In: Proceedings of the 2024 IEEE International Smart Cities Conference (ISC2), 29 Oct - 1 Nov 2024, Pattaya, Thailand
DOI: 10.1109/isc260477.2024.11004215

Automating Emergency Landing in Cities with Cognitive Stress Reducing Explanations to Pilot-in-the-Loop

Hamsa Balakrishnan[1], Bastien Lechardoy[1], Maxime Collignon[1], Anwesha Sen[1], Ahmed Ali[1], Ankit Verma[1], Mariusz Wisniewski[1], Paris Chatzithanos[1], Antonios Tsourdos[1], Yang Xing[1], Weisi Guo[1*]

***Abstract*—Future smart cities will integrate urban air mobility (UAM) where electric vertical take-off and landing aircrafts (eVTOL) will improve labour and logistic mobility. One of the challenges in urban flight is emergency landing, where an eVTOL needs to land in an unofficial roof top or clear area safely within a time frame. In semi-autonomous eVTOLs, artificial intelligence (AI) is expected to assist in identifying emergency landing locations and navigation vectors, whilst the pilot flies and lands the eVTOL. Autonomously searching and correctly identifying safe candidate locations is important for safety of both the eVTOL and ground stakeholders. Furthermore, the processes of using AI to identify such locations should ideally be explainable to the pilot to reduce cognitive stress and aid the preparation of the emergency landing procedure. Here, we have developed a simulated emergency urban landing platform, whereby an eVTOL rotorcraft is scanning the ground for suitable emergency landing locations and actively explaining the navigation vectors to the pilot-in-the-loop. Cognitive stress is measured using real experiments using heart rate monitor, and an actor-critic deep reinforcement learning is used to learn what explanations are useful to reduce cognitive stress. The end result is that the eVTOL can identify emergency landing candidates, navigate to the safe landing zone (SLZ) whilst performing obstacle avoidance, and explain its decision making to the pilot-in-the-loop whilst minimizing its cognitive stress. We show through 3 scenarios that we can indeed reduce cognitive load significantly and also reduce the average time to reach the SLZ.**

***Index Terms*—Autonomy; Urban Air Mobility; Navigation; Explanations; Human-Machine-Interface**

I. Introduction

Smart cities will increasingly integrate eVTOLs and air taxis for transportation [1], and mobile sensing/communications/computing [2]. Autonomous navigation has been a subject of intensive research and development over the past few decades, especially for urban and congested areas [3], [4]. Landing is one of the biggest challenges for rotor-crafts, due to the tight constraints in landing zone clearance. Data from Transport Safety Board of Canada shows that the greatest numbers of helicopter accidents were associated with landing (57%). Emergency landing is more challenging and can arise due to mechanical or power failure. All of the current aviation approaches assume that there are sufficient clear spaces available for landing - usually a discussion between the aircraft and air traffic control, but emergency scenarios in cities can mean there is no time to discuss.

A. Review of Emergency Landing Research

Most current research focus on 3 areas: (1) identification of safe landing zones (SLZ) [5], (2) risk-based trajectory planning [6], and (3) robust guidance in GNSS/GPS denied conditions [7]. There is widespread recognition that future autonomous rotorcrafts are not only electric vertical take-off and landing platforms (eVTOLs), but they are also hybrid autonomy where a human(pilot)-in-the-loop (HITL) is used to ensure rapid rollout and safety compliance. Here, the AI is assigned to support tasks typically given to co-pilot and air traffic control, such as SLZ identification and navigation vectors, whilst ensuring the reduction of cognitive stress to pilot [8]. In [5], various focus areas were identified for SLZ detection, including the stages of (i) ground feature filtering and sensor fusion, and (ii) autonomous landing area selection. The techniques involved include but are not limited to stereo ranging, homography estimation and adaptive control, SLAM and segmentation, optical flow, and digital elevation modeling.

The opaque nature of AI usage in safety-critical areas of aeronautics is a fundamental issue, and in aeronautics it is important to improve trustworthy collaboration with pilots in the loop [9]. One way of assessing that is to determine the cognitive load of operators using such systems [10]. Usually, the cognitive load is derived with metrics that aims at quantifying the mental capacity or state of an individual while performing certain tasks. Moreover, it is often based of physiological factors that are representative of one cognitive state. Traditionally, cognitive load has been measured using eye tracking technology, renowned for its precision in assessing resources

[1]All authors are with the School of Aerospace, Transport, and Manufacturing, Cranfield University, Bedford, UK. *Corresponding author: weisi.guo@cranfield.ac.uk. This work is supported by Leonardo UK and EPSRC TAS-S: Trustworthy Autonomous Systems: Security (EP/V026763/1).

Published by IEEE. This is the Author Accepted Manuscript issued with: Creative Commons Attribution License (CC:BY 4.0).
The final published version (version of record) is available online at DOI:10.1109/isc260477.2024.11004215. Please refer to any applicable publisher terms of use.

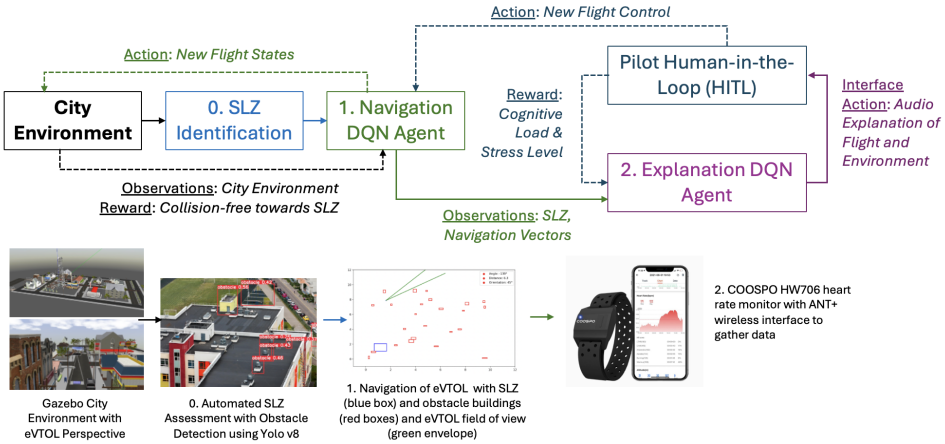


Fig. 1. System Diagram of Autonomous eVTOL Emergency Landing: 0. Safe Landing Zone (SLZ) Identification, 1. Navigation to SLZ, and 2. Cognitive Load Reduced Explanations to Pilot Human-in-the-Loop (HITL).

linked to attention and cognitive engagement through gaze patterns and pupil dilation. Other methods include speech [11]. The heart rate monitor offer a non-invasive, available and cost-effective alternative [12], and past studies suggest that fluctuations in heart rate can serve as an indirect indicator of cognitive load [13].

B. Gap and Novelty

What is not sufficient acknowledged are two important areas: (1) automatic emergency eVTOL SLZ identification with self-navigation, and (2) low cognitive load explanations to the HITL pilot. The gap is how to use sequential explanations to improve personalised cognitive load reduction which requires reinforcement learning. We aim to show how we can create realistic urban flight scenarios via a simulation platform and:

- Automatically identify the tilt and features that represent hazards to SLZ creation;
- Identify suitable SLZ candidates and create navigation vectors to the best candidate based on trajectory and urban obstacles;
- Learn through a Deep Q-Network what the best personalised explanations are to the pilot HITL in order to minimize cognitive stress;
- Train and evaluate navigation success in different city landing scenarios and landscapes;
- Evaluate cognitive stress with different information feedback to the pilot.

Combined, they offer a new level of human machine interfacing for real-time high stress situations for future urban air mobility in smart cities.

II. City Modeling and SLZ Identification

The overall system architecture shown in Fig.1 is formed of several intertwined components that aims at navigating to a SLZ whilst minimizing stress on the pilot HITL. The simulation, defined in ROS2 and Gazebo, lets a human operator pilot the eVTOL in a synthetic city environment. Chronologically, the eVTOL begins the segmentation and classification of suitable roofs near its field of view. Using roof mapping and obstacle detection algorithms, a SLZ is chosen. Two DQN reinforcement models are trained with the human in the loop.

1) The navigation model takes inputs from the city environment and SLZ location to create updated flight vectors in order to reach there whilst reducing the number of building obstacle collisions.
2) The explanation agent takes in the necessary information needed by the pilot and decides how to communicate this to the pilot in audio form, and uses the pilots stress data to improve its communication. The different instructions, their timing and frequency are the objective of the policy training of the model.

A. City Flight Simulation Environment in ROS-Gazebo

The choice of ROS2-Gazebo respectively as robot operating environment and visual software were motivated by several considerations. First, both works well together and are easy to pair and synchronize. Second, they are present in many different research papers as a flexible and suitable framework to perform reinforcement learning in 3D environment [4]. Last, it allows us to tackle one of the greatest challenges with regards to the research, operator immersion. In fact, the acquisition of the operator cognitive load during the simulation has values only if the latter can put the tester in real, concrete, and stressful situations. ROS2 and Gazebo,

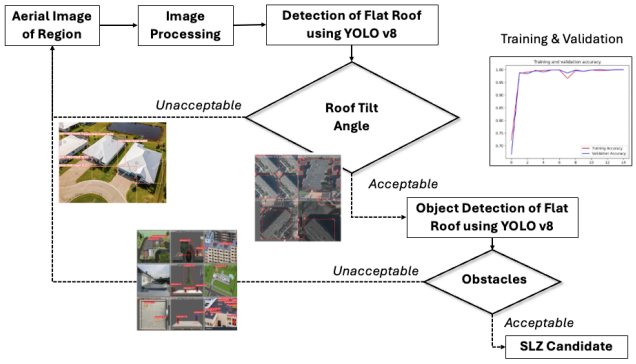


Fig. 2. System Diagram of SLZ Identification that Considers Flat Surfaces and Obstacles.

respectively with realistic flight dynamics and 3D building models, are designated candidates to perform that.

In the simulation, we model the eVTOL as a quad-copter eVTOL and is equipped with three sensors: two cameras and one sonar. The front camera is used to simulates the view of a real pilot while the bottom one is destined to be used for the roof mapping and obstacle detection. For its part, the sonar is used during the navigation process to detect potential obstacles through acoustics waves. The eVTOL is piloted through keyboard which allows the operator to modify the velocity and angular speed, the altitude and the directions of the device.

Last, an important focus is placed on the immersion of the operator and the realism. The precision and coherence of the cognitive load measurement rely on the device used, the methodology but also greatly on the simulation capacity to stress the experimenter. To this extent, ROS2 allows the spawned eVTOL to have very realistic flight dynamics since it is possible to define its inertia and many other characteristics. Moreover, the gazebo simulation coupled with ROS visualization (rviz) can put the front camera in full screen so that the experimenter feels like he is in the pilot compartment. Additionally, we were able to randomize the creation of the environment so tat the operator does not get too comfortable with the same city configuration. Finally, the 3D graphics, while outdated, offer plausible city reconstruction. Each one of this implementation ensured that the cognitive load acquired during training would be as relevant as possible in the context of the simulation.

B. Automated SLZ Identification

While operating in the airspace, the eVTOL will give the roof detection model an aerial image of the area, assisting in the identification of a flat region that could serve as the helicopter's helipad. The team experimented with several segmentation and classification models in the early stages of research, including Masked R-CNN and VGG16 CNN for roof identification. Although the Masked R-CNN performed poorly with erroneous segmentation and masking of the roofs, the VGG16 scored a decent accuracy (97%) with no overfitting. The team decided to use various models, with a particular emphasis on transfer learning, to improve the unsatisfactory findings; in subsequent phases of development, the YOLO v8 53 convolutional layer model yielded improved results. Training was with 50 epochs and batch size of 5 were used. The achieved precision was 0.65 and recall 0.38.

Detecting objects within the landing zone area is our next assignment after the flat roof has been identified in the landing zone recognition model. The primary purpose of an obstacle detection model is to locate any obstacles that might impede a landing. We conducted a few test runs using various detection models, including DETR and Detectron2 [14], however, in the early runs, both models performed poorly. YOLO v8 was the ultimate model used for obstacle detection. Training was with 50 epochs and batch size of 5 were used. The achieved precision was 0.72 and recall 0.61.

III. Navigation to SLZ and Cognitive Load Reduction

After detecting and determining a sufficient SLZ, it is necessary for the pilot to be able to safely reach the selected SLZ in a way that is efficient and safe. Here, we are interested in several goals, to minimize the time t since the emergency, to avoid obstacle buildings on the way to SLZ, and to communicate navigation vectors θ and distance d to the SLZ. The latter involves making the information/instructions provided by our system as stress-free as possible and least likely to increase cognitive load - see Fig.3.

A. Automated SLZ Navigation

We first begin with a machine learning model to automatically provide navigation vectors to the eVTOL to reach SLZ without crashing into obstacles [15]. We train our model for navigation (i.e. giving instructions to the pilot regarding the path to follow) using an Actor-Critic Reinforcement Learning (AC-RL) algorithm. State-of-the-art RL rely on SLAM algorithms which require two sub-models: (1) one for local mapping of the environment (using images or lasers) and (2) another one for path planning given the current position and the one of the target. Our model will have an input variable indicating the presence of an obstacle ahead or not. We use an Actor-Critic algorithm because the action space contains an angle of turn which must have continuous values. Actor-Critic or more widely Policy-based algorithms suit well a continuous action space, while this is more complicated to implement with algorithms based on Q-values. To train the

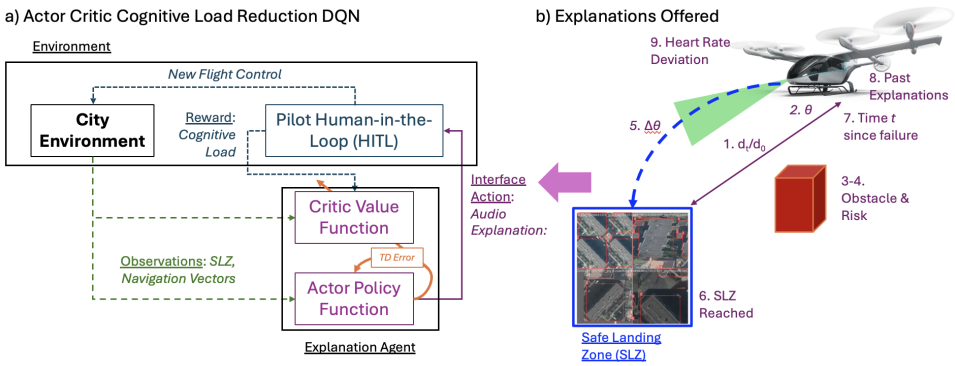


Fig. 3. a) Actor-Critic Cognitive Load Reduction DQN with Audio Explanations; b) Explanations Offered to Pilot including navigation data (1-6), eVTOL status (7), historical data (8), and heart rate deviation (9).

algorithm, we simulate a 2-D Python environment containing very simple obstacles (rectangles, representing high buildings that could be in the path to the target roof). The environment is totally random; the spawn point, the number of obstacles and their dimensions, and the location and dimensions of the target roof are randomly set, and the environment is randomly re-generated after each episode. This randomization increases the number of episodes before reaching reward stability, but it prevents overfitting (the agent will be able to adapt to a wide range of environments).

The navigation AC-RL model takes the following state inputs and uses the following rewards:

- The state inputs are the following: $x_1(1) = \frac{d_t}{d_0}$ ratio of distance to SLZ and initial distance, $x_1(2) = \frac{\theta_t}{\pi}$ angle difference to SLZ, $x_1(3) = \{1, 0\}$ presence of obstacle ahead or not, $x_1(4) = \{\frac{3-d_{obs}}{3}, 0\}$ collision risk.
- The reward function is the following: $R_1 = -100$ if agent collides, $R_1 = +150$ if SLZ is reached, and $R_1 = -\frac{d_t}{2d_0} - \frac{\Delta\theta}{4\pi}$ otherwise for non-terminal approach steps.

We fixed the network sizes with two hidden layers with 1024 and 512 neurons and performed basic hyper-parameter tuning with results in Fig.4 for three pairs of values. This can improve the rate of convergence from over 1 million epochs to 100,000 epochs. We also test the models with these different learning rates in random generated environments for a thousand episodes.

B. Cognitive Load Measurement

Traditionally, cognitive load has been measured using eye tracking technology, renowned for its precision in assessing resources linked to attention and cognitive engagement through gaze patterns and pupil dilation. However, due to emergency landing relying on as few components as possible and possible cockpit vibrations, deploying eye tracking technology was not considered the most feasible option. In light of these constraints, we adopted the use of a heart rate monitor. To enhance the rigor of our data collection process, we used the COOSPO Heart Rate Monitor [16].

C. Cognitive Load Reduction Learning

We train a DQN model that aims to determine at each instant which type of audio instruction will be given to the pilot. This training will enable us to obtain a model that helps slightly reduce the stress of the pilot when reaching a secure roof for landing. For this purpose, we use a AC-RL with a replay buffer memory and discrete outputs: it contains several variables, which are the type of information / instruction to give to the pilot, the one chosen by the model will be equal to 1, the other ones are equal to 0 - Fig.3a. This ensures that the pilot receives only one piece of information at a time (the opposite would surely increase their cognitive load).

The cognitive load AC-RL model 2 takes the following state inputs and uses the following rewards (Fig.3b):

- The physical state inputs are the same as the navigation model with the following: $x_2(1) = \frac{d_t}{d_0}$ ratio of distance to SLZ and initial distance, $x_2(2) = \frac{\theta_t}{\pi}$ angle difference to SLZ, $x_2(3) = \{1, 0\}$ presence of obstacle ahead or not, $x_2(4) = \{\frac{3-d_{obs}}{3}, 0\}$ collision risk.
- The augmented state inputs are derived from the navigation model with the following: $x_2(5) = \frac{\Delta\theta}{\pi}$ instruction of turn given by the Actor network at the current step, $x_2(6) = \{1, 0\}$ close to SLZ or not, $x_2(7) = \frac{t}{<T>}$ time since failure, $x_2(8) \in \{0, 1\}^n$ is previous information given to the pilot, and $x_2(8) = \frac{h-<h>}{20}$ which is the deviation in heart rate from average.
- The reward function is the following: $R_2 = -100$ if agent collides, $R_2 = +100$ if SLZ is reached, $R_2 = \sum_{i=0}^{T} CL_i(t)$ if the time since failure exceeds lifespan, and $R_2 = -\frac{d_t}{4d_0}$ for non-terminal approach steps.

We perform ϵ-greedy strategy; we set the initial value at a high value $\epsilon = 0.2$ so that our model can explore many

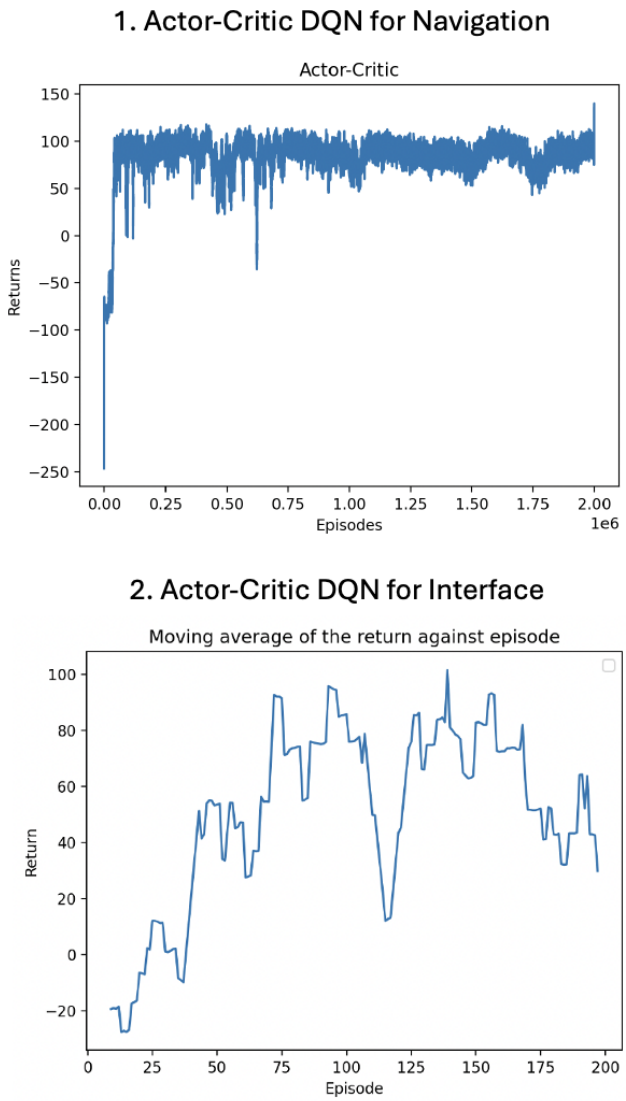


Fig. 4. 1. Navigation AC DQN RL training with learning rate hyper-parameter tuning ($\alpha_a = 10^{-5}, \alpha_c = 10^{-6}$). 2. Cognitive interface RL training with $\epsilon = 0.2$.

possible actions from the beginning and quickly determine a strategy that rapidly increases the reward, and we then gradually decrease the value.

IV. Results

In this section we test the global architecture, i.e. we evaluate its ability to reduce the pilot's cognitive load/ stress. To do this, we conduct a test experiment using our simulation environment (ROS, Gazebo) and our trained models. This experiment involves measuring the heart rate of the pilot as well as the time it takes for him/her to land the eVTOL on the target roof, in 3 different feedback situations:

	Situation 1	Situation 2	Situation 3
Average cognitive load	0.12	0.092	-0.074
Average time to reach the target	2.58	1.96	1,76

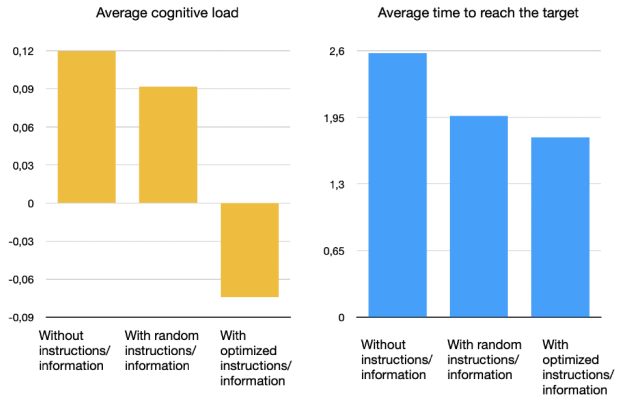


Fig. 5. Cognitive load impact from different feedback situations.

1) We only show to the pilot the position of the agent in the environment and the one of the target SLZ roof, without any instruction/ information;
2) We show instructions/ pieces of information to the pilot, but they are selected randomly among those presented in the previous part;
3) We use our trained models to provide the pilot with instructions/ pieces of information.

To conduct this experiment, we carry out 10 runs per situation with 2 different pilot operators. The situation is chosen randomly at each run, to avoid encountering some bias. For instance, an operator could start with a high heart rate that would decrease during the experiment, thus the different situations would correspond to different heart rates but not because of our system. We measure the average cognitive load (heart rate, normalized) as well as the average time to reach the target (normalized: divided by the shortest time possible to reach the target). We obtain the following results in Fig.5.

An example of the end-to-end global architecture results and experiment with pilot is shown in Fig.6. We first observe a slight decrease of cognitive load and of the time to reach the target when the operator uses our system. We show that random instructions seem to have a small cognitive load reduction compared to no instructions (0.12 to 0.09), and optimised instructions can lead to a sharp drop. For the time to reach target, we also see that random and optimised instructions does indeed reduce the time from 2.6 nominal time units to 1.8. It should be noted that several factors need to be considered in the analysis of the results:



Fig. 6. Example results from the experimentation: (top-left) SLZ hazard detection, (top-right) autonomous navigation success to SLZ, (bottom-left) first person immersive view of landing, (bottom-right) pilot HITL measuring heart rate and controlling the eVTOL.

- The experiment consisted in few runs. A more serious experiment could consist in more runs, with additional testing situations to enable further analysis.
- The operators were the same people than those who designed the system, i.e. us. Unconsciously, we may have influenced the results in a certain direction by ensuring better performance during the runs, where our system was in action, meaning being more focused or calmer during those runs, because we obviously want to prove the efficiency of our system.

V. Conclusions

Future smart cities will integrate urban air mobility (UAM) for eVTOLs. One of the challenges in urban flight is emergency landing, where a eVTOL needs to land in an unofficial roof top or clear area safely within a time frame. Here, in this paper, we have developed a simulated emergency urban landing platform, whereby a rotor-craft eVTOL is scanning the ground for suitable emergency landing locations and actively explaining the navigation decision processes to the pilot-in-the-loop. Cognitive stress is measured using real experiments using heart rate monitor, and a DQN is used to learn what explanations are useful to reduce cognitive stress. The end result is that a eVTOL rotor-craft can identify emergency landing candidates, navigate to the landing zone whilst performing obstacle avoidance, and explain its decision making to the pilot-in-the-loop whilst minimizing its cognitive stress. We show that we can indeed reduce cognitive load significantly and also reduce the average time to reach the target. Future research will focus on improving the experiment data size, explaining deep neural network reasoning when navigating under uncertainty, and the impact of cybersecurity compromises [17].

References

[1] M. C. Danciu, N. Yingta, I. Ur Rehman, and S. Aleshaiker, "Uav automated charging station and charging network in smart cities for telemedicine delivery," in *IEEE Int. Smart Cities Conference*, 2023.
[2] W. Guo, "Swarm UAV communications with diversity and delay trade-off using mobile caching," in *IEEE International Smart Cities Conference*, 2019, pp. 247–250.
[3] X. Yang and P. Wei, "Autonomous free flight operations in urban air mobility with computational guidance and collision avoidance," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 9, 2021.
[4] M. Wisniewski, I. O. Gałan, P. Chatzithanos, W. Guo, and A. Tsourdos, "Autonomous navigation in dynamic maze environments under adversarial sensor attack," in *IEEE International Conference on Unmanned Aircraft Systems (ICUAS)*, 2024.
[5] M. Alam and J. Olouch, "A survey of safe landing zone detection techniques for autonomous unmanned aerial vehicles (UAVs)," *Expert Sustems and Applications*, vol. 179, 2021.
[6] J. Sláma, P. Váňa, and J. Faigl, "Risk-aware trajectory planning in urban environments with safe emergency landing guarantee," in *IEEE International Conference on Automation Science & Engineering*, 2021.
[7] B. Xia, I. Mantegh, and W. Xie, "Integrated emergency self-landing method for autonomous uas in urban aerial mobility," in *IEEE International Conference on Control, Automation and Systems (ICCAS)*, 2021.
[8] S. J. Vine, L. Uiga, A. Lavric, L. J. Moore, K. Tsaneva-Atanasova, and M. R. Wilson, "Individual reactions to stress predict performance during a critical aviation incident," *Anxiety, Stress, & Coping*, vol. 28, no. 4, p. 467–477, 2015.
[9] "Artificial intelligence roadmap: a human-centric approach to AI in aviation," in *European Union Aviation Safety Agency (EASA)*, 2021.
[10] H. Wang, N. Jiang, T. Pan, H. Si, Y. Li, and W. Zou, "Cognitive Load Identification of Pilots Based on Physiological-Psychological Characteristics in Complex Environments," *Journal of Adv. Transportation*, 2020.
[11] M. Vukovic, M. Stolar, and M. Lech, "Cognitive load estimation from speech commands to simulated aircraft," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, 2021.
[12] C. Fernández-Morales, L. Espejo-Antúnez, V. J. Clemente-Suárez, F. B. Tabla-Hinojosa, and M. Albornoz-Cabello, "Analysis of heart rate variability during emergency flight simulator missions in fighter pilots," *BMJ Military Health*, 2022.
[13] D. McDuff, S. Gontarek, and R. Picard, "Remote measurement of cognitive stress via heart rate variability," in *IEEE Engineering in Medicine and Biology Society*, 2014.
[14] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *European Conference on Computer Vision*, 2020.
[15] E. C. Pinto Neto, D. M. Baum, J. R. de Almeida, J. B. Camargo, and P. S. Cugnasca, "A trajectory evaluation platform for urban air mobility (uam)," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 9136–9145, 2022.
[16] S. McGarry, A. Brown, M. Gardner, C. Plowright, R. Skou, and C. Thompson, "Immersive virtual reality: An effective strategy for reducing stress in young adults," *British Journal of Occupational Therapy*, vol. 86, no. 8, pp. 560–567, 2023.
[17] D. K. Panda and W. Guo, "Fragility impact of rl based advanced air mobility under gradient attacks and packet drop constraints," in *IEEE Vehicular Technology Conference*, 2023, pp. 1–7.



https://dspace.org/

Automating emergency landing in cities with cognitive stress reducing explanations to pilot-in-the-loop

Balakrishnan, Hamsa; Lechardoy, Bastien; Collignon, Maxime; Sen, Anwesha; Ali, Ahmed ...

2024-10-29
Institute of Electrical and Electronics Engineers (IEEE)
https://doi.org/10.1109/isc260477.2024.11004215
Balakrishnan H, Lechardoy B, Collignon M, et al., (2024) Automating emergency landing in cities with cognitive stress reducing explanations to pilot-in-the-loop. In: Proceedings of the 2024 IEEE International Smart Cities Conference (ISC2), 29 Oct - 1 Nov 2024, Pattaya, Thailand

Downloaded from a DSpace repository. https://dspace.org/