Learning Uncertain Convolutional Features for Accurate Saliency Detection
Supplementary Document

1. The UCF architecture

Tab. 1 shows the whole architecture of our proposed UCF model. The entire network forms an encoder-decoder FCN. All convolutional layers (Conv) are followed with batch normalization (BN) and ReLU activation layers. **#Kernel** refers to the number of kernels in each layer. **Output size** column shows the output size of each building block. We employ R-Dropout after several convolutional layers, i.e., Conv1-2, Conv2-2, Conv3-3, Conv4-3, Conv5-3, Conv5-1-D, Conv4-1-D, Conv3-1-D, Conv2-1-D, Conv1-1-D layers. The hybrid upsampling is obtained by three steps: (1) the low resolution feature maps are first up-scaled as $Deconv$ by 4×4 deconvolution with padding 1 and stride 2; (2) the low resolution feature maps are also up-scaled as $Inter$ by the bilinear interpolation; (3) $Deconv$ and $Inter$ are combined by element-wise summation.

2. Parameters for training the UCF model

We train the above UCF network on the augmented MSRA10K dataset [3] using the min-batch stochastic gradient descent (SGD) with a momentum, learning rate decay schedule. Tab. 2 shows the detailed parameter configurations.

3. Performance on other datasets

Tab. 3 and Tab. 4 show the results on all compared datasets in terms of Precision, Recall, F-measure and MAE. We also report the PR curves on the DUT-OMRON [6], HKU-IS [7], PASCAL-S [4] and SOD [5] in Fig. 1.

4. More samples on saliency detection

In this section, we report more saliency maps of our proposed algorithm and other saliency detection algorithms. Fig. 2, Fig. 3, Fig. 4, Fig. 5, Fig. 6, Fig. 7 and Fig. 8 show more examples from the DUT-OMRON [6], ECSSD [5], HKU-IS [7], PASCAL-S [4], SED1 [2], SED2 [1] and SOD [5], respectively.

5. More samples on upsampling methods

Fig. 2 shows more saliency maps generated by different upsampling methods.

References

[1] A. Borji. What is a salient object? a dataset and a baseline model for salient object detection. *IEEE Transactions on Image Processing*, 24(2):742–756, 2015. 1

[2] A. Borji, M.-M. Cheng, H. Jiang, and J. Li. Salient object detection: A benchmark. *IEEE Transactions on Image Processing*, 24(12):5706–5722, 2015. 1

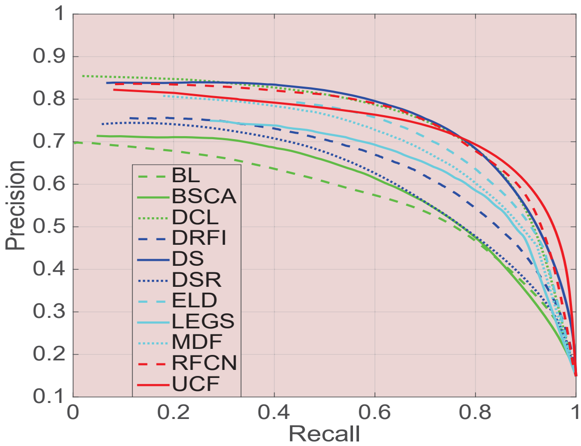
[3] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. S. Torr, and S.-M. Hu. Global contrast based salient region detection. *IEEE TPAMI*, 37(3):569–582, 2015. 1

[4] Y. Li, X. Hou, C. Koch, J. Rehg, and A. Yuille. The secrets of salient object segmentation. In *CVPR*, pages 280–287, 2014. 1

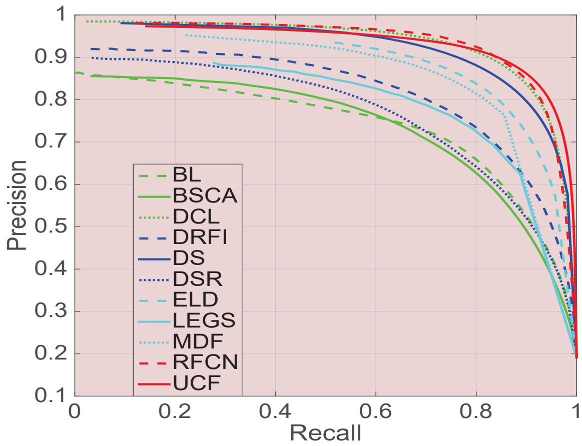
[5] Q. Yan, L. Xu, J. Shi, and J. Jia. Hierarchical saliency detection. In *CVPR*, pages 1155–1162, 2013. 1

[6] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang. Saliency detection via graph-based manifold ranking. In *CVPR*, pages 3166–3173, 2013. 1

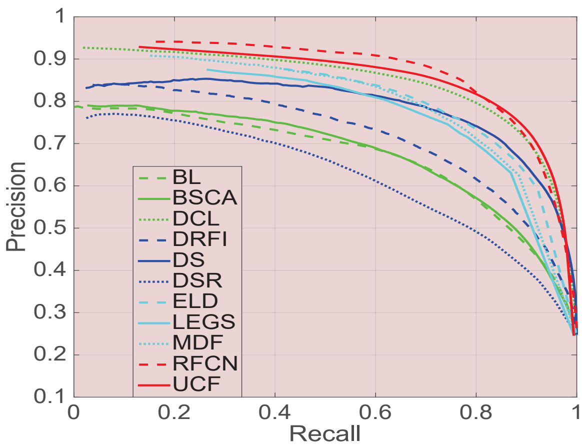
[7] R. Zhao, W. Ouyang, H. Li, and X. Wang. Saliency detection by multi-context deep learning. In *CVPR*, pages 1265–1274, 2015. 1



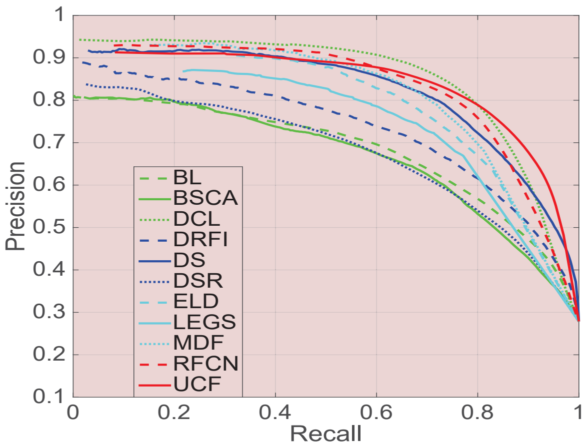
(a) DUT-OMRON



(b) HKU-IS



(c) PASCAL-S



(d) SOD

Figure 1. Performance of the proposed algorithm compared with other state-of-the-art methods.

Name	Type	#Kernel	Kernel size	Padding size	Stride	Output size
Input						448x448x3
Conv1-1	Conv+BN+ReLU	64	3x3	1	1	448x448x64
Conv1-2	Conv+BN+ReLU	64	3x3	1	1	448x448x64
Drop1-2	R-Dropout					448x448x64
Pool1	Pooling		2x2		2	224x224x64
Conv2-1	Conv+BN+ReLU	128	3x3	1	1	224x224x128
Conv2-2	Conv+BN+ReLU	128	3x3	1	1	224x224x128
Drop2-2	R-Dropout					224x224x128
Pool2	Pooling		2x2		2	112x112x128
Conv3-1	Conv+BN+ReLU	256	3x3	1	1	112x112x256
Conv3-2	Conv+BN+ReLU	256	3x3	1	1	112x112x256
Conv3-3	Conv+BN+ReLU	256	3x3	1	1	112x112x256
Drop3-3	R-Dropout					112x112x256
Pool3	Pooling		2x2		2	56x56x256
Conv4-1	Conv+BN+ReLU	512	3x3	1	1	56x56x512
Conv4-2	Conv+BN+ReLU	512	3x3	1	1	56x56x512
Conv4-3	Conv+BN+ReLU	512	3x3	1	1	56x56x512
Drop4-3	R-Dropout					56x56x512
Pool4	Pooling		2x2		2	28x28x512
Conv5-1	Conv+BN+ReLU	512	3x3	1	1	28x28x512
Conv5-2	Conv+BN+ReLU	512	3x3	1	1	28x28x512
Conv5-3	Conv+BN+ReLU	512	3x3	1	1	28x28x512
Drop5-3	R-Dropout					28x28x512
Pool5	Pooling		2x2		2	14x14x512
Upsample5	Hybrid upsampling	512	4x4	1	2	28x28x512
Conv5-3-D	Conv+BN+ReLU	512	3x3	1	1	28x28x512
Conv5-2-D	Conv+BN+ReLU	512	3x3	1	1	28x28x512
Conv5-1-D	Conv+BN+ReLU	512	3x3	1	1	28x28x512
Drop5-1-D	R-Dropout					28x28x512
Upsample4	Hybrid upsampling	512	4x4	1	2	56x56x512
Conv4-3-D	Conv+BN+ReLU	512	3x3	1	1	56x56x512
Conv4-2-D	Conv+BN+ReLU	512	3x3	1	1	56x56x512
Conv4-1-D	Conv+BN+ReLU	512	3x3	1	1	56x56x512
Drop4-1-D	R-Dropout					56x56x512
Upsample3	Hybrid upsampling	256	4x4	1	2	112x112x256
Conv3-3-D	Conv+BN+ReLU	256	3x3	1	1	112x112x256
Conv3-2-D	Conv+BN+ReLU	256	3x3	1	1	112x112x256
Conv3-1-D	Conv+BN+ReLU	256	3x3	1	1	112x112x256
Drop3-1-D	R-Dropout					112x112x56
Upsample2	Hybrid upsampling	128	4x4	1	2	224x224x128
Conv2-2-D	Conv+BN+ReLU	128	3x3	1	1	224x224x128
Conv2-1-D	Conv+BN+ReLU	128	3x3	1	1	224x224x128
Drop2-1-D	R-Dropout					224x224x128
Upsample1	Hybrid upsampling	64	4x4	1	2	448x448x64
Conv1-2-D	Conv+BN+ReLU	64	3x3	1	1	448x448x64
Conv1-1-D	Conv+BN+ReLU	64	3x3	1	1	448x448x64
Drop1-1-D	R-Dropout					448x448x64
Softmax	Conv	2	3x3	1	1	448x448x2

Table 1. The proposed UCF architecture. Output sizes are given for an example input of 448×448.

Name	Value
max iter	220000
momentum	0.9
weight decay	0.0001
batch size	8
learning rate	
type	step
base	1e-9
step size	122000

Table 2. Parameter configurations for fine-tuning the UCF network.

	DUT-OMRON				ECSSD				HKU-IS				PASCAL-S			
Methods	$Precis$	$Recall$	F_β	MAE	$Precis$	$Recall$	F_β	MAE	$Precis$	$Recall$	F_β	MAE	$Precis$	$Recall$	F_β	MAE
UCF	0.6302	0.8476	0.6283	0.1203	0.8760	0.8698	0.8517	0.0689	0.8171	0.9205	0.8232	0.0620	0.7671	0.8107	0.7413	0.1160
RFCN	0.6190	0.8292	0.6265	0.1105	0.8654	0.8223	0.8340	0.1069	0.8449	0.8677	0.8349	0.0889	0.7897	0.7795	0.7512	0.1324
DCL	0.7199	0.7296	0.6842	0.1573	0.9083	0.7772	0.8293	0.1495	0.8933	0.8211	0.8533	0.1359	0.8103	0.7021	0.7141	0.1807
DS	0.5953	0.8237	0.6028	0.1204	0.8569	0.8177	0.8255	0.1216	0.7775	0.8895	0.7851	0.0780	0.7118	0.7383	0.6590	0.1760
LEGS	0.5997	0.7157	0.5915	0.1335	0.8330	0.7372	0.7853	0.1180	0.7557	0.7235	0.7228	0.1193	-	-	-	-
MDF	0.6759	0.7026	0.6442	0.0916	0.8688	0.7332	0.8070	0.1049	0.8506	0.7581	0.8006	0.0957	0.8198	0.6200	0.7087	0.1458
ELD	0.5958	0.8285	0.6109	0.0924	0.8164	0.8581	0.8102	0.0796	0.7633	0.8643	0.7694	0.0741	0.7372	0.7795	0.7180	0.1232
BL	0.5280	0.6253	0.4988	0.2388	0.7579	0.6529	0.6841	0.2159	0.7034	0.6838	0.6597	0.2071	0.6649	0.5798	0.5742	0.2487
BSCA	0.5172	0.6713	0.5091	0.1902	0.7529	0.6920	0.7048	0.1821	0.6808	0.6951	0.6544	0.1748	0.6691	0.5899	0.6006	0.2229
DRFI	0.5536	0.7343	0.5504	0.1378	0.7803	0.7238	0.7331	0.1642	0.7492	0.7497	0.7218	0.1445	0.6869	0.6277	0.6182	0.2065
DSR	0.5354	0.6779	0.5242	0.1389	0.7260	0.6098	0.6621	0.1784	0.7138	0.6861	0.6772	0.1422	0.6300	0.5311	0.5575	0.2149

Table 3. The Precision, Recall, F-measure and MAE of different saliency detection methods on the DUT-OMRON, ECSSD, HKU-IS and PASCAL-S datasets. The best three results are shown in red, green and blue, respectively. The proposed methods rank first and second on these datasets.

	SED1				SED2				SOD			
Methods	$Precis$	$Recall$	F_β	MAE	$Precis$	$Recall$	F_β	MAE	$Precis$	$Recall$	F_β	MAE
UCF	0.9011	0.8627	0.8647	0.0631	0.8233	0.8760	0.8102	0.0680	0.8030	0.7197	0.7375	0.1478
RFCN	0.9044	0.7712	0.8502	0.1166	0.8125	0.7878	0.7667	0.1131	0.8260	0.6631	0.7426	0.1692
DCL	0.9442	0.7403	0.8546	0.1513	0.9310	0.6554	0.7946	0.1565	0.8875	0.6079	0.7413	0.1938
DS	0.8745	0.8213	0.8445	0.0931	0.7965	0.7832	0.7541	0.1233	0.7820	0.6608	0.6981	0.1889
LEGS	0.9206	0.7571	0.8542	0.1034	0.8050	0.6838	0.7358	0.1236	0.7811	0.5796	0.6834	0.1955
MDF	0.8991	0.8009	0.8419	0.0989	0.8645	0.7488	0.8003	0.1014	0.8650	0.5694	0.7205	0.1639
ELD	0.8960	0.8425	0.8715	0.0670	0.8027	0.7396	0.7591	0.1028	0.7830	0.6608	0.7116	0.1545
BL	0.8765	0.6838	0.7675	0.1849	0.7723	0.7182	0.7047	0.1856	0.7064	0.5125	0.5798	0.2668
BSCA	0.8771	0.7008	0.8048	0.1535	0.7611	0.6779	0.7062	0.1578	0.6729	0.5383	0.5835	0.2514
DRFI	0.8770	0.7357	0.8068	0.1480	0.7739	0.7393	0.7341	0.1334	0.7415	0.5494	0.6343	0.2238
DSR	0.8732	0.6697	0.7909	0.1579	0.7626	0.6887	0.7116	0.1406	0.6893	0.5298	0.5962	0.2339

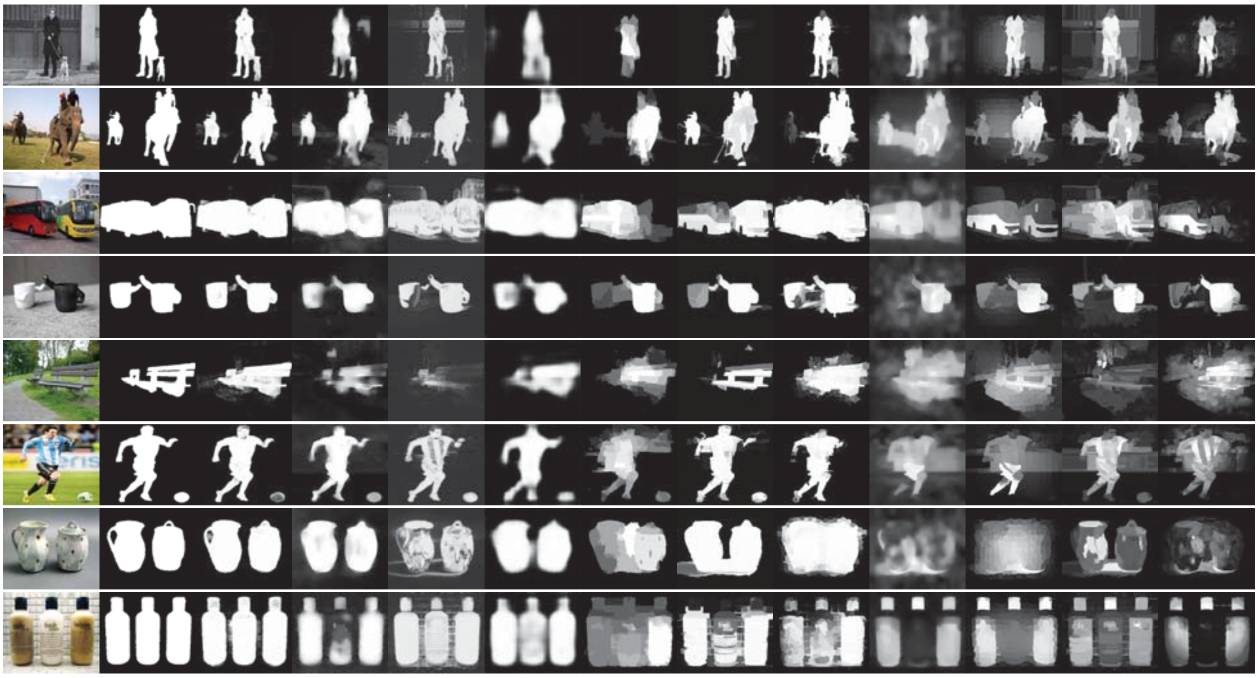
Table 4. The Precision, Recall, F-measure and MAE of different saliency detection methods on the SED1, SED2 and SOD datasets. The best three results are shown in red, green and blue, respectively. The proposed methods rank first and second on these datasets.

(a) (b) (c) (d) (e) (f) (g) (h) (i) (j) (k) (l) (m)

Figure 2. Comparison of saliency maps on the DUT-OMRON dataset. (a) Input images; (b) Ground truth; (c) Our UCF method; (d) RFCN; (e) DCL; (f) DS; (g) LEGS; (h) MDF; (i) ELD; (j) BL; (k) BSCA; (l) DRFI; (m) DSR.

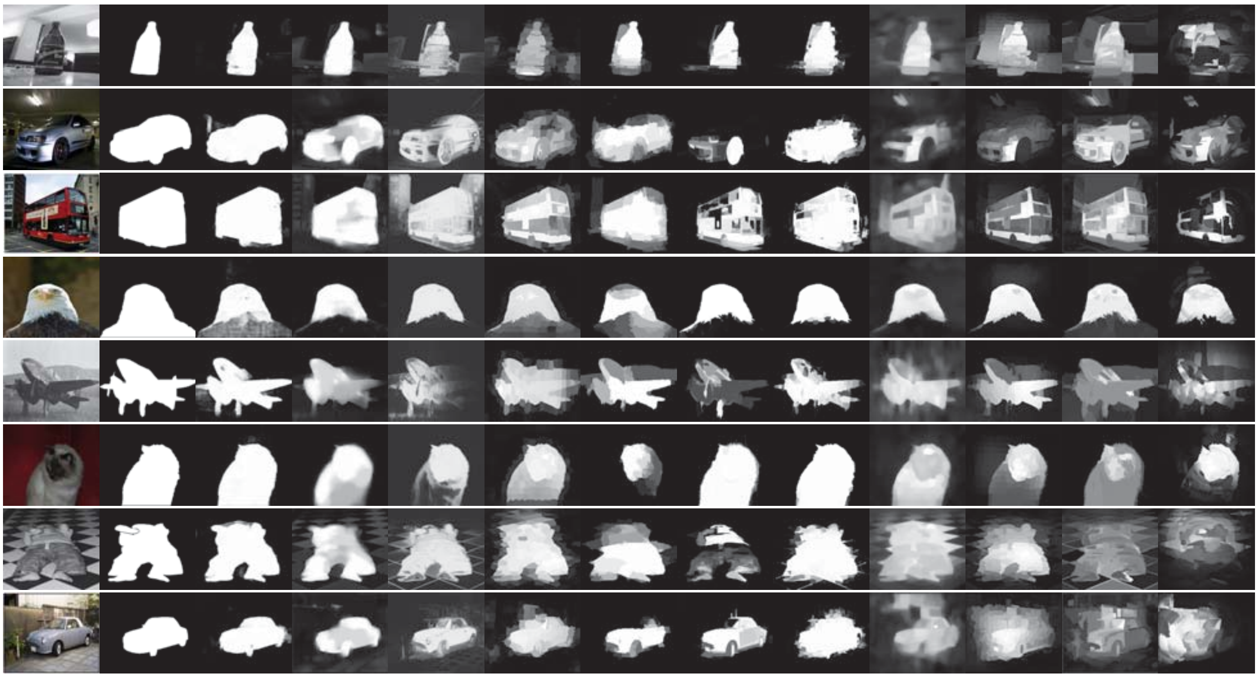
(a) (b) (c) (d) (e) (f) (g) (h) (i) (j) (k) (l) (m)

Figure 3. Comparison of saliency maps on the ECSSD dataset. (a) Input images; (b) Ground truth; (c) Our UCF method; (d) RFCN; (e) DCL; (f) DS; (g) LEGS; (h) MDF; (i) ELD; (j) BL; (k) BSCA; (l) DRFI; (m) DSR.



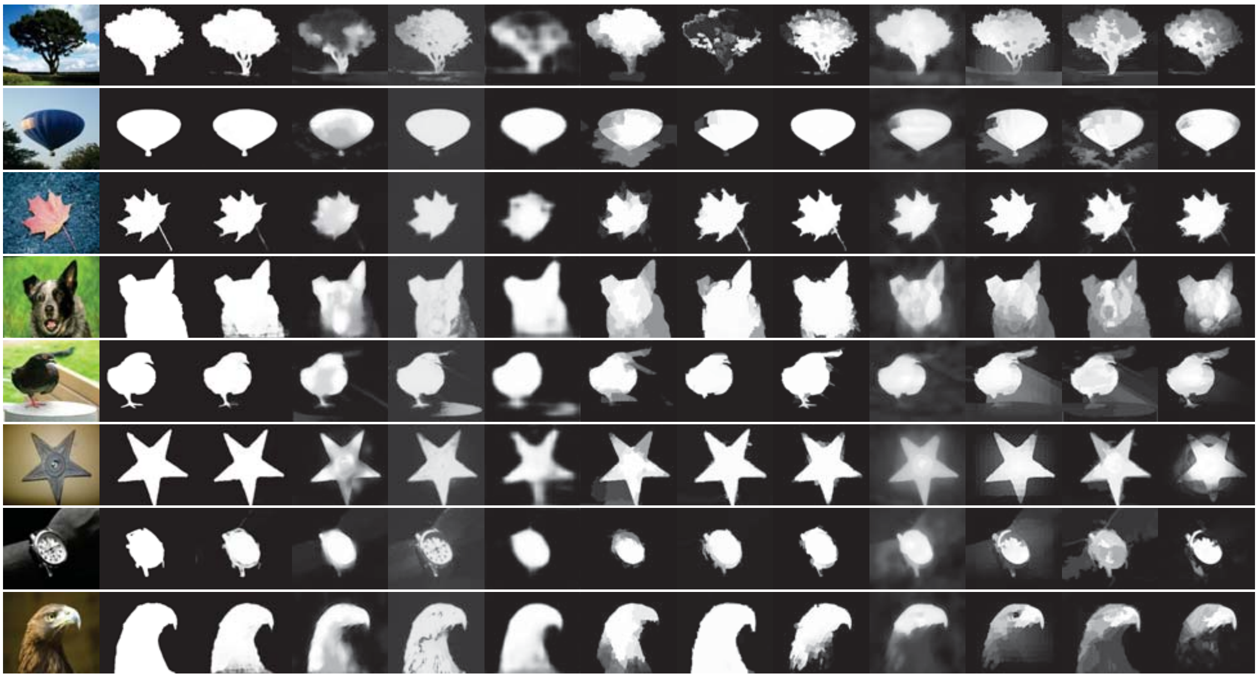
(a) (b) (c) (d) (e) (f) (g) (h) (i) (j) (k) (l) (m)

Figure 4. Comparison of saliency maps on the HKU-IS dataset. (a) Input images; (b) Ground truth; (c) Our UCF method; (d) RFCN; (e) DCL; (f) DS; (g) LEGS; (h) MDF; (i) ELD; (j) BL; (k) BSCA; (l) DRFI; (m) DSR.



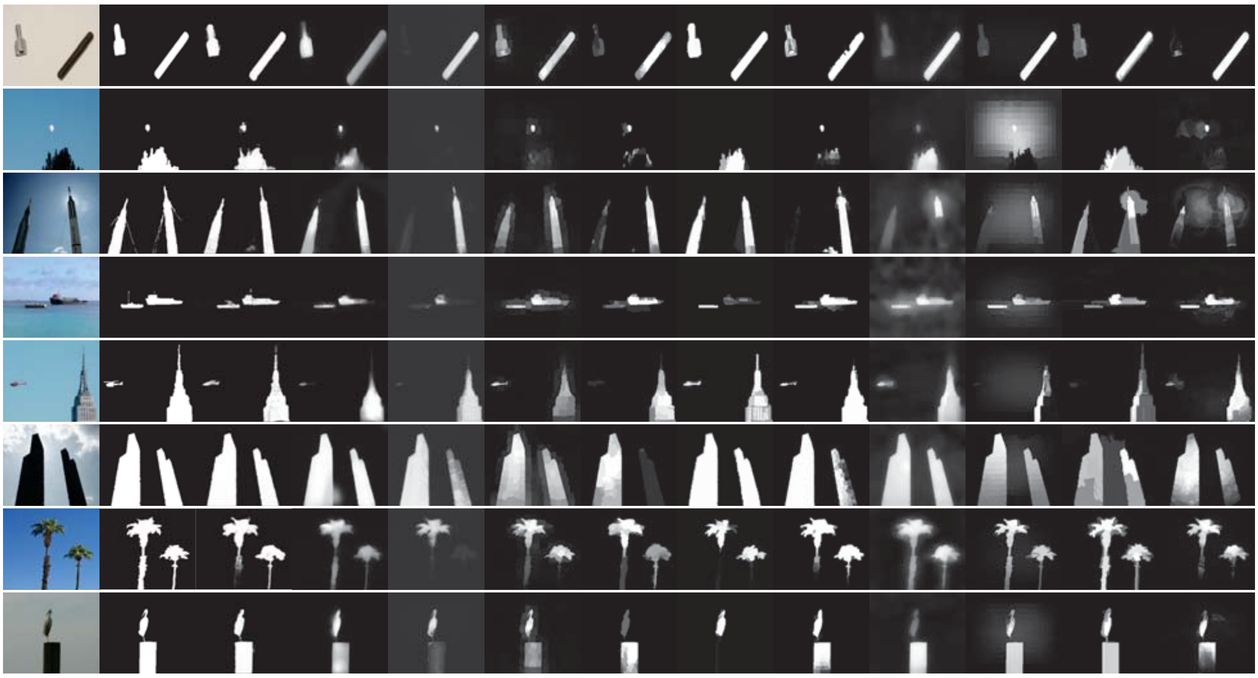
(a) (b) (c) (d) (e) (f) (g) (h) (i) (j) (k) (l) (m)

Figure 5. Comparison of saliency maps on the PASCAL-S dataset. (a) Input images; (b) Ground truth; (c) Our UCF method; (d) RFCN; (e) DCL; (f) DS; (g) LEGS; (h) MDF; (i) ELD; (j) BL; (k) BSCA; (l) DRFI; (m) DSR.



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j) (k) (l) (m)

Figure 6. Comparison of saliency maps on the SED1 dataset. (a) Input images; (b) Ground truth; (c) Our UCF method; (d) RFCN; (e) DCL; (f) DS; (g) LEGS; (h) MDF; (i) ELD; (j) BL; (k) BSCA; (l) DRFI; (m) DSR.



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j) (k) (l) (m)

Figure 7. Comparison of saliency maps on the SED2 dataset. (a) Input images; (b) Ground truth; (c) Our UCF method; (d) RFCN; (e) DCL; (f) DS; (g) LEGS; (h) MDF; (i) ELD; (j) BL; (k) BSCA; (l) DRFI; (m) DSR.

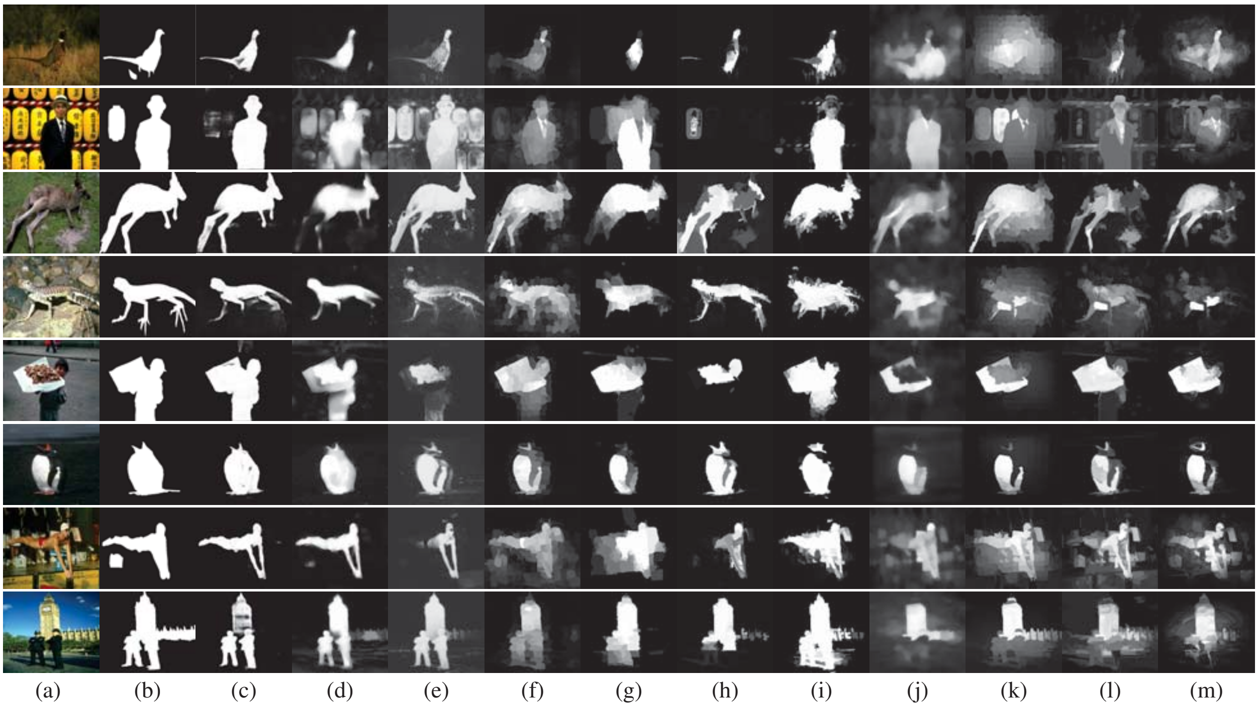


Figure 8. Comparison of saliency maps on the SOD dataset. (a) Input images; (b) Ground truth; (c) Our UCF method; (d) RFCN; (e) DCL; (f) DS; (g) LEGS; (h) MDF; (i) ELD; (j) BL; (k) BSCA; (l) DRFI; (m) DSR.

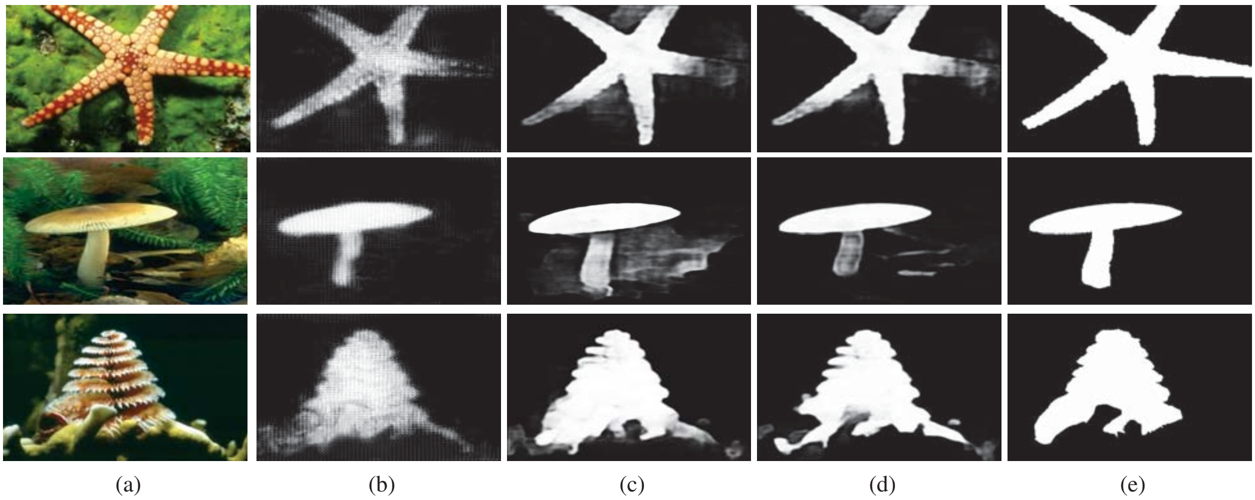


Figure 9. Comparison of different upsampling algorithms. (a) Input image; (b) Regular deconvolution; (c) Bilinear interpolation; (d) Our hybrid upsampling method; (e) Ground truth.