Multi-Type Self-Attention Guided Degraded Saliency Detection

Ziqi Zhou,[1] **Zheng Wang,**[1*] **Huchuan Lu,**[2,4] **Song Wang,**[1,3] **Meijun Sun**[1]
[1]College of Intelligence and Computing, Tianjin University, Tianjin, China
[2]School of Information and Communication Engineering, Dalian University of Technology, Dalian, China
[3]Department of Computer Science and Engineering, University of South Carolina, USA
[4]Peng Cheng Laboratory
{ziqizhou, wzheng, sunmeijun}@tju.edu.cn, lhchuan@dlut.edu.cn, songwang@cec.sc.edu

Abstract

Existing saliency detection techniques are sensitive to image quality and perform poorly on degraded images. In this paper, we systematically analyze the current status of the research on detecting salient objects from degraded images and then propose a new multi-type self-attention network, namely MSANet, for degraded saliency detection. The main contributions include: 1) Applying attention transfer learning to promote semantic detail perception and internal feature mining of the target network on degraded images; 2) Developing a multi-type self-attention mechanism to achieve the weight recalculation of multi-scale features. By computing global and local attention scores, we obtain the weighted features of different scales, effectively suppress the interference of noise and redundant information, and achieve a more complete boundary extraction. The proposed MSANet converts low-quality inputs to high-quality saliency maps directly in an end-to-end fashion. Experiments on seven widely-used datasets show that our approach produces good performance on both clear and degraded images.

Introduction

Salient object detection (SOD) aims to distinguish the visually distinctive regions or objects and then segment the foreground targets out from the background. It plays an important role in content-based image/video understanding, such as visual tracking (Avytekin, Cricri, and Aksu 2018), and person re-ID (Zhao, Oyang, and Wang 2016).

Early SOD researches capture local details and global context based on hand-crafted features, *e.g., color, texture, luminance* (Harel, Koch, and Perona 2007). However, due to the lack of guidance at the semantic level, the developed methods perform poorly in many complex scenarios. Recently, thanks to the capability of extracting both low-level details and high-level semantic information simultaneously, convolutional neural networks (CNNs) have become the main force of SOD. However, many existing CNN-based SOD methods mainly focus on single image, and most of them are "picky-eaters", which only accept clean and high-quality images. Nevertheless, due to the complexity of the natural environment and the diversity of acquisition methods, we may get many degraded images with lower quality in practice. Typical image degradations include *e.g. low resolution, blur, noise,* and *atomization* etc. On degraded images, the robustness of the SOD methods is inevitably influenced by losing information both inside and along the boundary of the objects of interest.

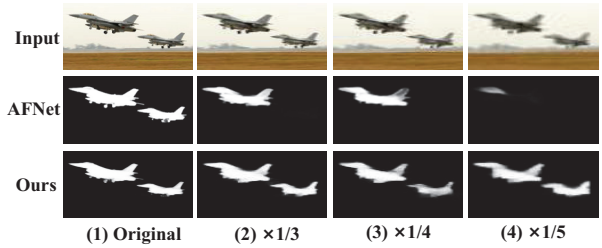

Figure 1: An illustration of salient object detection on degraded images. As the resolution decreases, AFNet (Feng, Lu, and Ding 2019) gradually loses the subject, while our MSANet can still pinpoint salient objects. $1/r$ means to reduce the length and width by a factor of r.

Similar to the discussion in (Guo et al. 2019), to address such a problem, a straightforward approach is to use the degraded image, denoted as x_d, as adversarial examples, together with the clear image, denoted as x_o, to form the training dataset. However, obvious gaps exist between x_o and x_d in terms of semantics and details and simple adversarial training cannot effectively improve the performance (see the performance of adv-Amulet in the later experiments). Another idea is to enhance the robustness by improving the quality of x_d. However, given the diversity of degradation types and the randomicity of the degradation levels, general image restoration algorithms cannot handle all the situations well, not to mention that pre-processing of image restoration can be very time-consuming and storage demanding. Furthermore, such a pre-processing may not be combined to the end-to-end CNN training and this is usually not desirable in SOD by forwarding the errors between network modules.

In this paper, we propose a new **MSANet** for degraded salient object detection by incorporating multi-type self-

*Corresponding author

attentions, e.g, both global and local ones, which directly outputs the salient object segment from the degraded images in an end-to-end manner (Some visualization results are shown in figure.1). In particular, we introduce the "teacher-student" network to extract better features from degraded images. Under the guidance of the teacher network, our student net can implicitly capture the hidden mode of x_d and extract more fully degraded features, thereby improving the accuracy of latter saliency inference. To sum up, the main contributions of this paper are:

- Systematically analyzing the current status of SOD on degraded images and developing a new MSANet, which greatly improves the SOD performance on degraded images without increasing the time cost.
- Proposing a multi-type self-attention network to select features from both global and local perspectives, thus to effectively suppress distracters while enhancing positive items.
- Proposing an attention transfer learning network to improve degraded SOD by narrowing the semantic gap between the target network and the source network.

Related work

Skip Connection Structure

Skip connection constructs a top-down path to transmit high-level semantic knowledge to shallower layers for enriched feature maps construction. For example, Zhang et al. (2017a) recursively embedded edge-ware low-level feature maps and the low-resolution predictions to promote boundary inference and semantic enhancement. Zhang et al. (2017b) fused high-level features to lower layers for acquiring adequate content information. For integration, they used a reformulated dropout to construct an uncertain ensemble of internal feature units. Zhang et al. (2018) designed symmetrical fully convolutional neural networks (FCNs) to learn complementary saliency features. For acceleration, Wu, Su, and Huang (2019) proposed a cascaded partial decoder framework, which discarded larger resolution features of shallower layers and directly utilized the generated saliency map to refine the features of backbone network.

However, such skip connection also passed through cluttered and noisy information. To solve this problem, some works introduced *attention* mechanism and *recurrent structure*. For example, Liu, Han, and Yang (2018) proposed a pixel-wise attention network. They generated attention maps for each pixel and selectively aggregated the contextual information to construct the attended contextual features. Liu et al. (2019) built two pooling-based modules, one module provided different layers the location information of potential salient objects, while the other merged features. Zhuge, Zeng, and Lu (2019) transformed prior information into an embedding space to select attentive features and to filter out outliers. Feng, Lu, and Ding (2019) proposed a boundary-enhanced loss for learning exquisite object boundaries. Deng et al. (2018) proposed a recurrent residual refinement network equipped with residual refinement blocks to detect salient regions. Wang et al. (2018b) constructed a global recurrent localization network and a local boundary refinement network. The former exploited contextual information of the weight map to avoid redundancy, while the latter learned the contextual message for each spatial position to recover object boundaries. Our approach also takes feature selection issues into account. The difference is that we are targeting degraded images. To this end, we first consider the completeness of features. And then, we propose a multi-type self-attention network to filter information from multiple perspectives.

Multi-Task Structure

Several studies employed multi-task strategies for rich supervision. *e.g.*, Wang et al. (2018a) employed both saliency and segmentation labels for model training. Li et al. (2018) leveraged the contextual information provided by a well-trained contour extraction network to obtain fine boundary. Li et al. (2017) also trained using both contour and saliency labels. Qin et al. (2019) proposed a hybrid loss to learn from ground truth information in pixel-, patch- and map- levels. Moreover, Zhang et al. (2019) leveraged caption as auxiliary semantic knowledge to learn discriminative semantic features for salient objects. Different from the above models, MSANet refines the salient object boundary from the perspective of feature selection, and does not use any auxiliary supervision information.

Furthermore, all the models mentioned above are sensitive to image quality – they are vulnerable to degraded images. On the contrary, our MSANet learns more fully degraded features through attention transfer training so that it can adapt to various degraded situations and get more accurate degraded saliency maps.

MSANet

We choose VGGNet16 as the backbone network and discard the last pooling layer and dense layers to maintain the spatial structure. As shown in figure.2, the core ideas of MSANet are: 1) attention translation and 2) feature selection.

For the former, we propose an attention transfer network (*ATN*) for guidance learning, which helps to percept the structural details and semantic content of degraded inputs. For the latter, we design a multi-type self-attention network (*MSA*), which implements pixel-level selection of features from multiple perspectives, so as to strengthen positive items while suppressing disturbances.

Attention Transfer Network (ATN)

When the input data is mixed with noise information (degraded), its distribution changes greatly compared with the clear one. The offset of data distribution will invalidate the feature extraction ability of the general network and cause the final detection to fail. Considering that the degraded image x_d and the clear image x_o largely share the same structural information, we design this attention transfer module(ATN) based on the "teacher-student" network to migrate the knowledge of the teacher network.

As shown in figure.3, ATN has a twin structure, which uses $\omega_t(.)$ pre-trained on x_o as the "teacher" net to guide the

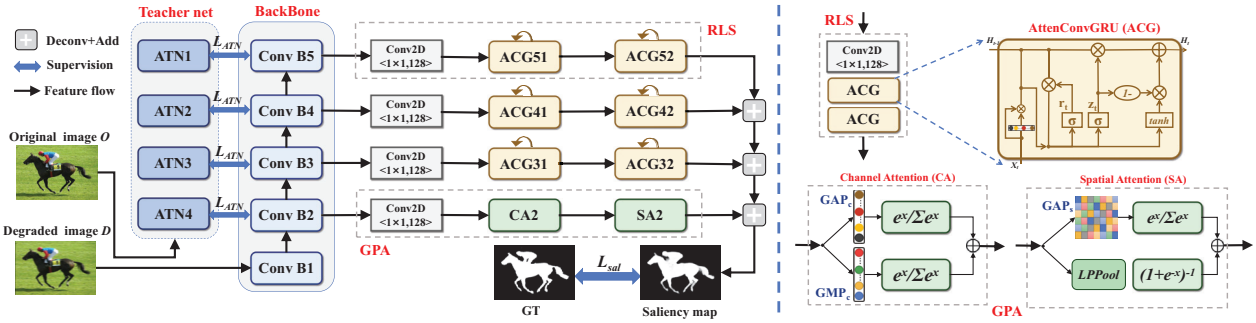

Figure 2: The overall architecture of our MSANet, consisting of Attention Transfer Network (*ATN*), Recurrent Local Self-attention module (*RLS*), and Global Pixel Self-attention module (*GPA*). Details of *RLS* and *GPA* are shown in the right side, and the structure of *ATN* is shown in figure.3.

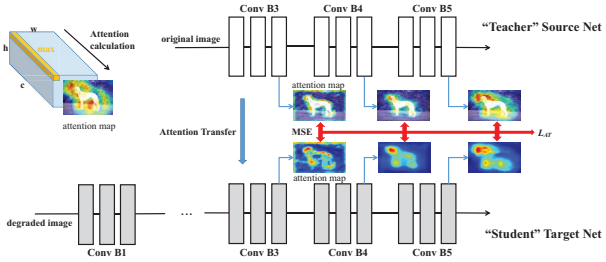

Figure 3: The architecture of attention transfer network.

learning process of the "student" net $\omega_s(.)$. Since the higher features extracted from $\omega_t(.)$ are semantic/task-related, it is reasonable to expect that the feature distribution of $\omega_t(.)$ pre-trained on clean images should be similar to the feature distribution of $\omega_s(.)$ fine-tuned using degraded images (Guo et al. 2019).

Specifically, let $\{(x_d, x_o); y\}$ represents a pair of training samples, we use $\{x_o; y\}$ to train $\omega_t(.)$, and then, we use $\{x_d; y\}$ to train $\omega_s(.)$, in this phase, we freeze $\omega_t(.)$ and use the encode features of $\omega_t(x_o)$ as the guidance information to pilot the detail perception and semantical feature learning of $\omega_t(.)$. Especially, we respectively calculate the spatial attention map M of ω_t and ω_s, and minimize the attention transfer loss ($\mathcal{L}_{AT}$). M decodes the attention of the current feature map, which is defined as $M^i = \delta(\max_{j=1,C} B_i^j)$, B_i is the feature of i-layer, δ is softmax normalization. The $\mathcal{L}_{AT}$ can be represented as:

$$\mathcal{L}^i(\omega_t; x_d, x_o) = \|M^i_{\omega_s} - M^i_{\omega_t}\|_2^2 \quad (1)$$

$$\mathcal{L}_{AT}(\omega_t; x_d, x_o) = \Sigma_{i\in\mathcal{I}}\mathcal{L}^i(\omega_t; x_d, x_o) \quad (2)$$

where M_{ω_s} and M_{ω_t} is the spatial attention map of ω_s, ω_t respectively. $\mathcal{I} = \{2, 3, 4, 5\}$.

In order to minimize the objective function $\mathcal{L}_{AT}$, we constantly update the parameters of ω_t and find a relatively better solution ω_t^* to achieve the purpose of attention migration.

$$\omega_t^* \leftarrow \text{argmin}_{\omega_t}\mathcal{L}_{AT}(\omega_t; x_d, x_o)) \quad (3)$$

During training, the parameters of ω_s is frozen and not be updated; during testing, we close ω_s and only used the results of ω_t^*.

Multi-type Self-Attention (MSA)

We calculate the multi-type self-attention score to redetermine the pixel weights. Specifically, we respectively implement recurrent local self-attention (*RLS*) estimation and global pixel self-attention (*GPA*) inference.

For the four encoding blocks B_{2-5}, we perform *RLS* on B_{3-5} and *GPA* on B_2. The main reasons are: **i.** Deploying *RLS* on lower-level features reduces computational speed and has little improvement in accuracy; **ii.** *GPA* calculates pixel-level relationship, while such correlation remains more complete in shallow layers.

Recurrent Local Self-Attention (RLS). In order to fully explore local context associations, we design an attention-based convolutional gated recurrent units, *i,e. ACG*. Similar to *ConvLSTM*, the proposed *ACG* also employ gates for information selection and integration (figure.2). Specifically, *ACG* consists of two convolutional gates (*reset, update*) and one hidden state (*h*). Unlike *ConvLSTM*, we weight the input information on each iteration by an attention layer, which enables more accurate local optimization of features while reducing the number of parameters.

As shown in figure.4, RLS consists of three layers: a convolution layer and two ACG layers. Convolution for channel scaling (128) and ACG for feature optimization.

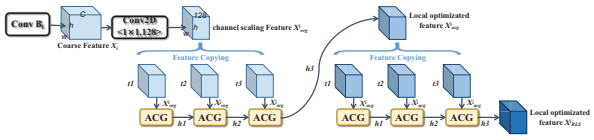

Figure 4: Details of RLS structure.

More specifically, ACG requires 4-D input data x_{acg}(t, w, h, c). Let the scaled feature is x_{org}, x_{acg} is a set of copies of x_{org}, and the number of copies is determined by t, that means, at each time step t, ACG accepts two sets of inputs, the old hidden state data h_{t-1}, and the new input data x_t.

Here, x_t is constant, that is, it is always x_{org}, and the hidden state h_t is continuously updated, which selectively fuses information from x_t and h_{t-1} through the gating structure, thus to locally optimize x_{org}. After a t times optimization, the final h_t is output as the optimized feature x_{RLS}.

Given the input feature x_t (x_{org}) at time step t, the status update process is driven by the following formula:

$$\bar{x_t} = x_t . \star (I + e^{gap(x_t)} / \Sigma e^{gap(x_i')}) \quad (4)$$

$$r_t = \sigma(w_{xr} * \bar{x_t} + w_{hr} * h_{t-1} + b_r) \quad (5)$$

$$\bar{h}_t = tanh(r_t . \star w_{h\bar{h}} * h_{t-1} + w_{x\bar{h}} * \bar{x_t} + b_{\bar{h}}) \quad (6)$$

$$z_t = \sigma(w_{xz} * \tilde{x_t} + w_{hz} * h_{t-1} + b_z) \quad (7)$$

$$h_t = (1 - z_t) * \bar{h}_t + z_t * h_{t-1} \quad (8)$$

where I is a matrix with all values of 1, $*$ is the convolution operation, $.\star$ is pixel-wise multiplication, X_{org} represents the weighted input. We discuss the setting of t in table.1.

	t=2	*t=3*	*t=4*	*t=5*	*t=6*
F_β	0.8925	0.9008	0.9005	0.9015	0.9010
MAE	0.0711	0.0497	0.0517	0.0561	0.0578

Table 1: Performance evaluation results of ACG on ECSSD under different step settings. Considering the computational cost, we finally choose *t*=3 to achieve the best balance.

We define RLS as a local mode because there is no explicit global information-based attention calculating process in the feature optimization process. The maintaining or suppressing of each feature point is implicitly completed. The visualized features in figure.5 show that the deployment of the RLS effectively captures the boundary information and keeps the attention on the foreground area.

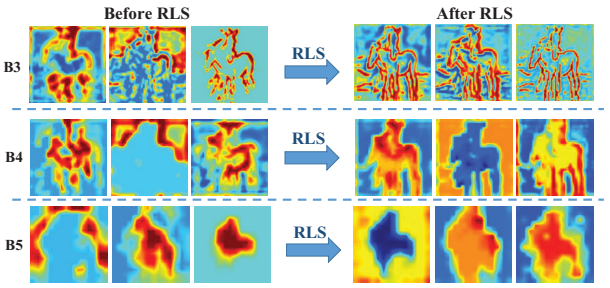

Figure 5: Visualization of B3-5 features of some channels before and after RLS implementation.

Global Pixel Self-Attention (GPA). GPA explicitly models channel attention (CA) and spatial attention (SA), the score for each position is obtained by comparison with all other positions, so GPA a global mode.

For feature $B_i^{w \times h \times c}$, we firstly perform CA. Since each channel focus on different feature, it is necessary to highlight those channels that focus on the foreground objects. We calculate max and mean values simultaneously to obtain soft attention:

$$c(B_i) = \delta^1(AP_c(B_i)) + \delta^2(MP_c(B_i)) \quad (9)$$

δ is a softmax normalization. When dealing with shallow features, each response can be thought as a detector for boundaries. Considering that δ^2 returns only a single response, which focus on one discriminative part and ignore others, while δ^1 encourages the detector to treat all locations on average, inevitably introducing noise. To this end, we calculate $c(B_i)$ to do a soft selection.

Then we calculate SA, which includes two items. The first item is similar to CA, which computes the spatial mean matrix and employs softmax for normalization. The second item *LPPool* solves the local patch similarity. We scale the channels to 1 by dot-convolution and use average pooling (2*2) to get the representative value of each patch, thus to ensure that the attention score of each pixel is calculated both locally and globally.

$$s(B_i) = \delta(AP_s(B_i)) + \sigma(LPPool(B_i)) \quad (10)$$

Where σ is a sigmoid function. It should be noted that we use softmax in spatial average pooling and use sigmoid in local-patch attention calculation, because the response of single position should be independent (sum to 1) while patch-wise response is related with others.

Finally, the global weighted feature $\bar{B}_i$ is calculated by:

$$\bar{B}_i = B_i . \star (I + c(B_i)) . \star (I + s(B_i)) \quad (11)$$

where $c(B_i)$ and $s(B_i)$ is the channel and spatial attention score respectively. I is a matrix with all values of 1. We first employ CA and then employ SA.

Joint Loss

Our MSANet calculates multiple losses. Let S_i represents the prediction, G_i is the ground-truth. The total loss L_{total} includes: **i.** Binary Cross-entropy loss $\mathcal{L}_{bce}$. **ii.** Structural Similarity loss $\mathcal{L}_{ssim}$, which helps to learn the structural information of G_i. **iii.** Attention Transfer loss $\mathcal{L}_{AT}$, which promotes the model to obtain more powerful feature representations. L_{total} is expressed as:

$$\mathcal{L}_{total} = \lambda_1 \mathcal{L}_{bce} + \lambda_2 \mathcal{L}_{ssim} + \lambda_3 \mathcal{L}_{AT} \quad (12)$$

where $\mathcal{L}_{bce} = -\Sigma_i G_i \log S_i + (1 - G_i)\log(1 - S_i)$, $\mathcal{L}_{ssim} = 1/N(1 - SSIM(G_i, S_i))$ and $SSIM()$ (Wang, Simoncelli, and Bovik 2003) is the standard structural similarity function. λ_1, λ_2 and λ_3 is set to 5, 10 and 2 respectively.

Experiment

Implementation Details

The whole architecture is built on the keras deep learning framework. We employ **DUTS-TR** (Wang et al. 2017) to train, which contains 10,553 images, and we perform image enhancement, i.e., *reverse*, *mirroring* to increase to 65k. All the experiments are run on NVIDIA Geforce GTX 1080Ti (11 GB memory) and i7-8700k cpu. The first five conv-blocks of MSANet are all initialized from VGGNet16, and the other parameters are all randomly assigned. All training and testing images are resized to 224×224. We use the Adam optimizer to train and the learning rate is initialized to 1e-5.

Datasets

Seven large benchmarks are used for evaluation, including **DUT-OMRON** (Yang et al. 2013), **ECSSD** (Shi et al. 2016), **HKU-IS** (Li and Yu 2016), **PASCAL-S** (Li et al. 2014), **SOD** (Movahedi and Elder 2010), **MSRA-B** (Jiang et al. 2013), and **DUTS-TE** (Wang et al. 2017).

For realistic degraded simulation, we perform multi-type and multi-scale degradation on clear images in table.2. During training, the synthesis process guarantees two randomnesses: random type and random scale, and for test, we representatively reported the experimental results in one fixed degraded condition under each effect.

	LR	GB	MB	GN	HZ
Scale factor	r	h	l	v	ϵ
Scale range	3	5	5	0.01	0.05
	4	7	7	0.03	0.10
	5	9	9	0.05	0.15

Table 2: Details of the synthetic dataset. LR: low resolution, r represents the resize scale; GB: gaussian blur, h is the fuzzy kernel size; MB: motion blur, l is the motion length; GN: Gaussian white noise with a mean of m=0 and a variance of v; HZ: haze and ϵ is the degree of haze.

Evaluation Metrics

Five evaluation criteria, including P-R curves, F-measure, mean absolute error (Borji et al. 2015), S-measure (Fan et al. 2017), and E-measure (Fan et al. 2018) are used to reflect the model performance.

Analysis

We compare our model with other 13 state-of-the-art CNN-based models, including **AFNet** (Feng, Lu, and Ding 2019), **BASNet** (Qin et al. 2019), **CPDNet** (Wu, Su, and Huang 2019), **MWSNet** (Zeng et al. 2019), **C2SNet** (Li et al. 2018), **PiCANet** (Liu, Han, and Yang 2018), **DGRL** (Wang et al. 2018b), **LFRNet** (Zhang et al. 2018), **RFCN** (Wang et al. 2018a), **Amulet** (Zhang et al. 2017a), **UCF** (Zhang et al. 2017b), **WSS** (Wang et al. 2017), and **MSRNet** (Li et al. 2017). To be fair, we use their released code with default settings to calculate the degraded saliency maps.

To fully illustrate the effect of proposed ATN in dealing with degraded images, we choose Amulet (Zhang et al. 2017a) for adversarial training, that is, based on the already trained model, we further use the hybrid dataset (mixed by degraded and clear images) for finetuning, and get adv-Amulet. The training process is in full compliance with the description in their paper to ensure fairness.

Quantitative Evaluation. The evaluated results are illustrated in tables.3-7, the best result is highlighted in **bold**. Since LFRNet, RFCN, UCF and Amulet were trained on MSRA10K and MSRA-B is a subset of MSRA10K, so we did not report their results on MSRA-B. The SE scores and PR curves are illustrated in figures.6 and 8.

Methods	*Original*		*Motion Blur*		*Gaussian Noise*	
	F_β	MAE	F_β	MAE	F_β	MAE
Ours	**0.9134**	0.0414	**0.9021**	0.0472	**0.8947**	**0.0501**
AFNet	0.8936	0.0427	0.8718	0.0526	0.8782	0.0517
BASNet	0.8993	**0.0356**	0.8829	**0.0418**	0.8693	0.0519
C2SNet	0.8663	0.0528	0.8664	0.0537	0.8660	0.0551
CPDNet	0.8956	0.0404	0.8671	0.0525	0.8232	0.0790
MWSNet	0.8666	0.0712	0.8655	0.0753	0.8597	0.0836
PiCANet	0.8670	0.0564	0.8406	0.0734	0.8290	0.0802
DGRL	0.8852	0.0434	0.8712	0.0470	0.8235	0.0635
MSRNet	0.8873	0.0422	0.8675	0.0505	0.8639	0.0527
WSS	0.8535	0.0761	0.8401	0.0846	0.8399	0.0813

Table 3: Comparison on MSRA-B. MB:l=7, GN:ν=0.10.

Methods	SOD					
	Original		*Motion Blur*		*Gaussain Noise*	
	F_β	MAE	F_β	MAE	F_β	MAE
Ours	**0.8043**	0.1132	**0.7738**	**0.1264**	**0.7368**	**0.1450**
adv-Amulet	0.7384	0.1517	0.7182	0.1618	0.6878	0.1732
Amulet	0.7518	0.1408	0.7166	0.1607	0.6916	0.1710
AFNet	0.7951	**0.1082**	0.7226	0.1415	0.7354	0.1454
BASNet	0.7474	0.1124	0.6927	0.1388	0.6468	0.1645
C2SNet	0.7596	0.1258	0.7330	0.1392	0.7165	0.1550
CPDNet	0.8018	0.1125	0.7130	0.1488	0.6260	0.1923
MWSNet	0.7339	0.1661	0.7146	0.1802	0.6892	0.1935
PiCANet	0.7265	0.1299	0.6641	0.1633	0.6660	0.1844
DGRL	0.7972	0.1128	0.7334	0.1255	0.6102	0.1659
LFRNet	0.7771	0.1243	0.7236	0.1524	0.7095	0.1658
MSRNet	0.7716	0.1118	0.7289	0.1252	0.7212	0.1548
RFCN	0.7664	0.1441	0.7183	0.1638	0.6726	0.1831
UCF	0.7500	0.1520	0.7184	0.1697	0.6799	0.1828
WSS	0.7271	0.1688	0.7011	0.1782	0.6807	0.1872

Table 4: Comparison on SOD. MB:l=7, GN:ν=0.10

According to the results shown in tables.3-7, the performance of most existing algorithms drops dramatically when dealing with degraded images. In particular, CPDNet (Wu, Su, and Huang 2019) and LFRNet (Zhang et al. 2018), which performed well on clear images, show a significant decrease in performance on degraded images. Especially in low-resolution scenes, whose performance has dropped by 37.5%, 25.0%, respectively. What's more, the overall best performing MSRNet (Li et al. 2017) and C2SNet (Li et al. 2018) can basically maintain a stable level when processing low-resolution images, but the performance decreases drastically when handling blurred and noisy images. This is because both two models leverage contour information. Noisy and motion blur struck the integrity of the contour, the failure of edge detection further leads to the failure of saliency detection. On the contrary, our MSANet perform stable in various degraded scenarios, whose overall performance ranks first. This proves that our model is more adaptable and can effectively deal with various complex scenarios. A clearer performance trend graph is shown in figure.6. The red five-pointed star represents our model. Whether it is dealing with low resolution, blur or noise scenes, MSANet has a smaller slope and less interference from negative information, while other models have experienced a drastic decline in performance.

In addition, adv-Amulet performs better in degraded scenarios than other models, indicating that adversarial training can improve model performance in degraded scenarios to a certain extent. However, such adversarial training has a

Methods	Pub.	FPS	DUT-OMRON						ECSSD					
			Original		*Low Resolution*		*Gaussain Blur*		*Original*		*Low Resolution*		*Gaussain Blur*	
			F_β	MAE	F_β	MAE	F_β	MAE	F_β	MAE	F_β	MAE	F_β	MAE
Ours	-	40	0.7470	**0.0562**	**0.6314**	**0.0896**	**0.7051**	**0.0685**	0.9074	0.0482	**0.8283**	**0.0808**	**0.8816**	**0.0599**
adv-Amulet	-	21	0.7119	0.0762	0.5722	0.1151	0.6514	0.0933	0.8897	0.0639	0.7817	0.1044	0.8572	0.0793
Amulet	ICCV2017	21	0.6990	0.0719	0.4835	0.1837	0.5103	0.1665	0.8690	0.8591	0.7209	0.1283	0.8021	0.0907
AFNet	CVPR2019	25	0.7342	0.0575	0.4807	0.1034	0.5264	0.0931	0.9017	0.0417	0.7138	0.1238	0.8103	0.0912
BASNet	CVPR2019	5	**0.7568**	0.0567	0.3201	0.1165	0.5988	0.0772	0.8796	0.0370	0.5096	0.1592	0.7758	0.0864
C2SNet	ECCV2018	19	0.6644	0.0787	0.5907	0.1062	0.6459	0.0831	0.8572	0.0574	0.8068	0.0821	0.8541	0.0692
CPDNet	CVPR2019	66	0.7400	0.0571	0.4241	0.1274	0.5431	0.1017	**0.9145**	**0.0402**	0.5964	0.1742	0.7950	0.1022
MWSNet	CVPR2019	3	0.6061	0.1097	0.5694	0.1215	0.6128	0.1027	0.8398	0.0965	0.7855	0.1193	0.8190	0.1120
PiCANet	CVPR2018	7	0.6887	0.0674	0.3228	0.1335	0.4448	0.1226	0.8481	0.0585	0.5624	0.1767	0.7347	0.1275
DGRL	CVPR2018	6	0.7289	0.0615	0.4577	0.1012	0.5967	0.0771	0.9059	0.0407	0.6969	0.1087	0.8449	0.0584
LFRNet	IJCAI2018	15	0.6388	0.1160	0.4574	0.2544	0.5045	0.2234	0.8766	0.0545	0.6906	0.1568	0.7935	0.0940
MSRNet	CVPR2017	13	0.6878	0.0700	0.4937	0.1431	0.6518	0.0791	0.8610	0.0564	0.6874	0.1352	0.8367	0.0701
RFCN	PAMI2019	7	0.6726	0.0781	0.5217	0.1141	0.5762	0.1004	0.8708	0.0672	0.7025	0.1389	0.7747	0.1145
UCF	ICCV2017	23	0.6767	0.0924	0.5465	0.1414	0.5837	0.1290	0.8936	0.0591	0.7676	0.1113	0.8296	0.0851
WSS	CVPR2017	29	0.5978	0.1118	0.5026	0.1365	0.5541	0.1262	0.8233	0.1035	0.7211	0.1591	0.7858	0.1289

Table 5: Comparison on DUT-OMRON and ECCSD. LR: r=4, GB: h=7.

Methods	Size(MB)	BackBone	HKU-IS						PASCAL-S					
			Original		*Low Resolution*		*Gaussain Blur*		*Original*		*Motion Blur*		*Gaussain Noise*	
			F_β	MAE	F_β	MAE	F_β	MAE	F_β	MAE	F_β	MAE	F_β	MAE
Ours	84.7	$VGGNet_{16}$	0.8962	0.0427	**0.7993**	**0.0727**	**0.8686**	**0.0523**	**0.8208**	0.0736	**0.8006**	**0.0829**	**0.7716**	**0.0982**
adv-Amulet	132.6	$VGGNet_{16}$	0.8756	0.0547	0.7636	0.0900	0.8464	0.0651	0.7954	0.0956	0.7512	0.1033	0.7372	0.1137
Amulet	132.6	$VGGNet_{16}$	0.8690	0.0591	0.6880	0.1171	0.7800	0.0775	0.7574	0.0997	0.7307	0.1129	0.7075	0.1321
AFNet	143.9	$VGGNet_{16}$	0.8886	0.0358	0.6764	0.1027	0.7948	0.0709	0.8149	0.0720	0.7817	0.0865	0.7709	0.0986
BASNet	348.5	$ResNet_{34}$	0.8955	0.0322	0.4970	0.1346	0.7845	0.0700	0.7733	0.0765	0.7401	0.0924	0.6971	0.1234
C2SNet	635.4	$VGGNet_{16}$	0.8427	0.0496	0.7784	0.0750	0.8302	0.0595	0.7575	0.0850	0.7649	0.0877	0.7422	0.1112
CPDNet	192	$VGGNet_{16}$	0.9003	0.0320	0.5850	0.1393	0.7888	0.0757	0.8199	**0.0721**	0.7771	0.0940	0.7131	0.1326
MWSNet	602.4	$DenseNet_{168}$	0.8141	0.0843	0.7523	0.1072	0.7932	0.0985	0.7125	0.1330	0.7081	0.1387	0.6940	0.1534
PiCANet	188.9	$VGGNet_{16}$	0.8502	0.0511	0.5459	0.1505	0.7251	0.1078	0.7509	0.0850	0.7312	0.1072	0.6995	0.1310
DGRL	648	$ResNet_{50}$	0.8905	0.0355	0.6590	0.0964	0.8158	0.0578	0.8144	0.0723	0.7927	0.0831	0.7052	0.1182
LFRNet	173.6	$VGGNet_{16}$	**0.9113**	**0.0263**	0.6720	0.1433	0.7982	0.0735	0.7553	0.1079	0.7234	0.1252	0.7003	0.1406
MSRNet	331.8	$VGGNet_{16}$	0.8662	0.0393	0.6535	0.1277	0.8359	0.0529	0.7673	0.0830	0.7511	0.0912	0.7367	0.1024
RFCN	1126.4	$VGGNet_{16}$	0.8557	0.0549	0.6800	0.1167	0.7567	0.0920	0.7684	0.1035	0.7375	0.1188	0.6870	0.1443
UCF	117.9	$VGGNet_{16}$	0.8651	0.0513	0.7291	0.1036	0.7993	0.0764	0.7824	0.0961	0.7514	0.1101	0.7210	0.1327
WSS	58.9	$VGGNet_{16}$	0.8237	0.0787	0.7138	0.1320	0.7842	0.1026	0.7151	0.1392	0.6996	0.1462	0.6883	0.1495

Table 6: Comparison on HKU-IS and PASCAL-S. LR: r=4, GB: h=7. MB: l=7, GN: ν=0.10.

fixed format depending on the type of training data. Excessive degraded data affects the performance of the model on clear images, and deficient degraded data limits the ability in dealing with various degraded scenes, coupled with the lack of explicit modeling of implicit features, so there is still an obvious gap between adv-Amulet and our MSANet, which proves the effectiveness of our ATN.

Methods	DUTS-TE					
	Original		*Haze ϵ=0.10*		*Haze ϵ=0.15*	
	F_β	MAE	F_β	MAE	F_β	MAE
Ours	0.7762	0.0517	0.7474	0.0643	**0.7084**	**0.0776**
adv-Amulet	0.7497	0.0701	0.6928	0.0847	0.6403	0.0986
Amulet	0.7080	0.0853	0.5490	0.1840	0.4918	0.2263
AFNet	0.7924	0.0458	0.7487	0.0616	0.6510	0.0838
BASNet	0.7911	0.0476	0.7319	0.0630	0.6588	0.0784
C2SNet	0.7120	0.0656	0.7069	0.0713	0.6692	0.0830
CPDNet	**0.8132**	**0.0429**	**0.7589**	**0.0619**	0.6705	0.0854
MWSNet	0.6478	0.1020	0.5915	0.1184	0.5029	0.1376
PiCANet	0.7359	0.0532	0.6544	0.0865	0.5223	0.1144
DGRL	0.7938	0.0497	0.7285	0.0658	0.6277	0.0849
LFRNet	0.6950	0.0845	0.6097	0.1193	0.5293	0.1513
MSRNet	0.7120	0.0656	0.6668	0.0780	0.6692	0.0829
RFCN	0.7087	0.0745	0.6612	0.0903	0.5903	0.1078
UCF	0.7212	0.0800	0.6608	0.1030	0.6130	0.1201
WSS	0.6531	0.1001	0.5880	0.1154	0.4968	0.1316

Table 7: Comparison on DUTS-TE. ϵ is the degree of haze.

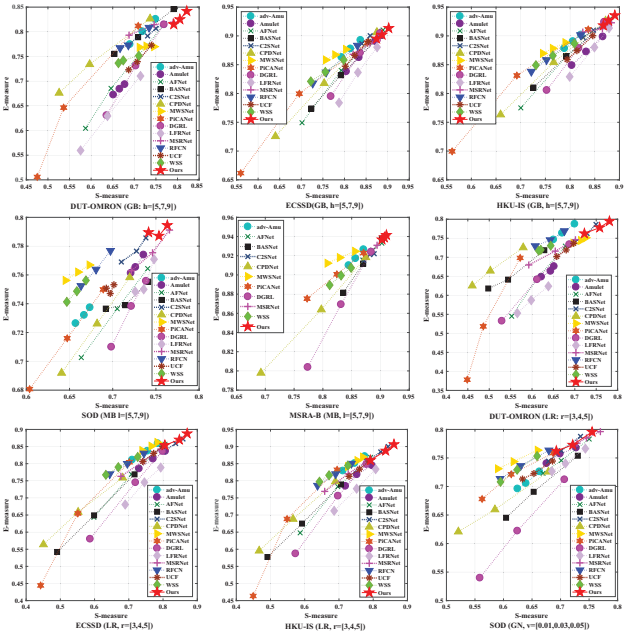

Figure 6: Performance trends of 15 models in different situations. Our model has the best stability.

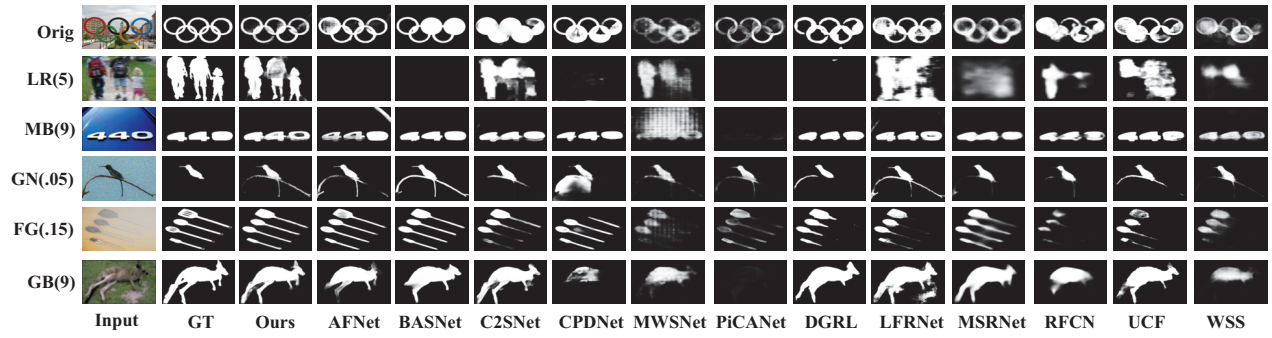

Figure 7: Visualization of our MSANet and 12 competitors in some representative scenarios. LR(r): reduct resolution ($\times 1/r$), MB(h): motion blur (length h), GN(ν): Gaussian white noise (covariance ν), FG(ϵ): haze (degree ϵ).

Qualitative Evaluation. Figure.7 provides some visualization of MSANet and other 12 methods in some challenging scenarios, where our MSANet is superior to other methods. For example, in low-resolution scenes, most of the models lose the position of salient objects (2nd). In blurring, foggy, and noisy scenes, many methods cannot accurately detect salient objects due to the side effects of interference information, or incorrectly regard the background as a salient object (3rd-6th), yet our proposed model is able to capture complete salient regions in a variety of complex scenarios.

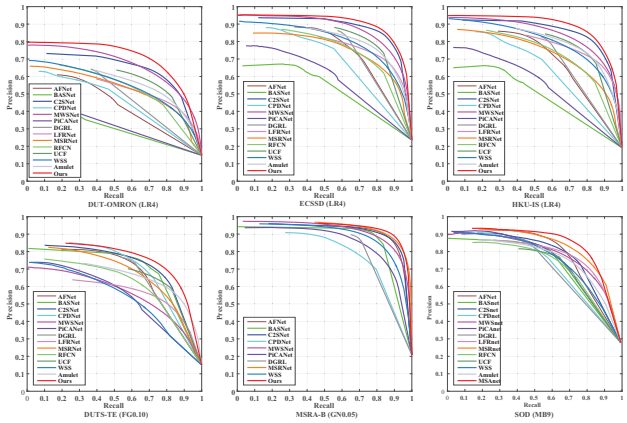

Figure 8: The precision-recall curves of 15 methods. The red curve represents our proposed MSANet. The number in parentheses represents the degradation scale, as detailed in Table.2.

Ablation Study. The results in rows 6-7 of table.8 show that the effect of introducing ATN is better than that of adversarial training, which indicates that ATN can effectively promote the understanding of the content of degraded images and help to obtain more abundant features. Meanwhile, the results of lines 5-6 also show that our proposed MSA structure is better than Amulet under the same adversarial training mode. It is worth noting how to calculate and where to teach has a great impact on performance. According to the results in lines 1-4, after the introduction of RLS, the performance has increased by 2.4%-4.3%, which indicates the superiority of RLS in suppressing the interference of non-target regions. And the introduction of GPA further improves performance slightly, which indicates that GPA is helpful for optimizing the details of salient objects.

	S_λ	E_m	F_β	MAE
†baseline En-De-V	.869	.901	.826	.078
†En-De-V+RLS	.893	.929	.869	.044
†En-De-V+GPA	.882	.920	.845	.049
†En-De-V+RLS+GPA	**.894**	**.942**	**.896**	**.043**
adv-Amulet	.746	.851	.764	.090
‡adv-En-De-V+RLS+GPA	.826	.880	.778	.081
‡En-De-V+RLS+GPA+ATN	**.833**	**.891**	**.795**	**.074**

Table 8: Ablation study on HKU-IS under different architectures. †: clear image, ‡: degraded image with LR scale r=4.

Conclusion

In this paper, we systematically studied the degraded SOD and proposed a multi-type self-attention network, MSANet. Through attention transfer learning, feature extractor can learn more abundant semantic details of degraded images and perceive their hidden feature patterns. Meanwhile, MSANet performs global and local attention inference on multi-scale features for information filtering. Extensive experiments were conducted to verify the effectiveness of our model. In the future work, we plan to further extend this work to detect video saliency under motion blurs, as well as SOD for small objects.

Acknowledgments

This work was supported by the Natural Science Foundation of China under Grant 61572351, 61876125, 61772360, 61672376 and U1803264.

References

Avytekin, C.; Cricri, F.; and Aksu, E. 2018. Saliency enhanced robust visual tracking. In *2018 7th European Workshop on Visual Information Processing (EUVIP)*, 1–5. IEEE.

Borji, A.; Cheng, M.-M.; Jiang, H.; and Li, J. 2015. Salient object detection: A benchmark. *IEEE transactions on image processing* 24(12):5706–5722.

Deng, Z.; Hu, X.; Zhu, L.; Xu, X.; Qin, J.; Han, G.; and Heng, P.-A. 2018. R3net: Recurrent residual refinement network for saliency detection. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 684–690. AAAI Press.

Fan, D.-P.; Cheng, M.-M.; Liu, Y.; Li, T.; and Borji, A. 2017. Structure-measure: A new way to evaluate foreground maps. In *Proceedings of the IEEE International Conference on Computer Vision*, 4558–4567.

Fan, D.-P.; Gong, C.; Cao, Y.; Ren, B.; Cheng, M.-M.; and Borji, A. 2018. Enhanced-alignment measure for binary foreground map evaluation. *arXiv preprint arXiv:1805.10421*.

Feng, M.; Lu, H.; and Ding, E. 2019. Attentive feedback network for boundary-aware salient object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1623–1632.

Guo, D.; Pei, Y.; Zheng, K.; Yu, H.; Lu, Y.; and Wang, S. 2019. Degraded image semantic segmentation with dense-gram networks. *IEEE transactions on image processing: a publication of the IEEE Signal Processing Society*.

Harel, J.; Koch, C.; and Perona, P. 2007. Graph-based visual saliency. In *Advances in neural information processing systems*, 545–552.

Jiang, H.; Wang, J.; Yuan, Z.; Wu, Y.; Zheng, N.; and Li, S. 2013. Salient object detection: A discriminative regional feature integration approach. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2083–2090.

Li, G., and Yu, Y. 2016. Visual saliency detection based on multiscale deep cnn features. *IEEE Transactions on Image Processing* 25(11):5012–5024.

Li, Y.; Hou, X.; Koch, C.; Rehg, J. M.; and Yuille, A. L. 2014. The secrets of salient object segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 280–287.

Li, G.; Xie, Y.; Lin, L.; and Yu, Y. 2017. Instance-level salient object segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, 247–256. IEEE.

Li, X.; Yang, F.; Cheng, H.; Liu, W.; and Shen, D. 2018. Contour knowledge transfer for salient object detection. In *European Conference on Computer Vision*, 370–385. Springer.

Liu, J.-J.; Hou, Q.; Cheng, M.-M.; Feng, J.; and Jiang, J. 2019. A simple pooling-based design for real-time salient object detection. *arXiv preprint arXiv:1904.09569*.

Liu, N.; Han, J.; and Yang, M.-H. 2018. Picanet: Learning pixel-wise contextual attention for saliency detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3089–3098.

Movahedi, V., and Elder, J. H. 2010. Design and perceptual validation of performance measures for salient object segmentation. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, 49–56. IEEE.

Qin, X.; Zhang, Z.; Huang, C.; Gao, C.; Dehghan, M.; and Jagersand, M. 2019. Basnet: Boundary-aware salient object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 7479–7489.

Shi, J.; Yan, Q.; Xu, L.; and Jia, J. 2016. Hierarchical image saliency detection on extended cssd. *IEEE transactions on pattern analysis and machine intelligence* 38(4):717–729.

Wang, L.; Lu, H.; Wang, Y.; Feng, M.; Wang, D.; Yin, B.; and Ruan, X. 2017. Learning to detect salient objects with image-level supervision. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.(CVPR)*, 136–145.

Wang, L.; Wang, L.; Lu, H.; Zhang, P.; and Ruan, X. 2018a. Salient object detection with recurrent fully convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Wang, T.; Zhang, L.; Wang, S.; Lu, H.; Yang, G.; Ruan, X.; and Borji, A. 2018b. Detect globally, refine locally: A novel approach to saliency detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3127–3135.

Wang, Z.; Simoncelli, E. P.; and Bovik, A. C. 2003. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, volume 2, 1398–1402. Ieee.

Wu, Z.; Su, L.; and Huang, Q. 2019. Cascaded partial decoder for fast and accurate salient object detection. *arXiv preprint arXiv:1904.08739*.

Yang, C.; Zhang, L.; Lu, H.; Ruan, X.; and Yang, M.-H. 2013. Saliency detection via graph-based manifold ranking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3166–3173.

Zeng, Y.; Zhuge, Y.; Lu, H.; Zhang, L.; Qian, M.; and Yu, Y. 2019. Multi-source weak supervision for saliency detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6074–6083.

Zhang, P.; Wang, D.; Lu, H.; Wang, H.; and Ruan, X. 2017a. Amulet: Aggregating multi-level convolutional features for salient object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, 202–211.

Zhang, P.; Wang, D.; Lu, H.; Wang, H.; and Yin, B. 2017b. Learning uncertain convolutional features for accurate saliency detection. In *Proceedings of the IEEE International Conference on Computer Vision*, 212–221.

Zhang, P.; Liu, W.; Lu, H.; and Shen, C. 2018. Salient object detection by lossless feature reflection. *arXiv preprint arXiv:1802.06527*.

Zhang, L.; Zhang, J.; Lin, Z.; Lu, H.; and He, Y. 2019. Capsal: Leveraging captioning to boost semantics for salient object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6024–6033.

Zhao, R.; Oyang, W.; and Wang, X. 2016. Person re-identification by saliency learning. *IEEE transactions on pattern analysis and machine intelligence* 39(2):356–370.

Zhuge, Y.; Zeng, Y.; and Lu, H. 2019. Deep embedding features for salient object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 9340–9347.