2019 27th European Signal Processing Conference (EUSIPCO)

Compensating for Object Variability in DNN–HMM Object-Centered Human Activity Recognition

Yikai Peng
The University of Birmingham
Birmingham, UK
yxp576@bham.ac.uk

Peter Jančovič
The University of Birmingham
Birmingham, UK
p.jancovic@bham.ac.uk

Martin Russell
The University of Birmingham
Birmingham, UK
m.j.russell@bham.ac.uk

***Abstract*—This paper describes a deep neural network – hidden Markov model (DNN-HMM) human activity recognition system based on instrumented objects and studies compensation strategies to deal with object variability. The sensors, comprising an accelerometer, gyroscope, magnetometer and force-sensitive resistors (FSRs), are packaged in a coaster attached to the base of an object, here a mug. Results are presented for recognition of actions involved in manipulating a mug. Evaluations are performed using over 24 hours of data recordings containing sequences of actions, labelled without time-stamp information. We demonstrate the importance of data alignments. While the DNN-HMM system achieved error rate below 0.1% for matched train-test conditions, this increased up to 26.5% for highly mismatched conditions. The error rate averaged over all conditions was 1.4% when using multi-condition training and decreased to 0.8% by employing feature augmentation. The use of FSR feature compensation, specific to weight variability, resulted in 0.24% error rate.**

***Index Terms*—Action recognition, deep neural networks, hidden Markov models, DNN-HMM, sensors, instrumented objects, compensation, feature augmentation**

I. Introduction

Human activity recognition (AR) from sensor data has attracted considerable research efforts during the past decade. This research has a large potential for applications in healthcare and smart environments. For instance, it could be used to help dementia or stroke patients with completing their daily tasks themselves at home and continue leading an independent life while also reducing financial costs [1]–[3].

A widely used approach to AR is to attach sensors to the body of users [4]–[7]. However, due to the need for users to wear sensors, this approach is not suitable for some applications, e.g., rehabilitation of stroke patients. A “scene-oriented” approach [8], [9], in which an external video sensor and image processing is used to identify and track the user and objects during a task is unobtrusive, however, it 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. We consider an “object-centric” view of AR, in which actions are characterised in terms of how they are “experienced” by the objects involved. A popular option for instrumentation is to use Radio Frequency Identification tags to identify which objects were picked up [10], [11]. However these do not provide sufficiently rich information, and an antenna needs to be worn.

This paper extends our recent research on AR using instrumented objects [12], [13], which demonstrated the use of a set of GMM-HMM detectors, each modelling a particular component of a task, in a scenario with a known set of objects. We present the development of a DNN-HMM AR system and study of compensation approaches to deal with object variability. We explore the use of multi-condition training and feature augmentation as techniques for dealing with a generic variability and also compare results to FSR-based feature compensation designed specifically to deal with varying the weight of the object. Experimental evaluations are performed on over 24 hours of data recordings.

II. Instrumentation and Sensors

The development of instrumented objects was based on an earlier work conducted as part of the CogWatch EU project [1], [12]. The sensors and circuitry, being small and discrete, are packaged into a sensorised ‘coaster’ (SC), fitted to the underside of an object. This ensures the instrumented objects appearing normal and functioning as expected.

Figure 1 depicts the developed SC, with a view of the 3D printed casing, inner structure and attachment to an object (mug). The SC contains a 3-axis accelerometer, 3-axis gyroscope, 3-axis magnetometer, 3 force-sensitive resistors (FSRs), microcontroller, motherboard and Lithium polymer battery. From these sensors, 12 sample values are acquired at a time. The sampling frequency is set to 20 Hz, as this was shown to be sufficient for AR [14], [15]. Sensor data were calibrated based on a few seconds of measurements obtained at the beginning of a recording session – the coaster was kept stationary for calibrating the accelerometer and gyroscope, and being rotated for calibrating the magnetometer. Data are then sent in bytes via Bluetooth Low Energy [16] to a desktop computer. We measured, on trials of 6 minutes long, data loss in transmission. The average data loss rate was 0.28% of samples, with no more than 3 consecutive samples lost. As data loss was sparse, any lost sample value was obtained as the average value of the adjacent samples.

In this paper, the coaster was fitted to a mug, which was manipulated by a person. Figure 2 shows an example of the output signals from all sensors (accelerometer, gyroscope, magnetometer, FSR) when a specific sequence of actions was performed with the mug. The x-axis represents the sample

978-9-0827-9703-9/19/$31.00 ©2019 IEEE

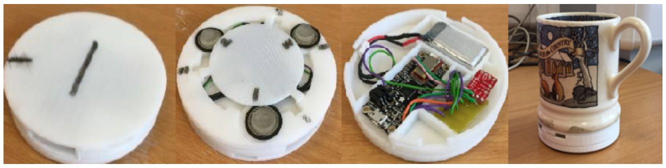


Fig. 1. The developed sensorised coaster. From left to right: the outer casing of the coaster from top and from bottom, the inner design, and the coaster fitted to a mug.

index and the y-axis the output of each sensor. The mug was stationary for the first 5.5 seconds (i.e., up to the sample index 110). It was then lifted up, tilted, and put down at time of 15 seconds (i.e., sample index 300) and left stationary for the final 5 seconds.

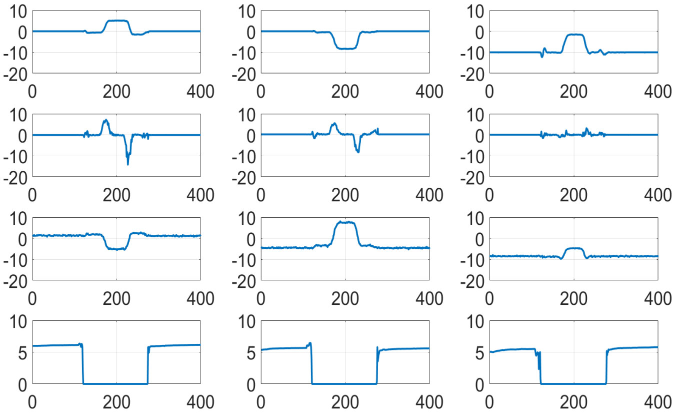


Fig. 2. An example of output signals from sensors of a coaster attached to a mug during the sequence of actions: ‘stationary’, ‘lift up’, ‘tilt’, ‘put down’, and ‘stationary’. The x-axis denotes the sample index and the y-axis denotes normalised analog output from a sensor. The sensors, from top row to bottom, are: accelerometer, gyroscope, magnetometer, FSR. The columns indicate x, y and z dimension from each sensor.

III. Action Recognition System

This section describes the AR system, including feature representation, HMM-based modelling of sensor data, and approaches we employed to compensate for object variability. In this paper, we consider that the following five types of actions are performed with a mug: lift the mug up (LU), put the mug down (PD), tilt the mug (TT), keep the mug stationary on the table (ST) and keep the mug stationary in air (ST_UP).

A. Feature extraction

The raw 12-dimensional samples received from the SC comprise of values from FSR, accelerometer, gyroscope, and magnetometer. As the measurements from the 3 individual FSRs are affected by variations of the elastic force within the self-adhesive bumper stops fitted to the FSRs, we used only an aggregated FSR value, denoted by F, calculated as the sum of the individual FSR values (F_x, F_y, F_z), i.e., $F = F_x + F_y + F_z$. This resulted in having a raw 10-dimensional data. In GMM-HMM system, these raw features were appended with their temporal derivatives (delta and delta-delta features), which were calculated using a regression over 2 proceeding and following frames [17]. This resulted in 30-dimensional feature representation. In DNN-HMM system, the raw 10-dimensional data were spliced with a window of ± 2 (i.e., 2 proceeding and following raw data vectors), resulting in 50-dimensional feature representation. This was then decorrelated using linear discriminant analysis (LDA) and reduced to 10-dimensional feature vectors.

B. DNN-HMM recognition system

To model each action, we employed a left-to-right HMM, with no state skip allowed. Initially, we used HMMs of 3 states for each action, considering these to represent the stationary part of the action and start and end transitions. However, experimental evaluations showed that better recognition results, as well as better alignments are obtained by having HMMs with 5 states. The training of a DNN-HMM system requires to have state-level alignments of data. Such alignments were obtained based on a GMM-HMM system. The quality of the alignments can significantly affect the quality of the trained DNN-HMM system [18]. This is in particular important in our case as we do not have available time-stamp information that would indicate the start and the end of each action in the sequence. The GMM-HMM system uses diagonal covariance matrices with the number of Gaussian mixture components per state set to 30. The DNN-HMM system contains 3 hidden layers, with 256 neurons in each layer. A mini-batch stochastic gradient descent, with a batch size of 128, is used to train the DNN. The learning rate was varied over the iterations – it was set to 0.001 at the beginning of the training but then decreased to 0.0001 after 15 iterations. The output layer in the DNN corresponds to the overall number of states across all action models, i.e., 25 for our 5 state HMMs. The parameters of the GMM-HMM and DNN-HMM models were chosen empirically. The GMM-HMM and DNN-HMM recognition systems are built using Kaldi [19].

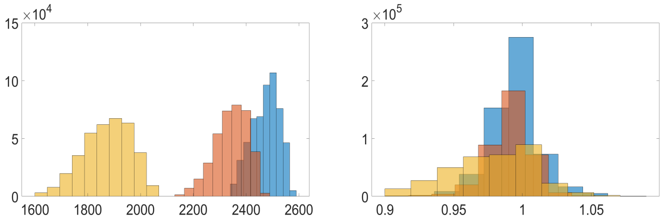
C. Compensating for object variability

Our overall aim is to study how models developed for one type of object could be reconfigured and transferred to create models for a different object. Objects may vary in different aspects, e.g., shape, size, or weight. In this paper, we focus on varying the weight of the object as an example to study different techniques for compensating this variability. Variations in weight of the object are expected to influence mainly data from the FSRs. We also confirmed this by conducting LDA analysis of the importance of features for classifying data from different weight conditions.

1) Multi-condition training: An approach for dealing with variability, often used in automatic speech recognition to improve robustness to noise, is to include a wide variety of conditions into training data [20]–[22]. This is referred to as a multi-condition training, or more recently as data augmentation. While this can provide better robustness to variability, it typically also degrades the performance to some extent when the test conditions match the training data conditions because of a greater ambiguity of trained models.

2) FSR feature compensation: To deal specifically with variability in object weight, we can exploit the FSR measurements during periods when the object is stationary to normalise out the effect of weight. It is reasonable to assume that the object is stationary for at least a short period before it is being manipulated. We considered this to be the initial 1 second (i.e., 20 samples) and calculated the average aggregated FSR value over this period, denoted by F_{stat}. The compensated FSR value at each sample index n, denoted by $F_{comp}(n)$, is then obtained by normalising the aggregated FSR value (see Section III-A) by the F_{stat} value as $F_{comp}(n) = (F_x(n) + F_y(n) + F_z(n))/F_{stat}$. Figure 3 shows the distribution of FSR values for a mug with different weights before (a) and after (b) the normalisation. It can be seen that while the unnormalised FSR values have a different range of values, the histograms for different weight conditions are well overlapping after the normalisation was applied. Note that the uneven location of distributions for different weights in Figure 3 (a) is due to the FSRs responding in a non-linear manner to the actual weight of the object.

3) Feature augmentation: A more general approach to deal with any type of variability in a DNN-HMM system is feature augmentation. This approach has been used in automatic speech recognition to inform the system about the speaker or conditions of the current data [21]. In this approach, each feature vector representing data is augmented with additional information that represents the object variability. We explored augmenting the same FSR value F_{stat} as was used in the feature normalisation and also the measured weight of the object for comparison. The use of such augmented features can enable the DNN-HMM to learn the relationship between the weight of the object and the features extracted from the sensor signals.



(a) (b)

Fig. 3. Histograms of aggregated FSR values for different weight conditions before (a) and after (b) the compensation.

IV. Experimental Results and Analysis

A. Data collection

Data collection used for experimental evaluations in this paper consists of 720 recordings made by a single subject; in total, over 24 hours of recordings. The use of a single subject was intentional at this stage of our research in order to eliminate subject variability in object manipulation. Each recording contained a prescribed sequence of actions, repeated several times. The same action sequences were used to make recordings with 3 different weight conditions of a mug: (1) fully-filled (FW); (2) a half-filled (HW) and (3) empty (NW). Each recording was associated with a label file containing the sequence of actions performed but no start-end time information of actions. Summary of the collected amount of data and its split into the training and testing set is given in Table I. Note that the amount of data in the testing set is larger than in the training set because the training set was fixed earlier in our experimentation and additional recordings performed at a later stage were then added to the testing set.

TABLE I
THE AMOUNT OF DATA COLLECTED FOR DIFFERENT CONDITIONS (IN HOURS) AND ITS SPLIT INTO THE TRAINING SET AND TESTING SET.

Dataset	Weight condition			Total
	Full (FW)	Half (HW)	Null (NW)	
Train	3.5h	3.2h	3.1h	9.8h
Test	4.9h	4.8h	4.8h	14.5h

B. Baseline system and the effect of alignments

First experiments were performed using DNN-HMM systems trained on specific weight conditions, with the aim of analysing the effect of mismatch between the weight of the object used during the training and testing. Initially, each of the DNN-HMM systems was trained based on data alignments obtained from a GMM-HMM system also trained on the corresponding condition-specific data. Results, presented in Table II, showed rather large error rates even when the train and test conditions were matched (i.e., diagonal in the table). We analysed possible causes of this and found that the obtained data alignments were considerably inaccurate in the start-end times of some actions, in particular, ST and actions neighbouring with ST. This then had consequences on the trained models and the recognition accuracy.

TABLE II
ERROR RATES (IN %) OBTAINED BY DNN-HMM SYSTEMS TRAINED ON WEIGHT-DEPENDENT DATA. DATA ALIGNMENTS OBTAINED BY CORRESPONDING CONDITION-SPECIFIC GMM-HMM SYSTEM.

Training data weight conditions	Testing data weight conditions		
	FW	HW	NW
FW	5.96	12.11	32.21
HW	48.60	12.17	21.19
NW	39.33	12.47	6.40

C. Multi-condition training and improved alignments

Results obtained using multi-condition training are presented in the first line in Table III. It can be seen that the use of multi-condition data resulted in significant improvements in comparison to condition-specific training, while also not requiring to know the type of conditions.

TABLE III
ERROR RATES (IN %) OBTAINED BY DNN-HMM SYSTEMS TRAINED ON MULTI-CONDITION DATA AND WEIGHT-DEPENDENT DATA. DATA ALIGNMENTS FROM MULTI-CONDITION GMM-HMM SYSTEM.

Training data weight conditions	Testing data weight conditions		
	FW	HW	NW
Multi-condition	0.06	1.37	2.72
FW	0.09	0.00	26.50
HW	0.19	0.09	7.37
NW	24.90	2.68	0.03

Analysing these experiments in more depth, we found that alignments produced here by the GMM-HMM system, which was now also trained using the multi-condition data, were considerably better than those obtained using the condition-specific training data. Thus, in order to see the effect of the multi-condition data when using the same data alignments, we repeated the condition-specific training experiments from Section IV-B and results are given in the following rows in Table III. It can be seen that all results improved significantly in comparison to those in Table 2, demonstrating that the quality of alignments is crucial when time-stamp annotations are not available for training data. Error rates achieved when the weight condition of the training and testing data match are very low. However, the error rate is still significantly high when there is a mismatch, such as between the FW and NW conditions. Note that recognition results are good for mismatch between FW and HW due to the closeness of their FSR distributions (see Figure 3 (a)). All following experiments used the alignment from the multi-condition GMM-HMM system.

D. FSR feature compensation

Results presented in Table IV are obtained by systems trained on multi-condition data and condition-specific data but using the compensated FSR features. For condition-specific training, it can be seen that the accuracy improved significantly for the weight mismatch between the train and test data, e.g., for NW training and FW testing, the error reduced from 24.90% to 1.07%. The error rates also decreased considerably for multi-condition training.

TABLE IV
ERROR RATES (IN %) OBTAINED BY DNN-HMM SYSTEMS TRAINED ON WEIGHT-DEPENDENT DATA AND MULTI-CONDITION DATA AFTER EMPLOYING THE FSR FEATURE COMPENSATION.

Training data weight conditions	Testing data weight conditions		
	FW	HW	NW
Multi-condition	0.03	0.09	0.59
FW	0.06	0.09	0.28
HW	0.34	0.09	0.34
NW	1.07	0.03	0.00

E. Feature augmentation

Results obtained when using multi-condition training with feature augmentation are presented in Table V. It can be seen that, in overall, the feature augmentation improved results of multi-condition training (as given in Table III) – the average error rate over the weight conditions decreased from 1.38% to 0.81%. However, results are little worse than those obtained by using the FSR feature compensation. This indicates that the DNN was able to utilise information from the augmented features but did not find as good solution as knowledge-based FSR feature compensation, perhaps due to reaching a local minima. However, while the presented FSR feature compensation is only applicable to weight, the feature augmentation approach presents a general approach to compensating variability.

TABLE V
ERROR RATES (IN %) OBTAINED BY THE DNN-HMM SYSTEM WITH MULTI-CONDITION TRAINING AND FEATURE AUGMENTATION.

Training data weight conditions	Testing data weight conditions		
	FW	HW	NW
Multi-condition	0.28	0.28	1.87

V. CONCLUSION

This paper presented a DNN-HMM system for recognition of actions involved in manipulating an instrumented object and explored different strategies to deal with object variability. Experiments were performed using 24 hours of data recorded using a mug attached with a newly developed sensorised coaster, containing an accelerometer, gyroscope, magnetometer and FSRs. We demonstrated the importance of having a good quality data alignment in the absence of time-stamp label information. Experiments using matched conditions resulted in below 0.1% error rate but this increased to up to 26.5% for strongly mismatched conditions. The use of multi-condition training resulted in 1.4% average error rate and incorporating also feature augmentation decreased the error further to 0.8%. The FSR feature compensation, only applicable for compansating weight variability, performed similarly as matched train-test conditions.

In this work, we demonstrated the effectiveness of the proposed techniques for compensating variability due to the weight of the object. In our future work, we plan to also consider variability of other properties of objects used, such as, size and shape, and we will also consider variability due to different subjects manipulating the objects.

REFERENCES

[1] "Cogwatch: Cognitive rehabilitation of apraxia and action disorganisation syndrome," *http://www.cogwatch.eu/*.

[2] W. L. Bickerton, M. J. Riddoch, D. Samson, A. Balani, B. Mistry, and G. W. Humphreys, "Systematic assessment of apraxia and functional predictions from the Birmingham cognitive screen," *Journal of Neurology, Neurosurgery, and Psychiatry*, pp. 513–521, 2012.

[3] E. M. D. Jean-Baptiste, R. Nabiei, M. Parekh, E. Fringi, B. Drozdowska, C. Baber, P. Jančovič, P. Rotshein, and M. Russell, “Intelligent assistive system using real-time action recognition for stroke survivors,” in *IEEE Int. Conf. on Healthcare Informatics*, 2014.

[4] L. Bao and S. S. Intille, “Activity recognition from user-annotated acceleration data,” in *Int. Conf. on Pervasive Computing*, 2004, pp. 1–17.

[5] N. Ravi, N. Dandekar, P. Mysore, and M. L. Littman, “Activity recognition from accelerometer data,” in *Innovative Applications of Artificial Intelligence Conf.*, 2005, pp. 1541–1546.

[6] O. Amft. and G. Tröster, “Recognition of dietary activity events using on-body sensors,” *Artificial Intelligence in Medicine*, vol. 42, pp. 121–136, 2008.

[7] J. Wagner, T. Ploetz, A. V. Halteren, J. Hoonhout, P. Moynihan, D. Jackson, and C. Ladha, “Towards a pervasive kitchen infrastructure for measuring cooking competence,” in *Int. Conf. on Pervasive Computing Technologies for Healthcare*, 2011, pp. 107–114.

[8] K. Nickel and R. Stiefelhagen, “Pointing gesture recognition based on 3D tracking of face, hands and head orientation,” in *Int. Conf. on Multimodal Interfaces, Vancouver, Canada*, 2003, pp. 140–146.

[9] J. Y.-H. Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, “Beyond short snippets: Deep networks for video classification,” in *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, 2015.

[10] J. Wu, A. Osuntogun, T. Choudhury, M. Philipose, and J. M. Rehg, “A scalable approach to activity recognition based on object use,” in *IEEE Int. Conf. on Computer Vision*, 2007.

[11] E. Berlin, J. Liu, K. V. Laerhoven, and B. Schiele, “Coming to grips with the objects we grasp: Detecting interactions with efficient wrist-worn sensors,” in *Proc. TEI*, 2010, pp. 57–64.

[12] R. Nabiei, M. Parekh, E. Jean-Baptiste, P. Jančovič, and M. Russell, “Object-centred recognition of human activity for assistance and rehabilitation of stroke patients,” in *IEEE Int. Conf. on Healthcare Informatics, Dallas, USA*, 2015, pp. 63–68.

[13] R. Nabiei, M. Najafian, M. Parekh, P. Jančovič, and M. Russell, “Delay reduction in real-time recognition of human activity for stroke rehabilitation,” in *IEEE Int. Workshop on Sensing, Processing and Learning for Intelligent Machines, Aalborg, Denmark*, 2016.

[14] A. Khan, N. Hammerla, S. Mellor, and T. Plötz, “Optimising sampling rates for accelerometer-based human activity recognition,” *Pattern Recognition Letters*, vol. 73, pp. 33–40, April 2016.

[15] R. Nabiei, “Action recognition using instrumented objects for stroke rehabilitation,” Ph.D. dissertation, University of Birmingham, UK, 2017.

[16] C. Gomez, J. Oller, and J. Paradells, “Overview and evaluation of bluetooth low energy: An emerging low-power wireless technology,” *Sensors*, vol. 12, pp. 11 734–11 753, August 2012.

[17] S. Young, G. Evermann, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, “The HTK book,” 2006.

[18] P. Lin, D. C. Lyu, Y. F. Chang, and Y. Tsao, “Temporal alignment for deep neural networks,” in *IEEE Global Conf. on Signal and Information Processing*, 2015.

[19] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlíček, Y. Qian, and P. Schwarz, “The KALDI speech recognition toolkit,” in *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2011.

[20] H. G. Hirsch and D. Pearce, “The Aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions,” in *Proc. of ITRW ASR, Paris, France*, 2000, pp. 181–188.

[21] M. L. Seltzer, D. Yu, and Y. Wang, “An investigation of deep neural networks for noise robust speech recognition,” in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, 2013.

[22] X. Cui, V. Goel, and B. Kingsbury, “Data augmentation for deep neural network acoustic modeling,” in *IEEE Int. Conf. on Acoustic, Speech and Signal Processing*, 2014.