17th International Conference on Machine Vision Applications (MVA)
Fully Online, July 25–27, 2021.

# Crack Segmentation for Low-Resolution Images using Joint Learning with Super-Resolution

**Yuki Kondo** **Norimichi Ukita**
Toyota Technological Institute
{sd18037, ukita}@toyota-ti.ac.jp

## Abstract

*This paper proposes a method for crack segmentation on low-resolution images. Detailed cracks on their high-resolution images are estimated by super resolution from the low-resolution images. Our proposed method*[*1] *optimizes super-resolution images for the crack segmentation. For this method, we propose the Boundary Combo loss to express the local details of the crack. Experimental results demonstrate that our method outperforms the combinations of other previous approaches.*

## 1 Introduction

Detecting cracks in buildings and pavements (Fig. 1) is important for preventive maintenance. Since these cracks are widely distributed over the huge surfaces of objects, it is not easy to inspect all of these cracks.

This inspection can be achieved by pixelwise binary segmentation in observed images. Various Semantic Segmentation (SS) models such as Fully-Convolutional Networks (FCN) [1] are proposed with deep learning. Compared to general SS, crack segmentation is more challenging, even though it is just binary classification. The reasons are as follows: (1) class imbalance due to the small number of crack areas, (2) ambiguous and complex boundaries, and (3) the small difference in contrast between the crack and its surrounding pixels due to shadows and so on. The class imbalance problem is particularly serious, and is called the "all-black" issue because the generated images turn black.

Furthermore, previous methods for crack segmentation are validated with High-Resolution (HR) image datasets such as CRACK500 [2]. However, in general real-world applications, images may be Low-Resolution (LR) due to several restrictions. For example, the camera performance is limited in high-temperature environments such as furnaces. Images must be taken from a distance by a drone for safe flight.

This paper proposes a method for crack segmentation on LR images, which provides the same quality of segmentation results as those obtained on HR images by utilizing Super Resolution (SR). In addition to automatic inspection, the reconstructed SR image is also useful for human-interpretable inspection by experts. Sample results are shown in Fig. 1.

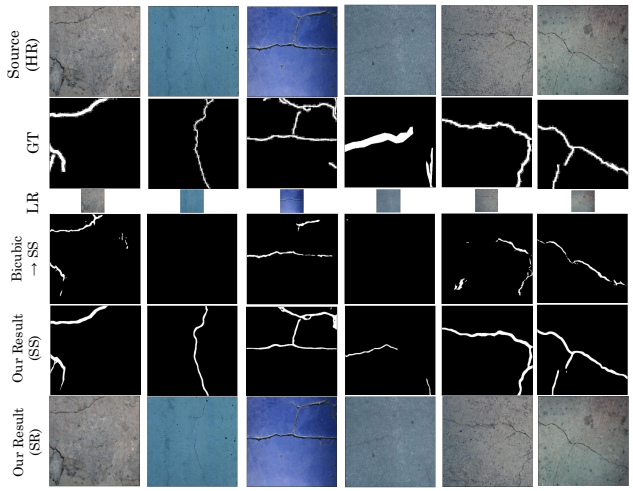


Figure 1: Samples in the Khanhha dataset [3], and crack segmentation results on these images.

## 2 Related Work

**Image Super Resolution:** Image SR [4–6] reconstructs a HR image from its LR image. Among recent SR methods [7–11], DBPN [12, 13] achieves accurate reconstruction by iteratively projecting and back-projecting the interrelationships between LR-HR image pairs. In this paper, DBPN is used as a baseline.
**Semantic Segmentation:** SS [14–17] identifies the semantic classes of pixels. FCN [1] consisting of only convolutions employs intermediate local features to improve the details of segmentation. Based on FCN, U-Net [18] uses skip connections to further improves the local details. Our method uses U-Net as a baseline.
**Crack Detection:** FPHBN [19] solves the all-black issue so that erroneous results in global features are reflected in local features. In CrackGAN [1], DCGAN [20] is used to suppress the all-black issue. While these methods [1, 19] cope with the all-black issue by designing the network architecture, a carefully-designed loss function can also resolve the all-black issue. In [21], the Dice loss [22] improves segmentation near crack boundaries [23]. The Weighted Cross Entropy (WCE) [18] can also be useful [24].
**End-to-End Joint Learning:** For example, in automatic speech recognition, preprocessing enhancement and recognition are optimized jointly for improving robustness [25–27]. In computer vision, TDSR [28] im-

*1 We release code for CSSR at https://github.com/Yuki-11/CSSR

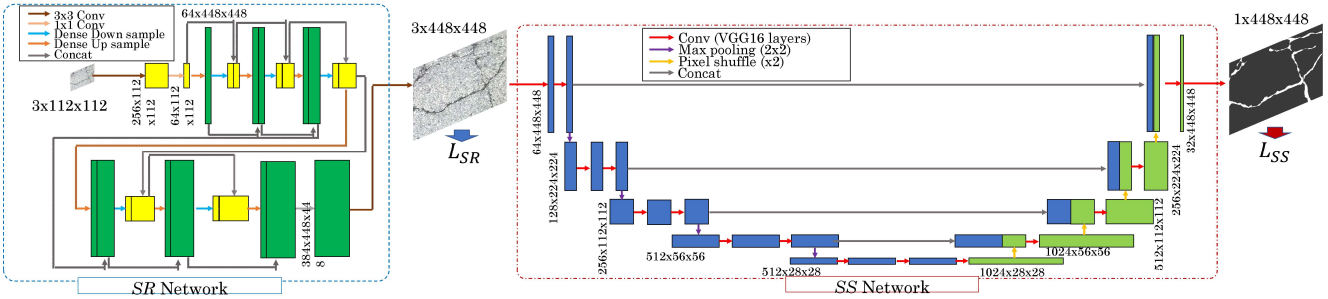


Figure 2: Implementation of CSSR in LR images. HR segmentation is done by end-to-end learning of SR and SS networks with the Boundary Combo loss proposed in Sec. 3.2.

proves object detection by joint learning of SR and detection. Our proposed method is based on TDSR for improving crack segmentation in LR images.

## 3 Crack Segmentation with SR (CSSR)

### 3.1 Network Architecture

The architecture of our method is shown in Fig. 2. A HR image enlarged by SR is fed into an SS network, which predicts the HR segmentation result.

**SR Network:** Our method uses the 6-stage dense-DBPN [12], where each stage has one upsampling layer and one downsampling layer. By stacking these stages, projection and inverse projection are performed iteratively for augmenting SR features.

**SS Network:** While our SS network is based on U-Net, it has two differences from the original. First, $3 \times 3$ conv layers in VGG-16 [29] pre-trained by ImageNet [30] are used in the encoder because the effectiveness of fine-tuning is validated in U-Net [31] as well as in many other networks. Secondly, the $2 \times 2$ deconv layer used for upsampling is replaced by the pixel shuffle layer, which is faster and more accurate.

### 3.2 Boundary Combo Loss

To cope with the all-black issue, we pay attention to medical image segmentation where local and tiny objects such as cancers and tumors are detected, and the same class-imbalance problem is critical.

The Boundary loss [23], proposed for medical image segmentation, focuses on local properties around crack boundaries. Specifically, the Boundary loss computes the distance-weighted 2D area between the ground-truth crack and its estimated one, which becomes zero in the ideal estimation, as follows:

$$\begin{aligned} D(\partial G, \partial S) &= \int_{\partial G} ||q_{\partial S}(p) - p||^2 dp \\ &\simeq 2 \int_{\Delta S} D_G(p) dp \\ &= 2\Big( \int_{\Omega} \phi_G(p) s(p) - \int_{\Omega} \phi_G(p) g(p) dp \Big), \end{aligned} \quad (1)$$

where $G$ and $S$ denote the pixel sets of the ground-truth crack and its estimated one, respectively. $p$ and $q_{\partial S}(p)$ denote a point on boundary $\partial G$ and its corresponding point on boundary $\partial S$, respectively. $q_{\partial S}(p)$ is an intersection between $\partial S$ and a normal of $\partial G$ at $p$. $\Delta S = (S/G) \cup (G/S)$ is the mismatch part between $G$ and $S$. $D_G(p)$ is the distance map from $G$. $s(p)$ and $g(p)$ are binary indicator functions, where $s(p) = 1$ and $g(p) = 1$ if $p \in S$ and $p \in G$, respectively. $\phi_G(q)$ is the level set representation of boundary $\partial G$: $\phi_G = -D_G(q)$ if $q \in G$, and $\phi_G = D_G(q)$ otherwise. $\Omega$ denotes a pixel set in the image. The second term in Eq. (2) is omitted as it is independent of the network parameters. By replacing $s(p)$ by the network softmax outputs $s_\theta(p)$, we obtain the Boundary loss function below:

$$\mathcal{L}_B = \int_{\Omega} \phi_G(p) s_\theta(p) dp \quad (2)$$

Since the Boundary loss falls into local minima due to class imbalance, it is employed with other losses such as CE-based losses (e.g., WCE) and region-based losses (e.g., Dice). In the literature, however, only simple combinations with two losses are explored. This paper further verifies the effectiveness of more complementary losses. We pay attention to the Combo loss [32], which is the weighted sum of WCE and Dice, that copes with imbalance. The Combo loss might improve tiny region segmentation because (1) WCE explicitly adjust the balance between false-positives and false-negatives and (2) Dice avoids local minima due to imbalance.

Based on the above discussion, we propose two loss functions. The first one consists of the Boundary loss and the Combo loss. This loss is called the Boundary Combo (BC) loss expressed by Eq. (3). In the second one, Dice in the Combo loss is replaced by the Generalized Dice (GDice) loss [33], which is called the Generalized Boundary Combo (GBC) loss in Eq. (4).

$$\mathcal{L}_{BC} = \alpha \mathcal{L}_B + (1 - \alpha) \left[ (1 - \gamma) \mathcal{L}_{Dice} + \gamma \mathcal{L}_{WCE}(w_{pos}) \right] \quad (3)$$

$$\mathcal{L}_{GBC} = \alpha \mathcal{L}_B + (1 - \alpha) \left[ (1 - \gamma) \mathcal{L}_{GDice} + \gamma \mathcal{L}_{WCE}(w_{pos}) \right] \quad (4)$$

where $\alpha, \gamma, w_{pos} \in [0, 1)$ are coefficients. $\alpha$ with the initial value of 0.01 is increased by 0.01 per epoch, $\gamma$

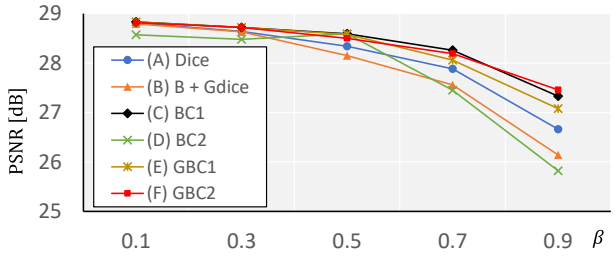


Figure 3: PSNR scores by different losses.

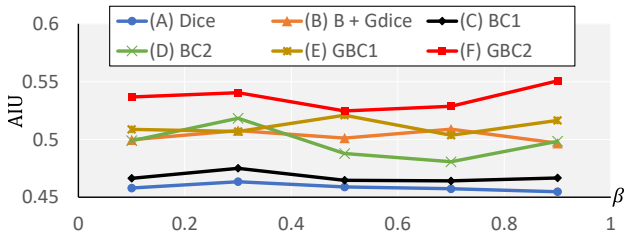


Figure 4: AIU scores by different losses.

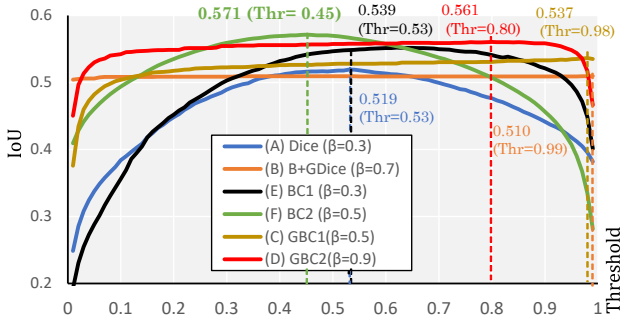


Figure 5: IoU vs. confidence threshold by different losses.

balances $\mathcal{L}_{Dice}$ and $\mathcal{L}_{WCE}$, and $w_{pos}$ is the weight for positive class.

### 3.3 Losses for Joint Learning

L1 loss $\mathcal{L}_{SR}$ (Eq. (5)) and the segmentation loss function $\mathcal{L}_{SS}$ (Eqs. (3, 4)) are used for SR and SS, respectively.

$$\mathcal{L}_{SR} = \frac{1}{|\boldsymbol{I}|} \sum_{i \in \boldsymbol{I}} \|I_{SR}(i) - I_{HR}(i)\|, \tag{5}$$

where $I_{SR}$ and $I_{HR}$ denote the reconstructed SR image and its ground-truth HR image, respectively. $\boldsymbol{I}$ is a set of pixels in a HR image. With the weighted sum of these two loss functions, $\mathcal{L}$ for the entire model is defined as expressed by Eq. (6):

$$\mathcal{L} = (1-\beta)\mathcal{L}_{SR} + \beta(\mathcal{L}_{SS}), \tag{6}$$

where $\beta \in [0, 1]$ is a weight. $\mathcal{L}_{SR}$ and $\mathcal{L}_{SS}$ in Fig. 2 show where the two loss functions are calculated.

## 4 Experimental results

**Datasets:** We used the Khanhha dataset [3], which contains the following original datasets: CRACK500 [2], GAPs [34], CFD [35], AEL [36], crack-tree200 [37], DeepCrack [38], CSSC [39] and CrackForest [35]. As shown in Fig. 1, a variety of images and annotations (e.g., thin and thick) are included. We used this combined dataset for validating the robustness to the change in image and annotation characteristics.

For making the Khanhha dataset, images with arbitrary sizes were cropped from each image in the original datasets, and resized to 448 × 448 pixels. In total, the Khanhha dataset consists of 9,603 training images and 1,695 test images. In our experiments, 481 validation images were excluded from the training images. These images are regarded as HR images. Each HR image is downsampled using Bicubic to 112 × 112 pixels. This LR image is fed into each method.

**Evaluation:** The quality of the SR image is evaluated by PSNR. The metrics for evaluating SS are IoU and AIU [19]. AIU is an averaged IoU overall confidence thresholds for predictive segmentation. Since the dataset includes images with no crack and IoU cannot be computed properly in such images, IoU is computed with a small constant $\epsilon$ in its denominator for numerical stability.

**Training details:** The learning rate is 1e-5, the batch size is 6, iteration is 100k, and the optimizer is Adam [40] . Data augmentation is done by random mirroring, photometric distortion, and random cropping with random scales.

### 4.1 Loss function and $\beta$ Analysis

We evaluate the change in segmentation accuracy with different loss functions and $\beta$ used for CSSR. Six loss functions are evaluated: (A) Dice [21, 22] ($\mathcal{L}_{SS} = \mathcal{L}_{Dice}$), (B) Boundary + GDice [23] ($\mathcal{L}_{SS} = (1-\alpha)\mathcal{L}_B + \alpha\mathcal{L}_{GDice}$), (C) BC1 (Eq. (3) with $w_{pos} = 19/20, \gamma = 1/2$), (D) BC2 (Eq. (3) with $w_{pos} = 1/2, \gamma = 1/2$), (E) GBC1 (Eq. (4) with $w_{pos} = 19/20, \gamma = 1/2$), and (F) GBC2 (Eq. (4) with $w_{pos} = 1/2, \gamma = 1/2$). Note that "(C) and (D)" and "(E) and (F)" are our BC and GBC losses with different parameters, respectively.

Figure 3 and 4 show the change depending on $\beta$. The accuracy of SR decreases monotonically with all losses except BC2 in Fig. 3. Our BC1, GBC1, and GBC2 are superior to the others. The optimal $\beta$ for SS differs among the losses in Fig. 4. Among the maximum $\beta$ values of all six losses, BC2, GBC1, and GBC2 are better than the other conventional losses.

The relationship between the confidence threshold and IoU when optimal $\beta$ is set for each loss is shown in Fig. 5. In this result, the maximum value of IoU is also higher than that of the conventional losses, suggesting that the threshold-independent property of IoU in GBC is caused by GDice loss.

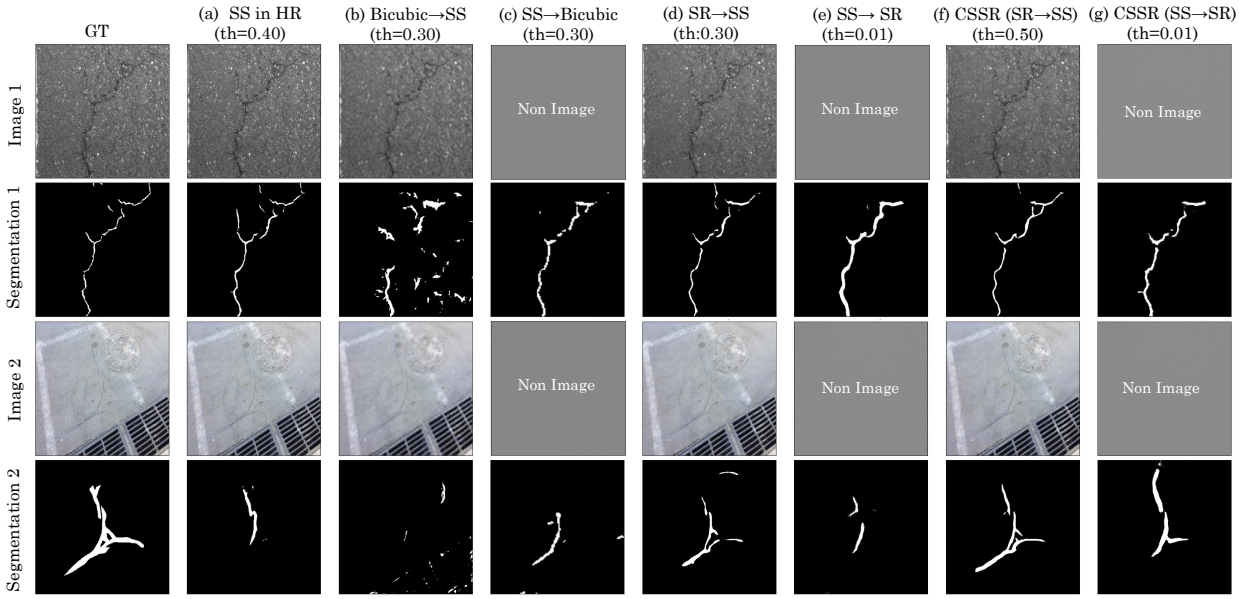


Figure 7: Qualitative comparison of our models with other models.

### 4.2 Model Comparison

Figure 6 and Table 1 show the results of the model comparison. The models are (a) SS in HR: HR image is fed into SS network, (b) Bicubic → SS: Image enlarged from LR images by Bicubic is fed into SS network, (c) SS → Bicubic: Segmentation image is magnified by Bicubic, (d) SR → SS: LR image is fed into SR and SS networks trained independently, (e) SS → SR: LR image is fed into SS and SR networks trained independently, (f) CSSR (SR → SS): Our method with jointly trained SR and SS networks shown in Fig. 2, and (g) Inverse CSSR (SS → SR). Note that LR images are fed into all models except (a). All of these models use the BC2, which got the best IoU in Fig. 5.

AIU of (f) CSSR is quite close to that of (a). The SR quality of (f) is also good (i.e., the second-best in Table 1), while the best score is obtained by (d). That is natural because the SR network is trained only for SR in (d), while (f) is trained jointly with SS. In Fig. 6, the max IoU of (f) is sufficiently high compared to (a); 0.571 vs. 0.587. Furthermore, (g) obtains a higher AIU than (a), while (g) cannot give us SR images.

SR images reconstructed by these models are shown in Fig. 7. (f) CSSR provides segmentation results closer to GT. In particular, subtle cracks are also detected by (f). This detail reproduction is vital in the inspection. This is because, from the viewpoint of fracture mechanics [41], crack length, direction and aperture are vital factors for inspection. Detailed segmentation by CSSR can reduce the risk of overestimating a structure's lifetime from the viewpoint of practical use. Also, the SR images of (f) are reliable enough to be used for human-interpretable inspection by experts.

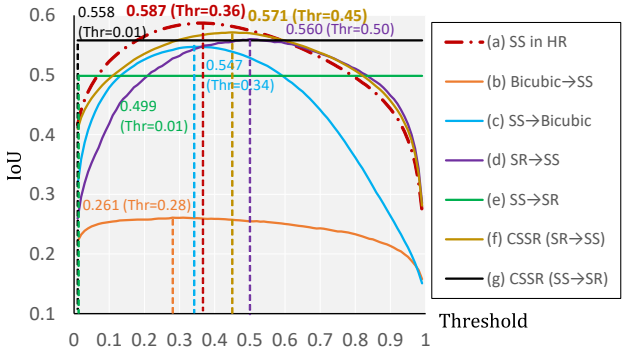


Figure 6: IoU vs. confidence threshold comparison among different models for crack segmentation.

Table 1: Quantitative evaluation of models. Red and blue indicate the best and the second best, respectively.

| Model | PSNR [dB] | AIU |
|---|---|---|
| (a) SS in HR | - | 0.525 |
| (b) Bicubic → SS | 27.6 | 0.243 |
| (c) SS → Bicubic | - | 0.447 |
| (d) SR → SS | **29.0** | 0.496 |
| (e) SS → SR | - | 0.499 |
| (f) CSSR (SR → SS) | **28.5** | **0.518** |
| (g) CSSR (SS → SR) | - | **0.558** |

## 5 Conclusion

We proposed Crack Segmentation with SR (CSSR) for HR crack segmentation from LR images. Joint learning with the Boundary Combo loss allows CSSR to be comparable to segmentation on HR images. Future work includes experiments on more realistic scenarios (e.g., using realistic blur kernels). CSSR can be extended to videos [42,43] for more robust segmentation. This work was supported by JSPS KAKENHI Grant Number 19K12129.

## References


[1] Kaige Zhang, Yingtao Zhang, and Heng-Da Cheng. CrackGAN: Pavement Crack Detection Using Partially Accurate Ground Truths Based on Generative Adversarial Learning. *TITS*, 22(2):1306–1319, 2020.

[2] Lei Zhang, Fan Yang, Yimin Daniel Zhang, and Ying Julie Zhu. Road crack detection using deep convolutional neural network. In *ICIP*, 2016.

[3] Khanhha. Crack segmentation, 2020. https://github.com/khanhha/crack_segmentation.

[4] Radu Timofte et al. Ntire 2018 challenge on single image super-resolution: Methods and results. In *NTIRE (CVPRW)*, 2018.

[5] Shuhang Gu et al. Aim 2019 challenge on image extreme super-resolution: Methods and results. In *AIM (ICCVW)*, 2019.

[6] Kai Zhang et al. Ntire 2020 challenge on perceptual extreme super-resolution: Methods and results. In *NTIRE (CVPRW)*, 2020.

[7] Assaf Shocher, Nadav Cohen, and Michal Irani. "zero-shot" super-resolution using deep internal learning. In *CVPR*, 2018.

[8] Shady Abu Hussein, Tom Tirer, and Raja Giryes. Correction filter for single image super-resolution: Robustifying off-the-shelf deep super-resolvers. In *CVPR*, 2020.

[9] Andreas Lugmayr, Martin Danelljan, Luc Van Gool, and Radu Timofte. Srflow: Learning the super-resolution space with normalizing flow. In *ECCV*, 2020.

[10] Kai Zhang, Luc Van Gool, and Radu Timofte. Deep unfolding network for image super-resolution. In *CVPR*, 2020.

[11] Majed El Helou, Ruofan Zhou, and Sabine Süsstrunk. Stochastic frequency masking to improve super-resolution and denoising networks. In *ECCV*, 2020.

[12] Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. Deep back-projection networks for super-resolution. In *CVPR*, 2018.

[13] Muhammad Haris, Greg Shakhnarovich, and Norimichi Ukita. Deep back-projection networks for single image super-resolution. *arXiv*, 2019.

[14] Hengshuang Zhao, Yi Zhang, Shu Liu, Jianping Shi, Chen Change Loy, Dahua Lin, and Jiaya Jia. PSANet: Point-wise spatial attention network for scene parsing. In *ECCV*, 2018.

[15] Hang Zhang, Han Zhang, Chenguang Wang, and Junyuan Xie. Co-occurrent features in semantic segmentation. In *CVPR*, 2019.

[16] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, 2017.

[17] Jun Fu, Jing Liu, Yuhang Wang, and Hanqing Lu. Stacked deconvolutional network for semantic segmentation. *arXiv*, 2017.

[18] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

[19] Fan Yang, Lei Zhang, Sijia Yu, Danil Prokhorov, Xue Mei, and Haibin Ling. Feature pyramid and hierarchical boosting network for pavement crack detection. *TITS*, 21(4):1525–1535, 2020.

[20] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *ICLR*, 2016.

[21] Chuncheng Feng, Hua Zhang, Haoran Wang, Shuang Wang, and Yonglong Li. Automatic pixel-level crack detection on dam surface using deep convolutional network. *Sensors*, 20:2069, 2020.

[22] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *3DV*, 2016.

[23] Kervadec Hoel, Bouchtiba Jihene, Desrosiers Christian, Granger Eric, Dolz Jose, and Ben Ayed. Boundary loss for highly unbalanced segmentation. In *PMLR*, 2019.

[24] Dimitris Dais, İhsan Engin Bal, Eleni Smyrou, and Vasilis Sarhosis. Automatic crack classification and segmentation on masonry surfaces using convolutional neural networks and transfer learning. *Autom. Constr.*, 125:103606, 2021.

[25] Bin Liu, Shuai Nie, Shan Liang, Wenju Liu, Meng Yu, Lianwu Chen, Shouye Peng, and Changliang Li. Jointly adversarial enhancement training for robust end-to-end speech recognition. In *ISCA*, 2019.

[26] Bin Liu, Shuai Nie, Yaping Zhang, Dengfeng Ke, Shan Liang, and Wenju Liu. Boosting noise robustness of acoustic model via deep adversarial training. In *ICASSP*, 2018.

[27] Cunhang Fan, Jiangyan Yi, Jianhua Tao, Zhengkun Tian, Bin Liu, and Zhengqi Wen. Gated recurrent fusion with joint training framework for robust end-to-end speech recognition. *TASLP*, 29:198–209, 2020.

[28] Muhammad Haris, Greg Shakhnarovich, and Norimichi Ukita. Task-driven super resolution: Object detection in low-resolution images. *arXiv*, 2018.

[29] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[30] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[31] Vladimir Iglovikov and Alexey Shvets. Ternausnet: U-net with VGG11 encoder pre-trained on imagenet for image segmentation. *arXiv*, 2018.

[32] Saeid Asgari Taghanaki, Yefeng Zheng, S Kevin Zhou, Bogdan Georgescu, Puneet Sharma, Daguang Xu, Dorin Comaniciu, and Ghassan Hamarneh. Combo loss: Handling input and output imbalance in multi-organ segmentation. *Comput Med Imaging Graph*, 75:24–33, 2019.

[33] Carole H. Sudre, Wenqi Li, Tom Vercauteren, Sébastien Ourselin, and M. Jorge Cardoso. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In *MICCAI*, 2017.

[34] Markus Eisenbach, Ronny Stricker, Daniel Seichter,

Karl Amende, Klaus Debes, Maximilian Sesselmann, Dirk Ebersbach, Ulrike Stoeckert, and Horst-Michael Gross. How to get pavement distress detection ready for deep learning? a systematic approach. In *IJCNN*, 2017.

[35] Yong Shi, Limeng Cui, Zhiquan Qi, Fan Meng, and Zhensong Chen. Automatic road crack detection using random structured forests. *TITS*, 17(12):3434–3445, 2016.

[36] Rabih Amhazand, Sylvie Chambon, Jérôme Idier, and Vincent Baltazart. Automatic crack detection on two-dimensional pavement images: An algorithm based on minimal path selection. *TITS*, 17(10):2718–2729, 2016.

[37] Qin Zou, Yu Cao, Qingquan Li, Qingzhou Mao, and Song Wang. Cracktree: Automatic crack detection from pavement images. *Pattern Recognition Letters*, 33(3):227–238, 2012.

[38] Yahui Liu, Jian Yao, Xiaohu Lu, Renping Xie, and Li Li. Deepcrack: A deep hierarchical feature learning architecture for crack segmentation. *Neurocomputing*, 338:139–153, 2019.

[39] Liang Yang, Bing Li, Wei Li, Liu Zhaoming, Guoyong Yang, and Jizhong Xiao. Deep concrete inspection using unmanned aerial vehicle towards cssc database. In *IROS*, 2017.

[40] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[41] T.L. Anderson. *Fracture Mechanics: Fundamentals and Applications, Third Edition*. Taylor & Francis, 2005.

[42] Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. Recurrent back-projection network for video super-resolution. In *CVPR*, 2019.

[43] Muhammad Haris, Greg Shakhnarovich, and Norimichi Ukita. Space-time-aware multi-resolution video enhancement. In *CVPR*, 2020.