Novel Artificial Mosaic Generation Technique Paper Discussion

By - Aishwarya Pani
and Tabitha Roemish

BE BOUNDLESS

Information about the paper

- A Novel Artificial Mosaic Generation Technique Driven by Local Gradient Analysis[1]
- Published June 2008
  - Conference: Computational Science - ICCS 2008, 8th International Conference, Kraków, Poland, June 23-25, 2008, Proceedings, Part II
- Authors
  - Sebastiano Battiato
    - University of Catania | UNICT · Department of Mathematics and Computer Science (DMI)
  - Gianpiero di Blasi
    - University of Catania, Italy | Computer Science, Ph.D.
  - Giuseppe Claudio Guarnera
    - The University of York · Department of Computer Science
  - Giovanni Gallo
    - University of Catania | UNICT · Department of Mathematics and Computer Science (DMI), Ph.D.
  - Giovanni Puglisi
    - Università degli studi di Cagliari | UNICA · Department of Mathematics and Information Technology, Ph.D.

[1] Battiato, Sebastiano & Blasi, Gianpiero & Gallo, Giovanni & Guarnera, Giuseppe & Puglisi, Giovanni. (2008). A Novel Artificial Mosaic Generation Technique Driven by Local Gradient Analysis. 5102. 76-85. 10.1007/978-3-540-69387-1_9.
[2]: https://www.timetravelturtle.com/mosaics-ravenna-art-unesco/

[2]

Abstract

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.

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

PC: https://etc.worldhistory.org/photos/7-roman-mosaics/

Why mosaics?

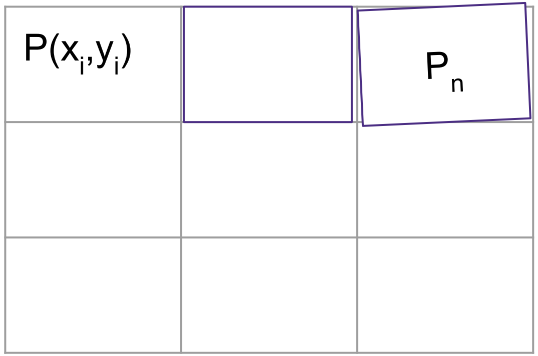
- Art can “provide hints for technological innovations”
- One of the first methods of synthesizing an image with discrete units
- It can be a challenging problem

UNIVERSITY of WASHINGTON

Introduction

1. What is a mosaic?
2. What is a mosaic mathematically?
3. How are others creating mosaics digitally?

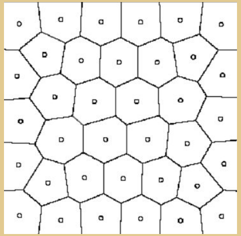
Image (I) in Plane (R^2)
Vector field $\mathbf{\Phi}(x,y)$



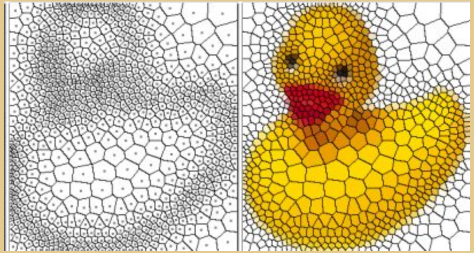
N sites $P_i(x_i,y_i)$

UNIVERSITY *of* WASHINGTON

Related Work

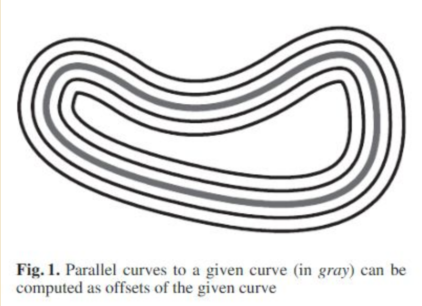


Centroidal Voronoi Diagram [3]

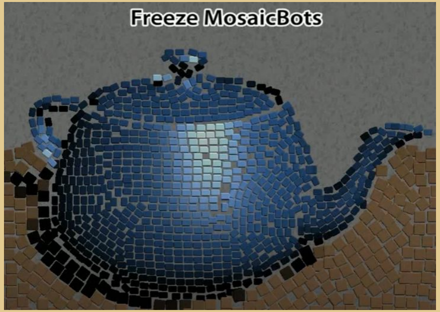


Centroidal Voronoi Diagram & density function [7]

- Simulating decorative mosaics. Hausner, A. [3]
    - Notes: Centroidal Voronoi Diagram, user selected features, Manhattan distance. Computationally slow
- Simple adaptive mosaic effects. Faustino, G.M., de Figueiredo, L.H.[7]
    - Notes: Uses centroidal Voronoi diagram and a density function to size tiles based on features.
- Rendering traditional mosaics. Elber, E.,Wolberg, G.[8]
    - Obtains curves from image and lays tiles along those curves.
- Renderbots — multi-agent systems for direct image generation. Schlechtweg, S., Germer, T., Strothotte, T.[11]
    - Uses stroke-based rendering. Divides image into bots that run a simulation and execute painting function.



Curve with guidelines for tiles [8]



renderBots [11]

Related Work

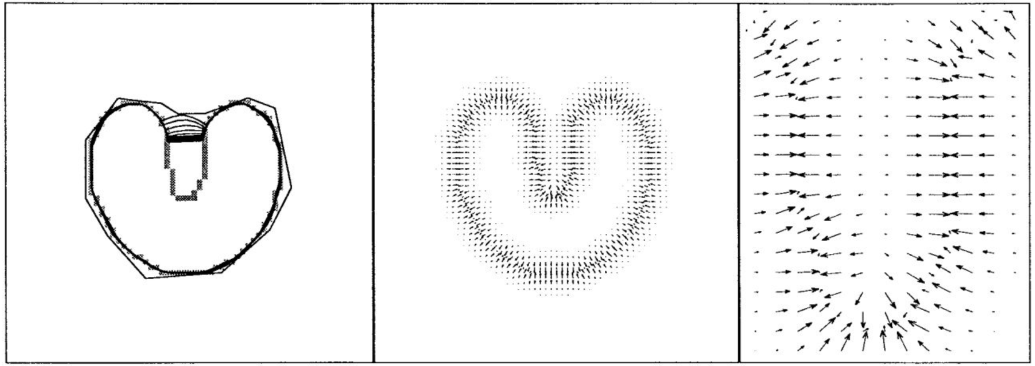
- Digital mosaic frameworks - an overview. **Battiato, S.**, **Di Blasi, G.**, Farinella, G.M., **Gallo, G. [10]**
    - A survey of the current techniques in mosaic image processing.
- A novel technique for opus vermiculatum mosaic rendering. **Battiato, S.**, **Di Blasi**, G., Farinella, G.M., **Gallo, G**. [5]
- Artificial mosaics. **Di Blas**i, **G.**, **Gallo, G. [4]**
    - Directional guidelines and distance transform
- Simulating classic mosaics with graph cuts. Liu, Y., Veksler, O., Juan, O.[6]
    - Notes: graph-cut optimization algorithm (energy minimization), tiles positioning without an explicit edge detection phase

Using Graph cuts [6]

Image reference: Snakes, Shapes, and Gradient Vector Flow Chenyang Xu, Student Member, IEEE, and Jerry L. Prince, Senior Member, IEEE

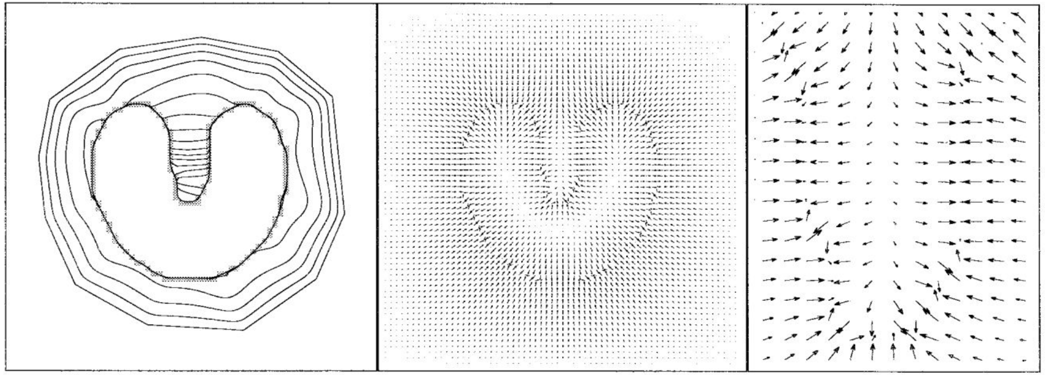
Proposed Algorithm

- Mosaic Generation is based on two main steps: GVF (Gradient Vector Flow) field computation and rule-based tile positioning.
- GVF snake algorithm is better than traditional snake algorithm.
- GVF has insensitivity to initialization and ability to move into boundary concavities



(a) (b) (c)

Fig. 1. (a) Convergence of a snake using (b) traditional potential forces, and (c) shown close-up within the boundary concavity.



(a) (b) (c)

Fig. 3. (a) Convergence of a snake using (b) GVF external forces, and (c) shown close-up within the boundary concavity.

Proposed Algorithm cont.

Image reference: Hausner, A.: Simulating decorative mosaics. In: Proc. SIGGRAPH 2001. (2001) 573–580

- GVF field obtained through energy minimization
- Balancing smoothness and alignment with image gradient

Energy Function:

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

First term

Second term

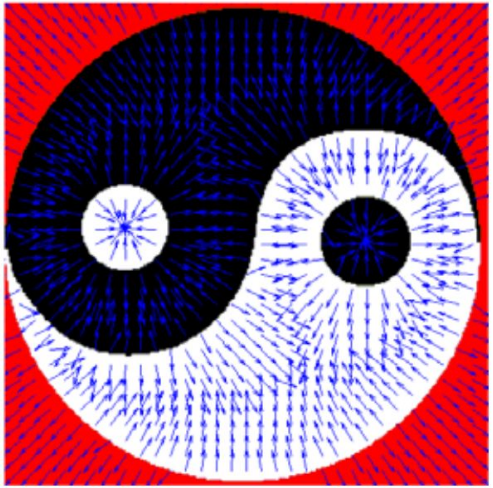


Fig 1: Input image and its corresponding GVF field.

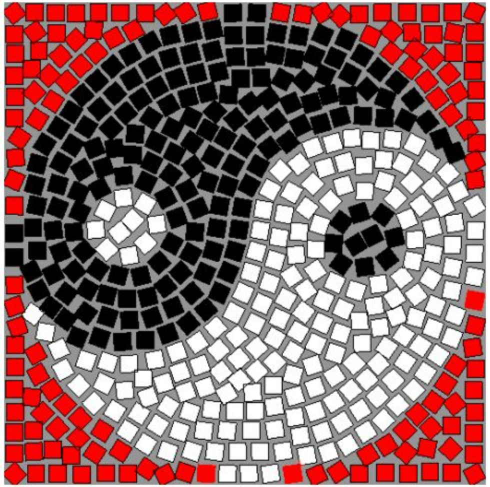


Fig 2: Tile placement according to edge alignment.

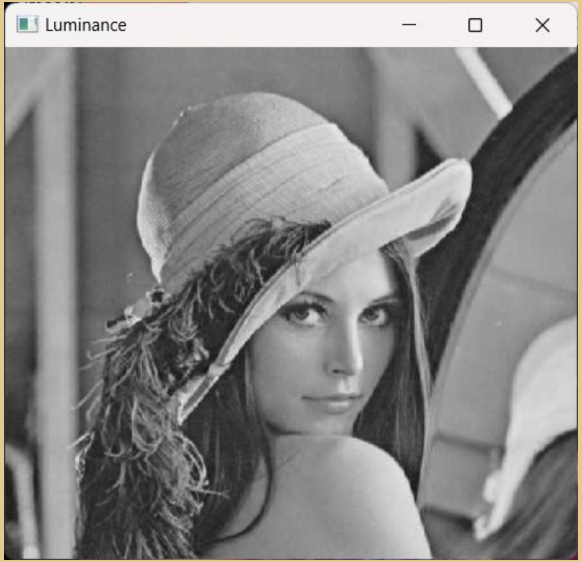
Proposed Algorithm cont.

Luminance Transformation and Gradient Distribution:

- Transformed to luminance channel for eliminating of hue and saturation information
- Luminance equalization for normalized gradient distribution

Gradient Computation:

- Robert's Kernel for adding randomness and sensitivity to noise



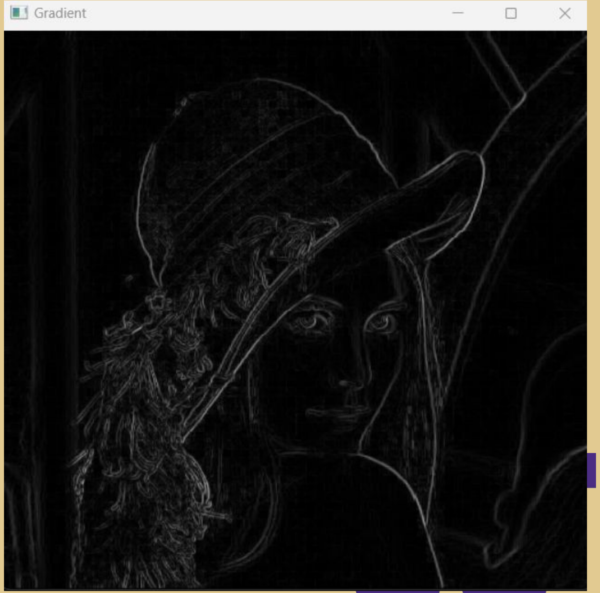


Image reference: Hausner, A.: Simulating decorative mosaics. In: Proc. SIGGRAPH 2001. (2001) 573–580

Proposed Algorithm cont.

Two-Step Tile Placement with GVF Field:

- Capture of main edges in the input images and align tiles with image's perceptual orientation:
    - Identifying local maxima of |GVF(I)|
    - Threshold-based selection and sorting of pixels
    - Chains of tiles placed following GVF direction at central points
- Homogeneous Region Coverage:
    - Systematic placement of tiles from left to right, top to bottom
    - Dense packing of tiles to cover significant pixel area

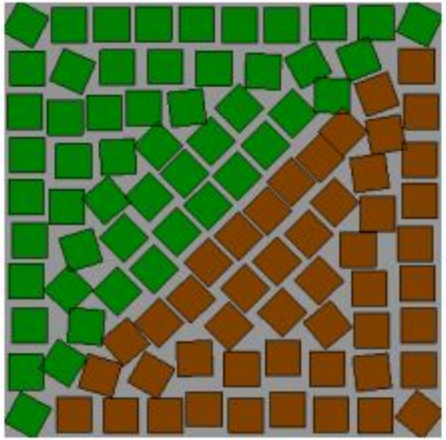
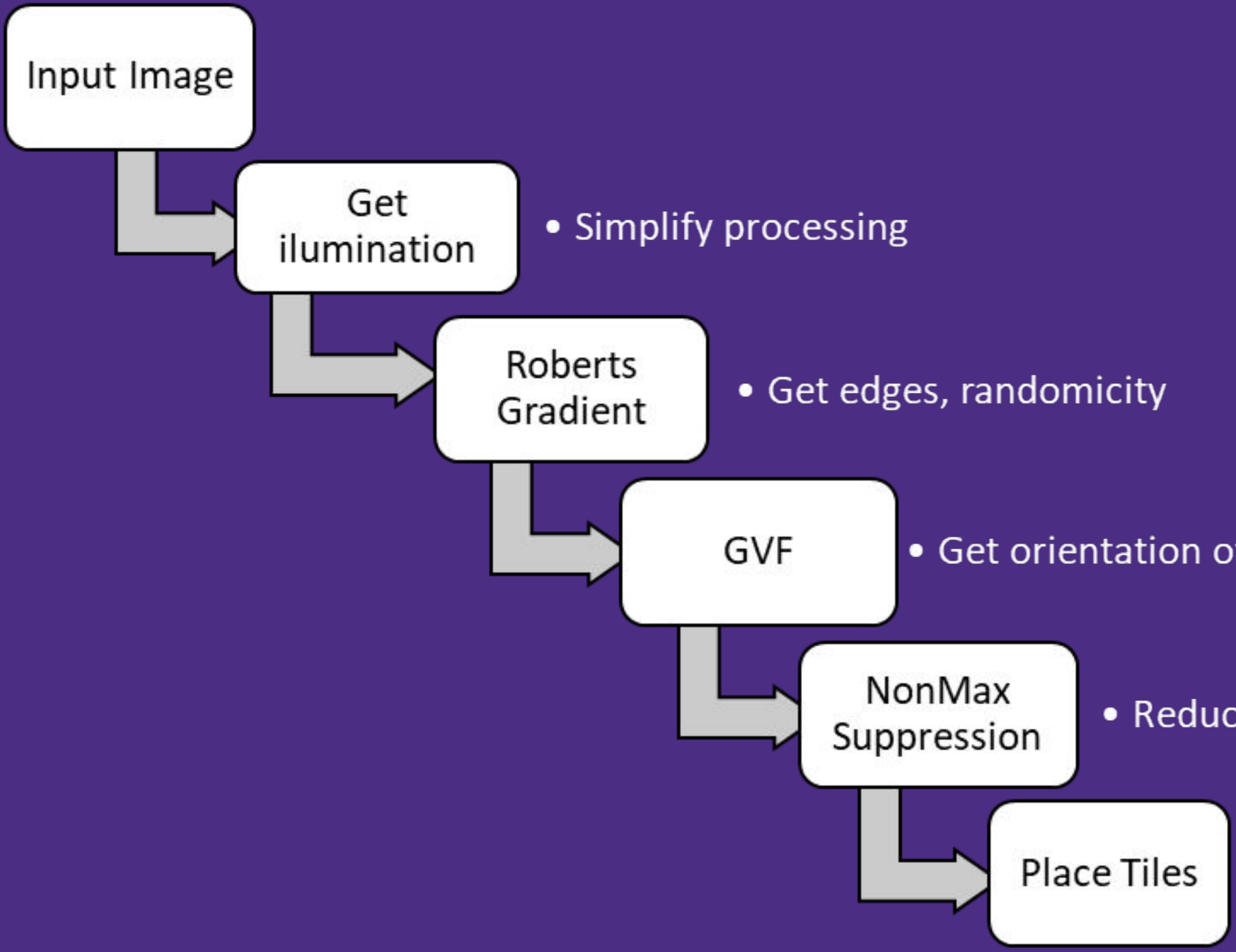


Fig 1: Tile placement

Pseudocode

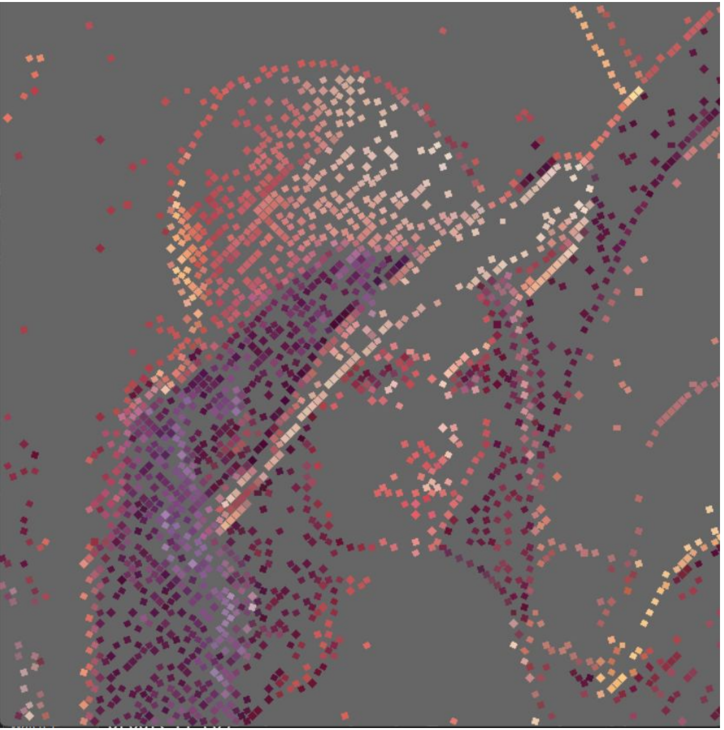
1. Input: a raster image I
2. $L(I) \leftarrow$ Luminance(I)
3. $G(I) \leftarrow$ Robert's Gradient(Equalize($L(I)$))
4. $[u, v] \leftarrow GVF(|G(I)|_\infty, \mu, nIterations)$
5. $gvf(I) \leftarrow (u^2 + v^2)^{1/2}$
6. $I_n(I) \leftarrow$ NonMaxSuppression($gvf(I)$)
7. let t_h, t_l be threshold values, 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 length(I)
20.   for $i \leftarrow 1$ to 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.

UNIVERSITY *of* WASHINGTON



- Simplify processing
- Get edges, randomicity
- Get orientation of edge
- Reduce edge points
- Use heuristics

Tile Placement



1. Fill queue, Q, in descending order of tiles that are greater than t_h
2. While Q not empty
   a. If tile is not marked,
      i. **Place tile** at angle specified by GVF
         1. **If overlap, skip tile positioning**
      ii. Starting from tile retrieved from Q, move in perpendicular directions to tile angle, finding a tile that is greater than t_l and is not already marked,
      iii. **Place tile** at angle specified by GVF at center
         1. **If overlap, skip tile positioning**
      iv. Continue until you reach a marked tile or a tile below threshold
      v. If you've gone in both directions and placed all the relevant tiles, move to next tile in Q
3. Fill in **remaining tiles placing** them at angle according to GVF (These tiles are less than t_l)
   a. **If overlap, skip tile positioning**

Pseudocode to code

Challenges:

a. Language and idea conveyance
   i. Tile placement vs positioning?
b. Differences in programming language
   i. Lay a tile before checking if placement is okay?
c. Missing details
   i. Thresholds?

Results from the paper

Aesthetic Quality Comparison:

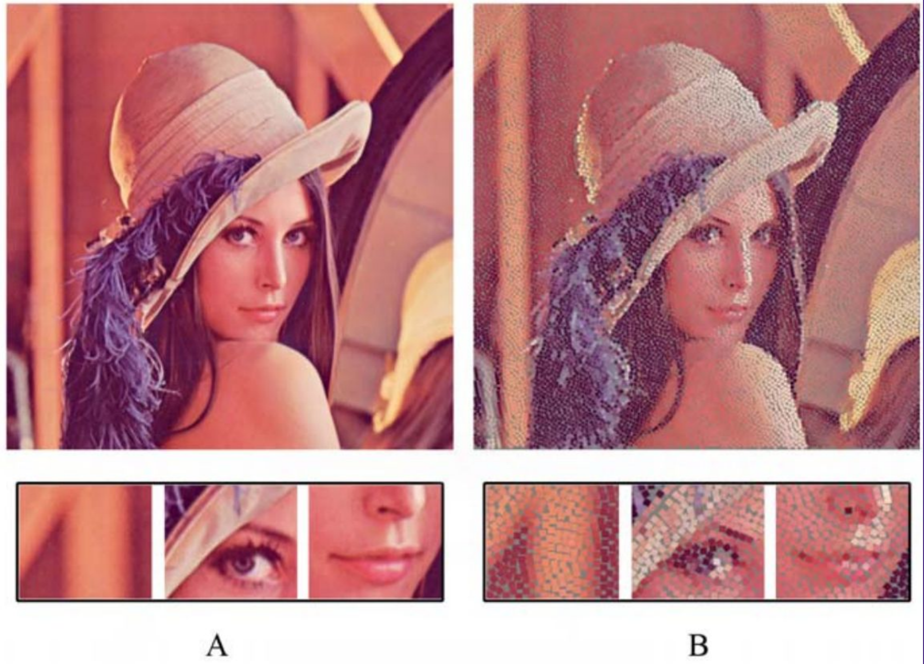
- The proposed algorithm preserves fine details and high-frequency areas are prioritized in tile placement.

Untiled Space and Coverage:

- Proposed technique has better distribution of gaps.
- Smoothness achieved in regions where picture C exhibits cracks
- Less uncovered area compared to picture D but the texture obtained is less chaotic

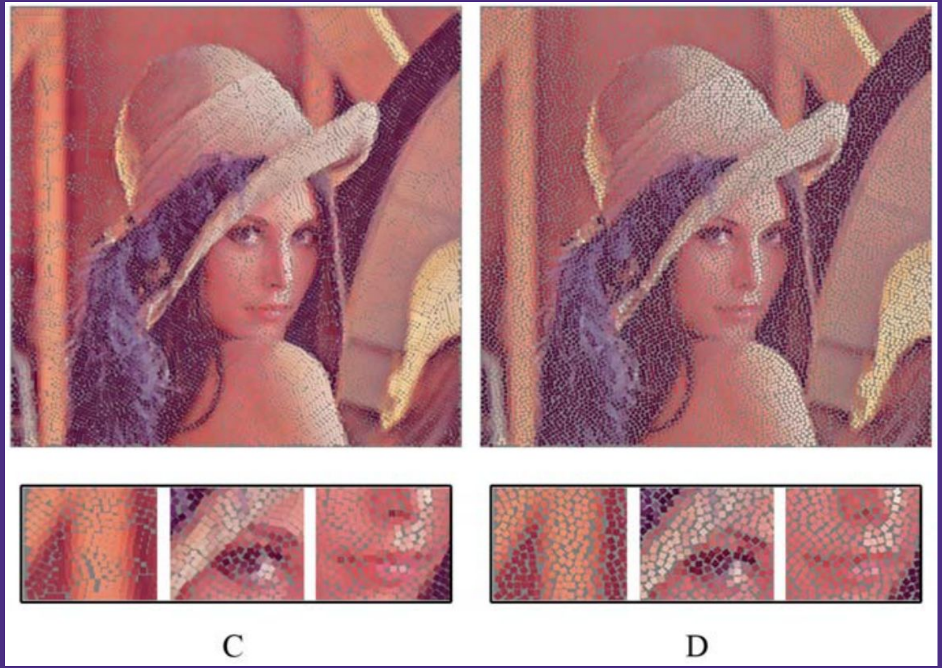
Input image

Mosaic image obtained from the proposed algorithm



Mosaic image: Di Blasi, G., Gallo, G.: Artificial mosaics. The Visual Computer 21(6) (2005) 373–383

Mosaic image: Liu, Y., Veksler, O., Juan, O.: Simulating classic mosaics with graph cuts. In: Proc. EMMCVPR. (2007) 55–70



Results cont.

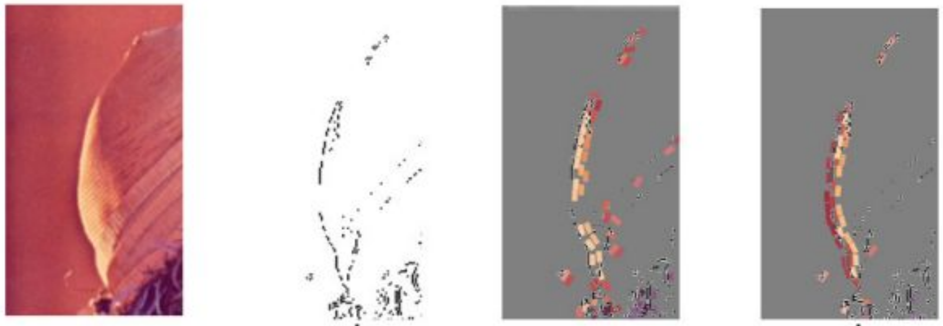
Table compares the number of tiles covered by using each technique:

a. **Our method**
b. **Di Blasi, G., Gallo, G.: Artificial mosaics. The Visual Computer 21(6) (2005) 373–383**
c. **Liu, Y., Veksler, O., Juan, O.: Simulating classic mosaics with graph cuts. In: Proc. EMMCVPR. (2007) 55–70**

The Figure presents a comparison between the paper approach and the approach: Battiato, S., Di Blasi, G., Gallo, G., Guarnera, G.C., Puglisi, G.: Artificial mosaics by gradient vector flow. In: Proc. EUROGRAPHICS 2008. (2008)

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

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



a b c d

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.

Results cont.

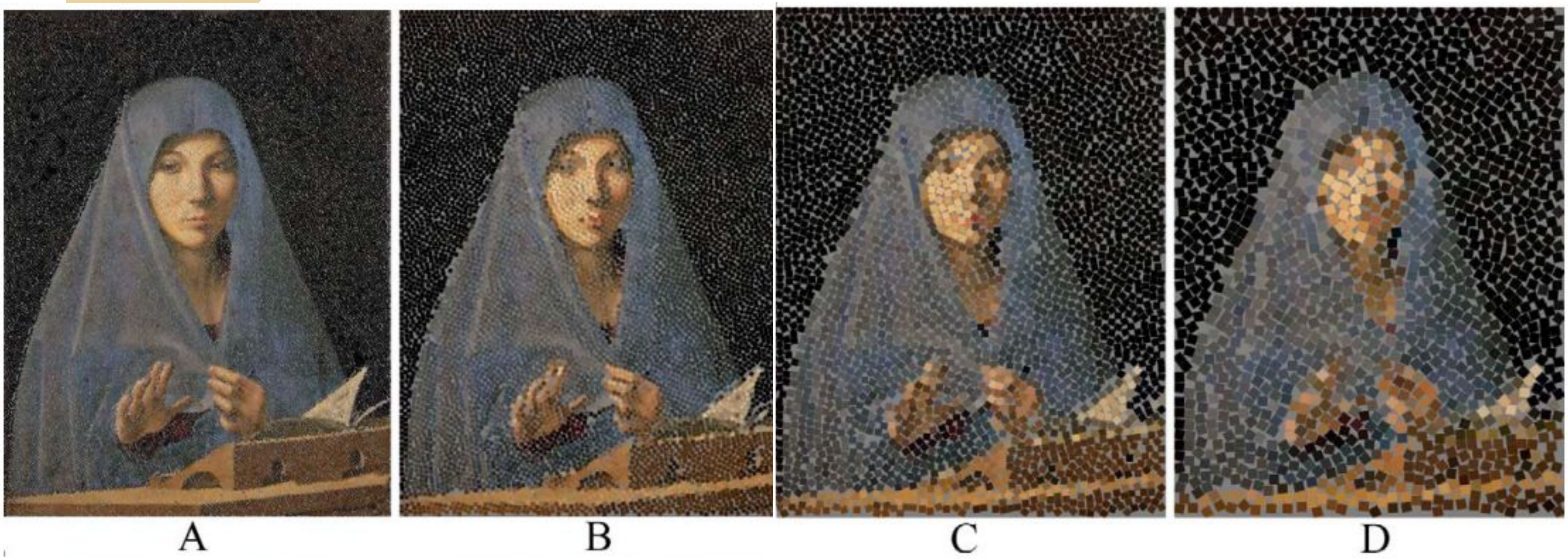


Fig. 4. Mosaics generated with increasing tiles size A (3x3), B (6x6), C (10x10), D (14x14).

Results cont.

- Mosaic generated from rectangular tiles

- The complexity of the proposed algorithm is expressed as O(kn) + O(nlogn),

  where n represents the number of pixels in the source image.

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

Results cont.

Few more mosaic examples using the proposed technique

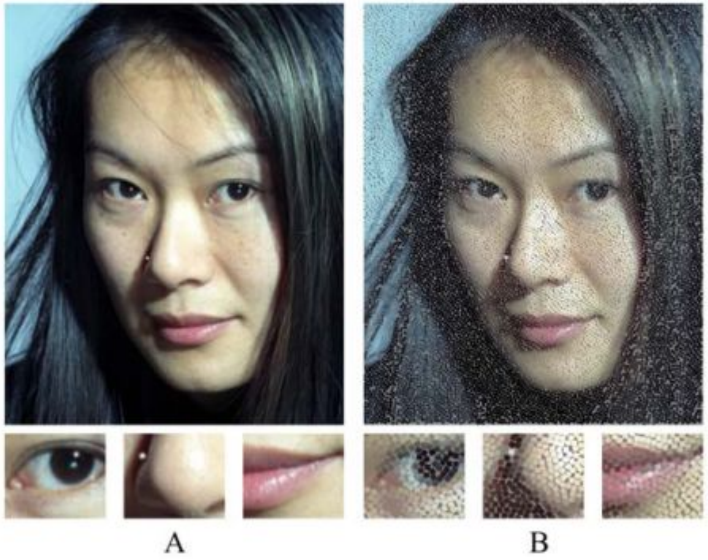


Fig. 6. Input image (A) and its mosaic (B) generated by our approach (image size 595x744, tile size 5x5).

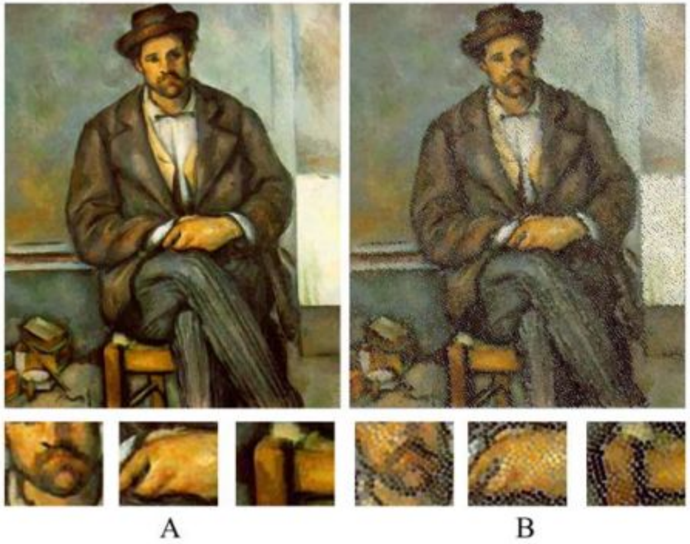


Fig. 7. Input image (A) and its mosaic (B) generated by our approach (image size 532x646, tile size 4x4).

Conclusion

- Proposed technique for creating a traditional-looking mosaic from a digital source image
- Utilizes Gradient Vector Flow (GVF) to address challenges associated with edge detection
- Tests demonstrate aesthetically pleasing mosaic images with improved fidelity and better gap management
- No tile cutting involved in the proposed technique
- Future research direction: integrating heuristics for tile cutting based on the proposed method
- Author’s interest in exploring color management and vectorial mosaic generation without raster-to-vector conversion techniques

Questions?

References

1. Images pulled from research paper: Battiato, Sebastiano & Blasi, Gianpiero & Gallo, Giovanni & Guarnera, Giuseppe & Puglisi, Giovanni. (2008). A Novel Artificial Mosaic Generation Technique Driven by Local Gradient Analysis. 5102. 76-85. 10.1007/978-3-540-69387-1_9.
2. https://en.wikipedia.org/wiki/Gradient_vector_flow
3. Alejo Hausner. 2001. Simulating decorative mosaics. In Proceedings of the 28th annual conference on Computer graphics and interactive techniques (SIGGRAPH '01). Association for Computing Machinery, New York, NY, USA, 573–580. https://doi.org/10.1145/383259.383327
4. Di Blasi, G., Gallo, G.: Artificial mosaics. The Visual Computer 21(6) (2005) 373–383
5. Battiato, S., Di Blasi, G., Farinella, G.M., Gallo, G.: A novel technique for opus vermiculatum mosaic rendering. In: Proc. ACM/WSCG 2006. (2006)
6. Liu, Y., Veksler, O., Juan, O.: Simulating classic mosaics with graph cuts. In: Proc. EMMCVPR. (2007) 55–70
7. Faustino, G.M., de Figueiredo, L.H.: Simple adaptive mosaic effects. In: Proc. SIBGRAPI 2005. (2005) 315–322
8. Elber, E.,Wolberg, G.: Rendering traditional mosaics. The Visual Computer 19(1)(2003) 67–78
9. Schlechtweg, S., Germer, T., Strothotte, T.: Renderbots — multi-agent systems for direct image generation. Computer Graphics Forum 24(2) (2005) 137–148
10. Battiato, S., Di Blasi, G., Farinella, G.M., Gallo, G.: Digital mosaic frameworks - an overview. Computer Graphics Forum 26(4) (2007) 794–812
11. Xu, C., Prince, L.: Snakes, shapes, and gradient vector flow. IEEE Transactions on Image Processing 7(3) (1998) 359–369
12. RenderBots - Multi-Agent Systems for Direct Image Generation - https://www.youtube.com/watch?v=sCA9eVcd1G0