High Resolution Low Altitude 2D Map Reconstruction Using UAV

Tomoki Kosuga
School of Engineering
Taylor's University
Selangor, Malaysia
tomokikosuga@sd.taylors.edu.my

Swee King Phang
School of Engineering
Taylor's University
Selangor, Malaysia
sweeking.phang@taylors.edu.my

Ming Foong Soong
Department of Mechanical Engineering
Universiti Malaya
Kuala Lumpur, Malaysia
mfsoong@um.edu.my

Manee Sangaran Diagarajan
School of Engineering
Taylor's University
Selangor, Malaysia
manee.diagarajan@taylors.edu.my

N. Amutha Prabha
School of Electrical Engineering
Vellore Institute of Technology
Vellore, Tamil Nadu, India
amuthaprabha@vit.ac.in

2023 Innovations in Power and Advanced Computing Technologies (i-PACT) | 979-8-3503-2518-8/23/$31.00 ©2023 IEEE | DOI: 10.1109/I-PACT58649.2023.10434362

***Abstract*—In the contemporary era, with the ready availability of high-resolution and real-time satellite imagery, the field of low-altitude mapping remains an area yet to be resolved. In this study, the aim was set to reconstruct maps utilizing aerial images obtained from Unmanned Aerial Vehicles (UAVs) through the utilization of open-source tools. Furthermore, an analysis of the influence of flight altitude and speed was undertaken. Promising results were achieved as maps were successfully generated not only from vertically captured videos but also from overhead videos. Additionally, a comprehensive analysis was conducted to partially elucidate the impact of flight altitude on the quality of the captured images. Through this research, an expansion of mapping techniques was enabled by harnessing the capabilities of UAVs and open-source tools, thereby introducing a new dimension to cartography. The importance of considering flight parameters such as altitude and speed for optimizing map reconstruction from low-altitude sources was underscored by the findings. These outcomes carry significant potential for various applications, including environmental monitoring, disaster management, and urban planning.**

Keywords—UAV, Aerial Mapping, Map Reconstruction

I. Introduction

In recent years, satellite imagery has become readily available for visualizing landscapes from above. However, mapping at low altitudes remains an unresolved challenge. Given that Unmanned Aerial Vehicles (UAVs) typically fly at altitudes within 100 meters from the ground, they hold great potential in this field. Yet, with the demand for high-resolution and real-time map generation today, it appears that current UAV mapping systems still have room for improvement.

In this paper, we aim to develop a fast and high-resolution 2D scene reconstruction algorithm using aerial images downloaded from UAVs. The reconstructed 2D scenes are expected to achieve a resolution of at least 5 pixels per meter, as demonstrated by ground markers measuring 1 meter by 1 meter within the surveillance area.

Upon reviewing the current utilization of UAVs in the fields of geography and surveying, it became evident that the primary focus of discussion lies in 3D mapping [1]–[3]. Many literature sources describe the process of creating 3D maps by initially synthesizing images and generating 2D maps. However, this process seems to be overlooked, possibly due to the reliance on commercially available image processing software or software provided by UAV manufacturers for specific products [4]–[7]. This oversight can be seen as a problematic aspect from the perspective of promoting the widespread adoption of UAV utilization. The requirement for an initial investment in paid software raises the barrier to entry and limits flexibility in terms of adapting UAV usage according to specific situations. Additionally, relying on software provided only for certain products hinders the flexibility of utilizing UAVs for various purposes.

Therefore, set two main objectives for this project. The first is to reconstruct 2D maps from videos captured by UAVs using open-source tools. The second is to analyze the influence of UAV flight altitude and angles on the resulting resolution.

By addressing these objectives, we seek to overcome the existing limitations in UAV-based mapping and contribute to the advancement of this field. The potential applications of high-resolution and real-time 2D map reconstruction are vast, ranging from urban planning and environmental monitoring to disaster response and infrastructure management. Additionally, the development of open-source tools allows for wider accessibility and collaboration among researchers and practitioners, fostering further innovation in this area.

In this paper, we will discuss the methodology, experimental setup, results, and analysis related to achieving the project objectives. Furthermore, we will examine the limitations and propose potential solutions to improve the accuracy and efficiency of UAV-based 2D map reconstruction. Through this research, we aim to contribute to the advancement of UAV mapping technology and facilitate its adoption in various domains.

II. Research Methodology

The main focus of this research is to reconstruct a map from videos captured using UAVs. Reconstructing a map can be seen as creating a single image, so considering this, we have decided to adopt a method of extracting frames from the videos and stitching them together. Additionally, due to the nature of using UAVs as the filming equipment, it was also considered necessary to correct the image distortion caused

Authorized licensed use limited to: University of Malaya. Downloaded on February 27,2024 at 11:10:38 UTC from IEEE Xplore. Restrictions apply.

by the filming angle before stitching, as the camera angle may not be perpendicular to the ground. To summarize, the algorithm needs to be formulated to achieve the workflow shown in Fig. 1. Therefore, we individually examined each process, conducted necessary experiments, and finally combine them, as shown in the following subsections.

A. Frame Extraction

A more detailed approach to frame extraction involves reading the input video and saving the corresponding frames as images at regular intervals specified in seconds. This process continues until the video is finished. The flow of the algorithm is shown in Fig. 2.

First, video loading can be done using the VideoCapture method from the open-source library OpenCV. From there, we can retrieve the total number of frames in the video and the number of frames per interval for image retrieval. Once that is done, we iterate through all the frames using the read method, starting from 0 and continuing until the end of the video. We save the frames as images only when the frame number is a multiple of the specified interval.

By following this process, it becomes possible to accurately retrieve frames from the video at the specified intervals.

B. Distortion Correction

We assume a general scenario where the vertical position relative to the ground is considered 0 degrees, and the angle can vary up to a maximum of 89 degrees facing forward. We do not consider special camera angles, such as diagonal or sideways orientations. In this situation, the image will exhibit distortion where the foreground appears larger and the background appears smaller. To address this, we use OpenCV to compress the bottom edge of the image, simultaneously equalizing the scale. Compression of the bottom edge is achieved by performing vertex manipulation using OpenCV's getPerspectiveTransform method.

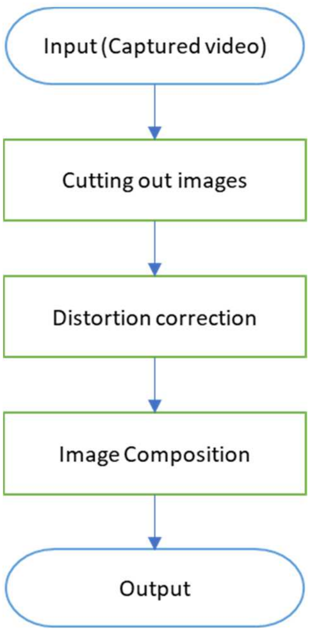


Fig. 1. The flow of algorithm to generate map images from video captured by UAV.

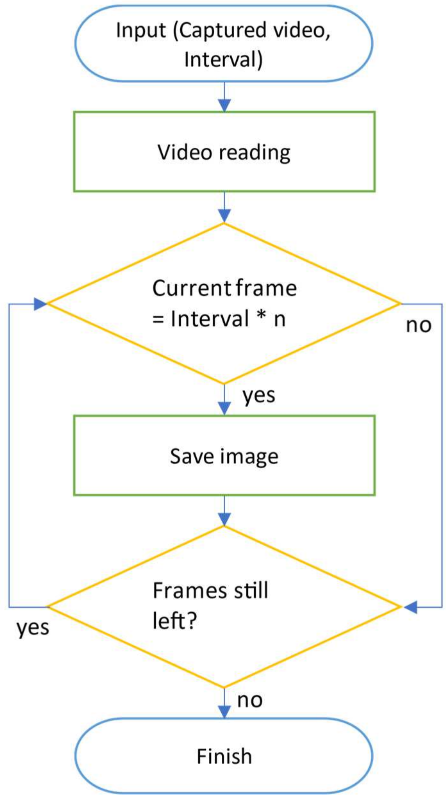


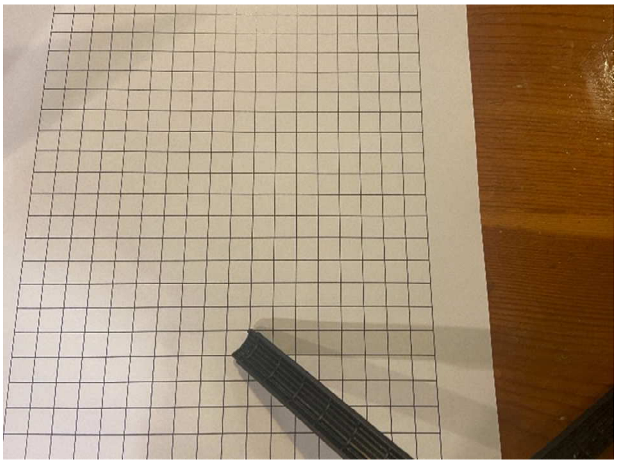
Fig. 2. Algorithm flow to obtain images at regular intervals of frames.

However, it is anticipated that the required compression ratio is not constant, and it needs to be automatically calculated. To accomplish this, we will manually adjust and analyze the captured images of calibration paper while changing the angle using a tripod. By obtaining and analyzing the necessary compression ratios, we can ensure that the distortion is corrected. The correction of distortion is confirmed by acquiring the coordinates of two reference points aligned vertically in the image and checking if their x-axis values match. The use of calibration paper serves the purpose of visually identifying the distortion through vertical lines and also to provide a reference for obtaining coordinates using OpenCV's setMouseCallback method. The calibration result is shown in Fig. 3.

It is important to note is that to avoid being affected by the number of pixels in the image, the compression should be done proportionally rather than using a constant value. Additionally, considering UAV-based photography, it is necessary to investigate whether distortion is influenced by not only the angle but also the height. Therefore, similar experiments and analyses should be conducted by changing the height.

By doing so, it becomes possible to automatically calculate the required compression rate based on the parameters of height and angle and use it to correct the distortion.

Authorized licensed use limited to: University of Malaya. Downloaded on February 27,2024 at 11:10:38 UTC from IEEE Xplore. Restrictions apply.



(a)

(b)

Fig. 3. (a) Captured image; and (b) Same image after distortion correction.

C. Image Composition

Image stitching is achieved using the Stitcher method in OpenCV. Image stitching is the process of finding matching key points and overlaying them. However, it is anticipated that synthesizing a large number of images at once could be challenging due to device memory limitations. Therefore, three methods have been devised, and a comparison will be conducted for each of them.

The first method involves stitching all the images at once. The second method involves stitching only the first 20 images from the image list and setting the result as the new first image, and then repeating the process of stitching the first 20 images from the updated list. The last method involves dividing the list into separate lists of 20 images each, and repeatedly stitching the results of each sub list, also consisting of 20 images.

In addition, for the image stitching experiments, a list of frames extracted from the video is required. Therefore, the experiments will be conducted simultaneously with the frame extraction process.

D. Full Process with UAV Flight

After conducting empirical experiments for the three phases: frame extraction, distortion correction, and image stitching. Aerial videos will be captured using the UAV to confirm the proper functioning of the entire algorithm and to perform analysis and possible improvement.

Based on the previous considerations, the overall flow of the system at this stage can be seen in Fig. 4. There are two experiments to be conducted using this setup. The first is to generate images from videos captured in a vertical orientation relative to the ground. The second is to generate images from videos captured at an angle relative to the ground. Both experiments require actual video captured from the airborne UAV. The successful generation of these images will be extracted for further analysis, including the determination of necessary overlap ratios. The experiments will be conducted at selected locations within Taylor's University Malaysia.

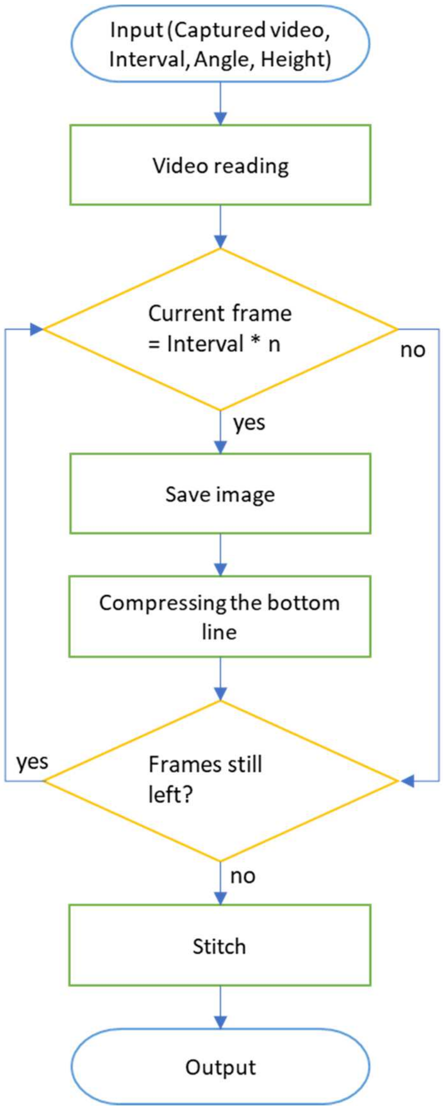


Fig. 4. Flow of the overall algorithm to stitch all images at once.

Authorized licensed use limited to: University of Malaya. Downloaded on February 27,2024 at 11:10:38 UTC from IEEE Xplore. Restrictions apply.

E. Accuracy Enhancement

Finally, we investigate the influence of parameters by running the formulated algorithm multiple times on the same video to examine the conditions for optimal generation. Detailed results will be presented in the Section III, but as a premise, it has been determined that image composition will be performed by combining 20 frames at a time and further synthesizing the results in batches of 20 frames.

Regarding the experimental setup, we attempted 10 patterns of frame extraction intervals ranging from 0.1 seconds to 1 second on the same video. Additionally, we conduct image composition using every 10 frames, 20 frames, and 30 frames to determine the optimal interval and number of compositions. To account for uncertainties in a single run, we perform 10 repetitions for each pattern, resulting in a total of 300 trials for one video.

In our work, the overlap ratio, η, for each frame extraction interval was obtained with the following equations:

$$x_{FRAME} = v_{UAV} \times t_{IMG} \quad (1)$$

$$x_{CUTOFF} = v_{UAV} \times \Delta t_{IMG} \quad (2)$$

$$\eta = \left(1 - \frac{x_{CUTOFF}}{x_{FRAME}}\right) \quad (3)$$

where x_{FRAME} and x_{CUTOFF} are UAV distance travelled on a single frame and the cutoff distance by the algorithm respectively; v_{UAV} is the actual flying speed of UAV, which can be controlled by our UAV autopilot algorithm; and t_{IMG} is the time taken to fly from one end of the image to another, while Δt_{IMG} is the interval time (update time) of our algorithm.

III. Results and Discussion

A. Distortion Correction

An algorithm as described in Section II was developed to compress the base of the images by a specified ratio. In this algorithm, a 90-degree counterclockwise rotation is also performed due to different axes between the captured images and the UAV. The main functionality is to receive an image and a compression ratio as input, and then compress the base of the image accordingly. Additionally, it is possible to obtain the coordinates of selected points using the cursor on the display during the result verification, and the alignment of their x-axis confirms the correction of image distortion.

For the experiment, images were captured at 5 different camera angles: 10, 20, 30, 40, and 50 degrees, combined with 3 different heights: 22.7 cm, 35.6 cm, and 44.1 cm. All captured images were manually adjusted. The results are presented in the Table I, and is plotted in Fig. 5.

Fig. 5 reveals several insights. Firstly, it shows that as the angle values increase, the compression ratio also increases. Moreover, this increase follows a linear proportionality. Secondly, it is evident that there is no significant difference in the trend based on height. This indicates that the only necessary parameter for correcting distortion is the angle.

TABLE I. THE COMPRESSION RATIO OF THE BASE TO ACHIEVE UNIFORM SCALING OF IMAGES

Height (cm)	Angle (degree)	Compress Ratio, η
22.7	11	0.088
	20	0.151
	31	0.223
	40	0.295
	49	0.355
35.6	10	0.085
	19	0.145
	31	0.227
	40	0.293
	49	0.355
44.1	10	0.086
	20	0.153
	30	0.210
	40	0.286
	49	0.357

B. Frame Extraction and Image Composition

Based on the algorithm described in Section II, the algorithm was coded to generate a list of extracted images. This image list is then passed to the image stitching phase. For the image stitching code, the images are stitched according to the three methods as discussed. Tests were performed for each case. To conduct the tests, sample of high-quality videos (1080p) that was captured using UAVs that were widely available on video-sharing platforms were used. This approach allows us to focus solely on verifying whether the code functions correctly and comparing the differences resulting from each method.

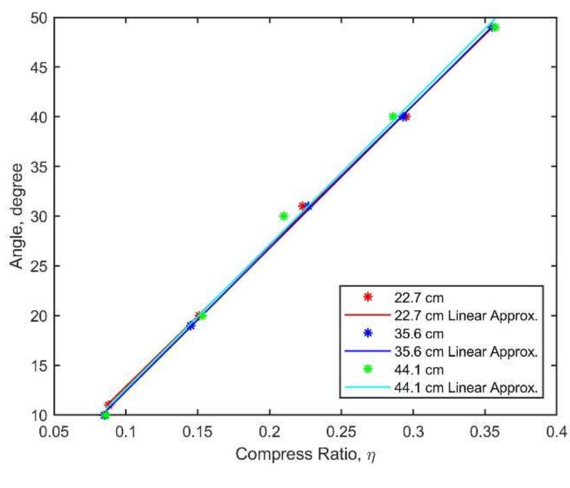


Fig. 5. Compression Ratios of the base against camera angles.

Authorized licensed use limited to: University of Malaya. Downloaded on February 27,2024 at 11:10:38 UTC from IEEE Xplore. Restrictions apply.

(a)

(b)

(c)

Fig. 6. (a) Results for the case of stitching all images at once; (b) stitching 20 images at a time from the front; and (c) stitching 20 images at a time and then further stitching the results in sets of 20.

Based on these experiments conducted on the same video using the three methods, the results are shown in Fig. 6.

Firstly, it is evident that the second method results in blur outputs. Furthermore, the blurriness is more pronounced on the left side, which can be attributed to the degradation of the image caused by repeated stitching. While the first and third results may initially appear similar, a clear difference emerges when increasing the number of images to be stitched. When conducting experiments simulating longer duration videos by reducing the frame extraction interval, the first method frequently failed due to insufficient memory, whereas the third method consistently produced successful results.

Therefore, for this research, the approach of stitching 20 images at a time and further stitching the results in sets of 20 has been chosen as the preferred method for image stitching.

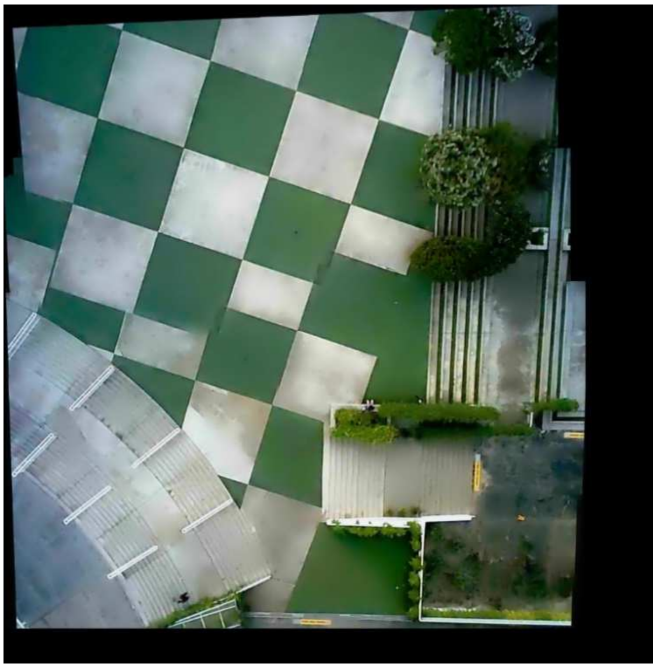
C. Actual UAV Experimental Footage

The construction of UAV for video recording purposes has been widely documented and therefore will not be included in this paper. The UAV used for this experiment was built in-house at Taylor's University, interested readers can refer to our previously published work [8]–[11].

There are two main changes from the initial predictions based on the research methodology. First, the height parameter was removed from the input parameters. Second, the image synthesis method was modified. Additionally, the code was transformed into a web application in streamlit library, allowing interested readers to try out the program at https://haniwa828-fyp-main-famx4w.streamlit.app.

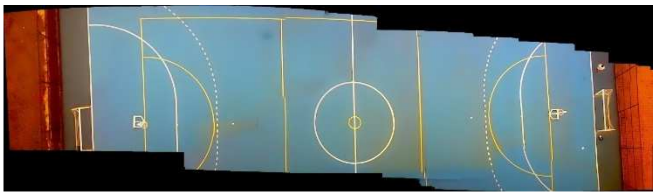
Using the customized program, we were able to successfully generate images from videos captured at various part of the Taylor's University Malaysia Campus. The image stitching results for various location from our work can be seen in Fig. 7. From the obtained results, although there are some minor losses, the generated images generally follow the flight path, indicating that the program is functioning correctly.

Next, for the experiment of videos with angled shots, the results obtained is shown in Fig. 8.



(a)

(b)

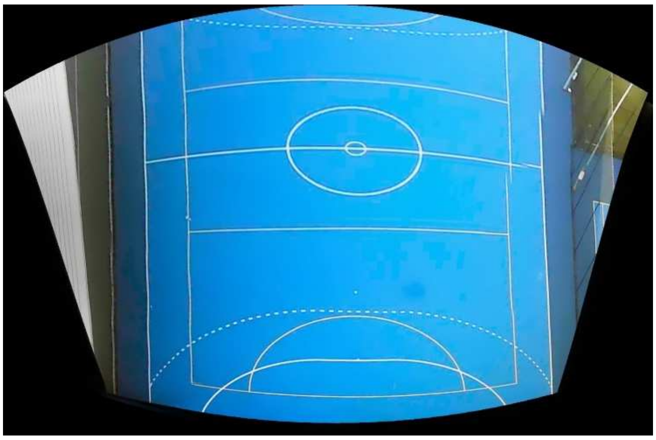


(c)

Fig. 7. Stitched images generated from video from (a) Courtyard; (b) Car park; and (c) Futsal court; all at Taylor's University Malaysia.

Authorized licensed use limited to: University of Malaya. Downloaded on February 27,2024 at 11:10:38 UTC from IEEE Xplore. Restrictions apply.

(a)



(b)

Fig. 8. Stitched images generated from video with camera angled (a) 50 degree relative to the horizontal plane; (b) 25 degree relative to the horizontal plane; at Taylor's University Malaysia.

The results were classified into four categories: Normal, Misclassification, Uncertain, and Failed. The criteria for each category are as follows: Normal indicates almost perfect reproduction without any distortion, Misclassification represents a reproduction rate of at least 80% with slight noticeable distortions, Uncertain indicates a situation where no errors occurred but a significant portion of the image is missing or has severe distortion, and Failed refers to cases where the generation itself failed. Success rate was only calculated for the Normal and Misclassification categories.

In conclusion, no clear trend could be identified regarding the influence of the interval and the number of images synthesised at a time on the generation, but some interesting points could be found.

1. There is a tendency for success to be concentrated in specific areas.
2. It is noteworthy that videos captured at a higher speed exhibit the highest success rate when the overlap rate is at least 73%. Although in this experiment, the frame extraction interval was limited to a maximum of 1 second, but it is possible that when reducing the overlap rate in other videos, different areas of success may be discovered.
3. It is particularly intriguing that increasing the number of frames synthesized at once does not necessarily result in improved accuracy.

The results obtained in this section has clearly shown that the algorithm and the implementation of the work on UAV has been successful, up to a certain degree of accuracy. However, certain challenges and issues arise during the implementation of the work.

The first issue is the low success rate, as can be seen from some of the videos. To address this issue, two potential improvements can be considered:

1. Enhancing the quality of video and image synthesis: The quality of the video encompasses not only image resolution but also the stability of the footage. In the UAV used for the experiment, the flight was not consistently stable, causing slight vibrations throughout the recording. Additionally, there were intermittent interruptions in the footage such as image freezing. These factors make it challenging to establish correspondence between features in consecutive frames. Regarding the quality of image synthesis, although OpenCV is an excellent open-source library, it has its limitations in terms of accuracy. It could be beneficial to explore the possibility to use alternative tools to achieve stable image generation, even with challenging footage.
2. Improvement of the algorithm: Through generating over 1,000 iterations in various patterns, it is discovered that there are specific areas in each video where success is more likely to occur. Therefore, by carefully identifying these areas, we can significantly reduce the number of attempts required to achieve success.

The next issue concerns the input of camera angles. As shown in Fig. 8, the camera angle was set to 50 and 25 degrees. However, successful results were obtained only when the angle was set to 25 degrees. Although a few successful cases were observed besides the ones shown in the images, they were all within the range of 25-35 degrees. When inputting 50 degrees, the generated images either failed or exhibited significant flaws. The cause of this issue could be the tilting of the UAV during movement. It was an oversight during the consideration phase, but UAVs achieve forward movement by tilting their bodies while pushing air downward for lift. As a result, the camera angle is effectively reduced when moving forward. This explains why the 25-degree input was successful while the 50-degree input failed. Considering that UAVs can vary in flying speed and direction, it may be challenging to account for this factor. However, improving accuracy by addressing this issue is undoubtedly possible.

The third issue is the limitations of the web application. Our experiments primarily involved opening the same code as the web application on a local server and conducting tests there. However, when we deployed it as an actual web application on streamlit, we encountered memory limitations. The allocated memory in the streamlit-hosted site was limited, and even with the most accurate and least memory-intensive image synthesis method, exceeding the limitation resulted in failed generation. To avoid this, it would be

Authorized licensed use limited to: University of Malaya. Downloaded on February 27,2024 at 11:10:38 UTC from IEEE Xplore. Restrictions apply.

necessary to either create a standalone application that performs the processing on the user's device or optimize the synthesis process to make it more efficient.

IV. CONCLUSION

The main objective of the research work - to improve the generation of 2D maps from videos taken by UAVs using open source - has been achieved. The fact that the camera angle is virtually unlimited and that the code created was made available as a web application, allowing it to be used by many people without depending on the environment or model, are also a major achievement. However, there are some clear challenges in the current work, such as development with open source and free services is not sufficient in terms of accuracy due to performance limitations. Therefore, we look forward to the development of new methods to solve these problems.

REFERENCES

[1] Lauterbach, H. A., Koch, C. B., Hess, R., Eck, D., Schilling, K., & Nüchter, A., 2019. The Eins3D project—Instantaneous UAV-based 3D mapping for Search and Rescue applications. *In 2019 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR),* pp. 1-6.

[2] Colorado, J., Mondragon, I., Rodriguez, J., & Castiblanco, C., 2015. Geo-mapping and visual stitching to support landmine detection using a low-cost UAV. *International Journal of Advanced Robotic Systems*, **12(9)**, p. 125.

[3] Goh, J. N., Phang, S. K., & Chew, W. J., 2021. Real-time and automatic map stitching through aerial images from UAV. *In Journal of Physics: Conference Series,* **2120(1)**, p. 012025.

[4] Ren, M., Li, J., Song, L., Li, H., & Xu, T., 2022. MLP-based efficient stitching method for UAV images. *IEEE Geoscience and Remote Sensing Letters*, **19**, pp. 1-5.

[5] Bang, S., Kim, H., & Kim, H., 2017. Vision-based 2D map generation for monitoring construction sites using UAV Videos. *In ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction*, **34**.

[6] Song, D. Y., Um, G. M., Lee, H. K., & Cho, D., 2021. End-to-end image stitching network via multi-homography estimation. *IEEE Signal Processing Letters,* **28**, pp. 763-767.

[7] Shahid, I. G., Phang, S. K., & Chew, W. J., ,2023. Fast 3D Mapping Solution with UAV. *In Journal of Physics: Conference Series,* **2523(1)**, p. 012019.

[8] Tai, J. J., Phang, S. K., & Wong, F. Y. M., 2022. COAA*—An Optimized Obstacle Avoidance and Navigational Algorithm for UAVs Operating in Partially Observable 2D Environments. *Unmanned Systems*, **10(2)**, pp. 159-174.

[9] Ho, J. C., Phang, S. K., & Mun, H. K., 2021. 2-D UAV navigation solution with LIDAR sensor under GPS-denied environment. *In Journal of Physics: Conference Series,* **2120(1)**, p. 012026.

[10] Ng, Z. Y., & Phang, S. K., 2020. Development of simultaneous localization and mapping algorithm using optical sensor for multi-rotor UAV. *In AIP Conference Proceedings*, **2233(1),** p. 030007.

[11] Tai, J. J., Phang, S. K., & Wong, Y. M. F., 2020. Optimized autonomous UAV design with obstacle avoidance capability. *In AIP Conference Proceedings,* **2233(1)**, p. 020026.

Authorized licensed use limited to: University of Malaya. Downloaded on February 27,2024 at 11:10:38 UTC from IEEE Xplore. Restrictions apply.