In this paper, a new algorithm is proposed for image re-ranking in web search applications. The proposed algorithm introduces a new layout expectation model for improving the image search results. The motivation for using the expectation model is that users may often have potential expectations about the desired image during the search process. By including the layout expectation model to describe users’ expectation on image layouts, the re-ranked search results can become more satisfactory to users. Experimental results demonstrate that our proposed algorithm can significantly improve the re-ranking precision compared with the state-of-the-art algorithms.

Index Terms—Layout-expectation-based model, CT, Image re-ranking

1. INTRODUCTION AND RELATED WORKS

Nowadays, with the prosperity of the Internet, web image search gains more and more popularities [1-7,10]. However, the returned image results from the existing mainstream search engines such as Google and Bing are still unsatisfactory with various query-irrelevant images or visually unappealing ones. Therefore, it is still a challenging task to improve the search result precision for web image search applications.

Many algorithms have been proposed for improving image search precisions [1-7]. Fergus et al. [1] proposed to use the object class model to filter the output of image search engines when searching object-based images. Berg and Forsyth [2] developed a visual-information-based system to collect a large number of animal pictures from the web. However, their approaches need to build different models towards specific queries and categories, thus are not applicable in daily web search.

Other works [4-7] introduced more generalized consistency- or similarity-based models for improving image search precisions. Their works are based on the assumption that images more related to the query are often visually similar and should be consistent to each other. However, due to the large variance of image contents, using the consistency model alone still has great limitations and other models are needed for obtaining more sophisticated results.

As users may often have potential expectations about the desired image during a search, if these expectations can be described or predicted properly, the search results can be greatly improved. Some works have already tried in this trend. Cao et al. [6-7] and Gao et al. [9] provided interactive interfaces for users to draw their expectations for improving image search results. However, these methods require users’ manual interaction during the search which is not always convenient in many scenarios. Lu et al. [3] inferred users’ image search goals according to the click-through records. However, their method is more focused on analyzing polysemous queries rather than re-ranking images according to their relevance to user expectations. Furthermore, in Huang et al.’s work [12], a new visual saliency model was introduced for re-ranking images. Based on the observation that users are more favorable for salient images than cluttered ones, the algorithm uses a saliency model for evaluating each image. By combining this saliency model and the consistency model, the re-ranked results can be greatly improved [12]. However, although this saliency model can partially predict user expectations, it still cannot effectively describe user expectations as this saliency model only evaluate images from the “attractiveness” point of view. Thus, it still creates less satisfactory results in many scenarios.

In this paper, we propose a new layout expectation model to re-rank the images returned from mainstream search engines. The contribution of this paper can be summarized as follows: (1) We propose a new layout-expectation-based model (LEM) which can properly describe users’ expectation on image layouts; (2) Based on the layout expectation model, we further propose a new image re-ranking algorithm (LEA) that can re-rank images according to user expectations without users’ manual interaction. Experimental results demonstrate that our proposed algorithm can significantly improve the re-ranking precision compared with the state-of-the-art algorithms.

The rest of the paper is organized as follows: Section 2 describes the motivation of our proposed layout expectation model. Section 3 describes the details of using our layout-expectation-based model for image re-ranking. Section 4 shows the experimental results and Section 5 concludes the paper.

2. MOTIVATION OF THE LAYOUT EXPECTATION MODEL

As mentioned, users may often have potential expectations about their desired image during search (e.g. layout, color, and saliency [12]). For example, when a user types the query “ocean”, he/she may actually expect to get images that have a layout with the ocean on one side and the seashore or sky on the other. In another example, when a user types the word “office” as the query, he/she may expect to get an indoor image including tables and chairs. Therefore, if these expectations can be properly modeled, the search results can be greatly improved by ranking those better-expectation-matching images with higher orders.

Based on the observation that image layouts are one of the
most important factors that affect user expectations, we will focus on the layout expectation in this paper and propose a new layout-expectation-based model (LEM) for modeling user layout expectations. Furthermore, we also propose a layout-expectation-based algorithm (LEA) for re-ranking images based on LEM. The proposed LEM and LEA are described in the following section.

It should be noted that our LEA differs from the previous sketch-based image search algorithms [6-7] in the following two parts: (a) The sketch-based search algorithms require users’ manual interference while our LEA performs re-ranking in an automatic way; (b) In some cases, the users only have rough expectations about the image layout that cannot be clearly sketched. In this case, the sketch-based algorithms may fail to work while our algorithm can still work properly, as will be shown in the experimental results.

### 3. THE LAYOUT-EXPECTATION-BASED MODEL AND ALGORITHM

#### 3.1 The layout-expectation-based algorithm

The framework of our layout-expectation-based algorithm (LEA) can be described by Fig. 1.

![Fig. 1. The framework of the layout-expectation-based algorithm.](image)

In Fig. 1, for an input query, the original returned image search results are first obtained from the public search engines such as Google or Bing. Then the input query and the original returned images are used to achieve the layout expectation description of the input query. At the same time, the layout descriptions for all original returned images are also calculated. Finally, the layout descriptions of the original returned images are compared with the layout expectation description for the input query to create the final re-ranked results, where images better match the query’s layout expectation description will be placed at higher orders.

From Fig. 1, we can see that achieving the layout expectation for the input query is one of the key parts in our algorithm. In this paper, we propose the following two strategies for achieving the query layout expectation.

(a) Take the top-ranked images from the original returned results and extract the most common layout as the expected layout for the input query (named as the TOP strategy in this paper).

(b) Extract and cluster the pseudo-images [3] from users’ click-through log history for the query (i.e., the combination of images clicked by the previous users when searching the same query). And then select images in the click-through logs which are around the center of the major cluster as the estimated user layout expectation for the query (i.e., the PSEUDO strategy).

After the layout expectation image is selected for the query, we need suitable feature vectors to describe its layout as well as the layouts of the original returned images. Furthermore, we also need proper dissimilarity metrics for comparing the layout differences between the query expectation image and the original returned images for re-ranking. These two parts are included in our layout-expectation-based model (LEM). It is described in the following section.

#### 3.2. The layout-expectation-based model

Our LEM mainly includes two parts: (a) the feature vector to describe the layouts of images; and, (b) the dissimilarity metric for comparing the layout dissimilarities between images.

##### 3.2.1. The feature vectors to describe image layouts

Since image layouts refer to the arrangement of different parts or objects on images, we observe that the following properties are important for describing image layouts: (a) the edge distribution in the image. It’s been proven that edge distribution plays the most important role for people to identify an object in images while people are generally insensitive to the detailed pixel values [8]; (b) the naturalness of the image (i.e., the degree that the image is from natural scene rather than man-made scene). The spatial layout of a scene strongly differs between man-made scene and natural scenes. Straight horizontal and vertical lines dominate man-made scenes while most natural landscapes have textured zones and undulating contours [10]; (c) the openness of the image. The existence of a horizon line and lack of visual references confer to a scene with high degree of openness and the degree of openness may decrease with the increase of boundary elements; and, (d) the roughness of the image. Roughness of a scene mainly refers to the size of its major components. So roughness is highly correlated with the fractal dimensions of the scene and thus, its complexity [10]. In all, in our LEM, we propose to introduce the following feature vector to describe image layouts:

$$\vec{V} = \left[ \tilde{C}, F_{na}, F_{op}, F_{ro} \right]$$  \hspace{1cm} (1)

where $\tilde{C}$ is the CENsus Transform hISTogram (CENTRIST) vector for describing the edge distribution of an image [8]. $F_{na}$, $F_{op}$, and $F_{ro}$ are the features describing the naturalness, openness, and roughness of the image, respectively. $\tilde{C}$ is calculated by Eqn. (2):

$$\tilde{C} = \left[ H_{1,1}, H_{1,2}, ..., H_{m,n} \right]$$  \hspace{1cm} (2)
where $H_{ij}$ is the Census Transform (CT) value histograms for block $i \times j$. The CT value for each pixel in the block can be calculated as in Fig. 2. In Fig. 2, CT mainly compares the intensity of a pixel with the eight neighboring pixels, and each comparison result encodes one bit in the final CT value for this pixel [8]. The eight bits are then combined together in a certain order, like from left to right, to form the final CT value. Normally, image patches with similar CT histograms are expected to have similar structural layouts. And by concatenating all blocks of an image, the CENTRIST feature $c^r$ can reflect the global structural layout of the image.

\[
\begin{array}{c|ccc}
32 & 64 & 96 & 110 \\
32 & 64 & 96 & 110 \\
32 & 32 & 96 & 110 \\
\end{array} \Rightarrow (11010110)_2 \Rightarrow \text{CT} = 214
\]

Fig. 2. The process for calculating CT.

The naturalness value $F_{na}$ and the roughness $F_{ro}$ in Eqn. (1) can be calculated by Eqn. (3) and (4), respectively.

\[
F_{na} = \sum_{i=L}^{L} (f_i^{\text{diag1}} + f_i^{\text{diag2}} - f_i^{\text{ver}} - f_i^{\text{hori}})
\]  
(3)

\[
F_{ro} = \sum_{i=L}^{L} (w_i^{\text{hori}} \cdot f_i^{\text{hori}} + w_i^{\text{ver}} \cdot f_i^{\text{ver}})
\]  
(4)

where $f_i^{\text{hori}}$, $f_i^{\text{ver}}$, $f_i^{\text{diag1}}$, $f_i^{\text{diag2}}$ are the pixel values of the frequency-domain image (i.e., the Fourier transformed image) in the horizontal, vertical, and the two diagonal directions. $L$ is the number of points that are included in calculating the feature values, as in Fig. 3 (a). $w_i^{\text{hori}}$ and $w_i^{\text{ver}}$ are the corresponding weights from a predesigned roughness spectral template where the white pixels mean positive effects (i.e., $w \geq 0$) and the dark ones stand for negative effects (i.e., $w \leq 0$), as in Fig. 3 (b) [10].

Finally, the openness $F_{op}$ in Eqn. (1) can be calculated by:

\[
F_{op} = \frac{B_{cen}}{B_{peri}}
\]  
(5)

where $B_{cen}$ is the total number of boundary pixels in the center area of the edge image (i.e., the Sobel filtered image in this paper), and $B_{peri}$ is the number of boundary pixels in the peripheral area.

As mentioned, the features used in our LEM are all ‘statistical’ features. That is, they do not correspond to specific layouts. Rather, they only reflect the statistical properties of the image layouts, such as the histogram of edge distribution ($c^r$) and the naturalness measure of the frequency-domain components over different directions ($F_{na}$). This ‘statistical’ characteristic has the following advantages:

(a) It allows multiple queries to share the same layout description model such that only a small number of layout classes need to be predefined in the layout expectation dataset. For example, although queries such as “office”, “bedroom”, or “kitchen” have different detailed layouts, their ‘statistical’ layouts are close to each other since they all belong to indoor scenes. Thus, they can be classified as the same class and share the same layout expectation model.

(b) It also allows “fuzzy” re-ranking. In many scenarios, it is very difficult to figure out a clear layout expectation for the input query. For example, the layout of the query “mountain” can vary with different mountain shapes and topographies. By using ‘statistical’ features, our LEA can still produce satisfactory results even when a clear layout expectation is not available. This will be further demonstrated in the experimental results.

3.2.2 The dissimilarity metric for comparing the layout dissimilarities between images

Based on the proposed layout-description feature vector in Eqn. (1), we further propose a new dissimilarity metric for comparing the layout dissimilarities between images. The proposed dissimilarity metric is described in Eqn. (6):

\[
S_{\text{LEA}}(q,k) = -\alpha_1 \cdot \text{HI}(\tilde{C}(q), \tilde{C}(k)) + \sqrt{\alpha_2 \cdot \text{sim}_{na}^2(q,k) + \alpha_3 \cdot \text{sim}_{ro}^2(q,k) + \alpha_4 \cdot \text{sim}_{op}^2(q,k)}
\]  
(6)

where $S_{\text{LEA}}(q,k)$ is the layout dissimilarity between the layout expectation image $q$ for the query and the returned image $k$. $\text{HI}()$ is the Histogram Intersection [8] result between the two CENTRIST vectors for $q$ and $k$. $\alpha_1$, $\alpha_2$, $\alpha_3$, $\alpha_4$ are weighting factors balancing the relative importance among the different features. These weighting factors are achieved in the following way: we first define several query classes. Then, for each class, an independent set of weighting factors are trained based on the train data [11]. Finally, in the re-ranking process, each input query will be first classified into one of these classes. And then, the corresponding weighting factor set will be used for re-ranking.

Furthermore, $\text{sim}_{na}(q,k)$, $\text{sim}_{ro}(q,k)$, and $\text{sim}_{op}(q,k)$ in Eqn. (6) are the dissimilarities for naturalness, roughness, and openness features between $q$ and $k$. They are calculated by:
\[ \text{sim}_j(q,k) = abs(1 - F_j(q) / F_j(k)) \] (7)

where \( j = na, ro, \) or \( op. \)

Based on our LEM, for each input query, we can first achieve its layout expectation image \( q. \) Then, after calculating the layout dissimilarity metric \( S(q,k) \) between the returned images and \( q, \) the re-ranking can be performed by ranking images with smaller dissimilarity values at higher places. Furthermore, it should also be noted that our proposed LEM can also be easily combined with other expectation models for further improving the results. For example, we can further combine LEM with the saliency and consistency models \([12]\) to form a saliency-consistency-layout-expectation-based \((S+C+LEA)\) re-ranking algorithm. This \( S+C+LEA \) algorithm is described in the following section.

3.3. The saliency+consistency+layout-expectation-based \((S+C+LEA)\) re-ranking algorithm

The \( S+C+LEA \) re-ranking algorithm can be described by Eqn. (8):

\[ R(\hat{X}) = W(\hat{V}_{LEA}, \hat{V}_{SAL}, \hat{V}_{CON}) \] (8)

where \( \hat{X} = \{1,2,\ldots,N\} \) is the set for all of the \( N \) original returned images and \( R(\hat{X}) \) is the image re-ranking results by the \( S+C+LEA \) algorithm. \( W(\cdot) \) is the random walk process \([4]\).

\( \hat{V}_{LEA} = [S_{LEA}(q,1), S_{LEA}(q,2), \ldots, S_{LEA}(q,N)] \) is the layout-expectation vector where the element \( S_{LEA}(q,k) \) is calculated by Eqn. (6).

\[ \hat{V}_{SAL} = [\hat{V}_{SAL}(1), \hat{V}_{SAL}(2), \ldots, \hat{V}_{SAL}(N)] \] is the saliency vector for the original returned images and \( \hat{M}_{CON} = [m(k,j)]_{N \times N} \) is the consistency matrix among images. The saliency value \( \hat{V}_{SAL}(k) \) for image \( k \) and the consistency value \( m(k,j) \) between images \( k \) and \( j \) can be calculated by Eqn. (9) and Eqn. (10), respectively:

\[ V_{SAL}(k) = \sum_{s} \sum_{t} \ln \left( \frac{P(\sigma_s(k) \mid l_{sal})}{P(\sigma_s(k) \mid l_{clut})} \right) \] (9)

\[ m(k,j) = \frac{l}{U} \sum_{u=1}^{U} \sigma_u(k,j) \] (10)

where \( \sigma_s(k) \) is the feature vector for image \( k \), and \( s \) and \( t \) denote the feature and scale indexes, respectively. \( l_{sal} \) and \( l_{clut} \) are labels of salient and cluttered images \([12]\). \( P \) is the estimated saliency values from the trained Support Vector Machine (SVM) models \([12]\). \( U \) is the total number of features, \( \sigma^* \) is the consistency variance for feature \( u \), and \( s(k,j) \) is the dissimilarity (or consistency) between \( k \) and \( j \) for the \( u \)-th feature \([12]\).

From Eqn. (8), we can see that our \( S+C+LEA \) algorithm utilizes a random walk to combine the saliency, consistency, and layout-expectation models. By this way, the re-ranking results can be further improved, as will be shown in the experiments.

4. EXPERIMENTAL RESULTS

In this section, we show experimental results for our proposed algorithm. The original returned images are collected from Yahoo and Google search engines for different queries. For each query, we collect 200 images and a total of 55 queries are considered. All the images returned from the search engine are resized to \( 128 \times 128 \). For CENTRIST, each image is divided into \( 8 \times 8 = 64 \) blocks with \( 16 \times 16 = 256 \) pixels in each block. \( L \) is set to be 64 in Eqn. (3) and (4). When calculating \( F_{op} \) in Eqn. (5), we also define the size of the center area as the \( 80 \times 80 \) block around the image center. Four methods are compared:

(a) The original image search results returned by Google or Yahoo (Ori in Table 1).

(b) The re-ranking results by using the saliency+consistency model \([12]\) (S+C in Table 1).

(c) The re-ranking results by our layout-expectation-based algorithm (LEA in Table 1).

(d) The re-ranking results by combining our layout-expectation-based model with the saliency and consistency models \((S+C+LEA)\) in Table 1.

Fig. 4 and Fig. 5 show the top-seven re-ranked images for the above four methods, where each row corresponds to the re-ranking results for method Ori, S+C, LEA, and S+C+LEA, respectively. In Fig. 4 and Fig. 5, the layout expectation for each query is achieved by the TOP strategy.

From Fig. 4, we can see that the original search results are not satisfactory with some query-irrelevant images. Although the \( S+C \) model can improve the searching results, some images still cannot match users’ expectation. For example, the 5th and the 6th images of the second row in Fig. 4 (a) are barely relevant to the query “waterlily”, and the 6th image of the second row in Fig. 4 (b) is more relevant to seashore instead of the query “island”. Compared to these, by including our LEA, the results of our \( S+C+LEA \) method are significantly improved. This further demonstrates the effectiveness of our proposed LEA model.

Fig. 5 shows the results of another two queries for the methods Ori, S+C, LEA, and S+C+LEA respectively. In Fig. 5 (a), the re-ranking results from the \( S+C \) model for query “office” are quite unsatisfactory. This is because the saliency-based model pays more attention to image saliency so that images with clean backgrounds tend to be ranked higher. However, in fact, for “office”, users are often expecting a complicated layout with desks and chairs everywhere. Thus, the \( S+C \) model will fail to work well under these queries. On the contrary, the results of our LEA are significantly improved by evaluating images based on the predicted layout expectation. Similarly, in Fig. 5 (b), since people may expect some complicated curves or layouts for “mountain” images, using \( S+C \) model produces less satisfactory results while LEA can provide much improved results. Furthermore, we can also see that in Fig. 5 (a), the results by our LEA algorithm are obviously better than the combined \( S+C+LEA \) method. This is because the re-ranking results by the \( S+C \) models are extremely
poor for query “office”, thus decreasing the results of the combined S+C+LEA method. However, the S+C+LEA results in the 4th row are still improved from the S+C results in the 2nd row by containing more query-relevant images. This also implies that our proposed LEA model can help reducing the adverse impacts from the other re-ranking models in cases when they fail to work.

Furthermore, Fig. 6 compares the results of Ori, S+C, LEA with the TOP layout expectation strategy, and LEA with the PSEUDO layout expectation strategy, respectively. From Fig. 6, we can see that the S+C model cannot efficiently re-rank the images. Comparatively, our LEA method with the TOP strategy can greatly improve the re-ranking performance from the S+C model. However, since the TOP strategy estimates the layout expectation only from the information of the current search results, the estimated layout may not be the most representative, thus may sometimes create less satisfying results (e.g., the 4th image in the third row of Fig. 6). Compared with the TOP strategy, since the PSEUDO strategy estimates user layout expectations from the more reliable user click-through log information [3], it is able to achieve more representative user layout expectations. Thus, it can achieve the best results.

In order to further demonstrate the effectiveness of our proposed algorithm. We also compare the objective average precision (AP@40) value [3, 12] for different re-ranking algorithms. Part of AP@40 values for different queries are shown in Fig. 7. The Mean Average Precision (MAP) over 55 queries is also shown in Table 1. From Fig. 7 and Table 1, we can see that the S+C method, the LEA method, and the S+C+LEA method can achieve better search precision than the original results. However, for many of the complex scene queries such as “office” and “desert”, the S+C method could not achieve satisfactory results. Comparatively, our proposed LEA method can achieve satisfactory results on most of the queries. Furthermore, by combining our LEA model with the S+C models, the MAP values can be further improved.
Table 1. Mean Average Precision (MAP) for different methods

<table>
<thead>
<tr>
<th></th>
<th>Ori</th>
<th>S+C</th>
<th>LEA</th>
<th>S+C+LEA</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAP@40</td>
<td>0.53</td>
<td>0.71</td>
<td>0.84</td>
<td>0.89</td>
</tr>
</tbody>
</table>

(a) The input query image

(b) Comparison of the top-seven re-ranked results between the original search results from Google [13] (the first row) and the re-ranking results by our LEA model (the second row).

Fig. 8. The results for image-based image search.

Furthermore, since our proposed LEA model effectively encodes the spatial layouts of the images with statistical properties, our model is very suitable to be used in the application of image-based image search. In this application, users input an image instead of a keyword as the input query and aim to find similar or relevant images to the query image. Since the input query image already includes the layout expectation of the users, in our LEA model, we can simply compare the layout dissimilarities between the query image and the original returned images for performing re-ranking. Fig. 8 shows one experimental result by using our LEA model for re-ranking image-based image search results. In Fig. 8, (a) is the input query image, the first row in (b) is the original returned images by Google Similar Images [13], and the second row in (b) is the re-ranking results by our proposed LEA model. From Fig. 8, it is obvious that the re-ranking results by our LEA model are more relevant to the input query images.

5. CONCLUSION

In this paper, we proposed a new layout-expectation-based algorithm for images re-ranking. By introducing the layout expectation model to describe users’ expectation on image layouts, the re-ranked search results can become more satisfactory to users. Experimental results demonstrate the effectiveness of our algorithm.

6. ACKNOWLEDGEMENTS

This work is supported in part by the following grants: National Science Foundation of China (61001146, 61025005, and 61101147), Chinese national 973 grants (2010CB731401), the Open Project Program of the National Laboratory of Pattern Recognition (NLPR), and SMC scholarship of SJTU.

7. REFERENCES