

Contrôler un drone Ryze Tello avec Spyder et Python

1) Installer Python

Sous **Windows**

1. Va sur le site officiel : <https://www.python.org/downloads/>
2. Télécharge **Python 3.12.x** (ou 3.11 si tu veux être plus conservateur).
3. Pendant l'installation :
 - **Coche "Add Python to PATH"** (important).
 - Choisis **Install for all users** si possible.

Une fois installé, ouvre **CMD** et tape :

```
python --version
```

Tu dois voir quelque chose comme :

```
Python 3.12.6
```

Sous **Linux (Ubuntu/Debian par ex.)**

La plupart des distros ont déjà Python installé. Vérifie avec :

```
python3 --version
```

Si besoin, installe Python 3.12 avec :

```
sudo apt update
sudo apt install python3.12 python3.12-venv python3.12-dev
```

2) Créer un environnement virtuel (recommandé)

C'est mieux d'isoler ton projet. Dans CMD ou Terminal, fais :

```
python -m venv tello-env
```

Puis active-le :

- **Windows :**

```
tello-env\Scripts\activate
```

- **Linux :**

```
source tello-env/bin/activate
```

Tu verras `(tello-env)` au début de ta ligne → c'est bon.

3) Installer les paquets nécessaires

Dans ton environnement actif, exécute :

```
python -m pip install -U pip setuptools wheel
pip install opencv-python spyder-kernels
pip install "spyder-kernels===2.5.*"
pip install djitellopy
```

☐ Compatible avec **Python 3.10 → 3.12**.

⚠ Évite pour l'instant **Python 3.13+** (encore trop récent, certaines libs n'ont pas de wheels stables).

4) Installer et configurer Spyder

Installation

- **Windows** : installe Spyder depuis <https://www.spyder-ide.org/> ou via `pip install spyder`.
- **Linux** : souvent disponible via `pip install spyder` ou ton gestionnaire (`sudo apt install spyder`).

Configuration

1. Lance Spyder.
 2. Va dans **Outils > Préférences > Python interpreter**.
 3. Sélectionne ton interpréteur Python de l'environnement `tello-env`.
 4. Redémarre Spyder.
-

5) Connexion au drone Tello

1. Allume le Tello.
 2. Sur ton PC, connecte-toi au Wi-Fi du drone (`TELLO-xxxx`).
⚠ Tu perds l'accès Internet le temps du vol, c'est normal.
-

6) Script de test minimal

Crée un fichier `tello_test.py` et mets-y :

```
from djitellopy import Tello
import time

def main():
    tello = Tello()
    try:
        tello.connect()
```

```

print("Batterie:", tello.get_battery(), "%")

tello.takeoff()
time.sleep(2)

tello.move_up(200)      # monte de 200 cm = 2 mètres
time.sleep(2)

tello.move_forward(100) # avance de 1 mètre
time.sleep(2)

tello.rotate_clockwise(90) # rotation de 90°
time.sleep(2)

tello.land()
print("Atterrissage terminé.")
except Exception as e:
    print("Erreur :", e)
    try:
        tello.land()
    except:
        pass
finally:
    tello.end()

if __name__ == "__main__":
    main()

```

7) Commandes utiles du Tello

. [] Base

- `tello.connect()` → se connecter au drone.
- `tello.end()` → fermer la connexion proprement.
- `tello.takeoff()` → décollage.
- `tello.land()` → atterrissage.

- `tello.emergency()` → arrêt immédiat des moteurs (urgence).

☐☐ Déplacements (20–500 cm)

- `tello.move_up(x)` / `tello.move_down(x)`
- `tello.move_forward(x)` / `tello.move_back(x)`
- `tello.move_left(x)` / `tello.move_right(x)`

☐☐ Rotations

- `tello.rotate_clockwise(x)` → rotation horaire en degrés.
- `tello.rotate_counter_clockwise(x)` → rotation antihoraire.

☐☐ Vitesse

- `tello.set_speed(x)` → règle la vitesse (10–100 cm/s).
- `tello.get_speed()` → retourne la vitesse actuelle.

☐☐ Flips

- `tello.flip_forward()`
- `tello.flip_back()`
- `tello.flip_left()`
- `tello.flip_right()`

☐☐ Info / télémétrie

- `tello.get_battery()` → % batterie.
- `tello.get_height()` → hauteur (cm).
- `tello.get_temperature()` → température interne.
- `tello.get_flight_time()` → temps de vol depuis décollage (s).
- `tello.get_acceleration_x()` / `_y()` / `_z()` → accéléromètre.
- `tello.get_barometer()` → altitude barométrique.

☐☐ Vidéo & photo

- `tello.streamon()` → active le flux vidéo.

- `tello.streamoff()` → coupe le flux vidéo.
- `tello.get_frame_read()` → récupère l'image actuelle (OpenCV compatible).
- `tello.take_picture()` → capture une photo.

☐☐ Positionnement avancé

- `tello.go_xyz_speed(x, y, z, speed)` → va à une position relative (cm) avec une vitesse.
- `tello.curve_xyz_speed(x1, y1, z1, x2, y2, z2, speed)` → trajectoire courbe.

☐☐ Réseau

- `tello.connect_to_wifi(ssid, password)` → connecter le Tello à un Wi-Fi (mode station, pas toujours stable).

8) Bonnes pratiques & sécurité

- Teste d'abord avec des petits déplacements (`move_up(20)` par ex.).
- Vérifie toujours la batterie avant le vol.
- Vol en intérieur uniquement si tu as de la place dégagée ; sinon, extérieur.
- Garde une main près du bouton **atterrissage d'urgence** (dans ton script ou via l'app officielle).

9) Dépannage rapide

- **Module introuvable** → Vérifie que tu es bien dans ton environnement (`tello-env`).
- **Pas de connexion** → Assure-toi d'être connecté au Wi-Fi du Tello.
- **Erreur OpenCV** → Sous Linux, installe `libgl1` :

```
sudo apt install libgl1
```

Revision #6

Created 5 July 2025 19:37:01 by Gaëtan Carron

Updated 16 September 2025 09:46:30 by Quentin Petrazoller