Evaluation of Stereo Matching Costs on Images with Radiometric Differences

Heiko Hirschmüller and Daniel Scharstein, *Member, IEEE*

***Abstract*— Stereo correspondence methods rely on matching costs for computing the similarity of image locations. We evaluate the insensitivity of different costs for passive binocular stereo methods with respect to radiometric variations of the input images. We consider both pixel-based and window-based variants like the absolute difference, the sampling-insensitive absolute difference, and normalized cross correlation, as well as their zero-mean versions. We also consider filters like LoG, mean, and bilateral background subtraction (BilSub) and non-parametric measures like Rank, SoftRank, Census, and Ordinal. Finally, hierarchical mutual information (HMI) is considered as pixelwise cost. Using stereo datasets with ground-truth disparities taken under controlled changes of exposure and lighting, we evaluate the costs with a local, a semi-global, and a global stereo method. We measure the performance of all costs in the presence of simulated and real radiometric differences, including exposure differences, vignetting, varying lighting and noise. Overall, the ranking of methods across all datasets and experiments appears to be consistent. Among the best costs are BilSub, which performs consistently very well for low radiometric differences; HMI, which is slightly better as pixel-wise matching cost in some cases and for strong image noise; and Census, which showed the best and most robust overall performance.**

***Index Terms*— stereo, matching cost, performance evaluation, radiometric differences**

I. Introduction

All passive stereo correspondence algorithms have a way of measuring the similarity of image locations. Typically, a *matching cost* is computed at each pixel for all disparities under consideration. The simplest matching costs assume constant intensities at matching image locations, but more robust costs can compensate for certain radiometric differences and noise.

Radiometric differences can be caused by the camera(s) due to slightly different settings, vignetting, image noise, etc. Radiometric pre-calibration can only compensate for some of these differences, and is not possible in all situations. Further differences may be due to non-Lambertian surfaces, for which the amount of reflected light depends on the viewing angle. While such differences can be reduced by making the stereo baseline smaller, this also reduces the geometric accuracy of the reconstruction. An example of real-world stereo data exhibiting many of the effects described above is given by Daimler AG's sequences taken by a calibrated stereo camera in a driving car [1].

Another source of radiometric differences is that the strength or positions of the light sources may change when images of a static scene are acquired at different times. For larger scenes, image acquisition will take some time and it may not be possible to control the light source (e.g., outdoors). Similar situations arise when matching aerial or satellite images.

Due to all of the above reasons, it is safe to say that any real-world stereo application requires radiometric robustness. This includes existing commercial systems, which employ different techniques, many of which are discussed in this paper. For example, Point Grey's Triclops stereo library [2] uses a band-pass filter, Videre's Small Vision System [3] uses a Laplacian of Gaussian filter, and Tyzx's Deep Sea system uses the census transform. Similarly, state-of-the-art multi-view stereo methods [4]–[6] use methods such as normalized cross correlation and mutual information for handling severe radiometric differences.

II. Related Work

Common pixel-based matching costs include absolute differences, squared differences, sampling-insensitive absolute differences [7], or truncated versions, both on gray and color images. Common window-based matching costs include the sum of absolute or squared differences (SAD / SSD) and normalized cross-correlation (NCC). In contrast to SAD and SSD, NCC accounts for gain differences (a multiplicative change) in the matching windows due to normalization. A constant offset (bias) of pixel values is often compensated by the zero-mean versions ZSAD, ZSSD and ZNCC. Alternatively, an offset change can also be reduced by filtering the images before matching using a mean filter, computing a gradient magnitude image (i.e. first derivative) [8] or Laplacian of Gaussian (i.e. smoothed second derivative) [9], [10]. Unfortunately, all of these filters result in a blurred disparity image. Ansar et al. [11] proposed background subtraction using a bilateral filter [12] for compensating radiometric differences without blurring.

Non-parametric matching costs were introduced for being robust against outliers that occur in window-based methods near object boundaries [13]–[15]. However, since non-parametric costs rely only on the relative ordering of pixel values, they are also invariant under all radiometric changes that preserve this order. The Rank and Census methods [13] can be implemented as a filter followed by a comparison using the absolute difference or Hamming distance. Ordinal measures [14], [16] compute the distance of rank permutations of corresponding windows.

Another category of methods tries to explicitly model the complex radiometric relationships between images. Mutual information (MI) has been introduced in computer vision by Viola and Wells [17]. MI has been first used for stereo matching by Chrastek and Jan [18], but with disappointing results. Later work on MI in window-based stereo methods [19]–[21] demonstrated its power to model complex radiometric relationships. Others used

H. Hirschmüller is with the Institute of Robotics and Mechatronics at the German Aerospace Center (DLR).

D. Scharstein is with Middlebury College, VT, USA.

approximations of MI [22] for a segment-wise stereo matching. It has been found [20], [21] that large windows are needed for collecting enough data for the required joint probability distribution, but large windows again result in blurring at object boundaries. Therefore, Fookes et al. [20] proposed a hierarchical method for estimating probability priors over the whole image at a lower resolution. These priors are fused with values collected from smaller matching windows, which results in a reliable probability distribution. Kim et al. [23] used MI pixel-based without matching windows in the global graph-cuts stereo method. The probability distribution is iteratively calculated over the whole image using a prior disparity, which is random at the beginning. Finally, it has been shown [24] that a hierarchical calculation of pixel-wise MI is as accurate as an iterative calculation, and just 15-20% slower than a direct calculation using absolute differences.

Zhang et al. [25] compute simultaneously the disparity image and an illumination ratio map in a BP framework for handling complex local intensity variations. We attempted to include the authors' implementation of this method in our comparison, but we were unable to find parameter settings to yield competitive performance across our test datasets.

In the multi-view case, the same techniques (e.g., NCC or MI) can be used for handling radiometric differences [6]. However, multiple images can also be used for explicitly modeling non-Lambertian scenes [26]–[28] or reflections [29]. Furthermore, special imaging setups, like multiple images with one light source that moves away [30], or "Helmholtz stereo" where camera and light source are interchanged [31], can be used for handling non-Lambertian scenes successfully. In this paper, however, we focus only on passive methods that work on a single stereo pair with unknown radiometric distortions and unknown light sources.

Recent stereo surveys [32], [33] and the Middlebury online evaluation [34] compare state-of-the-art stereo methods on test data with complex geometries and varied texture. Other evaluations focus on certain aspects like aggregation methods for real-time matching [35]. However, the insensitivity of matching costs is in these papers not evaluated since the stereo test sets are typically pairs of radiometrically very similar images.

Gautama et al. [36] compare ZNCC and Census for car-seat occupancy detection using window-based real-time stereo vision. The performance in the presence of radiometric differences was not explicitly tested. For their application, Census performed faster and more accurately than ZNCC. Banks and Corke [37] compared SAD, SSD, NCC, their zero mean variants, Rank and Census for window-based stereo matching. The evaluation includes visual inspection and the count of pixels that passed the left/right consistency check on images with real radiometric differences and synthetic images without differences. Rank and Census performed better than the classical matching costs. Fookes et al. [38] compared SAD, ZSAD, NCC, ZNCC, Rank and MI for window-based stereo matching. Their evaluation also measures the number of pixels that passes the validity check. They concluded that ZNCC and Rank performs best on images without radiometric changes, while the performance of MI is best on images with artificially changed radiometry. Sarkar and Bansal [21] compared MI and SSD for window-based matching on images with ground truth and artificial radiometric changes. They found that MI handles radiometric differences well, but its performance depends heavily on the window size.

The scope of this paper is the evaluation and comparison of parametric and non-parametric matching costs as well as MI on images with several common radiometric differences. In contrast to previous studies [21], [36]–[38], we test all costs not only for window-based matching, but (where applicable) also for pixel-based matching with a semi-global method (SGM) and graph cuts (GC) as a strong global method. Furthermore, in addition to simulated global and local radiometric changes, we perform experiments on stereo pairs with real radiometric differences. All tests on simulated variations and real changes are evaluated against ground-truth disparities.

The focus of this paper is on matching costs that explicitly or implicitly handle radiometric differences. This excludes popular methods like the correlation-based weighting according to proximity and color similarity [39], since this is an aggregation approach rather than a new matching cost. As mentioned earlier, we also exclude methods that require more than two views or calibrated light sources, and restrict our evaluation to passive methods that work on a single stereo pair with unknown radiometric distortions.

III. Matching Costs

It is important to distinguish between matching costs and methods that use these costs. In this paper we compare all possible combinations of 15 costs and 3 stereo methods. The costs are grouped into parametric costs, non-parametric costs, and mutual information. All parametric costs use the magnitude of pixel values and can be subdivided in methods that require identity, allow different offsets or scalings or both. Non-parametric costs use only the local ordering of intensities and can therefore handle all monotonic mappings. Mutual information can model even more complex relationships between images.

We initially define all matching costs on intensity (luminance) instead of color, which we store as 8-bit unsigned integers. See Fig. 1(a) for an example. Note that all costs can simply be extended to color by computing the costs for each color channel separately and then summing the costs over all channels; for some costs or filters there are more natural definitions, which we describe below. In our experiments below we focus mainly on the intensity versions of the costs, but we investigate the potential of color matching in Section V-F.

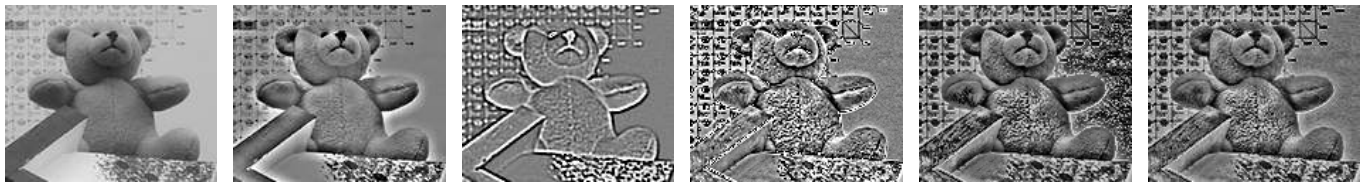
A. Parametric Matching Costs

Our first parametric cost function is the commonly-used *absolute difference (AD)*, which assumes brightness constancy (i.e. identity) for corresponding pixels, and which serves as a baseline performance measure of our evaluation. In global methods, the differences are used pixel-wise. Local stereo methods use the *sum of the absolute differences (SAD)* over all pixels $\mathbf{q}$ of a certain neighborhood $N_\mathbf{p}$, typically a square window. We use the notation $\mathbf{d} = [d \;\; 0]^T$ for the disparity. We assume rectified stereo pairs throughout. Thus, for a pixel $\mathbf{p}$ in the left image, the corresponding pixel in the right image is $\mathbf{p}-\mathbf{d}$.

$$C_{AD}(\mathbf{p},\mathbf{d}) = |I_L(\mathbf{p}) - I_R(\mathbf{p}-\mathbf{d})| \tag{1}$$

$$C_{SAD}(\mathbf{p},\mathbf{d}) = \sum_{\mathbf{q}\in N_\mathbf{p}} |I_L(\mathbf{q}) - I_R(\mathbf{q}-\mathbf{d})| \tag{2}$$

Additionally, we also test the sampling-insensitive absolute difference of Birchfield and Tomasi *(BT)* [7]. It computes the absolute distance between the extrema of linear interpolations of



(a) Intensity image (b) Mean filter (c) LOG filter (d) BilSub filter (e) Rank filter (f) SoftRank filter

Fig. 1. Different filters on a part of the Teddy image. The contrast of (b)–(d) has been increased for better visualization.

the corresponding pixels of interest with their neighbors. This method is often used for pixel-wise global methods, but can also be used for window-based matching. (Other window-based sampling-insensitive costs exist [40] but are not evaluated here.)

$$\begin{aligned} C_{BT}(\mathbf{p},\mathbf{d}) &= \min(A,B) \\ A &= \max(0, I_L(\mathbf{p}) - I_R^{max}(\mathbf{p}-\mathbf{d}), I_R^{min}(\mathbf{p}-\mathbf{d}) - I_L(\mathbf{p})) \\ B &= \max(0, I_R(\mathbf{p}-\mathbf{d}) - I_L^{max}(\mathbf{p}), I_L^{min}(\mathbf{p}) - I_R(\mathbf{p}-\mathbf{d})) \\ I^{min}(\mathbf{p}) &= \min(I^-(\mathbf{p}), I(\mathbf{p}), I^+(\mathbf{p})) \\ I^{max}(\mathbf{p}) &= \max(I^-(\mathbf{p}), I(\mathbf{p}), I^+(\mathbf{p})) \\ I^-(\mathbf{p}) &= \left(I\left(\mathbf{p} - [1\ 0]^T\right) + I(\mathbf{p})\right)/2 \\ I^+(\mathbf{p}) &= \left(I\left(\mathbf{p} + [1\ 0]^T\right) + I(\mathbf{p})\right)/2 \end{aligned} \tag{3}$$

Our next three cost functions are actually filters that change the input images separately before matching via absolute difference. The *mean filter* simply subtracts from each pixel the mean intensities within a neighborhood of 15×15 pixels centered at the pixel of interest. A constant offset of 128 is added to avoid negative numbers when storing the result back into an 8-bit image (Fig. 1(b)). Thus, the mean filter performs background subtraction for removing a local intensity offset.

$$I_{mean}(\mathbf{p}) = I(\mathbf{p}) - \frac{1}{|N_\mathbf{p}|}\sum_{\mathbf{q}\in N_\mathbf{p}} I(\mathbf{q}) + 128 \tag{4}$$

The *Laplacian of Gaussian* (LoG) is a bandpass filter, which performs smoothing for removing noise and removes an offset in intensities. The filter is often used in local real-time methods [9], [10]. Here we use a LoG filter with a standard deviation of $\sigma = 1$ pixel, which is applied by convolution with a 5×5 LoG kernel (Fig. 1(c)).

$$I_{LoG} = I \otimes K_{LoG}, \quad K_{LoG}(x,y) = -\frac{1}{\pi\sigma^4}\left(1 - \frac{x^2+y^2}{2\sigma^2}\right) e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{5}$$

Furthermore, we consider *background subtraction by bilateral filtering (BilSub)* [11]. The bilateral filter [12] sums neighboring values weighted according to proximity and color similarity. It smoothes without blurring high contrast texture. Background subtraction is implemented by subtracting from each value the corresponding value of the bilateral filtered image. This effectively removes a local offset without blurring high contrast texture differences that may correspond to depth discontinuities. We use a kernel of 15×15 pixels, a spatial distance (which defines the amount of smoothing) of $\sigma_s = 3$, and a radiometric distance (which prevents smoothing over high-contrast texture differences) of $\sigma_r = 20$. On intensity images, the radiometric distance is computed as the absolute difference of intensities as defined in (6). Fig. 1(d) shows the result. On color images, we use the distance in CIELab space, as originally suggested [12].

$$\begin{aligned} I_{BilSub}(\mathbf{p}) &= I(\mathbf{p}) - \frac{\sum_{\mathbf{q}\in N_\mathbf{p}} I(\mathbf{p}) e^s e^r}{\sum_{\mathbf{q}\in N_\mathbf{p}} e^s e^r} \\ s &= -\frac{(\mathbf{q}-\mathbf{p})^2}{2\sigma_s^2} \qquad r = -\frac{(I(\mathbf{q})-I(\mathbf{p}))^2}{2\sigma_r^2} \end{aligned} \tag{6}$$

For window-based stereo methods, there are further common costs for removing an offset in intensities. The *zero-mean sum of absolute differences (ZSAD)* subtracts the mean intensity of the window from each intensity inside the window before computing the sum of absolute differences. Note that the subtracted mean is the same for each pixel in the correlation window, in contrast to the mean filter where each pixel has its own window for computing the mean.

$$\begin{aligned} C_{ZSAD}(\mathbf{p},\mathbf{d}) &= \sum_{\mathbf{q}\in N_\mathbf{p}} |I_L(\mathbf{q}) - \bar{I}_L(\mathbf{p}) - I_R(\mathbf{q}-\mathbf{d}) + \bar{I}_R(\mathbf{p}-\mathbf{d})| \\ \bar{I}(\mathbf{p}) &= \frac{1}{N_\mathbf{p}}\sum_{\mathbf{q}\in N_\mathbf{p}} I(\mathbf{q}) \end{aligned} \tag{7}$$

Normalized cross-correlation (NCC) is another window-based matching technique that is commonly used. NCC compensates gain changes and is statistically the optimal method for dealing with Gaussian noise. However, NCC tends to blur depth discontinuities more than many other matching costs, because outliers lead to high errors within the NCC calculation [10].

$$C_{NCC}(\mathbf{p},\mathbf{d}) = \frac{\sum_{\mathbf{q}\in N_\mathbf{p}} I_L(\mathbf{q}) I_R(\mathbf{q}-\mathbf{d})}{\sqrt{\sum_{\mathbf{q}\in N_\mathbf{p}} I_L(\mathbf{q})^2 \sum_{\mathbf{q}\in N_\mathbf{p}} I_R(\mathbf{q}-\mathbf{d})^2}} \tag{8}$$

MNCC, due to Moravec [41], is a commonly-used variant of NCC. It is an approximation of NCC and can be computed faster. We selected the standard NCC as MNCC gave slightly inferior results in our experiments.

In addition to NCC, we separately consider the zero-mean variant *ZNCC* in our evaluation. ZNCC is the only parametric cost that can compensate for differences in both gain and offset within the correlation window.

$$\begin{aligned} &C_{ZNCC}(\mathbf{p},\mathbf{d}) = \\ &\frac{\sum_{\mathbf{q}\in N_\mathbf{p}} (I_L(\mathbf{q}) - \bar{I}_L(\mathbf{p}))(I_R(\mathbf{q}-\mathbf{d}) - \bar{I}_R(\mathbf{p}-\mathbf{d}))}{\sqrt{\sum_{\mathbf{q}\in N_\mathbf{p}} (I_L(\mathbf{q}) - \bar{I}_L(\mathbf{p}))^2 \sum_{\mathbf{q}\in N_\mathbf{p}} (I_R(\mathbf{q}-\mathbf{d}) - \bar{I}_R(\mathbf{p}-\mathbf{d}))^2}} \end{aligned} \tag{9}$$

B. Non-Parametric Matching Costs

Non-parametric matching costs are based on the local order of intensities. Some of these costs can be again implemented as filters that change the input images individually. The *Rank filter* replaces the intensity of a pixel with its rank among all pixels within a certain neighborhood. It was originally proposed [13] to increase robustness of window-based methods to outliers within

the neighborhood, which typically occur near depth discontinuities and leads to blurred object borders. Since all non-parametric costs only depend on the ordering of intensities and not the magnitude of intensities, they tolerate all radiometric distortions that preserve this ordering. Here we use a Rank filter with a square window of 15×15 pixels centered at the pixel of interest.

$$I_{Rank}(\mathbf{p}) = \sum_{\mathbf{q}\in N_\mathbf{p}} T[I(\mathbf{q}) < I(\mathbf{p})] \tag{10}$$

The function T[] is defined to return 1 if its argument is true and 0 otherwise. The transformed images are matched with the absolute difference.

The Rank filter is known to be susceptible to noise in textureless areas as can be seen in the area to the right of the teddy in Fig. 1(e). The *Soft Rank filter* was proposed by Zitnick [42] to reduce this problem by defining a linear, soft transition zone between 0 and 1 for values that are close together.

$$I_{SoftRank}(\mathbf{p}) = \sum_{\mathbf{q}\in N_\mathbf{p}} \min\left(1, \max\left(0, \frac{I(\mathbf{p}) - I(\mathbf{q})}{2t} + \frac{1}{2}\right)\right) \tag{11}$$

We used the threshold $t = 8$. The result in Fig. 1(f) is clearly less noisy in textureless areas.

We also consider the *Census filter* [13]. It defines a bit string where each bit corresponds to a certain pixel in the local neighborhood around a pixel of interest. A bit is set if the corresponding pixel has a lower intensity than the pixel of interest. Thus, Census not only stores the intensity ordering like Rank, but also the spatial structure of the local neighborhood. We use a window of 9×7 pixels and store the bit string in a 64-bit integer. The transformed images are matched by computing the Hamming distance between corresponding bit strings. The performance of Census is reported [13] to be superior to Rank, but the computation on standard CPU's is more time consuming due to the calculation of the Hamming distance.

The final non-parametric cost we consider is the *ordinal measure* proposed by Bhat et al. [43], which is based on the distance of rank permutations of corresponding matching windows. It cannot be implemented as a filter and requires window-based matching. Its potential advantage over Rank and Census filters is that it avoids the dependency on the value of the pixel of interest.

C. Mutual Information

Our last matching cost is based on *mutual information* (MI). MI enables registering of images with complex radiometric relationships [17]. The MI of two images is calculated by summing the entropy of the probability distributions (H_{I_1} and H_{I_2}) of the overlapping parts of each image and subtracting the entropy of the joint probability distribution (H_{I_1,I_2}) of pixel-wise correspondences of both images. The probability distributions are derived from the histograms of the corresponding image parts. The MI value directly expresses how well images are registered. This follows from the observation that the joint histogram of well-registered images has just a few high peaks in contrast to poorly registered images where the joint histogram is rather flat. Thus, for well-registered images, the entropy of the joint probability distribution H_{I_1,I_2} is low, while the entropy of the individual probability distributions H_{I_1} and H_{I_2} is nearly constant as long as the overlapping image parts are roughly the same.

It is straightforward to use MI for calculating how well two image regions correspond. However, typical windows of 9×9 or 11×11 pixel do not contain enough pixels for deriving meaningful probability distributions [20], [21]. Larger windows would be needed, but larger windows are known to increase blurring of discontinuities [10]. Therefore, we use a computation of MI that is based on the whole image and allows pixel-wise matching [23], [24]. It works by using an initial disparity image that defines corresponding pixels of both images for computing the required probability distributions. Since this computation considers the whole image, the probability distributions become very reliable. A Taylor expansion of MI allows the derivation of a cost matrix that defines the matching cost for each combination of intensities [23]. This lookup table can be used by any window or pixel-based stereo matching method. The required initial disparity image can be set to random values in the beginning and iteratively refined. Each iteration uses the previous disparity image for computing a new matching cost lookup table. It has been found [23] that 3 iterations result already in a nearly stable, final disparity image.

In this paper we use the efficient Hierarchical MI (HMI) method [24], which starts with images that are downscaled by factor 16 and random disparities. The cost matrix is calculated for matching, which leads to the first calculated disparity image by any stereo method. The disparity image is used for recalculating the cost matrix. The process is iterated a few times before the disparity is upscaled for serving as initial guess for matching at $\frac{1}{8}$th of the full resolution. Upscaling and matching is repeated until the full resolution is reached. It should be noted that the disparity image of the lower-resolution level is used only for calculating the matching costs of the higher-resolution level, but not for restricting the disparity range, as this could easily lead to missing small objects. It has been found [24] that the hierarchical calculation performs as well as the iterative one. However, its theoretical runtime overhead is compared to a non-iterative algorithm (i.e. with another matching cost like BT) just 14%, if the runtime of the stereo method depends linearly on the number of pixels and disparities.

D. Summary

In the experiments below, we evaluate the parametric costs AD, BT, Mean/BT, LoG/BT, BilSub/BT, NCC, ZSAD, and ZNCC; the nonparametric costs Rank/AD, Rank/BT, SoftRank/AD, SoftRank/BT, Census, and Ordinal; and HMI. Of these, NCC, ZSAD, ZNCC, and Ordinal can only be used in window-based matching. While we have tested all possible combinations of filters and AD/BT, here we include only those combinations that give significant differences.

IV. Stereo Algorithms

The performance of a matching cost can depend on the algorithm that uses the cost. We thus consider three different stereo algorithms: a local, window-based method (Window), the semi-global method of [24] (SGM), and a global method using graph cuts [44] (GC). We implemented each of the matching costs for each stereo method, except for NCC, ZSAD, ZNCC and Ordinal which can only be used with the local method.

Our local stereo method (Window) is a simple window-based approach [9], [10], [33]. We use a square window of 9×9 pixels. After aggregating the matching cost over the window, the disparity with the lowest aggregated cost is selected (winner-takes-all). Subpixel interpolation is performed by fitting a parabola to

the winning cost value and its neighbors. This is followed by a left-right consistency check for invalidating occlusions and mismatches, and invalidation of disparity segments smaller than 160 pixels [45]. Invalid disparity areas are filled by propagating neighboring small (i.e., background) disparity values. The reason we perform these post-processing steps is to reduce the overall errors. One might argue that comparing the "raw" results would provide a more direct assessment of the different costs. We have found, however, that the resulting large errors impede a fair comparison of the costs, while the post-processing greatly improves the discrimination between the costs.

Our second stereo algorithm is the semi-global matching (SGM) method [24]. We selected it as an approach in-between local and global matching. There are other approaches in this category, e.g., dynamic programming (DP) [33], [46], [47], but SGM outperforms DP and yields no streaking artefacts. SGM aims to minimize a global 2D energy function $E(D)$ by solving a large number of 1D minimization problems. Following [24], the actual energy used is

$$E(D) = \sum_{\mathbf{p}} \Big(C(\mathbf{p}, D_{\mathbf{p}}) + \sum_{\mathbf{q} \in N_{\mathbf{p}}} P_1 \, \mathrm{T}[|D_{\mathbf{p}} - D_{\mathbf{q}}| = 1] + \sum_{\mathbf{q} \in N_{\mathbf{p}}} P_2 \, \mathrm{T}[|D_{\mathbf{p}} - D_{\mathbf{q}}| > 1] \Big). \tag{12}$$

The first term of (12) calculates the sum of a pixel-wise matching cost $C(\mathbf{p}, D_{\mathbf{p}})$ (as defined in Section III) for all pixels $\mathbf{p}$ at their disparities $D_{\mathbf{p}}$. The second term penalizes small disparity differences of neighboring pixels $N_{\mathbf{p}}$ of $\mathbf{p}$ with the cost P_1. Similarly, the third term penalizes larger disparity steps (i.e., discontinuities) with a higher penalty P_2. The value of P_2 is adapted to the local intensity gradient by $P_2 = \frac{P_2'}{|I_{b\mathbf{p}} - I_{b\mathbf{q}}|}$ for the neighboring pixels $\mathbf{p}$ and $\mathbf{q}$. This results in sharper depth discontinuities as they mostly coincide with intensity variations.

SGM calculates $E(D)$ along 1D paths from 8 directions towards each pixel of interest using dynamic programming. The costs of all paths are summed for each pixel and disparity. The disparity is then determined by winner-takes-all. Subpixel interpolation is performed as well as a left-right consistency check. Disparity segments below the size of 20 pixels are invalidated for getting rid of small patches of outliers. Invalid disparities are again interpolated.

Finally, we use a graph-cuts (GC) stereo algorithm as a representative of a global method [44], [48], [49]. Our implementation is based on the MRF library provided by [50]. We tried to use the same energy function $E(D)$ as for SGM. However, we found that for GC it gives better results to adapt the cost P_2 not linearly with the intensity gradient, but rather to double the value of P_2 for gradients below a given threshold, as proposed in [44]. Like SGM, GC only approximates the global minimum of $E(D)$, but it utilizes the full 2D connectivity for the smoothness term in contrast to SGM, which optimizes separately along 1D paths. Our GC implementation, unlike Window and SGM, neither includes subpixel interpolation nor accounts for occlusions.

We manually tuned the smoothness parameters of SGM and GC individually for each cost for the best performance on the radiometrically unchanged Tsukuba, Venus, Teddy and Cones images of the Middlebury test [34]. After the tuning phase, all parameters were kept constant for all images and experiments. This approach allows to concentrate on the performance of the matching cost rather than the stereo method.

V. Evaluation

In this section, we test all possible combinations of matching costs with the local, semi-global, and global stereo algorithms on standard test images without radiometric changes (Section V-A), on images with simulated radiometric changes (Section V-B), and on images with real radiometric changes (Section V-C). Subsequently we investigate and discuss scene dependence (Section V-D) and cost discriminability (Section V-E). In all of these experiments, we focus on intensity images. We then explore the benefit of color matching (Section V-F), and finally compare the runtime of the different costs (Section V-G).

A. Results on Images without Radiometric Changes

As a baseline for our subsequent experiments, we use the standard Middlebury stereo datasets Tsukuba, Venus, Teddy, and Cones [33], [51]. Fig. 2 shows the left images of each set. Since these images were taken in a laboratory with the same camera settings and under the same lighting conditions, radiometric changes are expected to be very small. We use a disparity range of 16 pixels for Tsukuba, 32 pixels for Venus, and 64 pixels for Teddy and Cones.

Additionally, we have created new stereo datasets with ground truth using the structured lighting technique of [51], which are available at `http://vision.middlebury.edu/stereo/data/`. In this paper we use the six datasets shown in Fig. 3: Art, Books, Dolls, Laundry, Moebius, and Reindeer. Each dataset consists of 7 rectified views taken from equidistant points along a line, as well as ground-truth disparity maps for viewpoint 2 and 6. In this paper we only consider binocular methods, so we use images 2 and 6 as left and right input images. Also, we downsample the original images to one third of their size, resulting in images of roughly 460×370 pixels with a disparity range of 80 pixels.

We systematically tuned the smoothness parameters of SGM and GC individually for each cost for the best performance on the Tsukuba, Venus, Teddy and Cones images. After the tuning phase, all parameters were kept constant for all images and experiments. Thus, the radiometrically unchanged Tsukuba, Venus, Teddy and Cones set forms the training set, while radiometrically changed versions of them as well as the new data sets are the test sets.

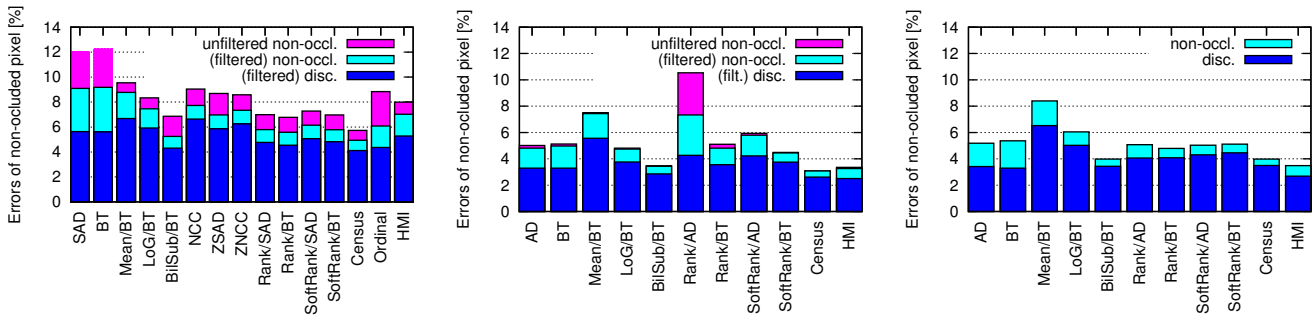
In all experiments, we evaluate the calculated disparity image by counting the number of pixels with disparities that differ by more than 1 from the ground truth. In our statistics we ignore occluded areas, because disparities at occlusions can by definition not be determined by matching of two images, but rather by extrapolation, which is not the focus of this paper. Also, our GC implementation does not consider occlusions, unlike Window and SGM. For the correlation results we also ignore an area of 4 pixels (i.e., the radius of the correlation window) at the image border. Our final error measure is the mean error percentage of all non-occluded pixels over the used datasets.

Fig. 4 shows, for all costs and stereo methods, the errors in all non-occluded areas without post-filtering, with post-filtering, and only near discontinuities of the Tsukuba, Venus, Teddy and Cones image set. Filtering is not done for GC, since its strong continuity model prevents small outlier regions. Clearly, for all costs the errors at discontinuities contribute most to the total error. Not surprisingly, the errors of the Window method are higher than that of SGM and GC.

Since many researchers use the BT cost for global methods, it is a bit surprising that in our test BT performs at the same

Fig. 2. The left images of the Tsukuba, Venus, Teddy, and Cones stereo pairs, which are used as training set.

Fig. 3. The new Art, Books, Dolls, Laundry, Moebius, and Reindeer stereo pairs, which are used as test set.



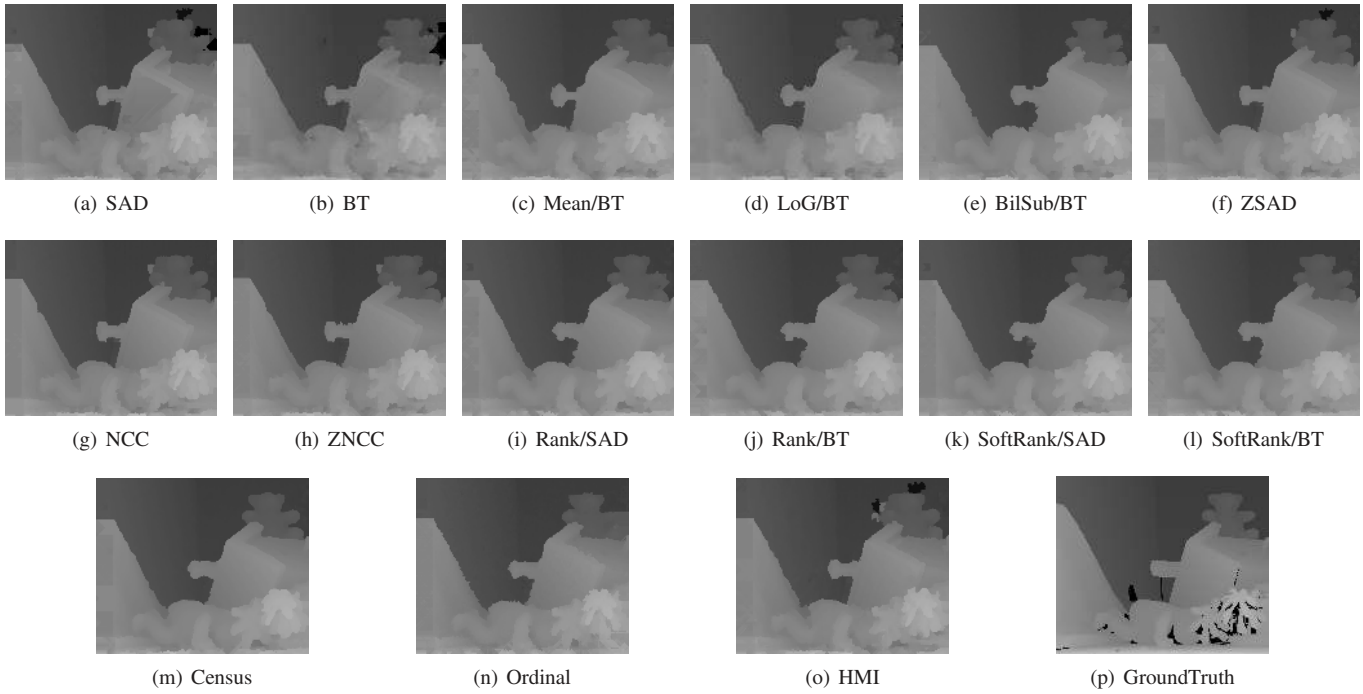
(a) Window (b) SGM (c) GC

Fig. 4. Mean errors over the Tsukuba, Venus, Teddy, and Cones training image pairs. Shown are the errors before and after post-filtering in all non-occluded areas as well as the fraction of these errors occurring near depth discontinuities.

level as AD for most tested stereo methods. It turns out that when evaluating the "raw" matching results of the SGM stereo method, AD yields in fact more errors than BT in regions with high-frequency texture. However, most such errors are detected by the consistency check (before post-filtering), and the missing disparities are mostly isolated pixels or very small areas that are easily recovered by interpolation. The Window method supports a decision by using the neighborhood, which contains many pixels that can be well matched by AD. Similarly, GC uses a strong 2D smoothness constraint that helps finding the correct disparities from the neighborhood as well. Thus, BT performs as expected, but the assumed disadvantage of AD is easily compensated by consistency checking and interpolation or strong smoothing constraints. An exception are Rank filtered images when using SGM. Here, AD is much less stable than BT.

The mean filter increases the errors near discontinuities and in case of SGM and GC the overall error. The LoG filter also blurs discontinuities, but reduces errors at other places, compared to AD or BT. In contrast, the BilSub filter reduces both errors and is one of the best cost for all three stereo methods. Although the ZSAD, NCC and ZNCC costs reduce the overall error compared to SAD and BT, they have the highest errors near discontinuities. NCC and ZNCC amplify the effect of outliers in the correlation window, which appear near discontinuities, due to the multiplication of intensities.

The performance of Rank and SoftRank is different for the three stereo methods. In the Window based method, SoftRank is slightly worse than Rank, while it is better when SGM is used. The performance is equal for GC. In the case of SGM, the combination with BT produces much lower errors than with AD. This may be explained by the property of BT to reduce the dissimilarity in high frequency regions. This appears to be more important for SGM, because SGM relies more on the matching cost as the smoothness constraint is applied along 1D paths, as opposed to Window and GC, which utilize the full 2D connectivity. As reported in the literature [13], [37], Census performs better than Rank and is among the best matching costs for all methods. The ordinal measure, however, performs slightly

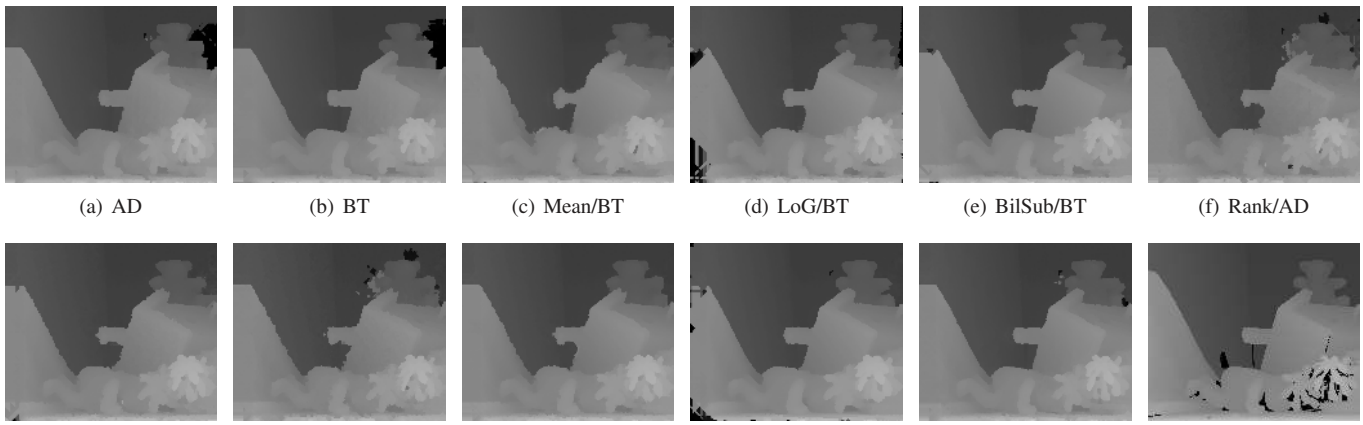


(a) SAD (b) BT (c) Mean/BT (d) LoG/BT (e) BilSub/BT (f) ZSAD

(g) NCC (h) ZNCC (i) Rank/SAD (j) Rank/BT (k) SoftRank/SAD (l) SoftRank/BT

(m) Census (n) Ordinal (o) HMI (p) GroundTruth

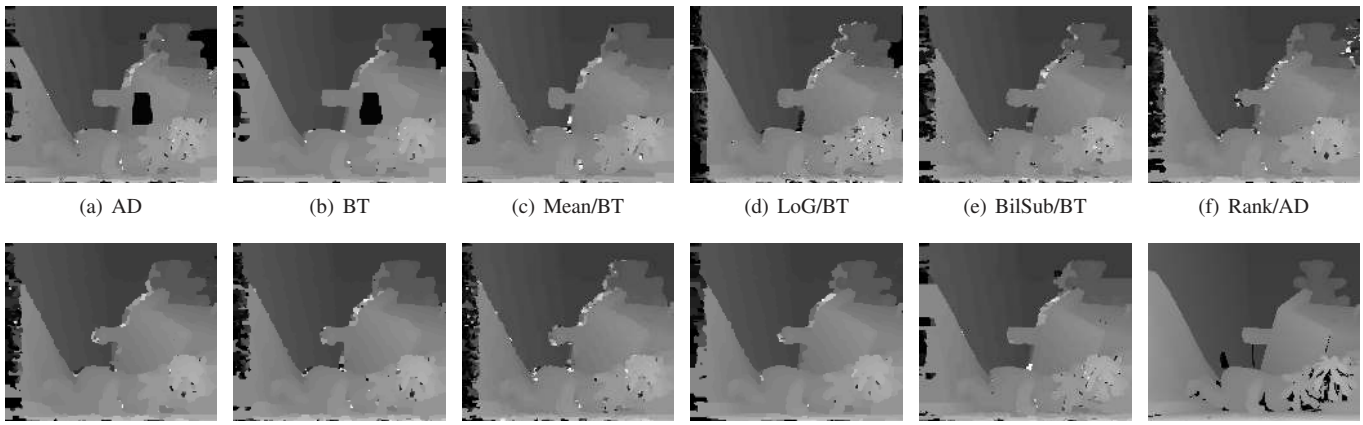
Fig. 5. Computed disparity images of the Teddy pair without radiometric transformations using the Window stereo method.



(a) AD (b) BT (c) Mean/BT (d) LoG/BT (e) BilSub/BT (f) Rank/AD

(g) Rank/BT (h) SoftRank/AD (i) SoftRank/BT (j) Census (k) HMI (l) GroundTruth

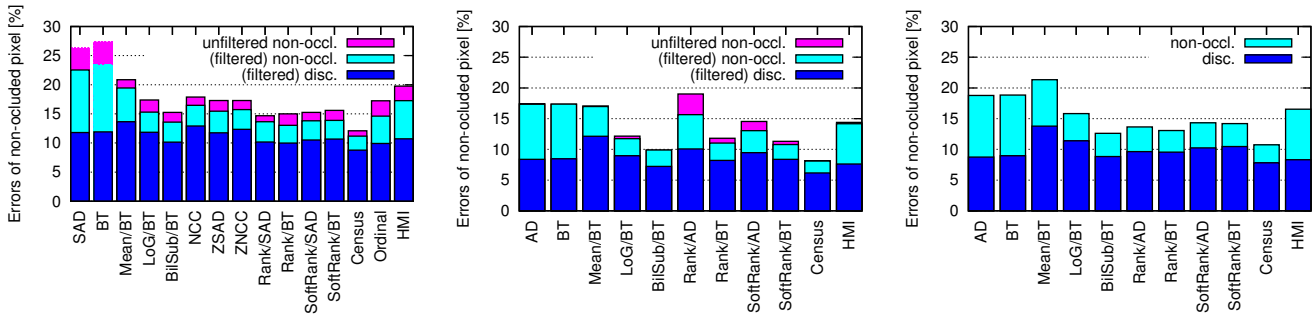
Fig. 6. Computed disparity images of the Teddy pair without radiometric transformations using the SGM stereo method.



(a) AD (b) BT (c) Mean/BT (d) LoG/BT (e) BilSub/BT (f) Rank/AD

(g) Rank/BT (h) SoftRank/AD (i) SoftRank/BT (j) Census (k) HMI (l) GroundTruth

Fig. 7. Computed disparity images of the Teddy pair without radiometric transformations using the GC stereo method.



(a) Window (b) SGM (c) GC

Fig. 8. Mean errors over the Art, Books, Dolls, Laundry, Reindeer, and Moebius test image pairs before and after post-filtering in all non-occluded areas, as well as the fraction of these errors occurring near depth discontinuities.

worse than Rank and Census. Finally, HMI appears not very successful in combination with the window-based method, but it performs very well with SGM and GC. The same observations can be made from the disparity images that are shown in Figs. 5–7 for the Teddy images. Recall that the GC implementation does not include a treatment for occlusions; thus, errors left to object borders should be ignored.

The same experiment has been done with the new Art, Books, Dolls, Laundry, Reindeer and Moebius image pairs. The result is shown in Fig. 8. It should be noted that our new images are more challenging than the standard test sets used in the previous sections, due to the increased disparity range, lack of texture, and the more complicated scene geometry. This is reflected in the higher matching errors: the best methods now have errors of about 8%, as opposed to about 3% before. However, the ordering of all costs is the same as the ordering in Fig. 4, except for BT, AD and HMI in combination with SGM and GC, which perform worse. We temporarily tried tuning the smoothness parameters for the new images, but this did not reduced errors visibly. Visual inspection of the computed disparity images revealed that objects in front of low textured background tend to be connected together with BT, AD and HMI in contrast to the best performing costs BilSub and Census. This makes sense, as the latter concentrate on small, high frequency texture variations, which are even there in low textured image parts. Thus, the worse performance is due to the more challenging scene content.

It may be surprising that many of the costs perform better than AD and BT on these input images without radiometric differences. It would rather appear logical that taking the absolute difference is best if corresponding points have exactly the same value. However, even though the images have been taken under controlled conditions, some radiometric differences are inherent, surfaces are not Lambertian, and the brightness constancy assumption is still violated. BilSub, Census, and HMI can compensate for these small differences.

To summarize, the performance of the matching costs can depend on the stereo method used. Nevertheless, BilSub and Census are among the best performers with all three stereo methods. HMI works equally well for the semi-global method and is best for the global method on some data sets.

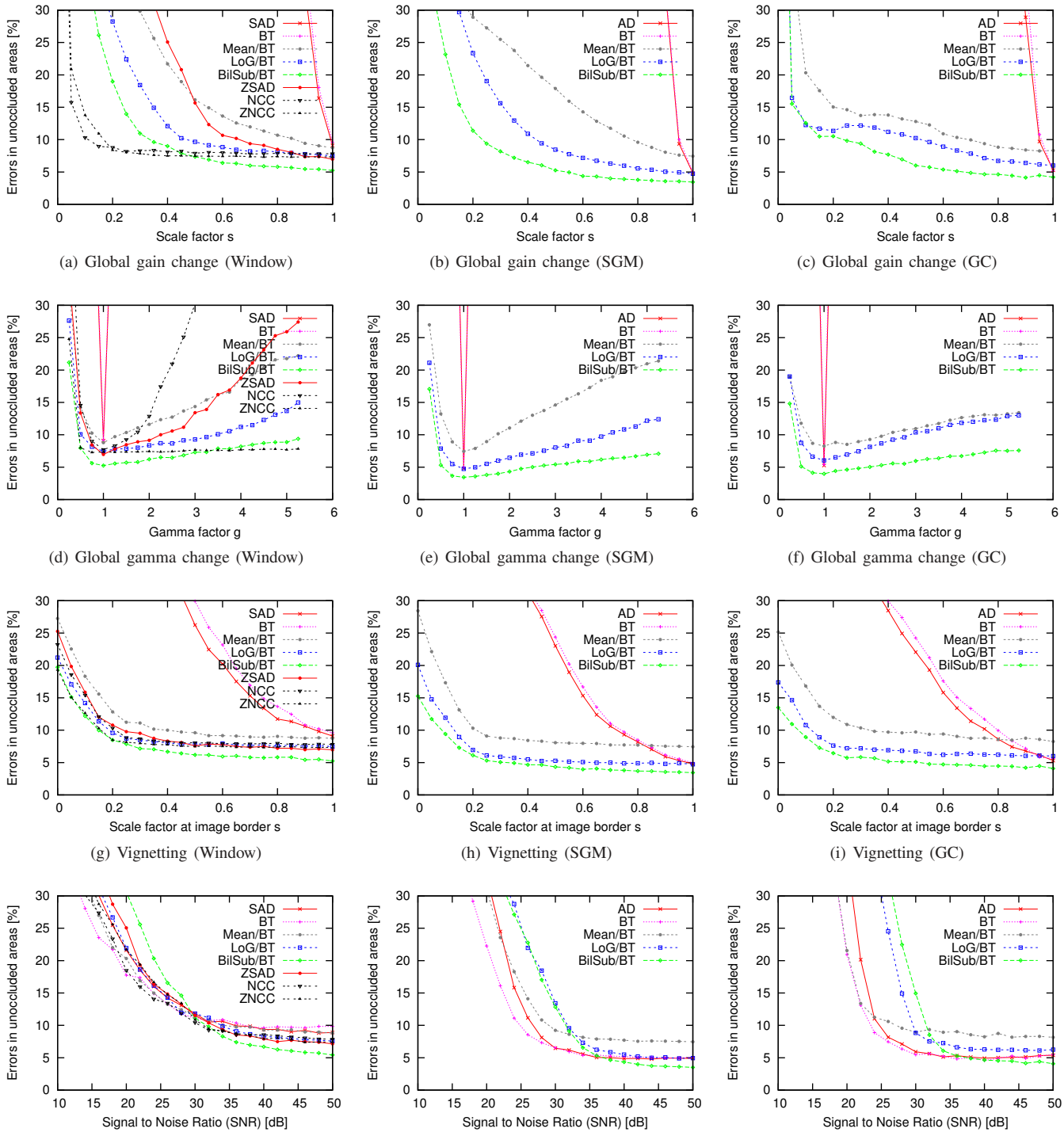
B. Simulated Radiometric Changes

In the next experiments, we explore the behavior of the matching costs on the Tsukuba, Venus, Teddy and Cones training image set (Fig. 2) with additional radiometric changes. Thus, we use the radiometrically changed versions of the training set as test set. First, the global brightness of the right stereo image is changed linearly (i.e., gain change) and nonlinearly (e.g., gamma change). The left stereo images remain unchanged. Furthermore, we apply a local brightness change that mimics a vignetting effect, i.e., the brightness decreases proportionally with the distance to the image center. This transformation is performed on both stereo images. Finally, we contaminate both stereo images with varying levels of Gaussian noise.

Since there are too many cost variants to show in one plot, we compare parametric and non-parametric costs separately (Figs. 9 and 10), and then compare the winners with HMI (Fig. 11).

Figs. 9(a)–9(c) shows the behavior of all parametric matching costs and filters on images with a gain change. The errors of AD and BT increase very quickly with decreasing brightness. This can be expected, because the absolute difference is based on the assumption that corresponding pixels have the same values, which is violated. The mean and LoG filters as well as ZSAD can compensate some of the differences, but they also degrade with higher differences. All three costs are designed for compensating an offset, but not a gain (i.e., scale) change. The bilateral background subtraction filter performs best for all stereo methods. It is only outperformed for s<0.5 by NCC and ZNCC, which show a very constant performance. The reason for the decreasing performance of BilSub with increasing differences is that BilSub, like LoG, mean and ZSAD only compensates for a constant offset, not for a gain change. NCC and ZNCC are the only parametric costs that explicitly account for a gain change. The reason for the sudden increase in errors below $s = 0.1$ is that that the transformed images are stored into 8 bits. Thus, low values of s also cause an information loss.

The same observations can be made for the case of global gamma changes as shown in Figs. 9(d)–9(f). The only exception is NCC, which performed in contrast to ZNCC much worse with increasing gamma values. It seems as if the nonlinear intensity change can be well compensated by the zero-mean calculation of ZNCC. In the case of the artificial vignetting effect (Figs. 9(g)–9(i)), AD and BT again degrade quickly, while all other costs can maintain their error level. BilSub is the best performing cost in all cases. The results for additive Gaussian noise with varying signal-to-noise ratios (SNR) are shown in Figs. 9(j)–9(l)). Higher SNR numbers mean lower noise. For the Window method the different costs perform quite similar, probably since summing over a fixed window acts like averaging, which reduces the effect of Gaussian noise for all costs. The situation is different for SGM



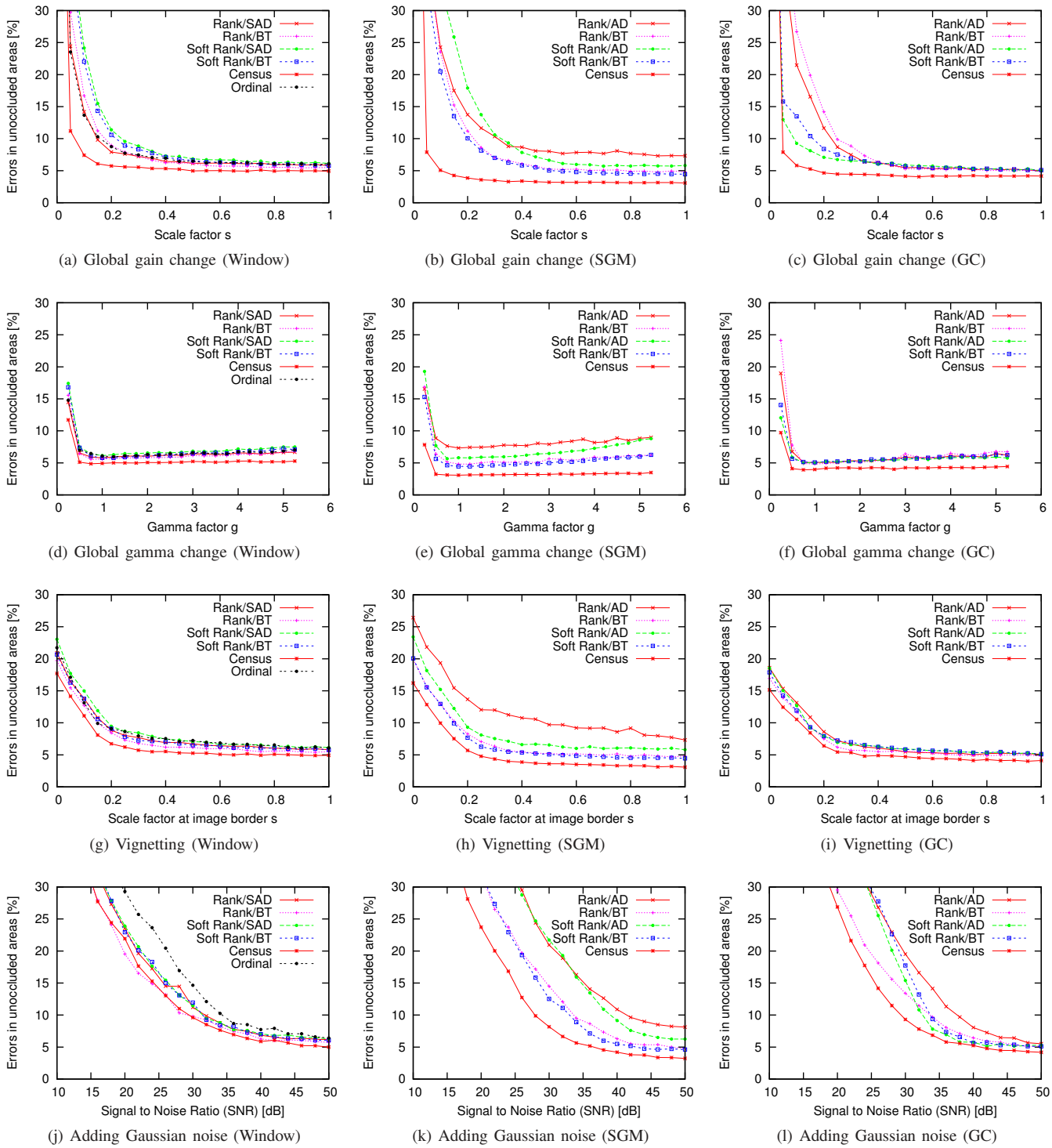
(a) Global gain change (Window) (b) Global gain change (SGM) (c) Global gain change (GC)

(d) Global gamma change (Window) (e) Global gamma change (SGM) (f) Global gamma change (GC)

(g) Vignetting (Window) (h) Vignetting (SGM) (i) Vignetting (GC)

(j) Adding Gaussian noise (Window) (k) Adding Gaussian noise (SGM) (l) Adding Gaussian noise (GC)

Fig. 9. Parametric matching costs on the Tsukuba, Venus, Teddy, and Cones datasets with simulated radiometric changes. All curves show the mean error in unoccluded areas over the four datasets using stereo methods with post-filtering. The columns correspond to the three stereo methods, while each row examines a different type of intensity change or noise.



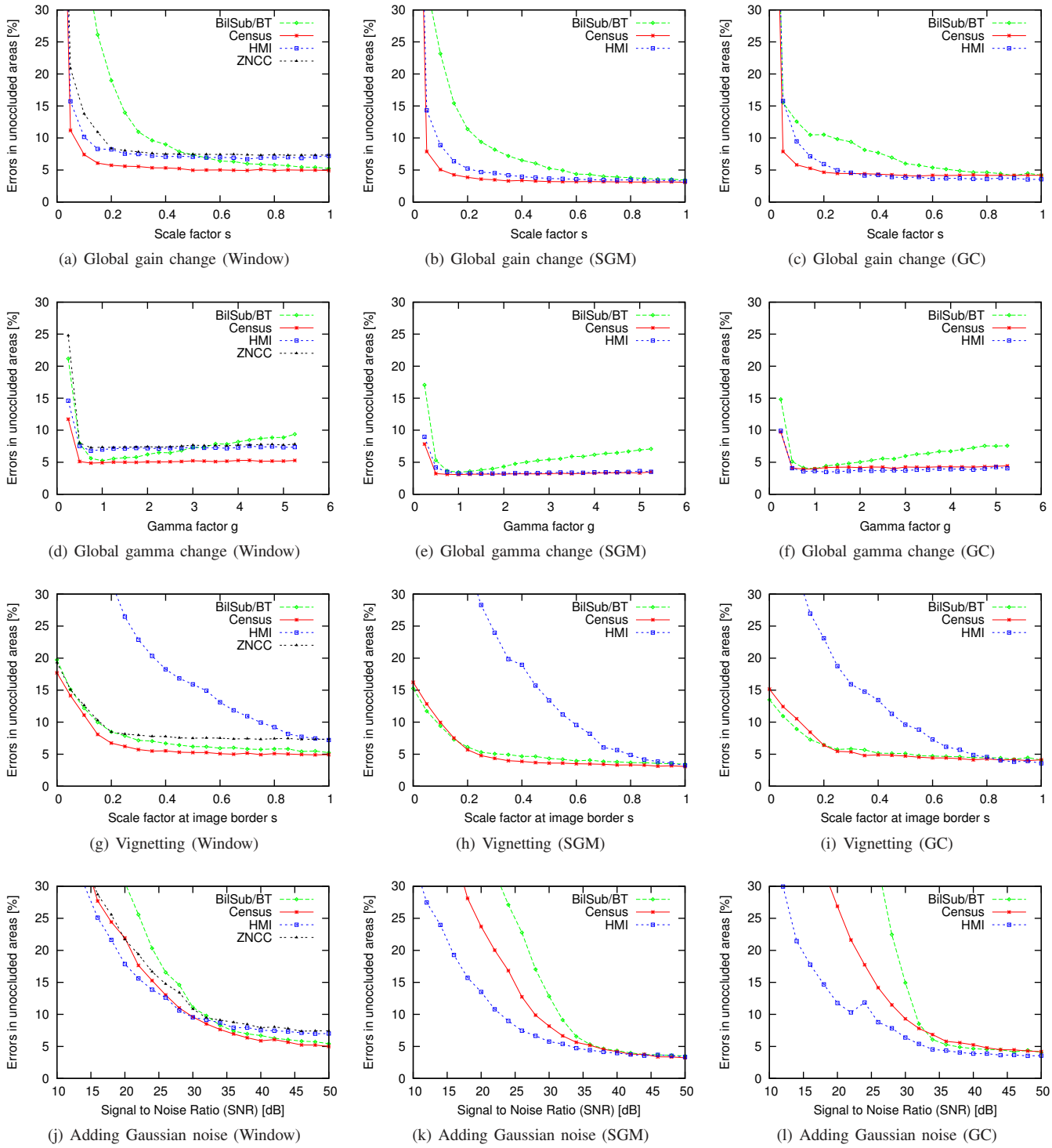
(a) Global gain change (Window) (b) Global gain change (SGM) (c) Global gain change (GC)

(d) Global gamma change (Window) (e) Global gamma change (SGM) (f) Global gamma change (GC)

(g) Vignetting (Window) (h) Vignetting (SGM) (i) Vignetting (GC)

(j) Adding Gaussian noise (Window) (k) Adding Gaussian noise (SGM) (l) Adding Gaussian noise (GC)

Fig. 10. Non-parametric matching costs on the Tsukuba, Venus, Teddy, and Cones datasets with simulated radiometric changes.



(a) Global gain change (Window) (b) Global gain change (SGM) (c) Global gain change (GC)

(d) Global gamma change (Window) (e) Global gamma change (SGM) (f) Global gamma change (GC)

(g) Vignetting (Window) (h) Vignetting (SGM) (i) Vignetting (GC)

(j) Adding Gaussian noise (Window) (k) Adding Gaussian noise (SGM) (l) Adding Gaussian noise (GC)

Fig. 11. The best non-parametric and parametric matching costs as well as HMI on the Tsukuba, Venus, Teddy, and Cones sets with simulated radiometric changes.

and GC, where LoG and BilSub perform worst at a certain noise level. Thus, the best parametric matching cost was the BilSub filter, except for large gain or gamma changes, since it does not explicitly handle gain changes. It has also problems with high noise levels. ZNCC has a higher initial error, but its performance is fairly constant, even with high radiometric changes.

Fig. 10 shows the same experiments with non-parametric matching costs. It can be seen that all non-parametric costs compensate the simulated changes quite well. Census is here the clear winner in all cases with all stereo methods.

In accordance with our findings, we selected BilSub and ZNCC as the best parametric costs and Census as the best non-parametric cost. These three costs are shown together with HMI in Fig. 11. In the direct comparison, Census performed as well as BilSub in the best case (i.e., without any changes), but the performance of Census is more constant, even if changes are higher. In comparison to ZNCC, Census has in all cases a lower error. The performance of HMI on images with global gain or gamma changes (Figs. 11(a)–11(f)) is similar to ZNCC in case of Window and similar to Census in case of SGM and GC. The likely reason is that Census also reduces the effect of outliers near depth discontinuities. This is important for a window-based method, but less so for pixel-based methods like SGM and GC. On images with the simulated vignetting effect (Figs. 11(g)–11(i)), the error of HMI increases much faster than that of all other method. The reason for the rather bad performance of HMI is that its cost is explicitly based on the assumption of a complex, but *global* radiometric transformation. The vignetting effect locally changes the brightness. BilSub and ZNCC can also only compensate global changes, but only related to their rather small windows. Furthermore, Census only requires an unchanged order, which is maintained. The situation is inverted on images with noise (Figs. 11(j)–11(l)), where HMI performs best for SGM and GC and at high noise levels also for Window. One reason for this is that HMI, unlike any of the other costs, implicitly models the noise distribution since the matching costs are derived from from the histograms, which are collected over the whole image.

We have also examined to what extent our results so far might be influenced by the scene structure, calibration errors, or the inherent radiometric distortions of the test images. To explore this issue, we created four new stereo pairs with constant disparities by simply shifting the left images of the Tsukuba, Venus, Teddy and Cones pairs (Fig. 2) by half of the disparity range used for each of the four original pairs. Thus, the resulting stereo pairs represent a scene with a perfectly fronto-parallel plane onto which real images are projected as texture. There is no calibration error and corresponding pixel are radiometrically exactly the same. We ran our entire set of experiments on these new images, and found that the behavior of the matching costs in the presence of different radiometric changes is essentially the same for the perfectly controlled case with planar images and the standard test images.

In summary, Census appears overall to be the most robust cost and it is in many cases the best. HMI can perform equally or slightly better on the pixel-wise matching methods SGM and GC and it is more stable in the presence of image noise. On the other hand, HMI performs worse on images with local changes like strong vignetting.

C. Real Exposure and Light Source Changes

As noted in the introduction, existing stereo test datasets are unusually radiometrically "clean" and do not require robust matching costs necessary for real-world stereo applications (unless, as in the previous sections, changes are introduced synthetically). To remedy this situation the six new stereo datasets (Fig. 3) additionally contain images of all scenes and viewpoints taken with three different exposures and under three different configurations of the light sources. We thus have 9 different images from each viewpoint that exhibit significant radiometric differences. Fig. 12 shows both exposure and lighting variations of the left image of the Art dataset.

We tested all combinations of costs and methods over all 3×3 combinations of either exposure or light changes. We found again that BilSub and ZNCC performed best among the parametric costs and that Census was the winner among the non-parametric costs. Here we thus only compare the winning costs, and we also include BT as "baseline" cost. The total matching error is calculated as before as the mean percentage of outliers (disparity error > 1) over all six datasets. The resulting curves are shown in Fig. 13.

Figs. 13(a)–13(c) show the result on pictures with different exposure settings. The change of exposure is supposed to be a global transformation, which should be similar to a global change of brightness, i.e., gain change. The behavior of BilSub, Census and ZNCC is as expected. Census and ZNCC can almost fully compensate the differences, while BilSub has problems with higher differences. We have already observed in Section V-A that HMI has more problems on this complex data set than the other costs. Of course, this does not change when introducing radiometric changes and HMI performs consistently much worse than Census.

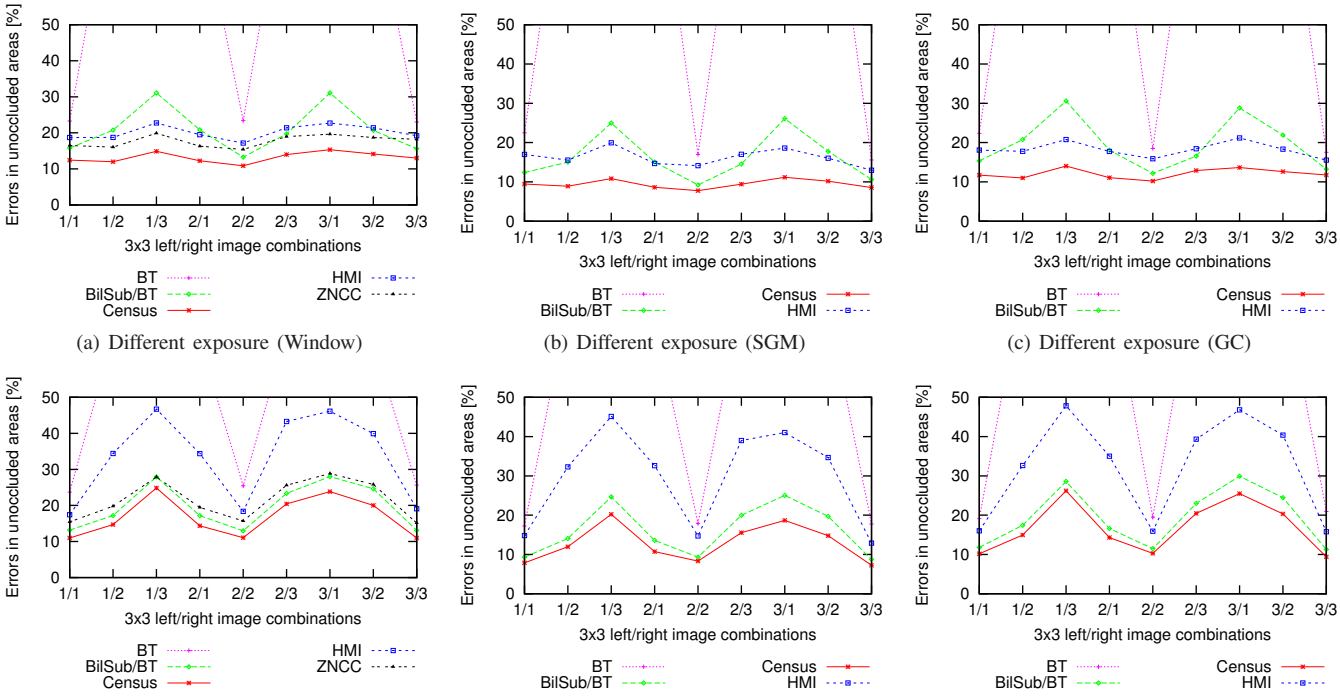
Changing the position and type of the light sources results in many local radiometric differences. The curves in Figs. 13(d)–13(f) show that matching images taken under different lighting conditions increases the error much more than before. However, the order of performance of all costs remains the same for all stereo methods. The rather bad performance of HMI can be expected in this experiment due to the many local radiometric differences.

Thus, the findings are essentially the same as for the images with simulated changes. Census performs best with all stereo methods on images with exposure and light changes. Next is BilSub, which only has more problems with very large changes. HMI has more problems even in the case of radiometrically similar images, which is due to the more complex scene structure. Also, HMI's inability to handle local radiometric changes can be observed again.

D. Variation of Results over Different Scenes

In our experiments so far, we show the mean error over the training or test set, which measures the average performance over images of different content and complexity. Additionally, it is an important question to what degree the performance of a certain cost depends on the scene content. This is statistically measured by the variance. However, simply reporting the variance of errors over all image pairs is not helpful since the variances are always large due to the widely varying complexity of the scenes. For instance, on the Venus image pair most costs yield errors of about one percent, while on the Art images the errors are about ten percent.

Fig. 12. The left image of the Art dataset with three different exposures and under three different light conditions. The right images have been captured under the same conditions, such that 3×3 combinations of matching are possible, separately for different exposures and light conditions.



(a) Different exposure (Window) (b) Different exposure (SGM) (c) Different exposure (GC)

(d) Different lighting (Window) (e) Different lighting (SGM) (f) Different lighting (GC)

Fig. 13. Results with different combinations of exposure or lighting conditions. The notation i/k indicates the combination of (exposure or lighting) settings, i.e., that the left image with setting i is matched with the right image with setting k. Thus, 1/1, 2/2 and 3/3 mean that the same settings are used for both images, which is the radiometrically unchanged case. For each cost we plot the mean error over all six stereo pairs.

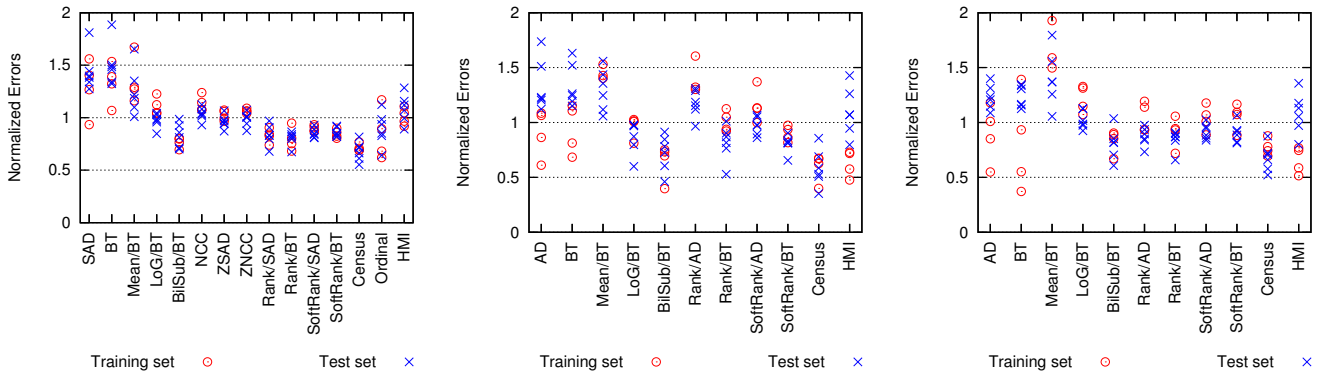
To obtain a meaningful comparison of errors across scenes with different complexity we normalize for each scene by the mean error over all costs. This is done for each stereo method and image pair separately. Thus, each error is divided by the mean error over all costs for the used stereo method and image pair, causing the normalized error to vary around the mean of 1.0. Fig. 14 shows that the performance of BilSub is mostly better than the mean and the performance of Census is always better than the mean. It also shows that the variation of errors is rather small for BilSub and Census, regardless of training or test images. In contrast, the performance of AD, BT and also HMI is rather wide-spread. They are pretty good for the training images and rather bad for the more complex test images. This has already been observed in the previous section.

To summarize, the performance for most matching costs is fairly independent of the scene. BilSub and Census perform particularly well on all scenes. However, the performance of some matching costs are scene dependent, in particular AD, BT and HMI.

E. Discriminability of Costs

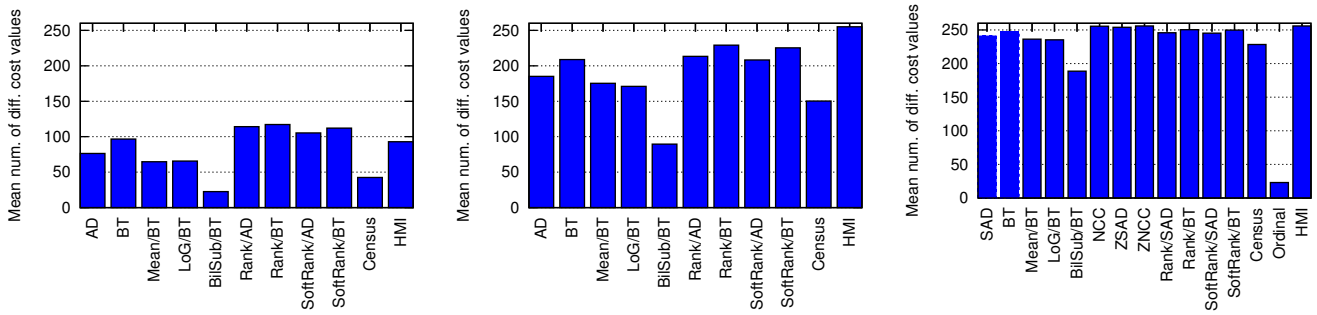
Another interesting issue is the discriminative power of matching costs. From a theoretical point of view, a cost like pixelwise Census with a 9×7 neighborhood can distinguish at most 62 different combinations. In contrast, the pixelwise absolute difference can distinguish up to 255 cases in 8-bit intensity images. However, the 62 different combinations of Census encode valuable high frequency variations of the local neighborhood. In contrast, it is probably not important to distinguish between highly differing intensities as in case of AD.

For an experimental evaluation of discriminability, we use all ten stereo images with a large disparity range of 256 and count the number of different responses of each matching cost along the disparity range. We ignore the left 255 pixels of each image in order to be able to utilize the full disparity range. Fig. 15 shows the average number of responses for all matching costs. The highest differences are visible when using the matching costs pixelwise (Fig. 15(a)). BilSub has the lowest discriminability, since the filter only leaves small high frequency variations. Census is second lowest and about half the value as AD. All Rank variants are highest, because they use a much larger neighborhood of



(a) Window (b) SGM (c) GC

Fig. 14. Visualization of variation of normalized errors over the four training and six test image pairs.



(a) Pixelwise (b) 3x3 Window (c) 9x9 Window

Fig. 15. Mean number of different cost values over a disparity range of 256 on ten stereo pairs.

15×15 pixels. The reason that HMI has a higher discriminability than AD is that HMI distinguishes between pairings of type (i,k) and (k,i), in contrast to AD.

Fig. 15(b) shows the mean number of different values using a small 3×3 neighborhood. Costs like BilSub and Census benefit most from this aggregation and reduce the distance to other costs. The result of HMI is already saturated due to the used disparity range of 256. The results of using the full correlation window size are shown in Fig. 15(c). Almost all costs give different responses along the whole disparity range. This figure also includes costs that can only be used with a correlation window.

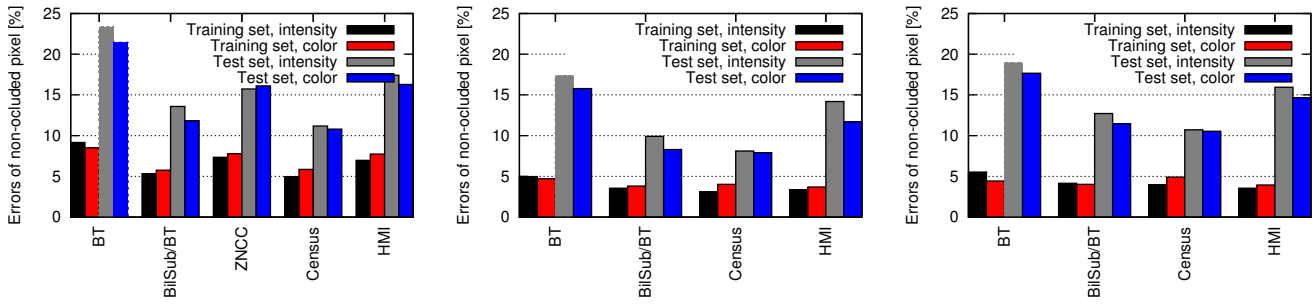
The Ordinal cost has a surprisingly low discriminative power. In theory, only 40 values can be distinguished with a 9×9 window. In practice, it appears to be about half. It would appear logical that a matching cost with such a low discriminability causes much more errors for increased disparity ranges. We have compared the Window method with the Ordinal cost on all data sets for the standard and the extended disparity ranges, and we found that the error is only marginally higher in case of using the extended disparity range of 256. The solution to this apparent contradiction is that the Ordinal cost compresses a wide range of mismatches to the same value. Thus, it discriminates only significant information. This is a good example that discriminative power is not necessarily correlated with performance.

To summarize, this test shows that the discriminability of the best-performing costs BilSub and Census is actually lowest. However, these costs benefit more from aggregation than other costs, which compensates the apparent drawback. Furthermore, the test shows that discriminative power is not necessarily correlated with performance.

F. Benefit of Color Matching

In all experiments up to now, we focused on one radiometric channel, i.e., intensity. In many applications, however, color images are available, and one might expect that utilizing color should increase matching performance. We therefore implemented the most promising costs for color, by applying them separately on the red, green and blue components. The final cost for a pixel is computed by summing the pixelwise costs over the color components. For BilSub we use the original definition [12] and compute the radiometric distance in CIELab color space. Figure 16 shows the comparison of intensity and color matching, separately for the training and test sets.

Surprisingly, it can be seen that using color results is little overall benefit. While there is a consistent (but small) improvement for the test images, color actually makes things worse for the training set in almost all cases. It appears that some of the training images, in particular Venus and Teddy, have non-uniform color variations that negatively affect the matching. The intensity-based costs in contrast seem to be robust to these color variations that are likely caused by color preprocessing done automatically by many consumer-grade cameras. Even on the (new) test images where such color distortions do not appear to be present, the performance gain for color is rather small. However, note that a dramatic benefit from using color could only be expected if locally disambiguating texture was lost when converting from color to intensity (grey) values. This appears quite unlikely in the real world, as most color changes also yield intensity changes. Thus, the benefit one would intuitively expect from using colors appears small in practice. In addition, note that (unless a multi-



(a) Window (b) SGM (c) GC

Fig. 16. Comparison of intensity and color matching on the training and test sets.

sensor camera is employed) each of the color channels has a lower effective resolution since it is interpolated from the Bayer pattern on the color sensor.

In summary, the potential benefit from using color information appears to be limited, and color might be less robust and more easily affected by the camera than intensity information. A deeper investigation into utilizing color for matching is beyond the scope of this paper, but is clearly an important topic for future research.

G. Comparison of Runtime

In addition to the qualitative and quantitative performance of different matching costs, the runtime can also be an important issue for different applications. We implemented all methods ourselves in C. We tried to make them efficient, but without putting too much effort into optimization. The runtime is measured on a 2.6 GHz Xeon CPU using the Teddy image pair, which has a size of 450×375 pixel and a disparity range of 64 pixels. The runtime includes reading the images and storing the costs in an array for all pixels and all disparities.

Table I lists all filters and matching costs that are suitable for pixelwise matching. The table shows the runtime for preprocessing both intensity input images, which depends on the number of pixels N, and for matching, which additionally depends on the disparity range D. The most simple and therefore fastest cost is AD. BT is much slower than AD, because it requires many comparisons in the innermost loop. The majority of the runtime for AD is actually used for storing the matching cost in the cost array, because it has a size of 450×375×64 integer values and is too large for the CPU cache. The creation of this array is required for global algorithms. Therefore, local, window based algorithms could be much faster. However, since the overhead is included in all measurements, we consider including it to be fair.

TABLE I
RUNTIME OF FILTERS AND PIXELWISE COSTS ON A 2.6GHZ XEON CPU FOR THE INTENSITY TEDDY IMAGE PAIR.

Method	Filter size	Preprocessing $O(N)$ C	Preprocessing $O(N)$ MMX	Matching $O(ND)$
AD	-	-	-	57 ms
BT	-	-	-	155 ms
Mean	15×15	150 ms		AD/BT
LoG	5×5	14 ms	10 ms	AD/BT
BilSub	15×15	281 ms		AD/BT
Rank	15×15	155 ms	29 ms	AD/BT
SoftRank	15×15	271 ms		AD/BT
Census	9×7	58 ms		110 ms
MI	-	10 ms		66 ms

The runtime of the alternate MMX implementations of the LoG and Rank filter shows that significant speedup is possible. However, it also shows that the performance gain depends heavily on the individual method. The same applies to hardware implementations. Real-time, hardware implementations have been reported for Rank and Census [52], [53], but it is unclear if other methods would benefit in the same way from a hardware implementation.

The runtime of all filters directly depends on the neighborhood size. A probably significant speed-up could be possible by recursive or separable implementations that update individual pixel or combine a horizontal and a vertical pass with 1-pixel wide windows. However, not all filters can be implemented in this way, e.g., the separable implementation of BilSub is only approximate, but real-time performance has been reported [11].

Computation of the Hamming distance for Census has been done by summing the results of 8-bit table lookups for the 64-bit values. MI appears very fast, because it has only to be computed once for each image pair. After the preparation, only a table lookup is required for getting the pixelwise matching cost. However, HMI needs to be computed hierarchically, which additionally increases the total runtime in case of Window and SGM by about 14%. The overhead of GC is lower, since its internal complexity is higher. Therefore, there is more benefit in the hierarchical processing, but the method itself is much slower.

Table II shows the time for computing the window based matching costs using a 9×9 window. The runtime can be significantly reduced and made independent of the window size W by using a recursive implementation, as reported in the literature [9], [10], [54], but we did not do that. For the ordinal measure, we tried an efficient implementation that sorts the intensities of both windows using quicksort, but maintains a linking between the original and sorted pixels for fast computation of the cost. A further significant speed-up is expected by a recursive implementation using a heap-tree as data structure [14].

TABLE II
RUNTIME OF WINDOW-BASED COST COMPUTATIONS ON A 2.6GHZ XEON CPU FOR THE INTENSITY TEDDY IMAGE PAIR.

Method	Matching $O(NDW)$
SAD	1.9 s
ZSAD	4.1 s
NCC	2.6 s
ZNCC	4.8 s
Ordinal	130.4 s

We give the runtimes of our implementations for showing the

differences in computation time of all matching costs. The actual runtimes should be taken with a grain of salt, since running the same code on different CPU architectures will not only scale all timings, but may change their relative sizes as well. Furthermore, some implementation tricks may increase the speed significantly. Nevertheless, the runtimes serve as upper bounds, and we feel that the order of the given runtimes reflects the expected computational burden of the individual methods.

VI. Conclusion

We have compared 15 different cost functions for stereo matching on images with simulated and real radiometric differences, and also on radiometrically "clean" images. Most costs were evaluated with three different stereo algorithms: a local correlation method, a semi-global matching method, and a global method using graph cuts. We found that the performance of matching cost functions can depend on the stereo method that uses it.

We identified four methods of particular interest. First, filtering with bilateral background subtraction (BilSub) followed by the sampling insensitive absolute difference performed in all experiments with all stereo algorithms as one of the best costs if radiometric changes are not too severe. While it only compensates for a local change of offset, it does not blur discontinuities as most other filters and costs do.

Second, for window-based matching, we found ZNCC to be better than BilSub in the case of strong radiometric changes, because ZNCC compensates for local gain and offset changes. However, it had the highest error of all costs at discontinuities, which makes BilSub to be more attractive if radiometric differences are expected to be moderate.

Third, Census performed very well throughout all experiments with simulated and real radiometric differences, except in the presence of strong image noise. Like all non-parametric matching costs, Census tolerates all radiometric distortions that do not change the local ordering of intensities. It was consistently better than ZNCC and in almost all cases better than BilSub.

Finally, we tested pixel-wise matching using Mutual Information, which was calculated hierarchically over the whole image (HMI). It compensates for complex global radiometric relations between the input images. It performed slightly better than Census in case of low radiometric changes and pixel-wise matching using the semi-global or global stereo method. It also performed best in case of strong image noise. However, HMI showed problems with large local radiometric differences, caused for example by the vignetting effect and by non-Lambertian surfaces and lighting changes. Promising directions for future research include creating local variants of MI that can handle such local changes.

We observed that costs that can compensate for strong radiometric changes do also well on images with little or no apparent radiometric changes. Thus, radiometrically tolerant matching costs are also useful in applications where large radiometric differences are not expected.

We also performed experiments to evaluate the variance of results and the importance of cost discriminability, and found that the cost performances are fairly independent of the scene and are not necessarily correlated with discriminative power.

We also investigated the potential benefit of using color information, which appears to be rather small, and in some cases color is even detrimental. This is clearly an important topic for future research.

In summary we found that BilSub performs consistently very well for low radiometric differences; HMI is slightly better as pixel-wise matching cost in some special cases and for strong image noise; and Census gives the best and most robust overall performance on all test sets with all stereo algorithms.

Acknowledgments

We would like to thank Anna Blasiak and Jeff Wehrwein for their help in creating the data sets used in this paper. We would also like to thank Larry Matthies for suggesting the BilSub cost and Larry Zitnick for suggesting the SoftRank transform. Support for this work was provided in part by NSF grant 0413169.

References

[1] "Daimler ag driving car stereo sequences." [Online]. Available: http://www.mi.auckland.ac.nz/EISATS/

[2] "Point grey – triclops stereo library." [Online]. Available: http://www.ptgrey.com/products/triclopsSDK/

[3] "Videre design – small vision system." [Online]. Available: http://www.videredesign.com/vision/svs.htm

[4] N. Campbell, G. Vogiatzis, C. Hernandez, and R. Cipolla, "Multiple hypotheses depth-maps for multi-view stereo," in *European Conference on Computer Vision*, Marseille, France, 2008.

[5] M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S. M. Seitz, "Multi-view stereo for community photo collections," in *International Conference on Computer Vision*, Rio, Brasil, October 2007.

[6] J.-P. Pons, R. Keriven, and O. Faugeras, "Multi-view stereo reconstruction and scene flow estimation with a global image-based matching score," *International Journal of Computer Vision*, vol. 72, no. 2, pp. 179–193, April 2007.

[7] S. Birchfield and C. Tomasi, "A pixel dissimilarity measure that is insensitive to image sampling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 4, pp. 401–406, April 1998.

[8] E. P. Baltsavias and D. Stallmann, "Spot stereo matching for digital terrain model generation," in *2nd Swiss Symposium on Pattern Recognition and Computer Vision*, 1993, pp. 61–72.

[9] K. Konolige, "Small vision systems: Hardware and implementation," in *Eighth International Symposium on Robotics Research*, Hayama, Japan, October 1997, pp. 203–212.

[10] H. Hirschmüller, P. R. Innocent, and J. M. Garibaldi, "Real-time correlation-based stereo vision with reduced border errors," *International Journal of Computer Vision*, vol. 47, no. 1/2/3, pp. 229–246, April-June 2002.

[11] A. Ansar, A. Castano, and L. Matthies, "Enhanced real-time stereo using bilateral filtering," in *3DPVT*, 2004.

[12] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," *IEEE International Conference on Computer Vision*, pp. 836–846, 1998.

[13] R. Zabih and J. Woodfill, "Non-parametric local transforms for computing visual correspondance," in *Proceedings of the European Conference of Computer Vision*, Stockholm, Sweden, May 1994, pp. 151–158.

[14] D. Bhat and S. Nayar, "Ordinal measures for image correspondence," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 20, no. 4, pp. 415–423, 1998.

[15] R. Sara and R. Bajcsy, "On occluding contour artifacts in stereo vision," in *IEEE Conference on Computer Vision and Pattern Recognition*, 1997.

[16] R. A. Gideon and R. A. Hollister, "A rank correlation coefficient," *Journal of the American Statistical Association*, vol. 82, pp. 656–666, 1987.

[17] P. Viola and W. M. Wells, "Alignment by maximization of mutual information," *International Journal of Computer Vision*, vol. 24, no. 2, pp. 137–154, 1997.

[18] R. Chrastek and J. Jan, "Mutual information as a matching criterion for stereo pairs of images," *Analysis of Biomedical Signals and Images*, vol. 14, pp. 101–103, 1998.

[19] G. Egnal, "Mutual information as a stereo correspondence measure," Computer and Information Science, University of Pennsylvania, Philadelphia, USA, Tech. Rep. MS-CIS-00-20, 2000.

[20] C. Fookes, M. Bennamoun, and A. Lamanna, "Improved stereo image matching using mutual information and hierarchical prior probabilities," in *IEEE International Conference on Pattern Recognition*, 2002.

[21] I. Sarkar and M. Bansal, "A wavelet-based multiresolution approach to solve the stereo correspondence problem using mutual information," *IEEE Trans. on Systems, Man, and Cybernetics - Part B: Cybernetics*, vol. 37, no. 4, August 2007.

[22] C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality video view interpolation using a layered representation," in *SIGGRAPH*, 2004.

[23] J. Kim, V. Kolmogorov, and R. Zabih, "Visual correspondence using energy minimization and mutual information," in *International Conference on Computer Vision*, October 2003.

[24] H. Hirschmüller, "Stereo processing by semi-global matching and mutual information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, February 2008.

[25] J. Zhang, L. McMillan, and J. Yu, "Robust tracking and stereo matching under variable illumination," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2006, pp. 871–878.

[26] L. Wang, R. Yang, and J. Davis, "BRDF invariant stereo using light transport constancy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 9, September 2007.

[27] A. Hertzmann and S. M. Seitz, "Example-based photometric stereo: Shape reconstruction with general, varying BRDFs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1254–1264, August 2005.

[28] H. Jin, S. Soatto, and A. J. Yezzi, "Multi-view stereo beyond Lambert," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, June 2003, pp. 171–178.

[29] Y. Li, S. Lin, and H. Li, "Multibaseline stereo in the presence of specular reflections," in *Proceedings of the International Conference on Pattern Recognition*, vol. 3, 2002.

[30] M. Liao, L. Wang, R. Yang, and M. Gong, "Light fall-off stereo," in *IEEE Conference on Computer Vision and Pattern Recognition*, Minneapolis, MN, USA, June 2007.

[31] T. E. Zickler, P. N. Belhumeur, and D. J. Krigman, "Helmholtz stereopsis: Exploiting reciprocity for surface reconstruction," *International Journal of Computer Vision*, vol. 49, no. 2-3, 2002.

[32] M. Z. Brown, D. Burschka, and G. D. Hager, "Advances in computational stereo," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 8, pp. 993–1008, August 2003.

[33] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, no. 1/2/3, pp. 7–42, April-June 2002.

[34] "Middlebury stereo website." [Online]. Available: http://vision.middlebury.edu/stereo/

[35] L. Wang, M. Gong, M. Gong, and R. Yang, "How far can we go with local optimization in real-time stereo matching," in *Third International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT)*, 2006.

[36] S. Gautama, S. Lacroix, and M. Devy, "Evaluation of stereo matching algorithms for occupant detection," *International Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems*, p. 177, 1999.

[37] J. Banks and P. Corke, "Quantitative evaluation of matching methods and validity measures for stereo vision," *International Journal of Robotics Research*, vol. 20, no. 7, pp. 512–532, July 2001.

[38] C. Fookes, A. Maeder, S. Sridharan, and J. Cook, "Multi-spectral stereo matching using mutual information," in *Proceedings of the International Symposium of 3D Data Processing, Visualization and Transmission*, 2004, pp. 961–968.

[39] K.-J. Yoon and I.-S. Kweon, "Adaptive support-weight approach for correspondence search," *IEEE Transactions on Pattern Matching and Machine Intelligence*, vol. 28, no. 4, pp. 650–656, 2006.

[40] J. Cech and R. Sara, "Complex correlation statistic for dense stereoscopic vision," in *SCIA*, 2005, pp. 598–608.

[41] H. Moravec, "Toward automatic visual obstacle avoidance," in *Proceedings of the Fifth International Joint Conference on Artificial Intelligence*, Cambridge, MA, August 1977, pp. 584–590.

[42] C. L. Zitnick, personal communication.

[43] D. N. Bhat, S. K. Nayar, and A. Gupta, "Motion estimation using ordinal measures," in *IEEE Conference on Computer Vision and Pattern Recognition*, 1997.

[44] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1222–1239, 2001.

[45] H. Hirschmüller, "Stereo vision based mapping and immediate virtual walkthroughs," Ph.D. dissertation, School of Computing, De Montfort University, Leicester, UK, June 2003.

[46] A. F. Bobick and S. S. Intille, "Large occlusion stereo," *International Journal of Computer Vision*, vol. 33, no. 3, pp. 181–200, September 1999.

[47] I. J. Cox, S. L. Hingorani, S. B. Rao, and B. M. Maggs, "A maximum likelihood stereo algorithm," *Computer Vision and Image Understanding*, vol. 63, no. 3, pp. 542–567, may 1996.

[48] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1124–1137, September 2004.

[49] V. Kolmogorov and R. Zabih, "What energy functions can be minimized via graph cuts?" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 2, pp. 147–159, February 2004.

[50] R. Szeliski, E. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agrawala, M. Tappen, and C. Rother, "A comparative study of energy minimization methods for markov random fields with smoothness-based priors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 6, pp. 1068–1080, June 2008.

[51] D. Scharstein and R. Szeliski, "High-accuracy stereo depth maps using structured light," in *IEEE Conference for Computer Vision and Pattern Recognition*, vol. 1, Madison, Winsconsin, USA, June 2003, pp. 195–202.

[52] J. Woodfill and B. von Herzen, "Real-time stereo vision on the parts reconfigurable computer," in *Proceedings of the 5th IEEE Symposium on FPGAs for Custom Computing Machines*, Napa Valley, CA, USA, April 1997, pp. 201–210.

[53] P. I. Corke, P. A. Dunn, and J. E. Banks, "Frame-rate stereopsis using non-parametric transforms and programmable logic," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 3, Detroit, USA, May 1999, pp. 1928–1933.

[54] O. Faugeras, B. Hotz, H. Mathieu, T. Vieville, Z. Zhang, P. Fua, E. Thron, L. Moll, G. Berry, J. Vuillemin, P. Bertin, and C. Proy, "Real time correlation-based stereo: algorithm, implementations and application," INRIA, France, Tech. Rep. 2013, August 1993.

Heiko Hirschmüller received his Dipl.-Inform. (FH) degree at Fachhochschule Mannheim, Germany in 1996, M.Sc. in Human Computer Systems at De Montfort University Leicester, UK in 1997 and Ph.D. on real time stereo vision at De Montfort University Leicester, UK in 2003. From 1997 to 2000 he worked as software engineer at Siemens Mannheim, Germany. Since 2003 he has been a computer vision researcher at the Institute of Robotics and Mechatronics of the German Aerospace Center (DLR) in Oberpfaffenhofen near Munich. His research interests include stereo vision and 3D reconstruction from multiple images.

Daniel Scharstein studied computer science at the Universität Karlsruhe, Germany, and received the PhD degree from Cornell University in 1997. He is an associate professor of computer science at Middlebury College, Vermont. His research interests include computer vision, particularly stereo vision, and robotics. He maintains online stereo vision and optical flow benchmarks at http://vision.middlebury.edu. He is a member of the IEEE and the IEEE Computer Society.