

## Master Thesis

# Vision-based safe proximity operation of a future Mars rotorcraft

Autumn Term 2024



# Chapter 1

## Methodology - Autonomous Landing Procedure

With the enhanced structure of the LSD output, having implemented a stereo camera as a low altitude alternative to SFM and after ensuring a correct ground truth comparison, the main contribution of this work could be faced: Bringing the visual landing site pipeline together with the autonomous framework in order to achieve reliable autonomous landing in unknown terrain.

This implementation can be split into the following parts:

- Landing Site Handling
- Landing Site Heuristic
- Landing Behavior

### 1.1 Landing Site Handling

Landing sites are constantly received by the autonomy's ROS connector alongside the mission tasks performed. They have to be processed in a separate thread. This is handled by a landing site manager (LSM) singleton class.

#### 1.1.1 Landing Site Manager

The incoming landing sites are ranked according to a loss function(1.2) and stored in a max-heap buffer in order to easily switch out the worst landing site for a new one at any given time.

### 1.2 Landing Site Heuristic

As mentioned in ?? the autonomy receives landing sites from LSD with the following properties:

- Position
- Roughness
- Uncertainty
- Size

- Obstacle Altitude

From these properties the characteristics that comprise the final heuristic are:

- Current Distance to Drone
- Roughness
- Uncertainty
- Size
- Verification Altitude

The final heuristic is in fact a square loss function defining the quality of a landing site:

$$L_{LS} = w_{dist}L_{dist} + w_{rough}L_{rough} + w_{var}L_{var} + w_{size}L_{size} + w_{verAlt}L_{verAlt} \quad (1.1)$$

### 1.2.1 Current Distance to Drone - $L_{dist}$

Well likely the single most important characteristic of an LSD-prefiltered<sup>1</sup> landing site. Each iteration the current distance to the drone's position is calculated for each retained landing site. The distance is then normalized by dividing it by the cruise altitude which is 100m. In practice there were easily enough landing sites found while moving to allow landing sites to fall off when being farther away than 100m.

### 1.2.2 Roughness - $L_{rough}$

The roughness property is the unaltered roughness value received from LSD. It is already normalized and enters the loss function as it is.

### 1.2.3 Uncertainty - $L_{var}$

The same holds for the uncertainty. It is already normalized by design and enters the loss function unaltered.

### 1.2.4 Size - $L_{size}$

Analogous to the roughness and uncertainty properties the size comes from the landing site detection directly. However unlike the two preceeding properties it is not normalized but simply denotes the metric radius of the largest circle of valid landing area that can be fit around a given landing site. This is achieved in LSD by performing a distance transform on the created landing site image.

In order to normalize this value the maximum landing site size is retained and each landing site's size is divided by it in order to achieve normalized size information.

### 1.2.5 Verification Altitude - $L_{verAlt}$

A site's verification altitude is the smallest vertical distance between the drone and the landing site at which that site was (re-) detected.

The verification altitude is a useful property because of numerous reasons.

---

<sup>1</sup>Each received landing site has already undergone a threshold filtering regarding slope and roughness.

### Further Indication of Certainty

First of all similar to the uncertainty metric the verification altitude indicates how certain we can be about a detected landing site as spots detected at lower flight altitudes are more likely correct due to the reduced depth error. Even though it might seem overlapping with the uncertainty property in this regard, these two characteristics are quite complementary as the uncertainty takes OMG convergence and camera specifics into consideration while the verification altitude is a purely location based metric.

### Landing Site Property Updates

As the verification altitude yields a simple and good estimation of the trustworthiness of an incoming landing site, it can be used as a flag to know, when a landing site's properties should be updated. When a landing site is redetected with a verification altitude lower than the previously stored one, the algorithm trusts it more and alters the previously stored properties to the new ones received.

### Verification

Continuously updating the verification altitude upon redetection allows us to determine the lowest altitude, at which a landing site was redetected. This information can be used to verify that a given site was considered a valid landing spot even at low altitudes.

## 1.3 Landing Behavior

### 1.3.1 Action Definition

The final landing behavior is implemented in the form of a behavior tree which allows adaptive decision making.

The autonomous framework already defined the core control flow nodes as well as some of the action nodes required for the landing behavior. (??) The following additional actions were defined in this work.

#### AscendToClearAltitude

Once a landing site has been chosen, the failsafe mechanism to go to that landing site is the following:

1. Ascend to a safe altitude
2. Traverse laterally to the landing site's xy-position
3. Descend to the

#### ChangeAltitudeLSAction

#### GetLandingSiteAction

#### GetNextPatternCenterWPAction

#### LandingSiteSearchAction

#### LandingSiteVerificationAction

### 1.3.2 Behavior Tree

# Bibliography