Image Mosaics

Adam Finkelstein and Marisa Range

Technical Report: TR-574-98
Computer Science Department, Princeton University
Princeton, NJ 08540, USA

http://www.cs.princeton.edu/gfx/proj/mosaic

This paper will appear in:

Roger D. Hersch, Jacques André, and Heather Brown (eds.),
Electronic Publishing, Artistic Imaging and Digital Typography,
Proceedings of the EP'98 and RIDT'98 Conferences,
St Malo: March 30 -April 3, 1998,
Lecture Notes in Computer Science Series, number 1375,
Springer-Verlag: Heidelberg, 1998.

Image Mosaics

Adam Finkelstein and Marisa Range

Technical Report: TR-574-98
Computer Science Department, Princeton University
Princeton, NJ 08540, USA
http://www.cs.princeton.edu/gfx/proj/mosaic

Abstract. We describe a process for creating an *image mosaic*—a collection of small images arranged in such a way that when they are seen together from a distance they suggest a larger image. To visually suggest the larger form, the small images are arranged to match a large picture as much as possible, and then their colors are adjusted to better suggest the overall form. Arrangement of the small images may be either manual or automatic. Adjustment of the colors in the small image to further suggest the larger picture is fully automatic and employs a new color correction scheme that generalizes traditional halftoning.

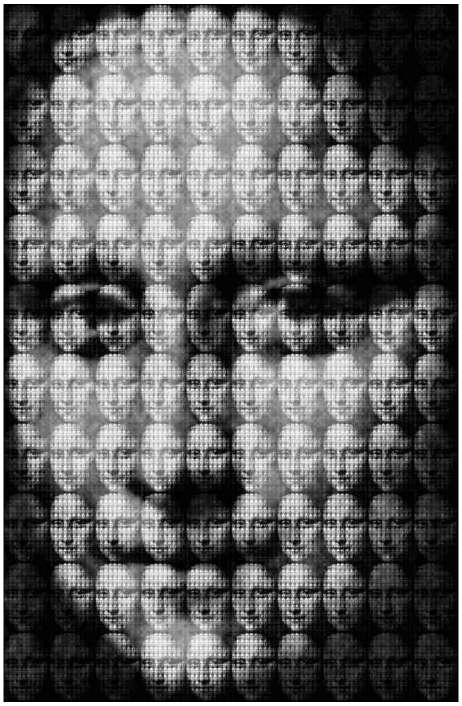


Fig. 1. Mona Lisa (three layers)

Fig. 2. John F. Kennedy

1 Introduction

Painters of the impressionist movement exploited a property of the human visual system that combines colors in a region such that the observer sees an overall average color for that region. When viewed up close, an impressionist painting appears to be a collection of small brush strokes of various colors, whereas at a distance those brush strokes combine to yield an overall impression that is typically the subject of the painting. More recently, artists and photographers have exploited the same principle to produce *image mosaics*—layered imagery, where the subject of the work is both the tiny features that only can be seen up close and the large scale features that only can be seen at a distance. This paper explores the use of computers to automatically or semi-automatically produce such imagery.

In this paper we describe methods for arranging a set of small images that we call *tile images* and adjusting their colors so that together they suggest a larger form, as shown in Figures 1, 2, and 3. The motivation for this work is

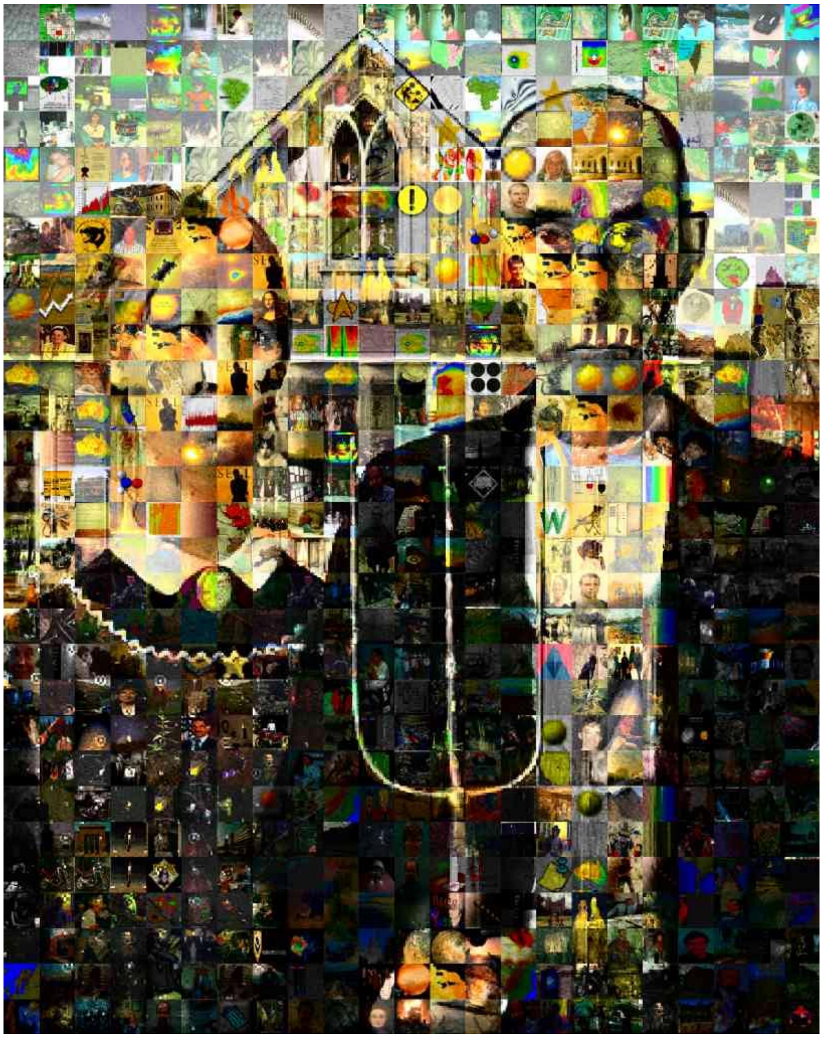


Fig. 3. *American Gothic* composed of pictures from the Web. (Color image)

primarily for artistic purposes. The ability to combine groups of pictures in this way affords opportunities for both aesthetic and associational juxtaposition of images. Additionally, there are other potential applications for this kind of technology that we have not emphasized in this work: encoding extra information in an image for transmission or security, new forms of halftoning screens for printing, and association of images for advertising.

The rest of this paper is organized as follows. In Sections 2 and 3, we describe related work and provide some background on traditional halftoning. Section 4 presents our technique for creating image mosaics. In section 5, we describe the individual images that appear in the paper. Section 6 concludes with areas for future work.

2 Related work

Artists have known for some time how to create pictures out of other pictures. Indeed, the efforts described in this paper were largely inspired by Salvador Dali's lithograph *Lincoln in Dalivision*. In his etching [2], Dali shows his wife Gala looking up through a block window at the Godhead. Up close, Gala is surrounded by a jumble of colors, textures, and small caricatures. From a distance, the entire work blends into a bust of Abraham Lincoln. A similar layered quality is exhibited in *Self Portrait I* by Chuck Close [1] in which the large-scale figure is his own face, but it is composed of hundreds of tiny abstract figures. To produce these works, the artists had to "see" the large-scale image before it was actually created, and then exactingly reproduce it by arranging and adjusting the smaller figures from which it was composed. In this paper, we describe a process for creating such images more automatically, while still providing the means for artistic expression through composition.

Several researchers have investigated the use of computers to produce pictures in the style of impressionist paintings. Haeberli described a method of creating an image whose overall form matches a given picture, but is composed of tiny brush strokes [5]. Meier extended this work to apply to 3D animation [9], and Litwinowicz subsequently extended it to video [7]. Like these projects, our work begins with the larger form; however, rather than using small brush strokes, we use small images to convey the larger form.

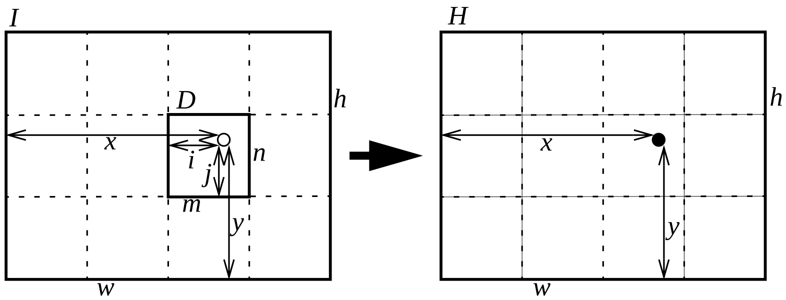
The process we describe is similar to the artistic screening technique developed by Ostromoukhov and Hersch [11]. In their approach, variations in brightness across the larger image are produced by varying the sizes and shapes of tiny subjects (for example fish, birds, or abstract blobs) or characters (for example, Roman letters, Kanji, or Islamic calligraphy). The tiny figures are described by closed contours separating black from white, so the entire work is ultimately composed of black and white. In contrast, our method varies the brightness of tile images composed of shades of gray (*grayscale*), so that the resulting mosaic is itself a grayscale image. Because we are working with images, we are also able to show how these techniques generalize from grayscale to color.

Photographers, and more recently researchers in computer vision, have addressed the problem of constructing a *panoramic mosaic*—a single, consistent view of a scene pieced together from a series of photographs capturing the scene from different perspectives [12]. The objective of panoramic mosaicing is to produce a mosaic which does *not* reveal that it was stitched together from different images. This contrasts with our goal of constructing a mosaic wherein the visual artifacts of the tile images are part of the subject of the work.

Our technique appears to be most similar to that of Silvers [13] who has applied this technology for both artistic and commercial purposes. It is difficult to describe the work precisely, as his techniques remain proprietary. It appears that Silvers focuses most of his effort on finding a suitable arrangement for the tile images (the subject of Section 4.3) and avoids correcting the tile images after arranging them (Section 4.4). Silvers tends to use many more image tiles in his mosaics, and is able to produce stunning reproductions of an original image without altering the tile images.

3 Background

Traditional halftoning employs black dots of varying sizes arranged in a regular grid to convey various shades of gray. Historically, the sizes of the dots have been "calculated" by a purely mechanical process called *screening*: the target image is photographed slightly out of focus through a mesh or screen on high-contrast film. With the advent of the digital age, screening is now performed almost exclusively by computers. While there are a host of schemes for computing the sizes, shapes and arrangement of dots of ink that faithfully reproduce the target image, many of them share a common feature called a *dither matrix*. Since this matrix plays a role similar to that of our small images tiles, we'll describe how it works.



Suppose we have (as shown above) a target image I of width w and height h whose pixels are grayscale values ranging from 0 for black to 1 for white. We wish to produce a wxh halftoned image H, each pixel $H[x,y]$ of which will either be black (ink) or white (paper). The most common approaches employ a mxn dithering matrix D, where $m<<w$ and $n<<h$. For each pixel $I[x,y]$, we find the entry $D[i,j]$ using:

$$i = x \text{ modulo } m$$
$$j = y \text{ modulo } n$$

We choose black or white dots for our halftoned image as follows: if $I[x,y] < D[i,j]$ then $H[x,y]$ is black; otherwise it is white. The key to designing an effective screening method is finding a good dither matrix D. There is a rich literature

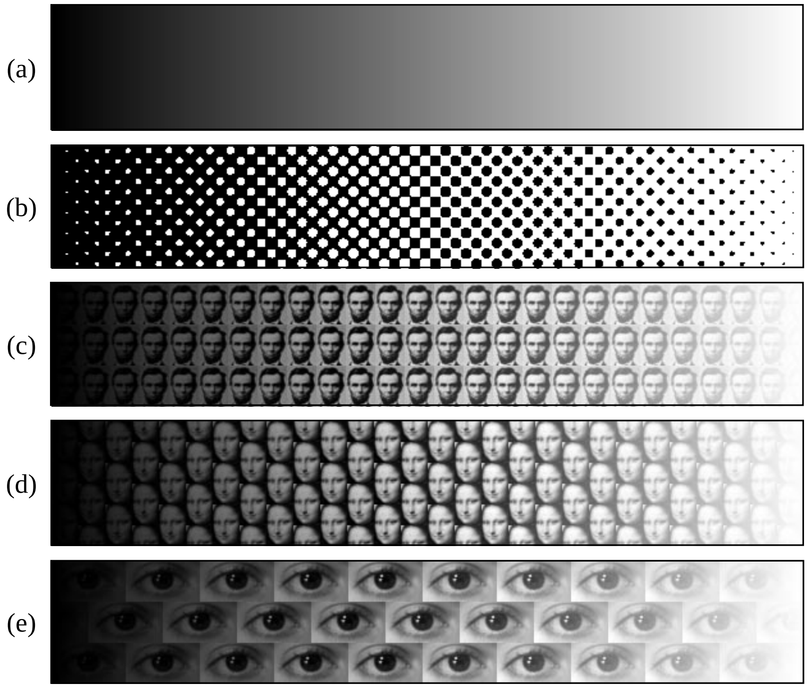


Fig. 4. (a) original grayscale ramp, (b) clustered-dot dither,
(c-e) image mosaics composed of Lincoln, Mona Lisa, and an eye.

on the subject [14] and we will not review it further here. Figure 4a shows a ramp, which is halftoned in Figure 4b using a 16x8 clustered-dot ordered dither [14].

4 Creating image mosaics

In this section we describe how to create an image mosaic. Specifically, we'll address the problem: given a collection of tile images T_i and a target image I, create an image mosaic M that resembles I at a coarse scale but is composed of a tiling of the T_i's. We solve the problem in four steps: (1) choose images, (2) choose a tiling grid, (3) find an arrangement for the image tiles within the grid, and (4) correct the tiles to match the target image. The first two stages (and often the third) are performed manually, and are typically iterative. These four steps are discussed in the following sections.

4.1 Choosing images

The choice of subject matter is purely an artistic one. Clearly, one of the attractions of image mosaics is that they juxtapose images at different scales. This property may be interpreted through composition of images themselves. For example, in Figure 5, a large picture of an abalone shell is screened through a group of micron-scale photographs of abalone shell from a scanning electron microscope. Alternately, the juxtaposition may be one of association, as in Figure 2 showing John F. Kennedy composed of smaller pictures of Marilyn Monroe. This association may be put to use for commercial purposes, such as advertisement; Silvers [13] created a picture of George Washington using hundreds of tiny credit cards, as part of a campaign for Mastercard.

Some images simply work better than others. Recognition plays an important role, so iconic figures such as political leaders, actors, famous works of art, and well-known scenes are desirable. Furthermore, figures that are easy to recognize at very low resolution tend to work well; surely this was part of Dali's motivation for selecting Lincoln as his subject in his lithograph *Lincoln in Dalivision*. The distribution of colors in an image can also be a factor. Tile images with relatively uniform distributions of brightness tend to be easier to identify when their colors are adjusted as described in Section 4.4.

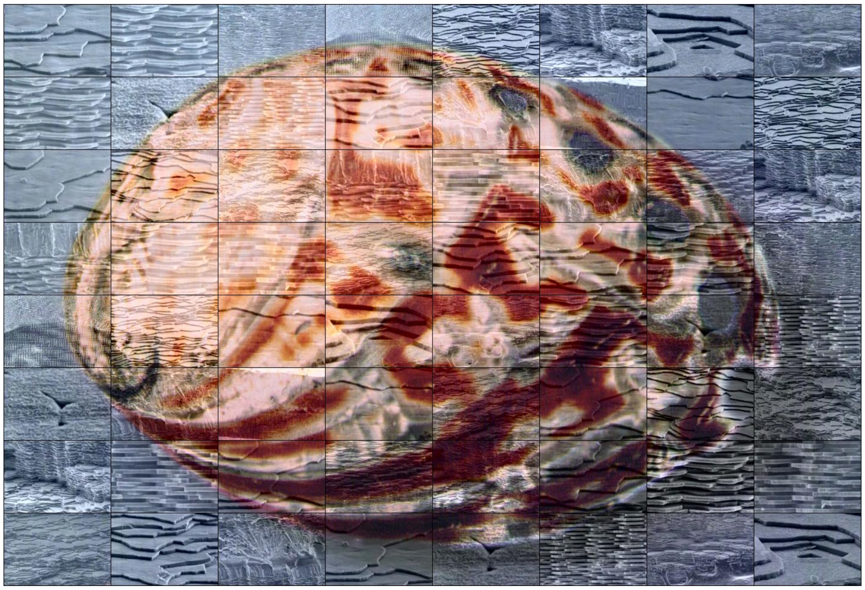


Fig. 5. An abalone shell screened through smaller pictures of abalone shells taken by a scanning electron microscope. (Color image)

4.2 Finding a tiling pattern

To find an appropriate arrangement for the image tiles, the first question that needs to be addressed is what kind of tiling will be used for the mosaic. We have investigated only semiregular rectangular grids for simplicity, although other tilings are possible. Hexagonal grids have also been used for traditional halftoning [14]. A more general (non-periodic) tiling might be found that achieves a better match for a given target image. However, optimizing over all possible tilings and all possible arrangements of small image within that tiling such that they best resemble a target image is computationally expensive and remains a challenge for future work. Furthermore, since the image tiles themselves tend to be rectangles, a rectangular tiling is a natural choice. Arranged in a rectangular grid, all the small image tiles must have the *same* aspect ratio. Thus, it is important to choose an aspect ratio appropriate for the entire set of image tiles, as they will either have to be cropped or stretched to conform to this ratio.

Among rectangular tilings, we have explored two varieties: a regular grid, as shown in Figure 4c, and angled grids as shown in Figures 4d and 4e. Studies in perception show that the eye is least sensitive to grids angled at 45^{o} and most sensitive to horizontal and vertical grids [14]. Thus, traditional screening methods have typically oriented the screen at an angle to reduce its visual impact. In the case of image mosaics, the image tiles are intended to be seen, and therefore the choice of whether or not to angle the grid is primarily an aesthetic decision.

4.3 Arranging the image tiles

Having selected a tiling grid, we need to place individual image tiles into the grid. We have explored a number of arrangement options:

- Use the same tile everywhere. (Figures 4c-e)
- Choose a random arrangement of the image tiles. (Figure 5)
- Arrange different tiles manually by eye. (Figure 2)
- Place image tiles by matching their average colors to the region of the target image that they cover. (Figure 8)
- Find a more detailed match between the tiles and the target image based on the shapes and colors within the images. (Figures 3 and 7)

For all but the first of these options, one has to consider whether a specific tile image may appear more than once in the final mosaic. Figure 7 does not permit repetition of the tile images; all other image mosaics shown here use repetition.

In order to convey the target image as effectively as possible, each location within the tile grid should contain the tile image that is most similar to the corresponding region of the target image. In this case, by “similar” we mean that the shapes, colors and textures of the tile image resemble those of the region in target image. Searching for the most similar tile image is a problem in content-based image retrieval, an area of active research. Several techniques that have been applied in this area are color histogram matching, texture analysis, shape analysis, edge matching, or a combination of these methods [10]. In creating Figures 3 and 7, we used the wavelets-based image matching algorithm due to Jacobs, Finkelstein and Salesin [6] to place tile images into the grid. The method uses wavelet analysis to distill each image down to a very small amount of data called a *signature*—essentially an extremely compressed version of the image, capturing only its broad forms and colors. Given the signature of a region in the target image, we can quickly search through the signatures of all possible tile images, choosing the one that best matches the target. To choose the best tile from among 20,000 images, the method requires less than a second on a conventional desktop workstation. It is beyond the scope of this paper to describe the method in detail; the reader is referred to [6] for a complete description.

4.4 Color correction

Now that we have placed the image tiles in some arrangement, perhaps based on the shapes or colors of the target image, the next task is to alter their colors to better match the target image. In most of this section, we describe the correction process assuming that we are working with a grayscale image. At the end of this section we generalize the process for color images by using the same method in each of three color channels.

Our objective is to match the color (brightness, in the case of our grayscale image) of the tile image to the color of the region in the target image that is covered by the tile. If the target image has a constant color x across this region, then we want to adjust the color of the tile image so that its average color is x. If the brightness of the target image ranges from dark on the left side of the region to light on the right side of the region, then we would like the brightness of the image tile to somehow match this gradient as well. In addition to coarsely matching the colors of the target image, we would like to preserve as much as possible the features of the tile images. The approach we have taken is inspired by the dither matrix-based halftoning schemes described in Section 3.

Specifically, we make use of a *correction rule*, which takes as input an image tile and a desired average color a, and generates a *correction function* $F: R^1 \to R^1$ that maps a color x in the image tile to a color $F(x)$ in the final mosaic such that the region of the mosaic covered by the image tile will have the average color a. There are a variety of families of correction functions that could fulfill this role. For example, the constant function $F(x)=a$ would achieve the correct overall average, but ignores the original colors of the image tiles. A more sensible solution might be to scale the colors in the input tile so that the desired average is achieved. Suppose, for example, that the desired average color a is less than the average color a_t of the image tile. Then our "scaling" correction rule would generate the correction function $F(x)=(a/a_t)x$. A different correction rule—"shifting"—yields a correction function that shifts all the colors in the tile image as follows: $F(x)=x+(a-a_t)$. This scheme, shown schematically in the middle of Figure 6, will in general shift colors out of the range of reproducible colors. Other more sophisticated rules might employ gamma correction or a distortion of the color histogram [4], with improved effectiveness at the cost of additional complexity.

We have found that a combination of the "shifting" and "scaling" rules (shown schematically in Figure 6) is sufficient for our purposes and is easy to compute. Specifically, if we can use only a shift without sending any of the colors in the tile out of range, then we're done. If not, we shift as much as possible, then scale the resulting colors until the desired average is attained. So, let us return to the example where the desired average color a is less than the average color a_t of the image tile. Our "shift-and-scale" rule works as follows: if the minimum color m_t of the image tile is greater than a_t-a, then we use the shift rule above: $F(x)=x+(a-a_t)$; otherwise we use a combination of shifting and

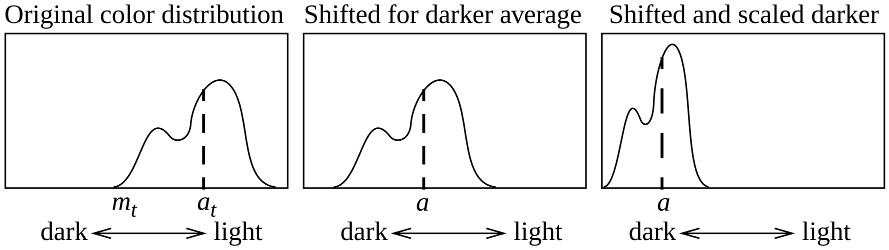
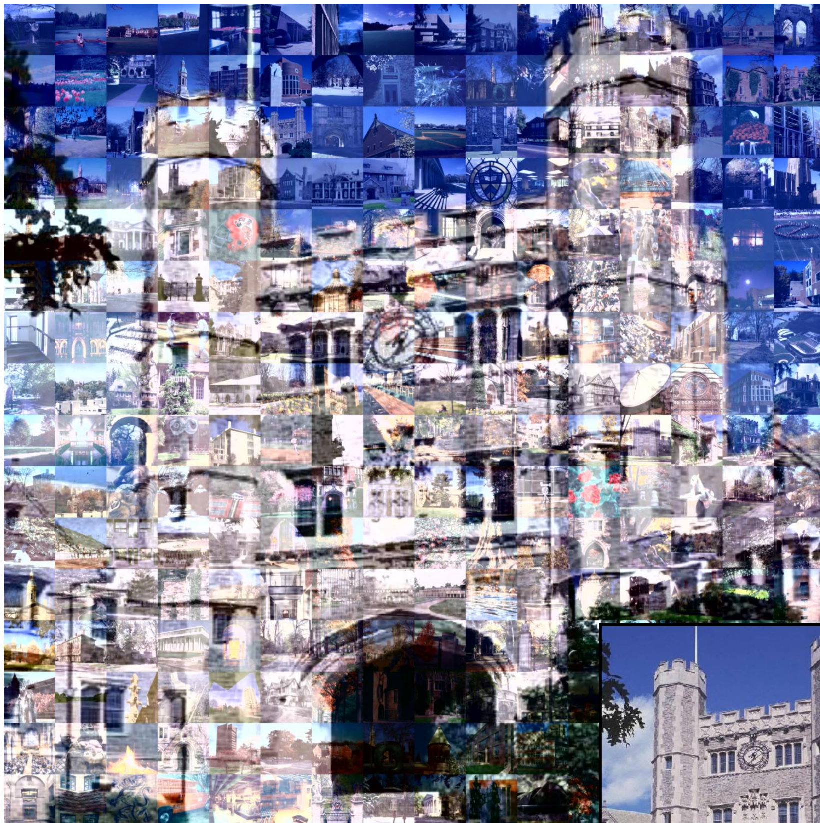


Fig. 6. Shifting and scaling colors to darken an image tile. The distribution is shown as a histogram. Dashed line indicates average.



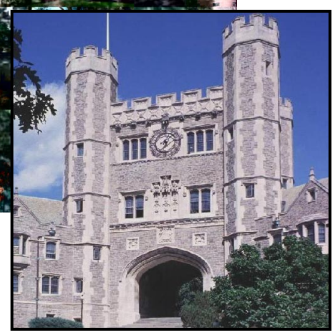


Fig. 7. Blair Arch composed of 250 scenes at Princeton University. Target image is shown on the lower right. (Color image)

scaling: $F(x)=a(x-m_t)/(a_t-m_t)$. There is a symmetric pair of cases when the desired average a is greater than the tile average a_t.

Now that we have a correction rule, we simply apply that rule for every pixel in every tile in the mosaic, supplying as the desired average color a the color of the corresponding pixel in the target image. Observe that in regions where the target image has a fairly constant color, the tile image will simply be shifted and scaled to achieve the target color. In regions where more complex shade variations occur, the tile image will vary similarly. This scheme generalizes the traditional halftoning scheme described in Section 3. If we use a repeated pattern of the dither matrix D for our tiling, then we can generate traditional halftoning using the rule: $F(x)$=black if $x<a$; $F(x)$=white otherwise.

If the tile images are larger than 16x16 pixels, then an efficiency improvement may be implemented by building a table once for each tile image. The table contains the two parameters of the correction function—how much to shift, and how much to scale—for every possible desired average a. Then, for every pixel p in the tile image, we use the corresponding pixel in the target image as our index a into the table, and then shift and scale p according to the entries in the table.

So far we have described correction only for grayscale. Because we are working with images, we can perform the same kind of correction independently in each of three color channels, just as traditional halftoning is often performed in each of three (or four) color separations for printing. We have found that the resulting colors tend to be slightly more vibrant if the calculations are performed in the YIQ color space rather than the RGB color space [4]. As the results of using different color spaces is subtle at best, we have not investigated using other color spaces such as HSV or LAB [4]. Figures 5, 7 and 8 were produced by screening in each of the Y, I and Q color channels. A different variation was employed in Figure 3, in which the Y channel (which specifies the brightness of the image) was corrected as described above, while the I and Q channels (containing all of the *color* information) of the resulting mosaic were simply copied from the tile images. This tends to emphasize the tile images in the resulting mosaic.

Fig. 8. Global currency (Color image)

5 Results

Figure 1 contains the only three-layer mosaic in this paper—an image of the face of Leonardo da Vinci's *Mona Lisa* composed of 100 smaller images of her face. Each of the smaller images in turn is composed of 100 tiny images of the face. By implication, the layering in these images could be infinite, reminiscent of fractal structure [8]. We have not as yet experimented with *varied* imagery that is more than two layers deep.

US President John F. Kennedy appears in Figure 2 composed of images of Marilyn Monroe. This image also appeared in the 1994 Xerox PARC Algorithmic Art Show. In the creation of this image, final arrangement of the tile images was performed by hand, although an algorithmic matching based on the shapes in the image tiles proposed an initial arrangement.

Grant Wood's painting *American Gothic* (1930) is composed in Figure 3 of pictures downloaded from the World Wide Web. The tile images were selected based on their colors and shapes from a collection of 20,000 images downloaded from all over the Web, using the image querying algorithm of Jacobs, *et al* [6].

In Figure 5, an abalone shell is shown screened by 16 microscopic images of abalone shells that were photographed using a scanning electron microscope. This figure demonstrates quite clearly the color correction process of Section 4.4, as the target image appears to burst through the tile images. In figures where the tiling is much more dense, this effect is more subtle.

In Figure 7, Princeton University's Blair Arch is shown composed of 250 scenes of the Princeton campus (in celebration of the University's 250th anniversary). The image tiles were arranged according to their shapes and colors as in Figure 3.

Finally, the globe shown in Figure 8 was created by texture-mapping [4] a sphere with a map of the earth. The map is actually an image mosaic composed of currency from around the world.

The color correction process we describe is not computationally expensive: it is linear in the number of pixels in the image mosaic. Each of the figures in this paper was produced on a desktop workstation in a few minutes or less.

Manually preparing the image tiles and the target image, as well as finding an arrangement for the tiles if it is done by hand tend to be the time-consuming stages in the overall process, particularly since they often are iterative.

6 Future work

This paper describes a method for creating image mosaics based on a target image and a collection of tile images. The project suggests a number of areas for future work, several of which are outlined here.

More complex tilings. As indicated in Sections 4.2 and 4.3, we use a fairly restrictive set of tilings: semiregular, rectangular grids. Perhaps we could find a more complex, irregular tiling that matches more carefully the tile images to a given target image. This problem combines both continuous optimization (for the exact placement of the tile boundaries as well as the scaling and cropping of tile images) and discrete optimization (over the arrangement of tile images within the tiling).

3D image mosaic. We intend to construct a gallery-sized image mosaic installation in which the image tiles are posters distributed in a seemingly-haphazard arrangement at varying heights and depths in the environment. However, from a *single* vantage point, the posters will visually align to form the large mosaic.

Video mosaics. Image mosaics may be extended in the time dimension to create video mosaics. In these mosaics, the tiles may change over time, while the target image remains constant. Alternately, the target could evolve while the image tiles remain constant. One challenge for video mosaics is that the resolution of typical video is substantially lower than that of printed media; perhaps multiresolution video [3] could alleviate the resolution problem.

References

[1] C. Close, J. Yau, *Recent Paintings*, Pace Wildenstein, 1995.

[2] S. Dali, *The Salvador Dali Museum Collection*, Bulfinch Press, 1996.

[3] A. Finkelstein, C.E. Jacobs, D.H. Salesin. Multiresolution Video. Proceedings of SIGGRAPH ‘96, *Computer Graphics,* Annual Conference Series, 1996, 281-290.

[4] J. Foley, A. van Dam, S.K. Feiner, J.F. Hughes, *Computer Graphics, Principles and Practice, Second Edition*, Addison-Wesley, Reading, Massachusetts, 1990.

[5] P.E. Haeberli. Paint by Numbers: Abstract Image Representations. Proceedings of SIGGRAPH ‘90, *Computer Graphics,* Annual Conference Series, 1990, 207-214.

[6] C.E. Jacobs, A. Finkelstein, D.H. Salesin. Fast Multiresolution Image Querying. Proceedings of SIGGRAPH ‘95, *Computer Graphics,* Annual Conference Series, 1995, 277-286.

[7] P. Litwinowicz. Processing Images and Video for an Impressionist Effect. Proceedings of SIGGRAPH ‘97, *Computer Graphics,* Annual Conference Series, 1997, 407-414.

[8] B.B. Mandelbrot, *The Fractal Geometry of Nature*, W. H. Freeman, New York, 1983.

[9] B. J. Meier. Painterly Rendering for Animation. Proceedings of SIGGRAPH ‘96, *Computer Graphics,* Annual Conference Series, 1996, 477-484.

[10] W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, G. Taubin. The QBIC Project: Querying Images By Content Using Color, Texture, and Shape. In *Storage and Retrieval for Image and Video Databases*, SPIE, 1993, 173-187.

[11] V. Ostromoukhov, R.D. Hersch. Artistic Screening. Proceedings of SIGGRAPH ‘95, *Computer Graphics,* Annual Conference Series, 1995, 219-228.

[12] S. Peleg, J. Herman. Panoramic Mosaics by Manifold Projection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1997, 338-343.

[13] R. Silvers, M. Hawley, *Photomosaics*, Henry Holt and Co., 1997.

[14] R. Ulichney, *Digital Halftoning*, MIT Press, Cambridge, Massachusetts, 1987.