IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

Aerial Transportation of Cable-Suspended Loads with an Event Camera

Fotis Panetsos[1], George C. Karras[2] and Kostas J. Kyriakopoulos[3]

***Abstract*—In this work, we investigate the integration of a Dynamic Vision Sensor (DVS) into an Unmanned Aerial Vehicle (UAV) with a cable-suspended load in order to achieve a robust and fast estimation of the cable's state during the transportation of the load. Based on the advantageous properties of event cameras, our ultimate goal is to design a computationally lightweight event processing method that persistently identifies the cable and estimates its complete state – required for any controller with feedback of the cable's state – within a much shorter time period compared to frame-based algorithms. Using a point cloud representation for the incoming event streams, the proposed method achieves the fast detection of the cable while the respective measurements are afterward fitted to a Bézier curve in order to approximate both the cable angle and angular velocity. Our method is initially validated in an indoor environment, where ground truth is available from a motion capture system, and is subsequently deployed in an outdoor one in order to evaluate its robustness against noise. Throughout the outdoor experiment, the feedback provided by the DVS is incorporated into a Nonlinear Model Predictive Control (NMPC) scheme which drives an octorotor towards reference setpoint positions while minimizing the cable angular motion.**

***Index Terms*—Aerial Systems: Applications, Aerial Systems: Perception and Autonomy, Event Camera, Multirotor, Slung-load Transportation**

I. INTRODUCTION

A. Motivation

TRANSPORTATION of loads with UAVs has risen in popularity during the last decade since it appears to be an efficient tool for various applications and tasks, e.g., cargo delivery, transportation of equipment, water sampling from polluted aquatic environments [1], [2], [3] or search and rescue missions. Especially the suspension of the loads through cables has the advantage of minimum mechanical intervention in the UAV platform and alleviates the need for complicated designs as opposed to the rigid attachment of the loads to the UAVs.

Towards this direction, various control schemes have been demonstrated throughout the literature that aim to control the UAV with the suspended load and address the issues that emerge from the underactuation of the whole system [4]. More precisely, Reinforcement Learning with discrete and continuous action spaces [5], [6], [7], geometric control [8], [9], Mixed Integer Quadratic Program trajectory generation [10], passivity based control [11], vision-based control [12], nonlinear hierarchical control [13], reformulation of the trajectory planning problem as a Mathematical Program with Complementarity Constraints (MPCC) [14], NMPC [2], and backstepping control [15] are among the numerous schemes that aimed to achieve either the elimination of the load's swinging motion or its aggressive maneuvering.

The majority of the aforementioned schemes require robust and high-speed estimation of the complete cable's state and, thus, were validated in indoor environments with accurate state feedback. Consequently, despite the significant number of control schemes throughout the literature, the estimation of the cable's state remains a challenging and open issue, especially when the UAVs with the cable-suspended load are deployed in outdoor environments and real-world conditions, where motion capture systems or markers cannot be used.

B. Related Works

In [2], the authors utilized a supervised learning approach in order to detect the cable with a camera, based on manually collected and labeled training data from the environments where the UAV operated, and estimated the complete state of the load with an Extended Kalman Filter (EKF). A monocular fish-eye camera and a novel encoder-based device were utilized in [16] and the respective measurements were fused with the aid of a Gaussian fusion-based estimation algorithm in order to obtain an estimate of the payload state. Moreover, in [17], a load transportation device (LTD), composed of a two-axis cardan joint with two magnetic encoders attached to each axis, was designed and integrated into a helicopter. However, the oscillations of the LTD and the rope severely affected the measurements and, hence, the authors were forced to develop an observer. In [18], the authors presented a disturbance observer, which based on the model of the system, an IMU, and a load cell, estimated the payload swing angles.

In order to avoid the usage of high-latency frame-based detection algorithms, the integration of multiple sensors which results in complex mechanical interventions in the UAV platform and lower sampling rates, and the development of appropriate fusion techniques and observers, we aim to exploit an event camera for estimating the cable's state. Event cameras offer multiple advantages compared to standard ones, including low latency, low power consumption, high dynamic range, and robustness against motion blur [19]. Additionally, almost

Manuscript received: July 25, 2023; Revised: September 12, 2023; Accepted: October 30, 2023.

This paper was recommended for publication by Editor P. Pounds upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by the European Union's Horizon 2020 Research and Innovation Program PATHOCERT - Pathogen Contamination Emergency Response Technologies under grant agreement No. 883484.

[1]Fotis Panetsos is with the School of Mechanical Engineering, Control Systems Lab, National Technical University of Athens, 15780 Athens, Greece fpanetsos@mail.ntua.gr

[2]George C. Karras is with the Dept. of Informatics and Telecommunications, University of Thessaly, 35100 Lamia, Greece gkarras@uth.gr

[3]Kostas J. Kyriakopoulos is with the Center of AI & Robotics (CAIR), New York University, Abu Dhabi kkyria@nyu.edu

Digital Object Identifier (DOI): see top of this page.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

every UAV comes with a mounting for a camera and, thus, the integration of an event camera is a plug-and-play solution. Various works in the literature have successfully demonstrated the deployment of event cameras in outdoor environments for a wide range of applications, such as powerline inspection [20], pole tracking [21], visual guidance for ornithopter robot flight [22], intruder monitoring [23], autonomous quadrotor flight despite loss of a single rotor [24], or event-based odometry [25]. However, due to the asynchronous nature of event cameras and the novel ways of acquiring visual data, each work is unique and highly dependent on the object to be identified and the task to be addressed. Hence, a novel solution adapted to the specific problem of identifying the cable and estimating its state during aerial transportation is required.

C. Contributions

In this work, an event camera, also known as Dynamic Vision Sensor (DVS), is exploited so as to reliably and efficiently estimate the complete state of a load which is suspended to a UAV through a cable. To the authors' knowledge, the concept of utilizing event cameras for aerial transportation is original. The events, acquired from the DVS, are represented in the form of a point cloud, where both spatiotemporal information and polarity are stored. Afterward, a computationally inexpensive method is developed in order to process the input point cloud and to robustly distinguish the events that correspond to the cable in a quasi-continuous manner, as illustrated in Fig. 1. Eventually, the extracted information is approximated by a Bézier curve, solving a constrained optimization problem, and, thus, both the cable's angle and angular velocity are obtained and the complete state of the cable is estimated. Initially, our approach is evaluated in an indoor environment, where the output of the proposed processing algorithm is compared against accurate measurements provided by a motion capture system. Finally, our method is implemented in a real-world experiment, where its capability to run onboard a resource-constrained UAV and to provide appropriate feedback of the cable's state in a noisy outdoor environment, during a transportation scenario with minimum swinging motion of the cable, is demonstrated. An NMPC, presented in [2], is used during the experiment so as to achieve the control of the vehicle and the minimization of the cable's swinging motion.

II. Equations of Motion

In this section, the kinematic and dynamic equations that govern the motion of the system are briefly presented. We also refer to our previous work [2] for further details.

Consider the vehicle with the load suspended through a cable, as illustrated in Fig. 2 and 3, and let $\mathbf{W}$ be a world frame and $\mathbf{B}$ a body-fixed one, the origin of which coincides with the center of mass (CoM) of the vehicle. Additionally, assume that the upper edge of the cable is attached exactly to the CoM of the vehicle; actually, the cable is attached as close as possible to the CoM of the UAV given the restrictions that emerge from the structure of the platform. Hence, by neglecting the flexibility of the cable, the positions of the lower cable's edge and the vehicle, i.e., ${}^W\mathbf{x}_L$ and ${}^W\mathbf{x}$ respectively, are related to the cable unit vector ${}^W\mathbf{n}$ and the cable length l according to the relationship:

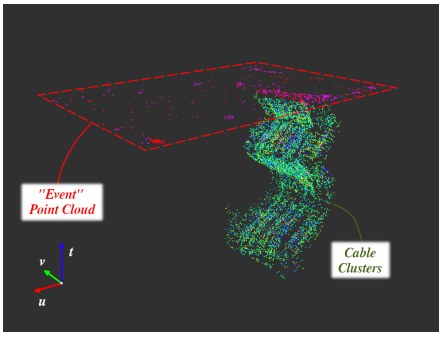


Fig. 1: The current "event" point cloud, obtained by the DVS, and a sequence of cable clusters, extracted according to the proposed method, throughout the last 2 s in an outdoor environment. The u-v axes denote the image plane and the t axis represents the time. The purple and red colored points of the "event" point cloud correspond to events with polarities $p = 1$ and $p = 0$ respectively.

$$ {}^W\mathbf{n} = \frac{1}{l}\left({}^W\mathbf{x}_L - {}^W\mathbf{x}\right) \tag{1} $$

It should be mentioned that the assumption of cable's tautness can be violated in case of considerable wind disturbances or aggressive motion of the UAV.

Moreover, the cable unit vector ${}^W\mathbf{n}$ is described with respect to the angles $\boldsymbol{\eta}_L = \begin{bmatrix}\phi_L & \theta_L\end{bmatrix}^T$ which define rotations around the x and y axes of the world frame $\mathbf{W}$ correspondingly. Hence, the following equations are extracted for the cable unit vector and the cable velocity:

$$ \begin{aligned} {}^W\mathbf{n} &= \mathbf{R}_y(\theta_L)\mathbf{R}_x(\phi_L)\begin{bmatrix}0 & 0 & -1\end{bmatrix}^T \\ \Rightarrow {}^W\mathbf{n} &= \begin{bmatrix}-s\theta_L c\phi_L & s\phi_L & -c\theta_L c\phi_L\end{bmatrix}^T \end{aligned} \tag{2} $$

$$ {}^W\dot{\mathbf{n}} = \begin{bmatrix} s\theta_L s\phi_L & -c\theta_L c\phi_L \\ c\phi_L & 0 \\ c\theta_L s\phi_L & s\theta_L c\phi_L \end{bmatrix}\dot{\boldsymbol{\eta}}_L = J\dot{\boldsymbol{\eta}}_L \tag{3} $$

By exploiting Eq. 1 and 3 and the Newton-Euler equations for the motion of the load, the following equation is derived, assuming that the products $\dot{\phi}_L^2$, $\dot{\phi}_L \cdot \dot{\theta}_L$, $\dot{\theta}_L^2$ are negligible:

$$ \ddot{\boldsymbol{\eta}}_L = \frac{1}{l}\left(J^T J\right)^{-1} J^T\left(-g\mathbf{e_z} - {}^W\dot{\mathbf{v}}\right) \tag{4} $$

where g is the gravity, $\mathbf{e_z}$ is the z-axis of the world frame, and ${}^W\mathbf{v}$ is the velocity of the UAV.

Finally, the full dynamic model of the UAV with the suspended load is defined by:

$$ \begin{aligned} {}^W\dot{\mathbf{x}} &= {}^W\mathbf{v} \\ {}^W\dot{\mathbf{v}} &= -g\mathbf{e_z} + F\mathbf{R}_{WB}\mathbf{e_z} + \frac{\|\mathbf{T}\|}{m}{}^W\mathbf{n} \\ \dot{\phi} &= \left(K_\phi\phi_d - \phi\right)/\tau_\phi \\ \dot{\theta} &= \left(K_\theta\theta_d - \theta\right)/\tau_\theta \\ \dot{\psi} &= \left(K_\psi\psi_d - \psi\right)/\tau_\psi \\ \dot{\boldsymbol{\eta}}_L &= \boldsymbol{\omega}_L \\ \dot{\boldsymbol{\omega}}_L &= \frac{1}{l}\left(J^T J\right)^{-1} J^T\left(-g\mathbf{e_z} - {}^W\dot{\mathbf{v}}\right) \end{aligned} \tag{5} $$

where $\boldsymbol{\omega}_L$ is the cable velocity, $\mathbf{T}$ is the tension of the cable,

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

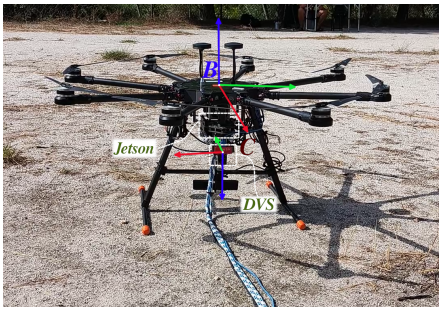


Fig. 2: The octorotor UAV with the onboard computer and the DVS camera. The body-fixed $\mathbf{B}$ and the camera $\mathbf{C}$ frames are also depicted. The red, green, blue colored arrows correspond to the x,y,z axes of each frame.

m is the mass of the vehicle, $\mathbf{R}_{WB}$ is the rotation matrix from $\mathbf{B}$ to $\mathbf{W}$, ϕ, θ, ψ are the roll-pitch-yaw angles of the UAV, ϕ_d, θ_d, ψ_d are the desired roll-pitch-yaw angles directly sent to the inner attitude controller of the autopilot, while τ_ϕ, τ_θ, τ_ψ and K_ϕ, K_θ, K_ψ are the time constants and gains of the inner attitude controller which is modeled by a first-order system owing to the ability of the autopilot, namely the Ardupilot firmware [26], to rapidly control the attitude subsystem [27]. A similar model is also adopted for the thrust $F = \left(g + \left(K_z {}^W v_{z_d} - {}^W v_z\right)/\tau_z\right)/c\theta c\phi$ with τ_z, K_z denoting the parameters of the thrust controller and v_{z_d} the desired velocity for the z axis of $\mathbf{W}$.

It should be noted that although the inner attitude controller of the autopilot is incorporated into the adopted dynamic model, the outer position loop of the autopilot is substituted since the addition of the cable-suspended load and the need for controlling the cable's state, i.e., $\boldsymbol{\eta}_L$ and $\dot{\boldsymbol{\eta}}_L$, entail the design of a control scheme which considers the full dynamic model of Eq. 5. Towards this direction, an NMPC [2] is utilized in order to control the state of the UAV and the cable and compute desired setpoints for the inner attitude controller.

III. Problem Formulation

Consider a UAV with a cable-suspended load and a downward-facing DVS, mounted near the vehicle's CoM, as illustrated in Fig. 2. Given a stream of events $\mathcal{E} = \{e_i\}_{i=0}^{m}$, provided by the DVS, the ultimate objective is to:

- continuously identify the events that correspond to the cable in either indoor or outdoor environments, based on a point cloud event representation (Fig. 4) and an efficient processing algorithm, and, thus, capture the whole range of the cable's motion (Fig. 1) during the transportation
- and approximate the measurements of the pixel coordinates of the lower cable's edge (u_L, v_L), obtained according to the above-mentioned processing pipeline, by a Bézier curve in order to eventually estimate the complete state of the cable, i.e., $\boldsymbol{\eta}_L$ and $\dot{\boldsymbol{\eta}}_L$, required for any controller with feedback of the cable's state.

In order to close the loop and evaluate the performance of our method in outdoor conditions, an NMPC is deployed so as to achieve the transportation of the UAV with minimum swing.

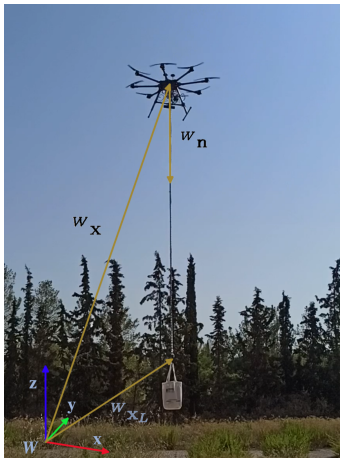


Fig. 3: The octorotor UAV with the cable-suspended load. The world frame $\mathbf{W}$, the positions ${}^W\mathbf{x}$ and ${}^W\mathbf{x}_L$ of the UAV and the load, and the cable unit vector ${}^W\mathbf{n}$ are also illustrated.

IV. Event-Based Estimation of the Cable's State

A. Event Representation

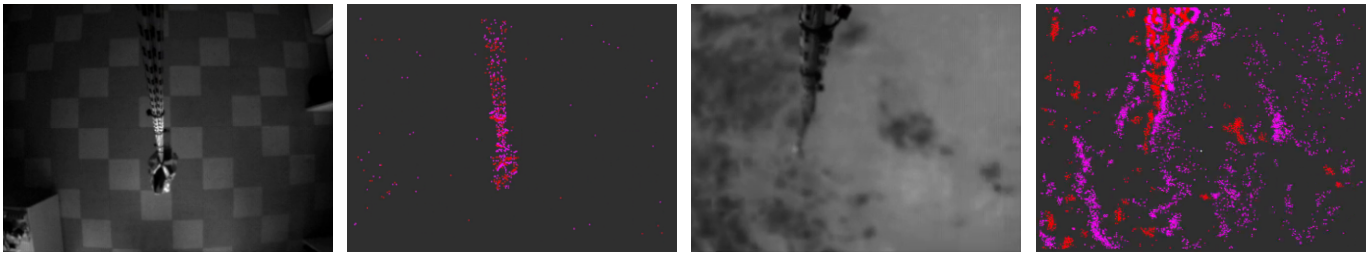
In contrast to frame-based conventional cameras, an event camera is a bio-inspired sensor which responds to luminance changes asynchronously and independently for each pixel. Hence, the output of a DVS camera is an asynchronous stream of "events" triggered by brightness changes in the scene. More precisely, an event $e_k = (u_k, v_k, t_k, p_k)$ is triggered at pixel coordinates (u_k, v_k) and at time t_k as soon as a change in log luminance above a predefined threshold is sensed compared to the last event at the same pixel. The polarity $p_k \in \{0, 1\}$ indicates the direction of luminance change.

According to the proposed algorithm, the events generated by the DVS are preprocessed in order to identify the ones that correspond to the cable and extract later significant information about its state. More precisely, as soon as an event e_i is triggered, it is stored in a point cloud $\mathcal{P} = \{e_i | t_i \in [t_0 - T, t_0]\}$, which consists of the most recent events up to the current time instant t_0. It is mentioned that both spatiotemporal information (u, v, t) and polarity p are stored in the point cloud. Indicative examples of "event" point clouds, along with the corresponding grayscale images captured by the DVS, are illustrated in Fig. 4, where an increased number of events, corresponding to the background or to noise, is observed in outdoor environments. The Point Cloud Library (PCL) [28], which distinguishes itself for its computational efficiency, is exploited for the representation of the "event" point cloud and the subsequent processing algorithms. Regarding the representation, a point cloud with points of type XYZI is selected from the PCL, where the X-Y coordinates are used for the pixel values, the Z coordinate for the time, and the intensity value for the polarity of the events.

B. Event Point Cloud Processing

The aforementioned point cloud $\mathcal{P}$ is downsampled by applying a 3D voxel grid filter within a spatiotemporal neighborhood of each event, defined by an $n_u \times n_v$ pixel window by n_t ms volume. Hence, the events that belong to each voxel are approximated by their centroid. It is highlighted that, besides the pixel coordinates and the timestamps, the polarity

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.



(a) (b) (c) (d)

Fig. 4: The grayscale images captured by the DVS in an (**a**) indoor and (**c**) outdoor environment at the current time instant t_0. A top-down view (u-v plane) of the corresponding "event" point clouds $\mathcal{P}$ for $T = 25\ ms$ in the (**b**) indoor and (**d**) outdoor environment. The purple and red colored points of $\mathcal{P}$ correspond to events with polarities $p = 1$ and $p = 0$ respectively.

of the events in each voxel is also averaged. Consequently, the newly generated point cloud $\mathcal{P}' = \left\{e'_j | t'_j \in [t_0 - T, t_0]\right\}$ consists of centroids e'_j with real-valued polarities $p'_j \in [0, 1]$, as illustrated in Fig. 5a.

Afterward, the point cloud $\mathcal{P}'$ is filtered based on the average polarity p'_j of each centroid e'_j. More specifically, only the centroids with polarity $p'_j \in (0, 1)$, i.e., $p'_j \neq 0$ and $p'_j \neq 1$, are considered as candidate events to belong to the cable, as depicted in Fig. 5b. This filtering step essentially implies that there should be at least 2 events e_i in each voxel of the point cloud $\mathcal{P}$ with opposite polarities in order to promote the respective centroid e'_j to a candidate event. This filtering approach is derived from the observation that, due to either the motion of the vehicle or the cable itself, an adequate number of events of opposite polarities is generated on a ms scale during the transportation of the suspended load. Additionally, when the UAV hovers at a specific position, both the UAV and the cable cannot be firmly stable due to existence of external disturbances, arising mainly from the wind in outdoor conditions. Another advantage of this filter is that the majority of noisy and spurious events are disregarded. More specifically, after an event is triggered, additional events of the same polarity and at the same pixel location are sometimes generated within a certain time interval [20]. Consequently, by applying the aforementioned filtering step, these same-polarity events are not promoted.

However, there exist some centroids e'_j that pass the above-mentioned filter but correspond to either noise or objects that lie on the terrain/background and not to the cable. Thus, an Euclidean Clustering algorithm in the spatiotemporal space is performed using the effective Kd-tree representation for the input point cloud $\mathcal{P}'' = \left\{e'_j | t'_j \in [t_0 - T, t_0], p'_j \in (0, 1)\right\}$ and subsequently employing the nearest neighbor search algorithm. It is highlighted that the cluster tolerance should be properly selected in order to avoid the cable being detected as multiple sub-clusters. Eventually, the cluster with the maximum size is promoted to the cable cluster $\mathcal{C}$ since the cable is closer to the DVS than any other object on the terrain and is assumed to have a sufficient diameter. Thus, the events that represent the cable typically occupy most of the image plane, as depicted in Fig. 1, 5c. Further checks are also employed between consecutive extracted cable clusters by comparing their relative position and, thus, ensuring the proper selection of the cable cluster when objects of similar shape lie on the terrain.

As soon as the cable cluster $\mathcal{C}$ is extracted, it is essential to find the event that corresponds to the load, i.e., the lower edge of the cable. Based on the specific configuration and relative pose of the DVS with respect to the cable, a naive approach would be to pick the event $e'_j \in \mathcal{C}$ with the maximum coordinate v'_j; however, this approach degrades the importance of the remaining events. Consequently, a 2D line fitting is performed considering the pixel coordinates $\left(u'_j, v'_j\right)$ of the events $e'_j \in \mathcal{C}$, as illustrated in Fig. 5c.

C. Approximation by a Bézier Curve

By employing the aforementioned processing pipeline, the pixel coordinates (u_L, v_L) of the lower edge of the cable are eventually obtained. However, the lower part of the cable might not generate an adequate number of events during some sparse iterations, which results in a noisy estimate of the cable's state and degrades the control performance. Consequently, Bézier curves are exploited in order to both smoothly approximate the coordinates (u_L, v_L) and obtain the pixel velocities $(\dot{u}_L, \dot{v}_L)$, required for the estimation of the complete state of the cable.

More specifically, a Bézier curve is given, according to the order n of the curve, as follows:

$$\mathbf{B}(\tau) = \sum_{i=0}^{n} b_{i,n}(\tau)\mathbf{c}_i \tag{6}$$

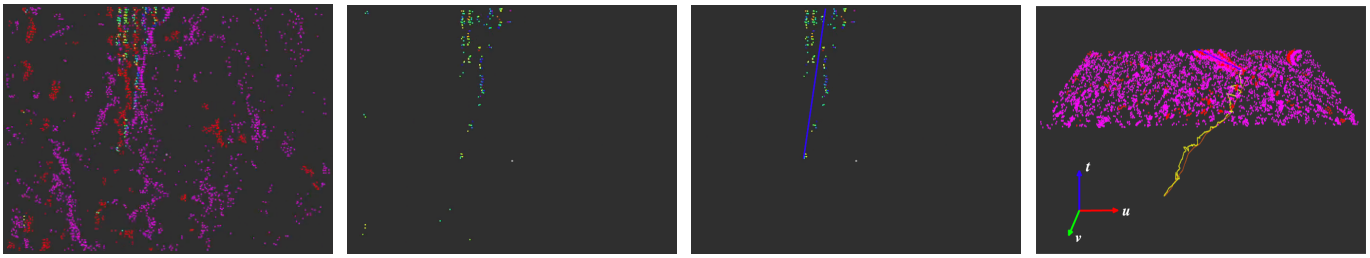
where $\tau \in \left[0, 1\right]$, $\mathbf{B}(\tau) \in \mathbb{R}^2$ (for the u-v image plane) is the Bézier curve, $b_{i,n}(\tau), i = 0, \cdots, n$ are the Bernstein basis polynomials of degree n, and $\mathbf{c}_i = \begin{bmatrix} c_{u,i} & c_{v,i} \end{bmatrix}^T \in \mathbb{R}^2$ are the $n + 1$ control points.

In order to compute the suitable set of control points $\mathbf{c} = \left[\mathbf{c}_0, \mathbf{c}_1, \cdots, \mathbf{c}_n\right]$ and, thus, approximate the trajectory of the cable in the image plane, a dataset $D = \{(t_{N-1}, u_{L,N-1}, v_{L,N-1}), \cdots, (t_0, u_{L,0}, v_{L,0})\}$ is constructed, consisting of the last N measurements, up to the current time instant t_0, of the pixel coordinates (u_L, v_L) along with the corresponding timestamp, as obtained by the processing pipeline of Section IV-B. The dataset has a constant length and is updated once a new measurement is available.

After performing a mapping between the parameter $\tau \in \left[0, 1\right]$ and the time interval $t \in \left[t_{N-1}, t_0\right]$, the optimal set of control points $\mathbf{c}$ is the solution of the optimization problem:

$$\min_{\mathbf{c}} \sum_{i=0}^{N-1} \left(w_{t_i} \left\| \mathbf{B}(\tau_i) - \begin{bmatrix} u_{L,i}, v_{L,i} \end{bmatrix}^T \right\|_2^2 \right) \tag{7}$$

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.



(a) (b) (c) (d)

Fig. 5: (**a**), (**b**), (**c**) The processing algorithm applied to the "event" point cloud $\mathcal{P}$ of Fig. 4d according to Section IV-B: (**a**) A top-down view (u-v plane) of the point cloud $\mathcal{P}'$ generated after the downsampling of $\mathcal{P}$. The purple and red colored centroids of $\mathcal{P}'$ correspond to centroids with polarities $p = 1$ and $p = 0$ respectively while the remaining centroids have polarities $p' \in (0, 1)$. (**b**) A top-down view of the point cloud $\mathcal{P}''$ after the filtering of $\mathcal{P}'$ based on the polarity of each centroid. (**c**) A top-down view of the cable cluster $\mathcal{C}$ and the respective blue colored line fitted to $\mathcal{C}$. (**d**) The Bézier curve (orange color) which approximates the measurements of the lower edge of the cable (yellow color), obtained according to Section IV-C.

which essentially minimizes the distance between the Bézier curve and the measurements, while the weights w_{t_i} boost the more recent measurements and impose penalties on the previous ones [29].

Moreover, constraints, related to the minimum and maximum velocity and acceleration of the cable in the image plane, are incorporated into the optimization problem, so as to find solely feasible solutions. More precisely, by differentiating the Bézier curve, the constraints are formulated as follows for the u axis (similarly for the v one):

$$\begin{gathered} n \cdot (c_{u,i} - c_{u,i-1}) \,/\, (t_0 - t_{N-1}) \in [-\dot{u}_{max}, \dot{u}_{max}] \\ \frac{n \cdot (n-1) \cdot (c_{u,i} - 2c_{u,i-1} + c_{u,i-2})}{(t_0 - t_{N-1})^2} \in [-\ddot{u}_{max}, \ddot{u}_{max}] \end{gathered} \tag{8}$$

The above-mentioned constrained Quadratic Programming problem (Eq. 7) with $2 \cdot (n+1)$ primal variables, i.e., the control points, and $2 \cdot n + 2 \cdot (n-1)$ constraints (Eq. 8) is solved with the aid of the OOQP software [30] and, eventually, an estimate of the pixel coordinates (u_L, v_L) and velocities $(\dot{u}_L, \dot{v}_L)$ at each current time instant t_0 is derived from the Bézier curve, as illustrated in Fig. 5d.

D. Estimation of the Cable's State

Given the Bézier curve, the complete state of the cable is computed. More precisely, based on the pinhole camera model, the position of the lower cable's edge ${}^C\mathbf{x}_L \in \mathbb{R}^3$ w.r.t. the camera frame $\mathbf{C}$ is extracted from the pixel coordinates $u_L \in [-W/2, W/2]$, $v_L \in [-H/2, H/2]$, obtained by the Bézier curve, as follows:

$${}^C\mathbf{x}_L = \begin{bmatrix} {}^C z_L u_L / f & {}^C z_L v_L / f & {}^C z_L \end{bmatrix}^T \tag{9}$$

where $W \times H$ is the resolution of the camera, f is the focal length of the camera and ${}^C z_L$ is the cartesian coordinate of the cable's edge w.r.t. the z axis of $\mathbf{C}$. Since the distance ${}^C z_L$ is unknown, the constant length l of the cable is exploited. Specifically, given that the cable is assumed to be taut and attached to the origin of the body-fixed frame $\mathbf{B}$, the following equation is derived:

$$\left\| {}^B\mathbf{x}_L \right\| = \left\| \mathbf{R}_{BC} \, {}^C\mathbf{x}_L + \mathbf{t}_{BC} \right\| = l \tag{10}$$

where ${}^B\mathbf{x}_L$ is the position of the cable's edge w.r.t. $\mathbf{B}$, and $\mathbf{R}_{BC}$ and $\mathbf{t}_{BC}$ are the known rotation matrix and translation vector from $\mathbf{C}$ to $\mathbf{B}$. Solving Eq. 10 results in a function $h : \mathbb{R}^2 \mapsto \mathbb{R}$ that maps the coordinates (u_L, v_L) – and known parameters, namely, the elements of $\mathbf{R}_{BC}$ and $\mathbf{t}_{BC}$, and the cable's length l – with the distance ${}^C z_L$, i.e., ${}^C z_L = h\,(u_L, v_L)$. The analytical form of the function h is omitted since it relies on the specific relative pose between the DVS and the body-fixed frame. Eventually, the cable unit vector ${}^B\mathbf{n}$ is computed:

$${}^B\mathbf{n} = \frac{{}^B\mathbf{x}_L}{\|{}^B\mathbf{x}_L\|} = \frac{\mathbf{R}_{BC} \, {}^C\mathbf{x}_L + \mathbf{t}_{BC}}{l} \tag{11}$$

Regarding the cable's velocity, it is obtained by differentiating Eq. 11 according to the following relationships:

$${}^B\dot{\mathbf{n}} = \frac{\mathbf{R}_{BC}}{l} \, {}^C\dot{\mathbf{x}}_L = \frac{\mathbf{R}_{BC}}{l} \frac{1}{f} \begin{bmatrix} {}^C\dot{z}_L u_L + {}^C z_L \dot{u}_L \\ {}^C\dot{z}_L v_L + {}^C z_L \dot{v}_L \\ f \, {}^C\dot{z}_L \end{bmatrix} \tag{12}$$

$${}^B\dot{\mathbf{n}} = \frac{\mathbf{R}_{BC}}{l \cdot f} \begin{bmatrix} \left(\frac{\partial h}{\partial u_L} \dot{u}_L + \frac{\partial h}{\partial v_L} \dot{v}_L \right) u_L + {}^C z_L \dot{u}_L \\ \left(\frac{\partial h}{\partial u_L} \dot{u}_L + \frac{\partial h}{\partial v_L} \dot{v}_L \right) v_L + {}^C z_L \dot{v}_L \\ f \left(\frac{\partial h}{\partial u_L} \dot{u}_L + \frac{\partial h}{\partial v_L} \dot{v}_L \right) \end{bmatrix} \tag{13}$$

with $\frac{\partial h}{\partial u_L}$, $\frac{\partial h}{\partial v_L}$ denoting the partial derivatives of the function h. Consequently, given the output of the Bézier curve, i.e., the coordinates (u_L, v_L) and the velocities $(\dot{u}_L, \dot{v}_L)$, both the cable unit vector ${}^B\mathbf{n}$ and the cable's velocity ${}^B\dot{\mathbf{n}}$ can be computed and then transformed to the world frame $\mathbf{W}$ so as to calculate the angles $\boldsymbol{\eta}_L$ and velocities $\dot{\boldsymbol{\eta}}_L$ (Eq. 2 and 3).

V. NMPC Scheme

Throughout the outdoor experiment, the cable's angles $\boldsymbol{\eta}_L$ and angular velocities $\dot{\boldsymbol{\eta}}_L$, obtained according to the Section IV, are fed into an NMPC scheme [2] in order to achieve the transportation of the UAV towards reference 3D positions ${}^W\mathbf{x}_{ref}$ while simultaneously minimizing the swinging motion of the cable. Towards this direction, the following Optimal Control Problem (OCP) is formulated:

$$\begin{aligned} \min_{\mathbb{U}} \int_{t_0}^{t_0+t_p} & \left(\|\mathbf{x}(t) - \mathbf{x}_{ref}(t)\|^2_{\mathbf{Q}} + \|\boldsymbol{\upsilon}(t)\|^2_{\mathbf{R}} \right) \mathrm{d}t \\ & + \|\mathbf{x}(t_0 + t_p) - \mathbf{x}_{ref}(t_0 + t_p)\|^2_{\mathbf{P}} \\ \text{s.t.: } & \mathbf{x}(\mathbf{t_0}) = \mathbf{x_0}, \;\; \dot{\mathbf{x}} = f(\mathbf{x}, \boldsymbol{\upsilon}) \text{ (Eq. 5)}, \;\; \boldsymbol{\upsilon} \in \mathbb{U} \end{aligned} \tag{14}$$

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

where $\mathbf{x} = \begin{bmatrix} {}^W\mathbf{x}^T & {}^W\mathbf{v}^T & \phi & \theta & \psi & \boldsymbol{\eta}_L^T & \dot{\boldsymbol{\eta}}_L^T \end{bmatrix}^T \in \mathbb{R}^{13}$, $\boldsymbol{\upsilon} = \begin{bmatrix} \phi_d & \theta_d & \psi_d & {}^W v_{z_d} \end{bmatrix}^T \in \mathbb{R}^4$ are the state and input vectors, t_p is the time horizon, $\mathbb{U}$ is the set of input constraints, $\mathbf{x}_{ref} = \begin{bmatrix} {}^W\mathbf{x}_{ref}^T & \mathbf{0}_{1\times3} & 0 & 0 & \psi_{ref} & \mathbf{0}_{1\times2} & \mathbf{0}_{1\times2} \end{bmatrix}^T$ is the reference state vector, and $\mathbf{Q}$, $\mathbf{R}$, and $\mathbf{P}$ are the block-diagonal state error, input, and terminal state error cost matrices.

VI. Experimental Results

A. Hardware Setup & Computational Cost

In order to validate the performance of the proposed method, indoor and outdoor experiments were carried out. During the outdoor experiments, the octorotor vehicle of Fig. 2, equipped with a downward-facing DAVIS 346 event camera, was utilized. The UAV is additionally equipped with the embedded computer Jetson AGX Xavier which communicates with the autopilot according to the MAVLink protocol [31] and, thus, the state of the UAV is available. Moreover, a load cell, located at the cable's lower edge, is exploited in order to measure the norm of the cable's tension $\mathbf{T}$, required for the NMPC scheme. As for the load, a bag with a load of mass 0.5 kg is utilized, while the length of the cable is equal to $l = 3.8$m and $l = 1.7$m during the outdoor and indoor experiments respectively.

	Time[ms]	
Step	**Indoor**	**Outdoor**
Point Cloud Processing	3.424	8.091
Bézier curve ($N = 100$)	0.302	0.337

TABLE I: The computational cost of the individual processing steps that comprise the proposed method.

Parameter	**Indoor**	**Outdoor**
T[ms]	25	25
$n_u \times n_v \times n_t$ [px × px × ms]	5×5×10	10×3×5

TABLE II: The parameters of the proposed method.

Regarding the computational cost of the proposed method, the execution time of the individual processing steps of the algorithm is measured on the Jetson AGX Xavier in the case of both indoor and outdoor environments. The mean processing time for each of the steps that comprise the overall method, i.e., the processing of the "event" point cloud, and the approximation by a Bézier curve, as well as the parameters of the proposed algorithm are summarized in Table I and II respectively. It is mentioned that the computational cost of the point cloud processing is directly related to the number of incoming events. Consequently, during the indoor experiments, where the DVS is not moving and the surrounding environment is static, the execution time of this step is reduced. In any case, the mean processing time of the proposed method is significantly lower compared to a standard frame-based detection algorithm with Convolutional Neural Networks (CNN), where the average time for the inference of the CNN is equal to 33 ms [2]. As for the NMPC scheme, the respective OCP is solved within approximately 3 ms using the ACADO Toolkit [32] and the qpOASES solver [33].

B. Indoor Experiments

The efficacy of the proposed method was initially evaluated through indoor experiments, where the output of Section IV was compared against accurate measurements provided by a motion capture system. More precisely, the DVS and the upper edge of the cable were rigidly mounted on a ceiling with a relative pose similar to the one on the UAV. It is mentioned that in this case, the world frame coincides with the body-fixed one. Ground truth was obtained by the Vicon motion capture system based on markers that were cautiously placed around the cable to ensure the continuous tracking of its swinging motion. An image of the indoor environment, acquired from the frame-based camera of the DAVIS 346, and the respective "event" point cloud are illustrated in Fig. 4a and 4b respectively.

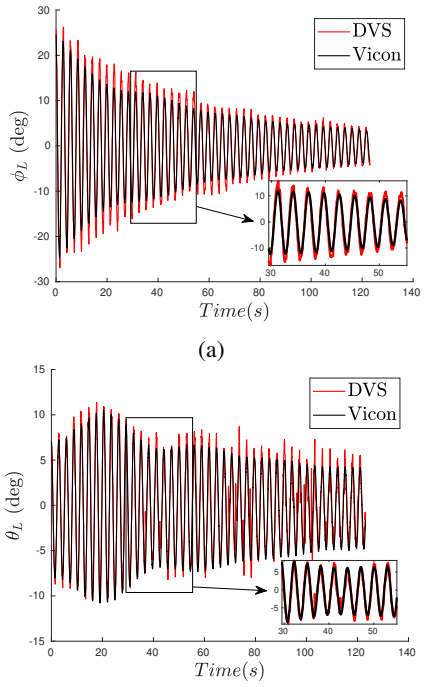
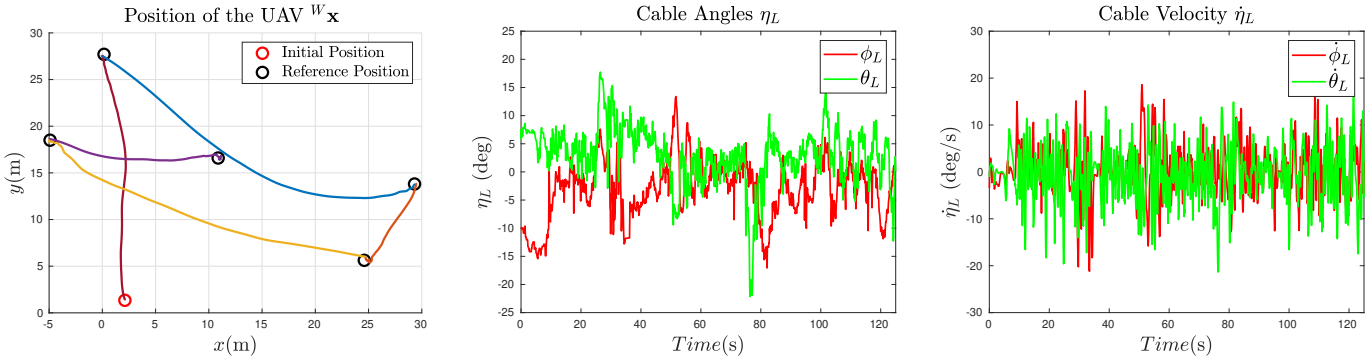


Fig. 6: The cable's angles, (**a**) ϕ_L and (**b**) θ_L, computed according to the algorithm explained in Section IV and compared against the accurate measurements provided by the Vicon system during the indoor experiment.

Throughout the experiments, data from both the DVS and the Vicon system were gathered by varying the initial angle of the load and releasing it with zero velocity. Hence, the load moved freely under the influence of gravity. Indicative results for an indoor experiment are demonstrated in Fig. 6 where the cable's angles $\boldsymbol{\eta}_L = \begin{bmatrix} \phi_L & \theta_L \end{bmatrix}^T$ obtained by the processing of the event stream are compared against the ground truth captured by the Vicon system. It is evident that our method successfully distinguishes the cable and captures the entire range of its swinging motion. The Root Mean Square Errors (RMSE) for the angles ϕ_L and θ_L were 2.204^o and 2.639^o correspondingly during the indoor tests. Finally, the success rate, i.e., the number of detections divided by the total number of iterations, was equal to 96.67%.

However, it should be highlighted that although our method captures the frequency of the cable's swinging motion, the magnitude of the cable's angles is overestimated, especially when the angles considerably move away from the zero



(a) (b) (c)

Fig. 7: (**a**) The position ${}^{W}\mathbf{x}$ of the UAV and the reference positions ${}^{W}\mathbf{x}_{ref}$ in the x-y plane, (**b**) the cable's angle $\boldsymbol{\eta}_L$, and (**c**) velocity $\dot{\boldsymbol{\eta}}_L$ during the outdoor experiment.

degrees. This behavior is mainly due to the fact that the lowest part of the cable may not generate an adequate number of events and, hence, our method may identify as the edge of the cable (u_L, v_L) a centroid that does not actually correspond to the edge but instead to a point located a few centimeters above. Nevertheless, the RMSEs for the angles are relatively small while also the ultimate goal of our work is the aerial transportation with minimum swing angles, where the aforementioned behavior is alleviated.

C. Outdoor Experiments

Following the validation in an indoor environment, the proposed algorithm was deployed on the octorotor with the cable-suspended load in order to test the performance of the method against real-world and challenging conditions, commonly encountered in outdoor environments. Despite the increased number of events, generated by the motion of the vehicle and corresponding to the surrounding environment, the proposed method reliably identified the events that were related to the swinging motion of the cable and provided robust feedback of the complete cable's state, fed into the NMPC.

Given a set of reference 3D positions ${}^{W}\mathbf{x}_{ref}$ and the cable's state, i.e., $\boldsymbol{\eta}_L$ and $\dot{\boldsymbol{\eta}}_L$, obtained by the processing of the event stream, the NMPC scheme computed desired setpoints for the inner attitude subsystem of the autopilot in order to minimize the error ${}^{W}\mathbf{x}_{ref} - {}^{W}\mathbf{x}$ and the vehicle fulfilled the mission, as depicted in Fig. 7a. Additionally, throughout the experiment, the swinging motion of the cable, estimated by the proposed method, was maintained below critical values (Fig. 7), since the upper part of the cable did not approach the structure of the UAV, specifically the legs of the platform, and the safety of the system was guaranteed.

In order to further investigate the reliability of the proposed algorithm, a second experiment was conducted where a mission consisting of more reference positions ${}^{W}\mathbf{x}_{ref}$ was commanded. Similarly, the robust identification of the cable during the mission and the transportation of the suspended load with minimum swinging motion of the cable were attained. Both experiments are better illustrated in the following video https://youtu.be/jFUQcZ0OhoM.

D. Comparative Experimental Study

In order to demonstrate the advantages of using an event camera for the task of aerial transportation of cable-suspended loads, a comparative study was conducted between the proposed method and a frame-based one [2], where a CNN is applied to the image captured by the downward-looking ZED 2 camera so as to detect the cable. Throughout the comparative study, the same mission, composed of 4 reference positions, was commanded and the feedback of the cable's state, provided by the two methods, was fed into the NMPC scheme of Section V. It is highlighted that the parameters of the controller, i.e., the cost matrices and the input limits, were identical. Additionally, the experiments were realized successively in order to ensure similar environmental conditions and, hence, a fair comparison.

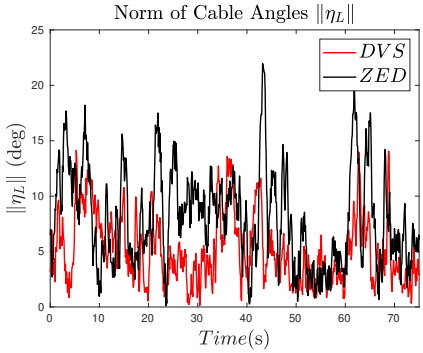


Fig. 8: The norm of the cable's angle $\|\boldsymbol{\eta}_L\|$ during the comparative study between the proposed method (DVS) and the frame-based solution (ZED) [2].

The two methods were compared based on the efficiency of the controller and, more specifically, its ability to minimize the norm of the cable's angle $\|\boldsymbol{\eta}_L\|$. As illustrated in Fig. 8, the rapid and reliable feedback provided by the DVS resulted in a better performance of the controller, whereas greater swing angles were observed with the 30 FPS frame-based approach. More precisely, the maximum cable's angle $\|\boldsymbol{\eta}_L\|$ for the proposed method and the frame-based one were equal to 14.13^o and 21.95^o respectively. Therefore, in the aerial transportation of cable-suspended loads, high-speed feedback can significantly impact the performance of the controller and, thus, the utilization of an event camera can further improve the efficiency of existing control schemes.

VII. CONCLUSIONS

In this work, we presented an event processing algorithm, based on a point cloud representation and a Bézier curve, which efficiently detects a cable, attached to a UAV, and estimates its swinging motion during the aerial transportation of a load. We experimentally validated our approach in an indoor environment and we subsequently deployed our algorithm in a real-world scenario where a closed-loop NMPC scheme was exploited in order to achieve the transportation of a load suspended to an octorotor through a cable with minimum swinging motion. We also compared our method with a frame-based solution so as to demonstrate the advantages of using event cameras instead of standard ones.

Regarding our future work, we aim to investigate methods for improving the accuracy of the algorithm such as the incorporation of an event tracker, the selection of regions of interest (ROIs) based on the cable's motion, or the sensor fusion with an IMU. Furthermore, we intend to integrate our method into an NMPC scheme with field-of-view constraints.

REFERENCES

[1] F. Panetsos, G. C. Karras, S. N. Aspragkathos, and K. J. Kyriakopoulos, "Precise position control of a multi-rotor uav with a cable-suspended mechanism during water sampling," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 1780–1786.

[2] F. Panetsos, G. C. Karras, K. J. Kyriakopoulos, O. Oikonomides, P. Kolios, D. Eliades, and C. Panayiotou, "A motion control framework for autonomous water sampling and swing-free transportation of a multirotor uav with a cable-suspended mechanism," *Journal of Field Robotics*. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.22182

[3] M. Schwarzbach, M. Laiacker, M. Mulero-Pázmány, and K. Kondak, "Remote water sampling using flying robots," in *2014 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2014, pp. 72–76.

[4] D. Villa, A. Brandao, and M. Sarcinelli-Filho, "A survey on load transportation using multirotor uavs," *Journal of Intelligent & Robotic Systems*, vol. 98, pp. 1–30, 05 2020.

[5] I. Palunko, R. Fierro, and P. Cruz, "Trajectory generation for swing-free maneuvers of a quadrotor with suspended payload: A dynamic programming approach," in *2012 IEEE International Conference on Robotics and Automation*, 2012, pp. 2691–2697.

[6] A. Faust, I. Palunko, P. Cruz, R. Fierro, and L. Tapia, "Learning swing-free trajectories for uavs with a suspended load," in *2013 IEEE International Conference on Robotics and Automation*, 2013, pp. 4902–4909.

[7] F. Panetsos, G. C. Karras, and K. J. Kyriakopoulos, "A deep reinforcement learning motion control strategy of a multi-rotor uav for payload transportation with minimum swing," in *2022 30th Mediterranean Conference on Control and Automation (MED)*, 2022, pp. 368–374.

[8] K. Sreenath, T. Lee, and V. Kumar, "Geometric control and differential flatness of a quadrotor uav with a cable-suspended load," in *52nd IEEE Conference on Decision and Control*, 2013, pp. 2269–2274.

[9] F. Goodarzi, D. Lee, and a. Lee, "Geometric control of a quadrotor uav transporting a payload connected via flexible cable," *International Journal of Control Automation and Systems*, 07 2014.

[10] S. Tang and V. Kumar, "Mixed integer quadratic program trajectory generation for a quadrotor with a cable-suspended payload," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 2216–2222.

[11] M. E. Guerrero, D. A. Mercado, R. Lozano, and C. D. García, "Passivity based control for a quadrotor uav transporting a cable-suspended payload with minimum swing," in *2015 54th IEEE Conference on Decision and Control (CDC)*, 2015, pp. 6718–6723.

[12] S. Tang, V. Wüest, and V. Kumar, "Aggressive flight with suspended payloads using vision-based control," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 1152–1159, 2018.

[13] X. Liang, Y. Fang, N. Sun, and H. Lin, "Nonlinear hierarchical control for unmanned quadrotor transportation systems," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 4, pp. 3395–3405, 2018.

[14] P. Foehn, D. Falanga, N. Kuppuswamy, R. Tedrake, and D. Scaramuzza, "Fast trajectory optimization for agile quadrotor maneuvers with a cable-suspended payload," in *Robotics: Science and Systems*, 2017.

[15] G. Yu, D. Cabecinhas, R. Cunha, and C. Silvestre, "Nonlinear backstepping control of a quadrotor-slung load system," *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 5, pp. 2304–2315, 2019.

[16] Y. Ren, J. Liu, H. Chen, and Y. Liu, "Vision-encoder-based payload state estimation for autonomous mav with a suspended payload," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 9632–9638.

[17] M. Bernard, K. Kondak, I. Maza, and A. Ollero, "Autonomous transportation and deployment with aerial robots for search and rescue missions," *Journal of Field Robotics*, vol. 28, no. 6, pp. 914–931, 2011. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.20401

[18] S. J. Lee and H. J. Kim, "Autonomous swing-angle estimation for stable slung-load flight of multi-rotor uavs," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 4576–4581.

[19] G. Gallego, T. Delbruck, G. Orchard, C. Bartolozzi, B. Taba, A. Censi, S. Leutenegger, A. J. Davison, J. Conradt, K. Daniilidis, and D. Scaramuzza, "Event-based vision: A survey," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 44, no. 01, pp. 154–180, jan 2022.

[20] A. Dietsche, G. Cioffi, J. Hidalgo-Carrió, and D. Scaramuzza, "Powerline tracking with event cameras," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 6990–6997.

[21] F. Tschopp, C. von Einem, A. Cramariuc, D. Hug, A. W. Palmer, R. Siegwart, M. Chli, and J. Nieto, "Hough^{2}map – iterative event-based hough transform for high-speed railway mapping," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2745–2752, 2021.

[22] A. G. Eguíluz, J. Rodríguez-Gómez, R. Tapia, F. Maldonado, J. Acosta, J. Martínez-de Dios, and A. Ollero, "Why fly blind? event-based visual guidance for ornithopter robot flight," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 1958–1965.

[23] J. Rodríguez-Gomez, A. G. Eguíluz, J. Martínez-de Dios, and A. Ollero, "Asynchronous event-based clustering and tracking for intrusion monitoring in uas," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 8518–8524.

[24] S. Sun, G. Cioffi, C. de Visser, and D. Scaramuzza, "Autonomous quadrotor flight despite rotor failure with onboard vision sensors: Frames vs. events," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 580–587, 2021.

[25] F. Mahlknecht, D. Gehrig, J. Nash, F. M. Rockenbauer, B. Morrell, J. Delaune, and D. Scaramuzza, "Exploring event camera-based odometry for planetary robots," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 8651–8658, 2022.

[26] A. P. group. (2016) Ardupilot documentation. [Online]. Available: https://ardupilot.org/ardupilot/

[27] M. Kamel, M. Burri, and R. Siegwart, "Linear vs nonlinear mpc for trajectory tracking applied to rotary wing micro aerial vehicles," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 3463–3469, 2017, 20th IFAC World Congress. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896317313083

[28] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, May 9-13 2011.

[29] Z. Han, R. Zhang, N. Pan, C. Xu, and F. Gao, "Fast-tracker: A robust aerial system for tracking agile target in cluttered environments," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 328–334.

[30] E. M. Gertz and S. J. Wright, "Object-oriented software for quadratic programming," *ACM Trans. Math. Softw.*, vol. 29, no. 1, p. 58–81, mar 2003. [Online]. Available: https://doi.org/10.1145/641876.641880

[31] M. M. A. V. Protocol. (2013) Mavlink to ros gateway with proxy for ground control station. [Online]. Available: https://github.com/mavlink/mavros

[32] B. Houska, H. Ferreau, and M. Diehl, "ACADO Toolkit – An Open Source Framework for Automatic Control and Dynamic Optimization," *Optimal Control Applications and Methods*, vol. 32, no. 3, pp. 298–312, 2011.

[33] J. Ferreau, C. Kirches, A. Potschka, H. Bock, and M. Diehl, "qpoases: A parametric active-set algorithm for quadratic programming," *Mathematical Programming Computation*, vol. 6, 12 2014.