Boundary-Aware Refinement with Environment-Robust Adapter Tuning for Underwater Instance Segmentation

Supplementary Material

Pin-Chi Pan R12942103@NTU.EDU.TW
Graduate Institute of Communication Engineering, National Taiwan University, Taiwan

Soo-Chang Pei PEISC@NTU.EDU.TW
Department of Electrical Engineering, National Taiwan University, Taiwan

1. Training Setup

- **Swin Transformer Backbone:** We utilized a Mask R-CNN-based architecture with Swin Transformer as the backbone to leverage its powerful hierarchical representation and environmental adaptability features. Our setup includes the BARDecoder module for multi-stage feature refinement and the ERA-tuning module to handle the domain shift inherent in underwater conditions. Key hyperparameters were set as follows: a base learning rate of 0.0001 was used with the AdamW optimizer, employing $(\beta_1, \beta_2) = (0.9, 0.999)$ for momentum parameters and a weight decay of 0.05 to prevent overfitting. A warmup phase was implemented with 1,000 iterations to gradually increase the learning rate, ensuring stable convergence. The model was trained for a total of 12 epochs, with a learning rate decay scheduled at epochs 8 and 11, following a step decay schedule to fine-tune performance in later stages.

- **ConvNeXt V2 Backbone:** Similarly, we used a Mask R-CNN-based architecture with ConvNeXt V2 as the backbone to explore its advantages in handling complex visual patterns common in underwater scenes. The core training configurations, including the optimizer, learning rate, warmup phase, and epoch schedule, mirrored those of the Swin Transformer backbone. We also incorporated environmental robustness features, tailoring ConvNeXt V2 with layer-wise decay to manage feature adaptation effectively. Specifically, a decay rate of 0.95 was applied over six layers, optimizing the balance between retaining pretrained knowledge and adapting to underwater specifics.

The configuration files included in our code repository provide an overview of additional setup details.

2. PyTorch-Like Code Implementation

We provide a PyTorch-like implementation of Boundary-Aware Cross-Entropy (BACE) loss, illustrating how range-null space decomposition enhances segmentation accuracy, particularly at object boundaries. This implementation projects predictions onto range-space and null-space components, refining object contours while preserving structural consistency. In the implementation, we first apply max pooling to downsample both predictions and ground truth masks, extracting dominant structures and reducing high-frequency noise. This is

```
# Boundary-Aware Cross Entropy (BACE) Loss
def boundary_aware_cross_entropy(pred, label, scale, class_weight):
    # Downsample the prediction (A * pred) using max pooling
    A_pred = MaxPooling(pred, kernel_size=scale)

    # Upsample the result (A^T * A * pred) back to original size
    AtA_pred = Upsample(A_pred, scale_factor=scale)

    # Compute orthogonal projection (I - A^T * A) * pred
    ortho_project = pred - AtA_pred

    # Downsample the ground truth (A * label) using max pooling
    A_label = MaxPooling(label, kernel_size=scale)

    # Upsample the ground truth (A^T * A * label) back to original size
    AtA_label = Upsample(A_label, scale_factor=scale)

    # Compute parallel projection (A^T * A * label)
    parallel_project = AtA_label

    # Combine orthogonal and parallel projections for refined mask
    refined_pred = parallel_project + ortho_project

    # Compute the binary cross-entropy loss with logits
    loss = BinaryCrossEntropyWithLogits(refined_pred, label, weight=class_weight)

    return loss

# Example inputs: pred (prediction), label (ground truth)
# Set scale (e.g., scale=4), and class_weight if needed.
# Call boundary_aware_cross_entropy(pred, label, scale, class_weight)
```

followed by nearest-neighbor interpolation to restore spatial resolution. The range-space component ensures consistency with non-boundary regions, while the null-space component captures finer details, correcting boundary misalignment. The final mask is computed by combining these components and applying Binary Cross-Entropy (BCE) loss for segmentation supervision. The BACE loss integrates seamlessly into modern segmentation pipelines with minimal computational overhead. Unlike standard loss functions, it explicitly refines boundary features, improving segmentation accuracy in complex scenarios. Its flexibility allows it to be used across different segmentation tasks with customizable linear operators $\boldsymbol{A}$, such as blurring or inpainting operators in inverse problems. Additionally, the scaling parameter in the implementation determines the downsampling factor, providing adaptability for different dataset resolutions and object complexities. Researchers and practitioners can easily incorporate this method into existing frameworks to enhance segmentation precision, particularly for tasks requiring fine-grained boundary refinement.

3. Computational Efficiency Analysis

We provide a comparison of frames per second (FPS) to evaluate the computational efficiency of BARD-ERA relative to baseline methods. Table 1 reports the FPS and parameter count for models using Swin Transformer backbones. While BARD-ERA achieves state-of-the-art segmentation performance, it maintains competitive inference speed. Compared to standard Mask R-CNN, our method introduces a moderate computational overhead due to multi-scale refinement and adapter-based tuning. However, BARD-ERA remains

significantly more efficient than USIS-SAM, which employs a ViT-H backbone, leading to substantially higher computational costs. The trade-off between accuracy and efficiency underscores the suitability of BARD-ERA for practical applications, balancing segmentation precision with feasible real-time performance.

Swin Transformer		
Method	**Params**	**FPS**
Mask R-CNN He et al. (2017)	106.75 M	8.325
Cascade Mask R-CNN Cai and Vasconcelos (2018)	139.79 M	7.430
Point Rend Kirillov et al. (2020)	118.84 M	7.430
SOLOv2 Wang et al. (2020)	109.00 M	6.775
Mask2Former Cheng et al. (2022)	106.75 M	4.401
WaterMask Lian et al. (2023)	110.40 M	9.597
USIS-SAM Lian and others. (2024)	698.12 M	2.750
BARD-ERA (Ours)	114.44 M	4.866

Table 1: Comparison of FPS and parameter efficiency among different instance segmentation methods using the Swin Transformer backbone.

Method	**Trained Params***	**%**	**Extra Structure**	**mAP**	**AP$_{50}$**	**AP$_{75}$**	**AP$_S$**	**AP$_M$**	**AP$_L$**
ConvNeXt V2									
Full Fine-Tuning	87.69 M	100.00 %	✘	28.5	46.0	32.3	7.9	22.1	40.9
BitFit	0.13 M	0.15 %	✘	27.9	47.6	29.9	9.8	21.9	38.0
NormTuning	0.04 M	0.05 %	✘	26.5	47.1	28.0	9.4	21.2	36.6
PARTIAL-1	8.46 M	9.64 %	✘	26.0	46.6	27.1	8.0	21.4	36.2
VPT	0.20 M	0.23 %	✓	26.8	47.2	28.0	9.8	20.8	36.5
Conv-Adapter	2.36 M	2.63 %	✓	24.4	43.7	25.5	8.9	19.0	34.9
ERA (Ours)	1.54 M	1.72 %	✓	29.9	50.2	33.2	11.3	22.9	41.3

Table 2: Quantitative comparison with different fine-tuning methods on UIIS dataset using ConvNeXt V2 backbones. Red indicates the best performance, and blue indicates the second-best. * denotes the trainable parameters in backbones.

4. Additional Fine-Tuning Comparisons

To complement the comparison of fine-tuning strategies in the main paper, which compare fine-tuning methods on Swin Transformer, we provide additional results for ConvNeXt V2 backbones. This comparison follows the same experimental setup, ensuring that parameter efficiency and segmentation performance are fairly evaluated across different architectures. As shown in Table 2, ERA achieves the highest mAP of 29.9, surpassing full fine-tuning by 1.4 mAP while requiring only 1.72% of the trainable parameters. The results reinforce the effectiveness of ERA across different model architectures, demonstrating its ability to efficiently adapt to varying feature representations while maintaining strong segmentation performance. These findings further validate ERA as an efficient alternative to traditional

full fine-tuning, significantly reducing computational overhead while maintaining state-of-the-art segmentation performance across different network backbones.

5. Effectiveness of Each Component in BARDecoder

To validate the effectiveness of BARDecoder, we ablate its two primary components: the Multi-Stage Gated Refinement Network (MSGRN) and the Depthwise Separable Upsample (DSU), as shown in Table 3. MSGRN progressively refines the top-level feature map F_4 using aligned lower-level features F_1 to F_3 through ROIAlign and gated attention. This selective refinement enhances spatially informative regions and improves boundary localization. Replacing MSGRN with standard convolutional fusion reduces performance by 3.1 mAP, highlighting the benefit of progressive attention-based refinement. DSU substitutes bilinear upsampling with multi-scale depthwise convolutions and pixel shuffle, enabling efficient reconstruction of fine structures often degraded in underwater scenes. Substituting DSU with bilinear upsampling leads to a 1.2 mAP drop, confirming its role in preserving spatial detail. These results support the design choices in BARDecoder and demonstrate their contribution to segmentation accuracy and boundary quality.

Method	**mAP**	**AP$_{50}$**	**AP$_{75}$**	**AP$_S$**	**AP$_M$**	**AP$_L$**	**Params**
Mask R-CNN	28.2	46.6	32.1	9.5	23.4	39.6	106.75 M
Replace MSGRN module	28.5	49.7	31.0	10.0	21.8	40.6	113.93 M
Replace DSU module	30.4	49.9	32.6	10.5	**24.1**	42.0	114.34 M
Full model (Ours)	**31.6**	**52.0**	**33.6**	**10.7**	24.0	**45.0**	114.44 M

Table 3: Ablation of BARDecoder. Swin-B backbone and 1× training schedule is adopted.

6. Effectiveness of Each Component in Environment-Robust Adapter

Table 4 presents an ablation study of the two main components in the proposed Environment-Robust Adapter (ERA): the Multi-Scale Feature Extraction (MSFE) module and the Environmental Adaptation (EA) module. MSFE is inspired by the Inception architecture and captures degradation patterns across multiple spatial frequencies using parallel convolutions with diverse receptive fields. This helps the model handle underwater degradations such as turbidity, color distortion, and reduced visibility. Removing MSFE causes a 1.4 mAP drop, indicating its importance in robust feature extraction. The EA module improves adaptability by applying channel-wise attention and pixel-wise gated modulation. It generates environmental embeddings that reflect local degradation conditions and modulates features accordingly. Removing EA results in a 0.7 mAP drop, confirming the benefit of adaptive feature modulation. Together, MSFE and EA form a lightweight and effective adapter for underwater robustness. These findings validate the complementary roles of both modules in enhancing representation under challenging conditions.

Method	mAP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	Params
Mask R-CNN	28.2	46.6	32.1	9.5	23.4	39.6	106.75 M
w/o MSFE module	30.2	49.4	33.5	**10.7**	23.7	41.7	112.37 M
w/o EA module	30.9	50.9	33.5	10.3	23.8	43.5	114.25 M
Full model (Ours)	**31.6**	**52.0**	**33.6**	**10.7**	**24.0**	**45.0**	114.44 M

Table 4: Ablation of Environmental Robust Adapter. EA: Environmental Adaptation.

7. Comparison with Laplacian-Based Boundary Loss

To highlight the effectiveness of the proposed Boundary-Aware Cross-Entropy (BACE) loss, we conduct a comparative experiment with a Laplacian-based variant that replaces the range projection term $\boldsymbol{A}^{\dagger}\boldsymbol{A}$ with a fixed edge-detection filter. This baseline mimics heuristic boundary enhancement methods such as those used in WaterMask, where a Laplacian kernel is applied to approximate boundary regions. As shown in Table 5, this substitution leads to a 1.3 mAP drop, demonstrating that the heuristic approach is less effective than our principled formulation. Unlike fixed filters that apply uniform weights to edge regions, BACE is grounded in range-null space decomposition, a well-established concept in inverse problem theory. This formulation explicitly separates low-frequency components, which capture global structure, from high-frequency components, which emphasize ambiguous and detailed boundaries. Such decomposition provides more meaningful supervision, enabling the model to better distinguish between confident interior regions and uncertain edges during learning. By aligning the loss design with the mathematical structure of the segmentation task, BACE improves boundary accuracy in a theoretically sound and empirically validated manner. These findings confirm that the range-null space decomposition offers a more effective strategy for guiding boundary learning than traditional edge-based losses.

Method	mAP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	Params
Mask R-CNN	28.2	46.6	32.1	9.5	23.4	39.6	106.75 M
w/ Laplace convolution	30.3	50.3	32.6	10.6	**24.0**	42.5	114.44 M
Full model (Ours)	**31.6**	**52.0**	**33.6**	**10.7**	**24.0**	**45.0**	114.44 M

Table 5: Comparison between BACE Loss and a Laplacian-based variant by replacing $A^T A$ in Eq. (15) with a Laplace convolution.

8. Impact of the Number of Refine Blocks

We investigate the effect of varying the number of Refine Blocks on segmentation performance and computational efficiency, as shown in Table 6. Increasing from two to three blocks improves mAP from 31.0 to 31.6, demonstrating the benefits of deeper feature refinement. However, further increasing to four or five blocks results in diminishing returns, with increased computational cost. Thus, we adopt three Refine Blocks as the optimal configuration, balancing segmentation quality and inference speed.

# Refine Block	mAP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	Params
2	31.0	51.2	**34.5**	10.2	**24.6**	43.6	114.20 M
3	**31.6**	**52.0**	33.6	**10.7**	24.0	**45.0**	114.44 M
4	30.0	50.4	33.2	9.8	23.0	42.7	114.87 M
5	31.0	50.6	33.7	**10.7**	24.1	44.3	115.69 M

Table 6: The impact of the number of Refine Blocks. **Bold:** best.

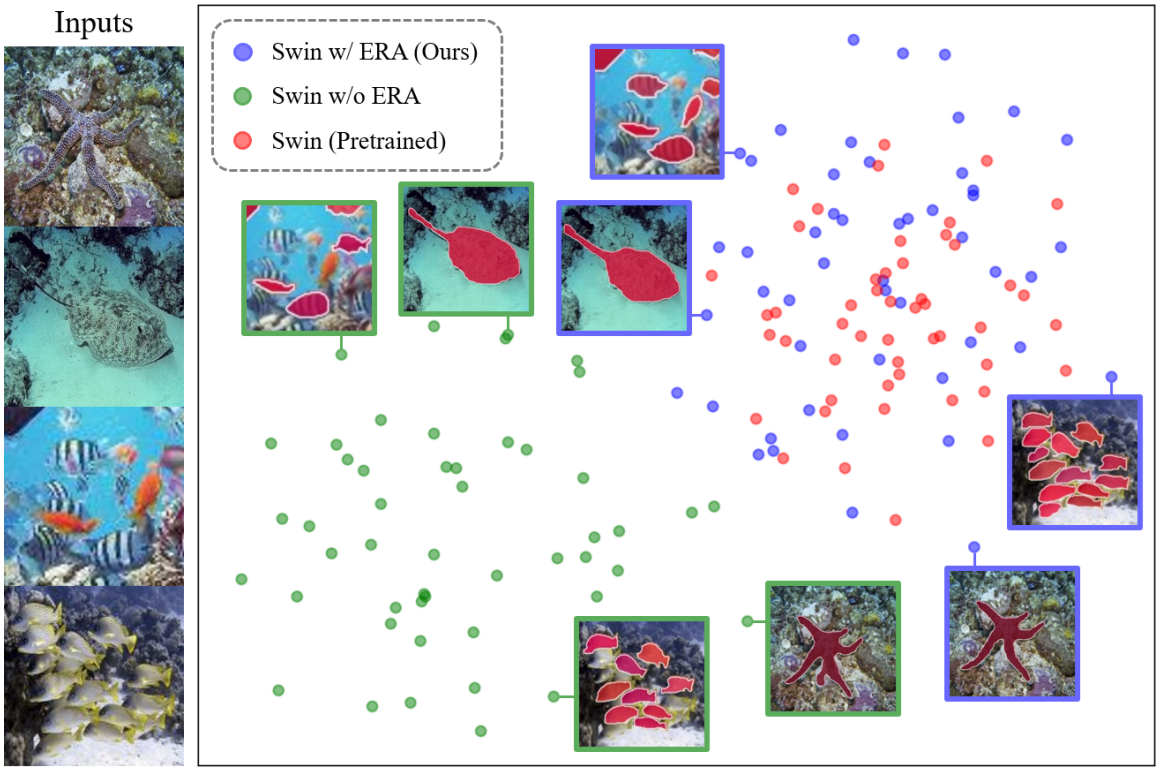

Figure 1: The t-SNE visualization of feature distributions illustrating the effectiveness of ERA in aligning underwater features with terrestrial distributions. "Swin w/ ERA (Ours)" (blue points) shows significant overlap with "Swin (Pretrained)," (red points) bridging the gap between underwater and land-based environments, while "Swin w/o ERA" (green points) remains distinct due to underwater-specific degradations.

9. Knowledge Transfer of ERA

The purpose of ERA is to adapt underwater image features by learning priors of various underwater degradations, allowing pretrained models on land-based data to process underwater imagery effectively. To evaluate the transferability of ERA, we present t-SNE visualizations in Figure 1. The figure shows that "Swin w/o ERA" (green points), which uses full fine-tuning, captures underwater-specific features with distributions affected by underwater degradation (e.g., color distortions, low visibility). In contrast, "Swin (Pretrained)" (red points) retains ImageNet Deng et al. (2009) features suited for terrestrial environments, demonstrating a distinct distribution. However, "Swin w/ ERA (Ours)"

(blue points) achieves significant overlap with "Swin (Pretrained)," (red points) illustrating the effectiveness of the proposed ERA in dynamically adapting underwater features to align with terrestrial feature distributions by mitigating underwater degradation effects. This alignment is crucial for stabilizing training and capturing robust features in challenging underwater conditions. These results highlight the capability of ERA to adapt models for underwater segmentation, effectively bridging the gap between underwater and land-based visual characteristics.

10. The Impact of the Projection Ratio γ in ERA.

We assessed the effect of the projection ratio γ in ERA using Swin Transformer and ConvNeXt V2 backbones (see Table 7). For Swin Transformer, $\gamma = 2$ achieved the highest mAP of 31.6, while $\gamma = 4$ balanced performance across multiple metrics. Higher ratios, such as $\gamma = 8$, led to declines in mAP. For ConvNeXt V2, $\gamma = 4$ yielded the best mAP of 32.3, with $\gamma = 2$ following closely behind. These results suggest that a lower γ is optimal for Swin Transformer, while moderate values work best for ConvNeXt V2. We used the best configurations in all experiments, highlighting the importance of selecting an appropriate γ for optimal ERA performance in underwater segmentation tasks.

Projection Ration (γ)	**mAP**	**AP$_{50}$**	**AP$_{75}$**	**AP$_S$**	**AP$_M$**	**AP$_L$**	**Params**	**mAP**	**AP$_{50}$**	**AP$_{75}$**	**AP$_S$**	**AP$_M$**	**AP$_L$**	**Params**
	Swin Transformer							**ConvNeXt V2**						
2	**31.6**	**52.0**	33.6	**10.7**	24.0	**45.0**	114.44 M	31.8	51.0	34.9	11.0	**24.0**	45.4	120.05 M
4	30.6	50.3	**34.5**	10.4	**24.2**	42.9	109.38 M	**32.3**	**51.4**	**36.3**	10.9	23.8	**45.7**	112.46 M
8	29.3	48.7	32.9	10.0	23.9	41.5	107.18 M	31.4	50.5	35.3	**11.3**	23.6	44.8	109.15 M

Table 7: The impact of the projection ratio $\boldsymbol{\gamma}$ in ERA. Results are obtained using the Swin Transformer and ConvNeXt V2 backbones with a 1× training schedule. **Bold:** best, underline: 2nd.

11. Learnable Environment Embeddings

We evaluated the effect of varying the number of learnable environmental embeddings, testing configurations with 4, 8, 16, and 32 embeddings. As shown in Table 8, the 16-embedding configuration achieved the highest mAP of 31.6 and the best AP$_{50}$ of 52.0, indicating strong accuracy. The 4-embedding setup yielded an mAP of 30.9, while 8 embeddings attained the highest AP$_{75}$ of 33.8 with a competitive mAP of 30.6. The 32-embedding configuration slightly underperformed with an mAP of 30.6. These results suggest that 16 embeddings strike the optimal balance for accuracy under varying underwater conditions. Figure 2 visualizes representative learnable environmental degradation prior embeddings ($\boldsymbol{E}_8$, $\boldsymbol{E}_{10}$, $\boldsymbol{E}_{13}$, $\boldsymbol{E}_{15}$), showing their complementary roles in mitigating challenges such as turbidity and reduced visibility, enabling adaptation to diverse underwater environments.

# Environment Embeddings	mAP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
4	<u>30.9</u>	50.7	33.0	**10.8**	24.4	43.5
8	30.6	50.8	**33.8**	10.3	<u>24.5</u>	43.0
16	**31.6**	**52.0**	<u>33.6</u>	<u>10.7</u>	24.0	**45.0**
32	30.6	<u>51.1</u>	33.5	10.6	**24.9**	42.4

Table 8: The impact of the number of learnable environment embeddings. Evaluation is conducted using the Swin Transformer backbone with a 1× training schedule. **Bold:** best, <u>underline</u>: 2nd.

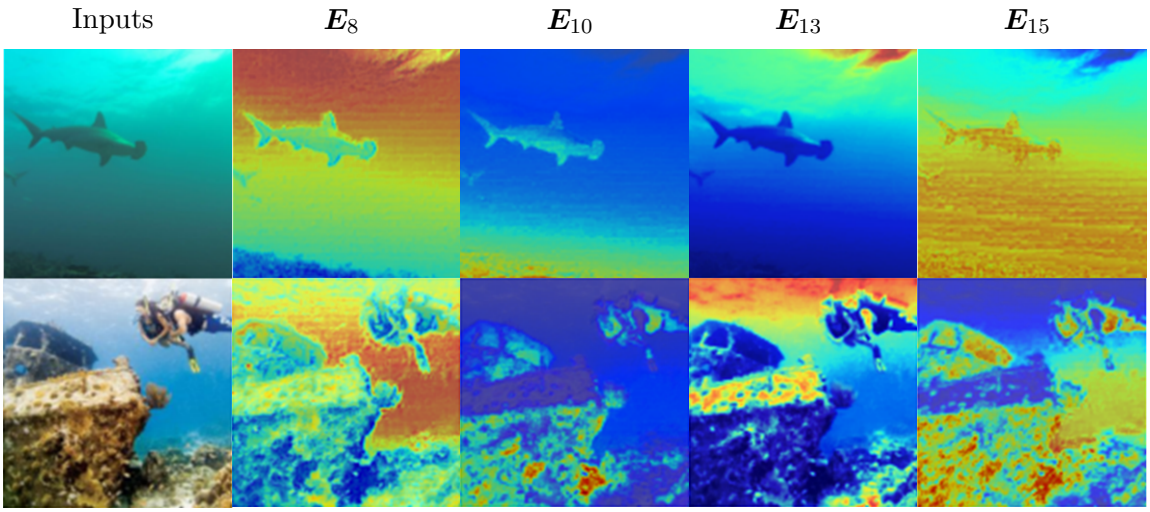

Figure 2: Visualization of learnable environmental degradation prior embeddings showing the effectiveness of our adaptation mechanism in addressing underwater degradations. Different embeddings (E_8, E_{10}, E_{13}, E_{15}) complement each other in adapting to diverse underwater conditions.

References

Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162, 2018.

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

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

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

Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. Pointrend: Image segmentation as rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9799–9808, 2020.

Shijie Lian and others. Diving into underwater: Segment anything model guided underwater salient instance segmentation and a large-scale dataset. In *ICML*, 2024.

Shijie Lian, Hua Li, Runmin Cong, Suqi Li, Wei Zhang, and Sam Kwong. Watermask: Instance segmentation for underwater imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1305–1315, 2023.

Xinlong Wang, Rufeng Zhang, Tao Kong, Lei Li, and Chunhua Shen. Solov2: Dynamic and fast instance segmentation. *Advances in Neural information processing systems*, 33: 17721–17732, 2020.