

A Novel Artificial Mosaic Generation Technique Driven by Local Gradient Analysis

Sebastiano Battiato, Gianpiero Di Blasi, Giovanni Gallo,
Giuseppe Claudio Guarnera, and Giovanni Puglisi

Dipartimento di Matematica e Informatica
University of Catania, Italy

{battiato,gdiblas,gallo,guarnera,puglisi}@dmi.unict.it
<http://www.dmi.unict.it/~iplab>

Abstract. Art often provides valuable hints for technological innovations especially in the field of Image Processing and Computer Graphics. In this paper we present a novel method to generate an artificial mosaic starting from a raster input image. This approach, based on Gradient Vector Flow computation and some smart heuristics, permit us to follow the most important edges maintaining at the same time high frequency details. Several examples and comparisons with other recent mosaic generation approaches show the effectiveness of our technique.

Keywords: Artificial Mosaic, Non Photo-Realistic Rendering, Gradient Vector Flow.

1 Introduction

Mosaics are artwork constituted cementing together small colored tiles. Probably they can be considered the first example of image synthesis technique based on discrete primitives. The creation of a digital mosaic of artistic quality is a challenging task. Many factors like position, orientation, size and shape of tiles must be taken into account in the mosaic generation in order to densely pack the tiles and emphasize the orientation chosen by the artist. Digital mosaic generation from a raster image can be formulated as an optimization problem in the following way:

Given an image I in the plane R^2 and a vector field $\Phi(x, y)$ defined on that region by the influence of the edges of I , find N sites $P_i(x_i, y_i)$ in I 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 [1].

Many mosaic generation algorithms have been recently developed. Hausner [1] uses Centroidal Voronoi Diagram together with user selected features and Manhattan distance. This approach obtains good results but, due to the high number of iterations necessary to reach convergence, it is computationally slow. In [2,3] the authors present an approach based on directional guidelines and distance transform. They obtain very realistic results with a linear complexity

with respect to image size. A novel technique for ancient mosaics generation has been presented in [4]. The authors, using graph-cut optimization algorithm, are able to work on tiles positioning without an explicit edge detection phase. Other related works can be found in [5,6,7]. A complete survey of the existing methods in the field of artificial mosaic is available in [8].

In this paper we propose a novel approach based on Gradient Vector Flow (GVF) [9] 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. Preliminary works [10] have shown that GVF properties permit us to preserve edge information and maintain image details. The novelty of this paper is related to the heuristics used to follow principal edges and to maximize the overall mosaic area covering. In particular the tiles positioning is not based only on gradient magnitude [10] but makes use of local considerations to link together vectors that share the same "logical" edge. Experimental results confirm the better quality of the new technique with respect to the state of the art proposals [8].

The paper is structured as follows. Next section describes in detail the proposed methodology. Experiments are reported in Section 3 whereas Section 4 closes the paper tracking direction for future works.

2 Proposed Algorithm

In order to emulate the style of an ancient artisan we have developed an automatic technique based on the following two steps:

- GVF (Gradient Vector Flow) field computation based on [9] algorithm;
- rule based tile positioning.

GVF is a dense force field designed by the authors of [9] in order to solve the classical problems affecting snakes (sensitivity to initialization and poor convergence to boundary concavity). Starting from the gradient of an image, this field is computed through diffusion equations.

GVF is a field of vectors $\mathbf{v} = [v, u]$ that minimizes the following energy function:

$$E = \int \int \mu(u_x^2 + u_y^2 + v_x^2 + v_y^2) + |\nabla f|^2 |\mathbf{v} - \nabla f|^2. \quad (1)$$

where the subscripts represent partial derivatives along x and y axes respectively, μ is a regularization parameter and $|\nabla f|$ is the gradient computed from the intensity input image. Due to the formulation described above, GVF field values are close to $|\nabla f|$ values where this quantity is large (energy E , to be minimized, is dominated by $|\nabla f|^2 |\mathbf{v} - \nabla f|^2$) and are slow-varying in homogeneous regions (the energy E is dominated by sum of the squares of the partial derivatives of GVF field). An example of GVF field is shown in Figure 1.

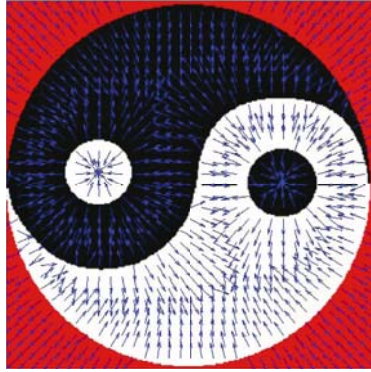


Fig. 1. Input image and its corresponding GVF field

This vector field can be used to effectively drive tiles positioning. Edge information is preserved, it is propagated in the close regions and merged together in a smoothly way.

Let I be the input image to be mosaiced. In order to simplify the algorithm we work only on the luminance channel of the image I , eliminating hue and saturation information. The luminance $L(I)$ is then equalized and the discrete gradient of the results is calculated, by means of crossing difference. The equalization process, especially for natural images, allows to normalize the overall gradient distribution. The horizontal and vertical components of the gradient $\nabla L(I)$ are used as input for the GVF algorithm. Notice that in the implementation gradient computation is performed using Robert's Kernel. This choice is more noise sensitive and hence incorporates in the final mosaic a little, aesthetically pleasant, randomness.

All the tiles have the same rectangular shape and the same size. Moreover we impose that tiles do not overlap. Placing ordering is hence fundamental in terms of visual overall effect. First we consider local $|GVF(I)|$ maxima with values greater than a threshold t_h . These pixels, sorted according to $|GVF(I)|$, are selected together with their neighbors with $|GVF(I)|$ greater than t_l (chains of tiles are built up and placed). We impose that the tile orientation is obtained according to the GVF direction in its central point. In this way we locate first the neighborhood of main edges in the input image, just to follow the perceptual orientation of the image itself. The second step of the algorithm is devoted to cover the homogeneous regions of the image. This is accomplished simply placing each tile one by one following the order left to right, up to bottom, starting from the upper-left corner of the image. This heuristic strategy, somewhat arbitrary, is justified by the properties of the GVF: this technique leads to aesthetically pleasant results, by preserving main orientations and covering a wide portion of pixels with tiles densely packed. The algorithm can be summarized as follows:

1. Input: a raster image I
2. $L(I) \leftarrow \text{Luminance}(I)$
3. $G(I) \leftarrow \text{Robert's Gradient}(\text{Equalize}(L(I)))$
4. $[u, v] \leftarrow \text{GVF}(|G(I)|_{\infty}, \mu, n\text{Iterations})$
5. $gvf(I) \leftarrow (u^2 + v^2)^{1/2}$
6. $I_n(I) \leftarrow \text{NonMaxSuppression}(gvf(I))$
7. let t_h, t_l be threshold values, with $t_h > t_l$
8. Sort in queue Q pixels (i, j) according to decreasing $I_n(i, j)$ values.
Only pixels whose I_n is greater than the threshold t_h are put into Q .
9. while Q is not empty
10. Extract a pixel (i, j) from Q
11. if (i, j) is not marked
12. Place a tile centering it in (i, j) at angle $\alpha = \tan^{-1}(v(i, j)/u(i, j))$
13. if in this way the tile overlaps with previously placed tiles
14. Skip tile positioning
15. Starting from (i, j) follow and mark as visited the chain of local maxima (i.e. $I_n(w, z) > t_l$) in both directions perpendicular to α , to obtain a guideline
16. Place a tile centering it in each (w, z) in the chain at angle $\beta = \tan^{-1}(v(w, z)/u(w, z))$
17. if in this way the tile overlaps with previously placed tiles
18. Skip tile positioning.
19. for $j \leftarrow 1$ to $\text{length}(I)$
20. for $i \leftarrow 1$ to $\text{width}(I)$
21. Place a tile in the pixel (i, j) at angle $\gamma = \tan^{-1}(v(i, j)/u(i, j))$
22. if in this way the tile overlaps with previously placed tiles
23. Skip tile positioning.

3 Experimental Results

The above algorithm has been implemented in JAVA, using for GVF computation an external MATLAB module. The method as been compared, to assess the aesthetic quality, with the mosaics obtained by [2] and [4] by using their original implementations. We wish to thank all the authors of these papers for providing the output of their techniques upon our request. For sake of comparison Figure 2 reports the mosaic obtained from the standard Lena picture. A first clear advantage of the novel technique is that it is able to better preserve fine details. This happens because high frequency areas are priority on tiles placing. Observe, to support this claim, the areas around Lena's nose, lips and high brow. Another performance index of mosaic algorithms is the amount and spatial distribution of untiled space. The area left uncovered by the proposed technique is comparable with the amount of uncovered area left by [2], but gaps are here better distributed. For example the constraints of [2] force the appearance of a long "crack" in the vertical band on the wall behind Lena, while the proposed approach achieves in the same region a pleasant smoothness. Relatively to [4] it should be observed that the uncovered area left by our technique is considerably less. Observe that a higher percentage of covered area leads to a better preservation of the original



Fig. 2. Visual comparison between mosaics generated by our approach (B), [2] (C) and [4] (D), applied on input image (A) considering in all cases the same tile size 5×5

colors of the same picture (see Table 1). The perceived texture obtained with the proposed technique appears, finally, less chaotic than the texture obtained with [4]. As for the parameter adopted to produce Figure 2, the tile size is 5×5 and the image size is 667×667 . Typically, considering other images, we obtain covered area greater than [4] but smaller than [2].

Table 1. Number of tiles and covered area comparison between various approaches

Technique	Number of tiles	Covered area
Our Method	13412	75.4%
[2]	13994	78.6%
[4]	11115	62.5%

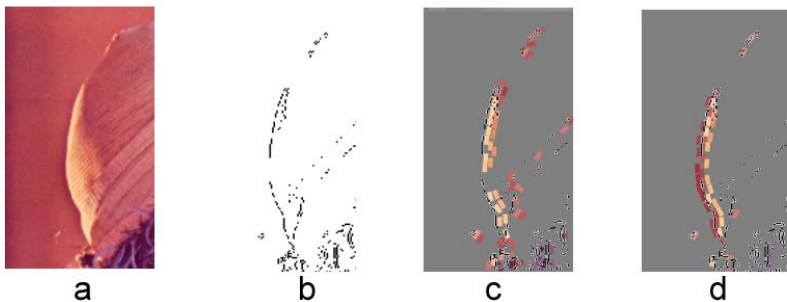


Fig. 3. Comparison between the proposed approach and [10] on a Lena image detail (a). The novel heuristics (c) are able to follow the underlying edges (b) maintaining higher fidelity than (d) also considering the original colors.

Figure 3 shows a comparison between the proposed approach and [10] on a Lena image detail (a). Both approaches do not consider the final step of linear tiles placing. The novel heuristics (c) are able to follow the underlying edges (b) maintaining higher fidelity than (d) also considering the original colors.

In Figure 4 we show the same image processed using increasing tile size. The right relation between image size (and its level of detail) of the input, and the tile size to be used in the mosaicing process, can be derived only by aesthetic considerations. Our algorithm is able to preserve the global appearance even with higher tile size. Finally we show in Figure 5 an example of mosaiced image by using rectangular tiles (3×9). This example shows how the proposed criteria are able to preserve fine details (due to GVF capabilities) maintaining at the same time the global orientation of almost all edge present in the original picture (due to the tile positioning rules).

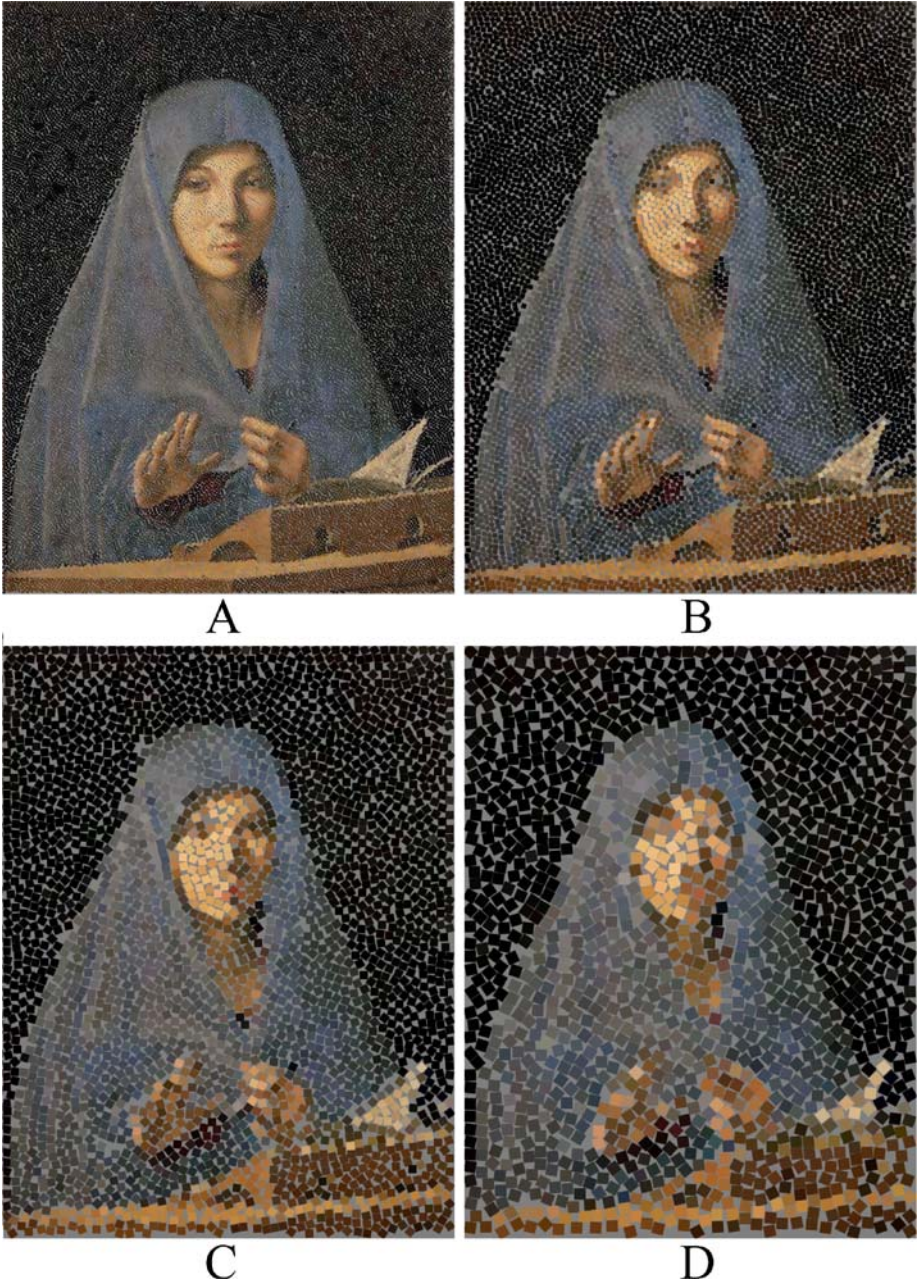


Fig. 4. Mosaics generated with increasing tiles size A (3x3), B (6x6), C (10x10), D (14x14)

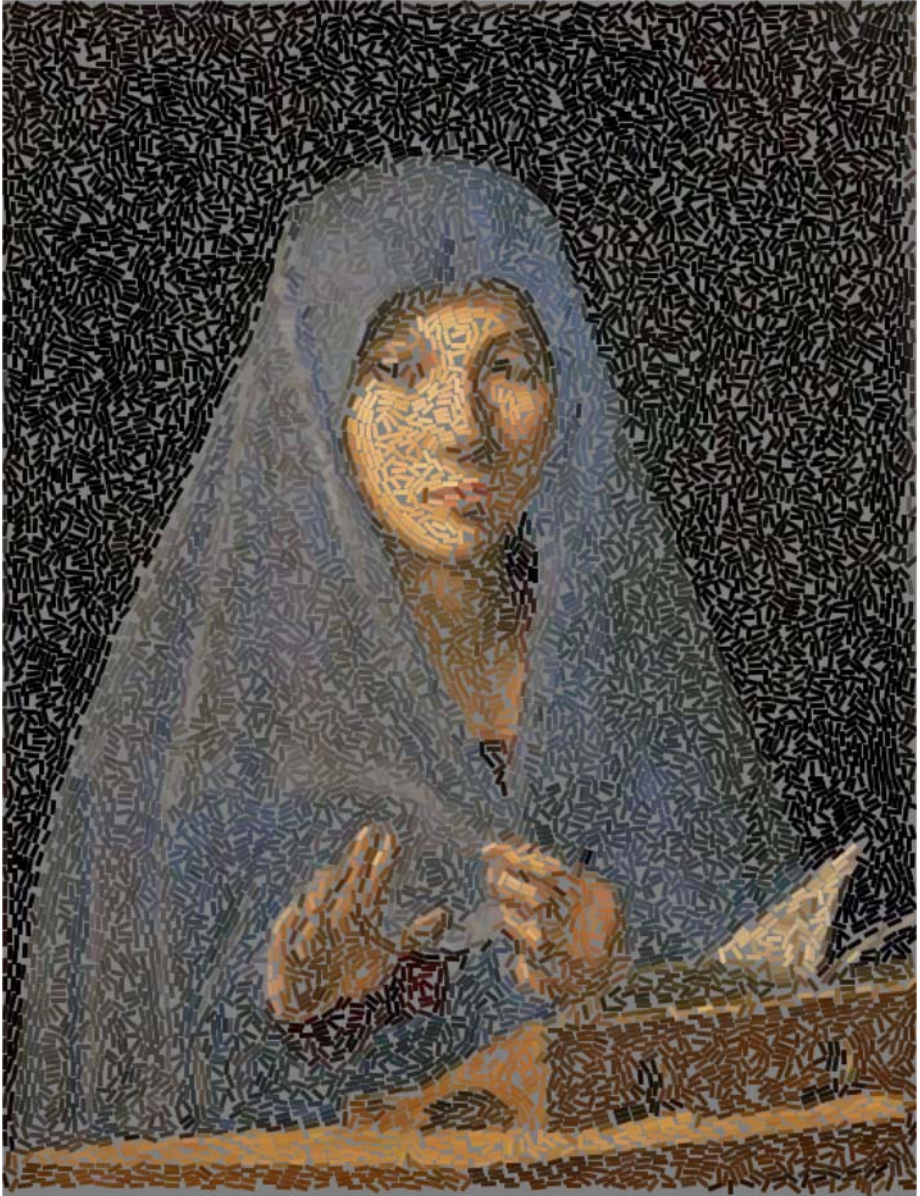


Fig. 5. An example of mosaic generated with rectangular tiles (3x9)

The overall complexity of the proposed technique is $O(kn) + O(n \log n)$, where n is the number of pixels in the source image. Further results are depicted in Figures 6, 7. The mosaicked images can be also downloaded at the following web address: <http://svg.dmi.unict.it/iplab/download/>

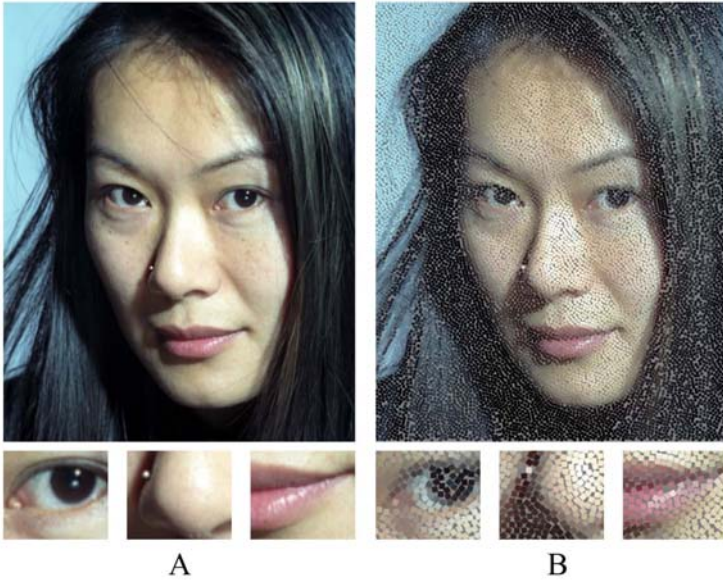


Fig. 6. Input image (A) and its mosaic (B) generated by our approach (image size 595x744, tile size 5x5)

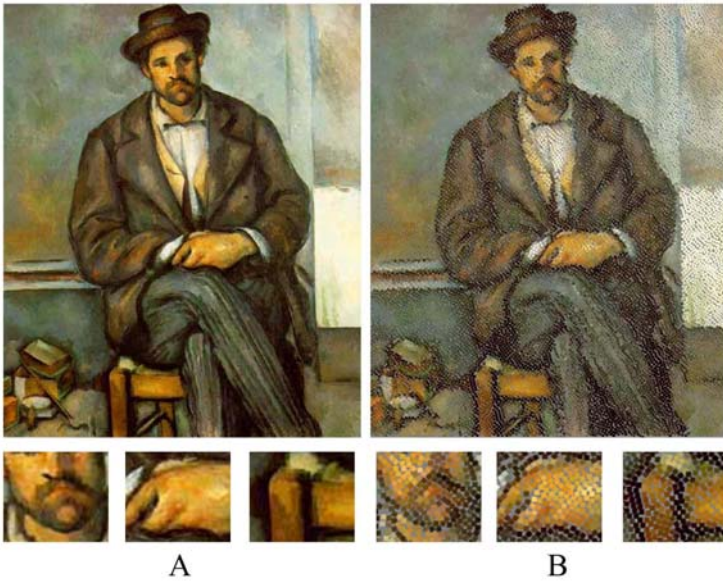


Fig. 7. Input image (A) and its mosaic (B) generated by our approach (image size 532x646, tile size 4x4)

4 Conclusions

We propose a novel technique to produce a traditionally looking mosaic from a digital source picture. The new technique tries to overcome the difficulties that rely on edge detection, using the Gradient Vector Flow. Tests show that the new technique produces aesthetically pleasant images that have a greater fidelity in dealing with fine details and a better management of gaps. The proposed technique does not cut tiles. Indeed the next research step will integrate heuristics like those proposed in [2] and [3] to cut tiles with the proposed method. Future works will be also devoted to color management and mosaic generation in vectorial format without using raster-to-vector conversion techniques [11].

References

1. Hausner, A.: Simulating decorative mosaics. In: Proc. SIGGRAPH 2001, pp. 573–580 (2001)
2. Di Blasi, G., Gallo, G.: Artificial mosaics. *The Visual Computer* 21(6), 373–383 (2005)
3. Battiato, S., Di Blasi, G., Farinella, G.M., Gallo, G.: A novel technique for opus vermiculatum mosaic rendering. In: Proc. ACM/WSCG 2006(2006)
4. Liu, Y., Veksler, O., Juan, O.: Simulating classic mosaics with graph cuts. In: Yuille, A.L., Zhu, S.-C., Cremers, D., Wang, Y. (eds.) EMMCVPR 2007. LNCS, vol. 4679, pp. 55–70. Springer, Heidelberg (2007)
5. Faustino, G.M., de Figueiredo, L.H.: Simple adaptive mosaic effects. In: Proc. SIBGRAPI 2005, pp. 315–322 (2005)
6. Elber, E., Wolberg, G.: Rendering traditional mosaics. *The Visual Computer* 19(1), 67–78 (2003)
7. Schlechtweg, S., Germer, T., Strothotte, T.: Renderbots — multi-agent systems for direct image generation. *Computer Graphics Forum* 24(2), 137–148 (2005)
8. Battiato, S., Di Blasi, G., Farinella, G.M., Gallo, G.: Digital mosaic frameworks - an overview. *Computer Graphics Forum* 26(4), 794–812 (2007)
9. Xu, C., Prince, L.: Snakes, shapes, and gradient vector flow. *IEEE Transactions on Image Processing* 7(3), 359–369 (1998)
10. Battiato, S., Di Blasi, G., Gallo, G., Guarnera, G.C., Puglisi, G.: Artificial mosaics by gradient vector flow. In: Proc. EUROGRAPHICS 2008 (2008)
11. Battiato, S., Farinella, G.M., Puglisi, G.: Statistical based vectorization for standard vector graphics. In: Alexandrov, V.N., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2006. LNCS, vol. 3992, pp. 334–341. Springer, Heidelberg (2006)