A WEB APPLICATION FOR MAKING MOSAIC ART IMAGES

Tefen Lin, Jie Wang
Department of Computer Science, University of Massachusetts, Lowell, MA 01854, U.S.A.

Pei-Yu Huang, Yan-Ping Tung, Hsiao-Yun Huang, Kai-Wen Yu
Lunghwa University of Science and Technology, Taoyuan Country, Taiwan

Keywords: TMA, Mosaic art image, Dithering algorithm, FWD-FSD, Ajax, Web service, XQuery.

Abstract: This paper presents Tile Mosaic Art (TMA), an e-commerce Web application system for making mosaic art images from regular pictures. TMA makes it possible for ordinary users to produce high-quality mosaic art images with tiles without special training. We describe how TMA resolves color bandings caused by current technologies of tile industries, where only a limited color palette for tiles is available. In particular, TMA uses the FWD-FSD error diffusion algorithm (Lin and Wang, 2011) to maintain the original structure of the input image and produce high-quality mosaic images. TMA also develops Ajax, Web Service, and XQuery technologies to save and modify mosaic images and obtain the effect of what-you-see-is-what-you-get mosaic picture. Our experiments show that TMA not only produces high-quality mosaic images but also is efficient in terms of response and request time.

1 INTRODUCTION

Becoming an artist of mosaic art may be a dream of many people. Realizing this dream, however, would require extensive training. This paper presents a Web application system that allows ordinary people to become seasoned mosaic artists using tiles. Traditional mosaic artists create mosaic arts with small pieces of colored glass, stone, china, or other materials. Mosaic art today provides sophisticated decorations for floors, walls, and ceilings. Mosaic art images are composed manually by artists.

We design and develop an e-commerce Web application system called Tile Mosaic Art (TMA) to produce mosaic art images from regular pictures, where a mosaic art image consists of a grid of small square tiles of one square centimetre. In particular, the user uploads the original picture to TMA, which then converts it to a mosaic art image using a given color tablet. Users can obtain a what-you-see-is-what-you-get mosaic art image, modify it, and display it in their blogs and other places. Users may also choose to a sheet of grids for a mosaic image produced by TMA, where each cell has a unique number corresponding to a specific tile (see Figure 1 (c)), making it easy to piece the tiles together to form the image. TMA combines techniques of digital photographic image (as shown in Figure 1 (a)) and mosaic image (as shown in Figure 1 (b)) together to produce a mosaic image (as shown in Figure 1 (d)). The mosaic art can be framed and hung on the wall. The size of a typical frame is 50 cm wide, with its height determined by the width/height ratio of the original image.

Listed below are two major problems in creating a mosaic art application:

1. Confined by current tile making technology, there are only limited colors available for tiles. For example, the tile manufacturers we work with can only make tiles with 48 different colors.
2. Both of the user request and system response time must be reasonable to make the system user friendly.

Applying graphic dithering algorithms to enlarge original image is a common technique in the field of graphic design and graphic art. We devise a fast dithering algorithm to produce mosaic images using a given set of tiles with limited colors while maintaining high quality. The final output of our Web application is a sheet of grids (as shown in Figure 1 (c)) for a mosaic image, where each cell is appropriately numbered, which corresponds to a

Lin T., Wang J., Huang P., Tung Y., Huang H. and Yu K..
A WEB APPLICATION FOR MAKING MOSAIC ART IMAGES.
DOI: 10.5220/0003483706410644
In *Proceedings of the 7th International Conference on Web Information Systems and Technologies* (WSPA-2011), pages 641-644
ISBN: 978-989-8425-51-5

particular type of tile (as shown in Figure 1 (c)). Using this printout and a supply of tiles, everyone can assemble the tiles to produce, with little artistic talent, a nice-looking mosaic art image (shown in the Figure 5).

We use a number of Web technologies to shorten the request and response time in our system, including Web Service, DOM, XML, Ajax, and XQuery, in addition to the fast dithering algorithm we developed. In this paper we will describe how to convert the original image to a digital tile mosaic image and how to transfer each pixel in an image between the browser and the Web server.

The rest of the paper is organized as follows. In Section 2 we will describe how to produce a mosaic image with high quality from a given photo provided by the user. In Section 3 we will present our mosaic image editor using several Web technologies, which allows users to modify their mosaic images. In Section 4 we will explain the architecture of the TMA system. We will conclude the paper in Section 5. In Appendix we will provide a number of examples of mosaic art images.

(a) (b) (c) (d)

Figure 1: Our procedure of making a mosaic image: (a) Image uploaded by the user. (b) The mosaic image produced by our dithering algorithm. (c) A sheet of grids for putting tiles to produce a mosaic image. (d) The final mosaic art product.

2 MOSAIC IMAGE PROCESSING

Our Tile Mosaic Art Web system (TMA) consists of the following four major components:

1. Resize the original image.
2. Apply the FWD-FSD error diffusion dithering algorithm [LW11] to the resized image and produce a mosaic image with high quality.
3. Edit the mosaic image.
4. Print the sheet of grids for placing tiles.

2.1 Resize the Original Image

After the user uploads the orignial image, TMA first resizes it to fit in a standard picture frame (see Figure 1(d)). A typical picture frame is 50 cm wide with its height determined by the width/height ratio of the original image. In particular, TMA first resizes the original image of size $w \times h$ pixels to the desired size of w' pixels wide and $w'h/w$ pixels high, where w represents width and h represents height. For example, the original image of a little girl (see Figure 1) is 413 × 523 pixels . TMA resizes it by retrieving its own embedded thumbnail and scales it to the size of 50 × 63 pixels .

2.2 Color Mapping

After the original image is resized appropriatly, TMA analyzes the color of each piexl and maps it to the cloest color in the given pallate of limited tile colors. The tile manufacturer we work with can only produce tiles with 48.

TMA allows the following four image formats to be uploaded:GIF, JPG/JPEG, PNG,, and BMP. The GIF format supports up to 8 bits per pixel, allowing a single image to reference a palette of up to 256 distinct colors chosen from the 24-bit RGB color space. The JPEG files embed an ICC color profile (color space). Commonly used color profiles include sRGB and Adobe RGB of 24 bits, resulting in 16 million colors. The PNG and BMP formats support 32 bit colors, resulting in about 4 billion clolors. Every input image needs to reduce its colors to 48 colors. After color reduction, the resulting output image will lose image quality and detail information, and color banding will appear (see Figure 2 (b)). We use the FWB-FSD Error diffusion dithering algorithm (Lin and Wang, 2011) to achieve a nice visual quality of the image.

(a) (b) (c)

Figure 2: Size of these images: 50 pixels wide and 55 pixels high. (a) Input image. (b) The image with reduced colors using the limited color palette of 48 colors. (c) The image produced by the FWB-FSD dithering algorithm.

FWB-FSD is a better algorithm on mosaic art images and other types of images that require enlargement of pixels. For the sake of completeness, we will briefly describe in Section 2.3 the FWB-FSD dithering algorithm. After running FWB-FSD, the system produces a mosaic image as shown in Figure 2 (c).

2.3 FWB-FSD Dither

FWB stands for "Four-Way Block" and FSD stands for Floyd-Steinberg Dither. FSD is a effective dithering error diffusion algirithm. In an error diffusion algorithm, each pixel in the original image will disffuse its quatization error to its neigboring pixels. It was shown that FWB-FSD is a better algorithm on mosaic art images and other types of images that require enlargement of pixels (Lin and Wang, 2011). In particular, FWB divides the input image into blocks of an equal size with each block consisting of four sub-blocks such that the size of each sub-block is suitable for an underlying FSD error-diffusion algorithm. Scanning blocks from left to right and from top to bottom, for each block being scanned, FWB starts from the center of the block and diffuses errors along four directions on each sub-block.

The quantization error of each pixel, which is the difference between the RGB value of the pixel in the input image and the closest color in the limited color palette, will be distributed to its neighbour pixels by the coefficient error diffusion matrix when system is scanning the sub block. FWB uses the coefficient matrix developed from Floyd and Steinberg dithering (FSD). The FSD method diffuses a pixel's quantization error to its four neighbour pixels with the coefficient error diffusion matrix shown in (1), where x is the pixel being processing.

$$\frac{1}{16} \times \begin{bmatrix} 0 & 0 & 0 \\ 0 & x & 7 \\ 1 & 5 & 3 \end{bmatrix} \quad (1)$$

3 MOSAIC IMAGE EDITOR

We have developed a useful Web tool to allow users to modify mosaic images with the effect of "what you see is what you get" efficiently. This tool is based on AJAX, XML, Web Service and XQuery technologies.

AJAX (Asynchronous JavaScript and XML) is a powerful Web development mode for browser-based Web applications. Technologies that form the AJAX mode, such as JavaScript, HTTP, and XHTML, have each been widely used and well known. AJAX combines these technologies to allow Web pages to retrieve small amounts of data from the server without having to reload the entire page. This capability makes it possible to update Web pages between the server and the client smoothly and interactively (Lei and Duan, 2007).

For raster images, Languages like XHTML which does not have such a structural feature in their environment. Instead, XML provides image elements' means such as pixel and RGB for the incorporation of raster images in text-based files (Hur, Lee and Kim, 2006). In the TMA, we developed the XML (see Figure 4 (b)) to present mosaic image which is raster image and presented by XHTML to display the mosaic image. Converting a raster formatted image into XML enables the user to utilize the information residing in the image and select, read and manipulate the parts of the XML file or the file as a whole in a number of ways in accordance with the pixel and its color space (Antoniou and Tsoulos, 2006). This section elaborates on the technological environment, which can be utilized for the XML encoding of raster mosaic images and the transmitting information to the existing datasets using Ajax and web service methods. In the TMA system, mosaic images, formed by the XML (see Figure 4 (b)) and saved in the database, are useful for editing, researching, retrieving and copyright checking the copyright.

XQuery, an XML query language, a query language that uses the structure of XML intelligently can express queries across all these kinds of data, whether physically stored in XML or viewed as XML via middleware. The XQuery is designed to be broadly applicable across many types of XML data sources (W3C, 2010). Mosaic image editor developed the XQuery to access the mosaic image in the image database and return the XML data of mosaic image.

The mosaic image editor, one of application in the TMA system, allowing users to modify mosaic image in the browser (see Figure 3), retrieved the mosaic image which XML formatted in the database by the XQuery. In Figure 3, there are three images are displayed. Image (a) represents the current mosaic art image, formed as a table of XHTML and CSS that retrieves the RGB values from the image database. The user is allowed to choose the colors in the color palette (shown in Figure (c)) to modify each cell's color in Figure (a). Simultaneously, the icon in Figure (b) will be changed according to the modification in the biggest image to get the overall image. When the user saves the modified mosaic image, the system will send the modified coordinate values and the RGB values to the server. AJAX and Web Service are used to store the image into database.

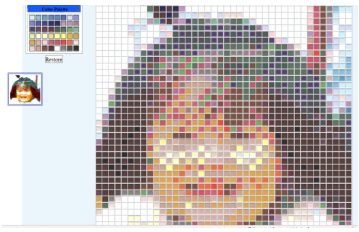
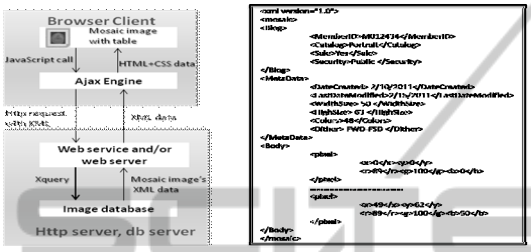


Figure 3: (a) The largest image, which is the current mosaic art in the main page. (b): The smaller portrait icon. (c) The last one is the color palette.



(a) (b)

Figure 4: (a) The architecture of mosaic image editor; (b) XML data in (a).

4 IMPLEMENTATION AND TESTING

After mosaic image process, the mosaic image is saved both the bmp format and formed XML data in the database. The amount of disk storage required for any bitmap is easy to calculate, given the bitmap dimensions (N × M) and colour depth in bits (B). The formula for the file size in KBytes is N × M × B / 8 × 1024. We set up the mosaic image width size 50 cm and high with the ration of original image. The size of a mosaic image should be less than 19 Kbytes (we use the image 50 × 100 and color depth is 32 bits). By the way, the TMA system was built on an Hp Proliant ML150 G5 Hot Plug Server with Dual-Core Intel Xeon Processor E5502 and 16 GB memory using an SQL Server 2008 database system. The C# programming language in the .NET framework was used to implement the system. As an experimental, the response and request time of mosaic image is fast, it means mosaic image is very easy to transmit.

5 CONCLUSIONS

This paper presents design and implementation of a feasible mosaic image solution. It uses an effective dithering algorithm and these technologies of Ajax, Web services and XQuery. TMA system provides mosaic image developers with sense to create their own mosaic art used by tiles. In order to let user to modify their own mosaic image, we used the Ajax, web service and XQuery to transmit the modified RGB values only portion of image data will be sent to web service and modify the mosaic image saved in the database. Total solution is feasible. Author not only can get high quality of mosaic image, but also fasten to finish the creation of mosaic image. We present two examples of mosaic art images using FWB-FSD (see Figure 5).

REFERENCES

R. Floyd and L. Steinberg, 1975. *An adaptive algorithm for spatial grey scale*, SID Intl. Svmp. Dig. Tech. Papers VI, 36—37.

Terfen Lin and Jie Wang, 2011. *Producing Automated Mosaic Art Images of High Quality with Restricted and Limited color palettes,* SciTePress

Byron Antoniou and Lysandros Tsoulos, 2006, *The potential of XML encoding in geomatics converting raster images to XML and SVG,* Computers & Geosciences Volume 32, Issue 2, March 2006, Pages 184-194.

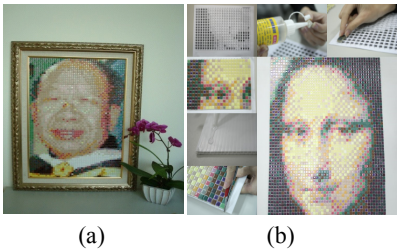
Lihui Lei and Zhenhua Duan, 2007, *Integrating AJAX and Web Services for Cooperative Image Editing,* IEEE Electronic Library Vol.9, Iss.3; p.29-29

Wonchang Hur, Jaebum Lee and C. Young Kim, 2006, *Web-based Diagnostic Imaging Service Using XML Forms*, Journal of Digital Imaging Volume 19, Number 4, 328-335

APPENDIX

Demonstration of Mosaic Art and e-Commerce Process

We present two examples of mosaic art images using FWB-FSD (see Figure 5).



(a) (b)

Figure 5: (a) A retired professor. (b) The process of piecing tiles together to create the mosaic art.