**ÜSKÜDAR UNIVERSITY FACULTY OF ENGINEERING AND NATURAL SCIENCES**



**Drone control using hand gestures**

**B.Sc. Thesis By:**

**Ömer Kasım**

|  |
| --- |
| **DEPARTMENT OF SOFTWARE ENGINEERING** |
|  |

**Thesis Advisor: Dr. Kristin Surpuhi Benli**

**January, 2024 THE THESIS WAS DEFENDED**

**CONTENTS**

**1. INTRODUCTION………………………………………………………………...2**

**2. LITERATURE REVIEW………………………………………………………..3**

**3. METHODOLOGY……………………………………………………….………4**

**4. RESULTS…………………………………………………………………………5**

**5. WORK PLAN……………………………………………………………..………5**

[**1. Introduction**](#_Toc52873032)

Drones, also known as unmanned aerial vehicles (UAV), are on the rise in a wide range of industrial, recreational and commerical applications, such as security, defense, agriculture, energy, insurance and hydrology [1]. Drones can essentially be considered as multi-functional special flying robots that peform functions such as capturing images and videos, and sensing multimodal data from its environment. This wide range of functions allow them to perform actions that may be expesnive and difficult for humans to do, which can make them especially useful in risky missions. There are two types of drones based on their shape and size, fixed-wing and multirotor. Because of their versatility and small size, multirotor drones can operate where humans cannot, collect multimodal data, and intervene in occasions. At present, flying robots are used in different businesses like parcel delivery systems [2]. New York Police Department uses quadcopters in crime prevention [3]. For the purposes of agriculture monitoring, for instance, the use of multiple sensors such as video and thermal infrared cameras are of benefit [4]. For the sake of clarity in the rest of this work, we define a drone as a multirotor flying robot, excluding fixed-wings.

One of the many useful sensors drones are equibbed with are visual cameras, these sensors can be used in many ways which increase the productivity and veratility of host drone depending of the applications. In a common scenario, the camera output is directed to the drone operator who may command the drone a new instruction based on the current visual environment via a remote controller, which serves as an agent between drone and its operator.

Drones are commonly controlled using designated controllers, either as a dedicated signal transmitter or software applications running on users’ hand-held device (such as mobile phones or tablets). In both cases, the controller sends commands with detailed movement information such as move the drone x units towards a certain direction through wireless channels (e.g., Wi-Fi or Bluetooth). However, this traditional way of communicating can have its drawbacks; For example controlling a drone using designated controllers can be difficult for the unexperinced user and can make them rather focused on operating the remote, which may limit their ability to perform other tasks simultaneously. Moreover, having a designated controller for the drone increases the production cost of the drone because additional hardware and software systems are developed to be able to realize this functionality.

For this reason, many other approaches for controlling drones has been explored. One of them is using hand gestures to control drone. This approach alleviates the existing camera sensor avaliable in the drone to control it using computer vision. These devices use the on-board camera to detect in real time where the user’s hand is and respond to it in intuitive ways. Products include the DJI Spark Drone [5]. This control mechanism offers a more natural, intuitive and hands-free approach to the user and requires no training to use. Additionally, it can cost less as the control logic is part of the drone no additional hardware is required. In this thesis we are going to purpose an application that sends commands to a custom made drone by processing six hand gestures corresponding to the movement commands: up, down, right, left, forward and backward.

[**2. LITERATURE REVIEW**](#_Toc52873034)

Extensive research and work has been made on the topic of Image-based hand gesture recognition problems for decades. Twenty-four basic signs of American Sign Language are detected and classified using a boosted cascade of classifiers trained using AdaBoost and informative Haar wavelet features. In this work, Dinh et al. [6] have proposed a new feature called Double L for best describing the hand gestures other than edge features, line features and edge surrounded features. A study done by Dardas et al. [7] detects and tracks hand gestures in cluttered backgrounds as well as in different lighting conditions. It uses skin detection and hand posture contours comparison algorithms by subtracting faces and only recognizes hand gestures using Principal Component Analysis. In each training stage, different hand gesture images with various scales, angles, and lightings are trained. The training weights are calculated by projecting training images onto the eigen vectors. During testing, the images that contained hand gestures are projected onto the eigen vectors and the testing weights are calculated. Finally, Euclidean distances are calculated between training weights and test weights to classify hand gestures. One of the most similar frameworks to the one we are developing is the framework proposed by Kathiravan [1]. The model is described to involve 3 key functions: image segregation from the video streams of front camera, creating a robust and reliable image recognition based on segregated images, and finally conversion of classified gestures into actionable drone movement, such as takeoff, landing, hovering and so forth. A set of five gestures are studied in this work. Haar feature-based AdaBoost classifier is employed for gesture recognition. Safety of the operator and drone’s action is also investigated by calculating the distance based on computer vision for this task. Classification accuracies show that well-lit, clear background, and within 3 ft gestures are recognized correctly over 90%.

In this application we will develop a system that is compatible with the hardware of our custom drone, recognizes 6 hand gestures: up, down, left, right, forward and backward; and lastly, the model will be able to seperate the hand recognition model from the gesture recognition model. This allows the training of new hand gesture detection model using new data more efficient, reliable and faster as will we elaborate on the following paragraphs.

[**3. METHODOLOGY**](#_Toc52873035)

In this section, we will go in depth of our proposed model of gesture-detection.

The first proposed model for the application was to make a Convolutional neural network (CNN) that directly identifies the hand gestures using an input RGB image from a video stream, where the model was supposed to detect the hand and the gesture at the same time. However, this approach was found to be neither reliable nor efficient due to many reasons, notably, the existance many environment variations (such as lightning, distance from camera, lightning, hand color, etc.) that should be realized for every one hand gesture data. For this reason, we opted for another model that is depicted in Figure 1.

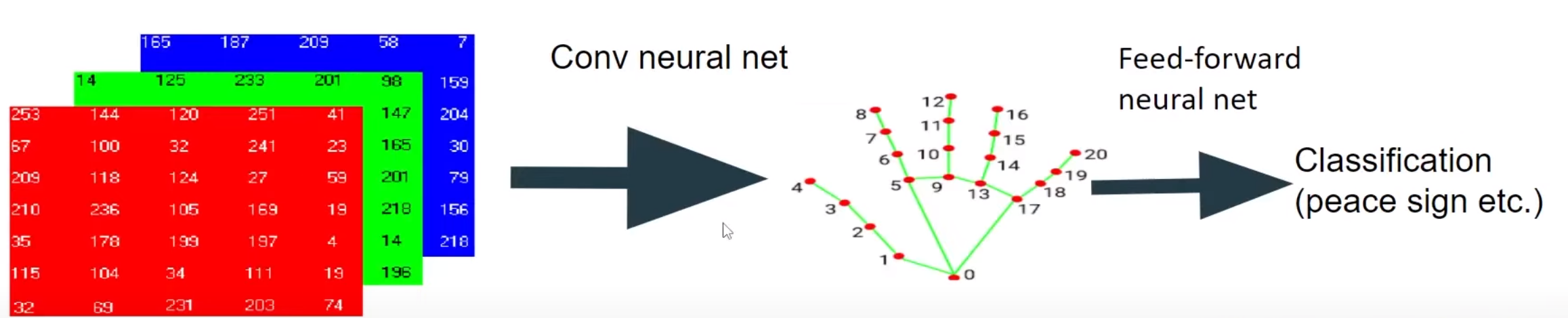


Figure 1. Gesture detection model

This model seperates the process into 2 steps. The first step detects the hand and extract its landmakrs from input RGB image from a video stream through a CNN. And the second step identifies the hand gestures from the extracted landmarks in the first step, through a Feed-Forward classifier neural network.

In this way instead of training the model on pixel images we train it instead on data representing the position of different hand landmarks, which are as explained ealier, extracted through the first model. In this approach we delegate the burden of detecting the hand and extracting its landmarks on the first model, which allows the classifier neural network to become very simple, efficient and easy to train.

However, the first model which detects the hands still poses the challange of being very hard and slow to train. Luckily, the problem of detecting a hand and extracting its landmarks is a well-known problem, which is why there many solutions exists that we can use to solve this problem. In our application we use the medipipe framework and its API to crop the image, detect the hand and output its extracted landmarks. We then use this output as an input to our custom model which classifies hand gestures.

As stated in the medipipe documentation; The hand landmark model bundle detects the keypoint localization of 21 hand-knuckle coordinates within the detected hand regions. The model was trained on approximately 30K real-world images, as well as several rendered synthetic hand models imposed over various backgrounds. The landmarks are depicted in Figure 2.

A screenshot of a computer

Description automatically generated

Figure 2. Hand landmarks

The classifier model can be easily trained using position of the landmarks to detects patterns in similar gestures. In the application we difine 6 gestures and identify each one with an index in a CSV file. Then, to train the model, we use the snapshot the relative pixel of a specific gesture landmarks many times which is outputed by the medipipe model, and store this data in another CSV file.

The gestures CSV file represents the data in rows and each row contains the gesture index along with its gesture data coordinates. These data is then used to train the classifier neural network to detect the hand gestures.

[**4. RESULTS**](#_Toc52873036)

The mediapipe API was swiftly embedded to the application. After capturing the data of 6 hand gestures, namely: Up, down, left, right, forward, backward. We train the model on these data, The classifier model scored an accuracy of 97.52% with my selected test data.

I developed the application to have different modes of operation. One of the operations is the Logging Key Point mode which is activated after the user presses “K” while the application is running. This mode allows the user to easily capture gesture data that is stored in the keypoint.csv file.

Also, another handy feature in the applicatoin, is the the FPS logger, which can be used to monitor the performance of the application.

Lastly I will share the classifier model confustion matrix as a measure of its peformance.

A black and orange squares with numbers

Description automatically generated

**5. Work plan**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **WP** | **WP Title** | **Weeks** | | | | | | | | | | | | | | | |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** |
| 1 | Tranlating the gestures into drone commands |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Programming the drone to work with the defined commands |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Developing computer-drone wirless communication |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | Embedding the app with the drone |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**6. REFRENCES**

[1] Kathiravan Natarajan, Truong-Huy D. Nguyen, and Mutlu Mete. Hand Gesture Controlled Drones: An Open Source Library. CoRR, abs/1803.10344, 2018

[2] V. Gatteschi, F. Lamberti, G. Paravati, A. Sanna, C. Demartini, A. Lisanti, and G. Venezia, “New frontiers of delivery services using drones: A prototype system exploiting a quadcopter for autonomous drug shipments,” in Computer Software and Applications Conference (COMPSAC), 2015 IEEE 39th Annual, vol. 2. IEEE, 2015, pp. 920–927.

[3] T. Moore, “Nypd considering using drones to fight crime - ny daily news,” May 2014. [Online]. Available: http://www.nydailynews.com/ new-york/nyc-crime/nypd-drones-fight-crime-article-1.1799980

[4] D. Turner, A. Lucieer, Z. Malenovsky, D. H. King, and S. A. ` Robinson, “Spatial co-registration of ultra-high resolution visible, multispectral and thermal images acquired with a micro-uav over antarctic moss beds,” Remote Sensing, vol. 6, no. 5, pp. 4003–4024, 2014.

[5] J. Goldman, “DJI Spark review: Ups the ante on selfie drones,” 2017. [Online]. Available: https://www.cnet.com/products/dji-spark/review/

[6] T. B. Dinh, V. B. Dang, D. A. Duong, T. T. Nguyen, and D.-D. Le, “Hand gesture classification using boosted cascade of classifiers,” in Research, Innovation and Vision for the Future, 2006 International Conference on. IEEE, 2006, pp. 139–144

[7] N. H. Dardas and E. M. Petriu, “Hand gesture detection and recognition using principal component analysis,” in Computational Intelligence for Measurement Systems and Applications (CIMSA), 2011 IEEE International Conference on. IEEE, 2011, pp. 1–6.