Recognition of Human Behavior Patterns Using Depth Information and Gaussian Feature Maps

Jens Spehr, Mensur Islami, Simon Winkelbach, Friedrich M. Wahl

{j.spehr, m.islami, s.winkelbach, f.wahl}@tu-bs.de

Abstract: The representation of human behavior patterns is challenging due to the complex dependencies between features gathered by a sensor and their spatial and temporal context. In this work we propose a new Gaussian feature map representation that uses the Kinect depth sensor, can easily be integrated in home environments, and allows learning unsupervised behavior patterns. The approach divides the living space into grid cells and models each grid cell with a Gaussian distribution of features like height, duration, magnitude and orientation of the velocity. Experimental results show that the method is able to recognize anomalies regarding the spatial and temporal context.

1 Introduction

Due to our aging society the support of elderly people in their home environments is of increasing interest. While the number of elderly people strictly increase, the number of people being able to care about the elderly decrease. As a consequence, the number of free places in residentials or nursing homes is far lower than the demands. One promising way to solve this dilemma is to build smart homes, that analyze automatically the human behavior patterns (HBP) and detect changes of the activities of daily living. These changes typically indicate emerging health problems.

We will now briefly review related work. Surveys about human behavior understanding could be found in e.g. [PPNH06, AIA10]. Veeraraghava et al. [VMRc05] analysed human movements by matching shape sequences. Their nonparametric model is based on Dynamic Time-Warping and is applied to gait-based human recognition. Oikonomopoulos et al. [OPP11] addressed the problem of localization and recognition of human activities in unsegmented image sequences. They proposed a method that is based on an implicit representation of the spatiotemporal shape of the activity by means of ensembles of feature descriptors. During evaluation, they were able to successfully apply their approach to sequences with a significant amount of clutter and occlusion. Chen et al. [CAA11] proposed a learning approach for HBP in work environments. They used simple motion history images to represent the actions and the k-nearest neighbors classifier to assign the actions to a class. Their HBP are finally defined by frequent sets of actions. Aztiria et al. [AIBA09] proposed a learning approach for common behaviors of the person in an intelligent environment based on speech recognition. Their HBP are defined using simple "if", "then" conditions. Another relevant work is proposed by Lühr et al. [LWV07]. They

used intertransaction association rules [LFH00] to detect anomalous behaviour in smart homes. Although the previous mentioned approaches solve parts of the challenging HBP representation problem they are still restricted to simple scenarios and the successful application in a real home environment is questionable.
The paper is organized as follows. In Sec. 2 we will briefly describe the Kinect sensor that is used in our human behavior recognition framework as the main sensor. We then motivate our model in Sec. 3 and describe it in more detail in Sec. 4. After presenting experimental results (Sec. 5) we conclude with a discussion in Sec. 6.

2 Depth Sensor

The Kinect sensor [Mic12] was originally designed as a more natural, controller-free game input. It contains, among others, a color camera and a depth sensor. The color camera delivers RGB color images, whereas the depth sensor is an ensemble of an infrared projector of a preset pattern and an appropriate CMOS sensor. In the following, we will use the terms *depth sensor* and *camera* interchangeably to avoid the risk of misunderstanding. The depth information is derived from the difference between the projected and the reflected IR-pattern.
Kinect has some properties that make it suitable for our purpose. For example it works equally well at daylight, as well as in darkness. It also delivers fairly precise 3D data, which allows for position determination of the person in space. Beside the raw color and depth data, it is possible to retrieve segmentation information provided by the OpenNI [Ope12] Framework, and thus discriminate persons from the background. The same framework allows the determination of other features as well, such as (body) pose detection, gesture recognition, etc., however those features have not been used in this work.

3 Human Behavior Patterns (HBPs)

Behavior patterns are defined by a specific order of sequential, as well as parallel activities. Due to the complexity of these sequences, an appropriate representation is challenging if not missing. Models like Hidden Markov Models (HMMs) [YOI92], or more complex models like hierarchical HMMs [Fin98], layered HMMs [Fin98], coupled HMMs [BOP97] or Bayesian networks [RR05] try to encode these structural dependencies into a graph representation. However, their robustness generally suffers from wrong assumptions, or inaccurate model structures. We therefore propose to use a simplified approximation, which tries to set local features into a spatial and temporal context.

The aim of our vision system is to recognize the presence of one or more persons within an area of interest. Furthermore, it should be able to recognize given HBP, such as sitting, walking to the door, lying down on the bed, etc., but also abnormal behavior patterns such as falling down or motionless lying on the floor.
Generally, a person's behavior within his/her living space is not uniform. In the course of

time, behavior patterns become apparent. Certain areas are for example occupied by obstacles such as furniture pieces and the person must usually walk around or in between them. Certain activities are usually bound to specific areas, such as watching TV while sitting on the couch, sleeping in a lying position on the bed, walking along established pathways, standing in front of kitchen appliances etc. So, there are not only spatial variations concerning duration, but also the concerning features like direction, speed and acceleration of the person's movement which tends to be correlated to specific regions of the living space.

4 Model for HBPs

In a living environment, one or more persons perform different activities such as walking to the door, sitting down on the chair, lying down or sleeping on the bed, eating at table, which are deemed to be normal behavior patterns. However, there are also other patterns such as abrupt falling, motionless lying on the floor, etc. which can be classified as abnormal behavior patterns. A system would have to be able to easily cope with the proper identification and classification of those activities. A context-aware model as described above is a difficult challenge to overcome and it gets even more complicated if new patterns need to be learned.

Dividing the living space into a grid of cells allows for context independent learning and recognizing of HBPs.

4.1 Calibration of Extrinsic Sensor Parameters

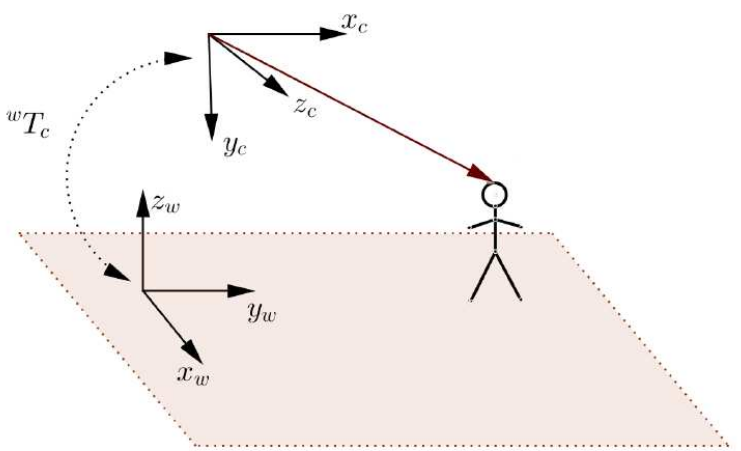

Figure 1: Camera vs. world coordinate representation

In order to build our map representation we need to transform the 3d data captured by the depth sensor, and thus defined in the camera coordinate system, into the world coordinate system. For that, we have to determine the camera-to-world coordinate transformation ${}^wT_c$ as depicted in Fig. 1. The x- and y-axis of the world frame lie in the ground plane and the z-axes points upwards. In order to determine the transformation ${}^wT_c$ we use a semi-automatic calibration routine. The person has just to label the ground plane within the camera image. The corresponding depth information is then used to fit automatically a plane into the 3d data and thus get the relative transformation between the camera and the plane. The fitting is performed using the RANSAC approach [FB81]. An example can be seen in Fig. 2 where the input and the outcome of the camera to world transformation is shown.

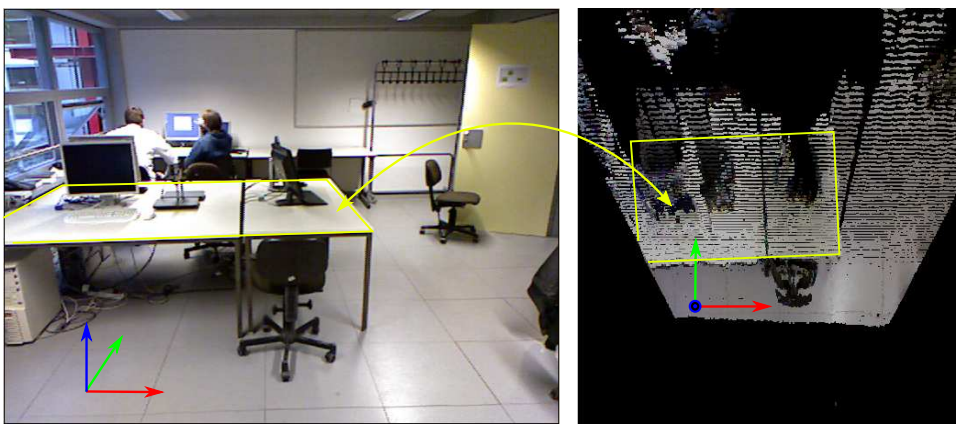

Figure 2: left - The area of interest as seen by the sensor; right - The same area as seen 'from above'

4.2 Local Features

In this section we describe the features of our representation. We assume that the set of all 3d points of person p at time step n is denoted as $\mathcal{P}_n^i$ and that the 3d points are already transformed into the world coordinate system as described before. We are interested in the person's height $\mathbf{h}_n^p$, respectively the position of the highest pixel on the persons label. Finding this pixel is a matter of testing all the label pixels and finding the one with the maximal height component.

$$\mathbf{h}_n^p = \arg \max_{\hat{\mathbf{h}}_n^p \in \mathcal{P}_n^p} \hat{\mathbf{h}}_{n,z}^p \tag{1}$$

The two primitive features from the Kinect are the persons' head $position$ in space and the timestamp of the data access. Let us analyze the person's movement. We denote the position of the p^{th} person's head at the moment t_n as a 4d vector

$$\mathbf{p}_n^p = [\mathbf{h}_n^p, t_n] \tag{2}$$

Let the previous reading of the same feature occur at the moment t_{n-1}, so

$$\mathbf{p}_{n-1}^{p} = \left[\mathbf{h}_{n-1}^{p}, t_{n-1}\right] \tag{3}$$

We can easily calculate the components of the $velocity$ vector

$$\mathbf{v}_{n}^{p} = \frac{\mathbf{h}_{n}^{p} - \mathbf{h}_{n-1}^{p}}{t_n - t_{n-1}} \tag{4}$$

as well as the $acceleration$ vector

$$\mathbf{a}_{n}^{p} = \frac{\mathbf{v}_{n}^{p} - \mathbf{v}_{n-1}^{p}}{t_n - t_{n-1}} \tag{5}$$

as the first and the second derivative of the position in time. The velocity and the acceleration are approximated with a frame rate of 30 Hz. We also use a counter variable to count the time intervals in which the user hasn't moved. The counter resets itself to zero each time the user enters a cell and counts the time units up for as long as he stayed within that cell. This provides us with $duration\ d$ as a further feature. As we will see in the following section, a feature vector measured in time step n

$$\mathbf{f}_{n,i,j}^{p} = \left(\mathbf{h}_{n,z}^{p}, \mathbf{v}_{n}^{p}, \mathbf{a}_{n}^{p}, d\right) \tag{6}$$

is associated to a grid cell (i, j).

4.3 Gaussian Feature Maps

The Gaussian feature map is represented by a matrix of Gaussian distributions

$$\mathcal{N}_{ij}(\mathbf{f}_{n,i,j}; \mu_{\mathbf{i,j}}, \widehat{\mathbf{\Sigma}}_{i,j}) \tag{7}$$

with $i = 1, ..., N$ and $j = 1, ..., M$. Each element of the matrix is associated to a grid cell defined in the world coordinate system. The mean feature vector $\mu_{\mathbf{i,j}}$ and the covariance matrix $\widehat{\mathbf{\Sigma}}_{ij}$ of the feature data are calculated according to Eq. (8) and (9). To illustrate the information that μ_{ij} and $\widehat{\mathbf{\Sigma}}_{ij}$ deliver to us, let's consider the two following situations for a single person:

Sitting area: The height component of the mean feature vector μ has a lower and the corresponding variance σ a higher value. The explanation is simple: when the person sits, his head is at a lower height, while before sitting down and after standing up his head reaches its maximum height. That is why the value of μ lies between the lowest head position (sitting) and highest head position (standing). The value of σ on the other hand is larger, due to the variations of the person head's positions.

Walking area: The height component of the mean feature vector μ has a higher value and σ a lower value. As the person is walking, his head position oscillates with a relatively small amplitude around his normal body height. That is why μ has a higher value, quite close to the persons height. The value of σ is related to the said amplitude, i.e. it's relatively small.

4.4 Learning of Gaussian Feature Maps

The parameters of the Gaussian distributions are estimated during an offline training step. Ideally,we capture a number of L samples during training for each grid cell. We can then estimate the model parameter as follows. The mean can be calculated as

$$\mu_{i,j} = \frac{1}{L} \sum_{n=1}^{L} \mathbf{f}_{n,i,j} \tag{8}$$

the covariance

$$\widehat{\mathbf{\Sigma}}_{i,j} = \frac{1}{L-1} \sum_{n=1}^{L} (\mathbf{f}_{n,i,j} - \mu_{i,j})(\mathbf{f}_{n,i,j} - \mu_{i,j})^{\mathrm{T}}. \tag{9}$$

Unfortunately, it is very unlikely, that the person accesses all grid cells uniformly and furthermore it is likely that not all behavior patterns were present during training. To deal with these issues and in order to abbreviate the training phase, we additionally use an online learning phase. For that, the Gaussian model of each grid cell is adapted to new samples, that are online incorporated into the model. Each cell has a feature set, where it can hold up to a predefined number of the most recent spatial and temporal feature data pertaining to that cell only. Every time a new reading of the cell features takes place, the newly read values are pushed into the set. If the predefined set size is exceeded, then the oldest record in the set is pulled out and discarded. Based on the feature records stored in the set we can calculate the mean and standard deviation value for each of the features.

4.5 Recognition of HBPs

The Gaussian model allows us to determine the likelihood of a current measurement, that is the feature vector $\mathbf{f}_{n,i,j}$, given our model. We use a predefined standard deviation threshold σ to decide if the current sample is normal or not, typically three standard deviations 3σ.

In order to robustly detect anomalies, or more precisely unseen patterns, we additionally use a temporal filter to reduce the number of false positives.

5 Experimental Results

We conducted a series of tests to see how robust our model is at detecting and learning HBPs. For the purpose of this paper, we only run tests with the height and duration features. However, the same idea also works for the velocity and acceleration vector components.[1]

Walking Area: In this test the person simulates walking along a typical pathway at a normal speed. During the walk, his height does not significantly change, so we expected the

[1] Velocity and acceleration vectors are constrained to their absolute values.

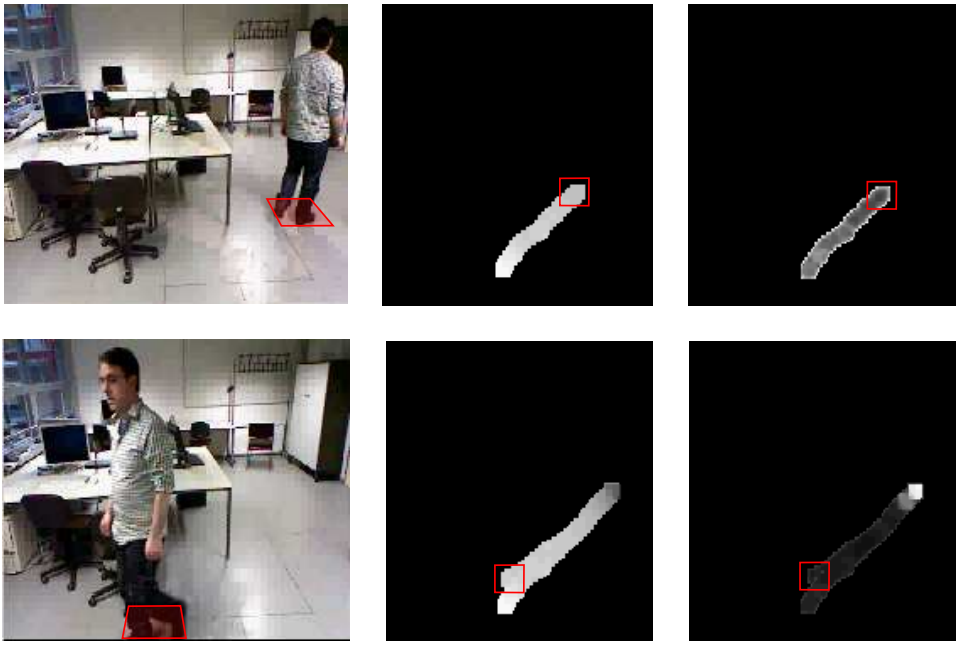

Figure 3: Input image (left), height (center) and variance (right) along a walking pathway (bright values indicate high height and variance values).

average height μ to be relatively close to the person's height, and a relatively small variance. An example is shown in Fig. 3. As expected the height distribution was relatively uniform along the walking path, corresponding to the persons height. The variance is also for the most part constant. At the far right end of both visualization patterns, we see a faulty height reading, leading to a smaller value for μ respectively a higher one for σ.

Sitting Area: In the next test the person walked to the chair and sat down on it. Subsequently he stood up and sat down again. We expected the value of μ to lie somewhere between the person's standing and sitting height. Further, we expected the value of σ to be proportional to the height difference between standing and sitting.
As Fig. 4 shows, in the sitting area the average of the height is depicted with a darker spot (lower value), where as the value of the variance at the same coordinates is greater then the rest of the pattern, which makes the spot clearly brighter.
Duration: Duration (time a person stays in a cell) is not as easy to depict in an image. We defined an expiration period and the method alert when the person stayed longer than that time over a certain cell. This test was 100% positive.

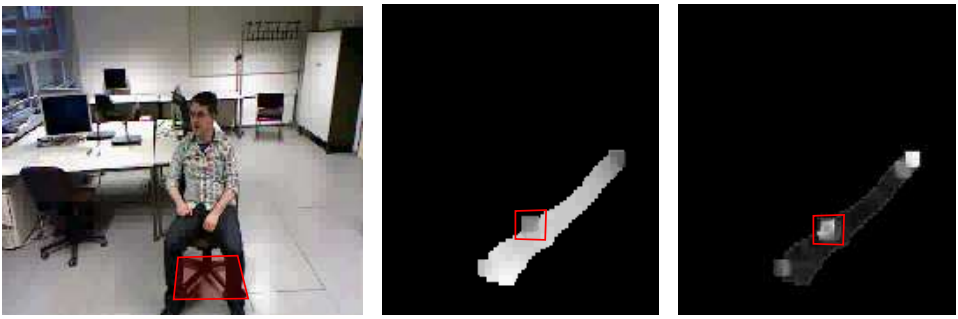

Figure 4: Input image (left), height (center) and variance (right) on a sitting area

5.1 Anomaly Detection Results

We ran tests for three different scenarios:

- Falling while walking
- Sitting down
- Longer pause in a place

In those three scenarios, we wanted to analyze the performance of the system by comparing the number of the correct detections with that of incorrect ones. We recorded the person performing the tests along with the relevant readings by the computer. This allowed for a repeatable analysis of the detection results.
Falling: During this test, the person simulates falling by ducking down thus lowering his height. We ran two different tests with results as depicted in Figure 5 and used a longer training sequence in the first series. As can be seen in the first series we get:

$$\text{true positive rate } tp_I = \frac{TP}{P} = \frac{12}{12+2} \approx 0.87 \tag{10}$$

$$\text{precision } pr_I = \frac{TP}{TP+FP} = \frac{12}{12+6} = 0.67 \tag{11}$$

$$\tag{12}$$

and for the second series:

$$\text{true positive rate } tp_{II} = \frac{TP}{P} = \frac{26}{26+3} \approx 0.90 \tag{13}$$

$$\text{precision } pr_{II} = \frac{TP}{TP+FP} = \frac{26}{26+5} = 0.84 \tag{14}$$

$$\tag{15}$$

More often then not, a ”fall” would provoke multiple (2,3) alarms, which is due to the distribution of the height value over the neighboring cells. Nevertheless, these detections

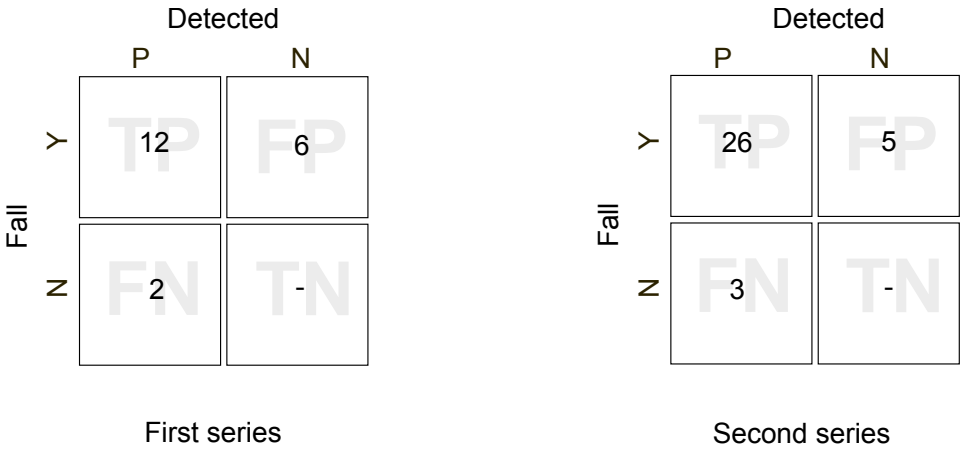

Figure 5: Fall detection contingency tables

count as positive, as we were interested in detecting an abnormal person height.
Quite a few falls passed undetected. The reason may be a previous large σ on the concerning cell or a hardware-software bottleneck. Finally, some falls were detected without actually a fall happening. In this case the reason is a relatively small previous σ or a correct detection which has been delayed by the hardware and/or the software.
Sitting: Sitting down for the first time was falsely detected as a fall, however in all subsequent cases it did not cause any more false alarms. It's worth mentioning here that the system adapts to the changes by constantly recalculating the mean μ and the standard deviation σ. That is why we can only make the system "believe" one time that a fall has occurred instead of sitting down. As we sit down, the mean value and the standard deviation get updated accordingly, so for the subsequent times sitting down is deemed as a normal behavior.
Staying in Place: Longer pause in a place was faultless every time.

6 Conclusion

In this paper we presented a new Gaussian feature map representation using the Kinect depth sensor, that can easily be integrated in living environments and allows for learning of unsupervised behavior patterns. After giving a brief introduction, we continued by describing the technical characteristics of the Kinect depth sensor that we used in our framework. We gave a definition of the human behavior patterns and a narrower definition of our task. We presented the model and the finer details (features, Gaussian feature maps, cell feature sets, learning, recognition, etc.) of the task. Our tests showed that the method presented is able to recognize the anomalies in regard of the spatial and temporal context.

7 Acknowledgements

This research has been done in the Lower Saxony research network "'Design of Environments for Ageing"' (GAL). We acknowledge the support of the Lower Saxony Ministry of Science and Culture through the "'Niedersaechsisches Vorab"' grant programme.

References

[AIA10] Asier Aztiria, Alberto Izaguirre, and Juan Carlos Augusto. Learning patterns in ambient intelligence environments: a survey. *Artif. Intell. Rev.*, 34(1):35–51, June 2010.

[AIBA09] Asier Aztiria, Alberto Izaguirre, Rosa Basagoiti, and Juan Carlos Augusto. Learning about preferences and common behaviours of the user in an intelligent environment. In *BMI Book*, pages 289–315, 2009.

[BOP97] M. Brand, N. Oliver, and A. Pentland. Coupled hidden Markov models for complex action recognition. 0:994, 1997.

[CAA11] Chih-Wei Chen, Asier Aztiria, and Hamid Aghajan. Learning Human Behaviour Patterns in Work Environments. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 47–52, Colorado Springs, CO, June 2011.

[FB81] Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, June 1981.

[Fin98] Shai Fine. The hierarchical hidden markov model: Analysis and applications. In *Machine Learning*, pages 41–62, 1998.

[LFH00] H.J. Lu, L. Feng, and J.W. Han. Beyond Intra-Transaction Association Analysis: Mining Multi-Dimensional Inter-Transaction Association Rules. *ACM Transactions on Information Systems*, 18(4):423–454, October 2000. Imported from EWI/DB PMS [db-utwente:arti:0000003302].

[LWV07] Sebastian Lühr, Geoff West, and Svetha Venkatesh. Recognition of emergent human behaviour in a smart home: A data mining approach. *Pervasive Mob. Comput.*, 3(2):95–116, March 2007.

[Mic12] Microsoft. Introducing Kinect for Xbox 360. `http://www.xbox.com/en-GB/kinect`, 2012. [Online; accessed 23-April-2012].

[Ope12] OpenNI. OpenNI TM. `http://openni.org`, 2012.

[OPP11] A. Oikonomopoulos, I. Patras, and M. Pantic. Spatiotemporal Localization and Categorization of Human Actions in Unsegmented Image Sequences. *IEEE Transactions on Image Processing*, 20(4):1126–1140, April 2011.

[PPNH06] Maja Pantic, Alex Pentland, Anton Nijholt, and Thomas Huang. Human Computing and Machine Understanding of Human Behavior: A Survey. In *SURVEY, PROC. ACM INTL CONF. MULTIMODAL INTERFACES*, pages 239–248, 2006.

[RR05] N.M. Robertson and I.D. Reid. Behaviour understanding in video: a combined method. volume 1, pages 808–814, Beijing, Chine, 2005.

[VMRc05] Ashok Veeraraghavan, Student Member, and Amit K. Roy-chowdhury. Matching shape sequences in video with applications in human movement analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27:1896–1909, 2005.

[YOI92] J. Yamato, J. Ohya, and K. Ishii. Recognizing human action in time-sequential images using hidden Markovmodel. pages 379–385, 1992.