Weighted Patch-Based Reconstruction: Linking (Multi-view) Stereo to Scale Space

Ronny Klowsky, Arjan Kuijper, and Michael Goesele

Technische Universität Darmstadt

Abstract. Surface reconstruction using patch-based multi-view stereo commonly assumes that the underlying surface is locally planar. This is typically not true so that least-squares fitting of a planar patch leads to systematic errors which are of particular importance for multi-scale surface reconstruction. In a recent paper [12], we determined the modulation transfer function of a classical patch-based stereo system. Our key insight was that the reconstructed surface is a box-filtered version of the original surface. Since the box filter is not a true low-pass filter this causes high-frequency artifacts. In this paper, we propose an extended reconstruction model by weighting the least-squares fit of the 3D patch. We show that if the weighting function meets specified criteria the reconstructed surface is the convolution of the original surface with that weighting function. A choice of particular interest is the Gaussian which is commonly used in image and signal processing but left unexploited by many multi-view stereo algorithms. Finally, we demonstrate the effects of our theoretic findings using experiments on synthetic and real-world data sets.

Keywords: multi-view stereo, multi-scale surface reconstruction.

1 Introduction

The basis of virtually all multi-view stereo algorithms are correspondences found between images. Hereby, the de facto standard is to find a planar patch in 3D whose projected region in (some of) the images is photo-consistent, i.e., looks similar. There are many ways to measure photo-consistency including normalized cross-correlation (NCC) or the sum of squared differences (SSD, see Hu and Mordohai [10] for an overview and evaluation of different measures). Whatever measurement used, the underlying assumption is that the original surface is locally planar or even has constant depth in the patch area. This leads to a systematic error in reconstruction which becomes especially important when combining multi-scale data [1, 2]. Recently, Klowsky et al. [12] analyzed this systematic error and proposed a reconstruction model where the 3D patch is fitted to the original surface in a least-squares sense. In the resulting linear system they identified the modulation transfer function to be a sinc. In other words, the reconstructed surface is equal to a convolution of the original surface with a box filter. Since this is no true low-pass filter it causes high-frequency artifacts such as amplitude inversion for some frequencies.

A. Kuijper et al. (Eds.): SSVM 2013, LNCS 7893, pp. 234–245, 2013.

In this paper, we develop an extended reconstruction model by weighting the fitting of the 3D patch. We derive constraints on the weighting function to ensure that the reconstructed surface is a convolution of the original surface with that weighting function. As a particular result, we will see that uniform weighting used in our previous work [12] causes the box filter effect. A much better choice for the weighting function fulfilling the derived constraints and allowing for true low-pass filtered reconstructions is the Gaussian, which is widely used in the imaging domain. When using different patch sizes (e.g., due to different image resolution or camera-object distances) the reconstructions reflect different levels of the scale space representation of the true surface. We show for one popular multi-view stereo algorithm [5] how to implement the weighting and discuss results on synthetic as well as real-world data sets. Our findings may influence a broad range of algorithms in multi-view stereo but also in the field of multi-scale surface reconstruction [2–4, 15] or geometry super-resolution [6, 20].

In summary the contributions of our paper are

- the generalization of a previously presented reconstruction model for (multi-view) stereo by introducing weights,
- the theoretical derivation of the (predicted) reconstructed surface without the detour in frequency space, and
- we show how a weighting, e.g., a Gaussian, can be implemented for a common multi-view stereo algorithm which expectably improves the frequency behavior of the reconstruction.

1.1 Related Work

While there is a large body of work on multi-view stereo (see, e.g., the survey paper and the constantly updated benchmark by Seitz at al. [14, 18]), the study of multi-scale depth reconstruction has long been neglected. In previous work we [12] introduced a theoretical reconstruction model and determined the modulation transfer function of patch-based stereo systems. We also discussed the (loosely) related work on multi-scale analysis of (multi-view) stereo to which we refer the reader for a more extensive discussion. Our current work builds upon this reconstruction model and demonstrates how more freedom in the reconstruction outcome is possible. As one particular result, we demonstrate that multi-view stereo can yield a scale space representation of the underlying geometry. In contrast to [12], we derive our results directly in geometry space without operation (at least in an intermediate step) in frequency space.

Our work is also related to existing work on patch-based photo-consistency measures. An overview and evaluation of confidence measures used in (multi-view) stereo is given by Hu and Mordohai [10]. In all their cost computations, however, a square patch of $N \times N$ pixels is used and all pixels are weighted uniformly. If we assume all measures aim at fitting a patch in 3D space, they all result in a box filter. Kanade and Okutomi [11] already tried to find optimal size and shape of the patch but still only used rectangular shapes. Habbecke and Kobbelt [8] propose a multi-view stereo system where matching is performed on

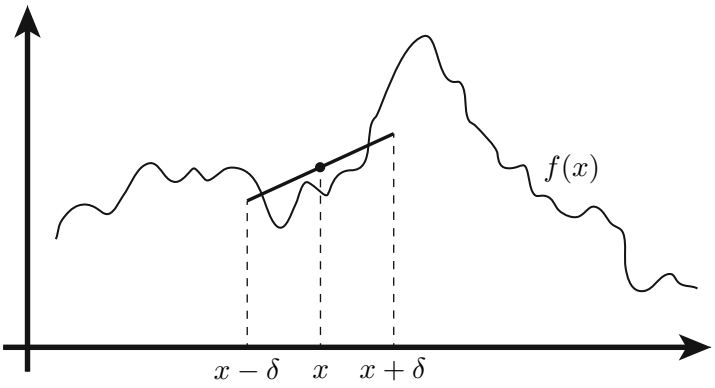

Fig. 1. Fitting a planar patch (line segment) to the geometry for each point x

circular disks in object space. The size of the disks is selected to achieve a minimum intensity variance on each disk. Totally different shapes are achieved by Micusik and Koseka [13] whose approach is suited for man-made environments with many planar surfaces. Here, the reference view is first segmented into superpixels, that are assumed to be planar in object space, and matching is then performed using those superpixels. Thus the shape of the matching window is adapted to the local scene structure and texture. Yoon and Kweon [21] were probably the first to compute weights for each pixel in the patch that steer the influence of that pixel in the matching process. Their weights are dependent on the color similarity and the spatial distance from the center pixel. Hosni et al. [9] improve on that by computing weights using the geodesic distance transform. In contrast to all these efforts, we investigate the influence of a specific weighting on the reconstructed geometry and derive the resulting (multi-scale) behavior of the resulting surface.

2 Theoretical Considerations

2.1 Extension of the Reconstruction Model

In this paper, we build upon our previously introduced reconstruction model [12]. We describe the process of photometric consistency optimization between images (e.g. using normalized cross-correlation (NCC), or sum of squared differences (SSD)) as a geometric least-squares fitting of a planar patch to the unknown geometry. Figure 1 visualizes this idea for a 2D geometry described as a height field $z = f(x)$. To obtain the reconstruction at some point x, a line segment (parameterized by slope m and offset n) with extent 2δ is fitted to the geometry in a least-squares sense minimizing the energy

$$E(m, n, x) = \int_{x-\delta}^{x+\delta} (mt + n - f(t))^2 dt. \tag{1}$$

The reconstructed surface is then represented by the central patch points. For this model we determined the modulation transfer function which turned out to be a sinc. Though not explicitly stated in our prior paper [12] this is equivalent to a convolution with a box filter. In the following we will show that the reason for this result is the uniform weighting of pixels during optimization. We suggest the following extension of the reconstruction model: Instead of considering each point in $[x-\delta, x+\delta]$ uniformly we introduce a weighting function g allowing for different areas of influence. Consequently, we alter the energy function to

$$E(m,n,x) = \int_{-\infty}^{\infty} g(x-t)(mt+n-f(t))^2\,dt \tag{2}$$

where $g(t)$ is a weighting function. Note that with $g(t) = \mathbf{1}_{[-\delta,\delta]}$ this is equal to the former energy in Eq. 1. This weighting function could be implemented as a weighting of the pixels during photo-consistency optimization. In Section 3 we will demonstrate this using a specific multi-view stereo algorithm. In the following subsection, we derive theoretically how this weighting function affects the reconstructed surface.

2.2 Reconstruction in 2D

For the sake of simplicity, we first look at a surface in 2D (a line) as illustrated in Figure 1. For now, we put no further constraints on $g(t)$ except for integrability. Later on, we will discuss further desirable properties. Minimizing E in Equation 2 requires taking the partial derivatives with respect to m and n:

$$\begin{aligned}\partial_m E &= 2\int_{-\infty}^{\infty} g(x-t)t(mt+n-f(t))\,dt \qquad (3)\\ &= 2m\int_{-\infty}^{\infty} g(x-t)t^2\,dt + 2n\int_{-\infty}^{\infty} g(x-t)t\,dt - 2\int_{-\infty}^{\infty} g(x-t)tf(t)\,dt\\ \partial_n E &= 2\int_{-\infty}^{\infty} g(x-t)(mt+n-f(t))\,dt \qquad (4)\\ &= 2m\int_{-\infty}^{\infty} g(x-t)t\,dt + 2n\int_{-\infty}^{\infty} g(x-t)\,dt - 2\int_{-\infty}^{\infty} g(x-t)f(t)\,dt\end{aligned}$$

We introduce a short notation for the zeroth, first and second moment of g

$$\mu_0 = \int_{-\infty}^{\infty} g(t)\,dt \qquad \mu_1(x) = \int_{-\infty}^{\infty} g(x-t)t\,dt \qquad \mu_2(x) = \int_{-\infty}^{\infty} g(x-t)t^2\,dt \tag{5}$$

and abbreviate the other convolution integrals using

$$(g*\cdot f)(x) = \int_{-\infty}^{\infty} g(x-t)tf(t)\,dt \tag{6}$$

$$(g*f)(x) = \int_{-\infty}^{\infty} g(x-t)f(t)\,dt. \tag{7}$$

W.l.o.g. we can assume that $\mu_0 = 1$ which corresponds to normalizing the weighting function g. Under the condition that $\mu_2(x) \neq 0$ we set the partial derivatives to zero and transpose the equations:

$$m = \frac{(g * \cdot f)(x) - n\mu_1(x)}{\mu_2(x)} \tag{8}$$

$$n = (g * f)(x) - m\mu_1(x) \tag{9}$$

We can now solve for m and n which leads to

$$\begin{aligned} m &= \frac{(g * \cdot f)(x) - ((g * f)(x) - m\mu_1(x))\mu_1(x)}{\mu_2(x)} \\ \Leftrightarrow\ m &= \left(1 - \frac{\mu_1(x)^2}{\mu_2(x)}\right)^{-1} \left(\frac{(g * \cdot f)(x)}{\mu_2(x)} - \frac{(g * f)(x)\mu_1(x)}{\mu_2(x)}\right) \\ &= \frac{(g * \cdot f)(x) - (g * f)(x)\mu_1(x)}{\mu_2(x) - \mu_1(x)^2} \end{aligned} \tag{10}$$

$$\begin{aligned} n &= (g * f)(x) - \frac{(g * \cdot f)(x) - (g * f)(x)\mu_1(x)}{\mu_2(x) - \mu_1(x)^2}\mu_1(x) \\ &= \frac{(g * f)(x)\mu_2(x) - (g * \cdot f)(x)\mu_1(x)}{\mu_2(x) - \mu_1(x)^2} \end{aligned} \tag{11}$$

Since the final surface is represented by the central patch points it can be written as

$$mx + n = \frac{(g * \cdot f)(x)(x - \mu_1(x)) + (g * f)(x)(\mu_2(x) - x\mu_1(x))}{\mu_2(x) - \mu_1(x)^2}. \tag{12}$$

Though valid for very general weighting functions g this result is not very satisfactory. On closer inspection we see that when $\mu_1(x) = x$, which is true for all normalized symmetric functions g, it can be easily simplified to

$$mx + n = (g * f)(x). \tag{13}$$

In other words, every function g with $\mu_0 = 1$, $\mu_1(x) = x$, $\mu_2(x) \neq 0$, and $\mu_2(x) \neq x^2$, used to weight the least-squares fitting results in a reconstruction that is the convolution of the true surface with g. Note, that a uniform weighting [12] naturally leads to the convolution with a box filter in this framework.

2.3 Building a Scale Space Representation

The derived constraints for the weighting function obviously allow for many different choices. One of particular interest is the Gaussian since convolutions with Gaussians are well studied and widely applied, e.g., in the image domain. If we set g to be a normalized Gaussian with standard deviation σ

$$g(t) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(\frac{-t^2}{2\sigma^2}\right). \tag{14}$$

we obtain the following moments

$$\mu_0 = 1 \qquad \mu_1(x) = x \qquad \mu_2(x) = \sigma^2 + x^2. \tag{15}$$

That is, the normalized Gaussian fulfills our constraints and we can determine the slope m and offset n of the fitted patch at each point x by

$$m = \frac{(g * \cdot f)(x) - (g * f)(x)x}{\sigma^2} \tag{16}$$

$$n = \frac{(g * f)(x)(\sigma^2 + x^2) - (g * \cdot f)(x)x}{\sigma^2}. \tag{17}$$

In order to create a scale space representation of the underlying surface we need to use Gaussians with varying standard deviations σ. However, during reconstruction we can influence σ only to a limited extent because it depends on the scene depth, image resolution and focal length of the camera. In that sense, if we reconstruct depth maps of the same geometry using a variety of images results in a natural variation of the standard deviation σ in real-world space. The only parameter one can actively steer is the standard deviation σ_i (linked with the window size due to approximation and clamping of the Gaussian) in image space used for patch-based optimization. When selecting σ_i one often has a rough depth estimate and also the camera parameters are known from registration. With that it is possible to indirectly steer the standard deviation σ in world space at least to a limited extent, e.g., for parts of the scene with different depths. In Section 3 we will conduct some experiments with varying the standard deviation σ_i but we first transfer our results into 3D.

2.4 Reconstruction in 3D

For the reconstruction in 3D we assume the 2D geometry is described as a height field $z = f(x, y)$. To obtain the reconstruction at some point (x, y), we fit a patch (surface segment) that is parameterized by 2 slopes m_1 and m_2 and an offset n. Again, the weighting function g allows for different areas of influence. As a result we now have the following energy

$$E(m_1, m_2, n, x) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x-t, y-s)(m_1 t + m_2 s + n - f(t,s))^2 \, dt \, ds. \tag{18}$$

Minimizing E requires taking the partial derivatives with respect to m_1, m_2, and n:

$$\partial_{m_1} E = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} 2tg(x-t, y-s)(m_1 t + m_2 s + n - f(t,s)) \, dt \, ds \stackrel{!}{=} 0 \tag{19}$$

$$\partial_{m_2} E = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} 2sg(x-t, y-s)(m_1 t + m_2 s + n - f(t,s)) \, dt \, ds \stackrel{!}{=} 0 \tag{20}$$

$$\partial_{n} E = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} 2g(x-t, y-s)(m_1 t + m_2 s + n - f(t,s)) \, dt \, ds \stackrel{!}{=} 0 \tag{21}$$

Similar to the reconstruction in 2D, we introduce the short notation μ_{00}, μ_{10}, μ_{01}, μ_{20}, μ_{11}, and μ_{02} for the moments of g with respect to x and y, respectively.

$$\mu_{00} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(t,s)\,dt\,ds, \qquad \mu_{10} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x-t,y-s)t\,dt\,ds \tag{22}$$

$$\mu_{01} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x-t,y-s)s\,dt\,ds, \quad \mu_{20} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x-t,y-s)t^2\,dt\,ds \tag{23}$$

$$\mu_{11} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x-t,y-s)st\,dt\,ds, \; \mu_{02} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x-t,y-s)s^2\,dt\,ds \tag{24}$$

For the sake of clarity we chose an even shorter abbreviation for the other convolution integrals:

$$\mathrm{gtf} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} tg(x-t,y-s)f(t,s)\,dt\,ds \tag{25}$$

$$\mathrm{gsf} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} sg(x-t,y-s)f(t,s)\,dt\,ds \tag{26}$$

$$\mathrm{gf} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x-t,y-s)f(t,s)\,dt\,ds. \tag{27}$$

Again, we can normalize g such that $\mu_{00} = 1$. With this notation we can rewrite Eqs. (19)-(21) as

$$\partial_{m_1} E = 2(m_1\mu_{20} + m_2\mu_{11} + n\mu_{10} - \mathrm{gtf}) \stackrel{!}{=} 0 \tag{28}$$

$$\partial_{m_2} E = 2(m_1\mu_{11} + m_2\mu_{02} + n\mu_{01} - \mathrm{gsf}) \stackrel{!}{=} 0 \tag{29}$$

$$\partial_{n} E = 2(m_1\mu_{10} + m_2\mu_{01} + n - \mathrm{gf}) \stackrel{!}{=} 0 \tag{30}$$

Solving these equations for m_1, m_2, and n yields

$$\alpha m_1 = \mathrm{gf}\,(\mu_{02}\mu_{10} - \mu_{01}\mu_{11}) + \mathrm{gsf}\,(\mu_{11} - \mu_{01}\mu_{10}) + \mathrm{gtf}\left(\mu_{01}^2 - \mu_{02}\right) \tag{31}$$

$$\alpha m_2 = \mathrm{gf}\,(\mu_{01}\mu_{20} - \mu_{10}\mu_{11}) + \mathrm{gsf}\left(\mu_{10}^2 - \mu_{20}\right) + \mathrm{gtf}\,(\mu_{11} - \mu_{01}\mu_{10}) \tag{32}$$

$$\alpha n = \mathrm{gf}\left(\mu_{11}^2 - \mu_{02}\mu_{20}\right) + \mathrm{gsf}\,(\mu_{01}\mu_{20} - \mu_{10}\mu_{11}) + \mathrm{gtf}\,(\mu_{02}\mu_{10} - \mu_{01}\mu_{11}) \tag{33}$$

where $\alpha = \mu_{20}\mu_{01}^2 - 2\mu_{10}\mu_{11}\mu_{01} + \mu_{02}\mu_{10}^2 + \mu_{11}^2 - \mu_{02}\mu_{20}$. Plugging in these expressions in the patch $P = m_1 x + m_2 y + n$, we obtain

$$P = \frac{1}{\alpha}\left(\mathrm{gf}\left(\mu_{11}^2 - \mu_{02}\mu_{20} - \mu_{01}\mu_{11}x + \mu_{02}\mu_{10}x - \mu_{10}\mu_{11}y + \mu_{01}\mu_{20}y\right) + \right. \tag{34}$$

$$\mathrm{gsf}\left(-\mu_{11}\mu_{10} + \mu_{01}\mu_{20} - \mu_{01}\mu_{10}x + \mu_{11}x + \mu_{10}^2 y - \mu_{20}y\right) + \tag{35}$$

$$\left.\mathrm{gtf}\left(-\mu_{11}\mu_1 + \mu_{02}\mu_{10} + \mu_{01}^2 x - \mu_{02}x - \mu_{10}\mu_{01}y + \mu_{11}y\right)\right). \tag{36}$$

Taking symmetric filters yields $\mu_{10} = x$ and $\mu_{01} = y$. Then immediately one gets

$$P = \mathrm{gf} \tag{37}$$

Of course we can use a classical anisotropic Gaussian characterized by σ and τ

$$g(t,s) = \frac{1}{2\pi\sigma\tau} \exp\left(\frac{-t^2}{2\tau^2} + \frac{-s^2}{2\sigma^2}\right) \tag{38}$$

because the moments are $\mu_{00} = 1$, $\mu_{10} = x$, $\mu_{01} = y$, $\mu_{02} = x^2 + \tau^2$, $\mu_{11} = xy$, $\mu_{02} = y^2 + \sigma^2$.

3 Experiments

In order to verify our theoretic findings in practice we now conduct some experiments. We hereby chose the depth map reconstruction method of Goesele et al. [5] because it does a pure photo-consistency optimization (going back to Gruen and Baltsavias [7]) to find depth and normal for a certain pixel and has no regularization force. For a small region around a pixel i, j in a reference view I_R the method aims to find depth d and normal $\boldsymbol{n}$ of the associated 3D patch such that it is photo-consistent with a set of neighboring views I_k. The algorithm minimizes (see [5, Sec. 6.2] ignoring the color scale)

$$\sum_{k,i,j} [I_R(s+i, t+j) - I_k(P_k^{d,\boldsymbol{n}}(s+i, t+j))]^2 \tag{39}$$

where P_k describes the projection of a pixel from the reference view in the neighbor view I_k according to some depth d and normal $\boldsymbol{n}$. We implement the weighting on the least-squares patch fit by weighting the pixels, i.e., we compute a weighted SSD:

$$\sum_{k,i,j} g(i,j)[I_R(s+i, t+j) - I_k(P_k^{d,\boldsymbol{n}}(s+i, t+j))]^2. \tag{40}$$

The remaining question is whether this weighted photo-consistency optimization still reflects the process of weighted least-squares fitting as described by Eq. 2. We test this using a synthetic data set because of two reasons: First, we can assure that our results are not affected by registration errors but solely reflect the photometric consistency optimization, and second, we know the ground truth surface and are able to compute the predicted reconstruction according to our model. Our ground truth surface is created as a random sum of one-dimensional B-Splines extruded into the third dimension. We then render five different views (one central view looking perpendicular onto the surface and four views distributed uniformly around it with a parallax of 35°) of this scene using the PBRT system [16] while a random texture is mapped onto the surface to guarantee matching success at all pixels (see Fig. 2). For the central view we now reconstruct a depth map by using the other four views as neighbors and minimizing

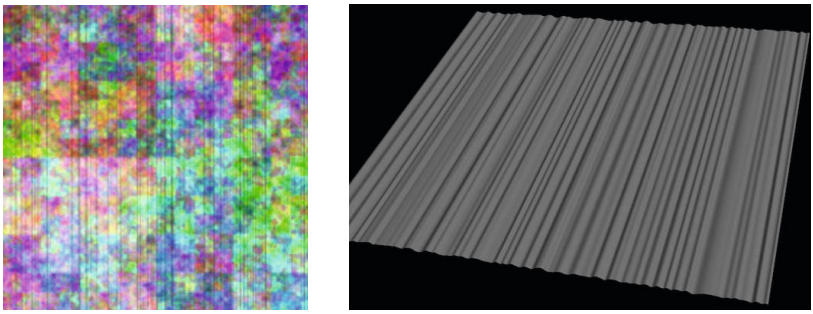

Fig. 2. Left: The central view of our synthetic data set. Right: The underlying mesh (shaded) used to render the views.

Table 1. Mean deviation of the reconstruction from the theoretical predicted surface (see Figs. 3&4)

Patch size	mean deviation (L_1-norm)	
in pixels	uniform weighting	Gaussian weighting
11×11	$1.9 \cdot 10^{-4}$	$1.3 \cdot 10^{-4}$
21×21	$4.1 \cdot 10^{-4}$	$2.8 \cdot 10^{-4}$
41×41	$6.9 \cdot 10^{-4}$	$5.8 \cdot 10^{-4}$
61×61	$6.3 \cdot 10^{-4}$	$7.0 \cdot 10^{-4}$

the weighted SSD from Eq. 40. We start the optimization for each pixel with the depth value obtained from PBRT and the normal representing a fronto-parallel patch. To reduce noise we average the reconstructed values along the constant dimension. Fig. 3 shows the reconstructions using a uniform weighting function. The quadratic windows in image space are 11 (blue), 21 (green), 41 (red), and 61 (cyan) pixels wide which corresponds to a patch size (2δ) of 0.06,0.12,0.24, and 0.36 in world coordinates, respectively. We also plotted the predicted reconstructions, i.e., convolutions of the original surface with box filters of the corresponding width. Overall, the reconstruction is close to the prediction although there is some local deviation. The best conformity is achieved for the small patch size which can also be seen in Table 1 where we computed the mean deviation. Note the occasional amplitude inversion visible in the prediction as well as the reconstruction, in particular for the largest filter at around -1.4.

In Fig. 4 we used Gaussian weighting with increasing standard deviation which leads to a scale space representation of the underlying surface. The window sizes are the same used for the uniform weighting and we always chose the standard deviation σ such that $\delta = 2.5\sigma$. That is, in world coordinates we used $\sigma = 0.012, 0.024, 0.048, 0.072$. We can see from the figure and also by studying the numbers in Table 1 that the deviation from the prediction again increases for larger σ.

Finally, we show reconstruction results on real world data. Figure 5 (top left) shows an input image of the Notre Dame data set consisting of 715 images

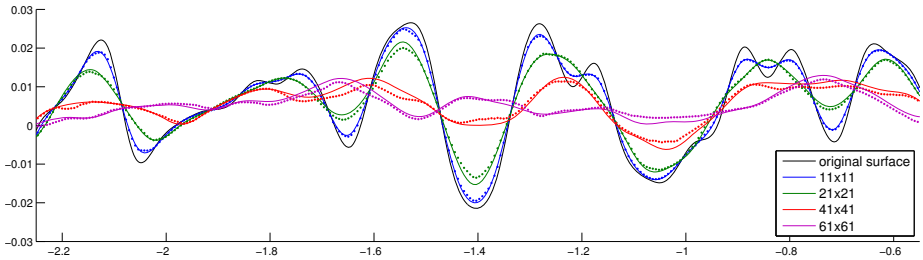

Fig. 3. Multi-view stereo reconstruction using a uniform weighting with increasing patch size. The black line denotes the original surface. The colored solid lines are the computed predictions while the corresponding dots are the reconstructed values.

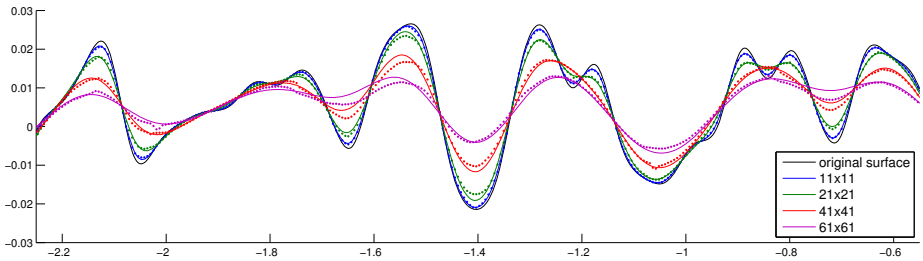

Fig. 4. Reconstructing a scale space representation using a Gaussian weighting with increasing standard deviation (see text). The black line denotes the original surface. The colored solid lines are the computed predictions while the according dots are the reconstructed values.

downloaded from the Internet. We use Snavely et al. [19] to register them and compute depth maps for the shown image using different weightings and window sizes. The middle and bottom row show reconstructions obtained using uniform and Gaussian weighting, respectively. Although hard to jugde, the Gaussian weighting seems to produce slightly more noise and less complete reconstructions. On the other hand it better preserves the low frequencies. One must consider though, that the algorithm [5] was tuned to work well with the uniform weighting and on a broad range of data sets. That is, playing with the parameters in the optimization or view selection might result in more favorable results for the Gaussian weighting.

4 Conclusion and Future Work

This paper extends a recently introduced model for patch-based depth reconstruction by adding a weighting function. We derive criteria on the weighting function such that we can predict the reconstructed surface as the convolution of the true surface with the applied weighting function. This includes using a Gaussian instead of a uniform weighting during reconstruction which corresponds to a

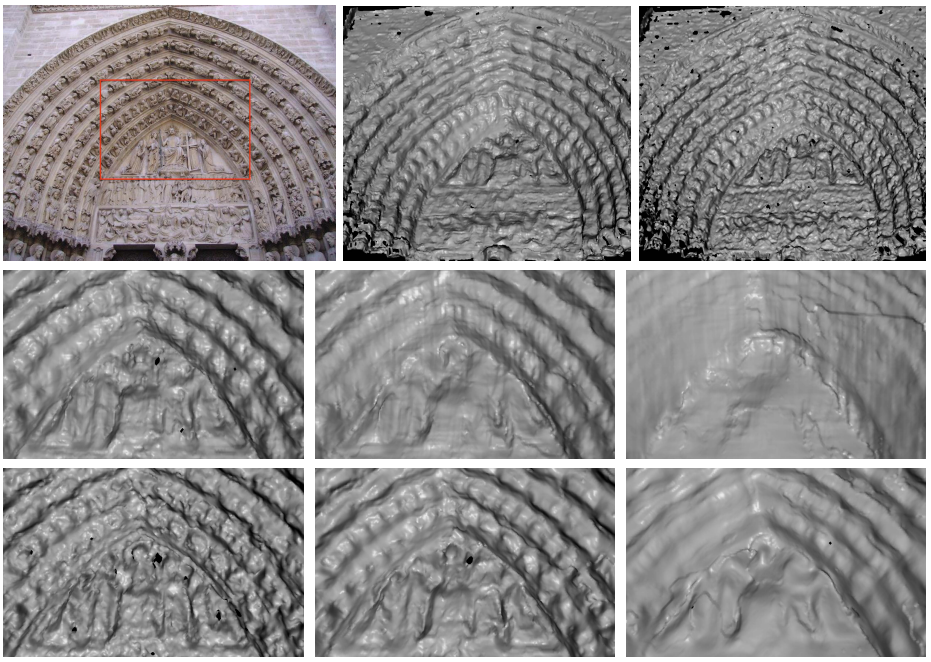

Fig. 5. *Top left*: Input image of the Notre Dame data set. The red box is roughly the area seen in the bottom rows. *Top middle,right*: Full rendered view of reconstructed depth map using uniform (middle) and Gaussian weighting (right) and a window size in images space of 7×7 pixels. *Middle+Bottom*: Enlarged area roughly corresponding to red box (top left) of the reconstructed depth map. We applied uniform (middle) and Gaussian weighting (bottom) using window sizes of 7×7, 11×11, and 21×21 pixels (from left to right) for reconstruction where the standard deviation of the Gaussian in image space is $\sigma_i = 1.2, 2.0, 4.0$.

Gaussian instead of a box filter in geometry space. In contrast to previous methods, we achieve a true low-pass filter avoiding the introduction of systematic high-frequency artifacts. Future work definitely includes to further investigate the correlation between weighted photo-consistency optimization and weighted least-squares fitting of a planar patch to the geometry.

Our findings are applicable in a broad range of applications. In contrast to [12], we give a local characterization of the reconstruction outcome at the same time offering more flexibility caused by the weighting. Multi-scale surface reconstruction methods like [2–4, 15] could take that knowledge into account when combining data from multiple depth maps. But also geometry super-resolution methods [6, 20] can benefit from our findings. Since we provide evidence for a generative model it is now possible to adapt well established methods from imaging, e.g., Bayesian super-resolution [17], to the geometry reconstruction context.

Acknowledgements. This work was supported in part by the DFG Emmy Noether fellowship GO 1752/3-1.

References

1. Bellocchio, F., Borghese, N.A., Ferrari, S., Piuri, V.: 3D Surface Reconstruction: Multi-Scale Hierarchical Approaches. Springer, New York (2013)
2. Fuhrmann, S., Goesele, M.: Fusion of depth maps with multiple scales. In: SIGGRAPH Asia (2011)
3. Furukawa, Y., Curless, B., Seitz, S.M., Szeliski, R.: Towards internet-scale multi-view stereo. In: CVPR (2010)
4. Gargallo, P., Sturm, P.: Bayesian 3D modeling from images using multiple depth maps. In: CVPR (2005)
5. Goesele, M., Snavely, N., Curless, B., Hoppe, H., Seitz, S.M.: Multi-view stereo for community photo collections. In: ICCV (2007)
6. Goldlücke, B., Cremers, D.: A superresolution framework for high-accuracy multi-view reconstruction. In: Denzler, J., Notni, G., Süße, H. (eds.) DAGM 2009. LNCS, vol. 5748, pp. 342–351. Springer, Heidelberg (2009)
7. Gruen, A., Baltsavias, E.P.: Geometrically constrained multiphoto matching. Photogrammetric Engineering & Remote Sensing 54(5), 633–641 (1988)
8. Habbecke, M., Kobbelt, L.: A surface-growing approach to multi-view stereo reconstruction. In: CVPR (2007)
9. Hosni, A., Bleyer, M., Gelautz, M., Rhemann, C.: Local stereo matching using geodesic support weights. In: ICIP (2009)
10. Hu, X., Mordohai, P.: A quantitative evaluation of confidence measures for stereo vision. PAMI 34(11), 2121–2133 (2012)
11. Kanade, T., Okutomi, M.: A stereo matching algorithm with an adaptive window: Theory and experiment. PAMI 16(9), 920–932 (1994)
12. Klowsky, R., Kuijper, A., Goesele, M.: Modulation transfer function of patch-based stereo systems. In: CVPR (2012)
13. Micusik, B., Kosecka, J.: Multi-view superpixel stereo in man-made environments. Tech. rep., Dept. Computer Science, George Mason University (2008)
14. Middlebury multi-view stereo evaluation, `http://vision.middlebury.edu/mview/`
15. Mücke, P., Klowsky, R., Goesele, M.: Surface reconstruction from multi-resolution sample points. In: VMV (2011)
16. Physically based rendering, `http://www.pbrt.org`
17. Pickup, L., Capel, D., Roberts, S., Zisserman, A.: Bayesian methods for image super-resolution. The Computer Journal 52(1), 101–113 (2007)
18. Seitz, S.M., Curless, B., Diebel, J., Scharstein, D., Szeliski, R.: A comparison and evaluation of multi-view stereo reconstruction algorithms. In: CVPR (2006)
19. Snavely, N., Seitz, S.M., Szeliski, R.: Skeletal sets for efficient structure from motion. In: CVPR (2008)
20. Yang, Q., Yang, R., Davis, J., Nister, D.: Spatial-depth super resolution for range images. In: CVPR (2007)
21. Yoon, K.J., Kweon, I.S.: Locally adaptive support-weight approach for visual correspondence search. In: CVPR (2005)