

Vision-based Landing Site Evaluation and Trajectory Generation Toward Rooftop Landing

Vishnu R. Desaraju, Nathan Michael
The Robotics Institute
Carnegie Mellon University
{rajeswar, nmichael}@cmu.edu

Martin Humenberger, Roland Brockers, Stephan Weiss, Larry Matthies
Jet Propulsion Laboratory
California Institute of Technology
{Martin.Humenberger, Roland.Brockers, Stephan.M.Weiss, lhm}@jpl.nasa.gov

Abstract—Autonomous landing is an essential function for micro air vehicles (MAVs) for many scenarios. We pursue an active perception strategy that enables MAVs with limited onboard sensing and processing capabilities to concurrently assess feasible rooftop landing sites with a vision-based perception system while generating trajectories that balance continued landing site assessment and the requirement to provide visual monitoring of an interest point. The contributions of the work are twofold: (1) a perception system that employs a dense motion stereo approach that determines the 3D model of the captured scene without the need of geo-referenced images, scene geometry constraints, or external navigation aids; and (2) an online trajectory generation approach that balances the need to concurrently explore available rooftop vantages of an interest point while ensuring confidence in the landing site suitability by considering the impact of landing site uncertainty as assessed by the perception system. Simulation and experimental evaluation of the performance of the perception and trajectory generation methodologies are analyzed independently and jointly in order to establish the efficacy of the proposed approach.

I. INTRODUCTION

Autonomous landing is an essential function for micro air vehicles (MAVs) for “perch-and-stare” mission scenarios, for conserving energy during lulls in mission timelines and dropping off payloads [18]. In this work, we develop a vision-based methodology to enable a MAV to autonomously detect and navigate to a rooftop landing site. We envision a scenario that deploys a MAV to a region with the objective to autonomously identify a suitable rooftop landing site that offers favorable vantages of an interest point for surveillance. We pursue an active perception strategy that enables the vehicle to concurrently assess feasible landing sites with a vision-based perception system while generating trajectories that balance continued landing site assessment and the requirement to provide visual monitoring of an interest point (see Fig. 1).

Most prior work on autonomous landing of unmanned air vehicles addresses landing on terrain, instead of finding elevated perches like rooftops. Due to the severe size, weight, and power (SWaP) constraints of MAVs, applicable methods for MAVs must use much lighter, lower performance sensing and computing resources than available on larger scale systems [21]. Approaches amenable to these SWaP constraints frequently employ monocular [2, 8, 15, 24] and binocular stereo [13, 25] camera systems to map and analyze terrain. Most approaches perform some form of 3D terrain recon-

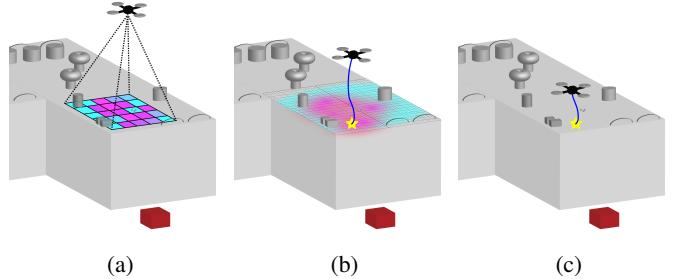


Fig. 1: Methodology overview. (a) A quadrotor micro aerial vehicle approaches a rooftop while modeling landing confidence via the vision-based perception system toward monitoring of an interest point (red box). (b) Landing site confidence is modeled as an evolving Gaussian process trained on new observations from the perception system. (c) Online trajectory generation balances landing site feasibility, interest point observability, and the impact of future measurements through the maximization of an information-theoretic objective.

struction, then assess planarity and slope of appropriately-sized terrain patches. Binocular stereo vision approaches are simpler algorithmically, due to the fixed inter-camera geometry, but are limited by the inter-ocular baseline and heavier. Three monocular approaches are particularly relevant here. The first tracks point features to estimate homographies from image pairs for predominantly planar terrain, then analyzes correlation coefficients for dense matches to segment in-plane and out-of-plane pixels [2]. The second uses a recursive filter at each pixel, image matching via gradient descent with intensity derivatives, and a plane plus parallax formulation of structure from motion to estimate dense elevation maps from image sequences [24]. Both of these address finding landing sites on the ground. The third uses multi-planar homography with tracked features to segment a planar ground-level surface from an elevated, planar landing site [3].

Our proposed perception system uses a monocular camera and employs a dense motion stereo approach that estimates a 3D model of the scene without requiring geo-referenced images, dominant planes, or external navigation aids like GPS. We present a frame list approach that chooses a variable baseline to enable arbitrary selection of depth accuracy of the 3D model as long as the motion between an image

pair can be found correctly. This approach is applicable to complex, non-planar scenes and is generalizable to multi-frame reconstruction. All processing is done onboard in real time.

The proposed online trajectory generation approach balances the need to concurrently explore available rooftop vantages of an interest point while ensuring confidence in the landing site suitability. When the position of the landing site is initially uncertain or unknown, trajectory generation approaches may seek to minimize the expected site uncertainty through future measurements by reducing the expected covariance of the target estimate propagated through a sensor-based prediction model [6, 10, 17]. However, such forward propagation is not viable given the proposed perception system, and so we pursue an information-based planning approach that seeks to generate optimal trajectories with respect to a spatially-varying field representing landing site confidence that is updated online via the perception system [16]. Similar to prior work, we choose a Gaussian process representation to guide trajectory planning toward minimizing the variance in the field estimate [11, 14, 26]. However, our approach differs in that we consider the impact of image space observations on the field estimate, rather than treating the entire image as a single point-observation of the field.

The organization of the presentation follows. We discuss the vision-based landing site evaluation methodology and the probabilistic representation of the spatially-varying conditional landing site confidence in Sect. II. The online information-based trajectory generation approach based on the observations from the perception system is detailed in Sect. III. Simulation and experimental evaluation of the performance of the perception and trajectory generation methodologies are analyzed independently and jointly in order to evaluate the efficacy of the proposed approach in Sects. IV–V, respectively.

II. VISION-BASED LANDING SITE DETECTION

The landing site detection algorithm consists of two parts: the 3D reconstruction of the visible scene below the MAV using dense motion stereo, and the analysis of the scene in order to search for potential landing candidates. The results of the algorithm are a binary landing map in image space with underlying 3D coordinates in the world frame and a confidence measure for the landing site quality. We transform this binary map in image space to a spatially-varying conditional probability distribution via a Gaussian process toward enabling the information-based trajectory generation techniques in Sect. III.

A. 3D Reconstruction of the Visible Scene

Conventional stereo vision uses two rigidly mounted cameras, separated by the baseline, to capture a scene from different points of view. To account for lens distortion effects and to determine the stereo geometry that is needed for metric 3D reconstruction of an observed scene, camera intrinsics and extrinsics are usually determined by an offline calibration procedure and are assumed to be constant during operation.

Dense motion stereo is based on the same principle, with the difference that the two views of the captured scene are generated by one moving camera instead of a rigid stereo bar. In this case the extrinsic parameters (rotation and translation between the two camera positions) have to be determined for each image pair individually. The intrinsic parameters do not change and can therefore be calibrated offline and in advance. We use a CAHVORE camera model [5] to model lens effects and to generate linearized camera models that describe the perspective projection.

The fact that one moving camera is used instead of two rigidly mounted cameras leads to the following challenges. (1) Image based motion estimation between two images at different camera positions can only be calculated up to scale. This scale factor must be calculated from additional information (e.g., vehicle pose) in order to convert camera coordinates to metric world coordinates. (2) Rotation and translation between the camera images must be determined precisely to enable proper stereo matching with subpixel accuracy. (3) The distance between the two viewpoints can greatly impact the accuracy of the calculation of depth.

Figure 2 provides an overview of the image processing pipeline of our dense motion stereo framework. We select image pairs from an image frame list with a fixed length time-window and extract feature points (STAR features and MSURF descriptor) in each image, tracking them in subsequent images. Correspondences are chained over time, so that feature track lists record how long each feature was tracked and where it appeared in each image. To select image pairs, we search a reference image for the most recent image using two criteria. First, given the camera pose in the world frame (estimated from inertial measurements using an extended Kalman filter), we look for images that are an appropriate distance apart from our current image. The minimum baseline is determined using the minimum required depth accuracy at ground level. Once images which fulfill this criterion have been found, we secondly choose the image which exceeds a minimum number of successive feature matches with the current image, based on the track lists.

Since camera motion estimates from the pose estimation EKF are not accurate enough for dense motion stereo, we estimate the rotation and translation between the images via a multi-planar homography alignment approach [4]. The translation vector is then scaled by the real-world baseline. Since the quality of motion estimation depends on the quality of feature detection and matching, and thus is strongly scene dependent, we calculate the average 3D reprojection error of the feature pairs to discard poor motion estimates. Finally, we use a SAD5 stereo matching algorithm to calculate a disparity map, from which we generate a 3D point cloud to model the captured scene beneath the MAV [7].

B. Landing Site Detection

We define a suitable landing site as: (1) approximately planar and level; (2) sufficiently large to permit MAV ingress, landing, and egress; and (3) free of obstacles to ensure vehicle

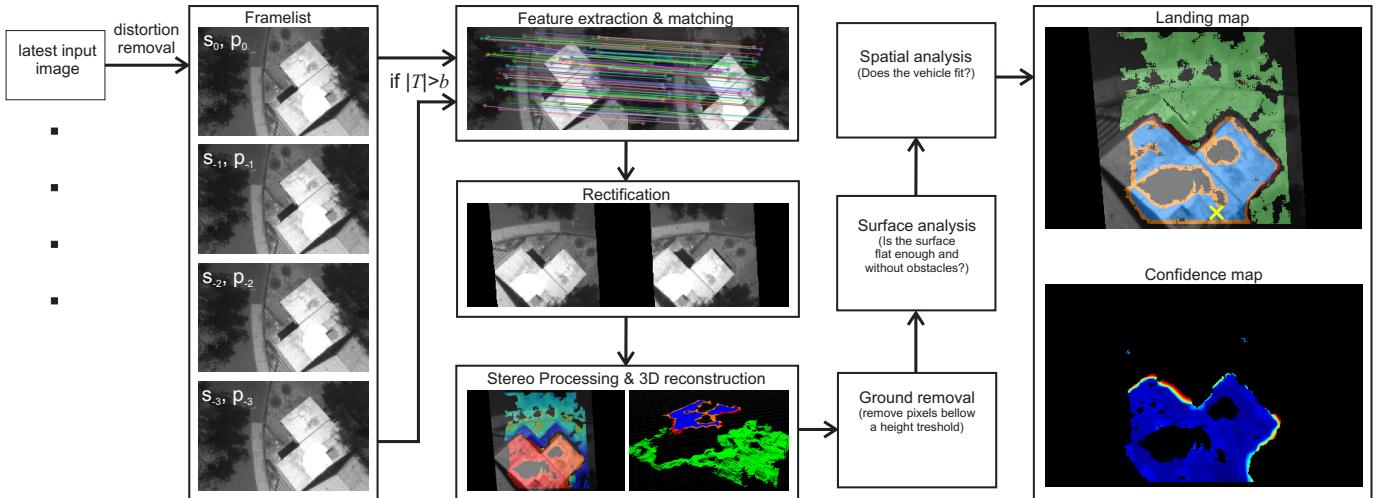


Fig. 2: Landing detection overview. A new input image is warped to remove distortion and stored together with detected features and camera pose in a frame list. An image pair with an appropriate distance (T , defined by the minimum baseline b) is selected. Features are matched between selected frames, in order to calculate a frame to frame transform for rectification. Stereo disparity based 3D reconstruction serves as an input to the landing site detection algorithm. The calculated landing map labels all pixels with a valid stereo disparity as: green (below roof top), red (on roof top but unsafe), orange (insufficient space), and blue (safe landing area). The location with the highest confidence is marked by \times (blue and red indicate high and low confidence, respectively, in the confidence map)

safety. To fulfill these requirements, we propose a probabilistic model based on the range data that captures the statistical variability of the range data.

We first reject candidates that are close to the ground level via a depth resolution dependent threshold. The resulting disparity map is used to determine planar regions in the camera image, ignoring pixels with invalid range data. We observe that the variance of the disparity map along the gravity vector after projection into world frame corresponds to the planarity of the landing surface. The smaller the standard deviation in the disparity map, the more planar the corresponding area. For each pixel, we compute the second moment over a local kernel along the world frame vertical axis, where the kernel dimensions are chosen to capture the dimensions defined by the vehicle size and ingress, egress requirements. The local kernel depends on the disparity value d and the MAV's altitude h . We define the MAV bounding radius r , the focal length f , and the target depth accuracy z_{acc} , where target depth accuracy is the first derivative of h with respect to the disparity. The minimum landing space in the image domain is the vehicle's size divided by the lateral resolution of the camera. The local kernel for each landing candidate is therefore

$$n(d, h) = \frac{rz_{acc}f}{h^2}d. \quad (1)$$

The standard deviation is further used to generate a normalized confidence value for each landing candidate. A color-coded confidence map is shown in Fig. 2.

C. Spatial Representation of the Landing Site Confidence as a Gaussian Process

The perception system detailed in Sect. II provides an estimate of landing confidence, $l \in [0, 1]$, for each point in image space. We spatially smooth the observations across frames using a Gaussian process (GP) whose inputs are positions, $r = (x, y, z)$. The GP can be viewed as a linear smoother that provides the probability of being able to land at a location, $P(l|r)$, using a standard Gaussian kernel to determine smoothing weights [20]. Specifically, given a set of training locations \mathcal{R}_T and the corresponding training data $\{(r, l) | r \in \mathcal{R}_T\}$ for the GP, the mean value at a test point r_t is a weighted sum of all the training data

$$\hat{l}_t = \sum_{i=1}^{|\mathcal{R}_T|} w_i l_i$$

where the weight function

$$w_i = \kappa(r_i - r_t, \sigma_\kappa) / \sum_{j=1}^{|\mathcal{R}_T|} \kappa(r_j - r_t, \sigma_\kappa) \quad (2)$$

is defined by the kernel function κ centered about r_t with variance σ_κ^2 [20]. We assume the hyperparameters have been optimized prior to the approach for landing using prior sensor observations. This could include observations collected from the initial exploration phase of the landing trajectory (see Sect. IV), which would also capture current environmental properties.

To reduce the computational load, we downsample the input data in the image space, as illustrated in Fig. 1a prior to

introducing the values as training data for the GP. Additionally, sparse [23] and incremental [19] GP techniques may be employed to further reduce the computational burden.

Potential landing sites are identified by thresholding the GP mean. We discretize the GP and label a grid cell as a landing site if its mean confidence value is above a threshold. The resulting set of candidate landing sites are then sorted based on the mean confidence value, and a landing site is selected from this list that minimizes the distance to the surveilled interest point (Fig. 1b).

III. INFORMATION-BASED TRAJECTORY DESIGN

After selecting a landing site, the MAV must plan a smooth trajectory to land. Following Shen et al. [22], we choose to minimize the angular velocity along the trajectory in order to reduce image distortion due to motion blur. Therefore, we compute the optimal polynomial trajectory from the current state to the selected landing site with a desired final heading ψ_f . We define a p^{th} -order spline for the flat outputs (for a quadrotor) $s \in \{x, y, z, \psi\}$ [12] as

$$s(t) = \sum_{i=0}^p c_{s_i} t^i$$

The spline coefficients c_{s_i} are determined via a multi-objective optimization. For the quadrotor platform, minimizing the angular velocity corresponds to minimizing the jerk with an associated cumulative cost

$$J_1 = \int_{t_0}^{t_f} \left[\left\| \frac{d^3 r}{dt^3} \right\|_2^2 + \left(\frac{d\psi}{dt} \right)^2 \right] dt \quad (3)$$

We introduce a second objective function to maximize information gain in order to ensure continued observation of the current landing site toward a reduction in the landing site confidence uncertainty. The information gain along a trajectory reflects the effect of expected future measurements on the belief distribution. Since the distribution here is given by a GP, the trajectory should produce additional training data that will reduce uncertainty on the confidence value at the current landing site.

We define a generative sensor model in order to predict the expected observations given the camera extrinsic and intrinsic properties. We can directly compute weights for the set of predicted measurement locations $r \in \mathcal{R}_P$ independent of the confidence values at those locations as the weight function in (2) is only a function of the GP inputs r and not the outputs l . These weights quantify the contribution of the predicted measurements to the GP belief at r_t . After augmenting the training data with the predicted measurement locations, the new weights are given by

$$w_i = \kappa(r_i - r_t, \sigma_\kappa) / \sum_{j=1}^{|\mathcal{R}_T \cup \mathcal{R}_P|} \kappa(r_j - r_t, \sigma_\kappa)$$

The total contribution of the predicted measurements relative to the existing training data is

$$W = \sum_{i|r_i \in \mathcal{R}_P} w_i$$

Given that the variance of a test point in a GP is determined by the total contribution of training points to the value at that test point, adding training points with a large relative contribution W will reduce the variance. Therefore, to maximize the information gain about a location r_t along a trajectory, we maximize the following objective function

$$J_2 = J_{\text{info}}(r_t, \sigma_\kappa) = \sum_{k=1}^K W_k \quad (4)$$

where W_k is computed at K points uniformly distributed along the trajectory. As \mathcal{R}_T is augmented with \mathcal{R}_{P_k} at each step k , this information metric also captures the diminishing returns from new data near the test point.

We can also use a variant of this information objective to promote exploration of potential landing sites that minimize the distance to an interest point or surveillance target, r_{target} . The exploration objective is defined as

$$J_3 = J_{\text{info}}(r_{\text{target}}, \sigma_{\text{explore}}) \quad (5)$$

where we select $\sigma_{\text{explore}} \gg \sigma_\kappa$ to reward measurements collected in a larger neighborhood of r_{target} . This also leads to diminishing returns earlier, automatically decreasing the value of the exploration objective over time and producing a natural transition from favoring exploration (J_3) to exploitation (J_2).

The optimization to determine the coefficients c_{s_i} is formulated as a minimization that combines the minimum jerk and maximum information objectives ((3) and (4), respectively) with weights $\alpha_1, \alpha_2, \alpha_3 > 0$

$$\begin{aligned} \arg \min_{c_{s_i}} \quad & \alpha_1 J_1 - \alpha_2 J_2 - \alpha_3 J_3 \\ \text{s.t.} \quad & s(t_0) = s_0 \\ & s(t_f) \geq s_f^- \\ & s(t_f) \leq s_f^+ \\ & \dot{s}(t_0) = \dot{s}_0 \\ & \dot{s}(t_f) = 0 \\ & \forall s \in \{x, y, z, \psi\} \end{aligned} \quad (6)$$

where s_0 and \dot{s}_0 are given by the vehicle's initial state. The bounds on the terminal state, s_f^- and s_f^+ , are determined by the discretization used to identify potential landing sites from the GP in Sect. II-C. The final velocity constraint enforces the requirement to land at the selected site. The trajectory is recomputed online as the perception system provides additional observations and the landing site confidence evolves.

IV. SIMULATION RESULTS

We now focus on evaluating the impact of the multiple objectives proposed in Sect. III on the autonomous landing performance. We consider the “perch-and-stare” scenario in

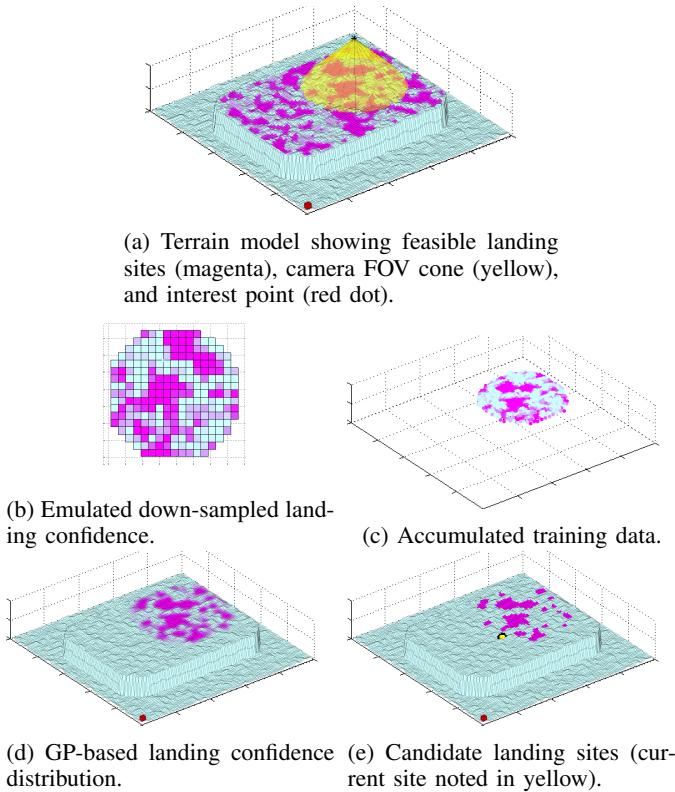


Fig. 3: A simulated autonomous landing scenario.

which the vehicle must land on a cluttered rooftop in proximity to a monitoring target, and the perception system is emulated via a synthesized rooftop surface with associated landing confidence values (Fig. 3a). A confidence measure similar to that proposed in Sect. II-B is generated (Fig. 3b) and introduced into a GP as training data (Fig. 3c), as described in Sect. II-C. Figure 3d shows the resulting GP superimposed on the true rooftop surface with thresholded candidate landing sites shown in Fig. 3e.

Toward comparing the performance of the objective functions, define J_4 and J_5 as objective functions that seek to minimize the distance from the center of the camera FOV to the current landing site and interest point, respectively. Table I shows a comparison of landing trajectories generated by six objective functions constructed from J_1 through J_5 . The vehicle's initial state, re-plan rate, trajectory duration, vision update rate, and target location are held constant across all objectives. The trajectories are evaluated based on their distance to the target after landing, the length of the resulting trajectory, and the cumulative jerk along that trajectory.

The dark gray cells in the table are ignored due to poor landing site selection, which stems from insufficient exploration. The other three objectives explore more due to the lack of the information objective (4), resulting in longer trajectories. These results suggest that the objective in (6) provides a reasonable balance between exploring the environment and exploiting information gain to focus on the landing site.

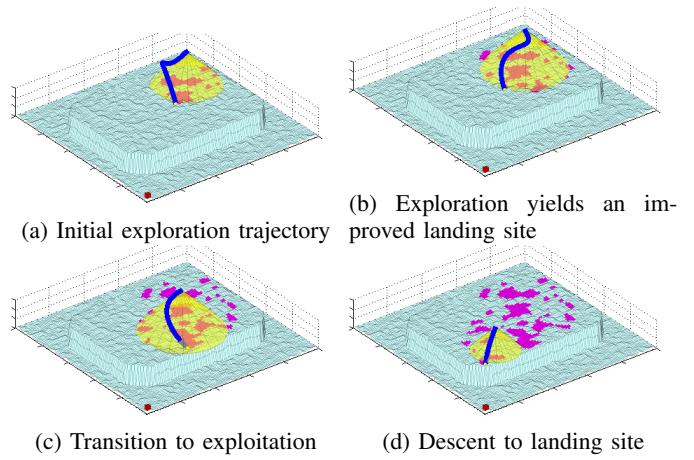


Fig. 4: Vinette depicting the exploration, exploitation phases of the autonomous landing maneuver.

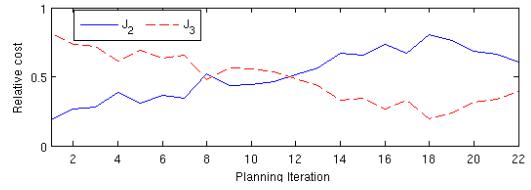


Fig. 5: Relative contribution of the two information terms to the trajectory cost at each planning iteration and the transition from exploration (J_3) to exploitation (J_2).

Figure 4 shows the evolution of the trajectory over several planning iterations.

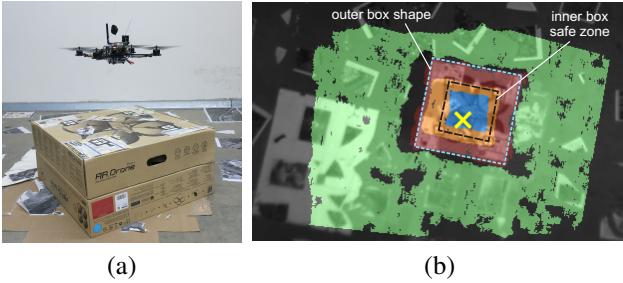
Figure 5 shows the relative contribution of the exploration and exploitation (information) objectives to the cost of the trajectory computed at each planning iteration. Trajectories initially favor exploration but gradually transition to exploitation due to diminishing returns from exploration and the increasing value of observations approaching the landing site.

V. EXPERIMENTAL EVALUATION

We now consider the experimental evaluation of the proposed perception system and trajectory generation approach in a rooftop landing environment. For the latter, we synthesize the trajectories offline using data from the outdoor perception system evaluation and are presently in the process of integrating the two methodologies toward enabling experimental evaluation of the complete active perception system.

TABLE I: Comparison of objective functions. Shaded cells indicate the minimum in each row.

Objective:	J_1	J_{1+4}	J_{1-2}	J_{1+5}	J_{1-3}	J_{1-2-3}
Site to Target (m)	75.57	28.56	78.33	29.15	28.56	28.56
Path Length (m)	24.65	129.42	23.40	259.78	122.74	81.58
Total Jerk (m/s^3)	24.45	142.98	27.35	175.66	96.17	68.62



(a)

(b)

Fig. 6: Indoor experiment: (a) box as surrogate roof-top; (b) hand labeled landing map, outer box shape is the edge of the box surface, inner box safety zone is the box surface without the 13 cm border region.

A. Implementation Details

1) *System Configuration*: For evaluation of the landing detection algorithm, we implemented the pose estimation framework and the landing detection software onboard an Asctec Hummingbird that is equipped with an Odroid U2 processing board (Exynos4412, 1.7GHz). The EKF framework is distributed according to [27] between the U2 flight computer where we run visual odometry and calculate filter updates at 30Hz, whereas the propagation of the quadrotor position, velocity and attitude is executed at 1kHz on the platform's embedded ARM7 processor. The U2 flight computer also hosts the landing detection software which is executed onboard at a rate of 1Hz. The platform is equipped with a gray-scale Matrix Vision mvBlueFOX camera (Aptina MT9V034 sensor) with a resolution of 752x480, global shutter, and 100° horizontal FOV.

2) *Pose Estimation*: An accurate pose estimate of the vehicle is required to enable the perception system evaluation. To estimate the pose in a gravity aligned navigation frame we use the approach described in [28]. It uses a modified version of PTAM [9], which is robust in fast motion and self-similar environments [1]. Since it is a key-frame based visual odometry approach, the computed camera pose does not drift while the quadrotor is hovering. We use the arbitrarily scaled 6 DoF output of this module in an EKF framework and fuse it with inertial readings as described in [28]. This not only scales the visual pose to metric units but also aligns the navigation frame with gravity. The system also continuously self-calibrates the camera-IMU extrinsics and IMU biases.

B. Perception System Performance Evaluation

1) *Indoor evaluation*: The indoor experiments were designed to evaluate proper system performance and generate quantitative error metrics from hand labeled ground truth data. We fly our quadrotor system at three different altitudes over a box ($57 \times 57 \times 27 \text{ cm}^3$) (Fig. 6) to simulate a rooftop landing scenario. To increase the number of calculated frames, we pre-recorded input images and vehicle poses during overflight and run the landing detection software off-line at frame rate. In all experiments, the vehicle radius was set to 13 cm to allow

for a sufficiently sized valid landing area in the middle of the box, and the arbitrary ground level cut off threshold was set to 20 cm. For ground truth, the true landing area in the middle of the box surface was marked by corner marks that were located at 13 cm distance from the box edges, and these were manually identified in the input images (see Fig. 6b).

The first three rows of Table II give an overview of the evaluation results. Altitude, baseline, and reprojection error during image alignment correspond to the average value for each experiment. For the evaluation, we only considered frames where the box surface was completely visible in the disparity image to avoid border effects and where valid stereo results could be calculated (i.e., with sufficient baseline and more than 40 feature matches for image alignment; denoted “frames visible”). Within these frames, we defined a successful detection (“frames successful”) as frames where at least one valid landing location was detected on the box and no landing location was detected falsely on the ground. For these successful frames, we also calculated false positive (FP) rates (pixels that were classified as valid landing areas but were located in the border area), false negative (FN) rates (pixels not classified as landing areas that were located in the correct center area of the box). Note, that we only consider pixels with valid disparity values in this metric.

Our approach is able to robustly detect the landing zone with a success rate of more than 90% in all experiments, with a false positive rate below 0.05%. The false positive (FP) and false negative (FN) rate is largely defined by the quality of the disparity input. Border fattening effects usually increase the FP rate, whereas missing disparity pixels on top of the target lead to increased FN rates, since we treat missing data as unsafe. To mitigate these two effects we introduced two thresholds to maximize safety: (1) at disparity edges, we disable all pixels that are located within half a correlation window size to the edge, and (2) we use a percentage threshold which defines the minimum number of pixels with disparity around a landing site (98% in our experiments).

To verify the accuracy of our 3D reconstruction, we plotted the height of the landing point with the highest confidence for one of the sequences (Fig. 7). The error follows the expected depth accuracy of 3 cm within the true box height of 27.5 cm.

TABLE II: Evaluation of landing site detection. Row 1-3: Indoor experiment at three different altitudes ($r = 13 \text{ cm}$, depth accuracy at ground level=3 cm); Row 4-5: outdoor experiment (depth accuracy at ground level=20 cm).

altitude [m]	baseline [m]	repro. err. [pixel]	No. of frames		FP [%]	FN [%]
			visible	successful		
1.47	0.39	0.21	116	106	91.4%	0.028 39.34
2.19	0.86	0.20	63	62	98.4%	0.046 43.57
3.34	2.00	0.18	67	63	94.0%	0.0026 53.14
6.51	1.16	0.28	337	325	96.3%	
10.15	2.28	0.22	480	433	90.2%	

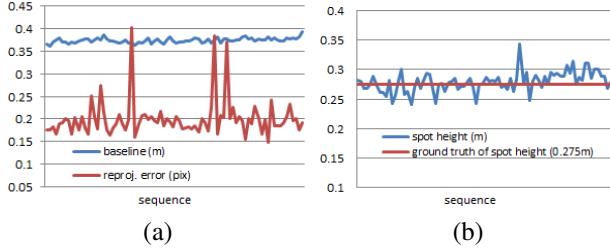


Fig. 7: Indoor experiment: (a) Box height calculation for valid landing point with highest confidence. (b) Reprojection error of feature matches and estimated baseline (depth accuracy 3 cm).

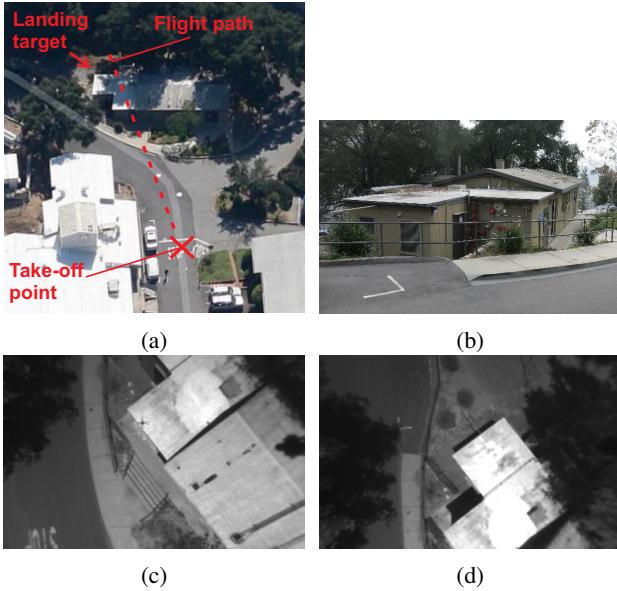


Fig. 8: Outdoor experiment environment and data set images. (a) aerial view of the flight area and target building; (b) the flat area on the roof is the landing target; (c) raw input image with good texture on roof; (d) image with saturation area which leads to missing stereo data.

The low average reprojection error confirms valid motion estimation results.

2) Outdoor evaluation: For the outdoor experiments, we conducted overflights over a one story building (Fig. 8) and recorded image sequences from the downward looking camera together with pose data for off-line analysis.

A quantitative evaluation for two different overflights is given in Table II rows 4-5. The average altitude of the first flight was 6.5 m, which lead to an average required baseline of 1.16 m. The second overflight was at a higher altitude of approx. 10.15 m requiring a slightly higher minimum baseline of 2.28 m on average. From all frames where at least a part of the safe landing zone on top of the building was visible in the disparity images, we could successfully identify a valid landing target in over 90% of the frames for both experiments.

C. Simulated Trajectory Generation via Outdoor Datasets

We also evaluated the GP-based landing site selection and trajectory generation components using the outdoor flight data. Figure 9a shows an approximate reconstruction of the environment from the point cloud data with the grayscale image data overlaid. The raised, light-colored section in the center is the building on which we wish to land (see also Fig. 8).

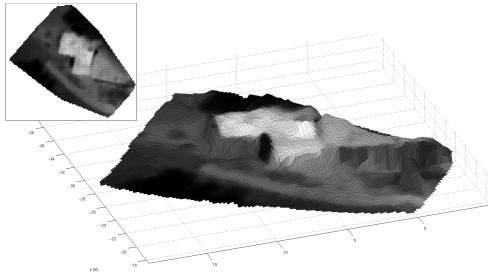
The confidence map from each image frame is binned and introduced as training data to the GP, as in the simulations. We compute the GP hyperparameters on the data from the initial frames, updating the covariance and kernel functions.

The distribution given by the GP after exploring the environment is shown in Fig. 9c and accurately identifies much of the rooftop as landing sites with high confidence. Figure 9d shows the discrete landing site candidates, which allows us to select the candidate closest to the point of interest and compute the remainder of the trajectory to land at the site.

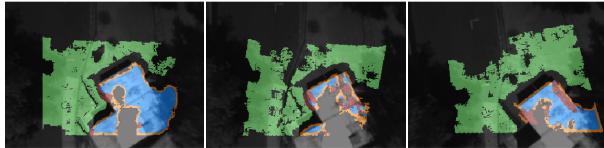
VI. CONCLUSION AND FUTURE WORK

We have addressed a MAV autonomous landing scenario involving landing on rooftops for surveillance of nearby areas. Due to the severe SWaP constraints of MAVs, our approach uses a single, down-looking camera and an IMU for all state estimation and landing site detection. GPS-denied pose estimation algorithms are drawn from previous work. Landing site detection uses a dense motion stereo-based approach with a frame list that chooses variable baselines to match range resolution to the environment and flight conditions. Simple planarity analysis of aircraft-sized regions of the disparity maps labels pixels as candidate landing sites with corresponding confidence measures. These are filtered over time in a Gaussian process-based representation, which in turn is used by an information-based planning approach. The planner seeks to generate optimal trajectories using an objective function that trades off exploration of the rooftop and minimizing uncertainty on the selected landing site. Simulations have validated the planner and experiments with indoor and outdoor data sets have validated the landing site detection technique.

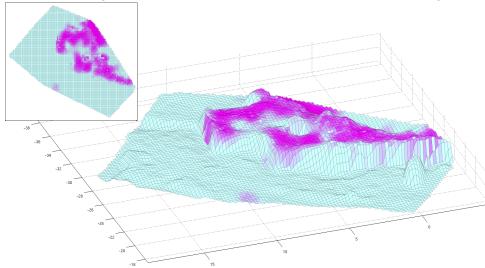
Currently, all components are implemented onboard testbed quadrotor aircraft, but full integration of perception with planning and end-to-end, outdoor testing remains to be done. The primary challenge with this integration stems from allocating sufficient resources to each component to run in real-time. However, since the cost functions query the GP in a relatively small area (i.e., a section of a rooftop) at a user-defined resolution, we can easily adjust these parameters to balance the computational requirements of the planner against those of the perception system. Future work will extend the system to address more sophisticated landing site selection criteria and to land on a wider class of elevated structures for perch-and-stare missions.



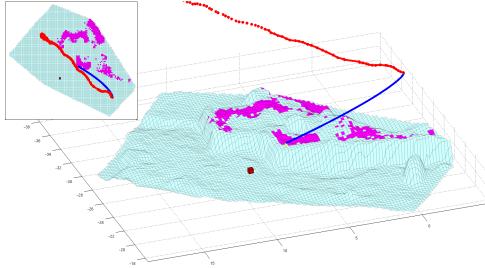
(a) Environment model constructed from point cloud data (light patch is rooftop).



(b) Image data frames data (viable landing sites in blue).



(c) GP after 90 frames overlaid on environment model



(d) Viable sites identified, trajectory to best landing site computed (blue) from end of traversed trajectory (red)

Fig. 9: Illustration of pipeline applied to outdoor flight data. Insets in 3D figures show corresponding bird's-eye view.

ACKNOWLEDGMENTS

We gratefully acknowledge the support of ARL Grant W911NF-08-2-0004.

REFERENCES

- [1] M. W. Achtelik, S. Weiss, S. Lynen, M. Chli, and R. Siegwart. Vision-based MAV navigation: implementation challenges towards a usable system in real-life scenarios. In *Proc. of Robot.: Sci. and Syst.*, Workshop on Integration of Perception with Control and Navigation for Resource-Limited, Highly Dynamic, Autonomous Systems, Sydney, Australia, July 2012.
- [2] S. Bosch, S. Lacroix, and F. Caballero. Autonomous detection of safe landing areas for an UAV from monocular images. In *Proc. of the IEEE/RSJ Intl. Conf. on Intell. Robots and Syst.*, pages 5522–5527, 2006.
- [3] R. Brockers, S. Susca, D. Zhu, and L. Matthies. Fully self-contained vision-aided navigation and landing of a micro air vehicle independent from external sensor inputs. *Proc. SPIE*, 8387:83870Q-1 – 83870Q-10, 2012.
- [4] Y. Cheng. Real-time surface slope estimation by homography alignment for spacecraft safe landing. In *Proc. of the IEEE Intl. Conf. on Robot. and Autom.*, pages 2280–2286, 2010.
- [5] K. Di and R. Li. CAHVOR camera model and its photogrammetric conversion for planetary applications. *J. Geo. Research*, 109(E04004), 2004.
- [6] E. W. Frew and S. M. Rock. Trajectory generation for constant velocity target motion estimation using monocular vision. In *Proc. of the IEEE Intl. Conf. on Robot. and Autom.*, pages 3479–3484, Taipei, Taiwan, September 2003.
- [7] H. Hirschmueller, P. Innocent, and J. Garibaldi. Real-time correlation-based stereo vision with reduced border errors. *J. Comp. Vision*, 47:229–246, 2002.
- [8] A. Johnson, J. Montgomery, and L. Matthies. Vision guided landing of an autonomous helicopter in hazardous terrain. In *Proc. of the IEEE Intl. Conf. on Robot. and Autom.*, pages 3966–3971, 2005.
- [9] G. Klein and D. Murray. Parallel tracking and mapping for small ar workspaces. In *Proc. of the Intl. Sym. on Mixed and Augmented Reality*, Nara, Japan, November 2007.
- [10] D. Levine, B. Luders, and J. P. How. Information-theoretic motion planning for constrained sensor networks. *J. Aero. Inf. Syst.*, pages 1–26, 2012.
- [11] K. H. Low, J. Chen, J. M. Dolan, S. Chien, and D. R. Thompson. Decentralized active robotic exploration and mapping for probabilistic field classification in environmental sensing. In *Intl. Conf. on Autom. Agents and Multiagent Syst.*, 2012.
- [12] R. Mahony, V. Kumar, and P. Corke. Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor. *IEEE Robot. Autom. Mag.*, 19(3):20–32, September 2012.
- [13] M. Meingast, C. Geyer, and S. Sastry. Vision based terrain recovery for landing unmanned aerial vehicles. In *Proc. of the IEEE Conf. on Decision and Control*, volume 2, pages 1670–1675, 2004.
- [14] A. Meliou, A. Krause, C. Guestrin, and J. M. Hellerstein. Nonmyopic informative path planning in spatio-temporal models. *AAAI*, pages 602–607, 2007.
- [15] J. Montgomery, A. Johnson, S. Roumeliotis, and L. Matthies. The Jet Propulsion Laboratory autonomous helicopter testbed: A platform for planetary exploration technology research and development. *J. Field Robot.*, 23(3-4):245–267, 2006.
- [16] J. Nguyen, N. Lawrence, R. Fitch, and S. Sukkarieh. Energy-constrained motion planning for information gathering with autonomous aerial soaring. In *Proc. of*

- the IEEE Intl. Conf. on Robot. and Autom.*, pages 3825 – 3831, Karlsruhe, Germany, May 2013.
- [17] S. S. Ponda. Trajectory optimization for target localization using small unmanned aerial vehicles. Master’s thesis, MIT, Cambridge, MA, September 2008.
- [18] S. D. Prior, S. Shen, M. Erbil, M. Brazinskas, and W. Mielniczek. HALO the winning entry to the DARPA UAVForge challenge 2012. In *Design, User Experience, and Usability*, pages 179–188. Springer Berlin Heidelberg, 2013.
- [19] C. E. Rasmussen and Z. Ghahramani. Infinite mixtures of Gaussian process experts. In *Adv. in Neural Inf. Process. Syst.*, pages 881–888, Vancouver, Canada, December 2001.
- [20] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.
- [21] S. Scherer, L. Chamberlain, and S. Singh. Autonomous landing at unprepared sites by a full-scale helicopter. *Robot. Auton. Syst.*, 60(12):1545–1562, December 2012.
- [22] S. Shen, Y. Mulgaonkar, N. Michael, and V. Kumar. Vision-based state estimation and trajectory control towards high-speed flight with a quadrotor. In *Proc. of Robot.: Sci. and Syst.*, Berlin, Germany, June 2013.
- [23] E. Snellson and Z. Ghahramani. Sparse Gaussian processes using pseudo-inputs. In *Adv. in Neural Inf. Process. Syst.*, pages 1257–1264, Vancouver, Canada, December 2005.
- [24] T. Templeton, D. H. Shim, C. Geyer, and S. S. Sastry. Autonomous vision-based landing and terrain mapping using an MPC-controlled unmanned rotorcraft. In *Proc. of the IEEE Intl. Conf. on Robot. and Autom.*, pages 1349–1356, 2007.
- [25] C. Theodore, D. Rowley, D. Hubbard, A. Ansar, L. Matthies, S. Goldberg, and M. Whalley. Flight trials of a rotorcraft unmanned aerial vehicle landing autonomously at unprepared sites. In *Forum of the American Helicopter Society*, Phoenix, AZ, May 2006.
- [26] P. Tokekar, J. Vander Hook, D. Mulla, and V. Isler. Sensor planning for a symbiotic UAV and UGV system for precision agriculture. Technical Report 13-010, University of Minnesota, Minneapolis, MN, March 2013.
- [27] S. Weiss, M. W. Achtelik, M. Chli, and R. Siegwart. Versatile distributed pose estimation and sensor self-calibration for an autonomous mav. In *Proc. of the IEEE Intl. Conf. on Robot. and Autom.*, pages 31–38, 2012.
- [28] S. Weiss, M. W. Achtelik, S. Lynen, M. C. Achtelik, L. Kneip, M. Chli, and R. Siegwart. Monocular vision for long-term micro aerial vehicle state estimation: A compendium. *J. Field Robot.*, 30(5):803–831, August 2013.