Paired CNN for Successful Object Recognition and Obstacle Avoidance in UAVs

Raymond Dueñas
University of Colorado Colorado Springs
duenas2100@gmail.com

Adham Ayabi
University of Colorado Colorado Springs
aatyabi@uccs.edu

Abstract

Autonomous uncrewed areal vehicles require the ability to navigate various environments without collision failures. These systems already serve important roles in a variety of fields ranging from entertainment to military application. There is a desire to replace costly multi-sensor based systems with a system based solely on computer vision. However, these systems suffer from varying success rates in object recognition and obstacle avoidance. In some cases object recognition is to slow to avoid a collision failure. Current state-of-the-art solutions for computer vision based systems implement artificial neural networks or an algorithm written specifically for a particular task. This work proposes a paired convolutional neural network architecture for the execution of object recognition and obstacle avoidance in uncrewed areal vehicles. This work anticipates the paired convolutional neural network architecture to produce a vision based autonomous system with high success rates in obstacle avoidance and low rate of collision failures while maintaining competitive flight speed.

Introduction

Uncrewed arial vehicles (UAVs) have a wide range of applications, from children's toys to military operations. One such environment is UAV racing as a sport which has produced championship UAV pilots that can fly UAVs through obstacle courses performing maneuvers including corkscrews, loops, suicide dives, and reaching speeds up to 120mph. There is a high demand for advancement in developing an autonomous UAV (AUAV) system with the ability to navigate through obstacles, avoid collisions, and reliably execute objectives. The UAV maneuverability achieved by champion UAV racing pilots serves as a benchmark for AUAVs, a benchmark with an extreme gap in performance. Lockheed Martin and CEO of The Drone Racing League, Nicholas Horbaczewski shared a vision of leveraging the sport to ramp up advancements in AUAV technology. This vision brought about the foundation of Lockheed Martin's AlphaPilot Innovation Challenge which challenges participants to design AUAV capable of piloting through professional drone racing courses. The metrics used to determine the winning AUAV are, firstly, the percentage of course completed, and secondly, the speed of completion. The AlphPilot Innovation Challenge finalist competed at Artificial Intelligence Robotic Racing (AIRR) World Championship. At this event, of the nine finalists, only two AUAVs successfully completed the course, a failure rate of 78%. Of the two AUAVs that completed the course, the winner had an average speed of 1.5m/s. The architecture utilized by this AUAV serves as a benchmark for this work. The Game of Drones is another drone racing platform and is where the second benchmark for this work is derived. A Microsoft initiative working to close the gap between AUAVs and piloted UAVs The Game of Drones runs on the AirSim virtual platform developed by Microsoft and Stanford. Teams load there computer vision-based navigation models onto a virtual drone that then uses the model to traverse a rigorous virtual course. Winning requires completing more gates than any other team or completing the same number of gates with a faster track time. This work utilizes recent advances in lightweight visual transformers. Applying the visual transformer model OFP a sequence to sequence framework presented by (Wang et al. 2022) in conjunction with the Separable Pyramidal Pooling EncordEr-Decoder (SPEED) presented by (Papa et al. 2022) for depth perception and object avoidance guided controllers based on (Zhang et al. 2020) work with monocular trajectory planning. Combining the three listed methods will yield a general solution for the computer vision based AUAV navigation situation. Providing enhanced object detection, route navigation, collision avoidance, and an increased success rate of monocular AUAV flight.

Related Work

Autonomous flight requires the successful syntheses of multiple dynamic objectives and systems. In this work we focus in on object detection, route estimation or obstacle avoidance and depth estimation. The following related works represent the top performing AUAV systems from three separate competitions and presents the methods utilized by each team with respect to the noted systems of focus for this work.

UZH Robotics and Perception Group: Optimal Methods meet Deep Learning for Autonomous Drone Racing

Object detection: Utilizing A deep network the team first delivers an input image to a shallow DroNet architecture based Convolutional Neural Network, the outputted caricaturists are then handled by two individual multilayer perceptrons.

Collision avoidance and Rout Estimation: Utilizing a successive two stage system, first a waypoint is derived from gate estimated location and a favorable path is chosen. In the next stage the onboard controller is relayed directions to navigate to the waypoint and flight path is tracked for improved stabilization between waypoints.

Depth Estimation: This model calculates the deep network derived regression of the input RGB image's mean in order to determine distance to a gate.

Sejong University: Report for Game of Drones A NeurIPS 2019 Competition

Object detection:: Using a Neural Network as an object detection model. The team implements U-Net segmentation an actor net and a critic net in the process of training the neural network. In order to develop a reward based guidance for navigation decisions derived from the initial detection of a gate and current estimated AUAV location.

Collision avoidance and Rout Estimation: Developing and implementing a rule-based control scheme called moveBySplineAsnc and moveOnSplineAsnc. The control scheme was trained by running the actor net through a virtual gate from a number of approach trajectories and presenting it with a risk reward value depending on if it makes it through the gate or suffers a collision failure while the segmentation compressed and preserved valuable data gained from the simulation.

Depth Estimation: (At this point it is unclear to me how they handle depth estimation. I believe it is derived from data gained from their object detection model)

MAVLab: A Computationally Efficient Vision-Based Navigation And Control Strategy

Object detection: The system utilizes deep learning based optic flow and algorithms they call Snake gate detection and Histogram gate algorithm which were uniquely designed for the competition.

Collision avoidance and Rout Estimation: Utilizing a PD controller and the Snake gate detection algorithm the AUAV is centered to the gate when ever there is a positive reading of a gate in view. To compensate for situations when no gate is in view the team implements a state estimator arc to turn the drone in the direction of the next gate.

Depth Estimation: Provided attitude estimate to the Snake Gate algorithm developed, also allows for depth estimate.

MAVLab, the winners of the AlphPilot Innovation Challenge, lay out the process of developing their benchmark AUAV in (Li et al. 2020) . A detailed system overview establishes that the system hardware utilized consists of a camera with six optical elements and 14 Megapixels sensor, Parrot p7 dual-core CPU cortex 9 (max 2GHZ), an MPU 6050 IMU and sonar with less than 8m range. Their AUAV utilizes a novel snake gate detection algorithm to identify and a PD controller to steer the drone to the center of the detectable rectangular-shaped gates. Utilizing classic complementary filter for attitude and heading reference systems (AHRS) then using Kalman filter to fuse AHRS and IMU measurements to estimate position. The system implemented a prediction-based feed-forward control scheme when the steer when snake gate detection algorithm does not detect a gate. Lastly, as a low-level attitude controller, their system employed an adaptive incremental nonlinear dynamic inversion (INDI). Utilizing the described AUAV was able to navigate a course at an average speed of 1.5m/s. However, (Li et al. 2020) explains that there are failure cases where the drone crashes into the gate due to late gate detection and complete detection failures.

Problem Statement

Working to develop a robust AUAV is a global effort. Currently, the success rate in developing an AUAV system that can navigate a course without crashing is at most 22% when considering the AIRR world championships, which–within the scope of the competition–consist of the most competitive AUAVs yet to be developed. This work aims to produce a general AUAV system that efficiently detects and classifies objects, develops superior navigational routes, and significantly reduces collision rates, all while increasing the AUAV's rate of travel. Lastly, the systems utilized by the benchmark AUAV employed a visual navigation algorithm specific to rectangular gate detection and will fail if the gate shape is changed. The successful execution of this work will produce a general system that does not rely on a mission specific algorithm, and instead accepts situational parameters to yield a significant gain in the operational scope of the AUAV.

Approach

Proposed System

To handle object detection, the lightweight vision transformer, OFA, which through a sequence-to-sequence learning framework, can perform vision and language tasks with state-of-the-art accuracy and competitive speed presented by (Wang et al. 2022) will be implemented. This transformer detects, classifies, provides objects in frame location, and can perform impressive image infilling. If this transformer can be implemented onto memory-constrained drones and tuned to increase throughput, these capabilities would provide a highly effective object detection model. In managing depth estimation and obstacle avoidance issues associated with monocular vision, this work proposes utilizing a paired CNN architecture, an overview of which can be seen in figure 1. The first CNN takes in a sequence of two one-dimensional frames that have been combined into one two denominational array and will produce optical flow-related values. The second CNN takes in the values produced by the first CNN and three-dimensional destination coordinate from which it outputs an optimal directional decision. The proposed CNN architecture aims to be as accurate as the RT-ViT model developed by (Ibrahem, Salem, and Kang 2022) and as fast as the SPEED model developed by (Papa et al. 2022). RT-ViT addresses depth estimation

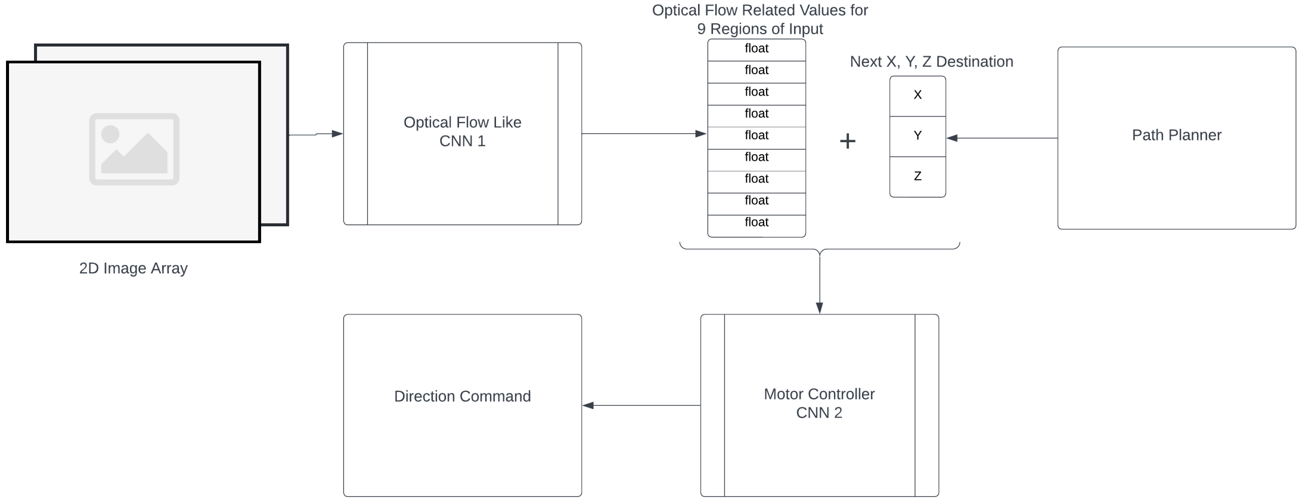


Figure 1: Paired CNN Architecture

in real-time situations when depth estimation must be conducted with only monocular data. The performance of RT-ViT, reached state-of-the-art accuracy on multiple data sets, including NYU-depthv2 and CITYSCAPES.
However, the fastest RT-ViT model, ViT-t16+DE, had a maximum frame rate of 20.83. SPEED addresses collision avoidance in real-time situations, with only monocular data available. The performance of SPEED is better than other fast throughput architectures, even on low-resource settings (Papa et al. 2022). The SPEED model utilizes two depth-wise separable pyramidal pooling layers, increasing the inference frequency and reducing computational complexity. Utilizing NYU Depth v2 and DIML Kinectv2 datasets to benchmark monocular depth estimation. SPEED achieves state-of-the-art results for fast throughput compared with related works on the DIML Kinect v2 data set and outstanding results in error estimation compared to more complex models. When presented in 2019, SPEED's performance on the NYU Depth v2 data set was near the state-of-the-art at the time(Papa et al. 2022).

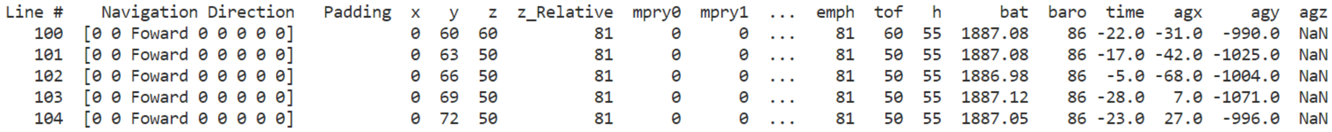
Furthermore, the obstacle avoidance scheme proposed by (Zhang et al. 2020) will be employed to provide reliable route estimation. The scheme will be implemented to develop the three-dimensional coordinates, which will be passed to the second CNN in the proposed paired CNN architecture. In their work (Zhang et al. 2020) states that reliable collision prevention estimation is unattainable with monocular data alone. Their work proposes an obstacle collision avoidance trajectory planning scheme as an alternative to collision prevention. Considering the characteristics of monocular optical measurement, they utilize two obstacle localization models based on relative range and relative angle. This model enhances the capability of AUAVs to avoid collision trajectories and achieve favorable results when compared with methods capable of geometric collision avoidance utilizing global knowledge.

Measuring Results

Environmental constraints make reproducing the MAVLab benchmark AUAV experiment unfeasible in this work. However, utilizing The Tello EDU model number: TLW004, this work will fit the benchmark model as tight as possible to the TLW004. Available specifications show that the TLW004 is equipped with 720p HD transmission, 5MP photos, FOV: 82.6, video: HD720P30, Intel processor, range finder, and barometer. After fitting the benchmark model to the available TLW004 the MAVLab metric tests will be run to establish a benchmark figure running MAVLab system on the TLW004. Once the benchmark has been established, the system proposed in this work will be fitted to the TLW004 and the tests will be repeated. Taking the percentage of course completed, average observed speed, gate detection hit/miss rate, and error distribution between estimated states and ground-truth states as metrics to compare the results of the proposed system with the established benchmark.

Microsoft's drone racing simulator, AirSim, will be employed in measuring this works proposed system against the top-performing system utilized by Sejoung University. Sejoung University provides metrics for the performance of their AUAV system in (Shin, Kang, and Kim). Loading the proposed work into the AirSim testing the performance of the system and measuring results with respect to the metrics provided by (Shin, Kang, and Kim) will allow for determination of the performance of the proposed system against that of the top-performing Sejoung University system.

After metric comparisons are complete, to test the scope of the proposed system's operational environment, the AUAV will navigate through three additional variations of



```
Line #    Navigation Direction    Padding  x   y   z  z_Relative  mpry0  mpry1  ...  emph  tof   h      bat  baro  time   agx     agy  agz
   100  [0 0 Foward 0 0 0 0 0]             0  60  60          81      0      0  ...    81   60  55  1887.08    86 -22.0 -31.0  -990.0  NaN
   101  [0 0 Foward 0 0 0 0 0]             0  63  50          81      0      0  ...    81   50  55  1887.08    86 -17.0 -42.0 -1025.0  NaN
   102  [0 0 Foward 0 0 0 0 0]             0  66  50          81      0      0  ...    81   50  55  1886.98    86  -5.0 -68.0 -1004.0  NaN
   103  [0 0 Foward 0 0 0 0 0]             0  69  50          81      0      0  ...    81   50  55  1887.12    86 -28.0   7.0 -1071.0  NaN
   104  [0 0 Foward 0 0 0 0 0]             0  72  50          81      0      0  ...    81   50  55  1887.05    86 -23.0  27.0  -996.0  NaN
```

Figure 2: Example of drone state data collected.

a test course. The first variation will replace all rectangular gates with circular ones. The second will replace all circular gates with rectangular obstacles that must be maneuvered around to avoid a collision. Lastly, the course will combine rectangular gates, circular gates, rectangular obstacles, and circular obstacles. The successful completion of these differing environments will demonstrate a degree of the scope for the operational environment provided by the proposed AUAV system. Possible data sets for this work include Wild-UAV, EuRoC MAV, TUM monoVO, NYU Depth V1/V2, RGB+D, PASCAL VOC12, MS COCO, ImageNet, Open Images V6, and a self-derived data set for control outputs.

Experiments and Results

Data Set

TThe development of a data set was necessarily added to the scope of this work to experiment and train the proposed paired CNN architecture. Data was collected by flying the Tello drone indoors in both congested and clear environments. The video feed was recorded at a rate of 30 frames per second and stored as an avi file for each session. During each flight session, the drone state information was also recorded at a rate of 30 states per second. This information includes drones x, y, z, acceleration values relative height, and more. An example of this data can be seen in figure 2. While the drone state information includes the drone's height, z coordinate, it does not include x and y coordinates. Using the drone's tunable travel rate in centimeters per second measurements were conducted to calibrate the drones in-flight x and y coordinates relative to its initial hover position after take-off. This information was added to the drone's state data and can be seen in figure 2 labeled as x, y, z, and z-relative. Z-relative is the drone's height relative to anything directly beneath it, while z is in reference to initial take-off. With this information, odometry maps were developed and stored for reference as visual representations of the drone's traveled path. Lastly, after flight sessions were completed, a folder for each flight containing the individual frames from each session recording was created. From these frame files, a data set was generated containing the optical flow information pairs of frames. The results for each optical flow calculation were further processed, parsing each result into nine non-lapping regions, each of which was reduced to a single floating point value. The frames utilized in the optical flow operation and nine region representing floating point values generated are aligned in the data set generated. An example of this data and a visual representation of the nine optical flow regions can be seen in Figures 3 and 4.

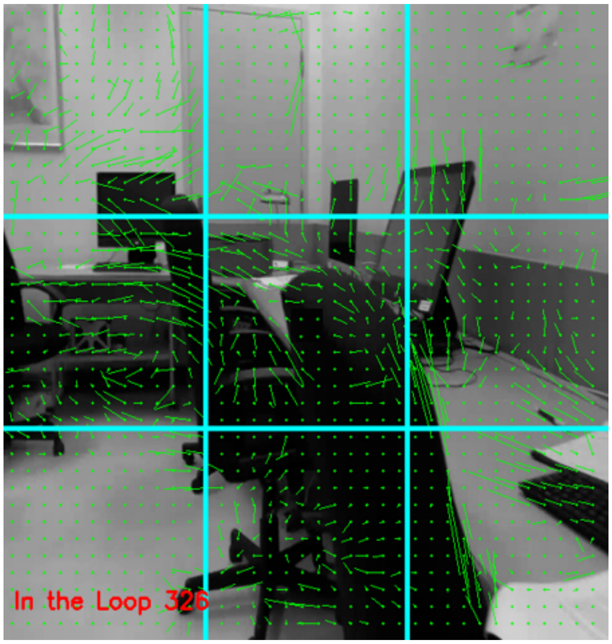


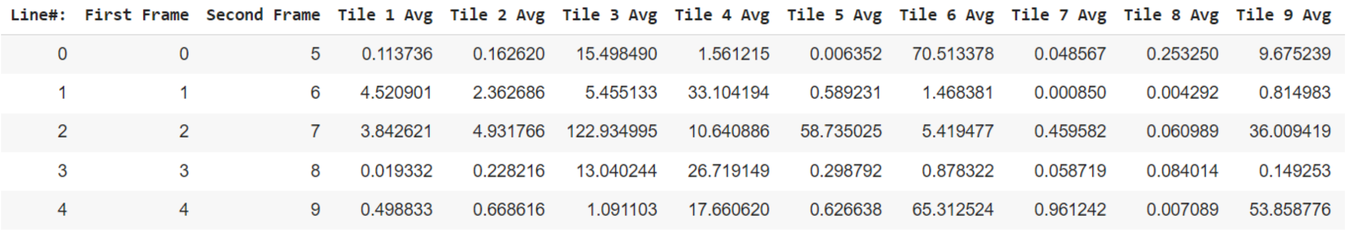
Figure 3: Visualisation of the nine optical flow region.

Model Development

The development of the paired CNN architecture has been split into two phases. The first of which develops the CNN in charge of taking in two sequential one-dimensional images paired together as one two-dimensional array and generating a nine-value representation of optical flow for the inputted sequence. After completion of this model, the second phase of model development would begin in which the model outputs drone motor control commands from the nine outputs of the first CNN and three denominational destination coordinate.

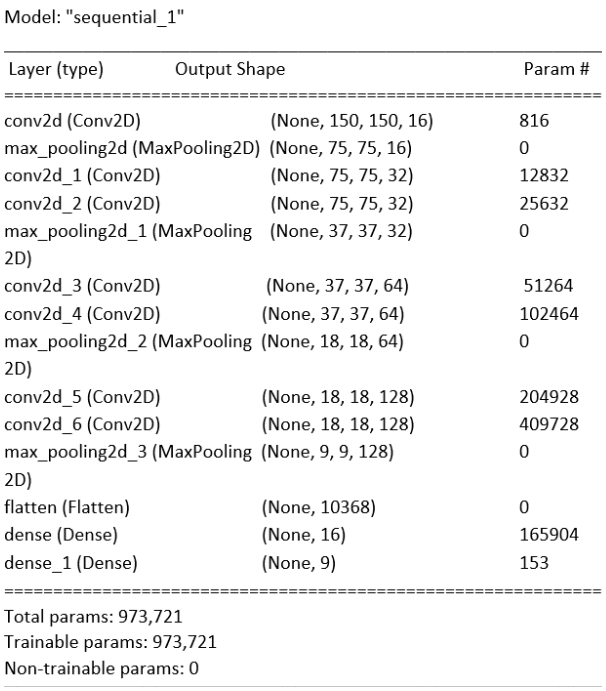
Results

In phase one, training the CNN took place in Colab Pro+ utilizing, Tensorflow, Keras, and Sklearn. The model would be required to take in an image as a variable and predict nine continuous values. From this, it was determined that this was a regression model task. As such mean squared error was implemented as the loss function, and mean absolute error was implemented as the metric for measuring the prediction error of the model. Experimenting with multiple architectures variations of the regression model and utilizing K-Fold cross-validation to provide the entirety of the data set, the



Line#:	First Frame	Second Frame	Tile 1 Avg	Tile 2 Avg	Tile 3 Avg	Tile 4 Avg	Tile 5 Avg	Tile 6 Avg	Tile 7 Avg	Tile 8 Avg	Tile 9 Avg
0	0	5	0.113736	0.162620	15.498490	1.561215	0.006352	70.513378	0.048567	0.253250	9.675239
1	1	6	4.520901	2.362686	5.455133	33.104194	0.589231	1.468381	0.000850	0.004292	0.814983
2	2	7	3.842621	4.931766	122.934995	10.640886	58.735025	5.419477	0.459582	0.060989	36.009419
3	3	8	0.019332	0.228216	13.040244	26.719149	0.298792	0.878322	0.058719	0.084014	0.149253
4	4	9	0.498833	0.668616	1.091103	17.660620	0.626638	65.312524	0.961242	0.007089	53.858776

Figure 4: Example of Optical flow values representing nine regions of optical flow.



```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)            Output Shape                  Param #
=================================================================
conv2d (Conv2D)                  (None, 150, 150, 16)      816
max_pooling2d (MaxPooling2D)     (None, 75, 75, 16)        0
conv2d_1 (Conv2D)                (None, 75, 75, 32)        12832
conv2d_2 (Conv2D)                (None, 75, 75, 32)        25632
max_pooling2d_1 (MaxPooling      (None, 37, 37, 32)        0
2D)
conv2d_3 (Conv2D)                (None, 37, 37, 64)         51264
conv2d_4 (Conv2D)                (None, 37, 37, 64)        102464
max_pooling2d_2 (MaxPooling      (None, 18, 18, 64)        0
2D)
conv2d_5 (Conv2D)                (None, 18, 18, 128)       204928
conv2d_6 (Conv2D)                (None, 18, 18, 128)       409728
max_pooling2d_3 (MaxPooling      (None, 9, 9, 128)         0
2D)
flatten (Flatten)                (None, 10368)             0
dense (Dense)                    (None, 16)                165904
dense_1 (Dense)                  (None, 9)                 153
=================================================================
Total params: 973,721
Trainable params: 973,721
Non-trainable params: 0
_________________________________________________________________
```

Figure 5: Architecture of optical flow value generating model.

lowest mean absolute error observed was 6.1520. Meaning that, on average, the model is 6.1520 units away from the correct prediction. The model architecture used to produce these results can be seen in figure 5.

Conclusion

The general solution presented in this work utilized the visual transformer model OFA and the proposed paired CNN architecture in conjunction with trajectory modeled for depth perception, obstacle avoidance, and motor controllers. The implementation of the first phase in model development has produced a model with prediction error that encourages improvement. Future work will include the implementation of phase two of model development, creating a model capable of producing optimal motor control commands towards its given destination. Completion of model development will prompt the highly anticipated testing of the proposed general solution against the benchmark AUAV models with respect to the defined metrics. Given the test results, further work to improve the model or further testing for robustness may be implemented..

References

Ibrahem, H.; Salem, A.; and Kang, H.-S. 2022. Rt-vit: Real-time monocular depth estimation using lightweight vision transformers. *Sensors* 22(10).

Li, S.; Ozo, M. M.; De Wagter, C.; and de Croon, G. C. 2020. Autonomous drone race: A computationally efficient vision-based navigation and control strategy. *Robotics and Autonomous Systems* 133:103621.

Papa, L.; Alati, E.; Russo, P.; and Amerini, I. 2022. Speed: Separable pyramidal pooling encoder-decoder for real-time monocular depth estimation on low-resource settings. *IEEE Access* 10:44881–44890.

Shin, S.-Y.; Kang, Y.-W.; and Kim, Y.-G. Report for game of drones: A neurips 2019 competition.

Wang, P.; Yang, A.; Men, R.; Lin, J.; Bai, S.; Li, Z.; Ma, J.; Zhou, C.; Zhou, J.; and Yang, H. 2022. Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. *arXiv preprint arXiv:2202.03052*.

Zhang, Z.; Cao, Y.; Ding, M.; Zhuang, L.; and Tao, J. 2020. Monocular vision based obstacle avoidance trajectory planning for unmanned aerial vehicle. *Aerospace Science and Technology* 106:106199.