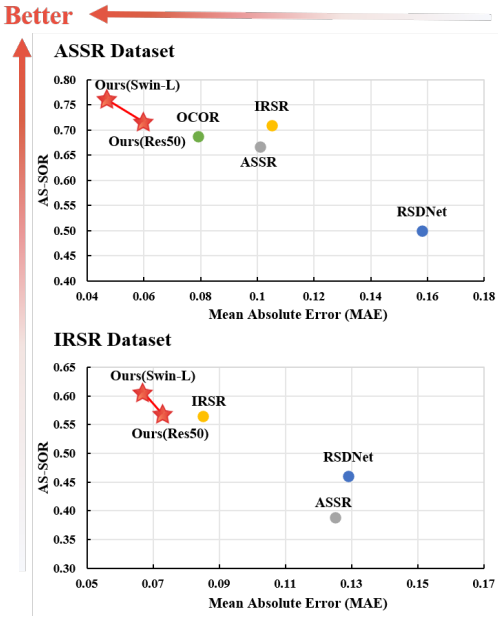
Domain Separation Graph Neural Networks for Saliency Object Ranking

Zijian Wu[1] Jun Lu[1] Jing Han[1] Lianfa Bai[1] Yi Zhang[1] Zhuang Zhao[1,†] Siyang Song[2,†]
[1]Nanjing University of Science and Technology [2]University of Leicester
{wuzijian, lujunchenhao, eohj, blf, zhy441}@njust.edu.cn
zhaozhuang3126@gmail.com, ss1535@leicester.ac.uk



(a) SA-SOR vs. MAE on ASSR and IRSR dataset.

(b) Results on the ASSR dataset of our methods with existing saliency ranking approaches.

Figure 1. Comparison with other state-of-the-art methods. (a) The comparison of SA-SOR and MAE on ASSR and IRSR dataset. The closer to the upper-left corner, the more accurate the saliency ranking and segmentation results. (b) A visual comparison of our method with other approaches illustrates that it is challenging for other methods to produce accurate segmentation and ranking results in scenarios with multiple saliency objects.

Abstract

Saliency object ranking (SOR) has attracted significant attention recently. Previous methods usually failed to explicitly explore the saliency degree-related relationships between objects. In this paper, we propose a novel Domain Separation Graph Neural Network (DSGNN), which starts with separately extracting the shape and texture cues from each object, and builds an shape graph as well as a texture graph for all objects in the given image. Then, we propose a Shape-Texture Graph Domain Separation (STGDS) module to separate the task-relevant and irrelevant information of target objects by explicitly modelling the relationship between each pair of objects in terms of their shapes and textures, respectively. Furthermore, a Cross Image Graph Domain Separation (CIGDS) module is introduced to explore the saliency degree subspace that is robust to different scenes, aiming to create a unified representation for targets with the same saliency levels in different images. Importantly, our DSGNN automatically learns a multi-dimensional feature to represent each graph edge, allowing complex, diverse and ranking-related relationships to be modelled. Experimental results show that our DSGNN achieved the new state-of-the-art performance on both ASSR and IRSR datasets, with large improvements of 5.2%

† Corresponding author.

and 4.1% SA-SOR, respectively. Our code is provided in https://github.com/Wu-ZJ/DSGNN.

1. Introduction

Saliency object ranking (SOR) aims to simulate the human visual attention system that jointly ranks the saliency degrees of multiple human-defined saliency objects in an image (i.e., their relative importance [31, 45, 49]). While SOR techniques can benefit various real-world applications, including image captioning [57], non-photorealistic rendering [18], human–robot interaction [48], etc, it is a challenging task as human attention is always influenced by various factors such as the spatial locations, sizes, color brightness, contrasts and clarity of the objects as well as their contexts in the image [45].

Current SOR methods [13, 45] usually start with detecting multiple saliency objects using existing instance segmentation methods (e.g., Mask-RCNN [20] and Mask2former [8]), based on which saliency degree-related features are specifically extracted to predict their saliency scores, where CNNs [20, 28], and Transformers [8, 50] have been frequently employed. While these methods have already demonstrated significant effectiveness in modeling relationships between each object and the scene through various spatial attention mechanisms, they frequently ignore the relationships between candidate saliency objects, which are also crucial for human observers to determine their relative importance within a scene [9, 36, 37]. Consequently, several approaches attempted to explore the relationships between saliency objects via either graph edges [31] or attention maps between objects [49].

Although the aforementioned approaches can partially model the relationship among objects as well as their relationships with the scene, they still failed to comprehensively explore the underlying saliency degree-related relationships between objects. Firstly, both shape and texture play crucial roles in determining objects' saliency degrees. However, none of existing methods [13, 31, 45, 49] considered to specifically model saliency degree-related relationship cues between objects' shapes as well as their textures, respectively, while suppressing unrelated noises contained in them. Secondly, since objects of the same saliency degree in different images may exhibit similar distributions in the task space [38], their relationships could also contribute to saliency degree estimation, which also has been ignored by existing approaches. Thirdly, existing graph-based method [31] describes the relationship between each pair of objects through a single-value graph edge, which failed to describe complex and diverse underlying relationships between them. Although attention-based method [49] could model more comprehensive relationships, they could introduce noises unrelated to the task.

In this paper, we propose a novel **Domain Separation Graph Neural Network (DSGNN)** for the saliency object ranking (SOR) task, which can specifically address the aforementioned three problems. Our DSGNN starts with separately extracting the shape and texture cues from each object, and builds an shape graph as well as a texture graph for all objects in the given image. Then, a novel **Shape-Texture Graph Domain Separation (STGDS)** module separates the task-relevant and irrelevant information in targets by explicitly modelling the relationship between shapes of all objects and texture of all objects via their graph edges, respectively. After that, a **Cross Image Graph Domain Separation (CIGDS)** module is introduced to seek a unified representation for targets with the same saliency level across different images via exploring a saliency degree subspace that is robust to different scenes. Importantly, our DSGNN automatically learns a multi-dimensional feature [47] to represent each graph edge, allowing complex, diverse and ranking-related relationships between objects' shape/texture as well as objects in different images to be comprehensively captured. In summary, the main contributions and novelties of this paper are summarized as follows:

- We propose a novel DSGNN for the SOR task, which specifically considering three key factors: (1) shape interactions of the objects in a scene; (2) texture interactions of the objects in a scene; and (3) intrinsic relationships among objects across different scenes. It is inspired by [3] but can explicitly decouple task-relevant and irrelevant cues while inferring the complex relationships among targets within and across scenes.
- We propose the first multi-dimensional edge GNN in the field of saliency object analysis, which can comprehensively and effectively model the relationship (i.e., saliency degrees) among target objects via a deep-learned multi-dimensional graph edge feature.
- The experimental results show that the proposed STGDS and CIGDS-based multi-level object relationship modelling as well as the introduced multi-dimensional edge feature learning strategy contribute complementary and crucial cues for the SOR task, making our DSGNN becoming the new state-of-the-art SOR approach, with 5.2% SA-SOR and 3.2% MAE improvements on ASSR dataset [45] while 4.1% SA-SOR and 1.8% MAE improvements on IRSR dataset [31] over the previous SOTA methods [31, 49].

2. Related Work

Saliency Object Detection: Saliency Object Detection, as a classical computer vision task, has been extensively studied and discussed in the community. In recent years, SOD has made great progress with the rapid development of deep learning algorithms, with convolutional neural networks (CNNs) [2, 19, 34, 44] being at the forefront, leading

to the emergence of diverse algorithms. Given that Fully Convolutional Networks (FCNs) can capture rich spatial and multi-scale information, numerous approaches have attempted to effectively integrate these multi-scale features to obtain more accurate representations. Early methods, such as [6, 22], involved resizing the feature maps from different stages of the backbone network to a unified spatial size and performing one-time fusion. In contrast, some recent works, for example [21, 52, 60–62], employ a decoder to progressively fuse the deep features in the encoder with the shallow ones.

Saliency Object Ranking: Saliency Object Ranking is built upon the foundation of saliency object detection, which not only requires detecting each saliency object but also determining the degree of saliency for each individual object. Islam et al. [24] proposed the task of saliency object ranking. However, their approach to determining the saliency degree of an instance involves predicting the saliency value for each pixel of the instance and then averaging them. Siris et al. [45] were the first to construct a model for the saliency object ranking task based on instance segmentation (Mask-RCNN) [20], which reduced the difficulty of the task. Similarly, Liu et al. [31] and Fang et al. [13] also chose to build saliency object ranking models based on Mask-RCNN [20]. The former introduced multiple graphs to infer relationships between objects, while the latter incorporated object position coordinates before extracting target features. Tian et al. [49] were the first to employ a query-based instance segmentation algorithm [14] to construct the saliency object ranking task, and they designed a sophisticated object-context attention mechanism to understand the importance of the object in the image.

Instance Segmentation: Instance segmentation aims to distinguish each interested instance in an image and assign a pixel-level mask with a corresponding category label to it. In 2017, He et al. [20]proposed Mask-RCNN, which builds upon Faster-RCNN [41] by introducing the ROI-Align module to more accurately extract target features. Mask-RCNN has been widely applied in various downstream tasks, including the field of Saliency Object Ranking, such as [13, 31, 45]. In recent years, with the rapid development of Transformers in the visual domain [11, 33], a novel and straightforward approach [5] for object detection has been devised. Instead of generating a huge number of proposals nor considering a complex anchor design, it simply uses a fixed number of queries to represent different instances. This has led to a series of query-based instance segmentation algorithms, such as [7, 8, 10, 14, 29].

Graph representation learning and Graph Neural Networks: Graph representation learning focuses on mapping graph structures while preserving the topological properties for more effective analysis, modelling, and prediction. Traditional methods in graph representation learning include DeepWalk [39], node2vec [16], etc. With the advancement of deep learning, graph neural networks (GNNs) [15, 42] have gradually become the cornerstone of graph representation learning. In recent years, numerous excellent GNN algorithms have emerged, such as GCN [27], GAT [51], GraphSage [17], GIN [58], GGCN [4], and many more. Graph representation learning has also piqued the interest of researchers in the computer vision domain, such as image captioning [59], action recognition [53], etc. In particular, automatically learned multi-dimensional edge features within graph representations have shown strong performances for facial analysis [1, 26, 35, 47, 56], audio analysis [23], recommendation system [32] and personality recognition [43, 46]. In the field of saliency object ranking, Liu et al. [31] also successfully employed GNNs to model relationships between objects [31].

3. Methodology

Algorithm 1: The pipeline of the proposed DSGNN.

Input: Input data samples x; initial shape queries Q^s; initial texture queries Q^t; a backbone named B; a Pixel Decoder named PD; a Shape Transformer Decoder named STD; a Texture Transformer Decoder named TTD; a STGDS module; and a CIGDS module.

Output: Saliency predictions of input data samples $p(x)$.

Generating image features $F \leftarrow \mathrm{B}(x)$;
Generating pixel embeddings $E \leftarrow \mathrm{PD}(F)$;
Generating shape queries of candidate saliency objects $\widehat{Q}^s \leftarrow \mathrm{STD}(Q^s, E)$;
Generating texture queries of candidate saliency objects $\widehat{Q}^t \leftarrow \mathrm{TTD}(Q^t, E)$;
Generating shape predictions, texture predictions and fusion features of candidate saliency objects $S, T, \widehat{V}^f \leftarrow \mathrm{STGDS}(\widehat{Q}^s, \widehat{Q}^t, E)$;
Generating saliency classification and ranking predictions of candidate saliency objects $sc, sr \leftarrow \mathrm{CIGDS}(\widehat{V}^f)$;
Combining S with the predicted sc, sr to generate the final saliency ranking map $p(x)$.

Network Overview: As illustrated in Fig. 2, our model is built at the top the Mask2Former [8], which extracts saliency object-related latent feature maps from the input image. Under this setting, our DSGNN starts with a pair of transformer decoders attached at the top of the Mask2former, where each individually extracts the shape Queries $Q^s = \{q_i^s\}_{i=0}^N$ and texture Queries $Q^t = \{q_i^t\}_{i=0}^N$ of the target objects detected by the Mask2Former (N

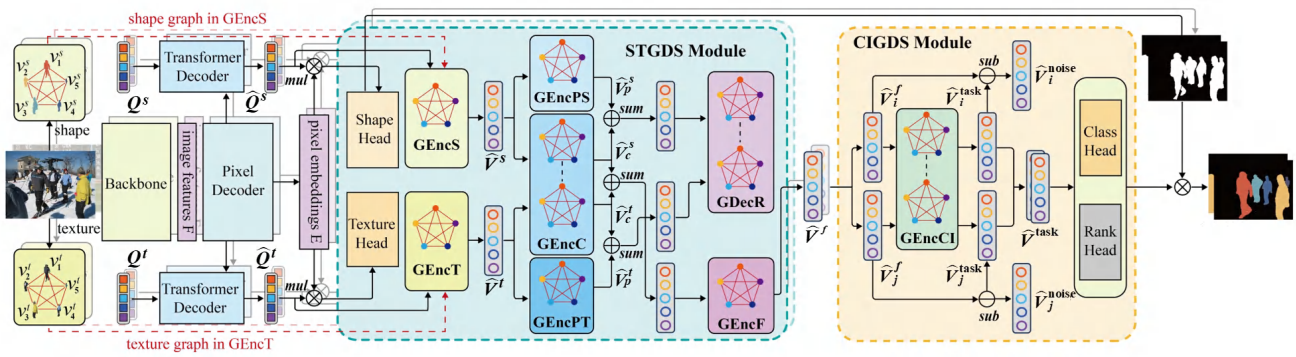


Figure 2. Overview of the proposed network architecture. Our model starts with a pair of transformer decoders attached at the top of the Mask2former [8], to extract the shape and texture relationship cues of all objects. Then, we use a STGDS Module to disentangle the saliency ranking-relevant cues from the shape and texture relationships and subsequently fuse these features. Furthermore, we use a CIGDS Module to capture a unified representations for objects with the same saliency level across different images. Finally, we combine the predictions from the Shape Head, Class Head, and Rank Head to obtain the ultimate saliency object ranking map.

deotes the number of object queries). Then, a **Shape-Texture Graph Domain Separation (STGDS)** module is proposed to disentangle the saliency ranking-relevant and irrelevant cues within both shape and texture features for each object, where multi-dimensional edge features are learned to comprehensively describe the saliency-relevant shape and texture relationships between every pair of objects, respectively. After that, a **Cross Image Graph Domain Separation (CIGDS)** module is proposed to further seek a unified representation for targets with the same saliency level across different images. The main novelty of our approach lies in: (1) in comparison to the methods [13, 31, 45, 49] that directly handle features of all objects, ours explicitly models the relationship between each pair of objects by individually considering their shapes and textures, where task-irrelevant information in both domains are specifically removed; (2) ours is the first approach that captures a unified representation for targets with the same saliency level across different images, which has been ignored by aforementioned approaches; (3) existing graph-based method [31] describes the relationship between objects via single-value graph edge, while our approach firstly learns multi-dimensional graph edges to capture complex, diverse and ranking-related relationships.

3.1. Shape-Texture Graph Domain Separation Module

Both shape and texture information contain not only crucial cues for determining the saliency level of the target objects but also ranking-unrelated noises. Thus, our STGDS module introduces a saliency ranking-irrelevant subspace and a saliency ranking-aware subspace for objects' shape and texture representation learning, aiming to disentangle ranking-specific properties from features that is irrelevant or detrimental to the saliency judgement.

As shown in Fig. 3, the STGDS module takes a set of objects' shape features $\widehat{Q}^s = \{\widehat{q}_i^s\}_{i=0}^N$ and texture features $\widehat{Q}^t = \{\widehat{q}_i^t\}_{i=0}^N$ as the input, where N denotes the number of object queries. These features are individually fed into a GNN Encoder to be first encoded as two individual graphs (i.e., shape graph $G^s = (V^s, E^s)$ and texture graph $G^t = (V^t, E^t)$), where nodes $v_i^s \in V^s$ and $v_i^t \in V^t$ represent the shape feature $\widehat{q}_i^s$ and texture feature $\widehat{q}_i^t$ of an object, respectively. For any two nodes v_i^s and v_j^s in G^s, there are two edges $e_{i,j}^s, e_{j,i}^s \in E^s$ connecting them, while each node v_i^s also has a self-loop edge $e_{i,i}^s$. Specifically, the edge $e_{i,j}^s$ from node v_i^s to v_j^s is generated by computing the cross attention [50] between them, where v_i^s is served as query and v_j^s is treated as key and value. This process is defined as:

$$e_{i,j}^s = \text{softmax}(\frac{v_i^s W_q (v_j^s W_k)^T}{\sqrt{d_k}}) v_j^s W_v \quad (1)$$

where W_q, W_k and W_v are learnable weights; d_k is a scaling factor. After obtaining the set of all edges $E^s = \{e_{i,j}^s\}_{i=0,j=0}^{N,N}$, we can construct the shape graph $G^s = (V^s, E^s)$, and the texture graph G^t has the same topology and generation method as G^s.

Then, the shape and texture graphs are separately passed through a L-layer GGCN [4] ($L = 2$ in this paper) to learn relationship cues between objects' shapes as well as their textures via graph edges, and encoded into the node features ultimately. In this paper, all the GNN Encoders mentioned below follow the same methodology for constructing graphs and have identical network architectures. The whole process can be described as:

$$\widehat{V}^s = \text{GEncS}(\widehat{Q}^s; \theta_s), \widehat{V}^t = \text{GEncT}(\widehat{Q}^t; \theta_t) \quad (2)$$

where $\text{GEnc}(x; \theta)$ represents a GNN Encoder parameterized by θ; $\widehat{V}^s = \{\widehat{v}_i^s\}_{i=0}^N$ and $\widehat{V}^t = \{\widehat{v}_i^t\}_{i=0}^N$ denote the graph node feature sets produced by GEncS and GEncT,

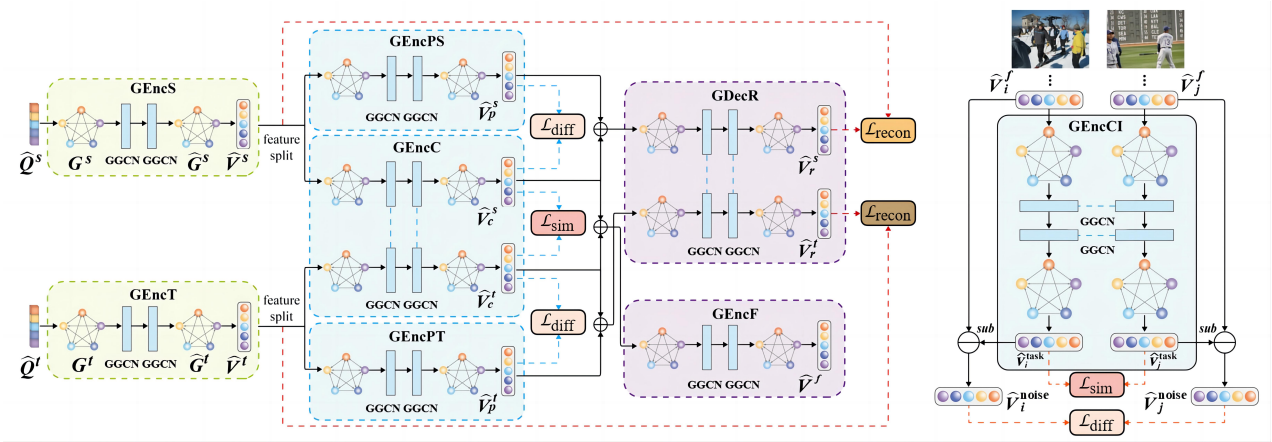


Figure 3. Structure of our STGDS (left) and CIGDS (right) module. Each unit employs a two-layer GGCN [4].

which encode saliency degree-related shape and texture cues of all objects, respectively. The obtained $\widehat{V}^s$ and $\widehat{V}^t$ are jointly fed to a GNN encoder GEncC with shared parameters that are responsible for capturing saliency-relevant shared domain representations $\widehat{V}_c^s$ and $\widehat{V}_c^t$ as:

$$\widehat{V}_c^s = \text{GEncC}(\widehat{V}^s;\theta_c), \widehat{V}_c^t = \text{GEncC}(\widehat{V}^t;\theta_c) \quad (3)$$

where $\widehat{V}_c^s$ and $\widehat{V}_c^t$ separated the saliency degree-related relationships cues between objects' shapes as well as their textures from $\widehat{V}^s$ and $\widehat{V}^s$, respectively.

Moreover, $\widehat{V}^s$ and $\widehat{V}^t$ are passed through their saliency degree-unrelated GNN Encoders at the same time to extract the task-irrelevant features:

$$\widehat{V}_p^s = \text{GEncPS}(\widehat{V}^s;\theta_p^s), \widehat{V}_p^t = \text{GEncPT}(\widehat{V}^t;\theta_p^t) \quad (4)$$

where $\widehat{V}_p^s$ and $\widehat{V}_p^t$ are node features obtained after GEncPS and GEncPT individually encoded the saliency degree-irrelevant cues that are specific to each domain in shape and texture relationship node features $\widehat{V}^s$ and $\widehat{V}^s$. By establishing orthogonal saliency degree-unrelated Encoders GEncPS and GEncPT to the shared parameter Encoder GEncC, we can ensure the extracted features $\widehat{V}_p^s$ and $\widehat{V}_p^t$ are distinct from the shared features. Then, to prevent the model from producing trivial solutions (ensure the functionality of the saliency degree-unrelated domain), we aim to use the features from the saliency degree-related and unrelated domain to recover the pre-disentangled features:

$$\widehat{V}_r^s = \text{GDecR}((\widehat{V}_c^s+\widehat{V}_p^s);\theta_r), \widehat{V}_r^t = \text{GDecR}((\widehat{V}_c^t+\widehat{V}_p^t);\theta_r) \quad (5)$$

where $\widehat{V}_r^s$ and $\widehat{V}_r^t$ represent the recovered shape and texture features of all objects, which contain saliency degree-related relationship cues as well as saliency degree-unrelated relationship cues. $\text{GDecR}(x;\theta)$ represents a GNN Decoder, which shares the same architecture as the GNN Encoder. Finally, we fuse the task-relevant portions $\widehat{V}_c^s$, $\widehat{V}_c^t$ and utilises the fused features to accomplish the final prediction task:

$$\widehat{V}^f = \text{GEncF}((\widehat{V}_c^s + \widehat{V}_c^t);\theta_f) \quad (6)$$

where $\widehat{V}^f$ represent the fused task-relevant node features.

3.2. Cross Image Domain Separation Module

As the goal is to learn a model that can accurately estimate saliency levels of objects in different images, this module aims to explicitly explore the saliency degree subspace that is robust to different scenes by creating a unified representation that can encompass the characteristics of targets with the same saliency levels in different images. Building upon this hypothesis, we employ a GNN Encoder to capture the unified representations across images (Fig. 3 right). The features $\{\widehat{V}_i^f\}$ of the target graph acquired by the STGDS module can be formulated as:

$$\{\widehat{V}_i^{\text{task}}\}_i^B = \text{GEncCI}(\widehat{V}_i^f;\theta_d)_i^B \quad (7)$$

where B represents the batch size of the input. As the saliency degree-unrelated attributes of each image are distinct, creating a saliency degree-unrelated subspace for each image is computationally inefficient. As a result, we directly subtract the original input features from the shared representation we captured in order to obtain the saliency degree-unrelated features for each image:

$$\{\widehat{V}_i^{\text{noise}}\}_i^B = \{\widehat{V}_i^f - \widehat{V}_i^{\text{task}}\}_i^B \quad (8)$$

where $\widehat{V}_i^{\text{noise}}$ represent the noise attributes for each image.

3.3. Training Strategy

To train our DSGNN, the overall loss function consists of three parts, including task-specific loss $\mathcal{L}_{\text{task}}$, loss $\mathcal{L}_{\text{st}}$ for the STGDS module and loss $\mathcal{L}_{\text{ci}}$ for the CIGDS module, which is formulated as:

$$\mathcal{L} = \alpha\mathcal{L}_{\text{task}} + \beta\mathcal{L}_{\text{st}} + \lambda\mathcal{L}_{\text{ci}} \quad (9)$$

where α, β and λ are weights to describe the relative importance of these three loss parts. Specifically, the task-related loss $\mathcal{L}_{\text{task}}$ jointly supervises the model to generate the final saliency predictions, which is a combination of segmentation loss, restoration loss, classification loss and saliency

ranking loss as:

$$\mathcal{L}_{\text{task}} = \mathcal{L}_{\text{seg}} + \mathcal{L}_{\text{restor}} + \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{rk}} \tag{10}$$

Within this framework, $\mathcal{L}_{\text{seg}}$ and $\mathcal{L}_{\text{cls}}$ utilize the same loss configurations as Mask2former [8]. Specifically, $\mathcal{L}_{\text{seg}}$ trains the model to predict the shape predictions from the shape head while $\mathcal{L}_{\text{restor}}$ is applied for the restoration of texture predictions from the texture head using the Mean Squared Error (MSE) loss. $\mathcal{L}_{\text{cls}}$ serves to discern whether predictions in class head qualifies as a saliency object and $\mathcal{L}_{\text{rk}}$ is guided by the saliency ranking loss [31] for ranking the saliency level of predictions in the rank head.

$\mathcal{L}_{\text{st}}$ denotes the loss used for the STGDS module. To separate the feature representations as intended, we apply a difference loss between $\widehat{V}_p^s$ and $\widehat{V}_c^s$, as well as between $\widehat{V}_p^t$ and $\widehat{V}_c^t$ to encourage the divergence of saliency degree-related and unrelated representations. Additionally, a similarity loss is employed between $\widehat{V}_c^s$ and $\widehat{V}_c^t$ for maintaining the similarity of the shared-domain representations. Finally, to prevent the model from producing trivial solutions, we introduce a reconstruction loss between between $\widehat{V}_r^s$ and $\widehat{V}^s$, as well as between $\widehat{V}_r^t$ and $\widehat{V}^t$:

$$\mathcal{L}_{\text{st}} = \mathcal{L}_{\text{recon}} + \mathcal{L}_{\text{diff}} + \mathcal{L}_{\text{sim}} \tag{11}$$

where $\mathcal{L}_{\text{recon}}$ is the MSE loss, $\mathcal{L}_{\text{diff}}$ utilizes the loss from [3], and $\mathcal{L}_{\text{sim}}$ is the Pearson correlation loss, where Pearson similarity can be expressed as:

$$\mathcal{P}_{\text{sim}}(x,y) = \frac{\sum_{i=1}^{n} x_i y_i - \frac{1}{n}\sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{\sqrt{(\sum_{i=1}^{n} x_i^2 - \frac{1}{n}(\sum_{i=1}^{n} x_i)^2)(\sum_{i=1}^{n} y_i^2 - \frac{1}{n}(\sum_{i=1}^{n} y_i)^2)}} \tag{12}$$

Hence, the Pearson correlation loss is defined as:

$$\mathcal{L}_{\text{pearson}}(x,y) = 1 - \mathcal{P}_{\text{sim}}(x,y) \tag{13}$$

where x, y are two node features.

Finally, $\mathcal{L}_{\text{ci}}$ is the loss function in the CIGDS module. To ensure the successful separation of the desired shared representations during this process, we also apply a similarity loss in $\{V_i^{\text{task}}\}_i^B$ to facilitate the learning of similar distributions. At the same time, we employ a difference loss in $\{V_i^{\text{noise}}\}_i^B$ to compel distinct features between the saliency degree-related and unrelated domain, ensuring the prevention of trivial solutions as:

$$\mathcal{L}_{\text{ci}} = \mathcal{L}_{\text{diff}} + \mathcal{L}_{\text{sim}} \tag{14}$$

where $\mathcal{L}_{\text{diff}}$ and $\mathcal{L}_{\text{sim}}$ have equivalent roles to the respective parts in $\mathcal{L}_{\text{st}}$. See Supplementary Material for more details.

4. Experiments

4.1. Experimental Setup

Datasets. Our experiments were conducted on two widely-used datasets, ASSR [45] and IRSR [31], both of which are derived from the MS-COCO [30] and SALICON [25] datasets. However, they differ in determining target saliency: The ASSR dataset primarily assesses target saliency based on the eye gaze order, which annotates the top-5 saliency objects in each image and provides 7,646, 1,436, and 2,418 images for training, validation, and testing, respectively. Meanwhile, the IRSR dataset assesses target saliency based on the eye gaze duration, which consists of 6,059 images for training and 2,929 images for testing, where each image is annotated up to 8 objects.

Metrics. We follow previous studies [31, 49] to employ three metrics: (1) Mean Absolute Error (MAE) that compares the difference between the predicted saliency ranking mask and the ground-truth at the pixel-level; (2) Saliency Object Ranking (SOR) that measures the degree of correlation between the predicted saliency value list and the ground-truth using Spearman correlation; and (3) Segmentation-Aware SOR (SA-SOR) that employs Pearson correlation to assess the correlation between the predicted saliency ranking and the ground-truth ranking, which considers both false positives and false negatives related to saliency objects. Please refer to [31, 45] for details.

Implementation Details. We employ ResNet [19] and Swin Transformer [33] pretrained on MS-COCO [30] training split as the backbones for our approach. Following the training strategy of Mask2former [8], we employed random horizontal flipping and multi-scale cropping as data augmentations. All images were fixed to 480×640 before feeding into the network. Our model was trained with a total batch size of 8 for 36,000 iterations. The initial learning rate was set to 5e-5 and was reduced by a factor of 0.1 at the 15,000th and 30,000th iteration. We utilized the AdamW optimizer with a weight decay of 0.05 to optimize our model. The number of queries was set to 5 and 8 for the ASSR [45] and IRSR [31] datasets. The weight ratios of the loss terms are set as $\alpha : \beta : \lambda = 1 : 1 : 1$. Our approach was implemented using the mmdetection toolkit and trained on four RTX 3090 GPUs.

4.2. Comparison to existing methods

Quantitative Evaluation: Table 1 shows the proposed method compared with state-of-the-art approaches, such as saliency object ranking (e.g., RSDNet [24], ASSR [45], IRSR [31], SOR [13], and OCOR [49] methods) and saliency object detection (e.g., S4Net [12], BASNet [40], CPD-R [54], and SCRN [55] methods). To ensure a fair comparison, we report the results using ResNet50 [19] and Swin-L [33] as backbones. Significant progress was noted, particularly in the MAE and SA-SOR metrics. Our ResNet50-based model even surpasses previous Swin-L backbone models, indicating significant advantages in segmentation and ranking accuracy. When Swin-L was used as the backbone, the best scores to date were observed. How-

Table 1. Quantitative comparison with the current state-of-the-art methods for saliency object detection and ranking. SID, SOD, and SOR represent saliency instance detection, saliency object detection and saliency object ranking, respectively. Methods marked with † denote results directly replicated from the original paper. Methods with * indicate results testing using their open-source model. – symbolizes missing results due to a lack of results/models. The best and second best results are indicated with bold font and brackets, respectively.

Method	Task	Backbone	ASSR Dataset [45]			IRSR Dataset [31]		
			MAE↓	SOR↑	SA-SOR↑	MAE↓	SOR↑	SA-SOR↑
S4Net† [12]	SID	ResNet-50	0.150	0.891	-	-	-	-
BASNet† [40]	SOD	ResNet-34	0.115	0.707	-	-	-	-
CPD-R† [54]	SOD	ResNet-50	0.100	0.766	-	-	-	-
SCRN† [55]	SOD	ResNet-50	0.116	0.756	-	-	-	-
RSDNet [24]	SOR	ResNet-101	0.158	0.717	0.499	0.129	0.735	0.460
ASSR [45]	SOR	ResNet-101	0.101	0.792	0.667	0.125	0.714	0.388
IRSR [31]	SOR	ResNet-50	0.105	0.811	0.709	0.085	[0.806]	0.565
SOR [13]	SOR	VOVNet-39	0.081	0.841	-	-	-	-
OCOR* [49]	SOR	Swin-L	0.079	**0.883**	0.687	-	-	-
Ours	SOR	ResNet-50	[0.060]	0.843	[0.716]	[0.073]	0.785	[0.568]
Ours	SOR	Swin-L	**0.047**	[0.853]	**0.761**	**0.067**	**0.807**	**0.606**

ever, we did not achieve SOTA results on the SOR metric, which disregards scenarios where targets are missed, thereby diminishing its reliability. This is also the reason why S4Net [12], despite recording the highest MAE results, still obtains exceptionally good outcomes on the SOR metric. Here, we did not compare our results with those reported in OCOR [49], as the results reproduced from their open-source code have been frequently claimed to be different from their reported results [1].

Qualitative Evaluation: We further qualitatively evaluate our approach. As shown in Fig. 1a, our DSGNN outperform current SOTA methods [24, 31, 45, 49] by a large margin. While we can further observed in Fig. 1b, most methods in the first two rows either miss or falsely identify the target, indicating that the proposed method can locate the saliency targets in the image more accurately. The third row shows most methods can accurately find the saliency subjects but fail to understand the context of the image, such as the "horse" in the example. This instance shows the method's powerful ability to understand contexts. The last two rows indicate all methods can find and segment the target but often find it challenging to judge saliency accurately, particularly cases with lower saliency rankings, which shows the strength of our saliency reasoning module.

4.3. Ablation Studies

To validate the effectiveness of each module, Table 2 shows the first experiments using the shape and texture information of the target. After the STGDS module is included, the effects of adding the CIGDS module are evaluated. The results indicate that, as each module is incorporated, performance metrics gradually improve. The final results outperform all others, indicating the effectiveness of each module. Fig. 4 shows that the targets can be segmented well by using either the shape or texture information. After adding the STGDS module, the judgement of target saliency is significantly improved, particularly in those with higher saliency. Due to similarity loss, most results for the shape and texture branches were consistent. However, after merging the shape and texture branches, the results were still enhanced. After adding the CIGDS module, clear improvements in saliency rankings were noted.

Table 2. Comparison between STGDS and CIGDS modules.

Backbone		STGDS		CIGDS	MAE↓	SOR↑	SA-SOR↑
shape	texture	shape	texture				
✓					0.054	0.835	0.739
✓	✓				0.054	0.836	0.744
✓	✓	✓			0.049	0.847	0.755
✓	✓		✓		0.052	0.849	0.758
✓	✓	✓	✓		0.049	0.851	0.759
✓	✓			✓	0.053	0.845	0.754
✓	✓	✓	✓	✓	0.047	0.853	0.761

Tables 3 and 4 validate the effects of the different processes in the STGDS and CIGDS modules. For instance, when the effects of individual or partial losses on the final performance were assessed, it was found that each process contributes to performance improvement. These results validate the two hypotheses: (1) task-irrelevant noise is present in the shape and texture information of the targets; task-relevant data shows similar distribution in saliency-aware space; (2) despite variations in the feature distribution of targets in different images, targets with the same saliency level share similar relationships.

[1] https://github.com/GrassBro/OCOR

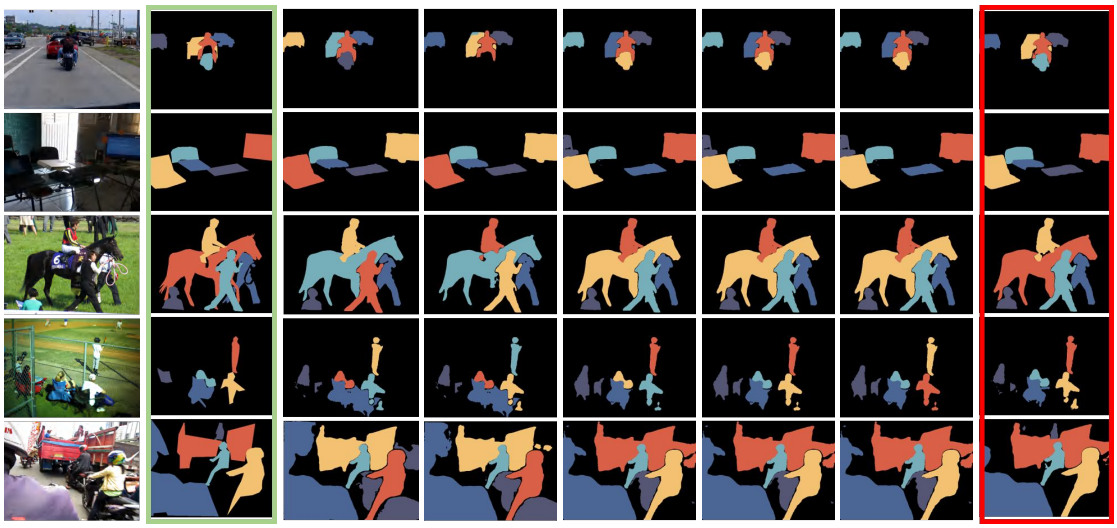


Figure 4. Qualitative Results for Ablation Study. From left to right, the qualitative results are presented for the baseline using the shape information, baseline using texture information, the results of the shape branch after incorporating the STGDS module, the results of texture branch after incorporating the STGDS module, the results of combining shape and texture information, and the final results after adding the CIGDS module. It is evident from the results that both saliency object recognition and ranking accuracy progressively improved.

Table 3. Comparison between losses in STGDS module.

STGDS			MAE↓	SOR↑	SA-SOR↑
similarity	difference	reconstruction			
✓			0.058	0.843	0.746
	✓		0.052	0.849	0.750
		✓	0.053	0.848	0.747
✓	✓		0.052	0.844	0.758
	✓	✓	0.051	0.849	0.756
✓		✓	0.054	0.849	0.753
✓	✓	✓	0.047	0.853	0.761

Table 4. Comparison between losses in CIGDS module.

CIGDS		MAE↓	SOR↑	SA-SOR↑
similarity	difference			
✓		0.051	0.849	0.759
	✓	0.050	0.852	0.760
✓	✓	0.047	0.853	0.761

Table 5 shows experiments with MLP, Conv layers, and Transformer Blocks [50] and various graph neural networks (GNNs) including GCN [27], GAT [51], GraphSage [17], and GGCN [4], where GGCN-ME denotes GGCN with multi-dimensional edges. The results indicate that, compared to MLP, Conv, and Transformer Blocks, graphs have a clear advantage in relation inference. Importantly, when multi-dimensional edges are employed, the proposed method shows significant improvement, confirming the complexity and diversity of relationships among targets. The details of the learned multi-dimensional edges are visualized and discussed in the Supplementary Material.

Table 5. Comparison between different components in STGDS and CIGDS modules.

Method	MAE↓	SOR↑	SA-SOR↑
MLP	0.052	0.849	0.749
Conv	0.053	0.844	0.750
Transformer [50]	0.051	0.852	0.753
GCN [27]	0.052	0.850	0.754
GAT [51]	0.050	0.846	0.755
GraphSage [17]	0.051	0.848	0.754
GGCN [4]	0.049	0.851	0.757
GGCN-ME [4]	0.047	0.853	0.761

5. Conclusion

In this paper, we propose a novel method for saliency object ranking that takes into account both the shape and texture information of targets while suppressing task-unrelated noise in these features. Additionally, we uncover a unified representation encompassing the characteristics of targets with the same saliency levels in different images. Finally, we infer complex relationships between targets using multi-edge graphs. Our approach achieves state-of-the-art performance on widely used open-source datasets.

References

[1] Nida Itrat Abbasi, Siyang Song, and Hatice Gunes. Statistical, spectral and graph representations for video-based facial expression recognition in children. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech*

and Signal Processing (ICASSP), pages 1725–1729. IEEE, 2022. 3

[2] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017. 2

[3] Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan. Domain separation networks. *Advances in neural information processing systems*, 29, 2016. 2, 6

[4] Xavier Bresson and Thomas Laurent. Residual gated graph convnets. *arXiv preprint arXiv:1711.07553*, 2017. 3, 4, 5, 8

[5] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 3

[6] Shuhan Chen, Xiuli Tan, Ben Wang, and Xuelong Hu. Reverse attention for salient object detection. In *Proceedings of the European conference on computer vision (ECCV)*, pages 234–250, 2018. 3

[7] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *Advances in Neural Information Processing Systems*, 34:17864–17875, 2021. 3

[8] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1290–1299, 2022. 2, 3, 4, 6

[9] Francis Crick and Christof Koch. Towards a neurobiological theory of consciousness. In *Seminars in the Neurosciences*, pages 263–275. Saunders Scientific Publications, 1990. 2

[10] Bin Dong, Fangao Zeng, Tiancai Wang, Xiangyu Zhang, and Yichen Wei. Solq: Segmenting objects by learning queries. *Advances in Neural Information Processing Systems*, 34: 21898–21909, 2021. 3

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3

[12] R Fan, MM Cheng, Q Hou, TJ Mu, J Wang, and SM Hu. S4net: Single stage salient-instance segmentation. computational visual media 6 (2), 191–204 (june 2020). 6, 7

[13] Hao Fang, Daoxin Zhang, Yi Zhang, Minghao Chen, Jiawei Li, Yao Hu, Deng Cai, and Xiaofei He. Salient object ranking with position-preserved attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16331–16341, 2021. 2, 3, 4, 6, 7

[14] Yuxin Fang, Shusheng Yang, Xinggang Wang, Yu Li, Chen Fang, Ying Shan, Bin Feng, and Wenyu Liu. Instances as queries. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6910–6919, 2021. 3

[15] Marco Gori, Gabriele Monfardini, and Franco Scarselli. A new model for learning in graph domains. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, pages 729–734. IEEE, 2005. 3

[16] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864, 2016. 3

[17] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017. 3, 8

[18] Jungong Han, Eric J Pauwels, and Paul De Zeeuw. Fast saliency-aware multi-modality image fusion. *Neurocomputing*, 111:70–80, 2013. 2

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2, 6

[20] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 2, 3

[21] Shengfeng He, Jianbo Jiao, Xiaodan Zhang, Guoqiang Han, and Rynson WH Lau. Delving into salient object subitizing and detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1059–1067, 2017. 3

[22] Qibin Hou, Ming-Ming Cheng, Xiaowei Hu, Ali Borji, Zhuowen Tu, and Philip HS Torr. Deeply supervised salient object detection with short connections. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3203–3212, 2017. 3

[23] Yuanbo Hou, Siyang Song, Chuang Yu, Wenwu Wang, and Dick Botteldooren. Audio event-relational graph representation learning for acoustic scene classification. *IEEE signal processing letters*, 2023. 3

[24] Md Amirul Islam, Mahmoud Kalash, and Neil DB Bruce. Revisiting salient object detection: Simultaneous detection, ranking, and subitizing of multiple salient objects. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7142–7150, 2018. 3, 6, 7

[25] Ming Jiang, Shengsheng Huang, Juanyong Duan, and Qi Zhao. Salicon: Saliency in context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1072–1080, 2015. 6

[26] Hyeongjin Kim, Jong-Ha Lee, and Byoung Chul Ko. Facial expression recognition in the wild using face graph and attention. *IEEE Access*, 2023. 3

[27] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016. 3, 8

[28] Youngwan Lee and Jongyoul Park. Centermask: Real-time anchor-free instance segmentation. In *CVPR*, 2020. 2

[29] Feng Li, Hao Zhang, Huaizhe Xu, Shilong Liu, Lei Zhang, Lionel M Ni, and Heung-Yeung Shum. Mask dino: Towards a unified transformer-based framework for object detection and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3041–3050, 2023. 3

[30] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In

Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13, pages 740–755. Springer, 2014. 6

[31] Nian Liu, Long Li, Wangbo Zhao, Junwei Han, and Ling Shao. Instance-level relative saliency ranking with graph reasoning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):8321–8337, 2021. 1, 2, 3, 4, 6, 7

[32] Xiao Liu, Shunmei Meng, Qianmu Li, Lianyong Qi, Xiaolong Xu, Wanchun Dou, and Xuyun Zhang. Smef: Social-aware multi-dimensional edge features-based graph representation learning for recommendation. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 1566–1575, 2023. 3

[33] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 3, 6

[34] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 2

[35] Cheng Luo, Siyang Song, Weicheng Xie, Linlin Shen, and Hatice Gunes. Learning multi-dimensional edge feature-based au relation graph for facial action unit recognition. *arXiv preprint arXiv:2205.01782*, 2022. 3

[36] Brad C Motter. Focal attention produces spatially selective processing in visual cortical areas v1, v2, and v4 in the presence of competing stimuli. *Journal of neurophysiology*, 70 (3):909–919, 1993. 2

[37] Ernst Niebur, Christof Koch, and Christopher Rosin. An oscillation-based model for the neuronal basis of attention. *Vision research*, 33(18):2789–2802, 1993. 2

[38] Sinno Jialin Pan, Ivor W Tsang, James T Kwok, and Qiang Yang. Domain adaptation via transfer component analysis. *IEEE transactions on neural networks*, 22(2):199–210, 2010. 2

[39] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710, 2014. 3

[40] Xuebin Qin, Zichen Zhang, Chenyang Huang, Chao Gao, Masood Dehghan, and Martin Jagersand. Basnet: Boundary-aware salient object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7479–7489, 2019. 6, 7

[41] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015. 3

[42] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE transactions on neural networks*, 20(1): 61–80, 2008. 3

[43] Zilong Shao, Siyang Song, Shashank Jaiswal, Linlin Shen, Michel Valstar, and Hatice Gunes. Personality recognition by modelling person-specific cognitive processes using graph representation. In *proceedings of the 29th ACM international conference on multimedia*, pages 357–366, 2021. 3

[44] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 2

[45] Avishek Siris, Jianbo Jiao, Gary KL Tam, Xianghua Xie, and Rynson WH Lau. Inferring attention shift ranks of objects for image saliency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12133–12143, 2020. 1, 2, 3, 4, 6, 7

[46] Siyang Song, Zilong Shao, Shashank Jaiswal, Linlin Shen, Michel Valstar, and Hatice Gunes. Learning person-specific cognition from facial reactions for automatic personality recognition. *IEEE Transactions on Affective Computing*, 2022. 3

[47] Siyang Song, Yuxin Song, Cheng Luo, Zhiyuan Song, Selim Kuzucu, Xi Jia, Zhijiang Guo, Weicheng Xie, Linlin Shen, and Hatice Gunes. Gratis: Deep learning graph representation with task-specific topology and multi-dimensional edge features. *arXiv preprint arXiv:2211.12482*, 2022. 2, 3

[48] Yusuke Sugano, Yasuyuki Matsushita, and Yoichi Sato. Calibration-free gaze sensing using saliency maps. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2667–2674. IEEE, 2010. 2

[49] Xin Tian, Ke Xu, Xin Yang, Lin Du, Baocai Yin, and Rynson WH Lau. Bi-directional object-context prioritization learning for saliency ranking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5882–5891, 2022. 1, 2, 3, 4, 6, 7

[50] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 2, 4, 8

[51] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017. 3, 8

[52] Tiantian Wang, Lihe Zhang, Shuo Wang, Huchuan Lu, Gang Yang, Xiang Ruan, and Ali Borji. Detect globally, refine locally: A novel approach to saliency detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3127–3135, 2018. 3

[53] Xiaolong Wang and Abhinav Gupta. Videos as space-time region graphs. In *Proceedings of the European conference on computer vision (ECCV)*, pages 399–417, 2018. 3

[54] Zhe Wu, Li Su, and Qingming Huang. Cascaded partial decoder for fast and accurate salient object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3907–3916, 2019. 6, 7

[55] Zhe Wu, Li Su, and Qingming Huang. Stacked cross refinement network for edge-aware salient object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7264–7273, 2019. 6, 7

[56] Jiaqi Xu, Siyang Song, Keerthy Kusumam, Hatice Gunes, and Michel Valstar. Two-stage temporal modelling framework for video-based depression recognition using graph representation. *arXiv preprint arXiv:2111.15266*, 2021. 3

[57] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057. PMLR, 2015. 2

[58] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018. 3

[59] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. Exploring visual relationship for image captioning. In *Proceedings of the European conference on computer vision (ECCV)*, pages 684–699, 2018. 3

[60] Lu Zhang, Ju Dai, Huchuan Lu, You He, and Gang Wang. A bi-directional message passing model for salient object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1741–1750, 2018. 3

[61] Pingping Zhang, Dong Wang, Huchuan Lu, Hongyu Wang, and Xiang Ruan. Amulet: Aggregating multi-level convolutional features for salient object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 202–211, 2017.

[62] Yunzhi Zhuge, Yu Zeng, and Huchuan Lu. Deep embedding features for salient object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 9340–9347, 2019. 3