

Projet de Robotique : Comportement du Robot

Groupe 7

April 26, 2018

1 Comportement du Robot

```
while True do  
    Détection des personnes()  
    Détection des groupes()  
    Rejoindre un groupe()  
end
```

Algorithm 1: Comportement général

1.1 Détection des personnes

C'est le noeud : `moving_person_detector` qui détecte les personnes.

```

nb_active_leg ← 0
nb_person ← 0
active_leg : les différentes jambes
score : les différents scores pour les jambes
persons : les différentes personnes
for  $i \leftarrow 0$  to 10 do
     $D \leftarrow laser\_data$ 
     $i \leftarrow 0$ 
    C : les différents clusters
     $c_i.debut \leftarrow d_0$ 
     $c_i.fin \leftarrow d_0$ 
    foreach  $d_j \in D$  do
        if  $distance(d_{j-1}, d_j) < 0.01$  then
             $c_i.fin \leftarrow d_j$ 
        end
        else
             $i \leftarrow i + 1$ 
             $c_i.debut \leftarrow d_j$ 
             $c_i.fin \leftarrow d_j$ 
        end
    end
     $i \leftarrow 0$ 
    Legs : Les jambes repérées
    foreach  $c_j \in C$  do
        if  $c_j \geq 0.15 \wedge c_j \leq 0.3$  then
             $Legs_i \leftarrow c_j$ 
             $i \leftarrow i + 1$ 
        end
    end
    foreach  $l_i \in Legs$  do
        if  $\exists l'_j | distance(l_i, l'_j) < 0.05$  then
             $score_j \leftarrow score_j + 1$ 
        end
        else
             $active\_leg_{nb\_active\_leg} \leftarrow l_i$   $score_{nb\_active\_leg} \leftarrow 1$   $nb\_active\_leg \leftarrow nb\_active\_leg + 1$ 
        end
    end
     $i \leftarrow 0$ 
    foreach  $l_j \in active\_leg$  do
        foreach  $l_k \in active\_leg$  do
            if  $l_k \neq l_j \wedge distance(l_k, l_j) \leq 0.75 \wedge score_j > 5 \wedge score_k > 5$  then
                 $person_i \leftarrow \frac{d_j + d_k}{2}$ 
                 $i \leftarrow i + 1$ 
            end
        end
    end
end

```

Algorithm 2: Détection des Personnes

1.2 Les groupes

On considère comme groupe, un ensemble d'au moins deux personnes, où chaque personne à un voisin à moins d'un metre. C'est le noeud group_detector qui détecte les groupes.

Si il ' y a au moins un groupe, le groupe vers lequel se déplace le robot est le groupe le plu proche du robot. Pour éviter que le robot reste devant le même groupe, le robot doit se déplacer vers un groupe qui se trouve à une distance minimale de e. C'est le noeud group_detector qui choisi le groupe à rejoindre.

```

clust_person : les différents clusters de personnes
i ← 0
clust_personi.debut ← p0 clust_personi.fin ← p0 foreach pj ∈ persons do
    if distance(pj-1, pj) < 1 then
        | clust_personi.fin ← pj
    end
    else
        | i ← i + 1
        | clust_personi.debut ← pj
        | clust_person1.fin ← pj
    end
end
groups : les différents groupes
nb_group ← 0
foreach ci ∈ clust_persons do
    if ci ≥ 2 then
        | groupsnb_group ← ci
        | nb_group ← nb_group + 1
    end
end

```

Algorithm 3: Détection des Groupes

1.3 Déplacement

```

goal_to_reach : le but à atteindre
token : le jeton
if nb_group ≠ 0 then
    | goal_to_reach ← closest_group()
    | token ← true
end
else
    | goal_to_reach ← random_point()
    | token ← false
end
publish(token)
publish(goal_to_reach)

```

Algorithm 4: Envoie du but

```

translation_to_do ← translation(goal_to_reach)
rotation_to_do ← rotation(goal_to_reach)
publish(rotation_to_do)
publish(translation_to_do)
if token then
    | rotation_to_do ←  $\pi$ 
    | publish(rotation_to_do)
end

```

Algorithm 5: Réception du but

2 architecture

