

ASSERVISSEMENT VISUEL D'UN DRONE ET
ESTIMATION DE L'ORIENTATION D'UNE SURFACE
PLANE À L'AIDE DE DONNÉES VISUELLES

4 septembre 2019

Maxime DO ROSARIO

CI 2019 - SPID/Robotique



Résumé

Avertissement

Ceci est le résumé de mon rapport demandé par M.Jaulin . Le rapport complet est confidentiel et ne peut être consulté librement.

Merci de votre compréhension.

Remerciement

Je tiens à remercier toutes les personnes qui ont contribué au succès de mon stage et qui m'ont aidé lors de la rédaction de ce rapport.

Tout d'abord j'aimerais remercier Monsieur Benoit ZERR de l'ENSTA Bretagne qui m'a permis de trouver ce contrat de professionnalisation chez Notilo Plus.

Je tiens également mon tuteur auprès de l'entreprise, Loïc Boisson qui m'a accueilli au sein de l'entreprise et a pris le temps pour répondre à mes questions.

Je remercie également toute l'équipe de Notilo Plus pour le sérieux, l'accueil et la bonne humeur durant toute la durée du contrat de professionnalisation.

Enfin, je remercie Marc, gérant du club de plongée de Chamagieu pour nous avoir permis d'effectuer nos tests dans le lac de Chamagnieu.

Résumé

Dans le cadre de notre cursus ENSTA Bretagne, la dernière année doit comporter 6 mois de stage afin de réaliser un travail d'ingénieur. Cette période a été effectuée dans l'entreprise Notilo Plus. Ce stage étudie la possibilité d'utiliser la caméra monoculaire des drones de Notilo Plus pour l'asservir visuellement dans des eaux à faible visibilité. L'algorithme calcule la matrice d'homographie entre l'image originale et l'image actuelle afin de déduire le mouvement du drone dans l'espace. L'algorithme est rendu robuste notamment grâce à l'utilisation d'un RANSAC. La décomposition de la matrice d'homographie renvoie plusieurs solutions potentielles, l'algorithme utilise également les données de l'IMU du drone ainsi que du capteur de pression afin de déterminer la solution correcte et ainsi d'obtenir un meilleur calcul de l'erreur. Le drone n'est jamais immobile dans l'eau, ses moteurs provoquent un mouvement parasite, l'algorithme permet de compenser ce mouvement. L'algorithme a besoin d'une cible pouvant être assimilée à un plan pour fonctionner. L'algorithme a été testé sur le simulateur open-source utilisant ros et gazebo UUV simulator, et également dans le lac de Chamagnieu dans un cas réel.

Liste d'abréviations

AUV	A utomatic U nderwater V ehicle, véhicule sous-marin autonome
ROV	R emotely O perated underwater V ehicle, véhicule sous-marin téléguédé
FAST	F eatures from A ccelerated S egment T est
BRIEF	B inary R obust I ndependent E lementary F eatures
ORB	O riented F AST and R otation-aware B RIEF
RANSAC	R ANdom S Amples C onsensus
FPS	F rame P er S econds, Image par second
IMU	I nertial and M easurement U nit, Centrale inertielle
RMS	R oot M ean S quare, valeur moyenne quadratique

Table des figures

1	IBUBBLE	6
2	SEASAM	6
3	Télécommande	6
4	Logo CMA-CGM	7
5	UUV simulator	8
6	Courbe d'erreur de l'asservissement visuel sur le simulateur	9

1 Introduction

Actuellement, les tâches d'inspection en milieu sous-marin restent effectuées par des humains. Cependant, l'utilisation de plongeur peut être contraignante que ce soit pour des raisons de sécurités, administratives ou de disponibilités des-dits plongeurs. Certaines entreprises commencent à commercialisés des drones permettant de remplacer les plongeurs pour certaines tâches. Pour le moment, les drones commercialiser restent des **R**emotely **O**perated underwater **V**ehicle (ROV), nécessitant donc des opérateurs pour les contrôler. Bien que leur taille tend à réduire, ces ROV restent encombrants.

Notilo plus crée des drones de petites tailles, permettant d'effectuer des missions jusque-là faites par des humains. Les missions peuvent être du suivi supervisé pour permettre de suivre des plongeurs et s'assurer qu'aucun problème ne leur arrivent, ou des missions d'inspection de coque ou de pipeline.

Notilo plus en partenariat avec CMA-CGM doit à terme réussir l'inspection d'une coque de porte-conteneurs. Ce stage explore la possibilité d'utiliser la caméra monoculaire des drones de Notilo Plus pour aider à l'inspection de la coque. Les inspections de porte-conteneurs se passent le plus souvent dans des port industriels. L'eau étant assez polluée dans ces zones, la visibilité est mauvaise. L'objectif du stage est d'utiliser ce drone afin, dans un premier temps, d'aider à l'inspection supervisée de la coque. L'algorithme utilisera la caméra monoculaire du drone afin d'estimer les mouvements du drone puis de les compenser.

2 Présentation générale

Notilo Plus est une startup d'une quinzaine d'employés possédant des bureaux sur Lyon et Marseille. Notilo Plus a été créée en 2016 par Nicolas Gambini et Benjamin Valtin.

Notilo Plus commercialise deux gammes de drones. Une gamme plutôt orientée loisir IBUBBLE et une gamme professionnelle SEASAM.



Figure 1 – IBUBBLE



Figure 2 – SEASAM

Grâce aux 7 moteurs, ces drones peuvent utiliser 6 degrés de libertés.

Les deux gammes embarquent des capteurs similaires :

- capteur de pression
- **Inertial and Measurement Unit (IMU)**
- sonar
- capteur de fuite dans le compartiment et la batterie
- capteur de charge de la batterie
- caméra
- hydrophones (4 ou 5 selon la gamme)



Figure 3 – Télécommande

Le drone possède également une télécommande. Cette télécommande émet des ondes acoustique qui sont captées par le drone et interprétées. Ces ondes acoustiques permettent d'envoyer des ordres au drone ainsi que la distance entre la télécommande et le drone. La télécommande possède également un capteur de pression. Grâce aux 4 (ou 5) hydrophones et au capteur de pression du drone et de la télécommande, le drone arrive à déduire la position de la télécommande dans son référentiel.

La gamme SEASAM peut embarquer d'autres capteurs si des clients en font la demande.

Les drones sont contrôlables de plusieurs façon.

La gamme IBUBBLE a pour objectif initial de suivre un plongeur sans fil afin de faire des vidéos. Les ordres sont envoyés du plongeur au drone grâce à la télécommande. Le contrôle est alors sans fil. Un mode ROV est aussi possible grâce à un fil.

La gamme SEASAM possède les mêmes modes de fonctionnement que la gamme IBUBBLE. Cette gamme peut de plus faire du suivi supervisé, le drone est connecté à un ordinateur à la surface, la vidéo est donc disponible mais le plongeur garde le contrôle du drone.

Le stage s'intègre dans un partenariat entre Notilo Plus et CMA-CGM. CMA-CGM est une entreprise de transport maritime possédant des porte-conteneurs. Le partenariat a pour objectif de leur fournir une solution d'inspection autonome de la coque afin de détecter d'éventuels défaut dans la coque ou la peinture de la coque. La peinture des porte conteneurs est une peinture spéciale améliorant l'hydrodynamisme des porte-conteneurs et donc de diminuer leur consommation.



Figure 4 – Logo CMA-CGM

Le sujet du stage permet d'explorer une possibilité pour stabiliser le drone pour dans un premier temps, permettre le contrôle de la coque en supervisé. Si l'opérateur détecte une zone qui nécessite d'être vérifiée, il stabilise le drone pour vérifier ladite zone. L'algorithme doit satisfaire plusieurs besoins :

- l'algorithme doit permettre au drone de compenser l'impact de ses moteurs ainsi que de faibles courant.
- L'algorithme doit pouvoir être lancé et arrêté plusieurs fois dans une mission
- Lorsque le programme est désactivé, le fonctionnement du drone doit être normal.

De plus, l'algorithme ne doit pas être lancé au démarrage

3 Techniques utilisées

L’algorithme détecte des points et les caractérise grâce à **O**riented **F**AST and **R**otation-aware **B**RIEF (ORB)[1]. ORB utilise **F**eatures from **A**ccelerated **S**egment **T**est (FAST)[2] comme détecteur de points d’intérêt auquel il ajoute une composante pour l’orientation. Le descripteur de ORB est **B**inary **R**obust **I**ndependent **E**lementary **F**eatures (BRIEF)[3] avec ajout d’une composante pour la gestion de la rotation.

L’algorithme utilise le flux optique de Lukas-Kanade[4] afin de suivre les points d’intérêt d’une image initiale dans le temps.

L’algorithme calcule la matrice d’homographie entre la position des points d’intérêt de l’image actuelle et ceux de l’image initiale[5]. En décomposant la matrice d’homographie [6], il est possible de calculer un déplacement du drone et donc une erreur à compenser dans notre cas. Le drone doit rester immobile.

Le calcul de la matrice d’homographie est rendu plus robuste grâce à l’utilisation d’un **R**ANdom **S**AMple **C**onsensus (RANSAC)[7].

4 Résultats

4.1 Simulateur

Dans un premier temps, l’algorithme a été implémenté sur un simulateur. Le simulateur utilisé est le simulateur UUV [8]. Ce simulateur simule un robot sous-marin avec un modèle physique. Ce robot est par défaut le rexROV, il est possible de refaire un modèle physique pour un autre robot. Certains capteurs sont également simulés tel qu’un capteur de pression, un IMU, et plusieurs caméras. Le robot simulé par défaut est composé de 8 moteurs, 4 verticaux et 4 horizontaux. Ce simulateur utilise ROS et Gazebo.

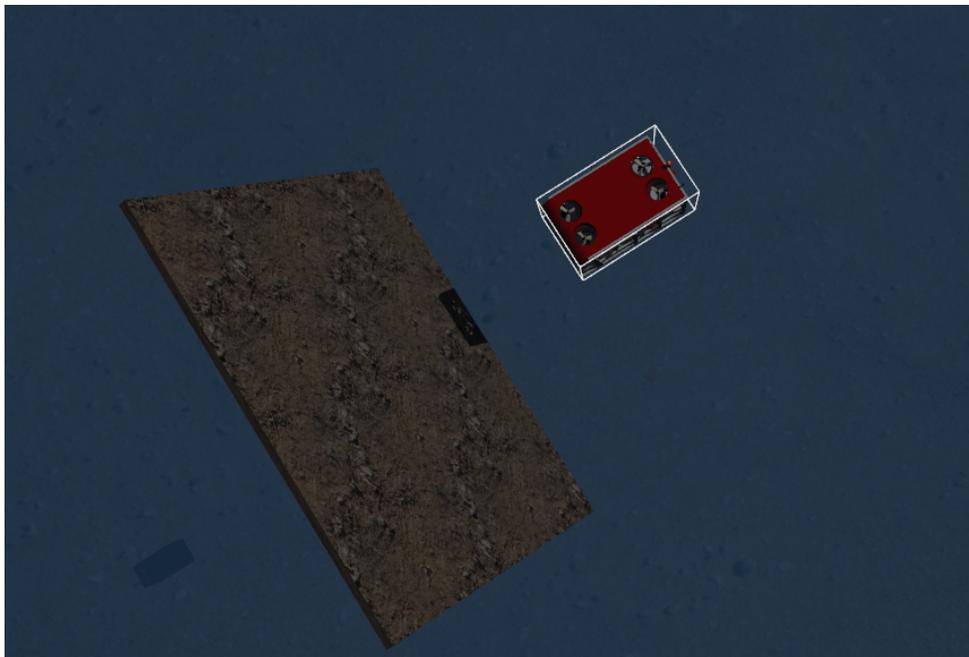


Figure 5 – UUV simulator

Le drone est situé devant un mur incliné, le mur permet de calculer la matrice d’homographie. L’eau simulée est claire et la surface du mur est un vrai plan. La texture du mur est une des textures fournies par Gazebo. Le simulateur représente ainsi le cas optimal pour la vision que l’on peut avoir.

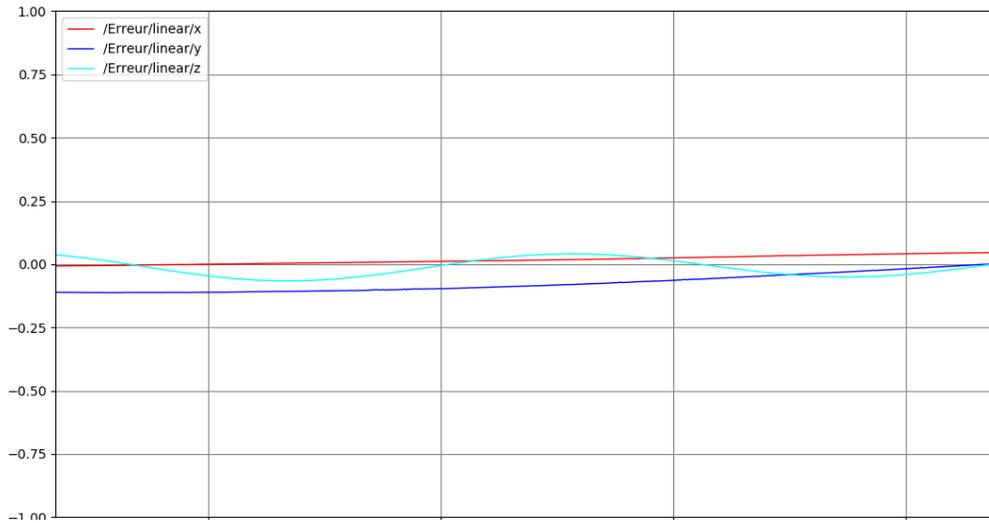


Figure 6 – Courbe d'erreur de l'asservissement visuel sur le simulateur

L'erreur est définie à une échelle près. A 1m de la surface, un décalage de 1m vaut 1m. La distance entre le drone et la surface n'étant pas connue, l'erreur est en unités arbitraires.

La matrice moteur du drone simulé n'est pas parfaite, les moteurs créent un mouvement parasite provoquant un déplacement du drone.

L'algorithme parvient à compenser les mouvements du moteurs. Cependant, vu que le PID est mal réglé, et que le poids du drone n'est pas cohérent (plusieurs tonnes) et à cause de l'inertie du drone, entre autres, le drone mets plusieurs minutes avant de se stabiliser.

L'objectif du simulateur était de vérifier que toutes les étapes de l'algorithme fonctionnaient avant de tester sur le vrai drone.

L'image de gauche est l'image originale/cible, celle de droite l'image actuelle (voir figure ??). Les lignes correspondent aux points appairés entre l'image originale/cible et l'image actuelle. La matrice d'homographie est calculable s'il y a plus de 4 lignes (ce qui est le cas actuellement). Il faut cependant avoir plus de 4 lignes pour gagner en robustesse via l'utilisation d'un RANSAC.

4.2 Robot réel

Dans le cas des tests réels, il a fallu enlever la distorsion de la caméra. Pour ce faire, la matrice intrinsèque a été calculée grâce à OpenCV, les images nécessaires à l'algorithme d'OpenCV ont été prises dans un bac d'eau avec le damier classique d'OpenCV visible. L'algorithme a été testé sur le robot réel dans le lac de Chamagnieu.

Le lac de Chamagnieu est une ancienne carrière située à 30km de Lyon. L'accès au lac est réglementé, le lac étant un lac privé lié au club de plongée de Chamagnieu.

L'eau est moins nette et l'image de moins bonne qualité que sur le simulateur. L'asservissement visuel a obtenu de moins bons résultats. L'asservissement fonctionne à condition que le drone ne soit pas trop loin (le mur n'est alors plus visible) ou trop près (le drone se retrouve trop réactif pour permettre un asservissement). L'erreur est établie à une échelle près, plus on est près du mur plus l'erreur se retrouve importante, le PID se retrouve réglé pour une distance donnée entre la cible et le drone.

Également, l'estimation de mouvement depuis une homographie suppose que la scène observée est un plan, ce qui n'est pas tout à fait juste dans notre cas. De plus, plus le drone est près de la cible plus l'hypothèse surface plane est douteuse et génère des erreurs supplémentaires dans la matrice d'homographie et donc dans l'asservissement. L'asservissement permet quand même, dans certains cas (distance pas trop faible, visibilité correcte et hypothèse surface plane envisageable) au drone de compenser l'impact de ses moteurs sur son mouvement et donc de se stabiliser face à la cible.

Des tests dans l'entreprise ont été effectués près de la coque des navires lorsqu'ils étaient à quai. La visibilité était trop faible pour permettre un asservissement visuel. Les tests se sont orientés pour essayer d'améliorer la visibilité avant de tester l'algorithme.

5 Conclusion

L'objectif de l'asservissement visuel était de compenser l'impact des moteurs pour permettre au drone de rester fixe par rapport à une cible. Cet objectif a été atteint. L'asservissement n'est néanmoins pas suffisant en l'état pour compenser des perturbations extérieures importantes par exemple un plongeur qui le décale.

L'asservissement visuel n'a pas besoin que la cible soit placée ou ait un pattern particulier, la seule hypothèse sur la cible est qu'elle soit assimilable à une surface plane.

L'erreur calculée par l'asservissement visuel est définie à une échelle près étant donné qu'on est en monoculaire. En connaissant la distance entre le mur et le drone, l'erreur calculée sera plus cohérente.

Les prochaines étapes sont de transférer l'algorithme pour être exécutable sur Nvidia Jetson TX2. L'utilisation de caméra basse luminosité, plus adaptée dans le milieu sous-marin est prévue. La matrice intrinsèque sera également recalculée dans un cas permettant un meilleur résultat. Des lampes seront utilisées afin d'avoir plus de visibilité sous l'eau et donc de pouvoir s'asservir de plus loin. La distance pourra également être obtenue que ce soit par l'utilisation d'un sonar ou d'une caméra stéréo.

Références

- [1] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb : an efficient alternative to sift or surf. *Proceedings of the IEEE International Conference on Computer Vision*, pages 2564–2571, 11 2011.
- [2] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. *Comput Conf Comput Vis*, 3951, 07 2006.
- [3] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief : Binary robust independent elementary features. *Eur. Conf. Comput. Vis.*, 6314 :778–792, 09 2010.
- [4] Bruce D. Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision (ijcai). 81, 04 1981.
- [5] Andrew Harlley and Andrew Zisserman. *Multiple view geometry in computer vision (2. ed.)*. Cambridge University Press, 01 2006.
- [6] Ezio Malis, Manuel Vargas. Deeper understanding of the homography decomposition for vision-based control. *[Research Report] RR-6303*, page 90, 2007.
- [7] R Fischler and M Bolles. Random sample consensus : A paradigm for model fitting with applications to image analysis and automated cartography. *Commun ACM*, 24 :619–638, 01 1981.
- [8] Musa Morena Marcusso Manhães, Sebastian A. Scherer, Martin Voss, Luiz Ricardo Douat, and Thomas Rauschenbach. UUV simulator : A gazebo-based package for underwater intervention and multi-robot simulation. In *OCEANS 2016 MTS/IEEE Monterey*. IEEE, sep 2016.