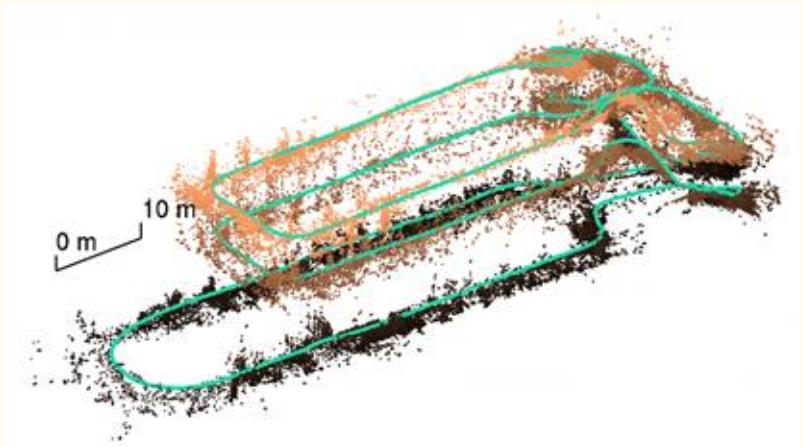


MODAL PHY473P

Projet Robot Autonome



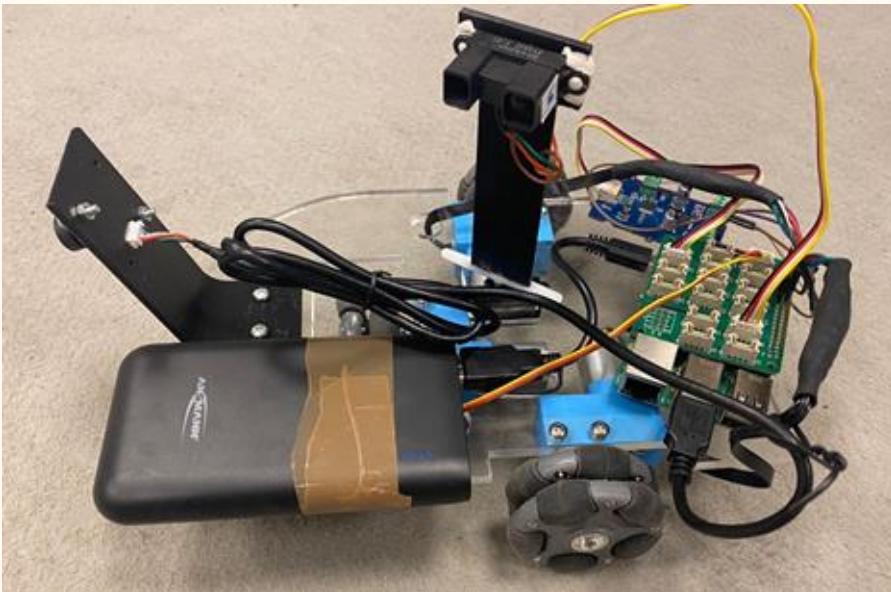
Suivie de ligne

SLAM



Aaron Metchinjin
Daniel Elmaleh

Introduction



- Constitution:
Châssis, Roues et moteurs, Batterie,
Camera, Moteur servo, Capteur IR

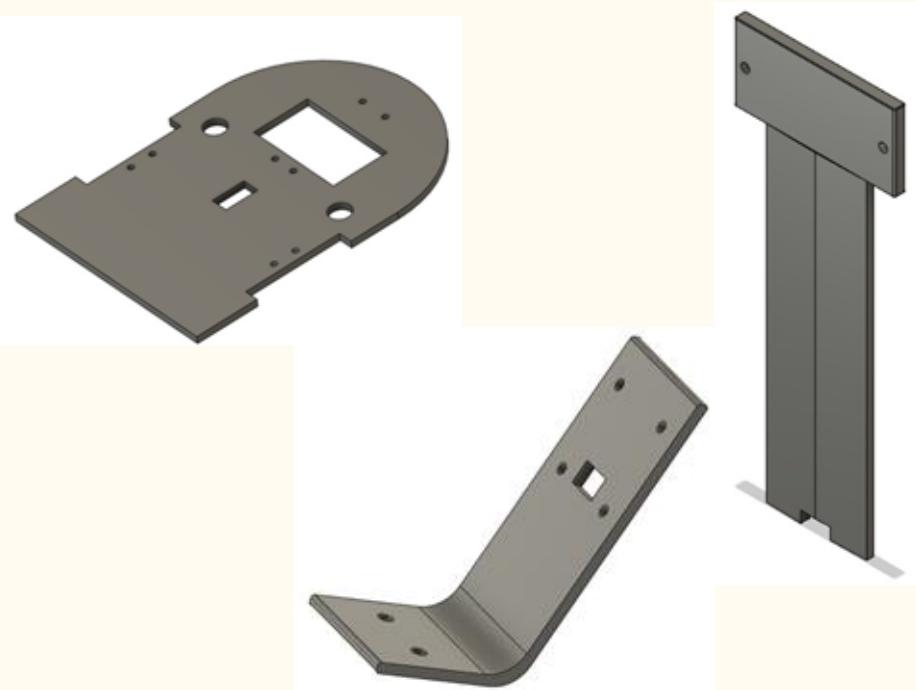
- Basé sur une raspberry pi-3
 - Suivie d'une ligne rouge au sol
 - Tour (Servo, IR)
 - Guide (Camera + Roues)
 - SLAM (Roues + Capteur IR)
-

Conduite effective du projet et répartition des tâches

- 4 phases
 - Un échéancier révisé
 - Etude et choix du matériel
 - Réalisation des pièces mécaniques et branchements
 - Algorithme de suivi ligne
 - Des pistes pour la seconde partie du projet
-

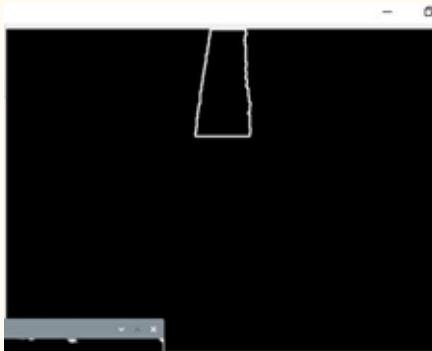
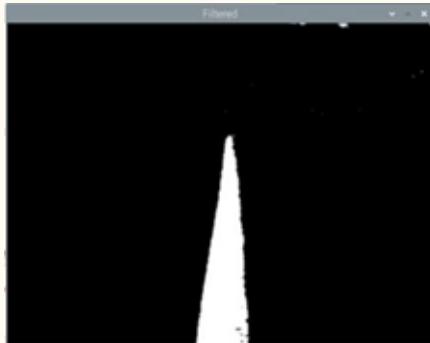
Conception & Réalisation

Les pièces mécaniques



- Modélisation CAO sur Fusion 360.
- Découpe Laser - plexiglas 6 mm.
- Impression 3D PLA.

Conception & Réalisation

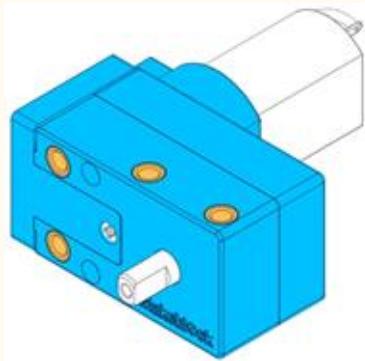


Traitement de l'image à l'aide de la caméra

- Conversion en format HSV et filtrage.
- Lissage
- Isolation des contours

Conception & Réalisation

Contrôle des roues



- Déclencher la rotation des roues (simplifier les syntaxes)
- Obtenir un feedback (une donnée de vitesse) et normaliser
- Asservir (PID)

```
float speed1 = (count1 - last_count1) * 60.0 / ENCODER_PPR / ((float)(time1 - last_time1) / 1000.0);
speed1 = speed1*100/160;
```

Conception & Réalisation

Servo-moteur



- Interfaçage entre moteur en carte.
soft PWM
- Gestion de la rotation du moteur &
algorithme. Par test.
- Ajustement des limites et des
parasites.

Conception & Réalisation

Capteur IR



```
float voltage_to_distance(int voltage) {
    // Use the inverse model of the Sharp 2Y0A02 IR sensor to convert voltage to distance.
    // Check the datasheet for more accurate values.
    float distance = 27.728 * pow((float)voltage / 1000.0, -1.2045);
    //distance = (distance/6.0) + 7.5;
    return distance;
}
```

- Interfaçage avec Carte. I2C, analogue.
- Calcul et conversion de mV en cm par tests empiriques
- Gestion de l'information, affichage et mise en forme cartésienne/angulaire.
- Threading
- SLAM - Gestion et publication sur un noeud ROS en format LaserScan

Conception & Réalisation

Prise de décision

```
double weights[] = {0.1, 0.5, 1.5};

if (currentGoal.x >= 0 && currentGoal.y >= 0) {
    double leftThreshold1 = frame.cols * 0.1;
    double leftThreshold2 = frame.cols * 0.25;
    double leftThreshold3 = frame.cols * 0.4;

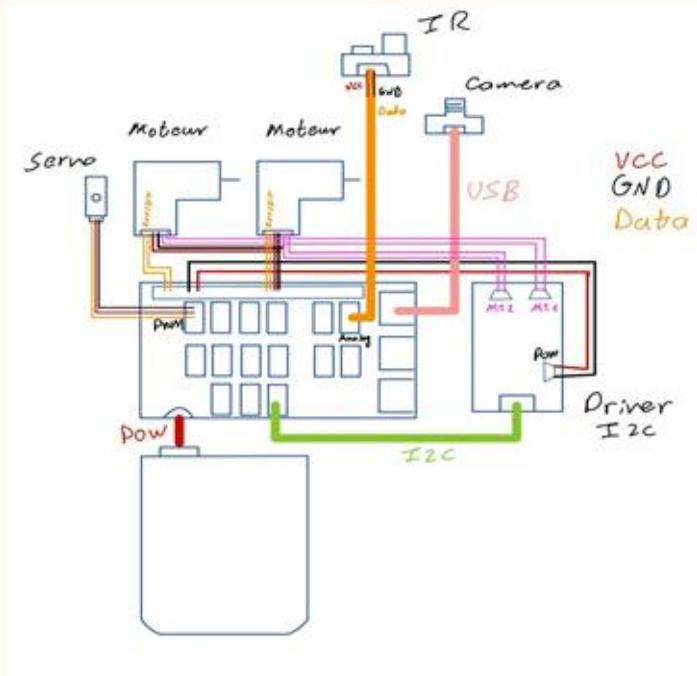
    double rightThreshold1 = frame.cols * 0.6;
    double rightThreshold2 = frame.cols * 0.75;
    double rightThreshold3 = frame.cols * 0.9;

    if (currentGoal.x < leftThreshold1) {
        printf("Turn left (proportion 1)\n");
        control_Mobilite(TURN_V, TURN_V*3, 0);
        last = 1;
    } else if (currentGoal.x < leftThreshold2) {
        printf("Turn left (proportion 2)\n");
        control_Mobilite(TURN_V, 2*TURN_V, 0);
        last = 2;
    } else if (currentGoal.x < leftThreshold3) {
        printf("Turn left (proportion 3)\n");
        control_Mobilite(TURN_V, TURN_V, 0);
        last = 3;
    }
}
```

- Division en 3 parties horizontalement en poids.
- Proportionnalité des vitesses.
- Mémorisation
- Perspectives pour éviter des obstacles (pas fini).

Conception & Réalisation

Mise en commun



- Connexions électriques et compatibilité. Communications de la carte.
- Gestion des boucles, séquentiel. Prises de décisions.
- Threading.
- Communication par noeuds avec ROS.

Bilan des tests difficultés rencontrées

Un suivi de ligne :

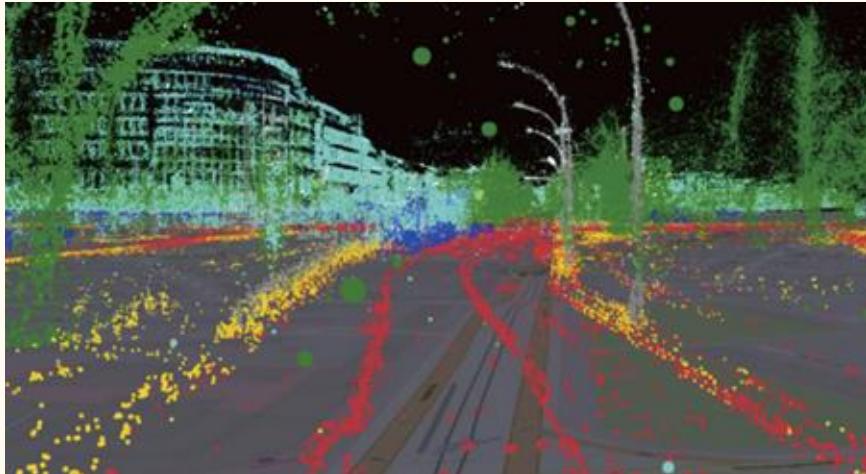
- D'une précision de +/- 2cm.
- D'une vitesse d'avancement de 20%
- Quelques instabilités

Difficultés rencontrées

- Difficultés organisationnelles.
 - Recherche de la bonne documentation.
 - Changement de carte Raspberry.
-

SLAM en extérieur

Défis à l'extérieur



- Variations de luminosité
 - Objets en mouvement
 - Environnement non structuré
 - Dimensionnalité et taille de la cartographie
 - Conditions météorologiques
-

SLAM en extérieur

Approches et techniques utilisées

- Choix des capteurs
 - Prétraitement des données
 - Fusion des capteurs
 - Techniques de gestion de mémoire
 - Méthode des points d'intérêt
-

Conclusion

- Un système qui réagit.
- Compréhension des fondamentaux en robotique.
- Manipulation des capteurs (techniques de traitement d'image)
- Travail en équipe et gestion de projet