Efficient and coordinated vertical takeoff of UAV swarms

Francisco Fabra, Jamie Wubben, Carlos T. Calafate, Juan Carlos Cano, Pietro Manzoni
Department of Computer Engineering (DISCA)
Universitat Politècnica de València (UPV)
Camino de Vera S/N, 46022, Valencia, Spain
E-mail: frafabco@cam.upv.es, wubben.jamie@outlook.com, {calafate, jucano, pmanzoni}@disca.upv.es

***Abstract*—As we witness the unrelenting growth of the UAV sector, novel and more sophisticated applications keep emerging every year, with many more in the horizon. Among these, applications that require the adoption of UAV swarms are among the most complex, as deploying swarms requires the interaction and cooperation of all the UAVs involved, which can become quite challenging. In this work we specifically focus on the swarm takeoff procedure for UAVs of the Vertical Take-Off and Landing (VTOL) type, proposing a heuristic that achieves reduced computing overhead while introducing near-optimal assignments of UAV positions in the swarm formation selected. Such heuristic is complemented by an efficient and collision-free takeoff approach that relies on adequate ordering and inter-UAV communications to achieve a sequential phased takeoff. A large number of experiments using our own ArduSim emulation platform, which is totally compatible with real drone code, evidence the improvements achieved in terms of time overhead and safety when compared to both ideal and agnostic approaches.**

***Index Terms*—UAV; swarm; safe takeoff; flight coordination.**

I. Introduction

Unmanned Aerial Vehicles (UAVs) are semi-autonomous or fully autonomous unmanned aircrafts that are nowadays being used for an increasing number of tasks and applications, having gradually moved from more established areas like aerial photography and video, to new areas like precision agriculture, border surveillance, package delivery, thermal inspections, and air taxis, among others [1].

In many cases, the use of UAV swarms multiply their potential capabilities. In fact, UAVs flying autonomously in cooperation can create or improve networks (e.g. solving infrastructure problems), transfer information, monitor weather or traffic, etc.

In general, the more drones are there in a swarm, the more capable the swarm becomes. However, deploying more drones means more inputs and increased complexity that could affect the swarm's behavior, and also some critical decisions to prevent drones from crashing into each other. For these reasons, the management of UAV swarms, and specifically those of the Vertical Take-Off and Landing (VTOL) type, to accomplish joint tasks is also being addressed by different research groups worldwide [2]–[4]. In 2016, Intel was a pioneer in this area, using 500 drones to create the first UAV-based Light Show with such a massive number of UAVs[1]. In December 2017, EHANG used 1180 drones to create a similar show in China[2], a number that was increased to 1374 drones in April 2018[3]. Later, in July 2018, Intel broke the Guinness World Record for simultaneously flying the largest number of unmanned aerial vehicles, with 2.018 drones[4]. In both cases, all the experiment was managed centrally, and very strict deployment conditions were enforced to guarantee its success. However, such singular experiment features are not flexible enough to adapt to any number of UAVs, under any conditions, and for any swarm layout, while keeping computational overhead to a minimum.

With regard to the specific issue of achieving a safe takeoff of a large number of UAVs participating in a swarm, we find that very few works address this topic. In [5] authors consider three simple takeoff options for a swarm: manual, sequential and simultaneous. However, when the swarm is large, and when the formation in the air remains unrelated to positions on the ground, these techniques can take too much time (manual, sequential), or be prone to cause collisions between UAVs (simultaneous), especially when UAVs are packed together on the ground.

To address the aforementioned challenge, in this paper we present a heuristic to efficiently compute the assignment of UAV positions in a swarm. In addition, we propose a novel mechanism that leverages wireless communications to achieve a fast, ordered and collision-free takeoff process. Using our own realistic UAV emulation tool [6], we demonstrate the effectiveness of the proposed solution when compared to the ideal solution, and also to a simpler (agnostic) solution based on random assignments.

The remainder of this paper is organized as follows: in section II we provide a more detailed overview of the problem, and describe the ideal solution for the swarm position assignment. Section III presents our solution that includes a heuristic to quickly assign UAVs to their position in the swarm, along with an enhanced takeoff procedure involving two phases. Then, in section IV, we assess the performance

[1]https://www.tokyo-motorshow.com/en/press_release/20191018.html
[2]https://www.popsci.com/china-drone-swarms/
[3]http://www.ehang.com/news/365.html
[4]https://newsroom.intel.com/news/intel-breaks-guinness-world-records-title-drone-light-shows-celebration-50th-anniversary/

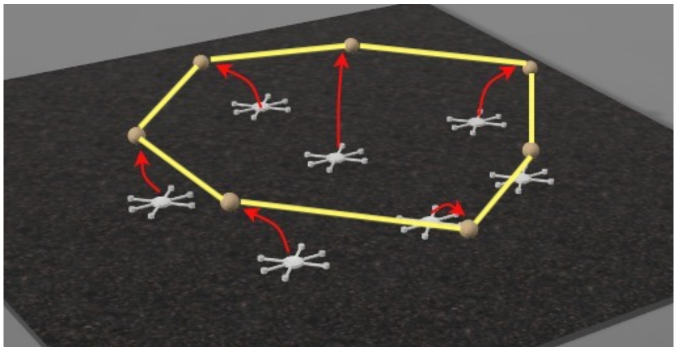

Fig. 1. Example of groud-to-air assignments for UAVs conforming a single swarm.

benefits achieved by our solution. Finally, section V concludes the paper, and discusses future works.

II. Problem overview

Creating a swarm of UAVs is a challenging task that involves defining the swarm size and layout, defining the assignment of ground UAVs to positions on the swarm when flying, making UAVs takeoff in a safe and yet effective manner while introducing small delays, moving the swarm in a coordinated manner and avoiding collisions, and finally landing the UAVs safely and in a small area.

In previous works we have proposed different swarm management solutions [7], [8], as well as precise landing techniques using a vision-based approach [9]. Here we address the takeoff problem, a problem that remains mostly untackled in the literature mostly because actually testing with a large number of UAVs is very challenging, and thus only very few large-scale experiments of this kind have been reported.

To have a better insight into the complexity of the swarm takeoff problem, in figure 1 we illustrate a scenario where several UAVs on the ground have to participate in a swarm following the layout depicted above them. In general, to have the UAV swarm ready for completing a mission, two requirements typically have to be met: (i) UAVs should be assigned positions on the swarm so as to minimize their flight time, e.g. avoiding that UAVs near one edge on the ground assume positions on the opposite edge in the formation; and (ii) the takeoff sequence should be such that it is quick enough, while guaranteeing that UAVs do not collide into each other during the process.

With regard to the first of these requirements, the optimal assignment that guarantees the minimal overall distance travelled by UAVs to their position on the swarm is presented formally in Algorithm 1. Basically this algorithm analyses the full universe of solutions to seek the ones that minimize the target function:

$$\Phi = \sum_{i=1}^{n} dist(P^i_{ground}, P^i_{aerial})^2 \tag{1}$$

Algorithm 1 idealTakeOff(numUAVs, groundLocations, flightFormation)

```
Require: groundLocations.size = numUAVs ∧
         flightFormation.size = numUAVs
bestFit = ∅
totalError = MAX_VALUE
for centerLocation in groundLocations do
    airLocations = f(centerLocation, flightFormation)
    IDsPermutations = f(groundLocations.IDs)
    while (P_IDs = IDsPermutations.next) ≠ ∅ do
        error = 0
        fit = ∅
        i = 0
        for i < numUAVs do
            aLocation = airLocations[i]
            id = P_IDs[i]
            gLocation = groundLocations[id]
            error += gLocation.distance(aLocation)^2
            fit ← (id, gLocation, aLocation)
            i++
        end for
        if fit.error < totalError then
            totalError = fit.error
            bestFit = fit
        end if
    end while
end for
return bestFit
```

where n is the number of UAVs, and $dist(P^i_{ground}, P^i_{aerial})$ returns the euclidean distance between the ground and the aerial position for a specific UAV i.

The main drawback of this ideal algorithm is that searching through all possible solutions makes calculation complexity have a factorial growth. In fact, as shown later in section IV, calculation times become prohibitive even for a low number of UAVs (<20). Thus, an alternative (suboptimal) method to accelerate these calculations is proposed in the next section.

With respect to the takeoff sequence, and timing, both should guarantee that collisions are avoided. In the literature we cannot find relevant proposals regarding the takeoff sequence, and concerning their timing, usually only standard approaches like sequential or simultaneous takeoff are considered [5], although simultaneous takeoff is prone to cause crashes in many cases, and sequential takeoff can take an excessive time when the size of the swarm is significant (>30 UAVs). Hence, these issues are also addressed in our proposed solution detailed in section III.

III. Proposed solution

In this section we present our proposed solution for the efficient and safe takeoff of UAVs belonging to a swarm. Our proposal encompasses three different contributions: (i) a heuristic for making an efficient UAV-to-swarm position assignment that offers a performance similar to algorithm 1, but with a significantly lower time overhead; (ii) a takeoff sequence that can adapt to any swarm layout, and that minimizes the risks associated to other simpler, agnostic schemes, like simultaneous or random takeoff; and (iii) a protocol based on wireless communications that allows UAVs to quickly negotiate the takeoff timings so as to avoid the high delays associated to more conservative schemes, like the sequential takeoff approach.

A. Proposed heuristic

Our proposed heuristic to associate UAVs to their target position in the flight formation is presented as algorithm 2.

Algorithm 2 proposedTakeOff(numUAVs, groundLocations, flightFormation)

```
Require: groundLocations.size = numUAVs ∧
         flightFormation.size = numUAVs
centerLocation = mean(groundLocations)
airLocations = f(centerLocation, flightFormation)
airList = ∅
for loc in airLocations do
    airList ← (loc, loc.distance(centerLocation))
end for
sort airList in descending distance order
fit = ∅
for aLocation in airList do
    bestError = MAX_VALUE
    for gLocation in groundLocations do
        error = gLocation.distance(aLocation)^2
        if error < bestError then
            bestError = error
            bestID = gLocation.ID
        end if
    end for
    fit ← (id, groundLocations[bestID], aLocation)
    groundLocations.remove(bestID)
end for
return fit
```

Basically, our approach consists in determining a location on the ground which is central with regard to the UAVs deployed. Then, such central position is used to compute the distance towards all the positions in the desired flight formation, which are then sorted in descending order. Using this list, the UAV closer to each of these positions is then assigned to it. Notice that, in terms of computational cost, algorithm 2 has a cost that grows with $O(n^2)$, while the ideal solution presented earlier (see algorithm 1) has a cost of $O(n! \cdot n^2)$. Thus, it is feasible to deploy our solution even in embedded platforms with low computing power such as a Raspberry Pi, while the ideal solution has a cost that becomes prohibitive for such embedded devices even for a very low number of nodes.

B. Takeoff sequence

Once the assignment of UAVs to their position in the swarm is obtained, the output of algorithm 2 can be used to determine the takeoff sequence. Notice that vector *fit* returned by the algorithm is ordered according to the distance to the center of the ground positions (descending order). Thus, for the takeoff, our proposal is just to follow a sequence that respects this descending order, meaning that UAVs moving to more remote locations will takeoff first, and UAVs very close to their target location will takeoff last. This strategy brings benefits both in terms of safety and time overhead. Regarding safety, the strategy of occupying first locations that are further away attempts to avoid that UAVs encounter other UAVs while moving to their position. In terms of time overhead, filling-in for longer distances first, and keeping shorter distances to the end, makes the process more brief and effective.

C. Negotiated takeoff protocol

In order to accelerate the takeoff process while keeping collision danger to a minimum, we propose a two-phase strategy that relies on direct UAV-to-UAV communications. Our strategy is the following: the first UAV in vector *fit* takes off vertically, until a specified security height λ is achieved. Upon reaching that height, it communicates with the next UAV on vector *fit*, telling it to takeoff; simultaneously the first UAV will start moving to its target position in the swarm. The second UAV, upon receiving the message, will start a similar takeoff procedure until achieving height λ, and then behaves similarly. The process continues until the last UAV takes off. This UAV does not need to communicate with any other UAV, and so it just moves to its target position once it reaches height λ.

Regarding the optimal choice for parameter λ (see section IV-A), notice that very high values will increase the overall takeoff time for the swarm, while very low values will increase the level of danger, as above this distance the UAV is no longer required to move vertically, being allowed to move in any direction. This could cause it to hit nearby obstacles, or even other UAVs.

IV. Performance results

In this section we will analyse the performance achieved by our solution. First we will detail the simulation setup, then assess the improvements achieved in terms of takeoff time, and lastly check the collision avoidance effectiveness of our approach.

A. Simulation setup

For our experiments we have relied on the ArduSim tool [6]. This tool allows to have a realistic emulation of UAV swarms in real time, with the advantage that the code developed is directly portable to real devices.

In terms of flight formations, we have made tests with 4 different formation layouts: linear, matrix, circle, and mesh. For more detail on these layouts please check [8].

For our study, height λ is set to 5 meters since it offers a reasonable trade-off between time overhead and safety (assuming nearby obstacles are lower that 5 meters). The number of UAVs used ranges from 2 to 50, and the UAVs are placed on the ground in a random but compact manner, guaranteeing a minimum distance between them of 2 meters. When assuming their flight formation positions, they keep a minimum distance between them of 20 meters.

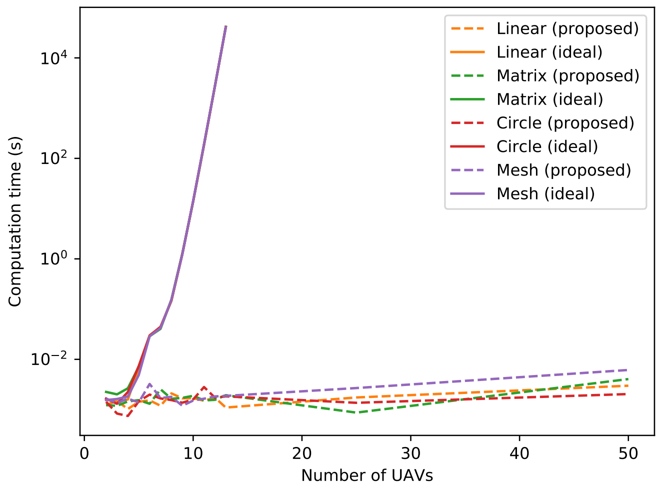

Fig. 2. Computation time (seconds) for the ideal solution and the proposed heuristic when having different swarm layouts.

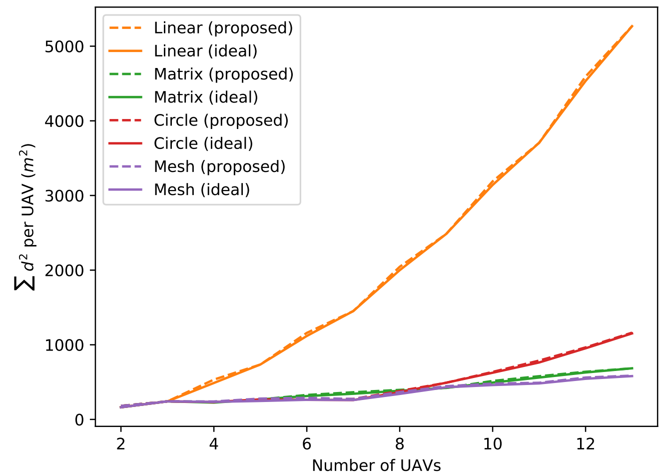

Fig. 3. Optimization function Φ value (m^2) for the ideal solution and the proposed heuristic when having different swarm layouts.

We compare our proposal against the ideal solution, as well as to a random takeoff sequence when checking collision avoidance effectiveness. The main performance metrics adopted are the UAV swarm assignment time, the time and distances associated to the takeoff, and the minimum distance between UAVs during the takeoff procedure (to detect potential danger). The total number of simulations made was of 408 to achieve representativeness for the mean values here presented.

B. *Takeoff time improvements*

To clearly understand the complexity of the problem of associating UAVs to their best swarm position so as to minimize overall flight distances/times, figure 2 shows the computation time involved in finding the association between UAVs (on the ground) and their corresponding position in the swarm. As we can see, for all types of formations, the ideal approach (algorithm 1) is too slow, having a cost that grows as $O(n! \cdot n^2)$, whereas our proposed heuristic (algorithm 2) is extremely fast, being that the computation time remains below 10ms even when having 50 UAVs; also, we find that differences between flight formations are not significant, and that the rate of growth with the number of UAVs is quite acceptable ($O(n^2)$).

In addition, figure 3 shows that the penalty associated to our heuristic, which is a suboptimal solution, in terms of sum of squares of the distances traveled by the UAVs (target function Φ), is not representative compared to the ideal case.

C. *Collision avoidance effectiveness*

We now proceed to highlight the performance of our solution in terms of collision avoidance effectiveness. For this purpose we compare our approach to the ideal case, and also to collision-agnostic case: UAVs are assigned positions in the swarm in a random manner. To have a fair comparison, notice that all protocols tested follow the negotiated takeoff procedure described in section III-C. The chosen formation for this comparison is the linear one.

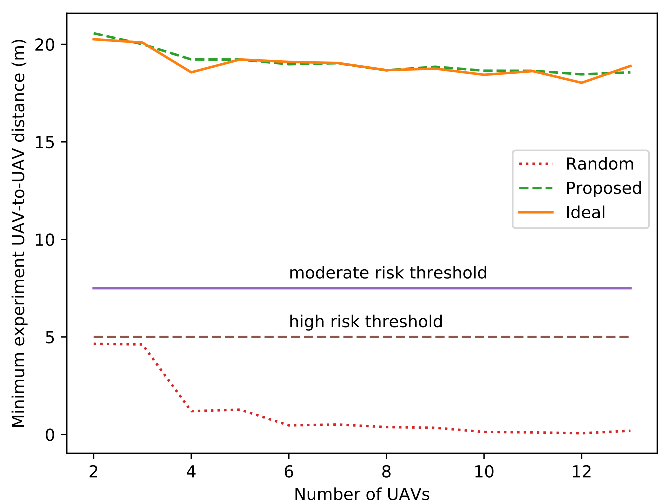

Fig. 4. Minimum UAV-to-UAV distances (linear flight formation) when comparing the proposed solution against the ideal and random approaches.

Figure 4 shows the effectiveness of our solution, which achieves a performance similar to the ideal case. In contrast, the "random" approach is prone to introduce situations of significant danger. In fact, according to Tahar et al. [10], the typical GPS error is of about 5 meters, meaning that situations where less than 7.5 meters between UAVs could represent a real risk (see the moderate risk threshold in figure 4), and for values below 5 meters the risk becomes high. The results for the random approach show that, in the experiments, distances that fall within the high-risk threshold do occur, whereas for our proposal the minimum distances remain close to the minimum distance in the flight formation (about 20 meters), not representing any real danger. A similar result is obtained for the ideal strategy. Hence, we can conclude that a simple, agnostic approach like the random takeoff is not adequate for real scenarios.

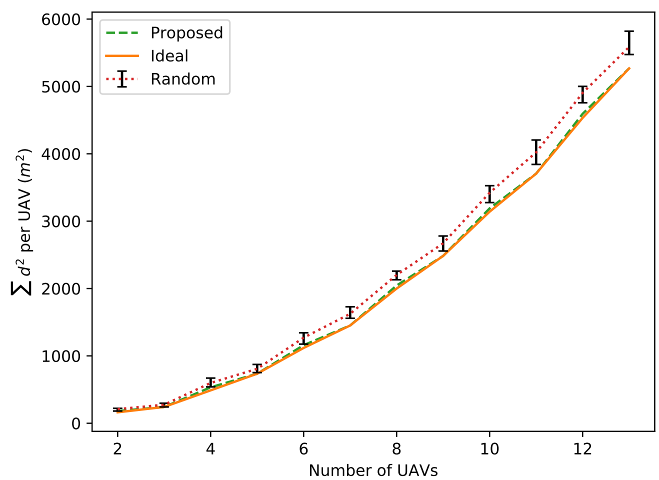

Fig. 5. Optimization function value when comparing our solution against the ideal and random approaches (linear flight formation).

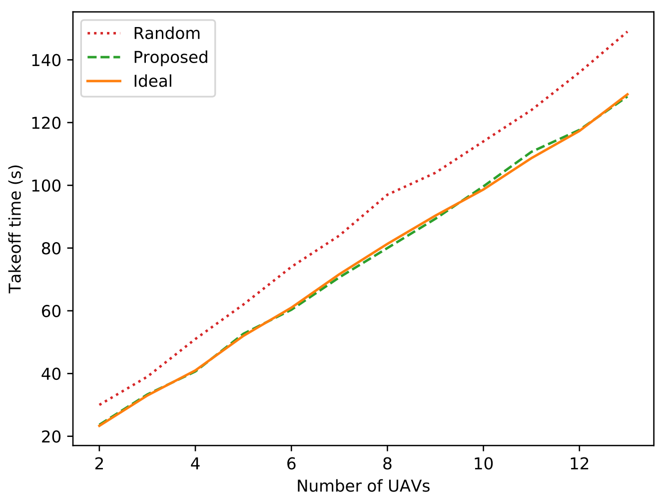

Fig. 6. Total takeoff time when comparing our solution against the ideal and random approaches (linear flight formation).

Additionally, as shown in figures 5 and 6, there is a price to be paid when adopting the random procedure in terms of total distance travelled by the UAVs, which increases by about 5%, and also in terms of overall takeoff time, which can be increased by up to 20 seconds in our experiments (about 20% overhead in the worst case). Figures 5 and 6 only show results related to the linear flight formation, as the takeoff time differences were found to be similar for the other flight formations.

V. Conclusions and Future Work

In this paper we presented a novel approach for the safe takeoff of UAV swarms that provides a good trade-off between position assignment optimization, and computation time, as well as between takeoff time, and safety. A wide set of experiments using our ArduSim emulation tool have shown that we were able to obtain near-optimal position assignments in a very short time, with minimal variations in the total distance travelled by the UAVs compared to the ideal situation. In addition, we have shown that our solution significantly improves safety compared to an agnostic approach (random), while also reducing takeoff time.

As future work we plan to perform a larger-scale analysis by introducing machine learning techniques for comparison, and also to perform tests with real UAVs.

Acknowledgment

This work was partially supported by the ”Ministerio de Ciencia, Innovación y Universidades, Programa Estatal de Investigación, Desarrollo e Innovación Orientada a los Retos de la Sociedad, Proyectos I+D+I 2018”, Spain, under Grant RTI2018-096384-B-I00.

References

[1] H. Shakhatreh, A. H. Sawalmeh, A. Al-Fuqaha, Z. Dou, E. Almaita, I. Khalil, N. S. Othman, A. Khreishah, and M. Guizani, “Unmanned aerial vehicles (uavs): A survey on civil applications and key research challenges,” *IEEE Access*, vol. 7, pp. 48 572–48 634, 2019.

[2] M. Campion, P. Ranganathan, and S. Faruque, “Uav swarm communication and control architectures: a review,” *Journal of Unmanned Vehicle Systems*, vol. 7, no. 2, pp. 93–106, 2019.

[3] N. Goddemeier, K. Daniel, and C. Wietfeld, “Role-based connectivity management with realistic air-to-ground channels for cooperative uavs,” *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 5, pp. 951–963, June 2012.

[4] A. Tahir, J. Böling, M.-H. Haghbayan, H. T. Toivonen, and J. Plosila, “Swarms of unmanned aerial vehicles — a survey,” *Journal of Industrial Information Integration*, p. 100106, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2452414X18300086

[5] N. Dousse, G. Heitz, and D. Floreano, “Extension of a ground control interface for swarms of small drones,” *Artificial Life and Robotics*, vol. 21, no. 3, pp. 308–316, Sep 2016. [Online]. Available: https://doi.org/10.1007/s10015-016-0302-9

[6] F. Fabra, C. T. Calafate, J. C. Cano, and P. Manzoni, “Ardusim: Accurate and real-time multicopter simulation,” *Simulation Modelling Practice and Theory*, vol. 87, pp. 170 – 190, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1569190X18300893

[7] F. Fabra, W. Zamora, P. Reyes, C. T. Calafate, J. Cano, P. Manzoni, and E. Hernandez-Orallo, “An uav swarm coordination protocol supporting planned missions,” in *2019 28th International Conference on Computer Communication and Networks (ICCCN)*, July 2019, pp. 1–9.

[8] F. Fabra, W. Zamora, J. Masanet, C. T. Calafate, J.-C. Cano, and P. Manzoni, “Automatic system supporting multicopter swarms with manual guidance,” *Computers & Electrical Engineering*, vol. 74, pp. 413 – 428, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0045790618323577

[9] J. Wubben, F. Fabra, T. Krzeszowski, J. Marquez-Barja, J. C. Cano, and P. Manzoni, “A vision-based system for autonomous vertical landing of unmanned aerial vehicles,” in *23rd International Symposium on Distributed Simulation and Real Time Applications (IEEE/ACM DS-RT 2019)*, Cosenza, Italy, 2019.

[10] K. N. Tahar and S. Kamarudin, “Uav onboard gps in positioning determination,” *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLI-B1, pp. 1037–1042, 06 2016.