Segmentation of cracks

Sandhya Kumari
T22109

Akanksha Sharma
T22110

Somit Gond
B21138

***Abstract*—Life of any non-living thing is determined through its structural properties. Detecting cracks is an important measure to do so. Crack detection can be done for buildings, roads, metals, pipes etc. Early detection of cracks in some cases (like roads, bridges etc.) can prevent several accidents as respective measures could be taken early. Traditionally, crack detection is done manually. This involves a person who visits the site and check it. This type of crack detection consumes time, needs periodic visits and is dependent on the eyes of the officer. Hence it involves various errors which can mislead about the actual situation. Therefore other methods such as traditional image processing methods, deep learning, machine learning, transfer learning are adopted. The approach we are using involves segmentation which clearly distinguishes between crack and background.**

I. Introduction

Detection of cracks is important to know about how long a thing would work. One can take precautions and be ready for future miss-happenings that can occur due to cracks. Large damages such as breaking of bridges, roads can be avoided by providing maintenance timely. This can avoid severe accidents and loss of life. Therefore timely recognition of cracks is crucial. Traditionally, the process is done manually by appointing a person for it. The person or officer visits the site and inspect. This consumes manpower and time. Even the method involves high human error. So, instead of manually detecting cracks, detection of cracks is done using images of the objects. Here we are detecting cracks on pavement. The images of pavement contain two surfaces i.e. either well pavement surface or crack. In order to deal with the problem, the approach involves segmentation for detection of cracks. The image data contains cracks with its background. If one can clearly distinguishes the two things, one can easily detect cracks. Hence problem of distinguishing cracks with background can be converted to segmentation task. Our objective turns to be as if an image of pavement is provided to model, it produces segmented image where cracks and background can be easily differentiated.

II. Related Work

A. Semantic Segmentation

Semantic segmentation is pixel wise labeling of various classes of an image.With the advancement of deep learning model [1] and rise of CNN architecture segmentation tasks has been possible to segment various classes pixel wise. Most of the segmentation model consist of encoder and decoder model. Encoder model down-samples the images and decoder model again up-samples the images taken from the encoder and finally assign probability value of a particular pixel belonging to a particular class.

B. Attention Mechanism

For better understanding between the feature maps extracted from CNN architecture attention mechanism is also used to get attention maps between spatial and channel dimension.This work [2] proposes a convolutional Block Attention Module (CBAM) architecture. This architecture provides the attention map which is multiplied with the input features for feature refinement.It can be integrated with any CNN architecture.

C. Residual Refinement

Deep learning architecture like CNN uses various convolutional and maxpooling layer causing repeated subsampling which decreases the image resolution.This work [3] proposes a feedforward neural network with pyramid pooling module for saliency detection.

III. Proposed Methodology

The problem of crack segmentation is approached by following methods:

A. Edge Detection

In this approach traditional image processing method i.e. edge detector is used. The edges are detected using Canny Edge Detector and Sobel filter. The image is taken and edge detector methods are implemented. This results in segmented image which contains background in black and cracks in white.

B. UNet

In this approach UNet architecture i.e. encoder, bottleneck and decoder is used. The encoder captures the features and provide it to the decoder. The decoder uses these extracted features for segmentation. The bottleneck is placed between encoder and decoder which provide compressed representation for features extracted by encoder. This compressed representation is provided to the decoder.

1) Data Pre-processing: The size of input image is 448 X 448 X 3 while UNet architecture takes input of size 128 X 128 X 3. Therefore input image is resize to 128 X 128 X 3.

2) Encoder: It contains two convolution layers each with 64 filters, kernel size is 3 X 3 and relu as activation function. After convolution layers, max-pooling of size 2 X 2 having dropout of 0.3 is there. The resize image is inputed to it.

3) Bottleneck: It contains two convolution layers each with 1024 filters, kernel size is 3 X 3 and relu as activation function. The output of encoder is inputed to it.

4) Decoder: It contains a convolution layer with 64 filters and kernel size 3 X 3 having relu as its activation function. The output of convolution is concatenated with the output of convolution of encoder. This concatenated output is fed to further network which contains two convolution layers each with 64 filters, kernel size is 3 X 3 and relu as activation function. After convolution layers, max-pooling of size 2 X 2 having dropout of 0.3 is there. This output is then goes to convolution layer having 2 output channels and softmax as activation function with kernerl size 1 X 1. The output of bottleneck is inputed to the decoder.
The output of decoder contains 2 channels which represent probability of each pixel belonging to each class (i.e. crack or non-crack). These probabilities results in segmented image for two classes.

C. UNet with DAM

In this approach UNet with DAM is used. The architecture of UNet used here is same as that of architecture of UNet proposed in earlier part. The output of bottleneck of UNet is now given to DAM. DAM (Dual Attention Module) extracts information which help in distinguishing cracks and non-crack part completely. The output of DAM is given to the decoder.

1) Data Pre-processing: The size of input image is 448 X 448 X 3 while UNet architecture takes input of size 128 X 128 X 3. Therefore input image is resize to 128 X 128 X 3.

2) DAM (Dual Attention Module): DAM consists of CBAM (Convoltional Block Attention Module) and Squeeze and Excitation [4] Module. CBAM extracts spatial as well as channel information while Squeeze and Excitation Module generates local discriptor which provides information about different channels. Inclusion of both the modules enhance the information which helps in distingshing crack and non-crack part completely.
The output of decoder contains 2 channels which represent probability of each pixel belonging to each class (i.e. crack or non-crack). These probabilities results in segmented image for two classes.

IV. Experimental Evaluation

A. Implementation details

1) Environment: The experiment was performed using python 3.11 and tensorflow. NVIDIA RTX 3080 12GB RAM system is used for training and testing.

2) Training setting: For traing SGD model is used and catagorical cross entropy loos function is used. We trained for 100 epochs with batch size of 20 with learning rate of 1e-3.

3) Inference setting: After training model was applied on kaggle crack segmentation dataset. Original image was resized to 128*128 size for testing purpose.

B. Details of the Dataset

The dataset contains 11,200 images of pavement. These images contain both cracks and non-cracks images along with their respective masked images. The size of crack and non-crack image is 448 X 448 X 3. The size of masked image is 448 X 448.

C. Results

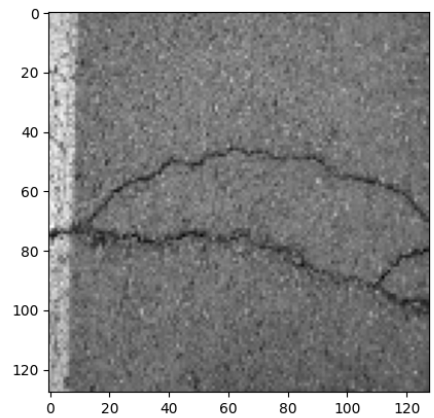

Fig. 1. Test image of pavement crack

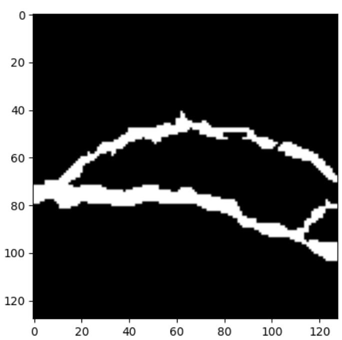

Fig. 2. Segmented image from SOTA method (UNet+DAM)

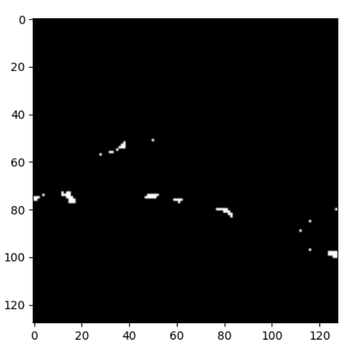

Fig. 3. Segmented image output from UNet model

Fig. 4. Crack Image for Edge Detection

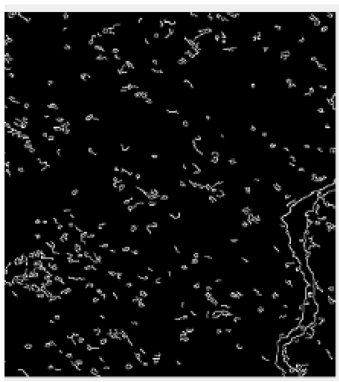

Fig. 5. Edge Detection Output

Architecture	mIoU
Edge Detector	0.057
UNet	0.0115
UNet+DAM	0.2

TABLE I
MEAN INTERSECTION OVER UNION FOR DIFFERENT ARCHITECTURES

V. CONCLUSION

The detection of cracks is important for determining the need of actions required in order to prevent miss-happenings. The detection can be done by various methods. Segmentation is one of the way to detect cracks. Traditional method involve edge detection which detect cracks as edges. This method contains noise which need to be removed for its betterment. Deep learning approaches, UNet and UNet with DAM, shows better performance than traditional method. UNet provides better segmentation than edge detection but lacks for some cracks. There is need for addition of more information to segment all kind of cracks. Such information is provided by DAM in UNet with DAM architecture. Hence UNet with DAM performs well among the methods used.

REFERENCES

[1] C. Liu, C. Zhu, X. Xia, J. Zhao, and H. Long, "Ffedn: Feature fusion encoder decoder network for crack detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 15 546–15 557, 2022.

[2] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[3] T. Wang, A. Borji, L. Zhang, P. Zhang, and H. Lu, "A stagewise refinement model for detecting salient objects in images," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[4] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.