Supportive Information to Find Victims from Aerial Video in Search and Rescue Operation

Shigeru Kashihara
Graduate School of Science and Technology
Nara Institute of Science and Technology
Ikoma, Japan
shigeru@is.naist.jp

Muh. Arief Wicaksono
Department of Informatics Engineering
Hasanuddin University
Makassar, Indonesia
wicaksonoma15d@student.unhas.ac.id

Doudou Fall
Graduate School of Science and Technology
Nara Institute of Science and Technology
Ikoma, Japan
doudou-f@is.naist.jp

Muhammad Niswar
Department of Informatics Engineering
Hasanuddin University
Makassar, Indonesia
niswar@unhas.ac.id

***Abstract*—An unmanned aerial vehicle (UAV) is expected to be one of the powerful IoT tools in a search and rescue (SAR) operation. Notably, an aerial video would be most promising. However, in a SAR operation, a searching task via an aerial video compels laborious tasks such as high concentration to practitioners. In this paper, to reduce such a laborious task, we propose a method to provide supportive information, i.e., the number of persons, the time and thumbnails, for finding victims via an aerial video with an object detection algorithm. Also, to use it in a SAR operation, we need to meet the requirements for a dataset and lightweight process. Through the evaluation, we showed that we could get supportive information reasonably and displayed them on a GUI.**

***Index Terms*—Aerial Video, YOLO, Object Detection, Machine Learning, Search and Rescue, Unmanned Aerial Vehicle (UAV)**

I. Introduction

Unmanned aerial vehicles (UAVs) are expected to become one of the powerful IoT tools for supporting a search and rescue (SAR) operation because of possessing a camera-based function. As a bird's eye view that a UAV provides gives practitioners mass information, it leads to execute a smooth and safe operation. However, in a SAR operation, a searching task via an aerial video is hard for flight operators than we imagine due to following such reasons:

- Long concentration for executing safe flight and searching task
- Poor operational environment, e.g., small monitor such as a tablet and brightness of screen outdoor

Besides, after extracting aerial video from the UAV, even if another practitioner except for flight operators tries to find a victim from the video in a stable environment, it may take a long time to check the whole video. Although a UAV can take approximately 20-minutes video at present, the recording time increases with expanding of flight time by the development of UAV.

We have studied the utilization of a UAV in SAR operation so far [1], [2]. In the study, in addition to a camera-based function, we proposed to utilize an aerial Wi-Fi sensing function (Wi-SF) for supporting a SAR operation with a UAV. Since Wi-SF tries to find a missing person via Wi-Fi beacons broadcast from a Wi-Fi terminal, e.g., smartphone, which the person carries, Wi-SF can compensate for the limitations of the camera-based function, e.g., a missing person exists out of the shooting range. However, to obtain full efficiency, we need to employ both a camera-based function and Wi-SF. That is, practitioners need to check the whole of a video in any case. Therefore, it is required to find a missing person via an aerial video efficiently.

In the paper, to reduce such a laborious task, we propose a method to provide supportive information for finding a missing person with an object detection algorithm. However, at the spot in a SAR operation, as there is no high-performance server and high-speed Internet connection, a light-weight method to process aerial videos is required. In the approach, we first extract periodic photos from an aerial video and then try to detect persons by using an object detection algorithm, YOLO v3[1] [3]. Besides, we present the results, i.e., supportive information, on the GUI that we made in references [1], [2]. Through the paper, our contributions are two-fold: a) Showing supportive information extracted from an aerial video and b) reducing a searching time for detecting persons via an aerial video.

II. Related Work

Since a SAR operation with a UAV has the potential to upgrade the current SAR operation highly, practitioners desire to utilize a UAV in various SAR operations fully. So far, many approaches have been proposed to utilize UAVs in a SAR operation. The section introduces related work for them.

This work was supported by Promotion Program for Scientific Fire and Disaster Prevention Technologies, and JSPS KAKENHI Grant Number JP17K00124, Japan.

[1]YOLO: Real-Time Object Detection, https://pjreddie.com/darknet/yolo/

978-1-7281-2516-9/19/$31.00 ©2019 IEEE

Reference [4] focused on searching strategies. In a SAR operation, practitioners need to consider the fundamental parameters such as search algorithm, sensor quality, battery, and environmental hazards. Authors studied some search strategies to design the control strategy of multiple UAVs.

As another approach for a SAR operation, reference [5] proposed a victim detection based on voice recognition. Voice information from victims directly leads to find victims. However, as reported in [5], it is necessary to improve the accuracy of voice recognition.

Also, references [1], [2], [6], [7] studied a way to find victims based on wireless signals from their terminals, e.g., smartphone. Reference [6] proposed a method to locate possible victims from wireless signals of smartphones. In [7], authors presented a stochastic method to estimate the location of victims based on Wi-Fi signals. We also proposed a method to support finding victims based on Wi-Fi signals that a UAV captured in [1], [2]. In [8], we presented a way to display a victim's presence area based on the results of [1], [2]. These approaches are helpful to support a SAR operation in addition to aerial videos.

Then, references [9]–[12] presented the proposals based on a camera-based function. In [9], authors proposed a way to find victim by utilizing lidar and infrared depth camera in a GPS-denied environment. Reference [10] described human body detection using image processing. In the approach, skin color is first extracted in RGB and converted to hue-saturation-value (HSV), and then all noises in the image are removed. Also, reference [11] studied a computer vision approach using Histogram of Oriented Gradients (HOG) as the primary feature extractor and Support Vector Machine (SVM) as classifiers. The authors showed that the approach gives good performance for classifying frames as containing person and distinguishing between safe and dangerous landing sites. In reference [12], authors presented an approach to detect humans using thermal and color imagery. The above approaches utilize a camera-based function, but they do not focus on the reduction of a laborious task for checking a long aerial video.

III. Supportive Information Extracted from Aerial Video

To achieve a smooth and safe operation with a UAV, we need to reduce practitioners' laborious tasks. Notably, the paper focuses on finding victims from an aerial video. The searching task compels the long and careful concentration to practitioners. Then, the section proposes a method to provide supportive information for finding a missing person with an object detection algorithm. In Section III-A, we first introduce our previous work to implement the proposal. Section III-B describe the system overview of our proposal. In Section III-C, we explain the processes to extract supportive information from an aerial video.

A. Previous Work

As described in Section II, we have been studying the utilization of a UAV in a SAR operation [1], [2]. In our previous work, to compensate for the limitations of a camera-based function, we proposed an aerial Wi-Fi sensing function (Wi-SF) to capture Wi-Fi beacons broadcast from Wi-Fi terminals in the sky. Fig. 1 shows the UAV that installed Wi-SF. In our system, while a UAV records an aerial video, Wi-SF installed on the UAV captures Wi-Fi beacons in parallel. After the UAV returns to the home position, Wi-SF analyzes the Wi-Fi beacon captured. Then, our system provides practitioners the aerial video and the location information based on the Wi-Fi beacons on a map.

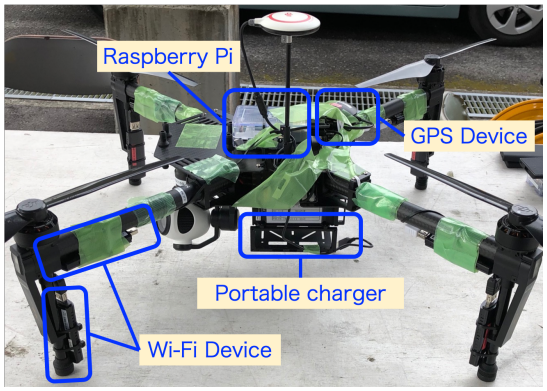

Fig. 1: UAV with aerial Wi-Fi sensing function

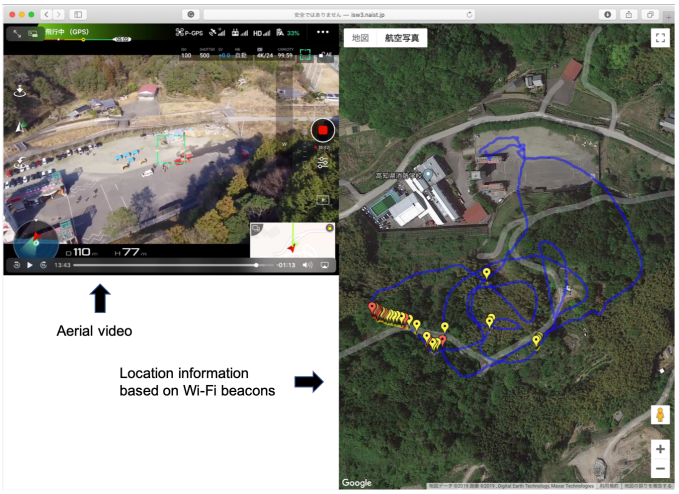

Fig. 2: GUI with aerial video and Wi-Fi beacon information

Fig. 2 presents the GUI of the system. The left window presents an aerial video, while the right one depicts the location information based on Wi-Fi beacons captured. The map helps practitioners to estimate the location of a victim from the information of the Wi-Fi beacons. Also, the aerial video provides mass information to them. However, if they cannot use the map, i.e., a UAV could not obtain any Wi-Fi beacons, they need to check the whole of the aerial video. As the task requires the long and careful concentration, the paper proposes to utilize an object detection algorithm to support the reduction of oversights for a victim by practitioners' eyes.

978-1-7281-2516-9/19/$31.00 ©2019 IEEE

B. System Overview

In the proposal, to reduce such a laborious task, we present more information, i.e., the number of persons in a screen and the time of the video, extracted from an aerial video. At present, a UAV can record approximately 20-minutes video. However, practitioners do not have sufficient time to check the aerial video from the beginning to the end because practitioners need to cope with an emergency promptly. Also, such a laborious task may lead to oversights for victims even if the video shot the victim.

We then propose a method to extract the number of persons and the video time from an aerial video and display the supportive information to reduce a laborious task for finding victims on the GUI that we implemented in the previous work. Fig. 3 shows the GUI implemented in the paper. The supportive information extracted from an aerial video is displayed at the left bottom on the GUI. It shows the video time and the number of persons in each scene. Also, thumbnails are provided. To implement it, we employ an object detection algorithm called YOLO to detect a person in an aerial video.

C. Supportive Information Extracted from Aerial Video

To extract supportive information from an aerial video, we need to satisfy the following two requirements.

1) Dataset for detecting persons from an aerial video
2) Lightweight processing for detecting persons

First of all, the dataset for detecting a person is the most important to extract supportive information by an object detection algorithm. To obtain accurate results, Section III-C1 describes the way to train dataset. Also, even if we get the high-accurate dataset, the high computational cost is necessary to process an aerial video. Then, in our method, we process screenshots extracted from an aerial video to make the process lighter, because the process of photos is lighter than one of a video. Section III-C2 explains the process of an aerial video to extract the supportive information.

1) Training a model: As far as we know, the dataset for finding persons from the sky is not released. We then train a dataset using the YOLO_mark[2] program. It labels 528 images of persons taken from a UAV that collected previously. We manually label each image as shown in Fig. 4 until all images in the dataset have been labeled. YOLO_mark automatically creates a txt format from the annotation file containing class id, x and y values and float values relative to the width and height of the image, which will be used for training. The annotation file is a particular annotation file used on the YOLO model. That is, it can be only used for training YOLO model and cannot be used to train other models.

After labeling all the dataset, we need to install YOLO, in this paper we use Alexey AB's[3] version of YOLO which has several advantages including improved detection by neural networks, improved performance, adding functions to calculate mAP, and fixing some codes, especially the code for storing videos that have been processed by YOLO.

[2]YOLO_mark, https://github.com/AlexeyAB/Yolo_mark

[3]Alexey AB YOLO Github, https://github.com/AlexeyAB/darknet

The environmental requirements needed to run YOLO effectively are:

- CUDA
- OPENCV
- cuDNN
- GPU
- GCC (for linux) or Microsoft Visual Studio (for windows)

Before training, we need to configure yolov3.cfg file provided from Alexey AB's Github. The configuration file is used to conduct training. We modified some codes, including:

1) max_batches = (number of classes * 2,000)
2) steps = (80% of max_batches), (90% of max_batches)
3) classes = 80 are replaced according to the number of classes we want to train
4) the value of filters = 255 is replaced by filters = (number of classes + 5) * 3

After all the preparations, we make training using Google Colaboratory[4], because to carry out the training, it required high hardware specifications. Google Colaboratory supports machine learning projects, but we can only use machines to run training every 12 hours.

To complete a model with one class, it takes at least 2,000 iterations; this depends on the number of max_batches set in the configuration file. During training, the resolution, width, and height of the image are changed every ten iterations to improve the accuracy of the model. Every 100 iterations, YOLO stores a weight backup file so that if an error occurs, we can restart training without having to repeat from the beginning. Training stops automatically after 2,000 iterations and generates a final weight file that is used to process object detection. After the training is done, it generates a weight file that is used to process object detection on images extracted from UAV videos. After the weights file is generated, we calculate the average precision of the trained model. If the average precision is less than 70%, then the new image is labeled and added to the data set, then the model is trained again. The workflow of the training steps can be seen in Fig. 5.

2) Extracting supportive information from aerial video: In our approach, to obtain supportive information, we employ snapshots extracted from an aerial video because the process of snapshots is lighter than one of a video. We first extract snapshots from the video at a certain interval by using the FFMPEG library. In this paper, as an instance, we obtain a snapshot per five seconds. The extracted images are stored in "video" folder, as shown in Fig. 6(a). Then, our process starts to execute to detect persons in all images. The images that have detected persons are labeled and stored in "output" folder (Fig. 6(b))

The process also counts how many people were in a screenshot. Concretely, it counts the number of words, i.e., "person" in the log, as shown in Fig.7, and it then saves the information into a text file with video time. In Fig.8, the file shows the video time and the number of persons at every

[4]Google Colaboratory, https://colab.research.google.com

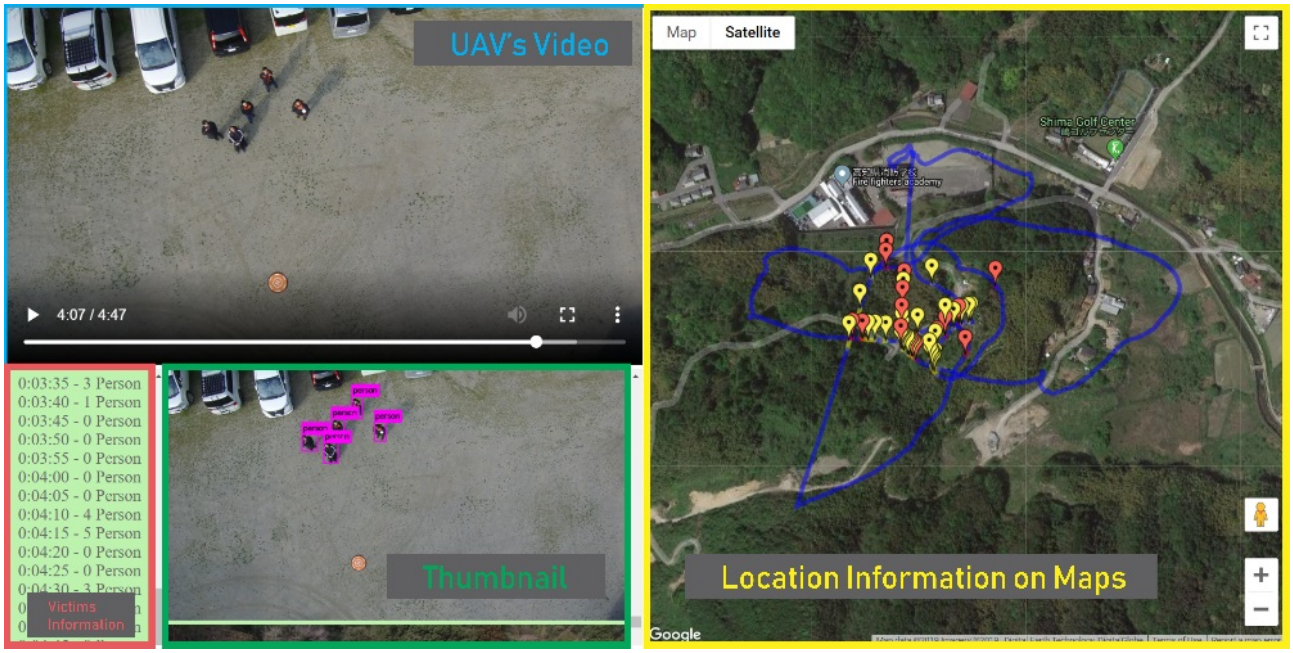

Fig. 3: Supportive information extracted from an aerial video

Fig. 4: Labeling images with YOLO_mark

five seconds. From the information, as we can estimate the video time that we may be able to detect a victim, we can reduce a laborious task for searching a victim. Two types of files are generated from this process. One is the images added bounding boxes to show the positions of persons and stored in "output" folder. Another is a text file including the number of persons and the video time. After the process, an aerial video, "output" folder, and "person.txt" file are moved to the local web server in order to show supportive information on the GUI.

IV. Evaluation

In a SAR operation, a lightweight process is required for detecting persons because, at the spot in a SAR operation, there is no high-performance server and high-speed Internet connection. To show the performance of our approach, the section evaluates the processing time when the object detection process is applied to images and videos.

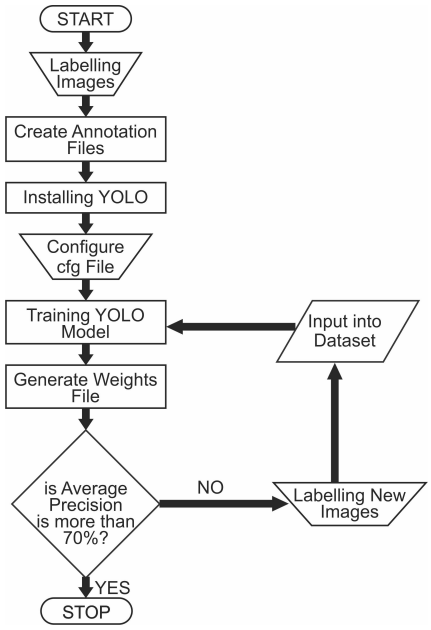

Fig. 5: Training steps

The experimental settings are as follows. We employ a PC with 8 GB of RAM and NVIDIA GTX 1050 Ti of GPU with 4 GB VRAM to run the object detection process. The video file used is a video length of 4 minutes 47 seconds with a resolution of 3840 x 2160 pixels. In the experiment, we evaluate images and video based on the file. Note that in our approach, we extract 58 snapshots from the aerial video because we employ five seconds as the extracting interval as one instance.

We explain an experimental result in the above experiments.

978-1-7281-2516-9/19/$31.00 ©2019 IEEE

(a) Extracted images in video folder

(b) Labelled images in output folder

Fig. 6: Images in "video" and "output" folders

```
Loading weights from yolov3-clone_final.weights...
 seen 64
Done!
 Used FMA & AVX2
 Used AVX
image.jpg: Predicted in 1247.820000 milli-seconds.
person: 49%
person: 64%
person: 97%
person: 97%
person: 78%
```

Fig. 7: Result of object detection process

As shown in Table I, the processing time for the video was 11202.655 seconds (approximately 186.7 minutes), while one for the 58 images was 517.394 seconds (approximately 8.6 minutes). From the result, we can see that the processing time for the 58 images is faster than one for the video. In actual, as the purpose of a SAR operation is to find victims, we do not need to process all frames of a video. That is, it is required to detect persons in different scenes. In the paper, we employ a five-seconds interval as one of the instances, but the scene-based approach is required to make it efficient.

TABLE I: Object Detection Processing Time

	Video	**58 snapshots**
Time (second)	11202.655	517.394

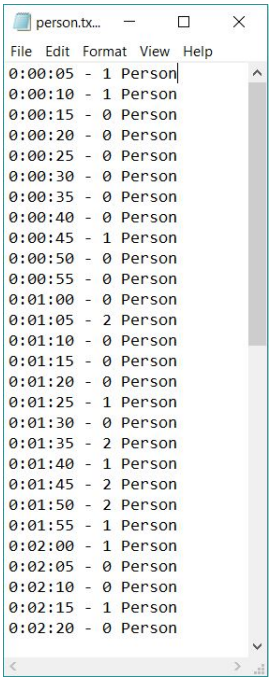

Fig. 8: Video time and the number of persons

V. Conclusion

Practitioners expect to utilize a UAV in a SAR operation fully. Although the UAV gives them mass information via an aerial video, the searching task via an aerial video is a laborious task for them. Also, it may lead to oversights for victims even if the video took the victims. To reduce such a laborious task, the paper presented a way to extract supportive information, i.e., the number of persons, the time, and thumbnails, from an aerial video and display them on a GUI. To use it in a SAR operation, we need to satisfy the requirements for a dataset and lightweight process. In the paper, we first explain the training of dataset for an aerial video. Then, to reduce the processing time for detecting a person from an aerial video, we proposed to use snapshots extracted from an aerial video. Through the comparison of the processing time for an aerial video and snapshots, we showed that we could get supportive information reasonably. In future work, we will make more accurate dataset and improve the GUI.

References

[1] S. Kashihara and K. Okamoto, "Toward Utilization of Drone for Gathering Disaster Information : Aerial Video and Wi-Fi Information," *The journal of the Institute of Image Electronics Engineers of Japan : visual computing, devices & communications*, vol. 45(3), pp. 397–404, 2016.

[2] S. Kashihara, A. Yamamoto, K. Matsuzaki, K. Miyazaki, T. Seki, G. Urakawa, M. Fukumoto, and C. Ohta, "Wi-SF: Aerial Wi-Fi Sensing Function for Enhancing Search and Rescue Operation," in *Proceedings of 2019 IEEE Global Humanitarian Technology Conference (GHTC)*, October 2019.

[3] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 779–788.

[4] S. Waharte and N. Trigoni, "Supporting search and rescue operations with uavs," in *2010 International Conference on Emerging Security Technologies*, September 2010, pp. 142–147.

[5] Y. Yamazaki, M. Tamaki, C. Premachandra, C. J. Perera, S. Sumathipala, and B. H. Sudantha, "Victim Detection Using UAV with On-board Voice Recognition System," in *2019 Third IEEE International Conference on Robotic Computing (IRC)*, 2019, pp. 555–559.

[6] Y. Ho, Y. Chen, and L. Chen, "Krypto: Assisting Search and Rescue Operations using Wi-Fi Signal with UAV," in *Proceedings of the First Workshop on Micro Aerial Vehicle Networks, Systems, and Applications for Civilian Use*, 2015, pp. 3–8.

[7] S. Ahn, G. Lee, and D. Han, "A Location Estimating Method of Buried Victims in Collapsing Area Using Wi-Fi Signals," in *Proceedings of the 2Nd International Conference on Vision, Image and Signal Processing*, ser. ICVISP 2018. New York, NY, USA: ACM, 2018, pp. 49:1—-49:5. [Online]. Available: http://doi.acm.org/10.1145/3271553.3275240

[8] M. Rosyidi, R. H. Puspita, S. Kashihara, D. Fall, and K. Ikeda, "A design of iot-based searching system for displaying victim's presence area," in *2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, vol. 02, July 2018, pp. 8–13.

[9] S. Lee, D. Har, and D. Kum, "Drone-Assisted Disaster Management: Finding Victims via Infrared Camera and Lidar Sensor Fusion," in *2016 3rd Asia-Pacific World Congress on Computer Science and Engineering (APWC on CSE)*, 2016, pp. 84–89.

[10] M. Zacharie, S. Fuji, and S. Minori, "Rapid Human Body Detection in Disaster Sites Using Image Processing from Unmanned Aerial Vehicle (UAV) Cameras," in *2018 International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS)*, vol. 3, 2018, pp. 230–235.

[11] F. N. Martins, M. De Groot, X. Stokkel, and M. A. Wiering, "Human Detection and Classification of Landing Sites for Search and Rescue Drones," in *Proceedings of European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, April 2016.

[12] P. Rudol and P. Doherty, "Human body detection and geolocalization for uav search and rescue missions using color and thermal imagery," in *2008 IEEE Aerospace Conference*, March 2008, pp. 1–8.