A Methodology for Evaluating Illumination Artifact Removal for Corresponding Images

Tobi Vaudrey[1], Andreas Wedel[2], and Reinhard Klette[1]

[1] The *.enpeda..* Project, The University of Auckland, Auckland, New Zealand
[2] Daimler Research, Daimler AG, Stuttgart, Germany

Abstract. Robust stereo and optical flow disparity matching is essential for computer vision applications with varying illumination conditions. Most robust disparity matching algorithms rely on computationally expensive normalized variants of the brightness constancy assumption to compute the matching criterion. In this paper, we reinvestigate the removal of global and large area illumination artifacts, such as vignetting, camera gain, and shading reflections, by directly modifying the input images. We show that this significantly reduces violations of the brightness constancy assumption, while maintaining the information content in the images. In particular, we define metrics and perform a methodical evaluation to firstly identify the loss of information in the images, and secondly determine the reduction of brightness constancy violations. Thirdly, we experimentally validate that modifying the input images yields robustness against illumination artifacts for optical flow disparity matching.

1 Introduction

Previous studies have shown that when using correspondence algorithms (i.e., stereo and optical flow) to provide reliable information, the results on synthetically generated data (e.g., [12]) do not compare well with results on realistic images [18]. Further studies have shown that illumination artifacts (such as shadows, reflections, and vignetting) and differing exposures have the worst effect on the matching [14]. This effect is especially highlighted in driver assistance systems (DAS), where illumination can change drastically in a short amount of time (e.g., going through a tunnel, or the "dancing light" from sunlight through trees).

For dealing with illumination artifacts, there are three basic approaches: simultaneously estimate the disparity matching and model brightness change within the disparity estimation [7], try to map both images into a uniform illumination model, or map the intensity images into images which carry the illumination-independent information (e.g., using colour images [11, 20]).

Using the first option, only reflection artifacts can be modelled without major computational expense. From experiments with various unifying mappings, the second option is basically impossible (or, at least, a very big challenge). The third approach has more merit for research; we restrain our study to using the more common grey value images.

An example of mapping intensity images into illumination-independent images is the structure-texture image decomposition [2, 15] (an example can be seen in Figure 1).

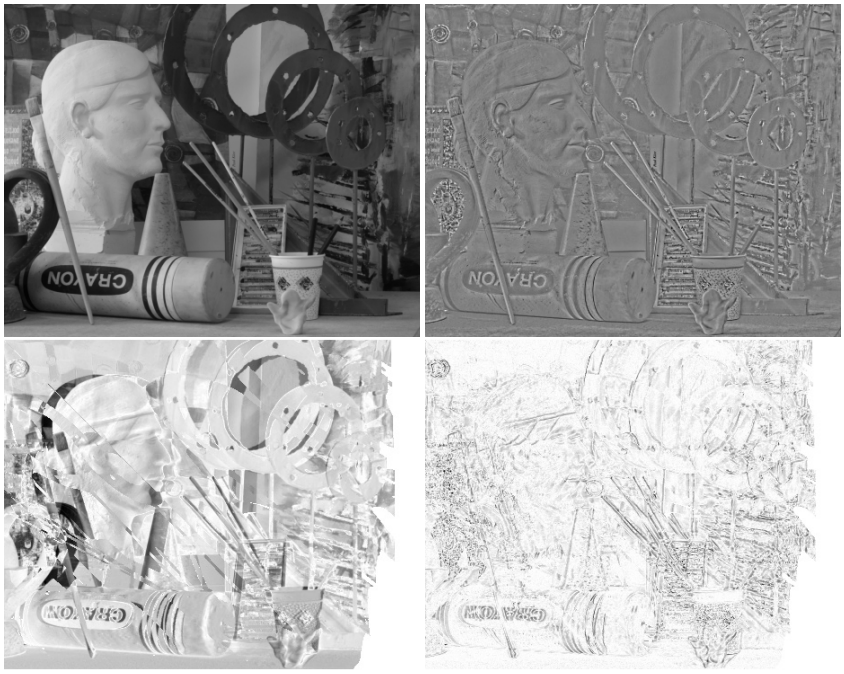

Fig. 1. Example for removing illumination artifacts due to different camera exposure in the *Art* image (left) by using its residual component (2nd from left). The brightness difference between the plain intensity images (3rd from left) shows laminar errors. The brightness difference of the residual images (right) contains spatially distributed noise but no large area illumination artifacts.

More formally, this is the concept of *residuals* [9], which is the difference between an intensity image and a smoothed version of itself. A subset of residual operators has been recently evaluated together with different matching costs in the context of stereo disparity matching in [8]. In this paper we systematically evaluate and compare residual operators as basic approach for preprocessing corresponding images, to reduce the effect of illumination variances.

The methodology is based on first showing that information is not lost by applying the filter, using co-occurrence matrix [6] based measures. We then show the effect on the corresponding images (using ground truth correspondence data), comparing the differences in illumination, and summarise this information with an error metric. We go on to show that they remove illumination artifacts using a mixture of synthetic and real-life images [5, 12]. The illumination effects are highlighted more drastically when the illumination and exposure conditions of the corresponding images are not the same. The chosen filters are the TV-L^2 [15], median, mean, sigma [10], bilateral [17], and trilateral filter [4]. All are effectively "edge preserving" filters, except the mean filter.

2 Methodology

Here we define the methodology of our process. It is defined by two parts; firstly, identifying if the images loose information, and secondly, determining reduction of the effect of illumination artifacts.

2.1 Co-occurrence Matrix and Metrics

The co-occurrence matrix has been defined for analysing different metrics about the texture of an image [6]:

$$C(i,j) = \sum_{\mathbf{x}\in\Omega} \sum_{\mathbf{a}\in\mathcal{N}\setminus\{(0,0)\}} \begin{cases} 1, & \text{if } h(\mathbf{x}) = i \text{ and } h(\mathbf{x}+\mathbf{a}) = j \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where $\mathcal{N}+\mathbf{x}$ is the neighbourhood of pixel $\mathbf{x}$, $\mathbf{a} \neq (0,0)$ is one of the offsets in $\mathcal{N}$, and $0 \leq i,j \leq I_{\max}$, for maximum intensity $I_{\max}$. h represents any 2D image (e.g., f). All images are scaled $\min \leftrightarrow \max$ for utilizing the full $0 \leftrightarrow I_{\max}$ scale.

In our experiments we chose $\mathcal{N}$ to be the 4-neighbourhood, and we have $I_{\max} = 255$. The loss in information is identified by the following metrics:

$$\textbf{Homogeneity:} \quad T_{homo}(h) = \sum_{i\,j} \frac{C(i,j)}{1+|i-j|} \tag{2}$$

$$\textbf{Uniformity:} \quad T_{uni}(h) = \sum_{i\,j} C(i,j)^2 \tag{3}$$

$$\textbf{Entropy:} \quad T_{ent}(h) = -\sum_{i\,j} C(i,j)\,\ln C(i,j) \tag{4}$$

An increase in homogeneity represents the image having more homogeneous areas, an increase in uniformity represents more uniform areas, and a decrease in entropy shows that there is less information contained in the image. To get a better representation of the effect of filters, we scale the result by the original image's metric, i.e., $T_*(h)/\,|T_*(f)|$, where h is the processed image (obviously, $h = f$ gives a value of 1). Results using these metrics on the smoothed images and residual images can be seen in Section 5.

2.2 Testing Illumination Artifact Reduction

Correspondence algorithms usually rely on the brightness consistency assumption, i.e., that the appearance of an object (according to illumination) does not change between the corresponding images. However, this does not hold true when using real-world images, this is due to, for example, shadows, reflections, differing exposures and sensor noise. It is well known, that illumination artifacts propose the biggest problem for correspondence algorithms; a recent study has shown that illumination artifacts may, in fact, be the worse type of errors [14]. Figure 2 shows our proposed approach for evaluating the effectiveness of a filter. In this paper, we choose the filtering operator H to be the residual image ($H = R$).

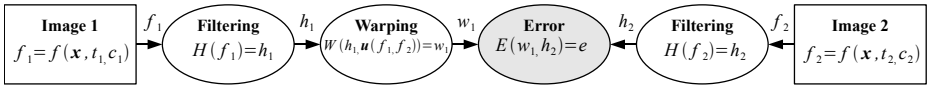

Fig. 2. Outline of the methodology used to obtain an error image. For our study $H = R$.

2.3 Image Warping

One way to highlight this (i.e., that the errors from residual images are lower than the errors obtained using the original images) is to warp one image to the perspective of the other (using ground truth) and compare the differences. The forward warping function W is defined by the following:

$$W\Big(h_1(\mathbf{x}), \mathbf{u}^*(\mathbf{x}, h_1, h_2)\Big) = w\big(\mathbf{x} + \mathbf{u}^*(\mathbf{x}, h_1, h_2)\big)\,,$$

where $h(\mathbf{x})$ is the image value at $\mathbf{x} \in \Omega$, and $\mathbf{u}^*$ is the 2D ground truth warping (remapping) vector from h_1 to the perspective of h_2. In practice, the warping is performed using a lookup table with interpolation (e.g., bilinear or cubic). In the stereo case, $\mathbf{u}^*$ is the ground truth disparity map from left to right (all vertical translations would be zero). Another common example is optical flow, where $\mathbf{u}^*$ is the ground truth flow field from the previous to the current frame.

2.4 Image Scaling

For the purposes of this paper, h is discrete in the functional inputs ($\mathbf{x}$), but continuous for the value of h itself. For a typical grey-scale image, the information is discrete ($0 \leq h \leq 2^n - 1 \in \mathbb{N}^2$, where n is usually 8 or 16). However, we find it easier to represent image data continuously by $-1 \leq h \leq 1 \in \mathbb{Q}^2$, which takes away the ambiguity for the bits per pixel (as any n-bits per pixel image can be scaled to this domain). We scale all images to this domain using $h(\mathbf{x}) = h(\mathbf{x}) / \max_{\mathbf{x}\in\Omega} |h(\mathbf{x})|$.

2.5 Error Images and Metrics

An error image e is the magnitude of difference between two images, $E(h, h^*) = e(\mathbf{x}) = \| \mathbf{h}(\mathbf{x}) - \mathbf{h}^*(\mathbf{x}) \|$, where, usually, $\mathbf{h}$ is the result of a process and $\mathbf{h}^*$ is the ground truth. For this paper, the error image is between $\mathbf{h}^* = h_2$ and the warped image $\mathbf{h} = W(h_1)$.

A common error metric is the *Root Mean Squared* (RMS) *Error*. The problem with this metric is that it gives an even weighting to all pixels, no matter the proximity to other errors. In practice, if errors are happening in the same proximity, this is much worse than if the errors are randomly placed over an image. Most algorithms can handle (by denoising or such approaches) small amounts of error, but if the error is all in the same area, this is seen as signal. We define the *Spatial Root Mean Squared Error*

(Spatial-RMS) to take the spatial properties of the error into account:

$$RMS_S(e) = \sqrt{\frac{1}{M} \sum_{\mathbf{x} \in \Omega} \Big(G\big(e(\mathbf{x})\big)^2 \Big)} \tag{5}$$

M is the number of pixels in the (discrete) non-occluded (when occlusion maps are available) image domain Ω, and G is a function that propagates the errors in a local neighbourhood $\mathcal{N}$. For our experiments, we chose a Gaussian error propagation using a standard deviation $\sigma = 1$.

3 Smoothing Operators and Residuals

Let f be any frame of a given image sequence (or stereo camera setup), defined on a rectangular open set Ω and sampled at regular grid points within Ω.

f can be defined to have an additive decomposition $f(\mathbf{x}) = s(\mathbf{x}) + r(\mathbf{x})$, for all pixel positions $\mathbf{x} = (x, y)$, where $s = S(f)$ denotes the *smooth component* (of an image) and $r = R(f) = f - S(f)$ the *residual* (Figure 1 shows an example of the decomposition). We use the straightforward iteration scheme:

$$s^{(0)} = f, \quad s^{(n+1)} = S(s^{(n)}), \quad r^{(n+1)} = f - s^{(n+1)}, \quad \text{for } n \geq 0.$$

The concept of residual images was already introduced in [9] by using a 3×3 mean for implementing S. We use the $m \times m$ mean operator and also an $m \times m$ median operator in this study. The other operators for S are defined below.

3.1 TV-L^2 filter

[15] assumed an additive decomposition $f = s + r$ into a *smooth component* s and a *residual component* r, where s is assumed to be in $L^1(\Omega)$ with bounded TV (in brief: $s \in$ BV), and r is in $L^2(\Omega)$. This allows one to consider the minimization of the following functional:

$$\inf_{(s,r) \in BV \times L^2 \wedge f = s + r} \left(\int_\Omega |\nabla s| + \lambda ||r||^2_{L^2} \right) \tag{6}$$

The TV-L^2 approach in [15] was approximating this minimum numerically for identifying the "desired clean image" s and "additive noise" r. Further studies (see [2]) identified s to be the "structure", and r to be the "texture". See Figure 1. The concept may be generalized as follows: any *smoothing operator* S generates a *smoothed image* $s = S(f)$ and a *residuum* $r = f - S(f)$. For example, TV-L^2 generates the smoothed image $s = S_{TV}(f)$ by solving Equ. (6).

3.2 Sigma filter

This operator [10] is effectively a trimmed mean filter; it uses an $m \times m$ window, but only calculates the mean for all pixels with values in $[a - \sigma_f, a + \sigma_f]$, where a is the central pixel value and σ_f is a threshold. We chose σ_f to be the standard deviation of f (to reduce parameters for the filter).

3.3 Bilateral filter

This edge-preserving Gaussian filter [17] is used in the spatial domain (using σ_2 as spatial σ), also considering changes in the colour domain (e.g., object boundaries). In this case, offset vectors $\mathbf{a}$ and position-dependent real weights $d_1(\mathbf{a})$ define a local convolution, and the weights $d_1(\mathbf{a})$ are further scaled by a second weight function d_2, defined on the differences $f(\mathbf{x}+\mathbf{a}) - f(\mathbf{x})$:

$$s(\mathbf{x}) = \frac{1}{k(\mathbf{x})} \int_\Omega f(\mathbf{x}+\mathbf{a}) \cdot d_1(\mathbf{a}) \cdot d_2\left[f(\mathbf{x}+\mathbf{a}) - f(\mathbf{x})\right] \,\mathrm{d}\mathbf{a} \tag{7}$$

$$k(\mathbf{x}) = \int_\Omega d_1(\mathbf{a}) \cdot d_2\left[f(\mathbf{x}+\mathbf{a}) - f(\mathbf{x})\right] \,\mathrm{d}\mathbf{a}$$

Function $k(\mathbf{x})$ is used for normalization. In this paper, weights d_1 and d_2 are defined by Gaussian functions with standard deviations σ_1 and σ_2, respectively. The smoothed function s equals $S_{BL}(f)$. It therefore only takes into consideration values within a Gaussian kernel (σ_2 for spatial domain, f for kernel size) within the colour domain (σ_1 as colour σ).

3.4 Trilateral filter

This gradient-preserving smoothing operator [4] (i.e., it uses the local gradient plane to smooth the image) only requires the specification of one parameter σ_1, which is equivalent to the spatial kernel size. The rest of the parameters are self tuning.

It combines two bilateral filters to produce this effect. At first, a bilateral filter is applied on the derivatives of f (i.e., the gradients):

$$g_f(\mathbf{x}) = \frac{1}{k_\nabla(\mathbf{x})} \int_\Omega \nabla f(\mathbf{x}+\mathbf{a}) \cdot d_1(\mathbf{a}) \cdot d_2\left(||\nabla f(\mathbf{x}+\mathbf{a}) - \nabla f(\mathbf{x})||\right) \,\mathrm{d}\mathbf{a} \tag{8}$$

$$k_\nabla(\mathbf{x}) = \int_\Omega d_1(\mathbf{a}) \cdot d_2\left(||\nabla f(\mathbf{x}+\mathbf{a}) - \nabla f(\mathbf{x})||\right) \,\mathrm{d}\mathbf{a}$$

Simple forward differences $\nabla f(x,y) \approx (f(x+1,y)-f(x,y), f(x,y+1)-f(x,y))$ are used for the digital image. For the subsequent second bilateral filter, [4] suggested the use of the smoothed gradient $g_f(\mathbf{x})$ [instead of $\nabla f(\mathbf{x})$] for estimating an approximating plane $p_f(\mathbf{x},\mathbf{a}) = f(\mathbf{x}) + g_f(\mathbf{x}) \cdot \mathbf{a}$ Let $f_\triangle(\mathbf{x},\mathbf{a}) = f(\mathbf{x}+\mathbf{a}) - p_f(\mathbf{x},\mathbf{a})$. Furthermore, a neighbourhood function

$$n(\mathbf{x},\mathbf{a}) = \begin{cases} 1 & \text{if } ||g_f(\mathbf{x}+\mathbf{a}) - g_f(\mathbf{x})|| < A \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

is used for the second weighting. A specifies the adaptive region and is discussed further below. Finally,

$$s(\mathbf{x}) = f(\mathbf{x}) + \frac{1}{k_\triangle(\mathbf{x})} \int_\Omega f_\triangle(\mathbf{x},\mathbf{a}) \cdot d_1(\mathbf{a}) \cdot d_2(f_\triangle(\mathbf{x},\mathbf{a})) \cdot n(\mathbf{x},\mathbf{a}) \,\mathrm{d}\mathbf{a} \tag{10}$$

$$k_\triangle(\mathbf{x}) = \int_\Omega d_1(\mathbf{a}) \cdot d_2(f_\triangle(\mathbf{x},\mathbf{a})) \cdot n(\mathbf{x},\mathbf{a}) \,\mathrm{d}\mathbf{a}$$

The smoothed function s equals $S_{TL}(f)$. Again, d_1 and d_2 are assumed to be Gaussian functions, with standard deviations σ_1 and σ_2, respectively. The method requires specification of parameter σ_1 only, which is at first used to be the radius of circular neighbourhoods at $\mathbf{x}$ in f; let $\overline{g}_f(\mathbf{x})$ be the mean gradient of f in such a neighbourhood. Let

$$\sigma_2 = 0.15 \cdot || \max_{\mathbf{x} \in \Omega} \overline{g}_f(\mathbf{x}) - \min_{\mathbf{x} \in \Omega} \overline{g}_f(\mathbf{x}) || \tag{11}$$

(Value 0.15 was recommended in [4]). Finally, also use $A = \sigma_2$.

3.5 Numerical Implementation

All filters have been implemented in OpenCV, where possible the native function was used. For the TV-L^2, we use an implementation (with identical parameters) as in [19]. All other filters used are virtually parameterless (except a window size) and we use a window size of $m = 3$ ($\sigma_1 = 3$ for trilateral filter[3]). For the bilateral filter, we use color standard deviation $\sigma_1 = I_r/10$, where I_r is the range of the intensity values (i.e., $\sigma_1 = 0.2$ for the scaled images).

4 Datasets

We illustrate our arguments with the Middlebury dataset [12] and the EISATS [5] synthetic data (Set 2).

4.1 Middlebury Stereo Dataset

This highlights the major importance of removing illumination artifacts. For the Middlebury dataset we include both the2005 and 2006 datasets (provided by [8, 16]). This data has 3 different exposures and 3 different illuminations (for both the left and right images). This enables us to test the brightness consistency assumption under extreme conditions. Again, we only use images with ground truth available. For the 2005 set, that includes: *Art* (see Figure 1), *Books, Dolls, Laundry, Moebius,* and *Reindeer*. For the 2006 set: *Aloe, Baby1-3, Bowling1-2, Cloth1-4, Flowerpots, Lampshade1-2, Midd1-2, Monopoly, Plastic, Rocks1-2,* and *Wood1-2*. We are not interested in "good quality" situations. Therefore, we only use images with differing exposure and illumination. To do this, for each image pair, we keep the left image with illumination $= 1$ and exposure $= 0$ (as defined by [12]). But for the right image, we make use of all all the differing illumination $(1, 2, 3)$ and exposure $(0, 1, 2)$ settings (excluding the exact same illumination $= 1$ and exposure $= 0$). This is a total of 8 different illumination/exposure combinations, for each image pair. That brings the total dataset to 216 (27×8).

[3] The authours thank Prasun Choudhury (Adobe Systems, Inc.) and Jack Tumblin (EECS, Northwestern University), for their implementation of the trilateral filter.

Fig. 3. Example frames from EISATS scene. Frame 1 (left) and 2 (middle) are shown with ground truth flow and key (HSV circle for direction, saturation for vector length) shown on the right.

4.2 EISATS Synthetic Dataset

This dataset was made public in [18] for Set 2. We are only interested in bad illumination conditions. We therefore use the altered the data to resemble illumination differences in time, as performed in [14]; the differences start high between frames, then go to zero at frame 50, then increase again. For all t (frame number) we alter the original image f using a constant brightness. For all $\mathbf{x}$ we use $f(\mathbf{x}) = f(\mathbf{x}) + c$. The constant brightness change is defined by:

$$\begin{aligned} \text{Even values of } t: &\quad c = t - 52 \\ \text{Odd values of } t: &\quad c = 51 - t \end{aligned}$$

An example of the data used can be seen in Figure 3.

5 Experimental Results

A previous study, has already pointed out that the results for slight illumination artifacts are improved using residual images [1]. We now show that these results get even better when illumination is a major issue (not just a minor one).

5.1 Co-occurrence Metrics

This subsection demonstrates that the important information for correspondence algorithms is contained in the residual image r. The residual image is, in fact, an approximation of the high frequencies of the image, and the smoothed image s is an approximation of a low-pass filter. Obviously, by iteratively running a smoothing filter, you will get a more and more smoothed image (i.e., you will be getting lower and lower frequencies, thus reducing the higher frequencies). In [1] the metrics were shown to represent this effect accurately.

The residual of an image is an approximation of the high frequencies of the image, so the information should not be reduced. We average the co-occurrence metric results over dataset [12] to highlight this (Figure 4). These graphs shows that the residual images do not lose information, in fact the homogeneity is slightly reduced (except for the trilateral and median filter). The increase in information could be seen as noise from the filter, or an increase in emphasis of the high-frequencies.

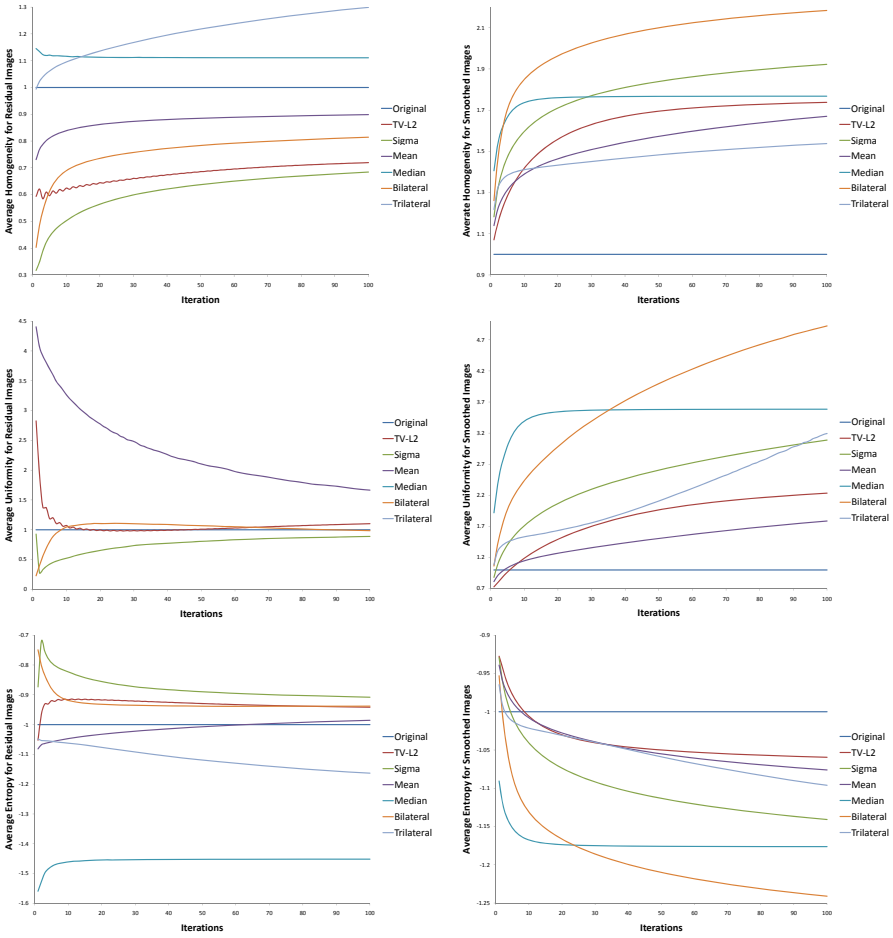

Fig. 4. Top to bottom: Average scaled homogeniety, uniformity and entropy metrics over all entire Middlebury dataset. Residual image (left) and smoothed image results (right) are shown, highlighting the effect of repeated iterations. The uniformity graph for residual images does not show median or trilateral filter as they are off the scale (trilateral $32 < 112$, median $39 < 66$). It is obvious that smoothing the images reduces information (smoothed image graphs), but the residual images seem to retain a lot of the information.

5.2 Illumination Differences

This subsection uses again the dataset [12]. A qualitative example of error images e can be seen in Figure 1. This specific error image is generated using the *Art* right image, with illumination and exposure both equal to 1 (left image is 1 and 0, respectively). The image is from [8], and has ground truth available (warping from left to right). The original error image (left) clearly shows how increasing the exposure (250 to 1000 ms)

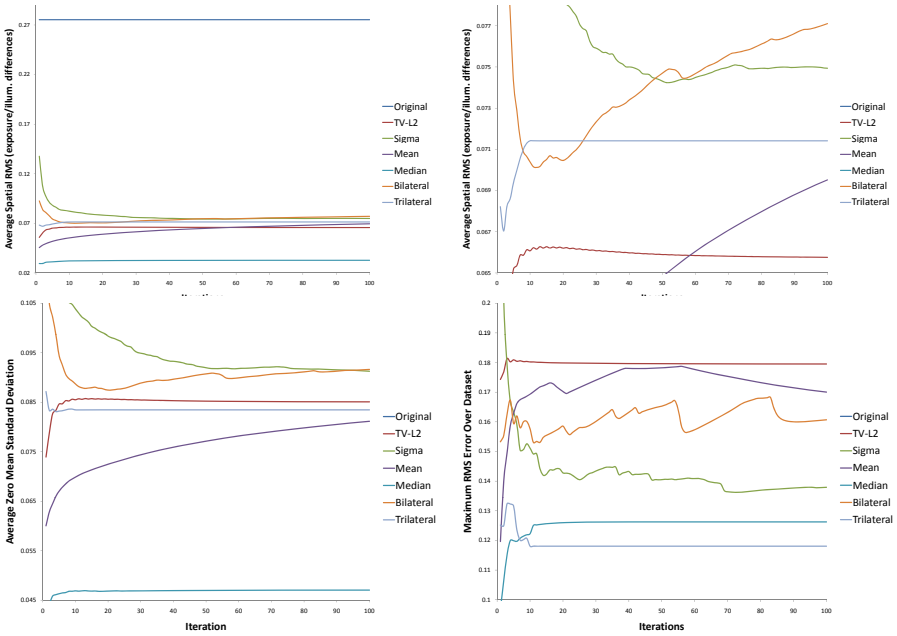

Fig. 5. Top Left: Average Spatial-RMS of intensity difference for f_1 and $W(f_2)$. Notice the huge benefit from using residual images. Top Right: zooming in on the stabalised results.The TV-L^2 and Mean filter seem to be the best. The trilateral filter is next (but this could be due to information loss). Closely followed by the sigma and bilateral filter. The median does have a lower RMS, but this is most probably due to the loss in information. Bottom Left: zero-mean standard deviation of Spatial-RMS over all results (original image = 0.31, much higher than the rest). Bottom Right: maximum Spatial-RMS for all dataset (original image = 0.64).

has very big consequences on the illumination differences between the left and right image. The error image using the TV-L^2 residual (right) reduces the error dramatically. Furthermore the magnitudes of the maximum errors are less; the original image is 1.83 and the TV-L^2 residual image is 1.25.

The results for the Spatial-RMS metric are shown in Figure 5. The trilateral filter was stopped at iteration 10. It is immediately obvious that the original images are far worse than residual images, around 3 times worse on average. This again highlights that with extremely different exposures and illuminations, the residual images provide the best information for matching.

The maximum and zero-mean standard deviation (square root of the sum of squared RMS error) of all Spatial-RMS values are also shown in Figure 5. The zero-mean standard deviation (ZMSD) aims to highlight how robust a filter is over the entire dataset. A low value indicates that more results (considering we are using 89 image pairs) are lower, compared to another algorithm. The maximum shows the extremes. The ZMSD is a better measure of robustness, as the maximum value could be an outlier. The original image is far worse than any of the residual images, so this is ignored here

# Its.	1			50			100			Time / iteration		Rank	
Filter	Ave.	S.D.	Max.	Ave.	S.D.	Max.	Ave.	S.D.	Max.	470×370	752×480	T_{homo}	RMS
Original	0.282	0.313	0.637	0.282	0.313	0.637	0.282	0.313	0.637	-	-	5	7
TV-L2	0.068	0.074	0.174	0.080	0.085	0.180	0.080	0.085	0.180	30	60	2	2
Sigma	0.168	0.172	0.267	0.089	0.092	0.141	0.089	0.091	0.138	30	100	1	6
Mean	0.055	0.060	0.120	0.073	0.077	0.178	0.077	0.081	0.170	1	2	4	3
Median	0.039	0.044	0.097	0.041	0.047	0.126	0.041	0.047	0.126	7	15	6	1
Bilateral	0.113	0.115	0.153	0.087	0.091	0.165	0.088	0.092	0.161	160	340	3	5
Trilateral	0.085	0.087	0.125	0.082	0.083	0.118	0.082	0.083	0.118	5000	11000	7	4

Table 1. Average (Ave.), zero-mean standard deviation (S.D.), and maximum (Max.) for the methodology performed on dataset [12] at different iterations for $r^{(n)}$ (Note: trilateral stopped at 10 iterations). Average running times per iteration are also included (right) for two image resolutions. The rank of the filters is also given for both evaluations.

for the comparison, but is a very important result. The median filter has the lowest results, but we can not be sure if this is from information loss. The trilateral filter was stopped at 10 iterations, and contains more information than the median filter (see Figure 4); it boasts quite good results. Although the TV-L^2 filter has the best average, it has the highest maximum and a high ZMSD. The bilateral and sigma filter both have small maximum errors, but have the highest ZMSD. The mean filter appears to have good results all around.

We have presented statistical results of the RMS after 1, 50, and 100 iterations (slices of the graphs in Figure 5). These results are shown in Table 1. You can see from these results that all the statistics for the original images are higher than any of the filters. The mean, trilateral, median, and, TV-L^2 filter seem to be the most robust; showing the lowest ZMSD. The mean, TV-L^2 and median filters have the best average. The timing information provided in this table is the average time per iteration, on two sizes of images (470×370, 752×480 pixel resolution), this is to highlight the scalability of the filters. The tests were under Windows, the CPU was an Intel Core 2 Duo 3G Hz (multi-core processing not exploited), with 4GB memory.

5.3 Optical Flow on EISATS Dataset

One of the most influential evaluations of optical flow in recent years is from Middlbury Vision Group [3]. This dataset is used to evaluate optical flow, in relatively simple situations. To highlight the effect of using residual images, we used a high ranking (see [13]) optical flow technique called TV-L^1 optical flow [21]. The results for optical flow were analysed on the EISATS dataset [5]; see [18] for Set 2. Section 4 has full details of data used. Numerical details of implementation are given in [19]. The specific parameters used were:

Smoothness:	35	Number of pyramid levels:	10
Duality threshold θ:	0.2	Number of iterations per level:	5
TV step size:	0.25	Number of warps per iteration:	25

The flow field is computed using $U(h_1, h_2) = \mathbf{u}$. This is to show that a residual image r provides better data for matching, than for the original image f. We computed

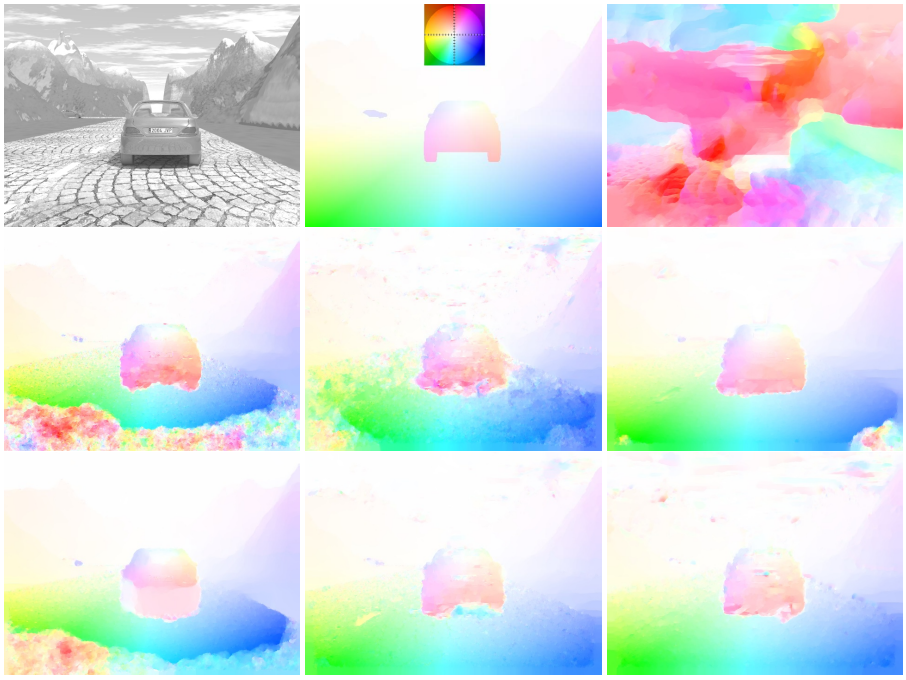

Fig. 6. Sample optical flow results on EISATS scene. Top row: sample original image, ground truth, and results using original images (respectively). Middle row (left to right): TV-L^2, sigma, and mean filter residual image optical flow results. Bottom row (left to right): median, bilateral, and trilateral filter residual image optical flow results.

the flow using $U(r_1^{(n)}, r_2^{(n)})$ with $n = 1, 2, 10, 50, 70$, and 100 to show how each filter behaves. The results are compared to optical flow on the original images $U(f_1, f_2)$. Figure 6 shows an example of this effect, obviously the residual image vastly improves optical flow results. In fact, the original image results are so noisy that they can not be used.

To compare the results numerically, we calculated the end-point-error using the error image $e = E(\mathbf{u}, \mathbf{u}^*)$ and Spatial-RMS (u^* is ground truth optical flow). The results can be seen in Figure 7. The zoomed out graph highlights that the results for the original image are unusable. The shape of the graph is appropriate as well, because the difference between intensities of the images gets closer together near the middle of the sequence, and further away near the end. The rest of the graphs show a zoomed in version showing Spatial-RMS values between 0 and 10. When $n = 1$ the results are vastly improved in general, except near the middle of the sequence where illumination differences are smaller. This shows that although the residual images after 1 iteration are only good in extreme cases. Looking at the $n = 10$ graph highlights some of the major impacts of using residual images. The trilateral filter and the bilateral filter beat the original images in almost every frame. Note that the trilateral filter was stopped at

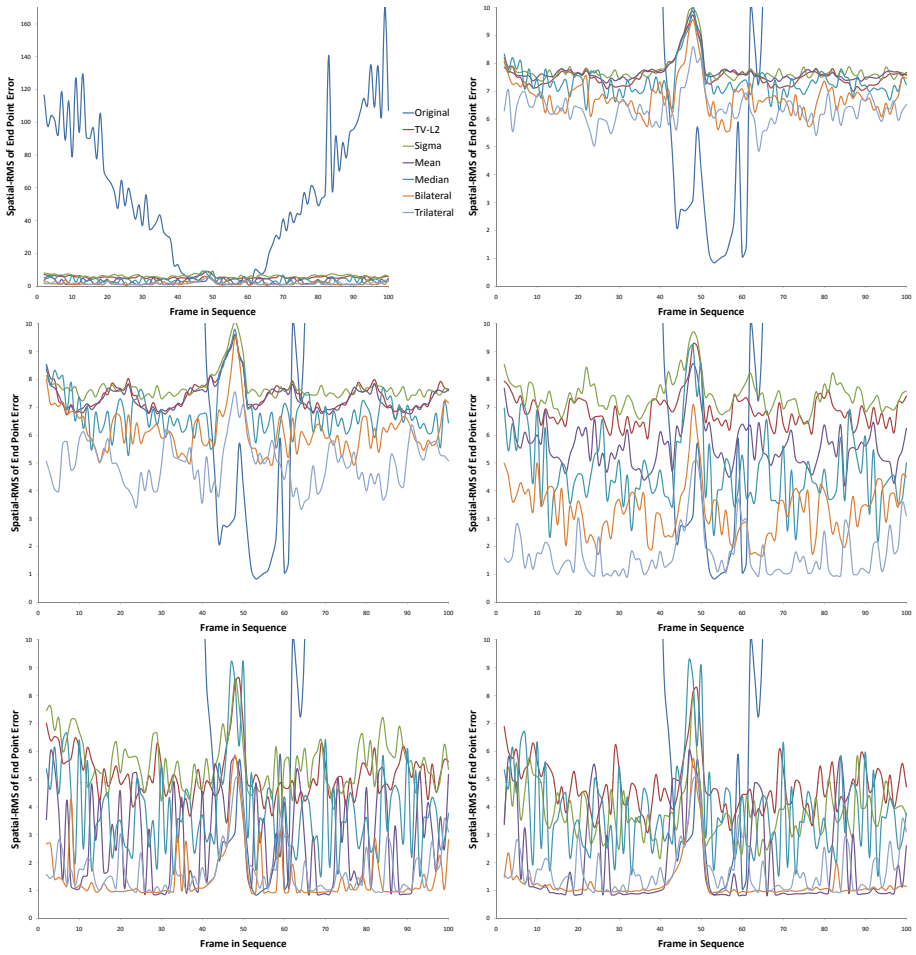

Fig. 7. Spatial-RMS results over entire EISATS sequence. Different numbers of filter iterations $r^{(n)}$ are shown. Top row: results for $n = 40$ zoomed out to highlight the poor quality of using original images (left) and the zoomed in version for $n = 1$ (right). Middle row: $n = 2$ (left) and $n = 10$ (right). Bottom row: $n = 50$ (left) and $n = 100$ (right).

$n = 10$ due to the long computation time. The $n = 50$ graph shows a lot of overlap in this situation, and $n = 100$ shows a more stable result. The point to note is that different filters provide different results at different numbers of iterations.

A major point to highlight is that at different frames in the sequence, there are different rankings for the filters. If you look, for example, at the $n = 100$ graph at frame 25, the rank is (best to worst): trilateral, bilateral, sigma, TV-L^2, median, then mean. But if you look at frame 75 (roughly the same difference in illumination) the rank is (best to worst): mean, bilateral, trilateral, median, sigma, then TV-L^2; a completely

n		Original	TV-L^2	Sigma	Mean	Median	Bilateral	Trilateral
1	Ave.	55.04	7.58	7.74	7.69	7.36	6.80	6.34
	ZMSD	69.41	7.59	7.76	7.71	7.38	6.83	6.36
	Rank	7	4	6	5	3	2	1
2	Ave.	55.04	7.42	7.71	7.37	6.84	6.15	4.98
	ZMSD	69.41	7.44	7.72	7.39	6.89	6.20	5.05
	Rank	7	5	6	4	3	2	1
10	Ave.	55.04	6.88	7.45	5.63	4.73	3.30	1.72
	ZMSD	69.41	6.91	7.47	5.69	4.93	3.44	1.93
	Rank	7	5	6	4	3	2	1
40	Ave.	55.04	5.36	6.14	2.79	3.85	1.63	1.72
	ZMSD	69.41	5.43	6.21	3.21	4.17	1.95	1.93
	Rank	7	5	6	3	4	2	1
50	Ave.	55.04	5.17	5.59	2.83	3.85	1.47	1.72
	ZMSD	69.41	5.24	5.67	3.27	4.16	1.75	1.93
	Rank	7	5	6	3	4	1	2
70	Ave.	55.04	4.94	4.65	2.36	3.84	1.32	1.72
	ZMSD	69.41	5.02	4.76	2.88	4.15	1.56	1.93
	Rank	7	6	5	3	4	1	2
100	Ave.	55.04	4.76	3.78	1.95	3.84	1.26	1.72
	ZMSD	69.41	4.85	3.89	2.53	4.16	1.46	1.93
	Rank	7	6	4	3	5	1	2

Table 2. Results of TV-L^1 optical flow on EISATS sequence. Results are shown for different number of iterations n. Statistics are presented for the average (Ave.), zero-mean standard deviation (ZMSD), and the rank based on ZMSD.

different order! From this it should be obvious that a smaller dataset will not pick up on these subtleties, so a large dataset (such as a long sequence) is a prerequisite for better understanding of the behaviour of an algorithm.

Since we have such a large dataset (99 results, 100 frames) we can calculate metrics for the results as in the previous subsection. We calculate the average and ZMSD for $n = 1, 2, 10, 50, 70,$ and 100. These results are shown in Table 2. Obviously, the original images are far worse than any residual image. From this table you can see that the order of the rankings shift around depending on the number of iterations for the residual image n. Another point to note is that the trialteral filter (which is stopped at 10 iterations) is the best until after 50 iterations of the other filters; when bilateral filtering becomes the best. Simple mean filtering (which is much faster than any other filter) comes in at rank 3 after 40 iterations, and gets better around 100 iterations. It is notable that the difference between the average and ZMSD highlights how volatile the results are, the closer together the numbers, the more consistent the results.

6 Conclusions and Future Research

We have identified a methodology for analysing the effect of illumination reducing filters using numerical comparisons, exploiting the co-occurrence metrics and

Spatial-RMS. We went on to show that the results for this test do align with the optical flow performance, on a scene with drastic illumination variation. The tests showed that generating a simple mean residual image, produces acceptable improvements, while being the fastest (computational time) and easiest (simplicity) to implement. The bilateral and trilateral filter were also very good. Future work should test the limits of the proposed methodology. Other smoothing algorithms and illumination invariant models need to be tested. Finally, a larger dataset can be used to further verify the illumination artifact reducing effects of residual images.

References

1. Anonymous: Residual images remove illumination artifacts. Submitted to *Pattern Recognition - DAGM*, (2009)
2. Aujol, J. F., Gilboa, G., Chan, T., and Osher, S.: Structure-texture image decomposition - modeling, algorithms, and parameter selection. *Int. J. Computer Vision*, **67**:111-136 (2006)
3. Baker, S., Scharstein, D., Lewis, J. P., Roth, S., Black, M., and Szeliski, R.: A database and evaluation methodology for optical flow, in Proc. *IEEE Int. Conf. Computer Vision (ICCV)*, pages 1–8 (2007)
4. Choudhury, P., and Tumblin, J.: The trilateral filter for high contrast images and meshes. In Proc. *Eurographics Symp. Rendering*, pages 1–11 (2003)
5. *.enpeda..* dataset 2 (EISATS): `http://www.mi.auckland.ac.nz/EISATS`
6. Haralick, R. M., and Bosley, R.: Texture features for image classification. In Proc. *ERTS Symposium*, NASA SP-351, pages 1219-1228 (1973)
7. Haussecker, H. and Fleet, D. J.: Estimating optical flow with physical models of brightness variation. *IEEE Trans. Pattern Analysis Machine Intelligence*, **23**:661–673 (2001)
8. Hirschmüller, H., and Scharstein, D.: Evaluation of stereo matching costs on images with radiometric differences. *IEEE Trans. Pattern Analysis Machine Intelligence*, to appear
9. Kuan, D. T., Sawchuk, A. A., Strand, T. C., and Chavel, P.: Adaptive noise smoothing filter for images with signal-dependent noise. *IEEE Trans. Pattern Analysis Machine Intelligence*, **7**:165–177 (1985)
10. Lee, J.-S.: Digital image smoothing and the sigma filter. *Computer Vision, Graphics, and Image Processing*, **24**:255–269 (1983)
11. Mileva, Y., Bruhn, A. and Weickert, J.: Illumination-robust variational optical flow with photometric invariants. In Proc. *Pattern Recognition - DAGM*, pages 152–162 (2007)
12. Middlebury dataset: `http://vision.middlebury.edu/stereo/data/`
13. Middlebury Optical Flow Evaluation: `http://vision.middlebury.edu/flow/`
14. Morales, S., Woo, Y. W., Klette, R., and Vaudrey, T.: A study on stereo and motion data accuracy for a moving platform. Technical report, MI-tech-32, `http://www.mi.auckland.ac.nz/`, University of Auckland (2009)
15. Rudin, L., Osher, S., and Fatemi, E.: Nonlinear total variation based noise removal algorithms. *Physica D*, **60**:259–268 (1992)
16. Scharstein, D., and Pal, C.: Learning conditional random fields for stereo. In Proc. *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, (2007)
17. Tomasi, C., and Manduchi, R.: Bilateral filtering for gray and color images. In Proc. *IEEE Int. Conf. Computer Vision*, pages 839–846 (1998)
18. Vaudrey, T., Rabe, C., Klette, R., and Milburn, J.: Differences between stereo and motion behaviour on synthetic and real-world stereo sequences. In Proc. *IEEE Image and Vision Conf. New Zealand*, Digital Object Identifier 10.1109/IVCNZ.2008.4762133 (2008)

19. Wedel, A., Pock, T., Zach, C., Bischof, H., and Cremers, D.: An improved algorithm for TV-L^1 optical flow. In Post Proc. *Dagstuhl Motion Workshop*, to appear (2009)
20. van de Weijer, J. and Gevers, T.: Robust optical flow from photometric invariants. In Proc. *Int. Conf. on Image Processing*, pages 1835–1838 (2004)
21. Zach, C., Pock, T., and Bischof, H.: A duality based approach for realtime TV-L^1 optical flow, In Proc. *Pattern Recognition - DAGM*, pages 214–223 (2007)