Large-Field Contextual Feature Learning for Glass Detection

Haiyang Mei, Xin Yang, Letian Yu, Qiang Zhang, Xiaopeng Wei, and Rynson W.H. Lau

Abstract—Glass is very common in our daily life. Existing computer vision systems neglect it and thus may have severe consequences, *e.g.*, a robot may crash into a glass wall. However, sensing the presence of glass is not straightforward. The key challenge is that arbitrary objects/scenes can appear behind the glass. In this paper, we propose an important problem of detecting glass surfaces from a single RGB image. To address this problem, we construct the first large-scale glass detection dataset (GDD) and propose a novel glass detection network, called GDNet-B, which explores abundant contextual cues in a large field-of-view via a novel large-field contextual feature integration (LCFI) module and integrates both high-level and low-level boundary features with a boundary feature enhancement (BFE) module. Extensive experiments demonstrate that our GDNet-B achieves satisfying glass detection results on the images within and beyond the GDD testing set. We further validate the effectiveness and generalization capability of our proposed GDNet-B by applying it to other vision tasks, including mirror segmentation and salient object detection. Finally, we show the potential applications of glass detection and discuss possible future research directions.

Index Terms—Glass detection, transparent surface, large-field contextual features, boundary cue.

✦

1 INTRODUCTION

GLASS is a non-crystalline, often transparent, amorphous solid that has widespread practical and decorative usages, *e.g.*, window panes, glass doors, and glass walls. Such glass regions can have a critical impact to the existing vision systems (*e.g.*, depth prediction and instance segmentation) as demonstrated in Figure 1. They may further affect intelligent decisions in many applications, such as robotic navigation and drone tracking (*i.e.*, the robot/drone might crash into the glass). Hence, it is essential for vision systems to be able to detect and segment glass from input images.

Some small glass-made objects, such as cup and wine glass, can be detected well by existing methods, *e.g.*, [2], [4], [5], [6], as these glass objects exhibit certain fixed patterns. However, automatically detecting glass regions from images like the ones shown in Figure 1(a) is an extremely challenging task. Due to the fact that a glass region does not have a fixed pattern, *i.e.*, arbitrary objects/scene can appear behind the glass. This makes the glass fundamentally different from other common objects that have been well-addressed by the state-of-the-art segmentation methods [2]. Meanwhile, directly applying existing salient object detection methods [7], [8] to detect glass is inappropriate, as not all glass regions are salient. Besides, the recent mirror segmentation method [9] may segment mirrors by detecting the content contrast between inside and outside the mirror. However, as the content presented by the transparent glass is mainly inherited from the background rather than totally reflecting the scene

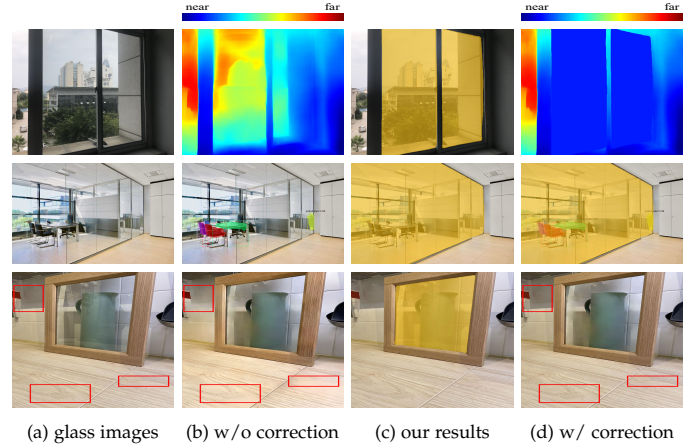(a) glass images   (b) w/o correction   (c) our results   (d) w/ correction

Fig. 1. Problems with glass in existing vision tasks. In depth prediction, existing method [1] wrongly predicts the depth of the scene behind the glass, instead of the depth of the glass (1*st* row of (b)). For instance segmentation, Mask RCNN [2] only segments the instances behind the glass, not aware that they are actually behind the glass (2*nd* row of (b)). Besides, if we directly apply an existing singe-image reflection removal (SIRR) method [3] to an image that is only partially covered by glass, the non-glass region can be corrupted (3*rd* row of (b)). Our method can detect the glass (c) and then help correct these failure cases (d).

in front of it like a mirror, the appearance of the glass is typically similar with the image background, making the glass detection problem more difficult.

To address the glass detection problem, a straightforward solution is to apply a reflection/boundary detector for glass detection. Unfortunately, this may fail if the glass has only weak/partial reflections or ambiguous boundary due in some complex scene, *e.g.*, the second image in Figure 1(a). In general, humans can identify the existence of glass well. We observe that humans typically would combine different contextual information to infer whether and where glass exists. These contexts not only include low-level cues

- *Haiyang Mei, Xin Yang, Letian Yu, Qiang Zhang, and Xiaopeng Wei are with the Faculty of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian, 116024, China. E-mail: mhy666@mail.dlut.edu.cn; xinyang@dlut.edu.cn; letianyu@mail.dlut.edu.cn; zhangq@dlut.edu.cn; xpwei@dlut.edu.cn.*
- *Rynson W.H. Lau is with the Department of Computer Science, City University of Hong Kong. E-mail: Rynson.Lau@cityu.edu.hk.*
- *Project homepage: https://mhaiyang.github.io/.*

Corresponding authors: Xin Yang, Xiaopeng Wei, and Rynson W.H. Lau.

(*e.g.*, the color difference between inside and outside of the glass, blur/bright spot/ghost caused by reflection), but also high-level contexts (*e.g.*, relations between different objects). This inspires us to leverage abundant contextual features for glass detection.

In this paper, we address the glass detection problem from two aspects. First, we construct a large-scale glass detection dataset (GDD), which consists of 3,916 high-quality images with glass and corresponding glass masks, covering various daily-life scenes. Second, we propose a novel glass detection network (GDNet-B), in which multiple well-designed large-field contextual feature integration (LCFI) modules and two boundary feature enhancement (BFE) modules are embedded to harvest abundant low-level as well as high-level contexts from a large receptive field and to integrate both high-level and low-level boundary information, respectively, for accurately detecting glass of different sizes in various scenes.

To sum up, our contributions are as follows:

- We contribute the first large-scale glass detection dataset (GDD) with glass images in diverse scenes and corresponding manually labeled glass masks.
- We propose a novel glass detection network (GDNet-B) with a well-designed large-field contextual feature integration (LCFI) module and a boundary feature enhancement (BFE) module for accurate glass detection, by exploring abundant contextual cues from a large receptive field and integrating boundary information.
- We achieve superior glass detection results on the GDD testing set, by comparing with state-of-the-art models fine-tuned for glass detection.

A preliminary version of this work was presented at CVPR 2020 [10]. This submission extends the work in the following four aspects:

- We add a summary of existing representative works on discriminative context learning (Section 2).
- We introduce the boundary cue into the glass detection process via embedding a well-designed boundary feature enhancement (BFE) module into the original GDNet [10], making our boundary-aware model (termed as GDNet-B) achieve a new state-of-the-art glass detection performance.
- We demonstrate the effectiveness and generalization capability of our proposed GDNet-B by applying it to mirror segmentation (Section 6) and salient object detection (Section 7) tasks.
- We analyze the limitations of our method, show the potential applications of glass detection, and discuss possible future research directions (Section 8).

2 RELATED WORK

In this section, we first briefly review state-of-the-art methods from relevant fields, including semantic/scene/instance segmentation, salient object detection, specific region detection/segmentation, and single image reflection removal. Then, we discuss some representative works on discriminative context learning.

Semantic/scene/instance segmentation. Semantic segmentation aims to segment and parse a given image into different regions associated with semantic categories of discrete objects. Scene segmentation further considers stuff when assigning a label for each pixel. Recently, great progress has been achieved benefited by the advances of deep neural networks. Based on fully convolutional networks (FCNs) [11], state-of-the-art models typically leverage multi-scale context aggregation or exploit more discriminative contexts to achieve high segmentation performance. For example, Chen *et al.* [12] introduce an atrous spatial pyramid pooling (ASPP) to capture multi-scale context information. Zhao *et al.* [13] employ a pyramid pooling module to aggregate local and global context. Ding *et al.* [14] explore contextual contrasted features to boost the segmentation performance of small objects. Zhang *et al.* [15] introduce a channel attention mechanism to capture global context. Fu *et al.* [16] leverage channel- and spatial-wise non-local attention modules to capture contextual features with long-range dependencies. Huang *et al.* [17] further propose a criss-cross attention module to efficiently capture information from long-range dependencies.

Instance segmentation aims to differentiate individual instances of the each category. A typical method is Mask-RCNN [2], which adds a branch of the object detection network Faster-RCNN [18] and achieves good results. PANet [19] further adds bottom-up paths to aggregates multi-level features for detection and segmentation.

However, applying the above segmentation approaches for glass detection (*i.e.*, treating glass as one of the object categories) may not be appropriate as arbitrary objects/scenes can appear behind glass, making glass fundamentally different from other objects. In this paper, we focus on the glass detection problem and formulate it as a binary classification problem (*i.e.*, glass or non-glass).

Salient object detection (SOD). A lot of image-based SOD methods have been proposed in the past decades. Early methods are mainly based on low-level hand-crafted visual features as well as heuristic saliency priors (*e.g.*, color [20] and contrast [21], [22]). These features, however, have limited capability to help distinguish between salient and non-salient objects. Recently, many state-of-the-art deep models are proposed. They utilize the integration of different levels of features to enhance network performances. Specifically, Liu *et al.* [23] progressively integrate local context information to predict saliency maps. Zhang *et al.* [24] propose a generic framework to integrate multi-level features at different resolutions. Zhang *et al.* [25] introduce an attention guided network to selectively integrate multi-level information in a progressive manner. Zhang *et al.* [26] design a bi-directional message passing module with a gated function to integrate multi-level features. Wang *et al.* [27] integrate high-level and low-level features by performing both top-down and bottom-up saliency inferences in an iterative and cooperative manner.

In general, the content presented by a glass region is from a real scene, which may or may not contain salient objects. Existing SOD methods may not be able to detect the whole glass region well.

Specific region detection/segmentation. We briefly review three binary classification tasks: shadow detection,

water hazard detection, and mirror segmentation.

Shadow detection aims to detect shadows from a single input image. Hu *et al.* [28] address the shadow detection problem by analyzing image context in a direction-aware manner. Zhu *et al.* [29] combine local and global contexts for shadow detection. Zheng *et al.* [30] consider shadow distractions in the shadow detection task. In general, there is an intensity difference between shadow region and non-shadow region, while glass typically does not have such an obvious intensity difference between inside and outside of the glass region, making the glass detection problem more difficult to address.

Water hazard detection is to detect water in puddles and flooded areas, on and off the road, to reduce the risk to autonomous cars. The reflection on the water surface typically is an inverted and disturbed transform of the sky or nearby objects above the water surface. Han *et al.* [31] present a reflection attention unit to match this pattern in the vertical direction. However, reflections on the glass can be generated from arbitrary directions and thus applying this method may not be suitable.

Mirror segmentation is a newly proposed research topic that aims to segment mirror regions from a single RGB image. Yang *et al.* [9] observe that there exists both high-level and low-level discontinuities between inside and outside of the mirror and leverage contextual contrasted features to segment mirrors. As the content presented in the transparent glass region is mainly inherited from the background rather than totally reflecting the scene in front of it like a mirror, the appearance of the glass is typically similar with the image background, leading to the little contrasts in terms of both semantics and low-level features between the glass region and its surroundings. Therefore, utilizing contextual contrasted features to detect glass may not obtain the desired results.

Single image reflection removal (SIRR). Reflection is a frequently-encountered source of image corruption when shooting through a glass surface. Such corruptions can be addressed via a single-image reflection removal (SIRR) task. Traditional SIRR methods employ different priors (*e.g.*, sparsity [32], smoothness [33], [34], and ghost [35]) to exploit the special properties of the transmitted and reflection layers. In recent deep-learning-based methods, edge information [36], [37], perceptual loss [38] and adversarial loss [39] are used to improve the recovered transmitted layer.

SIRR can be seen as an image enhancement problem. It aims to recognize where the reflections are and then remove them to enhance the visibility of the background scene. The ultimate goal of our glass detection problem is not to recognize only the reflections but to detect the whole glass region, which may contain only partial or weak reflections.

Discriminative context learning. Contextual information plays an important role in achieving high performance for many computer vision tasks. Many works are devoted to exploiting more discriminative contexts to enhance the ability of feature representation. For example, Chen *et al.* [12] introduce an atrous spatial pyramid pooling (ASPP) module to capture multi-scale contextual information, which is then adopted by many methods such as [40], [41], [42], [43]. Zhao *et al.* [44] employ a pyramid pooling module to aggregate local and global contexts. Liu *et al.* [45] design a receptive field

block (RFB) to simulate the configuration in terms of the size and eccentricity of the receptive fields in the human visual system via multi-scale contexts perception for enhancing the feature discriminability and robustness. In addition, multi-scale adaptive/dynamic contexts are proposed in [46], [47], [48], multi-level contexts are extracted in [49], [50], [51], [52], [53], switchable contexts are used in [54], large-field contextual features are captured in [55], direction-aware contexts are explored in [56], [57], and contextual contrasted features are leveraged in [9], [58], [59], [60], [61]. Besides, a non-local operation [62] is proposed to capture long-range contextual dependencies, and its variants are applied to various vision tasks, *e.g.*, image segmentation [63], [64], [65], [66] and image restoration [67], [68], [69].

Our method differs from the above works in that we explore rich contextual information in a very large field-of-view. Note that [55] only extracts large-field contexts via spatially separable convolutions at a fixed scale while our proposed LCFI module can explore multi-scale large-field contexts. We validate the effectiveness and generalization capability of the proposed large-field contextual feature learning with experiments.

3 A NEW DATASET FOR GLASS DETECTION

The emergence of new tasks and datasets (*e.g.*, [9], [70], [71], [72], [73]) has led to rapid progress in various areas of computer vision. For example, ImageNet [74] revolutionizes the use of deep models for visual recognition. With this in mind, our goals for studying and developing a new dataset for glass detection are: (i) to propose a new meaningful yet challenging task, (ii) to promote research on this new topic, and (iii) to spark novel ideas. In this paper, we contribute a large-scale glass detection dataset (GDD), which contains 3,916 pairs of glass and glass mask images. To the best of our knowledge, GDD is the first large-scale benchmark specifically for glass detection.

Dataset construction. Our glass images are captured with some latest cameras and smart phones. The pixel-level glass masks (with pixels inside the glass region set to 1 and 0 otherwise) are labeled by professional annotators with a professional labeling tool. The glass mask is checked by three viewers to ensure labeling accuracy. Our constructed glass detection dataset, GDD, covers diverse daily-life scenes (*e.g.*, bathroom, office, street, and mall), in which 2,827 images are taken from indoor scenes and 1,089 images are taken from outdoor scenes. Figure 2 shows some example glass and glass mask images in GDD. For dataset split, 2,980 images are randomly selected for training and the remaining 936 images are used for testing.

Dataset analysis. To validate the diversity of GDD and how challenging it is, we show its statistics as follows:

- **Glass type.** As shown in Figure 3(a), there are various types of common glass in GDD (*e.g.*, shopwindow, glass wall, glass door, glass guardrail, and glass on window and cabinet). Other relatively small glass objects such as glass bulbs and glass clocks are also included. The reason that such glass objects occupy only a small ratio in GDD is that in this work, we aim to detect relatively large transparent glass that could
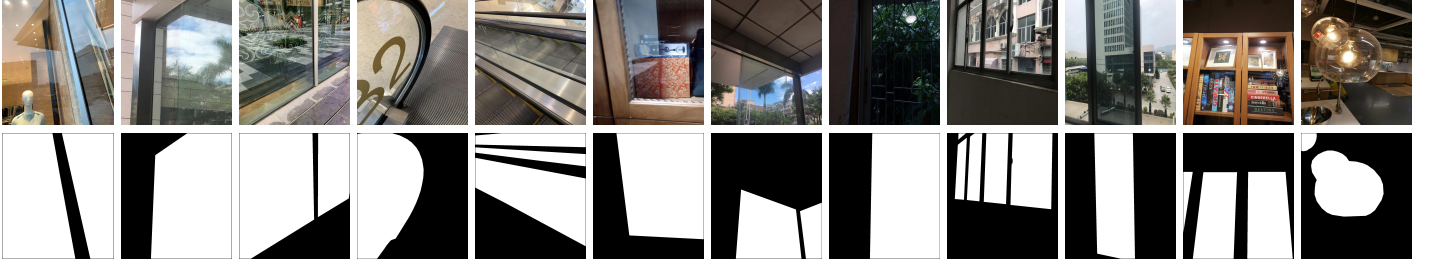
Fig. 2. Example glass image/mask pairs in our glass detection dataset (GDD). It shows that GDD covers diverse types of glass in daily-life scenes.

(a) glass type distribution   (b) glass location distribution   (c) glass area distribution   (d) mirror area distribution
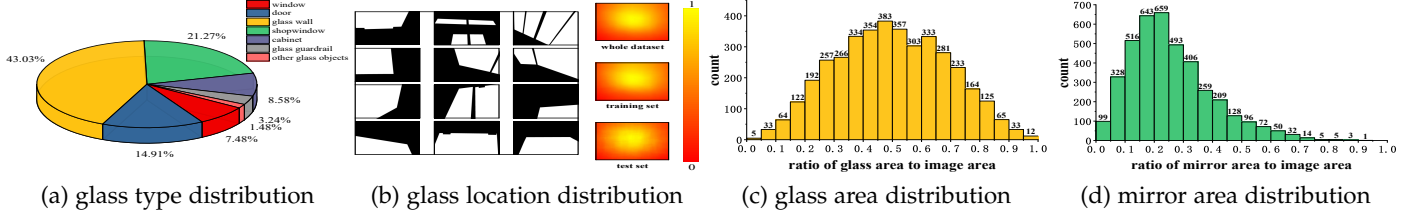
Fig. 3. Statistics of our dataset. We show that GDD has glass with reasonable property distributions in terms of type, location and area.

contribute critical effect to scene understanding. The small glass objects are mainly to add diversity.

- **Glass location.** Our GDD has glass located at different positions of the image, as illustrated in Figure 3(b). We further compute probability maps that indicate how likely each pixel belongs to a glass region, to show the location distributions of glass in GDD. The overall spatial distribution tends to be centered, as glass is typically large and covers the center. Besides, the glass spatial distributions for the training/test splits are consistent to those of the whole dataset.

- **Glass area.** We define the size of the glass region as a proportion of pixels in the image. In Figure 3(c), we can see that the glass in our GDD varies in a wide range in terms of size, and much of it falls in the range of [0.2, 0.8]. Glass falling in the range of (0, 0.2] represents small glass objects or glass corners. Such small glass regions can be easily cluttered with diverse background objects/scenes. Glass falling in the range of (0.8, 1.0) is typically located close to the camera. In this situation, the content of the image is dominated by the complicated scene behind the glass. Extreme cases, *i.e.*, glass area equals to 0 or 1, are not included in GDD. Compared with the distribution of the mirrors in the mirror segmentation dataset (MSD) [9] as shown in Figure 3(d), glass in our GDD typically has a larger area, which means that more objects/scenes would be presented inside the glass, making GDD more challenging.

4 METHODOLOGY

Considering that there may exist only a few or very weak reflections on the glass and the whole glass region has no fixed patterns, *i.e.*, arbitrary objects/scenes can appear behind the glass in real-world scenes, we do not leverage the off-the-shelf physical model to capture reflections nor focus on exploring a specific physical cue to detect the whole glass region. Instead, we observe that humans can identify the

existence of glass well, by considering contextual information in terms of low-level cues (*e.g.*, color difference between inside and outside the glass, blur/bright spot/ghost caused by reflection) as well as high-level contexts (*e.g.*, relations between different objects) and boundary information. This inspires us to leverage both abundant contextual features and boundary features for glass detection.

To this end, first, we propose a novel Large-field Contextual Feature Integration (LCFI) block to extract abundant contextual features from a large field for context inference and glass localization. Second, based on the LCFI block, a novel LCFI module is designed to effectively integrate multi-scale large-field contextual features for detecting glass of different sizes. Third, we embed multiple LCFI modules to the glass detection network (GDNet) to obtain large-field contextual features of different levels. Finally, we apply two boundary feature enhancement (BFE) modules to integrate both high-level and low-level boundary cues for the accurate glass detection under various scenes.

4.1 Network Overview

Figure 4 presents the proposed glass detection network (GDNet-B). It employs the LCFI module (Figure 5) to learn large-field contextual features and the BFE module (Figure 6) to integrate boundary information. Given a single input image, we first feed it into the multi-level feature extractor (MFE) to harvest features of different levels, which are further fed into four proposed LCFI modules to learn large-field contextual features. The outputs of the last three LCFI modules are fused to generate high-level large-field contextual features, which are then used to guide the first LCFI module to focus more on the glass regions as it extracts low-level large-field contextual features. We then apply the well-designed BFE module on both the high-level features and attentive low-level features to perceive and integrate boundary cues. Finally, we fuse high-level and attentive low-level large-field contextual features to produce the final glass detection result.
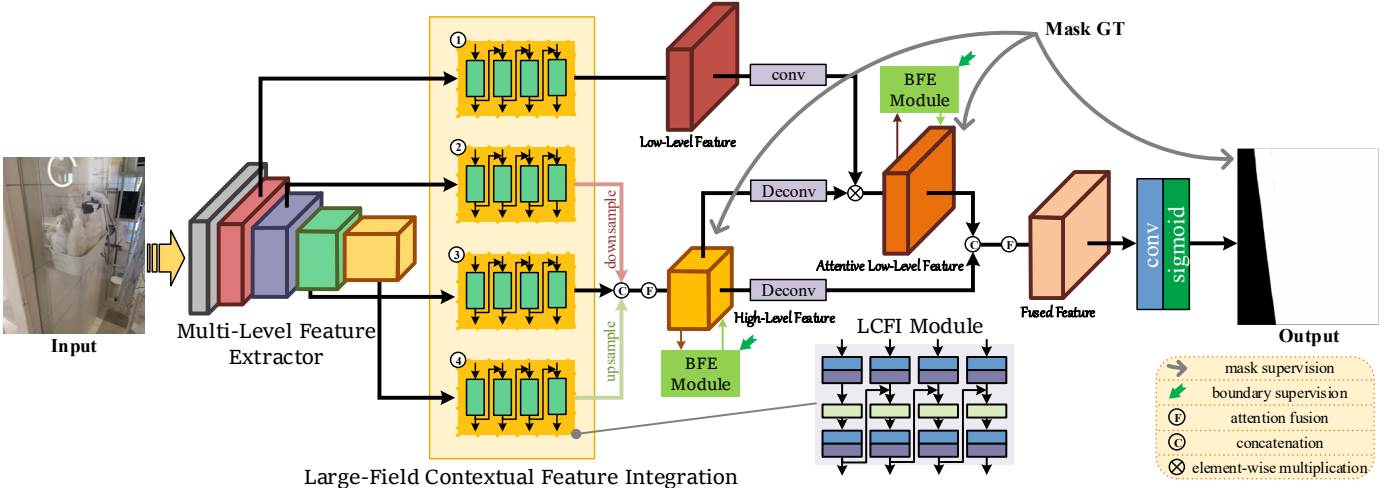
Fig. 4. The pipeline of the proposed GDNet-B. First, we use the pre-trained ResNeXt-101 [75] as a multi-level feature extractor (MFE) to obtain features of different levels. Second, we embed four LCFI modules to the last four layers of MFE, to learn large-field contextual features at different levels. Third, the outputs of the last three LCFI modules are concatenated and fused via an attention module [76] to generate high-level large-field contextual features. An attention map is then learned from these high-level large-field contextual features and used to guide the low-level large-field contextual features, *i.e.*, the output of the first LCFI module, to focus more on glass regions. Fourth, we apply two BFE modules on the high-level/attentive low-level large-field contextual features to further perceive and integrate boundary cues. Finally, we combine high-level and attentive low-level large-field contextual features by concatenation and attention [76] operations to produce the final glass detection map.
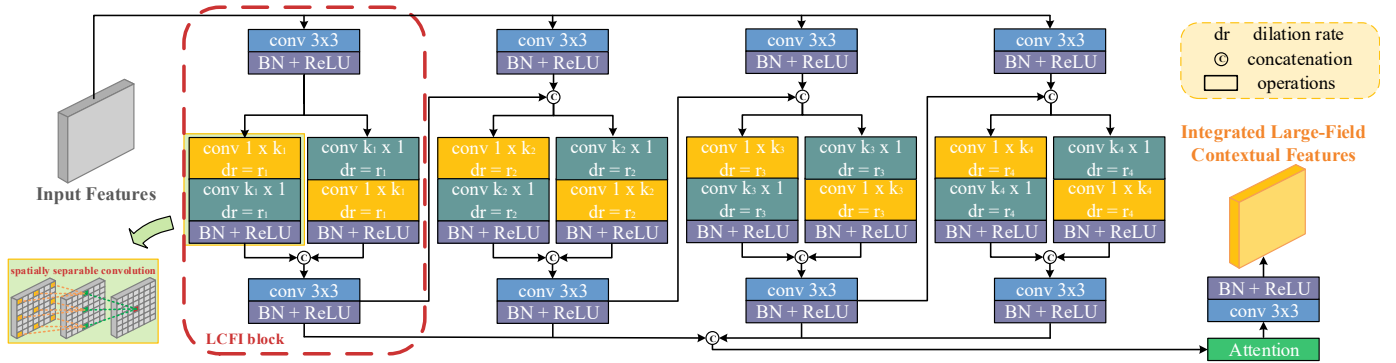

Fig. 5. The structure of the proposed LCFI module. The input features are passed through four parallel LCFI blocks, and the outputs of all LCFI blocks are fused to generate multi-scale large-field contextual features. In each LCFI block (red dashed box), input features are fed to two parallel spatially separable convolutions with opposite convolution orders to obtain large-field contextual features with different characteristics. The output of the current LCFI block is then fed to the next LCFI block to be further processed in a larger field.

4.2 Large-Field Contextual Feature Integration

Figure 5 illustrates the structure of our LCFI module. Given the input features, the LCFI module aims to efficiently and effectively extract and integrate multi-scale large-field contextual features, for the purpose of detecting glass of different sizes.

LCFI block. The LCFI is designed to efficiently extract abundant contextual information from a large field for context inference and glass localization. The common practice to obtain larger context information is to use convolutions with large kernels or dilated convolutions. However, large kernels would result in heavy computation and a large dilation rate would lead to sparse sampling. Non-local operations [77] could provide long-range dependencies but also suffer from huge computation. Here, we propose to use spatially separable convolutions. Spatially separable convolutions are convolutions that can be separated across

their spatial axes and enable densely connections within a large region in the feature map while at the same time using fewer parameters than the traditional convolutions. Thus, we propose to use them to achieve the goal of efficiently extracting abundant contexts from a large field:

$$F_c = \aleph(conv_h(conv_v(F))), \qquad (1)$$

where F denotes the input features. $conv_v$ and $conv_h$ refer to vertical convolution with a kernel size of $k \times 1$ and horizontal convolution with a kernel size of $1 \times k$, respectively. $\aleph$ represents batch normalization and ReLU operations. F_c denotes the extracted large-field contextual features.

As the content inside a glass region is typically complicated, contextual features with different characteristics are needed to eliminate the ambiguity. Thus, we use another spatially separable convolution with reverse convolution order, *i.e.*, $\aleph(conv_v(conv_h(F)))$, to extract complementary

large-field contextual features. As the contextual cues in the vertical and horizontal directions are usually not exactly the same, the intermediate features extracted by the first vertical/horizontal convolutions in the two paths would have different characteristics. Thus, the final features of the two paths are different and may be complementary as they are further learned by convolutions with orthogonal directions. Besides, we adopt dilated spatially separable filters to ensure that more contexts can be explored in a larger field. Finally, the large-field contextual features extracted from the two parallel paths are fused by a 3×3 convolution followed by BN and ReLU. The tasks of the LCFI block can be formulated as:

$$F_{lcfi} = \aleph(conv_2(concat(\aleph(conv_v(conv_h(F_l))),$$
$$\aleph(conv_h(conv_v(F_l)))))), \quad (2)$$
$$F_l = \aleph(conv_1(F_{in})),$$

where F_{in} denotes the input features of the LCFI block and F_{lcfi} denotes the integrated large-field contextual features. $conv_1$ and $conv_2$ are local convolutions with a 3×3 kernel.

LCFI module. Glass captured in an image can vary in size (Figure 3(a)). Given the kernel size k and dilation rate r, the LCFI block extracts contextual features from a large field of a fixed size. On the one hand, if this field is not large enough to cover the whole glass region, incomplete detection may occur. On the other hand, if this field is too large for a small glass region, too much noise would be introduced and cause a false positive detection. To address this problem, contexts of different scales should be considered. Hence, based on the LCFI block, we propose a LCFI module to harvest contextual features from large fields of different scales. Specifically, we feed the input features into four parallel LCFI blocks and fuse their outputs using an attention module [76]. To further explore more contextual features, we add information flow between adjacent LCFI blocks, *i.e.*, we feed the output of a current LCFI block to the next LCFI block. By doing so, local features F_l^i and large-field contextual features from the previous block F_{lcfi}^{i-1} are combined and further processed by the current LCFI block. For the spatially separable convolutions used in the four LCFI blocks, the kernel size k is set to 3, 5, 7, 9, and the dilation rate dr is set to 1, 2, 3, 4, respectively.

Although we draw inspiration (*i.e.*, adding information flow between different paths/blocks) from the integrated successive dilation (ISD) module in [78] in our module design, the proposed LCFI module is inherently different from ISD in both motivation and implementation. The ISD module aims to extract invariant features for the salient object embedded in various contexts while our LCFI module is designed to locate glass of different sizes by exploring contextual information from a large field of different scales. Besides, ISD uses 3×3 convolutions with a large dilation rate (*e.g.*, r=16) to capture large-field contexts. We argue that the contexts extracted in this way are insufficient for complete glass detection (see "base + LCFI w/ local" in Figure 7). Instead, in each LCFI block, we leverage spatially separable convolutions to explore abundant contextual cues from a large field (see "GDNet" in Figure 7).
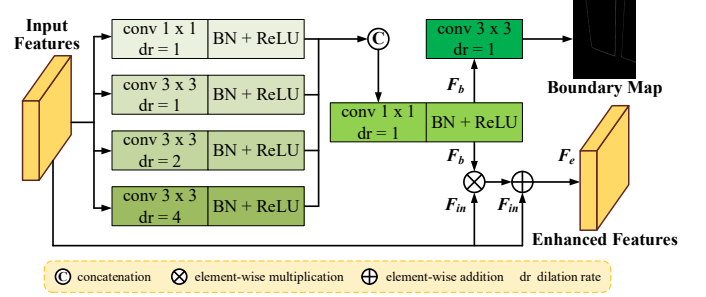

Fig. 6. The structure of the proposed BFE module. The input features are passed through four parallel branches, and the outputs of all branches are fused to generate glass boundary features, which are then used to predict the glass boundary map and complement the input features.

4.3 Boundary Feature Enhancement

Since the boundary is a strong cue for humans to distinguish a glass region, we further introduce the boundary cue to help boost the glass detection accuracy. Integrating boundary information can help glass detection in both segmentation and localization. Based on this idea, we propose a boundary feature enhancement (BFE) module to learn and integrate glass boundary features.

Figure 6 illustrates the structure of the proposed BFE module. Given the input features F_{in}, we first feed them into a multi-scale boundary feature extractor, which contains four parallel branches with different receptive fields (1, 3, 5, 9) and each branch consists of a convolution layer, a batch normalization (BN) layer and a ReLU nonlinearity operation. We then concatenate the outputs of all four branches and fuse them via a 1×1 convolution followed by BN and ReLU, to obtain the glass boundary features F_b. Finally, we extract the boundary-enhanced features F_e by aggregating the boundary features F_b and the input features F_{in} as:

$$F_e = F_b \otimes F_{in} \oplus F_{in}, \quad (3)$$

where $\otimes$ denotes element-wise multiplication and $\oplus$ denotes element-wise addition. We employ a 3×3 convolution on the glass boundary features F_b to predict the glass boundary map, which is supervised by the glass boundary ground truth generated by computing the gradients on the glass mask ground truth, to encourage F_b to obtain discriminative boundary features. Benefiting from the rich location information and boundary information in the glass boundary features F_b, the boundary-enhanced features F_e can help the network locate glass regions, especially their boundaries more accurately. As shown in Figure 4, we apply the BFE module on both the high-level features and attentive low-level features to harvest both high-level and low-level boundary-enhanced features, leading to a boundary-aware model.

Note that our method as well as [79], [80] explore the boundary cues for the specific task. Our BFE module differs from the method in [79] in both the generation and use of boundary features. [79] uses an additional boundary stream (*i.e.*, an encoder-decoder branch) to obtain the boundary features and then fuse these boundary features with different levels of main stream features. Our BFE module can generate the boundary features from the given input

features that needs to be enhanced. As different levels of features typically require different levels of boundary features (referred to as "HL" and "LL" maps in Figure 9, for an example), our BFE would be more effective and flexible. [80] only conducts the boundary feature enhancement on the lowest level of features while we apply our BFE module on both the high-level and low-level features to harvest both high-level and low-level boundary-enhanced features, leading to an omni-boundary-aware model.

4.4 Loss Function

For optimizing GDNet-B, we adopt three types of losses, *i.e.*, binary cross-entropy (BCE) loss ℓ_{bce} [81], IoU loss ℓ_{iou} [8] and edge loss ℓ_{edge} [82], during the training process. Specifically, for high-level large-field contextual features, we combine BCE loss and IoU loss, *i.e.*, $\mathcal{L}_h = \ell_{bce} + \ell_{iou}$, to force them to explore high-level cues for complete glass detection. For attentive low-level large-field contextual features, we want them to provide low-level cues for predicting glass maps with clear boundaries. Thus, we combine BCE loss and edge loss, *i.e.*, $\mathcal{L}_l = \ell_{bce} + \ell_{edge}$. The edge loss would implicitly help the network find glass boundaries. For the final output, the complete detection with clear glass boundary is desired. Hence, we combine BCE loss, IoU loss and edge loss, *i.e.*, $\mathcal{L}_f = \ell_{bce} + \ell_{iou} + \ell_{edge}$. We further use BCE loss to supervise the glass boundary map predicted by the BFE module. Finally, the overall loss function is:

$$\mathcal{L}_{overall} = w_h \mathcal{L}_h + w_l \mathcal{L}_l + w_f \mathcal{L}_f + w_h^b \mathcal{L}_h^b + w_l^b \mathcal{L}_l^b, \quad (4)$$

where w_h, w_l, and w_f represent the balancing parameters for $\mathcal{L}_h$, $\mathcal{L}_l$, and $\mathcal{L}_f$, respectively. $\mathcal{L}_h^b$ and $\mathcal{L}_l^b$ denote the losses between the glass boundary ground truth and the glass boundary prediction of the BFE module applied on the high-level features and attentive low-level features, respectively. And w_h^b and w_l^b represent the balancing parameters for $\mathcal{L}_h^b$ and $\mathcal{L}_l^b$.

5 EXPERIMENTS ON GLASS DETECTION

5.1 Experimental Settings

Implementation details. We have implemented GDNet-B on the PyTorch framework [83]. A PC with an Intel eight-core i7-7700K 4.2 GHz CPU, 32GB RAM and an NVIDIA GTX 1080Ti GPU (with 11GB memory) is used for both training and testing.

For training, input images are resized to a resolution of 416×416 and are augmented by horizontally random flipping. The parameters of the multi-level feature extractor are initialized by the pre-trained ResNeXt-101 network [75] and the other parameters are initialized randomly. Stochastic gradient descent (SGD) with a momentum of 0.9 and weight decay of 5×10^{-4} is used to optimize the network. We adjust the learning rate by the poly strategy [84] with a basic learning rate of 0.001 and a power of 0.9. The batch size is set to 6 and the balancing parameters w_h, w_l, w_f, w_h^b, and w_l^b are empirically set to 1, 1, 1, 10, and 50, respectively. Training GDNet-B takes about 22 hours for 190 epochs.

For testing, the input image is first resized to the resolution of 416×416 for network inference, and the output mask is then resized back to the original size of the input image.

Both resizing processes use bilinear interpolation. There is no post-processing (such as the fully connected conditional random field (CRF) [85]) needed to further enhance the final output. The average inference time of each image in GDD testing set is 0.034 seconds.

Evaluation metrics. For a comprehensive evaluation, we adopt five widely used metrics for quantitatively evaluating the glass detection performance: intersection over union (IoU), pixel accuracy (PA), maximum F-measure (F_β^m), mean absolute error (M), and balance error rate (BER).

The intersection of union (IoU) and pixel accuracy (PA) are widely used in the semantic segmentation field, which are defined as:

$$IoU = \frac{\sum\limits_{i=1}^{H} \sum\limits_{j=1}^{W} (G(i,j) * P_b(i,j))}{\sum\limits_{i=1}^{H} \sum\limits_{j=1}^{W} (G(i,j) + P_b(i,j) - G(i,j) * P_b(i,j))}, \quad (5)$$

$$PA = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} \mathbf{1}(G(i,j) = P_b(i,j)), \quad (6)$$

where G and P_b are the ground truth mask and the predicted mask binarized with a threshold of 0.5, respectively. H and W are the height and width of the predicted mask. $\mathbf{1}(\cdot)$ is the indicator function.

We also adopt the F-measure and mean absolute error metrics from the salient object detection field. F-measure is a harmonic mean of precision and recall, defined as:

$$F_\beta = \frac{(1 + \beta^2) Precision \times Recall}{\beta^2 Precision + Recall}, \quad (7)$$

where β^2 is set to be 0.3 to emphasize more on precision over recall, as suggested in [22]. The precision and recall of a detection map are computed by comparing the binarized predicted mask against the ground truth mask. Under a specific binarizing threshold, we can obtain a pair of average precision and recall over all binarized saliency maps in a dataset. By varying the threshold from 0 to 1 with a step of $\frac{1}{255}$, we obtain 256 precision-recall pairs and 256 F_β scores. We report the maximum F_β (F_β^m) in the comparison of glass detection performance.

Given the predicted mask P and the ground truth mask G, the mean absolute error (M) can be calculated by the element-wise difference between P and G,

$$M = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} |P(i,j) - G(i,j)|, \quad (8)$$

where $P(i,j)$ indicates the probability score at location (i,j) in the predicted mask and H and W are the map's height and width, respectively.

The last metric is the balance error rate (BER), which is a standard metric in the shadow detection field, defined as:

$$BER = (1 - \frac{1}{2}(\frac{TP}{N_p} + \frac{TN}{N_n})) \times 100, \quad (9)$$

where TP, TN, N_p, and N_n represent the numbers of true positive pixels, true negative pixels, glass pixels, and non-glass pixels, respectively.

Note that for IoU, PA and F_β^m, it is the higher the better, while for M and BER, it is the lower the better.

TABLE 1
Quantitative comparison to the state-of-the-art methods on the GDD testing set. All methods are re-trained on the GDD training set. * denotes using CRFs [85] for post-processing. "Statistics" means thresholding glass location statistics from our training set as a glass mask for detection. The first, second, and third best results are marked in **red**, **green**, and **blue**, respectively. Our method achieves the state-of-the-art performances on all five evaluation metrics.

Methods	IoU↑	PA↑	F_β^m↑	M↓	BER↓
Statistics	40.75	0.584	0.564	0.451	39.31
ICNet [86]	69.59	0.836	0.821	0.164	16.10
PSPNet [13]	84.06	0.916	0.906	0.084	8.79
DenseASPP [87]	83.68	0.919	0.911	0.081	8.66
BiSeNet [88]	80.00	0.894	0.883	0.106	11.04
PSANet [89]	83.52	0.918	0.909	0.082	9.09
DANet [16]	84.15	0.911	0.901	0.089	8.96
CCNet [17]	84.29	0.915	0.904	0.085	8.63
DSS [90]	80.24	0.898	0.890	0.123	9.73
PiCANet [7]	83.73	0.916	0.909	0.093	8.26
RAS [91]	80.96	0.902	0.895	0.106	9.48
R^{3}Net* [92]	76.71	0.869	0.869	0.132	13.85
CPD [93]	82.52	0.907	0.903	0.095	8.87
PoolNet [94]	81.92	0.907	0.900	0.100	8.96
BASNet [8]	82.88	0.907	0.896	0.094	8.70
EGNet [95]	85.04	0.920	**0.916**	0.083	7.43
DSC [28]	83.56	0.914	0.911	0.090	7.97
BDRAR* [29]	80.01	0.902	0.908	0.098	9.87
MirrorNet* [9]	**85.07**	0.918	0.903	0.083	7.67
TransLab [79]	81.64	0.903	0.892	0.097	9.70
Trans2Seg [66]	84.41	**0.922**	0.905	**0.078**	**7.36**
GDNet [10]	**87.63**	**0.939**	**0.937**	**0.063**	**5.62**
GDNet-B (ours)	**87.83**	**0.941**	**0.939**	**0.061**	**5.52**

TABLE 2
Ablation studies on the effectiveness of the LCFI module. "Base" denotes GDNet [10] with all LCFI modules removed. "one scale" and "two scales" denote that there are one and two LCFI blocks in the LCFI module. "local" denotes replacing the spatially separable convolutions in LCFI with local convolutions and keeping the parameters approximately the same. Based on "local", "sparse" adopts dilated convolutions to achieve a similar receptive field as spatially separable convolutions. "one path" denotes that there is only one spatially separable convolution path in each LCFI block. Our LCFI module contains four LCFI blocks and each of them contains two parallel paths.

Networks	IoU↑	F_β^m↑	BER↓
Base	84.89	0.923	7.40
Base + LCFI w/ one scale	86.22	0.931	6.51
Base + LCFI w/ two scales	86.78	0.932	6.34
Base + LCFI w/ local	86.93	0.932	6.36
Base + LCFI w/ sparse	87.13	0.933	5.88
Base + LCFI w/ one path	87.31	0.935	5.81
Base + LCFI (*i.e.*, GDNet [10])	**87.63**	**0.937**	**5.62**

| Image | Base | Base + LCFI w/ local | Base + LCFI w/ sparse | Base + LCFI |

Fig. 7. Ablation analysis on the effectiveness of the LCFI module.

5.2 Comparison with the State-of-the-arts

Compared methods. As a first attempt to detect glass from a single RGB image, we validate the effectiveness of our GDNet by comparing it with 18 methods selected from other related fields according to the following criteria: (i) classical architectures, (ii) recently published, and (iii) achieving state-of-the-art performance in a specific field. Specifically, we choose ICNet [86], PSPNet [13], DenseASPP [87], BiSeNet [88], PSANet [89], DANet [16] and CCNet [17] from the semantic segmentation field; DSS [90], PiCANet [7], RAS [91], R^{3}Net [92], CPD [93], PoolNet [94], BASNet [8] and EGNet [95] from the salient object detection field; DSC [28] and BDRAR [29] from the shadow detection field; MirrorNet [9] from the mirror segmentation field; and TransLab [79], Trans2Seg [66] and GDNet [10] from the glass segmentation field. For a fair comparison, we use either their publicly available codes or the implementations with recommended parameter settings. All models are retrained on the GDD training set.

Evaluation on the GDD testing set. Table 1 reports the quantitative results of glass detection on the proposed GDD testing set. We can see that our method outperforms all the other state-of-the-art methods on all five metrics. Figure 8 shows the qualitative comparison of our method with the state-of-the-art methods. We can see that our method is capable of accurately detecting small glass regions (*e.g.*, the 1*st* row), large glass regions (*e.g.*, the 2*nd* row), and multiple glass regions (*e.g.*, the 3*rd* row). This is mainly because multi-scale contextual features extracted by the LCFI module can help the network locate and segment

glass better. While the state-of-the-arts are typically confused by the non-glass regions which share similar boundaries/appearances with the glass regions, our method can successfully eliminate such ambiguities and detect only the real glass regions (*e.g.*, 4-6*th* rows). This is mainly contributed by the proposed large-field contextual feature learning, which provides abundant contextual information for context inference and glass localization. Benefited by the glass boundary feature enhancement, our GDNet-B achieve better glass detection results than GDNet [10] (*e.g.*, the last four rows).

5.3 Component Analysis

We conduct ablation studies on GDD to validate the effectiveness of two key components tailored for accurate glass detection (*i.e.*, the LCFI module and the BFE module) as well as the impact of different multi-level feature extractors, and report the comparison results in Table 2, Table 3, Table 4, Figure 7, and Figure 9.

Effectiveness of the LCFI module. In Table 2, by comparing the results of "Base + LCFI w/ one scale", "Base + LCFI w/ two scales" and "Base + LCFI" (also can be expressed as "Base + LCFI w/ four scales"), we can see that multi-scale convolutions can help improve the detection performance. In addition, using dilated convolution in the LCFI module (*i.e.*, "Base + LCFI w/ sparse") performs better than using local one (*i.e.*, "Base + LCFI w/ local"), as contexts can be explored from a larger receptive field. With approximately the same number of parameters, using spatially separable convolutions (*i.e.*, "Base + LCFI w/ one
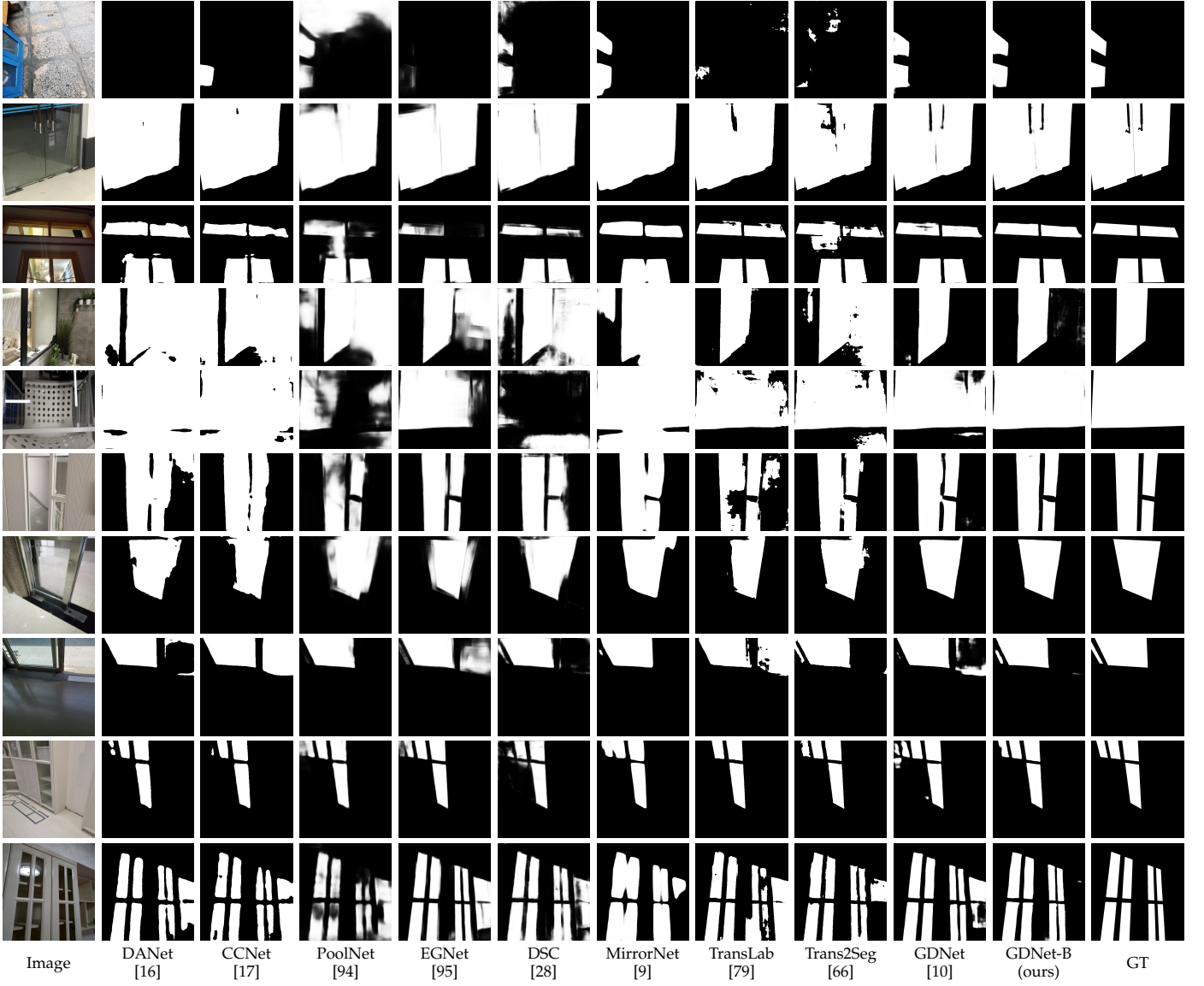
Fig. 8. Visual comparison of GDNet-B to the state-of-the-art methods on the proposed GDD testing set.

path") can harvest more contexts from a large field to further boost the performance. Finally, large-field contextual features with different characteristics can be obtained by two parallel spatially separable convolution paths and help "Base + LCFI" achieve more accurate glass detection results. Figure 7 shows a visual example. We can see that "Base + LCFI" successfully addresses the under-segmentation problem with the help of abundant contextual features extracted from the large field.

Effectiveness of the BFE module. By comparing the results of "GDNet-B w/o BFE" and "GDNet-B" in Table 3, we can see that BFE can help improve the glass detection performance. Figure 9 shows a visual comparison example where we can see that the BFE module can help the network detect the complete glass region (the top row) and generate glass detection result with more accurate glass boundary (the bottom row). This clearly demonstrates the effectiveness of our BFE module.

Impact of different multi-level feature extractors. We further explore the impact of different multi-level feature ex-

TABLE 3
Ablation studies on the effectiveness of the BFE module.

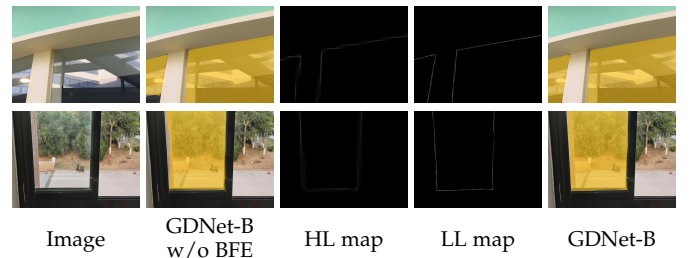| Networks	IoU↑	F_β^m↑	BER↓
GDNet-B w/o BFE	87.63	0.937	5.62
GDNet-B	**87.83**	**0.939**	**5.52**

Fig. 9. Ablation analysis on the effectiveness of the BFE module. "HL map" and "LL map" are the output boundary maps of the BFE module applied on the high-level features and attentive low-level features in GDNet-B, respectively.

Fig. 10. More glass detection results of our GDNet-B on images beyond the GDD testing set.

TABLE 4
Ablation studies on the impact of different multi-level feature extractors.

Multi-Level Feature Extractors	IoU↑	F_β^m↑	BER↓
GDNet-B w/ ResNet-101 [96]	86.84	0.934	5.95
GDNet-B w/ ResNeXt-101 [75]	87.83	**0.939**	5.52
GDNet-B w/ SENet-154 [97]	**88.09**	**0.939**	**5.48**

tractors in GDNet-B for glass detection via adopting different backbone architectures. In our experiment, three popular architectures are used (*i.e.*, ResNet-101 [96], ResNeXt-101 [75], and SENet-154 [97]) and the weights of these backbones are initialized with the corresponding pre-trained models on ImageNet. From Table 4, we can see that the performance of the multi-level feature extractor has a direct impact on the glass detection performance.

5.4 More Glass Detection Results

Figure 10 further shows some glass detection results of GDNet-B on images beyond the GDD testing set, *i.e.*, images selected from the ADE20K dataset [72] (the 1*st* column), Matterport3D dataset [98] (the 2*nd* column), and 2D3DS [99] dataset (the 3*rd* column), and images downloaded from the Internet (4-12*th* columns). We can see that GDNet-B performs well under these various cases, demonstrating its effectiveness.

6 EXPERIMENTS ON MIRROR SEGMENTATION

With the help of the well-designed large-field contextual feature integration module, our GDNet-B can explore abundant contextual information from a large field-of-view, and thus has the potential to handle other challenging vision tasks. In this section, we consider the mirror segmentation task. It has been pointed out by previous work [9] that there typically exists high-level and low-level contextual contrast between inside and outside the mirror. As our method can perceive rich contextual cues, we believe that the contextual contrast could be modeled implicitly by our GDNet-B in the end-to-end training process based on the mirror segmentation dataset.

6.1 Experimental Settings

Benchmark datasets. We conduct the experiments on the benchmark mirror segmentation dataset (MSD) [9]. MSD [9] is the first large-scale mirror dataset in the deep learning era. It consists of 4,018 images containing mirrors and their corresponding manually annotated mirror masks, taken from diverse daily life scenes. It is divided into 3,063 training images and 955 testing images.

TABLE 5
Quantitative comparison to state-of-the-art methods on the MSD [9] testing set. All methods are re-trained on the MSD [9] training set. * denotes using CRF [85] for post-processing. "Statistics" refers to thresholding mirror location statistics from the MSD [9] training set as a mirror mask for detection. The first, second, and third best results are marked in **red**, **green**, and **blue**, respectively. Our GDNet-B-M achieves superior performance on all five evaluation metrics.

method	Pub.'Year	IoU↑	PA↑	F_β^m↑	M↓	BER↓
Statistics	-	30.83	0.701	0.438	0.358	32.89
PSPNet [13]	CVPR'17	63.21	0.892	0.746	0.117	15.82
ICNet [86]	ECCV'18	57.25	0.878	0.710	0.124	18.75
Mask RCNN [2]	ICCV'17	63.18	0.884	0.819	0.095	14.35
DSS [90]	TPAMI'19	59.11	0.896	0.743	0.125	18.81
PiCANet [7]	CVPR'18	71.78	0.918	0.808	0.088	10.99
RAS [91]	ECCV'18	60.48	0.896	0.758	0.111	17.60
R^{3}Net* [92]	IJCAI'18	_73.21_	_0.933_	_0.847_	_0.068_	_11.39_
DSC [28]	CVPR'18	69.71	0.917	0.812	0.087	11.77
BDRAR* [29]	ECCV'18	67.43	0.907	0.804	0.093	12.41
MirrorNet* [9]	ICCV'19	*78.95*	*0.935*	*0.857*	*0.065*	*6.39*
GDNet-B-M	**Ours**	**81.70**	**0.952**	**0.894**	**0.050**	**6.17**

Implementation details. We re-train our GDNet-B on the MSD [9] training set and dubbed this model as GDNet-B-M. The training process takes about 23 hours for 200 epochs and the inference takes 0.034 seconds on each image.

Evaluation metrics. Following [9], we adopt five metrics to quantitatively evaluate the mirror segmentation performance of GDNet-B-M, *i.e.*, intersection over union (IoU), pixel accuracy (PA), maximum F-measure (F_β^m), mean absolute error (M), and balance error rate (BER).

6.2 Comparison with the State-of-the-arts

We compare our GDNet-B-M against 10 state-of-the-art related methods, including PSPNet [13], ICNet [86], Mask RCNN [2], DSS [90], PiCANet [7], RAS [91], R^{3}Net [92], DSC [28], BDRAR [29], and MirrorNet [9].

Table 5 shows the quantitative comparison results, in which we can see that GDNet-B-M outperforms all the other counterparts. This demonstrates that large-field contextual information and boundary cues can effectively boost the performance of mirror segmentation. In addition, Figure 11 shows some qualitative comparison results. Benefited by the rich contextual features extracted by the LCFI module and the boundary information perceived by the BFE module, GDNet-B-M successfully suppresses the mirror-like regions (*e.g.*, the first row) and handles the large variations inside the mirror regions (*e.g.*, the last row), leading a more robust mirror detection.
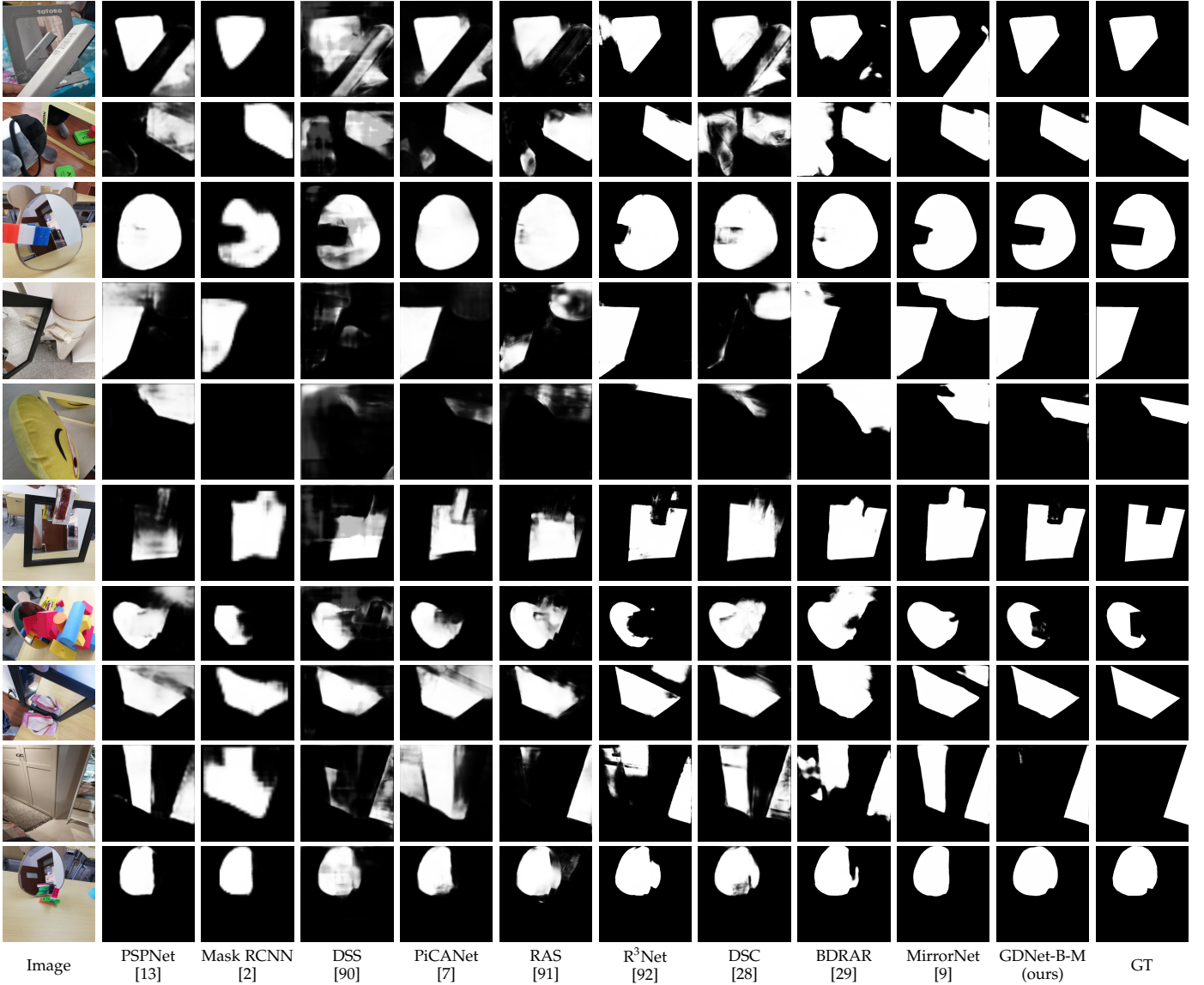
Fig. 11. Visual comparison of GDNet-B-M to the state-of-the-art methods on the MSD [9] testing set.

7 EXPERIMENTS ON SALIENCY DETECTION

In addition to glass detection and mirror segmentation, contexts also play an important role in salient object detection (SOD). High-level contexts describe the relations between different parts/objects and thus are helpful for discovering the specific locations of salient objects while low-level contexts could provide fine detail information for delineating the salient object boundaries. Benefiting from the exploration of rich contexts in the large field-of-view, our network has the potential to handle the salient object detection task.

7.1 Experimental Settings

Benchmark datasets. We conduct the experiments on five frequently used benchmark datasets: PASCAL-S [100], DUT-OMRON [101], ECSSD [102], HKU-IS [103], and DUTS [104]. PASCAL-S [100] includes 850 images with cluttered backgrounds and complex foreground objects. DUT-OMRON [101] has 5,168 images with one or two objects in each

image. Most of the foreground objects are structurally complex. ECSSD [102] consists of 1,000 images with complex and semantically meaningful objects. HKU-IS [103] contains 4,447 images, which usually contain multiple disconnected salient objects or salient objects that overlap with the image boundaries. DUTS [104] is currently the largest salient object detection dataset, which are divided into 10,553 training images (DUTS-TR) and 5,019 testing images (DUTS-TE). We follow previous works [80], [105], [106], [107], [108], [109] to use DUTS-TR as the training set and others as testing sets.

Implementation details. We name the GDNet-B retrained for salient object detection as GDNet-B-S. For training, input images are resized to a resolution of 320×320 and are augmented by randomly horizontal flipping. We set the batch size to 10 and keep other settings the same as the experiments on glass detection. It takes about 36 hours for the network to converge for 150 epochs. For testing, the image is first resized to 320×320 for network inference, and the output saliency map is then resized back to the original size of the input image. Both resizing processes use bilinear

TABLE 6
Comparison of the proposed method on the saliency detection task with 17 state-of-the-art methods on five benchmark datasets in terms of the structure-measure S_α (larger is better), the adaptive E-measure E_ϕ^a (larger is better), the weighted F-measure F_β^w (larger is better), and the mean absolute error M (smaller is better). The first, second, and third best results are marked in red, green, and blue, respectively. "*" means that the results are achieved with post-processing by CRF [85]. Our method achieves the state-of-the-art on five benchmark datasets under all four evaluation metrics.

| Method	Pub.'Year	PASCAL-S [100] 850 images				DUT-OMRON [101] 5,016 images				ECSSD [102] 1,000 images				HKU-IS [103] 4,447 images				DUTS-TE [104] 5,168 images			
		$S_\alpha\uparrow$	$E_\phi^a\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\phi^a\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\phi^a\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\phi^a\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\phi^a\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$
C2SNet [110]	ECCV'18	.830	.837	.756	.085	.782	.817	.643	.079	.886	.909	.844	.057	.878	.923	.824	.049	.820	.842	.705	.065
RAS [111]	ECCV'18	.801	.840	.735	.100	.812	.843	.695	.063	.894	.916	.860	.055	.889	.932	.849	.045	.839	.861	.740	.059
PAGRN [25]	CVPR'18	.822	.853	.732	.089	.775	.842	.622	.071	.889	.914	.833	.061	.887	.939	.820	.048	.838	.880	.724	.055
DGRL [112]	CVPR'18	.836	.848	.791	.072	.806	.848	.709	.062	.903	.918	.891	.041	.895	.944	.876	.035	.842	.879	.774	.050
R³Net* [92]	IJCAI'18	.807	.842	.755	.092	.817	.851	.728	.063	.910	.929	.902	.040	.900	.943	.884	.034	.835	.868	.765	.057
BMPM [113]	CVPR'18	.845	.843	.774	.074	.809	.838	.681	.064	.911	.914	.871	.045	.907	.937	.859	.039	.862	.860	.761	.049
PiCANet-R [114]	CVPR'18	.854	.838	.772	.076	.832	.841	.695	.065	.917	.913	.867	.046	.904	.936	.840	.043	.869	.862	.755	.051
DSS* [90]	TPAMI'19	.799	.848	.754	.094	.788	.844	.690	.066	.882	.917	.871	.053	.881	.936	.866	.040	.824	.882	.755	.056
BASNet [105]	CVPR'19	.838	.853	.793	.076	.836	.869	.751	.056	.916	.921	.904	.037	.909	.946	.889	.032	.866	.884	.803	.048
CPD [107]	CVPR'19	.848	.855	.794	.071	.825	.866	.719	.056	.918	.925	.898	.037	.909	.946	.879	.033	.869	.887	.795	.043
PAGE-Net [115]	CVPR'19	.842	.847	.783	.076	.824	.853	.722	.062	.912	.920	.886	.042	.903	.940	.865	.037	.854	.869	.769	.052
AFNet [116]	CVPR'19	.850	.851	.792	.070	.826	.853	.717	.057	.914	.918	.887	.042	.905	.942	.870	.036	.867	.879	.785	.046
BANet [80]	ICCV'19	.852	.859	.802	.070	.832	.860	.736	.059	.924	.928	.908	.035	.913	.950	.886	.032	.879	.892	.811	.040
GCPANet [117]	AAAI'20	.864	.853	.808	.062	.839	.860	.734	.056	.927	.920	.903	.035	.920	.949	.889	.031	.891	.891	.821	.038
F3Net [118]	AAAI'20	.861	.865	.816	.061	.838	.870	.747	.053	.924	.927	.912	.033	.920	.955	.905	.027	.888	.902	.835	.035
MINet-R [119]	CVPR'20	.858	.859	.810	.064	.834	.866	.737	.056	.926	.928	.910	.034	.920	.954	.896	.029	.885	.899	.824	.037
ITSD [120]	CVPR'20	.859	.856	.812	.066	.840	.863	.750	.061	.925	.927	.910	.034	.917	.952	.894	.031	.885	.895	.824	.041
GDNet-B-S	**Ours**	**.864**	**.872**	**.826**	**.058**	**.848**	**.879**	**.769**	**.047**	**.927**	**.927**	**.920**	**.031**	**.923**	**.958**	**.911**	**.026**	**.892**	**.913**	**.850**	**.032**
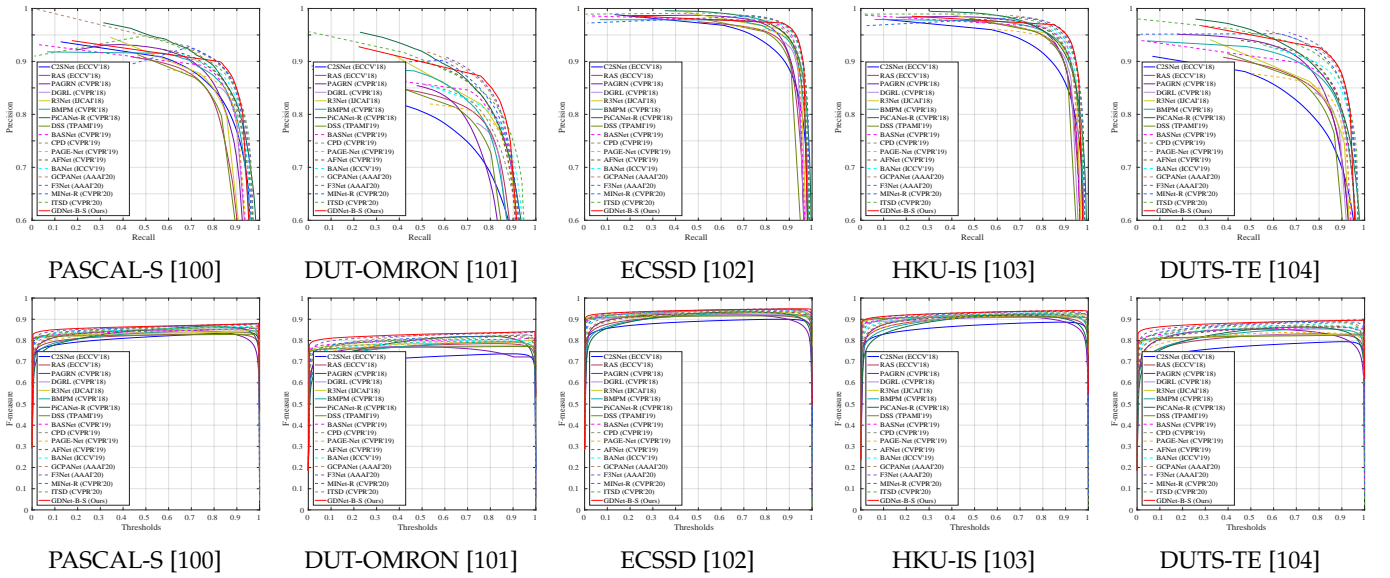
Fig. 12. The PR curves (the top row) and the F-measure curves (the bottom row) on five benchmark datasets. It can be seen that the proposed method performs favorably against the state-of-the-art methods.

interpolation. There is no post-processing (such as the fully connected conditional random field (CRF) [85]) applied to further enhance the final output. The inference time for a 320×320 image is only about 0.033 seconds.

Evaluation metrics. We use six popular metrics to evaluate GDNet-B-S: precision-recall (PR) curve, F-measure curve, structure-measure (S_α) [121], E-measure (E_ϕ) [122], weighted F-measure (F_β^w) [123], and mean absolute error (M).

The PR curve is a standard measure to evaluate the predicted saliency probability maps. The precision and recall of a saliency map are computed by comparing the binarized saliency map against the ground truth map. Under a spe-

cific binarizing threshold, we can obtain a pair of average precision and recall over all binarized saliency maps in a dataset. Varying the threshold from 0 to 1 with a step of $\frac{1}{255}$ produces 256 precision-recall pairs, which are plotted as the PR curve.

For a comprehensive measure on both precision and recall, F-measure (F_β) is computed by Eq. 7. Following most salient object detection works [105], [108], we plot the F-measure curve from all precision-recall pairs under different thresholds. Recent studies [121], [122] have suggested that the weighted F-measure (F_β^w) [123] can provide more reliable evaluation results than the traditional F_β. Thus, we also consider this metric in the comparison.
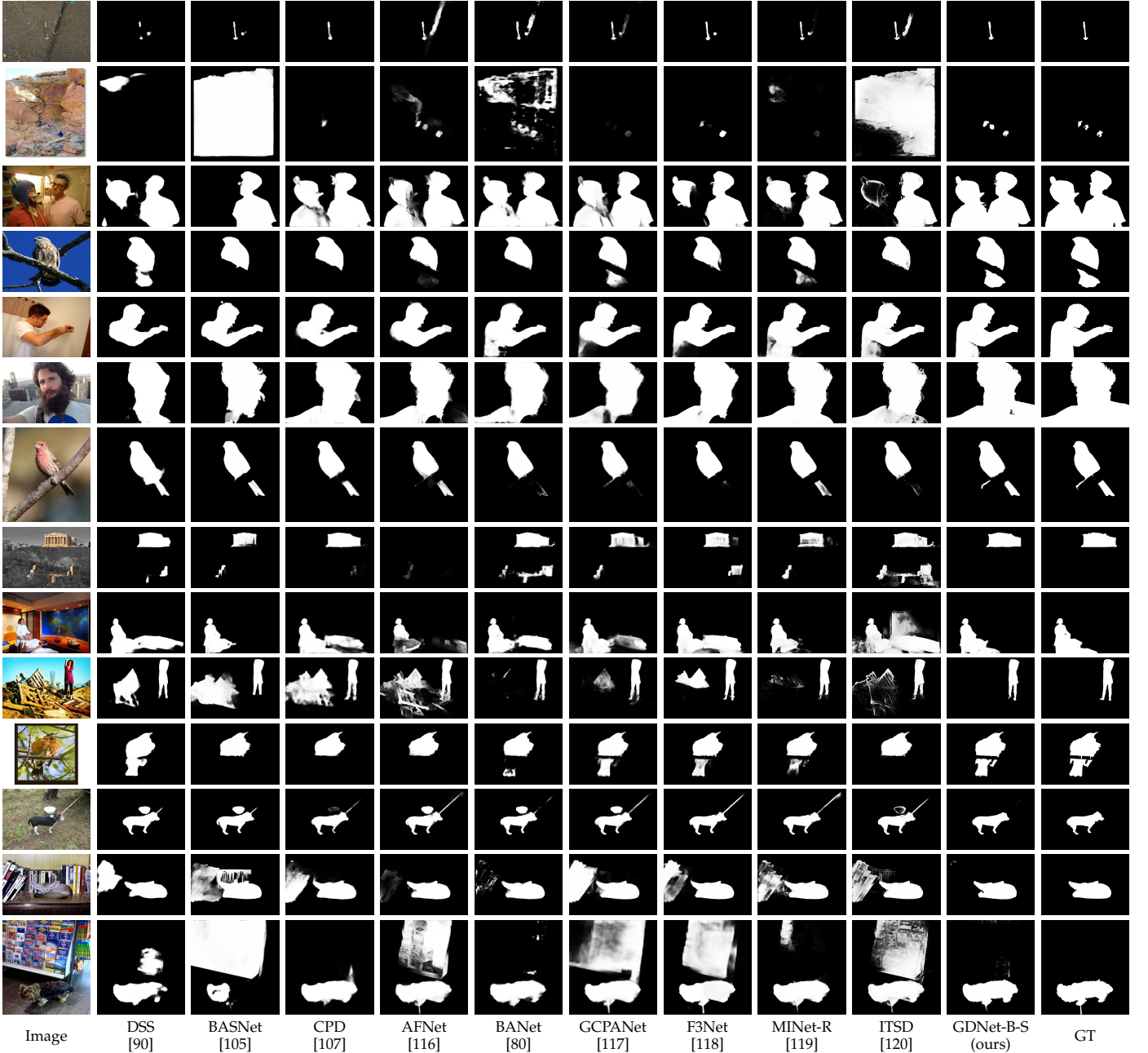
Fig. 13. Visual comparison of GDNet-B-S with state-of-the-art SOD methods. We can see that GDNet-B-S is capable of segmenting salient objects of different scales in various scenes more accurately.

Structure-measure (S_α) focuses on evaluating the structural information of the saliency maps, and is closer to the human visual system than F-measure. Thus, we include structure-measure for a more comprehensive evaluation. Structure-measure (S_α) is defined as:

$$S_\alpha = \alpha S_o + (1 - \alpha)S_r, \tag{10}$$

where S_o and S_r denote the object-aware and region-aware structural similarity, respectively. α is set as 0.5 by default. More details can be found in [121].

Recently, Fan *et al.* [122] propose a human visual perception based E-measure (E_ϕ), which simultaneously evaluates pixel-level matching and image-level statistics. This metric can be used to assess the overall and localized accuracy of the salient object detection results.

7.2 Comparison with the State-of-the-arts

To demonstrate the effectiveness of GDNet-B-S, we compare it against 17 state-of-the-art SOD methods, including C2SNet [110], RAS [111], PAGRN [25], DGRL [112], R³Net [92], BMPM [113], PiCANet-R [114], DSS [90], BASNet [105], CPD [107], PAGE-Net [115], AFNet [116], BANet [80], GC-PANet [117], F3Net [118], MINet-R [119], and ITSD [120]. For fair comparison, all saliency maps from the above methods are either provided by the authors or produced by running their released models. Besides, all saliency maps are evaluated with the same code.

Figure 12 shows the precision-recall (PR) curves and the F-measure curves on the five benchmark datasets. We can see from these results that our GDNet-B-S (denoted by

a solid red line) performs favorably against all the other methods. In addition, Table 6 summarizes the comparison results in terms of the structure-measure (S_α), the adaptive E-measure (E_ϕ^a), the weighted F-measure (F_β^w), and the mean absolute error (M). As we can see, our method outperforms other state-of-the-art SOD methods on almost all five datasets with respect to all four metrics. This clearly demonstrates the superior performance of GDNet-B-S. To further illustrate the advantages of our method for salient object detection, we show some qualitative comparison results in Figure 13. We can see that GDNet-B-S is capable of accurately detecting small objects in the scenes (*e.g.*, the first two rows) and segmenting out the complete salient objects from cluttered backgrounds (*e.g.*, 3*rd*, 4*th*, 5*th*, 6*th*, and 7*th* rows). While the state-of-the-art methods are typically confused by the messy background objects (*e.g.*, 8-12*th* rows) or the high similarities between foreground and background (*e.g.*, the last two rows), GDNet-B-S successfully handles such ambiguities and avoids false positive prediction. This is mainly contributed to the large-field contextual feature learning, which could provide rich contextual information for context inference and salient object localization.

8 DISCUSSION

In this section, we analysis the limitations of our method, and discuss the potential applications of glass detection and possible future research directions.

Limitations. Although the proposed method has achieved state-of-the-art glass detection performance on the GDD testing set, it does have its limitations. In particular, the proposed method may fail in cases where the scene is very complex or provides insufficient contexts both inside and outside the glass, as shown in Figure 14. We believe that a deeper research is needed to study these extreme cases.

Potential applications. Automatic glass detection has various possible applications. Here, we envision two potential ones: obstacle avoidance for drones and intelligence photography/editing [124], [125]. If a drone is equipped with the ability to detect glass, it can easily find out where the glass is and avoid crashing into the glass (*e.g.*, Figure 15(b)). In addition, if a phone camera is equipped with the ability to detect glass, it can identify the glass and then remove the reflections [3] from the glass region to obtain the desired result (*e.g.*, Figure 15(d)).

Future work. People may sometimes bump into glass, due to the transparency nature of glass. As the first attempt to address the glass detection problem with a computational method, we focus in this paper on detecting various types of glass in our daily life, from a single input image. This can benefit a wide range of applications. However, we have not addressed the extreme cases where even humans may not be able to identify where the glass is correctly. Studying these extreme cases can be an interesting future work. Besides, it would also be interesting to study glass detection in videos and the instance glass detection problem. New techniques such as weakly supervised learning [126], zero-shot learning [127], and multi-scale backbone [128] for glass detection may also be explored.
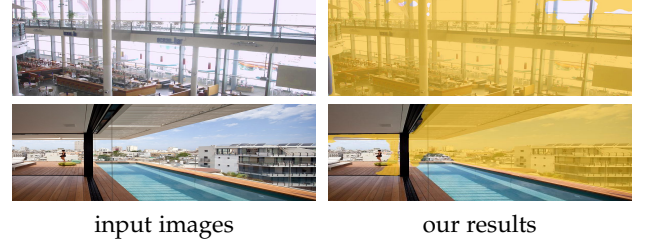

| input images | our results |

Fig. 14. Failure cases. Our glass detection method may fail in cases where the scene is extremely complicated (the top row) or contains insufficient contexts both inside and outside the glass (the bottom row).

(a) delivery drone w/o GDS   (b) delivery drone w/ GDS

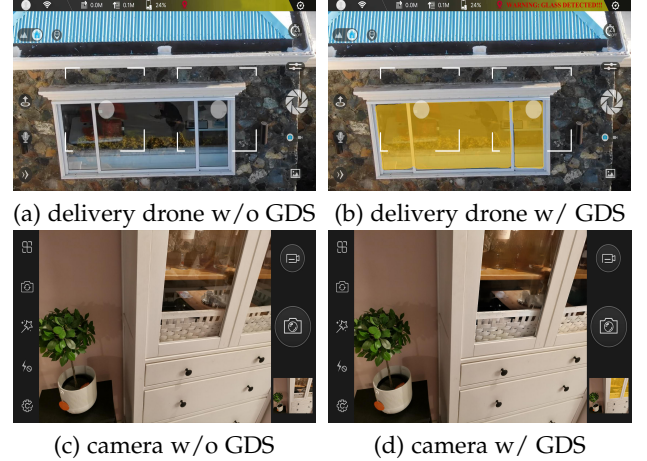(c) camera w/o GDS   (d) camera w/ GDS

Fig. 15. Potential applications of glass detection. If a drone is equipped with the ability to detect glass, it may easily find out where the glass is and avoid crashing into the glass (b). If a phone camera is equipped with the ability to detect glass, it may identify the glass region and then remove the reflections [3] only from the glass region to get the desired result (d).

9 CONCLUSION

In this paper, we have proposed an important problem of detecting glass from a single input image and contributed the first large-scale glass detection dataset (GDD) covering diverse scenes in our daily life. A novel network is proposed to address this challenging task. It leverages both high-level and low-level contexts extracted from a large field-of-view and the boundary cue to detect glass of different sizes in various scenes. Extensive evaluations on the images within and beyond the GDD testing set demonstrate the strong capability of our network to accurately detect glass. We have further applied our network to the other vision tasks (*i.e.*, mirror segmentation and salient object detection), validating the effectiveness and generalization capability of the proposed method. Finally, we have discussed some potential applications of glass detection and possible future research directions.

REFERENCES

[1] Z. Li and N. Snavely, "Megadepth: Learning single-view depth prediction from internet photos," in *CVPR*, 2018.

[2] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask r-cnn," in *ICCV*, 2017.

[3] J. Yang, D. Gong, L. Liu, and Q. Shi, "Seeing deeply and bidirectionally: A deep learning approach for single image reflection removal," in *ECCV*, 2018.

[4] Z. Cai and N. Vasconcelos, "Cascade r-cnn: Delving into high quality object detection," in *CVPR*, 2018.

[5] Z. Huang, L. Huang, Y. Gong, C. Huang, and X. Wang, "Mask scoring r-cnn," in *CVPR*, 2019.

[6] A. Kalra, V. Taamazyan, S. K. Rao, K. Venkataraman, and R. Raskar, "Deep polarization cues for transparent object segmentation," in *CVPR*, 2020.

[7] N. Liu, J. Han, and M.-H. Yang, "Picanet: Learning pixel-wise contextual attention for saliency detection," in *CVPR*, 2018.

[8] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, and M. Jagersand, "Basnet: Boundary-aware salient object detection," in *CVPR*, 2019.

[9] X. Yang, H. Mei, K. Xu, X. Wei, B. Yin, and R. W. Lau, "Where is my mirror?" in *ICCV*, 2019.

[10] H. Mei, X. Yang, Y. Wang, Y. Liu, S. He, Q. Zhang, X. Wei, and R. W. Lau, "Don't hit me! glass detection in real-world scenes," in *CVPR*, 2020.

[11] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *CVPR*, 2015.

[12] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE TPAMI*, 2017.

[13] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *CVPR*, 2017.

[14] H. Ding, X. Jiang, B. Shuai, A. Qun Liu, and G. Wang, "Context contrasted feature and gated multi-scale aggregation for scene segmentation," in *CVPR*, 2018.

[15] H. Zhang, K. Dana, J. Shi, Z. Zhang, X. Wang, A. Tyagi, and A. Agrawal, "Context encoding for semantic segmentation," in *CVPR*, 2018.

[16] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu, "Dual attention network for scene segmentation," in *CVPR*, 2019.

[17] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "Ccnet: Criss-cross attention for semantic segmentation," in *ICCV*, 2019.

[18] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time object detection with region proposal networks," *IEEE TPAMI*, 2017.

[19] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *CVPR*, 2018.

[20] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *CVPR*, 2009.

[21] D. A. Klein and S. Frintrop, "Center-surround divergence of feature statistics for salient object detection," in *ICCV*, 2011.

[22] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. Torr, and S.-M. Hu, "Global contrast based salient region detection," *IEEE TPAMI*, 2014.

[23] N. Liu and J. Han, "Dhsnet: Deep hierarchical saliency network for salient object detection," in *CVPR*, 2016.

[24] P. Zhang, D. Wang, H. Lu, H. Wang, and X. Ruan, "Amulet: Aggregating multi-level convolutional features for salient object detection," in *ICCV*, 2017.

[25] X. Zhang, T. Wang, J. Qi, H. Lu, and G. Wang, "Progressive attention guided recurrent network for salient object detection," in *CVPR*, 2018.

[26] L. Zhang, J. Dai, H. Lu, Y. He, and G. Wang, "A bi-directional message passing model for salient object detection," in *CVPR*, 2018.

[27] W. Wang, J. Shen, M.-M. Cheng, and L. Shao, "An iterative and cooperative top-down and bottom-up inference network for salient object detection," in *CVPR*, 2019.

[28] X. Hu, L. Zhu, C.-W. Fu, J. Qin, and P.-A. Heng, "Direction-aware spatial context features for shadow detection," in *CVPR*, 2018.

[29] L. Zhu, Z. Deng, X. Hu, C.-W. Fu, X. Xu, J. Qin, and P.-A. Heng, "Bidirectional feature pyramid network with recurrent attention residual modules for shadow detection," in *ECCV*, 2018.

[30] Q. Zheng, X. Qiao, Y. Cao, and R. W. Lau, "Distraction-aware shadow detection," in *CVPR*, 2019.

[31] X. Han, C. Nguyen, S. You, and J. Lu, "Single image water hazard detection using fcn with reflection attention units," in *ECCV*, 2018.

[32] A. Levin and Y. Weiss, "User assisted separation of reflections from a single image using a sparsity prior," *IEEE TPAMI*, 2007.

[33] R. Wan, B. Shi, T. A. Hwee, and A. C. Kot, "Depth of field guided reflection removal," in *ICIP*, 2016.

[34] Y. Li and M. S. Brown, "Single image layer separation using relative smoothness," in *CVPR*, 2014.

[35] Y. Shih, D. Krishnan, F. Durand, and W. T. Freeman, "Reflection removal using ghosting cues," in *CVPR*, 2015.

[36] Q. Fan, J. Yang, G. Hua, B. Chen, and D. Wipf, "A generic deep architecture for single image reflection removal and image smoothing," in *ICCV*, 2017.

[37] R. Wan, B. Shi, L.-Y. Duan, A.-H. Tan, and A. C. Kot, "Crrn: Multi-scale guided concurrent reflection removal network," in *CVPR*, 2018.

[38] X. Zhang, R. Ng, and Q. Chen, "Single image reflection separation with perceptual losses," in *CVPR*, 2018.

[39] K. Wei, J. Yang, Y. Fu, D. Wipf, and H. Huang, "Single image reflection removal exploiting misaligned training data and network enhancements," in *CVPR*, 2019.

[40] Y. Qiao, Y. Liu, X. Yang, D. Zhou, M. Xu, Q. Zhang, and X. Wei, "Attention-guided hierarchical structure aggregation for image matting," in *CVPR*, 2020.

[41] Y. Liu, J. Xie, Y. Qiao, Y. Tang, and X. Yang, "Prior-induced information alignment for image matting," *IEEE TMM*, 2021.

[42] Y. Liu, J. Xie, X. Shi, Y. Qiao, Y. Huang, Y. Tang, and X. Yang, "Tripartite information mining and integration for image matting," in *ICCV*, 2021.

[43] Y. Qiao, J. Zhu, C. Long, Z. Zhang, Y. Wang, Z. Du, and X. Yang, "Cpral: Collaborative panoptic-regional active learning for semantic segmentation," in *AAAI*, 2022.

[44] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *CVPR*, 2017.

[45] S. Liu, D. Huang *et al.*, "Receptive field block net for accurate and fast object detection," in *ECCV*, 2018.

[46] J. He, Z. Deng, L. Zhou, Y. Wang, and Y. Qiao, "Adaptive pyramid context network for semantic segmentation," in *CVPR*, 2019.

[47] J. He, Z. Deng, and Y. Qiao, "Dynamic multi-scale filters for semantic segmentation," in *ICCV*, 2019.

[48] H. Wu, Y. Wu, S. Zhang, P. Li, and Z. Wen, "Cartoon image segmentation based on improved slic superpixels and adaptive region propagation merging," in *ICSIP*, 2016.

[49] X. Yang, H. Mei, K. Xu, X. Zhang, B. Yin, Q. Zhang, and X. Wei, "Drfn: Deep recurrent fusion network for single-image super-resolution with large factors," *IEEE TMM*, 2019.

[50] J. Zhang, C. Long, Y. Wang, X. Yang, H. Mei, and B. Yin, "Multi-context and enhanced reconstruction network for single image super resolution," in *ICME*, 2020.

[51] Y. Nie, C. Xiao, H. Sun, and P. Li, "Compact video synopsis via global spatiotemporal optimization," *IEEE TVCG*, 2012.

[52] X. Tian, K. Xu, X. Yang, B. Yin, and R. W. Lau, "Weakly-supervised salient instance detection," in *BMVC*, 2020.

[53] ——, "Learning to detect instance-level salient objects using complementary image labels," *IJCV*, 2021.

[54] D. Lin, R. Zhang, Y. Ji, P. Li, and H. Huang, "Scn: switchable context network for semantic segmentation of rgb-d images," *IEEE Transactions on Cybernetics*, 2018.

[55] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun, "Large kernel matters–improve semantic segmentation by global convolutional network," in *CVPR*, 2017.

[56] X. Hu, L. Zhu, C.-W. Fu, J. Qin, and P.-A. Heng, "Direction-aware spatial context features for shadow detection," in *CVPR*, 2018.

[57] T. Wang, X. Yang, K. Xu, S. Chen, Q. Zhang, and R. W. Lau, "Spatial attentive single-image deraining with a high quality real rain dataset," in *CVPR*, 2019.

[58] H. Ding, X. Jiang, B. Shuai, A. Qun Liu, and G. Wang, "Context contrasted feature and gated multi-scale aggregation for scene segmentation," in *CVPR*, 2018.

[59] H. Mei, B. Dong, W. Dong, P. Peers, X. Yang, Q. Zhang, and X. Wei, "Depth-aware mirror segmentation," in *CVPR*, 2021.

[60] J. Lin, G. Wang, and R. W. Lau, "Progressive mirror detection," in *CVPR*, 2020.

[61] J. Lin, Z. He, and R. W. Lau, "Rich context aggregation with reflection prior for glass surface detection," in *CVPR*, 2021.

[62] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *CVPR*, 2018.

[63] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu, "Dual attention network for scene segmentation," in *CVPR*, 2019.

[64] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "Cc-net: Criss-cross attention for semantic segmentation," in *ICCV*, 2019.

[65] H. Mei, G.-P. Ji, Z. Wei, X. Yang, X. Wei, and D.-P. Fan, "Camouflaged object segmentation with distraction mining," in *CVPR*, 2021.

[66] E. Xie, W. Wang, W. Wang, P. Sun, H. Xu, D. Liang, and P. Luo, "Segmenting transparent object in the wild with transformer," in *IJCAI*, 2021.

[67] G. Li, X. He, W. Zhang, H. Chang, L. Dong, and L. Lin, "Non-locally enhanced encoder-decoder network for single image de-raining," in *ACM MM*, 2018.

[68] D. Liu, B. Wen, Y. Fan, C. C. Loy, and T. S. Huang, "Non-local recurrent network for image restoration," in *NeurIPS*, 2018.

[69] T. Dai, J. Cai, Y. Zhang, S.-T. Xia, and L. Zhang, "Second-order attention network for single image super-resolution," in *CVPR*, 2019.

[70] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *CVPR*, 2016.

[71] M. Everingham, S. M. Eslami, L. Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes challenge: A retrospective," *IJCV*, 2015.

[72] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba, "Scene parsing through ade20k dataset," in *CVPR*, 2017.

[73] X. Tan, K. Xu, Y. Cao, Y. Zhang, L. Ma, and R. W. Lau, "Night-time scene parsing with a large real dataset," *IEEE TIP*, 2021.

[74] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *CVPR*, 2009.

[75] S. Xie, R. Girshick, P. Dollar, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *CVPR*, 2017.

[76] S. Woo, J. Park, J.-Y. Lee, and I. So Kweon, "Cbam: Convolutional block attention module," in *ECCV*, 2018.

[77] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *CVPR*, 2018.

[78] J. Su, J. Li, Y. Zhang, C. Xia, and Y. Tian, "Selectivity or invariance: Boundary-aware salient object detection," in *ICCV*, 2019.

[79] E. Xie, W. Wang, W. Wang, M. Ding, C. Shen, and P. Luo, "Segmenting transparent objects in the wild," in *ECCV*, 2020.

[80] J. Su, J. Li, Y. Zhang, C. Xia, and Y. Tian, "Selectivity or invariance: Boundary-aware salient object detection," in *ICCV*, 2019.

[81] P.-T. de Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, "A tutorial on the cross-entropy method," *Annals of Operations Research*, 2005.

[82] T. Zhao and X. Wu, "Pyramid feature attention network for saliency detection," in *CVPR*, 2019.

[83] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *NeurIPS*, 2019.

[84] W. Liu, A. Rabinovich, and A. C. Berg, "Parsenet: Looking wider to see better," *arXiv:1506.04579*, 2015.

[85] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected crfs with gaussian edge potentials," in *NeurIPS*, 2011.

[86] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia, "Icnet for real-time semantic segmentation on high-resolution images," in *ECCV*, 2018.

[87] M. Yang, K. Yu, C. Zhang, Z. Li, and K. Yang, "Denseaspp for semantic segmentation in street scenes," in *CVPR*, 2018.

[88] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Bisenet: Bilateral segmentation network for real-time semantic segmentation," in *ECCV*, 2018.

[89] H. Zhao, Y. Zhang, S. Liu, J. Shi, C. Change Loy, D. Lin, and J. Jia, "Psanet: Point-wise spatial attention network for scene parsing," in *ECCV*, 2018.

[90] Q. Hou, M.-M. Cheng, X. Hu, A. Borji, Z. Tu, and P. Torr, "Deeply supervised salient object detection with short connections," *IEEE TPAMI*, 2019.

[91] S. Chen, X. Tan, B. Wang, and X. Hu, "Reverse attention for salient object detection," in *ECCV*, 2018.

[92] Z. Deng, X. Hu, L. Zhu, X. Xu, J. Qin, G. Han, and P.-A. Heng, "R3net: Recurrent residual refinement network for saliency detection," in *IJCAI*, 2018.

[93] Z. Wu, L. Su, and Q. Huang, "Cascaded partial decoder for fast and accurate salient object detection," in *CVPR*, 2019.

[94] J.-J. Liu, Q. Hou, M.-M. Cheng, J. Feng, and J. Jiang, "A simple pooling-based design for real-time salient object detection," in *CVPR*, 2019.

[95] J.-X. Zhao, J.-J. Liu, D.-P. Fan, Y. Cao, J. Yang, and M.-M. Cheng, "Egnet:edge guidance network for salient object detection," in *ICCV*, 2019.

[96] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[97] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *CVPR*, 2018.

[98] A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niessner, M. Savva, S. Song, A. Zeng, and Y. Zhang, "Matterport3D: Learning from RGB-D data in indoor environments," in *3DV*, 2017.

[99] I. Armeni, S. Sax, A. R. Zamir, and S. Savarese, "Joint 2d-3d-semantic data for indoor scene understanding," *arXiv:1702.01105*, 2017.

[100] Y. Li, X. Hou, C. Koch, J. M. Rehg, and A. L. Yuille, "The secrets of salient object segmentation," in *CVPR*, 2014.

[101] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *CVPR*, 2013.

[102] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *CVPR*, 2013.

[103] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *CVPR*, 2015.

[104] L. Wang, H. Lu, Y. Wang, M. Feng, D. Wang, B. Yin, and X. Ruan, "Learning to detect salient objects with image-level supervision," in *CVPR*, 2017.

[105] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, and M. Jager-sand, "Basnet: Boundary-aware salient object detection," in *CVPR*, 2019.

[106] T. Zhao and X. Wu, "Pyramid feature attention network for saliency detection," in *CVPR*, 2019.

[107] Z. Wu, L. Su, and Q. Huang, "Cascaded partial decoder for fast and accurate salient object detection," in *CVPR*, 2019.

[108] J.-X. Zhao, J.-J. Liu, D.-P. Fan, Y. Cao, J. Yang, and M.-M. Cheng, "Egnet: Edge guidance network for salient object detection," in *ICCV*, 2019.

[109] H. Mei, Y. Liu, Z. Wei, D. Zhou, X. Xiaopeng, Q. Zhang, and X. Yang, "Exploring dense context for salient object detection," *IEEE TCSVT*, 2021.

[110] X. Li, F. Yang, H. Cheng, W. Liu, and D. Shen, "Contour knowledge transfer for salient object detection," in *ECCV*, 2018.

[111] S. Chen, X. Tan, B. Wang, and X. Hu, "Reverse attention for salient object detection," in *ECCV*, 2018.

[112] T. Wang, L. Zhang, S. Wang, H. Lu, G. Yang, X. Ruan, and A. Borji, "Detect globally, refine locally: A novel approach to saliency detection," in *CVPR*, 2018.

[113] L. Zhang, J. Dai, H. Lu, Y. He, and G. Wang, "A bi-directional message passing model for salient object detection," in *CVPR*, 2018.

[114] N. Liu, J. Han, and M.-H. Yang, "Picanet: Learning pixel-wise contextual attention for saliency detection," in *CVPR*, 2018.

[115] W. Wang, S. Zhao, J. Shen, S. C. H. Hoi, and A. Borji, "Salient object detection with pyramid attention and salient edges," in *CVPR*, 2019.

[116] M. Feng, H. Lu, and E. Ding, "Attentive feedback network for boundary-aware salient object detection," in *CVPR*, 2019.

[117] Z. Chen, Q. Xu, R. Cong, and Q. Huang, "Global context-aware progressive aggregation network for salient object detection," in *AAAI*, 2020.

[118] J. Wei, S. Wang, and Q. Huang, "F3net: Fusion, feedback and focus for salient object detection," in *AAAI*, 2020.

[119] Y. Pang, X. Zhao, L. Zhang, and H. Lu, "Multi-scale interactive network for salient object detection," in *CVPR*, 2020.
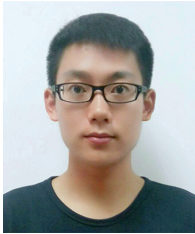
[120] H. Zhou, X. Xie, J.-H. Lai, Z. Chen, and L. Yang, "Interactive two-stream decoder for accurate and fast saliency detection," in *CVPR*, 2020.

[121] D.-P. Fan, M.-M. Cheng, Y. Liu, T. Li, and A. Borji, "Structure-measure: A new way to evaluate foreground maps," in *ICCV*, 2017.

[122] D.-P. Fan, C. Gong, Y. Cao, B. Ren, M.-M. Cheng, and A. Borji, "Enhanced-alignment measure for binary foreground map evaluation," in *IJCAI*, 2018.

[123] R. Margolin, L. Zelnik-Manor, and A. Tal, "How to evaluate foreground maps?" in *CVPR*, 2014.

[124] B. Sheng, H. Sun, M. Magnor, and P. Li, "Video colorization using parallel optimization in feature space," *IEEE TCSVT*, 2013.

[125] P. Li, H. Sun, C. Huang, J. Shen, and Y. Nie, "Interactive image/video retexturing using gpu parallelism," *Computers & Graphics*, 2012.

[126] Y. Shen, R. Ji, S. Zhang, W. Zuo, and Y. Wang, "Generative adversarial learning towards fast weakly supervised detection," in *CVPR*, 2018.

[127] Y. Zhu, M. Elhoseiny, B. Liu, X. Peng, and A. Elgammal, "A generative adversarial approach for zero-shot learning from noisy texts," in *CVPR*, 2018.

[128] S. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang, and P. H. Torr, "Res2net: A new multi-scale backbone architecture." *IEEE TPAMI*, 2019.

Qiang Zhang was born in Xian, China, in 1971. He received his M.Eng. degree in economic engineering and Ph.D. degree in circuits and systems from Xidian University, Xian, China, in 1999 and 2002, respectively. He was a lecturer at the Center of Advanced Design Technology, Dalian University, Dalian, China, in 2003 and was a professor in 2005. His research interests are bio-inspired computing and its applications. He has authored more than 70 papers in the above fields. Thus far, he has served on the editorial board of seven international journals and has edited special issues in journals such as *Neurocomputing* and *International Journal of Computer Applications in Technology*.
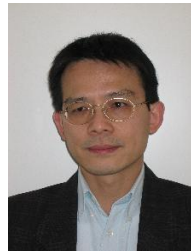

Xiaopeng Wei was born in Dalian, China, in 1959. He received his Ph.D. degree from Dalian University of Technology in 1993. He is a professor at Dalian University of Technology. His research areas include computer animation, computer vision, robots, and intelligent CAD. So far, he has (co-)authored approximately 200 published papers.

Rynson W.H. Lau received the Ph.D. degree from the University of Cambridge. He was on the Faculty of Durham University. He is currently with the City University of Hong Kong.

He serves on the Editorial Board of International Journal of Computer Vision (IJCV). He served as the Guest Editor of a number of journal special issues, including ACM Trans. on Internet Technology, the IEEE Trans. on Multimedia, IEEE Trans. on Visualization and Computer Graphics, and IEEE Computer Graphics & Applications. He also served in the committee of a number of conferences, including Program Co-chair of ACM VRST 2004, ACM MTDL 2009, IEEE U-MEDIA 2010, and Conference Co-Chair of CASA 2005, ACM VRST 2005, ACM MDI 2009, ACM VRST 2014. His research interests include computer graphics and computer vision.

Haiyang Mei received the B.Eng. degree in the Automation from Dalian University of Technology, Dalian China, in 2017. He is currently working toward the Ph.D. degree with the Department of Computer Science and Technology, Dalian University of Technology, Dalian, China. His research interests include image processing and computer vision.

Xin Yang is a Professor in the Department of Computer Science at Dalian University of Technology, China. Yang received his B.S. degree in Computer Science from Jilin University in 2007. From 2007 to June 2012, he was a joint Ph.D. student at Zhejiang University and UC Davis for Graphics and received his Ph.D. degree in July 2012. His research interests include computer graphics and robotic vision.

Letian Yu is a Ph.D. student in the Department of Computer Science at Dalian University of Technology. He received his B.S. degree in Computer Science and Technology in 2019. His research interests are computer vision and image processing.