# Comparison of Object Detectors for Fully Autonomous Aerial Systems Performance

Bowen Li
Ohio State University
USA
li.7652@osu.edu

Nat Shineman
Ohio State University
USA
shineman.5@osu.edu

Jayson Boubin
Ohio State University
USA
boubin.2@osu.edu

Christopher Stewart
Ohio State University
USA
cstewart@cse.ohio-state.edu

## CCS Concepts

**• Computer systems organization → Robotics**; **• Computing methodologies → Computer vision**; **• General and reference → Performance**.

## Keywords

UAV, neural networks, performance, object detection

**ACM Reference Format:**
Bowen Li, Nat Shineman, Jayson Boubin, and Christopher Stewart. 2021. Comparison of Object Detectors for Fully Autonomous Aerial Systems Performance. In *Companion of the 2021 ACM/SPEC International Conference on Performance Engineering (ICPE '21 Companion), April 19–23, 2021, Virtual Event, France.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3447545.3451170

## 1 Introduction

Unmanned aerial vehicles (UAVs) are gaining popularity in many governmental and civilian sectors. The combination of aerial mobility and data sensing capabilities facilitates previously impossible workloads. In aerial surveillance, search and rescue, and crop scouting, UAVs can access vast, high, and unsafe places to sense and relay photos to operators in real time [1, 10, 12]. These photos can then be processed for target search, localization, and tracking [8].

UAVs are normally piloted by remote operators who determine where to fly and when to sense data. Operations over large areas put a heavy burden on human pilots (e.g., difficult flight route management, repetition of waypoints) which lead to inefficiencies. Fully autonomous aerial systems (FAAS) have emerged as an alternative to human piloting [4]. FAAS software combines UAVs with edge and cloud hardware to execute autonomous missions, dynamically setting waypoints based on mission goals [2]. FAAS do not require human piloting, but do require considerable software support.

FAAS middleware manages object detection, pathfinding, UAV flight control and data sensing, along with other tasks specific to the application domain [12]. The compute and networking infrastructure required for these tasks has significant power and performance demands. FAAS deployed in remote environments, such as crop fields, must manage limited power and networking capabilities. Furthermore, UAV controlled by FAAS also have significant power demands. FAAS middleware must complete tasks quickly– any time spent on object detection or pathfinding while UAV are in flight wastes UAV battery power. To facilitate widespread adoption of FAAS, middleware must support heterogeneous compute and networking resources at the edge while ensuring that the workloads quickly produce effective and efficient autonomous flight paths.

SoftwarePilot [3] is an open source FAAS middleware. SoftwarePilot provides UAV flight control, data management, and machine learning routines that FAAS users can install on consumer hardware and use with DJI [5] UAVs. SoftwarePilot provides novel reinforcement-learning based FAAS pathfinding algorithms to execute crop scouting and autonomous photography routines. Object detectors are the backbone of these techniques, providing relevant information that pathfinding can leverage to complete missions.

In this poster, we analyze the performance of different object detection techniques recently implemented in SoftwarePilot for facial recognition. Facial recognition is an important task for FAAS, especially in autonomous photography. We analyzed the accuracy and performance of three facial recognition techniques provided in SoftwarePilot on two architectural configurations for FAAS edge systems. These findings can be used when selecting an object detector for any FAAS mission type and hardware configuration.

## 2 Design

SoftwarePilot is divided into routines and drivers [3]. Drivers group implementations of core FAAS functionality into API calls. SoftwarePilot includes drivers for UAV flight control, sensing, data management, and machine learning. Routines are SoftwarePilot applications. Routines group programming logic in Java with driver calls to create fully autonomous aerial missions. SoftwarePilot's Vision driver includes deep convolutional neural network (DCNN), Histogram of Oriented Gradients + support vector machine (HOG+SVM), and single shot detector (SSD) models for recognition.

SoftwarePilot has previously been used to benchmark autonomous photography for facial recognition [4] using HOG+SVM and DCNN models common FAAS edge architectures. Dlib's [7] Facial recognition HOG+SVM model combines two popular object detection


ACM ISBN 978-1-4503-8331-8/21/04.
https://doi.org/10.1145/3447545.3451170

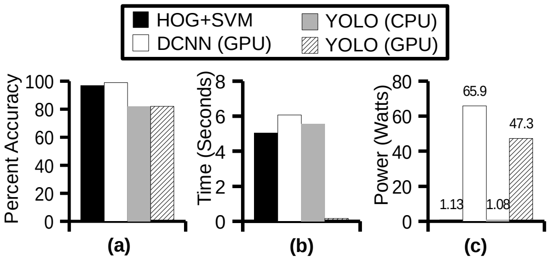


**Fig. 1: Accuracy, inference time, and power consumption for our 4 candidate models**

methods to create a highly accurate CPU-only model for facial recognition. In contrast, Dlib's DCNN model achieves even greater accuracy, but with a CPU inference time far above the latency requirements for FAAS. For this reason, highly accurate DCNN models are only recommended to run with a GPU-provisioned edge system. While these models are effective and highly accurate, their inference times on edge hardware can be punishing to UAVs awaiting critical decisions. Prior work has not yet explored the effectiveness of SSDs in FAAS applications.

SSDs, like YOLOv3 [9], use shallow neural network architectures and logistic regression to predict bounding boxes for objects within an image in a single pass. This results in a fast inference process that is highly parallelizable and capable of high accuracy. YOLOv3 is therefore a candidate for object detection on both highly and modestly provisioned edge architectures. In contrast, Dlib's HOG+SVM and DCNN models are exclusive to different architectures. HOG+SVM is not CUDA compatible, and DCNN's non-accelerated inference time is prohibitively high for FAAS.

Published accuracy statistics for object detectors may not be good predictors of success for FAAS. UAV must maintain safe distances from their targets, so the images they capture must be high-resolution, and targets may be small compared to conventional datasets. It is important to test and retrain detectors for use with FAAS. In the next section, we analyze the effectiveness of Dlib HOG+SVM and DCNN models compared to YOLOv3 on a highly provisioned edge system on an FAAS facial recognition data set.

## 3 Early Results

To test the effectiveness of various object detectors on FAAS tasks, we ran our three candidate models on a highly provisioned edge system using a dataset of labeled images from real FAAS missions. Our edge system runs Ubuntu 20.04 Linux with an 8-core i7 10700K CPU, 32GB of ram, and an NVIDIA GTX 1080 GPU. We tested the Dlib HOG+SVM and DCNN facial recognition classifiers, and YOLOv3 trained on the Wider Face [11] dataset. For testing we used 500 4k images captured with SoftwarePilot. We randomly selected 250 images containing one or more faces, and 250 images containing no faces.

Figure 1 shows the performance of these models on our dataset. Figure 1 (a) shows that HOG+SVM and DCNN both have incredibly high accuracy, 97% and 99% respectively, on our dataset of FAAS facial images. HOG+SVM and DCNN had precision and recall rates of (0.96, 0.98) and (0.98, 1.0) respectively. In contrast, YOLOv3 was 82% accurate, with precision and recall rates of (0.94, 0.66).

YOLOv3 outperforms in terms of inference time. As shown in figure 1 (b), HOG+SVM and YOLOv3 when executed on the CPU had similar inference times: 5.05s and 5.56s respectively. When executed on GPU, YOLOv3 had an inference time of 0.16s (6 frames per second) when compared to DCNN's 6.78s inference time. GPUs tax edge energy. Figure 1 (c) shows that CPU models consume around 1 Watt above idle power. When a GPU is added, Dlib's DCNN consumes 65.9W, and YOLOv3 consumes 47.3W over idle. The cost of YOLOv3, however, is further degraded from Dlib because of its far lower inference time.

YOLOv3 as trained is less accurate than either prior approach. We can see from precision and recall that inference inaccuracy is a product of false negatives, indicating that YOLOv3 misses distant and harder to find faces in the high resolution UAV images. While YOLOv3 is relatively inaccurate, its inference time is incredibly low and its inaccuracies stem primarily from distant or small targets in images. Scenarios that do not demand high accuracy but require high throughput, like tracking a close target or locating distant but large targets may benefit from YOLOv3.

Model selection for FAAS requires considerable future work. Adaptive model switching, hardware duty-cycling seen and embedded onboard computation using FPGAs and the NVIDIA Jetson, and other edge hardware configurations suggested in prior work should be tested [4, 6]. We plan to address these opportunities in future work using a wide arrange of FAAS collected data from multiple domains.

**Acknowledgments:** This research was funded in part by NSF Grants 1749501 and 1350941 with support from NSF CENTRA collaborations (grant 1550126), and the NSF Graduate Research fellowship (DGE-1343012).

## References

[1] Andreas Birk, Burkhard Wiggerich, Heiko Bülow, Max Pfingsthorn, and Sö Schwertfeger. 2011. Safety, Security, and Rescue Missions with an Unmanned Aerial Vehicle (UAV). *Journal of Intelligent Robotic Systems* 64, 1 (10 2011), 57–76. http://proxy.lib.ohio-state.edu/login?url=https://www-proquest-com.proxy.lib.ohio-state.edu/docview/889789980?accountid=9783

[2] J. Boubin, J. Chumley, C. Stewart, and S. Khanal. 2019. Autonomic Computing Challenges in Fully Autonomous Precision Agriculture. In *2019 IEEE International Conference on Autonomic Computing (ICAC)*. 11–17.

[3] Jayson Boubin and Christopher Stewart. 2020. SoftwarePilot: Fully Autonomous Aerial Systems Made Easier. In *1st IEEE International Conference on Autonomic Computing and Self-Organizing Systems*.

[4] Jayson G Boubin, Naveen TR Babu, Christopher Stewart, John Chumley, and Shiqi Zhang. 2019. Managing edge resources for fully autonomous aerial systems. In *Proceedings of the 4th ACM/IEEE Symposium on Edge Computing*. 74–87.

[5] DJI. 2021. DJI - Official Website. https://www.dji.com/

[6] H. Genc, Y. Zu, T. Chin, M. Halpern, and V. J. Reddi. 2017. Flying IoT: Toward Low-Power Vision in the Sky. *IEEE Micro* 37, 6 (November 2017), 40–51. https://doi.org/10.1109/MM.2017.4241339

[7] Davis E. King. 2009. Dlib-ml: A Machine Learning Toolkit. *Journal of Machine Learning Research* 10 (2009), 1755–1758.

[8] W. Meng, Z. He, R. Su, P. K. Yadav, R. Teo, and L. Xie. 2017. Decentralized Multi-UAV Flight Autonomy for Moving Convoys Search and Track. *IEEE Transactions on Control Systems Technology* 25, 4 (2017), 1480–1487.

[9] Joseph Redmon and Ali Farhadi. 2018. YOLOv3: An Incremental Improvement. *CoRR* abs/1804.02767 (2018). arXiv:1804.02767 http://arxiv.org/abs/1804.02767

[10] Ming-Der Yang, Jayson G Boubin, Hui Ping Tsai, Hsin-Hung Tseng, Yu-Chun Hsu, and Christopher C Stewart. 2020. Adaptive autonomous UAV scouting for rice lodging assessment using edge computing with deep learning EDANet. *Computers and Electronics in Agriculture* 179 (2020), 105817.

[11] Shuo Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. 2016. WIDER FACE: A Face Detection Benchmark. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[12] Zichen Zhang, Jayson Boubin, Christopher Stewart, and Sami Khanal. 2020. Whole-Field Reinforcement Learning: A Fully Autonomous Aerial Scouting Method for Precision Agriculture. *Sensors* 20, 22 (2020), 6585.