

Modeling Human Behavior and State

A ROS package for Human Robot interaction

Bård-Kristian Krohg



Thesis submitted for the degree of
Master in Informatics: Robotics and Intelligent Systems
60 credits

Institute for informatics
Faculty of mathematics and natural sciences

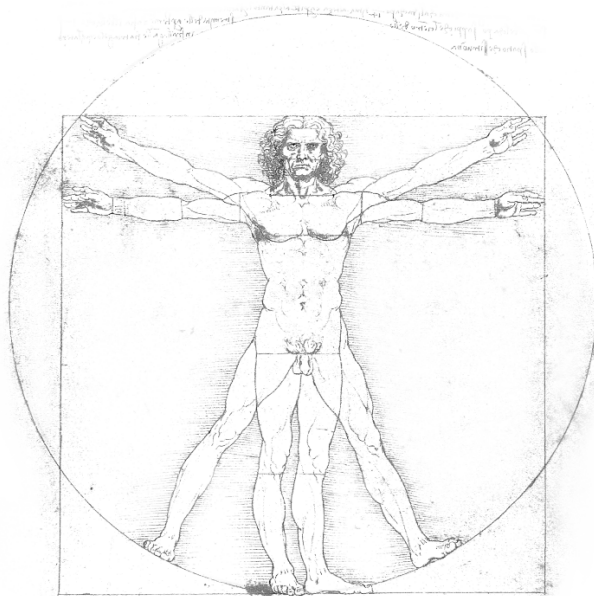
UNIVERSITY OF OSLO

Spring 2018

Modeling Human Behavior and State

A ROS package for Human Robot interaction

Bård-Kristian Krohg



© 2018 Bård-Kristian Krohg

Modeling Human Behavior and State

<http://www.duo.uio.no/>

Printed: X-press printing house

Modeling Human Behavior and State

Bård-Kristian Krohg

30th April 2018

Abstract

Short intro to the project (1/2 pages)

- what is it about (problem)
- what has been done to ready the problem (method, data)
- findings (main findings)
- precautions for the findings
- conclusion
- implications

This work implements an easy to use ROS package for human sensing and prediction for applications in geriatric care. Methods for sensing pose, respiration rate and heart rate are implemented. In addition, models for human activity recognition are implemented and tested. The system also features RoI extraction for some selected body parts.

Preface

TODO: When / Where / COINMAC / Kyushu University

Acknowledgements

I would like to thank Professor Jim Torresen and Vice Dean Ryo Kurazume for the opportunity to write this masters thesis at the lab here in Kyushu. TODO: friends, family Humantude Project [3]

This work is part of a Masters degree in Informatics: Robotics and Intelligent Systems at the University of Oslo. The work was written as a collaboration between the ROBIN lab at the University of Oslo, and the Kurazume Laboratory at Kyushu University as part of the exchange program COINMAC funded by The Norwegian Research Council.

A special thanks goes to my supervisors, Jim Tørresen and Ryo Kurazume, for both providing me with the fantastic opportunity to study abroad and for all their support and guidance throughout this project.

I would like to sincerely thank the staff and students at the laboratory for welcoming me to Japan, helping me with the administrative tasks in everyday life, including me in social activities and of course teaching me a little Japanese.

Many thanks goes to all my friends and family back in Norway for supporting me despite the vast distance and time difference. At last, I would like to thank my friends Karl Magnus, Soman, Mathias and the rest of the informatics students with whom I started at the university. Thank you for making my time here unforgettable.

Contents

1	Background	1
1.1	Motivation	1
1.2	Remote Vital Monitoring	1
1.3	Human Pose Estimation	2
1.4	Human Activity Recognition	2
1.4.1	Human habits	3
1.5	Technologies	3
1.5.1	Robotic Operating System	3
1.5.2	Red Green Blue Depth (RGB-D)	3
2	The Project	4
2.1	Human Robot Interaction Package for ROS	4
2.2	Human Pose Estimation	4
2.2.1	Formulas for constraining	4
2.2.2	Skeleton fitting	6
3	Experiments	7
3.1	Results	8
3.2	Discussion	9
4	Conclusion	10
5	Future Work	11
5.1	Human Pose	12
5.1.1	papers	12
5.1.2	datasets	13
5.2	Vital detection and mood	13
5.2.1	papers	13
5.2.2	datasets	14
5.3	Human Activity recognition	14
5.3.1	papers	14

5.3.2	datasets	14
5.4	Other	14
5.4.1	papers	14
5.4.2	datasets	15
Appendices		16
A	ROS package code	17

List of Figures

1.1	Numbering for OpenPose's keypoint markers	2
2.1	Constrained Lengths	5

List of Tables

1.1	IDs for OpenPose keypoint markers	2
2.1	Anthropometry constrain rules	5

Abbreviations

Nomenclature

Chapter 1

Background

1.1 Motivation

As life expectancy increases in Norway, so does the population who needs geriatric care either at home, or in a geriatric facility. In addition working in the health care sector with elderly care has not gained in popularity. This means that the time geriatric personnel already has must be spent more efficiently with the right people. We propose a novel robotic system that can monitor an elderly person (from here on called “the subject”) living at home and give feedback to healthcare personnel about their health. The system should also be able to detect and predict abnormal states of the subject, and warn the appropriate emergency services.

This work will focus on creating data that can be used to model the current state of the subject, as well as implementing models predicting future states. In this effort we wish to monitor the subject’s medical data such as respiration rate (RR) and heart rate (HR), daily activities and mood, as well as contextual input such as the ambient temperature, weather, nearby objects and current location. Machine learning models focusing on both short and long term prediction is discussed.

1.2 Remote Vital Monitoring

Vital data such as respiration rate (RR) and heart rate (HR) could be useful to detect whether a person is undergoing stress, is relaxed, or help paint a more complete picture of the subject’s current state. The temporal development of the vitals could also be valuable information to first responders in the event of an accident. In this project we will rely on eulerian video magnification [7] to extract this data. The color channel of a camera will be used to detect changes in the color of the skin at different keypoints on the face of a person to detect the HR. We also propose to use the depth channel of an RGB-D sensor to detect the rise and fall of a persons chest

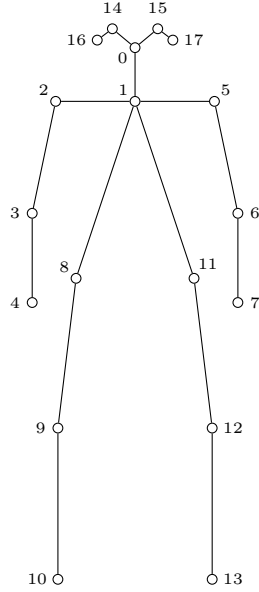


Figure 1.1: Numbering for OpenPose's keypoint markers.

ID	Name
0	Nose
1	Neck
2	Right Shoulder
3	Right Elbow
4	Right Wrist
5	Left Shoulder
6	Left Elbow
7	Left Wrist
8	Right Hip
9	Right Knee
10	Right Ankle
11	Left Hip
12	Left Knee
13	Left Ankle
14	Right Eye
15	Left Eye
16	Right Ear
17	Left Ear

Table 1.1: IDs for OpenPose's keypoint markers.

to estimate the RR.

1.3 Human Pose Estimation

A lot of work has already been done in detecting human pose, however our system has to work in real life environments where multiple different people can enter the scene. The system also needs to be able to run in real-time if rapid developments are happening. Therefore we will rely on the OpenPose [2] for initial 2D detection of people in the scene. However, to make the system robust to changes in perspective, so we wish to extract the 3D pose of each person using the depth information from an RGB-D sensor. A tracker should also be implemented to ensure object permanence and separate recorded data from multiple people.

1.4 Human Activity Recognition

In this part of the project we will compare Hidden Markov Models to Long Short Term Memory (LSTM) networks to recognize and predict different human activities

using datasets such as [5] and [4].

1.4.1 Human habits

We compare Hidden Markov Models to LSTM networks in an effort to detect abnormal patterns in the subject's daily/weekly routines.

1.5 Technologies

1.5.1 Robotic Operating System

The Robotic Operating System (ROS) is a open source framework for building robotic applications used by both academia and in an increasing degree by robots in the industry. ROS provides us with a wide variety of tools and libraries developed by specialized laboratories from around the world. These tools and libraries makes it easier to communicate with different sensors, ready libraries for computer vision and spacial transformations, visualize motion, simulate input and get an overview of the general architecture of the system. In this project we will for example use the `iai_kinect2` package [6] as well as the OpenCV library [1].

One of the most useful parts of ROS is that we can easily connect different programs together using ROS messages, topics and pipelines. We can therefore tie together the different parts of this system without much effort, and is why it was chosen as the framework for this project.

1.5.2 Red Green Blue Depth (RGB-D)

An RGB-D sensor was chosen for this project, as this kind of sensor is already widely used in robotic applications. This makes the package more portable, and can suit many different robot configurations. Using RGB-D data, we also get access to preexisting datasets for training our system. The Microsoft Kinect sensor was used in testing and development.

Chapter 2

The Project

2.1 Human Robot Interaction Package for ROS

A multipurpose package for tracking information about humans was made. The package features both detection of 2D joints as well as a method of manually fitting a whole 3D skeleton to the observed points.

2.2 Human Pose Estimation

The robot we're developing needs to be able to robustly detect humans, so information about their state can be gathered and analyzed. To accomplish this task, we propose a system that uses RGB-D data in combination with IR images to estimate the 3D pose of humans. In addition we propose a manual algorithm for estimation of occluded parts that can not be directly observed.

A lot of work has already been done on human pose estimation, however this approach focuses on creating a manual method for constraining the 3D skeleton and estimating the 3D pose of the observed person. We rely on 2D detection of joints for each person by the OpenPose software.

2.2.1 Formulas for constraining

For a fixed point a we want to move a point b so the Euclidean distance between them is equal to L . A few different methods was developed to accomplish this, based on the reliability of the different keypoints.

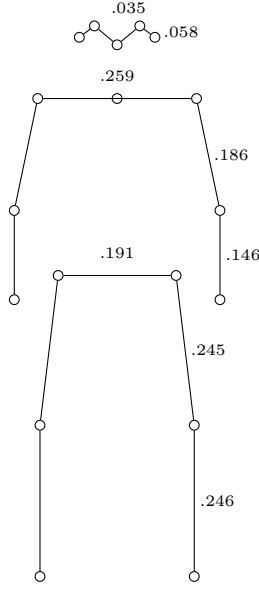


Figure 2.1: Constrained limb connections.

ID	Limb	Model	Pair
0	Left arm	.186	5 – 6
1	Left forearm	.146	6 – 7
2	Right arm	.186	2 – 3
3	Right forearm	.146	3 – 4
4	Hip	.191	8 – 11
5	Left thigh	.245	11 – 12
6	Left leg	.246	12 – 13
7	Right thigh	.245	8 – 9
8	Right leg	.246	9 – 10

Table 2.1: Constrain rules for correct anthropometry. (Note that the anthropometry for the head and shoulders are not yet implemented.)

Keypoint projection

If both \vec{a} and \vec{b} are well-observed points and the initial distance between the point \vec{a} and the line from the camera center through \vec{b} are less than L , we use the following formula to calculate the new position of \vec{b} . We get the length, x of the vector to the new position of \vec{b} by using equation 2.1. The constrained point is then simply defined as $x|\vec{b}|$.

$$x = \max \left(\frac{2(\vec{a} \cdot \vec{b}) \pm \sqrt{4(\vec{a} \cdot \vec{b})^2 - 4\|\vec{b}\|^2(\|\vec{a}\|^2 - L^2)}}{2(\|\vec{a}\|^2 - L^2)} \right) \quad (2.1)$$

However, if the minimum distance between \vec{b} and the line is more than L , we define the point as the point L away from point \vec{a} on the line through the coordinates of \vec{a} perpendicular to the line along \vec{b} . The point is then defined by equation 2.2.

$$\vec{p} = \vec{a} + L \left| \vec{a} - \frac{\vec{a} \cdot \vec{b}}{\vec{b} \cdot \vec{b}} \vec{b} \right| \quad (2.2)$$

Keypoint interpolation

This method could be used both to create one additional keypoint, or to determine the location of an obstructed keypoint. We assume we again have two well observed keypoints a and c . We wish to find the location of keypoint b . To interpolate, we imagine two spheres around point a and c with radiuses r_a, r_c . The interpolated point must be on the intersecting circle between the two spheres. The distance from point a to the plane of that circle is defined as $x = \frac{d^2 - r_c^2 + r_a^2}{2d}$ where d is the Euclidean distance $\|a - c\|$. The radius of the circle is defined as $h = \frac{1}{d} \sqrt{(-d + r_c - r_a)(-d - r_c - r_a)(-d + r_c + r_a)(d + r_c + r_a)}$. The keypoint b is then defined as the keypoint furthest away from the camera on this circle. In later versions this could be constrained by the angle between the previous limb and this one.

2.2.2 Skeleton fitting

We wish to fit a constrained skeleton to the observed points in 3D space. Our skeleton is defined as three separate graphs with constrained edges. The keypoints defining the nodes of the graphs (joints) are detected by the OpenPose network, and sorted based on the confidence of detection. For each graph a subset of n keypoints are picked as seeds, and constrained graphs are generated:

Scale (and thus limb length) is calculated based on the confidence of the keypoints using the weighted sum in equation 2.3 where L is the set of Euclidean lengths in each graph and S is the scores of those lengths. S is calculated by the Gaussian function of the confidence of the scores c_a and c_b with $\sigma = 0.33$ as described in equation 2.4.

$$S = \frac{\sum_{i=0}^N L(i) \cdot s(i)}{\sum_{i=0}^N s(i)} \quad (2.3)$$

$$s = \frac{1}{\sqrt{\sigma\pi}} e^{-\frac{1}{\sigma}(c_a-1)^2 + (c_b-1)^2} \quad (2.4)$$

Alternative notation for equation 2.4 because of possible cumbersome notation in e expression.

$$s = \frac{1}{\sqrt{\sigma\pi}} \exp \left(-\frac{1}{\sigma}(c_a-1)^2 + (c_b-1)^2 \right)$$

We then recursively constrain all points based on a seed point from the sorted keypoints. If no keypoints with confidence over a certain threshold t are detected, the graph is not placed. The constrained graph is moved to the center of the detected keypoint, again weighted by S , and we are finished constraining the graph.

The skeleton is defined as the subset of graphs best matching their respective keypoints where scale is similar.

Chapter 3

Experiments

What did we find out. dont overcomplicate the explanation. this could be the longest part of the thesis. about 15-20 pages? If you have more questions, use that as structure for this section. You can divide this into multiple chapters: subsidiary questions to the main theme, hypotheses, themes. One to three chapters are usually OK. the most important first, main findings. small nuances exceptions and discussions. discuss what youve found. this could be a chapter in itself.

3.1 Results

the main findings, as simply put as possible.

3.2 Discussion

Look at the results critically, weaknesses to the method. discuss how the findings can be explained. reasons to why you found what you found. how it compares to earlier research. can reference some earlier theory.

Chapter 4

Conclusion

About 10% of the length (means $\tilde{8}$ pages) often the only thing that is read by people who are just looking at the thesis.

- tell in short version what youve found. main findings first. short, simply put. the neuances and details can be fleshed out in the following sections.
- how your findings fit with earlier work and research. (dont repeat too much from the “earlier research” or “theory” chapters.) What fits, and suggestions as to why.
- The way your finings can have significance. Can we see the subject in a new way? should one change something in practice or how one does things because of your research? can the finds benefit society. Youre going to tell the world, and see what youre writing about in a bigger picture. Can other people learn something from this?

Chapter 5

Future Work

What research is missing, what do we want to know more about, what other methods should be tried out.

Preliminary Notes and sources

This chapter is not to be included in the final thesis. It is only here to provide easy access to relevant research papers while writing.

5.1 Human Pose

How to run NiTE2 on linux for comparison with windows software:

5.1.1 papers

Open Pose paper

- Multi view RGB-D approach for pose estimation

- Space-time representation of people based on 3d skeletal data

- Stacked hourglass networks for human pose estimation

- Baseline for human 3d pose estimation from 2d images Accompanying video

Github repo

- Mocap guided data augmentation for 3d pose estimation in the wild

- 2D human pose estimation

- Determination of 3D human body postures from a single view

- People detection and tracking using RGBD cameras for mobile robots Paper direct link

- Fast Human Detection in RGB-D Images based on color depth joint feature learning +RoI Extraction

- 3D skeleton-based body pose Recovery

- VNect real time 3D human pose estimation with single rgb camera Project page

- Skeletal graph based human pose estimation in real-time

- Human skeleton tracking from depth data using geodesic distances and optical flow

- Multi-modal Surface Registration for Markerless Initial Patient Setup in Radiation Therapy using Microsoft's Kinect Sensor

Accurate 3D pose Estimation from a Single Depth Image
3D Hand Skeleton Model Estimation from a Depth Image
Key Developments in Human Pose Estimation for Kinect
Single-Shot Multi-Person 3D body pose estimation from monocular rgb input
3D human pose estimation = 2D pose Estimation + Matching
LCR-Net: Localization-Classification-Regression for Human Pose
Image-based Synthesis for Deep 3D Human Pose Estimation
The Vitruvian Manifold: Inferring Dense Correspondences for One-Shot Human Pose Estimation
Learning to recognize affective body postures
Keep it SMPL: Automatic Estimation of 3D Human Pose and Shape from a Single Image

Skeleton Fitting

Skeleton Fitting Techniques for optical motion capture
Progressive Human Skeleton Fitting
Learning Inverse Kinematics for Pose-Constraint Bi-Manual Movements
Local and Global Skeleton Fitting Techniques for Optical Motion Capture
3D Body Pose Estimation Using an Adaptive Person Model for Articulated ICP
Paper

Head tracking

Face direction using depth maps
Detecting Human Heads with their orientation
POSEidon Face From Depth for Driver pose Estimation
Face Tracking in OpenCV
3D Face reconstruction from 2D images using nn

5.1.2 datasets

Human3.6m dataset for pose Berkeley MHAD (Multimodal Human Action Database)

5.2 Vital detection and mood

5.2.1 papers

Heart Rate detection using Microsoft Kinect Full text
Detecting Heart Rate with Kinect v2 Github repo

Non-contact, Wavelet-based Measurement of Vital Signs using Thermal Imaging

Eulerian Video Magnification for Detecting Subtle changes in the World

5.2.2 datasets

Radboud Faces Database (Emotions)

5.3 Human Activity recognition

5.3.1 papers

Human Activity Recognition system using skeleton data from rgb-d sensors

Tracking a Subset of Skeleton Joints: An Effective Approach towards Complex Human Activity Recognition

Human Motion Analysis from Depth data

A simple neural network module for relational reasoning

5.3.2 datasets

Human Activity Detection project at Personal Robotics Lab at Cornell University
github repo Dataset by this lab Results

5.4 Other

IAI Kinect2

Facebook's Detectron github repo

5.4.1 papers

Distilling a Neural Network to a soft Decision Tree

Generative Adversarial Nets (GANs)

(RGB-D Indoor Plane based 3d modeling using autonomous robot)

Q learning

Real Time range imaging in health care A survey pdf:

Deep Visual-Semantic Alignments for Generating Image Descriptions

5.4.2 datasets

[ETHZurich CVL datasets](#)

[List of RGBD datasets](#)

[Databases 4 kinect](#)

Appendices

Appendix A

ROS package code

The MECS monitor package is the final culmination and implementation of this work.

All packages are different stages of development. They are only here for reference for now. Remember to remove in final text

- [MECS monitor package](#)
- [RGBD pose](#)
- [KinOP Bridge](#)

Bibliography

- [1] G. Bradski. ‘The OpenCV Library’. In: *Dr. Dobb’s Journal of Software Tools* (2000).
- [2] Zhe Cao et al. ‘Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields’. In: *CVPR*. 2017.
- [3] Miwako Honda et al. ‘Reduction of Behavioral Psychological Symptoms of Dementia by Multimodal Comprehensive Care for Vulnerable Geriatric Patients in an Acute Care Hospital: A Case Series’. In: *Case Reports in Medicine* (2016).
- [4] Catalin Ionescu et al. ‘Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments’. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36.7 (July 2014), pp. 1325–1339.
- [5] Jaeyong Sung et al. ‘Unstructured Human Activity Detection from RGBD Images’. In: *International Conference on Robotics and Automation (ICRA)*. 2012.
- [6] Thiemo Wiedemeyer. *IAI Kinect2*. https://github.com/code-iai/iai_kinect2. Accessed June 12, 2015. University Bremen: Institute for Artificial Intelligence, 2014 – 2015.
- [7] Hao-Yu Wu et al. ‘Eulerian Video Magnification for Revealing Subtle Changes in the World’. In: *ACM Transactions on Graphics (Proc. SIGGRAPH 2012)* 31.4 (2012).