

**Orientungsprojekt**

**Implementierung einer grafischen Benutzerschnittstelle für einen Quadrokopter**

**Tomas Nguyen**

***19/124/41***

# Einführung

## Kurzfassung

Diese Projektarbeit beschreibt den Prozess der Erstellung einer grafischen Benutzeroberfläche (GUI) für einen Quadrokopter Tello Edu, der aus der Ferne gesteuert werden kann. Die GUI für dieses Projekt wurde mit der Matlab-Software und dem Support-Paket für Ryze Tello Drohne erstellt. Die Drohne verwendet eine Inertiale Messeinheit (IMU), die aus Beschleunigungs- und Drehratensensoren besteht, um die Lageänderung um die eigenen drei Achsen festzustellen. Die Drohne ist außerdem mit einem Ultraschallsensor ausgestattet, der den Abstand zum Boden misst Die gesammelten Daten werden dann auf der entwickelten grafischen Benutzeroberfläche (GUI) angezeigt. In diesem Dokument werden auch die auf der GUI angezeigten gesammelten Daten behandelt. Die grafische Benutzeroberfläche verfügt über weitere Funktionen, wie z. B. die Anzeige der 3D-Echtzeitausrichtung der Drohne und des Kamerastreamings in Echtzeit, auf die noch näher eingegangen werden soll.

Einleitung

In der heutigen Zeit spielt Quadrokopter in vielen Bereichen des menschlichen Lebens eine wichtige Rolle. Er wird in den Bereichen Überwachung, Fotografie, Landwirtschaft und Unterhaltung wie z. B. bei Rennen eingesetzt. Der Quadrokopter hat im Vergleich zu anderen Drohnentypen den Vorteil, dass er schweben kann. Darüber hinaus sind Quadrokopter aufgrund ihrer hervorragenden Manövrierfähigkeit in der Lage, in jeder Art von Umgebung einen Einsatz durchzuführen. Während des Fluges wird automatisch die Lage stabilisieren und verhindert. Die Entwicklung und Gestaltung einer grafischen Benutzeroberfläche (GUI) zur Überwachung der Drohne ist zwingend erforderlich, da ein System, das manuell von einem Menschen überwacht wird, auf große Entfernung und in großer Höhe nicht genau und manchmal nicht zuverlässig ist. Ein GUI-System hilft dem Menschen, die Orientierung und Position zu messen und das System und die Umgebung zu überwachen.

Die Bewegung des Quadrocopters und die Ausrichtung der Drohne

|  |  |  |  |
| --- | --- | --- | --- |
|  | Zeichen |  | Bedeutung |
|  | x  y  z  θ  φ  ψ |  | Vorwärtsgeschwindigkeit (m/s)  Horizontale Geschwindigkeit (m/s)  Vertikale Geschwindigkeit (m/s)  Nickwinkel (Grad)  Kippwinkel (Grad)  Gierwinkel (Grad) |

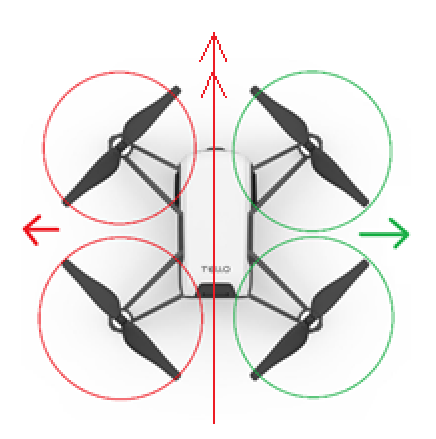
**Tabelle 1 Symbolverzeichnis**

**In der Dokumentation wird die Orientierung der Drohne anhand Euler-****ZYX-Konvention beschrieben**



**Abbildung 1: Ausrichtung der Drohne**

Die Rotationen beziehen sich immer auf die zunächst raumfeste rechtshändige Orthogonalbasis. Die Konvention der Orientierung der Winkel folgt der Mathematik korrekter Hand-Regel, ein positiver Winkel zeigt auch entgegen dem Recht. Alle weiteren Rotationsachsen ergeben sich dynamisch gemäß der festgelegten Reihenfolge der Rotationen. Die Konvention ist mit der Abbildung 1 gegeben.

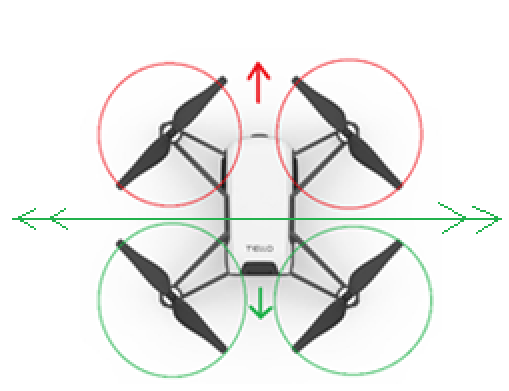
**Rollen - Die Drohne bewegt sich in Bezug auf die x-Achse (rote Farbe)**

(In der Abbildung 1.1) Um nach links zu rollen, wird der Auftrieb an den Motoren auf der rechten Seite erhöht. Die Drohne muss auch den Auftrieb an den Motoren auf der linken Seite verringern. Wenn sich die Drohne nach links bewegt, ändert der Rollwinkel (φ) von 0 bis 180 Grad.

Falls die Drohne nach rechts rollen will, wird die Leistung an den Motoren links erhöht. Die Drohne muss auch den Auftrieb an den Motoren auf der rechten Seite reduzieren und der Rollwinkel variiert (φ) von 0 bis -180 Grad.

**Abbildung 1.1: Drehrichtung des Rollens**

**(Doppelpfeil)**

**Nicken - Die Drohne bewegt sich in Bezug auf die y-Achse (grün)**

Um den Quadrocopter nach vorn zu bewegen, wird durch Neigung Vortrieb erzeugt. In diesem Fall dreht der hintere Rotor (grüner Kreis) schneller als der vordere (rote Kreis). Während der Vorwärtsbewegung kann der (θ) von 0 bis -180 Grad gesehen werden.

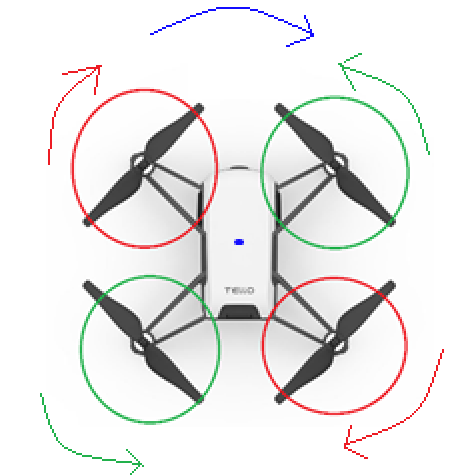
Um sich rückwärts zu bewegen, müssen Sie die Leistung der vorderen Motoren erhöhen und die Leistung der hinteren Motoren verringern. Der Neigungswinkel beim Rückwärtsfahren kann von 0 bis 180 Grad gesehen werden.

**Abbildung 1.2: Drehrichtung des Nickens**

**(Doppelpfeil)**

**Abbildung 1.2: Drehrichtung des Nickens**

**(Doppelpfeil)**

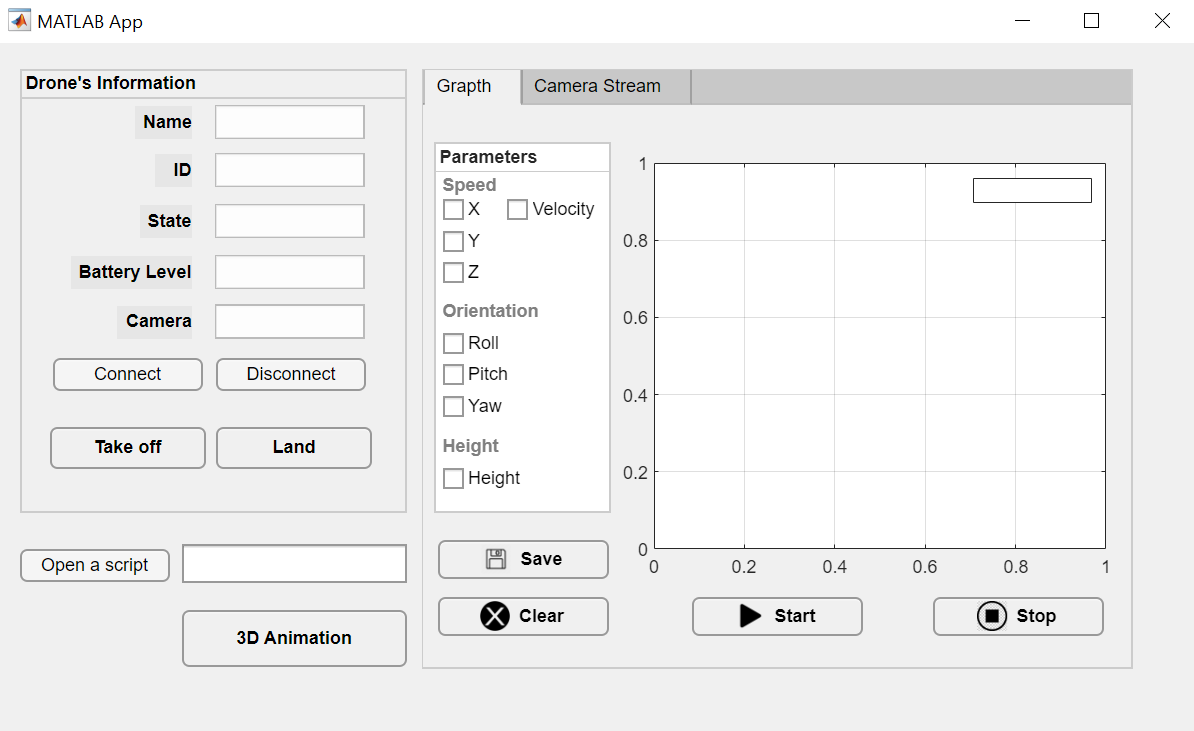


**Gieren - Die Drohne bewegt sich in Bezug auf die z-Achse (blau)**

Um die Drohne im Uhrzeigersinn zu drehen (Gieren). Bei den rechtsdrehenden Motoren (rote Kreis) muss der Auftrieb erhöht werden. Bei gegen den Uhrzeigersinn drehenden Motoren (grüne Kreis) müssen Sie zusätzlich den Auftrieb verringern. Dies geschieht, um die resultierende Kraft nach oben und nach unten auf Null zu halten. Dadurch entsteht auch ein Drehmoment gegen den Uhrzeigersinn. Um den Drehimpuls zu erhalten, dreht sich die Drohne im Uhrzeigersinn.

**Figure 5. Richtung der Gierung**

Gestaltung der grafischen Benutzeroberfläche

****

**Figure 6. Überblick über die grafische Benutzeroberfläche**

Alle Funktionen, die Benutzeroberfläche verwendet, stammen aus dem Support Package für Tello Drone von Matlab.

In den folgenden Abschnitten werden die verschiedenen Teile der grafischen Benutzeroberfläche beschrieben.

* **Das Informationspanel der Drohne (Drone’s Information):** Das Panel zeigt den allgemeinen Zustand des Systems und die wichtigsten Steuerbefehle an. Dieses Panel bietet eine Zusammenfassung von Informationen, die als kritisch angesehen werden und die ständig auf dem Bildschirm angezeigt werden müssen.
* **Messwertefeld**: Um die Werte der Drohne zu messen, werden Kästchen mit verschiedenen Werten angezeigt. Die ausgewählten Daten werden in Echtzeit auf dem Diagramm angezeigt.
* **Kamera-Stream:** Auf der Registerkarte Kamera können Sie die Ansicht der FPV-Kamera der Drohne während des Flugs anzeigen.
* **Schaltfläche Skript** **öffnen**: Um das Flugskript auszuwählen und zu verwenden
* **Schaltfläche für 3D-Animation**: Wenn es gedrückt wird, wird eine 3D-Ansicht der Drohnenorientierung geöffnet
* **Schaltflächen Start, Stop, Clear**: Zur Interaktion mit den Messdaten, um Daten in die Abbildung zu zeichnen, zu stoppen oder zu löschen.
* **Schaltfläche "Speichern":** Zum Speichern von Flugdaten als png-Datei

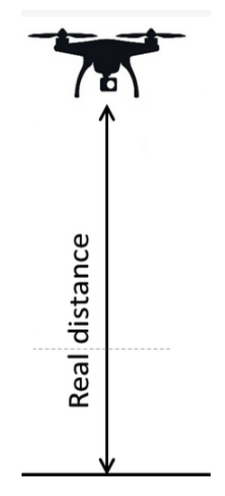
**Orientatierung**

Mit der Funktion *readOrientation(droneObj)* können wir die Körperausrichtung bestimmen. Die Funktion gibt ein Array von Euler-Winkeln zurück, die als [Gieren, Nicken, Rollen] dargestellt werden. Die Daten waren in Radiant und wurden in Grad umgewandelt. [(**In der Abbildung 1)**](#Euler)

**Geschwindigkeit**

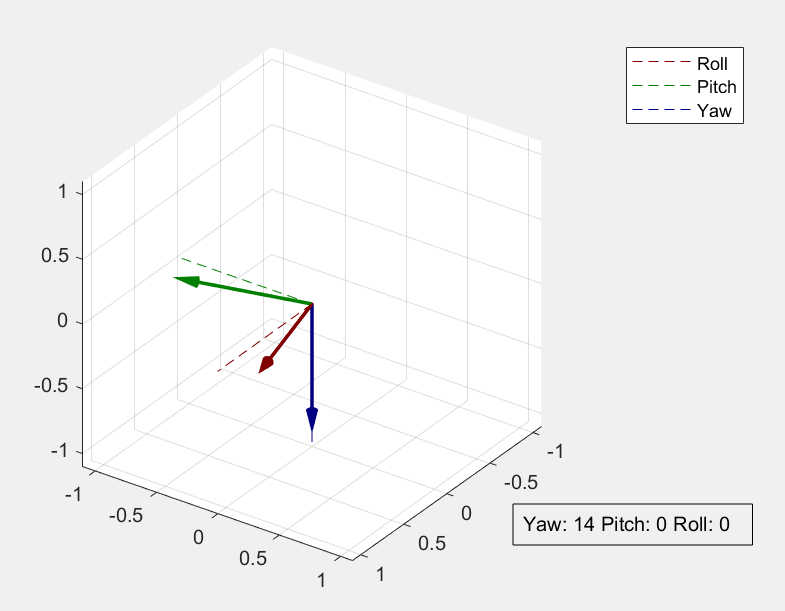
Für die Beschleunigungsmessung wurde die Funktion *readSpeed(droneObj)* verwendet, die die aktuelle Geschwindigkeit der Ryze Drohne entlang der x-, y- und z-Achsen in Bezug [ZYX Konvention](#ZYX).

**Messung der Höhe**

[](https://www.mdpi.com/1424-8220/20/21/6157/htm)

Die Rotoren der Quadcopter-Drohne funktionieren ähnlich wie Tragflächen. Sie erzeugen Auftrieb, indem sie sich schnell drehen, die Luft nach unten ziehen und die Quadrocopter-Drohne in die Luft treiben. Wenn sich der Auftrieb mit der Schwerkraft aufhebt, wird die Nettokraft null, und der Quadcopter schwebt in der Luft. Dann gibt es einen Schub, der den Quadcopter dazu bringt, sich in diese Richtung zu bewegen. Die Höhe der Drohne wird durch Verringern des Auftriebs gesteuert, um ihre Höhe zu ändern.

Die Funktion ReadHeight(droneObj) misst die aktuelle Höhe relativ zur Startfläche in Metern zusammen mit der Systemzeit. Die aktuelle Höhe der Drohne relativ zur Startfläche, angegeben in Metern.



**Figure 7. Überblick über die 3D-Animation**

**Literatur**

1. https://www.mathworks.com/help/supportpkg/ryzeio/referencelist.html?type=function&listtype=cat&category=index&blocktype=all&capability=&s\_tid=CRUX\_lftnav
2. Fathoni, M.F.; Lee, S.;Kim, Y.; Kim, K.-I.; Kim, K.H.Development of Multi-QuadrotorSimulator Based on Real-TimeHypervisor Systems. Drones 2021, 5,59. https://doi.org/ 10.3390/drones5030059
3. Natural user interfaces for human-drone multi-modal interaction, Journal: 2016 International Conference on Unmanned Aircraft Systems (ICUAS), : 2016, ISBN: 978146739334811