SEAMLESS TEXTURE STITCHING ON A 3D MESH BY POISSON BLENDING IN PATCHES

Arnaud Dessein, William A. P. Smith, Richard C. Wilson, Edwin R. Hancock

Department of Computer Science, University of York, UK

ABSTRACT

In this paper, we propose a novel approach to seamless texture stitching on a 3D mesh. The main idea is to blend the sampled images by least-angle selection of gradients in overlapping patches. This is in contrast to previous works which focus on vertex- or face-based strategies with additional heuristics for robustness. Patches are obtained by growing a uniform mesh segmentation via geodesic projections. Blending is achieved by formulating a screened Poisson equation using discrete differential operators. The processing pipeline further includes an optional step of color transformation for calibration and correction. This is applied to a zippered mesh based on range scans and a morphable model fitted to face photographs.

***Index Terms*—** 3D model, texture reconstruction, seamless stitching, Poisson blending, uniform overlapping patches

1. INTRODUCTION

The problem of stitching several textures to form a seamless texture arises from various applications in 2D image and 3D mesh processing. In particular, this problem is inherently tied to the process of image-based object modeling to capture real-world objects. Standard techniques in this context use an RGB-D scanner or camera to collect multiple view-dependent observations consisting of color and depth maps, with technologies including time-of-flight or structured light for range images and binocular stereo for intensity images. It is relatively easy to recover the 3D shape of the object from these depth maps, using classical methods such as the iterative closest point algorithm [1] to align observations, and zippering [2] or volumetric merging [3] to integrate them on a base shape mesh. However, it is much more difficult to rebind color information to recover texture.

A related problem is shape-from-X, where the 3D shape is estimated from 2D images via cues such as shading or texture. Common methods include bundle adjustment [4] for image registration, and silhouette-based visual hull [5] or space carving [6] for shape recovery. Another example is when a statistical model for a specific object class, such as 3D morphable models for human faces [7], is fitted to one or more 2D images. In these scenarios, the recovered shape must be textured from the available observations of the different scene points. Although established pipelines exist for reconstructing shape, texture stitching is, surprisingly, much less studied.

In this paper, we assume that geometry has already been processed and focus on texture. There is in general no exact correspondence between points on the images and on the recovered surface. Thus, one needs both to interpolate color and to integrate concurrent data from several images. For instance, a baseline method is to sample the images on the mesh by projection and color interpolation, and then merge the textures from overlapping images by averaging. However, such a back-projection of color is likely to contain seams. The main issues are that (i) the reconstructed geometry is only approximate, (ii) the views cannot be registered perfectly, (iii) the acquisition settings may vary across views, and (iv) the lighting may change between poses. Hence, texture reconstruction for 3D models is still a challenging problem. Several approaches have been developed to cope with these issues and obtain a seamless stitching.

This work was supported by grant DSTLX1000070369 from the Defence Science & Technology Laboratory.

1.1. Related work

Early works have focused on different weighting heuristics to average overlapping textures. Pulli et al. [8] couple viewing angle and proximity to the projected outline. Neugebauer and Klein [9] additionally consider texture deviation from a normal model. Matsushita and Kaneko [10] or Johnson and Kang [11] exploit projected face areas. Levoy et al. [12] merge several cues: projected area, lighting angle, proximity to the projected outline, to the image border, and to the silhouette edge seen from the light source or the mirror direction, in a confidence index which is then smoothed conservatively among adjacent vertices. Bernardini et al. [13] also combine different factors: viewing angle, distance to an edge, deviation from a normal model and resolution. Wang et al. [14] rather formulate an optimal weighting based on sampling theory. Similarly, Baumberg [15] proposes a multi-band approach to blend low and high frequencies over long and short ranges, respectively, where weights are smoothed ratios of projected area over actual area. More recently, Totz et al. [16] propose a multi-band weighting based on the viewing distance and angle, where textures are averaged after flattening based on disk-homeomorphic patches that overlap for increased robustness.

In the meantime, some alternatives to weighting schemes have also been devised. Rocchini et al. [17] select textures by least angle, smooth out selection along adjacent vertices to obtain larger contiguous regions that map to the same image, and only average border faces using barycentric coordinates as weights after a local linear registration. Lensch et al. [18] refine this approach and introduce additional heuristics to discard points based on the viewing angle, depth variation and proximity to the projected outline.

A common drawback of all these approaches is that they do not correct for high color discrepancies due to different acquisition settings or light variations across views. Moreover, ghosting and blurring artifacts appear as soon as the textures are misaligned: if the transition width of the weighting function is too large, then a slight misalignment produces a double-exposure effect where some features appear twice, whereas if it is too small, then texture merging is made within a few pixels only which is prone to seams.

To tackle the former issue, Beauchesne and Roy [19] propose a preprocessing step of automatic relighting, assuming almost convex objects, where differences in illumination between two images are corrected based on a lighting ratio. Agathos and Fisher [20] account for lighting and camera calibration by applying both a global linear correction estimated from overlapping views, and a local correction to smooth out remaining seams based on iterative averaging along

the mesh connectivity. Bannai et al. [21] extend this to remove propagation errors by accounting for a groupwise transform rather than pairwise transforms between images, where a single joint problem is solved instead of several binary problems. This approach is yet computationally demanding, requiring several hours to run. Moreover, these approaches do not deal with issues of misalignment.

Alternatively, Lempitsky and Ivanov [22] apply a global optimization based on a Markov random field to favor smoothness in texture assignment and penalize sharp seams, followed by a seam-leveling scheme similar to gradient-domain stitching. Gal et al. [23] extend this by also optimizing with respect to local translations to overcome misalignment problems. Following the success of gradient-domain stitching methods, Chuang et al. [24] formulate a robust stitching scheme via a screened Poisson equation, although the approach lacks a local alignment scheme. The interest of such methods is that the texture reconstruction error is optimized in the gradient domain, leading to minimized perceptual errors in respective frequency bands and accounting for large color offsets between views, without requiring any weighting heuristics. Moreover, they include texturing of unobserved points as part of the process, hence removing the need for postprocessing heuristics toward hole filling.

1.2. Contributions

In this work, we develop a novel patch-based approach for texture stitching on a 3D mesh. Our methods are fully intrinsic to the mesh surface, generic in that we make no restrictive assumptions on the mesh topology (e.g., watertight, closed) or geometry (e.g., convex, smooth), and only require a couple of minutes to run. After image sampling, we perform gradient-based stitching within a uniform mesh segmentation. Patch textures are first selected globally by least angle, and gradient textures are then retained locally in patch overlaps. We use an original variant on Poisson blending by coupling discrete differential operators with screening. We also consider an extension to handle color transformation via calibration and correction. Results are presented on both a zippered mesh based on range scans and a morphable model fitted to face photographs.

2. PROPOSED METHODS

In this section, we expose our methods for seamless texture stitching. The pipeline includes core steps of image sampling, mesh segmentation and gradient stitching, with an optional color transformation.

2.1. Image sampling

For each view, we determine the set of visible vertices on the 3D model. Occlusions can be tested by ray casting from the viewer, although we here use an approximate but much less demanding depth criterion with a z-buffer. The image is then sampled on the mesh by back-projection of textures for visible vertices after bilinear color interpolation within the pixel grid. Additionally, the viewing angles for each face and vertex are computed as part of the process and stored for later use. We notice finally that any of the heuristics discussed in the introduction could be employed here to discard some vertices that are potentially corrupted by considering them as occluded.

2.2. Mesh segmentation

We segment the mesh with a farthest-point strategy based on fast marching [25], enhanced with an original patch growing step to form an overlapping structure. This produces a uniform segmentation compared to the region-growing scheme in [16]. Moreover, we do not require patches to be disk-homeomorphic since our methods are intrinsic to the mesh surface. Not only does it eliminate undesirable distortions inherent to flattening, but it also allows to consider various mesh topologies with arbitrary genus and number of boundary components. Lastly, instead of using extrinsic 3D ball radii as thresholds for patch dilation, overlaps are obtained by growing patches intrinsically within neighbors via geodesic projections.

We consider a triangular base shape mesh $\mathcal{M}$ and assume it describes a 2D manifold with geodesic distance map D. The connectivity is given by a simplicial complex $\mathcal{K}$ whose elements are vertices $\{i\}$, edges $\{i,j\}$ or faces $\{i,j,k\}$, with indices $i,j,k \in [1\,..\,N]$, where N is the number of vertices. We write a vertex $\{i\}$ as i for simplicity. We first select vertices iteratively by adding a new sample one at a time. Denoting by $D_l(i)$ the geodesic distance map to the first l selected samples, we select sample $i^\star_{l+1}$ as the vertex that maximizes $D_l(i)$. The distance map $D_{l+1}(i)$ can simply be updated as the minimum between $D_l(i)$ and $D(i,i^\star_{l+1})$. We continue this process until a desired number M of vertices have been sampled.

Patches $\mathcal{P}_1,\dots,\mathcal{P}_M$ are then obtained via the geodesic Voronoi tessellation based on the samples. The segmentation thus defines a dual graph $\mathcal{G}=(\mathcal{V},\mathcal{E})$, where $\mathcal{V}=[1\,..\,M]$, and $(m,n)\in\mathcal{E}$ if $\mathcal{P}_m$ and $\mathcal{P}_n$ are neighbors, i.e., are connected by an edge $\{i,j\}\in\mathcal{K}$. To grow a patch $\mathcal{P}_m$, we consider separately each of its neighbor patches $\mathcal{P}_n$ with $(m,n)\in\mathcal{E}$, and define thresholds d_{mn} as follows:

$$d_{mn} = \sigma \times D(i^\star_m, i^\star_n) \ , \tag{1}$$

where $\sigma \geq 0$ is set by the user and can be seen as an overlap ratio or factor, and the geodesic distance D is restricted to the union $\mathcal{P}_m \cup \mathcal{P}_n$ of the reference patch and considered neighbor. The overlap $\mathcal{O}_{mn}$ of $\mathcal{P}_m$ onto $\mathcal{P}_n$ is then constructed by geodesic projections:

$$\mathcal{O}_{mn} = \left\{ i\in\mathcal{P}_n : \min_{j\in\mathcal{P}_m} D(i,j) \leq d_{mn} \right\} \ . \tag{2}$$

A given grown patch $\mathcal{Q}_m$ is eventually constructed by concatenation of the reference patch $\mathcal{P}_m$ with the respective overlaps:

$$\mathcal{Q}_m = \mathcal{P}_m \cup \bigcup_{n|(m,n)\in\mathcal{E}} \mathcal{O}_{mn} \ . \tag{3}$$

The time complexity of the whole process is in $O(N \log N \log M)$.

2.3. Gradient stitching

We stitch textures in the gradient domain by modifying a principled approach to Poisson's equation on a mesh with Dirichlet boundary conditions [26] so as to incorporate a screening term instead. This allows both to consider meshes with no boundary without having to arbitrarily define a virtual one, and to relax hard constraints on texture values at boundary vertices via a soft regularization distributed over all vertices. Since we rely on discrete differential operators on the mesh surface, our approach completely preserves conservative vector fields compared to extrinsic 3D finite elements in [24]. This makes our approach more natural from a theoretical perspective, even if non-conservative fields are rather formed in practice.

A discrete vector field V is a piecewise constant vector function defined for each triangle T_l by a coplanar vector v_l. A discrete potential field is a piecewise linear function $\phi(s) = \sum_{i\in\mathcal{K}} \phi_i B_i(s)$ on the mesh surface, where B_i is the piecewise linear basis function valued 1 at vertex i and 0 at other vertices, and ϕ_i specifies the value of ϕ at vertex i. The discrete gradient of ϕ for triangle T_l is

$\nabla\phi_l = \sum_{i\in\mathcal{K}} \phi_i \nabla B_{il}$, where ∇B_{il} is the gradient of B_i within T_l. The divergence of V at vertex i is $\operatorname{div} V(i) = \sum_{T_l\in\mathcal{K}_i} |T_l|\, \nabla B_{il}^\top\, \mathbf{v}_l$, where $\mathcal{K}_i$ is the set of triangles sharing vertex i and $|T_l|$ is the area of triangle T_l. Writing Poisson's equation $\operatorname{div}\nabla\phi = \operatorname{div} V$ in this framework leads to a linear system of equations $\mathbf{A}\mathbf{x} = \mathbf{y}$ for the unknown potential values $x_i = \phi_i$, where:

$$a_{ij} = \sum_{T_l\in\mathcal{K}_i} |T_l|\, \nabla B_{il}^\top \nabla B_{jl} \ , \quad y_i = \sum_{T_l\in\mathcal{K}_i} |T_l|\, \nabla B_{il}^\top\, \mathrm{v}_l \ . \quad (4)$$

This system is sparse since the sum for coefficients a_{ij} is non-null iff $\{i,j\} \in \mathcal{K}$ (it is an edge). The sum is then simply over the triangles T_l (two if not a boundary edge, one otherwise) sharing this edge.

This equation can be interpreted as seeking for a potential field ϕ whose gradient $\nabla\phi$ matches the guide vector field V. If V is conservative, i.e., it is the gradient of an existing potential field ϕ, then ϕ is the exact solution. Otherwise, a more general minimizer can still be obtained by least squares but its gradient differs from V. In addition, we regularize the minimization via screening:

$$\min_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2 + \lambda \left\|\mathbf{x} - \mathbf{x}'\right\|_2^2 \ , \quad (5)$$

where $\lambda > 0$ and $\mathbf{x}'$ defines a guide potential field ϕ' as $\phi'_i = x'_i$.

We apply this to solve for texture by considering each color channel independently as a potential field ϕ. For each view v, we compute the mean viewing angle of vertices in the different patches. Unobserved vertices, due either to occlusion or missing information, are assumed to have a viewing angle of $\pi/2$. Hence, patches with unobserved data are penalized and no difference on the nature of non-observability is made. For each patch now, we select texture from the view where the patch has the smallest viewing angle. For unobserved vertices, we also select texture from subsequent sorted views. We end up with partial textures $\phi^{(v)}$ that we stitch in overlaps by Poisson blending. To build up the guide vector field V, we select local texture gradients by least angle for each triangle T_l:

$$\mathrm{v}_l = \sum_{i\in\mathcal{K}} \phi_i^{(v_l)}\, \nabla B_{il} \ , \quad (6)$$

where v_l is the view whose angle is minimal for triangle T_l. We also fill in unobserved faces simply by setting their gradients to zero for smoothness. Screening is done via a rough estimate ϕ' obtained by averaging textures $\phi^{(v)}$, unobserved textures being discarded from the regularization. We use a small penalty $\lambda = 10^{-6}$ to remove color offset indeterminacies since we did not observe dramatic color bleeding issues compared to [24]. The time complexity for building up the linear system is in $O(M+N)$. A naive complexity for least squares optimization via Cholesky decomposition is in $O(N^3)$, though efficient solvers that exploit sparsity can be used instead.

2.4. Color transformation

To reduce texture discrepancies between views, we introduce two optional steps of color transformation via calibration and correction. Compared to [21], we consider pairwise interactions for computational efficiency, and assume that more subtle effects due to propagation errors are fixed by gradient-domain stitching. However, we allow for an affine instead of linear transform, which further accounts for differences in color offset and ambient light intensity, rather than just color gain and directional light intensity.

On the one hand, calibration is done if color charts are available. We first average pixels for each checking color to produce color maps under the different views. We then choose the color map of an arbitrary view or a virtual one (e.g., by averaging color maps) as reference, and solve for multiple pairwise affine transforms from color maps to this target by least squares. We finally apply these respective transforms to the sampled textures. On the other hand, correction is estimated directly from the sampled textures. Here, a robust regression method (e.g., least absolute deviations, least trimmed squares) is necessary to cope with potential outliers. We choose a reference view and compute pairwise affine transforms where views overlap. This not only compensates for different lighting conditions, but also for acquisition settings if views have not been calibrated before.

3. EXPERIMENTAL RESULTS

In this section, we present experimental results to evaluate our methods on real-world datasets. We consider both a zippered mesh based on range scans and a morphable model fitted to face photographs.

3.1. Range scans

We compare our methods on range scans with the only available tool we have found: the state-of-the-art texture stitcher [24]. We use their Rooster dataset, which consists of color information and a base shape mesh reconstructed from 8 depth maps by zippering (Fig. 1). The mesh features 68,612 vertices, a positive genus due to 1 visible handle, and 1 boundary component as a result of a small hole in the top. For a fair comparison, we do not apply color transformation since their tool does not include such a step. To assess the effect of the number of patches and benefits of using overlap, we try different values of M, σ. For illustration, we also compare to two baseline vertex-based strategies: averaging and least-angle selection.

On the one hand, these baseline strategies suffer from texture discrepancies across scans. Averaging textures produces seams at boundaries where some scans become or cease to be observable, and a general color offset resulting in a darker texture due to accounting for shadowed vertices (Fig. 2a). Selecting the less foreshortened vertices addresses these issues, but introduces more marked seams at transitions between selected scans (Fig. 2b). Moreover, inherent to these two approaches, many unobserved vertices are not textured.

On the other hand, our proposed approach tackles these issues at once and leads to much smoother color transitions. For a small number of patches ($M = 50, \sigma = 0.0$), however, we observe a blurring effect around the eye (Fig. 2c). Increasing the number of patches ($M = 100, \sigma = 0.0$), we are able to remove this artifact, yet we still observe slight seams (Fig. 2d). The best result is when further growing the patches so as to overlap ($M = 100, \sigma = 0.5$), where the obtained stitching is seamless (Fig. 2e). Looking at the frontal view, our result preserves sharp edges of specularities, details of the painted pattern, and color differences between the pattern and body (Fig. 2f). In contrast, the texture stitcher [24] blurs the specularities with the painted pattern, misses in particular the central specularity, and features a noticeable color bleed from the base so that details in the bottom half of the painted pattern are lost (Fig. 2g). The improvement is confirmed by computing the root mean square errors between scans and rendered models, with an average of 0.099 across views for our methods against 0.109 for the texture stitcher [24].

3.2. Face photographs

We now consider photographs from the CMU PIE face database [28]. We choose 3 views of a given subject under the same illumination (Fig. 3). We fit a state-of-the-art 3D morphable model [27] to estimate shape, texture, camera and lighting parameters for the 3

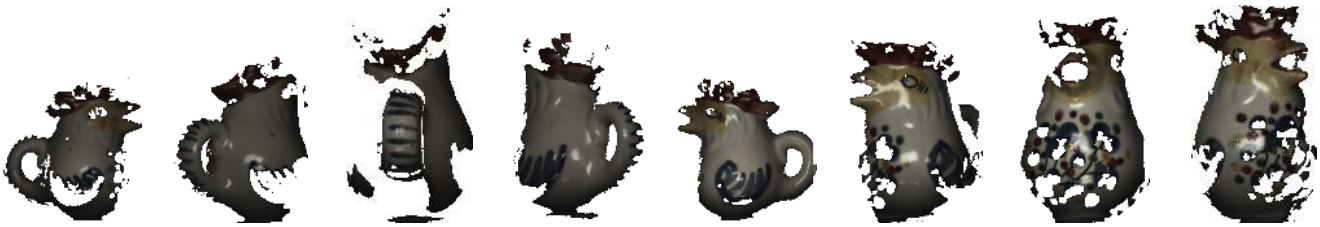
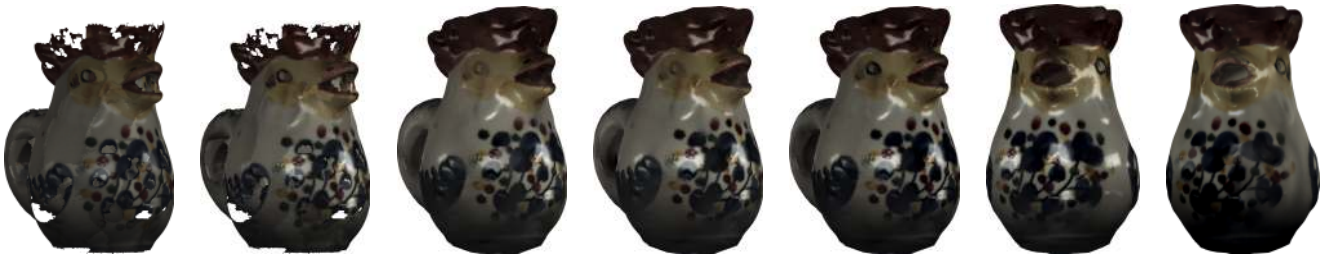


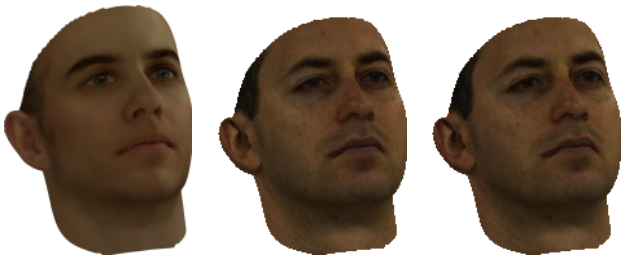
Fig. 1. Scans of the Rooster dataset. Colors from the 8 depth images are back-projected to the base shape mesh. The different views reveal substantial amounts of unavailable information and variations in illumination, hence seamless stitching for texture reconstruction is nontrivial.



(a) Average. (b) Least angle. (c) $M=50, \sigma=0$. (d) $M=100, \sigma=0$. (e) $M=100, \sigma=\frac{1}{2}$. (f) Idem. (g) Stitcher [24].

Fig. 2. Texture reconstruction of the Rooster dataset. We compare our patch-based approach to two baseline and one state-of-the-art vertex-based approaches. Merging vertex information is not sufficient on this difficult dataset, whereas patch integration improves quality by removing seams while preserving details, as soon as we blend the solutions with enough patches and overlaps for robustness and smoothness.

Fig. 3. Photographs from the CMU PIE face database. The 3 poses of a given subject are taken simultaneously with different cameras. Subtle texture discrepancies appear across views because of different acquisition settings, which might lead to artifacts such as seams or different eye colors if textures are not stitched properly.



(a) Fitted model [27]. (b) Core steps. (c) Color step.

Fig. 4. Texture reconstruction of face photographs on the morphable model. Rendering the texture from the fitted model does not provide a realistic appearance, whereas our result features plausible local asperities. However, care should be taken to compensate for texture discrepancies via color transformation and avoid related artifacts.

poses. For better accuracy in shape alignment, we keep the 3 fittings separately during image sampling, and average shapes to obtain a base shape mesh afterwards only. The model has 53,490 vertices, 1 boundary component as a cut in the back of the face, a null genus. We fix the segmentation parameters ($M=100, \sigma=0.5$), and demonstrate the effects of color transformation via calibration only, since lighting is consistent across views and color charts are available.

The results show that rendering a model fit [27] provides an inaccurate estimate of the actual face appearance and misses local asperities because of statistical regularization and smoothing (Fig. 4a). In contrast, our core stitching pipeline succeeds in improving texture details, although subtle color discrepancies still remain in that the right side of the face is lighter than the left one (Fig. 4b). The optional step of color transformation allows to tackle this latter problem so that the colors of both sides are more consistent (Fig. 4c). The results are again corroborated by the root mean square errors, which average to 0.0710 for the morphable model [27], against 0.0577 for our core pipeline, and 0.0570 with the additional color calibration.

4. CONCLUSION

We presented a novel technique for seamless texture stitching on a 3D model by Poisson blending in overlapping patches on the mesh surface. The obtained results demonstrate the relevance of our approach. Several perspectives were however left out for future work.

A drawback of our affine color transform is to ignore nonlinearities due to specularities and shadows, so that an enhanced lighting model could improve this step. In addition, we would like to evaluate color correction in more difficult conditions where calibration charts are not available and lighting varies between views. It would also be worth integrating a shape correction step to improve robustness against dramatic misalignment issues. Lastly, a more ambitious line is to couple shape and texture reconstruction in a unified paradigm.

5. REFERENCES

[1] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 239–256, Feb. 1992.

[2] G. Turk and M. Levoy, "Zippered polygon meshes from range images," in *Proc. SIGGRAPH*, Orlando, Florida, US, Jul. 1994, pp. 311–318.

[3] B. Curless and M. Levoy, "A volumetric method for building complex models from range images," in *Proc. SIGGRAPH*, New Orleans, LA, USA, Aug. 1996, pp. 303–312.

[4] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Bundle adjustment — a modern synthesis," in *Vision Algorithms: Theory and Practice*, ser. Lecture Notes in Computer Science. Berlin Heidelberg: Springer, 2000, vol. 1883, pp. 298–372.

[5] A. Laurentini, "The visual hull concept for silhouette-based image understanding," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 2, pp. 150–162, Feb. 1994.

[6] K. N. Kutulakos and S. M. Seitz, "A theory of shape by space carving," *Int. J. Comput. Vision*, vol. 38, no. 3, pp. 199–218, Jul. 2000.

[7] V. Blanz and T. Vetter, "A morphable model for the synthesis of 3D faces," in *Proc. SIGGRAPH*, Los Angeles, CA, USA, Aug. 1999, pp. 187–194.

[8] K. Pulli, H. Abi-Rached, T. Duchamp, L. G. Shapiro, and W. Stuetzle, "Acquisition and visualization of colored 3D objects," in *Int. Conf. on Pattern Recognition*, vol. 1, Brisbane, Australia, Aug. 1998, pp. 11–15.

[9] P. J. Neugebauer and K. Klein, "Texturing 3D models of real world objects from multiple unregistered photographic views," *Comput. Graph. Forum*, vol. 18, no. 3, pp. 245–256, Sep. 1999.

[10] K. Matsushita and T. Kaneko, "Efficient and handy texture mapping on 3D surfaces," *Comput. Graph. Forum*, vol. 18, no. 3, pp. 349–358, Sep. 1999.

[11] A. E. Johnson and S. B. Kang, "Registration and integration of textured 3D data," *Image and Vision Computing*, vol. 17, no. 2, pp. 135–147, Feb. 1999.

[12] M. Levoy, K. Pulli, B. Curless, S. Rusinkiewicz, D. Koller, L. Pereira, M. Ginzton, S. Anderson, J. Davis, J. Ginsberg, J. Shade, and D. Fulk, "The digital Michelangelo project: 3D scanning of large statues," in *Proc. SIGGRAPH*, New Orleans, LA, USA, Jul. 2000, pp. 131–144.

[13] F. Bernardini, I. M. Martin, and H. Rushmeier, "High-quality texture reconstruction from multiple scans," *IEEE Trans. Vis. Comput. Gr.*, vol. 7, no. 4, pp. 318–332, Oct./Dec. 2001.

[14] L. Wang, S. B. Kang, R. Szeliski, and H.-Y. Shum, "Optimal texture map reconstruction from multiple views," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, Kauai, HI, USA, Dec. 2001, pp. 347–354.

[15] A. Baumberg, "Blending images for texturing 3D models," in *British Machine Vision Conf. (BMVC)*, Cardiff, UK, Sep. 2002, pp. 404–413.

[16] J. Totz, A. J. Chung, and G.-Z. Yang, "Patient-specific texture blending on surfaces of arbitrary topology," in *Workshop on Augmented environments for Medical Imaging and Computer-aided Surgery (AMI-ARCS)*, London, UK, Sep. 2009, pp. 78–85.

[17] C. Rocchini, P. Cignoni, C. Montani, and R. Scopigno, "Multiple textures stitching and blending on 3D objects," in *Eurographics Workshop on Rendering (EGWR)*, Granada, Spain, Jun. 1999, pp. 119–130.

[18] H. P. A. Lensch, W. Heidrich, and H.-P. Seidel, "A silhouette-based algorithm for texture registration and stitching," *Graph. Models*, vol. 63, no. 4, pp. 245–262, Jul. 2001.

[19] E. Beauchesne and S. Roy, "Automatic relighting of overlapping textures of a 3D model," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Madison, WI, USA, Jun. 2003, pp. 139–146.

[20] A. Agathos and R. B. Fisher, "Colour texture fusion of multiple range images," in *IEEE 3-D Digital Imaging and Modeling (3DIM)*, Banff, Canada, Oct. 2003, pp. 139–146.

[21] N. Bannai, R. B. Fisher, and A. Agathos, "Multiple color texture map fusion for 3D models," *Pattern Recogn. Lett.*, vol. 28, no. 6, pp. 748–758, Apr. 2007.

[22] V. Lempitsky and D. Ivanov, "Seamless mosaicing of image-based texture maps," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Minneapolis, MN, USA, Jun. 2007, pp. 1–6.

[23] R. Gal, Y. Wexler, E. Ofek, H. Hoppe, and D. Cohen-Or, "Seamless montage for texturing models," *Comput. Graph. Forum*, vol. 29, no. 2, pp. 479–486, May 2010.

[24] M. Chuang, L. Luo, B. J. Brown, S. Rusinkiewicz, and M. Kazhdan, "Estimating the Laplace-Beltrami operator by restricting 3D functions," *Comput. Graph. Forum*, vol. 28, no. 5, pp. 1475–1484, Jul. 2009.

[25] G. Peyré and L. D. Cohen, "Geodesic remeshing using front propagation," *Int. J. Comput. Vis.*, vol. 69, no. 1, pp. 145–156, Aug. 2006.

[26] Y. Yu, K. Zhou, D. Xu, X. Shi, H. Bao, B. Guo, and H.-Y. Shum, "Mesh editing with Poisson-based gradient field manipulation," *ACM Trans. Graph. - Proc. of ACM SIGGRAPH*, vol. 23, no. 3, pp. 644–651, Aug. 2004.

[27] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter, "A 3D face model for pose and illumination invariant face recognition," in *IEEE Int. Conf. on Advanced Video and Signal Based Surveillance (AVSS)*, Genova, Italy, Sep. 2009, pp. 296–301.

[28] T. Sim, S. Baker, and M. Bsat, "The CMU Pose, Illumination, and Expression (PIE) database," in *IEEE Int. Conf. on Automatic Face and Gesture Recognition (FGR)*, Washington, DC, USA, May 2002, pp. 46–51.