

蛋雞飼料調配及狀況監控系統

指導老師：鄭煜輝 組員：徐千善、廖恩彬、陳昶渝、黃宗樑、謝宗霖

朝陽科技大學資訊與通訊系

作者聯絡方式：413 台中市霧峰區吉峰東路 168 號

Tel: (04) 23323000 ext. 7864 / 5352

Fax: (04)23305539

E-mail: yhcheng@cyut.edu.tw

摘要

近年來，隨著科技快速的進步，在自動養殖的應用越來越受到矚目，但在蛋雞養殖這一塊，仍然是非常稀少，仍需要依靠大量人力去執行養殖的動作，加上政府的政策導致沒有很多人去執行蛋雞自動化養殖場，因此本專題實現了遠端的雞隻監控並監測到死雞時發送通知，以及飼料的投放器。透過遠程的操作方式，以此來減輕養殖人員的在工作上的需求。

關鍵字：雞隻監控、死雞偵測、飼料車、影像辨識

Abstract

In recent years, with the rapid progress of technology, the application of automatic breeding has been attracting more and more attention, but in the field of chicken breeding, it is still very rare and still requires a lot of manpower to perform breeding actions. In addition, government policies have resulted in few people operating automated chicken breeding farms. Therefore, this project has implemented remote chicken monitoring and monitoring of dead machines and notification of feed dispensers. By adopting a remote operation mode, this can reduce the demand for breeding personnel in their work.

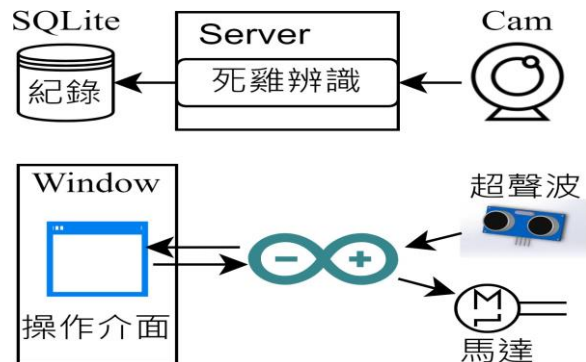
Keywords: chicken monitoring, dead chicken detection, feeder truck, image recognition

1. 前言

傳統式的養殖場全都是以傳統式的飼料車需要人力下去推動而且機器十分老舊，以及依靠養殖人員使用人眼在一個小雞籠裡面有三至四隻雞的情況下下去辨識雞隻狀況，所以在本次專題製作自動化飼料機以及透過影像辨識去辨識養殖場中是否有死雞的存在。這樣方可降低人力負擔，解決傳統飼料車所帶來的不便，同時透過遠距離操控的方法。以及透過影像辨識來解決長期使用大量人力檢查雞隻問題的，並透過記錄或是通知來進行人工處理雞隻處理的動作。

2. 系統架構

2.1 系統架構



2.2 系統說明

本專題使用的系統硬體有電腦、Arduino UNO[1]、28BYJ48 步進馬達[2]、L298N 馬達驅動模組[3]、直流馬達[4]、HC-SR04[5]、Webcam[6]，系統軟體套件有 PyCharm[7]、Arduino IDE[8]、Bootstrap5[9]、Django[10]、SQLite[11]、YOLOv4[12]。透過 Webcam 將影像傳入 Server 顯示影像並進行影像處理，並將紀錄顯示於網頁上。則飼料機透過 UART[12]傳輸方式來控制飼料機的動作。

2.3 硬體

2.3.1 Arduino UNO [1]

我們使用 Arduino UNO 控制板是因為我們要用來控制飼料機動作，相較於 Arduino Mega 控制板來說，Mega 控制板的效能溢出、與用的端口也不用那麼多，而且價格也較貴。



圖 1. Arduino Uno 開發版

2.3.2 28BYJ-48 步進馬達 [2]

我們這邊使用 28BYJ-48 步進馬達是用來控制我們飼料量投放的大小。

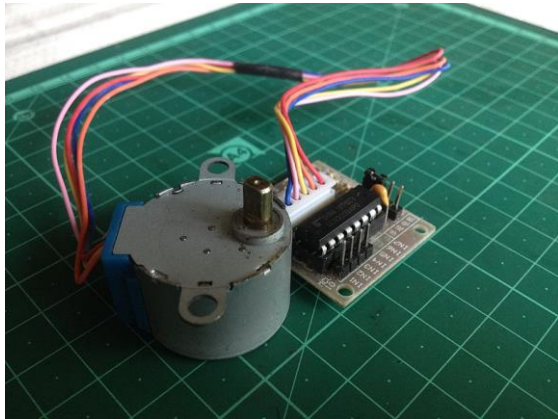


圖 2. 28BYJ48 步進馬達

2.3.3 L298n 驅動馬達模組 [3]

我們使用這一個驅動板來控制我們直流馬達前進或後退。

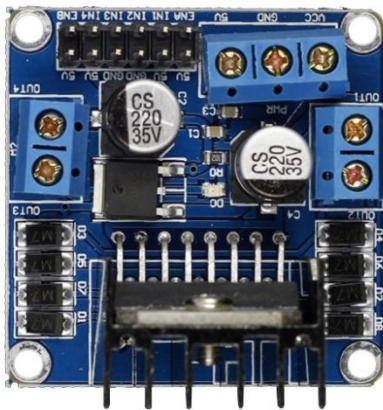


圖 3. L298N 驅動馬達模組

2.3.4 直流馬達 [4]

我們使用這顆直流馬達來控制飼料機的移動功能。由於移動的物品有一定的重量，所以選擇這一顆 9-12V 的直流馬達。



圖 4. 9V-12V 直流馬達

2.3.5 HC-SR04[5]

HC-SR04 是一種超聲波測距感測器，可以用於測量距離。它通過發射超聲波訊號並測量返回訊號的時間差來求出距離。



圖 5. HC-SR04 示意圖

2.3.6 Webcam [6]

鏡頭的部分是隨便鏡頭都可以，為了節省金費所以使用本來就擁有的鏡頭來做實驗。



圖 5. Webcam

2.4 軟體

2.4.1 PyCharm [7]

PyCharm 是一個功能強大的 Python 開發環境 (IDE)，由 JetBrains 公司開發。它提供了豐富的工具和功能，可以讓 Python 開發人員更加高效、輕鬆地進行程序開發。

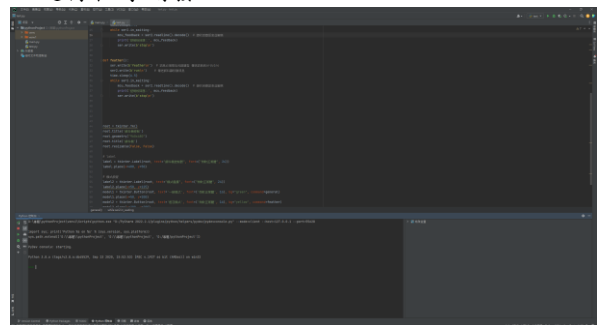


圖 7. PyCharm 開發環境

2.4.2 Arduino IDE [8]

Arduino IDE 是 Arduino 微控制器的官方開發環境 (IDE)。它是一款免費、開源的軟件，可以讓開發人員輕鬆地編寫、測試和上傳程序到 Arduino 微控制器。

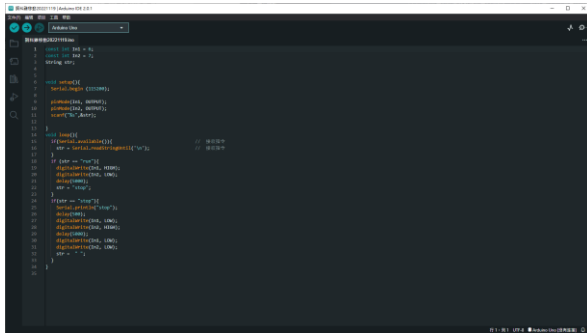


圖 8. Arduino IDE 開發環境

2.4.3 Bootstrap5 [9]

Bootstrap 是一個前端框架，可以讓開發人員快速地構建網站和應用。它提供了豐富的 HTML、CSS 和 JavaScript 組件，可以讓開發人員輕鬆地構建響應式、移動優先的網站。



圖 9. Bootstrap 5

2.4.4 Django [10]

Django 是一個 Web 應用框架，可以用於構建各種 Web 應用。它使用 Python 語言編寫，提供了許多實用的功能，例如模板系統、表單系統、管理後臺等，使開發人員能夠快速、高效地構建 Web 應用。

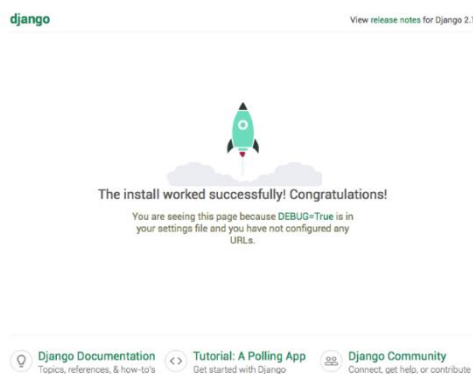


圖 10. Django

2.4.5 SQLite[11]

SQLite 是一個輕量級的關係資料庫管理系統，非常適合用於嵌入式設備和移動應用。它具有輕巧、快速、低功耗等優點，使得 SQLite 成為許多應用的首選資料庫。



圖 11. SQLite

2.4.6 YOLOv4[12]

YOLOv4 是一種計算機視覺模型，用於對圖像中的目標進行分類、定位和識別。它是 YOLO (You Only Look Once) 模型。YOLOv4 在性能上有顯著的提升，具有更快的速度和更高的精度。同時，它還支援多種新的架構和優化方法，使得它能夠在更複雜的情況下仍然取得良好的性能。



圖 12. YOLOv4

2.4.7 UART[12]

UART 是一種用於在不同電子裝置之間傳送資料的通訊協定。它使用串列傳輸接口，可以支援高達 4 Mbps 的傳輸速率，並且具有簡單易用、寬頻傳輸、高傳輸效率、和低成本的優點。

2.5 系統功能與實作

2.5.1 死雞辨識系統

我們使用 4 組 500 張死雞不同場景的圖片，以及 1 組死雞與活雞混合的圖片進行訓練，死雞檢測模型的準確率(mAP)達到 98.8%。那從圖 13 中可以看到，我們網頁主要提供雞籠內狀況的影像串流以及系統辨識死雞的時間、位置的紀錄。以及在途中左下角可以看到會發送 Line 通知至手機。

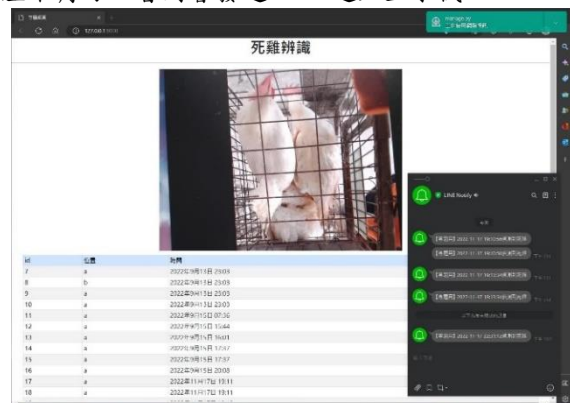


圖 13.系統示意圖

2.5.2 飼料車系統

圖 14 飼料投放 UI 介面我們使用 Python 的 GUI 工具包 Tkinter 來製作，我們使用超音波偵測的方式透過回彈的距離來判斷飼料槽裡的飼料厚度並控制飼料的投放大小，然後使用 UART 傳輸技術來進行電腦與 Arduino UNO 之間的溝通。

圖 15 飼料車的部分，我們自己 3D 列印製作我們的圖 16 飼料投放盒，我們利用圖 17 螺旋軸，讓它可以有效的控制飼料大小並防止有別於傳統飼料盒的堵塞問題。



圖 14. UI 介面



圖 15. 飼料車示意圖

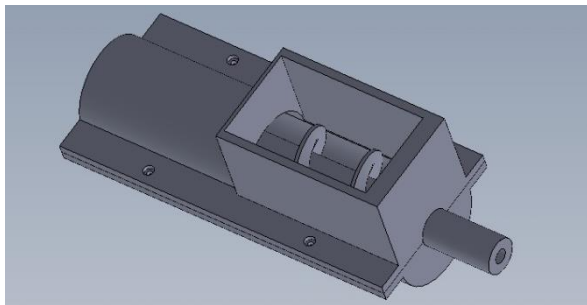


圖 16. 飼料盒 3D 模型

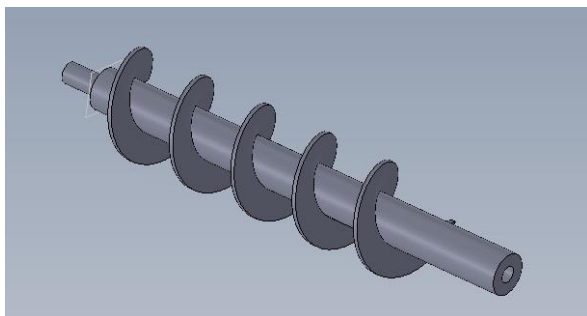


圖 17. 螺旋軸

2.5.3 功能說明

死雞辨識系統：可以從網頁上看到即時影像以及紀錄，同時配合 Line Notify 進行即時的訊息傳送。(圖 13 所示)

飼料車系統：透過控制器去操作要投放的飼料模式，進行定速的飼料大小投放。

功能：

死雞辨識系統：

1. 即時影像：在網頁上有雞籠內的影像串流，也可以選擇看到正在辨識的影像串流。
2. 辨識死雞：使用 YOLOv4 訓練出的模型，進行定時的辨識。
3. 傳送訊息：即時傳送 Line 訊息通知，通知內容有時間、地點。
4. 死亡紀錄：在網頁上可以看到死雞位置時間紀錄(辨識到的雞籠位置、死亡時間)。

飼料車系統：

1. 飼料車自動投放：我們在飼料車投放訂製了兩個模式，一個是平常投放雞隻飼料的一般模式；另外一個是在雞隻處於退羽毛季節時使用的退羽模式。

3. 參考文獻

3.1.1 資料來源

- [1] Mikechenx. (2021-11-11). YOLO v4 模型訓練實作. Retrieved from <https://ithelp.ithome.com.tw/articles/10282549?sc=pt>.
- [2] Liu, H.-W., Chen, C.-H. (ORCID), Tsai, Y.-C. (ORCID), Hsieh, K.-W., & Lin, H.-T. (2021, May 21). Identifying Images of Dead Chickens with a Chicken Removal System Integrated with a Deep Learning Algorithm. Department of Bio-Industrial Mechatronics Engineering, National Chung Hsing University, Taichung 402, Taiwan. <https://www.mdpi.com/1424-8220/21/11/3579>
- [3] 胡肄农, 窦少春, 王立方, 陆昌华. (2002). 蛋鸡规模化养殖场的自动监控系统. 江苏农业学报, (3). 江苏省农业科学院农业资源与环境研究中心, 扬州市银河科技公司. Retrieved from <http://www.cqvip.com/qk/92499x/200203/6821028.html>
- [4] 李卫国, 颜斌, 储新生. 蛋鸡规模养殖现状调查报告[J]. 中国畜禽种业, 2022, 18(1): 12-13. <http://xq.aiijournal.com/CN/>
- [5] 王玲, 王超峰, 郑奎. 基于 ARM 的蛋鸡养殖场网络视频监控系统设计[J]. 农业机械学报, 2012(2): 186-191. Retrieved from <http://www.cqvip.com/qk/90304x/201202/40969>

- 步進電機帶 ULN2003 驅動器.jpg [Image]. Retrieved from https://commons.wikimedia.org/wiki/File:28BYJ-48_unipolar_stepper_motor_with_ULN2003_driver.jpg
- [3] 超聲波. (2016, March 23). 超聲波.jpg [Image]. Retrieved from <http://wiki.sunfounder.cc/index.php?title=File:Ultrasonic.jpg>
- [4] Bootstrap5. (n.d.). Bootstrap5 [Image]. Retrieved from <https://www.tutorialrepublic.com/lib/images/bootstrap-5.0-illustration.png>
- [5] Django. (2018, August 2). Django 2.1 landing page [Image]. Retrieved from https://zh.wikipedia.org/zh-tw/File:Django_2.1_landing_page.png
- [6] SQLite. (2014, June 2). File:SQLite370.svg [Image]. Retrieved from <https://zh.wikipedia.org/zh-tw/File:SQLite370.svg>
- [7] Yolo Training Process. (n.d.). Yolo training process [Image]. Retrieved from https://picx.zhimg.com/v2-71fcd7457594445878580f4f59975e8d_1440w.jpg?source=172ae18b
- [8] "YoloV4 環境建置超詳細教學" (柯柯). (2020). <https://youtu.be/PVf16glhnek>.
- [9] "10 分鐘開始把玩 YOLO v4 ~Hands on YOLO v4 in 10 minutes" (JohnnyAI). (2020). <https://youtu.be/C9PKhIIZTYk>.
- [10] "2021 年 Django Web 基础课程 90 分钟完整版" (程序员小飞). (2021). <https://youtu.be/onDgK-bTvjm>.
- [11] "Django Live Stream Webcam | Django Openvc Live Streaming Web Cam | Openvc Django Video Stream | AIOC" (AIOC all in one code). (2021). <https://youtu.be/xz9MvyKGYio>.

4. 致謝

- 在此，要向所有幫助完成這項專題的人表示衷心的感謝。首先，我們要感謝我們的導師，他在研究過程中給予我們的指導和幫助非常寶貴。他不僅提供了有益的建議，還經常抽空檢查我們的進度，並給予我們許多寶貴的意見，使的專題能夠順利完成。此外，還要感謝學長，他們在我們的工作過程中給予我們的幫助和支持非常寶貴。他們不僅分享了他們的專業知識和經驗，還經常與我們一起討論問題，並提供寶貴的建議，使我們的專題得到了很大的提升。最後，還要感謝家人和朋友，他們在我們完成這項專題的過程中給予的鼓勵和支持。他們不僅在我們疲憊的時候給予鼓勵，還在我們遇到困難時提供幫助，使我們能夠堅持到最後。再次感謝所有給予我們幫助的人，沒有你們的幫助，我們不可能完成這項專題。