



HAL
open science

MAV tele-operation constrained on virtual surfaces for inspection of infrastructures

Florian Dietrich, Julien Marzat, Martial Sanfourche, Sylvain Bertrand,
Anthelme Bernard-Brunel, Alexandre Eudes

► **To cite this version:**

Florian Dietrich, Julien Marzat, Martial Sanfourche, Sylvain Bertrand, Anthelme Bernard-Brunel, et al.. MAV tele-operation constrained on virtual surfaces for inspection of infrastructures. ETFA 2020, Sep 2020, VIENNE, Austria. hal-03027471

HAL Id: hal-03027471

<https://hal.science/hal-03027471>

Submitted on 27 Nov 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

MAV tele-operation constrained on virtual surfaces for inspection of infrastructures

Florian Dietrich, Julien Marzat, Martial Sanfourche, Sylvain Bertrand, Antheleme Bernard-Brunel, Alexandre Eudes

Abstract—This paper presents a tele-operation system that enables a MAV to be controlled on virtual surfaces by an unskilled operator using high-level inputs. These virtual surfaces can be placed relatively to the infrastructure to be inspected, in order to ensure safety of the flight and repeatability of the acquisition conditions of the inspection data (e.g. at a constant distance from the infrastructure). The architecture, interface and embedded controller of the tele-operation system are described, and results from flight experiments in an industrial warehouse are provided for three typical inspection scenarios of infrastructures.

Index Terms—Micro Air Vehicle, tele-operation, inspection of infrastructures, geofencing, constrained motion, flight tests.

I. INTRODUCTION

Regular inspections of industrial infrastructures are necessary for default detection, predictive maintenance or updates of digital twins for Building Information Modeling. They require data to be collected at regular time intervals, with similar acquisition conditions and under coverage constraints. Such missions can be difficult for human operators since they can be time consuming, repetitive, and require accuracy and concentration. In case of tall infrastructures, inspection tasks are also prone to be dangerous for human operators and additional time and cost consuming constraints have to be imposed to ensure safety of the operations. In this context, Micro Air Vehicles (MAVs) have naturally emerged as promising tools for replacing or assisting human operators in inspection missions [1]–[7]. Different levels of automation can be considered for the MAV, from being remotely controlled by a telepilot to being able to perform autonomously the mission. In the first case, the success of the mission still relies on a human operator with almost the same constraints and difficulties as mentioned before. In addition, it requires high qualification of the operator (telepilot) to be able to perform complex flights in possibly cluttered environments or close to the infrastructures to be inspected. Finally, no guarantees can be obtained on the accuracy or repeatability of the flight trajectory and hence on the data collected. In the second case, more complex perception and control architectures have to be designed to enable automatic flight without (or with sparse) intervention of a human operator. Some examples can be found in [8]–[10] and in previous work done by the authors for autonomous inspection of industrial warehouses [11], [12]. In this case, accuracy and repeatability can be ensured, but acceptance by other human operators and obtaining outdoor

flight authorizations from legal authorities can be problematic for autonomous vehicles. In addition, it can be useful in some cases to leave some degrees of freedom to the human operator for the control of the vehicle (e.g. to stop and take a picture or fly back for doubt removal in case of default detection). Therefore, assisted tele-operation [13]–[15] can be proposed as an intermediate level of automation for the MAV and result in a good trade-off between full automated and telepiloted flights. In the EU FP7 project INCASS [16], a control software was proposed for the safe tele-operation of a MAV to carry out the inspection of cargo vessels according to the Supervised Autonomy paradigm which manages collision avoidance so that the surveyor can remain focused on the inspection itself. In a previous work [17] by the authors, a tele-operation system has been developed where high-level inputs (speed and yaw rate) are given by the operator (with no telepilot skills) to easily and safely control the motion of a MAV along a predefined reference trajectory. This assisted tele-operation scheme is of huge interest for inspection missions where repeatability is at stake. Nevertheless, in some cases the assignment constraint to the reference trajectory may be too stringent. For pillar or facade inspection for example, some additional degrees of freedom can be profitable to the operator. The contribution of this paper is a new tele-operation system that enables a MAV to be controlled on virtual surfaces by an unskilled operator. In geofencing or geocaging systems, e.g. [18], [19], virtual surfaces are traditionally used to limit the motion of the vehicle, forbidding entering or exiting predefined flight areas or volumes. In this work, the motion of the MAV is guaranteed to remain on the virtual surface along which the operator can freely guide it. Two types of surfaces, namely planes and cylinders, are considered in the proposed system which can be used for numerous typical inspection scenarios. Examples include inspection of pillars by flying around on a virtual cylinder, of facades by flying on a parallel virtual plane, and of more complex infrastructures (buildings, vehicles) by using several virtual planes connected together. The positioning of these surfaces with respect to the infrastructure will enable to control the conditions of inspection (e.g. constant distance) and the safety of the operations (motion restricted to a finite surface with limitations in its dimensions). In this system, the high-level inputs (tangential velocities and yaw rate) given by the operators are filtered to ensure that the motion remains on the surfaces. Localization of the MAV mainly relies on embedded vision, enabling uses in GPS-denied or GPS-perturbed environments. Finally, in case of virtual cylinders, a functionality is proposed to place

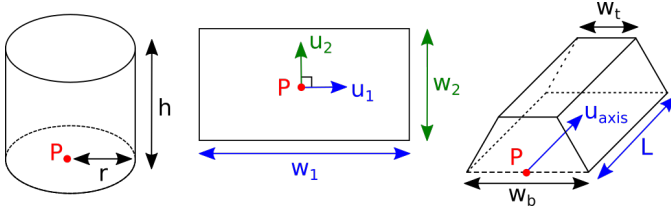


Fig. 1. Virtual surfaces \mathcal{C} , \mathcal{P} and \mathcal{T} and their parametrization.

automatically the surface around the detected vertical infrastructure to be inspected, using 3D perception. The embedded control architecture is presented in this paper along with field experiments for three typical scenarios of inspection in an industrial warehouse environment.

The paper is organized as follows. The next section presents the developed controller for surface-constrained tele-operation. The MAV platform, the tele-operator interface and the on-board architecture also including perception and environment modeling functionalities are presented in Section III. Section IV provides results from field experiments in an industrial warehouse for three typical inspection scenarios respectively using a cylinder, a plane and a multi-plane surface to constrain the motion of the MAV. Conclusions and perspectives are finally given in the last section of the paper.

II. SURFACE-CONSTRAINED CONTROLLER

A virtual reference surface \mathcal{S} is specified by the user with respect to a point \mathbf{P} in an inertial frame $\mathcal{I} = (\mathbf{O}; \mathbf{e}_x, \mathbf{e}_y, \mathbf{e}_z)$. Three possible configurations (illustrated in Figure 1) have been considered, each requiring a specific set of user-defined parameters, as follows.

- 1) *Cylinder* \mathcal{C} of radius r and height h .
- 2) *Plane* \mathcal{P} of width w_1 , height w_2 and vectors $\mathbf{u}_1, \mathbf{u}_2$.
- 3) *Tunnel* \mathcal{T} as the junction of three planes, parametrized by trapezoidal base lengths w_b, w_t , length L and longitudinal vector \mathbf{u}_{axis} .

A frame $(\mathbf{P}; \mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$ is attached to the surface. For \mathcal{C} , it identifies with the rotating cylindrical frame $(\mathbf{P}; \mathbf{e}_r, \mathbf{e}_\theta, \mathbf{e}_h)$, and for \mathcal{P} with the fixed frame $(\mathbf{P}; \mathbf{u}_1 \wedge \mathbf{u}_2, \mathbf{u}_1, \mathbf{u}_2)$. For a tunnel \mathcal{T} , each sub-plane has a frame defined similarly to \mathcal{P} . The basis $(\mathbf{e}_2, \mathbf{e}_3)$ will also be mentioned as the tangential basis in the following. The MAV current inertial position and velocity in \mathcal{I} are respectively denoted by $\boldsymbol{\xi}$ and \mathbf{v} .

The surface controller computes a reference position $\boldsymbol{\xi}_{\text{des}}$ based on the current MAV position and the type of surface considered. This reference position is a specific (constant-altitude, boundary-constrained) projection of the current MAV position on the reference surface, by keeping the inertial altitude unchanged to avoid any height variation that can be encountered when using classical orthonormal projection on tilted planes (tilted cylinders were not considered here). The projection also enforces $\boldsymbol{\xi}_{\text{des}}$ on the surface boundaries.

For a cylinder \mathcal{C} , the reference position is expressed in the cylindrical frame as

$$\boldsymbol{\xi}_{\text{des}} = r \mathbf{e}_r + \xi_\theta \mathbf{e}_\theta + \text{sat}_0^h(\xi_h) \mathbf{e}_h, \quad (1)$$

with r the cylinder radius, ξ_θ and ξ_h the components of the MAV current position in the cylindrical frame such that

$$\boldsymbol{\xi} = \xi_r \mathbf{e}_r + \xi_\theta \mathbf{e}_\theta + \xi_h \mathbf{e}_h, \quad (2)$$

and with

$$\text{sat}_a^b(x) = \begin{cases} a, & \text{if } x < a, \\ b, & \text{if } x > b, \\ x, & \text{otherwise.} \end{cases} \quad (3)$$

Note that since the radial component of $\boldsymbol{\xi}_{\text{des}}$ is equal to the cylinder radius, $\boldsymbol{\xi}_{\text{des}}$ lies on the cylinder surface.

For a plane \mathcal{P} , the reference position is expressed in the surface frame as

$$\boldsymbol{\xi}_{\text{des}} = \text{sat}_{-\frac{1}{2}w_1}^{+\frac{1}{2}w_1}(\xi_{\text{proj},1}) \mathbf{u}_1 + \text{sat}_{-\frac{1}{2}w_2}^{+\frac{1}{2}w_2}(\xi_{\text{proj},2}) \mathbf{u}_2, \quad (4)$$

where $\xi_{\text{proj},1}$ and $\xi_{\text{proj},2}$ are the components in the plane basis of the unbounded and same-altitude projection of the MAV position on the surface. This particular projection is computed as follows. If \mathcal{P} is a vertical plane, then the constant altitude projection is equivalent to the standard orthonormal projection. If \mathcal{P} is an horizontal plane, then the constant altitude projection is not suitable, and the standard orthonormal projection is used. Otherwise, the projection is computed in the inertial frame by horizontally translating the MAV position as follows.

$$\boldsymbol{\xi}_{\text{proj}} = \boldsymbol{\xi} - \frac{d_{\mathcal{P}}(\boldsymbol{\xi})}{\|\mathbf{n}\|^2} \mathbf{n}, \quad (5)$$

where

$$d_{\mathcal{P}}(\boldsymbol{\xi}) = \langle \mathbf{e}_1, (\boldsymbol{\xi} - \mathbf{P}) \rangle \quad (6)$$

is the signed distance between the MAV and \mathcal{P} , and where

$$\mathbf{n} = \langle \mathbf{e}_1, \mathbf{e}_x \rangle \mathbf{e}_x + \langle \mathbf{e}_1, \mathbf{e}_y \rangle \mathbf{e}_y, \quad (7)$$

is the horizontal component of the plane's normal \mathbf{e}_1 . The vector \mathbf{n} can also be seen as the normal to the line at the intersection of \mathcal{P} and the horizontal plane at the current MAV altitude. Therefore $\boldsymbol{\xi}_{\text{proj}}$ is also the translate of the MAV position along \mathbf{n} , with a magnitude equal to the distance between the plane and the MAV at constant altitude. For a tunnel \mathcal{T} , the reference position is computed using the previous plane case, applied to the sub-plane closest to the MAV current position. This implicit transition between adjacent sub-planes of \mathcal{T} was efficient in practice as is, although small time delays have been observed in some cases, see e.g. the middle plot of Figure 11.

The tele-operation remote control provides at each time instant a 2D desired velocity expressed in tangential basis as

$$\mathbf{v}_{\text{des}}^t = v_{\text{des},2} \mathbf{e}_2 + v_{\text{des},3} \mathbf{e}_3, \quad (8)$$

where each component is saturated and deadzoned such that

$$|v_{\text{des},i}| \in \{0\} \cup [v_{\text{min}}, v_{\text{max}}], \quad i \in \{2, 3\}, \quad v_{\text{max}} > v_{\text{min}} > 0 \quad (9)$$

In addition to the boundary-constrained projection of the MAV position on the surface, the desired velocity $\mathbf{v}_{\text{des}}^t$ given by the tele-operator is filtered to make the MAV remain inside the surface boundaries. The filtered velocity is denoted $\tilde{\mathbf{v}}_{\text{des}}^t$ and is computed as follows. Let E be an outer edge of the reference

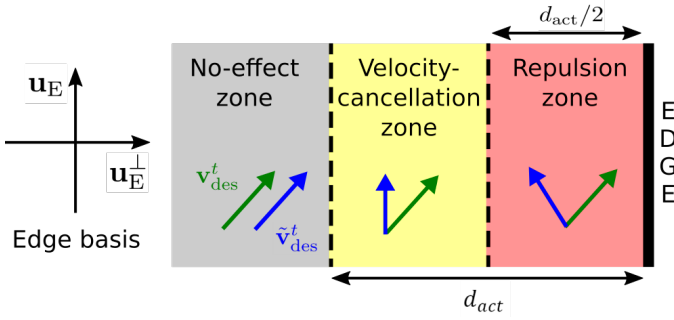


Fig. 2. Velocity filtering close to surface edges. Depending on the MAV distance to an edge, the filter acts on the tele-operator desired velocity component normal to the edge according to (13). Far from the edge, the velocity is unchanged. Each velocity component is canceled if it is outward-oriented. The normal velocity is set inward if too close to the edge.

surface and \mathbf{u}_E^\perp a unit vector normal to E and facing outward. Both \mathbf{v}_{des}^t and $\tilde{\mathbf{v}}_{des}^t$ can then be expressed in an orthonormal basis $(\mathbf{u}_E, \mathbf{u}_E^\perp)$ coplanar with the tangential basis $(\mathbf{e}_2, \mathbf{e}_3)$.

$$\mathbf{v}_{des}^t = v_E \mathbf{u}_E + v_E^\perp \mathbf{u}_E^\perp, \quad (10)$$

$$\tilde{\mathbf{v}}_{des}^t = \tilde{v}_E \mathbf{u}_E + \tilde{v}_E^\perp \mathbf{u}_E^\perp, \quad (11)$$

The distance between the MAV and the edge E is denoted d_E and d_{act} is the filter activation distance. The filtered velocity \mathbf{v}_{des}^t for the edge E is then computed as follows.

$$\tilde{v}_E = v_E, \quad (12)$$

$$\tilde{v}_E^\perp = \begin{cases} v_{rep}(d_E), & \text{if } d_E \in [0, d_{act}/2], \\ 0, & \text{if } d_E \in (d_{act}/2, d_{act}] \text{ and } v_E^\perp > 0, \\ v_E^\perp, & \text{otherwise,} \end{cases} \quad (13)$$

with the edge-repulsion velocity

$$v_{rep}(d_E) = -k_{rep} v_{max} \left(1 - \frac{2d_E}{d_{act}}\right), \quad (14)$$

where $k_{rep} \in (0, 1]$ is a gain controlling the intensity of the repulsive action. Consequently, the filter only acts if the MAV is within the activation distance d_{act} from an outer edge of the reference surface (the top and bottom edges of a cylinder, all four edges of a plane, and all eight outer edges of a tunnel), else the tele-operator desired velocity is left unchanged. If active, then the filter acts on the velocity component normal to the edge in two different ways: either a repulsion effect is applied when the MAV is too close to the edge, or the velocity component toward the edge is cancelled (see Figure 2). The complete filtered velocity is obtained after applying the filter to each outer edges of the reference surface. Since this filtering is applied on each component independently, the MAV can "slide" along the edge of the surface. The 3D MAV reference velocity \mathbf{v}_{des} is then computed by projecting the filtered velocity $\tilde{\mathbf{v}}_{des}^t$ into the inertial frame \mathcal{I} .

The following discrete-time model with sampling period t_s is considered for the MAV translation motion:

$$\begin{aligned} \boldsymbol{\xi}_{k+1} &= \boldsymbol{\xi}_k + t_s \mathbf{v}_k + \frac{t_s^2}{2} \mathbf{u}_k \\ \mathbf{v}_{k+1} &= \mathbf{v}_k + t_s \mathbf{u}_k \end{aligned} \quad (15)$$

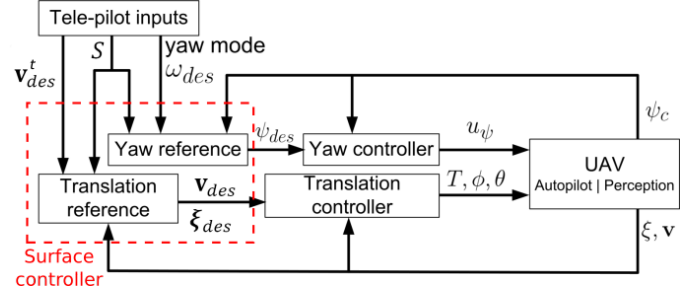


Fig. 3. MAV control loop for surface-constrained tele-operation.

with the standard definition $\mathbf{u} = -\frac{T}{m} R \mathbf{e}_z + g \mathbf{e}_z$ where m is the MAV mass, g the gravity constant and R the rotation matrix from body frame \mathcal{B} to \mathcal{I} . As in [17], a state feedback controller is directly used to track the filtered reference velocity \mathbf{v}_{des} and the surface-referenced position $\boldsymbol{\xi}_{des}$ as

$$\mathbf{u}_k = K_d (\mathbf{v}_{des}(k) - \mathbf{v}_k) + K_p (\boldsymbol{\xi}_{des}(k) - \boldsymbol{\xi}_k) \quad (16)$$

where the gain matrices K_p and K_d can be shaped independently in the horizontal and vertical axes.

Two orientation modes are offered. In **absolute** mode, an arbitrary reference angle ψ_{ini} is initialized at the current MAV yaw ψ_c . In **automatic** mode, it has been designed as follows:

- For \mathcal{C} , it is oriented towards the axis of the cylinder;
- For \mathcal{P} and the lateral planes of \mathcal{T} , it is perpendicular to the tangential basis; For the upper plane of \mathcal{T} , the MAV axis is taken colinear to \mathbf{u}_{axis} .

A yaw rate $\omega_{des} \in [-\omega_{max}, \omega_{max}]$ is also provided by the tele-operator so as to navigate around the yaw reference (for inspection purposes) as

$$\psi_{des}(0) = 0, \quad (17)$$

$$\psi_{des}(k) = \psi_{des}(k-1) + \omega_{des} \cdot t_s \quad (18)$$

and the yaw proportional controller is directly

$$u_{\psi}(k) = k_{\psi} (\psi_{ini} + \psi_{des}(k) - \psi_c(k)) \quad (19)$$

The control input \mathbf{u}_k is then decomposed into roll and pitch angles $\{\phi, \theta\}$ and a thrust T , and given to the low-level autopilot along with the yaw rate $u_{\psi}(k)$.

III. MAV TELE-OPERATION AND NAVIGATION SYSTEM

A DJI Matrice 100 quadrotor (see Figure 4) was used for the flight tests. This platform has a built-in IMU which is processed by a proprietary autopilot for attitude stabilization, and offers a ROS interface [20]. An embedded computer (quadcore Intel NUC7i7DNBE) handles the navigation architecture described in Figure 5. The state of the MAV, i.e. position and velocity in \mathcal{I} as defined in (15), is estimated at 100 Hz by a linear Kalman filter which combines the IMU measurements (including a standard attitude filter) with the 3D position computed by the eVO visual odometry algorithm [21] using the synchronized images acquired at 20 Hz by a stereorig comprising two USB2 cameras (100° fov) separated by a baseline of 27 cm. A laser telemeter also directly measures

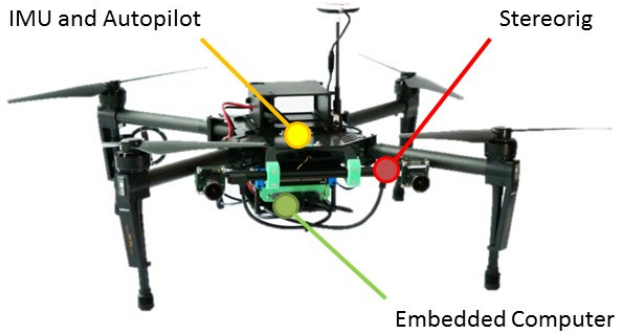


Fig. 4. Micro-Aerial Vehicle used during the experiments.

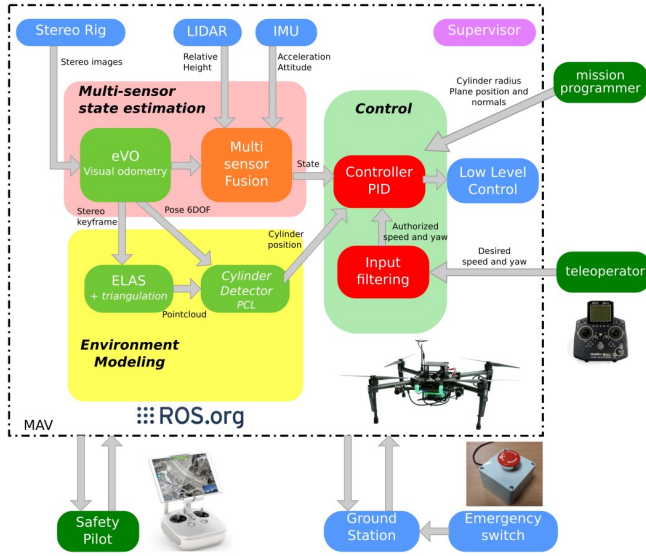


Fig. 5. Overview of the MAV on-board navigation architecture

height during automatic take-off and landing phases. The stereo images are also processed by the ELAS algorithm [22] to compute a depth map. This information is converted into a dense 3D point-cloud expressed in the inertial frame \mathcal{I} thanks to the estimated position and orientation. In the case of a cylinder-shaped inspection surface, a first pipeline exploits the PCL library [23] for fitting and locating a vertical cylinder after ground plane removal. The virtual cylinder is then positioned automatically along the axis of the detected cylinder (Figure 7). This pipeline is activated before take-off and requires a validation by the tele-operator. A second pipeline, active during the entire mission, integrates the point-cloud in an efficient TSDF-based map [24], [25] to provide a 3D reconstruction of the environment, which can be visualized in real-time on the ground station (see e.g. Figure 8).

The tele-operator gives the high-level orders mentioned in Section II to the MAV (tangential and yaw velocities, orientation mode) with a 14-channel remote control connected to the ground station computer via USB. Two stick mappings permits to handle vertical or horizontal surfaces, and one switch allows to toggle between the two orientation modes (Figure 6). A ROS component on the ground station reads the raw serial



Fig. 6. Tele-operation remote control. Left: sticks configuration for cylinder or vertical planes. Right: sticks configuration for tunnel surface.

data from the remote control, which is then formatted into messages and broadcasted to the on-board MAV computer using a Wifi router. A more secure wireless communication scheme could be considered for critical operations.

IV. FIELD RESULTS

Indoor flight tests¹ were carried out in a disused industrial warehouse located in the ONERA center of Meudon, France. The maximum linear velocity was set to 0.5 m s^{-1} and angular velocity to 0.5 rad s^{-1} . The velocity filter (13)-(14) was set to activate within $d_{act} = 20 \text{ cm}$ of a reference surface edge, and with a gain $k_{rep} = 1$, meaning the edge-repulsion magnitude can reach the maximum velocity. During the experiments, the system has been operated by skilled and unskilled pilots and tested several times on test cases corresponding to the three different types of virtual surfaces presented in Section II.

Cylinder test: The first use-case is the inspection of a pillar, a typical mission in railway and energy transportation industries. At the beginning of the mission, the MAV was placed on the ground with its stereo sensor facing the targeted pillar. Despite its square section, it was correctly located by the cylinder detector (described in Section III) and the mission began after the tele-operator had validated the positioning of the virtual surface on the ground station. In the illustrated example, the radius and the height of the cylinder were respectively 4.5 m and 7 m with a 1 m offset above the ground. Figure 7 shows the inspected pillar and an overview of the trajectory. Figure 9 shows the evolution of different performance indicators. The first one is the distance to the virtual surface. The proposed controller maintained the quadrotor during the entire mission in a small neighborhood around the virtual surface (80% of the time under 5 cm; more than 99.5% of the time under 10 cm). The next two indicators show the response of the controller to the velocity inputs of the tele-operator. The tracking of horizontal velocity is good with some small overshoots around stiff input transitions. The observed quality of the vertical velocity tracking was a little lower in this test, with an upward bias which could be

¹A video can be found at <https://tinyurl.com/surfaceteleop>

explained by the particular conditions of this flight test where a weight imbalance was present due to a heavy additional camera payload (which was not embedded during the other flight tests).

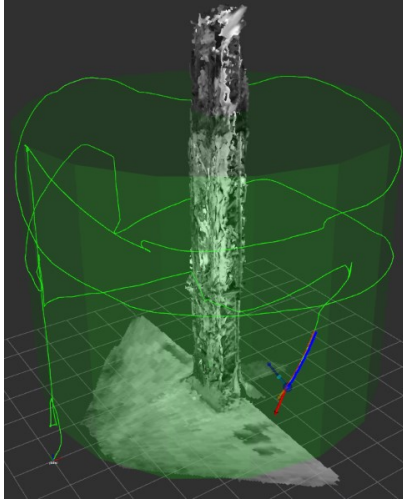


Fig. 7. MAV trajectory on cylinder and 3D model of the inspected pillar.

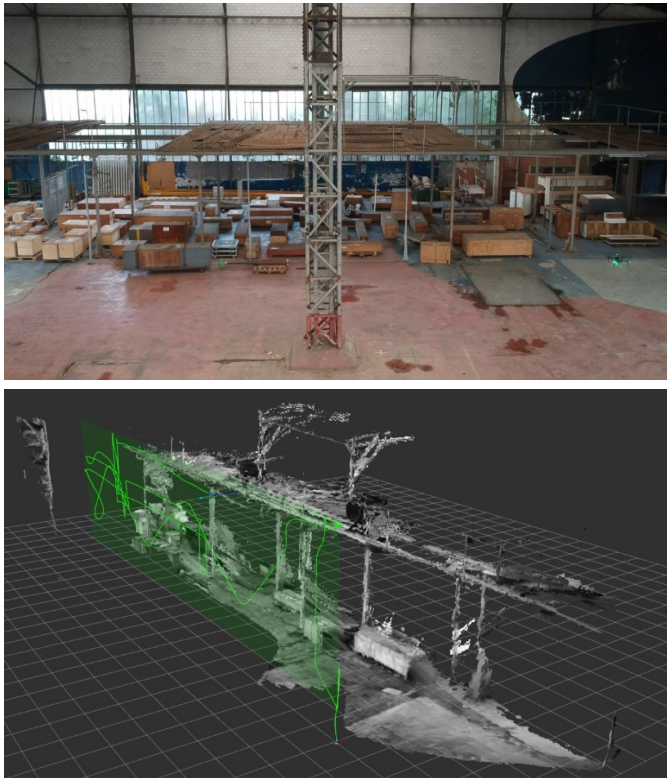


Fig. 8. Tele-operation on a virtual plane in an industrial environment. Top: picture of the environment after take-off. Bottom: MAV trajectory and 3D model of the environment.

Plane test: The second scenario illustrates a standard use-case, the inspection of a planar surface like a wall, a cliff or a facade. Here, the mission consisted in observing a storage

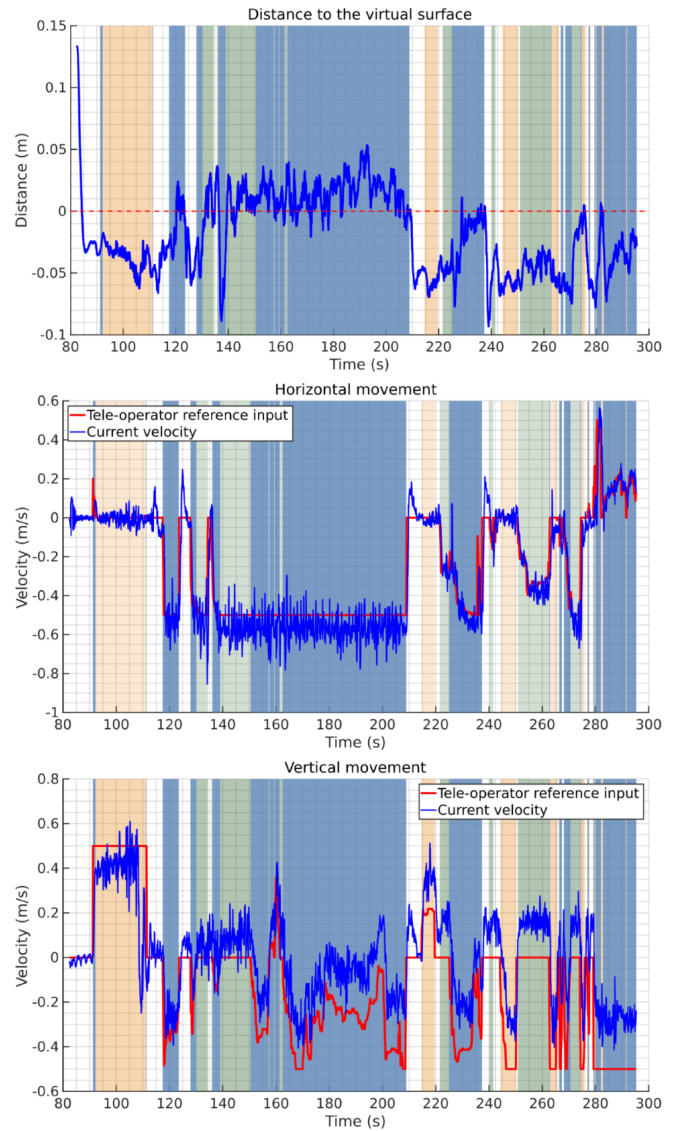


Fig. 9. Performance indicators for *Cylinder* mission. From top to bottom: distance to the virtual surface, horizontal velocity tracking, vertical velocity tracking. Background color indicates if the tele-operator input is purely vertical (yellow), purely horizontal (green) or mixed (blue).

area under a mezzanine. A virtual plane of length 19 m and height 5 m was positioned 1 m above the ground. Figure 8 gives an overview of the environment and the MAV trajectory accomplished during the mission, which shows a correct positioning constrained on the virtual surface. It is confirmed on the first performance curve of Figure 10: the MAV flies at a distance less than or equal to 5 cm during 90% of the mission time (and 99.8% under 10 cm). From the 340th second, the MAV is asked to spin around. Small disturbances between the decoupled yaw and position controllers generate some limited sinusoidal variations in the distance to the virtual surface. The system however successfully tracks the tele-pilot inputs, as shown on the 2nd and the 3rd curves of the same figure. The main exception is around the 105th second when the fencing

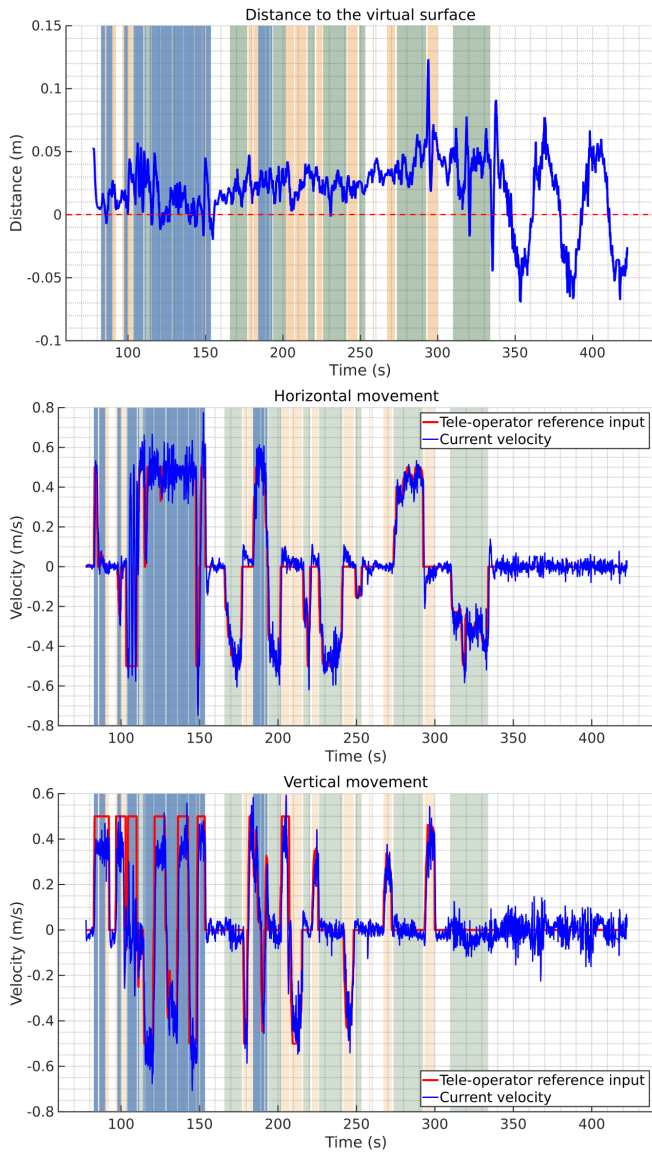


Fig. 10. Performance indicators for *Plane* mission. From top to bottom: distance to the virtual surface, horizontal velocity tracking, vertical velocity tracking. Background color indicates if the tele-operator input is purely vertical (yellow), purely horizontal (green) or mixed (blue).

mechanism prevents the drone to go outside of the virtual surface by the top-right corner, zeroing the velocity inputs provided to the controller. A second activation of the fencing mechanism also occurs as expected at the 205th second.

Tunnel test: The third scenario was inspired by another common mission in railway industries, i.e. the external examination of trains from above and from the sides to detect impacts or defaults. A test was conducted with a structure of length 16.5 m, height 3 m, and trapezoidal base lengths of 8 m (bottom) and 5 m (top). Figure 11 shows the tele-operated trajectory under two different viewpoints and the distance to the surface. The MAV is correctly constrained on the three-planes surface, remaining under a distance of 5 cm during 66% of the mission time (and 88% under 10 cm). Transitions

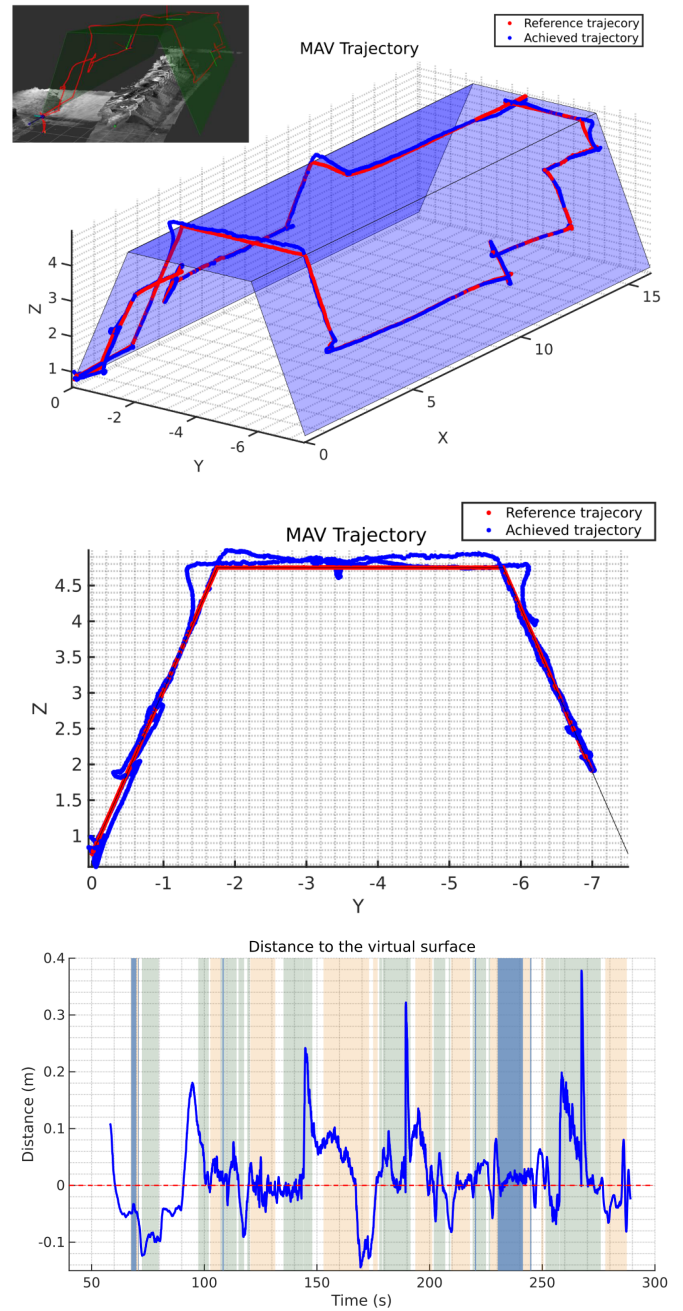


Fig. 11. Performance indicators for *Tunnel* surface. Top and middle: Reference and achieved trajectories during the mission (top left: 3D map of the inspected structure). Bottom: distance to the virtual surface (spikes are due to plane switching generating a reference step).

between the planes at maximal speed created some overshoots (of maximal size 40 cm). This is due to the instantaneous character of the proposed controller (16), which has to react to a sudden change in the reference velocity. This could be improved by considering for example a model predictive control scheme [11], [18] to better anticipate and design a smooth transition between the two velocity directions.

V. CONCLUSIONS

In this paper, a new assisted tele-operation system has been presented which enables a MAV to be controlled on virtual surfaces by an unskilled operator. The motion of the MAV is guaranteed to remain on the pre-defined virtual surface along which the operator can freely guide it using high-level inputs (velocities along the surface and yaw rate). Planes and cylinders are considered as elementary virtual surfaces that can be directly useful by themselves or combined together to address many different types of inspection missions of industrial infrastructures (pillars, facades, vehicles, etc.). The performance of this embedded MAV system has been evaluated in flight tests performed in an industrial warehouse for three typical inspection scenarios. Such a system allows safe and repeatable inspections of industrial infrastructures to be carried out by operators with no specific piloting skills. Future work will include additional flight tests for end-users on industrial sites and with inspection payloads as well as comparison of results (repeatability, accuracy of the acquired data) with respect to flights performed manually by professional telepilots.

ACKNOWLEDGEMENTS

This work has been supported by the research partnership between SNCF Réseau, Altametriz and ONERA.

REFERENCES

- [1] S. Choi and E. Kim, "Building crack inspection using small UAV," in *17th International Conference on Advanced Communication Technology, PyeongChang, South Korea*, 2015.
- [2] B. Chan, H. Guan, H. J. J., and M. Blumenstein, "Towards UAV-based bridge inspection systems: A review and an application perspective," *Structural Monitoring and Maintenance*, vol. 2, no. 3, pp. 283–300, 2015.
- [3] A. Ollero, G. Heredia, A. Franchi, G. Antonell, K. Kondak *et al.*, "The AEROARMS project: Aerial robots with advanced manipulation capabilities for inspection and maintenance," *IEEE Robotics Automation Magazine*, vol. 25, no. 4, pp. 12–23, 2018.
- [4] P. J. Sanchez-Cuevas, P. Ramon-Soria, B. Arrue, A. Ollero, and G. Heredia, "Robotic system for inspection by contact of bridge beams using UAVs," *Sensors*, vol. 19, no. 2, p. 305, 2019.
- [5] J. Quenzel, M. Nieuwenhuisen, and D. e. a. Droschel, "Autonomous MAV-based indoor chimney inspection with 3D laser localization and textured surface reconstruction," *Journal of Intelligent & Robotic Systems*, vol. 93, p. 317–335, 2019.
- [6] M. Korki, N. D. Shankar, R. Naymeshbhai Shah, S. M. Waseem, and S. Hodges, "Automatic fault detection of power lines using unmanned aerial vehicle (UAV)," in *1st International Conference on Unmanned Vehicle Systems-Oman (UVS)*, 2019, pp. 1–6.
- [7] K. Yu, P. Shanthakumar, J. Orevillo, E. Bianchi, M. Hebdon, and P. Tokekar, "View planning and navigation algorithms for autonomous bridge inspection with UAVs," in *Springer Proceedings in Advanced Robotics*, vol. 11, 2020, pp. 201–210.
- [8] S. Omari, P. Gohl, M. Burri, M. Achtelik, and R. Siegwart, "Visual industrial inspection using aerial robots," in *3rd International Conference on Applied Robotics for the Power Industry*, 2014, pp. 1–5.
- [9] M. Stokkeland, K. Klausen, and T. A. Johansen, "Autonomous visual navigation of unmanned aerial vehicle for wind turbine inspection," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2015, pp. 998–1007.
- [10] B. E. Schäfer, D. Picchi, T. Engelhardt, and D. Abel, "Multicopter unmanned aerial vehicle for automated inspection of wind turbines," in *24th Mediterranean Conference on Control and Automation, Athens, Greece*, 2016, pp. 244–249.
- [11] J. Marzat, S. Bertrand, A. Eudes, M. Sanfourche, and J. Moras, "Reactive MPC for autonomous MAV navigation in indoor cluttered environments: Flight experiments," in *20th IFAC World Congress, Toulouse, France*, 2017, pp. 15996–16002.
- [12] A. Eudes, J. Marzat, M. Sanfourche, J. Moras, and S. Bertrand, "Autonomous and safe inspection of an industrial warehouse by a multi-rotor MAV," in *Field and Service Robotics*. Springer, 2018, pp. 221–235.
- [13] P. Stegagno, M. Basile, H. H. Bühlhoff, and A. Franchi, "A semi-autonomous UAV platform for indoor remote operation with visual and haptic feedback," in *IEEE International Conference on Robotics and Automation*, 2014, pp. 3862–3869.
- [14] M. Odelga, P. Stegagno, and H. H. Bühlhoff, "Obstacle detection, tracking and avoidance for a teleoperated UAV," in *International Conference on Robotics and Automation*, 2016, pp. 2984–2990.
- [15] F. J. Perez-Grau, R. Ragel, F. Caballero, A. Viguria, and A. Ollero, "Semi-autonomous teleoperation of UAVs in search and rescue scenarios," in *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2017, pp. 1066–1074.
- [16] F. Bonnin-Pascual, A. Ortiz, E. Garcia-Fidalgo, and J. P. Company-Corcoles, "A reconfigurable framework to turn a MAV into an effective tool for vessel inspection," *Robotics and Computer-Integrated Manufacturing*, vol. 56, pp. 191–211, 2019.
- [17] J. Marzat, A. Eudes, A. Bernard-Brunel, M. Sanfourche, and S. Bertrand, "Collision-free MAV tele-operation along a reference trajectory in indoor cluttered environment," in *IROS Workshop Vision-based Drones: What's Next?, Madrid, Spain*, 2018.
- [18] S. Zhang, D. Wei, M. Q. Huynh, J. X. Quek, X. Ma, and L. Xie, "Model predictive control based dynamic geofence system for unmanned aerial vehicles," in *AIAA Information Systems-AIAA Infotech@ Aerospace, Grapevine, Texas*, 2017.
- [19] E. Hermand, T. W. Nguyen, M. Hosseinzadeh, and E. Garone, "Constrained control of UAVs in geofencing applications," in *26th Mediterranean Conference on Control and Automation, Zadar, Croatia*, 2018, pp. 217–222.
- [20] I. Sa, M. Kamel, R. Khanna, M. Popović, J. Nieto, and R. Siegwart, "Dynamic system identification, and control for a cost-effective and open-source multi-rotor MAV," in *Field and Service Robotics*. Springer, 2018, pp. 605–620.
- [21] M. Sanfourche, V. Vittori, and G. Le Besnerais, "eVO: A realtime embedded stereo odometry for MAV applications," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan*, 2013, pp. 2107–2114.
- [22] A. Geiger, M. Roser, and R. Urtasun, "Efficient large-scale stereo matching," in *Asian conference on computer vision*. Springer, 2010, pp. 25–38.
- [23] 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.
- [24] M. Nießner, M. Zollhöfer, S. Izadi, and M. Stamminger, "Real-time 3D reconstruction at scale using voxel hashing," *ACM Trans. Graph.*, vol. 32, no. 6, Nov. 2013.
- [25] T. Duhautbout, J. Moras, and J. Marzat, "Distributed 3D TSDF manifold mapping for multi-robot systems," in *9th European Conference on Mobile Robots (ECMR), Prague, Czech Republic*, 2019.