June Moh Goo¹, Xenios Milidonis¹, Alessandro Artusi¹, and Carlo Ciliberto¹

¹Affiliation not available

March 07, 2024

Abstract

Detecting and segmenting cracks in infrastructure, such as roads and buildings, is crucial for safety and cost-effective maintenance. In spite of the potential of deep learning, there are challenges in achieving precise results and handling diverse crack types. With the proposed dataset and model, we aim to enhance crack detection and infrastructure maintenance. This study proposes a novel approach termed Hybrid-Segmentor, which uses a convolutional neural network path that is well-suited for extracting fine-grained local features and a transformer path to extract global features that benefit from understanding the overall structure. This hybrid method makes the model more generalizable to various shapes, surfaces, and sizes of cracks. To achieve a balanced computational cost, the study incorporates efficient self-attention in the transformer path and introduces a comparatively simpler decoder compared to the complexity of the two encoder paths. This combination strategically optimizes the extraction of global and local features while maintaining computational efficiency. The model was trained using a combined binary cross entropy and Dice loss function on a large refined dataset of 12,000 crack images generated from 13 publicly available datasets. Our studies demonstrate that the model efficiently utilizes convolutional layers and transformers to extract local and global features. Hybrid-Segmentor outperforms existing benchmark models across 5 quantitative metrics (accuracy 0.971, precision 0.804, recall 0.744, F1-score 0.770, and IoU score 0.630), achieving state-of-the-art status. Finally, through careful qualitative analysis, we show that the model is capable of addressing discontinuities, detecting small non-crack regions, handling low-quality images, and detecting crack contours more accurately than existing models.
Abstract—Detecting and segmenting cracks in infrastructure, such as roads and buildings, is crucial for safety and cost-effective maintenance. In spite of the potential of deep learning, there are challenges in achieving precise results and handling diverse crack types. With the proposed dataset and model, we aim to enhance crack detection and infrastructure maintenance. This study proposes a novel approach termed Hybrid-Segmentor, which uses a convolutional neural network path that is well-suited for extracting fine-grained local features and a transformer path to extract global features that benefit from understanding the overall structure. This hybrid method makes the model more generalizable to various shapes, surfaces, and sizes of cracks. To achieve a balanced computational cost, the study incorporates efficient self-attention in the transformer path and introduces a comparatively simpler decoder compared to the complexity of the two encoder paths. This combination strategically optimizes the extraction of global and local features while maintaining computational efficiency. The model was trained using a combined binary cross entropy and Dice loss function on a large refined dataset of 12,000 crack images generated from 13 publicly available datasets. Our studies demonstrate that the model efficiently utilizes convolutional layers and transformers to extract local and global features. Hybrid-Segmentor outperforms existing benchmark models across 5 quantitative metrics (accuracy 0.971, precision 0.804, recall 0.744, F1-score 0.770, and IoU score 0.630), achieving state-of-the-art status. Finally, through careful qualitative analysis, we show that the model is capable of addressing discontinuities, detecting small non-crack regions, handling low-quality images, and detecting crack contours more accurately than existing models.

Note to Practitioners—The detection of damaged civil infrastructure is important for the prevention of hazardous situations and maintenance procedures. However, the current manual inspection method is labor-intensive, subjective, and relies heavily on the experience of inspectors. Our Hybrid-Segmentor architecture model has achieved state-of-the-art performance, exceeding previous deep learning models. Our model excels in qualitative analysis through its enhanced precision, outperforming previous models in crack contour detail and reliability across various surfaces and image qualities. However, our model struggles to detect fine cracks in complex patterns and is sensitive to visual distortions such as occlusions and watermarks. With more cities installing surveillance and traffic-monitoring cameras, our model can efficiently detect newly generated cracks in civil infrastructure, which holds promise for ensuring safety and reliability. Our model, aimed at general segmentation, is expected to perform well not only in crack detection but also in tasks such as product inspection, manufacturing quality control, and inspection of other types of infrastructure, offering versatility in various segmentation applications.

Index Terms—Deep Learning Applications, Semantic Segmentation, Convolutional Neural Networks, Transformers, Crack Detection, Crack Dataset

I. INTRODUCTION

THE detection of structural damage in civil infrastructure is crucial in preventing potential hazards and safeguarding lives while facilitating timely maintenance interventions. Cracks in roads, pavements, and buildings pose a serious threat to public safety, causing accidents and damage to vehicles on roads and pavements, and influencing public safety and financial burden on buildings. Traditionally, manual inspections have been used to identify cracks in civil infrastructure, but these methods are labor-intensive, subjective, and prone to human error, resulting in inconsistent results and potential disasters. Therefore, automated crack detection is necessary to provide an objective and highly accurate alternative. Machine learning methods, such as deep learning models, can be used to detect, segment, or classify damage to civil infrastructure, which can be facilitated by the widespread deployment of surveillance and traffic-monitoring cameras. However, training accurate models for crack segmentation is challenging due to a scarcity of high-quality and diverse datasets, which impacts model robustness and generalizability. Our research aims to address this crucial data gap and develop automated crack detection to prevent dangers and reduce financial risks to communities. Progress in this direction could lead to realtime identification of newly developed cracks in the future, ensuring a more dependable and safe utilization of critical concrete structures.

In this paper, we address two main goals: first, we introduce a large dataset for crack segmentation. This effort is aimed at filling a critical gap in the literature on crack detection and ultimately providing a valuable resource to the research community on the topic. Second, we propose a new deep learning architecture, empirically proving its effectiveness on real data. The model is designed to tackle crack detection tasks but has potential application to a wide range of computer vision applications. The main contributions of this work are:

- Combine and refine publicly available crack datasets to create an enhanced and comprehensive crack segmentation dataset.
- Introduce a data refinement methodology to combine publicly available datasets using image processing techniques.
- Develop and evaluate a deep learning-based crack detection model utilizing the merits of both convolutional neural networks and transformers.
- Emphasizes the remarkable ability of the proposed model
to perform effectively across a diverse range of surface types and under challenging imaging conditions, such as blurred images and areas with complex crack contours.

- The code, trained weights of the model, and the refined full dataset for experiments are publicly available and can be accessed here: [https://github.com/junegoo94/Hybrid-Segmentor](https://github.com/junegoo94/Hybrid-Segmentor)

II. BACKGROUND AND PREVIOUS WORK

In this Section we introduce the main Deep Learning tools used for crack detection and then review the state of the art in the domain.

A. Deep Learning Architectures

**Convolutional Neural Networks** CNNs have accelerated deep learning research by detecting patterns in data using convolution. Stacking multiple convolutional layers in CNNs can capture complex and abstract patterns. Various CNNs architectures [1]–[7] have been proposed, achieving performance levels in classification tasks that exceed human capabilities. Moreover, CNNs models extend their task to object detection [8]–[10] and semantic segmentation [11]–[13]. Regarding semantic segmentation tasks, pixel-wise segmentation is a popular technique in deep learning for image segmentation. FCN [13] and U-Net [12] are two commonly used models. FCN processes entire images in one go and is primarily used for semantic segmentation [13]. U-Net is specifically designed for biomedical image segmentation and has a U-shaped structure. It is highly effective in producing precise segmentation’s even with very few training images [12].

**Transformers** After a period of gradual performance improvements, the focus of computer vision research has shifted from CNNs towards transformer models, which originally demonstrated exceptional performance in Natural Language Processing (NLP) [14]. Vision Transformers (ViT) [15] is a subclass of transformers designed for computer vision tasks and treats images as sequences of patches or tokens. Transformers offer a novel approach to understanding visual data. When applied to computer vision [11], [16], [17], the attention mechanism proves essential in addressing the spatial complexities involved in image data [18]. A transformer achieves local and global contextual information by integrating the self-attention mechanism. Consequently, transformers are effective in deriving meaningful features, even in scenarios where interactions span diverse scales and locations within an image. Transformers address the limitations of CNNs by capturing long-range dependencies, offering a solution to be able to understand complex relationships across the input data [18]. As a result, they emerge as powerful tools in computer vision, providing insights into global context and improving interpretability.

Nevertheless, transformers also come with limitations, such as the need for extensive data and training time. To address these limitations, researchers are exploring various approaches, including the use of hybrid models, to complement and enhance the capabilities of transformers in computer vision tasks [19], [20].

B. Crack Segmentation

One of the earliest methods for crack detection was a CNN-based model for pixel-level crack detection using FCN [21]. This approach achieves end-to-end crack detection, significantly reducing training time compared to CrackNet [22], a CNN-based model that was the State-Of-The-Art (SOTA) in 2017 without using a pooling layer. While thin cracks can be accurately predicted across a variety of scenes, further enhancements are needed to capture real-time level predictions. In a similar aspect, DeepCrack [23] improves on the generalization of FCN architecture by incorporating batch normalization and side networks for faster convergence. Additionally, this research proposes the publicly available DeepCrack dataset [23], which enhances crack detection precision across diverse scenes. Cheng et al. propose a full crack segmentation model based on U-Net [24]. Subsequent research further demonstrated that the U-Net is particularly suited for crack segmentation tasks [24]–[27]. Some researchers pinpoint that using classical image classification structures as encoders, pre-trained with data such as ImageNet [28], strengthens feature extraction in crack segmentation networks, enhancing crack detection performance [26].

In addition, various encoder-decoder models have been introduced in the field. Amongst these, DeepCrack2 (not to be confused with ‘DeepCrack’ in [23] bearing the same name; we refer to this model as ‘DeepCrack2’ from this point onwards to avoid confusion) is a deep convolutional neural network designed to facilitate automated crack detection through end-to-end training [29]. It primarily focuses on acquiring high-level features that effectively represent cracks. This approach involves the integration of multi-scale deep convolutional features obtained from hierarchical convolutional stages. This fusion enables the capture of intricate line structures, with finer-grained objects in larger-scale feature maps and more holistic representations in smaller-scale feature maps. DeepCrack2 adopts an encoder-decoder architecture similar to SegNet [30] and employs pairwise feature fusion between the encoder and decoder networks at corresponding scales. DeepCrack2 is one of the most benchmarked models in the crack segmentation community.

Despite the abundance of studies that either employ existing deep learning models or enhance them, these approaches may
III. DATASET

We introduce a large refined dataset with the aim of creating a significantly larger and more diverse resource for crack segmentation data compared to the currently available datasets. Since the existing datasets contain a relatively small number of images compared to other well-known tasks in computer vision, large-scale deep learning models are at a high risk of overfitting in these settings. In contrast to most datasets for crack segmentation that collect data based on a single type of surface, the refined comprehensive dataset includes a wide range of surfaces to enhance the robustness and generalizability of trained models by adapting to the diversity of the real world. Due to the characteristics of some cracks, each image has a small proportion of crack pixels, which could result in a form of class imbalance. To counteract this bias, we employed a data augmentation strategy to increase the number of crack pixels in our dataset.

A. Sub-Dataset Details

In our dataset, we collect 13 open-source datasets, which include different surfaces of pavements, walls, stone, and bricks. Table I shows the details of each publicly available dataset. Aigle-RN, ESAR, and LCMS (AEL Dataset) are the datasets collected by their own acquisition system, which comprises a small number of asphalt crack images acquired in diverse background settings. CRACK500 is a pavement crack dataset with 500 images captured on Temple University’s main campus using smartphones. Each image has a pixel-level annotated binary map. They crop each image into 16 non-overlapping regions. CrackTree260, CRKWH100, CrackLS315 and Stone331 datasets are collected by DeepCrack. CrackTree260 with 260 visible-light road pavement images which is constructed based on the CrackTree206. CRKWH100 containing 100 road pavement images captured by a line-array camera at a 1 mm ground sampling distance, CrackLS315 consisting of 315 road pavement images captured under laser illumination using the same line-array camera specifications, and Stone331 comprising visible-light images of stone surfaces along with masks for precise performance evaluation within the stone surface area. DeepCrack dataset is created as a publicly available benchmark dataset consisting of crack images captured across various scales and scenes, specifically designed to evaluate the performance of crack detection systems. The German Asphalt Pavement Distress (GAPs) dataset, introduced in, addresses the comparability issue in pavement distress research, offering a standardised dataset with 1,969 high-quality gray valued images. It covers various distress classes, including cracks, potholes, and inlaid patches. The images have a resolution of 1,920 × 1,080 pixels with a per-pixel resolution of 1.2 mm × 1.2 mm. To enable pixel-wise crack prediction, 384 images are manually selected from GAPs and annotated, forming the GAPs384 dataset. Masonry is created consisting of images captured from masonry structures, which exhibit intricate backgrounds and a diverse range of crack types and sizes. CrackForest dataset (CFD), one of the most benchmarked datasets, is a labeled collection of road crack images, designed to represent the typical conditions of urban road surfaces. Finally, SDNET2018 is a dataset comprising more than 56,000 images of cracked and non-cracked concrete bridge decks, walls, and pavements, with crack widths ranging from 0.06 mm to 25 mm. Since the dataset does not contain ground truth masks, we use this dataset only for the collection of non-cracked image data.

B. Data Refinement

Datasets have different standards, leading to varying resolutions, distortions, and discontinuity. We address this with cropping, upsampling, and image processing. We drew inspiration from the data refinement process described in the CrackSeg9k. Following their methodology, we conducted our data refinement, incorporating similar techniques and approaches. Furthermore, we also add crack pixels to mitigate class imbalance. A closing method involving dilation followed by erosion is used to handle small holes and thin annotations. For thicker annotations, an erosion method is utilized.

The data refinement process includes several steps. First, the JPG files are converted to PNG files. For AEL datasets, black and white pixels are inverted. We then identify all pixels in an image that are neither black nor white. Next, compare each pixel to a threshold value of 255/2. If a pixel’s value is below this threshold, convert it to black (0); if above, convert it to white (255). This process converts all non-black and non-white pixels to either black or white. Appropriate image processing techniques are applied for each dataset. The images are then cropped to 256 × 256 resolution without overlapping. To address the class imbalance, data that includes more than 5,000 crack pixels is selected and augmented using Gaussian noise and rotation (randomly choosing between 90°, 180°, and 270°). Non-crack data from the SDNet2018 dataset is also added.
TABLE II: Dataset details after refinement, with each distortion type represented by a letter (A-G): A for Inverted ground truth, B for Hairline, C for Small holes, D for Thinness, E for Discontinuity, F for Thickness, and G for Smaller resolution than 256 x 256.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Distortion</th>
<th>Refinement (Kernel Size)</th>
<th>Size</th>
<th>Crack Proportion (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GAPS384</td>
<td>A, B, C</td>
<td>Image Complement Closing (3)</td>
<td>38</td>
<td>1.69</td>
</tr>
<tr>
<td>CRACK500</td>
<td>-</td>
<td>-</td>
<td>114</td>
<td>2.01</td>
</tr>
<tr>
<td>CrackLJ315</td>
<td>D</td>
<td>Closing (3)</td>
<td>102</td>
<td>1.07</td>
</tr>
<tr>
<td>CrackTree260</td>
<td>D, E</td>
<td>Closing (3)</td>
<td>391</td>
<td>1.28</td>
</tr>
<tr>
<td>CRKWH110</td>
<td>D</td>
<td>Closing (3)</td>
<td>59</td>
<td>1.37</td>
</tr>
<tr>
<td>DeepCrack</td>
<td>B, C</td>
<td>Closing (3)</td>
<td>898</td>
<td>4.62</td>
</tr>
<tr>
<td>ESAR</td>
<td>A</td>
<td>Image Complement</td>
<td>34</td>
<td>1.43</td>
</tr>
<tr>
<td>GAP384</td>
<td>F</td>
<td>Erode (3)</td>
<td>1338</td>
<td>2.28</td>
</tr>
<tr>
<td>LCMS</td>
<td>A, B</td>
<td>Image Complement Closing (3)</td>
<td>11</td>
<td>2.33</td>
</tr>
<tr>
<td>Masonry</td>
<td>G</td>
<td>Upscaling - Nearest-neighbors</td>
<td>229</td>
<td>4.58</td>
</tr>
<tr>
<td>SDNet2018</td>
<td>-</td>
<td>-</td>
<td>2208</td>
<td>-</td>
</tr>
<tr>
<td>Stone331</td>
<td>F</td>
<td>Erode (3)</td>
<td>1</td>
<td>0.82</td>
</tr>
<tr>
<td>Augmentation</td>
<td>-</td>
<td>-</td>
<td>2340</td>
<td>13.61</td>
</tr>
</tbody>
</table>

**Total Dataset**: 12000 8.49

Fig. 2: The figure shows the improvement in small holes, discontinuity and thinness of the ground truth after applying appropriate image processing methods.

Table II shows the details of the refined dataset. For each sub-dataset, we manually compared the original ground truth to the corresponding images and then tried to address irregularities respectively. Figure 2 shows how the original ground truth improved after applying image processing methods. Irregularities such as small holes, discontinuity, and thinness were corrected. Furthermore, adding the augmented dataset increased the proportion of the crack pixels by 5.8%, which can mitigate the class imbalance problems. As a result, we created a comprehensive refined dataset with a size of 12,000 images, which is the largest crack dataset to the best of our knowledge.

Table 2

IV. MODEL DESIGN

This section provides an in-depth overview of Hybrid-Segmentor, our end-to-end crack segmentation model. The model pipeline involves a sequential passage of input images through two distinct encoders: the CNN Path and the Transformers Path. Each of these encoders generates 5 multi-scale feature maps, which are then fused together. The fused feature maps are then utilized to produce the final output, as illustrated in Figure 3.

A. CNN Path

It is advantageous to use convolutional layers to identify localized patterns such as edges, textures, and intricate features in the receptive field. With increasing depth, CNNs can extract not only low-level features, such as edges, but also higher-level features, such as abstract shapes [43]. Several CNN models have been proposed to take advantage of this capability. In theory, deeper models are expected to achieve higher performance. As model depth increases, performance may not improve further. To resolve this, residual blocks are proposed as a way to enhance model performance. The basic idea behind the residual block is to add a shortcut connection between the input and output of each layer in the network. In this way, the network is able to learn the residual between the input and output. As a result, the model is prevented from gradient-vanishing, allowing it to have deeper layers.

We use CNN-based ResNet-50 as one of the encoders. In 2015, [3] introduced this model, which stacks 50 layers deep using residual blocks. Currently, ResNet-50 is one of the most popular image classification models.

There are also a number of skip connections in ResNet. Skip connections are connections that skip over some of the layers in the network. The skip connections allow the network to learn features at different levels of abstraction. Furthermore, it has been shown to be effective in transfer learning, where a pre-trained model is used as a starting point. To follow this trend, we incorporate a pretrained ResNet-50 model onto the ImageNet dataset [28]. Besides enabling us to extract features from various datasets, including the crack dataset, this allows for faster convergence as well.

B. Transformer Path

The use of transformers in computer vision has gained recognition for its performance and achievements. Consequently, we apply key ideas from the SOTA model, SegFormer [11], in the field of semantic segmentation. This involves implementing concepts such as Overlapping Patch Embedding, Efficient Self-Attention, and Mix-FFN (Feed Forward Network) within our transformer blocks. As a result of this approach, we can enhance the crack segmentation capabilities of our model by leveraging the successful techniques proposed by SegFormer [11].

1) Overlapping Patch Embedding: It can be challenging to maintain local continuity among patches when \( N \times N \times 3 \) image patches are represented as \( 1 \times 1 \times C \) vectors in the ViT [15] architecture. In order to address this issue, the Swin Transformer [17] introduced a Shifted Window, while the SegFormer employed overlapping patch embedding. Rather than simply dividing the image into non-overlapping \( 4 \times 4 \) patches for vector embedding, over-patch embedding takes inspiration from how CNNs use sliding windows with defined parameters such as kernel size (K), stride (S), and padding (P). It predefines these parameters to split the input image into patches of size \( B \times C \times K \times N \), where \( B \) represents the batch size, \( C \) is the number of channels times the stride squared, and \( N \) is the number of patches. Then, they perform merging operations to transform the reshaped patches to \( B \times C \times W \times H \).
Fig. 3: The architecture of Hybrid-Segmentor. The diagram illustrates Hybrid-Segmentor with two paths: the upper path for CNN and the lower for Transformers. Each path generates feature maps at every layer, and the central blue boxes represent the concatenation of these feature maps from both paths.

where $W$ and $H$ represent the width and height of the merged patches, respectively. As a result, the model can capture both local and global features effectively and address the issue of losing local continuity.

2) Efficient Self-Attention: Especially in models like SegFormer with smaller patch sizes like $4 \times 4$, the self-attention layer presents computational challenges. The traditional multi-head attention process involves creating matrices for query (Q), key (K), and value (V), all of which have dimensions $N(H \times W) \times C$, and performing computations using the scaled dot-product attention equation as shown in equation 1.

$$\text{Attention}(Q, K, V) = \text{Softmax} \left( \frac{QK^T}{\sqrt{d_{\text{head}}}} \right)V$$

(1)

When dealing with large input images, the computational complexity of the provided equation 1 can lead to a significant increase in model weight. Therefore, the method that reduces the $N(H \times W)$ channels of $K$ and $V$ by applying a sequence reduction process based on a predefined reduction ratio is proposed [11]. It is possible to reshape the equation by dividing $N$ by $R$ and multiplying $C$ by $R$, $C \times R$ dimensions can be reduced to $C$ dimensions by linear operation, resulting in $\frac{N}{R} \times C$ dimensions for Key and Value matrices. Especially useful for tasks like semantic segmentation, this method efficiently manages computational complexity while preserving representation power (equation 2).

$$K = \text{Linear}(\frac{C \cdot R}{C})(\hat{K})$$

(2)

3) Mix-FFN: ViT [15] uses positional encoding for local information, which comes with fixed input resolution constraints and suffers performance drops as resolution changes. In order to overcome this issue, researchers replace positional encoding with a Convolutional $3 \times 3$ kernel in the FFN, asserting its non-essential role and providing flexibility without resolution restrictions.

$$x_{out} = \text{MLP}(\text{GELU}(\text{Conv}_{3 \times 3}(\text{MLP}(x_{in})))) + x_{in}$$

(3)

In this regard, the equation 3 simply adds a Convolutional $3 \times 3$ layer to the existing FFN within the Transformer encoder. By replacing traditional positional encoding with this adaptation, the model performance is maintained while fewer parameters are required.

C. Decoder

Two paths, one for the CNN and the other one for the transformer, are utilized in the encoder, resulting in a substantial model weight. In order to balance this, the decoder is designed to be as simple as possible. A $256 \times 256 \times 1$ feature map is generated by concatenating the outputs from both paths, as shown in Figure 3. Feature maps from each stage are combined to create a multi-scale, multi-layer feature map, which is used to create the final output. By integrating the strengths of both paths, this approach optimizes performance and efficiency while simplifying the decoder.
V. EXPERIMENTAL SETTINGS

A. Training Setup

We train the benchmarked models with a batch size of 16 on an NVIDIA RTX A5000 with 24 GB of memory in all experiments. For all models, we use early stopping with a patience of 10 epochs to ensure convergence and avoid overfitting.

B. Data

The refined dataset contains 12,000 images with cracks and non-cracks, along with the ground truth for each image. A random shuffling method was used to distribute the dataset between training, testing, and validation sets, with a ratio of 8:1:1. As a result, our dataset consists of 9,600 samples for the training, 1,200 samples for the testing, and 1,200 samples for the validation.

VI. EXPERIMENTS

A. Benchmarks

To assess the performance improvement of our model over traditional segmentation models, we compared it with FCN [13] and UNet [12]. Additionally, we include the DeepCrack2 model [29], a widely benchmarked crack detection model. As part of our evaluation, we also take into consideration SegFormer [11], and HrSegNet [31], which represent SOTA models when it comes to semantic segmentation and crack segmentation, respectively. The performance of our model is assessed both quantitatively and qualitatively.

In our comparative analysis with benchmark models, we utilized several hyperparameters. Specifically, we employed the Adam optimizer, set the initial learning rate to 1.00e-04, and used a batch size of 16. The details of the remaining hyperparameters are provided in Table III.

TABLE III: Hyperparameter settings of benchmarked models (LR indicates Learning Rate)

<table>
<thead>
<tr>
<th>Model</th>
<th>LR Schedule</th>
<th>Pre-trained</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hybrid-Segmentor</td>
<td>ReduceLROnPlateau</td>
<td>ResNet (IMAGENET1K)</td>
</tr>
<tr>
<td>HrSegNet</td>
<td>ReduceLROnPlateau</td>
<td>None</td>
</tr>
<tr>
<td>DeepCrack2</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>SegFormer</td>
<td>PolynomialLR</td>
<td>None</td>
</tr>
<tr>
<td>UNet</td>
<td>None</td>
<td>VGG19 (IMAGENET1K)</td>
</tr>
<tr>
<td>FCN</td>
<td>None</td>
<td></td>
</tr>
</tbody>
</table>

B. Loss Functions

In order to deal with the class imbalance problem in the crack dataset, many researchers propose not only new deep learning architectures but also loss functions. Representatively, Binary Cross Entropy Loss (BCE loss), Dice Loss, their fusion function, and Recall Cross Entropy loss (RecallCE) [44].

BCE loss computes individual pixel loss which ensures proportional class contribution, mitigating the bias of the dataset. By treating pixels equally, the model is able to focus on accurately classifying minority classes, such as object pixels, without being biased by dominant classes. The equation for BCE loss is as follows:

$$BCE(y, \hat{y}) = -\frac{1}{N} \sum_{i} (y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i))$$

Where N is the total number of elements (pixels in the case of segmentation). \( y_i \) is the ground truth label (0 or 1) for the i-th element. \( \hat{y}_i \) is the predicted probability for the i-th element.

Dice loss, which is equivalent to F1-score, is also suited for segmentation tasks with class imbalance due to its focus on capturing the overlap between predicted and ground truth masks, which helps address the challenge of minority class representation. By emphasizing object boundaries and assigning non-vanishing gradients to the minority class, Dice loss ensures accurate prediction and better learning for smaller classes.

$$\text{Dice Loss} = 1 - \frac{2 \cdot \text{Intersections}}{\text{Union}}$$

Its ability to sensitively measure the similarity between prediction and ground truths makes it particularly useful for precise segmentation. It can be used alongside other losses such as BCE loss to keep a balance between handling class imbalance and capturing fine details. We use a combination of BCE and Dice losses as follows:

$$\text{BCE-DICE} = \lambda \ast \text{BCE loss} + (1 - \lambda) \ast \text{Dice loss}$$

where \( \lambda \) represents the weight that takes values between 0 and 1. Previous methods attempt to improve standard cross-entropy loss in segmentation tasks by incorporating weighted factors. However, this approach can lead to issues such as reduced precision and increased false positive rate for minority classes. To address this problem, Recall Cross Entropy (RecallCE) loss is proposed as a hard-class mining solution. It reshapes the traditional cross-entropy loss by dynamically adjusting class-specific loss weights according to real time recall score, offering a more effective way to handle class imbalance and improve segmentation precision [44]. We evaluate the performance of our model by contrasting the RecallCE loss with the other losses previously mentioned, to determine if it enhances our model’s effectiveness. The equation for RecallCE loss is as follows:

$$\text{RecallCE} = -\sum_{c=1}^{C} \sum_{n:y_{n,c} = c} (1 - R_{c,t}) \log(p_{n,t})$$

where \( R_{c,t} \) represents the recall of class c during optimisation iteration t.

VII. EVALUATION

We carry out two prior studies to examine specific aspects of the performance of our model: (1) assessing the impact of individual encoder paths and (2) examining various combinations of loss functions. Initially, we aim to understand how distinctively each encoder extracts features. Then, we investigate which combination of loss functions yields the best
results in crack segmentation by assigning different weights to the BCE and DICE loss and applying RecallCE loss.

A. Encoder Paths

We conduct an experiment involving the training and testing of two different encoders to assess their capabilities in feature extraction. Specifically, we aim to determine whether convolutional layers perform well at extracting local features while transformers are adept at capturing global features.

The results presented in Table IV indicate that the CNN path achieves a higher precision score. This suggests that the Transformer path tends to produce more false positives, which can mistakenly predict non-crack pixels as cracks. On the other hand, the CNN path tends to produce more false negatives, possibly misclassifying crack pixels as non-cracks. As a result of these findings, we can assume the Transformer path captures broader areas as cracks, while the CNN path captures finer details.

In Figure 4, the segmentations produced by each of the two encoders further illustrate the differences in their performance.

B. Loss functions

The model combines BCE and DICE for addressing class imbalances and capturing fine-grained details synergistically, leveraging the strengths of both. This combination provides a balance between recognizing dominant classes and accurately segmenting minority groups, resulting in a more effective model for imbalanced data. We assess these aspects by varying the weights assigned to BCE and DICE loss functions and also exploring the RecallCE loss for its class imbalance handling capabilities. Our experiments, as shown in Table [V] reveal that when BCE and DICE loss weights are roughly equal, the model generally performs better. The BCE-DICE loss when $\lambda$ is 0.2, outperforms in all metrics except for precision. Precision peaks at 0.817, but this trades off with recall, resulting in relatively lower performance in other metrics. According to our expectations, applying the RecallCE loss would result in the highest recall score. Due to the fact that this loss is designed to balance precision and recall, it produces well-balanced results. Furthermore, RecallCE loss is closely followed by BCE-DICE loss with a weight of 0.2, indicating that it is effective at addressing class imbalance. In summary, the BCE-DICE loss with a weight of 0.2 exhibits the best model performance, and was chosen as the loss function for our final model.

TABLE IV: Comparison and Performance Analysis of CNN and Transformer Paths. All models in the table utilized BCE-DICE loss with a weight of $\lambda = 0.5$.

<table>
<thead>
<tr>
<th>Model Name</th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
<th>F1 Score (Dice)</th>
<th>IOU score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hybrid-Segmentor</td>
<td>0.970</td>
<td>0.807</td>
<td>0.727</td>
<td>0.763</td>
<td>0.620</td>
</tr>
<tr>
<td>CNN path</td>
<td>0.969</td>
<td>0.802</td>
<td>0.722</td>
<td>0.758</td>
<td>0.614</td>
</tr>
<tr>
<td>Transformer path</td>
<td>0.965</td>
<td>0.717</td>
<td>0.772</td>
<td>0.741</td>
<td>0.592</td>
</tr>
</tbody>
</table>

C. Final Evaluation

We applied the most commonly used BCE-DICE loss ($\lambda = 0.5$, equal weighting between two losses) to all benchmark models in our experiment. As demonstrated in Table VI, our model notably outperforms the other five benchmarked models. Our model achieved an accuracy of 0.971, a precision of 0.804, a recall of 0.744, an F1-score of 0.770, and an IOU score of 0.630. These impressive results demonstrate the model’s exceptional proficiency in crack segmentation tasks.

Qualitatively, our model exhibits significant improvements relative to existing models (Figure 5). As shown by (A) and (C), our model handles crack discontinuity more accurately. Furthermore, in (B), our model excels at identifying vague cracks that other models fail to detect. Our model offers the advantage of being able to detect cracks regardless of the type of surface. While crack detection on brick surfaces is
Fig. 5: Example crack images segmented by our model and benchmarked models. The red ovals highlight the areas where our model outperforms other benchmarked models. In examples without red ovals, such as (F) and (H), our model demonstrated strong performance across overall structures.

TABLE VI: Performance of our crack segmentation model against state-of-the-art models

<table>
<thead>
<tr>
<th>Model Name</th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
<th>F1 Score (Dice)</th>
<th>IOU score</th>
</tr>
</thead>
<tbody>
<tr>
<td>FCN</td>
<td>0.968</td>
<td>0.802</td>
<td>0.702</td>
<td>0.746</td>
<td>0.598</td>
</tr>
<tr>
<td>UNet</td>
<td>0.968</td>
<td>0.789</td>
<td>0.720</td>
<td>0.750</td>
<td>0.603</td>
</tr>
<tr>
<td>DeepCrack2</td>
<td>0.968</td>
<td>0.788</td>
<td>0.704</td>
<td>0.741</td>
<td>0.592</td>
</tr>
<tr>
<td>SegFormer</td>
<td>0.965</td>
<td>0.750</td>
<td>0.719</td>
<td>0.730</td>
<td>0.580</td>
</tr>
<tr>
<td>HrSegNet</td>
<td>0.969</td>
<td>0.800</td>
<td>0.724</td>
<td>0.757</td>
<td>0.612</td>
</tr>
<tr>
<td>Hybrid-Segmentor</td>
<td>0.971</td>
<td>0.804</td>
<td>0.744</td>
<td>0.770</td>
<td>0.630</td>
</tr>
</tbody>
</table>

challenging due to the ambiguity between cracks and brick borders and resulting shadows, as shown in (D), our model is adept at handling such scenarios. On the other hand, models such as FCN incorrectly predict brick borders as cracks. Additionally, a challenge in crack detection involves identifying non-crack areas within cracked regions, which our model effectively addresses, as evident in (E) and (G). (H) demonstrates how our model is applicable to blurred images. Furthermore, (F) demonstrates the effectiveness of our model in detecting intricate crack contours.

VIII. LIMITATIONS

Although our model outperforms other benchmarked models in performance, it still exhibits certain limitations. Two
primary shortcomings of our model have been identified and presented in Figure 6. Firstly, our model struggles with the detection of thinner cracks within web-shaped crack patterns. Example images (A) to (D) illustrate web-shaped cracks, for which our model fails to detect the thin branches, while thicker ones are successfully identified.

Secondly, our model is sensitive to disruptions caused by distortions, such as occlusions and watermarks. (E) illustrates a situation where the watermark located within the crack, is identified as a non-crack area. However, in (F), even with the presence of a watermark, our model fails to predict the cracks hidden by a translucent occlusion. Furthermore, (G) demonstrates an issue where the model does not recognize letters on the road as part of the background. The variation in model performance may be attributed to the clear color contrast between the letters and the background, causing confusion for the model.

We believe that room for improvement remains to address these limitations by suggesting improved architecture. Additionally, techniques such as Generative Adversarial Networks (GANs) and meta-learning can be harnessed to generate synthetic data during the data pre-processing phase to deal with the lack of data. Furthermore, recognizing the increasing need for 3D crack image segmentation, the creation of high-quality 3D crack image datasets becomes imperative to advance this domain.

<table>
<thead>
<tr>
<th>Image</th>
<th>Ground Truth</th>
<th>Hybrid Segmentor</th>
<th>HrSegNet</th>
<th>DeepCrack2</th>
<th>SegFormer</th>
<th>UNet</th>
<th>FCN</th>
</tr>
</thead>
<tbody>
<tr>
<td>(A)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(B)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(C)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(D)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(E)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(F)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(G)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 6: The examples of Hybrid-Segmentor’s limitations, including failure to detect thin or web-shaped objects and difficulties with occlusions.

IX. CONCLUSION

In this research, we propose a novel model for crack segmentation called Hybrid-Segmentor. This architecture incorporates two distinct encoder paths, namely the CNN path and the Transformer path. For the CNN path, we use the well-established ResNet-50 architecture [3], which is renowned for its ability to extract local features. Additionally, we introduce the concept of Overlapping Patch Embedding, Efficient Self-Attention, and Mix-FFN in the Transformer path, derived from the SegFormer [11] model. In combining two encoders, these additions optimize computational efficiency and model size, thereby soothing capacity problems. We further simplify the model with a relatively simpler decoder to minimize its size.
Through experimentation, Hybrid-Segmentor emerges as a SOTA, outperforming other renowned benchmark models. Our model effectively takes advantage of the two encoder paths, as proved by prior studies evaluating its performance in extracting local and global crack features. Based on the findings of previous studies, the BCE-DICE loss, weighted at 0.2 on the BCE loss, yields the best performance. In qualitative analysis, our model improves in addressing discontinuities, detecting small non-cracked areas within cracks, and recognizing cracks even in low-quality images and diverse surfaces. It is capable of capturing more details in crack contours than previous models.

Furthermore, our study introduces a data refinement methodology for combining publicly available datasets comprising 13 open-source crack datasets with refined ground truths. Since these datasets initially used diverse standards for creating ground truth, we merge and improve them to ensure equivalence, thereby increasing their reliability and precision. In addition, we employ a specific data augmentation approach in order to address the issue of class imbalance within our dataset. By extracting data containing cracks with more than 5000 pixels and augmenting them, we are able to incorporate these samples into our dataset. Our effort resulted in a dataset consisting of 12,000 crack images, each with its corresponding ground truth.

To enhance our crack detection model, we need to concentrate on enhancing the architecture to efficiently recognize thin, web-shaped cracks, and those that are hidden by occlusions. We could also explore the possibility of using Generative Adversarial Networks (GANs) and meta-learning to create synthetic data to overcome data scarcity, particularly in the development of 3D crack image segmentation datasets.

X. ACKNOWLEDGMENT

This paper is an extension of the research conducted for MSc thesis under the supervision of Dr. Alessandro Artusi and Dr. Carlo Ciliberto. The research work of Dr. Alessandro Artusi and Dr. Xenios Milidonis has been partially funded from the European Union’s Horizon 2020 research and innovation programme under grant agreement No. 739578 and from the Government of the Republic of Cyprus through the Deputy Ministry of Research, Innovation and Digital Policy.

REFERENCES


June Moh Goo received a BEng in Engineering Mathematics from the University of Bristol, United Kingdom in 2019. He later pursued an M.Sc. in Data Science at the University of Bath, United Kingdom, which he completed in 2020. He is currently working towards a Ph.D. in civil, environmental, and geomatic engineering at University College London, UK. He is an active member of the photogrammetry and 3D imaging group and is being supported by the Engineering and Physical Sciences Research Council through an industrial CASE studentship with Ordnance Survey. Previously, he worked as a Research Intern within the multidisciplinary research group (MRG) of DeepCamera at CYENS Centre of Excellence in Cyprus. During his internship, he worked on developing deep learning architecture for crack segmentation. His research interests primarily include 3D point cloud processing and multi-domain label-efficient learning for point clouds.

Xenios Milidonis received a B.Sc. in Physics from the University of Cyprus in 2011, an M.Sc. in Physics and Computing in Medicine and Biology (Medical Physics) from the University of Manchester in 2012, and a Ph.D. in Molecular and Clinical Medicine (Neuroimaging) from the University of Edinburgh in 2017. Since then, he worked as a post-doctoral Research Associate at the School of Biomedical Engineering & Imaging Sciences, King’s College London, focussing on the fabrication of multimodality physical standards for the assessment and validation of dynamic cardiac first-pass perfusion imaging, as well as the development of algorithms for the automated quantitative analysis of magnetic resonance imaging data. He is currently a Senior Research Associate in the DeepCamera multidisciplinary research group (MRG) at CYENS Centre of Excellence in Cyprus, working towards the development of deep learning-based tools for image processing, visualization and evaluation, while co-managing the DeepCamera MRG. He is also a registered Medical Physicist in Cyprus and a Visiting Scholar at King’s College London.

Alessandro Artusi received the Ph.D. degree in computer science from the Vienna University of Technology, in 2004. He is currently a Research Associate Professor and the Managing Director of the DeepCamera Laboratory, CYENS, Cyprus. Recently, he joined as a Founding Member of the Moving Picture, Audio and Data Coding by Artificial Intelligence (MPAI), a not-for-profit standards organization established in Geneva. His research interests include visual perception, image/video processing, HDR technology, objective/subjective imaging/video evaluation, deep learning, computer vision, and color science. He was a recipient of the prestigious BSI Award for his work on the JPEGXGT standard. He was appointed as the Head of Delegation for the Cypriot National Body for the ISO/IEC/JCT 1 SC29 Committee. In the past, he has been a BSI Member of the IST/37 Committee and the U.K. Representative for the JPEG and MPEG Standardization Committees.

Carlo Ciliberto is Associate Professor with the Centre for Artificial Intelligence at University College London, He is member of the ELLIS society and of the ELLIS Unit based at UCL. He obtained his bachelor and master degrees in Mathematics at the Universita Roma Tre (Magna Cum Laude) and a PhD in machine learning applied to robotics and computer vision at the Istituto Italiano di Tecnologia. He has been Postdoctoral Researcher at the Massachusetts Institute of Technology with the Center for Brain Minds and Machines and became Lecturer (Assistant Professor) at Imperial College London before joining UCL, where he now carries out his main research activity. Carlo’s research interests focus on foundational aspects of machine learning within the framework of statistical learning theory. He is particularly interested in the role of “structure” (being it in the form of prior knowledge or structural constraints) in reducing the sample complexity of learning algorithms with the goal of making them more sustainable both computationally and financially. He investigated these questions within the settings of structured prediction, multi-task and meta-learning, with applications to computer vision, robotics and recommendation systems.