Deepcrack – A Deep Learning Approach for Image Based Crack Prediction

R. Venkateshwarlu
Associate Professor
Dept. of Computer Science and Engineering
Jyothishmathi Institute of Technology and Science
(JNTUH)
Karimnagar, Telanagana, India
rudrarapu.venkateshwarlu@jits.ac.in

Erram Akshaya Reddy
UG Student
Dept. of Computer Science and Engineering
Jyothishmathi Institute of Technology and Science
(JNTUH)
Karimnagar, Telanagana, India
akshayareddyerram@gmail.com

Edla Bhavani
UG Student
Dept. of Computer Science and Engineering
Jyothishmathi Institute of Technology and Science
(JNTUH)
Karimnagar, Telanagana, India
bhavaniedla7@gmail.com

Rimsha Mohammadi
UG Student
Dept. of Computer Science and Engineering
Jyothishmathi Institute of Technology and Science
(JNTUH)
Karimnagar, Telanagana, India
rims.md22@gmail.com

Sunkari Rahul
UG Student
Dept. of Computer Science and Engineering
Jyothishmathi Institute of Technology and Science
(JNTUH)
Karimnagar, Telanagana, India
rahulsunkari887@gmail.com.in

R.Shashank
UG Student
Dept. of Computer Science and Engineering
Jyothishmathi Institute of Technology and Science
(JNTUH)
Karimnagar, Telanagana, India
radandishashank@gmail.com

Abstract— Crack detection is critical for the safety and longevity of civil infrastructures like roads, bridges, and buildings. However, the conventional approach for crack detection involves manual visual inspection, which is not only time-consuming but also vulnerable to errors. To overcome these challenges, the present project proposes an automated system for crack detection, severity analysis, and cause prediction using a deep learning approach, named "DeepCrack: A Deep Learning Approach for Image-Based Crack Prediction." The proposed system is implemented using a full-stack approach with a React-based frontend and a Flask-based backend using the Python programming language. For image preprocessing, the system utilizes the OpenCV and Pillow libraries. For crack detection, the system utilizes the U-Net++ model implemented using the PyTorch library with various image augmentation techniques like rotation, flipping, scaling, and brightness adjustment. After crack detection, the system predicts the severity of the crack based on the crack area and distribution. Additionally, the system predicts the cause of the crack, including fatigue, moisture, thermal expansion, corrosion, and overloading.

Keywords— Deep Learning, Crack Detection, U-Net++, Crack Segmentation, Computer Vision, Structural Health Monitoring, Data Augmentation, Severity Analysis, Image Processing, Infrastructure Inspection.

I. INTRODUCTION

Civil infrastructures such as roads, bridges, and buildings play a vital role in transportation and human activities. Over time, these infrastructures tend to crack due to environmental factors, heavy load, fatigue, etc. If these cracks are not recognized early enough, they could cause serious damage to the structures. In the traditional approach, crack detection is done manually by engineers. This approach is time-consuming, costly, and inaccurate.

Recently, the development of computer vision and deep learning techniques has introduced a better solution for crack detection systems. Deep learning techniques such as CNNs can effectively analyze the images and identify the crack patterns. These techniques provide a better solution for crack detection systems compared to the traditional approach.

DeepCrack: The main goal of the DeepCrack project is to develop a crack detection system that uses deep learning techniques to analyze the images for crack detection. The system will be able to analyze the images, identify the crack areas, analyze the severity of the crack, and provide the possible causes. The results will be displayed through a website to help users understand the results easily.

PROBLEM STATEMENT

The project “Deepcrack: a deep learning approach for image based crack prediction” intends to carry out an analysis of the cracks using images. The project aims to develop a deep learning system for the prediction of cracks based on the images.

EXISTING SYSTEM

Relies on manual inspection, which is a time-consuming, expensive, and non-scalable process.

- Prone to human error, where small or hidden cracks may be overlokked.
- Basic image processing or traditional machine learning techniques with low accuracy.
- Not capable of accurate segmentation or localization of cracks in real-world scenes.

PROPOSED SYSTEM

The proposed system, DeepCrack, utilizes various deep learning methods for the automatic detection of cracks from images. The system allows users to upload an image using the web interface. The uploaded image is processed and analyzed using the deep learning algorithm for the detection of cracks. Once the cracks are identified, the system determines the severity of the cracks and the probable cause, such as fatigue, moisture, or stress. The final results are provided in the form of images where the cracks are highlighted.

II. LITERATURE REVIEW

A. in [1] - A novel algorithm for semantic segmentation is performed using the learning of the deep deconvolution network. The network on top of the convolutional layers adopted from the VGG 16-layer net. The deconvolution network uses the deconvolution and unpooling layers, which are used for pixelwise class label detection and segmentation masks. The trained network is applied for each proposal in the input image, and the final semantic segmentation map is created by combining the results from each of the proposals in a simple manner. The proposed algorithm overcomes the shortcomings of the existing algorithms using the fully convolutional network approach using the integration of the deep deconvolution network and the proposal-wise prediction. The proposed segmentation algorithm typically performs well in the detection of detailed structures and objects of various sizes naturally. The performance of the network is outstanding for the PASCAL VOC 2012 dataset, and the best accuracy of 72.5% is achieved for the methods trained without using the Microsoft COCO dataset using ensemble with the fully convolutional network

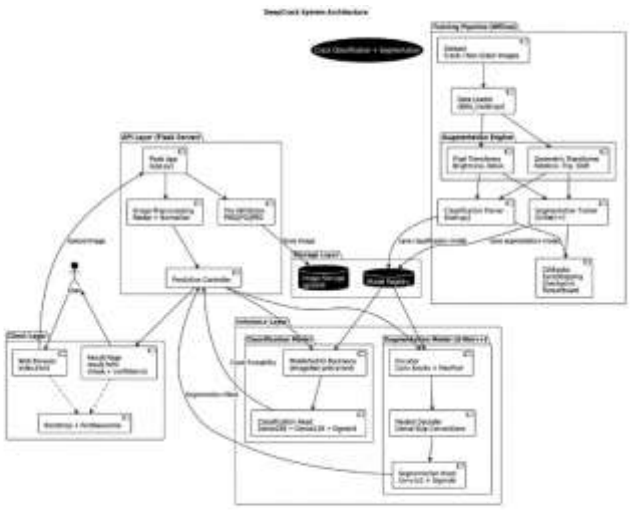
B. in [2] - - Automatic detection of pavement cracks is an important task in transportation maintenance for driving safety assurance. However, it is still a challenging task because of the inhomogeneity of the intensity of cracks and the complexity of the background, such as low contrast with the surrounding pavement and even the presence of shadows with similar intensity. Inspired by the recent success of the application of deep learning in solving various computer vision problems and medical applications, in this paper, a novel method based on the application of deep learning in automatic detection of pavement cracks is proposed. Quantitative evaluations carried out on the data set consisting of 500 images of size 3264 χ 2448, captured by a low-cost smart phone, show that the proposed method of learning deep features using the proposed deep learning framework is significantly better in terms of performance compared to existing hand-craft methods.

C. in [3] – It proposes an automated approach for the inspection of nuclear power plant components during the remote visual examination. For the automated approach, the detection of low-contrast cracks that are often surrounded by or may even be present in textures with similar visual appearances, including welding, scratches, and grind marks, is necessary. A crack segmentation approach for the videos of the remote visual examination is proposed by aggregating the pixel classification confidence of the crack from the various frames of the videos with different illumination conditions. The proposed approach is validated using a dataset of 685 pixel-level ground truth images with 37 cracks from the videos of the remote visual examination. The proposed approach shows significant improvement over the hand-crafted feature-based approach and the convolutional neural network-based approach by 9%.

D. in [4] - For supervised contour detection methods, typically many training images are needed for satisfying performance. Yet, sometimes such a large amount of annotated images might not be available or might require an extremely time-consuming labeling process. It aims at exploring the possibility of using semi-supervised learning (SSL) for contour detection with competitive accuracy using an extremely small number of training images (only three images are needed). A semi-supervised structured ensemble learning approach for contour detection using structured random forests (SRF) is proposed. To apply the structured random forests to the unlabeled images, an effective approach for sparse representation is provided, where the inherent structure within the image patches can be captured using the finding of the compact and discriminative low-dimensional subspace representation in an unsupervised way, so that the large number of unlabeled patches with their estimated structured labels can be exploited for improving the performance of node splitting in the structured random forests. The sparsity issue is revisited, and a novel and fast sparse coding algorithm is proposed for improving the learning efficiency. To the best of the authors' knowledge, for the first time, SSL is applied for contour detection, and the superiority of the proposed approach is evaluated using

E. in [5] – It discusses a computational approach for edge detection. The success of the approach is dependent on the specification of a complete and exhaustive list of goals for the computation of edge points. These goals must be specific enough to constrain the desired behavior of the detector while assuming the solution as little as necessary. The detection and localization criteria for a class of edges, and the present mathematical forms for these criteria as functionals on the operator impulse response. A third criterion is proposed for the assurance of the existence of only one response for a single edge. The criteria in numerical optimization for the derivation of detectors for several image features, including the detection of step edges is discussed. On specializing the analysis for the detection of step edges, it is derived that a natural uncertainty principle exists between detection and localization performance, which are the two main goals. By using the principle, the derivation is made that a single operator shape is optimal for any scale. The optimal detector has a simple approximate implementation in which edges are marked at maxima in the gradient magnitude of a Gaussian smoothed image. A general method, termed feature synthesis, for the fine-to-coarse integration of information from operators at different scales is discussed. The performance of the step edge detector is significantly better when the operator point spread function is extended along the edge.

III. SYSTEM ARCHITECTURE



The system architecture of the DeepCrack system is structured in such a way that the system integrates the web interface, the backend server, the deep learning system, and the offline training pipeline. The system architecture comprises various layers that include the client layer, the API layer, the preprocessing layer, the storage layer, the inference layer, and the training pipeline layer.

1. CLIENT LAYER

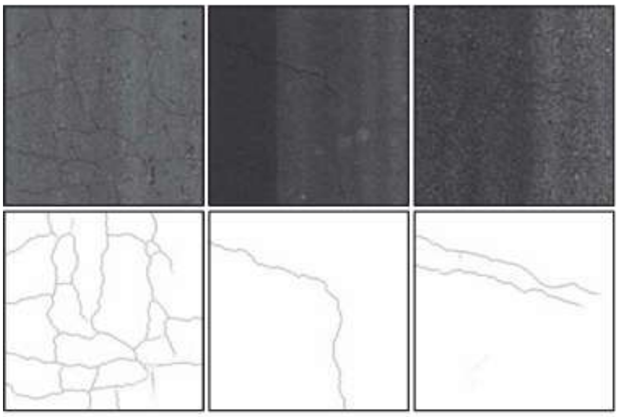
The client layer represents the part of the system that allows the user and the system to interact. The users interact with the system through the application, where the users can upload images of the infrastructure surfaces such as the road, the bridge, or the concrete wall. The system allows the users to upload the images, and the system shows the results of the image analysis through an interactive user interface. The system shows the results of the crack segmentation, the crack probability, and the crack region highlighted in an easy-to-understand format.

2. API LAYER

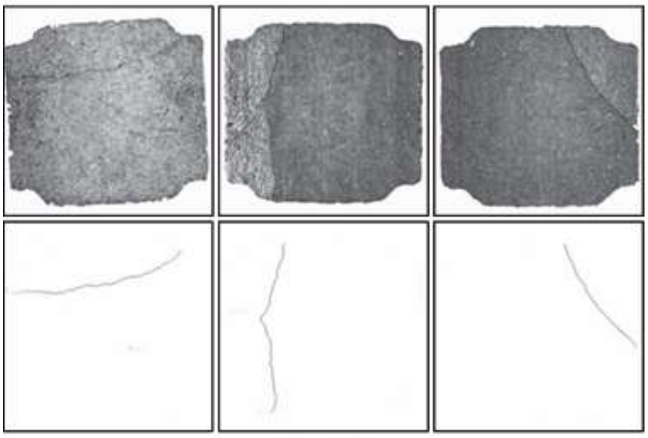
The API layer comprises the system that uses the Flask-based server for the system. The API layer allows the system to interact with the other layers of the system. The system receives the uploaded image from the user and performs the validation of the uploaded file. The system checks whether the uploaded image is in the required format such as PNG, JPG, or JPEG. If the uploaded image is valid, the system forwards the image for the preprocessing stage.

3. PREPROCESSING AND EDGE DETECTION LAYER

In the preprocessing layer, the uploaded image is subjected to various preprocessing stages that allow the system to detect the cracks within the image accurately. The system resizes the image and normalizes the image so that the image can meet the requirements of the input for the deep learning system. The system also uses edge detection methods such as the Canny edge detection method to highlight the edges within the uploaded image. The edge detection method allows the system to highlight the crack regions within the uploaded image.



CRKWH100



Stone331

4. STORAGE LAYER

The storage layer is used to store the uploaded images and the trained models. The uploaded images are stored temporarily and are used during the processing stage. The trained models are stored in the model registry. The correct trained models are used during the inference stage. The system is able to effectively manage the models used during the inference stage.

5. INFERENCE LAYER

The inference layer consists of the deep learning models used to perform the crack detection and segmentation.

Crack Classification Model :

The classification module is used to determine if the uploaded image contains any cracks. The image is passed through the MobileNetV2 network with an ImageNet pre-trained backbone to extract the image features. The features are then passed through fully connected layers to produce the probability score.

Crack Segmentation Model (U-Net++):

To perform the detailed segmentation of the cracks in the image, the U-Net++ segmentation model is used. The architecture consists of an encoder that uses convolution and pooling layers to extract the image features. The nested decoder is used to reconstruct the segmentation map from the image features. The segmentation head is used to produce the segmentation map highlighting the crack locations in the image.

IV. PROPOSED METHOD IMPLEMENTATION AND ALGORITHMS

The proposed DeepCrack system is an automated system for crack detection, severity assessment, and cause prediction using various deep learning and image processing approaches. The system takes an image input from the user using the web interface and processes the input image using the backend, programmed using the Python language and the Flask framework. The input image is first subjected to preprocessing using the OpenCV and Pillow libraries, including resizing the image to a specific size, normalization, and removing noise from the image. The image is then fed into the U-Net++ architecture, programmed using the PyTorch framework, for pixel-level crack segmentation. The segmented image provides the binary crack map for the input image.
The system further performs post-processing and analysis using the ratio of the total number of crack pixels and the total number of pixels in the image for determining the severity of the cracks. The severity level is classified into low, medium, and high using specific thresholds. The system uses the rule-based approach for determining the cause of the cracks, including fatigue, moisture, expansion, corrosion, and stress. The final results are provided using the React framework for the user interface. The results include the crack overlay image, severity level, cause prediction, and recommendations.

1. Crack Detection using U-Net++:
Input: Image III
Output: Crack Mask MMM
- •Upload the image and perform preprocessing operations such as resizing and normalization.
- •Input the image into the pre-trained U-Net++ model.
- •Probability map is obtained from the image.
- •Thresholding is applied to the image to produce the final crack mask.

2. Severity Estimation:
Input: Crack Mask M
- •Calculate total pixels and crack pixels.
- •Calculate percentage.
- •Determine the severity level as Low, Medium, or High.

3.Cause Prediction(Rule-Based) :
- •Low: Minor surface damage.
- •Medium: Moisture or thermal impact.
- •High: Structural stress or foundation.

4.Overall System Pipeline :
- •User inputs an image.
- •Image is preprocessed.
- •Crack detection is performed.
- •Severity and cause are estimated.
- •Results are displayed with crack visualization..

V. TOOLS & LIBRARIES

TOOLS	LIBRARIES
Python 3.9/ 3.10	PyTorch & torchvision
Flask	OpenCV & Pillow
Streamlit	NumPy

VI. RESULT ANALYSIS AND DISCUSSION

Fig 1

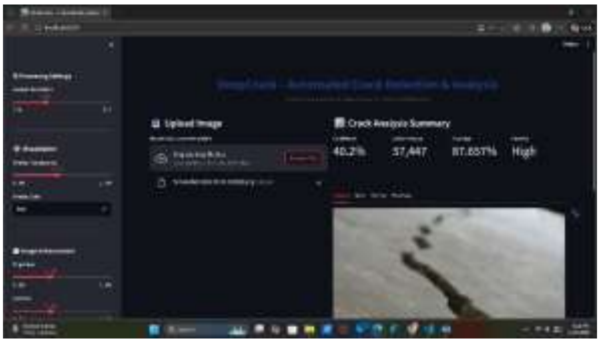


Fig 2

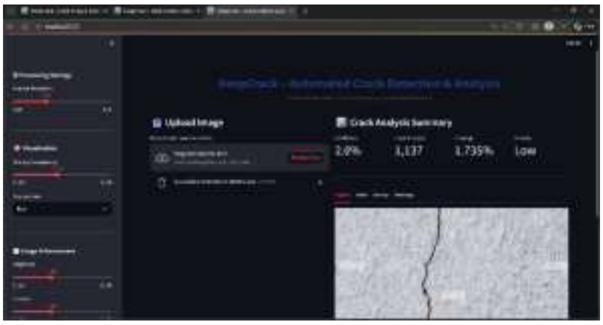


Fig 3

The performance of the DeepCrack system was evaluated using various images of civil infrastructure surfaces such as roads, walls, and concrete surfaces with visible cracks. The system performed well in processing the uploaded images using the preprocessing and deep learning-based segmentation methods for crack detection. The cracks are highlighted using the crack mask overlay on the original image. The system was able to calculate the area of the cracks and determine the severity of the cracks using various categories such as low, medium, and high severity. The system was also able to provide the predicted cause of the cracks, such as fatigue, moisture, thermal expansion, corrosion, and excessive loading.

From the results, the proposed system is able to identify cracks and perform meaningful analysis within a short amount of time. The system is able to visualize the crack overlays, thus improving the interpretability of the results. The results are able to provide users with a good understanding of the extent of the damage. Although the system performs well for the majority of the crack images, there are other factors that could influence the accuracy of the results. The performance of the system indicates that the application of deep learning for crack detection could improve the inspection of infrastructure surfaces significantly. The system could perform the inspection within a short amount of time using the automated approach compared to the traditional approach.

VII. CONCLUSION

The DeepCrack project has successfully proven the application of deep learning approaches for image-based crack detection and prediction. The system uses CNN-based approaches for the detection of complex crack patterns from surface images using image processing techniques. The results indicate that the application of deep learning approaches can enhance the efficiency of crack detection compared to other image processing approaches. The application of such approaches is not only restricted to structural health monitoring; rather, the approach can also be applied for the real-time analysis of cracks in various fields such as civil engineering, aerospace, and manufacturingKey Contributions

REFERENCES

[1] Hyeonwoo Noh, Seughoon, Bohyung Han, "Learning Deconvolution Network for Semantic Segmentation," (2015).

[2] L. Zhang, F. Yang, Y. D. Zhang, and Y. J. Zhu, "Road crack detec tion using deep convolutional neural network," in IEEE International Conference on Image Processing, 2016, pp. 3708–3712.

[3] S. J. Schmugge, L. Rice, J. Lindberg, R. Grizziy, C. Joffey, and M. C. Shin, "Crack segmentation by leveraging multiple frames of varying illumination," in IEEE Winter Conference on Applications of Computer Vision,, 2017, pp. 1045–1053.

[4] Z. Zhang, F. Xing, X. Shi, and L. Yang, "Semicontour: A semi supervised learning approach for contour detection," IEEE conference on computer vision and pattern recognition, pp. 251–259, 2016.

[5] J. F. Canny, "A computational approach to edge detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 8, no. 6, pp. 679–698, 1986..