

Puzzle-like Collage

Stas Goferman Ayellet Tal Lihi Zelnik-Manor

Technion – Israel Institute of Technology

Abstract

Collages have been a common form of artistic expression since their first appearance in China around 200 BC. Recently, with the advance of digital cameras and digital image editing tools, collages have gained popularity also as a summarization tool. This paper proposes an approach for automating collage construction, which is based on assembling regions of interest of arbitrary shape in a puzzle-like manner. We show that this approach produces collages that are informative, compact, and eye-pleasing. This is obtained by following artistic principles and assembling the extracted cutouts such that their shapes complete each other.

1. Introduction

A collage is a work of the visual arts, made from an assemblage of different forms, thus creating a new whole, often having a purposeful incongruity. This paper focuses on photo-collages, which assemble a collection of photographs by cutting and joining them together. A photo-collage can be used for art [Ade89] as well as for summarizing a photo collection, such as a news event, a family occasion, or a concept (e.g., the Beatles' album cover of "Sgt. Pepper's Lonely Hearts Club Band").

Manually creating a collage is a difficult and time-consuming task, since the pieces should be nicely cut and matched. Therefore, automation could be a welcomed tool. Prior work on automating collage creation extracts *rectangular* salient regions and assembles them in various fashions [RBHB06, WSQ*06, BCG*07, Pic]. This produces beautiful collages, however, since the extracted regions are rectangular, the variety of possible compositions is limited and uninteresting regions are included. In [RBHB06] the rectangular shapes are modified by graph cuts and alpha blending. This creates nicer transitions between images, however, the uninteresting regions (typically from the background) cannot be eliminated.

This approach to assemblage, while informative, does not match in spirit the way in which many artists construct collages. Artists extract the expressive regions of interest, which can be of arbitrary shape, as noted by Henri Matisse: "The paper cutouts allow me to draw with color". This approach is expressed in numerous artistic collages, for instance see

the pioneering works of "Just What Is It that Makes Today's Homes So Different, So Appealing?" by Richard Hamilton and the "Dada Siegt" by Raoul Hausmann. The critical boundaries of the important information are considered significant and are thus maintained. This artistic form of collage has gained popularity also among amateurs as can be seen by the hundreds of collage groups on *Flickr* and hundreds of thousands users in *Polyvore* [Pol], a web-based application where collages are *manually* assembled by users.

We propose a method for automating collage creation, which is inspired by artistic collage work and glues mostly the purposeful cutouts (see Figure 1). The fundamental difference between prior work and ours is that we compose a puzzle-like collage of arbitrary shaped images rather than rectangular ones. A user-study shows that this creates collages that are often considered more appealing. Moreover, this allows us to generate space-efficient collages, which are useful for summarization of image data sets.

The main contribution of this work is a complete system for image collage whose key idea is to assemble arbitrary-shaped cutouts of interesting regions. This requires solving two challenges: extraction of non-rectangular regions-of-interest, and assembly of arbitrary shapes.

For the former we propose an algorithm that extracts non-rectangular regions that coincide with the meaningful information – object boundaries and salient background features (Section 4). Image segmentation is a well studied and difficult problem. Here, we do not claim to solve it but rather we propose an effective solution for our specific application.

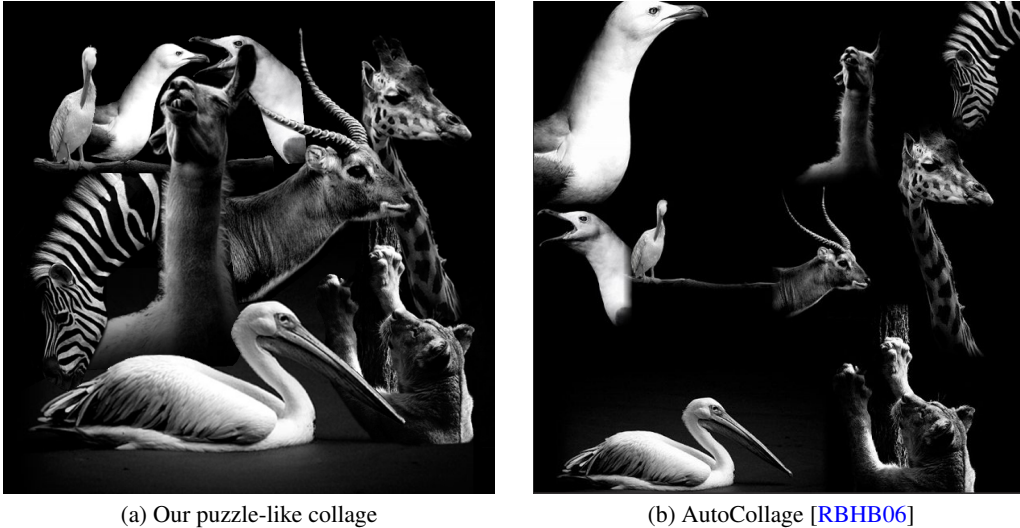


Figure 1: Dark zoo collages created from 9 images. Note how the assembly of the animals in our collage is like a puzzle. This is both more aesthetic as our user-study shows and uses the canvas more efficiently (hence most of the animals appear larger).

Unlike segmentation, the region-of-interest may include several objects and parts of the background to convey the context. Moreover, an important observation is that a perfectly accurate extraction is not compulsory due to the following three reasons. First, even when some ROIs are not perfect, their overall shape is typically close to the correct one and a placement can be computed. Second, the overlaps between the ROIs in the collage often conceal the inaccuracies. Finally, any remaining visible imperfections can then be manually corrected using, for example, *Soft Scissors* [WAC07].

For the assembly we propose an algorithm that composes the ROIs while finding a good balance between compactness and informativeness (Section 5). Since the shapes are non-rectangular, this resembles puzzle-solving. The shapes, however, cannot perfectly match, as assumed in a standard puzzle, and some overlap is allowed.

2. Related work

Techniques of collage were first used at the time of the invention of paper in China around 200 BC. Since then, these techniques have been used in various forms in other cultures. In spite of its early creation, the term “collage” was coined at the beginning of the 20th century, by both Georges Braque and Pablo Picasso. These were the times when the use of collages made a dramatic appearance among oil paintings and became a distinctive part of modern art.

Methods for automatic creation of photo-collages were proposed only recently. For example, [ADA*04] make collages of aligned nearly-identical images, while [Atk04] organize whole pictures on a page and [GC04] assemble faces. The works most related to ours are [RBHB06, WSQ*06,

BCG*07, Pic]. [RBHB06] define a method, *AutoCollage*, for constructing a seamless collage from an image set. In this work, rectangular salient image regions are stitched together seamlessly using edge-sensitive blending. In *picture collage* [WSQ*06], a 2D spatial arrangement of rectangular images is optimized in order to maximize the visibility of the salient regions. [BCG*07] propose an improvement to *picture collage*, by exploiting semantic and high-level information in saliency computation and a genetic algorithm to positioning. Google’s Picasa [Pic] features automatic collage generation of whole (or cropped) images, supporting different styles of compositions. We propose to use ROI images of arbitrary shapes, rather than rectangular ones.

Most previous work on ROI extraction was limited to detecting rectangular ROIs by finding a rectangular bounding box of the salient pixels [RKKB05, RBHB06, WSQ*06, BCG*07]. Usually such ROIs include many non-salient pixels. In [HXM*04], more space-efficient ROIs are created by using the convex hull of the salient points. This reduces the number of non-salient pixels, but is still not accurate enough. Methods for image segmentation [RKB04, LSTS04, LRAL08] aim at accurately extracting foreground pixels by cutting along image contrast boundaries. In images where the region-of-interest corresponds to a foreground region, such segmentation-based methods could suffice for ROI extraction. However, in many images parts of the background are essential for capturing the message they convey. Our approach combines saliency and contrast information, thus addressing both accuracy and informativeness.

Constructing a collage from image fragments of irregular shapes resembles assembling a 2D puzzle. The jigsaw puz-

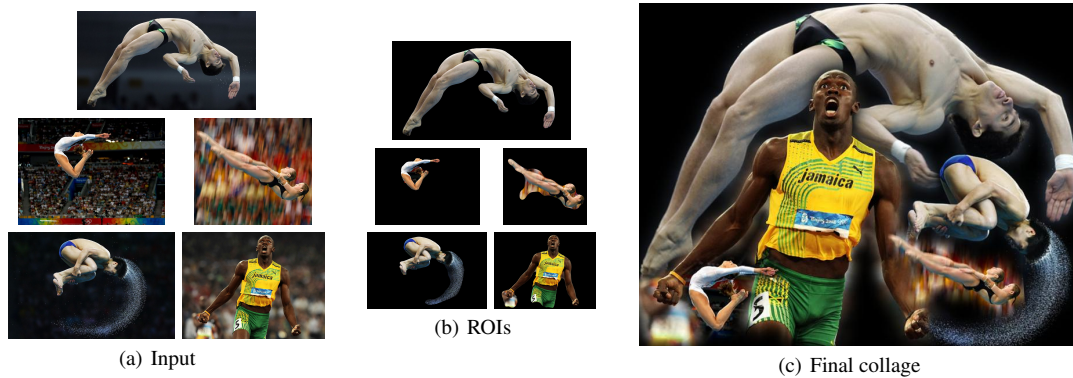


Figure 2: Our collage construction framework applied to an image collection describing highlights of the 2008 Olympic games.

zle problem is often approached in two stages (e.g. [RB80, KK01, SKKC00]). First, local shape matching finds pairs of fragments that fit perfectly. Then, a global solution is obtained by resolving ambiguities. Collages differ from puzzles in that the fragments typically do *not* match perfectly, and they are allowed to overlap each other. Therefore, our assembly algorithm aims at finding an informative, yet compact composition, rather than a unique and exact assembly.

3. Framework

Given a collection of images, we wish to construct an informative, compact, and visually appealing collage. Figure 2 illustrates the steps of our algorithm on a set of images describing events from the 2008 Olympic games. Given the images in Figures 2(a), we first extract the regions-of-interest (ROI), as shown in Figures 2(b). Note the non-rectangular ROIs that accurately cut out the salient regions. Figure 2(c) illustrates the final assembly, in which the runner and the forward spinning diver fit in the crevices created by the back-flipping diver's arched back and spread arms, respectively.

To construct informative collages, most of the ROIs should remain visible after assembly. For many images, the ROI consists of foreground pixels alone. This is typical for images with a shallow depth-of-field or images having a blank background (see Figure 3(top)). For images like these, standard image segmentation could suffice. However, for many images, both professional and non-professional, the informative regions consist of both foreground as well as background pixels that provide the context. For instance, in Figure 3(bottom), a meaningful ROI should include the car, the fire and part of the surrounding, and the kids as well as part of the bed they are sitting on. The proposed ROI extraction algorithm incorporates saliency and edge-based information, thus it can handle both types of images (Figure 3(d,g)).

To generate a collage the user can choose to click a button



Figure 3: ROI extraction. GrabCut is a semi-automatic segmentation algorithm, capable of extracting ROIs in images with a simple background like those on the top. It is less successful for complex images like those on the bottom. The result by Polyvore is fully automatic but less successful even for simple images. Our method can handle both cases.

and get a fully automatic composition. Alternatively, the user can do the following to better control the final result:

- Set the desired aspect ratio of the generated collage.
- Tune a parameter controlling the amount of allowed overlap between images.

- Attach importance weights to images, which will influence their size in the final collage.
- Manually fix the position of some images.
- "Glue" images together to be treated as one.

When the user selects the automatic mode, the images are randomly scaled (within a limited range) and default overlap and aspect ratio are applied. The assembly algorithm generates a puzzle-like collage by fitting the pieces. Most of the collages in this paper were created fully automatically, except for setting the aspect ratio. We mention explicitly when user interaction was applied.

4. Region-Of-Interest (ROI) extraction

To obtain informative collages, our goal is to extract from each image a region of interest (ROI) of an arbitrary shape, which takes a binary decision at each pixel and labels it as either "interesting" or "not interesting". To achieve this goal we define the following desired properties of an ROI:

1. The ROI should enclose most of the salient pixels.
2. The boundary of the ROI should coincide well with the natural edges of the image.
3. The boundary curve should enable visually appealing compositions (e.g., jagged boundaries are undesired).

These requirements emphasize the difference between ROI extraction and foreground-background segmentation. Image segmentation aims at accurately extracting the foreground objects, satisfying only Requirement 2. Conversely, ROIs should include most of the informative regions (Requirement 1) and enable pretty compositions later on (Requirement 3). In other words, ROIs should include pieces of the background, when these are helpful for understanding the picture. Note that not all the background should be included – only enough of it for conveying the context.

Our algorithm consists of four steps, which incorporate saliency and edge information, in order to comply with the requirements, as illustrated in Figure 4. First, a saliency map is computed and an initial curve is constructed from it. This curve is propagated towards the natural image edges and smoothed based on the saliency.

Saliency computation: We first compute the saliency of each image (Figure 4(b)). Though saliency can be computed using a variety of previously proposed algorithms [IK01, HZ07, LSZ*07], we apply our own implementation. For each pixel we consider the patch surrounding it and find the patches in the image that are most similar to it. Salient pixels are those that do not have similar patches elsewhere in the image. The result of this stage is a saliency map where each pixel i is assigned a saliency value S_i , which indicates the dissimilarity of the pixel to its K nearest neighbors:

$$S_i = 1 - \exp\left\{-\frac{1}{K} \sum_{k=1}^K d(p_i, q_k)\right\}$$

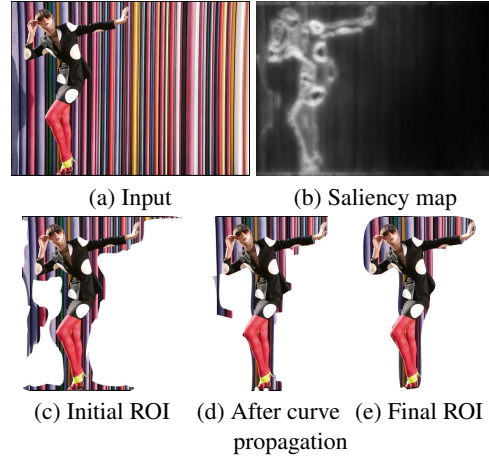


Figure 4: The 4 steps of our ROI extraction algorithm. The initial ROI contains the salient pixels, but is not visually appealing. After curve propagation, the ROI boundary coincides with the image edges, however, it is noticeably jagged. The final ROI is smooth in regions where the saliency is low (e.g., above the head), but coincides well with more salient image edges (e.g., knees and arms).

Here $d(p_i, q_k)$ is the appearance difference between patches p_i and q_k . This is enhanced by a standard face detection [VJ01].

Curve initialization: Curve initialization aims at satisfying the first requirement. Initialized by an empty ROI, pixels are added to it sequentially, starting from the highest saliency values to the lowest. This continues until the total saliency of the included pixels reaches 90% of the total image saliency. This results in a binary mask of one or more connected components of "interesting" regions. The boundary of the largest connected component serves as an initial contour $C(t = 0)$ for the next step (Figure 4(c)).

Curve propagation: To satisfy Requirement 2, the initial curve is propagated towards the image edges, while keeping in mind the saliency map. We base our approach on a geodesic active contours model [CKS97] and modify it to incorporate saliency. Here the curve C is represented implicitly via a function ϕ , by $C(t) = \{(x, y) | \phi(t, x, y) = 0\}$, and the evolution of the curve is given by the zero-level curve at time t of the function $\phi(t, x, y)$:

$$\begin{aligned} \frac{\partial \phi}{\partial t} &= \lambda |\nabla \phi| G(\phi) + \gamma |\nabla \phi| H(\phi), \\ G(\phi) &= \operatorname{div} \left(g(|\nabla u_0|) \frac{\nabla \phi}{|\nabla \phi|} \right) + \nu g(|\nabla u_0|), \\ H(\phi) &= h\kappa(\phi). \end{aligned} \quad (1)$$

In Equation (1) u_0 is the lightness channel, $g(\cdot)$ is an edge indicator function, h is a saliency indicator function and ν is a positive constant pushing the curve inwards. The curvature



Figure 5: Examples of our ROIs. Our approach can handle single objects as well as complex scenes. Irrelevant background is excluded while parts of informative background are kept.

of the level-set function is defined by $\kappa = \text{div}(\nabla\phi/|\nabla\phi|)$. Note that setting $\gamma = 0$ results in a geodesic model, where the zero-level curve moves in the normal direction with speed G and stops on the desired boundary, where g vanishes. Setting $\lambda = 0$, we get a saliency-based evolution in the normal direction with speed H , where the curve stops on salient regions.

The importance of our saliency term H is twofold. First, it accelerates the curve evolution in non-salient regions. This is especially important when the curve encounters locally strong, but non-salient edges, which occur at many background non-salient pixels. Second, it slows down the evolution in salient regions.

We set $g(\nabla u_0) = 1/(1 + |\nabla G_\sigma * u_0|^2)$, where $G_\sigma * u_0$ is a smoothed version of u_0 . G_σ is the Gaussian kernel with standard deviation 1.5. The saliency indicator function is selected by $h = \exp\{-\hat{S}^2/\sigma_s^2\}$, where \hat{S} is the saliency map and σ_s^2 is its variance.

The level-set evolution is implemented using a numerical scheme of [LXGF05], which eliminates the need of re-initialization of ϕ , where the zero-level curve at $t = 0$ is the curve from the previous step. In our implementation, we used $\lambda = 3$, $\gamma = 5$, $\nu = 1$.

The evolution continues until either a maximal number of steps (1000) is reached (i.e., we cannot go too far from the initial curve) or the sum of the image saliency values inside the curve drops below a certain level (50% of the total saliency).

Curve visual enhancement: As can be seen in Figure 4(d), the propagated curve bounds the region of interest. However, the curve itself might be still jagged, since it was pushed to pass through the image edges (which may be non-smooth). To satisfy the third requirement on ROIs we next further smooth the curve in accordance with the saliency, so that later on it can be nicely matched to other shapes in the collage (see the result in Figure 4(e)).

This is done by applying the level set-evolution with saliency term only ($\lambda = 0$), and enforcing the same stopping criteria. Note that in this formulation, the curve's curvature is smoothed while its length barely changes since its propaga-

tion is stopped by high-saliency values. Moreover, the evolution is stronger where the saliency is low, as intuitively it should be.

Results: Figure 5 presents several extracted ROIs. Our ROIs are of smooth irregular shapes that bound nicely the region of interest. When the boundary between foreground and background is clear, our approach cuts out the foreground accurately, as can be seen in the pictures of the blue-dressed Italian and the excited American speaker. In more complex images, such as the cheering kids and the weight-lifter, the ROI includes parts of the background, as expected.

5. Collage assembly

The last step of our framework is the assembly of the ROIs. Our assembly algorithm expects as input a set of n images, together with their ROIs and saliency maps. A set of additional, however optional, parameters such as the desired overlap ratio of the final collage, the amount of allowed overlap, preferences of ROIs (importance weights between 0 and 1), and constraints on their location lets the user control the creativeness of the final result. We next describe the fully automatic solution and how user input is incorporated.

Our goal is to generate collages that are: (1) Informative: The collage should include as much as possible from every image's region of interest. (2) Compact: The collage should utilize the canvas efficiently, subject to a desired aspect ratio.

To achieve this, we first formulate a cost function whose minimization would produce collages adhering to these properties. We then propose an algorithm that aims at minimizing the cost function in two steps: initial placement and local refinement. In the first step, the ROIs are scaled randomly (or according to importance weights when these were provided) and then efficiently packed in a puzzle-like manner, subject to a desired aspect ratio. To further improve collage properties, we apply local refinements in the second step, by allowing the following ROI transformations: scaling, shifting, rotating, and changing layering position.

Composition cost: Given a placement of ROIs and their layering, we define the composition cost C as a combination of

informativeness C_{info} and compactness $C_{compact}$ cost functions subject to a desired aspect ratio δ_{AR} :

$$C = (\alpha \cdot C_{info} + (1 - \alpha) \cdot C_{compact})^{\delta_{AR}} \quad (2)$$

where $0 < \alpha < 1$ controls the amount of overlap.

Overlap: The two cost terms C_{info} and $C_{compact}$ aim at opposite goals – a highly informative layout is not compact and vice versa. The tradeoff between them is controlled by the parameter α . Setting α to a low value could lead to a creative composition with high overlap, while informative image parts might be occluded. High values of α could eliminate occlusions at the cost of losing compactness. A typical value of $\alpha = 0.5$ is a good compromise between the two cost terms.

Aspect ratio: Let AR be a user-desired aspect ratio and ar be a current aspect ratio. To penalize placements of an undesired aspect ratio we define the penalty term δ_{AR} as:

$$\delta_{AR} = \exp\left\{-\frac{1}{2\sigma_{AR}^2}(AR - ar)^2\right\}, \quad (3)$$

where we set $\sigma_{AR} = 0.2$.

Informativeness cost: Given a placement of k ROIs and their layering, we define the informativeness cost as the sum of occluded salient regions normalized by the total amount of all salient regions. Let \tilde{S}_i be the saliency map of image i normalized by its ROI area, R_i be its ROI, and R_i^{occ} be the occluded part of R_i . Then,

$$C_{info} = \frac{\sum_{i=1}^k R_i^{occ} \tilde{S}_i}{\sum_{i=1}^k R_i \tilde{S}_i}. \quad (4)$$

Compactness cost: The compactness of a given configuration of ROIs can be measured by the amount of empty space captured in its axis-aligned bounding rectangle. Let $bound$ be the area of the bounding rectangle. We wish to minimize

$$C_{rect} = 1 - \frac{\bigcup_{i=1}^k R_i}{bound(\bigcup_{i=1}^k R_i)}. \quad (5)$$

This guarantees compact layouts, although it might be insufficient for matching protrusions and depressions of the ROIs (Figure 6(a)). Therefore, we also minimize the empty space between the ROIs. Let $conv$ be the area of the convex hull of the union of ROIs, we minimize the empty space in $conv$:

$$C_{conv} = 1 - \frac{\bigcup_{i=1}^k R_i}{conv(\bigcup_{i=1}^k R_i)}. \quad (6)$$

Minimizing each term alone does not suffice, since minimizing only C_{conv} could result in diagonal or elongated shapes which are unappealing (see Figure 6(e)) and minimizing only C_{rect} might create rectangular, yet less puzzle-like, compositions (Figure 6(d)). Therefore, we incorporate both terms and define the compactness cost as:

$$C_{compact} = C_{rect} \cdot C_{conv}^{\omega_c}, \quad (7)$$

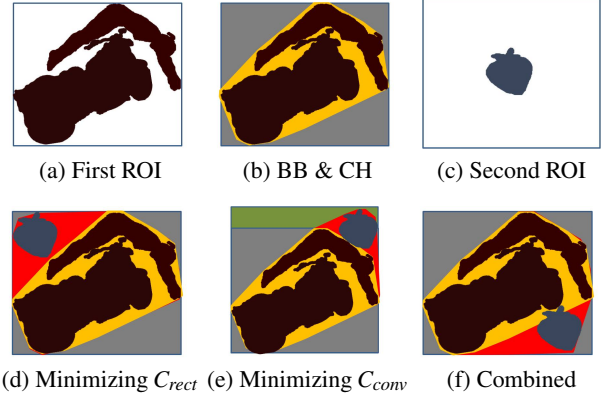


Figure 6: The ROI of the current assembly (a) needs to be combined with a new ROI (c). The bounding rectangle of the current assembly is marked in gray and its convex hull marked in yellow (b). Minimizing only C_{rect} (d) can place the strawberry at the top left corner since it does not change the bounding rectangle. But, this increases C_{conv} (in red). Minimizing only C_{conv} (e) places the strawberry at the top right. The increase in C_{conv} (red) is small but the bounding rectangle is now much larger (marked in green). The combination (f), which minimized $C_{compact}$, places the strawberry at the bottom right. This way the bounding rectangle is not increased at all and C_{conv} is increased only a bit and significantly less than when minimizing only C_{rect} .

where in our experiments we set $\omega_c = 0.5$.

Initial Placement: In search for an assemblage algorithm that minimizes the defined cost we turned to the puzzle-solving literature. Their solutions, however, were found inadequate, since in our case the shapes do not perfectly match as they do in puzzles. A more fruitful avenue to follow was to consider the fundamental problem of 2D bin packing. Our assembly problem can be viewed as a generalization of 2D bin packing, where in our case the parts are not constrained to be rectangles and overlaps are allowed. 2D bin packing has been shown to be NP-hard [Sle80], nevertheless, there exists a variety of approximation strategies [LMM02]. We draw inspiration from the general strategy proposed by [Joh73], in which a *best-first* decreasing approach is proposed.

The best-first strategy is well suited to our task both since it provides a near-optimal solution to the bin-packing problem, and due to its visual implications. Inspired by artistic collage principles we define the “best” ROI to place as the largest (i.e. the most important) one. Placing ROIs in a decreasing order of importance (and size) usually results in nicer compositions, where larger ROIs are condensed together near the center of the image. To realize this approach, the ROIs are first scaled, then placed in a “best-fit” manner, and finally layered. We elaborate below.

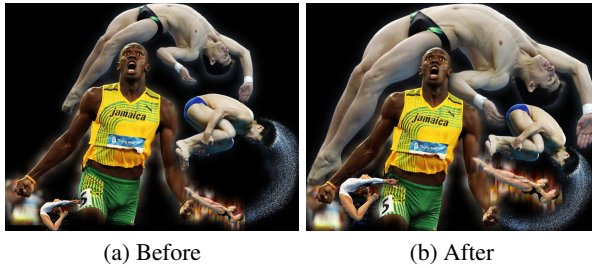


Figure 7: Local refinement

First, the ROIs are scaled. In spirit of art, collages created from images of different sizes are usually more visually interesting. When no input is provided by the user we randomly scale the ROIs such that the smallest ROI is at most 4 times smaller than the largest one (we never enlarge an ROI to maintain high quality). The creative user can manually set importance weights, w_i , to preferred (or all) images, which serve as scaling weights instead of the random scaling. For example, to create desired compositions we manually assigned high weights to the runner in Figure 2, the theme images in the fashion collages of Figure 8 and the portrait of Obama and 2008 logo in Figure 11. The rest of the images in these collages were scaled randomly.

Then, the ROIs are sorted by their area. After placing the ROIs the user positioned manually, if any, we start from the largest and position the ROIs one-by-one. At each iteration we consider the ROIs already placed as a single ROI. This reduces the problem to placing one new ROI with respect to one other. We compute the composition cost function of Eq. (2), for a subset of possible placements and select the one yielding minimal cost. We limit the computation of composition cost function to placements satisfying the following three conditions: (1) The intersection between the ROIs is not empty. (2) The aspect ratio of the composition must be within ± 0.5 from the desired one. (3) Placements are chosen on a sparse grid (10x10 pixels).

Finally, recalling that the merged ROI is a union of the already placed ROIs, there are multiple layering options for the new ROI. After a placement has been selected, we layer the new ROI such that the informativeness cost function, C_{info} is minimized.

Local refinement: We further refine the assembly via a random sampling process, which improves the collage compactness, informativeness and its visual appearance. Our method is inspired by a Markov chain Monte Carlo (MCMC) methodology and is based on a *hit-and-run* algorithm [SC91]. We adopt an effective random sampling process that reduces the composition cost function by applying random transformations (scale, rotation and translation) to the ROIs and by changing their layering positions.

At each time step we choose uniformly one of the ROIs to



Figure 8: Our view of fashion. Different random initializations produce different collages.



Figure 9: A collage of the wedding of Kate and Ian (Thanks to Rosie Parsons for letting us use these beautiful pictures). A few manual corrections to ROIs were applied at the final phase. That took less than 10 minutes.

be translated by \vec{r} pixels, rotated by θ degrees, and scaled by a factor s . These are sampled from normal distributions: $r \sim N(0, 30)$, $\theta \sim N(0, 5)$, $s \sim N(1, 0.2)$. With probability 0.5 we also change the ROI's layer by uniformly sampling a new layer. We consider only samples where $s \in [0.5, 2]$ and $\theta \in [-30, 30]$ and accept only those that reduce the composition cost function Eq. (2). The sampling is stopped when a cumulative acceptance rate of 5% is reached. Figure 7 illustrates the result.

6. Results

Our proposed algorithm deals well with a broad class of images, taken by professionals and amateurs alike. Figures 1–2, 8–13 illustrate some of our results. Most of the collages were created using the default parameters. The user needs only give the requested aspect ratio for each collage. Since the images are randomly scaled, we created a few different collages for each set of images (as illustrated in Figure 8) and



Figure 10: Parents love their children. The collages of Shelly (left) and the boys (right) are great gifts for their parents. Our survey (Figure 14) shows most people prefer these over the corresponding AutoCollage results.



Figure 11: A collage of 47 images, created by our system, summarizing news highlights of 2008.

selected the one we liked the most. Figure 9 shows another collage of images taken by a professional photographer.

Figure 10 shows two collages of everyday images. They consist of photographs of children taken mostly by their parents. These collages represent the sort of appealing summaries people would like to create of their home photos. To make the collage more interesting, we added to one of them images of Bratz dolls.

Figure 11 shows a collage that summarizes some of the important events of the year 2008 (the US elections, the Olympic games, the economic crisis, some disasters, etc.). This collage is comprised mostly of images of complex scenes with busy, yet informative backgrounds. Despite its unusual busyness (47 images, 47 events), most of the collage pixels are informative and the composition stays visually pleasing and interesting. Our user study (details reported below) showed that about 70% of people preferred this collage to the alternative. In this collage, we manually selected

the Obama images as the most important ones, together with the 2008 text-image.

Limitations: Our system is not perfect, due to its automatic nature. The assembly algorithm always produces an acceptable composition, however at times the user might find the result not pleasing enough, as some of our evaluators found Figure 12(a). This is easily solved by generating a different collage from the same images (Figure 12(b)). The randomness in the system allows to easily create a number of collages, letting the user select the preferred one (this approach was adopted from AutoCollage and Picasa).

Running time: Our algorithm is implemented in MATLAB and runs on a 2.4GHz, dual core desktop computer. ROI extraction takes about 90 seconds per image (500 pixels larger dimension). The initial assemblage takes about 5 seconds to place a pair of images and the refinement (if applied) takes a few minutes.

User study: Since beauty is subjective we evaluated our col-



(a) a less favored collage



(b) a nicer collage generated with a single mouse click

Figure 12: *Limitations*



(a) Our collage



(b) AutoCollage

Figure 13: (a) Our idea of an “advertisement” for the Olympic games. (b) AutoCollage produces a less compact collage of the same images. The survey of Figure 14 shows ~50% preference to our collage (a) over the AutoCollage result (b).

lages via a user study. We published on the web 10 pairs of collages, one created by our system and the other by AutoCollage [RBHB06] (in a random order). For each pair of collages, the participants had to mark the one they liked more or select “can’t decide”.

Figure 13 shows one such example – a collage of the 2008 Olympic games. It demonstrates how the shapes complete each other, just like a puzzle. Conversely, the collage of AutoCollage (Figure 13(b)) places the rectangular regions surrounding the athletes almost on a grid. Our result is more compact due to the elimination of the pixels that do not belong to the ROI and the puzzle-like placement.

160 unbiased people, who were unfamiliar with any of the approaches, participated in the survey. In 8 out of 10 cases our collages were favored. For instance, 103 people liked our collage better and 4 could not decide for the collages in Figure 13. In one case (Figure 12(a)) the AutoCollage composition was significantly preferred (58% vs. 42%), and in one other (“trip”) they had roughly the same score. Figure 14 shows the results; the complete survey (which includes collages that are omitted from the paper) is presented in the supplementary material.

We conclude that our puzzle-like collages would be a welcomed addition to the type of collages one can create today with existing technology.

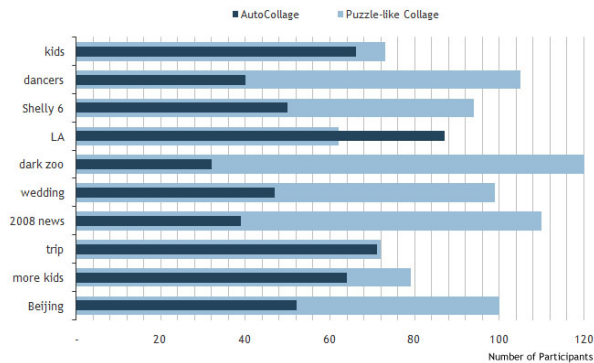


Figure 14: Survey results

7. Conclusion

This paper presented a framework for producing informative and pretty collages of exact cutouts of interesting regions in a puzzle-like manner. The paper makes several contributions. First, a novel region-of-interest (ROI) extraction algorithm is presented. It is shown to extract non-rectangular regions that coincide with the meaningful objects and background boundaries. Second, the paper describes a composition algorithm that places these non-rectangular pieces together. Finally, a system is presented, which can generate collages from everyday as well as professional images.

Our results and user study show that assembling non-rectangular shapes produces compact and pretty summaries of image sets. We have created collages of a variety of different scenarios, including sports event, news, and family photos, which demonstrate the general applicability of the proposed method.

Acknowledgements: We thank the authors of the following software that we've used, for making them public: Google's Picasa, Dirk B. Walther's Saliency toolbox, Xiaodi Hou's spectral residual code, and the trial version of Microsoft's AutoCollage 2008. Thanks to Nicolas Evariste for letting us use his Zoo beautiful images. This work was supported by the Fund for Promotion of Research at the Technion.

References

- [ADA*04] AGARWALA A., DONTCHEVA M., AGRAWALA M., DRUCKER S., COLBURN A., CURLESS B., SALESIN D., COHEN M.: Interactive digital photomontage. *ACM Transactions on Graphics* 23, 3 (2004), 294–302.
- [Ade89] ADES D.: *Photomontage (World of Art series)*. Thames & Hudson, 1989.
- [Atk04] ATKINS C.: Adaptive photo collection page layout. In *ICIP* (2004).
- [BCG*07] BATTIATO S., CIOCCA G., GASPARINI F., PUGLISI G., SCHETTINI R.: Smart photo sticking. In *Adaptive Multimedia Retrieval* (2007), IEEE.

- [CKS97] CASELLES V., KIMMEL R., SAPIRO G.: Geodesic active contours. *Int. J. of Computer Vision* 22, 1 (1997), 61–79.
- [GC04] GIRGENSOHN A., CHIU P.: Stained glass photo collages. In *UIST* (2004), pp. 13–14.
- [HXM*04] HU Y., XIE X., MA W., CHIA L., RAJAN D.: Salient Region Detection Using Weighted Feature Maps Based on the Human Visual Attention Model. *Lecture Notes in Computer Science* (2004), 993–1000.
- [HZ07] HOU X., ZHANG L.: Saliency detection: A spectral residual approach. In *CVPR* (2007), pp. 1–8.
- [IK01] ITTI L., KOCH C.: Computational modelling of visual attention. *Nature Reviews Neuroscience* 2, 3 (2001), 194–204.
- [Joh73] JOHNSON D.: *Near-optimal bin packing algorithms*. PhD thesis, Massachusetts Institute of Technology, 1973.
- [KK01] KONG W., KIMIA B.: On Solving 2D and 3D Puzzles Using Curve Matching. In *CVPR* (2001).
- [LMM02] LODI A., MARTELLO S., MONACI M.: Two-dimensional packing problems: A survey. *European Journal of Operational Research* 141, 2 (2002), 241–252.
- [LRAL08] LEVIN A., RAV-ACHA A., LISCHINSKI D.: Spectral matting. *IEEE Trans Pattern Anal Mach Intell* 30, 10 (2008), 1699–712.
- [LSTS04] LI Y., SUN J., TANG C., SHUM H.: Lazy snapping. *ACM Trans. Graph.* 23, 3 (2004), 303–308.
- [LSZ*07] LIU T., SUN J., ZHENG N., TANG X., SHUM H.: Learning to Detect A Salient Object. In *CVPR* (2007).
- [LXGF05] LI C., XU C., GUI C., FOX M.: Level Set Evolution without Re-Initialization: A New Variational Formulation. In *CVPR* (2005), p. 430.
- [Pic] PICASA: <http://picasa.google.com>.
- [Pol] POLYVORE: <http://www.polyvore.com>.
- [RB80] RADACK G., BADLER N.: *Jigsaw Puzzle Matching Using a Boundary-centered Polar Encoding*. PhD thesis, Graduate School of Arts and Sciences, University of Pennsylvania, 1980.
- [RBHB06] ROTHER C., BORDEAUX L., HAMADI Y., BLAKE A.: Autocollage. *ACM Trans. Graph.* 25, 3 (2006), 847–852.
- [RKB04] ROTHER C., KOLMOGOROV V., BLAKE A.: "Grab-Cut": interactive foreground extraction using iterated graph cuts. *ACM Trans. on Graphics* 23, 3 (2004), 309–314.
- [RKKB05] ROTHER C., KUMAR S., KOLMOGOROV V., BLAKE A.: Digital Tapestry. In *CVPR* (2005), p. 589.
- [SC91] SCHMEISER B., CHEN M.: *General hit-and-run Monte Carlo sampling for evaluating multidimensional integrals*. Tech. rep., School of Industrial Eng., Purdue U., 1991.
- [SKKC00] SEBASTIAN T., KLEIN P., KIMIA B., CRISCO J.: Constructing 2D Curve Atlases. In *IEEE Workshop on Mathematical Methods in Biomedical Image Analysis* (2000), pp. 70–77.
- [Sle80] SLEATOR D.: A 2.5 Times Optimal Algorithm for Packing in Two Dimensions. *Information Processing Letters* 10, 1 (1980), 37–40.
- [VJ01] VIOLA P., JONES M.: Rapid Object Detection Using a Boosted Cascade of Simple Features. In *CVPR* (2001).
- [WAC07] WANG J., AGRAWALA M., COHEN M.: Soft scissors: an interactive tool for realtime high quality matting. *ACM Trans. Graph.* 26, 3 (2007).
- [WSQ*06] WANG J., SUN J., QUAN L., TANG X., SHUM H.: Picture collage. In *CVPR* (2006), pp. 347–354.