Saliency Detection on Light Field

Nianyi Li[1] Jinwei Ye[1] Yu Ji[1] Haibin Ling[2] Jingyi Yu[1]

[1]University of Delaware, Newark, DE, USA. {nianyi,jye,yuji,yu}@eecis.udel.edu
[2]Temple University, Philadelphia, PA, USA. hbling@temple.edu

Abstract

Existing saliency detection approaches use images as inputs and are sensitive to foreground/background similarities, complex background textures, and occlusions. We explore the problem of using light fields as input for saliency detection. Our technique is enabled by the availability of commercial plenoptic cameras that capture the light field of a scene in a single shot. We show that the unique refocusing capability of light fields provides useful focusness, depths, and objectness cues. We further develop a new saliency detection algorithm tailored for light fields. To validate our approach, we acquire a light field database of a range of indoor and outdoor scenes and generate the ground truth saliency map. Experiments show that our saliency detection scheme can robustly handle challenging scenarios such as similar foreground and background, cluttered background, complex occlusions, etc.*, and achieve high accuracy and robustness.*

1. Introduction

Salient region detection is a long standing problem in computer vision. It aims to locate pixels or regions in an image that most attract human's visual attention. Accurate and reliable saliency detection can benefit numerous tasks ranging from tracking and recognition in vision to image manipulation in graphics. For example, successful saliency object detection algorithms facilitate automated image segmentation [27], more reliable object detection [11], effective image thumbnailling [30] and retargeting [28].

State-of-the-art solutions have focused on integrating low-level features (pixels or superpixels) and high-level descriptors (regions or objects). However, existing solutions have many underlying assumptions, *e.g.*, the foreground should have a different color from the background, the background should be relatively simple and smooth, the foreground is occlusion free, *etc*. In reality, many real images violate one or multiple assumptions as shown in Fig. 1.

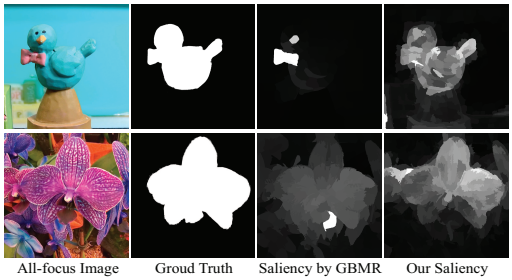


Figure 1. Light field vs. traditional saliency detection. Similar foreground and background or complex background imposes challenges on state-of-the-art algorithms (*e.g.*, GBMR [35]). Using light field as inputs, our saliency detection scheme is able to robustly handle these cases.

By far, nearly all existing saliency detection algorithms utilize images acquired by a regular camera. In this paper, we explore the salient object detection problem by using a completely different input: the light field of a scene. A light field [6] can be essentially viewed as an array of images captured by a grid of cameras towards the scene. Commercial light field cameras can now capture reasonable quality light fields in a single shot. Lytro, for example, mounts a lenslet array in front of the sensor (as shown in Fig. 2) to acquire a light field at a 360×360 (upsampled to 1080×1080) spatial resolution and 10×10 angular resolution. The Raytrix R11 camera can produce a higher spatial resolution at the cost of lower angular resolution. The multi-view nature of the light field has enabled new generations of stereo matching [17] and object segmentation algorithms [32]. In this paper, we explore how to conduct salient object detection using a light field camera.

Conceptually, the light field data can benefit saliency detection in a number of ways. First, the light field has a unique capability of post-capture refocusing [25], *i.e.*, it can synthesize a stack of images focusing at different depths. The availability of a focal stack is inline with the recently proposed "focusness" metric [16]. It is the reciprocal of blurriness and can be estimated in terms of edge scales via scale-space analysis. Second, a light field provides an ap-

proximation to scene depth and occlusions. In saliency detection, even a moderately accurate depth map can greatly help distinguish the foreground from the background. This is also inline with the "objectness" [16], *i.e*., a salient region should complete objects instead of cutting them into pieces.

In addition to focusness and objectness, we also exploit the recent background prior [33]. Instead of directly detecting salient regions, these algorithms aim to first find the background and then use it to prune non-saliency objects. Robust background detection, however, is challenging, especially when the foreground and background have similar appearance or the background is cluttered. To resolve this problem, we utilize the *focusness and objectness* to more reliably choose the background and select the foreground saliency candidates. Specifically, we compute a *foreground likelihood score (FLS)* and a *background likelihood score (BLS)* by measuring the focusness of pixels/regions. We select the layer with the highest BLS as the background and use it to estimate the background regions. In addition, we choose regions with a high FLS as candidate salient objects. Finally, we conduct contrast-based saliency detection on the all-focus image and combine its estimation with the detected foreground saliency candidates.

For validation, we acquire a light field database of a range of indoor and outdoor scenes and generate the ground truth saliency map. Experiments show that our saliency detection scheme can robustly handle challenging scenarios such as similar foreground and background, cluttered background, and images with multiple depth layers and with heavy occlusions, *etc*., and achieve high accuracy and robustness.

2. Related Work

The saliency detection literature is huge and existing solutions can be classified in terms of top-down vs. bottom-up, center vs. background prior, with vs. without depth cue, *etc*. Readers can refer to [4] for a comprehensive comparisons on state-of-the-art solutions. We discuss the most relevant ones.

Top-down vs. Bottom-up. Top-down approaches [22, 34] use visual knowledge commonly acquired through learning to detect saliency. Approaches in this category are highly effective on task-specified saliency detection, *e.g*., identifying human activities [24]. However, a large number of annotated images need to be used for training. Bottom-up methods do not require training and rely on low-level features such as color contrast [14], pixel/patch locations [29], histogram [10], *etc*., for saliency detection. Our approach falls into the category of bottom-up approaches where we add an additional class of focus-related cues.

Center vs. Background Priors. Many saliency detection schemes exploit contrast cues, *i.e*., salient objects are

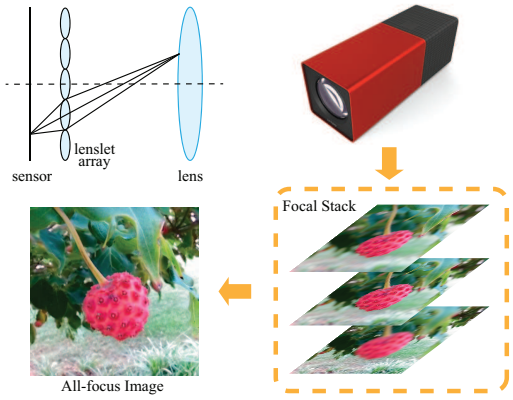


Figure 2. A Lytro light field camera can capture a light field towards the scene in a single shot. The results can be then used to synthesize a focal stack and further a all-focus image.

expected to exhibit high contrast within certain context. Koch and Itti [14] are the first to use center-surround contrast of low level features to detect saliency. Motivated by their work, many existing approaches compute the center-surround contrast either locally or globally. Local methods compute the contrast within a small neighborhood of pixels by using color difference [5], edge orientations [21], or curvatures [31]. Global methods consider statistics of the entire image and rely on features such as power spectrum [12], color histogram [7], and element distributions [29].

Although the center-surround approaches are proven highly effective, Wei *et al*. [33] suggested that background priors are equally important. In fact, one can eliminate the background to significantly improve foreground detection. Yang *et al*. [35] observed that connectivity is an important characteristics of background and used a graph-based ranking scheme to measure patch similarities. Since most existing approaches rely on color contrast, when the foreground and background have similar color, these approaches can easily fail. Our approach resolves this issue by combining color contrast, background prior, and focusness prior w.r.t. different depth layers obtain from the light field.

Depth Cue. Recent studies on human perception [18] have shown that depth cue plays a important role in determining salient regions. However, only a handful of works incorporate depth maps into saliency models. Maki *et al*. [23, 24] used depth cue to detect human motions. Their depth features are highly task-dependent and the detection is performed in a top-down fashion. Niu *et al*. [26] computed saliency based on the global disparity contrast in a pair of stereo images. Lang *et al*. [18] used a Kinect sensor to capture the scene depth. Ciptadi *et al*. [8] used 3D layouts and shape features from depth maps. In this paper, we exploit rich depth information embedded in the light field. Specifically, we use coarse depth information embedded in a focal stack to guide saliency detection. In addition, com-

pared with Kinect or stereo cameras, Lytro or Raytrix light field cameras have a much smaller form factor, *i.e.*, nearly the same size as a webcam.

3. Computing Light Field Saliency Cues

Fig. 3 shows our saliency detection approach using the light field. We first generate a focal stack and an all-focus image through light field rendering. For each image in the focal stack, we detect the in-focus regions and use them as the focusness measure. Next, we combine the focusness measure with the location prior to extract the background and the foreground salient candidates. We further couple the background prior with contrast-based saliency detection for detecting saliency candidates in the all-focus image. Finally, we use the objectness as weights for combining the saliency candidates from the all-focus image and from the focal stack as the final saliency map.

3.1. Focal Stack and All-Focus Images

A unique capability of light field is after-capture refocusing. Here we briefly reiterate its mechanism. A light field stores regularly sampled views looking towards the scene on a 2D sampling plane. These views form a 4D ray database and new views can be synthesized by querying existing rays. Given the light field of a scene, one can synthesize a Depth-of-Field (DoF) effects by selecting appropriate rays from the views and blending them, as shown in Fig. 2. Isaksen *et al*. [13] proposed to render DoF by reparameterizing the rays onto the focal plane and blending them via a wide aperture filter. Ng *et al*. [25] proposed a similar technique in the Fourier space and the solution has been adopted in the Lytro light field camera. Using the focal stack, we can fuse an all-focus image, *e.g*., through photomontage [2]. We refer the readers to the comprehensive survey on light field imaging [19, 36] for more details about the refocusing algorithm.

In this paper, we use the Lytro camera as the main imaging device to acquire the light field. The Lytro camera uses an array of 360×360 microlenses mounted on an 11 megapixel sensor, where each microlens resembles a pinhole camera. It can produce the refocused results at a resolution of 360×360.

We compose an all-focus image by focus fusion using existing online-tools [1] from the focal stack so that the all-focus image has the same resolution as the focal stack. In addition, it is worth noting that DoF effect is not significant in Lytro focal stack due to small microlens baseline. As a result, each slice is just slightly defocused. Therefore, brute-force approaches such as applying saliency detection on each slice and then combine the results are not directly applicable since all slices will produce similar results.

[1] http://code.behnam.es/python-lfp-reader/

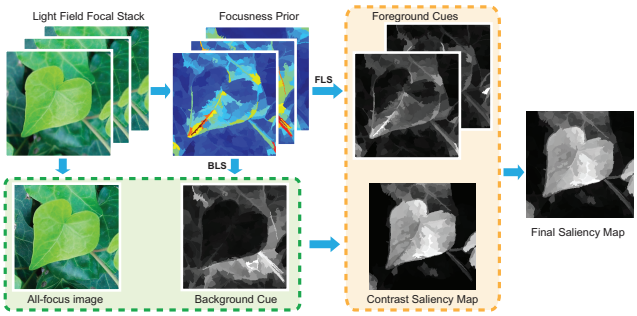


Figure 3. Processing pipeline of our saliency detection algorithm for light fields.

Before proceeding, we explain our notation. We denote $\{I^i\}$, $i = 1, ..., N$ as the focal stack synthesized from the light field and I^* the all-focus image by fusing the focused regions of $\{I^i\}$. Our goal is to compute a saliency map w.r.t. I^*. We segment each slice $\{I^i\}$ and I^* into a set of small non-overlapping regions (superpixels) using the mean-shift algorithm [9]. This segmentation helps to preserve edge consistency and maintain proper granularity. We use (x, y) index a pixel and r to index to a region.

3.2. Focusness Measure

We start with detecting the in-focus regions in each focal stack image I^i and use them as the focusness prior. In the recent focusness-based saliency detection work, Jiang *et al*. [16] measured focusness via edge sharpness. However, edge-based in-focus detection is only reliable when the out-of-focus regions appear severely blurred. In our case, the DoF of Lytro is not as shallow as the one in DSLR. Therefore, edges in out-of-focus regions are not as blurred as in the classical datasets, as shown in Fig. 2. It is hence difficult to use spatial algorithms to separate the in-focus/out-of-focus regions. Our approach is to analyze the image statistics in the frequency domain.

Given an $n \times n$ image I, we first transform I into frequency domain by the Discrete Cosine Transform (DCT)

$$\mathcal{D}(u, v) = \sum_{x=0}^{n-1} \sum_{y=0}^{n-1} \cos(\frac{\pi u}{2n}(2x + 1)) \cos(\frac{\pi v}{2n}(2y + 1)) I(x, y) \tag{1}$$

Next, we compute the image's response with respect to different frequency components. We first apply a series of M bandpass filters $\{P_m\}$, $m = 1, ..., M$ on $\mathcal{D}(u, v)$ for decomposing the signal and then transform the decomposed results back via the inverse DCT. Recall that out-of-focus blurs will remove certain high frequency components. Therefore, only regions with a sharp focus will have high responses at all frequencies. In our implementation, we use a sliding window of 8×8 pixels and compute the variance τ_m within each patch with respect to filter P_m. To ensure reliable focusness measurements, we use the harmonic vari-

ance [20] to measure the overall variance over all M filters:

$$\mathcal{F}(x,y) = \left[\frac{1}{M-1}\sum_{m=1}^{M}\frac{1}{\tau_m^2(x,y)}\right]^{-1} \quad (2)$$

We use $\mathcal{F}(x,y)$ as the focusness measure at pixel (x,y). Under this formulation, only when the response of all filters are high, the harmonic variance $\mathcal{F}(x,y)$ will be high. Any small τ_m will result in low $\mathcal{F}$. Therefore, this formulation ensures that only local windows preserving all frequency components would be deemed as in-focus. Since both DCT and harmonic variance computations are effective, we compute $\mathcal{F}$ for every pixel in the image. Finally, to measure the focusness of a region, we simply compute the average of all pixels within a region r

$$\mathcal{F}(r) = \sum_{(x,y)\in r}\frac{\mathcal{F}(x,y)}{A_r} \quad (3)$$

where A_r is the total number of pixels in r. We will use this region-based focusness prior $\mathcal{F}(r)$ for selecting background and saliency candidates in Section 3.3 and 3.4. It is worth noting that more sophisticated focusness estimation techniques such as scanning through the focal volume can be used. In practice, our measure is sufficient for the task of saliency detection and is much faster.

3.3. Background Selection

Next, we set out to find the background slice. Notice that the background slice is *not* equivalent to the farthest slice in the focal stack. Recall that we synthesize the focal stack without any knowledge on scene depth range. Therefore, the farthest slice may not contain anything in focus and hence provides little cues. Second, the slice that have the farthest object in focus does not necessarily translate to the background slice, *e.g.*, the object may be isolated from majority of the background and should be treated as an outlier.

Our approach is to analyze both the distribution of the in-focus objects with respect to their locations in the image: if the majority of in-focus objects (pixels) lies near the border of the image, then they are more likely to belong to the background. Further, if the corresponding depth layer is far away, its in-focus objects are also more likely to be background. We therefore scan through all focal slices. For each slice I^i, we integrate (project) the focusness measure $\mathcal{F}$ of all pixels along the x and y axes respectively to form two 1D focusness distributions as

$$D_x = \frac{1}{\alpha}\sum_{y=1}^{h}\mathcal{F}(x,y), \quad D_y = \frac{1}{\alpha}\sum_{x=1}^{w}\mathcal{F}(x,y). \quad (4)$$

where w and h are the width and height of the image and $\alpha = \sum_x \sum_y \mathcal{F}(x,y)$ is the normalization factor.

A common assumption in saliency detection is that an salient object is more likely to lie at the central area surrounded by the background [33]. If a focal slice corresponds to the background, its D_x and D_y should be high near the endpoints but low in the middle. To quantitatively measure it, we define a "U-shaped" 1D band suppression filter

$$\mathcal{U}(x,w) = \left(\frac{1}{\sqrt{1+(x/\eta)^2}} + \frac{1}{\sqrt{1+((w-x)/\eta)^2}}\right) \quad (5)$$

where η controls the suppression bandwidth in $\mathcal{U}$ depending on the image size/resolution, *i.e.*, a high resolution image should have a high η. The Lytro focal stack images have a uniform resolution of 360×360 and we use $\eta = 28$ in all experiments.

Finally, we scale the focusness distribution by the suppression filter to compute a Background Likelihood Score (BLS) for each focal slice I^i

$$BLS(I^i) = \rho \cdot \left[\sum_{x=1}^{w} D_x^i(x)\cdot\mathcal{U}(x,w) + \sum_{y=1}^{h} D_y^i(y)\cdot\mathcal{U}(y,h)\right] \quad (6)$$

where $\rho = \exp(\frac{\lambda\cdot i}{N})$ is the weighting factor of layer i in terms of depth, N is the total number of slices in the focus stack and $\lambda = 0.2$. We choose the slice with the highest BLS as the background slice I^B. It is important to note that each focal slice has a corresponding BLS even though it is not chosen as I^B.

3.4. Objectness and Foreground Measures

Alexe *et al.* [3] suggested that a salient object should be complete instead of being broken into pieces and refer to this property as the objectness. Given a focal stack image I^i, we measure the objectness of its focused region using a 1D gaussian filter with mean μ and variance σ as

$$\mathcal{G}(x) = \exp\left(-\frac{x-\mu}{2\sigma^2}\right) \quad (7)$$

μ corresponds to the centroid of the object and σ as its size. Recall that we have already computed the focusness distributions D_x or D_y. Therefore, we can directly obtain $\mu = x_p$ or y_p, that corresponds to the peak location of D_x or D_y respectively. If multiple peaks exist, we simply take their average.

Next we estimate σ as the size of the object. If σ is too small, isolated small superpixels would be treated as an object. If σ is too large, *i.e.*, it would treat the entire image as an object. In our implementation, we choose $\sigma = 45$, *i.e.* 50% Gaussian covers half of the D_x or D_y. We compute the objectness score (OS) for each focal slice

$$OS(I^i) = \sum_{x=1}^{w} D_x^i(x)\cdot\mathcal{G}(x,w) + \sum_{y=1}^{h} D_y^i(y)\cdot\mathcal{G}(y,h) \quad (8)$$

Conceptually, if an object in a given slice is salient, it should have a low BLS and high OS, indicating it belongs to

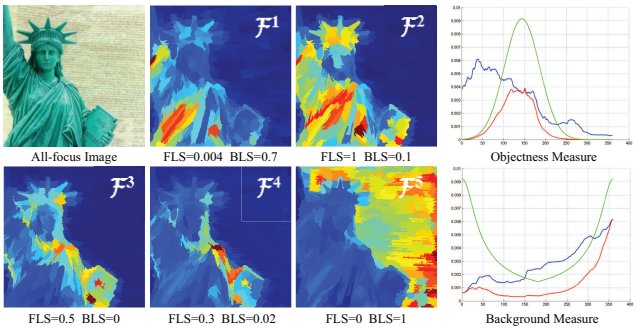


Figure 4. Separating the foreground and background using focusness cues. Left: the computed foreground likelihood score (FLS) and the background likelihood score (BLS) computed on different focal slices. Right: Examples on computing objectness measure (up) and background measure (bottom). Green curve is corresponding filter (U-shape or Gaussian); blue curve is sample D_x/D_y; red curve is the scaled distribution by the filter.

the foreground. We therefore define a foreground likelihood score (FLS) as

$$FLS(I^i) = OS(I^i) \cdot (1 - BLS(I^i)) \tag{9}$$

Same as how we select the background slice I^B, we choose the foreground slices $\{I^F\}$ as one with the higher FLS ($FLS > 0.7 \times max(FLS)$). Fig. 4 illustrates the process of finding the background and foreground slices on a sample image.

4. Saliency Detection

Finally, we combine the cues obtained from the light field focal stack to detect saliency in the all-focus image I^*.

Location Cues. We first locate the background regions in I^* using the focusness measure $\mathcal{F}^B(r)$ of the estimated background slice I^B. To incorporate the location prior [29], we scale the focusness measure for each region R_r in terms of its distance to the center of the image and use it as a new background cue

$$BC(r) = \frac{1}{\gamma}[\mathcal{F}^B(r) \cdot ||\mathbf{p}_r - \mathbf{c}||^2] \tag{10}$$

where γ is a normalization factor, $\mathbf{p}_r$ is the centroid of r and $\mathbf{c}$ is image center. We further threshold the BC for determining the background regions $\{B_{r'}\}$, $r' = 1, ..., K$ in I^* (where K is the total number of background regions). We can then compute the Location cue as:

$$LC(r) = \exp(-\beta \cdot BC(r)) \tag{11}$$

In our experiment, we use $\beta = 10$.

Contrast Cues. Once we obtain the background regions, we apply the color-contrast based saliency detection on the non-background region. For each non-background region r and background region r' in I^*, we calculate their color difference $\delta(r, r')$ w.r.t. r' as $\delta(r, r') = \max\{|red(r) - red(r')|^2, |green(r) - green(r')|^2, |blue(r) - blue(r')|^2\}$. To improve robustness, we use compute the harmonic variance of all $\delta(r, r')$ for r

$$HV(r) = \left[\frac{1}{K}\sum_{r'=1}^{K}\frac{1}{\delta(r, r')}\right]^{-1} \tag{12}$$

Combining the harmonic variance of color difference HV with location cue LC, we obtain a color contrast based saliency map as

$$S_C(r) = HV(r) \cdot LC(r) \tag{13}$$

Foreground Cues. From the detected foreground salient candidates $\{I_j^F\}$, $j = 1, ..., L$ via focusness analysis (where L is the total number of foreground slices), we compute the foreground cues the combining the focusness maps $\mathcal{F}_j^F(r)$ and the location cue LC:

$$S_F^j(r) = \mathcal{F}_j^F(r) \cdot LC(r) \tag{14}$$

Combine. Finally, We use the objectness measure as weight for combining the contrast based salience map $S_C(r)$ and foreground maps $S_F^j(r)$ as:

$$S(r) = \sum_{j=1}^{L}\omega_j \cdot S_F^j(r) + \omega_C \cdot S_C(r) \tag{15}$$

where $\{\omega_j\}$ and ω_C are the objectness weights calculated by Eqn. 8.

5. Experiments

Recall that most previous approaches use a single image as input where as our approach uses the light fields. Since a light field captures much richer information of the scene than a single image, our comparisons do not intend to show that our technique outperforms the state-of-the-art as any such comparisons would be unfair. Rather, our goal is to show that the additional information provided by the light field can greatly improve saliency detection tasks.

Further, traditional benchmark data sets [21, 1] are all single images and cannot be used to test our solution. We therefore first collect a dataset of 100 light fields using the Lytro light field camera. The dataset consists of 60 indoor scenes and 40 outdoor scenes. For each data, we ask three individuals to manually segment the saliency regions from the all-focus image. The results are deemed ground truth

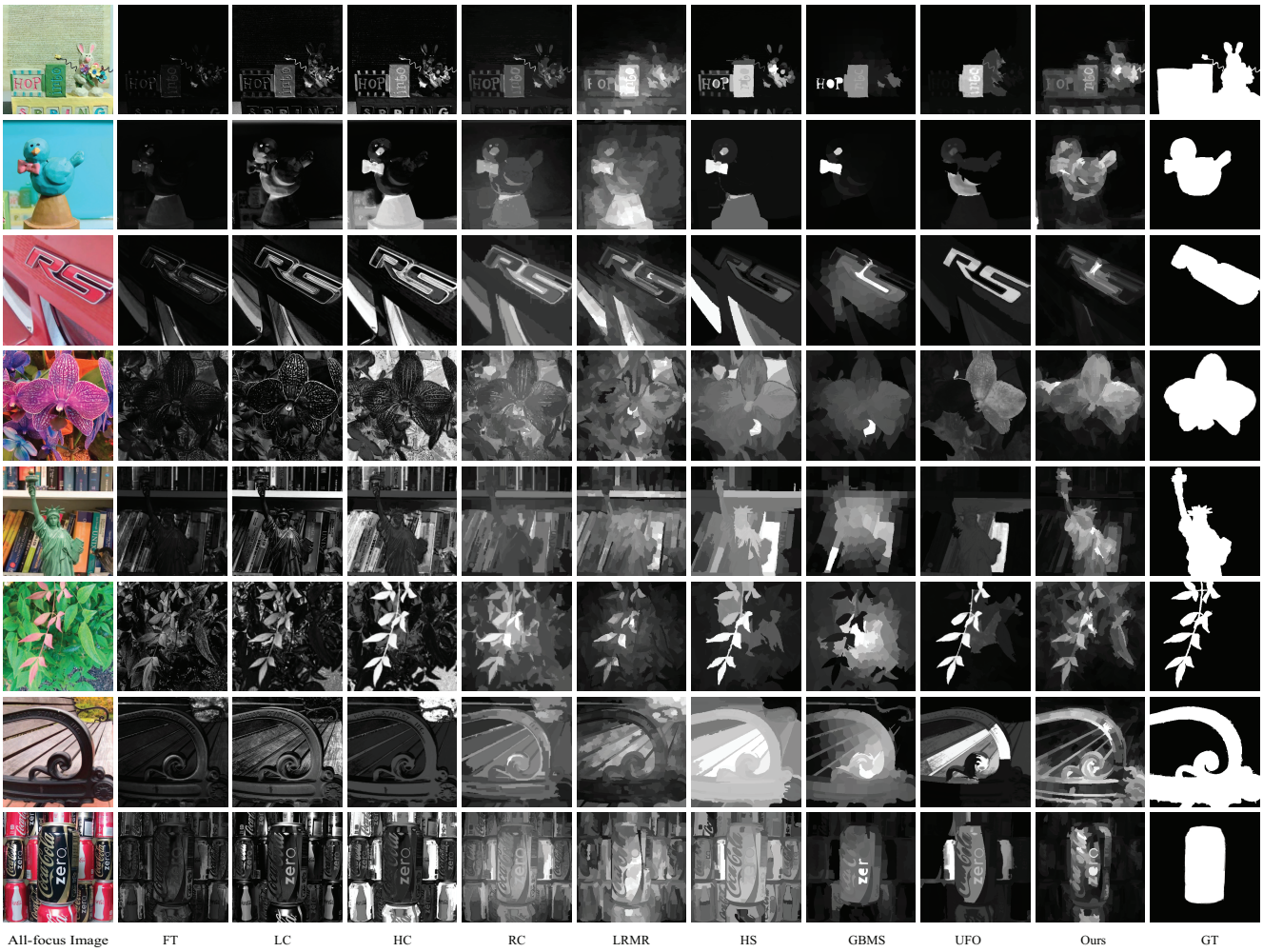


Figure 5. Visual Comparisons of different saliency detection algorithms vs. ours on our light field dataset.

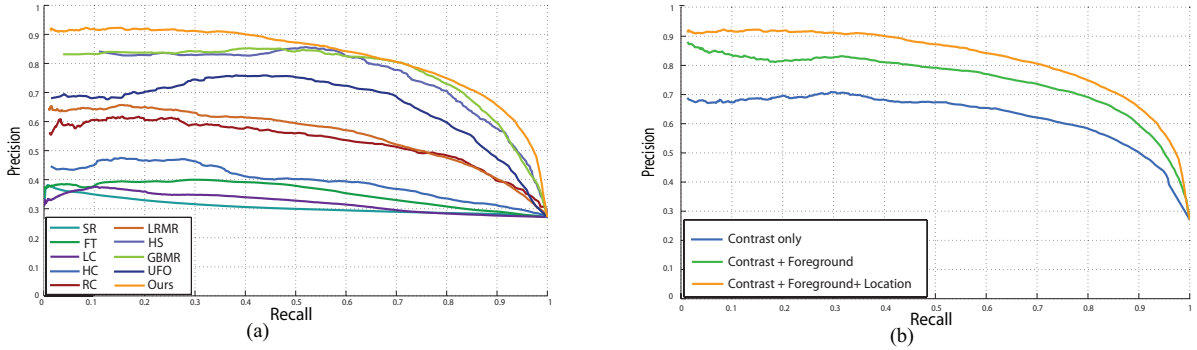


Figure 6. (a) PRC comparisons on our light field dataset; (b) PRC comparisons using different cues in our approach.

only when all three results are consistent (*i.e.*, they have an overlap of over 90%).

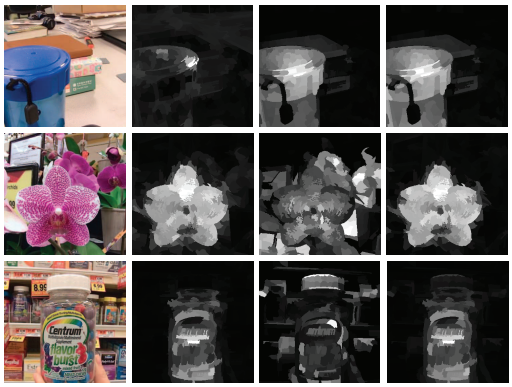
We show our light field saliency detection results and the results using a range of unsupervised schemes on the all-focus image. These include algorithms based on spectral residual (SR [12]), spatiotemporal-cues (LC[37]), graph-based saliency (GB [15]), frequency-tuning (FT [1]), global-contrast (HC and RC [7]), Low Rank Matrix Recovery (LRMR [29]), Graph-Based Manifold Ranking (GBMR [35]), focusness-based (UFO [16]) and Hierarchical Saliency (HS [34]). All these methods have open source code and we use the default parameter.

To quantitatively compare different methods, we use the canonical precision-recall curve (PRC) to evaluate the similarity between the detected saliency maps and the ground truth. Precision corresponds to the percentage of salient pixels that are correctly assigned and recall refers to the fraction of detected salient region w.r.t. the ground truth saliency. Fig 6(a) shows the PRC comparison result on our light field dataset. Our experiment follows the settings in [7], *i.e.*, we binarize the saliency map at each possible threshold within $[0, 255]$.

Notice that the PRC are less smooth than they appear in traditional saliency works. This is due to the small amount of data in our dataset (100 light field sets vs. 1000 images in classical benchmarks), although the curves still provide useful insights on the performance. Also note that a large number of scenes in our light field dataset is highly challenging to previous techniques, *i.e.*, many have complex background or similar foreground and background. Fig. 5 shows sample in-focus images of these difficult scenes. We observe that SR, HS, LC and HC produce a very low precision (0.3 0.4) whereas the recently proposed GBMR [35] and HS [34] can still achieve reasonable performance. This is partially due to the background connectivity prior used in GBMR and the multi-scale features used in HS. Results using our technique produces the highest precision in the entire recall range. This illustrates the importance of focusness and objectness prior provided by the light field. In Fig. 5, we show the saliency detection results for visual comparisons. For very challenging scenes such as the blue bird (second row) , our approach produces much better results than single-based techniques.

We further compare the saliency components obtained using different cues, *i.e.*, color contrast, location and focusness cues. Fig. 6(b) shows the PRC comparisons using individual vs. combined cues. The plot illustrates that each cue has its unique contribution to saliency detection, although in some cases, an image can be dominated by a specific cue as shown in Fig. 7. In the first row, color contrast provides most valuable cues and the estimated saliency from it resembles the final one. This is mainly because the blue mug lacks texture and hence is not robustly detected as the foreground object to provide focusness cues. In contrast, in the flower scene in the second row, the color contrast result treats both the foreground flower and the background clutter as saliency. The focusness cue, however, manages to correct the errors by removing the background. In the last example, the color contrast result misses the foreground bottle and the focusness cue manages to add it back.

Limitations. The performance of our algorithm is largely dependent on the quality of the acquired light field. Lytro, however, has a much narrow Field-of-View than regular cameras. Therefore, objects in our light fields generally appear "bigger" than in other benchmarks. With emerging interest on light field camera designs, we expect next-generation models to overcome this limitation.



(a) (b) (c) (d)

Figure 7. Saliency detection using different cues. (a) All-focus images; (b) Detected saliency using focusness cues; (c) Detected saliency using color contrast. (d) Saliency results by combining (b) and (c).

There are also alternative approaches to use the light field for saliency detection. For example, one can potentially first construct a depth map using stereo matching. However, the quality of stereo matching depends largely on scene composition. Nevertheless, even a low quality depth map may provide useful cues comparable to the focusness cue. Furthermore, it is also possible to first conduct saliency detection on the all-focus image and then use the results to improve the quality and speed of light field stereo matching.

6. Conclusions

We have presented a saliency detection algorithm tailored for light fields. We believe this is the first light field saliency detection scheme. The key advantage of using a light field instead of a single image is that it provides both focusness and depth cues. In recent works [26, 16], these new cues have shown great success in improving accuracy and robustness in saliency detection. Our solution echoes these observations and also provides an alternative and more robust method to extract these cues through the analysis of light fields. Experiments show that our technique can handle many challenging scenarios that cast problems on traditional single-image-based algorithms. Another contribution of our work is the construction of the light field saliency dataset which consists of the raw light field data, the synthesized focal stacks and all-focus images, and the ground truth saliency maps. Our immediate future work is to build a much larger and comprehensive database and share it with the community.

Acknowledgements

This research is supported by National Science Foundation Grant IIS-1218156.

References

[1] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk. Frequency-tuned salient region detection. In *CVPR*, pages 1597–1604, 2009. 5, 6

[2] A. Agarwala, M. Dontcheva, M. Agrawala, S. Drucker, A. Colburn, B. Curless, D. Salesin, and M. Cohen. Interactive digital photomontage. In *ACM SIGGRAPH 2004*, 2004. 3

[3] B. Alexe, T. Deselaers, and V. Ferrari. Measuring the objectness of image windows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2189–2202, 2012. 4

[4] A. Borji, D. N. Sihite, and L. Itti. Salient object detection: a benchmark. In *ECCV*, 2012. 2

[5] N. Bruce and J. Tsotsos. Saliency based on information maximization. In *NIPS*, pages 155–162, 2006. 2

[6] C. Buehler, M. Bosse, L. McMillan, S. Gortler, and M. Cohen. Unstructured lumigraph rendering. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '01, pages 425–432, 2001. 1

[7] M. Cheng, G. Zhang, N. Mitra, X. Huang, and S. Hu. Global contrast based salient region detection. In *CVPR*, pages 409–416, 2011. 2, 6, 7

[8] A. Ciptadi, T. Hermans, and J. M. Rehg. An In Depth View of Saliency. In *British Machine Vision Conference (BMVC)*, 2013. 2

[9] D. Comaniciu and P. Meer. Mean shift: a robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002. 3

[10] L. Duan, C. Wu, J. Miao, L. Qing, and Y. Fu. Visual saliency detection by spatially weighted dissimilarity. In *CVPR*, pages 473–480, 2011. 2

[11] D. Gao, S. Han, and N. Vasconcelos. Discriminant saliency, the detection of suspicious coincidences, and applications to visual recognition. *TPAMI*, 31(6):989–1005, 2009. 1

[12] X. Hou and L. Zhang. Saliency detection: A spectral residual approach. In *CVPR*, pages 1–8, 2007. 2, 6

[13] A. Isaksen, L. McMillan, and S. J. Gortler. Dynamically reparameterized light fields. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '00, 2000. 3

[14] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *TPAMI*, 20(11):1254–1259, 1998. 2

[15] H. J., K. C., and P. P. Graph-based visual saliency. In *NIPS*, pages 545–552, 2006. 6

[16] P. Jiang, H. Ling, J. Yu, and J. Peng. Salient region detection by UFO: Uniqueness, Focusness and Objectness. In *ICCV*, 2013. 1, 2, 3, 6, 7

[17] C. Kim, A. Hornung, S. Heinzle, W. Matusik, and M. Gross. Multi-perspective stereoscopy from light fields. *ACM Trans. Graph.*, 30(6):190:1–190:10, December 2011. 1

[18] C. Lang, T. V. Nguyen, H. Katti, K. Yadati, M. Kankanhalli, and S. Yan. Depth matters: influence of depth cues on visual saliency. In *Proceedings of the 12th European conference on Computer Vision - Volume Part II*, ECCV'12, pages 101–115, 2012. 2

[19] M. Levoy. Light fields and computational imaging. *Computer*, 39(8):46–55, 2006. 3

[20] F. Li and F. Porikli. Harmonic variance: A novel measure for in-focus segmentation. In *BMVC*, 2013. 4

[21] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, and H. Shum. Learning to detect a salient object. *TPAMI*, 33(2):353–367, 2011. 2, 5

[22] L. Mai, Y. Niu, and F. Liu. Saliency aggregation: A data-driven approach. In *CVPR*, 2013. 2

[23] A. Maki, P. Nordlund, and J.-O. Eklundh. A computational model of depth-based attention. In *ICPR*, 1996. 2

[24] A. Maki, P. Nordlund, and J.-O. Eklundh. Attentional scene segmentation: Integrating depth and motion. *Computer Vision and Image Understanding*, 78(3):351–373, 2000. 2

[25] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan. Light Field Photography with a Hand-Held Plenoptic Camera. Technical report, Apr. 2005. 1, 3

[26] Y. Niu, Y. Geng, X. Li, and F. Liu. Leveraging stereopsis for saliency analysis. In *CVPR*, 2012. 2, 7

[27] C. Rother, V. Kolmogorov, and A. Blake. "grabcut": interactive foreground extraction using iterated graph cuts. In *ACM SIGGRAPH 2004 Papers*, SIGGRAPH '04, pages 309–314, 2004. 1

[28] M. Rubinstein, A. Shamir, and S. Avidan. Improved seam carving for video retargeting. *ACM Transactions on Graphics (SIGGRAPH)*, 27(3):1–9, 2008. 1

[29] X. Shen and Y. Wu. A unified approach to salient object detection via low rank matrix recovery. In *CVPR*, pages 853–860, 2012. 2, 5, 6

[30] B. Suh, H. Ling, B. B. Bederson, and D. W. Jacobs. Automatic thumbnail cropping and its effectiveness. In *Proceedings of the 16th annual ACM symposium on User interface software and technology*, UIST '03, pages 95–104, 2003. 1

[31] R. Valenti, N. Sebe, and T. Gevers. Image saliency by isocentric curvedness and color. In *IEEE International Conference on Computer Vision*, 2009. 2

[32] S. Wanner, C. Straehle, and B. Goldluecke. Globally consistent multi-label assignment on the ray space of 4d light fields. In *CVPR*, 2013. 1

[33] Y. Wei, F. Wen, W. Zhu, and J. Sun. Geodesic saliency using background priors. In *ECCV*, 2012. 2, 4

[34] Q. Yan, L. Xu, J. Shi, and J. Jia. Hierarchical saliency detection. In *CVPR*, 2013. 2, 6, 7

[35] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang. Saliency detection via graph-based manifold ranking. In *CVPR*, 2013. 1, 2, 6, 7

[36] J. Ye and J. Yu. Ray geometry in non-pinhole cameras: a survey. *The Visual Computer*, pages 1–20, 2013. 3

[37] Y. Zhai. Visual attention detection in video sequences using spatiotemporal cues. *ACM MM*, 2006. 6