

Mobile Manipulation–based Deployment of Micro Aerial Robot Scouts through Constricted Aperture-like Ingress Points

Prateek Arora and Christos Papachristos

Abstract—This paper presents a novel strategy for the autonomous deployment of Micro Aerial Vehicle scouts through constricted aperture-like ingress points, by narrowly fitting and launching them with a high-precision Mobile Manipulation robot. A significant problem during exploration and reconnaissance into highly unstructured environments, such as indoor collapsed ones, is the encountering of impassable areas due to their constricted and rigid nature. We propose that a heterogeneous robotic system-of-systems armed with manipulation capabilities while also ferrying a fleet of micro-sized aerial agents, can deploy the latter through constricted apertures that marginally fit them in size, thus allowing them to act as scouts and resume the reconnaissance mission. This work’s contribution is twofold: first, it proposes active-vision based aperture detection to locate candidate ingress points and a hierarchical search-based aperture profile analysis to position a MAV’s body through them, and secondly it presents and experimentally demonstrates the novelty of a system-of-systems approach which leverages mobile manipulation to deploy other robots which are otherwise incapable of entering through extremely narrow openings.

I. INTRODUCTION

Advancing the real-world potential of autonomous robotic systems by leveraging heterogeneity [1] is an increasingly pursued area of research. As standalone system classes, Mobile Manipulation Systems (MMS) have been extensively deployed in exploration, construction, and inspection [2–5], while Micro Aerial Vehicles (MAVs) have been utilized for search and rescue, surveillance, industrial inspection, and broader commercial activities [6–11]. Exploration of unknown and unstructured environments remains a core mission objective with numerous algorithms developed [12–16]. However their focus remains on finding exploratory paths which follow collision-free routes achievable by a single agent –the exploring one– therefore disregarding edge cases where an agent may only fit through an opening safely with the help of a second robot. To this end, we argue that a heterogeneous system-of-systems robotic platform can gain entry into otherwise inaccessible areas, by autonomously detecting such marginal ingress opportunities (gaps or *apertures*) in an unstructured / partially-collapsed environment, and deploy a different class of miniature robot through them by leveraging high-precision mobile manipulation.

Motivated by the above, in this work we focus on the problem of autonomously deploying MAVs through a constrained



Fig. 1. Autonomous heterogeneous robotic system exploration into a partially-collapsed setting, by detecting *aperture*-like ingress points, inserting two micro-sized MAV scouts, and conducting visually-servoed exploration through remote guidance by the MMS. Corresponding video available at: <https://www.youtube.com/watch?v=Avt0Gnv3CAI>

arbitrarily-shaped aperture via precise mobile manipulation for the purpose of continued exploration. First, we propose an active vision based approach to detect and characterize any marginal apertures which may lend themselves as ingress opportunities. We propose an exhaustive search-based aperture profile analysis pipeline, to ascertain poses that can lead to a collision-free insertion of the MAV. Subsequently, we plan for robot arm trajectories to position and “slip” it through the aperture, and eventually launch it. Finally, we demonstrate using the MAV as a micro-scout, with the MMS processing the wirelessly relayed visual imagery to derive monocular depth-map estimates and remotely guide the MAV to perform visually-servoed exploration. The presented work is experimentally verified on real world mock-up environment illustrated in Figure 1 using an autonomous heterogeneous robotic system comprising a MMS ferrying two mounted micro-scout MAVs.

The rest of this paper is organized as follows: Section II presents related work, followed by the problem statement in III. The proposed approach is detailed in Section IV, while evaluation is presented in Section V. Finally, conclusions are drawn in Section VI.

II. RELATED WORK

MAV safe navigation in complex indoor environments through obstacle avoidance and collision-free planning [17–20] has been widely studied by the robotics community. Aggressive control accompanied by advanced estimation and trajectory planning strategies have been demonstrated to manage flying through gaps [21–24], by relying on monocu-

This material is based upon work supported by the NSF Award: AWD-01-00002751: RI: Small: Learning Resilient Autonomous Flight. The presented content and ideas are solely those of the authors.

The authors are with the Robotic Workers (RoboWork) Lab, University of Nevada, Reno, 1664 N. Virginia, 89557, Reno, NV, USA prateeka@nevada.unr.edu

lar visual/inertial information. In most works, the gaps have a priori known convex-shaped structures, while certain more recent works have come to address flight through structure-less gaps [25]. All the aforementioned efforts however consider openings with relatively large clearances (5 [cm] in [25]), to allow for perception uncertainty, and more importantly for reasonable tolerances in control error tracking during real-world MAV flight. In this work, we focus on deploying miniaturized MAVs into the tightest possible apertures, by changing the common paradigm and considering fitting them through, while their propellers are not spinning such that there is no risk to their structural integrity. This additionally renders the use of propeller guards / protective cages / etc. irrelevant, thus keeping the required MAV-scout body profile at a minimum. Our proposed strategy first detects *aperture* proposals for the micro-scout's deployment, followed by exhaustive search algorithm of a candidate's profile to fit the MAV. This caters to a generic, broad category of gap-like openings without requiring prior knowledge of a specific profile shape and/or size. The pipeline is able to obtain collision-free poses that can marginally "slip" the MAV body profile through the gap at tight clearances. To achieve the required high-precision motion, we leverage a MMS which ferries the MAV and is capable to perform autonomous motion planning to a) retrieve it from its body, b) exactly position it to fit through the *aperture* profile, c) level it to facilitate safe launching, and d) remotely guide it past the aperture to achieve continued exploration.

III. PROBLEM SETUP

The problem of exploration of particularly constricted environments—such as partially collapsed sections of buildings after catastrophic events—is considered within this work along the lines of autonomous deployment of heterogeneous robotic systems. More specifically, we aim to address such constraints that cannot be overcome by the standalone use of any one robotic class, but instead require the intelligent combination of the unique capabilities offered by each platform. Considering robotic operations within highly unstructured settings, littered with debris and presenting heavy structural damage, we distinguish two broad classes of navigation constraints, namely a) obstacles and structures that constitute significant collision volumes, and b) structural *apertures*, i.e. hole-like sections in the environment architecture. Within the proposed approach, the first class-ones are considered as part of a groundmap to be navigated while avoiding obstacles and respecting traversability conditions, while the latter class of structural *apertures* lend themselves as opportunities to gain entrance into caved-down locations that are otherwise inaccessible.

Given the existence of such ingress/egress points which can be situated at various points within an unstructured indoor environment, the natural robotic class of preference to gain access through them is the Micro Aerial Vehicle. At the same time however, this work does not consider openings that are by-design accessible (e.g. doors, windows, hatches), but highly constricted and arbitrarily shaped *apertures* that

may be the product of structural failures (wall holes, gaps in piled-up debris, etc.). As such, their narrow and irregular shape may preclude even the most aggressively maneuverable MAV from flying through them.

A novel robotic deployment approach to address this scenario is the vision presented in this work, which considers the careful "shoving" of the aerial robot body into the constrained *aperture* until it has come all the way through the other end, and then follow up with the takeoff and deployment. This determines the following required robotic modalities: a) a high-end robotic platform capable of long-term autonomous exploration and mapping, b) a versatile manipulation system to perform 6 – DoF positioning and insertion of a small rigid body into an *aperture*, and c) one-or-more MAV class robots miniaturized to the level of being able to fit through the narrowest possible opening size.

A. Definitions

The overarching mission objective is the exhaustive exploration of an unknown, unstructured location. We assume the existence of an onboard imaging-and-depth sensor $\mathbb{S}$, and denote its frame of reference as $\mathcal{F}_{\mathbb{S}}$. Within a subset bounded volume $\mathcal{V}$ of the environment, let $\mathcal{M}$ be a 3D volumetric map of it, which is incrementally constructed [26] by using the sensor $\mathbb{S}$. This is further distinguished into voxels m that in either of the occupied, free, and unknown subsets $\mathcal{M}_{occupied}$, $\mathcal{M}_{free}$, $\mathcal{M}_{unknown}$. Moreover, let $\mathcal{D}_{\mathbb{S}}$ define the 3D-depth dense pointcloud representation over a view of a subset of the environment as observed from a certain $\mathcal{F}_{\mathbb{S}}$.

We put forward the overarching definitions employed and the specific problems addressed within this work:

Definition 1 (Aperture) Given the $\mathcal{D}_{\mathbb{S}}$ dense representation, an *aperture* corresponds to a region of a planar subset of the map which is depressed by at least a 3D-depth threshold value d_C , such that a contour C_a can be detected. Additionally, given the $\mathcal{M}$ volumetric representations, the aperture region within C_a and up to a depth threshold offset d_a from the planar face corresponds to $\subset \mathcal{M}_{free}$, i.e. it is a "see-through" hole-like pocket.

Definition 2 (Launch Pose) Given a structural *aperture*, a launch pose corresponds to an Euler-angle parametrized reference pose $\xi_{MAV}^{ref} = [x, y, z, \phi, \theta, \psi]^T$ for the MAV such that the translation part lies past the aperture plane, and the orientation is bounded by $|\phi, \theta| \leq \alpha_{launch}$, where α_{launch} a roll-and-pitch constraint for the aerial robot to be able to take off safely. It is noted that if this value is too steep, an underactuated MAV will risk collision with the environment during aggressive takeoff thrusting.

Then the problems addressed are given as follows:

Problem 1 (Volumetric Exploration) It is defined as the Receding Horizon-based execution of motion plans which maximize the cumulative expected observation of unknown environment subsets $\mathbb{M}_{unknown}$ using the $\mathbb{S}$ depth sensor model. The concept is addressed in relevant works [12, 27–29]. In this paper, a relevant policy is executed, but assumed

to (almost) immediately reach the maximum-possible exploration gain, due to some intentionally imposed environment constraints. This problem is mentioned for completeness.

Problem 2 (Visually-Servoed Exploration) Defined as a reduced-order exploration problem, considering map-free [30, 31] robotic operation. It pertains to a reduced-modality sensor $\mathbb{S}_V$ (imaging-only) visiting uncharted locations of an unknown environment guided by a visual-servoing policy, and acquiring visual imagery. According to this paradigm, the goal is to inspect certain regions of the map, without providing consistent and comprehensive mapping, by making use of a limited-resources miniaturized robot.

Problem 3 (Constricted Aperture MAV Deployment) Defined as the problem of “fitting” the body of a miniaturized MAV through an *aperture* which has been selected for its potential to be an ingress point into an otherwise occluded part of the environment. More specifically, we focus on proposing the apparatus and algorithm to find a viable *launch pose* –as per the above definition– that can be achieved via 6 – DoF manipulation of the MAV held by a mobile robotic arm.

B. Heterogeneous Robotic System Setup

The proposed heterogeneous robotic system is closely tied to the aforementioned problem formulation. It is therefore presented as part of the overarching problem setup.

At the core lies a Mobile Manipulation System [32] comprising a differential-drive 4-wheeled base and a 6 – DoF arm and parallel gripper, while also incorporating LiDAR and camera based perception systems (mounted in eye-in-hand configuration) allowing it to flexibly scan the environment additionally to its inherent manipulation capabilities. Based on established open-source works and products of own prior contributions, it is armed with multi-modal Simultaneous Localization and Mapping, volumetric reconstruction, exploration, Model Predictive Control, and manipulation motion planning pipelines [27, 33–36]. It is mentioned that it relies exclusively on on-board perception (no external motion-capture support), executed all missions fully autonomously.

The MMS additionally acts as a carrier [37] for a number of miniaturized RYZE Tello MAVs, which are mounted on launching apparatuses positioned into holster-like attachments on the main robot body. The launching apparatuses can be retrieved by the MMS by use of the arm and gripper.

The MAVs represent a class of computational resource-limited robots, which are however sized to 0.1 [m]-scale form factor making them ideal to fit through narrow openings. They support WiFi (2.4 Ghz) communication, offering electronically stabilized video streaming, and are equipped with onboard optical stabilization making them capable of executing remote velocity command-based navigation. In our proposed approach, they are envisioned as remotely piloted (by the MMS) micro-scouts, feeding back an encoded video stream to the main robot carrier which leverages its superior computational resources to calculate the *Visual-Servoed Exploration* policy –as per the above definition– that guides the MAV actions. The overall system architecture is visualized in Figure 2.

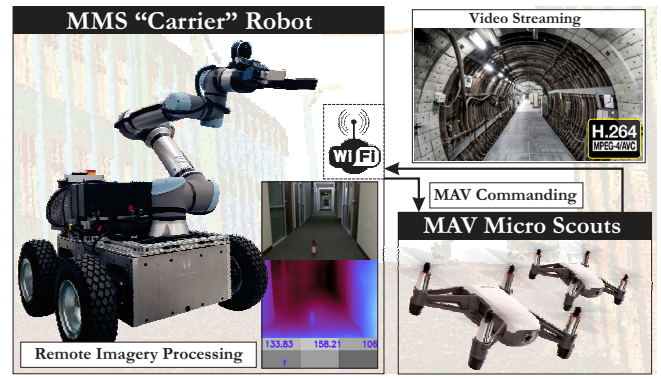


Fig. 2. The heterogeneous robotic system considered within our approach: A “carrier” MMS with significant computational resources, and a set of mounted micro-sized MAV scouts, deployed for remote visually-servoed exploration past constricted openings.

IV. PROPOSED APPROACH

This section outlines our contributions towards mobile manipulation-based micro-robotic scout deployment through constricted structural apertures.

A. Overarching Architecture

Robotic exploration missions are generally performed by operating in surroundings that are considered rigid, or present opportunities to reconfigure the environment accessibility through mobile manipulation of certain non-static entities [38, 39]. Our approach focuses on rigid environments (e.g. with large debris that cannot be cleared), which however present extremely *limited* access through very narrow openings. This is the takeoff point for our pipeline: we consider that the robot already performs autonomous exploration based on an pre-established receding horizon next-best-view approach, and we initiate at the point where no feasible exploration routes offer any further exploration gain [29], i.e. the robot is “stuck”. Figure3 presents a flowchart of this paper’s algorithmic components which are subsequently elaborated, illustrating the overarching logic of how these fundamental capabilities contributed through our work can be applied to autonomously gain entry into extremely constricted settings, by coordinating a heterogeneous robotic deployment which involves MMS and miniaturized MAV robots.

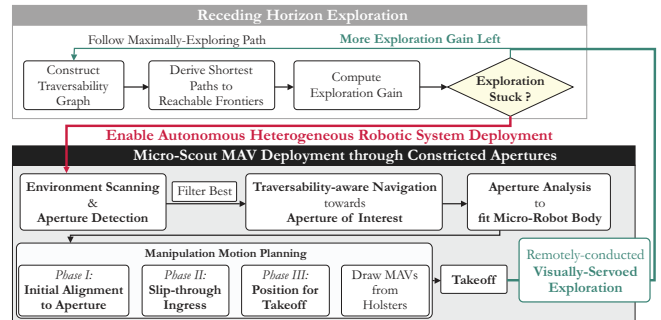


Fig. 3. Overarching architecture, indicating how the process of *aperture*-like ingress point search is autonomously triggered once the robot exploration mission is “stuck”, as well as the sequence of algorithmic steps.

B. Aperture of Interest Detection

The first module contributed in our work pertains to an efficient method to detect *apertures*, i.e. structural regions that are see-through and hollow, and at least marginally wide enough to fit the MAV in order to act as ingress points to further the robotic exploration mission.

We employ a N -frame sliding window of the dense point-cloud data $\mathcal{D}_S^N = \{\mathcal{D}_S^{\mathcal{F}_1}, \dots, \mathcal{D}_S^{\mathcal{F}_N}\}$, where all incoming data are transformed per a common frame $\mathcal{F}_\parallel$ to acquire a dense and consistent 3-D representation of the local environment manifold. $\mathcal{F}_\parallel$ is a frame instantaneously coincident by position and azimuth with $\mathcal{F}_S$, and inertially-aligned. We subsequently perform plane model segmentation, fitting a major *model plane* in the general form $\alpha x + \beta y + \gamma z = \delta$ using RANSAC, including the additional constraint of $\arccos(\hat{n} \cdot \hat{z}) \leq \theta_p$, where $\hat{n} = [\alpha \ \beta \ \gamma]^T$, $\hat{z} = [0 \ 0 \ 1]^T$ w.r.t $\mathcal{F}_\parallel$ and $\theta_p = 45^\circ$. The latter ensures that the plane doesn't correspond to the ground plane.

The dense pointcloud part that corresponds to the major *model plane* $\mathcal{D}_S^P$ is extracted and transformed to the “fronto-parallel” view of the sensor, i.e. it is rotated by $\mathbf{R}_{fp} \equiv q(-\hat{n}) \rightarrow \hat{z}$ such that $\hat{n} \cdot \hat{z} = -1$ (to represent the plane as perpendicular to the depth z -axis of $\mathcal{F}_S$). We employ pinhole camera model-based projection to acquire an equivalent image representation $\mathbb{I}$ with floating-point pixel intensity characterizing the depth value. It is highlighted important that the fronto-parallel view transformation is crucial to get a 2-D projection unaffected by skew; working with an image domain representation significantly reduces the search space. Taking a binary version $\mathbb{I}_{binary}$ of $\mathbb{I}$, we first perform dilation morphological operation to get $\mathbb{I}_{opening}$ followed by Canny to get $\mathbb{I}_{canny}$ and subsequently perform contour detection to derive a set of candidate aperture contours $\mathbf{A}_S$.

At the same time we compute the 2-D projection MAV^{2D} of the “fronto-parallel” view of the MAV mesh, and project it onto the $\mathbb{I}$ image plane using the pinhole scale factor for the *model plane* depth. From the previously detected contours $\mathbf{A}_S$ we only keep the ones which correspond to a) hollow regions, i.e. their $\mathbb{I}$ floating point depth differs from the *model plane* depth by a value larger than the d_C threshold value, b) marginally wide regions, i.e. their image area is at least larger than the MAV contour C_m area.

C. Candidate Aperture Filtering

The preceding pipeline can continuously detect candidate apertures while the eye-in-hand sensor $\mathbb{S}$ mounted on the manipulator arm performs environment scanning motions. It is expected that during this operation which is conducted at intermediate-to-longer range from the environment structures, discretization errors, sensor noise, and the simplified metric of accepting “large enough” contour areas (in terms of image moment as mentioned above) will lead to detection of inappropriate *aperture* candidates.

To allow a hierarchical focus of attention of the MMS, we operate by updating a probabilistic belief for each *aperture* centroid detection location. We consider the aforementioned volumetric environment representation $\mathcal{M}$, and for each

voxel m_i we update a log-likelihood metric L_a that corresponds to its probability p_a to be a good *aperture* candidate based on the sensor detections $z_{1:t}$ as:

$$\begin{aligned} L_a(m_i|z_{1:t}) &= L_a(m_i|z_{1:t-1}) + L_a(m_i|z_t) \\ \text{with: } L_a(m_i) &= \log\left[\frac{p_a(m_i)}{1 - p_a(m_i)}\right] \end{aligned} \quad (1)$$

, where this can be considered as equivalent to a hits-or-misses update model. It is noted that a “hit” corresponds to an *aperture* detection, while “miss” measurements are integrated for the remaining m_i within the sensor's unobstructed field of view (by performing raycasting). The overall operation successfully rejects unstable detections, while the buildup of the confidence metric $p_a(m_i)$ allows to prioritize detections with increased consistency.

D. Planning for Mobile Robot Navigation & Manipulator Arm Positioning

After acquiring the most consistent candidate *aperture* map-point location m_A , we compose the corresponding approach pose $\mathcal{P}_a = [t_a \ q_a]^T$, where $t_a \in \mathbb{R}^3$ and q_a a unit quaternion parametrization $\in SO(3)$ of the *model plane* normal vector. In order to achieve collision-free navigation of the MMS near the *aperture* and reach it via the manipulator arm end-effector, we leverage a sampling based approach supported by our prior work in connectivity-based groundmap estimation and traversability-aware wheeled robot navigation [38].

Originating from the desired setpoint $\mathcal{P}_a$ we project a cone of constant radius $\rho = 2 \ [m]$ onto the extracted groundmap plane, and randomly sample candidate poses for the mobile base within their intersection, which also qualifies as valid by a collision check query to the volumetric map $\mathcal{M}$. Then, for each surviving candidate base location b_i :

- Given the manipulator's current end-effector pose $curr\mathcal{P}_{ee} = [curr t_{ee} \ curr q_{ee}]^T$, we compute an end-effector goal pose $b_i\mathcal{P}_{ee} = [b_i t_{ee} \ b_i q_{ee}]^T$ translated by sampled mobile base location b_i and oriented to maintain sight of characterized aperture, where $b_i t_{ee} = curr t_{ee} + b_i$, $b_i q_{ee} = curr q_{ee} \cdot q_a \cdot curr q_{ee}^{-1}$
- Leverage the groundmap-connectivity graph of [38], we compute a reachable MMS motion plan that does not collide with the environment, or run into traversability constraints.
- Employ Inverse Kinematics-based collision-aware motion planning [36] to examine whether a feasible motion plan for the desired arm end-effector pose $b_i\mathcal{P}_{ee}$ exists.

Once a feasible solution is found the pipeline is terminated, the MMS base is commanded to navigate towards the goal such that the mounted arm workspace reaches the *aperture*, while the latter is also maintained within the eye-in-hand sensor $\mathbb{S}$ field-of-view.

E. Aperture Profile Analysis

This section presents our contributed approach to determine MAV launch poses by analysing the *aperture* profile. From a closer vantage point (after having approached the

aperture of interest with the MMS), we employ again the previously described pipeline in Section IV-B to get an *aperture* contour C_a in the binary image $\mathbb{I}_{binary}$ and the scaled projection of the MAV contour C_m . We proceed to perform a hierarchical exhaustive search over the corresponding 2 – d configuration space in order to find a solution where C_m lies in the interior of C_a (their boundaries remain non-intersecting):

i) We first create a set $\mathbb{G}$ of all possible point-pairs $x_i, x_j \in C_a$, $x_i \neq x_j$ with the additional constraints that $l_{x_i, x_j} \in C_a$ & $\|l_{x_i, x_j}\| \geq M_{C_m}$ i.e. the line segment l_{x_i, x_j} formed by joining x_i and x_j lies in the interior of the aperture contour C_a and euclidean norm of l_{x_i, x_j} is greater than the length of the Major axis of the MAV contour represented by M_{C_m} . Following this, we rank the elements of the set $\mathbb{G}$ by employing the following heuristic: $m_{l_{x_i, x_j}} = \min(m1, m2)$ where $m1 = w1 \cdot \|l_{x_i, x_j} - M_{C_a}\| + w2 \cdot \|\phi\angle(l_{x_i, x_j}, M_{C_a})\|$ & $m2 = w1 \cdot \|l_{x_i, x_j} - m_{C_a}\| + w2 \cdot \|\phi\angle(l_{x_i, x_j}, m_{C_a})\|$, i.e the weighted sum of the difference between length and orientation of l_{x_i, x_j} w.r.t. the Major and Minor axes of the aperture contour, denoted as M_{C_a} & m_{C_a} respectively. The set $\mathbb{G}$ at this point, contains sorted point-pairs such that the line segment l_{x_i, x_j} a) is closer in length and orientation w.r.t M_{C_a} & m_{C_a} , b) lies in the interior of C_a , and c) has Euclidean norm greater than M_{C_m} .

ii) Provided the ranked set $\mathbb{G}$ and an empty set $\mathbb{V} = \phi$, $\forall l_{x_i, x_j}; x_i, x_j \in \mathbb{G}$, we transform the MAV contour C_m such that its center of mass is placed at the midpoint of l_{x_i, x_j} , and its Major axis M_{C_m} is oriented along the line l_{x_i, x_j} , to get C_m^t . We proceed to check for intersection of the *aperture* and the transformed MAV contours C_a and C_m^t . If the contours are non-intersecting the corresponding configuration $\{p, \theta\}$, where p the mid point and θ the slope of l_{x_i, x_j} are appended to the set of valid configurations $\mathbb{V}$. Otherwise the pair $\{x_i, x_j\}$ is appended to the set of initially-failed options $\mathbb{F}$. If $\mathbb{V} \equiv \emptyset$ (no valid options found) we execute to the next step.

iii) As long as $\mathbb{V} \equiv \emptyset$, we iterate over the elements of the initially-failed set $\mathbb{F}$, performing a sliding-search along the line segments l_{x_i, x_j} . More specifically, with the Major axis of C_m oriented along the line l_{x_i, x_j} and the center of mass initially positioned at the midpoint, we keep shifting the C_m along the line l_{x_i, x_j} by one pixel-unit, moving closer towards one of the end points of l , and until the edge of the C_m coincides with that end point. This happens interchangeably; one step-shift on the left is followed by a one-step shift on the right, and vice versa. During each iteration we check for contour C_a , C_m^t intersection, and if a valid configuration $[p', \theta]$ is found, where p' the left-or-right-shifted center of mass, it is appended into the valid set $\mathbb{V}$.

Algorithm 1 summarizes this process, and Figure 4 presents relevant illustrations. Additionally, Table I provides average execution times according to different *aperture* shape classes. It is mentioned that $\mathbb{V}$ is upper-bound by N_v , returning as soon as a sufficient number of candidate configurations are found.

Algorithm 1 Aperture profile analysis

Inputs:

Depth sensor pointcloud sliding-window $\mathcal{D}_S^N$

$C_m \leftarrow \text{GetContour}(\text{MAV}_{proj2D}^s)$

Output: Valid & collision-free launch poses: ξ_{MAV}

```

1: \* Part 1: Aperture Characterization *
2:  $\mathcal{D}_S^N \leftarrow \text{TransformToFrame}(\mathcal{F}_I)$ 
3:  $[\alpha \beta \gamma \delta] \leftarrow \text{RANSAC}(\mathcal{D}_S^N, \theta_p) \triangleright$  model plane parameters
4:  $\mathcal{D}_S^P \leftarrow \text{ExtractInliers}(\mathcal{D}_S^N, [\alpha \beta \gamma \delta])$ 
5:  $\mathcal{D}_{fp}^P \leftarrow \text{FrontoParallelTransform}(\mathcal{D}_S^P)$ 
6:  $\mathbb{I} \leftarrow \text{ProjectToImagePlane}(\mathcal{D}_{fp}^P)$ 
7:  $\mathbb{I}_{binary} \leftarrow \text{Threshold}(\mathbb{I})$ 
8:  $C_a \leftarrow \text{GetIngressApertureContour}(\mathbb{I})$ 
9: \* Part 2: Aperture profile analysis *
10:  $\mathbb{G} \leftarrow \{\phi\}$   $\triangleright$  set of candidate pair points
11:  $M_{C_a}, m_{C_a} \leftarrow \text{GetMajorMinorAxis}(C_a)$ 
12:  $M_{C_m}, m_{C_m} \leftarrow \text{GetMajorMinorAxis}(C_m)$ 
13: for all  $l_{x_i, x_j}; x_i, x_j \in C_a, x_i \neq x_j$  do
14:   if  $l_{x_i, x_j} \in C_a$  &  $\|l_{x_i, x_j}\| \geq M_{C_m}$  then
15:      $\mathbb{G} \leftarrow \mathbb{G} + \{x_i, x_j\}$ 
16:   for all  $l_{x_i, x_j}; x_i, x_j \in \mathbb{G}$  do
17:      $m1 \leftarrow w1 \cdot \|l_{x_i, x_j} - M_{C_a}\| + w2 \cdot \|\phi\angle(l_{x_i, x_j}, M_{C_a})\|$ 
18:      $m2 \leftarrow w1 \cdot \|l_{x_i, x_j} - m_{C_a}\| + w2 \cdot \|\phi\angle(l_{x_i, x_j}, m_{C_a})\|$ 
19:      $m_{l_{x_i, x_j}} \leftarrow \min(m1, m2)$ 
20:      $\mathbb{G} \leftarrow \text{Sort}(\mathbb{G}, m_{l_{x_i, x_j}})$ 
21:  $\mathbb{V} \leftarrow \{\phi\}$ 
22:  $\mathbb{F} \leftarrow \{\phi\}$ 
23: for all  $l_{x_i, x_j}; x_i, x_j \in \mathbb{G}$  do
24:    $t = \frac{(x_i + x_j)}{2}$ 
25:    $\theta = \text{slope}(l_{x_i, x_j})$ 
26:    $\mathcal{T} = [\mathbf{R}_\theta \ t]$ 
27:    $C_m^T = \text{Transform}(C_m, \mathcal{T})$ 
28:   if  $C_a \cap C_m^T$  then
29:      $\mathbb{V} \leftarrow \mathbb{V} + \{t, \theta\}$ 
30:   else
31:      $\mathbb{F} \leftarrow \mathbb{F} + \{x_i, x_j\}$ 
32: if  $\mathbb{V} = \{\phi\}$  or  $\text{len}(\mathbb{V}) < N_{valid}$  then
33:   for all  $l_{x_i, x_j}; x_i, x_j \in \mathbb{F}$  do
34:      $\mathbb{V} \leftarrow \mathbb{V} + \text{SlidingCheck}(l_{x_i, x_j})$ 
35: return  $\mathbb{V}$ 

```

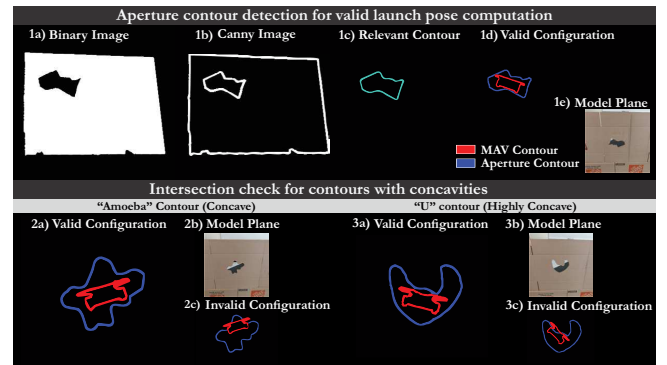


Fig. 4. Top row: 1a) $\mathbb{I}_{binary}$, 1b) $\mathbb{I}_{canny}$, 1c) Detected *aperture* contour C_a , 1d) Valid configuration solution (non-intersecting contours), 1e) Corresponding *aperture* cutout used. Bottom row – “Amoeba” and “U”-shaped examples: 2a) and 3a) Valid configuration solutions with no intersection, 2b) and 3b) Corresponding *aperture* cutouts used, 2c) and 3c) Example invalid configuration rejected during the Algorithm 1 search.

It should be highlighted that ranking the contour point-to-point line candidates by their proximity to the Major and

Minor axes of the contour shape is intended to prioritize searching for solutions assuming a *mostly convex* shape for the *aperture*, while remaining an exhaustive-search algorithm allows to uncover solutions for non-trivial shapes with higher degree of concavity. Figure 4 illustrates the process and characteristically addressed *aperture* morphologies. The valid poses in $\mathbb{V}$ are eventually transformed into 3-D “pre-launch poses” expressed in sensor frame of reference $\mathcal{F}_S$ and forwarded to the subsequent step.

TABLE I
EXECUTION TIMES OF CONTOUR FITTING PER SHAPE CLASS

Contour Shape	Average Solution Time [s]
Orthogonal (Fully Convex)	0.88
Collapsed (Light Concave)	0.97
Amoeba (Concave)	1.10
U-shaped (Highly Concave)	1.13

F. Planning for MAV Launch

In this section we detail the process involved in the motion planning for 6 – DoF manipulation of the MAV to “slip” it through the aperture, and followingly positioning it at an appropriate launch pose ξ_{MAV}^{ref} that respects the takeoff constraints as described in the previous section. To begin with, we define the “initial” pose for the end-effector frame $\mathcal{F}_{ee0}$ that corresponds to the arm configuration at the beginning of the pipeline. We proceed to plan the entire sequence for MAV deployment in a piece-wise manner. For a “pre-launch pose” ξ_{MAV} (centered within the *aperture* plane) given by the preceding pipeline, we proceed as follows:

Phase I) We examine a goal setpoint $g = d_l \cdot \hat{n} + [x \ y \ z]^T$ applying an offset of d_l of ξ_{MAV} (centered along the normal $\hat{n}$ of the *model plane*). This offset d_l represents the length of a custom 3D printed “launching apparatus” that is conveniently grasped by the gripper at one end with a supporting structure at the other end which firmly holds the MAV in place, and at the same time allows it to takeoff without significant friction. The computed motion plan begins from the “initial” pose and ends at g , where the MAV should be sitting within the launching apparatus which is held by the gripper, and centered near the *aperture* and rotated by the pose that can fit within its contour.

Phase II) At this stage, we proceed to find a collision-free ingress trajectory that gently “slips” the MAV through the aperture till the MAV egresses. With g as start state and ξ_{MAV} as end effector goal state, we compute the corresponding motion plan.

Phase III) At this last step, the MAV should have egressed from the back side of the *aperture* and held by the elongated apparatus. We proceed to plan to attain the launch pose configuration ξ_{MAV}^{ref} as per the definition 2 with the start state as ξ_{MAV} . It is highlighted that this last motion is mostly rotational in nature, since the MAV launch constraints $|\phi, \theta| \leq \alpha_{launch}$ refer to its roll and pitch during takeoff.

Following the success of all the three planning phases, we ultimately compute the required plan to grasp the MAVs which are carried by the MMS at a priori known holster-like mounting arrangements. It is important to note that

although the mounting arrangement is known, collision-free manipulation planning is executed regardless to account for the existence of nearby obstacles.

The entire sequence of trajectories computed by the motion plans is executed, starting by grasping the MAV, positioning it at the “initial” pose g , and moving it along Phases *I–III*, where the MAV is ready to takeoff based on an appropriately timed command.

G. Visual-Servoed MAV Exploration

This final section refers to the implemented policy for Visual-Servoed exploration with the MAV. The algorithmic components the facilitate this scheme are not the authors’ work, however our contribution in this part centers around the utility of a computationally-endowed MMS carrier robot as a remote-commander of highly miniaturized robotic systems that can afford minimal functionalities. In this case, the MAV is capable of transmitting visual-only imagery, which is monocular and compressed.

We propose a pipeline that leverages a trained model for monocular depth estimation [40]; the network is deployed on the MMS onboard powerful GPU and provides dense depth map estimates of the visual imagery relayed by the MAV. We divide the estimated depth image into three column subsets, and follow-up with pixel-averaging that allows us to compute a 3×1 action vector. A simple control allocation comprises of three actions: forward (f), clockwise yaw (r), counter-clockwise yaw (l), and aims to move and/or orient the MAV towards a region of greater depth.

We employ this simple policy to visually servo the MAV towards unexplored regions past the ingress point, and additionally leverage the relayed imagery to conduct automated detection of objects of interest using the [41] pipeline.

V. EVALUATION

We assessed our pipeline in an experimental study with an in-house developed autonomous Mobile Manipulation System, augmented as a heterogeneous robotic system by acting as a micro-scout MAV carrier. It is mentioned that the presented mission relies exclusively on on-board perception (no motion-capture) and is executed autonomously.

The specific mock-up environment setup illustrated in Figure 5 is littered with unstructured obstacle configurations, including boxes, wooden planks, concrete blocks, bricks, and other materials pertaining to a collapsed building interior. The mission process follows the outline presented in Subsection IV-A; the arrangement is such that soon after the beginning of the experiment, no further exploration route can be found due to ground navigation constraints. This initiates a scanning process of the environment with our *aperture* detection algorithm examining the local structure around the MMS. The most consistently detection is probabilistically ranked higher than erratic ones, and indeed corresponds to the environment slot annotated in the Figure.

This is designed to lead into a caved-up corridor like space (with raised irregular walls left-and-right; the top part of the mockup is however left open to allow overhead

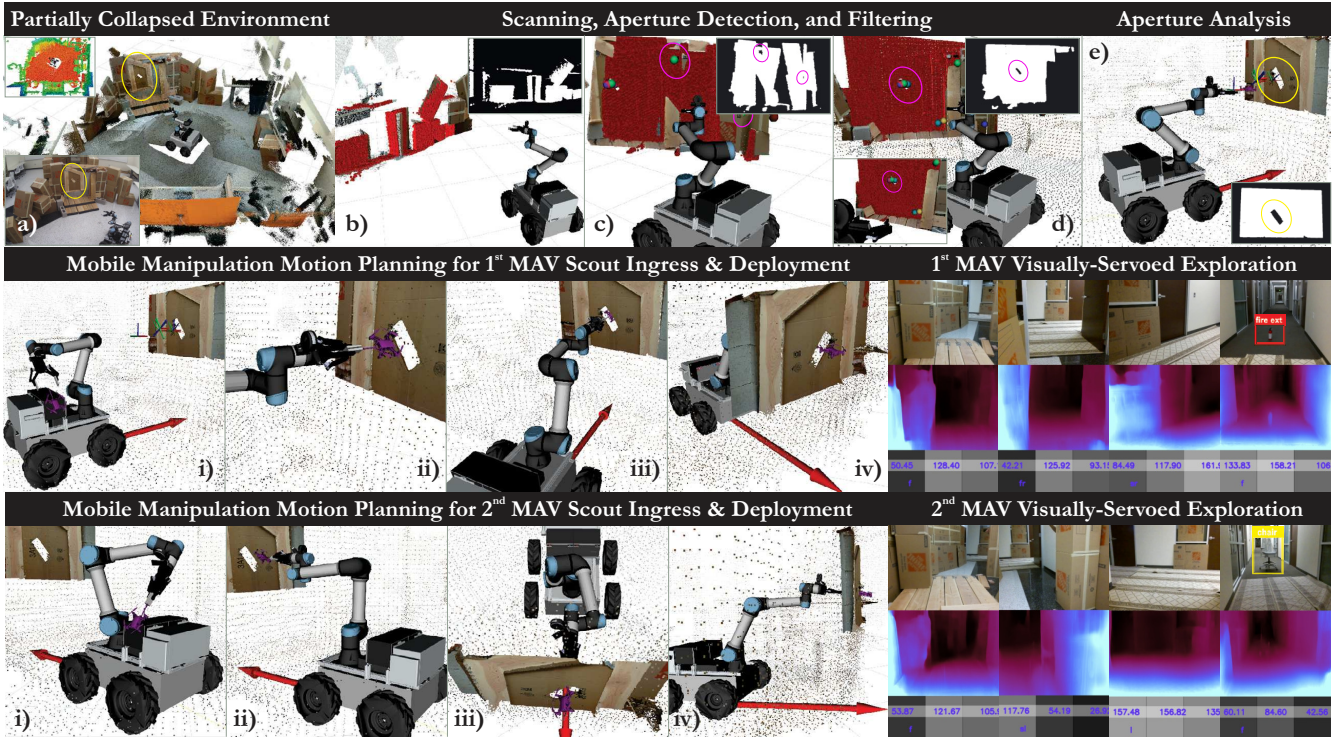


Fig. 5. *Top row: a)* Partially-collapsed mockup environment with a caved-in corridor hidden behind a constricted ingress point, *b) – d)* Scanning to detect *apertures* yields numerous detections, but only one (the ingress point) ranks consistently, *e)* Selected *aperture* approached and analyzed. *Middle and Lower row: Sequential deployment of MAV micro-scouts by i)* picking them up from the MMS body, *ii)* aligning them to the *aperture*, *iii)* inserting them, *iv)* flat-aligning them for takeoff. Deep-learned monocular depth prediction based visually-servoed guidance of the MAVs leads one robot to the corridor right segment (discovering a fire extinguisher) and the other to the left (discovering a chair).

visibility during the experiment). The MMS navigates in a collision-and-traversability-aware [38] manner, positioning itself against the aperture, and the Phases of Manipulation Planning for MAV launch are executed. It is noted that 2 MAVs are available, one carried on the left and the other on the right side of the MMS body. These are inserted into the *aperture* and launched in sequence; as soon as the exploration mission of the first is terminated, the second one is deployed.

As previously elaborated, each MAV micro-scout relays a x264-encoded video stream back to the MMS, which is leveraged by a deep-learned monocular camera depth prediction pipeline. Such an approach is excellently tailored to this case; potential communication latencies (leading to irregular image updates) and the image-stream quality is prone to lead to dubious monocular SLAM results if a first-principle approach were employed instead. Also, the MMS onboard computational resources naturally allow for the deployment of this network. The MAV is therefore guided by our simple visually-servoed exploration guidance scheme, proceeding down the mockup corridor avoiding the box pillars, exiting the door, and turning either left or right to continue down the building hallway. It is noted that although a directionality bias can be applied to make each MAV favor a right-or-left turn direction, this approach is not guaranteed to lead to different exploration routes for the 2 consecutively deployed aerial micro-scouts.

In the case presented, each robot proceeds down different directions, and an object detection and classification pipeline

[41] can be used to terminate each sequence as we have placed a fire extinguisher and a chair on either corridor side. Although this exploration does not lead to comprehensive mapping past the constricted *aperture*, the heterogeneous robotic system manages to acquire semantic information which should be viewed as vital in the context of autonomous robotic reconnaissance operations during catastrophic events.

VI. CONCLUSIONS

In this work, an autonomous heterogeneous robotic system deployment pipeline was proposed, centered around a “carrier” MMS and a set of micro-sized MAVs. The overarching vision is to tackle cases of highly constricted ingress points into caved-in locations, via the autonomous mobile manipulation-based careful insertion and launch of MAVs through them, in order to act as micro-robot scouts. This paper contributes a) an approach for the detection of such apertures and a methodology for hierarchical analysis in order to “fit” robot bodies through them, b) an overarching architecture for a heterogeneous robotic solution capable of these tasks, and c) the novelty of a system-of-systems approach that leverages mobile manipulation for the deployment of other robots which are otherwise incapable of entering such extreme *apertures* in a safe way. The proposed pipeline was evaluated in experimental study comprising relevant robotic systems and a partially-collapsed mockup environment.

REFERENCES

- [1] M. Tranzatto, F. Mascari, L. Bernreiter, C. Godinho, M. Camurri, S. M. K. Khattak, T. Dang, V. Reijgwart, J. Loeje, D. Wisth, S. Zimmermann, H. Nguyen, M. Fehr, L. Solanka, R. Buchanan, M. Bjelonic, N. Khedekar, M. Valceschini, F. Jenelten, M. Dharmadhikari, T. Homberger, P. De Petris, L. Wellhausen, M. Kulkarni, T. Miki, S. Hirsch, M. Montenegro, C. Papachristos, F. Tresoldi, J. Carius, G. Valsecchi, J. Lee, K. Meyer, X. Wu, J. Nieto, A. Smith, M. Hutter, R. Siegwart, M. Mueller, M. Fallon, and K. Alexis, "Cerberus: Autonomous legged and aerial robotic exploration in the tunnel and urban circuits of the darpa subterranean challenge," *Journal of Field Robotics*, 2021.
- [2] Y. L. Karavaev and V. Shestakov, "Construction of a service area of a highly maneuverable mobile manipulation robot," *Intellect. Sist. Proizv.*, vol. 16, no. 3, pp. 90–96, 2018.
- [3] J. Stücker, M. Schwarz, M. Schädler, A. Topalidou-Kyniazopoulou, and S. Behnke, "Nimbro explorer: Semiautonomous exploration and mobile manipulation in rough terrain," *Journal of Field Robotics*, vol. 33, no. 4, pp. 411–430, 2016.
- [4] S. Natarajan, S. Kasperski, and M. Eich, "An autonomous mobile manipulator for collecting sample containers." ESA, 2014.
- [5] G. Paul, S. Webb, D. Liu, and G. Dissanayake, "Autonomous robot manipulator-based exploration and mapping system for bridge maintenance," *Robotics and Autonomous Systems*, vol. 59, no. 7–8, 2011.
- [6] H. Balta, J. Bedkowski, S. Govindaraj, K. Majek, P. Musialik, D. Serano, K. Alexis, R. Siegwart, and G. De Cubber, "Integrated data management for a fleet of search-and-rescue robots," *Journal of Field Robotics*, vol. 34, no. 3, pp. 539–582, 2017.
- [7] T. Tomic, K. Schmid, P. Lutz, A. Domel, M. Kassecker, E. Mair, I. L. Grix, F. Ruess, M. Suppa, and D. Burschka, "Toward a fully autonomous uav: Research platform for indoor and outdoor urban search and rescue," *IEEE robotics & automation magazine*, vol. 19, no. 3, pp. 46–56, 2012.
- [8] A. Bircher, K. Alexis, M. Burri, P. Oettershagen, S. Omari, T. Mantel, and R. Siegwart, "Structural inspection path planning via iterative viewpoint resampling with application to aerial robotics," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 6423–6430.
- [9] A. Bircher, M. Kamel, K. Alexis, M. Burri, P. Oettershagen, S. Omari, T. Mantel, and R. Siegwart, "Three-dimensional coverage path planning via viewpoint resampling and tour optimization for aerial robots," *Autonomous Robots*, vol. 40, no. 6, pp. 1059–1078, 2016.
- [10] B. Grocholsky, J. Keller, V. Kumar, and G. Pappas, "Cooperative air and ground surveillance," *IEEE Robotics & Automation Magazine*, vol. 13, no. 3, pp. 16–25, 2006.
- [11] B. Rao, A. G. Gopi, and R. Maione, "The societal impact of commercial drones," *Technology in Society*, vol. 45, pp. 83–90, 2016.
- [12] T. Dang, F. Mascari, S. Khattak, C. Papachristos, and K. Alexis, "Graph-based path planning for autonomous robotic exploration in subterranean environments," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019.
- [13] R. Marchant and F. Ramos, "Bayesian optimisation for informative continuous path planning," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 6136–6143.
- [14] A. Jones, M. Schwager, and C. Belta, "A receding horizon algorithm for informative path planning with temporal logic constraints," in *IEEE International Conference on Robotics and Automation*. IEEE, 2013.
- [15] D. Berenson, P. Abbeel, and K. Goldberg, "A robot path planning framework that learns from experience," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE, 2012.
- [16] T. Stoyanov, M. Magnusson, H. Andreasson, and A. J. Lilienthal, "Path planning in 3d environments using the normal distributions transform," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*. IEEE, 2010, pp. 3263–3268.
- [17] Z. Gosiewski, J. Ciesluk, and L. Ambroziak, "Vision-based obstacle avoidance for unmanned aerial vehicles," in *2011 4th International Congress on Image and Signal Processing*, vol. 4. IEEE, 2011.
- [18] A. Al-Kaff, Q. Meng, D. Martín, A. de la Escalera, and J. M. Armingol, "Monocular vision-based obstacle detection/avoidance for unmanned aerial vehicles," in *2016 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2016, pp. 92–97.
- [19] G. Bai, X. Xiang, H. Zhu, D. Yin, and L. Zhu, "Research on obstacles avoidance technology for uav based on improved ptam algorithm," in *2015 IEEE International Conference on Progress in Informatics and Computing (PIC)*. IEEE, 2015, pp. 543–550.
- [20] O. Esrafilian and H. D. Taghirad, "Autonomous flight and obstacle avoidance of a quadrotor by monocular slam," in *2016 4th International Conference on Robotics and Mechatronics (ICROM)*, 2016, pp. 240–245.
- [21] E. Lyu, Y. Lin, W. Liu, and M. Q.-H. Meng, "Vision based autonomous gap-flying-through using the micro unmanned aerial vehicle," in *2015 IEEE 28th Canadian Conference on Electrical and Computer Engineering (CCECE)*. IEEE, 2015, pp. 744–749.
- [22] G. Loianno, C. Brunner, G. McGrath, and V. Kumar, "Estimation, control, and planning for aggressive flight with a small quadrotor with a single camera and imu," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 404–411, 2017.
- [23] D. Falanga, E. Mueggler, M. Faessler, and D. Scaramuzza, "Aggressive quadrotor flight through narrow gaps with onboard sensing and computing using active vision," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 5774–5781.
- [24] S. Lia, M. Ozoa, C. De Wagtera, and G. de Croona, "Autonomous drone race: A computationally efficient vision-based navigation and control strategy," *arXiv preprint arXiv:1809.05958*, 2018.
- [25] N. J. Sanket, C. D. Singh, K. Ganguly, C. Fermüller, and Y. Aloimonos, "Gapflyt: Active vision based minimalist structure-less gap detection for quadrotor flight," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 2799–2806, 2018.
- [26] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "Octomap: An efficient probabilistic 3d mapping framework based on octrees," *Autonomous robots*, vol. 34, no. 3, 2013.
- [27] C. Papachristos, M. Kamel, M. Popović, S. Khattak, A. Bircher, H. Oleynikova, T. Dang, F. Mascari, K. Alexis, and R. Siegwart, "Autonomous exploration and inspection path planning for aerial robots using the robot operating system," in *Robot Operating System (ROS)*. Springer, 2019, pp. 67–111.
- [28] C. Papachristos, S. Khattak, and K. Alexis, "Uncertainty-aware receding horizon exploration and mapping using aerial robots," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017.
- [29] A. Bircher, M. Kamel, K. Alexis, H. Oleynikova and R. Siegwart, "Receding horizon "next-best-view" planner for 3d exploration," in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2016. [Online]. Available: <https://github.com/ethz-asl/nbvplanner>
- [30] K. Hashimoto, *Visual servoing*. World scientific, 1993, vol. 7.
- [31] R. Reinhard, T. Dang, E. Hand, C. Papachristos, and K. Alexis, "Learning-based path planning for autonomous exploration of subterranean environments," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 1215–1221.
- [32] P. Arora and C. Papachristos, "Mobile manipulator robot visual servoing and guidance for dynamic target grasping," in *International Symposium on Visual Computing*. Springer, 2020, pp. 223–235.
- [33] M. Labbé and F. Michaud, "Rtab-map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation," *Journal of Field Robotics*, vol. 36, no. 2, pp. 416–446, 2019.
- [34] H. Oleynikova, Z. Taylor, M. Fehr, R. Siegwart, and J. Nieto, "Voxblox: Incremental 3d euclidean signed distance fields for on-board mav planning," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017.
- [35] C. Rsmann, F. Hoffmann, and T. Bertram, "Timed-elastic-bands for time-optimal point-to-point nonlinear model predictive control," in *2015 European Control Conference (ECC)*, 2015, pp. 3352–3357.
- [36] S. Chitta, I. Sucan, and S. Cousins, "Moveit![ros topics]," *IEEE Robotics & Automation Magazine*, vol. 19, no. 1, pp. 18–19, 2012.
- [37] P. Arora and C. Papachristos, "Launching a micro-scout uav from a mobile robotic manipulator arm," in *2021 IEEE Aerospace Conference (50100)*. IEEE, 2021, pp. 1–8.
- [38] —, "Environment reconfiguration planning for autonomous robotic manipulation to overcome mobility constraints," in *2021 International Conference on Robotics and Automation (ICRA)*. IEEE, 2021.
- [39] M. Quigley, S. Batra, S. Gould, E. Klingbeil, Q. Le, A. Wellman, and A. Y. Ng, "High-accuracy 3d sensing for mobile manipulation: Improving object detection and door opening," in *2009 IEEE international conference on robotics and automation*. IEEE, 2009.
- [40] C. Godard, O. Mac Aodha, M. Firman, and G. J. Brostow, "Digging into self-supervised monocular depth estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 3828–3838.
- [41] M. Bjelonic, "YOLO ROS: Real-time object detection for ROS," <https://github.com/leggedrobotics/darknet-ros>, 2016–2018.