How Fast Is Too Fast? The Role of Perception Latency in High-Speed Sense and Avoid

Davide Falanga, Suseong Kim, and Davide Scaramuzza

***Abstract*—In this letter, we study the effects that perception latency has on the maximum speed a robot can reach to safely navigate through an unknown cluttered environment. We provide a general analysis that can serve as a baseline for future quantitative reasoning for design tradeoffs in autonomous robot navigation. We consider the case where the robot is modeled as a linear second-order system with bounded input and navigates through static obstacles. Also, we focus on a scenario where the robot wants to reach a target destination in as little time as possible, and therefore cannot change its longitudinal velocity to avoid obstacles. We show how the maximum latency that the robot can tolerate to guarantee safety is related to the desired speed, the range of its sensing pipeline, and the actuation limitations of the platform (i.e., the maximum acceleration it can produce). As a particular case study, we compare monocular and stereo frame-based cameras against novel, low-latency sensors, such as event cameras, in the case of quadrotor flight. To validate our analysis, we conduct experiments on a quadrotor platform equipped with an event camera to detect and avoid obstacles thrown towards the robot. To the best of our knowledge, this is the first theoretical work in which perception and actuation limitations are jointly considered to study the performance of a robotic platform in high-speed navigation.**

***Index Terms*—Collision avoidance, visual-based navigation, aerial systems: perception and autonomy.**

I. Introduction

HIGH-SPEED robot navigation in cluttered, unknown environments is currently an active research area [1]–[7] and benefits of over 50 million US dollar funding available through the DARPA Fast Lightweight Autonomy Program (2015-2018) and the DARPA Subterranean Challenge (2018-2021).

To prevent a collision with an obstacle or an incoming object, a robot needs to detect them as fast as possible and execute a safe maneuver to avoid them. The higher the relative speed between the robot and the object, the more critical the role of *perception latency* becomes.

Perception latency is the time necessary to *perceive* the environment and *process* the captured data to generate control commands. Depending on the task, the processing algorithm, the available computing power, and the sensor (e.g., lidar, camera, event camera, RGB-D camera), the perception latency can vary from tens up to hundreds of milli-seconds [2]–[7].

At the current state of the art, the agility of autonomous robots is bounded, among the other factors (such as their actuation limitations), by their sensing pipeline. This is because the relatively high latency and low sampling frequency limit the aggressiveness of the control strategies that can be implemented. It is typical in current robots to have latencies of tens or hundreds of milli-seconds. Faster sensing pipelines can lead to more agile robots.

Despite the importance of the perception latency, very little attention has been devoted to study its impact on the agility of a robot for a sense and avoid task. Analyzing the role of sensing latency allows one to understand the limitations of current perception systems, as well as to comprehend the benefits of exploiting novel image sensors and processors, such parallel visual processors (e.g., SCAMP [8]), with a theoretical latency of few milli-seconds, or event cameras, with a theoretical latency of micro-seconds (e.g., the DVS [9]) or even nano-seconds (e.g., CeleX [10]).

In the context of robot navigation, it is also important to correlate the sensing latency to the actuation capabilities of the robot. Broadly speaking, the larger the acceleration a robot can produce, the lower the time it needs to avoid an obstacle and, therefore, the larger the latency it can tolerate. Consequently, the coupling between sensing latency and the actuation limitations of a robot represents a key research problem to be addressed.

A. Related Work

Sensing latency is a known issue in robotics and has already been investigated before. For example, this problem is particularly interesting when the state estimation process is done through visual localization. A number of vision-based solutions for low-latency localization based either on standard cameras [11], [12] or novel sensors (e.g., event cameras [2], [13], [14]) have been proposed. Impressive results have been achieved, however no information about the environment is available since visual localization only provides the robot the information about its pose.

It is not yet clear what the maximum latency of a perception system for a navigation task should be. A first step in that direction is available in [15], where the authors studied under which circumstances a high frame-rate is best for real-time

Manuscript received September 10, 2018; accepted January 30, 2019. Date of publication February 7, 2019; date of current version February 28, 2019. This letter was recommended for publication by Associate Editor H. Kurniawati and Editor N. Amato upon evaluation of the reviewers' comments. This work was supported by the SNSF-ERC Starting Grant and the Swiss National Science Foundation through the National Center of Competence in Research (NCCR) Robotics. *(Corresponding author: Davide Falanga.)*

The authors are with the Robotics and Perception Group, Department of Informatics, University of Zurich and Department of Neuroinformatics,University of Zurich and ETH Zurich, 8050 Zurich, Switzerland (e-mail: falanga@ifi.uzh.ch; suseong@ifi.uzh.ch; davide.scaramuzza@ieee.org).

This letter has supplementary downloadable multimedia material available at http://ieeexplore.ieee.org provided by the authors. This includes a file How_Fast_Video.mp4 shows the experiments conducted with a quadrotor, for which we presented results in the letter and a file How_Fast_Supp.pdf reports additional derivations for the case of vision-based perception, as well as a description of the experimental setup. This material is 9.23 MB in size.

Digital Object Identifier 10.1109/LRA.2019.2898117

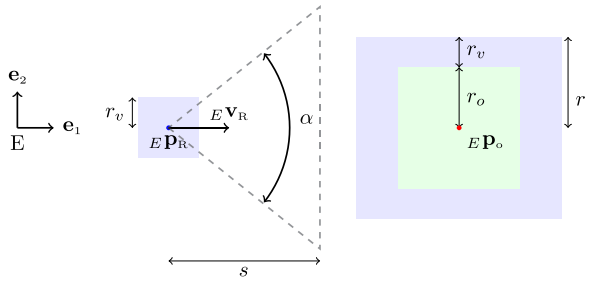

Fig. 1. A schematics representing the obstacle and the robot model in the frame E. The robot is represented as a square of size $2r_v$ centered at ${}_E\mathbf{p}_R$, and moves with a speed ${}_E\mathbf{v}_R$. The dashed triangle starting from the robot's position represents its sensing area, α is the field of view and s the maximum distance it is able to perceive. The obstacle, represented by the green square on the right side of the image, has size $2r_o$. We expand the square representing the obstacle by a quantity r_v such that the robot can be considered to be a point mass.

tracking, providing quantitative results that help selecting the optimal frame-rate depending on required performance. The results of that work were tailored towards visual localization for state estimation. In [16] the performance of visual servoing as a function of a number of parameters describing the perception system (e.g., frame-rate, latency) was studied, and a relation between the tracking error in the image plane and the latency of the perception was derived.

In [17], a framework to predict and compensate for the latency between sensing and actuation in a robotic platform aimed at visually tracking a fast-moving object was proposed and experimental results showed the benefits of that framework. Nevertheless, the impact of the latency on the performance of the executed task without the proposed compensation framework was not discussed.

The most similar work to ours is [18], where the authors studied the performance of vision-based navigation for mobile robots depending on the latency and the sensing range of the perception system. A trade-off among camera frame rate, resolution, and latency was shown to represent the best configuration for navigation in unstructured terrain. However, such results were only supported by experimental results, without any theoretical evidence. Different from our work, the actuation capabilities of the robot were not considered.

To the best of our knowledge, no previous works analyzed the coupling between sensing latency and actuation limitations in a robotic platform from a theoretical perspective. Similarly, the problem of highlighting their impact on the performance of high-speed navigation has not been addressed in the literature.

B. Contributions

In this letter, we focus on the effects of perception latency and actuation limitations on the maximum speed a robot can reach to safely navigate through an unknown, static scenario.

We consider the case where a generic robot, modeled as a linear system with bounded inputs, moves in a plane and relies on onboard perception to detect static obstacles along its path (cf. Fig. 1). We focus on a scenario where the robot wants to reach a target destination in as little time as possible, and therefore cannot change its longitudinal velocity to avoid obstacles. We show how the maximum latency the robot can tolerate to guarantee safety is related to the desired speed, the agility of the platform (e.g., the maximum acceleration it can produce), as well as other perception parameters (e.g., the sensing range). Additionally, we derive a closed-form expression for the maximum speed that the robot can reach as a function of its perception and actuation parameters, and study its sensitivity to such parameters.

We provide a general analysis that can serve as a baseline for future quantitative reasoning for design trade-offs in autonomous robot navigation, and is completely agnostic to the sensor and robot type. As a particular case study, we compare standard cameras against event cameras for autonomous quadrotor flight, in order to highlight the potential benefits of these novel sensors for perception. Finally, we provide an experimental evaluation and validation of the proposed theoretical analysis for the case of a quadrotor, equipped with an event camera, avoiding a ball thrown towards it at speeds up to 9 ms^{-1}.

To the best of our knowledge, this is the first work in which perception and actuation limitations are jointly considered to study the performance of a robot in high-speed navigation.

C. Assumptions

This letter is based on the following assumptions. First, we assume that the robot can be model as a linear system. Robotic systems are typically characterized by non-linear models. However, a large variety of them can be linearized through either static or dynamic feedback [19], rendering them equivalent from a control perspective to a chain of integrators. It is important to note that feedback linearization is different from Jacobian linearization: the first is an exact representation of the original non-linear system over a large variety of working conditions, while the second is only valid locally [20]. Linear models for mobile robots have already been used in the past [1], and come with the advantage of allowing a simple, yet effective mathematical analysis of the behaviour of the system in closed-form. Also, they cover a large variety of systems, rendering our analysis valid for different kinds of robots.

Second, we assume that the robot can execute holonomic 2D maneuvers. For non-holonomic systems, such as fixed-wing aircraft, the coupling of the longitudinal and lateral dynamics would break the assumptions of our model and would deserve a different analysis.

Finally, since we are interested in the role of sensing latency and actuation limitations on the agility of a robot, we assume that, for any other aspect, the sensing and actuation system are ideal. In other words, we assume that there is no uncertainty in the obstacle detection, no illumination issues, no artifacts in the measurements, and the robot's dynamics is perfectly known and can be controlled with errors. This allows us to clearly isolate and analyze the impact of sensing latency and actuation limitations in our analysis, where otherwise it would not be possible to distinguish the role of these two from the impact of other sources of non-ideality.

D. Structure of the Paper

In Sec. II, we provide the mathematical formulation of the problem and perform a qualitative analysis. In Sec. III, we particularize our study to vision-based navigation and analyze it for both standard and event cameras. A detailed mathematical analysis of these sensors is provided in the supplementary material. In Sec. IV, we compare standard cameras (monocular and stereo) against event cameras for the case study of autonomous

quadrotor flight. In Sec. V, we validate our analysis performing experiments on an actual quadrotor avoiding obstacle thrown towards it. Further details about the experiments are provided in the supplementary material. Finally, in Sec. VI, we draw the conclusions.

II. Problem Formulation

We consider the case of a mobile robot navigating in a plane, which covers a large number of scenarios, e.g. an aerial robot flying in a forest [1], where the third dimension would not help with the avoidance task. The robot moves along a desired direction with a desired speed, provided by a high-level planner, towards its goal, which has to be reached in as little time as possible. Therefore, the robot cannot change its longitudinal velocity. In the following analysis, we consider the case where the robot only faces one single obstacle along its path and then provide an intuitive explanation of how our conclusions can be extended to the case of multiple obstacles.

A. Modelling

1) Robot Model: Let E be the inertial reference frame, having basis $\{\mathbf{e}_1, \mathbf{e}_2\}$, and let ${}_E\mathbf{p}_\mathrm{R}$ and ${}_E\mathbf{v}_\mathrm{R}$ be the position and velocity, respectively, of the robot in E. Also, let ${}_E\mathbf{p}_\mathrm{O}$ be the position of an obstacle in E. In the remainder, we will refer to $\mathbf{e}_1$ as the *longitudinal* axis, and $\mathbf{e}_2$ as the *lateral* axis. Finally, let r_v be the half-size of the square centered at ${}_E\mathbf{p}_\mathrm{R}$ containing the entire robot (cf. Fig. 1).

We model both the longitudinal and lateral dynamics as a chain of integrators. As shown in [19], a large variety of mechanical systems can be linearized by using nonlinear feedback, which, from a control perspective, renders them equivalent to a chain of integrators. Additionally, the dynamics of the actuators is usually faster than the mechanical dynamics and can, therefore, be neglected.

The longitudinal and lateral dynamics are modeled by a position $\mathrm{p_i}$, a speed $\mathrm{v_i}$ and an input $\mathrm{u_i}$ given by:

$$\dot{\mathrm{p}}_1(t) = \mathrm{v}_1(t), \quad \dot{\mathrm{v}}_1(t) = \mathrm{u}_1(t), \tag{1}$$

$$\dot{\mathrm{p}}_2(t) = \mathrm{v}_2(t), \quad \dot{\mathrm{v}}_2(t) = \mathrm{u}_2(t). \tag{2}$$

Both inputs are assumed to be bounded such that $\mathrm{u_i} \in [-\bar{\mathrm{u}}_\mathrm{i}, \bar{\mathrm{u}}_\mathrm{i}], i = 1, 2$. We assume the robot to move only along the longitudinal axis with an initial speed $\mathrm{v}_{1,0} = \hat{\mathrm{v}}_1$, meaning that the lateral speed is zero before the avoidance maneuver starts. The case where the robot has non-zero lateral velocity can be analyzed using the same mathematical framework. Also, we assume that the robot cannot change its longitudinal speed, namely $\mathrm{u}_1(t) = 0\ \forall t$, and can therefore only exploit the lateral dynamics to avoid an obstacle. As shown in Sec. S1 of the supplementary material, a lateral avoidance maneuver requires less time at high speed, allowing faster navigation along the longitudinal axis.

2) Obstacle Model: We consider static obstacles enveloped by a square of width $2r_o$. To study the motion of the robot considering only the position of its center, we expand the obstacle width by a quantity r_v on each side. The expanded size of the obstacle is $r = 2(r_v + r_o)$, as shown in Fig. 1.

3) Sensor Model: In this letter, we assume that at least one edge of the obstacle must enter the sensing area to allow a detection. We define the sensing latency $\tau \in \mathbb{R}^+$ as the interval between the time the obstacle enters the sensing area and the moment the robot's initiates the avoidance maneuver. The latency of a sensor is typically the sum of multiple contributions, and in general depends on the sensor itself and the time necessary to process a measurement (which depends on the algorithm used, the computational power available, and other factors). In general, it is hard to provide exact bounds for each of these contributions, therefore we consider as latency the sum of the sensor's and the sensing algorithm's latency. We denote by $s \in \mathbb{R}^+$ the robot's sensing range, i.e. the largest distance it is able to perceive. We assume the field of view of the sensor to be such that the obstacle's edge is fully contained in the sensing area when the distance between the robot and the obstacle is equal to the sensing range. This provides a lowerbound for the field of view $\alpha \geq 2\arctan\left(\frac{r_o}{2s}\right)$.

B. Obstacle Avoidance

1) Time to Contact and Avoidance Time: We define the *time to contact* t_c as the time it takes the vehicle to collide with the obstacle once it enters the sensing range of its onboard sensor. Since the longitudinal motion has a constant speed $\hat{\mathrm{v}}_1$ and the distance between the vehicle and the obstacle at the time the obstacle enters the sensing area is s, the time to contact t_c is:

$$t_\mathrm{c} = \frac{s}{\hat{\mathrm{v}}_1}. \tag{3}$$

In order for the robot to avoid the obstacle, it has to reach a safe lateral position in an *avoidance time* t_s shorter than the time to contact (3).

$$t_\mathrm{c} \geq t_\mathrm{s}. \tag{4}$$

2) Time-Optimal Avoidance: The avoidance maneuver along the lateral axis leads to a safe navigation if $\mathrm{p}_2(t_\mathrm{c}) \geq r$. We consider the case $\mathrm{p}_2(t_\mathrm{c}) = r$, which represents the minimum lateral deviation for the avoidance maneuver to be executed safely. For this to happen, we assume the robot to use a time-optimal strategy $\mathrm{u}_2^*(t)$:

$$\begin{aligned} \mathrm{u}_2^*(t) = \underset{\mathrm{u}_2(t)}{\arg\min} \quad & t_\mathrm{s} \\ \text{subject to} \quad & \dot{\mathrm{p}}_2(t) = \mathrm{v}_2(t),\ \dot{\mathrm{v}}_2(t) = \mathrm{u}_2(t), \\ & \mathrm{p}_2(0) = 0, \quad \mathrm{v}_2(0) = 0, \\ & \mathrm{p}_2(t_\mathrm{s}) = r, \quad \mathrm{v}_2(t_\mathrm{s}) = 0, \\ & \mathrm{u}_2(t) \in [-\bar{\mathrm{u}}_2, \bar{\mathrm{u}}_2]\ \forall t. \end{aligned} \tag{5}$$

We require $\mathrm{v}_2(t_\mathrm{s}) = 0$ because there would be no advantage in having a non-zero lateral speed in terms of progressing towards the goal, since we considered the longitudinal axis to be the direction of motion. Leaving the final lateral speed free would lead to a lower execution time for the avoidance maneuver, but this could potentially result in a large lateral speed, which is typically not desirable because the robot is not able to sense the environment in such a direction. As well known in the literature [21], the problem (5) leads to a *bang-bang* solution:

$$\mathrm{u}_2^*(t) = \begin{cases} \bar{\mathrm{u}}_2 & \text{if } 0 \leq t \leq \hat{t} \\ -\bar{\mathrm{u}}_2 & \text{if } \hat{t} < t \leq t_\mathrm{s} \end{cases}, \tag{6}$$

where the $\hat{t} = \sqrt{\frac{r}{\bar{\mathrm{u}}_2}}$ is the switching time and $t_\mathrm{s} = 2\sqrt{\frac{r}{\bar{\mathrm{u}}_2}}$ is the avoidance time.

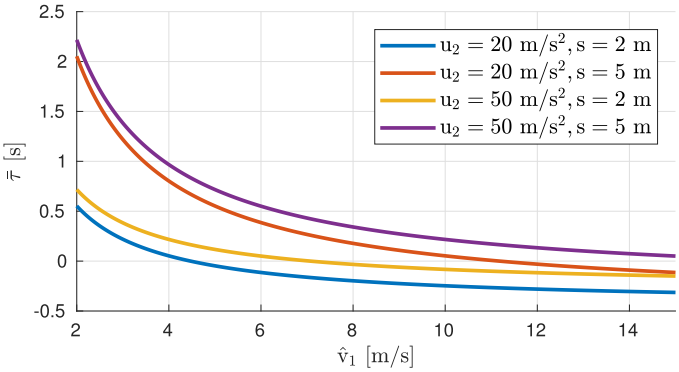

Fig. 2. Maximum latency $\bar{\tau}$ that the robot can tolerate in order to safely perform the avoidance maneuver when $r = 0.5$ m.

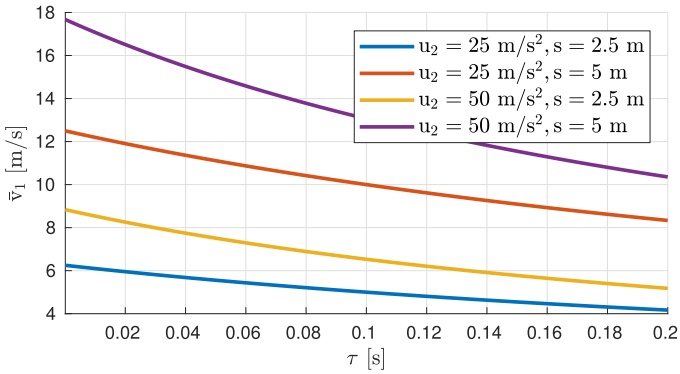

Fig. 3. Maximum speed $\bar{v}_1$ that the robot can move in order to safely perform the avoidance maneuver when $r = 0.5$ m.

3) Obstacle Avoidance with Sensing Latency: In Sec. II-B1 we defined the time to contact t_c as the time between when the obstacle enters the sensing range and the moment when the collision occurs, as defined in (3). However, in the presence of sensing latency, the time $t_c^{'}$ remaining to the collision when the robot is informed about the presence of the obstacle is $t_c^{'}(\tau) = t_c - \tau$. Therefore, in order for a robot equipped with a sensor with sensing range s and latency τ to safely avoid an obstacle, the condition $t_c^{'}(\tau) \geq t_s$ must hold. In this case, we can compute (4) as:

$$\frac{s}{\hat{v}_1} - \tau \geq 2\sqrt{\frac{r}{\bar{u}_2}}. \tag{7}$$

The worst case in which the robot manages to avoid the obstacle occurs when (7) is satisfied with equality. In this case, the robot passes tangent to the obstacle, whereas it would have some safety margin if (7) was satisfied with the inequality sign. We can study (7) to compute the maximum latency $\bar{\tau}$ the system can tolerate such that the avoidance can still be performed safely:

$$\bar{\tau} = \frac{s}{\hat{v}_1} - 2\sqrt{\frac{r}{\bar{u}_2}}. \tag{8}$$

Fig. 2 shows the maximum latency $\bar{\tau}$ for different values of $\bar{u}_2$ and s for the case $r = 0.5$ m. As one can notice, the importance of low latency increases as the navigation speed increases. Also, for some speeds $\hat{v}_1$ the robot is unable to perform the avoidance maneuver safely given its actuation capabilities and the sensing range of its sensor. This is clear from the negative values the maximum latency $\bar{\tau}$ assumes in some intervals. In this case the robot should be either more agile (i.e. capable of generating higher lateral accelerations) or should be equipped with a sensor with a higher sensing range in order to avoid the obstacle at such speeds.

Similarly, we can use (8) to compute the maximum longitudinal speed the robot can have to avoid the obstacle:

$$\bar{v}_1 = \frac{s}{\tau + 2\sqrt{\frac{r}{\bar{u}_2}}}. \tag{9}$$

Fig. 3 shows the maximum speed the robot can navigate safely (i.e., being still able to avoid the obstacle although this is perceived with some delay), depending on the latency of its sensing pipeline.

III. Vision-Based Perception

In the following, we particularize our analysis to the case of vision-based perception for three modalities: (i) a monocular frame-based camera, (ii) a stereo frame-based camera, (iii) a monocular event camera, and analyze the impact of their latency on the maximum speed. For brevity reasons, the mathematical derivation of the expressions for the sensing range and the latency of each of these sensing modalities is reported in the supplementary material attached to this letter.

A. Frame-Based Cameras and Event Cameras

Most computer vision research has been devoted to frame-based cameras, which have latencies in the order of tens of milliseconds, thus, putting a hard bound on the achievable agility of a robotic platform. By contrast, event cameras [9] are bio-inspired vision sensors that output pixel-level brightness changes at the time they occur, with a theoretical latency of micro-seconds or even nano-seconds. More specifically, rather than streaming frames at constant time intervals, each pixel *fires* an event (a pixel-level brightness change), independently of the other pixels, every time it detects a change of brightness in the scene. Broadly speaking, we can consider event cameras as *motion-activated*, asynchronous edge detectors: events fire only if there is relative motion between the camera and the scene.

Exploiting frame-based cameras for obstacle avoidance typically requires the analysis of all the pixels of the image to detect an obstacle, independently of the texture. Conversely, since the pixels of an event camera only trigger information when there is change of intensity, it has the advantage of requiring very little processing to detect an obstacle. Furthermore, since the smallest time interval between two consecutive events on the same pixel is in the order of 1 μs, or generally much smaller than the typical framerate of frame-based cameras, this can safely be neglected.

These factors result in a theoretical advantage of event cameras against frame-based cameras.

B. Sensing Range of a Vision-Based Perception System

1) Monocular Frame-Based Camera: The sensing range s_M of a monocular camera depends, as shown in Sec. S4-A of the supplementary material, on the size r_o of the obstacle, the number of pixels N it must occupy in the image to be detected, and the camera's angular resolution θ.

2) Stereo Frame-Based Camera: The sensing range s_S of a stereo camera depends, as shown in Sec. S5-A of the supplementary material, on the baseline b, the focal length f, the uncertainty in the disparity ϵ_P and the maximum percentual uncertainty k in the depth estimation.

3) Event Camera: In Sec. S6-A of the supplementary material we show that the sensing range s_E of an event camera can be computed using (S.1). It depends on how large the object must be in the image such that, when its edges generate an event, they are sufficiently far apart.

C. Latency of a Vision-Based Perception System

1) Monocular Frame-Based Camera: The latency τ_M of a monocular camera depends on the time t_f between two consecutive triggers of the sensor, the exposure time t_E, the transfer time t_T, the processing time and the number of images necessary to detect the obstacle. As shown in Sec S4-B of the supplementary material, if two consecutive images are sufficient to detect an obstacle, it can vary between $\tau_M = t_f + t_T + t_E$ and $\tau_M = 2t_f$.

2) Stereo Frame-Based Camera: In Sec. S5-B of the supplementary material, we analyze the possible range of the latency τ_S of a stereo camera. In general, it can span between a best-case value equal to the time between two consecutive frames, and a worst-case value, which we derive analyzing the datasheet of several stereo cameras.

3) Event Camera: The latency τ_E of an event camera depends, as shown in Sec. S6-B of the supplementary material, on the distance between the camera and the obstacle, the speed of the camera, the focal length, and the amount of pixels the projection of the obstacle must move in the image such that it fires an event. However, to derive the maximum speed achievable with an event camera, it is necessary to jointly consider the expression of the latency of an event camera and (4). We refer the reader to Sec. S6-B of the supplementary material for further details.

IV. Case Study: Vision-Based Quadrotor Flight

In this section, we analyze the case of vision-based quadrotor flight. We consider a quadrotor equipped with a sensing pipeline based on frame-based cameras in a monocular and stereo configuration, and a monocular event camera. For each sensing modality, we provide an upper and a lower-bound of the sensing range and the latency according to the model in Sec. III. We compute the maximum speed achievable with each sensor for a value of each parameter equal to its lower-bound, its upper-bound, and the average between the upper and the lower-bound. Finally, we consider four different values for the maximum lateral acceleration the quadrotor can produce. Three values correspond to commercially available state-of-the-art quadrotors with low, medium and high *thrust-to-weight* ratio. The fourth one, instead, represents a quadrotor with a *thrust-to-weight* ratio that is, as of today, particularly hard to achieve with current technology, but might become common in the future. This ideal platform serves us to show that more agile quadrotors would significantly highlight the benefits of lower-latency sensors for obstacle avoidance.

A. Sensing Range

1) Monocular Frame-Based Camera: We use the results of Sec. S4-A of the supplementary material to obtain the upper-bound and the lower-bound for the sensing range of a monocular camera. The best-case scenario occurs when the obstacle to be detected occupies 5% of the image, leading to an upper-bound $s_M = 6$ m. We consider as worst-case scenario when the obstacle occupies 10%, leading to a lower-bound $s_M = 2$ m .

2) Stereo Frame-Based Camera: We assume the robot to be equipped with a stereo system having a baseline $b = 0.10$ m and each camera having a VGA resolution. As shown in Sec. S5-A of the supplementary material, we consider $s_S = 2$ m and $s_S = 8$ m to be reasonable values for the lower-bound and the upper-bound of the sensing range.

3) Event Camera: As mentioned in Sec. S6-A of the supplementary material, the sensing range of an event camera can reach values above $s_E = 10$ m. Intuitively speaking, this is because to potentially detect an obstacle with an event camera, it is sufficient that the projection of its edges move on the image by 1 pxl and are far apart from each others by an amount that is at least on order of magnitude larger (i.e., at least 10 pxl apart). However, to render our comparison more fair and realistic, we consider a lower-bound that is comparable to the one of frame cameras. Indeed, when a robot navigates cluttered environments, its distance from the obstacles is typically lower than 10 m, which makes it necessary to consider a lower value for the smallest sensing range of event camera. Therefore, we assume $s_E = 2$ m as lower-bound for the sensing range of an event camera, and $s_E = 8$ m as its upper-bound.

B. Latency

1) Monocular Frame-Based Camera: We consider a frame-based camera with (i) a framerate of 50 Hz, meaning that $t_f = 0.020$ s; (ii) an exposure time of $t_E = 0.005$ s; (iii) VGA resolution and USB 3.0 connection, which leads to $t_T = 0.0004$ s. Therefore, based on Sec. S4-B of the supplementary material, the upper-bound and the lower-bound latency for the frame-based camera considered in this analysis are, respectively, $\tau_M = 0.040$ s and $\tau_M = 0.026$ s.

2) Stereo Frame-Based Camera: As mentioned in Sec. S5-B of the supplementary material, it is hard to evaluate the latency of a stereo system. However, based on the datasheet of commercially available stereo cameras suitable for quadrotor flight, we can obtain an estimate of the upper-bound and the lower-bound. As upper-bound, we consider the Bumblebee XB3, whose datasheet reports a latency of $\tau_S = 0.070$ s. For the lower-bound, since no further information are available in the datasheet of other stereo cameras, we assume it to be equal to the inverse of the frame-rate of the fastest available sensor (Intel RealSense R200) leading to $\tau_S = 0.017$ s.

3) Event Camera: In Sec. S6-B of the supplementary material we discuss how the latency of an event camera depends on the relative distance and speed between the robot and the obstacle. Also, we highlight that, in order to compute it, it is necessary to jointly consider the sensing range (Sec. S6-A), Eq. (8) and Eq. (S.5). Therefore, to analyze the maximum speed achievable with an event camera we proceed as follows: (i) we consider a value of the sensing range as described in Sec. III-B3; (ii) we plug (9) into (S.5) and solve it for $\hat{v}_1$ to compute the maximum speed achievable; (iii) we use (8) to obtain the corresponding value of the latency of an event camera, given its distance from the obstacle and its speed.

C. Quadrotor Model

The dynamical model of a quadrotor is differentially flat and the vehicle can be considered as a linear system using nonlinear feedback linearization [22] both from a control [23] and a planning perspective [24]. We considered four cases for the maximum lateral acceleration the robot can produce: $\bar{u}_2 = 10\ \text{ms}^{-2}$, $\bar{u}_2 = 25\ \text{ms}^{-2}$, $\bar{u}_2 = 50\ \text{ms}^{-2}$, and $\bar{u}_2 = 200\ \text{ms}^{-2}$. These values correspond to a *thrust-to-weight ratio* of approximately 1.5, 2.8 5.2 and 20, respectively. The first three cover a large range of the lift capabilities of commercially available drones, while the fourth represents a vehicle currently not yet available, but which might be available in the future. We assume $r_v = 0.25$ m and $r_o = 0.50$ m, leading to an expanded obstacle size of $r = 0.75$ m.

D. Results

The results of our analysis for vision-based quadrotor flight are available in Table I. For each sensing modality (first column) we combined three values for the sensing range (second column) and the latency (third column), and computed the maximum speed the robot can achieve depending on the maximum lateral acceleration it can produce (fourth column). For frame-based camera (monocular and stereo), we considered as values for the sensing range and the latency the lower-bound, the upper-bound and the average between upper-bound and lower-bound.

Similarly, we considered three values for the sensing range of an event camera. However, as mentioned in Sec. IV-B3, the latency of event cameras is strictly connected to the robot's agility. As shown in Sec. S6-B of the supplementary material, the theoretical latency of an event camera depends on both its distance to the obstacle and its velocity towards it (c.f. Eq. (S.5)). Broadly speaking, the faster the robot, the earlier the desired amount of events for the detection are generated. However, for the obstacle avoidance problem to be well-posed, the robot cannot be arbitrarily fast, but its speed must be such that the avoidance maneuver requires an amount of time smaller than the time to contact (Eq. (4) and (7)). This means that the theoretical latency of an event camera depends also on the maximum lateral input the robot can produce. Therefore, for a given sensing range and robots maximum input, one can compute the corresponding maximum velocity achievable and, consequently, the latency of an event camera mounted on such a robot. Since different robots maximum input would produce different maximum velocity, the same event camera will similarly have different latencies (Eq. (S.5)). This motivates the dashed values in Table I.

As one can notice, when the sensing range and the robot's agility are small, the difference among mononocular frame cameras, stereo frame cameras and event cameras is not remarkable. Conversely, frame cameras in stereo configuration and event cameras allow faster flight than a monocular frame camera when either the sensing range or the robot's agility increase. In particular, increasing the sensing range, as expected from Sec. S2, allows the robot to navigate faster thanks to a sensible increase of the time to contact.

Similarly, making the robot more agile (i.e., increasing $\bar{u}_2$) allows it to fly faster thanks to the decrease of the avoidance time. As one can notice by the results in the column of the quadrotor having and $\bar{u}_2 = 200\ \text{ms}^{-2}$, the difference between the maximum speed achievable with stereo frame-based cameras and event cameras become significant. Depending on the sensing range, low-latency event cameras allow the robot to reach a maximum speed that can be between 7% and 12% larger than the one achievable with a stereo frame-based camera. It is important to remark that, despite the numbers provided for the case $\bar{u}_2 = 200\ \text{ms}^{-2}$ are very high, they are not as far as one could think from what is currently achievable by agile quadrotors. Indeed, First-Person-View (FPV) quadrotors are currently capable of reaching speeds above $40\ \text{ms}^{-1}$ with thrust-to-weight ratios above 10 and, given the pace of the technological progress in the FPV community, it is not hard to believe that, in the near future, quadrotors will be able to reach speeds significantly beyond the current values. In FPV racing, a small increase in the maximum flight speed can represent the step necessary to outperform other vehicles participating in the race. This is particularly interesting in the contest of autonomous FVP drone racing, an extremely active area of research [25], [26].

V. Experiments

To validate our analysis, we performed real-world experiments with a quadrotor platform equipped with an Insightness SEEM1 sensor,[1] a very compact neuromorphic camera providing standard frame, events and Inertial Measurement Unit data. The obstacle was a ball of radius 10 cm thrown towards the quadrotor, and the vehicle only relied on the onboard event camera to detect it and avoid it. From the perspective of our model, this is equivalent to the case where the robot moves towards the obstacle, since the time to contact depends on the absolute value of the relative longitudinal velocity. This experimental setup allowed us to reach large relative velocities in a confined space. Further details about the experimental platform used in this work are available in Sec.S8-A of the supplementary material.

A. Obstacle Detection with an Event Camera

To detect the obstacle, whose size is supposed to be known, we use a variation of the algorithm proposed in [27] to remove events generated by the static part of the environment due to the motion of the camera. Different from [27], we do not compensate for the camera's motion using numerical optimization, but rather exploiting the gyroscope's measurements. This allows our pipeline to be faster, but comes at the cost of a higher amount of not compensated events.

We accumulate motion-compensated events over a sliding window of 10 ms, obtaining an *event-frame* containing the timestamp of the events due to the motion of moving objects. Such event-frame typically consists of several separated blobs, which are clustered together using the DBSCAN algorithm [28] based on their relative distance, their direction of motion (obtained using Lucas-Kanade tracking [29]) and the timestamp of the events. We fit a rectangle around the blobs belonging to the same cluster and look for the rectangle having the most similar aspect ratio to the expected one. Since we assume the size of the obstacle to be known, we compute its expected aspect ratio and, after finding the most similar cluster, we project its the centroid into the world frame using the standard pinhole camera projection model.

To render our algorithm most robust to outliers, we considered the obstacle to be detected only when at least n measurements in the world frame are obtained and their relative distance is

[1] http://www.insightness.com/technology

TABLE I
THE RESULTS OF OUR CASE STUDY. WE COMPARE MONOCULAR FRAME-BASED CAMERAS, STEREO FRAME-BASED CAMERAS AND EVENT CAMERAS FOR DIFFERENT ROBOT AGILITY VALUES. THE DASHES IN THE COLUMNS REPORTING THE MAXIMUM SPEED ACHIEVABLE WITH AN EVENT CAMERA ARE DUE TO THE FACT THAT, GIVEN A VALUE FOR THE SENSING RANGE AND THE MAXIMUM LATERAL ACCELERATION, WE CAN COMPUTE THE MAXIMUM ACHIEVABLE SPEED AND THE CORRESPONDING LATENCY (C.F. SEC. IV-D FOR A MORE DETAILED EXPLANATION)

Sensor Type	**Sensing Range** [m]	**Latency** [s]	**Max. speed** [m/s]			
			$\bar{u}_2 = 10\,\mathrm{m/s^2}$	$\bar{u}_2 = 25\,\mathrm{m/s^2}$	$\bar{u}_2 = 50\,\mathrm{m/s^2}$	$\bar{u}_2 = 200\,\mathrm{m/s^2}$
Mono Frame	2.0	0.026	3.48	5.37	7.38	13.47
	2.0	0.033	3.44	5.27	7.20	12.83
	2.0	0.040	3.40	5.17	7.02	12.30
	4.0	0.026	5.23	8.06	11.07	26.94
	4.0	0.033	5.17	7.91	10.79	25.73
	4.0	0.040	5.10	7.76	10.53	24.62
	6.0	0.026	6.97	10.74	14.76	40.41
	6.0	0.033	6.89	10.54	14.39	38.59
	6.0	0.040	6.81	10.35	14.03	36.93
Stereo Frame	2.0	0.017	3.54	5.51	7.64	14.37
	2.0	0.043	3.38	5.13	6.93	12.06
	2.0	0.070	3.24	4.80	6.35	10.39
	5.0	0.017	8.86	13.77	19.11	35.93
	5.0	0.043	8.50	12.83	17.34	30.16
	5.0	0.070	8.10	12.01	15.88	25.98
	8.0	0.017	14.17	22.03	30.57	57.50
	8.0	0.043	13.54	20.53	27.75	48.25
	8.0	0.070	12.95	19.21	25.40	41.56
Mono Event	2.0	0.002	-	-	-	16.12
	2.0	0.003	-	-	8.06	-
	2.0	0.004	-	5.70	-	-
	2.0	0.007	3.60	-	-	-
	5.0	0.004	-	-	-	39.53
	5.0	0.008	-	-	19.76	-
	5.0	0.011	-	13.98	-	-
	5.0	0.017	8.84	-	-	-
	8.0	0.006	-	-	-	62.06
	8.0	0.012	-	-	31.03	-
	8.0	0.018	-	21.94	-	-
	8.0	0.029	13.88	-	-	-

below a threshold. Our experimental evaluation showed that 2 consecutive measurements at a relative distance lower than 20 cm were sufficient to detect the ball in a reliable way. Also, we fixed the sensing range by discarding detections happening when the ball was at a distance from the robot larger than its sensing range.

It is important to note that our detection algorithm was designed with the aim of reducing the latency of the sensing pipeline and, during the tuning stage, speed was prioritized against accuracy. Accurate obstacle detection with event cameras of obstacles of unknown size and shape is beyond of the scope of this letter.

B. Expected and Measured Latency

Theoretically, a 1 pxl motion of the projection of point in the image is sufficient to generate an event. However, in our experiment we realized that a larger motion is necessary to obtain reliable obstacle detection with an event camera. More specifically, the algorithm was able to detect the obstacle thrown towards the vehicle whenever a displacement between of at least 5 pxl was verified. In Sec. S8-C of the supplementary material we analyze this aspect and discuss the main reasons causing the discrepancy between the theoretical ideal model and real data. Also, we exploited the model proposed in Sec. S6-B of the supplementary material to compute the theoretical latency for an event camera having the same resolution of the sensor used in our experiments, for a pixel displacement of 5 pxl. Sec. S8-B of the supplementary material reports the theoretical latency for an obstacle detection pipeline based on an Insightness SEEM1, and the measured latency for our algorithm. As one can see from Fig. 9, Fig. 10, and Tab. I in the supplementary material, the experimental data agree with the theoretical model. Sec. S8-C of the supplementary material discusses the discrepancy between our model and actual data.

C. Results

We performed experiments where the quadrotor described in Sec. S8-A of the supplementary material, equipped with an Insightness SEEM1 sensor and running the detection algorithm described in Sec. V-A, was commanded to avoid a ball thrown towards it. The ball was thrown with a speed spanning between $\hat{\mathrm{v}}_1 = 5\ \mathrm{ms}^{-1}$ and $\hat{\mathrm{v}}_1 = 9\ \mathrm{ms}^{-1}$. The sensing range was 2 m, meaning that any detection at distance larger than this amount was neglected. Therefore, the time to contact spanned between $t_\mathrm{c} = 0.22$ s and $t_\mathrm{c} = 0.40$ s. The robot was commanded to execute an avoidance maneuver either upwards, laterally or diagonally. The obstacle radius was $r_o = 10$ cm, while the robot's size was computed as either its height ($r_v = 15$ cm) or half its

tip-to-tip diagonal ($r_v = 25$ cm), depending on the direction of the the avoidance maneuver. Therefore, the expanded obstacle radius spanned between $r = 25$ cm and $r = 35$ cm. The avoidance spanned between $t_s = 0.17$ s and $t_s = 0.25$ s. In all the experiments , the ball would have hit the vehicle if the avoidance maneuver was not executed, as confirmed by ground truth data provided by the motion-capture system.

VI. Conclusions

In this letter, we studied the effects that perception latency has on the maximum speed a robot can reach to safely navigate through an unknown environment. We provided a general analysis for a robot modeled as a linear second-order system with bounded inputs. We showed how the maximum latency the robot can tolerate to guarantee safety is related to the desired speed, the agility of the platform (e.g., the maximum acceleration it can produce), as well as other perception parameters (e.g., the sensing range). We compared frame-based cameras (monocular and stereo) against event cameras for quadrotor flight. Our analysis showed that the advantage of using an event camera is higher when the robot is particularly agile. We validated our study with experimental results on a quadrotor avoiding a ball thrown towards it a speeds up to 9 ms^{-1} using an event camera. Future work will investigate the use of event cameras for obstacle avoidance on a completely vision-based quadrotor platform, using on-board Visual-Inertial Odometry for state estimation.

Acknowledgment

The authors thank Henri Rebecq, Julien Kohler, and Kevin Kleber for their help with the experiments.

References

[1] S. Karaman and E. Frazzoli, "High-speed flight in an ergodic forest," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2012, pp. 2899–2906.

[2] A. Censi and D. Scaramuzza, "Low-latency event-based visual odometry," in *Proc. IEEE Int. Conf. Robot. Autom.*, Jun. 2014, pp. 703–710.

[3] C. Richter, W. Vega-Brown, and N. Roy, "Bayesian learning for safe high-speed navigation in unknown environments," in *Proc. Int. Symp. Robot. Res.*, 2015, pp. 325–341.

[4] K. Mohta *et al.*, "Fast, autonomous flight in gps-denied and cluttered environments," *J. Field Robot.*, vol. 35, no. 1, pp. 101–120, Apr. 2017.

[5] C. Richter and N. Roy, "Safe visual navigation via deep learning and novelty detection," in *Robot.: Sci. Syst.*, Jul. 2017.

[6] A. J. Barry, P. R. Florence, and R. Tedrake, "High-speed autonomous obstacle avoidance with pushbroom stereo," *J. Field Robot.*, vol. 35, no. 1, pp. 52–68, Jan. 2018.

[7] S. Jung, S. Cho, D. Lee, H. Lee, and D. H. Shim, "A direct visual servoing-based framework for the 2016 IROS Autonomous Drone Racing Challenge," *J. Field Robot.*, vol. 35, no. 1, pp. 146–166, May 2017.

[8] C. Greatwood *et al.*, "Tracking control of a UAV with a parallel visual processor," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, Sep. 2017, pp. 4248–4254.

[9] P. Lichtsteiner, C. Posch, and T. Delbruck, "A 128 × 128 120 dB 30 mW asynchronous vision sensor that responds to relative intensity change," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, Feb. 2006, pp. 2060–2069.

[10] M. Guo, J. Huang, and S. Chen, "Live demonstration: A 768x215; 640 pixels 200meps dynamic vision sensor," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2017.

[11] C. Forster, M. Pizzoli, and D. Scaramuzza, "SVO: Fast semi-direct monocular visual odometry," in *Proc. IEEE Int. Conf. Robot. Autom.*, Jun. 2014, pp. 15–22.

[12] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint Kalman filter for vision-aided inertial navigation," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2007, pp. 3565–3572.

[13] A. Rosinol Vidal, H. Rebecq, T. Horstschaefer, and D. Scaramuzza, "Ultimate SLAM? Combining events, images, and IMU for robust visual SLAM in HDR and high speed scenarios," *IEEE Robot. Autom. Lett.*, vol. 3, no. 2, pp. 994–1001, Apr. 2018.

[14] A. Zhu, N. Atanasov, and K. Daniilidis, "Event-based visual inertial odometry," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, Jul. 2017, pp. 5816–5824.

[15] A. Handa, R. Newcombe, A. Angeli, and A. Davison, "Real-time camera tracking: When is high frame-rate best?" in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 222–235.

[16] M. Vincze, "Dynamics and system performance of visual servoing," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2000, vol. 1, pp. 644–649.

[17] S. Behnke, A. Egorova, A. Gloye, R. Rojas, and M. Simon, "Predicting away robot control latency," in *RoboCup 2003: Robot Soccer World Cup VII*. Lecture Notes Comput. Sci., vol. 3020, Berlin: Springer, 2004, pp. 712–719.

[18] P. Sermanet *et al.*, "Speed-range dilemmas for vision-based navigation in unstructured terrain," in *Proc. 6th IFAC Symp. Intell. Auton. Vehicles*, 2007, vol. 6, pp. 300–305.

[19] M. W. Spong, "Partial feedback linearization of underactuated mechanical systems," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, Sep. 1994, vol. 1, pp. 314–321.

[20] M. A. Henson and D. E. Seborg, Eds., *Nonlinear Process Control*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1997.

[21] D. Bertsekas, *Dynamic Programming and Optimal Control, vol. I*, 2nd ed. Nashua, NH, USA: Athena Scientific, 2005.

[22] D. Mellinger and V. Kumar, "Minimum snap trajectory generation and control for quadrotors," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2011, pp. 2520–2525.

[23] R. Lozano, J. Guerrero, and N. Chopra, "Quadrotor flight formation control via positive realness," *Proc. Int. Fed. Autom. Control*, vol. 45, no. 28, pp. 25–30, 2012.

[24] M. W. Mueller, M. Hehn, and R. D'Andrea, "A computationally efficient motion primitive for quadrocopter trajectory generation," *IEEE Trans. Robot.*, vol. 31, no. 6, pp. 1294–1310, Dec. 2015.

[25] E. Kaufmann, A. Loquercio, R. Ranftl, A. Dosovitskiy, V. Koltun, and D. Scaramuzza, "Deep drone racing: Learning agile flight in dynamic environments," *arXiv e-prints*, 2018. [Online]. Available: http://arxiv.org/abs/1806.08548

[26] T. Sayre-McCord *et al.*, "Visual-inertial navigation algorithm development using photorealistic camera simulation in the loop," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2018, pp. 2566–2573.

[27] A. Mitrokhin, C. Fermuller, C. Parameshwara, and Y. Aloimonos, "Event-based moving object detection and tracking," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, Oct. 2018, pp. 1–9.

[28] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. Second Int. Conf. Knowl. Discovery Data Mining*, Ser. KDD'96, 1996, pp. 226–231.

[29] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. Int. Joint Conf. Artif. Intell.*, 1981, pp. 674–679.