Joint Semantic Mining for Weakly Supervised RGB-D Salient Object Detection

Jingjing Li\textsuperscript{1,*}, Wei Ji\textsuperscript{1,* (✉)}, Qi Bi\textsuperscript{2}, Cheng Yan\textsuperscript{3}, Miao Zhang\textsuperscript{4}, Yongri Piao\textsuperscript{4}, Huchuan Lu\textsuperscript{4,5}, Li Cheng\textsuperscript{1}

\textsuperscript{1}University of Alberta, Canada \quad \textsuperscript{2}Wuhan University, China \quad \textsuperscript{3}Tianjin University, China \quad \textsuperscript{4}Dalian University of Technology, China \quad \textsuperscript{5}Pengcheng Lab, Shenzhen, China

Abstract

Training saliency detection models with weak supervisions, e.g., image-level tags or captions, is appealing as it removes the costly demand of per-pixel annotations. Despite the rapid progress of RGB-D saliency detection in fully-supervised setting, it however remains an unexplored territory when only weak supervision signals are available. This paper is set to tackle the problem of weakly-supervised RGB-D salient object detection. The key insight in this effort is the idea of maintaining per-pixel pseudo-labels with iterative refinements by reconciling the multimodal input signals in our joint semantic mining (JSM). Considering the large variations in the raw depth map and the lack of explicit pixel-level supervisions, we propose spatial semantic modeling (SSM) to capture saliency-specific depth cues from the raw depth and produce depth-refined pseudo-labels. Moreover, tags and captions are incorporated via a fill-in-the-blank training in our textual semantic modeling (TSM) to estimate the confidences of competing pseudo-labels. At test time, our model involves only a light-weight sub-network of the training pipeline, i.e., it requires only an RGB image as input, thus allowing efficient inference. Extensive evaluations demonstrate the effectiveness of our approach under the weakly-supervised setting. Importantly, our method could also be adapted to work in both fully-supervised and unsupervised paradigms. In each of these scenarios, superior performance has been attained by our approach with comparing to the state-of-the-art dedicated methods. As a by-product, a CapS dataset is constructed by augmenting existing benchmark training set with additional image tags and captions. Code and dataset are available at \url{https://github.com/jiwei0921/JSM}.

1 Introduction

As a fundamental computer vision task, salient object detection (SOD) aims at locating and segmenting visually distinctive objects in a scene. It plays an important role in a variety of downstream applications including image retrieval \cite{31, 65}, medical analysis \cite{48, 28, 25}, multimodal fusion \cite{79, 80} and video analysis \cite{88, 81, 90}. Recent progress in supervised RGB-D SOD \cite{5, 52, 10, 26, 89, 33} has demonstrated significant benefits of engaging depth information in saliency detection from complex scenes. The success of these fully-supervised methods, however, relies heavily on the large-scale, precise, pixel-level annotations, which are often laborious and time-consuming to acquire. On the other hand, an image usually comes with additional information in its meta-data such as tags and captions from users to describe the scene context and content, which may serve as cheap weak-supervision signals. These weak supervision signals are nonetheless noisy and have mixed qualities. Similar weak-supervision signals have been explored in RGB-image based SOD \cite{63, 70, 88}, where the noisy nature of these side information is unfortunately overlooked. To further complicate the matter, the lack of explicit pixel-level supervision brings new challenge to the RGB-D SOD task: the depth values from raw depth maps are often noisy and sometimes inconsistent. For example, in Fig. 1

*means equal contributions. Wei Ji (wji3@ualberta.ca) is the corresponding author.
similar depth values are shared by the cat and the underneath couch, making it difficult to discern the salient object from backgrounds. Without the explicit pixel-level supervision, existing cross-modal fusion strategies adopted by fully-supervised RGB-D methods [36, 55, 37, 27] would simply fail.

These observations motivate us to consider the new problem of weakly-supervised RGB-D salient object detection, which takes as input the RGB and depth images, as well as weak supervision signals such as image-level tags and captions, as illustrated in Fig. 1. By removing the demand for laborious per-pixel annotations, it also brings new challenges: 1) how to address the noisy nature of the weak supervision signals; 2) how to tackle the depth noise and inconsistency to facilitate proper separation of foreground and background regions.

This leads us to propose the use of pseudo-labels with iterative refinements in training: the pseudo-label provides internal pixel-level supervision signals, which is progressively updated by reconciling the multimodal input signals and the current information flow of the neural net, based on the previous pseudo-label. As illustrated in Fig. 2, this is realized by an interaction between two core modules, namely spatial semantic modeling (SSM) and joint semantic mining (JSM): SSM is designed to capture the saliency-specific depth semantics, to eliminate the background noises in the coarse saliency prediction, and to generate a depth-refined pseudo-label. This simple yet effective module is very generic, which could be easily plugged-in different setups, including unsupervised & fully-supervised scenarios; meanwhile, the JSM module is proposed to leverage depth semantics and weak supervision signals for attaining more reliable pseudo-labels. Specifically, a partial textual input, i.e., image-level tag and caption with its salient word being masked, is fed into a dedicated textual semantic modeling or TSM to estimate the confidence scores of competing pseudo-labels, and to fill-in-the-blank. Intuitively, a semantically consistent pseudo-label should provide better context cues to reconstruct the salient word; while a closer guess of the masked word would indicate a better pseudo-label. The alternation between SSM and JSM modules is thus expected to give rise to more trustworthy pseudo-labels. At test time, it is then sufficient to take an input RGB image and activate a light-weight network to deliver its final prediction. That is, test time input involves only an RGB image, without the need of any depth map or image-level tags and captions. This drastically simplifies the input requirement and reduces the computation burden at deployment stage. Moreover, given the lack of training dataset for the weakly-supervised setting, we adapt existing RGB-D training dataset to annotate additional image-level tags and captions, which is referred to as the CapS dataset.

The main contributions of this paper are as follows. (1) A new problem of weakly-supervised RGB-D salient object detection is considered. In this regard, a CapS dataset is curated by augmenting the existing RGB-D SOD training dataset with image-level tagging and captioning annotations. (2) The key ingredient of our approach involves the production of pseudo-labels with iterative refinements, realized by iterative updates between two internal modules, SSM and JSM. Empirical experiments demonstrate the effectiveness of our approach in weakly-supervised setting. Moreover, (3) after proper adaptation of our approach in unsupervised and fully-supervised scenarios, superior performance is also observed when comparing to the respective state-of-the-art methods. (4) Our test time inference amounts to executing a light-weight saliency network: as illustrated in dotted line at Fig. 2 only an RGB image is used as its input, thus allows efficient and effective inference.

2 Related Work

RGB-D salient object detection [87, 93, 94] has been an active field of research in the past few years, where the incorporation of depth cues has been demonstrated [13, 34, 82, 75, 13, 35, 60, 43, 85]
to improve performance especially in complex scenes. Existing RGB-D methods aim to design effective feature fusion strategies for learning representative cross-modal features. Typically, Chen et al. [69] explore self-consistency among multi-scale outputs and suppress noise. Visual feature prediction (\(\Phi_{\text{vis}}\)) and depth network (\(\Phi_{\text{dep}}\)) are illustrated in Fig. 2. The masked salient word is ‘cat’.

Figure 2: An overview of our approach. Its training pipeline consists of a saliency prediction network, a SSM (Sec. 3.2) to generate depth-refined pseudo-label, a TSM (Sec. 3.3) to estimate the confidences of different pseudo-labels, and a JSM (Sec. 3.4) to refine & update pseudo-label. Our testing process only involves activating a saliency network delineated in dotted lines. More details of SSM and TSM modules are illustrated in Fig. 3. The masked salient word is ‘cat’.

Instead of using costly pixel-level annotations, some recent efforts instead explore cheap alternatives such as image-level tags (categories) [38, 63, 71, 46], image captions [70], scribble pseudo-labels [69], and noisy pseudo-labels from handcrafted methods [45, 72, 73, 68, 47, 67, 49, 66]. The pioneering work [63] leverage image-level tags or categories that could be augmented onto existing large-scale dataset at low-cost. The same scenario is also considered by Li et al. [38], where a composite pipeline combining graphical model with CNNs is designed. The trained network however tends to highlight the most discriminative region instead of the intended salient object out of the scene due to the sparse image-level supervisions. Image captions are examined by Zeng et al. [70] as supervision input; in their work image classification network and caption generation network are jointly trained to obtain pseudo-labels, which achieves descent performance. Scribble is another type of weak supervision signal, where a tiny fraction of image pixels are labeled by users as being foreground or background. Due to the annotation sparsity, object structure and details cannot be easily inferred. Zhang et al. [66] introduce a gated structure-aware loss as well as an auxiliary edge detection network to uncover the complete object. Meanwhile, Yu et al. [69] explore self-consistency among multi-scale outputs and design a local coherence loss to propagate the labels to unlabeled regions based on image features, thus enabling the detection of objects with smooth textures. However, existing weakly supervised saliency methods are solely based on RGB image. Unlike the prevalence of fully-supervised RGB-D SOD, it has never been considered the incorporation of depth cues in existing literature.

This leads us to address this problem in the presence of image tags and captions as weak supervision signals. Different from existing methods [38, 24, 63, 70] that train image classification networks or caption generation networks to delineate potential salient regions, masked salient word in captions are to be reconstructed in our work by leveraging visual saliency features. This is then used to estimate the confidence scores of pseudo-labels.

3 Methodology
3.1 The Overall Architecture
An overview of our approach is illustrated in Fig. 2. It consists of a saliency network responsible for saliency prediction, a spatial (depth) semantic modeling (SSM) to generate depth-refined pseudo-label,
a textual (caption) semantic modeling (TSM) to estimate the confidences of different pseudo-labels, and a joint semantic mining (JSM) strategy to refine & update pseudo-label. Overall our pipeline aims to gradually improve the quality of noisy pseudo-labels by jointly mining the useful spatial semantics from the depth map and textual semantics from the tags and captions, to produce more trustworthy supervision signals, which in turns results in better training of the saliency network.

Specifically, the popular encoder-decoder architecture [64] in SOD is adopted in both saliency network and depth network. Initial supervision signal for the saliency network is provided by traditional handcrafted methods. The predicted saliency map, together with the raw depth map, are processed to generate the saliency-guided spatial supervision signal for the depth network. Then the predicted saliency-oriented depth semantics is utilized to eliminate background noises (non-salient regions) in coarse prediction, and to generate a depth-refined pseudo-label. This is followed by our JSM strategy, which takes in the image-level tags and captions through TSM to estimate the confidence scores of pseudo-labels; updated pseudo-label is then formed based on the confidence-weighted depth-refined pseudo-label and current pseudo-label, which provides more trustworthy supervision signal for the saliency network. Note our test time inference involves only the black dashed portion in Fig. 2 which takes as input only the RGB image, thus enables efficient saliency prediction.

### 3.2 Saliency-oriented Spatial (Depth) Semantic Modeling

Initial pseudo-labels are generated by traditional saliency models, which often contain excessive noise. As illustrated in Fig. 3, our spatial semantic modeling (SSM) is to produce a more reliable depth-refined pseudo-label, achieved by explicitly capturing saliency-specific depth semantics from the depth map to eliminate possible background noise in the coarse saliency prediction.

Concretely, during training, we first generate a saliency-guided depth mask \( D_{\text{mask}} \) by multiplying the rough saliency prediction \( S_{\text{pred}} \) with the raw depth map \( D_{\text{map}} \) in a spatial attention manner. Here, a Gaussian smooth operation is applied to smooth the predicted saliency area, to effectively perceive and capture more saliency areas from depth. The procedure is formulated as:

\[
D_{\text{mask}} = \Omega_{\text{max}}(F_{\text{gauss}}(S_{\text{pred}}, k), S_{\text{pred}}) \odot D_{\text{map}},
\]

where \( F_{\text{gauss}}(\cdot, k) \) indicates a convolution operation with Gaussian kernel \( k \) and zero bias; \( \Omega_{\text{max}}(\cdot, \cdot) \) is a maximum function to preserve the higher values between the smoothed and the original maps; \( \odot \) is element-wise multiplication. In this paper, the size and standard deviation of kernel \( k \) are learnable through the model training procedure and are initialized with values 32 and 4, respectively.

After obtaining \( D_{\text{mask}} \), a depth network is trained to learn the saliency-specific depth semantics \( D_S \), using the mean square error (MSE) loss function. The internal inspection evidences in Fig. 4 suggest that \( D_S \) (depth semantics, 6th column) is able to capture discriminative saliency cues from the raw depth map under the supervision of \( D_{\text{mask}} \) (5th column).

In addition, the learned \( D_S \) can be further processed to generate the depth-refined pseudo-labels. This procedure is only employed when performing pseudo-label update. Specifically, we feed \( D_S \) into an Background Noise Suppression block to help eliminate the background noises (non-salient
regions) in the coarse prediction $S_{\text{pred}}$. In this block, a reverse operation is first employed on $D_S$ to highlight background regions by $1 - D_S$. This is followed by the pixel-wise subtraction to suppress the non-salient negative responses in $S_{\text{pred}}$. Finally, a ReLU function and a normalization procedure are adopted to suppress negative numbers and normalize the result to the range of [0, 1]. This procedure of obtaining the depth-refined pseudo-label $S_D$ could be expressed by

$$S_{D}^{i,j} = \frac{S_{d}^{i,j} - \min(D_{d})}{\max(D_{d}) - \min(D_{d}) + \epsilon}, \quad S_{d} = \text{ReLU}(S_{\text{pred}} - \lambda_{d}(1 - D_{S})), \quad i \in [1, H], j \in [1, W]. \quad (2)$$

Here $\lambda_{d} \in [0, 1]$ is a constant to control the degree of the subtracted background noises and avoid negatively suppressing salient regions that have relatively low saliency scores in $S_D$. Throughout experiments, $\lambda_{d}$ is empirically set to 0.5, and $\epsilon$ to 1e-5. $H$ and $W$ are the height and width of the input image, respectively.

The pseudo-label refinement dynamics are visually inspected in Fig. 4 while Fig. 8 presents the corresponding quantitative error analysis over iterations. Empirical evidence suggests as the training proceeds, quality of the pseudo-label is significantly improved. Moreover, at this stage, we can directly utilize the depth-refined label $S_D$ to update pseudo-labels (as in Fig. 6(b)). This can be referred as unsupervised RGB-D SOD since no weak labels are used.

3.3 Saliency-oriented Textual (Caption) Semantic Modeling

Previously, the mainstream use of weak labels is to train a classification network or a caption generation network, where the by-product attention maps or Class Activation Maps [62] are leveraged to determine the potential salient regions [20, 64]. It is very different in our textual semantic modeling (TSM), where the main focus is to leverage side information (i.e., image-level tags and captions) to facilitate the production of reliable training signals. Inspired by the recent success of masked language models [14], captions with missing keywords are used as input, with the expectation of the complete text being reconstructed as output. In the proposed TSM, innovatively taking as input partial text with salient word being masked, as well as the saliency-filtered visual features, our TSM is to output the reconstructed text in a fill-in-the-blank manner and to estimate the confidence scores of competing pseudo-labels. The intuition is, a semantically matching pseudo-label could provide better context cues to reconstruct the masked salient word; meanwhile, a closer guess of the masked word would indicate a better pseudo-label.

Formally, for each training data, the weak labels contain caption description $c = \{c_{i}\}_{i=1}^{n_{c}}$, image-level category $k \in \{1, \ldots, K\}$, and the position $\chi$ of the salient word (object) in the caption, where $n_{c}$ is the word number of the caption. Let $c_{i} \in \mathbb{R}^{d \times 1}$ be the word embedding of the $i$-th word in the caption, $K$ the total number of salient categories, and $\chi$ an integral number. As shown in Fig. 5 during training, the input is a masked version of caption $\hat{c} \in \mathbb{R}^{d \times \chi n_{c}}$ where the salient word $c_{\chi}$ in caption $c$ is masked with a special symbol. In order to reconstruct the masked salient word, we filter the visual feature from the saliency network by multiplying it with the learned saliency cues $S_{\text{pred}}$. We then obtain the saliency-filtered visual feature, and transform it to a feature vector $\mathbf{f}_{\text{sal}} \in \mathbb{R}^{d \times 1}$ for subsequent cross-modal fusion using a pooling operation and a fully-connected (FC) layer.

The center component of the TSM module is a transformer-like network. Based on original Transformer [62], we add a multi-modal fusion sub-layer into network. Three parallel operations are used to promote sufficient cross-modal feature interactions: element-wise multiplication, element-wise
addition, and concatenation followed by FC. The three outputs are concatenated and followed by a FC to change the feature dimension. Note the cross-modal fusion operation is performed word-wise. Through the textual network, we can obtain the energy vector $e_x \in \mathbb{R}^{K \times 1}$ of the masked salient word, which is computed over all categories by a fully-connected layer and softmax function. For each training sample, the training objective loss for the textual network is $-\log(e_x[k])$, where $e_x[k]$ is the output probability of salient category $k$. Detailed structure for our TSM can be accessed in the supplementary material. Once trained, the TSM module is then used to estimate the confidence scores of pseudo-labels as described in Sec. 3.4.

3.4 Joint Semantic Mining for Label Updating

The label updating operation using joint semantic mining is iteratively conducted every $\tau$ epochs (as a training round) during training, i.e., the granularity of label updating. The choices are explored in the ablations of Sec. 4.4, where $\tau = 5$ is shown to work best empirically. In terms of training, the saliency network, the SSM module, and the TSM module are independently trained.

At the end of each training round, our JSM strategy performs the label updating operation to generate an up-to-date pseudo-label for each training image. Now define the pseudo-label for the saliency network in current iteration as $S^t_{\text{map}}$. As shown in the label update phase of Fig. 3, the SSM module is engaged to generate the depth-refined pseudo-label $S^t_D$. It is passed to the TSM module, where the $S^t_{\text{map}}$ and $S^t_D$ are taken as saliency attention maps to filter visual features, respectively. The TSM module infers the energy vectors $e^t_S$ and $e^t_D$ for $S^t_{\text{map}}$ and $S^t_D$, respectively. Thus their confidence scores $SC_t$ and $SC_D$ can be calculated by (taking $SC_t$ as an example):

$$SC_t = \frac{\exp(e^t_S[k])}{\exp(e^t_S[k]) + \exp(e^t_D[k])}.$$  \hspace{1cm} (3)

The updated label is the weighted average of $S^t_{\text{map}}$ and $S^t_D$: $S^{t+1}_{\text{map}} = SC_t \times S^t_{\text{map}} + SC_D \times S^t_D$. A fully-connected conditional random field [23] is applied to generate the final updated label which could provide more trustworthy supervision signal to train the saliency network.

3.5 The CapS Dataset

To train the weakly-supervised RGB-D SOD network, we relabel two widely-used RGB-D saliency datasets, NJUD [29] and NLPR [53], which contain a total of 2,185 training images. We provide various image-level weak annotations: categories, captions, and position of the salient word in each of the captions. The annotation process is conducted semi-automatically: the NeuralTalk2 [30] toolkit is utilized to automatically generate image captions, which are then manually checked, with unreasonable cases corrected. On average, there are 8.9 words in each caption. This is followed by categorically tagging each of the images, each corresponds to a salient category; this is different from the traditional image classification dataset ImageNet [32]. We summarize the 100 categories in the CapS. The positions of the salient words are first automatically identified through localizing the image categories in the captions which is subsequently followed by manual identification. Detailed statistics and examples of our in-house CapS dataset are relegated to the supplementary material.

4 Experiments

4.1 Datasets and Evaluation Metrics

Empirical evaluations are conducted over four large-scale RGB-D SOD benchmark datasets, including NJUD [29] with 1,985 RGB-D paired images, NLPR [53] with 1,000 samples, STERE [50] with 1,000 stereoscopic images, and DUTLF-Depth [55] with 1,200 RGB-D data. In train vs. test splits of the datasets, the popular setup of [18][19][24] is followed for a fair comparison. Training set consists of 1,485 samples from NJUD and 700 samples from NLPR. The remaining images in these datasets and other public test sets are reserved for testing purposes throughout the experiments. Four widely-used metrics are adopted for quantitative evaluation: they are E-measure ($E^c_\alpha$) [16], F-measure ($F_B^γ$) [11], weighted F-measure ($F_W^γ$) [44], and mean absolute error (MAE or $M$) [3].

4.2 Implementation Details and Setups

The code is implemented in Pytorch toolbox on a PC with a single Tesla P40 GPU. We use decoder part [64] with ResNet-50 [22] pre-trained on ImageNet as backbone, for both saliency network and depth network. For each word in the caption, we extract word embedding with dimension $d = 300$ using the pretrained Glove [54] word2vec network. The maximum caption length is set to 20. For
Table 1: Quantitative comparison with weakly-supervised and unsupervised saliency models. Note RGB-based methods are specifically marked by \( \dagger \). \( D_{S} \) and \( T_{S} \) represent the spatial semantic modeling and textual semantic modeling, respectively. ‘Un’ means unsupervised learning. ‘Cls’ is SOD with class label. ‘Cap’ represents using weak supervision signals, with both class label and image caption.

<table>
<thead>
<tr>
<th>*</th>
<th>Sup.</th>
<th>DUTLF-Depth [53]</th>
<th>STERE [53]</th>
<th>NJUD [29]</th>
<th>NLPX [53]</th>
</tr>
</thead>
<tbody>
<tr>
<td>RBD [60]</td>
<td>Un</td>
<td>( E_{2} )</td>
<td>( F_{2} )</td>
<td>( F_{2} )</td>
<td>( M )</td>
</tr>
<tr>
<td>MST [61]</td>
<td>Un</td>
<td>73.2</td>
<td>441</td>
<td>019</td>
<td>222</td>
</tr>
<tr>
<td>BSCA [57]</td>
<td>Un</td>
<td>808</td>
<td>479</td>
<td>682</td>
<td>181</td>
</tr>
<tr>
<td>DSR [60]</td>
<td>Un</td>
<td>797</td>
<td>478</td>
<td>640</td>
<td>164</td>
</tr>
<tr>
<td>ACSD [59]</td>
<td>Un</td>
<td>250</td>
<td>210</td>
<td>188</td>
<td>668</td>
</tr>
<tr>
<td>DES [41]</td>
<td>Un</td>
<td>733</td>
<td>386</td>
<td>668</td>
<td>280</td>
</tr>
<tr>
<td>LHM [53]</td>
<td>Un</td>
<td>767</td>
<td>350</td>
<td>659</td>
<td>174</td>
</tr>
<tr>
<td>GP [60]</td>
<td>Un</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CDB [50]</td>
<td>Un</td>
<td>808</td>
<td>436</td>
<td>713</td>
<td>166</td>
</tr>
<tr>
<td>SE [55]</td>
<td>Un</td>
<td>730</td>
<td>339</td>
<td>474</td>
<td>196</td>
</tr>
<tr>
<td>MB [60]</td>
<td>Un</td>
<td>691</td>
<td>464</td>
<td>577</td>
<td>156</td>
</tr>
<tr>
<td>CDCP [57]</td>
<td>Un</td>
<td>794</td>
<td>530</td>
<td>633</td>
<td>159</td>
</tr>
<tr>
<td>Ours (( D_{S} ))</td>
<td>Un</td>
<td>( \frac{845}{845} )</td>
<td>( \frac{629}{629} )</td>
<td>( \frac{741}{741} )</td>
<td>( \frac{116}{116} )</td>
</tr>
</tbody>
</table>

Image Depth GT Ours MSW CDCP LHM DMC DSR BSCA

Figure 5: Visual comparison of weakly-supervised and unsupervised saliency models.

The transformer-like net in TSM, we set the hidden state to 256, the number of layers to 3, and the number of head to 4. The model is optimized by Adam with batch size of 10, and the learning rate is set to \( 1 \times 10^{-4} \). During training, we use the standard BCE loss to train the saliency network. Each image is uniformly resized to \( 352 \times 352 \) and is performed by randomly rotating and cropping to avoid potential overfitting. Our network is trained in an end-to-end manner and converges around 50 epochs. Initial pseudo labels are generated by the handcrafted method [97], which is freely available for annotations.

### 4.3 Model Performance

We quantitatively evaluate the performance of our approach in Table 1 with visual results shown in Fig. 5. Since our approach is the first work for weakly-supervised RGB-D saliency detection, we show the results of two state-of-the-art RGB-based weakly-supervised methods (WSS [63] and MSW [70]) for reference. To make a fair comparison, we fine-tune them on the same training set using their published code with default setups. These results show the effectiveness of our proposed method. Furthermore, the SSM and TSM in our joint semantic mining framework do not introduce any additional inference cost since they only participate in the training procedure to provide more reliable supervisory signals for saliency network. This design leads to a light-weight network that works both efficiently and effectively. As shown in Table 2 our method runs fastest among RGB-D methods, and also on par with the most efficient RGB-based methods.

### 4.4 Empirical Analysis of Our Pipeline

Here we focus on evaluating the contribution of each component in our pipeline, and examining the internal performance of the pseudo-labels at different stages in training.

Table 2: Inference time of different unsupervised and weakly-supervised saliency models. The RGB-based methods are specifically marked by \( \dagger \).

### Table 1: Quantitative comparison with weakly-supervised and unsupervised saliency models.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Inference time (s)</td>
<td>2.13</td>
<td>2.79</td>
<td>12.98</td>
<td>1.2</td>
<td>1.57</td>
<td>0.718</td>
<td>0.089</td>
<td>0.1002</td>
<td>0.3758</td>
<td>0.1020</td>
<td>0.1025</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
We analyze the evolutions of pseudo-labels in our training process. Analysis of pseudo-labels.

As listed in Table 4, the parameter analysis of the update interval $\tau$ (epoch) in the JSM.

We provide the internal inspections of our approach in Fig. 4, in terms of better estimate the confidence or quality of pseudo-labels and generate the trustworthy supervision signals.

In addition, we further discuss the effect of different update intervals in our pipeline when performing label updating. As listed in Table 4, the larger or smaller update interval leads to inferior performance due to the insufficient or excessive learning of models.

Analysis of pseudo-labels. We analyze the evolutions of pseudo-labels in our training process. Here the quality of pseudo-labels is measured using the ground-truth labels of the training set (for evaluation only). Visual evidences in Fig. 7 show that the quality of pseudo-labels is gradually improved as our JSM is performed. We can see that the initial pseudo-labels unfortunately tend to miss important salient parts as well as fine-grained details. By adopting our proposed joint semantic mining for label updating, the missing parts could be gradually retrieved, with the object silhouette.

We present in Table 3 the ablation results of our pipeline on four benchmarks. To start with, we consider the backbone as the saliency network trained with initial pseudo labels. As our proposed SSM and TSM are gradually incorporated into the backbone to generate the depth-refined pseudo-labels and estimate their confidence scores for label updating, noticeable performance gains are consistently achieved in all datasets. The SSM significantly reduces the MAE metric and increases the F-measure score by 31% and 14.5% on average in four datasets, respectively. The TSM further boosts the saliency detection performance to a higher level where a significant amount of performance gains with 41% and 22.5% are finally achieved on MAE and F-measure metrics.

Ablation analysis. We present in Table 3 the ablation results of our pipeline on four benchmarks. To start with, we consider the backbone as the saliency network trained with initial pseudo labels. As our proposed SSM and TSM are gradually incorporated into the backbone to generate the depth-refined pseudo-labels and estimate their confidence scores for label updating, noticeable performance gains are consistently achieved in all datasets. The SSM significantly reduces the MAE metric and increases the F-measure score by 31% and 14.5% on average in four datasets, respectively. The TSM further boosts the saliency detection performance to a higher level where a significant amount of performance gains with 41% and 22.5% are finally achieved on MAE and F-measure metrics.

To further demonstrate the effectiveness of our SSM in exploiting the depth semantics to refine pseudo-labels, we retrain the saliency network using the self-updating strategy as in Fig. 6 (a). In this strategy, the saliency prediction with CRF are directly utilized to update pseudo-labels, at the end of each training round. As shown in Table 3 (3rd row vs. 4th row), when excluding saliency-oriented depth semantics captured by the SSM, the performance of model degrades greatly. This indicates our SSM can effectively suppress background noise and providing reliable training labels. Furthermore, we replace TSM with a heuristic historical moving average strategy where the pseudo-labels are assigned with equal weights as in Fig. 6 (c). Table 3 shows it achieves better performance than the backbone using SSM due to the consideration of historical information, but it is consistently inferior compared to our pipeline with the TSM module. These results suggest that our TSM can better estimate the confidence or quality of pseudo-labels and generate the trustworthy supervision signals. Meanwhile, we also provide the internal inspections of our approach in Fig. 4, in terms of the generation of pseudo-labels and their corresponding confidence scores, for better understanding.

In addition, we further discuss the effect of different update intervals in our pipeline when performing label updating. As listed in Table 4, the larger or smaller update interval leads to inferior performance due to the insufficient or excessive learning of models.

Table 3: Ablation study of our pipeline. ↑ (↓) denote performance gains (relative to backbone).

<table>
<thead>
<tr>
<th>Model Setups</th>
<th>DUTLF [65]</th>
<th>STERF [59]</th>
<th>NJUD [29]</th>
<th>NLPR [53]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Backbone trained on Pseudo Labels (i.e., $B^+$)</td>
<td>$F_2$</td>
<td>$F_3$</td>
<td>$F_4$</td>
<td>$F_5$</td>
</tr>
<tr>
<td>$B^+$ trained on Pseudo Labels with CRF</td>
<td>.512</td>
<td>.544</td>
<td>.163</td>
<td>.555</td>
</tr>
<tr>
<td>$B^+$ trained on Pseudo Labels with SSM and Historical Moving Average</td>
<td>.568</td>
<td>.676</td>
<td>.140</td>
<td>.603</td>
</tr>
<tr>
<td>$B^+$ with Joint Semantic Mining (i.e., Ours)</td>
<td>.616</td>
<td>.697</td>
<td>.130</td>
<td>.643</td>
</tr>
</tbody>
</table>

$\downarrow (\uparrow)$ denote performance gains (relative to backbone).

Table 4: Parameter analysis of the update interval $\tau$ (epoch) in the JSM.

<table>
<thead>
<tr>
<th>Interval ($\tau$)</th>
<th>STERF [59]</th>
<th>NJUD [29]</th>
<th>NLPR [53]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau = 1$</td>
<td>$F_3$</td>
<td>$F_4$</td>
<td>$F_5$</td>
</tr>
<tr>
<td>$\tau = 5$</td>
<td>$F_7$</td>
<td>$F_8$</td>
<td></td>
</tr>
<tr>
<td>$\tau = 10$</td>
<td>$F_9$</td>
<td>$F_{10}$</td>
<td></td>
</tr>
</tbody>
</table>

We present in Table 3 the ablation results of our pipeline on four benchmarks. To start with, we consider the backbone as the saliency network trained with initial pseudo labels. As our proposed SSM and TSM are gradually incorporated into the backbone to generate the depth-refined pseudo-labels and estimate their confidence scores for label updating, noticeable performance gains are consistently achieved in all datasets. The SSM significantly reduces the MAE metric and increases the F-measure score by 31% and 14.5% on average in four datasets, respectively. The TSM further boosts the saliency detection performance to a higher level where a significant amount of performance gains with 41% and 22.5% are finally achieved on MAE and F-measure metrics.

Ablation analysis. We present in Table 3 the ablation results of our pipeline on four benchmarks. To start with, we consider the backbone as the saliency network trained with initial pseudo labels. As our proposed SSM and TSM are gradually incorporated into the backbone to generate the depth-refined pseudo-labels and estimate their confidence scores for label updating, noticeable performance gains are consistently achieved in all datasets. The SSM significantly reduces the MAE metric and increases the F-measure score by 31% and 14.5% on average in four datasets, respectively. The TSM further boosts the saliency detection performance to a higher level where a significant amount of performance gains with 41% and 22.5% are finally achieved on MAE and F-measure metrics.

To further demonstrate the effectiveness of our SSM in exploiting the depth semantics to refine pseudo-labels, we retrain the saliency network using the self-updating strategy as in Fig. 6 (a). In this strategy, the saliency prediction with CRF are directly utilized to update pseudo-labels, at the end of each training round. As shown in Table 3 (3rd row vs. 4th row), when excluding saliency-oriented depth semantics captured by the SSM, the performance of model degrades greatly. This indicates our SSM can effectively suppress background noise and providing reliable training labels. Furthermore, we replace TSM with a heuristic historical moving average strategy where the pseudo-labels are assigned with equal weights as in Fig. 6 (c). Table 3 shows it achieves better performance than the backbone using SSM due to the consideration of historical information, but it is consistently inferior compared to our pipeline with the TSM module. These results suggest that our TSM can better estimate the confidence or quality of pseudo-labels and generate the trustworthy supervision signals. Meanwhile, we also provide the internal inspections of our approach in Fig. 4, in terms of the generation of pseudo-labels and their corresponding confidence scores, for better understanding.

In addition, we further discuss the effect of different update intervals in our pipeline when performing label updating. As listed in Table 4, the larger or smaller update interval leads to inferior performance due to the insufficient or excessive learning of models.

Analysis of pseudo-labels. We analyze the evolutions of pseudo-labels in our training process. Here the quality of pseudo-labels is measured using the ground-truth labels of the training set (for evaluation only). Visual evidences in Fig. 7 show that the quality of pseudo-labels is gradually improved as our JSM is performed. We can see that the initial pseudo-labels unfortunately tend to miss important salient parts as well as fine-grained details. By adopting our proposed joint semantic mining for label updating, the missing parts could be gradually retrieved, with the object silhouette.
Figure 7: Visualization of updated labels.

Table 5: Ablation analysis of the supervised variant of our approach, where the human annotations (i.e., ground truths) are used to train SOD models.

<table>
<thead>
<tr>
<th>Model Setups (fully)</th>
<th>STERE [50]</th>
<th>NJUD [29]</th>
<th>NLPR [53]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Backbone (i.e., B)</td>
<td>F_{F} F_{F} M</td>
<td>F_{F} F_{F} M</td>
<td>F_{F} F_{F} M</td>
</tr>
<tr>
<td>B + SSM (Ours)</td>
<td>876 896 0.09</td>
<td>885 906 0.03</td>
<td>892 905 0.02</td>
</tr>
</tbody>
</table>

Adapting to unsupervised setting. Our approach can be adapted to unsupervised setting by using the architecture illustrated in Fig. 6 (b), i.e., SSM, where only the depth semantics are mined to refine pseudo-labels without weak labels. The quantitative results in Table [1] show the effectiveness of our SSM in unsupervised setting.

Adapting to fully-supervised setting. Apart from the adaption to unsupervised setting, a variant of our approach can also be applied to fully-supervised RGB-D SOD scenario. This is achieved by modifying the generation of saliency-guided depth mask as \( D_{\text{mask}} = S_{GT} \odot D_{\text{map}} \) in Eq. [1] with \( S_{GT} \) being the ground-truth annotations. The saliency network and depth network are trained by \( S_{GT} \) and \( D_{\text{mask}} \), respectively. Then the background noise suppression block in SSM module is applied to obtain the final saliency. As ablated in Table [5] our fully-supervised variant achieves consistent performance improvement compared to the backbone trained on \( S_{GT} \). In addition, our results compare favorably with those of 21 fully-supervised RGB-D saliency models, as quantitatively shown in Table [6] and qualitatively illustrated in Fig. 9. Notice that, different from existing fully-supervised RGB-D SOD methods in designing the complicated cross-modal feature interaction strategy, our method directly exploits the learned depth semantics to promote saliency accuracy, which is simple yet effective and brand new for this field.

5 Failure Cases

Due to the sparsity of weak annotations, the network is usually difficult to identify the fine-grained object boundaries. As depicted in Fig. [10], although these models can effectively detect the salient objects, fine-grained details are still missing. A doable solution is to introduce auxiliary edge constraint during training. For example, the edge detection loss can be employed on the low-level features of the model, which could force the model to produce better features highlighting the object details [76, 41]. The edge maps can be generated by classical Canny operator [4] in an unsupervised manner.
This research was supported by the University of Alberta Start-up Grant, UAHJIC Grants, and NSERC Discovery Grants (No. RGPIN-2019-04575). The authors are grateful to the anonymous reviewers for their valuable suggestions in improving the quality of the paper.

6 Conclusion

To tackle the new problem of weakly-supervised RGB-D salient object detection, we propose in this paper an end-to-end approach based on iterative updates of the internal pseudo-labels. This allows us to leverage depth information in eliminating non-salient background noises and generating reliable depth-refined pseudo-labels. Textual semantics is incorporated in the fill-in-the-blank fashion, which is used to estimate the confidence scores of pseudo-labels. Extensive experiments demonstrate the effectiveness and efficiency of our approach. In addition, our method is very generic and can be easily adapted to fully-supervised and unsupervised paradigms. In these scenarios, our variants also obtain superior performance over the existing state-of-the-art dedicated methods.

Acknowledgement. This research was supported by the University of Alberta Start-up Grant, UAHJIC Grants, and NSERC Discovery Grants (No. RGPIN-2019-04575). The authors are grateful to the anonymous reviewers for their valuable suggestions in improving the quality of the paper.

References


