Workshop Proceedings of the 8th International Conference on Intelligent Environments
J.A. Botía et al. (Eds.)
IOS Press, 2012

doi:10.3233/978-1-61499-080-2-213

Real-Time Human Pose and Gesture Recognition for Autonomous Robots Using a Single Structured Light 3D-Scanner

Tim VAN ELTEREN and Tijn VAN DER ZANT
Cognitive Robotics Laboratory, Department of Artificial Intelligence, Faculty of Mathematics and Natural Sciences, University of Groningen, Groningen, The Netherlands

Abstract. We propose a method for real-time human pose and gesture recognition for autonomous robots using a structured light 3D-scanner. Poses are recognized using skeleton representations by performing classification using the Nearest Neighbour algorithm. The whole-body pose recognition approach uses the joint coordinate data from the processed depth images. The quality of the classification is determined by 10-fold cross validation in which the recognition rate is 99.9028%.

Keywords. machine learning, artificial intelligence, pose recognition

Introduction

Robust and realtime tracking of a person's body has applications in many domains such as human-computer interaction interfaces, telepresence and monitoring for security and healthcare. Another challenging and interesting application, which will be the focus of this paper, is the field of Human-Robot Interaction (HRI) for autonomous robots. The introduction of realtime depth aware cameras has made this challenge somewhat easier but the state of the art systems still have their limitations.

The recent introduction of depth cameras using a structured light 3D-scanner approach, such as the Kinect Sensor System[1], brings realtime human pose recognition at consumer prices.

This paper covers the recognition of custom defined poses and gestures by an autonomous robot as part of a multimodal HRI system to autonomously perform tasks within the RoboCup@Home competition[2]. The RoboCup@Home competition [1] is an international benchmark for domestic service robots. It aims to develop service and assistive robot technology with high relevance for future

[1]Microsoft Corp. Redmond WA. Kinect for Xbox 360
[2]`http://www.ai.rug.nl/robocupathome/`

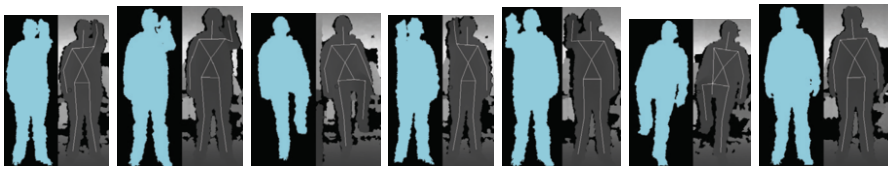

Figure 1. Seven examples of the set of seventeen classes of poses

personal domestic applications [2]. The performance and abilities of the robots in the competition are benchmarked using a series of tests. These tests all take place in a realistic non-standardized home environment that does not contain any artificial markers [3].

Natural interaction methods without the use of artificial markers are of special importance and relevance because of the applications for use of robots in real world domestic environments. The multimodal HRI system consisting of the combination of a speech recognition and markerless gesture recognition system is an integral part of the behavior-based architecture [4] [5] that has been developed at the Cognitive Robotics Laboratory of the University of Groningen[3].

Human pose estimation is an active area of research that has delivered a vast amount of literature surveyed in [6] and [7]. The advances made by [8] in real-time identification and localization of body parts from depth images and the research performed by [9] in 3d model based tracking approaches for human motion capture in uncontrolled environments show an object recognition and respectively a modeled approach. The vision based motion capture and analysis described in [10] performs real-time motion capture using a single time-of-flight camera.

The research performed by the Microsoft Research Cambridge and Xbox Incubation on real-time human pose recognition in parts from single depth images [11] forms the basis for our research on the recognition of human poses for natural HRI with an autonomous robot.

The depth images received from a structured light 3D-scanner such as Kinect are the result of an algorithm that performes dense 3D image acquisition using structured light [12] with a pattern of projected infrared points. The deformation of a speckle pattern projected on the scene, with respect to a reference pattern, reveils information about the distance of the objects and results in a calibrated depth mapping of the scene.

1. Methods

3D joint position data of a test subject is used to classify the pose. The performance of a total of twenty-six machine learning algorithms are used to perform classification on the dataset to perform pose recognition. Ten-fold cross validation is performed to determine the quality of the classification.

[3] http://www.ai.rug.nl/crl/

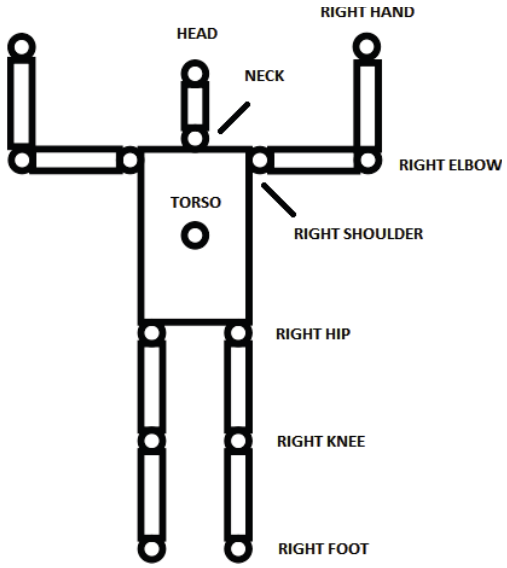

Figure 2. The circles in the skeleton represent the fifteen degrees of freedom or joints used for the classification of poses

1.1. Data

In our research we use the Kinect sensor system with the PrimeSense OpenNI[4] framework and the NITE[5] middleware which gives us access to the 3D-coordinates of the joints position of a calibrated test subject. Using this setup we create a dataset for machine learning that consists of the test subjects in a number of natural poses. The data that the poses consist of are fifteen 3D coordinates as shown in figure 2.

The current implementation uses the position data of fifteen joints. This implementation draws the corresponding skeleton over the depth map of the scene as can be seen in figures one, three and five. The poses used for training are shown as a person segmented on the left and the skeleton overlay over the depth image on the right.

The seventeen classes that are defined and used for classification cover a wide range of natural human poses. The complete set of classes are shown in figure one, three and five. Classification was performed using the WEKA[6] machine learning toolkit. The approaches used and their resulting performance are listed in table 1. Initial training of the classifier was performed with the joint data pose representations of over 26000 poses from four different test subjects.

In a follow up experiment the training of the classifier was performed with the joint data pose representations of 46863 poses from fifteen different test subjects. Building the dataset takes about fifteen minutes per person resulting in a total time to build the larger dataset of about four hours.

[4] http://www.primesense.com/en/openni
[5] http://www.primesense.com/en/nite
[6] http://www.cs.waikato.ac.nz/~ml/weka/

Figure 3. Five examples of the set of seventeen classes of poses

In addition to the aformentioned datasets a random subset consisting of 1% and 10% of the dataset consisting of 46863 poses was used for training to determine which algorithms are the most suitable for online implementation based on the execution time of the 10-fold cross validation.

1.2. Machine learning

The data is processed by a number of classification algorithms. The five best performing algorithms are shown in figure 4. A selection of the benchmarked machine learning algorithms are covered in more detail in the following subsections.

1.2.1. 1-Nearest Neighbour

The Nearest Neighbour algorithm [13], a type of instance based learning, uses past data instances, with known output values, to predict an unknown output value of a new data instance. Normalized Euclidean Distance was used as distance measure with a k value of 1.

1.2.2. Random Forest

The Random Forest consists of an ensemble of Tree-based classification algorithms in which the best performing classifier is selected and used for classification of the test data. The default parameters are used.

1.2.3. Random Tree

The Random Tree algorithm is a variant of the REPTree algorithm, Reduced Error Pruning Tree, which in turn is a type of Tree classifier. The default set of parameters were used and already showed sufficient performance.

1.2.4. J48

The J48 algorithm [15], also known as C4.5 algorithm, is an algorithm used to generate a decision tree which in turn is used for classification. Performance of the algorithm is tested using the default parameter set.

1.2.5. Voting Features Interval

Classification by Voting Features Interval [14] is performed in which a concept is represented by a set of feature intervals on each feature dimension separately. Each feature participates in the classification by distributing real-valued votes among classes. The class receiving the highest vote is declared to be the predicted class.

Figure 4. Five examples of the set of seventeen classes of poses

1.3. Ten-Fold Cross Validation

On all the algorithms in Table 1 10-fold cross validation is performed to test the accuracy and quality of the classification. k-fold cross validation is a technique in which the original sample is randomly partitioned into k subsamples. Of the k subsamples, a single subsample is retained as the validation data for testing the model, and the remaining k − 1 subsamples are used as training data.

The cross-validation process is then repeated k times (the folds), with each of the k subsamples used exactly once as the validation data. The k results from the folds are averaged to produce a single estimation. The advantage of this method over repeated random sub-sampling is that all observations are used for both training and validation, and each observation is used for validation exactly once.

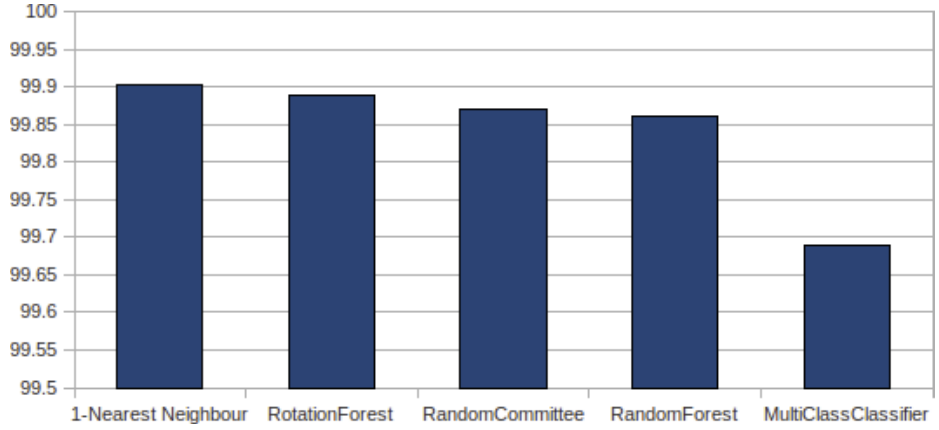

Figure 5. Performance of a selection of best performing machine learning algorithms by their performance and execution time

2. Results

The results from the experiments as shown in table 1 show state of the art performance in classification of the poses. The 1-Nearest Neighbour algorithm with euclidean distance shows the best performance with 99.9028% correct classification of the poses in approx. 165 seconds. The Random Tree algorithm takes some time in building a model but performs fast in 10-fold cross validation of approx.

10 seconds. It still has a high performance with over 98% correctly classified instances. The Voting Features Interval algorithm is fast in building a model, it only takes around .3 seconds, and in 10-fold cross validation which takes around 9 seconds. This however comes with a price: a lower performance of approx. 80 % correctly classified instances.

2.1. Experiments

The classification experiments performed with the twenty-six machine learning algorithms on the full dataset of 26000 poses from four test subjects ordered by their performance are shown in table 1.

Figure 6 shows the five algorithms that perform best using a 10% subset of the datset with respect to their execution time in 10-fold cross validation. These algorithms are the most suitable for online implementation because of their real-time behavior. Though a much smaller dataset is used for training the classifier, the performance of the algorithms, as shown in Table 2, does not drop below 77%.

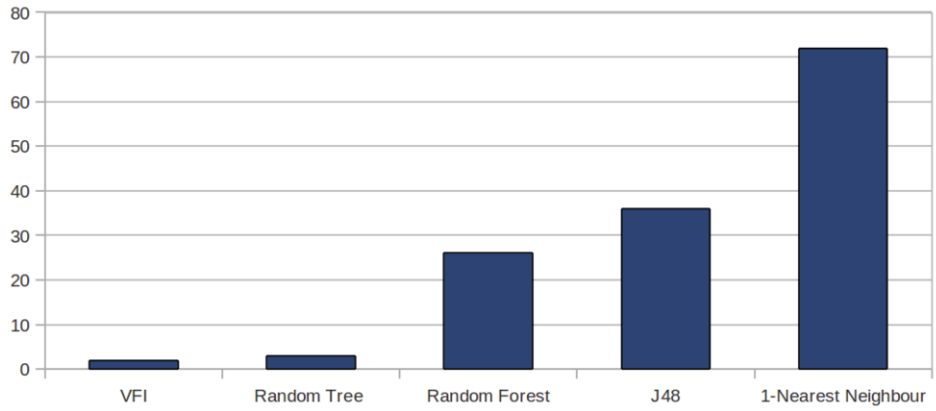

Figure 6. The five best performing algorithms ordered by their execution time

3. Discussion

A method for human pose and gesture recognition for human-robot interaction in autonomous robots is proposed and tested. The Voting Features Interval and Random Forest algorithm are the most suitable for online implementation. The Nearest Neighbour algorithm shows the best performance but this comes with a trade-off: it needs a considerable amount of time to perform matching of the pose which might make it less suitable for online implementation.

Name	Performance	Build time	Cross validation	Error
1-Nearest Neighbour	**99.9028**	**0.01**	**165**	**0.0104**
Rotation Forest	99.8878	281.54	2374	0.0144
Random Committee	99.8691	19.78	205	0.0156
Random Forest	**99.8616**	**15.69**	**159**	**0.0175**
Multi-Class Classifier	99.6896	581.15	3751	0.2275
Classification Using Regression	99.4765	151.1	1623	0.0248
NNge	99.2895	100.66	478	0.0281
Random Sub-Space	99.2858	53.9	557	0.0253
J48	**99.1587**	**21.92**	**195**	**0.0299**
PART	99.1474	107.96	959	0.0305
J48graft	99.0764	28.42	268	0.0315
Random Tree	**98.9680**	**2.05**	**19**	**0.0339**
JRIP Tree	98.8969	195.21	1560	0.0343
REP Tree	98.8140	10.09	86	0.0351
Data-Near-Balanced ND	98.7286	39.4	577	0.0373
SMO (SVM)	98.7212	16.92	172	0.2176
Nested Dichotomies	98.6987	62	1485	0.0376
Class-Balanced ND	98.6352	37.2	429	0.0386
Logit Boost	97.6816	321.45	3608	0.0525
Raced-Incremental Logit Boost	96.9338	49.06	945	0.0525
Bayes Net	93.5161	20.97	216	0.0825
Filtered Classifier	91.4968	20.41	222	0.0866
VFI	**79.4264**	**0.59**	**9**	**0.2070**
Decision Table	74.9654	63.99	814	0.1645
Hyper Pipes	65.1610	0.1	3	0.2260
One-R	35.5345	1.71	15	0.2676

Table 1. Performance comparison of the twenty-six classification algorithms. The columns from left to right show the name of the classification algorithm, the percentage correctly classified instances, the time it takes to build the model in seconds, the time it takes to perform 10-fold cross validation in seconds and the root mean squared error.

Name	Performance	Build time	Cross validation	Error
Voting Features Interval	77.3416	0.05	2	0.2189
Random Tree	94.5381	0.28	3	0.0802
Random Forest	98.8479	2.64	26	0.0433
J48	95.9462	4.18	36	0.0678
1-Nearest Neighbour	99.0186	0.02	72	0.0340

Table 2. The five best performing algorithms on 10% of the dataset ordered by their execution time in seconds. The columns from left to right show the name of the classification algorithm, the percentage correctly classified instances, the time it takes to build the model in seconds, the time it takes to perform 10-fold cross validation in seconds and the root mean squared error.

3.1. Future work

Future work consists of the integration of a system that performs online human pose recognition for human-robot interaction. The first step towards such a system requires the development of an interactive pose training behaviour for the autonomous robot. A possible follow-up will be to apply confidence scores in the classification of poses and create a behaviour that combines the confidence scores from both speech and gesture recognition systems to enhance the performance of the overall human-robot interaction system.

A first step towards benchmarking the performance of the Nearest Neighbour algorithm in online gesture recognition is to implement it using the Approximate Nearest Neighbors algorithm.

References

[1] van der Zant, T. and Wisspeintner, T. (2006). Robocup x: A proposal for a new league where robocup goes real world. In [Bredenfeld et al., 2006], pages 166-172.

[2] Wisspeintner, T., van der Zant, T., Iocchi, L., and Schiffer, S. (2009a). Robocup@home: Developing and benchmarking domestic service robots through scientific competitions.

[3] Wisspeintner, T., van der Zant, T., Iocchi, L., and Schiffer, S. (2009b). Robocup@home: Results in benchmarking domestic service robots. In Proceedings of the XIth RoboCup symposium.

[4] Brooks, R. (1986). "A robust layered control system for a mobile robot". Robotics and Automation, IEEE Journal of.

[5] Arkin, R. C. (1998). Behavior-Based Robotics. MIT Press.

[6] T. Moeslund, A. Hilton, and V. Kruger. A survey of advances in vision-based human motion capture and analysis. CVIU, 2006.

[7] R. Poppe. Vision-based human motion analysis: An overview. CVIU, 108, 2007.

[8] C. Plagemann, V. Ganapathi, D. Koller, and S. Thrun. Real-time identification and localization of body parts from depth images. In proceedings of the ICRA, 2010.

[9] A comparison of 3d model-based tracking approaches for human motion capture in uncontrolled environments. M. Shaheen, J. Gall, R. Strzodka, L. van Gool, and H.-P. Seidel. MPI Informatik, Germany. BIWI, ETH Zurich, Switerserland IBBT, ESAT-PSI, K.U.Leuven, Belgium.

[10] Real Time Motion Capture Using a Single Time-Of-Flight Camera. V. Ganapathi, C. Plagemann, D. Koller, S. Thrun. Stanford University, Computer Science Department, Stanford, CA, USA.

[11] Real-Time Human Pose Recognition in Parts from Single Depth Images, IEEE Conference on Computer Vision and Pattern Recognition (2008) (2011), Volume: 2, Issue: 3, Publisher: Ieee, Pages: 1297-1304, J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, A. Blake, Microsoft Research Cambridge and Xbox Incubation

[12] Fofi, D., Sliwa, T., Voisin, Y., A comparitive study on invisible structured light.

[13] Cover, T., Hart, P., Nearest neighbor pattern classification, Information Theory, IEEE Transactions on, vol.13, no.1, pp.21-27, January 1967

[14] Demiröz, Gülşen and Güvenir, H., Classification by Voting Feature Intervals, Machine Learning: ECML-97, vol. 1224, pp.85-92, 1997.

[15] Quinlan, J. R. C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, 1993.