



HAL
open science

ROS pour l'Analyse du Mouvement Humain et la Prévention des Risques Musculo-Squelettiques

Jessica Colombel, David Daney, Baptiste Busch

► **To cite this version:**

Jessica Colombel, David Daney, Baptiste Busch. ROS pour l'Analyse du Mouvement Humain et la Prévention des Risques Musculo-Squelettiques. Journées Nationales sur ROS, Jun 2018, Toulouse, France. hal-01955378

HAL Id: hal-01955378

<https://inria.hal.science/hal-01955378>

Submitted on 14 Dec 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

ROS pour l'Analyse du Mouvement Humain et la Prévention des Risques Musculo-Squelettiques

Jessica Colombel¹, David Daney¹ and Baptiste Busch¹

Abstract—ROS est utilisé afin de construire l'architecture d'un système *in situ* d'analyse du mouvement d'opérateurs industriels. Après avoir présenté le contexte, les performances attendues et l'architecture globale du système, nous exposerons plus particulièrement la façon dont nous gérons les traitements de données et les modules d'évaluation et de reconnaissance du geste.

I. INTRODUCTION

Les Troubles Musculo-squelettique (TMS) sont la première cause de maladie professionnelle en France, avec près de 80% de cas pour le régime général et 90% pour le régime agricole. L'entreprise AIO s'intéresse de près à ces problématiques. Dans le but d'améliorer la supply chain, AIO propose des solutions mécaniques astucieuses, nommées Karakuri Kaizen, permettant d'assister l'opérateur dans ses gestes et ainsi réduire la pénibilité au travail. Partant du constat qu'aucun système de mesures automatiques de la pénibilité n'était disponible et accessible en condition écologique, AIO a lancé le projet KOMBOS. Le premier produit, nommé Numii®(Figure 1), est un système composé d'une caméra RGB-D permettant de déterminer les postures des opérateurs et de calculer un indice ergonomique physique (risque de TMS). Afin de gérer la communication inter-process, Numii a été fait sur la base de ROS Kinetic.

L'architecture ROS a été choisie afin de garantir une certaine modularité dans notre système. Plus particulièrement nous avons pour objectifs de recevoir et fusionner n'importe quel type de données. Nous souhaitons mettre un place un système agile ayant la possibilité d'être utilisé en temps réel ou en calcul déporté. Nous envisageons également d'utiliser notre architecture pour de l'interaction humain-robot.

II. ARCHITECTURE

Notre utilisation de ROS s'inscrit dans une architecture système comprenant différents appareils en communication (Figure 2). Les capteurs envoient sous format json des données sur un réseau local émis par un ordinateur de traitement sous Linux. Celui-ci récupère et traite les données avec différents nœuds ROS avant d'envoyer les informations sur le réseau local à destination du Cloud et d'une application mobile. Cette dernière sert d'interface de visualisation des postures observées. Les packages ROS permettent ainsi la récupération de données brutes venant du ou des capteurs et les traitent avant visualisation des résultats par l'utilisateur.

*Ce travail a été supporté par l'entreprise AIO

¹ INRIA équipe AUCTUS

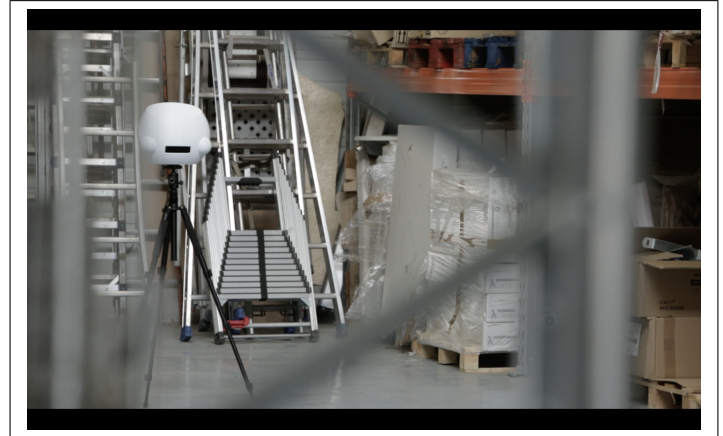


Fig. 1. Numii

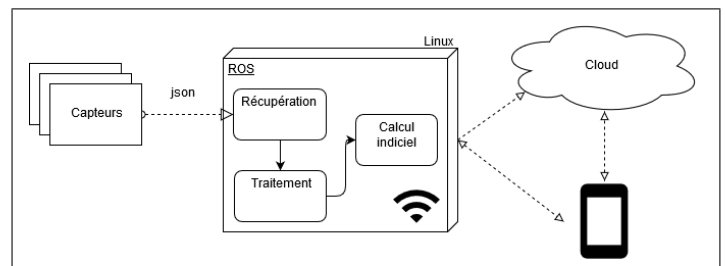


Fig. 2. Architecture globale du système Numii

Actuellement, le système fonctionne sous ubuntu 16.04 et ROS Kinetic. Nous avons développé des nœuds C++ ou python, en fonction des méthodes et besoins spécifiques. Les programmes sont répartis en trois packages : le premier comprend la récupération de données et les différents filtres, le deuxième les calculs indiciels et le troisième la définition des messages et services utilisés.

A. Traitement de données

La partie de traitement permet la fusion et l'analyse des données de différents types de capteurs en temps réel. A l'heure actuelle, les données sont principalement issues de capteurs RGB-D pré-filtrés (par OpenNI, SDK Microsoft, etc.). Nous avons ensuite fait le choix d'implémenter plusieurs filtres successifs. Le premier est un filtre dit à *1 euro* simple en terme d'implémentation et nécessitant peu de ressources computationnelles [1]. Le second est un filtre de *Kalman étendu* qui effectue une fusion des données provenant des multiples capteurs[3]. Le filtre de Kalman

est basé sur un modèle géométrique du corps humain permettant de passer en données articulaires, ce qui équivaut à une représentation minimale en nombre de paramètres de la posture d'une personne. Pour la fusion de données, nous avons mis en place une méthode de pré-fusion de données [2] permettant au système de fonctionner à 40Hz, au lieu de 8Hz avec une fusion classique.

L'architecture ROS a permis d'implémenter et de tester rapidement plusieurs filtres grâce à la modularité offerte par le middleware.

B. Calculs indiciels

Après avoir filtré les données, nous pouvons les traiter à plus haut niveau afin d'apporter une valeur sémantique aux postures observées. Nous avons ainsi implémenté des indices mécaniques et biomécaniques [5] ainsi que des indices ergonomiques comme le Rapid Upper Limb Assesment (RULA)[4] qui fournit une mesure entre 1 et 7 des risques de TMS liés à la posture observée. Ces indices sont ensuite transmis vers l'application pour informer l'utilisateur sur les risques liés à sa posture ou fournir un rapport plus détaillé à l'entreprise sur l'état de santé général de ses ouvriers.

Nous tenons également à prendre en compte des contraintes statiques et des contraintes d'équilibre qui nécessitent l'ajout de capteurs. Ceci est compatible avec l'architecture mais les phases d'identification des paramètres et l'intrusivité des capteurs peuvent freiner l'intérêt des utilisateurs. Nous cherchons donc plutôt à utiliser des apriori sur les tâches et des approximations de certains paramètres pour estimer les différentes contraintes.

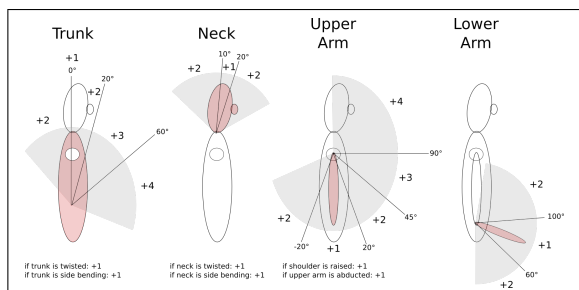


Fig. 3. Indices du Rula pour les membres supérieurs

C. Reconnaissance du geste

La reconnaissance d'actions est aussi une part importante du projet. Savoir que deux personnes effectuent une même action à forte valeur sémantique (e.g. visser, se déplacer...) permet une comparaison directe des indices calculés. Déterminer les gestes les moins pénibles permettant d'effectuer une tâche aura aussi un intérêt pédagogique lors de la formation des ouvriers.

Pour effectuer cette classification d'actions, nous avons entraîné un réseau à convolution à l'aide du framework PyTorch. Le réseau entraîné est ensuite déployé et reçoit les données "squelettes" transmises par message ROS.

L'intercompatibilité entre ROS et PyTorch n'est pas automatique du fait de l'absence de support python 3¹. A l'avenir, l'utilisation de ROS 2.0 devrait grandement simplifier ce problème.

III. PERSPECTIVES

L'architecture développée comprend de nombreux points à améliorer pour gagner en modularité. Pour cela, il sera probablement nécessaire d'effectuer une refonte totale ou partielle de l'architecture, dans le but de découpler et modulariser au mieux les traitements. Cette étape de refonte permettra, par ailleurs, le passage à ROS 2.0 ce qui simplifiera l'interfaçage avec les algorithmes d'apprentissage que nous avons développés. A terme, nous souhaitons également utiliser ROS comme base d'une architecture multi-agent pour améliorer l'analyse du mouvement humain.

BIBLIOGRAPHIE

- [1] G. Casiez, N. Roussel, and D. Vogel, "1 euro filter: A simple speed-based low-pass filter for noisy input in interactive systems," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM, 2012, pp. 2527–2530.
- [2] Q. Gan and C. J. Harris, "Comparison of two measurement fusion methods for kalman-filter-based multisensor data fusion," *IEEE Transactions on Aerospace and Electronic systems*, vol. 37, no. 1, pp. 273–279, 2001.
- [3] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.
- [4] L. McAtamney and E. N. Corlett, "Rula: A survey method for the investigation of work-related upper limb disorders," *Applied ergonomics*, vol. 24, no. 2, pp. 91–99, 1993.
- [5] A. Panchea, "Inverse optimal control for redundant systems of biological motion," Theses, Université d'Orléans, Dec. 2015. [Online]. Available: <https://tel.archives-ouvertes.fr/tel-01343186>.

¹Ce post internet résume bien les problèmes rencontrés lors de l'interfaçage entre ROS et PyTorch <http://scriptedonachip.com/pytorch-ros>.