ARTICLE IN PRESS

Robotics and Autonomous Systems ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at SciVerse ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

Autonomous landing at unprepared sites by a full-scale helicopter

Sebastian Scherer*, Lyle Chamberlain, Sanjiv Singh

Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA

ARTICLE INFO

Article history:
Received 23 June 2011
Received in revised form
24 March 2012
Accepted 8 September 2012
Available online xxxx

Keywords:
UAV
Rotorcraft
3D perception
Lidar
Landing zone selection

ABSTRACT

Helicopters are valuable since they can land at unprepared sites; however, current unmanned helicopters are unable to select or validate landing zones (LZs) and approach paths. For operation in unknown terrain it is necessary to assess the safety of a LZ. In this paper, we describe a lidar-based perception system that enables a full-scale autonomous helicopter to identify and land in previously unmapped terrain with no human input.

We describe the problem, real-time algorithms, perception hardware, and results. Our approach has extended the state of the art in terrain assessment by incorporating not only plane fitting, but by also considering factors such as terrain/skid interaction, rotor and tail clearance, wind direction, clear approach/abort paths, and ground paths.

In results from urban and natural environments we were able to successfully classify LZs from point cloud maps. We also present results from 8 successful landing experiments with varying ground clutter and approach directions. The helicopter selected its own landing site, approaches, and then proceeds to land. To our knowledge, these experiments were the first demonstration of a full-scale autonomous helicopter that selected its own landing zones and landed.

© 2012 Elsevier B.V. All rights reserved.

1. Introduction

Safe autonomous flight at low altitude is essential for widespread acceptance of aircraft that must complete missions close to the ground, and such capability is widely sought. For example, search and rescue operations in the setting of a natural disaster allow different vantage points at low altitude. Likewise, UAVs performing reconnaissance for the police, news stations or the military must fly at low altitude in the presence of obstacles. Large rotorcraft can fly at high altitude, but however have to come close to the ground during landing and takeoff. So far unmanned full-scale rotorcraft have to land on prepared land at prepared sites with prior knowledge of obstacle-free trajectories.

Assessing a landing zone (LZ) reliably is essential for safe operation of vertical takeoff and landing (VTOL) aerial vehicles that land at unimproved locations. Currently an UAV operator has to rely on visual assessment to make an approach decision; however, visual information from afar is insufficient to judge slope and detect small obstacles. Prior work in landing site evaluation has modeled LZ quality based on plane fitting, which only partly represents the interaction between vehicle and ground. Additionally, we want to be able to guide an unmanned rotorcraft to a safe landing location.

A NASA study of rotorcraft accidents from 1963 to 1997 found that collisions with objects, hard landings, and roll-overs were responsible for over 36% of the incidents [1]. A partial cause for accidents is a lack of situational awareness of the pilot regarding the suitability of landing sites. Reliable evaluation of landing sites will increase the safety of operation of rotorcraft by presenting vital information to the pilot or unmanned aerial vehicle before a landing approach.

Our algorithmic approach consists of a coarse evaluation based on slope and roughness criteria, a fine evaluation for skid contact, and body clearance of a location. Additionally we consider the ground path to a goal location and approach paths to the landing site. We investigated whether the evaluation is correct for using terrain maps collected from a helicopter and tested landing in unknown environments. This article defines the problem of evaluation, describes our incremental real-time algorithm and information gain map, and discusses the effectiveness of our approach.

We show results from two aircraft-based perception systems. The first is a downward-scanning lidar mounted on a manned EC-135 helicopter. The second system is a 3-D lidar-based perception system that guides the Unmanned Little Bird helicopter in fully-autonomous flight and landings. In results from urban and natural environments from these two different helicopter platforms, we were able to successfully classify and land at LZs from point cloud maps collected on a helicopter in real-time. The presented method enables detailed assessment of LZs from pattern altitude before descent.

* Corresponding author.
E-mail addresses: basti@cs.cmu.edu (S. Scherer), lyle@cs.cmu.edu (L. Chamberlain), ssingh@cs.cmu.edu (S. Singh).

0921-8890/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.robot.2012.09.004

Please cite this article in press as: S. Scherer, et al., Autonomous landing at unprepared sites by a full-scale helicopter, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.09.004

Fig. 1. The problem of finding good landing zones for rotorcraft and landing. A safe landing zone close to a ground goal has to be found by the helicopter. The picture shows the Boeing Unmanned Little Bird Helicopter touching down based on a landing site it selected itself.

The main contributions of this article are

- an analysis of the problem of landing site evaluation for rotorcraft.
- an incremental model-based method to calculate and evaluate landing sites for VTOL vehicles.
- an efficient lidar sensor design that utilizes the sensor for obstacle avoidance and landing site selection.
- results based on lidar sensor data that show LZs found in real environments.
- results of the first full-scale helicopter that selects its own landing sites and lands.

This article first surveys prior work, conveys the problem, our approach, and then presents results for the landing site evaluation and landing (see Fig. 1).

2. Related work

There has been some prior work on landing and landing site selection. A recent survey article by Kendoul [2] describes significant advances for intelligent rotorcraft including landing. From a control perspective the problem has been studied by Sprinkle et al. [3] to determine if a trajectory is still feasible to land. Saripalli and Sukhatme [4] have simulated landing on a moving target that was known to be a good landing site. Barber et al. [5] used optical flow based control to control a fixed wing vehicle to land.

Vision has been a more popular sensor modality for landing site evaluation because the sensor is lightweight. de Wagter and Mulder [6] describe a system that uses vision for control, terrain reconstruction, and tracking. A camera is used to estimate the height above ground for landing in [7], and similarly in [8] a single camera is used to detect and avoid powerlines during landing on a simulated crane setup.

Using a stereo camera, Hintze [9] developed an algorithm to land in unknown environments. Bosch et al. [10] propose an algorithm for monocular images that is based on detecting nonplanar regions to distinguish landing sites from non-landing sites. Another popular approach is to use structure from motion to reconstruct the motion of the camera and to track sparse points that allow reconstruction of the environment and plane fitting. This method has been applied to spacecraft by Johnson et al. [11], Montgomery et al. [12] and applied to rotorcraft by Templeton et al. [13].

Safe landing area detection (SLAD) algorithms have been compared recently by Sanfourche et al. [14]. In their paper a monocular and a LiDAR based approach by Whalley et al. [15] are compared. Another article by Takahashi et al. [16] focuses on benchmarking SLAD algorithms.

Lidar based landing site evaluation has not been studied very deeply except as suggested in [17] where a framework for multiple sources like lidar, radar, and cameras is proposed to infer the suitability of a landing site. For ground vehicles lidar based classification is popular because of its reliability and direct range measurement. It has been used in conjunction with aerial vehicle data for ground vehicles by Sofman et al. [18] and by Hebert and Vandapel [19].

Our work in landing site evaluation uses lidar data and some of the proposed techniques in related work such as plane fitting and roughness for coarse evaluation. However, our method goes beyond pure plane fitting to actually fit a model of the helicopter geometry to a triangulation of the environment and also incorporates further constraints based on approach, abort and ground paths.

3. Problem

The suitability of a landing zone depends to a large degree on the characteristics of the aircraft that will be landing on it. We consider larger vehicles such as manned helicopters; however the problem scales to smaller aircraft. Landing of a helicopter can roughly be separated into two independent phases: approach, and ground contact. During the approach, the helicopter ideally keeps a steady and relatively shallow descent angle ($\sim 6^\circ$) in a straight line to come to a hover at some altitude above the ground. It then orients itself with respect to the ideal location on the ground and descends vertically. For typical landings, approach and ground contact are separated for landing in unimproved terrain. In the particular problem we are considering a ground goal has to be reachable from the LZ so that a human or robot could traverse the last segment.

Assessment of a LZ needs to include two main factors: the ground conditions and the approach conditions. The ground conditions are all factors that are relevant when the aircraft is in contact with the ground, while the approach conditions are related to moving to the LZ.

The ground conditions that need to be considered for assessing an LZ are

- minimum size of the site
- skid contact
- static stability on ground based on the center of gravity of the aircraft
- load bearing capability of the contact surface
- loose foreign objects (FOD) and vegetation surrounding the site
- clearance of aircraft/rotor with surrounding terrain and objects

while the approach conditions are the

- clearance of the path with respect to the terrain
- wind direction
- direction of the abort paths.

The ground path needs to be evaluated for traversability cost and path length.

To ensure a reliable and successful landing it is necessary to consider all these factors in a landing site evaluation algorithm. However our geometric evaluation is insufficient to measure some factors such as the load bearing capability of the terrain or small foreign objects In our approach we assume that a geometrically good site will be able to hold the load of the helicopter and FOD objects will be classified as sufficient roughness to cause a rejection.

4. Approach

The input to our algorithm is a set of 3D points $\{x_1, x_2, \ldots, x_n\} \in \mathbb{R}^3$ that are registered in a global coordinate system. These points

Please cite this article in press as: S. Scherer, et al., Autonomous landing at unprepared sites by a full-scale helicopter, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.09.004

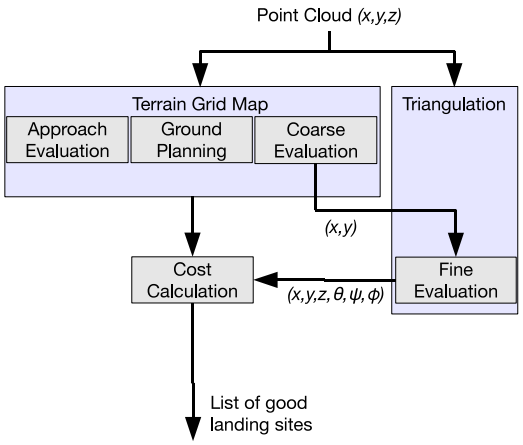


Fig. 2. The steps in our landing site evaluation algorithm. The input is a list of globally registered points. The output is a sorted list of good landing sites that include location as well as landing heading. Two representations of the environment are created. One representation is based on a terrain grid map and the other is based on a triangulation of the environment and is compared to a 3D model of the helicopter.

could be generated by a lidar range scanner and registered using an inertial navigation system with GPS, for example. Additionally the current wind direction θ_{wind} is useful in choosing an approach into the wind as it will avoid crosswind and reduce the required groundspeed. Fig. 2 shows the data flow in our approach where the raw measurements are processed to create a regular grid for the coarse evaluation of potential landing areas, and a separate triangulation is used for the fine evaluation. Any updated grid cell is reevaluated to reflect changes in the available landing sites. A list of the available landing sites can be displayed to an operator or can be transmitted to the guidance system of an autonomous vehicle. A touchdown point is defined by a 3D position, an orientation, an approach and abort path direction, and a ground path. Additionally we calculated a map of good places to search based on the sensor data already collected.

4.1. Coarse evaluation

The first step in our algorithm is a coarse evaluation of a regular terrain grid map. Each cell in the map, roughly the size of the landing gear, tests how the vehicle would come into contact with the ground. The map keeps a set of statistics in a two dimensional hash table of cells. The hash key is calculated based on the northing–easting coordinate at a fixed resolution. This provides a naturally scrolling implementation and cells can be set as data are measured. As the vehicle moves and the map scrolls, old cells will be overwritten. This data structure allows the algorithm to quickly add new data to the map. Each cell in the grid keeps a set of statistics: The mean, minimum, and maximum z-value, the number of points, and the moments necessary to calculate the plane fit (see Fig. 3).

The statistics accumulated in each terrain cell are evaluated to assess a particular location. Based on these tests a series of binary attributes and goodness values can be computed. The statistics are accumulated with neighboring cells which allow us to compute the suitability of larger regions. For each cell we consider a region that is the size of the vehicle with the rotor disc. The binary tests are designed to reject places where the ground is too rough or too step to be valid; furthermore, the algorithm cannot make a reliable estimate of a landing site, if not enough data is available.

The binary attributes are computed as follows. First, we check if enough points are available to perform a plane fit. We require at least 15 points per cell, which is an average density of one point every 20 cm for a 3 m cell. This average density is the required density for the smallest obstacle we consider; however, we make an optimistic assumption since points are not necessarily distributed evenly. Next, we consider the standard deviation of the z-value. The algorithm rejects cells if the vertical standard deviation is larger than 50 cm, rejecting any area with large altitude changes. The value has to be large enough to avoid interfering with the results of the plane fitting since a large change in altitude also characterizes a large slope. Finally, the slope and residual of a plane fit are checked for a planar surface. The residual will be a good indicator for the roughness of the terrain since it measures deviation from the plane. Currently the slope threshold is set at 5° and the residual threshold is set at 4.

The mean altitude and standard deviation of the plane is calculated based on Welford's algorithm by keeping the mean and the corrected sum of squares [20]. After receiving a new altitude measurement in a patch the mean altitude μ_c and sum of squares S_c are updated:

$$\mu_{\text{new}} = \mu_{c-1} + (z - \mu_{c-1})/n \tag{1}$$

$$S_c = S_{c-1} + (z - \mu_{c-1}) \cdot (z - \mu_{\text{new}}) \tag{2}$$

$$\mu_c = \mu_{\text{new}}.$$

The standard deviation σ can be calculated from S as follows:

$$\sigma_c = \sqrt{\frac{S_c}{c}}. \tag{3}$$

Additionally we want to evaluate the slope and need to fit a plane with respect to the measurements. A fast algorithm to calculate a least squares plane fit is a moment based calculation. The fit is a least squares fit along z and therefore assumes no errors in x and y exist (also see [21]). We only have to keep track of the moments

$$M = \begin{pmatrix} M_{xx} & M_{xy} & M_{xz} \\ M_{xy} & M_{yy} & M_{yz} \\ M_{xz} & M_{yz} & M_{zz} \end{pmatrix} \tag{4}$$

and the offset vector

$$M_o = \begin{pmatrix} M_x \\ M_y \\ M_z \end{pmatrix} \tag{5}$$

and the number of total points c. The moments can be updated for a new measurement (x, y, z) by updating the moments starting from 0 for all initial values:

$$M_x = M_x + x, \qquad M_y = M_y + y, \qquad M_z = M_z + z \tag{6}$$

$$M_{xx} = M_{xx} + x^2, \qquad M_{yy} = M_{yy} + y^2, \qquad M_{zz} = M_{zz} + z^2 \tag{7}$$

$$M_{xy} = M_{xy} + x \cdot y, \qquad M_{xz} = M_{xz} + x \cdot z, \qquad M_{yz} = M_{yz} + y \cdot z \tag{8}$$

$$c = c + 1. \tag{9}$$

Consequently, an update is the constant time required to update the 10 values and the plane fit is calculated in closed form. Another advantage is that we can accumulate the moments from the neighboring cells to fit a larger region in the terrain that represents the whole area of the helicopter, not just the landing skids. See [22] for more details on moment based fitting. The plane normal $n = \begin{pmatrix} A \\ B \\ C \end{pmatrix}$ and residual r are calculated as such Eqs. (10)–(14) are given in Box I.

We can infer the slope and roughness based on the normal and residual. The slope can be calculated as follows

$$\alpha = \arccos(v_{up}^T n) \tag{15}$$

Please cite this article in press as: S. Scherer, et al., Autonomous landing at unprepared sites by a full-scale helicopter, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.09.004

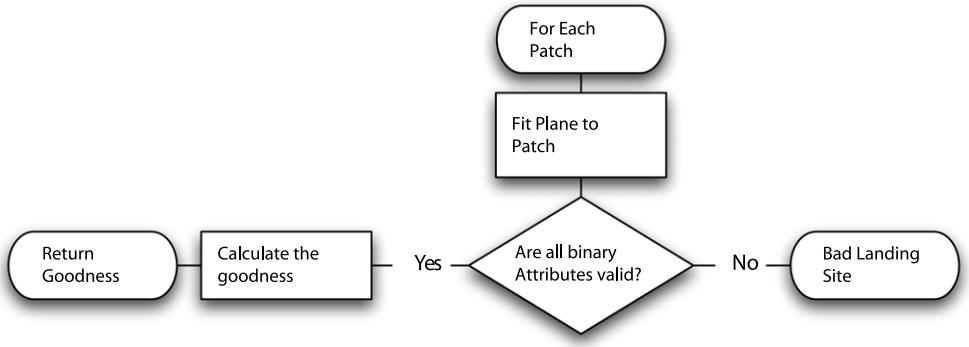


Fig. 3. A flow chart showing the control flow of evaluating a patch in the terrain grid map. Only if all binary attributes shown in Table 1 are valid a landing zone will be considered.

Table 1
The binary attributes considered in the coarse evaluation for landing site evaluation. Only if all operations evaluate to true can a landing site be available.

#	Description	Symbol	Operator	Value
1	Do we have more than x points for a plane fit?	c	$>$	15
2	Is the spread of the points less than x cm?	σ_n	$<$	50
3	Was the plane fit successful?		$=$	True
4	Is the residual for the plane fit less than x?	r	$<$	4
5	Is the slope of the plane less than x degrees?	α	$<$	5

$$M_n = M_{xy}^2 c - 2 \cdot M_x M_{xy} M_y, + M_x^2 M_{yy} + (M_y^2 - c M_{yy}) M_{xx} \tag{10}$$

$$A = (M_y^2 M_{xz} - (M_x M_{yz} + M_{xy} M_z) M_y + c(-M_{xz} M_{yy} + M_{xy} M_{yz}) + X_x M_{YY} M_z)/M_n \tag{11}$$

$$B = (-(M_{xy} M_z + M_{xz} M_y) M_x + c M_{xy} M_{xz} + M_x^2 M_{yz} + M_{xx}(-c M_{yz} + M_y M_z))/M_n \tag{12}$$

$$C = (-(M_x M_{yz} + M_{xz} M_y) M_{xy} + M_x M_{xz} M_{yy} + M_{xy}^2 M_z + M_{xx}(M_Y M_{yz} - M_{yy} M_z))/M_n \tag{13}$$

$$r = \sqrt{\left| C^2 c + 2ACM_x + A^2 M_{xx} + 2BCM_x + 2ABM_{xy} + B^2 M_{YY} - 2CM_z - 2AM_{xz} - 2BM_{yz} + M_{zz} \right| / c}. \tag{14}$$

Box I.

where $v_{up} = (0, 0, -1)$ is the up vector (in north-east-down coordinates) and n is the plane normal.

If all the binary attributes shown in Table 1 pass, the algorithm calculates the goodness attributes used in the final evaluation. All potentially good landing sites are now evaluated with the fine evaluation to decide if touching down is feasible.

4.2. Fine evaluation

Once we have determined a list of general locations, we finely evaluate each potentially good site with respect to a 2D Delaunay triangulation of all available data. A Delaunay triangulation is a triangular network of points that can be constructed efficiently (see [23] for more details). The reason for this staged processing is that it is easier to know when to incrementally update a cell-based representation, and the incremental calculation for the coarse grid is much faster than for the triangulation.

While the coarse evaluation considers the plane geometry and roughness of a landing site, the triangulation tries to simulate the helicopter static contact with the surface. In our experiments with a full scale unmanned helicopter it was possible to control a touch down to within about 30 cm even in crosswind. Therefore we assume in this evaluation that a precision landing of the vehicle will be possible. As a first step we incrementally update a hierarchical triangulation with a maximum number of points per patch area. For the implementation we use CGAL (see [24]). If the point density is exceeded, we randomly remove points from the triangulation to keep below a maximum amount of memory used. Currently, the patch size is set to 3 m and the maximum point density is set to 200 points/cell, which corresponds to an even point spacing of approximately 20 cm.

Once the coarse evaluation calculates a list of potential (x, y) locations, the fine evaluation tries to find the best orientation and determines if the chosen location is feasible. We test a series of possible orientations θ of the helicopter (8 in our experiments) by intersecting the virtual landing gear with the measured terrain to determine what the resulting pose of the helicopter would be (also see Fig. 4). Currently our algorithm only works for two landing skids, which is a very common configuration of landing gear.

First we determine the intersections of the skids with the triangulation by walking along the two lines representing the landing skids, and extract the height above ground as shown in Fig. 5. Now we assume that the helicopter's center of gravity is on top of the landing skids and calculate the two highest points of the intersection. These two points will be the points that determine the position of the skid in the static case. We repeat this process for all eight potential landing gear positions.

The algorithm to position the landing gear is as follows. First we determine all intersection points

$$p_i^l = (x_i^l, y_i^l, z_i^l) \tag{16}$$

where

$$p_i^l \in \mathbb{R}^3, \quad i \in \{1, \ldots, n\} \tag{17}$$

between the landing gear l and the triangulation and partition the points into two sets: Points aft, a_i^l, and in front, f_i^l, of a distance along

Please cite this article in press as: S. Scherer, et al., Autonomous landing at unprepared sites by a full-scale helicopter, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.09.004

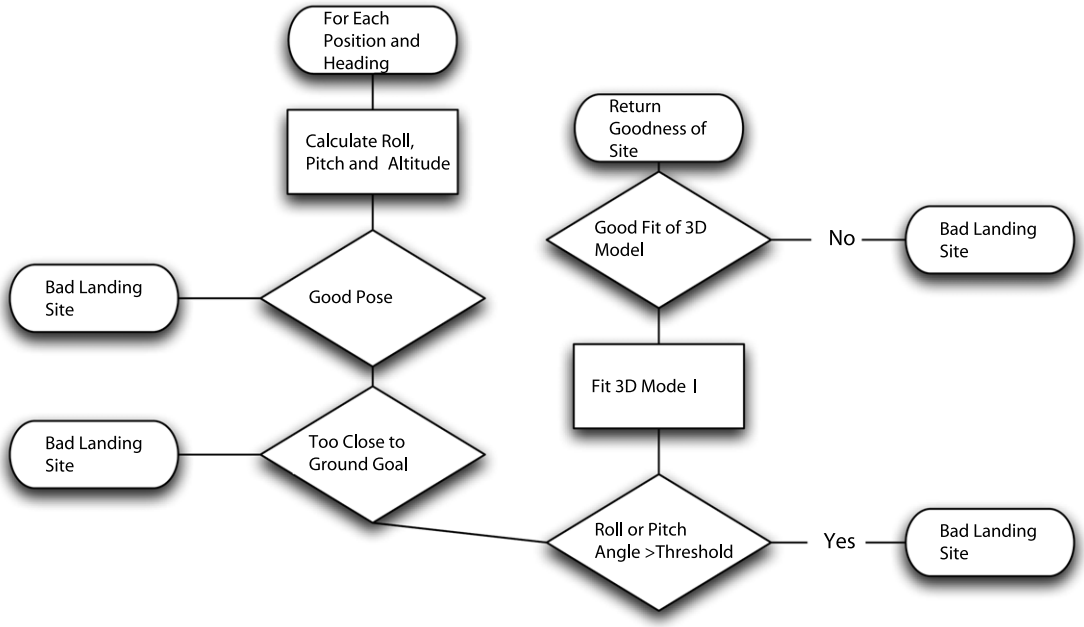


Fig. 4. The flow chart of the fine evaluation algorithm. The pose of the helicopter is calculated based on a (x, y, θ) position and heading. If the pose is calculated successfully a surface model of the aerial vehicle is evaluated.

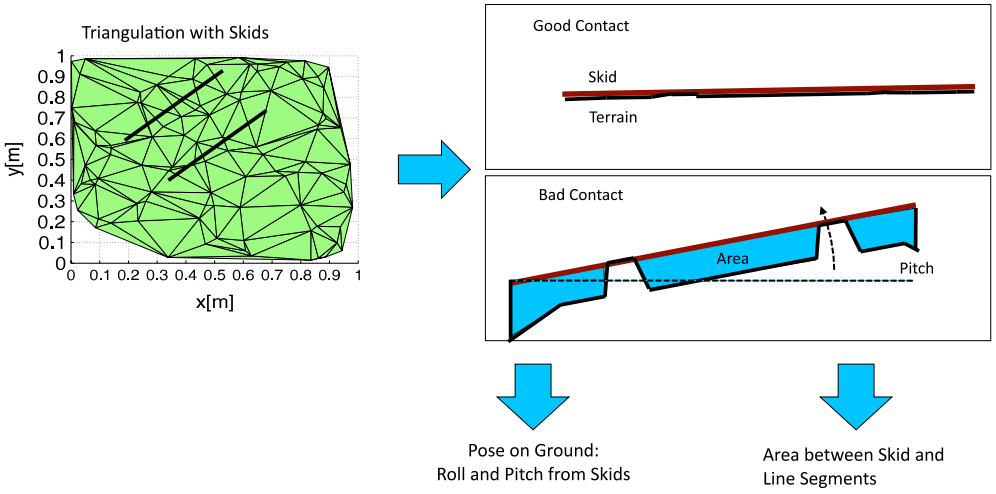


Fig. 5. Illustration of skid-triangulation contact. The skids in contact with the triangulation determine the roll and pitch of the vehicle. A good contact is characterized by a small angle and small area between the skid and triangulation.

the landing gear. Currently we set the distance to be half the length of the landing gear because we assume that the center of mass is centered on top of the landing gear; however. this position can be changed depending on the mass distribution.

Next we look at the z-values and find the maximum z-value in the first and in the second half of the landing skid for each side:

$$p^l_{\text{aft}} = \text{argmax}(a^l_i) \tag{18}$$

$$p^l_{\text{front}} = \text{argmax}(f^l_i). \tag{19}$$

A minimum distance between p_{aft} and p_{front} is set to ensure that the slope will fit to a stable value. Now that we know how the landing gear will be positioned, we calculate the area between the line of the skid and the triangulation to get a measure of contact area. For example, if we were to land centered on railroad tracks at an 90° angle our skids would rest on the rails because they are the highest points above the ground. In this case even though the roll, pitch, and roughness will be small (rails are very small deviations from the ground and usually level) the integral of the area under the landing skids will have a large value because most of the landing skid is elevated above the ground. For skids in full contact with the ground the integral would be zero.

The next step is to determine the roll and pitch of the helicopter when it is in contact with the ground. We use the center of the two landing skid lines to determine the roll angle. Finally we use the lower landing gear line to calculate pitch because the center of gravity will shift towards that line. Using this algorithm we are now able to predict the position and orientation of the helicopter when it is in contact with the ground. The roll angle α is calculated from the interpolated centers p^l_c of the skids

$$p^l_c = \frac{p^l_n - p^l_1}{2} \tag{20}$$

$$\alpha = \arctan 2\left(z^2_c - z^1_c, \sqrt{(x^2_c - x^1_c)^2 + (y^2_c - y^1_c)^2}\right). \tag{21}$$

The pitch of the helicopter depends on the roll angle since more weight will shift towards the lower skid. The pitch angle β

Please cite this article in press as: S. Scherer, et al., Autonomous landing at unprepared sites by a full-scale helicopter, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.09.004

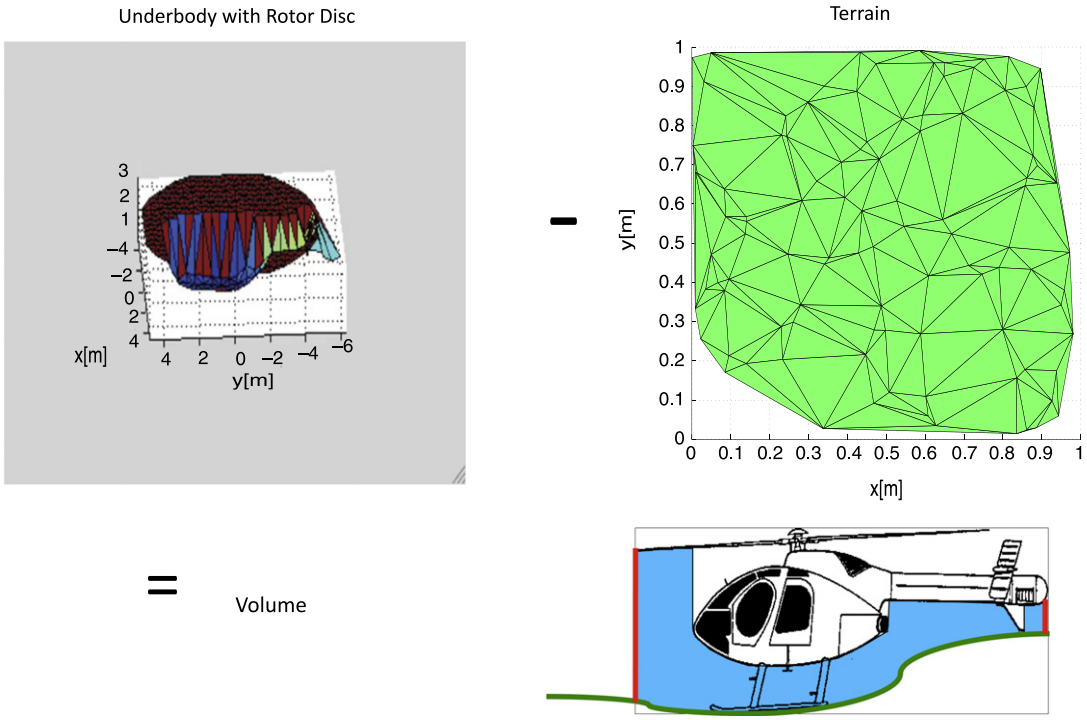


Fig. 6. Underbody volume between model and triangulation. The underbody model is positioned with the pose calculated in Fig. 5. The volume between the body surface (top-left) and triangulation (top-right) is used to check for collision and calculate a goodness measure of a landing site (bottom-right).

is calculated as

$$\lambda = \begin{cases} 1 & \text{if } \theta_{\text{roll}} > 0 \\ 2 & \text{otherwise} \end{cases} \tag{22}$$

$$\beta = \arctan 2\left(z^{\lambda}_{\text{front}} - z^{\lambda}_{\text{aft}}, \sqrt{(x^{\lambda}_{\text{front}} - x^{\lambda}_{\text{aft}})^2 + (y^{\lambda}_{\text{front}} - y^{\lambda}_{\text{aft}})^2}\right) \tag{23}$$

with the orientation and position, we now calculate the volume between the undersurface of a 3D model of the helicopter with the triangulation of the terrain as shown in Fig. 6. This volume allows us to predict if a tail strike or similar bad contact with the ground would occur given the predicted position and orientation of the helicopter. It also gives us a measure of goodness that will maximize the volume under the helicopter. The calculation of the volume is simplified by taking a regular sample of points on the surface of the helicopter at an interval e. Even though some errors can occur we can over-approximate this regular sampling when we create the model. Given the sampled surface as a set of points on the underbody of the helicopter p_s, we first have to transform the points given the pose of the helicopter p^h_c and the orientation (α, β, θ). The transformed points p'_s are then compared to the intersected point p_t of the triangulation. The volume is defined as

$$v_j = \begin{cases} e^2(z'_s - z_t) & \text{if } z'_s - z_t > 0 \\ -\infty & \text{otherwise} \end{cases} \tag{24}$$

$$V_{\text{underbody}} = \sum_{j=0}^{m} v_j. \tag{25}$$

If the underbody surface intersects with the triangulation, the volume becomes $-\infty$ because we have a collision with the terrain.

4.3. Approach and abort path evaluation

Once we have all the measures related to the goodness of landing sites we also need to verify that a landing approach is feasible. We therefore utilize the coarse terrain evaluation to check a linear landing approach path at the desired glide slope and final height above ground for the clearance of obstacles. Additionally the algorithm also checks that an abort path that deviates by at most 45° from the approach is obstacle free. This is an added measure of safety because we can abort an approach at any time by following the abort path from the current altitude.

The approach path is defined to start at some altitude a above ground with a glide slope κ and end at a final hover point. However we also want to ensure that there is some margin for the waypoint controller to converge onto the approach path and glide slope. Typically one can just expand the obstacles by a constant amount to keep a buffer. However since at the end of the glide path we want to be very close to the ground (1–3 m) this approach will not work. Instead the glide slope is offset by a lower glide slope $\kappa_{C-\text{space}} < \kappa$. This lower glide slope provides a one-sided funnel for the path controller to converge on the final trajectory.

From these parameters we can define the path as a line segment starting at p_{start} going to p_{hover}. Let the distance of a point p_k be given by the Euclidean distances. Additionally depending on the vehicle and control accuracy a lateral safety-buffer is defined as b_{lateral} and a vertical safety-buffer is defined as $0 < b_{\text{vertical}} < a$. Now all cells on the approach path that are in the range of the vertical safety buffer $c_k \in \{1, \ldots, o\}$ are checked.

The simple volume as a goodness measure of an approach path places an equal cost independent of the distance clearance from obstacles. Therefore the cost of being close to an approach path decreases exponentially as the distance increases:

$$b'_{\text{lateral}} = \min(b_{\text{lateral}}, \text{dist}(p_{\text{hover}}, p_k)\sin(\kappa_{C-\text{space}})) \tag{26}$$

$$c_k = \begin{cases} \exp(-|z_{c_k} - z_k|) & \text{if } (z_{c_k} - z_k - b'_{\text{lateral}}) > 0 \\ \infty & \text{otherwise} \end{cases} \tag{27}$$

$$C_{\text{approach}} = \sum_{k=1}^{o} c_k. \tag{28}$$

The lateral buffer is reduced once the helicopter comes close to the hover point because a collision with the terrain would be

Please cite this article in press as: S. Scherer, et al., Autonomous landing at unprepared sites by a full-scale helicopter, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.09.004

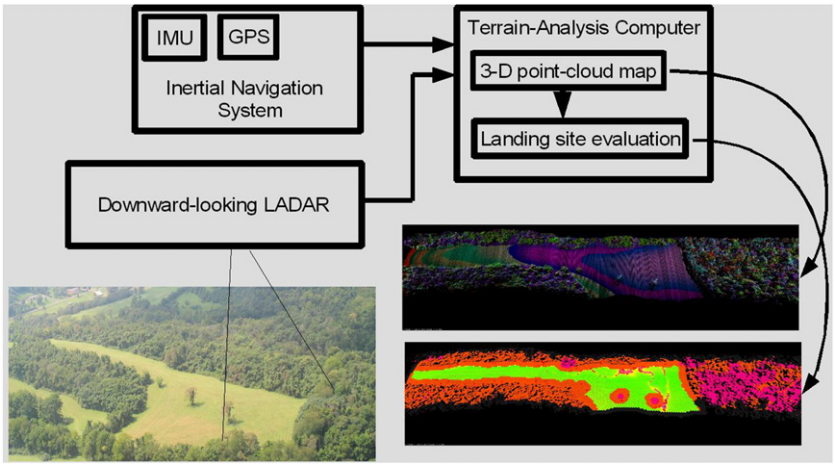


Fig. 7. Data flow for landing site evaluation. The INS/GPS system and lidar is registered to created a point cloud. The point cloud is evaluated to produce a landing zone evaluation.

Table 2
A linear combination of factors determines the goodness of a landing site. The weights need to be set or learned based on user preference to determine which of the acceptable sites is the best.

Description	Function	Weight
Goodness from rough evaluation	$w_s\alpha + w_r r$	ω_1
Area under the landing gears	$\frac{1}{A_{\text{gear}}}$	ω_2
Volume under the helicopter	V_{heli}	ω_3
Distance from the closest bad landing site	d_l	ω_4
Distance to the ground goal point	d_g	ω_5
Cost of the approach and abort path	C_{approach}	ω_6
Cost of the ground path	C_{ground}	ω_7

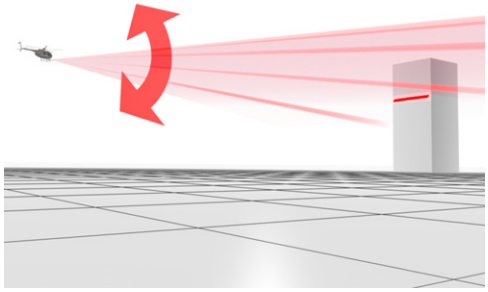


Fig. 8. Obstacle-detection scan configuration. Scanner faces forward and nods up and down to detect obstacles as the aircraft descends to lower altitudes.

detected otherwise. The approach path is checked for a set of headings and the goodness is based on the risk of the approach and abort path.

4.4. Ground path planning

The final factor in landing site evaluation is the path length from the goal location to the landing site on the ground. This planning is performed assuming that a human or robot has to be able to reach the goal location from our landing location. This cost-to-go is another factor for our final cost function.

It is important that the traversability planner is robust to missing and some wrong cost estimates since the plan dictates if a site is reachable and will even be considered for landing. For instance, a skipped lidar packet at a pinch point will cause a hole in the terrain map making any other landing site inaccessible.

The cost of traversability is therefore continuous with optimistic binary thresholds for non-traversability and optimistic estimates for missing data. The cost function is estimated from the

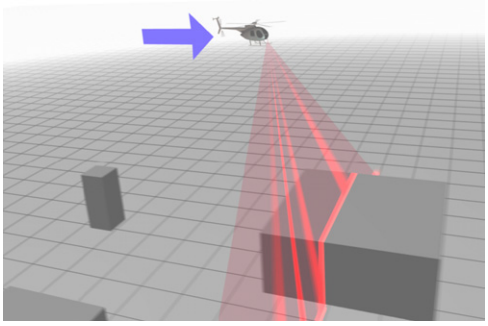


Fig. 9. Landing zone survey scan configuration. Planar scanning sweeps terrain as the aircraft overflies potential landing zones from a safe altitude.

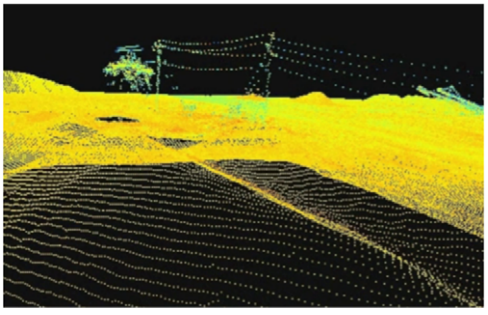


Fig. 10. Example point cloud showing terrain and power lines.

Fig. 11. Close-up of the sensor head mounted on the nose of the Unmanned Little Bird.

Please cite this article in press as: S. Scherer, et al., Autonomous landing at unprepared sites by a full-scale helicopter, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.09.004

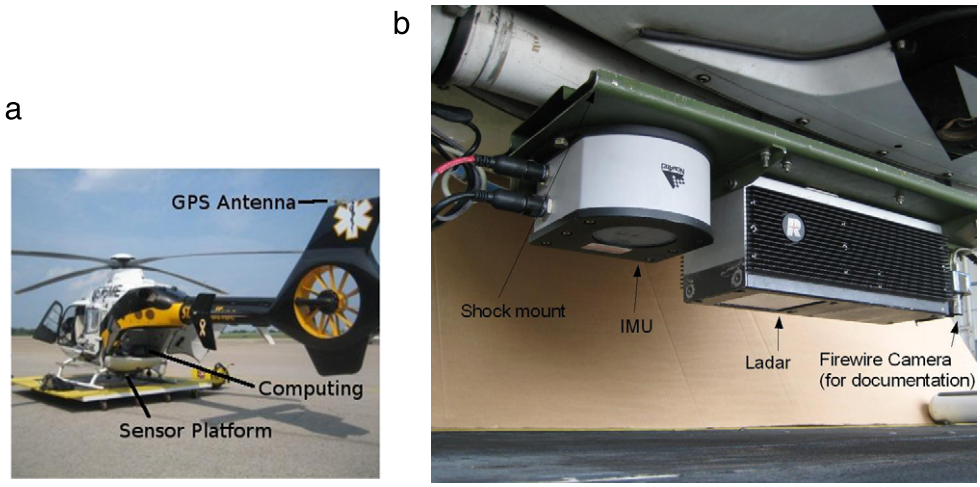


Fig. 12. The experimental setup for data collection on a EC 135 helicopter in Pittsburgh, PA, USA. We mounted a lidar, and inertial navigation system on a helicopter as shown in (a). The sensor platform is shown in (b). The inertial measurement unit was mounted next to the lidar to minimize errors in the registration of the data.

standard deviation σ_c and the altitude difference between the mean of neighboring cells $\Delta\mu_c = \max_{n\in\text{Neighbors}(c)}(|\mu_c - \mu_n|)$. The cost is calculated as follows

$$C_{\text{ground}} = \begin{cases} \max(\sigma_{\text{scale}}(\sigma_c - \sigma_{\min})), & \Delta\mu_{\text{scale}}(\Delta\mu_c - \Delta\mu_{\min}) \\ \infty & \text{if Known Obstacle.} \end{cases} \quad (29)$$

The cost-to-go is then repeatedly computed using a dynamic-programming wavefront algorithm [25] from the ground goal location p_{goal} up to a maximum cost-to-go that we consider. In very rough terrain or with lots of missing cells only short paths will be generated while on smooth terrain such as roads long paths will be found which represents the time to traverse. The cumulative cost is then used as another factor in the goodness evaluation.

4.5. Goodness assessment

Landing site selection is a problem where many factors need to be considered to evaluate the cost of a site. Depending on the weights for the goodness calculation, different sites will be chosen as best sites. A linear combination of the factors described in Table 2 is considered to calculate the goodness of a landing site. Currently we manually set the weights $\omega_1, \ldots, \omega_7$ to determine a goodness.

4.6. Computational efficiency

A naive online implementation of the algorithms above will be infeasible to evaluate in real-time. We parallelize and incrementally compute updates to enable a real-time implementation. The following techniques enable real-time computation on a multi-core processor:

- All trivial parallel evaluations are spread across multiple cores. This includes evaluating
  - the fine evaluation of a set of sites.
  - the detection if any landing approaches need to be recomputed.
  - a series of landing approach directions.
- A triangulation is only created if a fine evaluation is performed in that region.
- A fine evaluation is only performed if any of the rough terrain cells change.
- Fine evaluations are performed radially outward from the ground goal location p_{goal}. As computation time permits landing sites further and further away will be checked.
- An approach path is only evaluated if the terrain cells related to it have changed.
- Only a fixed number of approach and abort directions are evaluated.

Our current implementation will use as many cores as are available and can use those cores efficiently. Using this approach we are able to calculate landing sites in real-time at normal helicopter speeds.

4.7. Perception system design

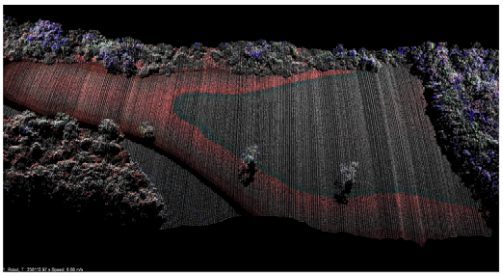
The sensing system for an unmanned platform operating in an unmapped environment must provide two types of information: terrain information for analysis of potential landing zones, and proximity information about obstacles it may encounter en-route. Many sensing modalities are available for both tasks, with varying suitability for each task. The driving requirement in our case was high-resolution measurement of the terrain while flying at a reasonable speed and altitude for a full-scale helicopter. This terrain measurement requirement rules out radar and Machine Vision. A radar sensor with sufficient angular resolution would require a large antenna, and the depth and angular resolution of vision systems is inadequate except at close range. Recent advances in scanning lidar technology have improved the range and bandwidth significantly, opening this modality as a prime candidate for a viable perception system on a full-scale helicopter (see Figs. 7 and 11).

We designed and built a 3-D scanning lidar that operates in two modes: forward-scanning for obstacle detection during low-altitude flight (Fig. 8), and downward scanning for terrain mapping and landing zone search from a higher altitude (Fig. 9). The scanner operates in two axes: a fast 100 Hz, 100° transverse scan and a slower "nodding" scan that cycles up and down. The fast-axis scan is provided off-the-shelf by the laser manufacturer, while the slow-axis is a single-axis gimbal that we fabricated in-house. The lidar measures with a sampling frequency of 84 kHz, giving the point density for a high-resolution terrain map. The sensor has a range of 150 m, centimeter accuracy, and can process multiple returns for obscurant and vegetation penetration. The laser is Class 1 and eye-safe and more details about the specific model can be found in Section 5.2.1.

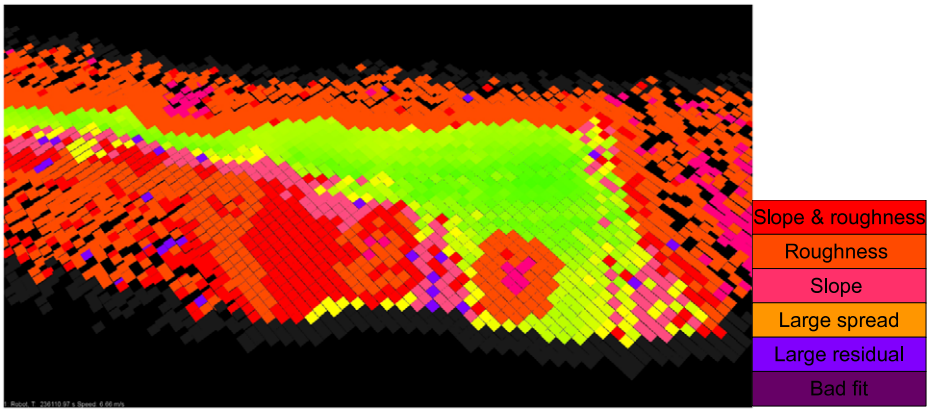
A critical parameter for continuous landing zone evaluation in a overflight configuration is the ability to receive a sufficient density of points to detect the smallest relevant obstacle ($d_{\text{smallestobst}}$). Given a lidar performance these parameters set speed (s) and altitude limits (a) to achieve a sufficient point resolution on the

Please cite this article in press as: S. Scherer, et al., Autonomous landing at unprepared sites by a full-scale helicopter, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.09.004

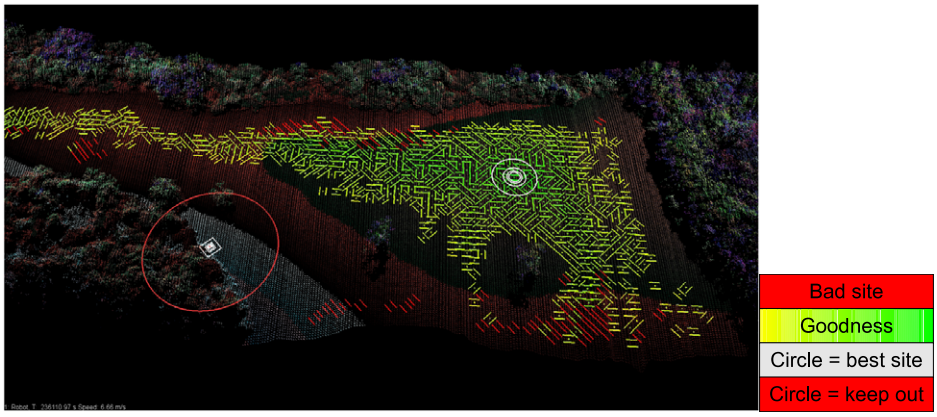
(a) Picture.



(b) Input point cloud.



(c) Coarse evaluation.



(d) Fine evaluation.

Fig. 13. Results from the "three trees" area. (a) shows an aerial view of the scanned area and (b) shows the point cloud at the end of the scan.

ground to achieve detection. The speed is given by the scan frequency:

$$s = d_{\text{smallestobst}} \cdot f.$$

The altitude depends on the field of view and beam divergence

$$a = \min\left(d_{\max}, \frac{d_{\text{smallestobst}}}{2 \cdot \alpha_{\text{separation}} - \alpha_{\text{div}}}\right)$$

and the point separation $\alpha_{\text{separation}}$ is a function of the number of points given the measurements in a field of view:

$$\alpha_{\text{separation}} = \frac{f \cdot \alpha}{p}.$$

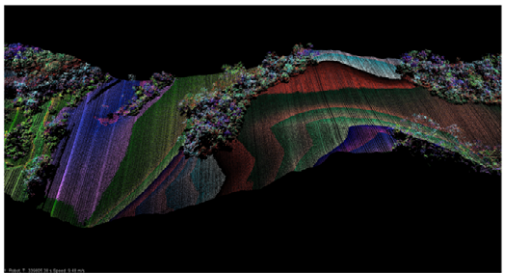
For the analysis we used the following parameters $f = 100$ Hz, $p = 85\,000$ Hz, $d_{\max} = 150$ m, $\alpha = 100°$, $\alpha_{\text{div}} = 0.0171°$, and $d_{\text{smallestobst}} = 0.15$ m. With these parameter we can fly at a maximum height of 58 m for a 15 cm obstacle and a speed of 20 m/s (39 knots). This analysis assumes that there will only be one overflight.

The motion control for actuation of both axes is precisely synchronized to a global timeframe so that for each measurement, the time is known and can be fused with INS data for point registration. Absolute encoders and a highly rigid frame allow the pointing angles to be known to $< 0.01°$.

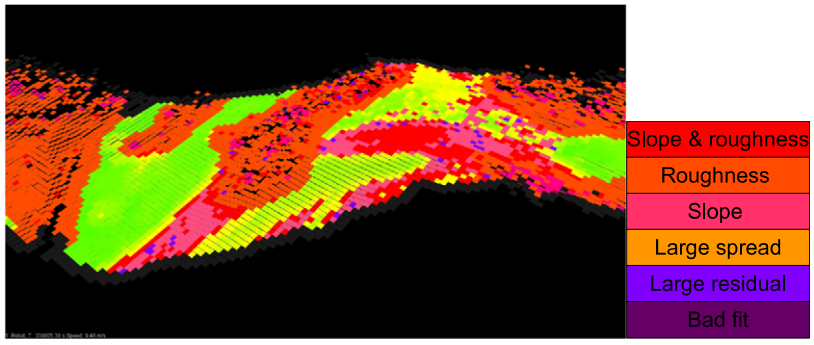
The lidar is rigidly coupled to a high-accuracy GPS/INS system so that the attitude of the whole sensor head is precisely known at all times despite vibration. Otherwise, the map will become "blurred" due to errors in the measurement of where the instrument was pointed when each point was measured. Each measured point is registered into a global reference frame. The INS uses a ring-laser gyro with $< 10°/\text{h}$ drift, differential GPS with 1 cm accuracy, and a Kalman filter that timestamps the measurements into a timeframe that is synchronized with the rest of the perception and computing system.

Please cite this article in press as: S. Scherer, et al., Autonomous landing at unprepared sites by a full-scale helicopter, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.09.004

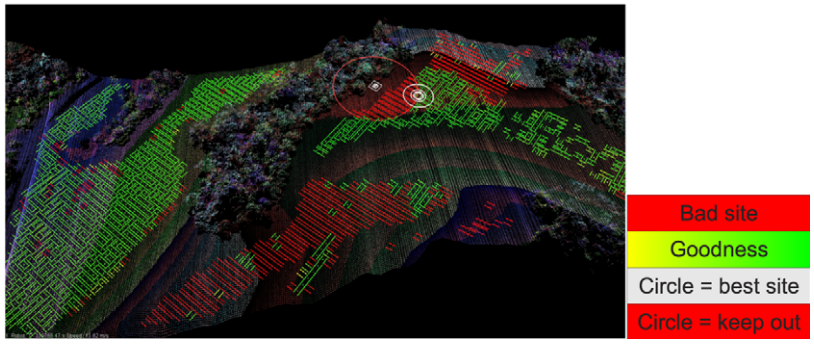
(a) Picture.



(b) Input point cloud.



(c) Coarse evaluation.



(d) Fine evaluation.

Fig. 14. Results from the "sloped field" area.

The system was shown to be capable of detecting chain link fences, wires (Fig. 10), and 4 in flat pallets (Fig. 19(d)) during flight on a helicopter. Maps built without differential GPS are still locally consistent after one map-building run, but possibly inconsistent over time. In our experiments, we operate with 1-centimeter differential GPS, but the maps would still have had sufficient local consistency to pick up small obstacles, even if their actual registered position was slightly off. This sensor was used in the experiments in Section 5.2.

Even though the inertial navigation system we are using has a high accuracy it is not sufficient to register the points with high enough accuracy to evaluate landing sites in front of the helicopter at the glide slope, because the smallest potentially lethal obstacle we consider is the height of a rail (15 cm). One could apply scan matching techniques such as the one proposed by Thrun et al. [26] to smooth the matching of adjacent scans.

5. Experiments

We present two experimental results in this section. The first set of results evaluates only the coarse and fine landing site evaluation algorithm in a variety of different urban and vegetated environments. The second set of results demonstrates the autonomous landing of the Boeing Unmanned Little Bird (ULB) helicopter in an unknown environment that includes ground path planning, and approach path generation.

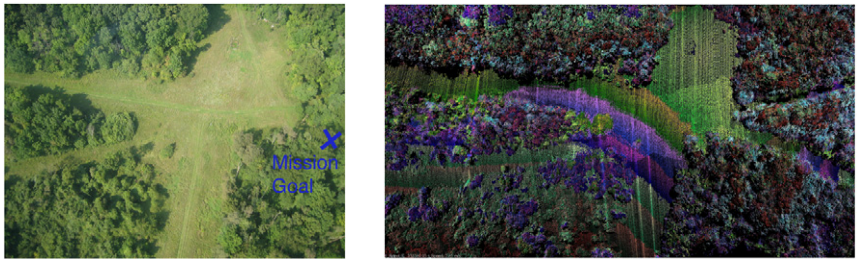
5.1. Fine and coarse landing zone evaluation

In this section we show results for the coarse and fine evaluation based on 3D point cloud datasets we collected. In these results we assume a straight line path from the landing site to a ground goal location. The results are shown for natural and man-made environments to illustrate several environments and varying cost weights. For all the results in this section we used the same parameter values for all experiments except in the case where we intentionally varied the values to demonstrate the influence on the site chosen.

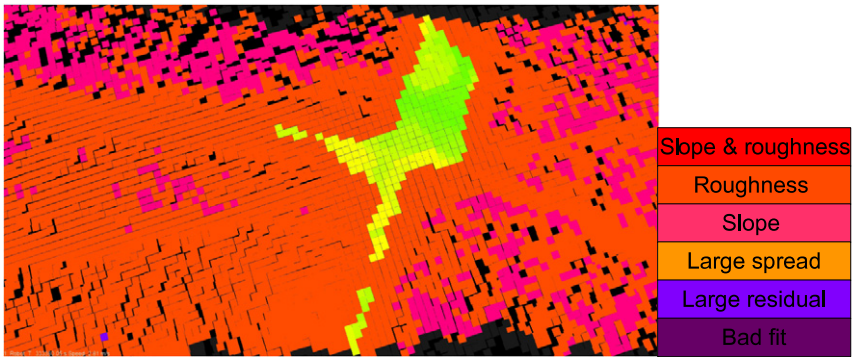
5.1.1. Setup

For our first experiment we collected point cloud datasets at various locations around Pittsburgh, PA, USA. We overflew the areas in approximately straight lines with speeds between 10 and 20 knots. On the bottom of the helicopter we mounted a laser scanner, inertial navigation system, and a documentation camera

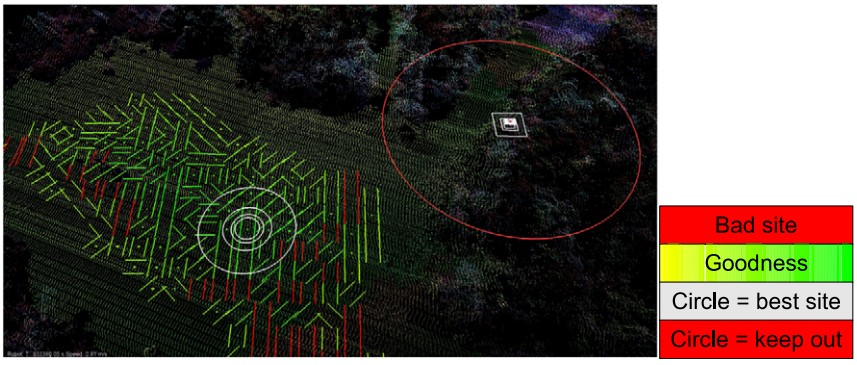
Please cite this article in press as: S. Scherer, et al., Autonomous landing at unprepared sites by a full-scale helicopter, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.09.004



(a) Picture. (b) Input point cloud.



(c) Coarse evaluation.



(d) Fine evaluation.

Fig. 15. Results from the "wooded intersection" area.

(see Fig. 12). The Riegl LMS-Q140i-60 scans a line perpendicular to the flight direction at 8000 points per second. The scan frequency of the lidar is 60 Hz. The data are fused by a Novatel SPAN INS/GPS to calculate a globally registered point cloud. The inertial measurement unit has a drift rate of 10°/h (0.02° reported accuracy) and the GPS is corrected with a base station to a reported 1 cm accuracy. In total we collected data on 9 sites with a flight time of 8 h. We discuss results from three representative sites.

The coarse evaluation is robust to errors in z since it performs a least squares fit to determine the slope. The fine evaluation relies on a good point cloud registration and is not robust to misregistered points. Nevertheless, other methods exist to improve point cloud registration to get a better input point cloud from lower performance INS systems.

5.1.2. Results

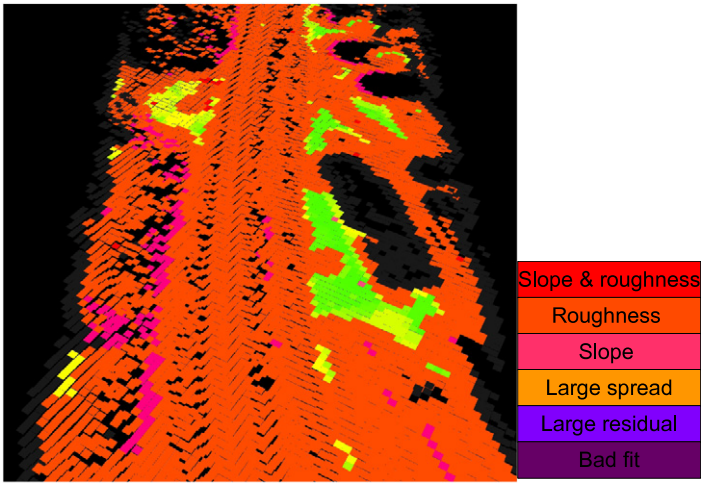
The first site was scanned at an altitude of about 100 m (Fig. 13). It is a field, and our ground goal point is located next to the slope on the hill (see Fig. 13(d) for exact goal location). The result shows the coarse and fine evaluation of landing sites in the environment. Several possibly good landing sites in the coarse evaluation are rejected in the fine evaluation phase. Finally, the best landing site is picked in a spot with small slope and a large clearance from obstacles.

In the "sloped field" dataset (Fig. 14) we have a slightly more complex scenario because a lot of landing sites exist and we must choose a best one. Also a large number of smooth slopes need to be rejected based on the plane fitting. The next dataset (Fig. 15) is very constrained and a possible landing site needs to be chosen that still fulfills all the constraints. We also scanned this dataset at an altitude of about 100 m. Just based on the coarse evaluation one can see that most points are not suitable and only a small area exists that can be considered suitable for landing. In the fine evaluation some too optimistic locations from the coarse evaluation are rejected. The main reason for the rejection is that the landing gears cannot be placed successfully. After evaluating the cost a point in the middle of the open area that is closer to the ground goal location is picked.

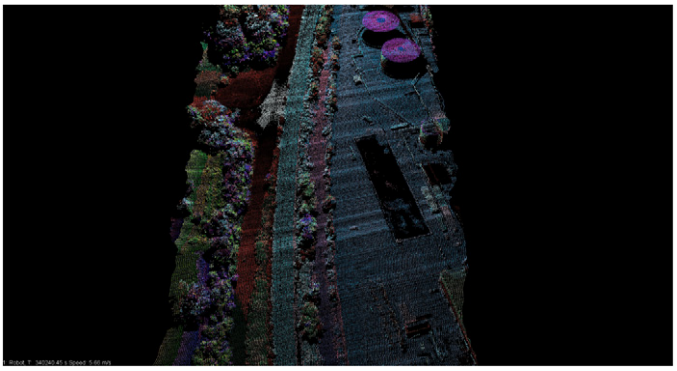
While the previous dataset was mostly in vegetated terrain we also collected a dataset at a coal power plant (Fig. 16) at 150 m elevation above ground. The geometry of obstacles at the power plant is very different from vegetation, but the algorithm is still able to determine good landing sites. The ground goal location is close to a car, and one can see how the fine evaluation rejects the

Please cite this article in press as: S. Scherer, et al., Autonomous landing at unprepared sites by a full-scale helicopter, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.09.004

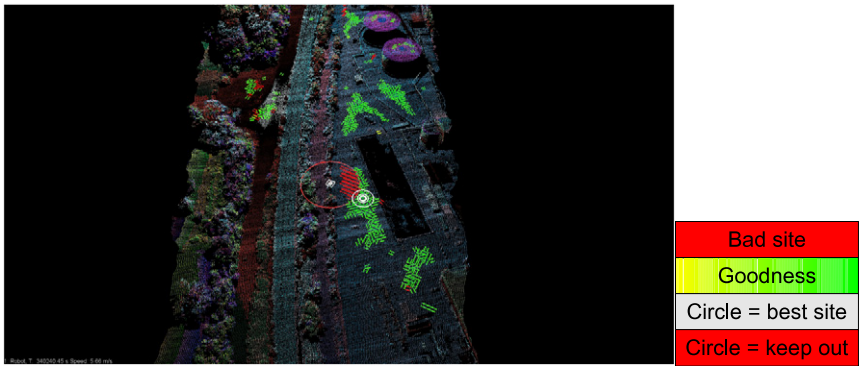
(a) Picture.



(b) Input point cloud.



(c) Coarse evaluation.



(d) Fine evaluation.

Fig. 16. Results from the "power plant" area.

car and the red region around the ground goal. The red region is a safety buffer that prevents landing directly on top of the ground goal. The overflight and real-time evaluation of landing sites is shown at http://youtu.be/IXbBNyt6tz4.

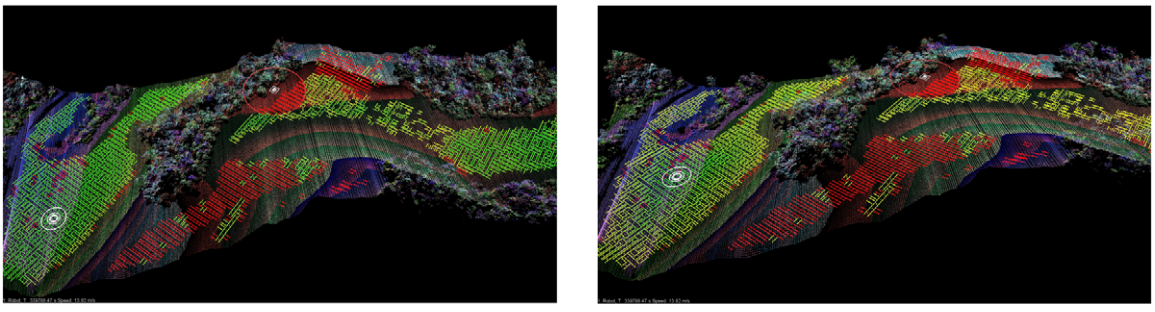
As a final experiment, we varied the weights for our evaluation function for four different parameters in the same terrain (Fig. 17). First we changed the weight for slope and roughness (Fig. 17(a)). Compared to the nominal evaluation, one can see that we now choose a landing site further away from the ground goal point because it is in smoother and less sloped terrain. A similar location is chosen in Fig. 17(b); however, since we try to emphasize the volume of the underbody, the shape of the LZ is actually more convex than the evaluation in Fig. 17(a).

If we give a large weight to the distances to the casualty, we expect the distance to be small. We can see in the evaluation for Fig. 17(c) that far points have low weight and close points have the lowest weight. In Fig. 17(d), the result is almost the same as Fig. 17(c), except that now points close to the boundary also have a low value. Varying the weights, one notices that the weight vector has a large influence on the resulting chosen landing site.

5.2. Landing the unmanned Little Bird helicopter

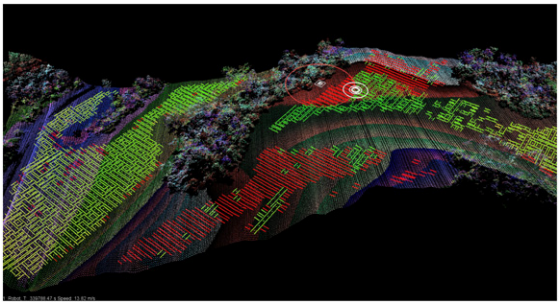
In the previous section we presented landing site evaluation results in a large variety of terrain around Pittsburgh, PA. In this section we present results from Mesa, AZ. In this environment

Please cite this article in press as: S. Scherer, et al., Autonomous landing at unprepared sites by a full-scale helicopter, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.09.004

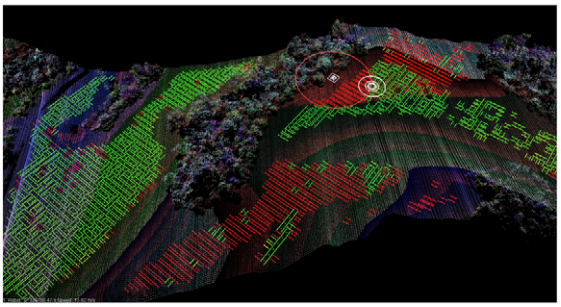


(a) $1000 \cdot w_1$.

(b) $1000 \cdot w_3$.



(c) $100 \cdot w_6$.



(d) $\frac{1}{1000} \cdot w_4$.

Fig. 17. The result of varying the weights for the final decision on selecting the best landing site. For a legend of the results see Fig. 13. In this plot we keep the function constant and vary the weight for some of the factors described in Table 2. In (a) we increase the weight of $1000 \cdot w_1$—the slope and roughness, for (b) we increase $1000 \cdot w_3$ to emphasize the clearance of the helicopter above the terrain, for (c) we try to minimize the distance to the casualty by increasing $100 \cdot w_6$, and in (d) we decrease the weight of the distance to the closest obstacles by changing $\frac{1}{1000} \cdot w_4$. One can see that depending on the cost function the distribution of good sites changes, as well as the best picked landing site. The weighting does not influence the binary decision of acceptable sites because it is based on hard thresholds of the helicopters constraints.

we tested the approach and ground path planning in addition to the coarse and fine evaluation. The ground path planning was performed to a ground goal location. Furthermore, the helicopter was completely autonomous and was able to land itself based on the evaluation.

5.2.1. Setup

We equipped the Boeing Unmanned Little Bird Helicopter [27] with a custom scanning Lidar that operates in two modes (described in Section 4.7): forward-scanning for obstacle detection and downward-looking for landing zone evaluation. The scanner has a 100 Hz, 100° fast horizontal scan and a controllable slower nodding scan at about 1 Hz. For landing zone evaluation it was pointed down and did not oscillate on the slow axis. The Riegl VQ180 pulses at about 85 kHz with full-waveform analysis and has a range of 150 m. The scanner is rigidly coupled to a Novatel SPAN INS/GPS to calculate a globally registered point cloud. The inertial measurement unit has a drift rate of 10°/h and the GPS is corrected with a base station to a reported 1 cm accuracy. The helicopter and main system components are shown in Fig. 18. The scanning lidar is mounted in the front with the IMU and the GPS antenna is located on the tail. The computing and flight control computers are located in the cargo bay. We discuss results from test flights at Boeing in Mesa, AZ (USA). The experiments were performed completely autonomously; however a test pilot and flight test engineer were onboard at all times to supervise and initiate the tests. The aircraft is optionally manned and can be flown with and without a pilot onboard. The autopilot was developed by Boeing and our algorithms send waypoint commands. In our runs we observed cross winds of about 15 knots.

5.2.2. Results

The following scenario was tested: an injured person (casualty) needs to be picked up and is located somewhere near a possible landing site as shown in Fig. 21(c). We want the vehicle to find a good landing site that is close to the ground location such that the ground goal can be reached from the landing site.

Fig. 18. System setup on the Boeing Unmanned Little Bird Helicopter. The sensor and IMU are shock mounted to the helicopter in the front. The GPS antenna is located in the tail and the computing and flight controller are located in the cargo bay. The rotor diameter is 8.3 m and has a payload weight of about 680 kg.

The helicopter takes off autonomously, is guided on a operator-specified search path and creates a terrain map. After the helicopter takes off the landing pads were made inaccessible. Fig. 19 shows pictures from observer and onboard cameras and views from the input data that show small obstacles such as crates, and black plastic cases that need to be avoided as landing sites. The landing site evaluation is performed as the vehicle flies over. The final landing site is picked after the ground traversability plans are not changing because if we still have potentially good landing sites that might be available we should keep searching for landing sites for a while.

After the overflight the landing site is picked as shown in Fig. 20. The coarse evaluation shows that crates, pelican cases, and a cart that are parked on the helipads are detected and rejected. After the landing sites are found with the fine evaluation in Fig. 20(c) the approach and abort paths are evaluated in (d).

Please cite this article in press as: S. Scherer, et al., Autonomous landing at unprepared sites by a full-scale helicopter, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.09.004

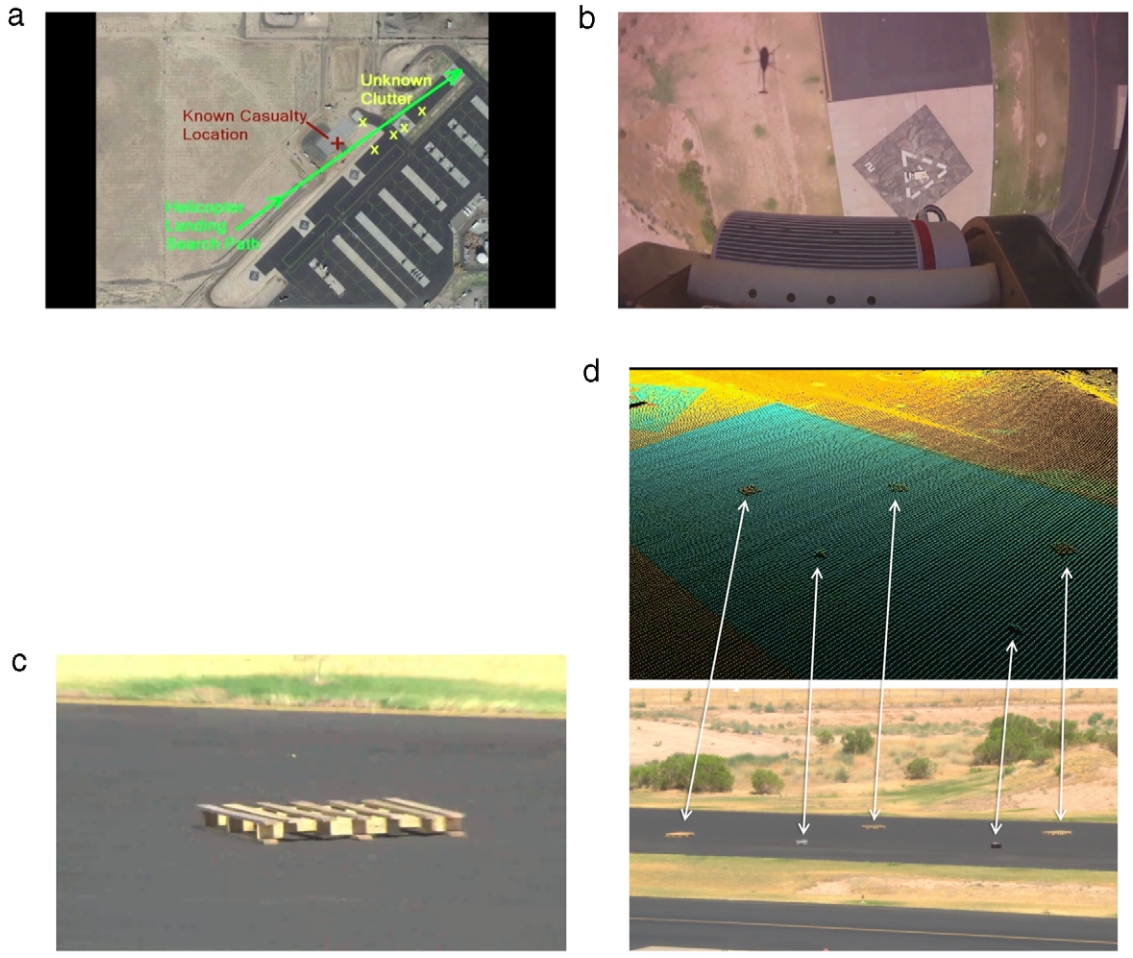


Fig. 19. The problem setup and input sample data. In (a) one can see the straight line search path that was performed to look for landing sites. A ground goal was given by a "casualty" that needs to be picked up. In (b) one can see a downward looking view from the helicopter looking at some obstacles on the runway. (c) shows an example obstacle and (d) shows a set of obstacles. (d) shows the correlation between obstacles (pallets, and plastic storage boxes) and the point cloud which is the input to the algorithm. The data was collected at 40 knots from 150 ft AGL.

The path to the ground goal is shown in (e). A video that shows one of the landing site search and landing runs is available at http://youtu.be/BJ3RhXjucsE.

From the parameters of the lowest cost landing site an approach path is computed and sent to the vehicle. On this trajectory the vehicle turns around and descends at the desired sink rate and comes to a hover about 3 m above the ground. After coming to a hover the vehicle orients itself to the optimal heading and then executes a touchdown. Part of the final approach segment is shown in Fig. 21 where the helicopter is close to the final landing site. The first view is a cockpit view of the clutter on the runway and vehicle on its way down to land. The second view shows the selected final landing site and other potential sites to land. The last picture shows the helicopter and the location of the ground goal. After a stable hover is achieved above the final goal location the vehicle touches down at the landing site.

After system integration and testing runs, a total of 8 landing missions were demonstrated with varying ground clutter and approach directions. Fig. 22 shows two typical landing missions. After an autonomous takeoff and climb out, the aircraft approaches the flight line from the south-west. It overflies the landing area at 40 knots and 150 ft AGL to search for acceptable landing sites. The computation time for landing zone evaluation is low enough that the vehicle can pick landing zones, and approach paths in real-time. The point cloud, aerial image and resulting path are shown in Fig. 22 for two typical missions. The approach cost ω_6 was varied between Fig. 22(a) and (b) to penalize for an unknown approach cost.

A close-up of two missions flown with a low ω_6 (approach cost) weight are shown in Fig. 23. Since no obstacles are present in Fig. 23(a) the vehicle prefers a landing site on the runway since it is a wide open area with a low landing site cost. As we cluttered the runway Fig. 23(b) the vehicle instead now prefers a closer site with a shorter path. This result was unexpected since we expected the vehicle to land on the runway instead because there was a no-fly zone to the north. However, the approach it found went east of the no-fly zone and landed in a feasible location.

In the next two experiments the approach cost weight was increased (Fig. 24). This did not change the outcome of landing without obstacles since it also previously landed on the runway. In the case of obstacle clutter however the vehicle now preferred an approach that flew over the scanned area. Since most of the obstacles were low it was able to overfly the obstacles before landing.

Table 3 shows the computation times for the optimized algorithms that minimize re-computation of new data. The computation was performed onboard with Intel 2.6 GHz Core 2 Quad computers Adding new data is fast and takes currently about 17 ms/100 000 points which is much faster than real-time. The coarse evaluation is very fast at 699 ms/100 000 cells. The fine evaluation is much slower which justifies the coarse evaluation to limit the number of fine evaluations performed. Approach evaluation also requires many evaluations and is therefore slower.

Please cite this article in press as: S. Scherer, et al., Autonomous landing at unprepared sites by a full-scale helicopter, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.09.004

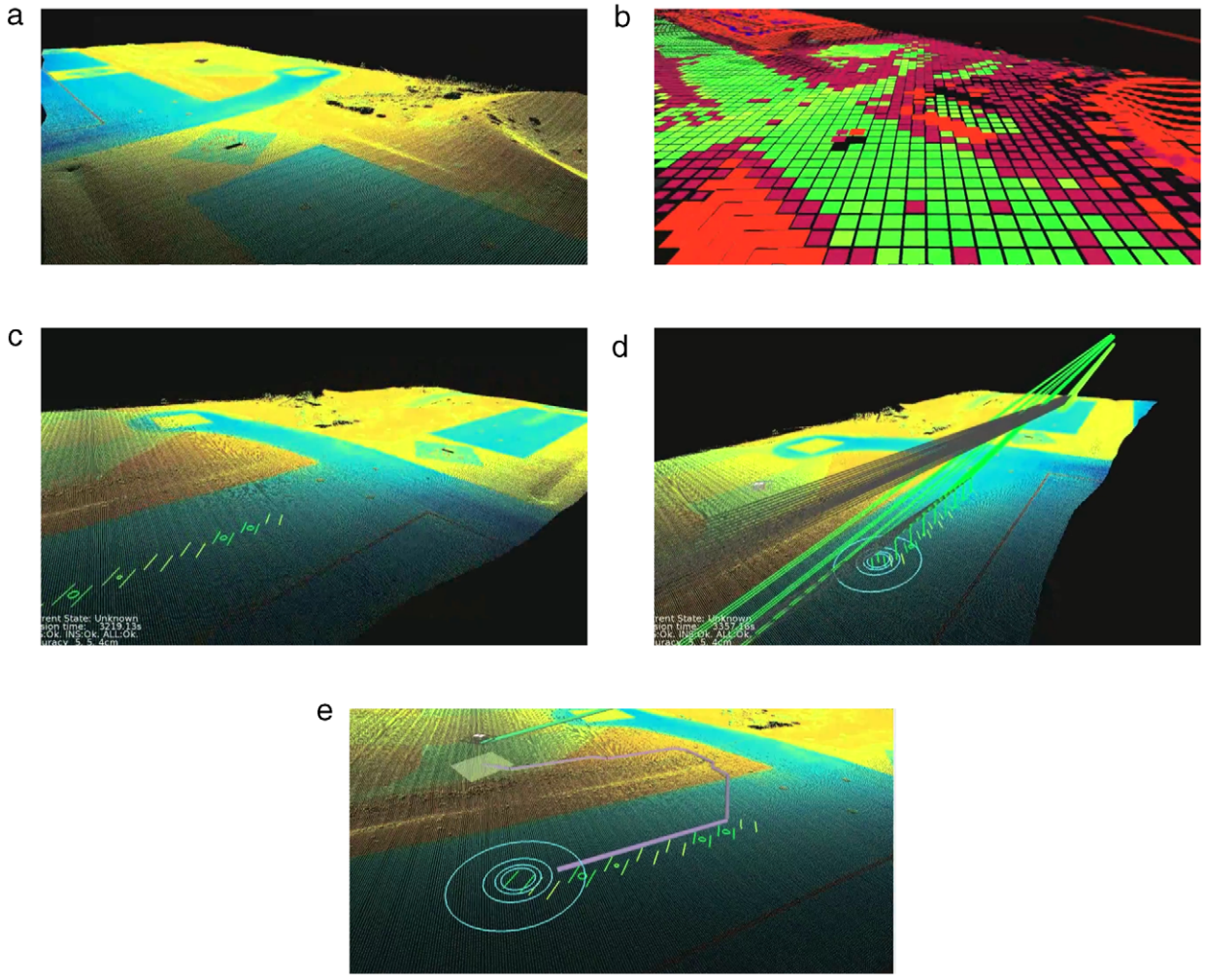


Fig. 20. Example results for the landing site evaluation in Phoenix, AZ. (a) Is the input point cloud. (b) shows the coarse evaluation. For a legend see Fig. 13. (c) shows good landing sites that were discovered by the vehicle and (d) shows the approach paths in green and the abort paths in grey. (e) shows the best site that is accessible from the ground goal. The purple trajectory shows the ground path to the ground goal ("casualty"). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 3
Computation times for five successful landing runs of the autonomous helicopter. Times are shown in ms. Landing zones are evaluated at 2 Hz.

Algorithm	Mean		Total		
	Time	Units	Time (ms)	Count (#)	Units
Adding points	17	ms/100 000 points	2723	15 826 707	Points
Coarse evaluation	699	ms/100 000 cells	7575	10 843 248	Cells
Fine evaluation	4578	ms/100 000 sites	5446	118 951	Sites
Approach evaluation	7663	ms/100 000 paths	9116	118 951	Paths
Approach change detection	21	ms/cycle			
Traversability planner	113	ms/cycle			
Average CPU utilization	30	%			

6. Discussion

We presented, to our knowledge, the first geometric model-based algorithm that gives estimates of available landing sites for VTOL aerial vehicles. In our algorithm we are able to incorporate many constraints that have to be fulfilled for a helicopter to be able to land. While the current state of the art is similar to our coarse evaluation algorithm, the fine evaluation allows us to consider more aerial-vehicle-relevant constraints and reject false positives in the coarse evaluation. Furthermore, since the two paths of evaluation are based on two different representations of the terrain, we conjecture that the combined system will be more robust to failure cases of a pure cell-based representation. We also include dynamic constraints based on the approach abort paths that respect the ideal landing profiles of full-scale rotorcraft. Finally, the system is capable of selecting sites that are accessible via paths on the ground.

We also presented results based on real sensor data for landing site evaluation at vegetated and urban sites. All of the hard constraints on accepting or rejecting a landing site are based on the actual constraints of the vehicle geometry and weight distribution. However, which of the good landing sites is preferable is an open problem because combining the different metrics is non-trivial. In future work we propose to learn the weight vector of the landing

Please cite this article in press as: S. Scherer, et al., Autonomous landing at unprepared sites by a full-scale helicopter, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.09.004

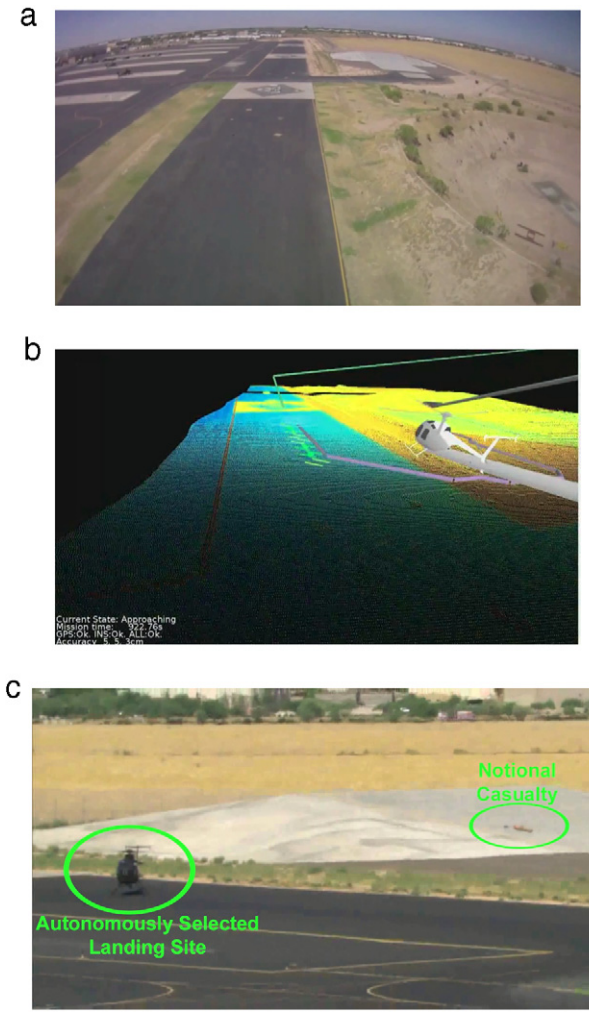


Fig. 21. Landing approach and touchdown. (a) shows a pilot's view during the approach and (b) shows a similar view in the data. In (c) one can see a picture of the helicopter after touchdown and the ground goal location.

site evaluation based on human preferences or self-supervised learning in simulation.

Results of tests for the first autonomous helicopter that is able to select its own landing sites were presented in this article. Results are shown in the context of a casualty evacuation mission where the path to a ground goal also has to be found.

We have observed two behaviors of water with our lidar scanner. With some water the beam is reflected and we get a smooth surface, and in other situations we do not get a reflection

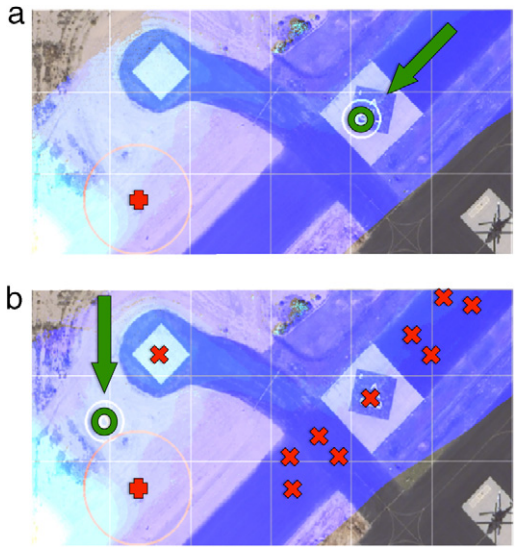


Fig. 23. Example of chosen landing sites for two test cases. The ground goal is shown as a red cross and the approach direction and touchdown point are indicated with a green arrow and green circle respectively. (a) shows a landing without obstacles where the vehicle picked a location on the runway since it has the lowest cost. In (b) the runway is cluttered with obstacles (red *x*'s. See Fig. 19(d)). The vehicle selected to land closer to the casualty instead even though the clearance is smaller. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

and therefore no measurement. No measurement in this case is good because we will reject such areas. On the other hand, we might keep searching the water areas because we are not able to gather data on them. Some ways to directly sense water would be with other sensor modalities such as a camera of a suitable spectrum, an improved lidar scanner, or a radar.

While water is a false positive, we will also reject potentially good landing sites (false negatives) because we currently classify vegetation as roughness. We have no information that allows us to distinguish grass from steel wire, for instance, and therefore we take a pessimistic approach.

We are making several assumptions that might not hold in all conditions: the algorithm assumes that small objects will not damage aircraft (no FOD), and the evaluation assumes that the terrain can bear the load of the helicopter. These assumptions could be relaxed by incorporating secondary sensors such as cameras that can classify the material of the terrain.

In future work one could encode the concept of prior known helipads as low cost regions in the landing site cost calculation. The only landing gear we model in the fine evaluation are two skids. In future work we would like to make the algorithm more flexible to

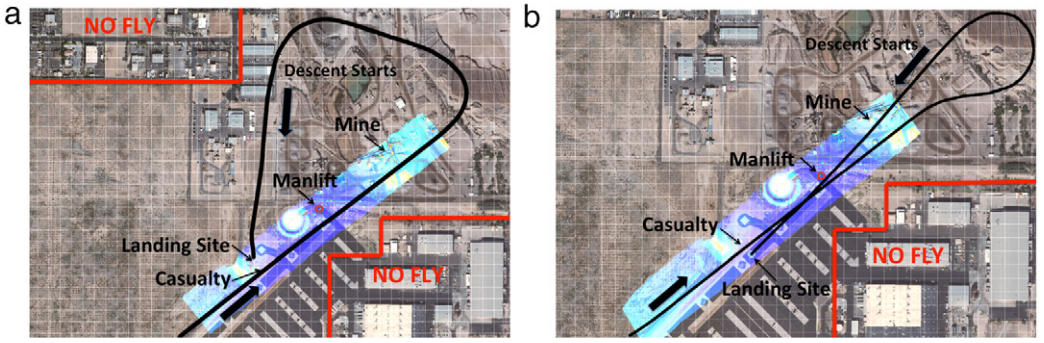


Fig. 22. Typical landing missions including survey overflight, descent on final and touchdown. In (a) the system is not penalized for flying the final approach through unsurveyed terrain and chooses to land from the north. In (b) the system is penalized for descending through unsurveyed terrain and therefore chooses an approach that maximizes coverage of already seen areas.

Please cite this article in press as: S. Scherer, et al., Autonomous landing at unprepared sites by a full-scale helicopter, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.09.004

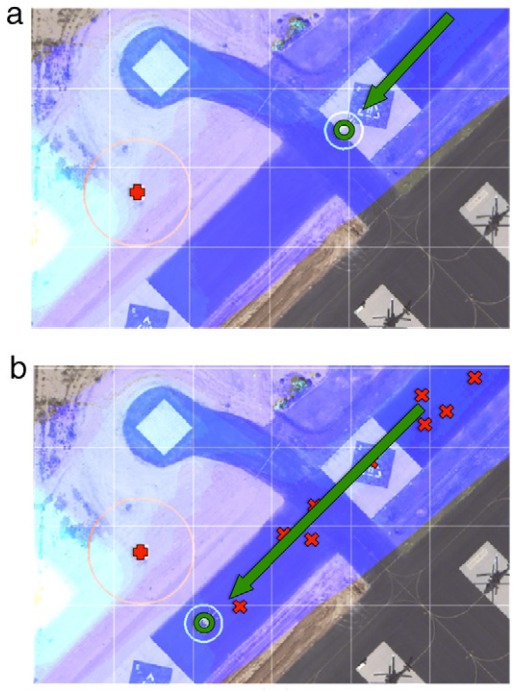


Fig. 24. Example of chosen landing sites with the system preferring known approach directions. In (a) the system selects a site on the runway that is reachable with a low-cost path the site selected is similar to Fig. 23(a). In (b) the vehicle will still prefer an approach through measured terrain. It picks a landing zone that is on the runway further away from the ground goal since it is the only clear area that has an obstacle clear glide slope. The area close to the ground goal is free and clear; however it is not chosen as a landing site since a hill (bright area in lower-left) increases the approach cost and the approach is not known to be free.

be applicable to different kinds of landing gear and test in more challenging environments.

Acknowledgments

Our team would like to thank TATRC (Telemedicine and Advanced Research Center) from the US Army Research and Materiel Command for funding this research. Our thanks go to our partners at Piasecki Aircraft Corp, whose expertise and financial commitment allowed us to test on a full-scale vehicle. The Boeing Company provided use of the Unmanned Little Bird and the test facilities in Mesa, AZ. We thank the ULB flight test team for making the integration and testing a safe and productive experience.

References

[1] F. Harris, E. Kasper, L. Iseler, US civil rotorcraft accidents, 1963 through 1997, Tech. Rep. NASA/TM-2000-209597, NASA, 2000.
[2] F. Kendoul, Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems, Journal of Field Robotics 29 (2) (2012) 315–378.
[3] J. Sprinkle, J. Eklund, S. Sastry, Deciding to land a UAV safely in real time, in: Proceedings of the American Control Conference, ACC, vol. 5, 2005, pp. 3506–3511.
[4] S. Saripalli, G. Sukhatme, Landing a helicopter on a moving target, in: Proceedings IEEE International Conference on Robotics and Automation, 2007, pp. 2030–2035.
[5] D. Barber, S. Griffiths, T. McLain, R.W. Beard, Autonomous landing of miniature aerial vehicles, in: Proceedings of the AIAA Infotech@Aerospace Conference, 2005.
[6] C. de Wagter, J. Mulder, Towards vision-based UAV situation awareness, in: Proceedings of the AIAA Conference on Guidance, Navigation, and Control, GNC, 2005.
[7] Z. Yu, K. Nonami, J. Shin, D. Celestino, 3D vision based landing control of a small scale autonomous helicopter, International Journal of Advanced Robotic Systems 4 (1) (2007) 51–56.
[8] L. Mejias, P. Campoy, K. Usher, J. Roberts, P. Corke, Two seconds to touchdown-vision-based controlled forced landing, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2006, pp. 3527–3532.
[9] J. Hintze, Autonomous landing of a rotary unmanned aerial vehicle in a non-cooperative environment using machine vision, Master's Thesis, Brigham Young University.
[10] S. Bosch, S. Lacroix, F. Caballero, Autonomous detection of safe landing areas for an UAV from monocular images, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2006, pp. 5522–5527.
[11] A. Johnson, J. Montgomery, L. Matthies, Vision guided landing of an autonomous helicopter in hazardous terrain, in: Proceedings IEEE International Conference on Robotics and Automation, 2005, pp. 3966–3971.
[12] J.F. Montgomery, A.E. Johnson, S.I. Roumelliotis, L.H. Matthies, The jet propulsion laboratory autonomous helicopter testbed: a platform for planetary exploration technology research and development, Journal of Field Robotics 23 (2006) 245–267.
[13] T. Templeton, D. Shim, C. Geyer, S. Sastry, Autonomous vision-based landing and terrain mapping using an MPC-controlled unmanned rotorcraft, in: Proceedings IEEE International Conference on Robotics and Automation, 2007, pp. 1349–1356.
[14] M. Sanfourche, G. le Besnerais, P. Fabiani, A. Piquereau, M. Whalley, Comparison of terrain characterization methods for autonomous UAVs, in: Proceedings of the 65th Annual Forum of the American Helicopter Society, Grapevine, TX, 2009.
[15] M. Whalley, M. Takahashi, G. Schulein, C. Goerzen, Field-testing of a helicopter UAV obstacle field navigation and landing system, in: Proceedings of the 65th Annual forum of the American Helicopter Society, AHS, Grapevine, TX, 2009.
[16] M. Takahashi, A. Abershitz, R. Rubinets, M. Whalley, Evaluation of safe landing area determination algorithms for autonomous rotorcraft using site benchmarking, in: Proceedings of the 67th Annual forum of the American Helicopter Society, AHS, Virginia Beach, VA, 2011.
[17] N. Serrano, A Bayesian framework for landing site selection during autonomous spacecraft descent, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2006, pp. 5112–5117.
[18] B. Sofman, J. Bagnell, A. Stentz, N. Vandapel, Terrain classification from aerial data to support ground vehicle navigation, Tech. Rep. CMU-RI-TR-05-39, The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, 2006.
[19] M. Hebert, N. Vandapel, Terrain classification techniques from ladar data for autonomous navigation, in: Proceedings of the Collaborative Technology Alliances Conference, 2003.
[20] B. Welford, Note on a method for calculating corrected sums of squares and products, Technometrics 4 (3) (1962) 419–420.
[21] T. Templeton, Autonomous vision-based rotorcraft landing and accurate aerial terrain mapping in an unknown environment, Tech. Rep. USB/EECS-2007-18, University of California at Berkeley, Berkeley, CA, 2007.
[22] S. Joon Ahn, Least Squares Orthogonal Distance Fitting of Curves and Surfaces in Space, in: Lecture Notes in Computer Science, Springer, 2005.
[23] F. Preparata, M. Ian Shamos, Computational Geometry: An Introduction, Springer, 1985.
[24] A. Fabri, G. Giezeman, L. Kettner, S. Schirra, S. Schönherr, The CGAL Kernel: a basis for geometric computation, in: M.C. Lin, D. Manocha (Eds.), Proc. 1st ACM Workshop on Appl. Comput. Geom., in: Lecture Notes Comput. Sci., vol. 1148, Springer-Verlag, 1996, pp. 191–202.
[25] H. Choset, K.M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L.E. Kavraki, S. Thrun, Principles of Robot Motion: Theory, Algorithms, and Implementations, The MIT Press, 2005.
[26] S. Thrun, M. Diel, D. Hahnel, Scan alignment and 3-D surface modeling with a helicopter platform, in: Proceedings of the International Conference on Field and Service Robotics, 2003.
[27] Boeing, http://www.boeing.com/rotorcraft/military/ulb, 2010.

Sebastian Scherer is a Systems Scientist at the Robotics Institute (RI) at Carnegie Mellon University (CMU). His research focuses on enabling unmanned rotorcraft to operate at low altitude in cluttered environments. He has developed a number of intelligent autonomous rotorcraft. Dr. Scherer received his B.S. in computer science, and his M.S. and Ph.D. in robotics from CMU in 2004, 2007, and 2010.

Please cite this article in press as: S. Scherer, et al., Autonomous landing at unprepared sites by a full-scale helicopter, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.09.004

Lyle Chamberlain is a Sensor Hardware Engineer at the Robotics Institute at Carnegie Mellon University. His work has focused on bringing robot perception technology to the world of flying vehicles, both large and small. Mr. Chamberlain received a B.S. in engineering and applied science from the California Institute of Technology in 2005.

Sanjiv Singh is a Research Professor at the Robotics Institute at Carnegie Mellon University. His research relates to the operation of robots in natural and in some cases, extreme environments. Dr. Singh received his B.S. in computer science from the University of Denver in 1983, his M.S. in electrical engineering from Lehigh University in 1985, and a Ph.D. in robotics from Carnegie Mellon University in 1995. He was a member of the research staff at the Robotics Institute from 1985 to 1989. He was a Postgraduate Fellow at Yale University in 1990 and an NSF Fellow at the Mechanical Engineering Laboratory in Tsukuba, Japan in 1992.

Please cite this article in press as: S. Scherer, et al., Autonomous landing at unprepared sites by a full-scale helicopter, Robotics and Autonomous Systems (2012), doi:10.1016/j.robot.2012.09.004