
Dynamic vision interface for TurtleBot

Auteur : Courtoy, Boris

Promoteur(s) : Franci, Alessio; Sacré, Pierre

Faculté : Faculté des Sciences appliquées

Diplôme : Master en ingénieur civil en informatique, à finalité spécialisée en "intelligent systems"

Année académique : 2023-2024

URI/URL : <http://hdl.handle.net/2268.2/20445>

Avertissement à l'attention des usagers :

Tous les documents placés en accès ouvert sur le site le site MatheO sont protégés par le droit d'auteur. Conformément aux principes énoncés par la "Budapest Open Access Initiative"(BOAI, 2002), l'utilisateur du site peut lire, télécharger, copier, transmettre, imprimer, chercher ou faire un lien vers le texte intégral de ces documents, les disséquer pour les indexer, s'en servir de données pour un logiciel, ou s'en servir à toute autre fin légale (ou prévue par la réglementation relative au droit d'auteur). Toute utilisation du document à des fins commerciales est strictement interdite.

Par ailleurs, l'utilisateur s'engage à respecter les droits moraux de l'auteur, principalement le droit à l'intégrité de l'oeuvre et le droit de paternité et ce dans toute utilisation que l'utilisateur entreprend. Ainsi, à titre d'exemple, lorsqu'il reproduira un document par extrait ou dans son intégralité, l'utilisateur citera de manière complète les sources telles que mentionnées ci-dessus. Toute utilisation non explicitement autorisée ci-avant (telle que par exemple, la modification du document ou son résumé) nécessite l'autorisation préalable et expresse des auteurs ou de leurs ayants droit.

Dynamic vision interface for TurtleBot

Master of Science in Computer Science and Engineering

Author : COURTOY Boris

Supervisors :
FRANCI Alessio
SACRÉ Pierre

Academic year 2023-2024

The goal of this master's thesis is to develop a ROS2 driver able to retrieve and communicate the different data captured by the event-based cameras from iniVation. The implementation includes a capture node which reads the data from a camera and publishes them to their corresponding topics, as well as two other nodes in charge of the visualization of the data. To assess the quality of the driver, we performed a latency analysis and ensured that it was able to transmit the data in a few milliseconds. Following this, we designed as a proof of concept a way to control a TurtleBot based on the output of the camera. Finally, we completely integrated the camera with the TurtleBot. Facing latency issues, we managed to show that the Raspberry Pi on which the TurtleBot is built was not powerful enough to process all the events captured by the camera. We eventually managed to overcome these problems by limiting the number of events processed by the Raspberry Pi.