

Automatic Take Off, Hovering and Landing Control for Miniature Helicopters with Low-Cost Onboard Hardware

Karl E. Wenzel and Andreas Zell

University of Tübingen, Department of Computer Science, Sand 1, 72076 Tübingen

Abstract. This paper details experiments for autonomous take off, hovering above a landing place and autonomous landing. Our visual tracking approach differs from other methods by using an inexpensive Wii remote camera, i.e., commodity consumer hardware. All processing is done with an onboard microcontroller and the system does not require stationary sensors. The only requirements are a stationary pattern of four infrared spots at the start and landing site and a roll and pitch attitude estimation of sufficient quality, provided by an separate inertial measurement unit.

1 Introduction

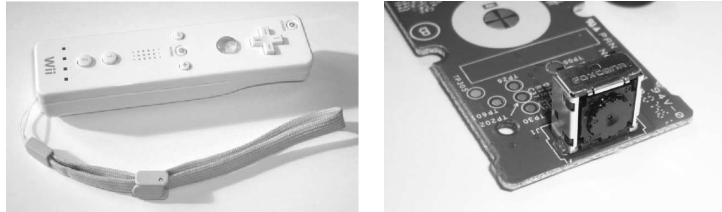
Considerable progress in high capacity batteries and energy efficient brushless motors allows for smaller and smaller miniature flying robots (MFRs). Valavanis constitutes in [1] that unmanned aerial vehicles (UAVs) should become smaller and smarter. As MFRs are a relatively new type of UAVs, they require innovative technologies in sensors and sensing strategies.

Most onboard vision-based tracking and navigation research was done on UAVs of significant size and weight using comparatively expensive industrial cameras and high performance control boards [2], [3], [4], [5]. Recent research copes with tracking and landing of MFRs, but often depends on base stations [6], [7], [8]. The accuracy achieved by tracking systems are comparable to our results [9], [10].

We demonstrated in [11] how a infrared camera can be used as main sensor for stable quasi-stationary flight control. In this paper we show how the same system can be used for autonomous take off, hovering and landing. The system works with onboard sensors only and does not require a ground station, as the entire control is achieved on an onboard microcontroller. The small positioning deviations allow for operations in narrow indoor environments.

After autonomous take off, our UAV moves to a hovering position for high-level behaviours like mapping or surveillance, which could be performed by optional controllers. After returning to the landing area, the aircraft lands autonomously on the docking station.

The aircraft we use is an miniature quadrocopter. The camera, the primary sensor in our configuration, is part of the Wii remote controller, distributed by



(a) The Nintendo Wii remote controller. (b) The internal camera sensor in its original frame.

Fig. 1. The main sensor of the system comes with the Wii remote.

Nintendo (Fig. 1). The integrated circuit provides the pixel position of tracked blobs at a high frequency. The key idea of our approach is to track a T-shaped pattern of infrared spots located at the landing place. The camera is attached to the center of the quadcopter frame, looking downwards. This guarantees clear sight to the pattern when the robot is above the landing place.

By analyzing the image of the pattern, the system is able to calculate the distance (z) and yaw angle (ψ) directly from geometric information. If the inertial measurement unit (IMU) of the aircraft provides accurate roll and pitch angles, the x and y positions can also be estimated. These four degrees of freedom are the inputs for the flight control loop.

Experiments with over 50 autonomous flight cycles were performed, in which the UAV autonomously took off, hovered for 15 seconds at 60 cm height and landed again. The standard deviation of positioning in the air is approximately 1.5 cm in x and y , below 3 cm in z position and 3° in yaw.

2 Features of the Wii Remote Infrared Camera

Size and weight are very important characteristics for miniature flying robots. The very lightweight camera we use was detached from a Wii remote controller and provides special functionalities. The sensor dimension of $8 \times 8 \times 5 \text{ mm}^3$ at a weight of 0.4 g makes it an ideal MFR onboard sensor. The Wii remote (informally known as the Wiimote) is a Bluetooth-compatible controller, designed for interfacing with the Nintendo Wii game console. The price of 40 € is relatively inexpensive considering the internal components. The internal camera is capable of blob tracking of up to four infrared (IR) sources. By eight times subpixel analysis, the native resolution of 128×96 pixels is scaled up to 1024×768 pixels. The sensor supports different modes, which differ in sensitivity and information about the IR blobs. The complete information includes dot size, intensity and bounding box. For our use, the basic mode is sufficient, providing only pixel positions. The horizontal field of view is approximately 45° and the refresh rate of 100 Hz in Bluetooth mode is adequate for fast optical tracking. When operating in I²C bus mode, 250 Hz can be achieved. Just a few electronic components

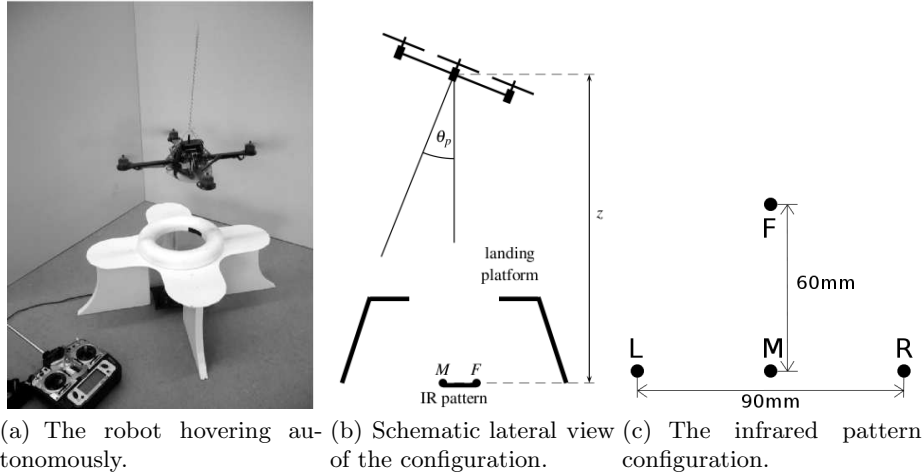


Fig. 2. The robot autonomously hovers above the landing pad. The pattern is placed below the landing platform.

are required to integrate the detached sensor in a microcontroller circuit. The camera runs at 3.3 V and requires an external synchronisation of 24 Mhz.

3 The UAV System

Our quadcopter system is an AscTec Hummingbird AutoPilot. Its diameter of 53 cm at a weight of 0.475 kg permits to fly indoor. The flight time of up to 23 minutes depends on additional payload and the flight maneuvers. (Fig. 2(a))

The AutoPilot platform provides a three-axis gyroscope, an accelerometer, a compass module, a GPS sensor and a pressure sensor. The sensors, the microcontroller and the flight control algorithm are running at 1 kHz. This ensures fast stabilisation of the quadcopter.

The ability to connect additional devices via two serial ports also permits high level applications to control the flight. The pressure sensor allows for keeping a specific height in the range of decimeters. The compass and GPS sensors are used for autonomous starting, keeping a given orientation and position in the air or to fly to waypoints. As GPS is for outdoor use only, other solutions must be found for indoor autonomy.

Our flight control processing is done onboard on an Atmel AVR 644P microcontroller. The board also acts as a gateway receiving information from the UAV and sending data to an optional base station. The base station is used for monitoring the current pose estimation and control status and can be used for varying the control loop parameters.

The camera is fixed in the center of the quadcopter frame, so the position of the aircraft can easily be calculated without additional translations.

4 Retrieving the Pose by Means of Camera and IMU

For accurate pose estimation, the information of the camera and the inertial measurement unit (IMU) of the aircraft must be combined. This results in the current position vector $\mathbf{p} = (x, y, z, \psi)^T$. The Cartesian coordinates of the landing place origin, relative to the camera x, y, z and the orientation in yaw ψ .

4.1 Pattern Analysis

The pattern we constructed measures 90 mm from the left to the right IR spot (s_1) and 60 mm from the middle to the front IR spot (s_4) (Fig. 2(c)). Each spot is represented by a single 940 nm wavelength IR LED. This configuration has proven to be of good size for indoor landing pad tracking, where the pattern must be recognized at a relatively close distance. Larger patterns would allow more precise tracking at a larger distance, but would no longer fit in the field of view when getting closer. Our pattern can be completely captured by the camera from a distance larger 15 cm. So, this vertical offset was chosen for the final landing position.

To get unambiguous position information, the disordered points \mathbf{F} (front), \mathbf{L} (left), \mathbf{M} (middle) and \mathbf{R} (right) have to be identified first. Our approach focuses on fast processing on a microcontroller, which means that we need to avoid floating point operations as well as trigonometric functions. After identifying the points, the position vector $\mathbf{p} = (x, y, z, \psi)^T$ of the aircraft, representing the position error of the UAV against the pattern, is calculated.

The x and y position relative to the center of the pattern can only be estimated when the camera orientation is known. When the quadcopter is tilted with respect to the pattern, as shown in Fig. 2(b), the pattern appears in a displaced position in the camera image. By combining the geometric and the physical roll and pitch angles, the real x and y values can be calculated with respect to the current z value. This is described in detail in [11].

5 Flight Control

The idea of our approach is to automatically take off and hover the helicopter, so that a high-level behavior can take over control. After returning to the hovering spot above the landing place our controller lands the aircraft automatically. This behavior is achieved by varying the expected altitude over time, until the desired position is reached. Fast controllers ensure sufficient position hold.

The only variable in our controller is the desired height z above the target. All other elements of \mathbf{p} should remain zero. The control algorithms are inspired by [10], where four independent controllers were operated.

The control loop is currently performed at a frequency of 20 Hz. The controller requires a recent pose estimate from the IMU. This request lasts approximately 30 ms in total. Additional 10 ms are required to send sensor information to the base station, where the current status is monitored. The remaining 10 ms

are available for retrieving sensor information, running the control algorithm and receiving configuration data from the base station. A considerably higher control frequency would be possible with an accelerated IMU request.

5.1 Height Controller

The height controller is divided into three parts. The first part adjusts the current altitude. If the robot is about to land, the desired height is decreased by 25 mm every 100 ms. This avoids the robot “falling down”. When starting, the desired height is increased every 100 ms. In both cases, the altitude is only adjusted if the difference between current and desired height is less than 45 mm.

The second and third part controls the thrust value, transmitted to the Hummingbird. The thrust control value needed to hover at a desired height has to be adjusted while flying, as it depends on the actual payload and battery charge. We obtained good results by counting the thrust value up whenever the robot is below the desired height for some cycles, and down otherwise. A fast controller is implemented as a proportional-integral-derivative (PID) loop with K_D being the major component.

5.2 Roll/Pitch and Yaw Controller

The x and y controllers are identical and were harder to derive, since the behavior response is not proportional to horizontal speed but to rotational velocities. A predictive control is required to achieve a stable position hold. By designing a cascaded control loop, where not only the speed but also the acceleration is highly weighted, the behavior obtains the desired prognostic ability.

By designing the landing pad center slightly concave, smooth landing and accurate positioning are guaranteed. The yaw controller is implemented as a PID loop with a high K_P and a small K_I . The quadcopter provides an integrated compass for controlling the coarse orientation. By measuring the angle relative to the pattern, the orientation can be kept very accurately during flight.

6 Experimental Results

In our experiments, the robot started the motors and performed autonomous take off, hovered at a distance of 60 cm above the ground for 15 seconds, then landed again and turned off the motors. Each cycle needed an average of 23 seconds. A reliable accumulator allows to repeat this procedure about 15 times. This is considerable less than our usual flight time of 15 minutes of the Hummingbird, probably due to higher power requirements for take off and permanent control maneuvers.

The Hummingbird quadcopter offers a GPS position controller and an air pressure sensor for height control. These functions were disabled during our indoor experiments. The yaw angle is controlled by the internal magnetic compass

Table 1. Controller timing of 50 autonomous flights (in seconds)

	Mean	Std. deviation (RMSD)	Minimum	Maximum
Taking off	5.45	1.76	3.05	13.32
Hovering	15.12	0.07	15.00	15.36
Landing	2.19	0.02	1.88	2.60

Table 2. Controller characteristics of 50 autonomous flights

	Mean	Std. deviation (RMSD)	Minimum	Maximum
Δx (mm)	0.01	14.46	-68	74
Δy (mm)	0.68	15.48	-68	90
Δz (mm)	-5.05	27.42	-78	90
$\Delta\psi$ ($^\circ$)	0.09	2.90	-17	20

of the quadcopter. The results were generated with assistance of the internal yaw controller of the quadcopter.

The following experimental results demonstrate the performance of the system. All position are measured using the Wii camera. Experiments with manual positioning have proven, that the position estimation achieved with the Wii camera is satisfactory and thus can be used as ground truth.

Table 1 shows the timing of the flight. One can notice that take off needs more than twice the time of landing. Take off also shows a larger root mean square deviation (RMSD). While the robot only reduces the thrust when landing, the robot has to adjust the thrust value for hovering when starting. Especially the first take off, when the accumulator charge is unknown, needs more time, leading to a large standard deviation.

Table 2 shows the controller characteristics of autonomous take off, hovering and landing without the time of standing still.

A detailed record of approximately one minute of flight is shown in Fig. 3. The plots show the position estimation sent to the base station while performing two test cycles. The sensor data is smooth and nearly noise-free. Outliers caused by external influences as infrequent reflective lights, can be filtered out easily.

Some oscillations with different frequencies still remain in the control sequence in hovering flight. A deviation in one axis leads to displacements in other directions, so accurate position hold can only be achieved by a combination of stable controllers for each axis.

The standard deviations of Δx and Δy position control are comparable. This was expected, as the controllers are identical. A greater standard deviations of Δz is obvious, as height deviations are induced in the take off and landing phases. However, a standard deviation of 3 cm and a maximum deviation below 10 cm allow for autonomous flights in narrow indoor environments. The orientation of the aircraft changed slightly during take off and is affected by balancing maneuvers and thus depends on the position changes in x -, y - and z -directions.

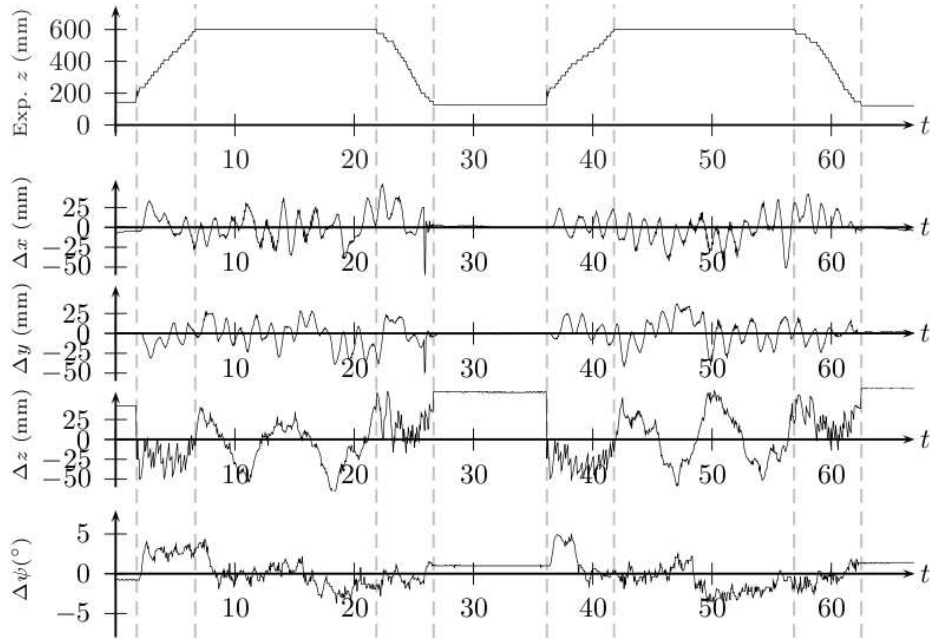


Fig. 3. Expected height (top curve) and position errors (lower four curves) of two take off, hover and landing phases.

One problem which we encountered are reflections of sunlight leading to wrong pattern interpretation, as sunlight contains a notable fraction of infrared light. A better contrast, achieved by using stronger infrared sources should allow for outdoor usage, as long as there are no specular reflections of sunlight on or around the landing pad.

7 Conclusion and Future Work

Experiments have proven that our approach offers repeatable, accurate control of a miniature unmanned aerial vehicle. The Wiimote camera provides well conditioned infrared source information. By using the Wiimote camera as main sensor, an onboard microcontroller provides sufficient processing power for pattern recognition and stable flight control. The small standard deviation of approximately 1.5 cm in x and y , below 3 cm in z position and 3° in yaw allows for safe take off, hovering and landing in narrow indoor environments. High-level autonomy tasks can be initiated from the hovering position.

The field-of-view of the Wiimote camera is large enough for tracking, even at small distances. Most disturbances caused by external light sources, except sunlight, are avoided by the infrared filter. The distance to the target is limited by the dimension of the pattern and the IR light emission. However, by using multiple IR LEDs per point or stronger IR sources, the operating distance can be

increased. An accurate roll and pitch estimation, provided by the aircraft's IMU, is essential. A small angular error leads to inadmissible position approximation. A second Wiimote camera or fusion with other sensors would yield a better positioning at larger distances.

The current frequency of 20 Hz is sufficient for robust flight control. However, by accelerating the IMU request, the control frequency could be increased considerably. A higher control frequency would advance the control accuracy. By tilting the camera during hovering, the control radius of the aircraft could be enlarged. The same system could be used for stable target tracking.

References

1. Valavanis K.P., editor. *Advances in Unmanned Aerial Vehicles. State of the Art and the Road to Autonomy*. Springer, 2007.
2. Andert F., Adolf F.M., Goormann L. and Dittrich J.S. Autonomous Vision-Based Helicopter Flights through Obstacle Gates. In *UAV'09 2nd Intl. Symp. on Unmanned Aerial Vehicles*, Reno, USA, 2009.
3. Hrabar S. 3D path planning and stereo-based obstacle avoidance for rotorcraft UAVs. In *IEEE Intl. Conf. on Intelligent Robots and Systems (IROS)*, pages 807–814, 2008.
4. Roberts J., Corke P. and Buskey G. Low-cost Flight Control System for a Small Autonomous Helicopter. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 546–551, Taipei, Taiwan, September 2003.
5. Saripalli S., Montgomery J.F. and Sukhatme G.S. Vision-based autonomous landing of an unmanned aerial vehicle. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 2799–2804, Washington, DC, USA, 2002.
6. Lange S., Sünderhauf N. and Protzel P. Autonomous Landing for a Multirotor UAV Using Vision. In *SIMPAR 2008 Intl. Conf. on Simulation, Modeling and Programming for Autonomous Robots*, pages 482–491, Venice, Italy, 2008.
7. Herisse B., Russotto F.X., Hamel T. and Mahony R.E. Hovering flight and vertical landing control of a VTOL Unmanned Aerial Vehicle using optical flow. In *IEEE Intl. Conf. on Intelligent Robots and Systems (IROS)*, pages 801–806, 2008.
8. Guenard N. and Hamel T. A Practical Visual Servo Control for an Unmanned Aerial Vehicle. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 1342–1348, 2007.
9. Mak L.C. and Furukawa T. A 6 DoF Visual Tracking System for a Miniature Helicopter. In *2nd Intl. Conf. on Sensing Technology (ICST)*, pages 32–37, 2007.
10. Gurdan D., Stumpf J., Achtelik M., Doth K.M., Hirzinger G., and Rus D. Energy-efficient Autonomous Four-rotor Flying Robot Controlled at 1 kHz. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 361–366, Roma, Italy, 2007.
11. Wenzel K.E., Rosset P. and Zell A. Low-Cost Visual Tracking of a Landing Place and Hovering Flight Control with a Microcontroller. In *UAV'09 2nd Intl. Symposium on Unmanned Aerial Vehicles*, Reno, USA, 2009.