

<u>Université 20 Août 1955 – Skikda</u>		<u>جامعة 20 أوت 1955 - سكيكدة</u>
<u>Faculté des Sciences</u>		<u>كلية العلوم</u>
<u>Departement d'informatique</u>		<u>قسم الإعلام الآلي</u>

Mémoire De fin d'étude en vue de l'obtention du Diplôme de Master en Informatique

Option : SIAA

Dans le cadre de l'arrêté ministériel n°1275 en date du 27 septembre 2022, qui définit les modalités d'élaboration d'un projet de fin d'études pour l'obtention d'un diplôme universitaire lié à la création d'une entreprise innovante (startup)

Sujet :

**Conception et réalisation d'un drone quadrirotor
pour le suivi des incendies**

Réalisé par les étudiants :

- Aissa Kenef
- Badr Eddine Bezaze

Dirigé par :

Dr. Laroum Toukik

Remerciement et Dédicaces

Nous voudrions exprimer à travers quelques lignes notre gratitude à tous ceux qui ont contribué de près ou de loin à l'aboutissement de ce projet de fin d'étude.

Tout d'abord à nos familles pour leurs soutiens psychologiques ainsi que pour les moyens mis à notre disposition pour réussir.

Ensuite à notre encadreur monsieur Laroum Toufik pour ses connaissances, son expérience, ses conseils et son encadrement, et grâce à qui nous avons pu obtenir un travail de qualité en si peu de temps.

Enfin à tous le camp enseignant du département informatique pour leur dévouement et leur travail, qui a contribué notre apprentissage.

ملخص

أثبتت الطائرات بدون طيار (UAV) فعاليتها الكبيرة في مراقبة البيئة والتعامل مع الكوارث. يهدف هذا المشروع إلى تطوير طائرة بدون طيار مزودة بكاميرا قادرة على كشف وجود الحرائق والأشخاص في الوقت الحقيقي. يتم استخدام Raspberry Pi كوحدة معالجة على متن الطائرة، حيث يتم التقاط مشاهد جوية وإرسالها عبر شبكة Wi-Fi إلى حاسوب للتحليل. يهدف النظام إلى تقديم حل سريع وذكي لمساعدة فرق الإنقاذ في الكشف عن الحرائق والتعرف على الأشخاص في المناطق الخطرة أو المعزولة، مما يساهم في تقليل وقت الاستجابة وتقليل المخاطر.

Résumé

Les drones Unmanned Aerial Vehicle (UAV) ont démontré une grande efficacité dans la surveillance de l'environnement et la gestion des catastrophes. Ce projet vise à développer un drone équipé d'une caméra capable de détecter en temps réel la présence d'incendies et de personnes. Un Raspberry Pi est utilisé comme unité de traitement embarquée, permettant de capturer des images aériennes et de les transmettre via une connexion Wi-Fi à un ordinateur pour analyse. Le système a pour objectif d'offrir une solution rapide et intelligente afin d'aider les équipes de secours à détecter les incendies et identifier les personnes dans les zones dangereuses ou isolées, ce qui permet de réduire le temps de réponse et de limiter les risques.

Abstract

Drones Unmanned Aerial Vehicle (UAV) have proven to be highly effective in environmental monitoring and disaster response. This project aims to develop a drone equipped with a camera capable of detecting the presence of fires and humans in real time. A Raspberry Pi is used as the onboard processing unit to capture aerial footage and transmit it via WiFi to a computer for analysis. The system is designed to provide a fast and intelligent solution to assist rescue teams in detecting fires and identifying people in hazardous or remote areas, thereby reducing response time and minimizing risks.

Table des matières

Remerciement et Dédicaces	2
Résumé	3
Table des matières	4
Table des figures	8
Introduction générale.....	10
Chapitre1 : System Embarqué Et Les Cartes Electroniques	12
1. Introduction	13
2. Les systèmes embarqués	13
3. Cartes Électroniques.....	14
3.1. Définition	14
3.2. Les types de carte électronique.....	15
3.3. Rôles de la carte électronique.....	15
4. La carte électronique Arduino, Raspberry Pi	16
4.1. Arduino.....	16
4.2. Raspberry Pi.....	18
5. Conclusion.....	20
Chapitre 02 : Drone quadri-rotor.....	21
1. Introduction	22
2. Vue ensemble sur les drones	22
2.1. Définition	22
2.2. Historique : Une invention militaire.....	22
2.3. Utilisation des drones	24
2.4. Utilisation de drone dans ce projet.....	28
3. Principe de fonctionnement et Modélisation d'un drone quadri-rotor	29
3.1. Fonctionnement de drone quadri-rotor.....	29
3.2. Les avantages quadri-rotor	29
3.3. Mouvements du quadri-rotor.....	30
4. Classification des drones	33
4.1. Selon le nombre d'hélices	33
4.2. Selon la taille.....	37
5. Avantages des drones à quatre rotors dans les incendies de forêt.....	38

6. Conclusion.....	38
Chapitre 03 : Conception du Système	40
1. Introduction	41
2. Conception général.....	41
3. Conception de drone.....	43
4. Transmission vidéo	44
5 Présentation de l’UML	45
5.1. Diagramme de sequence.....	45
5.2. Diagramme d’activite	46
6. Software (Setup and Test)	47
6.1. La procédure de configuration	47
6.2. L’étalonnage du régulateur de vitesse électronique (ESC).....	54
6.3. Procédure d’équilibrage des moteurs et hélices	55
6.4. Le code du contrôleur de vol.....	56
6.5. PID Controller (Stabilisation)	57
7. Conclusion.....	59
Chapitre 04 : Réalisation du Système	60
1. Introduction	61
2. Langages et outils de développement.....	61
2.1. Python.....	61
2.2. OpenCV.....	61
2.3. PyCharm.....	62
2.4. Arduino IDE.....	62
2.5. Yolov8.....	62
3. Les Mouvements d’une manette	63
3.1. Schéma d’une manette de drone.....	63
3.2. Move Up/Down (Throttle Control).....	64
3.3. Move Left/Right (Roll Control)	65
3.4. Move Forward/Backward (Pitch Control).....	66
3.5. Rotate Left/Right (Yaw Control)	67
4. Algorithme de l’application	69
4.1. GUI.....	69
4.2. Fire and Human Detection	70
4.3. Send E-mail	71
4.4. Screenshot	72

5. Description des composants (hardware)	73
6. Présentation de l'application	76
6.1. L'interface	76
6.2. Screenshot	76
6.3. Exemple de détection	77
6.4. Envoyer email	78
7. Conclusion.....	78
Chapitre5 : BMC (Business Model Canvas).....	79
1. Présentation du projet.....	80
1.1. Idée du projet (la solution proposée) :.....	80
1.2. Valeurs proposées :	80
1.3. Équipe de travail :	81
1.4. Objectifs du projet :.....	82
1.5. Calendrier de réalisation du projet	83
2. Les aspects innovants	83
2.1. Nature des innovations	83
2.2. Domaines des innovations.....	84
3. Analyse stratégique du marché	85
3.1. Présentation du secteur de marché	85
3.2. Analyse de la concurrence actuelle :	86
3.3. Stratégie marketing proposée	87
4. Plan de production et d'organisation.....	88
4.1. Processus de production	88
4.2. Approvisionnement des composants	89
4.3. Ressources humaines nécessaires.....	90
4.4. Ressources humaines nécessaires.....	91
5. Plan Financier.....	92
5.1. Coûts Estimés et Charges Financières	92
5.2. Chiffre d'affaires	93
Conclusion générale	94
Bibliographie.....	96

Table des figures

Figure 1 : Fonctionnement de système embarquer.....	14
Figure 2 : Les cartes électroniques.....	15
Figure 3 : Arduino UNO.....	18
Figure 4 :.Raspberry Pi 3.....	20
Figure 5 : Les drones dans l’histoire.....	23
Figure 6 : Drone agricole.....	25
Figure 7: Cartographie avec un drone.....	26
Figure 8: Drone logistique.....	28
Figure 9 : Sens de rotation des rotors du quadri-rotor.....	29
Figure 10 : Mouvement vertical.....	31
Figure 11: Mouvement de roulis.....	32
Figure 12 : Mouvement de tangage.....	32
Figure 13 : Mouvement de lacet.....	33
Figure 14 : Mouvement de translation.....	33
Figure 15 : Les différents nombres d’hélices.....	33
Figure 16 : Le drone mono-rotor.....	34
Figure 17 : Le tricoptère.....	34
Figure 18 : Le Quad-copter.....	35
Figure 19 : Hexacoptère.....	35
Figure 20 : L’Octocoptère.....	36
Figure 21: Les drones aux ailes fixes.....	36
Figure 22 : Le Catégories des drones.....	38
Figure 23 : Photo du début d'un incendie dans la forêt.....	41
Figure 24 : Schéma du scénario expérimental.....	42
Figure 25 : Le schéma fonctionnel du quad-copter.....	43
Figure 26 : Raspberry Pi 3 et Camera.....	44
Figure 27 : Séquence de fonctionnement du drone (Détection de Feu/Humain).....	45
Figure 28: Diagramme d'activité.....	46
Figure 29: YMFC-AL_setup fichier.....	47
Figure 30: Sélectionnez outils.....	47
Figure 31: Démarrer la connexion série.....	47
Figure 32: Full Throttle Position.....	48
Figure 33: Left Wing Up position.....	49
Figure 34: Nose Up Position.....	49
Figure 35: Nose Right Position.....	50
Figure 36: Lift Left Side 45°.....	52
Figure 37: Lift Nose Up 45°.....	52
Figure 38: Rotate Nose to Right 45°.....	53
Figure 39: Le sens de rotation du moteur.....	55
Figure 40: Les valeurs de calibration.....	56
Figure 41: Manette de drone.....	63

Figure 42: Algorithme Move Up/Down.....	64
Figure 43: Algorithme Move Left/Right.....	65
Figure 44: Algorithme Move Forward/Backward.....	66
Figure 45: Yaw Controle.....	67
Figure 46: Algorithme GUI.....	69
Figure 47: Algorithme Fire and Human Detection	70
Figure 48 : Algorithme Send E-mail	71
Figure 49 : Algorithme Screenshot	72
Figure 50 : Matériels de drone.	73
Figure 51: Drone Quadri-rotor.	75
Figure 52: L'interface GUI	76
Figure 53: Take Screenshot.....	77
Figure 54: Fire and human detection.....	77
Figure 55: Send Email.....	78

Introduction générale

Les avancées technologiques récentes dans le domaine des systèmes embarqués et des drones ont permis de développer des solutions innovantes pour la surveillance et la gestion des catastrophes naturelles. Parmi ces solutions, l'utilisation de drones quadri-rotors se révèle particulièrement prometteuse pour la détection précoce des incendies et le repérage des personnes en danger. Grâce à leur capacité à couvrir de vastes zones rapidement et à transmettre en temps réel des images et des données, les drones représentent une alternative efficace aux méthodes de surveillance conventionnelles.

Ce mémoire s'inscrit dans cette dynamique en proposant la conception et la réalisation d'un drone quadri-rotor équipé d'un système embarqué avancé pour la détection automatique des incendies et des êtres humains. Le projet repose sur l'intégration de technologies telles que la vision par ordinateur via OpenCV, le traitement d'images en temps réel et la transmission des données via des cartes électroniques comme le Raspberry Pi et l'Arduino.

L'objectif principal est de concevoir un drone autonome capable de détecter rapidement un départ de feu et de signaler la présence humaine, facilitant ainsi l'intervention des services de secours. Pour ce faire, nous étudierons les principes de fonctionnement des drones, les choix technologiques adoptés, ainsi que les défis rencontrés lors de l'implémentation du système.

Structure du mémoire

Le chapitre 1 : présente les systèmes embarqués et les cartes électroniques, en expliquant leur rôle et en détaillant les plateformes Arduino et Raspberry Pi utilisées dans le projet.

Le chapitre 2 : traite du drone quadri-rotor, en abordant son fonctionnement, sa classification et ses avantages pour la surveillance des incendies.

Le chapitre 3 : porte sur la conception du système, décrivant l'architecture du drone, ses composants matériels et la transmission vidéo en temps réel.

Le chapitre 4 : porte sur la réalisation du système, en expliquant les outils de développement, les algorithmes de contrôle et les tests effectués.

Le chapitre 5 : sera consacré à l'étude du Business Model Canvas (BMC) de la startup qui offre une vision structurée du fonctionnement économique de l'entreprise (la clientèle ciblée, les sources de revenus, les partenariats stratégiques, les dépenses et la structure des coûts.... etc.

**Chapitre1 :
System Embarqué
Et Les Cartes Electroniques**

1. Introduction

Le développement des drones repose sur l'intégration de * systèmes embarqués performants * et de * cartes électroniques spécialisées *. Ces systèmes, conçus pour fonctionner avec des ressources limitées, permettent de gérer en temps réel des tâches critiques telles que la stabilisation du vol, la navigation et l'interaction avec les environnements. Grâce aux cartes mères électroniques telles qu'Arduino et **Raspberry Pi**. Dans ce chapitre, nous aborderons les systèmes embarqués et les cartes électroniques utilisées pour contrôler les drones.

2. Les systèmes embarqués

Les systèmes embarqués jouent un rôle central dans notre monde technologique et électronique moderne. Ils se retrouvent dans une variété d'appareils, tels que les Smartphones, les voitures et les appareils électroménagers.

Combinant harmonieusement matériel (hardware) et logiciel (software), ces systèmes sont conçus pour fonctionner de manière autonome et répondre à des besoins spécifiques. Leur optimisation pour des performances dédiées les rend indispensables dans de nombreuses applications pratiques.

En constante évolution, la technologie des systèmes embarqués représente un domaine à la fois fascinant et complexe. La conception de ces systèmes nécessite une sélection minutieuse des composants matériels et le développement de logiciels adaptés aux contraintes spécifiques du produit.

Le contrôle des données y occupe une place centrale, car ces systèmes doivent collecter, traiter et transmettre des informations en temps réel. Les ingénieurs s'efforcent donc de concevoir des mécanismes efficaces de gestion de la mémoire, afin d'assurer une performance optimale, même avec des ressources limitées. [1]

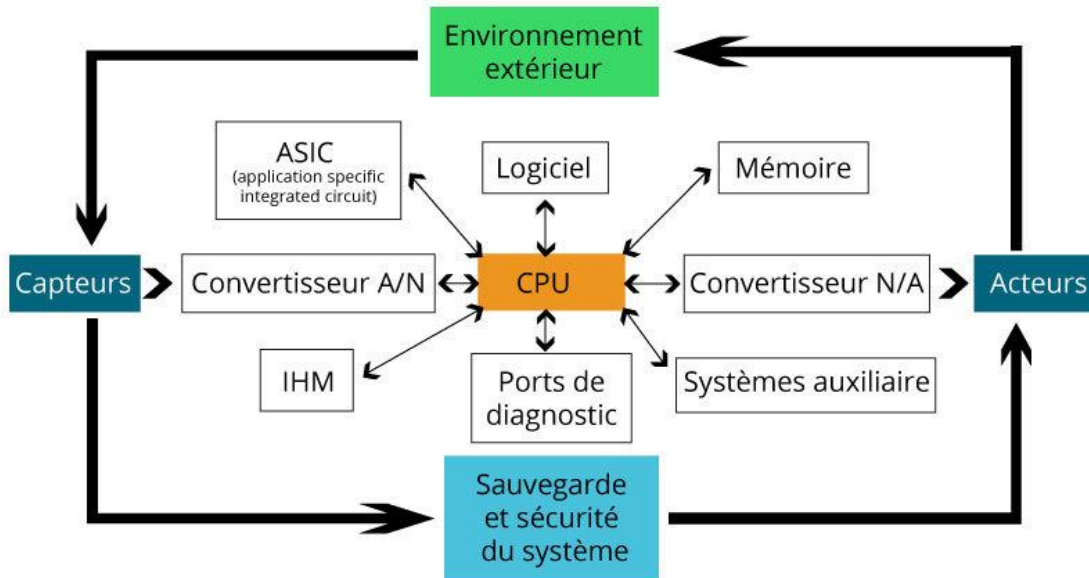


Figure 1 : Fonctionnement de système embarqué

3. Cartes Électroniques

3.1. Définition

La carte électronique est un dispositif qui, grâce au design électronique, participe à la création d'un circuit électrique complexe. Composée de différentes pistes, elle représente une liaison électrique pour un ensemble de composants électroniques.

Pour dire vrai, la carte électronique a été conçue pour remplacer le câblage conventionnel. De ce fait, elle est présente dans de nombreux appareils électroniques. On peut citer notamment les postes téléviseurs, les postes radio et les ordinateurs.

L'importance de la carte électronique réside dans le fait qu'elle a donné un nouvel aspect aux équipements électroniques. En effet, elle possède une taille et un poids réduits. De plus, sa fiabilité ainsi que son uniformité sont grandement optimisées.

[2]

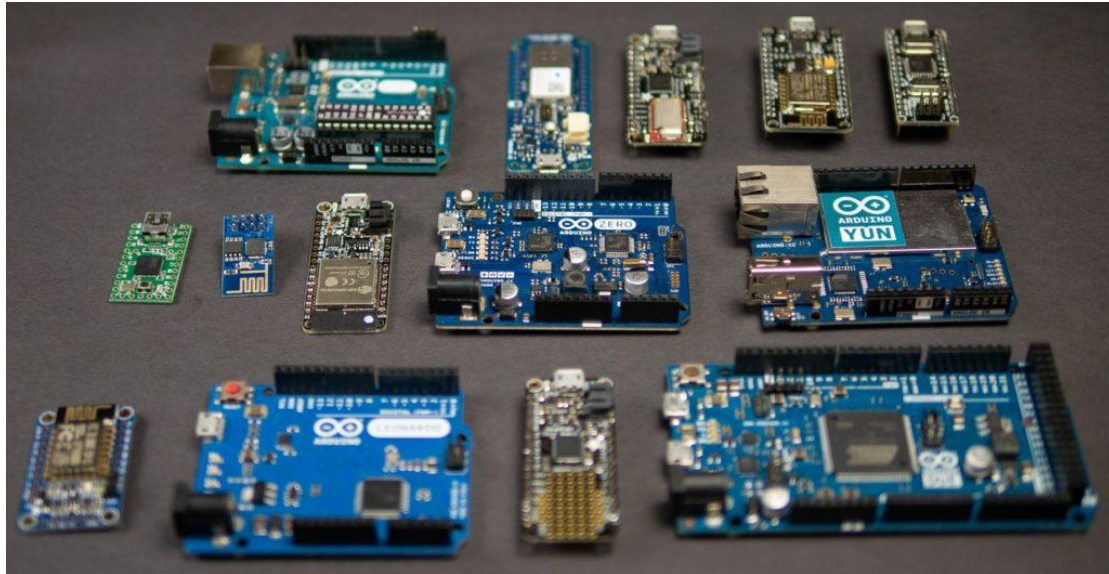


Figure 2 : Les cartes électroniques

3.2. Les types de carte électronique

Il existe de nombreux types de cartes électroniques en fonction de leur utilisation et de leur conception.

- **Cartes de Circuit Imprimé (PCB):** Cartes rigides: Ce sont les types les plus courants, généralement fabriqués en Flame Retardant 4 (FR4). Ces cartes sont utilisées dans presque tous les appareils électroniques. Les pistes conductrices sont gravées ou imprimées sur une planche isolante, formant ainsi le circuit.
- **Cartes Flexibles:** Ces cartes sont conçues pour être pliées ou courbées, ce qui les rend idéales pour les appareils qui nécessitent une certaine flexibilité, comme les caméras ou les appareils auditifs.
- **Cartes Rigide-Flex:** Combinaison des deux précédentes, ces cartes possèdent des zones rigides et des zones flexibles, permettant une plus grande polyvalence.[3]

3.3. Rôles de la carte électronique

La carte électronique joue un rôle essentiel dans le bon fonctionnement des appareils électroniques, en assurant leur performance optimale pour accomplir les tâches essentielles.

Certaines cartes électroniques sont spécialement conçues pour remplir une seule fonction, comme le transfert de données. D'autres, en revanche, sont polyvalentes et

capables de gérer plusieurs fonctions, telles que le stockage et l'affichage des données.

En outre, la carte électronique garantit une isolation électrique entre les différents composants. Cela prévient les courts-circuits et permet aux signaux électriques d'atteindre uniquement les composants ciblés. Elle protège également les éléments électroniques des influences extérieures, telles que l'humidité, la poussière et les chocs physiques, contribuant ainsi à leur durabilité et à leur fiabilité. [4]

4. La carte électronique Arduino, Raspberry Pi

Dans notre projet, nous allons utiliser des cartes électroniques **Arduino** et **Raspberry Pi**. Dans ce chapitre, nous aborderons chaque carte séparément.

4.1. Arduino

4.1.1. Définition

Arduino est une plateforme matérielle et logicielle open-source, conçue pour les programmeurs informatiques, les artistes industriels, les professionnels et ceux intéressés par le développement de dispositifs et d'applications interactifs dans un environnement de développement interactif. Arduino peut recevoir des signaux d'entrée de divers capteurs et entrées. En contrôlant des sources lumineuses, des moteurs ou d'autres actionneurs, Arduino peut modifier l'environnement qui l'entoure. Les programmes pour le microcontrôleur sur la carte Arduino sont écrits dans le langage de programmation Arduino (basé sur 'Wiring' - un cadre open-source pour les microcontrôleurs) et s'exécutent dans l'environnement de développement Arduino (basé sur 'Processing' - un langage de programmation open-source et un environnement de développement intégré).

Arduino peut fonctionner de manière autonome ou communiquer avec des logiciels exécutés sur un ordinateur (par exemple, Flash, Processing et MaxMSP). L'IDE Arduino, open-source et téléchargeable gratuitement, facilite l'écriture du code, son transfert vers la carte, ainsi que la création de dispositifs interactifs [5].

4.1.2. Historique

C'est en 2005 que la toute première carte **Arduino** est née dans les salles de classe de l'Institut de Design Interactif à Ivrea, en Italie. Eh bien, si vous n'êtes pas très familier avec le terme, un **Arduino** est une carte de développement basée sur un microcontrôleur open-source, qui a ouvert les portes de l'électronique à de nombreux designers et ingénieurs créatifs. C'est à l'Institut de Design Interactif qu'une thèse matérielle a été présentée pour un design de câblage par un étudiant colombien nommé Hernando Barragán. Le titre de la thèse était 'Arduino – La rivoluzionedell'open hardware' ('Arduino – La Révolution du matériel open-source'). Oui, cela semblait un peu différent des thèses habituelles, mais personne n'aurait imaginé que cela créerait une niche dans le domaine de l'électronique. Une équipe de cinq développeurs a travaillé sur cette thèse et, une fois la nouvelle plateforme de câblage terminée, ils ont travaillé pour la rendre beaucoup plus légère, moins chère et accessible à la communauté open-source. [6]

4.1.3. Différents Types De Cartes Arduino

Il existe plusieurs types de cartes **Arduino**, à savoir :

Arduino Uno (R3)	ArduinoDiecimilaRedBoard
Arduino Board	Arduino Mega (R3) Board
Arduino Leonardo Board	Arduino Robot
ArduinoEsplora	Arduino Uno
ArduinoDueLilyPad	Arduino Micro
Arduino Board	Arduino Bluetooth
Arduino Ethernet	Arduino Pro Micro
Arduino Zero Fastest	

Dans ce projet, nous avons utilisé le type d'Arduino Uno.

- **Arduino Uno:** Arduino Uno est une carte microcontrôleur basée sur l'ATmega328 (fiche technique). Il dispose de 14 broches d'entrée/sortie numériques (dont 6 peuvent être utilisées comme sorties PWM), 6 entrées analogiques, un oscillateur à quartz de 16 MHz, une connexion USB, une prise d'alimentation, un en-tête ICSP et un bouton de réinitialisation. Il contient tout le nécessaire pour prendre en charge le microcontrôleur; connectez-le simplement à un ordinateur avec un câble USB ou alimentez-le avec un adaptateur CA / CC ou une batterie pour commencer. L'Uno diffère de toutes les cartes précédentes en ce qu'elle n'utilise pas la puce de pilote FTDI USB verseriez. Au lieu de cela, il dispose de l'Atmega8U2 programmé comme un convertisseur USB-série. [7]

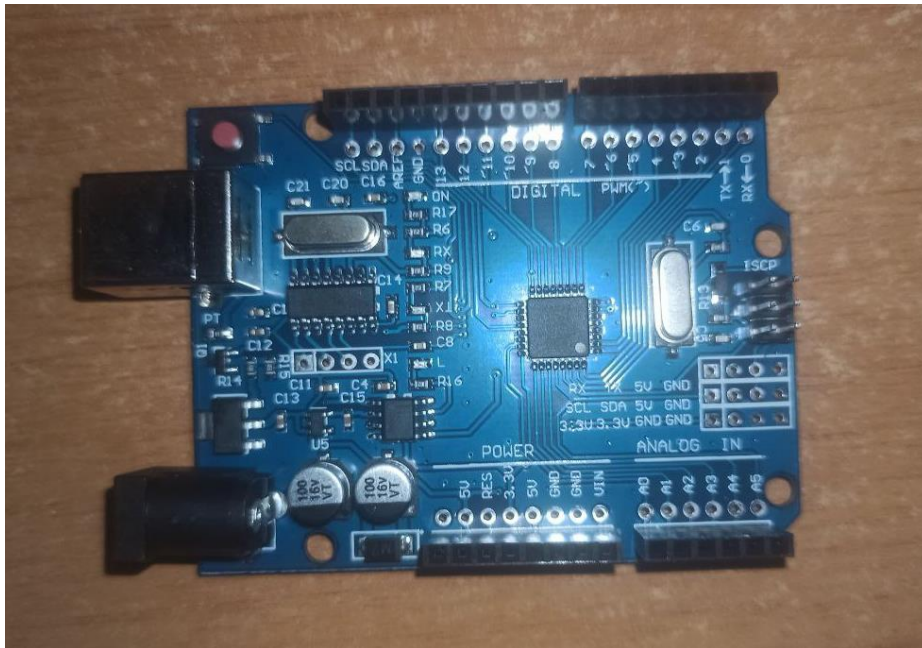


Figure 3 : Arduino UNO

4.2. Raspberry Pi

4.2.1. Définition

Raspberry Pi est défini comme un mini-ordinateur de la taille d'une carte de crédit qui est interopérable avec n'importe quel périphérique d'entrée et de sortie tel qu'un moniteur, un téléviseur, une souris ou un clavier-convertissant efficacement la configuration en un ordinateur à part entière à faible coût. [8]

4.2.2. Historique

La **Raspberry Pi** est née en Angleterre, à l'université de Cambridge. En 2006, des membres de cette université, constatant le déclin continu des étudiants dans la filière informatique après l'engouement des années 90, ainsi qu'une baisse des compétences, en ont cherché les origines. Ils ont constaté que les générations précédentes d'étudiants avaient été initiées à l'informatique dès leur enfance grâce aux micro-ordinateurs des années 80 dont les noms font encore briller les yeux des personnes ayant connu cette époque (Commodore, Amstrad, Sinclair, Texas Instruments...). Sur ces machines, pas d'environnement graphique permettant de tout faire à la souris. Il fallait nécessairement passer par du code et plonger dans les entrailles de la machine pour en extraire la substantifique moelle.

Ils ont donc cherché à concevoir une nouvelle plate-forme informatique à laquelle chacun pourrait avoir facilement accès, tant au niveau matériel que financier, et qui inciterait à la programmation. Plusieurs prototypes ont été conçus au sein de la fondation **Raspberry Pi** qu'ils ont créée pour se financer. Ce ne fut pas sans difficulté pour respecter le cahier des charges qu'ils s'étaient fixé, notamment le coût. Les 1ers exemplaires voient le jour en 2011 et la véritable commercialisation commence début 2012. Plein d'espoir, ses concepteurs s'imaginaient à l'origine pouvoir vendre 1000, peut-être même 10 000 exemplaires de leur machine. Fin 2016 la **Raspberry Pi** est vendue à plus de 10 millions d'exemplaires à travers le monde. [9]

4.2.3. Les meilleurs types de Raspberry Pi

Pour quiconque souhaite acquérir l'un des ordinateurs monocarte, il peut être difficile de faire le tri parmi les innombrables modèles et générations qui ont été commercialisés. Cependant, les modèles les plus remarquables de Raspberry Pi disponibles sur le marché sont:

Raspberry Pi Zero	Raspberry Pi 1
Raspberry Pi 2 B	Raspberry Pi 3
Raspberry Pi 4B	Raspberry Pi 400

Dans ce projet, nous avons utilisé le type **Raspberry Pi 3**.

Raspberry Pi 3 :

Le Raspberry Pi 3 est un ordinateur monocarte doté d'un processeur Broadcom BCM2837 64 bits, offrant des performances plus rapides que les modèles précédents. Il comprend une connexion Wi-Fi et Bluetooth intégrées pour la connectivité sans fil, tout en conservant la même disposition que les modèles précédents, comme un entête GPIO à 40 broches et quatre ports USB. [10]



Figure 4 :.Raspberry Pi 3

5. Conclusion

Les systèmes embarqués et les cartes électroniques constituent des piliers fondamentaux dans la conception et le contrôle des drones modernes. En intégrant des plateformes comme **Arduino** et **Raspberry Pi**, ces technologies permettent de répondre aux besoins de calcul en temps réel, de traitement des données, et d'interaction avec l'environnement. Elles offrent une flexibilité et une capacité d'adaptation qui encouragent des innovations continues dans le domaine des drones, que ce soit pour la stabilisation, la navigation ou l'amélioration des performances globales.

Chapitre 02 : Drone quadri-rotor

1. Introduction

L'un des paramètres complexes de la conception et du contrôle d'un UAV (Unnamed Aerial Vehicule) est de pouvoir déterminer et comprendre profondément ses différents mouvements, Cela est indispensable non seulement pour la conception du contrôleur de vol de l'UAV, mais également pour s'assurer qu'il existe une compatibilité entre les simulations de son comportement et la réalité lorsque la commande est appliquée pour son contrôle. Pour rappel, le quadri-rotor est classé dans la catégorie des systèmes volants les plus complexes vu le nombre d'effets physiques qui affectent sa dynamique à savoir :

- les effets aérodynamiques.
- la gravité.
- les effets gyroscopiques.
- les frottements et le moment d'inertie.

Cette complexité provient principalement du fait que chaque mode de vol se traduit par une expression particulière de chacun de ces effets. Par conséquent, les modèles dynamiques du quadrirotor que nous allons concevoir varieront en fonction du mode de vol.

2. Vue ensemble sur les drones

2.1. Définition

Un drone, également appelé véhicule aérien sans pilote (UAV, pour Unmanned Aerial Vehicle) ou système aérien sans pilote (UAS, pour Unmanned Aircraft System), est un appareil volant pouvant être contrôlé à distance ou fonctionner de manière autonome. Équipés de capteurs, de caméras, de systèmes de communication et parfois d'autres équipements spécialisés, les drones sont conçus pour répondre à des besoins spécifiques selon leur domaine d'application.[11]

2.2. Historique : Une invention militaire

Le premier aéronef sans pilote (UAV : Unnamed Aerial Vehicule) a vu le jour sur la base militaire d'Avord le 02 Juillet 1917.

C'est un français, Max Boucher, qui a réussi le décollage d'un avion de type Voisin 150 HP sans pilote. Il survolera une distance de 500 mètres au-dessus du sol.

Chapitre 03 : Conception du Système

L'idée, en pleine Première Guerre Mondiale, était de créer un engin capable d'effectuer des missions de reconnaissance sans engager la vie des pilotes.

En Angleterre, dès 1916 l'ingénieur Archibald Low développe un avion cible pilotable par télégraphie sans fil, l'Aerial Target.

Aux Etats-Unis en 1917, un avion, le Hewitt-Sperry Automatic Airplane est déjà capable de lancer des torpilles aériennes. En 1918, un prototype de drone-torpille a vu le jour : le Kettering Bug, mais, il n'a jamais été opérationnel sur le terrain.

D'autres, avions-cibles ont été construits dans les années 1934-1938 au Royaume-Uni et aux Etats-Unis comme le Radioplane QG-2. Puis en 1941, l'US Navy acquière un nouveau modèle baptisé Target Drone Denny (TDD-1).

C'est surtout pendant la seconde guerre mondiale que l'armée allemande développe à partir de 1938 des vecteurs guidés à distance, et notamment les tristement célèbres V1 et V2 bien qu'ils s'agissent plus de bombes volantes.

Depuis avec le perfectionnement des technologies, les drones aériens sont capables d'effectuer des missions de surveillance et même d'éliminer des cibles stratégiques et le grand essor des drones date des Guerre de Corée et du Viêt Nam, puis pendant la guerre froide. [12]



Figure 5 : Les drones dans l'histoire

2.3. Utilisation des drones

Bien que l'utilisation et le développement de drones soient toujours à des fins militaires, l'utilisation de drones dans les secteurs commerciaux et autres se généralise aujourd'hui.

Les évaluations des domaines d'utilisation encore en cours dans ce contexte, et dont la généralisation est attendue prochainement, sont présentées ci-dessous.

➤ **Agriculture :**

Dans les zones agricoles, il contribue de manière significative à fournir des données aux acteurs du secteur et à augmenter la productivité des producteurs en collectant des données avec une observation régulière des terres. Les drones dans les applications de pulvérisation, de fertilisation et de détection des dommages aux plantes inspectent les cultures plantées. L'utilisation de drones dans la production agricole est de plus en plus courante au quotidien et permet aux producteurs d'optimiser la production en augmentant l'efficacité et en réduisant la charge physique. Les drones offrent un gain de temps considérable dans la recherche agricole, la plantation de semences, la surveillance du bétail et la prévision des rendements des cultures. Les techniques agricoles intelligentes se généraliseront grâce aux données des drones et des satellites permettant aux producteurs de surveiller leurs produits et de planifier les périodes de plantation, de fertilisation et de pulvérisation. [13]

➤ **Environnement :**

En fonction de l'augmentation de la population urbaine, les drones sont utilisés avec succès dans les processus de contrôle de l'environnement et d'intervention d'urgence. Pour prévenir la pollution de l'environnement, les drones réalisent non seulement des projets visant à nettoyer les mers, mais apportent également une contribution importante à la lutte contre le braconnage et au suivi des animaux en voie de disparition. Le comportement et les conditions pathologiques des animaux peuvent être suivis grâce à des caméras thermiques. De plus, les compagnies pétrolières utilisent des drones pour inspecter les fuites de pétrole et de gaz. Les drones équipés de caméras thermiques effectuent des tâches importantes dans la détection des fuites rapidement et en prévenant les risques éventuels. [13]



Figure 6 : Drone agricole.

➤ **Santé :**

L'utilisation de drones à des fins médicales est utilisée pour transporter du matériel d'intervention sur les lieux dans les zones urbaines, pour diriger les personnes vers la scène avec des drones équipés de caméras et d'appareils audio, pour transporter du matériel tel que des médicaments et des trousse de traumatologie dans les zones rurales, ou pour faciliter les efforts de recherche et de sauvetage. Chez les patients présentant un rythme choquant lors d'événements pré hospitaliers, la délivrance d'un défibrillateur salvateur (DEA) par drone est 32% plus rapide dans les centres urbains et 93% plus rapide dans les zones rurales où les autres véhicules ne peuvent pas atteindre. De la même manière, il a été démontré qu'il est possible de transporter des médicaments / tissus et des produits sanguins avec des drones, avec les recherches menées avec des échantillons prélevés avant et après le vol sur l'effet des vibrations sur les tissus transportés. Ces matériaux peuvent être livrés dans des zones difficiles plus rapidement que les méthodes traditionnelles. [13]

➤ **Photographie et cinéma :**

Des tournages vidéo professionnels sont réalisés aujourd'hui à l'aide de drones dans des publicités, des séries télévisées et des plateaux de tournage, capturant avec succès des images spécifiques. Il est utilisé sur le marché direct de produits et en prenant des images aériennes pour montrer une ville, une plage ou un bâtiment à vol d'oiseau dans des tournages publicitaires. Les drones publicitaires deviennent de plus

en plus communs dans les événements bondés et les zones à forte densité. Il offre une grande commodité pour collecter des images et des informations à partir d'endroits qui ne peuvent pas être visités ou entrés, en particulier en raison de problèmes de sécurité. L'utilisation de drones peut apporter une contribution significative à l'innovation et à la qualité dans l'industrie cinématographique. [13]

➤ **Cartographie:**

Les drones, de plus en plus répandus en cartographie, peuvent cartographier presque tous les terrains rapidement et en trois dimensions. À cette fin, les drones LiDAR équipés de capteurs fournissent des données très performantes et précises. La technologie LiDAR offre des solutions importantes dans l'évaluation des produits agricoles ainsi que la cartographie des reliefs. [13]

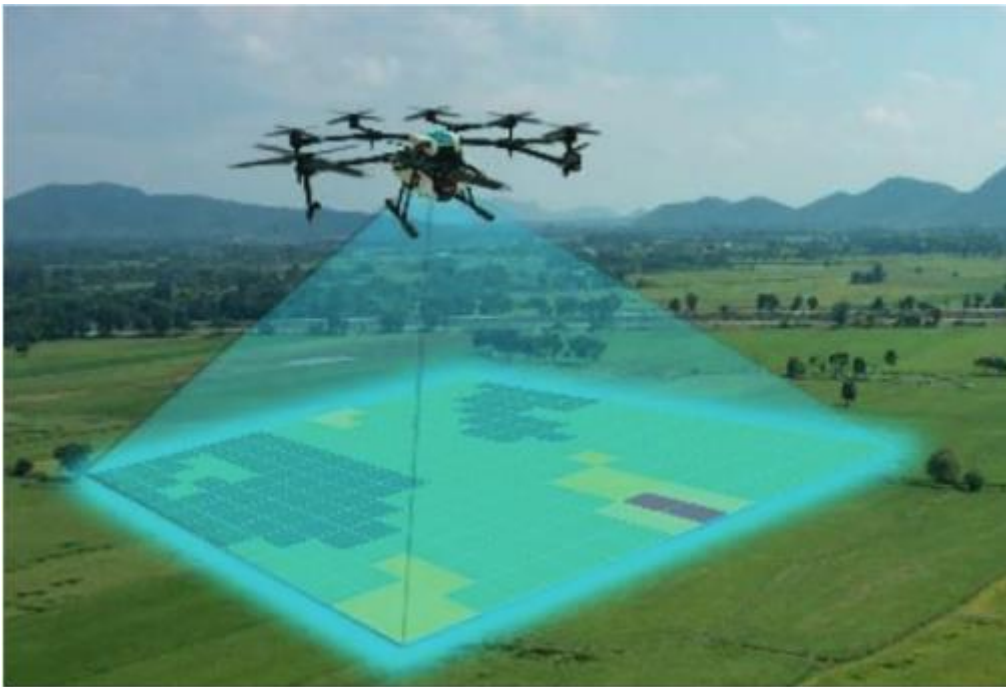


Figure 7: Cartographie avec un drone.

➤ **Logistique :**

Les drones sont utilisés dans l'industrie de la logistique pour transporter des aliments, des colis ou des marchandises. Il est préférable pour le transport de petits colis urgents ou fréquemment envoyés et pour la livraison dans des zones difficiles d'accès. L'un des problèmes les plus importants et en suspens dans l'utilisation de la logistique est que les drones ont besoin d'une capacité de charge suffisante.

Semblable à l'agriculture, il est également utilisé pour scanner différents matériaux d'entrepôt. Les drones lourds peuvent améliorer le trafic routier en remplaçant les transporteurs existants pour la gestion du matériel et le transport des marchandises entre les entrepôts. Bientôt, les drones pourraient être un acteur important dans les options de livraison des colis [13].

➤ **Urgences :**

Un autre domaine dans lequel les drones sont utilisés est celui des urgences. Les drones aident les pompiers, les policiers, les sauveteurs bénévoles, etc. Les drones sont utilisés pour la reconnaissance avant de guider l'équipe de secours, surtout lorsque l'étendue du sinistre est inconnue. Des véhicules sous-marins autonomes (AUV) préviennent d'éventuelles noyades et aident au sauvetage dans des situations comme lorsqu'un bateau chavire. Les drones sont également utilisés pour rechercher les blessés en cas d'avalanche. [13]

➤ **Conservation de la faune et des bâtiments historiques :**

Les drones sont une alternative économique et efficace pour surveiller les espèces sauvages et protéger la vie naturelle. Les observations aériennes rapides sont des outils importants pour avoir une meilleure idée de la santé de l'espèce et de l'écosystème, surveiller et rechercher les routes de migration en surveillant les groupes d'animaux et prévenir le braconnage. Les drones scannent également les sols forestiers détruits par les incendies, libérant des conteneurs de semences contenant des semences, des engrais et des nutriments pour aider les arbres à se régénérer. Considérant que le reboisement de ces zones avec du travail humain prendra de nombreuses années, l'importance de l'utilisation de la technologie des drones devient évidente. D'autre part, les drones jouent également un rôle important dans les efforts de préservation historique. Les drones offrent une grande commodité pour créer des cartes 3D de lieux ou de structures historiques. Les images 3D utilisées pour reconstruire les zones perdues fournissent des indices aux experts sur la culture et l'architecture. [13]



Figure 8: Drone logistique.

➤ **Militaire :**

La première utilisation connue des drones était à des fins militaires. Il est utilisé à différentes fins allant de l'armée, de l'espionnage, du système radar, de la détection de zone et de l'observation au transport de nourriture, d'armes et de munitions. Des drones hautement sophistiqués sont utilisés, équipés d'outils d'imagerie thermique, de télémètre laser et même de frappes aériennes. MQ-9 Reaper, l'un des drones utilisés à des fins militaires aujourd'hui, a une longueur de 36 pieds et est équipé d'un système de vol de 1852 kilomètres à une altitude de 50 000 pieds. [13]

2.4. Utilisation de drone dans ce projet

Dans ce projet, notre objectif est de concevoir un drone capable de détecter les départs de feu avant leur propagation, ainsi que de repérer les personnes éventuellement présentes, afin de faciliter les opérations de sauvetage.

3. Principe de fonctionnement et Modélisation d'un drone quadri-rotor

3.1. Fonctionnement de drone quadri-rotor

Un quadri-rotor est un drone doté de quatre rotors définis dans l'espace par six degrés de liberté (6 DDL), comprenant trois mouvements de rotation et trois mouvements de translation. Les quatre rotors sont généralement disposés aux extrémités d'une structure en croix, tandis que l'électronique de commande est placée au centre de cette croix.

Pour éviter que l'appareil ne tourne sur lui-même, les hélices opposées tournent dans des sens inverses : deux dans le sens horaire et deux dans le sens antihoraire. En tant que système sous-actionné, le quadri-rotor possède un nombre d'actionneurs inférieur à ses degrés de liberté, ce qui rend son fonctionnement spécifique.

En ajustant soigneusement les vitesses de rotation des moteurs, il est possible de contrôler ses mouvements : monter ou descendre, s'incliner à gauche ou à droite (roulis), en avant ou en arrière (tangage), ou encore pivoter sur lui-même (lacet). [14]

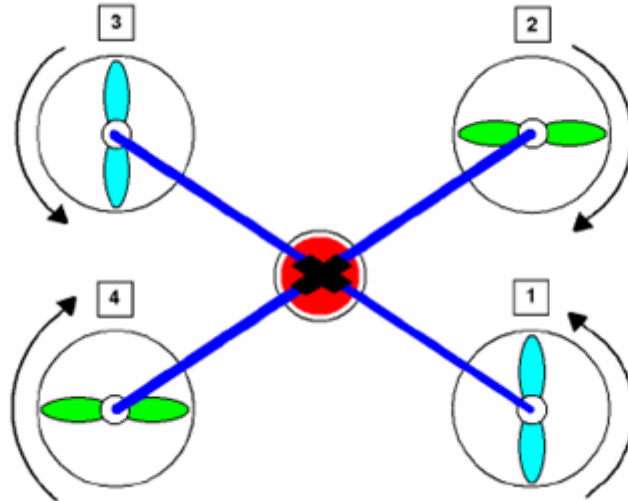


Figure 9 : Sens de rotation des rotors du quadri-rotor.

3.2. Les avantages quadri-rotor

Le quadri-rotor offre de réels avantages par rapport à d'autres configurations :

- ✓ Quatre petits rotors remplacent le grand rotor de l'hélicoptère ce qui réduit énormément l'énergie cinétique stockée et minimise les dégâts en cas d'accidents.

- ✓ Dynamique plus faible que celle de l'hélicoptère ce qui ne nécessite pas un temps de réaction rapide.
- ✓ Capacité de portance à cause de la présence de quatre rotors au lieu d'un.
- ✓ Commandé en variant seulement la vitesse de rotation des quatre moteurs.
- ✓ Décollage et atterrissage verticaux.
- ✓ Aucun embrayage n'est exigé entre le moteur et le rotor et aucune exigence sur l'angle d'attaque des rotors.
- ✓ Simplicité de la mécanique ce qui facilite la maintenance.
- ✓ Taille réduite et manœuvrabilité permettent de se déplacer dans des environnements fermés ou ouverts en évitant les obstacles. [14]

3.3. Mouvements du quadri-rotor

Les hélicoptères classiques utilisent un rotor principal qui génère un couple réactif, nécessitant un rotor de queue pour compenser cet effet, bien que celui-ci ne contribue pas à la poussée. À l'inverse, les quadri-rotors neutralisent le couple réactif grâce à leurs rotors opposés qui tournent dans des directions inverses (deux dans le sens horaire et deux dans le sens antihoraire). Cette configuration garantit la stabilité de l'appareil sans besoin d'un rotor de queue, permettant ainsi à toute l'énergie des moteurs de se concentrer sur la poussée.

Les mouvements d'un quadri-rotor sont contrôlés en ajustant individuellement les vitesses de rotation de ses moteurs. Ces ajustements inclinent le drone vers le rotor le plus lent, influençant plusieurs degrés de liberté simultanément. Par exemple, une augmentation de la vitesse d'un rotor gauche peut entraîner un roulis (inclinaison latérale), un lacet (rotation autour de l'axe vertical), et une translation latérale (déplacement horizontal). Les principaux mouvements contrôlables d'un quadri-rotor sont : [15]

- Mouvement vertical
- Mouvement de roulis
- Mouvement de tangage
- Mouvement de lacet
- Mouvement de translation

➤ **Mouvement vertical :**

Le mouvement vertical correspond tout simplement à la montée/descente du quadri-rotor. La montée est obtenue en augmentant à valeur égale la vitesse des quatre moteurs ce qui a pour effet d'annuler les couples de rotation générés. La descente, s'obtient par la réduction de la vitesse des moteurs. [14]

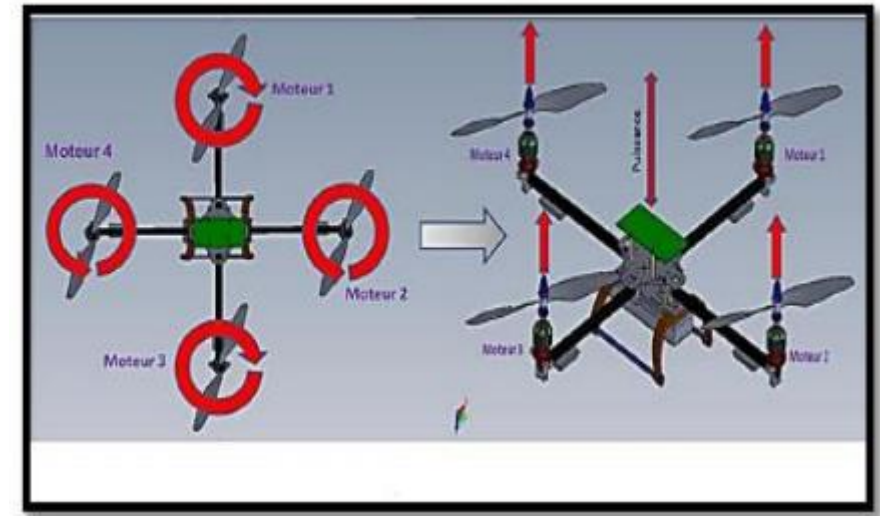


Figure 10 : Mouvement vertical.

➤ **Mouvement de roulis :**

Pour obtenir un mouvement de roulis, on applique un couple autour de l'axe, c'est-à-dire une différence de poussée entre le rotor (02) et le rotor (04). Ce mouvement (rotation autour de l'axe) est couplé avec un mouvement de translation selon l'axe y. [14]

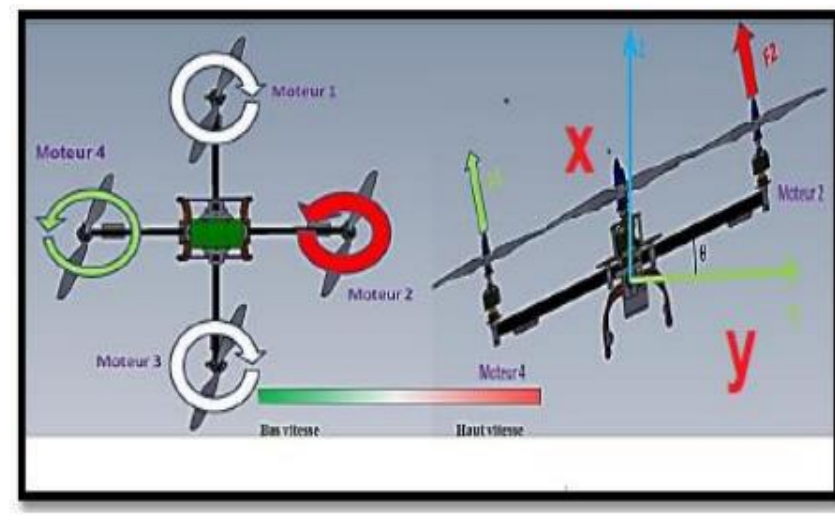


Figure 11: Mouvement de roulis.

➤ **Mouvement de tangage :**

De la même façon en appliquant un couple autour de l'axe y, donc une différence de poussée entre le rotor (01) et le rotor (03) on obtient un mouvement de tangage. Ce dernier est couplé avec un mouvement de translation selon l'axe. [14]

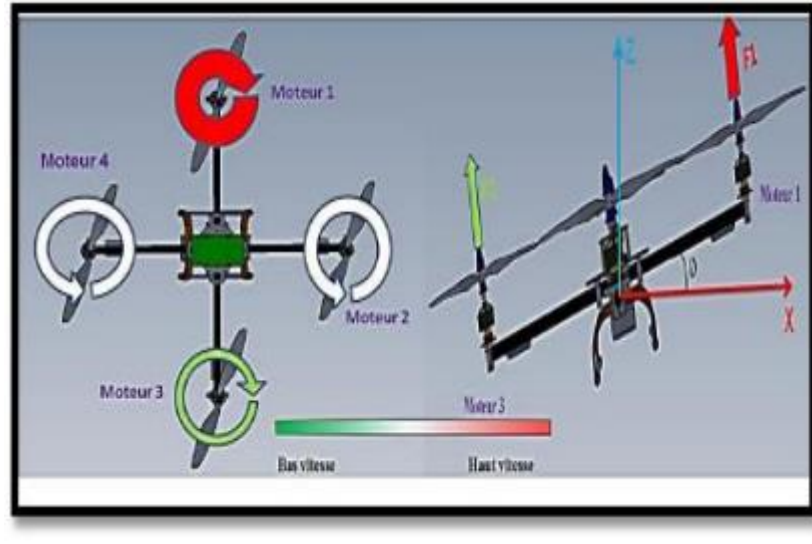


Figure 12 : Mouvement de tangage.

➤ **Mouvement de lacet :**

Le mouvement de lacet sert à faire tourner le quadri-rotor sur lui-même. Il est obtenu en augmentant la vitesse des rotors (1 et 3) et en diminuant proportionnellement la vitesse des rotors (2 et 4). [14]

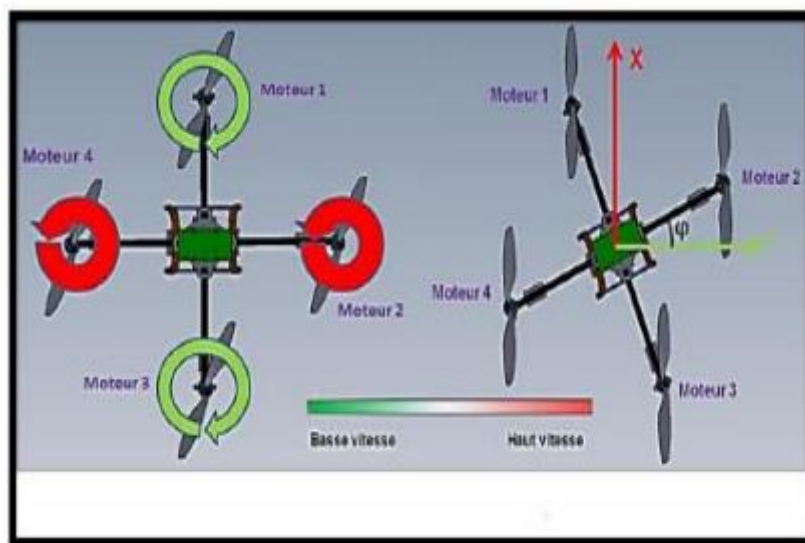


Figure 13 : Mouvement de lacet.

➤ **Mouvement de translation:**

Pour obtenir un mouvement de translation selon l'axe (X) ou (Y), il suffit de réaliser un roulis ou un tangage, en effet ces mouvements dépendent directement de la dynamique d'attitude. [14]

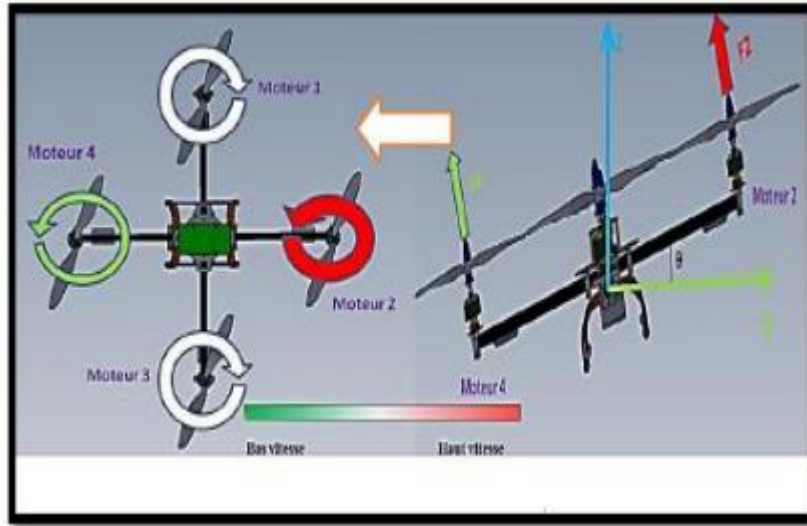


Figure 14 : Mouvement de translation.

4. Classification des drones

Les drones peuvent être classés de plusieurs manières, en fonction de leur taille, de leur utilisation, de leur type de propulsion ou de leur autonomie. Voici quelques-unes des classifications:

4.1. Selon le nombre d'hélices

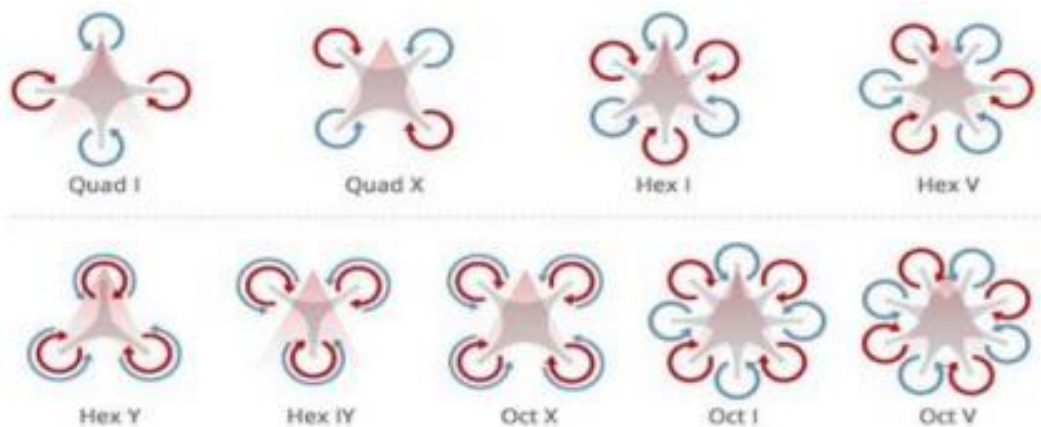


Figure 15 : Les différents nombres d'hélices.

➤ Drone à rotor :

Drone Mono-rotor (hélicoptères)

La construction la plus courante dans les drones à rotor est généralement celle des modèles de type multi-rotor qui possèdent plusieurs rotors pour maintenir la position de l'appareil, mais dans le cas d'un modèle mono-rotor, on trouve un seul rotor à l'intérieur. On y retrouve également un rotor de queue qui aide simplement à contrôler la direction de l'appareil. [16]



Figure 16 : Le drone mono-rotor.

Le tricoptère :

Il existe trois différents puissants moteurs à l'intérieur d'un tricoptère, trois contrôleurs, quatre gyros et un seul servo. Les moteurs sont simplement placés à chaque extrémité des trois bras et chacun d'entre eux porte un capteur de localisation. À chaque fois qu'on doit soulever le tricoptère, il est essentiel d'amorcer un mouvement du levier D'accélérateur, le capteur de gyroscope recevra immédiatement son signal et passera directement à la commande qui permet de contrôler la rotation du moteur. [16]



Figure 17 : Le tricoptère.

Quad-copter :

Lorsqu'un multi rotor est conçu avec quatre pales de rotor, alors il devient un quadricoptère. Ces dispositifs sont généralement contrôlés par des moteurs à courant continu de type brushless spécialement conçus pour. Deux des moteurs se déplacent dans le sens des aiguilles d'une montre, tandis que les deux autres fonctionnent dans le sens inverse des aiguilles d'une montre. [16]



Figure 18 : Le Quad-copter.

Hexacoptère:

Un hexacoptère servira pour de nombreuses applications potentielles avec son mécanisme à 6 moteurs, où 3 fonctionnent dans le sens des aiguilles d'une montre et les trois autres fonctionnent en sens inverse. Par conséquent, ces dispositifs peuvent obtenir une puissance de levage plus élevée par rapport aux quadri-coptères. [16]



Figure 19 : Hexacoptère .

Octocoptère:

Octo signifie huit. Donc l'octocoptère offre ses 8 puissants moteurs qui fournissent la puissance à 8 hélices fonctionnelles. Cet engin a naturellement des aptitudes de vol plus larges que les modèles évoqués plus haut, et il est également très stable. [16]



Figure 20 : L'Octocoptère.

➤ **Les drones aux ailes fixes:**

Voici une catégorie totalement différente de toutes celles présentées ci-dessus. Les designs sont tout à fait uniques par rapport aux drones de type multi-rotor couramment utilisés. Ils possèdent des ailes et ont l'apparence d'avions traditionnels. Ces drones ne sont pas capables de rester stables dans l'air car ils ne sont pas assez puissants pour lutter contre la force gravitationnelle. [16]



Figure 21: Les drones aux ailes fixes.

➤ Drones à ailes battantes :

Inspirés par les insectes, la construction de cette famille de drones consiste aux battements des ailes qui permettent de faire des vols stationnaires et d'imiter les trajectoires des insectes. [16]

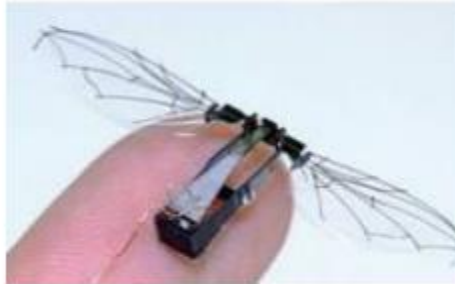


Figure 2.18 Drones à ailes battantes.

4.2. Selon la taille

On peut distinguer les catégories suivantes :

HALE (Haute Altitude Longue Endurance) :

Ce sont des drones, le plus souvent à voilure fixe, capables de rester très longtemps en vol et de collecter des informations sur de très longues périodes.

MALE (Moyenne Altitude Longue Endurance) :

Ayant une grande autonomie, ils sont utilisés pour des vols de longue durée à une moyenne altitude opérationnelle. Ces drones font partie de la classe de grande taille. Ils peuvent embarquer des armes, ce qui nécessite généralement d'avoir un humain dans la boucle, ce dernier doit garder la décision de tir et pouvoir à tout moment annuler la mission.

Micro drones :

Ce sont des drones ayant des tailles variant du centimètre à quelques dizaines de centimètres et caractérisés par une faible charge. Généralement, propulsé électriquement, ils permettent ainsi de faire des vols en intérieur.

Mini drones :

Ce sont des drones plutôt légers et de taille réduite (jusqu' à quelques kilogrammes et d'une envergure jusqu'à 1 à 2 mètres) facilitant la mise en œuvre d'une autonomie relativement faible (de 10 à 30 minutes). Ils sont, en général, utilisés pour l'observation de zones à accès difficiles. [17]

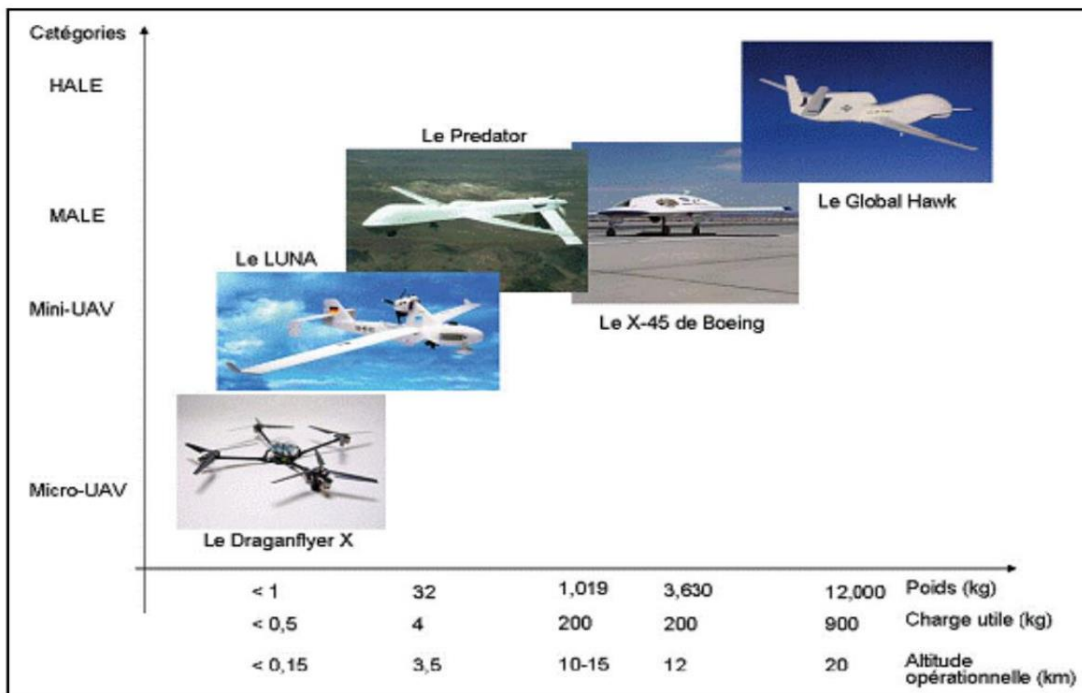


Figure 22 : Les Catégories des drones.

5. Avantages des drones à quatre rotors dans les incendies de forêt

- ✓ Surveillance en Temps Réel et Détection Précoce
- ✓ Soutien à la Sécurité des pompiers
- ✓ Coûts Réduits
- ✓ Accès aux Terrains Difficiles
- ✓ Collecte de Données

6. Conclusion

Dans ce chapitre, nous avons exploré les différentes composantes du quadrirotor ainsi que les principes de base de sa modélisation. Cette compréhension est

Chapitre 03 : Conception du Système

essentielle pour la conception et le contrôle efficaces de ces systèmes. En conclusion, les drones ont indéniablement révolutionné de nombreux secteurs, de l'agriculture à la sécurité civile en passant par l'industrie du divertissement. Leur capacité à atteindre des zones difficiles d'accès et à collecter des données précises en fait des outils précieux pour relever les défis du 21^{ème} siècle.

Chapitre 03 : Conception du Système

1. Introduction

Il est maintenant reconnu que le processus de développement du système d'information comprend deux étapes de base : l'étape de conception et l'étape de mise en œuvre. La phase de conception se concentre sur l'expression des besoins et la recherche d'une solution répondant à ces besoins, et vise à produire les différentes spécifications du système d'information, tandis que la phase de mise en œuvre se concentre sur l'application de la solution pour aboutir à un système logiciel performant. Dans ce chapitre, nous donnerons une description générale de notre système, ainsi qu'une explication des concepts de chacune de ses parties.

2. Conception général

Dans les forêts et les zones naturelles, il est essentiel de protéger les ressources et les vies humaines contre les incendies. En effet, ces dernières années, le nombre d'incendies dans les zones boisées a considérablement augmenté, souvent avec des conséquences dévastatrices. Par conséquent, la détection précoce des incendies et l'identification rapide des personnes présentes dans ces zones constituent des approches cruciales pour réduire les risques. Dans les systèmes de surveillance traditionnels, il est nécessaire de déployer de nombreux gardes forestiers et pompiers, nécessitant un énorme effort humain pour garantir la sécurité des personnes et des ressources naturelles.



Figure 23 : Photo du début d'un incendie dans la forêt

À cet égard, les drones peuvent être utilisés pour aider les équipes de sécurité en surveillant à distance les zones sensibles. Les drones offrent une immunité contre les dangers, et ils peuvent non seulement observer, mais aussi détecter et surveiller les signes d'incendie, tout en identifiant les personnes susceptibles de se trouver en danger. En exploitant des technologies de reconnaissance d'images, les drones peuvent assurer une surveillance efficace, et suivre les mouvements des individus en temps réel. Cette technologie permet une vue aérienne optimale pour surveiller les forêts, détecter les départs de feu et identifier les personnes présentes.

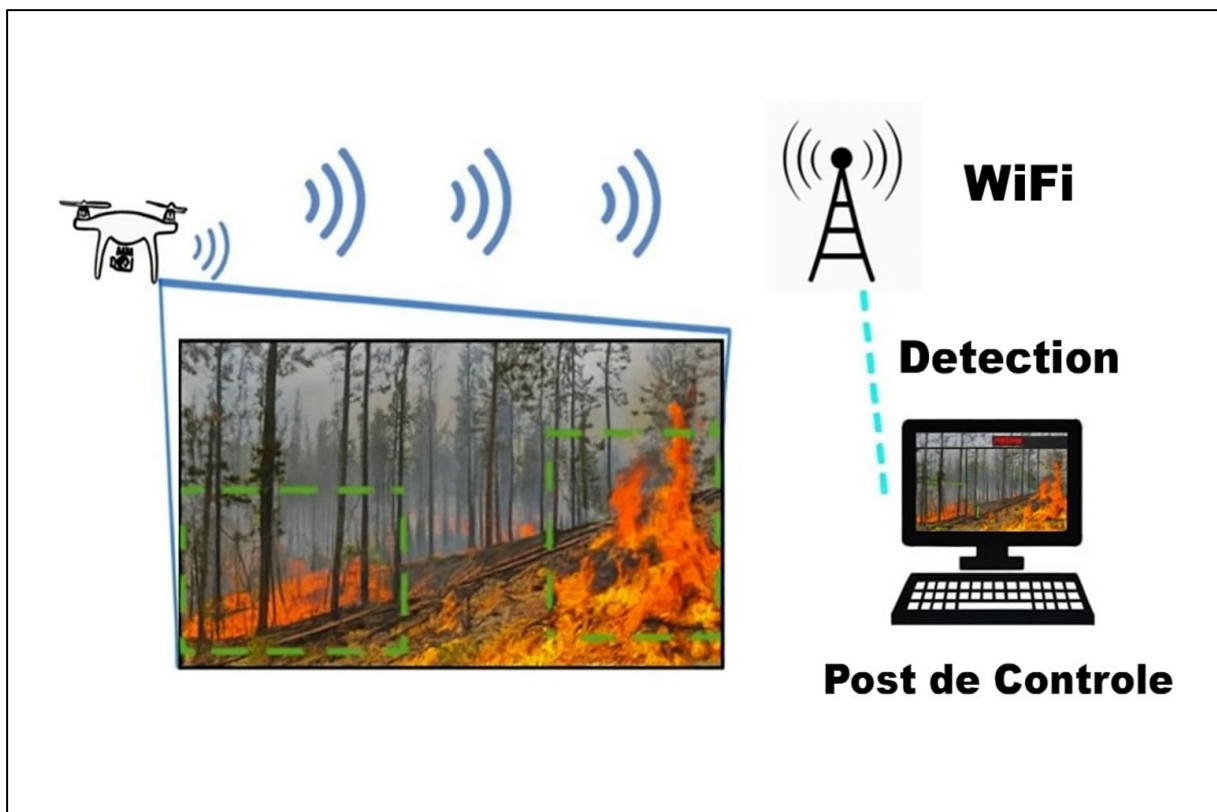


Figure 24 : Schéma du scénario expérimental

La figure 24 illustre le scénario expérimental envisagé. Dans ce cadre, un drone équipé d'une caméra est utilisé pour surveiller une zone spécifique et transmettre les images en temps réel via le réseau Wifi vers un poste de contrôle

La connexion Wifi permet à la station de contrôle d'accéder au flux vidéo en temps réel et d'analyser les images transmises. Dans le cadre de la surveillance des feux de forêt, un algorithme de détection (Yolov8) est appliqué pour identifier rapidement les zones brûlées.

L'algorithme Yolov8 est également utilisé pour détecter les personnes dans l'incendie. Le code source a été développé en utilisant le langage de programmation python.

3. Conception de drone

Afin de piloter notre drone, nous présentons ci-dessous le schéma descriptif du système (figure 25). Ce schéma met en évidence les différents composants :

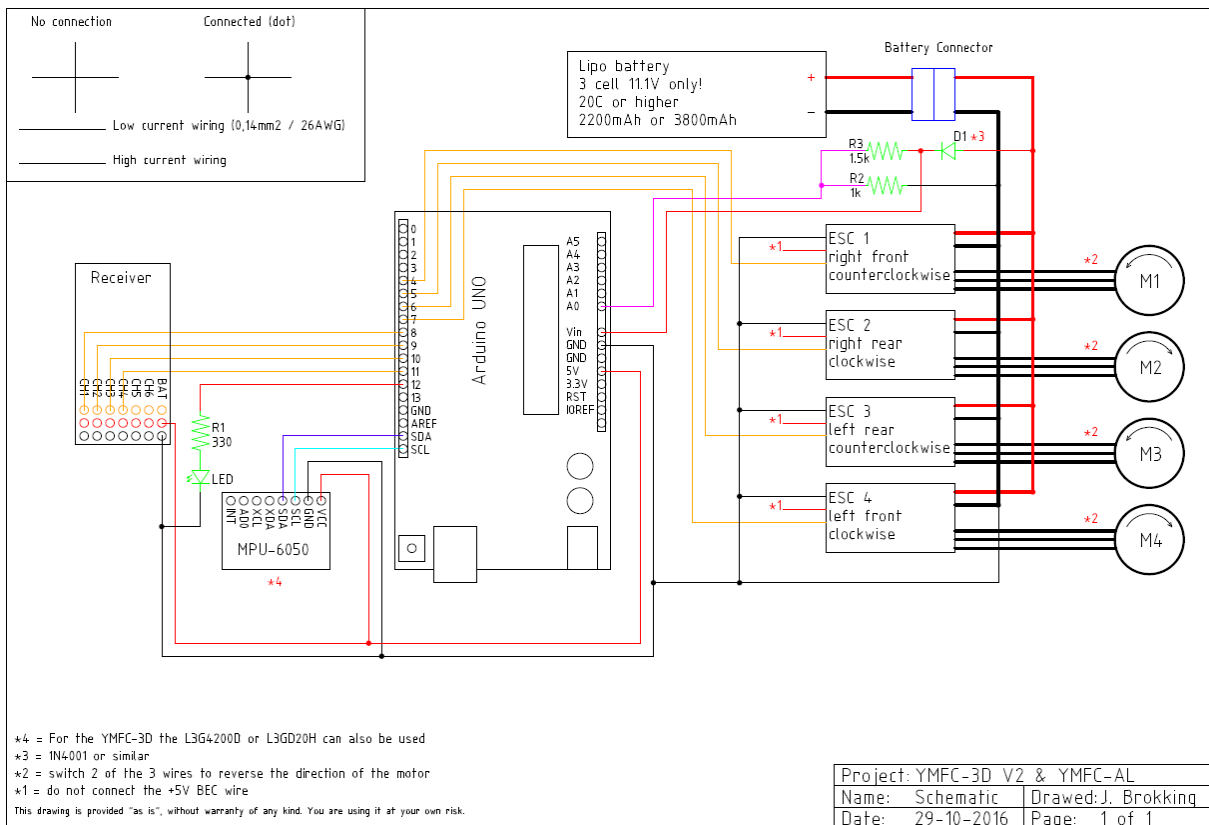


Figure 25 : Le schéma fonctionnel du quad-copter

4. Transmission vidéo

Nous souhaitons embarquer une caméra dans notre drone. Compte tenu des contraintes qui nous sont imposées spécialement de point de vue disponibilité du matériel et du poids additionnel à notre drone nous avons opté pour un module Raspberry Pi 3 avec une caméra. A l'aide de cet équipement nous serons en mesure de capter une vidéo et de la transmettre à travers le Wifi en temps réel.



Figure 26 : Raspberry Pi 3 et Camera

5. Présentation de l'UML

5.1. Diagramme de sequence

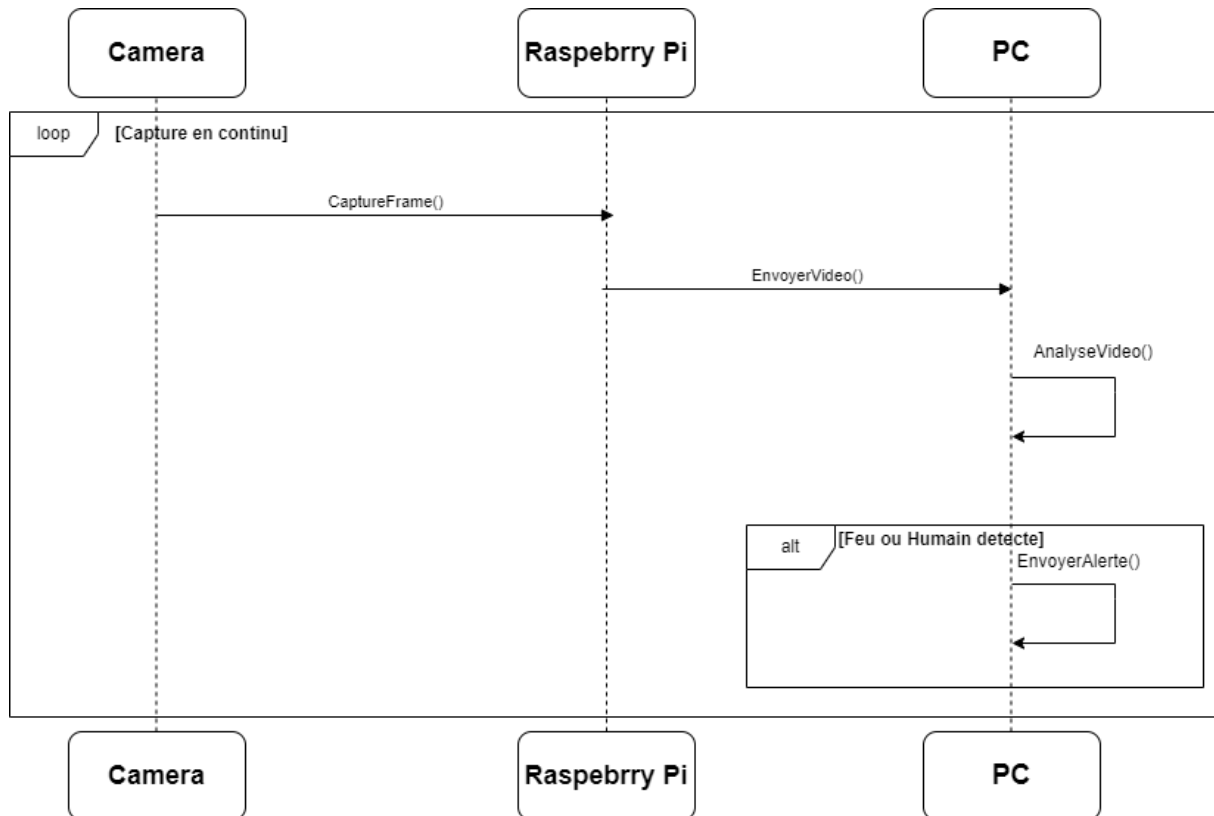


Figure 27 : Séquence de fonctionnement du drone (Détection de Feu/ Humain)

L'image représente un diagramme de séquence illustrant l'interaction entre une caméra, un Raspberry Pi et un PC dans un système de traitement d'images en temps réel. Le processus commence par la capture continue de vidéos ou d'images par la caméra, qui les envoie ensuite au Raspberry Pi pour un traitement préliminaire. Une fois les données reçues, le PC applique un algorithme d'analyse, comme la détection d'incendie ou la reconnaissance corps. Si un élément suspect est détecté, une alerte est générée, sinon le système poursuit la surveillance en boucle. Ce schéma illustre un système de surveillance automatisé basé sur un Raspberry Pi, utilisé comme passerelle entre la caméra et le PC, permettant une analyse efficace des données en temps réel.

5.2. Diagramme d'activité

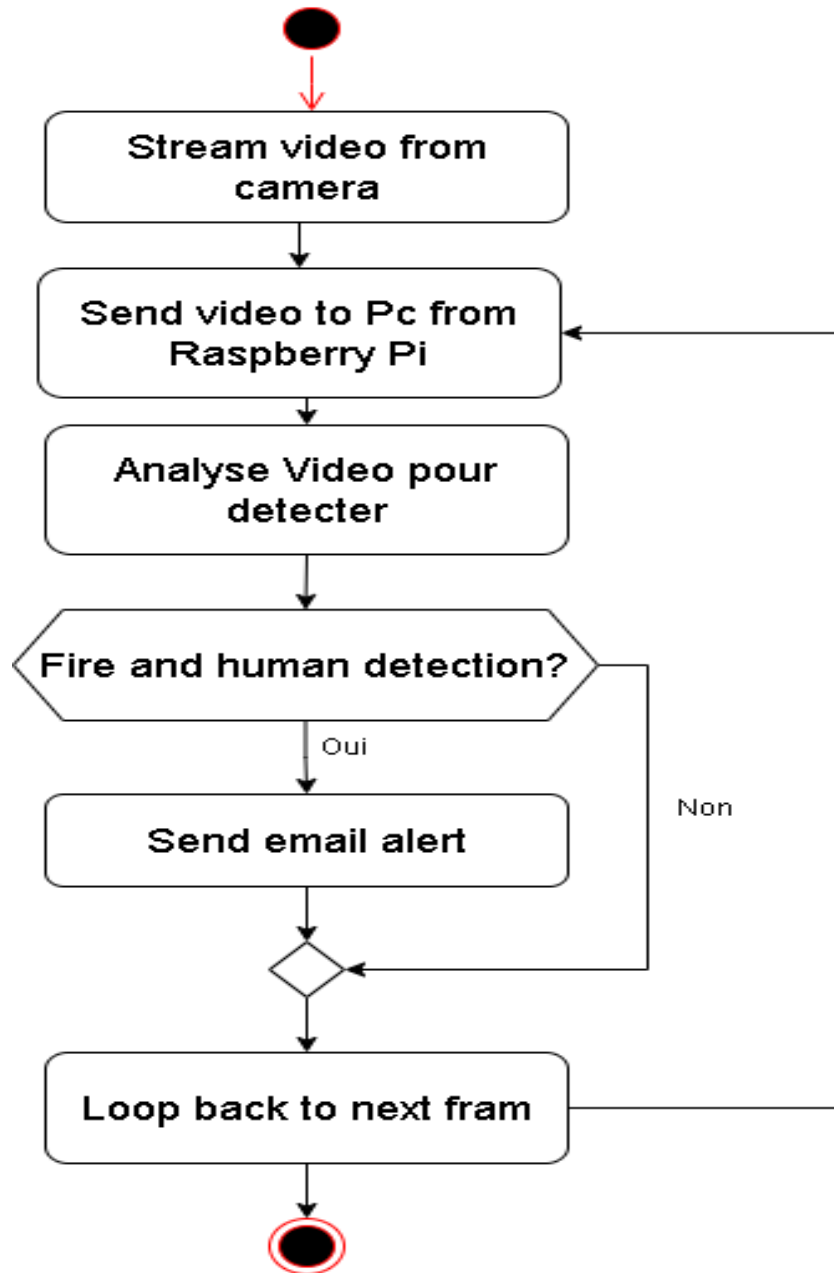


Figure 28: Diagramme d'activité

Ce diagramme montre le fonctionnement d'un système de surveillance où une caméra sur un drone diffuse une vidéo vers un Raspberry Pi, qui l'envoie ensuite à un PC pour analyser les images et détecter un feu ou une personne. En cas de détection, une alerte par email est envoyée ; sinon, le système passe à l'image suivante et recommence le processus en boucle.

6. Software (Setup and Test)

6.1. La procédure de configuration

1. Ouvrir YMFC-AL_setup fichier :

Nom	Modifié le	Type	Taille
YMFC-AL_setup	20/08/2016 22:19	Arduino file	40 Ko

Figure 29: YMFC-AL_setup fichier

2. Sélectionnez outils : Type de carte : Arduino Uno puis envoyez le code à Arduino :

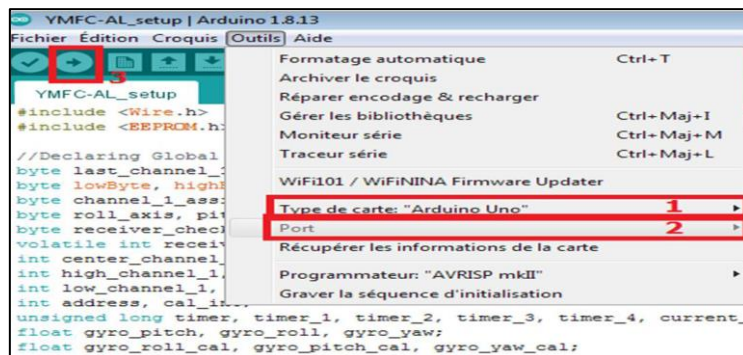


Figure 30: Sélectionnez outils

3. Démarrer la connexion série : vitesse 57600bps :

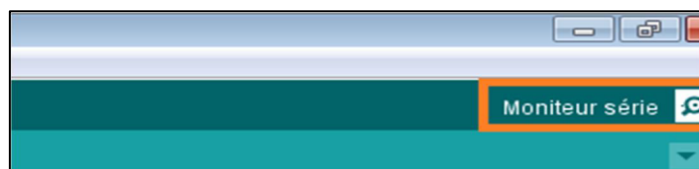


Figure 31: Démarrer la connexion série

4. Vérifie la vitesse de communication entre le contrôleur et le capteur MPU-6050 (réglé à 400 kHz) :

Affichage dans Arduino IDE

```
Checking I2C clock speed.  
Please wait  
TWRB =12  
I2C clock speed is correctly set to 400kHz.
```

5. Teste si le récepteur reçoit bien les signaux. Tu dois centrer les sticks pour enregistrer la position neutre :

Affichage dans Arduino IDE

```
Please wait
-Checking for valid receiver signals: OK
- Place all sticks and subtrims in the center position
within 10 seconds.
Center positions stored.
Digital input 08 = 1460
Digital input 09 = 1460
Digital input 10 = 1432
Digital input 11 = 1620
```

6. Tu bouges chaque stick (Throttle, Roll, YAW, Pitch) pour que le système reconnaisse quelle entrée est quoi :

Affichage dans Arduino IDE

```
Move the throttle stick to full throttle and back to center.
```

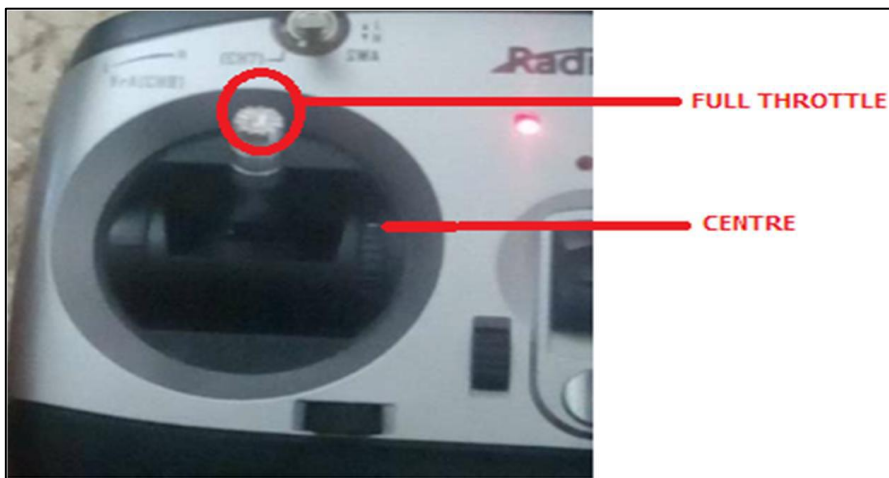


Figure 32: Full Throttle Position

Throttle is connected to digital input 10

Move the roll stick to simulate left wing up and back to center



Figure 33: Left Wing Up position

Roll is connected to digital input 8

Move the pitch stick to simulate nose up and back to center.



Figure 34: Nose Up Position

Pitch is connected to digital input 9

Move the yaw stick to simulate nose right and back to center.

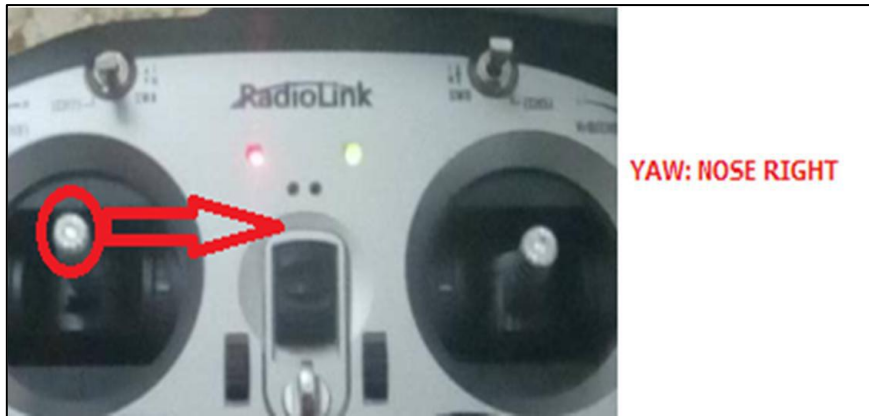


Figure 35: Nose Right Position

Yaw is connected to digital input 11

7. Le drone enregistre les valeurs minimum, maximum et centrale de chaque stick (utile pour la précision de vol) :

Affichage dans Arduino IDE

```
Gently move all the sticks simultaneously to their extends
When ready put the sticks back in their center positions
Measuring endpoints....
High, low and center values found during setup
Digital input 08 values : 996 - 1500 - 2000
Digital input 09 values : 996 - 1504 - 1996
Digital input 10 values : 996 - 1500 - 1964
Digital input 11 values : 996 - 1500 - 2000
Move stick 'nose up' and back to center to continue
```

8. Vérifie si le capteur MPU-6050 est bien connecté. Trouvé à l'adresse 0x68 :

Affichage dans Arduino IDE

```
Gyro search
Please wait
Searching for MPU-6050 on address 0x68/104
MPU-6050 found on address 0x68
```

9. Prépare le capteur en modifiant ses registres internes :

Affichage dans Arduino IDE

```
Gyro register settings
Register 0x6B is set to:0
Register 0x1B is set to:1000
```

10. Tu ne bouges rien. Il mesure les décalages (offsets) sur les 3 axes :

Affichage dans Arduino IDE

```
Gyro calibration
Don't move the quadcopter!! Calibration starts in 3
seconds
Calibrating the gyro, this will take +/- 8 seconds
Please wait.....
Axis 1 offset=-261.95
Axis 2 offset=38.16
Axis 3 offset=107.25
```

11. Tu inclines le drone (à gauche, nez vers le haut, rotation droite). Il vérifie que les directions sont correctes:

```
Gyro axes configuration
-Lift the left side of the quadcopter to a 45 degree angle
```

Affichage dans Arduino IDE

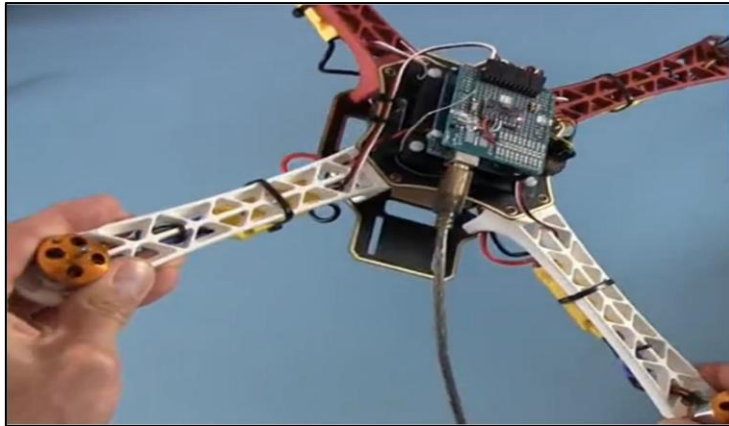


Figure 36: Lift Left Side 45°

OK!

Angle detection = 1

Put the quadcopter back in its original position

-Move stick 'nose up' and back to center to continue

-Lift the nose of the quadcopter to a 45 degree angle

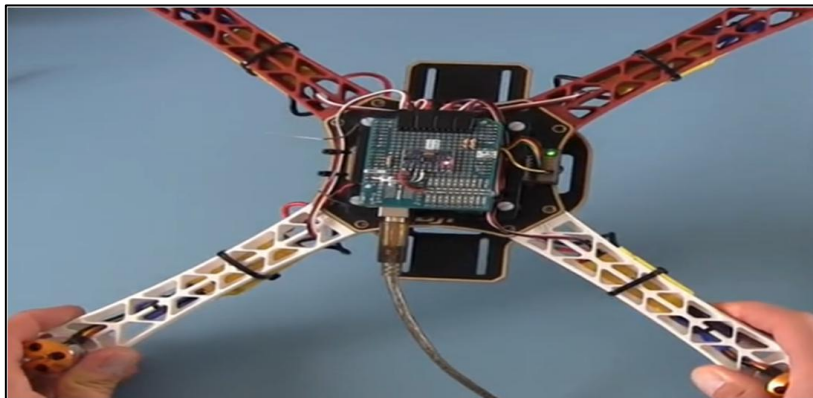


Figure 37: Lift Nose Up 45°

OK!

Angle detection = 2

Put the quadcopter back in its original position

-Move stick 'nose up' and back to center to continue

-Rotate the nose of the quadcopter 45 degree to the right within 10 seconds



Figure 38: Rotate Nose to Right 45°

```
OK!  
Angle detection = 3  
Move stick 'nose up' and back to center to continue
```

12. Allume une LED pour vérifier que le système réagit bien.

Affichage dans Arduino IDE

```
LED test  
The LED should now be lit  
-Move stick 'nose up' and back to center to continue  
Final setup check  
Receiver channels ok  
Gyro axes ok
```

13. Les canaux du récepteur et les axes gyro sont OK.

Affichage dans Arduino IDE

```
Storing EEPROM information
```

14. Enregistre toutes les données dans la mémoire (EEPROM) pour garder la config au redémarrage.

Affichage dans Arduino IDE

```
Writing EEPROM
Done!
Verify EEPROM data
Verification done
Setup is finished.
```

6.2. L'étalonnage du régulateur de vitesse électronique (ESC)

- 1- Ouvrir le fichier **YMFC-AL_esc_calibrate** dans l'IDE Arduino.
- 2- Lancer la connexion série avec une **vitesse de 57600 bps**.
- 3- Le programme démarre automatiquement en **mode calibration**.
- 4- Pour calibrer les ESCs :
- 5- Placer le stick throttle en **position haute**.
- 6- Connecter la batterie (LiPo).
- 7- Après quelques bips sonores, **placer le stick des throttle en position basse**.
- 8- Pour vérifier les entrées du récepteur est connecté ou non tapez : **r**.
- 9- Bouger les sticks et vérifier si les valeurs **correspondent bien aux mouvements** (min=1000us, center=1500us, max=2000us)
- 10- Pour vérifier les angles du quadcoptère tapez : **a**.
 - Ne pas bouger le drone pendant la calibration, car le gyroscope doit s'étalonner.
 - Une fois calibré, les **angles de roulis (roll) et de tangage (pitch)** sont affichés.
- 11- Modifier les **valeurs initiales de roll et pitch** dans le fichier YMFC-AL_Flight_controller en utilisant les nouvelles valeurs affichées.
- 12- Vérifier le **mouvement des angles** du quadcoptère :
 - Nose up positive pitch

- Nose down negative pitch
- Left wing up positive roll
- Left wing down negative roll
- Nose right positive yaw
- Nose left negative yaw

6.3. Procédure d'équilibrage des moteurs et hélices

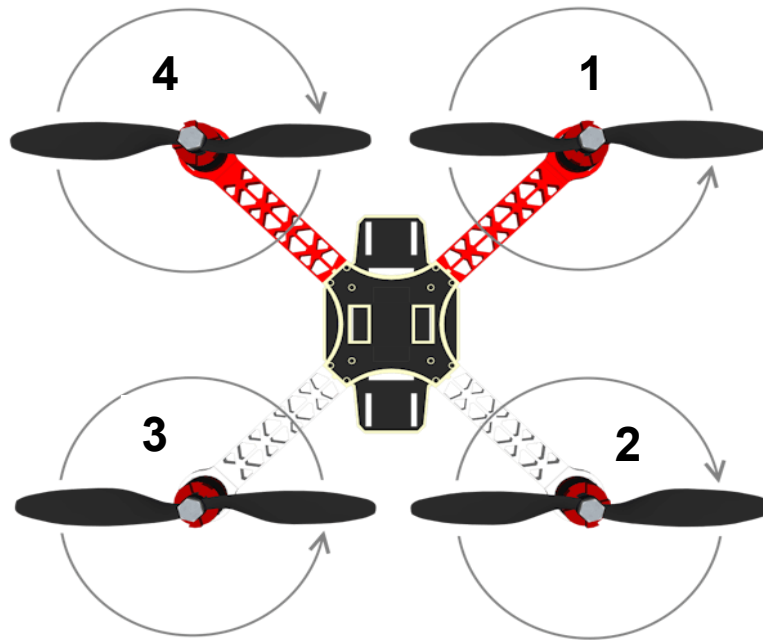


Figure 39: Le sens de rotation du moteur

Un bon équilibre est essentiel ! Des hélices ou des moteurs mal équilibrés provoquent des vibrations qui perturbent le gyroscope et l'accéléromètre, rendant le drone instable et incapable de se stabiliser.

Installez les ventilateurs et vérifiez qu'ils sont dans le bon sens (CW/ CCW).

Upload le code YMFC-AL_esc_calibrate sur Arduino.

Ouvrez le moniteur série (bauds: 57600).

Envoyez "1" à "4" pour tester chaque moteur séparément, ou "5" pour les tester tous ensemble.

Tenez fermement le drone, réglez throttle à zéro, puis connectez la batterie.

Augmentez lentement throttle pour faire tourner le moteur testé, vérifiez le sens de rotation et la poussée vers le haut.

Deux moteurs tournent dans le même sens horaire et deux moteurs tournent dans le sens antihoraire.

Deux moteurs opposés tournent dans le même sens.

Si le moteur fonctionne dans le mauvais sens, inversez deux des trois fils du moteur.

Dès que le moteur s'arrête, maintenez fermement le levier moteur, redémarrez - le au milieu de la pédale d'accélérateur, remarquez les valeurs de vibration affichées et ce que vous ressentez.

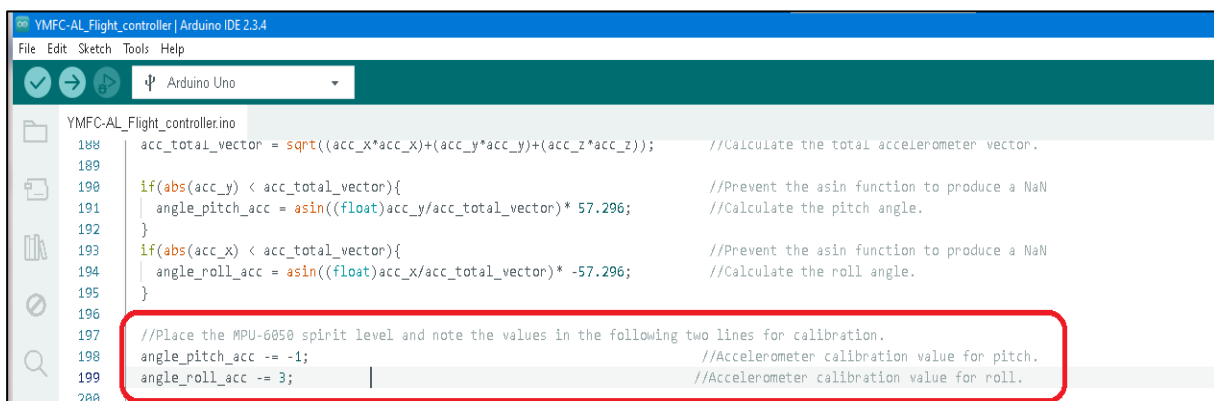
Ajoutez un petit morceau de ruban adhésif à la lame, testez-le à nouveau. Répétez jusqu'à ce que les vibrations soient minimales.

Répétez ces étapes pour chaque moteur.

Enfin, envoyez ' 5 ' pour tester les quatre moteurs ensemble.

6.4. Le code du contrôleur de vol

Voici les étapes à suivre pour modifier les valeurs de calibration de l'accéléromètre et uploader le fichier sur l'Arduino :



```
188 acc_total_vector = sqrt((acc_x*acc_x)+(acc_y*acc_y)+(acc_z*acc_z)); //Calculate the total accelerometer vector.
189
190 if(abs(acc_y) < acc_total_vector){ //Prevent the asin function to produce a NaN
191   angle_pitch_acc = asin((float)acc_y/acc_total_vector)* 57.296; //Calculate the pitch angle.
192 }
193 if(abs(acc_x) < acc_total_vector){ //Prevent the asin function to produce a NaN
194   angle_roll_acc = asin((float)acc_x/acc_total_vector)* -57.296; //Calculate the roll angle.
195 }
196
197 //Place the MPU-6050 spirit level and note the values in the following two lines for calibration.
198 angle_pitch_acc -= -1; //Accelerometer calibration value for pitch.
199 angle_roll_acc -= 3; //Accelerometer calibration value for roll.
200
```

Figure 40: Les valeurs de calibration

angle_pitch_acc = -1.0; Valeur d'étalonnage de l'accéléromètre pour la hauteur.

angle_roll_acc = 3,0; Valeur d'étalonnage de l'accéléromètre pour le roulis.

Modifiez les valeurs angle_pitch_acc et angle_roll_acc avec les valeurs indiquées dans vérification les angles .

Déconnectez la batterie et upload le fichier YMFC-AL_Flight_controller sur l'Arduino.
Déconnecter le câble USB et connectez la batterie.

6.5. PID Controller (Stabilisation)

Les contrôleurs PID sont souvent utilisés pour contrôler un drone. Il s'agit d'un programme dans le drone, qui évalue les données du capteur et entraîne les moteurs en conséquence via le contrôleur de moteur. PID signifie: proportionnelle-intégrale-dérivée.

Essayez de voler avec les paramètres PID standard.

```
float pid_p_gain_roll = 1.4;           //Gain setting for the roll P-controller
float pid_i_gain_roll = 0.04;          //Gain setting for the roll I-controller
float pid_d_gain_roll = 18.0;         //Gain setting for the roll D-controller
int pid_max_roll = 400;                //Maximum output of the PID-controller (+/-)

float pid_p_gain_pitch = pid_p_gain_roll; //Gain setting for the pitch P-controller.
float pid_i_gain_pitch = pid_i_gain_roll; //Gain setting for the pitch I-controller.
float pid_d_gain_pitch = pid_d_gain_roll; //Gain setting for the pitch D-controller.
int pid_max_pitch = pid_max_roll;       //Maximum output of the PID-controller (+/-)

float pid_p_gain_yaw = 4.0;            //Gain setting for the pitch P-controller. //4.0
float pid_i_gain_yaw = 0.02;           //Gain setting for the pitch I-controller. //0.02
float pid_d_gain_yaw = 0.0;            //Gain setting for the pitch D-controller.
int pid_max_yaw = 400;                 //Maximum output of the PID-controller (+/-)
```

Chaque quadricoptère nécessite des réglages PID spécifiques à sa configuration. l'utilisation de réglages PID préexistants a conduit à un comportement instable du quadrirotor. Il a donc été nécessaire de procéder à un ajustement personnalisé des paramètres, en s'appuyant sur la méthode de base de réglage des contrôleurs PID.

```
float pid_p_gain_roll = 1.4;           //Gain setting for the roll P-controller
float pid_i_gain_roll = 0.01;          //Gain setting for the roll I-controller
float pid_d_gain_roll = 16.0;         //Gain setting for the roll D-controller
int pid_max_roll = 400;                //Maximum output of the PID-controller (+/-)

float pid_p_gain_pitch = pid_p_gain_roll; //Gain setting for the pitch P-controller.
float pid_i_gain_pitch = pid_i_gain_roll; //Gain setting for the pitch I-controller.
float pid_d_gain_pitch = pid_d_gain_roll; //Gain setting for the pitch D-controller.
int pid_max_pitch = pid_max_roll;       //Maximum output of the PID-controller (+/-)

float pid_p_gain_yaw = 3.0;            //Gain setting for the pitch P-controller. //4.0
float pid_i_gain_yaw = 0.02;           //Gain setting for the pitch I-controller. //0.02
float pid_d_gain_yaw = 0.0;            //Gain setting for the pitch D-controller.
int pid_max_yaw = 400;                 //Maximum output of the PID-controller (+/-)
```

La méthode de base pour le réglage PID:

Set (pour roulis, tangage et lacet):

pid_p_gain=1

pid_i_gain=0

pid_d_gain=0

```
float pid_p_gain_roll = 1.0;
float pid_i_gain_roll = 0.0;
float pid_d_gain_roll = 0.0;
int pid_max_roll = 400;

float pid_p_gain_pitch = pid_p_gain_roll;
float pid_i_gain_pitch = pid_i_gain_roll;
float pid_d_gain_pitch = pid_d_gain_roll;
int pid_max_pitch = pid_max_roll;

float pid_p_gain_yaw = 1.0;
float pid_i_gain_yaw = 0.0;
float pid_d_gain_yaw = 0.0;
int pid_max_yaw = 400;
```

Tenez le multicoptère dans votre main et augmentez la manette throttle, déplacez-le dans 4 directions, vérifiez si le contrôleur de vol corrige la direction du vol.

Si ce n'est pas le cas, vérifiez les directions du gyroscope avec le fichier YMFC-AL_setup.

Set pid_p_gain_roll=0

pid_p_gain_yaw=3

pid_i_gain_yaw=0.02

```
float pid_p_gain_roll = 0.0;
float pid_i_gain_roll = 0.0;
float pid_d_gain_roll = 0.0;
int pid_max_roll = 400;

float pid_p_gain_pitch = pid_p_gain_roll;
float pid_i_gain_pitch = pid_i_gain_roll;
float pid_d_gain_pitch = pid_d_gain_roll;
int pid_max_pitch = pid_max_roll;

float pid_p_gain_yaw = 3.0;
float pid_i_gain_yaw = 0.02;
float pid_d_gain_yaw = 0.0;
int pid_max_yaw = 400;
```

Augmentez pid_d_gain_roll jusqu'à ce que le démarrage du quadricoptère oscille.

Diminuez pid_d_gain_roll jusqu'à ce que le quadricoptère fonctionne à nouveau en douceur.

Diminuez pid_d_gain_roll de 25%, dans mon cas résultats finaux pid_d_gain_roll=16.

Augmentez pid_p_gain_roll de l'étape 0.2 jusqu'à la première oscillation.

Diminuez la valeur de pid_p_gain_roll de 50%, dans notre cas pid_p_gain_roll=1.4.

Augmentez pid_d_gain_roll de l'étape 0.01 jusqu'à la première oscillation.

Diminuez la valeur de `pid_i_gain_roll` de 50%, dans notre cas `pid_i_gain_roll=0.01`.

Augmentez à nouveau `pid_p_gain_roll` jusqu'à la première oscillation.

Diminuez la valeur de `pid_p_gain_roll` jusqu'à atteindre un vol stable.

7. Conclusion

Ce chapitre a présenté un système de surveillance innovant basé sur des drones pour la détection précoce des incendies et l'identification des personnes dans les zones forestières. Face à l'augmentation des incendies et à leurs conséquences dévastatrices, les drones équipés de technologies de reconnaissance d'images offrent une solution efficace pour surveiller les zones à risque en temps réel, tout en réduisant la dépendance aux efforts humains. La conception technique du drone, incluant des composants tels que les hélices, les moteurs, la carte Arduino, la Raspberry Pi, et une caméra pour la transmission vidéo par Wifi, permet une surveillance aérienne optimale. Ce système combine innovation technologique et efficacité opérationnelle pour protéger les ressources naturelles et les vies humaines.

Chapitre 04 : Réalisation du Système

1. Introduction

Ce chapitre présente en détail le processus de développement de notre solution, en expliquant les différentes étapes mises en œuvre pour assurer son bon fonctionnement. Il décrit l'architecture du système, son mode de fonctionnement ainsi que son implémentation logicielle. Une attention particulière est portée aux composants matériels et logiciels utilisés, accompagnée d'illustrations permettant de visualiser le système en action. L'objectif est d'offrir une compréhension claire et technique de la réalisation, afin d'en faciliter l'utilisation, la maintenance et les améliorations futures.

2. Langages et outils de développement

2.1. Python

Python est un langage de programmation de haut niveau à usage général connu pour sa lisibilité et sa polyvalence. Prenant en charge plusieurs paradigmes tels que la programmation procédurale, orientée objet et fonctionnelle, il propose un typage dynamique et une gestion automatique de la mémoire. Créé par Guido van Rossum à la fin des années 1980 et publié pour la première fois en 1991, Python a évolué à travers les versions majeures, Python 3 remplaçant Python 2. Largement utilisé dans divers domaines, en particulier dans l'apprentissage automatique, Python reste l'un des langages de programmation les plus populaires en raison de sa simplicité et de sa bibliothèque standard étendue. [25]

2.2. OpenCV

OpenCV (Open Source Computer Vision Library) est une bibliothèque de fonctions de programmation principalement pour la vision par ordinateur en temps réel. Développé à l'origine par Intel, il a ensuite été pris en charge par Willow Garage, puis Itseez (qui a ensuite été acquis par Intel). La bibliothèque est multiplateforme et sous licence en tant que logiciel libre et open source sous licence Apache 2. À partir de 2011, OpenCV propose une accélération GPU pour les opérations en temps réel. [26]

2.3. PyCharm

PyCharm est un environnement de développement intégré (IDE) multi plateforme pour Python, développé par JetBrains et construit sur la plate-forme IntelliJ. Il offre une analyse de code, un débogueur graphique, des tests unitaires, une intégration de contrôle de version et un support pour le développement Web avec Django. Disponible en deux éditions, l'Édition Professionnelle (propriétaire) et l'Édition Communautaire (open-source sous licence Apache), PyCharm fournit des outils puissants pour le développement Python, l'Édition Professionnelle offrant des fonctionnalités plus avancées. [27]

2.4. Arduino IDE

L'IDE Arduino est une application multiplateforme de programmation de cartes Arduino, basée sur l'IDE de traitement et écrite en Java. Il dispose d'un éditeur de code avec coloration syntaxique, indentation automatique et outils simples pour compiler et télécharger du code. Prenant en charge C et C++, il utilise l'API de câblage et comprend une bibliothèque logicielle pour les opérations d'E/S courantes. L'IDE utilise avrdude pour convertir et télécharger le code exécutable sur la carte. Initialement conçu pour les Arduinos basés sur Atmel AVR, il prend désormais en charge diverses plates-formes telles que PIC, STM32, MSP430 et ESP 32, ce qui le rend largement utilisé dans le développement embarqué. [28]

2.5. Yolov8

YOLOv8 (You Only Look Once version 8) est la dernière itération de la série de modèles de vision par ordinateur en temps réel développée par Ultralytics. Il est conçu pour des tâches telles que la détection d'objets, la classification d'images, la segmentation d'instances et la détection de points clés.

YOLOv8 introduit plusieurs améliorations architecturales par rapport à ses prédécesseurs, notamment un mécanisme de détection sans ancrage (anchor-free), des modules convolutifs optimisés, ainsi qu'une interface Python simplifiée.

Ces avancées permettent d'atteindre une meilleure précision et une plus grande efficacité, rendant YOLOv8 adapté à un large éventail de plateformes matérielles, allant des appareils en périphérie (edge devices) aux serveurs cloud. [21]

3. Les Mouvements d'une manette

3.1. Schéma d'une manette de drone



Figure 41: Manette de drone


Cette manette sert à piloter le drone à distance en lui envoyant des signaux de commande :

Mise en route des moteurs, décollage, atterrissage et autres mouvements (tels mentionnés sur la figure ci-dessous) via une antenne radio.

Le contrôle de commande est correspond à :

- Stick gauche: contrôle Throttle (Up/Down) et Yaw (Left/Right).
- Stick droit : contrôle Pitch (Forward/Backward) et Roll (Left/Right).

3.2. Move Up/Down (Throttle Control)



```
1 // Read and set the throttle value from the receiver input (channel 3).
2 throttle = receiver_input_channel_3; // Throttle value ranges from 1000 to 2000 microseconds.
3
4 // Adjust motor speeds based on throttle.
5 if (start == 2) { // Only adjust motor speeds if the motors are started.
6   esc_1 = throttle - pid_output_pitch + pid_output_roll - pid_output_yaw; // Front-right motor.
7   esc_2 = throttle + pid_output_pitch + pid_output_roll + pid_output_yaw; // Rear-right motor.
8   esc_3 = throttle + pid_output_pitch - pid_output_roll - pid_output_yaw; // Rear-left motor.
9   esc_4 = throttle - pid_output_pitch - pid_output_roll + pid_output_yaw; // Front-left motor.
10
11 // Ensure motor speeds stay within safe limits.
12 if (esc_1 < 1100) esc_1 = 1100;
13 if (esc_2 < 1100) esc_2 = 1100;
14 if (esc_3 < 1100) esc_3 = 1100;
15 if (esc_4 < 1100) esc_4 = 1100;
16
17 if (esc_1 > 2000) esc_1 = 2000;
18 if (esc_2 > 2000) esc_2 = 2000;
19 if (esc_3 > 2000) esc_3 = 2000;
20 if (esc_4 > 2000) esc_4 = 2000;
21 }
```

Figure 42: Algorithmme Move Up/Down

Ce code gère le contrôle de Throttle Contrôle du drone quadricoptère, permettant sa montée et sa descente. Il commence par lire la valeur de Throttle depuis le canal 3 du récepteur, avec une plage comprise entre 1000 μ s (minimum) et 2000 μ s (maximum). Ensuite, si les moteurs sont activés, les vitesses de chaque moteur sont ajustées en fonction de la valeur de la Throttle et des sorties PID (pitch, roll et yaw) afin d'assurer la stabilité du drone. Chaque moteur reçoit une valeur spécifique en ajoutant ou soustrayant ces corrections PID pour contrôler l'inclinaison, l'orientation et la rotation du drone. Enfin, pour éviter des vitesses excessives ou trop faibles qui pourraient compromettre la stabilité du vol, les valeurs des moteurs sont limitées entre 1100 μ s et 2000 μ s. Ainsi, ce code permet de garantir un contrôle précis de l'altitude tout en maintenant l'équilibre et la sécurité du drone.

3.3. Move Left/Right (Roll Control)

```
1 // Calculate the roll setpoint based on receiver input (channel 1).
2 pid_roll_setpoint = 0; // Initialize the roll setpoint to zero.
3
4 // Check if the roll stick is moved left or right.
5 if (receiver_input_channel_1 > 1508) {
6     pid_roll_setpoint = receiver_input_channel_1 - 1508; // Move right.
7 } else if (receiver_input_channel_1 < 1492) {
8     pid_roll_setpoint = receiver_input_channel_1 - 1492; // Move left.
9 }
10
11 // Adjust the roll setpoint for auto-leveling (if enabled).
12 pid_roll_setpoint -= roll_level_adjust;
13
14 // Scale the roll setpoint to a reasonable range.
15 pid_roll_setpoint /= 3.0; // Divide by 3 to limit the maximum roll angle.
16
17 // Calculate the PID output for roll stabilization.
18 calculate_pid();
19
20 // Adjust motor speeds to tilt the drone left or right.
21 if (start == 2) { // Only adjust motor speeds if the motors are started.
22     esc_1 = throttle - pid_output_pitch + pid_output_roll - pid_output_yaw; // Front-right motor.
23     esc_2 = throttle + pid_output_pitch + pid_output_roll + pid_output_yaw; // Rear-right motor.
24     esc_3 = throttle + pid_output_pitch - pid_output_roll - pid_output_yaw; // Rear-left motor.
25     esc_4 = throttle - pid_output_pitch - pid_output_roll + pid_output_yaw; // Front-left motor.
26
27     // Ensure motor speeds stay within safe limits.
28     if (esc_1 < 1100) esc_1 = 1100;
29     if (esc_2 < 1100) esc_2 = 1100;
30     if (esc_3 < 1100) esc_3 = 1100;
31     if (esc_4 < 1100) esc_4 = 1100;
32
33     if (esc_1 > 2000) esc_1 = 2000;
34     if (esc_2 > 2000) esc_2 = 2000;
35     if (esc_3 > 2000) esc_3 = 2000;
36     if (esc_4 > 2000) esc_4 = 2000;
37 }
```

Figure 43: Algorithme Move Left/Right

Ce code contrôle l'inclinaison du drone vers la gauche ou la droite (Roll Control). Il commence par lire la position du joystick (canal 1 du récepteur).

- Si le joystick est déplacé vers la droite (valeur > 1508), le drone s'incline vers la droite.
- Si le joystick est déplacé vers la gauche (valeur < 1492), le drone s'incline vers la gauche.

Ensuite, il applique une correction automatique pour stabiliser le drone. La valeur du roulis est aussi divisée par 3 pour éviter une inclinaison trop forte.

Après cela, le code ajuste la vitesse des moteurs :

- Pour incliner le drone à droite, il réduit la vitesse des moteurs avant-droit et arrière-gauche.
- Pour incliner le drone à gauche, il réduit la vitesse des moteurs avant-gauche et arrière-droit.

Enfin, il s'assure que la vitesse des moteurs reste dans une plage sécurisée (entre 1100 μ s et 2000 μ s). Cela permet au drone de s'incliner en douceur tout en restant stable.

3.4. Move Forward/Backward (Pitch Control)

```
1 // Calculate the pitch setpoint based on receiver input (channel 2).
2 pid_pitch_setpoint = 0; // Initialize the pitch setpoint to zero.
3
4 // Check if the pitch stick is moved forward or backward.
5 if (receiver_input_channel_2 > 1508) {
6   pid_pitch_setpoint = receiver_input_channel_2 - 1508; // Move forward.
7 } else if (receiver_input_channel_2 < 1492) {
8   pid_pitch_setpoint = receiver_input_channel_2 - 1492; // Move backward.
9 }
10
11 // Adjust the pitch setpoint for auto-leveling (if enabled).
12 pid_pitch_setpoint -= pitch_level_adjust;
13
14 // Scale the pitch setpoint to a reasonable range.
15 pid_pitch_setpoint /= 3.0; // Divide by 3 to limit the maximum pitch angle.
16
17 // Calculate the PID output for pitch stabilization.
18 calculate_pid();
19
20 // Adjust motor speeds to tilt the drone forward or backward.
21 if (start == 2) { // Only adjust motor speeds if the motors are started.
22   esc_1 = throttle - pid_output_pitch + pid_output_roll - pid_output_yaw; // Front-right motor.
23   esc_2 = throttle + pid_output_pitch + pid_output_roll + pid_output_yaw; // Rear-right motor.
24   esc_3 = throttle + pid_output_pitch - pid_output_roll - pid_output_yaw; // Rear-left motor.
25   esc_4 = throttle - pid_output_pitch - pid_output_roll + pid_output_yaw; // Front-left motor.
26
27   // Ensure motor speeds stay within safe limits.
28   if (esc_1 < 1100) esc_1 = 1100;
29   if (esc_2 < 1100) esc_2 = 1100;
30   if (esc_3 < 1100) esc_3 = 1100;
31   if (esc_4 < 1100) esc_4 = 1100;
32
33   if (esc_1 > 2000) esc_1 = 2000;
34   if (esc_2 > 2000) esc_2 = 2000;
35   if (esc_3 > 2000) esc_3 = 2000;
36   if (esc_4 > 2000) esc_4 = 2000;
37 }
```

Figure 44: Algorithme Move Forward/Backward

Ce code contrôle l'inclinaison du drone vers l'avant ou l'arrière (Pitch Control). Il commence par lire la position du joystick (canal 2 du récepteur).

- Si le joystick est poussé vers l'avant (valeur > 1508), le drone s'incline vers l'avant.
- Si le joystick est tiré vers l'arrière (valeur < 1492), le drone s'incline vers l'arrière.

Ensuite, une correction automatique est appliquée pour stabiliser le drone, et la valeur d'inclinaison est divisée par 3 pour éviter une inclinaison trop forte.

Après cela, le code ajuste la vitesse des moteurs :

- Pour incliner le drone vers l'avant, il augmente la vitesse des moteurs arrière et réduit celle des moteurs avant.
- Pour incliner le drone vers l'arrière, il augmente la vitesse des moteurs avant et réduit celle des moteurs arrière.

Enfin, il s'assure que la vitesse des moteurs reste entre 1100 μ s et 2000 μ s pour un vol stable et sécurisé.

3.5. Rotate Left/Right (Yaw Control)

```
1 // Définition des paramètres PID pour le contrôle du Yaw
2 float pid_p_gain_yaw = 4.0; // Gain proportionnel
3 float pid_i_gain_yaw = 0.02; // Gain intégral
4 float pid_d_gain_yaw = 18.0; // Gain dérivé
5 int pid_max_yaw = 400; // Valeur maximale du PID
6
7 // Lecture de l'entrée du récepteur pour la commande de Yaw (canal 4)
8 int receiver_input_yaw = receiver_input_channel_4;
9
10 // Récupération de la vitesse de rotation autour de l'axe vertical depuis le gyroscope
11 float gyro_yaw_input = gyro_yaw_rate;
12
13 // Calcul du PID pour stabiliser la rotation (Yaw)
14 pid_output_yaw = calculate_pid_yaw(gyro_yaw_input, receiver_input_yaw);
15
16 // Ajustement des vitesses des moteurs pour contrôler la rotation
17 if (start == 2) { // Vérifie si les moteurs sont activés
18     esc_1 = throttle - pid_output_pitch + pid_output_roll - pid_output_yaw; // Avant-droit
19     esc_2 = throttle + pid_output_pitch + pid_output_roll + pid_output_yaw; // Arrière-droit
20     esc_3 = throttle + pid_output_pitch - pid_output_roll - pid_output_yaw; // Arrière-gauche
21     esc_4 = throttle - pid_output_pitch - pid_output_roll + pid_output_yaw; // Avant-gauche
22
23     // Limitation des vitesses des moteurs pour éviter des valeurs dangereuses
24     if (esc_1 < 1100) esc_1 = 1100;
25     if (esc_2 < 1100) esc_2 = 1100;
26     if (esc_3 < 1100) esc_3 = 1100;
27     if (esc_4 < 1100) esc_4 = 1100;
28
29     if (esc_1 > 2000) esc_1 = 2000;
30     if (esc_2 > 2000) esc_2 = 2000;
31     if (esc_3 > 2000) esc_3 = 2000;
32     if (esc_4 > 2000) esc_4 = 2000;
33 }
34
```

Figure 45: Yaw Controle

Ce code contrôle la rotation du drone autour de son axe vertical (**Yaw Control**).

1. Il commence par lire la position du joystick (canal 4 du récepteur).

2. Il récupère aussi la vitesse de rotation du drone à partir du gyroscope.
3. Ensuite, il utilise un contrôleur PID pour stabiliser la rotation.

Après cela, le code ajuste la vitesse des moteurs :

- Si le drone doit tourner à **droite**, il diminue la vitesse des moteurs avant-droit et arrière-gauche.
- Si le drone doit tourner à **gauche**, il diminue la vitesse des moteurs avant-gauche et arrière-droit.

Enfin, il s'assure que la vitesse des moteurs reste dans une plage sécurisée (entre **1100 μ s et 2000 μ s**), pour éviter des valeurs trop basses ou trop élevées.

4. Algorithme de l'application

4.1. GUI

```

1 from tkinter import *
2 from tkinter.simpledialog import askstring
3 from tkinter import messagebox
4 from Class import *
5 import sys
6
7 # --- Ask for Receiver Email ---
8 receiver_email = askstring("Receiver Email", "Enter the receiver's email:")
9 if receiver_email is None:
10     sys.exit()
11 while receiver_email == "":
12     messagebox.showwarning("Missing Email", "You must enter a receiver email.")
13     receiver_email = askstring("Receiver Email", "Enter the receiver's email:")
14     if receiver_email is None:
15         sys.exit()
16
17 # --- Exit Program ---
18 def exit_program():
19     window.quit()
20
21 # --- Create GUI Window ---
22 window = Tk()
23 window.title("Fire and Human Detection")
24 window.geometry("1280x720")
25 window.resizable(False, False)
26 window.configure(bg="white")
27
28 # --- Title Label ---
29 title = Label(window, text="Fire and Human Detection",
30               font=("Arial", 30, "bold"),
31               bg="white", fg="red")
32 title.pack(pady=20)
33
34 # --- Video Frame Placeholder ---
35 video_frame = Label(window, bg="black", width=720, height=480)
36 video_frame.pack(pady=10)
37
38 # --- Left Side Buttons (Start + Screenshot) ---
39 left_frame = Frame(window, bg="white")
40 left_frame.place(x=20, y=150)
41
42 btn_start = Button(left_frame, text="Start Video", width=15, font=("Arial", 12),
43                   command=lambda: fire_and_body_detection(video_frame, receiver_email))
44 btn_start.pack(pady=10)
45
46 btn_screenshot = Button(left_frame, text="Screenshot", width=15, font=("Arial", 12), command=screenshot)
47 btn_screenshot.pack(pady=10)
48
49 # --- Bottom Center Buttons (End + Exit) ---
50 bottom_frame = Frame(window, bg="white")
51 bottom_frame.place(relx=0.5, rely=1.0, anchor='s', y=-40)
52
53 btn_end = Button(bottom_frame, text="End Video", width=15, font=("Arial", 12), command=end_video)
54 btn_end.grid(row=0, column=0, padx=20)
55
56 btn_exit = Button(bottom_frame, text="Exit Program", width=15, font=("Arial", 12), command=exit_program)
57 btn_exit.grid(row=0, column=1, padx=20)
58
59 # --- Run GUI ---
60 window.mainloop()
61

```

Figure 46: Algorithme GUI

Ce code Python crée une interface graphique avec Tkinter permettant de contrôler un système de détection de feu et de présence humaine via une caméra. Lors du lancement, il demande à l'utilisateur de saisir une adresse email, nécessaire pour l'envoi d'alertes. L'interface comprend un cadre pour l'affichage vidéo en direct, un

bouton pour démarrer la détection, un bouton pour prendre une capture d'écran, un autre pour arrêter la vidéo, et enfin un pour quitter l'application. La fenêtre est bien structurée avec un titre, un fond blanc, des dimensions fixes (1280x720), et tous les boutons sont placés de manière ergonomique. Le programme repose sur des fonctions importées d'un module externe nommé Class, qui contient probablement la logique principale de détection.

4.2. Fire and Human Detection

```
1 def fire_and_body_detection(video_frame, receiver_email):
2     global cap, streaming, video_frame_ref
3     url = "http://192.168.8.100:5000/video_feed"
4     cap = cv.VideoCapture(0)
5     if not cap.isOpened():
6         messagebox.showwarning(title='Warning', message='Failed to open the camera.')
7         return
8     model_fire = YOLO('fire.pt') # Model for fire
9     model_human = YOLO('best.pt') # Model for human
10
11     video_frame_ref = video_frame
12     streaming = True
13
14     def update():
15         global streaming
16         if not streaming:
17             return
18
19         ret, frame = cap.read()
20         if not ret:
21             return
22
23         # Run YOLO detections
24         results_fire = model_fire(frame)[0]
25         results_human = model_human(frame)[0]
26
27         annotated_frame = frame.copy()
28
29         fire_detected = False
30         human_detected = False
31
32         # Fire detections
33         for box in results_fire.bboxes:
34             cls_id = int(box.cls[0])
35             conf = float(box.conf[0])
36             x1, y1, x2, y2 = map(int, box.xyxy[0])
37             if conf > 0.5:
38                 fire_detected = True
39                 cv.rectangle(annotated_frame, (x1, y1), (x2, y2), (0, 0, 255), 2)
40                 cv.putText(annotated_frame, "FIRE", (x1, y1 - 10),
41                           cv.FONT_HERSHEY_SIMPLEX, 0.8, (0, 0, 255), 2)
42
43         # Human detections
44         for box in results_human.bboxes:
45             cls_id = int(bo
```

Figure 47: Algorithm Fire and Human Detection

La fonction `fire_and_body_detection()` permet de détecter en temps réel la présence de feu et de personnes à partir d'un flux vidéo capturé par une caméra, en utilisant deux modèles YOLOv8 distincts : l'un pour les incendies (`fire.pt`) et l'autre pour les humains (`human.pt`). Après avoir capturé chaque image, le programme exécute les deux modèles pour détecter les objets d'intérêt, trace des rectangles (rouge pour le feu, vert pour les personnes) autour des objets détectés, et affiche l'image annotée dans une interface graphique Tkinter. Si un feu ou une personne est détecté pour la première fois, un e-mail d'alerte est automatiquement envoyé à l'adresse spécifiée. L'ensemble du processus se répète en boucle, offrant une surveillance continue et intelligente.

4.3. Send E-mail

```
1 def send_email(receiver_email):
2     sender = "ssaai2031@gmail.com"
3     password = "hahp mekv ddzg rnoi"
4     subject = "Warning!"
5     body = "Something Detected"
6
7     message = f"""From: {sender}
8 To: {receiver_email}
9 Subject: {subject}
10
11 {body}
12 """
13     server = smtplib.SMTP("smtp.gmail.com", 587)
14     server.starttls()
15     try:
16         server.login(sender, password)
17         server.sendmail(sender, receiver_email, message)
18     except smtplib.SMTPAuthenticationError:
19         messagebox.showwarning(title="Warning", message="Unable to LogIn!")
```

Figure 48 : Algorithme Send E-mail

La fonction commence par définir l'adresse de l'expéditeur (`sender`) ainsi que son mot de passe d'application (`password`), le sujet du message (`subject`) et le contenu du message (`body`). Ensuite, elle compose le message complet en format texte. Pour envoyer l'e-mail, elle utilise le serveur SMTP de Gmail (`smtp.gmail.com` sur le port 587), démarre une connexion sécurisée avec `starttls()`, puis essaie de se connecter

avec les identifiants de l'expéditeur. Si la connexion est réussie, le message est envoyé à l'adresse `receiver_email`. En cas d'échec d'authentification (mot de passe incorrect ou problème de sécurité), un message d'alerte s'affiche via une boîte de dialogue (`messagebox.showwarning`). Cette fonction est typiquement utilisée pour alerter lorsqu'un feu ou une personne est détecté par le drone.

4.4. Screenshot

A screenshot of a code editor window with a dark background and light text. The code is a Python function named 'screenshot'. It starts with a comment line, followed by a function definition. The function uses 'datetime.now().strftime' to generate a timestamp, 'pyautogui.screenshot' to capture the screen, and 'save' to save the image as a PNG file with the timestamp in the filename.

```
1 # --- Screenshot ---
2 def screenshot():
3     timestamp = datetime.now().strftime('%Y%m%d_%H%M%S')
4     screenshot = pyautogui.screenshot()
5     screenshot.save(f'image_{timestamp}.png')
```

Figure 49 : Algorithme Screenshot

Cette fonction Python, **screenshot ()**, capture une capture d'écran de l'écran actuel et l'enregistre en tant que fichier image avec un nom de fichier horodaté. Il génère d'abord un horodatage unique en utilisant **datetime.now().strftime ('%Y % m % d_%H % M%S')** pour s'assurer que chaque capture d'écran a un nom différent. **pyautogui.screenshot()** prend une capture d'écran, puis la méthode **save('image_{timestamp}.png')** l'enregistre au format PNG dans le répertoire courant.

Bluetooth sans fil, ce qui en fait la solution idéale pour les conceptions connectées puissantes. [19]

Camera Raspberry Pi :

La caméra Raspberry Pi Camera Module rev1.3 est une petite caméra de 2,5 cm x 2,5 cm équipée d'un capteur capable de capturer des images en haute résolution (2592 x 1944 pixels). [22]

Moteur :

Il s'agit d'un moteur brushless A2212 spécialement conçu pour alimenter les quadricoptères (drones) et les multicopters. C'est un moteur 2200kV. Il offre des performances élevées, une super puissance et une efficacité. Ces moteurs sont parfaits pour les quadricoptères de taille moyenne avec des hélices de 8 pouces à 10 pouces. [20]

Batterie Lipo :

Une batterie lithium polymère est une batterie rechargeable utilisant un électrolyte polymère au lieu d'un électrolyte liquide. Souvent abrégée en LiPo, LIP, Li-poly ou lithium-poly, elle est légère et offre une énergie spécifique plus élevée que de nombreux autres types de batteries. [23]

Contrôleur de vitesse :

Le contrôleur de vitesse électronique (ESC) reçoit un signal PWM du récepteur ou d'un contrôleur, puis ajuste la vitesse du moteur brushless. Il est alimenté par deux fils (positif et négatif), dispose de trois fils pour réguler la vitesse et trois autres pour connecter le moteur. Essentiel aux drones, l'ESC permet un contrôle précis de la vitesse et de la direction, assurant stabilité et maniabilité en vol. Les modèles modernes offrent des fonctionnalités avancées, comme le retour automatique en cas de perte de signal radio. [11]

MPU6050 :

Le module MPU6050 est un système micro électromécanique (MEMS) intégrant un accéléromètre et un gyroscope à 3 axes. Il permet de mesurer l'accélération, la vitesse, l'orientation, le déplacement et divers autres paramètres liés au mouvement d'un système ou d'un objet. [24]

Hélice:

Une hélice, également appelée hélice à pales ou hélice à vis, est un système de propulsion permettant de générer un déplacement en propulsant le fluide environnant vers l'arrière de manière axiale. Ce mouvement est obtenu grâce à la rotation des pales fixées sur un arbre entraîné par un moteur. La différence de pression qui en résulte crée une poussée transmise des pales vers l'arbre, puis à l'appareil. [22]

Le Châssis:

Le châssis d'un drone quadri-coptère est la structure principale qui soutient ses composants essentiels, tels que les moteurs, les hélices, l'électronique de contrôle et la batterie. Conçu pour être à la fois léger et résistant, il est souvent fabriqué en plastique, fibre de carbone ou aluminium. Sa forme varie selon l'application, mais il adopte généralement une structure en "X" ou "+", avec des bras reliant les moteurs au cadre central. L'électronique de contrôle, incluant la carte de vol et les contrôleurs de vitesse, est montée au centre du châssis pour assurer stabilité et performance. [11]



Figure 51: Drone Quadri-rotor.

6. Présentation de l'application

Dans cette section, nous présentons notre application :

6.1. L'interface

Cette fenêtre est l'interface principale de l'application de détection de feu et d'humains. Elle contient un titre rouge en haut, une zone noire pour l'affichage vidéo au centre, et des boutons à gauche pour démarrer la vidéo, prendre une capture, arrêter la vidéo ou quitter le programme.

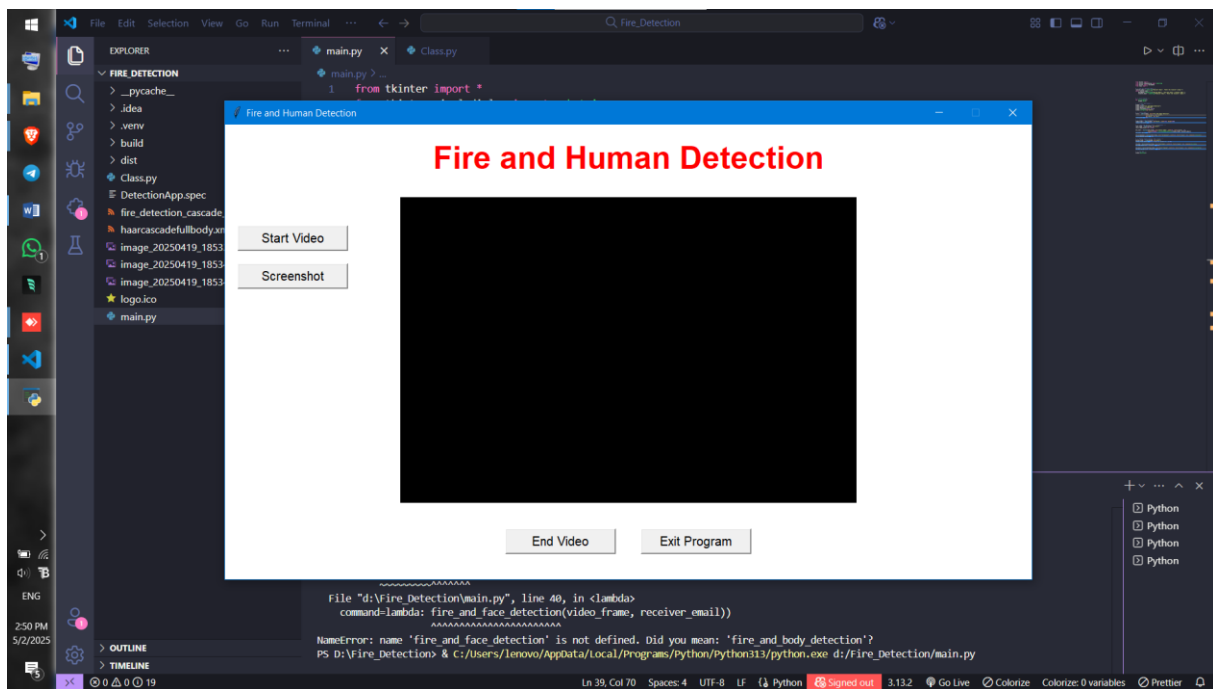


Figure 52: L'interface GUI

6.2. Screenshot

Lorsque le bouton "Screenshot" est enfoncé, une capture d'écran est prise.

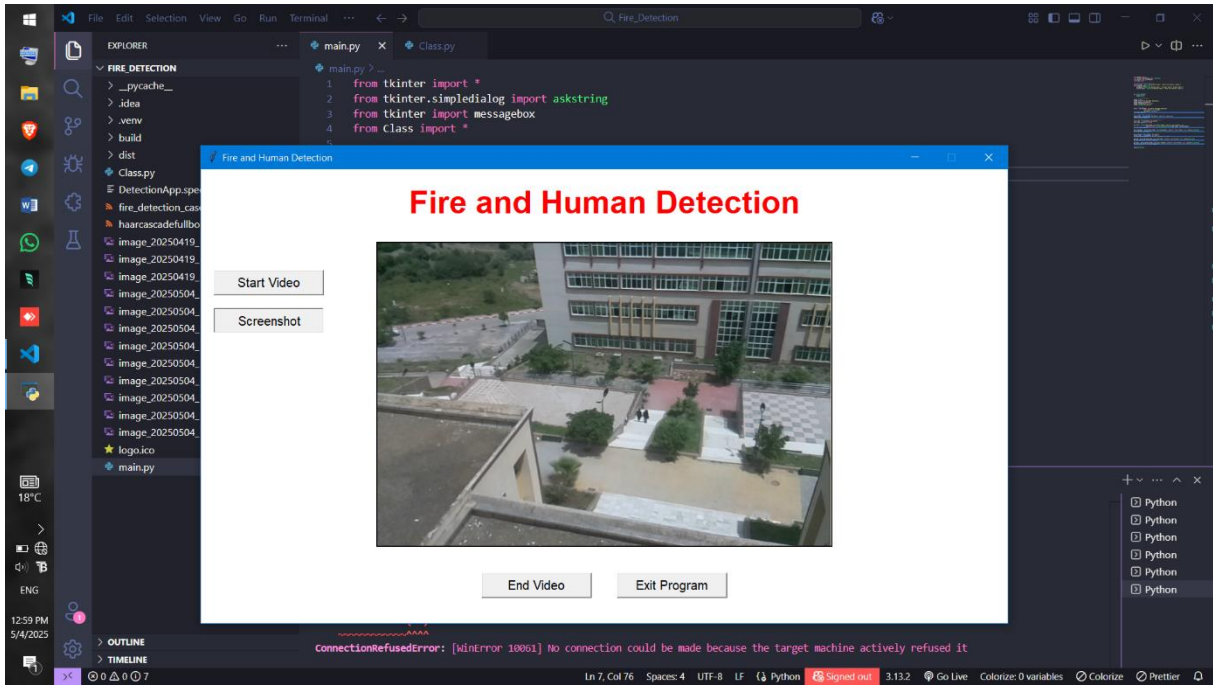


Figure 53: Take Screenshot

6.3. Exemple de détection

Lors de la détection d'un incendie ou de personnes, un cadre est dessiné.

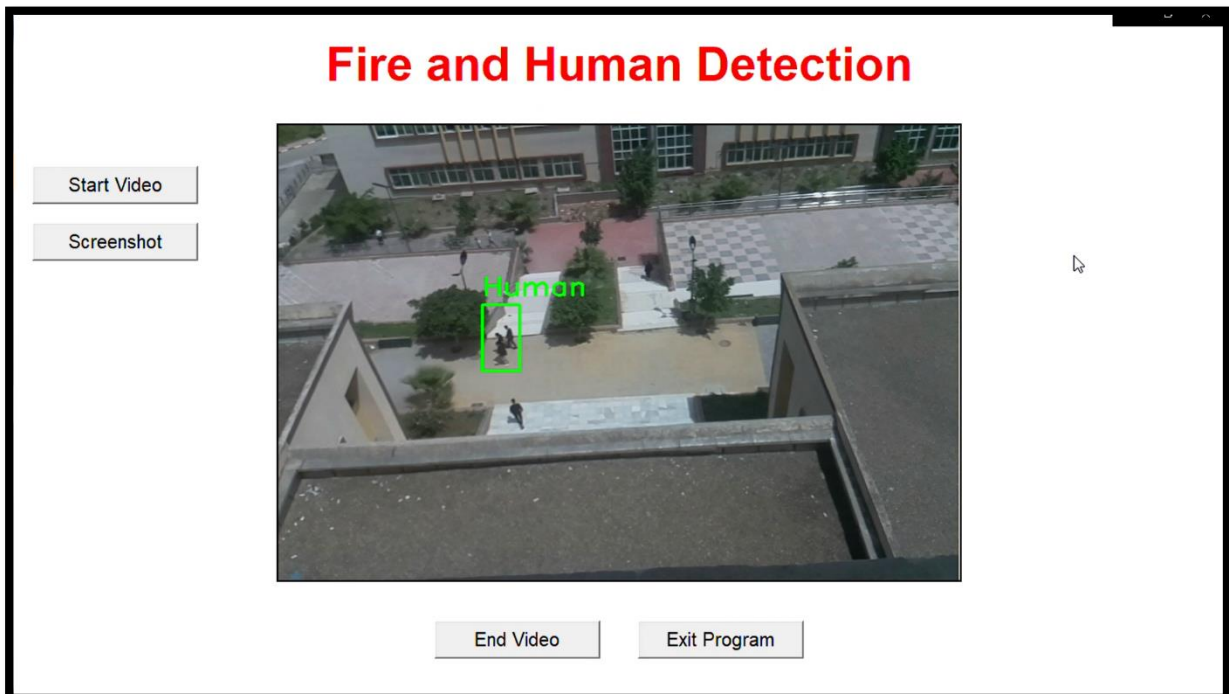


Figure 54: Fire and human detection

6.4. Envoyer email

En cas de détection d'un incendie ou de personnes, une alerte est automatiquement envoyée par e-mail.

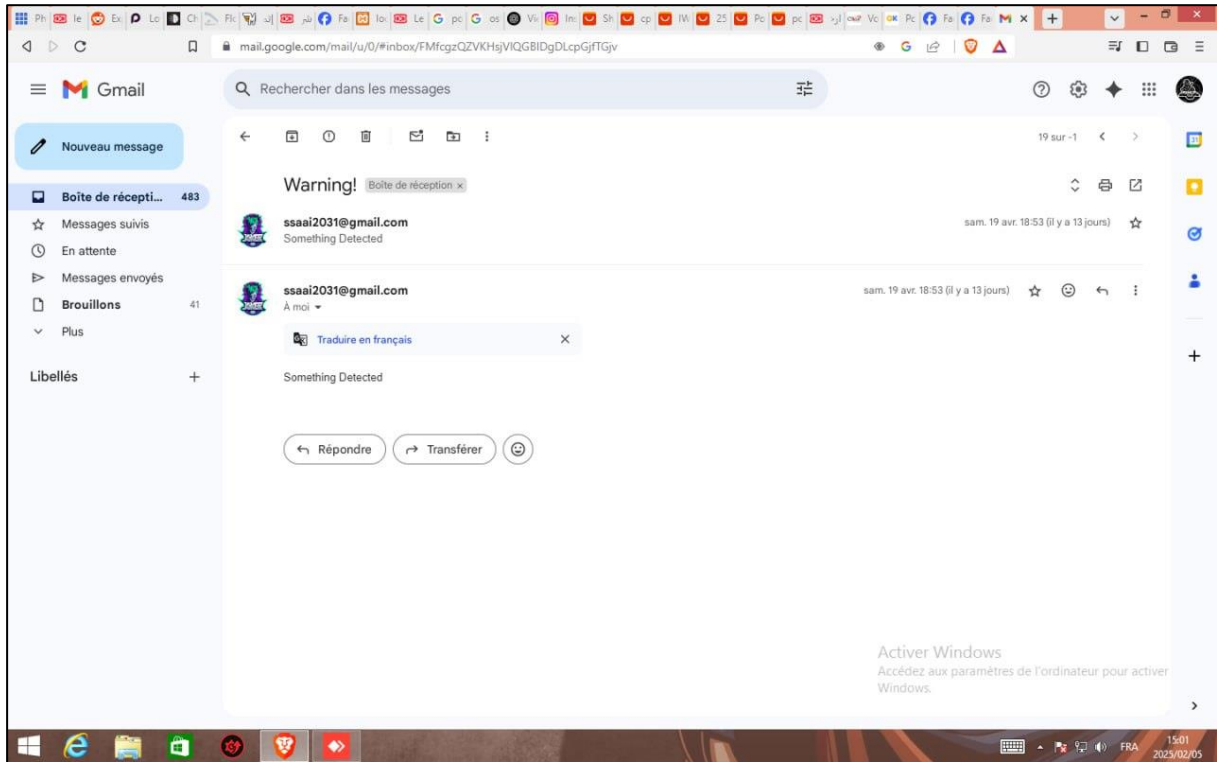


Figure 55: Send Email

7. Conclusion

Ce chapitre a détaillé la réalisation du système, en expliquant les outils de développement, les algorithmes de contrôle du drone et les fonctionnalités de détection d'incendies et de reconnaissance des personnes. Nous avons également présenté les composants matériels utilisés et leur rôle dans le bon fonctionnement du drone. Les tests réalisés ont validé l'efficacité du système, bien que des améliorations puissent être envisagées pour optimiser ses performances et son autonomie.

Chapitre5 :
BMC
(Business Model Canvas)

1. Présentation du projet

1.1. Idée du projet (la solution proposée) :

L'Algérie a connu, ces dernières années, une augmentation notable du nombre d'incendies, notamment dans les zones forestières et montagneuses. Ces incendies ont entraîné d'importantes pertes matérielles et constitué une menace directe pour la vie des habitants ainsi que pour leurs biens. Cette situation est en grande partie due au retard dans la détection des départs de feu, en particulier dans les zones difficiles d'accès. Par ailleurs, les opérations de recherche de survivants ou de personnes piégées dans de telles conditions représentent un danger considérable pour les agents de la protection civile.

Partant de ce constat, l'idée a émergé de développer un système intelligent reposant sur un drone quadrirotor, équipé d'une caméra et d'un dispositif de traitement vidéo en temps réel. Ce système vise à surveiller les zones à risque, à détecter précocement les incendies et à localiser les personnes présentes dans les zones sinistrées. Une telle solution pourrait véritablement faire la différence en accélérant les interventions, en facilitant les opérations de secours et en réduisant les pertes humaines et matérielles.

1.2. Valeurs proposées :

Les valeurs proposées par ce projet sont les suivantes :

➤ Innovation technologique :

Le projet repose sur l'utilisation de technologies modernes telles que l'intelligence artificielle et les drones pour surveiller les zones à risque d'incendie et les détecter précocement, ce qui en fait une solution innovante.

➤ Rapidité et efficacité d'intervention :

Le système contribue à réduire le temps de réponse en détectant les incendies et en envoyant des alertes en temps réel, ce qui aide à limiter la propagation du feu et à réduire les pertes humaines et matérielles.

➤ Renforcement de la sécurité des équipes de secours :

Grâce à l'imagerie aérienne et à l'analyse intelligente des données, le système permet de localiser les personnes piégées sans exposer les agents de la protection civile au danger.

➤ Flexibilité d'adaptation et d'utilisation :

Le système peut être adapté à différents environnements (forêts, montagnes, zones résidentielles) et peut également être développé à l'avenir pour inclure des missions supplémentaires telles que l'évaluation des dommages ou l'orientation des équipes de lutte contre les incendies.

➤ Facilité d'utilisation et de maintenance :

Le système a été conçu pour être facile à utiliser par les intervenants sur le terrain, avec une interface de commande simple et efficace, garantissant une mise en service rapide sans nécessiter de compétences techniques avancées.

➤ Durabilité et faible coût :

Comparé aux moyens traditionnels, ce système constitue une alternative économiquement viable à long terme grâce à l'utilisation de composants électroniques peu coûteux et réutilisables.

En somme, le projet vise à offrir une solution intelligente, pratique et efficace, alliant modernité technologique et dimension humaine, afin de réduire les effets des catastrophes naturelles et de contribuer à une meilleure gestion des crises en Algérie.

1.3. Équipe de travail :

Nous sommes deux étudiants, KENEF Aissa et BEZAZE Badr Eddine, titulaires d'une licence en informatique, spécialité ingénierie des systèmes d'information et logiciels, obtenue à l'Université du 20 Août 1955 de Skikda. Au moment du dépôt de l'idée de ce projet, nous sommes en deuxième année de master dans la même université.

En tant qu'étudiants en ingénierie, nous nous engageons sérieusement et avec détermination dans la réalisation de ce projet, visant à proposer une solution innovante et efficace pour la surveillance et la détection des incendies à l'aide d'un drone quadrirotor. Notre équipe combine des compétences techniques solides avec une

bonne capacité d'organisation et de coordination, ce qui renforce les chances de réussite du projet et l'atteinte de ses objectifs.

Nous disposons de compétences variées, notamment en programmation, en conception de systèmes intelligents, en traitement vidéo en temps réel, ainsi qu'une bonne connaissance en électronique et en pilotage de drones. Grâce aux compétences acquises durant notre parcours académique, nous sommes en mesure de développer un système intelligent et fonctionnel répondant aux besoins des organismes concernés par la gestion rapide des catastrophes.

Nous avons élaboré un plan de travail clair, avec une répartition efficace des tâches, un respect rigoureux du calendrier et une prise en compte de toutes les étapes de conception, depuis la phase théorique jusqu'à l'expérimentation du prototype. Chaque étape du projet est documentée afin d'assurer la qualité de l'exécution et la possibilité d'évolution future.

En résumé, nous sommes une équipe ambitieuse et complémentaire, travaillant avec passion et efficacité pour concrétiser un projet de véritable valeur, alliant innovation, service à la communauté et application pratique des connaissances scientifiques.

1.4. Objectifs du projet :

- ✓ Concevoir et développer un drone quadrirotor capable de voler de manière stable dans différentes conditions environnementales.
- ✓ Intégrer un système de détection des incendies en temps réel, en utilisant des techniques de vision par ordinateur et de reconnaissance automatisée.
- ✓ Localiser les personnes potentiellement présentes dans les zones à risque (par exemple les individus piégés dans les zones d'incendie), à travers l'analyse de la vidéo en direct.
- ✓ Permettre aux autorités concernées (protection civile, autorités locales) de réagir rapidement et avec précision aux situations d'urgence, afin de réduire les pertes humaines et matérielles.
- ✓ Concevoir une interface utilisateur simple et efficace, permettant d'afficher la vidéo en direct et les alertes de manière claire et accessible pour l'utilisateur final.
- ✓ Réaliser un prototype à faible coût et évolutif, pouvant être adopté comme système de soutien à la gestion des catastrophes.

- ✓ Acquérir une expérience pratique dans les domaines de la programmation, des systèmes embarqués et du traitement d'images et de vidéos, dans le cadre d'un projet concret sur le terrain.

1.5. Calendrier de réalisation du projet

Phase	Durée estimée (en mois)
Étude préliminaire du projet : analyse des besoins, recherche des solutions techniques, étude des composants	1
Achat et assemblage des composants : achat des pièces (Drone, Raspberry Pi, caméra, batterie...) et leur assemblage	2
Programmation du système et contrôle du drone : tests de vol de base et liaison avec les unités de communication	1
Intégration du système de détection des incendies et des personnes : programmation et test des algorithmes de détection via la caméra	1
Conception de l'interface utilisateur : affichage de la vidéo en direct et alertes sur PC ou smartphone	0.5
Tests en conditions réelles et calibration : essais du drone en environnement réel et ajustement des algorithmes	1

2. Les aspects innovants

2.1. Nature des innovations

Notre projet comprend plusieurs aspects innovants qui représentent la singularité de l'idée par rapport aux solutions traditionnelles utilisées dans la surveillance des incendies et la gestion des catastrophes :

- Intégration entre drone et techniques de détection :

L'une des innovations majeures de ce projet est l'intégration d'un drone équipé d'une caméra avec des algorithmes de traitement d'images pour la détection automatique des incendies et la localisation des personnes dans les zones à risque.

- Détection précoce et analyse en temps réel :

Le système est capable d'analyser la vidéo directement en vol, ce qui permet de détecter le début d'un incendie ou la présence de personnes piégées de manière instantanée, sans attendre une intervention humaine.

- Réduction des coûts par rapport aux systèmes traditionnels :

Le système est conçu avec des composants simples et peu coûteux (Raspberry Pi, caméra, etc.), ce qui le rend accessible aux administrations locales ou aux équipes de protection civile dans les zones isolées.

- Scalabilité et adaptabilité à différents environnements :

Le logiciel et les réglages du drone peuvent être modifiés selon la nature de la zone (forêts, montagnes, zones urbaines), ce qui confère une grande flexibilité au projet.

- Interface utilisateur simplifiée pour les non-spécialistes :

Une interface facile à utiliser a été développée pour permettre au personnel de secours de visionner la vidéo en direct et de recevoir les alertes sans complications techniques.

2.2. Domaines des innovations

Les innovations proposées par ce projet touchent plusieurs domaines techniques et opérationnels importants, ce qui lui confère une large capacité d'application et de développement. Parmi les domaines principaux, on peut citer :

- Domaine de la sécurité et de la prévention des catastrophes naturelles :

Le projet contribue au développement de solutions intelligentes pour la prévention et la surveillance des incendies, en particulier dans les zones sensibles telles que les forêts et les régions montagneuses, ce qui aide à réduire les pertes humaines et matérielles.

- Domaine de l'intelligence artificielle et du traitement d'images :

Le système repose sur des algorithmes d'analyse vidéo en temps réel, ce qui ouvre la voie au développement d'autres applications basées sur la vision par ordinateur pour la surveillance et l'analyse d'environnements complexes.

- Domaine des systèmes embarqués :

Le projet représente une application pratique de l'intégration de composants électroniques (comme le Raspberry Pi, la caméra, et les capteurs) dans un système intelligent autonome capable de prendre des décisions instantanées.

➤ Domaine du drone :

Le projet favorise l'utilisation des drones dans des missions humaines et vitales, telles que la recherche, le sauvetage et la surveillance des catastrophes, au-delà des simples usages récréatifs ou commerciaux.

➤ Domaine de la transformation numérique dans l'administration publique :

Grâce à la possibilité de fournir cette technologie aux municipalités ou aux équipes de protection civile, le projet peut contribuer à la numérisation des opérations de réponse aux urgences et à l'amélioration de leur efficacité.

3. Analyse stratégique du marché

3.1. Présentation du secteur de marché

3.1.1. Marché potentiel

Le marché potentiel pour notre projet inclut les organismes et institutions ayant besoin de solutions efficaces pour la surveillance des incendies et l'intervention rapide dans les zones exposées aux catastrophes, notamment les forêts et les régions montagneuses. Ce segment comprend :

- Les services de protection civile
- Les directions des forêts
- Les municipalités et conseils locaux des zones rurales
- Les compagnies d'assurance et les institutions agricoles
- Les associations environnementales et initiatives de la société civile

3.1.2. Marché ciblé

Nous définissons le marché ciblé comme un groupe spécifique d'institutions et d'organismes auxquels nous cherchons à fournir notre système, à savoir :

- Les directions de la protection civile cherchant des outils technologiques pour la détection précoce des incendies.

- Les municipalités souhaitant protéger les espaces verts et les forêts dans leur périmètre géographique.
- Les agriculteurs et propriétaires de grandes exploitations agricoles exposés aux risques d'incendie.
- Les administrations des forêts et de l'environnement assurant une surveillance continue des zones sensibles.

3.1.3. Justifications du choix de ce marché

- L'Algérie connaît chaque année des incendies dévastateurs qui se répètent souvent dans les mêmes zones, ce qui rend urgente la nécessité d'un système de surveillance efficace et proactif.
- Les outils actuellement utilisés reposent majoritairement sur l'intervention manuelle et la surveillance visuelle, ce qui est insuffisant en cas de propagation rapide des flammes.
- La prise de conscience croissante des administrations locales quant à l'importance de l'adoption des technologies intelligentes pour la protection de l'environnement et des populations.
- Offrir une solution technique locale, intelligente et à faible coût nous donne une forte opportunité de nous démarquer sur le marché national, notamment face aux alternatives importées coûteuses ou indisponibles.

3.2. Analyse de la concurrence actuelle :

Actuellement, la concurrence sur le marché local reste limitée en ce qui concerne les solutions intelligentes de détection des incendies utilisant des drones. La plupart des moyens utilisés en Algérie reposent sur la surveillance traditionnelle, ou dans le meilleur des cas, sur des caméras fixes de surveillance qui ne couvrent pas de larges zones et ne fournissent pas de réponse immédiate.

D'autre part, il existe certaines solutions étrangères (notamment européennes) qui utilisent des drones équipés de technologies d'imagerie thermique ou d'intelligence artificielle, mais elles sont généralement :

- Trop coûteuses, ce qui ne correspond pas aux budgets des administrations locales.
- Peu disponibles sur le marché algérien.

- Nécessitent une formation technique avancée pour leur fonctionnement.

Par conséquent, on peut affirmer que la concurrence actuelle est faible en termes de disponibilité de solutions locales intelligentes et intégrées. Cela offre à notre projet une forte opportunité de se positionner sur le marché, notamment grâce à :

- Un produit à faible coût.
- Facile à utiliser et adapté à l'environnement local.
- Supporté par un service après-vente et une formation simple pour les utilisateurs.

Cette situation encourage une entrée confiante sur le marché, avec une possibilité d'expansion vers des marchés similaires en Afrique du Nord ou dans les zones rurales d'autres pays.

3.3. Stratégie marketing proposée

La stratégie marketing de notre projet repose sur un ensemble de méthodes intégrées visant à atteindre efficacement la cible et à la convaincre de la valeur de la solution technologique que nous proposons.

3.3.1. Marketing direct aux institutions gouvernementales et locales

Nous adresserons nos offres directement aux administrations cibles telles que :

- Les directions de la protection civile
- Les directions des forêts
- Les municipalités

Cela se fera par l'envoi d'offres techniques et commerciales accompagnées d'un prototype pour démonstration et test.

3.3.2. Événements et salons spécialisés

Participation à :

- Salons universitaires des projets innovants
- Événements environnementaux
- Journées technologiques et conférences sur les catastrophes naturelles

Cela permettra de mettre en valeur le projet et d'attirer l'attention des partenaires potentiels.

3.3.3. Marketing digital et médiatique

- Création d'une page web et d'un dossier numérique présentant le projet
- Publication de vidéos démontrant le fonctionnement du drone sur le terrain
- Promotion via les réseaux sociaux ciblant les professionnels du secteur

3.3.4. Établissement de partenariats et soutien institutionnel

Se rapprocher des incubateurs universitaires et des institutions de soutien aux projets technologiques pour obtenir un appui financier et marketing, ainsi que pour ouvrir des canaux de communication avec les clients potentiels.

3.3.5. Services après-vente

- Garantie d'installation et maintenance initiale
- Formation simplifiée des utilisateurs finaux
- Suivi technique pour la mise à jour régulière du système

Cette stratégie nous permettra de consolider notre position sur le marché, de construire une image fiable et efficace autour de notre produit, avec la possibilité de s'étendre rapidement à de nouveaux secteurs et régions.

4. Plan de production et d'organisation

4.1. Processus de production

Le processus de production du drone se déroule à travers plusieurs étapes techniques et d'ingénierie intégrées, allant de la conception théorique à l'assemblage pratique et aux essais. On peut le résumer comme suit :

4.1.1. Conception du modèle théorique

À cette étape, nous nous basons sur un schéma technique résultant de la manière de connecter les composants électroniques du drone. Ce schéma inclut les détails des connexions des moteurs, de l'unité de contrôle de vol, de la batterie, etc.

4.1.2. Sélection et acquisition des composants

Les composants électroniques et mécaniques nécessaires sont identifiés et assemblés, notamment :

- Le châssis (Frame)

- Les moteurs brushless et les hélices (Brushless motors + propellers)
- L'unité de contrôle de vol (Flight Controller)
- La caméra
- La batterie

4.1.3. Assemblage et connexion

Les composants sont montés sur le châssis et reliés de manière sûre et organisée, en respectant les directions de fixation et le passage ordonné des câbles.

4.1.4. Programmation du système de contrôle

- Programmation de l'unité de contrôle de vol pour assurer la stabilité du drone et sa réactivité aux commandes.
- Programmation de l'unité de traitement pour exécuter les algorithmes de détection des incendies et des personnes.

4.1.5. Tests et calibrage

Réalisation de tests sur le terrain pour le vol et la détection, calibrage des capteurs et de la caméra, ainsi que l'ajustement des paramètres de contrôle selon la performance.

4.1.6. Modifications et améliorations

Sur la base des résultats des tests, des modifications sont apportées au design ou à la programmation pour assurer l'amélioration globale des performances du drone dans les environnements ciblés.

4.2. Approvisionnement des composants

Le processus d'approvisionnement repose sur une sélection précise des composants nécessaires à la réalisation du projet, en tenant compte de la qualité, du coût et de la compatibilité technique. Les composants électroniques et mécaniques ont été acquis auprès de sources fiables, que ce soit sur le marché local ou via des plateformes électroniques spécialisées.

L'approvisionnement comprend notamment :

- Moteurs brushless et hélices (Brushless Motors + Propellers) :

Sélectionnés en fonction du poids de l'appareil et de sa capacité à voler de manière stable.

- Contrôleur de vol (Flight Controller) :

Unité compatible avec un drone quadrirotor, supportant la stabilité et la réponse rapide.

- Unité de traitement (Raspberry Pi) :

Utilisée pour le traitement vidéo et l'exécution des algorithmes de détection des incendies et des personnes.

- Batterie et alimentation :

Batterie adaptée assurant une autonomie de vol suffisante, avec un système de charge sécurisé.

- Composants supplémentaires :

Câbles, modules de communication, capteurs, support de fixation pour la caméra, etc.

Chaque composant est choisi en fonction de sa fonction dans le projet, tout en respectant le plan d'approvisionnement défini et en restant dans le budget fixé.

4.3. Ressources humaines nécessaires

Nous sommes deux étudiants travaillant conjointement au développement technique et pratique de ce projet. Actuellement, nous réalisons la conception, l'assemblage et la programmation par nous-mêmes, avec une répartition des tâches selon les compétences et spécialités.

Les domaines couverts à ce stade sont :

- Programmation et traitement d'images et vidéos :

L'un de nous se concentre sur le développement des algorithmes de détection d'incendies et de reconnaissance des personnes utilisant la vision par ordinateur.

- Assemblage électronique et technique du drone :

L'autre s'occupe du montage des composants, du câblage électronique et du réglage des paramètres de vol.

Bien que nous soyons actuellement indépendants, nous prévoyons d'élargir notre équipe pour inclure :

- Des spécialistes en intelligence artificielle pour améliorer la performance et la précision de détection.
- Des experts en aviation autonome pour aider au développement de drones plus avancés.
- Des conseillers juridiques et techniques pour garantir l'homologation officielle du produit et son intégration sur le marché.

Nous aspirons également à collaborer avec la protection civile, les autorités locales et les ingénieurs en environnement afin d'adapter le projet à leurs besoins, et d'élargir l'utilisation du drone à divers domaines, tels que le sauvetage et l'intervention lors de catastrophes naturelles.

4.4. Ressources humaines nécessaires

Dans le cadre de notre projet, nous misons sur l'établissement de partenariats techniques et institutionnels efficaces pour soutenir le développement du système et renforcer sa capacité à être adopté sur le terrain. Ces partenariats représentent un élément essentiel pour assurer le succès du projet et le transformer en solution concrète et applicable à grande échelle.

Sur le plan opérationnel, nous visons une collaboration avec les directions de la protection civile et les conservations des forêts, en tant que bénéficiaires directs du système. Ces partenaires peuvent fournir des retours techniques et pratiques permettant d'améliorer les performances du drone et de l'adapter à leurs besoins spécifiques.

Nous aspirons également à établir des partenariats avec des laboratoires de recherche universitaires et des établissements d'enseignement supérieur spécialisés dans l'intelligence artificielle, les drones, ou les systèmes embarqués, afin de renforcer la base scientifique du projet et d'élargir le champ de la recherche et du développement.

- De plus, nous cherchons à collaborer avec :

Les laboratoires universitaires et centres de recherche : en coopération avec des enseignants et ingénieurs dans les domaines de l'intelligence artificielle, des systèmes

embarqués et de l'électronique, afin de nous guider techniquement et d'améliorer l'efficacité du prototype.

- Des entreprises locales spécialisées dans la vente de composants électroniques ou de drones :

Pour fournir un soutien technique, assurer la maintenance, et proposer des pièces de rechange à des prix abordables.

- Partenariats institutionnels :

- Directions de la protection civile et des forêts :

En tant que principaux bénéficiaires du système, nous envisageons de tester le drone sur le terrain avec elles et d'adapter le système à leurs besoins concrets.

- Municipalités et wilayas concernées par les risques d'incendies saisonniers :

Pour expérimenter le système dans des zones ciblées et renforcer leur capacité de réaction face aux urgences.

- Incubateurs universitaires et structures de soutien à l'innovation (comme Venture Algeria ou l'ANPT) :

Pour accompagner le projet après la phase de prototype, que ce soit en matière de conseil, financement ou industrialisation et commercialisation.

À travers ces partenariats, nous espérons garantir la qualité du système, sa faisabilité sur le terrain et assurer son évolution continue pour répondre aux besoins réels et aux défis de l'environnement local.

5. Plan Financier

5.1. Coûts Estimés et Charges Financières

Un tableau contenant les coûts approximatifs du projet et les investissements nécessaires.

Équipements	Coût estimatif (en dinars algériens)
Raspberry Pi 3	30.000
Arduino UNO	5.000
Contrôleur de vol (Flight Controller)	18.000
Moteurs Brushless + Hélices	10.000

Chapitre 05 : BMC

Batterie LiPo + Chargeur	6.000
Châssis (Frame)	5.000
Câbles, etc.	3.000

Tableau montrant l'évolution prévue des revenus financiers du projet

Année	Revenu financier estimé (en dinars algériens)
Première année	200.000
Deuxième année	500.000
Troisième année	750.000
Quatrième année	1.000.000
Cinquième année	1.500.000

5.2. Chiffre d'affaires

Tableau du chiffre d'affaires prévu (activité principale)

Chiffre d'affaires prévu (DA)	Année
1.200.000	2027
1.500.000	2028
1.800.000	2029
2.200.000	2030
2.600.000	2031

Tableau du chiffre d'affaires prévu (activité secondaire)

Chiffre d'affaires prévu (DA)	Année
800.000	2027
900.000	2028
1.000.000	2029
1.100.000	2030
1.200.000	2031

Conclusion générale

En exploitant des technologies de vision artificielle et des systèmes embarqués, nous avons conçu un drone capable de détecter le feu et d'identifier la présence humaine en temps réel. L'étude et la mise en œuvre de ce projet ont permis de démontrer l'efficacité des drones quadri-rotors dans la surveillance et la gestion des incendies.

Les résultats obtenus mettent en évidence l'apport considérable des drones dans l'amélioration des dispositifs de prévention et d'intervention d'urgence.

Cependant, des perspectives d'amélioration existent, notamment en matière d'autonomie énergétique, d'optimisation des algorithmes de reconnaissance et d'extension des capacités du drone à des environnements plus complexes. Le perfectionnement de ces aspects permettrait d'accroître l'efficacité et la fiabilité du dispositif, ouvrant ainsi la voie à une adoption plus large de cette technologie dans divers secteurs, allant de la gestion des risques environnementaux à la surveillance des infrastructures sensibles.

En conclusion, ce projet illustre le potentiel des drones dans la protection des forêts et des populations face aux incendies. Il pose les bases d'une solution technologique innovante et modulable qui pourrait être enrichie par des avancées futures en intelligence artificielle et en robotique autonome.

Bibliographie

- [1] : <https://www.ouformer.com/pages-guide/quest-ce-quun-systeme-embarque>
- [2] : <https://alovps.com/dfinition-et-rles-de-la-carte-lectronique/>
- [3] : <https://www.ceraelec.com/cartes-electroniques-le-guide-complet/>
- [4] : <https://alovps.com/dfinition-et-rles-de-la-carte-lectronique/>
- [5] : <https://www.gotronic.fr/pj2-34976-1794.pdf>
- [6] : <https://www.scribd.com/document/589999262/History-of-Arduino>
- [7] : <https://datasheet.octopart.com/A000066-Arduino-datasheet-38879526.pdf>
- [8] : <https://www.spiceworks.com/tech/networking/articles/what-is-raspberry-pi/>
- [9] : <https://pronflette.developpez.com/articles/decouverte-raspberry/>
- [10] : Halfacree, G., & Upton, E. (2016). Raspberry Pi Uber Guide. Wiley.
- [11] : SELLAOUI Dhiya Eddine "Etude et réalisation d'un drone quadrirotor", Université 8 Mai 1945 – Guelma, 2022/2023
- [12] : <https://reference-drone.fr/actualites-du-drone/histoire-du-drone/>
- [13] : https://www.researchgate.net/publication/372292802_Drone_Technologies_and_Applications
- [14] : Amrani Mohamed, Ounissi Nasreddine "Etude et réalisation d'un drone quadrirotor", Université 8 Mai 1945 – Guelma, Septembre 2020
- [15] : KHEBBACHE Hicham, " Application: Système UAV de type Quadrirotor" ,Université FERHAT ABBAS DE SETIF, 06 / 06 / 2012
- [16]: DELLANI CHAIMAÂ & DJILI SOMIA, " Investigation et tests des moteurs et des hélices pour les applications aéronautiques" ,Université SAAD DAHLEB-BLIDA, 2019/2020

[17] : Hamdani samir, " Suivi de trajectoire d'un drone type quadrirotor approche par modèles flous Takagi-Sugeno", Université MOULOUD MAMMARI DE TIZI-OUZOU.

[18] : http://www.techmania.fr/arduino/Decouverte_arduino.pdf

[19] : <https://us.rs-online.com/m/d/4252b1ecd92888dbb9d8a39b536e7bf2.pdf>

[20] : <https://www.micro-planet.ma/produit/a2212-6t-2200kv-moteur-brushless/>

[21] : <https://docs.ultralytics.com/fr/>

[22] : Latreche Bouftata Oussama, Boulkenafet Ahmed " Conception et réalisation d'une application de reconnaissance faciale par un Drone Quadri-rotors", Université 20 Aout 1955 – Skikda, 2019/2020

[23] : <https://www.techtarget.com/whatis/definition/lithium-polymer-battery-LiPo>

[24] : <https://components101.com/sensors/mpu6050-module>

[25] : https://www.intelligence-artificielle-school.com/langage_programmation/python/

[26] : https://kongakura.fr/article/OpenCV_Python_Tutoriel

[27] : <https://datascientest.com/pycharm-tout-savoir>

[28] : <https://www.positron-libre.com/electronique/arduino/arduino.php>