UNIVERSITÀ DI PISA

Scuola di Dottorato in Ingegneria "Leonardo da Vinci"

Corso di Dottorato di Ricerca in
Ingegneria dell'Informazione

Tesi di Dottorato di Ricerca

Uses of uncalibrated images to enrich 3D models information

Autore:

Matteo Dellepiane

Relatori:

Prof. Luca Simoncini

Dott. Roberto Scopigno

Anno 2009

Abstract

The decrease in costs of semi-professional digital cameras has led to the possibility for everyone to acquire a very detailed description of a scene in a very short time.
Unfortunately, the interpretation of the images is usually quite hard, due to the amount of data and the lack of robust and generic image analysis methods. Nevertheless, if a geometric description of the depicted scene is available, it gets much easier to extract information from 2D data.
This information can be used to enrich the quality of the 3D data in several ways. In this thesis, several uses of sets of unregistered images for the enrichment of 3D models are shown.
In particular, two possible fields of application are presented: the color acquisition, projection and visualization and the geometry modification.
Regarding color management, several practical and cheap solutions to overcome the main issues in this field are presented. Moreover, some real applications, mainly related to Cultural Heritage, show that provided methods are robust and effective.
In the context of geometry modification, two approaches are presented to modify already existing 3D models. In the first one, information extracted from images is used to deform a dummy model to obtain accurate 3D head models, used for simulation in the context of three-dimensional audio rendering. The second approach presents a method to fill holes in 3D models, with the use of registered images depicting a pattern projected on the real object.
Finally, some useful indications about the possible future work in all the presented fields are given, in order to delineate the developments of this promising direction of research.

Abstract

La diminuzione dei costi delle fotocamere digitali semi-professionali ha portato alla possibilit per tutti di acquisire immagini ad alta definizione in modo molto semplice. Tuttavia, l'interpretazione di queste immagini, nell'ambito di tecniche di analisi della scena di ricostruzione 3D della stessa, risulta ancora molto difficile a causa della ricchezza di informazione acquisita. Nel caso in cui si conosca per una rappresentazione, anche semplificata, della scena, possibile estrarre dati interessanti in maniera automatica o semi-automatica.
Questi dati possono essere utilizzati in diversi modi per arricchire la qualit dei dati 3D in possesso. Nell'ambito di questa tesi, sono quindi presentate alcune tecniche per l'uso di immagini non registrate per l'arricchimento di modelli 3D.
In particolare, due possibili campi di applicazione sono considerati: l'acquisizione, proiezione e visualizzazione dell'informazione di colore e la modifica della geometria di partenza.
Per quanto riguarda la gestione del colore, sono presentate alcune soluzione pratiche ed efficaci, che hanno portato a importanti risultati nell'ambito di svariati progetti nell'ambito in particolare dei Beni Culturali.
Considerando invece le tecniche di modifica della geometria, sono presentati due approcci che apportano cambiamenti nella topologia di modelli 3D gi esistenti. In particolare, nella prima tecnica le informazioni estratte dalle immagini sono usate per produrre modelli tridimesionali di teste umane, usati per simulazioni di *sound scattering* nell'ambito di applicazioni di *3D sound rendering*. Il secondo metodo permette invece di completare modelli 3D con buchi, utilizzando immagini dell'oggetto reale su cui sia stato proiettato un pattern laser predefinito.
Infine, sono presentate alcune interessanti indicazioni a proposito di possibili sviluppi futuri dei metodi proposti, per delineare la direzione di questo promettente argomento di ricerca.

Acknowledgments

Sono passati piú di sei anni da quando ho scritto i ringraziamenti della tesi. Un sacco di tempo. Mi ritrovo con un po' piú di esperienza e idee chiare, ma con ancora tanta strada davanti.
I primi ringraziamenti vanno a Cinzia che mi é stata vicino in questi anni di viaggi e lontananze. Spero possiamo mantenerci forti e sorridenti come riusciamo spesso ad essere.
Un abbraccio alla mia famiglia, che si è allargata in questi anni: ancora una volta vi chiedo scusa per i miei silenzi, ma giuro che ci sto provando, a fare meglio...
Devo molto poi a tutto il Visual Computing Lab, un posto fantastico dove lavorare e condividere esperienze. In particolare grazie a Claudio e Roberto per aver "puntato" su di me anche quando il primo a non essere convintissimo sulle mie qualitá ero io. Spero di poter ricambiare la vostra fiducia, e grazie per avermi mostrato come si possa avere successo mantenendo coerenza, umiltá, onestá.
Un pensiero va a tutti gli amici con cui il rapporto si è mantenuto, se non rafforzato, nonostante la lontananza (penso a Guido, Marco, Marcello), ma anche a tutti quelli con cui ci si è persi, a causa della vita ma anche delle nostre mancanze.
E poi grazie alla grande famiglia di Calvari, che con il suo calore semplice mi aiuta sempre a ricordare quali sono le cose importanti della vita.

Contents

1 Introduction **1**

2 Previous and related work **7**
2.1 Color acquisition and visualization 7
2.1.1 Image registration 8
2.1.2 Color projection and visualization 14
2.1.3 Lights and materials 16
2.2 Geometry modification using images 19
2.2.1 Morphing 19
2.2.2 Hole filling 20

3 Color acquisition and visualization **23**
3.1 TexAlign: a user-friendly registration tool 24
3.1.1 The graph of correspondences an its use 27
3.1.2 Processing the graph of correspondences: the workload minimizer 29
3.1.3 Usability tests and results 33
3.2 TexTailor: automatic color projection on 3D meshes 35
3.2.1 The blending function and the use of weights 37
3.2.2 Large dataset management and Results 42
3.3 Flash lighting space sampling 49
3.3.1 Definition, acquisition and processing of FliSS 51
3.3.2 Analysis and validation of the data obtained 55
3.3.3 Application of FLiSS for color projection 59
3.4 Applications 67
3.4.1 The Double David project: mapping highly detailed color information on extremely dense 3D models 68
3.4.2 The Cenobium project: Binding Up Interoperably Usable Multimedia 75

3.4.3 Multiscale acquisition and presentation of very large artifacts: the case of Ripoll's Portalada 80
3.4.4 A peculiar application of TexAlign: using 3D scanning to analyze a proposal for the attribution of a bronze horse to Leonardo da Vinci 89

4 Geometry modification **95**
4.1 Head related transfer function from images 95
4.1.1 The role of HRTF in 3D sound rendering 96
4.1.2 Head model reconstruction from images 99
4.2 Hole filling using registered images 112
4.2.1 The elements of the system 114
4.2.2 Testing and discussion. 117

5 Conclusions and future work **121**
5.1 Color acquisition and visualization 122
5.1.1 Image registration 122
5.1.2 Color projection 123
5.1.3 Color acquisition and artifacts removal 124
5.2 Geometry modification 125

6 List of publications **127**

Bibliography **131**

Chapter 1

Introduction

The research in the fields of Computer Graphics and Computer Vision has reached impressive results in the last few years, and the degree of integration of the two disciplines is nowadays substantial. These two fields of research are strongly related to the use of hardware which is able to provide visual and geometric data.
Hence, the data coming from simple digital cameras up to complex and expensive devices, like 3D Scanners or Computerized Tomography, are analyzed and combined for a variety of applications, e.g. security, recognition, virtual restoration.
Especially in Computer Graphics, the acquisition of the appearance of an object has been subject of intense study, and several methods and application have been proposed.

In particular, images (and videos) are considered as the simplest way to acquire information about the appearance of a scene. A number of different uses of images have been exploited, from automatic recognition or scene interpretation to 3D reconstruction, using both semi-automatic and automatic approaches.
The main difficulty in all the applications which use images is the interpretation of the data: in fact, some of the mechanisms of the human brain which concur in the interpretation of the two-dimensional data (perspective, proportion between objects, distinction between background and important elements of the scene) are very hard to implement. For this reason, automatic applications involving images are sufficiently robust as unsupervised computer programs only in specific applications (recognition of well defined objects, medical data), but tend to fail in more general cases.
Another possible disadvantage of using digital images is the amount of data to be analyzed: nowadays, consumer digital camera produce images with resolution of 8-10 Mpixels (or more). This richness of data can be a drawback due to the time needed for analysis and to the fact that it could be more difficult to extract important infor-

mation. Due to this issues, many methods using images work on subsampled data.

Another very interesting technology which is nowadays highly diffused and studied is 3D scanning. 3D scanners devices are essentially measurement devices: they allow to obtain accurate measurements of the geometry of the scene in a very short time.
In the last ten years, 3D scanning technology has considerably improved, so that the costs of both the devices and the servicing has consistently decreased. This was due to improvements both in hardware and software, so that the time needed to complete a scanning campaign (from the acquisition of the range maps to the production of the final model) has decreased from months to a few days.
Moreover, the offer of low-cost or free solutions to scan [111, 1] or to process the sampled data [44] made this technology interesting also for the wide public.
While being able to provide very precise geometric data, 3D scanning has a couple of main limitations. The first one is that it is sometimes impossible to acquire entire objects (or part of them) due to their constituent material or peculiar geometry. This could lead to incomplete or inaccurate 3D models.
The other main drawback is that 3D scanners are able to provide very precise geometry, but in order to have a convincing visual representation of the object it is also necessary to have some information about its surface properties (color and material). This kind of more complete acquisition is hard to obtain in a general case.

An interesting research direction is to find ways to interpret the information given by images to enrich the quality of scanned models. Given a set of images of a scene and some kind of geometric representation of it, it is possible to use geometric data to help the analysis of the content of images and the extraction of information from them.
The chance to associate portions of images to a well-defined position in space can greatly help in the analysis of the data, the recognition of the important elements of the scene and the extraction of data than can be used to improve the appearance of the 3D model.
In this thesis, some new methodologies to enrich the digital 3D information using a set of uncalibrated images will be presented. Three main fields of application will be analyzed: projection of color data on 3D models (which is the most common application of sampled images), geometry deformation and geometry completion.
The main contributions presented in this thesis concern several important open issues in Computer Graphics and Computer Vision:

- **Image registration.** Before being able to apply any method to enrich geometry, it is necessary that the set of images is registered to the 3D model. This

means that the (extrinsic and intrinsic) camera parameters associated to each image are known or estimated. Since these data are seldom available, a preliminary registration step is usually necessary. An image alignment tool called TexAlign, which permits to align sets of images on 3D models, is proposed. The main strong-points of this tool are: the management of tens of images, the use of image-to-image correspondences and of a workload minimizer (to improve registration speed and decrease user workload), the possibility to handle very different cases (from very small to architectural size models).

- **Color mapping and management.** State-of-the-art solutions for color mapping and visualization are usually not scalable, so that a robust method for the projection of lots of images on complex 3D models is currently not available. TexTailor, a tool for projecting and encoding color information on 3D models, is presented as a convincing solution to handle very complex cases. The color projection is obtained via a per-pixel color blending procedure, which weights the contribution of all the images based on their quality. The color is encoded in a color-per-vertex fashion, and the out-of-core implementation permits to project up to hundreds of images on extremely detailed models.

- **Color artifacts removal.** Most of the visible artifacts on a colored 3D model are related to the quality of the original set of images. In fact, since the lighting environment is usually not known in advance, effects like highlights, shadows or uneven lighting between images can't be removed automatically. Proposed solutions like lighting estimation or controlled lighting setups are scarcely usable in a general case. In order to be able to correct color and remove artifacts from images, Flash Lighting Space Sampling (FLiSS) method is proposed. FLiSS is a color correction space which is based on the analysis of the behavior of the flash of digital cameras. This kind of light is a very simple controlled setup, because it is bound to the camera. Hence, since the position of the light is known for every image of the set, it is possible to automatically correct the color artifacts and obtain high quality results.

- **Practical applications.** Several proposed methods, while technically sound and innovative, are very hard to apply in a general case. This is usually due to the complexity of setup or to the amount of data to be processed. For this reason, this thesis presents several practical applications (mainly in the field of Cultural Heritage) of the methods above mentioned, to show that the proposed solutions are really robust and adaptable to very different cases, from the acquisition of small objects to the visualization of very large and complex architectures.

- **Geometry morphing via images.** The most common application of images on 3D models is the color projection, but the amount of information provided by photos can go well beyond the pure appearance. In this thesis a method to morph a starting geometry using the information extracted from a small set of unregistered images is presented. The peculiar field of application is the production of accurate 3D head models for scattering calculation in the context of 3D audio rendering applications. The method needs a very small amount of input by the user; it is completely automatic and obtains very accurate final models.

- **Geometry completion.** The possibility to create geometry starting from images has been subject of intense study, but if no information about the scene is known in advance, there are still open issues (like the scale of the model, or the distinction between the background and the main objects) which are very difficult to resolve. But if some geometric description is already known, images can be used to complete or improve it. A semi-automatic method to fill big holes in 3D scanned models is presented in this thesis. The information to complete the model is extracted from set of images, depicting a pre-defined pattern projected on the real object. This brings to a coherent closure of the hole, since additional information about the real geometry of the object is obtained.

The structure of the thesis is organized as follows. Chapter 2 will provide a general overview about the state of the art in the use of images to enhance or enrich the geometry. The first part will be devoted to the color projection on 3D models. The Section will be divided in its main issues: the image registration (with overviews on the definition of the camera model and the automatic and semi-automatic estimation of the camera parameters), the color projection and visualization, and the estimation of light environment and object material.
The second part of the Chapter will present some works which use images to introduce modifications in the geometry of the object: particular attention will be given to the methods for geometry morphing and hole filling.
Chapter 3 will present the aspects and methodologies proposed for color information projection on 3D models. In particular, solutions for the issues of image registration (TexAlign) color projection (TexTailor) and light artifacts removal (FLiSS) will be presented in detail, together with results and a discussion about possible improvements.
Moreover, several practical applications of these methods in the field of Cultural Heritage will be presented, showing that the proposed solutions are able to cope with complex and heterogeneous cases. In particular, the difficulties presented by these projects in acquisition and data management were overcome with the use of the above mentioned tools.

Chapter 4 will present two applications of images to modify an already existing geometry. In the first example, a starting 3D dummy of an head is deformed in order to obtain a final results which fits to a very simple input (5 photos and few key-points) provided by the user. The application field of this method is the calculation of Head Related Transfer Function (HRTF) for 3D audio rendering.
The second approach presented used a pre-defined laser pattern projected on a 3D model to fill its holes. Starting from images of the object on which the pattern is projected, the laser projector position is estimated, and geometric data is extracted from images in order to be able to reconstruct missing parts of the model.
Finally, Chapter 5 presents the conclusion and an overview of the possible future improvements for the uses of images to enrich geometry.

Chapter 2

Previous and related work

As already stated, several possible uses of the information provided by a set of images can be exploited to enrich an already accurate 3D model of an object. The most straightforward application is the visualization of color information on the geometry. This operation has been extensively studied in the last few years, although generic and automatic solutions are still not provided. Section 2.1 presents an extensive overview of this field of research, with particular attention to three of the main related issues: image registration, color projection and light and materials estimation. Nevertheless, the information in images could be used to modify and enhance directly the geometry of the model: although this is a relatively less studied field, section 2.2 shows some approaches, especially for dummy morphing and geometry reconstruction.

2.1 Color acquisition and visualization

The accuracy of the acquisition of scanning devices has shown impressive improvement during the last few years. The use of short-range scanners (like the ones based on laser triangulation or structured light) can provide measurements with errors of a few tenths of millimeters. Hence, the quality of the geometric data that can be acquired is very good, as well as the sampling density.
Nevertheless, almost every object presents important information not only in its geometric features, but also in its appearance: colors and material reaction to light are other data that should be carefully acquired and visualized to provide full realism. Unfortunately, most of the scanning devices acquire a low quality color. This is due to two main reasons:

- The resolution of the acquisition for a single range map is always lower than the one of any off-the-shelf digital camera: for example the resolution of the Minolta Vivid 910 is 640x480 pixel. This produces low quality images in terms

of resolution, especially in the case when objects present very fine color details on quite simple geometry (e.g. vases)

- One of the advantages of scanning devices (especially of the laser-based ones) is that they are able to work efficiently even with poor illumination. Hence, no particular lighting setup is needed to acquire the geometry of an object. Unfortunately, in the case of a poor illumination, even if the color is acquired, its quality will be too low to use it for visualization. Moreover, some illumination conditions which can be considered as proper for color acquisition can create problems to the scanner: for example, halogen lights greatly interfere with the acquisition of Triangulation Laser scanners.

Hence, since the color information provided by the acquisition devices is not suitable for use, it is necessary to acquire it in a different moment. The most easy way to acquire the appearance of an object is taking a set of images with a camera. This operation can be done in a short time, and the high resolution provided by digital cameras gives the possibility to cover the entire surface of an object with much less shots than the ones required for the scanning campaign.
As it will be remarked in the next subsections, the acquisition of the images plays a key role in the production of the final model, so the setting of a proper lighting setup is recommended before taking the images: anyway, the photographic campaign is still the easiest and less invasive way to represent the color appearance of an object.

2.1.1 Image registration

The camera model

The alignment of a single image to a 3D model is performed by defining all the parameters of the virtual camera whose position and calibration gives an optimal inverse projection of the image on the 3D model. Camera parameters can be divided in two groups:

- *Extrinsic (or external) parameters:* these are the parameters associated to the position in the space of the camera.

- *Intrinsic (or internal) parameters:* these values are related to the peculiar characteristics of the camera, like the focal length (the ”zoom”) or the distortion introduced by the lenses.

This group of values give the possibility to transform any point in the space in the corresponding point on the image plane of the camera. In fact, given a simple camera

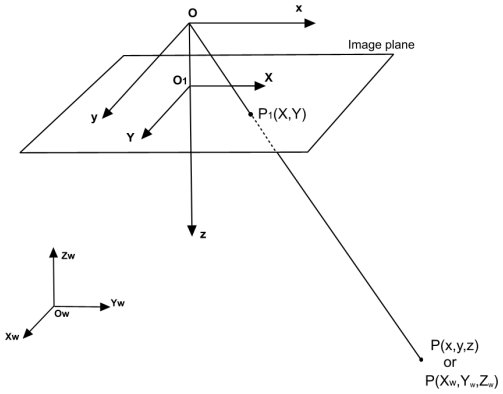


Figure 2.1: A simple scheme of a camera perspective projection

model like the one shown in Figure 2.1, extrinsic parameters can be used to transform a point from its world coordinates (x_w, y_w, z_w) to the camera 3D coordinate system (*x,y,z*):

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = R \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} + T \tag{2.1}$$

In this case the extrinsic parameters are a 3 X 3 rotation matrix R and a translation vector T, which define the *orientation* and *position* of the camera. In order to transform the 3D camera coordinate in 2D image plane coordinates (X_u,Y_u) it's necessary to know the measure of the distance between the point of view and the image plane: this value, indicated with f in Figure 2.1, is usually known as the *focal length*. The relation between the camera and image coordinates of the point can be expressed as follows:

$$X_u = f\frac{x}{z} \ , \ Y_u = f\frac{y}{z} \tag{2.2}$$

Another aspect of the structure of a camera that can be characterized is the distortion introduced by the lenses: if we suppose that the distortion is radial (performed along the radial direction respect to the center of distortion) we can calculate the undistorted image coordinates

$$X_d + D_x = X_u \ , \ Y_d + D_y = Y_u \tag{2.3}$$

where

$$D_x = X_d(k_1 r^2 + k_2 r^4 + ...) \ , \ D_y = Y_d(k_1 r^2 + k_2 r^4 + ...) \tag{2.4}$$

and

$$r = \sqrt{X_d^2 + Y_d^2} \tag{2.5}$$

so that the parameters that define the distortion are k_i. If we suppose also that the center of distortion is not corresponding to the center of projection, its displacement p_1 and p_2 in the image axes can be estimated as well.
In conclusion, in order to obtain a good registration of an image respect to a 3D model, it's necessary to estimate 12 extrinsic parameters (9 for rotation and 3 for translation) and from 1 (focal length) to 5 (focal, distortion coefficient and center of distortion) or more intrinsic parameters.

Estimation based on correspondences

Although some automatic methods will be presented in the next subsection, the most generic and robust way to align a set of images on a 3D model is to provide a set of 2D-3D points correspondences, and estimate the parameters by minimizing a predefined function.
Part of the intrinsic parameters can be provided by the camera manufacturer, or they can be estimated once and assumed as constant for every image acquired. Several automatic and semi-automatic systems to estimate intrinsics parameters have been created. As an example, the approach by Zhang [144] provides very good results using a very simple calibration pattern.
Once that intrinsics are known in advance, it is necessary to estimate only the position in the space of the camera: this problem is known as pose estimation, and one of the best solutions was provided by Kumar [89]. In the case of the more general problem, where all the parameters have to be estimated, several possible mathematic approaches have been proposed, but the most widely known and used is the Tsai method [133]. It uses a two-stage technique to compute: first, the position and orientation and, second, the internal parameters of the camera. The Tsai model describes the camera using 11 parameters: 6 for extrinsics (3 for translation component, 3 for rotation component) and 5 for intrinsics (1 for focal length and 4 for lens distortion). The method was implemented for both coplanar and non-coplanar points constellation, and it proved to be very simple and robust. Another strong point is the fact that an implementation of the method is freely available on the web [5].
Other calibration procedures, like the one proposed by Faugeras and Toscani [60], which will be presented in Section 3.1 differentiate mainly for the minimization function chosen for the estimation. An interesting overview and comparison of calibration methods can be found in [141]. Two different applications (hybrid architectural modeling from images and geometry and facial expressions) can be found in [51] and [119] (Figure 2.2).

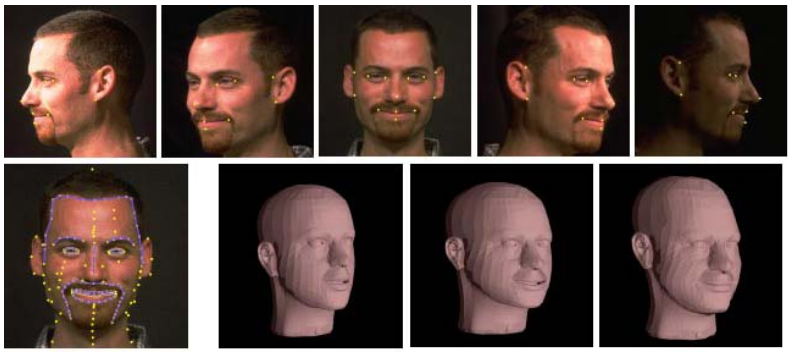


Figure 2.2: Some images from [51]: several points are indicated on both images and a dummy model; these correspondences are used to map images on the model, in order to produce realistic facial expressions.

Automatic estimation of camera parameters

The issue of automatically aligning a set of uncalibrated images to a 3D model is an important topic for both the Computer Graphics and Computer Vision community. An automatic planning of the number of images and of the positions of the camera could lead to good results [107] and reduce the importance of the registration step, but this cannot be considered as a general solution, since it would be needed to transport the object to acquire inside a specific setup. This is usually impossible especially for the application in the Cultural Heritage context, where frequently the objects must not even be touched during acquisition.
Neugebauer et al [110] presented an hybrid approach where the estimation based on correspondences is combined with a registration based on the analysis of the image features (see Figure 2.3). This semi-automatic approach needs a preliminary calibration of the intrinsics of the camera. Moreover, one of the hypotheses is that the illumination must be the same for all the images: this brings to lower quality color information, because there will be no possibility to eventually remove illumination artifacts during the color projection phase (see Sections 3.2 and 3.3).
Ikeuchi [81] presents an automated 2D-to-3D registration method that relies on the reflectance range image. However, also in this case the algorithm requires an initial estimate of the image-to-range alignment in order to converge.
Several other algorithms try to find the camera transformation by minimizing the error between the contour found in the image and the contour of the projected 3D model [30, 82, 101, 106]. The error is typically computed as the sum of distances between a number of sample points on one contour to the nearest points on the other

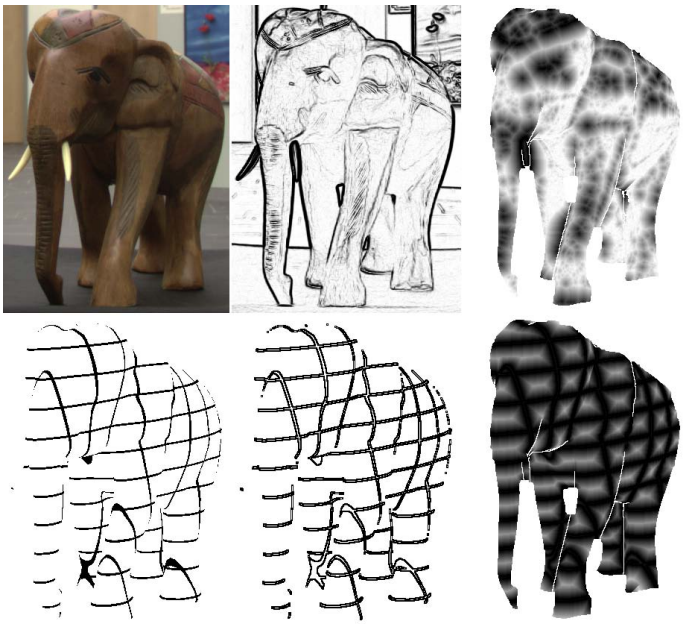


Figure 2.3: Image analysis of an object from [110]: image features are explored to enhance registration to 3D model.

[101]. Another approach computes the sum of minimal distances of rays from the eye point through the image contour to the model's surface, which are computed using 3D distance maps [30]. The work by Lensch [91] proposed a robust implementation of previous silhouette based techniques, introducing a similarity measure to compare them. Moreover, the whole pipeline from registration to texturing was covered with very robust and almost automatic solutions (Figure 2.4).

Unfortunately, the use of silhouette matching has two important limitations: the first one is that it must be easy to distinguish the object with respect to the background: this needs controlled setup acquisition, or a time-consuming "block out" manual session. The second one is that the object must be entirely present inside each image: this can be a very big drawback when a quite big object must be acquired, and the aim is to preserve the detail of color. In this case, it could be necessary to frame only portions of the object, and this prevents silhouette methods to work properly.

A recent paper for 3D-3D and 2D-3D automatic registration [99] was proposed to work in a more general case, but under the assumption that the 3D scene contains clusters of vertical and horizontal lines. Analogously, other previous approaches like [98] need to exploit orthogonality constraints. The main application for this group

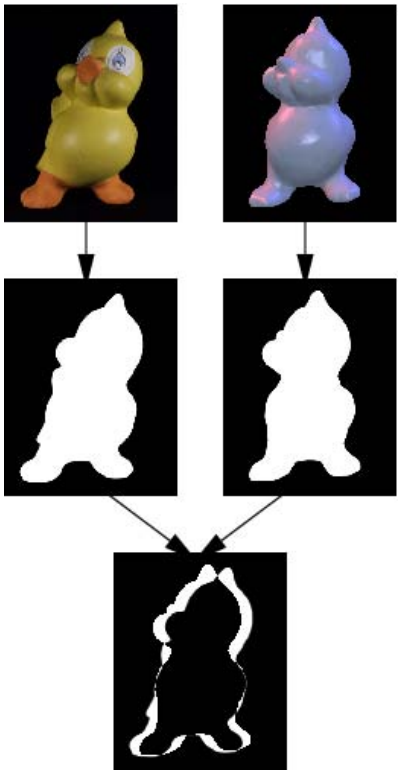


Figure 2.4: Silhouette comparison from [91]: the silhouette of image and model are compared to calculate a similarity measure.

of works stand in the field of architectural models (see Figure 2.5). Finally, another interesting field of work for the automatic image registration is the use of mutual information, which is a statistical measure of *dependency* between two data sources. Two of the first methods implementing it were the one by Viola and Wells [140] and the almost contemporary work by Maes et al. [103]. After these, several registration methods based on MI have been proposed. A comprehensive overview is presented in [121].
In general, most of these works regard simple geometric transformations like 2D roto-translation or affine transformation, so that some of the issues related to the camera model registration are not present. Another characteristic of these methods is related to the resolution of medical data (Figure 2.6), that is often quite poor in quality. Another key issue in the use of MI is the choice of the optimization strategy to achieve the maximization; the comparative evaluation in [104] presents the pros

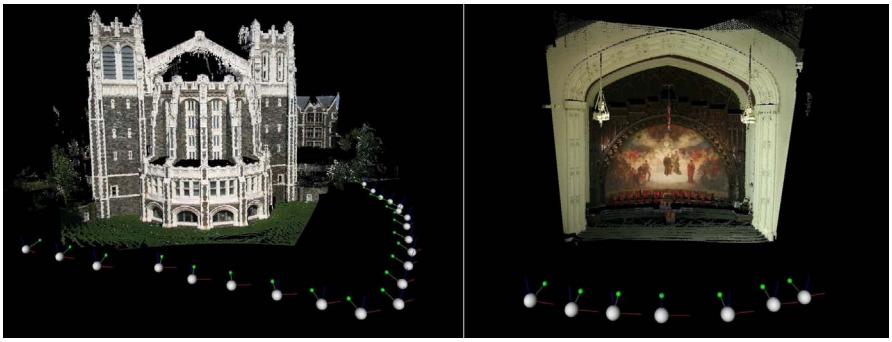


Figure 2.5: Two results for the approach presented in [99].

and cons of several methods. Recently, two exploitations of MI have been proposed

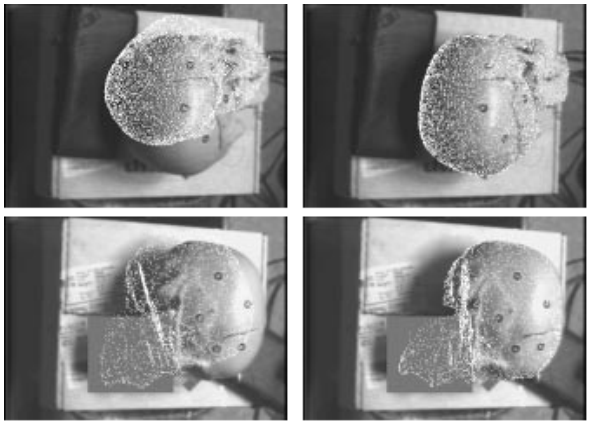


Figure 2.6: Registration via MI presented in [140].

for non-medical applications: 3D object tracking for simple template-based objects [115], and image registration improvement [45].

2.1.2 Color projection and visualization

Once that the images have been aligned to the 3D model, it is possible to project each pixel of them on the geometry. However, a major issue arises: each part of the geometry can be framed by different images and, due to different illumination, small misalignments or image artifacts, the color values associated to them can be

different. Given this problem, what is the best solution to project and visualize color information?
Various approaches have been proposed for selecting the most correct color which has to be applied to each part of the 3D model. The standard approach is to compute a texture by assembling subparts of the original input images, but an alternative solution can be to use color-per-vertex encoding. Advantages and disadvantages of these approaches will be discussed in depth in Section 3.2
However, texture mapping is currently the most used solution. The general idea is

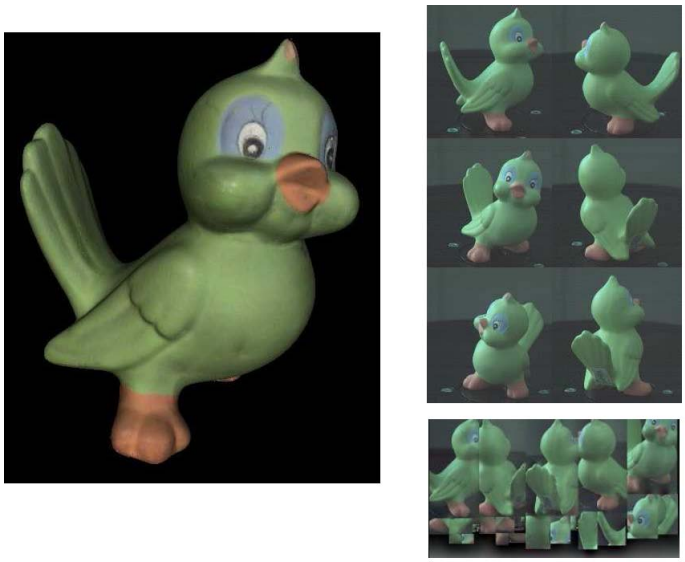


Figure 2.7: Texture mapped bird obtained with the approach in [33]: on the left is the texture-mapped 3D mesh, on the top-right are the six photos used to construct the final texture (bottom right).

to select some portions of the images, by choosing the ones which best depict the model in each part of the geometry. Additionally, some corrections can be applied to deal with incoherence in the borders between different images. An example of this approach is in Callieri et al. [33], where the mesh is covered using the most orthogonal image for each mesh portion, redundancy is used to correct color discontinuities in the boundary between images and then the correction is propagated to the whole texture space (Figure 2.7). Similarly, Bannai [9] used the redundancy to perform a matrix-based color correction on the original image. Camera orthogonality is used also in Lensch et al. [91] to choose which part of the 3D model is to be mapped in which photo; the images are then fully blended, using the entire redundant area

Figure 2.8: Image correction in [91]: left, the original image; right, the corrected image for texture mapping. Image redundancy is used to eliminate artifacts like highlights and shadows.

(Figure 2.8). Conversely, other approaches generate color mapping without reassembling the original images; an example is in Yu et al. [143], where the texture is filled directly texel by texel with values coming from an inverse rendering process based on the original images. Instead of cutting and pasting parts of the original images, as the previous approaches have done, it is possible to assign a weight (that express the ”quality” of the contribution) to each input pixel , and select the final color of the surface as the weighted mean of the input data, as in [123]. Both Bernardini et al.[20] and Baumberg [15] use a weighted blending function to compute the texture color, but without exploiting all the potentiality of this method. Per-pixel weighted blending has also been used in image processing, as shown in Rankov et al. [126].
However, the improvements in acquisition device, graphics hardware and multi resolution data visualization generated the need to be able not only to create high quality results, but also to handle very large amount of data, like hundreds of images and dense geometric data-sets. This brings to the necessity not only to create more effective methods, but also to find robust out-of-core implementations.

2.1.3 Lights and materials

In order to obtain a completely realistic rendering of an object, the goal should be to effectively figure out how the surface reacts to incident light, obtaining the true optical properties of the surface material; in this way it will be afterward possible

to recreate every possible illumination. What is needed is a way to represent how much of the incident light is reflected in each direction; this is called Bi-directional Reflection Distribution Function (BRDF).

A BRDF model is a reflection function that attempts to model the observed scattering

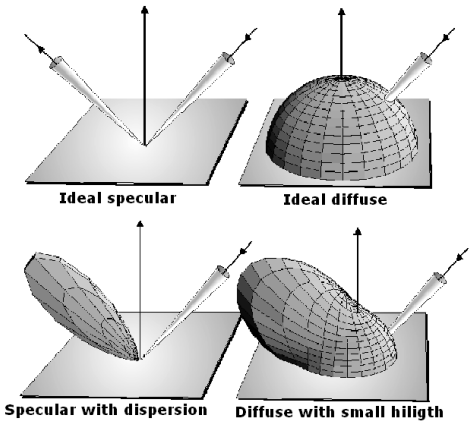


Figure 2.9: Top row: ideal light-matter interaction, the light is totally reflected toward specular direction or diffuse equally in every direction. Bottom row: real world interaction, quasi-specular surfaces presents some scattering from the specular direction (it is often addressed as material Hardness) and quasi-diffuse presents some small highlights too.

behavior of a class of real surfaces, usually in a physically motivated manner [56]. Most BRDF models have a fixed number of parameters. Each parameter has some physical meaning, usually related to some characteristic of the class of surfaces to which the model applies. With most models it is possible to choose parameter values that break the physical plausibility constraints. Some ideal and real world reflection models are shown in Figure 2.9.

Most BRDF calculation approaches rely on a controlled illumination setup (e.g. a single point light source [94], Figure 2.10) and a mechanical setup to move the object or the light. With this or a similar laboratory setup it is possible to acquire several images from several viewpoints and under different directions of illumination. In other words, the acquired images (up to several hundreds) are a dense sampling of the real BRDF (non-parametric approach). Another possibility is to use these image data to fit a pre-defined analytic model of the BRDF (parametric approach).

Marschner et al. [105] were among the first to design a laboratory setup that uses hundreds of images taken in a controlled environment for BRDF estimation. They

Figure 2.10: BRDF acquisition using Lensch et al. method [94]: the sampled object is illuminated with a strong quasi-punctual light source, the surface response is captured using a digital camera. Both light and camera position are calculated using the mirroring spheres

used a flash light as controlled light sources, with the object fixed on a roundtable. The effect of the flash was compensated only in the white component using a white calibration target. Debevec et al. [50] built another setup based on controlled image acquisition for the BRDF estimation. The high-quality results of this device have been demonstrated for the acquisition of the reflectance of the skin of human faces. Even in this case thousands of images are required and have to be processed. Lensch et al. [92] proposed a parametric approach that uses images taken in a controlled environment to estimate the parameters of a (1-lobe) Lafortune model. Other examples of parametric approach are the works of Sato [128] and Yu [143] which make the assumption of the specular part of the BRDF being constant over large regions of the object.

An analytic approach that does not require any particular setup has been developed by Ramamoorthi [125]. This method treats the problem of the simultaneous estimation of the BRDF and the illumination of an object starting from a set of images registered on the surface as a deconvolution problem. This theory is developed by representing the BRDF and the lighting environment with spherical harmonics, using their properties to obtain an elegant and theoretically sound method. More recently, Ramamoorthi [124] proposed a non-parametric approach based on an interpolation scheme in the mixed angular-spatial domain that accomplishes with the angular and spatial redundancy of the SBRDF in order to considerably reduce the number of images required for a high-resolution SBRDF estimation.

2.2 Geometry modification using images

The main application of registered sets of images on 3D models is the projection of color information, even though this is not the only use we can think of. In fact, the resolution of almost all digital cameras is much higher than the resolution of any geometry acquisition device. Unfortunately, even if the images are registered to a geometry, it can be quite hard to infer 3D data from them.
Nevertheless, for particular applications or with the aim of other "external helps" (user intervention, projected patterns, controlled illumination, markers) images can be used to extract further data not only about color, but also concerning shape and material properties.
In the next Sections, some approaches which use images for Morphing or Hole Filling will be presented, in order to introduce the fields of application of the works presented in Section 4.

2.2.1 Morphing

Geometry morphing has been quite extensively studied in the last years: the aim is to transform a 3D model into another. Several polygonally-based and volumetric techniques have been presented. While even in 1995 Chen et al [40] examined the theory of extending selected 2D warping techniques into 3D, only a few works about 2D warping applied on geometry have been presented.
The main field of application of these works is quite peculiar but extremely important: human faces modeling, recognition and visualization. Most of the applications in this field tend to simply map one or more images on a very simple model. Nevertheless, some papers present methods to obtain more realistic models by deforming geometric dummies to fit information extracted from images.
The method proposed by Lee [130] starts from a generic head dummy model and deforms it accordingly to the features extracted from one or more pictures of a subject (see Figure 2.11). Image features extraction is semi-automatic and the dummy model is represented with a very low level of detail, but this is a simple example of registered images used for 3D deformation.
The morphing of models to fit images has been extensively explored by Blanz et al for the application to human faces. A morphable model is used to create realistic 3D face models in [25] (Figure 2.12). The rules for morphing are automatically extracted from images. A similar concept is used for recognition in [23]. Images concur in the deformation of a dummy model to fit 3D scans of faces in [24].

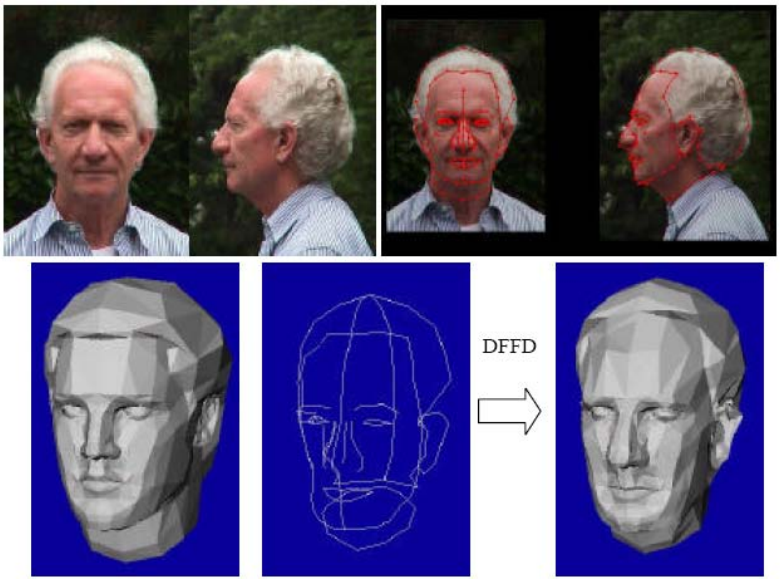


Figure 2.11: The morphing method from [130]: image features are detected and normalized in a semi-automatic way. Then, features are used to deform a head dummy.

2.2.2 Hole filling

One of the most annoying artifacts of scanned data is the frequent presence of holes in the final geometry: this can be consequence of the fact that part of the scene cannot be acquired by the scanner, due to its material or peculiar shape.
Some previous methods [54, 2] made use of images to complete the scanned data, but the result was reached with an hybrid use of photogrammetry and 3D scanning. Hence, in this case it is not correct to say that a set of images was used to correct geometry.

A more recent work by Xu et al [142] proposes an image guided geometry inference method which uses registered images to close holes in a geometry. First the 3D model is processed to compute estimates of the surface normals. An association of surface normals to image patches is then learned. Next, holes in the 3D model are detected and slightly dilated. The remaining geometry is then projected onto each of the aligned scan-view images, and the projected hole regions are detected. Within these hole regions, surface normals are estimated using the image intensity-to-normal mapping function that was learned. Finally, a surface is computed by integrating the estimated normals using the hole boundary as constraints.
Results provided by this method are quite interesting, and generally better than the

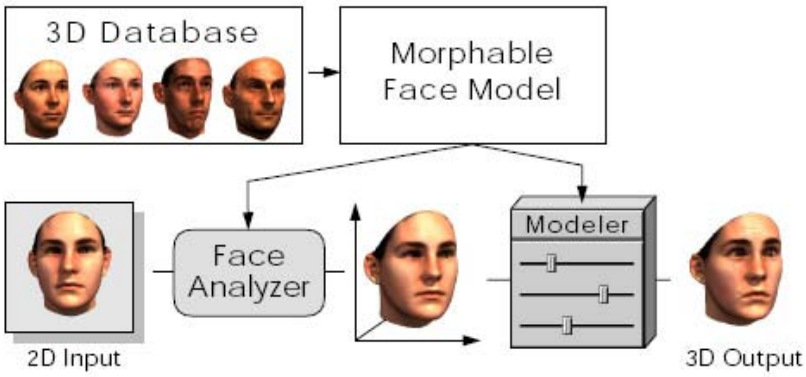


Figure 2.12: The workflow from [25]: 3D dummy models are morphed to fit information extracted from images.

state-of-art hole-filling techniques (based mainly on a volumetric approach). Nevertheless, used images are renderings of the original mesh, not real images. It is still to be shown that enough data can be inferred from images, although this possibility would bring to several possible interesting applications.

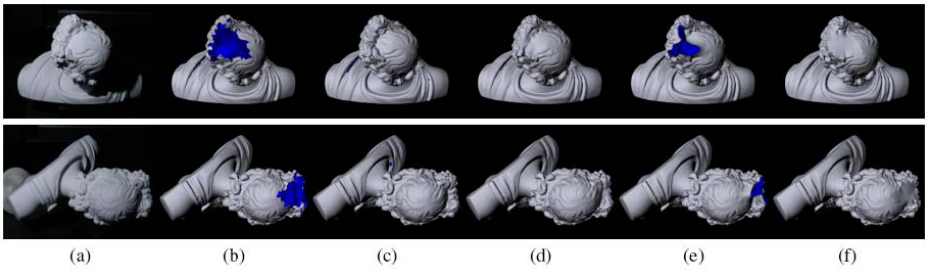


Figure 2.13: Hole Filling experiment with the Apollo statue. (a) Photograph of the statue; (b) the initial model before hole filling; (c) ground-truth model; (d) Xu [142] approach result; (e) result from VolFill [48]; (f) result from Polymender [84].

Chapter 3

Color acquisition and visualization

The so-called ”color” of a surface is a very difficult concept to define: this is simply because the appearance of the same object can greatly vary with the light conditions, and even this variability can be very different from one kind of material to the other. It is essentially true that the concept of color is strictly related to light, but in order to describe the visual peculiarities of the surface of an object we can describe it by trying to separate the information of its ”pure” color with a description of its surface properties. This is especially important in the case of Computer Graphics: once that the color of the object and a model of the reflection function of its material are known, then it should be possible to represent the object with any kind of possible illuminations. Hence, we need to know at least the value of the *unshaded diffuse component* of the surface.

The issue of the analysis of the surface appearance of an object can be divided in two operations: the acquisition and visualization of color information, and the acquisition of the reflectance properties of the surface. The work presented in this section is mainly devoted to the first operation, while a short overview of the work in the field of the second one can be found in Section 2.1.3.

Several devices, like the Colorimeter and the Spectrometer, have been created in order to obtain an accurate measurement of the color of a point: unfortunately the measurement they provide is valid only very locally, the acquisition procedure can be difficult and time-consuming and the hardware is still quite expensive.

Hence, if the goal is to acquire the color of an entire object, a simpler solution must be found: the best choice could be to have the color information together with the geometry, during the scanning. Unfortunately, the quality of color acquired by 3D Scanners is usually quite poor: this is due partly to the usually low resolution of the CCD devices used for 3D acquisition (a Minolta Vivid 910 Triangulation scanner is provided with a 640x480 CCD, which is a very low resolution if the aim is to capture

the color detail). But the main reason for the low quality of color provided by 3D scanners is that these devices are designed to reach maximum quality on the wavelength of the light emitter (e.g. on the wavelength of the laser band), to guarantee high accuracy of the shape reconstruction. Some of them even use a single-band CCD, and adopt RGB filters and multiple shots to acquire color.
On the contrary, the acquisition of color information needs the "pure" color of the object. This means that it has to be framed in a very diffuse illumination environment, that is to say that the light should arrive from all directions with the same intensity. This would prevent from the presence of shadows, highlights and other artifacts, that are not part of the "color" of an object.
Unfortunately, the lighting environment just described is very difficult to obtain, since there is the need of several lights and reflectors in the scene around the object. Hence, it would be very hard to scan an object in such a difficult setup. The only practical solution seems to be to conduce a photographic campaign on the object in a second time. In fact, it is much easier to move around an object with a digital camera instead that with a scanner: moreover, the higher resolution of digital cameras can lead to a complete coverage of the surface with a much lower number of shots respect to a scanning device.
Hence, considering the quality of results, the difficulties in the setup and the needed working time, the best input to project color information on a 3D model is a set of images. However, the projection of color information from a set of uncalibrated images presents several issues; some of them have already been discussed in the previous Chapter.
In this chapter some solutions to these problems will be presented: in Section 3.1 a semi-automatic tool for image registration will be shown in detail. Section 3.2 presents a robust and automatic solution for color projection, while Section 3.3 shows a method to correct image artifacts under a very simple controlled light environment. Finally, Section 3.4 describes several success cases where the above mentioned tools were used to acquire and visualize complex Cultural Heritage objects.

3.1 TexAlign: a user-friendly semi-automatic registration tool

In the previous Chapter, it was shown that there is no automatic approach to register images to a 3D model which can work in a general case. Some fields of applications, like medical data, can present very similar kinds of data, so that a custom approach (like Mutual Information in this peculiar case) can reasonably cover most of the cases.

Unfortunately, other possible applications, like Cultural Heritage, can present very different objects to be acquired, both in size and type of geometry. In fact, both a vase and the façade of a building can be interesting subjects for acquisition and visualization, and even a small vase can present a really simple or very complex geometry. Hence, state-of-the-art techniques like silhouette matching or Mutual Information are not able to provide acceptable results in a general case.
Consequently, if a tool to register images for Cultural Heritage artifacts has to be created, the semi-automatic estimation via correspondences must be implemented. This kind of operation can be tedious and time-consuming. Moreover, the alignment quality of estimated parameters is strictly related to the quality of the correspondences set by the user.
Hence, a good alignment tool should be user-friendly, so that correspondences can be easily found and set. Additional features to help the user, especially in the case of complex projects (tens of images to be aligned on a 3D model) would be very important to create a really useful and usable product.
Following these indications and needs a tool called TexAlign was created [65, 66]. TexAlign was designed with the aim of being able to deal with a wide variety of possible objects (from very small to very large ones). The registration process is organized as a work project, so that tens of images can be handled contemporaneously, and the alignment process can be saved and resumed at any time.
The interface of TexAlign is divided in three spaces: the WorkSpace Tab (Figure 3.1, top) contains all the elements of the registration project (visualized as thumbnails in the lower part of the screen). In this space, the user can set correspondences between all the elements of the project: not only between the 3D model and any image, but also between images. The use of these kind of correspondences will be shown in the next Subsections.
In the Calibration Tab (Figure 3.1, middle) it is possible to launch the alignment for any image which has enough correspondences set to the 3D model. The estimation of camera parameters can be calculated using two methods: the classic Tsai one [133], which needs at least 12 correspondences, and a non-linear method [55] derived from the approach of Faugeras and Toscani [60], which needs at least 6 correspondences and performs optimization on extrinsic parameters and the focal length value. The second approach needs less correspondences, but the final results is influenced by the initial position of the model respect to the image: hence, a preliminary rough alignment is needed.
The calibration algorithms are applied iteratively, in order to converge to the optimal solution: this is especially important if the focal length has to be estimated. In fact, the function assigned to this value has several local minima, due to the fact that the change of zoom or distance of the camera respect to the object results in similar visual

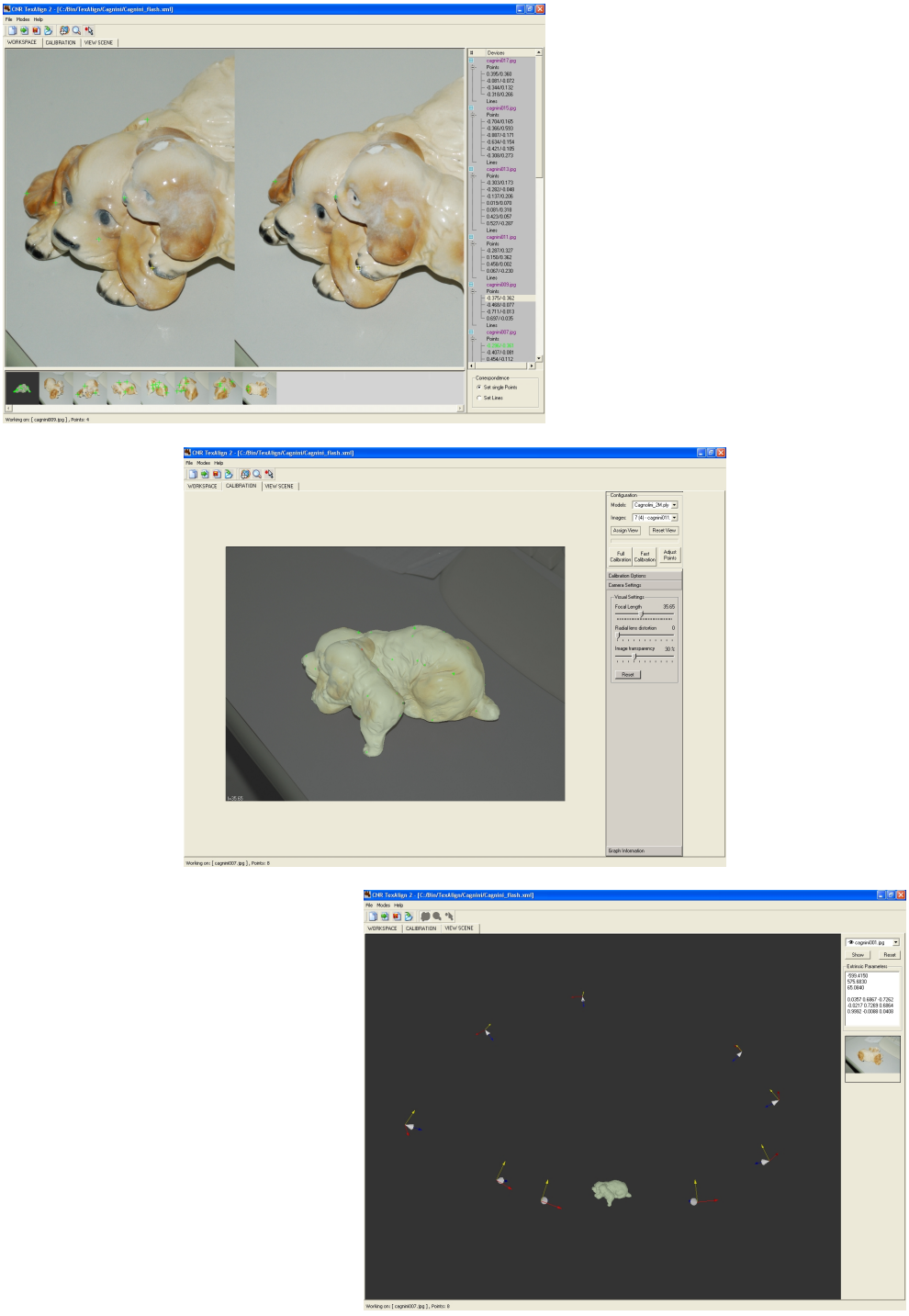


Figure 3.1: The three spaces of TexAlign: top, Workspace Tab; middle, Calibration Tab; bottom, Scene visualization Tab.

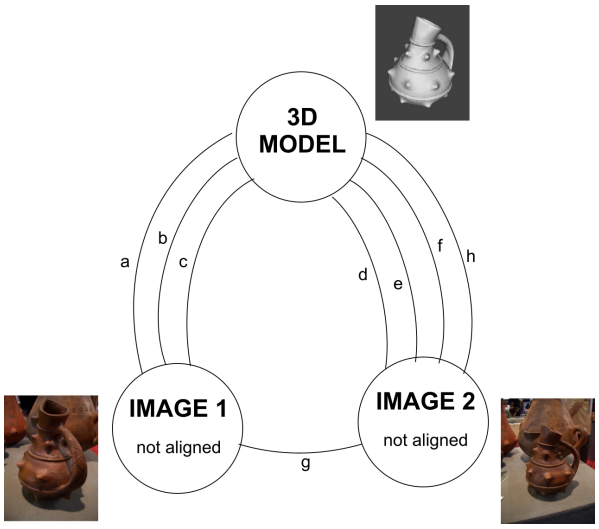


Figure 3.2: A simple example of correspondence graph.

results. In order to prevent the algorithms from stopping in non optimal solutions, the algorithm is iterated to converge to a better solution.

Once that the user is satisfied with the alignment, the associated view can be assigned to the image.

Finally, the Scene View Tab (Figure 3.1, bottom) shows the model and the camera positions associated to each registered image. This visualization space is intended to give the user the possibility to check if the estimated parameters correspond to realistic positions of the camera, and to have a visual feedback about the coverage of surface provided by aligned images.

The registration process can be saved in a .xml file, which can be easily read by any other application which could start from a set of registered images.

3.1.1 The graph of correspondences an its use

While providing user-friendly interface, the main contribution of the work presented in [65] is the use of the graph of correspondences.

A *correspondence graph*, where the 3D model and the images are represented by nodes, is defined as follows. Two nodes are connected by an arc *if* there is a correspondence between the respective entities, of type either I2G (image to geometry) or I2I (image to image). A very simple correspondence graph is shown in the example in Figure 3.2: IMAGE1 is connected with the 3D mesh with three correspondences

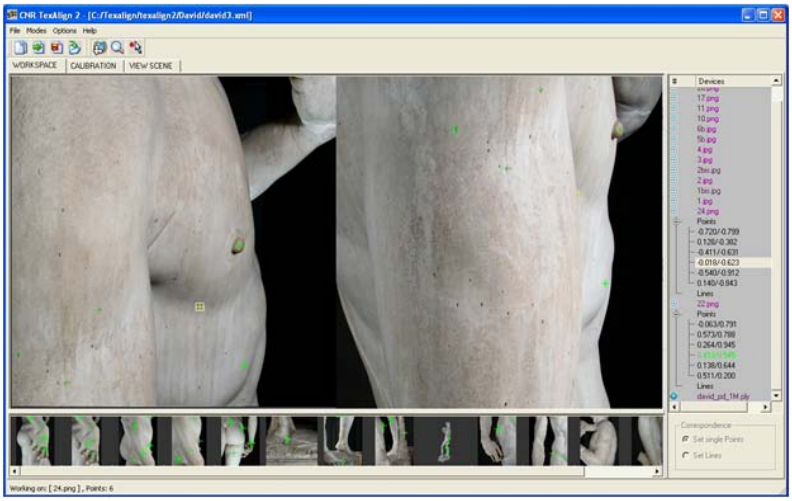


Figure 3.3: An example of registration project where image to image correspondences are easier to find.

(i.e. three corresponding point pairs have been selected); IMAGE2 has four correspondences and IMAGE1 and IMAGE2 are connected by an arc g which is a correspondence between points in the images (a image to image correspondence).

The last arc is one of the main new features of TexAlign: in fact, the user has the possibility not only to set correspondences between any image and the model, but also between two images. The image to geometry correspondences can be set only if there is a set of geometry features that can be linked to a corresponding set of points in 2D. In several cases, like the one shown in Figure 3.3, large parts of an image represent sections of the surface where there are no visible geometric features, hence it could be very hard to find links to the 3D model. But, as shown in Figure 3.3, there can be other images which overlap, and due to the peculiarities of the color information of the object (in this case the small spots on the marble) it is much easier to find correspondences between images.

Nevertheless, how is it possible to use image-to-image correspondences to speed up the registration process? The correspondences graph can help in this, with the automatic inference of new correspondences. In fact, once that an image is aligned to a 3D model, it is possible to associate any pixel (which maps on the geometry) in the image to a corresponding 3D point on the mesh. Hence, if one or more correspondences are set only between two images, if one of the images is aligned to a model, it is possible to automatically set a correspondence between the not yet aligned image and the model. A very simple example is shown in Figure 3.4: once that Image2 is aligned to the model, arc g is automatically transformed in an image-to-geometry

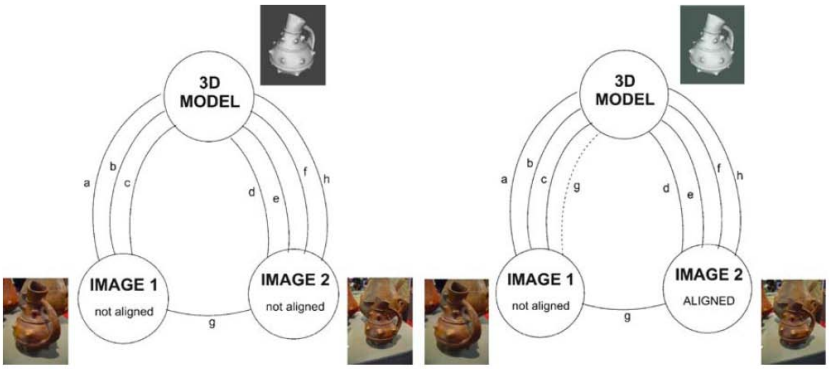


Figure 3.4: An example of automatic inference of correspondences. Left, the graph before image alignment; right, the graph after Image 2 alignment: a correspondence between Image 1 and the model is automatically inferred.

correspondence for Image1.
Besides being an important new feature to help the user in the alignment, the mechanism of the automatic inference brings to a new way to handle a registration project. In fact the user has now the possibility, once that all the elements of the projects are loaded, to first set a number of correspondences between the images, since usually this operation is much easier than setting image-to-geometry correspondences. Then, as images start to be aligned to a model, several new correspondences will be created automatically, and the final workload for the user will be much less than in the case of the classic image-to-geometry only approach.

3.1.2 Processing the graph of correspondences: the workload minimizer

The automatic inference of correspondences is a great help for the user in the registration process, but the possibility of setting correspondences between any couple of elements of the project has a little drawback: the generation of quite complex correspondences graphs. This is not a real issue in terms of data complexity or handling, but especially in the case of big alignment projects, with tens of images involved, the user could be unable to keep track of the changes in the correspondences graph. This could prevent from choosing the best ”path” (in terms of the order of alignment of images) to minimize the number of new needed correspondences.
A very simple example is shown in Figure 3.5: starting from a very simple graph (a) and supposing that we need at least five correspondences to align an image, if we align the images which need the least number of correspondences, the full registra-

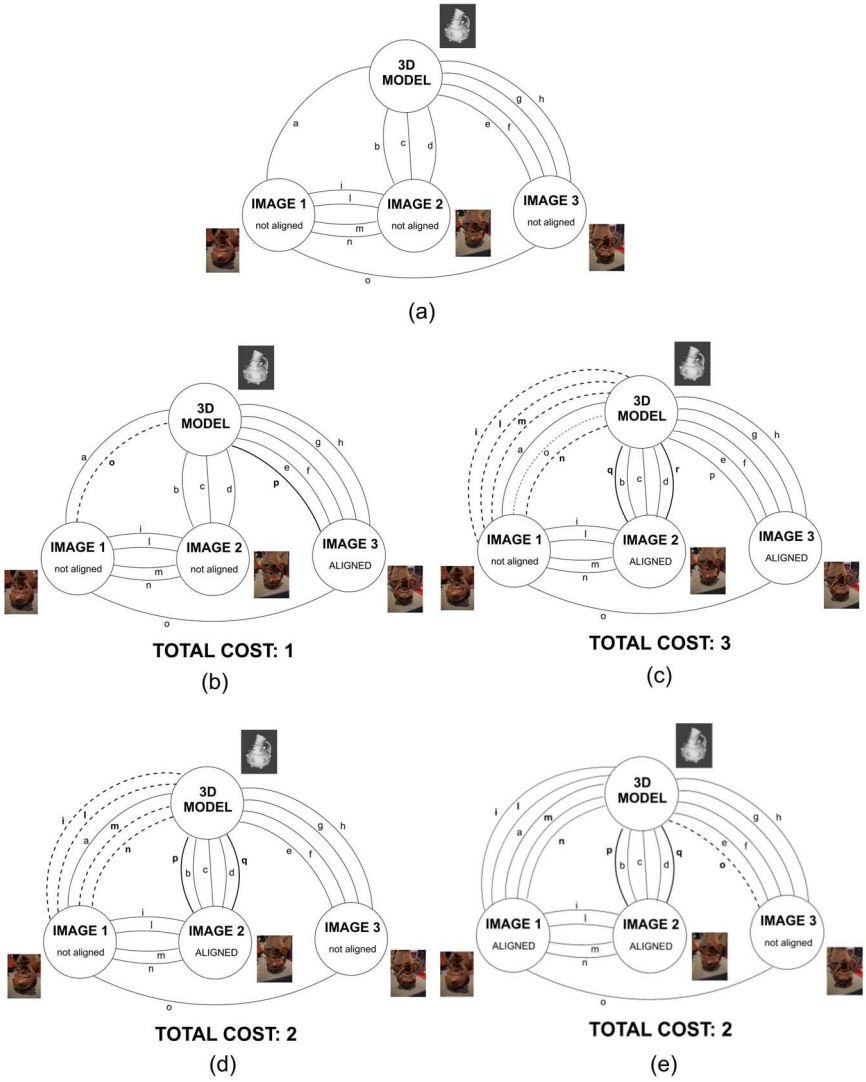


Figure 3.5: A simple example where registration cost can vary respect to the order of alignment of images. (a): the original graph. (b) and (c): graph update obtained by aligning first IMAGE3 and then IMAGE2. (d) and (e): graph update obtained by aligning first IMAGE2 and then IMAGE1.

tion is obtained by setting 3 new correspondences (see graphs (b) and (c)). But if we decide to align first Image2, then the final alignment is obtained with only 2 new correspondences (see graphs (d) and (e)).
Clearly, in the case of more complex graphs, it is impossible to follow the "shortest path" in terms of workload without an accurate analysis of the graph.
Following these considerations, with the aim to automatize and make faster the registration process, a mechanism to guide the user was created: it was called *workload minimizer*. The workload minimizer analyzes the correspondences graph, and all the possible "paths" (the order in which the images can be registered), taking into account the automatic inference of new correspondences. Hence, the system can suggest the best strategy in order to minimize the number of correspondences to place manually.
The problem was posed as a *state space search* problem [127]. Note that the correspondence graph encodes the *state* of the alignment, i.e. the set of correspondences that have been placed. If the system is in a state s, and the user places a correspondence, the system moves to the state s'. A goal state, a graph in which all the images are aligned, is a correspondence graph where every image is connected to the model by at least 12 (direct or indirect) corresponding I2G point pairs. More formally, it's possible to define the state space as the quintuple:

$$S = \{N, I, G, A, \sigma\}$$

where N is the set of states, I is the current state when the search is performed, $G \subseteq N$ is the set of goal states, A is the set of actions (in this case the singleton $\{$*place_a_correspondence*$\}$) and $\sigma : N \times A \rightarrow N$ is the set of transactions.
An exhaustive search on this space is prohibitive, since the branching factor would be $n!/2$, with n nodes. In fact, given a correspondence graph, the user can place a correspondence between any pair of nodes. Therefore, A was defined as the action of aligning an image, i.e. of placing all the correspondences necessary to align an image. In this way the branching factor became n, even if the optimum in terms of number of correspondences placed is not guaranteed anymore, since only a subset of the state space must be visited.
The Best First approach was used: starting from the current state, all the actions that can be performed are evaluated with a *heuristic function* and the corresponding states are put in a priority queue. The algorithm ends when a goal state is found and the corresponding path is reported.
A simple example is shown in Figure 3.6, where the shown State graph corresponds to the graph of Figure 3.5(a). The number of needed correspondences to obtain alignment is set to five, and the values associated to each arc are respectively the number of the aligned image and (between parenthesis) the number of new correspondences to be set to reach alignment. The shortest path (bottom line of Figure 3.5) is shown

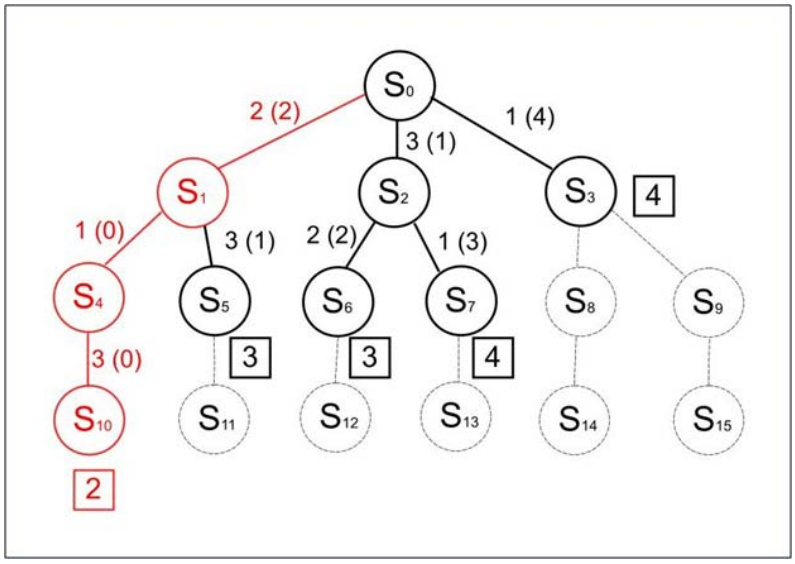


Figure 3.6: The state graph associated to the graph example in Figure 3.5(a): the shortest path is shown in red, the bold states are the ones explored during path search. The number of needed correspondences to obtain alignment is set to five.

in red, and the effectively explored states are shown in bold.

As it can be noted, only half of the possible States are explored by the analysis: hence, the workload minimizer can work in acceptable times even in the case of very complex correspondences graphs. Figure 3.7 shows the workload minimizer in the framework of the tool: once that an image is aligned to the model, TexAlign suggests the user the next image to be aligned, indicating how many correspondences are needed to be able to try to estimate parameters.

Additionally, the tool suggests if there are already aligned images which can be used to infer new correspondences as well. The name of these images are found by analyzing the correspondences graph, and choosing eventual aligned images which already have image-to-image correspondences with the candidate for alignment.

The main drawbacks of the current implementation of the minimizer are related to the hypotheses (it is assumed that the same number of correspondences is needed for every image, but it's clear that some images need more correspondences than others) and to the fact that the only analyzed data are the ones provided by the user. New features to find links between elements of the registration project would make the whole mechanism more robust and effective.

Nevertheless, even in the case of simple alignment projects, the use of image-to-image correspondences proved to cause important improvements in usability and

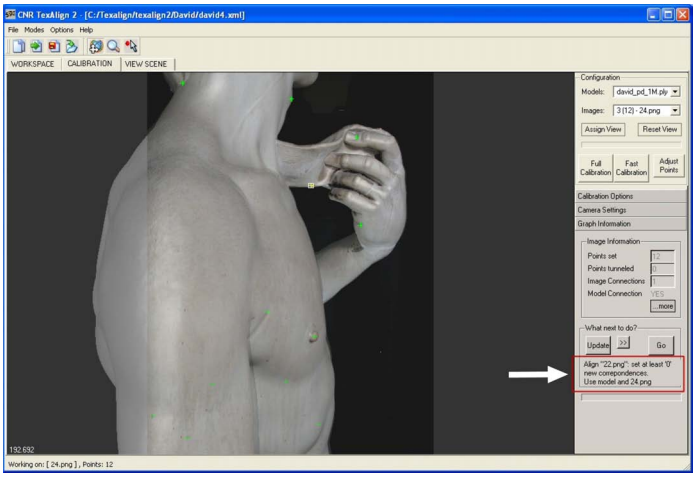


Figure 3.7: screenshot of TexAlign with workload minimizer (red box).

	Experienced user		**Unexperienced user**	
	I2G only	**I2G and I2I**	**I2G only**	**I2G and I2I**
Completion time	∼35 min	∼28 min	∼50 min	∼35 min
No. of correspondences selected	42	33	51	38
Avg. no. of correspondences for each image (min-max)	9 (8-11)	11 (8-13)	11 (7-14)	12 (9-13)

Table 3.1: Results of first test

completion speed (see next Subsection).

3.1.3 Usability tests and results

Some usability tests were performed, in order to have some indications about the usefulness of image-to-image correspondences and workload minimizer.

The sample dataset, shown in Figure 3.8, consisted of a 3D model (nearly 500K faces) of a painted ceramic dish and a set of 8 pictures, taken directly by the RGB unit of the scanner (Konica Minolta VI910). The pictures presented quite big overlapping areas and each one covered a small section of the dish. Moreover, the very simple geometry of the plate made the registration of pictures quite challenging, due to the difficulty in finding relevant surface features.

In the first test, it was asked to two subjects to perform a complete registration of the images using our system. The first subject had already used the application

	Experienced user		Unexperienced user	
	Without minimizer	**With minimizer**	**Without minimizer**	**With minimizer**
Completion time	∼23 min	∼12 min	∼32 min	∼17 min
Total cost (no. of new correspondences)	22	17	25	18

Table 3.2: Results of second test

Figure 3.8: The dataset used for the tests (3D model and all eight images), representing a ceramic dish.

previously, hence he was tagged as an "experienced" one. The other subject tried the application for the first time directly in the test. The subjects performed two registrations of the same dataset (see Figure 3.8), the first time using only I2G correspondences, the second time using I2G and I2I correspondences. A comparison of results is shown in Table 3.1. Experienced user had an improvement of nearly 7 minutes in registration time, and, by the end of registration, the number of explicit correspondences set from images to 3D model showed a 25% reduction, even if the medium number of correspondences defined for each image improved, due to the new I2G correspondences inferred from the I2I ones. Unexperienced user had an improvement of nearly 15 minutes (partially due to the improved skill gained while using the system), reduced by 13 the number of selected correspondences and obtained an improvement in the number of total correspondences (explicit and implicit) for each image as well. This very simple test showed that the use of graph correspondences can be very helpful for the user, reducing completion time and improving the registration quality.

The second test was performed to analyze the usefulness of the graph-based workload minimizer. The same users of first test were given an "intermediate" state of registration on the same *dish* dataset, where some I2G and I2I correspondences were already set, and two out of the eight images were already aligned to the geometry. Users had to complete registration with and without the use of the advices proposed by the workload minimizer. The minimizer estimated a minimum number of 15 new correspondences needed. In Table 3.2 we present a comparison of results. Without the help of minimizer, the experienced user took nearly 23 minutes to complete the registration, setting 22 new correspondences. Using the minimizer, the completion time was almost halved, with only 17 new correspondences (for two images the alignment became satisfactory with one more connection than the ones indicated by minimizer). The second user produced similar results, using approximately one half of the original time.
While interesting, these preliminary tests were widely confirmed by the massive use of TexAlign in the context of several other practical cases, some of which will be presented in Section 3.4. TexAlign (with its peculiar features and its easy interface) proved to be an extremely useful and reliable tool for image registration.

3.2 TexTailor: automatic color projection of registered sets of images on 3D meshes

Once that a set of calibrated images and a 3D model are at disposal, it is possible to project the color information on it. Unfortunately, other issues arise. There are

two main problems which have to be faced in order to project and visualize color on geometry.
The first one is related to visualization: there are two possible encoding approaches, each of which presents advantages and disadvantages. The first one is the texture mapping, the second one is the color per vertex.
The texture mapping [37] principle is very simple; the color of the 3D surface is stored in an image (the texture), when the object is rendered, for each surface point the texture image is accessed to retrieve the object color in that point. The mapping relies in the existence of a parametrization of the 3D model; it is necessary to have a function that maps the object coordinates (that are in a 3D space) onto the texture image pixels (2D space).
Color per vertex associates a single color value to each vertex of the geometry: this structure is much more simple and compact, while the detail of the color is strictly related to the density of geometry samples.
Texture mapping is a widely used approach and several methods have been proposed [16, 129, 28, 110, 93, 33, 27] to build up a parametrization of the 3D mesh that fits well the pool of images available, and to produce a new texture map, either by joining subregions of the input images or by resampling.
Unfortunately, the management of very dense geometric and photographic sampling is very complicated. The texture-based approach is ideal when we have both low-to-moderate resolution meshes (50K-1M faces), usually produced by simplification or subsampling, and moderate pixel datasets (1M-5M). Moreover, multiresolution encoding is usually a must for huge meshes, and the adoption of multiresolution approach for texture-based representation of color [27] implies the need of a multi-resolution texture atlas, with the associated redundancy and increased space occupancy.
On the base of these remarks, color-per-vertex seems to be much more usable in the case of complex (very big and dense 3D models, tens of images to be mapped) cases, like most of the ones shown in Section 3.4.
The other main issue related to projection of color is the policy of color assignment. In fact, as shown in Figure 3.9, a single point in 3D can be framed by several aligned images. The pixel color value associated to each framing image can be different, due to different lighting condition or small misalignments: which is the color to be assigned to the 3D point?
It is clear that a simple mean of the values tends to disadvantage the higher quality images, reducing the quality of color information. Hence an automatic method, able to recognize higher quality images and assign the best combination of the contribution of all images is needed.
The work of Visual Computing Lab produced a tool called TexTailor, which is based

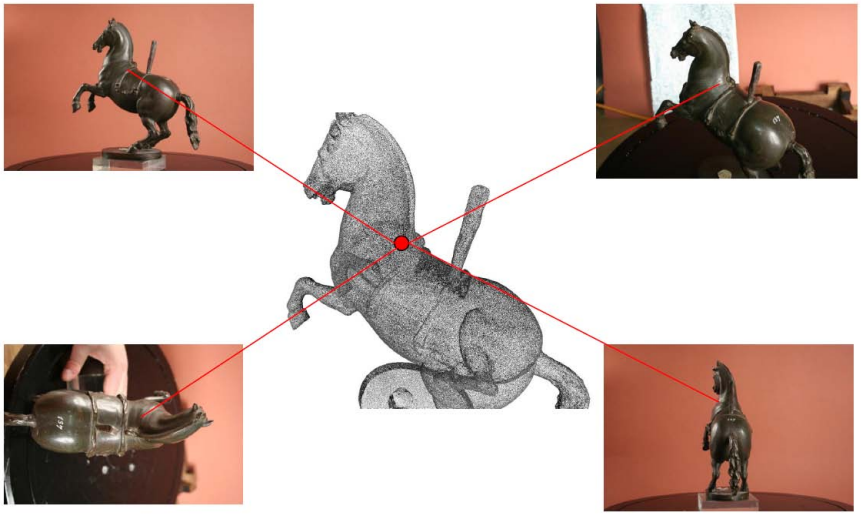


Figure 3.9: An example of a point in 3D on which several images map.

on a mechanism of a blending function [34] which is able to handle high amounts of data and produce extremely realistic results. The multivariate blending function is proposed to weight all the available pixel data w.r.t. geometric, topological and colorimetric criteria. In this way the aim is to maximize the use of the information contained in the input images and to correct incoherence between different photos, while at the same time trying to reduce the blur caused by simple blending approaches.

3.2.1 The blending function and the use of weights

The objective of the blending function is to be able to work with photos taken in almost arbitrary conditions since, especially when working in the Cultural Heritage field, having a calibrated de-shaded and artifact-free photo dataset is not always a viable option. To acquire models of Cultural Heritage artifacts it is often necessary to work inside museums or even outdoor, hence it is almost impossible to setup controlled lighting conditions.

The first need is to detect those pixels in each photo which sample the object surface (since some pixels could depict the background) and, for each of those, which are the coordinates of the surface point which are sampled. The projective mapping principle is well known. Since the photos follow the laws of perspective, if the camera parameters are known, then it's possible to determine if (and where) a point on the

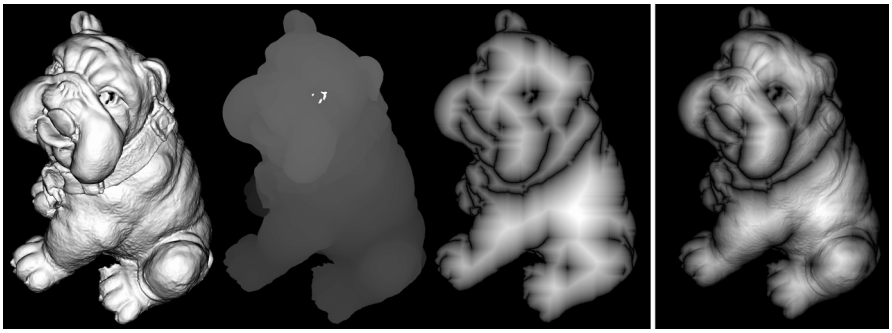


Figure 3.10: An example of the core weighting masks in [34]. From left to right: Angle Mask, Depth Mask, Border Mask. Rightmost, all the masks combined in the final mask. Caveat: the contrast of the depth and border masks has been increased for enhanced readability.

surface is mapped inside the image boundaries. An efficient manner to compute those pixel-to-surface correspondences is by rendering the 3D model with the same projection parameters of the given photo. In this way the resulting depth map could be used to discriminate, similarly to shadow mapping, if a point on the surface that projected inside the image is effectively visible from that point of view, or if it is occluded by other geometry. In this way it is possible to assign safely a pixel color, taken from that photo, to a point onto the surface.

Now that the geometric position where each pixel mapped is known, it is necessary to define a policy to specify which pixel had to be mapped in case of ambiguities. Since each pixel in the source images has a specific quality (that can be evaluated adopting various metrics), it is necessary to take into account this quality while blending. The approach is therefore to define an extensible masked blending approach, where multiple local image evaluation heuristics could be defined to weight each single pixel. Once a suitable set of weighting masks is defined, it is possible to assign the most correct color to each point of the surface as a weighted mean of all possible sources. The problem is how to choose those local quality evaluation heuristics for our weighting method: they should be local and efficient both in time and space. The core of the blending function is the *weighting mask* that is generated for each image. The weight mask states the quality of each pixel and, consequently, how much it will contribute to the final color of the 3D points it maps on.

Various metrics can be applied to evaluate the quality of image pixels. Since not only the images but also a faithful geometric representation of the object were at disposal, this information could be used to perform a higher quality evaluation. Each metric could take into account a different characteristic of the source image.

For example, pixel quality could be measured based on camera orthogonality to the corresponding surface point, because orthogonal views appear less "stretched". But a pixel that is "good" for this particular metric could at the same time be considered "bad" by another metric that considers the distance from the sensor (the farther the surface, the less dense is the sampling available in the image). Following this approach, multiple metrics were defined and mixed in a single measure. Following this strategy, it was possible to evaluate and mix an arbitrary number of metrics, making the system flexible and extensible.

Since a per-pixel blending was needed, all the masks had exactly the same resolution of the corresponding photo image. Once the camera parameters were known, it was possible to calculate each metric with a series of controlled renderings and video buffer processing. The final basic weights masks chosen for the system are:

Angle mask: this is the simplest quality evaluation for each image pixel, proportional to the cosine of the angle between the viewing ray and the surface normal. Similar to Lambertian illumination, the quality is higher when the view direction is orthogonal to the surface, and lower when glazing.

Depth mask: the weight of an image pixel is higher as the surface is closer to the camera. This is an approximation of the ratio pixel/surface (informally, how dense is the sampling encoded on this image part). Obviously this ratio depends also on view orientation, but this has already been taken in account when calculating the Angle mask.

Border mask: this mask measures how far a pixel in the image is from both image borders and discontinuities in the depth map (silhouette borders). The farther we are from these borders, the higher the quality of the photo sampling. To compute this mask, the first step is to mark as point of zero value both the image borders and the depth discontinuities. The latter can be calculated by simply using a Sobel filter on the non-normalized depth map (every pixel contains the actual distance from the sensor, not the normalized [0-1] value): using such wider range makes the Sobel filter much more precise. For the rest of the pixels, the weight is calculated as the image-space (pixel-to-pixel) distance from the borders on the image space.

The composed weighting mask is generated by assembling various metrics through multiplication. In this way the continuity and minima of each mask are maintained. Preservation of minima is very important to remove outliers. An example of the core masks described above and the final assembled weight mask is shown in Figure 3.10.

The most interesting feature of this approach is its flexibility; the system works by evaluating multiple metrics and then mixing them in a single quality measure. In order to improve our system, more metrics can be added to these *core masks*, to address for specific dataset problems. Two examples are:

Stencil mask: in some cases there is the need to exclude certain parts of the source image (e.g., people in front of a building, photographic artifacts). In those cases a simple stencil masking can be applied. Images to correct are (manually) marked with a specific color; the system will then assign zero weight to those pixels. In alternative, for each photo that requires a stencil mask, another grey-scale image that basically contains the 0-1 mask that has to be assigned. Border of the stencil mask are also considered in the computation of the Border Mask, in order to avoid discontinuities between stenciled regions.

Focus Mask: to build this mask the focusing of each pixel of the image is evaluated in a way that is almost identical to the procedure used by digital cameras while performing the autofocus selection. Typically, the autofocus procedure set the focus distance by maximizing the sharpness of some reference points in the framed image. Therefore, the per-pixel "focusness" of the source image is evaluated using a sharpness operator that is the energy of the image Laplacian [132], applied on a window of 20×20 pixels centered on each evaluated pixel.

Figure 3.11 shows an example of focus masking; in this case the focus problem was due to excessive depth of field. With this masking the out of focus areas are assigned a very low weight. In this way less blurring is introduced when blended with other images, but it can be available as color source if no other data can be mapped on the same surface region. Often for small objects, without laboratory lighting conditions, it is almost impossible to take fully focused photos, so multiple images from the same viewpoint are required with different focus settings. Without this kind of masking it would be impossible to reconstruct a sharp color information.

Once that all the masks are generated, given a point on the 3D surface, it is easy to determine the set of source images that effectively "see" that surface point and the corresponding pixel coordinates. These source colors will be multiplied by the weights contained in the image masks and accumulated, assigning the weighted mean as the final color of the 3D point.

A very nice characteristic of this function is its generality: given a point on the surface, a color is returned. It is then possible to use it to fill a texture map or, with a

Figure 3.11: Focus masking example. An image of the wooden statue with large depth extent. As detected by the focus mask,the hands are out of focus due to depth of field, while the red part of the cloth is perfectly focused.

simpler approach, to compute per-vertex colors on a 3D model. Moreover, the masks are *independent* from the resolution of the mapping target; after the weighting masks have been calculated once, it is possible to use them to apply the color on different level of resolution of the 3D object.

If a parametrization of the 3D model is available, it is possible to use the weighting function to compute the color of each texel in the texture map. Since there is a full correspondence between texels and points on the 3D surface, it is necessary to calculate the function for each texel. The generation of a suitable parametrization for an arbitrary model is not in the scope of this theses and, especially when dealing with very large or topologically unclean models, it is often a very hard task. The images shown in this Section and the next ones are all obtained using per-vertex color. This can be a very good choice especially in the case of 3D scanned models, where the density of samples is enough to be able to get rid of the parametrization.

Moreover, using all the redundancy contained in the image dataset, it is possible to obtain a high quality color mapping. Even if blending multiple images, high frequency details are not lost thanks to the weighting function. Image areas with more detail will have a higher weight, that will dominate over other images either blurred or with a lower sampling rate. At the same time, discrepancy between different photos is not detectable anymore, since blending the overlap area between different images will mask the discontinuities (see Figure 3.12).

Since the color blending function relies on redundancy to determine the most correct color to apply, a good level of overlapping is essential to overcome strong color discrepancies due to illumination changes or color biasing between different images. Having all points of the surface covered by at least three images can be enough to correct most of these problems. This may appear as a strong requirement, but such a dense sampling of the surface is common in 3D scanning. When sufficient overlap is not available, or some images present a poor color coherence with the others, some artifacts in the resulted mapping can be perceived.

3.2.2 Large dataset management and Results

Executing the proposed pipeline for color reconstruction from photos could require excessive memory resources on very large datasets.

The problem of dealing with large dataset depends on the size of both the 3D geometry and the photographic data. When an “all in memory” approach is adopted, sufficient RAM for storing the 3D geometry, each input photo with the associated depth map and weight mask (both float vector, same size of the image) is needed.

Figure 3.12: An example of color mapping. The 3D model is composed by 1 million triangles, with a mapped photographic dataset of 8 images 2560×1920. From left to right: original photos, geometry, color + lambertian shading, color only.

The memory capacity can be easily exceeded. It is also impractical to set up a system which trivially loads and discards the images and the maps upon request as a whole, since, by default, there is not much locality and the size of these "chunks" is quite large. It is much more efficient to compute and organize the data in a way that is easy to access on disk and to adopt Out-Of-Core (OOC) strategies.

Designing the system, two different cases of OOC management were distinguished: the most common case is when the 3D model fits in memory, but the image and mask dataset have to be managed out-of core. In the second case, also the 3D model is too large to be kept in memory, requiring the interaction of two OOC systems.

The mask computation can be done independently for each image, since it is a local processing; this greatly helps the OOC management, since it is possible to calculate one mask at a time, keeping the minimum data in main memory. Computing the mask is easy if the 3D model fits in memory, but it is also viable if the 3D model is kept OOC. Since only controlled renderings are used, it is just necessary that the used OOC structure for the 3D model supports rendering. In this way it is possible to generate all the data necessary to apply the blending function.
Moreover, the masking data for each photo can be stored in a single image-buffer file with a tight packing: for each pixel the color values (from the image), the depth value (for the shadow-mapping projection) and the mask value (for blending) are

stored. Thanks to these two properties (independent calculation and compact storage), building the OOC scheme becomes easier.

Organizing the mapping phase is straightforward when the model is in main memory. One image-buffer at a time can be loaded from disk, then for each mesh vertex which maps to this photo the associated color contribution, and the corresponding weight value according to the image mask are computed, and finally the color and weight are accumulated into the vertex data structure. After all images have been processed, the required mean weighted color is obtained by iterating again on each vertex and computing the mean of all the weighted contributions. This strategy, with a dataset of N images requires N+1 visits to all mesh vertices (one for each image, plus one for mean color calculation), but it minimizes memory-disk swapping. The same strategy of processing one image at a time can be applied in the case of texture-mapped color encoded to synthesize the new texture.

Conversely, when the 3D model is too large to fit in RAM and has to be managed using OOC structures, the mapping phase is a bit more tricky. Using a multiresolution data structure like the one presented in [42], it is quite difficult to accumulate color and weights by processing one image at a time, since for each image there is the need to traverse the mesh vertices in a random order; traversing the multiresolution data many times increases the processing time and the lack of supporting structures for accumulation is another issue. For this reason, it was created a color mapping strategy where the multiresolution data structure is traversed just once. For each vertex in the multiresolution mesh, the images which "see" this vertex are detected (by just using the inverse camera transformation); for each of these images, the color contribution and weight corresponding to the vertex are retrieved by directly accessing the various image-buffer files using memory mapping system calls. The overall reconstruction time is, in this case, increased wrt. the all-in-memory case, but in a sustainable manner.

To evaluate the effectiveness of our approach various datasets were tested. All 3D models were produced with 3D scanning technology, therefore the meshes were not watertight nor topologically clean. The images included in this section show some of the models obtained with per-vertex color. Table 3.3 lists some processed objects with details on their size, number and resolution of the source images used and time required for performing color mapping. The examples range from a small object with a relatively low geometric and photographic complexity like the dog of Figure 3.12 to a complex, high resolution model of a building with a very dense photographic coverage, such as the Apse of Figure 3.14.

In all dataset, the input photos were aligned to the 3D geometry by using the image registration tool TexAlign. All datasets were processed using at least three core masks. The focus mask was used on the painted wooden statue (as some photos

Figure 3.13: Life-size painted wooden statue (5 million triangles, photographic dataset 33 images 1694×2496). Left: shaded+color, center: color only, right: a more detailed view.

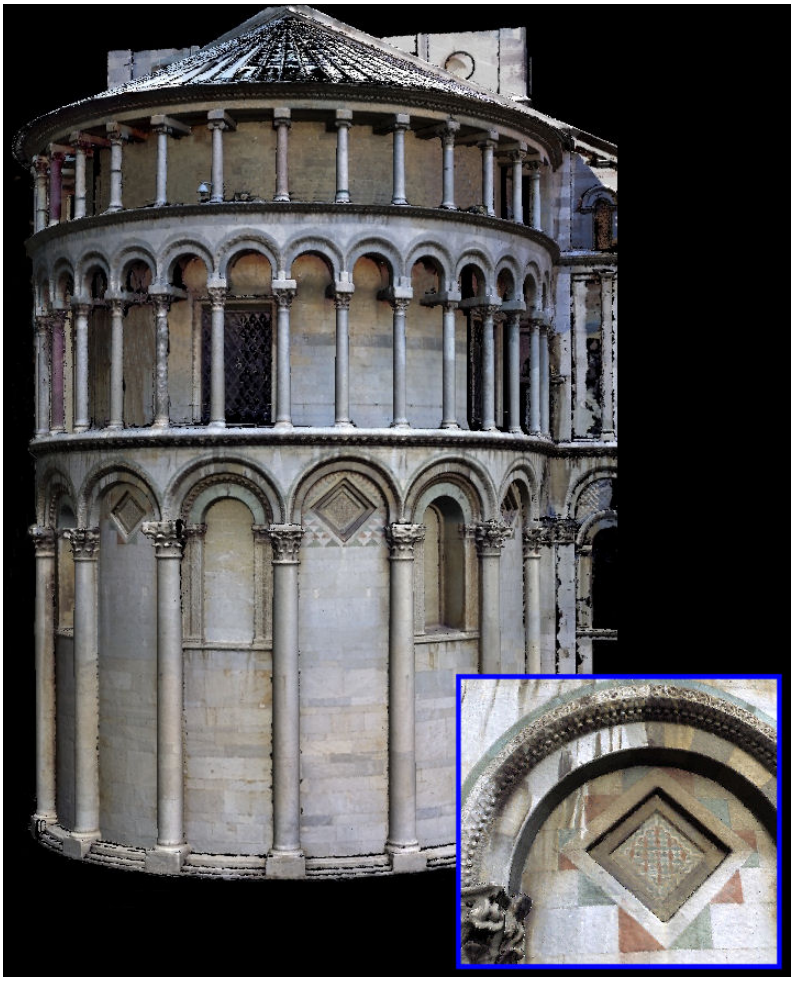


Figure 3.14: Apse of the Pisa Cathedral. A multiresolution model (46 million triangles) with color applied from 36 4000×4000 images.

Object	Res. (# tri)	Images	M pixel	Time
Dog statuette	1 M	8 (2560×1920)	39	3 min
Marble capital	6 M	8 (1800×1200)	17	8 min
Painted statue	5 M	33 (1694×2496)	138	15 min
David, single resolution (pre-restoration)	15 M	60 (1920×2560)	294	3.0 hr
David, single resolution (post-restoration)	15 M	64 (2336×3504)	523	3.5 hr
David, multiresolution (pre-restoration)	56 M	60 (1920×2560)	294	13.0 hr
David, multiresolution (post-restoration)	56 M	64 (2336×3504)	523	15.5 hr
Apse, multiresolution	46 M	36 (4000×4000)	576	10.5 hr

Table 3.3: Specifications of the color mapped objects (note that for multiresolution models, the number of triangles indicated is the size of the model extracted at full resolution and not the number of triangles in the entire multiresolution data structure, which usually is around six times the former).

had out-of-focus areas). Stencil mask was used to process the Apse dataset, to cull out part of the scaffolding and other extraneous elements (like pigeons and part of the scaffolding used to shot the photos).

Additionally, to demonstrate the possibility to manage a very large dataset, a test case of an extraordinary size was selected. Two complete photographic campaigns depicting the status of the statue before and after the restoration were mapped on the 3D model of Michelangelo's David, for a total of 129 images (see Section 3.4.1 for details). Two different geometric datasets for mapping were used: the first one composed by 15 million and the second composed by 56 million triangles. While in the first case it was possible to keep the geometry in main memory (and perform OOC mapping), in the second case both photographic *and* geometric data were accessed using OOC structures.
The time needed to generate the OOC data necessary for color blending was 30 minutes for the first dataset and 50 minutes for the second one. The size on disk of the temporary data was, respectively, 3.6 GB and 6.2 GB for pre- and post-restoration data. The second phase, color blending and mapping for the low resolution model, took 2.5 hours for both datasets (in this phase, time is almost only dependent on the number of function evaluation performed). The time increased while mapping on the more complex model, since also the geometry was stored OOC; mapping time was 12.5 hours for the first dataset and 14 for the second. Moreover, the increase in mapping time also follows the fact that the mapped model is a multiresolution data structure [42] that contains a continuous LOD representation of the geometry (hence, many more vertices to map).

The results are visible in Figure 3.15. There are still parts without color, but this is caused by a actual lack of photographic coverage and does not depend on the

Figure 3.15: Details of the David with the two mapped datasets. Leftside: before restoration, Rightside: after restoration. (upper part of head in the post restoration phase was not covered by the photographic campaign)

mapping process. These missing areas have been caused by the topology and size of the statue, and by the position of the scaffolding, that changed frequently over the course of restoration, making it difficult both to plan the shots and to reproduce them correctly on site. This is a typical situation when performing acquisition campaign in non-laboratory settings.

In conclusion, TexTailor proved to be a very robust and usable tool for color projection. The use of TexAlign and TexTailor brought to several very interesting practical results in the context of heterogeneous projects and kinds of objects (see Section 3.4). the main limitations to this kind of approach are related to the quality of input data: in case of misalignments, lack of coverage o uneven alighting between images, the final result can be not completely satisfying.

Figure 3.16: Two snapshots of colored 3D models, showing annoying artifacts like highlights and shadows projected as color information. Left, the model of a bronze horse, where projected highlights are clearly visible; right, a detail of a stone capital, where shadows effects are present even without lighting.

3.3 Flash lighting space sampling: an easy-to-use controlled lighting setup

The quality of the image set to project on a 3D model is extremely important with respect to the final result: in fact, if the starting images are low quality, or they present very strong artifacts, it is very hard to remove them during the projection phase.

Most of the image artifacts are related to the lighting conditions and to the material of the object. Two examples are shown in the close renderings of Figure 3.16: both snapshots show the models without illumination, which should result in a very ”flat” color. On the contrary, in the left image, the big highlights of the original images were projected on the geometry of a bronze horse. On the right image, some hard shadows where projected in a severely occluded part of a capital. This types of artifacts are frequently present in colored models, since it can be very difficult to remove them by setting a proper illumination environment.

While complex lighting environments are usually difficult to set, it could be possible to create some kind of very simple lighting, which needs no particular setup. All cameras are equipped with a flash: this tool can provide a practical, easy, cheap controlled lighting environment. Unfortunately, built-in flashes usually produce a variety of undesirable effects, like uneven lighting (overexposed in the near field and dark in the distance), highlights and distracting sharp shadows. Moreover, ”white balance” setting for flash is not effective because it applies the same correction throughout the

image. More interesting results could be obtained by knowing the geometry of the scene, and using a model of the behavior of flash light. However, a mathematical modeling of flash light spatial behavior can be hard to be obtained, because of the large variety of camera/flash models, and the irregular spatial light distribution produced by the flash reflectors and lenses.
To solve these problems Flash Lighting Space Sampling (FLiSS) [52], a correction space where a correction matrix is associated to each point in the camera field of view, was proposed. Once that basic information about the geometry of the scene is known, the proposed structure permits to correct each pixel according to the position of the corresponding point in space. This structure proves to be simple and effective, and it has several advantages.
This method shares some aspects with various subjects, such as computational photography, image enhancement and lighting modeling and estimation. We briefly review some of those previous works in the following.

Flash/No-Flash Digital Photography. The use of flash/no-flash pairs to enhance the appearance of photographs is a relatively recent research topic, where several interesting works appeared. The *continuous flash* [77] was a seminal work, where flash and no-flash pairs were combined to create adjustable images. Two almost contemporaneous papers [59, 117] proposed techniques to enhance details and reduce noise in ambient images, by using flash/no-flash pairs. These works provide features for detail transfer, color and noise correction, shadows and highlights removal. While the systems are not completely automatic, very interesting results can be easily obtained. The goal of a more recent work [3] is to enhance flash photography by introducing a flash imaging model and a gradient projection scheme, to reduce the visual effects of noise. Flash/no-flash pairs are used by [102] to detect and remove ambient shadows.

Color constancy and white balance. White balance is a key issue in the context of the *color constancy problem*, that studies the constancy of perceived colors of surfaces under changes in the intensity and spectral composition of the illumination. Several works in this field rely on the assumption that a single illuminant is present: the enhancement of photos can be based on geometric models of color spaces [61], statistical analysis of lights and colors [62] or natural images [69], study of the edges of the image [137].
Another group of papers deals with mixed lighting conditions. Methods can be semi-automatic [97] or automatic. Automatic methods usually work well under quite strong assumptions, like hard shadows and black-body radiators lights [88] or localized gray-world model [58]. A very recent work [79] proposes a white balance technique which renders visually pleasing images by recovering a set of dominant

material colors using the technique proposed by [114]. One of the assumptions is that no more than two light types (specified by the user) illuminate the scene. Most of the cited works share some of the main hypotheses of FLiSS method. Nevertheless, the knowledge of some information about the geometry of the scene eliminates the need for other restricting assumptions (such as smooth illumination, gray-world theory, need of user interaction).

Illumination estimation and light models. The works in the field of illumination estimation have two principal aims: the estimation of the lighting of an environment [49, 135] or the measure of the characteristics of a luminary. Our work is more related to the second group.
One of the first attempts to model both the distant and the near behavior of a light source is the *near-field photometry* approach of Ashdown [7]. Near-field photometry regards the acquisition of a luminary by positioning a number of pinhole cameras (or moving a single camera) around it, and measuring the incident irradiance on a CCD sensor. The results are mapped onto an hemicube that represents the final model of the luminary acquired. Heidrich et al used a similar method [76] by moving the camera on a virtual plane and representing the light sources with a Lumigraph [72]. This representation was named *canned light source*. More recently, Goesele et al. [71] improved the near-field photometric approach using a correction filter to compensate the fact that a digital camera is not a real pinhole camera. FLiSS approach recalls near-field photometry; the main difference is that we estimate the data to "correct" the effect of the light source on known colors, instead of building a model of the light source of interest.

3.3.1 Definition, acquisition and processing of FliSS

The creation of FLiSS corresponds to the building a spatial color correction function that associates a specific color correction procedure to each point in the camera frustum space. This particular data structure was called *color correction space*. Such an approach allows to override the limitations assumed in most of the color correction approaches [13, 14], that is that the illumination is constant, or easily model-able, across the scene. The main assumptions are: flash light can be considered the dominant illumination in the scene; the light interaction can be described using just sRGB space (full spectra data are not accounted); surfaces are non-emitting.
These hypotheses are common among existing techniques, which deal with single illumination, and they cover most of the real cases. Typically, the color calibration of digital photographs consists in taking a snapshot of a pre-defined color reference target, such as a Macbeth ColorChecker™ or an AGFA IT8™, placed near the subject of interest, and estimating the parameters of a transformation that maps the colors of

the reference target in the image into its real colors.
A quite simple approach to model the correction is a linear affine transformation $c' = Ac + B$. Obviously, due to the nonlinear nature of image color formation, this kind of correction is a rough approximation and many other approaches could be used [13, 14]. Moreover, the correction is effective for the image parts that are close (and with a similar illumination) to the reference target. On the other hand, in practice, this simple and compact approach works reasonably well in most cases. The linear transformation can be written as a 4×3 matrix:

$$\begin{pmatrix} R' \\ G' \\ B' \end{pmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \end{bmatrix} \begin{pmatrix} R \\ G \\ B \\ 1 \end{pmatrix} \tag{3.1}$$

In the following this matrix will be called as the color correction matrix $\mathcal{C}$ and its elements the correction parameters.
Roughly speaking, the parameters of $\mathcal{C}$ have the following meaning: (c_{11}, c_{22}, c_{33}) are related to the change in contrast of the color; $(c_{12}, c_{13}, c_{21}, c_{23}, c_{31}, c_{32})$ are related to the color deviation caused by the color of the flash light (if the flash is a purely white light, these components tend to zero); (c_{14}, c_{24}, c_{34}) are related to the intensity offset. The term contrast is used in the sense that the multiplication for the coefficients expands the range of values of the channels.
Given the assumptions above, it is possible to finally define FLiSS. Flash Lighting Space Sampling is a color correction space where a color correction matrix is associated to each point in a camera frustum. The aim of the work was to try to build a procedure which could be used in a general case, without using prototypal or expensive devices. The computation of the color correction space should have been necessary only once (or just few times) in a camera lifetime. But even with this assumption, it was important to define a simple and fast possible acquisition procedure. Hence, it was decided to sample the camera space view frustum by taking flash lighted photos of a small color target, calculating the correction matrix in those points and subsequently building the entire space by interpolation. The acquisition was performed with three different models of digital cameras, shown in Figure 3.17(left). These models were representative of three categories of non-professional cameras: compact, digital SLR (single-lens reflex) and digital SLR with external flash.
As a color target a Mini Macbeth ColorChecker was used; its size (about $3.5'' \times 2.5''$) allows the assumption that the light variation across it is negligible. The view frustum of the camera was sampled by slicing it with several planes of acquisition at different distances, moving the Mini Macbeth in different positions for each plane. A distance range between 50 and 220 cm in 7 planes, with 25 positions for each plane, was chosen, as shown in Figure 3.17(right). The color target was placed on a tripod, and

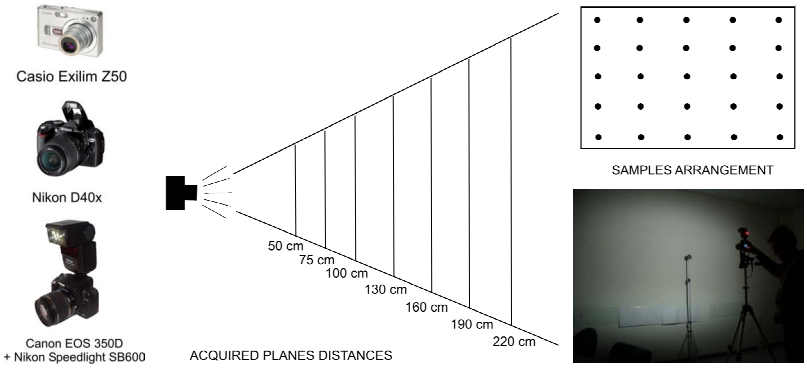


Figure 3.17: Left: Digital cameras used for light space sampling. Right: the scheme of acquisition for flash space sampling, with a snapshot of the acquisition setup.

always faced towards the camera. For each position multiple snapshots were taken (in order to deal with the known variability of flash behavior), keeping a fixed exposure time and aperture during the entire procedure.
The snapshots were acquired in sRGB RAW format to avoid any other internal processing by the digital camera, except for the Casio compact camera, with which it was necessary to use JPEG images. The acquisition procedure of all the needed images for each model took a couple of hours. The processing of acquired data was subdivided in two main phases. In the first one all the acquired images were calibrated, using the color target reference. In the second phase the color correction space was built through parameters interpolation. Figure 3.18 shows a schematization of the entire data processing.

Color Calibration.
The color calibration was done using the calibration model explained in the previous Subsection. The parameters of the matrix $\mathcal{C}$ were estimated by solving a linear system with the c_{ij} as unknowns. The parameter estimation algorithm takes inspiration from the RANSAC approach.
Since four colors could be sufficient to write a system with 12 equations and 12 unknowns, several random combinations of colors are used to find the best solution in terms of quality. The quality of the color correction is evaluated considering the CIELab distance [46] between the real colors and the corrected ones. Hence, after extracting the Mini Macbeth in a semi-automatic way and segmenting each color patch, several permutations of the colors are tried in order to find the best estimation in terms of quality. This robust approach produces better results, from a perceptual point of view, than a standard least square approach.

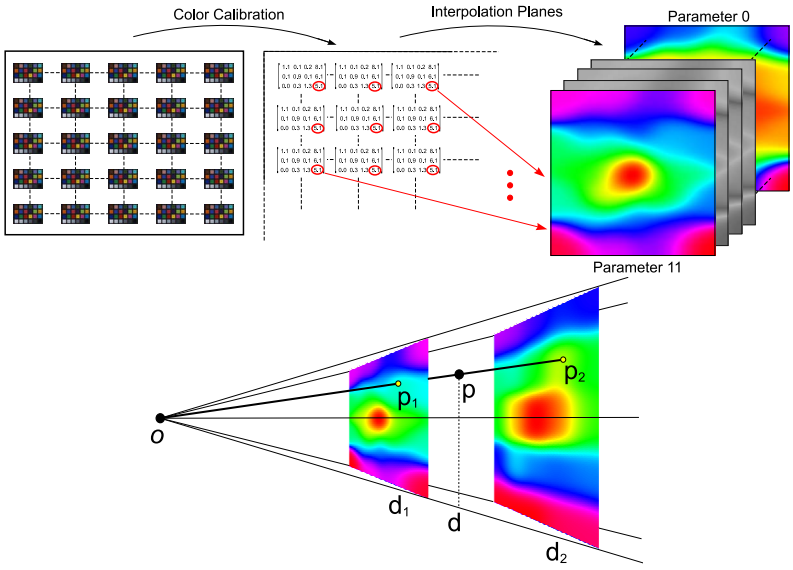


Figure 3.18: Data processing. Top: the acquired color targets are calibrated generating a set of color correction matrices, and then interpolated on each plane. Bottom: each correction parameter is interpolated to fill the whole camera space.

After calibration, each image is associated to: the 12 parameters of the matrix $\mathcal{C}$, the position of the Mini Macbeth relative to the camera view and the distance at which the image has been taken. Since several shots of the same position of the target are taken, obtained values are the means of all the estimations. Before this, the data are further processed with a statistical analysis [73] to remove the outliers from the acquired data.

Data Interpolation.
Starting from a set of color correction matrices for several points in the space, it could have been tried either to fit a mathematical function or to calculate the intermediate color correction matrix through interpolation. Here, it was opted for interpolation, leaving the first approach as an interesting direction of future research. Given a point p in camera space, the corresponding color correction matrix was calculated as the linear interpolation in the squared distance between the camera center O and the point p in the following way:

$$\mathcal{C}_{ij}(p) = \mathcal{C}_{ij}(p_1) + (d^2 - d_1^2)\frac{\mathcal{C}_{ij}(p_2) - \mathcal{C}_{ij}(p_1)}{d_2^2 - d_1^2} \tag{3.2}$$

where $\mathcal{C}(x)$ indicates the color correction matrix at the point x; p_1 and p_2 are the

intersection points between the line starting from O and passing through p. These points lie on the acquisition plane immediately before and after p (Figure 3.18 bottom); d_1, d and d_2 are the distances between the point O and p_1, p and p_2.
Since $C(p_1)$ and $C(p_2)$ are not known in advance, they have to be estimated to evaluate (3.2). In fact, only few positions on the acquisition plane are measured: hence another interpolation is required. For this planar interpolation radial basis method was used, with gaussians centered on each sample: standard deviation σ defines how much each sample influences the neighbors. In formulas:

$$\mathcal{C}_{ij}(p) = \frac{\sum_{i=1}^{N} \mathcal{C}_{ij}(P_i) \exp\left[-\frac{(p-P_i)^2}{\sigma^2}\right]}{\sum_{i=1}^{N} \exp - \left[\frac{(p-P_i)^2}{\sigma^2}\right]} \tag{3.3}$$

where N is the total number of samples for the plane and P_i are the positions of samples.
In conclusion, in order to calculate a color correction matrix for a point in the camera space, it is first needed to calculate two linear interpolations in the acquisition planes, then a linear squared interpolation for the final result has to be calculated.
Regarding the practical implementation, the interpolation planes are pre-computed and stored as floating point textures plus, for each texture, some additional information (the distance from the camera center and a scale factor). With this representation, the correction algorithm can be entirely implemented on the GPU, and the pre-computation of the interpolation planes reduces the evaluation of $C_{ij}(p_1)$ and $C_{ij}(p_2)$ in (3.2) to two texture lookups.

3.3.2 Analysis and validation of the data obtained

The analysis of the data obtained from the acquisition was performed before the validation of results. A first analysis was on the value ranges obtained for the single coefficients of the matrix. Table 3.4 shows the statistics relative to all the coefficients calculated for the Nikon camera. The single mean values of each group are very similar, and variance describes a general stability of data. Moreover, color deviation coefficients describe the flash as very near to a white light (with a slight deviation in green channel). Contrast and Intensity offset groups show quite low values in variance, but, as it could be expected, the needed modification of color values increases a lot with the distance.

Further information about the properties of flash light can be inferred from the analysis of the azimuthal and normal sections of the correction space. In Figure 3.19

	Contrast			Intensity offset			Color deviation					
Coefficient	c_{11}	c_{22}	c_{33}	c_{14}	c_{24}	c_{34}	c_{12}	c_{13}	c_{21}	c_{23}	c_{31}	c_{32}
Mean value	1.20	1.25	1.33	52.07	50.90	50.47	.07	-.06	.005	-.12	.07	-.31
Variance	0.06	0.09	0.09	81.57	123.5	141.3	.001	.003	.002	.009	.001	.012
Min value	0.83	0.84	0.87	27.5	15.4	14.2	-.06	-.43	-.29	-.67	-.21	-.85
Max value	2.59	2.84	2.86	72.0	75.0	76.8	.23	.01	.08	.01	.12	-.11

Table 3.4: Statistics of single coefficients for Nikon camera

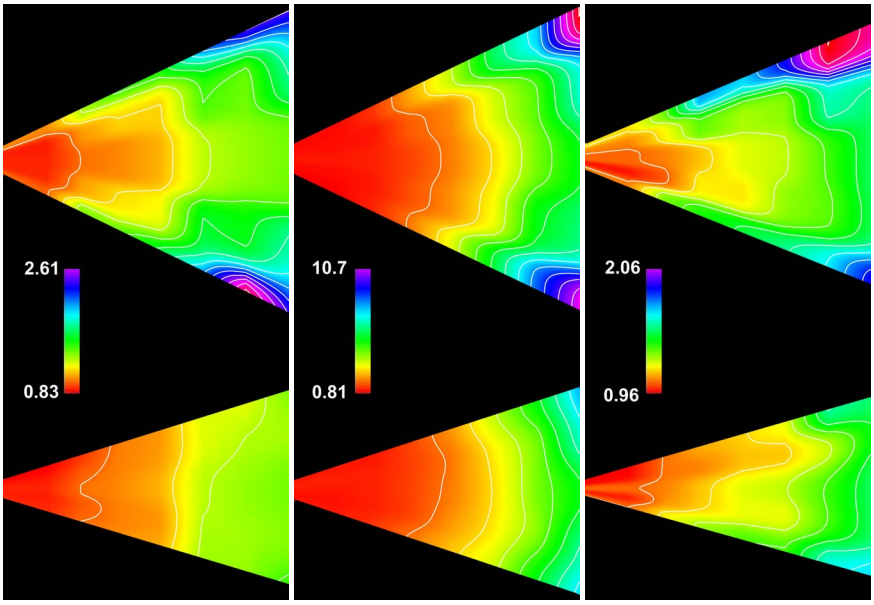


Figure 3.19: Azimuthal and normal plane for parameter c_{11} in camera space. (Left) Nikon camera. (Center) Canon camera. (Right) Casio camera.

the planes associated to coefficient c_{11} are shown: isolines help in understanding the "shape" of the light. The light wavefront is quite similar for all the models, but several differences arise as well. The most regular profile is the one associated to the external flash (Canon camera): this is probably the most reliable kind of illumination. Nevertheless, the isolines show that the shape of the light is not similar to a sphere, but it could be better approximated with series of lobes. Moreover, it can be noted that the maximum value of the coefficient is higher respect to the other two examples: this is because the external flash was set with the largest possible field of view, resulting in a very bright illumination for near points, with the need of more correction for the far ones. The Casio associated space is the less regular one, and this is probably due to the fact that the camera model was the least expensive one, and the images were processed and stored directly in JPEG format. Anyway, an interesting observation is that the correction space seems to be slightly shifted on the right, like the position of the flash with respect to the lens (Figure 3.17).

In order to evaluate the results of the correction introduced by FLiSS, a setup where the color values in selected regions were known in advance was created: the scene was formed by a series of identical objects, set at different distances from the camera. The reference objects were simple structures (formed by bricks of different colors) of LEGO DUPLO©, whose size and color are known to be practically identical for all components. The reference RGB color value of each brick was calculated by calibration with the Mini Macbeth. Seven blocks of LEGO© bricks were put in a setting shown in Figure 3.20(left), and the corresponding 3D model of this configuration was created. This scene has the advantage that, in every single photo, identical elements are present in different parts of the image and at different distances in space. Images were aligned to the 3D model using a registration tool: thus color correction was performed on each pixel which was framing a structure.
One possible criticism to our work could be: would a simple model, for example a point light, be enough to provide a good correction? For this reason, the flash was modeled as a point light, and the same images were corrected using this model. An acceptable estimation of the intensity of the light was obtained by modeling the light degradation with a quadratic law and taking into account the knowledge of the distance of each pixel in the scene, and its reference color: this light estimation was used to correct the color value. Results of corrections on one of the images are shown in Figure 3.20(left): the original, the Fliss corrected, and the point light corrected images are shown. While in both cases the colors across the scene seem similar, obtained color values are clearly different between the two results. In order to check the accuracy of color correction the perceptual distance between produced colors and the previously calculated reference colors was measured. It was decided to use the CIE76 Delta E measure, which is the Euclidean distance between the CIELAB values

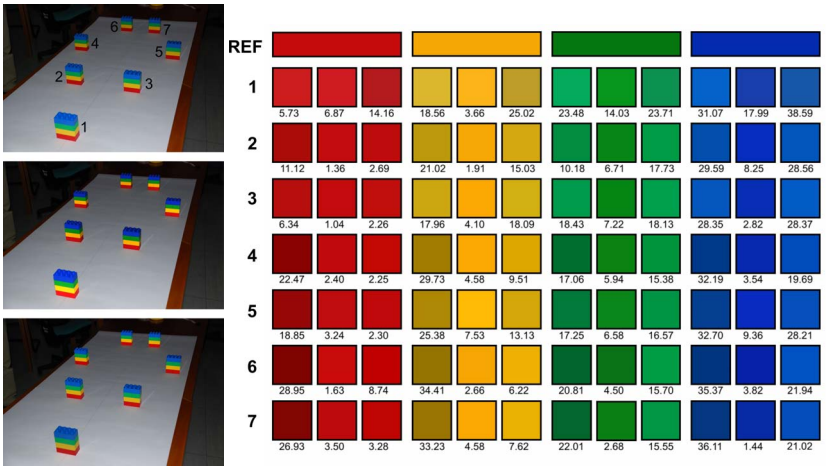


Figure 3.20: Left: example of images used for validation: top, original; center, Fliss corrected; bottom, point light model corrected. Only the Lego blocks part of the image was corrected. Right: correction accuracy estimation: for each Lego block, average color and Delta E value (respect to reference) in original (left column), Fliss corrected (center column) and Point Light corrected (right column) images are shown

associated to two colors. If Delta E is smaller than one, colors are perceived as identical; for values smaller than eight-ten, the colors can be considered as very similar.

Figure 3.20 shows the results of the calculation of these distance values. The top boxes display reference colors, then for each piece (each line is associated to the piece number indicated in Figure 3.20 left) the three columns represent the average color value for original, Fliss corrected and point light corrected image. The Delta E value for each color shows the distance with respect to the reference. It is quite clear that, although producing similar colors, the point light correction returns results which are different respect to the reference: only red color is accurately corrected. Slight improvements are introduced only for distant objects. In order to achieve better results, probably a different modeling for each channel would be necessary. On the contrary, Fliss correction proves to be very reliable, with an average Delta E value which is always smaller than ten. Only in the case of block number *1* the correction is less effective: this can be due to the fact that the object is out of focus, or the color in the original image may be saturated.

A second validation experiment was performed in order to show that the correction introduced by Fliss can be reliable also with low quality geometric information of the scene. The geometry of a common scene was reconstructed starting from a few photos, using the Arc3D web service [139]. Figure 3.21 shows the starting image,

Figure 3.21: Example of color correction using low quality geometry: Top left, original flash image; Top right, a snapshot of the extracted geometry; Bottom, Fliss corrected image.

together with the geometry obtained, and the corrected image (the correction was applied only where geometry data was present). While the obtained 3D model is only an approximated representation of the geometry in the image, the result of FLiSS use is satisfying: for example the color of the tablecloth after correction is the same throughout the scene.
Analysis of validation tests brings us to three main conclusions: first of all, Fliss is a very reliable way to correct images; moreover, simpler light models are not able to achieve comparable results. Finally, the approach is reliable regardless of the quality of the geometry associated to the image.

3.3.3 Application of FLiSS for color projection

Once the flash position and the color correction space have been reconstructed in the camera calibration phase, it is possible to detect and eliminate some of the macroscopic artifacts present in the flash images. As stated in the introduction, the starting input is formed by the 3D model of the artifact and a set of flash photos which have been registered to the 3D model [65]. Provided that the estimation of the reconstructed flash position is correct, there are two artifacts we can recover: highlights and shadows. Moreover, color is corrected by taking into account the reconstructed flash light attenuation and color transformation.

Highlight and shadows detection and removal

Highlights are present on the parts of the 3D surface where specular reflection can happen: specifically, where the ray from the light source would be reflected toward the camera viewpoint. Given the 3D model and the registered image, it is possible to find the highlight areas by using the same realtime technique used to display Phong specular highlights (the half vector technique). Unfortunately, geometric consideration alone are not enough to discriminate highlights in the images, due to local changes of the surface BRDF (that is unknown as well), minor discrepancies of the 3D model w.r.t. the real surface, and other similar irregularities. For this reason it is better to use this geometric considerations just to select candidates for a possible highlight, and then to set the actual highlight extent by performing a comparison with the corresponding regions of the other images. Using the redundancy between different photos, it is possible to compare the luminance of the candidate point with the luminance of the corresponding area on other photos. Since highlights depend on the camera position, is highly probable that the same area will be highlight-free in the other photos. The luminance value of the highlight candidate pixel is compared with the average luminance value of the corresponding pixels on the other images. If the difference in luminance is bigger than a fixed threshold, the pixel can be marked as an highlight. A two-level threshold is used: if the luminance value is between 150% and 180% the average, the pixel is on the border of the highlight, if it's bigger than 180% the pixel is considered as completely saturated. This two-levels threshold also reflects the nature of the highlight in low dynamic range images: the border of the highlight (marked in our system with a blending ramp) presents a luminance shift that rises progressively towards the central area, which is composed entirely by over-saturated pixels (marked as completely useless and thus not used in subsequent weighted average computation of final mesh color).

An example of a highlights detection result is shown in Figure 3.22. The presented model is characterized by a very reflecting material. Most of the highlights are detected automatically (see upper row images). Lower row images present a detail view: given all the pixels mapping on mesh vertices which have been detected as geometric candidates for highlights (green pixels), only a subset are detected as real highlights, and marked as border (blue) or over-saturated (cyan).

Detecting the parts in the images that are in shadow is also quite simple, by using the same shadow mapping technique used in realtime shadow generation. Using the camera associated to the specific flash image it is possible to obtain a depth map for the image. Similarly, given the flash position offset it is possible to generate the depth map for the light source; comparing the two depth maps it is possible to determine which parts of the flash image are under shadow. An example of the accurate results obtained for shadows detection is shown in Figure 3.23. The photo was taken using

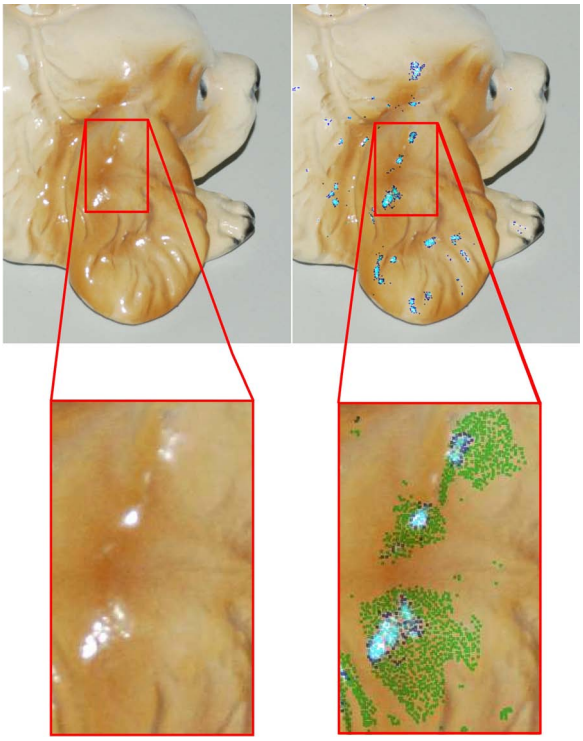


Figure 3.22: An example of highlights detection. Upper row: an input flash image and the same image after detection of highlights (blue is the highlight border, cyan is the internal highlight region). Lower row: detail view of a group of highlights with corresponding geometric candidates (in green) and detected highlight pixels.

Figure 3.23: An example of shadows detection: left, the original image; right, the shadows detection map.

the Canon camera with external flash, which is positioned to a greater distance from the camera. The shadow position is detected with great accuracy at any distance from the viewpoint. The good results in shadow removal are also an indirect demonstration that the flash position was estimated in a sufficiently accurate way in the camera calibration phase.

These automatic methods to locate these artifacts can be valuable for a number of uses: from simple image enhancement to material estimation or image relighting.

Color correction on flash images

Another automatic enhancement which can be applied to the set of flash images is the correction of pixel color values. Once that the camera viewpoint is known, every pixel which maps on the 3D model can be associated to a position in the space and consequently to a color correction matrix. A color correction result is shown in Figure 3.24.

Thanks to the color transformation, colors encoded in the flash images are brought back to values which approximate the real color values. The mask in the center is a normalized image of the medium value of the offset correction (c_{14}, c_{24}, c_{34}), which goes from 20 to 35. As it can be noted, the correction value increases with both the distance of the pixel with respect to the center of the image and the distance of the depicted surface point with respect to the view point. These color-transformed im-

Figure 3.24: An example of color correction applied on a flash photograph: the original image (left); a normalized map of the offset correction applied to the image (center); the color-corrected image (right).

ages can be used to produce a more accurate color mapping on the digital 3D model, as shown in the next section. It is also possible to apply only part of the correction matrix, to fix for example only the color drift, or to compensate for the poorly illuminated parts of an object.

Color projection on 3d models

The methodologies for spatial color correction and artifacts removal presented in the previous subsections are quite general, and can be profitably used in different situations. To show the potentiality of this kind of processing, its impact was tested in the framework of color mapping from photos.

A test set of objects was selected to assess the quality and the impact of the flash light approach. The test set was a group of objects including some Nativity statues of different heights (from 20 to 60 cm.), which was characterized by different colors and reflecting materials. All the objects were 3D scanned and a set of photos (from 13 to 32 10Mpixel photos for each object, depending on object size and complexity) was acquired. The photos were taken turning off the lights in the room, thus having flash light as the principal light. The Nikon D40x was used for most of the test presented. Images were aligned on the models, and for the color mapping the approach of TexTailor was extended by adding another weighting mask that takes into account

Figure 3.25: Details of two colored models, visualized with no illumination (only color values assigned to vertices): on left, the result with all enhancement except shadows removal, on right the model after applying shadows removal.

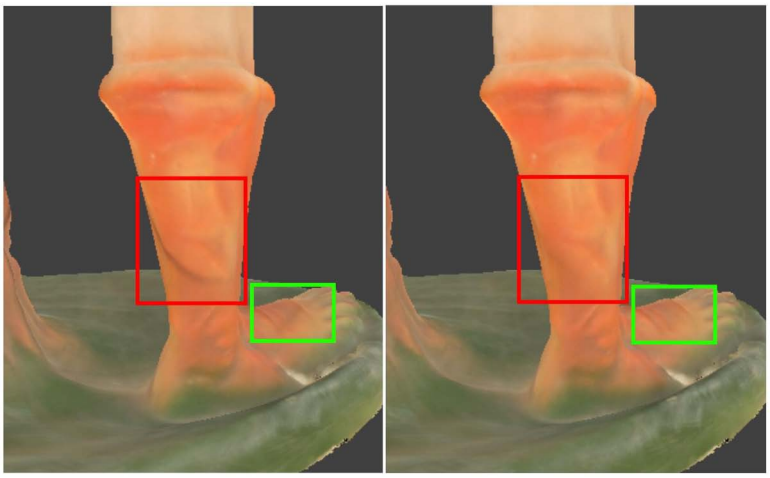


Figure 3.26: Details of two colored models, visualized with no illumination (only color values assigned to vertices): on left, the result with all enhancement except shadows removal, on right the model with also shadows removal.

the result of artifact removal methods.

Some of the results are shown in the next figures. Figure 3.25 shows two examples where the highlight removal produced considerable improvement in the final result. In the upper row, several spot-like highlights were removed. In the lower row, some highlights more complex in shape were completely removed from a 20 cm Nativity statue. The effect of shadows removal in most cases appears more subtle with respect to the highlight processing, this because, after the blended mapping, the residual trace of shadows is just a darkening of areas that can often go unnoticed. However, when the hard shadow line is visible, the advantage of the removal process is significant, as in the example in Figure 3.26, which shows a detail of a model with and without shadows removal. It can be noted that even the very small shadows are detected, such as the shadows projected on the back of the leg and on the top of the foot (see regions framed in the image).

The effects of color correction can be seen in Figure 3.27, where the Piper statue is visualized with no illumination. The model on the right, obtained with color correction, appears much more ”flat” respect to the model produced without color correction (left), which suffers from the discrepancies between the original images.

The improvements in the obtained results show that flash lighting space sampling can produce better and more faithful results in color projection on 3D models, but also contribute in increasing the performance of the mapping process, since all the artifacts removal methods are completely automatic. It is also not necessary for a

Figure 3.27: An example of color correction result, visualized with no illumination (only color values assigned to vertices); color correction is OFF in the left-most image and ON in the right-most one.

user to manually mark the artifacts, or to correct the final result.

Obviously, like all the contexts where no information about the material is known in advance, some conditions could lead to unsatisfying results. This happens for example when there is not enough redundancy in the photo data set, or when the object material presents a peculiar BRDF behavior.

Nevertheless, the use of FLiSS proved to be extremely useful, especially considering that the time to take shots is extremely limited respect to the controlled environment setting up. Moreover, the structure of FLiSS can be extended to a number of possible other applications:

Image enhancement The concept of image enhancement is not necessarily related to a complete color correction. Once one or more photos of a scene are provided, and basic information about the geometry is given by the user (and/or inferred from images), it could be possible to interactively apply color correction by using the elements of the matrix only partially (i.e. correct only color deviation) to obtain the desired effect, which could be either visual pleasantness or enhanced readability.

Appearance Acquisition Another potential application of FLiSS is in the framework of the capture of the appearance properties of a material. The reflectance properties of a surface can be represented by sampling the material BRDF or, for non-homogenous materials, by sampling a SBRDF (spatially-varying BRDF). Most estimation approaches rely on a controlled illumination setup

(e.g. a single point light source) and a mechanical setup to move the object or the light. The FLiSS framework does not include any geometric information about the light source characteristics that are required for each kind of inverse rendering approach, but if a slight variation of the FLiSS is applied, so that the flash light exhibits a behavior very similar to a point light source, therefore it could be used for BRDF recovery.

Flash Modeling The spatial intensity variation recovered from the FLiSS structure (the diagonal of the correction matrices) could allow to build a more grounded model of the flash light. While it was shown that the simple point light model exhibits strong limitations in capturing the actual flash light shape and variations, a more complex model (i.e. a set of closely packed textured point lights) could approximate in a very accurate way the behavior of the flash light. The data recovered by the FLiSS could provide the input data for a fitting procedure.

Several other possible uses of the correction space, like image registration or even maintenance and security, would involve a different contribution by the single elements of the matrix. Hence FLiSS can be considered as a very promising structure, which could have a real potential in several different application fields.

3.4 Applications

In the previous Sections, several practical solutions to common issues in the field of color projection have been presented. One of the main strong-points of these approaches is that they were designed with the aim of being usable in very general cases, and generality and flexibility were basic goals.
In fact, most of the automatic solutions presented to overcome the major issues in this field (image registration, color projection, material estimation) proved to be scarcely applicable in reality. This is usually due to several main aspects, like:

Complex setup Some of the proposed methods, especially in the field of color and material acquisition, are based on the use of prototypal or very expensive hardware. This means that usually the only practical cases which can be shown are the ones prepared for the publication of the corresponding papers. A group of works which show this kind of drawback are the ones related to the calculation of BRDF: complex setup and quite expensive hardware are usually needed, and it is often very difficult to re-create the laboratory condition in a museum or outdoors.

Data management Several solutions provided for color projection can be applied only with a limited amount of data. A typical example is the use of texture mapping: in fact, as soon as the model grows bigger in size and detail, its automatic parametrization becomes very hard. Moreover, some methods need to load all needed data in memory, and this can be a big limitation, especially when tens of images have to be taken into account. Finally, some methods need pre-processing operations (for example, silhouette matching works at its best when the images are manually de-contoured): this can be very time consuming, especially when the number of images is very big.

Manipulation of the object Finally, some methods (mainly in the field of material estimation and camera calibration) need severe manipulation of the object, with the use of markers. This operations cannot be made in a generic Cultural Heritage application, since usually the objects to be acquired can be valuable or fragile. Hence, only methods where the contact with the object is the minimum possible must be taken into account.

On the base of these considerations, the applications shown in the next Subsections present very different examples of use of the tools presented in previous Sections. This demonstrates that the choices made during the research phase allowed us to create really flexible and robust solutions.

3.4.1 The Double David project: mapping highly detailed color information on extremely dense 3D models

Michelangelo's David is one of the most popular art icon of our age, a rather old friend who deserves cure. The statue was always kept under control and subject to several restoration actions in the past. At the end of last century, a major restoration action was planned to allow it to begin the new millennium with an improved look and a complete assessment of its conservation conditions. The restoration planned was a light one, mostly focused on the removal of dust, spots and other deposits accumulated in the years on its surface, and on the replacement of the plaster fillings of some fractures (e.g. the ones filling the small gaps of the fragments of the arm broken in XV cent.). Therefore, no major change of the shape was planned, but just a selective cleaning which should have changed its surface appearance. Nevertheless, the restoration was preceded by an intense and very complete set of scientific investigations, aimed at making a severe screening of the statue conditions [29]. The interest of the curators to run a complete scientific assessment of the statue condition, together with the availability of a digital 3D model of the statue (scanned by the Stanford's Computer Graphics Group in the framework of the Digital Michelangelo

project [95]), made it possible to adopt a number of ICT technologies to both analyze and document the statue conditions and the restoration action [29, 31].

In particular, the David restoration project (started in Florence in 2002 and terminated in 2004) was a milestone for the definition and development of solutions which qualify a modern restoration, by integrating a number of different scientific approaches under the same global focus (IT was just one of those disciplines).

Among other tasks, it was asked to document the status of the statue before and after the restoration in a manner which should be both rigorous and able to transmit the information to the experts and the public in a simple and intuitive way. The opportunity to experiment modern visual data management approaches was immediately evident. Since the restoration was mostly a cleaning task, and perceptible modifications of the shape of the artwork were not forecasted, a new 3D scan after the conclusion of the restoration work was not planned. The changes were going to occur mostly in terms of different appearance of the surface. For this purpose, a high quality photographic essay of the pre- and post-restoration condition was the starting point and one of the major sources of data.

The Dataset

The Stanford's Digital Michelangelo project [95] produced a huge amount of different data (3D models, 2D RGB and UV images). The result of the scanning campaign was a very detailed 3D Model (56 million triangles, reconstructed from 4000 range images using a distance field with 1mm. cell size). The Digital Michelangelo project acquired also data on the surface color, but those data were not usable for our purposes both for some apparent contamination from an UV light source used while taking the RGB images and for the specific needs of the restoration documentation.

To obtain a concrete and accurate visual survey of the status of the surface, a high-resolution photographic survey of the David was performed by a professional photographer (Studio Rabatti and Domingie, Florence). The photographic sampling was done according to the specifications defined jointly by the IT staff and the restoration supervisor (a graphic representation of the planned photo survey is shown in Fig. 3.28). Photos were taken in two different periods (before the start of the restoration and at the very end), requiring around one complete week of work each. The amount of 2D data collected (61 images, res. 1920x2560 to document the pre-restoration status; 68 images, res. 2336x3504 for post-restoration status) was about 800 Mega pixels. Some sample images from both sets are shown in Figure 3.29.

The survey of the conservation status of the surface was one of the requested documents which should have been produced in the restoration process. The problem was how to plan the survey in order integrate the produced data with the digital data (2D or 3D). Nowadays, the more direct solution could be to design a tool which allows

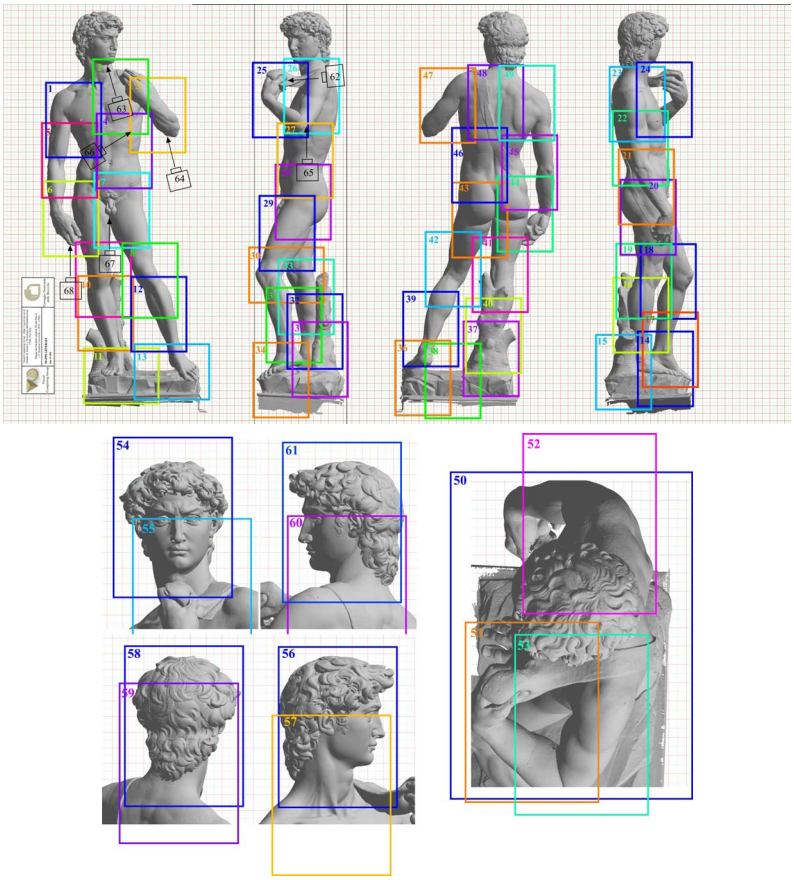


Figure 3.28: Examples schema of the photographic campaign, describing the coverage of the set of photos.

Figure 3.29: Examples of images from the pre- and post-restoration campaign.

the restorer to draw the survey directly on the 3D model, i.e. using a painting/drafting system to make the relief directly onto the 3D model. In 2004, the status of the technology did not allow us to even consider this approach. Another hard constrain was the limited experience of the restorers with IT and CG technology, and therefore this solution was immediately abandoned. The followed approach was therefore based on a manual relief drafting, followed by a digitization phase and a final mapping. The restorers performed a precise graphic survey on the status of the David's surface. They drew very accurate annotations on the high resolution photos (the ones from the photographic essay), covering all the surface of the David. These annotations described in a very detailed manner the presence of: (1) imperfections in the marble (small holes or veins); (2) deposits and strains (e.g. brown spots or the traces of straining rain); (3) surface consumption; and (4) traces of the Michelangelo's workmanship. These annotations were drawn by the restorers on transparent acetate layers positioned onto each printed images, using different color to indicate the same phenomena in the different sheets. Therefore, 4 different graphic layers for each one of the 61 high-resolution photos (documenting the pre-restoration status) were created. An example of the graphic relief is shown in Figure 3.30.

These graphic reliefs on A3 acetate sheets were scanned (using a commercial A3 flatbed 2D scanner), registered on the corresponding RGB image and saved at the same resolution.

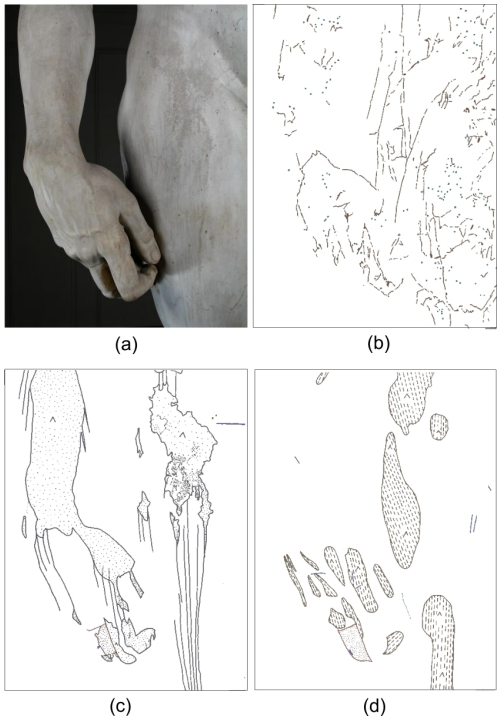


Figure 3.30: An example of an image of the photographic set with the corresponding (b) imperfections, (c) deposits and (d) consumption reliefs.

Mapping and visualizing RGB data on the 3D model

The amount of high quality data described in the previous sections was really a valuable documentation of the restoration process. In order to visualize, analyze and compare data with a paper-less mode, two approaches were chosen: (a) 'classic' 2D mapping of the reliefs on the corresponding 2D RGB images (implemented using web-based technologies), and (b) mapping of the RGB data (pre- and post-restoration images) on the digital 3D model. The 2D mapping was chosen as an easy way for experienced and unexperienced users to access the photographic and reliefs archive, and also because, at restoration time, there were several constraints to the projection and visualization of color on 3D models. Hence, an intuitive web-based system (that can be delivered either on DVD or on internet) was created. Following the scheme provided by the photographic campaign, the user can choose any of the provided views, and then visualize the corresponding photo and, selectively upon his/her choice, the superimposed reliefs related to imperfections, deposits and deteriorations. Images can be visualized with different zooming factors, so the high detail can be appreciated at its best.
One of the main drawbacks of a 2D visualization system was that since the pre- and post-restoration campaigns were taken in different moments, also the position for the surface coverage are not exactly the same. Hence, it was not possible to have a perfect overlapping between two points of view before and after the restoration.
The best solution to have this kind of visualization was clearly the projection of color information on the 3D model. In fact, a colored model could have been explored from any arbitrary position. Hence, it was decided to project the two photo sets on the high detail 3D model.
The two photographic sets were aligned on the high detail 3D model, for a total of 129 images registered. TexTailor was then used to produce two colored models, one for pre- and one for post-restoration. The subsequent operation was to find a way to show these models.
The most intuitive use for the two 3D models was interactive visualization, which could have given the user the possibility to analyze the appearance of the statue before and after restoration from any arbitrary point of view. Interactive visualization was made possible by the multiresolution technique presented in [42]; built over this data representation approach, the Virtual Inspector tool provided a framework which allowed the easy inspection and virtual manipulation of a complex and highly detailed 3D model.

A screenshot of the current application is shown in Figure 3.31. The pre- and post-restoration models are shown on the left and right side, respectively. The user can easily change the model position and the illumination, in order to frame arbitrary points of view. The main differences in the marble surface conditions can be seen

Figure 3.31: Screenshot of the Virtual Inspector visualization tool.

in a very intuitive way. Another interesting possibility is to render the model from arbitrary points of view. For example, we could use the camera parameters estimated for a photo being part of the pre-restoration set and render the post-restoration model from the corresponding camera position. In this way we can obtain an image of the restored model which is perfectly aligned to the starting pre-restoration photo, resolving the issue mentioned above. An example of this particular mapping is shown in Figure 3.32, where the left-most image is taken from the pre-restoration set and the central image is the corresponding one from the post restoration set. It can be easily observed that the point of view is slightly different, and superimposition is impossible. The right side image is a rendering of the 3D model with post-restoration color from the camera parameters estimated for the left side image. Now, this syntectic image could be very useful not only for superimposition of re-restoration reliefs but also to try to reproduce a similar illumination condition, which was slightly different between the two sets.

In conclusion, the mapping of the David was an interesting application of color projection in Cultural Heritage. In particular, 3D navigation added significant possibilities to compare the visual appearance of David before and after restoration. Moreover, highly detailed geometric information and highly detailed color information were put together in a real-time navigable system.

Figure 3.32: Left: one of the images of the pre restoration set; center: the corresponding image in the post restoration set; right: rendering of post restoration 3D model from camera position of pre restoration image

3.4.2 The Cenobium project: Binding Up Interoperably Usable Multimedia

The CENOBIUM (Cultural Electronic Network Online: Binding up Interoperably Usable Multimedia) [12] project faces the necessity to improve scientific and educational communication on the one hand and public information systems on the other hand, integrating new investigation instruments, not systematically connected until now. The final result will be to provide a web-based, openly accessible work environment, which includes 3D models created by scanning, CAD-representations, digitized historical photographs and digital photography of the highest professional quality. The technical work will be devoted to the integration and extension of available technologies (database, image-viewers, 3D-viewers, content management, etc.) now dispersed and not-interoperative. The project points to the introduction of multimedia investigation of artworks as a regular research-instrument in the service of its different user groups. The first specific case study considered for the assessment of our approach was a selected group of important capital-cycles in medieval cloisters of the Mediterranean region, starting with the cloister of Monreale [11].
The art-historical material was highly adequate for multi-dimensional representation, given the 3-dimensionality of the capitals and their spatial connection with the surrounding architecture - aspects that can not be explored adequately by relying exclusively on 2-dimensional photography. The cloister of the Cathedral of Monreale in Sicily demonstrates particularly well the diversity and the range of opportunities a Romanesque sculptor had in expressing his art. The monastic complex was commissioned by King William II and executed between 1174 and 1189. It unites various artistic currents of Romanesque monumental sculpture into an architecturally

homogeneous setting. Each of the cloister galleries consists of 26 twin colonnettes, whereby the corner piers join the columns and capitals into groups of four. The southern and western galleries merge by creating a small square courtyard with a fountain in the center and five additional twin colonnettes with capitals. Researchers identified various contemporaneous workshops composed of artists from various Mediterranean countries, such as mainland Italy, France and Spain, who worked on the spoliated marble shafts and capitals. In this respect the high-quality execution of the cloister capitals of Monreale unites, with its rich formal and iconographic repertoire, the main currents of artistic production of the second half of the 12^{th} century.

Data acquisition and processing

A comprehensive data acquisition campaign was performed in order to generate a dataset of heterogeneous material. A Sinar P3 digital camera was purchased by the Photo Library of the Kunsthistorisches Institut, providing for the integration of the digital backs Sinarback 54 H and Sinarback eMotion 22, both of them with a resolution of 22 million pixel (sensor resolution 5440×4080 pixel), as well as various Sinaron lenses. The size of a single image is approximately 65 Megabytes (TIFF

Figure 3.33: An example of the set of photos acquired for sampling the color of each capital.

format uncompressed, 8-bit-per-channel colour depth, 4000×4000 pixels - approximately 33 cm on a 300 dpi printout). All the capitals of the Cloister were acquired with a set of 8 images. A set of photos, documenting a capital, is shown in Figure 3.33.

High quality 3D models of the capitals were produced by using a Konica Minolta VI 910 Laser Scanner. Since the scanner works at a distance between 50 and 100 cm

]

Figure 3.34: The acquisition setup adopted to scan the capitals.

from the objects, it was necessary to put it on a scaffolding, as shown in Figure 3.34.

A number from 120 to 200 single scans (each scan samples around 0.3 Million points) was needed to cover the entire surface of each capital. In the first scanning campaign (February 2006), which lasted for an entire week on site, 20 out of the more than 100 capitals of the cloister were acquired. To sample this initial subset nearly 4000 range maps were shot. The ISTI-CNR tools [32] where used to produce the final models.

Twenty highly detailed 3D models of the most artistically interesting capitals of the cloister were reconstructed. Then, the color information provided by the photographic campaign was mapped on them using TexAlign and TexTailor. The screenshots of two models are shown in Figure 3.35.

The complete cloister acquisition was also the focus of other digital acquisition actions. One goal was to produce a 3D model of the entire cloister together with high-resolution panoramic images. Panoramic images were created by processing a set of digital photos (medium resolution, acquired with a consumer digital reflex camera). Moreover, the 3D model of the entire cloister were produced with a Leica Geosystems HDS 2500 time-of-flight scanner.

Figure 3.35: Left: the "Sh10" capital, ornamental leaves. Right: the model of "Sh37" capital with color information.

Exploring the cloister in a collaborative environment

The CENOBIUM project is still an undergoing work, in particular two additional interesting sites (the cloisters of Aosta and Cefalú) have been chosen for a similar acquisition campaign. The models of the capitals of these two cloisters are going to be integrated in the Cenobium website, so that the experts are able compare elements of the three sites in a user-friendly environment.

Figure 3.36 shows two screenshots of the current implementation of the Monreale website. Together with the general information about the cloister (left), each capital has its own information page, from which it is possible to access to the high detail material (both images and 3D models).

In order to be able to access and navigate the high resolution images, the system was implemented using the technology of Digilib [64], a web based client/server technology for images. Hence, the user is able to visualize the images at the best resolution (see Figure 3.37 left).

Moreover, it's possible to download and visualize the colored 3D models using a simple navigation tool [35] (see Figure 3.37 right).

In this way, analogously to other recent projects like [10], in a single website-like environment, it is possible to have access both to typical historical and artistic information and to high detail material (images and 3D models). The innovative aim of the Cenobium project stands in the setting of a collaborative environment, where additional information can be annotated by experts. Moreover, a mechanism to select, download and compare different elements of the site (for example images and models from different artistic sites) is under development. This would be a very interesting example of a really complete and easy to use environment.

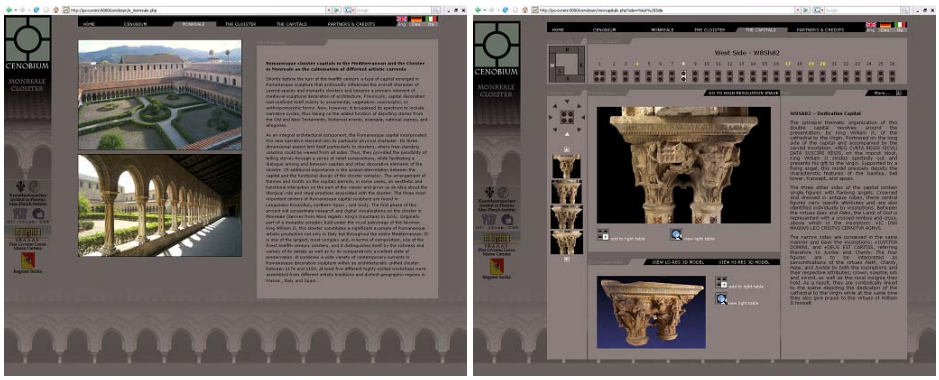


Figure 3.36: Two screenshots of the Monreale kiosk. Left: a general information page about the cloister. Right: the page which present a single capital.

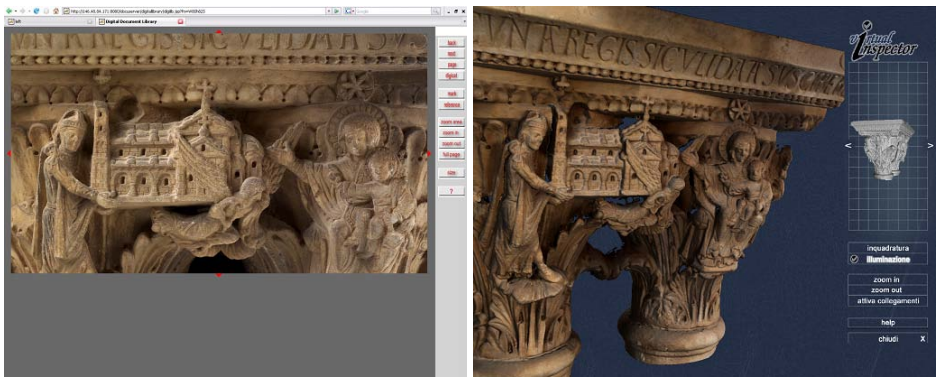


Figure 3.37: Two screenshots of the Monreale kiosk. Left: exploring an image using Digilib Right: exploring a model using Virtual Inspector.

Figure 3.38: The Portalada: romanesque entrance of the Ripoll Monastery. It is easy to perceive its impressing size but, at the same time, the richness of the sculpted relief.

Regarding the color projection issue, in this application it was shown that even images at really high resolution could be successfully projected on 3D models.

3.4.3 Multiscale acquisition and presentation of very large artifacts: the case of Ripoll's Portalada

The Benedictinian monastery of Ripoll was founded by count Guifré el Pilós in 879. The main work of art from the monastery, which is also the main Romanic sculpture from Catalonia, is the entrance (known as **Portalada**, see Figure 3.38), which dates back to the 12th century. This entrance has been defined as the "Stone Bible". It is a masterpiece of cultural, historical, social and scientific interest. Ripoll is located 90 Km North from Barcelona, quite apart from the main tourist routes. It was necessary to find novel approaches to show this impressive monument to the visitors in Barcelona, and disseminate its contents to a wider audience. This motivation, together with the progressive deterioration of the stone, was a strong argument for romanic experts and museum curators at the Museu Nacional d'Art de Catalunya (MNAC) in Barcelona to ask for a high-accuracy virtual model of this significant cultural heritage masterpiece.

The virtual reconstruction and presentation of the Portalada monument [21, 41] was performed to produce an interactive installation, open to the public, in the MNAC exhibition named "The Romanic Art and the Mediterraneum. Catalonia, Toulouse

and Pisa" from February to May 2008.
The project challenges derived from the requirements of the museum curators. The goal of the project was three-fold: to have a high-fidelity virtual reproduction for the exhibition, to create a tool for the study and analysis by the experts, and to allow the archival of the present status of the Monastery entrance.

Data acquisition and processing

The Portalada of Ripoll is an incredibly rich monument: every single section tells a story, sacred and profane mix continuously throughout its surface. For this reason, the acquisition of the portal needed a particular care in the preservation of detail. In the case of Portalada, the study of a single statue or bas-relief can be the topic of the work of an art historian for several years. Moreover, most of the single statues and bas-relieves present small details (either carved portions or deterioration traces), which can be very important for their artistic and historical message, or for conservation purposes. For these reasons, it was decided to digitize the monument using triangulation-based scanning, with the aim of having an inter-sampling density of at least 1mm. Since performing a triangulation-based scanning of such a huge carved surface (7x11 meters) was very challenging (difficulty in covering all the surface, complexity in registration step, etc.), it was decided to run a double acquisition, using both triangulation and time-of-flight laser scanners.

The reasons behind this choice were several:

- The area covered by a single range map acquired with a triangulation scanner is usually not bigger than 50x50 cm. Moreover, it is usually needed to sample this region from many different directions to produce a complete sampling of the bas-relief detail. Consequently, several hundreds range maps were needed to cover the entire surface. Processing all those range maps is not easy, especially the alignment phase. The accumulation of the alignment errors could have made impossible to reach the global alignment of all the range maps. Moreover, the accumulation of residuals could have led to a severe deformation of the final model. The use of a time-of-flight scanned model as a reference for alignment solved both these issues.

- The technical constraints of the triangulation scanner (acquisition distance between 50 and 120 cm, small areas covered by each shot) result in difficulties in having a complete sampling of such a large object. The different point of view and the more consistent coverage of a time-of-flight scanning could provide

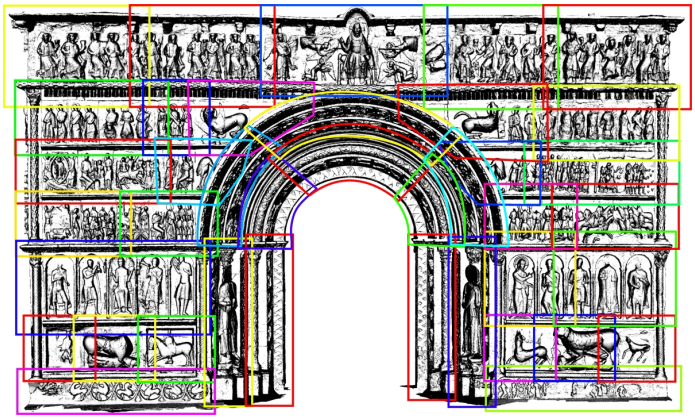


Figure 3.39: The planning scheme used to scan the Portalada with the triangulation-based technology: each box corresponds to a sub-section acquired

more information and cover parts which could not be acquired by the triangulation scanning.

- A low resolution scanning of the area around Portalada could be a good data set to locate it visually in the framework of the surrounding architecture of the monastery.

Several previous projects in the field of Cultural Heritage presented a mix of lower detail models (obtained with time-of-flight scanning, 3D modeling or photogrammetry) of large areas and triangulation-based scanning of more detailed areas [74, 131, 136, 19, 8]. In our specific case, the different resolution data were be mixed in the processing phase, in order to solve the general issues associated to triangulation and time-of-flight acquisition.

The Time of Flight scanning was performed by a team of 2 technicians using a Leica ScanStation. It took 1 day of work and 4 acquisition stations to sample the geometry of both the Portalada and the exterior monastery facade. The Portalada was scanned with an inter-sample distance of 0.5 cm, providing a data set of 36.2 Million points.

The triangulation-based scanning was performed in 5 days by a team of 4 people using 2 Minolta Vivid 910 scanners. During the acquisition planning the monument was subdivided in 62 overlapping sub-sections (the acquisition planning schemes for the frontal part of the portal is shown in Figure 3.39). Each sub-section was acquired with a number of range maps varying from 20 to 90, depending on the size and the

Figure 3.40: Left: the 7 meter high scaffolding used for acquisition. Right: a smaller scaffolding used for the acquisition of the internal part of the portal

shape complexity of the depicted relief. In order to be able to scan the upper part of the portal, a movable 7 meter high scaffolding plus a smaller one were used (Figure 3.40). The final data set was composed by 2212 range maps and more than half a Billion samples.

Finally, more than 200 digital photos were taken to sample the apparent color of the entire Portalada surface, to be used for color projection on the reconstructed digital surface.

The data processing presented some challenging issues especially in the alignment phase, due to both the amount of data and the possible deformation introduced by the size and shape of the object.
In Figure 3.39 it was already shown that the surface of the portal was subdivided in several subsections. Each subsection was acquired following a regular coverage pattern, with consistent overlap between adjacent range maps. This regularity gave the possibility to employ automatic approaches [120] for initial alignment of the range maps. This resulted in a speed up the alignment process during data processing; then, VCG tools [32] were used to obtain optimal alignment. The overall alignment of the subsections, which was the operation which needed the major amount of human intervention, was performed in nearly 10 days.

Each subsections contained a "border" of about 20-30 cm that was shared with its neighbors subsections. In principle, by using this overlap it should have been possible to use the standard alignment procedure between adjacent subsections to obtain a global positioning for all the range maps. Since it was not possible to compute global alignment on a group of more than 2000 range maps using a standard ICP-based aligner due to the sheer size of data, the alignment process was applied at multiple *levels*: subsection, border and again on the whole set of range maps.

The issue of possible final deformation was overcome by using the model obtained with TOF scanning. Since each TOF scan contained the entire extent of the Portalada, the resulting model could be used as an alignment reference, in order to obtain a rigid global alignment. Using the TOF model, it was possible to build an alignment process that proved to be viable in terms of required time and human intervention but, more importantly, to provide accurate results.

The entire alignment process was composed of three different steps, iterated several times in order to guarantee a stable and accurate final alignment.

- alignment between range maps inside each subsection
- alignment step of the range maps of a single subsection toward the TOF model
- alignment between range maps of adjacent subsections

Basically, a *manual, out-of-core, multiscale* global alignment step was performed. After generating a correct alignment for all the range maps, the next step was to create a single surface for the entire object. The total merging time required was 26 hours. The merging produced 406 separate blocks, for a total size of 170M triangles and 3.5 gigabytes.

The impossibility to reach some parts of the surface with the triangulation scanner resulted in small uncovered areas. This lack of data was clearly visible in some low detail areas of the middle arch and in some difficult-to-reach parts on high-reliefs. No major holes were present in the important and detailed areas. The missing parts were filled by carefully cutting small areas from the TOF range scans. The merging tool was also configured to use for this kind of data only when no other data were available. The result was a seamless integration of the missing area, with minimal quality degradation. The production of the final model took a further week of work, but needed human intervention was in this case very limited.

Once the geometry of the portal was reconstructed, it was necessary to add the color information, in order to enhance the realism of the visualization. In this particular case, due to the size of the portal and the peculiar lighting conditions, the acquisition of the images was quite hard, and it resulted in a set of more than 200 photographs.

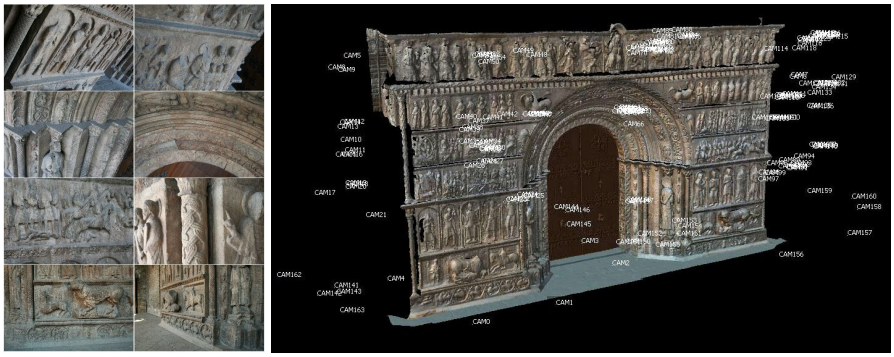


Figure 3.41: Left: some examples of images used for color projection. Right: camera position associated to the images used for color management.

While image acquisition is usually faster than 3D scanning, the photo sampling process was complicated by the need to maintain constant lighting conditions, so that the whole operation was performed in a short time, using the scaffolding. Analogously to the geometry acquisition, also in the context of the photographic campaign there was a coverage of the surface of the portal from a short range, but several images from greater distances were taken as well. In Figure 3.41(left) some examples of the images captured are shown: the portion of depicted space was also determined by the scaffolding position from which each part of the surface was acquired. Figure3.41(right) presents a snapshot of the 3D model, where the camera viewpoint of the images used for color projection is visualized as a string in the surrounding space: as it can be noted, camera viewpoints are grouped in correspondence of the scaffolding positions chosen to cover each part of the surface of the Portalada.

From the whole set of the images, a subset of 163 images, which could cover the entire surface, providing sufficiently coherent lighting, was selected. The color projection was obtained using TexAlign and TexTailor.

It is also interesting to note that the images taken from longer distances had a role similar to the one played by the time-of-flight scans in the model reconstruction: the eventual coverage of parts which were not taken by short range images. The color projection mechanism of [34] automatically assigned a lower weight to these low quality images in the zones where other images covered the surface with a better resolution/quality data.

The color coverage provided by the photographic campaign was almost complete; snapshots of details of the final colored model are shown in Figure 3.42.

Figure 3.42: Two snapshots of details of the colored model of Portalada

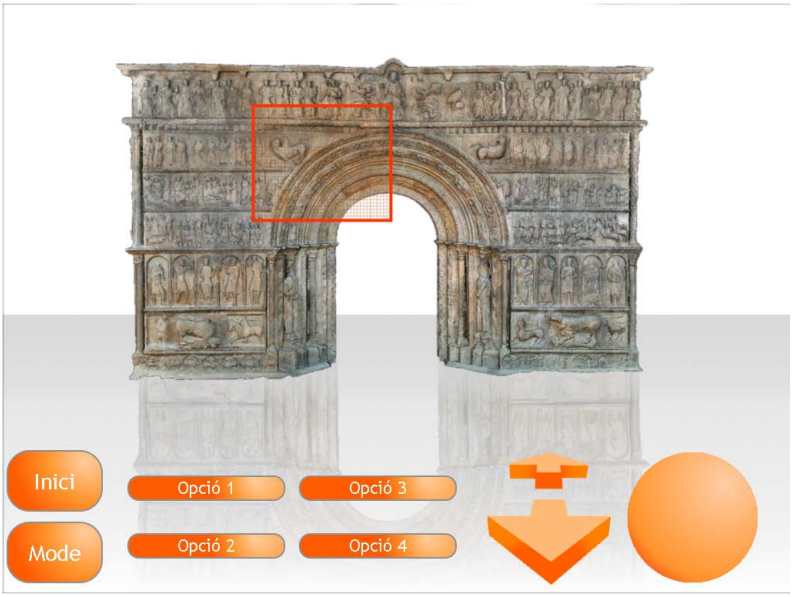


Figure 3.43: The interface of the touch-screen that controls the kiosk

The visualization interface

After the production of the model and an additional processing phase to enable multi-resolution visualization, the Portalada model was presented at the MNAC Museum in Barcelona.
Visitors were able interact with the digital model of the Portalada through two low-cost and easy to assembly VR kiosks.

The components of one kiosk included:

- One dual-core PC with an Nvidia 8800 GTX graphics board. This PC handles the visualization of the model. It manages the high-resolution meshes and computes the view-dependent front of the octree at every frame.
- Two DLP projectors with circular polarizing filters, fixed in a calibration frame that was designed in the framework of previous projects.
- A back-projection screen with a surface of 2 x 1.5 meters.
- A touch-screen for the interaction, see Figure 3.43.
- A second PC to manage the interface in the touch-screen.

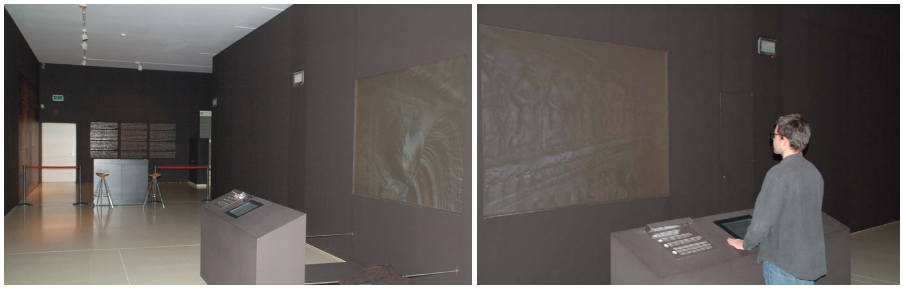


Figure 3.44: Setup of one of the VR kiosks at the MNAC museum

The user interface in the touch-screen (Figure 3.43) presented a view of the portalada and a few buttons and widgets. Visitors were able to approach by pressing on the perspective arrows, perform a ”pan” operation to go to specific parts of the monument by simply moving the finger along the Portalada to the place they are interested in, or rotate the camera by virtually rotating the ”track ball” at the lower right corner.

The user interface had two more buttons: a reset button and a ”mode” button. The application supported three modes: the navigation mode, the sun illumination mode and the information mode. The navigation mode allowed standard navigation as already described, the sun illumination mode simulated solar lighting while the user changes the time of the day and the day of the year using two standard sliders, and the information mode (the one shown in Figure 6) displayed additional information (historical photos or texts) on request.

Figure 3.44(left) presents the final arrangement of one of the two VR kiosks installed in the museum. Figure 3.44(right) shows the interaction with the 3D model at one of the kiosks. A simple usability test was performed on a group of ten persons, to tune some involved parameters like sensibility of the touch screen and speed and acceleration of the camera changes. The feedback, both from this test group and from the general public of the exposition, was very positive.

In conclusion, the Portalada project showed that the acquisition and visualization of colored model was possible also for extremely big objects: the tools for color projection were used for this particular and demanding case without the need of any modification.

Figure 3.45: Left: the Windsor silverpoint drawing (WRL #12358: The Royal Collection ©2007 Her Majesty Queen Elizabeth II). Right: the small bronze horse (Archeological Museum, Florence).

3.4.4 A peculiar application of TexAlign: using 3D scanning to analyze a proposal for the attribution of a bronze horse to Leonardo da Vinci

In this Section a peculiar application of image registration will be shown. In fact, in this case a drawing instead of an image was aligned to a 3D model, in order to provide additional clues to an attribution proposal [53].
Mark Fondersmith's attribution hypothesis is that the Leonardo metalpoint (RLW #12358, see Figure 3.4.4left) is an optically-traced drawing of a small bronze horse (inv. #19446, Archeological Museum, Florence, see Figure 3.4.4right). According to this hypothesis, the small bronze horse was modeled and cast by Leonardo, while an apprentice of Verrocchio (1470-1480.) Later, Leonardo used the small bronze horse as a reference for the Windsor metalpoint (1480).

Leonardo could have used a camera obscura (or some other optical device) to create the Windsor metalpoint. He traced the small bronze horse in 2 stages: first, the body, and then he rotated the head after rotating the bronze to the left. The two sets of front hooves are evidence of the rotation. The re-drawn neckline shows how Leonardo reconciled the second angle of the head with the original profile. The precisely crosshatched areas of the drawing describe the sculptural form of the small bronze horse. There are many points in common between the drawing and the sculpture.

Figure 3.46: A screenshot of the colored 3D model obtained with 3D scanning.

Figure 3.47: Images from the first alignment (main body).

Using 3D scanning to analyze the proposal

Visual Computing Lab was contacted for executing an accurate digital acquisition of the bronze, with the purpose of performing further study on a physical replica produced from the digital 3D model via rapid reproduction technology. After having acquired the digital model with 3D scanning technology (a screenshot of the colored model is shown in Figure 3.46), it was proposed to do some tests using TexAlign. In order to try to produce shape similarity evidences to help assess the attribution proposal, it was decided to treat Leonardo's drawing as if it was a photo. Were the camera positions and the alignments found be compatible with the use of a camera obscura? Were the matching details listed by the attribution proposer effectively

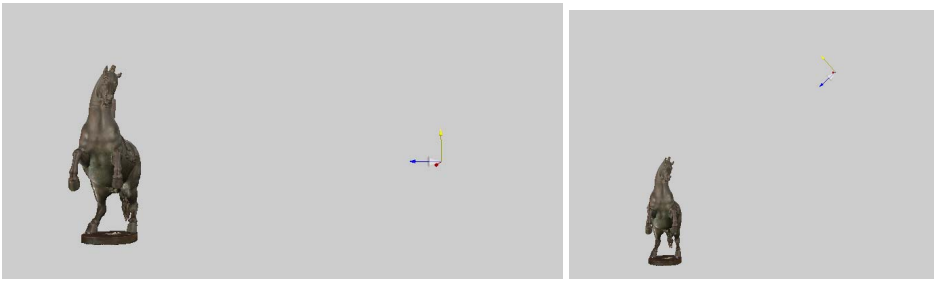


Figure 3.48: Left: relative position of the camera for the first alignment. Right: relative position of the camera for the second alignment.

coincident?

The alignment of the drawing was performed in the usual way: some correspondences points between the 3D model and the drawing were indicated. The correspondences on the drawing were set by following the indications provided by the attribution hypothesis. The two proposed points of view were considered: the first for the body silhouette and the other, rotated, for the head. Clearly, the points provided for the registration were different for the two cases. The focal length was fixed to a plausible value for the use of a camera obscura. The results of the first alignment are shown in Figure 3.47, where the overlapping of the 3D model and the drawing is shown at different transparency levels. As it can be noted, the alignment of some parts of the drawing is surprisingly precise: the lines of the neck and the back are coincident with the silhouette of the model, and the profiles of both fore and hind right legs are quite similar. Moreover, the position of the associated camera (denoted by a cone in the right-most section of Figure 3.4.4left) is clearly compatible with the point of view of a person using a camera obscura. The second alignment was performed to validate the second proposed position, from which the head would have been sketched. Results are shown in Figure 3.49: not only the profile of the head is very similar to the drawing, but also the line of the neck is overlapped to an internal line in the drawing. The associated camera position (Figure 3.4.4right) shows that the point of view is not only rotated but also raised with respect to the first alignment. The results of the alignment provided a visually significant proof and confirmed the compatibility of the use of a camera obscura to sketch the drawing from the bronze statue. It's important to stress that no deformation was introduced while performing the image-to-3D mapping. Two images of zoomed-in details are presented in Figure 3.50.

Unfortunately, in this case of image alignment it's very hard to provide numerical proofs of its quality: this can be done with photos, which exactly reproduce the

Figure 3.49: Images from the second alignment (head).

Figure 3.50: Two detailed view on the neck and saddle region (top) and on the head (bottom).

object, but it would be of no utility in this case, where the drawing is clearly not a mere tracing of the silhouette. Anyway, the surprising alignment obtained seemed to be too precise to be casual, especially considering the results obtained on similar objects. In fact, since the immediate objection to the results shown could be that, even though the alignment is very good, similar results could be achieved with other drawings or models, new alignments between the horse model and other similar Leonardo drawings and between the the model and an image of a different bronze horse were performed.
Obtained alignments, shown in Figure 3.51, were of very bad quality, with almost no feature in the drawings or in the image which fitted to the 3D model.
In conclusion, this peculiar application showed that it was possible to provide some visual proof to substain the attribution proposal. Obviously, this does not say the last word to the long lasting dispute relative to the attribution of the bronze horse, but it could help rising some serious concerns on the current attribution (i.e. to Benvenuto Cellini).

Figure 3.51: Results of alignment for another silverpoint drawing by Leonardo and an image of a different bronze horse

The debate should be solved by further study or analysis of the constituent material and by a more in-depth study of the historical documentation.

Chapter 4

Geometry modification

This Chapter is focused on the methodologies which allow to extract information from 2D images and to deploy this data to modify the shape of a 3D model. Approaches in two different fields (3D sound rendering and 3D scanning) are investigated.
The goal of the first method is to morph a geometry to fit the information extracted from images, analogously to the methods presented in Section 2.2. The final result will be an accurate model of a complete head, that can be used for the calculation of Head Related Transfer Function (HRTF). The next subsections will introduce the importance of HRTF in the context of sound rendering, the possible approaches for its calculation and an automatic system for 3D head reconstruction from images.
The second method uses an image and a laser pattern projected on an object to extract information for correct hole filling on incomplete 3D models. In practice, the method of laser triangulation is used starting from a registered image, and the redundancy of data (the already acquired geometry) is used to obtain precision from data which are not as accurately calibrated as a 3D scanner device. In this way, a low-cost method to create new geometry is obtained.

4.1 Head related transfer function from images

In order to present this approach, a preliminary overview on its peculiar application field is needed. Hence, in the next Subsections the goals, uses and open issues of the calculation of personal Head Related Transfer Function (HRTF) are briefly outlined. It is indeed necessary to clarify the importance and the requirements of the needed data to better understand the final structure of the proposed system.

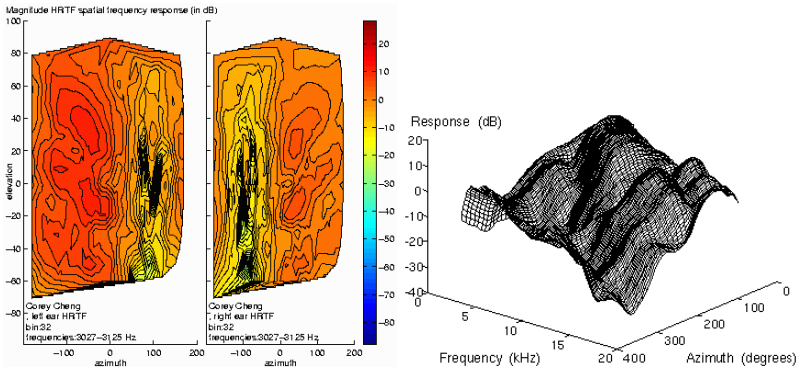


Figure 4.1: Two examples of visualization of HRTF.

4.1.1 The role of HRTF in 3D sound rendering

In the context of computer graphics, audio has always been considered as a minor resource to provide realism, but techniques like binaural rendering can greatly enhance perception. Binaural rendering is a technique which provides exact reproduction of the acoustic field over headphones. Unfortunately, realistic results can be achieved only by using individualized HRTF filters.

The head-related transfer function (HRTF) describes how a given sound wave input (parameterized as frequency and source location) is filtered by the diffraction and reflection properties of the head and pinna, before the sound reaches the eardrum and inner ear. Biologically, the source-location-specific prefiltering effects of the head and external ear aid in the neural determination of source location (see [26] for a comprehensive overview about psychophysics of spatial hearing). HRTFs are complicated functions of frequency and the three spatial variables (two examples of visualization are show in Figure 4.1); their shape can noticeably vary between subjects and it's closely related to the features of the head. If HRTFs are not accurately calculated, this could result in several perceptive problems due to localization artifacts (please refer to [17] for a comprehensive overview on 3D sound for Virtual Reality and Multimedia). Hence, the modeling of accurate individual HRTFs is a central issue in the context of audio rendering techniques. As soon as there is no convincing method to have HRTF calculated in a short time and with low input, the commercial application of binaural rendering (for example for DVDs and videogames) will be difficult.

To address this issue, several approaches have been proposed to model individualized HRTFs for 3D audio processing [68].

The current reference method consists in directly measuring HRTFs by placing mi-

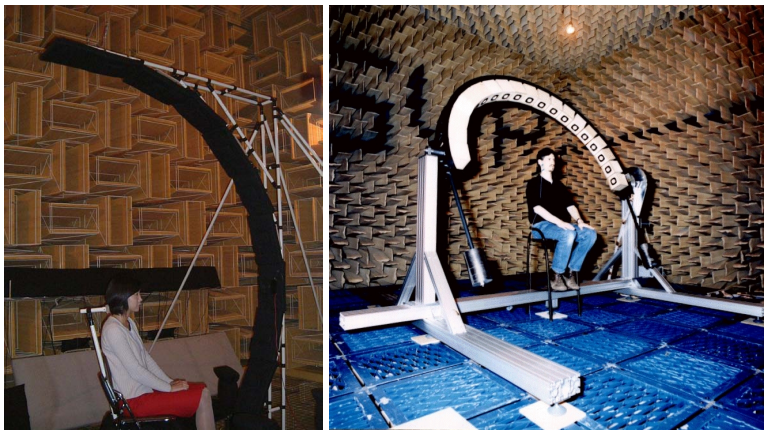


Figure 4.2: Two examples of anechoic chambers used for HRTF calculation.

crophones inside the ears of the subject [87], and then measuring the sound level perceived by them using a moving sound source. These measurements are usually conducted in an anechoic chamber (see Figure 4.2). While leading to improved 3D reproduction, measurement-based approaches usually require dedicated hardware and facilities. Moreover, the process can take up to several hours and is extremely uncomfortable for the user. Hence, there is actually no real applicability for the wide public.

Perceptually-oriented techniques use a series of short successive listening tests. They present the subject several sounds located at different target positions using various HRTF sets, and guide him in the choice of a composite set that best matches the targets [109]. This procedure might take up to 20 minutes. While ensuring that the results best fit to the user perception, accounting for the entire reproduction pipeline, it remains a challenge to design efficient protocols to quickly select the most appropriate matches.

A third group of approaches attempts to directly model an individualized filter starting from a generic HRTF set (e.g., a dummy head [90]) or by constructing analytical models of the various subcomponents of the HRTFs (e.g., pinna/head shadowing, shoulder reflection [57, 4]). These models can be driven by direct measurements of morphological parameters on the subjects, or by using photographs [90]. However, linking morphological parameters to changes in the features of the HRTF filters relies on correlation analysis between significant morphological features and spectral features of the HRTFs. This requires a large database and might not be fully reliable [90].

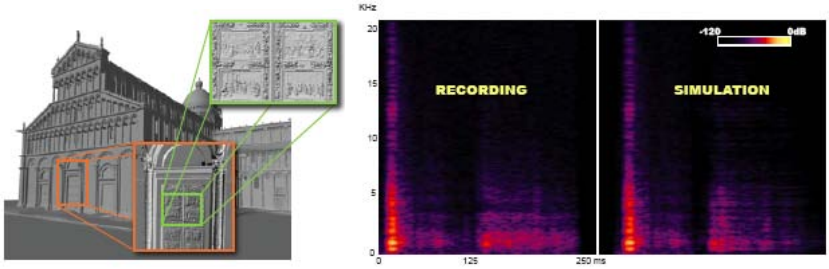


Figure 4.3: Left: the model of the Dome facade used for scattering calculation. Right, the comparison between real and calculated sound scattering respect to common sound source and listener position.

Finally, a last group of methods tries to simulate virtual measurements of HRTFs filters by running finite element simulations [86, 85] or ray-casting [6] on a 3D head model. This approach can be compared to virtual gonio-reflectometry, used to model BRDFs in computer graphics.
The main drawbacks to these methods are the prohibitive computation time, and the need of a closed and not very detailed 3D model of the head.
A collaboration between Visual Computing Lab and Reves Lab at INRIA brought to the creation of an alternative method for sound scattering calculation [134]. Instant sound scattering is based on the use of the Kirchoff approximation, with which it is possible to calculate the first bounce of sound scattering in a very short time. Thanks to a GPU implementation, the scattering calculation can be obtained in minutes instead than in hours. The results of the comparison between real measurements and scattering on models shows that the approximation brings to almost unnoticeable differences in scattering plots.

As shown in Figure 4.3, the scattering calculation for a sound source and listener position was compared to a on-site measurement in equal conditions. The plots of the two scatterings appear very similar, with the calculation taking only a few minutes to complete.
Another advantage of this method stands in the fact that the geometry quality doesn't need to be extremely "clean" to obtain good results: the calculation works regardless of holes or badly triangulated surfaces.
Nevertheless, in order to apply this method to HRTF calculation, it is necessary to provide an accurate model of a 3D head. In fact, as already stated, the HRTF profile can change a lot between subjects: this seems to be closely related to the features of the head. In particular, the primary role in determining sound perception seems to be associated with the peculiar features of ears, while a secondary contribution

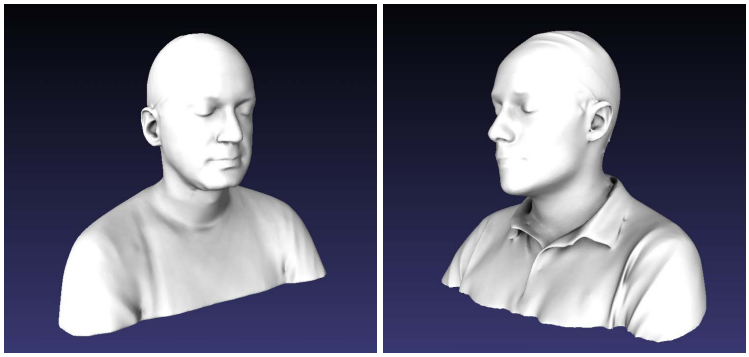


Figure 4.4: Screenshots of two head models obtained via 3D scanning

is assigned to the size of the main features of the head (nose, chin, head width and height...). Consequently, if the 3D head model is not accurate enough, especially in the ears zone, the produced HRTF could not ”fit” to the subject.
Hence, the main problem is to be able to provide a complete and accurate head model: a reliable technology to do this is 3D scanning. Although human heads presents several issues related to acquisition (the user needs to stay as still as possible throughout the scanning, hair is impossible to acquire), it is possible to obtain good quality head models (see Figure 4.4) with a scanning time of nearly 40 minutes and a post-processing time of 2-3 hours.
Unfortunately, even 3D scanning is not compatible to a possible application of HRTF calculation for the wide public, because of the cost of the hardware and the need of an experienced operator to provide a good model.
In order to be able to have a fast and automatic calculation of HRTF, it’s necessary to start from a much simpler set of input data. The next Section will present and automatic method to produce an accurate model of head starting from a very simple input: a set of uncalibrated photos.

4.1.2 Head model reconstruction from images

In order to be able to really apply HRTF calculation (and consequently binaural rendering techniques) to the wide public, the solution was to select the smallest possible input and to create an automatic system.
Nevertheless, another important constraint was the necessity of having an accurate 3D model: in fact, the shape of HRTF is primarily related to the peculiar features of the ear, and secondarily to the main features (nose, chin, head size) of the face.

Hence, it was critic to be able to represent the ear features with high accuracy.
Ear biometrics try to individuate a model for external ear features: Iannarelli's one [80] is often cited as very convincing. Unfortunately, these biometry measures are not well-defined, so it's very hard to use them in automatic methods. Ears are mainly considered in the field of recognition and security by analyzing both 2D [83] and 3D [38, 39] ear data. Unfortunately the information extracted and used for recognition is usually not directly linked to geometric features (like curves or size of the ear).
Together with the works presented in Section 2.2, head and face models can be obtained in even simpler ways [75, 47], but the extracted geometry is really simple and in all cases the ear are not taken into account. A complete head model (with hair) can be obtained by [67] can generate accurate models, but a very complex acquisition apparatus (28 digital cameras and two projectors!) is needed.
Provided that no convincing model of ear features was present, and existing head reconstruction methods from small input were not accurate enough, it was decided to build a custom system which could be able to automatically build a 3D head model from a few images.
The final system had to be:

- Usable for large scale application, with the very low amount of input needed
- Completely automatic. Once the input data are collected, the 3D model reconstruction and the HRTF calculation must be performed with no further intervention.
- Fast and reliable: the system has to be structured to produce realistic results even in the case of low quality or incongruent input data.
- Designed for accurately reproduce the ears shape, but also able to recreate the main face features.
- Geometrically accurate: the accuracy in reconstruction had to be adequate to calculate a satisfactory HRTF.

In order to be able to reach these goals, the whole system was designed to employ different techniques at their best for this peculiar application. It was decided that the final model would have been obtained via the morphing of a starting dummy head, following the information extracted from the input images.
A scheme of the whole system is shown in Figure 4.5. There are four main components which intervene in the generation of the 3D model. The whole process can be roughly divided in two parts: the first one deals with the treatment of the input given by the user, and the extraction of feature-based data from photographs, in order to select the best 3D dummy to be morphed. In the second part, the work is performed

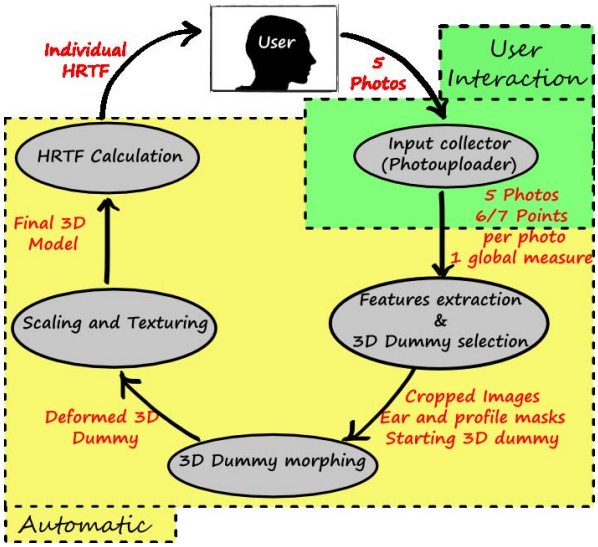


Figure 4.5: A scheme of the whole HRTF calculation system

on the selected 3D dummy mesh, in order to morph and scale it to resemble the geometric features of the head of the user.
Though, for sake of clarity, the scheme has been subdivided in several parts, only the first component (Input Collector) needs an intervention by the user. The rest of the system was designed to be completely automatic.

The components of the reconstruction system

The first issue to be addressed was: what is the lowest possible amount of work which can be asked to the user, if a robust, automatic, precise and fast system is wanted? Provided that digital imaging is the simplest way to encode needed information, several issues arose: how many photos? From which position? Are those data really sufficient?
One of the goals of the design phase was to ensure simplicity: no custom acquisition setup, no need of high quality cameras. The final choice, in order to try to cover the whole head, and to be able to infer 3D information, was to ask for five photos of the head, possibly taken from the same distance: one frontal, two profiles and two three-quarters portraits (see Figure 4.6 right). A small guide helps the user in taking the photos, by giving some simple hints, like: keep the same expression between the photos, remove carefully hair from ears, use flash if present etc.
Techniques to analyze human face photos can automatically extract a lot of features (eyes position, nose, profile...). Yet the aim of the work was to build a robust method,

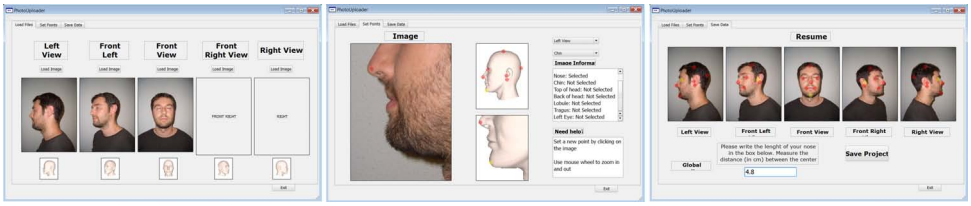


Figure 4.6: Screenshots of Photouploader: left, first tab for images upload; center, second tab for key-points picking; right, third tab for data resume and global scaling value insertion

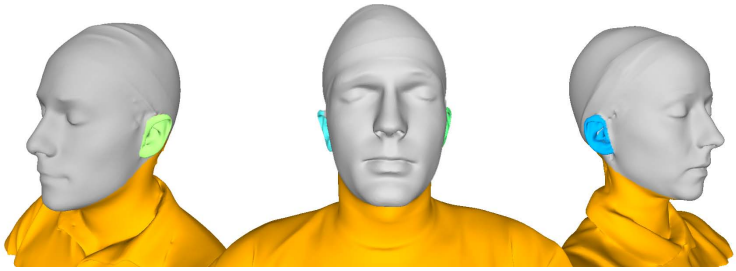


Figure 4.7: Three elements of the 3D dummy library

able to produce a good result even when particular face features (long beard, tatoos, peculiar haircuts...) were present, or when low quality images were uploaded. To do this, a small further contribution is requested to the user. In order to collect data in a user-friendly way, a simple tool called Photouploader was created. Photouploader was designed as web application, where the user can upload the images and is guided in the data provision.

Photouploader is divided in three tabs, shown in Figure 4.6. Once the photos are taken, the first tab guides the user in uploading the five photos. The second tab asks the user to indicate some key-points (6 or 7) on each photo. Dummy head images show where are the needed points in a very intuitive way. When all the required points have been selected, in the third tab the user is finally asked to insert a global scale measure (the length of the nose was chosen), and save the project into an xml file.

With this step the needed practical contribution from the user ends: regarding all the further stages, the system was designed to work automatically.

Once that input data are collected, the automatic model production process starts. The goal of the first element of the system is the selection of the best starting dummy from a library of 3D heads. The library of 3D heads is composed by ten models obtained via 3D scanning (see previous Section). These model were preferred to *artificial*

ones especially because the peculiar ear features were present, and the acquired people were different in age and gender. These models (250k triangles) represent both head and shoulders: they were used also as a reference for the validation of geometric accuracy and HRTF calculation. Moreover, as shown in Figure 4.7, each 3D model has differently colored parts, which will undergo different morphing policies.
As already stated, although face features are important and can't be ignored, the shape of the ears plays a key role in the final HRTF profile. So the dummy which best matches the ear features (extracted from images) is selected for morphing. Hence the first goal is to extract the external borders of both ears from images. The image is automatically cropped using two of the key-points provided by the user (lobule and tragus) as a reference. Cropped image is then scaled to 256x256 in order to remove high frequency details. Both edge intensity and orientation are calculated using the method proposed in [83]. The ear shape is extracted by finding a seed point near to the lobule key-point provided by the user, and then following the ear edge taking into account both intensity and direction of edges.
Several other controls (since the shape of ears is generally similar between subjects, it's possible to apply constraints on the final shape and continuity of extracted line) are added in order to deal with cases where the edges are hard to find or non present (i.e. hair over some part of the ear, earrings). Two examples of external mask extraction are shown in Figure 4.8. In the second one, even though part of the ear is covered by hair, extracted mask is very similar to ear shape. External masks are used both for dummy selection and as an input for morphing. The dummy selection is performed by analyzing each of the ears of the library models (Figure 4.9). A model of perspective camera is used and a rigid alignment is performed, by modifying extrinsic camera parameters so that the external borders of the 3D ear are best aligned to the extracted mask. Hence, a low accuracy morphing (see next Subsection) is applied on each aligned 3D ear model, so that it is slightly deformed to fit both the external mask and the internal features. After this operation, a similarity measure between a rendering of the 3D ear and the extracted image (based on the position of feature edges) is calculated, and the most similar dummy is selected. Camera parameters associated to the alignment are stored as well.
A similar approach to the one used for ears is applied to profile and frontal photos to provide best camera data for head deformation (Figure 4.10). The image is cropped and scaled to 256x256, using the the key-points provided by the user as a reference. Then, in a similar way to the ear extraction, the external mask of the face is created, by using the line extracted after having analyzed the features of the image. Where no information can be found on the image, the key-points, provided by user, help to define a line which resembles head shape. Once a mask is extracted, a rigid alignment which takes into account both extracted mask and user-provided key-points deter-

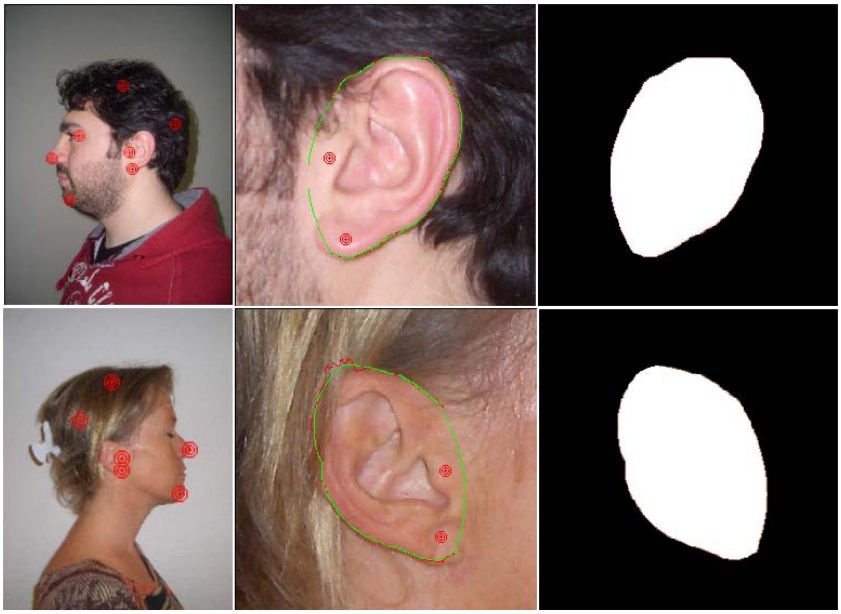


Figure 4.8: Two examples of ear external border extraction

Figure 4.9: An example of ear selection: starting image, dummy ear camera position before and after alignment, ear shape after low accuracy morphing

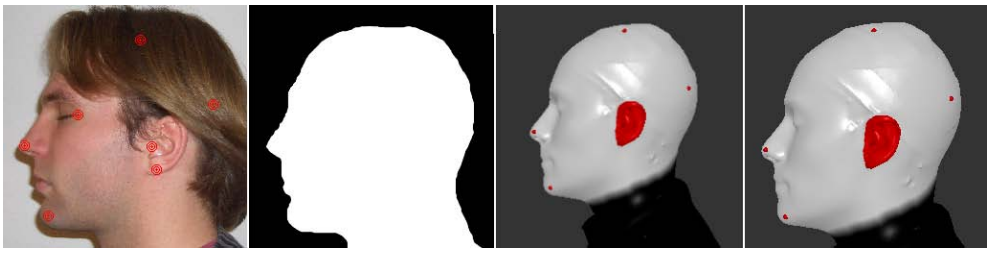


Figure 4.10: An example of head alignment: starting image, extracted mask, dummy head camera position before and after alignment

mines the best camera parameters for head morphing.
The final output of this stage includes: ten extracted images (four for the ears, six for the face) and an xml file containing all the necessary information needed for the following steps (new coordinates of the points set by the user, name of selected dummy, camera parameters associated to each view, global scaling value). This input set is passed to the 3D Morpher, which is the core of all the system: it applies a peculiar deformation to the dummy 3D mesh.
Using the set of cameras which defines the alignment of the dummy model with respect to each image, a set of viewport-dependent 2D-to-3D model deformation is calculated. The set of deformations is then combined to morph the dummy model to its final shape. The entire morphing process can be subdivided into the following steps:

Single View Head Deformation Three energy-driven deformations, one for each photo (right and left profile, frontal) are calculated. Each one tries to match the geometry with respect to a single point of view (*view dependent deformation*).

Global Head Deformation View dependent deformations are merged (according to camera positions and orientations) to a global smooth deformation.

Ear deformation Ears, which have been preserved in previous steps, undergo an accurate deformation using close-up ear cameras and images.

Morphing the geometry to match an input image requires the computation of a mapping between the photo and a rendered image of the dummy head (taken from the associated rigid-aligned camera position). This mapping operation is usually referred as *warping* in literature.

Warping Computation

Our warping function is an extension of [108] and [118]. It defines a 2D image deformation, that is then applied to the 3D model. It can be summarized as follows : feature images are processed with an edge detector, a stack of feature images is created by downsampling; finally automatic multilevel feature matching defines image deformation (details in [118]).

The original energy function (which has to be minimized in order to compute best deformation) was modified by adding a term $Kp = \sum_{P_i \in keypoints} |(P_i(photo) - P_i(model)|$ that measures the sum of distances between the user-defined key-points on the input photo and the relative key-points on the dummy head (transformed to screen-space coordinates).

Energy function can be schematized as follows:

$$E = L2 + \alpha * J + \beta * Kp \tag{4.1}$$

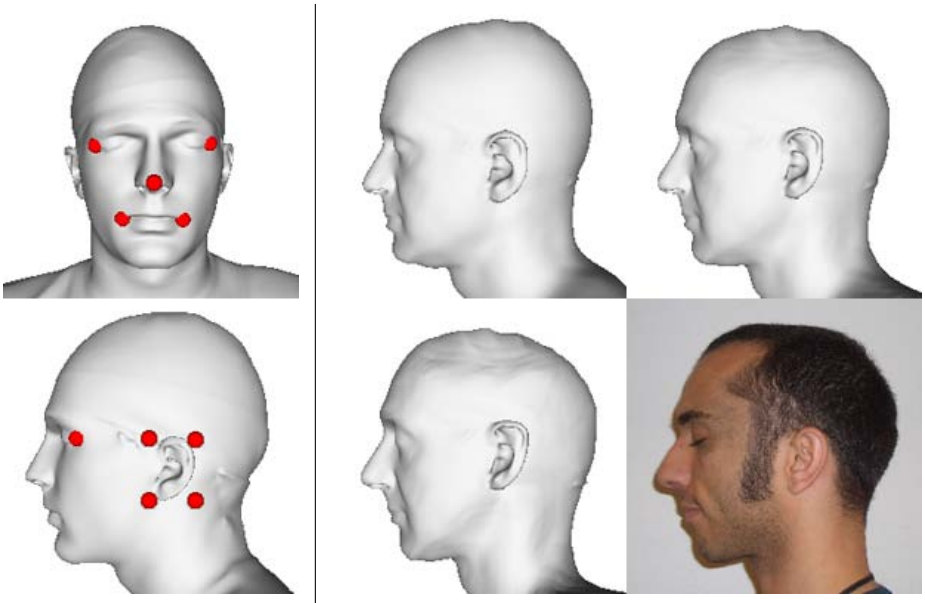
where $L2$ is the *per-pixel error* scalar feature strength and J is the *Jacobian term* which controls the smoothness of the warp field. In the system, the warping calculation starts from an 8x8 triangulated grid up to 64x64 (see [108] for details). The values of α and β are set respectively to 400 and 10000.

Once that the warping function is calculated, the displacement for each 3D dummy vertex is calculated by projecting the vertex on the associated camera plane, evaluating its warped position, and un-projecting it back to world space (without changing z-value).

The size of images used for morphing is 256x256. This value represents a good compromise between detail preservation and processing time: higher resolution images could be used, but the gain in detail would not justify the longer time necessary for computation.

Single View Head Deformation The subsequent phase is the warping between a rendered image of the dummy head and the input photograph, using associated camera parameters. In the original method, both external and internal features would be taken into account for deformation. But in a real scenario, head photographs could reveal strong sharp features that are difficult to be represented geometrically (such as beard, eyebrows..). Furthermore, peculiar lighting environments could lead to incorrect edges warping. So the deformation is applied using only the binary masks which define the external profile of the head.

The internal features of the face are then deformed by fitting a group of key-points associated to those indicated by user. Figure 4.11.1 shows the key-points involved in head warping: for frontal deformation we use 5 key-points: two for the eyes, one for the nose and two for the mouth; while for lateral deformation we chose to use one



(1) (2)

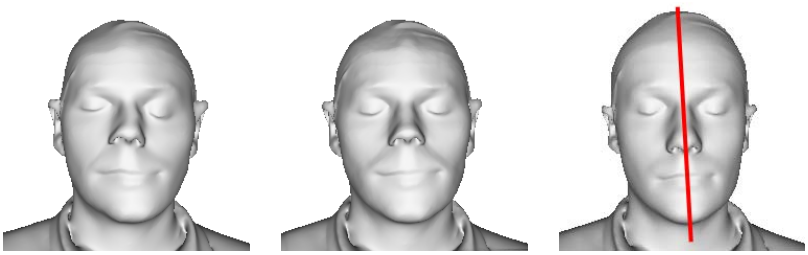
Figure 4.11: (1)Keypoints used for frontal and lateral head deformation. (2)Lateral head deformation sequence.

side eye constraint plus a set of four points around the ear. These points define the bounding box of the ear, so that it is preserved for a latter deformation (see later). Sequence 4.11.2 shows the deformation process involving the dummy mesh, using one lateral image.

Moreover, the frontal warping is controlled via *symmetrization*. Because of possible non-symmetric head contours extracted from frontal photo or non perfect input image (i.e. tilted or slightly rotated head), simple warping can produce asymmetric head shapes (see Figure 4.12.1). To overcome this problem the warping is symmetrized as follows:

- A symmetrization line on the rendered image is established, so that the rendered image is divided into two subspaces (Figure 4.12.3). The symmetrization line is defined as the line passing through the nose key-point and the point in the middle of the eyes key-points.

- A mapping $Mirr(x, y)$ between the two regions of the rendered dummy is obtained, by mirroring over the symmetrization line.



(1) (2) (3)

Figure 4.12: (1 and 2) Non-symmetrized versus symmetrized ; (3) Symmetrization line of undeformed model.

- The warping of mirrored points is averaged:

$$Warp(x,y) = \frac{Warp(x,y) + Mirr^{-1}(Warp(Mirr(x,y)))}{2} \tag{4.2}$$

Figure 4.12 .2 shows the effect of warping symmetrization.

Global Head Deformation At this morphing stage, each vertex can be translated in three different ways (one for each viewpoint). These three camera plane warpings are unified to a single smooth deformation as follows:

- Lateral deformed positions are unified through a weighted sum (weights decrease proportionally respect to ear distances)
- Unified lateral and frontal deformations are summed by assuming they are perpendicular, so that displacements in x- and z-axis are independent: the final displacement in the common direction (y-axis) is a weighted sum of the two contributions.

Ear deformation Accurate ear deformation is key for the final quality of the results: in this case both internal and external features extracted from images can be used to compute the deformation (Figure 4.13). The morphing sequence is organized as follows: 3D ear rendering is morphed to fit the external ear silhouette, then an additional warping (using the feature stack) matches the internal features.
The colors of input ear images are previously modified in order to match the histogram of frequency spectra of the rendering of the dummy model (see [108] for details): this further improves final deformation.
Scaling is one of the key issues about the accuracy of reconstruction. If the size of the model is incorrect, the computed HRTF will be wrong. Scaling is performed using

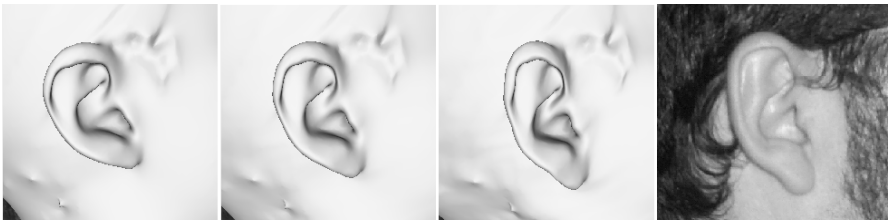


Figure 4.13: Example of ear morphing sequence.

Figure 4.14: Two results of processed heads.

the measure provided by user with the Photouploader (see above), which is the nose length. It's sufficient to scale the model by the ratio between this value and the nose length on the model (which can be calculated automatically by setting key-points which are translated accordingly to morphing). This leads back the geometry to its real measures. After a final translation to set the model ears position on the X axis (so that all the produced models are aligned in the same way), HRTF can be automatically calculated, and the process is complete.
A further feature of the system is the possibility to texture the obtained head model: the input photos are deformed to match more precisely the geometry (essentially by applying the inverse warping previously explained). Shown colored models are obtained by projecting warped images using TexTailor.

Results and validation

Two results of the application are shown in Figure 4.14. The entire system proves to be robust and quite fast: the overall time needed to produce the final 3D model, from input collection to model saving, it's less than ten minutes.

	Average	Maximum	Variance
Nose	11.3	22.1	0.89
Chin	10.8	21.2	0.03
Left Eye	9.8	16.6	1.22
Right Eye	8.1	11.4	0.46
Left Mouth	11.4	22.2	2.83
Right Mouth	9.8	21.5	1.69
Left Lobule	8.4	13.5	1.89
Left Tragus	6.7	11.5	1.04
Right Lobule	8.3	14.0	1.30
Right Tragus	7.3	10.8	1.83

Table 4.1: Distance in mm between key-points of scanned and reconstructed model

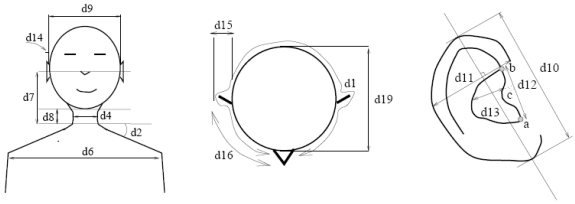


Figure 4.15: Some of the measures proposed by [90]

Although the visual resemblance of the obtained model is usually satisfactory, the main goal of the entire system was to be able to guarantee sufficient geometric accuracy. For this reason some tests were performed to compare eight models obtained from photos to their corresponding laser scanned models.
A sub-millimetric precision in geometry reconstruction from photos is a results which is possible only under very particular and controlled conditions. For this application, since the input is provided by the user, and the starting dummy can be very different from the final result, the main goal is to be able to reproduce head features as much as possible. Hence, instead of using purely geometric comparison tools like [43], the results were compared by taking into account two sets of measures, which could represent head features and their influence on the HRTF profile. The first set was the position in space of several key-points, picked on both model. Results of comparison are shown in Table 4.1. The average error is usually less than 1 cm and the variance of data is not big. These values can be considered as satisfying, especially considering that the input data is solely two-dimensional and the scale factor can introduce inaccuracies.

The second set of measures was extracted from [90], where several ear and head measures were statistically analyzed in order to find which ones were more related to the changes in HRTF profile (Figure 4.15). The set of 3D models was compared

	Average	Maximum	Variance
Neck (d4)	5.6	15	0.10
Head Size (d9)	5.9	14.5	1.30
Head Size (d19)	3.2	6.7	0.47
Ear size (R) (d10)	3.7	6.3	0.50
Concha size (R) (d12)	1.8	3.4	0.04
Concha size (R)(d13)	0.8	1.3	0.005
Ear size (L) (d10)	3.8	8.2	0.58
Concha size (L) (d12)	2.1	3.8	0.33
Concha size (L) (d13)	1.1	2.5	0.24

Table 4.2: Difference in mm between distances indicated in [90]

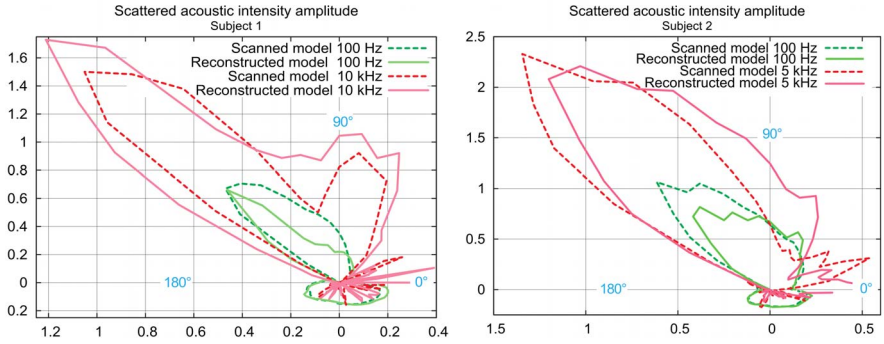


Figure 4.16: Two examples of polar plots for measurements on couples of scanned-reconstructed 3D heads.

using six measures (three for the head, three for the ears) which are indicated as very important in the conclusions of this work. Results are shown in Table 4.2. The difference between distances is often less than 5 mm; in particular, the internal features (concha size) seem to be preserved accurately.

Finally, a preliminary validation of HRTF simulation on 3D models was performed on couples of laser scanned and reconstructed 3D heads of the same subject.

To simulate HRTFs corresponding to the reconstructed geometry, the approach proposed in [134] was used.

The computation was performed to obtain the scattered field captured by two virtual microphones placed at the entrance of the left and right ear canal (90°), less than 5 millimeters away from the surface of the mesh. The field was produced by a source wave rotating around the head in the horizontal plane, at a 2 meter distance from the center. 8192 frequencies from D.C. and 22kHz were simulated. The hardware accelerated approach computes an HRTF pair for a given source position in about 200 sec. on a GForce 7600GT.

Computed data are only a subset of a complete individual HRTF, but they provide enough information for a preliminary comparison. Two polar plots of left ear intensity-amplitude for our reconstruction approach compared to a laser scanned model are shown in Figure 4.16. The intensity-amplitude variation features are aligned for the reconstructed and scanned models. In particular the maxima are reached at 125°, and rear features match between 180° and 360°.
It can be stated that obtained data from preliminary calculations prove to be encouraging for a future use of 3D reconstructed models for HRTF calculation. A further stage of validation between reconstructed and measured in anechoic chamber HRTFs will provide more information. Moreover, it will be possible to further investigate the importance of the head features (ears vs. face, possible contribution by shoulders) to improve results and possibly further simplify the system.
In conclusion, a peculiar use of a set of images to modify a 3D geometry was shown. In this case, the images are implicitly registered to the 3D model using the key-points provided by the user. Then each image is used to generate a 2D-3D deformation of a starting dummy, and the deformations are combined to create a final morphing which produces a 3D model. Differently from all the applications shown in previous Chapter, in this case the role of uncalibrated images was to provide information for geometry modification.

4.2 Hole filling using registered images

While in the previous section uncalibrated images were used to color and deform geometry, in the next application images will concur in generating new surfaces.
The input data for the method are the same of the previous ones: a 3D model and a set of images. Hence, the aim is not to generate geometry only from 2D data, but to help eliminating an annoying geometric artifact: the hole.
A 3D model resulting from a scanning operation presents several types of artifacts: triangles overlaps or intersections, singular vertices, zero area faces, complex edges, non manifold faces or vertices, holes. Most of these artifacts can be easily removed by applying simple algorithms; others present more difficulties.
In particular, a 3D scanned model is very rarely a closed model. This is due mostly to the fact that some parts of the geometry are very difficult, if not impossible, to acquire. Nevertheless, it is not unusual to have parts of the model which can be taken by a photo, but cannot be acquired by the scanner due to the angle between the camera and the laser emitter, or because of the fact that it is impossible to find the position of the scanner which permits to ”close the hole”.
Usually, most of the holes in a 3D scanned model are quite small, in the order of a

few edges. In this case, the holes can be filled without problems using trivial methods. But it is possible to find bigger holes (especially with the presence of handles or cavities). In this case, more complex techniques are needed to fill the hole with a ”plausible” geometry.
The hole filling methods can be roughly divided in two groups: volumetric and surface oriented.
Volumetric approaches put the model in a grid (which can be regular [113] or adaptive [22, 84]) and reconstruct the surface by filling the holes using Marching Cube or Dual Contouring. These methods are very robust, but they act a remeshing of the geometry (this could be something not needed or wanted) and produce watertight meshes (while only some of the holes may need closing).
The other approach is a local one, so that one hole at a time is treated. The main idea behind this group of methods is to reconstruct the geometry by analyzing the shape of the surface near the border. The strategy to reconstruct the geometry can be very simple [116] or mathematically more complex [36, 96]. Other approaches analyze the features of the geometry of the whole model, and fill the holes using a similar zones which were acquired [18, 112]. The main issues related to these methods are that they are less robust (ad-hoc strategies are usually needed to handle with isles of complex topologies) and that the computational complexity does not increase linearly with the number of border edges.
Finally, a common issue between all above cited methods is that the geometry created to fill the hole could be ”invented”, because there is no real data about the zone which was not acquired. This could be a problem in several application fields, like Cultural Heritage, where it could be preferable to preserve the hole instead of risking to introduce artifacts by filling with a wrong surface.
In the case of large holes, the idea should be to be able to infer some data about the real shape of the missing geometry: since scanning again the part can be too complex if not impossible, an easy way to convey information could be taking images. Moreover, most of the times the parts that cannot be covered by a scanner can be framed easily using a digital camera.
This is the main idea of the work by Hsu [79], presented in Section 2.2: unfortunately the presented results are not convincing (because they are obtained from renderings and not from real images) and the problem of interpreting the images stands again.
In this Section a method which uses images to help filling holes is presented. A further element of the system is a laser emitter which projects on the object a pre-defined pattern. In this way, when the images are aligned to the model and the laser position is estimated, it is possible to infer geometric information which helps to reduce the problem of filling a big hole in a series of small hole filling operations. In practice, a simple 3D Scanner is created, where the errors introduced by the estimation of posi-

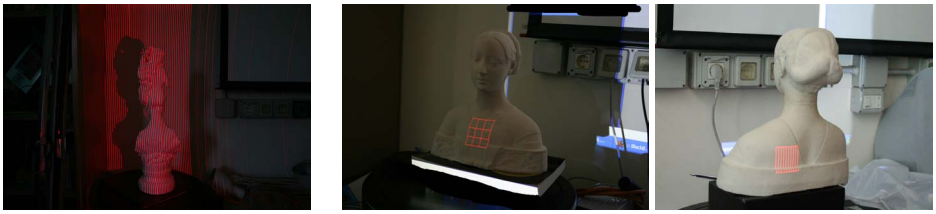


Figure 4.17: Three possible projected laser patterns

tions are corrected using the redundant information (the known geometry).

4.2.1 The elements of the system

The original idea behind the proposed system [138] was to create a low-cost solution for hole filling, with a laser diode fixed to a digital camera, so that the position of the emitter could have been known in advance. Unfortunately, due to the still high costs of custom pattern creation lenses for laser diodes, it was decided to create a more general system, where the position of the pattern emitter is estimated.

There were several possible shapes of the emitted pattern to be used (see Figure 4.17): the simple one shown in Figure 4.17right (a square withe eight internal vertical lines) was chosen, although the system can be easily adapted to any other peculiar shape, provided that enough data can be extracted for the emitter position estimation. The hole reconstruction framework can be divided in three main parts: image processing, laser projector position estimation and geometry reconstruction.

The input dataset is made by a 3D model, an image depicting the pattern (Figure 4.18) and the camera data which define the alignment of the image (obtained with TexAlign [65]).

Image processing. The first operation is the extraction of the information about the pattern from the image. It is thus necessary to apply image filters in order to extract the red lines from the image. The operation would be extremely simple if an ad-hoc filter were applied on the objective of the camera, but with the aim of having a low-cost system, it was decided to create a process for the extraction from a non-filtered image.

The first operation, which extracts the red laser pattern, is the application of a common Sobel filter [122], applied on each color canal, paying particular attention to the red one. After having applied the extraction filter, the work area is automatically extracted by finding the corners of the pattern. Then, a simple erosion filter is applied to reduce the extracted pattern. This brings the width of the red lines to a single pixel,

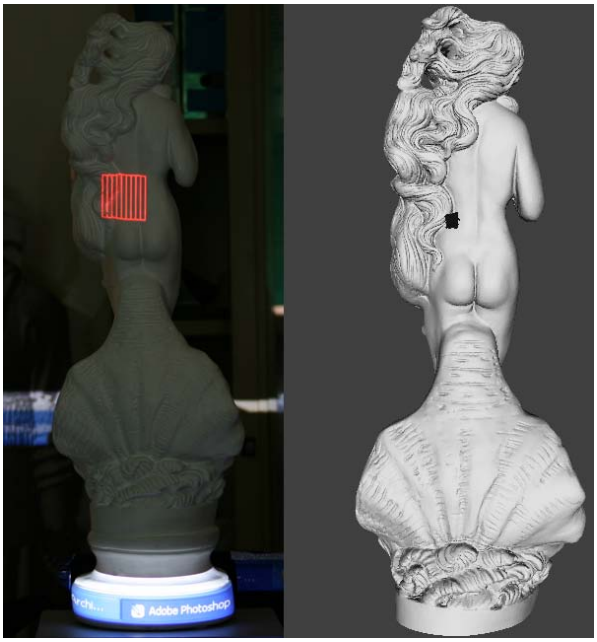


Figure 4.18: An example of model with hole and image with projected pattern

so that the subsequent calibration and geometry reconstruction operations can be applied more easily. An example of an extracted grid before and after erosion is shown in Figure 4.19.

Finally, the user is asked to manually indicate the position of the four corners of the pattern. It was chosen to keep this operation semi-automatic, due to the fact that a correct posing of the points is key to have an accurate projector estimation, as it will be shown in the next paragraph.

Estimation of pattern projector position. During this important phase (which would not be needed in the case of a projector fixed to the digital camera) the position of the pattern projector is estimated starting from the knowledge of the shape of the projected pattern and the information taken from the extracted pattern on the registered image.

A model of the projected pattern can be easily obtained with the measurement of the size of the grid when projected on perpendicular wall at different distances.

Then, a set of points p_1,...p_n is extracted from the border of the extracted pattern, starting from the four corner points indicated previously by the user. Their corresponding projection p'_1,...p'_n on the 3D model can be easily obtained using the registered image. Hence a minimization algorithm (an implementation [100] of the

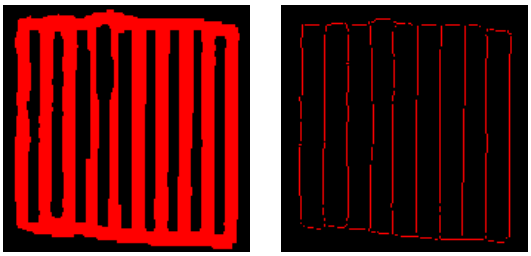


Figure 4.19: An example of an extracted laser pattern before and after erosion filter application.

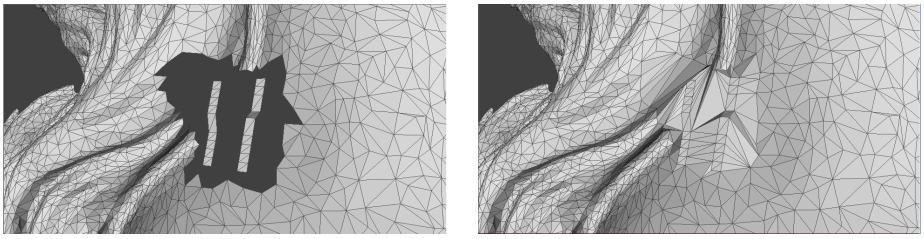


Figure 4.20: Left: the triangle strips generated using the image. Right: the rough hole filling

Levenberg-Marquardt method) is applied to fit the 3D position of the selected points to the planes generated from the model of the projected pattern. This brings to the estimation of the pattern projector position respect to the model.
The accuracy of the position estimation is key for the final quality of the reconstructed geometry. Hence, when laser diodes with diffracting lenses are available at lower prices, it will be possible to fix it to the camera so that no calibration will be needed.

Geometry reconstruction. Once that camera and projector positions are know, it is possible to calculate the corresponding 3D position of each pixel in the image. This is obtained with the usual technique of the triangulation. A 3D point is generated for each pixel of the extracted pattern which does not already map on the geometry. The produced points are used to create a series of strips (Figure 4.20 left) which split the original hole in a series of smaller holes. This is one of the main strong-points of the system: a complex problem is split in several simpler problems; moreover, the information used for creating the strips is extracted from the images, and it is a quite exact representation of the real geometry of the object. Hence, no new information until this point is ”invented”. Since the remaining holes are sensibly smaller than the starting one, the risk to introduce false data is smaller.

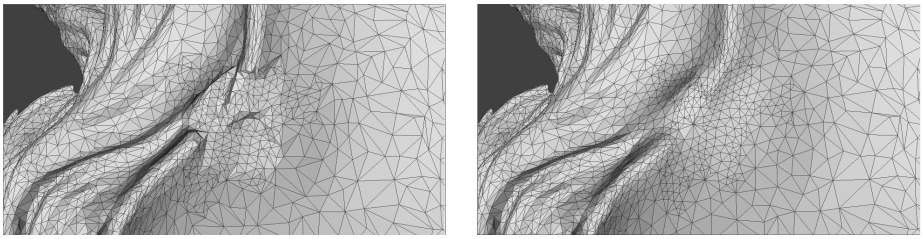


Figure 4.21: Left: the refinement step. Right: the smoothing step

The remaining steps of the hole filling procedure are the usual ones: a rough hole filling is applied to the mesh (Figure 4.20 right), then a refinement step (Figure 4.21 left) brings the size of the newly created triangles to the medium size of the triangles of the original mesh.

Finally, a smoothing step (which involves also some of the geometry of the border of the original mesh) produces a final surface (Figure 4.21 right) which is nearly indistinguishable respect to the rest of the model.

4.2.2 Testing and discussion.

In order to analyze the performances of the system, a test set of some 3D scanned objects was prepared. The 3D models were obtained from a group of medium size (50-70 cm height) statues (see Figure 4.22). Some holes where artificially created on peculiar parts of the models, in order to be able to compare the reconstruction with the original geometry. Then, some images of the pattern generated by a projector on the real objects were taken, and the holes were filled using our method.

Moreover, the results of our system were compared with the results obtained with two state-of-the-art hole filling methods: *Ear-Cut* [116], which is a simple and fast triangulation method, and Leipa [96], which is a more complex solution which integrates also the remeshing and smoothing of the reconstructed surface.

The results of two of the tests are shown in Figures 4.23 and 4.24. In the first case, while *Ear-Cut* and *Leipa* create smooth surfaces and loose the original crease of the model, our method is able to partially recover its real shape. In the second case, all the methods are able to produce convincing results, although *Ear-Cut* produces a low-quality triangulation.

Additionally to the visual results, a numerical test to measure the accuracy of the methods was performed. The Hausdorff distance, which is a very good metric for the differences between 3D models, was calculated between the original model and the

Figure 4.22: Two of the statues used for testing the proposed method.

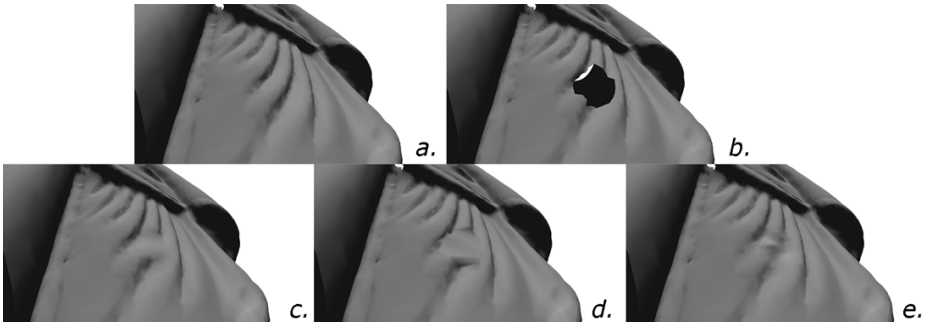


Figure 4.23: Test n.2: a) Starting model. b) Model with hole c) Hole filled with *Leipa* d) Hole filled with *Ear-cut* e) Hole filled with our method

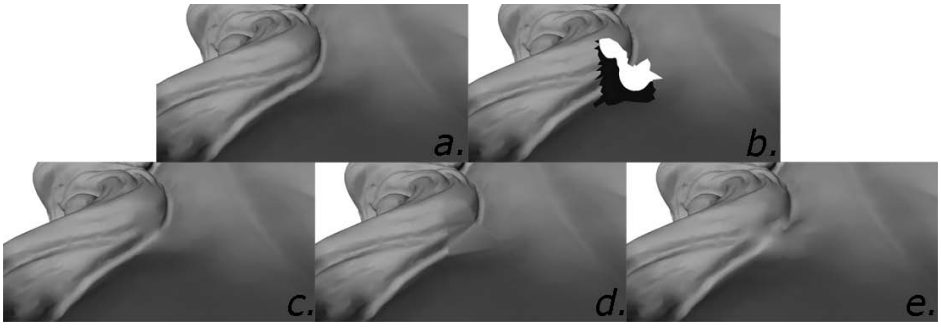


Figure 4.24: Test n.4: a) Starting model. b) Model with hole c) Hole filled with *Leipa* d) Hole filled with *Ear-cut* e) Hole filled with our method

Method	Average	Max	Rms
Test 1			
Leipa	0,002083	3,284013	0,052673
Ear-Cut	0,000593	0,687241	0,01462
Our Method	0,001201	1,708783	0,0295663
Test 2			
Leipa	0,515288	33,177967	3,379488
Ear-Cut	0,000553	0,79688	0,015343
Our Method	0,000222	0,642542	0,007969
Test 3			
Leipa	0,497988	33,177967	3,32634
Ear-Cut	0,00122	1,67301	0,027455
Our Method	0,000405	1,048663	0,013196
Test 4			
Leipa	0,002006	3,284013	0,051815
Ear-Cut	0,000294	0,485467	0,008804
Our Method	0,000493	1,919416	0,018686

Table 4.3: Analysis of the Hausdorff distance respect to the original model.

results of all the methods. The outcomes of this calculation, applied on 4 test cases, are shown in Table 4.3. From the analysis of the data, several consideration can be made:

- The geometric distance of the surface generated by Leipa method is always the highest: this is probably related to the remeshing and smoothing step. These operations produce pleasing shapes for the final surfaces, but they also introduce errors in the final result.
- The quality of Ear-Cut and our method results is comparable. Nevertheless, the visual appearance of Ear-Cut results is generally of low quality, due to the rough triangulation obtained. The use of remeshing and smoothing techniques to improve it would probably introduce severe errors in the final reconstruction.
- The quality of our method results is quite good also in terms of numerical comparison. Another very interesting observation is that the remeshing and smoothing phases described in the previous Subsection are almost equal to the ones used by Leipa. In our case, the error introduced by them is very low. This an interesting indication about the fact that the data obtained using the projected pattern represent the real geometry of the object.

In conclusion, the preliminary tests show that our method outperforms the other state-of-the-art algorithms, especially in terms of geometric accuracy. This can be very important to be able to guarantee an accurate reconstruction of the missing parts of a 3D model. The system proves to reliable and automatic, especially considering the fact that the input data is represented by an easy-to-obtain photo of the real object.

Chapter 5

Conclusions and future work

The goal of the methods presented in the context of this thesis is to use the impressive amount of information collected with digital cameras to enrich the geometric description of a scene.
It is very hard to extract data from a set of images, due to both amount of data encoded (up to 10-12 MPixels per image) and the complexity of the depicted scenes. Several semi-automatic approaches can generate simple 3D models starting from set of uncalibrated images. Nevertheless, if some information about the objects in the scene is known in advance, it is possible to obtain very good results in the analysis of the 2D data.
Hence, images can be used to enhance the quality of already existing 3D models: the most straightforward application in these field is the projection of color information on the geometry. But it has be shown that images can be used also to modify the 3D data by morphing the existing geometry, or creating data where the geometric sampling is incomplete.
The color projection issue has been quite extensively studied in the last few years, but several open problems still lack of a convincing and generic solution. Other kinds of applications are still under development, and fewer examples can be found in literature.
The next subsections will provide a resume of the main results obtained, together with an overview of the possible future work for any of the proposed fields of application of uncalibrated images on 3D models.

5.1 Color acquisition and visualization

The color acquisition, projection and visualization on 3D models is still a quite open issue in the context of Computer Graphics. While the solution proposed in Chapter 3 are very robust and useful, some bottlenecks are still present in the processing pipeline.
Almost every practical case can be faced with a combination of the proposed solutions, but no existing approach can be applied both to a low quality and to a very detailed 3D model. Nevertheless, several interesting directions of research can be spotted for all the issues related to color projection.

5.1.1 Image registration

The main bottleneck in color management stands in the registration process, because the most robust solution needs a massive intervention by the user. TexAlign, the solution proposed in this thesis, severely reduces the workload without limiting the generality of use. The main strong-points of the approach are:

- The possibility to set image-to-image correspondences and use them to align images. This reduces the effort in finding the links between the elements of the registration project.
- The analysis of the correspondences graph to guide the user to the completion of the registration process. The workload minimizer keeps track of all the connections between the elements, and suggests the steps to complete the registration with the minimum effort.
- The flexibility of the approach: it is possible to handle very complex datasets (tens of images, highly detailed 3D models).

The possible research directions to further improve these results can pursue two aims: a robust and scalable automatic registration method, or a even simpler semi-automatic methodology to cut down the user intervention to a few clicks.
The first direction is mainly related to the search for an automatic method to link the 3D features of an object and the information that can be extracted from images. The best solution could be to generate several possible views associated to an image, then to select the best starting candidate and finally to use some kind of metric to converge to the optimal solution. One of the main obstacles to this kind of approach was the processing time needed to consider the huge amount of possibilities (considering the combination of extrinsic and intrinsic parameters that generate the possible views of a digital camera). But the outstanding development of graphics hardware gives now the

possibility to generate and process several thousands of possibilities in a very short time: so, if a convincing metric to measure alignment can be found, it is plausible to be able to converge to the best solutions in minutes, if not in seconds.
Therefore, it is key to develop a measure of the alignment of an image. A promising candidate is the Mutual Information (MI), which has been mainly used in the context of medical data, due to the similar appearance of the types of data and their generally low resolution. But, exploiting again the potentials of GPUs, it is now possible to generate convincing renderings of a 3D model in a very short time. Hence, if the renderings are very similar to the real image, MI can be a very robust and fast way to obtain a convincing alignment. This idea is currently under implementation, and preliminary results are encouraging: the first experiments on a completely automatic system to align images on a 3D model will be started soon.
The second direction of research tends to the reduction of the workload of the user, so that with very few clicks he could obtain the registration of a lot of images. This can be exploited thanks to techniques to automatically find new correspondences. RANSAAC-like approaches [63] can be implemented so that the user has to set only a few correspondences, or so that the main effort could stand in the alignment of the first images, and then the rest of the set is registered with very low workload.

Finally, some work could be devoted to the refinement of the alignment: in a similar way to the range maps alignment of the 3D scanning pipeline, and automatic "fine registration" could follow the "rough registration" provided, with very few correspondences, by the user. To obtain a precise registration, instead of modifying the camera parameters, some small deformation of the images could lead to more precise projection. Techniques like the optical flow [78, 70] could eliminate small registration errors automatically.

5.1.2 Color projection

In the field of color projection, the provided solution is a tool called TexTailor, with which it is possible to project registered images on a 3D model. The main contributions of this solution are:

- The use of a per-pixel blending policy for color assignment: instead of selecting portions of images, all the elements which frame a point concur in the calculation of the color value to choose.
- The automatic calculation of quality masks, which implement several "common sense" concepts in order to assign higher weights to the best images during the calculation of the color value to assign. Moreover, this mechanisms

implicitly blends the possible differences (like uneven lighting) between images.

- The out-of-core implementation of the system solves the data processing issues, typical of most of the state-of-the-art solutions. Up to hundreds of images can be now projected on extremely detailed 3D models.

While TexTailor provides very good results, there are open issues whose solving could further improve the quality of the colored models. For example, the quality of the projected color is mainly related to the quality of the registration and of the original photo set (see next Subsection). Moreover, another important open issue is the visualization of color information.
Texture mapping is a compact and used method, and it produces realistic results even in the case of low detail 3D models. But its scalability is a main issue: this is related mainly to the absence of an automatic parametrization method for complex meshes.
Also the difficulty of integrating texture mapping in the framework of multi-resolution method is linked to the parametrization issue, since every level of detail would need a different parametrization.
On the other side, color per vertex quality is strictly related to the density of the samples in the model, leading to problem in complexity and disk occupation.
An hybrid interesting approach could stand in the application of point based rendering solutions, with the application of point splatting techniques which take into account the images to depict the appearance of the model. With this solution, the color visualization could be reduce to several very local parametrizations, avoiding the complexity problems above mentioned.

5.1.3 Color acquisition and artifacts removal

The artifacts generated by the original photo set are the most annoying parts of a colored 3D model.
Flash Lighting Space Sampling is a very simple solution to be able to correct color deviations and eliminate image artifacts in an automatic way. Moreover, no complex lighting setup is needed for the color acquisition. The main contributions of this solution are:

- The fact that the color correction space must be calculated only once in a camera lifetime.

- The use of a very simple type of light, now present in every digital camera model.

- The automatic color correction and artifacts removal, obtained as soon as the images are registered to the model.
- The extensibility of the idea of FLiSS to several other applications, from image enhancement to material estimation.

Nevertheless, some more work can be done to obtain even better results. There are two possible approaches to consider.
The first one is the creation of easy setups to acquire good quality images, possibly using types of light which are more ”reliable” than the flash. For example, some study on prototypal light-sets could lead to an easy (and possibly low-cost) creation of nearly diffuse lighting environments.
Since taking the photos is a quite difficult task, and in some cases it is very hard to work on the field, the other possible approach stands in the removal of the artifacts, before, during or after the color projection. This is mainly related to the knowledge of the lights used in the scene.
One idea could be to find an easy way to have information about the position and nature of the lights, using possibly some kind of simple device to acquire it. The other idea is to work on other simple controlled light environments, so that light position is know for every image of the set, and artifacts and color correction are applied automatically due to the previously known mathematical model of the lights.
Finally, the quality of the final rendering would gain in realism if it were possible to estimate the material properties of the object in an easy way. Although this is a very complex field of research, it could be possible to obtain simplified models of the materials using simple lights (like the flash).

5.2 Geometry modification

Two different approaches to modify an already existing geometry have been proposed in the previous Chapter:

- A method to morph an 3D model to fit the data extracted from a small set of images. The most interesting contribution is that the 3D morphing is obtained as the combination of three 2D morphings applied from pre-defined directions.
- A technique to fill holes in a 3D scanned model, using a digital camera and a laser pattern projector. In this case, a triangulation reconstruction is obtained in a 3D scanning fashion, but the acquisition of data is easy to obtain (some images of the objects are needed) and the accuracy is assured by the redundancy of data (the geometry previously acquired).

As already stated, geometry modification using images is a quite new field of research. Due to the not huge number of previous papers on the subject, it is not easy to spot precise directions of research.
Regarding geometry morphing, the goal could be to automatize the processes, and to be able to extract much more information from the images. This could be done more effectively by finding custom solutions for each application case: for example, in the case of face and head reconstruction, it could be possible to apply the very robust methods proposed in Computer Vision to extract very detailed information about the appearance of the subject.
Regarding possible approaches for geometry reconstruction and cleaning, the main goal should be to get rid of external devices (like the pattern projector used in Section 4.2) and be able to extract geometry information directly from images. An interesting direction could be using the redundancy between images and the concepts of the stereo vision to extract geometry information and integrate it with the existing 3D model.

In conclusion, the presented methods are interesting examples of use of unregistered images to enrich 3D models, but the amount of information which is not used is still remarkable.
The main advantage of these methods is that, differently from most of the Computer Vision approaches, the knowledge of the geometry of the scene helps greatly in the analysis of data. While, in the past, it was hard to obtain a geometric description of a scene, it is now easy and cheap (due to the decrease in the costs of devices and in processing time, or to the possibility to obtain geometry from images only) to have a satisfying (if not accurate) model of the depicted objects.
Hence, it is possible to think about using already existing methods to obtain better results, and there are chances to think about new methodologies.

In conclusion, the road to a fast and easy acquisition and visualization of reality is still long, but now it is not utopian to think of being able to recreate reality on the computer with a degree of realism and immersion which could reproduce the practical experience. This could be a real breakthrough not only for entertainment or pure visualization, but also and especially for the support to experts in very important fields, like Medical, Security and Cultural Heritage.

Chapter 6

List of publications

International Journals

M. Dellepiane, M. Callieri, M. Corsini, P. Cignoni, R. Scopigno, Flash Lighting Space Sampling, *Lecture Notes in Computer Science* Volume 5496, page 217-229, Springer (MIRAGE 2009)

M. Dellepiane, N. Pietroni, N. Tsingos, M. Asselot, R. Scopigno, Reconstructing head models from photographs for individualized 3D-audio processing, *Computer Graphics Forum*, Volume 27, Number 7, page 1719-1727 - 2008

M. Dellepiane, M. Callieri, F. Ponchio, R. Scopigno, Mapping highly detailed color information on extremely dense 3D models: the case of David's restoration, *Computer Graphics Forum*, Volume 27, Number 8, pp. 2178-2187 - 2008

T. Franken, M. Dellepiane, F. Ganovelli, P. Cignoni, C. Montani, R. Scopigno, Minimizing user intervention in registering 2D images to 3D models, *The Visual Computer*, Springer Verlag, vol. 21(8-10), 2005, pp.619-628

M. Corsini, M. Dellepiane, M. Callieri, R. Scopigno, Reflection Transformation Imaging on larger objects: an alternative method for virtual representations, *British Archeological Reports*, BAR International Series Archaeopress, vol. 1568, 30 Nov - 4 Dec 2006, Rome

International Conferences

M. Callieri, M. Dellepiane, F. Ganovelli, R. Scopigno, Multi-scale geometry acquisition and visualization: issues and techniques, *Advances in Remote Sensing for Ar-*

chaeology and Cultural Heritage management, Earsel, pp 214-217, 30 September - 4 October, 2008, Rome

M. Dellepiane, M. Callieri, E. Paribeni, E. Sorge, N. Sulfaro, V. Marianelli, R. Scopigno, Multiple uses of 3D scanning for the valorization of an artistic site: the case of Luni, *Eurographics Italian Chapter Conference*, Eurographics, pp 7-13, July 2008, Salerno

P. Cignoni, M. Callieri, M. Corsini, M. Dellepiane, F. Ganovelli, G. Ranzuglia, MeshLab: an Open-Source Mesh Processing Tool, *Eurographics Italian Chapter Conference*, Eurographics, pp. 129-136 ,July 2008, Salerno

I. Besora, P. Brunet, M. Callieri, A. Chica, M. Corsini, M. Dellepiane, D. Morales, J. Moys, G. Ranzuglia, R. Scopigno, Portalada: A Virtual Reconstruction of the Entrance of the Ripoll Monastery, *3DPVT08: Fourth International Symposium on 3D Data Processing, Visualization and Transmission*, June 2008, Atlanta

M. Dellepiane, M. Callieri, M. Fondersmith, P. Cignoni, R. Scopigno, Using 3D scanning to analyze a proposal for the attribution of a bronze horse to Leonardo da Vinci , *The 8th International Symposium on VAST International Symposium on Virtual Reality, Archaeology and Cultural Heritage*, Eurographics, page 117-124 - Nov. 2007, Brighton

N. Tsingos, C. Dachsbacher, S. Lefebvre, M. Dellepiane, Instant sound scattering , *Proceedings of Eurographics Symposium on Rendering*, Eurographics, pp. 111-120, Jul 2007, Grenoble

N. Tsingos, C. Dachsbacher, S. Lefebvre, M. Dellepiane, Extending geometrical acoustics to highy detailed architectural environments, *19th International Congress on Acoustics*, pp 1-8, June 2007, Madrid

C. Baracchini, M. Callieri, M. Corsini, M. Dellepiane, U. Dercks, D. Keultjes, C. Montani, M. Scognamiglio, R. Scopigno, R. Sigismondi, G. Wolf, CENOBIUM Cultural Electronic Network Online: Binding Up Interoperably Usable Multimedia , *Proceedings of Electronic imaging & the visual arts*, EVA 2007, March 2007, Florence

M. Dellepiane, M. Corsini, M. Callieri, R. Scopigno, High quality PTM acquisition: Reflection Transformation Imaging for larger objects, *The 7th International*

Symposium on VAST International Symposium on Virtual Reality, Archaeology and Cultural Heritage 2006, Eurographics, pp. 179-186, 4-7 Dec 2006, Cyprus

C. Baracchini, M. Callieri, M. Corsini, M. Dellepiane, U. Dercks, D. Keultjes, C. Montani, M Scognamiglio, R. Scopigno, R. Sigismondi, Starting the CENOBIUM project: the cloister of Monreale (Sicily) revealed, *The 7th International Symposium on VAST International Symposium on Virtual Reality, Archaeology and Cultural Heritage 2006*, Eurographics, pp. 100 -110, 4-7 Dec 2006, Cyprus

T. Franken, P. Cignoni, F. Ganovelli, M. Dellepiane, C. Montani, R. Scopigno, Assisting the user in image to geometry alignment, *Workshop Italy-Canada 2005*, Conference Proceedings, , 17-18 May 2005, Padova

R.Borgo, M.Dellepiane, P.Cignoni, Extracting Meta-Information From 3-Dimensional Shapes With Protg, *8th International Protg Confereence*, Conference Proceedings, 18-21 July 2005, Madrid

M. Balzani, M. Callieri, G. Caputo, P. Cignoni, M. Dellepiane, C. Montani, P. Pingi, F. Ponchio, R. Scopigno, A. Tomasi, F. Uccelli, Using Multiple Scanning Technologies for the 3D Acquisition, *3D Arch 2005*, Conference Proceedings, 22-24 Aug 2005, Mestre-Venice

F. Bellotti, R. Berta, M. Dellepiane, A. De Gloria, E. Ferretti, M. Margarone, A. Poggi, Implementing 3D Videos on digitally reconstructed artistic sites, *EVA 2004 London, Eleventh International Conference on Electronic Imaging & the Visual Arts*, England, Conference Proceedings, pp. 18.1-18.8, 26-30 July 2004, London

Various

Khoa-Van Nguyen, M. Dellepiane, I. Viaud-Delmon, O. Warusfel, Investigation of auditory-visual integration in VR Environments ,*8th International Multisensory Research Forum*, Poster Presentation, June 2007, Sydney.

M. Dellepiane, M. Callieri, Visualization of colour information on highly detailed 3D models, *ERCIM News*, n. 67, October 2006

Bibliography

[1] David low-cost laser scanner. More info and software download can be found on: http://www.david-laserscanner.com/.

[2] A. Abdelhafiz, B. Riedel, and W. Niemeier. Towards a 3d true colored space by the fusion of laser scanner point cloud and digital photos. In *Proceedings of the ISPRS Working Group V/4 Workshop (3D-ARCH*, 2005.

[3] Amit Agrawal, Ramesh Raskar, Shree K. Nayar, and Yuanzhen Li. Removing photography artifacts using gradient projection and flash-exposure sampling. *ACM Tr. on Graphics*, 24(3):828–835, 2005.

[4] V. Ralph Algazi, Richard O. Duda, Reed P. Morrison, and Dennis M. Thompson. Structural composition and decomposition of hrtfs. In *Proc. IEEE Workshop on Appl. of Signal Proc. to Audio and Acoustics (WASPAA'01), New Paltz, USA*, October 2001.

[5] D. Aliaga. Capturing, modeling, rendering 3d structures course. [web page] http://www.cs.purdue.edu/homes/aliaga/cs590g-04/, 2004.

[6] Sven Andres, Niklas Röber, and Maic Masuch. Hrtf simaulations through acoustic raytracing. Technical report, Otto v. Guericke Un. Magdeburg, Germany, 2006.

[7] Ian Ashdown. Near-Field Photometry: Measuring and Modeling Complex 3-D Light Sources. In *ACM SIGGRAPH '95 Course Notes*, pages 1–15, 1995.

[8] Marcello Balzani, Marco Callieri, Gianmatteo Caputo, Paolo Cignoni, Matteo Dellepiane, Claudio Montani, Paolo Pingi, Federico Ponchio, Roberto Scopigno, Andrea Tomasi, and Federico Uccelli. Using multiple scanning technologies for the 3d acquisition of torcello's basilica. In *Int. Workshop 3D-ARCH'2005 - 3D Virtual Reconstruction and Visualization of Complex Architectures, August 22-24, 2005, Mestre-Venice, Italy*, 2005.

[9] N. Bannai, A. Agathos, and R.B. Fisher. Fusing multiple color images for texturing models. In *3DPVT04*, pages 558–565, 2004.

[10] C. Baracchini, A. Brogi, M. Callieri, L. Capitani, P. Cignoni, A. Fasano, C. Montani, C. Nenci, R. P. Novello, P. Pingi, F. Ponchio, and R. Scopigno. Digital reconstruction of the arrigo vii funerary complex. In *VAST 2004: 5th International Symposium on Virtual Reality, Archaeology and Intelligent Cultural Heritage*, pages 145–154, December 2004.

[11] Clara Baracchini, Marco Callieri, Massimiliano Corsini, Matteo Dellepiane, Ute Dercks, Dagmar Keultjes, Claudio Montani, Matteo Scognamiglio, Roberto Scopigno, Roberto Sigismondi, and Gerhard Wolf. Starting the cenobium project: The cloister of monreale (sicily) revealed. In F. Niccolucci M. Ioannides, D. Arnold and K. Mania, editors, *The 7th International Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST 2006)*, pages 100–110, nov. 2006.

[12] Clara Baracchini, Marco Callieri, Massimiliano Corsini, Matteo Dellepiane, Ute Dercks, Dagmar Keultjes, Claudio Montani, Matteo Scognamiglio, Roberto Scopigno, Roberto Sigismondi, and Gerhard Wolf. Cenobium cultural electronic network online: Binding up interoperably usable multimedia. In Cappellini Vito and Hemsely James, editors, *Proceedings of Electronic imaging & the visual arts, EVA 2007 Florence,*. Pitagora Editrice, March 2007.

[13] K. Barnard, V. Cardei, and B. Funt. A comparison of computational color constancy algorithms. I: Methodology and experiments with synthesized data. *Image Processing, IEEE Trans.*, 11(9):972–984, 2002.

[14] K. Barnard, L. Martin, A. Coath, and B. Funt. A Comparison of Computational Color Constancy AlgorithmsPart II: Experiments With Image Data. *Image processing, IEEE Trans. on*, 11(9):985, 2002.

[15] A. Baumberg. Blending images for texturing 3d models. In *BMVC 2002*. Canon Research Center Europe, 2002.

[16] E. Beauchesne and R. Roy. Automatic relighting of overlapping textures of a 3d model. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '03)*, volume 2, page 166, 2003.

[17] Durand R. Begault. *3D Sound for Virtual Reality and Multimedia*. Academic Press Professional, 1994.

[18] G. H. Bendels, R. Schnabel, and R. Klein. Detail-preserving surface inpainting. In Mark Mudge, Nick Ryan, and Roberto Scopigno, editors, *The 6th International Symposium on Virtual Reality, Archaeology and Cultural Heritage*, pages 41–48, Pisa, Italy, 2005. Eurographics Association.

[19] J.-A. Beraldin. integration of laser scanning and close-range photogrammetry the last decade and beyond. In *Proc. IAPRS 35(5) Istanbul, Turkey*, pages 972–983, 2004.

[20] F. Bernardini, I.M. Martin, and H. Rushmeier. High-quality texture reconstruction from multiple scans. *IEEE Transactions on Visualization and Computer Graphics*, 7(4):318–332, 2001.

[21] Isaac Besora, Pere Brunet, Marco Callieri, Antoni Chica, Massimiliano Corsini, Matteo Dellepiane, Daniel Morales, Jordi Moyés, Guido Ranzuglia, and Roberto Scopigno. Portalada: A virtual reconstruction of the entrance of the ripoll monastery. In *3DPVT08: Fourth International Symposium on 3D Data Processing, Visualization and Transmission*, pages 89–96, June 2008.

[22] Stephan Bischoff, Darko Pavic, and Leif Kobbelt. Automatic restoration of polygon models. *ACM Trans. Graph.*, 24(4):1332–1352, 2005.

[23] Volker Blanz. Face recognition based on a 3d morphable model. In *FGR '06: Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition*, pages 617–624, Washington, DC, USA, 2006. IEEE Computer Society.

[24] Volker Blanz, Kristina Scherbaum, and Hans-Peter Seidel. Fitting a morphable model to 3d scans of faces. In *IEEE ICCV 2007*, pages 1–8, 2007.

[25] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3D faces. In Alyn Rockwood, editor, *Siggraph 1999, Computer Graphics Proceedings*, pages 187–194, Los Angeles, 1999. Addison Wesley Longman.

[26] J. Blauert. *Spatial Hearing : The Psychophysics of Human Sound Localization*. M.I.T. Press, Cambridge, MA, 1997.

[27] Louis Borgeat, Guy Godin, Francois Blais, Philippe Massicotte, and Christian Lahanier. Gold: interactive display of huge colored and textured models. *ACM Trans. Graph.*, 24(3):869–877, 2005.

[28] Alexander Bornik and Joachim Bauer. High quality texture reconstruction from multiple views. *Journal of Visualisation and Computer Animation*, 12:263–276, 2001.

[29] S. Bracci, F. Falletti, M. Matteini, and R. Scopigno. *Exploring David: diagnostic tests and state of conservation*. Giunti Editore, 2004.

[30] Lionel Brunie, Stéphane Lavallée, and Richard Szeliski. Using force fields derived from 3d distance maps for inferring the attitude of a 3d rigid object. In *ECCV '92: Proceedings of the Second European Conference on Computer Vision*, pages 670–675, London, UK, 1992. Springer-Verlag.

[31] M. Callieri, P. Cignoni, F. Ganovelli, G. Impoco, C. Montani, P. Pingi, F. Ponchio, and R. Scopigno. Visualization and 3d data processing in David restoration. *IEEE Computer Graphics & Applications*, 24(2):16–21, Mar.-Apr. 2004.

[32] M. Callieri, P. Cignoni, F. Ganovelli, C. Montani, P. Pingi, and R. Scopigno. Vclab's tools for 3d range data processing. In A. Chalmers D. Arnold and F. Niccolucci, editors, *VAST 2003*, pages 13–22, Bighton, UK, Nov. 5-7 2003. Eurographics.

[33] M. Callieri, P. Cignoni, and R. Scopigno. Reconstructing textured meshes from multiple range rgb maps. In *7th Int.l Fall Workshop on Vision, Modeling, and Visualization 2002*, pages 419–426, Erlangen (D), Nov. 20 - 22 2002. IOS Press.

[34] Marco Callieri, Paolo Cignoni, Massimiliano Corsini, and Roberto Scopigno. Masked photo blending: mapping dense photographic dataset on high-resolution 3d models. *Computer & Graphics*, 32(4):464–473, Aug 2008.

[35] Marco Callieri, Federico Ponchio, Paolo Cignoni, and Roberto Scopigno. Virtual inspector: a flexible visualizer for dense 3d scanned models. *IEEE Computer Graphics and Applications*, 28(1):44–55, Jan.-Febr. 2008.

[36] Jonathan C. Carr, Richard K. Beatson, Jon B. Cherrie, Tim J. Mitchell, W. Richard Fright, Bruce C. McCallum, and Tim R. Evans. Reconstruction and representation of 3d objects with radial basis functions. In *SIGGRAPH*, pages 67–76, 2001.

[37] E. Catmull. *A Subdivision Algorithm for Computer Display of Curved Surfaces*. PhD thesis, Computer Science Department, University of Utah, December 1974.

[38] Hui Chen and Bir Bhanu. Contour matching for 3d ear recognition. In *WACV-MOTION '05: Volume 1*, pages 123–128, Washington, DC, USA, 2005.

[39] Hui Chen and Bir Bhanu. Human ear recognition in 3d. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(4):718–737, April 2007.

[40] M. Chen, M. W. Jones, and P. Townsend. Methods for volume metamorphosis. In *Image Processing for Broadcast and Video Production*. Springer-Verlag, 1995.

[41] Antoni Chica, Marco Callieri, Isaac Besoora, Matteo Dellepiane, Jordi Moyes, Massimiliano Corsini, Guido Ranzuglia, Roberto Scopigno, and Pere Brunet. Multiscale acquisition and presentation of very large artifacts: the case of portalada. *ACM Journ. on Computers and Cultural heritag*, (Submitted), 2009.

[42] P. Cignoni, F. Ganovelli, E. Gobbetti, F. Marton, F. Ponchio, and R. Scopigno. Batched multi triangulation. In *Proceedings IEEE Visualization*, Conference held in Minneapolis, MI, USA, October 2005. IEEE Computer Society Press.

[43] P. Cignoni, C. Rocchini, and R. Scopigno. Metro: measuring error on simplified surfaces. *Computer Graphics Forum*, 17(2):167–174, June 1998.

[44] Paolo Cignoni, Marco Callieri, Massimiliano Corsini, Matteo Dellepiane, Fabio Ganovelli, and Guido Ranzuglia. Meshlab: an open-source mesh processing tool. In *Sixth Eurographics Italian Chapter Conference*, pages 129–136, 2008.

[45] Ioan Cleju and Dietmar Saupe. Stochastic optimization of multiple texture registration using mutual information. In *DAGM-Symposium*, pages 517–526, 2007.

[46] Commission Internationale de l'Eclairage (CIE). *Colorimetry. CIE 15:2004*, 2004.

[47] N. D'Apuzzo. Human face modeling frommulti images. In *Proc. of 3rd Int. Image Sensing Seminar on New Dev. in Digital Photogrammetry, Gifu, Japan*, pages 28–29, 2001.

[48] James Davis, Steven R. Marschner, Matt Garr, and Marc Levoy. Filling holes in complex surfaces using volumetric diffusion. In *In First International Symposium on 3D Data Processing, Visualization, and Transmission*, pages 428–438, 2002.

[49] Paul Debevec. Rendering synthetic objects into real scenes: bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *SIGGRAPH '98*, pages 189–198. ACM, 1998.

[50] Paul Debevec, Tim Hawkins, Chris Tchou, Haarm-Pieter Duiker, Westley Sarokin, and Mark Sagar. Acquiring the reflectance field of a human face.

In *SIGGRAPH '00: Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 145–156, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co.

[51] P.E. Debevec, C.J. Taylor, and J. Malik. Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach. In Holly Rushmeier, editor, *SIGGRAPH 96 Conference Proceedings*, Annual Conference Series, pages 11–20. ACM SIGGRAPH, Addison Wesley, August 1996.

[52] Matteo Dellepiane, Marco Callieri, Massimiliano Corsini, Paolo Cignoni, and Roberto Scopigno. Flash lighting space sampling. *Lecture Notes in Computer Science*, 5496:217–229, may 2009.

[53] Matteo Dellepiane, Marco Callieri, Mark Fondersmith, Paolo Cignoni, and Roberto Scopigno. Using 3d scanning to analyze a proposal for the attribution of a bronze horse to leonardo da vinci. In *The 8th International Symposium on VAST International Symposium on Virtual Reality, Archaeology and Cultural Heritage*, pages 117–124. Eurographics, Nov 2007.

[54] Paulo Dias, Vítor Sequeira, Francisco Vaz, and João G. M. Gonçalves. Registration and fusion of intensity and range data for 3d modelling of real world scenes. In *3DIM*, pages 418–426, 2003.

[55] F. Dornaika and C. Garcia. Robust camera calibration using 2d to 3d feature correspondences. In *Proceedings of the International Symposium SPIE –Optical Science Engineering and Instrumentation, Videometrics V, Volume 3174*, pages 123–133, 1997.

[56] Julie Dorsey, Holly Rushmeier, and François Sillion. *Digital Modeling of Material Appearance*. Morgan Kaufmann / Elsevier, 2007.

[57] Richard O. Duda, Carlos Avendano, and V. Ralph Algazi. An adaptable ellipsoidal head model for the interaural time difference. In *Proc. IEEE (ICASSP)*, pages II:965–968, 1999.

[58] Marc Ebner. Color constancy using local color shifts. In *ECCV*, pages 276–287, 2004.

[59] Elmar Eisemann and Frédo Durand. Flash photography enhancement via intrinsic relighting. In *ACM Trans. on Graphics*, volume 23. ACM Press, 2004.

[60] O.D. Faugeras and G. Toscani. The calibration problem for stereo. In *Proceedings CVPR '86, Miami Beach, Florida*, pages 15–20, 1986.

[61] G. D. Finlayson and S. D. Hordley. Improving gamut mapping color constancy. *Image Processing,IEEE Trans.*, 9(10):1774–1783, 2000.

[62] Graham D. Finlayson, Steven D. Hordley, and Paul M. Hubel. Color by correlation: A simple, unifying framework for color constancy. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(11):1209–1221, 2001.

[63] M.A. Fischler and R.C. Bolles. Random sample consensus: A paradigm for model fitting with application to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, June 1981.

[64] Max-Planck-Insitute for the History of Science and University of Bern. Digilib, a versatile image viewing environment for the internet. More info on: `http://digilib.berlios.de`.

[65] T. Franken, M. Dellepiane, F. Ganovelli, P. Cignoni, C. Montani, and R. Scopigno. Minimizing user intervention in registering 2D images to 3D models. *The Visual Computer*, 21(8-10):619–628, sep 2005.

[66] Thomas Franken, Paolo Cignoni, Fabio Ganovelli, Matteo Dellepiane, Roberto Scopigno, and Claudio Montani. Assisting the user in image to geometry alignment. In *3D Digital Imaging and Modeling: Applications of Heritage, Industry, Medicine and Land, Workshop Italy-Canada*, 17-18 May 2005 2005.

[67] Kouta Fujimura, Yasuhiro Oue, and Tomoya Terauchi. Improved 3d head reconstruction system based on combining shape-from-silhouette with two-stage stereo algorithm. In *ICPR '04: Volume 3*, pages 127–130, Washington, DC, USA, 2004.

[68] W.G. Gardner. Spatial audio reproduction: Towards individualized binaural sound. *National Academy of Engineering*, 2005.

[69] A. Gijsenij and Th. Gevers. Color constancy using natural image statistics. In *Int. Conf. on Comp. Vision and Pat. Recogn.*, pages 1–8, Minneapolis, USA, June 2007.

[70] B. Glocker, N. Komodakis, G. Tziritas, N. Navab, and N. Paragios. Dense image registration through mrfs and efficient linear programming. *Medical Image Analysis*, 12(6):731–741, 2008.

[71] Michael Goesele, Xavier Granier, Wolfgang Heidrich, and Hans-Peter Seidel. Accurate light source acquisition and rendering. *ACM Tr. Gr.*, 22(3):621–630, 2003.

[72] Steven J. Gortler, Radek Grzeszczuk, Richard Szeliski, and Michael F. Cohen. The lumigraph. In *SIGGRAPH '96*, pages 43–54, 1996.

[73] Frank Grubbs. Procedures for detecting outlying observations in samples. *Technometrics*, 11:1–21, Feb 1969.

[74] Gabriele Guidi, Bernard Frischer, Michele Russo, Alessandro Spinetti, Luca Carosso, and Laura Loredana Micoli. Three-dimensional acquisition of large and detailed cultural heritage objects. *Mach. Vision Appl.*, 17(6):349–360, 2006.

[75] R. Hassanpour and V. Atalay. Delaunay triangulation based 3d human face modeling from uncalibrated images. *Computer Vision and Pattern Rec. Workshop*, pages 75–75, 2004.

[76] Wolfgang Heidrich, Jan Kautz, Philipp Slusallek, and Hans-Peter Seidel. Canned lightsources. In *Rend. Tech.*, pages 293–300, 1998.

[77] H. Hoppe and K. Toyama. Continuous flash. Technical Report MSR-TR-2003-63, Microsoft Research, 2003.

[78] Berthold K.P. Horn and Brian G. Schunck. Determining optical flow. Technical report, Cambridge, MA, USA, 1980.

[79] E. Hsu, T. Mertens, S. Paris, S. Avidan, and F. Durand. Light mixture estimation for spatially varying white balance. In *SIGGRAPH '08*. ACM Press, 2008.

[80] A. Iannarelli. Ear identification. *Paramount Publishing Company, Freemont, California*, 1989.

[81] Katsushi Ikeuchi, Takeshi Oishi, Jun Takamatsu, Ryusuke Sagawa, Atsushi Nakazawa, Ryo Kurazume, Ko Nishino, Mawo Kamakura, and Yasuhide Okamoto. The great buddha project: Digitally archiving, restoring, and analyzing cultural heritage objects. *Int. J. Comput. Vision*, 75(1):189–208, 2007.

[82] Horace Ho-Shing Ip and Lijun Yin. Constructing a 3d individualized head model from two orthogonal views. *The Visual Computer*, 12(5):254–266, 1996.

[83] E. Jeges and L. Mate. Model-based human ear identification. *World Automation Congress, 2006. WAC '06*, pages 1–6, 24-26 July 2006.

[84] Tao Ju. Robust repair of polygonal models. *ACM Trans. Graph.*, 23(3):888–895, 2004.

[85] Y. Kahana and P.A. Nelson. Numerical modelling of the spatial acoustic response of the human pinna. *Journal of Sound and Vibration*, 292(1-2):148–178, april 2006.

[86] B. Katz. Boundary element method calculation of individual head-related transfer function. part I: Rigid model calculation. *Journal Acoustical Soc. Am.*, 110(5):2440–2448, 2001.

[87] B. Katz and D.R. Begault. Round robin comparison of HRTF measurement systems: preliminary results. In *Proc. 19th Intl. Congress on Acoustics (ICA2007), Madrid, Spain*, 2007.

[88] Rei Kawakami, Katsushi Ikeuchi, and Robby T. Tan. Consistent surface color for texturing large objects in outdoor scenes. In *ICCV '05: Int. Conf. on Computer Vision*, pages 1200–1207, Washington, DC, USA, 2005.

[89] Rakesh Kumar and Allen R. Hanson. Robust methods for estimating pose and a sensitivity analysis. *CVGIP: Image Underst.*, 60(3):313–342, 1994.

[90] Veronique Larcher. *Techniques de spatialisation des sons pour la ralit virtuelle*. These de doctorat, Universit Paris 6 (Pierre et Marie Curie), Paris, 2001.

[91] Hendrik P. A. Lensch, Wolfgang Heidrich, and Hans-Peter Seidel. Automated texture registration and stitching for real world models. In *PG '00: Proceedings of the 8th Pacific Conference on Computer Graphics and Applications*, page 317, Washington, DC, USA, 2000. IEEE Computer Society.

[92] Hendrik P. A. Lensch, Jan Kautz, Michael Goesele, Wolfgang Heidrich, and Hans-Peter Seidel. Image-based reconstruction of spatial appearance and geometric detail. *ACM Trans. Graph.*, 22(2):234–257, 2003.

[93] H.P. Lensch, W. Heidrich, and H.P. Seidel. Automated texture registration and stitching for real world models. In *Proc. 8th Pacific Graphics 2000 Conf. on Computer Graphics and Application*, pages 317–327, Los Alamitos, CA, 2000. IEEE.

[94] P. A. Lensch, Jan Kautz, Michael Goesele, Wolfgang Heidrich, and Hans-Peter Seidel. Image-based reconstruction of spatially varying materials. In *Rendering Techniques 2001 (Proc. 12th Eurographics WS on Rendering*, pages 104–115, London(UK), June, 25-27 2001. Springer, Wien.

[95] 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 *SIGGRAPH 2000, Computer Graphics Proceedings*, Annual Conference Series, pages 131–144. Addison Wesley, July 24-28 2000.

[96] Peter Liepa. Filling holes in meshes. *Eurographics Symposium on Geometry Processing*, 2003.

[97] Dani Lischinski, Zeev Farbman, Matthew Uyttendaele, and Richard Szeliski. Interactive local adjustment of tonal values. *ACM Tr. Gr.*, 25(3):646–653, 2006.

[98] Lingyun Liu and Ioannis Stamos. Automatic 3d to 2d registration for the photorealistic rendering of urban scenes. *CVPR*, 2:137–143, 2005.

[99] Lingyun Liu, Ioannis Stamos, Gene Yu, George Wolberg, and Siavash Zokai. Multiview geometry for texture mapping 2d images onto 3d range data. *Computer Vision and Pattern Recognition*, 02:2293–2300, 2006.

[100] M.I.A. Lourakis. levmar: Levenberg-marquardt non-linear least squares algorithms in C/C++. [web page] `http://www.ics.forth.gr/~lourakis/levmar/`, Jul. 2004.

[101] David G. Lowe. Fitting parameterized three-dimensional models to images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13:441–450, 1991.

[102] Cheng Lu, Mark S. Drew, and Graham D. Finlayson. Shadow removal via flash/noflash illumination. *W. on Mult. Signal Processing*, pages 198–201, 2006.

[103] F. Maes, A. Collignon, D. Vandeermeulen, G. Marchal, and P. Suetens. Multimodality image registration by maximization of mutual information. *IEEE Transactions in Medical Imaging*, 16:187–198, 1997.

[104] F. Maes, D. Vandermeulen, and P. Suetens. Comparative evaluation of multiresolution optimization strategies for multimodality image registration by maximization of mutual information. *Medical Image Analysis*, 3(4):373386, 1999.

[105] Stephen R. Marschner, Stephen H. Westin, Eric P. F. Lafortune, Kenneth E. Torrance, and Donald P. Greenberg. Image-based brdf measurement including

human skin. *Proceedings of 10th Eurographics Workshop on Rendering*, pages 139–152, June 1999.

[106] K. Matsushita and T. Kaneko. Efficient and handy texture mapping on 3D surfaces. In P. Brunet and R. Scopigno, editors, *Computer Graphics Forum (Eurographics '99)*, volume 18(3), pages 349–358. Blackwell Publishers, 1999.

[107] Kenji Matsushita and Toyohisa Kaneko. Efficient and handy texture mapping on 3d surfaces. *Computer Graphics Forum*, 18(3):349–358, 1999.

[108] W. Matusik, M. Zwicker, and F. Durand. Texture design using a simplicial complex of morphable textures. *SIGGRAPH (ACM Transactions on Graphics)*, 2005.

[109] J.C. Middlebrooks, E.A. Macpherson, and Z.A. Onsan. Psychophysical customization of directional transfer functions for virtual sound localization. *Journal Acoustical Soc. Am.*, 108(6):3088–3091, 2000.

[110] Peter J. Neugebauer and Konrad Klein. Texturing 3d models of real world objects from multiple unregistered photographic views. *Computer Graphics Forum*, 18(3):245–256, 1999.

[111] NextEngine. A desktop 3d scanner. More info on: `https://www.nextengine.com/`.

[112] Minh X. Nguyen, Xiaoru Yuan, and Baoquan Chen. Geometry completion and detail generation by texture synthesis. In *Proceeding of Pacific Graphics (PG05) Macao*, pages 8–10. Springer-Verlag, 2005.

[113] F.S. Nooruddin and G Turk. Simplification and repair of polygonal models using volumetric techniques. *IEEE Transactions on Visualization and Computer Graphics*, 9(2):191–205, 2003.

[114] Ido Omer and Michael Werman. Color lines: Image specific color representation. In *CVPR 04*, volume II, pages 946–953. IEEE, June 2004.

[115] Giorgio Panin and Alois Knoll. Mutual information-based 3d object tracking. *Int. J. Comput. Vision*, 78(1):107–118, 2008.

[116] Michal VarnuVarnuškaand Jindøich Parus and Ivana Kolingerová. Simple holes triangulation in surface reconstruction. In Angela Handlovivèová, editor, *Algoritmy 2005*. Slovak University of Technology, mar 2005.

[117] Georg Petschnigg, Richard Szeliski, Maneesh Agrawala, Michael Cohen, Hugues Hoppe, and Kentaro Toyama. Digital photography with flash and no-flash image pairs. *SIGGRAPH '04*, pages 664–72, 2004.

[118] Nico Pietroni, Miguel A. Otaduy, Bernd Bickel, Fabio Ganovelli, and Markus Gross. Texturing internal surfaces from a few cross sections. *Comp. Graph. Forum*, 26(3), 2007.

[119] Frederic Pighin, Jamie Hecker, Dani Lischinski, Richard Szeliski, and David H. Salesin. Synthesizing realistic facial expressions from photographs. In *SIGGRAPH '98: Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 75–84, New York, NY, USA, 1998. ACM Press.

[120] Paolo Pingi, Andrea Fasano, Paolo Cignoni, Claudio Montani, and Roberto Scopigno. Exploiting the scanning sequence for automatic registration of large sets of range maps. *Computer Graphics Forum*, 24(3):517–526, 2005.

[121] J. P. W. Pluim, J. B. A. Maintz, and M. A. Viergever. Mutual-information-based registration of medical images: a survey. *Medical Imaging, IEEE Transactions on*, 22(8):986–1004, 2003.

[122] William K. Pratt. *Digital image processing*. Wiley, New York :, 1978.

[123] K. Pulli, H. Abi-Rached, T. Duchamp, L. Shapiro, and W. Stuetzle. Acquisition and visualization of colored 3d objects. In *Proceedings of ICPR 98*, pages 11,15, 1998.

[124] Ravi Ramamoorthi, Sebastian Enrique, and Peter N. Belhumeur. Reflectance sharing: Predicting appearance from a sparse set of images of a known shape. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(8):1287–1302, 2006. Member-Todd Zickler.

[125] Ravi Ramamoorthi and Pat Hanrahan. A signal-processing framework for inverse rendering. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 117–128, New York, NY, USA, 2001. ACM.

[126] V. Rankov, R. Locke, R. Edens, P. Barber, and B. Vojnovic. An algorithm for image stitching and blending. In *Proceedings of SPIE. Three-Dimensional and Multidimensional Microscopy: Image Acquisition and Processing XII*, volume 5701, pages 190–199, March 2005.

[127] S. Russell and P. Norvig. *Artificial Intelligence a Modern Approach.* AI. Prentice_Hall, 1995.

[128] Yoichi Sato, Mark D. Wheeler, and Katsushi Ikeuchi. Object shape and reflectance modeling from observation. In *SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 379–387, New York, NY, USA, 1997. ACM Press/Addison-Wesley Publishing Co.

[129] A. Sheffer, E. Praun, and K. Rose. Mesh parameterization methods and their applications. *Foundations and Trends in Computer Graphics and Vision*, 2(2):105–171, 2006.

[130] Won sook Lee and Nadia Magnenat Thalmann. Head modeling from pictures and morphing in 3d with image metamorphosis based on triangulation. In *Lecture Notes in Artificial Intelligence, 1537*, pages 254–267. Springer, 1998.

[131] J. Stumpfel, C. Tchou, T. Hawkins, P. Martinez, B. Emerson, M. Brownlow, A. Jones, N. Yun, and P. Debevec. Digital reunification of the Parthenon and its sculptures. In *4th International Symposium on Virtual Reality, Archaeology and Intelligent Cultural Heritage*. VAST, 2003.

[132] M. Subbarao, T. Chio, and A. Nikzad. Focusing techniques. *Optical Engineering*, 32(11):2824–2836, 1993.

[133] R. Tsai. A versatile camera calibration technique for high accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE Journal of Robotics and Automation*, RA-3(4), August 1987.

[134] Nicolas Tsingos, Carsten Dachsbacher, Sylvain Lefebvre, and Matteo Dellepiane. Instant sound scattering. In *Proc. of the Eurographics Symposium on Rendering*, 2007.

[135] J. Unger, A. Wenger, T. Hawkins, A. Gardner, and P. Debevec. Capturing and rendering with incident light fields. In *EGRW '03*, pages 141–149, 2003.

[136] V. Valzano, A. Bandiera, Beraldin J.-A., M. Picard, S. El-Hakim, G. Godin, E. Paquet, and M. Rioux. Fusion of 3d information for efficient modeling of cultural heritage sites with objects. In *Proc. CIPA 2005 XXth Int. Symposium, Torino, Italy*, 2005.

[137] Joost van de Weijer and Theo Gevers. Color constancy based on the grey-edge hypothesis. In *ICIP*, pages 722–725, 2005.

[138] A. Venturi. Metodo di hole filling su mesh poligonali mediante proiezione di pattern laser. Thesis, Department of Computer Science, Pisa University, Apr. 2009.

[139] Maarten Vergauwen and Luc Van Gool. Web-based 3d reconstruction service. *Mach. Vision Appl.*, 17(6):411–426, 2006.

[140] Paul Viola and III William M. Wells. Alignment by maximization of mutual information. *Int. J. Computer Vision*, 24(2):137–154, 1997.

[141] G.Q. Wei and S.D. Ma. Implicit and explicit camera calibration: Theory and experiments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(5):469–480, 1994.

[142] Songhua Xu, Athinodoros Georghiades, Holly Rushmeier, Julie Dorsey, and Leonard McMillan. Image guided geometry inference. In *3DPVT '06: Proceedings of the Third International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT'06)*, pages 310–317, Washington, DC, USA, 2006. IEEE Computer Society.

[143] Y. Yu, P. Debevec, J. Malik, and T. Hawkins. Inverse global illumination: recovering reflectance models of real scenes from photographs. In A. Rockwood, editor, *SIGGRAPH 99 Conf. Proc.*, Annual Conf. Series, pages 215–224. ACM SIGGRAPH, Addison Wesley, July 1999.

[144] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:1330–1334, 2000.