

Region-Based Retrieval and Manipulation of Image Collections

David GAVILAN[†] Suguru SAITO[†] Masayuki NAKAJIMA^{†‡}

[†]Graduate School of Information Science & Engineering, *Tokyo Institute of Technology*

[‡] *National Institute of Informatics*

1 Introduction

Creating image collections is easier and cheaper every day. Photo organizers let us sort pictures using comments, date, or keywords, but these information or metadata requires manual input. Furthermore, editing pictures is usually limited to global image filters and further processing requires external programs. For doing a simple composite or a collage, for instance, we need to manually search for images with appropriate content, and then use external cropping and compositing tools. In this work we present a tool that starts directly from the result, a collage depicted by a simple hand-drawn sketch, and that automatically searches appropriate image parts in an image database for creating the final composite.

2 Related work

Di Sciascio et al. proposed a query-by-sketch system to access image collections [5]. Every image is represented by a set of regions, their local properties, and the spatial relation between them. In our work, their graph-like representation is simplified by means of histograms representing each local property, since our main focus is retrieving individual regions.

The purpose of the retrieval in our application is creating an artistic collage. Systems like AutoCollage [4] automatically create collages retrieving images and selecting regions of interest given some predefined heuristic rules, such as the appearance of faces in the picture. In our system the user can interactively edit and explore the collage.

3 System overview

The proposed system has three main subsystems; the segmentation module, the retrieval system, and the compositing module. Photographs are segmented in a few regions using morphology and color categorization. Then, images are stored in a database, storing separate properties for each region, used for retrieval [1]. The user draws a sketch with simple strokes that are used to retrieve particular segments that are then used in the composite module. The retrieval and compositing are interactive, so the user can select different images and create several arrangements in the composite.

Figure 1 shows the application interface with an example of retrieval and composite. The user paints with a color palette of a few bins of basic color categories, and with each bins several variations of the

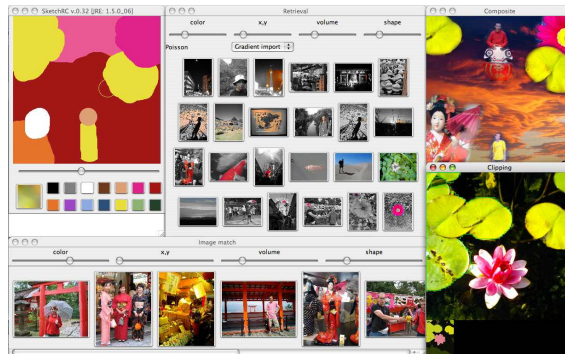


Figure 1: Retrieval and compositing interface. The top-left window is the canvas where the user paints a sketch. The window in the middle shows individual matches for every stroke. The bottom window shows matches for the whole sketch. The bottom-right window is the current selected image, where the user can select additional regions. The top-right window is the resulting collage.

same color are displayed. For every stroke on the top-left canvas, images are queried on the middle window. The color part on the retrieved images is the region that matched our stroke. Selecting a retrieved image automatically places the masked relevant part onto the top-right composite canvas.

4 Image querying

Every image is partitioned in a few segments. For every segment we compute the following features: center of gravity \mathbf{p}_0 ; a vector $\mathbf{v} = (\mathbf{v}_1, \mathbf{v}_2)$ containing the volume v_1 normalized by the image size and the volume v_2 respect its bounding box; color, represented by a Gaussian distribution on each band in the $L^*a^*b^*$ color space, $\theta = (\mu, \sigma, \mathbf{s})$, being μ, σ and \mathbf{s} the average, deviation and skewness on each band, respectively; and shape, represented by an *elongation* measure \mathcal{E} , and its orientation β . Every stroke of the sketch drawn by the user is also represented by the same features, and the matching regions from the database are the ones with closer distance d^{ij} between two objects i and j ,

$$d^{ij} = w_p d_p^{ij} + w_v d_v^{ij} + w_c d_c^{ij} + w_o d_o^{ij}, \quad (1)$$

where $d_p^{ij} = \langle \mathbf{p}_0^i, \mathbf{p}_0^j \rangle$, $d_v^{ij} = \langle \mathbf{v}^i, \mathbf{v}^j \rangle$, $d_c^{ij} = \langle \theta^i, \theta^j \rangle$, and d_o are the position, volume, color, and orientation distance, respectively; $\langle \cdot \rangle$ is the euclidean distance;

w_p , w_v , w_c , and w_o are weights that can be modified by the user. The orientation distance d_o is defined as

$$d_o^{ij} = \frac{1}{2} (\mathcal{E}^i \mathcal{E}^j (1 - \cos \alpha^{ij}) + |\mathcal{E}^i - \mathcal{E}^j|) \quad (2)$$

$$\cos \alpha^{ij} = \cos \beta^i \cdot \cos \beta^j + \sin \beta^i \cdot \sin \beta^j \quad (3)$$

$$\beta = \text{atan} \left(\frac{y'' - x'y'}{x'' - x'y'} \right) \quad (4)$$

$$\mathcal{E} = \min \left(\frac{|x'' - y''| + |x'y'|}{\max(x'', |x'y'|, y'')}, 1 \right), \quad (5)$$

where x'' , y'' and $x'y'$ are the second spatial moments of a region.

The retrieved region is used as a simple mask representing the region of interest and it is automatically selected for a composite. The user can select other areas of interest from the related image.

As the user adds more strokes to the sketch, the system retrieves new regions from the database. Moreover, it also retrieves images that match the current sketch. This image query is quickly performed using histograms. We define the distance between two images A and B as

$$d_I^{AB} = w_p \langle \mathbf{h}_p^A, \mathbf{h}_p^B \rangle + w_s \langle \mathbf{h}_s^A, \mathbf{h}_s^B \rangle + w_c \langle \mathbf{h}_c^A, \mathbf{h}_c^B \rangle + w_v \langle \mathbf{h}_v^A, \mathbf{h}_v^B \rangle, \quad (6)$$

where \mathbf{h}_p^A is the position histogram of image A , \mathbf{h}_s^A its shape histogram, \mathbf{h}_c^A its color histogram, \mathbf{h}_v^A its size histogram, and w_p , w_s , w_c , and w_v are weights for each feature. The position histogram is built by dividing the space in a 3×3 grid and counting the total volume v_1 of the regions whose center of gravity falls in a given position of the grid. The volume histogram has ten bins, each one counting the number of regions whose volume v_1 is closer to 60%, 40%, 20%, 10%, or 5% of the total volume, and greater than 50% of the volume v_2 relative to its bounding box. The color histogram contains thirteen basic color categories, obtained using a neural network. And the shape histogram contains three bins representing rounded regions, that is, no elongated, and two elongated boxes of $\mathcal{E} = 1$ vertically and horizontally oriented.

5 Color segmentation

Segmentation is performed by quantizing the image in a few color categories using a neural network and then applying N-sieves—a morphological closing followed by an opening—independently to each color category or image layer. The resulting regions are merged together over a support background layer following the criteria of occlusion maximization, and thus, preventing oversegmentation [1].

6 Image compositing

For compositing each segment in the final collage, we obtain an initial binary mask from the hard segmentation stored in the database. The boundary



Figure 2: Poisson blending. Two examples with Poisson optimization importing the source image gradient (left image of each example), and with just alpha matting (right image on each example).

is then optimized like in the Drag-and-Drop pasting method [2], and the collage so far and the new object are blended using Poisson image editing [3]. An alpha mask is also computed in a small band centered in the object boundary, following a Gaussian distribution.

The user can select whole images, disable the Poisson editing, and also disable the boundary optimization. Figure 2 shows two examples with and without Poisson blending enabled. For the collage on the left, Poisson smoothly blends the color of the images. However, Poisson optimization can make the source image colors vary greatly, even with the optimized boundary. The effect in the right collage is that the flower seems almost transparent. If the intention of the user is for the flower to be just over the water tank, then just the alpha blending (right-most result) provides the desired effect.

7 Conclusions and future work

We have proposed an interactive approach for creating image collages. Our system makes use of a sketch-based queries to retrieve parts of the target collage and composites each part seamlessly using Poisson image editing and alpha matting. An interesting topic for future research is using information such as the order of the strokes or their direction to infer relations between regions and use this information for retrieval.

References

- [1] D. Gavilan, H. Takahashi, S. Saito, and M. Nakajima. Mobile image retrieval using morphological color segmentation. In IPSJ, editor, *3rd Int'l Conference on Mobile and Ubiquitous Computing*, pages 51–58, October 2006.
- [2] J. Jia, J. Sun, C.-K. Tang, and H.-Y. Shum. Drag-and-drop pasting. *ACM Trans. Graph.*, 25(3):631–637, 2006.
- [3] P. Pérez, M. Gangnet, and A. Blake. Poisson image editing. *ACM Trans. Graph.*, 22(3):313–318, 2003.
- [4] C. Rother, L. Bordeaux, Y. Hamadi, and A. Blake. Autocollage. *ACM Trans. Graph.*, 25(3):847–852, 2006.
- [5] E. D. Sciascio, F. M. Donini, and M. Mongiello. Spatial layout representation for query-by-sketch content-based image retrieval. *Pattern Recognition Letters*, 23(13):1599–1612, 2002.