Chapter 0

Point Set Analysis: an image analysis point of view for Rapid Prototyping technologies

Nicolas Loménie, Daniel Racoceanu and Georges Stamon*

CNRS, *University Paris Descartes
France, Singapore

1. Introduction

This chapter is dedicated to the review and presentation of an emerging paradigm in the world of computer vision: how to conciliate image analysis algorithms with computer graphics representations of shapes as point sets or meshes? In other words, what are the latest attempts to adapt algorithms performing on regular grids with fixed neighborhood system onto unorganized point sets (UPS)? In (Lomenie & Stamon, 2011) it is shown that the underlying theoretical challenges to be solved boil down to the topological definition of continuity and neighborhood in these kinds of mathematical spaces. In particular, this work wraps up a decade of various contributions and developments about shape analysis and representation within unorganized point sets (UPS) carried out by the authors in (Lomenie et al., 2003; Lomenie, 2004; Lomenie & Stamon, 2008, 2011) in the framework of mathematical morphology.

Beyond theoretical aspects, a wide scope of practical studies already make use of intelligent solutions designed to process visual data represented by geometric point sets, in particular in the world of rapid prototyping. Digging into recent literature about the topic makes it clear that industrial countries like China are specifically interested in this emerging topic, even though first prototypes of 3D printers date back to approximatively the mid-nineties. While still lightly lagging behind in terms of creativity and design, there are high chances that these emerging superpowers will catch up with design and innovation by their ability to quickly devise machines and systems able to manufacture products customized according to the individual consumer wishes. In (Cui et al., 2010), an integrated, distributed, web-based rapid product design platform enables design resource sharing and reusability of various CAD models for rapidly responding to various customer requirements. Similarly, many new multimedia and industry applications will rely on these kinds of technologies. The need for new interactions in multimedia applications is highlighted in (Li et al., 2011; Lin et al., 2010) by calling for considerable progress in terms of content production and representation for television and motion picture. Recent applications can be found in 3D-ink printing (Stanic and Lozo, 2010), automotive industry (Sansoni and Docchio, 2004), 3D building reconstruction and
city modeling (Tarsha et al., 2007), 3D shape blending (Li et al., 2009). Measuring technologies and mechatronics automation are recently following the needs for 3D representation and handling of mesh representations (Sun et al., 2011) and even holography is relying on such considerations (Ishikawa and Saito, 2008). All these promising applications make an intensive use of 3D point cloud handling, processing and rendering and it is a fact that, while the toolbox for processing images displayed over regular grids has reached a very mature level, much need to be done for visual data displayed over unorganized point sets (UPS). As a matter of fact, the lack of a natural parametrization of point cloud data introduces specific challenges by making it difficult to estimate analytic concepts like angles and areas over surfaces for instance (Lohani and Singh, 2008).

Recent works about biomodel prototyping have proved that bio-imaging is a promising emerging field of applications for rapid prototyping techniques. In (Huang et al., 2009), human ear 3D models are used as discriminant candidates for personal identification. In (Sareen et al., 2009), point cloud modeling helps to design facial prosthesis by implementing a contour-based 3D point cloud simplification for modeling free form surfaces. That application is based on the capture of the geometry of the patient’s existing facial features within few seconds using non-contact 3D scanners but must tackle with the resultant point data set which is very large and corrupted. In that perspective, the point set must be reduced to reconstruct accurate facial surfaces. In (Grunert et al., 2006), the ElePhant (Electronic Phantom) system implements an anatomically correct simulation system based on 3D rapid prototyping models for the otologic surgical intervention called mastoidectomy. In (Miller et al., 2009), advances in 3D rapid-prototyping printers, 3D modeling software, and casting techniques allowed for the fabrication of cost-effective, custom components in gamma-ray and x-ray imaging systems. In (Ma et al., 2010), preoperative planning for complex fracture cases enhances the outcomes of orthopaedic surgery while in (Lee et al., 2010) interactive digital design and manufacturing technologies make it possible to go for custom-made denture design, analysis, and production.

Beyond specific applications like in medicine or industry, we attempt in this chapter to explain why rapid prototyping has not reached yet a more extended market as expected in The Horizon Report 2004 NMC (2004) for education for instance.

To support all these recent applications, any new toolbox designed to efficiently - algorithmically and theoretically - process any UPS in order to filter out shapes within it will bring more end-users into the market. The CGAL library¹ for instance provides very interesting tools related to this perspective but always needs to be extended with sound new filtering techniques coming from the the image analysis toolbox for instance. This chapter takes the point of view of 3D printing to introduce new or recent ideas about rapid prototyping. In particular, Section 2 will be dedicated to a short overview of the point set analysis techniques with the theoretical frameworks inherited from mainstream image analysis filtering tools. In Section 3 we introduce our contribution to the field of mathematical morphology filtering tools for UPS and illustrate how it can benefit point set handling. Section 5 draws a few perspectives on the topic.

¹ http://www.cgal.org
2. Unorganized Point Set Filtering

When dealing with visual data represented by unorganized point sets, the implementation of mainstream filtering algorithms designed for radiometric image representations over a regular lattice is far from straightforward. Recently, many attempts to do so can be found in the literature. The first reason is related to the level of theoretical maturity reached by the image analysis toolbox while the second one comes from the technological advances that currently yield visual data at growing resolution and faster pace. Subsequently, working on huge, redundant radiometric images become quite impossible if interactive time constraint is required, not mentioning storage and networking issues. Hence, the computer vision community needs to adapt its paradigms to new kinds of visual data representations like interest points sets.

A thorough overview of this new paradigm consisting in processing point sets with an image analysis point of view can be found in (Lomenie & Stamon, 2011). The pioneers about the topic are H. Edelsbrunner’s team (Edelsbrunner & Shah, 1992) and Nina Amenta’s team (Amenta et al., 1998; Amenta & Bern, 1999) who proposed smart solutions aiming to algorithmically, and in a sense theoretically, define the shape of a point set for 3D surface reconstruction purposes. In parallel, M. Melkemi’s works also came along with interesting definition of 2D shapes underlying unorganized point sets (Melkemi & Djebali, 2001a; b). From a technological perspective, point modeling is gaining momentum not only in the field of computational geometry (Boubekeur et al., 2006) but also in the field of image analysis and related topics. For the sake of illustration, we can mention PointShop3D (Zwicker et al., 2002), a platform aimed at the processing of point sets comparable to usual platforms for 2D image processing such as The Gimp2.

Recent references are directly related to the field of rapid prototyping. In particular, feature extraction, registration and simplification are the most important visual processing tools recently worked upon in the rapid prototyping community. In (Mérigit et al., 2011; Novatnack and Nishino, 2007; Zhao et al., 2010; Zheng et al., 2009), normal estimation and corner extraction over unorganized point sets are algorithmically defined in order to perform UPS registration (Lin and He, 2011; Myronenko and Song, 2010; Rusu et al., 2008) or simplification (Sareen et al., 2009; Song et al., 2009; Xiao and Huang, 2010) for instance. A lot of works also deal with surface segmentation issues like in (Douillard et al., 2010; Huang & Mend, 2001; Jagannathan & Miller, 2003; Rabbani et al., 2006).

Beyond these adaptations of mainstream image analysis tools to UPS processing, we recently developed a new framework for mathematical morphology filtering dedicated to UPS. Contrary to the above statistic or analytic solutions, mathematical morphology is able to provide analysis tools at a more structural level and subsequently more resilient to shape noise or outliers. This toolbox can be used either as preprocessing filters to enhance the statistic and analytic solutions or as a per se analysis toolbox to discriminate between shapes and structures within the scene.

---

2 http://www.gimp.org
3. UPS Mathematical Morphology toolbox

As mentioned in (Peternell et al., 2003; Pottmann et al., 2004), a very few works have attempted to extend morphology to curved manifolds or to meshes and cell decompositions on curved manifolds so far (Heijmans et al., 1992; Roerdink, 1990, 1994; Rossel et al., 2000; Vincent, 1989). In the pioneering works of Luc Vincent et al., graph morphology was considered in a very abstract way in which the structuring element is a graph making the algorithmic handling of such filtering rather intractable from a time complexity point of view. In (Lomenie & Stamon, 2008), a more operational mathematical morphology toolbox for UPS has been devised in order to deal with visual point clouds and more specifically over the meshed triangulation providing the necessary neighborhood structures. We will call it MM4UPS. Practical applications of it are presented in (Lomenie & Stamon, 2011). Graph morphology is regaining attention as illustrated with recent studies by the pioneer in the field of mathematical morphology (Cousty et al., 2009; Levillain et al., 2010).

The MM4UPS unifies two concepts in the same framework: the concepts of α-objects first exposed to define “what is the shape” of a point set as described in (Edelsbrunner & Kirkpatrick, 1983; Edelsbrunner & Mucke, 1994) and the concept of mathematical lattice as described in (Heijmans & Ronse, 1990; Serra, 1988) to ground the mathematical foundations of mathematical morphology. In the following, we briefly describe both concepts with theoretical and algorithmic considerations. We refer the interested reader to (Lomenie & Stamon, 2008; 2011) for more details.

3.1 Notations

**Topological and geometrical structures.** Let $S$ be a point set in $\mathbb{R}^2$. (Edelsbrunner & Mucke, 1994) details how to compute the spectrum of α-shapes $S_\alpha(S)$ for any visual point sets in 2D or 3D for any $\alpha \in [0, \infty]$ with: $S_\infty = \text{conv}(S)$, where $\text{conv}$ stands for the convex hull, and $S_0 = S$ as limit cases (see Figure 1).

Let us define $k$-simplices $\sigma_T = \text{conv}(T)$, $T \subseteq S$ and $|T| = k + 1$ for $0 \leq k \leq 2$. Let us remind that the α-objects rely on the Delaunay triangulation $\text{Del}(S)$ of $S$ and the $\phi(T)$ values associated with any triangle $T \in \text{Del}(S)$, being the inverse radius of the circumscribe sphere to $T$ in the framework of α-objects. Then, for the algorithmic design, we just need to remember that for each simplex $\sigma_T$ in $\text{Del}(S)$, there is a single interval so that $\sigma_T$ is a face of the α-shape $S_\alpha$, i.e. if, and only if, $\alpha$ is contained in this interval.

**Lattice structures.** For any point set $S \in \mathbb{R}^2$, $\mathcal{M}(\text{Del})$ is the set of meshes on $\text{Del}(S)$, i.e., the set of mappings from the triangles $T$ in $\text{Del}$ to $\phi_T$ values. As for now, $T$ stands for any triangle in $\text{Del}$. A mesh $M \in \mathcal{M}(\text{Del})$ is defined by $\{(T, \phi)\}_{T \in \text{Del}}$ or equivalently by a mapping $\phi : T \in \text{Del} \rightarrow [0, \infty]$.

$\wp(\text{Del})$ is the set of all the corresponding sub-triangulations $D_i$ of $\text{Del}$. We can define a complete lattice structure within the functional theory frame for a point set including an order relation called $\mathcal{L} = (\mathcal{M}(\text{Del}), \leq)$, where the partial ordering $\leq$ is defined by: $\forall M_1$ and $M_2 \in \mathcal{M}(\text{Del})$, $M_1 \leq M_2 \iff \forall T \in \text{Del}, \phi_1^T \leq \phi_2^T$.

**Point set and mesh morphological operators.**
To define morphological operators, we need to affect to each triangle values $e_T$ and $d_T$ in addition to the measure $\phi_T$, defined by:

$$e_T = \min \{ \phi_{T'} | T' \in \nu(T) \}$$

$$d_T = \max \{ \phi_{T'} | T' \in \nu(T) \}$$

(1)

where $\nu(T)$ (in the framework of $\alpha$-objects) is the set of all triangles $T'$ of Del sharing at least one vertex with the triangle $T$, that is:

$$\nu(T) = \{ T' \in \text{Del} | T' \cap T \neq \emptyset \}$$

(2)

In our case, $\nu_T$ plays the role of a structuring entity (element or graph [Heijmans et al., 1992]). We proved in [Lomenie & Stamon, 2008] that with this definition of a structuring entity the following designed operators are actual mathematical dilation and erosion.

Then, in the lattice framework, we define two operators $e(M)$ and $d(M)$ on the complete lattice $\mathcal{L}_2$ by:

$$\forall M \in \mathcal{M}(\text{Del}), e(M) = \{ T \in \text{Del}, e_T \} \quad \text{and} \quad d(M) = \{ T \in \text{Del}, d_T \}$$

with $e_T$ and $d_T$ defined in Eq. (1).

The interesting point is that with this lattice structure, we inherit all the properties of classical morphology and particularly for the opening and closing filtering operators.

**MM4UPS toolbox.**
As for now, $\phi_T$ values are limited to the interval $[0, 1]$ and is related to a notion of visibility of the triangle or of membership to an object of interest within the mesh. Thus, we can define interactively or automatically sub-triangulations of interest as regions of interest in the meshed image.

Thus, we can define the whole set of operators in a way similar to regular grid formulation used with radiometric images. The structuring entity is the neighborhood $\nu(T)$ associated to each triangle whose definition can vary according to a specific relevant morphological operator. Having proved that the operators designed to obtain these new structures are theoretically sound as mathematical adjunctions, they can provide the whole set of mathematical morphological operators like opening acting on an unorganized point set $S$ or on a mesh $M$. We can define opening $o(M)$ and closing $c(M)$:

$$\forall M \in \mathcal{M}(\text{Del}(S)), \quad o(M) = d \circ e(M) \quad \text{and} \quad c(M) = e \circ d(M)$$

In functional radiometric mathematical morphology, an opening is idempotent but the size of the structuring element is flexible. To adapt the size of the structuring element in the case of mesh operators, we need to define opening of order $n$ as:

$$\forall M \in \mathcal{M}(\text{Del}(S)), \quad o^n(M) = d^n \circ e^n(M)$$

so that:

$$\forall n > 1, \quad o^n(M) \neq o(M)$$

but we still get the idempotent property of the mathematical morphology opening:

$$\forall M \in \mathcal{M}(\text{Del}(S)) \quad \text{and} \quad \forall n \in \mathbb{N}, (d \circ e(M))^n = d \circ e(M).$$

We can also benefit of all the inherited operators based on the erosion and the involution operators $c$:

$$\forall M \in \mathcal{M}(\text{Del}(S)), M^c = \{ T \in \text{Del}, 1 - \phi_T \} \quad \text{and} \quad e(M) = d(M^c)^c$$

![Fig. 2. (a) A point set $S$ in $\mathbb{R}^2$; (b) The $\alpha$-complex($S$) for $\alpha = \alpha_{opt}$ acting as a binarization operator; (c) The eroded structure $e(S)$; (d) The open structure $o(S)$.](image_url)
3.2 MM4UPS algorithmic issue

The complexity of all these operators is the same as that of the Delaunay triangulation: at worst in $O(N\log(N))$ (Boissonnat & Yvinec, 1995) for a set of $N$ points. Besides, contrary to most mesh visualization algorithms, determining the triangle and vertex adjacency relations efficiently on-the-fly is crucial. This is the critical issue when compared with mainstream image analysis algorithms that can intrinsically rely on a fixed neighborhood structure. We use the quadedge data structure introduced by Guibas and Stolfi (Guibas & Stolfi, 1985) (see 3). To create and modify the graph on-the-fly only two basic functions are used: “MakeEdge” and “Splice” to connect or disconnect edges. More specifically, as suggested by Shewchuk, 1996, we used a recasting of the quadedge data structure dedicated to triangular meshes: the tri-edge structure. However, and despite a good data structure to store neighborhood relationships between the various $k$-simplices within the simplicial complex, building a triangular mesh that preserves a rapid and valid access to adjacent edges or triangles using the low-level construction operators is neither simple nor intuitive. Delaunay triangulation algorithms can be classified as follows:

- higher dimensional embedding (Avis & Bremner, 1995);
- on-line incremental insertion (Edelsbrunner & Shah, 1992), holding the theoretical lower worst case time complexity and being simple to program;
- incremental construction (Dobkin & Laszlo, 1989);
- divide and conquer, for which managing adjacency is a rather hard problem during the merging phase. But, for instance, the DeWall algorithm (Cignoni et al., 1998) does not guarantee worst case optimality although it offers good performances in practical situations with optimization techniques.

Nevertheless, the DeWall algorithm fails in easily managing the adjacency tri-edge structures in some specific but common merging phases at the splice function level (see Lomenie & Stamot, 2008). From a software point of view, the CGAL library provides good support for the development of such tools in C++. However, we chose to implement our software in a more portable environment like Java3. An implementation within the ITK/VTK toolkit4 and library is currently under consideration.

---

3. To test the presented results, a Java applet and its source code are available in the public domain at http://sip-crip5.org/lomn/ in order to be used as the basis for many new projects.

4 http://www.itk.org
http://www.vtk.org
4. Results and Applications

We illustrate in the following the three types of immediate use of the MM4UPS toolbox for rapid prototyping purposes.

4.1 Preprocessing filtering

The three analytic and statistic modules consisting of registering, simplifying and extracting features from UPS can gain in robustness by preprocessing the UPS with the shape filters provided by the MM4UPS toolbox. We define the shape filtering algorithm for the registration module as the following:

**Algorithm 1** RegistrationFilter($S_1, S_2$)

**INPUT:** two 3D point clouds $S_1$ and $S_2$.

1. Compute $s_1 = \text{Proj}_{2D_e}(S_1)$;
2. Compute $s_2 = \text{Proj}_{2D_e}(S_2)$;
3. Compute $e_1(s_1)$ with $k_1$ such that $n_{cc}(e_1(s_1)) = 3$ or 2;
4. Compute $e_2(s_2)$ with $k_2$ such that $n_{cc}(e_2(s_2)) = n_{cc}(e_1(s_1))$;
5. Compute $\hat{S}_1 = \text{BackProj}_{3D}(e_1(s_1))$ and $\hat{S}_2 = \text{BackProj}_{3D}(e_2(s_2))$

**OUTPUT:** sub-sets $\hat{S}_1$ and $\hat{S}_2$

where $\text{Proj}_{2D_e}(S)$ is the orthogonal projection of the 3D point cloud $S$ onto the plane defined by the normal direction $e$ and $\text{BackProj}_{3D}(s)$ is the back-projection of the planar point set $s$ projected from the 3D point cloud $S$. In the following, the set $\{e_1, e_2, e_3\}$ will refer to the three eigen vectors corresponding to the three principal axes in the point cloud $S$, the corresponding eigen values $\lambda_1$, $\lambda_2$, $\lambda_3$ being ordered in ascending order. Thus, $e_1$ corresponds to the direction of the space with the least variance and is normal to what we call the principal plane of $S$.

The registration algorithm is more efficient in terms of robustness and time complexity when applied to the sub-sets $\hat{S}_1$ and $\hat{S}_2$ which are reduced, localized versions of the initial sets $S_1$ and $S_2$.

4.2 3D reconstruction tool

Laser or multi-view stereo 3D reconstruction devices usually provide noisy 3D point sets (Feng et al., 2008; Mederos et al., 2005). We define the shape filtering algorithm for the 3D reconstruction module as the following:

**Algorithm 2** ReconstructionFilter($S$)

**INPUT:** one 3D point cloud $S$.

1. Compute $s_i = \text{Proj}_{2D_e}(S)$ for $i \in \{1, 2, 3\}$;
2. Compute $\hat{s}_i = o(s_i)$ for $i \in \{1, 2, 3\}$;
3. Compute $\hat{S}_i = \text{BackProj}_{3D}(\hat{s}_i)$ for $i \in \{1, 2, 3\}$;
4. Compute $\cap \hat{S}_i$ for $i \in \{1, 2, 3\}$;

**OUTPUT:** the filtered subset $\hat{S}$

The algorithm is illustrated in Figure 4 for one projection $e_1$. For the 3D city modeling projects, we illustrate how by clustering according to the methodology described in (Lomenie, 2004) and then filtering out the shape noise from the various objects detected in the scene one is able to reconstruct a proper form of a tree as illustrated in Figure 5. Subsequently, a rapid
prototyping of an outdoor scene is within reach with an autonomous robot exploring an unknown environment for instance (see Figure 5).

Fig. 4. 3D reconstruction process of an obstacle (the chair) using the designed mesh analysis operators. From left to right: Original image - Stereo-reconstruction of the cluster associated to the obstacle - Projection of the 3D point set onto its eigen plane and morphological mesh opening - 3D retroprojection of the opened mesh

4.3 Analysis tool

Once a 3D point cloud model has been obtained and its shape has been filtered out, the MM4UPS toolbox can be very useful for automatically segmenting the different parts of the object as illustrated in Figure 7.

In the recent literature, resorting to contoured slices from the reconstructed point set is often used in order to filter out the ultimate 3D shape. Interestingly, the operators developed hereby work as well in mesh mode as in contour mode since acting either on the faces or the edges of the meshed triangulation as illustrated in Figure 8. We start form a 3D point cloud obtained by a stereoscopic system. The point cloud is quite noisy and non uniform in terms of density. The clustering algorithm method in (Lomenie, 2004) leads to a few clusters among them an obstacle in the scene to be prototyped for the rapid modeling of the unknown outdoor scene. The Figure 7(b) is the ultimate outcome of the analysis processing lane described in the rest of the figure. The vertical obstacle silhouette is outlined over the two meshed connected components extracted by successive erosions to get the seeds (the opening in Figure 8(c)(d)). Then geodesic dilations constrained to the optimal complex of binarization $C_{\alpha}$opt corresponding to the point set $S$ isolated in Figure 8.

5. Conclusion

While today’s systems of rapid prototyping are technologies heavily used by design engineers to make rapid tools to manufacture their products in aerospace or motor car industries mostly from CAD data models, it is not long before biologists, architects, artists but also individuals will be able to automatically manufacture objects of every description with no limit of complexity or input data. For example, by means of today’s 3D printers, anyone
10 Point Set Analysis: an image analysis point of view for Rapid Prototyping technologies

Fig. 5. (a), (b) and (c) Unorganized Point Sets clustered with the method in (Lomenie, 2004); (d) the 3D reconstruction of the isolated tree in the scene for rapid prototyping of a 3D scene model of a city should be able to copy a 3D object captured by a low-cost camera device for personal use (Salzmann and Fua, 2010). Of course this scenario is a long way off. Commercial packages have a lot of limitations (NMC, 2004) but was predicted back to 2004 by The New Horizon Report as a two-to three years adoption-time technology. They are still way in 2011 from making the capture of real-world objects into 3D a trivial task for the novice end user. We think that the presented mesh filtering operators, in association with the new trends to adapt theoretically sound algorithms from the image analysis toolbox to UPS, can help to model intricate organic shapes or regular polygonal objects placed in a cluttered environment as expected in (NMC, 2004).

6. References

Fig. 6. (a) The outdoor scene to be prototyped; (b) The clustering process to extract the various objects within the scene; (c) The outcome of rapid prototyping of the unknown scene in terms of obstacles after a dynamic exploration with a robotic stereoscopic system in autonomous exploration mode; (d) Samples of scenes modeled with the system.
Fig. 7. (a) Seed extraction by successive erosions; (b) Geodesic reconstruction of parts by constrained dilation of the seeds.


Fig. 8. (a) A 3D point cloud representing an outdoor scene with obstacles and its clustering into significant objects; (b) Geodesic reconstruction in contour mode of the silhouette of the vertical obstacle; (c) The whole obstacle clustered point set $s$ in mesh mode $C_\alpha(s)$ and (d) in contour mode; (e) $o(s)$ in (f) contour mode; (g) the constrained $d(o(s))$ to $C_\alpha(s)$ in (h) contour mode.
320-323.


