Visual Aids for Safe Operation of Remotely Piloted Vehicles in the Controlled Air Space

Fabrizio Giulietti,[1] Lorenzo Pollini,[2] and Giulio Avanzini[3]

Abstract

The paper presents a set of visual aids for enhancing remote pilot perception of potential violations of allowed fly areas or conflicts with conventional air traffic when operating remotely piloted aerial vehicles. Assuming a video stream from an on-board camera is available to the remote pilot, visual aids are provided in a head-up display modality by means of reality augmentation techniques. The main visual element consists of a dynamic set of fences allowing for a safe aircraft separation away from no-fly zones and from neighboring vehicles. The shape of the fences is varied according to aircraft current speed and altitude, in order to allow for a sufficient safety margin, also in case of a loss-of-control accident. As a further visual aid, the predicted future position of the aircraft is determined and fence color is changed in order to highlight potential violations of the allowed operational area. The proposed methodology is validated by means of simulations in a reference operational scenario. Results demonstrate the effectiveness of the proposed approach in improving pilot awareness.

Keywords

Safe Operation, Visual Aid, Remotely Piloted Vehicles, Augmented/Virtual Reality

[1] Department of Industrial Engineering (DIN), Università di Bologna, Forlì Italy

[2] Department of Information Engineering, Università di Pisa, Pisa, Italy

[3] Department of Engineering for Innovation (DII), Università del Salento, Lecce, Italy

Corresponding author:

Fabrizio Giulietti, Università di Bologna, Department of Industrial Engineering (DIN), Via Fontanelle 40, 47121 Forlì, Italy.
Email: fabrizio.giulietti@unibo.it

Introduction

The use of remotely piloted vehicles is becoming a reality in many civil applications, in particular for remote sensing, environmental monitoring, and natural disasters' prevention and relief tasks.[1–3] This fact poses several challenges for the integration of this class of aircraft within conventional air traffic. Civil aviation authorities are introducing stringent requirements and limits for the use of unmanned aircraft in order to prevent the risk of possible incidents or even accidents with other flying vehicles, violations of protected areas and risks for the population. In this respect, air rules dedicated to Remotely Piloted Vehicles (RPV's) started appearing at the end of the first decade of the 21st century.[4;5]

The main aim of this paper is to propose, analyse, and test in a high-fidelity simulation scenario, a set of visual aids designed to enhance remote pilot's perception of constraints on admissible flight conditions and operation boundaries. The applicative scenario is primarily focused on small- and medium-size vertical takeoff and landing (VTOL), possibly multi-rotor, platforms. The objective is to create a set of augmented reality tools providing the pilot with:

1. a strong and intuitive perception of current position of the remotely operated aircraft with respect to the limits of the area where the aircraft can be safely operated;
2. an alarm for possible violations of no-fly zones, within a prescribed time horizon, based on an estimate of expected trajectories.

A remote pilot can fly an Unmanned Aerial Vehicle (UAV) as a conventional aircraft under instrumental flight rules (IFR), basing the control action on telemetry information on speed, altitude, and other relevant flight data, plus a map, all reported on a glass cockpit. Quite obviously, when the glass cockpit also features a video stream from the on-board camera, piloting the UAV can be based also, if not primarily, on visual information. In both cases, latency times for data transmission from and to the vehicle need to be taken properly into account for avoiding annoying flying quality degradation. If on one side, the use of video from the on-board camera allows the pilot to actually see the environment of flight, at the same time the perception of the surroundings is different from that of a conventional pilot, due to several factors (limited field of view of the camera, absence of motion cues that complement the visual information, etc.). Moreover, in many civil applications, especially those related to disaster relief and aerial monitoring, unmanned aerial vehicles are expected to operate over populated areas, possibly in the presence of other manned or unmanned aircraft. This makes the remote pilot task extremely difficult. In this respect, it is necessary to enhance pilot's perception of the flight environment, by providing the pilot visual clues that improve his/her ability to maneuver the vehicle avoiding violation of no-fly areas or potential conflict with neighboring vehicles. This visual clues are expected to allow the pilot to focus on mission task, without the need for constantly monitoring the position of the vehicle on a map, but at the same time the remote pilot should be warned against potential violations of operational boundaries and risk of collisions with other aircraft in an efficient and intuitive way.

The use of a tunnel-in-the-sky proved to be an extremely effective means for providing the pilot with a visual feedback that greatly enhances the capabilities of precise tracking of a prescribed trajectory.[6] Techniques for optimizing the trajectory towards the prescribed flight path when large deviations occur were also discussed.[7] The use of tunnel-in-the-sky display was thus proposed for head-up displays for both conventional and remote pilots of fixed wing aircraft. In this framework, VTOL vehicles pose several challenges, especially provided that their hovering and low-speed capabilities is the distinctive feature that makes them useful in many scenarios, such as aerial monitoring during disaster relief operations. When the aircraft is expected to freely fly over a given area for monitoring purposes at low or zero airspeed it is not even possible to provide a reference trajectory, and the tunnel becomes no longer useful.

Rather then providing a prescribed trajectory, the visual aids developed in the present work try to provide a cue for the admissible bounds of aircraft motion within the area of operations. This is a particularly relevant feature for rotary-wings aircraft, that can fly in highly constrained areas. The bounds are changed dynamically as a function of flight condition (airspeed and altitude), in order to account for possible loss-of-control accidents that may result into the vehicle freely falling on the ground. This requires accounting for a safety distance from the no-fly zone that increases with vehicle current altitude and speed component in the direction perpendicular to the boundary of the no-fly zone. The bounds are reported to the pilot in terms of a virtual transparent fence, that is projected over the current mission scenario, as obtained from the on board camera. The bounds can also be reported on the map, when present on the glass cockpit of the UAV Ground Control Station (GCS). As a further contribution, a trajectory prediction is performed,[8;9] where future position of the UAV is evaluated as a function of current flight condition and maneuver state (that is, velocity and acceleration). This allows for evaluating potential conflicts with other aircraft of known position and/or violations of the bounds that surround the no-fly areas within a prescribed time interval. When the predictor indicates that a conflict is possible, the color of the fence is changed from yellow to green, than orange and finally red, depending on how imminent the violation is expected to be and how aggressive the maneuver required to avoid it.

The rest of the paper is structured as follows. In the next Section the models used for describing vehicle motion are briefly recalled. Details on the implementation of the different types of visual clues and system elements (fence and trajectory predictor) are reported in Section 3. In Section 4 the practical implementation of these elements is described, together with the results of a few relevant simulations. A section of concluding remarks ends the paper.

Mathematical Modeling

Reference frames

Three sets of mutually orthogonal reference axes are introduced for describing aircraft motion:

- an Earth Fixed Frame, $\mathbb{F}_E$;
- a Local Vertical Local Horizontal (LVLH) Frame, $\mathbb{F}_V$;
- a Body Frame, $\mathbb{F}_B$.

The origin of the Earth Fixed Frame lies in a point of the Earth surface relevant to the mission scenario; the z_E-axis lies along the local vertical, in the direction of gravity acceleration, **g**, in the origin; the x_E-axis direction is arbitrary, and the y_E-axis completes a right-handed triad. In the absence of requirements related to the particular mission segment considered, the x_E axis can be chosen pointing towards the North and consequently the y_E axis is directed Eastwards. In this case the Earth-fixed frame is also termed as the North-East-Down (or simply NED) frame. For most operations with an Unmanned Aerial System (UAS), centrifugal and Coriolis acceleration terms related to Earth rotation around the polar axis and Earth's surface curvature are negligible and can be dropped in the expression of the equations of motion, that is, the assumption of flat not-rotating Earth applies, under which $\mathbb{F}_E$ can be considered as an acceptable approximation of an inertially fixed reference frame.

In the framework of the flat, non-rotating Earth model, the position of the vehicle is expressed in terms of displacement along the NED axes, that is

$$\mathbf{r}_E = (x_E, y_E, -h)^T \tag{1}$$

where h is aircraft altitude. The velocity vector, defined by North, East, and Down velocity component, is thus given by

$$\mathbf{V}_E = (v_N, v_E, v_D)^T = (\dot{x}_E, \dot{y}_E, -\dot{h})^T \tag{2}$$

The subscript E indicates that vectors **r**, and **V** are referred to the Earth-fixed frame. In what follows, the subscript will be dropped for the sake of simplicity.

The axes of the LVLH Frame have the same direction of the axes chosen for the Earth-fixed frame, but its origin lies in the center of gravity of the vehicle. Consequently, $\mathbb{F}_V$ translates with the aircraft while keeping a fixed orientation with respect to $\mathbb{F}_E$. Finally, $\mathbb{F}_B$ is attached to the aircraft, assumed as a rigid body. Its origin is the center of gravity of the vehicle, the x_B and the z_B-axes lie in the symmetry plane of the aircraft (when present), where x_B points in the forward direction and z_B is directed downwards when the vehicles flies horizontally. The y_B-axis completes a right-handed triad and it is thus perpendicular to the symmetry plane. For multirotor systems characterized by a polar symmetry, the z_B is directed along the symmetry axis, pointing again downwards during horizontal flight, whereas the choice of x_B and y_B axes becomes almost arbitrary, on the plane perpendicular to z_B. As an example, for quadrotor configurations, it is possible to choose the x_B axis along a line parallel to one of the rotor mount structure (+ configuration) or along a line parallel to the bisector of the angle between two adjacent mounts ($\times$ configuration). Data acquired from gyros and accelerometers are referred to the body frame. A subscript B indicates that a vector quantity is represented in terms of components in the body-fixed frame.

Vehicle models

Three models, with increasing level of complexity, will be considered in the present work. At the lowest end, a simple 3 degrees of freedom (DoF), point-mass ballistic model is used for estimating the maximum distance travelled by the vehicle after a loss-of-control (LOC) accident. This is used for the definition of a safety margin from

no-fly areas. A navigation 3 DoF navigation model is then used for the implementation of a trajectory predictor, for estimating potential violations of the no-fly zones. Finally a refined high-fidelity, fully nonlinear 6 DoF model of a quadrotor vehicle is implemented, to be used for a realistic simulation of aircraft operations.

Ballistic model The ballistic model is given by the following set of differential equations:

$$\begin{bmatrix} \dot{\mathbf{r}} \\ \dot{\mathbf{V}} \end{bmatrix} = \begin{bmatrix} \mathbf{V} \\ \mathbf{g} \end{bmatrix} \tag{3}$$

where one assumes that only gravity $\mathbf{g}$ acts on the vehicle, as thrust drops to zero after the failure is recognized by the pilot and drag is neglected. The ballistic model is thus extremely simple and only the initial speed V_0 is required for analytically estimating the distance travelled in free fall. Details for the assumptions listed above are discussed in the next section, together with a method for determining a reasonable value for the initial speed V_0 that results in a worst-case scenario for the maximum distance travelled in free fall after LOC. This estimate is then used for dynamically generating a fence that provides bounds to vehicle motion, sufficiently far away for avoiding violations of no-fly areas or interference with conventional air traffic that takes place out of the flight volume where UAV operations are allowed. Model complexity is kept to a minimal level for two reasons: (i) computational burden is modest and (ii) it is possible to generate the visual aid with the same procedure for different vehicles, without the need for tuning the algorithm.

Navigation equations The navigation equations are a set of differential equations that provide position and velocity of the vehicle with respect to the NED reference system by using inertial and GNSS sensor measurements. These equations are obtained from the kinematic relation, Eq. (2), and Newton's second law, achieving a very simple form in $\mathbb{F}_E$, under the flat Earth approximation:

$$\begin{bmatrix} \dot{\mathbf{r}} \\ \dot{\mathbf{V}} \end{bmatrix} = \begin{bmatrix} \mathbf{V} \\ \mathbb{T}_{EB}\,\mathbf{a}_B + \mathbf{g} \end{bmatrix} \tag{4}$$

where $\mathbf{g} = \begin{bmatrix} 0 & 0 & g \end{bmatrix}^T$ is the gravity acceleration, $\mathbb{T}_{EB}$ is the transformation matrix between the Earth-fixed and body-fixed frame, that depends on vehicle attitude, $\mathbf{a}_B$ is the acceleration vector as measured by accelometers in the body-axis frame. Since the effect of external forces enters the model through measured body axis accelerations, the navigation model can be applied to any vehicle. For the purposes of the present work, a multi-rotor aerial platform is considered and then one has

$$\mathbf{a}_B \approx \frac{\mathbf{T}}{m} - \mathbb{T}_{EB}^{-1}\mathbf{g} \tag{5}$$

where $\mathbf{T} = \begin{bmatrix} 0 & 0 & -T \end{bmatrix}^T$ is the total delivered thrust and m is the vehicle mass and other aerodynamics forces are assumed small and, thus, neglected. These equations are used for estimating future vehicle position on the basis of data available at a given time, as obtained from the GNSS sensors. This allows for providing an early warning of a possible violation of the bounds of the admissible area of operations.

High-fidelity 6 DoF model In order to evaluate the validity of the approach in a realistic, although virtual, simulation environment, a high-fidelity simulation model is implemented for a quad-rotor vehicle. This model features the complete set of 6 DoF rigid-body equations of motion.[10] Under the assumption of incompressible flow, the rotor aerodynamics is described by Momentum Theory (MT) and Blade Element Theory (BET).[11–13]

Visual Aids for Remote Pilots

As briefly outlined in the introduction, two types of tools are envisaged with the objective of enhancing pilot awareness: a dynamic virtual fence and a boundary violation alarm. The dynamic virtual fence is implemented by graphically representing the limits of operations of the UAV by means of a transparent wall that follows a curved shape, up to the operational ceiling prescribed by current regulations on the use of UAV's. The shape of the fence is determined as a function of the boundary of the no-fly areas, including corridors where other aircraft are expected to operate. The actual shape of the fence is then dynamically modified, depending on the current flight condition of the vehicle, primarily velocity and altitude. Provided that, after a loss-of-control accident, the UAV freely falls along an approximately parabolic trajectory, the fence is shaped in such a way that it is curved towards the vehicle's position at higher altitudes. Moreover, its position moves farther away from the actual limits of the no-fly area, depending on vehicle's airspeed, thus adding a safety margin for the loss-of-control accidents. When the velocity of the UAV decreases, the fence gets closer to the actual boundary, thus allowing the vehicle to operate close to the boundary only at low speed or in hover, when a minimum distance is sufficient for enforcing a safety margin.

The fence is represented by means of a yellow grid, when violation of the boundary is not expected within a prescribed time interval. The yellow color was selected because it was found to stand out well over the background scenery, especially in those cases when there are trees or fields in the background. A trajectory predictor is implemented, which can estimate the "future position" of the vehicle, given its current flight condition and maneuver state, that is, velocity and acceleration vectors. When a violation is expected, the color of the grid turns from yellow to green, indicating a safe situation requesting pilots attention. Given the smaller distance, a green fence is well visible even on a green background. The fence color is then changed smoothly from green to orange and finally to red as a function of the distance from the fence and vehicle maneuvering capabilities required to move away from he boundary. When the vehicle exits the risk region, the fence color becomes yellow again. Quite obviously, other coloring schemes are possible, depending on the environment.

Shape and Position of the Safety Fence

As outlined above, position and shaper of the fence on the map and in the visual scenario are required to account for a safety margin, that should prevent the vehicle from entering the no-fly-zone even during a major accident with total loss-of-control, that would result in a non-predictable trajectory. In this framework, it is assumed that the UAV is equipped with a system that powers off the vehicle when a loss-of-control
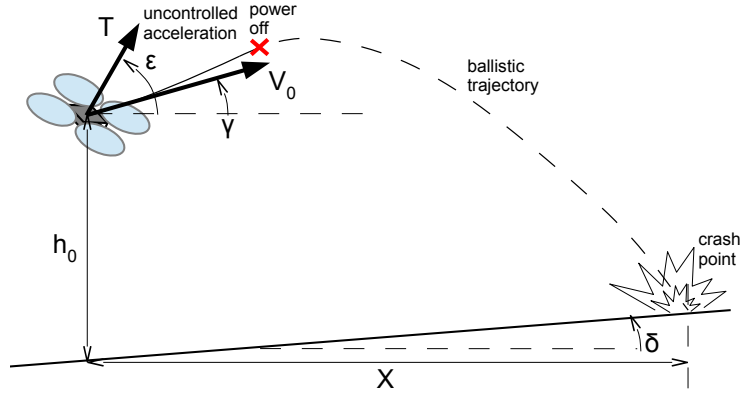
Figure 1. Sketch of multirotor UAV crashing into the ground.

accident occurs. This can be done by means of a remote pilot command or an on-board failure detection system. In both cases a latency time τ needs to be accounted for, before the engines are powered off after the failure at time t_F. In the worst case scenario, the propeller of the UAV provides maximum thrust T in a constant direction during the latency time, with an inclination $\hat{\phi}$ with respect to the local vertical. Note that this is an unlikely situation to occur in practice, as long as in a loss-of-control scenario the UAV is expected to point the thrust vector in a (possibly rapidly changing) time-varying direction, which results in a more modest velocity increment than that obtained assuming a constant direction for thrust. The situation is depicted in Fig. 1, where h_0 and V_0 are the initial altitude and velocity, respectively, γ the climb angle, δ the slope of the terrain, and $\varepsilon = \pi/2 - \hat{\phi}$ is the angle between the thrust vector and the horizontal plane, where the complementary angle, $\hat{\phi}$, provides the angular displacement of the z_B axis from the local vertical, that depends on pitch and roll angles only, with $\cos\hat{\phi} = \cos\theta\cos\phi$. After the latency time, the engines are switched off, and the vehicle approximately follows a ballistic trajectory under the action of gravity acceleration g.

In order to keep the number of problem parameters down to a minimum level, the UAV, with an overall mass m, is assumed to be initially in level flight over a horizontal terrain. Clearly, the worst case scenario occurs when the thrust vector is pointed in the vertical plane that contains the velocity vector. During the time interval $t \in [t_F, t_F + \tau]$ vehicle motion can be approximately described by the following velocity and position vectors, $\mathbf{V}$ and $\mathbf{r}$, projected in a fixed frame where the vertical (h) and horizontal (x) axes identify the plane of the motion:

$$\mathbf{V} = (\dot{x}, \dot{h})^T = (V_0 + (T/m)\cos\varepsilon(t - t_F),\ [(T/m)\sin\varepsilon - g](t - t_F))^T \quad (6)$$

$$\mathbf{r} = (x, h)^T = (V_0(t - t_F) + (T/m)/2\cos\varepsilon(t - t_F)^2,$$
$$h_0 + [(T/m)\sin\varepsilon - g](t - t_F)^2/2)^T \quad (7)$$

The ballistic solution for $t > t_F + \tau$ is given by

$$\mathbf{V} = (V_B,\ \dot{h}_B - g\Delta t)^T \quad (8)$$

$$\mathbf{r} = (x_B + V_B\Delta t,\ h_B + \dot{h}_B\Delta t - g\Delta t^2/2)^T \quad (9)$$

with $\Delta t = t - (t_F + \tau)$. The initial conditions for the ballistic arc, x_B, h_B, V_B, and $\dot{h}_B$, are obtained from Eqs. (6) and (7) evaluated at time $t = t_F + \tau$, that is

$$V_B = V_0 + (T/m)\tau \cos \varepsilon$$
$$\dot{h}_B = [(T/m)\sin \varepsilon - g]\,\tau$$
$$x_B = V_0\tau + (T/m)(\tau^2/2)\cos \varepsilon$$
$$h_B = h_0 + [(T/m)\sin \varepsilon - g]\tau^2/2$$

The time interval between the end of the latency time and vehicle crashing on the ground at $t_C = t_F + \tau + \Delta t_C$ is thus obtained by letting $h(t_C) = 0$ in Eq. (9). One thus obtains:

$$\Delta t_C/\tau = \alpha \sin \varepsilon - 1 + \sqrt{\beta + \alpha \sin \varepsilon\,(\alpha \sin \varepsilon - 1)}$$

where $\alpha = T_{\max}/(mg)$ and $\beta = 2h_0/(g\tau^2)$ are two nondimensional parameters related to maximum thrust and initial altitude, respectively. The overall distance X covered by the vehicle after failure thus satisfies the equation

$$X/(g\tau^2) = \xi(\varepsilon; \alpha, \beta, \sigma) \tag{10}$$
$$= \sigma + (\alpha/2)\cos \varepsilon + (\sigma + \alpha \cos \varepsilon)\left[\alpha \sin \varepsilon - 1 + \sqrt{\beta + \alpha \sin \varepsilon(\alpha \sin \varepsilon - 1)}\right]$$

where $\sigma = V_0/(g\tau)$ is the non-dimensional parameter related to the initial airspeed.

For $\sigma = 0$, an initial hovering condition is assumed. For $\sigma_{\max} = V_{\max}/(g\tau)$, the maximum speed in forward flight is considered. When a latency time in a range between 1 to 2 s is assumed and a maximum speed between 20 and 40 m/s is expected for remotely operated multirotor platforms, σ varies in a range between approximately 1 and 4. If an operational ceiling for RPV of 150 m is considered, β varies between 0 and $\beta_{\max}$, with $\beta_{\max}$ between approximately 7 (for $\tau = 2$ s) and 30 (for $\tau = 1$ s). Finally, a value of $\alpha = 2$ is assumed as reasonable for multirotor platforms.

Provided that the only unknown in the problem is the inclination of the thrust angle over the horizontal plane, ε, it is possible to determine the worst case scenario, $X_{\max}$, for different values of β and σ. The maximum distance covered after failure is thus given by:

$$X_{\max} = g\tau^2 \max_{\varepsilon} \left[\xi(\varepsilon; \alpha, \beta, \sigma)\right] \tag{11}$$

To the authors' knowledge, an analytical solution for the equation $\mathrm{d}\xi/\mathrm{d}\varepsilon = 0$ is not available. As a consequence, $X_{\max}$ and the corresponding tilt angle $\varepsilon_{X_{\max}}$, are determined by numerically evaluating the maxima for the function $\xi = \xi(\varepsilon)$ in Eq. (10), assuming prescribed values for α, that depends on vehicle characteristics, β and σ, that depend on the operating condition at failure time t_F. Figure 2 shows the contour plot of the nondimensional distance parameter $\xi = X/(g\tau^2)$ as a function of β and σ for $\alpha = 2$. The values of the pitch angle $\theta = \varepsilon - 90$ deg that results into the farthest distance traveled after failure are reported in Fig. 3 as a function of the altitude parameter β for $\sigma = 0$ (continuos line), for an initial hovering condition, and $\sigma = \sigma_{\max} = 4$ (dotted line), for an initial airspeed equal to the maximum one.
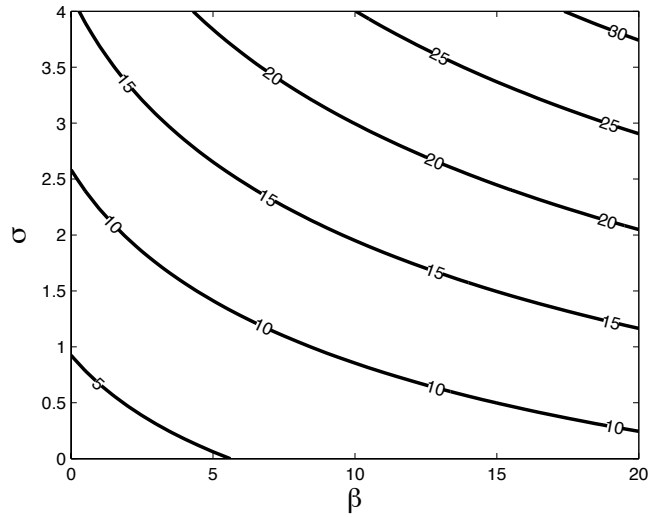
Figure 2. Contour plot of distance parameter $X_{\max}/(g\tau^2)$ as a function of altitude and speed parameters.
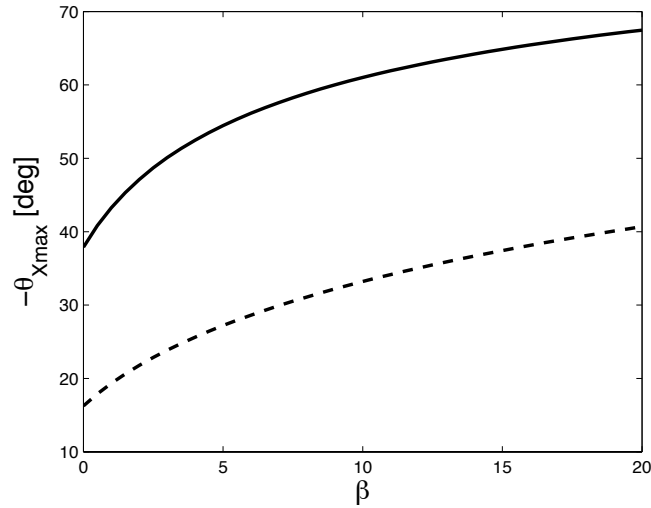

Figure 3. Worst-case acceleration pitch angle $\theta_{X_{\max}}$ for $\sigma = 0$ (—) and $\sigma = \sigma_{\max}$ (- - -).

Note that the result for the hover condition ($\sigma = 0$) at very low altitude (i.e. $h_0 \approx 0$, that results into a small value of β) indicates that the worst-case pitch angle is $\varepsilon_{X_{\max}} \approx 40\,\mathrm{deg}$. This is close to the well know condition for maximum range of a projectile in ballistic flight. The result is not identical since in the present problem an initial acceleration phase is also present. When the altitude of the vehicle increases, $\varepsilon_{X_{\max}}$ decreases. This indicates that the worst case is obtained for higher values of the horizontal acceleration component. This is no longer true when a high horizontal

airspeed is already present, and longer free–fall trajectories are obtained for smaller pitch angles, that is, higher initial altitude gain.

Remark 1 : The assumption of a ballistic trajectory implies that aerodynamic drag is neglected during the approximate evaluation of the distance travelled after the failure. When there is no wind, drag decreases the actual distance covered in free fall, making the estimate even more conservative. Conversely, in the presence of a strong wind, the trajectory can be affected significantly, especially for lightweight vehicles. The distance covered becomes shorter in the presence of a headwind, or longer, when a tailwind is present. The first case is of no interest, as the trajectory would be shorter and the ballistic estimate is, once again, conservative. To undestand what happens in the tailwind case, aerodynamic effects should be taken into consideration. Drag force F_d is proportional to the square of the difference between the inertial speed V and wind speed, V_W, that is $F_d \propto (V - V_W)^2 = \Lambda \cdot (V - V_W)^2$. The proposed worst case analysis neglects wind and assumes that the vehicle accelerates uncontrolled to reach the worst possible velocity V_B, thus the amount of favorable air drag neglected in the ballistic analyisis (with no wind) is $\Lambda \cdot (V_B)^2$. This let us conclude that, unless $V_W > V_B$, the proposed ballistic analysis still produces a conservative landing distance estimate. As a matter of fact, neglecting drag is equivalent to assuming a wind speed which is as fast as the vehicle itself at initial time (when $V_W = V_B$ the drag force nulls). Such an operational condition is considered unlikely to occur in practice, provided UAS are flown when wind speed is a (possibly small) fraction of the maximum vehicle airspeed.

Remark 2 : The type of fault envisioned in this simple analysis of the maximum traveled distance prior to impact, that is, a sudden lateral acceleration, is not unlikely to occur in reality. As a matter of fact, this is exactly what happens when an uncontrolled fault to a single motor or rotor (e.g. after an impact of the rotor with an obstacle) occurs to a multi-rotor with 4 propellers. As an example, consider a quad-rotor vehicle in "plus" $(+)$ configuration (2 motors placed fore and aft, and 2 to the left and to the right). Due to an electronic motor controller failure, propeller breakage or other incidents, one of the motors stops producing thrust. Suppose, without loss of generality, that the faulty motor is the right one. As a result of loss of thrust on the right side, the vehicle suddenly banks to the right due to generation of an uncontrolled (and uncontrollable) roll torque. When this happens, the flight control system is likely to idle the left motor to suppress additional unwanted roll torque. At this point the flight control system has lost the capability to control the roll axis and the vehicle is likely banked by a certain roll angle and it also starts spinning around the yaw axis due to loss of yaw torque balance. As a side product of having lost roll control, the vehicle starts losing altitude and the control system will likely increase the two remaining motors rotational rates to the maximum allowed thrust in an effort to climb back to the prescribed altitude. Unfortunately now the vehicle is banked and the vehicle acceleration is not directed up but sideways, because of the current bank angle. Something similar would happen in the longitudinal direction as a consequence of loosing either the forward or rear rotor.

Remark 3 : Another very common fault that might produce an uncontrolled acceleration is a fault of the altitude estimation system. An altitude sensor that produces an erroneous sensor reading (e.g. an altitude much lower then the current one) for a significant duration might trigger an instantaneous climb of the vehicle. If the

vehicle control system does not handle properly this event, it might also happen that the propellers reach a thrust close the maximum one reducing significantly the attitude control authority, especially in the presence of crosswinds. As a result, the vehicle acceleration might not be oriented along the local vertical. Similarly, wrong readings from the Global Positioning System (GPS) may result in uncontrolled accelerations, provided that especially small-scale UAV's rely on GPS as a primary means for navigation. Many cases were reported of unmanned vehicles of all sizes instantaneously deviating from their programmed trajectory due to erroneous GPS fixes, where a GPS blunder may produce errors as large as hundreds of meters. In this respect, larger errors are less dangerous as they are easier to detect and filter out. If such an error is not appropriately detected, the net result would probably be an instantaneous acceleration directed in a practically random direction. The only way to recover from this situation is to go back to manual control (no GPS feedback), re-gain immediately spatial awareness (manual control requires the pilot knowledge of, at least, vehicle heading) and bring back the vehicle from its current position to a safe one. This requires a highly trained and skilled pilot. A less experienced pilot is more likely not to recover spatial awareness in time, command the vehicle toward the wrong direction and finally drop the throttle to stop the vehicle from flying away or into nearby obstacles.

A sensor integrity monitoring system is advisable on every closed loop system, and such system should be able to correctly handle some types of fault and, at least, mitigate their adverse effects. Unfortunately the large and fast proliferation of small drones piloted for leisure by people unaware of the risks of flying even the smallest heavier-than-air vehicle, together with an enduring lack of a comprehensive set of regulations and authorities to enforce them, are paving the way for a possibly dramatic increase in the occurrence of accidents like those described above.

Early Warning for Fence Violation

As an additional visual tool a prediction algorithm is implemented, which allows the operator to avoid fence violation by indicating when current course and/or maneuver state are likely to cause the vehicle to trespass the no-fly area or violate the safety region surrounding it. The prediction algorithm numerically integrates navigation equations (4) to generate a prediction of the future position of the aircraft, using data derived from inertial and GNSS sensor measurements. The trajectory predictor is based on a second-order Runge-Kutta method (also known as midpoint method). Letting $\mathbf{y} = (\mathbf{r}^T, \mathbf{V}^T)^T$ be the state vector of the navigation equations, position and velocity are propagated forward in time as follows (see Innocenti *et al.*[8] for more details on the predictor):

$$
\begin{aligned}
\mathbf{k}_1 &= f(\mathbf{y}_t)\,\Delta t \\
\mathbf{k}_2 &= f(\mathbf{y}_t + \mathbf{k}_1/2) \\
\mathbf{y}_{t+\Delta t} &= \mathbf{y}_t + \mathbf{k}_2
\end{aligned}
\tag{12}
$$

where $\mathbf{y}_t$ is aircraft state at the current time t and $f(\cdot)$ represents Eq. (4) in compact form. The prediction interval, Δt, is a design parameter and it should take into account both pilot ability and aircraft maneuvering performance limits. For the quadrotor considered in the simulations described in the next section, a prediction horizon $\Delta t = 3$ s was used.

When the predicted position indicates that the aircraft is expected to remain safely within the fence limits within the prescribed prediction horizon Δt, the fence is enlighten by a bright yellow color. As outlined above, the fence color is varied when the expected trajectory violates the safe limits prescribed. Fence trespassing is predicted when the condition

$$\mathbf{n}_f \cdot \mathbf{d}_{\Delta t} > 0 \tag{13}$$

is violated, where $\mathbf{n}_f$ is the normal to the plane tangent to the fence at the current altitude, directed from the fence into the volume of admissible flight, and $\mathbf{d}_{\Delta t} = \mathbf{r}_{\Delta t} - \mathbf{r}_F$ is the vector that connects the vehicle predicted position P, corresponding to the position vector $\mathbf{r}_{\Delta t}$, to its projection Q on the fence, indicated by the position vector $\mathbf{r}_F$. The normal $\mathbf{n}_f$ to the fence in Q is easily obtained from the coordinates of three out of the four points of the grid element of the fence that surround the point Q. Letting Q_i, $i = 1, 2, 3, 4$ be these points, identified by the position vectors $\mathbf{r}_{Q_i}$, the normal $\mathbf{n}_f$ is parallel to the vector $\mathbf{q}_{12} \times \mathbf{q}_{13}$, where $\mathbf{q}_{ij} = \mathbf{r}_{Q_j} - \mathbf{r}_{Q_i}$, $i \neq j$. At the same time, $d_{\Delta t} = \|\mathbf{d}_{\Delta t}\|$ represents the minimum distance between the predicted position and the fence.

The first warning, represented by the fence color turning to green, appears when the predicted trajectory of the aircraft intersects the fence and a corrective maneuver within a limit of 75% of maximum aircraft maneuvering capability is sufficient not to violate the fence. Such a maneuver is obtained by increasing the current value of the measured acceleration with increments that represent a pull-up/pull-down and a roll maneuver.[9] If maneuvering performance limits are prescribed in terms of maximum load factor, $n_{z_{\max}}$, the trajectory to avoid violation will be characterized by a bank angle ϕ such that $\cos \phi = 1/(0.75 n_{z_{\max}})$. It is easy at this point to correct the acceleration by the maximum increment, evaluate a future position in the presence of the escape maneuver and check for the fence violation condition.

As the vehicle gets closer to the fence, the color fades to orange, when the violation is predicted in less than $\Delta t/2$ seconds or the current flight condition of the aircraft requires a trajectory correction employing 90% of the aircraft maneuver capability. Finally the fence turns red when the avoidance maneuver would require 100% or more of the aircraft capability, thus indicating the inability to avoid a fence violation within the prescribed prediction horizon. The fence remains red until the pilot successfully moves the vehicle out of the no-fly zone and recovers an adequate safety margin with respect to it.

Implementation of Visual Clues and Testing in High Fidelity Simulation Environment

The simulation environment

The simulation environment adopted for testing the visual aids is derived from that described in Pollini &Metrangolo [10]; Di Corato et al. [14],[15], and consists of several hardware and software components: a vehicle dynamics simulator, the hardware of an actual autopilot, a virtual environment, a ground station, and a Commercial-Off-The-Shelf (COTS) remote control radio. The vehicle dynamics simulator runs in pseudo-real time on a desktop PC, and simulates the vehicle dynamics and external

disturbances, and emulates all the sensor readings for successive elaboration by the Guidance, Navigation, and Control system embedded in the autopilot; in addition, the simulator receives all control inputs from the autopilot, thus closing the simulation loop with the actual autopilot hardware.

Hardware In the Loop (HIL) simulation is a common approach for validation and qualification of the functionalities of pieces of actual hardware and firmware before installation into the operational environment. For the scope of this paper, the piece of hardware and associated firmware under test is the autopilot. Just as reference, the autopilot and the simulator were interfaced using CANbus, and exchanged 10 CANbus packets at a rate of 100 Hz: 7 from simulator to autopilot for accelerometers, gyroscopes, magnetometers, baro-altimeter and GPS data, and 3 for autopilot commands and various debug data. The simulator timing was slaved to that of the autopilot (that was actually running in real-time) so that the CANbus datalink served both for data exchange and timebase synchronization. More details can be found in the cited references. In addition, since, for this research the man-machine interface is a relevant issue that must necessarily provide a fully immersive and realistic experience for the pilot, the simulation system employs the "real" input device (actually a COTS remote control radio that was connected to the autopilot with the real wireless communication channel), and not a virtual reproduction of it.

As for the synthetic environment, it is well known, and experimental evidence can be easily found of the importance of using a scenery that is representative of real places on earth as opposed to imaginary non-existent places: full immersion sensation and focus on the tasks is higher when the pilot is presented with a 3D scenery of a place he knows, and the same place is represented in 2D (map view) on the ground station screen. The simulation environment employed for the virtual experiments presented in this paper is built using a complex communication architecture that uses Dynamic HTML tools and the UDP/IP protocol to interface GoogleEarth with our simulator in real-time. This allows to prepare simulations exactly at the geographical location of interest and benefit of the immense GoogleEarth database of terrain elevation, buildings, trees and other landmarks. Just as a reference, the scenery graphics refresh rate was around 30 Hz, thus fast enough for immersive simulation. Figure 4 compares a sample view taken at the ground level from the Simulator with an actual photograph shot approximately from the same location, and with a similar focal length. The actual photograph also shows an overlay of synthetic imagery.

Finally, another important component of the simulation environment is the ground station; since an HIL simulation was performed, there was no need to use a mock-up or a virtual ground station but the actual ground station hardware, software and communication channels were used. Modern unmanned vehicles require advanced functionalities, thus the control and mission management software may be very complex and difficult to validate and qualify; an hardware in the loop simulator is a useful tool since it allows testing of the actual hardware/software combination that will be employed in operation.

This virtual prototyping and testing approach constitutes a powerful engineering tool for development, initial testing and tuning of the algorithms and graphical means used to depict the safety regions and virtual fences. At the same time, the simulator can be regarded as a practical tool for pilot training with the visual tools proposed in this
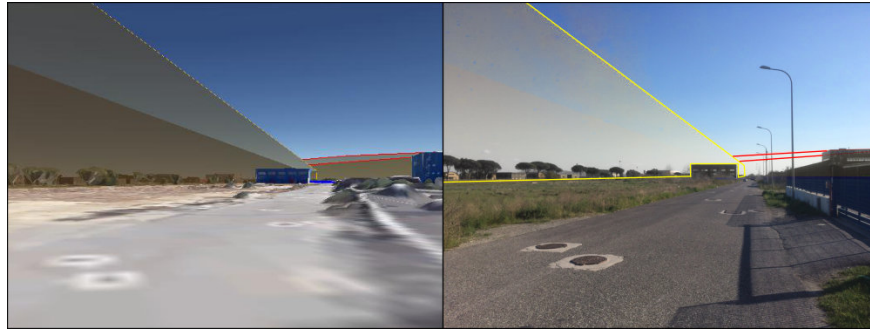
Figure 4. Sample comparison of virtual (left) and real (right) imagery.

paper. It may also serve as a briefing tool for pre-mission analysis: the intended mission may be pre-performed virtually with the simulator to discover and highlight potential issues.

In addition to this, augmented reality techniques [17;18] are becoming more and more reliable and precise, thus the proposed approach can be surely implemented, as depicted in Fig. 4, within an augmented reality display where the pilot will see actual imagery coming from vehicle camera with the synthetic graphics of the safety fences as tridimensional geo-referenced overlay. This approach would render our proposed solution even more useful and intuitive.

The operational scenario

A demanding operational scenario of solar panel farm health assessment and survey using airborne sensors [16] will be considered in what follows, for evaluating the proposed set of visual aids for a remote pilot, and in particular the safety-fencing technique. Large-scale solar facility operators can spend thousands of man hours inspecting, identifying and isolating faulty, dirty or shaded solar panels. These maintenance operations can be very expensive and may have also an impact on the competitiveness of solar energy pricing. As the solar energy industry continues to grow, large-scale solar facilities will become more common and competition will urge novel health assessment and management techniques. Defective panels convert less radiation into energy than functioning panels, thus they become warmer; at the same time, similar overheating issues may be due to local electric network malfunctioning. This temperature imbalance can be easily measured from distance using an airborne infrared camera [16]. Thus, the use of unmanned aerial vehicles may allow solar farm operators to obtain real-time and automatically geo-referenced inspection data at a significantly lower cost than having ground inspectors reach each single panel by walking in the field.

Sometimes solar farms are located far away from populated areas allowing to perform unsupervised automatic missions by unmanned aerial vehicles; this is often not the case for European plants where the population density throughout the country does not allow enough distance from populated areas. This is indeed the case of our test scenario: an actual solar farm in the South neighborhood of the town of Pisa, Italy. The

mission we took as reference for demonstrating the safety-fence technique presented in this paper consists thus in performing the survey of the solar panel farm mentioned above, which is located in close proximity of critical infrastructures and populated areas. A satellite view of the area where the simulation tests were performed is reported in Fig. 5.
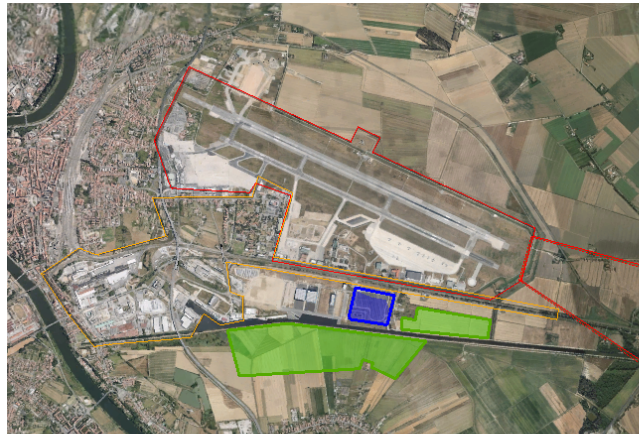

Figure 5. Map of the area of the simulation tests (solar panels area in blue).

This scenario presents several issues:

- the area is close to an airport (Pisa International Airport) even if not directly on the landing corridor. Flying on these areas is prohibited by law and an emergency controlled fault must be avoided as well;
- area is very close to a road with high traffic density; an emergency landing or post-fault crash in this area must be absolutely avoided;
- area is close to waterways and an inland marina used by local shipyards; landing or falling into water must be avoided, provide this would imply serious damage or loss of the equipment, even if, from a safety standpoint (that is, minimizing the risk of harming people or goods), these areas might be a better option than others;
- area is close to populated areas (private houses, public offices, industrial area); no fly over, emergency landing or crash is permitted.

As a further problem, one should consider that landing inside the surveyed site should be also avoided to limit possible damages to the solar panels, whereas unpopulated areas in the surroundings should be used for safe automatic landing in case of emergency landing, following vehicle faults, or emergency landing in case Air Traffic Control requires to clear the area (as when an aircraft with technical problems, or in emergency, is approaching for landing on the nearby runway).

All these different issues can be mapped into areas or volumes of different types and shapes, where an area is defined by a 2D domain at the ground level (and could be easily visualized on a map), whereas a volume is defined by a 3D domain. The safety fence envisioned in this paper must guarantee that the constraints represented by areas

and/or volumes introduced above will never be infringed. To this extent, three different domains were defined to model the possible constraints coming from the specific real problem:

1. No Fly Volume (NFV);
2. No Landing Area (NLA);
3. Safety Landing Area (SLA).

The NFVs, together with the NLAs concur to define the safety-volume defined as: *the 3D volume of all possible positions that the vehicle may reach while guaranteeing that, in case of a major accident with total loss-of-control resulting in a non-predictable trajectory, and the activation, after the prescribed latency, of a safety power-off function, the vehicle will never land on any NLA, and will never fly inside any NFV.*

The safety landing areas do not have a direct impact on the safety volume; nonetheless we have found, during experiments, that pilots benefit in terms of spatial awareness during the flight by seeing SLAs displayed as patches over the ground. The safety fence that will be shown in 3D graphics to the pilot, is actually the volume safety volume border; the fence coloring will be adapted according to the distance between vehicle predicted position and the fence itself and vehicle maneuvering state and capabilities as described in the previous sections. It must be stressed that the safety volume and thus its frontier, the safety fence, are dynamic objects that change in time depending of two major factors:

- vehicle approach speed (towards the fence);
- changing environmental conditions (other vehicles passing in the vicinities etc.).

When the vehicle is approaching quickly the border of NLAs or NFVs, the fence should be adapted to vehicle speed since, as the impact point analysis proposed in the previous section reveals, the risk of falling beyond a NLA is higher than that of a slower vehicle at the same distance from the NLA border. If the vehicle slows down, then the fence can be bent back by an appropriate amount according to the impact point analysis. In order to simplify the implementation of the safety fence graphics, the safety fence shape could be kept constant for a given value of velocity (i.e. for a constant σ), and only its coloring might be adapted. The net result would be a 3D static fence but with a color scheme that takes into consideration vehicle speed. This solution is the actual one implemented in the simulations presented in the following section.

Another vehicle passing in the area can be treated within this framework as a time varying NFV: in such a case, the safety fence concept is used as a collision avoidance system. This is particularly simple in all scenarios where the mission is performed by a fleet of cooperative vehicles and GPS signals from different vehicles are available to the GCS. Each aircraft behaves as a position constraint depicted to the pilot as a spherical moving fence, as in Fig. 7, where a red sphere surrounds another quad-rotor vehicle flying too close, thus increasing its visibility. The intruder vehicle can be tracked in position, velocity and acceleration and an appropriate prediction of its feasible and possible future trajectory can be obtained with standard target tracking and prediction techniques.[8;9] A No Fly Volume can then be built around the intruder predicted trajectory and displayed to the pilot as an additional constraint. Possibility

of a violation of the moving fence surrounding another vehicle can be estimated in almost the same way as that of the fence surrounding the allowed operational area, although in this second case it is necessary to predict also the future position of the neighboring vehicle(s). This is relatively easy, in the cited cooperative scenario, when flight condition and maneuver state are available also for other vehicles, and provided that the computational effort of the prediction is modest. The same color code can be used for highlighting those cases in which a neighboring vehicle is in a potentially dangerous collision course. The dynamic nature of the fence, in terms of both shape and color, can also be adopted in order to change the bounds or the ceiling of the operational area (e.g. creating a corridor for a larger manned vehicle, such as an helicopter, along a known route). This additional NFV, together with the static ones defined previously, contribute to the definition of a new dynamic safety fence.

The specific scenario selected for our tests, the solar panel field south of Pisa, was modeled using two NFVs (both red in Fig. 5), one NLA (orange in Fig. 5) and two SLAs (green in Fig. 5). The first NFV corresponds to the volume contained above the perimeter of the Pisa Airport. The second NFV consists in an approximated landing corridor for Pisa Airport runway RWY 04. The NLA consists in the perimeter of all nearby populated areas including the road passing by the solar panel field. The two SLAs were selected as the closest unpopulated areas around the survey consisting in crop fields. Finally the area in blu is the target area to be surveyed where the actual solar panels are. Figure 6 shows a sample perspective view, taken from an altitude of about 100 meters of a portion of the scenery.
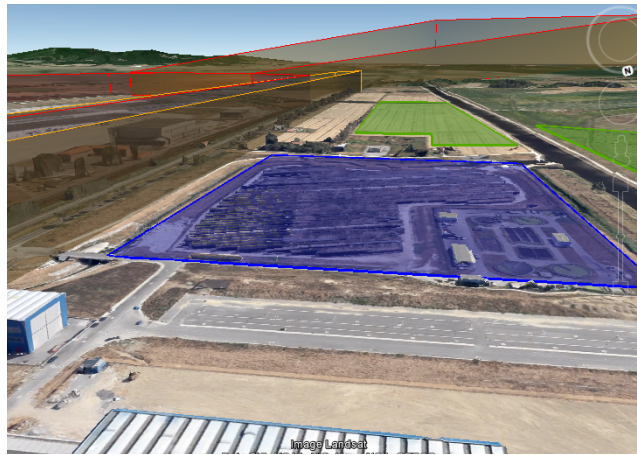

Figure 6. Perspective view of different areas and volumes.

Given all these constraints, the safety volume was built according to the impact distance analysis performed in the previous section, namely on Equation 11. The safety fence represents the frontier of the safety volume; clearly the the bounds of this 3D domain starts from the limits plotted on the map, but it shrinks toward the inside of the safety volume at higher altitude since it takes into account the possibility of reaching areas farther away from the current position of the vehicle in case of uncontrolled faults.

As a consequence, the fence appears curved toward the inner parts of the safety volume. Figure 7 shows a snapshot taken during one of the tests showing a view taken from the simulated on-board camera of the vehicle; the safety fence is the yellow translucent grid in this case. The fence altitude was set, in these tests, to 70 meters since this is the altitude limit set currently by Italian National Regulations for Remotely Piloted Aerial Vehicles.
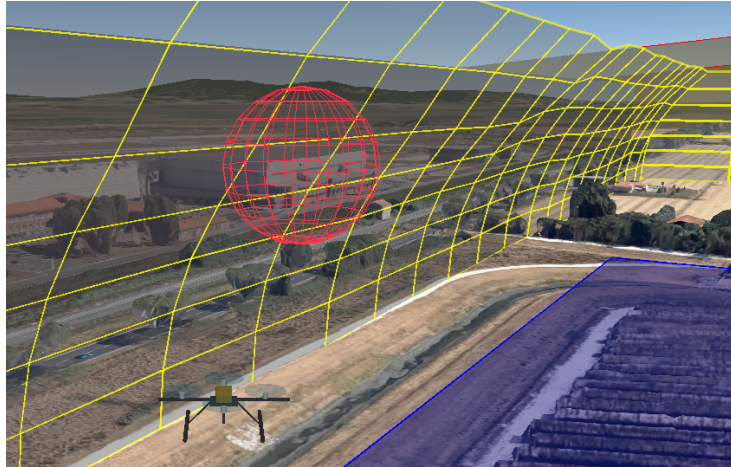

Figure 7. Sample simulated pilot view.

Test Results

Several simulated missions were performed using the simulator described in the previous section. Aim of the simulations was to test the overall system with remotely piloted vehicles and assess the effectiveness of the safety fence warning system with respect to avoiding penetration into areas where a loss of control event might result into violation of a NFV or landing/crashing onto a NLA. Although a throughout analysis and assessment of the system performance would require a set of structured tests with at least two sets of pilots flying in different augmentation conditions (with and without fence), the preliminary tests performed with the system already outlined a net situation. Figure 8 shows the results of three such tests. The pilot in the three tests was instructed to take off and fly toward the insertion point in the survey pattern (the east most corner of the blue area); then the pilot had to conduct the vehicle along the solar panel area edge to simulate a slow flight along the first leg of the aerial survey. The pilot was flying the vehicle in (simulated) First Person View (FPV) mode: the pilot controlled the vehicle, as he was on-board, by looking at camera images coming from a (simulated) fixed forward looking camera mounted on the vehicle. The FPV display was augmented, in a (simulated) augmented reality, with imagery of the safety fence, the SLAs and the target area.

In the first two flights, performed at different altitudes, the pilot was able to see the safety fence as in Fig. 7, while the display was switched off for the third test. In the

first two flights the pilot easily noticed the transition from the normal coloring of the fence to the color indicating the need to reduce speed; the pilot soon decreased speed and adjusted the trajectory of the vehicle to remain well inside the fence, and not to reduce excessively the distance from it. These safety maneuvers happened at different distances from the fence footprint (the intersection of the fence with the terrain at 0 elevation) because the pilot was flying at different altitudes; flight 1 was flown at a higher altitude, and, as expected, the safety maneuver was initiated earlier than in flight 2 because the fence is curved towards the inside of the safety volume. The rest of the mission was flown at approximately constant distance from the fence for both Flight 1 and 2.

In test flight 3 instead, the safety fence display was off, and the pilot could only see the outside environment. As a matter of fact, the pilot was not aware of the actual distance from the road (or from the borders of the NLAs and NFVs), did not receive any graphical warning, thus he did not try to correct his trajectory. We should stress the fact that FPV flight is very complex and it is very prone to spatial disorientation; furthermore perception of depths and very simple navigation tasks require a very expert pilot. We could also notice that the trajectory flown in this third test flight was smoother in the initial part, from take off to insertion along the survey leg, but later, along the solar filed, it appears "less linear" than in the other two flights. A possible explanation is that following the survey leg (actually parallel to the fence) was easier for the pilot when the safety fence was displayed, than when it was not present.

During initial tests we also noticed that the fence coloring has a strong beneficial effect on the capability of the pilot to judge the vehicle distance from the fence itself: the pilot was able to keep an almost constant distance from the fence by "tracking a constant coloring", while it was very difficult for him to judge his distance when the fence color was always constant. Actually fence coloring is needed only when the pilot is getting too near to the fence: when the pilot is far away enough from the fence, so that it is not necessary to pay much attention to it, the color can be kept constant and, must only be easy to be seen from afar; while when he gets too close to the fence he must be warned of this, and we found that adapting fence color with distance (either continuously or in steps) provides sufficient information for the task of flying at a certain distance.

Fair and statistically sound conclusions cannot be drawn from a small set of flights with just one pilot; nonetheless, this preliminary set of tests indicates that the safety fence concept represents a valuable visual/graphic element easy to detect and also to interpret. Showing the fence as a grid was found useful for telling relative velocity and, to some extent, distance. The sudden change of fence color (from yellow to green), happening at the border of the area of attention, was very easy to detect and exploit in terms of implementing corrective maneuvers. The fence curvature toward the inside of the safety volume, although formally and mathematically justified by the impact point analysis, was also found very useful as a visual reference for altitude with respect to ground. Upon all these considerations, we believe that the visual aids for conflict prevention that we propose in this paper could be effectively implemented on unmanned vehicles ground control stations to help pilots safely performing their missions both in the presence of static constraints (i.e. fixed infrastructures) and dynamic constraints (i.e. other vehicles approaching the area).
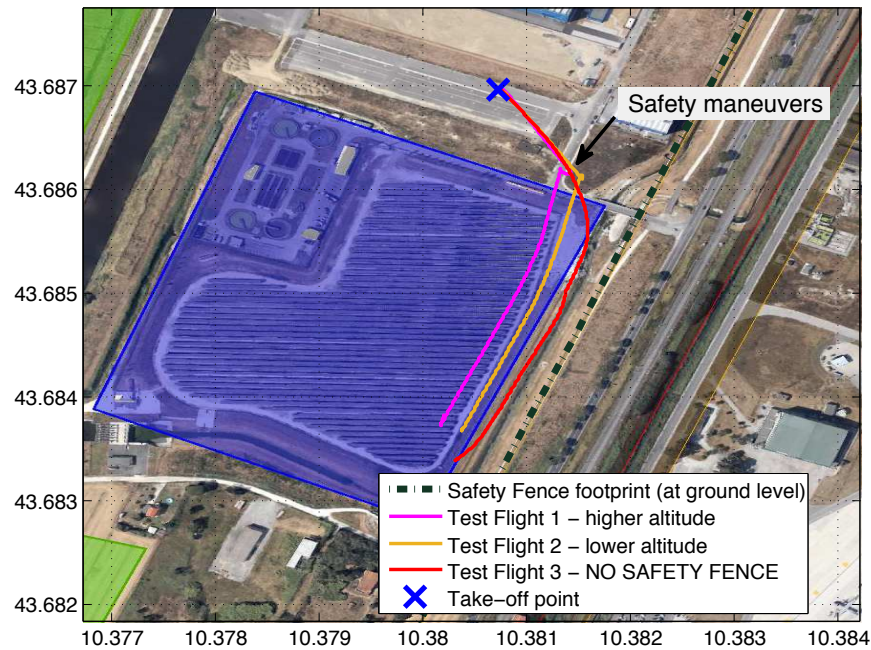
Figure 8. Map view of the three sample flights.

Conclusions

The paper presented a novel approach for increasing situational awareness and safety for missions with remotely piloted vehicles. A worst case analysis was proposed for a certain class of faults that may cause, even if for a short period of time, an uncontrolled vehicle motion followed by a crash into the ground. The analysis results were used to design what was defined as the Safety Fence, that is, a tri-dimensional dynamic graphical element to be used in an augmented reality fashion. The Safety Fence shape, curvature and color change during the mission, and depend not only on the surrounding pre-defined no-fly-zones and no-landing-areas, but also on vehicle current state (altitude and speed), and possibly on the presence of other vehicles in the vicinities. Flying in the volume contained inside the Safety Fence guarantees that, for the class of faults taken into considerations, the vehicle will never fly inside or crash into a forbidden area. A complex solar panel field survey scenario, originated from a specific need, was proposed for evaluation of the proposed techniques. Several simulated flight were performed within a highly realistic flight simulator, in which the Safety Fence proved to be a visual element easy to detect, interpret and exploit to improve pilot perception of vehicle position, and associated risks.

References

1. Casber D.W., Beard R.W., McLain T.W. (2005). Forest Fire Monitoring With Multiple Small UAVs. *2005 American Control Conference, Portland, OR, June 8-10 2005.*

2. Rudol P., Doherty P. (2008). Human Body Detection and Geolocalization for UAV Search and Rescue Mission Using Color and Thermal Imagery. *2008 IEEE Aerospace Conference. Big Sky, MT, 1-8 March 2008.*

3. Harwin S., Steve and Lucieer A. (2012). Assessing the Accuracy of Georeferenced Point Clouds Produced via Multi-View Stereopsis from Unmanned Aerial Vehicle (UAV) Imagery. *Journal of Remote Sensing*, Vol. 4, No. 6 : 1573-1599.

4. EASA, Unmanned Aircraft Systems (UAS) and Remotely Piloted Aircraft Systems (RPAS), Available from *https://easa.europa.eu/unmanned-aircraft-systems-uas-and-remotely-piloted-aircraft-systems-rpas*; [1 September 2015].

5. JARUS, Joint Authorities for Rulemaking on Unmanned Systems, Available from *http://jarus-rpas.org/*; [1 September 2015].

6. Sachs, G. (2000). Perspective Predictor/Flight-Path Display with Minimum Pilot Compensation *Journal of Guidance, Control, and Dynamics*, 23(3):420-429.

7. M. Mulder, A. R. Veldhuijzen, M. M. van Paassen, and J. A. Mulder (2005). Integrating Fly-by-Wire Controls with Perspective Flight-Path Displays. *Journal of Guidance, Control, and Dynamics*, 28(6):1263-1274.

8. Innocenti, M., Pollini, L., Giulietti, F. (2001). Visual Tools for Man-Machine Interface Real Time Simulation. Proceedings of IEEE Symposium on Information Technology in Mechatronics, ITM01, Istanbul, Turkey, October 2001.

9. G. Avanzini (2004). Frenet-Based Algorithm for Trajectory Prediction Journal of Guidance, Control, and Dynamics, 27(1): 127-135.

10. L. Pollini, A. Metrangolo (2008) Simulation and Robust Backstepping Control of a Quadrotor Aircraft, AIAA Modeling and Simulation Technologies Conference, Honolulu, Hawaii, 2008.

11. Dreier, M.E., Introduction to Helicopter Flight and Tilt Rotor Flight Simulation, AIAA Education Series, USA, Feb. 2007, Chs. 2, 4, and 11. doi: 10.2514/4.862083

12. Leishman, J.G., Principles of Helicopter Aerodynamics, Second Edition, Cambridge Aerospace Series, USA, Apr. 2006, Ch. 2.

13. Mettler, B., Identification Modeling and Characteristics of Miniature Rotorcraft, Kluwer Academic Publishers, Norvell, 2003, Ch. 3.

14. F. Di Corato, M. Innocenti, L. Pollini (2012). Combined Vision-Inertial Navigation for Improved Robustness, IEEE Israel Itzhack Y. Bar-Itzhack Memorial Symposium on Estimation, Navigation, and Spacecraft Control, Haifa, Israel 14-17 October 2012.

15. F. Di Corato, M. Innocenti, L. Pollini (2013). Robust Vision-Aided Inertial Navigation Algorithm via Entropy-Like Relative Pose Estimation. Gyroscopy and Navigation, 4(1): 1-13.

16. Denio, H. (2012) Aerial solar Thermography and condition monitoring of photovoltaic systems. 38th IEEE Photovoltaic Specialists Conference (PVSC). pp. 613-618. 2012.

17. Hutton J., Mostafa M.M.R (2005) 10 Years of Direct Georeferencing For Airborne Photogrammetry. GIS Business (GeoBit), 11(1), 33-41.

18. Nebiker S., Eugster H., Flckiger K., Christen M. (2011) Planning and Management of Real-Time Geospatial UAS Missions within a Virtual Globe Environment UAV-g 2011, Conference on Unmanned Aerial Vehicle in Geomatics.