

Album Quickview in Comic-like Layout via Quartet Analysis

zhibin zheng^{1,2} yan zhang^{†1,2} zheng miao^{1,2} zhengxing sun^{1,2}

¹State Key Lab for Novel Software Technology, Nanjing University, Jiangsu, nanjing, 210093

²Department of Computer Science and Technology, Nanjing University, Jiangsu, nanjing, 210093

Abstract

For clear summary and efficient search of images for album, which carries a story of life record, we propose a new approach for quickview of album in comic-like layout via quartet analysis. How to organize the images in album and in what way to display images in collage are two key problems for album quickview. For the first problem, we take the idea of model organization method based on quartet analysis to construct categorization tree to organize the images; while for the second problem, we utilize the topological structure of categorization tree to decompose it into multiple groups of images and extract representative image from each group for subsequent collage. For the collage part, we choose comic-like layout to present collage because comic provides a concise way for storytelling and it has variability in layout styles, which is suitable for album summary. Experiments demonstrate that our method could organize the images effectively and present images in collage with diverse styles as well.

Categories and Subject Descriptors (according to ACM CCS): I.4.0 [Computer Processing and Computer Vision]: General—Image displays

1. Introduction

With the advent of the era of digital photography, more and more people take photos to record life by different digital devices. People usually take several photos with different positions or angles under the same scene, and save these photos in a digital album for later browse or recall. However, as there may exist many images in the album, browsing the image one by one may be time-consuming and tedious. Therefore, a rational quickview of album with large amounts of images will save us lots of trouble. For the quickview of album [WQS*06] [RBHB06] [ZH12], we organize images in album and then present them in comic-like layout for image summary to achieve above goal.

The main contributions are as follows:

1) Inspired by model organization method [HSS*13], we construct categorization tree to organize images effectively, which keeps the topology information of images well.

2) We propose a new comic-like layout generation method to display image summary with rich styles of layout, based on a simple template library construction method and the optimization of layout deformation.

2. Related work

Image classification. Researchers generally classify images into different subjects given the labeled training data. Distance between images are defined [MS05] [OT01] for subsequent learning methods [DF07] [VFJZ01] to classify images. However, when comparing two dissimilar images the numerical distance values are not totally reliable, leading to error of classifying. By contrast, we create quartets and construct categorization tree for images only when the numerical distances between images are reliable, and then we classify images into different groups based on the high-level topology information of categorization tree instead of distance from low-level features of images.

Image collage. Collage is one of the most popular ways to display a bunch of images, which stitches several images into one page to achieve the summary of images in a dense layout. Image Collage mainly starts from the perspective of image browsing [WQS*06] [RBHB06] [YLG*14] [ZH12], focusing on the space utilization of canvas or the compact and smooth sense of image stitching, but it's usually not easy to search image about certain scene as almost all the images are displayed in a single canvas.

Comic-like layout. Some researchers display key frames in comic-like layout in video summary [WHY*12] [Gir03]

[†] zhangyannju@nju.edu.cn

[CCL12], comic-like layout can not only present concise styles to display images but also keep the time order of key frames according to the reading order of comic. Above methods usually take several comic template layouts to display key frames, however, as the number of manually created templates is limited, the layout result is usually monotonous and lack of diversity. We take optimization for layout deformation to solve the above problem.

3. Image organization

Given an album of images, we organize them effectively inspired by model organization method [HSS*13] in two steps: quartet creation and categorization tree construction.

3.1. Quartet creation

A quartet for images contains four images with two pairs, which follows topological constraint that two images in one pair are similar but images in different pairs are dissimilar under certain feature evaluation. We try several common features of images to create quartets and finally select three features for subsequent use, including color(color histogram for calculation), SIFT [MS05] and GIST [OT01], which are widely used for calculating the similarity of images.

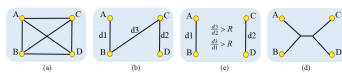


Figure 1: Quartet creation. (a):initial state for four nodes; (b):the state after removing three largest edges; (c):the state judging by threshold R ; (d): a reliable quartet where A, B are in one pair, and C, D are in the other pair.

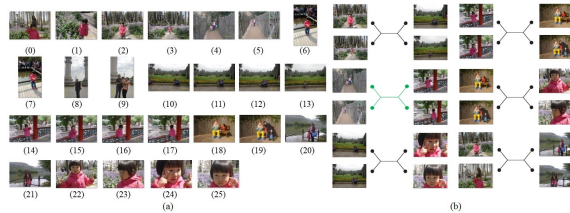


Figure 2: A subset of input album and corresponding quartets under the GIST feature. (a):input images; (b):corresponding quartets, the left two images in a quartet form one pair, and the right two images form another pair.

The process of creating quartets is shown in Figure 1. For any four images we will determine if they can form a reliable quartet by follows: we construct a undirected connected graph with four nodes A, B, C, D shown in Figure 1(a), where each node denotes an image and there exists an edge between every pair of images with weight denoting distance between images under one feature. Then we sort the six

edges according to their weight and delete the three largest edges, these four images can form a reliable quartet only if both the following two conditions are satisfied: 1) the graph is still connected; 2) denote the largest edge weight as $d3$ and the other two as $d1$ and $d2$ in the remaining three edges, $d3/d1$ and $d3/d2$ are both greater than certain threshold R . As shown in Figure 2(b), there are some quartets for the image subset of an input album under GIST feature, which shows that the quartets satisfy the topological constraint.

We conduct above process to create reliable quartets under Color, SIFT [MS05], and GIST [OT01] features. The threshold R may be different under different features, the smaller the R is, the more quartets we will get. For 60 images, we get about 8100 quartets under Color feature ($R = 2.4$), about 6400 quartets under SIFT feature ($R = 3.2$) and about 8700 quartets under GIST feature ($R = 2.2$), totally about 23200 quartets for subsequent categorization tree construction.

3.2. Categorization tree construction

After creating quartets for images, we adopt the method [HSS*13] to construct categorization tree for quartets. The categorization tree is an unrooted tree which maximally satisfies the topological information of quartets. Each leaf node of the tree denotes an image, non-leaf node is parent or ancestor of other nodes and it represents a scene category. If two leaf nodes share the same parent or ancestor, they are likely to belong to the similar scene, so we could organize images hierarchically based on the categorization tree. Topological information in quartets is kept inherently in the structure of categorization tree, for the input image of Figure 2(a), we construct the corresponding categorization tree as shown in Figure 6(b), the topological relationship between four images in the quartet which has been labeled green in Figure 2(b) can be found in the categorization tree in Figure 6(b) where the path has been labeled green.

4. Image summary in collage way

After organizing images hierarchically in the categorization tree, we need to collage images to present concise way to display images. We extract representative images from categorization tree and put them in comic-like layout. To enhance the diversity of layout styles, we construct template library for layouts, from which to select appropriate templates for subsequent optimization, presenting the collage in comic-like layout with rich styles.

4.1. Extracting representative images

After constructing categorization tree, we need to extract representative images to collage them. Through previous analysis we know that the bottom non-leaf node represents a minimum scene, and the affiliation of nodes denotes the similarity between different scenes. Besides, there exists topological information between nodes in categorization tree,

therefore we could combine the affiliation and topological information of nodes to decompose the categorization tree and extract representative images. We define the degree of separation between two nodes as the shortest path (number of traversed edges) between two nodes in the categorization tree. For the categorization tree of Figure 6(b), the degree of separation between images is shown in Figure 3(a). As the two images connected by two red edges in Figure 6(b), traversing from one image to the other needs to pass at least 2 edges, so their degree of separation is 2.

After calculating the degree of separation between leaf nodes, we decompose the categorization tree to multiple groups greedily as following steps: **step1**: starting from a random leaf node, we use breadth-first search to find all the other leaf nodes whose degree of separation from starting node is no bigger than d_{thresh} , all these nodes will be added into current group and removed from the categorization tree; **step2**: repeat step1 until there are no left nodes in the categorization tree.

We use above steps to decompose the categorization tree to multiple groups, where images in each group depict a similar scene. As shown in Figure 3(b), when we set $d_{thresh} = 2$, the input images in Figure 2(a) could be divided into 8 groups, images in each group are taken under a similar scene.



Figure 3: Degree of separation between images and groups deduced from categorization tree. (a): degree of separation between images in Figure 2(a); (b): groups deduced greedily from the categorization tree in Figure 6(b).

After getting groups of images from the categorization tree, we need to take a representative image from each group for collage. For each group, we hope to select the image which has the smallest difference between other images in the group as the representative one. The selection process is as follows: for each image in the group, we define the difference as the sum of degree of separation between the image and other images in the group, after calculating and sorting the difference for each image, we take the image whose difference is the smallest as the representative image of the group (we randomly choose one image to break the tie for several smallest difference). As shown in Figure 3(b), the image with yellow rectangle is the representative image of each group.

4.2. Template library construction

Given the representative images, we divide them randomly into summary sets where each set contains 6-8 representative images, and each set of images will be put in comic-like layout to present collage. As comic page can be represented in a hierarchical way, specifically that a comic page consists several layers and each layer consists several panels, we utilize simple enumeration method based on the universal and intuitive rules of comic to construct template library. We first enumerate the number of layers and then enumerate panels within each layer as follows: (1) each page could be divided to at most L_N layers denoted as L_1, L_2, \dots, L_{L_N} ; (2) each layer could be divided at to at most L_p panels. Considering the space limitation of comic page and the characteristic that the height of comic page is bigger than the width, we set L_N to 4 and L_p to 3 in order to keep balance of comic page and avoid crowding the page (such setting will cover most of the comic layouts created by cartoonists [Mcc06]). After enumerating the number of layers and the number of panels within each layer in a template, the position of panels within a layer is finite that we can simply enumerate all the possible cases: if the number of panels within a layer is three, there are three cases of division from layer to panel, otherwise there is only one division case as shown in Figure 4. By enumerating the string representation of template as shown in Figure 4 (each panel is represented by its layer and its position within layer), we enumerate different cases of layers and panels within each layer to create a template library. For each template, we divide the whole comic page into layers equally and then divide each layer into panels equally as shown in Figure 5(a).

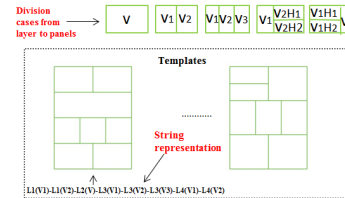


Figure 4: Enumeration of panels in each layer and templates with corresponding string representation.

4.3. Selecting appropriate templates

Given a library of templates, we need to select appropriate templates for each summary set of images. For a summary set of images and a given template layout, we scale each image using the same proportion and put it into corresponding panel following the center-aligned manner as shown in Figure 5(b). However, due to the different sizes of images and panels, there may be some phenomenon such as gap and overlap between current panel and adjacent panels within a layer as shown in Figure 5(b). Besides, for each layer, scaled image may overflow boundary of corresponding panel and overlap adjacent layers. We seek to preserve the content of

each image maximally, but above phenomenon of overlap or gap will cause the crop of scaled images to adapt the size of corresponding panel. Considering above phenomenon, we define penalty function for each template, based on which to select appropriate templates from the template library.

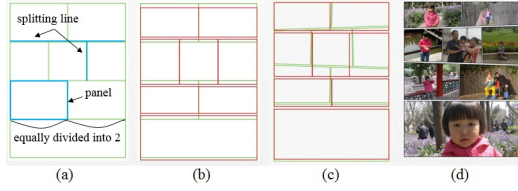


Figure 5: Layout optimization example. (a):layout template; (b):initial panels (green) after putting scaled images (red); (c):optimized layout result (green denotes deformed panels and red means scaled images); (d):final layout result.

Before introducing penalty function, we define symbols as follows: for a summary set of images S , denote its corresponding template library as $\{T_m\}_{m=1}^M$ (M is the number of templates in template library); for current template T_m , denote its layer as $L_i(i \in \{1..4\})$, for layer L_i , denote the panel on L_i as $P_{i,j}(j \in \{1..3\})$, which means the j^{th} panel of L_i . The scaled image after put into corresponding panel is denoted as $I_{i,j}(j \in \{1..3\})$, which corresponds to panel $P_{i,j}(j \in \{1..3\})$. \cap is the intersection operator, which intersects two regions, and \cup is the merge operator, which merges two regions. For template T_m , we will introduce the factors of influence.

1): The overlap within layer $L_i(i \in \{1..4\})$: as we don't want to crop too much region of scaled image to fit corresponding panel, we hope the overlap between panels within layer L_i as small as possible. The overlap between panels within layer L_i is defined as follows:

$$E_o(L_i) = \sum_{j=1}^{J-1} (Area(I_{i,j}) \cap Area(I_{i,j+1})) \quad (1)$$

Where $E_o(L_i)$ means the overlap within layer L_i , $J \in \{1..3\}$ is the number of panels of layer L_i , $Area(I_{i,j})$ means the area of scaled image $I_{i,j}$.

2): The gap within layer $L_i(i \in \{1..4\})$: as we don't want to rescale and crop too much of the image, we also hope the gap as small as possible, penalty for gap within layer L_i is defined as follows:

$$E_g(L_i) = Area(L_i) - \sum_{j=1}^{J-1} (Range(L_i) \cap (Area(I_{i,j}) \cup Area(I_{i,j+1}))) \quad (2)$$

Where $E_g(L_i)$ means the gap within layer L_i , $J \in \{1..3\}$ is the number of panels of layer L_i , $Area(L_i)$ means the area of layer L_i . $Area(I_{i,j})$ means the area of scaled image $I_{i,j}$. $Range(L_i)$ is the scope of the rectangle of layer L_i .

3): The overlap between layer L_i and $L_{i+1}(i \in \{1..3\})$: for adjacent layers, if scaled image overflows the boundary of panel and covers adjacent layers, it will not only cause cropping of scaled image but also affect the overall appearance of layout. We define the overlap between layer L_i and L_{i+1} as follows:

$$D(L_i, L_{i+1}) = Area\left(\sum_{j=1}^J I_{i,j}\right) \cap Area\left(\sum_{k=1}^K I_{i+1,k}\right) \quad (3)$$

Where $D(L_i, L_{i+1})$ means the overlap between layer L_i and L_{i+1} , $Area(I_{i,j})$ means the area of the j^{th} scaled image which corresponds to $P_{i,j}$ on layer L_i , $J \in \{1..3\}$ is the number of panels on layer L_i . $Area(I_{i+1,k})$ means the area of the k^{th} scaled image which corresponds to $P_{i+1,k}$ on layer L_{i+1} , $K \in \{1..3\}$ is the number of panels on layer L_{i+1} .

Based on above phenomenon we define penalty function for template T_m as follows:

$$Penalty(T_m) = \alpha \sum_{i=1}^{NL} (\aleph(E_o(L_i)) + \aleph(E_g(L_i))) + \beta \frac{\sum_{i=1}^{NL-1} D(L_i, L_{i+1})}{AREA} \quad (4)$$

where $Penalty(T_m)$ is the penalty function for template T_m . NL is the number of layers of T_m , \aleph is the MIN-MAX normalization operator for $E_o(L_i)$ and $E_g(L_i)$. $AREA$ is the area of comic page, which acts as normalization factor for $D(L_i, L_{i+1})$. α, β are corresponding weights for local constraint and global constraint, we set $\alpha = 0.6, \beta = 0.4$ by experiment. We could choose the top- k templates whose penalty function is the top- k minimum as appropriate templates from template library for each summary set S . After putting scaled image into corresponding panel of each appropriate templates, we get corresponding initial layout results.

4.4. Layout optimization

Given the initial layouts for summary set S , as each panel in a template is rectangle, the diversity and visual effect of layout is limited. Besides, for each image, if the height-width ratio of the image is greatly different from that of corresponding panel, we need to rescale and crop the image much to fit the panel. Considering above two facts, we could deform each panel not only to add variability for its shape to enhance the diversity and visual effect, but also adjust its size to make the height-width ratio between corresponding image as close as possible to casue less crop of rescaled image. Since each panel in template corresponds to a quadrilateral intersected by four lines (splitting lines and boundary lines, as shown in Figure 5(a)) of comic page, we move the splitting lines to optimize the deformation of panels to make the height-width ratio between images and corresponding panels as close as possible. Fitness function for optimization is defined as fol-

lows:

$$Y(T_m) = \sum_{i=1}^{NL} \sum_{j=1}^J (Ratio(P_{i,j}) - Ratio(I_{i,j}))^2 \quad (5)$$

Where T_m is current template of top- k templates, $NL \in (1..4)$ is the number of layers of T_m , $J \in (1..3)$ is number of panels of layer L_i . $P_{i,j}$ denotes the j^{th} panel on layer L_i , and $I_{i,j}$ means the scaled image which needs to be put into panel $P_{i,j}$. $Ratio(x)$ calculates the height-width ratio of x (if x is irregular quadrilateral, it calculates the height-width ratio of its minimum bounding box). We aims at minimizing $Y(T_m)$.

We use PSO [KE95] algorithm to optimize the fitness function, each particle in PSO encapsulates the parameters of splitting lines of comic page, splitting lines will move to determine the optimized shape of panels after every iteration, and converged to determine the final shape of panels. After optimization, panels have variability in shapes and size to enhancing the diversity of layout while optimally preserving the content of images, as shown in Figure 5(c). Besides, there is usually row spacing and column spacing in real comic layout, so we take a simple movement of splitting lines in comic page to add such effects (in our method, we set row spacing as 15 pixels and column spacing as 8 pixels). The final comic layout result is shown as Figure 5(d), after optimization, panels have variability in shapes and preserves the content of images as much as possible.

5. Experiments and analysis

In this section, we will introduce the experiments and analysis of our method. We also carry out user evaluation to verify the effectiveness of our method, the limitation of our method is discussed as well.

5.1. Experimental results

We implement the whole process in C++ on a PC with Intel Core 2 Duo E7300 and 3.1GHz CPU with 4G memory. We carry out experiments on several albums, each containing dozens of pictures which were taken under different scenes by different digital devices.

To validate the effectiveness of our method, we first validate the effectiveness of the organization of images by comparing the categorization tree with ground truth, then compare our collage with AutoCollage [RBHB06] and photo-collage [YLG*14].

For each set of images, 10 users are asked to construct the ground truth of categorization tree based on the principle that images taken under a similar scene should locate nearer in the ground truth of categorization tree and each leaf node represents an image. They construct the ground truth iteratively as follows: when the ground truth is accomplished by one user it is transferred to another user to modify, the process stops when all users are satisfied with the

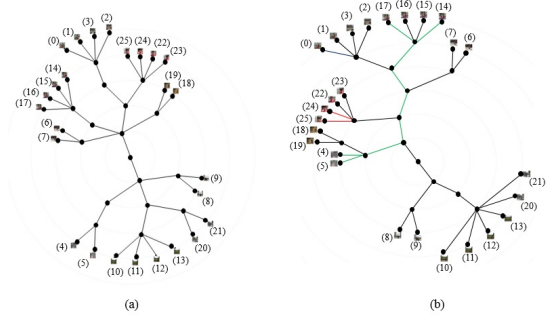


Figure 6: Comparison with ground truth for categorization tree for input images of Figure 2(a). (a):ground truth constructed by users; (b):categorization tree of ours.



Figure 7: Comparison with AutoCollage [RBHB06] and photo-collage [YLG*14] for input images of Figure 2(a). (a):collage by AutoCollage; (b):collage by photo-collage; (c):collage of our method.

ground truth. The comparison between categorization tree with ground truth is shown in Figure 6, which shows our categorization tree keeps similar topology structure with the ground truth, demonstrating the effective organization of images based on the categorization tree.

In addition, we compare our collage with AutoCollage [RBHB06] and photo-collage [YLG*14] as shown in Figure 7. AutoCollage and photo-collage both display images in a compact way, however, they don't take into account of the similarity between images, all input images are arranged randomly in the canvas, making it hard to search specific image. In comparison, we organize images by considering the similarity between images and display the summary of images in simple comic-like layout, making it easy to navigate and search desired image.

5.2. User evaluation and limitations

We carry out user evaluation to validate the effectiveness of our collage. We recruit 40 participants (20 males and 20 females) to compare our results (shown in attachment material) with results from AutoCollage and photo-collage from two perspectives: visual effect for browsing and efficiency for search certain image.

Visual effect for browsing: Is the collage result visual

pleasing and clear for view on the whole? Participants are asked to assign a score from 1 to 5 for each result (higher value means better), we compare the average value for each result as shown in Figure 8, which demonstrates that out collage gives clearer and more visual pleasing results.

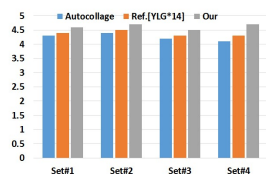


Figure 8: Comparison with AutoCollage [RBHB06] and photo-collage [YLG*14] from the visual effect perspective.

Efficiency for search: For the specific images, what time takes participants to find them in the collage? We count the average time for participants to find each specified image. For the 5 specified images, the average time for search that in collage is shown in Figure 9, which shows that our collage is more efficient for search specific image, we analyses that our method summary the images and display them in a concise way.

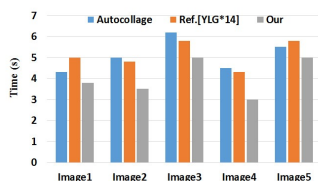


Figure 9: Comparison with AutoCollage [RBHB06] and photo-collage [YLG*14] from the perspective of efficiency for search.

Limitations: There also exist some limitations for our approach. Firstly, our method is not applicable to all types of input images, but only to the images that each subset of them are taken under a similar scene so that we can create reliable quartets from them. Besides, we haven't consider any high-level semantics of images, we think that if we have the support of high-level features of images, our quartets will be more reliable. Finally, user interaction is not supported in the process of creating collage, which is a shortcoming of our collage.

6. Conclusion and future work

This paper proposes a new approach for album quickview. By constructing categorization tree for album we organize images effectively, and we also propose a new comic-like layout generation method for image summary in album, presenting a quickview of album with diverse layout styles. For

the future work, we aim to add high-level features of images to create more reliable quartets, and add user interaction in the process of collage creation to present more abundant collage styles..

7. Acknowledgement

We would like to thank all anonymous reviewers for their constructive comments. This research has been supported by the National Science Foundation of China (61321491, 61100110,61272219) and the Science and Technology Program of Jiangsu Province (BY2012190, BY2013072-04).

References

- [CCL12] CAO Y., CHAN A. B., LAU R. W. H.: Automatic stylistic manga layout. *ACM Trans. Graph.* 31, 6 (Nov. 2012), 141:1–141:10. 2
- [DF07] DUECK D., FREY B. J.: Non-metric affinity propagation for unsupervised image categorization. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on (2007)*, IEEE, pp. 1–8. 1
- [Gir03] GIRGENSOHN A.: A fast layout algorithm for visual video summaries. In *ICME (2003)*, IEEE, pp. 77–80. 1
- [HSS*13] HUANG S.-S., SHAMIR A., SHEN C.-H., ZHANG H., SHEFFER A., HU S.-M., COHEN-OR D.: Qualitative organization of collections of shapes via quartet analysis. *ACM Trans. Graph.* 32, 4 (July 2013), 71:1–71:10. 1, 2
- [KE95] KENNEDY J., EBERHART R. C.: Particle swarm optimization. In *Proceedings of the IEEE International Conference on Neural Networks (1995)*, vol. 4, pp. 1942–1948. 5
- [McC06] MCCLOUD S.: *Making Comics: Storytelling Secrets of Comics, Manga*. Harper Paperbacks, 2006. 3
- [MS05] MIKOLAJCZYK K., SCHMID C.: A performance evaluation of local descriptors. *IEEE Trans. Pattern Anal. Mach. Intell.* 27, 10 (Oct. 2005), 1615–1630. 1, 2
- [OT01] OLIVA A., TORRALBA A.: Modeling the shape of the scene: A holistic representation of the spatial envelope. *Int. J. Comput. Vision* 42, 3 (May 2001), 145–175. 1, 2
- [RBHB06] ROTHER C., BORDEAUX L., HAMADI Y., BLAKE A.: Autocollage. *ACM Trans. Graph.* 25, 3 (July 2006), 847–852. 1, 5, 6
- [VFJZ01] VAILAYA A., FIGUEIREDO M. A. T., JAIN A. K., ZHANG H.: Image classification for content-based indexing. *IEEE Transactions on Image Processing* 10, 1 (2001), 117–130. 1
- [WHY*12] WANG M., HONG R., YUAN X.-T., YAN S., CHUA T.-S.: Movie2comics: Towards a lively video content presentation. *IEEE Transactions on Multimedia* 14, 3-2 (2012), 858–870. 1
- [WQS*06] WANG J., QUAN L., SUN J., TANG X., SHUM H.-Y.: Picture collage. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on (2006)*, vol. 1, IEEE, pp. 347–354. 1
- [YLG*14] YU Z., LU L., GUO Y., FAN R., LIU M., WANG W.: Content-aware photo collage using circle packing. *IEEE Transactions on Visualization and Computer Graphics* 20, 2 (2014), 182–195. 1, 5, 6
- [ZH12] ZHANG L., HUANG H.: Hierarchical narrative collage for digital photo album. *Comp. Graph. Forum* 31, 7pt2 (Sept. 2012), 2173–2181. 1