

Chapter 1

Processing sampled 3D data: reconstruction and visualization technologies

MARCO CALLIERI, MATTEO DELLEPIANE, PAOLO CIGNONI, ROBERTO SCOPIGNO

Istituto di Scienza e Tecnologie dell'Informazione
Consiglio Nazionale delle Ricerche
Pisa, Italy

Email: m.callieri@isti.cnr.it, m.dellepiane@isti.cnr.it,
p.cignoni@isti.cnr.it, r.scopigno@isti.cnr.it

1.1 Introduction

This chapter describes the so-called 3D scanning pipeline (i.e., how raw sampled 3D data have to be processed to obtain a complete 3D model of a real-world object). This kind of raw data may be the result of the sampling of a real-world object by a 3D scanning device [1] or by one of the recent image-based approaches (which returns raw 3D data by processing set of images) [2]. Thanks to the improvement of the 3D scanning devices (and the development of software tools), it is now quite easy to obtain high quality, high resolution three-dimensional sampling in relatively short times. Conversely, processing this fragmented/raw data to generate a complete and usable 3D model is still a complex task, requiring the use of several algorithms and tools; the knowledge of the processing tasks and solutions required is still the realm of well-informed practitioners and it often appears as a set of obscure black boxes to most of the users. Therefore, we focus the chapter on: (a) the geometric processing tasks that have to be applied to raw 3D scanned data to transform them into a clean and complete 3D model and, (b), how 3D scanning technology should be used for the acquisition of real artifacts. The sources of sample 3D data (i.e., the different hardware systems used in 3D scanning), are briefly presented in the following subsection. Please note that the domains of 3D scanning, geometric processing, visualization and applications to CH are too wide to provide a complete and exhaustive bibliography in a single chapter; we decided to describe and cite here just a few, representative references to the literature. Our goal is to describe the software processing, the pitfalls of current solutions (trying to cope both with existing commercial systems and academic tools/results) and to highlight some topics of interest for future research, according to our experience and sensibility. The presentation follows in part the structure of a recently published paper on the same subject [3].

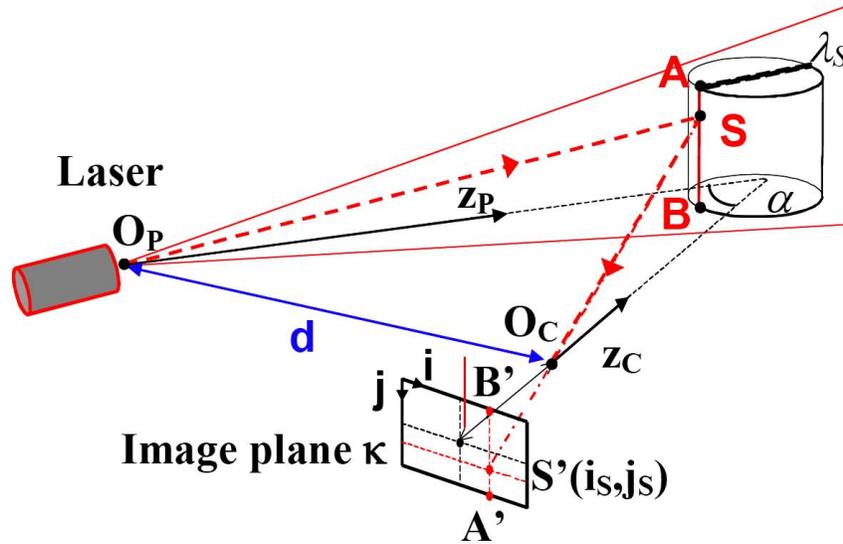


Figure 1.1: Schema of the *geometric triangulation* principle adopted by *active optical* scanners (Image courtesy of Giovanna Sansoni et al. [5]).

1.1.1 Sources of sampled 3D data

Automatic 3D reconstruction technologies have evolved significantly in the last decade; an overview of 3D scanning technologies is presented in [4, 1, 5]. The technological progress in its early stages has been driven mostly by industrial applications (quality control, industrial metrology). Cultural Heritage (CH) is a more recent applications field, but its specific requirements (accuracy, portability and also the pressing need to sample and integrate the color information) were often an important testbed for the assessment of new, general purpose technologies.

Among various 3D scanning systems, the more frequently used for 3D digitization are the so called *active optical* devices. These systems shoot some sort of controlled, structured illumination over the surface of the artifact and reconstruct its geometry by checking how the light is reflected by the surface. Examples of this approach are the many systems based on the *geometric triangulation* principle (see Figure 1.1). These system project light patterns on the surface, measure the position of the reflected light by a CCD device located in a known calibrated position with respect to the light emitter. Light is either *coherent*, usually a laser stripe swept over the object surface, or *incoherent*, such as the more complex fringe patterns produced with video- or slide-projectors. Due to the required fixed distance between the light emitter and the CCD sensor (to be accurate, the apex angle of the triangle that interconnects the light emitter, the CCD sensor and the sampled surface point should be not too small, i.e. 10-30 degrees), those systems usually support working volumes that range from a few centimeters to around one meter.

The acquisition in a single shot of a much larger extent is supported by devices that employ the so-called *time-of-flight* (TOF) approach (see Figure 1.2). TOF systems send a pulsed light signal towards the surface and measure the time elapsed until the reflection of the same signal is sensed by an imaging device (e.g., a photo-diode). We can compute, given the light speed, the distance to the sampled surface; since we know the direction of the emitted light and the distance over this line, the XYZ locations of the sampled point can be easily derived. An advantage of this technology is the very wide working volume (we can scan an entire building facade or a city square with a single shot); when compared with the capabilities of triangulation-based systems, disadvantages of TOF devices are the lower accuracy and lower sampling density (i.e. the inter-sampling distance over the measured surface is usually in the range of the centimeter). The capabilities of TOF devices have been improved recently with the introduction of variations of the basic technology, based on modulation of the sensing probe, that allowed to increase significantly the sampling speed and

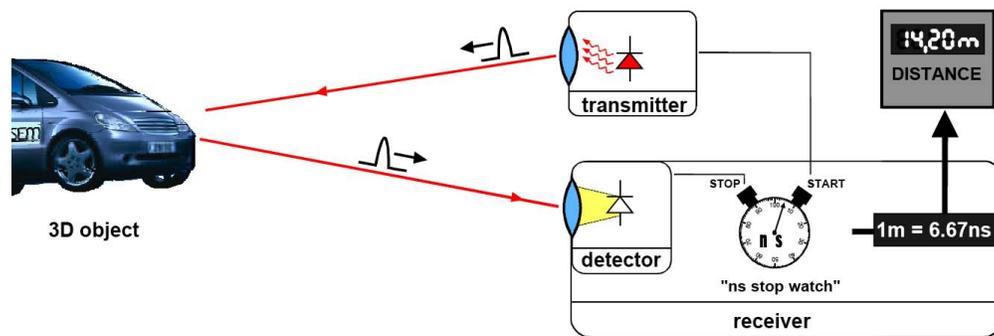


Figure 1.2: Basic principle of a *time-of-flight* scanning system (Image courtesy of R. Lange et al. [6]).

maintain very good accuracies (in the order of a few millimeters).

Very promising but still not very common are the *passive optical* devices, where usually a large number of images of the artifact are taken and a complete model is reconstructed from these images [2, 7, 8]. These approaches, mostly based on consumer digital photography and sophisticated software processing, are presented in Chapter 18. **(Note for Editors: the book chapter by van Gool has to be referred here. Please check the numeric references.)**

The quality of the contemporary commercial scanning systems is quite good if we take into account accuracy and speed of the devices; unfortunately, cost is still high, especially for CH applications, which are usually characterized by very low budgets. The introduction of inexpensive low-end laser-based systems (e.g., the Next Engine device [9]), together with the impressive improvement of image-based 3D reconstruction technologies (e.g., the Arc3D web-based reconstruction tool [10, 11]) are very beneficial for the field. The availability of low cost options is strategic to increase the diffusion of 3D scanning technology and to rise awareness and competence in the application domain. Moreover, a wider user community could ultimately drive to significant price reduction of the high-end scanning systems as well.

Finally, most of the existing systems consider just the acquisition of the external shape (geometric information), while a very important aspect in many applications is *color sampling*. This limitation is caused mainly by the fact that most 3D sensors come from the industrial world and therefore are designed to focus more on geometry acquisition than color. Color is the weakest feature of contemporary technology, since the color-enabled scanners produce usually just a low quality sampling of the surface color (with a notable exception of the technology based on multiple laser wavelengths [12], unfortunately characterized by a very high price that made their market share nearly negligible). Moreover, existing devices sample only the *apparent* color of the surface and not its *reflectance properties*, that constitute the characterizing aspect of the surface appearance. There is wide potential for improving current technology to cope with more sophisticated surface reflection sampling.

1.2 Basic Geometric Processing of Scanned Data

Unfortunately, almost every 3D scanning system does not produce a final, complete 3D model but a large collection of raw data which have to be post-processed. This is especially the case of all *active optical* devices, since only the portion of the surface directly visible from the device is captured in a single shot. The main problem of a 3D acquisition is indeed this fragmentation of the starting data: scanners do not produce a complete model by simply *pushing a button* (unless the application field is very restricted, such as some devices designed specifically for scanning small objects e.g., teeth). On the other hand, the good news is that most of the scanning technologies produce raw data which are very similar: the so-called *range maps* (see Figure 1.3). This homogeneity in the input data makes the 3D scan processing quite independent

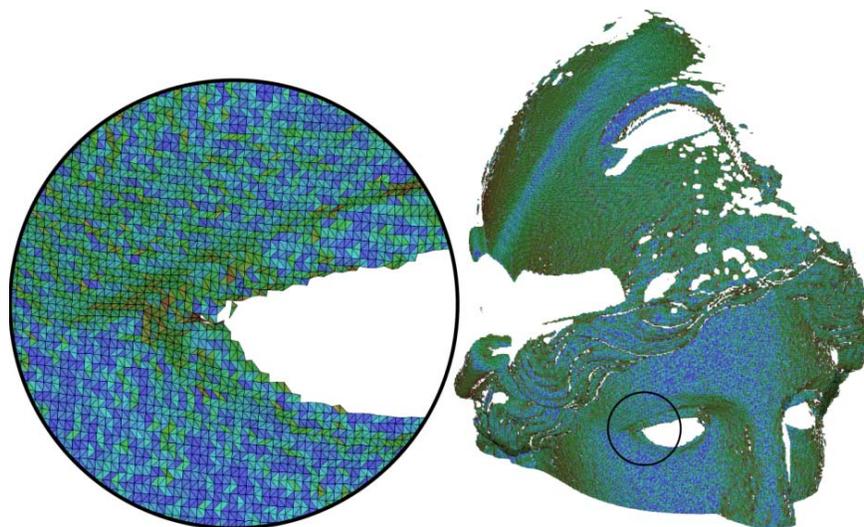


Figure 1.3: An example of a range map (from the 3D scanning of the Minerva of Arezzo, National Archeological Museum in Florence, Italy).

from the specific sampling device adopted. Of course, depending on: the technology and specifications of the employed sensor; the scale, material and nature of the target object; and the kind of 3D model needed, the various processing phases might slightly vary, but the main phases and the workflow will generally still fall in the scheme we present here.

The result of a single scan, a range map, is the counterpart of a digital image: while a digital image encodes in each pixel the specific local color of the sampled surface, the range map contains for each sample the geometrical data (a point in XYZ space) which characterize the location in the 3D space of the corresponding small parcel of sampled surface. The range map therefore encodes the geometry of just the surface portion which can be seen and sampled from a selected view point (according to the specific features of the scanning device).

The complete scan of an artifact requires usually the acquisition of many shots taken from different viewpoints, to gather complete information on its shape. Each shot produces a range map. The number of range maps required to sample the entire surface depends on: the working volume of the specific scanner; the surface extent of the object; and its shape complexity. Usually, we sample from a few tens up to a few hundred range maps for each artifact. Range maps have to be processed to convert the data encoded into a *single, complete, non-redundant* and *optimal* 3D representation (usually, a triangulated surface or, in some cases, an optimized point cloud). Examples of digital models produced with 3D scanning technology are presented in Figure 1.4 .

1.2.1 The 3D scanning pipeline

As previously stated, the homogeneous nature of the raw data makes the processing quite independent from the scanning device used and the processing software adopted. The processing of the raw data coming from the 3D scanning device is divided in subsequential steps, each one working on the data produced in the previous one; hence the name *3D scanning pipeline*. These steps are normally well recognizable in the various software tools, even if they are implemented with very different algorithms. Before or after each step, there may be a *cleaning stage*, aimed at the elimination of small defects of the intermediate data that would make the processing more difficult or longer. The overall structure of the 3D scanning pipeline is presented in an excellent overview paper [13], which describes the various algorithms involved in each step. Some new algorithms have been proposed since the date of publication of that review paper, but the overall organization of the pipeline is not changed.



Figure 1.4: Two examples of results obtained with: a time-of-flight (TOF) scanner, depicting the Dome square at S. Gimignano (Italy); laser-line triangulation scanner, depicting a capitol from the cloister of the Dome at Cefalú (Italy) .

The processing phases that compose the 3D scanning pipeline are:

- *Alignment* of the range maps. By definition the range map geometry is relative to the current sensor location (position in space, direction of view); each range map is then in a different reference frame. The aim of this phase is to place the range maps into a common coordinate space where all the range maps lie in the correct position with respect to each another. Most alignment methods do require user input and rely on the *overlapping regions* between adjacent range maps.
- *Merge* of the aligned range maps to produce a single digital 3D surface (this phase is also called *reconstruction*). After the alignment, all the range maps are in the correct position, but the object is still composed by multiple overlapping surfaces with lot of redundant data. A single, non redundant representation (usually, a triangulated mesh) has to be reconstructed out of the many, partially overlapping range maps. This processing phase, generally completely automatic, exploits the redundancy of the range maps data in order to produce (often) a more correct surface where the sampling noise is less evident than in the input range maps.
- *Mesh editing*. The goal of this step is to improve (weather possible) the quality of the reconstructed mesh. For example, usual actions are to remove or reduce impact of noisy data or to fix the unsampled regions (hole filling).
- *Mesh simplification* and *multiresolution encoding*. 3D scanning devices produce huge amount of data; it is quite easy to produce 3D models so complex that it is impossible to fit them in RAM or to display them in realtime. This complexity has usually to be reduced in a controlled manner, by producing discrete Level Of Details (LOD) or Multiresolution representations.
- *Color mapping*. The information content is enriched by adding color information (an important component of the visual appearance) to the geometry representation.

All these phases are supported either by commercial [14, 15, 16] or academic tools [17, 18, 19]. Unfortunately, given the somehow limited diffusion of 3D scanning devices (mainly due to their cost) and the still restricted user base of this kind of technology in non-industrial environments, it is not so easy to find free software tools to process raw 3D data. The availability of free tools would be extremely beneficial to the field, given the considerable cost of the major commercial tools, to increase awareness and dissemination

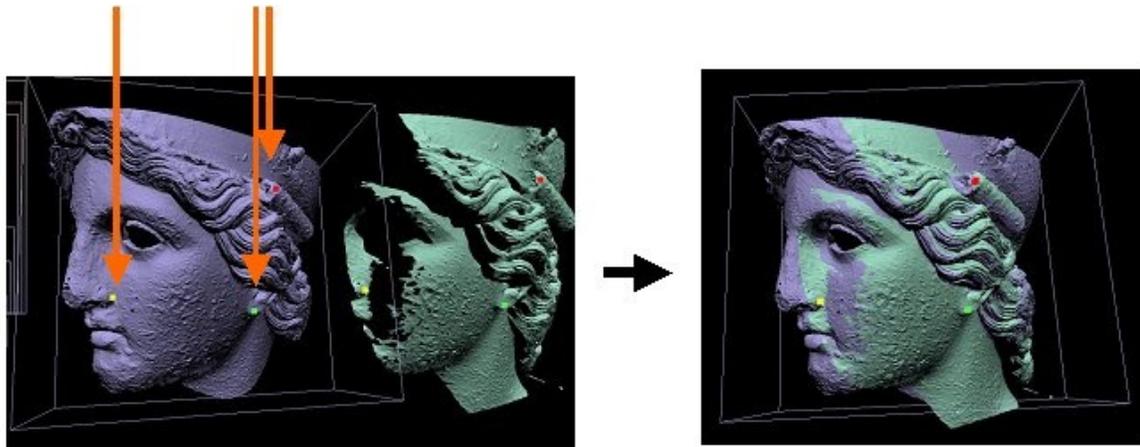


Figure 1.5: An example of a pair-wise alignment step: the selection of a few corresponding point pairs on the two range maps allows to find an initial alignment, which is then refined by the ICP algorithm.

to a wider users community. These instruments could be very valuable for the CH institution that want to experiment the use of this technology on a low budget. Most of the non-commercial development of processing tools has been carried out in the academic domain; some of those efforts are sponsored by European projects [20, 21]. The very recent availability of low cost scanning devices or of image-based 3D acquisition approaches, paired with availability of free processing tools, could boost the adoption of 3D scanning technologies in many different application domains.

The different phases of the scanning pipeline are presented in a more detailed manner in the following subsections.

Alignment

The *alignment* task converts all the acquired range maps in a common reference system. This process is usually partially manual and partially automatic. The user has to find an initial, raw registration between any pair of overlapping range maps; the alignment tool then uses this approximate placement to compute a very accurate registration of the two meshes (*local registration*). The precise pair-wise alignment is usually performed by adopting the Iterated Closest Point (IPC) algorithm [22, 23], or variations of the same. This pairwise registration process (repeated on all pairs of adjacent and overlapping range maps) is then used to automatically build a *global registration* of all the meshes, and this last alignment is enforced among all the maps in order to move everything in a unique reference system [24]. Range map alignment has been the focus of a stream of recent papers; we cite here only one single result [25], the interested reader can find references to other recent approaches in the previous work section of that paper.

Quality of the alignment is crucial for the overall quality of the merged model, since errors introduced in alignment may be the cause of wrong interpolation while we merge the range maps. Since small errors could add one to the other, the resulting global error can be macroscopic. An example of 3D acquisition where this potential effect has been monitored and accurately measured is the acquisition of Donatello's Maddalena [26]. A methodology for quality control is proposed in that paper, where potential incorrect alignment produced by ICP registration are monitored and corrected by means of data coming from close-range digital photogrammetry.

The alignment phase is usually the most time-consuming phase of the entire 3D scanning pipeline, due to the substantial user contribution required by current commercial systems and the large number of scans sampled in real scanning campaigns. The initial placement is heavily user-assisted in most of the commercial and academic systems (requiring the interactive selection and manipulation of the range maps). Moreover, this kernel action has to be repeated for all the possible *overlapping range map pairs* (i.e. in average 6-

8 times the number of range maps). Please consider that the scanning of a 2 meters tall statue generally requires from 100 up to 500 range maps, depending on shape complexity and sampling rate required. If the set of range maps is thus composed by hundreds of elements, then the user has a very complex task to perform: for each range map, find which are the partially-overlapping ones; given this set of overlapping range maps, determine which one to consider in pair-wise alignment (either all of them or a subset); finally, process all those pair-wise initial alignments. If not assisted, this becomes the major bottleneck of the entire process: a slow, boring activity that is also really crucial for the accuracy of the overall results (since an inaccurate alignment may lead to very poor results after the range maps merging).

An improved management of large set of range maps (from 100 up to 1000) can be obtained by both providing a hierarchical organization of the data (range maps divided into groups, with atomic alignment operations applied to an entire group rather than to the single scan) and by using a multiresolution representation of the raw data to increase efficiency of the interactive rendering/manipulation process over the range maps. Moreover, since the standard approach (user-assisted selection of each overlapping pair and creation of the correspondent alignment arc) becomes very impractical on large set of range maps, tools for the automatic setup of most of the required alignment actions have to be provided (see Subsection 1.2.2).

Merging

Reconstruction/merging from point samples or range maps has been one of the more active field on scanning-related research in the last few years. A very important feature of a reconstruction code is to perform a *weighted integration* of the range maps (or pointset) and not just joining them. Since we usually have a high degree of overlap and that sampled data are noisy, a weighted integration can significantly improve the accuracy of the final result by reducing the impact of most of the noisy samples. Another important feature of a reconstruction code is the capability to fill up small holes (i.e. region not sampled by the scanner, see Subsection 1.2.3). Finally, since reconstruction algorithms requires a very large memory footprint on big dataset, they have to be designed to work locally on sub-sections of the data, loading only the data subset involved in the generation of a single portion of the final results (*out-of-core* reconstruction).

Fit results to application requirements

Simplification/multiresolution is another key ingredient in scanned data processing. When we chose a reconstruction kernel region approximately equal to the inter-sampling distance used in scanning, the reconstructed models may become huge in size (i.e. many millions triangles). Most applications require significant complexity reduction in order to manage these models interactively. Two problems arise when we try to simplify such models: we need solutions working on external memory to cope with these big models [27]; simplification has to be accurate [28, 29] if we want to produce high-quality models and accurate interactive visualization (possibly, based on multiresolution data representation schemes). These issues are presented in Section 1.5.

Color Mapping

Color is usually the final stage of the pipeline. Here we have to support the reconstruction of textured meshes from a sampling of the object's surface reflection properties. The easiest and most common approach is the acquisition and mapping of the *apparent color* (reflected color, illumination-dependent) using a digital photo camera. But we can also immerse the artifact into a controlled lighting setup up, to produce a more sophisticated acquisition of the reflection properties of the object's surface (see for example the methodology for the acquisition of bi-directional reflectance distribution function - BRDF proposed in [30]). This latter approach is unfortunately often impractical for those applications, such as acquisition of CH artifacts, which

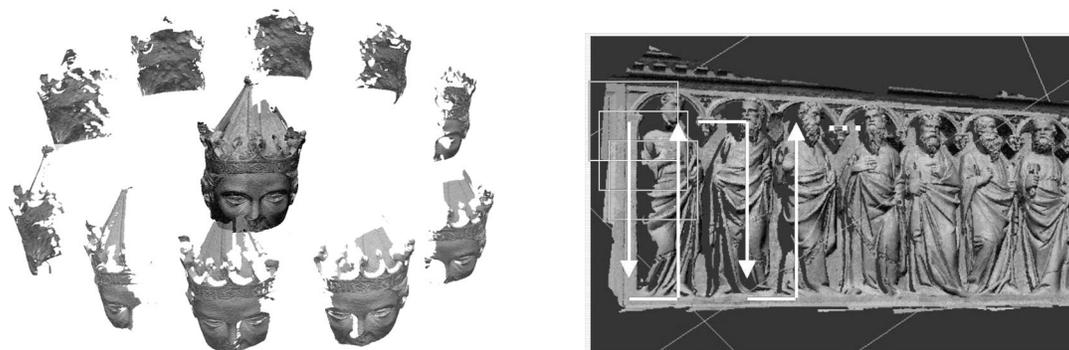


Figure 1.6: Range maps are often taken according to a regular order: an example of *circular* radial acquisition performed around a statue's head (left); an example of a *raster-scan* scanning order adopted for the acquisition of a bas-relief (right).

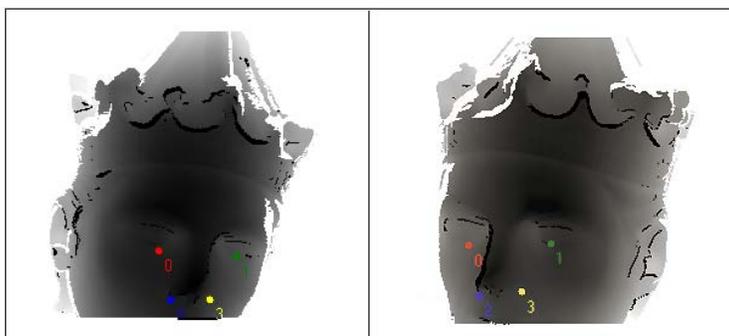


Figure 1.7: An example of four matching point pairs selected on two range maps by an automatic matching algorithm.

require to perform the acquisition in real-life conditions (i.e. when we cannot move the object to be scanned in an acquisition lab). The issues involved in color acquisition and management are discussed in detail in Section 1.3.

1.2.2 Implementing range maps alignment as an automatic process

We have already introduced that range maps alignment is usually the most time consuming phase in the 3D scanning pipeline. But the goal is obviously to make scanning as much as possible an automatic process. Therefore, completely automatic scanning systems have been proposed. Some of those systems are based on the use of complex and costly positioning/tracking machinery (e.g., see [31] or the several commercial scanners based on optical or magnetic tracking of the scanning head location); others adopt passive silhouette-based approaches which do not extend well to the acquisition of medium or large scale artifact. An alternative approach is to design new algorithms for processing the data produced by standard scanning systems, able to transform the classical scanning pipeline into a mostly unattended process. In particular, the range maps alignment phase is the only task where a considerable human intervention is required. Several papers proposed methods for *automatic alignment*, usually based on some form of shape analysis and characterization [32].

The general alignment problem can be made more easy to manage by considering some assumptions which usually hold in practical use. While designing a new automatic registration solution [33], we started from a few initial conditions directly gathered by our experience in 3D scanning. First, the *detection of*

the pairs of overlapping range maps can be reduced to a simpler task, once we notice that 3D acquisition is usually done by following simple scanning pose paths. Users usually acquire range maps in *sequences*, following either a *vertical*, *horizontal*, *raster-scan* or *circular* translation of the scanning system (see Figure 1.6). The different types of sequences share a common property: they contain an ordered set of n range maps, such that range map R_i holds a significant overlapping with at least R_{i-1} and R_{i+1} . Vertical, horizontal or raster-scan stripes are often produced when acquiring objects like bas-reliefs, walls or nearly-planar items. Circular stripes are indeed more useful when acquiring objects like statues, columns or objects with an axial symmetry.

If we can assume that the acquisition has been performed using one of these stripe-based patterns, then we may reduce the search for overlapping and coarse registration to each pair of consecutive range maps (R_i, R_{i+1}). An automatic registration module can process each couple (R_i, R_{i+1}), to produce in output the roto-translation matrix M_i that aligns R_{i+1} to R_i . Matrix M_i can be computed by applying some basic geometric processing to the two range maps: *feature points* can be detected by evaluating a shape descriptor on the two meshes (or point set); potential *corresponding feature points pairs* can be detected by adopting RANSAC-like approaches: from these possible point pairs, we can select the one which produces, after ICP alignment, the matrix M_i which performs the best alignment (see [33] as an example of this type of solutions). Many other approaches for the design of shape feature characterization and matching are also possible [34, 35]. As an alternative to geometry-based solutions, it is also possible to work with an image-based approach [36]: correspondent point pairs can be also retrieved by looking to image features in the RGB channel associated to the XYZ samples (under the assumption that the scanner produces self-registered shape and color samples).

The subset of registration arcs is not complete nor sufficient by itself when we restrict the search to consecutive pairs in linear sequences, since we usually have many other potential overlaps between range maps. On the other hand, information on those consecutive pairs is sufficient for the application of an intelligent ICP-based solution that allows to complete the graph. Designing a smart alignment tool able to complete the needed arcs (interconnecting R_i with all the overlapping range maps, not just R_{i-1} and R_{i+1}) is easy. We can represent the space of the set of range maps with a *spatial indexing* data structure, a regular grid which encodes for each 3D cell the set of range maps passing through that sub-volume. This data structure allows an easy automatic detection of the potential overlaps between generic pairs of range maps. We can then iterate automatic ICP-based alignment on all those overlapping range map pairs which have not been already processed according to the linear ordering. The alignment tool can therefore introduce all needed arcs (in a completely unattended manner), by selecting and processing only those arcs which satisfy a minimum-overlap factor.

The automatic registration approach sketched above [33] has been tested on many complex scanning campaigns (where each range map is usually affected by noise, artifacts and holes). An example concerning a bas-relief is shown in Figure 1.8, whose approximate length is 2.5 meters. In this case two raster-scan (snake-like) stripes were acquired and processed, for a total of 117 meshes (about 45.5M vertices). The overall automatic alignment requires usually much less than the raw scanning time and it is therefore sufficiently fast to run in background during the acquisition, processing all the scans in sequence as soon as they are produced.

Solutions similar to the ones presented in this subsection are unfortunately still not provided by commercial software; inclusion of automatic alignment would speedup significantly the processing of scanned data, reducing the manpower required and the overall costs.

1.2.3 Enhancing incomplete surface sampling

According to 3D scanning experience, obtaining a *complete sampling* of a complex artifact surface is often impossible [31]. Various are the reasons why we usually end up with an incomplete scan: presence of self-obstructing surfaces; presence of small cavities or folds; sections of the surface which are not cooperative with respect to the optical scanning technology adopted (highly reflective materials like polished metals,



Figure 1.8: The coarse alignment obtained over a set of range maps representing a bas-relief (*top*) and the final model (*bottom*) obtained after automatic completion of all overlapping-pairs; alignment has been performed in non-attended mode using the solution presented in [33].

transparent components such as glass or gems, surfaces which do not reflect the emitted light probe, etc). In all those cases, we have to decide if the model has to be completed or if we have to keep it incomplete. In the CH domain we are usually asked to produce models which should contain only sampled geometry, i.e. the use of software solutions which fill up the gaps is not allowed. On the other hand, incomplete digital models perform very poor in visualization, since the holes usually attract the observer's attention much more than the other clean parts. *Clean* sampled surfaces are therefore needed for visualization, obtained by closing all the gaps with plausible surface patches. A very nice extension to available data formats would be an attribute that could allow us to differentiate between sampled and interpolated geometry (a sort of *confidence value* assigned to each geometric element), making it possible to make visually evident those two different data components. This would be an interesting addition to provenance data encoding.

Gaps filling can be obtained by two orthogonal approaches: *volumetric* and *surface oriented*. In the first case the holes can be filled:

- At reconstruction time, for example by enhancing volumetric reconstruction approaches based on a discrete distance field with a diffusion process which extends the distance field in regions not covered by scanned samples [37] or by adopting methods based on the Poisson reconstruction formulation [38];
- After the reconstruction, by putting the model in a volumetric grid and devising solutions able to produce a water-tight surface [39, 40].

Unfortunately, these approaches make it very hard to differentiate sampled and interpolated geometry. Moreover, methods based on volumetric diffusion are very complicated to use because steering the diffusion process to obtain a plausible completion surface is not easy; a time-consuming trial and error process is usually needed to find the parameters best fitting a given dataset. Poisson-based reconstruction is more frequently used, due to ease of use and free availability (both as source code distributed by authors and as a feature available in open source tools, e.g., MeshLab [41]).

On the other hand, *geometric processing* solutions can be devised to detect and fill unsampled regions. Accordingly, *surface-oriented* approaches try to detect and close holes preserving triangle shape and curvature [42, 43]. The problem is not simple because we need geometrically robust solutions, able to close any gap with a surface patch which should share curvature continuity with respect to the surface regions adjacent to the open border. Some issues are the necessity to deal with self-intersections and isles, the algorithm speed and robustness, and the difficulty in creating fully automatic but reliable methods. Moreover, in

some cases a basic concept of curvature continuity is not enough, since a missing region can contain surface features (such as a given texture or a carved detail) that we may want to reproduce in a way conforming with adjacent regions. So called *surface inpainting* methods have been proposed to support intelligent cut & paste of surface detail from one completely sampled region to a partially sampled region [44, 45].

1.3 Color Sampling and Processing

There are many application domains which require not just shape sampling, but also accurate acquisition and management of color data. Sophisticated approaches for sampling the surface reflection characteristics have been proposed (e.g., generic BRFD sampling [30] or technologies for the acquisition of the reflectance of human faces [46, 47]). In some cases, those approaches are unfortunately too complicated to be massively applied in all those fields where we do not have the pleasure to work in controlled lab conditions, as it is the case of CH digitizations (usually performed in crowded museums and under a not proper illumination setup). Let us describe first the easier approach to gather color data and then how those data can be encoded in digital models. At the end of the section, we will review the other more sophisticated approaches to gather samples of the surface reflection properties.

1.3.1 Basic acquisition of color data

The basic approach, acquiring just the so-called *apparent color* and mapping those samples to the 3D model, is still widely used in most of the practical cases. A series of pictures can be taken with a digital camera, trying to avoid shadows and highlights by taking them under a favorable lighting setup; these photographs are then stitched onto the surface of the object. However, even in this simpler case, the processing needed to build a plausible texture is not straightforward [48]. Naive mapping of apparent color on the mesh can produce severe discontinuities that are due to the varying illumination conditions of the surface sampled by the photos (perceived color depends on specific illumination and direction of the viewer in the instant the photo is shot). Many different approaches have been proposed to reduce the aliasing and to produce seamless color mapping; we cite here only some representative papers: using the range intensities produced by some active optical scanning devices to correct the color information [49]; detecting and removing cast shadows [50], which are usually a major problem in color acquisition in outdoor scenes; devising methods for computing the inverse illumination (i.e., recovering approximate surface reflectance from a sampling of the real surface under known illumination conditions [51, 52]).

1.3.2 Recovering camera parameters

A basic problem in managing color information is how to register the images with the geometric data in a time-efficient way. Once intrinsic (focal length and distortion of the camera lenses) and extrinsic parameters (view specifications) have been computed for each image by registering it onto the geometric model, many approaches exist to map the color info on the 3D model, based on mesh parameterization or color-per-vertex encoding. The bottleneck in the color pipeline is the *image-to-geometry registration* phase, a complicated time-consuming phase which requires substantial intervention of a human operator since the current approach is based on the selection of several corresponding point pairs which link each 2D image to the 3D mesh [53].

We designed a tool to support image registration, *TexAlign* [54], which solves the image-to-geometry registration problem by constructing a *graph of correspondences*, where: the 3D model and all the images are represented as nodes; a link is created for any correspondence defined between two nodes (implementing either an *image-to-geometry* or an *image-to-image* correspondence). This graph of correspondences is used: (a) to keep track of the work done by the user; (b) to infer automatically new correspondences from the instantiated ones; and (c) to find the shortest path, in terms of the number of correspondences that must be provided by the user, to complete the registration of all the images. The goal is to assist the user in



Figure 1.9: The David model is shown with color mapping; on the left is the pre-restoration status (61 images mapped), while the post-restoration status is shown on the right (another set of 68 images). The two colored David models are rendered in real time with the *Virtual Inspector* system [56].

the management of large set of images. This system has been used to map several complex photographic sampling (e.g., in the framework of the Michelangelo's David restoration we mapped on the digital 3D model 61 images showing the pre-restoration status and 68 images depicting the post-restoration condition [55], see Figure 1.9).

Analogously to the range map alignment problem, this image registration phase should be as much as possible solved automatically. We need fully automatic and robust approaches able to solve the general problem (i.e., a large and complex object, where each image covers only a subset of its overall extent). Finding automatically the correspondences between a set of 2D images and a 3D mesh is not an easy task, also because the geometry features are usually less dense than the image features we can retrieve in the photographs. A possible research direction could be to move from the usual search for *image-to-geometry* correspondences to a context where we use both *image-to-geometry* and *image-to-image* correspondences. As shown in [36] and other recent works, finding correspondences in images is simpler than detecting *image-to-geometry* correspondences. Since a large number of *image-to-image* correspondence pairs can be detected in an automatic manner, we can deploy that information to speed up the overall image registration process, or to solve those cases where a single image covers a region where the surface has not sufficient shape features to allow an accurate selection of *image-to-geometry* correspondences. We have recently proposed an approach based on *Mutual Information* [57], a statistical measure of non-linear correlation between two data sources often used in medical images registration. The main idea is to use mutual information as a similarity measure between each image to be registered and some renderings of the geometric 3D model, in order to drive the registration in an iterative optimization framework. The problem can be divided into two sub-problems: finding an approximate solution, the *rough alignment* where we find a view of the geometric model that sufficiently matches the photographic image we want to align. For example, a rough alignment can be found with a simple greedy approach, by generating a large number of suitable initial views and



Figure 1.10: The image presents a result of the automatic image-to-geometry registration: given the image on the right, the proper projective transformation is computed by finding the best matching between the input image and renderings of the mesh with vertices colored according to a combined normal vector and accessibility shading factor [57].

checking which one best matches the input image. Then, this view is refined to find the optimal one, the *fine alignment*, by an iterative process that slightly modifies the view over the 3D model, produces a rendering and checks the similarity of the rendering with the input image. We demonstrate that some illumination-related geometric properties, such as surface normals, ambient occlusion and reflection directions can be efficiently used for this purpose, improving the convergence speed of the search. After a comprehensive analysis of such properties we proposed a way to combine these sources of information in order to improve the performance of our automatic registration algorithm [57]. The proposed approach can robustly cover a wide range of real cases and can be easily extended.

1.3.3 Mapping complex photographic detail on 3D models

Another issue concerning the high quality mapping of color information over a 3D model is how to manage the very high resolution and redundant color sampling. Just using plain standard or prosumer digital cameras one can easily obtain a very high resolution color sampling. Currently, the resolution of prosumer digital cameras allows to obtain a sampling that is an order of magnitude higher than the one of triangulation-based scanning devices; therefore, we might have at least 10 color/reflectance samples falling in the vicinity of each geometric sampled point. A medium size object (e.g., a statue) can be easily sampled with 50-100 images leading to hundred (or even thousand) of million pixels. The need to manage such texturing resolutions in real time opens several issues concerning the selection of the best texture mapping representation, how to extend multiresolution representation to hold both geometry and texture data, and the impact on the rendering speed. Beside the issues concerning the size of the raw data, the problem of mapping complex photographic detail on 3D models raises at least two main problems:

- *Efficient storing of color.* The possible higher density of the color data with respect to geometric data can make unsuitable direct approaches that store the color information over the primitives of the mesh (e.g., color-per-vertex). Texture mapping techniques would be more appropriate, but scanned meshes are usually highly complex and topologically unclean and these conditions make difficult the task of building a good texture parameterization.
- *Coherently mixing the photographic information.* Creating a high quality texture content is a critical step. Merging all the photos by a resampling process is a valid option, but it should not decrease the quality of the color information (e.g., resampling from multiple images often introduces blurring and ghost effects).

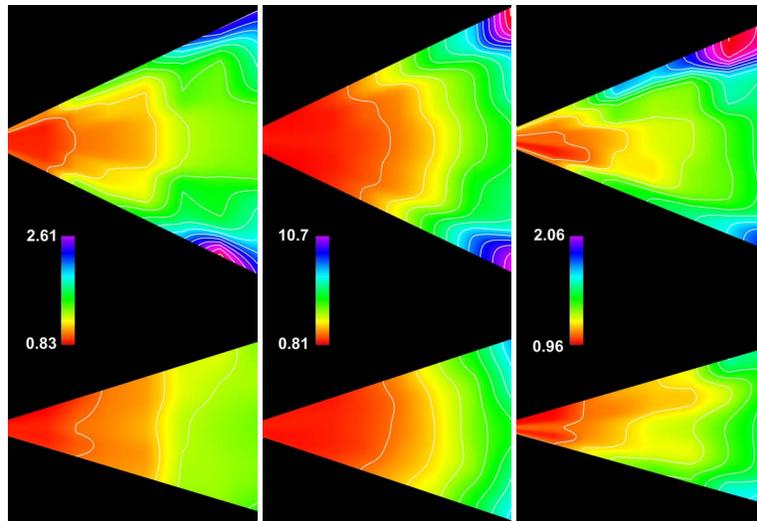


Figure 1.11: Some examples of the flash characterization obtained on three different cameras. The images show plots of an horizontal plane (top) and a vertical plane (bottom) intersecting the working camera space, for a Nikon reflex camera (Left), a Canon reflex camera (Center) and a compact Casio camera (Right).

Texture parameterization

For the specific task of building a mesh parameterization (taking in input the 3D mesh and the mapped photos) one approach could be to determine, given a set of images, an optimal partition of the mesh such that each surface patch could be mapped with (a portion of) one single input image. This approach has been used in a number of early papers (see for example [58, 59]). The main disadvantage is that the number of pieces in which the mesh is partitioned grows up rapidly with the number of involved photos [48].

Another possibility is to adopt classical mesh parameterization techniques that are able to build a good quality parameterization of the mesh over a texture space. Then, we can resample a texture from the input set of images, at the proper resolution required by the specific application. These approaches (see [60] for a survey on this subject) have often the drawback that, to avoid great distortion in the parameterization, the result is composed by many small texture pieces whose discontinuities can give rise to many visualization issues. A different approach has been recently proposed [61] that tries to overcome these limitations by computing the resulting parameterization in a continuous abstract texture domain that has the same topology of the input mesh and presents no discontinuity.

Resampling color data

Given a 3D model and the cameras associated to each input photo, we need a robust algorithm for color resampling from several input photographs, which can produce in output a coherently mixed blending of all the input images either as a texture map or as resampled per-vertex colors. A multi-variate blending approach has been recently proposed [62]. This blending framework allows: to use many different heuristics to characterize the image content (pixel by pixel) in terms of intrinsic quality of each sample; to produce, for each vertex on the surface or pixel on the output texture space, an optimal weighted average of all the available pixel samples which maps on the requested geometric location/parcel.

1.3.4 Advanced color data acquisition: sampling surface reflection properties

Obviously, we envision future systems able to encode not only the reflected radiation (apparent color), but able to sample the reflection properties of the surface. The goal is to move from apparent color acquisition towards BRDF sampling or at least the acquisition of approximations of the surface BRDF. To make those

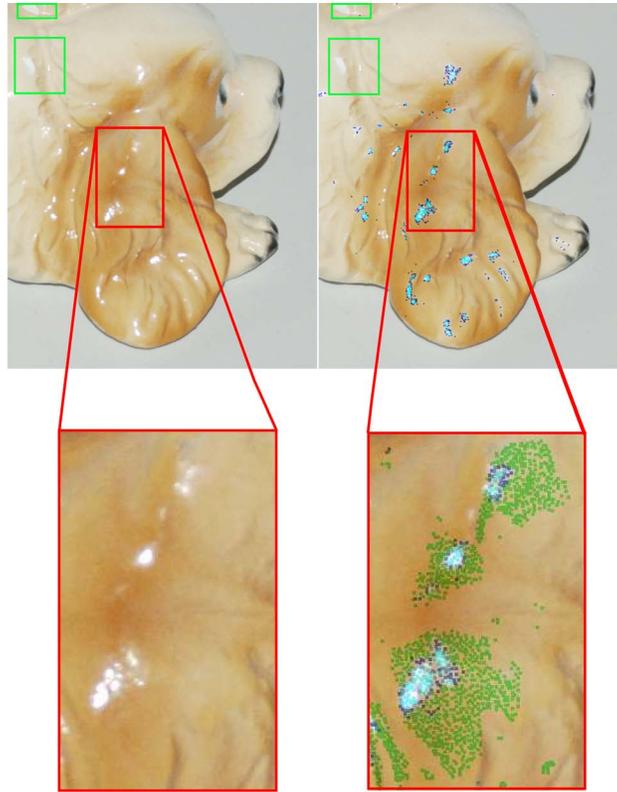


Figure 1.12: An example of highlights detection. Upper row: an input flash image and the same image after detection of highlights (blue is the highlight border, cyan is the internal highlight region). Lower row: two detail views, where highlights candidates selected by taking into account just geometric criteria (surface normals) are rendered in green and the ones more robustly detected are rendered in cyan.

solutions practical we need to improve current methods in terms of easy of operation, reduced dependency from highly controlled lighting environment, and reduced processing complexity.

An approximate approach has been recently proposed, with the aim of adopting a very easy to deploy lighting setup that should allow to remove lighting artifacts from the acquired photos. Flash light is a very easy way to illuminate an object or an environment, but it is rarely considered in most of the Computer Graphics and Computer Vision literature concerning color acquisition. This is due to the big amount of artifacts introduced by this specific light source, and to the difficulty in modeling its behavior in space. A simple method to use flash light in the context of color acquisition and mapping on digital 3D models has been recently proposed [63], based on a technique that allows to characterize the emission in the 3D space of the flash light (see Figure 1.11). The calibration process allows to acquire accurate information on how the light produced by a given flash is distributed in the 3D space sampled by the camera and how the color values sampled in that region are modified by the flash light. Moreover, given a calibrated flash and the extrinsic parameters for each single photo (location and direction of view of the camera), we are able to accurately estimate the flash position with respect to the framed object. We are thus able to apply automatic methods to detect and remove artifacts (specular highlights and shadows, see figures 1.12 and 1.13) from a set of images which are registered to a 3D model. These methods are integrated in the context of a color mapping framework and allow to obtain improved colored 3D models. These results fit very well the context of Cultural Heritage, where the acquisition of color has often to be performed on site, with very light constraints on the lighting setup. This approach can be considered as a first results to fill the gap between gathering just basic reflected color vs. complex BRDF acquisition methods.

Other promising approaches support sophisticated sampling of the reflection properties by adopting *image-based rendering* approaches: instead of producing accurate 3D encoding with poor-quality surface



Figure 1.13: An example of shadows detection: left, the original image; right, the shadow detection map.

reflection data, we could aim at acquiring and using the classical 2D media, but enhanced with sophisticated reflection encoding. We have a spectrum of possibilities, from the one-view and multiple-lighting approach of Polynomial Texture Maps (PTM) [64], to the more demanding multi-view approach granted by Light Field rendering [65]. In all these cases (see [66] for an overview), we get rid off the 3D model and adopt approaches which sample the artifact with a multitude of images. To produce any view requested at visualization time we process/interpolate the knowledge granted by the sampled image set. In most cases, this implies a huge quantity of image data to be acquired (thus long acquisition times are usually needed), stored and accessed in real time, for example making web-based visualization not easy.

1.4 MeshLab: an Open Source Tool for Processing 3D Scanned Data

As we have described in previous sections, processing 3D sampled data is not an easy task. We need several algorithms and the pipeline is neither fixed nor easy to be managed, particularly in the case the operator should not be an ICT expert. Even if a number of commercial solutions target the market of processing 3D sampled data, we found that the community is also searching for complete software solutions, possibly much cheaper than current SW tools. The introduction of new 3D scanners sold at the cost of a good digital camera makes the request of cheap SW solutions even more pressing. Moreover, most of the commercial packages in this field targets the reverse engineering application domain, that has different needs and priorities with respect to the requirements of the CH domain, so there is a demand for a set of tools that are more tailored to the processing of sampled 3D historical artifacts.

This objective was at the base of ISTI-CNR effort in designing and implementing MeshLab as an open source and extendable mesh processing tool [19, 41]. MeshLab was designed as a general 3D mesh processing system tool with the following primary objectives in mind:

- **Mesh processing oriented.** The system should try to stay focused on mesh processing tasks instead of aspiring to contribute to mesh editing and mesh design, where a number of fierce competitors already crowd the software arena (notably Blender, 3D Max, Maya, and many others).
- **Ease of use.** The tool should be designed so that users without high 3D modeling skills could use it (at least for the most basic functionalities).
- **Deep of use.** The tool should be designed so that advanced users can tweak and extend it by adding functionality and or by modifying all the involved parameters.
- **Efficiency.** 3D scanning meshes easily reach several millions of primitives, so the tool should be able to manage efficiently very large meshes.

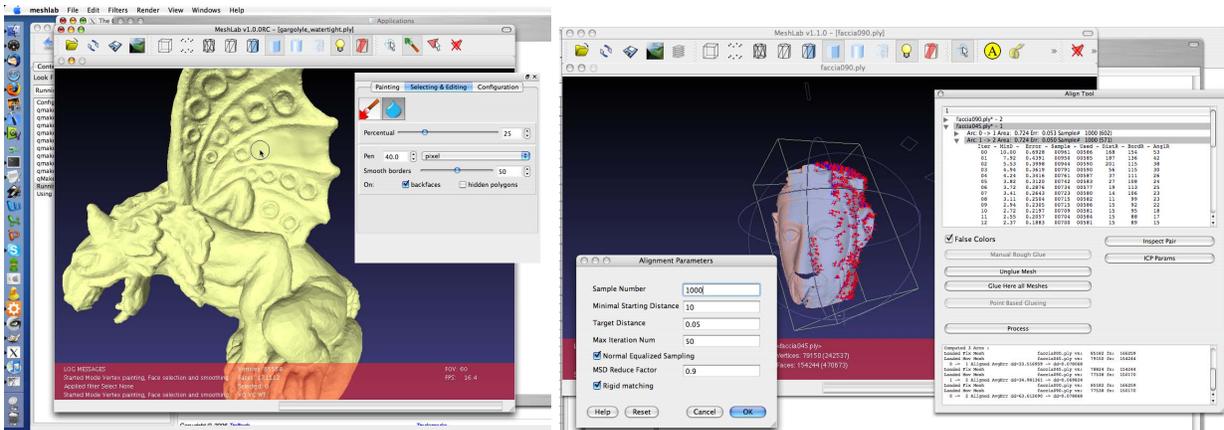


Figure 1.14: Snapshots of MeshLab in action: the mesh under processing is interactively displayed and the user can work on it by mean of a large set of unattended parametric filters, or by mean of interactive tools, like the one shown in the image on the left, where the user is smoothing out some features of the object with some simple mouse strokes; on the right we show a range maps alignment session.

As a result, MeshLab presents itself as an intuitive *mesh viewer* application, where digital 3D models, stored in a variety of formats, can be loaded and interactively inspected in a easy way, by simply dragging and clicking on the mesh itself. MeshLab supports a ever growing variety of 3D formats (all the most common formats are supported) to accommodate the broadest set of users. Once loaded a mesh, the user can work on it by mean of a large set of direct parametric filters, that performs unattended automatic task like surface reconstruction, smoothing, re-meshing or simplification, or by mean of interactive tools (for example, range maps registration).

The system has proved a success over any initial prediction. MeshLab is usually downloaded more than 6000 – 8000 times every month; MeshLab’s user community is thousands of users from all the world. Users come from hundred of universities and renowned commercial companies that have found MeshLab useful in many different contexts, widening the the original CH domain we were focusing to.

One of the interesting characteristic of MeshLab is the presence of all the basic resources for processing 3D scanning data. MeshLab already provides tools for: cleaning sampled range maps, performing range maps alignment, merging/reconstruction, simplification, transferral of color information from range maps to meshes, measuring differences between objects and providing texture parameterization algorithms. MeshLab is an evolving system, being one of the technologies under further development in the framework of the EC IP ”3D-COFORM” project (2009-2012) [21]. Some of the color management algorithms described in the previous section are among the new functionalities that we are including in MeshLab.

1.5 Efficient Visualization and Management of Sampled 3D Data

Some issues arise from the very dense sampling resolution granted by modern scanning devices. Being able to sample in the order of ten points per squared millimeter or more (in the case of triangulation-based systems) is of paramount value in many applications which need a very accurate and dense digital description. On the other hand, this information is not easy to process, render and transfer. Therefore, excessive data density becomes a problem for many applications. In effect, the availability of 3D scanned data was a driving force for intense research on more efficient geometric data management and rendering solutions. Some issues arising from the impressive increase in data complexity (and richness) provided by the evolution of 3D scanning technology are as follows: how to manage/visualize those data on commodity computers; how to improve the ease of use of the visualization tools (as potential users are often not expert with interactive graphics); finally, due to the accuracy of the 3D medium, we can think to use it as the main representation

media, able both to represent an artifact but also to integrate other multimedia information.

1.5.1 Simplification and multiresolution management of huge models

Data complexity can be managed by adopting a *data simplification* approach, reducing the data resolution at the expenses of a (controlled) loss of geometric accuracy. Many solutions have been proposed for the accurate simplification of 3D triangulated surfaces, usually based on the iterative elimination of selected vertices or faces, driven by an error-minimization cost function. This approach allows the construction of any level of resolution we need, usually with a rather expensive computation (from a few seconds to a few hours, depending on the solution used and the complexity of the initial surface) which has to be executed only once. Simplification is very handy to produce models which fit the specific application requirements (e.g., a simple small model for a web presentation which should be downloadable in a given short time, or a model to be used for an accurate rapid reproduction by a specific 3D printer).

Another approach is the so-called *multiresolution encoding*, (i.e., storing not just the final simplified model but all the intermediate results reached during the iterative simplification). All these intermediate results have to be encoded in an efficient data structure (the specific *multiresolution representation scheme*) that should allow the interactive application to extract different resolutions on the fly, supporting data feeding rates to the GPU compatible with real time applications (visualization is an example of those). A view-dependent variable resolution representation can be produced for each frame from these schemes, according to the current view specification (e.g., higher resolution for the portions in foreground, progressively lower resolution for data in the background) and the requested visualization accuracy.

Recent research on *multiresolution schemes* has produced a number of solutions based on higher granularity than previous methods (i.e., instead of focusing on single triangles, patches of triangles become the elementary entity). These solutions allow to manage huge 3D scanned models at interactive frame rate on consumer PC's [67, 68]. *Virtual Inspector* [56] is an example of a visualization system which adopt this approach to support inspection of large complex 3D models in real time (see Figure 1.9).

After around fifteen years of intense research on simplification and multiresolution technologies, we can nowadays consider those technologies sufficiently mature. But there is still some lack of information in the 3D scanning community (at the application level) on the potential of those methods. One of the more common negative concern raised by practitioners against 3D scanning is the size of the models obtained, which according to this common believe makes them unusable in real applications. This is not true, since the availability of simplification and multiresolution technologies can successfully cope with the data complexity of 3D scanned data. In this context, multiresolution encoding is of paramount value, supporting transparent real-time selection of the level of accuracy and producing at interactive rates a data density that best fit the current application requirements.

1.5.2 Mesh-based vs. point-based encoding and rendering

We have presented briefly the *triangle-based* simplification and multiresolution methodologies (i.e., data optimization/encoding strategies based on geometric processing applied to a triangle meshes). *Point-based* representations have also been used a lot in the 3D scanning domain, especially to present data acquired with TOF scanning devices. It is important to distinguish between naive point-based and the more sophisticated point-based representation and rendering approaches available. With the term *naive point-based* we mean the use of simple 3D points to render sampled data. This is the usual approach of many scanning devices, which usually render the samples as colored points, using the color channel to map the perceived reflectivity of the sampled surface or the estimated accuracy of the sample.

Naive point rendering is very easy to implement, but it is also a major responsible of the very poor acceptance of 3D scanning in several applicative domains. Just splatting points on the screen gives very poor images, where it is not easy at all to understand the relations between different elements, perception of depth is lacking, it is very hard to spot errors or inaccuracies of the data (see Figure 1.15). This does not

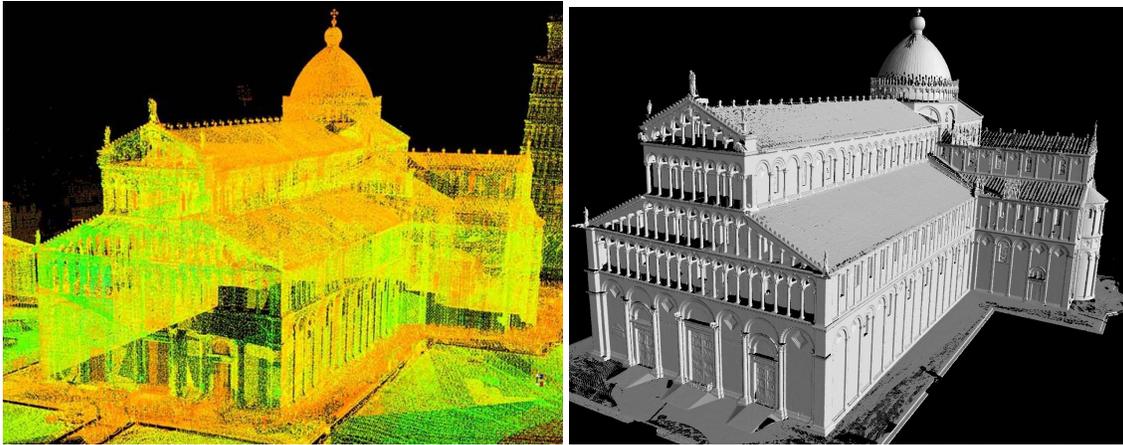


Figure 1.15: An example of two different visualization modes: on the left we render just the sampled points, on the right we present a rendering where shading of the surface element plays an important role for an improved insight over the details of the represented architecture (from a scanned model of the Cathedral of Pisa, Italy).

mean that point-based rendering is a wrong solution, but that choosing a naive implementation can be a very bad decision.

First, we should remember that even if we endorse a point-based approach, all the processing phases presented in Section 1.2 are valid and have to be performed in an accurate and correct way. The several point clouds (or range maps) acquired have to be aligned precisely one to the other and globally (please note that the alignment results can be checked and assessed only when we have the possibility to render the points as a surface, adding shading). A merging phase should be applied even if we want to endorse a point-based representation: it is not sufficient to simply join the several point clouds in a bigger set, because sampled data are redundant and contain noise; we have already seen that a merging phase (performing an intelligent interpolation among sampled data) can improve the data and reduce unnecessary redundancy. Moreover, we need to adopt multiresolution to make large dataset tractable (as an example, the Pisa Dome shown in Figure 1.15 is around 200 M samples after merging).

Once established that we have to process the dataset even when a point-based approach is endorsed, let us focus on visualization constrains. Even when we use points, we should be able to render color (this is very easy, since we can assign a single color to each point) but also to render the surface slope associated to each point. This means that the representation should store a normal vector for each sample, and this vector has to be computed by integrating a small region around the given sample.

Many different advanced point-based representations and rendering methodologies have been presented in the last few years. Presenting an overview of the extensive research results on point based graphics is well beyond the focus of this chapter; this specific domain is the topic of a series of successful symposia, the *EG Point Based Graphics* series; interested readers can consult those symposia proceedings or a recent tutorial on point-based techniques [69]. Point-based representation as well can be managed in an efficient manner by endorsing simplification or multiresolution methods. The latter are usually based on hierarchical structures that allow to encode the dataset at different levels of detail and to extract view-dependent representation very easily in real time [70, 71, 72].

1.5.3 Usability of Virtual Heritage worlds

Easy of use of visual CH tools oriented to ordinary people (that are still not very competent with 3D graphics and computer games, especially if not part of the young generation) is an important factor for their success. One of the most complicated actions to perform nowadays with 3D tools is to drive navigation in the virtual space, especially when we have a clear objective (e.g., I want to reach a specific location in the scene and

see my focus of interest from a specific view). Therefore, free navigation should be requested only in those cases where this action really adds something to the learning experience. The risk is to have the visitor losing orientation (e.g., discovering himself lost in void space, maybe just because he turned his back to the scene and is erroneously looking at the sky), losing faith to be able to drive the inspection or navigation session and quitting the system.

Other important characteristics of a visualization system are its flexibility, completeness and configurability. To fulfill this objective developers could be induced to design complicated systems characterized by a very complete set of functionalities and involute user interfaces (for an example, consider the commercial 3D modeling systems). Conversely, while designing our *Virtual Inspector* tool [56] as a system oriented to non-expert users (e.g., museum visitors), our approach was to define a restricted set of functionalities and to provide the tool with an easy installation interface for the selection of the subset of these functionalities that the designer of the specific installation want to endorse (e.g., to build up a new museum kiosk).

Another important factor of success is nowadays web availability of those resources, to allow a much wider community of users to experiment, navigate and enjoy the reconstructed scenes. Implementing a Virtual Heritage reconstruction to make it deployable on the web introduces a number of design constraints, but the technology is going to be nearly mature to support that type of distribution channel.

1.5.4 Not just 3D data: adding other knowledge

Visualization tools usually focuses on just the visual presentation of the shape characteristics of the artifact. This is not enough if we want to provide a comprehensive presentation of the global knowledge available on the artifact. On the other hand, the 3D shape can become some sort of visual 3D map which allows us to integrate, link and present all the available information in an intuitive and visually pleasing way. The goal is therefore to transform standard 3D browsers into tools able to connect the shape (or region of the latter) with all the available multimedia (MM) data that is linked with the artifact. This opens the wider topic on how to support and implement data annotation on 3D models.

Hot spots can be a very handy resource to associate multimedia data to any point or region of a 3D model. This allows to design interactive presentations where the 3D model becomes a natural visual index to historical/artistic information, for example presented using standard HTML format and browsers. But hot spots are only a first step, which implements a point-wise association metaphor. We should devise more flexible and powerful instruments to associate information to 3D meshes (e.g., supporting links between sub-regions of the surface to any possible MM document). Therefore, tools should provide easy to use features for the segmentation of the shape model in components. Segmentation can be driven by pure shape-oriented criteria (usually, working on surface curvature is one of the most diffuse approach) but it should also include the user in the loop, by giving him the lead in driving the segmentation process according to the metadata that characterize the specific artifact and the message he wants to convey with the specific segmentation.

Data presentation of a multitude of information tokens can become an issue (many different types of point-wise links, associated to different types of information; several different segmentations available for the same object, each one focusing on a different interpretation or context). Intelligent approaches to data hiding and visualization will have to be endorsed also in this type of applications, as it has already been the case of other scientific visualization or data analytics contexts. Moreover, visualization instruments should be extended to provide tools supporting the easy integration and update of the data associated to the 3D digital model, supporting a democratic and cooperative approach such as the one at the base of the Wikipedia effort.

1.5.5 Presenting 3D data on the web

In the modern word, we cannot avoid to consider the issues related to the web-based access and distribution of 3D data. The peculiar aspect of scanned data with respect to other 3D models is the average size of the models. To make these data usable on the web we should either deploy efficient geometric compression

technology [73], or to adopt remote rendering approaches [74]. A web-based application which could boost considerably the usage of 3D scanning is GoogleEarth (or the similar systems), since those applications could allow very nice opportunities for announcing on the web the availability of 3D models, showing to the users their availability while navigating the geographical space. Another very promising platform are the WebGL specifications for the inclusion of 3D graphics among the data directly managed by common web browsers [75]. The new generation of web browsers (now in beta version) will therefore support natively digital 3D content, without the need to install specific plug-ins. Some examples of how 3D content could be represented and processed using WebGL are presented on [76]

1.6 3D Digitization: How to Improve Current Procedures and Make It More Practical and Successful

While 3D scanning hardware and software technologies have considerably evolved in the last years and their use in the Cultural Heritage field has gained consent and acceptance, these technologies are far from being easy to use and error-free. Many problems can arise during a scanning campaign; most of these issues are related to various deficiencies in the technologies and the improper background/skills of the operators. Some of these problems have already been briefly mentioned through the chapter; we discuss them in detail in this section.

1.6.1 Technology - Limitations perceived by practitioners

First of all, 3D scanning technologies are not able to capture many kind of materials that may occur in the CH field: transparent or semi-transparent surfaces (like glass, jewels and some stones); mirroring, polished and very shiny objects (like metals); fluff and fuzzy substances like feathers, furs or some tissues. All those materials are quite difficult to sample with the current off-the-shelf optical technologies. For some of the previous cases, experimental research projects have shown the feasibility of technological solutions that overcome these limitations (in some cases, by adopting enhanced 2D image media, rather than a pure 3D encoding); in several cases, their practical applicability to real projects has still to be assessed.

The working space of the devices is another issue. The CH domain is a very good example of an application where the objects of interest can span the entire interval: from tiny objects (few millimeters) to an entire building or even an archaeological site or a city. With such a wide domain, an important feature of a scanning device would be the flexibility of the working space supported. Conversely, the devices often offer just a very restricted working space (or require a time-consuming calibration to select a slightly different working range). This forces to select different devices for the different acquisition contexts, increasing the technical skills required to master technology and the overall cost. In this sense, current acquisition methodologies based on pure images (multi stereo matching and similar approaches) present a big advantage with respect to the classical scanning devices, due to the much wider working space of digital cameras (changing focal lenses is much easier than switching to a different scanning device).

Moreover, as already noted before, most of current scanning technologies focus only on shape, ignoring the issues related to the acquisition of the optical appearance of the scanned surface. Even when the adopted technologies are able to capture the “color” (for example by mapping photos over the surface), in almost all the cases this leads only to the acquisition of the apparent reflected color, without considering the reflectance properties of the surface (e.g., its shininess). The acquisition of a correct representation of the reflectance properties of a surface together with its 3D shape is still the domain of research experiments and it is not an off-the-shelf technology.

Another sensible issue of 3D Scanning is the cost of the hardware equipment: usually it is very high and it can become prohibitive for many low-budgeted cultural heritage institutions. On the other hand, the field is enlarging and very cheap solutions are appearing on the market.

On the positive side we have to note that the research in this field is very active, so we can easily hope that in the next future the capabilities of 3D scanning hardware will improve gradually covering the

shortcomings listed above.

1.6.2 Misuse of technology

The people involved in a scanning campaign can be roughly partitioned in two sets: *technical staff*, who actually perform the scanning task with a not-so-strong CH background and sensibility; *CH operators/experts*, who knows very well the specific field and the digitization objectives, but often lacks a deep knowledge of the technological details. Obviously there are some notable exceptions, typically in most of the successful 3D scanning projects run so far. But, especially if the field will increase in terms of number of devices sold and application testbeds run, the situation will turn more and more towards the case of users with a limited technological background. Designers of software tools should keep this in mind while designing the tools.

Selecting the wrong device

As we have sketched in the previous sections, there are many possible technologies available, each one with its own pros and cons. We should underline that scanning system producers are usually very poor in illustrating the conditions which make a given device not fit for the planned task. External constraints (like for example the availability of a given device or of consolidated experience with a specific technology) might affect the choice of the preferred hardware leading to the selection of a non-optimal device.

Wrong data post-processing

The wish of providing clean and nice results to purchaser of a scanning service often causes the undocumented (and often not required) editing of the acquired data. 3D scanning technologies are often unable to completely recover the whole surface, leaving holes and un-sampled regions in the final model. Smoothing can reduce the sampling noise in the final model, but it can also cancel important high-frequency details (e.g., traces of deterioration of the surface, traces of the tools used to produce the artwork). Any editing action over the originally acquired data should be documented in the most evident way in order to make it possible to distinguish between *ground truth* sampled data and the parts that have been added or heavily edited by the subsequent geometric processing phases. Excessive data smoothing or inaccurate simplification are clear examples of actions which could destroy data accuracy and trustability.

1.6.3 Better management of 3D data

On the other hand, lack or incomplete technical knowledge on 3D scanning can cause various inconvenience in the overall process of creating trusted digital reproduction of works of art.

Evaluating and assessing data quality

Being able to evaluate in a objective way the final quality of a 3D scanned model is a basic resource for a wide adoption of this technology. Conversely to a more traditional media such as photography, where established protocols and procedures exists and the quality can be determined by an analysis of the (digital) photo and its provenance data, quality assessment of the final digital 3D model is much more complex. Quality depends on a large number of factors: scanning HW characteristics, modality of use of the scanning device, pipeline and parameters of the SW post-processing which transforms the raw data in the final 3D model. Therefore the whole pipeline, including HW and SW used to produce the digital object, has to be considered while assessing the quality of the digital model. Provenance data are therefore more complex than the ones characterizing the 2D media (photographs). While common practices for the use of general 3D visualization in the research and communication of Cultural Heritage have been already discussed in the London Charter [77], there is still an absence of clear protocols for the management of 3D scanning technologies and data. Some attempts should be done to define the concept of provenance data for 3D models, in a sufficiently general and standardized way. This is one of the major focus of the EC research

initiative IST IP "3D-COFORM" [21]. 3D-COFORM technologies are steered towards a comprehensive management of provenance data, by providing integrated management at the level of both the modeling tools and the repository for 3D assets and metadata [78].

The availability of free and standard tools supporting the quality assessment (ideally, by considering both geometry and reflectance data) would be of paramount value, especially for the usually not-technical people who order a scanning campaign. This is an area that desperately needs research and practical results made available to the community. Even a well informed consumer at this point does not have the tools and standards available to make a purchasing decision and a good assessment of the results that are produced by a (usually expensive) project commissioned to an external service company.

Preserving the data for the future

An interesting point of discussion is which could be the most fruitful use of the acquired data; a given artifact can be scanned now for a specific application, but once we have the digital model we can envision many different applications of the same 3D model in the future. Therefore, the results of scanning efforts should be made available for future uses, even by different entities. This can be also a way to recover from the cost of the digitization: CH institutions should develop policies to play a role in the market of digital 3D content, finding on the market the financial resources to perform 3D digitization.

Another consideration, that is often neglected, is the *long term preservation* of these data. While for traditional media, like written material, photography, and films, the issues about the preservation, classification and future access to the acquired data is a rather well studied subject, there are no established practices for the preservation of digital 3D content. Without going into details, let us just mention here the issues related to the format of the delivered 3D data: often the results of a 3D scanning campaign are delivered only as files encoded in a closed format, accessible only through proprietary software whose longevity is not guaranteed at all. We are seriously risking that the data produced in several important initiatives are buried in a hidden format without the possibility of being utilized in the near future.

1.7 Conclusions

The chapter presented the software technologies available for processing and visualization of sampled 3D data. Due to the vast amount of material published in the current literature on this subject, we were forced to present here just a selection of the approaches proposed. Therefore, the bibliography cites only a very small subest of the available literature (otherwise the extent of the bibliography could have easily become excessively long). One of the focus of the presentation has been to give more emphasis to the open problems or to subjects which are currently under study, rather than presenting in details the algorithms mentioned. Therefore, the current chapter should be read by consulting also the papers cited in literature, according to the interests of the reader. Some of those papers are very good resources for gathering a state of the art of the specific problem and to get a more complete list of citations to the other works in the selected sub-domain. Finally, as usually happens with review papers on a subject where the authors have dedicated time and efforts, we have probably given an emphasis on our own results and papers; we apologize with the reader for that but, as we explicitly mentioned in the text, some results are presented as a representative of several other research efforts which produced comparable outputs.

Acknowledgement

We acknowledge the financial support of the EC IST IP project "3D-COFORM" (IST-2008-231809) and the project "START" (funded by the Tuscany Region, 2008-2011).

Bibliography

- [1] F. Blais, “A review of 20 years of range sensor development,” in *Videometrics VII, Proceedings of SPIE-IS&T Electronic Imaging, SPIE Vol. 5013*, pp. 62–76, 2003.
- [2] F. Remondino and S. El-Hakim, “Image-based 3D modelling: a review,” *The Photogrammetric Record*, vol. 21, no. 115, pp. 269–291, 2006.
- [3] P. Cignoni and R. Scopigno, “Sampled 3D models for ch applications: A viable and enabling new medium or just a technological exercise?,” *ACM J. Comput. Cult. Herit.*, vol. 1, no. 1, pp. 1–23, 2008.
- [4] B. Curless and S. Seitz, “3D Photography,” in *ACM SIGGRAPH 00 Course Notes, Course No. 19*, 2000.
- [5] G. Sansoni, M. Trebeschi, and F. Docchio, “State-of-the-art and applications of 3D imaging sensors in industry, cultural heritage, medicine, and criminal investigation,” *Sensors*, vol. 9, no. 1, pp. 568–601, 2009.
- [6] R. Lange, P. Seitz, A. Biber, and R. Schwarte, “Time-of-flight range imaging with a custom solid-state image sensor,” in *Proceedings of the SPIE Vol. 3823*, pp. 180–191, 1999.
- [7] M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S. M. Seitz, “Multi-view stereo for community photo collections,” in *Proceedings of ICCV 2007*, pp. 1–8, 2007.
- [8] H. Bay, A. Ess, T. Tuytelaars, and L. van Gool, “Speeded-up robust features (surf),” *Computer Vision and Image Understanding (CVIU)*, vol. 110, no. 3, pp. 346–359, 2008.
- [9] NextEngine, Inc., “The NextEngine laser scanning system.” More info on: <https://www.nextengine.com/>, 2010.
- [10] M. Vergauwen and L. V. Gool, “Web-based 3D reconstruction service,” *Mach. Vision Appl.*, vol. 17, no. 6, pp. 411–426, 2006.
- [11] M. Vergauwen and L. V. Gool, “The Arc3D web-based reconstruction tool.” More info on: <http://www.arc3d.be/>, 2010.
- [12] F. Blais, J. Taylor, L. Cournoyer, M. Picard, L. Borgeat, L.-G. Dicaire, M. Rioux, J.-A. Beraldin, G. Godin, C. Lahnanier, and G. Aitken, “Ultra-high resolution imaging at 50m using a portable xyz-rgb color laser scanner,” in *International Workshop on Recording, Modeling and Visualization of Cultural Heritage (Ascona, Switzerland, May 22-27)*, 2005.
- [13] F. Bernardini and H. E. Rushmeier, “The 3D Model Acquisition Pipeline,” *Computer Graphics Forum*, vol. 21, pp. 149–172, March 2002.
- [14] INUS Technology, “Rapidform™- 3D Scanning and Inspection Software from INUS Technology, Inc. and Rapidform, Inc.,” More info on: <http://www.rapidform.com/>, 2010.
- [15] InnovMetrics, “PolyWorks™- inspection and modeling solutions.” More info on: <http://www.innovmetric.com/>, 2010.
- [16] Raindrop Geomagic, “Geomagic™studio: Reverse Engineering and Custom Design Software.” More info on: <http://www.geomagic.com/>, 2010.
- [17] B. Curless, “Vrippack users guide,” Tech. Rep. TR-UUCS-04-019, University of Washington (<http://www-graphics.stanford.edu/software/vripp/>), November 25 2006.
- [18] M. Callieri, P. Cignoni, F. Ganovelli, C. Montani, P. Pingi, and R. Scopigno, “VCLab’s tools for 3D range data processing,” in *VAST 2003* (A. C. D. Arnold and F. Niccolucci, eds.), (Bighton, UK), pp. 13–22, Eurographics, Nov. 5-7 2003.

- [19] P. Cignoni, M. Callieri, M. Corsini, M. Dellepiane, F. Ganovelli, and G. Ranzuglia, "Meshlab: an open-source mesh processing tool," in *Sixth Eurographics Italian Chapter Conference*, pp. 129–136, 2008.
- [20] "EPOCH: The European Network of Excellence on ICT Applications to Cultural Heritage (IST-2002-507382)." More info on: <http://www.epoch-net.org/>, 2008.
- [21] "EU IP 3DCOFORM: Tools and expertise for 3D collection formation (FP7/2007-2013 grant 231809)." More info on: <http://www.3d-coform.eu/>, 2010.
- [22] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, pp. 239–258, Feb. 1992.
- [23] Y. Chen and G. Medioni, "Object modelling by registration of multiple range images," *International Journal of Image and Vision Computing*, vol. 10, pp. 145–155, Apr. 1992.
- [24] K. Pulli, "Multiview registration for large datasets," in *Proc 2nd Intl Conf. on 3D Digital Imaging and Modeling*, pp. 160–168, IEEE, 1999.
- [25] N. Gelfand, N. J. Mitra, L. J. Guibas, and H. Pottmann, "Robust global registration," in *In M. Desbrun and H. Pottmann, editors, Eurographics Association, ISBN 3-905673-24-X*, pp. 197–206, 2005.
- [26] G. Guidi, J.-A. Beraldin, and C. Atzeni, "High-accuracy 3D modeling of cultural heritage: the digitizing of donatello's "maddalena"," *IEEE Transactions on Image Processing*, vol. 13, no. 3, pp. 370 – 380, 2004.
- [27] P. Cignoni, C. Montani, C. Rocchini, and R. Scopigno, "External memory management and simplification of huge meshes," *IEEE Transactions on Visualization and Computer Graphics*, vol. 9, no. 4, pp. 525–537, 2003.
- [28] M. Garland and P. Heckbert, "Surface simplification using quadric error metrics," in *SIGGRAPH 97 Conference Proceedings*, Annual Conference Series, pp. 209–216, Addison Wesley, Aug. 1997.
- [29] H. Hoppe, "New quadric metric for simplifying meshes with appearance attributes," in *Proc. IEEE Conference on Visualization (VIS99)*, (New York), pp. 59–66, ACM Press, Oct. 25–28 1999.
- [30] H. P. A. Lensch, J. Kautz, M. Goesele, W. Heidrich, and H.-P. Seidel, "Image-based reconstruction of spatial appearance and geometric detail," *ACM Transaction on Graphics*, vol. 22, pp. 234–257, Apr. 2003.
- [31] 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, pp. 131–144, Addison Wesley, July 24-28 2000.
- [32] R. J. Campbell and P. J. Flynn, "A Survey Of Free-Form Object Representation and Recognition Techniques," *Computer Vision and Image Understanding*, vol. 81, no. 2, pp. 166–210, 2001.
- [33] A. Fasano, P. Pingi, P. Cignoni, C. Montani, and R. Scopigno, "Automatic registration of range maps," *Computer Graphics Forum (Proc. of Eurographics '05)*, vol. 24, no. 3, pp. 517–526, 2005.
- [34] U. Castellani, M. Cristani, S. Fantoni, and V. Murino, "Sparse points matching by combining 3D mesh saliency with statistical descriptors," *Computer Graphics Forum*, vol. 27, no. 2, pp. 643–652, 2008.
- [35] D. Aiger, N. J. Mitra, and D. Cohen-Or, "4-points congruent sets for robust pairwise surface registration," in *SIGGRAPH '08: ACM SIGGRAPH 2008 papers*, (New York, NY, USA), pp. 1–10, ACM, 2008.
- [36] G. H. Bendels, P. Degener, R. Wahl, M. Körtgen, and R. Klein, "Image-based registration of 3D-range data using feature surface elements," in *5th International Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST 2004)* (Y. Chrysanthou, K. Cain, N. Silberman, and F. Niccolucci, eds.), pp. 115–124, Eurographics, December 2004.
- [37] J. Davis, S. Marshner, M. Garr, and M. Levoy, "Filling holes in complex surfaces using volumetric diffusion," in *First Int. Symp. on 3D Data Processing, Visualization and Transmission (3DPVT'02)*, pp. 428–438, IEEE Comp. Soc., 2002.
- [38] M. Kazhdan, M. Bolitho, and H. Hoppe, "Poisson surface reconstruction," in *SGP '06: Proceedings of the fourth Eurographics symposium on Geometry processing*, (Aire-la-Ville, Switzerland, Switzerland), pp. 61–70, Eurographics Association, 2006.
- [39] T. Ju, "Robust repair of polygonal models," *ACM Trans. on Graphics*, vol. 23, no. 3, pp. 888–895, 2004.

- [40] J. Podolak and S. Rusinkiewicz, "Atomic volumes for mesh completion," in *EG/ACM Symposium on Geometry Processing (SGP)*, pp. 33–41, Eurographics, 2005.
- [41] P. Cignoni, "MeshLab: an open source, portable, and extensible system for the processing and editing of unstructured 3D triangular meshes." More info on: <http://meshlab.sourceforge.net/>, 2010.
- [42] P. Liepa, "Filling holes in meshes," in *EG/ACM Symposium on Geometry Processing (SGP)*, pp. 200–206, Eurographics, June 2003.
- [43] L. S. Tekumalla and E. Cohen, "A hole filling algorithm for triangular meshes," Tech. Rep. TR-UUCS-04-019, University of Utah, 2004.
- [44] A. Sharf, M. Alexa, and D. Cohen-Or, "Context-based surface completion," *ACM Trans. on Graphics*, vol. 23, no. 3, pp. 878–887, 2004.
- [45] G. H. Bendels, R. Schnabel, and R. Klein, "Detail-preserving surface inpainting," in *The 6th International Symposium on Virtual Reality, Archaeology and Cultural Heritage* (M. Mudge, N. Ryan, and R. Scopigno, eds.), (Pisa, Italy), pp. 41–48, Eurographics Association, 2005.
- [46] C. Donner, T. Weyrich, E. d'Eon, R. Ramamoorthi, and S. Rusinkiewicz, "A layered, heterogeneous reflectance model for acquiring and rendering human skin," *ACM Trans. Graph.*, vol. 27, no. 5, pp. 1–12, 2008.
- [47] A. Ghosh, T. Hawkins, P. Peers, S. Frederiksen, and P. Debevec, "Practical modeling and acquisition of layered facial reflectance," *ACM Trans. Graph.*, vol. 27, no. 5, pp. 1–10, 2008.
- [48] 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*, (Erlangen (D)), pp. 419–426, IOS Press, Nov. 20 - 22 2002.
- [49] K. Umeda, M. Shinozaki, G. Godin, , and M. Rioux, "Correction of color information of a 3D model using a range intensity image," in *Fifth International Conference on 3-D Digital Imaging and Modeling (3DIM 2005)*, pp. 229 – 236, June 13-16 2005.
- [50] A. Troccoli and P. K. Allen, "Relighting acquired models of outdoor scenes," in *Fifth International Conference on 3-D Digital Imaging and Modeling (3DIM 2005)*, pp. 245 – 252, June 13-16 2005.
- [51] H. Rushmeier and F. Bernardini, "Computing consistent normals and colors from photometric data," in *Proc. Second Int. Conf. on 3D Digital Imaging and Modeling*, (Ottawa, Canada), pp. 99–108, 1999.
- [52] P. Debevec, C. Tchou, A. Gardner, T. Hawkins, C. Poullis, J. Stumpfel, A. Jones, N. Yun, P. Einarsson, T. Lundgren, M. Fajardo, , and P. Martinez, "Digitizing the parthenon: Estimating surface reflectance properties of a complex scene under captured natural illumination," in *VMV04 Proc.*, p. 99, 2004.
- [53] 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*, vol. RA-3, Aug. 1987.
- [54] 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*, vol. 21, pp. 619–628, sep 2005.
- [55] M. Dellepiane, M. Callieri, F. Ponchio, and R. Scopigno, "Mapping highly detailed color information on extremely dense 3D models: the case of davids restoration," *Computer Graphics Forum*, vol. 27, no. 8, pp. 2178–2187, 2008.
- [56] M. Callieri, F. Ponchio, P. Cignoni, and R. Scopigno, "Virtual inspector: a flexible visualizer for dense 3D scanned models," *IEEE Computer Graphics and Applications*, vol. 28, no. 1, pp. 44–55, 2008.
- [57] M. Corsini, M. Dellepiane, F. Ponchio, and R. Scopigno, "Image-to-geometry registration: a mutual information method exploiting illumination-related geometric properties," *Computer Graphics Forum*, vol. 28, no. 7, pp. 1755–1764, 2009.
- [58] C. Rocchini, P. Cignoni, C. Montani, and R. Scopigno, "Multiple textures stitching and blending on 3D objects," in *Rendering Techniques '99* (D. Lischinsky and G. Ward, eds.), pp. 119–130, Springer-Verlag Wien, 1999.
- [59] C. Rocchini, P. Cignoni, C. Montani, and R. Scopigno, "Acquiring, stitching and blending diffuse appearance attributes on 3D models," *The Visual Computer*, vol. 18, no. 3, pp. 186–204, 2002.
- [60] M. S. Floater and K. Hormann, "Surface parameterization: a tutorial and survey," in *Adv. in Multires. for Geom. Model.* (N. A. Dodgson, M. S. Floater, and M. A. Sabin, eds.), Math. and Vis., pp. 157–186, Springer, 2005.

- [61] N. Pietroni, M. Tarini, and P. Cignoni, “Almost isometric mesh parameterization through abstract domains,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 16, no. 4, pp. 621–635, 2010.
- [62] M. Callieri, P. Cignoni, M. Corsini, and R. Scopigno, “Masked photo blending: mapping dense photographic dataset on high-resolution sampled 3D models,” *Computers and Graphics*, vol. 32, no. 4, pp. 464–473, 2008.
- [63] M. Dellepiane, M. Callieri, M. Corsini, P. Cignoni, and R. Scopigno, “Improved color acquisition and mapping on 3D models via flash-based photography,” *ACM Journ. on Computers and Cultural heritage*, vol. 2, pp. 1–20, Feb. 2010.
- [64] T. Malzbender, D. Gelb, and H. Wolters, “Polynomial texture maps,” in *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, (New York, NY, USA), pp. 519–528, ACM Press, 2001.
- [65] M. Levoy and P. Hanrahan, “Light Field Rendering,” in *Computer Graphics Proceedings, Annual Conference Series, 1996 (ACM SIGGRAPH '96 Proceedings)*, pp. 31–42, 1996.
- [66] M. Levoy, “Light fields and computational imaging,” *IEEE Computer*, vol. 39, no. 8, pp. 46–55, 2006.
- [67] P. Cignoni, F. Ganovelli, E. Gobbetti, F. Marton, F. Ponchio, and R. Scopigno, “Adaptive tetrapuzzles: Efficient out-of-core construction and visualization of gigantic multiresolution polygonal models,” *ACM Trans. on Graphics (SIGGRAPH 2004)*, vol. 23, no. 3, pp. 796–803, 2004.
- [68] P. Cignoni, F. Ganovelli, E. Gobbetti, F. Marton, F. Ponchio, and R. Scopigno, “Batched multi triangulation,” in *IEEE Visualization 2005*, pp. 27–35, 2005.
- [69] L. Kobbelt and M. Botsch, “A survey of point-based techniques in computer graphics,” *Computers & Graphics*, vol. 28, no. 6, pp. 801 – 814, 2004.
- [70] S. Rusinkiewicz and M. Levoy, “QSplat: A multiresolution point rendering system for large meshes,” in *Comp. Graph. Proc., Annual Conf. Series (SIGGRAPH 00)*, pp. 343–352, ACM Press, July 24-28 2000.
- [71] E. Gobbetti and F. Marton, “Layered point clouds – a simple and efficient multiresolution structure for distributing and rendering gigantic point-sampled models,” *Computers & Graphics*, vol. 28, December 2004.
- [72] E. Gobbetti and F. Marton, “Far Voxels – a multiresolution framework for interactive rendering of huge complex 3D models on commodity graphics platforms,” *ACM Transactions on Graphics*, vol. 24, pp. 878–885, August 2005. Proc. SIGGRAPH 2005.
- [73] J. Rossignac, “Surface simplification and 3D geometry compression,” in *Handbook of Discrete and Computational Geometry, CRC Press, 1997, 2004* (J. E. Goodman and J. O’Rourke, eds.), vol. 2, 2004.
- [74] D. Koller, M. Turitzin, M. Levoy, M. Tarini, G. Croccia, P. Cignoni, and R. Scopigno, “Protected interactive 3D graphics via remote rendering,” *ACM Trans. Graph.*, vol. 23, no. 3, pp. 695–703, 2004.
- [75] Khronos Group, “WebGL - OpenGL ES 2.0 for the Web.” More info on: <http://www.khronos.org/webgl/>, 2010.
- [76] Di Benedetto, Marco, “SpiderGL - 3D Graphics for Next-Generation WWW (a JavaScript 3D Graphics library which relies on WebGL for realtime rendering).” More info on: <http://www.spidergl.org/>, 2010.
- [77] The London Charter, “For the use of 3D visualization in the research and communication of cultural heritage.” More info on: <http://www.londoncharter.org/>, 2010.
- [78] M. Theodoridou, Y. Tzitzikas, M. Doerr, Y. Marketakis, and V. Melessanakis, “Modeling and querying provenance by extending CIDOC CRM,” *Distributed and Parallel Databases*, vol. 27, no. 2, pp. 169–210, 2010.