

RAPPORT DE PROJET DE FIN D'ÉTUDES



ETUDE D'UN ROBOT DE SURVEILLANCE AUTONOME

Evaluation des fonctionnalités du robot et intégration de nouveaux
éléments

TBC-France

Stagiaire

ZACHARIE EL ABDALAOUI

Encadrant Ecole

BENOIT ZERR

Encadrant Entreprise

RENATO CUDICIO

Résumé

L'entreprise TBC-France est une nouvelle branche du groupe TBC-World. Cette filiale souhaite s'implanter sur le marché de la sécurité et de la vidéosurveillance en France grâce à son robot de surveillance terrestre. Durant ce projet, le robot a été étudié dans le but de comprendre son fonctionnement et de déterminer les capacités à l'heure actuelle de celui-ci. Cette étude a été menée grâce à un échange avec le fabricant du robot (SMP Robotics) et grâce à une phase de tests nous permettant de valider ou non les informations données par le fabricant. Dans un second temps, à la lumière des éléments découverts, il a fallu décider des composants à garder et à remplacer sur le robot. Et déterminer les composants additionnels à rajouter sur ce robot en fonction des exigences européennes en matière de certification, des demandes clients, et des contraintes temporelles et structurelles liées à l'entreprise.

Table des matières

1	Introduction	1
1.1	Présentation de l'entreprise	1
1.2	Description du robot	1
1.2.1	Présentation du robot	1
1.2.2	Caractéristiques globales	2
2	Fonctionnement du robot	4
2.1	Interface utilisateur	4
2.2	Module Inertiel	4
2.3	Détection de points d'intérêts	4
2.4	Correction GPS	5
2.5	Localisation RFID	6
2.6	Détection d'obstacles	6
2.7	Vidéosurveillance	8
2.8	Récapitulatif	9
3	Phase de test	11
3.1	Prise en main des robots	11
3.1.0.1	Win Navigator	11
3.2	Système de communication	13
3.2.1	Configuration réseau initiale	13
3.2.2	Configuration réseau finale	14
3.2.3	Intégration de la base GPS	15
3.2.3.1	Première configuration de la base GPS	15
3.2.3.2	Connexion entre la base GPS et le robot	16
3.3	Tests de précision	17
3.3.1	Protocole de test	18
3.3.2	Localisation inertielle seule	18
3.3.3	Test de la détection de points d'intérêts	19
3.3.4	Tests de navigation RFID	19
3.3.4.1	Premier test	20
3.3.4.2	Deuxième test	21
3.3.4.3	Troisième test	24
3.4	Corrections GPS	25
3.5	Détection d'obstacles	25
3.5.1	Module stéréoscopique et radars de recul	25
3.5.2	Cartographie du terrain	25

3.6	Vidéosurveillance	26
3.6.1	Détection de mouvement	26
4	Conclusions	27
4.0.1	Analyse des résultats de tests	27
4.0.2	Décisions concernant les appareils liés au robot	27
4.0.3	Problèmes rencontrés	28
4.1	Remerciements	28
A	Journal de bord du stage	30
B	Caractéristiques du robot	49
C	Configuration de la base GPS	52
D	Rapports de tests	67

Table des figures

1.1	Taille du robot par rapport à une personne	2
1.2	Schéma global du robot	3
2.1	Serveur Web de la base GPS	5
2.2	Vue du Logiciel StéréoViewer (Images des deux caméras en haut, carte de disparité en bas à gauche)	7
2.3	Vue du dispositif de vidéosurveillance démonté	8
2.4	Visualisation du flux vidéo	9
2.5	Visualisation du flux vidéo	10
3.1	Menu principal de WinNavigator	11
3.2	Menu principal de WinNavigator avec carte	12
3.3	Menu principal de WinNavigator avec carte	13
3.4	Schéma de l'architecture réseau de l'interface Base GPS/Utilisateur/Robot	14
3.5	Interface Multivision (En rouge la fonction find pour trouver l'adresse IP de la base GPS)	16
3.6	Parcours du robot	18
3.7	Résultats du test de la localisation visuelle (R :Portes Ratées, D : Portes passées après modification de la trajectoire, P :Porte)	19
3.8	Parcours n°1 du Robot avec RFID (Bornes RFID en Orange, Périmètre défini en bleu)	20
3.9	Résultat du 1er test avec module RFID (C :Collision avec un cône, R :Porte ratée, D :Recalcul de la trajectoire, H :Hésitation sans modification de trajectoire)	21
3.10	Parcours n°2 du Robot avec RFID (Bornes RFID blanc, Périmètre défini en bleu, Chemin indiqué par les points oranges, et portes en rouge)	22
3.11	Résultat de la première partie du test (C :Collision avec un cône, X :Porte ratée, R :Recalcul de la trajectoire, H :Hésitation sans modification de trajectoire)	23
3.12	Résultat de la deuxième partie du test (C :Collision avec un cône, X :Porte ratée, R :Recalcul de la trajectoire, H :Hésitation sans modification de trajectoire)	23
3.13	Résultat de la deuxième partie du test (C :Collision avec un cône, X :Porte ratée, R :Recalcul de la trajectoire, H :Hésitation sans modification de trajectoire)	24
3.14	Classification des différents éléments dans le champ de vision du robot à l'atelier (Blanc : Mur, Bleu : Sol, Noir : matériau entre le mur et le sol, Vert : Zone d'ombre, Violet : zone de surbrillance)	26

Chapitre 1

Introduction

1.1 Présentation de l'entreprise

TBC-France est une filiale du groupe TBC-World, un groupe franco-canadien de commerce international agé de plus de 150 ans, présent à l'international (Afrique, Inde, Amérique du sud). Ce groupe propose des services dans des domaines allant de la communication digitale, à l'intégration de nouvelles technologies, et également du consulting pour les entreprises.

TBC-France est la dernière filiale de ce groupe, et veut s'implanter sur le marché européen de la sécurité et de la vidéosurveillance en partant de la France et de la Belgique. Le robot de surveillance dont il est question dans ce rapport est le produit phare de l'entreprise pour se lancer. Il est à noter que TBC-France, n'est pas le fabricant des dits robots. Ces robots ont été développés à Moscou, et sont fabriqués par SMP Robotics, une entreprise basée en Californie. Il est important de noter que TBC-France est le distributeur des robots en France et en Belgique.

1.2 Description du robot

1.2.1 Présentation du robot

Les robots distribués par TBC-France, sont des robots de surveillance destinés à compléter l'action d'un agent de sécurité humain sur des sites industriels isolés. Cette solution présente un intérêt pécunier non négligeable, car elle permet de limiter les coûts d'envoi d'agents de sécurité pour faire des levées de doute, notamment pour des cas de fausses alertes. En revanche ce robot n'est en aucun cas un robot doté de capacités offensives, il ne possède pas d'équipement approprié pour maîtriser un intrus de manière autonome. TBC-France a opté pour un modèle économique de location des robots avec prise en charge de la maintenance. En effet, pour l'intégrité de la machine, il est plus sûr que nous prenions en charge le suivi que de vendre le robot et de laisser tous les éléments d'entretien au client.

Le robot étudié dans cette première partie du projet est la version 5.0 du produit. C'est un prototype du robot qui sera certifié et commercialisé par la suite. Le coeur du projet à été d'étudier les fonctionnalités du produit actuel, pour définir les améliorations à apporter par le fabricant par la suite.

1.2.2 Caractéristiques globales

L'UGV (Unmanned Ground Vehicle) a une dimension de 136.9x765x155 cm (155 cm pour la tour) pour un poids de 110Kg. La majorité du poids du robot est liée aux batteries utilisées pour l'alimenter.



FIGURE 1.1 – Taille du robot par rapport à une personne

Deux batteries Acide-Plomb faisant 32Kg chacune, utilisées au profit de batteries Li-ion ayant un rapport poids/performances plus élevé, mais dont le prix n'est pas compétitif.

Le drone possède deux modes de fonctionnement, un mode manuel, contrôlé par l'utilisateur, soit par l'intermédiaire d'une manette connectée par radio, soit grâce à un appareil connecté via Bluetooth.

Et un mode autonome que l'utilisateur peut gérer depuis le PC de commande. Ce mode autonome est muni de 4 modules de localisation qui en fonctionnant simultanément rendent le robot plus précis.

La vitesse maximale du robot dépend du mode dans lequel on l'utilise. En mode manuel, le robot peut aller jusqu'à 8 Km/h, et en mode autonome, la vitesse de croisière est réglée à 2.5Km/h pour permettre au robot de traiter les données reçues par les modules de localisation, tout en évitant de causer des accidents avec des éléments de son environnement.

Les modules de navigation sont gérés via 7 cartes de contrôles distinctes. 5 d'entre elles gèrent un module en particulier, la 6ème (Navigator) permet de centraliser, et de traiter les informations récupérées par les capteurs, et la 7ème (Autopilot) actionne le moteur du robot (sur le train arrière). La communication entre les capteurs et la carte de contrôle Navigator est assurée par un CAN bus, pour limiter le câblage présent dans le robot.

Les caractéristiques globales du robot sont présentées en annexe de manière plus détaillée. Et la figure ci-dessous permet de visualiser l'ensemble des éléments du robot.

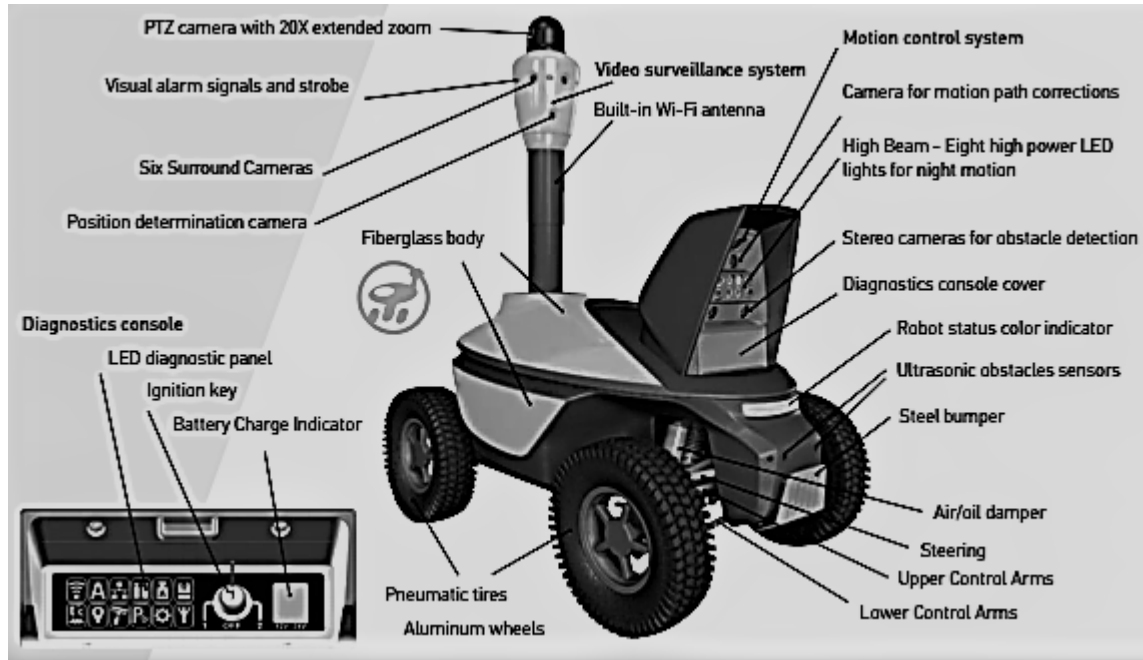


FIGURE 1.2 – Schéma global du robot

Les éléments ayant trait à la navigation et à la vidéosurveillance seront traités plus en détail dans les sous parties dédiées.

Chapitre 2

Fonctionnement du robot

2.1 Interface utilisateur

A l'heure actuelle, le robot a été fourni par le fabricant avec des applications permettant d'interagir avec le robot via un PC sous Windows. Chaque aspect de la navigation du robot possède un logiciel propre grâce auquel l'utilisateur va pouvoir régler les paramètres liés au module considéré. Toutefois le logiciel WinNavigator permet de gérer la majorité des actions du robot depuis le PC de contrôle. Dans ce logiciel l'utilisateur va pouvoir gérer l'apprentissage du parcours du robot ainsi que son comportement en mode autonome. L'apprentissage d'un parcours pour le robot sera détaillé après avoir abordé chacun des modules de localisation.

N.B : Les informations de précision des modules données par la suite sont apportées par SMP Robotics. Elles sont basées sur les capacités théoriques des éléments inclus dans les modules de localisation concernés.

2.2 Module Inertiel

Une centrale inertielle composée de gyromètres et d'accéléromètres est embarquée sur le robot. Celle-ci permet au robot de se localiser dans son repère local lorsqu'aucun autre module de localisation n'est disponible. Les données récupérées sont traitées par la suite par un filtre de Kalman pour réduire les imprécisions liées à l'environnement. Ce module permet de localiser le robot dans un rayon de 3m autour de sa position dans le pire des cas. Il apparaît donc que ce module est insuffisant par lui-même pour assurer une navigation efficace et relativement précise du robot.

2.3 Détection de points d'intérêts

Ce module utilise deux caméras embarquées sur la tour du robot lui permettant de voir devant et derrière lui. Les images récupérées subissent ensuite un algorithme de traitement d'image par régions pour détecter les variations d'intensité dans les pixels de 2 régions consécutives. Ceci permet au robot de détecter des points d'intérêts dans l'image captée. Par la suite, l'image captée au temps t est mise en relation avec l'image captée en $t-1$. En utilisant un algorithme SIFT, le robot peut déterminer les éléments restant à l'image suffisamment longtemps pour pouvoir s'y référer durant son trajet. Lorsqu'il repassera à ce point, il sera en mesure de mettre en relation l'image qu'il capte

et l'image qu'il a apprise la première fois qu'il y est passé pour vérifier sa position et la corriger le cas échéant. Selon SMP le robot est en mesure d'être précis à 20cm à l'aide de cette technologie.

2.4 Correction GPS

Ici, le robot utilise la combinaison de son GPS embarqué, et d'une Base GPS fixe reliée au réseau du robot. Cette base GPS capte la position du robot, et sa position GPS, pour ensuite envoyer des corrections en cap au robot. La communication entre la base GPS et le robot se fait par protocole IP, en réglant les deux éléments (La Base et le fichier de configuration de la navigation) le robot envoie sa position à l'adresse IP de la base. Comme précisé par la suite, cet élément a été important dans la compréhension de l'architecture du réseau à mettre en place pour commander les robots. Cela permet au robot d'avoir une précision théorique à 40cm.

La base GPS est fournie pour l'instant par SMP Robotics. Elle est composée d'une antenne GPS Tallysman TW2405 et d'un serveur vidéo Tral 5 Poe, utilisé ici pour envoyer les corrections GPS par IP au robot. Cette base est accessible par un ordinateur de plusieurs manières :

- Directement en entrant son adresse IP dans un navigateur web. Cela permet d'accéder à un serveur web pour modifier l'adresse IP de l'appareil comme montré en 2.1.
- A travers le logiciel Multivision qui permet d'accéder à tous les paramètres du Tral 5 Poe.
- SMP Robotics a également fourni une application permettant en théorie de trouver l'adresse IP de la base (Nous reviendrons sur ce point dans la partie 3.2.3.1)

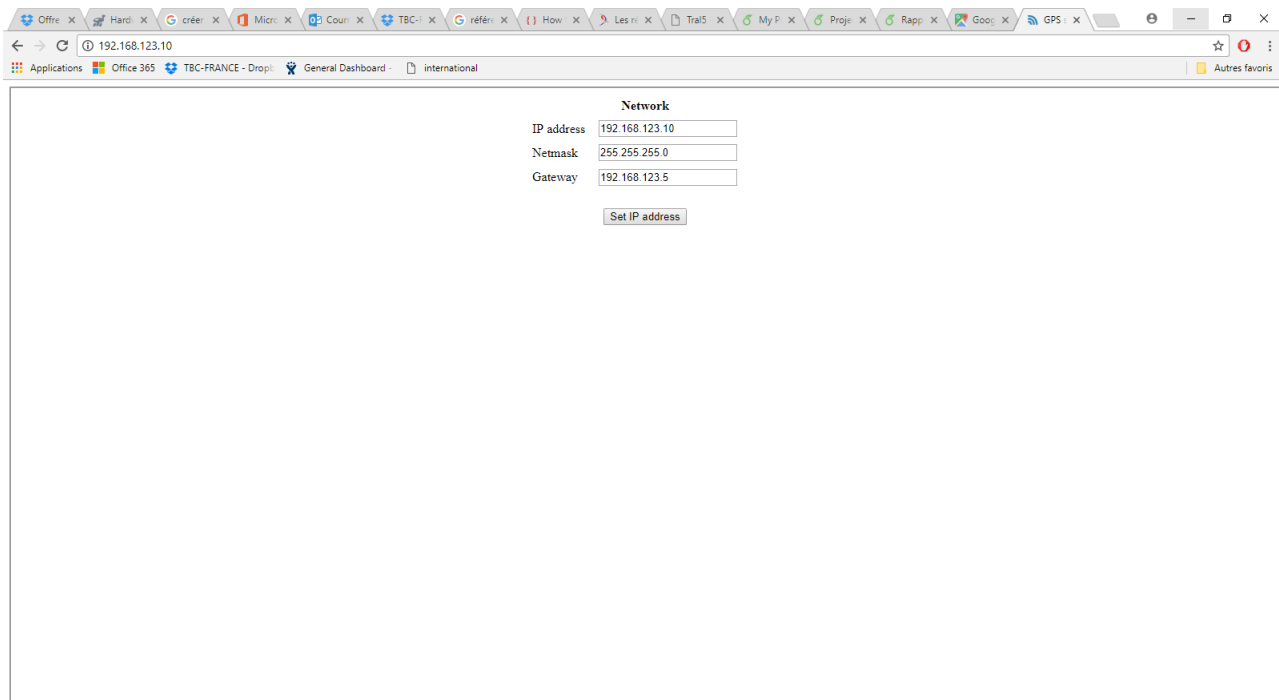


FIGURE 2.1 – Serveur Web de la base GPS

2.5 Localisation RFID

Le Robot possède 3 récepteurs RFID (Radio Frequency IDentification) embarqués, un sur la tête, et 2 sur la tour. Ces récepteurs permettent de communiquer avec des balises RFID placées dans l'environnement du robot. Le système utilise un algorithme Time of Arrival (ToA) pour se localiser par rapport aux bornes. Le robot et les balises communiquent entre eux, et la vitesse de propagation du signal est connu. En fonction du temps mis par le signal pour arriver au robot, les balises peuvent déterminer un cercle sur lequel le robot peut se trouver. Pour obtenir le positionnement en 2D du robot, il suffit alors de poser 3 balises pour trouver sa position en trouvant l'unique point d'intersection entre les 3 cercles. Ici le module de positionnement permet de localiser le robot dans un environnement en 3D. Le principe est le même sauf que l'altitude est prise en compte ici. En vue de cette donnée, les balises ne renvoient plus des cercles, mais des sphères sur laquelle le robot peut se trouver, en posant 4 balises, le robot peut se positionner en déterminant le point auquel les 4 sphères se coupent.

$$\begin{cases} (x - a_1)^2 + (y - b_1)^2 + (z - c_1)^2 = r_1 \\ (x - a_2)^2 + (y - b_2)^2 + (z - c_2)^2 = r_2 \\ (x - a_3)^2 + (y - b_3)^2 + (z - c_3)^2 = r_3 \\ (x - a_4)^2 + (y - b_4)^2 + (z - c_4)^2 = r_4 \end{cases}$$

Le module résout donc un système de 4 équations d'ordre 2 à 3 inconnues. Ce qui explique le besoin de 4 balises. Les a_i, b_i, c_i correspondent aux centres des sphères qui sont connus. En effet avec 3 équations, il y aurait 2 solutions à cette équation, et avec 4 il n'en reste qu'une.

La localisation par RFID permet en théorie au robot d'être précis à 20cm. Le module est fourni avec son application d'interaction (RadioNavigator). La manière de mettre en place le système de localisation RFID est la suivante :

- Mettre en place les balises et allumer RadioNavigator.
- Connecter le robot de façon à ce qu'il détecte les balises.
- Choisir 4 balises à ajouter à l'environnement interne du robot.
- Le cas échéant lier ces nouvelles balises à un périmètre de RFID déjà connu du robot.
- Bouger le robot tel que le logiciel le demande (Arc de 3m et de 30°).
- Modifier la position des balises sur la carte interne du robot pour coller à la réalité.

Selon SMP les balises peuvent être retirées après que le robot ait effectué son apprentissage sans affecter la précision de ce module. Ceci nous a immédiatement paru étrange et nous a amené à tester cette affirmation (cf 3.3.4.2).

2.6 Détection d'obstacles

En parallèle de tous ces modes de localisation, ce module permet de détecter les obstacles imprévus dans l'environnement du robot. Le fabricant a opté pour une solution basée sur du traitement d'images plutôt que pour un LIDAR (Light Detection And Ranging) . Selon SMP Robotics

le LIDAR serait moins compétitif en terme de coût que le traitement d'images stéréoscopique utilisé.

Deux caméras stéréoscopiques sont embarquées à l'avant du robot. Les images captées par ces deux caméras sont ensuite traitées, pour en retirer la géométrie épipolaire, et la carte de disparité. Grâce à cette carte de disparité, le robot est capable de discerner les variations d'altitude dans le terrain immédiatement devant lui, et donc de déterminer les obstacles présents dans son chemin.

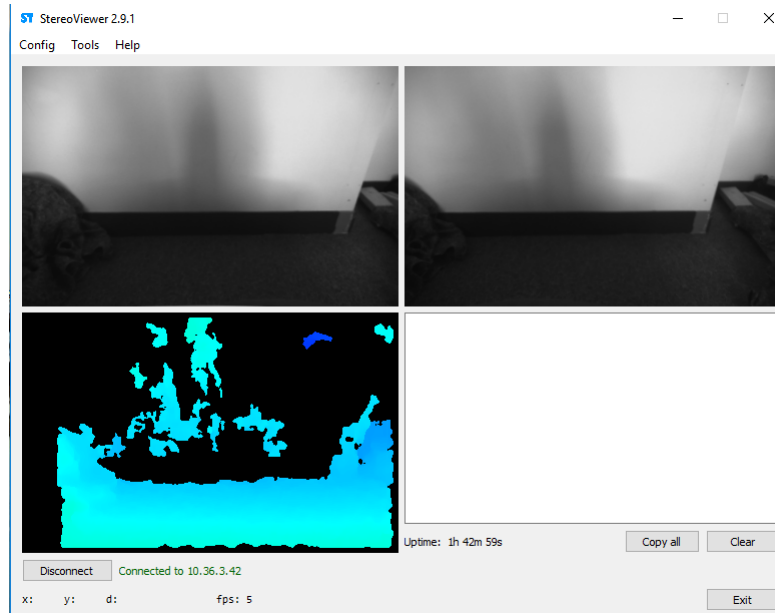


FIGURE 2.2 – Vue du Logiciel StéréoViewer (Images des deux caméras en haut, carte de disparité en bas à gauche)

Ceci est une partie de l'algorithme de cartographie du terrain qu'effectue le robot en temps réel pour analyser son environnement. En effet grâce à un programme de classification des données, il est possible de faire apprendre au robot les terrains qu'il peut considérer comme une route, et ceux qui ne doivent pas être considérés comme un chemin en théorie. Pour cela il suffit de délimiter une zone de l'image correspondant à un type de terrain, et de faire de cette zone un ensemble d'apprentissage pour le robot. Ensuite, l'utilisateur peut choisir si le terrain est censé être une route ou non. Dans ce même logiciel, il est possible de choisir la hauteur des objets qui seront considérés comme des obstacles.

Pour compléter ce dispositif, deux radars de reculs à ultrasons sont intégrés à l'arrière du robot pour que ce dernier puisse reculer sans risques le cas échéant. Ces radars permettent de détecter des obstacles présents jusqu'à 2m derrière lui.

2.7 Vidéosurveillance

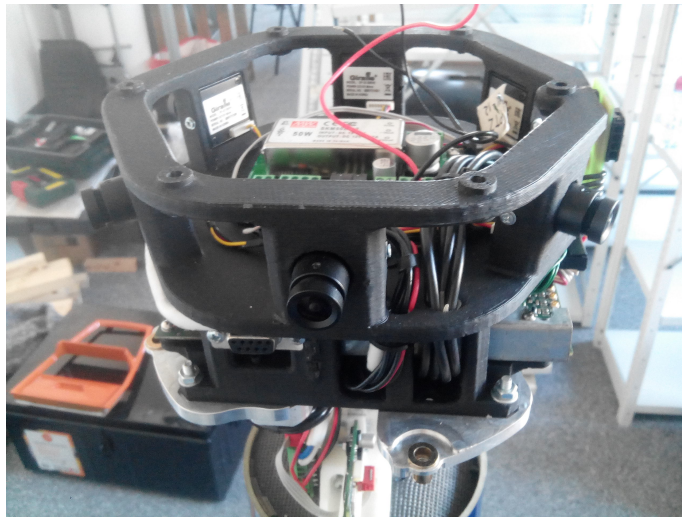


FIGURE 2.3 – Vue du dispositif de vidéosurveillance démonté

Le système de vidéosurveillance du robot est composé des éléments suivants :

- Une caméra PTZ (Pan-Tilt-Zoom) Ismart IS-VS2800.
- 6 caméras analogiques Giraffe GF-Q1325HE sont présentes tout autour de la tête du robot pour avoir une vision panoramique à 360° autour du robot.
- Un DVR (Digital Video Recorder) Tral-Patrol 7 permet l'enregistrement du flux vidéo et la re-transmission des images en direct.
- Un Disque dur est également embarqué pour stocker les enregistrements.

Un algorithme de détection de mouvement est également intégré à ce dispositif. Les caméras panoramiques détectent le mouvement et font remonter une alarme, à la fois à l'utilisateur, et à la caméra PTZ pour faire pointer cette dernière vers la zone où le mouvement a été détecté.

L'accès au flux vidéo se fait par le logiciel Multivision, en se connectant à l'adresse IP du DVR. Cela permet de visualiser l'ensemble des caméras comme on peut le constater ci dessous :

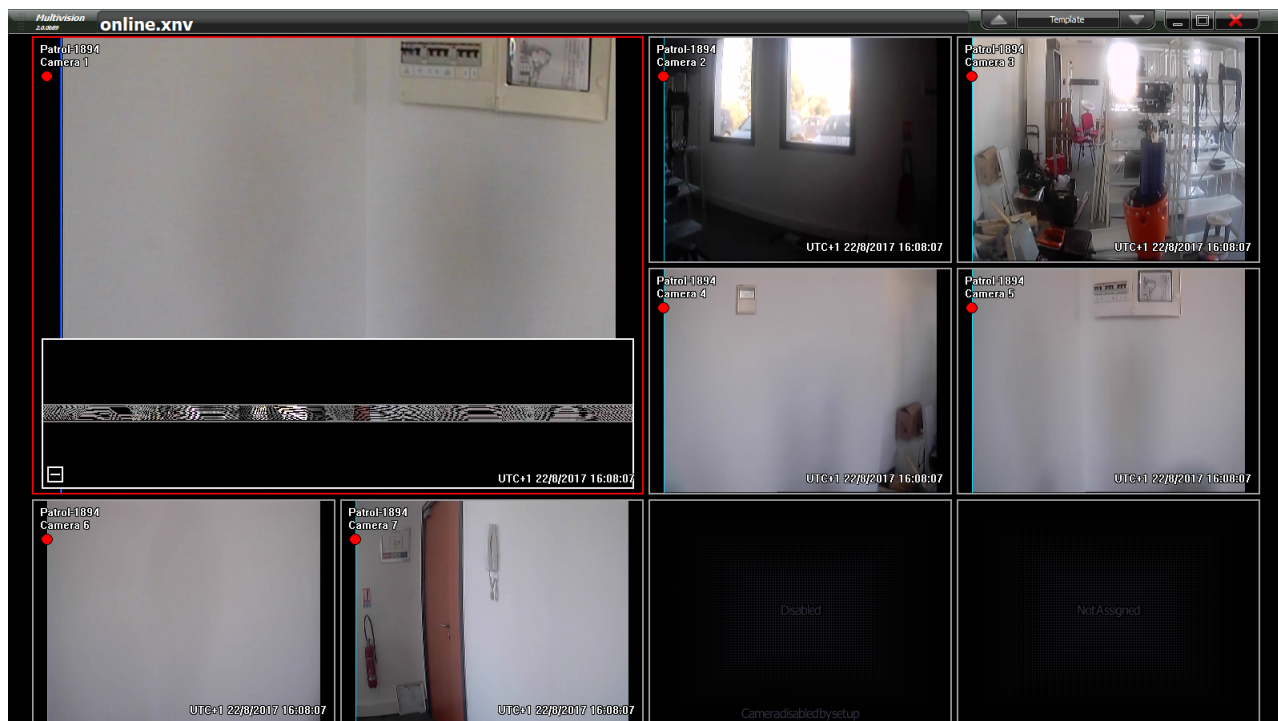


FIGURE 2.4 – Visualisation du flux vidéo

La vidéo de la caméra PTZ est affichée dans l'image en haut à gauche, avec une image du panorama observé par les six caméras du robot en dessous de celle-ci. Les autres images sont le rendu visuel de chacune des six caméras.

Cette image du panorama sur l'écran de la PTZ permet de pointer la caméra vers la direction souhaitée pendant une dizaine de secondes.

2.8 Récapitulatif

L'architecture, et la manière dont les modules s'intègrent au robot est présentée dans le schéma suivant.

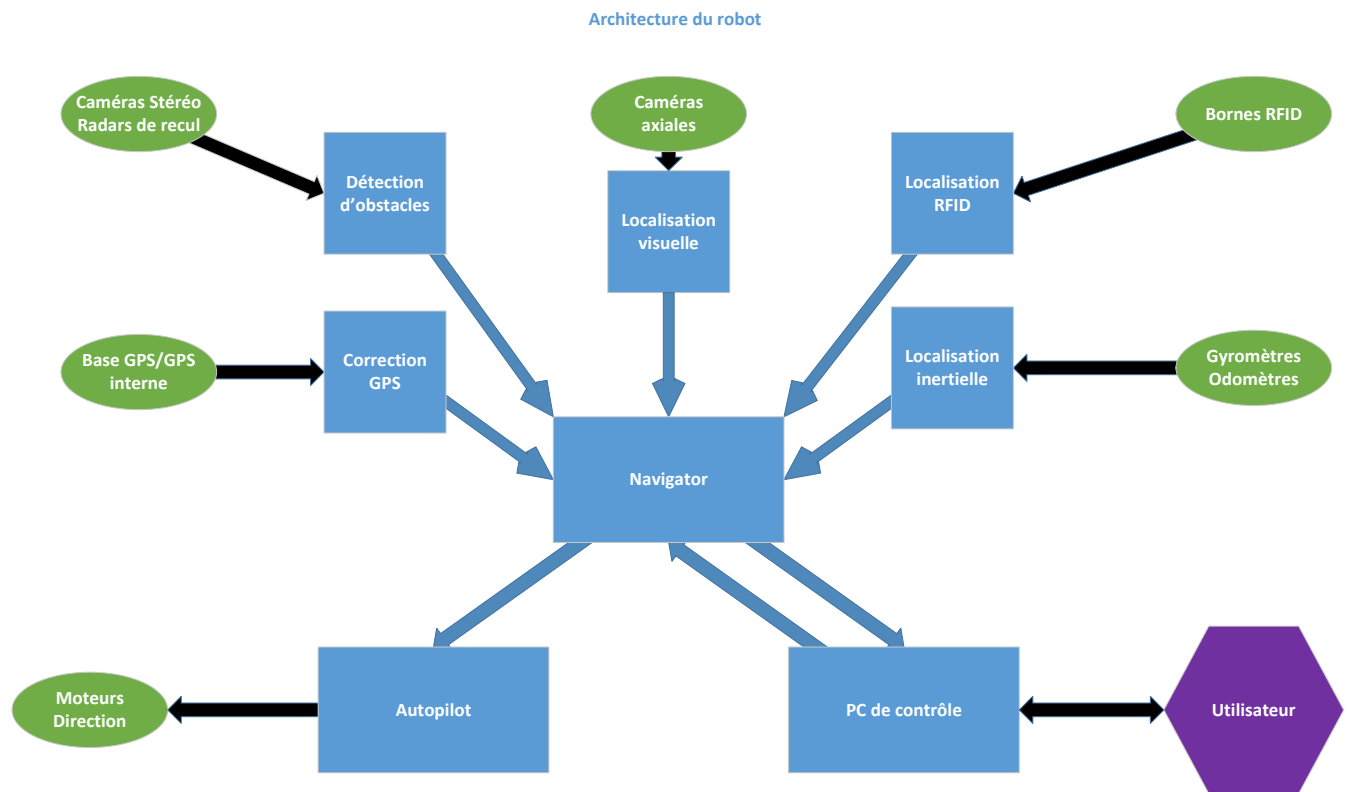


FIGURE 2.5 – Visualisation du flux vidéo

Chapitre 3

Phase de test

3.1 Prise en main des robots

Les premières semaines ont été consacrées à la prise en main des robots, et à maîtriser leur principe de fonctionnement. Il y a deux manières d'effectuer l'apprentissage d'un parcours. L'une se fait en effectuant le parcours une première fois manuellement, en contrôlant le robot à l'aide de la manette radiocommandée. L'autre se fait directement sur le PC de contrôle.

3.1.0.1 Win Navigator

Ce logiciel (mentionné en 2.1) est développé par SMP Robotics et permet de gérer la majeure partie des interactions avec le robot. la fenêtre principale s'affiche comme dans la figure ci dessous.

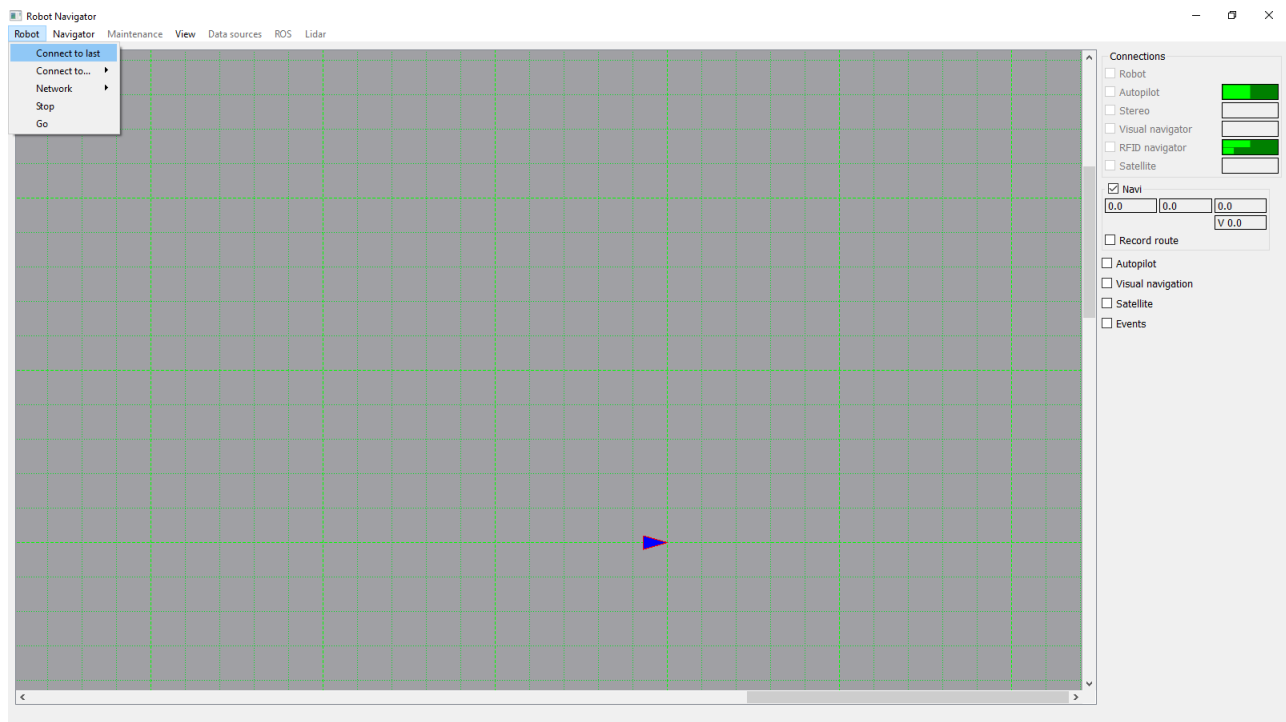


FIGURE 3.1 – Menu principal de WinNavigator

La flèche représente le robot. Sa couleur change suivant le mode dans lequel il se trouve (Point

mort en bleu, Manuel en vert, Autonome en rouge, Charge/Batterie faible en rose).

La connexion au robot se fait par adresse IP. Il suffit de configurer le logiciel pour trouver l'adresse IP correspondante (et d'être connecté au bon réseau) pour se connecter au robot. Une fois la connexion effectuée, il est possible de définir une carte locale pour le robot en téléchargeant une image dans ce dernier. Le logiciel demande pour cela de connaître l'échelle de l'image en mm/px. Ce qui donne la figure suivante.

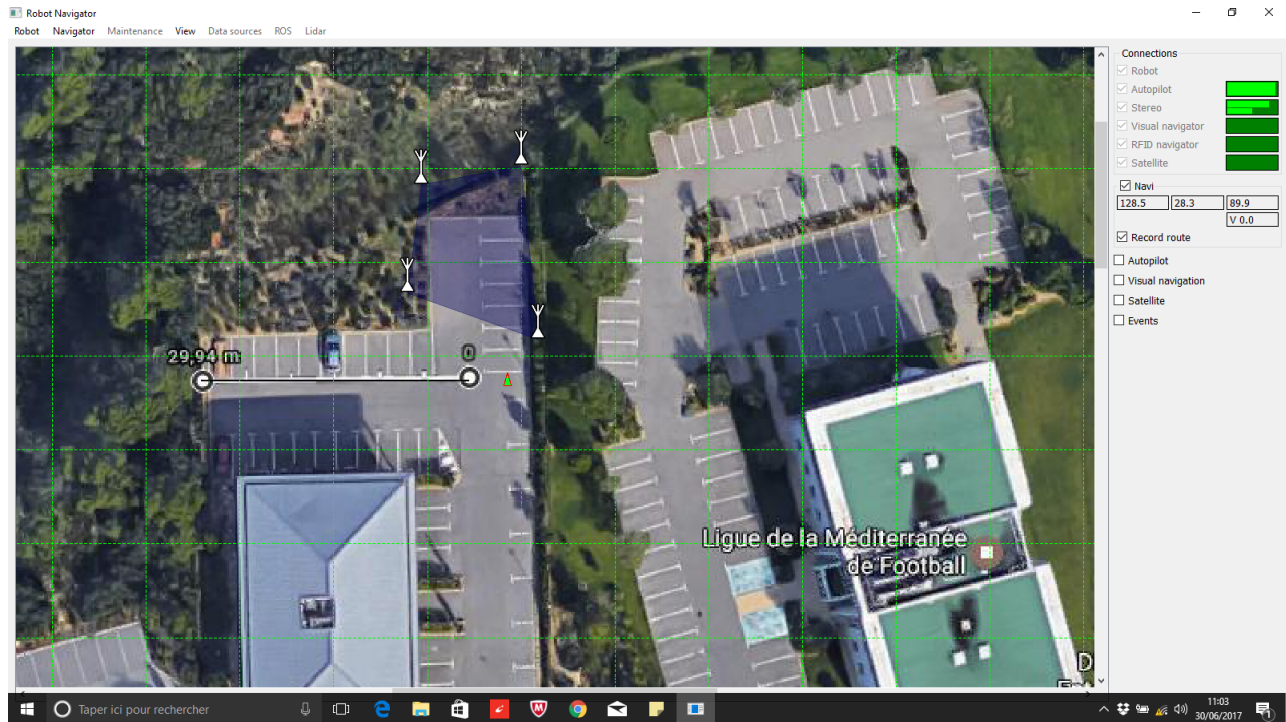


FIGURE 3.2 – Menu principal de WinNavigator avec carte

Cela va permettre à l'utilisateur de placer le robot sur la carte de façon à pouvoir visualiser son parcours. Cette carte est une carte locale pour le robot. Elle n'a aucune corrélation avec des données de localisation globales (le GPS).

En haut à droite de la fenêtre l'utilisateur va pouvoir visualiser le fonctionnement des différents modules de localisation à l'aide de barres vertes. Il y a deux barres vertes par module. La barre supérieure permet de visualiser la qualité des mesures effectuées par le module en question, la barre inférieure va indiquer le degré d'utilisation du module au temps t , dans la navigation en général.

Pour lancer un apprentissage du robot, deux solutions, soit créer le chemin en pilotant le robot par radiocommande, soit créer les points directement sur la carte. Nous avons très souvent opté pour la première option pour des raisons de sécurité. La procédure est la suivante :

- Positionner le robot sur la carte.
- Cocher la case "Record Route" (A droite de la fenêtre).
- Effectuer le parcours.
- Décocher la case "Record Route" une fois que le parcours est terminé.

Tout ceci nous donne la figure 3.3

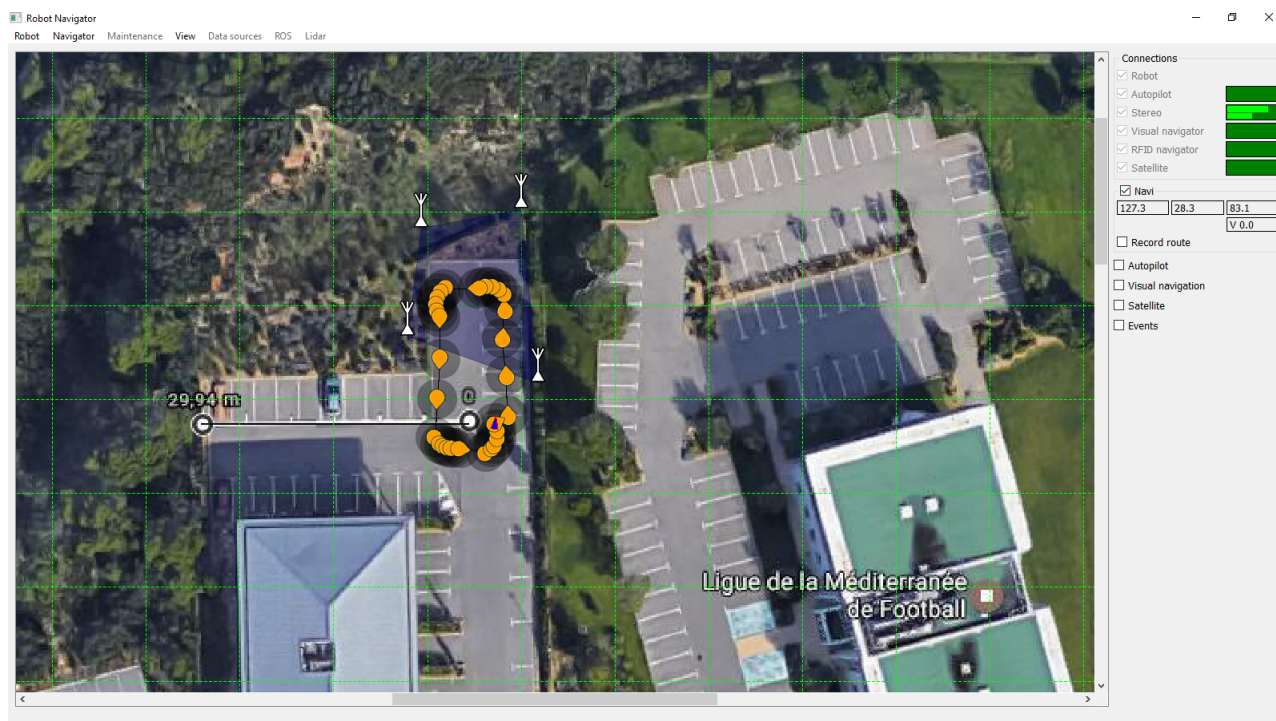


FIGURE 3.3 – Menu principal de WinNavigator avec carte

Les points oranges sont les points de passage du robot. Il est possible de les modifier (Pour en faire des points d'arrêt momentanés, ou de modifier la position sur la carte.) ou de les supprimer si besoin est.

Par la suite l'utilisateur peut soit envoyer directement le robot faire sa ronde, soit compléter l'apprentissage avec la détection de points d'intérêts sur le parcours appris. Il suffit juste de cocher les cases "Visual Navigation" et "Learn" pour que le robot se lance en mode autonome pour effectuer le parcours afin de capter les points d'intérêts.

3.2 Système de communication

Le robot possède une antenne Wi-Fi Ubiquiti Bullet M2 embarquée dans sa tour. Cette antenne permet au système embarqué du robot de communiquer avec le PC de contrôle par liaison Wi-Fi et de recevoir les commandes envoyées depuis le PC. Nous allons aborder dans cette partie les configurations réseau qui ont été utilisées lors de l'utilisation des robots.

3.2.1 Configuration réseau initiale

Lors de notre première utilisation des robots, ces derniers n'étaient pas configurés de la même façon :

- Bobby (Orange Bleu) avait son antenne Wi-Fi réglée en point d'accès, ce qui permettait de créer son propre réseau Wi-Fi afin d'accéder aux fonctionnalités du robot.
- Jack (Gris Camouflage) avait son antenne Ubiquiti réglée en mode station. Dans cette configuration, il n'était pas possible de s'y connecter, car l'antenne recherchait un réseau Wi-Fi qui nous était inconnu. En outre nous n'avions pas encore les éléments nécessaires afin de créer notre propre

architecture réseau et donc d'utiliser ce mode de fonctionnement.

Nous avons donc opté au début pour l'utilisation des antennes en mode point d'accès pour accéder aux robots rapidement, et pouvoir progresser dans notre compréhension des robots.

3.2.2 Configuration réseau finale

Lors de l'étude du système de correction GPS, il nous est apparu que les antennes Ubiquiti devaient être configurées en mode station pour que la base GPS puisse envoyer les corrections au robot. Nous avons alors mis en place l'architecture réseau nécessaire au bon fonctionnement de ce module. En outre, cela permet de gérer plusieurs robots depuis le même réseau en changeant simplement l'adresse IP visée ce qui simplifie l'utilisation générale.

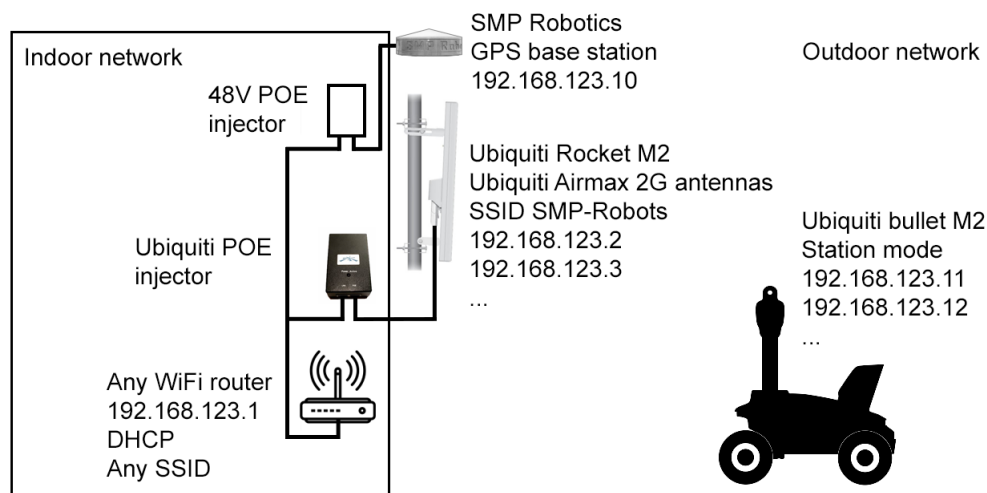


FIGURE 3.4 – Schéma de l'architecture réseau de l'interface Base GPS/Utilisateur/Robot

Comme indiqué sur la figure 3.4, afin d'étendre la portée du réseau nous avons opté pour l'utilisation d'une autre antenne Wi-Fi. Nous avons choisi d'utiliser l'Ubiquiti Rocket M2 comme conseillé par le fabricant. Pour utiliser cette architecture réseau, nous avons configuré l'Ubiquiti et l'ordinateur de contrôle avec des adresses IP statiques pour permettre la connexion. Nous utilisons le sous-réseau 192.168.123.x dans notre réseau. Il a donc fallu régler l'intégralité des adresses IP présentes sur le réseau sur ce format (Routeur, Ubiquiti Rocket, Ordinateur).

D'autre part, les antennes Ubiquiti présentes sur les robots redirigent les ports des cartes de contrôle du robot (Adresse IP de classe A, 10.36.3.x) vers une adresse de classe C en sortie, afin de rendre les appareils embarqués disponibles depuis l'ordinateur de contrôle (L'adresse de sortie est unique, pour pouvoir se connecter aux différents appareils via les applications ou en SSH, il suffit de choisir le bon port de connexion). Afin de pouvoir se connecter au robot depuis le réseau créé par l'antenne Rocket, nous avons dû également régler ces adresses sur le sous-réseau approprié.

Cette configuration nous a permis de passer à l'intégration de la base GPS (cf 3.2.3.1) et de pouvoir augmenter la distance de contrôle des robots de quelques dizaines de mètres, à plus de 100m.

3.2.3 Intégration de la base GPS

Après avoir mis en place le réseau pour pouvoir contrôler les robots, l'intégration de la base GPS dans ce réseau (Qui est la finalité de l'architecture utilisée) a été l'étape suivante. Cette étape a été beaucoup plus longue que prévue initialement notamment à cause des indications (et du manque d'indications par moment) données par le fabricant.

3.2.3.1 Première configuration de la base GPS

Comme indiqué dans la partie 2.4 le GPS communique via un protocole IP avec le robot. Pour que le robot envoie sa position à la base, ce dernier doit connaître son adresse IP. Ceci est configuré dans la carte Navigator du robot dans le fichier Qtslam.ini, auquel on peut accéder via une console SSH. L'un des problèmes concerne la connaissance de l'adresse IP de la base GPS. En effet SMP Robotics nous avait indiqué que la base possédait deux adresses IP :

- Une adresse IP de classe C, réglée par l'utilisateur et qui peut ne pas être disponible.
- Une adresse de classe B dans la catégorie zeroconf (169.254.x.x), qui elle est toujours disponible et invariante.

Selon SMP pour accéder à la base GPS et régler l'adresse IP sur le sous-réseau approprié, il suffisait d'utiliser l'application GPSFinder, développée par leur équipe, pour trouver l'adresse zeroconf, et accéder au robot via un navigateur web. Il s'est avéré que leur logiciel était inefficace pour trouver cette adresse. Dans un premier temps nous avons trouvé l'adresse IP par hasard en passant dans la documentation envoyée par SMP. Mais cela ne nous aurait pas permis de configurer une base en général. Du coup en démontant la base GPS, et en remarquant la présence du Tral 5 Poe (cf 2.4) nous avons pu récupérer la documentation de ce dernier et retrouver l'adresse IP zeroconf en utilisant la fonction "find" du logiciel Multivision en connectant la base en ethernet au PC.



FIGURE 3.5 – Interface Multivision (En rouge la fonction find pour trouver l’adresse IP de la base GPS)

Une fois cela effectué, nous avons enfin eu le protocole à appliquer pour une première configuration de la base GPS.

3.2.3.2 Connexion entre la base GPS et le robot

Dès que nous avons eu accès à l’adresse IP de la base GPS, nous avons tenté d’appliquer le schéma de l’architecture réseau qui nous a été présenté par SMP Robotics. Sans succès au départ. En effet, il nous est assez vite apparu que la base GPS n’arrivait pas à communiquer avec les éléments internes du robot (Ne serait ce que via un ping). Nous avons donc testé plusieurs configurations des différents éléments du réseau pour arriver à mettre en relation les deux éléments. Afin

de tester la connexion entre le robot et la base GPS, nous avons procédé de la manière suivante :

- Connexion à la carte Navigator du robot en SSH.
- Tentative de ping de l'adresse de la base GPS.
- Connexion en SSH à la base GPS.
- Tentative de ping de l'antenne Ubiquiti Bullet M2 embarquée sur le robot.
- Tentative de ping de la carte Navigator

Première configuration testée :

Routeur Asus en mode pont connecté en Wi-Fi au réseau créé par l'Ubiquiti Rocket M2 et pour transmettre les données de la base GPS vers ce même réseau.

Antenne Ubiquiti Rocket M2 en mode point d'accès/Routeur, pour créer le réseau interfaçant la base GPS, l'utilisateur, et le robot.

Antenne Ubiquiti Bullet M2 (embarquée sur le Robot) en mode station/Routeur connectée au réseau créé par le Rocket M2.

Deuxième configuration testée :

Routeur Asus en mode point d'accès connecté en ethernet au réseau créé par l'Ubiquiti Rocket M2.

Antenne Ubiquiti Rocket M2 en mode point d'accès/pont, pour créer le réseau interfaçant la base GPS, l'utilisateur, et le robot, et assurer en théorie le passage des informations du routeur Asus vers le robot.

Antenne Ubiquiti Bullet M2 (embarquée sur le Robot) en mode station/Routeur connectée au réseau créé par le Rocket M2.

Après ces tests il nous est apparu que le transfert de données rencontrait une difficulté au niveau de l'antenne Wi-Fi embarquée sur le robot. Nous avons donc tenté de brancher directement la base GPS au Robot via la switch embarquée sur ce dernier.

Après modification de l'adresse IP de la base GPS pour la configurer sur le même sous-réseau que les autres composants internes du robot (adresse IP de classe A, sous réseau 10.36.3.x), nous avons pu envoyer un ping à la base GPS depuis le robot. Il est donc apparu que le problème était un problème de redirection de port sur une des deux antennes Ubiquiti.

En redirigeant le port du GPS depuis l'Ubiquiti Rocket M2 vers l'adresse 10.36.3.54, nous avons enfin pu connecter la base GPS au robot à travers le réseau sans-fil.

3.3 Tests de précision

Afin de vérifier les informations données par SMP Robotics, nous avons cherché à récupérer les données d'erreur entre le robot et le point cible. Seul problème, SMP Robotics ne nous a pas donné accès au code source de l'algorithme de correction, et nous a indiqué que l'erreur se voyait graphi-

quement sur l'application d'interface WinNavigator (Pas de valeur chiffrée à exploiter donc). Ceci étant dit, nous avons d'abord cherché à vérifier que le fonctionnement simultané de plusieurs modules de localisation rendaient effectivement le robot plus précis, et nous avons testé la robustesse de l'algorithme de correction du robot, en lui faisant effectuer des parcours relativement complexes en autonomie, et en le faisant passer à travers des points de passages, durant ces parcours, dont la largeur est de 2m50 sauf pour la porte de départ qui est large de 5m.

Nous avons choisi de parcourir le chemin suivant dans un premier temps :

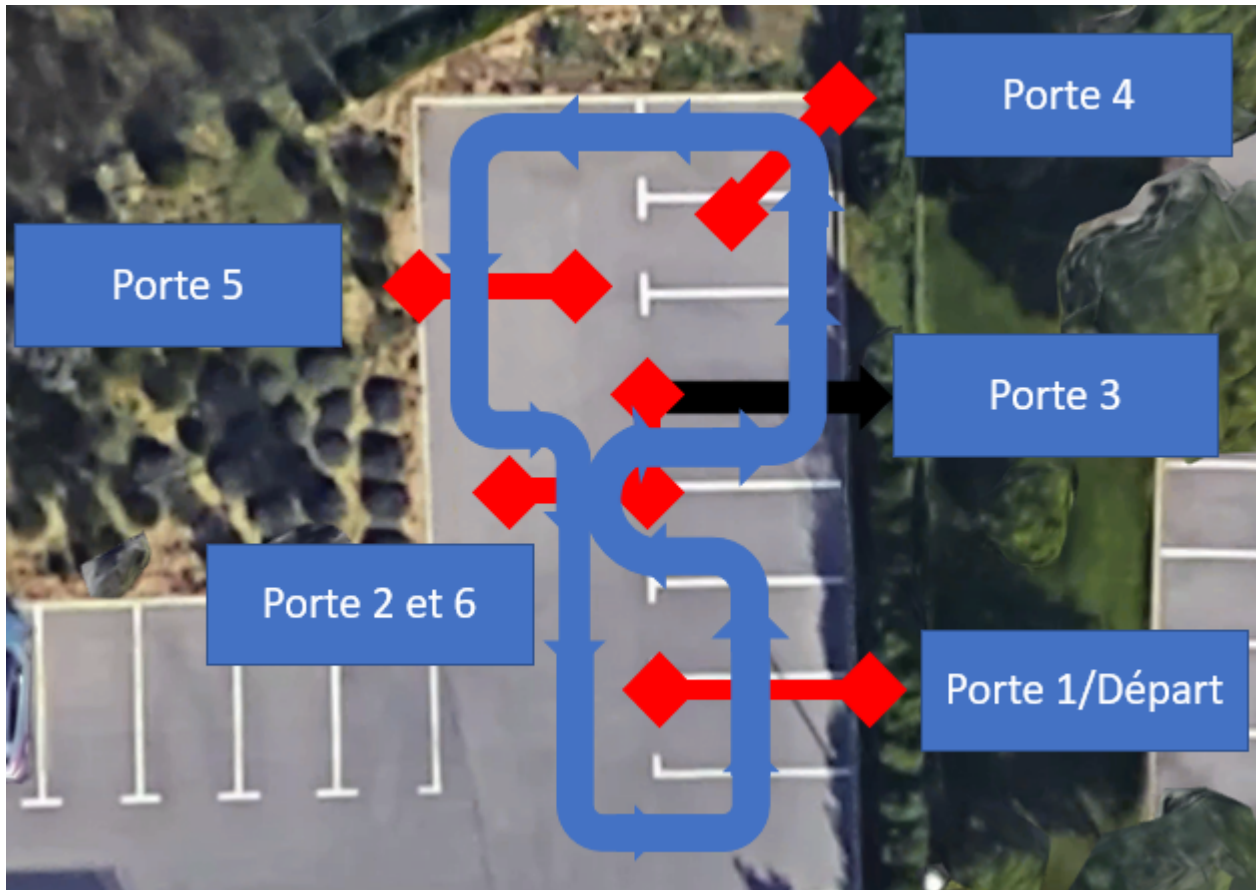


FIGURE 3.6 – Parcours du robot

3.3.1 Protocole de test

Le test est censé se dérouler sur 10 tours de piste. Au cours de ces 10 tours, nous avons noté l'intégralité des erreurs commises par le robot (Ratage de porte, passage de porte après arrêt et correction de la trajectoire initiale, perte critique de la trajectoire). Les tests ont eu lieu dans un environnement fermé pour minimiser les risques d'accident avec un autre véhicule.

En outre nous avons enregistré le retour du positionnement du robot via une capture vidéo de l'écran de contrôle pour observer les informations données par le robot.

3.3.2 Localisation inertielle seule

Pour commencer nous avons testé la localisation inertielle sans aucune autre interaction, pour voir si il était possible de ne dépendre que de cette dernière sur une longue période.

Dans la figure 3.6 les éléments en rouge sont les portes dans lesquels le robot devait passer, et les flèches en bleu représentent son parcours.

Nous nous sommes rendus compte durant le deuxième tour que le robot avait subi une erreur fatale dans son positionnement, nous obligeant à arrêter le robot. Il s'est avéré après visionnage de la vidéo que le robot a subi une discontinuité dans son positionnement qui rendait l'intégralité de la carte et du parcours non valide. D'où l'erreur observée. Ceci, en plus des observations faites durant la prise en main des robots, nous montre que le positionnement inertiel n'est pas stable pour l'instant et ne peut être utilisé seul pour la navigation.

3.3.3 Test de la détection de points d'intérêts

Au vue de l'échec du test précédent, nous avons ajouté le processus d'apprentissage de points d'intérêts à la navigation inertielle, en gardant le même parcours qu'en 3.6. Les résultats sont affichés dans la figure ci-dessous.

Tour	Erreurs commises
Tour 1	R:P6
Tour 2	R:P6 D:P2
Tour 3	D:P2,P3,P4,P6
Tour 4	R:P2,P3 D:P5,P6
Tour 5	R:P6 D:P2,P3,P4,P5
Tour 6	R:P2,P3 D:P4,P6
Tour 7	D:P2,P5,P6
Tour 8	D:P6
Tour 9	R:P2,P3
Tour 10	R:P2,P3,P5,P6

FIGURE 3.7 – Résultats du test de la localisation visuelle (R :Portes Ratées, D : Portes passées après modification de la trajectoire, P :Porte)

Comme on peut l'observer en 3.7, le robot a tendance à perdre en précision au fur et à mesure du test. Une partie aléatoire (Liée à la détection d'obstacle) se dégage également de ce test. Ceci dit le résultat est nettement plus concluant ne serait ce qu'au niveau de la complétion du test sans erreur fatale à son déroulement. Sachant que les portes faisaient 2.5m pour la plupart, après observation et en prenant en compte la détection d'obstacle (Qui modifie la trajectoire des robots), nous en avons conclu que les robot sont précis à 2m avec la localisation visuelle.

3.3.4 Tests de navigation RFID

Selon le fabricant, le module de localisation par transmission RFID est prévu pour les manoeuvres complexes du robot. Comme un virage serré, ou un passage dans une porte large de 2m. Durant ces tests nous avons non seulement vérifié la stabilité du module, mais aussi la capacité

du robot à effectuer ce type de mouvements précis. Pour ce faire, nous avons utilisé deux parcours différents :

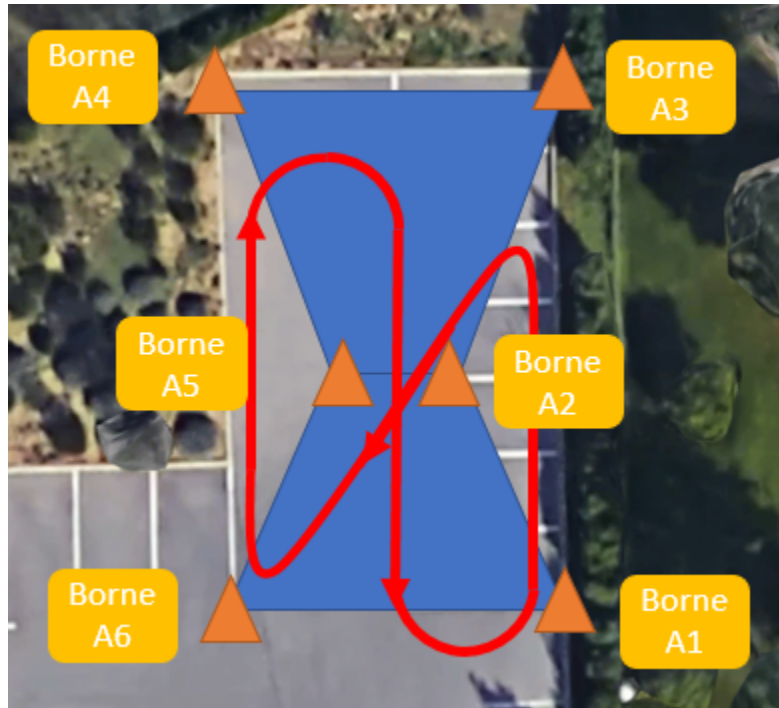


FIGURE 3.8 – Parcours n°1 du Robot avec RFID (Bornes RFID en Orange, Périmètre défini en bleu)

Ce parcours reproduit l'environnement autour d'une porte. Nous avons défini le périmètre formé par les RFID de cette manière afin de reproduire les effets d'un mur. (En effet la transmission RFID est impossible à travers les murs, typiquement le genre de situation présente lorsque le robot doit passer une porte.)

3.3.4.1 Premier test

Après les dix tours prévus dans le protocole de test voici les résultats que nous en avons retiré.

Tour	Erreurs commises
Tour 1	C:A2 R
Tour 2	C:A2 D:A6 R
Tour 3	
Tour 4	
Tour 5	D:A5,A6
Tour 6	H:A5 D:A6
Tour 7	D:A6
Tour 8	D:A5,A6 C:A2
Tour 9	C:A2, R
Tour 10	C:A2

FIGURE 3.9 – Résultat du 1er test avec module RFID (C :Collision avec un cône, R :Porte ratée, D :Recalcul de la trajectoire, H :Hésitation sans modification de trajectoire)

Deux conclusions se sont détachées de ce test. D’une part, le passage dans la porte immédiatement après le virage a été la manoeuvre la plus compliquée pour le robot, c’est en effet là que le ratage de la porte a eu lieu à chaque fois. Le Robot n’a pas reçu suffisamment de corrections de la part des bornes RFID pour modifier la trajectoire en fonction. Le fait que le robot n’était pas dans le périmètre à ce moment explique ce problème.

D’autre part, même si ce n’était pas le but premier de ce test, nous avons remarqué que les caméras stéréoscopiques, assurant la détection d’obstacles avaient des angles morts. Ce qui explique les collisions avec les cônes (Combinaison d’une taille trop petite des cônes, du virage effectué, et du choix des caméras ou de la conception de la coque du robot.)

3.3.4.2 Deuxième test

SMP Robotics nous a assuré que les bornes RFID n’avaient pas à être présentes sur le parcours à tout moment pour assurer un positionnement plus précis grâce au module dédié. (Ie : Les bornes sont posées lors de l’apprentissage du robot pour l’aider à se localiser, puis retirées lors de la fin du processus, sans affecter la précision du positionnement du robot)

C’est ce que nous avons voulu vérifier ici.

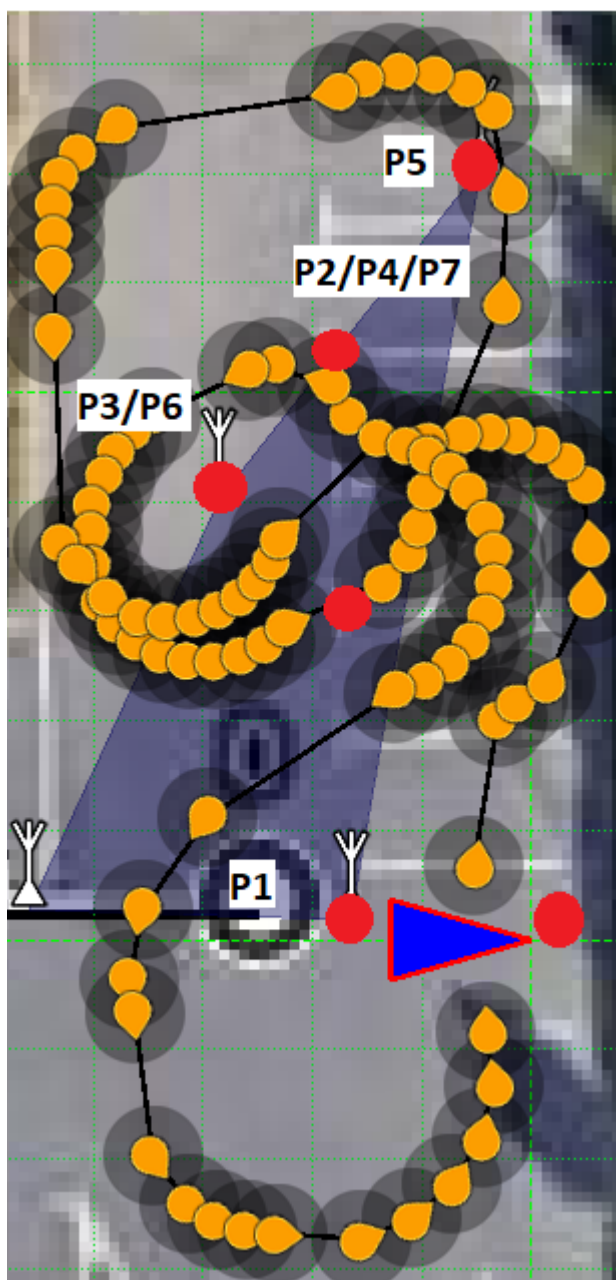


FIGURE 3.10 – Parcours n°2 du Robot avec RFID (Bornes RFID blanc, Périmètre défini en bleu, Chemin indiqué par les points oranges, et portes en rouge)

Comme indiqué sur la figure 3.10 nous avons augmenté la complexité du trajet pour tester la stabilité du robot.

Tour	Erreurs commises
Tour 1	R:P3,P6*
Tour 2	R:P3,P5
Tour 3	R:P3
Tour 4	R:P3
Tour 5	X:P5
Tour 6	X:P5 R:P3
Tour 7	X:P5
Tour 8	Non effectué
Tour 9	Non effectué
Tour 10	Non effectué

FIGURE 3.11 – Résultat de la première partie du test (C :Collision avec un cône, X :Porte ratée, R :Recalcul de la trajectoire, H :Hésitation sans modification de trajectoire)

NB : Lors de l'apprentissage nous avons fait passer le robot trop près d'un des cônes de la porte 6. Ce qui provoquait des erreurs "normales" (Le robot n'est pas censé avoir ce degré de précision). En conséquence nous avons modifié la position de ces points par ordinateur pour éliminer cette erreur. En outre nous n'avons effectué que 7 tours car le robot commettait systématiquement les mêmes erreurs.

Après cette première phase de test nous avons retiré les bornes RFID.

Tour	Erreurs commises
Tour 1	R:P3 H:P3
Tour 2	R:P3
Tour 3	R:P3
Tour 4	R:P3
Tour 5	R:P3
Tour 6	Non effectué
Tour 7	Non effectué
Tour 8	Non effectué
Tour 9	Non effectué
Tour 10	Non effectué

FIGURE 3.12 – Résultat de la deuxième partie du test (C :Collision avec un cône, X :Porte ratée, R :Recalcul de la trajectoire, H :Hésitation sans modification de trajectoire)

NB : Nous n'avons fait que 5 tours ici car les erreurs étaient les mêmes à chaque tour.

Lors de l'aquisition du chemin à parcourir, nous avons remarqué une augmentation des discontinuités de positionnement du robot, qui rendait les points marqués précédemment faux par rapport à la position relative du robot. En outre lorsque nous avons retiré les RFID, le robot était dans l'ensemble moins précis (Virages plus larges, en P3), mais aussi moins erratique dans son comportement.

Les résultats observés ainsi que les informations données par le robot sur Win Navigator nous montrent que les RFID doivent être présentes à tout moment pour que le module fonctionne

Nous en avons également déduit l'hypothèse suivante :

Bien que le module de localisation RFID permette de localiser le robot plus précisément, la combinaison avec les autres modes de localisation rend l'ensemble plus instable dans son positionnement. En effet la visualisation de la position calculée par le module RFID était précise, mais son apport dans le positionnement en général du robot se faisait parfois ressentir de façon négative.

3.3.4.3 Troisième test

Pour vérifier l'hypothèse précédente, nous avons repris le même parcours en modifiant le protocole d'apprentissage.

Nous avons désactivé le module RFID pendant l'apprentissage du chemin par le robot pour le réactiver après.

Tour	Erreurs commises
Tour 1	X:P6 R:P3,P4,P5
Tour 2	R:P2,P6 H:P2,P7
Tour 3	R:P5,P6 X:P7
Tour 4	
Tour 5	R:P5,P6
Tour 6	Non effectué
Tour 7	Non effectué
Tour 8	Non effectué
Tour 9	Non effectué
Tour 10	Non effectué

FIGURE 3.13 – Résultat de la deuxième partie du test (C :Collision avec un cône, X :Porte ratée, R :Recalcul de la trajectoire, H :Hésitation sans modification de trajectoire)

Cette configuration n'a pas éliminé les discontinuités durant l'apprentissage. Après avoir échangé avec SMP sur la question, l'absence de connexion à la station GPS serait la raison de ces irrégularités.

3.4 Corrections GPS

Cette partie sera un peu plus succincte que les autres tests car nous n'avons pas eu le temps de tester de manière approfondie la stabilité du système avec les corrections GPS.

Toutefois, dès la connexion de la base au robot, un problème est survenu. Lorsque nous avons lancé l'apprentissage, le robot a immédiatement connu un saut de position qui a invalidé le trajet. Ce phénomène s'est répété systématiquement ce qui nous pousse à croire que le problème de stabilité du système mentionné dans la partie 3.3.4.2 est encore présent ici.

Le fabricant nous a également mentionné une série de raisons pouvant expliquer ce phénomène (Centrale inertielle défaillante, Bornes RFID, ou points de repères visuels faussant la perception du robot). Ceci nous amènera à effectuer des tests supplémentaires à l'avenir pour trouver le problème.

3.5 Détection d'obstacles

En parallèle des tests développés dans 3.3, nous avons également pu de facto, tester la détection des obstacles faites par le robot.

3.5.1 Module stéréoscopique et radars de recul

Lors de ces tests nous avons placé un ou plusieurs obstacles sur la route du robot. Nous avons également tenté de vérifier la tolérance du robot à l'apparition d'un obstacle (Plot, ou bien nous mêmes) dans son champ de vision dans un laps de temps très court avant collision.

Il apparait que l'algorithme de détection d'obstacle est au point, lorsque la combinaison des caméras stéréoscopiques à l'avant du robot et des radars de recul à l'arrière détecte un obstacle, le robot ne rentre pas en collision avec celui-ci, quitte à s'arrêter et à reculer pour contourner l'obstacle. Le retour au chemin d'origine est également robuste et se fait globalement sans accroche.

Le véritable souci ici a déjà été soulevé précédemment, la conception du robot et le choix des caméras fait que la détection stéréoscopique possède certains angles morts qui empêchent la détection. Ces problèmes sont rares mais ont déjà été constaté lorsqu'un objet de la taille d'un plot se trouve dans un virage serré.

3.5.2 Cartographie du terrain

L'algorithme de classification du terrain a été également abordé pour vérifier son bon fonctionnement. L'utilisation du logiciel MapperViewer a été concluante dans la classification de différents types de terrains, et dans l'aide au parcours du robot.

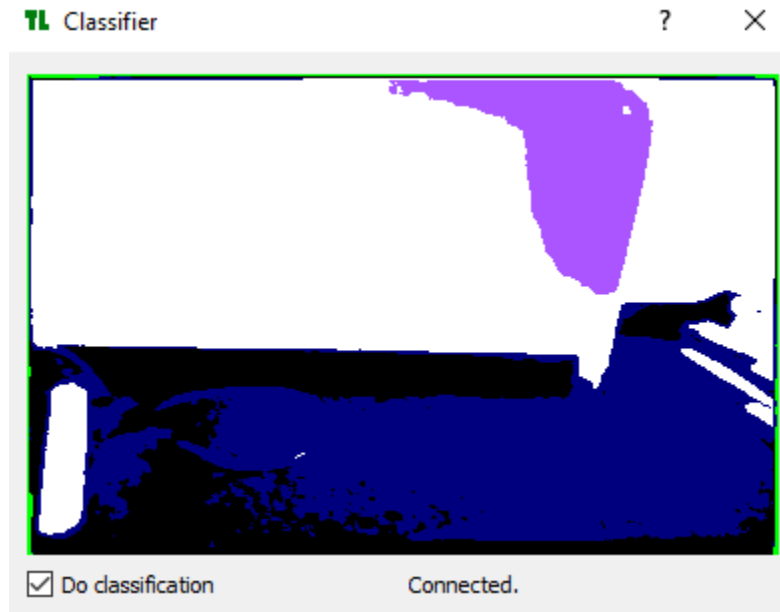


FIGURE 3.14 – Classification des différents éléments dans le champ de vision du robot à l’atelier (Blanc : Mur, Bleu : Sol, Noir : matériau entre le mur et le sol, Vert : Zone d’ombre, Violet : zone de surbrillance)

La classification est assez fidèle à l’image d’origine au niveau des matériaux, malgré quelques zones d’erreurs (dues à la luminosité assez faible)

3.6 Vidéosurveillance

Les éléments à tester dans le dispositif présenté en 2.7 sont la distance maximale de détection de mouvement, et la compatibilité des éléments présents avec différents systèmes de sécurité (En particulier Genetec). En effet la plupart des systèmes de sécurité utilisent un logiciel pour tout centraliser, et il faudra intégrer le robot et au moins ses composantes liées à la vidéosurveillance à l’avenir.

Après discussion avec des représentants de Genetec, il semble très improbable que les composants présents à l’heure actuelle sur le robot soient compatibles.

3.6.1 Détection de mouvement

Pour tester cette fonctionnalité, il nous a suffi de bouger devant les caméras panoramiques, en s’éloignant, pour déterminer la distance à laquelle la caméra PTZ ne pointait plus sur nous. En effectuant ce procédé, il s’est avéré que le robot ne pointait plus vers nous au delà de 11m. Il peut y avoir deux raisons à cela :

- Le robot a détecté un mouvement plus proche de lui.
- Le robot ne capte pas au delà de ces 11m

En attendant des tests plus approfondis nous penchons pour la deuxième solution car à part le mouvement de la végétation causé par le vent il n’y avait pas d’autre élément mouvant dans la scène.

Chapitre 4

Conclusions

4.0.1 Analyse des résultats de tests

Les résultats concernant la navigation autonome et le positionnement du robot sont mitigés, le robot est capable de reproduire un circuit de manière autonome, mais les gros problèmes de stabilité évoqués précédemment sont un problème pour la commercialisation en l'état de la machine. Le fabricant doit nous envoyer une nouvelle version du robot (Version 5.1) d'ici fin Août, qui est censée être beaucoup plus stable à ce niveau. La détection d'obstacle est dans l'ensemble opérationnelle, et au cours de nos tests nous n'avons pas eu de collision impliquant le robot et un autre véhicule, ou une autre personne.

La décision de ne pas utiliser de LIDAR peut paraître étrange aujourd'hui, mais cette technologie ne s'étant démocratisé au niveau des prix que récemment, il est possible que lors de la conception du robot l'équipe de SMP a effectivement trouvé les LIDAR trop chers, et que les algorithmes de vision stéréoscopiques étaient suffisant pour détecter les obstacles. Ceci dit dans une version future il pourrait être intéressant d'embarquer un LIDAR pour permettre une cartographie de l'environnement du robot à 360°.

En revanche, concernant la vidéosurveillance le robot n'est pas encore au niveau des standards que l'on attend (Détection à 100m et rendu vidéo HD), et les informations que nous avons à notre disposition concernant le dispositif sur la prochaine machine ne sont pas encourageantes.

4.0.2 Décisions concernant les appareils liés au robot

En vue des résultats évoqués précédemment, il a été décidé que nous allons garder la partie physique (Moteur, Châssis....) et la partie Navigation du robot en l'état pour l'instant. En effet, bien que la précision observée ne soit pas à la hauteur de nos attentes (Et des valeurs annoncées), nous partons du principe que la précision du robot n'est pas une fin en soi. Tant que le drone est en capacité de reproduire le parcours demandé sans aide, et sans entrer en collision avec un autre objet, il pourra assurer sa mission de surveillance.

En revanche tout ce qui a trait aux caméras et à l'enregistrement des machines sera changé pour atteindre les standards requis. Nous sommes actuellement en train de rechercher les composants que nous voulons intégrer au robot pour remplacer les caméras et le DVR.

Par ailleurs, nous avons également été mis en relation avec certaines entreprises proposant des appareils et des capteurs que nous pourrions incorporer au robot en tant qu'option. Typiquement, il a été évoqué l'intégration d'un spray ADN pour marquer un intrus, et pouvoir le retrouver et le

condamner par la suite.

Pour cela, sachant que nous n'avons pas pour l'instant accès au code source du robot, nous allons incorporer un PC embarqué durci, afin de pouvoir intégrer les capteurs au robot.

Enfin, si la partie Navigation du robot va rester en l'état, la base GPS sera changée pour une base GPS basée sur le système RTK (qui nous permettra d'avoir davantage de précision dans la localisation du robot (une dizaine de cm) car les précisions observées jusque là ne sont pas entièrement satisfaisantes.

4.0.3 Problèmes rencontrés

Le problème principal auquel nous avons été confrontés jusqu'ici, a été la communication avec le fabricant du robot. En effet les informations que nous avons reçues de leur part ont assez souvent été imprécises ou incomplètes. Il a par exemple été assez compliqué d'avoir ne serait ce que les informations concernant les caméras. Il sera d'autant plus important d'améliorer et de clarifier notre relation avec eux, que nous allons avoir besoin de l'intégralité de la nomenclature et des composants présents dans le robot pour la certification européenne de celui-ci.

Certains problèmes de communications sont également dues à des différences de vision concernant la machine. En effet SMP Robotics, n'est pas une entreprise qui vise le marché de la sécurité à la base. Du coup, ils n'attendent pas les mêmes standards que nous pour la machine (Notamment au niveau des caméras).

4.1 Remerciements

Je tiens dans ce rapport à remercier M. Cudicio pour m'avoir donné l'opportunité d'effectuer ce stage à TBC-France. Ainsi que M. Etienne et M. Willerval pour l'aide qu'ils ont apporté pendant le projet.

Bibliographie

Lien vers les caractéristiques de la caméra PTZ : <http://www.ismart-video.com/products/car/2015111456.html>

Lien vers le site de SMP Robotics : <http://www.smprobotics.com>

Cours de positionnement par vision, Vision 3D, par H el ene Thomas https://moodle.ensta-bretagne.fr/pluginfile.php/57902/mod_resource/content/1/Diapos2-Vision3D.pdf

Annexe A

Journal de bord du stage

Semaine 1 - Journal de bord 17/06/12

Début du stage de Kevin et Zacharie.

Journal de bord 17/06/13

Position de la clé en mode OFF, ou en mode CHARGE change juste la vitesse de charge de la batterie. (Potentiel dysfonctionnement d'une batterie)

Établissement de la documentation concernant les manettes et la charge des batteries en fonction des informations actuellement à notre disposition (TBC-France_RemoteSpecs_170613_01_KE, TBC-France_ChargeSpecs_170613_01_ZE)

Création d'une liste de tests à effectuer, et d'une liste de questions à poser. Ces listes sont amenées à évoluer. (TBC-France_Test_Plan_Robot_170613_01_ZE, TBC-France_Questions_Robot_170613_01_ZE)

Installations des API d'interaction avec les robots fournies par SMP.

Étude de la documentation concernant lesdites API.

Connexion des PC avec Bobby, récupération des flux vidéo, des caméras PTZ et stéréoscopiques.

Établissement de la documentation concernant la connexion des PC aux robots (TBC-France_Logiciel_ConfigurationRobot_170613_01_KE)

Établissement de la documentation concernant la marche à suivre pour initialiser une mission avec les robots via les API avec l'aide de la documentation fournie par SMP (TBC-France_Navigation_Specs_170613_01_ZE)

Établissement de la documentation concernant l'utilisation du logiciel « visual navigator » (TBC-France_Logiciel_VisualNavigator_170613_01_KE)

Préparation de la démo du 14/06 (Bobby en mouvement qui renvoie un feedback vidéo, Jack immobile au stand, affiches, dépliants...)

Journal de bord 17/06/14

Démonstration au Fort Ganteaume dans le cadre de l'évènement « Cercle du Capital Humain »

Scénario mettant Bobby en mouvement/feedback vidéo et Jack à l'arrêt/tour qui clignote. (1/4 de la batterie consommé)

Résumé des questions des invités:

Les robots sont-ils autonomes ou télécommandés? Comment fonctionne le parcours?

Quelle est la durée de la batterie? Quel est le mode de charge?
À quelle distance la camera PTZ peut-elle voir?

Intérêt particulier de M. Guillaume Rioux (Bobby/Jack), et M. Thierry (Châssis d'auto)

Aperçu du suivi GPS du Robot avec « Navigator » (fonctionnel – test complémentaire à effectuer)

Journal de bord 17/06/15

Établissement de la documentation concernant le logiciel en charge des caméras de proximité (TBC-France_Logiciel_StereoViewer_170615_02_ZE)

Établissement de la documentation concernant le logiciel en charge de l'apprentissage de la route (TBC-France_Logiciel_TerrainLearner_170615_02_KE)

Installation d'une machine virtuelle Linux pour interagir à bas niveau avec les robots à l'avenir (Poste TBC-France_1)

Tentative de connexion au système interne des robots via Linux (Connexion SSH).

Visite des nouveaux locaux et planification de l'espace de travail.

Journal de bord 17/06/16

Installation dans les nouveaux locaux, réarrangement de l'espace de travail.

Tentative réussie de connexion aux robots (Rez-de-Chaussée) depuis les PC du bureau (1^{er} étage)

Établissement de la documentation concernant le logiciel MvLauncher (TBC-France_Logiciel_MVLauncher_170616_03_KE)

Début de l'établissement de la documentation concernant le logiciel RadioNavigator (TBC-France_Logiciel_RadioNavigator_170616_02_KE), en vue des tests de lundi prochain, et des démonstrations prévues la semaine prochaine.

Test préliminaire du délai entre allumage et contrôle sur Bobby. Valeurs comprises entre 1m50 et 1min30 environ (1'51", 1'40, et 1'27, lorsque le mode télécommandé est activé aussi vite que possible soit environ 10 15 sec après démarrage de Bobby). Conclusions provisoires :

- Corrélation probable avec le temps d'activation du mode télécommandé via la manette.
- Délai minimum entre allumage et première tentative de prise de contrôle via la manette.
- Lancement complet en mode télécommandé en 2 min grand maximum normalement.
- Test supplémentaires requis d'une part pour plus de précision (en binôme plutôt qu'en monôme), d'autre part pour tester le temps de démarrage pour le mode automatique.

Ébauche de tutoriel concernant l'optimisation de l'allumage des robots (à compléter au fur et à mesure de la complétion des tests évoqués précédemment). (Documentation Répertoire Technique\TBC-France_Doc_ConfigurationRobot_170616_02_ZE, partie "Connexion et prise de contrôle via la manette")

Création d'une méthode pour déterminer l'échelle mm/px + tableur excel.

Semaine 2 - Journal de bord 17/06/19

- Tentative de navigation autonome via RFID (Infructueuse : Problème alimentation/connexion des antennes)
- Tentative de connexion au réseau de « Jack » (Infructueuse : Problème de configuration d'adresse IP)
- Échange des batteries de « Jack » et « Bobby » pour poursuivre les tests.
- Test de navigation autonome via satellite/Wi-Fi (Réussite)
 - o WinNavigator
 - 1) Placer le point d'origine du repère de la carte puis le robot en position sur celui-ci dans la bonne direction.
 - 2) Utiliser le bouton « Record Route » et manuellement diriger le robot dans son parcours.
 - 3) Gérer les points obtenus dans l'acquisition (espace convenable entre deux points), anticiper les obstacles potentiels en donnant des directives préventives au robot.
 - 4) Passer à VisualNavigator (Utiliser l'option « Learn »)
 - o VisualNavigator
 - 1) Utiliser la fonction « Start Learn Route)
 - 2) Lancer via WinNavigator le robot en mode « Autopilot » pour entamer la séquence de reconnaissance visuelle.
 - 3) Arrêter l'acquisition en appuyant sur « Stop Learn Route » puis laisser le robot gérer son parcours. (Camera stéréo sont fonctionnelles et permettent d'éviter des obstacles)
- **Un tutoriel exhaustif suivra. *******

Journal de bord 17/06/20

- Déplacement Aix → Paris.

Journal de bord 17/06/21

- Démonstration pour la société SEGRO à Blanc Mesnil.
 - o Bobby en mode automatique après avoir fait un tour de reconnaissance (Record, Learn, Go).
 - o Problème en mode autopilote à étudier. (Arrêt et perte de positionnement momentané, évitement d'obstacles)
 - o Intérêt remarqué mais demande d'information au niveau de la concurrence.
- Test à faire au niveau de la batterie. (Autonomie, sous condition de canicule, etc.)

Journal de bord 17/06/22

- Démonstration pour le séminaire organisé par Keolis
- Pb au niveau de la reconnaissance du sol (Distinction mur noir sol carrelage noir)
- Pb au niveau de l'évitement d'obstacles incapacité du robot à reculer pour changer sa trajectoire.
- Perte de précision importante au bout de quelques tours (Intérieur? Besoin de corrections supplémentaires via GPS, ou RFID)

Journal de bord 17/06/23

- Déplacement Paris → Aix.

Semaine 3 - Journal de bord 17/06/26

Établissement de la documentation concernant les mesures de sécurité du site et du bâtiment (TBC-France_Doc_CodeSécurité_1706126_01_KE)

Établissement d'un tableau aide-mémoire de poche des mesures de sécurité du site et du bâtiment (TBC-France_Tableau_CodeSécurité_1706126_01_KE)

Établissement du début d'un manuel d'introduction basique des robots (TBC-France_Doc_IntroRobots_1706126_01_ZE)

Finalisation d'un premier ensemble de question (TBC-France_Doc_QuestionsRobots_170611_01_RC)

Commande de piles rechargeable pour les émetteurs RFID - **14500, 3.7V, 800 mAh.**

Établissement des objectifs des deux prochaines semaines :

- Contrôle de la navigation RFID
- Contrôle du système de correction GPS
- Démonstrations 6 et 7 juillet 2017 au bureau
- Connaissance des certifications de Haivision – certification CE européen, pour implantation future
- Connaissance du mode de prévention de collision du robot (mesure de sécurité en place)

- Connaissance du type d'interaction entre les robots et les logiciels Genetec, Milestones, Azursoft.
- Connaissance du coupe circuit (quelles fonctions?)
- Création d'un document informatif sur les fonctionnalités et la configuration du mode de navigation en 4G*****

Journal de bord 17/06/27

Établissement du début d'un manuel d'introduction basique des robots (TBC-France_Doc_UserManuelFR_1706127_01_KE)

Établissement d'un document de « formation/mise à niveau » pour le briefing de lundi matin avec Manuel et Frédéric (TBC-France_PresentationRobotDraft_170627_01_ZE)

Établissement d'un powerpoint de « formation/mise à niveau » pour le briefing de lundi matin avec Manuel et Frédéric (TBC-France_PresentationRobot_170627_02_KE)

Décision au niveau du routeur de transmission 4G – **Cradlepoint Core COR IBR900**

Journal de bord 17/06/28

Arrivé des batteries **14500, 3.7V, 800 mAh** pour les antennes RFID.

- Modification des socles pour batteries (pas de contact entre le connecteur du socle et le connecteur de la batterie) + (demande aux Russes d'adapter leur RFID, ou achat de batteries différentes?)
- Découverte de 2 modes sur les antennes (static and blinking)
 - o « Blinking » est identifiable sur Radio Navigator
- Minimum de 4 antennes nécessaire pour activer le mode RFID.

Détection par le robot dans le mode clignotant

Tuto sur RadioNavigator à compléter

Phare coloré avant du robot :

- Orange en mode learn
- Vert en mode auto
- Clignotement du phare implique traitement d'informations

Tentative de faire fonctionner la navigation radio. Réussie précision accrue évitement de cônes sur parcours en 8

Meeting avec Laurent pour définir les questions à poser à Alexei.

Tentative de switch des manettes (maintenir touche swap enfoncée)

Peaufinage du PPT (Encore à compléter)

Expérimentation sur les caméras stéréo (Fenêtre d'interpolation pour le calcul de la carte de disparité)

Journal de bord 17/06/29

Découverte de l'unit principale des deux robots

- Bobby U9
- Jack U1
- Rapidement enfoncer le 5ème bouton en partant de la gauche pour voir l'unit et maintenir enfoncé pour changer celui-ci.
- Toujours impossible de faire d'une télécommande, une télécommande universelle.

On peut se connecter à Jack via WiFi !!!!!

- Établissement de la doc pour régler les adresse IP des robots pour pouvoir y accéder.
- Échange des classes des adresses IP des réseaux Wan et Lan
- Définition du "range" des adresses IP dans leur classes correspondante.

Tests de fonctionnement sur Jack concernant les modes de navigation maîtrisés à ce jour :

- Connexion à Robot Navigator via l'adresse IP 192.168.123.11 (Pourquoi ?)
- Test de l'apprentissage du chemin (Validé)
- Test de l'apprentissage visuel (Validé)
- Test des RFID (Validé, Mais besoin de ligne directe non obstruée entre les RFID pour initialiser les bornes, typiquement pas de végétation entre les bornes d'où l'intérêt de les mettre en hauteur)
- Test des caméras stéréo (Validé plus capacité de reculer pour éviter obstacle trop proche, mais présence d'angles morts qui empêche la bonne détection des plots en virages)
- Tests de parcours en 8, et plus complexes.

Par la suite, il faudra récupérer de quoi fabriquer des poteaux rudimentaires pour y accrocher les Bornes RFID (4 mini, 8 idéalement)

Actualisation du tutoriel concernant les bornes RFID (TBC-France_Logiciel_RadioNavigator_170629_01_ZE) en vue des connaissances nouvelles acquises ces derniers jours.

Établissement d'un tutoriel expliquant le fonctionnement des bornes RFID (TBC-France_Tuto_BornesRFID_170629_01_ZE)

Journal de bord 17/06/30

Tests de fonctionnement sur Jack concernant le mode RFID

- La complexité du parcours peut être augmentée en corrélation avec le nombre d'antennes RFID

- L'antenne « A5 » semble défectueuse - problème de connexion

Mise au point de la théorie concernant le mode et le procédé satellite du robot – le robot est bel est bien équipé d'un système de localisation interne.

- Barre graphique verte de Win Navigator a l'appui. (Première barre = niveau de précision des données, seconde barre = niveau d'utilité de la méthode.)

Test de navigation autonome sans carte uploadée au préalable.

- Le robot fonctionne en mode autonome dans cette configuration. La carte n'est pas indispensable au robot. Ceci dit elle est indispensable si on veut définir le chemin directement sur le PC.

Test de navigation autonome en définissant le chemin directement sur le PC

- La navigation autonome via ce procédé est moins fluide mais fonctionne. Ici la position initiale du robot sur la carte (Et à priori la carte son échelle, et la direction du robot) est essentielle.

Semaine 4 - Journal de bord 17/07/03

Présentation à Frédéric, Mathilde et Manuel

Découverte, configuration et ébauche de documentation du logiciel de contrôle/interface des robots (TBC-France_Doc_RobotDiagnosticControlTool_170703_01_ZE)

Retour sur les réponses de l'échange avec Alexei :

- Actualisation du document (TBC-France_Doc_RobotSpecs_170703_01_KE) – définition des logos LEDs et code couleur.
- Actualisation du document (TBC-France_Logiciel_WinViewer_170703_01_ZE) – explication approfondie des fonctions et de la configuration.
- Actualisation du document (TBC-France_Doc_UserManuelFR_170703_01_KE) - section Appareil.

Journal de bord 17/07/04

Préparation de la démonstration de jeudi et vendredi.

Identification du problème pour Bobby au niveau de la marche arrière : Radar de recul défectueux. (Problème Hardware certainement)

Prise de vidéos pour un clip avec Bobby.

Test des RFID à prévoir un week-end (Laisser les RFID pendant 1 jours ou 2 pour valider les 24h de préparations).

Préparation du setup pour les démos de jeudi et vendredi.

Journal de bord 17/07/05

Suite de l'élaboration des manuels utilisateurs - Manette

Suite de l'élaboration du logiciel de diagnostics (TBC-France_Logiciel_RobotDiagnosticsAndControlTool_170705_01_ZE) Explication onglet Device info.

Modifications du document de questions

Démo et prise de photos/vidéos avec Bobby (orange bleu) en vue de l'élaboration d'un clip promotionnel. (Marketing → TBC-France Photos Bobby & Jack)

Journal de bord 17/07/06

Réunion concernant les questions à poser à SMP Robotics.

Élaboration du second document de questions, et du template utilisé pour ce type de document.

Démonstration du robot. Utilisation du logiciel de diagnostics pour checker la température du robot, et autres paramètres.

Journal de bord 17/07/07

Fin de l'élaboration des questions pour SMP, modification du template utilisé.

Suite de la création du manuel utilisateur

Démonstration l'après-midi. Tests concernant la détection d'obstacles, et à quel point la trajectoire corrigée peut être précise.

Réunion concernant les tâches pour la semaine prochaine :

- Tenter de mettre en place le système de correction GPS
- Étudier la solution de l'antenne Ubiquiti proposée dans le schéma du correcteur GPS
- Étudier les composants hardware du robot directement, et schématiser la structure de celui-ci
- Si les caméras se révèlent non compatibles avec Génétéc trouver des solutions alternatives
- Étudier les capteurs possibles pour les détecteurs de gaz, caméras thermiques, marqueurs ADN

- Pistes possibles (mobotics.com, caméras avec une haute résolution générale + Solution de caméras embarquées fabriquées dans la région d'Aix.

- Commencer à travailler avec un contact de chez Génétéc sur l'intégration du logiciel au robot.

Journal de bord 17/07/11

JIRA

- Présentation du logiciel de gestion de projet JIRA, pour permettre un suivi des projets, en temps réel et par différents protagonistes qui opèrent sur des fuseaux horaires différents.
- Dans les jours à venir, catégoriser les catégories de bugs, et fixer une nomenclature (si ce n'est déjà fait) pour les numéros de série des produits.

Découverte

Radars de recul

- Question concernant les radars de recul, y'en a-t-il deux ou un, comment fonctionnent-ils? (Un seul câble pour les relier au système, tout en identifiant les radars, et sans la présence apparente d'autres composants pour l'émission des ultrasons).

Tour Wifi

- Après le démontage de la tour, il apparait que l'architecture est à revoir de façon à pouvoir retirer entièrement la coque protégeant la tour. En effet à part la caméra PTZ, il est difficile d'accéder aux autres composants pour l'instant.

Caméras (PTZ et autres)

- Ceci dit puisque la caméra PTZ est relativement facile d'accès, il pourrait être intéressant de changer de modèle (Meilleure qualité, intégration sur Génétéc), voire d'architecture physique (module PT séparé pour y monter une caméra thermique par exemple, ce qui serait moins cher, en retirant le socle de la PTZ, il y'aurait assez de place pour cela).
- D'autre part l'intérêt de placer des caméras avec une très haute résolution n'a pas d'autre intérêt que d'impressionner le client. En effet via le Wi-Fi ou la 4G, on ne peut pas transmettre des vidéos à plus de 20 IPS pour une seule caméra. Il semble judicieux alors de ne pas privilégier la qualité des caméras à tout prix, car on ne pourra pas exploiter ces outils à leur plein potentiel à part via l'enregistrement sur disque dur, et même là on perdra en capacité d'enregistrement pure. L'implantation d'UNE SEULE caméra très haute définition pour rassurer les clients semble donc être une approche plus appropriée ici.

Module de correction GPS

- Il apparait après lecture du document présentant la mise en place du système de correction GPS que pour le mettre en place l'utilisateur doit se connecter au module navigator (?) via une connexion SSH. Or nous ne possédons pas le mot de passe pour se connecter ainsi au robot. Des explications supplémentaires sont requises pour savoir si on a besoin d'un mot de passe pour se connecter au robot, ou si la connexion doit se faire sur un module particulier du robot.

Disque dur

- Le disque dur intégré à Bobby n'est que de 60 Gb et non 1Tb (Pas très grave, mais à vérifier sur la nouvelle version à venir.)

Premier survol sur les réponses d'Alexey

Actualisation du manuel d'utilisateur (TBC-France_Doc_UserManuelFR_170711_02_ZE

Journal de bord 17/07/12

Débriefing du retour des questions/réponses d'Alexei : fichier spécifique à part

Processus de décision montré dans Mapperviewer (?)

Tentative d'accès au DVR via l'adresse indiquée sans succès

Le disque dur de 60 Gb n'est pas le lieu de stockage des vidéos : ce dernier se trouverait dans la tour

La PTZ n'est pas faite pour l'instant pour un contrôle manuel par joystick (Double clic droit sur l'image panoramique pour diriger la PTZ vers le point indiqué pdt 10 sec avec un temps de latence de 1 sec.

Test à venir de la capacité des RFID à augmenter la précision supposée du robot.

Impossible d'avoir une donnée chiffrée quant à la précision du robot (Quid des chiffres concernant la précision du robot donnés par SMP)

Tentative d'accès aux vidéos du robot via MvLauncher, (en grande partie infructueuse, car impossible de choisir une vidéo en particulier (une seule vidéo?) ou de sélectionner un segment)

Semaine 6 - Journal de bord 17/07/17

Meeting avec Mathilde

- Retour sur les meetings avec Samarte et Vehante
- Estimation de l'arrivée de la version 5.1
- Retour sur la relation avec RAD et SMP
- Planification de la semaine

Connection SSH sur Jack

Début d'un document de demande technique à SMP

Actualisation du manuel d'utilisateur (TBC-France_Doc_UserManualFR_170717_01_KE)

Journal de bord 17/07/18

Création d'un diagramme du fonctionnement du robot via Visio (TBC-France_Diagramme_Schéma_Fonctionnement_Robot_170718_01_ZE)

Création d'un diagramme séquentiel de la configuration du robot via Visio (TBC-France_Diagramme_SequenceOperationsRobot_170718_01_KE)

Actualisation finale de plusieurs tutoriels de logiciel

- MvLauncher (TBC-France_Logiciel_MvLauncher_170718_01_KE)

- TerrainLearner (TBC-France_Logiciel_TerrainLearner_170718_01_KE)
- StereoViewer (TBC-France_Logiciel_StereoViewer_170718_01_KE)
- RadioNavigator (TBC-France_Logiciel_RadioNavigator_170718_01_KE)
- VisualNavigator (TBC-France_Logiciel_VisualNavigator_170718_01_KE)
- WinViewer (TBC-France_Logiciel_WinViewer_170718_01_KE)

Actualisation du manuel d'utilisateur (TBC-France_Doc_UserManualFR_170718_03_ZE)

Journal de bord 17/07/19

Réunion concernant le Manuel utilisateur/ Certification à proposer au client.

Réunion concernant les questions à renvoyer à Alexei, les tests à effectuer, et le cahier des charges à envoyer à SMP.

Tournage de vidéo pour une vidéo promotionnelle du robot.

Journal de bord 17/07/20

Test de connexion via bus CAN, infructueux

Test de précision en conduite autonome avec Jack (document à joindre)

- Pilotage inertielle
- Pilotage inertielle + RFID

Journal de bord 17/07/21

Poursuite des Tests de Précision avec localisation inertielle, visuelle, et RFID.

Au bout de ces différents tests il en ressort une réelle amélioration de la précision du robot lors de son parcours (Bien que ce ne soit pas parfait). Durant ces tests, nous avons fait parcourir un chemin au robot ponctué par plusieurs 'portes' (portes formées par un duo de cônes ou un duo cône trottoir, 6 portes au total, d'une taille allant de 1.5 à 2m).

En lançant le robot uniquement avec son module de localisation inertielle, la précision n'était pas satisfaisante pour effectuer 2 tours au complet sans risques. Après revisionnage du feedback du robot via l'API WinViewer, il apparaît qu'une discontinuité dans le positionnement du robot a provoqué cette localisation mauvaise. Or ce genre de discontinuité est assez récurrent comme nous l'avons remarqué par la suite.

Après avoir fait un tour d'apprentissage visuel du même parcours le résultat est plus concluant. Le robot est capable de répéter le parcours 10 fois (comme nous l'avons prévu afin d'avoir un échantillon un tant soit peu représentatif des erreurs commises par le robot).

Ceci dit, au cours de ces 10 tours le robot a très vite perdu en précision (Visible dès le 3^{ème} tour). Le robot est alors capable d'effectuer le parcours en butant sur une seule difficulté comme de commettre 5 à 6 erreurs sur le même parcours au tour suivant.

Les 2 premiers tests concernant les RFID ne peuvent pas être considérés comme représentatifs pour des raisons techniques (Batteries RFID à plat dans certains cas)

Pour le 3^{ème} test nous avons placé les RFID dans un petit périmètre dans lequel le robot passait par moment (pour représenter un Docking charge). Ceci visait à vérifier l'utilisation de la localisation RFID lorsque le robot sortait légèrement du périmètre. On constate encore une baisse dans le nombre d'erreurs commises par le robot. Ce chiffre ne dépasse plus 3, et on constate assez souvent (11/13) que le robot ne rate aucune porte dans cette configuration (Le robot bute sur une seule porte durant le parcours, mais n'en rate aucune).

Lors du 4^{ème} test nous avons agrandi le périmètre de RFID pour déterminer si la précision augmentait de manière significative. En effet le robot continue d'hésiter devant certaines portes, mais il n'en rate plus.

Ceci dit durant ces tests le robot à parfois eu une tendance très inquiétante à effectuer des choix aberrants surtout au niveau de la correction de sa trajectoire après détection d'obstacles, et de son positionnement.

Durant l'intégralité de ces tests, nous avons également fait des captures vidéo du feedback du robot sur l'ordinateur de l'utilisateur (WinViewer). En complément des conclusions de ces tests, nous repasserons au début de la semaine prochaine sur ces vidéos afin d'en tirer plus d'informations.

Semaine 7 - Journal de bord 17/07/24

Poursuite de l'étude des vidéos de feedback des tests. Il apparait que le robot subit régulièrement des discontinuités dans son positionnement.

D'où le caractère aléatoire des erreurs commises par celui-ci. Ce problème ne disparaît pas avec l'ajout de modules supplémentaires (RFID) ce qui indique que ces discontinuités sont le fruit de la correction de la position du robot.

Problème, le phénomène se produit sur des parcours très courts et les erreurs tendent à se reproduire très vite, ce phénomène risque donc de s'accroître lors de rondes prolongées.

Journal de bord 17/07/25

Poursuite des tests de précision RFID. Le système d'identification du type de terrain configuré, TerrainLearner a été jumeler a WinViewer et RadioNavigator. (TBC-France_RapportTest_170725_01_KE)

- Tentative de test avec chemin programmé via le PC de commande, pour réduire la marge d'erreur commise par le robot lors du premier parcours effectué en manuel. Test non concluant (Erreur de position pendant le parcours, ou erreur humaine de positionnement des bornes sur le pc de contrôle).
- Tentative d'apprentissage des différents types de terrain présents. Influence encore à déterminer, problème, difficulté pour distinguer un certain nombre de textures avec la luminosité présente et les caméras à disposition. (Pelouse jaunie, asphalte au soleil, ombres et arbustes.....). Ceci dit, le robot n'a jamais traversé un des trottoirs présents durant les tests, comme cela était prévu au départ. Des tests supplémentaires uniquement dédiés aux réponses du robot en fonction du terrain seront nécessaires.

Actualisation du document de planification de tests. (TBC-France_Campagne de tests_170725_02_KE)

Actualisation du manuel d'utilisateur. (TBC-France_Doc_UserManualFR_170725_01_KE)

- Configuration de la table de matière
- Ébauche des réponses de la section FAQ

Formatage du DVR de MV Launcher. Tentative de récupération des vidéos depuis le DVR via MV Launcher mais, impossible de récupérer des fichiers durant + de 35 min.

- Manip effectuée : MV Launcher/Settings/Archive/Save Video Fragment (Bouton pellicule de film)
Déterminer la date/heure de début et de fin.
- Mais l'API plante trop régulièrement pour être utilisée telle qu'elle. (Besoin de plus d'infos de la part des développeurs).
- D'autre part, le flux récupéré ne concerne qu'une seule caméra à la fois et peut être directement obtenue dans un fichier .avi (lisible par la plupart des logiciels de lecture vidéos).
- En revanche, il apparait qu'après formatage il est plus simple d'accéder aux vidéos depuis l'application Multivision.
- (Formatage : MV Launcher/Settings/Archive/Drive/Format internal drive. L'opération n'est pas instantanée et peut nécessiter une dizaine de minutes jusqu'à complétion)
- Dernière chose les vidéos récupérées ne sont pas en HD (Deux possibilités, caméras non HD, ou compression trop importante)

Poursuite de l'étude des vidéos de feedback des tests.

Journal de bord 17/07/26

Poursuite des tests de précision RFID. Tentative de test de correction RFID après avoir retiré les bornes (Dès le départ, après un parcours manuel, après l'apprentissage visuel). Ces tests nous indiquent qu'il n'y a pas de correction RFID a proprement dit une fois les balises retirées. Test supplémentaires prévus demain.

Actualisation du manuel d'utilisateur. (TBC-France_Doc_UserManualFR_170726_02_ZE)

- Configuration de la table de matière
- Correction

Documentation du logiciel MapperViewer (TBC-France_Logiciel_MapperViewer_170726_01_ZE)

Actualisation de la documentation concernant TerrainLearner (TBC-France_Logiciel_TerrainLearner_170726_01_ZE)

Journal de bord 17/07/27

Nouveaux tests de fonctionnement des RFID, cette fois ci le parcours a été appris et répété plusieurs fois avec les RFID avant de retirer ces dernières. Ce test n'a pas été concluant, le robot est toujours sujet à des imprécisions très gênantes (le menant proche des clôtures parfois).

Autre hypothèse envisagée : La correction RFID entraîne des discontinuités pour corriger la position erronée du robot. Une autre tentative a été effectuée dans laquelle les RFID étaient installées puis désactivées (via RadioNavigator) pendant l'apprentissage, et réactivées par la suite pour vérifier cette hypothèse. Malheureusement le robot a fait face aux mêmes difficultés que précédemment ce qui invalide cette théorie pour le moment. (TBC-France_RapportTest_170727_01_ZE)

Réception du matériel nécessaire à l'implémentation du système de correction GPS.

Ébauche du manuel d'utilisateur en anglais. (TBC-France_Doc_UserManualENG_170727_01_KE)

Démontage de la base station et tentative de connexion d'un câble CAT5 – échec.

Semaine 8 - Journal de bord 17/07/31

Configuration du point d'accès extérieur Ubiquiti Rocket M2

Configuration du routeur Asus

Reverse engineering de la tour PTZ

Actualisation du manuel d'utilisateur en anglais (TBC-France_Doc_UserManualENG_170731_01_KE)

Journal de bord 17/08/01

Configuration complète du routeur ASUS, et de l'Ubiquiti pour créer le réseau sur lequel vont opérer les robots.

Tentative de connexion des robots à ce réseau.

Journal de bord 17/08/02

Configuration de la base GPS, Fin de la configuration du réseau des robots. Connexion à ce réseau.
Changement des adresses IP de connexion des robots (cf document associé).
Tentative de mise en fonctionnement du système GPS complet.

Journal de bord 17/08/03

Tentative de mise en fonctionnement du réseau complet. Il apparait que le robot ne peut pas communiquer avec la base GPS en l'état. Recherches et questionnement de SMP pour résoudre ce problème.

Actualisation du manuel d'utilisateur en anglais (TBC-France_Doc_UserManualENG_170803_01_KE)

Réunion avec un représentant de chez Génétéc pour présenter les solutions apportées. Décision d'embarquer un PC de contrôle pour y installer Génétéc et les futurs capteurs additionnels. Changement prévu des caméras panoramiques et de la PTZ pour choisir des modèles compatibles avec Génétéc. Coût supplémentaire sur les robots environ 3K€ chacun.

Journal de bord 17/08/04

Nouvelle tentative de communication entre la base GPS et le robot à la lumière des éléments fournis par SMP. Choix de PC à embarquer sur le robot.

Retour sur les réponses d'Alexei non concluant, décision de changer de système GPS et de délaisser le Tral 5 POE (<http://ts-market.com/upload/iblock/7c3/Tral5%20Catalogue%20File%20.pdf>)

Recherche de Flight Case pour demo de jeudi prochain.

Semaine 9 - Journal de bord 17/08/07

Commande de trois Pelicase 1730 pour la demo de jeudi/demo en attendant le camion de visionnement

Perte de la connection GPS

- Impossibilité de rejoindre le GPS par câble ou par WIFI.
- Reconfiguration du routeur – échec.
- Débranchement du GPS – échec.

Journal de bord 17/08/08

Retour sur les réponses d'Alexei – troubleshooting de la station GPS

Réception des valises Pelicase 1730

Journal de bord 17/08/09

Retour sur les réponses d'Alexei – troubleshooting de la station GPS

Aménagement des valises Pelicase pour demo à Sécuritas

Meeting pour Sécuritas

Journal de bord 17/08/10

Déplacement Aix → Lyon

Demo Sécuritas

Déplacement Lyon → Aix

Journal de bord 17/08/11

Débriefing de la démo chez Sécuritas à Lyon

Établissement d'une check list à vérifier avant chaque démo

Recherche d'une caméra PTZ munie d'un dispositif de tracking pour avoir une idée de la gamme de prix.

Choix d'un PC durci (processeur Intel I3, conso globale 60 W) à essayer par la suite.

Semaine 10 - Journal de bord 17/08/14

Compte rendu de la discussion avec Steve

- Système de navigation de la 5.1 - performant
- Batterie de la 5.1 – changé
- Caméras – changé

Connection avec la base GPS rétablie

Test de « Route Scheduling » - échec

Ébauche d'un document réponse à Alexei – questionnement GPS

Journal de bord 17/08/15

Jour férié

Journal de bord 17/08/16

Tentative de configuration des Ubiquitis avec les fichiers de configuration que nous a envoyé Alexei

Tentative de port forwarding pour connecter la base GPS

Visite de SVD pour des choix de caméras compatibles Génétec.

Tentative de port forwarding sur le Rocket et de reconfiguration des Ubiquitis (Wan=WLAN, Lan=LAN)

Pour une raison inconnue, on peut pinguer La base GPS depuis le robot, mais on ne peut plus accéder au web server de la rocket. (A approfondir ce jeudi)

Journal de bord 17/08/18

Test de détection de mouvement

- Porté d'environ 20 mètres
- Incapable d'identifier comment le système détermine quel est le mouvement le plus important
- Habilité à identifier dans le noir – inconnu

Test de navigation satellite

- Augmentation des discontinuités remarquée
- Localisation du satellite déroutante

Journal de bord 17/08/19

Ébauche d'un document réponse à Alexei – questionnement GPS

Configuration de la nouvelle base de données Dropbox

Semaine 11 - Journal de bord 17/08/14

Test du routeur Cradlepoint et de la carte SIM d'Orange

Annexe B

Caractéristiques du robot

Robot : S5 PTZ – Bobby & Jack

Spécifications

PROVIDER : http://smprobotics.com/products_autonomous_ugv/

WIFI DISTANCE : 1000 ft

WIFI PROTOCOL : 802.11a/b/g/n

WIFI FREQUENCIES : 2.5, 5.0 GHz

SPEED OF AUTOPILOT MOVEMENT : 2.5 mph

OPTIMAL AUTOPILOT ROUTE WIDTH : 3 ft

OPTIMAL TURNING RADIUS : 17 ft

TEMPERATURE RANGE : -4 : +113 F

RUN TIME : 12 hours

CHARGING TIME : 4 hours / 4h30 hours

BATTERY : 2 x 12V, Lead Acid 100 A

HUMAN DETECTION OF PTZ: 160 ft

COMPRESSION : H.264

BUILD IN HDD : 1 TB

DVR RECORDING TIME : 4 weeks

Code couleur antenne LED

- Aucune lumière : Mode Manuel
- Orange : Mode Automatique

Code couleur frontal LED

- Aucune lumière : Mode Manuel
- Orange : Mode Automatique – Apprentissage
 - o Clignotement : Réception/Analyse d'information
- Vert : Mode Automatique – Reproduction
 - o Clignotement : Réception/Analyse d'information
 - o Inclinaison : Direction

Code couleur bouton LED

- Vert : Système opérationnel
- Blue : Système en recalibrage
- Rouge : System non opérationnel / défectueux

Fonctionnalité bouton LED

- Bouton « Signal » : Signal Wifi.
- Bouton « A » : Signal Autopilot.
- Bouton « Flowchart » : Signal Internal Switch.
- Bouton « Batterie Double » : Signal Steréo Caméras.
- Bouton « Batterie Flèche » : Signal Visual Navigation.
- Bouton « Camera » : Signal PTZ Caméra.
- Bouton « t c » : Signal Température Inside.
- Bouton « GPS » Signal GPS – Position du robot sur la route actuelle.

- Bouton « Rouleau » : Signal Navigation Inertielle.
- Bouton « P Signal » : Signal Ultrasonic Sensors.
- Bouton « Réglages » : Signal Low Level Devices.
- Bouton « Pied De Poulet » : Signal RFIDs.

Fonctionnalité clef

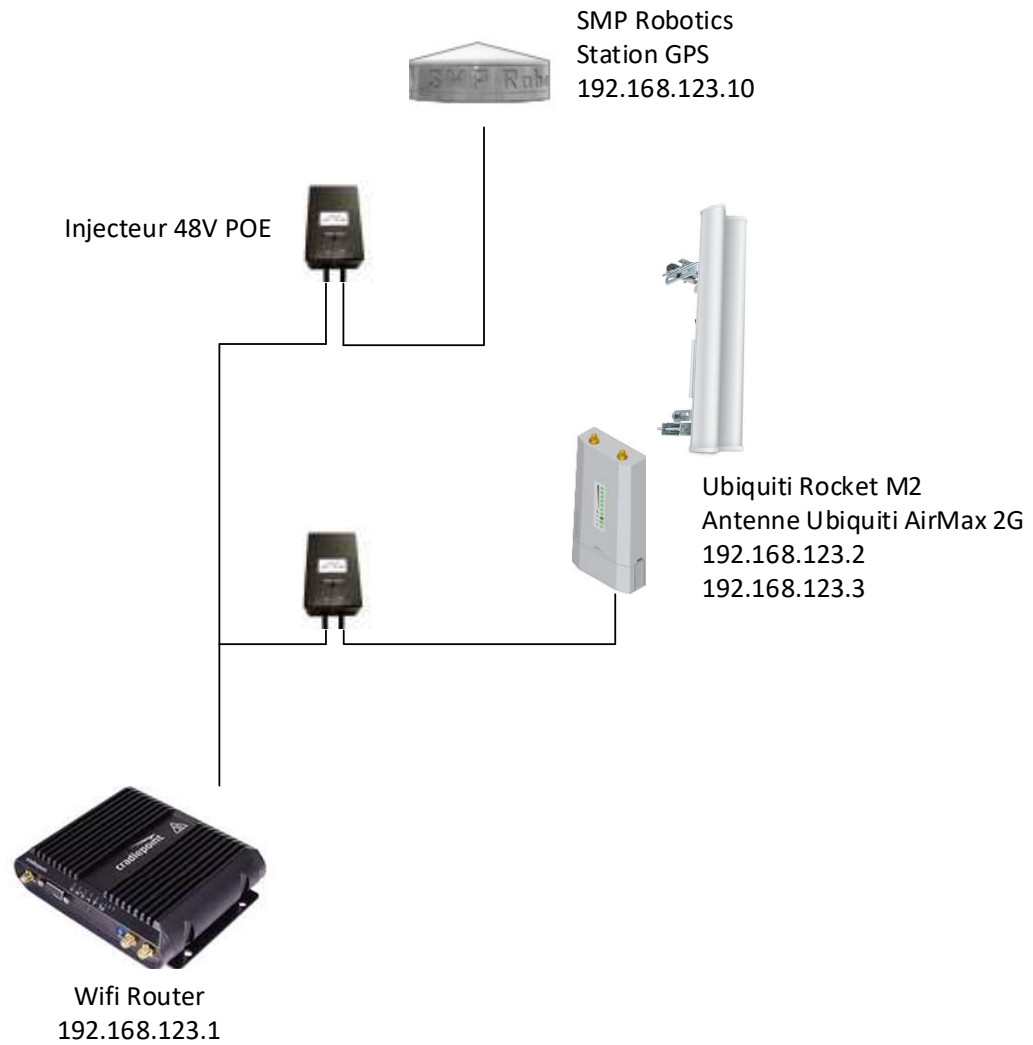
- Rotation clef « sens horaire » : Mode « charge » - Alimentation des circuits internes ; Fonction Autopilot, Payload Power Supply, Dispatcher Board, Wi-Fi Bullet, Switch. Possibilité de faire le diagnostic du robot et de mettre à jour le système opérationnel du robot.
- Rotation clef « neutre » : Mode inactif – point mort total.
- Rotation clef « sens antihoraire » : Mode actif – robot totalement fonctionnel.

Annexe C

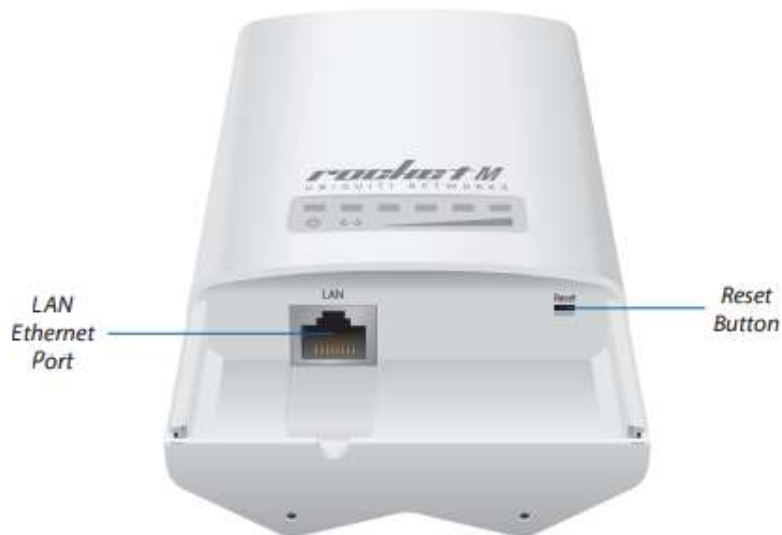
Configuration de la base GPS

Base Wifi – GPS








→ A modifier si nécessaire



Ubiquiti Rocket M





-  **Power** The Power LED will light green when the device is connected to a power source.
-  **Ethernet** The Ethernet LED will light steady green when an active Ethernet connection is made and flash when there is activity.
-  **Signal** In airOS, you can modify the wireless signal strength threshold values for each LED on the *Advanced* tab under *Signal LED Thresholds*. The default values are shown below:
-  LED will light red when the wireless signal strength is above -94 dBm.
 -  LED will light amber when the wireless signal strength is above -80 dBm.
 -  LED will light green when the wireless signal strength is above -73 dBm.
 -  LED will light green when the wireless signal strength is above -65 dBm.

Spécifications

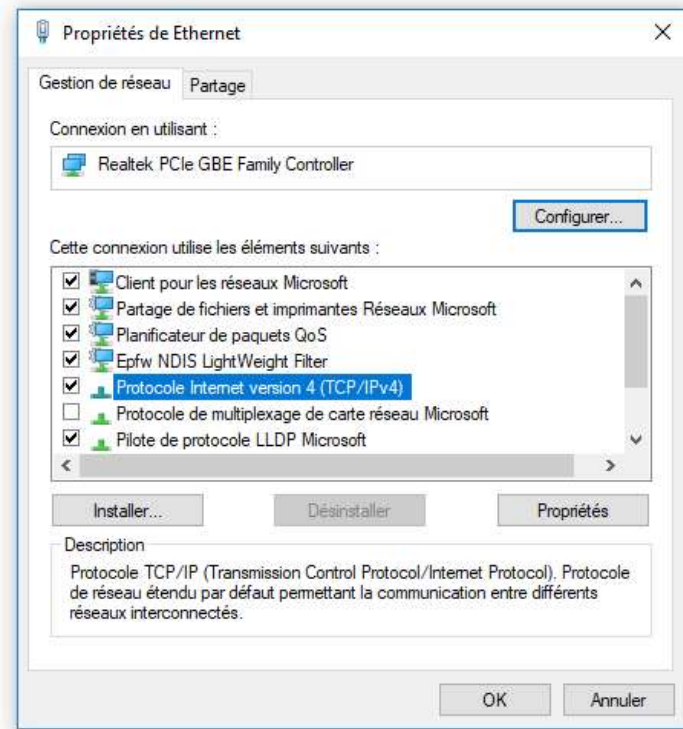
RocketM	
Dimensions (Mount Included)	160 x 80 x 30 mm (6.30 x 3.15 x 1.18")
Weight (Mount Included)	500 g (1.1 lb)
Port	(1) 10/100 Ethernet Port
RF Connectors	(2) RP-SMA (Waterproof)
Enclosure	Outdoor UV Stabilized Plastic
Max. Power Consumption RocketM900/2/3/365 RocketM5/6	6.5W 8W
Power Supply	24V, 1A PoE Adapter (Included)
Power Method	Passive PoE (Pairs 4, 5+; 7, 8 Return)
Certifications	CE, FCC, IC
Operating Temperature	-30 to 75° C (-22 to 167° F)
Operating Humidity	5 to 95% Noncondensing
Shock and Vibration	ETSI300-019-1.4

RocketM900/M2/M3/M365/M6 Operating Frequency (MHz)	
RocketM900	902 - 928
RocketM2	2402 - 2462
RocketM3	3370 - 3730*
RocketM365	3650 - 3675
RocketM6	5955 - 6400

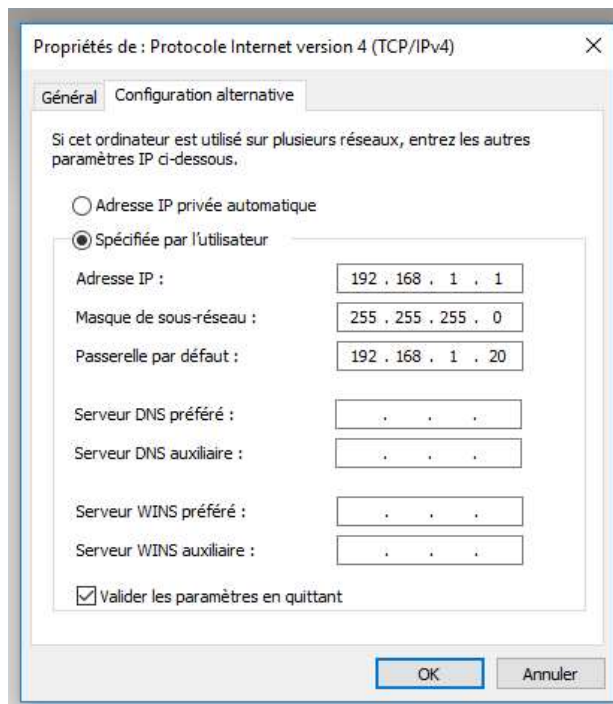
Configuration du PC :

Panneau de configuration

Propriétés de l'éthernet




Configuration TCP/IPv4



1^{ère} Connexion Air OS

Non sécurisé | <https://192.168.1.20/login.cgi?uri=/>



Username:

Password:

Username : ubnt

Password : ubnt

Paramétrage :

rocket M2 TITANIUM **air OS**

MAIN WIRELESS NETWORK ADVANCED SERVICES SYSTEM Outils : Se déconnecter

Paramètres sans fil de base

Mode :

WDS (Mode pont transparent) : Activer

SSID :

Verrouiller à la MAC PA :

Code pays :

Mode IEEE 802.11 :

Largeur de bande :

Décalage de canal :

Liste des fréquences scannées, MHz : Activer

Ajustement auto limite PIRE : Activer

Gain antenne : dBi Pertes du câble : dB

Puissance de sortie : dBm

Data Rate Module :

Débit TX max, Mbps : Automatique

sécurité

Securité :

 © Copyright 2006-2013 Ubiquiti Networks, Inc.

Modifier :

Code Pays : France

Gain antenne: 15 (AirMax)

Puissance de sortie : 20dB

Network Role

Network Mode: Bridge
Disable Network: None

Configuration Mode

Configuration Mode: Simple

Management Network Settings

Management IP Address: DHCP Static
IP Address: 192.168.1.20
Netmask: 255.255.255.0
Gateway IP: 192.168.1.1
Primary DNS IP:
Secondary DNS IP:
MTU: 1500
Management VLAN: Enable
Auto IP Aliasing: Enable
STP: Enable

Change



Configuration : Base GPS

La base possède deux adresses IP :

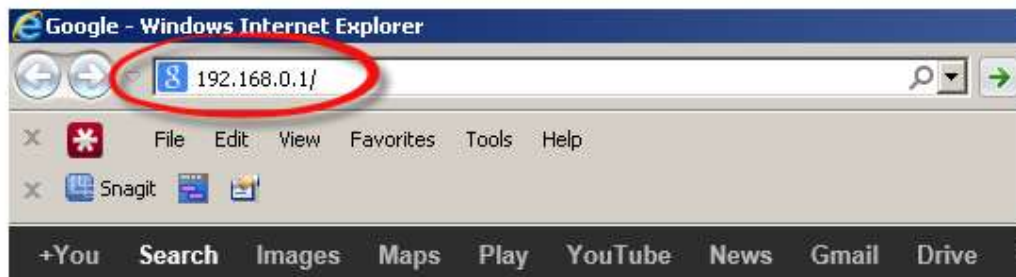
- Une adresse IP de classe C, réglée par l'utilisateur et qui peut ne pas être disponible.
- Une adresse de classe B dans la catégorie zeroconf (169.254.x.x), qui elle est toujours disponible et invariante.

Pour accéder à l'adresse zeroconf :

- brancher le GPS directement en ethernet au PC
- régler l'adresse IP de ce dernier sur la plage zeroconf (169.254.x.x)
- Lancer Multivision, et appuyer sur le bouton find
- Cliquer sur l'adresse qui apparait comme étant connectée (fond bleu texte blanc)

Configuration : ASUS AC87U

Adresse IP par défaut : 192.168.1.1



Enter Network Password

Please type your user name and password.

Site: 192.168.0.1

User Name

Password

Save this password in your password list

OK Cancel

Login : admin

Password : admin

Asus RT-AC87U Usernames	Asus RT-AC87U Passwords
admin	admin

Connexion au GPS après configuration du routeur.

ATTENTION : Lors de la configuration du routeur, ne pas sécuriser la connexion en demandant un mot de passe avant de connecter la base GPS. Sélectionner la méthode d'authentification 'système ouvert' (ou équivalent selon le routeur) !! Après la connexion de

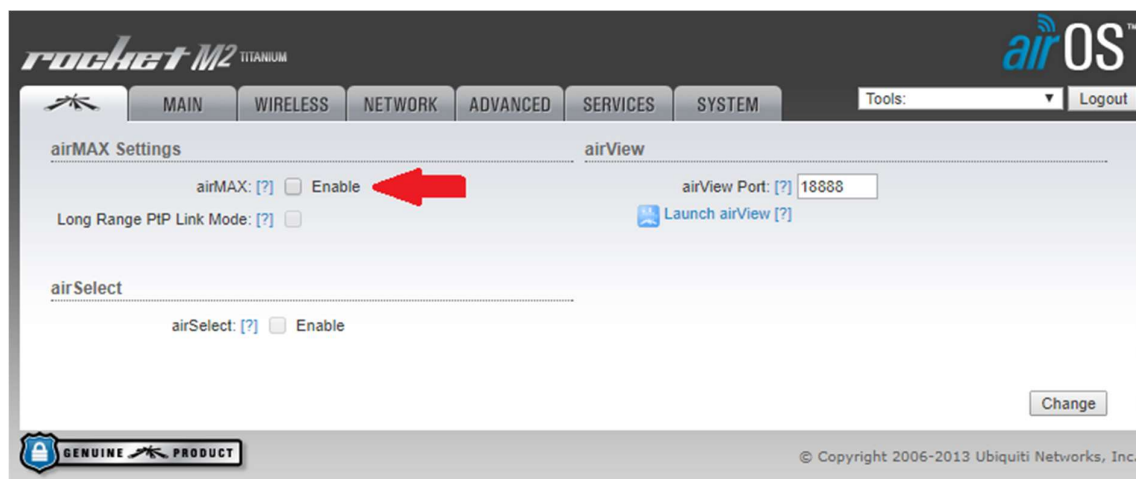
la base vous pouvez reconfigurer la sécurité du routeur.

Configuration de l'Ubiquiti Rocket M2

Eviter de laisser cette étape à une personne non qualifiée. Ne pas oublier d'appuyer sur 'change' (en bas de page) pour entériner le changement des paramètres, puis sur 'test' pour tester les paramètres durant 3 min (utile pour éviter les fausses manip menant à une perte d'accès au webserver), ou 'apply' pour appliquer les changements de manière définitive.

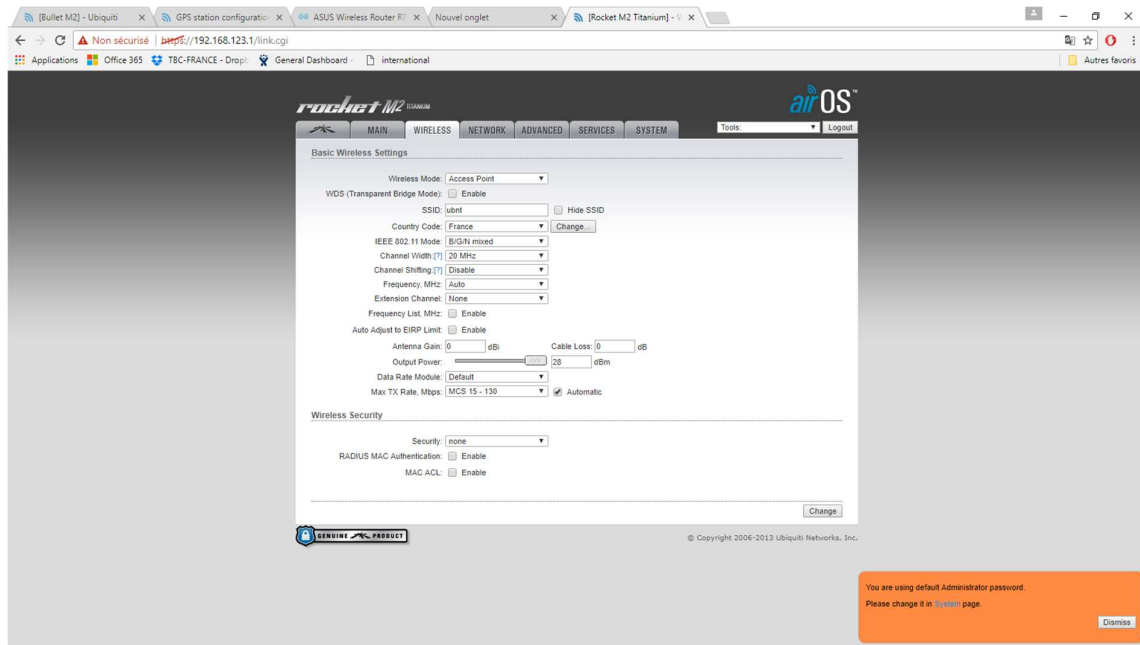
Tout d'abord connectez-vous à l'Ubiquiti Rocket via l'adresse 192.168.1.20

Onglet Ubiquiti



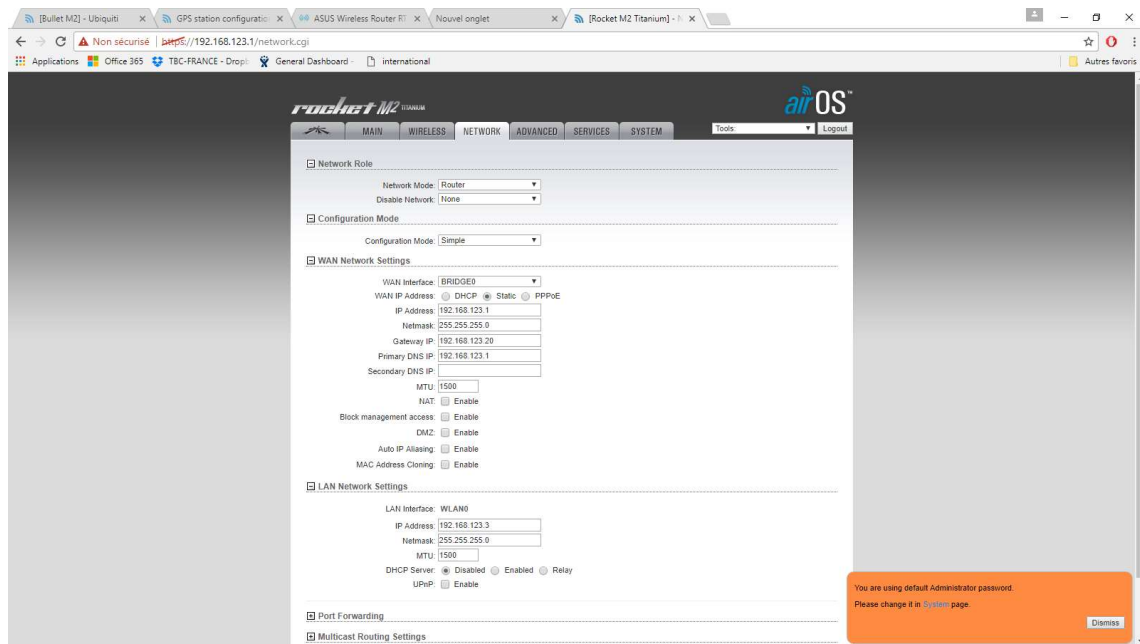
Dans l'onglet du logo Ubiquiti, décocher la case 'airMAX' (indiquée par la flèche rouge) si celle-ci est cochée pour pouvoir se connecter au réseau Wifi créé par l'Ubiquiti. (Cela permettra de changer les paramètres par la suite avec moins de risques d'erreur fatale, et est indispensable pour permettre la connexion des robots.) Ensuite accéder à l'onglet 'Wireless'.

Onglet Wireless



Régler l'Ubiquiti en mode point d'accès, et changer la largeur de chaîne à 20 Mhz (La page finale après les réglages doit être identique à l'image ci-dessus.) Si vous le souhaitez modifiez les réglages de sécurité de votre réseau. Accédez à l'onglet Network.

Onglet Network



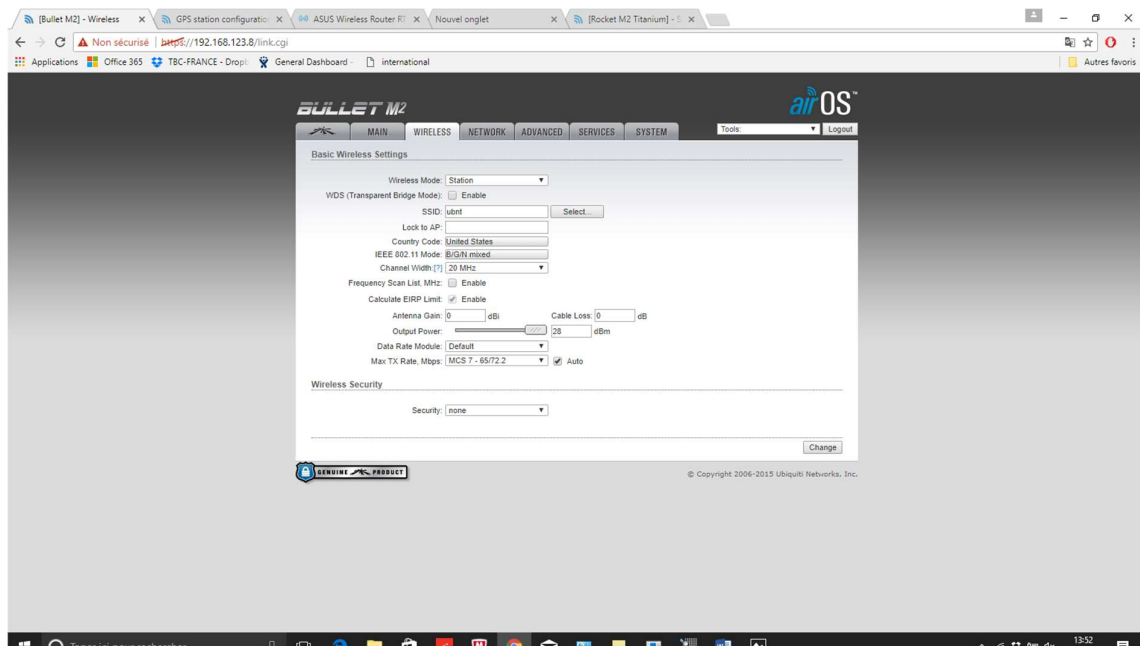
Régler les paramètres réseaux tels qu'indiqué ci-dessus (Adapter les adresses IP en fonction du subnet que vous utilisez pour le GPS, et des adresses IP que vous voulez utiliser). Ceci va régler les

adresses IP auxquelles vous pourrez vous connecter pour continuer à avoir accès à ce webserver. (Attention au moment de confirmer les changements ('apply'), passez d'abord par une phase de test ou les changements seront appliqués pendant environ 3 min, pour vérifier que vous avez toujours accès au webserveur. Si vous n'arrivez pas à accéder au webserveur, les changements seront remis à l'état d'avant test, si l'utilisateur ne confirme pas les changements.

Ubiquiti du robot

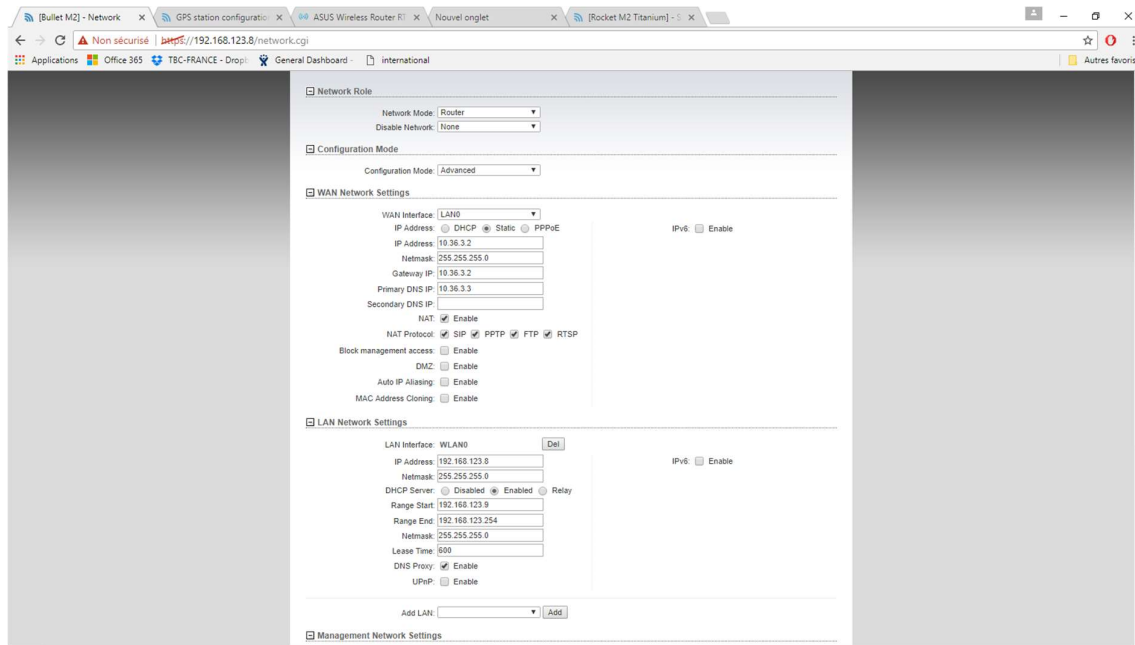
Une fois ces étapes effectuées, connectez-vous à l'Ubiquiti du robot, et procédez aux modifications suivantes.

Onglet Wireless



Passez l'Ubiquiti en mode station, et recherchez le réseau créé par l'Ubiquiti Rocket M2 que vous avez configuré précédemment. N'oubliez pas de préciser le mot de passe associé à ce réseau si vous en avez créé un (Sous-section Wireless Security). L'image ci-dessus vous montre la page que vous devriez obtenir après ces modifications.

Onglet Network



Puis dans l'onglet Network, modifiez les paramètres d'adresses IP de façon à obtenir l'image ci-dessus (en adaptant votre subnet pour le réseau WLAN au subnet utilisé par la base GPS).

Si la configuration des deux Ubiquitis a été correctement exécutée, vous devriez pouvoir accéder à toutes les fonctionnalités du robot en vous connectant sur le réseau crée par l'Ubiquiti Rocket (ici ubnt).

Problème de connexion

Si toutefois il vous était impossible de vous connecter aux robots sur votre PC. Vérifiez les adresses IP et les ports dans l'onglet Network, sous-section Port Forward dans le webserver de l'Ubiquiti embarqué sur le robot (ici 192.168.123.8).

Enabled	Interface	Private IP	Port	Type	Source IP/Mask	Public IP/Mask	Port	Comment	Action
<input checked="" type="checkbox"/>	WLAN0	10.36.3.199	5049	TCP	0.0.0.0/0	192.168.2.2/32	5049		Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.199	22	TCP	0.0.0.0/0	192.168.2.2/32	22199		Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.200	5050	TCP	0.0.0.0/0	192.168.2.2/32	5050		Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.200	22	TCP	0.0.0.0/0	192.168.2.2/32	22200		Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.200	5000	TCP	0.0.0.0/0	192.168.2.2/32	5000		Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.45	21784	TCP	0.0.0.0/0	192.168.2.2/32	21784		Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.45	22	TCP	0.0.0.0/0	192.168.2.2/32	22045		Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.45	21774	TCP	0.0.0.0/0	192.168.2.2/32	21774		Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.46	21784	TCP	0.0.0.0/0	192.168.2.2/32	21785		Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.46	22	TCP	0.0.0.0/0	192.168.2.2/32	22046		Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.199	21701	TCP	0.0.0.0/0	192.168.2.2/32	21701		Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.66	11111	TCP	0.0.0.0/0	192.168.2.2/32	11111		Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.66	21311	TCP	0.0.0.0/0	192.168.2.2/32	21311		Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.66	22	TCP	0.0.0.0/0	192.168.2.2/32	22066		Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.42	22	TCP	0.0.0.0/0	192.168.2.2/32	22042		Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.42	26702	TCP	0.0.0.0/0	192.168.2.2/32	26702		Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.42	26703	TCP	0.0.0.0/0	192.168.2.2/32	26703		Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.42	26704	TCP	0.0.0.0/0	192.168.2.2/32	26704		Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.42	26705	TCP	0.0.0.0/0	192.168.2.2/32	26705		Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.42	26707	TCP	0.0.0.0/0	192.168.2.2/32	26707		Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.42	26709	TCP	0.0.0.0/0	192.168.2.2/32	26709		Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.45	21764	TCP	0.0.0.0/0	192.168.2.2/32	21764	Video front camera	Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.46	21764	TCP	0.0.0.0/0	192.168.2.2/32	21765	Video rear camera	Edit Del
<input checked="" type="checkbox"/>	WLAN0	10.36.3.199	23232	TCP	0.0.0.0/0	192.168.2.2/32	23232	Cam/Atom	Edit Del
<input checked="" type="checkbox"/>	LAN0	192.168.123.10	5055	TCP	0.0.0.0/0	192.168.2.2/32	5055	GPS?	Edit Del

Les adresses IP publiques doivent toutes être dans le même subnet d'adresses IP que l'Ubiquiti Bullet (192.168.123.x ici par exemple).

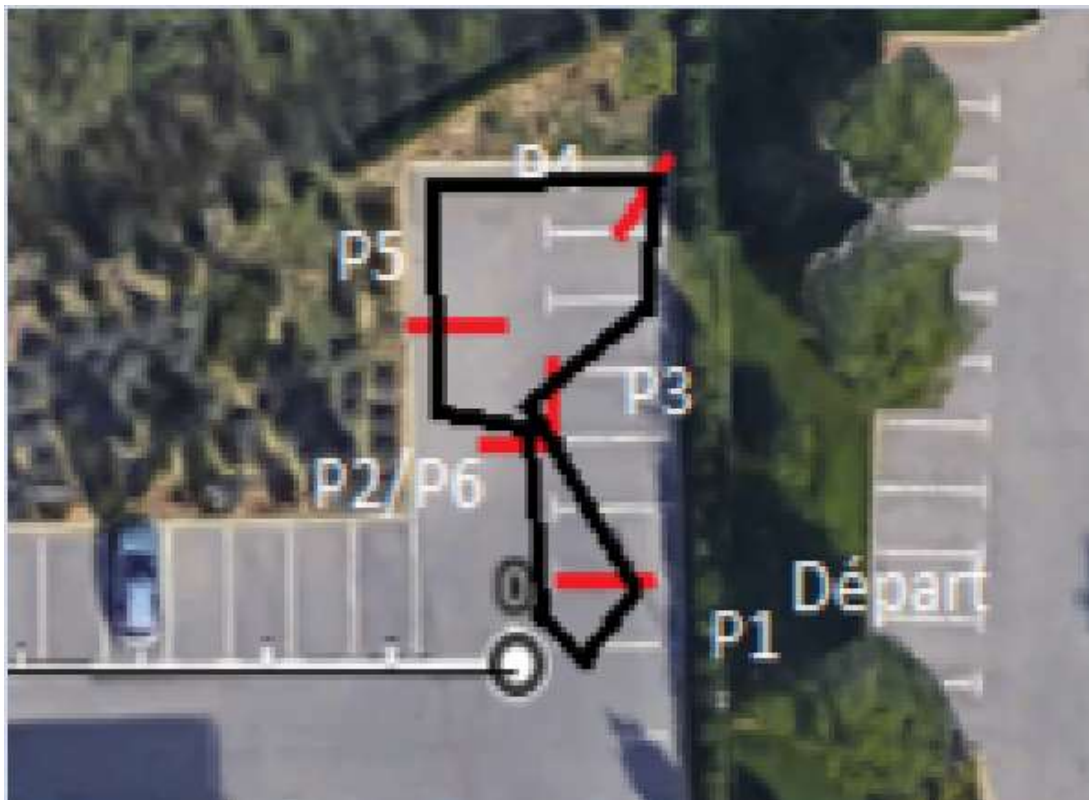
Annexe D

Rapports de tests

Test RFID - 17/07/21

Légendes

- R : Recalcul l'angle nécessaire pour passer la porte.
- H : Hésitation momentanée mais aucune déviation de trajectoire.
- C : Collision d'un « cadre » de la porte.
- RAS : Rien à signaler, parcours sans faute.
- X : Conduite hors route, perte de signal



Tour	Rapport
1	R - P2
2	H - P3
3	H - P2, R - P3
4	R - P3, R - P4
5	H - P3
6	H - P1
7	RAS
8	H - P3
9	R - P6
10	R/H - P3, C - P5



Tour	Rapport
1	C - A2, X
2	C - A2, R A6, X
3	RAS
4	RAS
5	R - A5, R - A6
6	H - A5, R - A6
7	R - A6
8	R - A5, C - A2, R - A6
9	C - A2, X
10	C - A2

Test RFID/TerrainLearner - 17/07/25

Commentaire : Reprise du test RFID de la semaine dernière sur un parcours semblable avec une configuration additionnelle de paramètres sur TerrainLearner.

Légendes

- R : Recalcul l'angle nécessaire pour passer la porte.
- H : Hésitation momentanée mais aucune déviation de trajectoire.
- C : Collision d'un « cadre » de la porte.
- RAS : Rien à signaler, parcours sans faute.
- X : Porte ratée.
- ? : Emplacement aléatoire

Tour	Rapport
1	H - A2, H/R - Z
2	H - A5, H/R - Z (60s)
3	H - ?, H - A5, H - ?, H/R - Z (60s)
4	H/R - Z
5	X, H/R - Z (180s)
6	H - ?, H - A5, H/R - Z
7	X, H/R - Z
8	H/R - Z
9	X, H/R - Z
10	RAS

Retour :

1. Le robot est désaxé et dévie vers la gauche. Ce fait doit être à la source du problème avec la bordure dans la zone « Z ».
2. Définir la bordure comme zone restreinte avec TerrainLearner à aider à minimiser la conduite hors route.



Commentaire : Reprise du même parcours avec retouche sur la console au niveau des points de relais.

Tour	Rapport
1	X, H - Z
2	RAS
3	RAS
4	H/R - Z
5	X, H - Z
6	H/R - Z
7	H - ?, H/R - Z
8	X, H/R - Z
9	RAS
10	H/R - Z

Retour :

1. Le robot est désaxé et dévie vers la gauche. Ce fait doit être à la source du problème avec la bordure dans la zone « Z ».
2. 3 parcours sans faute comparer à un seul précédemment.



Commentaire : Création d'un parcours plus complet (3 passages à travers la porte) à l'aide du même environnement. L'ensemble des points de relais ont été posés via WinViewer. Cette approche est censée permettre de réduire les erreurs dans le parcours potentiellement causées par des discontinuités dans le positionnement durant le parcours d'apprentissage du robot effectué en manuel.

Tour	Rapport
1	X^2
2	X^2
3	X^2
4	X^2
5	X^2
6	$X^2, H - ?$
7	$H/R - ?, X^2$
8	X^2
9	$H - A5^1$
10	X^1, X^2

Retour :

1. S'éloigner de la bordure a aidé à éviter les problèmes de déviation précédent.
2. Il est fort probable que les problèmes concernant « X^2 » soit lié à un point de relais mal placé via console.



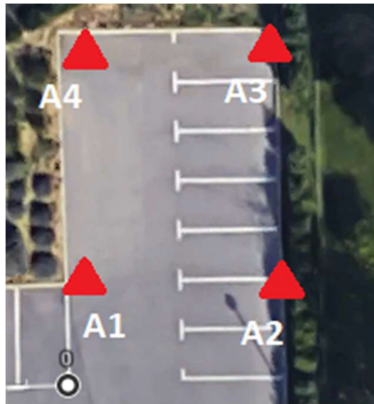
Test RFID n°4

Tentative avec positionnement puis retrait des RFID (tel que nous l'a indiqué Alexei)

Positionnement des RFID autour du parcours du robot

Puis plusieurs approches utilisées :

- Retirer immédiatement les RFID
- Effectuer le parcours manuellement et retirer les RFID
- Effectuer le parcours manuellement puis l'apprentissage et ensuite retirer les RFID



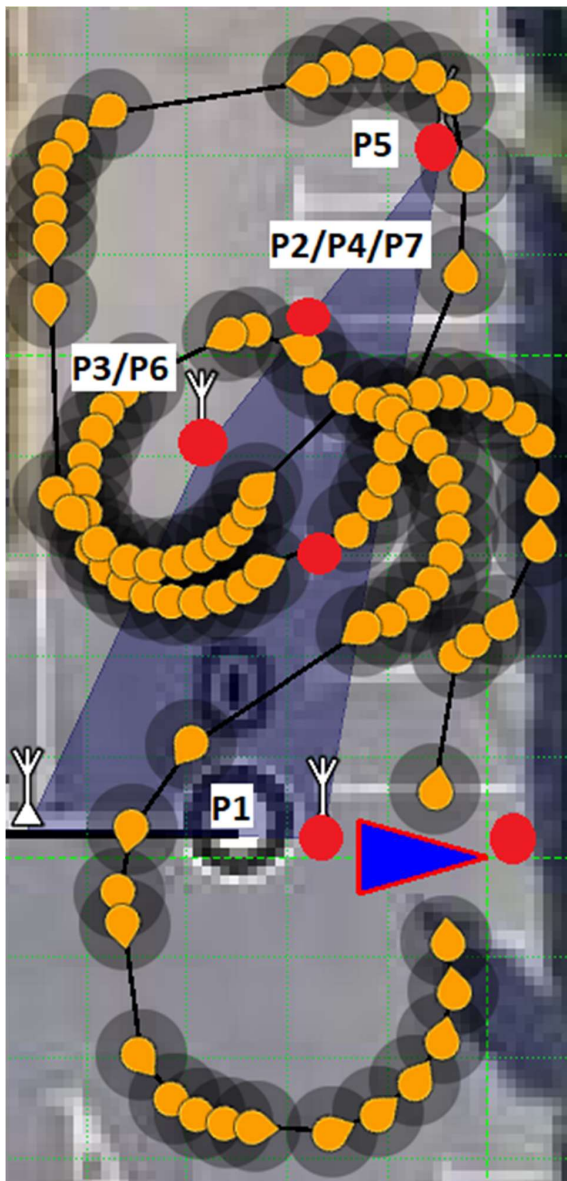
En observant sur le logiciel RadioNavigator engineer, on peut visualiser si le module de RFID effectue des corrections pendant le parcours. Aucune de ces trois approches ne nous a montré de corrections effectuées par les RFID.

Il est possible que le module doive être actif pendant un certain nombre de tours avant d'être effectif après avoir retiré les bornes. Nous effectuerons donc un test ultérieurement où nous ferons parcourir 10 tours au robot avec les corrections RFID, puis 10 autres tours après avoir retiré ces mêmes RFID.

Test RFID - 17/07/27

Procédure

Mise en place des bornes RFID puis parcours de 10 tours en mode autonome du robot. Après ces 10 tours enlèvement des bornes et à nouveau parcours de 10 tours du même parcours. (Le déroulement du test a fait que nous avons effectué 7 tours avec RFID et 5 sans, pour des raisons de sécurité et des similitudes dans les erreurs à chaque tour)



TBC-France S.A.S.

57 Esplanade du Général de Gaulle, 92081 Paris La Défense CEDEX | SIRET 825087919 / 00015
www.tbc-france.com | requete@tbc-france.com | +33 (0)6 70 76 93 40

Nb : Nous avons modifié la position de certains points notamment aux alentours de P6. En effet, lors du parcours manuel, nous avons fait passer le robot trop près du cône ce qui expliquait les erreurs dans cette zone.

Légendes

- R : Recalcul l'angle nécessaire pour passer la porte.
- H : Hésitation momentanée mais aucune déviation de trajectoire.
- C : Collision d'un « cadre » de la porte.
- RAS : Rien à signaler, parcours sans faute.
- X : Conduite hors route, perte de signal

Tour	Rapport
1	R – P3, P6*
2	R – P3, P5
3	R – P3
4	R – P3
5	X – P5
6	X – P5 R – P3
7	X – P3

Après ces parcours, nous avons donc retiré les bornes RFID, pour vérifier si l'affirmation faite par Alexei est bel et bien vraie (Les RFID ne doivent pas être constamment sur le terrain pour fonctionner)

Tour	Rapport
1	R – P3 H – P3
2	R – P3
3	R – P3
4	R – P3
5	R – P3

Conclusions : Le robot est dans l'ensemble moins précis sans les bornes RFID, mais son comportement semble moins erratique sans ces dernières. (Les recalculs en P3 étaient plus large, mais plus de passage dans la pelouse droit vers la clôture en P5). Des problèmes de discontinuité sont apparus lors de l'apprentissage manuel du chemin. Ce qui amène à l'hypothèse suivante :

La correction apportée par les RFID, tout en rendant le robot plus précis dans son positionnement dans l'absolu, rend sa cartographie interne plus instable. Durant son parcours manuel, il est donc amené à poser des points de passages qui sont incohérents entre eux (ie un saut de position lui fait placer un point qui dans l'absolu se

trouve hors route, dans un obstacle) Le test suivant nous a donc amené à positionner les bornes RFID puis à désactiver le module de localisation RFID via l'API dédiée (RadioNavigator), faire apprendre le parcours au robot, puis réactiver la navigation RFID, et faire le parcours en autonome au robot.

Le parcours effectué est le même que précédemment, sur 5 tours cette fois.

Tour	Rapport
1	X – P6 R – P3, P4, P5
2	R – P2, V5, H – P2, P7
3	X – P7 R – V5, P5
4	RAS
5	R – P5, V5

Conclusion : Le fait de retirer les RFID n'a pas réglé les problèmes de discontinuités durant l'apprentissage. Test ultérieure une fois que le système de correction GPS aura été monté.