IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received June 29, 2020, accepted July 16, 2020, date of publication July 21, 2020, date of current version July 30, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3010885

Detecting and Tracking Sinkholes Using Multi-Level Convolutional Neural Networks and Data Association

HOAI NAM VU[1], CUONG PHAM[1], NGUYEN MANH DUNG[2], AND SOONGHWAN RO[3]

[1]Department of Computer Science, Posts and Telecommunications Institute of Technology, Hanoi 12110, Vietnam
[2]Department of Electronic Engineering, Posts and Telecommunications Institute of Technology, Hanoi 12110, Vietnam
[3]Department of Information and Communication, Kongju National University, Cheonan 314701, South Korea

Corresponding author: Soonghwan Ro (rosh@kongju.ac.kr)

This work was supported in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education under Grant NRF2018R1D1A1B07049146, and in part by the Domestic Scholarship Program of the Vingroup Innovation Foundation under Grant VINIF.2019.TS.41.

ABSTRACT Sinkholes can cause severe property damage and threaten public safety. Therefore, the early prediction and detection of sinkholes are important measures for protecting both citizenry and infrastructure. Although many studies have made significant progress on sinkhole detection, challenges remain, including long-term data collection and the discovery of lightweight machine learning models that can be deployed to analyze sinkhole images. In this paper, we propose a method that takes advantage of the recent success of deep learning models to detect and track sinkholes via video streaming. Our system consists of two main stages: sinkhole detection with a cascaded convolutional neural network and sinkhole tracking with a data association algorithm. The experimental results show that a sinkhole can be tracked in real time using the dataset [1]. Furthermore, we implement the system on a Jetson TX2 embedded board (weighing 85 grams), which can operate at 13.2 FPS (frames per second). With an average IoU (intersection over union) score of 88% for sinkhole tracking and an accuracy of 97,6% for sinkhole detection on a 45-minute dataset, this study demonstrates the feasibility of sinkhole detection and tracking using IR images and their suitability for practical applications.

INDEX TERMS Sinkhole detection, convolutional neural network, imagenet, embedded system, data association.

I. INTRODUCTION

A sinkhole can be characterized as a depression or hole in the ground caused by some form of surface layer degradation. The majority of sinkholes are the consequence of karst processes [2]. With the development of underground revelation procedures, the four main sources of sinkholes have been shown to be disintegration, cover-subsidence, cover-collapse, and anthropogenic impacts. The occurrence of human-induced sinkholes has been rapidly increasing because of large amounts of construction in urban regions. As recently reported in Vietnam, an increasing number of human-induced sinkholes have been appearing in both Ha Noi and Ho Chi Minh cities, two of the largest cities in Vietnam, causing a great loss of lives and wealth. In some cases, sinkholes have been used as disposal sites for various forms of waste, which has led to serious groundwater pollution and negative effects on the health and quality of life of local people. Therefore, the early warning and detection of sinkholes are emerging issues that must be addressed by governments and social organizations to improve the quality of human life where human-induced sinkholes occur frequently.

Although sinkholes can affect the quality of human life and economic development, only a few studies have been conducted on sinkhole prediction and management. In the literature, sinkhole identification can be categorized into two main groups: traditional non-camera-based methods and camera-based methods. In the first group, these methods for detecting sinkholes have utilized various kinds of specialized equipment, such as cone penetrometer testing (CPT), ground penetrating radar (GPR), electrical resistivity tomography (ERT), and airborne lasers (LiDAR). These methods

The associate editor coordinating the review of this manuscript and approving it for publication was Yonghong Peng.

can achieve high accuracy by using expensive specialized equipment. However, the drawback of these methods is that they are not able to monitor a large area, which is a mandatory requirement in a wide range of applications. The second group takes advantage of camera systems to monitor large areas and advanced image processing techniques to detect sinkholes. The images captured from the cameras are processed to obtain the results automatically without any human operations. With the advancements in camera hardware technology, building these surveillance systems has become cheaper and more practical. In fact, sinkholes often occur outside areas monitored by conventional camera systems. There is a highly effective alternative solution that uses a camera on an unmanned aerial vehicle (UAV) device to perform surveillance. The use of UAVs has several benefits, including a flexible monitoring area, automatic operation, and lower cost than fixed camera networks. Furthermore, an image processing algorithm can be implemented in an embedded board mounted on a UAV to obtain immediate results to control the operation of the UAV. The sinkhole positions and their respective images can be transferred to a ground station and drawn on an online map. However, traditional RGB cameras can experience difficulties in bad weather and lightning conditions. A thermal camera can be an additional source of information to traditional cameras. The benefit of using thermal cameras is that they can work in various kinds of weather and lightning conditions. In particular, thermal cameras can be useful when there is a high temperature difference at the monitored object.

This paper proposes a method to process infrared images from thermal cameras using a multilevel convolutional neural network (CNN). The proposed method consists of two CNN models and a rule-based filter in the middle. When a sinkhole position is detected, a data association algorithm is implemented to track the sinkhole object in real time. The remainder of this paper is organized as follows. Section 2 reviews the related works. Section 3 describes the proposed system. Section 4 reports on experimental results. Future research directions and a discussion are provided in Section 5.

II. RELATED WORK

A. NON-VISION BASED- METHODS

The most conventional approach for identifying a sinkhole is to use blind drilling in areas where sinkholes may appear. However, this naive method is usually ineffective; in addition, there are risks of negatively impacting the environment and exacerbating sinkhole development [3]. Another method for identifying sinkholes uses cone penetrometer testing (CPT), which utilizes the geotechnical engineering properties of soils and delineates soil stratigraphy to predict a sinkhole. In practice, CPT is relatively quick and simple to implement; however, it could make the sinkhole even worse. Ground penetrating radar (GPR) is currently the most common method used to investigate potential sinkhole activity. GPR sends a radar signal into the ground and analyzes the received signal. The drawback of this method is that the signal can be scattered because of the extremely heterogeneous conditions of the underground layers. Similar to the GPR-based approach, seismic waves can also be used to analyze the probability of a sinkhole appearing in a specific area, as reported in [4]. Electrical resistivity tomography (ERT) and dynamic penetration super-heavy (DPSH) measurements are also used to determine the location of sinkholes in local municipality areas [4].

Airborne laser scanning technology [5] is primarily perceived as a way to collect detailed 3D information about a surface and the objects located on it. The 3D data contain information about surface features and sinkholes, which are usually embedded in the terrain and form distinct shape transitions from their surroundings. The author of [6] developed a LiDAR-based detection system to detect sinkholes. The model first creates a digital elevation model (DEM), fills the depressions in the DEM, extracts the depressions with DEM differences, and converts the depressions to a polygon shape file. The system in [7] utilized airborne LiDAR data in combination with context information to improve the accuracy of sinkhole detection. Reference [8] proposed a conceptual framework for detecting sinkholes by airborne LiDAR that consists of three steps: data prepossessing, preliminary sinkhole map development, and final sinkhole map development. Airborne LiDAR can provide a solution to identify subsidence areas by using ground object temperature, which cannot be determined using traditional photography. However, to collect the data, flights over several locations are required, which is time consuming and expensive. In addition, the quality of the atmosphere also affects the results of this method.

B. VISION BASED- METHODS

In recent years, deep learning [9] has led to breakthroughs in a wide range of applications. There have been many kinds of deep learning models, such as convolutional neural networks (CNNs), deep belief networks (DBNs), and deep recurrent networks (DRNs). The author of [10] reviewed these models applied to computer vision problems. CNN-based models show the best performance because of the suitability of an image as input data. CNN architectures are designed to account for the 2-dimensional structure of an input image, which may be the main reason for the state-of-the-art performance in this domain. Furthermore, CNN architectures have a parameter sharing strategy and pooling layer to reduce the training time. In contrast, a DBN architecture usually experiences high computational cost associated with the training process, and the further optimization of the network based on maximum likelihood is unclear. More importantly, the DBN architecture is not designed for a 2-dimensional structure, which may lead to decreased network performance when training with images. DRN architectures has been proposed to solve different kinds of problems, such as language modeling, machine translation, and time-series data. In the literature, CNN-based architectures have been applied successfully in a wide range of computer vision applications, including gold immune chromatographic strip (GICS) image segmentation in the medical domain [11], [12], salient object detection [13], [14], facial expression recognition [15]–[18], image retrieval [19], [20] and video surveillance such as smoke [21], [22] and fire [23] detection, anomaly detection [24], [25] and activity recognition [26]. Among these video surveillance system applications, anomaly detection methods have attracted attention from the research community because of their importance in high-security monitoring systems. However, anomaly detection and recognition using modern deep learning have encountered difficulties in lacking training data in anomalous situations. The authors of [24] proposed an incremental spatiotemporal learner (ISTL) to address the challenges and limitations of traditional anomaly detectors. Their ISTL is an unsupervised deep learning model that utilizes active learning with fuzzy aggregation. This approach can help their model evolve when additional data are fed to the model over time. In [25], the authors take advantage of deep model transfer learning to train their model to recognize anomalous situations with less training data.

Remote sensing images captured by satellites or UAVs precisely meet the demands for gathering an overview of ground object locations and situations; on the other hand, because of overhead camera angles, the problems of object size and object occlusion can be avoided. Furthermore, UAVs can travel to monitor regions that are difficult for humans to reach. This increases the efficiency of drone use in practical applications. In the current literature, it is possible to find many studies focusing on the applications of UAVs for monitoring ground-based objects. To track and detect obstacles for UAV navigation, the authors of [27] proposed a method of real-time object localization and tracking from monocular image sequences. Their method consists of two main stages: a detection stage with a saliency map computed via background connectivity and the tracking stage with a Kalman filter. The authors of [28] proposed a method to detect and count ground vehicles in high-resolution aerial images. Their method utilizes a convolutional neural network to regress a vehicle spatial density map across aerial images. UAVs have also been used to detect multiple objects. The authors of [29] proposed an approach with a CNN and the Hungarian algorithm (HA) to track multiple humans from drone images. They implement a faster RCNN for object localization at the first stage, after which they solve the data association problem in visual tracking by the HA.

Another important use of drone images is monitoring land use in smart agriculture applications. The authors of [30] proposed a system to predict the harvest yield from low-altitude UAV images. They implement an image processing algorithm that combines K-means clustering with a graph cut to segment rice grain areas. The graph cut algorithm was applied to extract the foreground and background of the images. Then, the foreground images were converted to the LAB color space, and K-means clustering was used to label the pixels based on color information. After that, the area of the rice grains in the images was calculated from the clustered images. UAV-based hyperspectral image processing was proposed to monitor land use in agricultural applications in [31]. The proposed solution is based on a commercial DJI Matrice 600 drone and a Specim FX10 hyperspectral camera. Their system consists of an embedded board with advanced processing capabilities that is mounted on the drone to control its trajectory, manage the data acquisition, and allow on-board processing, such as the evaluation of different vegetation indices.

Among the abovementioned surveillance systems, UAVs are equipped with a traditional imaging sensor platform that can output an RGB image. This kind of image can suffer from bad weather and illumination changes between day and night. However, thermal images have a valuable advantage over traditional RGB images. Thermal images do not depend on illumination, and the output is the projection of thermal sensors of the heat emissions of the objects. This unique trait gives rise to effective object segmentation applications. Ultimately, surveillance measurements using UAVs improve significantly with thermal cameras. Reference [1] proposed a system to take advantage of a thermal camera to detect a sinkhole position from a UAV. Their system includes two main parts. The first part detects the candidate regions by analyzing the cold spots in the thermal images based on the fact that a sinkhole usually has a lower temperature than the surrounding area. The second part classifies the candidate region as a real sinkhole by applying a light CNN and boosted random forest with handcrafted features. The authors successfully applied the proposed ensemble method to sinkhole data of various sizes and depths under different environmental conditions. In our previous work [32], a deep learning transfer approach was proposed to solve the sinkhole detection problem. We used a simple thresholding algorithm to segment an input image and Resnet transfer learning [33] to classify the segmented sinkhole candidate. Although that work did achieve promising results in detecting sinkholes from infrared images, the results were not generalized due to the fixed threshold value of the proposed algorithm. In addition, the processing speed did not fulfill the real-time requirements due to the complexity of the Resnet architecture. In addition, a solution to the problem of UAV surveillance systems is to use successful deep learning-based detectors such as two-stage detectors [34] and single-stage detectors [35]–[38]. However, the accuracy of a single-stage detector cannot be guaranteed in detecting small object sizes, while the processing speed of a two-stage detector cannot achieve the real-time requirement of the UAV context. Therefore, it is difficult to apply these detectors directly to the problem of UAV surveillance, especially the sinkhole detection problem that this paper is trying to solve.

III. PROPOSED METHOD

Our proposed method consists of two main stages: sinkhole detection with a multilevel CNN model and sinkhole tracking using data association, as shown in Fig. 1. Our system can

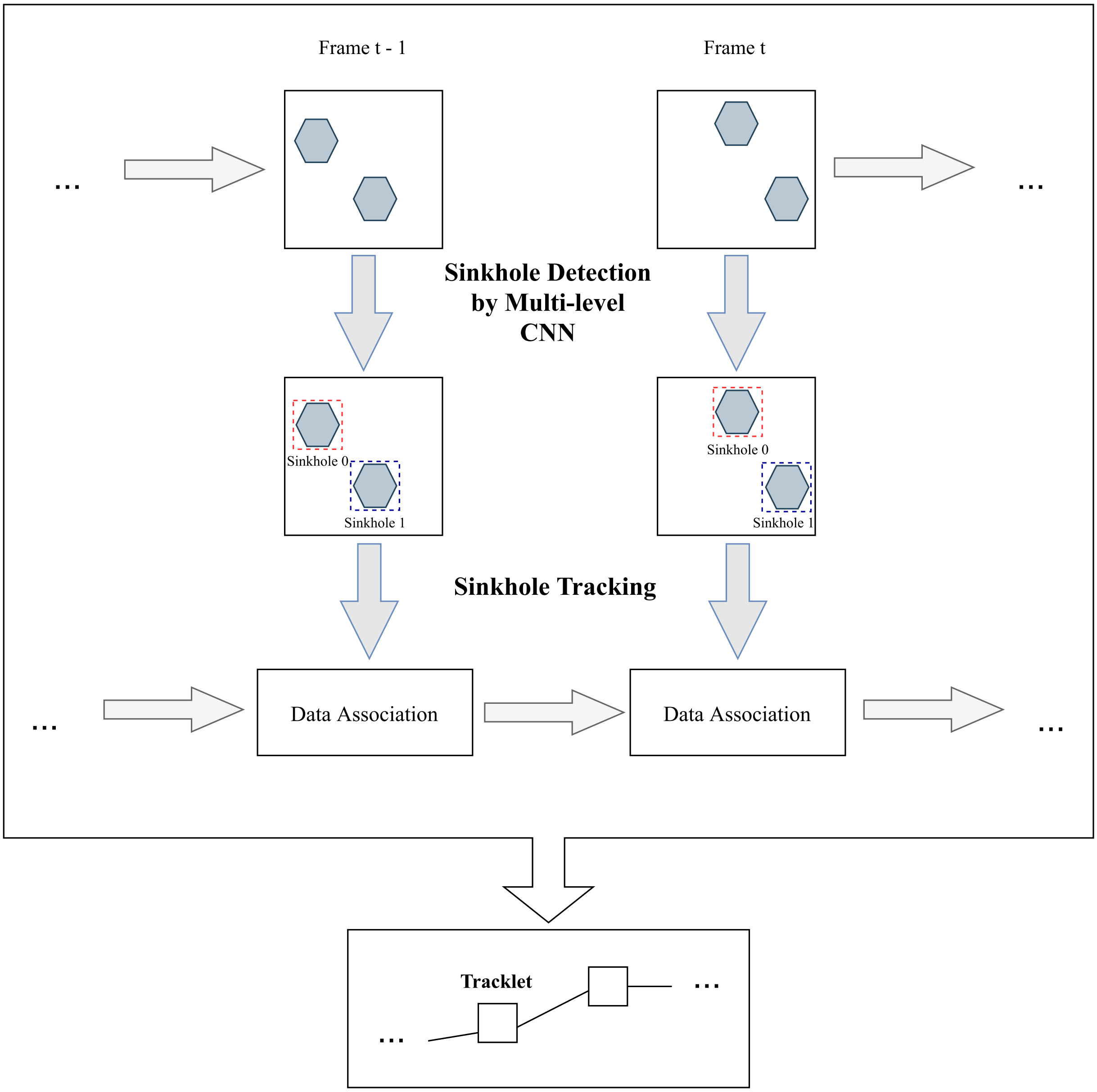


FIGURE 1. Flowchart of proposed method.

track multiple sinkholes in a single frame. For example, in Fig. 1, two sinkholes occur in the $(t-1)^{th}$ frame. The first stage of our proposed method is to detect all sinkholes appearing in the current frame, which results in a set of the bounding boxes for the association the sinkholes such as Sinkhole 0 and Sinkhole 1. We have a set of bounding boxes for every frame. The locations of all bounding boxes are the inputs of the sinkhole tracking algorithm by data association [40]. The main purpose of data association is to assign the bounding boxes of the current frame (i.e., the t^{th} frame) to their exact trajectory from the previous $(t-1)^{th}$ frames. The tracklet means the trajectory of each sinkhole, which is a chronological sequence consisting of multidimensional locations of the sinkhole's center point. The final output of the proposed system is tracklets associated with real-sinkhole locations in the current frame. In the experiment, the occurrence of the sinkhole appears to be flickering due to the drone camera's moving speed and the noisy input image. We have adopted a direction voting technique to address this issue while tracking for more stable results.

A. SINKHOLE DETECTION BY A MULTILEVEL CNN MODEL

In our multilevel CNN model, as shown in Fig. 2, there are two CNN models stacked together and a rule-based filter in the middle. The purpose of this model is twofold. The first CNN model is a UNET based semantic segmentation architecture [41], which is used as a weak but high-speed classifier to eliminate easy-to-remove noise. The second CNN model is the MobileNet v3 architecture, which serves as a strong classifier to distinguish between real sinkholes and sinkhole candidates. The outputs of the first CNN model and the rule-based filter are the sinkhole area and the associated bounding box (a rectangle). Thereafter, we crop the original image with the position of the bounding box to obtain original grayscale sinkhole images, which are used as the training data for the second CNN model. The second CNN model is used to distinguish the sinkhole objects from the non-sinkhole objects that the first CNN model could not identify due to the lack of information in the binarization image. An example of training data for the second CNN model is shown in Fig. 5, where the first row shows the real sinkhole objects, and the second row shows the non-sinkhole objects. The motivation for adopting multilevel CNN is to increase efficiency. In the literature, the single end-to-end CNN object detection models still include two main parts: region proposal and region classification. If the number of proposed regions is too high, the model's total speed will be slow. Therefore, we use a CNN model to segment sinkhole candidates based on contextual information from infrared images. The CNN segmentation model significantly reduces the number of regions that need to be classified in the following CNN model, which accounts for classification jobs.

1) CNN LEVEL 1 BY MODIFIED UNET ARCHITECTURE

The UNET semantic segmentation model transforms the input image into the binary images with white pixels for the foreground and black pixels for the background. The UNET model was first applied successfully to segment medical images and then widely applied to other problems such as aerial image segmentation, vehicle segmentation, and agriculture image segmentation. The UNET model can learn hierarchical features and has a particularly competitive processing speed. We build the model with fewer layers and a smaller input image size than the original UNET model. Compared to the medical image segmentation problem in the original paper, our segmentation dataset looks much less detailed; therefore, a smaller model with a smaller input image size is sufficient to obtain the expected binary image. As mentioned earlier, this CNN model is used as a weak but fast classifier; thus, the goal of this model is to segment all objects with high sinkhole probability that may include a noisy object. The noisy object is addressed with the rule-based filter and the second CNN model. Fig. 1 shows a flowchart of our proposed method. The benefit of using CNN-based binarization segmentation rather than traditional binarization segmentation algorithms used in [1] is that the accuracy does not depend on a hard threshold, and the accuracy improves greatly if more annotated data are fitted to the model.

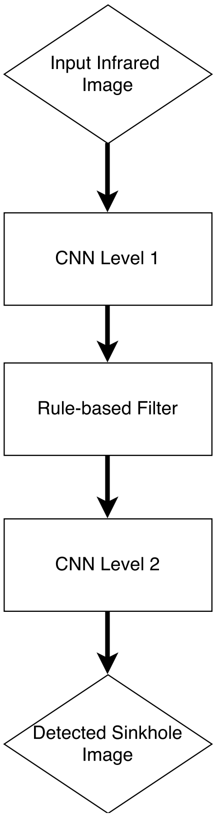


FIGURE 2. Sinkhole detection pipeline.

The CNN model is shown in Fig. 4, where the input image size is 256×336. The model architecture consists of three sections: the contraction, bottleneck, and expansion sections. The contraction section comprises three contraction blocks. Each block receives an input that applies two 3×3 convolution layers followed by a 2×2 max pooling. The number of feature maps after each block doubles so that the architecture can learn the complex structures effectively. The bridge layer mediates between the contraction layer and the expansion layer. It uses two 3×3 CNN layers followed by a 2×2 up-convolution layer. Similar to the contraction layer, the expansion layer also consists of three expansion blocks. Each block passes the input to two 3×3 CNN layers followed by a 2×2 up-convolution layer. After each block, the number of feature maps used by the convolutional layer is halved to maintain symmetry.

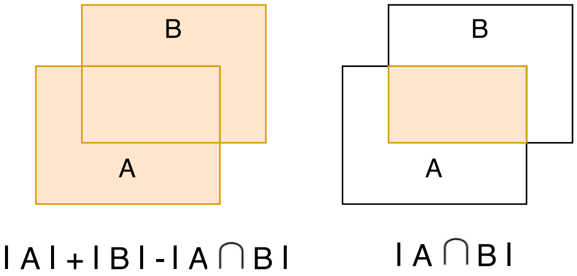


FIGURE 3. Set operation illustration.

The Jaccard loss is chosen to train the model. The Jaccard loss is often referred to as the intersection-over-union score, which is a measure score commonly used in segmentation models. Let A represent the ground truth image of segmentation, in which the sinkhole regions are manually identified, and let B represent a system-generated image. The Jaccard score is defined as:

$$J(A,B) = \frac{|A \cap B|}{|A| + |B| - |A \cap B|} \tag{1}$$

Equation 1 is constructed based on set theory, where $\cap$ is the intersection operator as shown in Fig. 3. The Jaccard index approaches 1 when the intersection area approaches the maximum value, while the Jaccard index approaches 0 when the intersection area approaches 0. This behavior of the Jaccard index is suitable for a loss function which is presented in equation 3. This Jaccard score is then taken as an average over the entire set of pixels producing a value between 0 and 1. These set operations are, however, not differentiable. To apply these set operations in their true form, the pixel values in images A and B need to be absolute 1s and 0s, respectively. However, while the ground truth image (A) contains these absolute values, the system-generated image (B) contains a float value between 0 and 1 due to the activation function in the final upsampling layer of the network. Therefore, an approximation of the Jaccard score by the probabilities can be used. Then, the intersection operator ($\cap$) is replaced by the elementwise multiplication (*) of images A and B.

$$J(A,B) = \frac{A * B}{A + B - (A * B)} \tag{2}$$

After reducing the Jaccard score from the set operations to arithmetic operations, the formula is differentiable. As a loss function, the error needs to approach 0 when the result improves as mentioned earlier. To achieve this result, the loss function is defined in terms of the Jaccard score as follows:

$$L_J = 1 - J(A,B) = 1 - \frac{A * B}{A + B - (A * B)} \tag{3}$$

Let $t_{i,j} \in \{0, 1\}$ be the actual class of pixels at $\{i, j\}$ with $t_{i,j} = 1$ for sinkholes and $t_{i,j} = 0$ for background. In addition, $p_{i,j}$ represents an estimated posterior probability that the pixel at $\{i, j\}$ belongs to the sinkhole object. Therefore, $t_{i,j}$ accounts for ground truth pixels, while $p_{i,j}$ accounts for the predicted result of the pixels. The elementwise multiplication between A and B can become the sum of $t_{i,j}$ and $p_{i,j}$ of each pixel. We can rewrite the loss function as follows:

$$L_J = 1 - \frac{\sum\limits_{i,j}(t_{i,j}p_{i,j})}{\sum\limits_{i,j} t_{i,j}^2 + \sum\limits_{i,j} p_{i,j}^2 - \sum\limits_{i,j}(t_{i,j}p_{i,j})} \tag{4}$$

From this point, the loss function is differentiable. The derivative of the loss function can be represented by the following formula:

$$\frac{\partial L_J}{\partial p_{i,j}} = -\frac{t_{i,j}[\sum\limits_{i,j} t_{i,j}^2 + \sum\limits_{i,j} p_{i,j}^2 - \sum\limits_{i,j}(t_{i,j}p_{i,j})]}{[\sum\limits_{i,j} t_{i,j}^2 + \sum\limits_{i,j} p_{i,j}^2 - \sum\limits_{i,j}(t_{i,j}p_{i,j})]^2} + \frac{(2p_{i,j} - t_{i,j})[\sum\limits_{i,j}(t_{i,j}p_{i,j})]}{[\sum\limits_{i,j} t_{i,j}^2 + \sum\limits_{i,j} p_{i,j}^2 - \sum\limits_{i,j}(t_{i,j}p_{i,j})]^2} \tag{5}$$

These derivatives can be efficiently integrated into backpropagation during the network training procedure.

Algorithm 1 Rule-Based Filter

```
Connected component initialization
while $CC_i \in Connected\ Component\ Set$ do
    if $w_i$ or $h_i \leq t_1$ then
        remove $CC_i$
    end
    if $w_i$ or $h_i \geq t_2$ then
        remove $CC_i$
    end
    if $\frac{w_i}{h_i} \leq t_3$ then
        remove $CC_i$
    end
    if $\frac{w_i}{h_i} \geq t_4$ then
        remove $CC_i$
    end
    if shape of $CC_i$ is not convex hull then
        remove $CC_i$
    end
    if $s_i \leq t_5$ then
        remove $CC_i$
    end
    if $s_i \geq t_6$ then
        remove $CC_i$
    end
    if the variance of pixel value in $i^{th}$ connected component
    is larger than $t_7$ then
        remove $CC_i$
    end
end
```

2) RULE-BASED FILTER

As mentioned above, our proposed system is built based on a cascade scheme, which stacks weak but fast filters during the early stages. Therefore, the output of the UNET-like semantic segmentation model also contains a large amount of noise due to oversegmentation, where sinkhole-like objects are also segmented to cover all real sinkhole objects in the image. The rule-based filter is used as a post-processing step for the model [43]. Without the rule-based filter, all connected components from UNET-like models will be the input of the second CNN model for classification, which causes abundant computation. The rule-based filter is a lightweight filter that effectively eliminates unwanted connected component noises. As a result, only a few potential sinkhole candidates are passed to the second CNN model, which reduces the classification computation. In the rule-based filter, morphological operations such as opening, closing, dilating, and eroding, are applied to remove the noise that occurs on the edges of the segmented regions. After removing the noise with morphological operations, we apply a connected component analysis to remove the other sinkhole-like objects such as trees, buildings, and cars. Each block of white pixels in the

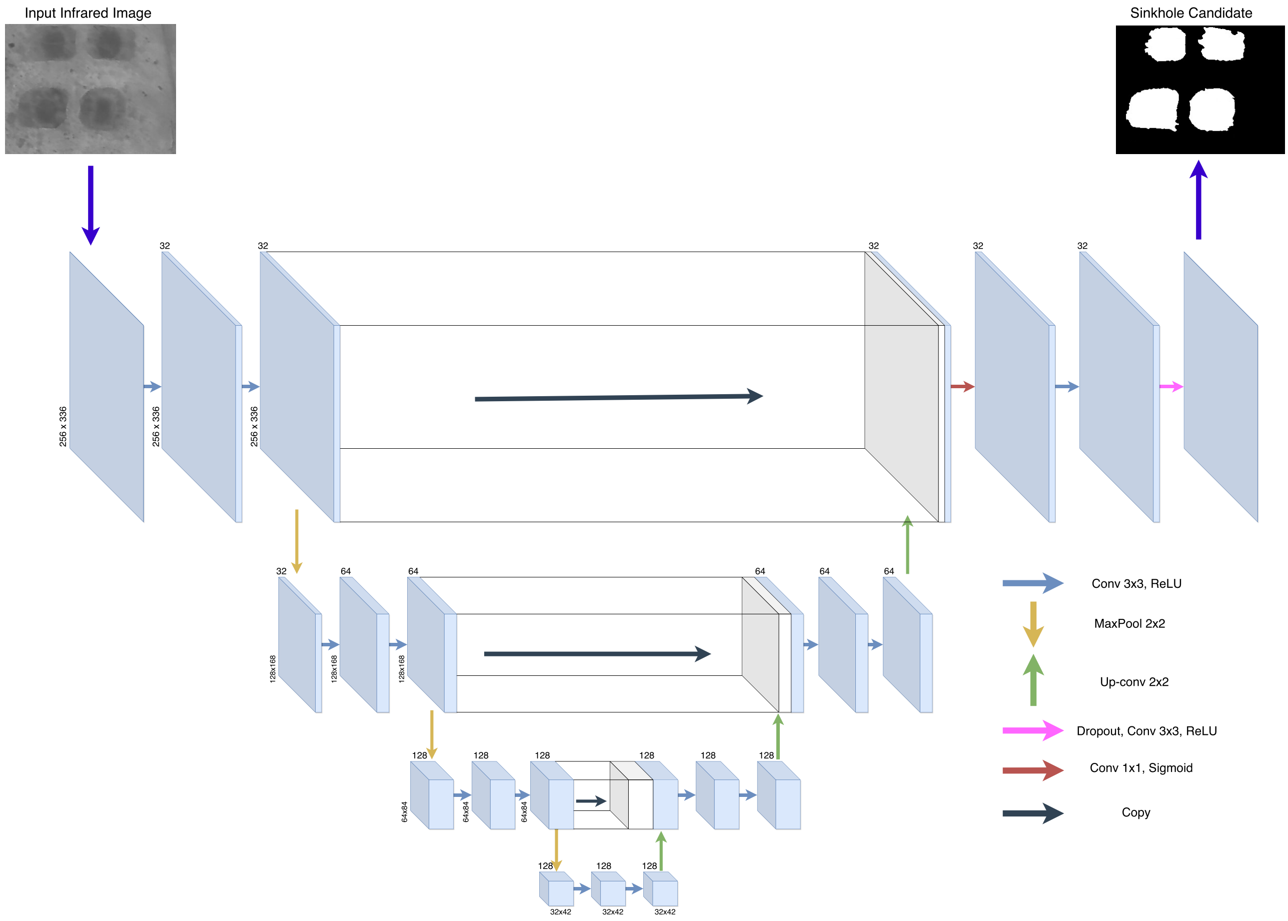


FIGURE 4. Modified UNET Architecture.

segmented image is considered to be connected components that are passed through the rule-based filter. CC_i denotes the i^{th} connected component in the image, w_i is the width of CC_i, h_i is the height of CC_i, and s_i is the area of CC_i, which is the number of white pixels in the i^{th} connected component. The rule-based filter is described by Algorithm 1. In addition, the flicker energy classification [44] is also considered to remove noise in the segmented images. The set of thresholds that have been used in the heuristic filter are chosen empirically. These are $(t_1, t_2, t_3, t_4, t_5, t_6, t_7) = (5, 100, 0.2, 5, 75, 1000, 15)$.

3) SINKHOLE CLASSIFICATION BY CNN TRANSFER LEARNING

With the recent developments in computational technology, deep learning methods have exhibited state-of-the-art performance in solving image classification problems. In this paper, we have implemented a CNN classifier for distinguishing real sinkholes from other objects. Image classification using CNN transfer learning includes two phases: training and prediction. In the training phase, the CNN model is trained by using a known dataset of images labeled with their corresponding types. Once the models are learned, they are used to predict the object types of new images. Because of the numerous parameters, a large dataset and enormous computational resources are required to train CNN models. This leads to difficulties when training with a deficient dataset. Instead, a pretrained model can be transferred to work on the categories that do not belong to the original dataset. In this paper, the transfer learning approach has been used to train a CNN sinkhole classification model.

After the celebrated success of AlexNet [45] at the LSVRC2012 classification contest [46], many CNN models with increased accuracy for image classification have been proposed. However, most of these models include hundreds of millions of parameters, making their real-time application in devices with limited resources difficult. Among these models, MobileNet [47] is a model with fewer parameters that still maintains competitive accuracy compared to other state-of-the-art models. The MobileNet model can run on a mobile device in real time. We use the MobileNet v3 [48] model that has been trained with the ImageNet dataset to perform transfer learning. The use of this MobileNet v3 model ensures that the entire sinkhole detection system can operate in real time. The MobileNet-small model is used, which has one convolutional layer, 11 bottleneck layers, one convolutional layer

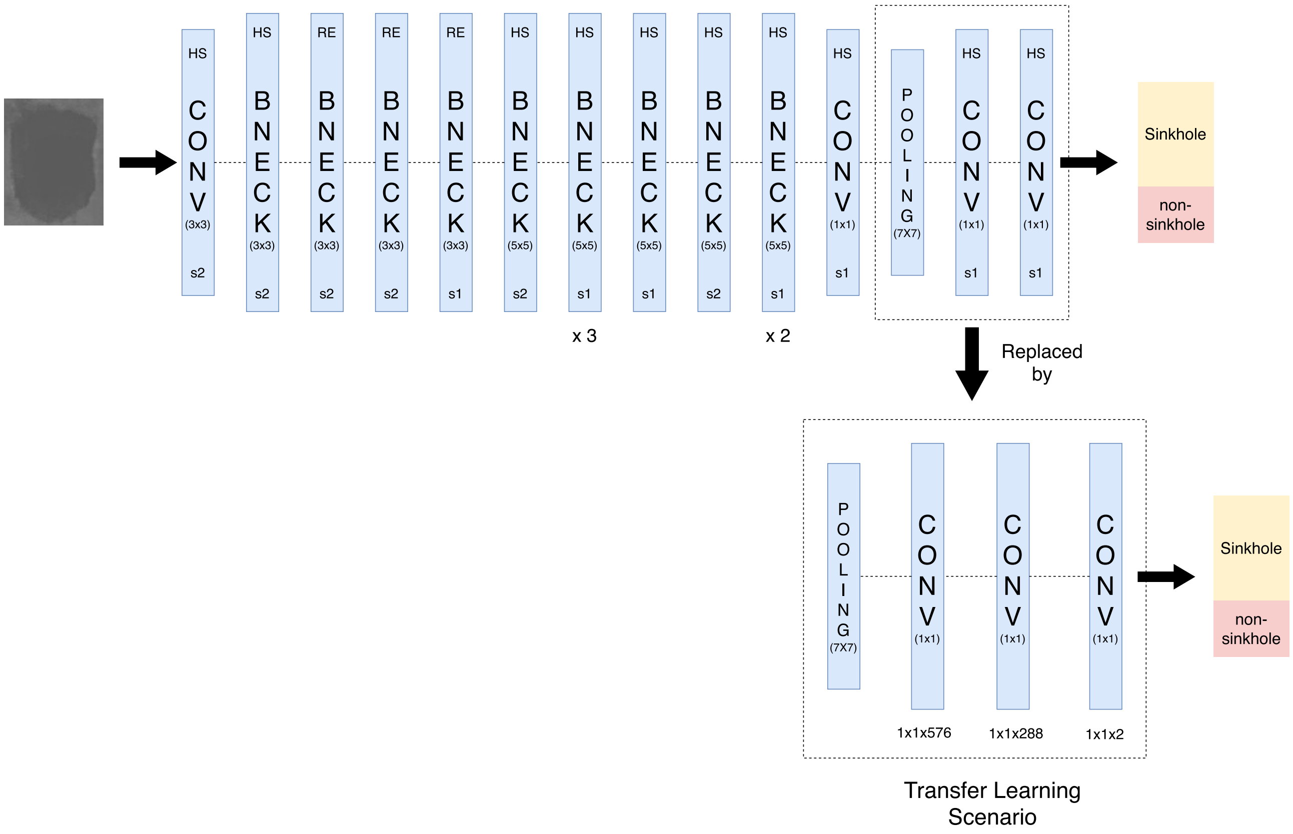


FIGURE 5. MobileNet transfer learning scenario.

with a global pooling layer, and two fully connected layers. For the purpose of classifying sinkhole regions, we replaced the last 3 layers (one pooling layer, two fully connected layers) of the base model with several layers: one pooling layer, two fully connected layers, and one softmax layer. These layers are trained from scratch by using the back-propagation fine-tuning approach with our dataset. The training scenario and MobileNet v3 model are illustrated in Fig. 5.

B. SINKHOLE TRACKING BY HA ALGORITHM

After detecting and recognizing a sinkhole using the CNN classifier, the challenge with tracking is to assign a sinkhole to respective tracklets, which are the trajectories of objects in consecutive frames. To overcome this challenge, we implement the Hungarian algorithm (HA) for data association, which is detailed below. Assuming that we have N detected sinkholes in a video frame, the challenge is how to identify the tracklet to which each detected sinkhole belongs. In addition, let s_{ij} be the score between the i^{th} sinkhole and the data distribution of the j^{th} tracklet. The score is calculated by using correlations between images of detected sinkholes and images of sinkholes in the tracklet. It is obvious that the correlation score is high, and thus, the probability that the sinkhole belongs to the respective tracklet is high. To use the HA to solve the problem, we obtain the distance $d_{ij} = \frac{1}{s_{ij}}$. In addition, x_{ij} defines the relation of the i^{th} sinkhole and the j^{th} tracklet, i.e., $x_{ij} = 1$ if and only if the i^{th} sinkhole is a part of the j^{th} tracklet; otherwise, $x_{ij} = 0$. Therefore, $\sum_{j=1}^{N} x_{ij} = 1$ or one detected sinkhole belongs to only one tracklet, where $i = \overline{1, N}$, and N is also the number of tracklets. The HA is one method used for solving this kind of optimization problem. It minimizes the total cost function, which is addressed in (1). The total cost of the Hungarian algorithm assignment problem is given in the following equation:

$$d = \sum_{i=1}^{N} \sum_{j=1}^{N} d_{ij} x_{ij} \tag{6}$$

The distance matrix of the assignment problem is given as follows:

$$\begin{bmatrix} d_{11} & d_{12} & d_{13} & \dots & d_{1N} \\ d_{21} & d_{22} & d_{23} & \dots & d_{2N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ d_{N1} & d_{N2} & d_{N3} & \dots & d_{NN} \end{bmatrix} \tag{7}$$

In the first step, the HA identifies the minimum distance between each sinkhole and the tracklets; then, it subtracts all the weights d_{ij} from the respective minimum weight.

$$d_{ij} = d_{ij} - min\{d_{ij}\}, j = \overline{1, N} \tag{8}$$

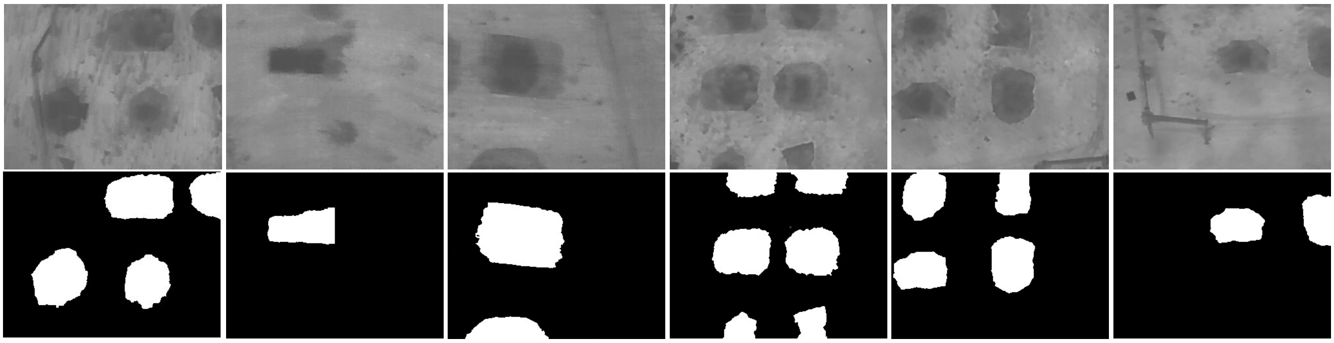


FIGURE 6. Segmented examples using UNET.

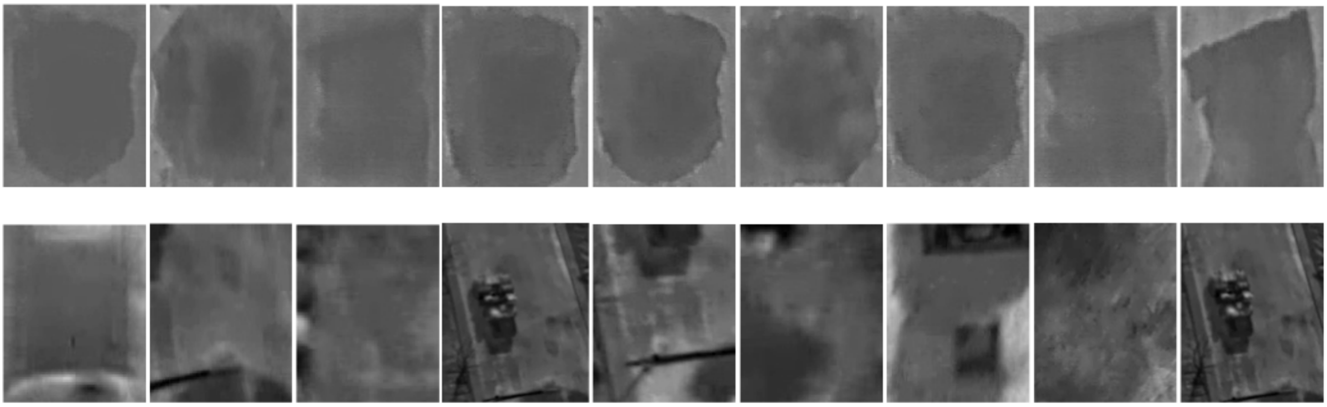


FIGURE 7. Examples of sinkholes (top) and non-sinkholes (bottom).

Similarly, the HA subtracts the smallest distance in each column from all entries in the same column of the distance matrix.

$$d_{ij} = d_{ij} - min\{d_{ij}\}, i = \overline{1, N} \tag{9}$$

After these steps, the distance matrix D contains zero values. In the third step, let n be the minimum number of horizontal/vertical lines that cover all the zero entries of D. If n = N, then the assignment x_{ij} can be made based on the zero values in the distance matrix D. Otherwise, we need to repeat the second step and the third step until the condition in the third step is satisfied.

The dataset that we collected is the video captured from the drone. Therefore, the sinkhole and other objects are stationary; only the camera was moving, and the video was not stable. Sometimes, the segmentation and classification step fails to detect the real sinkhole in the image. In that case, we use the previous sinkhole location in the previous frame with a small translating distance to place into the current frame. The translation distance is then calculated based on the optical flow algorithm. This translation correction is implemented by assuming that all sinkholes in the video frame should be moving in the same direction. This approach helps the system overcome the problem of missing sinkholes in the detection and recognition process.

C. DATASETS

We use the video datasets provided by [1] for our experiments. The videos are collected using a UAV-mounted camera to capture artificial sinkholes on the ground. Sinkholes are dug manually, and half of these artificial sinkholes are filled with water to simulate real conditions. In addition, sinkholes are formed at different depths from 0.5 m to 2 m with a diameter of 1 m or less in 0.5 m increments. To find the optimal time of day to detect sinkholes, we examined the intensive difference between the sinkholes and surrounding areas at different times. The video dataset consists of 16 videos with sinkholes captured at different distances at a resolution of 256×336.

1) THE DATASET FOR UNET-LIKE MODEL

We create a dataset for training the UNET-like model by extracting a separated frame from the video dataset. An individual frame may contain only one sinkhole, a few sinkholes, or even no sinkholes. These frames are annotated to set the sinkhole as the foreground area. Every pixel that belongs to a sinkhole has a value of 1, while the other pixels have a value of 0. Examples of sinkhole annotation images are shown in Fig. 6, and these examples are the same size as the input frame: 256×336. The total number of annotated images is 1236. The dataset is split into a training set that includes 1100 images and a validation set that includes 136 images.

2) THE DATASET FOR THE MobileNet TRANSFER LEARNING MODEL

Our training dataset has 7000 sinkhole images and 7000 non-sinkhole images, while our evaluation dataset has 1000 images for each class. Non-sinkhole images could be regular objects in video frames, such as vehicles, humans, or trees, that have surface temperatures similar to those of real sinkholes; they can also simply be background images. Fig. 7 shows representative images from our dataset. The total number of images in this training dataset is larger than the number in the training dataset for the first CNN model because this training dataset consists of small images of sinkhole objects in the original full-size images (256×336). One original full-size image may contain several smaller images of a sinkhole object.

IV. EXPERIMENTAL RESULTS

A. EXPERIMENTAL SETTINGS

1) UNET-LIKE MODEL

We train our model from scratch with a data augmentation technique to overcome the problem of a limited dataset. We set the maximum number of epochs to 200 and the learning rate to 0.001. The batch size of the training process is 16, which is relatively small compared to other studies of image classification and image segmentation. This batch size setting is selected because our training dataset is less detailed than other popular datasets such as ImageNet. The detailed hyperparameter setting is given in Table 1. The training accuracy and loss of the training process are illustrated in Fig. 8. The model converges after 94 epochs, with the IoU loss decreasing to 0.0230 and the validation accuracy increasing to 95.6%.

TABLE 1. Hyper-parameter setting for UNET model.

Parameter	Value
Maximum Number of Epochs	200
Batch size	16
Initial Learning Rate	0.001
Optimizer	RMSprop

2) MobileNet TRANSFER LEARNING MODEL

The training process of this transfer learning approach is shown in Fig. 9. We started the transfer learning process with a learning rate of 0.01 and dropped it by a factor of ten every 5 epochs. The small learning rate is initially set as the pretrained CNN weights were often good and they would not be too fast distorted. The optimization process runs for a maximum of 100 epochs, which results in an accuracy of over 99% for the trained CNN classifier. Furthermore, the batch size of the training model is 32, which is commonly used in the literature. The detailed hyperparameter setting is given in Table 2. We set the maximum number of epochs as 100. In the actual training process, after 20 epochs, the model seems to converge to the optimal status.

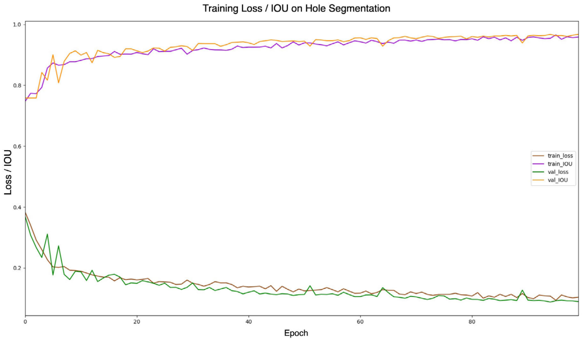


FIGURE 8. Training modified UNET.

TABLE 2. Hyper-parameter setting for MobileNet transfer learning model.

Parameter	Value
Maximum Number of Epochs	100
Batch size	32
Initial Learning Rate	0.01
Optimizer	Adam

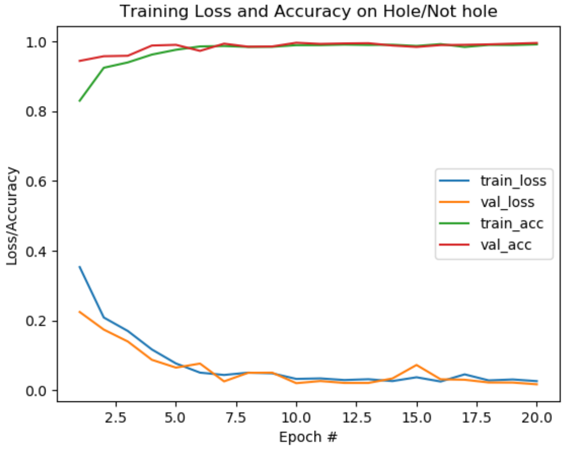


FIGURE 9. Training the MobileNet transfer.

B. PERFORMANCE EVALUATION

1) SEMANTIC SEGMENTATION EVALUATION

We evaluate the UNET performance by using the IoU index, as presented in equation (1). For sinkhole detection, the IoU index is more suitable than the pixel accuracy index. Assessing the model with the pixel accuracy index results in the model predicting small noisy objects in the foreground. The UNET performance is compared with other semantic segmentation models, as shown in Table 3. The UNET models with different contraction layers are also compared. Due to the complexity of the segmentation problem and the small training dataset, we compare UNET with a few contraction layers to avoid overfitting problems that may occur and thus falsify the comparison. The accuracy of the UNET architecture increases by approximately 4% from the 2-layer model to the 3-layer model. The accuracy increases by only 0.2% when the 4-layer model is used. This result proves that the 3-layer UNET model is optimal for sinkhole detection. In comparison with the Otsu method [49] and the fixed thresholding method, the UNET model is superior, with an IoU index greater than 20%. Semantic segmentation is the first step in the entire sinkhole detection process, and the accuracy of semantic segmentation affects the final result. The goal of semantic segmentation is to detect all sinkholes, including any noisy object that will be eliminated by the filters at the next stage. The results shown in Table 3 also illustrate the impact of applying a rule-based filter as the next step in multilevel CNN sinkhole detection. The IoU of the 3-layer UNET model is increased by 2% after utilizing the rule-based filter to segmented sinkhole images. As we have mentioned earlier in this paper, our method has an advantage over the [1] method, as our method does not need to set the predefined threshold to segment the sinkhole image. These kinds of thresholds have been learned inside the UNET model. However, we implement a rule-based filter to eliminate noise appearing after the segmentation process. This filter has thresholds that are set through data observation. Using this filter may increase the accuracy of the system but may also encounter the threshold problem of the method proposed in [1]. This rule-based filter is optional because practical applications often require threshold-free models. It is thus possible to remove the rule-based filter to meet the requirement while the accuracy is assuredly reasonable.

TABLE 3. Segmentation evaluation.

Model	IoU Score	With Rule-based Filter
Unet-like 2 Blocks	0.902	0.912
Unet-like 3 Blocks	**0.94**	**0.961**
Unet-like 4 Blocks	0.942	0.962
Otsu Binarization	0.765	0.812
Fixed Thresholding	0.673	0.721

Among the proposed system's stages, semantic segmentation is the stage that takes the most processing time. Therefore, we conducted an assessment of the resource consumption of the proposed models. The evaluation details are given in Table 4, which are suitable for application to resource-constrained systems. The most common memory consumption model is the 4-block CNN with 9.5 MB. This model size is efficient enough compared to memory capacity of modern embedded boards such as Jetson TX2.

TABLE 4. Resource consumption comparison of different Unet-like models.

Model	Model Parameter	Model Size (MB)
Unet-like 2 Blocks	466,529	3.9
Unet-like 3 Blocks	926,753	7.8
Unet-like 4 Blocks	1,127,777	9.5

2) CLASSIFICATION EVALUATION

The accuracy of the classification model is assessed on the validation set, which includes 2000 images of sinkholes and non-sinkholes. We evaluate models such as 3-layer CNNs, 4-layer CNNs, MobileNet transfer learning and HOG + SVM. The results presented in Table 5 show that the transfer learning model that uses MobileNet as the base model achieves a superior accuracy of 97.6%. The MobileNet model is also a lightweight model that can help the system achieve high accuracy but still ensure real-time operation. The MobileNet model was originally proposed for deployment on a resource-constrained mobile device. Therefore, its architecture was optimized to be as lightweight as possible compared to other state-of-the-art classification models. Although the use of the MobileNet model can be slightly more complex than the CNN model, the MobileNet's accuracy is significantly higher. Therefore, we utilized the MobileNet model to solve this classification problem to balance accuracy and processing time. Furthermore, the sinkhole objects in this sinkhole image dataset have a simpler structure than the objects in the ImageNet dataset, and the MobileNet architecture can classify them efficiently. There are few failure cases of this classification model due to sinkhole-like objects, which have similar shapes and brightness, such as cars, roofs, and groups of trees. We have also gathered sinkhole-like objects into negative examples (non-sinkhole group). In addition, the real sinkhole objects have various depths so that the color distribution (calculated via the temperature of the object surface by a thermal camera) of the training data is uneven, somehow leading to the difficulty of the model convergence.

TABLE 5. Classification evaluation.

Model	Accuracy
3 Layers CNN from Scratch	0.902
4 Layers CNN from Scratch	0.904
MobileNet Transfer Learning	**0.976**
HOG + SVM	0.88

3) OVERALL SINKHOLE DETECTION EVALUATION

The overall accuracy of the sinkhole detection method is compared with that of the method in [1]. We use the 6 videos in the dataset, which was mentioned earlier, to perform this assessment. These videos are not used in the process of creating two sets of the dataset for the UNET segmentation model and classification model training to objectively evaluate the performance of the method. To compare the performance of our proposed method with that of the method in [1], we estimate the average detection precision and the average detection recall with the following equations.

$$AP = \frac{TP}{TP + FP} \tag{10}$$

$$AR = \frac{TP}{TP + FN} \tag{11}$$

where TP is the number of true positives, FN is the number of false negatives, and FP is the number of false positives in the dataset. Based on the overlapping threshold, we can identify when a detected sinkhole is an FP, FN or TP. A detected object (sinkhole or non-sinkhole) is an FP when it is classified as a sinkhole but is not a real sinkhole. A detected object is a TP when it is classified as a sinkhole, is a real sinkhole and when the overlapping area between the detected sinkhole and the ground truth is larger than an overlapping threshold. A detected object is an FN when it is classified as a non-sinkhole but is a real sinkhole. The higher overlapping threshold may result in a decreasing TP ratio if the detected sinkhole does not fit exactly with the ground truth. Setting an overlapping threshold is necessary to evaluate the performance of the system because the risk of sinkhole formation has to be evaluated by using information on both the sinkhole location and the sinkhole area in a timely manner. Our method implements a UNET model for the semantic segmentation problem, and the correct detection rate of TPs is high even when the overlapping threshold is set too high. In this evaluation, we set the overlapping threshold to 0.6 to compare our method and the method in [1] and two methods in [50], Where the author compared two methods for object detection problem which are Faster RCNN [34] based on VGG16 model [51] and two stages architecture based on AlexNet [45] and Gaussian Mixture Model (GMM) [52]. We estimate the F1 score, which is obtained by the following equation to be the evaluation metric of the comparison, which can then be described by the equation for the overlapping threshold.

$$F1\ Score = 2 \times \frac{AR \times AP}{AR + AP} \tag{12}$$

The average F1 scores per video are described in Fig. 11. Our method achieves the highest F1 score compared to the other methods for all videos. Notably, our method outperforms the AlexNet+GMM and HOG+SVM+SlidingWindow detector by a large margin. This can be explained by the fact that the AlexNet+GMM method utilizes GMM background modeling for extracting the moving objects that are often highly noisy in the case of complex background. In addition, the background environment from the UAV image is not stable, which leads to the performance of GMM not being good for detecting the sinkholes. The performance of the HOG+SVM+SlidingWindow method significantly depends on the size of the sliding window and the size of the real sinkholes. The Faster RCNN by VGG16, in contrast, can achieve a comparable performance to our proposed method. However, the processing speed of the Faster RCNN by VGG16 is languid, so this model cannot be applied for the real-time sinkhole detection. A processing speed comparison is given in detail in Table 8. The improvement of our proposed method compared to other methods in accuracy and processing speed is because of the use of the multilevel CNN model. The UNET segmentation model helps the system overcome the hard threshold that the traditional binary segmentation has to set to obtain the binary image. Additionally, the rule-based filter helps the system reduce false positive detection. Furthermore, our proposed system can operate as an end-to-end system during training and testing time by using two CNN models. The model is able to automatically adapt according to the training data. We also plot an ROC curve [53]–[55] to compare these mentioned methods in Fig. 10. ROC curves are created by plotting the true positive rate and false positive rate at various threshold settings (changing thresholds). In this sinkhole detection scenario, the changing thresholds are the set of overlapping thresholds (ranging from 0 to 1). The higher the threshold is, the more difficult it is for a predicted sinkhole to become a true positive sample. As shown in this figure, the area under the curve of our proposed method is the largest, which proves the discriminative ability of our approach.

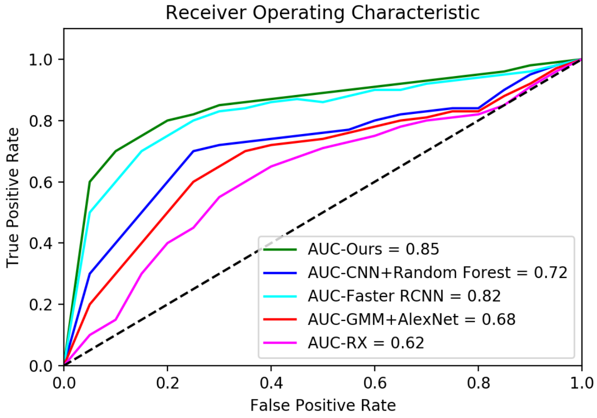


FIGURE 10. ROC curve of sinkhole detection methods.

Conventionally, CPT (cone penetrometer testing) and GPR (ground-penetrating radar) are operated in the field to detect sinkholes. However, since CPT methods are destructive test methods, the sinkhole may be worsened in some cases, and the cost and time for detecting one sinkhole are more inefficient than those of the method proposed in this paper. In addition, GPR is a method of analyzing the reflected signal by sending radar into the ground, and the accuracy of the reflected signal may be greatly distorted due to a nonuniform medium inside the ground. In addition, there are disadvantages that are greatly affected by the detectable depth and the resolution of the reflected radar signal depending on the radar frequency. For example, the use of a low frequency results in deep ground detection but low resolution, and a high frequency decreases the transmittance, resulting in low depth detection. Therefore, the method proposed in this paper is more efficient than a traditional method in terms of cost and time, and it is expected that the accuracy will be very high by using deep learning.

4) SINKHOLE TRACKING EVALUATION

To evaluate sinkhole tracking by data association, we calculated the IoU score for 6 videos. Detailed results are described in Table 6, with an average IoU of 0.88. The best video is video 11, which has an IoU of 0.92, while the worst is video 1 with an IoU of 0.83. This result is also consistent with the

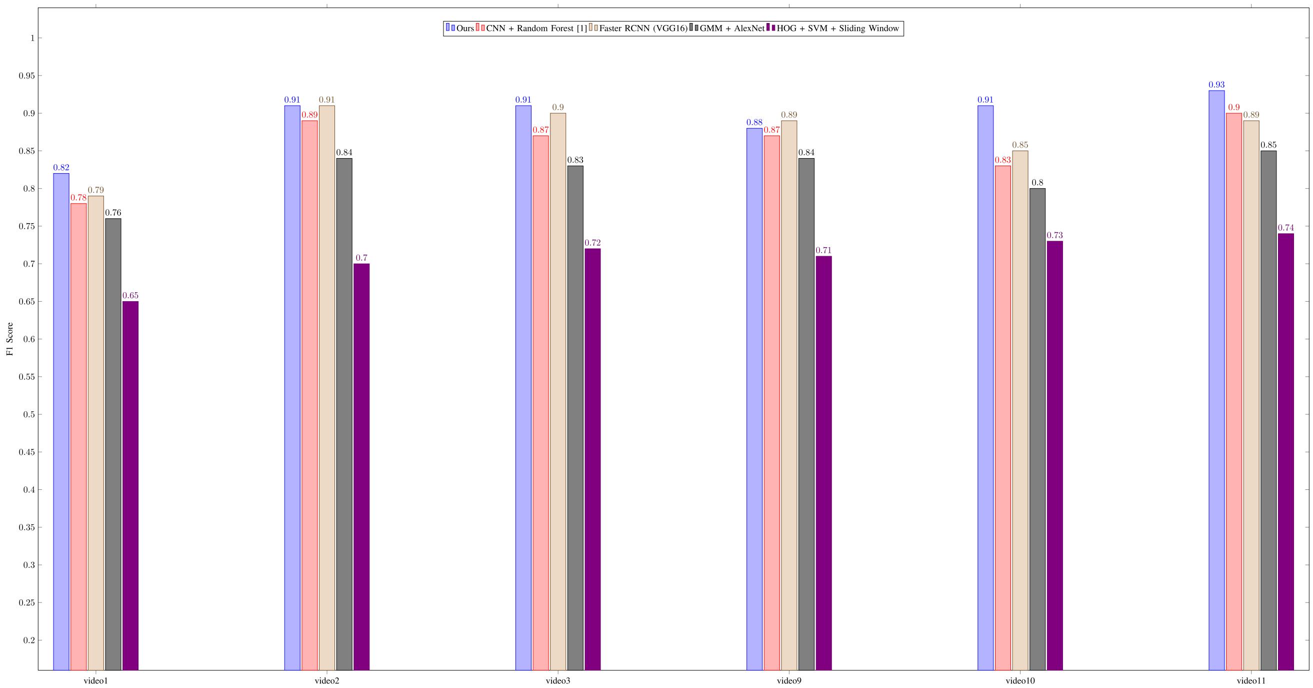


FIGURE 11. Sinkhole detection comparison.

analysis given in this database. Video 1 was collected in the spring when the temperature difference between the sinkhole and the surrounding area was lower than the large difference between the sinkhole and the surrounding area in the summer when video 11 was captured. Using the tracking algorithm in the context of detecting sinkholes presents several challenges. The UAV can travel at different heights, resulting in a significant change in the size of the sink. In addition, high-speed UAV movement can lead to blurred sinkhole images. These situations present considerable challenges for the detection algorithm; in some cases, the output videos are unstable and unreliable. To address these issues, we apply the tracking algorithm to minimize these weaknesses that make the output detection results smoother. In reality, the number of sinkholes appearing in the single monitoring frame is not too high. Therefore, the implementation of the sinkhole tracking model is lightweight and computationally efficient. The tracking algorithm helps improve the accuracy of sinkhole detection compared to that with the use of sinkhole detection only but still assures the real-time requirement of the system. From an application perspective, this improvement is mandatory when the output of the system has to be exactly the same as the number of sinkholes in the monitoring area.

5) PROCESSING TIME EVALUATION ON THE EMBEDDED DEVICE

To evaluate the performance of the proposed method and to confirm that the system is able to work in real-world applications, we implement our proposed system in Jetson TX2 by Nvidia, which is a processing board equipped with GPU cores. In reality, the drone can carry a Jetson TX2 board to process images from the air, after which some important information, such as the image of the sinkhole and the location of the sinkhole by GPS coordinates of the drone, can be transmitted to the ground station via other communication protocols. The processing times for all stages are shown in Table 7, where the modified UNET is the most demanding stage, as it requires an average of 50 ms to process one image. Our proposed system can process streaming video at 13.2 FPS on an embedded board, which proves the feasibility and practicality of deploying this system in real-time applications. The performance of the whole system can be greatly improved by replacing the traditional CNN elements in UNET with other efficient elements from the literature. Furthermore, the development of artificial intelligence accelerators in modern processors makes the whole proposed system even lighter in weight and ensures real-time processing. In future work, we plan to make the processing board lighter to enhance the operating time of the UAV.

TABLE 6. Tracking evaluation.

Video	Average IoU Score
Video 1	0.83
Video 2	0.87
Video 3	0.86
Video 9	0.91
Video 10	0.90
Video 11	0.92
Average	0.88

TABLE 7. Evaluation of the embedded system.

Processing Step	Processing Time on Jetson TX2 (ms)
Modified U-net	49
Rule-based Filter	1.2
MobileNet Transfer	24
Other Stage	1.3
Total	75.6

A comparison of our proposed method to the other methods on the Jetson TX2 embedded board is described in Table 8. One important thing worth noting in the comparison is that the Jetson TX2 board contains GPU (graphics processing unit) cores that support deep learning models. Therefore, deep learning models that run on this board are greatly accelerated compared to classical object detectors. Our proposed method achieves the highest performance of 13.2 FPS, while the slowest method is Faster RCNN by VGG16 with 1.3 FPS. The slow processing speed of Faster RCNN by VGG16 can be explained by the fact that Faster RCNN has a region proposal network to obtain all potential object locations, which are then classified by a CNN-based classifier. The classifier will classify these object locations into different object types. The number of possible object locations is enormous, which leads to the entire process being slow. The HOG+SVM+SlidingWindow object detector is quite fast because of its lightweight model. However, the accuracy of the model depends heavily on the size of the sliding window and the size of the real sinkhole, which can be barriers to applying the HOG+SVM+SlidingWindow object detector in practice.

TABLE 8. Processing time comparison of different methods on Jetson TX2.

Method	**Processing speed (FPS)**
CNN + Random Forest [1]	5.3
Faster RCNN by VGG16 [50]	1.3
GMM + AlexNet [50]	4.7
SVM + HOG + Sliding Window	11.4
Our proposed method	**13.2**

V. DISCUSSION

Sinkholes that appear in residential areas cause great damage to infrastructure and great harm to society. Unlike traditional methods based on CPT, GPR, ERT, and GB-InSAR, the method using thermal imaging on UAVs has the advantage of monitoring large areas at a lower cost. Although the method of using thermal imaging on artificial datasets shows promise, sinkhole detection simulation still has many limitations for evaluating performance comparatively because a sinkhole can exist in various forms and can be affected by soil and weather conditions. In addition, the size of the sinkhole can vary from a few meters to several dozens of meters. Furthermore, the method of using thermal imaging on a drone to collect data and monitor sinkholes also has many additional limitations, such as the following:

- The limitation of the power supply does not allow the drone to continuously operate for several hours, resulting in a limited surveillance area that the drone can monitor.

- The limitation of computational resources does not allow for the deployment of complex image processing algorithms on the drone.

- Drone activities in residential areas may lead to dangerous accidents. The drone may collide with trees and buildings. If the drone falls, people can be harmed. Therefore, drone use should be restricted in some areas.

- Extreme weather conditions also seriously affect drone operations.

- The distance of the drone to the sinkhole might affect the accuracy of the detection system, as the drone is not able to capture the details of the sinkhole if the distance between the drone and sinkholes is too great.

For future work, a thermal camera mounted on the roof of a car is suggested as an alternative method to detect sinkholes using thermal imaging in populated areas, on roads and on pedestrian walkways. In addition, the camera should be able to rotate around the vehicle at a 360-degree angle to expand the view. The height of the camera should also be adjustable to optimize the distance from the camera to the sinkhole. The concept of utilizing a thermal camera mounted on a car will have the following benefits:

- It ensures safe operation in residential areas. In addition, a thermal camera-equipped car can travel flexibly, allowing the system to monitor a wide range of areas.

- System performance is limited less by power supply. Therefore, the system can operate for several days without charging the battery.

- High-performance computers can be installed in the car to allow the system to operate in real time with expected accuracy.

- Available GPS in the car also allows the information of detected sinkholes to be mapped in real time via the on-board networking system.

- System performance is less affected by bad weather than it is in a drone-based system.

The real size of a sinkhole is an important factor that a monitoring system must determine. In this scenario, the size of the sinkhole can be estimated with some calibration steps and the assumption that the camera view is perpendicular to the ground plane based on the simple formula of camera optical geometry, which is shown in the following equation:

$$\frac{D}{F} = \frac{S^r}{S^i} \tag{13}$$

where D is the distance from the sinkhole to the camera, F is the focal length of the camera, Sr is the size of the real sinkhole, and Si is the size of the sinkhole in the captured image, as shown in Fig. 11. Then,

$$F = \frac{D \times S^i}{S^r} \tag{14}$$

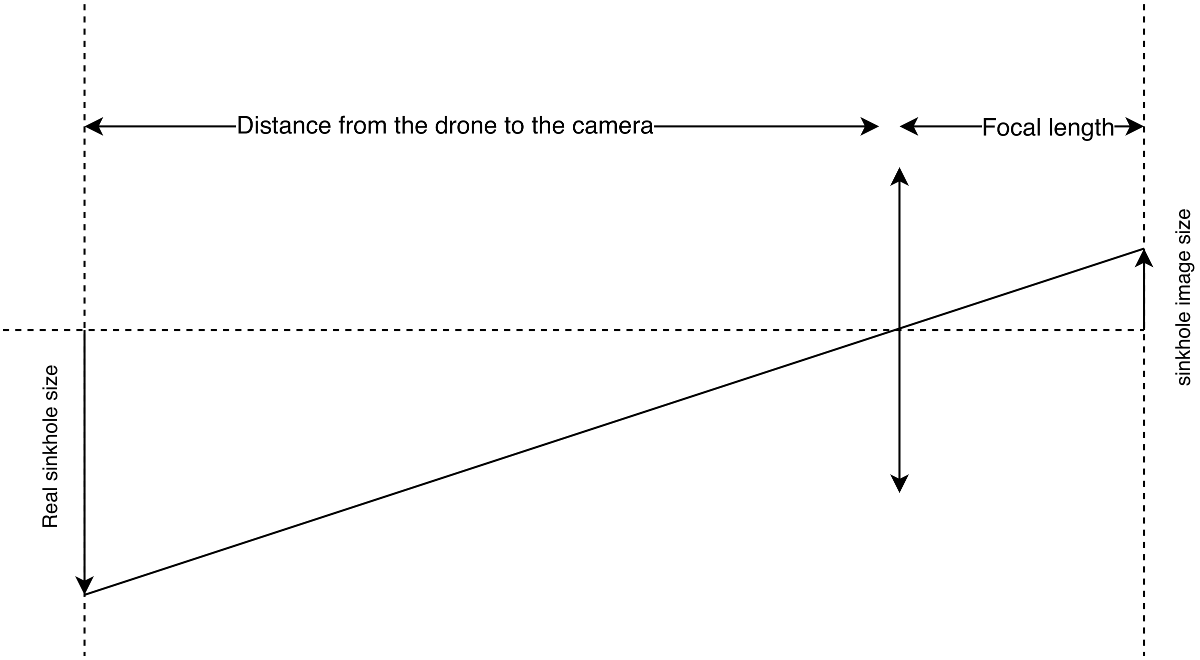


FIGURE 12. Calculation of the real size of the sinkhole.

We obtain the following formula:

$$\frac{D_1 \times S_1^i}{S_1^r} = \frac{D_2 \times S_2^i}{S_2^r} \tag{15}$$

where D_1 and D_2 are the distances from the camera to the sinkholes; S_1^r and S_2^r are the sizes of the real sinkhole; S_1^i and S_2^i are the sizes of the sinkhole in the captured image at two different times. The distance from the camera to the sinkhole is extracted from the height of the drone (in meters). The size of the sinkhole in the image is calculated in pixels. At the calibration stage, this information (D_1, S_1^i, S_1^r) is extracted as the reference value. Then, the real size of the detected sinkhole with the height of drone D_2 and the sinkhole size in image S_2^i is calculated as follows:

$$S_2^r = \frac{D_2 \times S_2^i \times S_1^r}{D_1 \times S_1^i} \tag{16}$$

VI. CONCLUSION

In this paper, we propose an approach that combines a multilevel CNN model for sinkhole detection and HA data association to efficiently detect and track sinkholes in real time. Our experiments show that the proposed method achieves promising results compared to existing approaches. Our proposed method has some weaknesses. For example, when the input video is not stable, the segmentation step requires more annotated data to fit the model, and the input of potential candidates is missing from the training set. Despite some weaknesses, our proposed method shows feasibility for sinkhole detection using infrared cameras, with an accuracy as high as 97,6% for sinkhole detection and an IoU of 88% for sinkhole tracking, which is suitable for practical applications such as use in surveillance systems. Furthermore, the advantages of our proposed models and system can be highlighted as follows:

- The proposed model is threshold-free and therefore applicable in various environmental conditions for detecting and tracking sinkholes.

- The system is deployable on embedded boards while can be processed in real time.

- The trained models can be utilized to perform transfer learning to other similar applications such as [11], [12] in medical image analysis and [15], [16] in facial emotion recognition. These applications process the detection task in 2 stages: image segmentation using UNET-like models and image classification using a deep pretrained model.

In future work, the system can be expanded to control a group of UAVs to monitor larger areas, and the detected sinkholes with their respective GPS locations can be transferred to the ground station to draw a real-time map of sinkholes. The real-time map can be integrated to fire alarms to provide alerts to the existence of sinkholes for public safety and facilities.

ACKNOWLEDGMENT

Sinkhole video used in this article for learning and testing had been provided by Prof. B. C. Ko from Keimyung University.

REFERENCES

[1] E. J. Lee, S. Y. Shin, B. C. Ko, and C. Chang, "Early sinkhole detection using a drone-based thermal camera and image processing," *Infr. Phys. Technol.*, vol. 78, pp. 223–232, Sep. 2016.

[2] E. Intrieri, G. Gigli, M. Nocentini, L. Lombardi, F. Mugnai, F. Fidolini, and N. Casagli, "Sinkhole monitoring and early warning: An experimental and successful GB-InSAR application," *Geomorphology*, vol. 241, pp. 304–314, Jul. 2015.

[3] T. L. Dobecki and S. B. Upchurch, "Geophysical applications to detect sinkholes and ground subsidence," *Lead. Edge*, vol. 25, no. 3, pp. 336–341, Mar. 2006.

[4] A. Billi, L. De Filippis, P. P. Poncia, P. Sella, and C. Faccenna, "Hidden sinkholes and karst cavities in the travertine plateau of a highly-populated geothermal seismic territory (Tivoli, central Italy)," *Geomorphology*, vol. 255, pp. 63–80, Feb. 2016.

[5] D. Bloomquist, R. L. Shrestha, and C. Slatton, "Early sinkhole detection and verification using airborne laser and infrared technologies," Dept. Civil Coastal Eng., Univ. Florida, Gainesville, FL, USA, Tech. Rep. BC-354-54, 2005.

[6] Shannon, J. Clint, D. Moore, Y. Li, and C. Olsen, "LiDAR-based sinkhole detection and mapping in Knox County, Tennessee," *Pursuit J. Undergraduate Res. Univ. Tennessee*, vol. 9, no. 1, p. 3, 2019.

[7] S. Zhang, S. Baros, and S. Bogus, "Karst sinkhole detecting and mapping using airborne LiDAR," Transp. Consortium South-Central States, Louisiana State Univ., Baton Rouge, LA, USA, Tech. Rep. 18GTUNM01, 2019.

[8] S. Zhang, S. Baros, S. Bogus, P. Neville, and R. Dow, "Karst sinkhole detecting and mapping using airborne LiDAR-A conceptual framework," in *Proc. MATEC Web Conf.*, vol. 271, 2019, Art. no. 02005.

[9] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, Dec. 1989.

[10] A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, "Deep learning for computer vision: A brief review," *Comput. Intell. Neurosci.*, vol. 2018, pp. 1–13, Feb. 2018.

[11] N. Zeng, Z. Wang, H. Zhang, W. Liu, and F. E. Alsaadi, "Deep belief networks for quantitative analysis of a gold immunochromatographic strip," *Cognit. Comput.*, vol. 8, no. 4, pp. 684–692, Aug. 2016.

[12] N. Zeng, Z. Wang, H. Zhang, K.-E. Kim, Y. Li, and X. Liu, "An improved particle filter with a novel hybrid proposal distribution for quantitative analysis of gold immunochromatographic strips," *IEEE Trans. Nanotechnol.*, vol. 18, pp. 819–829, 2019.

[13] Q. Wang, L. Zhang, W. Zou, and K. Kpalma, "Salient video object detection using a virtual border and guided filter," *Pattern Recognit.*, vol. 97, Jan. 2020, Art. no. 106998.

[14] N. Wang and X. Gong, "Adaptive fusion for RGB-D salient object detection," *IEEE Access*, vol. 7, pp. 55277–55284, 2019.

[15] N. Zeng, H. Zhang, B. Song, W. Liu, Y. Li, and A. M. Dobaie, "Facial expression recognition via learning deep sparse autoencoders," *Neurocomputing*, vol. 273, pp. 643–649, Jan. 2018.

[16] J.-H. Kim, B.-G. Kim, P. P. Roy, and D.-M. Jeong, "Efficient facial expression recognition algorithm based on hierarchical deep neural network structure," *IEEE Access*, vol. 7, pp. 41273–41285, 2019.

[17] B.-F. Wu and C.-H. Lin, "Adaptive feature mapping for customizing deep learning based facial expression recognition model," *IEEE Access*, vol. 6, pp. 12451–12461, 2018.

[18] B. Yang, J. Cao, R. Ni, and Y. Zhang, "Facial expression recognition using weighted mixture deep neural network based on double-channel facial images," *IEEE Access*, vol. 6, pp. 4630–4640, 2018.

[19] Q. Qi, Q. Huo, J. Wang, H. Sun, Y. Cao, and J. Liao, "Personalized sketch-based image retrieval by convolutional neural network and deep transfer learning," *IEEE Access*, vol. 7, pp. 16537–16549, 2019.

[20] K. Song, F. Li, F. Long, J. Wang, and Q. Ling, "Discriminative deep feature learning for semantic-based image retrieval," *IEEE Access*, vol. 6, pp. 44268–44280, 2018.

[21] X. Li, Z. Chen, Q. M. J. Wu, and C. Liu, "3D parallel fully convolutional networks for real-time video wildfire smoke detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 1, pp. 89–103, Jan. 2020.

[22] N. M. Dung, D. Kim, and S. Ro, "A video smoke detection algorithm based on cascade classification and deep learning," *KSII Trans. Internet Inf. Syst.*, vol. 12, no. 12, pp. 1–16, 2018.

[23] K. Muhammad, J. Ahmad, I. Mehmood, S. Rho, and S. W. Baik, "Convolutional neural networks based fire detection in surveillance videos," *IEEE Access*, vol. 6, pp. 18174–18183, 2018.

[24] R. Nawaratne, D. Alahakoon, D. De Silva, and X. Yu, "Spatiotemporal anomaly detection using deep learning for real-time video surveillance," *IEEE Trans. Ind. Informat.*, vol. 16, no. 1, pp. 393–402, Jan. 2020.

[25] S. Bansod and A. Nandedkar, "Transfer learning for video anomaly detection," *J. Intell. Fuzzy Syst.*, vol. 36, no. 3, pp. 1967–1975, Mar. 2019.

[26] A. Ullah, J. Ahmad, K. Muhammad, M. Sajjad, and S. W. Baik, "Action recognition in video sequences using deep bi-directional LSTM with CNN features," *IEEE Access*, vol. 6, pp. 1155–1166, 2018.

[27] Y. Wu, Y. Sui, and G. Wang, "Vision-based real-time aerial object localization and tracking for UAV sensing system," *IEEE Access*, vol. 5, pp. 23969–23978, 2017.

[28] H. Tayara, K. Gil Soo, and K. T. Chong, "Vehicle detection and counting in high-resolution aerial images using convolutional regression neural network," *IEEE Access*, vol. 6, pp. 2220–2230, 2018.

[29] H. D. Nguyen, I. S. Na, S. H. Kim, G. S. Lee, H. J. Yang, and J. H. Choi, "Multiple human tracking in drone image," *Multimedia Tools Appl.*, vol. 78, no. 4, pp. 4563–4577, Feb. 2019.

[30] M. N. Reza, I. S. Na, S. W. Baek, and K.-H. Lee, "Rice yield estimation based on K-means clustering with graph-cut segmentation using low-altitude UAV images," *Biosyst. Eng.*, vol. 177, pp. 109–121, Jan. 2019.

[31] P. Horstrand, R. Guerra, A. Rodriguez, M. Diaz, S. Lopez, and J. F. Lopez, "A UAV platform based on a hyperspectral sensor for image capturing and on-board processing," *IEEE Access*, vol. 7, pp. 66919–66938, 2019.

[32] N. V. Hoai, N. M. Dung, and S. Ro, "Sinkhole detection by deep learning and data association," in *Proc. 11th Int. Conf. Ubiquitous Future Netw. (ICUFN)*, Jul. 2019, pp. 211–213.

[33] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[34] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[35] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[36] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 7263–7271.

[37] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*. [Online]. Available: http://arxiv.org/abs/1804.02767

[38] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015.

[39] Z. Zou, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," 2019, *arXiv:1905.05055*. [Online]. Available: http://arxiv.org/abs/1905.05055

[40] J. Munkres, "Algorithms for the assignment and transportation problems," *J. Soc. Ind. Appl. Math.*, vol. 5, no. 1, pp. 32–38, Mar. 1957.

[41] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent*, Cham, Switzerland: Springer, 2015, pp. 234–241.

[42] Y. Yuan, M. Chao, and Y.-C. Lo, "Automatic skin lesion segmentation using deep fully convolutional networks with jaccard distance," *IEEE Trans. Med. Imag.*, vol. 36, no. 9, pp. 1876–1886, Sep. 2017.

[43] H. N. Vu, T. A. Tran, N. I. Seop, and S. H. Kim, "Extraction of text regions from complex background in document images by multilevel clustering," *Int. J. Netw. Distrib. Comput.*, vol. 4, vol. 1, pp. 11–21, 2016.

[44] N. M. Dung and S. Ro, "Algorithm for fire detection using a camera surveillance system," in *Proc. Int. Conf. Image Graph. Process. ICIGP*, 2018, pp. 38–42.

[45] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[46] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[47] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*. [Online]. Available: http://arxiv.org/abs/1704.04861

[48] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, Q. V. Le, and H. Adam, "Searching for MobileNetV3," 2019, *arXiv:1905.02244*. [Online]. Available: http://arxiv.org/abs/1905.02244

[49] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, no. 1, pp. 62–66, Jan. 1979.

[50] J. E. Espinosa, S. A. Velastin, and J. W. Branch, "Vehicle detection using alex net and faster R-CNN deep learning models: A comparative study," in *Proc. Int. Vis. Informat. Conf.*, Cham, Switzerland: Springer, 2017, pp. 3–15.

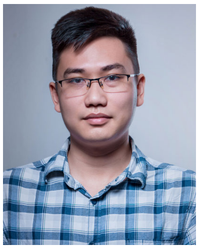
[51] K. Simonyan and A. Zisserman, ''Very deep convolutional networks for large-scale image recognition,'' 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556

[52] D. Reynolds, ''Gaussian mixture models,'' in *Encyclopedia of Biometrics*. Springer, 2015, pp. 827–832.

[53] S. Matteoli, T. Veracini, M. Diani, and G. Corsini, ''A locally adaptive background density estimator: An evolution for RX-based anomaly detectors,'' *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 1, pp. 323–327, Jan. 2014.

[54] R. Zhao, B. Du, and L. Zhang, ''Hyperspectral anomaly detection via a sparsity score estimation framework,'' *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 6, pp. 3208–3222, Jun. 2017.

[55] N. M. Nasrabadi, ''Regularization for spectral matched filter and RX anomaly detector,'' *Proc. SPIE*, vol. 6966, Apr. 2008, Art. no. 696604.



HOAI NAM VU was born in Hanoi, Vietnam, in 1990. He received the B.E. degree in electronic and telecommunication engineering from the Hanoi University of Science and Technology, Hanoi, in 2013, and the M.S. degree in electronic and computer engineering from Chonnam National University, Gwangju, South Korea, in 2015. He is currently pursuing the Ph.D. degree in computer science with the Posts and Telecommunications Institute of Technology, Hanoi. Since 2016, he has been a Lecturer with the Computer Science Department, Posts and Telecommunications Institute of Technology. His research interests include drone-based image processing, machine learning, and deep learning.

CUONG PHAM received the B.S. degree in computer science from Vietnam National University, in 1998, the M.S. degree in computer science from New Mexico State University, USA, in 2005, and the Ph.D. degree in computer science from Newcastle University, U.K., in 2012. He was a Marie Curie Research Fellow with Philips Research, Eindhoven, The Netherlands. He is currently an Associate Professor of computer science with the Posts and Telecommunications Institute of Technology (PTIT). His main research interests include ubiquitous computing, wearable computing, human-activity recognition, and machine learning/deep learning.

NGUYEN MANH DUNG received the B.S. degree from the Department of Electronics and Telecommunication Engineering, Hanoi University of Science and Technology, in 2005, and the M.S. and Ph.D. degrees from the Department of Information and Communication, Kongju National University, in 2009 and 2019, respectively. Since 2020, he has been a Lecturer with the Electronic Engineering Department, Posts and Telecommunications Institute of Technology, Vietnam. His research interests include embedded systems, image processing, and video analysis algorithms for surveillance camera systems.

SOONGHWAN RO received the B.S., M.S., and Ph.D. degrees from the Department of Electronics Engineering, Korea University, in 1987, 1989, and 1993, respectively. He was a Research Engineer with the Electronics and Telecommunications Research Institute, University of Birmingham, in 1997 and 2003, respectively. Since March 1994, he has been a Professor with Kongju National University, South Korea. His research interests include 5G communication, mobile networks, and embedded systems.