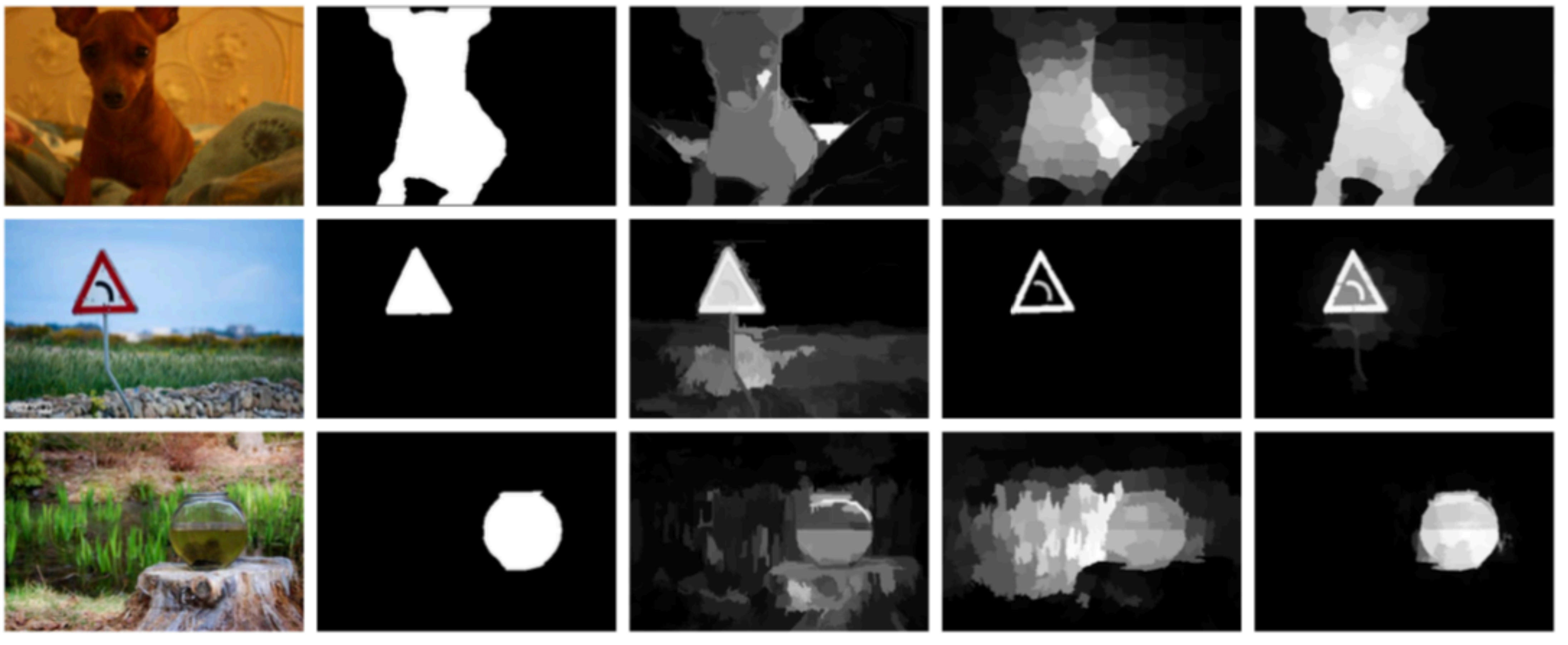
An Unsupervised Game-Theoretic Approach to Saliency Detection

> *这篇论文从一个新颖的博弈论的角度思考unsupervised saliency detection。*
> *首先，saliency detection问题被formulate为non-cooperative game，称为Saliency Game，*
> *其中图像区域是选择作为其pure strategies的“background”或“foreground”的player。*
> *通过利用多个线索并组合互补特征来构造一个payoff函数。*
> *根据提出的Saliency Game的Nash equilibrium中的每个区域的策略生成Saliency maps。*
> *其次，通过探索color feature和deep feature之间的互补关系，并提出一种iterative random walk算法，结合使用不同特征的Saliency Game产生的saliency maps。*
> *iterative random walk允许跨特征空间共享信息并检测很难检测的对象。*
> *本文在六个具有挑战性的数据集上进行的大量实验证明了提出的unsupervised算法与几种最先进的supervised算法相比的优越性。*

Ideas

- *提出了一种新的unsupervised Saliency Game来检测salient objects。采用了两个独立的priors来提高robustness。 game equilibria的性质可确保两个先验unsatisfactory时的准确性。*
- *利用从pre-trained的CNN中提取的semantically-rich的特征，所提出的方法能够识别复杂场景中的salient objects，而如果基于手工特征的传统方法可能失败（Figure 1(c)）。*



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

Fig. 1. Saliency detection results by several methods. (a) Input images, (b) Ground truth maps, (c) DRFI method [4], a supervised method based on handcrafted features, (d) MR method [5], an unsupervised method taking image boundary regions as background seeds, (e) Our method.

- *提出了跨两个特征空间（color feature和deep feature）的Iterative Random Walk算法，该算法利用color feature空间和deep feature空间之间的互补关系来进一步细化结果。*

Framework

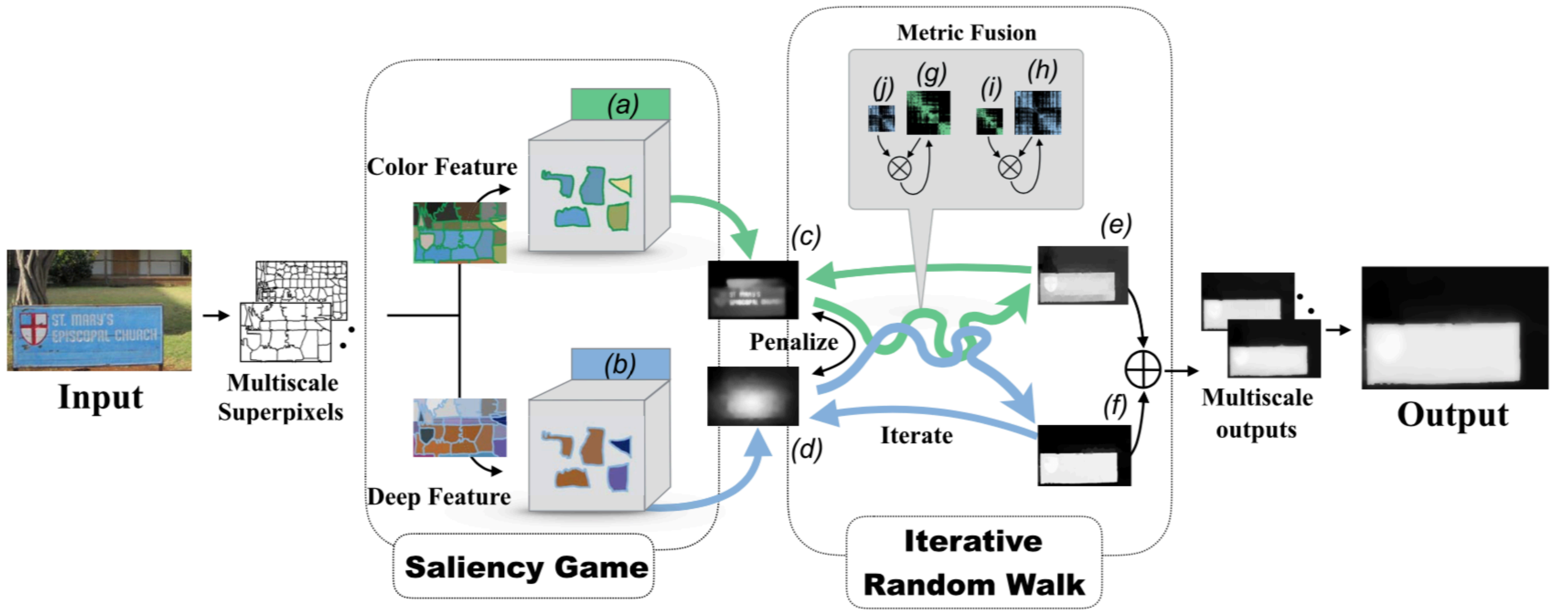


Fig. 2. The pipeline of our algorithm. The input image is segmented into superpixels in several scales. The following process is applied to each scale, and the average of the results in all scales is taken as the final saliency map. First, the *Saliency Game* among superpixels is formulated in two feature spaces ((a) and (b)), respectively to generate corresponding results ((c) and (d)). Second, an *Iterative Random Walk* is constructed to refine the results of the Saliency Game. Then outputs (e) and (f) are summed (with weights) as the result in one scale. In the Iterative Random Walk, complete affinity metric ((g) in deep feature space and (h) in color space) in one feature space is fused with neighboring affinity metric ((i) in color space and (j) in deep space) in another feature space. This is done to exploit the complementary relationship between the two feature spaces. Finally, we average the results in all scales to form the final saliency map.

Saliency Game

1. *首先通过SLIC算法将输入图像分割成N个superpixel，其作为Saliency Game中的players;*
2. *每个player选择“background”或“foreground”作为其pure strategy，其mixed strategy对应于此superpixel的saliency value;*
3. *在展示他们的strategy后，player可以根据自己和其他player的strategy获得一些payoff;*

Payoff Function：

$$\pi_{ij}(s_i, s_j) = \lambda_1 \cdot \mathrm{pos}_i(s_i) + \lambda_2 \cdot \mathrm{obj}_i(s_i) + \mathrm{spt}_{ij}(s_i, s_j), \quad (4)$$

$$\mathrm{pos}_i(s_i) = \begin{cases} \frac{1}{N}\exp\{-(x_i - x_0)^2 - (y_i - y_0)^2\} \text{ if } s_i = 1, \\ \frac{1}{N}(1-\exp\{-(x_i-x_0)^2 - (y_i-y_0)^2\}) \text{ if } s_i = 0. \end{cases} \quad (5)$$

$$
\mathrm{obj}_i(s_i) = \begin{cases} \dfrac{1}{N \cdot N_o} \displaystyle\sum_{j=1}^{N_o} \dfrac{\sum_{x,y} O_j(x,y) \times P_i(x,y)}{\sum_{x,y} P_i(x,y)} & \text{if } s_i = 1, \\ \dfrac{1}{N}\Big(1 - \dfrac{1}{N_o} \displaystyle\sum_{j=1}^{N_o} \dfrac{\sum_{x,y} O_j(x,y) \times P_i(x,y)}{\sum_{x,y} P_i(x,y)}\Big) & \text{otherwise.} \end{cases} \tag{6}
$$

$$
\mathrm{spt}_{ij}(s_i, s_j) = \begin{cases} A(i,j) - \dfrac{\alpha}{N} \displaystyle\sum_{k=1}^{N} A(i,k) & \text{if } s_i = s_j, \\ 0 & \text{if } s_i \neq s_j, \end{cases} \tag{7}
$$

> *本文在提出的Saliency Game的Nash equilibrium中使用每个player的mixed strategy作为输出Saliency maps中该superpixel的saliency value。*
> *这种equilibrium对应于稳定状态，其中每个player在剩余player的策略保持固定时发挥最大化其自身payoff的策略，这提供了全局合理的saliency detection结果。*

Iterative Random Walk

> *传统的color features具有高分辨率，因此在color space中生成的saliency maps是详细的并具有清晰的边界。*
> *但由于缺乏high-level信息，有时他们无法准确定位目标（Figure 4(c)）。相反，由于deep features很好地编码了对象的high-level概念，因此在deep features空间中生成的saliency映射能够在图像中找到正确的salient objects。但是由于几层convolution和pooling，这些特征太粗糙了。因此生成的saliency maps是不明显的（Figure 4(d)）。*

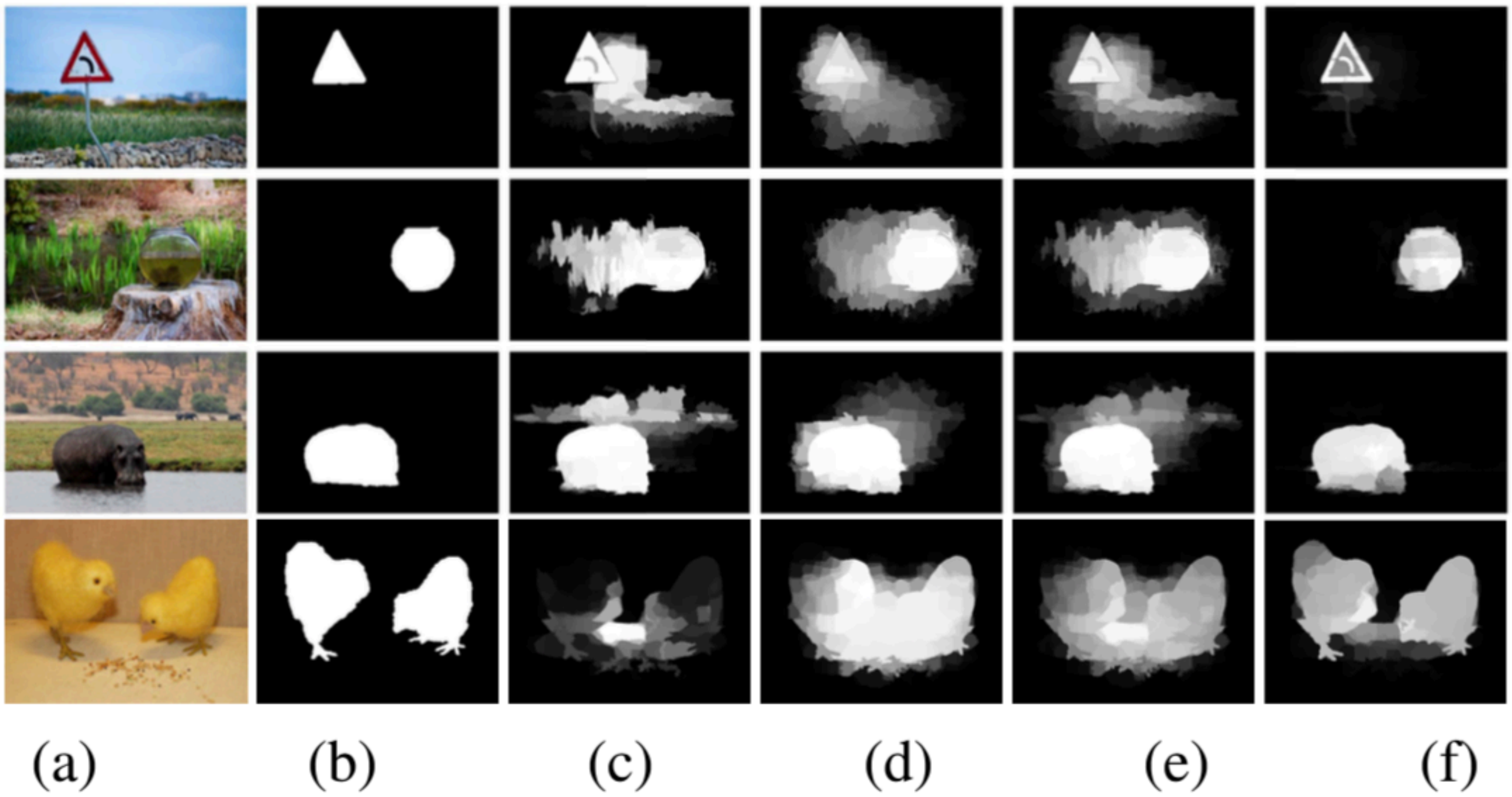


Fig. 4. Effect of the Iterative Random Walk. (a) Input images. (b) Ground Truth maps. (c) Saliency maps generated by our Saliency Game in the color feature space. (d) Saliency maps generated by our Saliency Game in the deep feature space. (e) The weighted summation of (c) and (d). (f) Saliency maps after refinement by the Iterative Random Walk proposed in this section.

> *本文使用两个这特征互补以获得更好的结果。但是，如Figure 4(e)所示，尽管两者的加权和略好一些，但并不令人满意。*
> *为了解决这个问题，本文受metric fusion的启发，提出了一种Iterative Random Walk方法，以最好地利用这两个互补的特征空间。*

1. *在所提出的Iterative Random Walk中，两个特征空间中的metrics被融合。*

With superpixels as nodes, a neighbor graph and a complete graph are constructed in both feature space (deep and color features). The affinity between two superpixels is assigned to the edge weight. Four weight matrices are defined:

- $\mathcal{W}^d$ and W^d: weight matrices of neighbor and complete graphs in the deep feature space, respectively.
- $\mathcal{W}^c$ and W^c: weight matrices of neighbor and complete graphs in the color space, respectively.

Following [28], we firstly fuse these four affinity matrices as follows,

$$\begin{cases} P^d_{(t+1)} = \mathcal{P}^c \times P^d_{(t)} \times \mathcal{P}^c, \\ P^c_{(t+1)} = \mathcal{P}^d \times P^c_{(t)} \times \mathcal{P}^d, \end{cases} \tag{16}$$

in which t is the number of iterations, $\times$ denotes matrix multiplication, $P^d_{(0)}(i, j) = W^d(i, j)/\sum_{j=1}^{N} W^d(i, j)$, $\mathcal{P}^d_{(0)}(i, j) = \mathcal{W}^d(i, j)/\sum_{j=1}^{N} \mathcal{W}^d(i, j)$. $P^c_{(0)}$ and $\mathcal{P}^c_{(0)}$ are defined similarly but using $W^d(i, j)$ and $\mathcal{W}^d(i, j)$.

2. 此外还制作了两个random walk energy function由最新的结果互相regularized。

Then, using the fused affinity matrices, we let the results in the two feature spaces regularize each other. Two random walk energy functions are defined as follows,

$$E^{d}_{(t+1)}(\boldsymbol{l}) = \sum_{i,j} P^{d}_{(t)}(i, j)(l_i - l_j)^2 + \beta \sum_{i=1}^{N}(l_i - l^{c}_{i(t)})^2, \quad (17)$$

$$E^{c}_{(t+1)}(\boldsymbol{l}) = \sum_{i,j} P^{c}_{(t)}(i, j)(l_i - l_j)^2 + \beta \sum_{i=1}^{N}(l_i - l^{d}_{i(t)})^2, \quad (18)$$

in whcih $\boldsymbol{l}$ is the label vector, l_i is the i-th superpixel's label, and β is a parameter.

By minimizing the two energy functions above, we have,

$$\boldsymbol{l}^{d}_{(t+1)} = \arg\min_{\boldsymbol{l}} E^{d}_{(t+1)}(\boldsymbol{l}) = (L^{d}_{(t+1)} + \beta I)^{-1}\boldsymbol{l}^{c}_{(t)}, \quad (19)$$

$$\boldsymbol{l}^{c}_{(t+1)} = \arg\min_{\boldsymbol{l}} E^{c}_{(t+1)}(\boldsymbol{l}) = (L^{c}_{(t+1)} + \beta I)^{-1}\boldsymbol{l}^{d}_{(t)}, \quad (20)$$

where L is the Laplacian matrix. $\boldsymbol{l}^{c}_{(0)}$ and $\boldsymbol{l}^{d}_{(0)}$ are set to the results of the Saliency Game stated in Section III. After T rounds, the iteration converges and the final saliency map is obtained as:

$$S = (1 - \rho) \cdot \boldsymbol{l}^{c}_{(T)} + \rho \cdot \boldsymbol{l}^{d}_{(T)}, \quad (21)$$

in which ρ controls the weight of the two results.

Experimental Results

Ablation Study

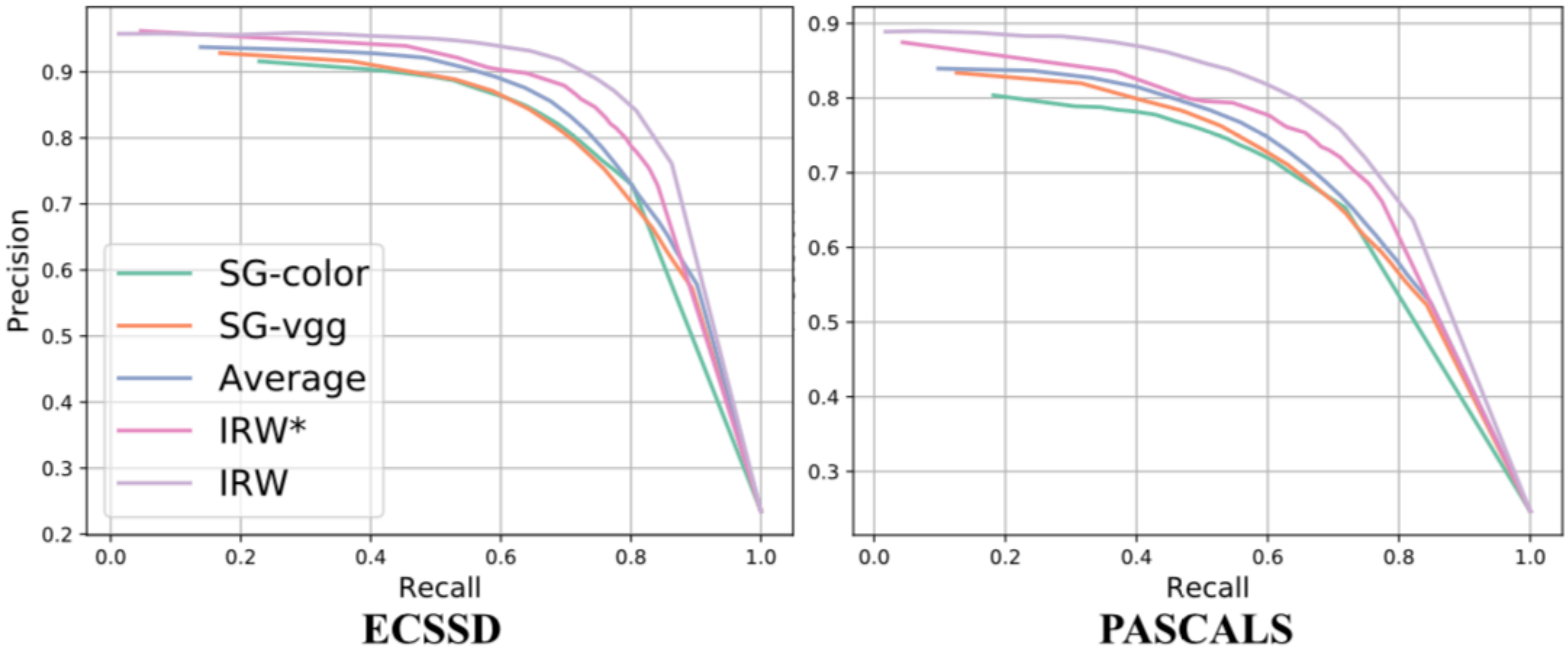


Fig. 5. Effect of each component of the proposed method. Left) PR curves on ECSSD dataset. Right) PR curves on PASCAL-S dataset. SG-color: Saliency Game with color features. SG-vgg: Saliency Game with VGG features. Average: a weighted sum of SG-color and SG-vgg. IRW*: Saliency Game refined by the Iterative Random Walk without metric cross fusion. IRW: Saliency Game refined by the Iterative Random Walk with metric cross fusion.

Computational Complexity

TABLE I

THE AVERAGE RUN-TIME (IN SECONDS) OF OUR METHODS AND SEVERAL STATE-OF-THE-ARTS. SUPERVISED METHODS ARE IN BOLD

Methods	BL	DSR	wCO	LR	HS
Run-time Code	21.5161 Matlab	3.4796 Matlab	0.1484 Matlab	10.0259 Matlab	0.3821 EXE
Methods	RC	**DRFI**	**MCDL**	**LEGS**	Ours
Run-time Code	0.1360 C	8.0104 Matlab	2.2521 Python	1.9050 Matlab+C	1.3819 Matlab

Comparison With State-of-the-Art Methods

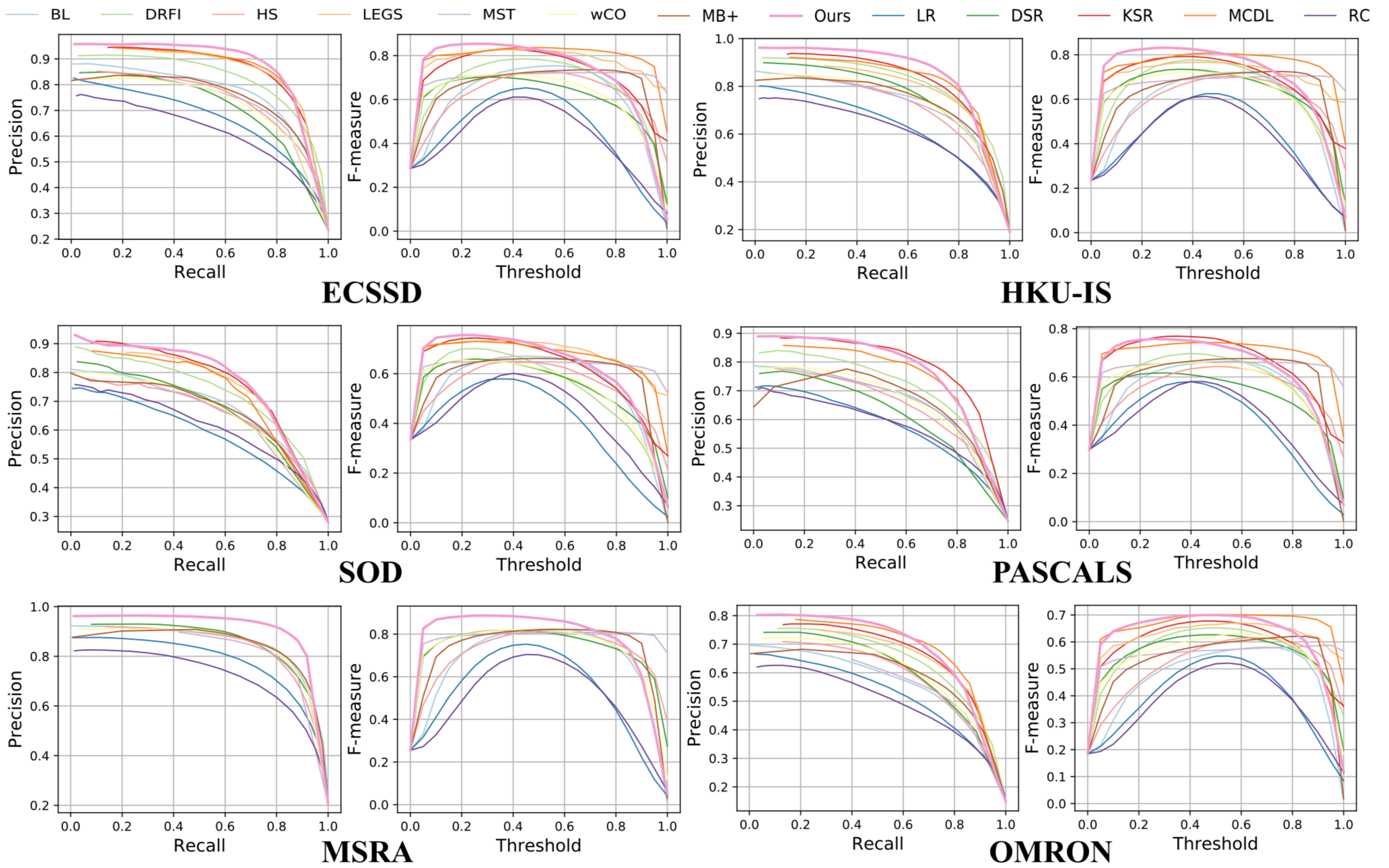


Fig. 6. Comparison of our method against-state-of-the-arts in terms of PR curves and F-measure curves.

TABLE II

F-Measure Scores (the Larger the Better). Ours-F Denotes our Methods With FCN Features, and Ours-V Denotes our Methods With VGG Features. The Best and the Second Best Results are Shown in Red and Green, Respectively. Supervised Methods are Marked in Bold

dataset	ECSSD	PASCALS	MSRA	HKU-IS	SOD	OMRON
BL	.6838	.5742	.7840	.6597	.5723	.4989
DSR	.6618	.5575	.7841	.6774	.5962	.5243
HS	.6347	.5314	.7671	.6359	.5212	.5108
RC	.4560	.4039	.5754	.5008	.4184	.4058
wCO	.6764	.5999	.7937	.6770	.5987	.5277
LR	.5629	.4791	.6940	.5546	.4843	.4531
MST	.6778	.6095	.7803	.6574	.5916	.5178
MB+	.6746	.6077	.7911	.6641	.5894	.5195
DRFI	.7329	.6182	-	.7219	.6470	.5505
MCDL	7959	.6912	-	.7573	.6772	.6250
LEGS	.7851	-	-	.7229	.6834	.5916
KSR	.7817	.7039	-	.7468	.6679	.5911
Ours-F	.8215	.7062	.8666	.8015	.6896	.5981
Ours-V	.8214	.6905	.8693	.7979	.6852	.6190

TABLE III

AUC SCORES (THE LARGER THE BETTER). OURS-F DENOTES OUR METHODS WITH FCN FEATURES, AND OURS-V DENOTES OUR METHODS WITH VGG FEATURES. SUPERVISED METHODS ARE MARKED IN BOLD. THE BEST AND THE SECOND BEST RESULTS ARE SHOWN IN RED AND GREEN, RESPECTIVELY

dataset	ECSSD	PASCALS	MSRA	HKU-IS	SOD	OMRON
BL	.9143	.8671	.9535	.9140	.8503	.8778
DSR	.8619	.8118	.9382	.9008	.8208	.8787
HS	.8838	.8362	.9279	.8782	.8145	.8586
RC	.8342	.8139	.8951	.8530	.7924	.8476
wCO	.8814	.8482	.9360	.8952	.8026	.8846
LR	.8619	.8119	.9225	.8645	.7787	.8556
MST	.8713	.8307	.9133	.8814	.7858	.8529
MB+	.9026	.8608	.9491	.9159	.8319	.8891
DRFI	.9404	8950	-	.9435	.8813	9157
MCDL	.9186	.8699	-	.9175	.8163	.9014
LEGS	.9235	-	-	.9026	.8268	.8841
KSR	.9268	.9012	-	.9099	.8403	.8921
Ours-F	.9272	.8724	.9583	.9183	.8481	.8869
Ours-V	.9091	.8614	.9463	.8974	.8184	.8680

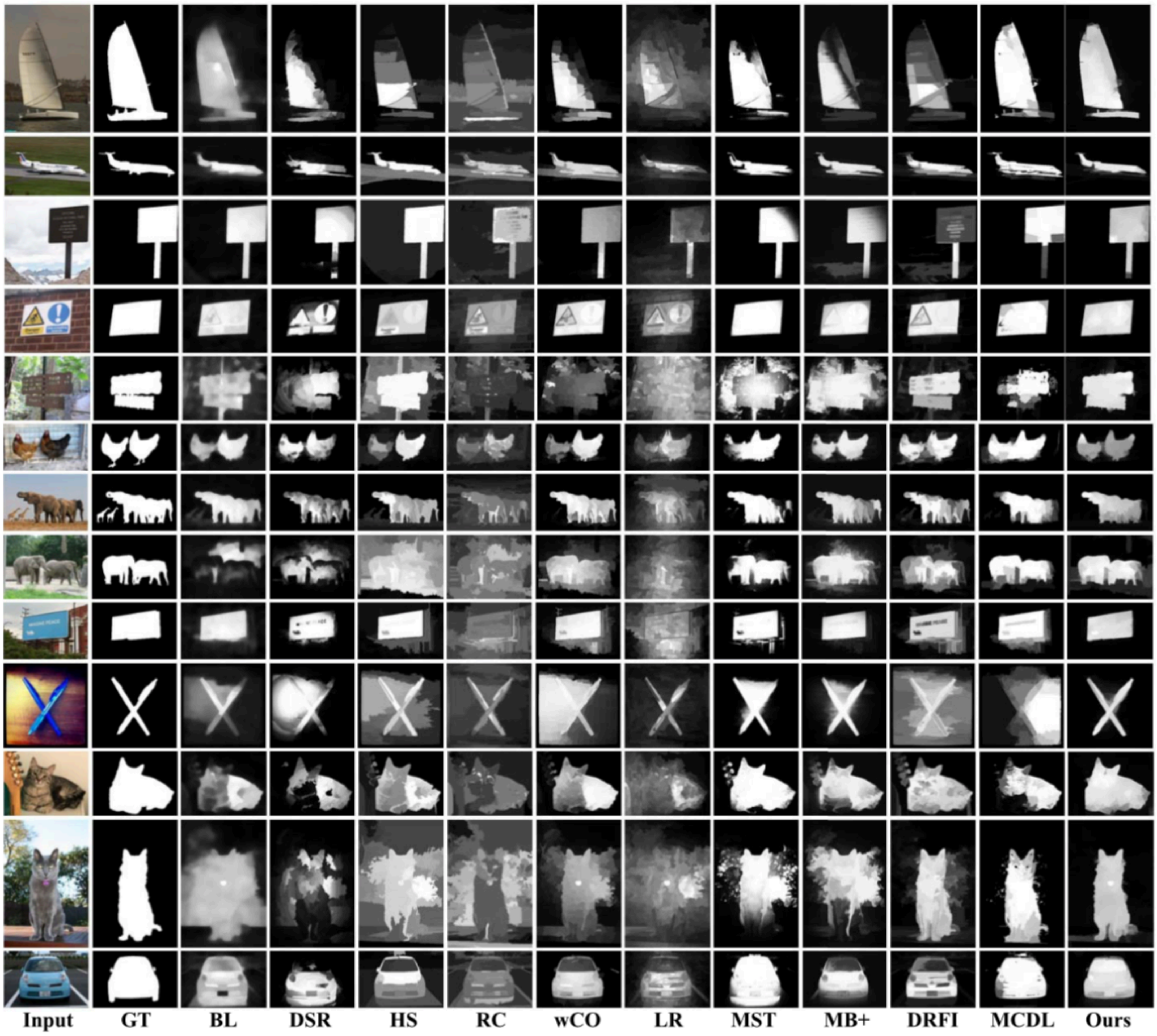


Fig. 7. Visual comparison of our method against state-of-the-arts.

Sensitivity Analysis

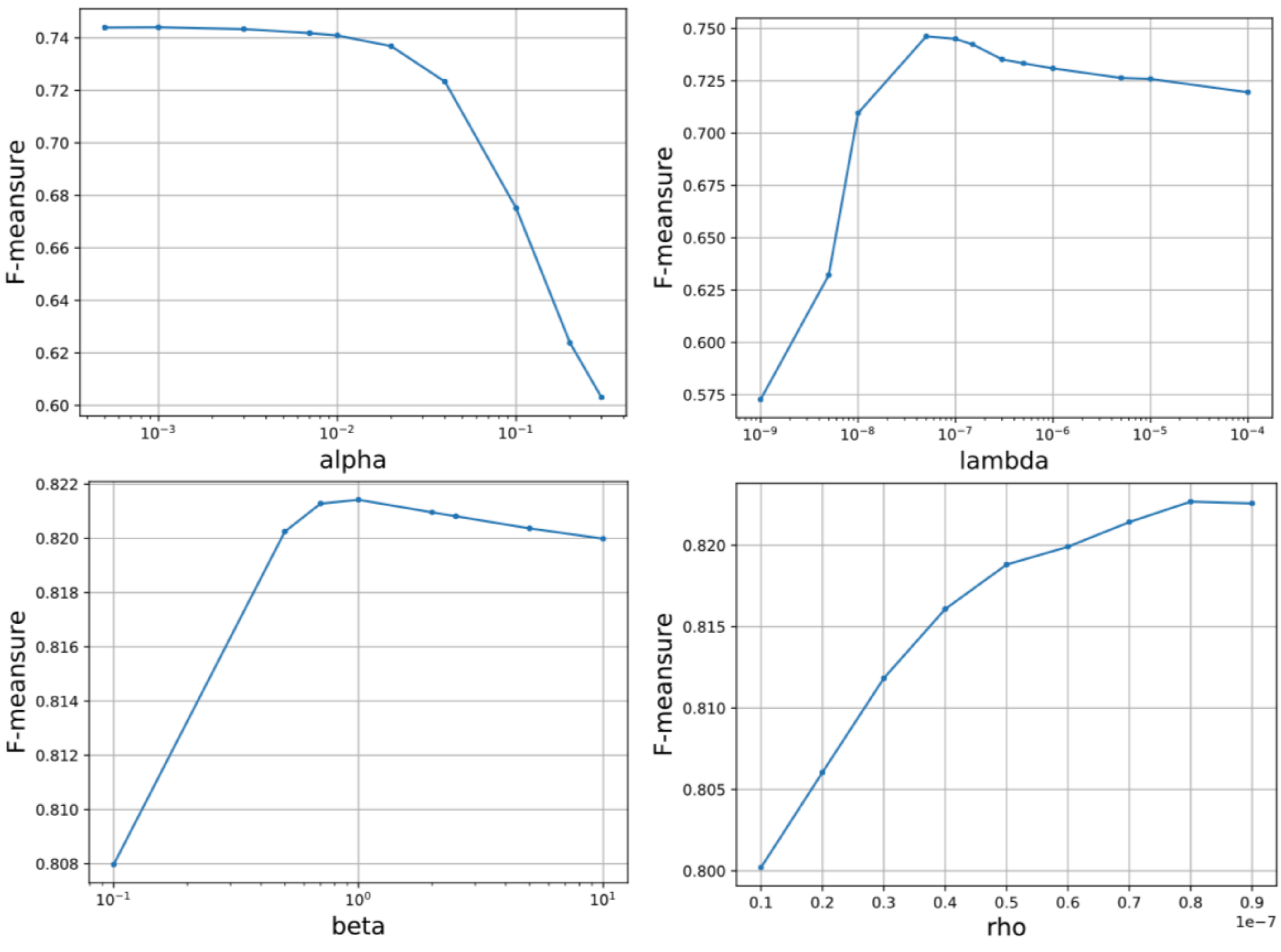


Fig. 8. Sensitivity of the Saliency Game to parameters α, λ and Iterative Random Walk to β, ρ, evaluated on ECSSD dataset.

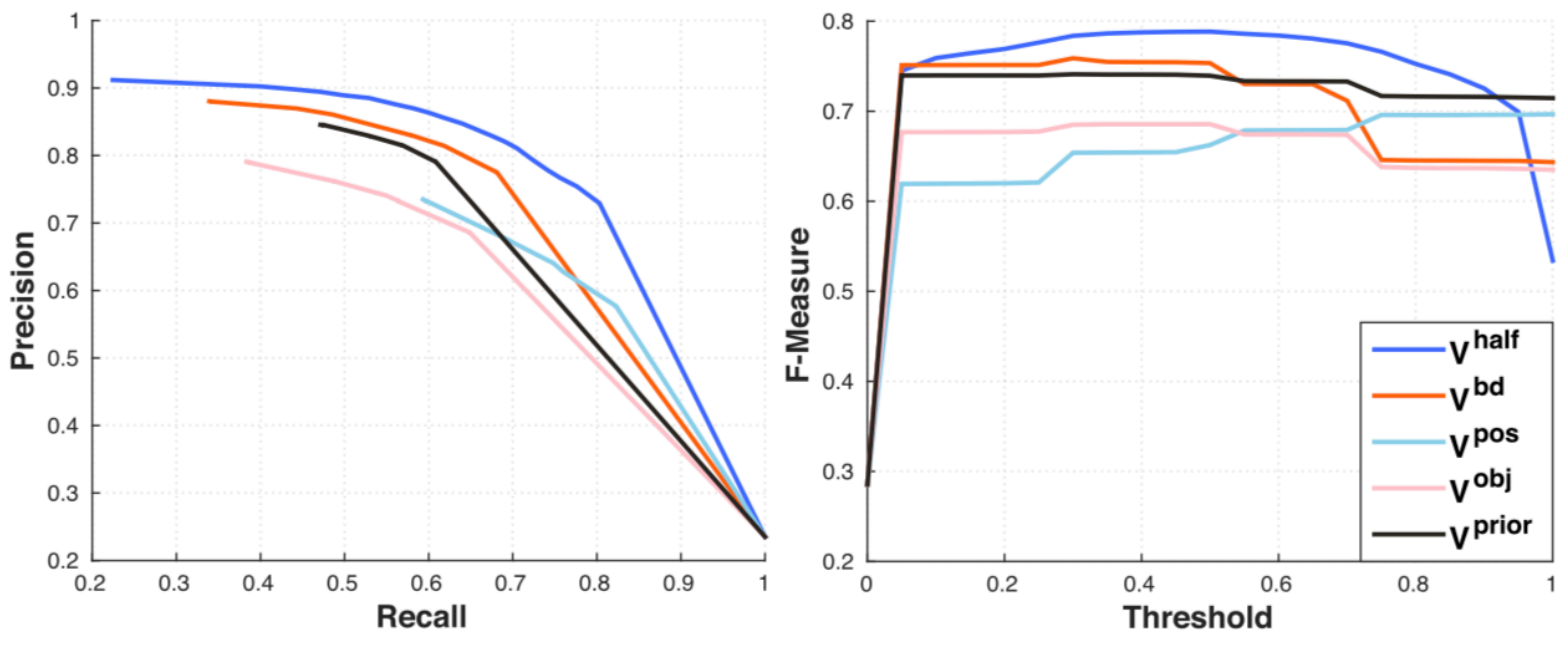


Fig. 9. Effect of different initialization, evaluated on PASCALS dataset. V^{half}, V^{pos}, V^{obj}, V^{bd} and V^{prior}: Saliency detection results corresponding to different initial state V^{half}, V^{pos}, V^{obj}, V^{bd} and V^{prior}, respectively.