Delay Reduction in Real-Time Recognition of Human Activity for Stroke Rehabilitation

Nabiei, Roozbeh; Najafian, Maryam; Parekh, Manish; Jancovic, Peter; Russell, Martin

DOI:
10.1109/SPLIM.2016.7528413

Delay reduction in real-time recognition of human activity for stroke rehabilitation

Roozbeh Nabiei[1], Maryam Najafian[2], Manish Parekh[1], Peter Jančovič[1], Martin Russell[1]
[1]School of Electrical and Systems Engineering, University of Birmingham, Birmingham, B15 2TT, UK
[2]Center for Robust Speech Systems (CRSS), University of Texas at Dallas, Richardson, TX, USA
Email: [rxn946, m.parekh, p.jancovic, m.j.russell]@bham.ac.uk[1], and [m.najafian]@utdallas.edu[2]

Abstract—**Assisting patients to perform activities of daily living (ADLs) is a challenging task for both human and machine. Hence, developing a computer-based rehabilitation system to re-train patients to carry out daily activities is an essential step towards facilitating rehabilitation of stroke patients with apraxia and action disorganization syndrome (AADS). This paper presents a real-time hidden Markov model (HMM) based human activity recognizer, and proposes a technique to reduce the time-delay occurred during the decoding stage. Results are reported for complete tea-making trials. In this study, the input features are recorded using sensors attached to the objects involved in the tea-making task, plus hand coordinate data captured using Kinect$^{\text{TM}}$ sensor. A coaster of sensors, comprising an accelerometer and three force-sensitive resistors, are packaged in a unit which can be easily attached to the base of an object. A parallel asynchronous set of detectors, each responsible for the detection of one sub-goal in the tea-making task, are used to address challenges arising from overlaps between human actions. The proposed activity recognition system with the modified HMM topology provides a practical solution to the action recognition problem and reduces the time-delay by 64% with no loss in accuracy.**

I. Introduction

Apraxia and Action Disorganization Syndrome (AADS) is a broad term that describes a compromised ability to use objects and gestures in a goal-directed manner in a naturalistic setting. Most often, AADS is caused by damage to one of the brain hemispheres caused by CardioVascular Accident (CVA). A large number of stroke survivors suffer from apraxia which leads to an impairment of cognitive abilities to complete activities of daily living (ADLs) [1–3]. These patients often perform an incorrect sequence of actions, skip steps, or misuse objects with possible safety implications.

Assisting patients with their routine activities is a challenging task for both human and machine. Hence, the objective of the CogWatch project [4–6] is to develop an intelligent computer-based rehabilitation system to (1) monitor the patient's progress through the ADL and (2) provide appropriate guiding cues for the patient when an error is detected or anticipated and re-train patients to carry out daily activities. Designing such system is an essential step towards facilitating rehabilitation of stroke patients suffering from AADS.

Although the objective of CogWatch is the wider development of technology for cognitive rehabilitation of stroke patients, the focus of the present paper is reducing the delay occurred during the decoding stage of the proposed action recognition system, based on HMMs and instrumented objects. The paper is organized as follows. Section IV-B describes the task. Section III describes our approach to instrumentation of objects. Section IV-A describes how features are extracted from the sensors. Sections IV-C and IV-B describe the conventional action recognition system, and section IV-D presents a real-time action recognition system with reduced time-delay. Section VI presents the experimental results and analysis. Section VII presents our conclusions.

II. Related work

ADL can be captured through sensors on the patients or their environment, or the objects that they interact with [7–14], but decomposition of an ADL into sub-goals and recognition of these sub-goals has received less attention. The use of sensorised objects promotes an "object-centric" view of action recognition, in which a sub-goal is characterized in terms of how it is "experienced" by the objects involved. This contrasts with "scene-oriented" approaches, in which an external video sensor plus image processing is used to identify and track the hands and objects during a task, or approaches where sensors are attached to the body (for example [13,15]). The object-centred and scene-oriented approaches are both unobtrusive, since neither requires the user to wear sensors. However, the scene-oriented approach normally requires careful installation and calibration of cameras, which may be an issue if the system is intended to be widely deployed and stand-alone, for example in an ordinary household kitchen. A popular option for instrumentation is to use Radio Frequency Identification (RFID) tags to identify which objects have been picked up [16,17], however these do not provide sufficiently rich information and an antenna bracelet needs to be worn.

III. Instrumentation and Sensors

The initial ADL in CogWatch is "making a cup of tea". These sub-goals are recognized from the outputs of sensors attached to the objects involved, and the location of the hands. The objects involved in the tea-making task are a kettle, water, jug, mug, milk jug, spoon and containers for the tea-bags, sugar and used tea-bags. In the current system only the kettle, mug and milk jug are instrumented. The sensors and circuitry are packed into an instrumented 'coaster', the 'CogWatch Instrumented Coaster (CIC)', that is fitted to the underside of the object (figure 1). The CIC contains a 3-axis accelerometer, 3 force sensitive resistors (FSRs), a PIC, a Bluetooth and a battery. For the kettle, which is 'cordless' with a separate base, the CIC was split into two packages, with the accelerometer attached to the kettle body and the FSRs attached to the base. The accelerometer is an Analog Devices ADXL335, providing acceleration measurements on 3 axes in a range of ±3g.

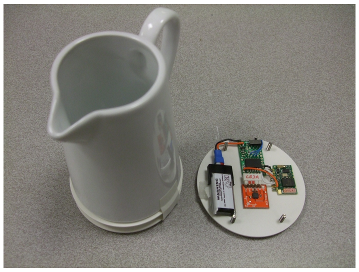

Fig. 1. *A jug fitted with a CogWatch Instrumented Coaster (CIC) and an 'open' CIC, showing the accelerometer, PIC, Bluetooth module and battery*

Its function is to respond to changes in motion, tilting, and disturbances of the object due to the addition of materials, stirring, collisions or (in the kettle) vibration during boiling. The FSRs can detect whether the object is standing on a surface of lifted in the air, changes in weight due to the addition or removal of materials, and more subtle changes in weight distribution across the base of the object (making it possible, for example, to detect stirring). The output of an individual CIC at any time is a six dimensional vector, comprising x, y, z accelerometer outputs plus the outputs of the three FSRs. The data from the FSRs attached to the mug show the increase in weight of the mug as it is filled. The data from the kettle FSRs identifies the points where the kettle is lifted from and then returned to the table. In addition to outputs from CICs, the system uses hand-coordinate data captured using Kinect [18], using software based on the 'Kinect-Arms' libraries [19].

IV. Proposed system

In this section we describe an HMM based action recognition system. Furthermore, we present an approach that reduces the delay occurred during the conventional decoding approach presented in our previous work [20].

A. Feature Extraction

The raw data (comprising hand coordinates from Kinect, and FSR and accelerometer data from the three CICs) are streamed to the system and synchronized at $50\,\text{Hz}$. Each sub-goal is characterized by a different combination of raw sensor data and features extracted from the raw sensor data. For example, detection of the sub-goal "Pour Kettle" uses the outputs from the kettle CIC, the FSRs in the CIC attached to the mug, and hand position. Hand position is given relative to x and y axes parallel to edges of the table and centered at the center of the table. A 2D "Gaussian neighborhood" associated with each object, is used to indicate when the hand is in the vicinity of that object. The mean and covariances of the Gaussian neighborhood for an object is calculated using the location of the hand when it is stationary and interacting with that object. The hand is assumed to be stationary if the difference between successive samples is less than 3mm. The distance that the hand has traveled between times t and $t+1$ is the Euclidean distance:

$$d(h_t, h_{t+1}) = \sqrt{(h_{1,t+1} - h_{1,t})^2 + (h_{2,t+1} - h_{2,t})^2}.$$

Here $h_t = (h_{1,t}, h_{2,t})$ is the position of the hand at time t.

A number of features are extracted from the raw data for AR. For example, to calculate the change in weight of the mug a low pass filter is used to smooth the data from FSRs in the CIC attached to the mug, before the derivative is calculated. Also, the FSR data obtained from the FSRs under the kettle and in the CIC attached to the milk jug is used to determine whether or not that object has been picked up. Variance in the energy of the outputs from the accelerometer attached to the kettle body, caused by vibration of the kettle during the process of heating the water, is used to determine whether the water in the kettle had reached boiling point and hence detect the sub-goal "Boil Water". The feature vector y_t at time t is calculated from a window comprising sensor outputs at times $t-20, ..., t$ and passed to the recognizer.

B. HMM-Based action recognition

Variations in the sequences of sensor outputs that result from individual differences in the ways that users execute the task, variations in the way that the same user executes the same task on different occasions, or sensor noise are captured using a statistical model (a sub-goal HMM (for example, [21])). The partially-ordered structure of the sub-goal lattice, in which sub-goals occur in overlapping time, or even at the same time, is accommodated using a parallel set of asynchronous HMM-based detectors, each responsible for detecting a specific sub-goal. The proposed system provides assistance for four types of tea-making "black tea", "black tea with sugar", "tea with milk" and "tea with milk and sugar". Using task analysis [22], each variant is decomposed into a hierarchy of sub-goals. The list of tea-making sub-goals can be summarized as following. Here, 7 sub-goals, a common error (9), and a potential hazard (10) are identified.

1) "Fill kettle" (using water from a pre-filled jug)
2) "Pour kettle" (i.e. pour boiling water into the mug)
3) "Add tea-bag"
4) "Add sugar"
5) "Add milk"
6) "Remove tea-bag"
7) "Stir"
8) "Toy milk" (pour milk outside the mug)
9) "Toy kettle" (pour boiled water outside the mug)

It is not a prescription for a linear sequence. The execution of sub-goals may overlap, so that one sub-goal begins before another is complete (for example, if both hands are used "Add tea-bag" could start during "Pour kettle"). Even when the sub-goals do occur in sequence the order may vary.

HMMs are a generic framework for statistical sequential pattern processing, but they have received most attention in the area of automatic speech recognition (ASR) (for example, see [23–26]). The key process in a typical HMM-based ASR system is a Viterbi decoder [23]. Given a sequence of feature vectors $y = y_1, ..., y_T$ the Viterbi decoder finds the sequence of HMMs $M = M_1, ..., M_N$ such that an approximation to the probability $p(M|y)$ is maximized. Since y is fixed, from Bayes' rule this is equivalent to finding M such that $p(y|M)P(M)$ is maximised. The probability $P(M)$ is based on a language model which defines the probability of any given sequence of words. In speech recognition, the language model and the individual HMMs are compiled into a single network and the most probable path through this network is found using Viterbi decoding. However, in ASR words occur one-after-another, whereas in AR actions can occur in overlapping time, so that the natural structure is a partially-ordered lattice rather than a sequence. Overlap may occur, for example, if the subject uses both hands, or executes one or more sub-goals while the kettle is boiling. Therefore a conventional ASR decoder, which will compute the most probable sequence of actions given the data, is not appropriate for AR.

Our AR system consists of 5 independent real-time HMM-based detectors which together are capable of identifying occurrences of the 7 sub-goals of tea-making at any time during completion of the tea-making task. These detectors run in parallel and are completely separate from each other. Each detector takes as input those parts of the feature vector that are useful for detecting its sub-goal(s). A detector consists of one or more multiple state HMMs, each representing a unique sub-goal, and these HMM states are associated with Gaussian mixture models (GMMs). In addition, the detector includes a single state "background" (or "toying") HMM, whose state is associated with a multiple-component GMM. The five detectors are as follows:

The "Front actions" detector consists of three "sub-goal" models (corresponding to "Add sugar", "Add tea-bag" and "Remove tea-bag") and a background "toying" model. This detector is primarily influenced by the Gaussian neighborhood features for the mug, tea-bag container, sugar container and used tea-bag container, which are calculated from Kinect, and the outputs of the FSRs in the CICs under the mug.

The "Pour kettle' and "Add milk" detectors each consist of a single sub-goal model (for "Pour water" or "Add milk") and a "toying" model which corresponds to picking up the kettle or milk jug but not pouring water or milk into the mug. These detectors exploit the accelerometer and FSR outputs of the CICs attached to the kettle or milk jug, to indicate that this object has been picked up, moved, tilted, moved and put down, and the synchronized FSRs in the CIC attached to the mug to detect that at the time that the first object is tilted the mug begins to get heavier.

The "Fill kettle" detector has a single sub-goal HMM for "Fill kettle" and a "toying" model. The inputs to this detector are Gaussian neighborhood values associated with the jug and kettle and the outputs of the CIC under the kettle to detect movement and an increase in weight.

The "Stir" detector has a single sub-goal HMM for "Stir" and a "toying" model. The inputs to this detector are Gaussian neighborhood values associated with the mug and the outputs of the CIC under the mug to detect movement.

C. Real-time Viterbi decoder

An identical implementation of the Viterbi algorithm (for example see [23]) runs independently in each decoder. Briefly, each detector works as follows: At each time t the detector receives a new feature vector, y_t. For each state i of each of its HMMs, a quantity $\alpha_t(i)$ is calculated which can be thought of as an approximation to the probability of the best explanation of data $y_1, ..., y_t$ up to and including y_t ending in state i at time t. Intuitively, if the detector is for "Add milk" and the i^{th} state corresponds to tipping the jug, then $\alpha_t(i)$ can be thought of as the probability of the best explanation of data up to time t culminating in the tipping action at t. Formally $\alpha_t(i)$ is given by the recursion:

$$\alpha_t(i) = max_j \alpha_{t-1}(j) a_{j,i} b_i(y_t) \quad (1)$$

$$\rho_t(i) = argmax_j \rho_{t-1}(j) a_{j,i} b_i(y_t) \quad (2)$$

where $a_{j,i}$ is the probability of a transition from state j to state i and $b_i(y_t)$ is the probability of the sensor data y_t given state i. Note that the 'preceding' state j can be in the same HMM as state i, or, if i is an initial state, j can be the final state of another HMM in the detector. $\rho_t(i)$ provides a record from which the best explanation of the data up to time t in state i can be recovered.

In the "conventional" implementation of Viterbi decoding described above, the best explanation of the data is not recovered until the final time T. However, in a "real-time" implementation there is no final time. The memory required to store the $\rho_t(i)$s and $\alpha_t(i)$s will increase and no output will be produced. The solution is to use a technique called 'partial traceback' [27]. Each detector's output up to a time s is generated as soon as its classification of the data up to that point is unambiguous, in the sense that all of the $\rho_t(i)$s can be traced-back to a common state at time s in the past. The memory used to store alternative explanations of the data up to s is then freed. In this way the decoders can run indefinitely. If the convergence point s is significantly less than t then there will be a delay in the output of the decoder. Therefore, care is needed in the construction of the HMMs to avoid the ambiguity that will cause this to happen. Whenever a sub-goal HMM provides the most probably explanation of a section of input, a label indicating that sub-goal is output. Otherwise the best explanation of the data is "toying" and nothing is output.

D. Modified HMMs for real-time Viterbi time-delay reduction

The main source of the delay in real-time Viterbi decoding for our AR task is the inevitable similarity between the end states of a sub-goal model and "toying" action caused during the iteration of embedded training where parameters of HMMs are optimized based on the alignment of the training data with the model's states. The modification is achieved by deleting the self-loop transition for states similar to background data (toying), to prohibit the model from staying too long in these states (Figure 2). Consequently the recognition path in the Viterbi algorithm will exit the sub-goal model as soon as it reaches the final stage of the sub-goal. After applying changes into the state transition matrix of the sub-goal models, full-trial recordings are decoded using the real-time Viterbi algorithm and modified HMMs, to make sure the spotting performance of the system is maintained after modification.

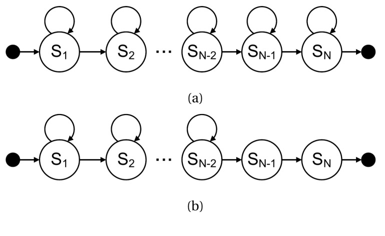

Fig. 2. *Sub-goal left-to-right HMM topology (a) before and (b) after the modification. Here, S_i represents i-th state.*

V. Data Collection

Recordings were gathered from 38 participants, aged between 18 and 80, completed multiple individual sub-goals and full tea-making trials. In all cases synchronized CIC and Kinect outputs were recorded. In the full trial recordings, subjects were asked to make 4 different types of tea as described in section IV-B. In total, there are 1,124 recordings of isolated actions (4.01 hours) and 70 recordings of complete tea-making sessions (1.6 hours) (Table I).

TABLE I. *Data used in AR development. Durations are in hours.*

Sub-goal	Trials	Dur.	Sub-goal	Trials	Dur.
Pour kettle	148	0.50	Stir	138	0.56
Add milk	69	0.22	Toy with kettle	26	0.07
Add sugar	220	0.40	Boil water	125	0.22
Add teabag	237	0.44	Toy with milk	30	0.11
Fill kettle	180	0.73			
Remove teabag	168	0.41	Full trial	70	1.6

VI. Experimental results and analysis

In this section we describe the experiment results for full trial experiments (Table II) and report the time-delay reduction after using the modified HMMs for real-time Viterbi decoder for detection of sub-goals in full-trials (Table III) .

During the full trial experiments, all of the isolated sub-goal recordings were used for model training. The number of states in the sub-goal HMMs , N ($5 \leq N \leq 60$), and the number of GMM components in the single-state "toying" model, M ($1 \leq M \leq 512$), were determined empirically on the development data. Each state of the sub-goal HMM was associated with a single component Gaussian probability density function (PDF). Best results were achieved by using $N = 20, 20, 50$ and 70 states for the sub-goal model, and $M = 256, 512, 512$ and 32 GMM components for the "toying" models for "Front actions" ("Add sugar", "Add tea-bag" and "Remove tea-bag"), 'Add Milk', 'Pour kettle' and 'Fill kettle' detectors, respectively. The results of the recognition experiments on full trials is shown in Table II.

TABLE II. *Results of full-trial sub-goal detection experiments (Ins = number of insertions, %Acc. = % Recognition accuracy, %FA = % False alarms, and %FR = % False rejections).*

Sub-goal	Samples	Correct	Ins	%Acc.	%FA	%FR
Pour kettle	53	53	0	100	0	0
Add milk	38	37	1	94.7	2.6	2.6
Add sugar	56	53	3	89.2	5.4	5.4
Remove tea-bag	60	56	6	83.3	10	6.7
Add tea-bag	60	58	5	88.3	8.3	3.3
Fill kettle	66	59	10	74.2	15.2	10.6
Stir	71	62	24	70	34	13

Detection accuracy for full trials is calculated as follows: A sub-goal occurring in a full trial is considered to have been correctly detected if and only if the sub-goal is detected by the corresponding detector and the detected and actual sub-goals overlap by 75%. If an actual sub-goal does not overlap with a detected sub-goal by 75% then a deletion (False Rejection (FR)) has occurred. If a detected sub-goal does not overlap with an actual sub-goal by 75%, then an insertion (False Alarm (FA)) has occurred. The % accuracy is given by:

$$\%Acc = \frac{Samples - Deletions - Insertions}{Samples} \times 100 \quad (3)$$

As shown in Table II, the best performance is achieved for the sub-goals "Add milk" and "Pour kettle". These are the only sub-goals for which all of the objects that are involved are fully instrumented (i.e. fitted with a CIC). Recognition of the sub-goals "Add tea-bag", "Add sugar" and "Remove tea-bag" relies mainly on hand coordinate data from Kinect, plus small perturbations of the outputs from the CIC sensors attached to the mug caused by the weight-changes or movement due to adding a sugar cube or tea-bag to the mug, or removing a tea-bag from the mug. Since "Remove tea-bag" involves putting the spoon into the mug and moving it to pick up the tea-bag, the outputs of the mug CIC and the Kinect hand coordinates will be very similar to those for "Stir". Hence the insertion of "Stir" is to be expected. A solution would be to break down the sub-goals into smaller actions, so that "Stir" and the start of "Remove tea-Bag" are both characterized by the same model.

Using the modified HMM set (Section IV-D) results in output time-delay reduction during the real-time decoding without loss in accuracy. Table III shows the mean and variance of the output delays among all detections of a sub-goal in the full-trial recordings, using the real-time Viterbi algorithm. The detections of sub-goals are repeated using the models trained in the full-trail experiment and the modified models. Modification to the HMM-sets reduced the average output delay of detections for all sub-goals. The maximum 0.16 and minimum 3.74 seconds improvement, was achieved for detection of the "Remove teabag" and "Pour kettle" sub-goals, respectively. Also, the modification to HMMs reduced the average delay-time of outputs from 2.6 seconds to an acceptable delay of less than one second.

TABLE III. *Detection's output delay for each sub-goal using the original and modified HMMs*

Models	Sub-goal Output Delay Time (ms)			
	Before Fix		After Fix	
	Mean	Variance	Mean	Variance
Pour kettle	4.553	1.836	0.804	0.459
Add milk	2.025	0.939	0.314	0.321
Fill kettle	3.940	3.054	1.695	1.695
Add sugar	2.206	2.474	1.059	0.837
Add teabag	2.663	1.808	0.264	0.435
Remove teabag	0.737	0.784	0.572	0.339
Stir	2.834	1.543	0.986	0.724
Average	2.647	1.543	0.953	0.744

VII. Conclusion

This paper presents a real-time HMM-based architecture for AR and presents a novel time-delay reduction approach to speed up the decoding phase. The results show that HMMs combined with instrumented objects provide a viable approach to action recognition and using the proposed HMM topology can reduce the time-delay by 64% with no loss in accuracy. Future work will report the results obtained by the system using Deep Neural Network models rather than GMMs.

Acknowledgment

The authors wish to thank all CogWatch partners for their contributions. This work has been supported by the European Commission under the grant FP7-ICT-2011-288912.

References

[1] A. Go, D. Mozaffarian, V. Roger, E. Benjamin, J. Berry, W. Borden, D. Bravata, S. Dai, E. Ford, C. Fox *et al.*, "On behalf of the american heart association statistics committee and stroke statistics subcommittee," *Heart disease and stroke statistics2013 update: a report from the American Heart Association. Circulation*, vol. 127, no. 1, pp. e1–e240, 2013.

[2] C. Barbieri and E. De Renzi, "The executive and ideational components of apraxia," *Cortex*, vol. 24, no. 4, pp. 535–543, 1988.

[3] W.-L. Bickerton, M. J. Riddoch, D. Samson, A. B. Balani, B. Mistry, and G. W. Humphreys, "Systematic assessment of apraxia and functional predictions from the birmingham cognitive screen," *Journal of Neurology, Neurosurgery & Psychiatry*, vol. 83, no. 5, pp. 513–521, 2012.

[4] J. Hermsdörfer, M. Bienkiewicz, J. M. Cogollor, M. J. Russell, E. M. D. Jean-Baptiste, M. Parekh, A. M. Wing, M. Ferre, and C. M. L. Hughes, "CogWatch - automated assistance and rehabilitation of stroke-induced action disorders in the home environment," in *Engineering Psychology and Cognitive Ergonomics. Applications and Services - 10th International Conference, EPCE 2013, Held as Part of HCI International 2013, Las Vegas, NV, USA, July 21-26, 2013, Proceedings, Part II*, 2013, pp. 343–350.

[5] A. Hazell, A. Matthews, A. Worthington, C. Walton, and A. Wing, "Cogwatch-cognitive rehabilitation of apraxia and action disorganisation syndrome: assessing the requirements of healthcare professionals, stroke survivors and carers," in *INTERNATIONAL JOURNAL OF STROKE*, vol. 7. WILEY-BLACKWELL 111 RIVER ST, HOBOKEN 07030-5774, NJ USA, 2012, pp. 78–78.

[6] E. Jean-Baptiste, R. Nabiei, M. Parekh, E. Fringi, B. Drozdowska, C. Baber, P. Jancovic, P. Rotshein, and M. Russell, "Intelligent assistive system using real-time action recognition for stroke survivors," in *Healthcare Informatics (ICHI), 2014 IEEE International Conference on*. IEEE, 2014, pp. 39–44.

[7] O. Amft and G. Tröster, "Recognition of dietary activity events using on-body sensors," *Artificial Intelligence in Medicine*, vol. 42, no. 2, pp. 121–136, 2008.

[8] M. K. Hasan, H. A. Rubaiyeat, Y.-K. Lee, and S. Lee, "A reconfigurable hmm for activity recognition," in *ICACT*, vol. 8, 2008, pp. 843–846.

[9] H. M. Hondori, M. Khademi, and C. V. Lopes, "Monitoring intake gestures using sensor fusion (microsoft kinect and inertial sensors) for smart home tele-rehab setting," in *2012 1st Annual IEEE Healthcare Innovation Conference*, 2012.

[10] Y. Hong, I. Kim, S. C. Ahn, and H. Kim, "Activity recognition using wearable sensors for elder care," in *The Second International Conference on Future Generation Communication and Networking, FGCN 2008, Volume 2, Workshops, Hainan Island, China, December 13-15, 2008*, 2008, pp. 302–305.

[11] H. Junker, P. Lukowicz, and G. Tröster, "Continuous recognition of arm activities with body-worn inertial sensors," in *8th International Symposium on Wearable Computers (ISWC 2004), 31 October - 3 November 2004, Arlington, VA, USA*, 2004, pp. 188–189.

[12] T. Maekawa and S. Watanabe, "Unsupervised activity recognition with user's physical characteristics data," in *15th IEEE International Symposium on Wearable Computers (ISWC 2011), 12-15 June 2011, San Francisco, CA, USA*, 2011, pp. 89–96.

[13] J. Wagner, A. van Halteren, J. Hoonhout, T. Plötz, C. Pham, P. Moynihan, D. Jackson, C. Ladha, K. Ladha, and P. Olivier, "Towards a pervasive kitchen infrastructure for measuring cooking competence," in *5th International Conference on Pervasive Computing Technologies for Healthcare, PervasiveHealth 2011, Dublin, Ireland, May 23-26, 2011*, 2011, pp. 107–114.

[14] H.-W. Gellersen, M. Beigl, and H. Krull, "The mediacup: Awareness technology embedded in an everyday object," in *Handheld and Ubiquitous Computing*. Springer, 1999, pp. 308–310.

[15] T. Stiefmeier, D. Roggen, G. Ogris, P. Lukowicz, and G. Tröster, "Wearable activity tracking in car manufacturing," *IEEE Pervasive Computing*, vol. 7, no. 2, pp. 42–50, 2008.

[16] E. Berlin, J. Liu, K. Van Laerhoven, and B. Schiele, "Coming to grips with the objects we grasp: detecting interactions with efficient wrist-worn sensors," in *Proceedings of the fourth international conference on Tangible, embedded, and embodied interaction*. ACM, 2010, pp. 57–64.

[17] J. Wu, A. Osuntogun, T. Choudhury, M. Philipose, and J. M. Rehg, "A scalable approach to activity recognition based on object use," in *IEEE 11th International Conference on Computer Vision, ICCV 2007, Rio de Janeiro, Brazil, October 14-20, 2007*, 2007, pp. 1–8.

[18] J. M. Cogollor, C. Hughes, M. Ferre, J. Rojo, J. Hermsdörfer, A. Wing, and S. Campo, "Handmade task tracking applied to cognitive rehabilitation," *Sensors*, vol. 12, no. 10, pp. 14 214–14 231, 2012.

[19] A. M. Genest, C. Gutwin, A. Tang, M. Kalyn, and Z. Ivkovic, "Kinectarms: a toolkit for capturing and displaying arm embodiments in distributed tabletop groupware," in *Proceedings of the 2013 conference on Computer supported cooperative work*. ACM, 2013, pp. 157–166.

[20] R. Nabiei, M. Parekh, E. Jean-Baptiste, P. Jancovic, and M. Russell, "Object-centred recognition of human activity," in *Healthcare Informatics (ICHI), 2015 International Conference on*. IEEE, 2015, pp. 63–68.

[21] L. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[22] C. M. L. Hughes, C. Baber, M. Bienkiewicz, and J. Hermsdörfer, "Application of human error identification (HEI) techniques to cognitive rehabilitation in stroke patients with limb apraxia," in *Universal Access in Human-Computer Interaction. Applications and Services for Quality of Life - 7th International Conference, UAHCI 2013, Held as Part of HCI International 2013, Las Vegas, NV, USA, July 21-26, 2013, Proceedings, Part III*, 2013, pp. 463–471.

[23] M. Gales and S. Young, "The application of hidden markov models in speech recognition," *Foundations and trends in signal processing*, vol. 1, no. 3, pp. 195–304, 2008.

[24] M. Najafian, A. DeMarco, S. J. Cox, and M. J. Russell, "Unsupervised model selection for recognition of regional accented speech." in *INTERSPEECH*, 2014, pp. 2967–2971.

[25] M. Najafian, S. Safavi, A. Hanani, and M. Russell, "Acoustic model selection using limited data for accent robust speech recognition," in *Signal Processing Conference (EUSIPCO), 2014 Proceedings of the 22nd European*. IEEE, 2014, pp. 1786–1790.

[26] M. Najafian, "Acoustic model selection for recognition of regional accented speech," Ph.D. dissertation, University of Birmingham, 2016.

[27] J. C. Spohrer, P. F. Brown, P. H. Hochschild, and J. K. Baker, "Partial traceback in continuous speech recognition," in *Proc. of the IEEE International Conference on Cybernetics and Society*, 1980.