Colour Texture Fusion of Multiple Range Images

Alexander Agathos, Robert B. Fisher
School of Informatics, University of Edinburgh.
A.Agathos@ed.ac.uk, rbf@inf.ed.ac.uk

Abstract

Texture mapping of manifolds that were reconstructed from different range views requires the fusion of the texture maps acquired from different views. Illumination conditions and camera technology prevents colour continuity in the fusion process leading to observable discontinuities in the colour (and also lightness). This paper presents a simple method for the correction of this discontinuity. First a global correction is applied that largely removes the systematic differences between the individual texture maps. Local corrections refine the small observable variations in colour that may still exist near the fusion boundaries.

1. Introduction

When constructing complete 3D models, the fusion of different colour images taken with a CCD camera requires several aspects to be considered. This is in addition to the 3D shape fusion arising from merging multiple 3D scans. First a calibration of the camera with the range data is needed: for each point of the range image a pixel (texel) of the picture corresponding to this view should be found. For this kind of calibration various approaches have been proposed (eg [1]). When computing the texture of a manifold reconstructed from multiple range images, a 3D point could be observed in more than one intensity image. This means that the manifold may be coloured by patches from different images. This can lead to an unrealistic representation of the object's texture because of :

- Ghosting effects (images not calibrated correctly with the range views)
- Colour differences between the images, an issue that has to do with illumination and camera quality

In our experiments we used a flatbed 3D scanner with a colour camera attached to the robot arm that drives the head.

The calibration technique described in [1] ensures a correct texture map spatial registration between different views, reducing the appearance of ghosting effects to a state that does not generate visually displeasing results. If these effects can't be removed by a calibration technique then image based methods [2] can be applied.

Even with careful spatial registration and use of a good colour still camera (Kodak DC-265) we could observe a significant discontinuity between images. What was very distinct was slightly different tints to the different texture maps, which was observable as a discrete hue boundary at the point where the selection of texture maps changed. Some averaging is possible in overlapping regions, but there is still a distinct hue difference in other areas. This discrepancy probably arises because illumination conditions are slightly different (causing lightness variations) and the camera hue capture is non-linear. Even with uniform lighting conditions these discontinuities persisted. An example is shown in Figure 1 (Image can be seen clearer at:
http://homepages.inf.ed.ac.uk/s0234242/Texture/)

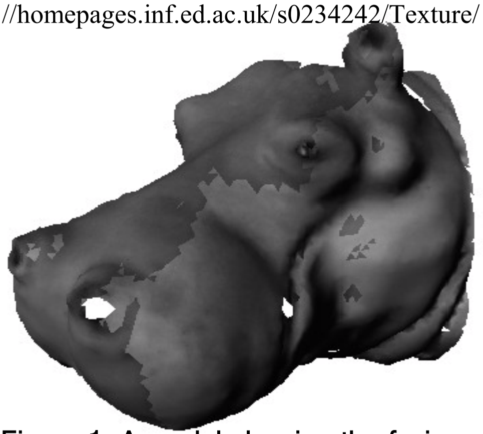


Figure 1. A model showing the fusion of two images belonging to two different views of a model. At the above mentioned URL the hue difference between these two views is clearly visible. Here we have exaggerated the lightness differences to make clearer the discontinuity.

In this paper we concentrate on the fusion of the textures that belong to two different range views. This can easily be expanded to more. We assume that the range views are registered in a common coordinate system (e.g using ICP [7], [8], [9]) and the overlap is known or can be computed with a nearest neighbour algorithm. We assume that we have similar but not identical camera photo response.

There exist alternative ways of acquiring the colour in range images using e.g. RGB lasers [5]. This is an expensive way but helps in acquiring consistent colour over the whole object. Our approach for correcting the color of the images acquired from two different views has two stages:

1) A global correction that does a colour transform estimated from the two different colours observed at pixels in the overlap region.
2) A local correction to smooth out small colour variation boundaries that are left out.

To our knowledge the only direct correction of the RGB channels was done in [3, 4, 11]. In [4] the authors only considered corrections to the boundaries where there is a texture transition from one view to the other. In [3] they propose a multiresolution approach that smooths in a hierarchical manner the different patches. By this way they proceed from local to global correction. While this reduces observable colour discontinuity boundaries, it does not overcome the problem of a slight greenish hue blending into a slight pink hue, even though both hues should be identical. Our approach is different than [3] because we first consider a global transformation and then do some local correction near the boundaries. We believe that by this way the user can choose a view that best represents the object s/he wants to texture and then propagate this illumination and colour to all other views.

In [11] the authors applied a global correction transform treating the colour channels independently. Our approach is different than [11] since we use instead a 3x3 global correction matrix, which allows for cross-talk between the channels, since there is always some correlation between them [10].

2. Global Correction

What is normally expected when someone takes two different images of an object is for the illumination and colour to be different. Even with perfect illumination conditions and high quality CCD cameras this is true. What one could expect is that there should be a colour transformation that could correct this difference in colour and illumination. It turns out that there exists a simple colour transform that could achieve this. Let $S_k(\lambda)$ be the *surface reflectance* at wavelength λ of the nth surface point, $I_n^{(j)}(\lambda)$ *the illumination* of the jth view of the point and $C_i^{(j)}(\lambda)$ the sensor response of the ith *colour channel* of the jth view.

The measured colour $M_{in}^{(j)}$ of channel i that the camera of the jth view receives from point n is:

$$M_{in}^{(j)} = \int_\lambda N_{in}^{(j)}(\lambda) S_k(\lambda) d\lambda, \qquad (1)$$

$$N_{m}^{(j)} = C_i^{(j)}(\lambda) I_n^{(j)}(\lambda)$$

Let's assume that we have two views j, k and we want to correct the jth view to the kth. From now on we will drop the subscript 'n' but recall that the same process occurs at each point 'n'. Also let's assume that there exists a linear transformation matrix $T_{j\to k}$ such that:

$$T_{j\to k}\left(N_i^{(j)}(\lambda)\right) = N_i^{(k)}(\lambda) \qquad (2)$$

This formula transforms the environmental (camera + illumination) conditions of one view to the other.

We have not found any exact previous theoretical justification for this function, but it is motivated by the standard colour gamut transformation used in colour matching.

In the simplest case of a pure brightness change, the transformation is a diagonal matrix $\sigma \mathbf{I}$, where σ is a scalar representing the brightness change. We allow for off-diagonal terms in $T_{j\to k}$ to account for cross - talk between colour channels. The linear model is justified because it does reduce the colour discrepancy, as seen below.

Replacing the channels with Red, Green and Blue we can rewrite the above equation as:

$$T_{j\to k}\begin{pmatrix} N_R^{(j)}(\lambda) \\ N_G^{(j)}(\lambda) \\ N_B^{(j)}(\lambda) \end{pmatrix} = \begin{pmatrix} N_R^{(k)}(\lambda) \\ N_G^{(k)}(\lambda) \\ N_B^{(k)}(\lambda) \end{pmatrix} \qquad (3)$$

$$\begin{pmatrix} M_R^{(k)} \\ M_G^{(k)} \\ M_B^{(k)} \end{pmatrix} = \begin{pmatrix} \int N_R^{(k)}(\lambda) S(\lambda) d\lambda \\ \int N_G^{(k)}(\lambda) S(\lambda) d\lambda \\ \int N_B^{(k)}(\lambda) S(\lambda) d\lambda \end{pmatrix} =$$

$$\int \begin{pmatrix} N_R^{(k)}(\lambda) \\ N_G^{(k)}(\lambda) \\ N_B^{(k)}(\lambda) \end{pmatrix} S(\lambda) d\lambda = \int \begin{pmatrix} T_{j\to k} N_R^{(j)}(\lambda) \\ T_{j\to k} N_G^{(j)}(\lambda) \\ T_{j\to k} N_B^{(j)}(\lambda) \end{pmatrix} S(\lambda) d\lambda =$$

$$T_{j\to k}\int\begin{pmatrix}N_R^{(j)}(\lambda)\\ N_G^{(j)}(\lambda)\\ N_B^{(j)}(\lambda)\end{pmatrix}S(\lambda)d\lambda = T_{j\to k}\begin{pmatrix}M_R^{(j)}\\ M_G^{(j)}\\ M_B^{(j)}\end{pmatrix}$$

The above equation shows that there is a linear transformation to convert the colour of one view to the other:

$$\begin{pmatrix}M_R^{(k)}\\ M_G^{(k)}\\ M_B^{(k)}\end{pmatrix} = T_{j\to k}\begin{pmatrix}M_R^{(j)}\\ M_G^{(j)}\\ M_B^{(j)}\end{pmatrix} \tag{4}$$

For the estimation of $T_{j\to k}$ colour correspondences between the two images are needed. We will use the registered range views that the images correspond to in order to get these correspondences. Specifically we calculate the points ($\mathbf{X}_j$, $\mathbf{X}_k$) that overlap in the two range views. With the registration algorithm [1] we then compute the colour image pixels that these points are projected in the images, ($\mathbf{P}_j$, $\mathbf{P}_k$), $|\mathbf{P}_j| = |\mathbf{P}_k|$. We expand each of these pixels in a neighbourhood (e.g. 5x5) in order to enrich our selection because the spatial resolution is usually smaller than the pixel resolution. This enriches the selection of pixels for the calculation of the matrix.

Not all of the correspondences can be used for the computation of the matrix, because some pixels may lie in highlights or shadows. Also some pixels may vary a lot from the mean of the correspondences. In order to acquire a good correspondence set for an unbiased estimation of the transformation matrix we use two different thresholds. We first introduce the ratio of colour between the corresponding pixels, $r_{ci} = \frac{P^c{}_{ji}}{P^c{}_{ki}}$, i = {1, …, $|\mathrm{P}_j|$}, c = {Red, Green, Blue}. The first threshold is introduced for the calculation of the mean of each of the three ratios, $\overline{r}_c$. For this calculation we do not use pixels that correspond to highlight and shadows by using a minimum and maximum threshold on each pixels intensity. The second threshold r_T actually selects which of the correspondences will be used for the calculation of the transformation matrix. We accept the correspondences where $|\, r_{ci} - \overline{r}_c \,| \leq r_T$. This threshold should be set to a small value (in our experiments we used a ratio of 5-15 % of the mean value). We have observed that by varying the thresholds by small amounts the colour transformation was not significantly different. This means that the ratio should be small but the exact value is not crucial.

Figure 2 shows in black the pixels in the overlap region that are declared to be sufficiently consistent for the estimation of the global correction matrix.

Let V_k, V_j denote the 3xn colour pixel array formed from n point correspondences from views k, j respectively. We can estimate the transformation matrix $T_{j\to k}$ as follows:

$$V_k = T_{j\to k} V_j \Leftrightarrow$$

$$T_{j\to k} = \left(V_k V_j^T\right)\left(V_j V_j^T\right)^{-1}$$

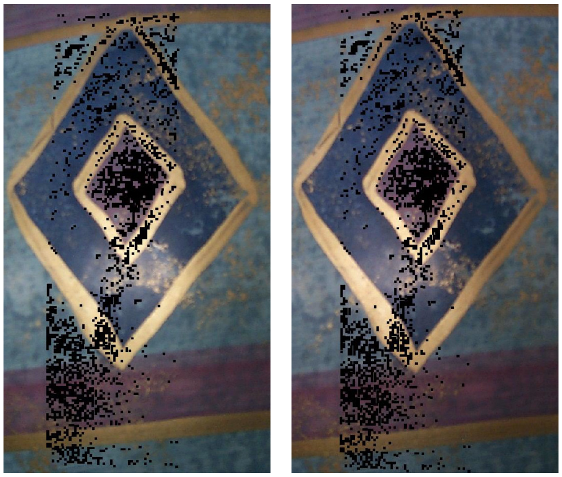


Figure 2. The two images corresponding to two different views. The correspondences and their expansions are marked by black.

Figure 5 shows the matrix $T_{j\to k}$ for the global correction in Figure 3. The reader can observe that it's considerably non-diagonal.

Figure 3 shows the model of Figure 1 after the global correction. We can see here that the differences in lightness have been reduced. At URL http://homepages.inf.ed.ac.uk/s0234242/Texture/ you can see that the differences in hue have been reduced.

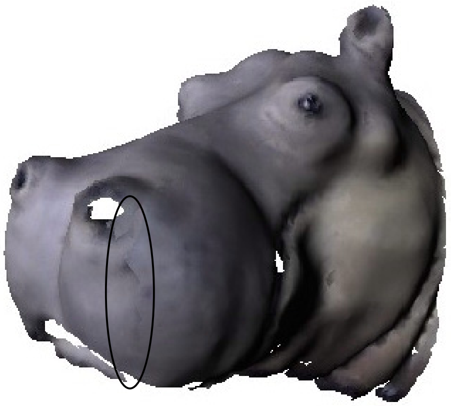


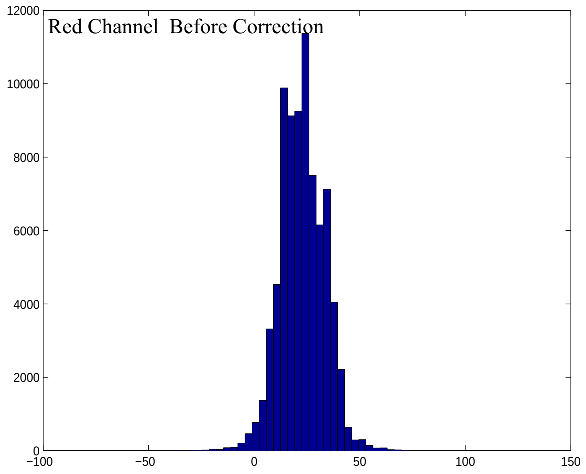
Figure 3. The model represented in Figure 1 after global correction.

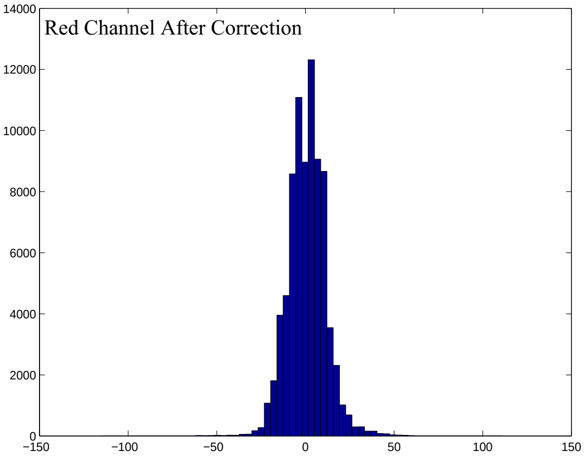
We created a metric to measure the improvement of our global colour correction scheme. For this we use an approach similar to [6]. We built three colour histograms of the RGB values of the two views. Ideally these histograms should be identical except for some noise. To test this hypothesis we used the colour similarity in [6] which measures the difference between the colour histograms of the corresponding points.

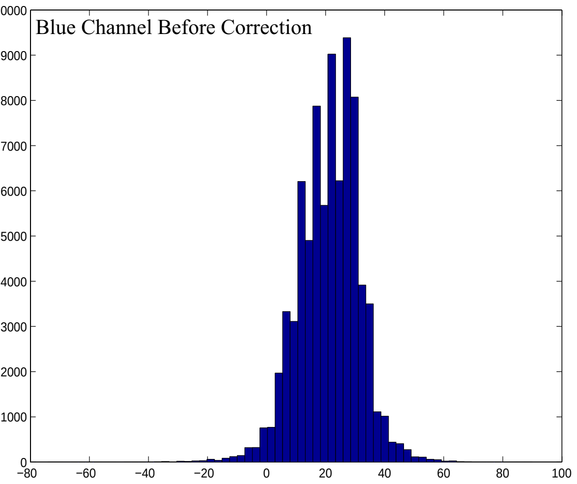
This measure confirmed that there was a colour difference between the two views. The similarity of correspondences before correction in Figure 1 was 0.9464, and the similarity of correspondences after correction in Figure 3 was 0.9795. This analysis for the objects shown in Figures 7 and 9 had similar results with scores of 0.8950 and 0.9668 before correction and scores of 0.9671 and 0.9844 afterwards.

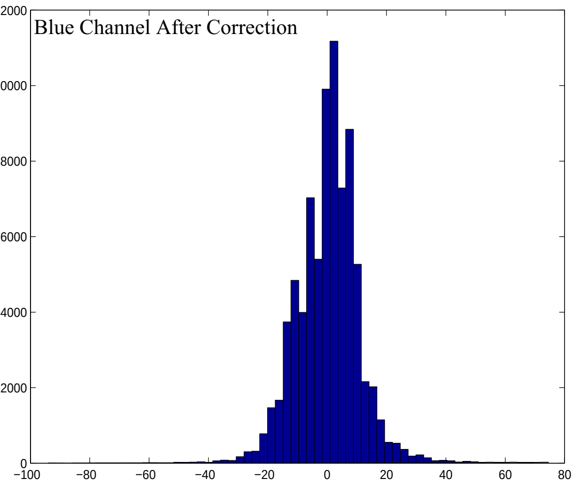
We also computed a second type of histograms of the differences in RGB values for corresponding pixels. Figure 4 shows the colour difference in the Red, Green, Blue components respectively for the model of Figures 7, 8 before and after correction respectively. What we can see in the RGB histograms before correction is a bias away from 0, which is observable as a hue difference. After correction the histogram peaks centre on 0 and thus the hue difference is greatly reduced.

In addition to the observable improvements this helps support our claim that the global correction scheme reduces the colour dissimilarity.









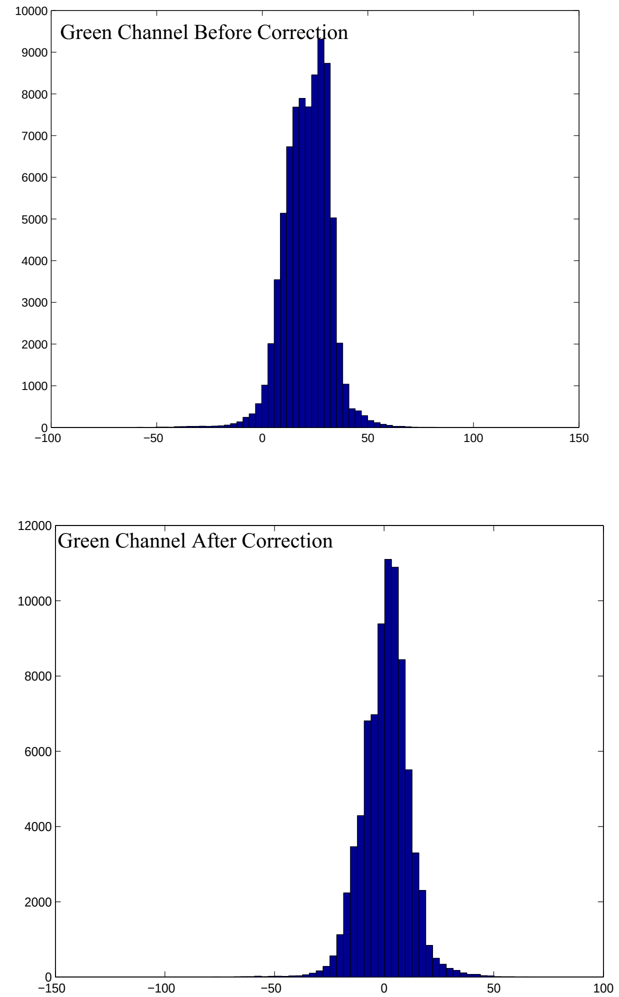


Figure 4. Colour differences histograms for the model in Figures 10, 11

0.7631	0.4918	-0.2127
0.4694	0.6836	-0.1175
0.1416	0.0931	0.9175

Figure 5. The 3x3 global transformation matrix which transforms the colour of the model in Figure 1 to the colour in Figure 3 .

We have also experimented with the lab colour space [10] and achieved similar results. Specifically we multiply the brightness term l_j of image j with the term

$$L_{j \to k} = \frac{\sum_{i \in |Pj|} l_i^{(k)} l_i^{(j)}}{\sum_{i \in |Pj|} l_i^{(j)} l_i^{(j)}}$$

The a, b terms are corrected by a 2x2 transformation matrix estimated in the same way as we have described for the RGB colour space. We observed that this matrix was also not diagonal.

3. Local Corrections

As we already mentioned, after the global transformation the colour similarity between the two different views increases but in practise is never perfect. This means that at the borders of texture transition there might be some observable colour dissimilarities. Figure 3 shows some of these in an area marked in a circle. These dissimilarities can be further reduced by local corrections. Figure 6 illustrates the border regions of this model.

Our local reduction scheme is similar to [3], however we do not use a hierarchical approach. Specifically each triangle whose vertices appear in more than one view is called a *frontier* triangle.

We can use these triangles to smooth the texture map along the boundary outline by diffusing the colour differences of the vertices of the frontier triangles along the border region towards the interior. The vertex colour at the frontier triangles are averaged exactly. Texture map pixels not lying on a vertex are averaged iteratively with a uniform neighbourhood smoothing window until the texture map near the border regions is blended. See [3] for more details. If the pixel to real 3D data resolution is significantly smaller then we could introduce additional 3D points to enrichen the initial colour differences between the different views. These differences can be spread to the border region by averaging.

Figure 7 illustrates the result of applying a smoothing operator near the boundary regions.

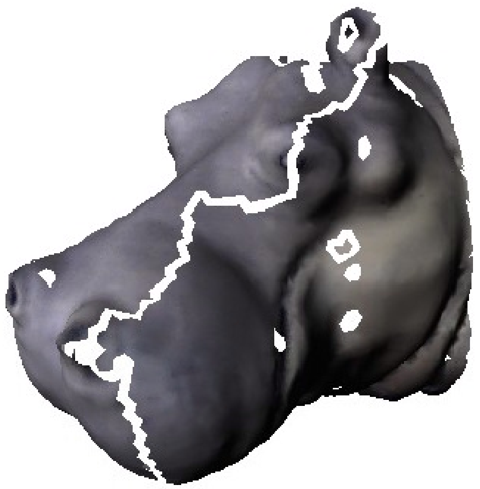


Figure 6. The frontier triangles of the mesh in Figure 3 have been taken out, leaving regions of contiguous texture.

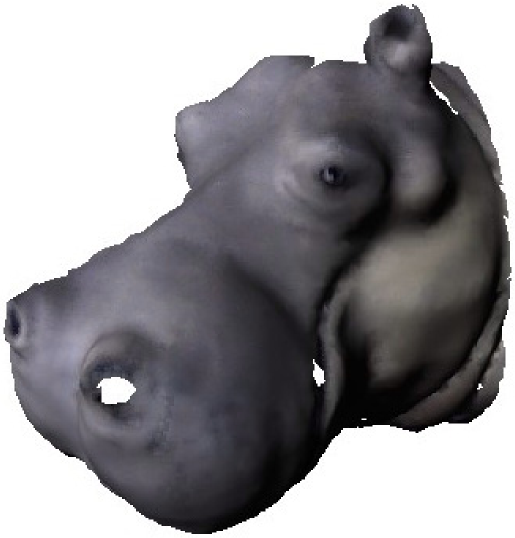


Figure 7. The model of Figure 1 after global and local corrections.

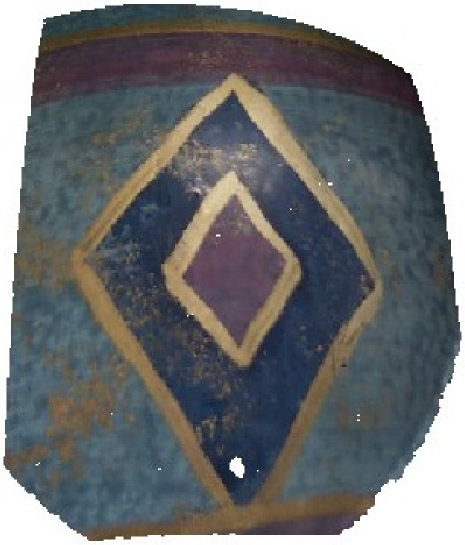


Figure 9. Model after correction

4. Experiments

We show here two additional models in Figures 8, 9, 10, 11 to illustrate results before and after the application of global and local corrections. The hue improvements can really only be seen at this URL: http://homepages.inf.ed.ac.uk/s0234242/Texture/, but some interesting intensity discontinuities can also be seen in the circled areas. All these models were implemented in VRML format.

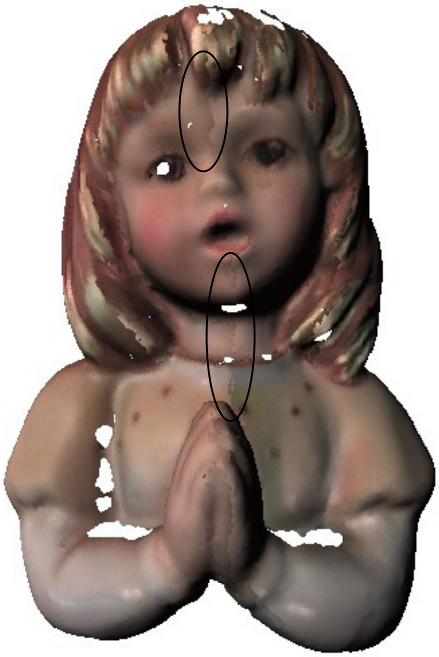


Figure 10. Model before correction

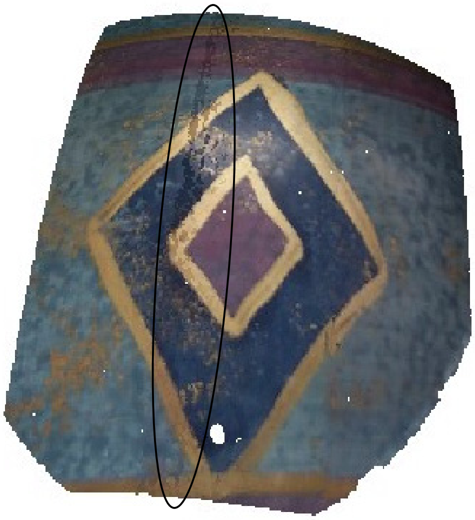


Figure 8. Model before correction

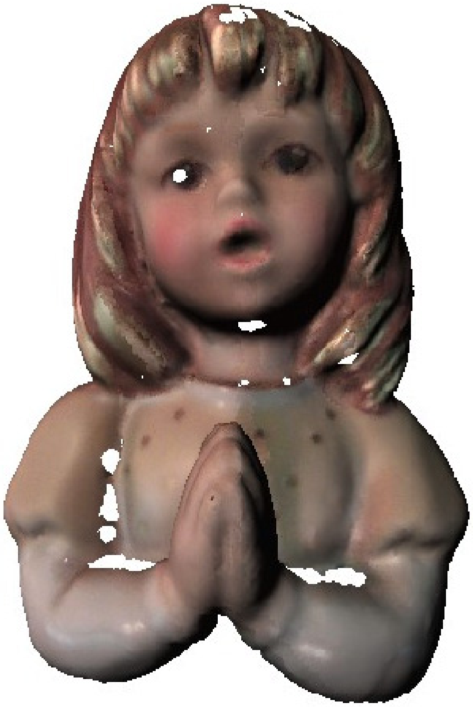


Figure 11. Model after correction

We have also experimented with an extreme case where the colour of the images corresponding to the range views were significantly different. In general though we recommend the use of similar colour. The results are shown in Figures 12, 13.

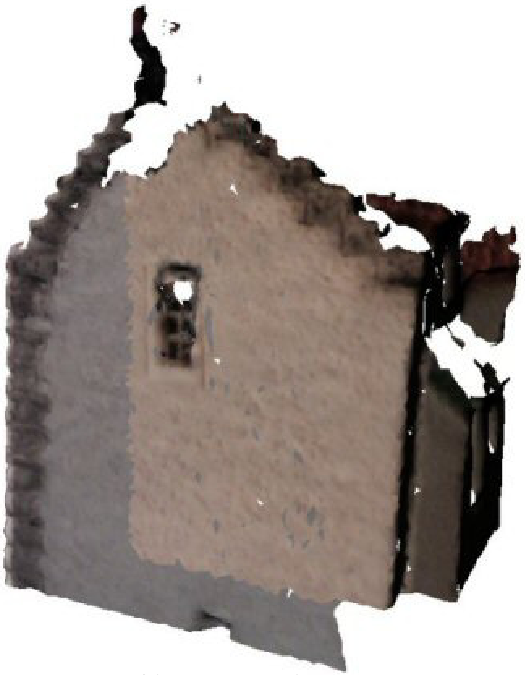


Figure 12. Model before correction

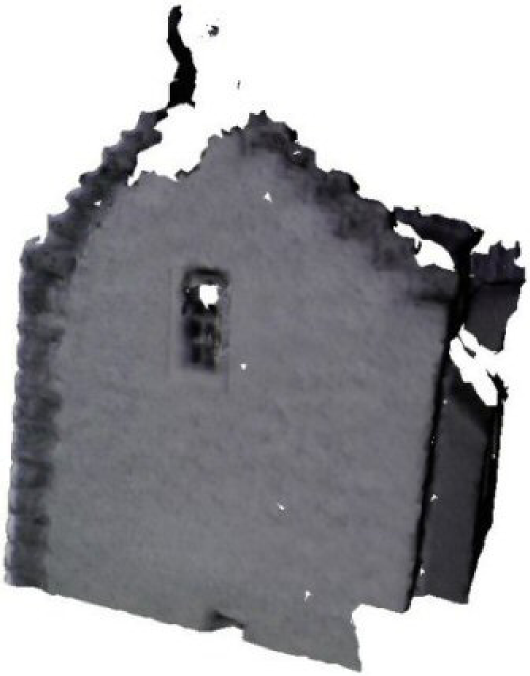


Figure 13. Model after correction

5. Discussions / Conclusions

Summarising our work:

- We used global and local corrections in order to diffuse the texture from two different views.
- Our method works as supported by the visual evidence and histogram analysis.

In order for us to do correct texture fusion we needed

- a good 3D registration
- a good quality camera, i.e. not a web camera
- close colour to begin with, highlights and shadows are still a different problem

6. Future Work

A correction of the highlights and shadows effects is needed. In the case of the fusion of multiple images

- a single reference best image can be selected
- or a mean "true colour" can be computed that all images should match

Also anisotropic smoothing can be used to reduce the small amount of blurring at reflectance boundaries.

7. References

[1] C. Robertson, and R. B. Fisher, “Empirical Calibration Method for Adding Colour to Range Images”, *3D Data Processing, Visualization and Transmission*, Int. Conf., Padova, 2002, pp. 558-561

[2] F. Bernardini, I. Martin, and H. Rushmeier, “High-quality Texture Reconstruction from Multiple Scans”, *IEEE Transactions on Visualization and Computer Graphics,* 7(4), 2001, pp. 318-332

[3] M. Callieri, P. Cignoni, C. Rocchini, and R. Scopigno, “Weaver, an automatic texture builder”, *3D Data Processing, Visualization and Transmission*, Int. Conf., Padova, 2002, pp. 562-565

[4] C. Rocchini, P. Cignoni, C. Montani, and R. Scopigno, “Acquiring, Stitching and Blending Appearance Attributes on 3D Models”, *The Visual Computer*, Springer International, 18(3), 2002, pp. 186-204

[5] M. Soucy, G. Godin, R. Baribeau, F. Blais, and M. Rioux, “Sensors and Algorithms for the Constructionof Digital 3-D Colour Models of Real Objects”, *ICIP Proceedings*, Lausanne, Switzerland, 1996, pp. 409-412

[6] A. Jain, and A. Vailay, “Image Retrieval Using Colour and Shape”, *Pattern Recog.*, 29(8), 1996, pp. 1233-1244

[7] P. Besl, and N. McKay, “A Method for Registration of 3-D Shapes”, *Trans. PAMI*, 14(2), 1992, pp. 239-256

[8] Y. Chen, and G. Medioni, “Object Modeling by Registration of Multiple Range Images”, *Proc. IEEE Conf. on Robotics and Automation*, 1991, pp. 145-155

[9] S. Rusinkiewicz, and M. Levoy, “Efficient Variants of the ICP Algorithm”, *Proc. 3DIM*, 1997, pp. 145-152

[10] E. Reinhard, M. Ashikhmin, B. Gooch, P. Shirley, "Color Transfer Between Images", *IEEE Computer Graphics and Applications*, 21(5), (2001)(special issue on Applied Perception), pp. 34-41

[11] H. Rushmeier, F. Bernardini, “Computing consistent normals and colors from photometric data”, *Proc. 3DIM*, 1999, pp. 99-108