ISPRS Journal of Photogrammetry and Remote Sensing 91 (2014) 29–49

Contents lists available at ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: www.elsevier.com/locate/isprsjprs

On accurate dense stereo-matching using a local adaptive multi-cost approach

C. Stentoumis [a,*], L. Grammatikopoulos [b], I. Kalisperakis [b], G. Karras [a]

[a] Laboratory of Photogrammetry, Department of Surveying, National Technical University of Athens, GR-15780 Athens, Greece
[b] Laboratory of Photogrammetry, Department of Surveying, Technological Educational Institute of Athens, GR-12210 Athens, Greece

ARTICLE INFO

Article history:
Received 29 April 2013
Received in revised form 10 February 2014
Accepted 10 February 2014
Available online 6 March 2014

Keywords:
Photogrammetry
Matching
Stereo
Hierarchical
Cost
Reconstruction

ABSTRACT

Defining pixel correspondences among images is a fundamental process in fully automating image-based 3D reconstruction. In this contribution, we show that an adaptive local stereo-method of high computational efficiency may provide accurate 3D reconstructions under various scenarios, or even outperform global optimizations. We demonstrate that census matching cost on image gradients is more robust, and we exponentially combine it with the absolute difference in colour and in principal image derivatives. An aggregated cost volume is computed by linearly expanded cross skeleton support regions. A novel consideration is the smoothing of the cost volume via a modified 3D Gaussian kernel, which is geometrically constrained; this offers 3D support to cost computation in order to relax the inherent assumption of "fronto-parallelism" in local methods. The above steps are integrated into a hierarchical scheme, which exploits adaptive windows. Hence, failures around surface discontinuities, typical in hierarchical matching, are addressed. Extensive results are presented for datasets from popular benchmarks as well as for aerial and high-resolution close-range images.

1. Introduction

Generation of dense 3D information is a fundamental task in most applications in the fields of photogrammetry and computer vision (3D reconstruction, DSM production, object detection and recognition, automatic navigation, novel view synthesis, augmented reality). Methods for acquiring 3D information can be distinguished as *passive* and *active*. Image-based approaches (*passive*) are lately proven to be competitive to laser and optical scanners (*active*) in terms of accuracy, while exhibiting a clear advantage as regards cost and flexibility. Several theoretical alternatives exist for exploiting images in producing 3D information (shape from X). A core procedure is *image matching*, i.e. essentially the determination of correspondences among pixels. These approaches may be seen as consisting of two processes: establishment of *sparse* correspondences among images for camera calibration/orientation; and *dense* matching for 3D surface reconstruction. *Stereo-matching* algorithms for dense reconstruction, as that described in this paper, mostly exploit the *epipolar constraint*, hence they typically operate on rectified images to produce a *disparity map* (map ping of disparity values for every pixel of the reference image).

A significant number of efficient algorithms have been proposed for creating accurate disparity maps from single stereo-pairs. The effectiveness of such algorithms has been extensively discussed in several surveys (Dhond and Aggarwal, 1989; Banks and Corke, 2001; Scharstein and Szeliski, 2002; Brown et al., 2003). Scharstein and Szeliski (2002) have categorized algorithms by splitting them into four main components: *matching cost computation*, *support aggregation*, *disparity optimization* (*local* and *global*) and *disparity refinement*; publications addressing these components will be referred to below. Gong et al. (2007); Tombari et al. (2008) discuss the question of support region formation, while Hirschmüller and Scharstein (2009) evaluate the cost function itself under different optimization schemes. Enlightening comments are also found in Dhond and Aggarwal (1989); Brown et al. (2003). Wang et al. (2006); Nalpantidis et al. (2008) provide respective surveys focusing on criteria for hardware implementation and for real-time performance, while Tombari et al. (2010) have discussed the capabilities of fast stereo methods with low memory footprint. Evidently, difficulties exist in assessing stereo-methods due to diverging criteria set by different applications, as methods serving one purpose sometimes fail when the scenario changes.

* Corresponding author. Tel.: +30 2107722683; fax: +30 2107722677.
E-mail addresses: cstent@mail.ntua.gr (C. Stentoumis), lazaros@teiath.gr (L. Grammatikopoulos), ikal@teiath.gr (I. Kalisperakis), gkarras@central.ntua.gr (G. Karras).

http://dx.doi.org/10.1016/j.isprsjprs.2014.02.006
0924-2716/© 2014 International Society for Photogrammetry and Remote Sensing, Inc. (ISPRS) Published by Elsevier B.V. All rights reserved.

In more detail, in *matching cost computation* a dissimilarity measure is attributed to each pixel for every value in the disparity range. A wide spectrum of such matching measures has been proposed over the years. Most common among them are the absolute difference of pixel intensities, their squared difference, their normalized cross correlation, as well as measures relying on input images transformed by filters such as the median, the mean, the LoG, or more sophisticated tools like bilateral filtering (Tomasi and Manduchi, 1998). Non-parametric image transformations, such as *rank* and *census* (Zabih and Woodfill, 1994), produce robust results based on relationships of pixels with their neighbourhood. Birchfield and Tomasi (1998) have proposed a dissimilarity measure to cope with differences in image sampling. Recently, the *mutual information* approach has been proposed for effectively handling radiometric differences (Hirschmüller, 2008); on the other hand, pixel-wise descriptor measures, like DAISY (Tola et al., 2008) or SIFT variations (e.g. Strecha et al., 2011), have yielded promising results in global formulations for wide-base stereo.

Cost computed per pixel is supported by a neighbourhood around pixels in the *cost aggregation* step. There exist three main approaches through which this question may be addressed: use of support weights, support regions of arbitrary shapes and variations of rectangular windows. Methods based on *support weights* make use of a window fixed in size and shape, and adjust the weights attributed to each neighbouring pixel. Weights can be calculated according to colour similarity and geo metric proximity (Yoon and Kweon, 2006) or additional criteria (Xu et al., 2002). Support regions of *arbitrary shape* re present an attempt to establish an optimal window shape and size. Theory from the field of image filtering has contributed the idea of shape-adaptive windows based on separate circular sectors across multiple directions around a pixel (Foi et al., 2007; Lu et al., 2008). Cross-based windows have been proposed by Zhang et al. (2009). *Rectangular windows*, and their variations for improving efficiency, are the most obvious choice thanks to their simplicity of implementation. Shiftable windows or windows anchored at pixels other than the central one (Kang et al., 1995; Fusiello et al., 1997; Bobick and Intille, 1999), as well as multiple windows relying on local variation of intensity and disparity (Kanade and Okutomi, 1994; Veksler, 2003), have also been proposed.

Disparity estimation in *local* (region-based) methods is usually performed in the *winner-takes-all* (WTA) mode, i.e. the disparity with the lowest aggregated cost is chosen. *Global* methods, on the other hand, perform *disparity optimization* on an energy function defined over all image pixels by simultaneously imposing a smoothness constraint. Regarding the latter, various approaches have been implemented based on partial differential equations (Faugeras and Keriven, 1998; Strecha et al., 2004; Ranftl et al., 2012), dynamic programming (Veksler, 2005), simulated annealing (Barnard, 1986), belief propagation (Sun et al., 2003; Felzenszwalb and Huttenlocher, 2004) and graph-cuts (Kolmogorov and Zabih, 2001; Boykov et al., 2001).

The last step, *disparity refinement*, aims at elaborating the disparity map. This can include correcting inaccurate disparity values and handling occlusion areas (Bobick and Intille, 1999). A common approach is to enforce constraints which are not explicitly implemented in the disparity optimization phase (Marr and Poggio, 1976; Yuille and Poggio, 1984; Brown et al., 2003). Typically, a sub-pixel estimation step is taken to increase the resolution of the disparity map, as most algorithms search in a discrete disparity space. This includes adjusting a curve to pixel cost for each disparity (Hirschmüller, 2008) and sub-pixel interpolation to disparity values (e.g. Yang et al., 2009 use a mean filter). In the past, Tian and Huhns (1986) had proposed intensity interpolation, a differential method and phase correlation. For more details on the variety of stereo-matching algorithms one may refer to dedicated on-line evaluation platforms, such as the Middlebury evaluation platform (Section 8.1) and the KITTI benchmark site (Section 8.5), where developments and new trends in the field are being continuously reported.

In this publication we address a number of open questions which, in principle, regard the performance of both local and global stereo methods. The successful estimation of disparities around discontinuities, which denote surface boundaries in 3D space, and within poorly textured areas is always a challenge. Regarding the latter case, sparse matching methods tend to fill untextured areas through interpolation, but it is of course preferable to obtain actual disparities. Furthermore, typical deficiencies to be dealt with are the "fronto-parallel effect" (flat surfaces in 3D space parallel to the image plane are favoured) in highly inclined surfaces and the quantization of disparities in discrete methods (e.g. MRF models, most local and semi-global approaches). At the same time, we wish to retain the computational efficiency required for real-time (and lately on-line) processes or high-resolution images. Our effort also aims at defining parameter-stable models, which is not a trivial task; most global methods involve handling a variety of parameters (i.e. the smoothness term is highly dependent on parameters defining the function, and hence on the image scenario), but also many local methods are sensitive to their empirically determined parameters.

Local methods are, typically, considered to be more straightforward and simple, and hence adequately fast for real-time applications, but of lower accuracy. On the other hand, methods based on a global optimization framework use elaborate models to describe the matching process; this often results in disparity maps of high quality, but at the cost of a computational load which may be restrictive for real-time tasks or large data manipulation. Notwithstanding this general remark, research work demonstrated in major evaluation platforms indicates that the limitations of each category are partially losing in significance: theoretical improvements and hardware upgrade make fast global implementations possible; at the same time, local methods achieve disparity maps of high accuracy, superior to several global methods, mainly via elaborate cost aggregation. Obviously, the increase of computational resources in personal computers, and even small dedicated processors, have made the exploitation of highly elaborate and demanding algorithms feasible, but at the same time image acquisition hardware keeps evolving and offering images with high spatial and radiometric resolution. Thus, algorithms offering lower complexity are still required in certain applications, while local approaches will probably continue to be easier to implement regardless of hardware.

Furthermore, manipulating large data, real-time and recent on-line applications, all require speed. This objective is partially served through hardware implementation. A wide range of such implementations can be found in surveys mentioned later in the text. Local methods have an inherent ability to bundle with commercial (e.g. in FPGAs) and special purpose hardware, as well as with their existing supportive software, e.g. CUDA for Nvidia GPUs. On the other hand, stereo matching methods based on global optimization algorithms are in general difficult to implement on hardware. Their complex and usually iterative nature makes defining and running of parallel processes unappealing, i.e. parallelism is difficult in MRF because of variable connectivity. This, of course, is not to suggest that fast and accurate global algorithms do not exist, but constructing them needs more time and effort.

Bearing in mind the above considerations, this work presents a hierarchical matching scheme based on local patch-based matching in a way that the previously discussed requirements are met, while keeping the complexity of the algorithm substantially low. We have chosen to extend the cross-based support regions aggregation method, because this allows forming highly adaptive irregular shapes; it is independent of the cost function; and it degenerates from area to single dimensional aggregation in order to maximize computational efficiency. The presented matching framework not only scores high at the Middlebury tests, but also performs satisfactorily for high resolution images and outdoor scenes. It is evaluated with applications which demand high accuracy, e.g. ortho-maps from aerial imagery and small objects of archaeological interest with rich detail.

Thus, this contribution presents a stereo-matching algorithm which is based on the combination of existing state-of-the-art techniques and novel considerations. Novel aspects include the use of a linear threshold in the cross-window formulation; the use of census transformation on image principal gradients, which appears to yield enhanced sub-pixel accuracy; and the combination of multiple matching measures in the final cost. We also present a geometrically constrained smoothing process for the cost volume to provide 3D support, in order to improve the 2D aggregated cost as well as weaken the "fronto-parallel effect" and the effect of disparity quantization in scenes acquired with strong inclination. The above successive steps are integrated into an intuitive hierarchical scheme which responds to known problems around discontinuities. Finally, a robust refinement procedure for the disparity map is proposed as a combination of already known actions. Results for a wide range of imaged scenes are shown to evaluate the competitive performance of the algorithm in major bench marking platforms and high-resolution real-life scenes. It is noted that the reported matching scheme has produced robust results without special tuning for each particular dataset, which is rather uncommon in matching methods. We also wish to demonstrate here that a local method can offer quality reconstructions for high resolution images, whereas the common idea is to select local methods simply for speed. Certain aspects of this contribution have been presented in previous publications (Stentoumis et al., 2012, 2013).

Section 2 describes the proposed matching measures and their integration in a robust cost function. Section 3 refers to the formation of the adaptive support regions, which are based on combined cross-skeletons of the reference and matching images. After Section 4, which briefly discusses the disparity image representation, in Section 5 a three-dimensional smoothing process for the cost function is introduced, based on geometric constraints, to establish 3D local support for non-frontoparallel surfaces. The aforementioned steps are integrated into a hierarchical scheme (Section 6), and the estimated disparity map is refined through a series of post-processing steps (Section 7). Section 8 reports results from various scenes, including images from major bench marking platforms as well as aerial and close range high resolution images, to evaluate the performance of the algorithm. Finally, conclusions and a discussion follow in Section 9.

2. Matching/cost functions

2.1. Census on intensity principal derivatives

Census (T_c) is a widely used non-parametric image transformation (Zabih and Woodfill, 1994). For a support neighbourhood $N(m \times n)$ of pixel $\mathbf{p}$, a map of neighbouring pixels with intensities less than that of $\mathbf{p}$ is formed. As a result, a binary vector T_c of length $m \times n$ is assigned to each pixel. In case $m \times n < 255$, a 255 bit string can store the descriptive vector of each pixel. Census transformation T_c depends on how a pixel relates to its surroundings within the image patch. It is hence robust against changes in brightness/contrast which do not modify the ordering of intensity values. Moreover, in this binary approach the actual values of individual pixel intensities do not affect the overall measure, but only a specific bit of the binary descriptor of $\mathbf{p}$. This makes T_c robust against individual outliers around discontinuities and in cases of noisy pixels.

Unlike usual approaches, the transformation is performed here not on grey-scale image intensity function I but on its principal derivatives $\partial I/\partial x$, $\partial I/\partial y$. Image derivatives are related to characteristic structural image features (points, edges) and are, of course, widely used as a contributing source of information in matching, e.g. in gradient-based methods in global optimization formulations, feature-based methods and local stereo (Scharstein, 1994; Brown et al., 2003; Klaus et al., 2006). Derivatives are helpful in treating constant bias in pixel values. The present approach provides an extended binary vector, whereby if $\mathbf{q}$ is a neighbour of $\mathbf{p}$:

$$T_c(\mathbf{p}) = \underset{\mathbf{p}\in\left\{\frac{\partial I}{\partial x},\frac{\partial I}{\partial y}\right\}}{\otimes}\ \underset{\mathbf{q}\in N_{\mathbf{p}}}{\otimes} c(\mathbf{p},\mathbf{q}),\ \ c(\mathbf{p},\mathbf{q}) = \left\{\begin{matrix} 0, \partial I(\mathbf{p}) \leqslant \partial I(\mathbf{q}) \\ 1, \partial I(\mathbf{p}) > \partial I(\mathbf{q}) \end{matrix}\right\} \tag{1}$$

which strengthens the positive aspects of the original transformation. In Eq. (1) $\otimes$ denotes the act of concatenation, following the original definition of T_c, and $\partial I(\mathbf{p})$ represents the image gradient in the x or y directions. The direct introduction of the gradients in two image directions into the binary vector doubles the size of the produced vector T_c, thus exploiting the representational potential of image gradients. Finally, the matching cost C_{census} between a pixel $\mathbf{p}$ of the reference image and its corresponding pixel $\mathbf{p}'$ in the matching image is calculated as the *Hamming distance*, which represents the number of unequal elements in the two binary vectors. This is applied for all potential disparity values $d_{\mathbf{p}}$ ($\mathbf{p}' = \mathbf{p} + (d_{\mathbf{p}}, 0)^T$).

The performance of the proposed census matching function on the *Tsukuba* stereo-pair of the Middlebury evaluation platform (http://vision.middlebury.edu/stereo/) is presented in Fig. 1 (left), which also shows on the right the improvement of the disparity map for the *Teddy* pair. For comparison, in Fig. 1 the disparity maps derived by matching with the original census transformation are also seen.

Extensive tests were performed in order to evaluate the effect of applying census on gradients compared to conventional census cost on intensity, without using the rest of the matching steps. Table 1 presents the improvement (+) or deterioration (of disparity maps in various available datasets, namely the typical 4-pair dataset and the new 2006 dataset (Hirschmüller and Scharstein, 2007) of Middlebury, and the KITTI dataset (Section 8.5)) for non-occluded pixels (*nonocc*) and the whole image (*all*) after the aggregation step (see Section 3).

Although the large sub-pixel improvement in the typical Middlebury 4 pair dataset can be attributed to the artificial Tsukuba stereo pair, there is a certain improvement in most pairs when using census on gradients. A more thorough inspection of the obtained disparity maps has revealed that results deteriorate in texture-less regions (e.g. white wall and shiny plastic surface in the *Mid* and *Plastic* stereo-pairs, respectively), while they significantly improve in highly textured areas (e.g. periodic table [*Bowling* and *Teddy* pairs], textile [*Cloth*] and map [*Baby*] background). For the purposes of stereo matching, census transformation based on image gradients appears to be less sensitive to radio metric differences and repetitive patterns, while the discriminative capability of the binary descriptor increases, leading to results of higher accuracy. The above results should be further investigated to be theoretically illuminated. The recent work of Hafner et al. (2013) interprets the functionality of census transformation by transferring the discrete binary vector to continuous space; census efficiency is generally superior when compared, using a variational scheme, against gradient constancy assumption for the purposes of optical flow. Such a reinterpretation of a binary descriptor could be used in our case. Finally, Vogel et al. (2013) offer more insights

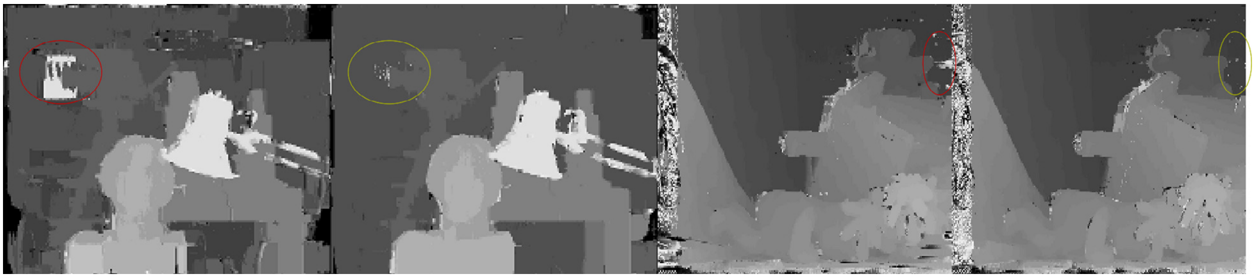


Fig. 1. Examples of disparity maps for the *Tsukuba* and *Teddy* stereo pairs obtained via the default census transformation (first and third image) and census on gradients (second and fourth image) after the aggregation step. Indicated are examples for areas of improvement.

Table 1
Percentage of change in disparity map using census on gradient.

	1 px Threshold		0.75 px Threshold		3 px Threshold	
	Nonocc	All	Nonocc	All	Nonocc	All
Middlebury 4 pairs	+1.32	+1.24	+6.43	+6.27		
Middlebury 2006	+0.63	+0.90	+0.78	+1.05		
KITTI					0.00	−0.01

regarding the relation between census, centralized absolute differences and gradient constancy under a variational scheme.

2.2. Absolute difference on image colour

The absolute difference on colour channels (ADc), or on intensity, is a simple and easily implementable measure, widely used in matching (L_1 correlation). Although sensitive to radiometric differences, it has been proven as an effective measure when combined with flexible aggregation areas and involving combination of co lour layers. The cost term C_{ADc} is defined as the average AD value of all three channels:

$$C_{\mathrm{ADc}}(\mathbf{p},d) = \frac{1}{3}\sum_{i}\left|I_i^{ref}(\mathbf{p}) - I_i^{mat}\left(\mathbf{p}+(d,0)^T\right)\right|, \forall i \in \{r,g,b\} \tag{2}$$

This turns out to improve results compared to matching on separate channels or grey-scale (a partial form of Eq. (2) is used when only grey-scale images exist). Truncation is common for eliminating spikes from the cost function. As defined here, the measure includes no truncation threshold on the colour difference range, since such a precaution is incorporated in the cost fusion (Section 2.4).

2.3. Absolute difference on image principal gradients

Here, the derivatives of image intensity in the two principal directions are extracted, and the sum of absolute differences of each derivative value in the x and y directions is used as a cost measure. The use of directional derivatives separately, i.e. before summing them up to the single measure ADg (Eq. (3)), introduces the directional information for each derivative into the cost measure:

$$C_{\mathrm{ADg}}(\mathbf{p},d) = \sum_{x,y}\left|\nabla I^{ref}(\mathbf{p}) - \nabla I^{mat}\left(\mathbf{p}+(d,0)^T\right)\right| \tag{3}$$

A mild Gaussian filter (size 3 × 3, σ = 0.5) is applied on the grey-scale images before calculating partial derivatives for reducing noise and for smoothing around image edges.

2.4. Total matching cost

The final matching cost C is derived by merging the three different costs: census transformation on image gradients (expressed through the Hamming distance), absolute difference in colour (or intensity) values and absolute difference on principal image gradients. A robust exponential function (Yoon and Kweon, 2006; Mei et al., 2011), which resembles a Laplacian kernel, has been preferred for cost combination:

$$C(\mathbf{p},d) = 1 - \exp\left(-\frac{C_{\mathrm{census}}(\mathbf{p},d)}{\lambda_c}\right) + 1 - \exp\left(-\frac{C_{\mathrm{ADc}}(\mathbf{p},d)}{\lambda_{\mathrm{ADc}}}\right) + 1 - \exp\left(-\frac{C_{\mathrm{ADg}}(\mathbf{p},d)}{\lambda_{\mathrm{ADg}}}\right) \tag{4}$$

When compared to a linear combination of individual cost measures, function C has the advantage of truncating costs. By smoothing the original cost functions, this truncation prevents the final cost from being contaminated by large penalties due to outlying individual cost values. This function takes values in the field $[0,1)$ for $C \geqslant 0$. The values of the three costs C_{census}, C_{ADc} and C_{ADg} are thus scaled in the same value field. The values of each cost should be normalized by λ to ensure equal contribution to the final cost, or tuned differently to accordingly adjust their impact on cost. Tests performed on the Middlebury dataset for stereo-matching are presented in Figs. 2 and 3.

3. Cost aggregation

Local approaches of stereo-matching are based on the definition of pixel neighbourhood. It is assumed that pixels within this neighbourhood share the same disparity; fronto-parallel surfaces are thus favoured. Foundation of adaptive approaches is the fact that pixels within a support region ought to have similar colours and are expected to decrease in coherence with their distance from the reference pixel in image space.

3.1. Support region formation

Here, a modification of the cross-based support region approach is used (Stentoumis et al., 2012). The construction of such cross-based support regions is achieved by expanding around each pixel

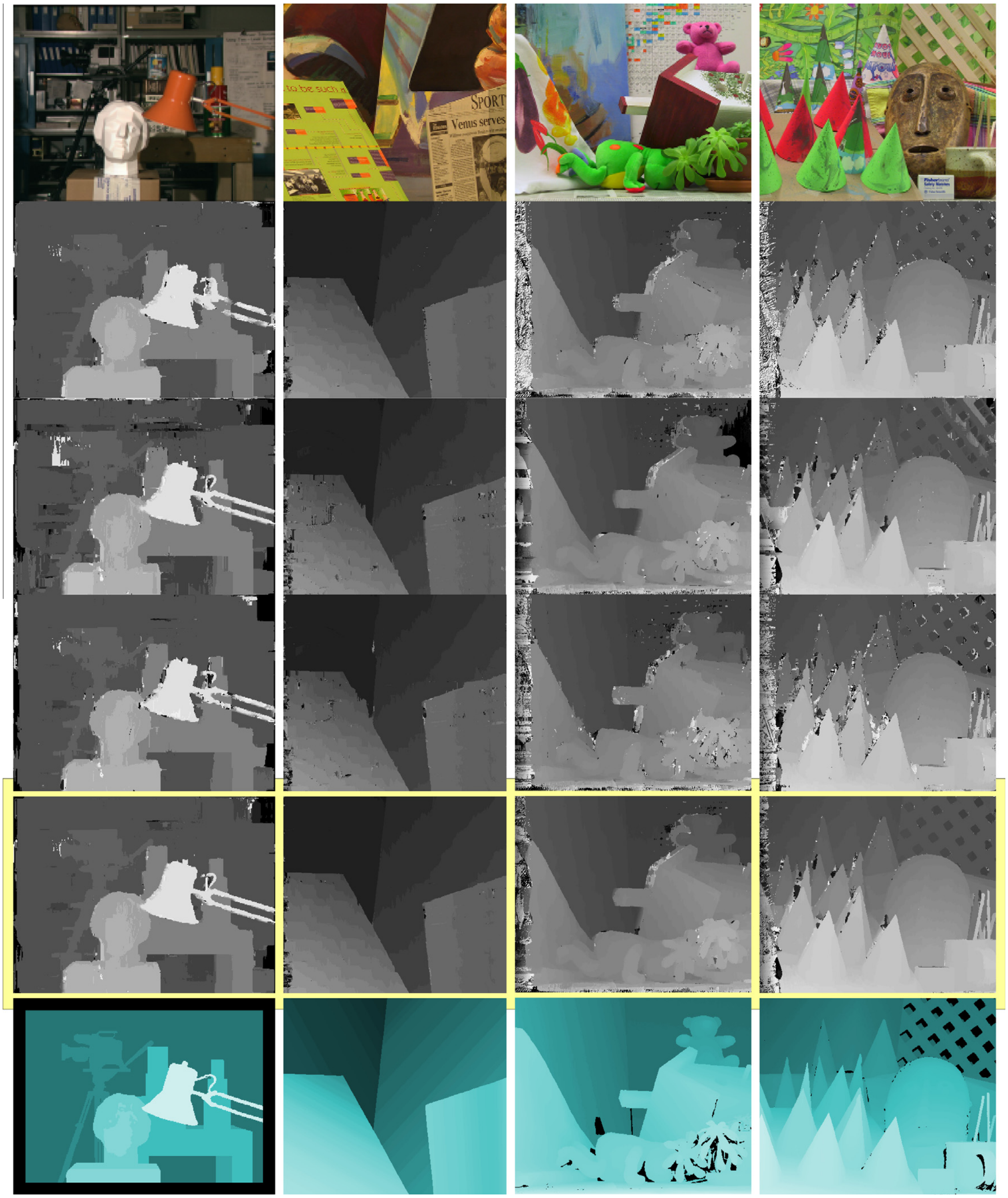


Fig. 2. Comparison of different matching cost functions for the four Middlebury stereo pairs (*Tsukuba*, *Venus*, *Teddy*, *Cones*). Disparity maps are presented for the individual and the overall matching functions after the aggregation step. From top to bottom: left image, proposed modified census transformation, AD in image channels, AD in principal images gradients, combined cost. The lowest row presents the reference disparity maps. The refinement steps described at Section 7 have not been used here, in order to illustrate individual results and the improvement achieved by fusing the three costs.

p a cross-shaped skeleton to create four segments (sk^{+}_{Hz}, sk^{-}_{Hz}, sk^{+}_{V}, sk^{-}_{V}) defining the corresponding sets of pixels $H(\mathbf{p})$ and $V(\mathbf{p})$ in the horizontal and vertical directions, as seen in Fig. 4 (Zhang et al., 2009; Mei et al., 2011).

Mei et al. (2011) have proposed two thresholds for colour similarity and two further thresholds for spatial closeness. In our approach, a linear threshold is imposed on window expansion:

$$\tau(l_{\mathbf{q}}) = -\frac{\tau_{\max}}{L_{\max}} \times l_{\mathbf{q}} + \tau_{\max} \tag{5}$$

This linear threshold in colour similarity involves the maximum semi-dimension $L_{\max}$ of the window size, the maximum colour dissimilarity $\tau_{\max}$ between pixels **p** and **q**, and the spatial closeness $l_{\mathbf{q}}$ (Fig. 5). This, next to producing somewhat better results for the Middlebury datasets, renders two of the manually given input variables redundant; at the same time, thresholding of colour difference τ according to spatial closeness $l_{\mathbf{q}}$ from the skeleton becomes smoother. The difference τ between successive pixels is also checked after Mei et al. (2011). Typical support regions generated according to the above considerations are presented in Fig. 6.

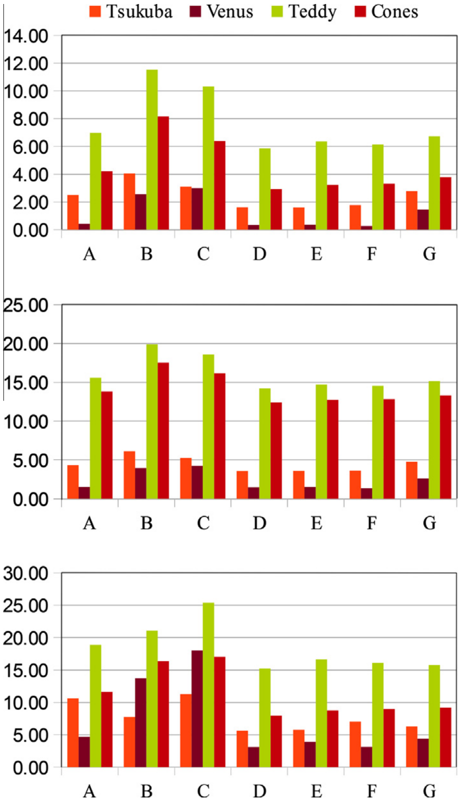


Fig. 3. Visualised performance of cost functions (in % of erroneous disparities) by comparing different cost combinations against true image disparities. The charts correspond, from top to bottom, to comparisons against *non occluded* pixels, all image pixels and areas *near discontinuities*. A: extended census, B: AD on colour, C: AD on gradients, D: exponential combination of previous costs, E: extended census plus AD on colour, F: extended census plus AD on gradient, and G: AD on colour plus AD on gradient.

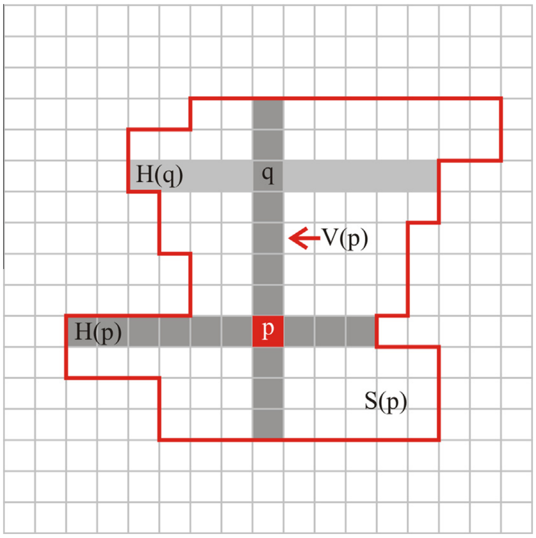


Fig. 4. Expansion of the cross-based support region $S(\mathbf{p})$ driven by the skeleton of each pixel. The skeleton pixels for $V(\mathbf{p})$ and $H(\mathbf{p})$ sets are calculated only once per pixel. When pixel $\mathbf{q}$ belongs to $V(\mathbf{p})$, the corresponding horizontal arm $H(\mathbf{q})$ is added to $S(\mathbf{p})$. $S(\mathbf{p})$ consists of the union of $H(\mathbf{q})$ for all pixels $\mathbf{q}$ which participate in $V(\mathbf{p})$. [After Zhang et al. (2009)].

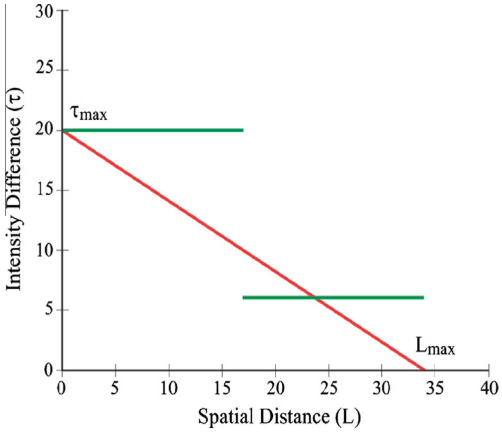


Fig. 5. Threshold $\tau(l_\mathbf{q})$ imposed on colour difference between pixels $\mathbf{p}$ and $\mathbf{q}$ is linearly reduced (red curve) as $\mathbf{q}$ approaches the limit of maximum window size. The green lines show the form of the two thresholds originally proposed by Mei et al. (2011) for handling extended texture-less image areas. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

The determined cross-window can be based either solely on the left (reference) image or also generated on the right (matching) image as well, hence S depends also on d. This second case of *combined* windows involves the intersection of the two cross windows ($S(\mathbf{p},d) = S^{ref}(x,y) \cap S^{mat}(x+d,y)$. Results from combined windows are expected to be more robust, since the projective distortions and radiometric differences between patches on the reference and matching images are taken into account.

Generally, a [3 × 3] median filter is applied for cross-skeleton determination. Moreover, the minimum length of all cross-segments is 1 pixel to ensure a minimum support region S of 9 pixels.

3.2. Aggregation

The cost aggregation step of the algorithm is computationally expensive, since support regions are variable for each pixel, but it involves repetitions of summations. When the support neighbourhood has a rectangular shape of constant size, the aggregation of pixel-wise costs can be efficiently performed by convolving through filters. In cases of variable support region size, *integral* images (Viola and Jones, 2001), or *summed area tables*, can be exploited for speeding up the cost aggregation process (in fact, these had been originally proposed for efficient computations on texture mapping by Crow, 1984). Their use in cost aggregation can drastically reduce computational load, because summations involving matching cost for a pixel need to be performed only once (Veksler, 2003; Zhang et al., 2009).

Aggregated pixel costs C_{aggr} are normalized by the number of pixels in the support region to ensure that costs per pixel have the same scale:

$$C(\mathbf{p},d) = \frac{C_{aggr}(\mathbf{p},d)}{\|S(\mathbf{p},d)\|} \tag{6}$$

4. Representation

4.1. Disparity space images

The total cost function $C(x,y,d)$ produces values for each pixel $(x,y)^T$ per each potential disparity value d. An effective way of representing the field of values of function C is a cost volume defined in the three dimensions x, y, d (cf. Fig. 12). This representation of the stereo-matching function is referred to as *disparity space image* or DSI (Yang et al., 1993; Bobick and Intille, 1999; Brown et al., 2003). Slices parallel to the x–y and x–d planes offer a thorough

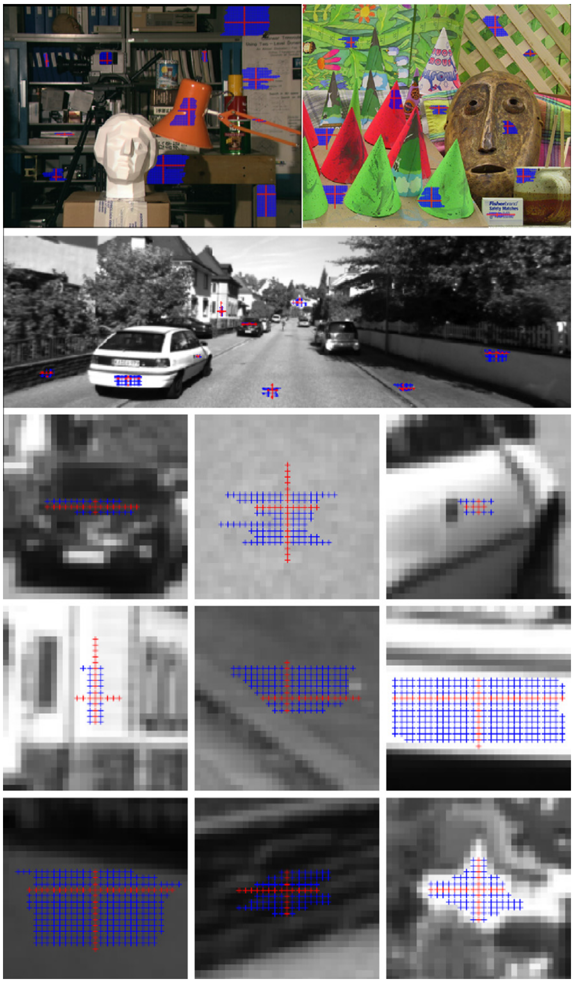


Fig. 6. Examples of the support regions formed with the linear approach for the generation of cross-based windows.

understanding of the cost function performance. An alternative method for representing stereo-matching cost function through a DSI is via a graph, in which the axes present the corresponding cost values for pairs of homologue epipolar lines (left/right scan-lines). This representation is exploited when optimizing through dynamic programming (Ohta and Kanade, 1985; Cox et al., 1996). DSI is a convenient representation, alternative to the volumetric object coordinates representation (x,y,Z_{space}) which is common in the field of multi-view matching (Moravec, 1996; Collins, 1996).

In Fig. 7 x–d DSI slices (left) are presented at three scan-lines of the Middlebury *Teddy* pair. On each such diagram per epipolar line a minimum path $d(x,y)$ in cost volume is defined; these cost paths form the disparity map. Some cases worth noting are marked in Fig. 7 on the original base image and the respective minimum cost positions on the x–d DSI slices. Pixels in positions **a**, **e**, **b** belong to disparity map edges and indicate break-lines on the physical object (abrupt increase in depth). Such a decrease in the disparity value relates to an occlusion on the left (base) image. On the other hand, gaps in DSI slices, like those at segments (**fg**) and (**cd**) in Fig. 7, indicate occlusion areas on the right (match) image. These appear on DSI slices as an increase of disparity value followed by a diagonal shift of the minimal path, which defines a poor minimum cost solution. Repetitive patterns on the two images will also be visible on the DSI slice as multiple local minima on the cost function (as in segment **bc**). The two upper images on the right in Fig. 7 are x–y DSI slices of the cost volume at two disparity values d. The lower image on the right is the final WTA selection of smallest costs at each pixel position.

5. Geometrically constrained smoothing of cost volume

Aggregation of cost has the inherent limitation of assuming that all pixels in a neighbourhood share the same depth, an assumption favouring fronto-parallel surfaces. Its consequences are clearer when actually reconstructing a 3D scene from two images than when simply assessing the quality of a disparity map. The assumption of fronto-parallel planar surfaces is also present in the smoothness constraint in several matching functions optimized globally (Terzopoulos, 1986; Strecha, 2007), since cost aggregation for inclined surfaces is not a common approach. Relevant research seeking 3D support in local stereo uses: a surface continuity prior by assuming that disparity differences in a neighbourhood follow the normal distribution (Prazdny, 1985); limitations in disparity differences (Pollard et al., 1985); a statistical model for adaptive windows which incorporates fluctuations of disparity and intensity (Kanade and Okutomi, 1994); 3D cost aggregation with explicit occlusion detection (Zitnick and Kanade, 2000). Furthermore, Ogale and Aloimonos (2004) handle surfaces horizontally slanted with respect to the stereo base; Zhang et al. (2008) have proposed weighted aggregation in 3D disparity space based on initial disparity gradients; Bleyer et al. (2011) project support regions on estimated segmented planes; and Antunes and Barreto (2013) have imposed slanted surfaces on histogram aggregation hypothesis.

Here, we propose the weighted aggregation of 2D aggregated costs $C_0(x,y,d)$ belonging to geometrically possible disparities around a pixel through the convolution of cost volume with a 3D Gaussian filter:

$$f(\xi) = \frac{1}{(2\pi)^{3/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\xi - \bar{\xi})^T \Sigma^{-1} (\xi - \bar{\xi})\right) \tag{7}$$

Smoothing cost values with a 3D filter to attribute weights to all cost elements may be regarded as aggregating costs $C_0(x,y,d)$ for all possible neighbouring pixels $\mathbf{q}(x,y)$ and all possible disparities. This is equivalent to using a 3D support region

$$C(x,y,d) = k * C_0(x,y,d) \tag{8}$$

The partial Gaussian kernel k (Fig. 8) is adapted in order to serve the *ordering* (Yuille and Poggio, 1984) and *uniqueness constraints* (Marr and Poggio, 1976). This kernel has the properties of attributing weights to neighbouring costs inversely proportional to their spatial distance in the DSI. This approach of 3D local support exploits the attributes of DSI representation and has the advantage of avoiding the need for explicit identification of slanted surfaces in 3D world space.

In Fig. 9 (top) the disparity function with respect to pixel position x on a scan-line is presented. The respective scan-line on the right image is seen below. Identical shades of grey represent pixel correspondences between the two diagrams. The 7th pixel (**p**) is the one under inspection. This x–d DSI slice is divided in four areas, based on the possibility that the neighbours of **p** have these specific disparities, and are defined by the two dashed lines. Positions on the 45° diagonal all correspond to the 3rd pixel of the epipolar line on the right image, hence this is the maximum increase in disparity ($d = x^{ref}$–x^{mat}). Since only one-to-one correspondences are allowed, these positions should not contribute to the diffused cost of **p**. Positions above and below **p** (vertical dashed line) are also invalid, because they correspond to multiple disparity values for **p**. The two described directions (dashed lines) on the graph define four areas, only two of which contribute to the cost diffusion (hatched areas). Positions included in the two non-hatched triangles formed by the dashed lines in the graph violate the *ordering constraint*; a

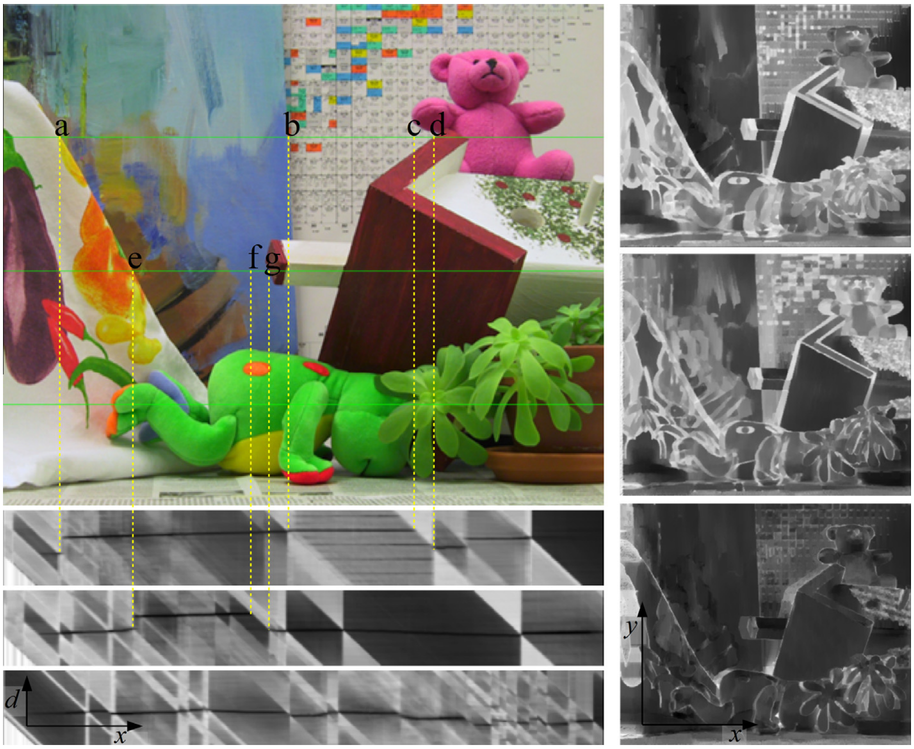


Fig. 7. Disparity space image (DSI) representation of the cost function for the Teddy pair. Left: *x*–*d* DSI slices at epipolar lines *y* = (100, 200, 300). Right: *x*–*y* DSI slices at *d* = (20, 40) and the WTA solution costs. Darker areas depict low cost matching values; white areas show locations of high cost.

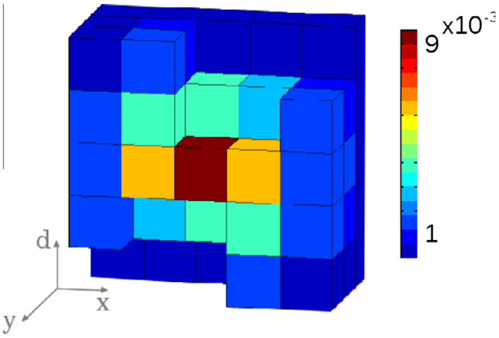


Fig. 8. 3D intersection at the *x*–*d* plane of a Gaussian filter restricted by occlusion borders (half of the 5 × 5 × 5 kernel *k* is displayed here).

larger decrease in *d* would result in an inversion in the sequence of pixels.

The standard deviation of the Gaussian function is set in accordance to the *full width at half maximum* parameter:

$$\sigma = \frac{l_k}{2} \cdot \frac{1}{2\sqrt{2\ln(2)}} \tag{9}$$

with l_k being the length of the kernel. Thus, σ in each filter direction does not need to be set manually, and filter shape is independent of its size. The full width at half maximum parameter is the denominator in Eq. (9). The 3D smoothing process described here can also be seen as a *local diffusion* process, based on 3D disparity space and imposed as a smoothness constraint. Usual diffusion processes are iteratively applied separately at each DSI level, i.e. in 2D. Although iterative diffusion is usually part of global approaches, local iterative diffusion has also been proposed (Scharstein and Szeliski, 1998). This smoothness constraint is applied here on costs already aggregated in 2D; hence its impact is restricted by the outcome of the previous step of the algorithm. Yet, disparity maps and resulting reconstructions on challenging datasets indicate that the result is more robust. Results for the disparity map for pair #6 of the KITTI benchmark site (Section 8.5) are seen in Fig. 10. In a scene with many inclined surfaces, where local methods often fail, the pro-

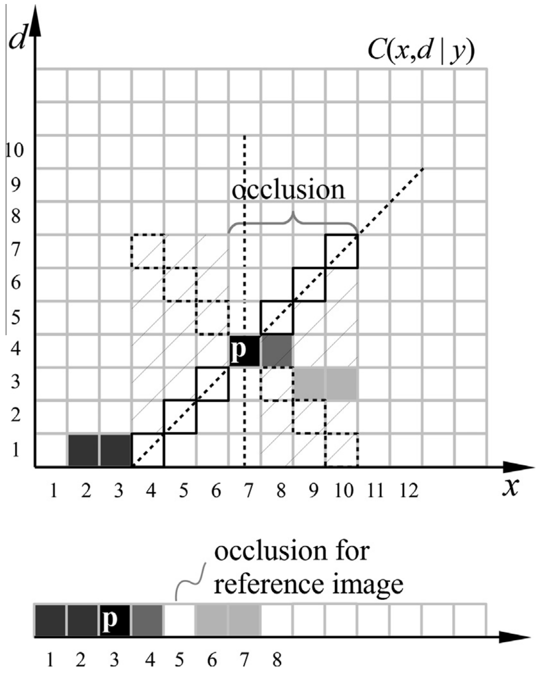


Fig. 9. Above: positions excluded by the ordering and *uniqueness constraints* in an *x*–*d* slice for the cost of reference image. Below: correspondences in the matching image; identical grey shades indicate pixel correspondences between images.

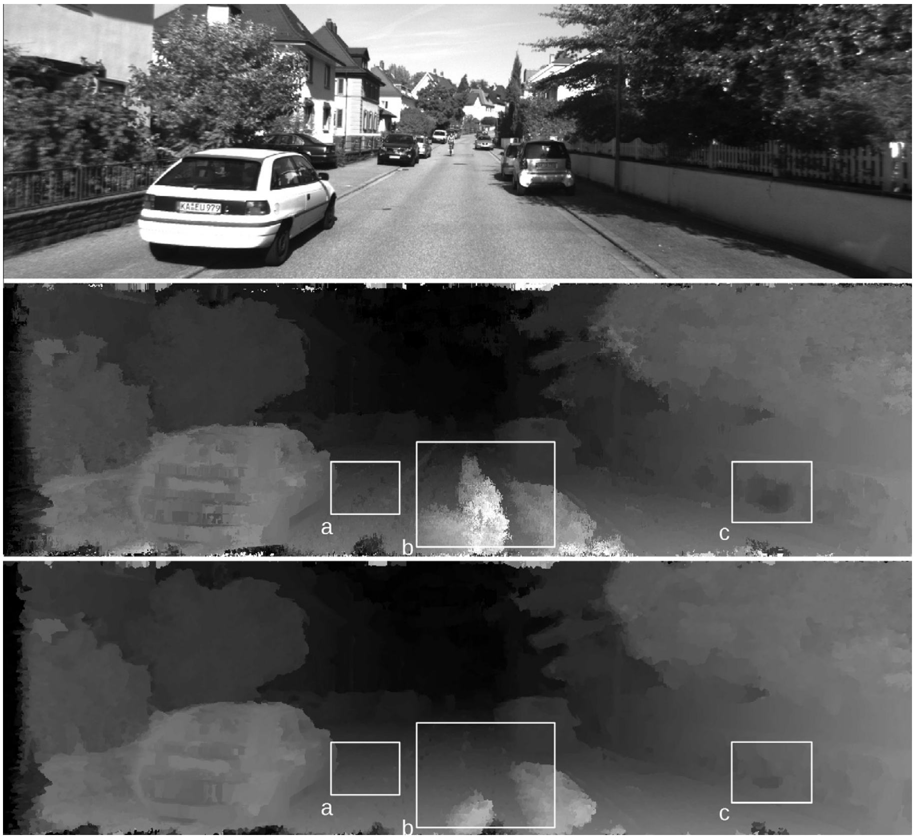


Fig. 10. Improvement of the disparity map through geometrically constrained smoothing of the cost function showing the estimated disparity map without (middle) and with (bottom) cost smoothing. Windows *a*, *b* and *c* indicate areas of improvement.

posed geometrically constrained smoothing of the cost function yields an improvement of ~10%. Indicated rectangles *a*, *b*, *c* include areas where surfaces of high inclination are correctly constrained. At *a* and *b*, in particular, the surface direction is almost perpendicular to the camera plane, thus severely violating the assumption of fronto-parallelism made in the aggregation step. The average improvement across the complete KITTI dataset is ~5%, and the improvement through constrained smoothing compared to typical 3D Gaussian smoothing is ~1.5%.

The estimation of disparity is carried out in the WTA mode, as in most local and semi-global approaches. The disparity label with the lowest cost is selected, i.e. $d_{\text{WTA}} = \text{argmin}_d(C(\mathbf{p}, d))$.

6. Hierarchical approach

An increasing variety of high resolution imaging systems is now available both to public and professionals in several areas of interest. Hence, algorithms capable of handling large amounts of data are needed. An extension of the method reported in the preceding sections to high resolution images is necessarily based on scaled representations of the stereo-pair. The aim is to limit the disparity search space to a computationally manageable range, and also guide matched disparities in a coarse-to-fine context through scale-space. This approach also reveals structures in different layers of image pyramids, which lead from a rough, yet close to reality, 3D surface to finer detail as one proceeds through the image pyramid. Matching is steered through this "flow of information", and the possibility of getting stuck to local minima is narrowed down (Moravec, 1980; Quam, 1986). Gaussian pyramids are employed here with a 3 × 3 ($\sigma = 0.5$) filter for all scales and subsequent down-sampling by a factor of 2.

The disparity map is expanded to the next finer level by propagating disparities via bilinear interpolation and smoothing by Gauss filtering for removing spikes. Also, gaps in the intermediate disparity maps due to outliers detected during image match consistency check (Section 7.1) are filled by interpolation to neighbouring valid pixels. The initial disparity map is regarded as a "zero map", in the sense that the search space is bounded between zero and the width of the lowest pyramid layer, which means that no initial rough object model (e.g. derived from SIFT key points) is needed.

While the exploitation of a multi-resolution scheme produces a faster algorithm and a more firmly bounded matching procedure, errors originating from remaining local minima in the initialization step, or ill-defined/inadequate disparity search space, can still be transferred across the pyramid. Outliers due to mismatches may well find their way up to the final result, especially since cost measure is not thresholded (as opposed, for example, to correlation techniques). These mismatches usually appear as local extrema in the disparity function; thus, low-pass filtering is needed. Nevertheless, this approach often fails due to the small filter size, required for avoiding excessive smoothing, particularly in case of discontinuities (and also of large repetitive patterns or untextured areas). Discontinuities are in fact of special interest, as matching based on hierarchical representation often fails in determining correct boundaries. This is partially due to the accumulation of erroneous disparities around object borders, which occur either by the upscaling of disparity maps or the reprojection of erroneous pixels in coarser scales in object space. At this point, therefore, the cross windows mentioned in Section 3.1 are introduced, extended by a number of pixels (here 2 pixels), to restrict the field of values of the disparity function per pixel.

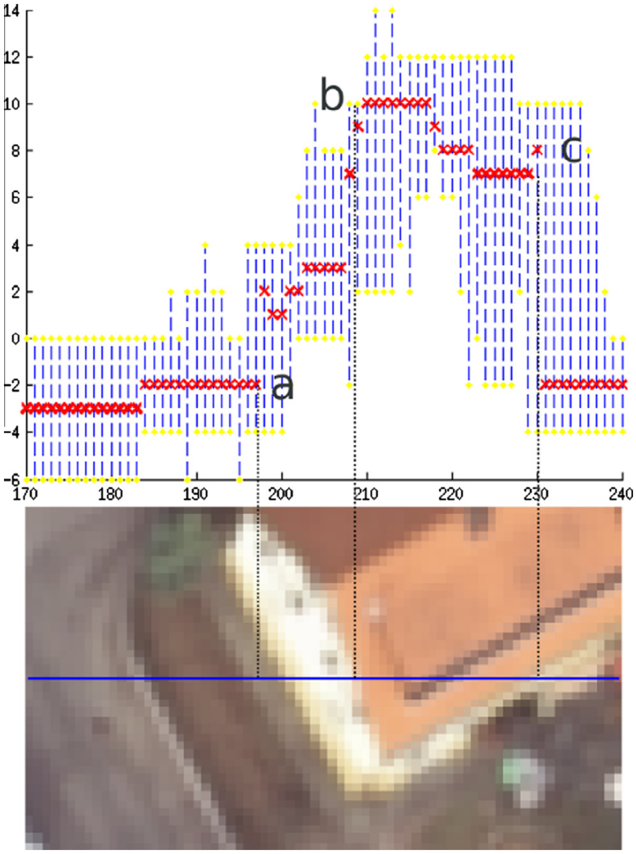


Fig. 11. The field of values for pixel disparities is displayed. The diagram contains the disparity values against x positions. The disparities of the minimum cost solution for each pixel along the scan-line are marked with "x"; the dashed lines mark the range of potential disparities. The line below presents the epipolar line in question. Each pixel has a different disparity range defined by the scale of the image and its support region. The interval between pixels a and b is an occlusion area; pixel c is on a disparity discontinuity. The disparity range is significantly widened near edges, whereas it is narrowed in areas of small slope of the disparity function.

In particular, it is accepted that pixels $\mathbf{q}$, which belong to the support region $S(\mathbf{p},d)$ of $\mathbf{p}$ (Section 3.1), can adequately define the disparity $d^s_\mathbf{p}$ range for $\mathbf{p}$ via their approximate disparity $d^{s-1}_\mathbf{p}$, which has been computed in the coarser layer. Thus, if s denotes the current pyramid layer and $s1$ the coarser layer, the disparity range is:

$$\min_{\mathbf{q}\in S(\mathbf{p},d)} d^{s-1}_\mathbf{q} \leqslant d^s_\mathbf{p} \leqslant \max_{\mathbf{q}\in S(\mathbf{p},d)} d^{s-1}_\mathbf{q} \tag{10}$$

In this way mismatches near edges, where "jumps" in the disparity map occur, are dealt with. The search interval is sufficiently wide to overcome misplacement of edges in the disparity map of lower resolution layers, but also remains restricted within a support region (Fig. 11), which is adapted not only to image space but also in scale. This is an important aspect, since one thus avoids explicit treatment of discontinuities (Sizintsev and Wildes, 2010; Sun et al., 2011) or the use of more elaborate approaches (e.g. through edge-preserving filters) for creating image pyramids for matching.

In Fig. 12 the refinement of the initial cost volume is presented. At each scale, the range of disparity $d^s_\mathbf{p}$ of position $\mathbf{p}$ is restricted by Eq. (10). The lowest scale representation of stereo-pairs result in a compact cost volume since all potential disparity labels participate in the solution. Movement towards larger scales produces an "eroded" cost volume, as the range of $d^s_\mathbf{p}$ is being gradually narrowed. Slices x–d defined per epipolar line y depict the matching function for the specific line. The minimum solution of this function is distinguished as blue in Fig. 12.

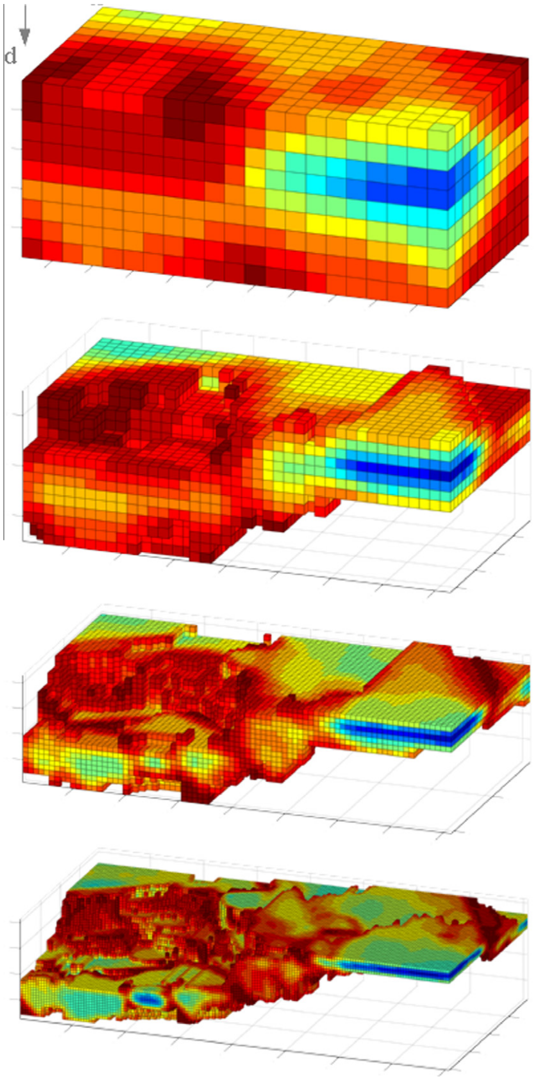


Fig. 12. Cost volume computation starting from a low resolution stereo pair (a low resolution cost volume). The cost volume is refined as the solution progresses from coarser to finer scales. The search space for disparity of each pixel is independently defined and is usually larger in areas near discontinuities. The minimum cost solution is seen in blue. Slices in x–d and x–y levels (discussed in Section 4.1) are also visible. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

6.1. Hierarchical vs original approach

The use of the hierarchical scheme is evaluated here against the original approach regarding the two aforementioned objectives: guidance of matching across scales and computational efficiency. In Fig. 13 the improvement of results is displayed when exploiting the hierarchical scheme on pair #51 of the KITTI dataset. The figure represents, admittedly, an extreme case as the error percentage without the hierarchical scheme is 67%, while dropping to 17% when it is activated. One may notice that texture-less regions, e.g. road or wall surfaces, result in very poor local minima for the cost function. It is in these cases that the hierarchical scheme helps the most, by gathering wider spatial information in low resolutions and using it to restrain the solution in larger scales. Another way

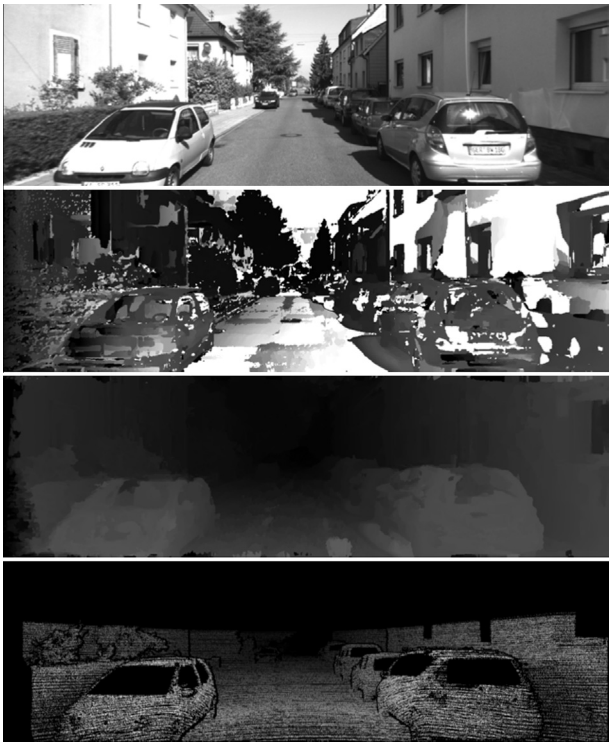


Fig. 13. Disparity maps (levelled between 0 and 90 greyscale values) for stereo pair #51 of the KITTI training set. From top to bottom: original base image; disparity map produced without matching across scales; disparity map by coarse-to-fine matching; ground truth map.

Table 2
Percentage of erroneous disparities for the complete KITTI training set (*nonocc*).

Core matching		Refinement process	
No hierarc	Hierarc	No hierarc	Hierarc
26.38	14.05	20.85	12.89

for strengthening matching solution would be to use very large support regions by relaxing the thresholds defining cross regions, but this would assume fronto-parallel surfaces and yield low accuracy.

In Table 2 the results from the complete KITTI training set are reported and compared regarding the achieved accuracy. The error rate for the core matching process is 26.38 when images are matched only on the original resolution and 14.05 when matched across image pyramid. On the other hand, if the refinement process (Section 7) is used results improve by 8% when a coarse-to-fine matching is performed.

An intermediate observation is that refinement improves the results by ~6% when the coarse-to-fine option is disabled in the method, while without refinement matching performed across scale spaces achieves an improvement of more the 12%. This observation is of importance, because one can choose to drop the refinement procedure when speed is more essential than accuracy. The refinement procedure actually represents a time bottleneck of the algorithm performance (Section 8.7). Moreover, matching directly on the finer scale requires 85% more time for the basic procedure (matching and DSI smoothing), a percentage which is predominantly due to the large disparity range. Because of this, the gap in running time between the presented hierarchical matching and the single-scale version widens as the image size grows.

7. Post-processing

7.1. Constraints

7.1.1. Left–right consistency

Image matching consistency (*cross-checking*) between reference and matching images is a common reliable tool for evaluating the quality of disparity maps (Banks and Corke, 2001; Brown et al., 2003). In local stereo algorithms this is easy to implement by computing matches from the matching to the reference image, thus creating the disparity map of the reference image. A pixel **p** is characterized as valid (inlier) if its absolute disparity value and the absolute value of its match in the matching image are equal. The left–right consistency check does not make any distinction among outliers of different origins (i.e. mismatches, occlusions), yet it performs well in eliminating erroneous disparity (depth) estimates. Results from this check can also be exploited for tuning the parameter set which controls a matching algorithm. Finally, it is noted that (as mentioned in Section 5) the ordering and uniqueness constraints are also used.

7.2. Outlier median smoothing via cross-based regions

Post-processing of the disparity map by utilizing cross-based support regions has been proposed by Lu et al. (2008); Zhang et al. (2009). Originally, a method of *voting bins* was described for selecting the most frequent disparity value in the support area. This can be helpful in cases of a poor cost function and of support regions of high confidence. Unfortunately, such an approach aggravates the mentioned problems due to assumed fronto-parallelism in local methods. Nonetheless, this is not the case for outliers which have been located through the left–right consistency check; the use of adaptive cross-based regions can provide useful information for these incorrect disparities. Thus, for an outlying pixel **p** its cross-based region $S(\mathbf{p})$ is formulated on the left image and the inliers in $S(\mathbf{p})$ are detected. The median of these valid disparities is attributed to **p** if the number of inliers within $S(\mathbf{p})$ exceeds a certain threshold. The above is particularly beneficial in occlusion areas (Fig. 14). The method described is iterative since it is a “region closing” technique for large areas of outliers, such as occlusion areas, which are progressively filled with neighbouring disparities.

7.3. Occlusion/mismatch labeling

In the preceding steps, outliers are located in the disparity map and corrected through cross-based region smoothing. Although disparities are improved, errors still exist since the effectiveness of the previous step depends on the suitability of the support regions. Initial outliers located in Section 7.1 may be categorized into *occlusions* and *mismatches*. A thorough investigation of occlusion detection techniques is found in Egnal and Wildes (2002). Here, the technique of Hirschmüller (2008) based on epipolar geometry has been implemented. Occlusions have been corrected via cross-based region smoothing, since no actual information is available in both images. For mismatches, however, the left–right check runs again to locate remaining errors and correct them through interpolating from nearest inlier neighbours.

7.4. Sub-pixel estimation

Finally, a sub-pixel estimation is performed (Yang et al., 2009). This is done by interpolating a 2nd order polynomial curve to $C_{\mathbf{p}}(d)$, which is the cost function with respect to disparity per pixel (d direction of DSI):

$$C(d) = \alpha \cdot d^2 + \beta \cdot d + \gamma \qquad (11)$$

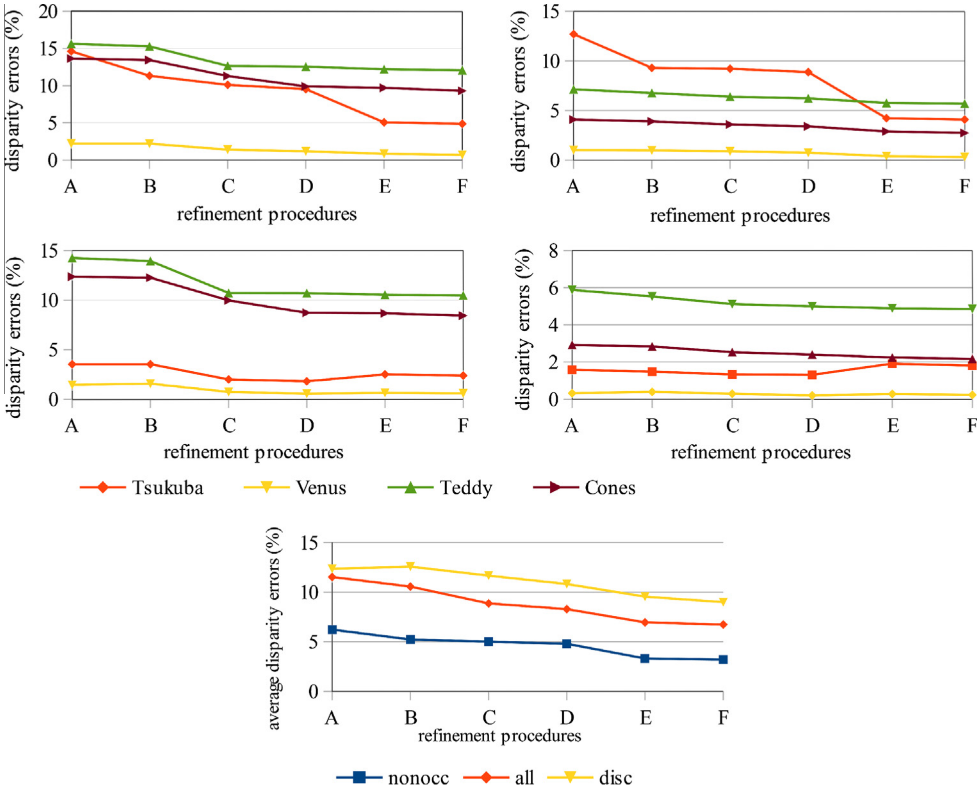


Fig. 14. Performance of each refinement procedure regarding disparity map accuracy. Top: percentage of erroneous pixels in the complete image (*all*) and nonocclusion areas (*nonocc*) for the 0.75 pixel threshold. Middle: same as above for the 1 pixel threshold. Bottom: average errors of all four Middlebury datasets for each comparison scheme separately: nonoccluded pixels (*nonocc*), all image pixels (*all*), areas near discontinuities (*disc*). A: initial disparity map, B: cost smoothing, C: outlier cross-based filtering, D: remaining occlusion/mismatch handling, E: sub-pixel estimation, and F: bilateral filter and median smoothing.

This curve is defined by the cost values of the disparities of the preceding and following pixels of the *winner-takes-all* solution. The optimal sub-pixel disparity value $d_{\text{sub-opt}}$ is determined by the minimum cost position around the WTA solution (d_{WTA}) through a closed-form solution for the three cost values [$C(d_{\text{WTA}}-1)$, $C(d_{\text{WTA}})$, $C(d_{\text{WTA}}+1)$]:

$$d_{\text{sub-opt}} = \left(C_{\mathbf{p}}(d_{\text{WTA}}+1) - C_{\mathbf{p}}(d_{\text{WTA}}-1)\right)/\left(2\cdot\left(C_{\mathbf{p}}(d_{\text{WTA}}+1) - 2\cdot C_{\mathbf{p}}(d_{\text{WTA}}) + C_{\mathbf{p}}(d_{\text{WTA}}-1)\right)\right) \tag{12}$$

Information from the cost function is thus exploited for the sub-pixel estimation of disparity map.

7.5. Disparity map smoothing

Edge-preserving smoothing is needed after sub-pixel estimation to remove noise from disparity maps and, as a consequence, improve the quality of reconstructed surfaces. Here, *bilateral filtering* is preferred (Tomasi and Manduchi, 1998) because it is non-linear and non-iterative, in contrast to other efficient filters based on iterative, computationally expensive schemes (e.g. total variation). The filter is widened along the scan-lines (3 × 21) to achieve smoother 3D reconstruction along epipolar lines. In the tests with the Middlebury stereo-pairs, the overall result remains basically unaffected by bilateral filtering, but for all other datasets improvement is noticeable. This is obvious mainly in the quality of the reconstructed point clouds in Section 8. A final median kernel is applied on the disparity map, as a "stronger" edge-preserving filter. This helps restore coherence among smoothened disparities of adjacent epipolar lines and improve the edges of the disparity map; the effect of this last step is observed in areas around discontinuities, as seen in the lowest diagram of Fig. 14. The effect of the overall post-processing refinement is illustrated further below (Fig. 17).

Finally, the charts in Fig. 14 present the improvement obtained at each refinement step for the 0.75 and 1 pixel thresholds for the Middlebury pairs. The first two rows display the improvement per step for every stereo-pair. The errors refer to non-occluded areas (*nonocc*) and to the whole image (*all*). In the last chart, the average error of all four stereo-pairs under the 0.75 pixel threshold is presented per each category (*nonocc*, *all* and *disc*, i.e. areas near discontinuities).

In the lowest chart one may observe that 3D cost smoothing (B) slightly deteriorates the disparities in areas around discontinuities. This is to be expected since the Gaussian function used here as basis of the 3D smoothing kernel does not preserve edges. Despite this, the contribution of 3D smoothing to disparity improvement is seen at the *nonocc* and *all* curves, and primarily at the results presented later for other datasets. Also, the improvement from step E to F is small, but distinct for the areas near discontinuities (as shown in the lowest diagram of Fig. 14).

8. Results

The presented algorithm has been evaluated on the Middlebury on-line platform and further tested using other available data sets as well as our own imagery.

8.1. Evaluation on the Middlebury platform

The set of parameter values used here are seen in Table 3; results are shown in Figs. 15 and 16.

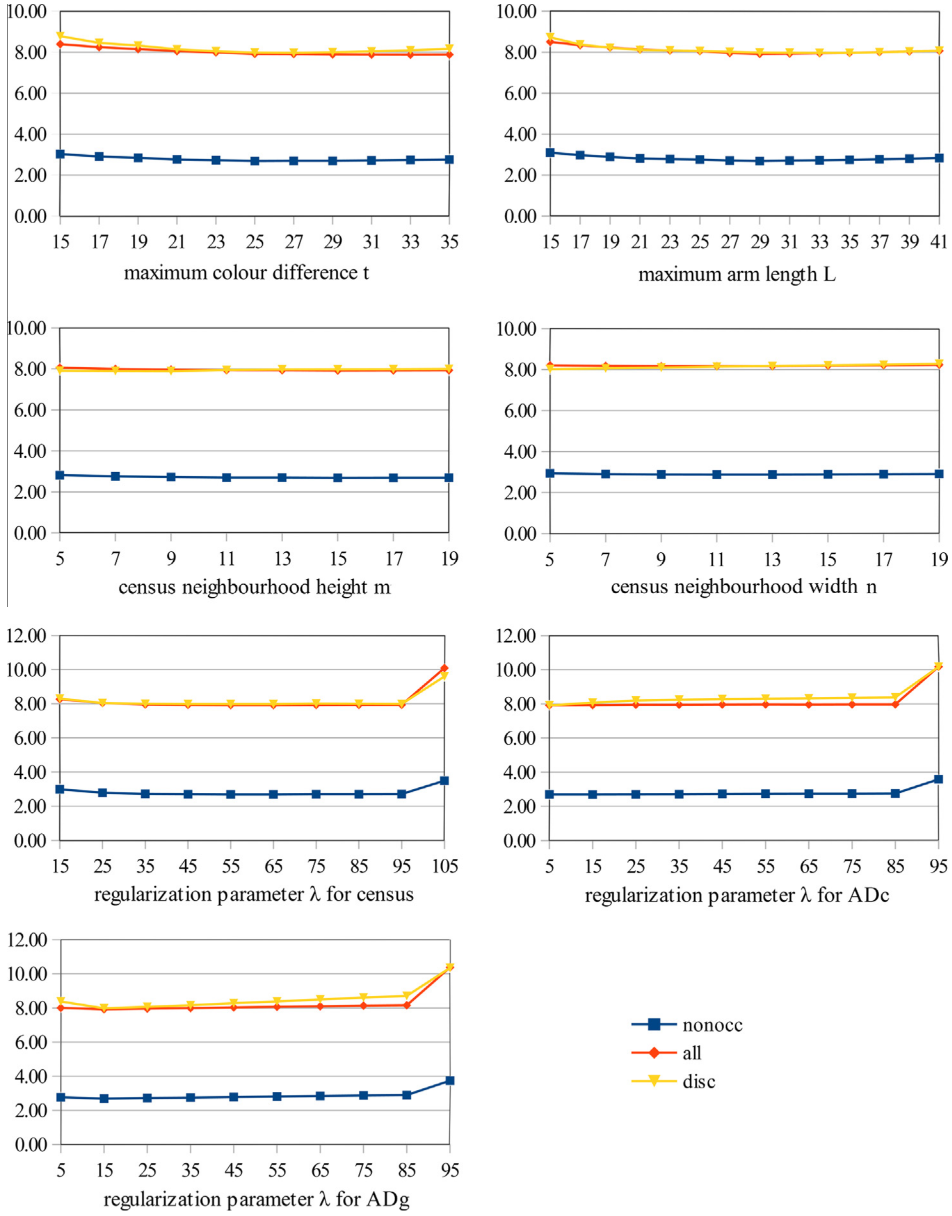


Fig. 15. Diagrams presenting the response of the algorithm to the tuning of individual parameters with the rest of the parameter set remaining constant. The average errors of all four stereo pairs for each evaluation area (*nonocc*, *all*, *disc*) are displayed.

Figs. 15 and 16 display the stability of the algorithm to parameter tuning. Results are stable within a wide range of values for each parameter. Only the maximum colour difference t and the maximum length L play a more essential role in tuning as the corresponding curves have a more obvious minimum. The size of census filter (m,n) transforming the image does not affect significantly the performance, hence it is reasonable to select a small size to reduce computational load. Regularization parameters λ also prove to be stable, and it is only for extreme values that the performance deteriorates. It is noted that in areas around discontinuities the errors are more affected by the change of λ, especially for parameter λ_{ADg}. When examining the effects of parameter changes on individual stereo-pairs of the dataset (Fig. 16) variation in performance is stronger than for the average of the stereo-pairs.

Final results are seen in Figs. 17 and 18. In Fig. 17 the estimated disparity maps, the erroneous pixels and the true disparity maps are shown for each stereo pair. The algorithm performance is indeed encouraging, as it is rated among the top algorithms reported in the Middle bury evaluation platform. When comparing for the >1 pixel error tolerance, our approach scores worse than *ADCensus* (Mei et al., 2011) which is currently at the 2nd position of this plat form rating and uses a similar aggregation support region based on support skeletons (Zhang et al., 2009). On the other hand, a major improvement is observed when comparing for subpixel accuracy. The pipeline proposed in this paper results in a high performance under the error thresholds of 0.5 and 0.75 pixels in disparity values differences. In fact, it rates 4th in the 0.75 pixels threshold comparison (2nd when initially submitted), while most topper forming methods for the 1 pixel threshold give poor results for sub-pixel testing, regardless of optimization method (local/global). Average % error in our case is 6.15 (LAMC-DSM: *Local Adaptive Multi-Cost Dense Stereo-Matching* in Fig. 18). Good sub-pixel ac cu racy is important for several applications since it significantly improves the quality of 3D reconstruction and of the triangular meshes subsequently produced. The lack of sub-pixel accuracy

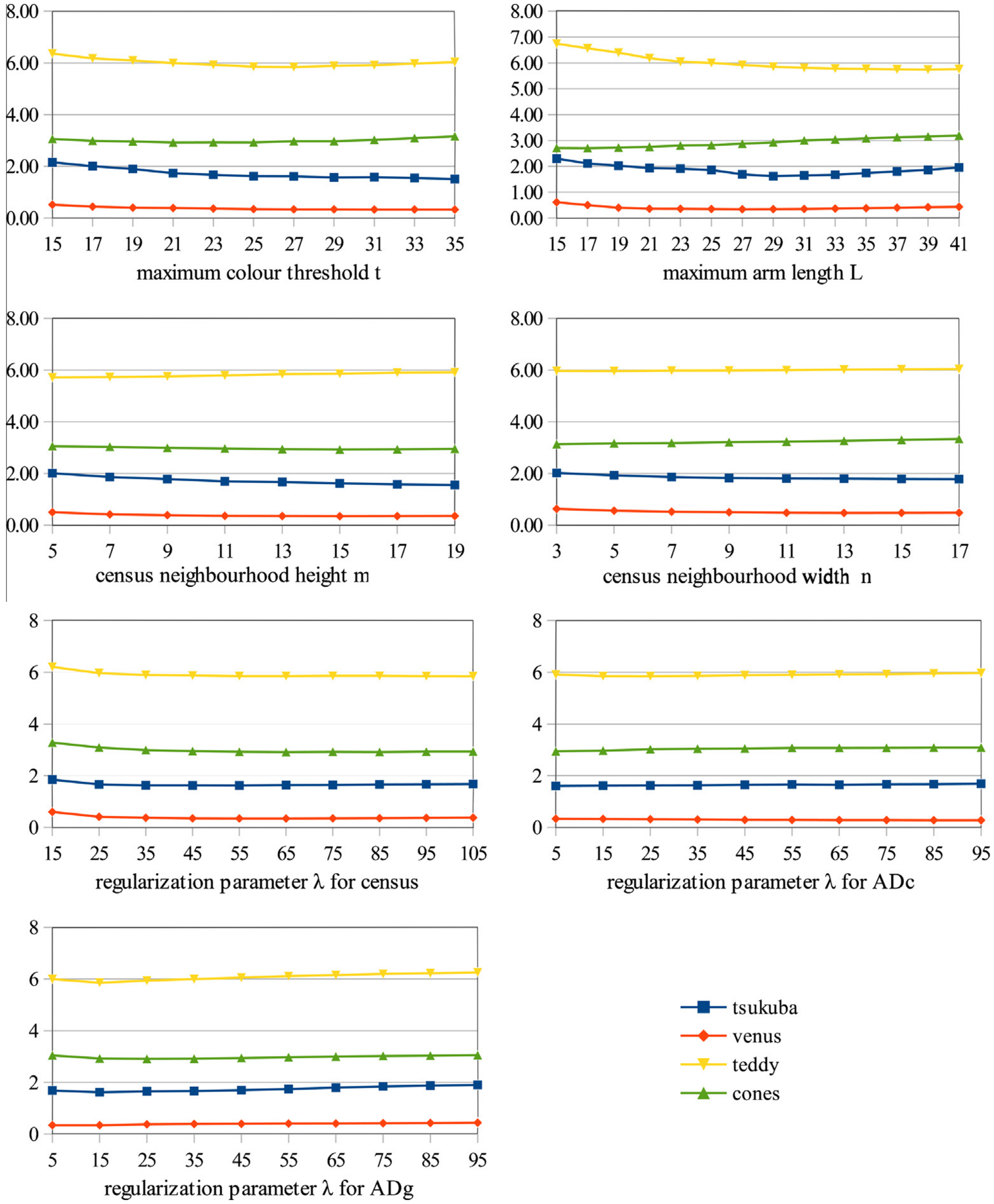


Fig. 16. Diagrams presenting, as in Fig. 15, the response of the algorithm to the tuning of individual parameters, but in this case the response to parameter tuning is presented separately for each data-set only for non-occluded pixels.

leads to poor 3D meshes because of the discrete values of depth information (quantization effect).

At this point one might mention that the Middlebury platform has been "saturated" over the years. Although this evaluation framework has indeed triggered new theories and implementations, the differences today in algorithm ranking may in some cases be attributed to over-fitting. Images from the 2006 datasets are interesting, but do not form part of the ranking framework. Hence, these tests will probably continue to serve stereo matching, but new datasets with more realistic and diverse scenarios are definitely required.

8.2. EPFL multi-view data-set

Strecha et al. (2008) published a benchmark for high resolution images. Ground truth data are derived from laser scanner at Ecole Polytechnique Fédérale de Lausanne (EPFL) computer vision laboratory (http://cvlabwww.epfl.ch/~strecha/multiview/). Fig. 19 shows a stereo pair (6 Mp images: 0006.png, 0007.png) from the *Herz-Jesu-K7* multi-view data set and respective matching results. An indication for the accuracy of reconstruction has been gained by registering the generated point cloud onto the ground truth data via the ICP surface matching algorithm (Fig. 19, bottom). The overall mismatch is represented by an average distance of 10 mm and a standard deviation of 19 mm. Reduced to mean image scale, these values correspond to ~1.8 and ~3.4 pixels, which are considered as quite satisfactory.

8.3. Column capital

The proposed method has also been tested on an object of high archaeological interest (Fig. 20), namely a column capital from the temple of Athena Nike on the Acropolis of Athens (http://ysma.gr/en/athena-nike). The images (camera: 12 Mp Canon EOS5; pixel size: 8.24 μm) had originally been taken for creating an orthomosaic of the capital with conventional photogrammetric techniques

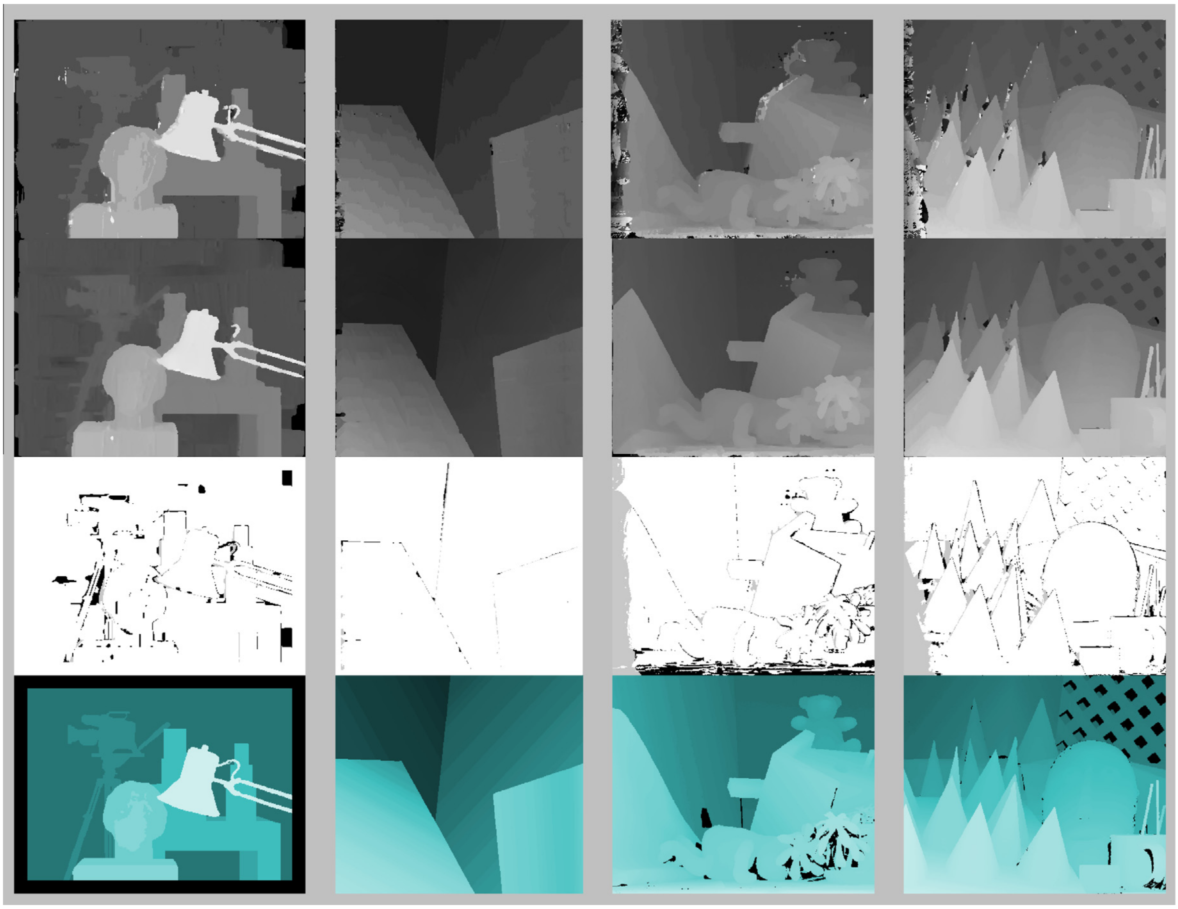


Fig. 17. First row: resulting disparity maps before multi-step refinement process on the four stereo pairs of the Middlebury evaluation platform. Second row: final resulting disparity maps after the matching procedure is completed. Third row: "bad" pixels of the produced disparity maps evaluated under the 0.75 pixel error threshold; mismatched pixels in occluded areas are indicated by grey colour, in non-occluded areas in black. Last row: true disparity maps referring to the left image of each stereo-pair.

Table 3
Parameters values used for all images.

Census transformation	m	11	Lambda λ_c	45
	n	9	Lambda λ_{ADc}	5
Length threshold $L_{\max}$		31	Lambda λ_{ADg}	18
Colour threshold $\tau_{\max}$		24	Cost smooth k	5
Cross-based smooth iterations		5	Bilateral filter σ_I, σ_D	1,7.5

and form part of the Acropolis Restoration Service archive. The orientation parameters were determined with our automatic bundle adjustment software (control points simply served for scaling purposes). Three pairs were used for complete reconstruction, yet matching was based on stereo.

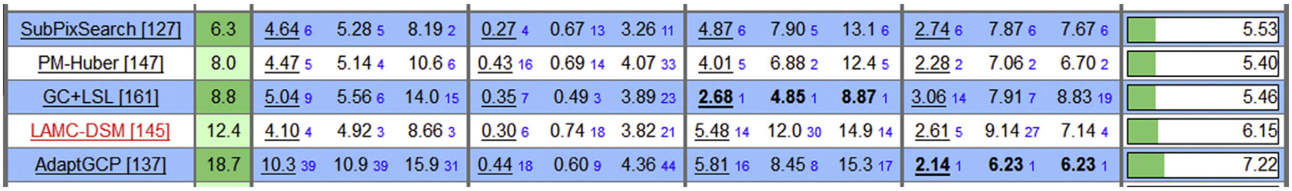
Although in this case no ground truth was available, optical inspection supports the claim that the 3D reconstruction is quite satisfactory.

8.4. DMC aerial images

In Fig. 21 the performance of the algorithm on a part of an aerial image pair is presented. The DMC images and their orientations were obtained from the *Photomod* software demo. Estimated disparity maps, textured isometric plots and the reconstructed model are presented.

8.5. KITTI dataset

Finally, the algorithm was tested on the Karlsruhe Institute of Technology and Toyota Techno logical Institute (KITTI) benchmark for stereo vision in autonomous navigation (http://www.cvlibs.net/datasets/kitti; Geiger et al., 2012). This consists of 195 particularly challenging stereo-pairs depicting real world scenes. Highly inclined surfaces, non-textured or shaded areas and extreme



SubPixSearch [127]	6.3	4.64 6	5.28 5	8.19 2	0.27 4	0.67 13	3.26 11	4.87 6	7.90 5	13.1 6	2.74 6	7.87 6	7.67 6	5.53
PM-Huber [147]	8.0	4.47 5	5.14 4	10.6 6	0.43 16	0.69 14	4.07 33	4.01 5	6.88 2	12.4 5	2.28 2	7.06 2	6.70 2	5.40
GC+LSL [161]	8.8	5.04 9	5.56 6	14.0 15	0.35 7	0.49 3	3.89 23	**2.68** 1	**4.85** 1	**8.87** 1	3.06 14	7.91 7	8.83 19	5.46
LAMC-DSM [145]	12.4	4.10 4	4.92 3	8.66 3	0.30 6	0.74 18	3.82 21	5.48 14	12.0 30	14.9 14	2.61 5	9.14 27	7.14 4	6.15
AdaptGCP [137]	18.7	10.3 39	10.9 39	15.9 31	0.44 18	0.60 9	4.36 44	5.81 16	8.45 8	15.3 17	**2.14** 1	**6.23** 1	**6.23** 1	7.22

Fig. 18. Results from the Middlebury evaluation platform for the 0.75 pixel threshold. Columns from left to right: method; average rank; errors for the Tsukuba, Venus, Teddy and Cones stereo pairs; and average percent of bad pixels (errors are recorded for cases of non-occluded pixels, all pixels, discontinuities). Date of evaluation: February 3, 2014.

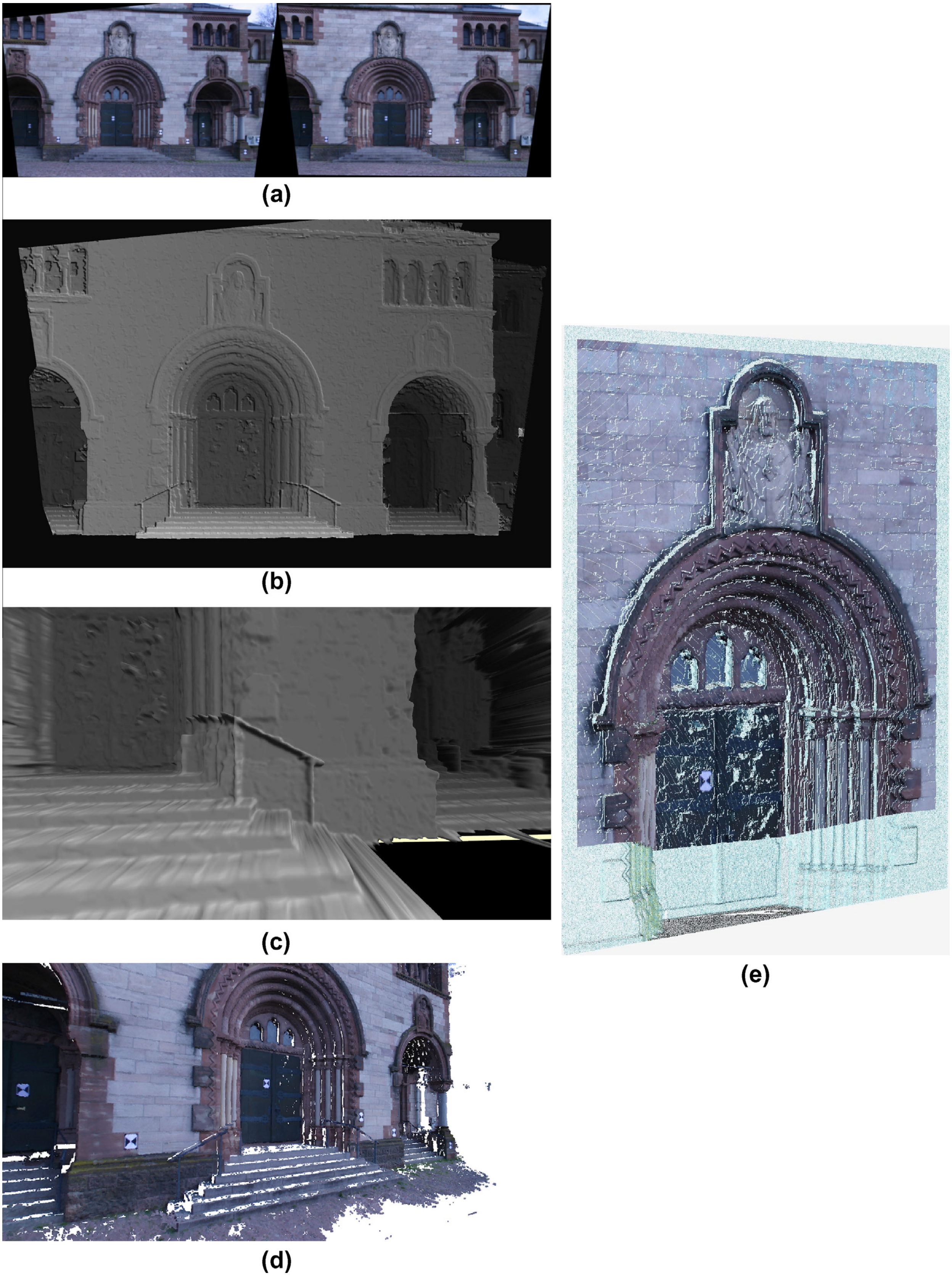


Fig. 19. Epipolar Herz-Jesu-K7 stereo-pair (a); disparity map (b); detail of the 3D representation of the disparity map (isometric plot of the disparity map) (c); textured point cloud (d); reconstructed point cloud registered to ground truth data (e).

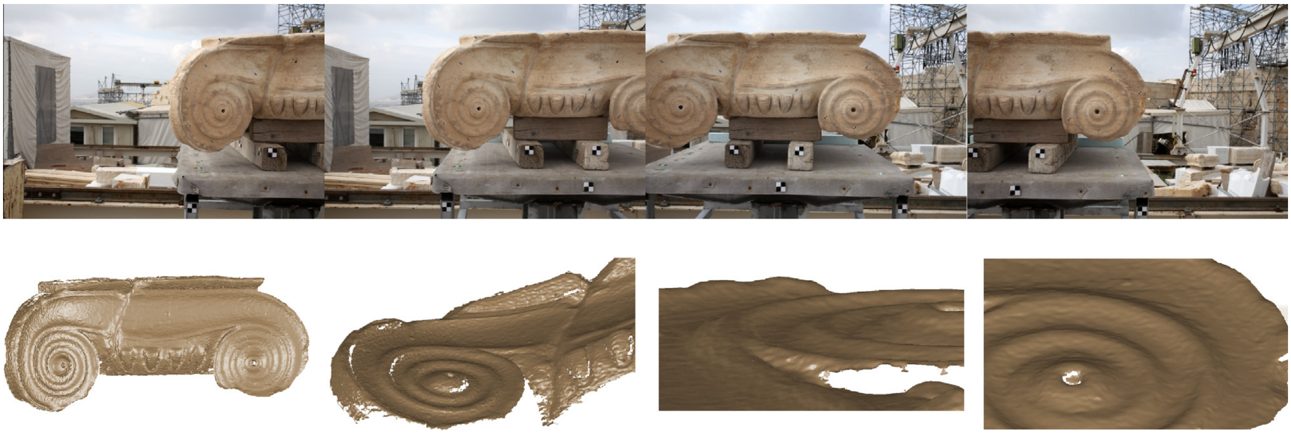


Fig. 20. Stereo-matching results for a column capital (the original four images are seen on top).

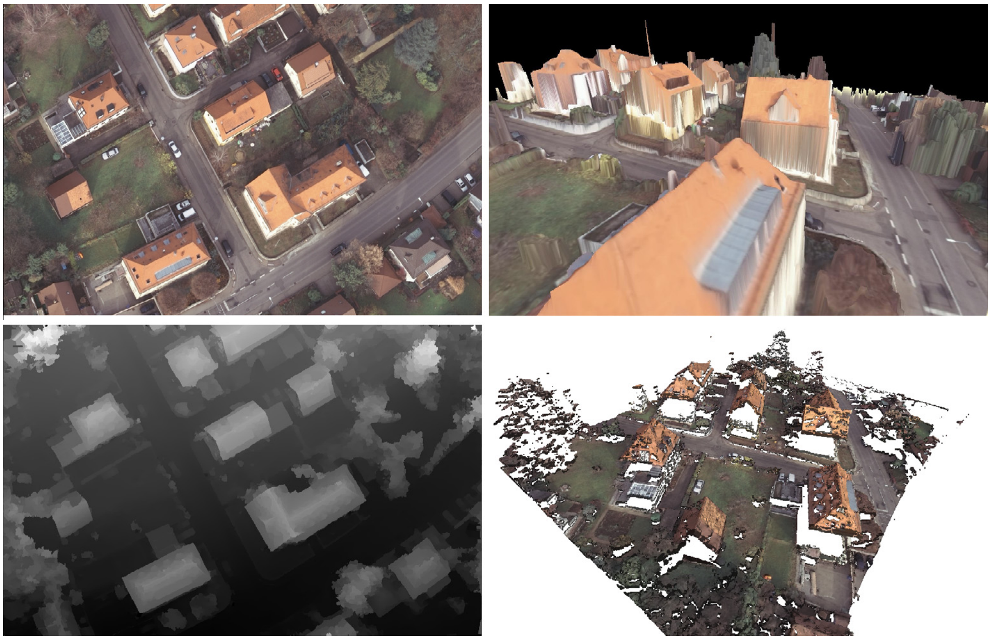
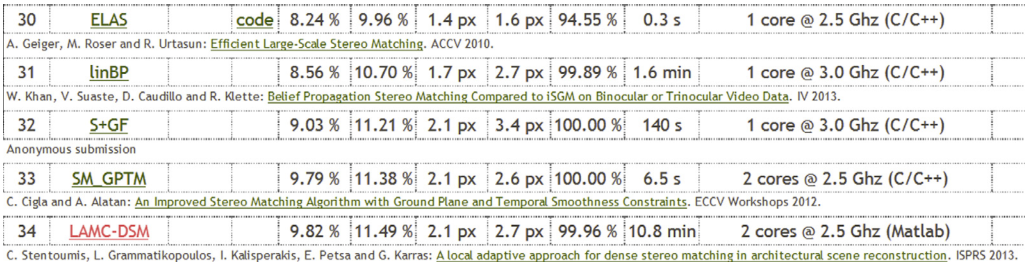


Fig. 21. Stereo-matching results for aerial images. Reference image, textured isometric plot of the disparity map, reconstructed point cloud and estimated disparity map (clockwise from top left).



30	ELAS		code	8.24 %	9.96 %	1.4 px	1.6 px	94.55 %	0.3 s	1 core @ 2.5 Ghz (C/C++)
A. Geiger, M. Roser and R. Urtasun: Efficient Large-Scale Stereo Matching. ACCV 2010.										
31	linBP			8.56 %	10.70 %	1.7 px	2.7 px	99.89 %	1.6 min	1 core @ 3.0 Ghz (C/C++)
W. Khan, V. Suaste, D. Caudillo and R. Klette: Belief Propagation Stereo Matching Compared to iSGM on Binocular or Trinocular Video Data. IV 2013.										
32	S+GF			9.03 %	11.21 %	2.1 px	3.4 px	100.00 %	140 s	1 core @ 3.0 Ghz (C/C++)
Anonymous submission										
33	SM_GPTM			9.79 %	11.38 %	2.1 px	2.6 px	100.00 %	6.5 s	2 cores @ 2.5 Ghz (C/C++)
C. Cigla and A. Alatan: An Improved Stereo Matching Algorithm with Ground Plane and Temporal Smoothness Constraints. ECCV Workshops 2012.										
34	LAMC-DSM			9.82 %	11.49 %	2.1 px	2.7 px	99.96 %	10.8 min	2 cores @ 2.5 Ghz (Matlab)
C. Stentoumis, L. Grammatikopoulos, I. Kalisperakis, E. Petsa and G. Karras: A local adaptive approach for dense stereo matching in architectural scene reconstruction. ISPRS 2013.										

Fig. 22. Results from the KITTI evaluation platform for the default 3 pixel threshold. Columns from left to right: rank; method; percentage of erroneous pixels in non-occluded areas; percentage of erroneous pixels in total; average disparity/end-point error in non-occluded areas; average disparity/end-point error in total; density of disparity map; runtime; environment. Date of evaluation: February 3, 2014.

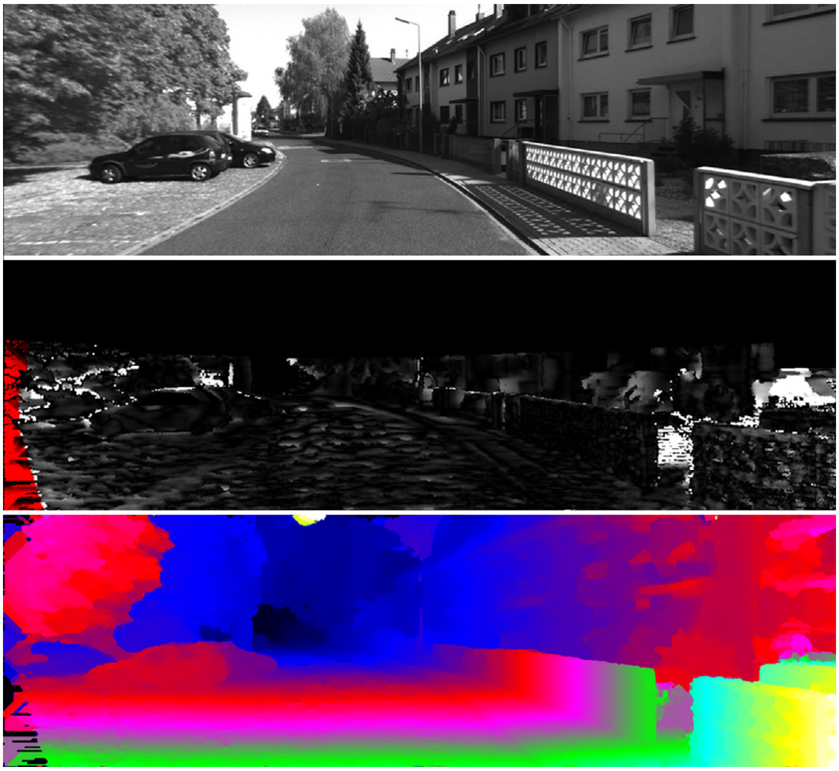


Fig. 23. Matching results for stereo pair 5 of the KITTI evaluation platform. Top: reference (left) image. Middle: disparity error map. The error map accounts for 0 (black) to ⩾5 (white) pixel error; pixels outside the right image area are marked in red. Bottom: estimated disparity map. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

radiometric changes are some of the aspects differentiating this benchmark from others. Besides, these grey-scale images do not allow the algorithm to take advantage of colour. Most methods top-performing in Middlebury benchmark rank low in these scenes.

Fig. 22 presents a screenshot of the on-line evaluation platform. In Fig. 23 an example is seen including the estimated error and disparity maps. Detailed results for the overall performance of our algorithm and the performance for individual stereo-pairs on the KITTI benchmark are found on the evaluation page http://goo.gl/X4OhL.

These results are indeed promising, however further considerations are needed to improve the performance of *LAMC–DSM* in such scenes. Most errors are due to mismatches on the coarser scale which are carried across the pyramid because of the limitations of the cross-based windows. When compared to other images tested, the KITTI images are generally blurred and sometimes partly over-saturated, and hence more prone here to incorrect matches when compared to global approaches. Also, large portions of these images represent flat surfaces highly inclined against the image plane, thus severely violating the assumption of fronto-parallelism. Moreover, such flat surfaces suffer from a lack of texture. Global methods can improve the disparity map thanks to smoothness constraints, but local methods have to balance the support region size with the ability to describe non-frontoparallel surfaces. Finally, the absence of colour is a further limiting factor.

Very "difficult" stereo-pairs had errors close to 25%, thus lowering the overall performance of the algorithm. A possible treatment of this problem may be in the direction of restricting disparity tolerance by robust feature points (and certainly of defining a new set of parameters for the algorithm). The rank of our algorithm (34th) has dropped significantly (∼15 positions) since the results were first submitted. This rapid change over a period of some months might be seen as an indication of the "potential" of this particular dataset.

8.6. A comparison with semi-global matching

In this section, results from comparing the presented matching scheme against *semi-global matching* (SGM) approaches (Hirschmüller, 2008) are reported. We use our own implementation of SGM in 8 directions to optimize the cost function discussed in Section 2 and evaluate the results on Middlebury and KITTI benchmarking platforms. We also compare the above two stereo configurations against the results for SGM under *Mutual Information* (MI) cost (for Middlebury images) and *census* cost (for KITTI images) reported by the author. When using our multi-cost function the hierarchical local disparity selection performs better than the semi-global optimization in both evaluation platforms, but results are more complicated if the cost changes.

In more detail, our hierarchical local adaptive approach yields results which are better by 1.15% than SGM in non-occluded areas of the Middlebury tests and by 6.33% in the vicinity of discontinuities when optimizing the proposed cost function. Using this cost, the local matching scheme also outperforms SGM in KITTI evaluation by 5%. The comparisons refer to the core procedure without the post-processing steps. If the latter are used, LAMC–DMC is still better (by 2.63%) than SGM under MI cost in Middlebury images. On the other hand, SGM with *census* is reported as being better than LAMC–DMC by 4% in KITTI (SGM reports 85% completeness of the disparity map, whereas we produce complete maps). Obviously, when the full matching scheme (i.e. including post-processing) is used the different post-processing steps differentiate the comparisons, but they are indicative of the limits of each method. However, the matching cost measure optimized in

SGM is not the same for the reported results in the two platforms and, apparently, neither are the parameter values. In any case, the reported SGM result is better in KITTI, which might be partly attributed to the different tuning of the algorithm, compared to Middlebury stereo pairs (we have kept the same values for our parameters, as already mentioned).

8.7. Algorithm efficiency and bottlenecks

Here we will give an insight about the computational efficiency of the algorithm; memory consumption and running time will also be commented upon. The presented matching scheme has been implemented in Matlab technical language for rapid prototyping; hence, very little attention has been given, at this stage, to optimizing computational load. Although our algorithm is not adapted to hardware, nor is it optimized for real-time applications, its complexity can be computed based on theoretical considerations and relevant publications (Wang et al., 2006; Nalpantidis et al., 2008; Sizintsev and Wildes, 2010).

The theoretical complexity of the hierarchically structured algorithm, which is reduced by appropriate treatments as discussed later on, is described by the following equation:

$$\begin{aligned} &O(w_I h_I d n_S) + O((w_I h_I)/4 \cdot d n_S) + O((w_I h_I)/16 \cdot d n_S) + \ldots \\ &\quad + O\Big((w_I h_I)/2^{2(s-1)} \cdot d n_S\Big) < (1+1/3) \cdot O(w_I h_I d n_S) \simeq O(w_I h_I d n_S) \end{aligned} \tag{13}$$

where w_I and h_I denote the width and height of image I, d is the disparity range, s the pyramid layer and n_S the number of pixels in support region S. Disparity labels d and pixel number n_S are variable, as they both depend on the adaptive shape of support regions S on each scale; consequently, the maximum load is determined by the maximum disparity range and dimensions of the support region. However, d and n_S are generally much smaller than w_I and h_I. Besides, although the number of pixels participating in the *aggregation* step (Section 3) is arbitrary, the use of *integral* images decomposes area summation, so that each pixel participates once in two sequential 1D summations of constant time, increased by the time needed to query a lookup table for the cross skeleton arms. Complexity does not increase for post-processing steps, as each pixel participates a constant number of times in operations, hence complexity can be roughly approximated by $O(N_{\text{pix}} \cdot d)$, where $N_{\text{pix}} = w_I \times h_I$ is the number of image pixels. Cost smoothing is performed through convolving with a constant 3D filter, and $O(N_{\text{pix}} \cdot d)$ implementations can be used.

Markov Random Fields models represent a common and effective approach in stereo. MRFs can be optimized by solving network flow problems on graphs, which has complexity $O(\mathrm{V} \cdot \mathrm{E}^2)$. For a full graph with one node for every pixel at every disparity (e.g. Ishikawa et al., 1998) this amounts to $O(N_{\text{pix}}^3 \cdot d^3)$, since the number of edges per vertex is constant. Alternatively, the α-*expansion* solves the multi-label graph-cut as a sequence of binary graph-cuts over the d disparities, which is iterated i times until convergence, hence $O(N_{\text{pix}}^3 \cdot d \cdot i)$. Empirically, the complexity on regular image grids is however only about $O(N_{\text{pix}}^{1.2} \cdot d^{1.3})$ (Roy and Cox, 1998; Boykov and Kolmogorov, 2004). If MRF inference is instead done with i iterations of (loopy) *belief propagation*, the complexity is $O(N_{\text{pix}} \cdot d^2 \cdot i)$ (Sun et al., 2003), but it can be reduced to $O(N_{\text{pix}} \cdot d \cdot i)$ for image rasters with the help of distance transforms (Felzenszwalb and Huttenlocher, 2004). Historically, *dynamic programming* ("scanline stereo") was common in stereo matching (Ohta and Kanade, 1985), with complexity $O(N_{\text{pix}} \cdot d)$. The more recent semi-global method requires solving the dynamic program for a constant number of times (one per direction) and thus has the same complexity $O(N_{\text{pix}} \cdot d)$ (Hirschmüller, 2008).

Regarding memory, the largest variable to be stored is the DSI representation, which is an $O(N_{\text{pix}} \cdot d)$ volume that has to be processed. It is possible for local methods to bypass the construction of DSI and its complete allocation to memory, but in the present algorithm we exploit it for integral image summation and geometrically constrained cost smoothing, hence there is a memory/speed gain trade-off. The DSI volume is stored in uint32 variables, but a more dedicated implementation would use fewer bits for integer values. Nevertheless, memory consumption is kept significantly lower than GC, BP. Variational methods consume $O(N_{\text{pix}}^2)$ memory based on *total variation* regularization. Our algorithm, as most local or semi-global implementations, and in contrast to global methods, allows images tiling with tolerable effort. We can thus process images of any size, if appropriate tile overlap is used. It is noted that in our case tiling has been needed only for handling aerial images. Tiling can also be exploited for parallel processing in order to improve speed.

Concerning speed, a main advantage of local matching is that it can, inherently, be implemented for parallel processing on commercial graphic cards and FPGAs (Nalpantidis et al., 2008). The hierarchical scheme imposes a sequential structure of the algorithm, but this is actually not a serious drawback since tasks can be effectively programmed per scale. The cost volume is computed independently for each pixel, thus it can run simultaneously for any number of threads, and *census* matching cost computation is fast (binary descriptor). Computing cross skeletons and smoothing via cross-regions represent the speed bottlenecks of the algorithm, but both processes can be implemented in parallel for each direction or/and for each pixel. The adaptive definition of disparity range does not allow reducing worst-case complexity, but in practice a very limited range is searched in most regions except areas near surface boundaries.

Running times reported at the cited evaluation platforms correspond to a non-optimized Matlab implementation with a 2.5 GHz processor. No special consideration has been given to reducing running time or memory consumption, since the main focus was here on the quantitative and qualitative assessment of accuracy. This said, the low computational complexity and the flexibility of the algorithm are not compromised, since each component respects the inherent simplicity and scalability of local methods.

9. Concluding remarks

The stereo matching algorithm presented here relies on a multi-component cost function, an adaptive support region and a novel smoothing of cost in the disparity space, supported by a hierarchical scheme. This formulation dispenses with the need for global optimization, as it shows a remarkable tolerance to inclined surfaces, relaxing the assumption of "fronto-parallelism" which is a major limitation of local stereo methods. A series of post-processing steps are applied for refining the initial disparity map and achieve good sub-pixel performance. Our *LAMC–DSM* algorithm has been tested under a variety of circumstances: large-scale images, wide and short-based views, high and lower resolutions. Disparity maps and reconstructed 3D scenes have been presented for evaluating matching results. A comparison to semi-global matching has also been included. The performance is considered as promising, or even satisfactory, according to imaged scene. It is noted that, so far, no differentiation of scenes has been done for the evaluation of the algorithm, although different tuning and extra restrictions, or considerations based on scene particularity, are possible (e.g. stereo-pairs for autonomous driving mainly depict streets and surfaces perpendicular to the image plane). Thus, *LAMC–DSM* is presented here as a general purpose matching scheme, but adaptation according to dataset is expected to improve performance. Special care has been taken in order to ensure that the individual steps of the algorithm are reproducible.

Contributions presented include the use of *census* transformation on image principal gradients and the combination of multiple matching measures in the final cost; a cost smoothing process for 3D support; and a suitable hierarchical scheme. Further aspects are the employment of a linear threshold in the cross-window formulation and a robust post-processing scheme based on known steps. Future research topics include expansion of the algorithm for multiple base-line stereo-matching, but also improvements in the matching algorithm itself and a theoretical justification of census on gradients. Finally, further work needs to be done towards improving 3D local support and further weakening the assumption of fronto-parallel surfaces inherent to local methods.

References

Antunes, M., Barreto, P.J., 2013. Efficient stereo matching using histogram aggregation with multiple slant hypotheses. In: Proc. 6th Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA).

Banks, J., Corke, P., 2001. Quantitative evaluation of matching methods and validity measures for stereo vision. Int. J. Rob. Res. 20 (7), 512–532.

Barnard, S.T., 1986. A stochastic approach to stereo vision. In: Proc. 5th National Conference on Artificial Intelligence, Philadelphia, Penn., pp. 676–680.

Birchfield, S., Tomasi, C., 1998. A pixel dissimilarity measure that is insensitive to image sampling. IEEE Trans. Pattern Anal. Mach. Intell. 20 (4), 401–406.

Bleyer, M., Rhemann, C., Rother, C., 2011. PatchMatch stereo – stereo matching with slanted support windows. In: Proc. British Machine Vision Conference (BMVA), pp. 14.1–14.11.

Bobick, A.F., Intille, S.S., 1999. Large occlusion stereo. Int. J. Comput. Vision 33 (3), 181–200.

Boykov, Y., Kolmogorov, V., 2004. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. IEEE Trans. Pattern Anal. Mach. Intell. 26 (9), 1124–1137.

Boykov, Y., Veksler, O., Zabih, R., 2001. Fast approximate energy minimization via graph cuts. IEEE Trans. Pattern Anal. Mach. Intell. 23 (11), 1222–1239.

Brown, M.Z., Burschka, D., Hager, G.D., 2003. Advances in computational stereo. IEEE Trans. Pattern Anal. Mach. Intell. 25 (8), 993–1008.

Collins, R.T., 1996. A space-sweep approach to true multi-image matching. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition CVPR), pp. 358–363.

Cox, J.I., Hingorani, L.S., Rao, B.S., Maggs, M.B., 1996. A maximum likelihood stereo algorithm. Comput. Vis. Image Underst. 63 (3), 542–567.

Crow, F.C., 1984. Summed-area tables for texture mapping. ACM Siggraph Comput. Graphics 18 (3), 207–212.

Dhond, U.R., Aggarwal, J.K., 1989. Structure from stereo – a review. IEEE Trans. Syst., Man Cyber. 19 (6), 1489–1510.

Egnal, G., Wildes, R.P., 2002. Detecting binocular half-occlusions: empirical comparisons of five approaches. IEEE Trans. Pattern Anal. Mach. Intell. 24 (8), 1127–1133.

Faugeras, O., Keriven, R., 1998. Variational principles, surface evolution, PDE's, level set methods, and the stereo problem. IEEE Trans. Image Process. 7 (3), 336–344.

Felzenszwalb, P.F., Huttenlocher, D.P., 2004. Efficient belief propagation for early vision. In: Proc. Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, pp. 261–268.

Foi, A., Katkovnik, V., Egiazarian, K., 2007. Pointwise shape-adaptive DCT for high-quality denoising and deblocking of grayscale and color images. IEEE Trans. Image Process. 16 (5), 1395–1411.

Fusiello, A., Roberto, V., Trucco, E., 1997. Efficient stereo with multiple windowing. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 858–863.

Geiger, A., Lenz, P., Urtasun, R., 2012. Are we ready for Autonomous Driving? The KITTI vision benchmark suite. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3354–3361.

Gong, M., Yang, R., Wang, L., Gong, M., 2007. A performance study on different cost aggregation approaches used in real-time stereo matching. Int. J. Comput. Vision 75 (2), 283–296.

Hafner, D., Demetz, O., Weickert, J., 2013. Why is census transformation good for robust optical flow computation? Scale Space and Variational Methods in Computer Vision, Lecture Notes in Computer Science, Springer 7893, 210–221.

Hirschmüller, H., 2008. Stereo processing by semiglobal matching and mutual information. IEEE Trans. Pattern Anal. Mach. Intell. 30 (2), 328–341.

Hirschmüller, H., Scharstein, D., 2007. Evaluation of cost functions for stereo matching. Proc. Comput. Vis. Pattern Recognit. (CVPR), 1–8.

Hirschmüller, H., Scharstein, D., 2009. Evaluation of stereo matching costs on images with radiometric differences. IEEE Trans. Pattern Anal. Mach. Intell. 31 (9), 1582–1599.

Ishikawa, H., Geiger, D., Burkhardt, H., Neumann, B., 1998. Occlusions, discontinuities, and epipolar lines in stereo. In: Proc. European Conference on Computer Vision (ECCV), vol. 1406, pp. 232–248.

Kanade, T., Okutomi, M., 1994. A stereo matching algorithm with an adaptive window: theory and experiment. IEEE Trans. Pattern Anal. Mach. Intell. 16 (9), 920–932.

Kang, S.B., Webb, J., Zitnick, C., Kanade, T., 1995. A multibaseline stereo system with active illumination and real-time image acquisition. In: Proc. IEEE International Conference on Computer Vision (ICCV), pp. 88–93.

Klaus, A., Sormann, M., Karner, K., 2006. Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. In: Proc. IEEE International Conference on Pattern Recognition (ICPR), vol. 3, pp. 15–18.

Kolmogorov, V., Zabih, R., 2001. Computing visual correspondence with occlusions using graph cuts. In: IEEE International Conference on Computer Vision (ICCV), pp. 508–515.

Lu, J., Lafruit, G., Catthoor, F., 2008. Anisotropic local high-confidence voting for accurate stereo correspondence. In: Proc. SPIE on Image Processing: Algorithms and Systems, vol. 6812, pp. 1–12.

Marr, D., Poggio, T., 1976. Cooperative computation of stereo disparity. Science 194 (4262), 283–287.

Mei, X., Sun, X., Zhou, M., Jiao, S., Wang, H., Zhang, X., 2011. On building an accurate stereo matching system on graphics hardware. In: Proc. ICCV Workshop on GPU in Computer Vision Applications, pp. 467–474.

Moravec, H., 1980. Obstacle avoidance and navigation in the real world by a seeing robot rover. PhD Thesis, Stanford University.

Moravec, H., 1996. Robot spatial perception by stereoscopic vision and 3D evidence grids. In: Technical Report CMU-RI-TR-96-34, Robotics Institute, Carnegie Mellon University.

Nalpantidis, L., Sirakoulis, G.C., Gasteratos, A., 2008. Review of stereo vision algorithms: from software to hardware. Int. J. Optomechatronics 2 (4), 435–462.

Ogale, A.S., Aloimonos, Y., 2004. Stereo correspondence with slanted surfaces:critical implications of horizontal slant. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 568–573.

Ohta, Y., Kanade, T., 1985. Stereo by intra- and inter-scanline search using dynamic programming. IEEE Trans. Pattern Anal. Mach. Intell. 7 (2), 139–154.

Pollard, S., Mayhew, J., Frisby, J., 1985. PMF: A stereo correspondence algorithm using a disparity gradient limit. Perception 14 (4), 449–470.

Prazdny, K., 1985. Detection of binocular disparities. Biol. Cybern. 52 (2), 93–99.

Quam, L.H., 1986. Hierarchical warp stereo. In: Proc. DARPA Image Understanding Workshop, pp. 149–155.

Ranftl, R., Gehrig, S., Pock, T., Bischof, H., 2012. Pushing the limits of stereo using variational stereo estimation. Proc. IEEE Intell. Vehicles Symp., 401–407.

Roy, S., Cox, I.J., 1998. A maximum-flow formulation of the N-camera stereo correspondence problem. In Proc. IEEE International Conference on Computer Vision (ICCV), pp. 492–499.

Scharstein, D., 1994. Matching images by comparing their gradient fields. In: Proc. IEEE International Conference on Pattern Recognition (ICPR), pp. 572–575.

Scharstein, D., Szeliski, R., 1998. Stereo matching with nonlinear diffusion. Int. J. Comput. Vision 28 (2), 155–174.

Scharstein, D., Szeliski, R., 2002. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. Int. J. Comput. Vision 47 (1), 7–42.

Sizintsev, M., Wildes, R.P., 2010. Coarse-to-fine stereo vision with accurate 3D boundaries. Image Vis. Comput. 28 (3), 352–366.

Stentoumis, C., Grammatikopoulos, L., Kalisperakis, I., Karras, G., 2012. Implementing an adaptive approach for dense stereo-matching. Int. Arch. Photogramm., Rem. Sens. Spatial, Inform. Sci. XXXVIII/5, 309–314.

Stentoumis, C., Grammatikopoulos, L., Kalisperakis, I., Petsa, E., Karras, G., 2013. A local adaptive approach for dense stereo matching in architectural scene reconstruction. Int. Arch. Photogramm., Rem. Sens. Spatial, Inform. Sci. XL–5/W1, 219–226.

Strecha, C., 2007. Multi-view stereo as an inverse inference problem. In: PhD Thesis, Katholieke Universiteit Leuven, Belgium.

Strecha, C., Fransens, R., Van, Gool L., 2004. A probabilistic approach to large displacement optical flow and occlusion detection. In: Statistical Methods in Video Processing. Springer, Berlin/Heidelberg, pp. 25–45.

Strecha, C., von Hansen, W., van Gool, L., Fua, P., Thoennessen, U., 2008. On benchmarking camera calibration and multi-view stereo for high resolution imagery. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–8.

Strecha, C., Bronstein, A., Bronstein, M., Fua, P., 2011. LDAHash: improved matching with smaller descriptors. IEEE Trans. Pattern Anal. Mach. Intell. 34 (1), 66–78.

Sun, J., Zheng, N.-N., Shum, H.-Y., 2003. Stereo matching using belief propagation. IEEE Trans. Pattern Anal. Mach. Intell. 25 (7), 787–800.

Sun, X., Mei, X., Jiao, S., Zhou, M., Wang, H., 2011. Stereo matching with reliable disparity propagation. In Proc. IEEE International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission, pp. 132–139.

Terzopoulos, D., 1986. Regularization of inverse visual problems involving discontinuities. IEEE Trans. Pattern Anal. Mach. Intell. 8 (4), 413–424.

Tian, Q., Huhns, M.N., 1986. Algorithms for subpixel registration. Comput. Vis., Graph., Image Process. 35 (2), 220–233.

Tola, E., Lepetit, V., Fua, P., 2008. A fast local descriptor for dense matching. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–8.

Tomasi, C., Manduchi, R., 1998. Bilateral filtering for gray and color images. In: Proc. International IEEE Conference on Computer Vision (ICCV), pp. 839–846.

Tombari, F., Mattoccia, S., Di Stefano, L., Addimanda, E., 2008. Classification and evaluation of cost aggregation methods for stereo correspondence. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–8.

Tombari, F., Mattoccia, S., Di Stefano, L., 2010. Stereo for robots: quantitative evaluation of efficient and low-memory dense stereo algorithms. In: Proc. 11th International Conference on Control Automation Robotics Vision (ICARCV), pp. 1231–1238.

Veksler, O., 2003. Fast variable window for stereo correspondence using integral images. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 556–561.

Veksler, O., 2005. Stereo correspondence by dynamic programming on a tree. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 384–390.

Viola, P., Jones, M., 2001. Rapid object detection using a boosted cascade of simple features. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. I-511–518.

Vogel, C., Roth, S., Schindler, K. 2013. An evaluation of data costs for optical flow. In: Proc. German Conference in Pattern Recognition, 8142, pp. 343–353.

Wang, L., Gong, M., Gong, M., Yang, R., 2006. How far can we go with local optimization in real-time stereo matching? In: Proc. International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT), pp. 129–136.

Xu, Y., Wang, D., Feng, T., Shum, H., 2002. Stereo computation using radial adaptive windows. Proc. IEEE Conf. Pattern Recogn. (ICPR), 595–598.

Yang, Y., Yuille, A., Lu, J., 1993. Local, global, and multilevel stereo matching. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 274–279.

Yang, Q., Wang, L., Yang, R., Stewénius, H., Nistér, D., 2009. Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling. IEEE Trans. Pattern Anal. Mach. Intell. 31 (3), 492–504.

Yoon, K.J., Kweon, I.S., 2006. Adaptive support-weight approach for correspondence search. IEEE Trans. Pattern Anal. Mach. Intell. 28 (4), 650–656.

Yuille, A.L., Poggio, T., 1984. A Generalized Ordering Constraint for Stereo Correspondence. MIT, AI Lab., Memo, 777.

Zabih, R., Woodfill, J., 1994. Non-parametric local transforms for computing visual correspondence. Proc. Eur. Conf. Comput. Vis. (ECCV), 151–158.

Zhang, Y., Gong, M., Yang, Y.H., 2008. Local stereo matching with 3D adaptive cost aggregation for slanted surface modeling and sub-pixel accuracy. In: Proc. IEEE International Conference on Pattern Recognition (ICPR), pp. 1–4.

Zhang, K., Lu, J., Lafruit, G., 2009. Cross-based local stereo matching using orthogonal integral images. IEEE Trans. Circuits Syst. Video Technol. 19 (7), 1073–1079.

Zitnick, C.L., Kanade, T., 2000. A cooperative algorithm for stereo matching and occlusion detection. IEEE Trans. Pattern Anal. Mach. Intell. 22 (7), 675–684.