

# Efficient Optimization of Photo Collage

Yichen Wei  
Visual Computing Group  
Microsoft Research Asia  
yichenw@microsoft.com

Yasuyuki Matsushita  
Visual Computing Group  
Microsoft Research Asia  
yasumat@microsoft.com

Yingzhen Yang  
State Key Lab of CAD& CG  
Zhejiang University  
yangyingzhen@cad.zju.edu.cn

## Abstract

*Photo collage is an effective representation for photo summarization and visualization. However, its practical usage is limited by its high computational cost. In this paper, we present efficient optimization techniques based on a novel formulation on markov random fields and several insights unexploited by previous approaches. Our method achieves up to hundreds-fold performance improvement, and user/application specific constraints can be easily integrated. We present new applications including interactive collage refinement and dynamic collage for photo browsing. Preliminary evaluation and user study indicates that our approach extends the usability of photo collage.*

## 1. Introduction

With an explosively growing number of digital photos nowadays, photo collage has recently been gaining attention in academia (vision [15], graphics [9], multimedia [16], user interface [2]) and industry [8] to meet the increasing needs for photo browsing/visualization. It is a compact and informative representation of photos arranged on a two-dimensional canvas. Overlap between photos is allowed to better utilize the space. Figure 1 shows some examples.

Given a photo collection, it takes three stages to create a photo collage: 1) *photo selection*: high quality and representative photos are selected to be shown; 2) *saliency computation*: important regions in the photos are identified; 3) *arrangement optimization*: photo arrangement is optimized towards an objective function. Various vision techniques can benefit these tasks, such as photo quality evaluation [7], representative photo selection [11], image saliency analysis [6], face detection [13] and discrete optimization methods [12]. Based on such components, previous methods [15, 9] automatically compute a photo collage by optimizing an energy function formulated with certain objective criteria, e.g., maximizing visible visual saliency. Their results prove to be a better summarization of photos than previous methods [8, 10].

Evaluation of a photo collage could, however, depend on certain subjective criteria or even vary with users. Such criteria usually depends on high level photo content and is hard to be pre-defined or identified, e.g., ‘my photos must be included’, ‘the same face is shown only once’, or ‘my favourite photo should be in the center.’ In general, such goals cannot be realized by optimizing a pre-defined objective function. A natural solution is to involve user interaction [2, 16]. The user study conducted by an automatic system [9] also indicates needs, such as ‘I’d also like to include a specific image.’

An interactive system should be able to 1) incorporate user inputs into the optimization framework, and 2) respond to the user in a short time. An example is shown in Figure 1. Previous approaches [15, 9] are unsuitable for such a task because it is unclear how to incorporate user constraints in their problem formulation without altering the optimization method. The number of photos that could be used is also small given a short time for optimization (around 25 photos takes a few seconds).

This paper presents a novel and efficient optimization framework for photo collage to address the above issues. Our approach is inspired by the recent development of energy minimization techniques on markov random fields (MRFs) [1, 4, 17, 14] and their success for various computer vision problems [3, 5, 12]. Using similar criteria as in [15, 9], the photo collage optimization problem is re-formulated on MRFs. We show that the problem is in general computationally intractable due to its inherent high complexity. Based on several new insights unexploited before, we develop efficient approximate optimization techniques and obtain up to hundreds-fold performance improvement (Figure 5). Consequently, more photos can be easily used (1 second optimization for 50 photos) and user specific constraints can be conveniently integrated.

Two new applications are developed. *Interactive collage* allows a user to refine the result at a responsive time with various actions. *Dynamic collage* creates a continuous browsing experience of an infinite number of sequential photos, e.g., web search images. Preliminary evaluation in-



Figure 1. Three photo collages of the same photo set. Left: result of the method [15]. Middle: result of our approach (pre-computation=3.6 sec, optimization=0.3 sec). Their visual quality is similar, yet such evaluations could be user-dependent, *e.g.*, the user (the girl in the middle) finds it unsatisfactory and wants to move her picture to the center as indicated by the arrow. Right: refined result with our approach in responsive time (0.3 sec).

indicates that these applications extend the usability of photo collage (much faster, easy to edit, able to use more photos).

## 2. Problem Formulation

In this paper, we focus on the arrangement optimization problem. We assume all input photos are of high quality and representative. Unlike previous approaches [15, 9, 16] that are limited to use rectangular regions-of-interest (essentially cropped photos) to represent important regions, we adopt a general model that uses pixel-wise saliency values. Multiple (probably non-equally weighted) important regions (*e.g.*, several faces) can be naturally represented and any saliency computation method can be used. Our implementation of the saliency computation is illustrated in Figure 2.

Given  $N$  input photos, their saliency maps and a two-dimensional canvas, our goal is to compute the optimal configuration  $X = \{x_i\}_{i=1}^N$ , where  $x_i$  is the  $i$ -th photo's state, including position  $p_i$ , rotation angle  $\theta_i$  and layer index  $l_i$  that determines the order of photo placement. The optimization problem is formulated using the following objective criteria. (1) *Overlap*: Overlap between photos is minimized so the visible information is maximized. This introduces a term  $L_i(X)$  that measures the lost saliency of  $i$ -th photo. Saliency information is lost when the photo is either out of canvas or occluded by other photos with higher layers. Therefore we have

$$L_i(X) = C_i(x_i) + O_i(x_i, \{x_k | l_k > l_i\}), \quad (1)$$

where terms  $C$  and  $O$  correspond to the out-of-canvas and occlusion cases, respectively.

(2) *Layer uniqueness*: Each photo has a unique layer index. This introduces a term  $U(X)$  which is  $\infty$  if any two photos have the same layer and 0 otherwise. For technical reasons explained later, it is defined as a set of terms involving all

photo pairs, each of which is a Potts model,

$$U(X) = \sum_{i,j} \delta(l_i - l_j) \cdot \infty,$$

where  $\delta(\cdot)$  is the delta function.

(3) *Angular Diversity* Using diverse rotation angles for adjacent photos is visually appealing and favorable [8, 15]. This is realized by imposing a penalty  $w_d$  for overlapping photos with the same angle, giving rise to the following term

$$D(X) = w_d \sum_{i,j} I(x_i, x_j) \delta(\theta_i - \theta_j), \quad (2)$$

where  $I()$  is 1 if the  $i$ -th and  $j$ -th photos overlap and 0 otherwise. The parameter  $w_d$  is empirically set as a small constant by default, yet can be easily adjusted to obtain different visual effects, as illustrated in Figure 6.

Therefore, our goal is to minimize the energy function

$$E(X) = \sum_{i=1}^N C(x_i) + \sum_{i=1}^N O_i(x_i, \{x_j | l_j > l_i\}) + D(X) + U(X). \quad (3)$$

No user constraints are considered yet because they are hard to be pre-defined. As shown later in sections 5 and 6, our optimization framework is flexible to incorporate certain additional energy terms/constraints to support different application scenarios.

**Challenges and our solution** There are two major challenges in optimizing Eq. (3). Firstly, the occlusion term  $O_i$  is multi-variate and its exact evaluation is very expensive. Previous methods [15, 9] alleviate this problem by using a simple rectangular saliency model. Polygon intersection is used in [15] to evaluate  $O_i$  and a rectangle-packing problem is solved in [9] to avoid direct evaluation. Secondly, the number of variable values is large. Let  $C_p$ ,  $C_r$  and  $C_l$  denote this number for position, rotation and layer parameters,

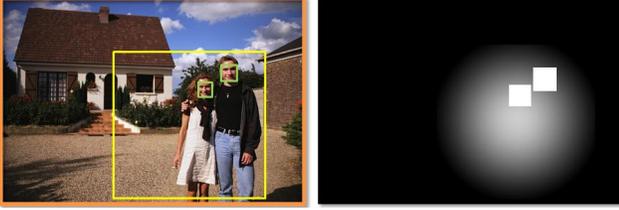


Figure 2. Illustration of our implementation of the per-pixel saliency model. Left: a simplified version of the method in [6] is used to compute a salient region (yellow rectangle). Viola and Jones’s face detector [13] is used to find faces (green rectangle). Right: resulting saliency map where higher intensity indicates larger weight. The underlying parametric form and relative weights are empirically determined and work well in our experiments. The gaussian distribution is used in order to tolerate the inaccuracy and uncertainty of general salient object detection [6].

$C_p$  is typically millions (for a thousands by thousands canvas),  $C_r$  is a few (discretized angles) and  $C_l$  is  $N$  (number of photos). Optimization in such a huge solution space is in general computationally intractable. Previous techniques [15, 9] use effective search strategies (Markov chain Monte Carlo in [15] and divide-and-conquer in [9]) but still have super linear complexity.

By contrast, our approach uses a general saliency model, is much faster and runs in linear time in the number of photos (see Figure 5). It is based on efficient energy minimization methods for MRFs [12] and several problem insights unexploited before.

Minimizing energy functions in the form of Eq. (3) can be justified in terms of maximum a posteriori estimation of MRFs [12]. Powerful energy minimization techniques have been developed in recent years, such as graph cuts [1, 4] and message passing methods [17, 14]. They have been successfully applied to various vision problems, such as stereo/motion estimation and image segmentation/stitching/denoising/super-resolution/inpainting [3, 5, 12]. They prove to be able to obtain strong local optimal solutions very efficiently for energy functions with only unary and binary terms.<sup>1</sup>

A major difficulty when applying such methods is that the term  $O_i$  is multi-variate. Fortunately, it turns out that an approximation with binary terms is reasonable for our problem,

$$O_i(x_i, \{x_k | l_k > l_i\}) \approx \sum_{l_j > l_i} O(x_i, x_j), \quad (4)$$

where  $O(x_i, x_j)$  is the lost information in the  $i$ -th photo occluded by the  $j$ -th photo. Our observation is that, *overlap*

<sup>1</sup>Although methods for high-order markov random fields exist, they do not satisfy our performance goal and are not considered here.

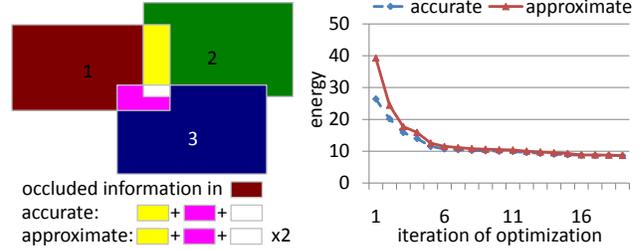


Figure 3. Illustration of approximation in Eq. (4), which introduces errors when the same photo portion is occluded more than once. Left: Photo 1 is occluded by photo 2 and 3. Its occluded information in white region (better viewed in color) is counted twice by Eq. (4). This error is small in a nearly optimal configuration since overlap between photos is also small. Right: plot of accurate and approximate energies during the iterative optimization from a random initialization using 50 photos. The error is large at the beginning and becomes much smaller after a few iterations, showing that Eq. (4) is an asymptotically good approximation.

between photos becomes smaller during an iterative optimization process and this makes Eq. (4) an asymptotically good approximation. This observation is illustrated and experimentally verified in Figure 3.

An approximate energy function with only unary and binary terms is obtained via Eq. (4), but direct optimization is still infeasible, *e.g.*,  $\alpha$ -expansion graph cut [1] needs  $C = C_p \times C_r \times C_l$  iterations and message passing methods [17, 14] result in message vectors of length  $C$ , which are clearly unaffordable. Our approach is inspired by the observation that, *the terms in energy function Eq. (3) are loosely correlated and a natural decomposition gives rise to several smaller and easier sub-problems*. Specifically, the position, rotation and layer parameters constitute three subsets and a sub-problem is defined as minimizing the energy by varying one subset while keeping the other two fixed. The resulting energy function of each sub-problem is greatly simplified, *e.g.*, term  $U(X)$  only appears in layer optimization and  $D(X)$  vanishes in layer optimization.

The three subset parameters are alternately optimized within their own subspaces until the energy cannot be decreased, and a local optima in original space is obtained. The effectiveness of such a strategy heavily depends on how large the subspace of each sub-problem is. Considering an extremely simple case, the well known Iterated Conditional Modes method, it only updates one variable each time and computes a weak local optima since the result is extremely sensitive to the initial configuration, especially for high-dimensional spaces with a huge number of local optima. Our approach computes a strong local optima in the sense that a large subspace is used and all photos are simultaneously updated. The energy decreases fast and converges in a few iterations, as illustrated in Figure 3.

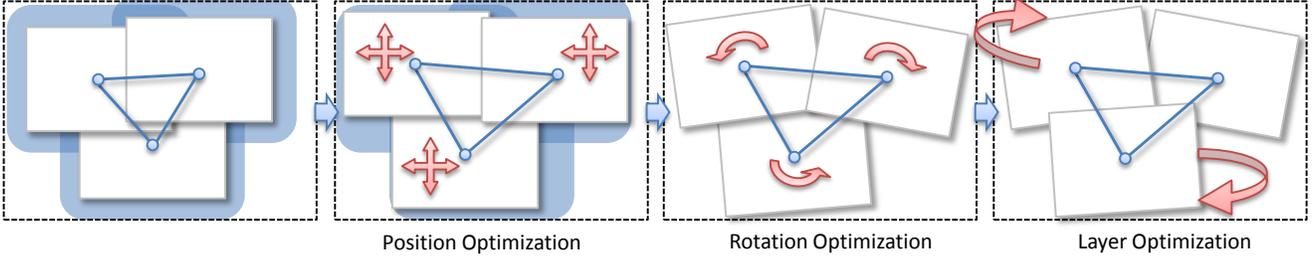


Figure 4. Position, rotation, and layer parameters are optimized in a sequential manner until convergence.

### 3. Optimization

Given an arbitrary initial configuration  $\hat{X}$ , it is updated by alternately optimizing one of the three parameter sets, *i.e.*, position  $X_p$ , rotation  $X_r$  and layer  $X_l$ , while fixing the other two. For example, the position optimization problem is  $\min_{X_p} E_p(X_p) = E(X_p \cup \hat{X}_r \cup \hat{X}_l)$ . The update is terminated until the energy function in Eq. (3)(4) cannot be decreased any more. A graphical model is used to represent the energy function with only unary and binary terms, where each node represents a variable and each edge connecting two variable nodes corresponds to the binary term depending on the variables. The alternate optimization process and graphical models are illustrated in Figure 4.

Among various optimization methods [12], we use loopy belief propagation (LBP) [17] (section 3.1) to optimize position (section 3.2) and rotation (section 3.3) parameters because of its simplicity compared with tree re-weighted message passing [14] and its ability to handle general energy functions compared with graph cuts [4]. Although LBP is also applicable for layer optimization, it turns out that this sub-problem can be effectively solved by a variant of the topological sort algorithm (section 3.4).

#### 3.1. Loopy Belief Propagation

Denoting unary terms as  $S(x_i)$  and binary terms as  $B(x_i, x_j)$ , belief propagation method works by propagating and updating messages  $m_{ij}$  along edges  $(i, j)$  as follows,

$$m_{ij}(x_j) = \min_{x_i} \{S(x_i) + B(x_i, x_j) + \sum_{k \in \mathcal{N}(i) \setminus j} m_{ki}(x_i)\}, \quad (5)$$

where  $\mathcal{N}(i)$  is the set of neighbors of node  $i$ . The marginal of variable  $i$  is estimated as

$$b_i(x_i) = S(x_i) + \sum_{k \in \mathcal{N}(i)} m_{ki}(x_i), \quad (6)$$

where belief  $b_i(x_i)$  roughly states how likely the variable takes value  $x_i$ . Therefore, the MAP(maximum a posteriori) estimate<sup>2</sup> is obtained as  $x_i^{MAP} = \arg_{x_i} \min b_i(x_i)$ .

<sup>2</sup>The terminologies are for probability distributions. They are unambiguously used here since maximizing posterior is equivalent to minimizing

Eq.(5) is known as max-product algorithm. Replacing  $\min$  with  $\sum$  in Eq.(5) gives rise to sum-product algorithm and MMSE(minimum mean-squared error) estimation is obtained. When the underlying graph is acyclic, marginals in Eq. (6) are exactly computed by one pass of message propagation. When the graph contains loops, messages are iteratively updated, known as loopy belief propagation (LBP). Although the convergence is not guaranteed, LBP works quite well for many computer vision problems [3, 12]. Max-product LBP is used in our implementation. We observed that sum-product LBP is slower but achieves similar results.

#### 3.2. Position optimization using LBP

In this step, the energy function  $E_p(X_p) = \sum_i C(x_i) + \sum_{i,j} O(x_i, x_j) + D(X)$  is optimized. Allowing each photo to take all possible locations on the canvas is unfavorable, because this gives rise to binary terms for all photo pairs and a complete graph, which is too expensive. It also causes drastic layout changes that are unfavorable for the applications in sections 5 and 6, where the visual coherency of photo layout should be retained.

Based on the above considerations, the position parameter  $p_i$  is constrained to be within a neighborhood of current position  $\hat{p}_i$ ,  $p_i \in [\hat{p}_i - \Delta^- p_i, \hat{p}_i + \Delta^+ p_i]$ . Since distant photos cannot overlap, the number of binary terms is greatly reduced and the resulting graph is sparse. Let  $\mathcal{G}_i$  be the set of discrete samples in the range  $[-\Delta^- p_i, \Delta^+ p_i]$  and  $p_i \in \{\hat{p}_i + \Delta p_i | \Delta p_i \in \mathcal{G}_i\}$ , the problem becomes to find an optimal value for each  $\Delta p_i$  from  $\mathcal{G}_i$ . LBP is used to solve this discrete optimization problem.

The computational performance depends on the discrete sampling  $\mathcal{G}$ , since message update in Eq. (5) is computed in  $\mathcal{O}(|\mathcal{G}_i| |\mathcal{G}_j|)$  time. Typically  $|\mathcal{G}|$  is large and this step accounts for most of the computation time (more than 95% in the final implementation). Section 4 discusses various speedup techniques for this step, as well as how to set the range parameters  $\Delta^{+(-)} p_i$  and discrete sampling  $\mathcal{G}_i$ .

ing energy in  $-\log$  domain.

### 3.3. Rotation optimization using LBP

Using rotated photos is motivated for aesthetic reasons [8, 15]. Empirical user study shows that excessive rotation degrades the visual quality and a small set of rotation angles  $\{k\Delta\theta\}$  are used. In our implementation, these parameters are empirically set as  $k = [-3..3]$  and  $\Delta\theta = 5$  degrees, yet could be easily adjusted by the user.

LBP is used to minimize the energy function  $E_r(X_r) = \sum_i C(x_i) + \sum_{i,j} O(x_i, x_j) + D(X)$ . Since photos are rotated and only close ones could overlap, the resulting graph is sparse. The number of possible angular values is also small, and this step is very fast (2% – 3% of total time).

### 3.4. Layer optimization using topological sort

In this step, energy function  $E_l(X_l) = \sum_{i,j} O(x_i, x_j) + U(X)$  is minimized. Although LBP could be used, message update in Eq. (5) takes  $\mathcal{O}(N^2)$  time and is inefficient for a large  $N$ . Instead, a variant of the topological sort algorithm is developed based on the observation that, *overlapping photo pairs that define the occlusion terms are fixed during layer optimization, and only relative layer order between such photo pairs is important*. The algorithm is extremely fast and running time is negligible.

A weighted and directed graph is created where each edge  $e = (i, j)$  corresponds to the  $i$ -th and  $j$ -th photos that overlap. Let  $w(i)$  and  $w(j)$  be the summed saliency values in the overlapped area of the  $i$ -th and  $j$ -th photos, respectively, the edge direction  $d_e$  indicates the favored layer order,  $i \rightarrow j$  if  $w(i) < w(j)$ , or  $j \rightarrow i$  otherwise. The edge weight  $w_e = |w(i) - w(j)|$  is the amount of additional saliency loss when the direction  $d_e$  is violated.

If the directed graph is acyclic, all edge directions are satisfied by following a node ordering generated by topological sort, *i.e.*, a node with no incoming edge (in-edge) is always visited before other nodes that have in-edges, and removed from the graph together with all its outgoing edges. While our graph is cyclic, instead of trying to find a node with no in-edge, the node with the smallest summed weights of in-edges is always visited first. Its in-edges are then discarded and the standard topological sort is applied. Consequently, the lost saliency in these discarded in-edges is minimized and the optimal node ordering is generated. The layers are then determined accordingly.

## 4. Fast computation methods

LBP for position optimization is the most expensive step (more than 95% of total time). The bottleneck is the iterative message update in Eq. (5). Various speedup techniques are used for this step, as summarized in Table 1.

**Integral Image** The terms in Eq. (1) are summed saliency values over the intersection of two rectangles, *i.e.*, the  $i$ -th photo and canvas for  $C(x_i)$ , and the  $i$ -th and  $j$ -th photos for

technique	speedup factor	for computing
integral image	hundreds/thousands	$C(x_i)$ , $O(x_i, x_j)$
vectorization	several	vector add/min
Priority BP	faster convergence	
multi-scale grid sampling	thousands(dozens)	$O, I(x_i, x_j)$ (vector add/min)
quotient set for dimension reduction	$\mathcal{O}(n^2)$ , $n$ is number of samples on $x(y)$ axis	$O(x_i, x_j)$ , $I(x_i, x_j)$

Table 1. Summary of speedup techniques in section 4.

$O(x_i, x_j)$ . For upright photos, this intersection is an upright rectangle and the sum is computed in a constant time using an integral image [13]. For rotated photos, new upright rectangular saliency maps are created as the bounding box of rotated original saliency maps, with empty areas filled with zeroes. New integral images are then created and used in the same way. All the integral images are pre-computed according to the discretized rotation angles.

**Vectorization** SIMD instructions (SSE2 in our implementation) are used to exploit the parallelism and obtain a several-fold speed-up for vector add/min computation in Eq.(5).

**Priority belief propagation** It is observed in [5] that firstly propagating more informative messages from nodes that are more confident about its labels gives rise to faster convergence. The sequential message update according to node priority is called *Priority BP*. It is shown to be effective for image completion [5], where node priorities are heuristically determined using node beliefs.

In our problem, it is observed that, *collage update usually starts from a “trigger” photo (by user interaction) and propagates to others*. For example, when a photo is moved, it first squeezes nearby photos to reduce their overlaps. Parallel message update for distant photos is ineffective since they are still in a near-optimal configuration. Priority BP is used whenever a “trigger” photo  $\tilde{I}$  can be identified. The photo priorities are defined according to their distances to  $\tilde{I}$ , that is, nearer photos have higher priorities. It is observed that Priority BP decreases the energy faster than standard BP in the same number of iterations.

**Multi-scale grid sampling** The range parameters  $\{\Delta^{+(-)}p\}$  and the discrete sample  $\mathcal{G}$  are crucial for running performance. A multi-scale strategy is adopted. Let level zero be the finest scale, at level  $k$  the sampling step is set as  $\{s^k\}$  and the range parameters are  $\Delta_k^{+(-)}p_i = (s^{k+1}, s^{k+1})$ . The position parameters are optimized in a coarse-to-fine manner using the solution of the previous level as the initial guess. The complexity of a naive approach at level  $K$  is  $\mathcal{O}(s^{4K+4})$ , which is reduced to  $\mathcal{O}(Ks^4)$  in the multi-scale approach. The sampling step  $s$  and maximum level  $K$  determine how large the photos

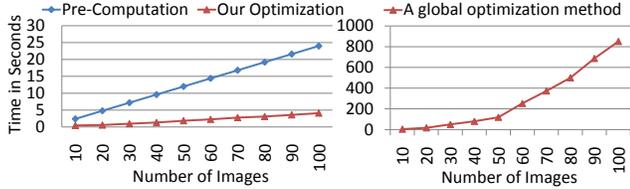


Figure 5. Running time of our approach(left) and the method in [15](right) using different number of images. Our approach obtains dozens- to hundreds-fold speedup, and has linear complexity in the number of photos.

are allowed to move and how many rounds of optimization are needed. After testing several different combinations, we found  $s = 3$  and  $K = 3$  to be a good trade-off. A reasonably large movement can be made in a short time.

**Quotient Set for Dimension Reduction** Naive computation of  $O(x_i, x_j)$  and  $I(x_i, x_j)$  in Eq. (1)(2) takes  $\mathcal{O}(|\mathcal{G}_i||\mathcal{G}_j|)$  time. Based on the observation that, *those terms depend on relative position  $\overrightarrow{p_i p_j}$  instead of absolute positions  $p_i$  and  $p_j$* , the position pair set  $\mathcal{P} = \{(\Delta p_i, \Delta p_j) | \Delta p_i \in \mathcal{G}_i, \Delta p_j \in \mathcal{G}_j\}$  is partitioned into equivalence classes based on the equivalence relation

$$(\Delta p_1, \Delta p_2) \sim (\Delta p_3, \Delta p_4) \text{ iff } \overrightarrow{\Delta p_1 \Delta p_2} = \overrightarrow{\Delta p_3 \Delta p_4}.$$

For each equivalence class  $P$  in the quotient set  $\mathcal{QP} = \mathcal{P}/\sim (P \in \mathcal{QP}, P \subset \mathcal{P})$ , those terms are computed only once for all  $(\Delta p_i, \Delta p_j) \in P$ . The complexity is reduced from  $\mathcal{O}(|\mathcal{G}_i||\mathcal{G}_j|)$  to  $\mathcal{O}(|\mathcal{QP}|)$ . Let  $|\mathcal{G}_i| = |\mathcal{G}_j| = n^2$  ( $n$  uniformly sampled positions in the two axes), it turns out  $|\mathcal{QP}| = \mathcal{O}(n^2)$ . Such a  $\mathcal{O}(n^2)$  (typically around 10) factor saving is significant since the computation of  $O(x_i, x_j)$  accounts for a large portion (about 20%) of the total time.

The quotient set  $\mathcal{QP}$  is uniquely determined by discrete samples  $\mathcal{G}_i$  and  $\mathcal{G}_j$ . It is computed only once for each such sample pair.

**Performance Summary** With the above techniques, the optimization is very fast. As shown in Figure 5, once the pre-computation (of rotated saliency maps and integral images) is done, optimization is performed in a responsive time (0.5 – 2 seconds for 50 photos on a Pentium4 3.2 GHz cpu). Running time is linear in the number of message updates (edges), which is  $\mathcal{O}(N)$  when photos are evenly distributed (there are roughly  $2N$  edges for adjacent cells in a  $\sqrt{N} \times \sqrt{N}$  grid), as experimentally verified in Figure 5.

## 5. Interactive Collage

“Interactive collage” is an application by which a user can refine the result in a responsive time. Various user actions are supported and integrated into the optimization framework. The result is then updated by restarting the optimization. Figures 1, 6 and 7 show several examples.

*Photo add/removal* This is realized by adding/removing corresponding variables in the energy function (Figure 6).

*Photo move* A user drags  $i$ -th photo to a location  $p_i^*$ , implying a positional prior that favors  $p_i$  closer to  $p_i^*$ . This is realized by adding a unary term  $w_p^i \|p_i - p_i^*\|$  into Eq. (3).

The weight  $w_p^i$  is empirically determined as follows. When the drag distance is small (the underlying graph topology is not changed), it is interpreted that the user wants to make a local fine adjustment, and a large  $w_p^i$  is used to realize this intent. When the drag distance is large (the underlying graph topology is changed), it is interpreted that the user wants to make a large scale layout change (Figure 1), and a small  $w_p^i$  is used.

*Send photo to front/back* A common operation is to send an occluded photo to front. This is realized as a hard constraint in the layer optimization. The photo is assigned a fixed layer index (e.g.,  $N$  to be on the top) and removed from the graph during the topological sort optimization.

*Add/modify a salient object* This is necessary when important photo regions are not correctly identified, e.g., a face is miss-detected. This is realized by re-generating the corresponding saliency maps and integral images.

*Others* The user can change parameters (Figure 6) or initialization (Figure 7) to achieve different effects.

**Comparison with previous approaches** As stated, our contribution is a novel optimization framework that handles an objective energy function and subjective user constraints in a consistent and efficient manner. To the best of the author’s knowledge, “interactive collage” is the first application that performs interactive refinement within an energy minimization framework.

There are other interactive collage systems [2, 16]. They are limited to use a pre-defined template canvas that only takes a few photos, because a large number of photos will need a formidable number of templates when different photo sizes/aspect ratios are considered. The arrangement optimization problem is significantly simplified with only a few possible locations. The expressiveness and diversity of a collage representation is therefore sacrificed.

Approaches in [15, 9] also solve energy minimization problems and compute global optimal solutions, at the cost of using simpler saliency models and slower performance, as analyzed before. Nevertheless, it is arguable to claim that those results are ‘best’ from a user’s point of view since such evaluation is highly subjective and user-dependent. The necessity of involving user interaction to make a good photo collage has been well justified in [2, 16] and the user study in [9]. Similarly, we cannot compare those results in Figures 6 and 7 in terms of their energies.



Figure 6. Interactive photo collage. From a random initialization, a user obtains the photo collage result (left) without using the angular diversity term by setting  $w_d = 0$ . The user then removes two photos (indicated by ‘X’) since they are similar to other photos in the collage. He then adds two new photos (indicated by ‘O’) to obtain another result (middle). If more visual diversity is desired, the user can adjust  $w_d$  to obtain another result (right). The whole process is finished in a few seconds.

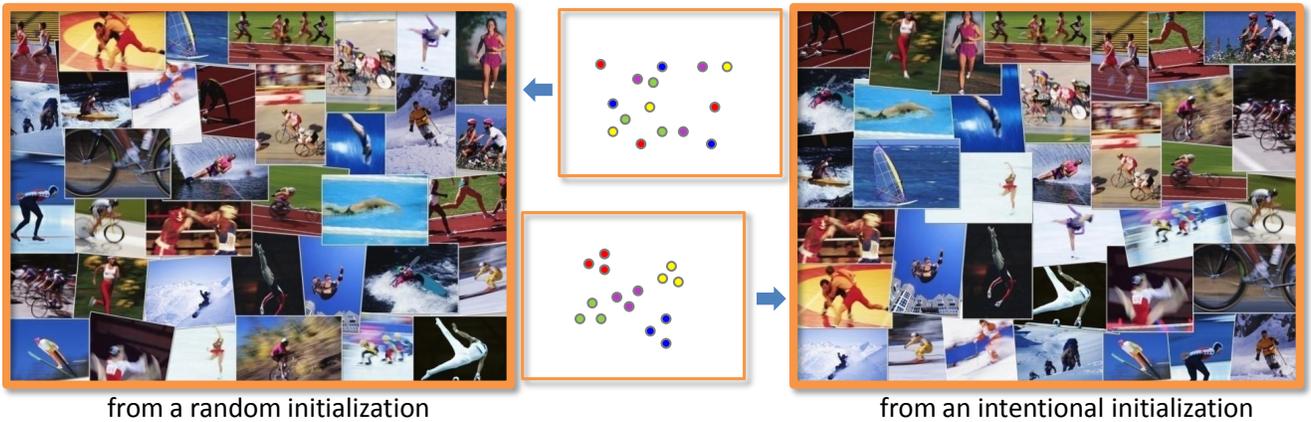


Figure 7. Given a sports photo collection with different categories (‘running’, ‘skating’,...), as indicated by different colors (middle), a user obtains a collage result (left) from a random initialization (middle top). If he favors a different style, *e.g.*, photos in the same category are close to each other, he can easily create such a result (right) from an intentional initialization (middle bottom) where photos in the same category are grouped together and different categories are placed separately (*e.g.*, ‘running’ on the top and ‘bicycle’ on the right). Note that it is easy to create such an initialization if appropriate meta data (*e.g.*, photo’s tagging) are available. It takes a few seconds to generate both results.

## 6. Dynamic Collage

The number of used photos is limited by the spatial dimension of the canvas. The efficient optimization framework naturally exploits the temporal dimension to break the limit. The resulting application, called “dynamic collage”, updates the collage with continuous addition/removal of photos. It achieves a spatially compact and temporally coherent browsing experience of (infinite) sequential input photos, *e.g.*, web search images or streaming video frames. Figure 8 shows an example of dynamic collage application.

Drastic motion during the dynamic update is visually un-

pleasant. Layer parameters are fixed based on photos’ arrival order. Temporal smoothness prior is introduced to retain visual coherency by adding the following terms into Eq. (3) for optimization at the time  $t$ ,

$$\sum_i w_{sr} \|\theta_i^t - \theta_i^{t-1}\| + \sum_{i,j} \mathbf{I}(p_i^{t-1}, p_j^{t-1}) w_{sp} \|\overrightarrow{p_i^t p_j^t} - \overrightarrow{p_i^{t-1} p_j^{t-1}}\|.$$

The first term penalizes severe rotation change and is handled in rotation optimization. The second term encourages the relative position of two overlapped images to be stable and prohibits large movement. This term is handled in the

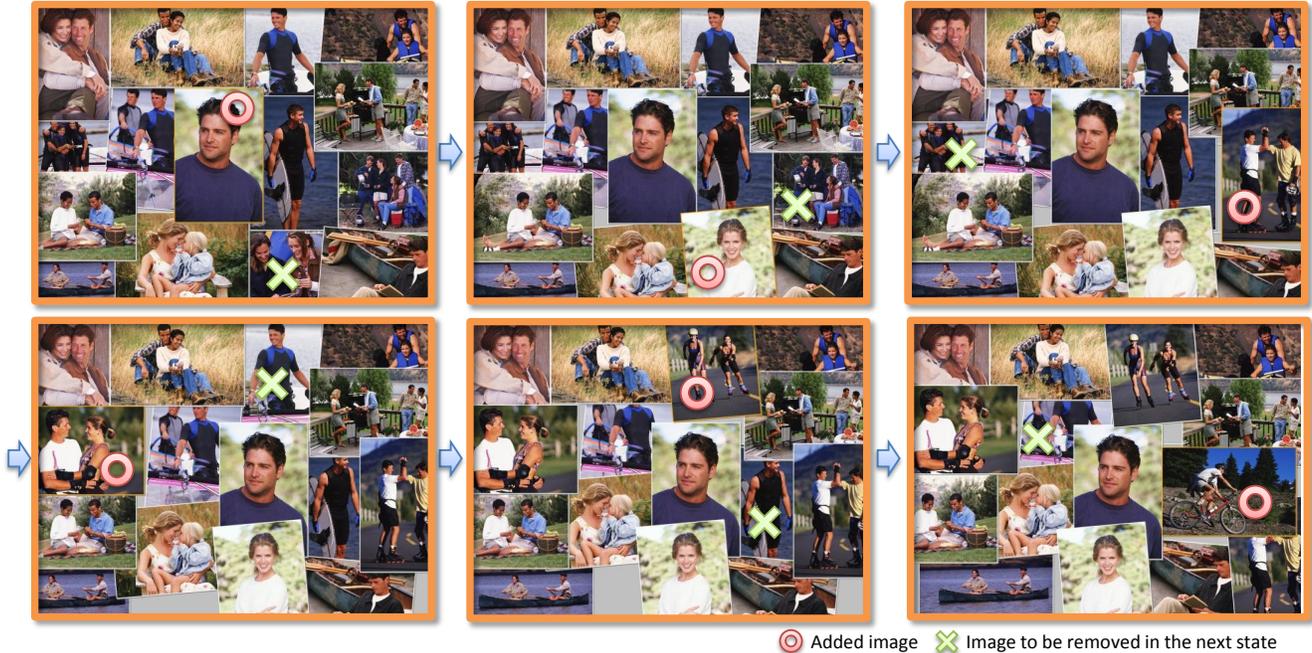


Figure 8. Dynamic photo collage achieves temporally coherent and spatially compact image browsing experience. Photo collage is dynamically updated by replacing old images with new ones and image transitions are smoothly rendered.

position optimization step and computed in a similar way as the occlusion term. The weights  $w_{sr}$  and  $w_{sp}$  are empirically set as small constants.

**User Study** A user study is performed to evaluate the effectiveness of dynamic collage (DC) representation for photos. Two other approaches are used for comparison: a simple slideshow (SS) where photos are shown one by one, and a slideshow of collages (SC) where sequential photo collages are shown one by one. Three videos with the same length are created using the three methods to display a hundred high quality photos of buildings, natural scenes, and people. Seventeen users were asked to watch the three videos in a random order and answer the following questions with a score from 1 (definitely no) to 5 (definitely yes).

**Q1.** Is the presentation visually pleasing? DC(3.9), SC(3.3), SS(2.4)

**Q2.** Is the display layout good? DC(3.4), SC(3.3), SS(not applicable)

**Q3.** Are you willing to use it, *e.g.*, as a screen saver? DC(4.0), SC(3.6), SS(3.1)

Results show that dynamic collage makes a noticeable improvement over the other two methods and is potentially useful for photo browsing/visualization tasks.

## 7. Discussions

Our approach is based on a novel formulation on markov random fields and several problem specific insights. It is

tuned to achieve a high performance and improve the usability of photo collage. The underlying ideas for approximate optimization, however, could be useful for other problems, *e.g.*, removal of complicated but unimportant factors in the energy function and decomposition of a large problem into several small ones. The formulation of various user specific constraints and prior terms would be helpful for other interactive systems and photo based applications.

This paper is focusing on the algorithmic aspect (arrangement optimization stage) instead of developing a comprehensive new system. Such a task would require systematic investigation of all three stages and a thorough user study considering various options. This is beyond the goal of this paper.

## References

- [1] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001. 1, 3
- [2] N. Diakopoulos and I. Essa. Mediating photo collage authoring. In *Proceedings of ACM symposium on User interface software and technology*, pages 183–186, New York, NY, USA, 2005. ACM. 1, 6
- [3] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael. Learning low-level vision. *International Journal of Computer Vision*, 40(1):25–47, 2000. 1, 3, 4

- [4] V. Kolmogorov and R. Zabini. What energy functions can be minimized via graph cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2004. 1, 3, 4
- [5] N. Komodakis and G. Tziritas. Image completion using global optimization. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, 2006. 1, 3, 5
- [6] T. Liu, J. Sun, N. N. Zheng, X. Tang, and H.-Y. Shum. Learning to detect a salient object. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, 2007. 1, 3
- [7] Y. Luo and X. Tang. Photo and video quality evaluation: Focusing on the subject. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2008. 1
- [8] Picasa. <http://picasa.google.com/>. 1, 2, 5
- [9] C. Rother, L. Bordeaux, Y. Hamadi, and A. Blake. Autocollage. In *Proceedings of ACM SIGGRAPH*, 2006. 1, 2, 3, 6
- [10] C. Rother, S. Kumar, V. Kolmogorov, and A. Blake. Digital tapestry. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, 2005. 1
- [11] I. Simon, N. Snavely, and S. M. Seitz. Scene summarization for online image collections. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2007. 1
- [12] R. Szeliski, R. Zabini, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother. A comparative study of energy minimization methods for markov random fields. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2006. 1, 3, 4
- [13] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, 2001. 1, 3, 5
- [14] M. Wainwright, T. Jaakkola, and A. Willsky. Tree-reweighted belief propagation algorithms and approximate ml estimation via pseudo-moment matching. *AISTATS*, 2003. 1, 3, 4
- [15] J. Wang, J. Sun, L. Quan, X. Tang, and H.-Y. Shum. Picture collage. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, 2006. 1, 2, 3, 5, 6
- [16] J. Xiao, X. Zhang, P. Cheatele, Y. Gao, and C. Atkins. Mixed-initiative photo collage authoring. In *Proceedings of ACM Multimedia*, 2008. 1, 2, 6
- [17] J. S. Yedidia, W. T. Freeman, and Y. Weiss. Understanding belief propagation and its generalizations. pages 239–269, 2003. 1, 3, 4