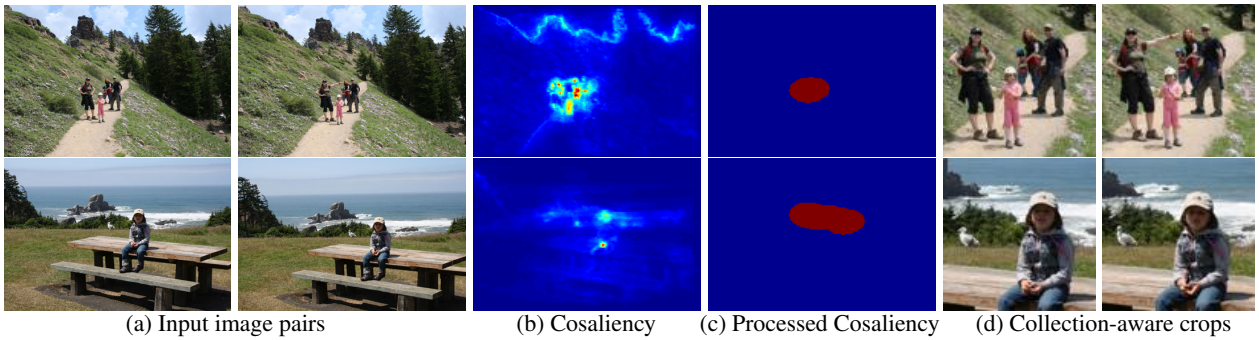
Cosaliency: Where People Look When Comparing Images

David E. Jacobs
Stanford University
353 Serra Mall
Stanford, CA, USA
dejacobs@cs.stanford.edu

Dan B Goldman *Eli Shechtman*
Adobe Systems
801 North 34th Street
Seattle, WA, USA
{dgoldman, elishe}@adobe.com



(a) Input image pairs (b) Cosaliency (c) Processed Cosaliency (d) Collection-aware crops

An overview of our algorithm and its results for two pairs of images. From left to right: Standard thumbnails for the input image pair, our calculated model for image cosaliency, its processed version and our automatically generated collection-aware crops. Note that small image features like the position of the woman's arm or the angle of the bird's head are nearly impossible to see using standard thumbnails alone.

ABSTRACT

Image triage is a common task in digital photography. Determining which photos are worth processing for sharing with friends and family and which should be deleted to make room for new ones can be a challenge, especially on a device with a small screen like a mobile phone or camera. In this work we explore the importance of local structure changes–e.g. human pose, appearance changes, object orientation, etc.–to the photographic triage task. We perform a user study in which subjects are asked to mark regions of image pairs most useful in making triage decisions. From this data, we train a model for image saliency in the context of other images that we call *cosaliency*. This allows us to create *collection-aware* crops that can augment the information provided by existing thumbnailing techniques for the image triage task.

ACM Classification: H5.2 [Information interfaces and presentation]: User Interfaces. - Graphical user interfaces.

General terms: Algorithms, Experimentation, Human Factors

Keywords: Cosaliency, automated thumbnailing, saliency, collection-aware cropping.

INTRODUCTION

Designing user interaction for small or mobile displays is an ongoing challenge in the human computer interaction community. One task for which this problem is particularly acute is the comparison of high resolution image data, a very common task in digital photography [21][10]. During a day of shooting a photographer is often forced to perform photographic triage: Which photos should be saved and processed for sharing with friends and family? Which photos should be deleted to make room for new ones? Do I need to take another shot? In the field, the only feedback a photographer has to help answer these questions is the small LCD panel on the back of his or her camera or other mobile device. Though this offers a vast improvement in usability over the days of film photography, it is still difficult to effectively convey the high resolution image data on the low resolution display. This problem is doubly true when comparing multiple images simultaneously.

We informally discussed this idea with a variety of amateur photographers and asked them what kinds of factors they consider important to image triage. Common responses focused on either low-level image quality concerns such as image noise, focus, motion blur, exposure or abstract high-level issues such as artistic composition composition. However, many mid-level concerns such as human pose, facial expression, object orientation, parallax occlusions and dis-occlusions, and appearance changes were also raised. We broadly categorize these issues as *local structure*.

For many of these concerns, there exist simple ways to effectively visualize and compare these features across images. For example, the image histograms provided by many cameras allow one to quickly determine which photograph is better exposed. Composition is also easily evaluated by simply viewing the standard thumbnails (scaled down versions) for each image. Even noise and blur can usually be seen by zooming in to full resolution at an arbitrary image location. Change in local structure, however, can be difficult to compare because it is not always obvious from a thumbnail where to zoom in.

In this paper, we develop a model of how local structure is interpreted in the image triage task. Specifically, we evaluate its importance when examining a pair of similar images, such as those captured within the same burst of shots. We focus on image pairs because small differences in their local structure are particularly difficult to see using standard thumbnails. We propose a learned model for calculating the importance, or saliency, of image pixels in the context of other images. We call this feature *cosaliency*.

Cosaliency is fundamentally distinct from traditional definitions of image saliency because it is a property of an image set rather than a single image. A salient object in a single image may not be cosalient in an image set if it does not change in an interesting way. Likewise, a cosalient object may not be salient without the context of the second image. Accordingly, naïve approaches–such as simple pixel-level differences–will not be a good approximation of cosaliency because not all local structure changes or motion are equally salient for human observers. For example, dynamic backgrounds and parallax due to camera motion are generally non-salient when compared to foreground objects, but would generate high pixel-level differences. A proper cosaliency model could help to visualize the most cosalient features that summarize the differences across an image set. A pair of summarizing thumbnails, for example, could greatly simplify the image triage task.

RELATED WORK

Our work is closely related to three areas of computer graphics and vision: image summarization, retargeting, visual saliency, and change detection.

Image summarization is the process of creating a summary representation for image data that is smaller than the original, yet still contains all the meaningful image content. Image epitomes [8] are summaries in the sense that they contain enough data to reconstruct their original images, but look very little like their input. Bidirectional similarity [19], however, treats summarization as an optimization problem that constrains the summary to appear similar to the input image. These methods try to summarize the salient content in a single image, whereas in this work we try to summarize the cosalient content in an image set.

Retargeting is a similar process of adapting a large image for display on a device with a different screen size and aspect ratio. In 2003, Suh et al. [22] demonstrated that simple cropping could make browsing image collections easier by displaying smaller portions of images that focused on important image content like faces. In their 2005 work, Liu et al. [11] propose a more complicated retargeting method that performs a non-linear warp emphasizing interesting image features. No current retargeting methods consider the context of other images in the same collection, however.

Implicit in each of these summarization and retargeting techniques is the assertion that certain parts of images are more important, or salient, than others in the eyes of human viewers. The computer vision community has long studied mechanisms for automatically detecting these salient regions. Faces, for example, are well known as high-level salient image features and can be detected with high accuracy using modern algorithms [23], [4]. Studies have also shown that the human visual system is sensitive to low-level image features [7], [15]. In 2009, Judd et al. [9] performed a comprehensive user study which showed a linear combination of both high and low-level image features can yield better saliency maps than those based on low-level features alone.

Most research in image saliency, however, only considers a single image at time and therefore may be less useful for image comparison tasks. A more related field may be that of automated surveillance and video analysis. Boiman and Irani's irregularity detection technique [3] tries to reconstruct a query image or video using patches from a database of "familiar" content. Any regions that cannot be reconstructed are marked as suspicious. Change detection algorithms [20] [17] instead learn statistical properties for regions of an image and then compute the probability of each pixel in the new image being an outlier in the learned model. Motion detection techniques [6] [25] can accurately detect objects in motion, but are restricted to operating only on continuous video sequences. Both algorithms operate on sets of aligned images, which are difficult to acquire using a hand-held camera–dynamic scenes and substantial parallax often foil alignment algorithms. Perceptual studies have shown that motion is a strong cue for attention [16]. These studies are relevant to the human visual system or a continuous video sequence input, but not necessarily for image burst input. Moreover, locations that are likely to draw human short-term attention are not necessarily going to be useful for image triage.

Our approach combines multiple features using a machine learning framework. The research in the areas described above provides a wealth of image and image set features that we can leverage to improve the quality of our learned cosaliency model.

OVERVIEW

This paper is primarily an exploration of the notion that image saliency changes as a function of context. Accordingly, we motivate and validate our approach using studies of human observers considering the image triage task. In our first experiment ("Detecting Changes"), we ask users to identify the regions in the images with the most salient differences. We then use this data to learn a model for computing the cosaliency of one image in the context of another ("Learning What Matters"). Finally, we perform a validation study ("Collection-Aware Cropping") that weighs the utility of detail crops generated with cosaliency against the utility of single image derived crops for the image triage task.

DETECTING CHANGES

Before we can create a model for saliency in the context of other images, we must first verify that saliency does indeed change with context. To answer this question, we perform a short user study in which we ask users to manually generate crops that highlight the most salient differences between pairs of similar photographs. In this section, we explain the details of this study and discuss some of the results.

Methodology

In this study, we ask participants to create pairs of crops that, when used in conjunction with standard thumbnails, would be most useful in image triage. After a brief explanation of the image triage task, subjects view pairs of similar images at a medium resolution (fit within a 400x400 pixel box). They then select the best crop windows from the image pair using a provided web tool (Figure 1). Users may mark any number of regions for an image pair, but are restricted to marking square regions that map to 100x100 detail crops for simplicity–all crops discussed in the remainder of this paper are 100x100 pixels unless otherwise noted. Each user marks a total of 35 image pairs. We collected these photographs from various personal photo collections. Although leveraging online photo repositories could have provided more data points, our focus is on images that have not yet been triaged. This set is better represented by these personal collections. We randomize the presentation order of the image pairs to counterbalance any learning or fatigue effects. Additionally, the web tool is designed such that "lazy" participants do not strongly influence our results–it is easier to add zero crop windows than it is to add many bad ones. We employed Amazon's Mechanical Turk matchmaking service [14] to recruit our study subjects. Each user was only allowed to participate once and was paid $0.50 for his or her efforts.

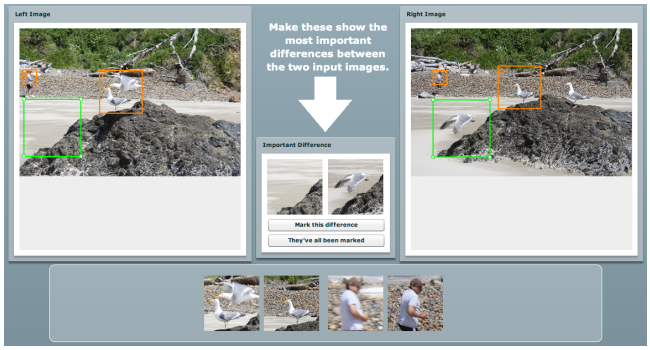


Figure 1: The web tool provided for simultaneously cropping similar images. Users were allowed to mark as many regions as they liked for any particular pair of images before moving on.

Results

In total, 59 users generated 4,396 sets of crops across all image pairs. Due to the sheer quantity involved, we aggregate the crop windows selected by subjects as user-generated cosaliency maps. We are interested in how important each pixel is in the image triage task. As a proxy for this importance, we examine how often any particular pixel is included in a crop marked by a user. If we perform a count of occurrences in crops, we can generate a cosaliency map, but it makes no distinction about which pixels are important within a user's selected window. In their single image saliency work, Judd et al. [9] found users put more importance on the centers of images. In addition, we observed that many subjects would add margins around objects when marking them for crops. Accordingly, we should weight included pixels by their normalized positions within each crop window to generate the more continuous cosaliency map shown in Figure 2(b). The user-generated cosaliency for a pixel (x, y) is defined as

$$cosaliency(x, y) = \sum_i w_i^2 f_i(x, y)$$

where w_i is the width of the ith crop window selected by a user and f_i is a 2D Gaussian weighting function centered within the ith crop window with standard deviation $\sigma = w_i/3$. The factor of 3 is added to ensure the Gaussian falls off nicely within the crop window's boundaries. The resulting cosaliency maps tend to place high weight on moving humans and areas with large changes in image content.

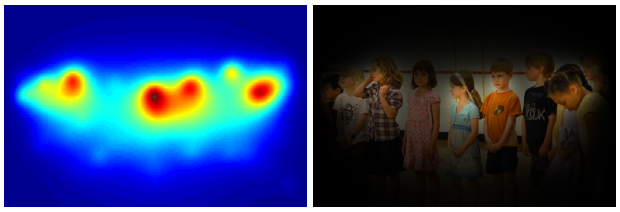
At first glance, it may appear that users are simply selecting the subjects of each scene. Closer inspection, however, reveals that only subjects with significant changes in pose are given high weight. For example, consider the line of children shown in Figure 2(a). The cosaliency map generated by our user study puts much higher weight on the girl in blue over her counterpart in pink. Though both seem equally salient in the context of a single image, the girl in pink becomes much less interesting when comparing the two images. As Figure 3(a) shows, the pink girl's facial expression and pose change negligibly across the image pair, and thus are less relevant to the image triage task. These user-generated maps fit well our idea of cosaliency and can be used as a training goal maps for machine learning. For the remainder of the paper, we will use the term *goal map* to refer to these user-generated cosaliency maps.

LEARNING WHAT MATTERS

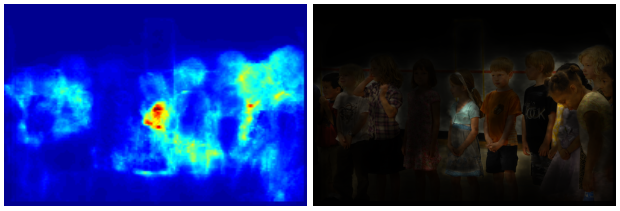
Our goal in this section to construct a function that behaves similarly to the goal maps produced above but can be computed automatically using only features of the images themselves. We choose to model the underlying cosaliency rather than the cropping function because it enables other applications such as retargeting and image abstraction. We learn how to compute the cosaliency function using a machine learning approach inspired by that of Judd et al. [9]: Each user-generated goal map is treated as a binary classifier on saliency for pixels in the image pairs. A pixel is classified as salient if it is in the top 30% of the goal map's histogram. These classifications act as training examples to a linear support vector machine [5] along with a number of calculated image features. The result is a set of linear combination weights for the image features that together form a good approximation of the goal map.

Over the course of this work, we experimented with many different kinds of image features. We were particularly interested in features computed from multiple input images, as these inherently include some notion of context. Figure 4 shows a selection of the features we tried (not all were useful). A description of how each is calculated follows.

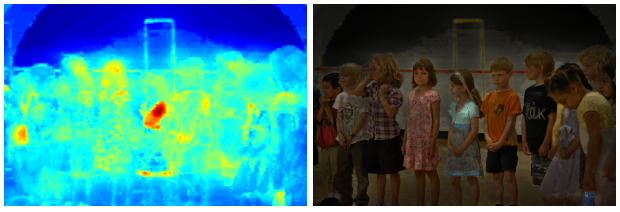
(a) Input image pair



(b) User-generated cosaliency (Goal map)



(c) Our computed cosaliency



(d) Single image saliency [9]

Figure 2: An input image pair and cosaliency maps. 2(b) weights each pixel by its position within the user-generated crop windows. 2(c) shows the result of our learned cosaliency model. Each cosaliency map is also overlaid on an input image to show correspondence between cosalient regions and image content. The brighter a region is shown, the more cosalient its content. The cosaliency maps for only the left image in the pair are shown here. The right image's cosaliency map is very similar.

Single Image Features

- *Gaussian Prior* (Fig. 4(a)) – A simple Gaussian fit to fall off nicely from the middle of the image.
- *Contrast* (Fig. 4(b)) – The gradient magnitude of a grayscale version of the image.
- *Faces* (Fig. 4(d)) – Face detection algorithms such as [23] and [4] mark rectangular regions of images that appear to contain faces. We create a map from this by filling this regions with Gaussians.
- *Oliva Saliency* (Fig. 4(e)) – The single image saliency algorithm of Oliva et al. [15].
- *Judd Saliency* (Fig. 4(c)) – The single image saliency algorithm of Judd et al. [9]. Judd saliency is a composition of many other single image saliency metrics, and thus generally replaces their roles in our learning framework.

(a) A static subject and its cosaliency

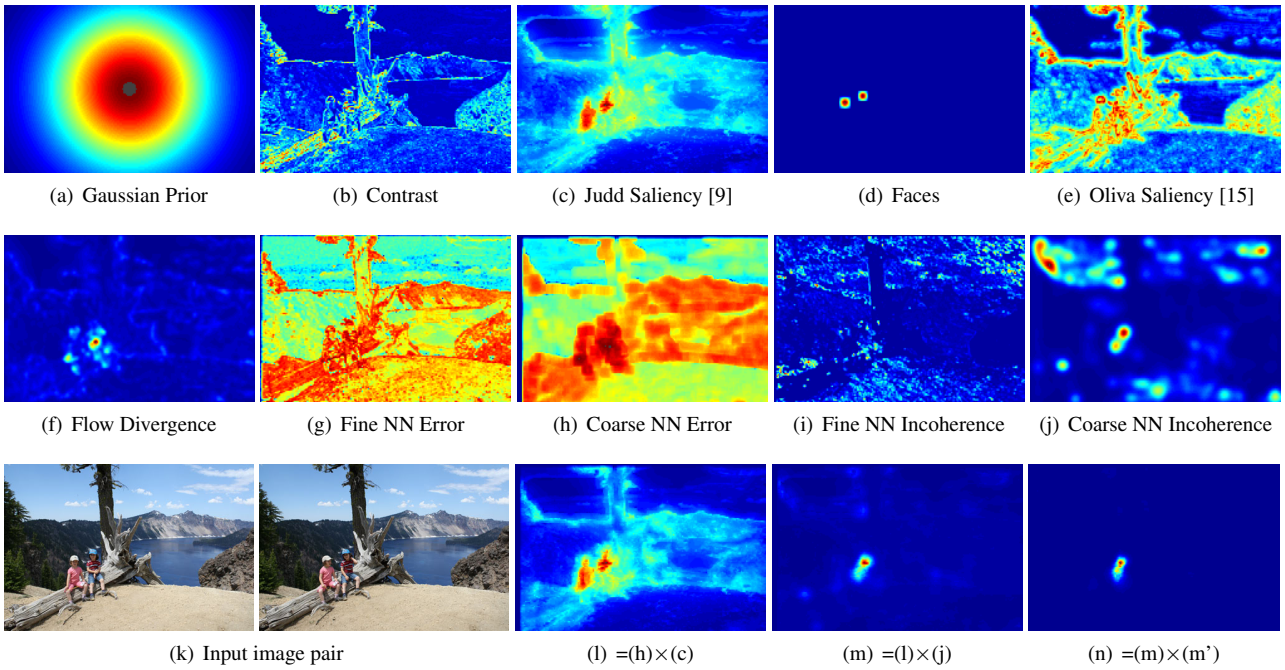
(b) A dynamic subject and its cosaliency

Figure 3: Corresponding crops and their user-generated cosaliencies from the image pair showing the line up of children seen in Figure 2(a). Note that the first girl's facial expression changes very little between the two images and has low cosaliency. The second girl's pose changes dramatically and is strongly cosalient.

Multi-Image Features

- *Flow Divergence* (Fig. 4(f)) – The absolute value of the divergence of an optical flow field. Optical flow algorithms, such as [2], try to compute a dense vector flow field that describes the motion of every pixel in an image with respect to another image in a time series. Discontinuities in the flow field can indicate interesting changes. We find discontinuities by computing the divergence of the flow field.
- *Nearest Neighbor Error* (Fig. 4(g), 4(h)) – The total error between an image patch in the source image and its closest matching neighbor in the target image, as calculated by [1]. If the target image does not contain the same object or if that object has changed appearance significantly, the patch error will be high. This can be run at multiple scales to find changes ranging from local to global.
- *Nearest Neighbor Incoherence* (Fig. 4(i), 4(j)) – The gradient magnitude of the nearest neighbor offset field as calculated by [1]. Generally, similar images will have large coherent regions in their nearest neighbor offset fields where the images match. Portions of the image that contain interesting differences tend to have incoherent flow fields. This can also be computed at multiple scales.

The best features turned out to be multiplicative combinations of single image features and image set features (Figures 4(l) to 4(n)), essentially image changes weighted according to single image saliency. The product of nearest neighbor patch error, gradient magnitude of nearest neighbor offsets, and Judd saliency works particularly well for images with moving animals or people. This feature is computed at many scales to capture changes of all sizes. The model learned from the entire image set is shown in Table 1. Figure 5 shows a selection of goal maps and their corresponding calculated cosaliency maps.



(a) Gaussian Prior (b) Contrast (c) Judd Saliency [9] (d) Faces (e) Oliva Saliency [15]

(f) Flow Divergence (g) Fine NN Error (h) Coarse NN Error (i) Fine NN Incoherence (j) Coarse NN Incoherence

(k) Input image pair (l) =(h)×(c) (m) =(l)×(j) (n) =(m)×(m')

Figure 4: An image pair and its calculated image features. The first row 4(a) - 4(c) shows features calculated using a single image as input. The second row 4(f) - 4(j) shows features calculated on both images together. The features shown in the third row are multiplicative products of the other features. Judd weighted error 4(l)is the product of NN error 4(h) and Judd saliency 4(c). Multiplying in the incoherence feature 4(j) yields 4(m). Features computed on multiple images are often asymmetric, where one image acts as a source and the other as a target. Accordingly, each of the above features has a corresponding feature with the roles of the images reversed. 4(n) is the product of 4(m) with its corresponding feature (after translational alignment).

Weight	Feature
0.049	Faces
0.112	Flow Divergence
0.108	NN Error × NN Incoherence × Judd Saliency
-0.108	Same feature at coarser 2^{-1} scale
0.529	Same feature at coarser 2^{-2} scale
-0.293	Same feature at coarser 2^{-3} scale
1.040	Same feature at coarser 2^{-4} scale

Table 1: A learned model for cosaliency. All features are normalized to have mean $\mu = 0$ and standard deviation $\sigma = 1$ to give the weights a common scale. Negative weights appear because linear SVM has no non-negativitity constraints.

COLLECTION-AWARE CROPPING

Now that we have the ability to generate cosaliency maps, we validate that cosaliency offers an improvement over existing saliency algorithms for use in the image triage task. In this section we discuss how to generate collection-aware crops given a cosaliency map and then evaluate their perceived utility for image triage against existing thumbnailing techniques.

Generating Crops

Good crops are generated from any saliency map using a simple two step procedure. First, we generate a binary thresholded version of the saliency map. Binary saliency maps are preferred for cropping because continuous saliency maps have a tendency to contain small "hot spots" that can bias crops towards small image features. After applying a simple percentile threshhold, we perform morphological closure and opening operations to eliminate holes in connected regions and small disconnected segments. Given a clean binary saliency map, we find optimal crop windows by looking for its scale space extrema. Scale space extrema have previously been used by Lowe to detect scales for SIFT feature points [13]. We use the $(x, y, scale)$ position of the extremum with the greatest magnitude to determine the contents of the crop. Figure 6 illustrates the stages of this process.

One point that merits discussion is that both images in a pair may have different cosaliency maps, depending on the scene and amount of subject movement. Accordingly, both will have different positions for their scale space extrema. If one simply generates crop windows for the images independently, the result is often crop pairs that don't correspond to the same parts of the scene or are otherwise confusing for users. We find a better approach is to locate the optimal crop from just one of the images and then find its corresponding region in the other image using a simple translational alignment of the image pair. We translate one image using the mode offset of the nearest neighbor field [1]. This simple method could be extended to account for also rotation and scale differences, but translation alone was sufficient in all our experiments.

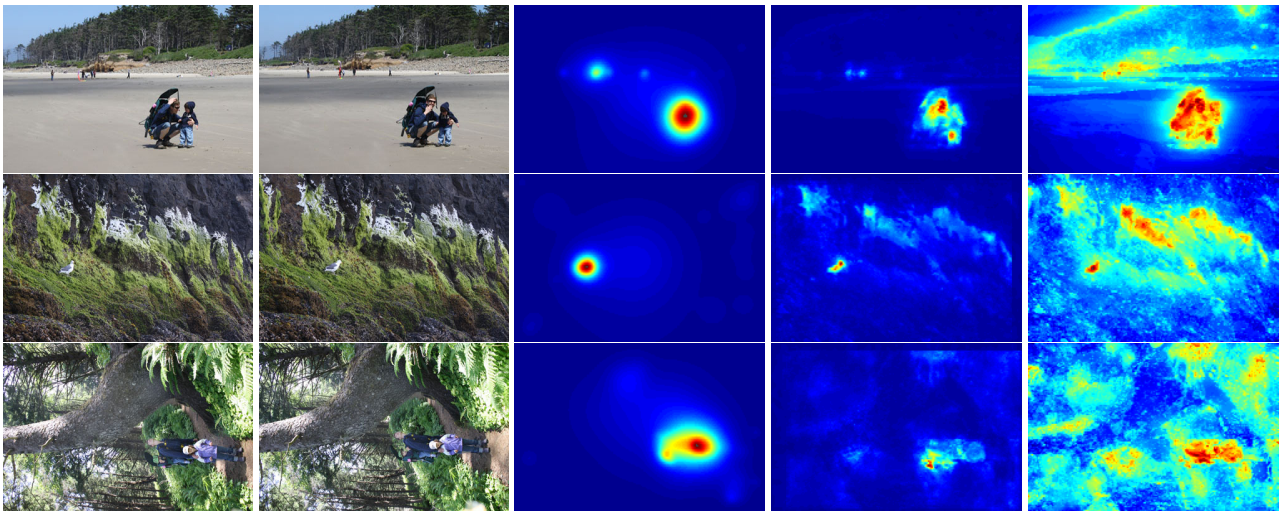


Figure 5: Input images and saliency maps produced from user study data (goal map), the learned cosaliency model, and Judd single image saliency. Note that the Judd Saliency maps often contain high levels of saliency outside the regions marked by users as useful for image triage. The computed cosaliency maps tend to limit their influence to regions reported to have high utility for the triage task.

Evaluation

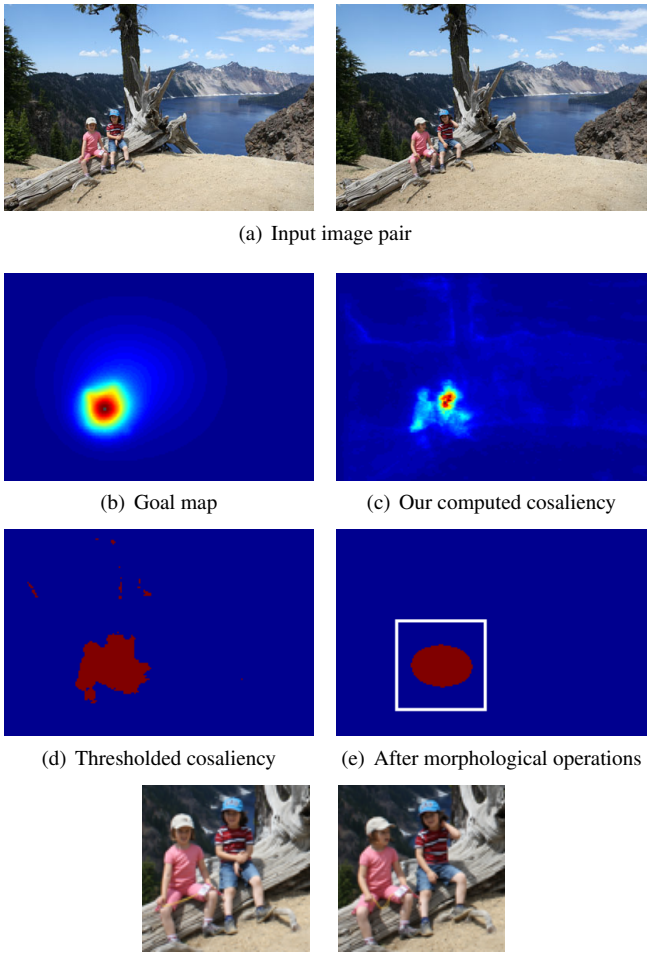
In order to confirm the utility of our collection-aware crops, we perform a validation study that compares our crops against standard techniques.

Methodology In this study, we ask users to strictly rank the utility of the different kinds of detail crops for the image triage task. Each user views a pair of uniformly scaled thumbnails along with four pairs of close up, detail crops. These four pairs represent four different approaches for creating detail crops to be used in the triage task: cosaliency based crops, single image saliency based crops, a crops containing the center 30% of the image, and the most typical crop produced by the previous user study. The intention is that an end user would have sufficient screen space on a mobile device to see four total images simultaneously, the two regular thumbnails and a pair of detail insets. In order to avoid biasing our results, we use 7-fold cross-validation to generate our detail crops: Training data cannot be used to evaluate our learning technique, so instead we train multiple models using different portions of our dataset. In this study we evaluate the effectiveness of each model on the portion of the dataset that was not included during training. Table 1 provides the parameters of a model trained on the entire dataset, not used in our validation study. The cross-validation results provide evidence that this model should extrapolate well to novel image pairs. Again, we randomize the presentation order of image pairs and crop types to counterbalance any learning or fatigue effects. This also prevents "lazy" participants from skewing our results. We also again employ Amazon's Mechanical Turk service to recruit participants. Each subject participated only once, ranking a random set of 10 image pair crops and receiving $0.25 for his or her time.

Results In total we had 198 users participate in the study for a total of 2094 individual rankings. Figure 7 shows the study results aggregated across all images. On average, users ranked the thumbnails generated using the goal map higher than the other thumbnail types, as one would expect. We also observed that users generally found cosaliency based crops better than singe image saliency crops for identifying significant image changes. Across all image pairs, 56.5% of rankings held cosaliency higher than saliency–a statistically significant result using a Wilcoxon signed-rank test with p-value 2.1×10^{-9}.

The aggregate results only tell a small portion of the story, however. Rankings between thumbnail types were not consistent across all image pairs. For 29 of the 35 pairs, our method provides crops that are as effective or better at depicting changes than the other automated methods. Rankings for 13 of these show significant preference for cosaliency over single image saliency (using a signed-rank test at 5% significance). Rankings for the remaining 6 image pairs show a significant preference for single image saliency, but can largely be explained by differences in framing. Figure 8 shows results for a few different image pairs.

The framing issue that came up repeatedly during the validation study is the tendency for users to select a well framed thumbnail that shows little change over a poorly framed thumbnail that shows more drastic scene changes. In a more realistic setting, the thumbnails that we generate would be part of an interactive system to help users compare and navigate images. Our technique would simply provide automated suggestions for interesting regions to examine. Within the context of such a system, perfect framing would be less critical because users could easily shift the regions of interest produced by our algorithm to encompass the areas that mat-



(a) Input image pair

(b) Goal map

(c) Our computed cosaliency

(d) Thresholded cosaliency

(e) After morphological operations

(f) Resulting detail crops

Figure 6: The detail cropping process illustrated. The white box in 6(e) represents the final crop window automatically selected from all possible scales and positions.

ter most to them. We could also add a post-processing step that adjusts crop framing to be more aesthetically pleasing in the manner of Liu et al. [12].

LIMITATIONS AND FUTURE WORK

Although we have shown that cosaliency outperforms saliency for the triage task, there are still issues left to be addressed by future works. Below are a few areas we believe could improve upon this work and provide a better user experience for photographers in the future.

First, this work is limited to the pair of images case in order to make direct testing more feasible. Generally, similar images in personal photo collections appear in groups of arbitrary size, not just pairs. The triage task can still be performed in a pairwise fashion, using pairwise elimination, but a more complete algorithm would take all such similar images into consideration when determining cosaliency. This case is particularly relevant when considering desktop photo management applications where a user may deal with tens of images simultaneously.

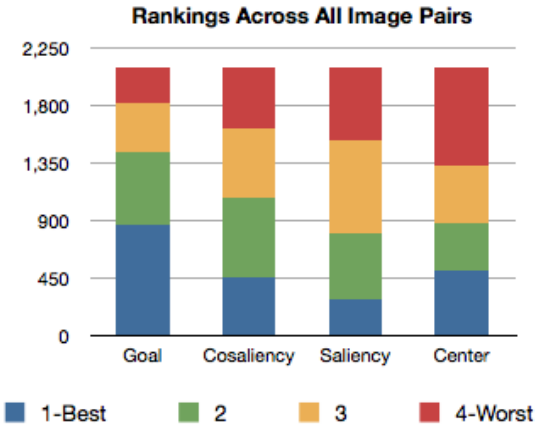


Figure 7: Aggregate rankings across all subjects and image pairs.

Second, our image pairs were drawn from un-triaged photo collections of vacations and family gatherings. Strictly speaking, the model we present may not be appropriate for other types of photography (sports, studio portraiture, etc.) The learning framework, however, is applicable to general photography. In future work, the generality of the cosaliency model could be extended by sampling from a larger, more diverse dataset.

This work only measured the effectiveness of static detail crops. One of the most powerful features of the LCD screen on a mobile device is the ability to animate the information it displays. It would be interesting to investigate the effect of applying various animations when displaying cosalient regions of similar images. Flipping back and forth between images rapidly could take advantage of a user's preattentive visual system while allowing larger image regions to be shown. An animated tour through cosalient regions of the images may also help provide more context so users better understand the quality of the image as a whole.

One final area we would like to see explored more is non-photorealistic visualization of cosalient image features. Although crops provide perfect representations of the data contained within the images, applying other techniques such as image abstraction [24] may prove more useful in the triage task. One could also combine our approach with other techniques for visualizing image quality such as the representative thumbnails work of Samadani et al. [18].

CONCLUSION

The primary contribution of this paper is recognizing the need for collection-aware image analysis. Images are rarely used in isolation, yet are still generally processed one at a time. In this paper, we propose a new notion of context-aware image saliency we call cosaliency. We also propose a model for calculating cosaliency for novel image pairs and show that users believe it outperforms single image saliency. We hope that this work will be a first step in a direction that will foster future research in collection-aware image summarization.

ACKNOWLEDGEMENTS

The authors would like to thank Marc Levoy, David Salesin, and the Stanford graphics group for their useful discussions and feedback throughout the course of this work. We also thank Connelly Barnes, Tilke Judd, and Lubomir Bourdev for providing implementations of their saliency features. David E. Jacobs acknowledges support from a Hewlett Packard Fellowship and the Adobe Systems Creative Technologies Lab. Finally, we would like to thank the anonymous reviewers for their insightful comments and suggestions for improvement.

REFERENCES

1. Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. PatchMatch: A randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 2009.
2. Michael J. Black and P. Anandan. The robust estimation of multiple motions: parametric and piecewise-smooth flow fields. *Computer Vision Image Understanding*, 1996.
3. Oren Boiman and Michal Irani. Detecting irregularities in images and in video. *International Journal of Computer Vision*, 2007.
4. Lubomir Bourdev and Jonathan Brandt. Robust object detection via soft cascade. In *Proc. CVPR*, 2005.
5. Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 2008.
6. Eric Hayman and Jan-Olof Eklundh. Statistical background subtraction for a mobile observer. In *Proc. ICCV*, 2003.
7. Laurent Itti, Christof Koch, and Ernst Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Pattern Recognition*, 1998.
8. Nebojsa Jojic, Brendan J. Frey, and Anitha Kannan. Epitomic analysis of appearance and shape. In *Proc. ICCV*, 2003.
9. Tilke Judd, Krista Ehinger, Frédo Durand, and Antonio Torralba. Learning to predict where humans look. In *Proc. ICCV*, 2009.
10. Lalatendu Satpathy, Bridget Lewis, Saara Kamppari, Benjamin Elgart, Ajay Prasad, Yong Woo Rhee, Brad Myers and Sue Fussel. Digital photo lifecycle. http://www.webvizdesigners.com/projects_pluto.php.
11. Feng Liu and Michael Gleicher. Automatic image retargeting with fisheye-view warping. In *Proc. UIST*, 2005.
12. Ligang Liu, Renjie Chen, Lior Wolf, and Daniel Cohen-Or. Optimizing photo composition. *Computer Graphics Forum (Proc. Eurographics)*, 2010.
13. David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 2004.
14. Amazon Mechanical Turk. https://www.mturk.com/.
15. Aude Oliva and Antonio Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 2001.
16. D A Poggel, H Strasburger, and M MacKeben. Cueing attention by relative motion in the periphery of the visual field. *Perception*, 2007.
17. Richard J. Radke, Srinivas Andra, Omar Al-Kofahi, and Badrinath Roysam. Image change detection algorithms: A systematic survey. *IEEE Transactions on Image Processing*, 2005.
18. Ramin Samadani, Timothy A. Mauer, David M. Berfanger, and James H. Clark. Image thumbnails that represent blur and noise. *IEEE Transactions on Image Processing*, 2010.
19. Denis Simakov, Yaron Caspi, Eli Shechtman, and Michal Irani. Summarizing visual data using bidirectional similarity. In *Proc. CVPR*, 2008.
20. Chris Stauffer and W.E.L. Grimson. Adaptive background mixture models for real-time tracking. In *Proc. CVPR*, 1999.
21. Steven Drucker, Curtis Wong, Asta Roseway, Steve Glenner, and Steve De Mar. Photo-triage: Rapidly annotating your digital photographs. Technical report, MSR, 2003.
22. Bongwon Suh, Haibin Ling, Benjamin B. Bederson, and David W. Jacobs. Automatic thumbnail cropping and its effectiveness. In *Proc. UIST*, 2003.
23. Paul Viola and Michael Jones. Robust real-time object detection. *International Journal of Computer Vision*, 2001.
24. Holger Winnemöeller, Sven C. Olsen, and Bruce Gooch. Real-time video abstraction. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 2006.
25. Hulya Yalcin, Martial Hebert, Robert Collins, and Michael J. Black. A flow-based approach to vehicle detection and background mosaicking in airborne video. In *Proc. CVPR*, 2005.

Figure 8: A selection of results for individual image pairs in our validation study. For the top two image pairs (11 and 34), users rankings showed a significant preference for our cosaliency crops over single image saliency crops. The next two groups (17 and 31) yielded no statistically significant preference for our cosaliency or single image saliency crops. User rankings for the final image pair (23) above showed a significant preference for single image saliency crops. Although the cosaliency crop has content that is closer to the goal crop, most users chose the saliency crop that has aesthetically better framing.