

Chapter 1

Digital Reproduction of Ancient Mosaics

SEBASTIANO BATTIATO

Dipartimento di Matematica ed Informatica
Università degli Studi di Catania
Viale A. Doria 6, 95125, Catania, Italy
Email: battiato@dmf.unict.it

GIANPIERO DI BLASI

Dipartimento di Ingegneria Informatica
Università degli Studi di Palermo
Viale delle Scienze, Ed. 6, 90128, Palermo, Italy
Email: dibiasi@dinfo.unipa.it

GIOVANNI GALLO

Dipartimento di Matematica ed Informatica
Università degli Studi di Catania
Viale A. Doria, 95125, Catania, Italy
Email: gallo@dmf.unict.it

GIOVANNI PUGLISI

Dipartimento di Matematica ed Informatica
Università degli Studi di Catania
Viale A. Doria, 95125, Catania, Italy
Email: puglisi@dmf.unict.it

1.1 Art and Computer Graphics

Art often provides valuable hints for technological innovations especially in the field of Image Processing and Computer Graphics. In this chapter we survey in a unified framework several methods to transform raster input images into good quality mosaics. For each of the major different approaches in literature the chapter reports a short description and a discussion of the most relevant issues. To complete the survey comparisons among the different techniques both in terms of visual quality and computational complexity are provided.

Non-Photorealistic Rendering (NPR) is a successful area of Computer Graphics and it is nowadays applied to many relevant contexts: scientific visualization, information visualization and artistic style emulation ([1],

[2]). NPR's goals may be considered complementary to the traditional main goal of the Computer Graphics which is to model and render 3D scenes in a natural (i.e., photorealistic) way. Within NPR the recent approach to digitally reproduce artistic media (such as watercolors, crayons, charcoal, etc.) and artistic styles (such as cubism, impressionism, pointillism, etc.) have been gaining momentum and are very promising ([3], [4] and [5]). Several denominations have been proposed for this narrower area within NPR: Artistic Rendering (AR) [2] and Computational Aesthetics (CA) [6] are among the most popular ones. None of these names have reached general acceptance within the research community. The authors of this chapter believe that, beyond names, a proper definition of the area that explains in a suitable way its purpose and its aim may be the following: *to reproduce the aesthetic essence of arts by mean of computational tools*.

The focus of this chapter is to review the state of the art for the problem of ancient digital mosaic creation. The survey restricts its scope only to the techniques that explicitly bear the "ancient mosaic" name (or equivalent) and that make use of primitives larger than pixels, points or lines. Stippling ([7], [8], [9] and [10]) and hatching ([11], [10]) are hence not covered here, although their visual similarity to mosaic naturally leads to approaches for these techniques that closely resemble the mosaic techniques. Mosaics, in essence, are images obtained cementing together small colored fragments (see Figure 1.1). Likely, they are the most ancient examples of discrete primitive based images. In the digital realm, mosaics are illustrations composed by a collection of small images called tiles. The tiles tessellate a source image with the purpose of reproducing the original visual information rendered into a new mosaic-like style. The same source image may be translated into many strikingly different mosaics. Factors like tile dataset, constraints on positioning, deformations and rotations of the tiles are indeed very influent upon the final results. As an example, the creation of a digital mosaic resembling the visual style of an ancient looking man-made mosaic is a challenging problem because it has to take into account the polygonal shape of the tiles, the small size of the tiles, the need to pack the tiles as densely as possible and, not last, the strong visual influence that tile orientation has on the overall perception of the mosaic. In particular orientation cannot be arbitrary but it is constrained to follow the gestalt choices made by the author of the source picture. Tiles, hence, must follow and emphasize the main orientations chosen by the artist.

The rest of this chapter is organized as follows: in Section 1.2 a brief history of ancient mosaics is summarized; Section 1.3 states the problem of digital mosaic creation into a mathematical framework; in Section 1.4 we present the crystallization mosaic techniques, while Section 1.5 explains the ancient mosaic methods. In Section 1.6 we extend the problem in a 3D environment. Finally Section 1.7 is devoted to final discussions and suggestions for future works.

1.2 History of Ancient Mosaics

The first examples of mosaic goes back some 4,000 years or more, with the use of terracotta cones pushed pointfirst into a background to give decoration [13] as demonstrated by unusual cone shaped-tesserae of various length found in Sumerian ancient manufacts. Other relevant examples, realized in almost the same period, can be found in some Egyptian and Phoenicia mosaics. After that we remind different pebble pavements, using different coloured stones to create patterns, although these tended to be unstructured decoration (about eighth century BC). In the four centuries BC the Greek artists re-discovered the pebble technique to an art form, with precise geometric patterns and detailed natural scenes. Next specially manufactured pieces (tesserae) were being used to give extra detail and range of colour to the work. Using small tesserae with different shapes and colors, the mosaics could imitate paintings. Many of the mosaics preserved at, for example, Pompeii were the work of Greek artists. Although not so famous it should be mentioned the important mosaic art in Mexico due to the ancient Maya civilisation.

The next evolution in the field was due to the expansion of the Roman Empire although the level of skill and artistry was diluted. Typically Roman subjects were scenes celebrating their gods, domestic themes and geometric designs. Different characteristics were exploited with the rise of the Byzantine Empire from the 5th century onwards, centred on Byzantium (now Istanbul, Turkey). In particular the main novelties were involved in style (i.e., eastern influences) and the use of special material (e.g., glass tesserae called *smalti*).



Figure 1.1: Examples of ancient mosaics [12].

Differently than before the Byzantines specialised in covering walls and ceilings. The *smalti* were ungrouted, allowing light to reflect and refract within the glass. Also, they were set at slight angles to the wall, so that they caught the light in different ways. The gold tesserae sparkle as the viewer moves around within the building. Roman images were absorbed into the typical Christian themes of the Byzantine mosaics, although some work is decorative and some incorporates portraits of Emperors and Empresses. Another important aspects of the mosaici history is based on the west of Europe where the Moors brought Islamic mosaic and tile art into the Iberian peninsula in the 8th century, while elsewhere in the Muslim world, stone, glass and ceramic were all used in mosaics. Islamic motifs are mainly geometric and mathematical [14].

In central Europe, mosaic went into general decline throughout the Middle Ages even if the tile industry led to mosaic tiling patterns in abbeys and other major religious buildings. During Gothic Revival there was some influence of the medieval themes. Some famous artist names like Antonio Salviati and Antoni Gaudi were able to give new emphasis to the mosaic world in the modern area.

1.3 The Digital Mosaic Problem

A first step toward the solution of the problem of digital mosaic creation is to state it within a mathematical framework. In particular the translation of a raster source image into a digital mosaic may take the form of a mathematical optimization problem as follows:

Given a rectangular region I^2 in the plane R^2 , a tile dataset and a set of constraints, find N sites $P_i(x_i, y_i)$ in I^2 and place N tiles, one at each P_i , such that all tiles are disjoint, the area they cover is maximized and the constraints are verified as much as possible.

The above definition is general and is suitable for many applications beyond Computer Graphics field. Indeed Harmon in 1973 ([15]) published the first results related with this kind of problems in the context of modeling human perception and automatic pattern recognition. Within this framework the problem can be viewed as a particular case of the cover problem or as a search and optimization problem. The mosaic construction as

formulated above can also be regarded as a low-energy configuration of particles problem. In the specific case of mosaics four different definitions can be given to solve specific problems:

Crystallization Mosaic - Given an image I^2 in the plane R^2 and a set of constraints (i.e., on edge features), find N sites $P_i(x_i, y_i)$ in I^2 and place N tiles, one at each P_i , such that all tiles are disjoint, the area they cover is maximized, each tile is colored by a color which reproduces the image portion covered by the tile. In this case in order to allow a solution the requirements have to be relaxed asking only that the constraints are verified as much as possible.

Ancient Mosaic - Given an image I^2 in the plane R^2 and a vector field $\phi(x, y)$ defined on that region by the influence of the edges of I^2 , find N sites $P_i(x_i, y_i)$ in I^2 and place N rectangles, one at each P_i , oriented with sides parallel to $\phi(x_i, y_i)$, such that all rectangles are disjoint, the area they cover is maximized and each tile is colored by a color which reproduces the image portion covered by the tile ([16]).

Photo-mosaic - Given an image I^2 in the plane R^2 , a dataset of small rectangular images and a regular rectangular grid of N cells, find N tile images in the dataset and place them in the grid such that each cell is covered by a tile that “resembles” the image portion covered by the tile.

Puzzle Image Mosaic - Given an image I^2 in the plane R^2 , a dataset of small irregular images and an irregular grid of N cells, find N tile images in the dataset and place them in the grid such that the tiles are disjoint and each cell is covered by a tile that “resembles” the image portion covered by the tile.

Different solutions to the problems above have been proposed. Most of these solutions are reviewed in this chapter within a unified framework, focusing mainly on the solutions proposed for the first two types of mosaics. More precisely we single out two different mosaic types:

1. crystallization mosaics (a.k.a. tessellation);
2. ancient mosaics.

These two types of mosaics decompose a source image into tiles (with different color, size and rotation), reconstructing the image by properly painting the tiles. They may hence be grouped together under the denomination of *tile mosaics*. Many mosaic techniques may fit in more than a single class and it is likely that other new types of mosaics will appear in the future. Other classifications of the known techniques for digital mosaics may be chosen if other criteria are taken into account. For example mosaics could be classified into:

1. fixed tile and variable tile (picture) size;
2. Voronoi based and non-Voronoi based approach;
3. deterministic and non-deterministic (probabilistic, random) algorithm;
4. iterative and one-step method;
5. interactive and batch system.

Finally, the relative performances with respect to the computational complexity has to be taken into account for an effective evaluation and classification. In the following most of all previously published algorithms are reviewed. The reader is provided with a general idea about the working of each technique. A discussion of the weakness and the strong points of each method is provided and, whenever possible, the relationships between algorithmic choices and visual qualities of the resulting images are emphasized.

1.4 The Crystallization Mosaics

Many sophisticated mosaic approaches adopt smart strategies using computational geometry (e.g., Voronoi diagrams) together with image processing. These techniques generally lead to mosaics that simulate the typical effect of some glass windows in the churches. A Voronoi diagram is a geometric structure that represents proximity information about a set of points or objects ([17]). Given a set of sites or objects, the plane is partitioned by assigning to each point its nearest site. The points whose nearest site is not unique, form the Voronoi diagram. That is, the points on the Voronoi diagram are equidistant to two or more sites. From a geometric point of view Voronoi cells can be considered convex polygons. Voronoi diagrams were first discussed by Lejeune-Dirichlet in 1850, but it was more than a half of a century later, in 1908, that these diagrams were written about in a paper by Voronoi, hence the name Voronoi diagrams. The Voronoi cells/polygons are sometimes also called Dirichlet Regions. There are a variety of algorithms available to construct Voronoi diagrams (see for example [17] and [18]), but the most famous algorithm was presented by Fortune [19]; he developed a plane-sweep algorithm which is more efficient in time than any other incremental algorithm. The algorithm guarantees an $O(n \ln n)$ complexity in the worst case.

Haeberli [20] used Voronoi diagrams, placing the sites at random and filling each region with a color sampled from the image. This approach tessellates the image with tiles of variable shapes and but it does not attempt to follow edge features; the result is a pattern of color having a cellular-like look (see Figure 1.2(b)). The effect may be efficiently implemented by z-buffering a group of colored cones onto a canvas. This allows to make the best use of hardware acceleration provided by modern graphics cards. Although very simple, Haeberli's idea is a milestone in this field and a starting point for many subsequent techniques.

In [21] Dobashi et al. extended the original idea of Haeberli. Their results are aesthetically more pleasant because the technique that they propose integrate edge information with Voronoi tessellation. The method, however, suffers from the variability that Haeberli's algorithm produces on tile shapes (Figure 1.2(c)). The strategy to better approximate the source image is simple. An error function E for the output image is defined as:

$$E = \sum_{x,y,c} \left(P_{(x,y,c)}^{Input} - P_{(x,y,c)}^{Output} \right)^2 \quad (1.1)$$

The function E is minimized by iteratively moving the centers of the Voronoi's polygons; the movement is limited to the 8-pixel neighborhood. The authors introduce also an heuristic strategy to speed up the moving process. Faustino and Figueiredo [22] presented a technique similar to Dobashi's. The main difference is that the sizes of tiles vary along the image: they are smaller near image details and larger otherwise (Figure 1.2(d)). To obtain this result, the authors use centroidal Voronoi diagrams with a density function that depends on image features (edge magnitude in this case). Differently from Dobashi, they do not start from an hexagonal lattice, but the seeds of the first Voronoi diagram are found by sampling the image by using a quadtree ([23]): the seeds are the centers of the leaf cells. A leaf is created when the color of its corresponding cell pixels are close to the average color of the cell. In particular, for each cell, they test if:

$$\max_{p \in C} d(I(p), c)^2 \leq \epsilon \quad (1.2)$$

where $I(p)$ is the color of the pixel p in C , c is the average color in C , and ϵ is an user-selected tolerance value. The image in Figure 1.2d has been obtained by using 2557 seed points, 10 iterations, $\epsilon=0.150$ and a minimum cell size equal to 36 pixels.

Figure 1.2 is provided to compare the above techniques in terms of visual appearance. Taking into account high-frequency details (edge features and their orientation) the general image structure is in some way preserved. Only Faustino and Figueiredo's approach makes use of tile size according to the different edge magnitude but without using the relative orientation.

A different approach is presented in [24] where Mould proposes a technique to reproduce medieval stained glass windows. He presents an unsupervised method for transforming an arbitrary image into a stained-glass version (see Figure 1.3). The key issues in designing a stained glass window are tile boundaries and colors.

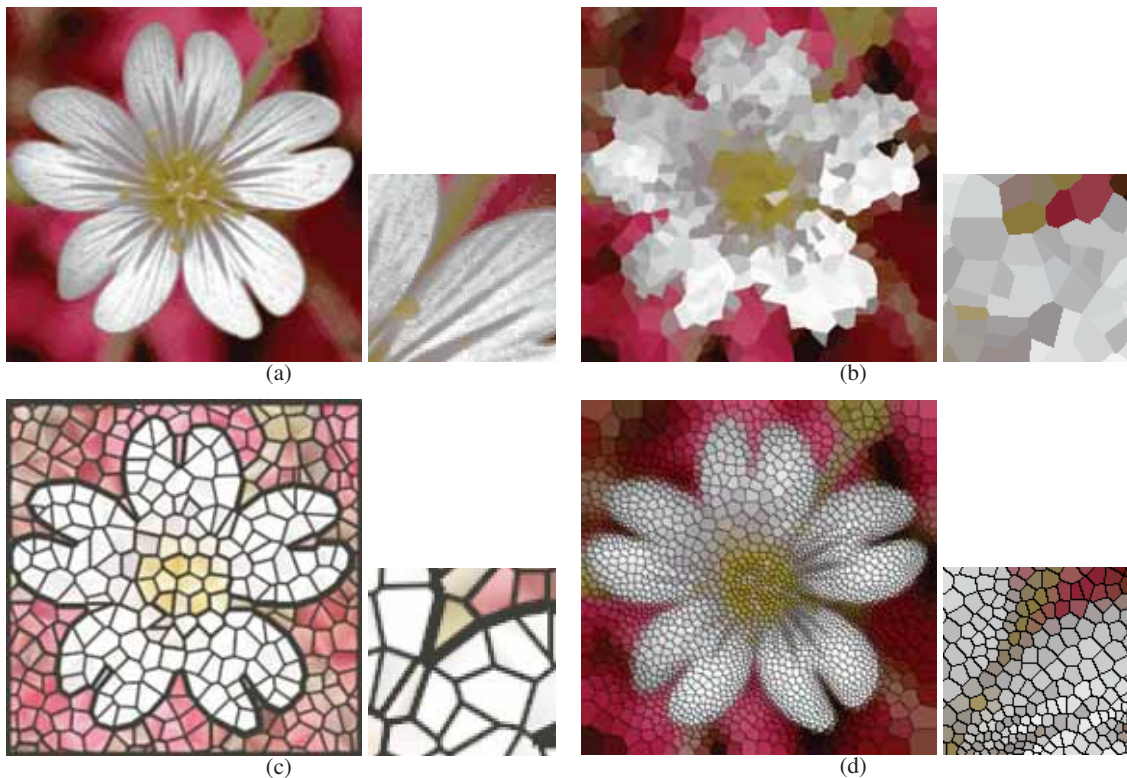


Figure 1.2: Crystallization mosaics: (a) the original image, (b) Haeberli [20], (c) Dobashi et al. [21], (d) Faustino and Figueiredo [22].

He uses erosion and dilation operators to manipulate and smooth an initial region segmentation into a tiling. The algorithm chooses tile colors from the palette of heraldic tinctures and renders a displacement-mapped plane to obtain the final image. The algorithm can be summarized as follows:

1. obtain an initial segmentation of the image by using an image processing algorithm (for example, he uses EDISON: [25], [26] and [27]);
2. evolve the segmentation to obtain an appropriate tiling having smooth boundaries and approximately convex pieces and lacking excessively large or excessively small pieces; the smoothing is obtained by the application of simple erosion and dilation operators from mathematical morphology;
3. choose a color for each tile; in particular it is possible to adopt the "heraldic" palette: for a given tile, determine its average color in the original image and the distance of this color from each heraldic color; the tile is then colored with the nearest heraldic color;
4. apply a displacement map to a plane, representing the leading and irregularities in the glass;
5. render the result.

The method is able to reproduce stained-glass images in a very effective way. Only a few parameters are involved; the edge magnitude and relative orientation are implicitly considered by combining segmentation and morphological operators.

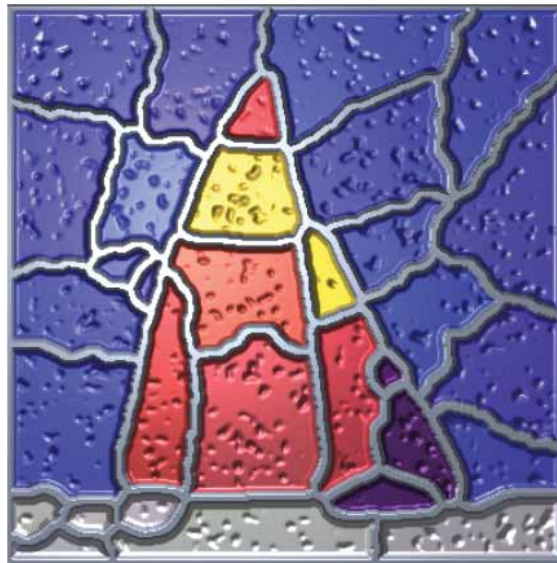


Figure 1.3: Medieval stained glass by Mould [24].

1.5 The Ancient Mosaics

Since ancient times the art of mosaic has been extensively used to decorate public and private places. Today it is still possible to see some of such artistic works realized first by Greeks and Romans and later during the Byzantine Empire. Different kind of mosaics were produced also by old pre-Columbian people. Finally also in the modern era several artists have continued to deal with artistic mosaics ([28], [13]).

Ancient mosaics are artworks constituted by cementing together small colored tiles. A smart and judicious use of orientation, shape and size may allow to convey much more information than the uniform or random distribution of N graphic primitives (like pixels, dots, etc.). For example, ancient mosaicists avoided lining up their tiles in rectangular grids, because such grids emphasize only horizontal and vertical lines. Such evidence may distract the observer from seeing the overall picture. To overcome such potential drawback, old masters placed tiles emphasizing the strong edges of the main subject to be represented. In our context we are not interested into physical design of a mosaic work (e.g., cementing materials, etc.), but in the way that the individual mosaic pieces - known to us as the *tesserae* - are laid down. By using different materials and/or combining the tesserae in various ways, many different artistic styles and effects can be obtained. The general “flow” of the mosaic is known as *andamento*. The typical ancient mosaics, today available in Computer Graphics, have a specific categorization in the field of Cultural Heritage. The Latin term “opus” is used to describe the overall look of the mosaic. In particular the implemented techniques until now are the *opus musivum* and the *opus vermiculatum* (see Figure 1.4).

As the epithet of “opus musivum” says this means of “quality worth of the muses”, of great visual refinery and effect. “Opus vermiculatum” takes its name from the Latin for “worm”. It refers to lines of tiles that snake around a feature in the mosaic. Often two or three rows of “opus vermiculatum” appear like a halo around something in a mosaic picture, helping it stand out from the background. The rendering of “opus vermiculatum” mosaics requires a clear separation between foreground and background because the two regions of the image have to be managed in different ways. The foreground region is covered as an “opus musivum”, while the background region have to be covered by a regular grid of tiles (possibly perturbed by a random noise in size, position and rotation).

The first attempt to reproduce a realistic ancient mosaic was presented by Hausner [16]. He proposed the mathematical formulation of the mosaic problem as described in Section 1.3. He obtained very good results using Centroidal Voronoi Diagrams (CVD), user selected edge features, L_1 (Manhattan) distance and graphic hardware acceleration (Figure 1.5(b)). In particular the method uses CVDs (which normally arrange points

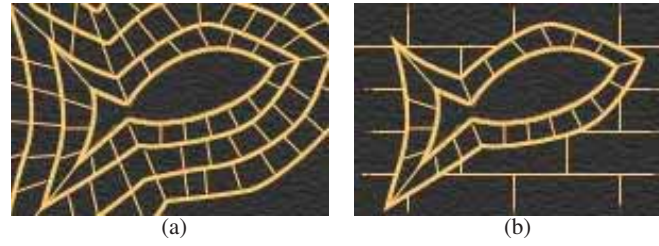


Figure 1.4: Examples of ancient mosaics styles ([28]): (a) opus musivum, (b) opus vermiculatum.

in regular hexagonal grids) adapted to place tiles in curving square grids. The adaption is performed by an iterative process which measures distances with the Manhattan metric whose main axis is adjusted locally to follow a chosen direction field (coming from the edge features). Computing the CVD is made possible by leveraging the z-buffer algorithm available in many graphics cards. Hausner's algorithm can be outlined as follows:

1. S = list of random points on the input image;
2. while not converged:
 - for each p in S , place a square pyramid with apex at p ;
 - rotate each pyramid about the z axis to align it to the field direction;
 - render the pyramid with an orthogonal projection onto the xy plane, producing a Voronoi diagram;
 - compute the centroid of each Voronoi region;
 - move each p to the centroid of its region.

The number of iterations to reach convergence is one of the main drawback of this technique, mainly when there is no direct access to the graphic acceleration engine.

Another algorithm for the creation of ancient mosaics is presented in [29] and in [30]; this approach is based on directional guidelines and distance transform and leads to very realistic results (Figures 1.5(c) and 1.7(b)). The algorithm uses some known image processing techniques in order to obtain a precise tile placing that can be summarized as follows:

1. segment the image by using the Statistical Region Merging algorithm ([31]);
2. subdivide the image into background and foreground regions (optional);
3. for each pixel of the image evaluate the distance transform from the segmented region bounds;
4. evaluate the gradient matrix and the level line matrix;
5. place the tile.

In particular point 5 can be described in more detail as follows:

1. while there are chains of pixels not yet processed in the level line matrix:
 - select a chain;
 - starting from an arbitrary pixel on it "follow" the chain;
 - place new tiles at regular distances along the path (the orientation of the tiles is assigned using the gradient information from matrix).

The authors, in order to obtain a high degree of similarity in terms of style with respect to ancient mosaics, also consider tile cutting (Figures 1.5(c) and 1.7(b)).

A novel technique for ancient mosaics generation has been presented in [10]. The authors present an approach for stroke-based rendering that exploits multi-agent systems; they call the agents *RenderBots*. RenderBots are individual agents representing in general one stroke. They form a multi-agent system and undergo a simulation to disseminate themselves in the environment. The environment consists of a source image and possibly additional G-buffer support images (edge image, luminance image, etc.). The final image is created when the simulation is stopped by having each RenderBot executed its painting function. The complete mosaic generation process can be described as follows:

1. setup the environment (a number of RenderBots of a specific class are created and distributed in the environment);
2. distribute the RenderBots (randomly or interactively by the user);
3. while the image is not finished:
 - simulate each bot (control of the bot physical behavior, computation of the new direction and velocity values and, possibly, change of the internal state of the RenderBot);
 - move each bot (perform the actual movement of the bot by computing a new position);
 - (eventually) paint each bot.

RenderBot classes differ in their physical behavior as well as their way of painting so that different styles can be created in a very flexible way. Thus they provide a unified approach for stroke based rendering. Different styles such as stippling, hatching, painterly rendering and mosaics can be created using the same framework. This is achieved by providing a specific class of RenderBots for each style. Figures 1.6(a) and 1.8(a) show two images obtained by RenderBot; before generating the mosaics, the images had been manually segmented. This is necessary because MosaicBots orient themselves using the nearest edges. The generation of NPR-images using RenderBots is an (iterative) interactive process, so production-times depend on the artist's requirements and the desired quality. It took about an hour to produce the results presented here. The number of MosaicBots involved were approximately 9000 and 8000 respectively for the images in Figures 1.6(a) and 1.8(a).

A very advanced approach to the rendering of traditional mosaics is presented in [32] (Figure 1.9(a)). This technique is based on offset curves that get trimmed-off the self intersecting segments with the guidance of Voronoi diagrams. The algorithm requires a mathematical description, as B-splines, of the edges and allows a very precise tile placement. Another point of this approach is the use of variable size tiles. Although the results are very good the technique seems limited to the case of a single, user-selected and closed edge curve.

A very interesting technique can be found in [33]; the authors present a new and efficient method to interactively create visually pleasing and impressive ancient mosaics. The algorithm is based again on the Lloyds method for CVT (Centroidal Voronoi Tessellation) computation and can be viewed as a smart extension and/or optimization of the technique proposed by Hausner [16]. They use a placement algorithm in an interactive fashion enabling the user to arrange tiles of various shapes and sizes. The user can easily control the distribution process by adding some other data such as contour lines and directional information. Tiles can be sized or shaped in order to better approximate the master image features. Additionally, this technique is less time expensive than using heuristic controlled automatic methods. An interactive tool is preferred because "the proper arrangement of individual tiles is a highly artistic process" [33]. These authors claim that heuristic methods produce unwanted artifacts such as mis-aligned tiles. In more details this algorithm can be summarized as follows:

1. M = set of randomly distributed and randomly oriented polygons P ;
2. while movement/rotation is above a threshold value:
 - for each P approximate its Voronoi region by using geometric modeled distance;
 - perform the Principal Component Analysis on polygons and their Voronoi regions;
 - compute the center of gravity CG of the polygons and of their Voronoi regions;
 - move the polygons in order to match the CG of the polygons and the CG of their Voronoi regions;
 - rotate polygons in order to align their principal component with the principal component of their Voronoi regions.

To create a mosaic a user must choose a master image and he has to define its feature lines and control polygons on the basis of the master image. He can hence choose the desired mosaic tile prototype (circles, quads or user-defined n -edges) and input the rough number of tiles to be inserted. After a preliminary unsupervised insertion the tiles can be manually inserted/deleted by making use of interactive tools. Figure 1.9(b) shows a typical output image of this technique. This algorithm clearly outperforms all previously presented techniques in terms of aesthetic result. Unfortunately it requires a (crucial) user intervention and it is strictly dependent on the user's aesthetic skill and experience: two different users could obtain two totally different aesthetic results starting from the same input data (image, tile prototype, etc.).

A novel technique for ancient mosaics generation has been presented in [34]. The authors, using graph cuts optimization algorithm, are able to overcome some typical weaknesses of the artificial mosaics approaches: user interaction, fixed tiles number, explicit edge detection phase (see Figure 1.6(b)). First it generates tile orientation field and then packs the tiles. These two steps can be summarized as follows:

1. Tile orientation field is generated considering the strong edges of the underlying image. Moreover orientations are forced to vary smoothly in order to produce pleasing mosaics and reduce the gap between tiles. This field is obtained by using a global optimization approach (α -expansion algorithm) [35].
2. The packing of the tiles is performed in two steps:
 - A set of mosaic layers $(M_1, M_2, \dots, M_n)$ is generated. Starting from a random pixel p each mosaic layer is created with a region growing strategy based on a greedy assumption (the nearby pixel s that does not overlap and with the minimum gap space with respect to p is chosen).
 - The mosaic layers are stitched together taking into account gap space minimization, the absence of broken tiles and the crossing of strong edge intensity. This task is performed through the graph cuts algorithm.

In [36, 37] the authors propose a novel approach based on Gradient Vector Flow (GVF) [38] computation together with some smart heuristics used to drive tiles positioning. Almost all previous approaches filter out high frequencies in order to simplify mosaic generation. On the contrary GVF properties permit to preserve edge information maintaining hence image details (see Figure 1.6(c) and 1.8(b)). Some heuristics are then used to follow principal edges and maximize the overall mosaic area covering. In particular the tiles positioning is not based only on gradient magnitude but makes use of local considerations to link together vectors that share the same logical edge. The algorithm can be summarized as follows:

1. compute GVF from image I ;
2. put in a queue Q the pixels whose GVF magnitude is greater than a threshold t_h ;
3. Sort Q with respect to GVF magnitude;
4. while Q is not empty:
 - extract a pixel (i, j) from Q and place a tile in according to the corresponding GVF phase skipping tile positioning if there is overlap with previously placed tiles;
 - starting from (i, j) follow the chain of local maxima (GVF magnitude greater then a threshold t_l with $t_l \leq t_h$) placing a tile according to the GVF phase (skipping tile positioning if there is overlap with previously placed tiles);
5. considering all the pixel, from up to bottom, from left to right, place in sequential order the tiles according to GVF phase (skipping tile positioning if there is overlap with previously placed tiles).

To summarize this Section the key of any technique aimed at the production of digital ancient mosaics is clearly the tile positioning and orientation. The methods presented in this Section use different approaches to solve this problem, obtaining different visual results. Some techniques are based on a CVD approach ([16], [32] and [33]) whereas other methods ([34], [36], [37]) compute a vector field by making use of different strategies (i.e., graph cuts minimization, gradient vector flow). Tile positioning is then performed with iterative strategies ([16], [32], [33], [34]) or reproducing the ancient artisans style by using a “one-after-one” tile positioning ([29], [30], [36] and [37]). A different non-deterministic approach is used in [10].

1.6 The Ancient Mosaics in a 3D environment

Recently [39] some attempts have been done to reproduce digital mosaics in a 3D environment extending in some sense the original ideas by a series of heuristics devoted to generalize the rendering process; in particular in the three dimensional representation the usage of vectorial primitives together with smart tricks able to simplify the original cuts described in [29] allow to reach very effective results. Just adding some simple random changes on tiles color, position and tilt angle, bring a strong realistic impact. Moreover, some features have been implemented, which attain a three dimensional mapping of the mosaicked image to different 3D surfaces (domes, pyramids, cylinders, etc.). The overall way to access to the mosaic changes giving to the users the possibility to highlight the requested level of details just navigating inside the 3D virtual space where the mosaic is rendered. Figure 1.10 shows some examples of such work obtained from an ancient mosaic realized by [36, 37] without tile cutting. Also we mention a method that simulate mosaic sculptures using tiles with irregular shapes, a method known by mosaicists as *opus palladium*, or simply crazy paving, due to the inherent freedom of mixing the tiles [40]. Special mosaic-like effects are obtained just controlling the tile distribution just considering proper texture mapping strategy.

1.7 Final Discussions

In this Section we briefly summarize the various characteristics of each presented method. Table 1.1 shows in a compact way, grouped by category, some specific details. The overall computational complexity is reported with respect to the number n of pixels of the input image and the number k of iterations for iterative methods. For iterative methods the parameter k is not always known a-priori: usually a suitable tuning phase is required. Note that for sake of clarity in some techniques we consider the grid size and/or granularity g proportional to the number of pixels n of the input image. To have a low computational complexity (almost linear in the

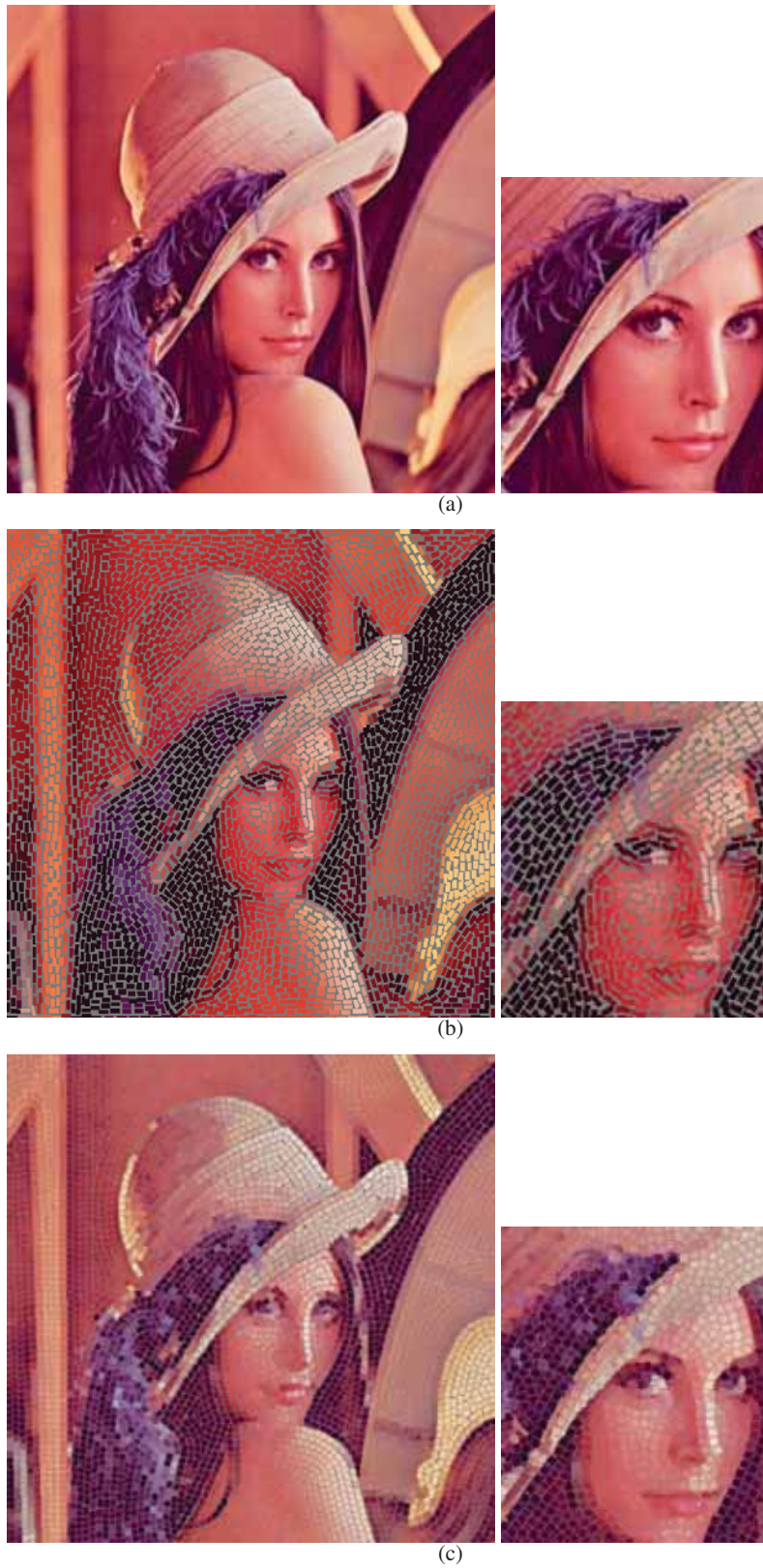


Figure 1.5: Ancient Mosaics: (a) the original image, (b) Hausner [16], (c) Di Blasi and Gallo [29].



Figure 1.6: Ancient Mosaics: (a) Schlechtweg et al. [10], (b) Liu et al. [34], (c) Battiato et al. [37].

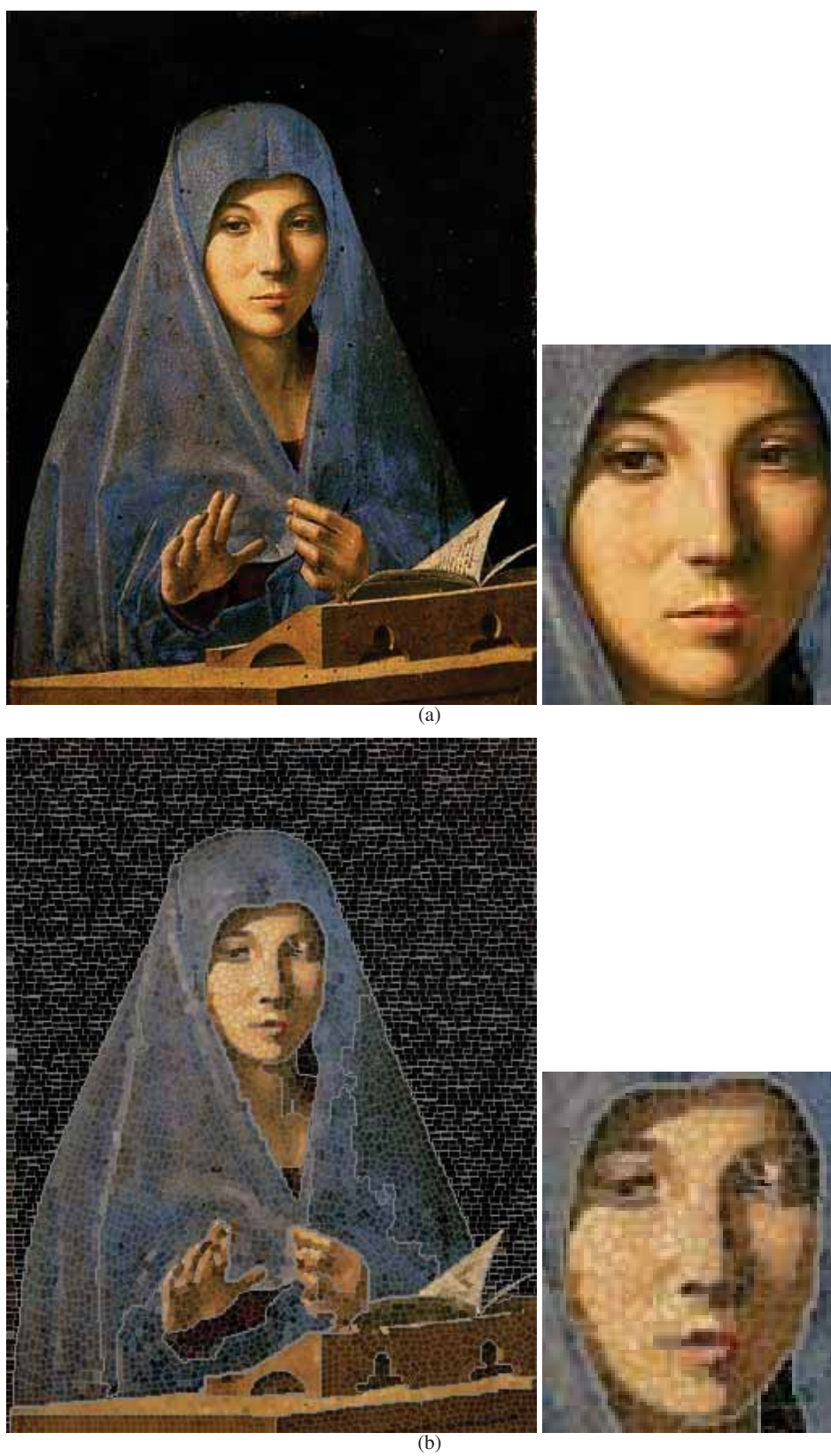


Figure 1.7: Other examples of ancient mosaics: (a) the original image, (b) Battiato et al. [30].

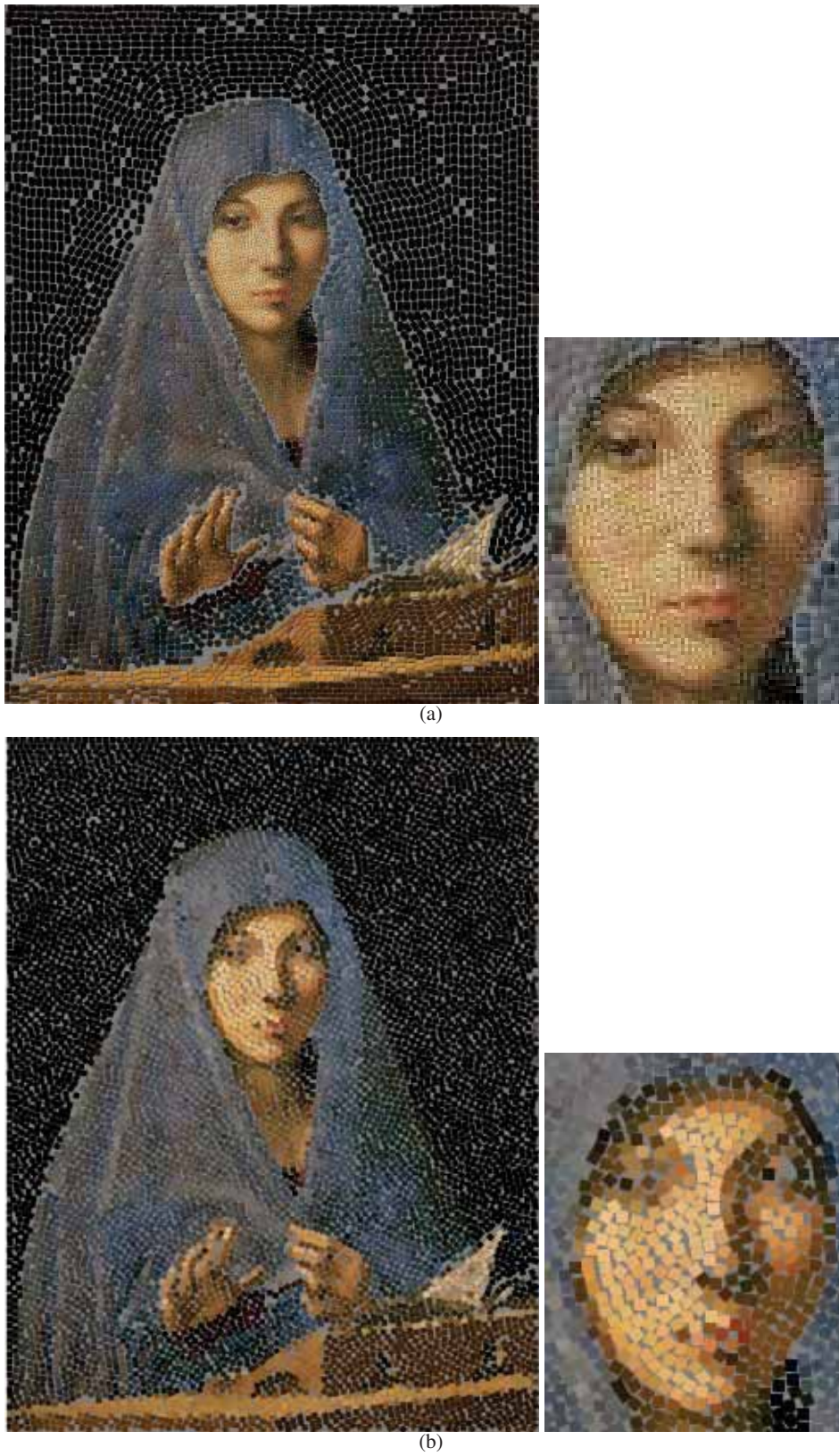


Figure 1.8: Other examples of ancient mosaics: (a) Schlechtweg et al. [10], (b) Battiato et al. [37].

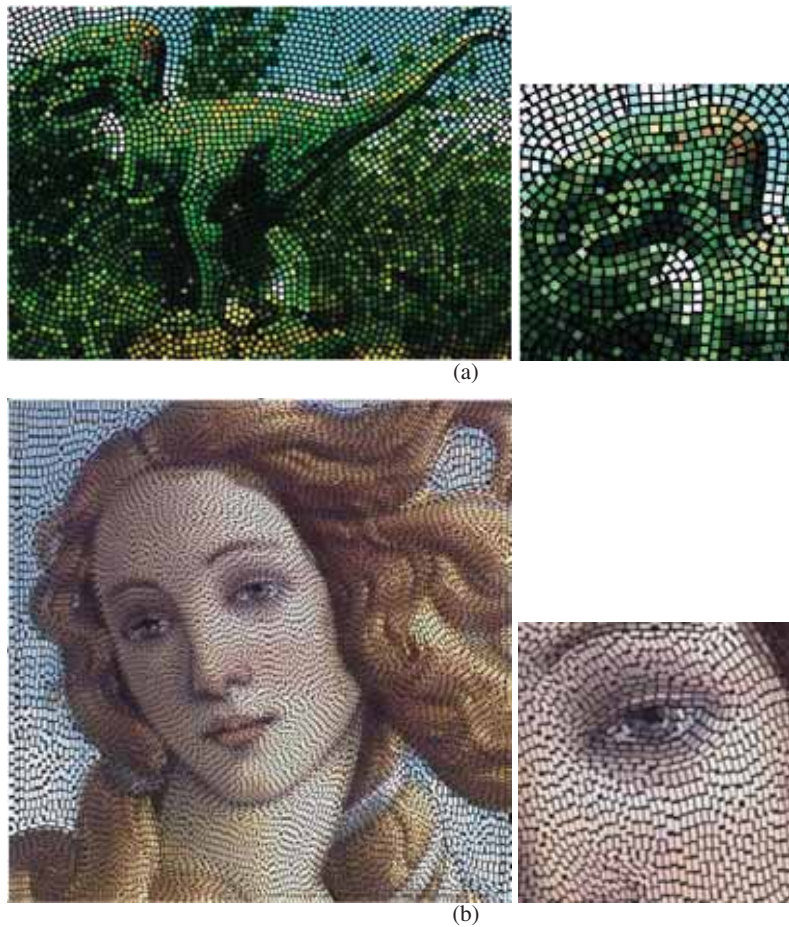


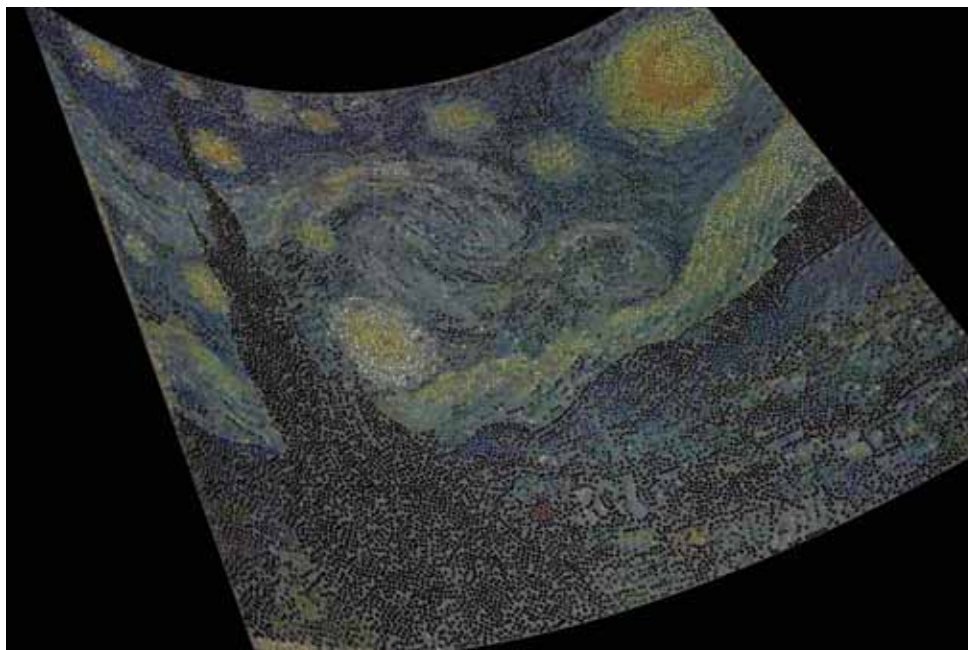
Figure 1.9: Other ancient mosaics: (a) Elber and Wolberg [32], (b) Fritzsche et al. [33].

number of involved pixel) is fundamental to be able to reproduce high-resolution mosaics without requiring expensive hardware resources. The remaining columns in Table 1.1 list other parameters and methodologies. One of the main effects of using fixed or variable tile size is the different emphasis given to specific details of the input image; the variability of the tile sizes produces a degradation in size along edges and textured areas especially if compared with classical mosaic appearance. Of course the particular partitioning strategy used by each method impacts the final result even if the tile positioning (orientation, cutting, etc.) is more crucial especially for unsupervised techniques (i.e., the key factor of each proposed strategy).

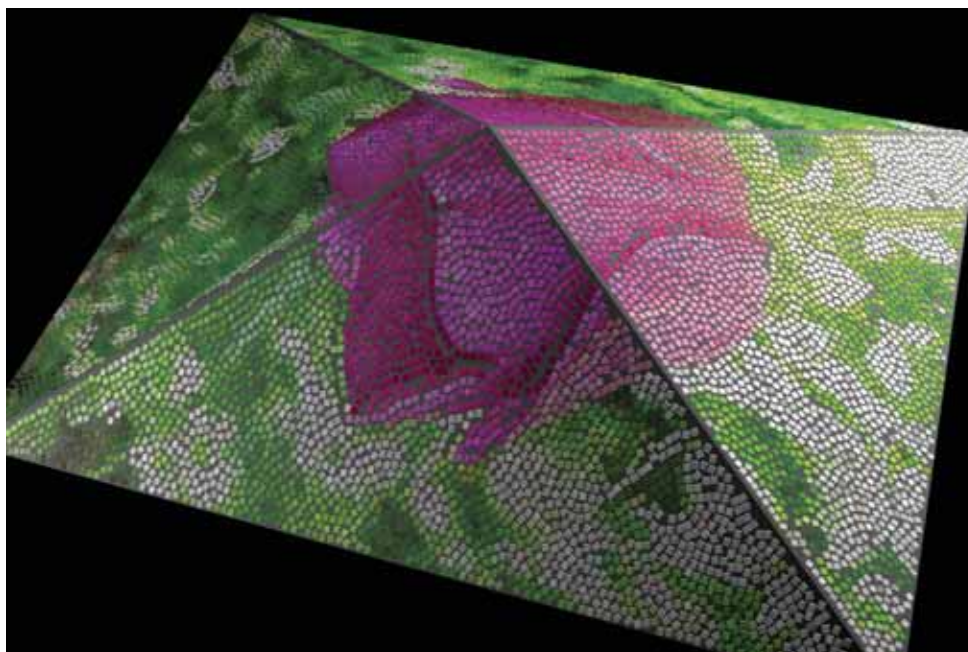
Although some criteria (e.g., covered area) have been used [36, 37] to compare different techniques, no evaluation is given in terms of aesthetic pleasure: no objective metrics are available to measure the effectiveness/accuracy of any digital mosaic. Aesthetic evaluation of any work produced by using supervised or unsupervised CA techniques is not so easy, because any objective metric is clearly inadequate ([2]). Only an artist could give a more reliable, but subjective judgment. The aesthetics of an output could be further evidenced by non-scientific and non-academic interests; for example the software proposed by Di Blasi and Gallo [29] has received interest from many artists, companies and Fine Arts Academies.

There are several ways to improve the aesthetic of the results and several ideas started from these works. In particular:

- a different strategy for choosing, in case overlapping tiles, which tile has to be cut; heuristic rules or, perhaps, randomized choices could produce different outcomes;
- generalization of the “mosaicists’ heuristic” to other kinds of primitive-based non photorealistic image processing seems possible and quite promising;



(a)



(b)

Figure 1.10: Some examples of 3D mosaics on cylinder (a) and pyramid (b).

- some generalizations proposed in [32], such as variable size tiles and photo-mosaic, are also considered for future work and research; it is also interesting to explore the possibilities offered using different basic shapes than rectangular tiles;
- automatic optimized choices of tile scale relative to each input image is an open problem worth further investigation (Figure 1.11);
- the extension of mosaic techniques to other kinds of mosaics as proposed in [32];

Table 1.1: List of the presented digital mosaic approaches and related features.

		Computational Complexity (n = number of pixels) (k = number of iterations)	Size	Partitioning	Deterministic	Iterative	Interactive
Crystallization Mosaics	Haeberli [20]	$O(n \ln n)$	Fixed	Voronoi	No	No	No
	Dobashi et al. [21]	$O(k \cdot n \ln n)$	Fixed	Voronoi	No	Yes	No
	Mould [24]	$O(n)$	Variable	Segmentation	Yes	No	No
	Faustino and Figueiredo [22]	$O(k \cdot n \ln n)$	Variable	Voronoi	No	Yes	No
Ancient Mosaics	Hausner [16]	$O(k \cdot \pi BufExecTime)$	Variable	CVD	No	Yes	No
	Elber and Wolberg [32]	$O(k \cdot n \ln n)$	Variable	CVD	No	Yes	No
	Battiato et al. [30]	$O(n)$	Fixed	None	Yes	No	No
	RenderBot [10]	$O(k \cdot botExecTime)$	Fixed	None	No	Yes	Yes
	Fritzsche et al. [33]	$O(k \cdot n \ln n)$	Variable	CVD	Yes	Yes	Yes
	Liu et al. [35]	$O(k \cdot n)$	Fixed	None	No	Yes	No
	Battiato et al. [36, 37]	$O(k \cdot n)$	Fixed	None	Yes	Yes	No

- a different method to better find the directional guidelines is an important research investigation issue;
- exploitation of hardware graphics primitives to accelerate the mosaic synthesis;
- the usage of methods able to infer from high quality picture of real ancient mosaics the underlying structure or original schema [41];
- inferring the tiles distribution and novel aesthetic procedures to arrange the overall mosaic just considering the work of some artists that have tried to "copy" some painting in a mosaic (see Figure 1.12).

1.7.1 Final Summary

In this chapter, we surveyed several approaches to non-photorealistic rendering of digital images in the field of mosaic generation. The various methods have been grouped together according to the main features. In particular, we singled out two different kind of mosaics: crystallization and ancient mosaics. It is also possible to group the mosaic creation methods by using different criteria. The common and different ideas among the methods had been reported and described. The various techniques had been compared also with respect to the overall performances both in terms of achieved visual effects and computational complexity.



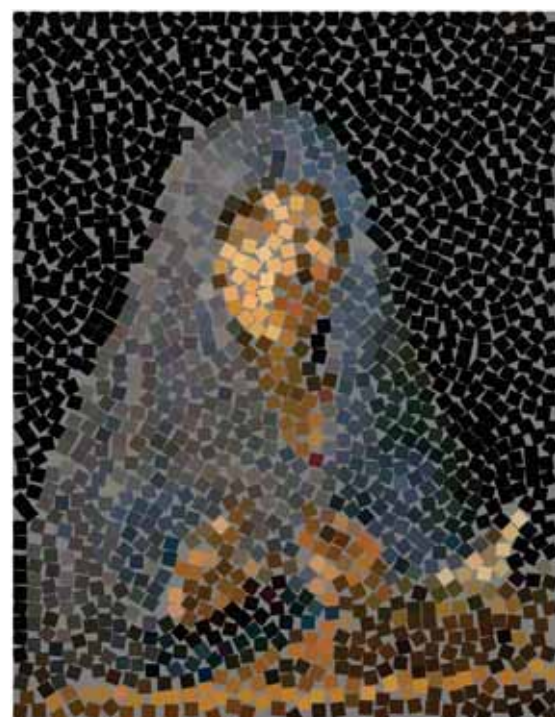
(a)



(b)



(c)



(d)

Figure 1.11: Mosaics generated with increasing tiles size (a) 3x3, (b) 6x6, (c) 10x10, (d) 14x14.



(a)



(b)

Figure 1.12: Example of a mosaic realized by an artist starting from a painting (a) and a detail (b).

Bibliography

- [1] T. Saito and T. Takahashi, “Comprehensible rendering of 3-d shapes,” in *Proceedings of the 17th annual conference on Computer graphics and interactive techniques (SIGGRAPH'90)*, (Dallas, TX, USA), pp. 197–206, ACM Press, 1990.
- [2] J. Collomosse, *Higher Level Techniques for the Artistic Rendering of Images and Video*. PhD thesis, University of Bath, UK, May 2004.
- [3] J. Collomosse and P. Hall, “Cubist style rendering from photographs,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 9, pp. 443–453, October 2003.
- [4] C. Curtis, S. Anderson, J. Seims, K. Fleischer, and D. Salesin, “Computer-generated watercolor,” in *Proceedings of the 24th annual conference on Computer graphics and interactive techniques (SIGGRAPH'97)*, (New York, NY, USA), pp. 421–430, ACM Press/Addison-Wesley Publishing Co., 1997.
- [5] A. Hertzmann, C. Jacobs, N. Oliver, B. Curless, and D. Salesin, “Image analogies,” in *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, (New York, NY, USA), pp. 327–340, ACM, 2001.
- [6] L. Neumann, M. Sbert, B. Gooch, and W. Purgathofer, eds., *Computational Aesthetics 2005: Eurographics Workshop on Computational Aesthetics in Graphics, Visualization and Imaging 2005, Girona, Spain, May 18-20, 2005*, Eurographics Association, 2005.
- [7] O. Deussen, S. Hiller, C. Overveld, and T. Strothotte, “Floating points: A method for computing stipple drawings,” *Computer Graphics Forum (EG'00)*, vol. 19, no. 3, pp. 40–51, 2000.
- [8] A. Secord, “Weighted voronoi stippling,” in *2nd International Symposium on Non-Photorealistic Animation and Rendering (NPAR'02)*, (Annecy, France), pp. 37–43, June 3-5 2002.
- [9] S. Hiller, H. Hellwig, and O. Deussen, “Beyond stippling - methods for distributing objects on the plane,” *Computer Graphics Forum*, vol. 22, September 2003.
- [10] S. Schlechtweg, T. Germer, and T. Strothotte, “Renderbots-multi-agent systems for direct image generation,” *Computer Graphics Forum*, vol. 24, pp. 137–148, June 2005.
- [11] A. Secord, W. Heidrich, and L. Streit, “Fast primitive distribution for illustration,” in *Thirteenth Eurographics Workshop on Rendering (2002)*, 2002.
- [12] N. Tellis, “Ancient mosaics; <http://www.mosaic-tile-art.com/mosaic.html>.”
- [13] I. Fiorentini Roncuzzi and E. Fiorentini, *Mosaic: materials, techniques and history*. MWeV, 2002.
- [14] M. Du Sautoy, *Symmetry: A Journey Into the Patterns of Nature*. HarperCollins, 2008.
- [15] L. Harmon, “The recognition of faces.,” *Scientific American*, vol. 229, no. 5, pp. 71–82, 1973.
- [16] A. Hausner, “Simulating decorative mosaics,” in *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, (New York, NY, USA), pp. 573–580, ACM, 2001.
- [17] F. Preparata and M. Shamos, *Computational Geometry: An Introduction*. New York: Springer-Verlag, 1985.
- [18] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf, *Computational Geometry*. Berlin: Springer-Verlag, 1997.
- [19] S. Fortune, “A sweep-line algorithm for voronoi diagrams,” *Algorithmica*, vol. 2, pp. 153–174, 1987.

- [20] P. Haeberli, "Paint by numbers: Abstract image representation," in *Proceedings of the 17th annual conference on Computer graphics and interactive techniques (SIGGRAPH'90)*, (Dallas, TX, USA), pp. 207–214, ACM Press, 1990.
- [21] J. Dobashi, T. Haga, H. Johan, and T. Nishita, "A method for creating mosaic images using voronoi diagrams," in *Computer Graphics Forum (EG'02)*, pp. 341–348, 2002.
- [22] G. Faustino and L. de Figueiredo, "Simple adaptive mosaic effects," in *Proceedings of the 18th Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI 2005)*, (Natal, RN, Brazil), pp. 315–322, IEEE Computer Society, 2005.
- [23] R. Finkel and J. Bentley, "Quad trees: A data structure for retrieval on composite keys," *Acta Informatica*, vol. 4, pp. 1–9, March 1974.
- [24] D. Mould, "A stained glass image filter," in *Fourteenth Eurographics Workshop on Rendering (2003)*, pp. 20–25, 2003.
- [25] C. Christoudias, B. Georgescu, and P. Meer, "Synergism in low level vision," in *ICPR '02: Proceedings of the 16th International Conference on Pattern Recognition (ICPR'02)*, vol. 4, (Washington, DC, USA), pp. 150–155, IEEE Computer Society, 2002.
- [26] D. Comaniciu and P. Meer, "Mean shift: a robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.
- [27] P. Meer and B. Georgescu, "Edge detection with embedded confidence," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 12, pp. 1351–1365, 2001.
- [28] "Mosaic art in vitreous glass, millefiori, tesserae mosaics by shelby glass studio; <http://www.mosaic-tile-art.com/mosaic.html>," 2006.
- [29] G. Di Blasi and G. Gallo, "Artificial mosaics," *The Visual Computer*, vol. 21, pp. 373–383, July 2005.
- [30] S. Battiato, G. Di Blasi, G. Farinella, and G. Gallo, "A novel technique for opus vermiculatum mosaic rendering," in *14th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision (WSCG'06)*, pp. 133–140, 2006.
- [31] R. Nock and F. Nielsen, "Statistical region merging," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 11, pp. 1452–1458, 2004.
- [32] E. Elber and G. Wolberg, "Rendering traditional mosaics," *The Visual Computer*, vol. 19, no. 1, pp. 67–78, 2003.
- [33] L. Fritzsche, H. Hellwig, S. Hiller, and O. Deussen, "Interactive design of authentic looking mosaics using voronoi structures," in *In Proc. 2nd International Symposium on Voronoi Diagrams in Science and Engineering VD 2005 Conference*, pp. 1–11, 2005.
- [34] Y. Liu, O. Veksler, and O. Juan, "Simulating classic mosaics with graph cuts," in *Proceedings of Energy Minimization Methods in Computer Vision and Pattern Recognition*, pp. 55–70, 2007.
- [35] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, pp. 1222–1239, Nov 2001.
- [36] S. Battiato, G. Di Blasi, G. Gallo, G. C. Guarnera, and G. Puglisi, "Artificial mosaics by gradient vector flow," in *Short Proceedings of EUROGRAPHICS*, 2008.
- [37] S. Battiato, G. Di Blasi, G. Gallo, G. C. Guarnera, and G. Puglisi, "A novel artificial mosaic generation technique driven by local gradient analysis," in *Proceedings of Computational Science ICCS*, pp. 76–85, 2008.
- [38] C. Xu and L. Prince, "Snakes, shapes, and gradient vector flow," *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 359–369, 1998.
- [39] S. Battiato, G. Di Blasi, G. Gallo, and A. Milone, "Realizzazione di mosaici artificiali in ambiente 3D," in *Proceedings of V Convegno AIAR Associazione Italiana di Archeometria*, Siracusa, February 2008.
- [40] V. A. Dos Passos and M. Walter, "3D virtual mosaics: Opus palladium and mixed styles," *The Visual Computer*, vol. 25, no. 10, pp. 939–946, 2009.
- [41] L. Benyoussef and S. Derrode, "Tessella-oriented segmentation and guidelines estimation of ancient mosaic images," *Journal of Electronic Imaging*, vol. 17, October 2008.