FAST RANGE IMAGE BASED LANDING FIELD DETECTION

R.M. Leitner and O. Heinzinger
EADS Deutschland GmbH. / EADS Innovation Works
Sensor, Electronics & Systems Integration
D-81663 Munich
Germany

OVERVIEW

The paper presents a simple landing field detection algorithm for rotary wing UAVs based on perspective range image data. The algorithm provides the detection of the largest, nearly planar, circular landing pad of each shot, which satisfies several preset constraints, e.g. lower bound of landing field diameter or attitude deflection of landing pad normal with respect to the earth gravity vector, in unfamiliar terrain. Landing site pose (position and orientation) relative to the range image camera and furthermore to the vehicle body can also be computed, to use the landing field detection process for automatic landing applications of autonomous rotorcrafts in near future.

Keywords: Photogrammetry, 2.5D Surface Simplification, Delaunay triangulation, maximum inscribed circle.

1. INTRODUCTION

One of the crucial basic features for autonomous operations of unmanned aircrafts is the automatic landing in more or less known terrain. Thereby cost-efficient sensors, especially optical sensors, can be applied to achieve this goal.

The automatic landing procedure that is presented here can be seen as a two-phase process, which is divided into a *Detection*- and *Guidance-Phase*. As the first phase is dealing with the localization of a potential circular landing site, the second phase guides the vehicle to the landing point, which could be the center point of the located circular landing site. The paper at hand is engaging with the first phase, the *detection problem*.

Before describing the *Detection-Phase* in detail, the Search-Field and the Search-Criterions have to be explained.

The Search-Field or terrain can be a complex structure with several characteristics, e.g. *smoothness*, *softness*, et cetera. For the consideration of a simple landing field detection technique, these characteristics must be merged into significant categories. These are *Geometry* and *Softness* of the terrain surface and furthermore the knowledge of *Restricted-Areas*, that contain hazards of any type, e.g. dynamic obstacles.

With the help of these three categories a terrain classification can be arranged. If all categories are known *a priori*, a well-known terrain *Type-A* is available; *vice versa* an unfamiliar terrain *Type-C* exists. If only one or two categories are familiar a partially-known terrain *Type-B* is at hand. Well-known (*Type-A*) terrain illustrates the best and safest case; everything is well-established, but this is not the case in reality and is only a theoretical assumption.

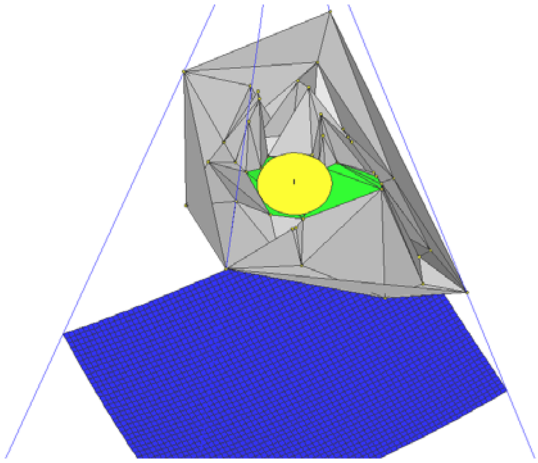

FIG 1. Simplified Image with detected Landing Pad

Basically the determination of the terrain geometry constitutes no great technical barrier. This can be solved faultlessly with optical sensors, like e.g. laser scanners. It is much more complicated for the two other categories. The *Softness* of terrain is difficultly and expensively to measure and just as the knowledge of *Restricted-Areas* only pretended from external. Chances for a reliable landing field location are increasing with better knowledge of the terrain characteristics.

The Search-Criterions for landing field detection for rotary aircrafts can be splitted up in similar categories as for the Search-Field detection. A potential landing field should have certain shape and size, which allows a safe landing for the rotorcraft. Because of the rotor disk of a helicopter, circular landing pads are suited best for potential landing field candidates. The size of the pad shall include a safety factor to prevent rotor contact with potential surrounding obstacles detected by the sensor.

Moreover, the angular tilt of the located pad shall not exceed limits in order to minimize the downhill-slope-force and the risk of sliding and tilting of rotorcraft. In addition to the knowledge of areas that bear the threat of sinking (softness of the surface), the system also should be aware of other hazardous objects in the area (danger of collision).

Because of these fundamentals the landing field detection is based on the *Geometry* with adequate sensors. The *Softness* and the knowledge of *Restricted-Areas* are not measured by the system itself, they are communicated by an external means.

In this case for the determination of the terrain geometry, optical 3D sensors are used.

As already stated, computation speed is very important for the algorithm to achieve *real-time* capability. Here a compromise between rapidity and accuracy must be found. That is the reason why a fast coarse-to-fine method is in use to maintain quick approximations with an appropriate level of detail of the original range image.

According to this approximation, the algorithm computes the largest circular landing field and its pose (position and orientation) relative to the camera and furthermore to the vehicle body.

The paper is organized as follows: The following chapter deals with the pre-processing of the distance values including the collinearity equation derived from pin-hole camera model. Coarse-to-fine process and computation of maximum feasible, nearly planar circular landing pad are presented in the following two chapters. A descriptive result, the summary and future steps conclude this paper.

2. RANGE IMAGE PRE-PROCESSING

Due to the use of optical sensors, which supply perspective range data, the pre-processing of raw data has to be mentioned.

A photograph is essentially a projection of a 3D object-space onto a 2D image-space. A simplified approximation of the geometric relation between these two spaces can easily be demonstrated with the pin-hole camera model which is based on perspective projection, see **Figure 2**.

For each image pixel *P* of the image-matrix σ a respective range ray value λ_P to any object point *T* is measured. Because of the perspective projection all range rays go through the projection center *C* and intersect the image matrix.

If the intrinsic camera geometry is known, this means the size and center point *S* of the image matrix, the principal point *H*, the focal length *f* and the radial lens distortion, the relative Cartesian position of each object point T to the projection center C can be computed by the collinearity equation [1], see **Equation 1**.

(1) $$\vec{\mathbf{r}}_{CT,c} = \left(\left\|\vec{\mathbf{r}}_{CP,c}\right\|_2 - \lambda_P\right)\vec{\mathbf{e}}_{CP,c}$$

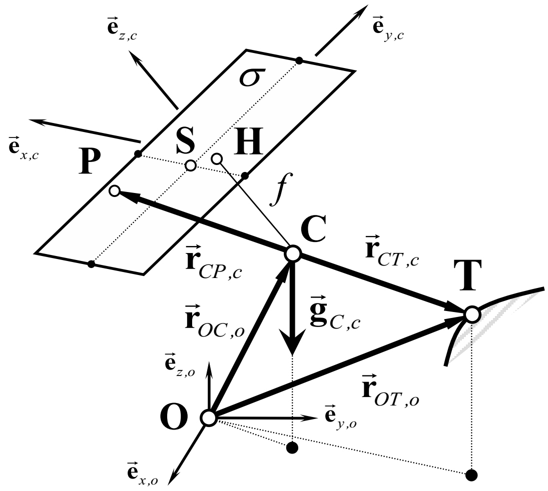

FIG 2. Interior and exterior pin-hole camera geometry.

3. COARSE-TO-FINE METHOD

3.1. Philosophy

After computing the relative Cartesian positions of the object points, a 2.5D point cloud exists. These points must be triangulated to represent the relief including occlusions of the scanned terrain. This triangle mesh can be performed with different triangulations methods, e.g. Delaunay Triangulation or Data-Dependent Triangulation.

For this algorithm the Delaunay Triangulation is applied, which is a purely two dimensional method, but can be used when operating in image space, see **Figure 3**. The Delaunay Triangulation meshes a set of points, such that no point is inside the circumcircle of any triangle ➔ *Delaunay Criterion*.

The landing field detection requirements e.g. lower bound of landing field diameter and/or attitude deflection respectively to earth gravity vector, demand a strategy to find nearly smooth pads of given attitude in the 3D meshed range image in a very short time period to guarantee real-time computation. Therefore the coarse-to-fine method or iterative refinement will be considered to maintain quick approximations with an appropriate level of detail of the original meshed range image [2, 3].

The method of surface simplification based on triangles under observance of Delaunay Criterion, begins with an initial approximation of two triangles and iteratively adds new vertices to the current triangulation until a predefined target is achieved. This goal is typically stated in terms of desired error threshold or a maximum number of vertices. Each step of iteration inserts a single point (pixel) with maximum local error ε (e.g. perpendicular distance to its appropriate triangle, see **Equation 2** and **Figure 3**) to current triangular approximation.

(2) $$\varepsilon_\perp = \left|\vec{\mathbf{n}}^T_{\pi,c}\vec{\mathbf{r}}_{T_iT,c}\right|, \quad T_i \in \pi, \quad \left(i = 1,2,3\right)$$

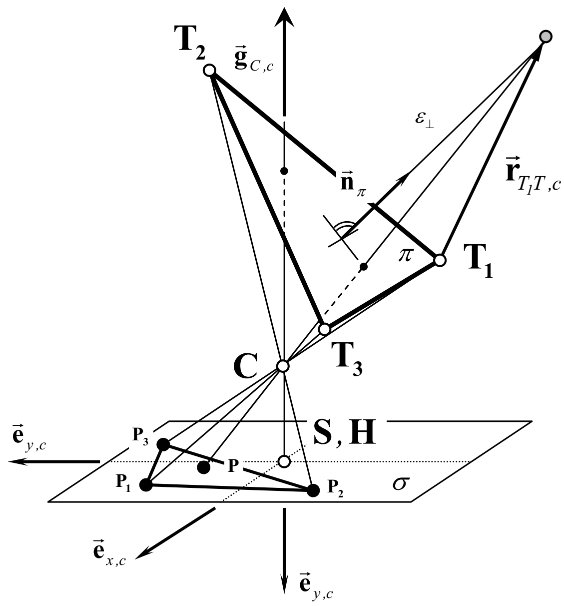

FIG 3. Geometry of central projection

In the following, an overview over the coarse-to-fine procedure is given step-by-step:

(1) Build an initial approximation of two triangles.

(2) Find point with the largest local error (best vertex).

(3) Insert point to the current approximation.

(4) Split triangle into three.

(5) Do edge flip, if incircle test is positive.

(6) Stop if any threshold is reached, otherwise go to 2.

A short introduction of the incircle test and edge flip will be given.

3.2. The incircle test and edge flip

The Delaunay Criterion states that no point is inside the circumcircle of any triangle. If this happen, an edge flip must be accomplished, to achieve this criterion. To check if a point is inside a triangle the following incircle test based on determinates will be applied.

Check if point T is inside a circle determined by three points T_1, T_2 *and* T_3. First the orientation O of the circle appropriate points must be calculated by help of inline test, see **Equation 3**.

$$(3)\quad O = sign\begin{vmatrix} r^x_{T_3T_1} & r^y_{T_3T_1} \\ r^x_{T_3T_2} & r^y_{T_3T_2} \end{vmatrix}$$

If result of equation above is positive, point T1 lies at left hand side of vector $\vec{\mathbf{r}}_{T_2T_3}$ and the direction of rotation is mathematically positive, vice versa for a negative result.

Then the incircle test will be computed by **Equation 4**

$$(4)\quad IC = \begin{vmatrix} r^x_{TT_1} & r^y_{TT_1} & -\left(\left(\mathbf{r}^{x/y}_{CT}\right)^T\right)^2\left(\mathbf{r}^{x,y}_{CT_1}\right)^2 \\ r^x_{TT_2} & r^y_{TT_2} & -\left(\left(\mathbf{r}^{x/y}_{CT}\right)^T\right)^2\left(\mathbf{r}^{x,y}_{CT_2}\right)^2 \\ r^x_{TT_3} & r^y_{TT_3} & -\left(\left(\mathbf{r}^{x/y}_{CT}\right)^T\right)^2\left(\mathbf{r}^{x,y}_{CT_3}\right)^2 \end{vmatrix}$$

to check if point T is inside, outside or at the circle.

$$(5)\quad ICO \begin{cases} \geq 0 & \text{inside circle/cocircular} \\ < 0 & \text{outside circle} \end{cases}$$

If the product of *IC* and *O (ICO)* is positive, an edge flip must be executed, as shown in **Figure 3** to achieve the Delaunay Criterion.

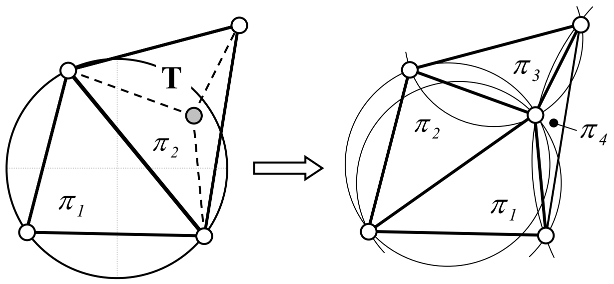

FIG 4. Delaunay Criterion and Edge Flip.

Only the triangles, whose normal have low angular deflection to earth gravity vector, are suitable for the following computation of maximum circular landing field.

Each triangle must be tested if it executes the angular deflection constraint; this means if the scalar product is greater than a given threshold, to get part of set G.

$$(6)\quad \begin{gathered} \pi \in G \Leftrightarrow \left|\vec{\mathbf{n}}^T_{\pi,c}\mathbf{M}_{cu}\vec{\mathbf{g}}_{C,u}\right| \geq \delta_{min} \\ \vec{\mathbf{g}}_{C,u} = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}^T_u \end{gathered}$$

The triangles of set G have to be checked, if they generate connected surfaces. Therefore a triangle gathering loop must be executed which is equal to a breadth first search process to get one or more triangle structures.

Because the gravity vector is given in another frame than the camera *c* or vehicle body fixed frame *b*, a linear transformation must be executed. The orthogonal matrix

$\mathbf{M}_{cu}$ transforms the gravity vector $\mathbf{g}$ which is stated in navigation system u to the camera fixed frame c.

The transformation matrix **Equation 7** consists of three orientation angles including the attitude of the flying vehicle to any navigation frame (e.g. ***North-East-Up*** - or ***North-East-Down*** - frame) and the attachment angles of the camera to the vehicle body.

$$\mathbf{M}_{cu} = \mathbf{M}_{cb}\mathbf{M}_{bu} = f\left(\alpha_{c2}^{cu}(t), \beta_{21}^{cu}(t), \gamma_{1u}^{cu}(t)\right) \tag{7}$$

4. MAXIMUM INSCRIBED CIRCLE (MIC)

By now the triangle structures (gathered triangles) and their boundary points are known. For the next step of computing the **M**aximum **I**nscribed **C**ircle (**MIC**), the boundary of each structure should remain more points to ensure no intersection between the circle and the boundary tangents, see **Figure 5**.

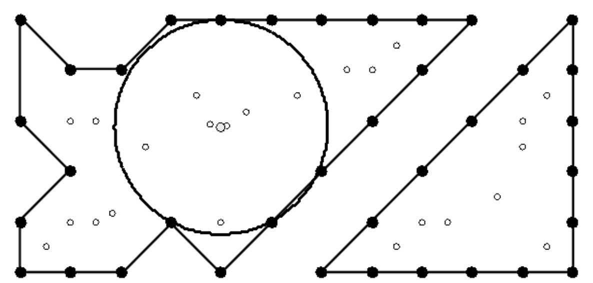

FIG 5. MIC with Voronoi points.

Afterwards the boundary points are transformed into the navigation frame u. Thereafter only the projected 2D surfaces are of note, hence the x and y components comprised the 2D boundary values to apply to the MI-C computation with Voronoi points. Voronoi points are the nearest neighbours for a set of points and can be computed fast and reliable and are ideal for determining the maximum inscribed circle. For the computation of these points the Voronoi algorithm from [4] is in use.

5. RESULTS

The following three figures show an original range image and its approximation with coarse-to-fine procedure, as described in chapters before. The original scene is only an example, showing two obstacles in front of a buckled wall.

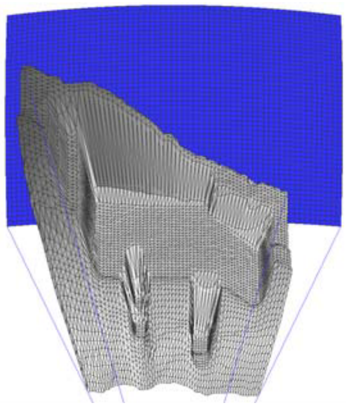

FIG 6. Original perspective range image

The iterative refinement procedure yields a good level of detail after less iteration steps.

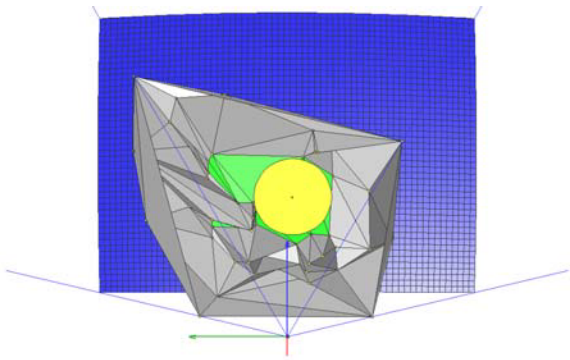

FIG 7. Approximation of perspective range image

The gravity vector $\boldsymbol{g}$ is pointing to the middle of the scene. The maximum circular landing pad is well located at the wall between the two obstacles, as seen in **Figure 7**.

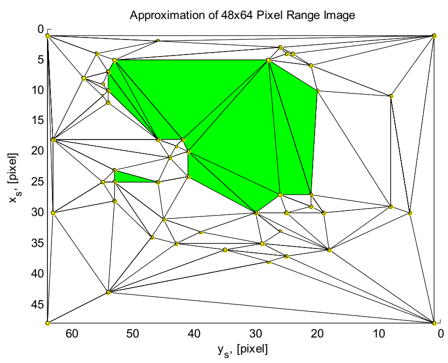

FIG 8. Simplified Delaunay mesh for figure above

6. CONCLUSION AND FUTURE STEPS

The landing field code gives the possibility to detect the largest, nearly planar, circular landing pad of each image shot, which achieves several preset constraints, e.g. lower bound of landing field diameter and/or attitude deflection of landing pad normal with respect to the earth gravity vector, in partially-known terrain. It works fast and gives sufficient results.

Not only perspective range images can be used for landing field detection. It is also possible to take an ortho-photo, which can be understood as an orthogonal projection of an object to any plane.

Testing the code on the target hardware is in progress with real time capability achieved. However, it remains to check how optimization for efficiency will increase performance. The testing environment for the system is being increased in complexity in the current tests.

REFERENCES

[1] F.H. Moffit, E.M. Mikhail: Photogrammetry, Third Edition, Harper and Row, 1980.

[2] A. Ardeshir Goshtasby: 2-D and 3-D Image Regression for medical, remote sensing, and industrial applications, John Wiley & Sons, 2005.

[3] M. Garland, P.S. Heckbert: Fast Polygonal Approximation of Terrains and Height Fields, School of Computer Science, Carnegie Mellon University, Pittsburgh, 1995.

[4] Qvoronoi-Voronoidiagramm, http://www.qhull.org