AIAA SciTech Forum
January 3-7, 2022, San Diego, CA & Virtual
AIAA SCITECH 2022 Forum

10.2514/6.2022-1014

Use of Dynamic Pose to Enhance Passive Visual Tracking

Dennis J. Marquis* and Craig A. Woolsey†
Virginia Tech, Blacksburg, Virginia 24061

Passive visual tracking performance has drastically improved in recent years, thanks to advances in hardware and algorithms, but scenarios in which small mobile threats are indistinguishable from complex backgrounds remain challenging. There is an opportunity to explore whether the maneuverability of a multirotor can be leveraged in order to strategically influence perspective, an approach that can complement existing computer vision strategies. This paper describes an algorithm to determine candidate perspectives and plan host motion that will produce and maintain these perspectives, in order to improve threat detection. A baseline problem abstraction of tracking a mobile sphere is presented with appropriate assumptions, to validate the methodology.

I. Introduction

Drone based, passive intruder detection and tracking is valuable because it is discreet, lightweight, and highly maneuverable. Specifically, electro-optical (EO) tracking, also known as visual tracking, has been used increasingly for situational awareness due to its low cost and continuously improving algorithms, including feature extraction and blob analysis algorithms [1]. However, commercial off-the-shelf (COTS) vision-based systems capable of achieving high detection rates often suffer from a large number of false positives, making them unreliable in some environments [2]. Efforts to improve visual tracking include the addition of skyline extraction algorithms [3], background subtraction algorithms, or even use of trained classifiers [4], but these methods can still often fail to discriminate small threats in complex scenes. A relatively unexplored possibility is to complement existing strategies by leveraging the maneuverability of a drone in order to influence the background perspective.

Theory about motion planning with respect to a moving threat has been developed for other applications. LaValle et al. presented a control law to maintain visibility of threats [5], using a Boolean concept of visibility. However, the focus is on two-dimensional planning when the view can be obstructed by obstacles. Cancemi et al. broadened the concept of visibility to a stochastic model [6], but this formulation emphasizes the issue of obstacles, similar to an art gallery problem. Potentially most promising in the context of aerial robotic vision is the use of model predictive control (MPC) with quadrotors [7], which reduces complexity of the planning problem by iteratively minimizing a cost function over a limited time horizon. MPC algorithms have been implemented with drones to support aerial videography [8] [9], with cost functions that consider properties such as size of target, viewing angle of target, and target position within the image plane.

The research described here explores whether the maneuverability of an aircraft can be exploited to produce and maintain perspectives that will improve visual threat detection and tracking. The objectives are twofold: develop a strategy to predict optimal host-threat-background perspectives and develop a control law capable of enforcing these perspectives. We consider an abstracted problem which isolates the general prediction and control challenges from problem-specific issues of computer vision within a particular scene. The remainder of this paper is organized as follows. Section II establishes the host, threat, and background assumptions that will be used when applying the proposed strategy. Section III details the methodology of the threat tracking strategy. Section IV presents some scenarios that highlight how the algorithm might behave under various initial conditions and tuning parameters. Finally, Section V provides some concluding remarks and opportunities for future work.

II. Problem Formulation and Assumptions

Detecting and tracking a moving threat in an arbitrary environment is a complex task, whose performance is dependent on choice of detection and tracking algorithms, choice of hardware and sensors, visual and geometric characteristics of the background with respect to the threat, and dynamics of the threat, among other variables. Although a long-term objective is to develop and implement a complete solution in a field experiment or photo-realistic simulation,

*Graduate Student, Crofton Department of Aerospace and Ocean Engineering, Student Member AIAA. dennisjm@vt.edu
†Professor, Crofton Department of Aerospace and Ocean Engineering, Associate Fellow AIAA. cwoolsey@vt.edu

this paper presents a general strategy, for an abstracted problem, that is agnostic to the underlying computer vision algorithms. This strategy is formulated to address the following simplified task: 3D tracking of a constant velocity black sphere against an arbitrary grayscale background by controlling the motion of a remote visual sensor. Relevant assumptions are detailed below and summarized in Table 1.

A. Host Assumptions

Consider a multirotor host with a mounted gimballed camera. This host has sole responsibility for tracking the threat. (This formulation will not consider how a network of hosts should coordinate to collectively optimize coverage of the threat.) The host is able to perfectly self-localize, so its position and orientation will always be known in a 3D world coordinate frame; there is no uncertainty stemming from imperfect GPS or IMU readings. In addition to live video feed from the camera, the host has knowledge of the range to its threat and background. If using solely passive methods, this range estimate could come from monocular ranging techniques [10], stereo vision techniques [11], or acoustic techniques [12]. If allowing for active technologies, camera data could be supplemented with radar [13] to obtain range. Use of an off-board sensor such as ground-based radar to obtain range is also viable. The camera can freely translate its perspective due to the multirotor's vertical and lateral mobility. Camera attitude in all three axes is controlled by the multirotor and gimbal attitude controllers in such a way that the camera remains level, so that only its pitch and yaw attitude vary. Considering translational camera motion in 3 dimensions and rotational motion about 2 axes yields 5 total degrees of freedom. Finally, the camera is assumed to be a pinhole camera with fixed focal length, so lens distortion effects and zoom will not be considered.

B. Threat Assumptions

The threat is a solid black sphere moving at constant velocity. Its color is a property that is known to the host. The sphere exhibits Lambertian reflectance, so both its geometry and visual characteristics are invariant when viewed from different perspectives. The constant velocity implies that the threat is both non-cooperative and non-antagonistic, so it will not modify its trajectory in an attempt to defeat the tracking algorithm. Furthermore, assume the threat has already been detected and resides in the center of the image plane of the camera by means of a visual-servoing strategy. Since the host knows the range to the threat and its location in the camera's image plane, it effectively knows the 3D world coordinates of the threat as well. Although the particular tracking algorithm is not specified, assume a traditional combination of corner detector, optical flow algorithm, and Kalman filter with a constant velocity model, an approach that has been validated by Kang et al [14].

C. Background Assumptions

The host continuously tracks the threat against a grayscale background. Justification for the use of grayscale will be discussed in Section III. This background occupies the entire image plane and objects in the background are assumed to be the same distance from the host (i.e. no variability in depth of background). The background is defined to be at a range that is substantially larger than the range from the host to the threat, in order to assume that the background is invariant to translation of the host aircraft position for small time scales. There are no additional obstacles in the environment that could occlude the threat or disrupt the motion of the host or threat.

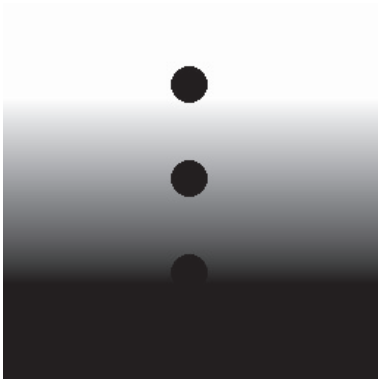
III. Proposed Methodology

A. Quantifying Visibility

The first step towards producing an optimal perspective is to use a metric to identify candidate locations in the background, relative to the host position, that provide greatest contrast between the threat and the background so that the threat is more visible. Intuitively, the threat will stand out more clearly against a simpler background with a distinct color. Visibility and background complexity are extremely contextual properties, so there is no universal metric that can be applied. One approach is to consider "image discriminability," defined as the visibility of the difference between a pair of images, which arguably quantifies object detectability [15]. For threat detection, the images considered would be an image of the unobstructed background and a nearly identical image with the threat present. Attempts to model image discriminability by a human observer include the HDR-VDP-2 metric [16], which applies to a generic scene, as well as the Spatial Standard Observer (SSO) metric [17], which applies specifically to aircraft. For automated detection

Table 1 Host, Threat, and Background Assumptions

#	Assumption
1	Single host
2	Perfect host self-localization
3	Host has knowledge of finite threat and background ranges
4	5 DOF camera motion (no roll)
5	Pinhole camera model
6	Single threat
7	Constant velocity threat motion
8	Threat is solid black sphere that exhibits Lambertian reflectance (host knows this)
9	Threat has been detected and is centered in image plane
10	Grayscale background with no depth variability
11	Range to background is much larger than range to threat
12	No obstacles (no occlusion, no collision risk)



(a) Image of three spheres in gradient background. The spheres become progressively more occluded.

(b) Sobel edge detection is applied to the image. Only the uppermost portion of the bottom sphere is detected.

(c) Canny edge detection is applied to the image. More edges are found than with Sobel edge detection.

Fig. 1 Relationship between pixel intensity and detection success

using computer vision techniques, a summary of approximately 50 candidate metrics is provided by Peters et al. [18]. In general, the applicability of each metric varies based on how camera frames are processed and which detection algorithm is used.

The problem of selecting the perfect metric can be circumvented by allowing the control strategy to accept any scalar metric. Assume such a metric has been selected for a particular use case and can be computed either per pixel or per discretized segment of each camera frame. Candidate regions in the background correspond to segments that produce the highest metric. Finding a host trajectory that improves perspective then becomes a purely geometric problem, which is outlined in upcoming sections.

For the abstract problem of tracking a black sphere, it is sufficient to use the grayscale intensity of the background as a metric. Any region in the image plane with intensity above some designated threshold can be considered a “good" prospective background. To illustrate this, consider the gradient background depicted in Figure 1a. Three spheres have been placed at varying relative altitudes within the scene. The upper two spheres are clearly visible and would likely be detected by a detection algorithm. The bottom sphere is partially obscured by the background. Even for such a simple metric, detection success could vary by choice of algorithm, which is apparent when comparing the bottom sphere’s edges produced by the Sobel and Canny edge detectors in Figure 1b and Figure 1c, respectively. Each edge detector would have its own ideal intensity threshold, where the Sobel algorithm requires a stricter threshold.

B. Maintaining Perspective with respect to a Reference Point

Maintaining perspective requires that the motion of the host responds appropriately to the motion of the threat. The host to threat to background geometry for one discrete time step of length ΔT is outlined in Figure 2. Note that although this scenario takes place in 3D, the threat initial position, threat final position, and reference point define a plane, the x-y plane shown in Figure 2. A vector drawn from the host through the threat will intersect the background at some reference point, which is the origin of this coordinate frame for convenience. The host and threat are initially at distances d_h and d_t, respectively. The threat is moving at constant velocity v_t at a heading angle ϕ with respect to its position vector, defined from $[-\pi, \pi)$. It travels distance $v_t \Delta T$ over the time step, which sweeps angle θ. The defined distances are related to θ by:

$$\tan\theta = \frac{v_t \Delta T \sin\phi}{d_t + v_t \Delta T \cos\phi} \tag{1}$$

Assume the host and threat are positioned in such a way that the host has an unobscured view (i.e. the visibility metric is high). The host's objective is to maintain this relative orientation, so that the threat does not shift with respect to the background. If attempting to make the background appear static with respect to the threat, the vector from the origin to the threat must also intersect the host after the ΔT time step. The shortest host trajectory from its initial location to this vector is one that is perpendicular and within the same plane, which reduces to a 2D problem. Therefore, the geometry in Figure 2 is applicable for both vertical and lateral movement. The host must travel at a velocity v_h, defined by:

$$v_h = \frac{d_h \sin\theta}{\Delta T} \tag{2}$$

When background to threat distance, d_t, increases towards infinity, θ approaches 0 and $d_t + v_t \Delta T \cos\phi$ approaches d_t, resulting in the approximation:

$$v_h \approx \left(\frac{d_h}{d_t}\right) v_t \sin\phi \tag{3}$$

There are a few important takeaways from this construction. The effort (i.e. v_h) required by the host to maintain perspective is a function of background distance, threat distance, and threat heading angle. The host can remain stationary when the threat is moving towards or away from it, corresponding to heading angles of 0 and $\pm\pi$. The ratio $\frac{d_h}{d_t}$ will always be greater than 1, so for some heading angles v_h will be greater than v_t, placing a performance requirement on the host relative to the threat. Maintaining perspective is most challenging when the threat moves perpendicular to the background-threat-host vector, at heading angles $\pm\frac{\pi}{2}$, since (3) is maximized with respect to ϕ. If the host is in visual range of the threat, this implies that the ratio $\frac{d_h}{d_t}$ approaches 1. Maintaining perspective of the threat then reduces to the trivial case of matching the threat's velocity that is orthogonal to the background-threat-host vector, described by the term $v_t \sin\phi$.

C. Defining Locations with Acceptable Threat Visibility in 3D

Maintaining a good view of the threat does not require maintaining perspective with respect to a single reference point. Instead, it requires maintaining perspective with respect to any reference point designated as "good" by the background visibility metric. Regions of pixels with acceptable visibility metrics can be can be used to define 3D volumes within which a host can view the threat with high contrast. Consider a circular region, for example, and suppose one draws a vector from each point in this region to the threat, thus defining a conical volume. Extending the vectors through the apex defined by the threat, one obtains a second cone within which the host would be able to view the threat. See Figure 3. The interior of this cone defines the set of all 3D points that produce a "good" perspective.

In this implementation, background regions with high visibility scores are assumed to be elliptical. This choice should be considered an abstraction for illustrative purposes, though one may choose to inscribe a maximal ellipse within an irregular region of high visibility as a means of simplifying later computations. Considering elliptical regions provides more flexibility than circular regions, but also retains elements of symmetry that will be useful in the upcoming cost function derivation. Consider the ellipse defined in Figure 4, centered at (x_c, y_c), with semi-major axis a, semi-minor axis b, and rotation γ. The set $\mathcal{E}_{\text{ell,pix}}$ of all camera pixel coordinates contained within this ellipse is:

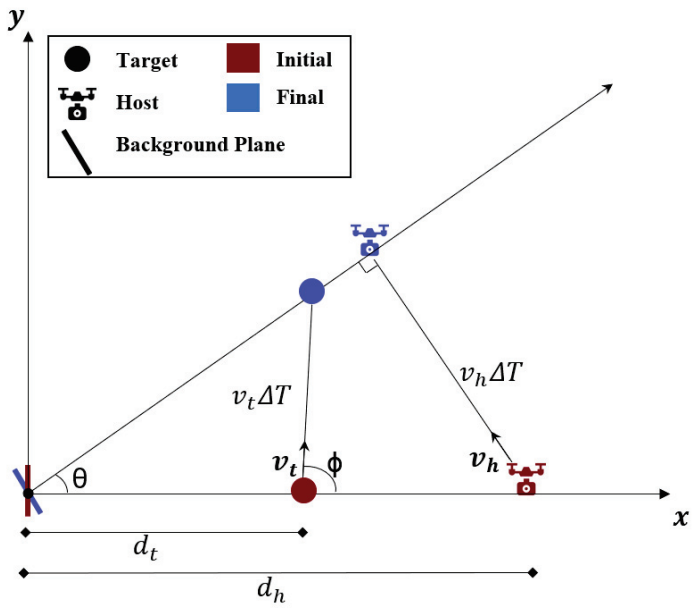


Fig. 2 Geometry of maintaining threat and background perspective for one time step. The threat, initially at distance d_t from a reference point in the background, travels from the initial to final position, covering distance $v_t\Delta T$ and sweeping angle θ with respect to the reference point. The host, initially at distance d_h from the reference point, and located along the same line defined by the reference point and threat, responds by finding the closest point along the new vector from reference point to threat. Attaining this new position requires traveling at velocity v_h during the ΔT time step.

$$\mathcal{E}_{\text{ell,pix}} = \{\mathbf{x} | (\mathbf{x}-\mathbf{x_c})^T \mathbf{R}^T(\gamma)\mathbf{P}^{-1}\mathbf{R}(\gamma)(\mathbf{x}-\mathbf{x_c}) \le 1\}$$
$$\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix}, \quad \mathbf{x_c} = \begin{bmatrix} x_c \\ y_c \end{bmatrix}, \quad \mathbf{P} = \begin{bmatrix} a^2 & 0 \\ 0 & b^2 \end{bmatrix}, \quad \mathbf{R}(\gamma) = \begin{bmatrix} \cos\gamma & -\sin\gamma \\ \sin\gamma & \cos\gamma \end{bmatrix} \tag{4}$$

Equivalently, $\mathcal{E}_{\text{ell,pix}}$ can be parameterized by u_1 and u_2:

$$\mathcal{E}_{\text{ell,pix}} = \{\mathbf{x_c} + \mathbf{A}\mathbf{u} | \, \|\mathbf{u}\|_2 \le 1\}$$
$$\mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \quad \mathbf{A} = \left(\mathbf{R}^T(\gamma)\mathbf{P}\mathbf{R}(\gamma)\right)^{1/2} \tag{5}$$

For an ideal perspective camera [19], a physical coordinate (X, Y, Z) in the world frame maps to a pixel coordinate (x, y):

$$\lambda \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} fs_x & fs_\theta & o_x \\ 0 & fs_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{T} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{6}$$

Parameter λ represents the range to the point from the camera and f is the known focal length of the camera. Parameters s_x and s_y are scaling factors where fs_x and fs_y represent the number of horizontal and vertical pixels per unit length, respectively. Parameter s_θ is a skew factor, which is zero when pixels are rectangular. Parameters o_x and o_y are x and y coordinates of the camera's principal point (usually the center of the image plane), measured in pixels. Matrix $\mathbf{R}$ is a matrix in SO(3) that represents the camera's known orientation with respect to the world frame and $\mathbf{T}$ is a 3x1 vector that represents the camera's known position with respect to the world frame's origin. The set $\mathcal{E}_{\text{ell,pix}}$ can therefore be mapped to set $\mathcal{E}_{\text{ell,world}}$, expressed in 3D world coordinates, assuming the background is at known depth λ.

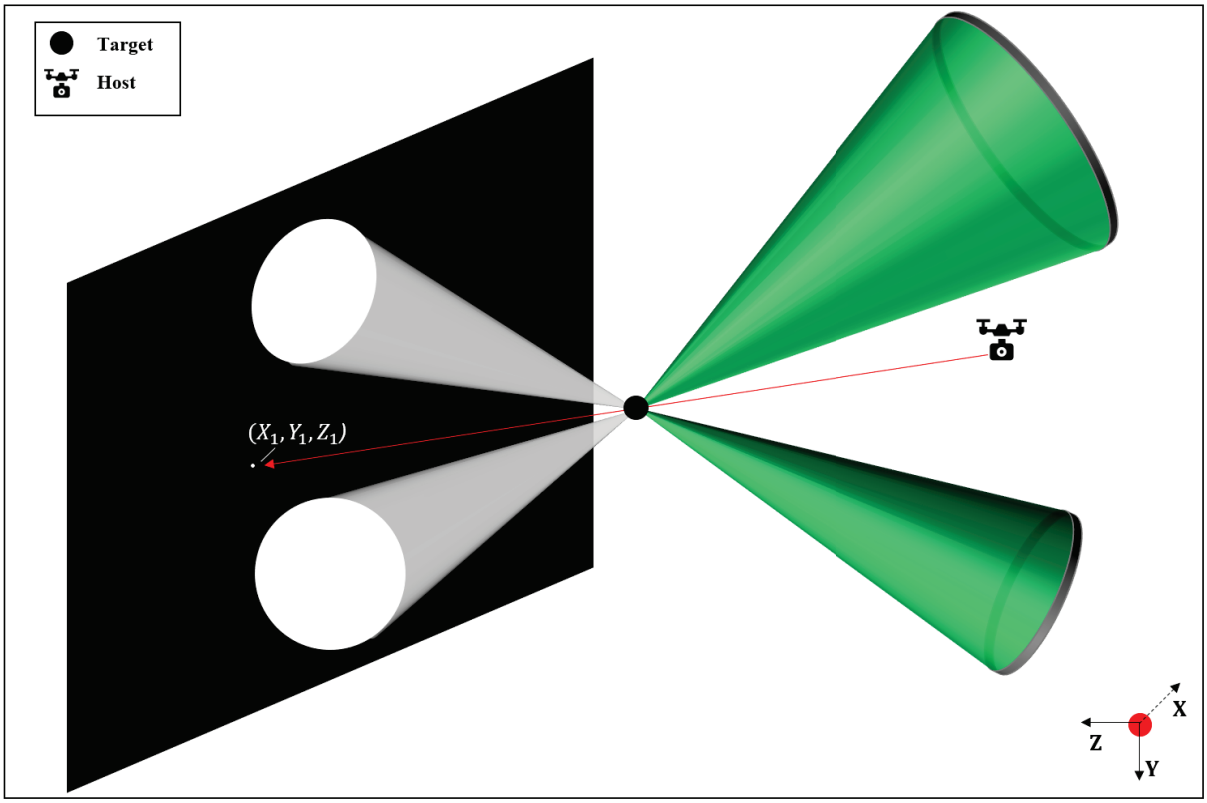


Fig. 3 Visualization of the perspective problem. A black background (left) contains two white regions, against which the black threat would be visible. The host's line of sight to the threat intersects background coordinate (X_1, Y_1, Z_1), yielding poor visibility. If the host were to move into one of the green conic volumes, its line of sight to the threat would intersect the white regions, improving visibility.

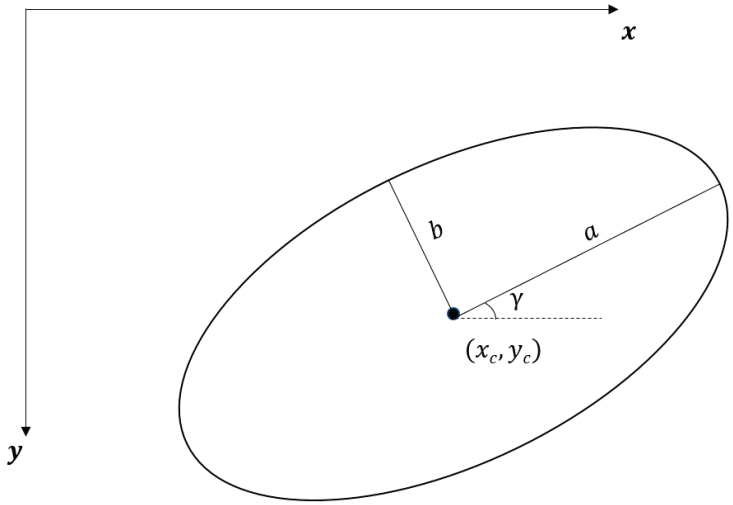


Fig. 4 Ellipse in image plane corresponding to "good" visibility, with center (x_c, y_c), semi-major axis a, semi-minor axis b, and rotation γ

A 3D conical volume can be parameterized in terms of a fixed point, its apex, and the interior of a curve; the curve is known as the directrix. In this derivation, the elliptic cone that defines good visibility of a threat, set $\mathcal{E}_{\text{cone}}$, has apex at

the threat's position in world coordinates, $\mathbf{X}_t$, and planar directrix interior equal to $\mathcal{E}_{\text{ell,world}}$. $\mathcal{E}_{\text{cone}}$ is defined:

$$\begin{gathered}\mathcal{E}_{\text{cone}} = \{\mathbf{X}_t + v(\mathbf{X}_{\text{ell}} - \mathbf{X}_t) | \mathbf{X}_{\text{ell}} \in \mathcal{E}_{\text{ell,world}} \subset \mathbb{R}^3,\ v \in \mathbb{R}\} \\ \mathbf{X}_t = \begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix}, \quad \mathbf{X}_{\text{ell}} = \begin{bmatrix} X_{\text{ell}} \\ Y_{\text{ell}} \\ \lambda \end{bmatrix}, \quad v \leq 0\end{gathered} \tag{7}$$

The scalar parameter v can be thought of as a measure along the cone's axis of symmetry. The apex of the cone corresponds to $v = 0$, while the ellipse in the background plane corresponds to $v = 1$. Parameter v is constrained to negative values to enforce that the threat is always between the host and the background, i.e. the "green" regions in Figure 3.

For backgrounds with multiple candidate regions, multiple ellipses can be defined and elliptic cones can be generated for each ellipse. The task of finding the optimal viewing location becomes the task of selecting the best elliptic cone, as well as finding the best host position within that elliptic cone.

D. Elliptic Cone Selection via Cost Function

There are multiple competing factors to consider when comparing elliptic cones. One viable strategy is to use an optimization algorithm to minimize a nonlinear cost function $J(\mathbf{X}, \mathbf{X}_h, \mathbf{X}_t, \mathbf{v}_t)_i$ iteratively for each cone, , with respect to $\mathbf{X}$, the host aircraft's proposed position vector, subject to the constraint that $\mathbf{X} \in \mathcal{E}_{\text{cone}}$. $\mathbf{X}_h$, $\mathbf{X}_t$ and $\mathbf{v}_t$ are the host's current position vector, threat's current position vector, and threat's current velocity vector, respectively. Objectives can be prioritized within the cost function by applying relative weights. A multi-objective cost function that could be applied to the i^{th} elliptic cone, assuming N candidate cones is the following weighted combination of 4 objectives:

$$\begin{gathered}J(\mathbf{X}, \mathbf{X}_h, \mathbf{X}_t, \mathbf{v}_t)_i = w_1 J_{\text{proximity},i}(\mathbf{X}, \mathbf{X}_h, \mathbf{X}_t) + w_2 J_{\text{robust},i}(\mathbf{X}, \mathbf{X}_t) + w_3 J_{\text{rate},i}(\mathbf{X}, \mathbf{X}_t, \mathbf{v}_t) + w_4 J_{\text{visibility},i} \\ i = 1, 2, ..., N\end{gathered} \tag{8}$$

In this expression, $0 \leq w_i \leq 1$ and $\sum_i w_i = 1$.

1. $J_{\text{proximity},i}(\mathbf{X}, \mathbf{X}_h, \mathbf{X}_t)$

The Euclidean distance that must be traveled by the host aircraft to obtain the required threat visibility can be penalized by defining:

$$J_{\text{proximity},i}(\mathbf{X}, \mathbf{X}_h, \mathbf{X}_t) = \|\mathbf{X} - \mathbf{X}_h\|_2 \tag{9}$$

When $w_j = 0$ for $j = 2, 3, 4$, minimizing the cost in (8) is equivalent to navigating to the nearest point on the surface of the the nearest elliptic cone, i.e. the shortest distance required to obtain improved threat visibility.

2. $J_{\text{robust},i}(\mathbf{X}, \mathbf{X}_t)$

The size of the elliptic cone's cross section can be seen as a measurement of robustness to disturbances in the host's position as well as uncertainties in $\mathbf{X}_t$ and λ; a host within a large cross-section is less likely to exit the volume and lose threat visibility. For an elliptic cone, this measurement can be expressed as the radius of a circle with area equivalent to the elliptic cross-section. Higher values are desirable, so the robustness cost can be expressed as the negative radius:

$$J_{\text{robust},i}(\mathbf{X}, \mathbf{X}_t) = -|v|\sqrt{ab} \tag{10}$$

Scalar v is dependent on $\mathbf{X}$ and $\mathbf{X}_t$, obtained from the parameterization used in (7), while a and b are once again the semi-major and semi-minor axes of the respective background ellipse. Increased $|v|$ correlates with increased distance between the host and threat, so minimizing a weighted combination of $J_{\text{proximity},i}(\mathbf{X}, \mathbf{X}_h, \mathbf{X}_t)$ and $J_{\text{robust},i}(\mathbf{X}_h, \mathbf{X}_t)$ involves compromising between the competing proximity and robustness objectives. Note that this negative robustness term is unbounded below, so there must always be a competing proximity objective.

It should also be noted that robustness is direction dependent; ellipses with very high eccentricities could have much greater robustness along the semi-major axis compared to the semi-minor axis. The quantity $\sqrt{ab}$ does not take eccentricity into account, but one could include an eccentricity term to factor in directional robustness.

3. $J_{\text{rate},i}(\mathbf{X}, \mathbf{X}_t, \mathbf{v}_t)$

The elliptic cones are dynamic since their apex, the threat, is moving at some velocity. A rate term is included to account for the fact that the host must move with a given elliptic cone in order to maintain threat visibility. Assuming that the background stays relatively invariant with respect to the host motion requires that host's range to background λ is much greater than the host's range to the threat. When this assumption holds, the approximation in (2) can be expressed in terms of the defined problem variables to produce the penalty:

$$\begin{aligned} J_{\text{rate},i}(\mathbf{X}, \mathbf{X}_t, \mathbf{v}_t) &= \frac{\lambda}{\lambda - \|\mathbf{X} - \mathbf{X}_t\|} \mathbf{v}_t \sin\phi \\ \phi &= \arccos\left(R\hat{e}_3 \cdot \frac{\mathbf{v}_t}{\|\mathbf{v}_t\|}\right) \end{aligned} \tag{11}$$

The term $\lambda/(\lambda - \|\mathbf{X} - \mathbf{X}_t\|)$ corresponds to the ratio $\frac{d_h}{d_t}$ in (3), while ϕ corresponds to the angle between the camera's line of sight vector, $R\hat{e}_3$, and the threat velocity vector.

Although the rate term will increase as the host's distance to threat increases, $\lambda/(\lambda - \|\mathbf{X} - \mathbf{X}_t\|)$ will remain relatively close to 1, meaning that the term remains relatively constant for ellipses of equal background depth λ. This term becomes much more important, however, if extending to more complex situations where the background is of variable depth, especially if the near-infinite background assumption is violated. In these situations, the ratio $\lambda/(\lambda - \|\mathbf{X} - \mathbf{X}_t\|)$ can be much greater than 1, which means that the rate cost term will help enforce the velocity limits of the host aircraft.

4. $J_{\text{visibility},i}$

The final term, the visibility term, is invariant with respect to the cost function arguments. Instead, it is intrinsic to the conic volume being analyzed, so the value will vary amongst the N elliptic cones. The visibility cost is related to the visibility metric applied to the relevant elliptic region, where visibility metric is context-dependent, as discussed in Section III.A. In this simple grayscale example, $J_{\text{visibility,i}}$ is defined as the negative of the average grayscale intensity of the elliptic region, which varies from 0 to 255.

E. Accounting for Threat Dynamics

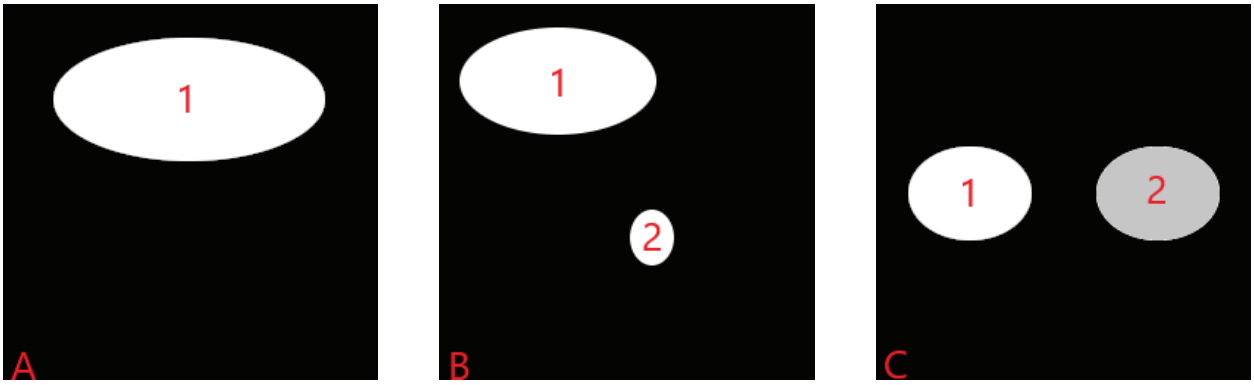
The constraint $\mathbf{X} \in \mathcal{E}_{\text{cone}}$ has been derived with respect to a static threat. Extension to a dynamic threat is straightforward; the host extrapolates the future position of the threat for some time step ΔT to compute where the elliptic volume will be at a future time. For this constant velocity abstraction, a Kalman filter with constant velocity model is sufficient. For more complex threat dynamics models, viable trajectory generation strategies are summarized by Li et al [20]. For a fixed-wing threat, the threat's pose data can also be considered to improve trajectory prediction [21]. In this implementation, the host can solve a similar nonlinear optimization problem, where the i^{th} cone's modified constraint is $\mathbf{X} \in \mathcal{E}_{\text{cone}_i,\text{predict}}$, which uses the predicted threat position as apex, rather than the current threat position; $\mathbf{X}_t$ in (7) is replaced by $(\mathbf{X}_t + \Delta T \mathbf{v}_t)$:

$$\begin{aligned} \mathcal{E}_{\text{cone}_i,\text{predict}} &= \{(\mathbf{X}_t + \Delta T \mathbf{v}_t) + v(\mathbf{X}_{\text{ell}} - (\mathbf{X}_t + \Delta T \mathbf{v}_t)) | \mathbf{X}_{\text{ell}} \in \mathcal{E}_{\text{ell,world}} \subset \mathbb{R}^3,\ v \in \mathbb{R}\} \\ \mathbf{X}_t &= \begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix}, \quad \mathbf{X}_{\text{ell}} = \begin{bmatrix} X_{\text{ell}} \\ Y_{\text{ell}} \\ \lambda \end{bmatrix}, \quad v \leq 0 \end{aligned} \tag{12}$$

Note that this optimization process assumes the host is currently outside an elliptic cone. If the host is currently within a cone, its modified control objective is to navigate to the cone's axis of symmetry. The axis of symmetry of a elliptic cone is desirable because it increases robustness with respect to uncertainty of the exact conic boundary, due to uncertainties in background range estimate λ and threat position $\mathbf{X}_t$.

IV. Examples

To illustrate the trade-offs that exist with varying objective function weights in different scenarios, optimization examples were constructed in MATLAB for (256 x 256) resolution scenes, shown in Figure 5. **Scene A** consists of a single white ellipse centered near the top of the image plane. **Scene B** consists of two white ellipses, where Ellipse 1 is larger but further from the center of the image plane, while Ellipse 2 is smaller but more centered. **Scene C** consists of



(a) Binary scene with a single elliptic region of visibility, centered near the top of the image plane.

(b) Binary scene with two elliptic regions of visibility that vary by shape and position.

(c) Grayscale scene with two elliptic regions of visibility that vary by grayscale intensity.

Fig. 5 Scenes used in MATLAB Examples with Labeled Ellipses

two ellipses that are symmetric about the image plane, but vary by grayscale intensity; Ellipse 1 is white while Ellipse 2 is gray.

The geometry of the example scenarios is visualized in Figure 6. A host at the origin of a right-handed world coordinate frame in $\mathbb{R}^3$ is looking at a threat at $\mathbb{X}_{\approx} = (0, 0, 50)$, moving at velocity $\mathbf{v}_t$ against a background of range $\lambda = 1000$. The parameters of the virtual camera are presented in Table 2:

Table 2 Virtual Camera Parameters

T	**R**	λ	f	s_x	s_y	s_θ	o_x	o_y
$\begin{bmatrix}0\\0\\0\end{bmatrix}$	$\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}$	1000	1	50	50	0	128	128

For each scene, Canny edge detection was performed and ellipses were detected using a 1D Hough transform [22]. The properties of the detected ellipses are displayed in Table 3:

Table 3 Properties of Detected Ellipses

Scene	**Ellipse #**	x_c (pixels)	y_c (pixels)	a (pixels)	b (pixels)	γ (degrees)	**Intensity** (0-255)
A	1	128	66	92	42	0	255
B	1	82	53	67	37	0	255
B	2	146	160	19	15	-85	255
C	1	64	130	42	32	0	255
C	2	192	130	42	32	0	199

The nonlinear cost function defined in (8) was minimized for each elliptic cone in a given scene using MATLAB's nonlinear solver `fmincon`, subject to the constraint that $\mathbf{X} \in \mathcal{E}_{\text{cone}_i,\text{predict}}$. To make the effects of varying weights w_i more intuitive, the cost terms were normalized:

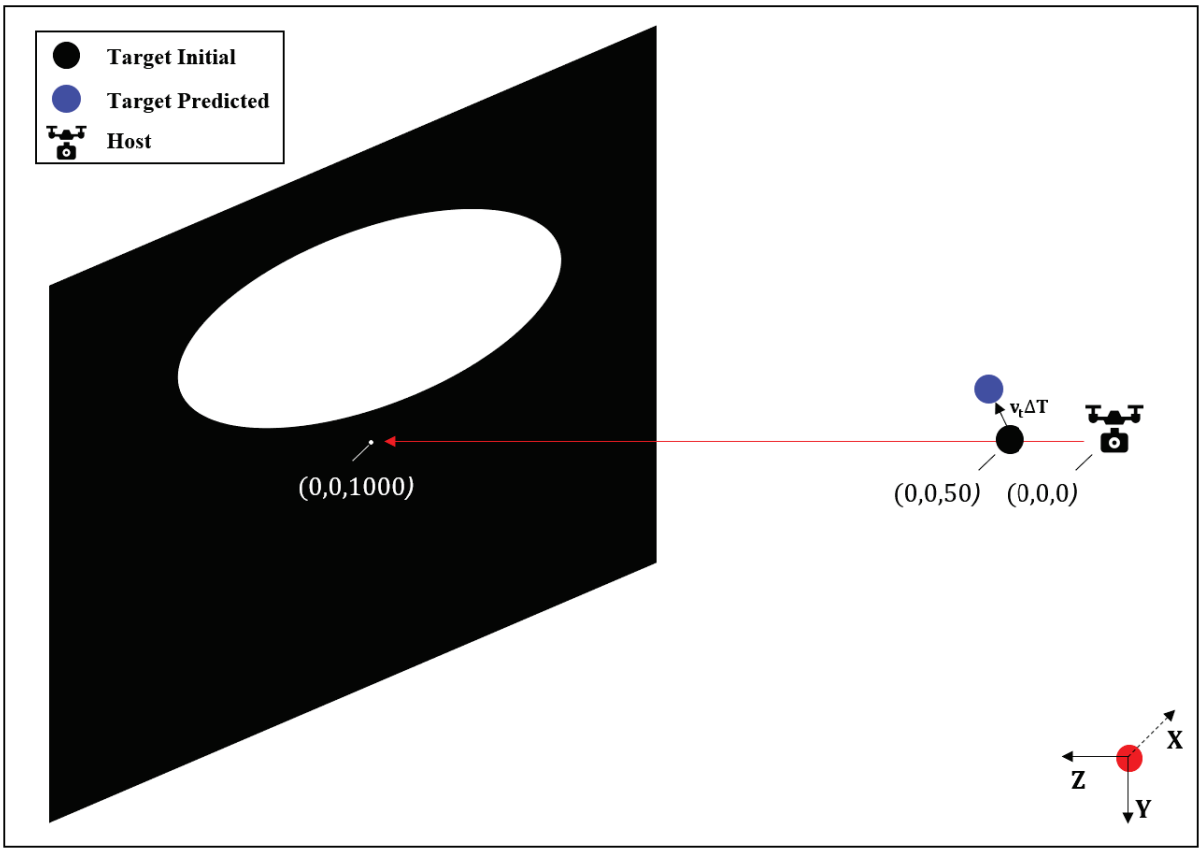


Fig. 6 Example Problem Setup (shown for Scene A). A host with $\mathbf{X}_h = (0, 0, 0)$ is looking at a threat centered at $\mathbf{X}_t = (0, 0, 50)$ against a grayscale scene of range $\lambda = 1000$. The threat's predicted translation over time step ΔT is defined by vector $\mathbf{v}_t \Delta T$. The host's line of sight is along the positive Z axis, with an image plane aligned with the XY plane. The problem is to find a host position that yields improved threat visibility.

$$
\begin{aligned}
\hat{J}_{\text{proximity},i}(\mathbf{X}, \mathbf{X}_h, \mathbf{X}_t) &= \frac{J_{\text{proximity},i}(\mathbf{X}, \mathbf{X}_h, \mathbf{X}_t)}{d_{\text{ref}}}, \quad d_{\text{ref}} = 50 \\
\hat{J}_{\text{robust},i}(\mathbf{X}, \mathbf{X}_t) &= \frac{J_{\text{robust},i}(\mathbf{X}, \mathbf{X}_t) + r_{\text{ref}}}{r_{\text{ref}}}, \quad r_{\text{ref}} = 100 \\
\hat{J}_{\text{rate},i}(\mathbf{X}, \mathbf{X}_t, \mathbf{v}_t) &= \frac{J_{\text{rate},i}(\mathbf{X}, \mathbf{X}_t, \mathbf{v}_t)}{v_{\text{ref}}}, \quad v_{\text{ref}} = 20 \\
\hat{J}_{\text{visibility},i} &= \frac{J_{\text{visibility},i} + I_{\text{ref}}}{I_{\text{ref}}}, \quad I_{\text{ref}} = 255
\end{aligned}
\tag{13}
$$

The scalar reference values d_{ref}, and v_{ref} are the distance and velocity values that map to a cost of 1 for the proximity and rate terms, respectively. The scalar reference values r_{ref}, and I_{ref} are the conic radius and pixel intensity values that map to a cost of 0 for the robustness and visibility terms, respectively. Each normalization is a simple affine transformation that maps the respective cost term to a value generally between 0 and 1. In a real implementation, the normalization process would be part of the weight tuning process, where the scaling factors would be determined via calibration and absorbed into the weights $\{w_j\}_{j=1}^4$. Any constant offsets would have no effect on the optimization process.

Table 4 summarizes the threat initial velocity $\mathbf{v}_t$, time step ΔT and weights $\{w_j\}_{j=1}^4$ used for each example scenario. It also includes the returned costs $\{J_i\}_{i=1}^2$, with lower cost highlighted, and optimal host positions $\{\mathbf{X_i}\}_{i=1}^2$ for each elliptic cone in the example's corresponding background:

Table 4 Example Scenario Parameters and Results

		Threat Motion		Weights				Costs		Optimal Host Positions					
Ex #	**Scene**	$\mathbf{v}_t$	ΔT	w_1	w_2	w_3	w_4	J_1	J_2	X_1	Y_1	Z_1	X_2	Y_2	Z_2
I	A	[0, 0, 2]	1	0.5	0.5	0	0	0.378	N/A	0.1	25.1	-6.1		N/A	
II	A	[0, 0, 2]	1	0.4	0.6	0	0	0.326	N/A	0.1	36.1	-31.6		N/A	
III	B	[0, 0, 2]	1	0.25	0.25	0.25	0.25	0.344	0.302	10.9	34.4	14.0	-10.2	-15.4	3.7
IV	B	[0, 0, 2]	1	0.15	0.35	0.25	0.25	0.300	0.346	22.1	69.6	-25.0	-11.1	-16.7	-0.6
V	B	[0, -5, 2]	1	0.25	0.25	0.25	0.25	0.383	0.387	9.9	29.6	11.9	-10.5	-19.1	4.5
VI	C	[0, 0, 2]	1	0.25	0.25	0.25	0.25	0.270	0.324	23.3	-0.5	1.8	-23.3	-0.5	1.8
VII	C	[5, 0, 2]	1	0.25	0.25	0.25	0.25	0.362	0.364	28.3	-0.5	2.4	-18.3	-0.5	1.3
VIII	C	[5, 0, 2]	1	0.3	0.3	0.3	0.1	0.435	0.393	28.3	-0.5	2.4	-18.3	-0.5	1.3

Scene A represents an extremely common real-world occurrence, where the top portion of the image plane yields good visibility (e.g. a skyline). **Ex I** considers a threat moving away from the host, with a prediction step ΔT of 1 second. The solution shows that the closest trajectory that will yield improved robust visibility is mainly along the positive Y axis (i.e. by causing the host aircraft to descend). **Ex II** considers the same scenario, except the robustness term is prioritized more heavily than the proximity term. In this case, the host aircraft is made to descend and also move away from the threat along the negative Z axis. For both **Ex I** and **Ex II**, rate and visibility terms have been ignored, since there is no competing ellipse. Furthermore, the visibility cost is static in this formulation, rather than a function of distance from the threat. In a real scenario, the robustness and visibility terms describe the inherent trade-off between robust visibility and pixels-on-target that all digital imaging systems face, so their relative weights would have to be tuned accordingly.

The next three examples consider **Scene B**, where one of two ellipses of differing size and location must be selected. In **Ex III**, the host must move in the positive X and positive Y directions to enter the elliptic cone defined by Ellipse 1 or move in the negative X and Y directions to enter the elliptic cone defined by Ellipse 2. When all weights are equal, the closer, smaller ellipse, Ellipse 2, is selected, due to its lower associated cost. However, when robustness is slightly more prioritized, as in **Ex IV**, the lower cost is associated with Ellipse 1, due to its larger size and corresponding larger conic cross-section. Furthermore, when the threat is given a velocity component in the negative Y direction (ascending), as in **Ex V**, the equally distributed weights actually select Ellipse 1 as the better option. This is because the algorithm predicts that the threat's Y velocity will move the larger elliptic cone towards the host.

The final three examples consider **Scene C**, where the elliptic backgrounds yield different levels of visibility. In **Ex VI**, weights are all equal, which results in selecting the whiter ellipse, Ellipse 1. Note that the optimal host perspectives, $\mathbf{X_1}$ and $\mathbf{X_2}$ are symmetric about the X axis for this example, as expected. **Ex VII** is identical to the previous example, except the threat is given velocity along the X axis. This results in nearly identical cost between Ellipse 1 and Ellipse 2; any additional velocity in the X direction will cause the lower-visibility ellipse to be selected. When the same conditions are repeated, but visibility weight is reduced, as in **Ex VIII**, the lower-visibility ellipse is selected.

V. Conclusion and Future Considerations

This paper hypothesizes a method to determine candidate perspectives and plan host motion that will produce and maintain these perspectives, in order to improve threat detection. The approach models acceptable threat to background perspectives as three-dimensional conic volumes and suggests minimizing a nonlinear cost function over a short time horizon to find the optimal host position. A baseline problem abstraction of tracking a mobile sphere is presented with appropriate assumptions, to validate the methodology.

There are many logical extensions of this approach. For example, the scene could be made more complex by considering backgrounds of different colors and geometries. The threat motion could extended to random speeds and directions, or even antagonistic motion (e.g. intentional camouflage). The three-dimensionality of the background could be considered by relaxing the assumption of constant depth background, as mentioned in the discussion about $J_{\text{rate},i}(\mathbf{X}, \mathbf{X}_t, \mathbf{v}_t)$. This change would require a method (likely non-visual, given the limitations of visual ranging methods) to map the ranges of different background areas, but this would increase the applicability of the algorithm to scenarios with more complex background geometry, such as urban areas, mountainous regions, or near tree lines. The algorithm could be adapted to consider a multi-host scenario, where the goal is to collectively maximize coverage of the

threat. Hosts could coordinate to assign regions of coverage to one another, to ensure that at least one host always has a suitable view of the threat.

One important extension is to consider host performance limitations by implementing this approach in a simulation environment. This approach was designed with MPC in mind, since controls are selected to optimize some cost over a finite time horizon. The optimal position $\mathbf{X}$ could then be tied to an actual trajectory with associated cost that considers the flight modes of the aircraft (e.g. cost to ascend vs. cost for lateral movement). AirSim [23] is a viable tool for testing, since it is photorealistic with a powerful physics engine and library of available sensors. Photorealism allows one to better tie the visibility cost term to a more complex visibility metric or actual detection score. Furthermore, AirSim supports integration with ROS [24], so field-ready controllers and algorithms can be implemented.

References

[1] Hu, S., Goldman, G. H., and Borel-Donohue, C. C., "Detection of unmanned aerial vehicles using a visible camera system," *Applied Optics*, Vol. 56, No. 3, 2017, p. B214. https://doi.org/10.1364/ao.56.00b214, URL https://doi.org/10.1364/AO.56.00B214.

[2] Sevil, H. E., Dogan, A., Subbarao, K., and Huff, B., "Evaluation of extant computer vision techniques for detecting intruder sUAS," *2017 International Conference on Unmanned Aircraft Systems, ICUAS 2017*, 2017, pp. 929–938. https://doi.org/10.1109/ICUAS.2017.7991373.

[3] Lie, W. N., Lin, T. C., Lin, T. C., and Hung, K. S., "A robust dynamic programming algorithm to extract skyline in images for navigation," *Pattern Recognition Letters*, Vol. 26, No. 2, 2005, pp. 221–230. https://doi.org/10.1016/j.patrec.2004.08.021.

[4] Yazdi, M., and Bouwmans, T., "New trends on moving object detection in video images captured by a moving camera: A survey," , 5 2018. https://doi.org/10.1016/j.cosrev.2018.03.001.

[5] LaValle, S. M., Gonzalez-Banos, H. H., Becker, C., and Latombe, J. C., "Motion strategies for maintaining visibility of a moving target," *Proceedings - IEEE International Conference on Robotics and Automation*, Vol. 1, IEEE, 1997, pp. 731–736. https://doi.org/10.1109/robot.1997.620122.

[6] Cancemi, L., Innocenti, M., and Pollini, L., "Guidance augmentation for improved target visibility," *AIAA Guidance, Navigation, and Control Conference*, 2014. https://doi.org/10.2514/6.2014-1477, URL http://arc.aiaa.org.

[7] Bangura, M., and Mahony, R., "Real-time model predictive control for quadrotors," *IFAC Proceedings Volumes (IFAC-PapersOnline)*, Vol. 19, IFAC Secretariat, 2014, pp. 11773–11780. https://doi.org/10.3182/20140824-6-za-1003.00203.

[8] Nageli, T., Alonso-Mora, J., Domahidi, A., Rus, D., and Hilliges, O., "Real-time motion planning for aerial videography with real-time with dynamic obstacle avoidance and viewpoint optimization," *IEEE Robotics and Automation Letters*, Vol. 2, No. 3, 2017, pp. 1696–1703. https://doi.org/10.1109/LRA.2017.2665693.

[9] Falanga, D., Foehn, P., Lu, P., and Scaramuzza, D., "PAMPC: Perception-Aware Model Predictive Control for Quadrotors," *IEEE International Conference on Intelligent Robots and Systems*, Institute of Electrical and Electronics Engineers Inc., 2018, pp. 5200–5207. https://doi.org/10.1109/IROS.2018.8593739.

[10] Witus, G., and Hunt, S., "Monocular visual ranging," *Unmanned Systems Technology X*, Vol. 6962, 2008, p. 696204. https://doi.org/10.1117/12.778686, URL https://www.spiedigitallibrary.org/terms-of-use.

[11] Lv, X. Z., Wang, M. T., Qi, Y. F., Zhao, X. M., and Dong, H., "Research on ranging method based on binocular stereo vision," *Advanced Materials Research*, Vol. 945-949, Trans Tech Publications Ltd, 2014, pp. 2075–2081. https://doi.org/10.4028/www.scientific.net/AMR.945-949.2075, URL www.scientific.net.

[12] Benyamin, M., and Goldman, G. H., "Acoustic detection and tracking of a Class I UAS with a small tetrahedral microphone array," Tech. Rep. September, 2014. URL http://www.arl.army.mil/arlreports/2014/ARL-TR-7086.pdf.

[13] Fasano, G., Accardo, D., Tirri, A. E., Moccia, A., and De Lellis, E., "Radar/electro-optical data fusion for non-cooperative UAS sense and avoid," *Aerospace Science and Technology*, Vol. 46, 2015, pp. 436–450. https://doi.org/10.1016/j.ast.2015.08.010.

[14] Kang, C., Chaudhry, H., Woolsey, C. A., and Kochersberger, K., "Development of a Peripheral–Central Vision System for Small Unmanned Aircraft Tracking," *Journal of Aerospace Information Systems*, Vol. 18, No. 9, 2021, pp. 645–658. https://doi.org/10.2514/1.i010909.

[15] Rohaly, A. M., Ahumada, A. J., and Watson, A. B., "Object detection in natural backgrounds predicted by discrimination performance and models," *Vision Research*, Vol. 37, No. 23, 1997, pp. 3225–3235. https://doi.org/10.1016/S0042-6989(97)00156-9.

[16] Mantiuk, R., Joong Kim, K., Rempel, A. G., and Heidrich, W., "HDR-VDP-2: A calibrated visual metric for visibility and quality predictions in all luminance conditions," *ACM Transactions on Graphics*, Vol. 30, No. 4, 2011. https://doi.org/10.1145/1964921.1964935, URL http://doi.acm.org/10.1145/1964921.1964935.

[17] Watson, A., Ramirez, C. V., and Salud, E., "Predicting visibility of aircraft," *PLoS ONE*, Vol. 4, No. 5, 2009. https://doi.org/10.1371/journal.pone.0005594.

[18] Peters, R., and Strickland, R., "Image complexity metrics for automatic target recognizers," Tech. rep., 1990. URL http://pdf.aminer.org/000/320/072/image_metrics_for_clutter_characterization.pdf.

[19] Ma, Y., Soatto, S., Kosecka, J., and Sastry, S. S., *An Invitation to 3-D Vision: From Images to Geometric Models*, SpringerVerlag, 2003.

[20] Li, X. R., and Jilkov, V. P., "Survey of Maneuvering Target Tracking. Part I: Dynamic Models," , 10 2003. https://doi.org/10.1109/TAES.2003.1261132.

[21] Kang, C., Davis, J., Woolsey, C. A., and Choi, S., *Sense and avoid based on visual pose estimation for small UAS*, Vol. 2017-Septe, 2017. https://doi.org/10.1109/IROS.2017.8206188.

[22] Simonovsky, M., "Ellipse Detection Using 1D Hough Transform," , 2021. URL https://www.mathworks.com/matlabcentral/fileexchange/33970-ellipse-detection-using-1d-hough-transform.

[23] Shah, S., Dey, D., Lovett, C., and Kapoor, A., "AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles," Springer, Cham, 2018, pp. 621–635. https://doi.org/10.1007/978-3-319-67361-5{_}40, URL https://doi.org/10.1007/978-3-319-67361-5_40.

[24] Foundation, O. S. R., "Robotic Operating system website," , 2015. URL http://www.ros.org.