High-Resolution Stereo Datasets with Subpixel-Accurate Ground Truth

Daniel Scharstein[1], Heiko Hirschmüller[2], York Kitajima[1],
Greg Krathwohl[1], Nera Nešić[3], Xi Wang[1], and Porter Westling[4]

[1] Middlebury College, Vermont, USA
[2] German Aerospace Center, Oberpfaffenhofen, Germany
[3] Reykjavik University, Iceland
[4] LiveRamp, San Francisco, USA

Abstract. We present a structured lighting system for creating high-resolution stereo datasets of static indoor scenes with highly accurate ground-truth disparities. The system includes novel techniques for efficient 2D subpixel correspondence search and self-calibration of cameras and projectors with modeling of lens distortion. Combining disparity estimates from multiple projector positions we are able to achieve a disparity accuracy of 0.2 pixels on most observed surfaces, including in half-occluded regions. We contribute 33 new 6-megapixel datasets obtained with our system and demonstrate that they present new challenges for the next generation of stereo algorithms.

1   Introduction

Stereo vision is one of the most heavily researched topics in computer vision [5, 17, 18, 20, 28], and much of the progress over the last decade has been driven by the availability of standard test images and benchmarks [7, 14, 27, 28, 30, 31]. Current datasets, however, are limited in resolution, scene complexity, realism, and accuracy of ground truth. In order to generate challenges for the next generation of stereo algorithms, new datasets are urgently needed.

In this paper we present a new system for generating high-resolution two-view datasets using structured lighting, extending and improving the method by Scharstein and Szeliski [29]. We contribute 33 new 6-megapixel datasets of indoor scenes with subpixel-accurate ground truth. A central insight driving our work is that high-resolution stereo images require a new level of calibration accuracy that is difficult to obtain using standard calibration methods. Our datasets are available at `http://vision.middlebury.edu/stereo/data/2014/`.

Novel features of our system and our new datasets include the following: (1) a portable stereo rig with two DSLR cameras and two point-and-shoot cameras, allowing capturing of scenes outside the laboratory and simulating the diversity of Internet images; (2) accurate floating-point disparities via robust interpolation of lighting codes and efficient 2D subpixel correspondence search; (3) improved calibration and rectification accuracy via bundle adjustment; (4) improved self-calibration of the structured light projectors, including lens distortion, via robust

|			
(a)	(b)	(c)	(d)
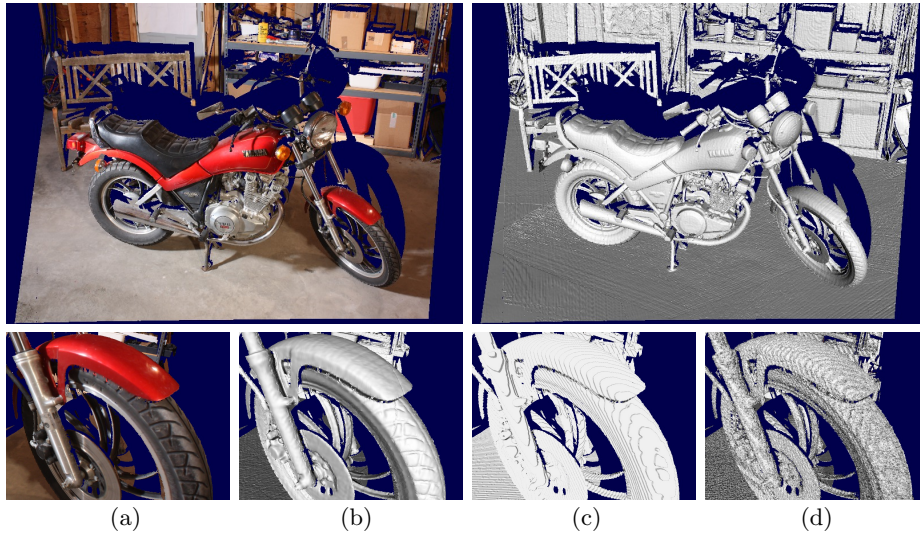
Fig. 1. Color and shaded renderings of a depth map produced by our system; (a), (b) detail views; (c) resulting surface if disparities are rounded to integers; (d) resulting surface without our novel subpixel and self-calibration components.

model selection; and (5) additional "imperfect" versions of all datasets exhibiting realistic rectification errors with accurate 2D ground-truth disparities. The resulting system is able to produce new stereo datasets with significantly higher quality than existing datasets; see Figs. 1 and 2 for examples.

We contribute our new datasets to the community with the aim of providing a new challenge for stereo vision researchers. Each dataset consists of input images taken under multiple exposures and multiple ambient illuminations with and without a mirror sphere present to capture the lighting conditions. We provide each dataset with both "perfect" and realistic "imperfect" rectification, with accurate 1D and 2D floating-point disparities, respectively.

2   Related work

Recovery of 3D geometry using structured light dates back more than 40 years [3, 4, 25, 32]; see Salvi et al. [26] for a recent survey. Applications range from cultural heritage [21] to interactive 3D modeling [19]. Generally, 3D acquisition employing active or passive methods is a mature field with companies offering turnkey solutions [1, 2]. However, for the goal of producing high-resolution stereo datasets, it is difficult to precisely register 3D models obtained using a separate scanner with the input images. Existing two-view [7] and multiview [30, 31] stereo datasets for which the ground truth was obtained with a laser scanner typically suffer from (1) limited ground-truth resolution and coverage; and (2) limited precision of the calibration relating ground-truth model and input images. To
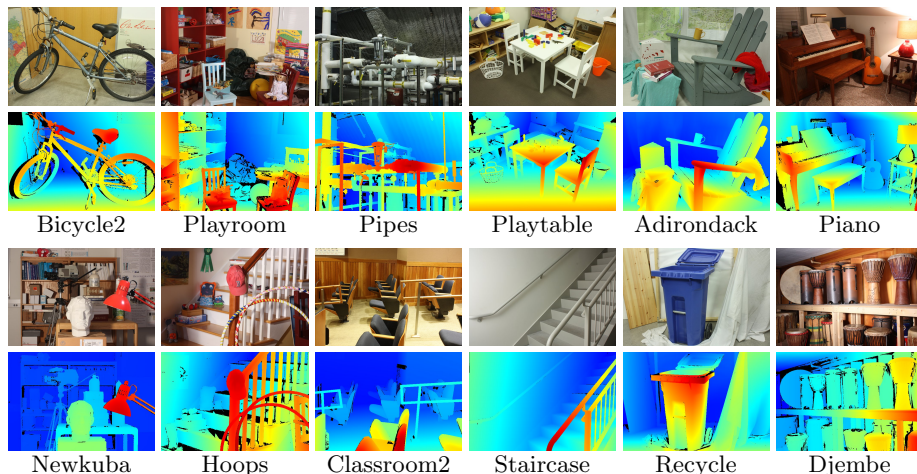
Fig. 2. Left views and disparity maps for a subset of our new datasets, including a restaging of the Tsukuba "head and lamp" scene [24]. Disparity ranges are between 200 and 800 pixels at a resolution of 6 megapixels.

address the second problem Seitz et al. [30] align each submitted model via ICP with the ground-truth model before the geometry is evaluated, while Geiger et al. [7] recently re-estimated the calibration from the current set of submissions.

Establishing ground-truth disparities from the input views directly avoids the calibration problem and can be done via unstructured light [1, 6, 34], but only yields disparities for nonoccluded scene points visible in both input images. Scharstein and Szeliski [29] pioneered the idea of self-calibrating the structured light sources from the initial nonoccluded *view disparities*, which yields registered *illumination disparities* in half-occluded regions as well. We extend this idea in this paper and also model projector lens distortion; in addition, we significantly improve the rectification accuracy using the initial correspondences.

Gupta and Nayar [10] achieve subpixel precision using a small number of sinusoidal patterns, but require estimating scene albedo, which is sensitive to noise. In contrast, we use a large number of binary patterns under multiple exposures and achieve subpixel precision via robust interpolation. We employ the maximum min-stripe-width Gray codes by Gupta et al. [9] for improved robustness in the presence of interreflections and defocus.

Overall, we argue that the approach of [29] is still the best method for obtaining highly accurate ground truth for stereo datasets of static scenes. The contribution of this paper is to push this approach to a new level of accuracy. In addition, by providing datasets with both perfect and imperfect rectification, we enable studying the effect of rectification errors on stereo algorithms [13]. In Section 4 we show that such errors can strongly affect the performance of high-resolution stereo matching, and we hope that our datasets will inspire novel work on stereo self-calibration [11].
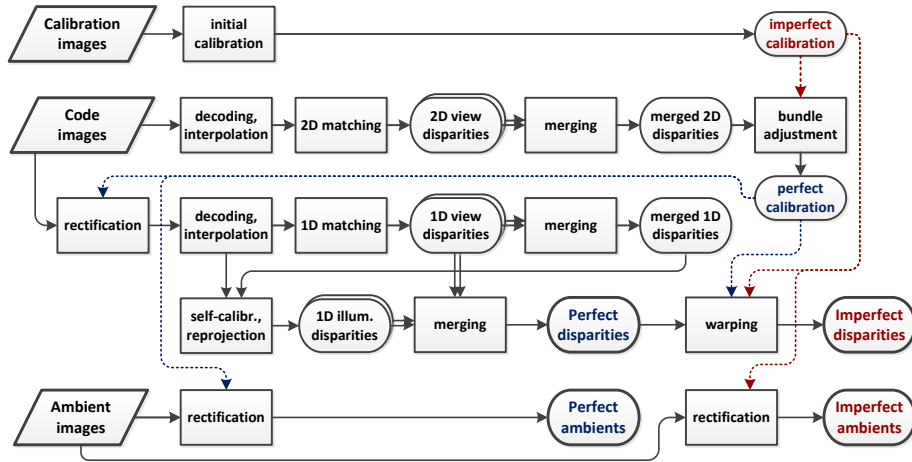
Fig. 3. Diagram of the overall processing pipeline.

3 Processing pipeline

The overall workflow of our 3D reconstruction pipeline is illustrated in Fig. 3.

The inputs to our system are (1) calibration images of a standard checker-board calibration target; (2) code images taken under structured lighting from different projector positions; and (3) "ambient" input images taken under different lighting conditions. The main processing steps (rows 2–4 in Fig. 3) involve the code images taken with the two DSLR cameras.

First, the original (unrectified) code images from each projector are thresholded, decoded, and interpolated, yielding floating-point coordinates of the projector pixel illuminating the scene. These values are used as unique identifiers to establish correspondences between the two input views, resulting in subpixel-accurate *2D view disparities*, which are used in a bundle-adjustment step to refine the initial "imperfect" calibration. The processing then starts over, taking rectified images as input and producing *1D view disparities* (row 3 in the diagram). The merged disparities are used to self-calibrate each projector (row 4), from which *1D illumination disparities* are derived. All sets of view and illumination disparities are merged into the final "perfect" disparities, which are then warped into the imperfect rectification. Corresponding sets of ambient images are produced by rectifying with both calibrations (row 5). We next discuss the individual steps of the processing pipeline in detail.

Image acquisition: To capture natural scenes outside the laboratory we employ a portable stereo rig (Fig. 4) with two Canon DSLR cameras (EOS 450D with 18–55 mm lens) in medium resolution (6 MP) mode. The cameras are mounted on a horizontal optical rail with variable baseline from 140 mm to 400 mm. We optionally include two point-and-shoot cameras (Canon PowerShot A495, A800, A3000, or SD780), in order to simulate the variability of Internet images. We

Fig. 4. Top: Portable stereo rig with 2 DSLRs and 2 consumer cameras, and painting of glossy scene. Bottom: Ambient views under different lighting conditions and disparities.

control the main cameras using DSLR Remote Pro, and trigger the PowerShots with CHDK firmware via their USB power cables.

For each scene we take calibration and ambient images with all cameras, then code images with the two DSLRs only. We typically use 4 different ambient lighting conditions and 8 different exposures. We take a second set of all images with a mirror sphere added to the scene to allow recovering the approximate lighting conditions. We store the original images both in JPG and RAW, and do all subsequent processing in lossless formats.

For the code images we project a series of binary patterns that uniquely encode each pixel location (u, v) of the projector. We use ViewSonic $1024{\times}768$ DLP projectors, thus 10 patterns are sufficient in each of u and v. Instead of standard Gray codes [29] we use maximum min-stripe-width Gray codes [9] for increased robustness to interreflections and defocus. Following [29, 9], we project each pattern and its inverse, and take all code images under 3 exposures (typically 0.1s, 0.5s, and 1.0s) in order to handle a wide range of albedos. We illuminate the scene from $P{=}4\ldots 18$ different projector locations, depending on the complexity of the scene. This allows us to focus the projectors on different scene depths and to minimize shadowed areas.

Our technique allows precise 3D capture of static indoor scenes. If the room can be darkened, we are able to handle very low albedos via long exposures. In the Motorcycle scene shown in Fig. 4 we painted high-gloss surfaces with a clay solution after acquiring the ambient images in order to recover the geometry of even truly reflective surfaces. This is the only scene where we applied paint; most of the other scenes contain semi-glossy surfaces which our system can recover, unless they are close to parallel to the viewing direction.

Decoding and interpolation: Given the 120 code images from one projector as input, we compute the signed difference between each pattern and its inverse for the 3 color bands, and select at each pixel the exposure yielding the largest absolute average difference $|d|$. We output "1" if $d \geq t_d$, "0" if $d \leq -t_d$, and "unknown" otherwise (we use $t_d{=}16$). We then concatenate the individual bits

initial values   hole-filled

u code values   x-interpolated   y-interpolated
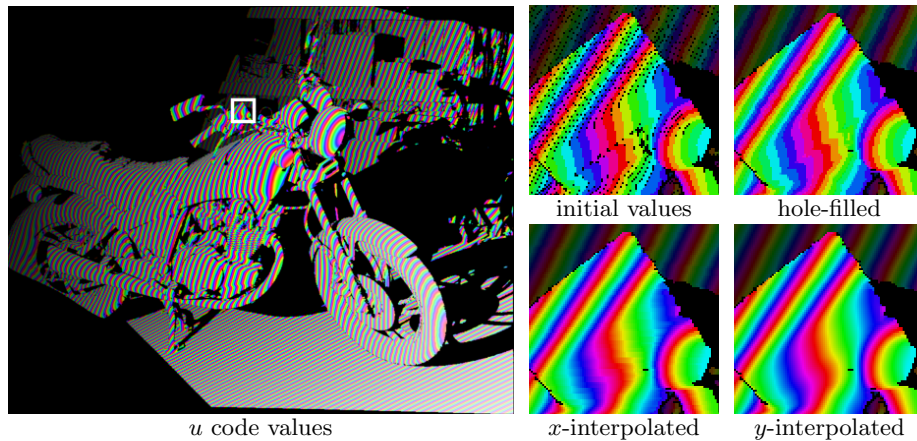
Fig. 5. Left: Recovered u codes for one of 12 projector positions, visualized using a helix in HSV color space. Right: Zoomed views illustrating the processing of code values.

and decode them into integer code pairs (u, v). Since the cameras have higher resolution than the projectors, a single projector pixel typically spans 2–4 camera pixels. On smooth surfaces, we expect the code values to increase monotonically in the prominent code direction (i.e., rightwards for u and downwards for v). This knowledge allows smart hole-filling and robust interpolation without blurring across depth boundaries. The process is illustrated in Fig. 5 and involves filling holes less than 6 pixels wide whose border values differ by no more than 2 in the code direction, followed by interpolation by convolution with a bilinear ("tent") kernel with radius 7 in both x and y directions. Only values that fit the expected monotonic ramp are included, the other values are extrapolated. The resulting floating-point u and v code images form the input for the subsequent stages.

Fast 2D correspondence search: In order to establish view disparities, we need to find for each pixel the most similar code pair in the other image. Given the initial unrectified images, we have no prior knowledge of the search range. A naive search would take quadratic time in the number of pixels. Instead, we developed a linear algorithm that precomputes the bounding boxes for each integer (u, v) code pair appearing in the target image. Except at depth boundaries, each of these bounding boxes only spans a few pixels. We can find the closest matching code value in expected constant time by restricting the search to the merged bounding boxes of the rounded code values and its 8 neighbors.

Once the pixel with the closest code value has been located, we derive a 2D subpixel disparity estimate by fitting planes to the u and v code values of the pixels surrounding the target code, respectively. If the fitted planes have poor residuals (e.g., if the planes straddle a depth boundary), no subpixel correction is attempted. Otherwise, the subpixel offset is determined using a least-squares fit of the source values to the u and v planes in the target image.
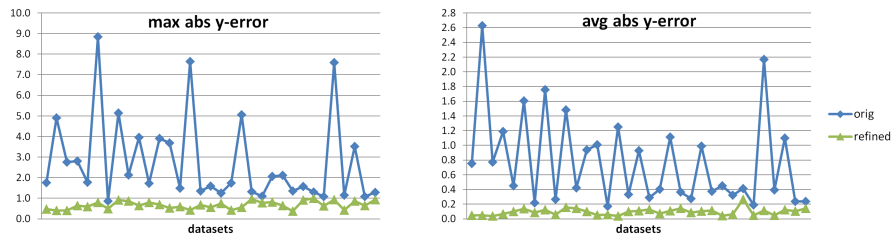
Fig. 6. Reduction of maximum (left) and average (right) absolute vertical rectification errors through our novel calibration refinement technique over all 33 datasets.

Filtering and merging: Given $2P$ disparity maps from 2 views and P projectors, we first cross-check each pair of disparity maps by requiring that left and right disparities agree to within 0.5 pixels. During 1D processing we also apply a robust 3×3 smoothing filter, remove isolated disparity components of less than 20 pixels, fill holes of up to 200 pixels if the surrounding disparities fit a planar model, and cross-check again. We then merge each set of P disparity maps, requiring at least two estimates per pixel, average those within 1.0 of the median value, and cross-check the final pair of merged disparities.

Calibration refinement: In our experience, the stereo rectification resulting from a standard calibration procedure (we use OpenCV) has limited accuracy. At a resolution of 6 MP residual vertical disparities of several pixels are common, especially in image corners. Sources of calibration errors include (1) incorrect lens distortion estimation if the calibration target never occupies a large fraction of the image; and (2) errors relating tilt angle and vertical camera position if the calibration target does not adequately sample the depth range of the scene. We have developed a novel method for correcting such errors via bundle adjustment using the precise 2D correspondences recovered from structured lighting.

We first sample these 2D disparities with a stride of 10 pixels and remove outliers by requiring that at least 50% of the pixels in each 9×9 window fit a planar disparity model. We then reproject the resulting points back into 3D using the initial calibration parameters, and perform sparse bundle adjustment using SBA [23] to minimize the residual y-disparities. Since we are only working on image correspondences and do not have any constraints on the 3D reconstruction, we can only (re-)calibrate the cameras up to an unknown perspective transformation [12]. In order to avoid arbitrary perspective distortions, we keep all parameters of camera 0 fixed and only optimize for lens distortion. For camera 1 we optimize lens distortion parameters, joint focal length $f_{1x} = f_{1y}$, and c_{1y} of the principal point, but keep c_{1x} fixed, because it does not affect y-disparities. We also optimize the extrinsics while keeping the length of the baseline constant. We tried lens distortion models of up to four parameters (radial κ_1, κ_2; tangential p_1, p_2), and found that the 2-parameter model without tangential distortion is sufficiently accurate for the lenses we employ.

Fig. 6 plots the maximum and average absolute residual y-disparities for our datasets, rectified with the original "imperfect" and the refined "perfect" cali-

bration. Averaged over all datasets, the maximum absolute y-error is reduced by a factor of 4.1 (from 2.76 to 0.67) while the overall average absolute error is reduced by a factor of 8.1 (from 0.77 to to 0.096). Notably, the maximum absolute error in the refined rectification never exceeds 1.0 pixels. This shows that we can achieve excellent rectification even with a simple 2-parameter lens-distortion model, if accurate correspondences are available. Recall from Fig. 3 that we next match the rectified code images; the resulting high-confidence 1D view disparities are used to derive *illumination disparities* as explained next.

Self-calibration of projectors: So far we have only utilized the interpolated code values (u, v) as unique "scene color" allowing us to establish unambiguous subpixel correspondences between the two cameras. But since (u, v) also encodes the pixel location in the projector's illumination pattern, we get additional correspondences for each projector-camera pair. While traditional structured lighting systems require an external calibration of the projector, we follow [29] and self-calibrate the projectors in terms of the already recovered view disparities, which we can interpret as projective depth. This amounts to recovering the projection matrices M_{pc} relating projector p with camera c: $\mathbf{u}_p \cong M_{pc}\mathbf{x}_c$, where $\mathbf{x}_c = [x\,y\,d\,1]^T$ is a (homogeneous) 3D point with disparity d in camera c, and $\mathbf{u}_p = [u\,v\,1]^T$ is the corresponding image location in projector p. As in [29] we can solve for M_{pc} via robust least squares and then derive new d estimates (illumination disparities) from the code values, including in half-occluded regions. We have found, however, that this linear model often yields high residuals on the order of several pixels since projector lens distortion is not modeled.

We therefore employ a novel nonlinear refinement, using the linear solution as start value, where we estimate up to six additional parameters: the distortion center (c_u, c_v), radial distortion parameters κ_1, κ_2, and tangential distortion parameters p_1, p_2. We use the Levenberg-Marquardt (LM) method [22] to minimize the squared reprojection error in u and v, and then invert the estimated nonlinear projection model (also via LM) to translate the code values into illumination disparities. One complication is that for some projector-camera combinations the lens distortion cannot be stably estimated, for instance, if only a small part of the projection pattern is visible. In such cases the disparities derived from the inverted projection model can be dramatically wrong. We solve this problem using a model-selection approach: we compute the illumination disparities for 0, 3, 4, and 6-parameter models, and select the one yielding the smallest residuals. Here 0 means linear; 3 includes c_u, c_v, κ_1; 4 includes κ_2; and 6 includes p_1, p_2. Using this technique we reduce the average absolute error from 0.47 to 0.26; the average reduction per dataset is by a factor of 2.1. More importantly, we significantly reduce the "bad" residuals greater than 1.0 from 7.3% to 0.75%.

Merging and accuracy estimates: In the final merging step, we combine all $2P$ view and illumination disparity maps per camera, again using robust averaging. We relax the cross-checking criterion to allow half-occluded regions behind matched surfaces. For each pixel we not only store its final floating-point disparity d, but also the number of disparity samples n and their sample standard

deviation s. While we do not have an absolute measure of reconstruction accuracy, we can use s as a quality measure, given that the disparity estimates are derived from separately decoded projection patterns. For the Motorcycle scene with $P = 12$ projector positions, the average number of samples per pixel is $\bar{n} = 5.7$ and the average sample standard deviation at pixels with $n \geq 2$ is $\bar{s} = 0.16$. Averaged over all datasets, the values are $\bar{n} = 7.7$ and $\bar{s} = 0.20$. A qualitative impression of the reconstruction accuracy is provided by the 3D renderings in Fig. 1 and on the dataset webpage. Without our novel subpixel estimation and self-calibration components, the reconstruction is very noisy and exhibits visible aliasing effects (Fig. 1d); the sample standard deviations are typically about twice as large.

We can also assess accuracy in manually selected planar scene regions by fitting a plane to the recovered disparites and measuring the residuals. For 17 planar regions selected from the Motorcycle, Djembe, and Piano datasets, we obtain average absolute residuals of $\bar{r} = 0.032$, an improvement by a factor of 8 over integer disparities ($\bar{r} = 0.252$), and by a factor of 4 over our system *without* the novel subpixel and self-calibration components ($\bar{r} = 0.135$).

Manual cleanup: In some cases we get erroneous reconstructions, mostly due to reflective surfaces close to parallel to the viewing direction. Our pipeline converts disparities to 3D meshes, which can be edited (e.g., using MeshLab) to remove erroneous surfaces. The edited meshes are then used to invalidate the corresponding pixels. Less than half of our scenes required manual editing.

Perfect and imperfect rectification: The last step in our pipeline is to compute 2D disparity maps for the initial "imperfect" calibration. We do this by selecting pairs of corresponding pixel locations and warping them, first using the inverse perfect, then the forward imperfect rectification transform. Finally we crop images and disparity maps to remove any invalid borders due to the rectification transforms. We also introduce a disparity offset to ensure positive disparities over the entire depth range. We provide disparities in PFM format together with camera calibration information for all views. Fig. 2 shows a subset of our new datasets available at `http://vision.middlebury.edu/stereo/data/2014/`.

4 Stereo experiments

We test our new datasets using 3 state-of-the-art stereo methods that are able to handle high-resolution images: a correlation method employing a 7×7 census transform [33] and aggregation with overlapping 9×9 windows [16]; the fast ELAS method by Geiger et al. [8]; and the semi-global matching (SGM) method by Hirschmüller [15]. We run all three methods on our new datasets, and also on the Middlebury Cones and Teddy benchmark images [29] for comparison. Fig. 7 shows the error percentages. The first plot demonstrates that our new datasets provide a range of challenges that significantly exceeds those of existing datasets (plotted on the left). The second plot demonstrates that imperfect rectification can yield significantly higher errors over accurate rectification; visual inspection
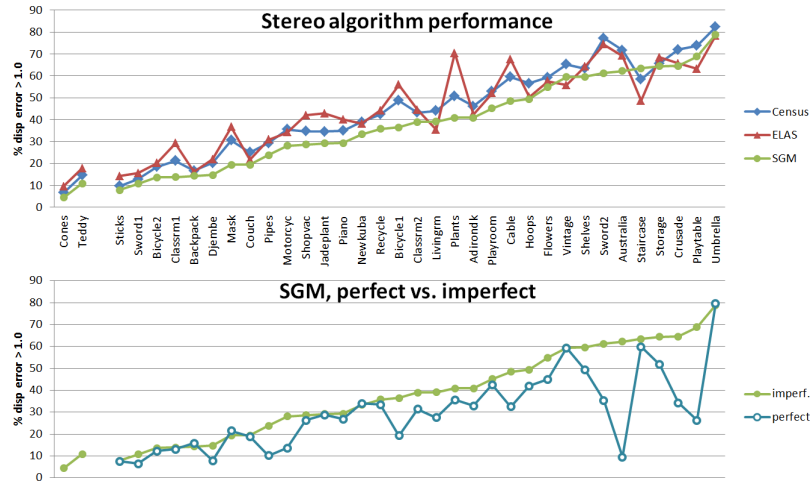
Fig. 7. Stereo algorithm performance on our new datasets. Top: Bad pixel percentages (% error > 1.0) for Census, ELAS, and SGM, sorted by SGM error, for imperfect calibration. Bottom: SGM errors for imperfect and perfect calibration. Two existing datasets, Cones and Teddy [29], are included on the left for comparison.

Avg abs y-disp range	0.0–0.5	0.5–1.0	1.0–1.5	1.5–3.0
Number of datasets	18	6	6	3
Avg disp error increase	9%	49%	61%	154%

Table 1. Increase in disparity error (from perfect to imperfect) for different amounts of rectification error as measured by average absolute y-disparity.

of the error maps shows that errors are particularly severe in misaligned regions of high-frequency texture. When averaged over multiple datasets, there is also a clear correlation between increase in disparity error and amount of vertical rectification error (see Table 1).

5   Conclusion

We have presented a structured lighting system with novel subpixel and self-calibration components for creating high-resolution stereo datasets of static indoor scenes. We achieve significantly higher disparity and rectification accuracy than those of exisiting datasets. We contribute 33 new datasets to the community at `http://vision.middlebury.edu/stereo/data/2014/`, including versions with realistic rectification errors. Experiments demonstrate that our new datasets present a new level of challenge for stereo algorithms, both in terms of resolution and scene complexity. The challenge will be even greater when our images from different exposures, illuminations, or the point-and-shoot cameras are used. We hope that our datasets will inspire research into the next generation of stereo algorithms.

Acknowledgments

References

1. 3dMD: 3D imaging systems. `http://www.3dmd.com/`
2. Acute3D: 3D photogrammetry software. `http://www.acute3d.com/`
3. Batlle, J., Mouaddib, E., Salvi, J.: Recent progress in coded structured light as a technique to solve the correspondence problem: a survey. Pat. Rec. 31(7), 963–982 (1998)
4. Besl, P.: Active optical range imaging sensors. MV&A 1(2), 127–152 (1988)
5. Brown, M., Burschka, D., Hager, G.: Advances in computational stereo. TPAMI 25(8), 993–1008 (2003)
6. Davis, J., Nehab, D., Ramamoorthi, R., Rusinkiewicz, S.: Spacetime stereo: A unifying framework for depth from triangulation. TPAMI 27(2), 296–302 (2005)
7. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? The KITTI vision benchmark suite. In: CVPR (2012)
8. Geiger, A., Roser, M., Urtasun, R.: Efficient large-scale stereo matching. In: ACCV (2010)
9. Gupta, M., Agrawal, A., Veeraraghavan, A., Narasimhan, S.: A practical approach to 3D scanning in the presence of interreflections, subsurface scattering and defocus. IJCV 102(1-3), 33–55 (2013)
10. Gupta, M., Nayar, S.: Micro phase shifting. In: CVPR. pp. 813–820 (2012)
11. Hansen, P., Alismail, H., Rander, P., Browning, B.: Online continuous stereo extrinsic parameter estimation. In: CVPR. pp. 1059–1066 (2012)
12. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press (2000)
13. Hirschmüller, H., Gehrig, S.: Stereo matching in the presence of sub-pixel calibration errors. In: CVPR. pp. 437–444 (2009)
14. Hirschmüller, H., Scharstein, D.: Evaluation of stereo matching costs on images with radiometric differences. TPAMI 31(9), 1582–1599 (2009)
15. Hirschmüller, H.: Stereo processing by semiglobal matching and mutual information. TPAMI 30(2), 328–341 (2008)
16. Hirschmüller, H., Innocent, P., Garibaldi, J.: Real-time correlation-based stereo vision with reduced border errors. IJCV 47(1-3), 229–246 (2002)
17. Hosni, A., Bleyer, M., Gelautz, M.: Secrets of adaptive support weight techniques for local stereo matching. CVIU 117(6), 620–632 (2013)
18. Hu, X., Mordohai, P.: A quantitative evaluation of confidence measures for stereo vision. TPAMI 34(11), 2121–2133 (2012)
19. Izadi, S., et al.: KinectFusion: Real-time 3D reconstruction and interaction using a moving depth camera. In: ACM UIST. pp. 559–568 (2011)
20. Lazaros, N., Sirakoulis, G., Gasteratos, A.: Review of stereo vision algorithms: from software to hardware. J. Optomechatronics 2(4), 435–462 (2008)
21. Levoy, M., et al.: The digital Michelangelo project: 3D scanning of large statues. In: SIGGRAPH. pp. 131–144 (2000)
22. Lourakis, M.: levmar: Levenberg-Marquardt nonlinear least squares algorithms in C/C++. `http://www.ics.forth.gr/~lourakis/levmar/`

23. Lourakis, M., Argyros, A.: SBA: A software package for generic sparse bundle adjustment. ACM Trans. Math. Software 36(1), 1–30 (2009), `http://www.ics.forth.gr/~lourakis/sba/`
24. Nakamura, Y., Matsuura, T., Satoh, K., Ohta, Y.: Occlusion detectable stereo – occlusion patterns in camera matrix. In: CVPR. pp. 371–378 (1996)
25. Posdamer, J., Altschuler, M.: Surface measurement by space-encoded projected beam systems. CGIP 18(1), 1–17 (1982)
26. Salvi, J., Fernandez, S., Pribanic, T., Llado, X.: A state of the art in structured light patterns for surface profilometry. Pat. Rec. 43(8), 2666–2680 (2010)
27. Scharstein, D.: Middlebury stereo page. `http://vision.middlebury.edu/stereo/`
28. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. IJCV 47(1), 7–42 (2002)
29. Scharstein, D., Szeliski, R.: High-accuracy stereo depth maps using structured light. In: CVPR (2003)
30. Seitz, S., Curless, B., Diebel, J., Scharstein, D., Szeliski, R.: A comparison and evaluation of multi-view stereo reconstruction algorithms. In: CVPR. vol. 1, pp. 519–528 (2006)
31. Strecha, C., von Hansen, W., Van Gool, L., Fua, P., Thoennessen, U.: On benchmarking camera calibration and multi-view stereo for high resolution imagery. In: CVPR (2008)
32. Will, P., Pennington, K.: Grid coding: A preprocessing technique for robot and machine vision. In: IJCAI. pp. 66–70 (1971)
33. Zabih, R., Woodfill, J.: Non-parametric local transforms for computing visual correspondence. In: ECCV. pp. 151–158 (1994)
34. Zhang, L., Curless, B., Seitz, S.: Spacetime stereo: Shape recovery for dynamic scenes. In: CVPR. vol. II, pp. 367–374 (2003)