

Robust Video-Based Pothole Detection and Area Estimation for Intelligent Vehicles with Depth Map and Kalman Smoothing

Dehao Wang, Haohang Zhu, Yiwen Xu, and Kaiqi Liu

Abstract—Road potholes pose a serious threat to driving safety and comfort, making their detection and assessment a critical task in fields such as autonomous driving. When driving vehicles, the operators usually avoid large potholes and approach smaller ones at reduced speeds to ensure safety. Therefore, accurately estimating pothole area is of vital importance. Most existing vision-based methods rely on distance priors to construct geometric models. However, their performance is susceptible to variations in camera angles and typically relies on the assumption of a flat road surface, potentially leading to significant errors in complex real-world environments. To address these problems, a robust pothole area estimation framework that integrates object detection and monocular depth estimation in a video stream is proposed in this paper. First, to enhance pothole feature extraction and improve the detection of small potholes, ACSH-YOLOv8 is proposed with ACmix module and the small object detection head. Then, the BoT-SORT algorithm is utilized for pothole tracking, while DepthAnything V2 generates depth maps for each frame. With the obtained depth maps and potholes labels, a novel Minimum Bounding Triangulated Pixel (MBTP) method is proposed for pothole area estimation. Finally, Kalman Filter based on Confidence and Distance (CDKF) is developed to maintain consistency of estimation results across consecutive frames. The results show that ACSH-YOLOv8 model achieves an AP(50) of 76.6%, representing a 7.6% improvement over YOLOv8. Through CDKF optimization across consecutive frames, pothole predictions become more robust, thereby enhancing the method’s practical applicability.

Index Terms—Pothole area estimation, object detection, monocular depth estimation, Kalman Filter.

I. INTRODUCTION

AUTONOMOUS driving technology has developed rapidly in recent years, with the enhancement of road safety becoming one of its key objectives. To ensure driving safety and improve ride comfort, autonomous driving systems must accurately perceive and interpret road conditions. Potholes represent a prevalent type of road surface damage, which may adversely affect driving safety [1]. Their formation is influenced by a complex array of factors, including natural elements such as climate change and soil composition [2], as well as human-induced factors such as improper road design, inadequate maintenance, and excessive traffic load [3]. Therefore, it is difficult to predict where potholes will appear on roads.

It is sponsored by State Key Laboratory of Intelligent Green Vehicle and Mobility under Project No. KFY2416. (*Corresponding Author: Kaiqi Liu*)

Dehao Wang is with the School of Integrated Circuits and Electronics, Beijing Institute of Technology, Beijing 100081, China (e-mail: wonderhow@bit.edu.cn).

Haohang Zhu, Yiwen Xu and Kaiqi Liu are with the School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China (e-mail: 1120221140@bit.edu.cn; 1120221068@bit.edu.cn; liukaiqi@bit.edu.cn).

Pothole is one of the major causes of traffic accidents. According to the British Automobile Association, 631,852 pothole-related accidents were reported in 2022, marking a five-year high [4]. Similarly, the Chicago Sun-Times reported 3,597 traffic accidents caused by potholes in the first two months of 2018 alone [5]. In addition to compromising road safety, potholes also negatively impact passenger comfort as vehicles traverse these damaged road surfaces [6]. Therefore, real-time detection of road potholes has become a critical area of research [7].

While extensive research has focused on detecting and localizing potholes, further estimating their area provides even greater practical value for real-world applications [8]. From a safety perspective, the size of a pothole directly influences the choice of obstacle avoidance strategies [9]. For instance, the vehicle can maintain its course when encountering small potholes, but large potholes might necessitate rerouting or emergency braking to ensure safety. From a comfort perspective, estimating pothole area allows vehicles to determine whether to reduce speed, thereby avoiding severe jolts caused by traversing large potholes at high speeds [10]. With the rapid development of connected vehicle technologies [11], information about pothole areas can be shared across intelligent traffic systems, enabling other autonomous vehicles to plan routes more efficiently and reduce traffic congestion. Furthermore, analyzing changes in pothole area over time can provide valuable insights into road aging trends, supporting road maintenance and urban planning efforts [12].

Research on pothole area estimation generally follows two main approaches. The first involves using LiDAR or similar sensors to obtain 3D point cloud data, from which pothole areas are calculated. The second relies on purely vision-based methods that estimate area using object detection and pre-defined geometric models. However, the former often suffers from high computational costs, while the latter is highly sensitive to camera angles and struggles to perform well on complex terrains. Moreover, most existing methods process only single-frame data, whereas leveraging video streams for pothole area estimation has the potential to significantly enhance robustness and holds strong promise for practical applications.

To improve the accuracy and robustness of pothole area estimation while enhancing processing efficiency and reducing costs using only 2D images, a novel and robust pothole area estimation framework is proposed. The framework is built upon a newly designed pothole detection model, ACSH-YOLOv8, combined with the advanced monocular metric depth estimation network, DepthAnything V2. A novel Minimum Bounding Triangulated Pixel (MBTP) method is introduced to estimate pothole areas with improved reliability. To reduce the

impact of factors such as lighting variation and camera motion, the Kalman Filter based on Confidence and Distance (CDKF) algorithm is proposed, which leverages consecutive video frames and adjusts estimations based on detection confidence and distance between pothole and camera.

The main contributions of the paper are as follows:

- 1) In order to achieve high-accuracy and robust pothole area estimation, this paper proposes a novel pothole detection and area estimation framework, where a dedicated MBTP method is introduced as the core module for precise pothole area estimation by integrating pothole regions with the depth map.
- 2) To enhance the model's detection accuracy for potholes with varying scale and blur, the ACSH-YOLOv8 model is proposed by adding an additional detection head for small potholes, and incorporating a hybrid attention mechanism, ACmix, in the neck of the architecture to improve detail awareness.
- 3) To enhance the robustness of area estimation in video streams, the CDKF method is proposed, which refines area estimates based on pothole tracking results, utilizing detection confidence and distance as optimization.

The remainder of the paper is organized as follows. Section II provides a review of related work on pothole detection and pothole area estimation. Section III provides a detailed explanation of the core methodology for pothole area estimation, covering pothole detection, tracking, depth estimation, area calculation, and consecutive frame optimization. Section IV introduces the dataset, evaluation metrics, and experimental setup for both the detection model and area estimation algorithm. Section V presents a quantitative and visual analysis of the results, and the work of this paper is summarized in Section VI.

II. RELATED WORKS

A. Pothole Detection

There is considerable research on pothole detection, including both traditional machine learning algorithms and deep learning approaches. Traditional machine learning methods such as Otsu's thresholding [13], spectral clustering [14], and morphological operations [15] are used to extract and identify potential pothole regions. While these algorithms have the advantage of lower computational load, their classification performance and robustness are often limited. Some studies employ 3D point cloud data, using surface normal information for pothole geometric modeling [16]. However, 3D point cloud data acquisition is often costly and computationally demanding. Nowadays, with the rapid development of deep learning technologies, numerous CNN-based deep learning networks for object detection are proposed, creating significant opportunities for pothole detection development. These methods significantly enhance the accuracy and robustness of pothole detection, and the localization of potholes is obtained precisely [17]. Among pothole detection algorithms, the one-stage algorithm You Only Look Once (YOLO) [18] gains widespread application due to its high accuracy and real-time

processing capabilities. Ukhwah [19] demonstrates the effectiveness of YOLOv3 and its variants in road pothole detection. Shaghouri [20] introduces CSPDarknet53 as a backbone based on YOLOv4, achieving a balance between accuracy and speed. Mahalingesh [21] integrates the YOLOv8 algorithm and deploys it on a Raspberry Pi for hardware testing, highlighting the significant potential of YOLO-based algorithms in practical applications.

B. Pothole Area Estimation

Current research on pothole area estimation falls into two main approaches: one involves obtaining 3D point cloud arrays for area estimation, and the other uses 2D images and image processing techniques. In terms of 3D point cloud analysis, Ravi [22] utilizes LiDAR to capture road point clouds and applies a vehicle motion mapping model to achieve high-precision pothole area estimation. Chen [23] employs drones to slice images into 3D point clouds and proposes the UAV-Structure-from-Motion algorithm, which uses motion sensing for pothole area estimation. Although these methods achieve high accuracy, they rely on motion models, which may introduce significant errors in trajectory and speed control if the motion varies greatly. This reduces the accuracy of the estimation and limits their practical applicability. Additionally, using LiDAR or high-resolution images to generate point clouds requires considerable computational resources, making these methods costly and unsuitable for real-time applications.

For 2D image processing, most research focuses on proposing new area estimation methods based on object detection algorithms. Heo [24] uses a pinhole camera model and prior distance equations to estimate pothole areas. Kharel [25] employs Inverse Perspective Mapping (IPM) to convert camera intrinsic parameters and estimate pothole areas. Chitale [26] applies distance priors and triangle similarity to estimate pothole areas. However, these methods strongly depend on geometric models and are highly sensitive to the camera's viewing angle, making them less adaptable for widespread use. Furthermore, these methods operate under the assumption of planar road surfaces, whereas potholes predominantly form in complex, non-uniform terrains. Conventional prior models fail to adequately represent these geometric irregularities, resulting in significant estimation inaccuracies.

III. METHOD

The proposed framework for pothole detection and area estimation is illustrated in Fig. 1. First, video streams captured by the vehicle's front camera are fed into ACSH-YOLOv8 object detection model to localize potholes and extract their bounding boxes. Next, BoT-SORT algorithm is applied to track the detected potholes across consecutive frames, assigning a unique ID to each pothole to ensure tracking consistency. Simultaneously, the video stream is processed by the pre-trained monocular metric depth estimation model, DepthAnything V2, which generates corresponding depth maps. Subsequently, by combining the object detection results with depth maps, MBTP algorithm is introduced with a pinhole camera model-based 3D mapping and faced-based method to estimate the pothole

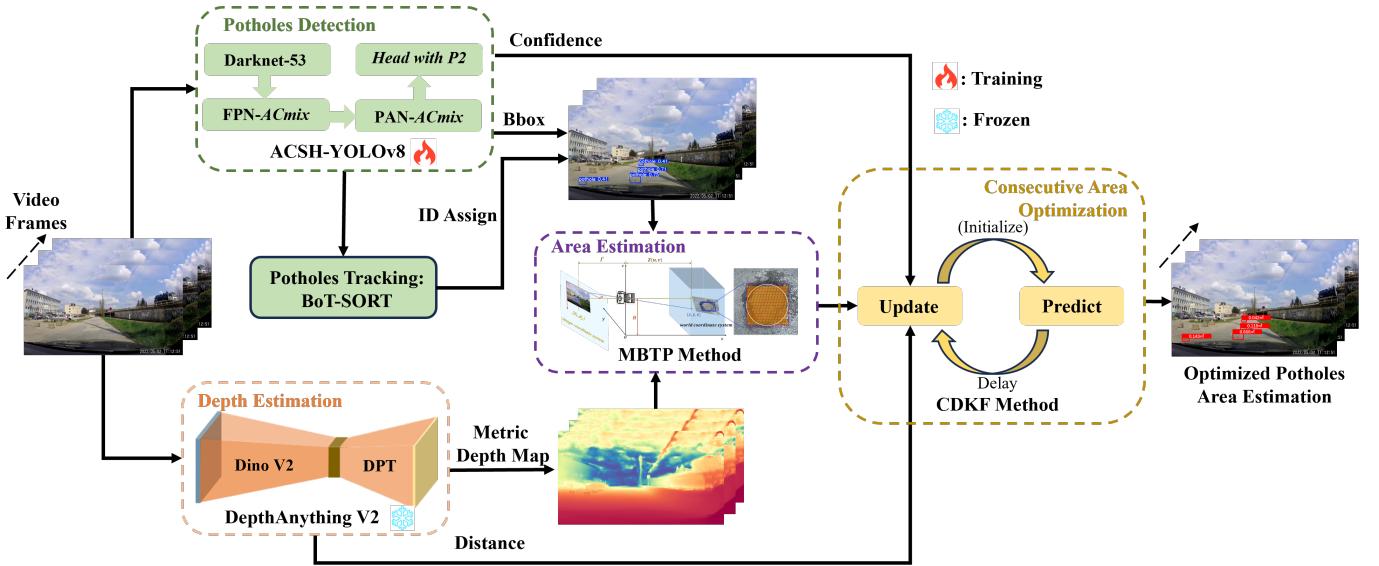


Fig. 1. Overall flowchart of the proposed pothole area estimation model. The Proposed ACSH-YOLOv8 model is trainable, while DepthAnything V2 uses a pre-trained model.

area. Finally, to enhance the robustness of the system, the potential fluctuations in the estimated area of the same pothole across consecutive frames are constrained by Kalman Filter based on Confidence Distance (CDKF), which incorporates the detection confidence and the distance between the pothole and the camera as uncertainty factors into a novel Kalman filtering algorithm, ultimately yielding an optimized pothole area estimation.

A. Potholes Detection

Object detection algorithms are commonly used for pothole detection to obtain bounding boxes of the target regions [20], [21]. Given the varying sizes of potholes and their differing distances from the camera, detecting potholes accurately and consistently remains a challenging task. To address this, a novel pothole detection model named ACSH-YOLOv8 is proposed. ACSH-YOLOv8 introduces two key innovations: the addition of a small object detection head to improve detection across different scales, particularly small and distant potholes, and the integration of the ACmix attention mechanism [27] in the Neck to better focus on pothole-relevant features. The overall architecture is illustrated in Fig. 2.

The model consists of three main components: Backbone, Neck, and Head. The Backbone adopts the CSPDarknet structure and replaces the C3 module (used in earlier models) with the more lightweight C2f module, enhancing gradient flow while reducing computational complexity. The Neck utilizes a Feature Pyramid Network (FPN) and Path Aggregation Network (PAN) pyramid structure for multi-scale feature fusion, enabling the model to capture targets at varying scales. The Head features a decoupled design with separate branches for classification and localization, employing an anchor-free approach for bounding box prediction.

To further enhance detection of small-scale potholes, a dedicated P2 detection head is introduced. In contrast to standard detection heads operating on downsampled feature

maps (P3: 80×80, P4: 40×40, P5: 20×20), the P2 head operates on a high-resolution 160×160 feature map. This is achieved by upsampling intermediate features in the Neck and fusing them with shallow features from the Backbone. As a result, the model retains more fine-grained visual details and significantly improves the detection of small and distant potholes, showcasing strong potential for real-world road surface analysis.

To better focus on pothole features, we introduce the ACmix module in Neck part (including FPN and PAN), a hybrid feature extraction module that combines self-attention and convolution [27]. Its structure is shown in Fig. 3. The process begins by applying three 1×1 convolution layers to obtain three distinct feature maps. These feature maps are then processed using Shift Operation and Self Attention. The Shift Operation first uses a fully connected layer to map the features and then applies a shift operation, similar to convolution, to aggregate the features. The Self Attention mechanism divides the extracted features into Query, Key, and Value, and uses the attention mechanism to extract key information. The specific computation formulas are as follows:

$$\mathbf{F}_{att} = \text{softmax} \left(\frac{(\mathbf{W}_q^{(l)} f_{ij})^\top (\mathbf{W}_k^{(l)} f_{ij})}{\sqrt{d}} \right) (\mathbf{W}_v^{(l)} f_{ij}) \quad (1)$$

where $\mathcal{N}_k(i, j)$ represents a local pixel region centered at pixel (i, j) with a spatial width of k ; f_{ij} denotes the tensor input for pixel (i, j) , and \mathbf{W}_q , \mathbf{W}_k , and \mathbf{W}_v are the mapping matrices for the three corresponding features. d denotes the feature dimension of $\mathbf{W}_q^{(l)} f_{ij}$. The final feature \mathbf{F}_{att} is obtained based on self-attention. The final output is obtained by aggregating and summing the results of the Shift Operation and Self Attention, as shown below.

$$\mathbf{F}_{out} = \alpha \mathbf{F}_{conv} + \beta \mathbf{F}_{att} \quad (2)$$

where \mathbf{F}_{conv} is the feature obtained from the Shift Operation, and α and β are learnable weights.

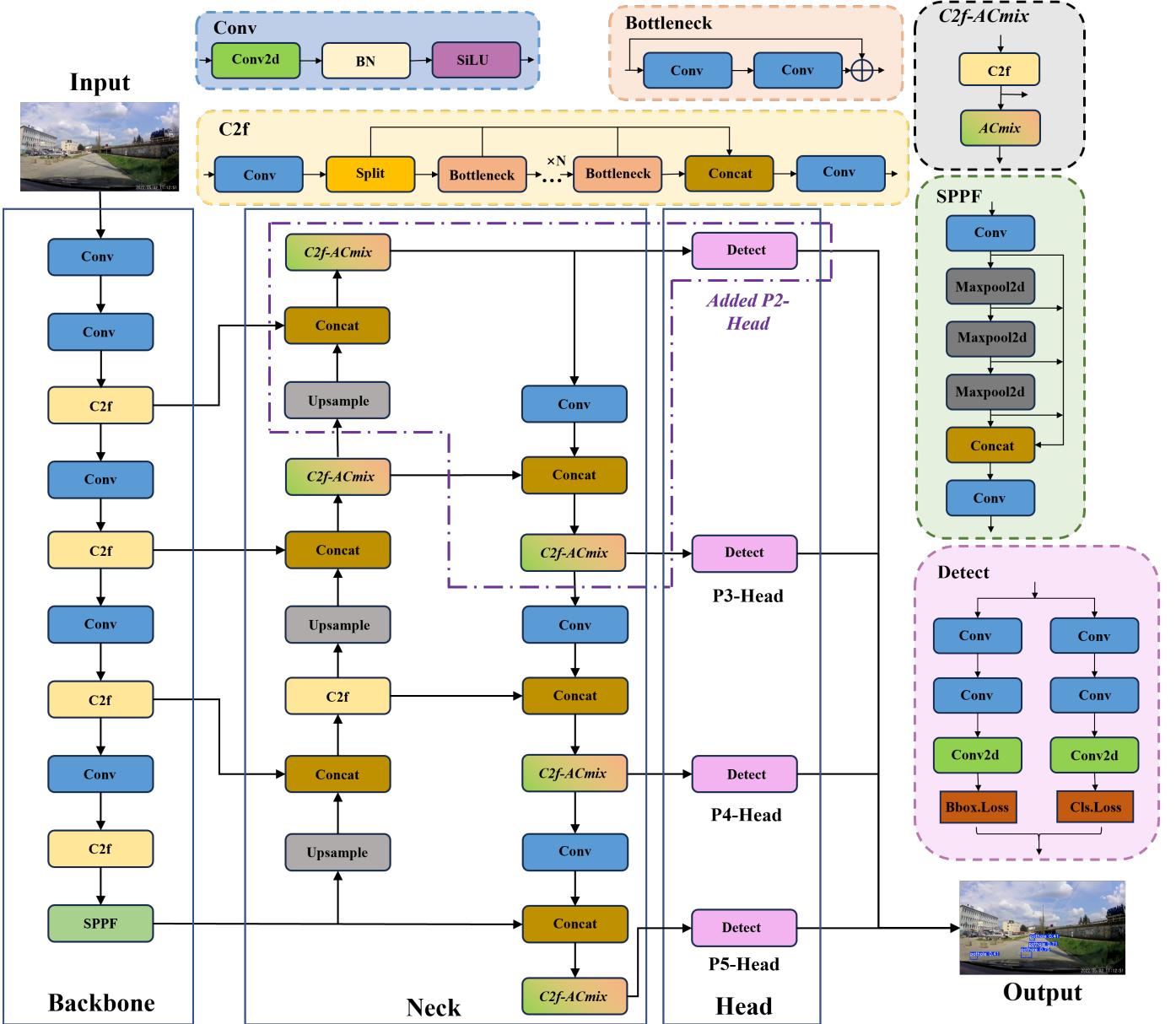


Fig. 2. The ACSH-YOLOv8 model is used for pothole detection, incorporating the P2 detection head and the ACmix hybrid attention mechanism, which are highlighted in italics in the figure.

B. Potholes Tracking

To track and assign consistent IDs to the same pothole across successive frames, we employ the BoT-SORT algorithm [28], which is well-suited for object tracking under vehicle-mounted camera motion. To ensure robust tracking, ego-motion is compensated for using sparse optical flow [29], which estimates global scene motion caused by camera shifts such as rolling, pitching, and translation. Specifically, the corner keypoints \mathbf{p}_i^{k-1} are detected in frame $k-1$ and tracked into frame k as \mathbf{p}_i^k by minimizing:

$$\Delta\mathbf{p}_i = \arg \min_{\Delta\mathbf{p}} \sum_{\mathbf{q} \in \Omega_i^{k-1}} [I_k(\mathbf{q} + \Delta\mathbf{p}) - I_{k-1}(\mathbf{q})]^2 \quad (3)$$

where I_k denotes frame k 's intensity and Ω_i^{k-1} is a local patch around keypoint \mathbf{p}_i^{k-1} . A RANSAC procedure then fits a transformation matrix \mathbf{T} to the inlier flow vectors [30], isolating the dominant camera motion. Each detected pothole bounding box $\mathbf{z}_k = (x_k, y_k, w_k, h_k)$ is compensated by transforming its center (x_k, y_k) according to equation:

$$\begin{pmatrix} x'_k \\ y'_k \\ 1 \end{pmatrix} = \mathbf{T}^{-1} \begin{pmatrix} x_k \\ y_k \\ 1 \end{pmatrix} \quad (4)$$

thereby reducing apparent motion due to vehicle movement. The updated bounding box $\mathbf{z}'_k = (x'_k, y'_k, w_k, h_k)$ then serves as input for the subsequent tracking steps.

Each track is represented by an eight-dimensional Kalman filter state $\mathbf{x}_k = (x, y, w, h, \dot{x}, \dot{y}, \dot{w}, \dot{h})^\top$, where (x, y) specifies

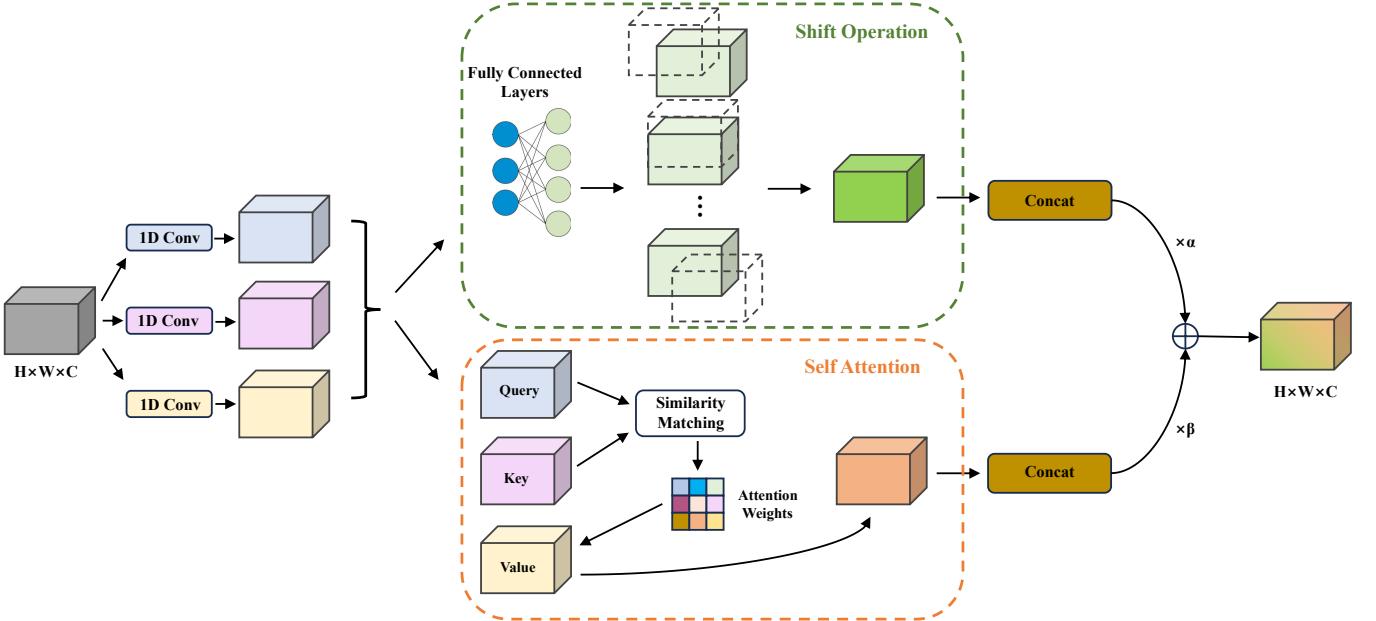


Fig. 3. Schematic diagram of the ACmix hybrid attention mechanism.

the bounding-box center, (w, h) specifies the width and height, and $(\dot{x}, \dot{y}, \dot{w}, \dot{h})$ denotes the corresponding velocities. The state is propagated using a constant-velocity transition matrix $\mathbf{F} \in \mathbb{R}^{8 \times 8}$, which can be partitioned into sub-blocks for position and velocity.

Given the predicted state from the previous frame, the Kalman filter uses a constant-velocity model to project the current state and its uncertainty forward. Upon receiving a new bounding box detection, the filter updates the predicted state by incorporating the measurement, adjusting both the estimate and the associated uncertainty based on the Kalman gain. This process refines the position, size, and velocity estimates of the tracked object while mitigating measurement noise and residual motion, thereby stabilizing the pothole trajectory across frames.

A two-stage data association procedure is then performed on the predicted states. In the first stage, detection scores exceeding a predefined threshold are labeled as high-confidence, while the rest are considered low-confidence. An assignment matrix $\mathbf{X} \in \{0, 1\}^{N \times M}$ is computed by minimizing the total IoU-based cost:

$$\min_{\mathbf{X}} \sum_{i=1}^N \sum_{j=1}^M (1 - \text{IoU}(\text{box}_i, \text{det}_j)) \quad (5)$$

where IoU measures the overlap between track i 's predicted bounding box and detection j . The optimization is subject to one-to-one assignment constraints, ensuring that each track is matched to at most one detection and vice versa. The Hungarian algorithm [31] is used to solve this assignment problem. In the second stage, unmatched tracks are associated with low-confidence detections under a relaxed threshold to recover occluded or ambiguous potholes. Tracks that remain unmatched over several frames are deleted, while unmatched detections initialize new tracks. This approach enables robust identity maintenance without relying on appearance features.

C. Monocular Depth Estimation

To reduce the reliance on costly 3D sensors [32], DepthAnything V2 [33], a monocular metric depth estimation model pre-trained on the KITTI dataset, is employed. It is shown to generalize well in outdoor scenes, making it suitable for estimating both the absolute distance and relative depth of potholes. The model is composed of a self-supervised visual backbone (DINO V2) and a dense prediction head (DPT). Global and local features are extracted from monocular images through transformer-based layers in the backbone, and multi-scale features are fused by the DPT head to reconstruct a full-resolution depth map. This design allows both large-scale scene structure and fine-grained pothole geometry to be captured effectively, which is essential for accurate area estimation. To enable practical deployment, a distilled version of the model is used, where the feature dimensions are reduced, and the model size is compressed from 4439.5MB to 86.2MB while maintaining high accuracy. Detailed architecture and processing flow can be found in Fig. 4.

D. Potholes Area Estimation

To estimate the pothole area based on the previously obtained object detection bounding boxes and monocular metric depth estimation maps, we propose the MBTP method that combines the 3D minimum bounding rectangle and the pixel-based triangular area accumulation. The process is illustrated in Fig. 5. First, based on the pinhole camera model, the image plane pixel coordinates (u, v) are converted into 3D points $(X_{u,v}, Y_{u,v}, Z_{u,v})$ in the camera coordinate system. Ignoring the effects of image distortion to simplify the model, the projection relationship from pixel (u, v) to the 3D coordinates $(X_{u,v}, Y_{u,v})$ can be calculated using following formulas.

$$X_{u,v} = \frac{(u - p_u)}{f_u} Z_{u,v} \quad (6)$$

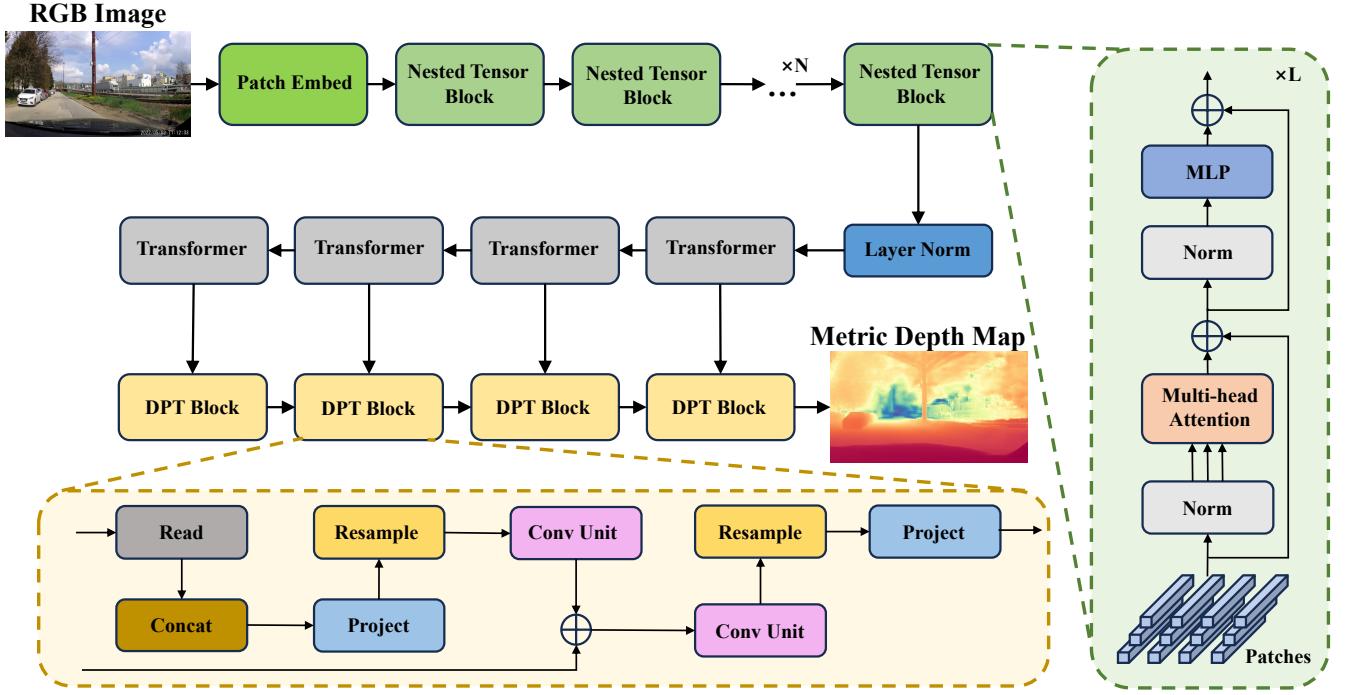


Fig. 4. Schematic diagram of the monocular depth model, DepthAnything V2, used to obtain the depth values of pothole pixels.

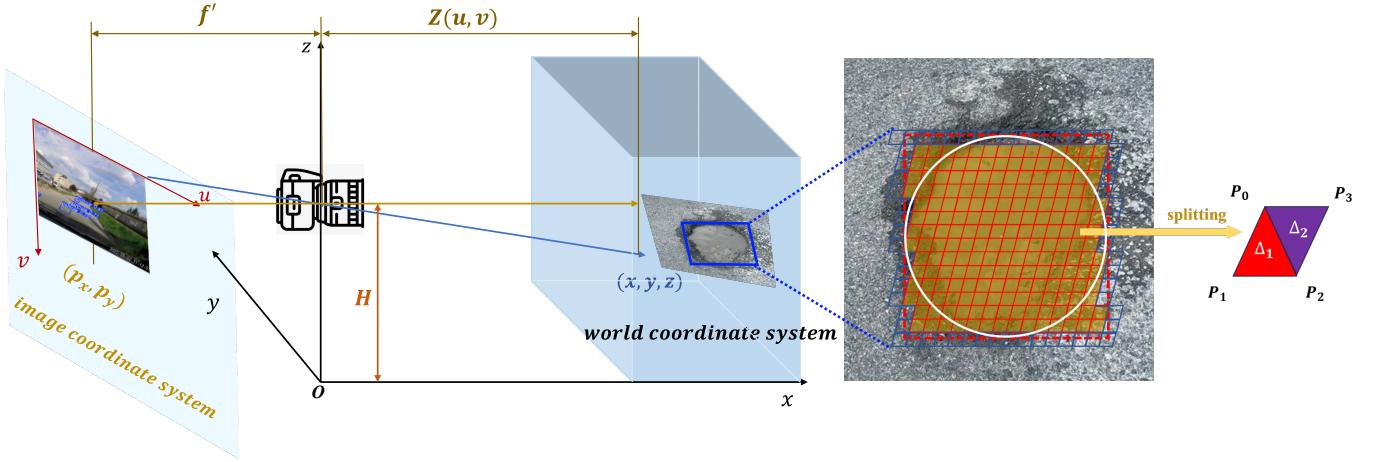


Fig. 5. Schematic diagram of the proposed MBTP method.

$$Y_{u,v} = \frac{(v - p_v)}{f_v} Z_{u,v} \quad (7)$$

where p_u and p_v represent the pixel coordinates of the camera's optical center, while f_u and f_v are the focal lengths in pixels along the horizontal and vertical axes, respectively. $Z_{u,v}$ indicates the depth value corresponding to each pixel (u, v) in the depth estimation map. For a given object bounding box region, denoted as \mathcal{R}_{uv} in the image coordinate system, the projection of all pixels in this region onto the 3D space plane (X, Y) is obtained.

$$\mathcal{D} = \{(X_{u,v}, Y_{u,v}) \mid (u, v) \in \mathcal{R}_{uv}\} \quad (8)$$

To facilitate pixel-based calculations, the minimum bounding rectangle for the region is computed. By calculating the

minimum and maximum values in the \mathcal{D} , and rounding down to avoid overestimation, the rectangular region is determined.

$$\text{Rect}_{XY} = [\min X, \max X] \times [\min Y, \max Y] \quad (X, Y) \in \mathcal{D} \quad (9)$$

Within the resulting rectangular region Rect_{XY} , the area is subdivided into segments formed by adjacent groups of 2×2 pixels. The 3D projection coordinates of these four neighboring points are represented as $P_0 = (X_{u,v}, Y_{u,v})$, $P_1 = (X_{u+1,v}, Y_{u+1,v})$, $P_2 = (X_{u,v+1}, Y_{u,v+1})$, $P_3 = (X_{u+1,v+1}, Y_{u+1,v+1})$. To simplify further area calculations, each diamond is split into two triangles, $\Delta_1(P_0, P_1, P_2)$ and $\Delta_2(P_0, P_2, P_3)$. The area of each triangle is then calculated

using the vector cross product formula as described below.

$$\text{Area}(\Delta) = \frac{1}{2} |(x_2 - x_1)(y_3 - y_1) - (y_2 - y_1)(x_3 - x_1)| \quad (10)$$

In this context, x and y denote the coordinate components of the three vertices of each triangle. By summing the areas of the two triangles, we obtain the area of the corresponding 2×2 pixel block, denoted as A_{uv}^{patch} . The total area of the rectangular region S_{final} is then determined by accumulating the areas of all valid blocks within that region. Since most potholes are elliptical, the final area is approximated by multiplying the rectangular area by a coefficient of $\frac{\pi}{4}$.

$$A_{u,v}^{\text{patch}} = \text{Area}(\Delta_1) + \text{Area}(\Delta_2) \quad (11)$$

$$S_{\text{final}} = \left(\sum_{\substack{(u,v) \in \mathcal{R}_{uv} \\ P_i \in \text{Rect}_{XY}}} A_{u,v}^{\text{patch}} \right) \times \frac{\pi}{4} \quad (12)$$

E. Consecutive Frame Area Optimization

To address the uncertainty and noise interference in estimating pothole areas from video frames, this paper proposes Kalman Filter based on Confidence and Distance (CDKF), a robust estimation method based on Kalman filtering. The method leverages the Kalman filter's predict-update mechanism to recursively estimate and dynamically smooth the pothole area state. Additionally, it incorporates an adaptive measurement noise adjustment strategy based on detection confidence and distance to achieve more robust estimation in clear and dark environments.

For an individual pothole, it is assumed that its area remains constant across consecutive frames, conforming to a constant state model. Due to unavoidable noise introduced during processes such as bounding box detection and monocular depth estimation, process noise covariance is incorporated to update the state uncertainty. In the prediction step, the state and covariance are updated as follows:

$$\mathbf{A}_{k|k-1} = \mathbf{A}_{k-1}, \mathbf{P}_{k|k-1} = \mathbf{P}_{k-1|k-1} + \mathbf{Q} \quad (13)$$

where \mathbf{A}_k denotes the estimated pothole area in the k th frame, \mathbf{P} represents the state uncertainty, and \mathbf{Q} is the process noise covariance. During the measurement update phase, the predicted state is adjusted using the pothole area measurement obtained from the current frame detection. The update of the Kalman filter result relies on the Kalman gain, which is computed using the following formula:

$$\mathbf{K}_{k-1} = \frac{\mathbf{P}_{k|k-1}}{\mathbf{P}_{k|k-1} + \mathbf{R}_{k-1}} \quad (14)$$

In this context, \mathbf{K} represents the Kalman gain, which determines the degree of trust placed in the current observation, while \mathbf{R} denotes the measurement noise covariance. Compared to conventional Kalman filter methods, a key innovation of this approach is the dynamic adjustment of \mathbf{R} . This adjustment considers two crucial factors affecting measurement accuracy: the confidence level of the bounding box from the object detection algorithm and the distance between the pothole and the camera. Specifically, a higher bounding box confidence

and a closer proximity to the camera both contribute to higher measurement accuracy. Based on these considerations, the measurement noise covariance is determined using the following equation:

$$\mathbf{R}_{k-1} = \frac{\lambda}{c} + \theta \cdot \max\{d, d_0\} \quad (15)$$

where d denotes the distance from the center of the pothole bounding box to the camera, and c represents the confidence level of the detection bounding box. Since the pothole area estimates are generally more reliable at closer ranges, a trusted distance range d_0 is defined, within which \mathbf{R} remains unaffected by changes in distance. The parameters λ and θ serve as tuning factors to balance the influences of both the confidence and distance on \mathbf{R} . Ultimately, the Kalman filter state update and covariance correction are expressed as follows:

$$\mathbf{A}_k = \mathbf{A}_{k|k-1} + \mathbf{K}_{k-1}(z_k - \mathbf{A}_{k|k-1}) \quad (16)$$

$$\mathbf{P}_k = (1 - \mathbf{K}_{k-1})\mathbf{P}_{k|k-1} \quad (17)$$

In the above equation, $(z_k - \mathbf{A}_{k|k-1})$ represents the measurement residual (innovation), whose magnitude reflects the inconsistency between the observed value and the prior prediction.

IV. EXPERIMENTAL EVALUATIONS

A. Dataset Construction

1) *Dataset Configuration:* The dataset used in this study originates from previous research by Bučko [34]. It consists of images captured by a camera mounted on the front side of a vehicle. The overall dataset collection includes several datasets captured under different times of day and weather conditions. For this study, two representative datasets were selected: the Clear Road Dataset, representing clear weather conditions, and the Dark Road Dataset, representing low-light conditions such as dusk and nighttime. These two datasets were used separately for training and evaluation.

The Clear Road Dataset contains 1,052 images, including 1,896 pothole instances and 232 manhole instances. The Dark Road Dataset comprises images taken during dusk and nighttime, with 250 and 310 images respectively, totaling 560 images. This subset includes 506 pothole instances and 95 manhole instances. All images have a resolution of 1920×1080.

The statistical distributions of pothole targets in the Clear Road Dataset and Dark Road Dataset are shown in Fig. 6 and Fig. 7, respectively. These figures provide a representative overview of the potholes captured under different conditions in reality. In each figure, the left side illustrates the distribution of pothole center points across the image, while the right side shows the normalized width and height distributions of the detected targets. The two datasets exhibit similar patterns. Most potholes are relatively small in size, with their widths and heights concentrated between 0.02 and 0.06 along the normalized horizontal and vertical axes. This highlights the

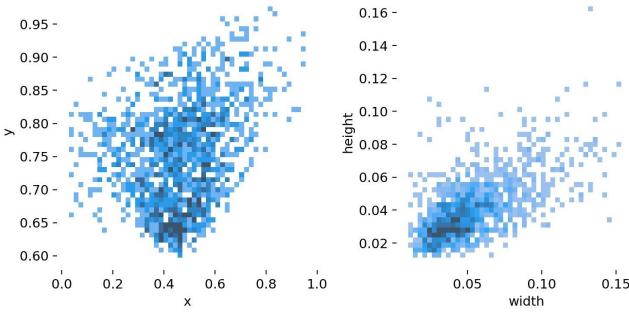


Fig. 6. Distribution of object center and size in Clear Road Dataset.

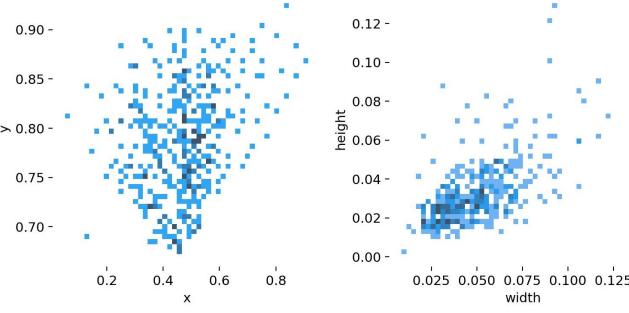


Fig. 7. Distribution of object center and size in Dark Road Dataset.

inherent challenge of small object detection in pothole scenarios. Due to occlusion from the vehicle's front end, the lower portion of the image becomes a detection blind spot, resulting in most potholes being detected in the upper region. At closer distances, potholes are primarily detected near the center of the image, as detection on the sides is often hindered by occlusions or poor lighting conditions. The vertical distribution of potholes ranges from 0.6 to 0.95, indicating that the dataset captures potholes at varying distances—from far to near. This makes it suitable for validating the proposed continuous-frame pothole area estimation and optimization algorithm.

2) *Data Augmentation*: In order to improve the robustness and accuracy of the pothole detection model, we apply extensive data augmentation techniques as follows. Firstly, data augmentation methods include horizontal flipping, applied to half of the images, leveraging the inherent symmetry of potholes to effectively double the dataset. To simulate variations in lighting and environmental conditions, images are transformed into the HSV color space with controlled adjustments: hue varies by 1.5%, saturation by 70%, and brightness by 40%. Additionally, image scaling within a 50% range is implemented to mimic potholes appearing at different distances and sizes, enhancing the model's flexibility. Furthermore, scaling at various distances imitates potholes captured from diverse viewpoints, improving the model's responsiveness to varying object sizes. The Mosaic augmentation technique proves particularly beneficial for improving the detection accuracy of smaller potholes. It combines four randomly selected images into one composite image, enriching the context and diversity presented to the model during training. These strategies collectively aim to boost the generalizability and reliability of the

pothole detection model across varying operational scenarios.

B. Training and Evaluation of Pothole Detection

1) *Pothole Detection Evaluation Metrics*: To evaluate the performance of pothole detection and demonstrate the effectiveness of proposed ACSH-YOLOv8 over the baseline model, we introduce multiple evaluation metrics, including precision, recall, AP (50), AP (50-95), and GFLOPs. Since the primary focus is on pothole detection, the detection results for manholes in the dataset are not included in the evaluation. The formulas for calculating precision and recall are as follows:

$$P = \frac{TP}{TP + FP}, R = \frac{TP}{TP + FN} \quad (18)$$

where P represents precision, R represents recall, TP represents the number of correctly detected samples, FP denotes the number of falsely detected samples, and FN refers to the number of missed detections. Setting the Intersection over Union (IoU) threshold at 70%, a predicted bounding box is considered a true positive only if its IoU with the ground truth exceeds this threshold. To balance precision and recall, we use the F1-score and AP (Average Precision) metric, which are calculated as follows:

$$F_1 = \frac{2 \cdot P \cdot R}{P + R} \quad (19)$$

$$AP = \int_0^1 P(R) dR \quad (20)$$

The F1 score is a harmonic mean of precision and recall, providing a balanced measure for evaluating classification models, especially in imbalanced datasets. The AP metric represents the integral area under the Precision-Recall (P-R) curve. AP is calculated based on different IoU thresholds, with two commonly used variants: AP(50) and AP(50-95). AP(50) refers to the AP score computed with a fixed IoU threshold of 0.5, providing a more lenient evaluation of detection performance. In contrast, AP(50-95) is a more stringent metric that averages AP scores calculated at IoU thresholds ranging from 0.5 to 0.95, in increments of 0.05. This comprehensive evaluation better reflects the model's localization accuracy. Additionally, to assess the computational complexity of the model, we use GFLOPs, which measures the number of floating-point operations required for a single forward inference.

2) *Model Training Configuration*: During model training, the input image size is set to 1080×1080 to ensure that fine details of potholes can be effectively captured. The batch size is set to 4. Training is performed using the SGD momentum optimizer, with a momentum value of 0.937 and a weight decay of 0.0005. The initial learning rate is set to 0.01, gradually decreasing to a final learning rate of 0.0001. A warmup training strategy is applied, with the first 3 epochs dedicated to warmup training. During this phase, the momentum is set to 0.8, and the bias learning rate is 0.1.

Both training and validation are conducted using the PyTorch 2.3.1 deep learning framework with CUDA version 12.6. The hardware setup includes an RTX 4090 GPU with 24GB of VRAM and 16 vCPU Intel Xeon Gold 6430 processor.

C. Evaluation of Pothole Area Estimation and Optimization

1) *Pothole Area Estimation Evaluation Metrics*: For an ideal pothole area estimation algorithm, the estimates for the same pothole should be similar across different frames, demonstrating that the model can reliably predict potholes regardless of their position and size. Given the absence of ground truth measurements, we evaluate the estimation method's accuracy and consistency using statistical measures based on multiple observations of the same pothole. To assess this consistency, we introduce three evaluation metrics: Mean Absolute Error (MAE), Coefficient of Variation (CV), and adjacent frame differences (AFD). In all cases, lower values indicate better performance. Mean Absolute Error (MAE) measures the average deviation between each estimated area and the mean of all estimates for a single pothole.

$$\text{MAE} = \frac{1}{N} \sum_{k=1}^N |A_k - \bar{A}| \quad (21)$$

In the above, \bar{A} represents the mean area estimate for a specific pothole, calculated as follows.

$$\bar{A} = \frac{1}{N} \sum_{k=1}^N A_k \quad (22)$$

Next, to assess the relative dispersion and consistency of the area estimates, we introduce the Coefficient of Variation (CV) metric.

$$\text{CV} = \frac{\sqrt{\frac{1}{N} \sum_{k=1}^N (A_k - \bar{A})^2}}{\bar{A}} \quad (23)$$

Additionally, to quantify the variation between consecutive frame estimates and reflect the smoothness of the filter output, we incorporate a metric based on adjacent frame differences (AFD).

$$\text{AFD} = \frac{1}{N-1} \sum_{k=2}^N |A_k - A_{k-1}| \quad (24)$$

To further assess the reliability of our improved Kalman filtering method, we introduce the Normalized Innovation Squared (NIS) metric to evaluate the filter's internal consistency, reflecting how well the noise model matches the actual measurement data. The NIS is calculated using a specific formula as follows.

$$\text{NIS} = \nu_k^\top (\mathbf{P}_{k|k-1} + \mathbf{R}_k)^{-1} \nu_k \quad (25)$$

where ν_k represents the difference between the observed measurement and the predicted value at each time step. $\mathbf{P}_{k|k-1}$ represents the uncertainty level of the predicted value $\mathbf{A}_{k|k-1}$ and reflects the filter's uncertainty estimate during the prediction step, while \mathbf{R}_k indicates the filter's expected level of uncertainty in the measurement data. Theoretically, if the filter correctly models both the system and measurement noise, the NIS should statistically follow a chi-square distribution with an expected value related to the measurement dimension. In this study, since the area measurement is one-dimensional, the NIS should ideally be as close to 1 as possible.

2) *Kalman Filter Bayesian Parameter Optimization*: In optimizing the area estimation results using Kalman filtering, the calculation of noise covariance is crucial. In our study, the noise covariance consists of two components: confidence and pothole distance, as defined in Eq. 15. To determine the optimal weights lambda and theta for these two noise factors, we employ a Bayesian optimization algorithm to maximize overall filtering performance.

$$J(\lambda, \theta) = 10 \cdot \text{MAE} + \text{CV} + \text{AFD} + \text{NIS} \quad (26)$$

To comprehensively evaluate the filter's performance, we define a combined evaluation metric J , which integrates multiple indicators. The calculation methods for MAE, CV, AFD, and NIS are described in the previous section. Since MAE has a relatively small magnitude compared to the other metrics, we multiply it by a factor of 10 for better balance. These indicators depend on the filtering process, which in turn is influenced by $\mathbf{R}(\lambda, \theta)$, making J an implicit function of λ and θ . The objective is to find the optimal parameters that minimize J .

$$(\lambda^*, \theta^*) = \arg \min J(\lambda, \theta) \quad (27)$$

Specifically, we use the BayesianOptimization method from Python's bayes_opt library. The search range for both parameters is set between 0 and 2, with an initial exploration of five trials followed by 30 iterations to determine the optimal values.

D. Potholes Detection Results

To further evaluate the effectiveness of our pothole detection model, we conduct a series of comparative experiments under Clear Road Dataset and Dark Road Dataset, as shown in Table. I. To ensure objective comparison, this study selects multiple representative baseline models for evaluation. For the benchmark YOLOv8 series, we employ the baseline YOLOv8n [35], its lightweight variant YOLOv8n-ghost [36], and the YOLOv8s [35] model to conduct comprehensive comparisons. The models were assessed using precision, recall, F1-score, AP(50), AP(50-95), and GFLOPs.

The results show that on the Clear Road Dataset, the proposed ACSH-YOLOv8n model achieves the best performance in recall, F1-score, AP(50), and AP(50-95). It reaches an F1-score of 69.8% and an AP(50) of 76.6%, outperforming the second-best YOLOv5n by 2% in F1-score and YOLOv8s by 4.5% in AP(50). On the Dark Road Dataset, ACSH-YOLOv8n also achieves the highest scores in key metrics, including F1-score, AP(50), and AP(50-95). It records an AP(50) of 72.2%, surpassing the next best model, YOLOv8s, by 3.7%. Overall, while YOLOv3-tiny and YOLOv3-spp attain the highest precision on the two datasets respectively, their recall is significantly lower, resulting in weaker overall performance. The YOLOv8s model delivers competitive AP(50-95) scores comparable to ACSH-YOLOv8n across both datasets, along with strong performance in other metrics, reflecting the robustness of the YOLOv8 architecture. However, YOLOv8s has more than twice the number of parameters compared to ACSH-YOLOv8n. This further demonstrates that the proposed model not only achieves high detection performance but also

TABLE I
COMPARISON OF DIFFERENT MODELS RESULTS FOR POTHOLE DETECTION.

Dataset	Pothole Detection Model	Precision ↑	Recall ↑	F1-score ↑	AP (50) ↑	AP (50-95) ↑	GFLOPs ↓
Clear Road	YOLOv3-tiny [37]	72.5%	58.7%	64.9%	69.9%	26.7%	18.9
	YOLOv3-spp [38]	63.5%	60.6%	62.0%	64.9%	25.6%	12.0
	YOLOv5n [39]	71.0%	64.8%	67.8%	69.0%	26.9%	7.1
	YOLOv6n [40]	53.7%	68.1%	60.1%	62.1%	25.7%	11.9
	YOLOv8n [35]	55.0%	71.3%	62.1%	69.0%	28.5%	8.2
	YOLOv8n-ghost [36]	66.8%	64.8%	65.8%	65.5%	24.0%	5.0
	YOLOv8s [35]	71.2%	64.4%	67.6%	72.1%	28.6%	28.4
	ACSH-YOLOv8n	67.9%	71.8%	69.8%	76.6%	28.7%	13.4
Dark Road	YOLOv3-tiny [37]	69.8%	55.4%	61.8%	59.0%	20.3%	18.9
	YOLOv3-spp [38]	81.1%	57.1%	67.0%	67.3%	26.4%	12.0
	YOLOv5n [39]	65.8%	61.8%	63.7%	62.8%	24.6%	7.1
	YOLOv6n [40]	63.1%	44.6%	52.3%	48.1%	15.1%	11.9
	YOLOv8n [35]	71.9%	59.5%	65.1%	64.4%	25.8%	8.2
	YOLOv8n-ghost [36]	60.2%	60.7%	60.4%	66.3%	24.7%	5.0
	YOLOv8s [35]	53.6%	67.9%	59.9%	68.5%	27.0%	28.4
	ACSH-YOLOv8n	73.3%	62.5%	67.5%	72.2%	27.0%	13.4

Note: The best results values for each metric are highlighted in bold.

maintains a lightweight structure suitable for deployment. The performance gains are primarily attributed to architectural improvements rather than merely increasing model size.

The visual comparisons of detection performance on the Clear Road Dataset and Dark Road Dataset are shown in the first three rows of Fig. 8 and Fig. 9, respectively. In Fig. 8, the YOLOv5n model exhibits issues such as duplicate detections in the second image, where multiple bounding boxes are assigned to the same pothole, and missed detections in the third image. The YOLOv8n model produces a false positive in the first image by misidentifying a road step as a pothole. Additionally, in the second, third, and fourth images, it fails to detect several potholes, particularly small ones. In contrast, ACSH-YOLOv8n model demonstrates significantly improved performance, effectively detecting small potholes and achieving a higher recall rate. It also shows better alignment with the actual pothole contours, indicating strong potential for enhancing safety in autonomous driving scenarios. In Fig. 9, both YOLOv5n and YOLOv8n miss detections in the first three images. In the fourth image, YOLOv5n detects the same pothole three times with separate bounding boxes, while YOLOv8n continues to miss the pothole entirely. In comparison, ACSH-YOLOv8n model successfully addresses all these issues, demonstrating robust performance even in low-light conditions such as dusk or nighttime. This highlights the model's strong adaptability and potential for improving pothole detection reliability in challenging lighting environments, ultimately contributing to safer driving.

TABLE II
COMPARISON OF DIFFERENT AREA ESTIMATION ALGORITHMS AND CONSECUTIVE FRAME OPTIMIZATION STRATEGIES ON THE TWO DATASETS.

Dataset	MBTP	KF	Confidence	Distance	MAE ↓	CV ↓	AFD ↓	NIS ↓ 1
Clear Road	✓				0.168	0.379	0.143	/
	✓	✓	✓		0.147	0.262	0.123	/
	✓	✓		✓	0.054	0.199	0.035	1.404
	✓	✓	✓	✓	0.036	0.111	0.056	2.120
Dark Road	✓				0.038	0.119	0.020	1.530
	✓				0.085	0.539	0.072	/
	✓	✓	✓		0.054	0.454	0.048	/
	✓	✓		✓	0.033	0.318	0.026	1.278
	✓	✓	✓	✓	0.018	0.257	0.012	1.435
	✓	✓	✓	✓	0.015	0.230	0.009	1.303

Note: The best and near-best results values for each metric are highlighted in bold.

E. Area Estimation and Optimization Results

As the only available related approach in prior studies [41], the Corner Point (CP) method is used as a baseline to validate the effectiveness of our proposed MBTP method in area estimation. The evaluation is conducted using three key metrics: MAE, CV, and AFD, with the results presented in the upper section of Table. II. Our method consistently outperforms the corner point method across all three metrics, demonstrating its robustness and reliability.

The CP method estimates the pothole area by using the depths of two diagonal points of the bounding box, assuming

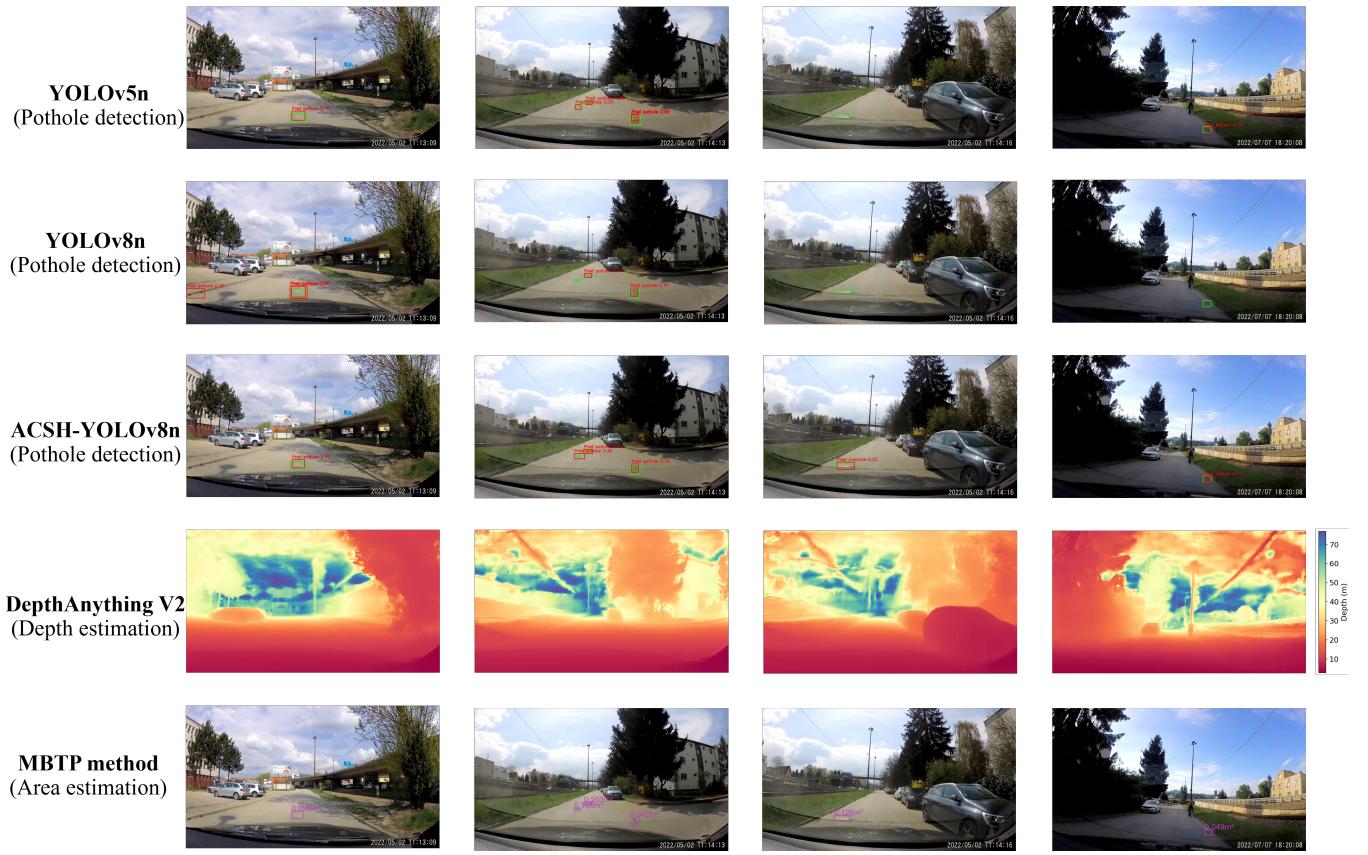


Fig. 8. Comparison of pothole area estimation workflows under Clear Road Dataset. The first, second and third rows show pothole detection results using YOLOv5n, YOLOv8n and the proposed ACSH-YOLOv8n models, respectively, where red boxes indicate predicted bounding boxes and green boxes represent ground truth. The fourth row displays the monocular metric depth estimation results generated by the DepthAnything V2 model. The fifth row presents the estimated pothole areas obtained by combining detection and depth information using the proposed MBTP method.

the pothole as a flat rectangular region. However, in real-world scenarios, object detection and depth estimation models introduce errors, particularly on damaged road surfaces, leading to significant inaccuracies. In contrast, our newly proposed MBTP method first maps the pothole to its minimum bounding rectangle, providing a more realistic approximation of its true shape. It then subdivides the pothole into multiple pixel-level triangular facets and incorporates depth information for estimation. This approach minimizes the impact of depth errors from individual pixels, resulting in a more accurate and robust estimation. The superior performance of our method highlights its effectiveness and reliability in pothole area estimation.

For the CDKF optimization of pothole estimation across consecutive frames, we compare different measurement noise covariance strategies: using only confidence, using only pothole distance, and combining both with weighted integration, as defined in Eq. 15. The weights, λ and θ , are optimized using Bayesian optimization. For the Clear Road Dataset, λ is set to 1.026 and θ to 0.7179. For the Dark Road Dataset, λ is set to 1.51 and θ to 1.227. The results of these three measurement noise covariance strategies are shown in the lower section of Table. II.

To validate the effectiveness of the proposed MBTP pothole area estimation algorithm and the consecutive frame optimization strategy CDKF, a series of comparative and ablation

experiments were conducted on both datasets, as shown in Table. II. The best and near-best results are highlighted in bold. The area estimation methods are evaluated using three metrics: MAE, CV, and AFD.

The first two rows of the Table. II present the results for the CP method and the proposed MBTP method, respectively. The MBTP method outperforms the CP method across all three metrics, demonstrating its robustness and reliability. The CP method estimates the pothole area by using the depths of two diagonal points within the bounding box and assumes the pothole is a flat rectangle. However, in real-world scenarios, due to inherent errors in both object detection and depth estimation models, particularly under challenging road conditions, this approach often results in significant inaccuracies. In contrast, the MBTP method first maps the pothole to its minimum bounding rectangle, which provides a more realistic representation of the pothole's shape. It then divides the region into multiple pixel-level triangular facets and integrates depth information to calculate the area. This helps reduce the impact of individual pixel-level depth errors and results in more accurate and robust estimates.

The following three rows present the results of applying Kalman filtering to MBTP-based estimates across consecutive video frames. An additional metric, NIS, is introduced to evaluate the consistency and noise modeling performance of

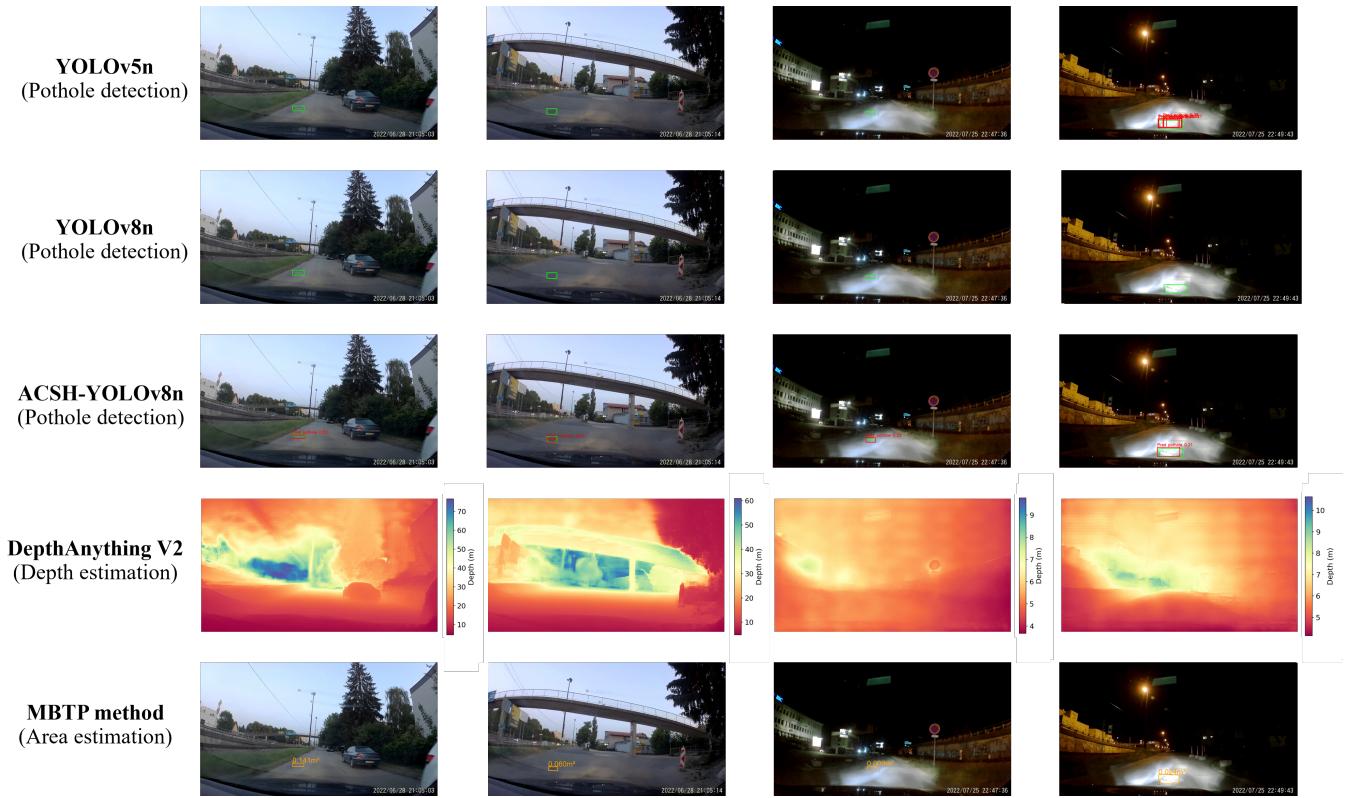


Fig. 9. Comparison of pothole area estimation workflows under Dark Road Dataset. The first, second and third rows show pothole detection results using YOLOv5n, YOLOv8n and the proposed ACSH-YOLOv8n models, respectively, where red boxes indicate predicted bounding boxes and green boxes represent ground truth. The fourth row displays the monocular metric depth estimation results generated by the DepthAnything V2 model. Due to varying depth visibility across images under low-light conditions, the measurement range differs. Therefore, a scale bar is provided on the right side of each depth map for reference. The fifth row presents the estimated pothole areas obtained by combining detection and depth information using the proposed MBTP method.

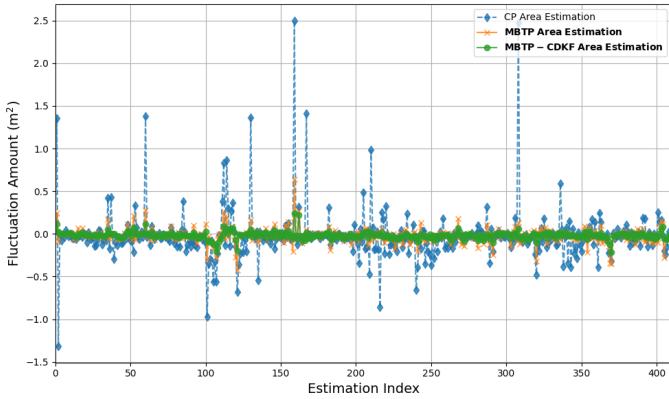


Fig. 10. Fluctuation comparison of pothole area estimation using different estimation methods and consecutive frame optimization strategies on the Clear Dataset.

the filter.

Across both datasets, applying confidence-based or distance-based noise covariance significantly improves MAE, CV, and AFD, indicating enhanced consistency and robustness. When using confidence alone, the NIS values of 1.404 and 1.278 are closest to the ideal value of 1, suggesting more accurate noise modeling. However, this method performs worse on MAE and CV, likely due to the high variability of confidence scores under different scenes. Confidence scores

are also inherently non-linear and non-smooth and tend to reflect classification certainty rather than spatial accuracy of the detected regions.

Using distance alone produces near-best MAE and CV values, indicating low overall fluctuations and a strong correlation between distance and the reliability of area estimation. However, this approach results in the worst AFD and NIS scores. The high AFD may be due to abrupt changes in the filter's reliance on measurements when a pothole moves from far to near, leading to inconsistency between consecutive frames. The poor NIS indicates that modeling noise solely with distance is incomplete, likely underestimating the true noise level and resulting in overly large innovations.

To address these shortcomings, the combined approach CDKF is proposed. It fuses both confidence and distance through a weighted sum. This combined approach achieves the best AFD scores of 0.02 and 0.009 for the two datasets. It also delivers the best or near-best performance across the other three metrics, achieving a balanced performance overall. For the AFD metric in particular, the use of both factors helps smooth out sudden changes by allowing one factor to compensate when the other varies sharply. This prevents large fluctuations in the Kalman gain and ensures smoother outputs across frames. The MAE and CV results are close to those of the distance-only method, while the NIS values are similar to the confidence-only method.

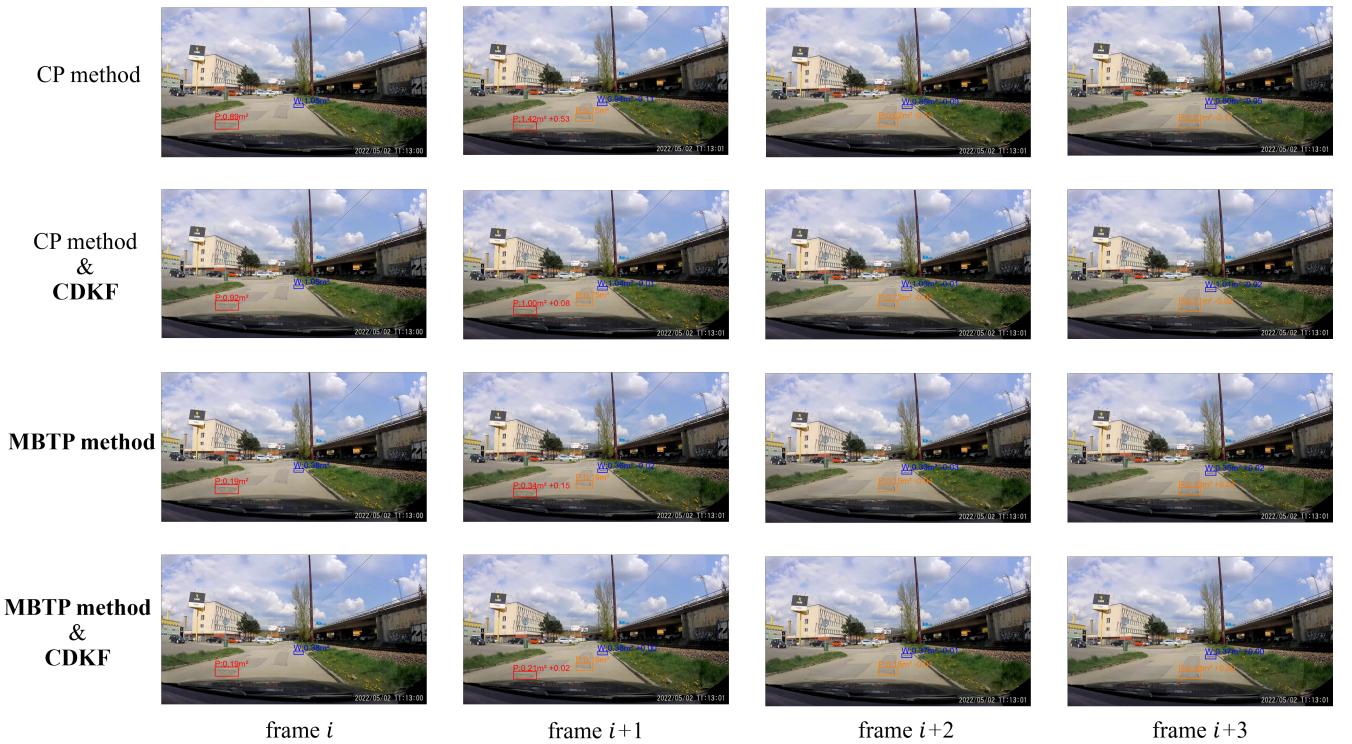


Fig. 11. Visualization of area estimation fluctuations for the same pothole across consecutive frames using the different area estimation method and optimization algorithm.

The line chart of area estimation fluctuations for the same pothole using different estimation methods and consecutive frame optimization strategies is shown in Fig. 10. The blue dashed line represents the CP method, which exhibits significant fluctuations and poor stability. The yellow line corresponds to the proposed MBTP method, which shows noticeably reduced variation. The green line represents the MBTP method with CDKF optimization, further minimizing fluctuations and demonstrating improved robustness.

The visualization of area estimation results across consecutive frames is shown in Fig. 11. Each column represents a single video frame, illustrating the area estimation results for four consecutive frames. The four rows correspond to the CP method and its CDKF-optimized version, as well as the MBTP method and its CDKF-optimized counterpart. Bounding boxes of the same color denote the same pothole or well with a consistent ID.

The CP method tends to produce overestimated results. As shown in the Fig. 11, medium-sized potholes are predicted to be approximately 1 m^2 , with significant fluctuations between adjacent frames. For instance, the first two frames exhibit a variation of 0.53 m^2 . This instability may stem from the method's reliance solely on corner features, making it highly sensitive to geometric modeling parameters. Although the CDKF-based optimization reduces prediction volatility, it fails to fully address the systematic overestimation issue.

In contrast, the proposed MBTP method yields more reasonable area estimates. The predicted pothole sizes in the figure average around 0.2 m^2 , aligning well with ground truth measurements. While minor fluctuations persist for the

same target, the MBTP method demonstrates significantly better stability than the CP approach. Further optimization with CDKF enhances robustness, delivering both precise and consistent predictions for the same pothole across frames. These findings demonstrate that combining both uncertainty measures leads to more stable and reliable area estimations.

F. Operation Time

The runtime of the proposed framework for each frame is measured, as summarized in Table. III. The framework consists of five main steps, each timed separately. Among these components, the detection module (ACSH-YOLOv8), the area estimation module (MBTP), and the consecutive frame optimization module (CDKF) are the methods proposed in this paper. All three exhibit low processing times, with ACSH-YOLOv8 taking 23.4 ms, MBTP taking 6.2 ms, and CDKF requiring less than 0.1 ms per frame. To accelerate the computationally intensive area estimation process, the Numba library is employed for just-in-time (JIT) compilation in Python, reducing the per-frame area estimation time from 135 ms to just 6.2 ms. The most time-consuming component is depth estimation, which requires 104 ms per frame. It is expected that with future advances in monocular depth estimation algorithms, further improvements in overall runtime can be achieved. Both serial and parallel processing schemes are evaluated. Since pothole detection and tracking are independent of depth estimation, they can be executed in parallel. Results show that parallelization reduces the overall processing time by 24.5 ms compared to the serial execution, reaching 110.2 ms for each frame.

TABLE III
OPERATION TIME OF THE PROPOSED FRAMEWORK AND ITS COMPONENTS.

Running Time/Frame		Detection	Tracking	Depth Estimation	Area Estimation		Optimization
Serial	Parallel	ACSH-YOLOv8	BoT-SORT	DepthAnything V2	MBTP	MBTP (JIT)	CDKF
134.7 ms	110.2 ms	23.4 ms	1.1 ms	104 ms	135 ms	6.2 ms	< 0.1 ms

V. CONCLUSION

In this paper, a robust pothole area estimation framework for video streams is proposed, which integrates object detection and monocular depth estimation. The estimation is further refined using CDKF for consecutive frame optimization. To address the challenges posed by small potholes and complex edge features, the ACSH-YOLOv8 detection network is proposed with a P2 detection head for small objects and integrating the ACmix attention mechanism into the Neck structure. Then the pre-trained monocular metric depth estimation model is utilized to generate pixel-wise depth maps. This paper proposes MBTP, a novel method for pothole area estimation. Using the pinhole camera model, potholes are mapped to 3D space and enclosed by a minimum bounding rectangle. The area is then calculated by tessellating the pothole into triangles and summing their areas. Finally, leveraging video stream data, the CDKF method is proposed, which optimally adjusts the estimation by incorporating confidence scores and distance information. Experiments show that our method significantly improves detection accuracy, especially for small potholes and complex edges. For area estimation, the MBTP method and CDKF yield more reliable and robust results.

The proposed fully vision-based pothole area estimation framework offers an efficient and reliable solution for enhancing the safety and comfort of autonomous driving. However, certain limitations remain. The method struggles with detecting highly blurred potholes, and modeling noise solely based on confidence and distance may not fully capture real-world variations. Additionally, the overall pipeline lacks dedicated runtime optimization to reduce latency. In future work, we plan to further refine the pothole detection network, incorporate factors such as ambient lighting and vehicle stability into noise modeling, and explore shared backbone architectures or parallel optimization techniques to improve area estimation speed, enhancing the framework's practical viability.

REFERENCES

- [1] Aidi Wang, Hong Lang, Zhen Chen, Yichuan Peng, Shuo Ding, and Jian John Lu. The two-step method of pavement pothole and raveling detection and segmentation based on deep learning. *IEEE Transactions on Intelligent Transportation Systems*, 25(6):5402–5417, 2024.
- [2] Nur Ridzuan Bukhari and Mohamad Yusri Aman. Review study of identify main factor that causes the cracking and potholes on asphalt pavement in malaysia. *Recent Trends in Civil Engineering and Built Environment*, 4(2):338–350, 2023.
- [3] J Muller and A Marnewick. Pothole-and patch repair failure recurrence in gauteng: The human influence. In *2016 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, pages 243–247. IEEE, 2016.
- [4] BBC News. Antarctica warming up: Scientists uncover climate secrets, November 2023. Accessed: 2024-11-14.
- [5] Rick Snyder. Potholes & more potholes: is it just us?, 2018. Accessed: 2024-11-20.
- [6] Ufuk Kirbaş. Effects of pothole type pavement distress on whole-body vibration. *Road Materials and Pavement Design*, 24(6):1403–1424, 2023.
- [7] Nachuan Ma, Jiahe Fan, Wenshuo Wang, Jin Wu, Yu Jiang, Lihua Xie, and Rui Fan. Computer vision for road imaging and pothole detection: a state-of-the-art review of systems and algorithms. *Transportation safety and Environment*, 4(4):tdac026, 2022.
- [8] Amita Dhiman and Reinhard Klette. Pothole detection using computer vision and learning. *IEEE Transactions on Intelligent Transportation Systems*, 21(8):3536–3550, 2019.
- [9] Hangbin Wu, Lianbi Yao, Zeran Xu, Yayun Li, Xinran Ao, Qichao Chen, Zhengning Li, and Bin Meng. Road pothole extraction and safety evaluation by integration of point cloud and images derived from mobile mapping sensors. *Advanced Engineering Informatics*, 42:100936, 2019.
- [10] Giuseppe Loprencipe and Pablo Zoccali. Ride quality due to road surface irregularities: Comparison of different methods applied on a set of real road profiles. *Coatings*, 7(5):59, 2017.
- [11] Hamideh Taslimasa, Sajjad Dadkhah, Euclides Carlos Pinto Neto, Pulei Xiong, Suprio Ray, and Ali A Ghorbani. Security issues in internet of vehicles (iov): A comprehensive survey. *Internet of Things*, 22:100809, 2023.
- [12] Sayim Niyaz Baba and Er Brahmjeet Singh. Identification of problems faced in road maintenance. *International Journal of Innovative Research in Engineering & Management*, 10(3):29–37, 2023.
- [13] Ta Yang Goh, Shafriza Nisha Basah, Haniza Yazid, Muhammad Juhairi Aziz Safar, and Fathinul Syahir Ahmad Saad. Performance analysis of image thresholding: Otsu technique. *Measurement*, 114:298–307, 2018.
- [14] Emir Buza, Samir Omanovic, and Alvin Huseinovic. Pothole detection with image processing and spectral clustering. In *Proceedings of the 2nd International Conference on Information Technology and Computer Networks*, volume 810, page 4853, 2013.
- [15] Yashon O Ouma and Michael Hahn. Pothole detection on asphalt pavements from 2d-colour pothole images using fuzzy c-means clustering and morphological reconstruction. *Automation in Construction*, 83:196–211, 2017.
- [16] Rigen Wu, Jiahe Fan, Libo Guo, Lei Qiao, M Usman Maqbool Bhutta, Brett Hosking, Sergey Vityazev, and Rui Fan. Scale-adaptive pothole detection and tracking from 3-d road point clouds. In *2021 IEEE International Conference on Imaging Systems and Techniques (IST)*, pages 1–5. IEEE, 2021.
- [17] Linchao Li, Jiazen Liu, Jiabao Xing, Zhiyang Liu, Kai Lin, and Bowen Du. Road pothole detection based on crowdsourced data and extended mask r-cnn. *IEEE Transactions on Intelligent Transportation Systems*, 25(9):12504–12516, 2024.
- [18] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.
- [19] Ernini Niswatal Ukhwah, Eko Mulyanto Yuniarso, and Yoyon Kusnendar Suprapto. Asphalt pavement pothole detection using deep learning method based on yolo neural network. In *2019 International Seminar on Intelligent Technology and Its Applications (ISITA)*, pages 35–40. IEEE, 2019.
- [20] Anas Al Shaghouri, Rami Alkhatib, and Samir Berjaoui. Real-time pothole detection using deep learning. *arXiv preprint arXiv:2107.06356*, 2021.
- [21] TC Mahalingesh, Harshit Mishra, RV Arun, Anshuman Anand, et al. Pothole detection and filling system using image processing and machine learning. In *2024 International Conference on Smart Systems for applications in Electrical Sciences (ICSSES)*, pages 1–5. IEEE, 2024.
- [22] Radhika Ravi, Ayman Habib, and Darcy Bullock. Pothole mapping and patching quantity estimates using lidar-based mobile mapping systems. *Transportation Research Record*, 2674(9):124–134, 2020.
- [23] Siyuan Chen, Debra F Laefer, Xiangding Zeng, Linh Truong-Hong, and

- Eleni Mangina. Volumetric pothole detection from uav-based imagery. *Journal of Surveying Engineering*, 150(2):05024001, 2024.
- [24] Dong-Hoe Heo, Ji-Yoon Choi, Sang-Baeg Kim, Tae-Oh Tak, and Sheng-Peng Zhang. Image-based pothole detection using multi-scale feature network and risk assessment. *Electronics*, 12(4):826, 2023.
- [25] Subash Kharel and Khaled R Ahmed. Potholes detection using deep learning and area estimation using image processing. In *Proceedings of SAI Intelligent Systems Conference*, pages 373–388. Springer, 2021.
- [26] Pranjal A Chitale, Kaustubh Y Kekre, Hrishikesh R Shenai, Ruhina Karani, and Jay P Gala. Pothole detection and dimension estimation system using deep learning (yolo) and image processing. In *2020 35th International Conference on Image and Vision Computing New Zealand (IVCNZ)*, pages 1–6. IEEE, 2020.
- [27] Xuran Pan, Chunjiang Ge, Rui Lu, Shiji Song, Guanfu Chen, Zeyi Huang, and Gao Huang. On the integration of self-attention and convolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 815–825, 2022.
- [28] Nir Aharon, Roy Orfaig, and Ben-Zion Bobrovsky. Bot-sort: Robust associations multi-pedestrian tracking. *arXiv preprint arXiv:2206.14651*, 2022.
- [29] Jean-Yves Bouguet et al. Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm. *Intel corporation*, 5(1-10):4, 2001.
- [30] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.
- [31] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.
- [32] Sk Abu Talha, Dmitry Manasreh, and Munir D Nazzal. The use of lidar and artificial intelligence algorithms for detection and size estimation of potholes. *Buildings*, 14(4):1078, 2024.
- [33] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *arXiv preprint arXiv:2406.09414*, 2024.
- [34] Boris Bučko, Eva Lieskovská, Katarína Zábovská, and Michal Zábovský. Computer vision based pothole detection under challenging conditions. *Sensors*, 22(22):8878, 2022.
- [35] Shruti Kumari, Anjali Gautam, Suvramalya Basak, and Nidhi Saxena. Yolov8 based deep learning method for potholes detection. In *2023 IEEE International Conference on Computer Vision and Machine Intelligence (CVMI)*, pages 1–6. IEEE, 2023.
- [36] Mohammed Hussein and Wen-Xing Zhu. A real-time ghost machine learning model built on yolov8 for traffic road signs detection and classification in germany. *Multimedia Systems*, 30(6):344, 2024.
- [37] Pranav Adarsh, Pratibha Rathi, and Manoj Kumar. Yolo v3-tiny: Object detection and recognition using one stage improved model. In *2020 6th international conference on advanced computing and communication systems (ICACCS)*, pages 687–694. IEEE, 2020.
- [38] Tianxin Liu, Jiaxuan Li, Meiyi Cai, Yuyong Cui, and Quan-Yong Fan. An improved yolov3-spp algorithm for image-based pothole detection. In *International Symposium on Neural Networks*, pages 328–335. Springer, 2024.
- [39] Sudhakar Ajmera, C Ashok Kumar, P Yakaiah, Bittu Kumar, and K Yashwanth Chowdary. Real-time pothole detection using yolov5. In *2022 International Conference on Advancements in Smart, Secure and Intelligent Computing (ASSIC)*, pages 1–5. IEEE, 2022.
- [40] Chui Li, Lulu Li, Hongliang Jiang, Kaiheng Weng, Yifei Geng, Liang Li, Zaidan Ke, Qingyuan Li, Meng Cheng, Weiqiang Nie, et al. Yolov6: A single-stage object detection framework for industrial applications. *arXiv preprint arXiv:2209.02976*, 2022.
- [41] Dehao Wang, Yiwen Xu, Haohang Zhu, and Kaiqi Liu. A novel framework for pothole area estimation based on object detection and monocular metric depth estimation. In *2024 IEEE International Conference on Signal, Information and Data Processing (ICSIDP)*, pages 1–6. IEEE, 2024.