AIAA SCITECH 2024 Forum, 8-12 January 2024, Orlando, USA. Paper number AIAA 2024-1300
DOI:10.2514/6.2024-1300

Autonomous Navigation with Taxiway Lines Identification using Camera Vision and Airport Representations

Quentin Delezenne* Ivan Petrunin † and Zhengjia Xu ‡
Cranfield University, Cranfield, Bedfordshire, MK43 0AL

Jonathan Neptune § and Timothy Bleakley¶
General Atomics Aeronautical Systems Inc, Poway, CA, United States

With increasing demands of unmanned aerial vehicle (UAV) operations envisioned for the future of aviation, the number of pilots will be much lower than the number of drones, necessitating an increased level of autonomy in drones to alleviate workload. Autonomous UAV taxiing enables autonomy to move on the ground, specifically from the gate to the runway and vice versa without human intervention. This study presents a lightweight vision-based autonomous taxiway navigation system, exploring the fusion of camera vision feed under the nose and airport map data to offer guidance and navigation. A sliding window mechanism is applied in centreline identification to detect line divergence. Centreline representations including divergence, direction and heading are cross-referenced with airport database for localisation and generating navigation solutions. A simple proportional integral derivative (PID) controller is developed over aircraft dynamic models aligned with Eagle Dynamic's Digital Combat Simulator to demonstrate the centreline following function. The overall system performance is assessed through simulations, encompassing individual functionality performance tests including centreline extraction test, line matching test, line-to-follow test, generalisation capability test, and computational complexity test. The performance evaluations indicate the promising potential of camera visions in enabling autonomous UAV taxiing with 71% successful rate of detecting correct lines to follow and the remaining 29% as background. The proposed system also suggests a high generalisation capability of more than 67% success rate when testing over other paths. The source code of this proposition is open-sourced at https://github.com/DelQuentin/TaxiEye.

I. Introduction

Ongoing development of unmanned aerial vehicle (UAV) capabilities points to a promising future dominated by an increased number of autonomous UAV operations in aviation. Surge UAV applications in aerial surveillance, transportation, and mapping are anticipated to surpass the number of crew pilots. The importance of taxiway operations efficiency and safety in the context of large airports has been identified by the Single European Sky airport traffic management (ATM) Research consortium and according to the ATM Master Plan [1], delivered advanced solutions concerning taxiway operations. Therefore, aircraft automation aiming to reduce crew number during operation becomes prominent which anticipates assuring safety and efficiency, and providing possible guidance to aircraft crews while enhancing user safety.

Autonomous taxiing enables the autonomy of an aircraft to move on the ground, specifically from the gate to the runway and vice versa without human intervention, where autonomous navigation is typically applied. In the standard taxiway operation procedures, described by aviation authorities such as federal aviation administration (FAA) [2], the target path an aircraft will follow from one point to another is given by an air traffic controller (ATC) to avoid runway incursions [3]. The design of taxiways, the roads of the airport, and the markings and signs use standardisation [4] to unify taxiing procedures, where the centreline refers to taxiway mainline with markings for taxiway navigation.

To enhance autonomy in taxiing navigation, one concept named "Follow the greens" [5] proposed updating airport lighting infrastructure to be linked with the taxiway planning and management system. The taxiway localisation employs

*School of Aerospace Transport and Manufacturing, Cranfield University, Cranfield, Uk, quentin.delezenne@orange.fr.
†School of Aerospace Transport and Manufacturing, Cranfield University, Cranfield, Uk, i.petrunin@cranfield.ac.uk
‡School of Aerospace Transport and Manufacturing, Cranfield University, Cranfield, Uk, billy.xu@cranfield.ac.uk
§General Atomics Aeronautical Systems Inc, Poway, CA, United States, jonathan.neptune@ga-asi.com
¶General Atomics Aeronautical Systems Inc, Poway, CA, United States, timothy.bleakley@ga-asi.com

global positioning system (GPS) service as a common solution to offer absolute positioning, whilst the GPS services may not always be available on every aerodrome [6] and are anticipated to be spoofed easily in practice.

The centreline identification using visualisation methods can be seen as a segmentation problem in computer vision, and solved using segmentation methods such as density-based spatial clustering of applications with noise (DBSCAN) [7]. The extraction of geometric information in line detection uses line-fitting transforming functions like Hough transform. For offering position estimation, pattern recognition algorithms are commonly applied on the camera vision and comparing what is seen to the airport map, using the Kullback-Liebler deviation [6]. The sliding window mechanism presents advantages of focusing on smaller regions to accelerate processing using the iterative principle [7]. Regarding controller designs, [8] presented a model-predictive controller (MPC) using landing gear kinematics analysis and line fitting solution to achieve following of the taxiway lines.

It is found that the onboard camera envisions enabling new features in different taxiway navigation applications, for instance, path verification to mitigate ATC mistakes [9], and real-time object detection to reduce conflicts during airport surface operation [10] [11]. Therefore, exploring the power of vision-based solutions in taxiway navigation tends to be promising to release reliance on GPS, and improve situational awareness capabilities.

Consequently, to enhance intelligence, and autonomy, as well as mitigate incursion and GNSS loss risks in UAV taxiing navigation, this study proposes and explores using realistic simulation to develop an end-to-end vision-based autonomous UAV taxiing navigation solution by identifying centerline and matching with airport representations. The source code of this proposition is open-sourced at https://github.com/DelQuentin/TaxiEye.

II. Proposed Autonomous Taxiway Navigation Solution

The proposed autonomous taxiway navigation system is composed of primary steps of centreline extraction, matching and navigation, and aircraft control. A high-level system architecture is illustrated in Figure 1a and a detailed diagram is presented in 1b. The camera feed containing the taxiway, line position, and direction is first extracted by segmentation and line detection methods to distinguish the background and taxiway centrelines. The targeted trajectories are represented and generated from airport maps, and stored coexisting with mission files and flight plans. After estimating relative distance between the extracted centreline information and ground truth centreline cross-referenced from the airport, the control input hereby is given to achieve the line following purpose in a typical flight control system (FCS) for path following to assure safety from hazardous commands.

A. Centreline Identification

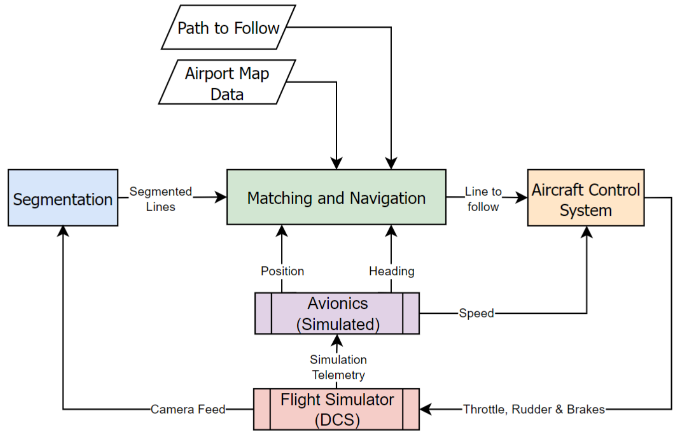
1. Pixel Extraction with Noise Cancellation

The developed centreline identification first performs segmentation based on features of: yellow colouration, contrast with the taxiway gound colour, and the image's colouration and edges. The procedure of centreline segmentation from the background is illustrated in Figure 2a. A combination of HSV filtering and edge detection generates masks representing regions existing centrelines. Edge detection is applied to compensate for line weathering in HSV, where the pixel contrast characteristic is represented in the edge detection approach. An erosion step applies on a combined mask from the edge detection mask and the HSV filtering mask to eliminate and cancel valueless noises after an "OR" operation. A dilate operation is applied to restore the mask areas to their right scale.

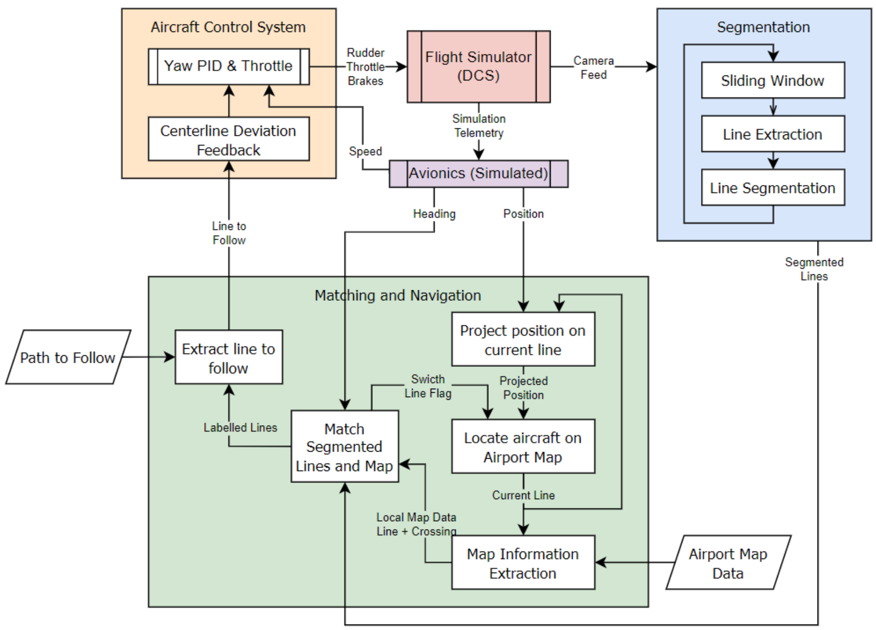
2. Line Divergence Identification using Sliding Windows

Given the noise-cancelled pixel maps, detecting divergence of centrelines provides insights into routes to be followed, hence the sliding window algorithm [7] is developed for this purpose with preliminary assumptions of visibility of centrelines and continuity of the line. The general principle of divergence detection is highlighted as follows:

• Given having a start point in a centreline, the next point for each line uses the last horizontal deviation to estimate the future lateral position of the line. The point is generated by one height of the sliding window above the last point of the line. This process is based on the sectioning of lane lines existing inline detection methods [12], and forces the algorithm to continue the lines by going further away from the aircraft. By allowing lateral deviation of points and forcing a step ahead in the point coordinates, the algorithm is designed to detect diverging lines and avoid considering converging lines. The consequence is that the algorithm is more stable in recognising the diverging path where the aircraft will have to decide which line to take, and will tend to ignore converging lines that the aircraft would never physically be able to follow. The theoretical filtering it applies on lines is illustrated in Figure 2b



(a) High-Level Functionality Architecture



(b) Detailed architecture of the proposed system

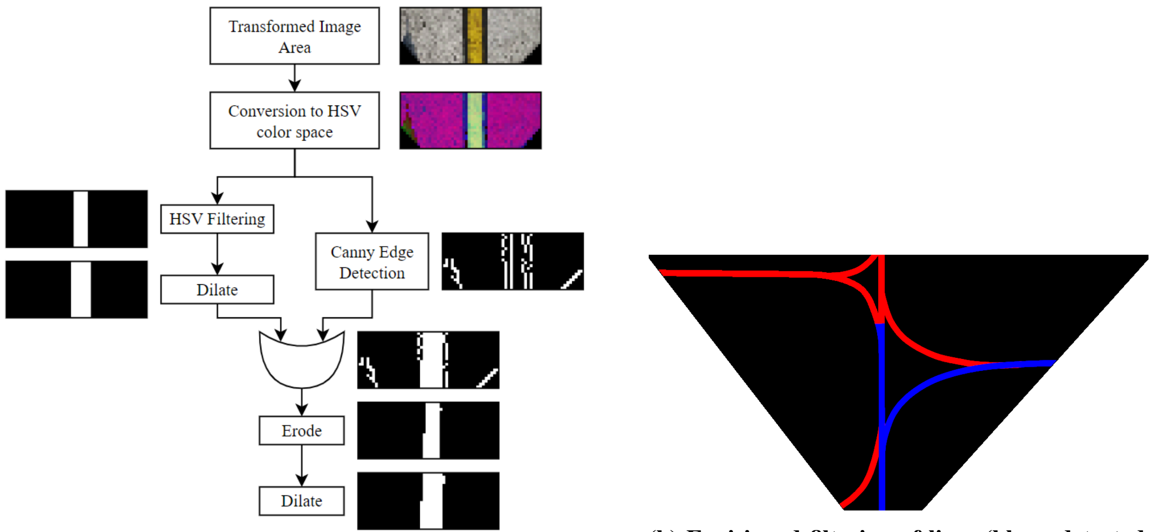
Fig. 1 Autonomous Taxiway Navigation System Architecture.

• The end of the algorithm filters the found lines to keep the ones concerning the next immediate crossing, the next steering decision, as illustrated in Figure 2b. It also reduces the amount of noise induced by the far field lines subject to the blur of the homographic dilatation.

• The kernel generation is implemented using a Gaussian kernel basis and represents the union of multiple kernels representing one point at a time. When two kernels get too close, only one peak is selected by using a threshold. The sensitivity, i.e. measuring the minimum closeness of two points that will be regarded as separate points is determined by standard deviation values in the kernel generation function.

B. Airport Map Matching and Navigation

The matching and navigation step aims to search and match the current taxiway centreline or divergence points with the airport map database. Assuming the initial point of aircraft is known to users, providing the divergence extracted by the sliding window, locating the relative position and matching with a map are enabled when the ground truth centrelines are labelled with representatives of heading and connectivity between other centrelines. Therefore, the path following in navigation essentially is to present a list of point labels as the generic procedures highlighted below:



(a) Taxiway line pixels extraction

(b) Envisioned filtering of lines (blue: detected, red: ignored) (non-segmented)

Fig. 2 Block diagram of crossing referencing airport maps with line following illustration.

• The matching process relies on comparing the number of lines identified, i.e. whether the right number of lines in the crossing is detected or will compare the relative directions of each line to determine the best quantitative match based on a correlation matrix.

• The algorithm will then search for the best matches over all the candidate set to provide the highest possibility corresponding to the identified line or crossing.

• This switching and matching process considers the aircraft heading to convert data between the ground and aircraft frames. Consequently, given the sequence of taxiway labels, the navigation strategy with real-time video feed is given to follow the route.

To make the right decision on the aircraft steering and correctly label the lines segmented on the camera view, the system needs to know its current location and, more precisely, on which line it is currently taxiing. A mistake in the current line label determination would induce the matching algorithm to be fed with the wrong expected lines, resulting in an incorrect matching and, therefore, a navigation failure. To help the navigation process, the system will require a connection to the avionic system to get the position information. It must also be considered that this information feed may contain noise or bias.

When the system is initialised on a known correct taxiway centreline, a continuous position estimation projects on the line to be aware of whether the aircraft is approaching the end of the line. Following the extracted navigation solutions and the predefined path, the navigation system decides whether to switch to the next line or stay on the current line when the aircraft moves across crossings. On converging lines, the aircraft can still follow in the converging direction; therefore, identifying the crossing and waiting for it to be visually passed has no effect.

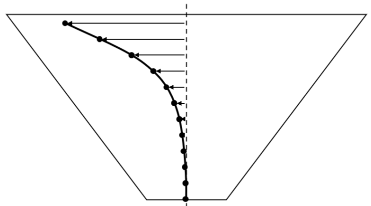
C. Aircraft Control

From navigation output, the controller is designed and demonstrated with a basic PID controller by using centreline deviation feedback represented by horizontal positioning deviations (see Figure 3a). The controller inputs are selected as a rudder, throttle, and brake accounting for steering yaw attitude, and speed maintenance (see Figure 3b).

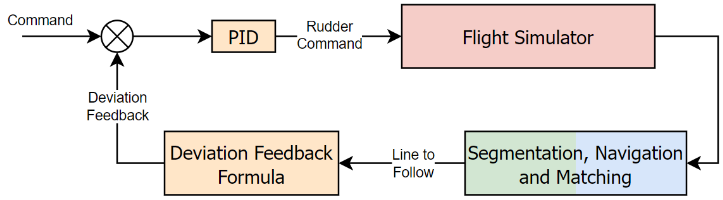
III. Simulation And Analysis

A. Flight Simulator and Data Collection

The proposed autonomous taxiway navigation solution is demonstrated using Eagle Dynamic's Digital Combat Simulator (see Figure 4a) due to its high-fidelity, accessibility to front-camera feeds and expansive environment including



(a) **Example of line to follow representation**

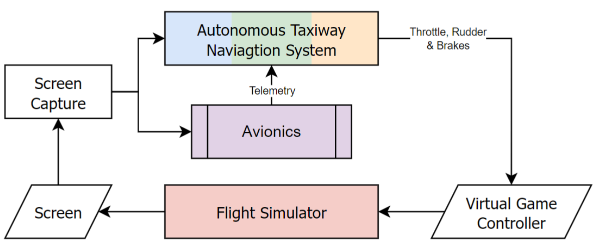


(b) **Representation of the rudder control in a closed loop**

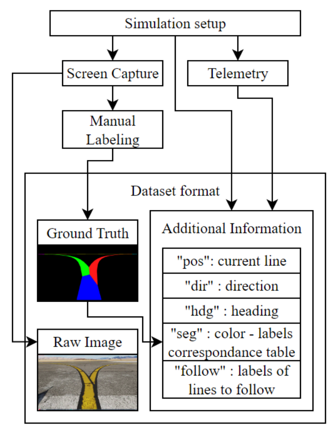
Fig. 3 Illustration diagrams for aircraft control.

aircraft maps. The airport considered is Creech Air Force Base, a real airbase where a part of the United States Air Force drone fleet is stationed.

Manual lines are labelled for each image taken from the recording to differentiate the lines from the rest of the picture and indicate the different lines for the segmentation process. To measure the path following performance in terms of success rate, each image also contains the ground truth centreline information to follow (see Figure 4b).



(a) Simulation framework



(b) Flowchart of data collection

Fig. 4 Block diagrams for simulation and data collection.

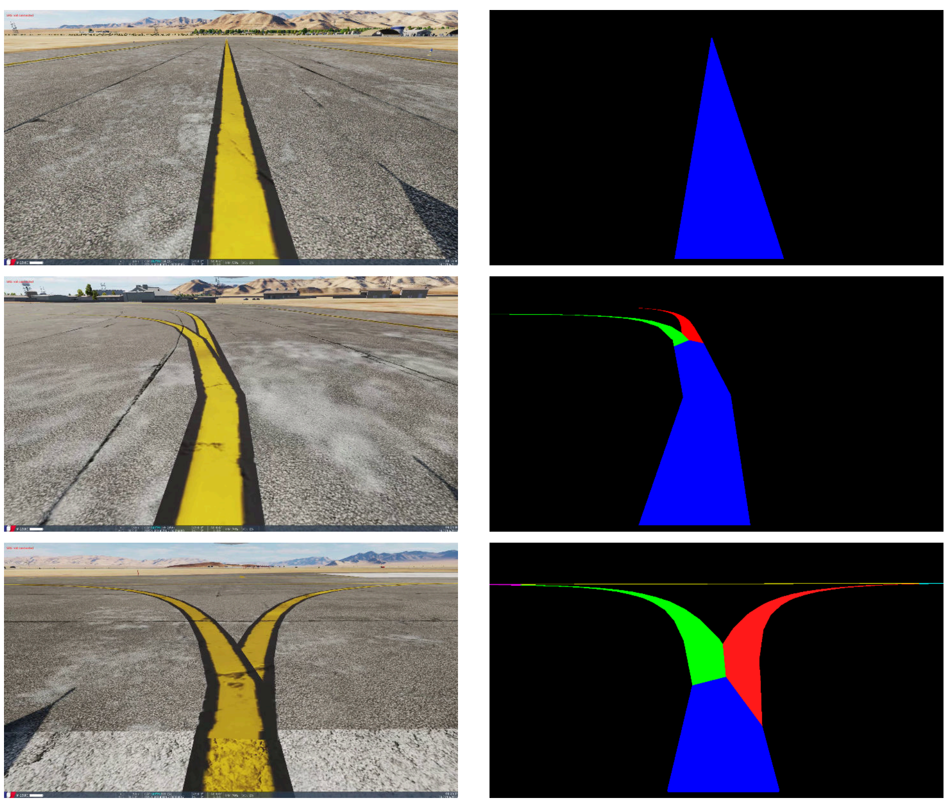
The airport representation is saved in plain text and tabular format detailing coordinates of the reference point, coordinates of the boundaries, the map picture and the list of taxiway line objects. Each taxiway line is described as a direct connection represented by two extreme points with X and Y coordinates. An example of the "A1" line is presented below. Figure 5 shows the map picture with the different lines from the airport map representation data on it.

```
"A1":
{
    "X_s":60795,
    "Y_s":4680,
    "N_s":{"A1_P1_1":1,"A1_P1_2":1},
    "H_s":271,
    "X_e":60559,
    "Y_e":4571,
    "N_e":{},
    "H_e":353},
}
```

Fig. 5 Airport map representations with implementations of lines in yellow.

B. Performance Analysis of Taxiway Centreline Extraction

Figure 6 demonstrates labelling of taxiway centreline with different colours to form a ground truth dataset.



1

2

3

Raw Ground Truth

Fig. 6 Demonstration of taxiway centreline extraction using realistic simulations.

Given the proposed centreline identification with sliding window, its detection performance is evaluated at the pixel level per image, where the performance indicators select Accuracy, Precision, Recall, F1 Score and (intersection over union) IoU with results presented in Table 1 below. Accuracy measures the success rate of predictions to the total instances. Precision measures the accuracy of positive predictions. Recall estimates the ratio of correctly predicted positive instances to the actual positives. F1 Score indicates harmonic mean of precision and recall. IoU measures the overlap region between predicted and ground truth.

Line Extraction Evaluation Metrics				
Accuracy	Precision	Recall	F1 Score	IoU
0.99	0.93	0.65	0.75	0.62

Table 1 Line extraction evaluation metrics table

From Table 1, it is noticed that accuracy presents a high value which means most background regions are correctly classified as not being a line. The precision value means that 93% of the picture areas are correctly detected as lines. The remaining 7% misses may result from blurriness in the far field of the image due to homographic transforming for converting bird views to top-to-bottom views. Moreover, the line extraction driven by a sliding window will not consider the image's borders which also adds degradation.

The recall value indicates that the algorithm is detecting only 65% on average of the taxiway line information available in the image. The low recall value can be explained by the principle of the sliding window mechanism, which is designed to bypass detecting certain lines, as explained in Figure 2b. Specifically, all converging lines and lines beyond the first visible crossing lines are undetected. Similar to the IoU value, the results are bound to the proportion of lines the algorithm is designed to detect that cause low numbers.

C. Performance Analysis of Line Matching

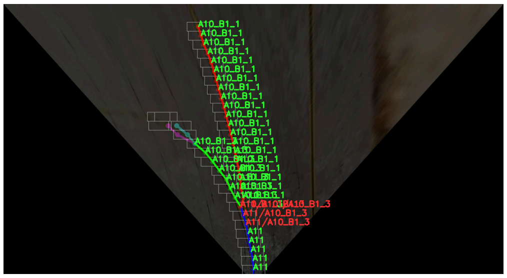
To evaluate whether the identified taxiway centrelines are correctly matched with the database with aligned labels, this measurement aims to understand the line-matching performance in terms of accuracy, precision, recall, F1 score and IoU. A demonstration of the line-matching results is illustrated in Figure 7, where the green labels stand for correct matching while the red labels represent incorrect classification and matching. The metrics results for this evaluation are displayed in Table 2 below.

Line Matching Metrics				
Accuracy	Precision	Recall	F1 Score	IoU
0.873	0.455	0.454	0.454	0.446

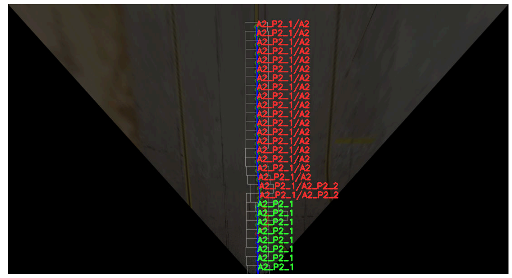
Table 2 Line matching evaluation metrics table

From Table 2, low values of performance metrics arise from accumulative errors since the iterative processing is adopted in the sliding window mechanism. Similar to the centreline extraction result, the values of precision, recall and IoU are also affected by ignorance of converged lines leading to failure in updating labels as demonstrated in Figure 7b. The diverging points of lines also present a small area of error with demonstrations presented in Figures 7a, 7c and 7d. It is worth noting that the accumulative errors in one image will be continuously updated and mitigated to a small number for the time being as most cases are correctly identified.

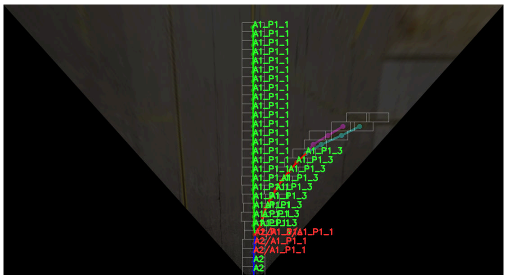
When we consider line matching as a multi-label classification task, each class stands for a centreline in the airport taxiway, and the class "None" represents the background. Figure 8 presents a confusion matrix result to evaluate identification and matching performance by reception of video stream input. The proposed system builds true knowledge of each picture with continued updates when approaching crossings. Given the high classification numbers, the matrix shows how the algorithm is ultimately able to differentiate lines in the image from the video stream. The trend of the good classification, which represents 87% of the classification based on the Accuracy metric, is visible by the brighter diagonal of the matrix. The column and row representing the background display specific results compared to the rest, with much more non-zero values. These values result from situations where lines are classified as background when the algorithms ignore them. Among the outlying values, the classification of the background as lines results from issues of



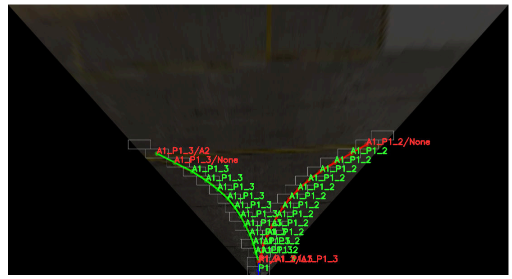
(a) Sample 2



(b) Sample 3



(c) Sample 4



(d) Sample 5

Fig. 7 Example of matching results illustrated on segmentation image

detection in the long distance, and the classification of A1_P1_1 as A2, or A2 as A1_P1_1, or even A2_P2_1 as A3, which are adjacent lines one to another, illustrate the phenomenon where the algorithms are not detecting the diverging point and concatenating the lines together, in one way or another.

D. Performance Analysis of Line to Follow

This measurement evaluates the system performance with controller design and aircraft dynamics taken into account. With the automatic detected and selected taxiway centrelines, the evaluation method measures the success rate, and failure rate of following a given path and the result is presented in Figure 9.

This matrix in Figure 9 measures the behaviour performance of the system including accumulative phenomenons induced by previous algorithms including effects of incorrect classification coming from the segmentation, and incorrect matching outcomes. A general analysis of this matrix values is presented below:

→ The image's background, composed of the ground or objects, is successfully rejected with a rate higher than 99%. Therefore, it can be concluded that the algorithms are not confusing the background information with lines, either line to follow or other taxiway lines.

→ The overall classification of all lines reveals that, while 29% of other taxiway lines are correctly detected and classified, 64% of the lines present in the field of view and not part of the line to follow are qualified as background. This is mostly due to the design of the segmentation algorithm, which ignores on purpose several lines in the image that will not be lines to follow due to their geometry.

→ Still, some lines in the image that are not meant to be followed are classified as lines to follow, at a rate of 7%. This is due to the combination of segmentation and matching in cases a line to follow leads to a crossing, but only one line of the crossing is detected.

→ Finally, the algorithms are successfully detecting 71% of the line to follow, and the rest of the line to follow is classified as background. This proves that the algorithms are focusing on the right path and that when the path is not detected, it is not detectable by the algorithms. Indeed, a part of a line to follow is considered as background, this means that the extraction has not been able to get to this point in the image.

E. Generalisation Capability Analysis

The generalisation capability of this proposition is tested over multiple scenarios following test-driven development (TDD) process. The evaluation results, as shown in Table 3, demonstrate that this proposition presents promising generalisation capability to other selected cases. Specifically, when the system is tested on a path not considered in the

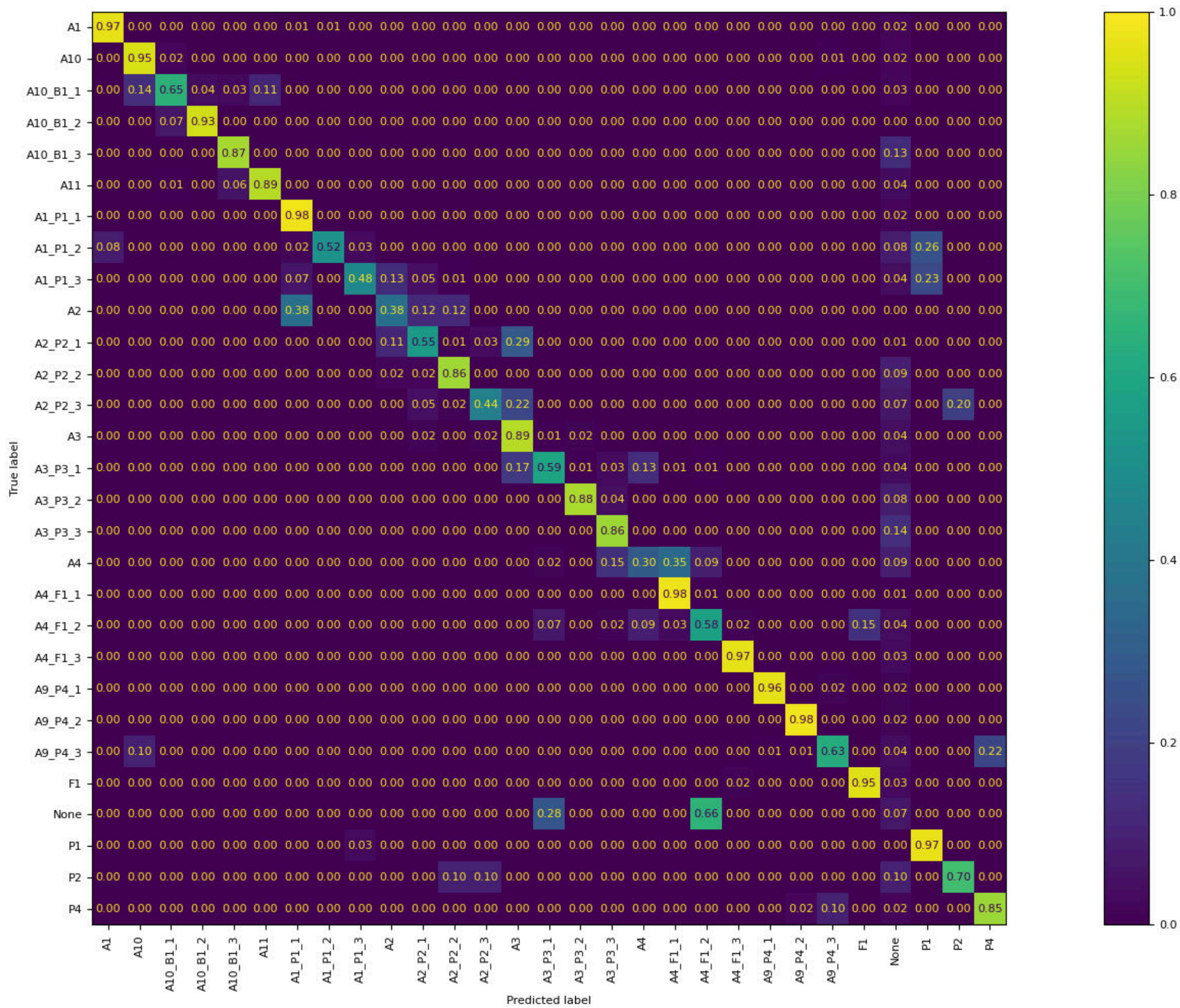


Fig. 8 Confusion matrix of the matching evaluation showing the 27 lines seen over the 95 samples of the dataset

development process, the system is successful on only eight paths over twelve, representing 67% of success.

System tests in real-time simulation		
Window setting (w × h in m)	Tests on paths used in TDD	Tests on other airports
2.5×1	8 / 8	8 / 12
3×0.5	N/A	4 / 4
Total Achieved	8 / 8	12 / 12

Table 3 Results of the testing batch of the system in real-time simulation

The analysis of the reasons behind the system's failures in these four cases reveals the significance of sliding window size under crossing environments involving multiple and physically close lines. The first test and the development used a default window size of 2.5×1 m. To evaluate if the algorithm can solve these cases with another window size, a new run on these four cases is done with a window size of 3×0.5 m. The change in the window size presents a decrease by 50% of the height of the window, allowing an increased vertical tiling of the image; this is equivalent to augmenting the sample frequency, and therefore gathers more information. The new tests are successful on each of the four paths.

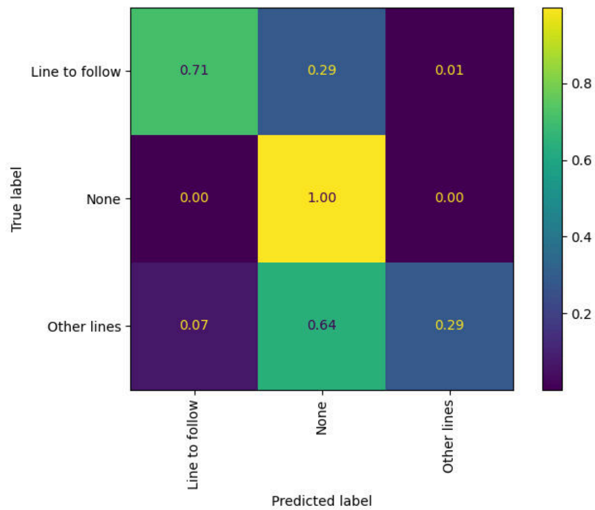


Fig. 9 Confusion matrix of the line to follow generation over the 95 samples of the dataset

Consequently, this measurement suggests the importance of fine-tuning parameters to optimise the overall system performance regarding environmental complexity and diversity.

F. Computational Complexity

The main axis driving the analysis of this evolution is the number of lines present in the camera vision the system will tend to detect. The sliding window algorithm performs its task based on points, and the more lines are present in the camera image, and the longer the lines are, the more points will be generated by the sliding window algorithms.

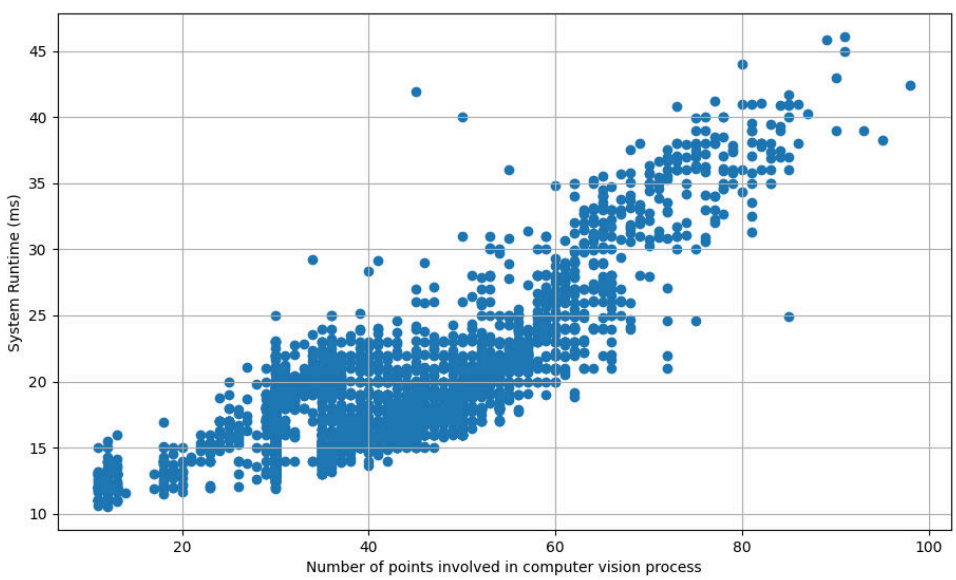


Fig. 10 Point cloud of measured runtime by number of points considered by the sliding window algorithm

Figure 10 illustrates the evolution of the runtime, expressed as a function of the number of points detected by the sliding window algorithm. By using library functions, the correlation of the two variables has been estimated at 83%, which demonstrates a clear correlation. This correlation is firstly explained by the sliding algorithm process which runs multiple sub-process based on the detected points. Moreover, the segmented lines are a subset of these points, and other subsystems will run processes and computations based on these points.

Consequently, it is found that to maintain the runtime performance by performing correlations, taking fewer points into calculation in the sliding window mechanism is suggested with solutions, for instance, by increasing the size of the window, diminishing the tiling of the picture and therefore the number of points possibly extracted.

IV. Conclusion

To enhance autonomy in UAV taxiway navigation, a vision-based autonomous navigation system is presented and demonstrated with the adoption of a camera feed mounted under the aircraft nose. The high-level navigation system designs and detailed functionality solutions are presented, specifically including methods of lightweight centreline identification, airport matching and navigation, and aircraft control in a realistic simulation environment. Via thorough performance assessment in terms of indicators like accuracy, precision, recall, F1 score, and IoU metrics, it is found that this proposition presents limitations in identifying merged paths with correct labels due to centreline similarities. However, those misfunctions shall not degrade the ultimate path-following capability given iterative compensations under the sliding window mechanism. The performance evaluations indicate the promising potential of camera visions in enabling autonomous UAV taxiing with 71% successful rate of detecting correct lines to follow and the remaining 29% as background. The proposition also presents a good generalisation capability of more than 67% success rate when testing over other airport scenarios verified following a test-driven development process. Future work could be exploring advanced line extraction and map matching methods upon this platform to further improve the path-following accuracy and generalisation capability.

References

[1] *European ATM Master Plan: Digitalising Europe's Aviation Infrastructure: Executive View: 2020 Edition*, 2nd ed., European Union, 2019.

[2] *Order JO 7110.65, Air Traffic Control*, 2023.

[3] United States Federal Aviation Administration, "Runway Incursions," , 2022. URL `https://www.faa.gov/airports/runway_safety/resources/runway_incursions`, (Accessed 4th August 2023).

[4] *CA 150-5340-1M, Standards for Airport Markings*, 2020.

[5] *SESAR Solutions Catalogue 2019*, third edition ed., Publications Office of the European Union, 2019.

[6] Lu, B., Li, B., and Chen, W.-H., "Map-Enhanced Visual Taxiway Extraction for Autonomous Taxiing of UAVs," *IFAC-PapersOnLine*, Vol. 48, No. 9, 2015, pp. 49–54. https://doi.org/https://doi.org/10.1016/j.ifacol.2015.08.058, URL https://www.sciencedirect.com/science/article/pii/S2405896315009258, 1st IFAC Workshop on Advanced Control and Navigation for Autonomous Aerospace Vehicles ACNAAV'15.

[7] Batra, A., and Gauci, J., "Aerodrome Taxiway Line Detection and Cross-Track Error Estimation using Computer Vision Techniques," *2020 IEEE Aerospace Conference*, 2020, pp. 1–9. https://doi.org/10.1109/AERO47225.2020.9172787.

[8] Wamster, A., "Taxiway Centreline Tracking," , 2022. Canfield University MSc Thesis.

[9] He, S., Du, K., Wilhelm, J., and Jeannin, J.-B., "Verification of an Airport Taxiway Path-Finding Algorithm," *2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC)*, 2020, pp. 1–10. https://doi.org/10.1109/DASC50938.2020.9256698.

[10] Weiszer, M., Burke, E. K., and Chen, J., "Multi-objective routing and scheduling for airport ground movement," *Transportation Research Part C: Emerging Technologies*, Vol. 119, 2020, p. 102734. https://doi.org/https://doi.org/10.1016/j.trc.2020.102734, URL https://www.sciencedirect.com/science/article/pii/S0968090X20306483.

[11] Zhou, H., and Jiang, X., "Research on Taxiway Path Optimization Based on Conflict Detection," *PLOS ONE*, Vol. 10, No. 7, 2015, pp. 1–17. https://doi.org/10.1371/journal.pone.0134522, URL https://doi.org/10.1371/journal.pone.0134522.

[12] Wang, Y., Teoh, E. K., and Shen, D., "Lane detection and tracking using B-Snake," *Image and Vision Computing*, Vol. 22, No. 4, 2004, pp. 269–280. https://doi.org/https://doi.org/10.1016/j.imavis.2003.10.003, URL https://www.sciencedirect.com/science/article/pii/S0262885603002105.

2024-01-04

Autonomous navigation with taxiway crossings identification using camera vision and airport map

Delezenne, Quentin

AIAA

Delezenne Q, Petrunin I, Xu Z, et al., (2024) Autonomous navigation with taxiway crossings identification using camera vision and airport map. In: AIAA SCITECH 2024 Forum, 8-12 January 2024, Orlando, USA. Paper number AIAA 2024-1300

https://doi.org/10.2514/6.2024-1300