J. Vis. Commun. Image R. 96 (2023) 103933

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

Full Length Article

# Detecting Water in Visual Image Streams from UAV with Flight Constraints ☆

Harin Samaranayake, Oshan Mudannayake, Dushani Perera, Prabhash Kumarasinghe *, Chathura Suduwella, Kasun De Zoysa, Prasad Wimalaratne

*University of Colombo School of Computing, Colombo 07, Sri Lanka*

ARTICLE INFO

*Keywords:*
Water Surface Identification
Unmanned Ariel Vehicles
Drones
UNet
Dense Optical Flow

ABSTRACT

Unmanned Ariel Vehicles (UAVs) require identifying water surfaces during flight maneuvers, mainly for safety in execution and its applications. We introduce two novel techniques to identify water surfaces from front-facing and downward-facing cameras mounted on a UAV. The first method — UNet-RAU, a unique architecture based on UNet and Reflection Attention Units, segments water pixels from front-facing camera views, utilizing the reflection property of water surfaces. On the On-Road and Off-Road datasets of Puddle-1000, UNet-RAU improved its performance by 2% over the state-of-the-art FCN-RAU. Additionally, the UNet-RAU generated an F1-score of 80.97% on our Drone-Water-Front dataset. The second method — Dense Optical Flow based Water Detection (DOF-WD), detects water surfaces in videos of downward-facing cameras. This method utilizes downwash-generated ripples and natural texture features on a water surface to identify water in low and high altitudes, respectively. We empirically validated the performance of the DOF-WD method using our Drone-Water-Down dataset.

## 1. Introduction

Identification of water surfaces is crucial for Unmanned Ariel Vehicle (UAV)'s safety [1] and for applications such as detecting water retention areas [2], assisting in water quality management systems [3], and aiding ground vehicles to map upcoming terrains [1]. For a UAV itself, identifying safe flight paths and determining suitable landing grounds is crucial since its electronics can be damaged when in contact with water. The identification task can be challenging as water surfaces possess no exact shape or color, and the appearance of water changes due to various factors like viewing angle, surrounding scene, and weather conditions.

This research provides two separate approaches to identifying water surfaces based on the camera orientation, i.e. (1) when the camera is facing forward: to identify upcoming water surfaces, and (2) when it is facing downward: to identify water surfaces located under the UAV (see Fig. 1).

We used UNet-RAU, a unique image segmentation technique based on deep learning, for the task of identifying water surfaces using a front-facing camera of a UAV. To the best of our knowledge, deep learning-based segmentation techniques have not been tested for water puddle detection in drone imagery. When observed from a distance parallel to the ground, a water surface functions similarly to a mirror, reflecting the surrounding environment. This unique characteristic of water surfaces was utilized in this research to improve the performance of deep-learning models.

We carried out our evaluations on the Puddle-1000 [4] dataset and our own Drone-Water-Front (DWF) dataset. The puddle-1000 dataset consists of three sub-datasets: On-Road (ONR), Off-Road (OFR), and ONR-OFR combined (BOTH). It contains images of water puddles captured using a front-facing camera of a Unmanned Ground Vehicle (UGV). Our proposed UNet-RAU combination achieved F1-scores of 80.11% and 84.0%, On-Road, and Off-Road datasets respectively, outperforming the state-of-the-art FCN-RAU [4] in the water detection task.

In order to validate the UNet-RAU in the UAV context, we have curated a novel dataset named Drone-Water-Front (DWF), comprising annotated images of various water surfaces captured by a front-facing camera mounted on a drone. UNet-RAU achieved an F1-score of 80.97% on the DWF, supporting evidence for the suitability of UNet-RAU architecture in identifying water surfaces through drone imagery.

We proposed Dense Optical Flow based Water Detection (DOF-WD) techniques to identify water surfaces using a downward-facing camera of a UAV. We used two optical-flow-based methods, namely the Water-Ripples and the Water-Texture, both based on Farnebäck [5] Dense

☆ This paper has been recommended for acceptance by Zicheng Liu.

* Corresponding author.
*E-mail addresses:* harin.ncsl@gmail.com (H. Samaranayake), oshan.ivantha@gmail.com (O. Mudannayake), udi@ucsc.cmb.ac.lk (D. Perera), jpk@ucsc.cmb.ac.lk (P. Kumarasinghe), cps@ucsc.cmb.ac.lk (C. Suduwella), kasun@ucsc.cmb.ac.lk (K. De Zoysa), spw@ucsc.cmb.ac.lk (P. Wimalaratne).

https://doi.org/10.1016/j.jvcir.2023.103933
Received 21 September 2021; Received in revised form 16 May 2023; Accepted 3 September 2023
Available online 7 September 2023

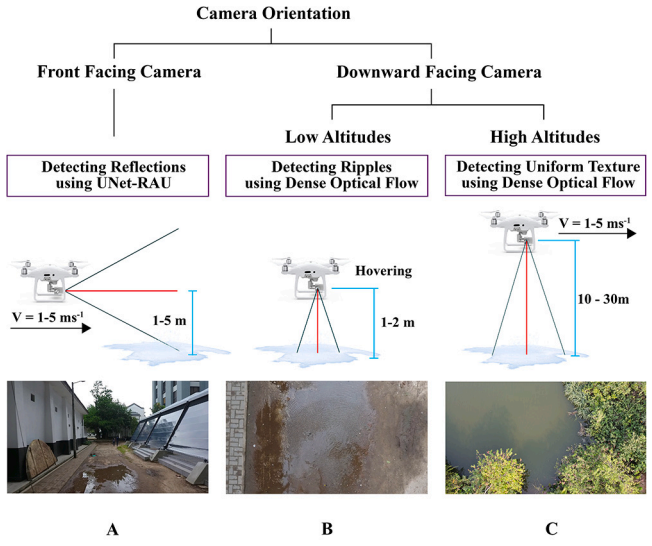


**Fig. 1.** UAV flight maneuvers performed during the data collection process. Image A: Capturing images using a front-facing camera. In this scenario, UAV was flown in the forward direction with a speed ranging from 1–5 $ms^{-1}$ while maintaining an altitude between 1–5 m. Image B: Capturing videos using a downward-facing camera while hovering. Here the UAV hovered in a low altitude range of 1–2 m. Image C: Capturing videos using a downward-facing camera while moving forward. Here the UAV was flown in the forward direction with a speed ranging from 1–5 $ms^{-1}$ while maintaining an altitude between 10–30 m.

Optical Flow algorithm. Using the proposed methods, we identified water surfaces in two scenarios: Firstly, when the drone hovers in low altitudes (1–2 m), the downwash of the drone generates ripples on the water surface. We identified these water surfaces using optical flow patterns of ripples with the Water-Ripples technique. Secondly, when the UAV moves in high altitudes (10–30 m), ripples are not getting generated on the water's surface. However, when the UAV moves forward, the non-water surfaces show an optical flow in the opposite direction of the UAV. Such movement is not visible on water surfaces due to their uniform texture as seen from higher altitudes. Based on these observations we identified stationary water surfaces in high-altitude videos using the Water-Texture approach.

To evaluate the effectiveness of DOF-WD techniques, we created the Drone-Water-Down (DWD) dataset, which consists of videos captured by a drone's downward-facing camera at different altitudes.

Our contribution through this work is as follows.

1. We investigate the combination of UNet [6] with the Reflection Attention Unit (RAU) for the purpose of detecting water surfaces situated in front of UAVs. As a result, we propose a novel architecture named UNet-RAU.
2. We propose Dense Optical Flow based Water Detection (DOF-WD), a methodology for identifying water surfaces located beneath UAVs. This approach utilizes optical flow patterns of ripples in lower altitudes and the uniform texture properties of still-water surfaces in higher altitudes.
3. We introduce a new dataset named Drone-Water (DW), which contains images and videos of water surfaces captured by drones at various altitudes and camera orientations. The Drone-Water-Front (DWF) sub-dataset includes annotated images captured using a drone's front-facing camera, while the Drone-Water-Down (DWD) sub-dataset consists of videos captured using a drone's downward-facing camera.

## 2. Related work

### 2.1. Water surface identification using ground vehicles

Water surface identification has been mainly conducted for ground vehicles [4,7–12] since water puddles generate risks for both vehicles and their passengers [4]. Water leakages can damage a vehicle's internal circuitry and cause accidents like skidding, crashes, and drowning risking passenger lives.

Matthies et al. [10] have compared four sensor types namely color images, Lidar, Short Wave Infrared (SWIR), and Mid Wave Thermal Infrared (MWIR) to detect water surfaces using UGVs. Sky conditions as clear, partly cloudy, and overcast were considered to detect water surfaces in color images. In this research, Lidar detected the depth of a water body, while SWIR detected whether the surface is water, snow, or ice. MWIR detected water at night.

Rankin et al. [12] visually detected water surfaces by localizing the sky's reflection pixels below the horizon level based on color similarity and visual features of the terrain.

Rankin and Matthies [11] utilized far-range reflections of the sky as a strong cue for water. At close ranges, the color coming out of a water body dominated sky reflections. In [11], a stereo-vision perception system on a UGV was used. This detector worked on the principle that the change in saturation to brightness ratio across a water body from the leading to trailing edge is uniform and distinct from other terrain types. Rankin and Matthies [11] method suited best for water bodies in wide open areas.

#### 2.1.1. Multi-cue approach

Daytime water detection technique which uses a fused multi-cue approach was presented by Rankin et al. [7] for autonomous off-road navigation. Color, texture, and reflections in stereo-range data were considered as cues to detect water. When using hue, saturation, and value (HSV) color space, water surfaces that reflect the sky consist of low saturation values and high brightness values, while deep bodies of water consist of high blue hue counts. Texture cue targeted water regions having low textures in grayscale images. In the range cue, range reflections in images obtained from a pair of color cameras were utilized. Finally, all the cues were fused together based on a rule base.

Mettes et al. [13] developed an algorithm that uses a hybrid descriptor based on both spatial and temporal local behavior of water surfaces in videos. Inputs were captured using a color camera sensor. Spatial behavior was characterized by using local binary pattern histograms while temporal behavior was quantified using temporal brightness signals of local patches. Findings were evaluated on two data sets, namely, Water database [13] and DynTex database [14].

Mettes et al. [15] segmented to detect water in videos by analyzing several motion properties of water. A new database named Video Water database was introduced by Mettes et al. [15].

#### 2.1.2. Polarization

The light coming from the sky which is non-polarized gets partially linearly polarized when reflected from a water surface. Nguyen et al. [8] detects water using 3D stereo cameras by considering the polarization of light off a water surface that changes from that of from sky, represented by the function of azimuth and reflection angles, accompanied with considerations to the color variation of light. Xie et al. [9] proposed a polarization-based water hazard detection technique for off-road navigation. By using 0°, 45°, and 90°polarization filters mounted in front of three cameras, three images are captured simultaneously to calculate the phase image followed by partial polarization. Afterward, an adaptive threshold-based segmentation algorithm and morphological filters have been applied to the phase image that identifies water as having phase-similar phases.

Polarization is another feature that was used by researchers to identify water surfaces [7–9]. The light coming from the sky is non-polarized. Once it gets reflected from a water surface, it becomes

partially linearly polarized. In the 3D approach of Nguyen et al. [8], they used polarized stereo cameras to capture the input. Xie et al. [9] proposed a polarization-based water hazard detection technique for off-road navigation.

### 2.2. Water surface identification using drones

Pombeiro et al. [1] proposed an active vision-based water detection model based on optical flow patterns in downward-looking camera images induced by the down-wash effect on the water surface during vertical and take-off and landing of UAV. This assists the UAV to identify suitable landing grounds.

Ridolfi and Manciola [16] proposed a technology that uses a sensing platform, which encompasses a drone and a camera to determine the water level. Images obtained from the sensing platform were analyzed using the Canny method to detect the edges of the water level. Ground control points were used as reference points. The water level is then retrieved from images and compared to a benchmark value obtained by a traditional device.

### 2.3. Water surface identification using satellites

Remote sensing literature provides information on orbital satellites used to investigate ground settings. Those satellites normally take inputs perpendicular to the ground [17–20].

However, water detection which uses space-borne remote sensing is challenging. Spatio-temporal scale issues, integration with in-situ-hydro-logical data and elevation data, and obscuration caused by clouds and vegetation [21] are among them. Several space-borne missions have addressed this situation with some special sensors such as Moderate Resolution Imaging Spectrometer (MODIS)[21], SPOT VEGETATION (VGT)[21], the Thematic Mapper (TM)[17], and Enhanced Thematic Mapper Plus (ETM+)[17].

### 2.4. Advancements in deep learning related water surface identification

Han et al. [4] developed a single image deep learning approach to detect water puddles using a Fully Convolutional Network (FCN). Researchers combined FCN with a special unit called Reflection Attention Unit (RAU) [4] to improve its performance. The assumption for RAU is that the reflections in an image lie in a vertical line. The FCN-8s was utilized as the FCN, and focal loss was used as the loss function. They introduced a new dataset named Puddle-1000 and used it for evaluations. The proposed approach outperforms Gaussian Mixture Model (GMM) with polarizers, and DeepLab-V2 [22]. FCN-8s integrated with focal loss function and 5 reflection attention units (FCN-8s-FL-5-RAU) was presented as their finest model.

Li et al. [23] introduced a method for pixel-level sea–land segmentation, which uses a unique Deep Convolutional Neural Networks (DCNN) named DeepUNet [23], which has been derived from UNet. A new dataset called the sea–land dataset [23] was introduced for evaluations. Researchers compared the findings of DeepUNet [23] with UNet and SegNet, where it concluded DeepUNet [23] performs better.

A method to detect water lines in images captured from a moving camera mounted on an autonomous boat was introduced by Steccanella et al. [24]. A FCN for obtaining a pixel-wise image segmentation was used. Experiments were conducted on the IntCatch Vision dataset [24] which contained images and videos with multiple floating obstacles on water.

Wang and Wang [25] used a Conditional Generative Adversarial Networks (cGAN) to tackle water hazard detection. An investigation was carried out as to where to place the proposed RAU by Han et al. [4] in different layers of cGAN. The best configuration was presented as cGAN-mRAU. Re-annotated Puddle-1000 dataset [25] was used for evaluations. cGAN outperformed on OFR subset and cGAN-mRAU outperformed on ONR and ONR and OFR combined (BOTH) subsets compared to FCN-8s-FL-5-RAU [4] in F1-measure.

## 3. Proposed method

We propose the following two methodologies for identifying the water surfaces.

1. UNet-RAU — Identifying water surfaces located in front of a UAV using image streams captured from a UAV's front-facing camera.
2. DOF-WD — Identifying water surfaces located underneath a UAV using videos captured from a UAV's downward-facing camera.

### 3.1. Identifying water surfaces located in front of the UAV

#### 3.1.1. UNet

Originally designed for biomedical image segmentation, the UNet [6] architecture was chosen for water surface identification due to its ability to effectively train with a smaller number of images and its faster training process [6], which is particularly advantageous in resource-constrained environments.

The architecture of the network consists of two main components: the concatenating path and the symmetric expanding path. The concatenating path focuses on capturing contextual information, while the symmetric expanding path is responsible for precise localization. The contracting path comprises two consecutive $3 \times 3$ unpadded convolutions, each followed by a Rectified Linear Unit (ReLU) activation function. Subsequently, a $2 \times 2$ max pooling operation with a stride of 2 is applied for downsampling. At each down-sampling step, the number of feature channels is doubled.

During the expansive path, the feature maps are upsampled using $2 \times 2$ convolutions, which reduces the number of feature channels by half. These upsampled feature maps are then concatenated with the corresponding cropped feature maps from the contracting path. Following this, two additional $3 \times 3$ convolutions are applied, followed by ReLU activation. In the final layer, a $1 \times 1$ convolution is employed to map each 64-component feature vector to the desired number of classes. The Adam optimizer with a learning rate of 1e-6 is utilized, and the binary cross-entropy function is employed as the loss function.

#### 3.1.2. UNet-RAU

We extended the original UNet architecture by incorporating RAU modules. In the downsampling path, RAUs were inserted after the two convolutional layers and before the max-pooling layer. Similarly, in the upsampling path, RAUs were positioned after the up-convolutional layer and before the concatenation with the contracting path. The resulting modified architecture, named UNet-8-RAU, is illustrated in Fig. 2.

### 3.2. Identifying water surfaces located underneath the UAV

Optical Flow (OP) characterizes the motion of objects across consecutive frame sequences, arising from the relative movement between the object and the camera. Within the context of OP, the intensity image, denoted as $I$, is defined as a function of spatial coordinates $(x, y)$ and time $(t)$. As the intensity at a specific location $(x, y)$ changes by $\Delta x$ and $\Delta y$ over a time interval $\Delta t$, a new intensity image, $I'$, emerges. A foundational assumption in this context is that the intensity of an object remains consistent across successive frames, as articulated in Equation 1.

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t) \tag{1}$$

In our two approaches Water-Ripples and Water-Texture, we utilized Farnebäck [5] Dense Optical Flow (DOF) method to identify water surfaces. It is a two-frame motion estimation algorithm based on polynomial expansion. In DOF, the flow vectors of all the pixels are calculated. Although this method is computationally slower, we selected it because of its high accuracy.

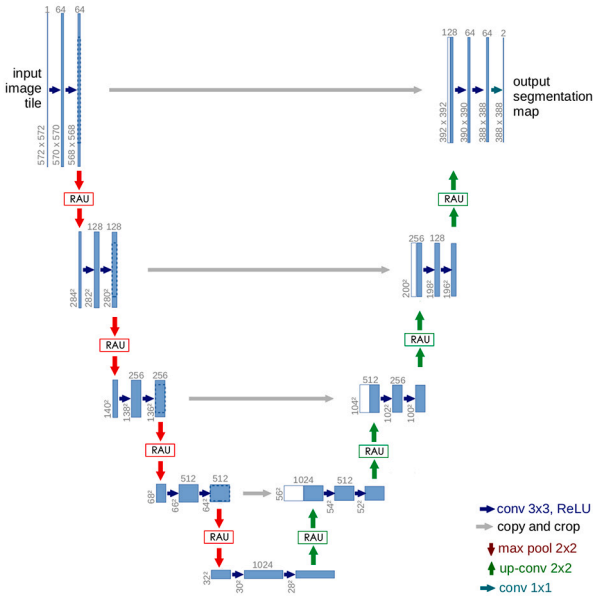


**Fig. 2.** Proposed UNet-8-RAU Architecture.

#### 3.2.1. In low altitudes

When a UAV hovers over a water surface, the down-wash generates ripples on the surface. These ripples show a circular-like outward motion from the pivot point.[1] This was observed during take-off and landing by Pombeiro et al. [1]. In their approach, a histogram of sparse optical flow orientation was generated and it was compared against a model histogram.

Inspired by their work, we tested out detecting water surfaces located under the UAV, using the ripples generated by the down-wash induced during hovering. An algorithm was proposed based on the assumption that on a water surface, the downwash creates ripples that move in all directions from the pivot point. In this method, the angle and magnitude of the flow vectors of each pixel were calculated. Using the calculated values, moving pixels were selected by thresholding the magnitude. Out of the moving pixels, their moving directions were categorized into 8 bins. If each bin exceeds a pixel count of 1% of frame pixels, then the frame is considered as a frame with water. This algorithm was named as Water-Ripples.

#### 3.2.2. In high altitudes

It was observed that when a UAV moves at a higher altitude, the ground seems to move in the opposite direction relative to the UAV. However, water surfaces do not show this movement due to their uniform texture, which is visible from higher altitudes.

An algorithm was proposed based on the assumption that when a UAV moves at a higher altitude, the ground moves in the opposite direction of the UAV while the water surface remains static. In this method, the angle and magnitude of the flow vectors of each pixel were calculated. Using the calculated values, moving pixels were selected by thresholding the magnitude. Moving pixels were labeled as the ground pixels while the rest were categorized as water pixels. We named this algorithm as Water-Texture.

## 4. Datasets

We introduced a new dataset named Drone-Water for the evaluation process of the proposed methodologies. It contains two sub-datasets named Front and Down based on the camera orientation which was used to capture inputs. We used a DJI Phantom-04 and a DJI Mavic Mini (Gen-1) drone in the data collection process.

When evaluating the performance of deep learning models, we used two datasets namely Puddle-1000 and Drone-Water-Front. It contains 119 annotated images (1920 $x$ 1080 resolution) of different water surfaces taken from a drone's front-facing camera. These images were extracted from videos taken from a drone, mounted with a front-facing camera that was flown at an altitude range between 1–5 m, with a speed of 1–5 $ms^{-1}$.

To evaluate the performance of the DOF-WD method, we used the Drone-Water-Down dataset. It contains videos of different surfaces like water, land, and water–land combined taken from a drone's downward-facing camera. Videos of this dataset are of 1920 $x$ 1080 resolution with a 30/60 fps. In the DOF-WD low altitude method, the drone was hovering at a static altitude between 1–2 m. In the DOF-WD high altitude method, the drone was flown forward at a speed of 1–5 $ms^{-1}$ while maintaining a static altitude between 10–30 m.

## 5. Implementation

Implementation, training, and testing of deep learning models and the evaluation of DOF technique was performed on a Ryzen 3900X, with an NVIDIA 3090 and 64 GB of RAM.

## 6. Experiments

### 6.1. Identifying water surfaces located in front of the UAV

We evaluated the performance of UNet combined with different numbers of RAU units. These models were named UNet-X-RAU models. Here X is the number of RAUs combined with the UNet, and X was assigned with the values of 4, and 8. We evaluated the UNet-X-RAU combinations using Puddle-1000 and our own Drone-Water-Front dataset. This experiment was conducted to analyze how the RAU units contribute to improving the efficiency UNet in water detection tasks.

We selected our best-performing UNet-X-RAU combination and compared the performance with UNet++ [26], DeepLabV3+ [27], and the state-of-the-art FCN-8s-FL-5-RAU [4] in water identification.

During the experimental phase, the UNet, UNet-X-RAU, UNet++, and DeepLabV3+ models were trained for 0.5 K epochs on 352 $x$ 640 image resolution. Models were implemented using Tensorflow 2.11, and data augmentation was not applied to the training data. The results for FCN-8s-FL-5-RAU in Table 2 were obtained from Han et al. [4] paper which was trained for 60 K iterations on 360 $x$ 640 image resolution.

### 6.2. Identifying water surfaces located underneath the UAV

#### 6.2.1. In low altitudes

We evaluated the proposed Water-Ripples method on low-altitude videos. These videos were taken by a drone, mounted with a downward-facing camera that hovered over water, land, and water–land combined surfaces. Considering the safety of the drone from water surfaces, 1 m was selected as the lower altitude limit. We gradually increased the altitude of the drone and evaluated the behavior of the proposed algorithm at each level.

#### 6.2.2. In high altitudes

Evaluation of the Water-Texture method was performed on high-altitude videos. Videos were captured using a drone, mounted with a downward-facing camera that was flown in the forward direction over water, land, and water–land combined surfaces while maintaining a static altitude. In compliance with local government restrictions of the experimental area, 30 m was selected as the upper altitude limit. We gradually decreased the flying altitude and evaluated the behavior of the proposed algorithm at each level. The speed of the drone was maintained between 1–5 $ms^{-1}$ to preserve the video clarity.

[1] point on the water surface vertically underneath the UAV

**Table 1**
Evaluation of UNet [6] when combined with different numbers of RAU units on ONR, OFR, and Drone-Water-Front(DWF) datasets.

| Dataset | Architecture | Resolution | Accuracy | Precision | Recall | F1-Measure |
|---|---|---|---|---|---|---|
| ONR | UNet | 352 × 640 | 98.88 | 51.34 | **85.9** | 64.27 |
| | UNet-4-RAU | | 99.17 | 75.63 | 79.33 | 77.44 |
| | UNet-8-RAU | | **99.22** | **78.2** | 82.11 | **80.11** |
| OFR | UNet | 352 × 640 | 99.34 | **93.84** | 72.96 | 82.09 |
| | UNet-4-RAU | | **99.35** | 91.2 | 76.8 | 83.37 |
| | UNet-8-RAU | | 99.34 | 88.98 | **79.56** | **84.0** |
| DWF | UNet | 352 × 640 | 99.14 | 75.98 | 77.95 | 76.95 |
| | UNet-4-RAU | | **99.23** | 79.41 | 81.47 | 80.43 |
| | UNet-8-RAU | | **99.23** | **79.44** | **82.57** | **80.97** |

## 7. Results and evaluation

### 7.1. Identifying water surfaces located in front of the UAV

A pixel-wise evaluation was performed on the model when generating results. We calculated four evaluation metrics namely, Accuracy, Precision, Recall, and F1-score (macro). Accuracy and F1-score were calculated for both water and non-water pixels while precision and recall were calculated only for water pixels. Precision indicates the number of correctly predicted water pixels out of all predicted water pixels. Recall indicates the number of correctly predicted water pixels out of all actual water pixels.

Table 1 shows the accuracy, precision, recall, and F1-scores of UNet combined with different number of RAU units: UNet-4-RAU, and UNet-8-RAU. Among the above UNet-X-RAU combinations, UNet-8-RAU performed best with F1-scores of 80.11%, and 84.0%on ONR, and OFR datasets respectively, further improving the performance of UNet. The UNet-8-RAU generated an F1-score of 80.97% on the Drone-Water-Front dataset, which was a 1% improvement over UNet. Results depict that UNet-8-RAU is most suitable for identifying water surfaces in UAV imagery.

Table 2 shows the results of UNet-8-RAU, UNet++ [26], DeepLabV3 + [27], and FCN-8s-FL-5-RAU [4] on On-Road, Off-Road, and Drone-Water-Down datasets. UNet-8-RAU outperformed ONR and OFR datasets in F1-score by 8% and 2% respectively, when compared with the state-of-the-art FCN-8s-FL-5-RAU [4] architecture.

Table 3 shows the cross-dataset validation,[2] of each model on Drone-Water-Front dataset against ONR OFR datasets.

### 7.2. Identifying water surfaces located underneath the UAV

#### 7.2.1. In low altitudes

During the experiments, it was observed that when the altitude of the UAV was increased gradually from 1 m, down-wash decreases resulting in a gradual decrease in the formation of visible water ripples. Depending on the drone models used in our experiments, the maximum altitude at which the drones generated clear visible water ripples was around 3 m. Based on our observations we selected 2 m as the upper altitude limit. Therefore the Water-Ripples algorithm was successful in identifying water surfaces in the 1–2 m altitude range. Fig. 3 shows the water detection by the proposed method when the altitude increased from 1 m to 5 m. Furthermore, we observed that if natural wind currents flow beneath the UAV, it will distraught the down-wash reaching the water surface. As a result ripples were not getting generated, making it difficult to be recognized as a water or non-water surface.

[2] Training on one dataset and testing on another dataset.

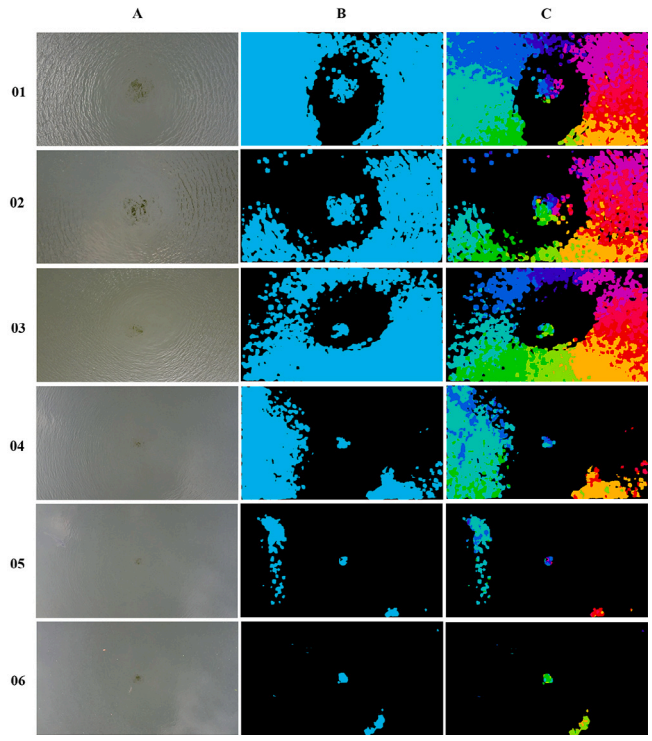


**Fig. 3.** Dense optical flow observed at different altitudes when a UAV hovering over a water surface. In this experiment, the altitude of the UAV was increased gradually from rows 1 to 6. Rows 1–3 contain frames captured at altitudes between 1–3 m, while rows 4–6 contain frames captured at altitudes between 4–5 m. Column A shows the original image, column B shows the detected water pixels (in blue color), and Column C shows water pixels based on the moving directions.

#### 7.2.2. In high altitudes

When we gradually decreased the altitude from 30 m to around 5 m altitude range, water ripples started appearing on the water's surface. That is at low altitudes, the downwash of the drone generates water ripples distorting the uniform texture of water. Based on our observations, the drone models used, and to eliminate unexpected ripples we selected 10 m as the lower altitude limit. Therefore the proposed Water-Texture algorithm was successful in identifying water surfaces in the 10–30 m altitude range. Fig. 4 indicates water detection by the proposed algorithm at different altitude ranges. Fig. 5 shows the results over different water–land surfaces.

## 8. Discussion

When identifying water surfaces using a front-facing camera of a UAV, the UNet-RAU combination is suitable when the water surface is still, which will reflect the surrounding environment clearly. In contrast, when natural wind currents flow over a water surface it distorts the surface reflections making the identification task difficult.

Based on our observations, detecting underlying water surfaces using the Water-Ripples technique depended on several factors, (1) the altitude at which it was being hovered, (2) the type of the UAV being used, and (3) the effect of natural wind currents on the water surface.

The Water-Texture method can be extended up to 5 m as the lower altitude limit but the drone should be flown at higher speeds over the water surface, i.e. the frame should be captured before the downwash reaches the water surface. Furthermore, we propose investigating the proposed Water-Texture method combined with a thermal camera, to reduce false detection of non-water uniform texture surfaces like carpeted roads in the video streams.

**Table 2**
Evaluation of UNet-8-RAU with UNet++ [26], DeepLabV3+ [27], and FCN-8s-FL-5-RAU [4] on ONR, OFR and Drone-Water-Front(DWF) datasets. (Results of FCN-8s-FL-5-RAU were taken from Han et al. [4] original work).

| Dataset | Architecture | Resolution | Accuracy | Precision | Recall | F1-Measure |
|---|---|---|---|---|---|---|
| ONR | UNet-8-RAU | 352 × 640 | 99.22 | **78.2** | **82.11** | **80.11** |
| | UNet++ | 352 × 640 | 98.85 | 67.49 | 68.96 | 68.22 |
| | DeepLabV3+ | 352 × 640 | 98.81 | 67.73 | 61.81 | 64.64 |
| | FCN-8s-FL-5-RAU [4] | 360 × 640 | **99.35** | 67.78 | 72.61 | 70.11 |
| OFR | UNet-8-RAU | 352 × 640 | 99.34 | **88.98** | **79.56** | **84.0** |
| | UNet++ | 352 × 640 | 98.83 | 67.53 | 65.37 | 66.42 |
| | DeepLabV3+ | 352 × 640 | 98.83 | 68.73 | 62.13 | 65.27 |
| | FCN-8s-FL-5-RAU [4] | 360 × 640 | **99.38** | 87.21 | 76.79 | 81.67 |
| DWF | UNet-8-RAU | 352 × 640 | **99.23** | **79.44** | **82.57** | **80.97** |
| | UNet++ | 352 × 640 | 98.8 | 67.53 | 62.42 | 64.87 |
| | DeepLabV3+ | 360 × 640 | 98.85 | 69.49 | 62.68 | 65.91 |

**Table 3**
Cross dataset validation of Drone-Water-Front(DWF) dataset with ONR, OFR datasets.

| Train | Test | Architecture | Resolution | Accuracy | Precision | Recall | F1-Measure |
|---|---|---|---|---|---|---|---|
| ONR | DWF | UNet | 352 × 640 | 87.83 | 7.41 | **29.82** | 11.87 |
| | | UNet-4-RAU | | **92.8** | 9.05 | 17.91 | 12.03 |
| | | UNet-8-RAU | | 92.76 | **10.32** | 21.15 | **13.85** |
| | | UNet++ | | 91.58 | 8.98 | 22.53 | 12.84 |
| | | DeepLabV3+ | | 92.02 | 9.04 | 20.95 | 12.63 |
| OFR | DWF | UNet | 352 × 640 | 91.17 | 7.77 | 20.26 | 11.16 |
| | | UNet-4-RAU | | **93.45** | **10.63** | 18.62 | 13.53 |
| | | UNet-8-RAU | | 92.6 | 10.21 | **21.66** | **13.88** |
| | | UNet++ | | 91.95 | 8.99 | 21.11 | 12.61 |
| | | DeepLabV3+ | | 92.14 | 9.11 | 20.69 | 12.65 |

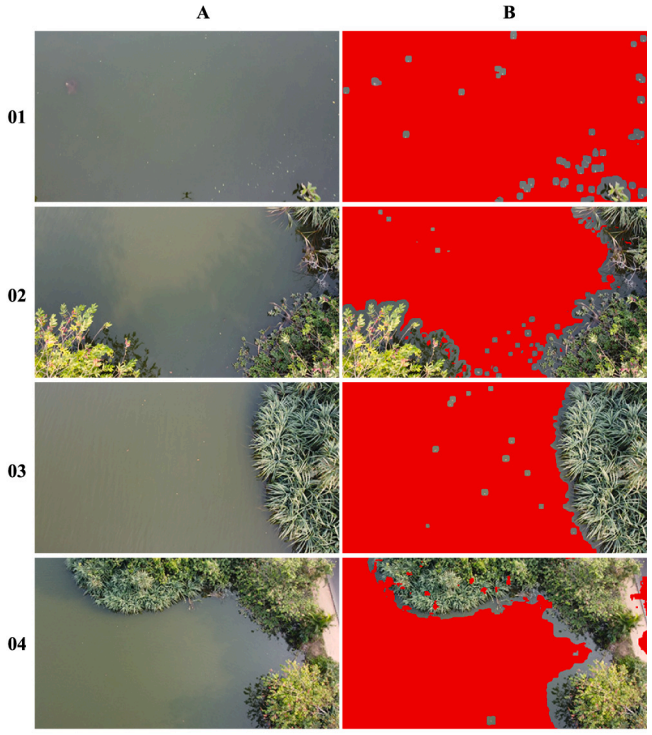


**Fig. 4.** Water detection at different altitude ranges using the Water-Texture method. The UAV was flown in the forward direction, with the camera oriented downwards. Rows 1 and 4 represent water detection in 2–5 m, 5–10 m, 10–20 m, and 20–30 m altitude ranges respectively. Column A represents the original image while column B represents the segmented image generated from the proposed algorithm. Detected water pixels are indicated in red color.

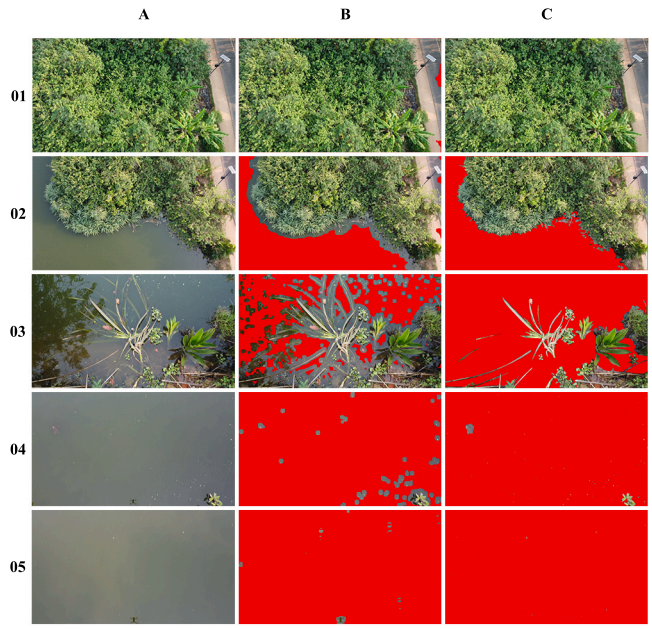


**Fig. 5.** Water detection over different terrains using the Water-Texture method. The UAV was flown in the forward direction, with the camera oriented downwards. Column A shows the original image while column B shows the segmented image generated by the proposed algorithm. Column C shows the ground truth image. Water pixels are shown in red color.

## 9. Conclusion

We investigated the UNet-RAU combination and Dense Optical Flow techniques in identifying water surfaces from front-facing and downward-facing camera views, respectively. When identifying water surfaces located in front of a UAV using a front-facing camera, reflections off a water surface were observed as a unique characteristic.

UNet-8-RAU was evaluated using the Puddle-1000 dataset [6] and achieved an F1-score of 80.11% on ONR, and 84.0% on OFR sub-datasets. It was an improvement of F1-scores by 8% on ONR and 2% on OFR datasets over the state-of-the-art FCN-8s-FL-5-RAU [6] combination. We extended the evaluation of UNet-8-RAU architecture on the Drone-Water-Front dataset and achieved an F1-score of 80.97%, indicating the applicability of the proposed architecture on UAV imagery.

When identifying water surfaces located underneath a UAV using videos captured by downward-facing cameras, we identified two unique characteristics of water: ripples and uniform texture, which can be utilized in low-altitude and high-altitude water identification tasks, respectively. At lower altitude ranges, ripple formation by the downwash of the UAV was analyzed. It was discovered that the Water-Ripples method was successful in detecting water surfaces underneath the UAV at an altitude range of 1–2 m. When detecting water surfaces at high altitude ranges, the Water-Texture method generated promising results for the altitude range of 10–30 m. The Water-Texture method utilized the uniform texture property visible on water surfaces at such high altitudes.

In conclusion, the proposed methodologies provide promising approaches to detect water surfaces in front of a UAV and underneath a UAV. The new Drone-Water dataset will be made publicly available and we believe that the findings of this research merit further investigations in this domain.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Data availability

Data will be made available on request.

## References


[1] R. Pombeiro, R. Mendonca, P. Rodrigues, F. Marques, A. Lourenco, E. Pinto, P. Santana, J. Barata, Water detection from downwash-induced optical flow for a multirotor UAV, in: OCEANS 2015 - MTS/IEEE Washington, (September) 2016.

[2] C. Suduwella, A. Amarasinghe, L. Niroshan, C. Elvitigala, K. De Zoysa, C. Keppetiyagama, Identifying mosquito breeding sites via drone images, in: DroNet 2017 - Proceedings of the 3rd Workshop on Micro Aerial Vehicle Networks, Systems, and Applications, co-located with MobiSys 2017, (June) 2017, pp. 27–30.

[3] C. Koparan, A.B. Koc, C.V. Privette, C.B. Sawyer, In situ water quality measurements using an unmanned aerial vehicle (UAV) system, Water 10 (3) (2018) 264.

[4] X. Han, C. Nguyen, S. You, J. Lu, Single Image Water Hazard Detection Using FCN with Reflection Attention Units: 15th European Conference, Munich, Germany, September 8–14, 2018, Proceedings, Part VI, 2018, pp. 105–121, http://dx.doi.org/10.1007/978-3-030-01231-1_7.

[5] G. Farnebäck, Two-frame motion estimation based on polynomial expansion, in: J. Bigun, T. Gustavsson (Eds.), Image Analysis, Springer Berlin Heidelberg, Berlin, Heidelberg, 2003, pp. 363–370.

[6] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9351, 2015, pp. 234–241.

[7] A.L. Rankin, L.H. Matthies, A. Huertas, Daytime water detection by fusing multiple cues for autonomous off-road navigation, in: Transformational Science and Technology for the Current and Future Force: (with CD-ROM), World Scientific, 2006, pp. 177–184.

[8] C.V. Nguyen, M. Milford, R. Mahony, 3D tracking of water hazards with polarized stereo cameras, in: Proceedings - IEEE International Conference on Robotics and Automation, 2017, pp. 5251–5257.

[9] B. Xie, Z. Xiang, H. Pan, J. Liu, Polarization-based water hazards detection for autosnomous off-road navigation, in: IEEE International Conference on Intelligent Robots and Systems, 2007, pp. 3186–3190.

[10] L.H. Matthies, P. Bellutta, M. McHenry, Detecting water hazards for autonomous off-road navigation, Unmanned Ground Veh. Technol. V 5083 (1) (2003) 231.

[11] A. Rankin, L. Matthies, Daytime water detection based on color variation, in: IEEE/RSJ 2010 International Conference on Intelligent Robots and Systems, IROS 2010 - Conference Proceedings, 2010, pp. 215–221.

[12] A.L. Rankin, L.H. Matthies, P. Bellutta, Daytime water detection based on sky reflections, in: Proceedings - IEEE International Conference on Robotics and Automation, 2011, pp. 5329–5336.

[13] P. Mettes, R.T. Tan, R. Veltkamp, On the segmentation and classification of water in videos, in: VISAPP 2014 - Proceedings of the 9th International Conference on Computer Vision Theory and Applications, 1, (November) 2014, pp. 283–292.

[14] R. Peteri, S. Fazekas, M. Huiskes, DynTex: A comprehensive database of dynamic textures, Pattern Recognit. Lett. 31 (2010) 1627–1632.

[15] P. Mettes, R.T. Tan, R.C. Veltkamp, Water detection through spatio-temporal invariant descriptors, Comput. Vis. Image Underst. 154 (2017) 182–191.

[16] E. Ridolfi, P. Manciola, Water level measurements from drones: A Pilot case study at a dam site, Water (Switzerland) 10 (3) (2018).

[17] L. Bertels, B. Smets, D. Wolfs, Dynamic water surface detection algorithm applied on PROBA-V multispectral data, Remote Sens. 8 (12) (2016).

[18] S. Elhassan, X. Wu, J.P. Walker, Standing Water Detection Using Radar, (December) 2013, pp. 1–6.

[19] D.E. Alsdorf, Geophysics: Tracking Fresh Water from Space, Science 301 (5639) (2003) 1491–1494.

[20] D.R. Lyzenga, Shallow-water bathymetry using combined lidar and passive multispectral scanner data, Int. J. Remote Sens. 6 (1) (1985) 115–125.

[21] J.F. Pekel, C. Vancutsem, L. Bastin, M. Clerici, E. Vanbogaert, E. Bartholomé, P. Defourny, A near real-time water surface detection method based on HSV transformation of MODIS multi-Spectral time series data, Remote Sens. Environ. 140 (2014) 704–716.

[22] L.C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs, IEEE Trans. Pattern Anal. Mach. Intell. 40 (4) (2018) 834–848.

[23] R. Li, W. Liu, L. Yang, S. Sun, W. Hu, F. Zhang, W. Li, DeepUNet: A Deep Fully Convolutional Network for Pixel-Level Sea-Land Segmentation, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 11 (11) (2018) 3954–3962.

[24] L. Steccanella, D. Bloisi, J. Blum, A. Farinelli, Deep learning waterline detection for low-cost autonomous boats, Adv. Intell. Syst. Comput. 867 (2019) 613–625.

[25] L. Wang, H. Wang, Water Hazard Detection Using Conditional Generative Adversarial Network with Mixture Reflection Attention Units, IEEE Access 7 (2019) 167497–167506.

[26] Z. Zhou, M.M.R. Siddiquee, N. Tajbakhsh, J. Liang, Unet++: A nested U-net architecture for medical image segmentation, 2018, arXiv:1807.10165.

[27] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-decoder with atrous separable convolution for semantic image segmentation, in: ECCV, 2018.