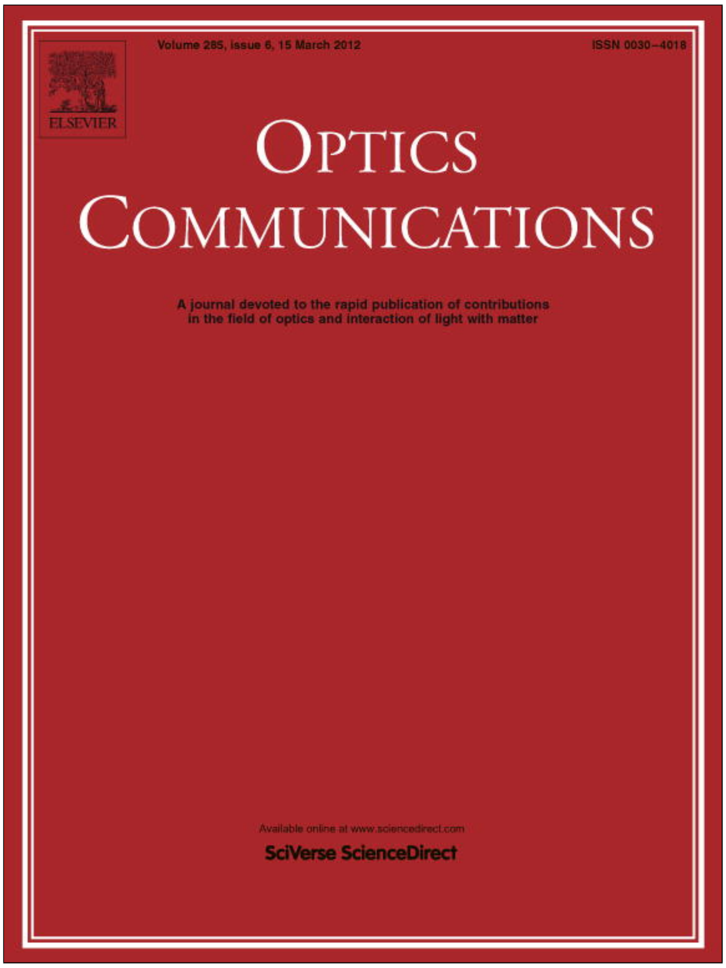
Volume 285, Issue 6, 15 March 2012
ISSN 0030-4018
ELSEVIER
OPTICS COMMUNICATIONS
A journal devoted to the rapid publication of contributions
in the field of optics and interaction of light with matter
Available online at www.sciencedirect.com
SciVerse ScienceDirect

Optics Communications 285 (2012) 1017–1022

Contents lists available at SciVerse ScienceDirect

Optics Communications

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

Stereo computation combining structured light and passive stereo matching

Tomislav Pribanic [a,*], Nenad Obradovic [a], Joaquim Salvi [b]

[a] Faculty of Electrical Engineering and Computing, University of Zagreb, Unska 3, HR-10000, Croatia
[b] Institute of Informatics and Applications, University of Girona, Av. Lluis Santalo s/n, E-17071 Girona, Spain

ARTICLE INFO

Article history:
Received 30 July 2011
Received in revised form 10 October 2011
Accepted 18 October 2011
Available online 31 October 2011

Keywords:
Phase shifting
Phase unwrapping
Structured light
Pasive stereo
3D profilometry
Computer vision

ABSTRACT

In this work we propose a stereo computation method which initially borrows structured light strategy based on single phase shifting approach. An accurate phase shifting code allowed us to considerably decrease the candidate set of points compared to passive stereo matching. Then, once the most similar match was found using area based matching, we re-use the accurate wrapped code to refine the initially found disparity value. Our method is extremely simple to implement and in that sense very promising for real time applications. Shown comparison results demonstrate that we can produce accuracy comparable with the state-of-the art methods in stereo matching.

1. Introduction

Stereo matching is the process of taking two or more images and estimating a 3D model of the scene by finding matching pixels in the images and converting their 2D positions into 3D depths [1]. As a rule there are two approaches available for matching: area based and feature based [2]. The former algorithms match small image windows centered at a given pixel, assuming that the grey levels are similar. They yield dense depth maps, but typically experience serious problems in the vicinity of 3D depth boundaries and fail within occluded areas and/or poorly textured regions. The later algorithms, feature based, heavily rely on feature extraction (e.g. edges, lines, corners) and although they provide more robust features for matching, unfortunately they also provide a sparse depth map ([3, 4]). Besides, many feature descriptors work using multiple resolution images [5], but they are also interesting improvements which tend to be more computationally efficient [6]. Nowadays a vast majority of research is devoted to area based matching algorithms where the main contributions are offered to cope with the three major problems: i) occlusion areas, i.e. areas near 3D depth discontinuities ii) textureless regions iii) a substantial computational time needed for high quality depth maps using global optimization [7].

A powerful alternative to passive stereo matching is the use of active illumination within a framework known as structured light (SL) [8]. SL assumes projection of one or more patterns on the 3D scene. The task of pattern(s) is to eventually provide camera image(s) of the 3D scene with the identifiable features on it (i.e. a unique code for matching) thus providing an efficient way to solve the correspondence problem either between cameras or a single camera and a source of illumination — commonly a video projector [9]. Considering various SL strategies it seems that the best assurance to solve afore mentioned problems, typical for passive stereo matching, is provided by some of the time multiplexing strategies [10]. A well defined time multiplexing strategy is a single phase shifting (PS) method which can provide a very accurate code [11]. In regard to other SL strategies, PS has the following main advantages. PS is a pixel based SL strategy, meaning that to compute a code for certain pixel it is only necessary to consider the gray level of that particular pixel only (no neighboring image pixels are considered). In addition, to compute a pixel code one only needs to use relatively simple image processing operations such as straightforward addition and multiplications of gray level values. Such feature being simple and thus attractive by itself, has a further potential for a parallel implementation of the image processing (code computation) algorithm. Finally PS can provide rather accurate subpixel code and in theory it is robust to albedo/color. Still, due to a periodic nature of projected patterns and phase shifting, a provided code is also periodic and said to be wrapped within module $2 \cdot \pi$. Thus, such code is still potentially ambiguous, unless slowly varying 3D surface is in the context. Typically multiple phase shifting approach or Gray code combined with a single phase shifting will allow unwrapping procedure and eventually unambiguous code ([8,10,12]). The biggest disadvantage in those cases is the need to project a fair amount of additional patterns in order to unwrap unambiguous code for straightforward matching. Therefore, it is certainly advantageous to provide an unambiguous SL code with as few patterns as possible and yet to assure its robustness for color/albedo, occlusions and sharp changes in 3D depth, along with a short computation. Actually, there are contributions which use very few patterns aimed at real time applications (including even 3D reconstruction of dynamic scenes), due

* Corresponding author.
E-mail address: tomislav.pribanic@fer.hr (T. Pribanic).

0030-4018/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.optcom.2011.10.045

to the development of a high speed pattern projection hardware and/or assumption about relatively slow moving objects ([13-15]).

Apart from the using a projector to provide a texture on the scene, based on the various published work on stereo matching ([8,16-19]) it seems there are no contributions which try to combine the best features of active and passive stereo matching. As it will be shown below, our method thanks to ambiguous, but period wise relatively an accurate phase shifting code for some pixel on first image, successfully extracts only a small amount of candidates to match against on the second image. Such approach simultaneously reduces all mentioned problems for a passive stereo matching. Once we have found the best match out of small amount of candidates, we have effectively bracketed our initial solution within a small image area and then we re-use an accurate phase code to refine our disparity estimates.

2. Brief overview of stereo matching, challenges and solutions

For a given image location on the first (left) image (x_L, y_L), stereo matching algorithms try to find disparity value d which will yield a correspondent image location on the second (right) image (x_R, y_R). If camera pair images are rectified [20], the following condition holds:

$$x_R = x_L - d \quad y_R = y_L \tag{1}$$

As observed in [7], a procedure to compute disparity d usually demands four steps: a) matching cost computation b) cost aggregation c) disparity computation (optimization) and d) disparity refinement. A common area (window) based costs assume squared or absolute differences between individual pixels (step a)) which is then summed (aggregated) across some window area (step b)) [16]. Alternatively and also popular, cost measures like normalized cross correlation and rank transform combines step a) and step b) [1]. For the local algorithms after step b), given some image point (x, y) we have for each disparity candidate (within considered range d_{min} and d_{max}) an aggregated cost C(x, y, d) (Fig. 1).

To resolve step c) local algorithms employ winner-take-all (WTA) strategy where the minimum cost directly points to the disparity computed value. On Fig. 1 point *A* represents solution according to WTA. However, point *A* is so called 'weak' minimum since there is clearly another point *B* which could have been the true solution instead, but due to noise it did not turn out to be. Moreover, the entire range of disparity values, marked as Δ on Fig. 1, is questionable and in practice any point from Δ could have been perhaps picked up by WTA also. Therefore, a true solution can be quite off from the correct one (*A* vs. *B*) and/or it could be hard to discriminate from many neighboring values (Δ interval) which have very much similar costs. Besides in the later case, a subpixel disparity computation, by approximating the cost function locally using a parabola, is clearly prevented. Mentioned problems on Fig. 1 are very common in practice due to several causes [3]: first, due to unavoidable photometric and projective distortions. Second, lack of texture or even repetitive texture. Third, points near the boundaries

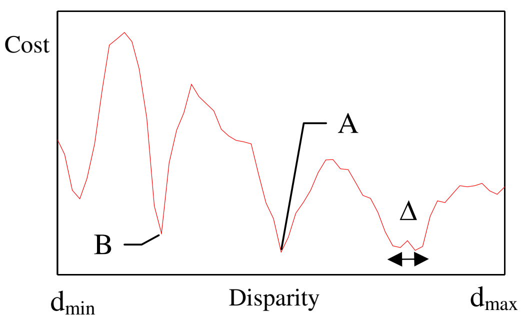

Fig. 1. Cost vs disparity. Point *B* and Δ interval are likely to cause problems in determining assumed point *A* as a correct solution.

are hard to identify and a fixed support window used for cost aggregation will include pixels not belonging to the same depth, but eventually the acquired cost could have a undesired low cost value. Forth, half-occluded points (point seen from one camera and not from the other) which theoretically should have a large minimum cost value, but due to similar texture in images and large disparity range considered will eventually end up with having small cost. To cope with the mentioned problems (sometimes as part of step d)) various solutions are proposed. For instance, employing more sophisticated cost functions, e.g. normalized cross correlation, were expected to give a more robust and discriminative cost values in the case of photometric distortion [16]. Shiftable windows, varying size windows, multiple windows were all aimed at determining the most appropriate aggregation support size, particularly for points near the depth border [18]. Proposing 3D support computation is an attempt to handle slanted surfaces [21]. Using unconstrained window shapes and/or associating different weights to window points, were additional ideas targeted to compute correct disparities [22]. Initial color segmentation should also help out detection of 3D object boundaries (i.e. problematic points nearby) but introducing a rather strong assumption that object depth segments will coincide with the object color segments [23]. In addition, different physical constraints were imposed, e.g. a reduced disparity range considered, ordering constraint (the relative orientation of two points on the one image should be the same for their correspondences on the other image), left-right consistency check are some ideas which should identify a large number of initially incorrectly assigned disparities [7].

In comparison to local algorithms, more sophisticated solutions were offered in the form of global optimization algorithms ([24,25]). These algorithms are less sensitive to initial disparity solutions and in principle more powerful, but usually require considerably more amount of memory storage, computational time etc. which makes a real-time application hardly conceivable. In fact, even real time applications based on local algorithms are quite constrained with respect to image resolution and disparity range considered [17].

3. Proposed method

Except for idea that explicitly restricts a disparity range (note that is not always either practical or possible), a common feature to all above mentioned proposed solutions is that they usually operate on a large set of candidate points. However, we were inspired by the preference from the early days where featured based stereo was prevailing. It was partly due to insufficient computer processing power and it was partly motivated by the fact that less point candidates to consider will significantly reduce the chance of wrong matches [1]. At the same time we were not comfortable with the fact of having a sparse matching solution, but we rather decided to retain a possibility of dense matching. Therefore we decided to mix two types of stereo matching, passive and active within a scope known as structured light. SL offers a large number of pattern projection strategies [8]. We have chosen a phase shifting largely due to its robustness to object albedo/color and the fact that every pixel is coded based only on its own gray level values, therefore not affected by any (occluded) neighborhood [10]. The phase shift method typically assumes a projection of periodic sinusoidal patterns *N* times, where between projections a sinusoidal pattern is shifted by an amount of φ_i, equally covering the entire period:

$$\varphi_i = \frac{2 \cdot \pi}{N} \cdot i \quad i = 0, 1, ..N-1 \tag{2}$$

It can be shown that for $N \geq 3$ and based solely on camera pixel detected gray level intensity I_i after each projection, it is possible to compute for every pixel a wrapped phase φ:

$$\varphi = \text{atan}\left(-\sum_{i=0}^{N-1} I_i \cdot \cos(\varphi_i), \sum_{i=0}^{N-1} I_i \cdot \sin(\varphi_i) \right) \tag{3}$$

where *a*tan is the four-quadrant inverse tangent function. Thus, PS assigns to every image pixel a unique code (either along horizontal or vertical coordinate) within a certain period, i.e. image portion. Fig. 2 shows an example of original camera image, its periodic PS counterpart and change of periodic PS code (wrapped value within $2\cdot\pi$) along one horizontal line. Therefore, for any point on the line we know its wrapped value and we know for fact that its corresponding point on the second image must have the same wrapped value. The only problem is that, similarly as more than one point on the original image have the same wrapped phase (in theory number of periods determines how many points along the horizontal direction have the same wrapped phase), there are also more than one point on the second image (on the corresponding epipolar line) which have (almost) the same wrapped phase. To resolve this ambiguity, given some point on the first image and its wrapped phase φ_1 we pick from the second image all those points having wrapped phase φ_2 satisfying:

$$\varphi_1 - \varepsilon < \varphi_2 < \varphi_1 + \varepsilon \tag{4}$$

where ε is an arbitrary set threshold due to the fact that on realistic noisy images it is very unlikely to find correspondent points having exactly the same wrapped phases. Extracting all points on the second image having φ_2 according to Eq. (4) yields eventually several group of points, each group belonging to a different period on PS image. We emphasize that the total number of such candidate points is rather small compared to the size typically used during a traditional passive stereo matching. On this set of points, relatively modest in size, we perform passive stereo matching using absolute differences as a cost measure and WTA approach. Then, finding the candidate point from the second image that is the most similar to the point from the first image, we know the period where our solution is. Following a traditional stereo matching we would normally stop here. Recall that is quite realistic that our found correspondent point is dubious one from the Δ interval as explained on Fig. 1. Luckily, PS is known for providing a very accurate code, i.e. image location. Therefore we interpolate wrapped phase curve from candidate points within a period where initial match was found, and subsequently we find with subpixel accuracy a point on the second image which has the same wrapped phase as the point from the left image. In fact, such interpolation in terms of accurate correspondences is always advantageous, not only in the case of Δ interval ambiguous matches (Fig. 1).

4. Evaluation and discussion

In order to facilitate as fair as possible comparison between our approach and various others, our evaluation methodology is analogous to the one adopted on the well known Middlebury Stereo Evaluation site [26], where four images sets are used for analysis: Tsukuba, Venus, Teddy, and Cones. We compare estimated disparities against ground true disparities for regions near depth discontinuities (*disc*; according to definition on [26] that includes half occluded points as well), nonoccluded regions (*nonoc*) and all regions (*all* = *nonoc U disc*). An estimated disparity is considered correct if it is within a ground truth disparity ±1. Our proposed approach includes the use of structured light, i.e. wrapped phase images acquired from PS (Fig. 2 b)). Unfortunately, those are not available on [26], however we have synthesized wrapped phase images ourselves from the ground truth data. We defined a projector as it were placed on the half of camera pair baseline and pointing in the same direction as camera's optical axis.

Stereo matching algorithms are quite known to have variable performance if relevant algorithm parameters are not tuned properly ([7,16]). One of the most sensitive ones is the window size, particularly for the local methods. On one hand, one is tempted to work with small windows to preserve object boundaries. On the other hand, to confidently measure a similarity between corresponding pixels neighborhoods we need to have sufficiently large areas to match. Basically, an optimal window size is hard to a priory specify for general scenery. It turns out that an optimal window size for

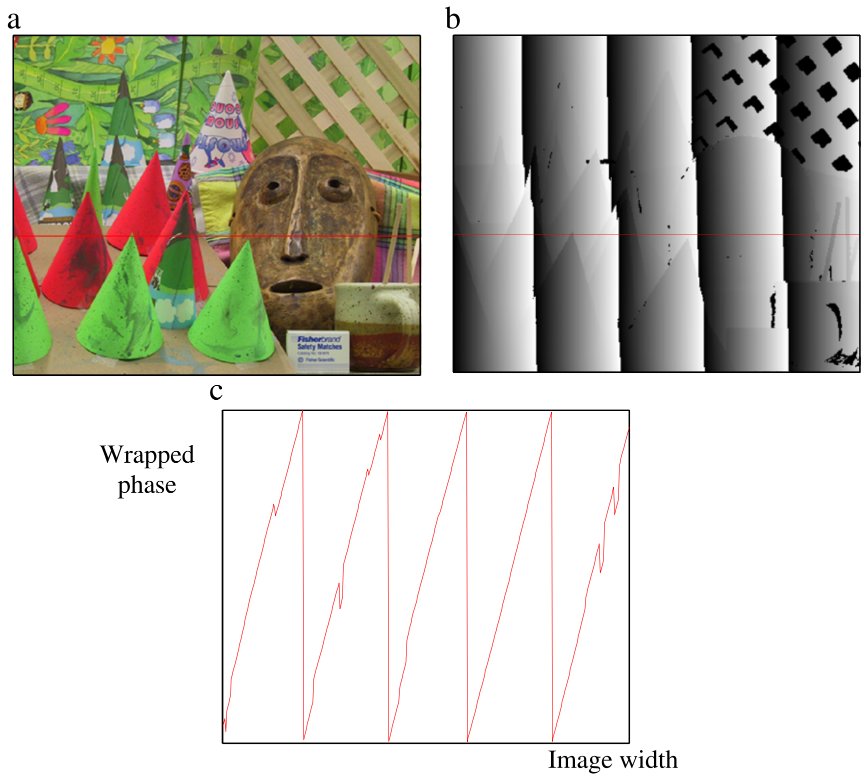

Fig. 2. a) Original 'cones' image b) Wrapped phase image acquired from PS c) change of wrapped phase for particular horizontal line marked on a) and b).

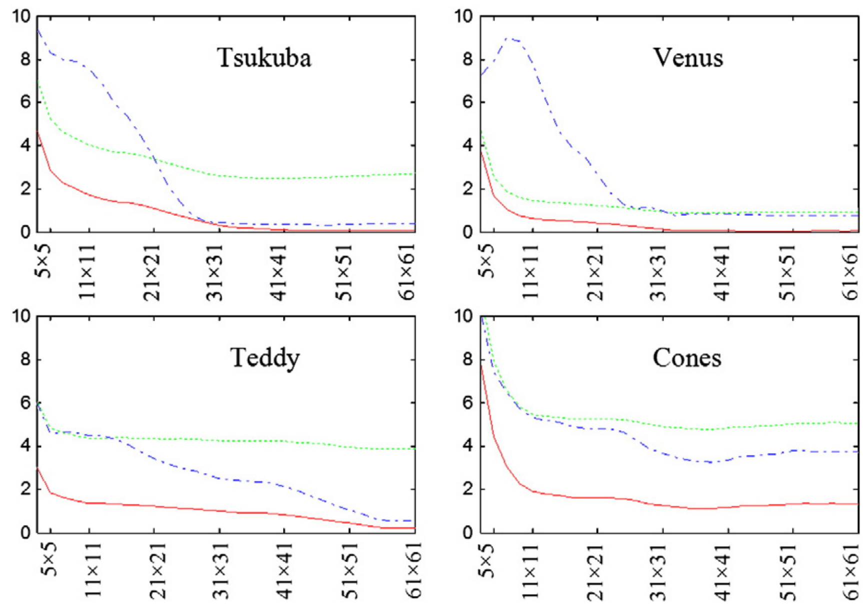

Fig. 3. Error rates with respect to various window sizes. Solid line: non occluded areas. Dash dotted line: near discontinuities areas. Dotted line: all areas (half occluded points included).

this evaluation set and in the case of fixed square window is around 5×5 (near depths discontinuities) and around 17×17 in nonoccluded areas [17]. Evidently, it is very hard to tune the system simultaneously for both nonoccluded and near depth discontinuities image areas, unless additional processing is undertaken which aims to detect areas near depth discontinuities. We tested our approach for various window sizes as well (Fig. 3). It seems that our approach favors larger windows. It is after a certain point quite robust with respect to a window size, both in case of nonoccluded and near depth discontinuities image areas, with no additional processing related to window size, position, shape and/or weights adjustments [17]. In the remainder of this document we use 31×31 window size.

One of the key aspects in our approach is the number of periods p used for periodical sinusoidal pattern to carry out PS method. In the trivial case where we have single period no unwrapping procedure is required since there is no ambiguity caused by periodicity. However, within the context of SL it is known that increase in the number of periods significantly improves the overall accuracy ([8,10]). Within a context of this work more periods means more candidate points that will satisfy condition (4), i.e. more points to be tested for matching using passive stereo matching. Thus, starting with the modest value for p and then gradually increasing it, we would expect to experience an improvement in accuracy up to certain point where the effect of relatively large total number of candidates points would undo a beneficial effect of an accurate PS code. Actually, even excessive values for p alone would start decreasing accuracy due to the fact that then it is no longer possible to produce reliable wrapped phases [10]. Table 1 shows error rates (percentage) for our method in the case of various periods p considered while ε was kept fixed at 0.020. It appears that optimum value for p is around 8, in particular for generally considered more complex scenes, Teddy and Cones [18].

For our method to show its best, the number of candidates that will satisfy (4) should be kept to minimum. More precisely, from every period several points should be chosen for a reliable initial passive stereo matching and for a subsequent (optional) subpixel disparity interpolation (once the initial match is bracketed within several neighbor candidates). In order to do that we need to tune the parameter ε from (4). A detailed analytical analysis of ε dependence on a system noise is out of the scope of this work. However, our experiments show that, at least for given data sets, keeping ε within range [0.015, 0.030] shows about equally good performance (Table 2).

Image evaluation set used in this work has been extensively used by numerous researches where results for comparisons are not only available in the corresponding papers, but also the performance result of more than 100 methods is readily available on a Middlebury stereo evaluation. Out of some 100 methods, Table 3 replicates performance scores for the best ranked methods (according to average rank; see table on [26] for more details). We emphasize that basically all those methods include global optimization and/or other significant data processing such as color segmentation. At the same time we

Table 1
Performance of the proposed approach in terms of accuracy for a various number of periods. The window size was 31×31 and parameter ε was 0.02.

Number of periods	Tskuba			Venus			Teddy			Cones		
	Nonoc	All	Disc	Nonoc	All	Disc	Nonoc	All	Disc	Nonoc	All	Disc
5	0.30	1.88	0.70	0.43	1.21	1.49	2.01	5.15	4.12	1.90	5.65	4.39
6	0.18	1.89	0.44	0.35	1.15	1.52	2.06	5.20	3.87	2.20	5.91	5.45
7	0.24	1.95	0.74	0.32	1.15	1.65	2.32	5.48	4.36	2.33	6.00	5.99
8	0.21	1.89	0.43	0.22	1.03	1.04	1.64	4.84	3.14	1.74	5.39	4.57
9	0.41	2.73	0.60	0.27	1.08	1.22	1.25	4.48	3.03	1.71	5.41	4.94
10	0.42	2.76	0.87	0.28	1.09	1.21	1.41	4.66	3.44	2.62	6.35	5.56
11	0.43	2.75	0.89	0.36	1.20	1.49	1.43	4.70	3.70	3.04	6.87	6.28
12	0.44	2.82	0.75	0.48	1.34	2.07	1.65	4.96	4.14	3.10	7.20	6.39
13	0.48	2.80	0.85	0.61	1.51	2.65	1.99	5.31	5.04	3.80	7.94	8.13
14	0.53	2.83	1.03	0.66	1.57	2.76	2.70	6.05	6.56	4.09	8.41	8.90
15	0.55	2.89	0.86	0.81	1.76	3.13	5.14	8.35	7.45	3.66	8.00	7.67

Table 2
Performance of the proposed approach in terms of accuracy for a various values of parameter ε. The number of periods was set to p = 8 and window size was 31 × 31.

ε	Tskuba			Venus			Teddy			Cones		
	Nonoc	All	Disc	Nonoc	All	Disc	Nonoc	All	Disc	Nonoc	All	Disc
0.01	0.92	3.43	0.68	0.30	1.29	2.01	1.08	4.59	2.91	1.85	5.27	3.67
0.015	0.38	2.93	0.53	0.17	1.06	1.47	0.99	4.38	2.69	1.67	5.09	3.62
0.02	0.32	2.63	0.46	0.13	0.94	0.97	1.00	4.26	2.52	1.28	4.91	3.44
0.025	0.21	1.89	0.43	0.22	1.03	1.04	1.64	4.84	3.14	1.25	5.39	4.57
0.03	0.27	1.95	0.58	0.37	1.19	1.38	2.16	5.32	4.23	2.12	5.80	5.35
0.035	0.38	2.14	0.72	0.41	1.20	1.36	2.28	5.43	4.42	2.34	6.11	5.78
0.04	0.44	2.44	0.82	0.47	1.25	1.52	2.34	5.49	4.58	2.52	6.35	6.13
0.05	0.75	2.85	1.36	0.64	1.39	2.07	2.87	6.00	5.64	3.07	6.95	7.27

Table 3
Performance in terms of accuracy for top ranking methods, as evaluated by Middlebury stereo site [26], which employ a significant additional processing in the form of (global) optimization and/or color segmentation.

Method	Tskuba			Venus			Teddy			Cones		
	Nonoc	All	Disc	Nonoc	All	Disc	Nonoc	All	Disc	Nonoc	All	Disc
ADCensus [31]	1.07	1.48	5.73	0.09	0.25	1.15	4.10	6.22	10.9	2.42	7.25	6.95
AdaptingBP [32]	1.11	1.37	5.79	0.10	0.21	1.44	4.22	7.06	11.8	2.48	7.92	7.32
CoopRegion [33]	0.87	1.16	4.61	0.11	0.21	1.54	5.16	8.31	13.0	2.79	7.18	8.01
DoubleBP [34]	0.88	1.29	4.76	0.13	0.45	1.87	3.53	8.30	9.63	2.90	8.78	7.79

recall that our method employs straightforward WTA approach. However due to inclusion of PS we are able to produce results which, for the large range of parameters (Table 1), are quite competitive with the state-of-the-art in stereo matching (Table 3). For completeness, we also compare our method with the best results for local methods as reported in [18] (Table 4). In that case, our proposed approach clearly outperforms other local methods, especially in the cases of Teddy and Cones which are regarded as the most demanding and realistic scenes ([16,18]).

For a qualitative comparison we provide Fig. 4, where the appearance of computed disparity maps can be compared with the ground truth. It can be noticed that basically the most discrepancies from the ground truth values takes place near depth discontinuity boundaries, i.e. in occluded areas. That's expected since our method does not use an explicit mechanism for an occluded area detection.

To further appreciate shown results we note that very often disparity range to be considered is defined in advance [16]. That is usually done in order to speed up processing, save memory requirements, and largely to prevent errors in matching due to a large number of candidates considered. For example, sometimes a rationale to determine a half-occluded point is relatively low similarity measure value [21]. Clearly, even for occluded point matching against lot of candidates on the second image (i.e. considered disparity range >>) with a variable texture it is possible to acquire a highly similar measure. Our approach does not restrict any disparity range in advance. In more detail for every point on the left image having abscissa coordinate x_L we initially consider the entire theoretically possible range of points on the right image which have abscissa value $x_R < x_L$ and which of course satisfy condition (4), as explained earlier.

Table 4
Performance in terms of accuracy for top ranking methods, as reported in [18], which use plain WTA framework.

Method	Tskuba		Venus		Teddy		Cones	
	Nonoc	Disc	Nonocc	Disc	Nonoc	Disc	Nonoc	Disc
Segment support [28]	2.28	7.50	1.21	5.88	10.99	22.01	5.42	11.83
Adaptive weight [22]	4.66	8.25	4.61	13.30	12.70	22.40	5.50	11.90
VariableWindows [29]	4.10	10.79	10.66	9.94	13.93	25.53	7.24	13.86
Reliability [30]	5.14	18.31	3.86	11.51	16.96	30.62	13.52	21.55
ShiftableWindows [24]	6.53	21.80	6.60	13.54	16.16	30.19	9.55	22.99

We have designed our algorithm to be as simple as possible, attractable for a software parallel implementation and thus in large part aiming at real time applications [17] (Currently our code is written in Matlab and timing evaluation is left for future work). However, we point out that we did not propose any additional disparity refinement, frequently applied by other algorithms (e.g. invalidation of small disparity segments, median-filtering clean up, left-right consistency check [1]), since we believe our results are already rather appealing within a context of passive stereo matching. Of course, strictly speaking our method cannot be regarded as a typical stereo matching method. In fact, one may classify it as a structured light method (but using in this case uncalibrated projector though), however with embedded a passive stereo matching strategy, to circumvent a rather challenging task of phase unwrapping (typically solved through an additional patterns projection and processing). In that sense we feel it would be appropriate to compare our method against structured light methods, and that is one of our further works.

Actually, one may argue that we evaluated our method on the actual images only when it comes down to the passive stereo matching part of evaluation, whereas the structured light data (i.e. wrapped phase images) are synthesized as explained above. While that is true, we note that our primary objective in this work was to compare our combined method with a typical passive stereo approaches which performance on the popular data sets is available on [26]. Nevertheless, we feel that generally drawn conclusion about advantages of our proposed method will still hold even in the case of noisy structured light data. In fact, in our future work we do intend to provide a comparison involving the real noise structured light scanning as well. Besides, we note that the ground truth data for many evaluation sets available on Middlebury Stereo site are actually computed using structured light as explained in [27].

5. Conclusion

In this work we have proposed a stereo computation method which initially borrows SL strategy, based on single phase shifting approach. However, the unwrapping of unique codes is avoided using traditional area based techniques for passive stereo matching. In that sense our proposed method can be also viewed as a contribution within a context of phase unwrapping.

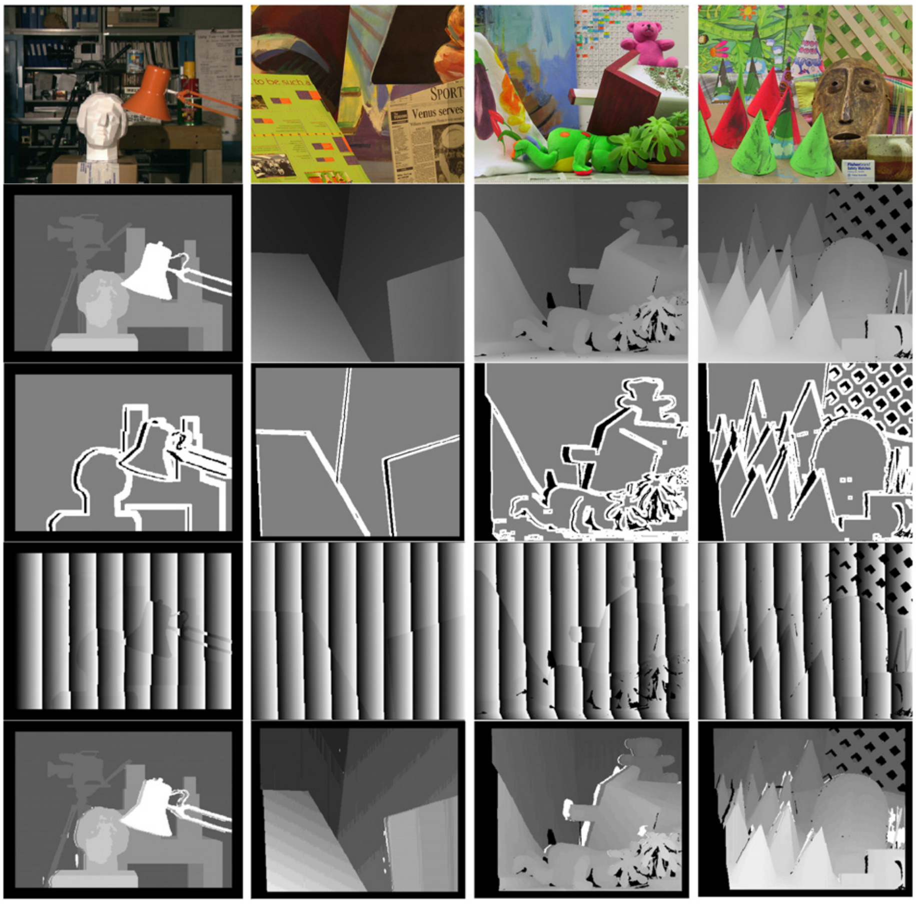

Fig. 4. First row: left view images: Tsukuba, Venus, Teddy and Cones. Second row: ground truth disparities. Third row: regions near depth discontinuities (white), occluded and border regions (black), and other regions (gray). Forth row: wrapped phase images. Fifth row: proposed method disparities. Quantitative error results for this example are provided in Table 1, a row where the number of periods is p = 8.

The use of an accurate phase shifting code allowed us to considerably decrease the candidate set of points for matching. Once the most similar match was found using area based approach we re-use the accurate wrapped code to refine initial disparity value. In terms of cost used for a stereo matching phase we employed a straightforward fixed rectangular window, which size is found to be quite robust both for nonoccluded and near depth discontinuities image areas. Besides we do not put any restrictions on a disparity range considered, we do not employ any additional processing for false match detection. Thus, our method is extremely simple to implement and in that sense very promising for real time applications. Shown comparison results demonstrate that we can produce accuracy comparable with the state-of-the art methods in stereo matching.

References

[1] R. Szeliski, Computer Vision: Algorithms and Applications, Springer, 2011.
[2] X. Hu, N. Ahuja, IEEE Transactions on Pattern Analysis and Machine Intelligence 16 (10) (2002) 1041.
[3] A. Fusiello, V. Roberto, E. Trucco, International Journal of Pattern Recognition and Artificial Intelligence 14 (8) (2000) 1053.
[4] A. Woodward, D. An, G. Gimel'farb, P. Delmas, IEEE International Conference on Multimedia and Expo, 2006, p. 2057.
[5] D. Lowe, International Journal of Computer Vision 20 (2) (2004) 91.
[6] E. Tola, V. Lepetit, P. Fua, IEEE International Conference on Computer Vision and Pattern Recognition, 2008, p. 1.
[7] D. Scharstein, R. Szeliski, International Journal of Computer Vision 47 (1/2/3) (2002) 7.
[8] J. Salvi, S. Fernandez, T. Pribanic, X. Llado, Pattern Recognition 43 (8) (2010) 2666.
[9] R. Klette, K. Schluns, A. Koschan, Computer Vision: Three-Dimensional Dana from Images, Springer Verlag, 1998.
[10] T. Pribanic, S. Mrvos, J. Salvi, Image and Vision Computing 28 (8) (2010) 1255.
[11] J. Huntley, H. Saldner, Measurement Science and Technology 8 (9) (1997) 986.
[12] G. Sansoni, A. Patrioli, F. Docchio, The Review of Scientific Instruments 74 (4) (2003) 2593.
[13] A. Wong, P. Niu, X. He, Computer Vision and Image Understanding 98 (3) (2005) 398.
[14] P. Huang, S. Zhang, F. Chiang, et al., Optical Engineering 44 (12) (2005) 142.
[15] O.A. Hall-Holt, S. Rusinkiewicz, ICCV, 2001, p. 359.
[16] H. Hirschmüller, D. Scharstein, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (9) (2009) 1582.
[17] M. Gong, R. Yang, L. Wang, M. Gong, International Journal of Computer Vision 75 (2) (2007) 283.
[18] F. Tombari, S. Mattoccia, L. Di Stefano, E. Addimanda, IEEE International Conference on Computer Vision and Pattern Recognition, 2008, p. 24.
[19] S. Seitz, B. Curless, J. Diebel, D. Scharstein, R. Szeliski, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2006, p. 519.
[20] A. Fusiello, E. Trucco, A. Verri, Machine Vision and Applications 12 (1) (2000) 16.
[21] C.L. Zitnick, T. Kanade, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (7) (2000) 675.
[22] K.-J. Yoon, I.S. Kweon, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (4) (2006) 650.
[23] L. Wang, S.B. Kang, H.Y. Shum, G. Xu, Proc. Asian Conference on Computer Vision, 2004, p. 366.
[24] A.F. Bobick, S.S. Intille, International Journal of Computer Vision 33 (3) (1999) 181.
[25] V. Kolmogorov, R. Zabih, 8th IEEE conference on Computer Vision, 2001, p. 508.
[26] Access date: June, http://vision.middlebury.edu/stereo/2011.
[27] D. Scharstein, R. Szeliski, *IEEE* International Conference on Computer Vision and Pattern Recognition, 2003, p. 195.
[28] F. Tombari, S. Mattoccia, L. Di Stefano, Proc. Pacific-Rim Symposium on Image and Video Technology, 2007.
[29] O. Veksler, Proc. Conf. Computer Vision and Pattern Recognition, 2003, p. 556.
[30] S. Kang, R. Szeliski, J. Chai, Proc. Conf. Computer Vision and Pattern Recognition, 2001, p. 103.
[31] X. Mei, X. Sun, M. Zhou, S. Jiao, H. Wang, X. Zhang, GPUCV, 2011, In conjunction with ICCV 2011).
[32] A. Klaus, M. Sormann, K. Karner, 18th International Conference on Pattern Recognition, 2006, p. 15.
[33] Z. Wang, Z. Zheng, IEEE Conference on Computer Vision and Pattern Recognition, 2008, p. 1.
[34] Q. Yang, L. Wang, R. Yang, H. Stewénius, D. Nistér, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (3) (2009) 492.