

Probabilistic Hough Voting for Attitude Estimation from Aerial Fisheye Images

Bertil Grelsson^{1,2} and Michael Felsberg¹

¹ Computer Vision Laboratory, Linköping University, Sweden

² Saab Dynamics, Linköping, Sweden

firstname.lastname@liu.se

Abstract. For navigation of unmanned aerial vehicles (UAVs), attitude estimation is essential. We present a method for attitude estimation (pitch and roll angle) from aerial fisheye images through horizon detection. The method is based on edge detection and a probabilistic Hough voting scheme. In a flight scenario, there is often some prior knowledge of the vehicle altitude and attitude. We exploit this prior to make the attitude estimation more robust by letting the edge pixel votes be weighted based on the probability distributions for the altitude and pitch and roll angles. The method does not require any sky/ground segmentation as most horizon detection methods do. Our method has been evaluated on aerial fisheye images from the internet. The horizon is robustly detected in all tested images. The deviation in the attitude estimate between our automated horizon detection and a manual detection is less than 1° .

Keywords: Fisheye images, attitude estimation, horizon detection, Hough voting.

1 Introduction

For autonomous navigation of unmanned aerial vehicles (UAVs), continuous position and attitude estimation (pitch and roll angle, fig 1a) is essential. Inertial measurement units (IMUs) are standard sensors for this purpose but they suffer

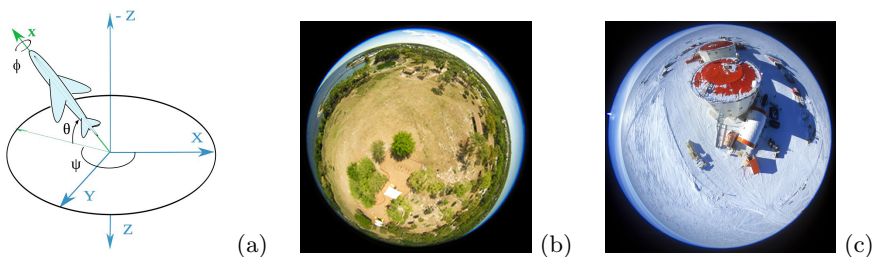


Fig. 1. (a) Definition of vehicle pose, $\psi =$ yaw, $\theta =$ pitch, $\phi =$ roll. Typical aerial fisheye images, courtesy markmarano.com (b) and gdargaud.net (c).

from drift and need support from other sensors to give accurate absolute pose estimates over time. Visual methods have proven to be potent sensors for absolute attitude estimation, where their applicability depends on the scene and camera lens type. Our specific interest is to estimate the attitude from aerial fisheye images with a field of view (fov) larger than 180° . As a means to achieve the objective, we detect the horizon in the images. We assume that the images are taken at sufficient altitude for buildings and trees not to occlude most of the horizon. Our goal is thus not to obtain a perfect segmentation between the sky and the ground but to infer the camera pitch and roll angles, θ and ϕ , from the estimated horizon. The main goal is a robust attitude estimation method for aerial fisheye images that can be run onboard the vehicle for navigation support. Typical aerial fisheye images used are shown in fig 1(b) and (c).

Most methods using horizon detection to infer the camera attitude consider sky/ground segmentation as the initial and most crucial step, [1–3]. Our proposed method for attitude estimation of fisheye images is similar to [4] in that it does not require sky/ground segmentation but instead uses an edge detector and a Hough voting [5] scheme. A major difference compared to [4] is that we weight the Hough votes based on the probability distributions for the altitude and the pitch and roll angles. In a flight scenario, there is often some knowledge of the current altitude and attitude and we want to exploit this prior to make the attitude estimate more robust.

For a calibrated camera and given the altitude and attitude, the position and orientation for all horizon points in the image plane are deterministic. We use this information reversely in the voting. We project each edge pixel onto the unit sphere and use the edge orientation to compute the tangent vector of the horizon plane on the unit sphere. For a given altitude, the horizon is projected as a circular disc with known radius on the unit sphere. Given the projection point, the tangent vector and the disc radius, the horizon plane on the unit sphere is uniquely defined and the pitch and roll angles can be computed. We let all edge pixels in the image vote on a pitch-roll accumulator array, and rely on that votes from the true horizon will aggregate over a small area of the array whereas other edge pixels will spread their votes in a more random fashion over the array. To suppress local minima, we convolve the accumulator array with a gaussian kernel prior to extracting the pitch-roll cell with the maximum value.

The main contribution of this paper is the combination of (1) Computing attitude votes from projection of edge pixels and their orientation on the unit sphere, and (2) Weighting the votes based on the prior probability distributions for the altitude and pitch and roll angles, in order to obtain a robust and geometrically sound attitude estimate.

Our method has been evaluated on real images, obtained by searching the internet for series of aerial fisheye images. For these images, there is no attitude ground truth available, and the evaluation criterion is how well the horizon can be estimated compared to a visual estimate.

1.1 Related Work

Shabayek et al. [1] give a good overview of available methods where horizon detection is used to infer the camera attitude. Demonceaux et al. [2] use a Markov Random Field formulation with projections of image color components on the unit sphere for sky/ground segmentation. A least-squares fit of horizon points to a plane on the unit sphere is used to infer the attitude. No computation times are reported but MRFs are often time consuming. Thurrowgood et al. [3] propose a very simple use of the brightness and a linear combination of the RGB channels called C as a first step to discriminate sky and ground pixels, based on statistics from a training set. A histogram of the C values in the query image is used to tune a threshold based on the prior expectation of the number of sky/ground pixels in the image. Horizon points are projected to the viewsphere where a best fit to a plane is used for attitude estimation. Their method is quick (2 ms for a 300x300 image) but 10 of 124 test images are claimed to be "unusual" in color content leading to misclassification. Shabayek et al. [1] propose to use three linear polarization filters on the image and base the sky/ground segmentation on the phase and degree of the polarization. The method is only evaluated on one example image. In [6] they use a support vector machine (SVM) classifying pixels according to color for sky/ground segmentation.

For low altitude images in urban environment, Hwangbo et al. [7] propose a method that utilizes the fact that in man-made structures line segments are often vertical and the attitude can be inferred from vertical vanishing points in the image. For mountainous scenes, Baatz et al. [8] suggest a method with sky/mountain segmentation and matching the contour with a DEM of the whole country (Switzerland). For perspective images, Bao et al. [4] do not perform an explicit sky/ground segmentation but instead use an image edge detector. Edge pixels then vote for horizon line directions and positions in a Hough-like manner to infer the camera attitude.

2 Fisheye Lens and Earth Models

The fisheye lens is modelled as in [9] by first projecting a world 3D point M onto the unit sphere as point m , fig 2(c). The points on the unit sphere are then projected onto the image plane by a perspective camera model with its optical center a distance L from the center of the unit sphere, and focal distance f to the image plane. Ideally, camera and lens distortion parameters should be included in the model, but since these parameters are not known to us for the internet images used, they are omitted in our model.

2.1 Earth and Horizon Model

We model the earth as a sphere with radius $R_e = 6371$ km. Since we are mainly interested in altitudes $h < 1$ km, the assumption $h \ll R_e$ is valid.

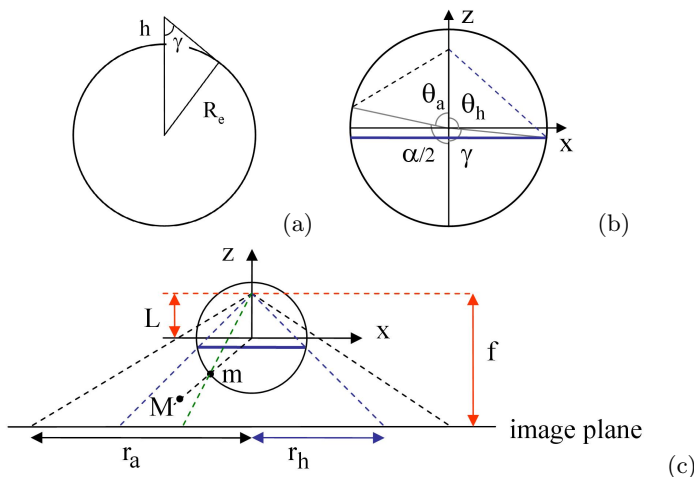


Fig. 2. (a) Earth and angle to horizon. (b) Maximum viewing angle and angle to horizon on unit sphere. (c) Unit sphere with image of maximum viewing angle and horizon on image plane.

The camera altitude h in fig 2(a) is exaggerated for clarity. The angle γ to the horizon is

$$\gamma = \arcsin \frac{R_e}{R_e + h} \approx \arcsin(1 - \frac{h}{R_e}) \tag{1}$$

As illustrated in fig 2(b), the angle θ_a between the maximum viewing angle ($\alpha/2$) and the z-axis, and the angle θ_h between the z-axis and the projection of the horizon are, respectively, given by

$$\theta_a = \pi - \frac{\alpha}{2}, \quad \theta_h = \pi - \gamma \tag{2}$$

The radius for the fisheye circle (maximum viewing angle) and the horizon circle (assuming vertical camera) on the image plane will be, fig 2(c),

$$r_a = f \frac{\sin \theta_a}{L - \cos \theta_a}, \quad r_h = f \frac{\sin \theta_h}{L - \cos \theta_h} \tag{3}$$

Normally, the calibration of a fisheye camera could be performed as in [10]. For the internet images used, we have replaced the calibration with these steps: (1) Set the maximum viewing angle for the lens, assume a value if not given on the web site. (2) Determine the radius of the fisheye circle, r_a , in pixels for one image, (black border in image due to maximum viewing angle). (3) Determine the radius of the horizon circle, r_h , for an image with its optical axis close to vertical and make an assumption on the altitude for the camera. (4) Solve for L and f using eqs (1 - 3).

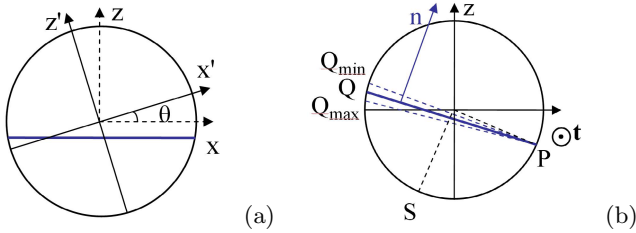


Fig. 3. (a) Horizon on unit sphere for pitch angle change. (b) Estimate of horizon normal \mathbf{n} from edge points. Tangent vector \mathbf{t} is directed out of paper.

2.2 Image of Horizon

Given the above camera and earth models and assuming no camera tilt, we can combine eqs (1 - 3) to infer that the image radius of the horizon will vary with the altitude h as

$$r_h(h) \approx r_h(0) \left(1 - \sqrt{\frac{2h}{R_e}}\right) \tag{4}$$

neglecting occlusion of the horizon at low altitudes. As the camera altitude is increased, the image radius of the horizon will decrease very slowly.

If we tilt the camera, e.g. changing the camera pitch angle, i.e. rotating the camera around the unit sphere y-axis an angle θ , as in fig 3(a), the horizon will effectively be rotated an angle $-\theta$ on the camera fixed unit sphere. The projection of the horizon on the image plane will be more elliptic as the tilt angle is increased. Even for rather small tilt angles, part of the horizon will be projected above the fisheye circle on the unit sphere and will not be seen in the image plane.

3 Horizon Estimation Method

Our horizon estimation method incorporates a probabilistic Hough voting [5] scheme where edge pixels on the image are weighted in the voting, based on how likely they are to be a horizon edge pixel given the probability distributions for the camera altitude and the pitch and roll angles. We assume these distributions to be roughly known in a true flight scenario and we want to exploit that information. If the distributions are unknown, wide distributions can be assumed. Prior to the Hough voting, we perform some image processing steps.

3.1 Image Processing

Edge Detector. The first step in our method is an edge detection and we use the Canny detector [11] as it has proven to give robust results. Before applying the Canny detector, the color image is converted to grayscale and the image is smoothed with a gaussian 5x5 kernel.

Fisheye Circle Detection and Removal. The fisheye circle in the images, caused by the maximum viewing angle, is not in the exact same location for every image. The location is estimated by best fitting a circle to the border points. The first and last edge pixel along each image row and column are extracted and collected as potential border points. We then apply a RANSAC loop [12]. We pick three random border points and fit a circle through them. We compute the consensus set of border points for this circle, and count all points that are within a distance $r_{thr} = 1.0$ px from the circle. The circle giving the largest consensus set is taken as the border circle with center point (x_0, y_0) and radius r_0 . Note e.g. in fig. 5 (second row) where sun effects will introduce border points that are outside the true fisheye circle. There are also other examples when the ground is very dark and the first edge pixel along a column or row is inside the true fisheye circle.

Since the fisheye circle would give false contributions to the subsequent voting, we remove it from the edge image. We make one revolution around the estimated fisheye circle and remove all pixels from the edge map that are within a 3x3 matrix around the fisheye circle periphery point. In addition, we remove all edge pixels that are further away than the radius r_0 from the center point (x_0, y_0) .

Blank Central Disc. From the probability distributions for the altitude and attitude angles, we can calculate a maximum displacement of the circle center pixel and a minimum radius of curvature for the horizon. We can then remove a central disc from the edge map that could definitely not contain horizon edge pixels given the probability distributions. The reason for removing these edge pixels is for computational speed. It is a quick way to remove edge pixels from the voting that would obtain a zero or negligible weight.

Estimate Horizon Normal. For an image edge pixel $p = (x, y)$, the projection onto the unit sphere is at point P . We compute the gradient in p with 3x3 sobel filters, and define the edge direction in p as $(-\nabla_y, \nabla_x)$, i.e. normal to the gradient. We define the image point p_e as the point one pixel away from p along the edge direction. The projection of p_e onto the unit sphere is at P_e . If p is a horizon point, the vector $\overrightarrow{PP_e}$ is a tangent vector on the unit sphere lying in the plane of the projected horizon. Let \mathbf{t} be a vector of unit length in the direction of $\overrightarrow{PP_e}$. If we look at a cross section of the unit sphere, orthogonal to the vector \mathbf{t} , as in fig 3(b), we search for a second point Q in the plane of the horizon. For a certain altitude h , the radius of the horizon circle on the unit sphere is known. To find Q, we define the vector

$$\overrightarrow{OS} = \overrightarrow{OP} \times \mathbf{t} \quad (5)$$

where O is the origin in the unit sphere. We then obtain the vector

$$\overrightarrow{OQ} = \overrightarrow{OP} \cos 2\gamma + \overrightarrow{OS} \sin 2\gamma, \quad (6)$$

where γ is given by eq (1) for a certain altitude h . The points Q_{max} and Q_{min} denote the horizon points for the maximum and minimum altitudes given the probability distribution p_h in the subsequent voting.

A unit normal vector $\hat{\mathbf{n}}$ to the horizon plane can now be obtained as

$$\hat{\mathbf{n}} = \frac{\overrightarrow{PQ} \times \mathbf{t}}{\|\overrightarrow{PQ} \times \mathbf{t}\|_2}. \quad (7)$$

The pitch and roll angle estimates for the edge point p are then given by

$$\theta = \arcsin \hat{\mathbf{n}}_y, \quad \phi = -\arctan \frac{\hat{\mathbf{n}}_x}{\hat{\mathbf{n}}_z} \quad (8)$$

Note that angle estimates can easily be computed for various altitudes h . Vectors \overrightarrow{OP} and \overrightarrow{OS} remain constant, and it is only the angle γ that needs to be recomputed to get a new vector \overrightarrow{OQ} .

3.2 Probabilistic Hough Voting

For an each edge pixel p , we have shown how to compute the estimated pitch and roll angles for the horizon plane, given an assumed altitude h . It is then natural that the accumulator cells in our Hough voting is a pitch and roll angle grid. We have chosen a cell resolution of 0.25° . Min and max angles are set to $\pm 60^\circ$ as this range covers the practical angles to be estimated. In the probabilistic Hough voting scheme, we want the weight w for each vote to be proportional to the likelihood that the edge pixel is a horizon pixel given the probability distributions p_h , p_θ and p_ϕ , i.e. we want

$$w(x, y) \propto \iiint p(x, y | h, \theta, \phi) d\phi d\theta dh \quad (9)$$

Varying Altitude. Let us first analyze how the estimated attitude angles will vary with the altitude. As an example, we have picked three true horizon edge pixel points from fig 1(b). We have assumed that the camera altitude is in the range 20 to 80 m. The estimated attitude angles for the three edge pixels over this altitude range are shown in fig 4(a) as three clusters, one for each edge pixel. The estimated attitude for each edge pixel varies less than 0.2° for this rather wide relative range of altitudes. Since this attitude change is in the same order as the accumulator cell resolution, we have chosen to divide the altitude range into rather few altitude segments, $N_h = 11$, when calculating the weights. We set the weight w_h for each segment to

$$w_h = \int_{h_{min}}^{h_{max}} p_h(h) dh \quad (10)$$

where h_{min} and h_{max} are the altitude limits for each segment. For a normal distribution, we vote for altitudes in the range $\mu \pm 2\sigma$. Note that these altitude weights can be precomputed for all edge pixels.

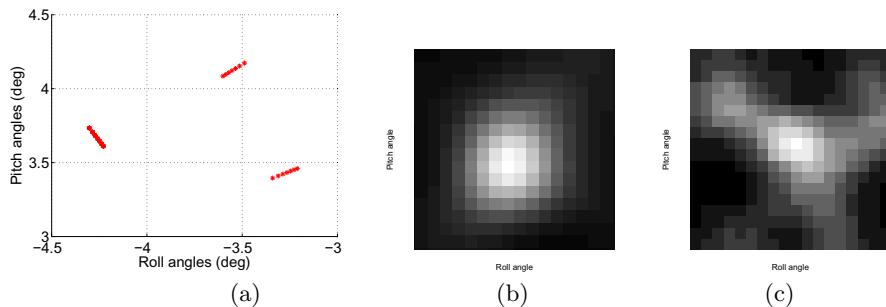


Fig. 4. (a) Estimated attitude when varying altitude h between 20 and 80 m for three edge pixels. (b) and (c): Accumulator arrays around max value for images in fig 5(a) and (h).

Voting. Using Bayes' theorem and assuming that the probability distributions for h , θ and ϕ are independent, we calculate the weights as

$$w(x, y) \propto \int p_h(h) dh \int p_\theta(\theta) d\theta \int p_\phi(\phi) d\phi = w_h w_\theta w_\phi \quad (11)$$

For each edge pixel p , we compute the estimated pitch and roll angles for each altitude h and give a weighted vote in the nearest neighbor pitch-roll cell in the accumulator array. For the internet images used, we have no prior information on the pitch and roll angles, and the weights w_θ and w_ϕ are therefore set to 1. In a true scenario, these weights should be set in accordance with p_θ and p_ϕ over the pitch-roll grid.

Attitude Estimate. In order to suppress local maxima, we convolve the values in the accumulator array with a gaussian kernel of size 7×7 and pick the index for the cell with maximum score as the attitude estimate. Since we only have a coarse camera calibration, we do not perform any interpolation to further refine the attitude estimate.

4 Evaluation

Our horizon estimation method has been evaluated using three image sequences from two internet sites [13, 14], totalling 25 images. In a cross-validation scheme, we have used one image in each sequence to perform the simplistic camera calibration described in section 2.1. For all images, we have assumed a normal distribution for the altitude with $\mu = 50$ m and $\sigma = 15$ m. The weights w_θ and w_ϕ were set to 1. For comparison, we also made a manual detection of the horizon in all images.

Our method robustly estimates the horizon in all 25 images. Results for 11 images are shown in fig 5, where the estimated horizon is overlaid as a red line.

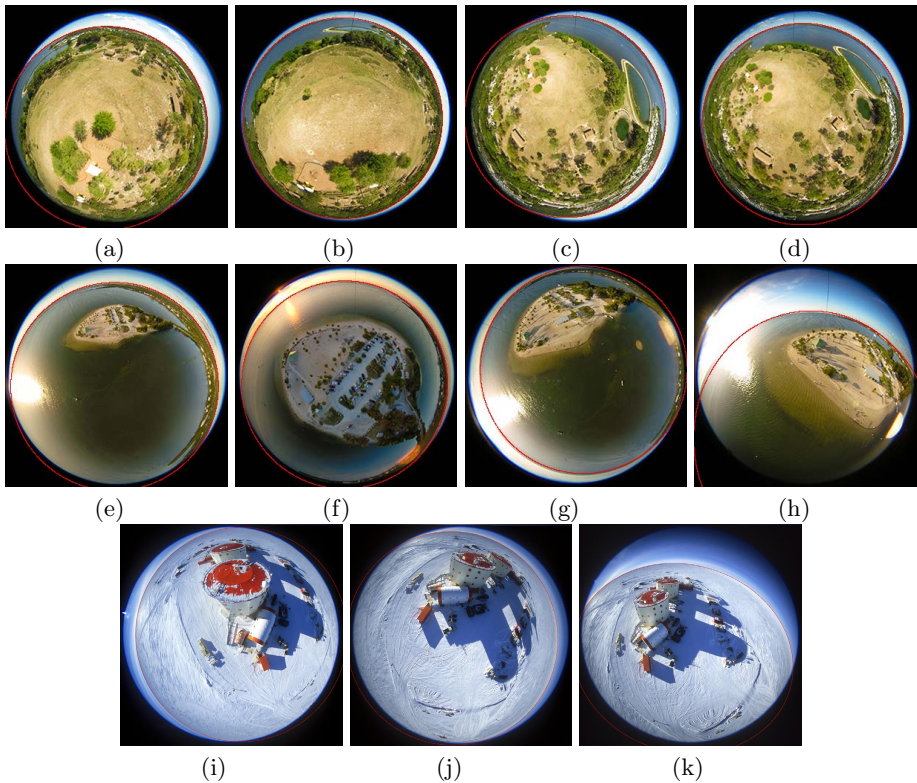


Fig. 5. Images overlaid with estimated horizon. Two top rows are from mark-marano.com, bottom row is from gdargaud.net.

For all images, the deviation between the automated and manually detected horizon corresponds to an attitude difference less than 1° . The largest attitude deviation was obtained for the image in fig 5(h). When the pitch angle is quite large, the radial lens distorsion will deform the horizon in the image plane from the ideal elliptic shape. Since lens distorsion is not accounted for in our simplistic camera calibration for the internet images, the attitude votes from true horizon edge pixels in the image will depend on the radial distance from the center of the image. This gives a wide and less accurate peak in the accumulator array, fig 4(c), compared to a case with a small attitude change from vertical which gives a well defined peak in the accumulator array, fig 4(b).

The accuracy for our attitude estimation method compares well with results in [2]. They also report an attitude estimate accuracy within 1° for synthetic images, but they have the great advantage of knowing the true camera and lens distorsion parameters.

5 Implementation

Our method is coded in C++ where the OpenCV implementations of the Canny detector and the Hough circle detector have been used as the base. Computation times are from 100-400 ms on 0.1-1 Mpixel images on a standard laptop, an Intel Core i5 CPU M560 @ 2.67 GHz.

6 Concluding Remarks and Future Work

A method for attitude estimation from aerial fisheye images through horizon detection is presented. The method is based on edge detection and a probabilistic Hough voting scheme. By letting the edge pixel votes be weighted based on the probability distributions for the altitude and pitch and roll angles, we exploit the prior knowledge of the altitude and attitude of the vehicle to make the attitude estimation more robust. An advantage is that the method does not require any sky/ground segmentation as most horizon detection methods do. Our method has been evaluated on aerial fisheye images from the internet. The horizon is robustly detected in all tested images. The deviation in the attitude estimate between our automated horizon detection and a manual detection is less than 1° .

Our horizon modelling assumes the earth to be spherical with no change in topography. In the images we have evaluated, the landscape is relatively flat and our model assumption applies. Our plan is to capture aerial fisheye images with a calibrated camera in areas with more topography changes to determine the robustness of our method. A conceivable feature to be added to make the method more robust is to vote not just for the nominal edge direction in each pixel, but for a range around the nominal direction.

Acknowledgements. This work was funded by the Swedish Governmental Agency for Innovation Systems, VINNOVA, under contract NFFP5 2010-01249, and supported by the Swedish Foundation for Strategic Research through grant RIT10-0047 (CUAS). This research has received funding from the Swedish Research Council through a grant for the project Extended Target Tracking (within the Linnaeus environment CADICS).

References

1. Shabayek, A.E.R., Demonceaux, C., Morel, O., Fofi, D.: Vision Based UAV Attitude Estimation: Progress and Insights. *Journal of Intelligent Robot Systems* (2012)
2. Demonceaux, C., Vasseur, P., Pégard, C.: Omnidirectional vision on UAV for attitude computation. In: *International Conference on Intelligent Robots and Systems* (2006)
3. Thurrowgood, S., Soccol, D., Moore, R.J.D., Bland, D., Srinivasan, M.V.: A Vision Based System for Attitude Estimation of UAVs. In: *International Conference on Intelligent Robots and Systems* (2009)
4. Bao, G., Zhou, Z., Xiong, S., Lin, X., Ye, X.: Towards Micro Air Vehicle Flight Autonomy Research on The Method of Horizon Extraction. *IMTC* (2003)

5. Hough, P.: Method and means for recognizing complex patterns. U.S. Patent 3069654 (1962)
6. McGee, T.G., Sengupta, R., Hedrick, K.: Obstacle Detection for Small Autonomous Aircraft Using Sky Segmentation. ICRA (2005)
7. Hwangbo, M., Kanade, T.: Visual-Inertial UAV Attitude Estimation Using Urban Scene Regularities. In: International Conference on Robots and Automation (2006)
8. Baatz, G., Saurer, O., Köser, K., Pollefeys, M.: Large Scale Visual Geo-Localization of Images in Mountainous Terrain. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part II. LNCS, vol. 7573, pp. 517–530. Springer, Heidelberg (2012)
9. Ying, X., Hu, Z.: Can We Consider Central Catadioptric cameras and Fisheye Cameras within a Unified Imaging Model. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004. LNCS, vol. 3021, pp. 442–455. Springer, Heidelberg (2004)
10. Scaramuzza, D., Martinelli, A., Siegwart, R.: A toolbox for Easily Calibrating Omnidirectional Cameras. In: International Conference on Intelligent Robots and Systems (2006)
11. Canny, J.: A computational approach to edge detection. PAMI 8, 679–698 (1986)
12. Fischler, M.A., Bolles, R.C.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM 24, 381–395 (1981)
13. <http://www.markmarano.com> (2012)
14. <http://www.gdargaud.net> (2012)