

Rapport - Robot de surveillance Arduino

CUVELIER Mathieu

LAMOUR Loïc

Table des matières

I Introduction	1
I.1 Motivations	1
I.2 Objectifs et cahier des charges	1
II Présentation du projet	1
II.1 Matériel	1
II.2 Schéma de montage	2
II.3 Algorithme de fonctionnement	2
III Réalisation	3
III.1 Planning (initial et réel)	3
III.2 Estimation du coût	4
III.3 Difficultés rencontrées	5
IV Conclusion et perspectives	5
V Bibliographie	5

I Introduction

I.1 Motivations

Dans le cadre de notre projet Arduino, nous souhaitons mettre au point un objet électronique utile et résolvant une problématique réelle. Nous avons ainsi choisi de réaliser un robot de surveillance permettant de surveiller les personnes présentes dans une certaine zone.

I.2 Objectifs et cahier des charges

Les objectifs principaux pour le robot étaient de surveiller une zone dans laquelle il se trouve, de déclencher des alertes s'il détecte une personne inconnue et de transmettre toutes ces informations sur une interface distante.

Concernant le cahier des charges, voici les principales contraintes.

- [Suivi et détection de personnes](#)

Le robot doit être capable de détecter toutes les personnes présentes dans une zone, et pouvoir les suivre si besoin.

- [Identification des personnes](#)

Le robot doit pouvoir identifier toutes les personnes qu'il a détectées parmi une base de données et déterminer si elles sont autorisées à se trouver dans la zone.

- [Déplacement sur deux axes](#)

Le robot doit pouvoir se déplacer sur deux axes afin de se déplacer librement dans une zone quelconque.

- [Balayage d'une zone donnée](#)

Le robot doit balayer avec la caméra toute la zone donnée afin de pouvoir détecter toutes les personnes présentes.

- [Caméra avec rotation sur deux axes](#)

La caméra doit pouvoir tourner horizontalement et verticalement pour permettre une détection la plus précise possible.

- [Notifications des alertes sur une interface dédiée](#)

Le robot doit pouvoir envoyer toutes les alertes, pour qu'elles s'affichent ensuite sur une interface dédiée avec un design propre et original.

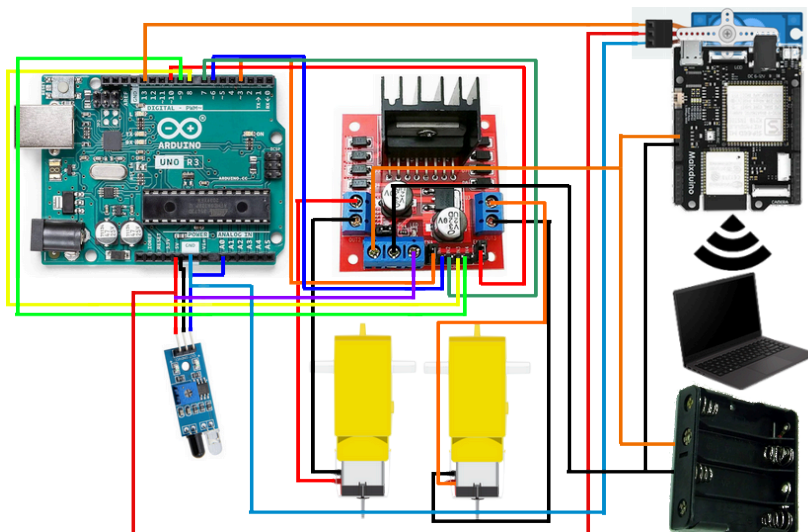
II Présentation du projet

II.1 Matériel

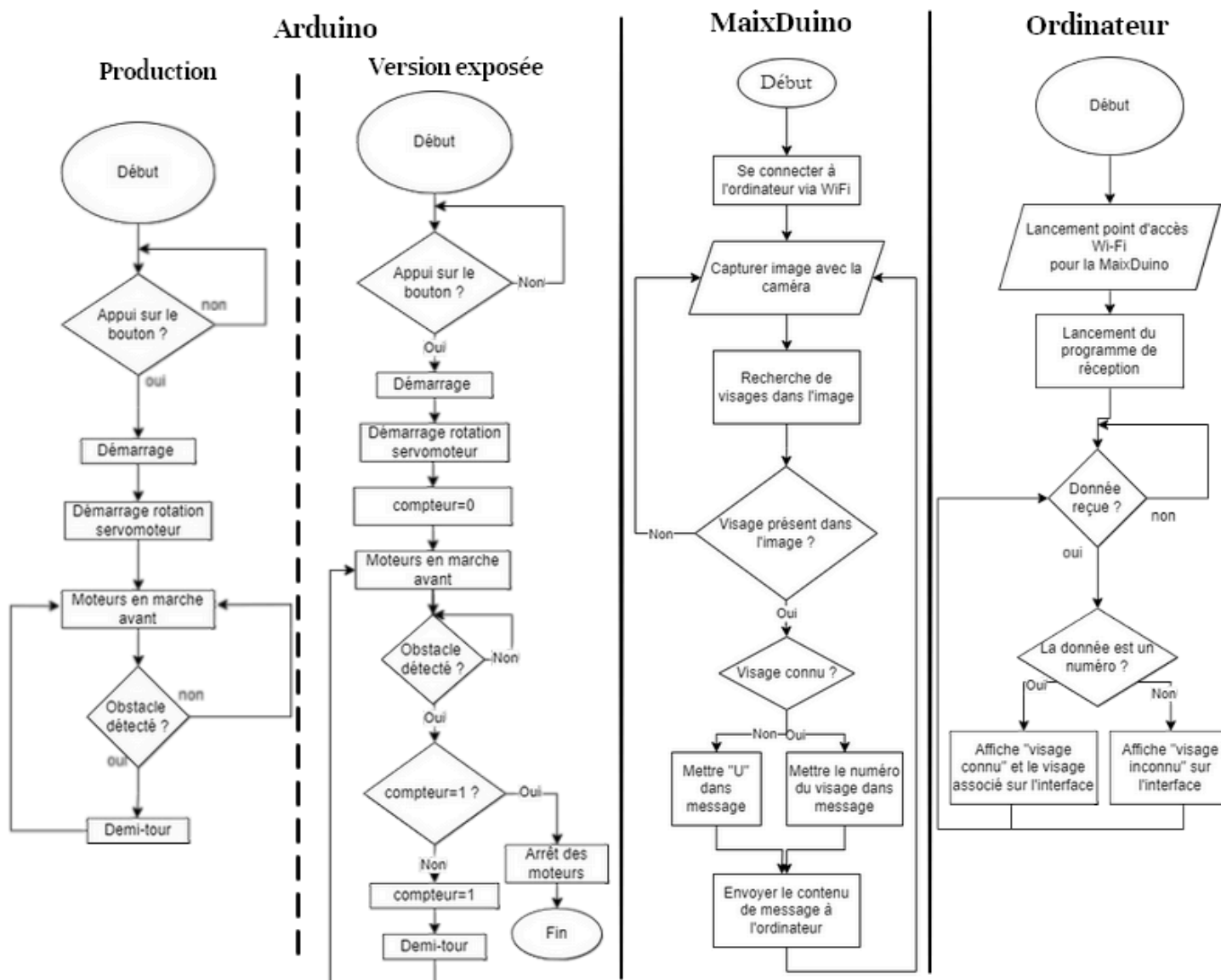
Concernant le matériel, nous avons utilisé

- Une carte Arduino qui gère toute la partie mécanique
- Des moteurs de roue
- Une carte MaixDuino qui gère la partie capture et identification de visages
- Un servomoteur sur laquelle est montée la carte Arduino
- Un pont en H qui gère principalement l'alimentation des moteurs
- Un capteur de distance infrarouge pour gérer les demi-tours ainsi que les arrêts du robot quand cela est nécessaire

II.2 Schéma de montage

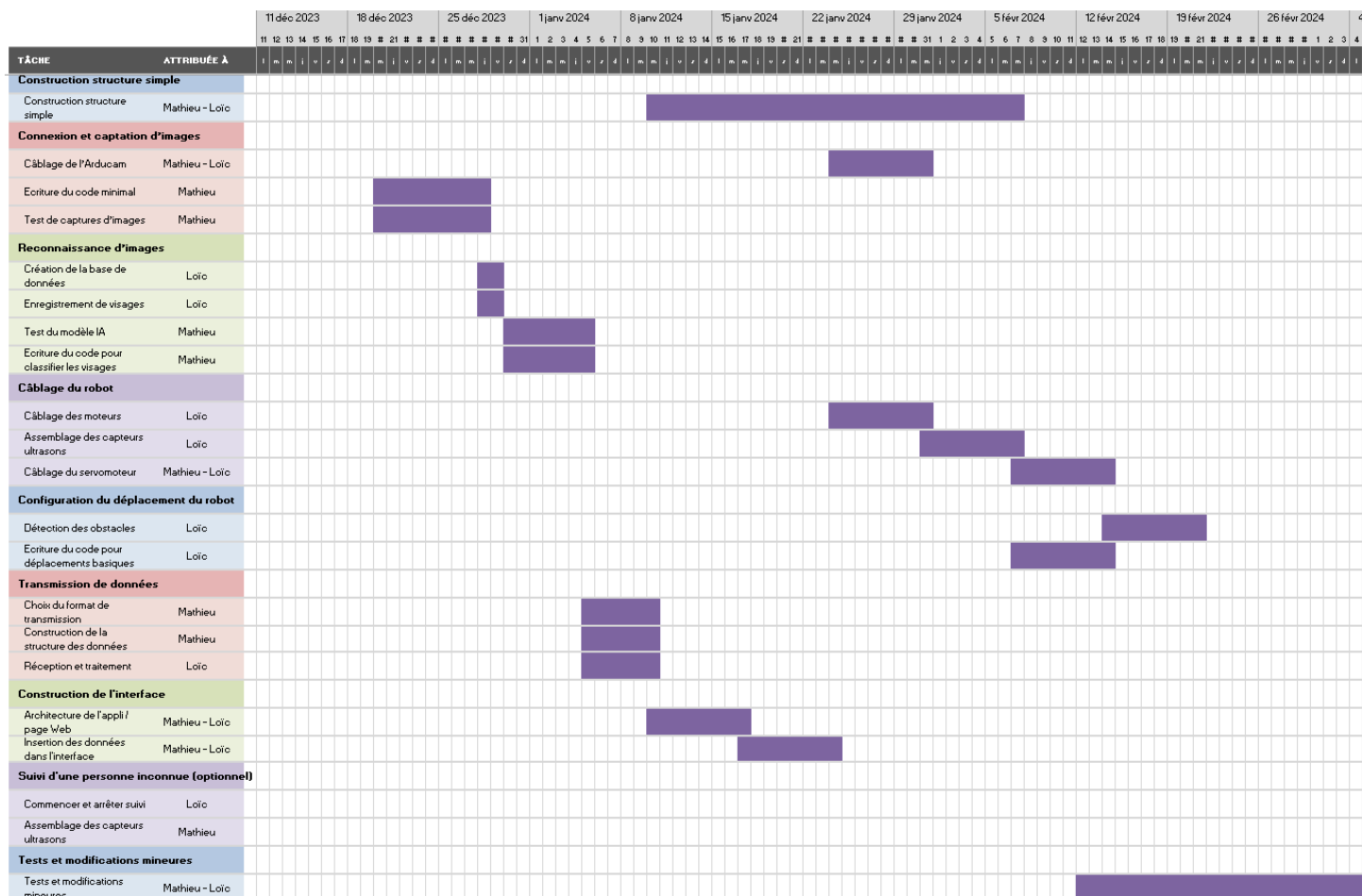


II.3 Algorithme de fonctionnement



Planning initial





Au niveau du planning, nous avons plusieurs observations : tout d'abord, la majeure partie des tâches nous a pris plus de temps que prévu ; de plus, de nombreuses tâches ont été réorganisées ; enfin, la construction est la phase dont nous avons le plus sous-estimé le temps. Le suivi a également finalement été enlevé du planning car il n'a pas été réalisé.

III.2 Estimation du coût

Item	Coût unitaire	Quantité	Coût
Temps Loïc	23,75 €	55	1 306,25 €
Temps Mathieu	23,75 €	55	1 306,25 €
Moteurs de roue	15,00 €	2	30,00 €
Servomoteur	5,00 €	1	5,00 €
Arduino	5,00 €	1	5,00 €
MaixDuino	30,00 €	1	30,00 €
Capteur IR	2,00 €	1	2,00 €
Pont en H	3,00 €	1	3,00 €
Support de piles	15,00 €	1	15,00 €
Lot de piles	10,00 €	1	10,00 €
Planches structure	0,20 €	4	0,80 €
Planche côté	0,60 €	1	0,60 €
Elements 3D	10,00 €	1	10,00 €
Entretoises, vis...	5,00 €	1	5,00 €
Total			2 729 €

III.3 Difficultés rencontrées

- Nous avons commencé par utiliser la carte Arducam, mais il y avait de nombreux problèmes, notamment une bande passante très faible et un modèle IA de détection qui ne correspondait pas exactement à nos besoins. Nous avons donc changé pour MaixDuino.
- Pour pouvoir transmettre les données sans fil entre le robot et l'ordinateur, nous avons prévu au début d'utiliser un module WiFi Arduino mais après quelques tests, il y avait trop de WiFi environnants (notamment Eduroam, Unice-Hotspot et des points d'accès WiFi) par rapport à la puissance du module WiFi.
- Concernant la MaixDuino, après avoir essayé d'envoyer un flux vidéo permanent vers l'ordinateur, nous nous sommes rendus compte que la carte mettait énormément de temps à envoyer la vidéo et figeait donc quasiment en permanence ; nous avons donc commencé à envoyer des images uniquement en cas de détection mais même cela était problématique. Nous avons donc fini par prendre la décision d'envoyer une information texte (« U » pour unknown ou le numéro du visage dans la base de données).
- En testant les premiers moteurs, il y avait une différence de vitesse de rotation entre les deux pour la même tension appliquée. Nous avons donc changé un des deux moteurs avec succès.
- Au début nous avons essayé d'alimenter la carte Arduino grâce au pont en H mais cela n'a pas été concluant. En se penchant un peu plus précisément sur la documentation du pont en H, nous nous sommes rendus compte qu'il fallait ajouter un jumper pour activer la sortie +5V du pont en H et ainsi pouvoir alimenter la carte Arduino.

IV Conclusion et perspectives

Par rapport à notre cahier des charges quasiment tous les objectifs fixés sont atteints.

- La fonctionnalité de suivi n'a pas été implémentée car celle-ci nécessite un retour image en temps réel. Or ceci implique d'avoir une bande passante suffisante. Seulement le module Wi-Fi de la carte MaixDuino ne possède pas cette caractéristique.
Nous avons donc décidé de garder la carte MaixDuino et ainsi d'abandonner cette fonctionnalité.
Une perspective d'amélioration serait de changer de carte (MaixDuino) ou l'utilisation d'un routeur plus puissant.
- Au niveau de la détection des visages, nous avons obtenu de nombreux faux négatifs, et les visages n'étaient pas toujours repérés sur l'image (faible précision / assez aléatoire).
Une perspective d'amélioration sur ce point serait un changement de caméra ou de carte MaixDuino.
- Pour les déplacements, notre version actuelle de notre robot possède des déplacements assez limités, à savoir en ligne droite avec de la détection d'obstacles entraînant le demi-tour du robot puis le retour en sens inverse. La perspective d'amélioration de ce point serait l'utilisation de capteur lidar : en effet, ces capteurs permettent de mesurer des distances et des mouvements précis en temps réel, ce qui permettrait au robot d'avoir des déplacements beaucoup plus évolués.

V Bibliographie

Lien	Catégorie	Description
https://arduino.blaisepascal.fr/pont-en-h-l298n/	Arduino	Fonctionnement et câblage Pont en H L298N
https://passionelectronique.fr/tutoriel-l298n/		
https://ledisrupteurdimensionnel.com/arduino/controler-un-servomoteur-avec-une-plaque-arduino-servo-sg90/		Câblage et fonctionnement servomoteur
https://www.locoduino.org/spip.php?article16		Possibilités d'alimentation de la carte arduino
https://www.edgemicrotech.com/tutorial-1-getting-started-with-the-maixduino/	MaixDuino	Présentation et programmes simples avec la MaixDuino
https://github.com/Sharif-Smart-and-Secure-Edge-Cloud-Lab/MaixDuino		Dépôt Github pour aider au démarrage et à la configuration de la MaixDuino
https://maixduino.sipeed.com/en/cores/digital.html		Description des fonctions d'entrée/sortie (I/O) de la MaixDuino
https://github.com/andriyadi/Maix-Azure-FacialRecog		Projet Github de reconnaissance faciale

Lien	Catégorie	Description
https://flask-socketio.readthedocs.io/en/latest/api.html#flask_socketio.SocketIO.test_client	Ordinateur	Documentation du module SocketIO de Flask
https://www.geeksforgeeks.org/how-to-use-web-forms-in-a-flask-application/		Utilisation des formulaires avec Flask
https://iximiuz.com/en/posts/writing-web-server-in-python-sockets/		Création d'un serveur web avec socket
https://docs.python.org/fr/3/howto/sockets.html		Programmation avec socket
https://openclassrooms.com/fr/courses/4425066-concevez-un-site-avec-flask/4526123-decoupez-votre-projet-en-templates		Utilisation des templates Flask