A Robust Detection and Tracking on an Unmanned Rotorcraft for Ground Target Following  
Manoj Prabhakaran¹, Subhashini J²

Abstract—In this paper, we present the systematic design and implementation of a robust real-time embedded vision system for an unmanned rotorcraft for ground target following. The hardware construction of the vision system is presented, and an onboard software system is developed based on a multithread technique capable of coordinating multiple tasks. To realize the autonomous ground target following, a sophisticated feature-based vision algorithm is proposed by using an onboard color camera and navigation sensors. The vision feedback is integrated with the flight control system to guide the unmanned rotorcraft to follow a ground target in flight. The overall vision system has been tested in actual flight missions, and the results obtained show that the overall system is very robust and efficient.

Keyterms—Image processing, real-time systems, target detection and following, unmanned aerial vehicles (UAVs), vision systems, binary large object, CVBlob, Blo detection and tracking.

I. INTRODUCTION

UNMANNED AERIAL VEHICLES (UAVs) have recently aroused much interest in the civil and industrial markets, ranging from industrial surveillance, agriculture, and academic research to wildlife conservation [1, 2, 3, 4, 5]. In particular, owing to its vertical takeoff-and-landing, hovering, and maneuvering capabilities, the unmanned rotorcraft has received much attention in the defense and security community [6]. More specifically, an unmanned rotorcraft equipped with a vision payload can perform a wide range of tasks, such as search and rescue, surveillance, target detection and tracking, etc., as vision provides a natural sensing modality—in terms of human comprehension—for feature detection and tracking [6], [7]. Instead of vision being merely a payload, many research efforts have also been devoted to vision-aided flight control [7], [8], [9], tracking [9], [10], terrain mapping [10], and navigation [11], [12]. We note that most of the works reported in the literature, however, focus on only a certain part of vision systems for UAVs, such as hardware construction or vision algorithms. Many of them are adopted from those designed for ground robots, which are not very suitable for applications on UAVs.

To the best of our knowledge, there is hardly any systematic documentation in the open literatures dealing with the complete design and implementation of the vision system for unmanned rotorcrafts, which includes architectural and algorithmic designs of real-time vision systems. In addition, although the target tracking in video sequences has already been studied in a number of applications, there has been very little research related to the implementation of vision-based target following for UAVs.

In this paper, we present the design and implementation of a comprehensive real-time embedded vision system for an unmanned rotorcraft, which includes an onboard embedded hardware system, a real-time software system, and mission-based vision algorithms. The proposed vision scheme is integrated with the onboard navigation sensors to estimate the relative distance between the target and the UAV.

In a vision system on an unmanned rotorcrafts for target following [12], which an on-board vision system performed separate function of frame segmentation, vision module in respectively

- Frame grabber- PC/104-(plus)-standard frame grabber, a Colory 104, and
- Single board computer -a separated onboard PC104 embedded computer, Cool RoadRunner III, is employed to process the digitalized video signal and execute the vision algorithms by Real-time vision software is developed, which is running on a real-time operating system QNX Neutrino as a microkernel.

From [12], we modify which frame segmentation and vision computer are performed in only single embedded board- Friendly ARM 11 as a development platform which achieved frame segmentation and vision module, accuracy of image tracking and detection, also high speed data rates transmission of video surveillances, which is running on a real-time embedded operating system of LINUX kernel is employed as the cross-developing platform, has a microkernel that requires fewer system resources and performs more reliably and efficiently for embedded systems during runtime compared to the traditional monolithic kernel.

Finally, using the vision feedback, a two-layer target tracking control framework is utilized to control a pan/tilt servomechanism to keep the target in the center of the image and guide the UAV to follow the motion of the target An advanced and efficient vision algorithm is then proposed and implemented to realize the ground target tracking, which is suited for the UAVs.

K.Manoj Prabhakaran is with Electronics And Communication Engineering Department, SRM University, Kattankulathur, Chennai(email:atr.mano21@gmail.com).
J.Subhasini is with Electronics And Communication Engineering Department, SRM University, Kattankulathur, Chennai(e-mail:subhasinaj@gmail.com).

Adhiparasakthi Engineering College, Melmaruvathur
II SYSTEM MODEL

III. HARDWARE CONFIGURATION OF VISION SYSTEM

The hardware configuration of the proposed onboard vision system for the UAV, as shown in Fig. 1, consists of the following five main parts: a visual sensor, an image acquisition module, a vision processing module, a pan/tilt servomechanism, and video and data links.

A. Visual Sensor: Video Camera

A visual sensor is employed on board to obtain in-flight visual information of the surrounding environment of the UAV. Interesting visual information is composed of silent and dynamic features, such as the color and shape of landmarks, and motions of vehicles. A color video camera is selected as the onboard visual sensor in our system, which has a compact size and a weight less than 30 g, as well as 380 TV line resolution and 40° field of view.

B. Vision Processing Module: Vision Computer

From Fig. 1, the digitized visual signals are provided by the frame grabber and are transferred to the onboard vision computer that is the key unit of the vision system. The vision computer coordinates the overall vision system, such as image processing, target tracking, and communicating with the flight control computer. In this paper, the configuration of using two separated embedded computers in the onboard system for UAVs is proposed: one for flight control and another one for machine vision algorithms.

We choose such a configuration for onboard system because of the following reasons: 1) the computation consumption of flight control task and vision program are very heavy, which can hardly be carried out together in a single embedded computer; 2) the sampling rate of the flight control computer is faster than the vision computer, since the faster sampling rate is required to stabilize the unmanned rotorcraft; 3) the decoupled structure reduces the negative effect of data blocking caused by the vision program and flight control system and thus makes the overall system more reliable. An onboard Arm 11 embedded computer, Rapsberry pi, is employed to process the digitalized video signal and execute the vision algorithms. The core of the board is an ARM1176JZF-S (armv6k)700 MHz can dynamically increase clockspeeds, and some can temporarily reach speeds up to 1 GHz and A compact flash memory card is used to save the captured images.

C. Pan/Tilt Servomechanism

In the application of the ground target following, it is required to keep the target objects in the field of view of the camera to increase the flexibility of vision-based tracking. As such, we decide to mount the camera on a pan/tilt servomechanism that can rotate in the horizontal and vertical directions.

D. Wireless Data Link and Video Link

In order to provide ground operators with clear visualization to monitor the work that the onboard vision is processing during flight tests, the video captured by the onboard camera is transmitted and displayed in a ground.

A. Hardware Requirements

More specifically, the onboard hardware system is designed to fulfill the image processing requirements by using the commercial off-the-shelf products, such as the ARM 11 single board computer as the development platform. The hardware comprises of the ARM 11 SBC, customized IR sensitive camera, Wi-Fi Module. The embedded system target platform used in this paper is ARM1176JZF-S which based on ARM 11 embedded processor core.

B. Software Requirements

The software implementation is based on the Linux kernel and Qt framework with porting of cross-compiled CVblob(OpenCV) and Gui libraries. Owing to the use of open source technologies and choosing embedded linux as the development platform, the development cost has reduced tremendously.

The Linux of released version is not fit the hardware of embedded system, so the cross-development environment is needed to customize Linux operating system. It describes the methods and progress of transplanting the embedded Linux to the target board based on the ARM1176JZF-S processor, including the establishment of cross-compiler environment the reduction and compilation of startup code (bootloader) and Linux kernel 2.6.29 and the construction of root file system with the point focused on the structure and function of bootloader as well as the transplantation is feasible and using OpenCV(Computer Vision) library, the image identification application is developed.

The remainder of this paper is organized as follows: Sections III and IV present the development of hardware and software of the embedded vision system for a UAV, respectively. V details the vision-based ground target detection and tracking algorithms, as well as the target-following scheme based on vision signal feedback. The experimental results of the vision system obtained through actual flight tests are presented in Section VI. Finally, we draw some concluding remarks in Section VII.
control station. An airborne 2.4-GHz wireless video link is used to transmit the live video captured to the ground control station.

Fig 2 Configuration of overall vision system

IV. SOFTWARE CONFIGURATION OF VISION SYSTEM

Based on the proposed hardware system, the configuration of the onboard vision software system is presented. The purpose of the vision software system is to coordinate the work of onboard devices and implement vision algorithms. Since the vision software system targets for real-time applications and runs in an embedded ARM computer, LINUX, a real-time embedded operating system is employed as the developing platform. LINUX has a microkernel that requires fewer system resources and performs more reliably and efficiently for embedded systems during runtime compared to the traditional monolithic kernel. Because of the numerous economic and technical benefits, we are seeing strong growth in the adoption of Linux for embedded devices. This trend has crossed virtually all markets and technologies. LINUX has been adopted for embedded products in the worldwide public switched telephone network, global data networks, wireless cellular handsets, and the equipment that operates these networks. Linux has enjoyed success in automobile applications, consumer products such as games and PDAs, printers, enterprise switches and routers, and many other products. The adoption rate of embedded Linux continues to grow, with no end in sight.

Some of the reasons for embedded Linux are as follows: Linux has emerged as a mature, high-performance, stable alternative to traditional proprietary embedded operating systems. Linux supports a huge variety of applications and networking protocols, deployed without the royalties required by traditional proprietary embedded operating systems. Linux has attracted a huge number of active developers, enabling rapid support of new hardware architectures, platforms, and devices. The vision software program coordinates tasks such as capturing video, controlling pan/tilt servomechanism, and performing the vision detecting and tracking algorithms. To make the vision software system easy to design and robust to perform, the entire vision software system is divided into several main blocks.

V. VISION-BASED GROUND TARGET FOLLOWING

To realize the vision-based ground target detection, many vision approaches have been proposed worldwide, such as template matching [2], [3], background subtraction [4], [5], optical flow [6], [7], stereo-vision-based technologies [8], and feature-based approaches [8], [9], [10], [11]. In this paper, a sophisticated vision-based target detection and tracking scheme is proposed, as shown in Fig. 3, which employs robust feature descriptors and efficient image-tracking techniques. Based on the vision sensing data and navigation sensors, the relative distance to the target is estimated. Such estimation is integrated with the flight control system to guide the UAV to follow the ground target in flight.

Blob Tracking Modules

CVBlob is a library for computer vision to detect connected regions in binary digital images. CVBlob performs connected component analysis (also known as labeling) and features extraction. In cvblob the blob tracking module algorithm is used for video surveillance system.

A blob (alternately known as a binary large object, basic large object, BLOB, or BLOB) is a collection of binary data stored as a single entity in a database management system. Blobs are typically images, audio or other multimedia objects, though sometimes binary executable code is stored as a blob. Database support for blobs is not universal. "Blob" was originally used as a term for moving large amounts of data from one database to another without filters or error correction. This speeded up the process of moving data by putting the responsibility for error checking and filtering on the new host for the data. The act of moving huge amounts of data was called "blobbing", as in the sentence, "Just blob that data over." This came about by the image of somebody grabbing fistfuls of material from one container and putting it in another without regard to what was in the "blob" they were grasping. In the area of computer vision, blob detection refers to visual modules that are aimed at detecting points and/or
regions in the image that differ in properties like brightness or color compared to the surrounding. There are two main classes of blob detectors (i) differential methods based on derivative expressions and (ii) methods based on local extrema in the intensity landscape. With the more recent terminology used in the field, these operators can also be referred to as interest point operators, or alternatively interest region operators (see also interest point detection and corner detection). There are several motivations for studying and developing blob detectors. One main reason is to provide complementary information about regions, which is not obtained from edge detectors or corner detectors. In early work in the area, blob detection was used to obtain regions of interest for further processing. These regions could signal the presence of objects or parts of objects in the image domain with application to object recognition and/or object tracking. In other domains, such as histogram analysis, blob descriptors can also be used for peak detection with application to segmentation. Another common use of blob descriptors is as main primitives for texture analysis and texture recognition. In more recent work, blob descriptors have found increasingly popular use as interest points for wide baseline stereo matching and to signal the presence of informative image features for appearance-based object recognition based on local image statistics. There is also the related notion of ridge detection to signal the presence of elongated objects. This is an attempt to document the OpenCV Video Surveillance facility. It is at the moment, just a collection of insights from working with the under-documented code, and not a complete full documentation. In fact, at the moment this document will only focus on the Foreground / Background Discrimination part of the complete algorithm.

2. A blob detector which groups adjacent "foreground" pixels into blobs, flood-fill style.
3. A blob tracker which assigns ID numbers to blobs and tracks their motion frame-to-frame.

The blob tracking system includes 5 modules as depicted on diagram

V. EXPERIMENTAL RESULTS

To verify the proposed vision system, multiple tests of the complete system were conducted. During these tests, the proposed vision-based unmanned helicopter Shelion was hovering autonomously at a certain position. If the moving target entered into the view of the onboard camera, the target would be identified and tracked in the video sequence by the vision system automatically. Based on the vision information, the pan/tilt servomechanism was controlled to keep the target in a certain position in the image. Then, the operator can command the UAV to enter into the following mode, in which the UAV followed the motion of the target autonomously based on the estimated relative distance.

The experimental results of the vision-based target detection and tracking in flight are shown in fig 6, which indicate that the proposed vision algorithm could effectively identify and track the target in the video sequence in the presence of the disturbance of unknown motion between the UAV and the target. In the fig 6 shown as experimental results of the Frames per second of FG/BG detection, blob detection and tracking of a vision system. From fig 6.1 shown as input image of the vision based system. From fig 6.2 and 6.3 we shown as a 1st frame per second of foreground/background detection, blob detection and tracking of a vision system respectively. From fig 6.4 and 6.5 we shown as a 2nd frame per second of foreground/background detection, blob detection and tracking of a vision system respectively. From fig 6.6 and 6.7 we shown as a 3rd frame per second of foreground/background detection, blob detection and tracking of a vision system respectively.
V. CONCLUSION

In this paper, we have presented the comprehensive design and implementation of the vision system for the UAV, including hardware construction, software development, and an advanced ground target seeking and following scheme. Multiple real flight tests were conducted to verify the presented vision system. The experimental results show that this vision system is not only able to automatically detect and track the predefined ground target in the video sequence but also able to guide the UAV to follow the motion of the target in flight. The robustness and efficiency of the developed vision system for UAVs could be achieved by the current system. Our future research focus is to utilize the system for implementing vision-based automatic landing of the UAV on a moving platform in an environment without GPS signals.

REFERENCES


From fig 6.2, 6.4, 6.6, we shown as a foreground/background detection convert frames from an input image, and fig 6.3, 6.5, 6.7, we shown as rectangular box in blue colour of their binary co-ordinates of frame to identify the motion of object by the module of blob detection and tracking of a vision system.