

A Superresolution Framework for High-Accuracy Multiview Reconstruction

Bastian Goldlücke and Daniel Cremers

Computer Science Department
University of Bonn, Germany
{bastian,dcremers}@cs.uni-bonn.de

Abstract. We present a variational approach to jointly estimate a displacement map and a superresolution texture for a 3D model from multiple calibrated views. The superresolution image formation model leads to an energy functional defined in terms of an integral over the object surface. This functional can be minimized by alternately solving a deblurring PDE and a total variation minimization on the surface, leading to increasingly accurate estimates of photometry and geometry, respectively. The resulting equations can be discretized and solved on texture space with the help of a conformal atlas. The superresolution approach to texture reconstruction allows to obtain fine details in the texture map which surpass individual input image resolution.

1 Introduction

Modern image-based 3D reconstruction algorithms achieve high levels of geometric accuracy. However, due to intrinsic limitations like voxel volume resolution and mesh size, or limits imposed by the application, the geometric resolution of the model is usually well below the pixel resolution in a rendering. This leads to a number of problems if one wants to estimate a texture for the model from the camera images. Mainly, since geometry is never perfectly accurate, the image registration cannot be exactly correct, which leads to a blurry estimated texture, Fig. 1. Consequently, previous methods on texture generation usually employ some form of additional registration before estimating texel color [1–3].

In methods fitting a local lighting model on a per-texel basis, it is generally true that the fewer source cameras influence the result for a single texel, the sharper the resulting texture will be. However, if only the contributions of few cameras are blended for a given texture patch, it is likely that seams and discontinuities arise at visibility boundaries, so some form of stitching has to take place to smoothen the result [4, 5]. Furthermore, not using all available source images implies discarding a lot of potentially useful information.

The superresolution framework presented in this paper is designed to alleviate these problems. We account for the interdependency of geometry and photometry by minimizing a single functional with respect to both a displacement map as well as a super-resolved texture. The image formation model is based on current state-of-the-art super-resolution frameworks [6–8], for which there is a well-developed theory [9]. Because



Fig. 1: From left to right: (a),(b) Two out of fourty input-images for a multiview reconstruction. (c) Close-up of one of the low-resolution input images. (d) Rendered model with blurry texture initialized by weighted averaging of input images. (e) High-quality texture optimized with the proposed superresolution approach.

every patch of the surface is captured from several cameras, by adopting this model we are able to recover the texture in higher resolution and level of detail than provided by the input images. By design, the method scales very well with the number f input cameras: more cameras will always lead to a more accurate solution.

While we introduced superresolution textures in [10], the presented framework is the first formulation for joint geometry optimization and superresolution texture estimation in multiview stereo. The resulting models are of excellent quality and can be rendered from arbitrary viewpoints. The displacement map helps with high-fidelity re-lighting.

2 Displacement Maps and Texture Superresolution

In this section, we introduce a superresolution image formation model, where the camera images depend on the unknown displacement map and texture. The model induces an energy functional which is minimized by the desired optimal maps. Let $\mathcal{I}_1, \dots, \mathcal{I}_n : \Omega_i \rightarrow \mathbb{R}$ be the input images captured by cameras with known projections $\pi_1, \dots, \pi_n : \mathbb{R}^3 \rightarrow \mathbb{R}^2$. The cameras observe a known Lambertian surface $\Sigma \subset \mathbb{R}^3$, which is textured with the unknown texture map $T : \Sigma \rightarrow \mathbb{R}$. At each point of the surface, we allow a small displacement of the geometry in normal direction. This displacement is given by a displacement map $D : \Sigma \rightarrow \mathbb{R}$, the second unknown in the model.

2.1 Variational Formulation

The basic idea is to recover both the unknown texture map as well as the displacement map as the minimizer of a joint energy functional, consisting of a data term and a regularization term for both maps,

$$E(T, D) := E_{\text{data}}(T, D) + E_{\text{tv}}(T, D),$$

$$\text{with } E_{\text{tv}}(T) := \int_{\Sigma} \sigma_t \|\nabla_{\Sigma} T\|_{\Sigma} + \sigma_d \|\nabla_{\Sigma} D\|_{\Sigma} \, ds. \quad (1)$$

Here, $\sigma_t, \sigma_d \geq 0$ are parameters controlling the desired smoothness of the texture and displacement map, respectively. Reasonable choices are $\sigma_t, \sigma_d = 1$. The differential operators on the surface and the norm on the tangent space are explained in detail in [11].

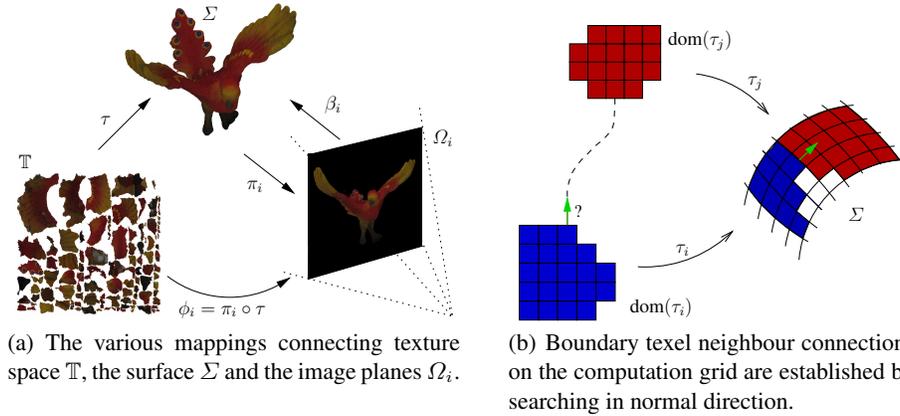


Fig. 2: Texture space and computation grid.

The total variation norm of the texture was chosen as the regularizer, because compared to alternatives, it is better suited to preserve a crisp texture with sharp high-resolution features.

The data term is based on the current state-of-the-art superresolution model [8], with the limitation that we currently do not take noise in the input images into account explicitly. The idea is that a real-world camera downsamples the input by integrating over the visible texels inside each sensor element. This integration process is modeled with a convolution kernel b , which can be derived from the properties of the camera. Possible choices are discussed in [9], we use a Gaussian with standard deviation of half the pixel size. The resulting data term is

$$E_{\text{data}}(T, D) := \sum_{i=1}^n \int_{\hat{S}_i} (b * (T \circ \beta_i^D) - \mathcal{I}_i)^2 dx. \quad (2)$$

$T \circ \beta_i^D$ denotes the visible texture intensity of the high-resolution input in the image plane. The backprojection mappings $\beta_i^D : S_i \rightarrow \Sigma$ assign the visible point on the surface to each point in the silhouettes $S_i := \pi_i(\Sigma) \subset \mathbb{R}^2$, Fig. 2(a). Note that this backprojection depends on D . Actual integration takes place over the smaller subset $\hat{S}_i \subset S_i$ where all of the kernel covers only points within the silhouette.

2.2 Transformation of the Data Term to the Surface

In order to find a local minimum of the energy, it is necessary that integration for both data and regularization term takes place over the surface. A straightforward transformation of the integral yields

$$E_{\text{data}}(T, D) = \sum_{i=1}^n \int_{\Sigma} v_i^D (\mathcal{J}_i^D \mathcal{E}_i^2) \circ \pi_i^D ds. \quad (3)$$

Here, the error images \mathcal{E}_i are defined for abbreviation as the difference between the current rendering of the object and the original images,

$$\mathcal{E}_i := b * (T \circ \beta_i^D) - \mathcal{I}_i. \quad (4)$$

The binary functions $v_i^D : \Sigma \rightarrow \{0, 1\}$ indicate visibility of a surface point in an image,

$$v_i^D(s) := \begin{cases} 1 & \text{if } \pi_i^D(s) \in \hat{S}_i \text{ and } s = \beta_i^D \circ \pi_i^D(s), \\ 0 & \text{otherwise.} \end{cases} \quad (5)$$

Finally, \mathcal{J}_i^D is the inverse surface area element with respect to the backprojection,

$$\mathcal{J}_i^D(x, y) = \left\| \frac{\partial \beta_i^D}{\partial x} \times \frac{\partial \beta_i^D}{\partial y} \right\|^{-1}. \quad (6)$$

\mathcal{J}_i accounts for foreshortening of the surface in the input views, and is small in regions where the backprojection varies strongly, which is usually the case at silhouette boundaries or discontinuities of the backprojection due to self-occlusions. As a consequence, we have the desirable property that in those regions where texture information from the image is unreliable, the input is assigned less weight. Note that we did not need any heuristic assumptions to arrive at this weighting scheme. It is rather a direct mathematical consequence of the variational formulation.

While in general \mathcal{J}_i^D and v_i^D depend on the displacement D , in the following we approximate both by $\mathcal{J}_i := \mathcal{J}_i^0$ and $v_i := v_i^0$. This simplification is necessary to make the computation of a local minimum technically feasible.

2.3 Solving for the Superresolution Texture

The first step is to keep the displacement D constant and solve for the superresolution texture T . To this end, we minimize the functional

$$E(T) = \int_{\Sigma} \|\nabla_{\Sigma} T\|_{\Sigma} + \sum_{i=1}^n \left(\frac{v_i}{\sigma_t} (\mathcal{J}_i \mathcal{E}_i^2) \circ \pi_i^D \right) ds. \quad (7)$$

by solving the Euler-Lagrange equation

$$\operatorname{div}_{\Sigma} \left(\frac{\nabla_{\Sigma} T}{\|\nabla_{\Sigma} T\|_{\Sigma}} \right) + \sum_{i=1}^n \frac{v_i}{\sigma_t} (\mathcal{J}_i (\bar{b} * \mathcal{E}_i)) \circ \pi_i^D = 0, \quad (8)$$

which is a PDE on the surface Σ . The mirrored kernel $\bar{b}(x) := b(-x)$ stems from directional derivative of the convolution operation [12]. After transformation to 2D texture space, the Euler-Lagrange equation can be solved via a gradient descent scheme resembling a deblurring process.

2.4 Solving for the Displacement Map

In the second optimization step, we keep the texture constant. Thus, the functional to be minimized for the displacement map is

$$E(D) = \int_{\Sigma} \|\nabla_{\Sigma} D\|_{\Sigma} + \sum_{i=1}^n \left(\frac{v_i}{\sigma_d} (\mathcal{J}_i \mathcal{E}_i^2) \circ \pi_i^D \right) ds. \quad (9)$$

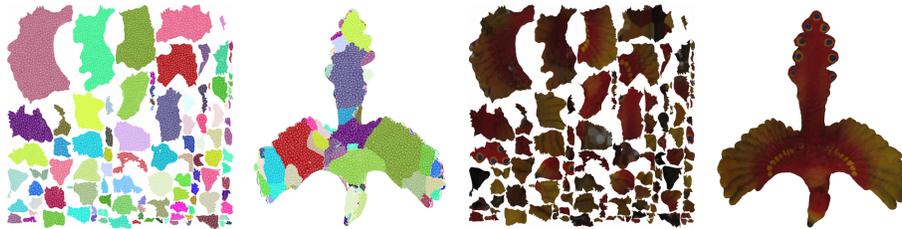


Fig. 3: Illustration of the charts and parametrization mappings. From left to right: (a) Chart domains $\text{dom}(\tau_j)$ in \mathbb{R}^2 forming the texture space \mathbb{T} . (b) Corresponding regions C_j on the surface. (c) Texture map $\mathcal{T} = T \circ \tau_j$ on texture space. (d) Texture T mapped on surface.

Because the data term is non-convex and no good initialization is readily available, we minimize it with a different approach. We make the simplifying assumption that for each point, D is constant in the sampling area of the kernel b , so the energy takes the form

$$E(D) = \int_{\Sigma} \|\nabla_{\Sigma} D\|_{\Sigma} + \rho_s \circ D \, ds. \quad (10)$$

with a point-wise data term ρ_s . Based on [13], we introduce an auxiliary variable U and decouple the regularization from the point-wise optimization by defining a convex approximation to the energy,

$$E(U, D) = \int_{\Sigma} \|\nabla_{\Sigma} U\|_{\Sigma} + \frac{1}{2\theta}(U - D)^2 + \rho_s \circ D \, ds. \quad (11)$$

For $\theta \rightarrow 0$, the solution of this auxiliary problem approaches the solution to the original problem, as the coupling term forces U to be close to D . The idea is that for fixed U , we can perform a point-wise optimization in D , since no spatial derivatives of D appear in the functional. On the other hand, for fixed D , the resulting energy functional resembles the ROF model, which is convex and thus can also be optimized globally. Thus, by alternating two global optimization steps, one can arrive at a good minimizer for the original energy (9), which will however in the general case be only a local minimum.

3 PDE-based Energy Minimization on Texture Space

In order to obtain a high-resolution representation of texture and displacement map, we require a global parametrization of the surface and define texture und displacement map on a grid in 2D space. As the goal is to solve a PDE on the surface, it is desirable to have the parametrization conformal, because then one gets a particularly simple representation of the differential operators [14, 15]. Our method to compute a conformal atlas is a straightforward implementation of [14]. It is fully automatic and has the desirable property that chart boundaries tend to coincide with high-curvature edges on the surface.

Start with $\mathcal{D} = 0$ and set \mathcal{T} to the per-textel weighted average of the projected pixel colors in the input views, with weights given by the backprojection area element \mathcal{J}_i . Then, alternate between solving the following two optimization problems until convergence.

Superresolution texture optimization: Keep \mathcal{D} fixed and via gradient descent, obtain a texture map $\mathcal{T} : \mathbb{T} \rightarrow \mathbb{R}$ which satisfies the Euler-Lagrange equation

$$\frac{1}{\lambda} \operatorname{div} \left(\sqrt{\lambda} \frac{\nabla \mathcal{T}}{\|\nabla \mathcal{T}\|} \right) + \sum_{i=1}^n \frac{v_i}{\sigma_i} \left((\mathcal{J}_i \mathcal{E}_i) \circ \phi_i^{\mathcal{D}} \right) = 0 \quad (12)$$

on the chart domains $\operatorname{dom}(\tau_j)$, $j = 1, \dots, k$. Here, $\phi_i^{\mathcal{D}} := \pi_i^{\mathcal{D}} \circ \tau$ are the mappings from texture space into the image planes, taking into account the current displacement. λ assigns the conformal factor of the parametrization to each point in \mathbb{T} , and $v_i := v_i \circ \tau$ indicates visibility of a texel in image i . \mathcal{E}_i and \mathcal{J}_i are defined according to Eqns. (??) and (??), respectively.

Displacement map optimization: Keep \mathcal{T} fixed and alternate between solving the following two optimization problems until convergence.

- For \mathcal{D} fixed, find the solution \mathcal{U} for the Euler-Lagrange equation of the ROF model,

$$\frac{1}{\lambda} \operatorname{div} \left(\sqrt{\lambda} \frac{\nabla \mathcal{U}}{\|\nabla \mathcal{U}\|} \right) + \frac{1}{\theta} (\mathcal{U} - \mathcal{D}) = 0 \quad (13)$$

via gradient descent.

- For \mathcal{U} fixed and every $x \in \mathbb{T}$, find the global optimum $\mathcal{D}(x)$ of

$$(\mathcal{U}(x) - \mathcal{D}(x))^2 + \rho_{\tau(x)}(\mathcal{D}(x)) \quad (14)$$

using a complete search in the allowed displacement range.

Fig. 4: Algorithm for joint displacement map and superresolution texture.

3.1 Conformal Maps and Differential Operators

Assume for the following that we have a collection of k charts (C_j, τ_j) with chart areas $C_j \subset \Sigma$ and mappings $\tau_j : \operatorname{dom}(\tau_j) \rightarrow C_j$. The union $\mathbb{T} := \cup_{j=1}^k \operatorname{dom}(\tau_j)$ of the chart domains is called the texture space, and to simplify notation, the single mappings τ_j are combined to form a global mapping $\tau : \mathbb{T} \rightarrow \Sigma$. Fig. 3 illustrates the concept. Since the parametrization is conformal, the Jacobian of each τ_j is everywhere a scalar λ , called the conformal factor, times a rotation matrix. Let $\lambda : \mathbb{T} \rightarrow \mathbb{R}$ be the mapping assigning the conformal factor to each point in texture space. Then the smoothness term of the Euler-Lagrange equation (8) can be expressed by pulling it back onto texture space [11]:

$$\operatorname{div}_{\Sigma} \left(\frac{\nabla_{\Sigma} \mathcal{T}}{\|\nabla_{\Sigma} \mathcal{T}\|_{\Sigma}} \right) = \frac{1}{\lambda} \operatorname{div} \left(\sqrt{\lambda} \frac{\nabla \mathcal{T}}{\|\nabla \mathcal{T}\|} \right), \quad (15)$$

where $\mathcal{T} := \mathcal{T} \circ \tau$ is the texture map of the surface defined on the texture space \mathbb{T} , see Fig. 3(c). An analogous expression holds $\mathcal{U} := \mathcal{U} \circ \tau$ in the gradient descent equation of the ROF model, Eq. ???. We also define the displacement map $\mathcal{D} := \mathcal{D} \circ \tau$ on texture space.

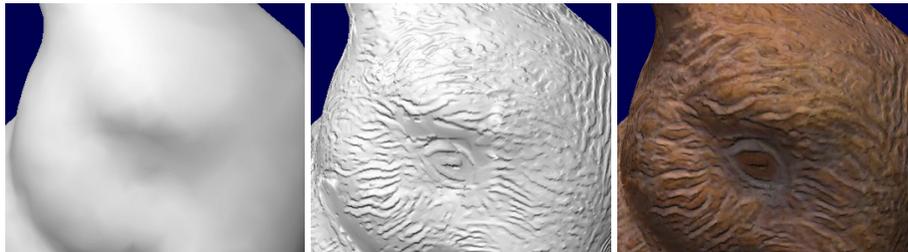


Fig. 5: Estimated displacement map for the *Bunny* dataset. From left to right: (a) Rendering with Gouraud shading. The underlying mesh has low geometric detail. (b) Normal map lighting showing improved geometric detail from the estimated displacement map. (c) Rendering with superresolution texture and normal map lighting.

3.2 Discretization and Computation Grid

For discretization, the texture space is subdivided into a grid of texels. The grid needs to admit a flexible topology, since only texels in the interior of chart domains are connected to their direct neighbours. On the boundary of charts, neighbourhood is established according to the correct relationships on the surface. To achieve this, we take the outer normals of boundary texels in texture space and transform them up onto the surface. Then, we search for the closest texel of the neighboring chart in this direction, and assign that one as a neighbour, Fig. 2(b). For discretization of (15), we employ the scheme from [16] which offers improved rotation invariance. The diffusion tensor G is set to the (isotropic) regularized TV flow,

$$G = \frac{\sqrt{\lambda}}{\max(\epsilon, \|\nabla T\|)} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad (16)$$

where $\epsilon > 0$ is a small regularization parameter.

3.3 Final Algorithm Implementation

Summarizing the results from the previous sections, in order to arrive at an optimal displacement map \mathcal{D} and texture \mathcal{T} , we need to solve the problem described in Fig. 4. For reasonable performance, an efficient parallelized implementation of the terms occurring in the equations is crucial. The most time-consuming part is to compute the backprojection mapping β_i^D , for which we raytrace the surface using CUDA, employing the algorithm in [17] to account for the displacement. \mathcal{J}_i is obtained numerically from the backprojection via Eq. (6) using central differences. Having available the backprojection, we can easily perform the rendering $T \circ \beta_i^D$ of the surface into the i th view using the current displacement map and texture. For this, we just need to color each image pixel x with the texel color of the corresponding surface point $\beta_i^D(x)$.

Note that while the model is formulated for grayscale textures, it can readily be extended to color using a multidimensional total variation norm [18] or color image diffusion [19].



(a) Per-texel weighted average (b) Superresolution only (c) With displacement map

Fig. 6: While the superresolution texture estimate (b) already improves over the commonly used weighted average (a), the jointly estimated displacement map leads to a much more detailed result (c).

4 Experiments

We performed experiments on three different real-world datasets at input image resolution 768×584 , see Fig. 7. An initial 3D reconstruction was obtained using an implementation of the algorithm in [20]. For the initial texture map, each texel was assigned the weighted average color in each camera, with weights given by the backprojection area element. An optimized displacement map and texture was computed using the proposed algorithm, which takes about 5 hours until convergence, running on a 2.8 GHz Core 2 Duo processor with CUDA enhancements running on a GeForce GTX. Main memory required is around 6 GByte. All parameters are set according to the recommendations in the previous sections, and remained the same for all data sets.

Fig. 6 shows that the initial texture is very blurry due to small inaccuracies in the geometry, and can already be improved significantly just by applying the superresolution texture reconstruction. Only when including small scale displacements from the estimated displacement map, however, almost perfect sharpness can be achieved. A rendering of the final textured model is of at least similar quality than an input image from the same viewpoint, and in many cases the level of detail is even exceeded, see Fig. 7. The displacement map can be leveraged to include additional effects into the rendering, like relighting used the derived normal map, as exemplified in Fig. 5.

5 Conclusion

We proposed the first superresolution approach to multiview reconstruction. Based on a unifying and elegant mathematical formalism, an algorithm for jointly estimating a displacement map as well as a high-quality texture for an approximate 3D model was derived. Both unknowns appear as the solutions to PDEs on the input surface, which can be solved via total variation minimization techniques on planar 2D texture space with the help of a conformal atlas. Experiments on several real-world objects demonstrate that the resulting displacement map improves the accuracy of the geometric model. Moreover, the computed superresolved textures typically exhibit more visible details than individual input images.

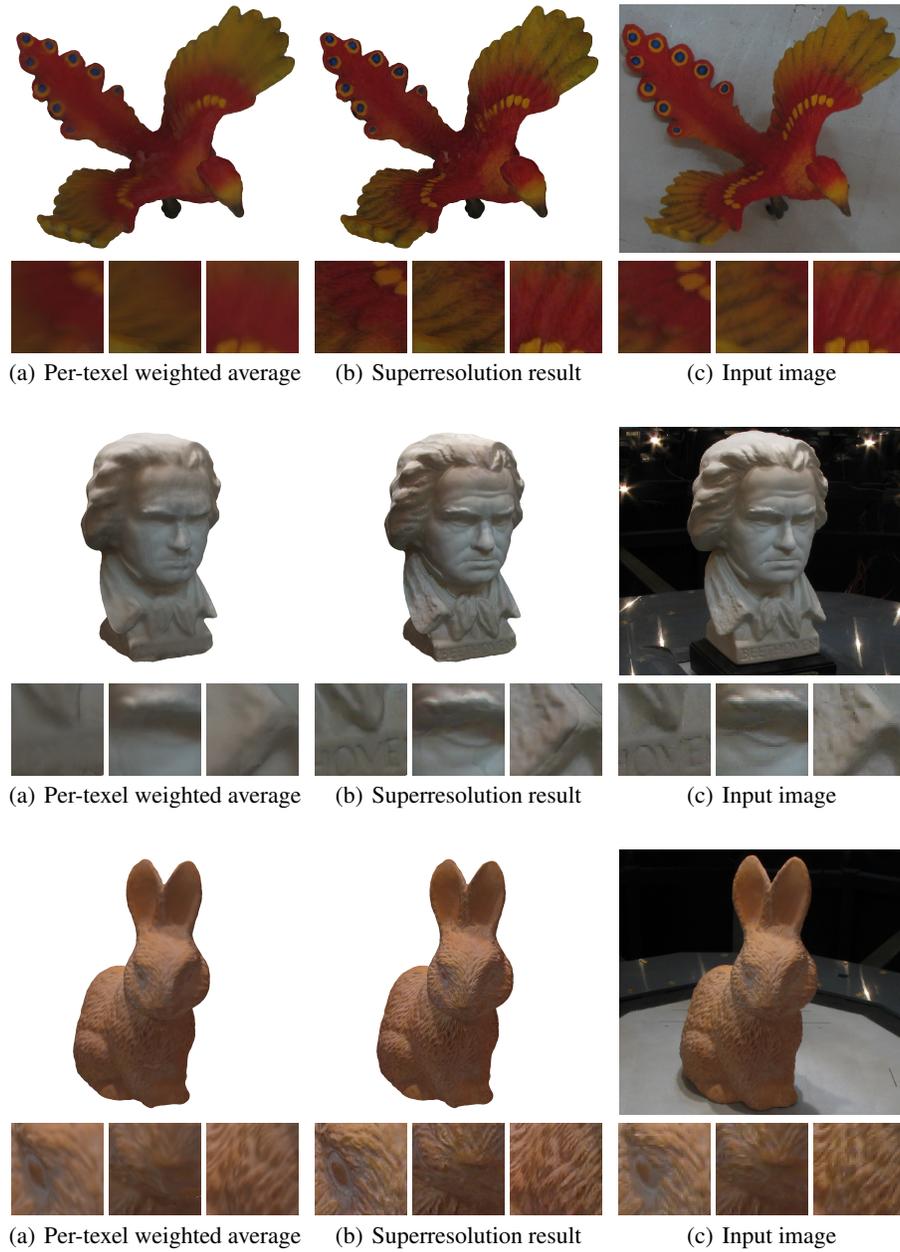


Fig. 7: Results from three real-world multiview datasets. The 3D model is rendered with the texture from texel-wise initialization (a) and the texture resulting from the proposed joint displacement map and superresolution algorithm (b). The result has more visible details than an input image taken from the same viewpoint (c). The rows below the large images show some close-ups. The reader is invited to zoom in on the electronic version to better appraise the differences.

References

1. Bernardini, F., Martin, I., Rushmeier, H.: High-quality texture reconstruction from multiple scans. *IEEE Transactions on Visualization and Computer Graphics* **7**(4) (2001) 318–332
2. Lensch, H., Heidrich, W., Seidel, H.P.: A silhouette-based algorithm for texture registration and stitching. *Graphical Models* **63**(4) (2001) 245–262
3. Theobalt, C., Ahmed, N., Lensch, H., Magnor, M., Seidel, H.P.: Seeing people in different light-joint shape, motion, and reflectance capture. *IEEE Transactions on Visualization and Computer Graphics* **13**(4) (2007) 663–674
4. Allne, C., Pons, J.P., Keriven, R.: Seamless image-based texture atlases using multi-band blending. In: *19th International Conference on Pattern Recognition*. (2008)
5. Lempitsky, V., Ivanov, D.: Seamless mosaicing of image-based texture maps. In: *Proceedings of CVPR*. Volume 1. (2007) 1–6
6. Fransens, R., Strecha, C., van Gool, L.: Optical flow based super-resolution: A probabilistic approach. *Computer Vision and Image Understanding* **106**(1) (2007) 106–115
7. Schoenemann, T., Cremers, D.: High resolution motion layer decomposition using dual-space graph cuts. In: *Proceedings of CVPR*. (2008) 1–7
8. Sroubek, F., Cristobal, G., Flusser, J.: A unified approach to superresolution and multichannel blind deconvolution. *IEEE Transactions on Image Processing* **16**(9) (2007) 2322–2332
9. Baker, S., Kanade, T.: Limits on super-resolution and how to break them. *PAMI* **24**(9) (2002) 1167–1183
10. Goldluecke, B., Cremers, D.: Superresolution texture maps for multiview reconstruction. In: *Proceedings of ICCV*. (2009) accepted
11. Lui, L.M., Wang, Y., Chan, T.F.: Solving PDEs on manifold using global conformal parameterization. In: *Variational, Geometric, and Level Set Methods in Computer Vision: Third International Workshop (VLSM)*. (2005) 309–319
12. Welk, M., Theis, D., Brox, T., Weickert, J.: PDE-based deconvolution with forward-backward diffusivities and diffusion tensors. In: *Scale-Space and PDE Methods in Computer Vision*. Volume 3459 of *Lecture Notes in Computer Science*. Springer (2005) 585–597
13. Chambolle, A.: An algorithm for total variation minimization and applications. *Mathematical Imaging and Vision* **20** (2004) 89–97
14. Lvy, B., Petitjean, S., Ray, N., Maillot, J.: Least squares conformal maps for automatic texture atlas generation. *ACM Transactions on graphics (SIGGRAPH)* **21**(3) (2003) 362–371
15. Wang, Y., Gu, X., Hayashi, K., Chan, T.F., Thompson, P., Yau, S.T.: Surface parameterization using Riemann surface structure. In: *Proceedings of ICCV*. Volume 2. (2005) 1061–1066
16. Weickert, J., Schar, H.: A scheme for coherence-enhancing diffusion filtering with optimized rotation invariance. *Journal of Visual Communication and Image Representation* **13**(1–2) (2002) 103–118
17. Donnelly, W.: *Per-Pixel Displacement Mapping with Distance Functions*. In: *GPU Gems 2*. Addison-Wesley Longman (2005)
18. Blomgren, P., Chan, T.F.: Color TV: Total variation methods for restoration of vector-valued images. *IEEE Trans. Image Processing* **7** (1998) 304–309
19. Weickert, J.: Coherence-enhancing diffusion of colour images. *Image and Vision Computing* **17**(3–4) (1999) 201–212
20. Kolev, K., Cremers, D.: Integration of multiview stereo and silhouettes via convex functionals on convex domains. In: *Proc. ECCV*. Volume 5302 of *Lecture Notes In Computer Science*. (2008) 752–765