RECONSTRUCTING WHITE WALLS: MULTI-VIEW, MULTI-SHOT 3D RECONSTRUCTION OF TEXTURELESS SURFACES

Andreas Ley, Ronny Hänsch, Olaf Hellwich

Computer Vision & Remote Sensing Group, Technische Universität Berlin, Berlin, Germany
(andreas.ley, r.haensch, olaf.hellwich)@tu-berlin.de

KEY WORDS: Structure from Motion, Multi-View Stereo, 3D Reconstruction, Image Enhancement, Denoising

ABSTRACT:

The reconstruction of the 3D geometry of a scene based on image sequences has been a very active field of research for decades. Nevertheless, there are still existing challenges in particular for homogeneous parts of objects. This paper proposes a solution to enhance the 3D reconstruction of weakly-textured surfaces by using standard cameras as well as a standard multi-view stereo pipeline. The underlying idea of the proposed method is based on improving the signal-to-noise ratio in weakly-textured regions while adaptively amplifying the local contrast to make better use of the limited numerical range in 8-bit images. Based on this premise, multiple shots per viewpoint are used to suppress statistically uncorrelated noise and enhance low-contrast texture. By only changing the image acquisition and adding a preprocessing step, a tremendous increase of up to 300% in completeness of the 3D reconstruction is achieved.

1. INTRODUCTION

The reconstruction of the 3D geometry of a scene based on image sequences has been a very active field of research. The joint effort in the development of keypoint detectors, matching techniques, path estimation methods, bundle adjustment, and dense reconstruction has resulted in very successful solutions that are able to robustly and accurately reconstruct a 3D scene from given images. Nevertheless, there are still existing challenges in particular for difficult acquisition circumstances (e.g. inhomogeneous or nonconstant lighting), difficult objects (e.g. with homogeneous or reflective surfaces), or certain application scenarios (e.g. facade reconstruction). One especially critical example are weakly-textured parts of objects, where the lack of correct matches leads to holes and topological errors within the 3D point cloud.

This paper proposes to enhance the 3D reconstruction of weakly-textured surfaces by using standard cameras as well as a standard multi-view stereo (MVS) pipeline. Only changing image acquisition and adding a preprocessing step enabled a tremendous increase in completeness of the achieved reconstruction.

The underlying idea of the proposed method is based on the insight that in the real world there is no such thing as a textureless surface. What is usually meant with this phrase is that the existing physical texture is either too fine-grained to be captured by the spatial resolution of a given camera, or that it has insufficient contrast. Especially the latter causes the contribution of the physical texture to the measured signal to drop below the contribution of the measurement noise, which makes a distinction of the two close to impossible. Consequently, all that is necessary to exploit the existing texture in this case is to enhance the signal-to-noise-ratio. Based on this premise, this paper proposes to use multiple shots per viewpoint to suppress statistically uncorrelated noise and enhance low-contrast texture. It models the measured signal as a mixture of truly random as well as fixed-pattern noise and leads to a significant improvement of image quality and (more importantly) of the completeness of the reconstruction. An example is shown in Figure 1, where 3D points are color coded depending on how many images per viewpoint were at least necessary for their reconstruction (1≙blue to 30≙red). A flexible and open-source C++ implementation of the proposed framework is publicly available (Ley, 2016).

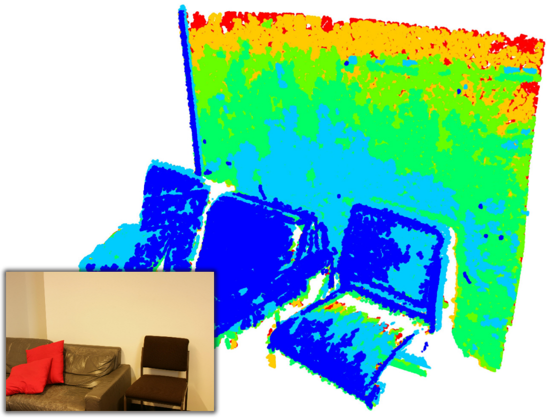

Figure 1. Exemplary reconstruction: 3D points are color coded depending on how many images per viewpoint were at least necessary for their reconstruction (1≙blue to 30≙red)

2. RELATED WORK

The principle idea to increase the image quality to ease the task of 3D reconstruction is not new. Previous approaches can be coarsely divided into two groups: Exploiting multiple images or using local enhancement methods based on a single image.

There are rarely any 3D reconstruction pipelines that explicitly use multiple images from the same viewpoint. On the contrary, many state-of-the-art approaches seek to automatically reject images that have a too small baseline, since these image pairs do not provide strong constraints on the estimated depth. One of the seldom examples that does use multiple images is High Dynamic Range (HDR) imaging. The lowest and highest light intensity of real-life scenes can easily reach a ratio of $500,000:1$ (Debevec and Malik, 1997). This high dynamic range can usually not be handled by standard cameras leading to a clipping of dark areas to zero or bright areas to 255 and therefore a loss of details in these regions. HDR imaging has been proposed as a solution to this problem by combining multiple images with different exposure times or shutter speeds to a single radiance map capturing a large dynamic range (Mann and Picard, 1995, Debevec and Malik, 1997, Robertson et al., 1999, De Neve et al., 2009).

Only few works try to exploit the advantages of HDR images for computer vision applications in general and multi-view stereo in particular. The advantages of HDR photogrammetry have been shown in laboratory experiments in (Cai, 2013). While HDR images are used in (Ntregkaa et al., 2013) and (Guidi et al., 2014) to improve 3D reconstructions by simultaneously enhancing contrast in dark and bright regions of buildings and vases/plates of cultural heritage, respectively, (Kontogianni et al., 2015) investigates the influence of HDR images on keypoint detection.

It might not be possible or desirable in all cases to acquire multiple images from the same viewpoint or to rely on more advanced HDR image composition techniques that can handle moving cameras. The next best choice is to use contrast enhancement techniques that are based on a single image. It has been shown that contrast enhancement increases the performance of keypoint detectors (e.g. (Lehtola and Ronnholm, 2014)). The work of (Ballabeni et al., 2015) investigates the influence of multiple image preprocessing techniques including color balancing, image denoising, RGB to gray conversion, and content enhancement on the performance of the 3D reconstruction pipeline.

3. PROPOSED METHODOLOGY

Weakly-textured areas in images generally pose two crucial problems for any 3D reconstruction pipeline. First, as discussed above, “textureless” often means that the texture is “hidden” because the contribution of the physical texture to the measured signal is approximately as strong as the measurement noise. Consequently, the first task is to enhance the signal, i.e. to significantly suppress the noise. The second problem is that the restored texture remains weak compared to image areas with higher contrast but needs to be stored within the same image data. HDR images for example solve this problem by allowing enough numerical precision to simultaneously represent small and large signal changes. However, freely available MVS pipelines that can process HDR images (or similar advanced formats) are rare ((Rothermel et al., 2012) is one of the few exceptions). That is why the second task consists of saving weak and strong texture components within the same 8-bit image to use common reconstruction software. An overview of the proposed noise reduction is visualized in Figure 2.

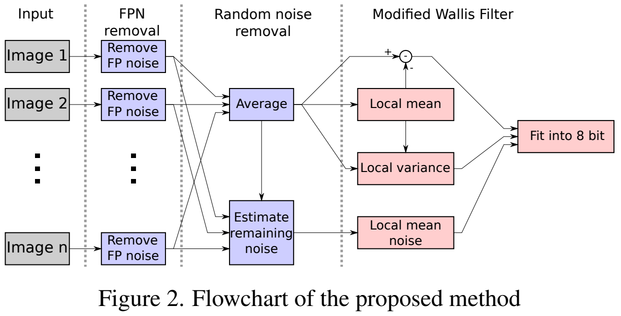

Figure 2. Flowchart of the proposed method

To actually increase the signal-to-noise ratio (SNR) we explore the idea of averaging multiple images per viewpoint. The photographer is required to take a whole range of images per viewpoint by shooting with a tripod to guarantee alignment. The images are averaged to suppress the stochastic noise. In theory this should result in a reduction of the noise’s standard deviation by $1/\sqrt{n}$ for n images under the assumption of iid. noise. The combination of multiple exposures into a single image is similar to the HDR radiance acquisition proposed by e.g. (Debevec and Malik, 1997) which was shown to improve 3D reconstructions with under- and overexposed regions (Guidi et al., 2014). Contrary to (Guidi et al., 2014) we do not try to acquire as few images per viewpoint as possible while covering a large exposure range. Instead we use a large batch of images per viewpoint covering a small exposure range, which can easily be accomplished with extended camera firmwares that support scripting.

The approach of averaging can only deal with random fluctuations that are temporally decorrelated. Realistic image noise is more complicated than that and is actually a mixture of this spatially and temporally random process and a spatially random but temporally deterministic process (i.e. fixed pattern noise, FPN). We model both effects by an extended signal model (Section 3.1) and estimate the parameters of the FPN beforehand (Section 3.2).

To put the gains of averaging multiple exposures into perspective, a second option to increase the SNR is explored: The BM3D filter (Dabov et al., 2007) suppresses image noise by exploiting the self similarity of small patches in natural images. This noise reduction filter was shown to improve the reconstruction (Ballabeni et al., 2015) and requires only a single image per viewpoint. Our own empirical observations (see Section 4.1) support this, especially with weakly-textured surfaces.

The output of the fused images as well as the BM3D filter are stored with floating point precision. Within these images only the high frequency components (i.e. fine textured regions) are of actual interest for 3D reconstruction since only those can be matched with the desired spatial precision. A high-pass filter would amplify the high frequency components, but tweaking the filter gain to sufficiently enhance weakly-textured regions without clamping/saturating strongly-textured parts proves difficult. The proposed solution is to adaptively amplify the signal depending on the local variance. This is accomplished by the application of the Wallis filter (Wallis, 1974) to the non-quantized images. It removes the local mean and amplifies the output by the reciprocal of the local standard deviation (see Section 3.4). The result is a signal with removed low frequency components and constant local variance, which can be reduced into 8-bit images while keeping the quantization noise small. The resulting images are fed into an off-the-shelf reconstruction pipeline (an in-house SfM implementation or VSFM (Wu, 2007, Wu, 2013, Wu et al., 2011), followed by PMVS2 (Furukawa and Ponce, 2010)).

The following subsections discuss in more detail the individual processing steps as well as our assumptions concerning the noise.

3.1 Noise model

The image noise usually observed in images is created, modified, and shaped by many camera components. The irradiance itself, due to the quantized nature of photons, is already a random process. But also the electrical signals and charges in the sensor are subject to noise. In addition to being random, the expected values of these processes can also differ slightly from pixel to pixel, resulting in a fixed pattern noise. Signal and noise progress further through A/D conversion, demosaicing, color matrix multiplication, tone mapping, and JPG compression, each step of which further obfuscates the nature of the noise. To simplify the situation, we use demosaiced raw images, bypassing the effects of color correction, tone mapping, and JPG compression. In addition, the bit depth of the raw images is usually slightly higher than the common 8 bit per channel. Since SfM/MVS pipelines are largely robust to changes in brightness, contrast, and color temperature, we assume that aesthetically pleasing color correction and tone mapping is not of importance and can be omitted.

Without the need to model these difficult effects, we assume for each channel a linear relationship between the measured value and the actual exposure perturbed by noise. Let $v_c(x, y)$ be the pixel value of channel c at pixel location x, y. Let further $e_c(x, y)$ be the true exposure that we seek to estimate. We assume a linear relationship between $e_c(x, y)$ and $v_c(x, y)$ that is individually modeled by Equation 1 for each pixel and channel by a scale $s_c(x, y)$ and an offset $o_c(x, y)$.

$$\underbrace{v_c(x,y)}_{\text{output}} = \underbrace{s_c(x,y)}_{\text{scale}} \cdot \big(\underbrace{e_c(x,y)}_{\text{input}} + \underbrace{n_c(x,y)}_{\text{noise}}\big) + \underbrace{o_c(x,y)}_{\text{offset}} \quad (1)$$

The additive noise term $n_c(x, y)$ is assumed to have zero mean (i.e. $E[n_c(x, y)] = 0$). In other words, we assume that if the expected value of the noise should be nonzero, that it can be modeled by the scale and offset of the linear relationship. We refer to the random, mean-free noise term $n_c(x, y)$ as the *random noise*, whereas the patterns caused by differing scales $s_c(x, y)$ and offsets $o_c(x, y)$ will be referred to as the *fixed pattern noise* (FPN).

Given this signal model, denoising is a simple process of reversing the linear model and averaging. The N images $v_{i,c}(x, y)$ of each vantage point are first freed of their FPN by Equation 2 and then averaged by Equation 3 to suppress the random noise.

$$\hat{e}_{i,c}(x,y) = \frac{v_{i,c}(x,y) - o_c(x,y)}{s_c(x,y)} \quad (2)$$

$$\bar{e}_c(x,y) = \frac{1}{N}\sum_{i=1}^{N} \hat{e}_{i,c}(x,y) \quad (3)$$

A precise estimation of the absolute pixel values is not of importance at this point since SfM/MVS pipelines are robust with respect to intensity changes. It is only necessary to ensure that neighboring pixels behave equally. Otherwise, high frequency noise is created which is against the principles of our method.

It should be noted that we apply this model to the image after demosaicing. At this point of the image formation process, the measured intensity $v_c(x, y)$ does actually not only depend on $e_c(x, y)$ at x, y. It is also influenced by the exposures and noise terms in the local neighborhood, where the precise nature of this influence depends on the exact shape and nature of the demosaicing filter. Nonetheless, we ignore this spatial dependency and leave a thorough investigation of its effects for future work.

3.2 Estimating FPN parameters

This section deals with the estimation of the per pixel x, y and channel c scales $s_c(x, y)$ and offsets $o_c(x, y)$ that define the FPN. The FPN is camera dependent and must be estimated with the very camera which is also used to capture the images of the 3D reconstruction. We conducted a couple of simple experiments to investigate the stability of the FPN noise. It seems to stay unchanged over a long period of time, to be independent from the battery charge level, and does neither change with camera movement nor turning the camera on/off. However, future work should investigate the stability with respect to e.g. temperature and camera age. Dust grains on or in the lens are a very transient form of FPN that needs to be kept in check by repeated cleaning.

Central to our approach of calibrating the FPN is the observation that only (spatially) high frequency texture is of interest and thus only high frequency noise needs to be suppressed. Smooth transitions between slightly darker and slightly brighter regions, like vignetting, are acceptable because of the aforementioned robustness of SfM/MVS to those variations.

By covering the lens with a white, translucent cap, setting the focus to infinity, and opening the aperture as far as possible, a homogeneous (or at least smooth) image is projected onto the sensor. In this setup, multiple images are taken with different exposure times to cover different exposures. We use $M = 7$ stops with $N = 85$ images per stop. Let $v_{l,i,c}(x, y)$ be the ith image captured for the lth stop. For each stop, the average image is computed to suppress the random noise.

$$\overline{v}_{l,c}(x,y) = \frac{1}{N}\sum_{i=1}^{N} v_{l,i,c}(x,y) \quad (4)$$

At this point the seven average images $\overline{v}_{l,c}(x, y)$ should be smooth if not for FPN. All deviations are simply attributed to FPN.

We estimate the true exposure by applying a Gaussian blur to the seven average images (Equation 5) and perform a weighted least squares (LS) fit to estimate the scale and offset (Equation 6).

$$\tilde{e}_{l,c}(x,y) = \overline{v}_{l,c}(x,y) * G_{\sigma=30}(x,y) \quad (5)$$

$$\begin{pmatrix} s_c(x,y) \\ o_c(x,y) \end{pmatrix} = \underset{s,o}{\arg\min} \sum_{l=1}^{M=7} \frac{\big(s \cdot \tilde{e}_{l,c}(x,y) + o - \overline{v}_{l,c}(x,y)\big)^2}{\alpha_{l,c}(x,y)} \quad (6)$$

From the variance between the individual images $v_{l,i,c}(x, y)$ that were merged into the average images $\overline{v}_{l,c}(x, y)$, a confidence interval and subsequently a weight $\alpha_{l,c}(x, y) = \mathrm{CI}_{l,c}(x, y)^2$ can be computed where $\mathrm{CI}_{l,c}(x, y)$ is the width of the confidence interval for $\overline{v}_{l,c}(x, y)$ (see Equation 9).

Care must be taken with slightly imprecise exposure times or changes in the ambient light which result in an overestimation of the confidence intervals. We equalize the brightnesses of the images $v_{l,i,c}(x, y)$ by scaling the image values such that the local brightnesses of each image are close to $\tilde{e}_{l,c}(x, y)$:

$$v'_{l,i,c}(x,y) = v_{l,i,c}(x,y) \cdot \frac{\overline{v}_{l,c}(x,y) * G_{\sigma=30}(x,y)}{v_{l,i,c}(x,y) * G_{\sigma=30}(x,y)} \quad (7)$$

With these brightness adjusted images, the remaining random noise in the average $\overline{v}_{l,c}(x, y)$ can be estimated by Equation 8.

$$\hat{n}_{l,c}(x,y) = \frac{1}{N^2}\sum_{i=1}^{N} \big(v'_{l,i,c}(x,y) - \overline{v}_{l,c}(x,y)\big)^2 \quad (8)$$

The actual confidence interval width follows from the assumption of a Student-t distributed average

$$\mathrm{CI}_{l,c}(x,y) = 2 \cdot c_{95\%,N-1} \cdot \sqrt{\hat{n}_{l,c}(x,y)} \quad (9)$$

where $c_{95\%,N-1}$ is the 95% percentile of the Student-t distribution with $N - 1$ degrees of freedom.

The fitted lines can be seen for the red channel of 4 × 4 example pixels in Figure 3. Even though the confidence intervals are quite large, slight variations in the pixels' sensitivities are perceivable. At least the stronger deviations are well expressed by the linear relationship. A LS fit with uniform weighting is shown in blue for comparison.

Figure 4 visualizes the scales $s_c(x, y)$ and offsets $o_c(x, y)$ per pixel x, y and color channel c as images. Several patterns are noticeable, such as the out of focus dust grains in the scale image, or the vertical stripes in the offset image.

3.3 Noise reduction evaluation

In addition to a practical evaluation with current 3D reconstruction software in Section 4, we verify the effectiveness of our noise suppression in a more controlled test setup. We acquire a new set

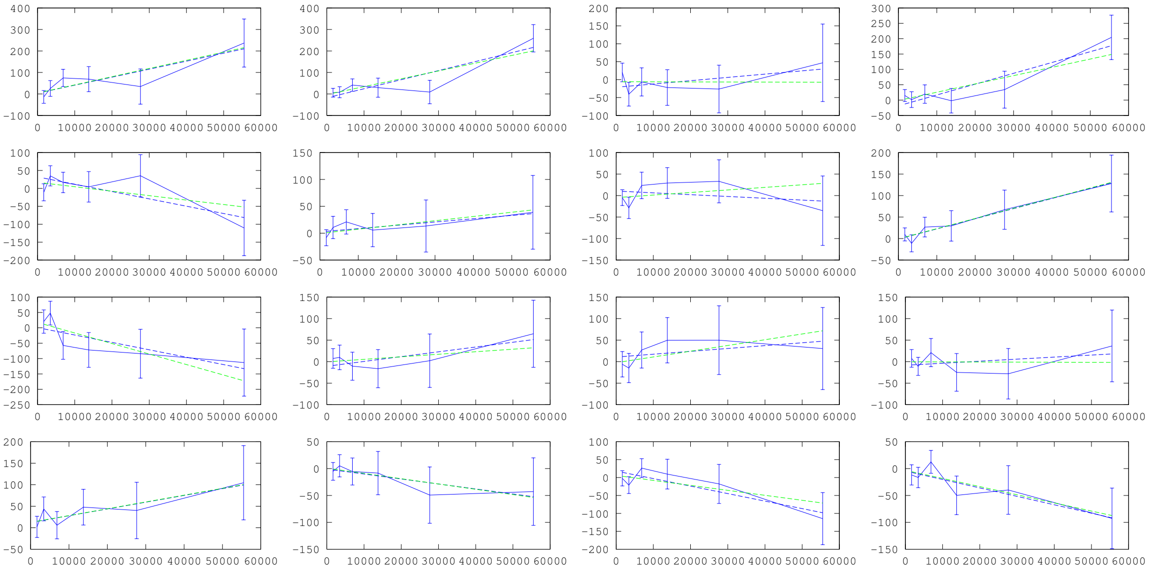

Figure 3. Deviations $\overline{v}_{l,c}(x, y) - \tilde{e}_{l,c}(x, y)$ over $\tilde{e}_{l,c}(x, y)$ for the red channel. Note that the 7th stop saturated the red channel and is thus not included. Dashed lines are fitted linear FPN models (blue LS, green weighted LS of Equation 6).

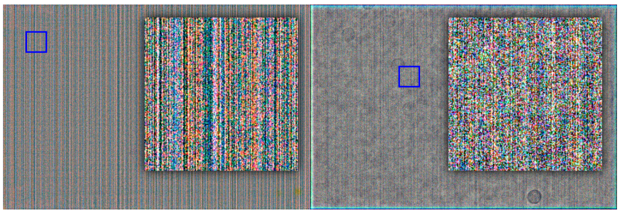

Figure 4. Contrast enhanced visualization of the FPN model. Offset o (left) and scale s (right) for all three color channels.

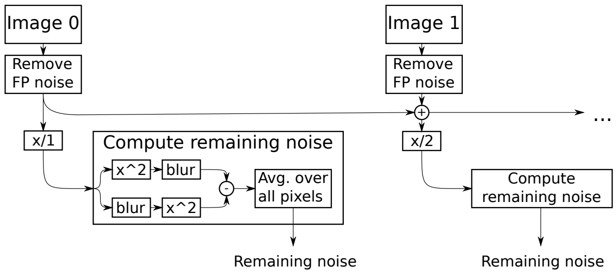

Figure 5. Evaluation of the noise reduction for varying numbers of images by iteratively averaging images and computing the average local variance as an estimate of the remaining noise.

of 100 images with the same camera setup as in Section 3.2, i.e. a set of 100 images that should be smooth, if not for noise. The exposure time was tweaked to achieve an image that is neither bright nor dark. The actual tone mapped sRGB color values are approximately (160 120 60). In the (16-bit) raw images, this is on average (11474.1 6148.32 2586.93).

Since the images *should* be smooth in the absence of noise, the (random and fixed pattern) noise in an image can be measured by estimating the local variance in each color channel and averaging that variance over the image. The local variance $\mathrm{var}_{\bar{e}_c}(x, y)$ of an image $\bar{e}_c(x, y)$ is computed by replacing the usual summation in the estimation of expected values with a Gaussian convolution that serves as a soft windowing function (Equations 10-12).

$$\underbrace{\mathrm{var}_{\bar{e}_c}(x, y)}_{\text{local second central moment}} = \underbrace{E\left[(\bar{e}_c(x, y))^2\right]}_{\text{local second moment}} - \underbrace{E\left[\bar{e}_c(x, y)\right]^2}_{\text{local first moment}} \quad (10)$$

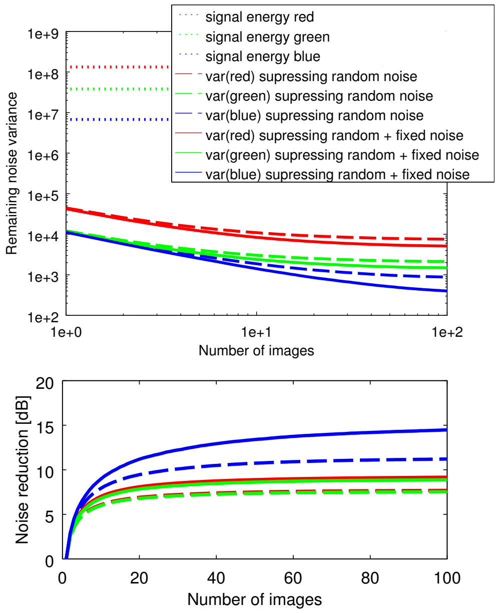

Figure 6. Noise reduction of the proposed approach with and without FPN for increasing numbers of images within the different color channels (denoted in corresponding color). Top: Remaining noise variance in the 16-bit value range. Bottom: Noise reduction in dB relative to the initial noise.

$$E\left[(\bar{e}_c(x, y))^2\right] = (\bar{e}_c(x, y))^2 * G_{\sigma=30}(x, y) \quad (11)$$

$$E\left[\bar{e}_c(x, y)\right] = \mathrm{mean}_{\bar{e}_c}(x, y) = \bar{e}_c(x, y) * G_{\sigma=30}(x, y) \quad (12)$$

We iteratively process an increasing subset of the 100 verification images to plot the remaining noise against the number of used images. Figure 5 visualizes this iterative process. The results can be seen in Figure 6 alongside the curves without FPN removal. For reference, a “signal energy” is plotted as well which is simply the sum of squared values in the image. We do not compare against BM3D here, because the strength of BM3D is to reduce the noise while keeping the underlying structure intact. This aspect of BM3D can’t be evaluated here, since there is no real structure in this test.

As can be seen in the plots, averaging images significantly reduces the image noise. However, even with FPN removal, the curves start to converge after about 30 to 40 images which indicates that the FPN model still leaves room for improvement. Of more practical significance is the observation, that for the first couple of images the gains of the FPN removal are rather small. This means that even without the, by comparison, complicated estimation of the FPN, significant gains can be achieved by just averaging a hand full of images.

3.4 Packing into 8-bit

Most SfM/MVS implementations handle images as grayscale or rgb images with 8 bits per channel. The *peak* signal to quantization noise ratio is approximately $6.02 \cdot Q$ dB for Q bits. Considering this, the PSNR due to quantization noise of 48 dB is more than acceptable if the full range of 256 intensity values is used.

In the case of SfM/MVS, the matched signals are high frequency signals extracted from small patches. In the showcased "White Walls" scenario (see Section 4.1), these local image patches most definitely do not make use of the full intensity range.

In the test performed in Section 3.3, the standard deviation of the residual noise in the blue channel can be suppressed to about 29/65535 or 0.11/255 with just 20 images. By just linearly mapping the denoised images into the 8-bit range, a substantial amount of the gained signal fidelity would be lost.

To retain these weak signals, they must be amplified, i.e. mapped to a larger interval of the 8-bit range. High frequency image structures are much more important for SfM/MVS than low frequency image structures. By using a high pass filter, the low frequency components of the image are discarded to allow the high frequency components to make full use of the 256 value range. Since the high frequency textures are of different power in different image regions (e.g. compare the strong texture on the seat and sofa with the weak texture on the wall in Figure 7), we use a modified Wallis filter (Wallis, 1974) which adaptively adjusts the gain based on the local variance.

As in Equation 10 let $\text{var}_{\bar{e}_c}(x, y)$ be the local variance of $\bar{e}_c(x, y)$ and let $\text{mean}_{\bar{e}_c}(x, y)$ be its local mean (see Equation 12). The standard Wallis filter normalizes the signal by subtracting the local mean and dividing by the local standard deviation. Fitting e.g. the 3σ range of this normalized signal into the 8-bit value interval becomes a simple matter of scaling and shifting. Contrary to Equations 10 and 12, however, we use a Gaussian width of $\sigma = 10px$ for the Wallis filter.

$$w_c(x, y) = \left\lfloor \frac{\bar{e}_c(x, y) - \text{mean}_{\bar{e}_c}(x, y)}{\sqrt{\text{var}_{\bar{e}_c}(x, y)}} \cdot \frac{127}{3} + 127.5 \right\rfloor \tag{13}$$

A big problem with this approach (apart from the fact that the local signal variance might be estimated as zero) is that some SfM/MVS implementations implicitly assume a constant amount of noise in the images. This can be due to reasons as simple as fixed thresholds during feature detection. However, the variance of the remaining image noise is far from constant. Firstly, the sensor noise itself is already not homoscedastic. Secondly, does the Wallis filter drastically change the amplification from image region to image region. We alleviate this problem by imposing a maximal amplification on the Wallis filter such that the 3σ range of the remaining random image noise in the local neighborhood is below $10/255$ (see Equations 14-16).

$$u_c(x, y) = \lfloor (\bar{e}_c(x, y) - \text{mean}_{\bar{e}_c}(x, y)) \cdot \lambda_c(x, y) + 127.5 \rfloor \tag{14}$$

$$\lambda_c(x, y) = \min\left(\frac{127}{3 \cdot \sqrt{\text{var}_{\bar{e}_c}(x, y)}}, \frac{10}{3 \cdot \sqrt{\text{noise}_c(x, y)}} \right) \tag{15}$$

$$\text{noise}_c(x, y) = \hat{n}_c(x, y) * G_{\sigma=10}(x, y) \tag{16}$$

The variance of the remaining random noise $\hat{n}_c(x, y)$ is estimated from the spread of the original images $v_{i,c}(x, y)$. These images are, similar to Section 3.2, not all of the same brightness and if not compensated for, these fluctuations would lead to a large overestimation of the remaining noise. We proceed as in Equation 7 by adjusting brightnesses and estimate $\hat{n}_c(x, y)$ as in Equation 8, albeit both with the smaller Gaussian kernel of $\sigma = 10px$.

Note that we normalize each color channel c individually. We also experimented with normalization based on the full local 3×3 covariance matrix of the three color channels but observed no gain over the individual approach (see Figure 9). We believe that normalizing each color channel individually is sufficient to overcome quantization noise and that a full "whitening" does not add any additional benefit beyond that.

4. EXPERIMENTAL EVALUATION

A purely quantitative evaluation of the proposed noise reduction is already provided in the previous section (see Figure 6) and is based on a rather theoretical example of an image consisting completely of a homogeneous area. It was shown, that the signal model incorporating truly random as well as fixed pattern noise showed the best reduction of noise within the processed images, although saturating considerably earlier than the theoretical bound. This section focuses on the impact of the proposed method on multi-view stereo reconstruction methods.

Two datasets have been acquired by a standard Canon EOS 400D, both containing weakly-textured areas but objects of different scales and geometry.

4.1 Wall dataset

The *Wall* dataset consists of images of a scene containing a couch and a chair in front of a white wall. The top of Figure 7 shows an example. Large parts of the scene are planar and visually homogeneous areas (i.e. weakly-textured), but of different color and intensity. The scene is pictured from seven different viewpoints with 30 images per viewpoint and an average baseline of 40 cm.

In order to analyze the respective influence of the individual components, four different test cases are defined:

1. Single exposure (SE): A single JPG image is provided per viewpoint. No noise reduction or image enhancement is applied (besides the camera internal processing).
2. Single exposure with Wallis filter (SEW): A single JPG image is provided per viewpoint. The contrast of the image is enhanced by the Wallis filter.
3. Single exposure with BM3D and Wallis filter (SEBW): As 2), but the noise is locally suppressed by the BM3D filter before the Wallis filter is applied.
4. Multi exposure with modified Wallis filter (ME-N): A set of N images is provided per viewpoint. The noise is suppressed by combining multiple images as described in Section 3.1. Contrast is enhanced by the modified Wallis filter of Section 3.4.

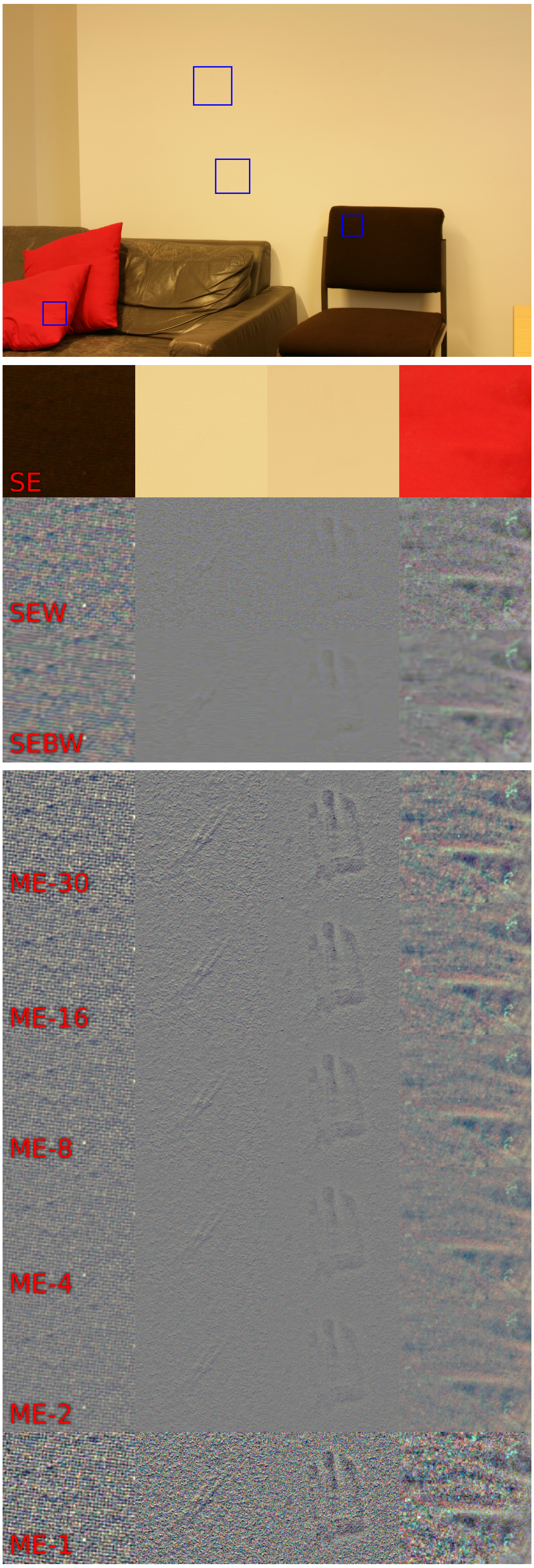

Figure 7. Noise suppression and contrast enhancement: Top: Example image (Cropped details marked in blue); From top to bottom: SE, SEW, SEBW, ME-30, -16, -8, -4, -2, -1 crops.

The blue squares within the image shown at the top of Figure 7 denote four different image details. While the first row shows a zoom into these parts within the JPG image (SE), the second and third row show these image patches after being processed by either the Wallis filter alone (SEW) or in combination with the BM3D filter (SEBW), respectively.

The images appear grayish since the Wallis filter normalizes mean and variance of the color channels. The contrast in general and especially for fine structures is considerably increased, making small details visible that had been hidden before. The texture of the chair is clearly recognizable now, as well as small foldings of the cushion. It should be noted, that the SEW images still contain the full amount of noise, only the contrast is enhanced.

The noise in the SEBW images in the third row is locally suppressed by the BM3D filter before contrast is enhanced by the Wallis filter. The fine textures are mostly preserved, while the noise appears to be reduced. The image patches from the wall seem to contain image structures as well, which are not visible at all in the original SE images, but become slowly recognizable in the enhanced images (SEW as well as SEBW).

The fourth till the last row show the results of the proposed framework with decreasing number of images per viewpoint (ME-30, -16, -8, -4, -2, -1, respectively). Especially in the fourth row (corresponding to ME-30) the structures on the wall become clearly recognizable: A small scratch in the wall plaster as well as a hand print. The intensity difference of these structures is so small, that they are hardly visible even for the human eye in the real world. The fewer images are used per viewpoint, the less obvious the corresponding image texture is i.e. it becomes dominated by noise. The last row (ME-1) basically corresponds to the second row (SEW). The visual discrepancy is caused by a couple of subtle differences during the processing: Firstly, while the second (and third) row are based on JPG images, the proposed approach is using raw data. Thus, there are no artefacts of the JPG compression. Secondly, while the maximal gain had to be adjusted manually for the Wallis filter, it is automatically computed by the proposed method based on an estimate of the remaining image noise. If only a single image is used, the remaining noise is estimated as zero, which leads to an unconstrained gain. While the increased contrast of ME-1 is visually pleasing, it is important to note that the apparent details are mostly random noise and not actual structure. The details in ME-30 on the other hand are real, as evidenced by the plausible shading in the wall crops.

It should be noted that the visualization as discussed above is at best a qualitative cue for an increased performance. Since the application of the proposed framework is image-based 3D reconstruction, the influence on the final 3D point cloud is analyzed instead of assessing the quality of the intermediate results.

Figure 8 shows the point cloud based on the same image data as visualized in Figure 7 obtained by an successive application of the custom SfM pipeline (to obtain the camera parameters) and PMVS2 (to compute the dense reconstruction). The first row shows the reconstruction for SE (left), SEW (middle), and SEBW (right). The remaining rows show results of the proposed pipeline with increasing number of images (i.e. ME-1,-2,-4,-8,-16,-30). The number of reconstructed points is summarized in Figure 9.

There is no significant difference in the number of reconstructed points between SE and SEW. However, the points of the wall around the chair are wrongly reconstructed in the first case. They do not lie on the planar surface of the wall in the reconstruction but are strongly distorted towards the chair. These wrong reconstructed parts are not existent in SEW-based reconstruction, while

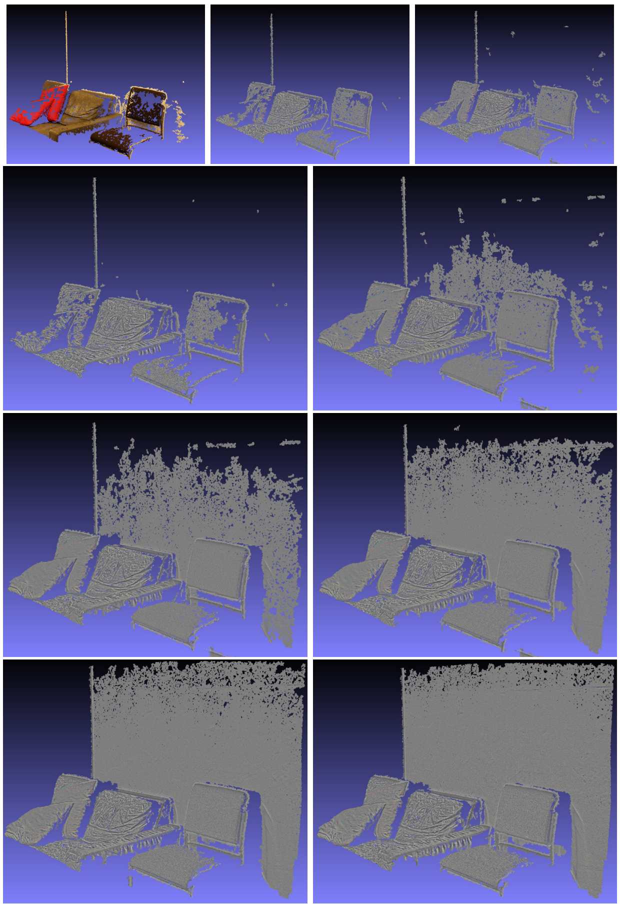

Figure 8. Wall dataset. Top row: SE, SEW, SEBW; Second to fourth row: ME-1, -2, -4, -8, -16, -30 images, respectively.

other parts of the scene, majorly on the backrest of the chair, are more complete.

For SEBW the changes are more prominent, visible on the right top part of Figure 8 as well as in Figure 9. Chair and cushion are significantly more completely reconstructed (the number of points increased by 35% from $130,939$ for SEW to $176,782$ for SEBW). Interestingly, the first spots on the white wall got reconstructed as well. These points do lie on or close to the plane in contrast to the wall points reconstructed only on the basis of the unprocessed JPG images.

This indicates the importance of noise suppression for 3D reconstruction, which is even more emphasized by the results of the proposed processing chain. While the reconstruction based on a single image (ME-1) is (not surprisingly) close to the SEW point cloud, already using two images (ME-2) outperforms the SEBW-based reconstruction by far. Not only cushion as well as seat and backrest of the chair are well reconstructed, but also a significant amount of wall pixels. The number of points increased by nearly 80% from $132,443$ using one image to $237,562$ using two images. This trend is continued by using more images per viewpoint, although the steepness of the increase becomes significantly smaller after eight images. For the maximum of 30 used images the white wall is completely reconstructed besides a few small remaining holes at the top where the cameras have less overlap. The final number of reconstructed points is $591,890$ - about three times as many as in the standard case of using only one unprocessed JPG image.

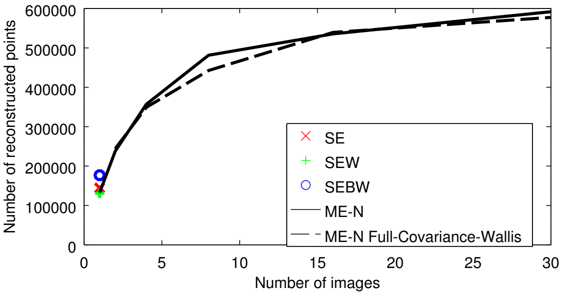

Figure 9. Wall dataset. Number of points of the dense reconstruction for different numbers of images per viewpoint

4.2 Cat dataset

The *Cat* dataset consists of images of a cat sculpture. The surface is homogeneously white and shows barely any texture (in the images as well as to the human eye in the real world). The top left of Figure 10 shows an example image. The dataset consists of 30 images for each of the 14 different viewpoints with an average baseline of 10 cm.

Two reconstructions are conducted based on VSFM and PMVS2 using the standard data (single exposure JPG images) on the one hand as well as the proposed approach (multi exposure with modified Wallis) on the other hand. The top right of Figure 10 shows the number of points within the dense reconstructions divided into two groups: Points on the cat (blue bars) and points in the background (red bars). The background (in particular the surface of the ground) contains strong texture. Consequently, both methods are able to reconstruct it well and lead to a similar number of reconstructed points. Even for the cat there is not much difference with respect to the number of points, if only a few images are used: The number of points increased from $160,465$, over $160,639$, to $242,250$, for the standard approach and the proposed approach with 2 and 30 images, respectively. However, the accuracy of the reconstruction differs largely. While the second row of Figure 10 shows the reconstructed point clouds of SE and ME-30, respectively, the last four rows show the meshing result based on SE, ME-1, ME-2, and ME-30, respectively. The standard approach results in as many points as ME-2 (and nearly twice as many as ME-1), but roughly 20% of them contain severe errors. Especially the back of the cat is either missing or reconstructed at wrong positions. The ME-1 based reconstruction (despite containing less points) contains less outliers. The ME-30 based reconstruction is very complete as well as (visually) accurate.

5. CONCLUSION AND FUTURE WORK

Weakly-textured surfaces pose a great challenge for MVS reconstructions because the texture is effectively hidden by sensor noise and can no longer reliably be matched between images. We propose to suppress random as well as fixed pattern noise. The former through averaging of multiple exposures, the latter with the aid of a fixed pattern noise model that is estimated from test images. The enhanced image is high-pass filtered with locally varying filter gains to make better use of the limited numerical precision in 8-bit per channel images. Experiments show that the enhanced images result in significantly better reconstructions of weakly-textured regions in terms of precision and completeness.

The improved performance comes at the expense of a significant increase in the required amount of images and the necessity of shooting with a tripod. Both resulting in longer acquisition times and reduced convenience. The overhead of calibrating the fixed

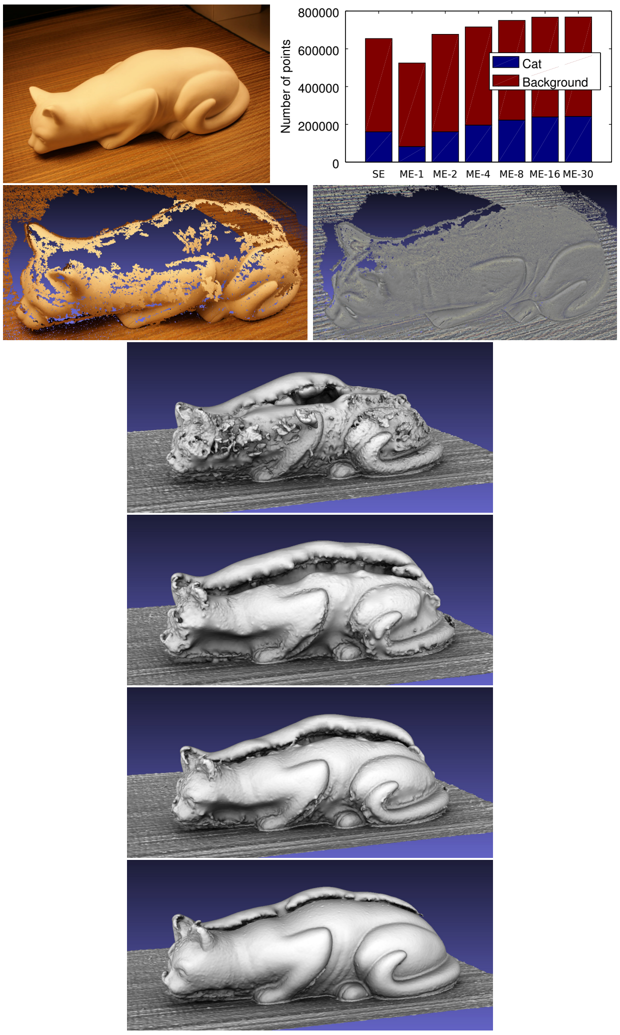

Figure 10. Cat dataset. Top to bottom, left to right: Example image, number of points in reconstruction, point cloud for SE, point cloud ME-30, mesh for SE, mesh for ME-1, mesh for ME-2, and mesh for ME-30.

pattern noise might as well be undesirable in some situations. The achieved gains quickly diminish after about 20 to 30 images, although this is already sufficient for most use-cases. Standard reconstruction pipelines benefit the most of the proposed approach in cases where the texture strength is just below the threshold necessary for a successful matching.

Besides the obvious usage in MVS reconstruction, there are additional application scenarios in other areas. Light field capture is usually performed with a fixed camera rig or a slide. In this scenario repeated shooting from the same vantage point is not an inconvenience. Automatic alignment methods (such as implemented by Google's HDR+ in Android) might be explored as well for applications without tripods. It is unclear though, what the impact on the internal camera calibration is. The proposed approach can easily be extended to capture wider, HDR exposure ranges by combining images of different exposure times. Initial experiments look promising, though a more thorough analysis of the benefits and use cases is necessary. Operating directly on the JPG images instead of the RAW images was tested shortly, since not all cameras allow access to the RAW image data. Although some improvements can be seen, the compression artifacts seem to be insufficiently random. It is possible that better results can be achieved if the sensor noise and thus the randomness of the artifacts is increased by selecting a higher ISO level.

ACKNOWLEDGEMENTS

This paper was supported by a grant (HE 2459/21-1) from the Deutsche Forschungsgemeinschaft (DFG).

REFERENCES

Ballabeni, A., Apollonio, F. I., Gaiani, M. and Remondino, F., 2015. Advances in image pre-processing to improve automated 3d reconstruction. In: *3D-Arch - 3D Virtual Reconstruction and Visualization of Complex Architectures*, pp. 315–323.

Cai, H., 2013. High dynamic range photogrammetry for light and geometry measurement. In: *AEI 2013: Building Solutions for Architectural Engineering*, pp. 544–553.

Dabov, K., Foi, A., Katkovnik, V. and Egiazarian, K., 2007. Image denoising by sparse 3d transform-domain collaborative filtering. *IEEE Trans. Image Process* 16(8), pp. 2080–2095.

De Neve, S., Goossens, B., Luong, H. and Philips, W., 2009. An improved hdr image synthesis algorithm. In: *Image Processing (ICIP), 16th IEEE International Conference on*, pp. 1545–1548.

Debevec, P. E. and Malik, J., 1997. Recovering high dynamic range radiance maps from photographs. In: *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH 97, pp. 369–378.

Furukawa, Y. and Ponce, J., 2010. Accurate, dense, and robust multi-view stereopsis. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 32(8), pp. 1362–1376.

Guidi, G., Gonizzi, S. and Micoli, L. L., 2014. Image preprocessing for optimizing automated photogrammetry performances. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* pp. 145–152.

Kontogianni, G., Stathopoulou, E., Georgopoulos, A. and Doulamis, A., 2015. Hdr imaging for feature detection on detailed architectural scenes. In: *3D-Arch - 3D Virtual Reconstruction and Visualization of Complex Architectures*, pp. 325–330.

Lehtola, V. and Ronnholm, P., 2014. Image enhancement for point feature detection in built environment. In: *Systems and Informatics (ICSAI), 2nd International Conference on*, pp. 774–779.

Ley, A., 2016. Project website. `http://andreas-ley.com/projects/WhiteWalls/`.

Mann, S. and Picard, R. W., 1995. On being 'undigital' with digital cameras: Extending dynamic range by combining differently exposed pictures. In: *Proceedings of IST*, pp. 442–448.

Ntregkaa, A., Georgopoulosa, A. and Quinterob, M., 2013. Photogrammetric exploitation of hdr images for cultural heritage documentation. In: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pp. 209–214.

Robertson, M. A., Borman, S. and Stevenson, R. L., 1999. Dynamic range improvement through multiple exposures. In: *Proc. of the Int. Conf. on Image Processing ICIP'99*, pp. 159–163.

Rothermel, M., Wenzel, K., Fritsch, D. and Haala, N., 2012. Sure: Photogrammetric surface reconstruction from imagery. In: *Proceedings LC3D Workshop, Berlin*.

Wallis, K. F., 1974. Seasonal adjustment and relations between variables. *Journal of the American Statistical Association* 69(345), pp. 18–31.

Wu, C., 2007. Siftgpu: A gpu implementation of scale invariant feature transform (sift). `http://cs.unc.edu/~ccwu/siftgpu`.

Wu, C., 2013. Towards linear-time incremental structure from motion. In: *3D Vision - 3DV 2013, 2013 International Conference on*, pp. 127–134.

Wu, C., Agarwal, S., Curless, B. and Seitz, S. M., 2011. Multicore bundle adjustment. In: *Conference on Computer Vision and Pattern Recognition*, pp. 3057–3064.