Methods for Photomosaic Generation Based on Different Image Similarity and Division Strategies

Daqing Chang[1], Changshui Zhang[1], and Shifeng Weng[2,⋆]

[1] State Key Laboratory on Intelligent Technology and Systems,
Tsinghua National Laboratory for Information Science and Technology (TNList),
Department of Automation, Tsinghua University, Beijing, 100084, P.R. China
[2] School of Electronics and Information,
Zhejiang Wanli University, Zhejiang, 315000, P.R. China

Abstract. Photomosaic is an image that is composed of small images called tiles. In our work, different measurements of similarity between tiles in the desired original image and alternative tiles in the database are used in the generation of photomosaic after classifying tiles in the original image into tiles of purity and edge. And a heuristic position searching strategy is proposed to keep the original length-width ratio of each tile in the photomosaic generated unchanged if tiles in the database have different sizes.

Keywords: photomosaic, image classification, image similarity.

1 Introduction

Photomosaic is a kind of mosaic techniques [1] and a photomosaic is an image constituted by small images selected in a database of small images called tiles, as shown in Fig 1. When viewed at a distance, the details of each tile vanish and we will see the desired original image. Elements that affect the effectiveness of algorithms for generating photomosaic is studied in [2]. The impact of the similarity between tiles, granularity of tiles in the original image and the variety of tiles in the database is compared.

The technique to generate a photomosaic with computer is first studied by Robert Silver[3]. In his method,the original image is firstly divided with grid into tiles of the same size. Then photomosaic is generated by replacing each tile with the most similar tile alternative in the database.

From the procedure above, we can see that for each tile to be matched, every tile in the database is checked whether it is the most similar one which is time

⋆ This work is supported by NSFC (No. 61021063), 973 Program (2009CB320602), Beijing Municipal Education Commission Science and Technology Development Plan key project under (No. KZ201210005007) and Zhejiang Provincial Natural Science Foundation of China (No. Y1110661).

C. Sun et al. (Eds.): IScIDE 2013, LNCS 8261, pp. 682–690, 2013.

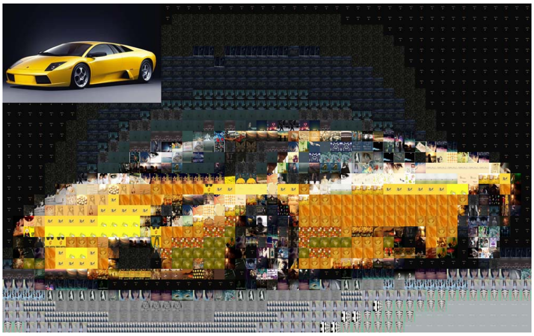


Fig. 1. An example of photomosaic generated by our method.The original image is scaled on the top left.

consuming if the capacity of database is big. G.D.Blasi uses a data structure of antipole tree to speed up this procedure[4].

Essentially, the process of generating photomosaic can be taken as an optimization problem, variables of which are indexes of tiles chosen at each position in the original image and the objective function to be minimized is the sum of matching error. Methods of evolutionary computing such as genetic algorithm(GA) are applied to solve the problem [5][6][7].

However, generally among algorithms exist, a uniform measurement of similarity is used to find the most similar tile in the database for each tile in the original image. It means that for each tile features used in matching are the same. However, the criterion we use when we judge with our eyes whether a tile is similar to another may vary between different tiles. For example, when matching a tile which color changes little, perhaps the most important factor we consider is whether their main color is the same. While if a tile is part of the main outline of the original image, the similarity of shape may be considered as well as the factor of color. Therefore, in our method, tiles in the original image are classified and different measurements of similarity are used for each class. And results we get usually have a good visual effect.

Furthermore, we can see that each tile in the photomosaic generated is usually scaled to the same size of the grid used to divided the original image in existing methods. Nevertheless the length-width ratio of each tile in the database may differ from the grid and the deformation may affect the content of the tile to some extent. To avoid this,we try to describe a strategy of division to keep the length-width ratio of tiles used in the photomosaic unchanged.

The rest of this paper is organized as follows. In section 2, we describe the measurement of similarity we used. Our method for photomosaic generation with classification of tiles in the original image is described in section 3. The method that keep the length-width ratio of each tile unchanged is shown in section 4. And we conclude our work in section 5.

2 Measurement of Similarity

The procedure of generating a photomosaic can been simplified as finding the most similar tile in the database to replace each tile in the original image. The importance of the measurement of similarity during this procedure is obvious. However, different kinds of similarity can be defined from different views. Intuitively, if the color distribution of two tiles is similar at corresponding position, we may probably feel that they are similar. And if the two tiles have similar shapes which means similar edge distribution, they can also been think as similar. Besides, the content of tiles can also be taken as a factor of similarity. Therefore, both the distribution of color and edge are considered in our measurement of similarity. However, as each tile in the original image is only a small part of the whole image which usually has no meaningful content, the content of the tile is not included.

2.1 Color Similarity

Let P_1 be a tile in the original image and P_2 be a tile in the database. Then color Similarity between P_1 and P_2 is formulated in (1).

$$D_{col} = \sum_{R,G,B} \sum_{i=1}^{m} \sum_{j=1}^{n} (A_1(i,j) - A_2(i,j))^2 \tag{1}$$

As there are RGB channels for a color image, the color similarity is D_{col} defined as the sum of L-2 norm distance of each channel and $A_1(i,j)$ and $A_2(i,j)$ represent the pixel value of corresponding position in each channel. Besides, as tiles in the original image and tiles in the database are not have to be of the same size. Both tiles being matched are resized to an equal size $m \times n$ before matching.

2.2 Edge Similarity

Just like the color similarity above, the edge similarity between P_1 and P_2 is formulated in (2)

$$D_{edge} = \sum_{i=1}^{m} \sum_{j=1}^{n} (E_1(i,j) - E_2(i,j))^2 \tag{2}$$

In this formulation, D_{edge} is the edge similarity we get. E_1 is the edge strength of P_1 and E_2 is the edge strength of P_2 computed by (3) where E_h and E_v represent the horizontal gradient and vertical gradient which are the responses of mask M_1 and M_2 in (4)

$$E(i,j) = \sqrt{E_h(i,j)^2 + E_v(i,j)^2} \tag{3}$$

$$M_1 = \begin{bmatrix} -1\ 0\ 1 \\ -2\ 0\ 2 \\ -1\ 0\ 1 \end{bmatrix} M_2 = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \tag{4}$$

We can see from above that the measurement of similarity can be seen as some distance between different tiles. The smaller the distance is, the more similar the two tiles compared are.

3 Generation with Classification Of Tiles

As mentioned in section 1, tiles in the original image may have different types. For example,a tile which is part of sky in the original image may have no notable edge and a tile with the main color blue could be taken as a similar tile in the database while for a tile on the edge of some object, we prefer a similar tile with similar edge as well as similar color distribution.Therefore, to improve the visual effect of photomosaic, classification of tiles in the original image and apply different measurements of similarity to each class is used here.We propose a method to generate photomosaic with classification of tiles in the original image in this section.

3.1 Classification of Tiles in the Original Image

In section 2, we describe the measurement of similarity with color and edge considered. Just as the discussion at the beginning of this section, when matching a tile in the original image, we may focus on the factor of color for a tile smoothly changed, and perhaps both factors of color and edge are considered for a tile remarkably changed.So the classifier is trained according to the degree of change. The features we use in training are histogram, Tamura contrast and variance. As there are RGB channels for a color tile, in fact these three features are extracted in each channel and a nine-dimension vector of feature will be used for each tile in the training set.

Histogram. Histogram of each channel represents the color distribution of this tile. The denser the histogram is,the more likely that this tile is smoothly changed. In our method, the density of histogram is formulated as (5)

$$D_{hist} = \frac{\sum_{i=a}^{b} H(i)}{\sum_{j=1}^{N} H(j)} \tag{5}$$

where

$$a = \max(1, i_{peak} - 0.1 \times N)$$
$$b = \min(N, i_{peak} + 0.1 \times N)$$

$H(i)$ is the value of the i-th grayscale level in histogram of some channel. i_{peak} is the index with maximum value in $H(i)$. N is the number of grayscale level.

Variance. Since variance reflects the divergency of a group of data. We take it as a measurement of changing degree too. The variance is computed by (6)

$$\sigma^2 = \sum_{i=1}^{N} (i - \mu)^2 h(i) \tag{6}$$

where μ is the mean value of this channel and $h(i)$ is the i-th grayscale level in histogram normalized by the its maximum value.

Tamura Contrast. Contrast measures the change of brightness which is also an aspect of changing degree in a tile. In our method we exploit the definition of contrast in [8] which is formulated in (7).

$$F_{con} = \frac{\sigma}{(\alpha_4)^{1/4}} \qquad \alpha_4 = \frac{\mu_4}{\sigma^4} \tag{7}$$

where F_{con} is the contrast called Tamura contrast. σ is the standard deviation which is the root of variance and μ_4 is the fourth-order moment of this channel defined as

$$\mu_4 = \sum_{i=1}^{N} (i - \mu)^4 h(i)$$

We randomly pick out thousands of tiles divided from thousands of original images randomly chosen as the training set. Then, a clustering algorithm of K-means is applied to label the training set. After that the classifier is trained by SVM with this labeled data.

3.2 Tile Similarity

In the last part, we have trained the classifier that used to judge whether a tile belongs to a tile of purity that means smoothly changed or to a tile of edge that means remarkably changed. Let P_1 be a tile in the original image and P_2 be a tile in the database. Then the similarity between P_1 and P_2 is formulated in (8)

$$D = \begin{cases} D_{col} & P_1 \in purity \\ (1-\alpha)D_{col} + \alpha D_{edge} & P_1 \in edge \end{cases} \tag{8}$$

where D is the measurement of similarity. D_{col} and D_{edge} are the similarity of color and edge described in section 2. α is the weight that make balance of the two parts. Generally, we take the value of α as the L-2 norm of the edge strength of P_1 defined in (3) normalized by the maximum value that edge strength may take at a position. We know from this that for a tile belonging to tile of edge, the weight of edge similarity increases with the grow of edge strength which is obviously reasonable.

3.3 Algorithm

After defining the similarity between tiles, we describe the algorithm of our method here and Fig 2 shows some results of our method.

1 Divide the original image into tiles of the same size with grid.
2 classify each of the tiles into tile of purity and tile of edge with classifier trained before.
3 For each tile in the original image,compute the similarity with each tile in the database and take the one with the smallest value as the most similar tile in the database.
4 Replace the tiles in the original image with tiles found in the last step to achieve the photomosaic.

4 Generation without Equal Division

It is easy to see that a common characteristic of "traditional" photomosaic is that every tile in the photomosaic has the same size. However, tiles in the database perhaps have different sizes when we get them. Therefore, the length-width ratio of them maybe changed when we resize them to the fixed size used in photomosaic which will affect the content of them to some extent.Because of this, we describe a method to generate photomosaic assembled with tiles of variant sizes in this section.

4.1 Position Searching Strategy

As we want tiles appeared in the photomosaic to keep the original length-width ratio unchanged, the position of tiles cannot be decided in advance like method in the last section by dividing the original image into equal tiles with gird. Therefore, we should decide the position that the next tile to be put at before finding the most similar tile in the database.

The bad thing that comes with the variety of sizes and arbitrariness of position is overlap. We can see that the main situation that overlap may probably be produced is that we have to fill narrow gaps or small regions unmatched with tiles which are bigger than them. Hence, our target is to decrease the generation of narrow gaps and small regions unmatched. Under this criterion, our strategy is that each time we take one of the corners of region that has not been matched as the position where we put the next tile in the database. This procedure is explained in Fig 3.

4.2 Tile Similarity

After the position of the next tile to be placed at is determined, let P_1 be the tile in the original image with the same size of the tile to be measured in the database

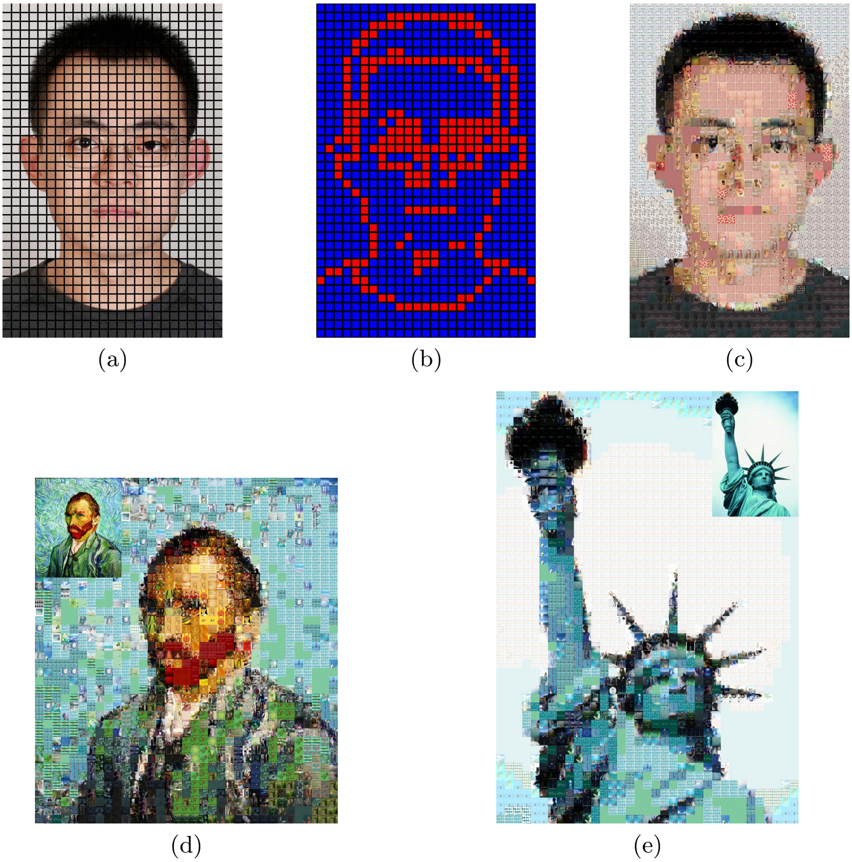


Fig. 2. Results of our method in section 3. In (a), The original image is divided in to 38×25 tiles. (b)is the result of classification that red blocks mean tiles of edge and blue blocks mean tiles of purity. (c) is the photomosaic generated.The other two results are shown in (d) and (e). The capacity of database is 5000.

at the chosen position and P_2 be the tile in the database. The measurement of similarity between P_1 and P_2 is shown in (9).

$$D = D_{col} + D_{var} + D_{shape} \tag{9}$$

where D represents the similarity we get, D_{col} is the color similarity defined in section 2. D_{var} is the sum of variances in each channel of P_1. If P_1 is complex and sharply changed, this term is big and a smaller tile will be chosen, as the smaller the tile is, the less details there are in this tile which means the variance is relatively small. Otherwise, if P_1 is smoothly changed,this term is small and replacement with a bigger tile is permitted . D_{shape} is proportion of overlap with region that has been matched and the area out of the border of the original image in the whole area of P_1 which is a penalty for overlap.The smaller this term is, the less overlap happens.

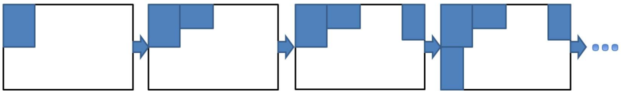
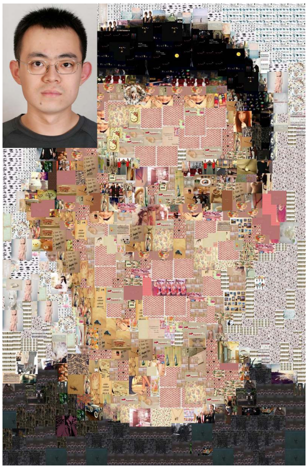


Fig. 3. An explain of position searching strategy,each tile is put at one of the corners in the unmatched region

4.3 Algorithm

At last of this section, we describe the complete algorithm as follows and results of our algorithm are shown in Fig 4:

1 Find the next position to be matched next according to the position searching strategy.
2 Compute the similarity between the tile at the chosen position in last step and tiles in the database and replace the tile in the original image with the most similar tile in the database.
3 Repeat the first two steps until all regions in the original image are replaced.



(a)

(b)

Fig. 4. Results of method in section 4. 1006 tiles are used in (a) and 839 tiles are used in (b). The capacity of database is 5000.

5 Conclusion

In our work, we propose a method for photomosaic generation with classification of tiles in the original image and different measurements of similarity are applied to find the best tile in the database. Besides, we also describe a method to generate photomosaic with tiles of arbitrary sizes. However, we don't classify tiles in the original image here as in the first case. This is because the features we use to train the classifier don't have the character of invariance with the change of sizes and how to deal with the problem is one of the work we need to do in the future.

Furthermore, another thing we will do next is reducing the repeat of a single tile in the photomosaic. If we restrict the maximum times that a single tile can be used in the photomosaic,we can't get the best photomosaic of the minimum sum of error by matching each tile line-by-line, which is more evident if the capacity of the database is not big enough.

Also, how to speed up the procedure of generation will be considered in our future work.

References

1. Battiato, S., Di Blasi, G., Farinella, G.M., Gallo, G.: Digital mosaic frameworks - an overview. Computer Graphics Forum 26(4), 794–812 (2007)
2. Tran, N.: Generating photomosaics: an empirical study. In: Proceedings of the 1999 ACM Symposium on Applied Computing, SAC 1999, pp. 105–109. ACM, New York (1999)
3. Silvers, R.: Photomosaics. Henry Holt and Co., Inc., New York (1997)
4. Di Blasi, G., Petralia, M.: Fast photomosaic (2009)
5. Mat Sah, S.B., Ciesielski, V., D'Souza, D., Berry, M.: Comparison between genetic algorithm and genetic programming performance for photomosaic generation. In: Li, X., Kirley, M., Zhang, M., Green, D., Ciesielski, V., Abbass, H.A., Michalewicz, Z., Hendtlass, T., Deb, K., Tan, K.C., Branke, J., Shi, Y. (eds.) SEAL 2008. LNCS, vol. 5361, pp. 259–268. Springer, Heidelberg (2008)
6. Mat Sah, S.B., Ciesielski, V., D'Souza, D.: Refinement techniques for animated evolutionary photomosaics using limited tile collections. In: Di Chio, C., Brabazon, A., Di Caro, G.A., Ebner, M., Farooq, M., Fink, A., Grahl, J., Greenfield, G., Machado, P., O'Neill, M., Tarantino, E., Urquhart, N. (eds.) EvoApplications 2010, Part II. LNCS, vol. 6025, pp. 281–290. Springer, Heidelberg (2010)
7. Narasimhan, H., Satheesh, S.: A randomized iterative improvement algorithm for photomosaic generation. In: World Congress on Nature Biologically Inspired Computing, NaBIC 2009, pp. 777–781 (December 2009)
8. Tamura, H., Mori, S., Yamawaki, T.: Textural features corresponding to visual perception. IEEE Transactions on Systems, Man and Cybernetics 8(6), 460–473 (1978)