Authors' accepted manuscript
Paper presented at 32nd IEEE International Conference on Robot and Human Interactive Communication (IEEE RO-MAN).
Busan, Korea. Aug. 28-31, 2023.

The eyes and hearts of UAV pilots: observations of physiological responses in real-life scenarios

Alexandre Duval[1], Anita Paas[2], Abdalwhab Abdalwhab[1] and David St-Onge[1]

***Abstract*— The drone industry is diversifying and the number of pilots increasing rapidly. In this context, flight schools need adapted tools to train pilots, most importantly with regard to their own awareness of their physiological and cognitive limits. In civil and military aviation, pilots can train on realistic simulators to tune their reaction and reflexes, but also to gather data on their piloting behavior and physiological states, helping to improve their performance. As opposed to cockpit scenarios, drone teleoperation is conducted outdoors in the field, with only limited potential from desktop simulation training. This work aims to provide a solution to gather pilot behavior in the field and help them increase their performance. We combined advanced object detection from a frontal camera with gaze and heart rate variability measurements. We observed pilots and analyzed their behavior over three flight challenges. We believe this tool can support pilots both in their training and in their regular flight tasks.**

I. Introduction

The industry of teleoperated drones for service, such as in infrastructure inspection, crop monitoring, and cinematography, has expanded at least as fast as the technology that supports it over the past decade. However, in most countries the regulations are slow to adapt. Nevertheless, several regulating bodies have already recognized human factors as a core contributor to flight hazards. While core technical features of the aerial systems are evolving, namely autonomy, flight performance, and onboard sensing, the human factors related to UAV piloting remain under-researched. Using physiological measurements, including eye-based measures (changes in pupil diameter, gaze-based data, and blink rate), heart rate variability (HRV), and skin conductance, is one way of providing direct feedback on the pilots cognitive states [1]. Conventional aircraft pilots leverage this already in training, often referred to as Evidence-Based Training (EBT). In this context, it is widely accepted that physiological measurements provide objective data on pilot performance [2], [3], enabling trainers to identify areas for improvement and track progress over time. Additionally, physiological measurements can help trainers to develop individualized training programs [4] and assist pilots in stress management [5], [6]. Furthermore, EBT can improve safety by identifying potential safety concerns and inform training methods based on scientific evidence [7], [8].

*We thank NSERC USRA and Discovery programs for their financial support. We also acknowledge the support provided by Calcul Québec and Compute Canada and the work of Shouyue Bruce Hu on the code.

[1]Alexandre Duval, Abdalwhab Abdalwhab and David St-Onge are with the Lab INIT Robots, Department of Mechanical Engineering, Ecole de technologie supérieure, Canada name.surname@etsmtl.ca

[2]Anita Paas is with the Department of Psychology, Concordia University, Canada anita.paas@concordia.ca

However, EBT for UAV pilots needs to take into account differences in training environment, risk profile, mission requirements, and automation. Conventional aircraft pilots train in a physical cockpit environment, while UAV pilots may train in both virtual and physical environments [9], [10]. Conventional aircraft pilots operate in a highly regulated environment with established safety protocols, while UAV pilots may operate in more unstructured and less regulated environments. The mission requirements for conventional aircraft pilots and UAV pilots are also different, with UAV pilots responsible for a wider range of tasks, including surveillance, reconnaissance, and target acquisition [11], [12]. Finally, UAVs are typically more highly automated than conventional aircraft, which can affect the types of training required, such as human-automation interaction and decision-making [13], [14].

Eye-tracking glasses have been used to monitor pilot workload and to assess situational awareness in map-based command center of vehicles [15]. However, for UAV pilots, conventional methods can hardly be deployed: they rely on static scenes in front of the user and pre-defined region of interest (ROI). Our approach is to leverage object detection in the scene to infer gaze behavior related to these objects.

Multisensory configurations can be more robust to capture cognitive load. While each sensor is susceptible to some noise, these sources of noise do not overlap between sensors; for example, HRV is not influenced by luminance whereas pupil measurements are. We thus decided to combine these measurements in our use case.

Section II opens the path with an overview of the various inspirational domains to this work. We then build our solution on a biophysical capturing software (sec. III) and a detector trained on a custom dataset (sec. IV). Finally, we present the results of a small user study in sec. V and discuss our observations of the pilots' behaviors.

II. Related Works

A. On gaze-based behavioral studies

The benefit of eye tracking is that we can measure gaze behavior in addition to changes in pupil diameter. Gaze behavior can provide information about the most efficient way to scan and monitor multiple sources of input. For example, in surveillance tasks, operators monitoring several screens can be supported by systems that track gaze behavior and automatically notify the operator to adjust their scanning pattern [16]. In a simulated task, Veerabhadrappa, et al. [17] found that participants achieved higher performance on a simulated UAV refuelling task when they maintained a

longer gaze on the relevant region of interest compared to less relevant regions. Further, in training scenarios, gaze-based measures can identify operator attention allocation and quantify progress of novice operators [18]. Gaze-based measures of novices can also be compared with those of experts to determine training progress and ensure efficient use of gaze.

In a review paper focused on pilot gaze behavior, Ziv [19] found that expert pilots maintain a more balanced visual scanning. Expert pilots scan the environment more efficiently and spend less time on each instrument compared to novices. However, in complex situations, experts spend more time on the relevant instruments which enables them to make better decisions than novices. Overall, Ziv concluded that the differences in gaze behavior between expert and novice pilots are related to differences in flight performance.

B. On object detection

A plethora of models has been developed for object detection [20], both for two-stage detectors [21], [22] and single-stage detectors like YOLO [23] and others [24]. Generally speaking, two-stage detectors are more accurate but single-stage detectors are faster and more applicable in real-time applications. While these solutions are common for robotic perception [25] and static sensing [26], only a handful of works assess their potential for understanding user behavior.

Geert Brône, et al [27] combined gaze data with object detection for automatic gaze data analysis. [28] investigated the use of detection models to relate the gaze location to the visible human's head and hands during conversations.

Recently [29] trained classifiers to classify images cropped around the gaze location to annotate volume of interest. The issue with classifiers is that they only assign one class per image crop, missing the cases where there could be multiple objects in the same image crop whereas object detection identifies all objects in the image. Further, [29] also compared their classifier to an object detection model. Unlike these previous works, our work presents a model tailored and tuned to a realistic application and uses the tools to extract meaningful behavioral data of UAV pilots.

Closer to our work, Miller et al. [30] finetuned an InceptionV2 detection model to annotate objects of interest in eye-tracking video data and integrated it with body motion capture data. Then, they studied the distance from the gaze point to the target object in a target interception task. Similarly, our work uses a detection model to annotate objects in a video, but we have a different application scenario, we use biophysical measures and a different learning model, and we provide a more in-depth analysis of our user study.

III. Operator biophysics

Physiological signal capture and analysis are fundamental to human-centered robotic system design and validation. Several works demonstrate the relevance of these measurements, but each with its own implementation. It can then be time-consuming to reproduce any of these studies accurately and to deploy their tools and methods in other contexts.

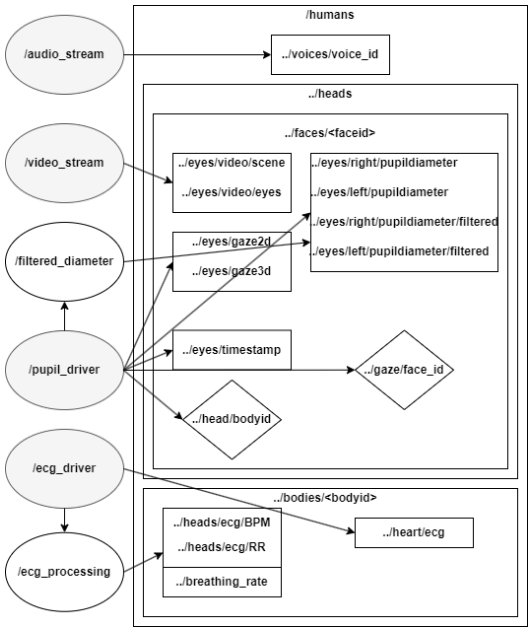

Fig. 1. Our contribution to the ROS4HRI pipeline. The input nodes (ellipses) are on the left. Each node publishes in a set of specific topics and geometric transforms (diamond) under the standard namespaces (faces and bodies), integrated with subcategories (eyes, ECG, breathing_rate).

In robotics, the Robot Operating System (ROS) positions itself as a standard to share and collaborate on code and software infrastructure. Its community is slowly integrating user-based measurements and wearable device drivers to cope with the reproducibility and sharing challenge of the Human-Robot Interaction (HRI) community [31]. A standard to include HRI considerations in ROS is currently in progress [32].

Our work aims to contribute to the community effort with the integration of a new sensor modality in the standard. The standard recommends splitting the human-related aspects into five sub-namespaces (faces, bodies, voices, persons and interactions). To properly integrate pupillometry data in that structure, we added the *eyes* category to the */human/faces/<face id>*. Eye data consist of gaze, pupil diameter, and blinks. The gaze messages (2D and 3D) are compose of a *geometry_msgs/Point* and two timestamps: one from ROS, the other from the device (if available). We keep both timestamps to ease network streaming issue debugging. The pupil messages (raw and filtered) publish the diameter value in millimeters as a *float64* alongside the same two timestamps. The pupil driver node considers a web socket connection, generally over WiFi, compatible with several brands, namely Tobii and Pupil Labs. Most commercial pupillometry devices also grant access to live stream of the eye cameras and the scene camera: a frontal camera giving a first person view of the scene as viewed by the user. Complementary nodes subscribes to these streams and converts them into ROS image (video_stream) and audio data (audio_stream) topics. If available, an inertial measurement unit can provide the head orientation, which we publish as a ROS geometric transformation (TF), alongside a TF for the gaze, following the TF tree in [32].

The second sensing modality is based on the electrocardiogram (ECG) signal. The driver provides the raw ECG values (in mV) in the */humans/bodies/<body id>* namespace, under a subcategory *heart*. We also provide more advanced features extracted live from the signal: RR intervals and beats per minutes. The breathing rate can also be extracted from the ECG signal, or, if available, from an accelerometer located on the chest. All these data are published as stamped *float64* ROS messages.

Figure 1 illustrates our contribution to the ROS4HRI pipeline [32], based on adding these two new sensing modalities. Each new user, or robot operator, is given a unique ID, based on the available modules to perform person identification. The current ROS4HRI pipeline [31] recommends using face recognition, but one could also leverage pupil shape as a unique biometric identifier using our contribution. The sensors reference frame can be linked to the body with geometrical transformation (TF) messages. All these values are published live when using the devices, but they can easily be recorded in a ROSbag for post-processing and data analysis. However, to monitor the data quality live and add some keypoints to the recordings, we also developed a Jupyter Notebook visualization interface. All code is publicly available online[1].

Live data processing

Most pupillometry devices provide gaze information in the eye camera frame as well as in the user space, following a calibration step using markers. However, pupil diameter and ECG cannot be used in their raw format to infer cognitive state. A first pass on the pupil value prevents propagation of any physical outliers ($9 > d(mm) > 2$). Then the pupil diameter data are processed over sliding windows of 1 second duration with:

$$d_{filtered} = d_{lp} - \tilde{d}_{base} \tag{1}$$

where d_{lp} is the raw value after a low pass filter was applied with cutoff frequency of 4Hz, and $\tilde{d}_{base}$ is the median of the baseline values. A baseline of at least 30 seconds must be recorded before running any experiment, for instance when the participant is undergoing the calibration process of the glasses.

As for the ECG data, we rely on a well maintained Python library popular among neuroscience academic projects, *NeuroKit2*. We provide it with the raw signal, in sliding windows of 30 seconds, and it includes the signal cleaning step and feature extraction. However, since several features may be selected by the developer depending on the study context, our node provides only RR interval, BPM outputs, and breathing rate extracted from the cleaned signal.

IV. Learn to see robots

A. Dataset building

Object detection is core to transferring the potential of gaze behavior analysis into UAV piloting tasks. Notwithstanding the accuracy of the existing models, it is known that a machine learning model is as good as the data used to train it. That dataset must contain enough images and instances for each class that the experiment targets. We collected our own dataset including four classes: UAV (drone), robot manipulator (arm), mobile robot base (rover) and remote controller.

TABLE I
Dataset Split Details : Number of Each Object in the Set

Class	No. Train. instances	No. Valid. instances	All
drone	425	116	541
robot arm	165	75	240
mobile robot	272	100	372
controller	217	77	294
Total	1079	368	1447

For the UAV, we collected images of a small indoor collision-resilient quadcopter designed by de Azambuja et al. [33]. For the robot arm, we picked several photos of the Gen3 lite robot arm from Kinova. For the mobile robot, we used images of the Clearpath Dingo-D. Lastly for the controller, we selected images of the PS4 Dualshock controller, one of the most common controllers in mobile robotics and the one provided with the Clearpath base.

We used a total of 909 images with a total of 1447 object instances. We used our lab's archives, we automated the download of relevant images from a search engine, we downloaded online public videos showing the objects and then manually extracted frames, and we took several new pictures and videos, mostly to for our custom UAV. We then annotate the images using the open-source tool LabelImg [34].

That number is still too low for training from deep neural net so we leveraged transfer learning from pre-trained models on the COCO dataset [35] then used our dataset for fine-tuning and validation.

For the dataset split, instead of directly splitting the image files for training and testing, we tried to split them based on the number of class instances. We aimed to maintain a percentage of around 80% of each class instance for training and around 20% for validation (some images may have 2 or 3 instances of an object and others may have 1 or 0) to have a better representation for each class. With this aim in mind, it was more convenient to use the traditional 80:20 split than K-fold cross-validation. This split resulted in 684 images for training and 225 images for validation. Table I represents the dataset and the split details.

B. Algorithm comparison

As mentioned in sec. II, YOLO is well known for its object detection speed and accuracy with several versions of this model to choose from: YOLOv1 to YOLOv7. Aiming at live detection eventually, we restrict our model selection to compare the YOLO algorithm family, considering that our model could be deployed in different use cases with huge differences in available computational power (e.g., a hand-held device, a lightweight end-user laptop, or a powerful

[1] https://git.initrobots.ca/aduval/bio_physics

TABLE II

ALGORITHM COMPARISON RESULTS: THE EVALUATION FOR EACH MODEL PRETRAINED IN MS COCO, FINE-TUNED, AND EVALUATED ON OUR COLLECTED DATASET. F1 MEASURE AND MEAN AVERAGE PRECISION (BETWEEN PRACTICES) ARE REPORTED FOR EACH MODEL FOR EACH CLASS

Class	V3T	V3D	V3SPP	V5	V7
all	0.938 (0.958)	**0.977** (0.981)	0.971 **(0.984)**	0.974 (0.973)	0.976 (0.982)
drone	0.915 (0.935)	**0.968** (0.969)	0.954 (0.966)	0.964 **(0.974)**	0.964 (0.968)
robot arm	0.966 (0.991)	**0.996** (0.995)	0.98 (0.994)	0.99 (0.995)	0.994 **(0.996)**
mobile robot	0.989 (0.995)	0.992 (0.995)	0.99 (0.995)	**0.998** (0.995)	0.997 **(0.996)**
controller	0.879 (0.911)	0.953 (0.964)	**0.958 (0.98)**	0.945 (0.929)	0.947 (0.966)

workstation). Thus we compared different versions of YOLO that require different computational power.

We first started with the evaluation of YOLOv3, YOLOv5, and YOLOv7 performances in our specific detection problem. For that, we selected three versions of YOLOv3: Tiny (V3T), default (V3D), and SPP (V3SPP), as well as the large version of YOLOv5 and YOLO7 largest model (e6e). Table II presents the results of the evaluation for each model pre-trained in MS COCO, fine-tuned, and evaluated on our collected dataset. Data augmentation was employed during the model training to increase the effective size of the dataset and enhance the model generalization. The training and validation were conducted on Compute Canada clusters [36]. For each model, we reported the F1 measure and mean average precision (between practices) at the intersection over union of 0.5.

Secondly, we needed to be able to use those trained detection models within ROS to fit into the pipeline detailed in sec. III and to align the detection information with the person's gaze information. To do that, we looked into the publicly available open-source ROS packages for YOLOv3 [37], YOLOv5 [38] and YOLOv7 [39] using the weights from our training. When running the feed video through each model live, we end up getting 8 FPS, 3 FPS, and 1 FPS for YOLOv3, YOLOv5, and YOLOv7 respectively. The tests were conducted on an Intel® Xeon(R) E-2276M CPU @ 2.80GHz × 12 CPU, with an NVIDIA Quadro P620 512 CUDA core GPU with 4 GB GDDR5, 16 GB DDR4 RAM running Ubuntu 20.04.4 LTS, and ROS Noetic.

C. Gaze to objects

Gaze tracking in a 3D environment is easy by itself, but combining gaze tracking on objects in a dynamic 3D space makes it more difficult. Detected objects from the video feed have bounding boxes around them. Adding the gaze coordinates based off the same image reference, we can deduce what the person is looking at. As shown in Figure 2, the gaze is inside a bounding box, which means the person is looking at that particular object, in this case, a PS4 controller.

To avoid any collisions, the person needs to sweep their gaze over the surrounding objects. To note if the person is looking at the action or not, we added a dynamic ROI (Region Of Interest) that stayed attached to the detected object in the scene. We can see the ROI in green in Figure 3. We used functions made by Dinu C. Gherman to define the ROI by giving points in an image.

Fig. 2. Gaze on a dualshock 4 controller. A person was asked to look at the remote controller for 10 seconds. In pink, we see the gaze. Each object in the video frame is bound and tagged with the class and confidence of prediction.

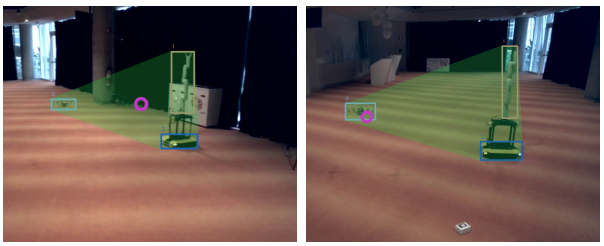

Fig. 3. The gaze position within the ROI (green). The ROI is dynamic to follow the moving objects. The gaze (pink) is moving in this area between the drone and the robot arm.

V. USER STUDY

We conducted a user study to investigate if we can detect changes in gaze behavior and heart rate variability due to task difficulty when people are flying a drone. We analyzed behavioral performance, SDNN (a measure of HRV), and eye gaze data. Data processing steps are discussed in Section 3. We expected that gaze behavior would be more stable and that HRV would decrease with increasing task difficulty. We also were interested in exploring differences in gaze behavior and HRV based on drone flight expertise, however we did not have specific hypotheses about those differences.

A. Method

The study was conducted indoors to maintain a more controlled environment. Figure 4 shows the map of the environment - a large and long room with a ceiling height of 3 meters. Participants (n = 8) piloted a drone around a stationary white 6 axis robot arm on top of a mobile robot (either stationary or moving, see conditions below and Fig. 3) with the goal of flying the drone around the arm, placed in

a vertical position, as many times as possible. Biophysical measurements, including pupil diameter, eye gaze, and heart rate (ECG), were recorded from each participant during each of the tasks. The study was divided into 5 tasks, with a duration of 1 minute each. The first two tasks were training tasks to familiarize the participant with the flying task and the environment, and to use as a baseline for comparison with the test tasks. In the first task, the participant piloted the drone freely with no obstacles around. In the second task, the participant piloted the drone around a static vertical Kinova robot arm that was affixed to a Clearpath Dingo-D robot. The 3 test tasks were as follows:

1) Pilot the drone around a vertical Kinova arm affixed to a Clearpath Dingo-D moving back and forth at 25 cm/s on a distance of 4 meters;
2) Piloting the drone around a vertical Kinova arm affixed to a Clearpath Dingo-D moving side to side at 25 cm/s on a distance of 4 meters;
3) Piloting the drone around a vertical Kinova arm affixed to a Clearpath Dingo-D moving in a slalom (sinusoidal) motion.

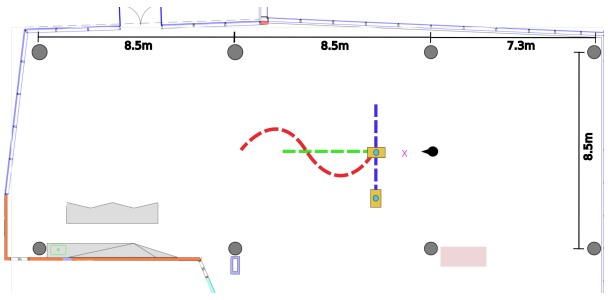

Fig. 4. Map of the indoor environment. The black dot represent the participant. The yellow rectangles with blue dot are the Dingo with the Kinova arm on top on their starting point. The blue line is the side to side path. The green line is the back and forth path. The red curve is the slalom path. The pink X represents the Cognifly drone.

Prior to the experiment, each user was asked about their experience with drone piloting. This needed to be considered because it can have a great impact on their performance. A person piloting a drone for the first time will not have the same reflexes or familiarity as someone with many hours of flight.

Before starting the tasks, the participant put on the Polar H10 heart rate chest band sensor under the shirt to record heart rate measures and a pair of Tobii Pro Glasses 3 eye-tracking glasses. Then, we explained the controls of the Cognifly drone [33]. Participants were instructed to stand still on a mark on the ground (black dot in Figure 4).

To establish a baseline of luminance for each of the areas participants would be looking at, we had the participant look for 10 seconds each at the remote controller held in their hands, the drone, the Dingo-D, and the Kinova robot arm. Once done, we began Task 1.

After completing Task 1 (freely flying the drone in the area), participants completed Task 2 (stationary condition). After Task 2, participants completed either Task 3 (back and forth motion) or Task 4 (side to side motion). Tasks 3 and 4 were counterbalanced, so all participants did both, but not in the same order. Finally, participants completed Task 5 (slalom motion). During the experiment, drone crashes occurred for some participants (n=3). There is no particular task that had more crashes than others. In the case of a crash, the task was restarted. Figure 5 shows the classifier distribution over all participants for the last three tasks: almost every frame had at least two objects detected, enough to compute a meaningful ROI.

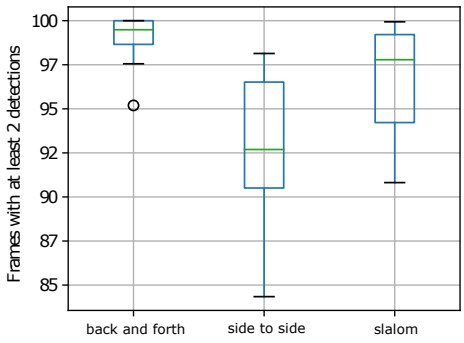

Fig. 5. Detector performance: percentage of frames with at least 2 objects detected out of 3 (arm, base, uav).

B. Results

On average, participants completed the most circles around the Kinova arm in the back and forth (Task 3) and side to side (Task 4) tasks, and the least in the slalom task (Task 5), suggesting that the slalom task was most challenging. Pupil diameter measures support this idea, as participants had the largest pupil diameter in the slalom task, on average (see Figure 6). Heart rate variability measures (SDNN) showed a small trend towards this pattern as well between the back and forth and slalom tasks.

We then split the data between expert and novice drone pilots, with experts having at least 15 hours of flight experience based on training classes offer by a drone school[2]. There were four experts and four novices with this division. For overall performance, experts outperformed novices in all tasks except the side to side task (see Figure 7). However, this could be due to additional task-specific training. Due to the counterbalancing (which did not take expertise into account), three out of the four novices performed the side to side task later in the experiment than most of the experts. The expert group maintained similar performance levels across tasks, while the novice group had a drop in performance for the slalom task.

The pupil measures differed slightly between the groups (see Figure 8), with expert pilots having the largest pupil diameter in the slalom task, followed by the back and forth task, and finally the side to side task. Novice pilots had the largest pupil diameter in the back and forth task, followed by slalom, and then the side to side task. This suggests that

[2]https://www.koptr.ca/reservation?lang=en

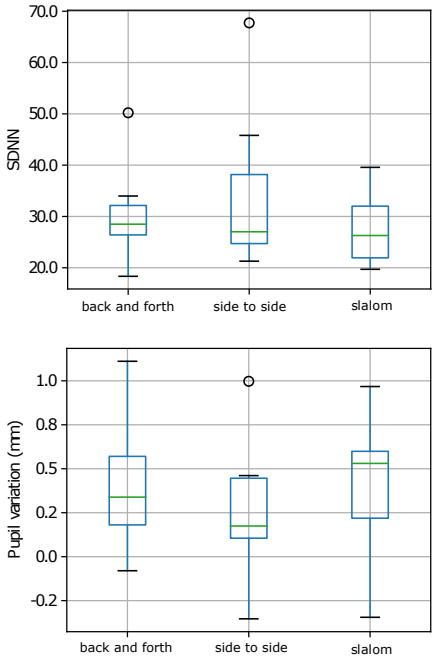

Fig. 6. Cognitive load metrics average for all participants by task.

cognitive load in expert pilots may have been highest in the slalom task, whereas in novice pilots, cognitive load may have been highest in the back and forth task.

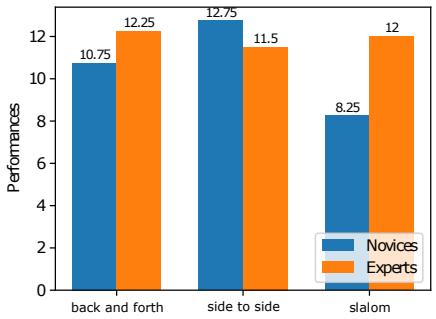

Fig. 7. Average number of circles around the obstacle (arm) by group for the final three tasks. All participants performed the slalom task last. Most novices performed the back and forth task before the side to side task, and the reverse for most experts.

For the heart rate variability measures, experts had higher SDNN for the back and forth task than for the side to side or slalom conditions. Novices had higher SDNN for the side to side task, less for the slalom task, and least for the back and forth task. As with the pupil and performance measures, this pattern of results shows a trend of familiarization, as each group experienced increased SDNN (an indicator of lower cognitive load) in the task most in the group performed fourth.

Finally, as seen in Figure 8, experts shifted their gaze more than novices, but their gaze was more focused and intentional (see heatmap in Figure 9). This may suggest that experts were more comfortable with piloting a drone and were able to monitor the area more efficiently to perform the tasks compared to novices.

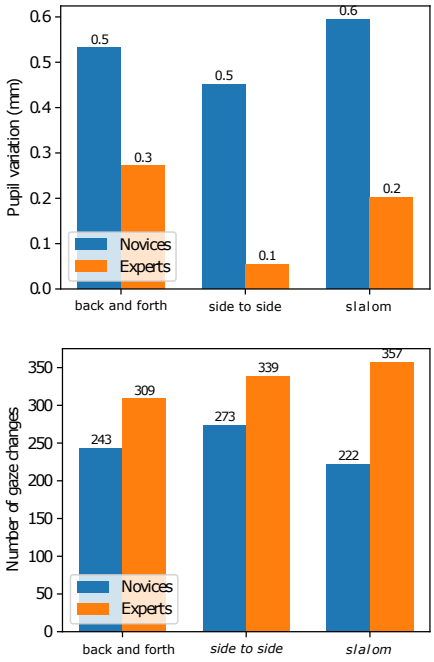

Fig. 8. Differences in pupil diameter variance (top) and gaze changes (bottom) between experts and novices.

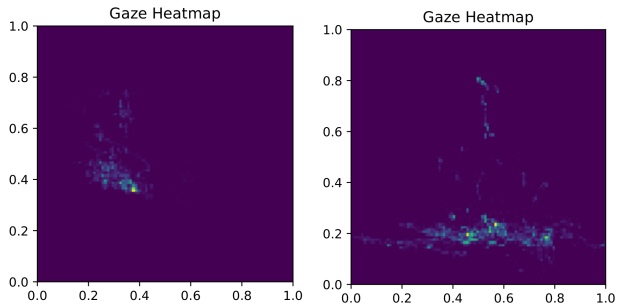

Fig. 9. The gaze heatmap (gaze spatial distribution) for the slalom task: left for an expert operator, and right, for a novice.

VI. Conclusion

In order to better understand how UAV pilots operate, we introduced a complete integration of pupil and heart rate features into the ROS4HRI pipeline. We presented a set of ROS nodes to capture and process live data from these sensors and we built a dataset to train a model for object detection on the pilot frontal camera feed. Our trained algorithm showed detection performance close to perfect. We then conducted a short user study, showing gaze behaviors that differ between expert and novice pilots and comparable cognitive load variation between tasks of different difficulty. Ultimately, we will extend this study to more participants and explore the relation with flight patterns. We will then explore how biofeedback can support novice pilots in their training by deploying our method as part of an official training in a flight school.

References

[1] J. Liu, A. Gardi, S. Ramasamy, Y. Lim, and R. Sabatini, "Cognitive pilot-aircraft interface for single-pilot operations," *Knowledge-*

Based Systems, vol. 112, pp. 37–53, 2016. [Online]. Available: http://dx.doi.org/10.1016/j.knosys.2016.08.031

[2] A. Alaimo, A. Esposito, C. Orlando, and A. Simoncini, "Aircraft pilots workload analysis: Heart rate variability objective measures and nasa-task load index subjective evaluation," *Aerospace*, vol. 7, no. 9, 2020. [Online]. Available: https://www.mdpi.com/2226-4310/7/9/137

[3] Y.-H. Lee and B.-S. Liu, "Inflight workload assessment: Comparison of subjective and physiological measurements," *Aviation, Space, and Environmental Medicine*, vol. 74, no. 10, pp. 1078–1084, 2003.

[4] B. MJ, S. SA, S.-C. K, C.-S. S, S. RO, and I. G., "Relative effectiveness of worker safety and health training methods," *American Journal of Public Health*, vol. 96, no. 2, pp. 315–324, 2006.

[5] "Heart Rate Variability Biofeedback Improves Emotional and Physical Health and Performance: A Systematic Review and Meta Analysis," *Applied Psychophysiology Biofeedback*, vol. 45, no. 3, pp. 109–129, 2020. [Online]. Available: https://doi.org/10.1007/s10484-020-09466-z

[6] P. S. Cowings, M. A. Kellar, R. A. Folen, W. B. Toscano, and J. D. Burge, "Autogenic feedback training exercise and pilot performance: Enhanced functioning under search-and-rescue flying conditions," *International Journal of Aviation Psychology*, vol. 11, no. 3, pp. 303–315, 2001.

[7] G. Borghini, L. Astolfi, G. Vecchiato, D. Mattia, and F. Babiloni, "Measuring neurophysiological signals in aircraft pilots and car drivers for the assessment of mental workload, fatigue and drowsiness," *Neuroscience & Biobehavioral Reviews*, vol. 44, pp. 58–75, 2014, applied Neuroscience: Models, methods, theories, reviews. A Society of Applied Neuroscience (SAN) special issue. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0149763412001704

[8] M. R. Endsley, "From Here to Autonomy: Lessons Learned from Human-Automation Research," *Human Factors*, vol. 59, no. 1, pp. 5–27, 2017.

[9] G. G. De la Torre, M. A. Ramallo, and E. Cervantes, "Workload perception in drone flight training simulators," *Computers in Human Behavior*, vol. 64, pp. 449–454, 2016. [Online]. Available: http://dx.doi.org/10.1016/j.chb.2016.07.040

[10] T. R. Bates, P. R. Lee, and S. Y. Kille, "Situational Awareness Training for Operators of Unmanned Aerial Vehicles," pp. 391–396, 2021. [Online]. Available: https://corescholar.libraries.wright.edu/isap_2021https://corescholar.libraries.wright.edu/isap_2021/65

[11] W. Budiharto, A. Chowanda, A. A. S. Gunawan, E. Irwansyah, and J. S. Suroso, "A review and progress of research on autonomous drone in agriculture, delivering items and geographical information systems (gis)," in *2019 2nd World Symposium on Communication Engineering (WSCE)*, 2019, pp. 205–209.

[12] H. Wang, H. Cheng, and H. Hao, "The use of unmanned aerial vehicle in military operations," in *Man-Machine-Environment System Engineering*, S. Long and B. S. Dhillon, Eds. Singapore: Springer Singapore, 2020, pp. 939–945.

[13] R. Parasuraman and D. H. Manzey, "Complacency and bias in human use of automation: an attentional integration," *Human factors*, vol. 52, no. 3, p. 381–410, 2010.

[14] C. Wickens, W. Helton, J. Hollands, and S. Banbury, *Engineering Psychology and Human Performance (5th ed.)*, 2021.

[15] J. T. Coyne and C. M. Sibley, "Impact of task load and gaze on situation awareness in unmanned aerial vehicle control," *18th International Symposium on Aviation Psychology*, pp. 458–463, 2015. [Online]. Available: \url{https://corescholar.libraries.wright.edu/isap_2015/29}

[16] A. Marois, D. Lafond, A. Williot, F. Vachon, and S. Tremblay, "Real-Time Gaze-Aware Cognitive Support System for Security Surveillance," *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 64, no. 1, pp. 1145–1149, 2020. [Online]. Available: https://doi.org/10.1177/1071181320641274

[17] R. Veerabhadrappa, I. Hettiarachchi, S. Hanoun, and D. Jia, "Identification and Evaluation of Effective Strategies in a Dynamic Visual Task Using Eye Gaze Dynamics," *preprint*, pp. 0–15, 2021. [Online]. Available: https://doi.org/10.21203/rs.3.rs-778091/v1

[18] M. A. Shahab, M. U. Iqbal, B. Srinivasan, and R. Srinivasan, "Metrics for objectively assessing operator training using eye gaze patterns," *Process Safety and Environmental Protection*, vol. 156, pp. 508–520, 2021. [Online]. Available: https://doi.org/10.1016/j.psep.2021.10.043

[19] G. Ziv, "Gaze Behavior and Visual Attention: A Review of Eye Tracking Studies in Aviation," *International Journal of Aviation Psychology*, vol. 26, no. 3-4, pp. 75–104, 2016. [Online]. Available: https://doi.org/10.1080/10508414.2017.1313096

[20] S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, and B. Lee, "A survey of modern deep learning based object detection models," *Digital Signal Processing*, p. 103514, 2022.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[22] S. Qiao, L. Chen, and A. L. Yuille, "Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution," *CoRR*, vol. abs/2006.02334, 2020. [Online]. Available: https://arxiv.org/abs/2006.02334

[23] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," 2022. [Online]. Available: https://arxiv.org/abs/2207.02696

[24] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 10 781–10 790.

[25] J. Ruiz-del-Solar, P. Loncomilla, and N. Soto, "A survey on deep learning methods for robot vision," *CoRR*, vol. abs/1803.10862, 2018. [Online]. Available: http://arxiv.org/abs/1803.10862

[26] Z. Zou, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," *arXiv preprint arXiv:1905.05055*, 2019.

[27] G. Brône, B. Oben, and T. Goedemé, "Towards a more effective method for analyzing mobile eye-tracking data: Integrating gaze data with object recognition algorithms," *PETMEI'11 - Proceedings of the 1st International Workshop on Pervasive Eye Tracking and Mobile Eye-Based Interaction*, pp. 53–56, 2011.

[28] T. Callemein, K. Van Beeck, G. Brône, and T. Goedemé, "Automated analysis of eye-tracker-based human-human interaction studies," in *International Conference on Information Science and Applications*. Springer, 2018, pp. 499–509.

[29] M. Barz and D. Sonntag, "Automatic visual attention detection for mobile eye tracking using pre-trained computer vision models and human gaze," *Sensors*, vol. 21, no. 12, p. 4143, 2021.

[30] H. L. Miller, I. Raphael Zurutuza, N. Fears, S. Polat, and R. Nielsen, "Post-processing integration and semi-automated analysis of eye-tracking and motion-capture data obtained in immersive virtual reality environments to measure visuomotor integration," in *ACM Symposium on Eye Tracking Research and Applications*, 2021, pp. 1–4.

[31] Y. Mohamed and S. Lemaignan, "ROS for Human-Robot Interaction," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Prague, Czech Republic: IEEE, 2021, pp. 3020–3027.

[32] "New rep proposal: Human-robot interaction in ros (ros4hri)," https://discourse.ros.org/t/new-rep-proposal-human-robot-interaction-in-ros-ros4hri/23776, accessed: 2022-08-24.

[33] R. de Azambuja, H. Fouad, Y. Bouteiller, C. Sol, and G. Beltrame, "When being soft makes you tough: A collision-resilient quadcopter inspired by arthropods' exoskeletons," in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 7854–7860.

[34] Tzutalin, "Labelimg," https://github.com/tzutalin/labelImg, 2015.

[35] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár, "Microsoft coco: Common objects in context," 2014. [Online]. Available: https://arxiv.org/abs/1405.0312

[36] S. Baldwin, "Compute canada: advancing computational research," in *Journal of Physics: Conference Series*, vol. 341, no. 1. IOP Publishing, 2012, p. 012001.

[37] M. Bjelonic, "YOLO ROS: Real-time object detection for ROS," https://github.com/leggedrobotics/darknet_ros, 2016–2018.

[38] V. Vasilopoulos and G. Pavlakos, "yolov3 pytorch Ros: Object detection for ROS using PyTorch," https://github.com/vvasilo/yolov3_pytorch_ros, 2019.

[39] L. Ewecker, "Ros package for official yolov7," https://github.com/lukazso/yolov7-ros, 2021.