

Design and implementation of the upper computer of airport intelligent bird-driving system

Yutong Chen^{1*}, Yufen Liu², Yuqin Chen¹, LiCheng Liu³

1. GuangZhou Civil Aviation College

2. Civil Aviation University of China

3. Guangdong University of Technology

ytchen520@126.com, 160915978@qq.com, bonnie310@163.com

Abstract—The single, repetitive and aimless approach to the driving of birds in the existing airport field caused the birds to quickly "develop" the inertial indication, which was almost "immune" to the equipment, and put heavy pressure on the work of the airport. Nowadays, with the increasing number of flights, how to design and develop an intelligent system for driving birds, so as to make driving birds more scientific, intelligent and effective has become the research focus of the current work on driving birds. The intelligent bird drive system studied in the project links the motion of the bird drive equipment with the bird situation information, and combines the current popular Internet of things technology to effectively avoid the blind startup of the bird drive equipment caused by human wrong commands. It not only responds to the adaptability of birds to the drive equipment, but also improves the effect of the drive equipment.

Keywords—Bird repellent device; Inertial indication; Internet of things deep learning

I. INTRODUCTION

With the rapid development of aviation industry, people pay more and more attention to the damage and loss caused by aviation disaster. A large proportion of aviation disasters occur during takeoffs and landings, and the main cause of this type of aviation disaster is bird strikes [1].

Bird strikes, also known as bird strikes, are accidents caused by collisions between aircraft operating at high speeds in the air or on the ground and birds in the air. Bird strikes have the characteristics of suddenness and variability. The flight accidents caused by bird strikes not only bring economic loss, but also bring life safety hazards, which always threaten the personal safety of passengers and aircrew. Slight bird strikes cause damage to aircraft parts, and severe bird strikes can lead to the destruction of aircraft [2]. It can be seen that from the perspective of the harm caused by bird strikes to aircraft, and from the perspective of its development trend, bird strikes cannot be ignored. Effective methods must be found to curb the occurrence of such events, that is, how to achieve "efficient" drive birds and change the status quo of driving birds from "passive" to "active". In view of this problem, and under the background of the integration of military and civil, our school bird strike disaster application research center has jointly applied for many projects together with a department of the Chinese Air Force, Civil Aviation University of China, Guangdong University of Technology and other units. Aiming at this problem, this paper proposes to build a bird driving system based on the Internet of things, which can solve the

resource cost problem caused by the increase of device load, and lay a foundation for the promotion of future projects.

II. SYSTEM ARCHITECTURE DESIGN

A. The Concept of Large-scale Network

The main purpose of establishing the IoT system network is to provide network support for the existing system, which is the hardware foundation and the core part of building the remote control system platform. This system according to the requirements of the particularity of the airport, the use of a wireless network for data transmission, network topology structure of the system is shown in figure 1, it USES the Zigbee network and short wave form in the airport area wireless network interactive way bird information transmission network, the gas gun, titanium ray guns, double bang firecracker, block bird net and voice QuNiao fusion in wireless QuNiao network, form a complete bird information query, data processing and intelligent linkage QuNiao efficiently regulate QuNiao system.

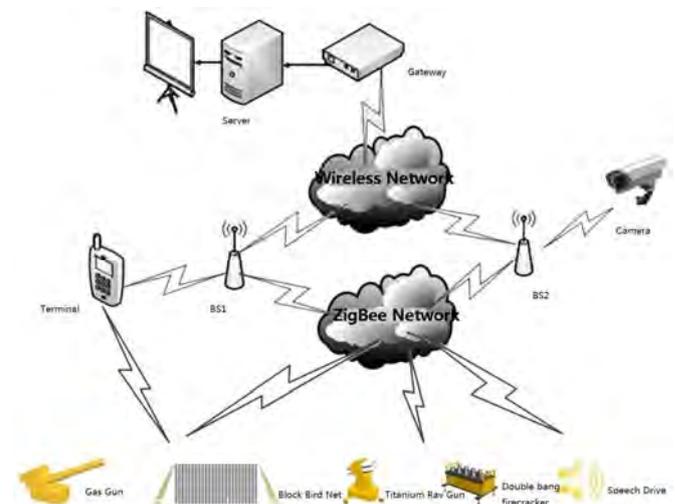


Fig. 1. Schematic diagram of system architecture.

ZigBee technology is a communication technology based on IEEE802.15.4 protocol standard [3]. It has become a research hotspot in wireless communication networks. ZigBee technology is characterized by low speed, low power consumption, short time delay, high capacity and low cost.

ZigBee network node devices in the WSN network [5] are divided into three common types: network coordinator nodes, routing nodes and terminal nodes. Any network in topology form needs a network coordinator node, and it is unique. The routing node can carry out data forwarding and network extension, and the terminal node is generally used for signal acquisition and processing.

A wireless ZigBee module is installed on the hand-holding machine for bird situation information acquisition, which can communicate and control with Gas Gun, Block Bird Net, Titanium Ray Guns, Double Bang Firecracker and Speech Drive connected to the ZigBee network. The base station can communicate with the main station through the short-wave network. After receiving, parsing and processing the command information of the platform received from the short-wave network, it will be sent to the relevant bird-driving equipment through the ZigBee network for the bird-driving action control. Gas Gun, Block Bird Net, Titanium Ray Guns, Double Bang Firecracker and Speech Drive can also be able to send the status information through the ZigBee network to the base station. The base station analyzes the information received and uploads to the server platform by short wave communication. The server online records whether the working state of each device is normal or not, and if there is abnormal, it will make a report to the police response[4]. In addition, relay nodes can communicate with each other to prevent repeated transmission of information, causing congestion of communication lines, reduce communication load of the main station, prevent unnecessary communication pressure on the main station of the server platform, and display the work status information of the relay station on the software platform.

III. DESIGN OF SERVER UPPER COMPUTER

A. Upper computer interface design

In order to make the iot system more user-friendly, it is necessary to create the visual software of airport bird monitoring system. According to terminal bird-driving personnel needs, based on past observations of bird data, using advanced computer software development technology, put the pictures of the birds, shape characteristics, feather characteristics, living habits, flying height, the threat level of the aircraft and bird populations, quantity, activity and other relevant information input to the computer. The next step is to design and develop the system. Considering the software of portable mobile device needs to be developed in the later stage of project, the development tool adopts the scheme of QT 5.7.0 + ACCESS 2010, so as to develop the upper computer software of airport bird-driving system with simple interface, easy to use, portability, expansibility and good maintenance. Through the analysis of the system's detailed demand, the following functions are designed and implemented for the system. The system use case diagram is shown in figure 2:

1) The user of the system is divided into different levels and authority control, specifically, different users are given different reasonable permissions to ensure the security of the system.

2) Management of basic information: management personnel of airport bird information data can maintain the information stored in the system, including adding new information and deleting existing error information.

3) Management of query: general staff of the airport can query the bird information of the bird information query management system, which mainly provides convenience for the daily bird strike prevention work of the airport.

4) Be able to maintain bird information in the database at any time.

5) The system database can be updated at any time to add the latest bird information to the system.

6) Be able to send equipment control commands and receive equipment status information through established network.

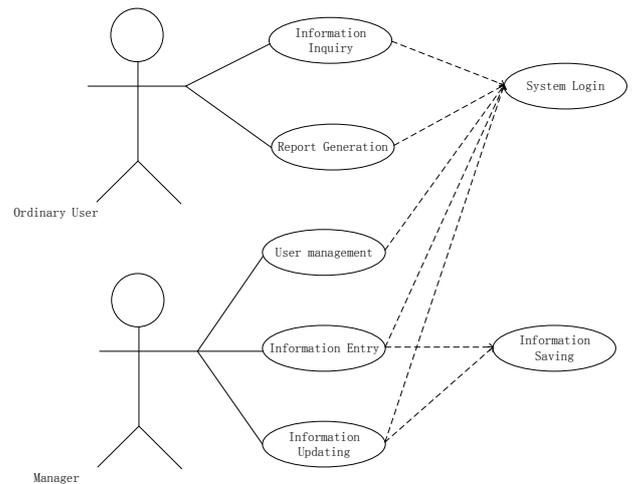


Fig. 2. System use case diagram.

B. Establishment and connection of bird database

According to the long-term cooperation and investigation between the bird strike disaster application technology research center of guangzhou civil aviation vocational and technical college and a department of the Chinese air force, a total of 38 species of birds were involved in the airport, including birds such as egret, cattle back egret, green wing duck, sandpiper, red falcon and turtle dove. The related birds are shown in FIG. 4.



Fig. 3. Images of birds involved in the field

Qt mainly operates the database through QSql module, so a line of code "Qt += SQL" needs to be added in the project file (.pro file) to use the database module. In order to read the data in the database in the main window, you can create an associated database class firstly and then instantiate it in the

main window. The database class implementation code are as follows:

```

data_base::data_base(QObject *parent) : QObject(parent)
{
    db= &QSqlDatabase::addDatabase("QODBC");
    //Set the database Driver
    db->setDatabaseName("connectDatabase");
    bool ok = db->open();
    if (!ok)
    {
        qDebug()<<"Database open failed. ";
        qDebug()<<db->lastError();
        this->connected=0;
    }
    else
    {
        qDebug()<<"Database open success. ";
        this->connected=1;
        row=new QSqlRecord();
        model=new QSqlTableModel(this,*db);
        model->setTable("BirdInfo");
        model->setSort(0,Qt::AscendingOrder);
        //Set column name
        model->setHeaderData(1,Qt::Horizontal,tr("Chinese
name "));
        model->setHeaderData(2,Qt::Horizontal,tr("English
name "));
        model->setHeaderData(3,Qt::Horizontal,tr("Hazard
rating "));
        model->setHeaderData(7,Qt::Horizontal,tr("Lowest
altitude "));
        model->setHeaderData(8,Qt::Horizontal,tr("Highest
altitude "));
        model->setHeaderData(10,Qt::Horizontal,tr("Note"));
        model->setHeaderData(11,Qt::Horizontal,tr("The
cluster type "));
        model-
>setEditStrategy(QSqlTableModel::OnManualSubmit);
        if(!model->select())
        {
            qDebug()<<"The database of model set failed : "<<
model->lastError();

```

```

        }
    }
    else
    {
        qDebug()<<"The database of model set success : ";
    }
}
}
}

```

C. Communication interface design

The program written by Qt is used as the upper computer software. If you need to communicate with the lower computer via USB, you need to communicate with the serial port in Qt. The serial port is the bridge connecting the upper machine with the lower machine, so its design is related to the success or failure of the project to a certain extent. To communicate with the underlying machine in the main window, you can create a QSerialPort serial object and then manipulate it. The system provides users with two connection modes, namely automatic connection and manual connection, in which the manual connection code are as follows:

```

bool communication_layer::auto_connect_cood(){
    bool state=false;
    #ifndef CONNECT_WITHOUT_ZIGBEE
    foreach(const QSerialPortInfo &info,
QSerialPortInfo::availablePorts())
    {
        qDebug()<<"Name:"<<info.portName();
        qDebug()<<"Description:"<<info.description();
        qDebug()<<"Manufacturer:"<<info.manufacturer();
        serial->setPort(info);
        if(serial->open(QIODevice::ReadWrite))
        {
            //Set Baudrate
            serial->setBaudRate(115200);
            // Set Data bits
            serial->setDataBits(QSerialPort::Data8);
            // Set Check digit
            serial-
>setParity(QSerialPort::QSerialPort::OddParity);
            // Set Flow control
            serial-
>setFlowControl(QSerialPort::NoFlowControl);
            // Set stop bits
            serial->setStopBits(QSerialPort::OneStop);
            qDebug()<<info.portName()<<"open success";

```

```

state=true;
connection_state=true;
emit zigbee_connected_sig();//发送连接成功信号
break;
}
else
{
qDebug()<<"open failed";
}
}
#else
qDebug()<<"open success";
state=true;
connection_state=true;
emit zigbee_connected_sig();
#endif
return state;
}.

```

IV. DESIGN OF MESSAGE PROCESSING MODULE

The message exchanged between the upper computer and the lower machine is the device open command sent by the host and the device state information sent from the machine. With the increase of the number of devices mounted in the field, the data collected by the sensor also increases exponentially. A large amount of data will inevitably lead to problems such as increased CPU occupancy and large memory overhead. In order to avoid bottlenecks at the later stage of the project, which leads to doubling of hardware costs, the design of the message processing module refers to the encoding and decoding mode of 4G mobile communication, namely asn.1 encoding and decoding.

Asn.1 includes several standard coding rules, such as basic encoding rules (BER) - x.209, specification encoding rules (CER), identification name encoding rules (DER), compression encoding rules (PER) and XML encoding rules (XER). These coding rules describe how the values defined in asn.1 are encoded for transmission, regardless of factors such as the computer, programming language, or how it is represented in the application. Asn.1's encoding method is more advanced than many competing markup systems, which support fast and reliable transmission of expandable information - an advantage in wireless broadband, hence the introduction of the codec technology in the project.

Where PER is an asn.1 encoding rule designed on the basis of BER, its advantage is that it can effectively reduce the encoding and decoding overhead of messages [5]. Considering the increasing data volume overhead at the later stage of the

project, the system adopted PER coding rule. The PER code structure is shown in figure 5. Its essence is the "nesting of coding on demand" structure. When relevant information needs to be transmitted, resources can be reallocated.

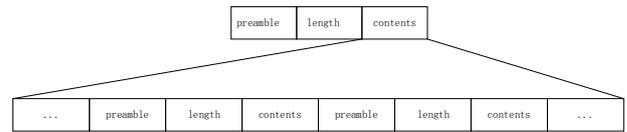


Fig. 4. PER code structure diagram

According to asn.1 encoding and decoding rules, the encoding rules as shown in table 1 can be designed, in which the data head and tail are of type char, the commands and data are of typedef, and "int bits_justons" are used to indicate the missing bits in a binary string.

TABLE I. ASN.1 CONVERSION RULE

Asn.1 type	C type
HEAD	char
COMMAND	typedef struct COMMAND_s{ uint8_t* buf; int size; int bits_unused; }
DATA	typedef struct DATA_s{ uint8_t* buf; int size; int bits_unused; }
TAIL	char

V. THE EXPERIMENTAL TEST

1) Database connection test. In the main interface, the bird information in the field can be accurately queried through the query function, which helps the field staff to make a better judgment.



Fig. 5. successful bird information query

2) Communication interface test. When the USB cable is connected with the lower computer, the connection on the main interface is realized through the serial port connection function. When the corresponding equipment is turned on, the sending command data can be observed in the printing window of the serial port, which indicates that the serial port is working

normally and can communicate with the lower computer. When the USB is disconnected from the lower computer, the software cannot be connected, which is expected.



Fig. 6. Test diagram for serial connection

ACKNOWLEDGMENT

This paper designs and implements the upper computer of airport intelligent bird-driving system, and puts forward some

improvement methods for its previous problem of sharp increase of data quantity after the increase of equipment quantity. Finally, the effectiveness of these methods is verified through simulation and test results, which is of great significance for the system to mount more bird driving equipment in the future. In the next step, the deep learning method will be combined to explore the smart bottom line of the system and help the airport to do a good job of security.

REFERENCES

- [1] NING Huan-sheng, CHEN Wei-shi, MAO Xia, et al. Bird-aircraft strike avoidance radar[J]. IEEE Aerospace and Electronic Systems Magazine, 2010, 25(1): 19-28..
- [2] David Bradbeer, Eoin Ryan, Ian Witter. Wildlife Hazard Management Handbook (Second Edition) [M]. Airports Council International, 2013.
- [3] Alliance Z B. ZigBee standards overview[J]. Retrieved from www.zigbee.org, 2017.
- [4] Rappaport T S, Annamalai A, Buehrer R M, et al. Wireless communications: past events and a future perspective[J]. IEEE Communications Magazine, 2002, 40(5): 148-161.
- [5] Stables E R, New N D. Birds and aircraft: the problems[C]. The Problems of Birds as Pests: Proceedings of a Symposium Held at the Royal Geographical Society, London, on 28 and 29 September 1967. Elsevier, 2013.
- [6] Li M, Wei H. Overview of Physical Bird Driving Technology[J]. Agricultural Engineering, 2013: S1