RESEARCH PAPER

Engineering

Volume : 3 | Issue : 6 | June 2013 | ISSN - 2249-555X

Secret-Key-Generation for Mosaic Image

KEYWORDS

secret-fragment-visible mosaic image, computer art, information hiding, greedy search, covert communication.

K.G.S Venkatesan

Associate Professor, Department Of C.S.E, Bharath University – selaiyur, Chennai – 600073, Tamil Nadu

D. Priya

Final Year – M.Tech (C.S.E), Department Of C.S.E, Bharath University - selaiyur, Chennai – 600073, Tamil Nadu

ABSTRACT *Based on the "SECRET-KEY-GENERATION FOR MOSAIC IMAGE" paper for information hiding, a new type of computer art image is implemented, which is created automatically by composing small fragments of a given image to become a target image in a mosaic form, achieving an effect of embedding the given image visibly but secretly in the resulting mosaic image. This effect of information hiding is useful for communication or secure keeping of secret images. To create a mosaic image of this type from a given secret color image, the 3-D color space is transformed into a new 1-D color scale, based on which a new image similarity measure is proposed for selecting from a database a target image that is the most similar to the given secret image. A fast greedy search algorithm is being used to find a similar tile image in the secret image to fit into each block in the target image. The information of the tile image fitting sequence is embedded into randomly-selected pixels in the created mosaic image by a lossless LSB replacement scheme using a secret key; without the key, the secret image cannot be recovered. The experimented method, originally designed for dealing with color images, is also extended to create grayscale mosaic images which are useful for hiding text-type grayscale document images. An additional measure to enhance the embedded data security is also implemented.*

INTRODUCTION

MOSAIC is a type of artwork created by composing small piece of materials, such as stone, glass, tile, etc. Invented in ancient time, they are still used in many applications today. Creation of mosaic images by computer is a new search direction in recent years. Many methods have been proposed to create different types of mosaic images by computer. A new computer art image called secret-fragment- visible mosaic image is implemented, which is created automatically by composing small fragments of a given image to become a target image in a mosaic form, achieving an effect of embedding the given image visibly but secretly in the resulting mosaic image. The source image may be said to be secretly embedded in the resulting mosaic image, though the fragment pieces are all visible to the observer. And this is the reason why the resulting mosaic image is named secret-fragment-visible. This is a new technique of information hiding, not found in literature so far.

A secret image is divided into rectangular-shaped fragments, called tile images, which are fitted next into a target image selected from a database to create a mosaic image. The number of usable tile images for this operation is limited by the size of the secret image and that of the tile images. This is not the case in traditional mosaic image creation where available tile images for use essentially are unlimited in number because the tile images are not generated from the secret image and may be used repeatedly. Then, the information of the tile-image fitting is embedded into some blocks of the mosaic image, which are selected randomly by a secret key. Accordingly, an observer possessing the key can reconstruct the secret image by retrieving the embedded information, while a hacker without the key cannot.

More specifically, a secret image is first divided into rectangular-shaped fragments, called tileimages, which are fitted next into a target image selected from a database to create a mosaic image. The number of usable tile images for this operation is limited by the size of the secret image and that of the tile images. This is not the case in traditional mosaic image creation where available tile images for use essentially are unlimited in number because the tile images are not generated from the secret image and may be used repeatedly. Then, the information of tile-image fitting is embedded into some blocks of the mosaic image, which are selected randomly by a secret key. Accordingly, an observer possessing the key can reconstruct the secret image by retrieving the embedded information, while a hacker without the key cannot.

GRAYSCALE FEATURES OF BLOCKS AND MOSAIC IMAGE CREATION AND RECOVERY

The Gray scale image is obtained through various ways like scanning, from paper documents mainly with text contents. In this case, the selected target image obviously should be of the same type, namely, a grayscale image; and the generated mosaic image is also a grayscale one. First, the color image database should be converted into a grayscale version. For this, the color values (r,b,g) of every pixel in each image in the database is transformed in this study into a 1-D grayscale value Y by the equation $Y=0.177*r+0.813*g+0.011*b$ where the weights for r,g and b are taken to be the coefficients of the luminance (the component) used in the transformation from the RGB model to the YUV one.

The reason for adopting such weights instead of the conventional value of 1/3 for each color channel is based again on the previously-mentioned human eye's higher sensitivity to the green color. Then, the average of the grayscale values of all the pixels in each image block is computed as a feature, called the Y-feature. This feature is used further as a measure like described previously in the database construction process to compose the Y-feature histogram of each candidate target image D in the database.

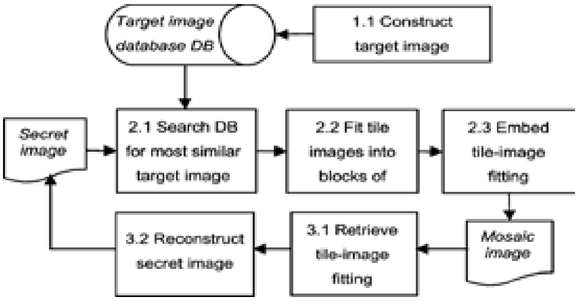
Target image database DB
1.1 Construct target image
Secret image
2.1 Search DB for most similar target image
2.2 Fit tile images into blocks of
2.3 Embed tile-image fitting
Mosaic image
3.1 Retrieve tile-image fitting
3.2 Reconstruct secret image

Figure 3. Processes for secret-fragment-visible mosaic image creation and secret image recovery.

Secret Image Selection

A Secret image is selected from the system and is uploaded to the database. And this secret image is divided into 9 fragments. The color transformation function, h and y is applied to each of the fragments of the image and is saved in the database.

The color transformation function is implemented using the following expression:

h (r', g', b') = b'+Nb*r'+Nb*Nr*g'

Where, the numbers of levels, N_r, N_g and N_b , are all set to be 8, and the largest weight, namely, the value N_b*$N_{r,}$ is assigned to the green channel value g' and the smallest weight, the value 1, is assigned to the blue channel value b'. This way of weight assignment is based on the fact that the human eye is the most sensitive to the green color and the least sensitive to the blue one, leading to a larger emphasis on the intensity of the resulting mosaic image.

The above function is for color images and for gray images the transformation function is as follows:

Y=0.177*r+0.813*g+0.011*b

where the weights for r,g and b are taken to be the coefficients of the luminance (the component) used in the transformation from the RGB model to the YUV one. The reason for adopting such weights instead of the conventional value of 1/3 for each color channel is based again on the previously-mentioned human eye's higher sensitivity to the green color. Then, the average of the grayscale values of all the pixels in each image block is computed as a feature, called the Y-feature

Mosaic Image Creation

Selecting the most similar target image:

First calculate the average of the h value for the secret and the target image for both the color and the gray image. Compare the average of the secret and the target image, and select the most similar target image D from DB.

<u>Image similarity measure:</u>(color)

m(S, D)= 1/ ∑h=0 to 512 | Hs(h)- Hd(h) |

Where,

s=secret image; d=target image

Hs (h) = h value of secret image.

Hd (h) =h value of target image.

h= h-feature value

<u>Image similarity measure:</u>(gray)

m(S,D)= 1/∑y=0 to 255 | Hs(y)-Hd(y) | Where,

y=y-feature

s=secret image; d=target image.

Hs(y)=y value of secret gray image.

Hd(y)=y value of target gray image.

Fitting tile images into target blocks:

Calculate the h-feature values of all the tile images from the secret image and take out the h-feature values of all the target blocks of D_o from DB.

In a raster-scan order of the target blocks in $D_{o,}$ perform the greedy search process to find the most similar tile images $s_{1,}s_{2,\ldots}s_9$ in S and corresponding to the N target blocks d_1,d_2,…d_9 in D_o , respectively, to construct the secret recovery sequence L_R=0,1,..9 Using the h-feature values. And finally, fit the tile images $s_{1,}$ s2 $_{\ldots}s_9$ into the corresponding target blocks d_1, d2…d_9 respectively, to generate a preliminary secret-fragment-visible mosaic image U.

Key Generation:

A secret key is generated randomly in each of the fragments of the images using a random class. This secret key is used for recovering the secret image from the mosaic image. Without this key secret image cannot be recovered.

Embedding tile-image fitting information:

Concatenate the data of recovery key for the secret image S and transform the concatenation result into a binary string, and embed it into the first ten pixels of the first block of mosaic image U in a raster-scan order by the lossless LSB replacement scheme. Take the final Mosaic image U with L_R embedded as the desired secret-fragment-visible mosaic image for the input secret image S.

To record the mappings of embedding process a sequence , called the secret recovery sequence is used and embed into randomly-selected blocks in the created mosaic image using a technique of lossless least-significant-bit (LSB) replacement. In more detail, to get the mappings, we start from the top leftmost target block d1 in the selected target image Do , and find for it the most similar tile image in the secret image Si , and form the first mapping Si-> d1 to be included in Lr . Next, in a raster-scan order, we process the target block d2 to the right of d1 to find the most similar tile image Sj in the remaining tile images to form the second mapping sj->d2 for Lr . Then, we do similarly to find the third mapping Sk->d3, and so on. We continue this greedy search process until the last target block at the bottom-rightmost corner in the target image is processed. The resulting Lr may be regarded to include two block-index sequences L1=I,j,k.., and L2=1,2,3… with mappings i->1,j->2,k->3, and so on. Since L2 is a well-ordered sequence of 1,2,3, we can ignore it and take Lr to include just L1 to reduce the data volume of to be embedded.

The number Nr of bits required to represent the secret recovery sequence Lr is as follows:

N=Ws * Hs / Zt
Nx=[log2N] + 1
Nr=N *Nx
Where N= number of tile images in S.

Nx=number of bits to specify the index of a tile image.

Ws, Hs=width and height of the secret image.

Secret Image Recovery

Retrieving tile-image fitting information:

Retrieve the recovery key of the tile images from the first ten pixels in the first block of image in a raster-scan order using a reverse version of the lossless LSB replacement scheme. Repetitively select randomly an unselected block other than the first block from using the random number generator with the secret key as the seed, extract bits from all the pixels of using a reverse version of the lossless LSB replacement scheme proposed. Transform every bits of L_R into an integer which specifies the index of a tile image in the original secret image (to be composed), resulting in the secret recovery sequence $L_{R=}$ 0, 1…9.

Reconstructing the secret image:
Construct the mappings of the indices of the tile images of the original secret image S (to be composed next) to those of the corresponding target blocks of U. For example, U as 0=1 ,1=2,2=4,3=6,4=0,5=3,6=5,7=6,8=7,9=8.

Compose the tile images of the desired secret image S in a raster-scan order according to the N mappings by taking block 1 of U to be tile image 0=1 in S , block 2 of U to be tile image 1=2 in S , and so on, until all blocks of are fitted into S. Thus, the secret image is recovered without any loss.

A new type of digital art, called secret-fragment-visible mosaic image, has been proposed, which can be used for secure keeping or covert communication of secret images. This type of mosaic image is composed of small fragments of an input secret image; and though all the fragments of the secret image can be seen clearly, they are so tiny in size and so random in position that people cannot figure out what the source image looks like. Specifically, a new color scale and a new grayscale have been proposed to define a new h-feature and a new y-feature, which are then used to define appropriate similarity measure for images and blocks for generating secret-fragment-visible mosaic images more effectively.

A greedy search algorithm has also been proposed for searching the tile images in a secret image for the most similar ones to fit the target blocks of a selected target image more efficiently. Tile-image fitting information for secret image recovery is embedded into randomly selected tile images in the resulting mosaic image controlled by a secret key. An additional security enhancement measure was also proposed. The method was extended to generate grayscale mosaic images with grayscale secret images as input. Good experimental results have been shown to prove the feasibility of the proposed method.

FUTURE SCOPE
Good mosaic image creation results are guaranteed only when the database is large in size so that the selected target image can be sufficiently similar to the input secret image. Future works may be directed to allowing users to select target images from a smaller-sized database or even without using a database, as well as to developing more information hiding applications using the proposed secret-fragment-visible mosaic images. Furthermore, an additional secret key can be used to prevent hackers from trying to extract the secret key so, without the help of the second key; the original bit pattern cannot be recovered. Even if a hacker's random trial leads to correct extraction, the extracted index will be still in the form of random bit pattern.

REFERENCE 1. A. Hausner, "Simulating decorative mosaics," in Proc. SIGGRAPH, Los Angeles, CA, Aug. 2001, pp. 573–580. | 2. G. Elber and G. Wolberg, "Rendering traditional mosaics," Vis. Comput., vol. 19, pp. 67–78, 2003. | 3. P. Haeberli, "Paint by numbers: Abstract image representations," in Proc. SIGGRAPH, Dallas, TX, 1990, pp. 207–214. | 4. R. Silver and M. Hawley, Photomosaics. New York: Henry Holt, 1997. | 5. S. Battiato, G. Di Blasi, G. M. Farinella, and G. Gallo, "Digital mosaic framework: An overview," Eurograph.—Comput. Graph. Forum, vol. 26, no. 4, pp. 794–812, Dec. 2007. | 6. Y. Dobashi, T.Haga, H. Johan, and T. Nishita, "A method for creating mosaic image using voronoi diagrams," in Proc. Eurographics, Saarbrucken, Germany, Sep. 2002, pp. 341–348. |