2(W(\mathbf{p}_2))\|_1 < |W|u_{\max}$ where $|W|$ is the number of pixels in the window and $u_{\max} = 15$ in our experiments, and, 3. in order to avoid erroneous correspondences along epipolar lines which coincide with edges in the images, we eliminate the matches where the ratio of the maximal correlation coefficient in the sub-windows

$$([X_{\min}; X_2 - 1] \cup [X_2 + 1; X_{\max}]) \times [Y - 1; Y + 1], \quad (2)$$

and c_0 (second-best to best) exceeds a threshold γ , which is usually 0.9. Here X_{\min}, X_{\max} in (2), are specified according to (1). An alternative way to handle the mismatches is using more cameras, as described, for example, in [7]. Further research on this topic will be part of our future work.

Three concluding remarks will be given at the end of present subsection:

1. It is not necessary to use *every* point in *every* triangle for determining corresponding points. It is recommendable not to search corresponding points in lowly textured areas but to take the points with a maximal (within a small window) response of a suitable point-detector. In our implementation, it is the *Harris-operator*, see [8], so the *structural tensor* A for a given image as well as the "cornerness" term $\text{trace}(A) - 0.04 \det(A)$ can be precomputed and stored once for all.

2. It also turned out to be helpful to subdivide only triangles with area exceeding a reasonable threshold (100-500 pel^2 in our experiments) and *non-compatible* with the surface, which means that the highest correlation coefficient for the barycenter \mathbf{p}_1 of the triangle T was obtained at X_2 and for $\mathbf{v} = \mathbf{v}_T$ computed according to **Result 1**, we have $|\mathbf{v}\hat{\mathbf{p}} - X_2| > 1$. After obtaining correspondences, the triangulation could be refined by using edge-flipping algorithms, but in the current implementation, we do not follow this approach.
3. The coordinates of corresponding points can be refined to subpixel values, according to one of four methods discussed in [23]. For the sake of computation time, subpixel coordinates for correspondences are computed according to correlation parabolas. We denote by c_- and c_+ the correlation values in the pixels left and right from X_2 . The correction term \hat{X}_2 in x-direction is then given by:

$$\hat{X}_2 = X_2 - \frac{c_+ - c_-}{2(c_- + c_+ - 2c_0)}.$$

Also the value of X_2 is corrected for triangles compatible with the surface according to **Result 1**.

2.3 Color-Distribution Algorithms for Occlusion Detection

The main drawback of the initialization with an (expanded) set of disparities are the outliers in the data as well as the occlusions since the sharp edge of depth in the triangle on the left and on the right of edge with disparity discontinuities will be blurred. While the outliers can be efficiently eliminated by means of disparities of their neighbors (a procedure which we apply once before and once after the expansion), in the case of occlusions, we shall investigate how the color-distribution algorithms can restore the disparities at the edges of discontinuities.

At present, we mark all triangles for which the standard deviation of disparities at the vertexes exceeds a user-specified threshold ($\sigma_0 = 2$ in our experiments) as unfeasible. Given a list of unfeasible triangles, we want to find similar triangles in the neighborhood. In our approach this similarity is based on color distribution represented by three histograms, each for a different color in the color space RGB (red, green and blue).

A histogram is defined over the occurrence of different color values of the pixels inside the considered triangle T . Each color contains values from 0 to 255, thus each color histogram has b bins with a bin size of $256/b$. Let the number of pixels in a triangle be n . In order to obtain the probability of this distribution and to make it independent of the size of the triangle, we obtain for the i -th bin of the *normalized histogram*

$$H_T(i) = \frac{1}{n} \cdot \# \left\{ \mathbf{p} \mid \mathbf{p} \in T \text{ and } \frac{256 \cdot i}{b} \leq I_1(\mathbf{p}) < \frac{256 \cdot (i + 1)}{b} \right\}.$$

The three histograms H_T^R, H_T^G, H_T^B represent the color distribution of the considered triangle. It is also useful to split big, inhomogeneous, unfeasible triangles

into smaller ones. To perform splitting, characteristic edges ([4]) are found in every candidate triangle and saved in form of a binary image $G(\mathbf{p})$.

To find the line with maximum support, we apply the radon transformation ([6]) to $G(\mathbf{p})$:

$$\check{G}(u, \varphi) = R\{G(\mathbf{p})\} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} G(\mathbf{p}) \delta(\mathbf{p}^T \mathbf{e}_\varphi - u) d\mathbf{p}$$

with the Dirac delta function $\delta(x) = \infty$ if $x = 0$ and 0 otherwise and line parameters $\mathbf{p}^T \mathbf{e}_\varphi - u$, where $\mathbf{e}_\varphi = (\cos\varphi, \sin\varphi)^T$ is the normal vector and u the distance to origin. The strongest edge in the triangle is found if the maximum of $\check{G}(u, \varphi)$ is over a certain threshold for the minimum line support. This line intersects the edges of the considered triangle T in two intersection points. We disregard intersection points too close to a vertex of T . If new points were found, the original triangle is split in two or three smaller triangles. These new smaller triangles consider the edges in the image.

Next the similarity of two neighboring triangles has to be calculated by means of the color distribution. Two triangles are called neighbors if they share at least one vertex. There are a lot of different approaches measuring the distance between histograms [5]. In our case we define the distance of two neighboring triangles T_1 and T_2 as follows:

$$d(T_1, T_2) = w_R \cdot d(H_{T_1}^R, H_{T_2}^R) + w_G \cdot d(H_{T_1}^G, H_{T_2}^G) + w_B \cdot d(H_{T_1}^B, H_{T_2}^B) \quad (3)$$

where w_R, w_G, w_B are different weights for the colors. The distance between two histograms in (3) is the sum of absolute differences of their bins.

In the next step, the disparity in the vertices of unfeasible triangles will be corrected. Given an unfeasible triangle T_1 , we define

$$T_2 = \operatorname{argmin}_T \{d(T_1, T) | \operatorname{area}(T) > A_0, d(T_1, T) < c_0 \text{ and } T \text{ is not unfeasible}\},$$

where $c_0 = 2$, $A_0 = 30$ and $d(T_1, T)$ is computed according to (3). If such T_2 does exist, we recompute the disparities of pixels in T_1 with \mathbf{v}_{T_2} according to **Result 1**. Usually this method performs rather well as long as the assumption holds that neighboring triangles with similar color information lie indeed in the same planar region of the surface.

2.4 Refining of the Results with a Global Algorithm

Many dense stereo correspondence algorithms improve their disparity map estimation by minimizing disparity discontinuities. The reason is that neighboring pixels probably map to the same surface in the scene, and thus their disparity should not differ much. This could be achieved by minimizing the energy

$$E(D) = \sum_{\mathbf{p}} \left\{ C(\mathbf{p}, d_{\mathbf{p}}) + P_1 \cdot N_{\mathbf{p}}(1) + P_2 \cdot \sum_{i=2}^{\infty} N_{\mathbf{p}}(i) \right\}, \quad (4)$$

where $C(\mathbf{p}, d)$ is the cost function for disparity $d_{\mathbf{p}}$ at pixel \mathbf{p} ; P_1, P_2 , with $P_1 < P_2$ are penalties for disparity discontinuities and $N_{\mathbf{p}}(i)$ is the number of pixels \mathbf{q} in

the neighborhood of \mathbf{p} for which $|d_{\mathbf{p}} - d_{\mathbf{q}}| = i$. Unfortunately, the minimization of (4) is NP-hard. Therefore an approximation is needed. One approximation method yielding good results, while simultaneously being computational fast compared to many other approaches, was developed by Hirschmüller [10].

This algorithm, called Semi-Global Matching (SGM), uses mutual information for matching cost estimation and a path approach for energy minimization. The matching cost method is an extension of the one suggested in [11]. The accumulation of corresponding intensities to a probability distribution from an initial disparity map is the input for the cost function to be minimized. The original approach is to start using a random map and iteratively calculate improved maps, which are used for a new cost calculation. To speed up this process, Hirschmüller first iteratively halves the original image by downsampling it, thus creating image pyramids. The random initialization and first disparity approximation take place at lowest scale and are iteratively upsampled until the original scale is achieved.

To approximate the energy functional $E(D)$, paths from 16 different directions leading into one pixel are accumulated. The cost for one path in direction \mathbf{r} ending in pixel \mathbf{p} is recursively defined as: $L_{\mathbf{r}}(\mathbf{p}, d) = C(\mathbf{p}, d)$ for \mathbf{p} near image border and

$$L_{\mathbf{r}}(\mathbf{p}, d) = C(\mathbf{p}, d) + \min[L_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, d), L_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, d \pm 1) + P_1, \min_i (L_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, i)) + P_2]$$

otherwise. The optimal disparity for pixel \mathbf{p} is then determined by summing up costs of all paths of the same disparity and choosing the disparity with the lowest result. Our method comes in as a substitution for the random initialization and iterative improvement of the matching cost. The disparity map achieved by our algorithm is simply used to compute the cost function once without iterations.

In the last step, the disparity map in the opposite direction is calculated. Pixels with corresponding disparities are considered correctly estimated, the remaining pixels occluded.

3 Results

In this section, results from three data sets will be presented. The first data set is taken from the well known *Tsukuba* benchmark-sequence. No camera rectification was needed since the images are already aligned. Although we do not consider this image sequence as characteristic for our applications, we decided to demonstrate the performance of our algorithm for a data set with available ground truth. In the upper row of Fig. 1, we present the ground truth, the result of our implementation of [10] and the result of depth maps estimation initialized with ground truth. In the bottom row, one sees from left to right, the result of Step 1 of our algorithm described in Sec. 2.2, the correction of the result as described in Step 2 (Sec. 2.3) and the result obtained by Hirschmüller algorithm as described in Sec. 2.4 with initialization. The disparities are drawn in pseudo-colors and with occlusions marked in black.

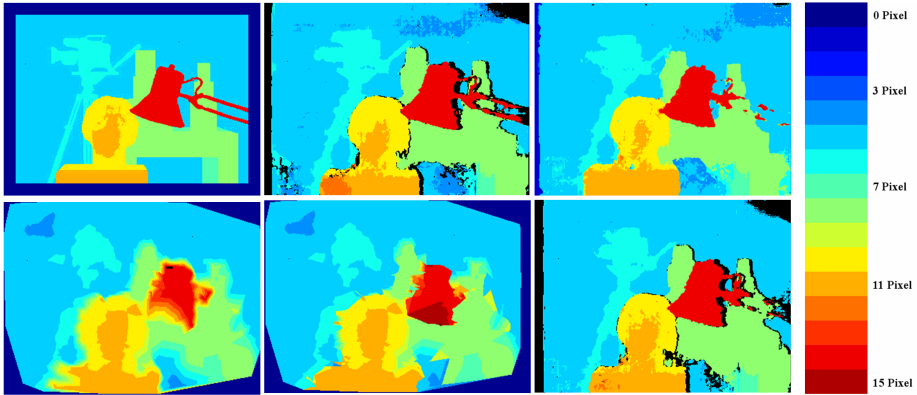


Fig. 1. Top row, left to right: the ground truth from the sequence *Tsukuba*, the result of disparity map rendered by [10], the result of disparity map rendered by [10] initialized with ground truth. Bottom row, left to right: initialization of the disparity map created Step 1 by our algorithm, initialization of the disparity map created Step 2 by our algorithm and the result of [10] with initialization. Right: color scale representing different disparity values.

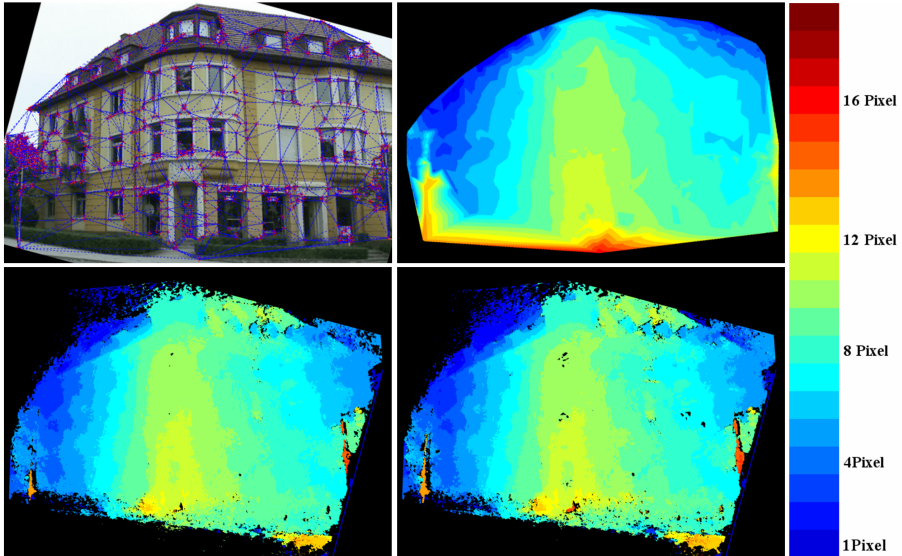


Fig. 2. Top row: left: a rectified image from the sequence *Old House* with the mesh from the point set in the rectified image; right: initialization of the disparity map created by our algorithm. Bottom row: results of [10] with and without initialization. Right: color scale representing disparity values.

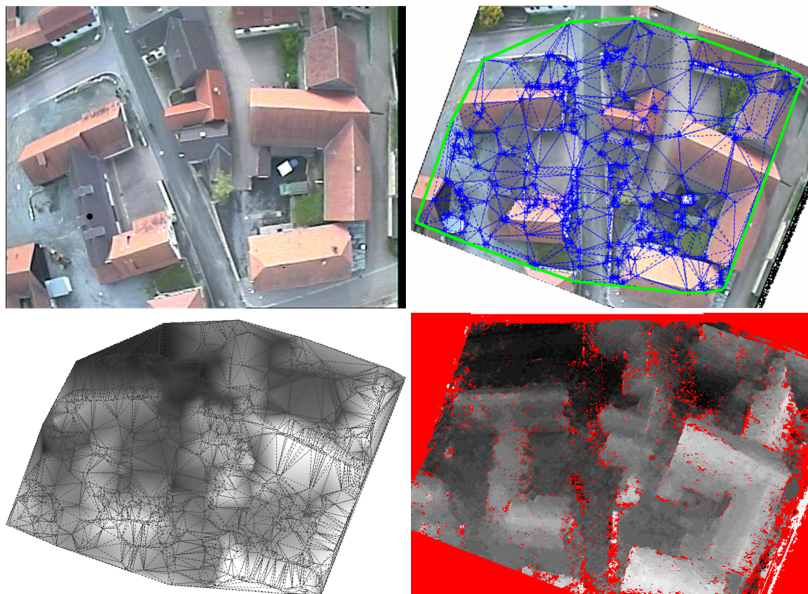


Fig. 3. Top row: left: a frame from the sequence *Bonnland*; right: the rectified image and mesh from the point set. Bottom row: initialization of the disparity map created by our algorithm with the expanded point set and the result of [10] with initialization.

The data set *Old House* shows a view of a building in Ettlingen, Germany, recorded by a handheld camera. In the top row of Fig. 2, the rectified image with the triangulated mesh of points detected with [8] as well as the disparity estimation by our method is shown. The bottom row shows the results of the disparity estimation with (left) and without (right) initialization drawn with pseudo-colors and with occlusions marked in black.

The data set *Bonnland* was taken from a small unmanned aerial vehicle which carries a small inexpensive camera on board. The video therefore suffers from reception disturbances, lens distortion effects and motion blur. However, obtaining fast and feasible depth information from these kinds of sequences is very important for practical applications. In the top row of Fig. 3, we present a frame of the sequence and the rectified image with triangulated mesh of points. The convex hull of the points is indicated by a green line. In the bottom row, we present the initialization obtained from the expanded point set as well as the disparity map computed by [10] with initialization and occlusions marked in red.

The demonstrated results show that in many practical applications, the initialization of disparity maps from already available point correspondences is a feasible tool for disparity estimation. The results are the more feasible, the more the surface is piecewise planar and the less occlusions as well as segments of

the same color lying in different support planes there are. The algorithm maps well triangles of homogeneous texture (compatible with the surface), while even a semi-global method produces mismatches in these areas, as one can see in the areas in front of the house in Fig.2 and in some areas of Fig.3. The results obtained with the method described in Sec.2.2 and 2.3 usually provide an acceptable initialization for a semi-global algorithm. The computation time for our implementation of [10] *without initialization* was around 80 seconds for the sequence *Bonnland* (two frames of size 823×577 pel, the algorithm run twice in order to detect occlusions) and *with initialization* about 10% faster. The difference of elapsed times is approximately 7 seconds and it takes approximately the same time to expand the given point set and to compute the distance matrix for correcting unfeasible triangles.

4 Conclusions and Future Work

The results presented in this paper indicate that it is possible to compute acceptable initialization of the disparity map from a pair of images by means of a sparse point set. The computing time of the initialization does not depend on the disparity range and is less dependent on the image size as state-of-the-art local and global algorithms since a lower point density not necessarily means worse results. Given an appropriate point detector, our method is able to consider pairs of images with different radiometric information. In this contribution, for instance, we extract depths maps from different frames of the same video sequence, so the correspondences of points are likely to be established from intensity differences; but in the case of pictures with significantly different radiometry, one can take the SIFT-operator ([15]) as a robust point detector and the cost function will be given by the scalar product of the descriptors.

The enriched point clouds may be used as input for scene and surface reconstruction algorithms. These algorithms benefit from a regular density of points, which makes the task of fast and accurate retrieving additional 3D-points (especially) in the areas of low texture extremely important. It is therefore necessary to develop robust color distribution algorithms to perform texture analysis and to correct unfeasible triangles, as we have indicated in Sec.2.3.

The main drawback of Sec.2.2 are outliers among the new correspondences as well as occlusions which are not always corrected at later stages. Since the initialization of disparities is spanned from triangles, the complete regions around these points will be given wrong disparities. It has been shown that using redundant information given from more than two images ([22], [7]) can significantly improve the performance; therefore we will concentrate our future efforts on integration of multi-view-systems into our triangulation networks. Another interesting aspect will be the integration of 3D-information given from calibrated cameras into the process of robust determination of point correspondences, as described, for example, in [17], [7]. Moreover, we want to investigate how the expanded point clouds can improve the performance of the state-of-the-art surface reconstruction algorithms.

References

1. Baker, S., Szeliski, R., Anandan, P.: A layered approach to stereo reconstruction. In: Computer Vision and Pattern Recognition (CVPR), pp. 434–441 (1998)
2. Bleyer, M., Gelautz, M.: Simple but Effective Tree Structures for Dynamic Programming-based Stereo Matching. In: International Conference on Computer Vision Theory and Applications (VISAPP), (2), pp. 415–422 (2008)
3. Boykov, Y., Veksler, O., Zabih, R.: A variable window approach to early vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 20(12), 1283–1294 (1998)
4. Canny, J.A.: Computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 8(6), 679–698 (1986)
5. Cha, S.-H., Srihari, S.N.: On measuring the distance between histograms. *Pattern Recognition* 35(6), 1355–1370 (2002)
6. Deans, S.: *The Radon Transform and Some of Its Applications*. Wiley, New York (1983)
7. Furukawa, Y., Ponce, J.: Accurate, Dense, and Robust Multi-View Stereopsis. In: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Anchorage, USA, pp. 1–8 (2008)
8. Harris, C.G., Stevens, M.J.: A Combined Corner and Edge Detector. In: Proc. of 4th Alvey Vision Conference, pp. 147–151 (1998)
9. Hartley, R., Zisserman, A.: *Multiple View Geometry in Computer Vision*. Cambridge University Press, Cambridge (2000)
10. Hirschmüller, H.: Accurate and Efficient Stereo Processing by Semi-Global Matching and Mutual Information. In: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), (2), San Diego, USA, pp. 807–814 (2005)
11. Kim, J., Kolmogorov, V., Zabih, R.: Visual correspondence using energy minimization and mutual information. In: Proc. of International Conference on Computer Vision (ICCV), (2), pp. 1033–1040 (2003)
12. Klaus, A., Sormann, M., Karner, K.: Segment-Based Stereo Matching Using Belief Propagation and a Self-Adapting Dissimilarity Measure. In: Proc. of International Conference on Pattern Recognition, (3), pp. 15–18 (2006)
13. Kolmogorov, V., Zabih, R.: Computing visual correspondence with occlusions using graph cuts. In: Proc. of International Conference on Computer Vision (ICCV), (2), pp. 508–515 (2001)
14. Loop, C., Zhang, Z.: Computing rectifying homographies for stereo vision. Technical Report MSR-TR-99-21, Microsoft Research (1999)
15. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision (IJCV)* 60(2), 91–110 (2004)
16. Matas, J., Chum, O.: Randomized Ransac with $T_{d,d}$ -test. *Image and Vision Computing* 22(10), 837–842 (2004)
17. Mayer, H., Ton, D.: 3D Least-Squares-Based Surface Reconstruction. In: Photogrammetric Image Analysis (PIA 2007), (3), Munich, Germany, pp. 69–74 (2007)
18. Morris, D., Kanade, T.: Image-Consistent Surface Triangulation. In: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), (1), Los Alamitos, pp. 332–338 (2000)
19. Nistér, D.: Automatic dense reconstruction from uncalibrated video sequences. PhD Thesis, Royal Institute of Technology KTH, Stockholm, Sweden (2001)
20. Scharstein, D., Szeliski, R.: Stereo matching with nonlinear diffusion. *International Journal of Computer Vision (IJCV)* 28(2), 155–174 (1998)

21. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision (IJCV)* 47(1), 7–42 (2002)
22. Stewart, C.V., Dyer, C.R.: The Trinocular General Support Algorithm: A Three-camera Stereo Algorithm For Overcoming Binocular Matching Errors. In: *Second International Conference on Computer Vision (ICCV)*, pp. 134–138 (1988)
23. Tian, Q., Huhns, M.N.: Algorithms for subpixel registration. In: *Graphical Models and Image Processing (CVGIP)*, vol. 35, pp. 220–233 (1986)