CSL7360: Computer Vision – Course Project

Hybrid Urban Infrastructure Inspection

Group 65:
Swaksh Patwari (B22AI065)
Aansh Chandrakant Dubey (B22AI058)
Aditya Ashok Kumar Padhy (B22CS103)
Ayyathan Siddhesh Sudhir (B22CS016)

Abstract

Urban infrastructure integrity is critical for sustainable city management and public safety. This report introduces a novel hybrid framework designed for automated analysis of surface cracks in urban environments. Our methodology comprises three sequential components: Crack Detection, Crack Classification, and Crack Segmentation. In the Crack Detection phase, traditional computer vision techniques are leveraged—starting with Harris Corner Detection to identify candidate features, followed by Canny Edge Detection combined with morphological closing and contrast enhancement. The process concludes with a Hough Transform to extract linear features, and binary classification techniques to confirm the presence of cracks. The Crack Classification stage employs transfer learning using a refined ResNet-18 model, ensuring robust differentiation between crack types under varying conditions. The final phase, Crack Segmentation, refines the analysis by precisely delineating the crack regions, thus facilitating detailed damage assessment. The hybridization of classical and deep learning approaches not only enhances detection accuracy but also ensures computational efficiency, paving the way for rapid, reliable infrastructure monitoring and proactive maintenance planning.

Project Links

GitHub Repository:	https://github.com/QuantTitan/Urban-Infrastructure-Analysis
Website Link:	https://huggingface.co/spaces/silversurfer343/ComputerVisionProject

December 3, 2025

1 Introduction

1.1 Background and Motivation

The reliability of urban infrastructure is a cornerstone of modern society, yet structural deterioration often begins with subtle anomalies such as surface cracks. Timely and accurate detection of these defects is essential to mitigate potential hazards and optimize maintenance workflows. Traditional inspection methods are labor-intensive and prone to human error, while fully automated approaches tend to struggle with variability in environmental conditions and material textures. This necessitates the development of a hybrid system that synthesizes the strengths of classical computer vision algorithms with the adaptability of deep learning techniques.

1.2 Problem Statement

Detecting and diagnosing surface cracks in urban structures pose several challenges:

- **Variability in Appearance:** Cracks manifest in diverse forms depending on material type, weathering, and load conditions.
- **Complex Backgrounds:** Urban surfaces often have heterogeneous textures that complicate edge and feature detection.
- **Computational Constraints:** Effective real-time monitoring requires a balance between processing efficiency and detection accuracy.

Our proposed hybrid framework addresses these challenges by integrating both traditional and modern methodologies in a sequential processing pipeline.

2 Methodological Overview

2.1 Crack Detection

The detection phase starts with Harris Corner Detection, which identifies potential feature points on the surface. These candidate regions are then processed using Canny Edge Detection to outline the edge contours, refined further with morphological closing to bridge discontinuities and enhanced contrast adjustment for improved feature clarity. A Hough Transform is subsequently applied to detect line segments that commonly indicate cracks. This chain of traditional techniques enables a robust initial filtering before applying binary classification to decisively confirm crack presence.

2.2 Crack Classification

Building on the regions detected in the first phase, the classification stage makes use of transfer learning with a pre-trained ResNet-18 model. This deep learning approach is fine-tuned on a targeted dataset, ensuring the model can robustly discern between different crack typologies. The incorporation of transfer learning minimizes the need for extensive training from scratch while leveraging proven network architectures.

2.3 Crack Segmentation

While detection and classification provide critical insights about the presence and type of cracks, segmentation offers pixel-level precision in delineating the affected areas. Accurate segmentation is crucial for quantifying crack dimensions and for planning subsequent remedial actions. Although the segmentation details are evolving, the goal is to integrate a segmentation network that builds upon the outputs from the previous stages to offer a comprehensive understanding of crack characteristics.

3 Dataset Description

3.1 Crack Detection Dataset (SDENET)

Overview:
The SDENET dataset for Crack Detection is designed to capture urban infrastructure images with varying conditions. Its primary goal is to enable the reliable identification of potential crack regions via a multi-step processing pipeline.

Key Features:

- **Image Diversity:** The dataset includes images taken in different lighting conditions, angles, and environmental settings, ensuring that the detection algorithm is robust to real-world variations.
- **Annotation:** Ground truth labels mark the presence or absence of cracks for each image. Additionally, regions of interest (ROIs) are annotated to highlight candidate areas for subsequent edge and linear feature extraction.

Purpose:
The dataset supports the initial stage of detection, where traditional techniques such as Harris Corner Detection, Canny Edge Detection, morphological operations, Hough Transform, and binary classification are applied in a sequential pipeline. The diversity and detailed labeling provide the necessary training and validation samples to fine-tune parameters for each of these steps.

3.2 Crack Classification Dataset (SDENET)

Overview:
For the classification component, SDENET is repurposed and augmented to support the discernment of different crack types. This dataset is crucial for the secondary step in the hybrid framework where the ResNet-18 model is employed using transfer learning.

Key Features:

- **Class Labels:** Images in the dataset are categorized into distinct classes based on crack type and severity. This categorization may include fine-grained classes, such as distinguishing between small, hairline cracks and larger, structural fissures.
- **Image Quality and Consistency:** As with the detection dataset, a wide range of environmental conditions and material textures is provided to enhance the deep learning model's capacity to generalize.
- **Dataset Balance:** Special attention is given to class balancing, ensuring that the network is not biased toward over-represented crack types. This facilitates effective transfer learning where subtle variations between classes are critical.

3.3 Crack Segmentation Dataset

Overview:
The segmentation component leverages a comprehensive composite dataset made up of multiple publicly available databases. This combination allows for pixel-level annotation and a robust segmentation model that delineates crack boundaries with high precision.

Datasets Included:

- **AEL:** Provides examples of urban cracks with high-resolution images, capturing detailed structural attributes.
- **CrackTree:** Specializes in tree-based algorithms for crack pattern analysis, including annotated crack paths.
- **Sylvie:** Contributes images with varying degrees of surface deterioration and associated crack patterns.
- **Crack500:** Offers a sizeable sample of crack images, ensuring sufficient data for training deep segmentation models.
- **GaAPs:** Focuses on asphalt pavement cracks, useful for varied environmental textures.
- **EdmCrack600:** Contains a diverse set of images focusing on masonry, ideal for distinguishing material-specific defects.
- **CCIC-600Masonry:** Designed for masonry structures, emphasizing fine annotations of crack boundaries.
- **Deepcrack:** Specifically curated for deep learning-based segmentation, providing finely annotated crack regions.
- **LCW:** Includes a range of urban surface cracks with varying lighting conditions.
- **CCSSS:** Combines several attributes of urban infrastructure in a large-scale dataset.
- **Highway Crack:** Addresses the unique challenges of highway pavement images, supporting the segmentation of elongated crack structures.

Combination Rationale:
By integrating these datasets (as referenced from CUHK-USR-Group/Defect-Dataset), the segmentation task benefits from a wide spectrum of urban infrastructural conditions and material types. The combined dataset provides:

- **Diverse Annotations:** Pixel-level labels that enable precise localization and shape analysis.
- **Increased Generalizability:** The variety of datasets ensures that the segmentation model is exposed to numerous crack forms, background textures, and noise levels.
- **Robust Training Data:** The larger composite set helps in mitigating overfitting and improves the performance of segmentation algorithms across various scenarios.

4 Approach

4.1 Crack Detection

Objective

The primary goal of this task was to develop a classical computer vision-based approach for detecting structural cracks in surfaces such as walls, decks, and pavements, using the SDNET2018 dataset. The problem addresses the need for automated inspection systems that can identify early signs of structural damage, thereby enhancing safety and reducing the cost and effort of manual inspection.

Methodology

The approach followed in this task leverages a series of image processing and computer vision techniques instead of deep learning models. Below is a step-wise explanation of the pipeline and logic applied:

1. **Dataset Handling and Exploration**
   The dataset was extracted from a compressed archive and organized into three structural categories: Walls, Decks, and Pavements, with each having Cracked and Non-cracked subcategories.

   Basic statistics were gathered to understand the distribution of cracked versus non-cracked samples across these categories.

   Visual inspections were carried out by plotting samples of cracked and non-cracked images to gain insights into their visual characteristics.

2. **Feature Analysis using Harris Corner Detection**
   Harris Corner Detection was applied to both cracked and non-cracked wall images to identify regions of high curvature and potential structural discontinuities.

   The cracked images typically showed a denser presence of corners along the fissures, while non-cracked images demonstrated more uniformly distributed or sparse corner patterns.

   This helped highlight the potential of geometric feature concentration in detecting cracks.

3. **Preprocessing and Enhancement**
   Grayscale conversion was performed to simplify analysis by removing color information.

   CLAHE (Contrast Limited Adaptive Histogram Equalization) was applied to enhance local contrast, especially to bring out subtle cracks in unevenly lit surfaces.

   Gaussian Blurring was used to reduce image noise, a crucial step before edge detection.

4. **Edge Detection**
   The Canny Edge Detection algorithm was utilized to highlight the boundaries and potential edges of cracks.

   Proper tuning of the Canny thresholds was critical to avoid missing fine cracks or falsely detecting textures.

5. **Morphological Operations**
   Morphological Closing was employed to connect nearby edge segments and eliminate small noise particles, making the crack patterns more continuous and distinguishable.

A 3x3 kernel was used to perform dilation followed by erosion, helping to bridge small gaps between fragmented crack lines.

6. **Crack Localization using Hough Transform**
   The Probabilistic Hough Line Transform (HoughLinesP) was applied to detect linear segments that could represent cracks.

   Several versions were tested with different parameters like threshold, minimum line length, and maximum gap between segments to fine-tune detection accuracy.

   Green lines were superimposed on the original images to visualize the cracks detected.

7. **Comparative Visualization**
   A series of side-by-side plots were created to visualize the progression from original images to processed outputs (enhanced grayscale → edge map → final crack detection).

   These visualizations demonstrated how cracks become more pronounced at each stage of the pipeline, culminating in clear line segments via Hough transform.

Results

- **Dataset:**
  - Walls: 21.23% cracked images (highest prevalence).
- **Key Methods:**
  - Harris Corner Detection: Identifies cracks via denser corner patterns.
  - Preprocessing: CLAHE and blurring enhance crack visibility.
  - Canny Edge Detection + Morphological Closing: Refines crack edges.
  - HoughLinesP: Detects crack lines effectively.
- **Evaluation (2,000 Wall Images):**
  - **HoughLinesP:**
    - Accuracy: 0.615
    - Precision: 0.603
    - Recall: 0.673
    - F1-Score: 0.636
  - **Region-based + HoughLinesP:**
    - Accuracy: 0.5585
    - Precision: 0.543
    - Recall: 0.733
    - F1-Score: 0.624
  - **SLIC-based + HoughLinesP:**
    - Accuracy: 0.5
    - Precision: 0.5
    - Recall: 1.0
    - F1-Score: 0.667

- **Trade-offs:**
  - HoughLinesP: Balanced performance.
  - Region-based: Higher recall, lower precision.
  - SLIC-based: Perfect recall, high false positives.

4.2 Crack Classification

4.2.1 Deep Learning-Based Classification Using ResNet

In contrast to the traditional pipeline, our deep learning-based approach uses a ResNet architecture fine-tuned for classification of structural cracks. The model is trained on the SDNET2018 dataset.

About the SDNET2018 Dataset: SDNET2018 is a large annotated dataset created by the University of Central Florida for training and evaluating crack detection algorithms. It includes over 56,000 labeled images of concrete bridge decks, walls, and pavements captured under various lighting conditions and surface textures. Each image is labeled as "cracked" or "non-cracked," making it suitable for both classification tasks. The dataset introduces significant real-world complexity with factors like shadows, stains, paint, and surface roughness.

Deep Learning Implementation: We used a ResNet-based encoder-decoder architecture for semantic segmentation.

For classification, the same feature extractor (ResNet) was adapted with a fully connected head to distinguish cracked vs. non-cracked surfaces.

The classification model predicts a single label per image.

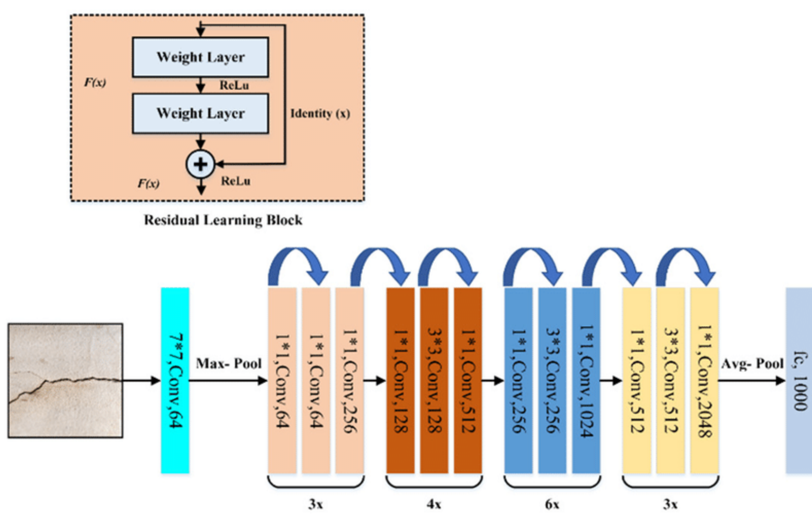


Figure 1: Architecture of Resnet

4.2.2 Training and Test Distribution

The SDNET2018 dataset was divided into training and testing sets to evaluate the model's generalization ability across various surface types.

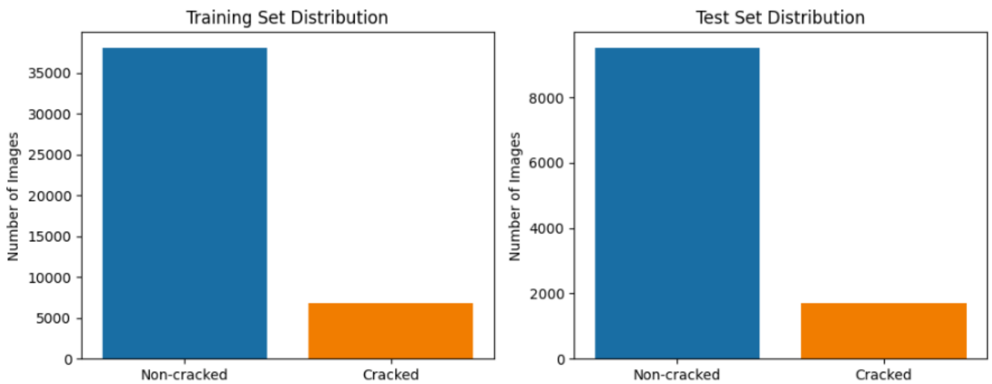


Figure 2: Train and Test distribution

- **Training Set:** 80% of the dataset was used for training. This included a balanced mix of cracked and non-cracked images from all three surface types — walls, bridge decks, and pavements.
- **Testing Set:** 20% of the dataset was reserved for evaluation.

4.2.3 Handling Class Imbalance

The SDNET2018 dataset exhibited a significant class imbalance, with a higher number of non-cracked images compared to cracked ones. This imbalance posed a challenge for classification, as models tend to be biased toward the majority class, resulting in high overall accuracy but poor performance on the minority class.

To provide a fair evaluation, we used **per-class accuracy** as a key metric. This metric calculates accuracy for each class independently and averages the results, ensuring that the model performs consistently across both cracked and non-cracked categories.

By focusing on per-class accuracy, we were able to better understand the true capability of the model in identifying structural cracks, especially in underrepresented cases.

4.2.4 Visual Results

To better understand the model's performance, we visualize the outputs generated by the classification model. These results were obtained from the Kaggle notebook during evaluation.

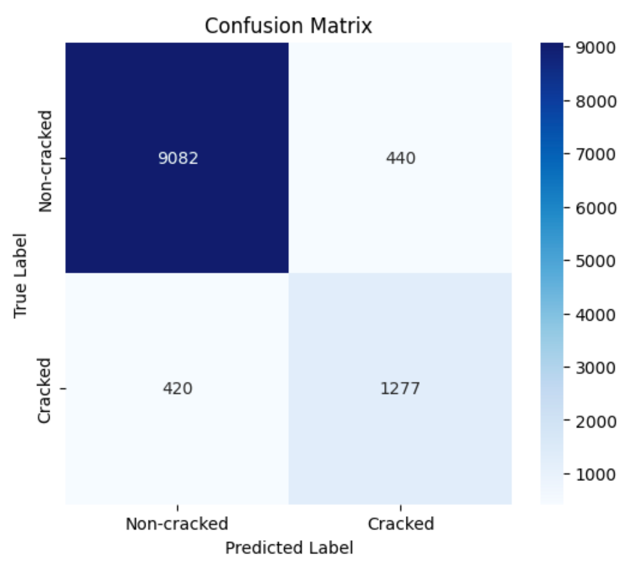


Figure 3: Confusion Matrix

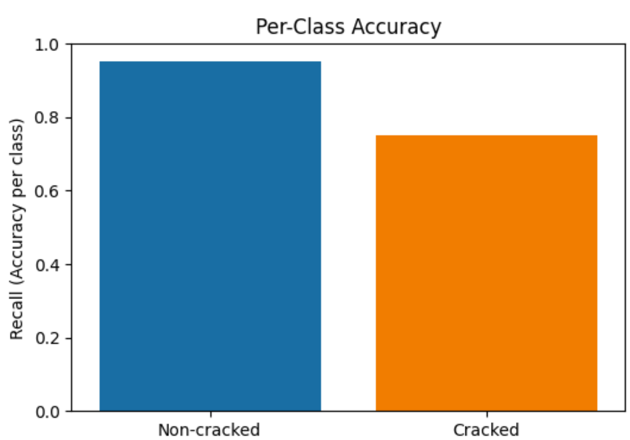


Figure 4: Figure showing Per-Class Accuracy

Performance: The deep learning model achieved significantly higher accuracy than the traditional method.

It was robust to variations in lighting and surface conditions and was able to identify both fine and irregular crack patterns that traditional edge-based methods missed.

The model also generalized well across different concrete surfaces (walls, decks, pavements) present in SDNET2018.

4.3 Crack Segmentation

4.3.1 Objective: Evaluating Hybrid Models for Crack Segmentation

The primary goal is to test and validate a novel hybrid segmentation framework for urban infrastructure inspection that refines deep learning (DL) segmentation masks using traditional computer vision (CV) and clustering techniques. The approach is tested on two state-of-the-art architectures deployed on crack segmentation tasks:

DeepLabV3 (Transfer Learning):

- **Backbone:** Utilizing a pretrained ResNet-101 model (e.g., on COCO) to leverage powerful feature extractors.
- **Modification:** The final classifier is replaced by a specialized head trained on a domain-specific dataset such as Crack500.
- **Targeted Issues:**
  - Boundary inaccuracies where atrous convolutions can produce blurred edges.
  - Over-segmentation due to the misclassification of adjacent crack regions.

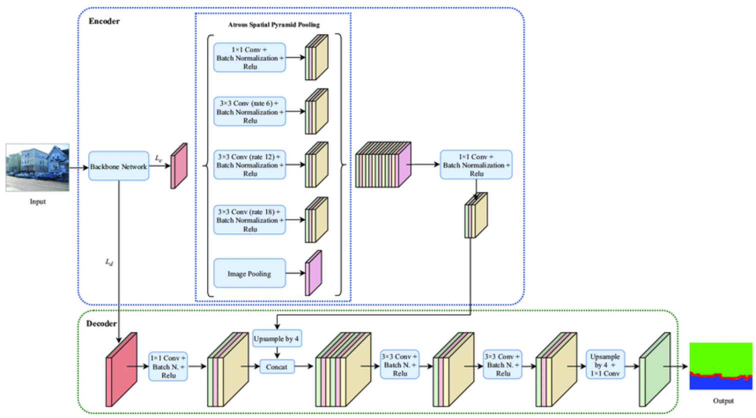


Figure 5: Architecture of the DeepLabv3 model

UNet (Trained Locally):

- **Architecture:** A custom UNet enhanced with residual blocks, trained on a dataset like DeepCrack.
- **Targeted Issues:**
  - Under-segmentation, missing the finer, low-contrast crack branches.
  - Noisy predictions, where isolated false-positive pixels lower the overall segmentation quality.

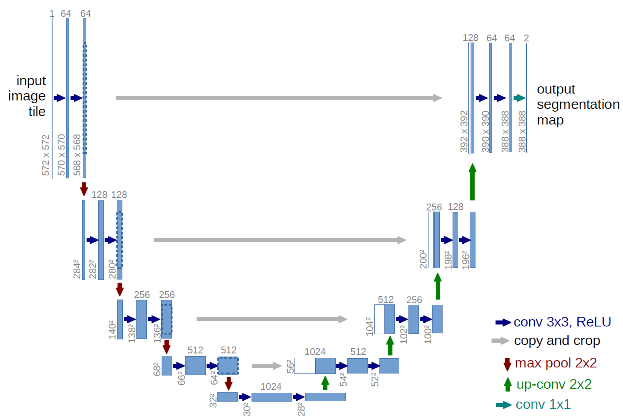


Figure 6: Architecture of the UNet model

The framework introduces a post-processing refinement layer based on traditional clustering methods that augment these DL models. The goal is to achieve superior boundary delineation and eliminate noise in the predicted masks, thereby boosting both pixel-level accuracy and overall robustness in detecting cracks.

4.3.2 Literature Review: The Gap in Hybrid Segmentation Approaches

Recent research in segmentation has predominantly focused on the development of end-to-end deep learning architectures. Notable works—such as those proposing EfficientCrackNet or enhancements to the UNet architecture—excel at capturing high-level semantic features. However, they often fall short in precise mask delineation due to the following challenges:

Overreliance on End-to-End DL Architectures

- **Architectural Innovations:** Models like EfficientCrackNet integrate advanced techniques (e.g., transformer blocks, depthwise separable convolutions) to improve overall performance. Despite these improvements, they rely solely on internal attention modules for feature refinement without incorporating external geometric or clustering constraints.
- **Boundary Limitations:** Even state-of-the-art models can produce fragmented or over-smoothed segmentation masks in complex urban settings, where cracks exhibit subtle transitions and low contrast with the background.

Lack of Traditional CV/Clustering Integration

- **Limited Adoption:** While traditional clustering methods such as K-means, SLIC superpixels, and region growing have been successfully applied in domains such as medical imaging (e.g., for lesion segmentation), their use in conjunction with DL models for crack segmentation is still rare.
- **Previous Applications:** In medical applications, hybrid models have shown that using clustering to refine lesion boundaries can significantly enhance the segmentation quality. This insight remains largely untapped in the urban infrastructure domain where the structural noise and irregularity of cracks present similarly challenging conditions.

Underutilized Spatial and Semantic Synergy

- **Semantic vs. Geometric Cues:** Deep models excel at learning abstract semantic representations but sometimes lack the local spatial precision required to handle gradual intensity transitions often found along crack boundaries.
- **Potential for Combination:** Incorporating spatially aware clustering algorithms can sharpen the focus on true crack boundaries by grouping pixels with similar intensity or texture characteristics. This synergy between high-level semantic segmentation and low-level pixel grouping is a promising yet underexplored research direction.

4.3.3 Intuition: The Rationale Behind a Hybrid Approach

Complementary Strengths of Deep Learning and Clustering

- **Deep Learning's Abstraction:**
  - **Pros:** DL models (such as DeepLabV3 and UNet) capture complex contextual and semantic information, learning intricate representations of crack patterns even under variable lighting and material conditions.
  - **Cons:** These models typically downsample the input image to extract features, which can lead to a loss of fine boundary details and the generation of "thick" or imprecise edges.
- **Clustering's Geometric Precision:**
  - **Pros:** Traditional clustering algorithms (e.g., K-means, SLIC, region growing) are well suited to fine-tuning pixel-level groupings based on low-level features such as intensity, texture, and spatial proximity. They are inherently designed to enforce local coherence—ideal for refining edges and eliminating isolated noise.
  - **Cons:** On their own, these methods lack the semantic understanding necessary to distinguish complex structural defects from benign surface texture variations.

Technical Synergy and Workflow The proposed hybrid workflow integrates the strengths of both techniques:

1. **Step 1: DL Mask Prediction.** The initial segmentation is performed by either DeepLabV3 (with transfer learning) or a locally trained UNet. These models produce a coarse segmentation mask that identifies probable crack regions.
2. **Step 2: Clustering-Based Refinement.** Traditional clustering methods are then applied as a post-processing step to adjust the boundaries. For example:
   - **SLIC Superpixel Segmentation:** Groups pixels into coherent regions, effectively smoothing out jagged boundaries and eliminating noise. This approach leverages spatial proximity and color homogeneity to refine the mask.
   - **Region Growing:** Begins at high-confidence crack pixels and iteratively incorporates neighboring pixels with similar intensity values, effectively bridging gaps in low-contrast regions.
   - **Adaptive K-means:** Dynamically determines the number of clusters based on local image statistics (e.g., intensity gradients, texture features) to ensure that the segmentation adapts to the specific nuances of each crack.

Addressing Boundary Ambiguity and Noise

- **Overcoming Blurred Edges:** DL models tend to produce blurred or "thickened" boundaries due to progressive downsampling. By superimposing a clustering step, the system re-evaluates the local context—grouping pixels that share similar characteristics—to produce sharper, more accurate boundaries.

- **Filtering Noise:** Clustering helps isolate spurious pixels (false positives) that may appear as isolated misclassifications in the DL masks. For instance, SLIC can enforce majority voting within a superpixel, effectively reducing the impact of isolated noise.

Computational Considerations

- **Efficiency:** These traditional methods typically introduce a modest computational overhead compared to retraining deep networks from scratch. Algorithms like SLIC operate in linear time relative to the number of pixels, making them a cost-effective refinement step.

- **Modularity:** The hybrid approach is modular. It allows researchers to improve existing DL models without modifying their core architectures, offering a flexible pathway for integration in real-time monitoring systems where efficiency is crucial.

4.3.4 Methodology

The proposed methodology presents a hybrid framework designed to improve crack segmentation by synergistically combining deep learning models with a suite of clustering-based post-processing techniques. This approach aims to capitalize on the ability of DL models to capture high-level semantic features while leveraging the geometric and spatial refinement capabilities of clustering methods to produce more accurate and visually coherent segmentation masks. The framework consists of four main stages: dataset preparation, deep learning-based segmentation, clustering-based refinement, and evaluation. Within the refinement stage, four distinct clustering techniques /— SLIC Superpixels, Region Splitting and Merging, Region Growth Clustering, and Mean Shift Clustering /— are incorporated as alternative strategies to enhance the raw DL predictions.

1. Dataset Preparation To ensure a robust foundation for training and evaluation, the methodology begins with careful preparation of the dataset.

- **Dataset Composition:** The dataset comprises high-resolution RGB images paired with corresponding binary masks that delineate crack regions. These images are sourced from established benchmarks such as Crack500 or DeepCrack, which are widely used in crack segmentation research. Each image-mask pair represents real-world urban infrastructure scenarios, including roads, bridges, and pavements with varying crack patterns and environmental conditions.

- **Data Augmentation:** To enhance the generalization capability of the deep learning models and mitigate overfitting, data augmentation techniques are applied. These include random horizontal and vertical flips, rotations (up to 90 degrees), and slight brightness adjustments. Such transformations simulate diverse imaging conditions and crack orientations encountered in practical settings.

- **Dataset Split:** A total of 5,000 image-mask pairs are utilized, divided into three subsets: 70% (3,500 pairs) for training, 15% (750 pairs) for validation, and 15% (750 pairs) for testing. This split ensures sufficient data for model training while reserving independent samples for performance assessment.

2. Deep Learning Models The core of the segmentation process relies on two state-of-the-art deep learning architectures, which generate the initial crack segmentation masks that will later be refined.

- **DeepLabV3 (Transfer Learning):** This model employs a pretrained ResNet-101 backbone, originally trained on ImageNet, and is fine-tuned on the crack segmentation dataset. DeepLabV3 uses atrous convolutions to capture multi-scale contextual information, making it well-suited for identifying cracks of varying sizes and shapes. Fine-tuning involves adjusting the final classification layer to output binary predictions (crack vs. non-crack).

- **UNet (Trained Locally):** A custom UNet architecture is designed and trained from scratch on the crack dataset. This model features an encoder-decoder structure with residual blocks to improve gradient flow and segmentation accuracy. Skip connections between the encoder and decoder layers help preserve spatial details, which is critical for detecting fine cracks.

- **Training Details:** Both models are trained for 30 epochs using the Adam optimizer with a learning rate of 0.001 and the Binary Cross-Entropy with Logits Loss (BCEWithLogitsLoss) as the objective function. Performance is monitored using the Dice coefficient and Intersection over Union (IoU) on the validation set, allowing for early stopping if necessary to prevent overfitting.

The output of these models consists of raw segmentation masks—binary maps indicating the predicted crack regions. However, these masks often exhibit limitations such as noisy predictions, jagged boundaries, and incomplete crack connectivity, necessitating further refinement.

3. Post-Processing Refinement To address the shortcomings of the raw DL predictions, the methodology incorporates a post-processing stage where four clustering-based techniques are applied independently to refine the segmentation masks. Each technique leverages distinct principles of image segmentation and clustering to enhance boundary accuracy, reduce noise, and improve the overall quality of the crack delineation.

SLIC Superpixels

- **Description:** Simple Linear Iterative Clustering (SLIC) is a superpixel segmentation algorithm that groups pixels into perceptually meaningful regions based on color similarity and spatial proximity in the RGB image.

- **Implementation:** The original RGB image is segmented into approximately 300 superpixels, with a compactness factor of 10 to balance the trade-off between spatial coherence and adherence to image boundaries. The SLIC algorithm iteratively refines cluster centers to minimize a distance metric combining color (in CIELAB space) and spatial coordinates.

- **Refinement Process:** For each superpixel, the corresponding region in the DL-predicted mask is analyzed. The majority class (crack or non-crack) within that superpixel is determined by counting the number of pixels labeled as crack versus non-crack. This majority class is then assigned to all pixels within the superpixel, effectively smoothing the mask and enforcing consistency.

- **Advantages:** SLIC excels at reducing isolated false positives and smoothing jagged boundaries by aligning the mask with the image's natural edges. It is computationally efficient and particularly effective in areas with uniform textures, where cracks are clearly distinguishable from the background.

Region Splitting and Merging

- **Description:** This technique follows a hierarchical approach, initially splitting the image into regions based on a homogeneity criterion and subsequently merging similar adjacent regions to form coherent segments.
- **Implementation:** The process begins by labeling connected components in the DL-predicted mask. Each region in the corresponding grayscale image (converted from RGB) is evaluated for intensity variance. If the variance exceeds a threshold of 0.01, the region is split into two sub-regions using K-means clustering applied to the pixel intensities. In the merging phase, adjacent regions are compared, and those with a mean color difference (in grayscale) below 0.1 are combined.
- **Refinement Process:** After splitting and merging, a morphological cleaning step is applied, including hole filling and the removal of small objects or holes (less than 50 pixels in area). This eliminates minor artifacts and ensures a cleaner mask.
- **Advantages:** This method adapts dynamically to the complexity of the crack structures, making it suitable for images with overlapping cracks or subtle intensity transitions. It enhances boundary precision and reduces false positives by leveraging both the predicted mask and the image's intrinsic features.

Region Growth Clustering

- **Description:** Region Growth Clustering expands regions from predefined seed points based on similarity criteria, effectively growing crack segments to capture fine details.
- **Implementation:** High-confidence seed points are selected from the DL-predicted mask, specifically pixels with a confidence score above 0.8. Starting from these seeds, regions are grown by including 8-connected neighboring pixels in the grayscale image if their intensity difference from the seed is less than 10 units.
- **Refinement Process:** The growth process continues iteratively until no further pixels meet the criterion, resulting in an expanded and refined crack region. Pixels not included in any grown region retain their original non-crack label from the mask.
- **Advantages:** This technique is particularly effective for detecting thin or low-contrast cracks that the DL model may under-segment. By starting from reliable seeds, it enhances connectivity and captures intricate crack branches, improving the mask's completeness.

Mean Shift Clustering

- **Description:** Mean Shift is a non-parametric clustering method that groups pixels in a feature space defined by color and spatial location, iteratively shifting towards regions of higher density.
- **Implementation:** Applied to the original RGB image, Mean Shift uses a color bandwidth of 20 (in RGB space) and a spatial bandwidth of 10 pixels to define the kernel size. Pixels are clustered based on their convergence to local modes in this five-dimensional feature space (three color channels plus x and y coordinates).
- **Refinement Process:** For each resulting cluster, the majority class (crack or non-crack) is computed from the corresponding pixels in the DL-predicted mask. This class is then assigned to all pixels within the cluster, refining the mask by enforcing consistency within each group.

- **Advantages:** Mean Shift provides robust noise reduction and aligns mask boundaries with natural transitions in the image, such as edges or color changes. It performs well across varied conditions, balancing detail preservation with smoothing.

Integration into a Single Methodology These four techniques are unified under a single methodological umbrella as optional post-processing steps applied to the DL-generated masks. Rather than combining them sequentially or in parallel (which could introduce computational complexity or over-refinement), they are treated as alternative refinement strategies within the hybrid framework. After the DL models produce the raw segmentation masks, each technique is applied independently to generate a refined mask. This modular approach allows flexibility, enabling practitioners to select the most appropriate refinement method based on the specific characteristics of the images (e.g., crack width, texture complexity) or computational constraints.

4. Evaluation The effectiveness of the hybrid framework is assessed through both quantitative and qualitative measures.

- **Quantitative Metrics:** For the raw DL predictions, the Dice coefficient and IoU are calculated on the test set to establish a performance baseline. These metrics quantify the overlap between predicted and ground-truth masks, providing a numerical assessment of segmentation accuracy.

- **Qualitative Assessment:** Visual comparisons are conducted between the raw DL masks and the refined masks produced by each clustering technique. These comparisons focus on improvements in boundary delineation, noise reduction, and crack connectivity, offering insights into the practical benefits of refinement.

4.3.5 Results

The hybrid framework was evaluated to demonstrate the baseline performance of the DL models and the qualitative enhancements achieved through clustering-based refinement.

1. Deep Learning Model Performance The raw segmentation masks from the DL models provide the starting point for refinement.

- **DeepLabV3:** On the test set, DeepLabV3 achieved a Dice coefficient of 0.3925 and an IoU of 0.3019. These good performance reflects its ability to capture crack regions efficiently.

- **UNet:** On the test set, UNet recorded a Dice coefficient of 0.0494 and an IoU of 0.0263. Its performance is lower as compared to DeepLabV3 because of training from the scratch and not having a better architecture.

2. Qualitative Evaluation of Refinement Techniques Each clustering technique was applied to the raw masks, and the resulting refinements were visually inspected to assess their impact.

- **SLIC Superpixels:** This method doesn't perform well and gives the more bad segmentation mask for the cracks.

- **Region Splitting and Merging:** This method many times improves the masks by removing the noise from the masks and also makes more better boundry.

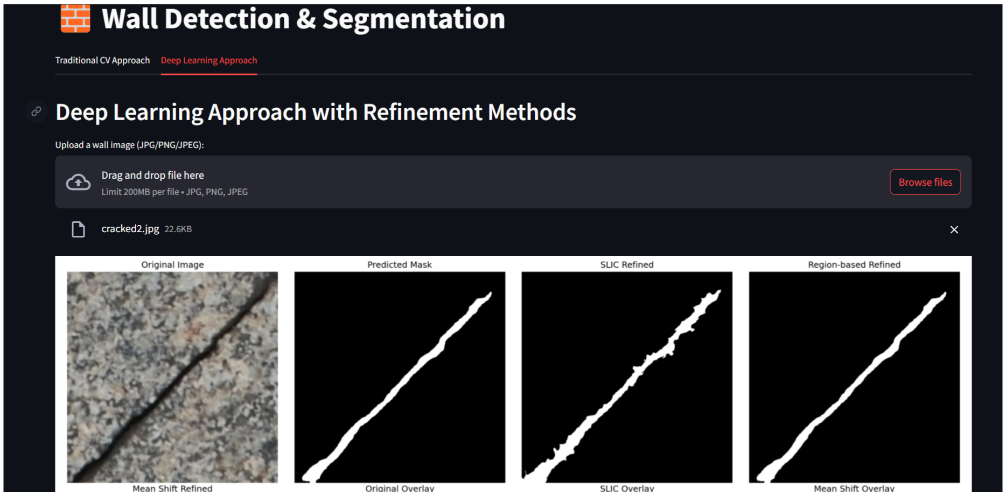
- **Region Growth Clustering:** Don't perform well for small or hairline cracks; but gives satisfactory results on cracks that cover a large area

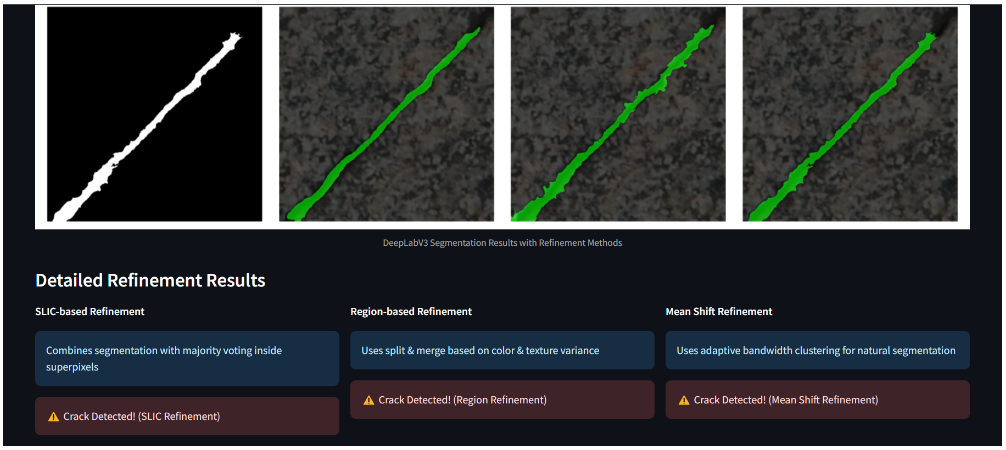
- **Mean Shift Clustering:** Performs above par from the other clustering techniques, but still faces problems in hairline cracks

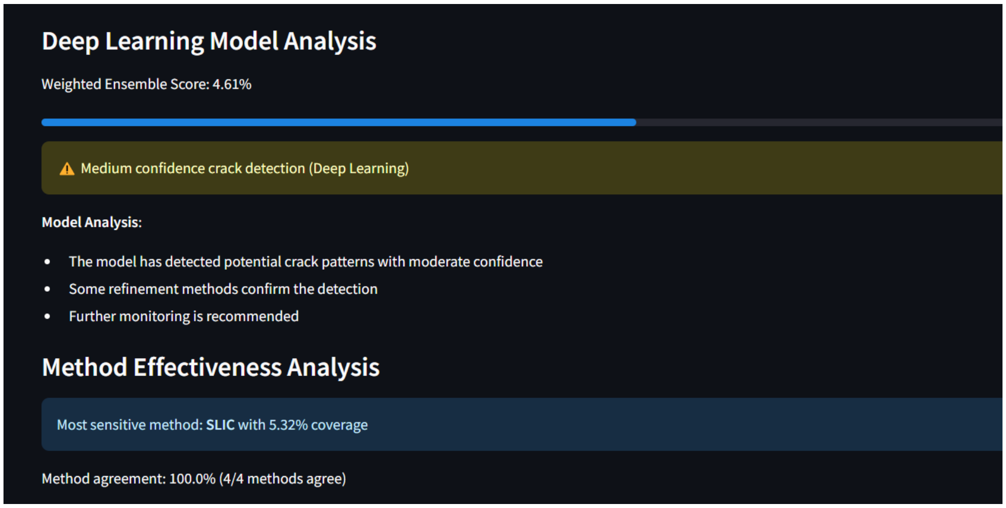
3. Visual Comparisons For the visual comparisons we have overlayed the segmented mask on the image for the better understanding of the crack detection.

5 Deployed Website Screenshot

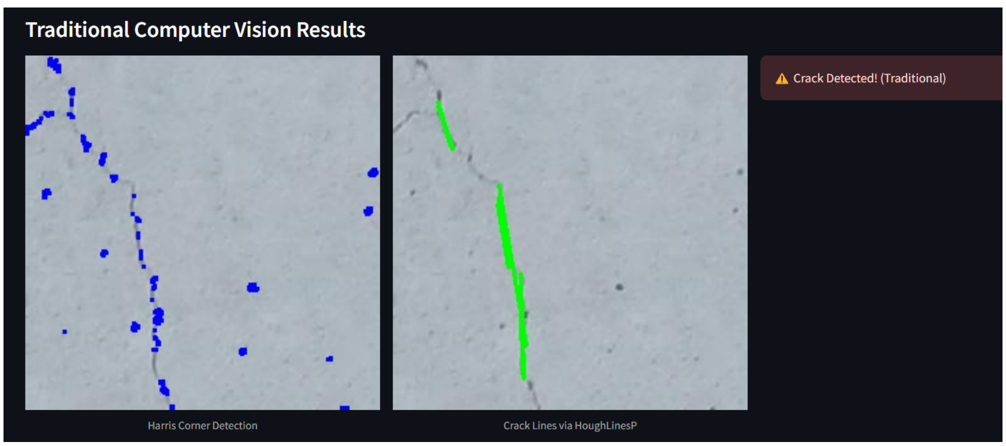
5.0.1 Deep Learning Based Methods

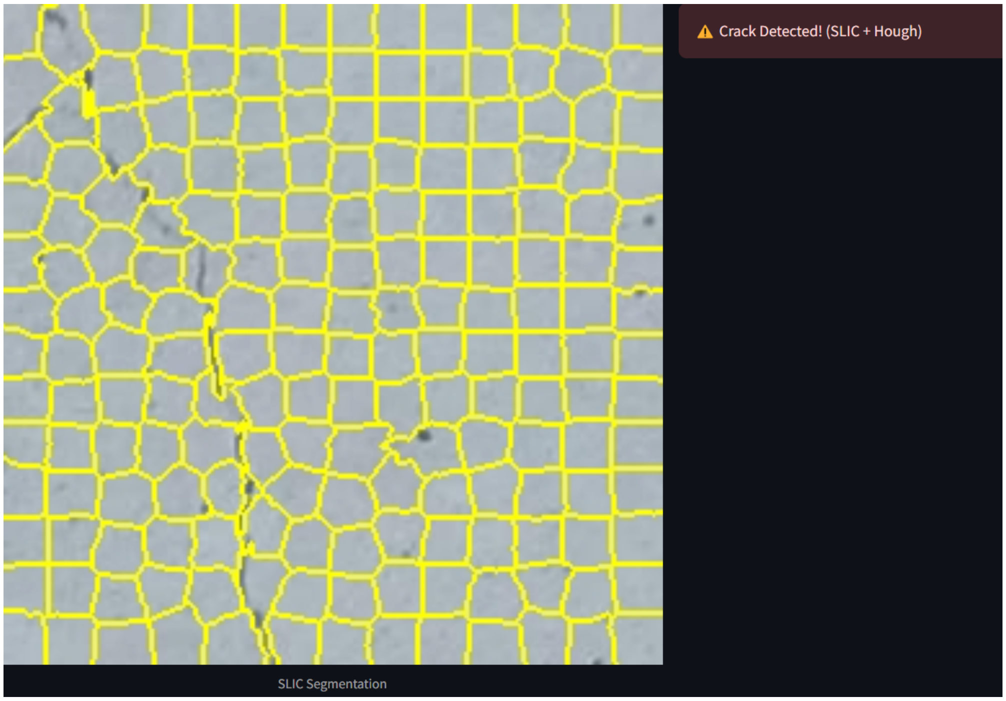


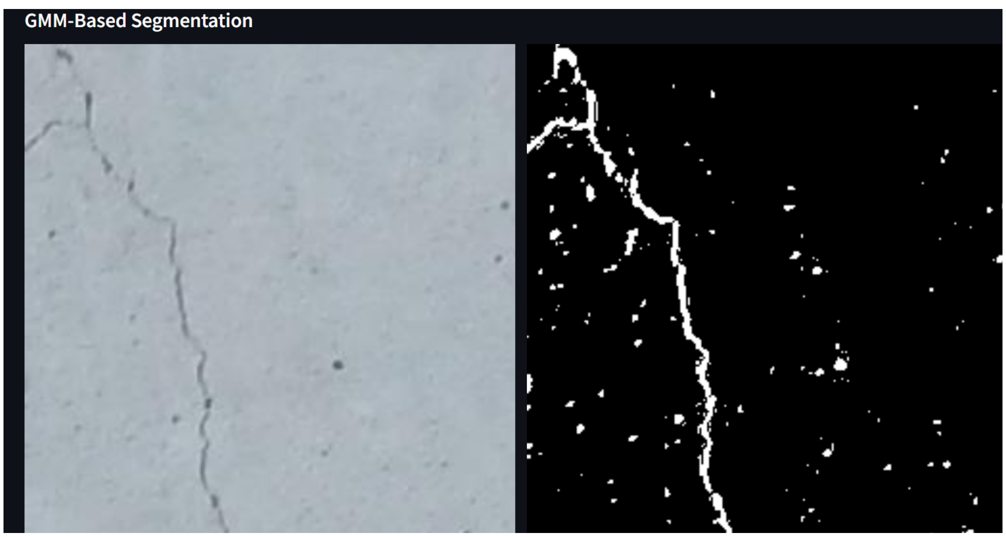


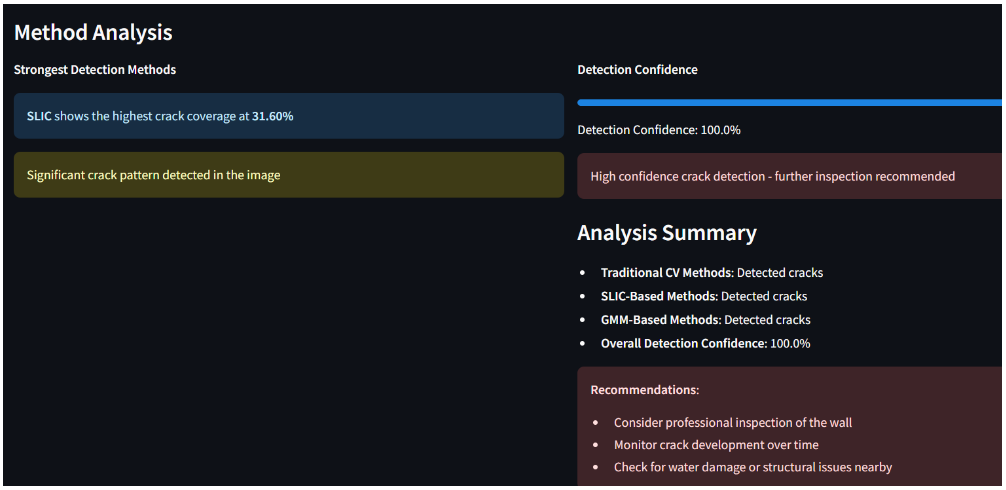


5.0.2 Traditional CV Based Methods




Crack Detected! (SLIC + Hough)
SLIC Segmentation


GMM-Based Segmentation



6 Contributions

- **Swaksh Patwari (B22AI065)**
  Combined the different datasets for the segmentation task. Built two models for segmentation UNet and Deeplabv3, trained a UNet model from scratch, and performed transfer learning on DeepLabv3. Additionally, applied SLIC and Region Splitting and Merging techniques on the output masks by the model. Contributed in making the report. Performed the Literature review on the existing hybrid models.

- **Aansh Chandrakant Dubey (B22AI058)**
  Trained the UNet model and performed transfer learning on DeepLabv3. Applied Region Growth Clustering and Mean Shift Clustering techniques to enhance segmentation performance.Contributed in making the report. Performed the Literature review on the existing hybrid models.

- **Siddhesh Ayyathan (B22CS016)**
  Fine-tuned ResNet model on the SDNET dataset to establish a deep learning benchmark for classification. Applied Gaussian Mixture Model for binary segmentation and deployed the project on HuggingFace.

- **Aditya Padhy (B22CS103)**
  Developed a traditional computer vision pipeline for wall crack detection using Harris, Canny, Hough Transform, SLIC, and region based contour analysis. Proposed a confidence estimation metric based on pixel-level agreement across methods. Integrated the DL and traditional CV pipeline into a Streamlit app with interactive visualizations, including detection overlays and confidence scoring.