Deep Drone Acrobatics (Extended Abstract)

Elia Kaufmann[1,*] , **Antonio Loquercio**[1,*] , **Rene Ranftl**[2] , **Matthias Müller**[2] ,
Vladlen Koltun[2] , **Davide Scaramuzza**[1]

[1]University of Zurich
[2]Intel
{ekaufmann, loquercio}@ifi.uzh.ch

Abstract

Acrobatic flight with quadrotors is extremely challenging. Maneuvers such as the loop, matty flip or barrel roll require high thrust and extreme angular accelerations that push the platform to its limits. Human drone pilots require years of practice to safely master such maneuvers. Yet, a tiny mistake could make the platform lose control, and brutally crash. This short paper describes an approach to safely train acrobatic controllers in simulation and deploy them with no fine-tuning zero-shot transfer on physical quadrotors. The approach uses only onboard sensing and computation.

1 Introduction

Acrobatic flight with quadrotors is extremely challenging. Human drone pilots require many years of practice to safely master maneuvers such as power loops and barrel rolls[1]. Existing autonomous systems that perform agile maneuvers require external sensing and/or external computation [Lupashin *et al.*, 2010; Abbeel *et al.*, 2010; Bry *et al.*, 2015]. For aerial vehicles that rely only on onboard sensing and computation, the high accelerations that are required for acrobatic maneuvers together with the unforgiving requirements on the control stack raise fundamental questions in both perception and control. Therefore, they provide a natural benchmark to compare the capabilities of autonomous systems against trained human pilots.

Acrobatic maneuvers represent a challenge for the actuators, the sensors, and all physical components of a quadrotor. While hardware limitations can be resolved using expert-level equipment that allows for extreme accelerations, the major limiting factor to enable agile flight is reliable state estimation. Vision-based state estimation systems either provide significantly reduced accuracy or completely fail at high accelerations due to effects such as motion blur, large displacements, and the difficulty of robustly tracking features over long time frames [Cadena *et al.*, 2016]. Additionally, the harsh requirements of fast and precise control at high speeds make it difficult to tune controllers on the real platform, since

---

*These two authors contributed equally.
[1]https://www.youtube.com/watch?v=T1vzjPa5260

even tiny mistakes can result in catastrophic outcomes for the platform.

The difficulty of agile autonomous flight led previous work to mostly focus on specific aspects of the problem. One line of research focused on the control problem, assuming near-perfect state estimation from external sensors [Lupashin *et al.*, 2010; Abbeel *et al.*, 2010; Hwangbo *et al.*, 2017; Bry *et al.*, 2015]. While these works showed impressive examples of agile flight, they focused purely on control. The issues of reliable perception and state estimation during agile maneuvers were cleverly circumvented by instrumenting the environment with sensors (such as Vicon and OptiTrack) that provide near-perfect state estimation to the platform at all times. Recent works addressed the control and perception aspects in an integrated way via techniques like perception-guided trajectory optimization [Falanga *et al.*, 2018; Falanga *et al.*, 2017; Shen *et al.*, 2013] or training end-to-end visuo-motor agents [Zhang *et al.*, 2016]. However, acrobatic performance of high-acceleration maneuvers with only onboard sensing and computation has not yet been achieved.

In this paper, we show for the first time that a vision-based autonomous quadrotor with only onboard sensing and computation is capable of autonomously performing agile maneuvers with accelerations of up to 3g, as shown in Fig. 1. This contribution is enabled by a novel simulation-to-reality transfer strategy, which is based on abstraction of both visual and inertial measurements. We demonstrate both formally and empirically that the presented abstraction strategy decreases the simulation-to-reality gap with respect to a naive use of sensory inputs. Equipped with this strategy, we train an end-to-end sensimotor controller to fly acrobatic maneuvers *exclusively* in simulation. Learning agile maneuvers entirely in simulation has several advantages: (i) Maneuvers can be simply specified by reference trajectories in simulation and do not require expensive demonstrations by a human pilot, (ii) training is safe and does not pose any physical risk to the quadrotor, and (iii) the approach can scale to a large number of diverse maneuvers, including ones that can only be performed by the very best human pilots.

Our sensorimotor policy is represented by a neural network that combines information from different input modalities to directly regress thrust and body rates. To cope with different output frequencies of the onboard sensors, we design an asynchronous network that operates independently of the sensor
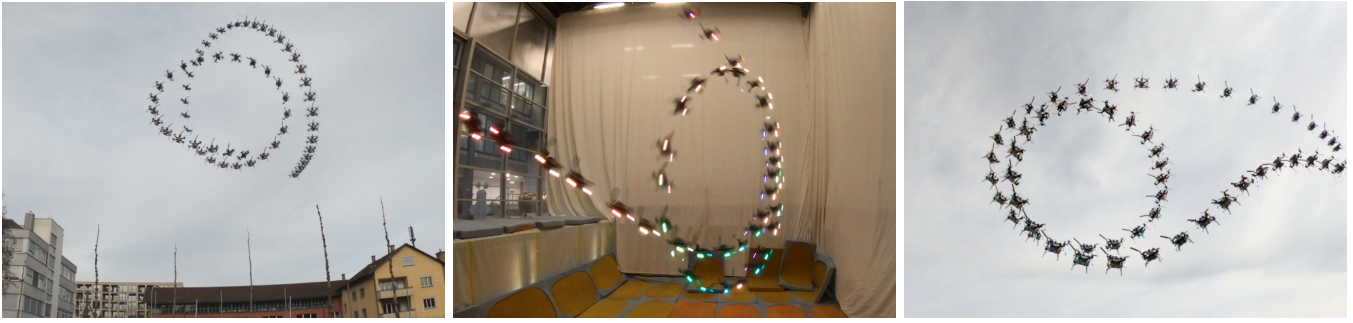
Figure 1: A quadrotor performs a Barrel Roll (left), a Power Loop (middle), and a Matty Flip (right). We safely train acrobatic controllers in simulation and deploy them with no fine-tuning (*zero-shot transfer*) on physical quadrotors. The approach uses only onboard sensing and computation. No external motion tracking was used.
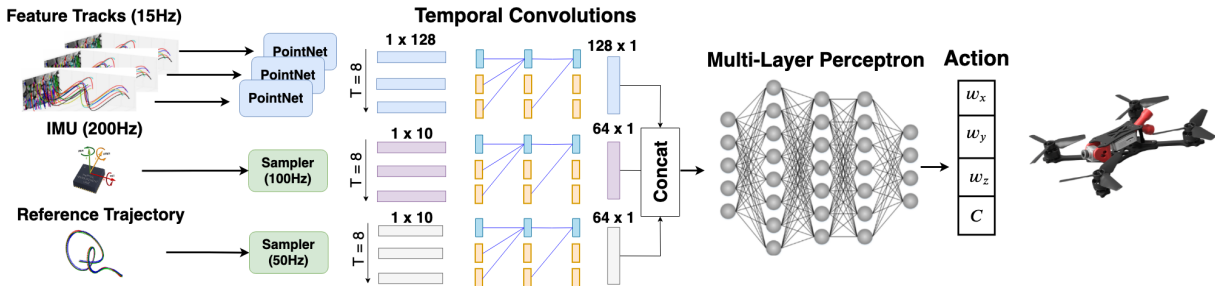

Figure 2: Network architecture. The network receives a history of feature tracks, IMU measurements, and reference trajectories as input. Each input modality is processed using temporal convolutions and updated at different input rates. The resulting intermediate representations are processed by a multi-layer perceptron at a fixed output rate to produce collective thrust and body rate commands.

frequencies. This network is trained in simulation to imitate demonstrations from an optimal controller that has access to privileged state information.

We apply the presented approach to learning autonomous execution of three acrobatic maneuvers that are challenging even for expert human pilots: the Power Loop, the Barrel Roll, and the Matty Flip. Through controlled experiments in simulation and on a real quadrotor, we show that the presented approach leads to robust and accurate policies that are able to reliably perform the maneuvers with only onboard sensing and computation.

2 Approach Overview

We train a sensorimotor controller to predict low-level actions from a history of onboard sensor measurements and a user-defined reference trajectory. An observation $o[k] \in \mathbb{O}$ at time $k \in [0, \dots, T]$ consists of a camera image $\mathcal{I}[k]$ and an inertial measurement $\phi[k]$. Since the camera and IMU typically operate at different frequencies, the visual and inertial observations are updated at different rates. The controller's output is an action $u[k] = [c, \omega^\top]^\top \in \mathbb{U}$ that consists of continuous mass-normalized collective thrust c and bodyrates $\omega = [\omega_x, \omega_y, \omega_z]^\top$ that are defined in the quadrotor body frame.

The controller is trained via *privileged learning* [Chen *et al.*, 2019]. Specifically, the policy is trained on demonstrations that are provided by a privileged expert: an optimal controller that has access to privileged information that is not

available to the sensorimotor student, such as the full ground-truth state of the platform $s[k] \in \mathbb{S}$. The privileged expert is based on a classic optimization-based planning and control pipeline that tracks a reference trajectory from the state $s[k]$ using MPC [Falanga *et al.*, 2018].

3 Deep Sensorimotor Controller

In contrast to the privileged expert, the deep sensorimotor controller is only provided with onboard sensor measurements from the forward-facing camera and the IMU. There are three main challenges for the controller to tackle: (i) it should keep track of its state based on the provided inputs, akin to a visual-inertial odometry system, (ii) it should be invariant to environments and domains, so as to not require retraining for each scene, and (iii) it should process sensor readings that are provided at different frequencies.

We represent the policy as a neural network that fulfills all of the above requirements. Figure 2 illustrates the network architecture. We account for the different input frequencies by allowing each of the input branches to operate asynchronously. The output of the neural network is then processed by a low-level controller, which converts the provided thrust and body-rates in rotor commands. While the low-level controller is platform specif and informed about the physics of the drone, the network actions are mostly platform independent. The network is trained with an off-policy learning approach. More details about the training procedure can be found in the main manuscript [Kaufmann *et al.*, 2020].
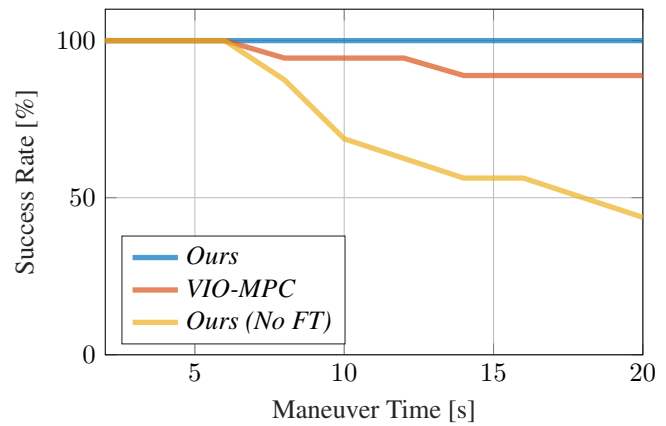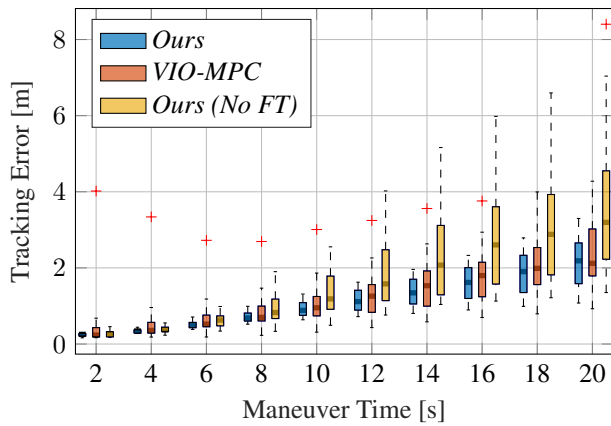
Figure 3: Tracking error (left) and success rate (right) over time when a maneuver is executed repeatedly in simulation. The controllers were trained to complete the maneuver for six seconds and generalize well to longer sequences. Our learned controller, which leverages both IMU and visual data, provides consistently good performance without a single failure.

3.1 Training Methodology

Training the above controller requires significant amount of data. Not only is collection of this data with a real robot tedious and expensive, but also challenging. Specifically, the two main challenges are: (i) How to provide perfect state information to a real drone? and (ii) How to protect the platform from damage when a partially trained network is in control? To circumvent these challenges, we train *exclusively* in simulation. This significantly simplifies the training procedure, but presents a new hurdle: how do we minimize the difference between the sensory input received by the controller in simulation and reality?

Our approach to bridging the gap between simulation and reality is to leverage *abstraction*. Rather than operating on raw sensory input, our sensorimotor controller operates on an intermediate representation produced by a perception module. This intermediate representation is more consistent across simulation and reality than raw visual input. We formally show that training a network on abstraction of sensory input reduces the gap between simulation and reality [Kaufmann *et al.*, 2020].

In our case, we use feature tracks as an abstraction of camera frames. The feature tracks are provided by a visual-inertial odometry (VIO) system. Thanks to this abstraction, we do not require any randomization of the geometry and appearance of the scene during data collection. In contrast to camera frames, feature tracks primarily depend on scene geometry, rather than surface appearance. We also make inertial measurements independent of environmental conditions, such as temperature and pressure, by integration and de-biasing.

4 Results

Influence of Maneuver Length. As shown in Figure 3, our experimental evaluation reveals that for very short maneuvers (up to 6 seconds) IMU measurements were sufficient for acrobatic flight. However, for longer flight duration, visual information was necessary to successfully address the IMU drift and complete the maneuver. Indeed, visual information reduces the odds of a crash by up to 30% in the longest maneuvers. Interestingly, the neural network learns to find a bal-

Input	Train		Test 1		Test 2	
	Error ($\downarrow$)	Success ($\uparrow$)	Error ($\downarrow$)	Success ($\uparrow$)	Error ($\downarrow$)	Success ($\uparrow$)
Image	90 ± 32	80%	∞	0%	∞	0%
Ours	$\mathbf{53 \pm 15}$	**100%**	$\mathbf{58 \pm 18}$	**100%**	$\mathbf{61 \pm 11}$	**100%**

Table 1: Sim-to-sim transfer for different visual input modalities.

ance between feature tracks and inertial measurements. Indeed, when looking at images with low features (for example when the camera points to the sky), the neural net will mainly rely on IMU. When more features are available, the network uses them to correct the accumulated drift of the IMU. On all maneuvers, we outperform the tradition pipeline of state-estimation and control (VIO-MPC) in term of tracking error and odds of a crash.

Abstraction Helps Training & Generalization. To validate the importance of input abstraction, we compare our approach to a network that uses raw camera images instead of feature tracks as visual input. We then compare the results of this naive approach with our proposed abstraction procedure in Table 1. In the training environment, the image-based network has a success rate of only 80%, with a 58% higher tracking error than the controller that receives an abstraction of the visual input in the form of feature tracks (Ours). Even more dramatically, the image-based controller fails completely when tested with previously unseen background images (Test 1, Test 2). In contrast, our approach maintains a 100% success rate in these conditions.

Real World Results. We further evaluate the learned controllers with a series of quantitative experiments on the physical platform. Without any finetuning on real data, our controllers can fly all maneuvers with no intervention. The real-world experiments confirm the findings obtained in simulation, *i.e.* visual information is crucial to compensate the drift of inertial measurements during long maneuvers [Kaufmann *et al.*, 2020]. We refer the reader to the supplementary video to understand the dynamic nature of the experiments.

5 Conclusion

Our approach is the first to enable an autonomous flying machine to perform a wide range of acrobatics maneuvers that are highly challenging even for expert human pilots. The approach relies solely on onboard sensing and computation, and leverages sensorimotor policies that are trained entirely in simulation. The presented methodology is not limited to autonomous flight and can enable progress in other areas of robotics.

References

[Abbeel *et al.*, 2010] Pieter Abbeel, Adam Coates, and Andrew Y Ng. Autonomous helicopter aerobatics through apprenticeship learning. *The International Journal of Robotics Research*, 29(13):1608–1639, 2010.

[Bry *et al.*, 2015] Adam Bry, Charles Richter, Abraham Bachrach, and Nicholas Roy. Aggressive flight of fixed-wing and quadrotor aircraft in dense indoor environments. *The International Journal of Robotics Research*, 34(7):969–1002, 2015.

[Cadena *et al.*, 2016] Cesar Cadena, Luca Carlone, Henry Carrillo, Yasir Latif, Davide Scaramuzza, José Neira, Ian D. Reid, and John J. Leonard. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on Robotics*, 32(6):1309–1332, 2016.

[Chen *et al.*, 2019] Dian Chen, Brady Zhou, Vladlen Koltun, and Philipp Krähenbühl. Learning by cheating. In *Conference on Robot Learning (CoRL)*, 2019.

[Falanga *et al.*, 2017] Davide Falanga, Elias Mueggler, Matthias Faessler, and Davide Scaramuzza. Aggressive quadrotor flight through narrow gaps with onboard sensing and computing using active vision. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2017.

[Falanga *et al.*, 2018] Davide Falanga, Philipp Foehn, Peng Lu, and Davide Scaramuzza. PAMPC: Perception-aware model predictive control for quadrotors. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018.

[Hwangbo *et al.*, 2017] Jemin Hwangbo, Inkyu Sa, Roland Siegwart, and Marco Hutter. Control of a quadrotor with reinforcement learning. *IEEE Robotics and Automation Letters*, 2(4):2096–2103, 2017.

[Kaufmann *et al.*, 2020] Elia Kaufmann, Antonio Loquercio, René Ranftl, Matthias Müller, Vladlen Koltun, and Davide Scaramuzza. Deep drone acrobatics. In *Proceedings of Robotics: Science and Systems*, Corvalis, Oregon, USA, July 2020.

[Lupashin *et al.*, 2010] Sergei Lupashin, Angela Schöllig, Michael Sherback, and Raffaello D'Andrea. A simple learning strategy for high-speed quadrocopter multi-flips. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1642–1648, 2010.

[Shen *et al.*, 2013] Shaojie Shen, Yash Mulgaonkar, Nathan Michael, and Vijay Kumar. Vision-based state estimation and trajectory control towards high-speed flight with a quadrotor. In *Robotics: Science and Systems (RSS)*, 2013.

[Zhang *et al.*, 2016] Tianhao Zhang, Gregory Kahn, Sergey Levine, and Pieter Abbeel. Learning deep control policies for autonomous aerial vehicles with MPC-guided policy search. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 528–535, 2016.