3D Reconstruction for damaged documents: imaging of The Great Parchment Book

Kazim Pal
Dept of Computer Science
University College London

Melissa Terras
Dept of Information Studies
University College London

Tim Weyrich
Dept of Computer Science
University College London

ABSTRACT

Digitization of historical documents is extremely useful as it allows easy access to the documents from remote locations and removes the need for potentially harmful physical handling. Traditional imaging methods are unsuitable for documents with complex geometry as they will produce images containing perspective distortions, and 3D imaging methods previously proposed for document scanning will often suffer from occlusions and/or require manual alignment of individual range scans. We present a lightweight pipeline for imaging and generating 3D reconstructions of severely damaged and distorted documents which exhibit such complex geometry. We demonstrate our pipeline on The Great Parchment Book of The Honourable The Irish Society, a 17th century survey of the Ulster estates managed by the City of London, which was severely damaged by a fire in 1786.

Categories and Subject Descriptors

I.3.3 [**Computer Graphics**]: Picture/Image Generation—*Digitizing and scanning*; I.7.5 [**Document and Text Processing**]: Document capture—*Scanning*

General Terms

Document digitisation, scanning, 3D reconstruction

1. INTRODUCTION

In this paper we present a pipeline for imaging and generating 3D reconstructions of highly distorted historical documents such as those shown in Figure 1.

This method is particularly relevant to archives and museums who possess such valuable historical documents whose contents are inaccessible and which cannot be fully restored by conventional means, and are difficult to read in person due to high levels of damage and the fragile nature of the material. Such damaged documents are surprisingly common in archives across the world. Due to the fragile state of the folios, access is restricted to a select few persons and all handling must be done by a trained conservator. Digital representations of such documents help to overcome this problem by allowing people to read their contents at any time, from remote locations, and without necessitating harmful physical handling of the document.

HIP '13 August 24 2013, Washington, DC, USA

http://dx.doi.org/10.1145/2501115.2501125.

Figure 1: Folios with strong geometric distortions caused by fire damage. (Reproduced with the permission of The Honourable The Irish Society and the City of London Corporation, London Metropolitan Archives)

Digital representations can also allow for types of virtual conservation, which is not possible on the physical folios. These could include removal of geometric distortions, correction of discolouration, and contrast enhancement on faded text, all of which would risk irreparable damage to the documents if attempted on the physical documents themselves.

We apply our pipeline to an early modern document called the Great Parchment Book of The Honourable The Irish Society, a property survey of the Ulster estates managed by the City of London commissioned by Charles I in the 17th century whose contents are of great interest to historians studying the history of this region. The book fell victim to a fire in 1786, causing such strong distortions which conservators tell us cannot be restored by traditional conservation methods.

To demonstrate the high quality of our reconstructions, we also introduce a novel quality metric to gauge the effective digitisation resolution of our reconstruction according to

archival standards.

2. RELATED WORK

A number of methods have been proposed for imaging and reconstructing the 3D shape of a document. These methods can be broadly grouped into three different classes: single-image methods, stereo-image methods, and structured-light scanning methods.

Single-image methods are typically designed to reconstruct the shape of modern printed documents with the aim of rectifying their text to improve the performance of OCR algorithms. Wada et al. [23] and Zhang et al. [27] propose methods to reconstruct the surface of bound documents scanned with a flat-bed scanner by using the shading cues in the image. Tian & Narasimhan [20] reconstruct the shape of smoothly folded pages by detecting horizontal line directions and vertical strokes in the text, fitting a 2D text grid to the image, and then estimating the 3D position of each grid cell. Such methods are not applicable to damaged historical documents since we cannot make such strong assumptions about shading and textual cues.

Methods such as those proposed by Ulges [21], Lampert et al. [14], and Koo et al. [12] capture stereo images of a document from which a 3D surface can be reconstructed. These methods are demonstrated on open books, and assume there is no self-occlusion in the pages.

Structured-light scanners have also been used by Brown & Seales [3], Brown et al. [4], and Sun et al. [19], and Bianco et al. [2] to capture a document's shape. Here the document surface is illuminated with a known pattern, and the distortion of the pattern as seen from a camera is used to compute the 3D shape. The approach makes no assumptions about the content of the document, but makes the same assumptions as the stereo-image approach regarding the absence of self-occlusions in the document. If there are any regions of the page which cannot be seen by both the camera and the projector, they will not be reconstructed.

As an alternative method, the cultural-heritage digitisation community increasingly makes use of multi-view stereo techniques, which compute a 3D representation of an object or scene from an unordered collection of images taken from arbitrary camera positions. Some multi-view stereo algorithms require the camera calibration parameters as input, while others estimate these parameters as part of the reconstruction process. The types of 3D representations produced include point clouds [18, 25, 8], volumetric models [13, 7], and triangle meshes [22, 1]. ARC 3D [22] and Autodesk 123D Catch [1] are both free end-to-end web services which compute textured 3D models from an uncalibrated set of images, and are popular in cultural-heritage digitisation.

3. OUR APPROACH

For our acquisition problem, we adopt multi-view stereo. The approach is very well suited for practical, manual acquisition of the deformed parchments, as it allows a user to freely choose viewpoints to reach all parts of the wrinkled surface, while using commodity hardware only.

Figure 2: Acquisition setting.

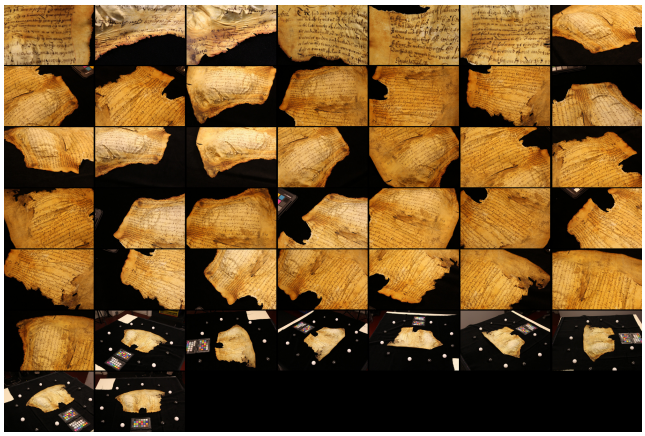

Figure 3: An input image set for a single parchment page.

3.1 Imaging

We image the parchments using a hand-held DSLR camera (Canon 5D Mark III). Each parchment is placed on a black velvet cloth and illuminated with three large, evenly spaced diffuse lights to provide uniform illumination (Figure 2). A calibration target is also placed on the table next to the parchment, which can be used to measure scale and to calibrate colours. For each page, we first take a set of images (typically between eight and ten) in a circular formation such that the entire parchment is visible in each image. We then take many more close-up images, making sure to cover the entire surface of the page thoroughly. For highly distorted areas of the parchment where the text has shrunk to a very small size, we use a macro lens to obtain extreme close-up images of the text. Typically, a single image set will contain between forty and sixty images. Figure 3 shows an example of a complete image set for a single page.

Previous approaches such as top-down cameras or structured-light scanners would not be able to cope with the self-occlusions in the pages and would produce incomplete reconstructions. Using a hand-held camera, however, allows us to adapt the imaging process to the highly varying shapes of the parchment, and guarantee full coverage of the parchment surface.

Archives and museums are also unlikely to have access to such specialised scanning equipment or the expertise required to use it. Since our image method is simple and requires only a digital camera, it is far more accessible to such institutions.

3.2 Reconstruction

3.2.1 Structure from Motion

We process the image sets with Wu et al.'s VisualSFM [25] software which uses Wu et al.'s GPU implementation of SIFT [15, 24] and their multi-core bundle adjustment algorithm [26] to generate a sparse 3D reconstruction using structure from motion, along with camera calibration parameters for each image. These parameters are a focal length f, a 3×3 camera rotation matrix R, and a 3-vector camera translation $\mathbf{t}$.

We then apply Furukawa and Ponce's PMVS algorithm [8] to generate a dense point reconstruction, examples of which are shown in Figure 6. The PMVS algorithm is considered state-of-the-art in the area of dense reconstruction, and both it and VisualSFM's structure from motion procedure perform very well on even highly unstructured image sets containing variations in lighting, image exposure, lens type, etc. Source code and binaries for these algorithms are available from the authors' websites.

The reconstruction is performed up to an arbitrary scale, so the distances in the resulting object space do not correspond to the true distances in real-life space. To correct for this we allow a user to mark points on the colour checker which are a known distance apart in reality and then triangulate their positions in object space to compute a scaling factor which can be used to scale distances in object space to match those in real-life space.

3.2.2 Surface Reconstruction

The next step in our pipeline is to compute a triangle mesh from the dense point cloud, for which we use Kazhdan et al.'s Poisson Surface Reconstruction algorithm [11], which fits a surface to a set of oriented points. It forumulates the surface reconstruction as a single Poisson problem which is solved using a multiscale approach. This algorithm requires very little parameter-tuning (we use the exact same parameters for every reconstruction), is resilient to noisy data, is able to interpolate holes in the point reconstruction very well, and since it makes use of the normals that are generated by PMVS, it is a natural choice for fitting a surface to the points. We use the Poisson Surface Reconstruction implementation provided in MeshLab [16]. Examples of reconstructed meshes are shown in Figure 7.

3.2.3 Texture Mapping

The final part of our reconstruction pipeline is to generate texture maps for the triangle meshes. The problem of computing a well-structured texture-coordinate parametrization for an arbitrarily shaped triangle mesh is still an active area of research, and is a difficult problem especially for meshes generated by 3D reconstruction algorithms which often contain bad geometry. For the sake of simplicity and to circumvent these difficulties we use the texture atlas method by Esteban & Schmitt [10], originally proposed by Schmitt and Yemez [17].

Each triangle in the mesh is mapped to its own right-angled triangle of edge length N in the texture atlas. For each vertex $\mathbf{p}$ in the mesh, and for each input camera $\mathcal{C}$, we assign a weight $\tilde{w}_{\mathbf{p}}^{\mathcal{C}}$ according to how well $\mathcal{C}$ sees $\mathbf{p}$. The weighting scheme is adapted from Buehler et al.'s Unstructured Lumigraph interpolation [5] scheme, and takes into account visibility, relative angle, and resolution of the vertex in each camera.

For a vertex $\mathbf{p}$ with normal vector $\mathbf{n}$, and a camera C_i in the set of observing cameras $\mathcal{C}$, with centre of projection $\mathbf{o}_i$ and viewing ray $\mathbf{v}_{\text{view}}$ we define the penalty terms as:

$$
\begin{aligned}
E_{\text{ang}} &= \arccos(\mathbf{n} \cdot \mathbf{v}_{\text{view}}) \\
E_{\text{res}} &= \|\mathbf{p} - \mathbf{o}_i\| \\
E_{\text{vis}} &= \begin{cases} 0, & \text{if } \mathbf{p} \text{ is visible in } C_i \\ \infty, & \text{otherwise} \end{cases} \\
E_{\text{total}} &= E_{\text{vis}} + w_{\text{ang}} E_{\text{ang}} + w_{\text{res}} E_{\text{res}}
\end{aligned}
$$

In our system we choose $w_{\text{ang}} = 5$ and $w_{\text{res}} = 1$. The visibility term, E_{vis}, is computed by first checking that the projection of $\mathbf{p}$ into C_i lies within the image, and then testing for occlusions by performing a ray cast from $\mathbf{o}_i$ to $\mathbf{p}$ and checking that the ray does not intersect the mesh in before reaching $\mathbf{p}$.

The angular term, E_{ang}, prefers cameras whose viewing angle is as close to parallel with the surface normal at $\mathbf{p}$. This aims to reduce blurriness in the texture by avoiding the use of images which contain foreshortening effects.

The resolution term, E_{res}, prefers cameras whose optical centres are close to $\mathbf{p}$. These close-up images capture the surface details in the highest resolution and are therefore ideal for generating the highest resolution textures possible.

For a given vertex, we rank each input camera according to their total penalty and then define T_{adaptive} as the 3^{rd} largest penalty value. The weight $\tilde{w}_{\mathbf{p}}^{C_i}$ is then computed as:

$$
\begin{aligned}
w_{\mathbf{p}}^{C_i} &= \max\left(1 - \frac{E_{\text{total}}}{T_{\text{adaptive}}}, 0\right), \\
\tilde{w}_{\mathbf{p}}^{C_i} &= \frac{w}{\sum_{C_j \in \mathcal{C}} w_{\mathbf{p}}^{C_j}},
\end{aligned}
$$

corresponding to Unstructured Lumigraph interpolation with $k = 3$, ensuring that only the two cameras with lowest penalty are given non-zero weights, while continuous variations in the penalties lead to smooth variations of camera weights.

To generate the texture for triangle T with vertices $\mathbf{p}_0$, $\mathbf{p}_1$, and $\mathbf{p}_2$ (each of which has been assigned the two cameras that see it best), we form the set of cameras $\mathcal{C}_T$ containing the cameras associated with the vertices of T. $\mathcal{C}_T$ will contain at most six cameras but will often contain fewer than six since it is likely that neighbouring vertices will be best seen by the same cameras. Then for each camera $C_i \in \mathcal{C}_T$, we set the alpha value at each vertex $\mathbf{p}_j$ of T to be $\tilde{w}_{\mathbf{p}_j}^{C_i}$, i.e., the weight of C_i at the j^{th} vertex of T, and then render the triangle into the appropriate section of the texture map. An example texture atlas is shown in Figure 4.

The size N of the triangles in the texture atlas is selected to make most use of the high resolution pixel data in the input images. For a given triangle T in the mesh $\mathcal{M}$, we compute the average length of its edges in pixels when projected into the camera in $\mathcal{C}$ which sees it best. This is repeated for all n triangles in the mesh to compute an overall average projected

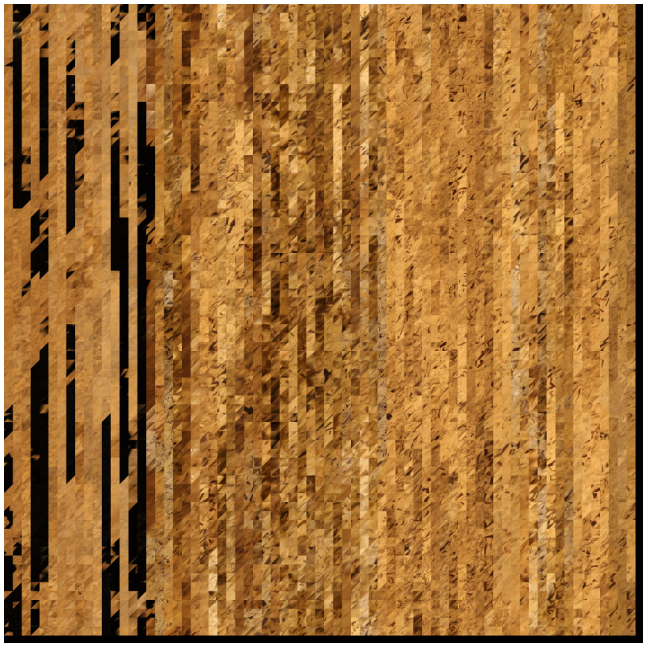

Figure 4: An example texture atlas generated by our method.

edge length which we use as N. These triangles are then packed inside k $M \times M$ texture atlases. We set M to be 4096 in our reconstructions but this can be changed to generate smaller or larger atlases.

When using many close-up images of parchment, the value of N can become very large and generate a number of texture atlases too large to hold in typical GPU memory. For this reason, we include an optional parameter k_{max} which limits the total number of texture atlases generated and selects the value of N which will pack the triangles as tightly as possible into the available texture space.

The advantage of selecting camera weights on a per-vertex instead of per-triangle basis is that we avoid discontinuities across triangle boundaries. Given two neighbouring triangles T_1 and T_2 which share two vertices $\mathbf{p}_0$ and $\mathbf{p}_1$ along an edge $e_{0,1}$, the choice of best cameras for $\mathbf{p}_0$ and $\mathbf{p}_1$ is independent of which triangle we are considering, and so the texture in T_1 will blend smoothly over $e_{0,1}$ into the texture in T_2.

If we just consider the texture within a single triangle T, however, there may still be discontinuities if T is not entirely visible in all of the cameras used to texture it. Such discontinuities happen at visibility boundaries, where a camera C sees some but not all of the vertices of T, and may therefore have a non-zero camera weight even at parts of T which it cannot actually see. To avoid this situation, we erode the visibility mask of C so that a vertex is only considered visible by C if each triangle incident on that vertex is completely visible in C (i.e., if C sees that vertex as well as its neighbours). We use this eroded visibility mask in our computation of E_{vis}.

4. QUALITY METRIC

Professional archival standards for document digitisation require minimum resolutions for the resulting raster images [6]. In the case of planar 2D artefacts that are imaged with flatbed scanners or in a fronto-parallel camera image, this minimum resolution is expressed in dots per inch (DPI). In our case, however, the effective[1] sampling density of the reconstructed parchment varies across the parchment surface, dependent on the imaging conditions of the images contributing to each surface point, therefore assigning a single DPI quality label would not sufficiently characterise the dataset.

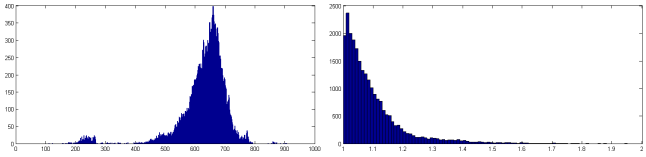

Figure 5: Visualisations of effective sampling density of the parchment reconstruction shown in Figure 8, left. *Left:* Histogram of effective DPI. Most mesh vertices are sampled at around 850 DPI, with a much smaller number of vertices having been captured at approximately 240 DPI. *Right:* Histogram of local sampling stretch. The vast majority vertices exhibit significantly less than 1:1.5 anisotropy.

In order to allow for an assessment of the reconstruction quality in familiar terms, we propose looking at the *distributions* (or histograms) of both effective DPI and local stretch in the sampling across the parchment. Figure 5 shows these distributions at the example of the parchment shown in Figure 8, left. We argue that such histograms provide an excellent way to gauge the quality of a dataset in terms easily communicated to archivists.

We determine the local, effective sampling density at each mesh vertex $\mathbf{p}$ by looking at the mapping of the camera image that contributed the most to the texture at $\mathbf{p}$. Let the corresponding camera's projection matrix be P, we approximate its perspective mapping from the image to the mesh domain as being locally affine, which allows us to express this mapping around $\mathbf{p}$ by its Jacobian $J_P(\mathbf{p})$ [9]. Its determinant $\det J_P(\mathbf{p})$ denotes the inverse surface area an image pixel projects to; hence $d = \sqrt{\det J_P(\mathbf{p})}$ is the local mean sampling density in DPI (assuming units of inches). However, for grazing camera views, the sampling may be anisotropic. The level of anisotropy is given by the Jacobian's condition number, $a = \kappa(J_P(\mathbf{p}))$, implying that the local sampling is $d\sqrt{a}$ DPI in the direction of maximum stretch and $d/\sqrt{a}$ DPI perpendicular to it.

5. RESULTS

Figures 6, 7, and 8 show the sequence of our reconstruction pipeline. Figure 6 shows examples of dense point clouds computed by VisualSFM and PMVS, each containing in the region of 4-million points. Applying Poisson Surface Reconstruction to these point clouds generates triangle meshes with in the region of 400,000 triangles. Figure 7 shows these triangle meshes after being simplified to 50,000 triangles using quadric edge collapse decimation. Finally, Figure 8 shows the textured triangle meshes generated by our texture mapping procedure. In these reconstructions, we restricted

[1]While it is possible to store the same data at arbitrary resolution (by simply up-sampling it), the effective resolution corresponds to the actual resolution at which features are still resolved.

the texture mapping algorithm to generate a maximum of five 4096×4096 texture atlases. The camera calibrations produced by multi-view stereo inevitably contain some error, which can lead to ghosting artefacts in the texture. We found, however, that such artefacts were never sufficiently large to impact the legibility of the text.

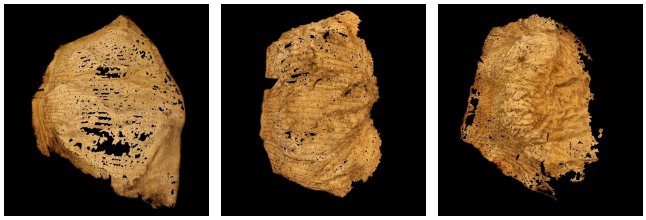

Figure 6: Point clouds generated by VisualSFM and PMVS.

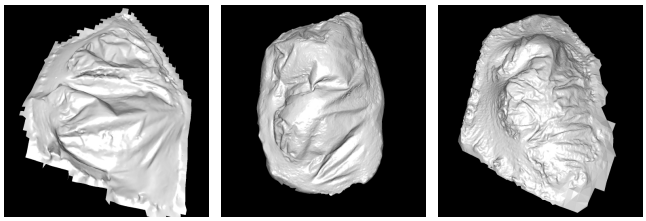

Figure 7: Triangle meshes generated by Poisson Surface Reconstruction.

Figure 8: Textured triangle meshes generated by our reconstruction pipeline.

5.1 System Comparisons

We evaluated our reconstruction method by comparing it to two state-of-the art multi-view reconstruction systems, ARC 3D [22] and Autodesk 123D Catch [1]. Surprisingly, ARC 3D proved unable to produce reconstructions from our input images. We can only speculate that the system uses internal heuristic assumptions that are not met in our dataset.

We hence compare the results of our method with those obtained from 123D Catch using the same input image sets. Figure 10 shows examples of differences in the reconstructions of three different parchments. The images on the left show regions of the reconstructions generated by our pipeline, and on the right the corresponding regions of the reconstructions generated by 123D Catch.

The 123D Catch reconstructions often show significant ghosting effects, suggesting a significant error in the camera calibrations or the reconstructed geometry. These can be seen in Figures 10.h and 10.j, whereas our corresponding reconstructions in Figures 10.g and 10.i are significantly sharper and contain no obvious ghosting.

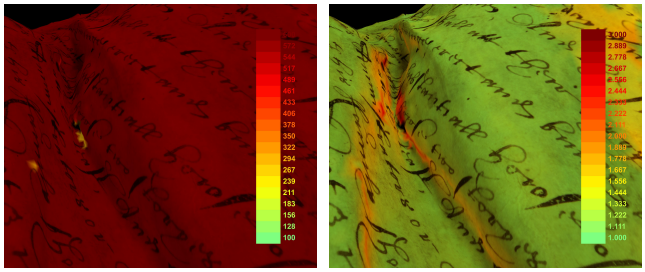

Figure 9: *Left:* effective DPI and *Right:* sampling anisotropy inside a deep crease.

Figure 10.f also shows examples of holes which sometimes appear in the 123D Catch reconstructions. In this case, the holes lie in a region of text and cause loss of textual information. In our reconstruction (10.e) these areas of text are fully reconstructed.

Even in areas where 123D Catch reconstructs the surface accurately the textures almost always have a lower effective resolution than ours. For one parchment it generated only three 4096×4096 texture atlases (each with a large proportion of unused pixels), compared with 11 atlases of the same size in our reconstruction. We suspect this is due to heavy down-sampling of the input images. This can be seen in Figures 10.b and 10.d where the 123D Catch reconstructions are noticeably blurrier than our corresponding results.

5.2 Reconstruction Quality

Figure 11 shows visualisations of how the local effective DPI and sampling anisotropy varies over the surface of eight folios, as well as the corresponding histograms showing the distributions of the sampling rate and anisotropy for each.

We can see that in all eight cases the majority of the surface is sampled at well over the recommended minimum of 300 DPI, and in most cases more than the high quality archival standard of 600 DPI [6], and with low anisotropy of less than 1:1.5. Some areas of the documents are seen to be undersampled. These areas tend to lie on the borders, beyond the extent of the text, where the folios were imaged less densely.

We can also see that the anisotropy tends to be greatest inside creases and on the sides of ridges where the folio is likely to be imaged at a more acute angle. This can be seen in Figure 9 which shows the inside of a very narrow crease. The anisotropy is high since there is no way to image this area in a fronto-parallel manner, but even still we manage to achieve a high effective DPI.

6. CONCLUSION

We have proposed a pipeline for imaging and reconstruction of historical documents, particularly those with complex geometric distortions which would be difficult to capture using standard document scanning methods. We also proposed an intuitive visualisation of the spatially-varying sampling density, providing a means to gauge data quality with respect to archival standards.

Our imaging approach uses a single hand-held camera to capture a set of overlapping images which fully captures the

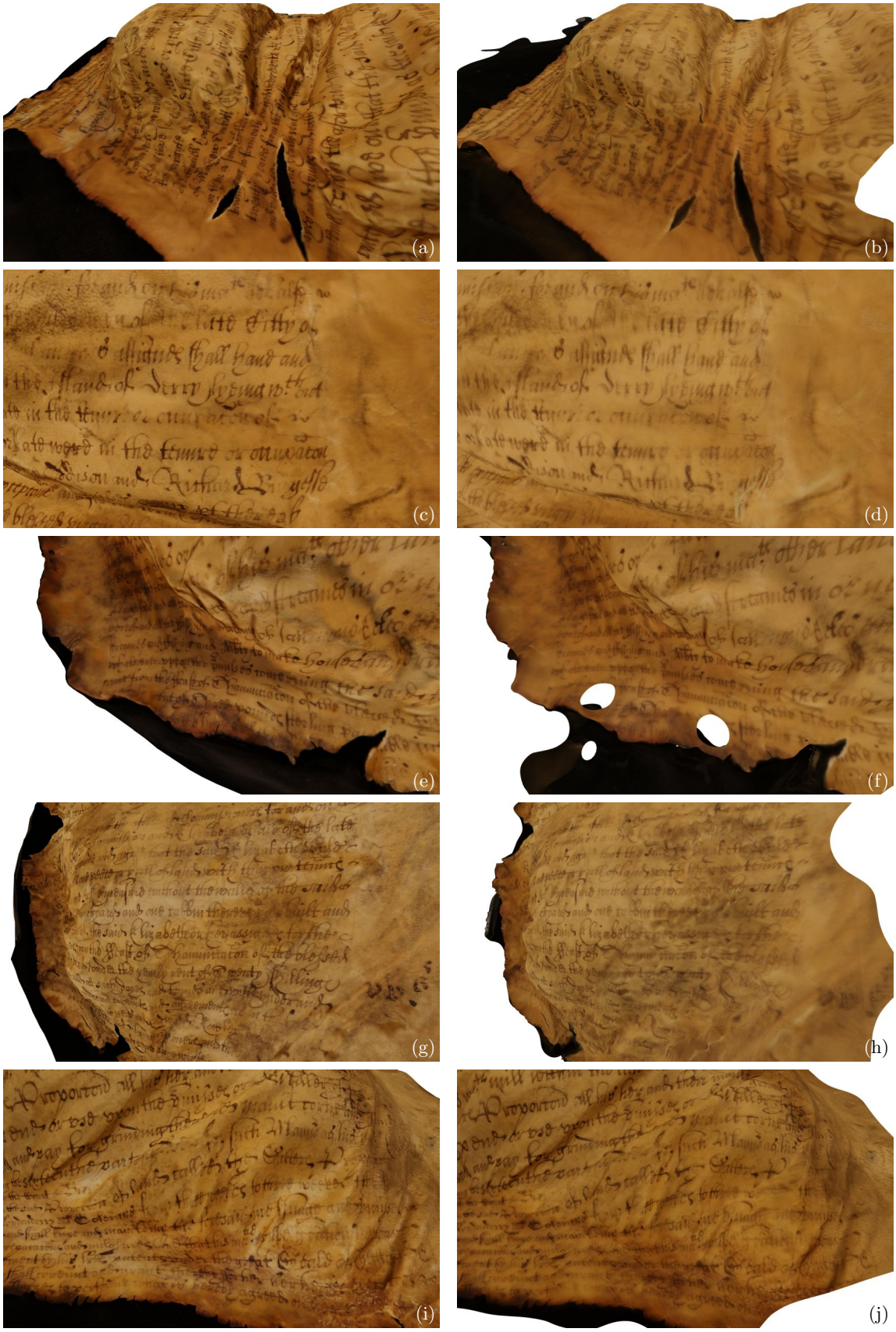

Figure 10: Comparing the output of our reconstruction pipeline with the output of 123D Catch. *Left:* Output of our reconstruction pipeline *Right:* Output of 123D catch

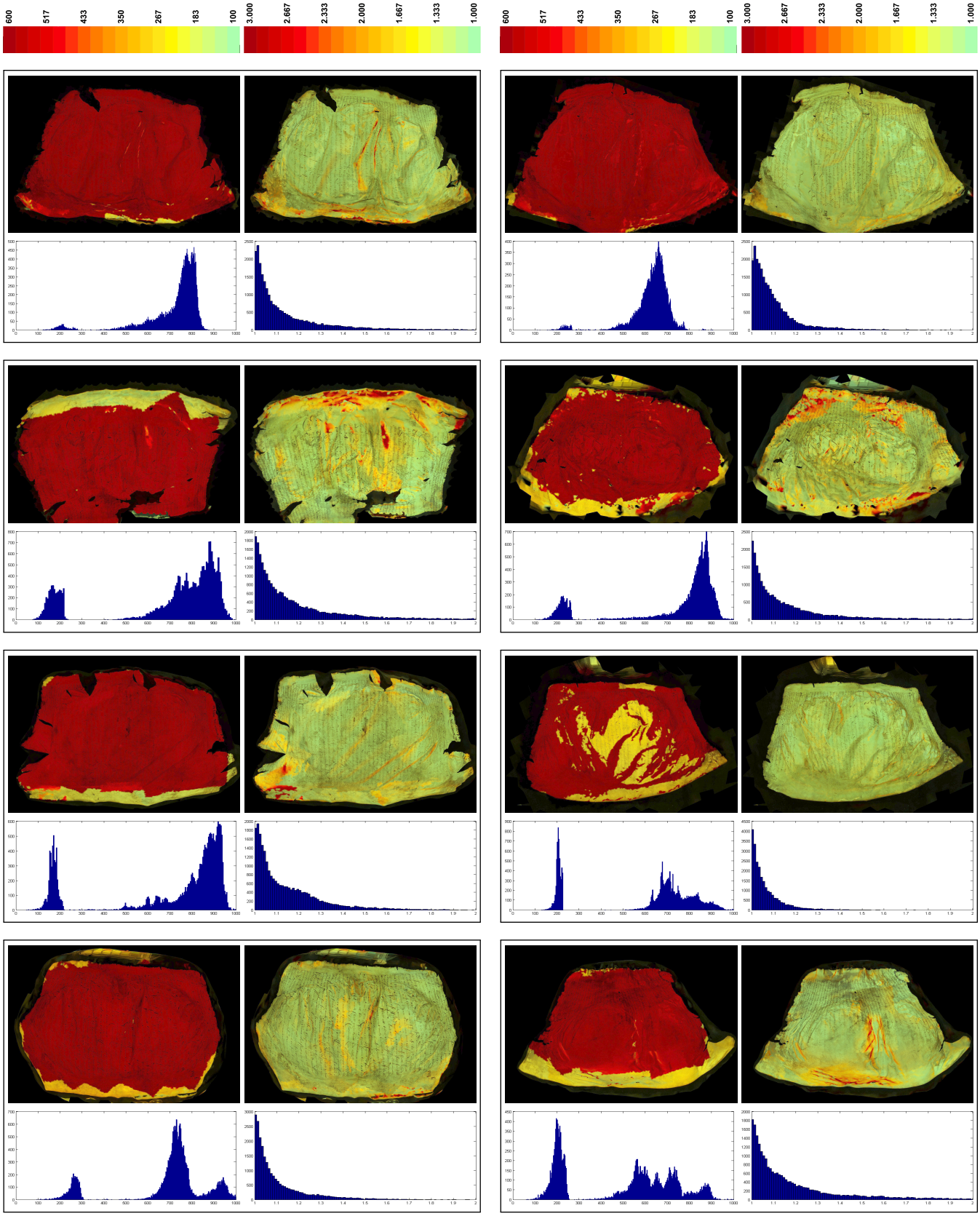

Figure 11: For eight folios we visualise the local effective DPI (left column within each box) and sampling anisotropy (right column within each box) over the folio surface alongside plots of the distributions of these quality metrics.

document surface. This approach gives the flexibility to adapt the imaging procedure to a wide variety of document surface shapes, in a way that a fixed camera arrangement cannot. Since the process does not require specialist equipment, such as structured-light or laser scanners, or large imaging rigs containing many cameras, it could be easily adopted by other archives or museums who are likely to have access to a high-resolution SLR camera.

The reconstruction method generates a high resolution textured 3D model of the document, meeting quality requirements of professional conservation and archival applications.

Acknowledgements

This work was supported by the UCL EngD VEIV Centre for Doctoral Training and by London Metropolitan Archives, who we would like to thank for access to the Great Parchment Book. We would specifically like to thank Caroline De Stefani for helping schedule access to the document, and Alberto Campagnolo for supervising the entire imaging process.

7. REFERENCES

[1] Autodesk. 123D catch. `http://www.123dapp.com/catch`, 2012.

[2] G. Bianco, F. Bruno, A. Tonazzini, E. Salerno, P. Savino, B. Zitová, F. Sroubek, and E. Console. A framework for virtual restoration of ancient documents by combination of multispectral and 3d imaging. In *In Proc. Eurographics Italian Chapter Conference*, pages 1–7, 2010.

[3] M. Brown and W. Seales. Document restoration using 3D shape: a general deskewing algorithm for arbitrarily warped documents. In *Proc. IEEE International Conference on Computer Vision*, volume 2, pages 367–374, 2001.

[4] M. S. Brown, M. Sun, R. Yang, L. Yun, and W. B. Seales. Restoring 2d content from distorted documents. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(11):1904–1916, Nov. 2007.

[5] C. Buehler, M. Bosse, L. McMillan, S. Gortler, and M. Cohen. Unstructured lumigraph rendering. In *Proc. 28th ACM Conference on Computer Graphics and Interactive Techniques*, pages 425–432, 2001.

[6] Federal Agencies Digitization Initiative (FADGI) – Still Image Working Group. *Technical Guidelines for Digitizing Cultural Heritage Materials: Creation of Raster Image Master Files*, Aug. 2010. Available at `http://www.digitizationguidelines.gov/guidelines/FADGI_Still_Image-Tech_Guidelines_2010-08-24.pdf`.

[7] Y. Furukawa and J. Ponce. Carved visual hulls for image-based modeling. *International Journal of Computer Vision*, 81(1):53–67, Jan. 2009.

[8] Y. Furukawa and J. Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 32(8):1362–1376, 2010.

[9] P. S. Heckbert. Fundamentals of texture mapping and image warping, May 1989. Master's thesis, UCB/CSD 89/516. CS Division, EECS Dept, UC Berkeley.

[10] C. Hernández Esteban and F. Schmitt. Silhouette and stereo fusion for 3D object modeling. *Computer Vision and Image Understanding*, 96(3):367–392, 2004.

[11] M. Kazhdan, M. Bolitho, and H. Hoppe. Poisson surface reconstruction. In *Proc. 4th Eurographics Symposium on Geometry processing*, pages 61–70, 2006.

[12] H. I. Koo, J. Kim, and N.-I. Cho. Composition of a dewarped and enhanced document image from two view images. *Image Processing, IEEE Transactions on*, 18(7):1551–1562, 2009.

[13] K. N. Kutulakos and S. M. Seitz. A theory of shape by space carving. *International Journal of Computer Vision*, 38(3):199–218, 2000.

[14] C. H. Lampert, T. Braun, A. Ulges, D. Keysers, and T. M. Breuel. Oblivious document capture and real-time retrieval. In *In Proc. First International Workshop on Camera-Based Document Analysis and Recognition*, pages 79–86, 2005.

[15] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 2004.

[16] Meshlab. `http://meshlab.sourceforge.net/`.

[17] F. Schmitt and Y. Yemez. 3D color object reconstruction from 2d image sequences. In *Proc. International Conference on Image Processing (ICIP)*, volume 3, pages 65–69, 1999.

[18] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: Exploring photo collections in 3D. pages 835–846, 2006.

[19] M. Sun, R. Yang, L. Yun, G. Landon, B. Seales, and M. Brown. Geometric and photometric restoration of distorted documents. In *Proc. IEEE International Conference on Computer Vision*, volume 2, pages 1117–1123, 2005.

[20] Y. Tian and S. Narasimhan. Rectification and 3D reconstruction of curved document images. *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 377–384, 2011.

[21] A. Ulges. Document capture using stereo vision. In *In Proc. ACM Symposium on Document Engineering*, pages 198–200. ACM, 2004.

[22] M. Vergauwen and L. V. Gool. Web-based 3D reconstruction service. *Machine Vision Applications*, 17:411–426, 2006.

[23] T. Wada, H. Ukida, and T. Matsuyama. Shape from shading with interreflections under a proximal light source: Distortion-free copying of an unfolded book. *International Journal of Computer Vision*, 24(2):125–135, 1997.

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

[25] C. Wu. Visualsfm: A visual structure from motion system. `http://www.cs.washington.edu/homes/ccwu/vsfm/`, 2011.

[26] C. Wu, S. Agarwal, B. Curless, and S. M. Seitz. Multicore bundle adjustment. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 3057–3064, 2011.

[27] Z. Zhang, C. Tan, and L. Fan. Restoration of curved document images through 3D shape modeling. In *Computer Vision and Pattern Recognition, 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 1, pages I–10–I–15 Vol.1, 2004.