AN ACTION-TUNED NEURAL NETWORK ARCHITECTURE FOR HAND POSE ESTIMATION

Giovanni Tessitore, Francesco Donnarumma and Roberto Prevete
Department of Physical Sciences, University of Naples Federico II, Naples, Italy

Keywords: Neural networks, Grasping action, Hand pose estimation, Mixture density networks.

Abstract: There is a growing interest in developing computational models of grasping action recognition. This interest is increasingly motivated by a wide range of applications in robotics, neuroscience, HCI, motion capture and other research areas. In many cases, a vision-based approach to grasping action recognition appears to be more promising. For example, in HCI and robotic applications, such an approach often allows for simpler and more natural interaction. However, a vision-based approach to grasping action recognition is a challenging problem due to the large number of hand self-occlusions which make the mapping from hand visual appearance to the hand pose an inverse ill-posed problem. The approach proposed here builds on the work of Santello and co-workers which demonstrate a reduction in hand variability within a given class of grasping actions. The proposed neural network architecture introduces specialized modules for each class of grasping actions and viewpoints, allowing for a more robust hand pose estimation. A quantitative analysis of the proposed architecture obtained by working on a synthetic data set is presented and discussed as a basis for further work.

1 INTRODUCTION

Over the last few years, there has been a keen interest in developing computational models for action recognition. Notably, grasping actions are of particular interest for various research areas including robotics, neuroscience, motion capture, telemanipulation, and human–computer interaction (HCI). Several works have been proposed in literature which address the problem of recognizing of grasping actions (Palm et al., 2009; Ju et al., 2008; Aleotti and Caselli, 2006). More specifically, as the direct use of the hand as input source is an attractive method, most of these works make use of wired gloves in order to perform input computation. Moreover, the only technology that currently satisfies the advanced requirements of hand-based input for HCI is glove-based sensing. For example, recognition can be based on wired glove kinematic information (Ju et al., 2008), or hybrid approaches in which glove information is put together with tactile sensors information (Palm et al., 2009; Keni et al., 2003; Aleotti and Caselli, 2006).

This technology has several drawbacks including the fact that it hinders the natural user interactions with the computer-controlled environment. Furthermore, it requires time-consuming calibration and setup procedures.

In contrast with this, vision-based approaches have the potential to provide more natural, non-contact solutions, allowing for simpler and more natural interactions between user and computer-controlled environment in HCI, as well as in robotics, where grasping actions recognition is mainly framed in the context of programming by demonstration (PbD).

There are approaches which make use of ad hoc solutions, like markers, in order to simplify this task (Chang et al., 2007). In general, as reported in (Weinland et al., 2010; Poppe, 2007), markless vision-based action recognition is acknowledged to be a challenging task. In particular, a major problem with grasping actions is the occlusion problem: hand-pose estimation from an acquired image can be extremely hard because of possible occlusions among fingers or between fingers and object being grasped.

For this reason, a body model approach seems more appropriate in this context. A body model approach usually consists of two steps: in a first step one estimates a 3D model of the human body (in the case of grasping actions this step coincides with the estimation of hand pose), in a second step recognition is made on the basis of joint trajectories.

Vision-based hand pose estimation is itself a chal-

Tessitore G., Donnarumma F. and Prevete R..
AN ACTION-TUNED NEURAL NETWORK ARCHITECTURE FOR HAND POSE ESTIMATION.
DOI: 10.5220/0003086403580363
In *Proceedings of the International Conference on Fuzzy Computation and 2nd International Conference on Neural Computation* (ICNC-2010), pages 358-363
ISBN: 978-989-8425-32-4

lenging problem (see (Erol et al., 2007) for a review). More specifically, addressing such a problem without any constraints on the hand pose makes the mapping from visual hand appearance to hand configuration very difficult to estimate. In this work, we present a neural network architecture composed of a series of specialized modules, each one implementing a mapping from visual hand appearance to hand configuration only when the hand belongs to a particular grasping action and is observed from a specific viewpoint.

The paper is organized as follows: in Section 2, we present the functional architecture and its actual implementation by means of Mixture Density Networks. In Section 3, we present the experimental set up in which the tests of Section 4 are carried out. Section 5 is devoted to conclusions and future work.

2 MODEL ARCHITECTURE AND IMPLEMENTATION

The main idea behind the present approach is to develop a set of specialized functional mappings tuned to predefined classes of grasping actions, and to use these specialized mappings to estimate hand poses of visually presented grasping actions. This approach is based on the assumption that grasping actions can be subdivided into different classes of grasping actions, and that coordinated movements of hand fingers result, during grasping actions, in a reduced number of physically possible hand shapes (see (Santello et al., 2002; Prevete et al., 2008)).

Moreover, view-independent recognition was obtained by developing view-dependent functional mappings, and by combining them in an appropriate way. Thus, the system is basically based on a set of specialized functional mappings, and a selection mechanism. The specialized mapping functions perform a mapping from visual image features to likely hand pose candidates represented in terms of joint angles. Each functional mapping is tuned to a predefined viewpoint and a predefined class of grasping actions. The selection mechanism selects an element of the set of candidate hand poses. In the next two subsections we will first provide an overall functional description of the system, followed by a detailed description of the system neural network structure.

2.1 Functional Model Architecture

The system proposed here can functionally be subdivided into three different modules: Feature Extraction (FE) module, Candidate Hand Configuration (CHC) module, and Hand Configuration Selection (HCS) module. The whole functional architecture of the system is shown in Figure 1. The functional roles of each module can be described as follows:

FE module. This module receives an $R \times S$ gray-level image as input. The output of the module is a $1 \times D$ vector $\mathbf{x}$. The FE module implements a PCA linear dimensional reduction.

CHC module. The CHC module receives the output $\mathbf{x}$ of FE as input. This module is composed of a bank of $N \times M$ sub-modules CHC_{ij}, with $i = 1, 2, ..., N$ and $j = 1, 2, ..., M$. N is the number of predefined classes of grasping actions, and M is the number of predefined viewpoints. Given the input $\mathbf{x}$, each sub-module CHC_{ij} provides the most likely hand configuration, in terms of a $1 \times N_{dof}$ vector $\mathbf{t}^{ij}$. Moreover, each $\mathbf{t}^{ij}$ is associated with an estimation error err^{ij}, assuming that the input $\mathbf{x}$ is obtained during the observation of a grasping action belonging to the i-th class from the j-th viewpoint. Thus each sub-module CHC_{ij}, for the i-th grasping action and j-th viewpoint, performs a specialized mapping from visual image features to hand poses. The basic functional unit of each sub-module CHC_{ij} is an inverse-forward model pair. The inverse model extracts the most likely hand configuration $\mathbf{t}^{ij}$, given $\mathbf{x}$, while the forward model gives as output an image feature vector $\mathbf{x}^{ij}$, given $\mathbf{t}^{ij}$. The error err^{ij} is computed on the basis of $\mathbf{x}$ and $\mathbf{x}^{ij}$.

HCS module. The HCS module receives the output of CHC as input. It extracts a hand pose estimation on the basis of estimation errors err^{ij}, by selecting the hand pose associated with the minimum error value.

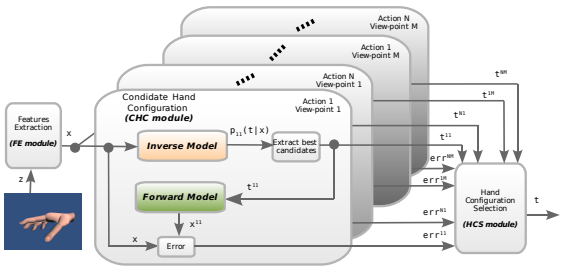


Figure 1: The system is functionally composed of three different modules: Feature Extraction (FE) module, Candidate Hand Configuration (CHC) module, and Hand Configuration Selection (HCS) module.

2.2 Neural Network Implementation

The FE module implements a PCA linear dimensional reduction by means of an autoassociative neural network (Bishop, 1995). This is a multilayer perceptron composed of $R \times S$ input nodes, D hidden nodes, and $R \times S$ output nodes. Both hidden nodes and

output nodes have linear activation and output function. The input vectors are obtained by linearizing the gray-level input images of size $R \times S$ into single vectors of size $1 \times R \cdot S$. The network is trained to associate input vectors with themselves by a standard back-propagation algorithm. Once trained, the vectors $\mathbf{x}$ are obtained as the output of the hidden units.

As the *CHC* module is solely composed by CHC_{ij} modules, we focus now on a generic CHC_{ij} module, and describe how it can be developed by means of a neural network. As described above, a CHC_{ij} module is composed of an inverse-forward model pair. Let us firstly consider the inverse-model. This provides a hand configuration $\mathbf{t}$, given a visual image feature vector $\mathbf{x}$. A major mathematical problem arising in this context concerns the ill-posed character of the required transformation from $\mathbf{x}$ to hand configurations insofar as the same visually presented hand can be associated with various hand configurations. Therefore the mapping from $\mathbf{x}$ to hand configurations $\mathbf{t}$ is not a functional mapping, and assumes the form of an inverse ill-posed problem (Friston, 2005; Kilner et al., 2007). According to (Bishop, 1995), one can cope with this problem by estimating $p(\mathbf{t}|\mathbf{x})$ in terms of a Mixture Density Network (MDN) approach: $p(\mathbf{t}|\mathbf{x}) = \sum_{i=1}^{K} c_i(\mathbf{x})\phi_i(\mathbf{t}|\mathbf{x})$. The $\phi_i(\mathbf{x})$ are kernel functions, which are usually Gaussian functions of the form: $\phi_i(\mathbf{t}|\mathbf{x}) = \frac{1}{(2\pi)^{D/2}\sigma_i^D(\mathbf{x})} \exp\left\{-\frac{\|\mathbf{t}-\mu_i(\mathbf{x})\|^2}{2\sigma_i^2(\mathbf{x})}\right\}$. The parameters $c_i(\mathbf{x})$ can be regarded as prior probabilities of $\mathbf{t}$ to be generated from the i-th component of the mixture. The coefficients of the mixture, $c_i(\mathbf{x})$, and the parameters of the kernel functions, $\phi_i(\mathbf{t},\mathbf{x})$ ($\mu_i(\mathbf{x})$ and $\sigma_i(\mathbf{x})$ for a Gaussian kernel), depend on the sensory inputs $\mathbf{x}$. A two-layer, feed-forward neural network can be used to model the relationship between visual inputs $\mathbf{x}$ and corresponding mixture parameters $c_i(\mathbf{x})$, $\mu_i(\mathbf{x})$ and $\sigma_i(\mathbf{x})$. Accordingly, the problem of estimating the conditional probability distribution $p(\mathbf{t}|\mathbf{x})$ can be approached in terms of neural networks by combining a multi-layer perceptron and a Radial Basis Function (RBF) like network. The RBF network will be composed of N_{dof} input, K hidden nodes and one output node. The form of basis function is the same as the Gaussian functions expressed above, with the Gaussian parameters of the first layer set to $\mu_i(\mathbf{x})$ and $\sigma_i(\mathbf{x})$, and the second layer weights set to $c_i(\mathbf{x})$. Thus, given a previously unseen hand visual description $\mathbf{x}$, one can obtain an estimate of hand configuration $\mathbf{t}$ as the central value of the more probable branch of $p(\mathbf{t}|\mathbf{x})$.

The network was trained using a dataset composed of visual feature vectors $\mathbf{x}^n$ and hand configurations $\mathbf{t}^n$ collected during the observation of grasping actions belonging to the C_i-th class from the j-th viewpoint, and by using $E = -\sum_n \{\sum_j c_j(\mathbf{x}^n)\phi(\mathbf{t}^n|\mathbf{x}^n)\}$ as error function. Once trained, the network output, i.e., the candidate hand configuration $\mathbf{t}^{ij}$, is obtained as the vector $\mu_h(\mathbf{x})$ associated to the highest $c_h(\mathbf{x})$ value. This operation is achieved by the module *Extract best candidate*.

The design of the forward-model, involves a multi-layer perceptron composed of N_{dof} input and D output units. The neural network receives a hand configuration $\mathbf{t}$ as input, and computes an expected image feature vector $\mathbf{x}^{ij}$ as output. The vector $\mathbf{x}^{ij}$ is the expected image feature vector corresponding to the hand configuration $\mathbf{t}$ when the hand is observed during a grasping action belonging to the C_i-th class from the j-th viewpoint. The network was trained using the *RProp* learning algorithm by exploiting again a dataset composed of hand configurations and visual feature vectors, collected during the observation of grasping actions belonging to the C_i-th class from the j-th viewpoint.

The error err^{ij} associated with the candidate hand configuration $\mathbf{t}^{ij}$ is computed as the sum-of-square error between the vectors $\mathbf{x}$ and $\mathbf{x}^{ij}$. Finally, given the set of candidate hand configurations $\mathbf{x}^{ij}$ and associated errors err^{ij}, with $i = 1,2,..,N$ and $j = 1,2,...,M$, the candidate hand configuration with the lowest associated error is identified as the output of the whole system.

3 EXPERIMENTAL SETTING

We performed two main types of experiments to control the performance of the proposed architecture. In the first type of experiments we used different action classes (DA-TEST); and in the second type we used different viewpoints (DV-TEST).

A major problem in testing the performance of a hand pose estimation system is to obtain a ground truth. In fact, in the case of a quantitative analysis, one needs to know the actual hand configuration corresponding to the hand picture which is fed as input to the system. This is difficult to achieve with real data. For this reason, we decided to work with a synthetic dataset constructed by means of a dataglove and a 3D rendering software. The dataglove used for these experiments is the HumanGlove (*HumanGlove, Humanware S.r.l., Pontedera (Pisa), Italy*) endowed with 16 sensors. This dataglove feeds data into the 3D rendering software which reads sensor values and constantly updates a 3D human hand model. Thus, this experimental setting enables us to collect *hand joints configuration - hand image* pairs.

In the DA-TEST two different types of grasps

were used in accordance with the treatment in (Napier, 1956): *precision-grasp* (PG) and *power-grasp*, the latter being also known as whole hand grasp (WH). In performing a power grasp, the object is held in a clamp formed by fingers and palm; in performing a precision grasp, the object is pinched between the flexor aspects of the fingers and the opposing thumb. Two different objects were used: a tennis ball was used for the WH actions, and a pen cup for the PG actions. We collected 20 actions for each class of actions. In the DV-TEST we rendered the 3D hand model of the PG grasping actions data, from 9 different viewpoints as reported in Table 2. In both DA-TEST and DV-TEST, the capability of the proposed architecture in recovering a hand pose was measured in terms of Euclidean distance between the actual hand pose $\mathbf{t}$ and the estimated hand pose $\hat{\mathbf{t}}$, that is $E^{NORM} = \| \mathbf{t} - \hat{\mathbf{t}} \|$. This measure was reported in other works such as (Romero et al., 2009). However, due to the differences in the experimental settings it is difficult to make a clear comparison between results. For this reason, we decided to compute a further term, that is, the Root-Mean-Square (RMS) error, in order to obtain a more complete interpretation of our results. The RMS error over a set of actual hand poses $\mathbf{t}^n$ and estimated hand poses $\hat{\mathbf{t}}^n$ is computed as: $E^{RMS} = \frac{\sum_{n=1}^{N} \|\hat{\mathbf{t}}^n - \mathbf{t}^n\|^2}{\sum_{i=1}^{N} \|\mathbf{t}^n - \bar{\mathbf{t}}\|^2}$. Here $\bar{\mathbf{t}}$ is defined as the average of actual hand pose vectors, that is: $\bar{\mathbf{t}} = \frac{1}{N} \sum_{n=1}^{N} \mathbf{t}^n$. In this way the RMS error approaches 1 when the model predicts the mean value of the test data, and approaches to 0 value when the model's prediction captures the actual hand poses. Thus, we expect to have a good performance when the RMS error for our system is very close to zero.

Moreover, we have computed the "selection error" (E^{SEL}) which measures how many times the input image belonging to the i-th action class and j-th viewpoint does not result in the lowest error for the module CHC_{ij}. The E^{SEL} error is expressed as percentage of frames (belonging to the action class i and the viewpoint j) which do not give rise the lowest error for the module CHC_{ij}.

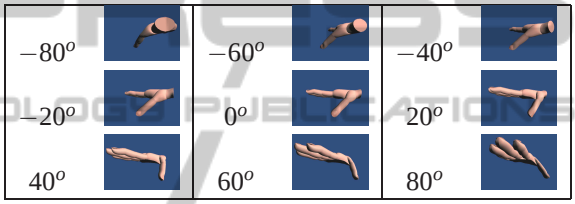
The architecture of the model used for these tests is composed of a number of CHC_{ij} modules depending on the number of different action classes and viewpoints. For each CHC_{ij} module the MDN component, implementing the inverse model, was trained using different values of H hidden units and K kernels. The feedforward neural network (FNN), implementing the forward model, is instead trained with different values of the hidden unit number L. For both MDN and FNN only the configuration was considered which gives rise to the highest likelihood, for the MDN, and to the lowest error for the FNN, computed on a validation set.

Table 1 summarizes the parameters used for both DA-TEST and DV-TEST.

Table 1: Parameters used for both DA-TEST and DV-TEST.

	DA-TEST	DV-TEST
viewpoints (N)	1	9
Action class (M)	2	1
Hidden Nodes (H)	From 5 to 10 at step 2	
Kernels (K)	From 3 to 10 at step 1	
Hidden nodes (L)	From 5 to 10 at step 2	
Num inputs (D)	30	
Input Image dim (R,S)	135×98	

Table 2: Sample input images from the 9 different viewpoints used for DV-TEST.



4 DA-TEST RESULTS

In this test we used 20 PG grasping actions and 20 WH grasping actions. The architecture consists of two CHC modules. Each of these modules was trained on 5 of the corresponding actions (PG or WH) whereas 5 actions have been used for validation, and the remaining 10 actions for testing.

In Table 3 the mean and the standard deviation of the E^{NORM} error over all hand poses contained in the 10 test actions is reported. Moreover the RMS Error is reported there, insofar as it provides more meaningful error information. Finally, we reported the error of selection E^{SEL}.

As one can see from the E^{SEL} error, the system is almost always able to retrieve the correct module CHC for processing the current input image. Moreover, the system gives a good hand pose estimation since the RMS Error is close to zero for both PG actions and WH actions. The E^{NORM} error is also reported for comparison with other works. In Table 4 sample input images together with their hand estimations [1] and corresponding E^{NORM} errors are reported. One can see that if the E^{NORM} error is close to mean

[1]The picture of the estimated hand configuration is obtained by means of the 3D simulator.

error (reported in Table 3), then the input and estimated hand images are very similar.

Table 3: RMS error and the Mean and standard deviation of the E^{NORM} error for test actions belonging to class PG and WH. Moreover the E^{SEL} error is reported.

	PG	WH
RMS	0.24	0.12
E^{NORM} $(\mu \pm \sigma)$	20.1 ± 27	11.7 ± 19
E^{SEL}	3%	9%

Table 4: Sample input images together with the corresponding hand estimated images drawn from both PG and WH test actions.

	Actual	Estimated	Error
PG			118.3
PG			19.7
WH			74.9
WH			12.3

5 DV-TEST RESULTS

The DV-TEST is divided into two phases. In a first phase, we test the system on the same viewpoint which it was trained. Note that although test viewpoints are the same as training viewpoints, when an image frame of a test action is fed as input, the system does not "know" what is the viewpoint corresponding to that action, and must retrieve such information from the input. In a second phase, we test the system on viewpoints that are different with respect to the ones it has been trained with.

In the first phase, we used 20 PG grasping actions rendered from 9 different viewpoints. The model architecture consists of 9 CHC_{ij} modules, one for each viewpoint, with $i = 1$ and $j = 1, \ldots, 9$. The forward and inverse models of the CHC_{ij} module were trained on data related to the j-th viewpoint only. In particular, 5 actions were used for training and 5 other actions for validation. Once trained, the remaining 10 actions for each viewpoint were fed as input to the system.

Table 5 shows the selection error E^{SEL} together with the two estimation errors, E^{NORM} and RMS for DV-TEST. One can see that the system is able to retrieve the right viewpoint for the hand input image (E^{SEL} almost 0 for all viewpoints) and is able to give a reasonable hand pos estimation as confirmed by the RMS error which is close enough to zero.

Table 5: Selection error E^{SEL}, estimation error E^{NORM}, and RMS error for test viewpoints used in DV-TEST first phase.

viewpoint	0^o	20^o	40^o
RMSError	0.02	0.04	0.2
E^{NORM} $(\mu \pm \sigma)$	9.1 ± 9	12.4 ± 15	25.2 ± 38
E^{SEL}	0%	0%	1%
viewpoint	60^o	80^o	-20^o
RMSError	0.08	0.13	0.06
E^{NORM} $(\mu \pm \sigma)$	15.9 ± 24	21.2 ± 30	14.9 ± 21
E^{SEL}	0%	1%	0%
viewpoint	-40^o	-60^o	-80^o
RMSError	0.19	0.08	0.03
E^{NORM} $(\mu \pm \sigma)$	23.6 ± 37	16.5 ± 23	12.1 ± 14
E^{SEL}	0%	0%	0%

In the second phase of the DV-TEST we used five CHC modules only, corresponding to the viewpoints at 0, 40, 80, -40, and -80 degrees. The system was tested on the remaining viewpoints at 20, 60, -20, and -60 degrees.

Table 6 reports errors in recovering hand pose from viewpoints that the system was not trained on: only for viewpoint corresponding to -20 degrees the error is acceptable, in the other cases, the error is high.

Table 6: Selection error E^{SEL} together with estimation error E^{NORM} and RMS error for all test viewpoints used in DV-TEST second phase.

viewpoint	20^o	60^o
RMSError	1.69	2.22
E^{NORM} $(\mu \pm \sigma)$	119 ± 58	137.5 ± 65
viewpoint	-20^o	-60^o
RMSError	0.42	1.47
E^{NORM} $(\mu \pm \sigma)$	59.8 ± 28	114.5 ± 46

6 CONCLUSIONS

The neural architecture described in this paper was deployed to address the problem of vision-based hand pose estimation during the execution of grasping actions observed from different viewpoints. As stated in the introduction, vision-based hand pose estimation is, in general, a challenging problem due to the large amount of self-occlusions between fingers which make this problem an inverse ill-posed problem. Even though the number of degrees of freedom is quite large, it has been showed (Santello et al.,

2002) that a hand, during grasping action, can effectively assume a reduced number of hand shapes. For this reason it is reasonable to conjecture that vision-based hand pose estimation becomes simpler if one "knows" which kind of action is going to be executed. This is the main rationale behind our system. The results of the DA-TEST show that this system is able to give a good estimation of hand pose in the case of different grasping actions.

In the first phase of the DV-TEST comparable results with respect to the DA-TEST have been obtained. It must be emphasized, moreover, that although the system has been tested on the same viewpoints it was trained on, the system does not know in advance which viewpoint a frame drawn from a test action belongs to.

In the second phase of the DV-TEST an acceptable error was obtained from one viewpoint only. This negative outcome is likely to depend on excessively high differences in degrees between two consecutive training viewpoints. Thus a more precise investigation must be performed with a more comprehensive set of viewpoints. A linear combination of the outputs of the CHC modules, on the basis of the produced errors, can be investigated too. Furthermore, the FE module can be replaced with more sophisticated modules, in order to extract more significant features such as Histograms of Oriented Gradients (HOGs) (Dalal and Triggs, 2005). A comparison with other approaches must be performed. In this regard, however, the lack of some benchmark datasets make meaningful comparisons between different systems difficult to produce. Finally, an extension of this model might profitably take into account graspable object properties (Prevete et al., 2010) in addition to hand visual features.

ACKNOWLEDGEMENTS

This work was partly supported by the project Dexmart (contract n. ICT-216293) funded by the EC under the VII Framework Programme, from Italian Ministry of University (MIUR), grant n. 2007MNH7K2 003, and from the project *Action Representations and their Impairment* (2010-2012) funded by Fondazione San Paolo (Torino) under the Neuroscience Programme.

REFERENCES

Aleotti, J. and Caselli, S. (2006). Grasp recognition in virtual reality for robot pregrasp planning by demonstration. In *ICRA 2006*, pages 2801–2806.

Bishop, C. M. (1995). *Neural Networks for Pattern Recognition*. Oxford University Press.

Chang, L. Y., Pollard, N., Mitchell, T., and Xing, E. P. (2007). Feature selection for grasp recognition from optical markers. In *IROS 2007*, pages 2944 – 2950.

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *CVPR'05 - Volume 1*, pages 886–893, Washington, DC, USA. IEEE Computer Society.

Erol, A., Bebis, G., Nicolescu, M., Boyle, R. D., and Twombly, X. (2007). Vision-based hand pose estimation: A review. *Computer Vision and Image Understanding*, 108(1-2):52–73.

Friston, K. (2005). A theory of cortical responses. *Philos Trans R Soc Lond B Biol Sci*, 360(1456):815–836.

Ju, Z., Liu, H., Zhu, X., and Xiong, Y. (2008). Dynamic grasp recognition using time clustering, gaussian mixture models and hidden markov models. In *ICIRA '08*, pages 669–678, Berlin, Heidelberg. Springer-Verlag.

Keni, B., Koichi, O., Katsushi, I., and Ruediger, D. (2003). A hidden markov model based sensor fusion approach for recognizing continuous human grasping sequences. In *Third IEEE Int. Conf. on Humanoid Robots*.

Kilner, J., James, Friston, K., Karl, Frith, C., and Chris (2007). Predictive coding: an account of the mirror neuron system. *Cognitive Processing*, 8(3):159–166.

Napier, J. R. (1956). The prehensile movements of the human hand. *The Journal of Bone and Joint Surgery*, 38B:902–913.

Palm, R., Iliev, B., and Kadmiry, B. (2009). Recognition of human grasps by time-clustering and fuzzy modeling. *Robot. Auton. Syst.*, 57(5):484–495.

Poppe, R. (2007). Vision-based human motion analysis: An overview. *Computer Vision and Image Understanding*, 108(1-2):4 – 18. Special Issue on Vision for Human-Computer Interaction.

Prevete, R., Tessitore, G., Catanzariti, E., and Tamburrini, G. (2010). Perceiving affordances: a computational investigation of grasping affordances. *Accepted for publication in Cognitive System Research*.

Prevete, R., Tessitore, G., Santoro, M., and Catanzariti, E. (2008). A connectionist architecture for view-independent grip-aperture computation. *Brain Research*, 1225:133–145.

Romero, J., Kjellstrom, H., and Kragic, D. (2009). Monocular real-time 3d articulated hand pose estimation . In *IEEE-RAS International Conference on Humanoid Robots (Humanoids09)*.

Santello, M., Flanders, M., and Soechting, J. F. (2002). Patterns of hand motion during grasping and the influence of sensory guidance. *Journal of Neuroscience*, 22(4):1426–1235.

Weinland, D., Ronfard, R., and Boyer, E. (2010). A Survey of Vision-Based Methods for Action Representation, Segmentation and Recognition. Technical report, INRIA.