



# Étude Bibliographique Robotique Mobile

**Fida Ben Abdallah**

24 Mai 2021

**Équipe du projet PIF "GT2", Lermab : P-J.MÉAUSOONE,  
A.BOUALI, F.BEN ABDALLAH**

# Ordre de Jour

- 1 Projet PIF-GT2
- 2 Modélisation
- 3 Navigation Autonome
  - Cartographie
  - Localisation
  - Planification
  - Commande
- 4 Détection de tronc d'arbre
  - Scan Laser
  - Vision
  - Vision et Laser combinés
- 5 Outil de Désherbage
- 6 Conclusion

## Objectif du projet PIF-GT2

Étudier la faisabilité d'un outil robotisé pour effectuer les travaux de dégageement en plantation (peupleraie)

### Planning Projet PIF-GT2 :

- (1) **Étude de l'existant dans la robotique mobile**  
(application à la forestier et l'agriculture)
  - *en cours de réalisation*
- (2) **Analyse des contraintes environnementales**  
(visite de sites forestier et peupleraies) ✓
- (3) **Identification des besoins technologiques** ✓  
(définition des spécifications techniques)
- (4) **Rédaction du cahier des charges** ✓
- (5) **Proposition de concept d'un désherbeur forestier**
  - *en cours de réalisation*



FIGURE – Plantation des peupliers (visite le 12/03)

# Fonctionnalités techniques

## Fonctions principales :

- (1) **FP1** : Naviguer et déplacer l'outil désherbeur de manière autonome dans un milieu forestier.
- (2) **FP2** : Reconnaître un plan et l'identifier.
- (3) **FP3** : Désherber au pied du plan (peuplier).

## Fonctions contraintes :

- (1) **FC1** : Contourner les obstacles ( $50\text{cm} < \text{hauteur}$ ).
- (2) **FC2** : Éviter le renversement (pente longitudinale  $45^\circ$ ).
- (3) **FC3** : Ne pas abîmer les jeunes plants (écorce).
- (4) **FC4** : Installer et assurer la stabilité de l'outil lors d'une manœuvre (fonction anti-basculement).
- (5) **FC5** : Être d'un gabarit compact (dimension mini-pelle).
- (6) **FC6** : Permettre au forestier d'effectuer les travaux de maintenance courants (changement d'outil).
- (7) **FC7** : Installer des instruments de mesure adéquates
- (8) **FC8** : Être autonome (capacité batterie : jusqu'à 8H).
- (9) **FC9** : Respecter le budget fixé au prototype.
- (10) **FC10** : Être conforme aux normes d'ergonomie.
- (11) **FC11** : Être conforme aux normes de sécurité.
- (12) **FC12** : Éviter l'utilisation des herbicides.

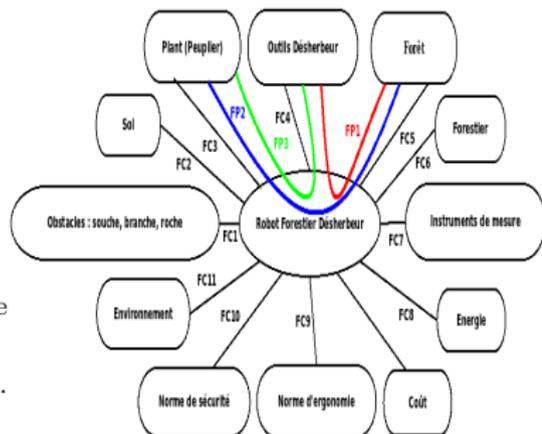


FIGURE – Diagramme des interacteurs (voir annexe A)

# Modélisation en Robotique Mobile

Selon les besoins et les capacités de calcul, il est possible d'interconnecter différents sous-systèmes représentant chacun un modèle dynamique d'un organe du véhicule.

- Modélisation de la liaison pneumatique/chaussée
- Modélisation de la dynamique des roues
- Modélisation de la suspension
- Modélisation du châssis : Dynamique longitudinale et latérale

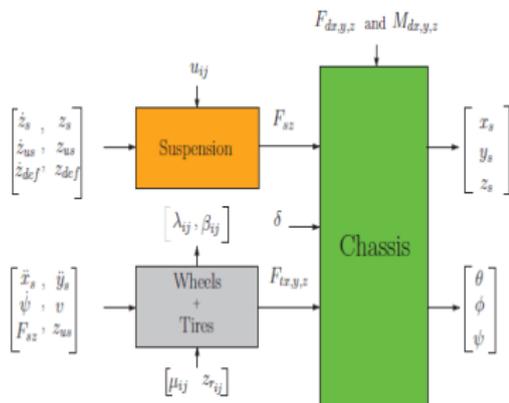


FIGURE – Modélisation en robotique mobile



# Étude Bibliographique

Principalement, il y a deux thèses :

- (1) Nagham Shalal, Orchard mapping and mobile robot localisation using on-board camera and laser scanner data fusion, University of Southern Queensland, 2015
- (2) Omer Shalev, Robot Navigation in Orchards Using Top-View Imagery, Technion Institute of Technology, 2018

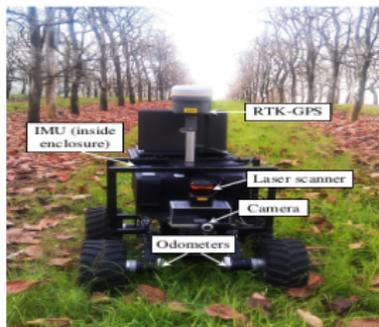


FIGURE – Robot

FIGURE – Robot et drone

Capteur	Mesures/output
Lidar 3D et 2D	distance et angle
GPS/GNSS	position $(x, y, z)$
Centrale inertielle IMU	orientation $(\alpha, \beta, \gamma)$ , vitesse angulaire $(\dot{\alpha}, \dot{\beta}, \dot{\gamma})$ , accélération $(\ddot{x}, \ddot{y}, \ddot{z})$
Encodeur roues	vitesse linéaire $(\dot{x}, \dot{y}, \dot{z})$
Caméra	image (couleur, forme ...)

# Première Approche : Construire la Cartographie

Pour construire la cartographie

- (1) Détecter le tronc d'arbre et les autres objets à partir des données capteurs.
- (2) Déterminer les coordonnées et vérifier s'ils existent ou pas sur la carte.
- (3) Mettre à jour la carte.

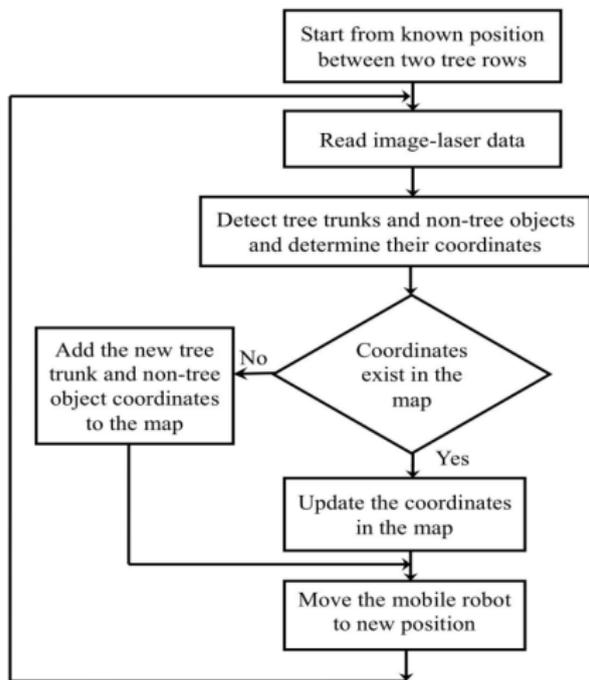


FIGURE – Algorithme de la cartographie

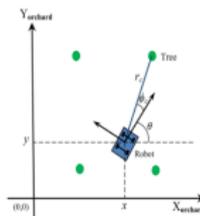


FIGURE – Troncs d'arbres simulés

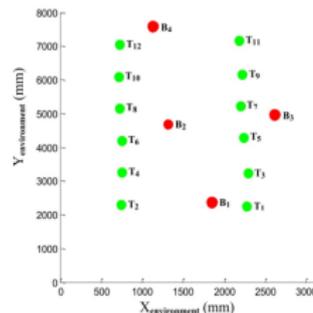


FIGURE – Construction de la Carte

# Deuxième Approche : Construire la Cartographie

Pour construire la cartographie

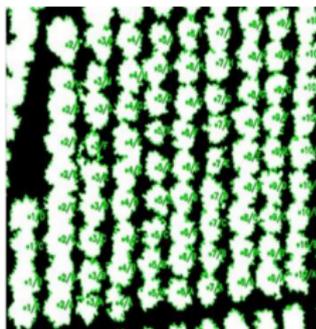
- (1) Acquisition d'une image vue de dessus par un drone à une altitude de 80 m
- (2) Traitements numériques de l'image (déterminer les positions approximatives du tronc d'arbre)

⇒ La cartographie est virtuellement une grille, représentée comme une matrice dont les valeurs décrivent :

- +1 : un arbre est présent dans la grille
- -1 : un arbre n'est pas forcément présent dans la grille
- 0 : un arbre peut être présent dans la grille, mais il n'appartient pas à cette parcelle



FIGURE – DJI Mavic Air + caméra 4k



0	+1	+1	+1	+1	+1	+1	+1	+1	+1
0	+1	+1	+1	+1	+1	+1	+1	+1	+1
-1	+1	+1	+1	+1	+1	+1	+1	+1	+1
-1	+1	+1	+1	+1	+1	+1	+1	+1	+1
-1	+1	+1	+1	+1	+1	+1	+1	+1	+1
+1	+1	+1	+1	+1	+1	+1	+1	+1	+1
+1	+1	+1	+1	+1	+1	+1	+1	+1	+1
+1	+1	+1	+1	+1	+1	+1	+1	+1	+1
+1	+1	+1	+1	+1	+1	+1	+1	+1	+1

FIGURE – Construction de la Carte

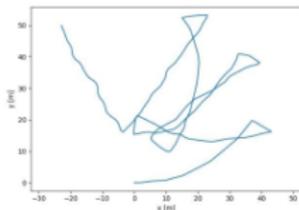
## Contraintes environnementales

- la rugosité du terrain entraîne un dérapage important du véhicule.
- les grandes dimensions du verger conduisent à une erreur accumulée.
- occlusions de la canopée des arbres dégrade le signal GPS au sol

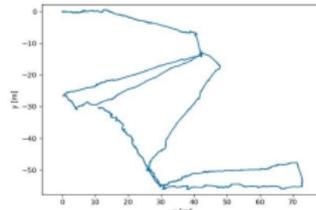
Omer a testé des algorithmes en téléopérant un UGV le long de la trajectoire :



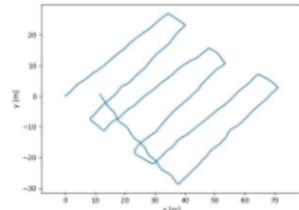
La trajectoire commence et se termine au même point



Résultat  $n^{\circ}1$  : trajectoire estimée (odométrie brute)



Résultat  $n^{\circ}2$  : trajectoire estimée (Lidar)



Résultat  $n^{\circ}3$  : trajectoire estimée (odométrie + IMU)

Résultat  $n^{\circ}1$  L'odométrie de la roue n'est pas fiable pour la localisation. Ces erreurs proviennent d'une estimation incorrecte du cap

Résultat  $n^{\circ}2$  Algorithme Iterative Closest Point qui fonctionne sur les lectures du LiDAR

Résultat  $n^{\circ}3$  Fusionner les données de l'odométrie brute avec l'IMU, en appliquant le Filtre de Kalman Étendu, apporte une amélioration significative, sauf que l'erreur accumulée au point final est toujours importante (environ 15 mètres).

# Approche Filtre de Kalman Étendu pour la Localisation

L'approche proposée par Nagham est :

- Estimer la pose du robot à partir de données fusionnées (odométrie, centrale inertielle, télémètre laser et caméra) en appliquant le filtre de Kalman étendu.
- Le GPS RTK offre une précision nominale d'environ dix centimètres
- Le robot se déplace entre les rangées et les arbres de la même rangée.

⇒ Malgré une amélioration significative en précision, cette approche nécessite la connaissance de la position initiale du robot

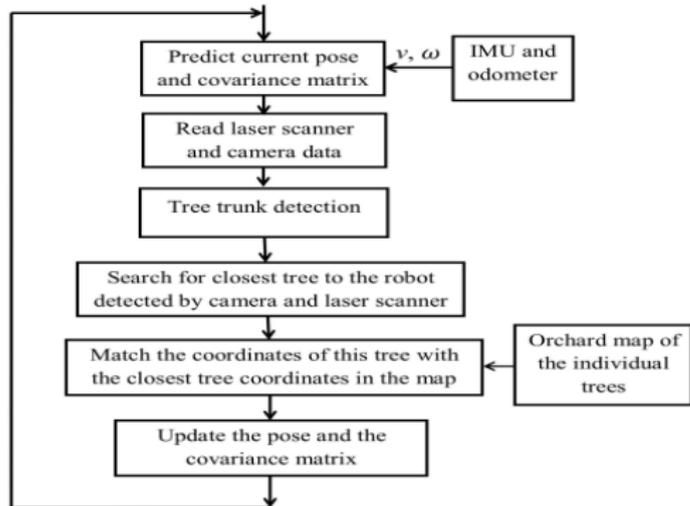


FIGURE – Approche localisation

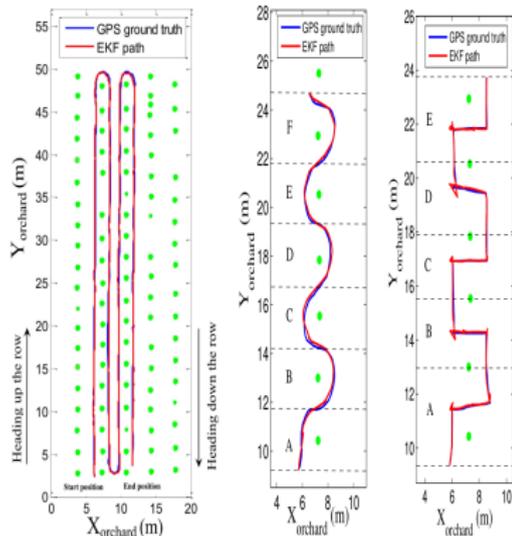


FIGURE – Localisation robot

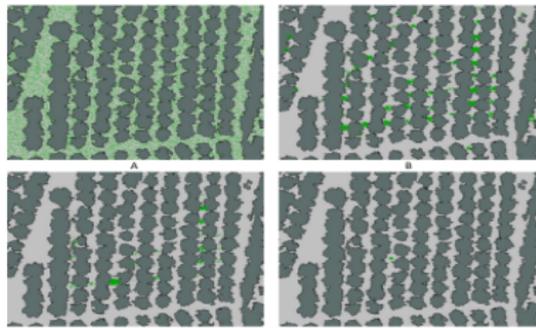
# Approche Filtre à Particules pour la Localisation

L'approche proposée par Omer est :

- approche probabiliste : Localisation Monte Carlo Adaptative (AMCL).
- utilise un filtre à particules pour suivre la pose d'un robot par rapport à une carte.
- répond au problème du robot kidnappé : capacité à fonctionner sans connaissance préalable de l'état initial du robot.



Acquisition d'une image de haute altitude (60m) et d'une vidéo à basse altitude (20m)



Le processus de convergence progressive des particules AMCL

La procédure à suivre est :

- Pour chaque trame du flux, le véhicule est segmenté à partir de l'image et son emplacement est utilisé comme point d'origine pour générer le « scan virtuel de la canopée ». Il est calculé à l'aide des techniques de traitement d'image
- les scans sont utilisés pour la localisation globale en faisant correspondre les scans à une carte précédente, et en résolvant ainsi le problème du robot kidnappé

# Planification de chemin

## Planificateur de chemin global :

- à partir d'une image vue de dessus.
- créer la carte CostMap dont le coût d'une cellule augmente au fur et à mesure qu'elle se trouve à l'intérieur des contours.
- utiliser l'algorithme A-star

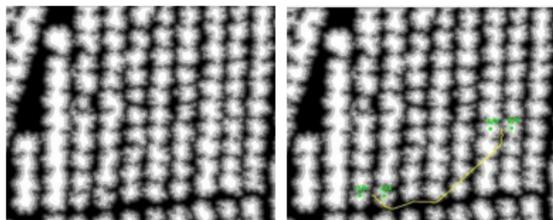


FIGURE – Cost Map



FIGURE – Planification de chemin

## Mettre à jour le chemin global :

- Des images vue de dessus périodiques permet d'identifier les obstacles
- Suite à la détection d'obstacles, de nouvelles trajectoires sont calculées

# Autre Approche

Pour déterminer le chemin souhaité entre les rangées d'arbres :

- détecter les rangées d'arbres en tant que lignes à l'aide de la transformation de Hough
- utiliser des bandes réfléchissantes pour marquer le début et la fin de chaque rangée
- les arbres individuels n'ont pas été détectés.

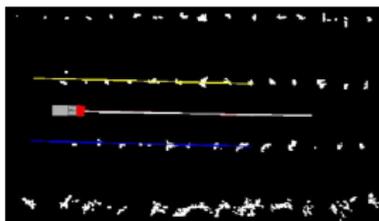


FIGURE – détection à l'aide d'un scanner laser 2D

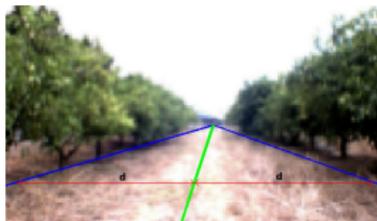


FIGURE – Détection à l'aide d'un capteur de vision



FIGURE – utilisation bande réfléchissante

Pour détecter les arbres individuellement :

- utiliser des balises, les « Tags RFID » attaché aux arbres.
- ces balises permettent aux lecteurs RFID de détecter les arbres.
- coûts d'acquisition des balises et la main-d'œuvre élevés



FIGURE – Tags RFID

# Commande : suivi chemin

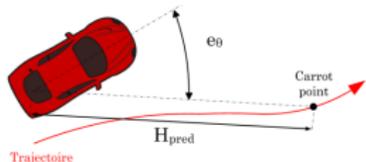
Pour assurer le suivi, les méthodes proposées déterminent les commandes à appliquer aux actionneurs.

- Angle de braquage  $\delta_f$  : pour ajuster la position du véhicule latéralement à la trajectoire
- accélération/décélération : pour ajuster les mouvements longitudinaux.

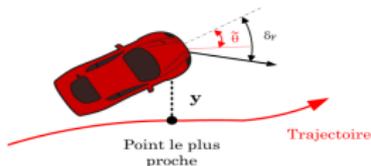
⇒ Ces méthodes sont applicables sous l'hypothèse que la position et l'orientation du robot par rapport à la trajectoire sont connues

## Contrôleur Stanley

### Contrôleur Follow the carrot

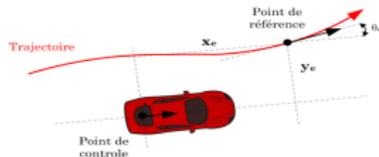


$$\delta_f = k_c e_\theta \quad (1)$$



$$\delta_f = (\theta - \theta_{courb}) + \arctan \frac{k_y}{k_v + v} \left\{ \begin{array}{l} v = v_r \cos \theta_e + k_x x_e \\ w = w_r + v_r (k_y y_e + k_\theta \sin \theta_e) \end{array} \right. \quad (2)$$
$$(3)$$

## Contrôleur de Kanayama



avec  $k_x, k_y, k_c, k_v,$  et  $k_\theta$  ... des gains positifs et constants à déterminer et  $(v_r, w_r)$  les vitesses désirées au point de référence

# Commande : suivi chemin

Le contrôleur de Kanayama à la fois géométrique et temporelle [vidéo] :

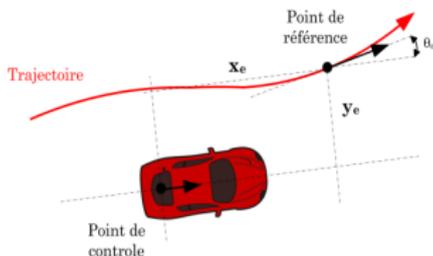
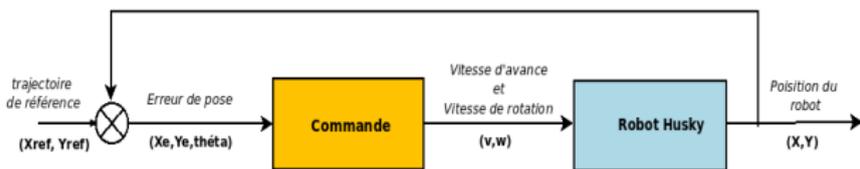


FIGURE – Représentation schématique du contrôleur de Kanayama.

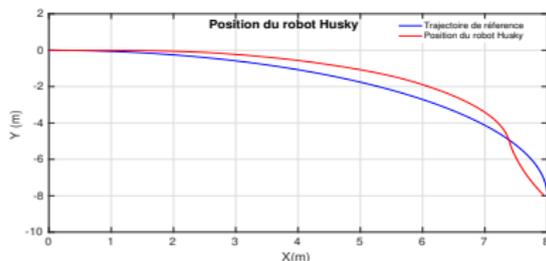


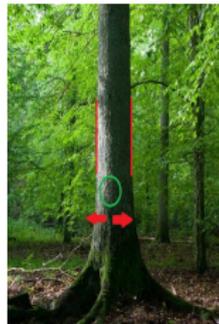
FIGURE – Trajectoire de référence (bleu) et suivi de chemin (rouge) à  $1\text{m}\cdot\text{s}^{-1}$

# Détection de tronc d'arbre

## 2<sup>e</sup> Fonction principale :

- **Reconnaître un plan et l'identifier.**

Le processus de détection révèle les questions suivantes :



(1) Quelle sont les caractéristiques d'un tronc d'arbre?

- Largeur du tronc
- Couleur du tronc
- Bord parallèle d'un tronc

(2) Comment une telle caractéristique est-elle extraite/détectée?

- scan laser 2D : porté ( $r_1, r_2$ ) et angle ( $\phi_1, \phi_2$ )
- caméra : nuance  $H_d$  et bord du tronc

(3) Comment une telle caractéristique est-elle quantifiée?

- $d$  : largeur du tronc
- $ROC_L$  : taux de confiance de l'objet à partir du laser
- $ROC_C$  : taux de confiance de la couleur du tronc de l'arbre
- $ROC_E$  : taux de confiance de la détection des bords pour chaque tronc d'arbre

(4) Comment une telle caractéristique est-elle évaluée?

- Déterminer les valeurs seuils prédéfinies  
( $TH_{Tree}, TH_L, TH_C, TH_E$ )

# Détection de tronc d'arbre : Algorithme A

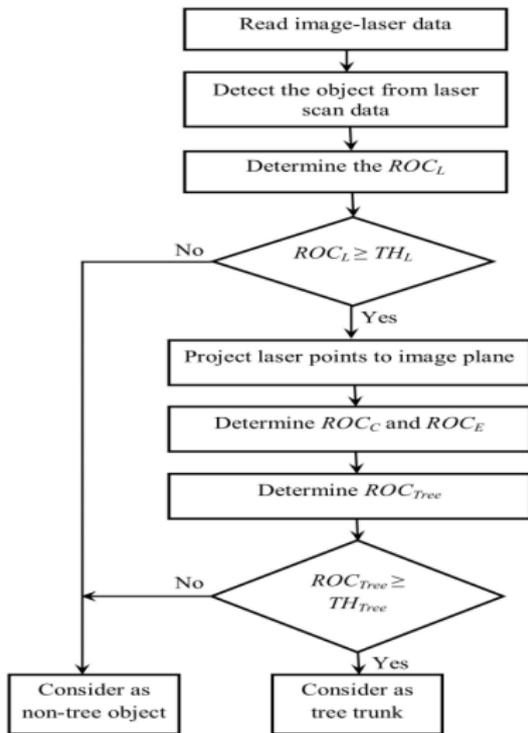


FIGURE – Algorithme A

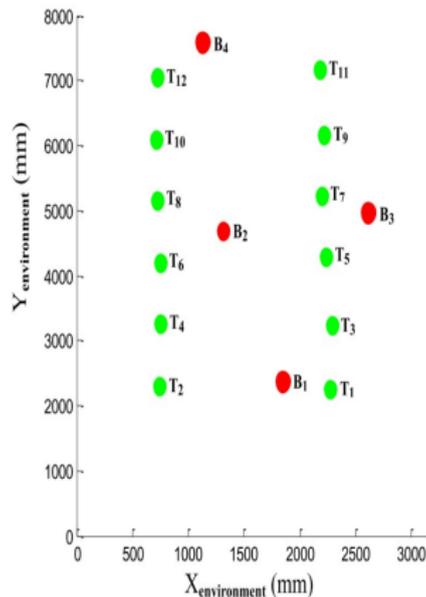


Figure 5.4: The map of the simulated environment.

FIGURE – Résultat

# Détection de tronc d'arbre : Algorithme B

- (1) Détecter l'objet et déterminer ses caractéristiques (largeur, couleur, forme, ...).
- (2) Classifier les objets détectés selon les critères de sélection en tronc (normal et petit), support (grand et petit) et autres objets.

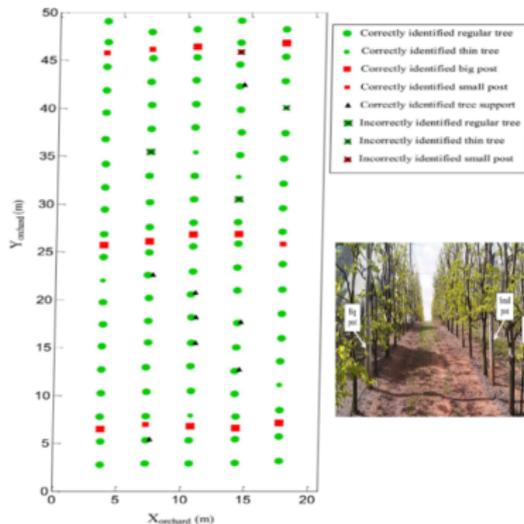
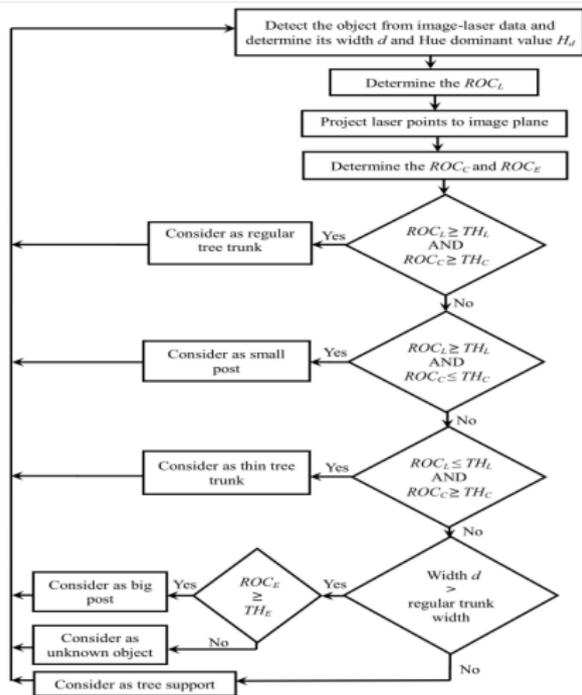


Fig. 8. The constructed map of the selected area of the orchard.

FIGURE – Algorithme détection de tronc d'arbre

Le taux de confiance pour le tronc d'arbre  $ROC_{tree}$  :

$$ROC_{Tree} = ROC_L W_L + ROC_C W_C + ROC_E W_E \quad (4)$$

Les taux de confiance sont calculé à partir de la densité de probabilité (*pdf*) de chaque mesure :

$$ROC_L = \frac{pdf(d)}{pdf(\mu_d)} \quad (5)$$

$$ROC_C = \frac{pdf(H_d)}{pdf(\mu_{H_d})} \quad (6)$$

$$ROC_E = w_1 R_1^2 + w_2 R_2^2 + w_3 E_p \quad (7)$$

avec

- $W_L$ ,  $W_C$  et  $W_E$  : pondérations liées respectivement à  $ROC_L$ ,  $ROC_C$  et  $ROC_E$ .

Largeur du tronc d'arbre

$$d = \sqrt{r_1^2 + r_2^2 - 2r_1r_2 \cos(\Delta\phi)} \quad (8)$$

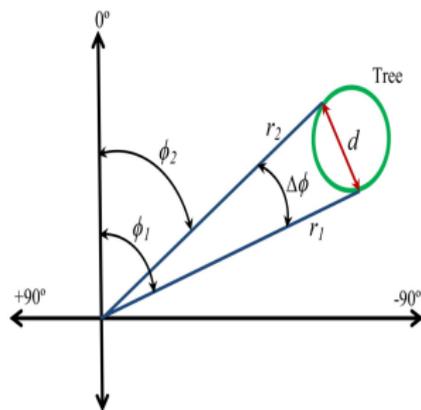
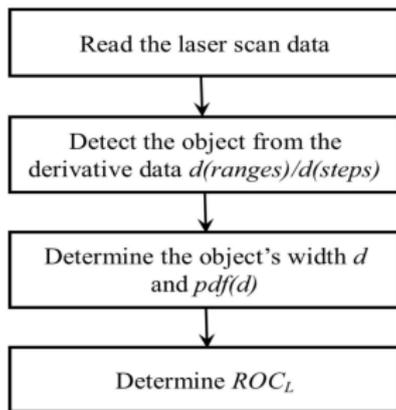


FIGURE – Détection de tronc d'arbre par laser

FIGURE – Détermination de la largeur de l'objet à partir du scanner laser Hokuyo UTM-30LX/LN

# Vision : Nuance par zone d'intérêt

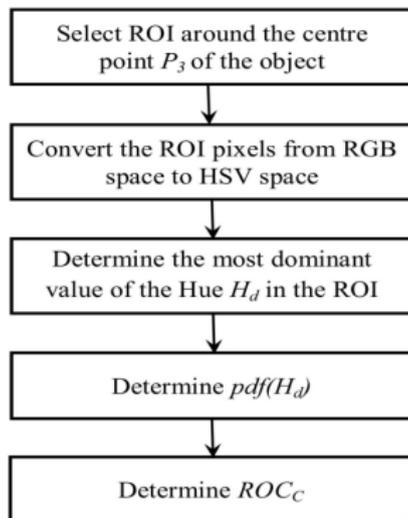


FIGURE – Détection de la couleur du tronc d'arbre

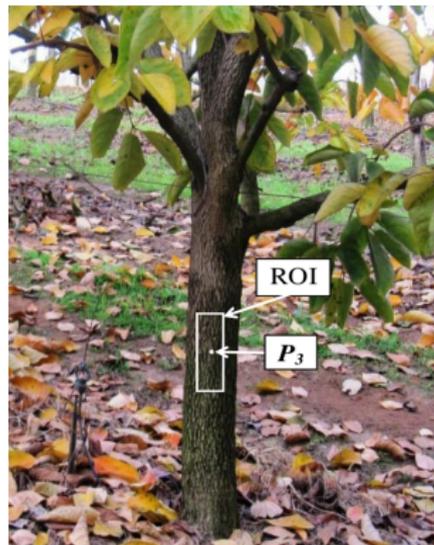


FIGURE – ROI : zone d'intérêt pour la détection des couleurs

# Vision et laser : bord vertical d'un tronc d'arbre

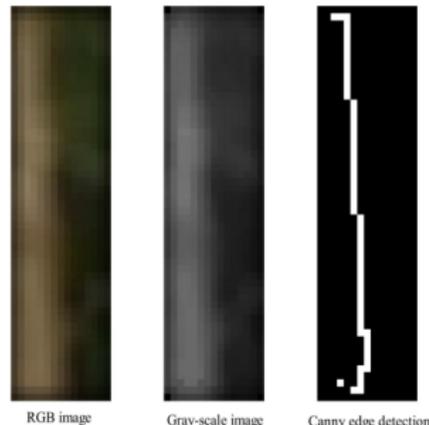
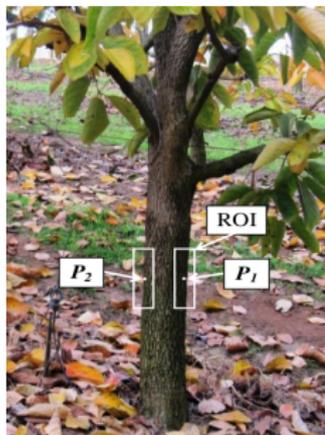
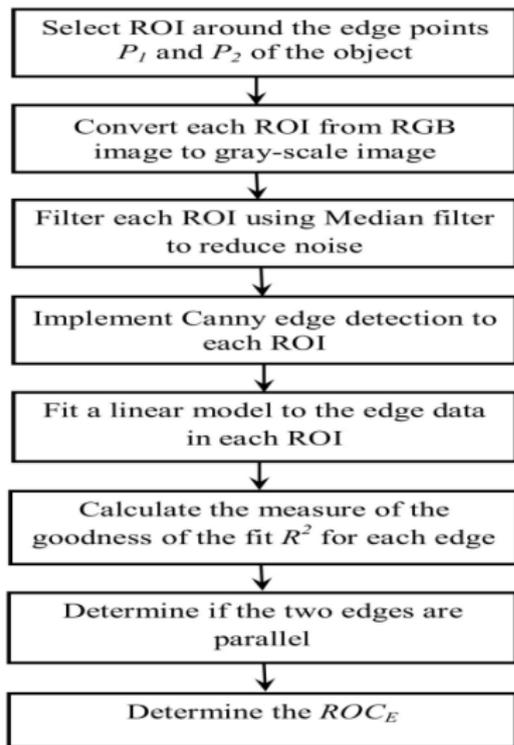


FIGURE – ROI : zone d'intérêt pour la détection des bords

FIGURE – ROI images in RGB, gray-scale and Canny edge detection.

FIGURE – Détection des bords de tronc d'arbre

# Fusionner les données laser-caméra

Pour fusionner les données laser-caméra :

(1) **calibration d'une caméra** : déterminer la transformation ponctuelle faisant passer du point 3D exprimé dans un repère absolu à son image.

- modéliser l'optique de la caméra (paramètres intrinsèques) en utilisant un damier dessiné sur un plan.
- déterminer la transformation repère absolu/ repère caméra

(2) **calibration laser-caméra** :

- déterminer la transformation entre les repères des deux capteurs

⇒ Concevoir un système de navigation permettant au robot mobile d'éviter les obstacles et reconnaître le plant en utilisant SLAM



FIGURE – calibration d'une caméra

## 3<sup>e</sup> Fonction principale :

- **Désherber au pied du plan (peuplier).**

L'outil de désherbage est composé :

- (1) **Tête de désherbage** équipée par des griffes ou des épines
- (2) **Moteur** pour faire tourner la tête de désherbage
- (3) **Support** spécial placé à l'avant du robot ou sur l'attelage



FIGURE – outil de désherbage 1 [video]



FIGURE – outil de désherbage 2 [video]



FIGURE – outil de désherbage 3 [video]

# Conclusion : Récapitulatif de l'étude bibliographique

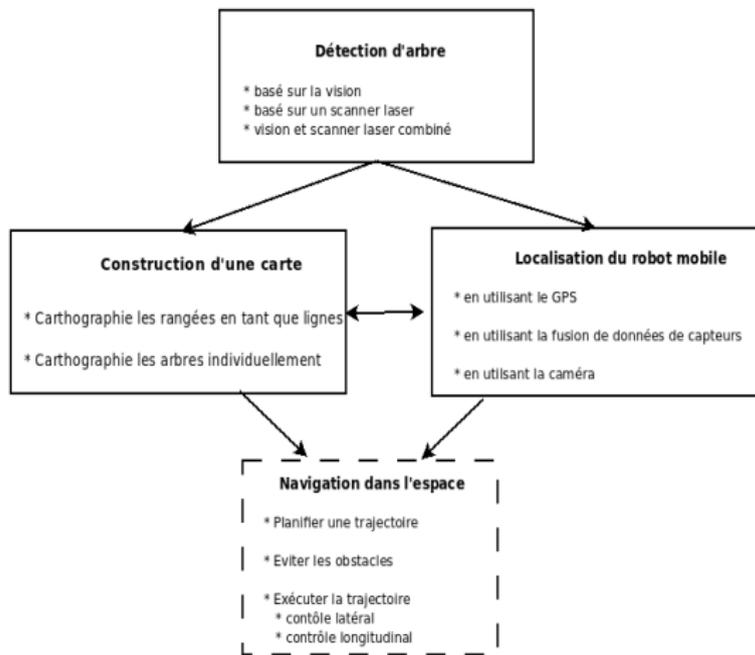


FIGURE – Les principaux thèmes de la revue de littérature

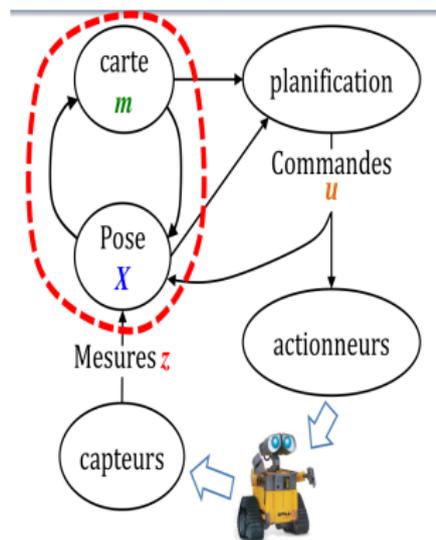


FIGURE – Navigation autonome



# Étude Bibliographique Robotique Mobile

**Fida Ben Abdallah**

24 Mai 2021

**Équipe du projet PIF "GT2", Lermab : P-J.MÉAUSOONE,  
A.BOUALI, F.BEN ABDALLAH**