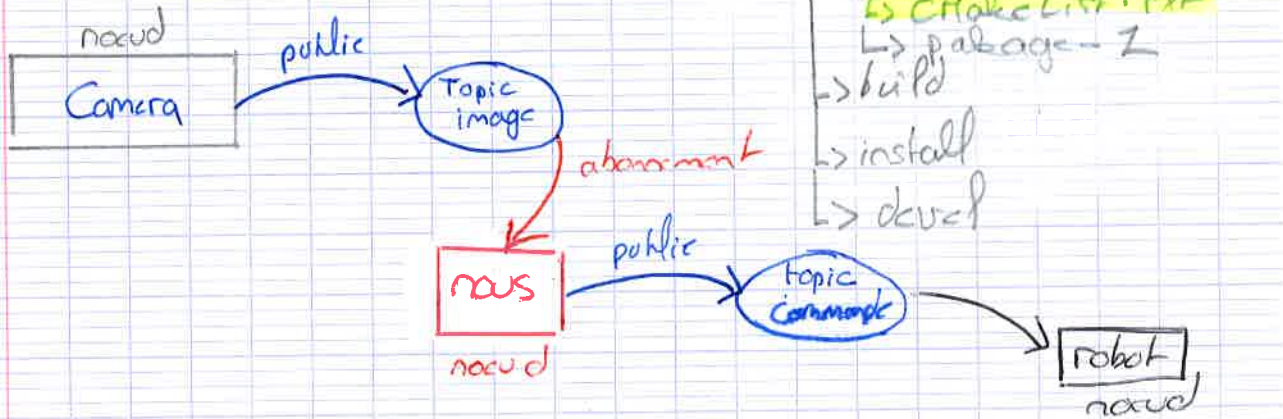


24/11/2022

Visual Servoing



Récupérer le sujet de TP

Depuis le workspace cathin/src taper
git clone <https://github.com/divier-stasse/tp-ros-visual-servo.git>

Se connecter à la branche git checkout noetic

Modifier ligne 20 du **createLists.txt** par **find-package (opencv 3 REQUIRED)**

Partie 6 du polycopié

Etape ① Compiler le paquet récupéré
cathin-make fonctionne

Autre méthode : pour partir d'un ws propre supprimer
build, dev, install
utiliser **cathin-build**

Etape ② : **roslaunch tiago-gazebo tiago-gazebo.launch**
public_sim := true robot := titanium world := objects-on-table
pour lancer une simulation du robot avec objet sur table
L'objectif est de tracker la boîte fluo

Etape ③ Donner à `rviz` l'accès au message publié
`roslaunch rviz rviz` pour traiter les topics avec un environnement visuel

ADD > Image puis sélectionner Image topic dans la fenêtre de gauche / `rgb/image-raw`

Astuce : choisir une image Compressed si échange sur un réseau Wi-Fi.

Etape ④ : Récupérer un Image transport

↳ cf `wiki.ros.org` sur internet
pour créer une image subscriber ou publisher

L'image subscriber gère une exception dans le image callback
doit mettre en place en `ros::NodeHandle nh`
`image_transport::ImageTransport it(nh)`

API: interface
d'une fonction

Modifier le code C++ du TP

`nh.subscribe("nom-topic", taille_buf, fonction_callback)`

Etape ⑤ compiler `catkin build`

Etape ⑥ contrôler manuellement la tête du robot

`roslaunch rgb-joint-trajectory-controller rgb-joint-trajectory-controller`

Etape ⑦ Lancer l'application `roslaunch tp-ros-visual-servo`

Régler les valeurs de seuil : `tp-ros-visual-servo-node`

Low H = 33

Low V = 32

High H = 166

High V = 255

Low S = 192

High S = 255

Rappel
code,
ouvre VS
code avec le
folder courant

Etape (8): Créer le publisher dans le code
`cmd_vel_pub = nl_.advertize<geometry_msgs::Twist>("Servoing-um-vel")`
 Calculer le centre de gravité

Implémenter la commande en vitesse angulaire autour de $\vec{J}^>$

↳ commande proportionnelle de gain k_p

↳ dx-clamp pour éviter au bruit

↳ min et max par la saturation de commande

Autre écriture
 var = cond ?
 val true :
 val false ;

`compute_vel_msg.angular.z (cx = lngThresholded.cdx/2.)`
 ↑
 coordonnées
 du centre

`if (nb_pts == 0)`

`command.angular.z = 0.2; // rotation à 0,2 rad/s`

si pas de points trouvés
 pour l'objet