

Adoption of the activation function fusion approach to identify human activity recognition in a semi-supervised neural network

Netzahualcoyotl Hernandez¹, Chris Nugent¹, Ian McChesney¹, Shuai Zhang¹, Jesus Favela²

¹Ulster University, Newtownabbey, Belfast, Northern Ireland, UK

²Ensenada Center for Scientific Research and Higher Education (CICESE), Baja California, Ensenada, Mexico

Abstract

INTRODUCTION: Neural networks are a popular type of algorithm for human activity monitoring which can build intelligent systems from labelled data in an automated fashion. Obtaining accurately labelled data is costly; it requires time and effort, which can be cumbersome because it interrupts the user activity stream. In conjunction with the ubiquitous presence of embedded technology, neural networks present new research opportunities for human activity monitoring in smart home environments.

OBJECTIVES: We propose a human activity classification method that requires a limited amount of labelled data, which consists of a concatenation method for classifying human activities built upon the fusion of neural network activation functions.

METHODS: Our methodology builds a neural network model that receives the sensor data through the input layer to then distribute it among the different vertical hidden layers, which implement different activation functions simultaneously. Next a hidden layer combines activation functions by utilising a concatenation method. Finally, the neural network provides classes to the unlabelled sensing data. We conducted an evaluation utilising an open-access dataset. We compared the activity recognition accuracy of our approach utilising 25%, 50%, and 75% of labelled data against a conventional shallow neural network trained with the 100% of labelled data available.

RESULTS: Results show an improvement in the accuracy of the activity classification regardless of the portion of labelled data available. It was observed that the highest achieved accuracy when using 25% of activation function fusion data outperformed results compared to when using 100% of labelled data in a conventional shallow network (i.e., increase in accuracy of 2.7%, 3.7%, 4.8%, and 0.9% across the activity recognition of four subjects).

CONCLUSION: The approach proposed showed an improvement in the accuracy of classifying human activity when a limited amount of labelled data is available.

Received on 23 September 2019; accepted on 08 November 2019; published on 19 November 2019

Keywords: activation function, data fusion, neural networks, activity recognition

Copyright © 2019 Netzahualcoyotl Hernandez *et al.*, licensed to EAI. This is an open access article distributed under the terms of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/3.0/>), which permits unlimited use, distribution and reproduction in any medium so long as the original work is properly cited.

doi:10.4108/eai.30-10-2018.161669

1. Introduction

In recent years we have experienced the widespread use of mobile devices such as smartphones and smart-watches. Similarly, the interest in research on embedded sensors in daily life objects (e.g., kitchen appliances)

*Corresponding author. Email: hernandez_cruz-n@ulster.ac.uk

is growing, which enables opportunities in the field of computer engineering that makes user's interactions richer through applications which can build on this variety of sensing data. This ubiquitous presence of sensing technology opens the opportunity to assist users in areas such as smart homes [27], rehabilitation [25], health monitoring [33], and activity recognition [23].

Activity recognition is the problem of identifying human actions given a collection of sensed data. It takes advantage of off-the-shelf technologies available within the user's environment (e.g., smartphones, smart-watches) to enable the collection of data from daily living scenarios. In this context, machine learning, can help in the automatic recognition of daily living tasks due to its capacity to handle differences between sensor readings and features in the domain, for what neural networks have been shown to be an effective machine learning approach in such scenarios [23].

1.1. Neural networks

A neural network (NN) is an interconnected assembly of simple processing elements, whose functionality is inspired by the human neuron. The processing ability of the network rely in the inter-unit connection strengths obtained by process of adaptation by learning from a set of training patterns, modelled after the human brain to recognise patterns in a given set of data [24, 26].

In order to recognise patterns, a NN learns to approximate a relationship between inputs and outputs (assuming they relate by correlation of causation) [26]. In the process of learning and classification, a NN can rely on two alternatives: i) experience, in which case, a successful outcome depends upon labelled dataset. Alternatively, ii) characterising data and finding similarities among them, in which case there is no labelled data available; therefore, the NN is meant to learn from clustering segregation or association rules. These approaches are also known as supervised learning [29] and unsupervised learning [8] respectively.

Supervised learning is an approach that requires a large amount of labelled data available for training purposes [3, 4], the generation of which is expensive and time-consuming. Unlike supervised learning, however, semi-supervised learning methods, maximize the potential of the portion of labelled data available. They aim at leveraging the amount of labelled data to provide a higher generalisability within the classification classes.

The problem with activity recognition methods requiring a large amount of labelled data is the limited annotation available in day-to-day scenarios where human activities are unexpected and the impact that the lack of unlabelled data can have on those activity

recognition models. Unlike supervised learning, the unsupervised approach does not depend on labelled data to cope with its task. However, considering that there is no knowledge known apriori, unsupervised approaches can lead to lower accuracy.

Overall, achieving high-performance in activity recognition with limited labelled data is a challenging topic that has attracted researchers' attention in recent years [16]. The problem of human activity recognition motivates different approaches, such as feature learning adjustments and weight extrapolation for which NN has proven to be effective. For example, Ding et al. proposed a method to efficient NN using weight extrapolation under the same feature space, demonstrating the effectiveness of feature treatment to achieve correct class classification [6]. Other approaches, include handcrafted annotation under the concept of active learning, where a trained model can iteratively be re-trained upon user's feedback annotation as new data emerge [2]. Stikic et al. proposed utilising a graph-based approach, which connects labelled and unlabelled data and builds multiple graphs to propagate the labels based on the similarity between features [31]. Although feasible, these approaches present the issue of interrupting the user's day-to-day activities or rely on a single sensory modality. In the context of human activity recognition, disrupting the user's activities can affect the correlation between labelled and unlabelled data, whereas a single sensor modality limits the acquisition of rich data available nowadays from the ubiquitous technology available in activities of daily living.

1.2. Data fusion

In this context, data fusion exploits the natural synergy brought about by use of multiple data sources. It consists of the integration of multi-sensor data that exploits the natural synergy brought by multiple sources to achieve inferences that could not be feasible from a single sensor [20]. Some recent research has successfully adopted data fusion as a technique to recognise human activities, for example G. Abebe and A. Cavallaro described a study in which they concatenated feature information from different technology domains (i.e., inertial sensing and first person video images) by implementing a classification model over a NN [1]. Similarly, under the concept of transfer learning, N. Hernandez et al. showed the benefit of data fusion by selecting and fusing features from across two different classifier models, with activity recognition higher compared to the accuracy achieved by the models individually [11]. Z. Wang et al. approach used kernel fusion upon extreme learning machine techniques in which they utilised Gaussian kernels [32]. Other research in sensor fusion has shown positive

results when analysing mobile and ambient sensing technology for activity recognition. These studies achieved high classification accuracy by fusing different data features, sensor types, or algorithm kernels which enable them to maximise the information contained across these multi-modal sensor streams.

In this paper, therefore, we focus on the benefits of fusion data as an elegant approach to design a form of feature derived from activation function fusion when small portions of labelled data are available. We elaborate on fusion data being a technique that has been explored to improve the recognition of activities in machine learning. Our interest in NN relies on the computational benefit of Deep Learning in NN from implementing multiple layers to progressively extract higher-level features. In this paper, therefore, we elaborate and present the results of our methodology that proves to be effective when utilising two hidden layers; one of which consists of three vertical layers, which shows the benefit of fusing activation functions and opens the opportunity for developing more complex NN.

Although to the best of our knowledge, activation function fusion in NN is a topic that has not been explored before, in Section 2, we discuss the closely related work. Section 3 presents our approach and methodology, in which we give a detailed description of the dataset and experimental setting. We conclude the paper with results, future work, and a brief discussion along Section 4 and 5; respectively.

2. State of the art

In general, a NN consist of three layers, Input-, Hidden-, and Output -layer. In contrast to a traditional NN, which builds upon a single hidden layer, modern NN structures consist of recurrent feedforward networks that are organised having two or more hidden layers. This approach benefits from the computational power derived from implementing multiple layers to extract higher-level features progressively [9]. Layers are constructed out of nodes, which is where the computation happens. Nodes connect in such a way that each layer's output represents the subsequent layer's input. A node combines input from the data with a set of coefficients calculated based on a given activation fusion, which fires or is activated when it encounters sufficient stimuli.

2.1. Activation function

Activation functions are a biological inspiration from activity in the human brain, where different stimuli activate different neurons. In this context, an activation function is an algorithm that shapes the behaviour of a NN by assigning significance to inputs concerning what the algorithm is trying to learn; hence, it will

either amplify or dampen input data aiming to uncover patterns within the given data [10].

To illustrate the activation function functionality, in Figure 1 we show a neuron, which receives data as a set of numerical inputs x_1, x_2, \dots, x_n which is then combined with a set of weights w_1, w_2, \dots, w_n and a bias element in order to produce a single numerical value y , computed by the activation function α as $y = \alpha((w_1x_1, w_2x_2, \dots, w_nx_n) + b)$. In general, the activation function determines the type of function that the NN represents. Some of the most common activation functions are the linear, hyperbolic tangent, logistic, and rectified linear [21].

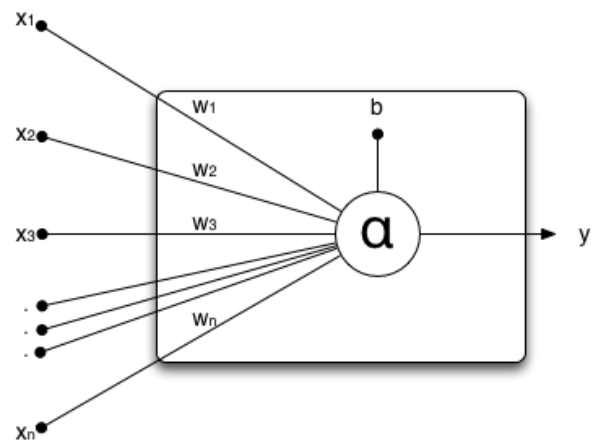


Figure 1. Graphical representation depicting a basic unit of a neuron. Where x = numerical inputs, w = weight, b = bias, α = activation function, and y = outcome.

Given that each activation function processes their input based on particular mathematical properties, some of them will activate a neuron when their value overtakes a certain threshold, while other activation function would remain inactive. Hence, performing an arithmetical calculation (e.g., adding or subtracting) on the outcome from different activation functions can lead to the cancellation of their properties. In this regards, data fusion offers the opportunity to associate data without altering their properties.

2.2. Multimodal fusion

Data fusion can be defined as a multiple level or multifaceted process to detect, associate, correlate, estimate, and combine data and information from several sources [19]. The integration of multi-sensor data exploits the natural synergy brought by multiple sources allowing for useful information gain and achieving inferences that could not be feasible from a single sensor. In this context, therefore, a large volume of heterogeneous sources (e.g., technology capacity

or device position) might leverage data analysis and enhance decision-making [20].

There are different architectures to data fusion, which fall into three general categories: direct fusion (i.e., competitive type), representation via feature vectors (i.e., complementary type), and processing of each sensor to achieve high-level inference (i.e., cooperative type). Depending on the data conditions, a particular mechanism should drive the data towards one or another approach. For instance, redundant data can be treated at the raw level involving classic estimation methods, while complimentary data can operate at the feature level [13]. Due to the intrinsic nature of the methodology proposed in this paper, we focus on data fusion using two architecture categories. We utilise cooperative fusion in which the sensing data provided by the input sources represents different parts of the scene (i.e., wearable and kitchen appliances) to achieve a piece of complete global information. Also, we utilise complementary fusion by adopting different NN's activation functions, in which the information provided represents different perspectives from the same scene and is combined to achieve an outcome with properties which are more global than the scene achieved by a single view. Building the appropriate models for multimodal fusion (i.e., models where information consists of various sensing technology), is not trivial. The complexity relies on the technical differences between sensor data.

In recent years, data fusion techniques have been shown to be effective in problems aligned to activity recognition. For example, in previous work, the authors explored the feasibility of recognising walking and standing activities from acceleration and thermal image sensing by fusing the features from both sensing technologies. Results showed an improvement in the classification of about 10% compared to conventional ensembles [11]. Song et al., investigated and conclusively demonstrated an alternative method to create dense embedding for data using kernel similarities and adopting NN architectures [30]. Oswald Ludwig et al. proposed a classifier-fusion schema using learning algorithms, in which they utilised feature extractors and classifier combinations to achieve a higher activity recognition accuracy [12]. In related work, Friday et al. compared the performance of single and multi-sensor fusion for human activity recognition using accelerometer and gyroscope sensors. They considered seven classification algorithms. The evaluation results show the significant impact of multi-sensor fusion for recognising human activities demonstrating the feasibility of data fusion [22].

Another related data fusion approach proposes to combine different kernels to enhance the discrimination power of performance in convolutional NN classifiers [14]. In this context, the idea behind kernel fusion relies

on utilising multiple kernels instead of feature selection in order to create a discriminative matrix of the kernel. Kernel fusion has attracted significant attention in the research community, many approaches have been studied [17] and been verified effective in areas such as Extreme Machine Learning [5], and Multiple Kernel Learning [15]. For example, Liu et al. studied Optimal Neighbourhood Kernel Learning which treats a pre-specified kernel as a "noisy" observation of an optimal kernel and learns the optimal kernel within the neighbourhood by building a constraint within a parametrised model [18]. Wang et al. investigated the benefit of kernel fusion for NN; they presented a practical method consisting of building a particular form of feature-level fusion derived from combining two or more kernels [32].

In this regards, different fusion techniques have been explored in different abstraction levels such as data, features, and kernels in convolutional NN; however, activation function fusion remains unexplored. Given the relevance of activation function in NN, in this paper, we focus on investigating the benefit of fusing activation functions as an elegant approach to exploit inferences that could not be feasible from a single activation fusion.

3. Overview of methodology

As illustrated in Figure 2, the proposed method focuses on scenarios in which only a portion of labelled data is available. Collected sensor data is first prepared by synthesising motion-sensing data and adopting an imputation method to address missed data (Figure 2-a). The NN receives the sensor data through the input layer to then distribute it among the different hidden layers (Figure 2-b), which are meant to implement different activation functions (Figure 2-c). A next hidden layer fusion previous activation functions by utilising a concatenation method (Figure 2-d). Finally, the NN provides respective classes to the unlabelled sensing data (Figure 2-e).

Unlike conventional NN, our proposed methodology suggests implementing a collection of hidden layers growing vertically (Figure 2-c). The motivation for this approach is that we can extract the pattern signature from different activation functions. As shown in Figure 2-c and detailed in line 6 of Algorithm 1, the flexibility of this approach is the ability to use an unlimited number of functions. As presented in the pseudo-code of Algorithm 1, we first define the collection of activation functions to fusion (i.e., α) and feed the *inputLayer* with the semi labelled dataset (i.e., X_I). We then create a sequence of *hiddenLayers* (i.e., line 4-6) utilising the previously defined activation function (i.e., $\alpha[]$); note that the characteristic of the *hiddenLayer* is flexible, hence each collection of activation functions

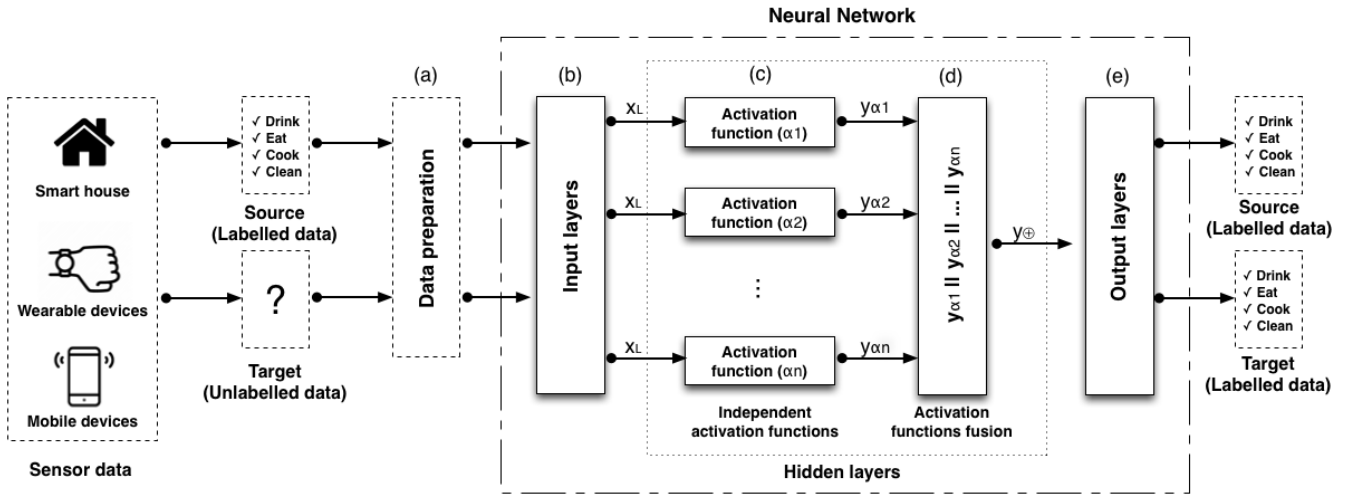


Figure 2. Flow diagram that illustrate the activation function fusion flow process. It begins by retrieving data from the different sensing technology (left side), fusing data collection of steps as it is the core structure of our methodology (centre), and output achieved (left side).

will outcome different results. Finally, we make use of a concatenation function commonly provided by machine learning libraries¹ (i.e., *NN.concatenate*) to fusion the NN's layers and build our activity recognition model (i.e., *NN.buildModel*).

Algorithm 1 Model building algorithm utilising fusion activation functions for activity recognition.

INPUT: X_L
OUTPUT: y_{\oplus}

- 1: Library neural network as NN
- 2: $\alpha[] = \{\text{'relu', 'sigmoid', 'tant', 'elu', ...}\}$
- 3: `inputLayer = NN.inputData(X_L)`
- 4: `hiddenLayer1 = NN.setLayer($\alpha[1]$)`
- 5: `hiddenLayer2 = NN.setLayer($\alpha[2]$)`
- 6: `hiddenLayerN = NN.setLayer($\alpha[n]$)` ▶ This represent the flexibility of the approach by illustrating how other activation functions can be adopted.
- 7: `outputLayer = NN.classify()`
- 8: `fusionLayers = NN.concatenate(hiddenLayer1, hiddenLayer2, hiddenLayer...)`
- 9: `model = NN.buildModel(inputLayer, fusionLayers, outputLayer)`
- 10: $y_{\oplus} = \text{NN.classify}(\text{model})$

Given that each activation function consists of a particular mathematical property that then models the received data, our methodology is designed under

the architectural schema of cooperative and complementary data fusion. The cooperative approach combines the sensing data from two different technology domains, such as wearable devices and kitchen appliances with the benefit of producing a more comprehensive view. On the other hand, the complementary approach enables the representation of different parts of a scene by adopting different activation functions as part of the NN structure and thus obtain global information to achieve high-level inference. Overall, our methodology's contribution relies upon the benefit from the NN's activation function level, where it receives a set of inputs which then are processed in parallel by a defined set of activation functions.

In Figure 3, we observe how a single entry provided thought the Input layer is replicated and simultaneously distributed in n number of Hidden vertical layers; each one implementing a different activation function. The activation function fusion process happens at the second hidden layer, combining the outputs of the n individual activation functions. The Output layer will then analyse the new set of values.

To properly define our methodology, with reference to Figure 3, let α_1 and α_2 be two valid activation functions and X_L the semi-labelled dataset, then α_3 expressed as following is also a valid activation function:

$$\alpha_3 = \alpha_1(X_L) || \alpha_2(X_L) \quad (1)$$

The activation function fusion consists of the concatenation of all outputs of the form $y_{\alpha_1} y_{\alpha_2}$ where y_{α_1} is an outcome from α_1 and y_{α_2} is an outcome from α_2 . For example:

¹www.tensorflow.org

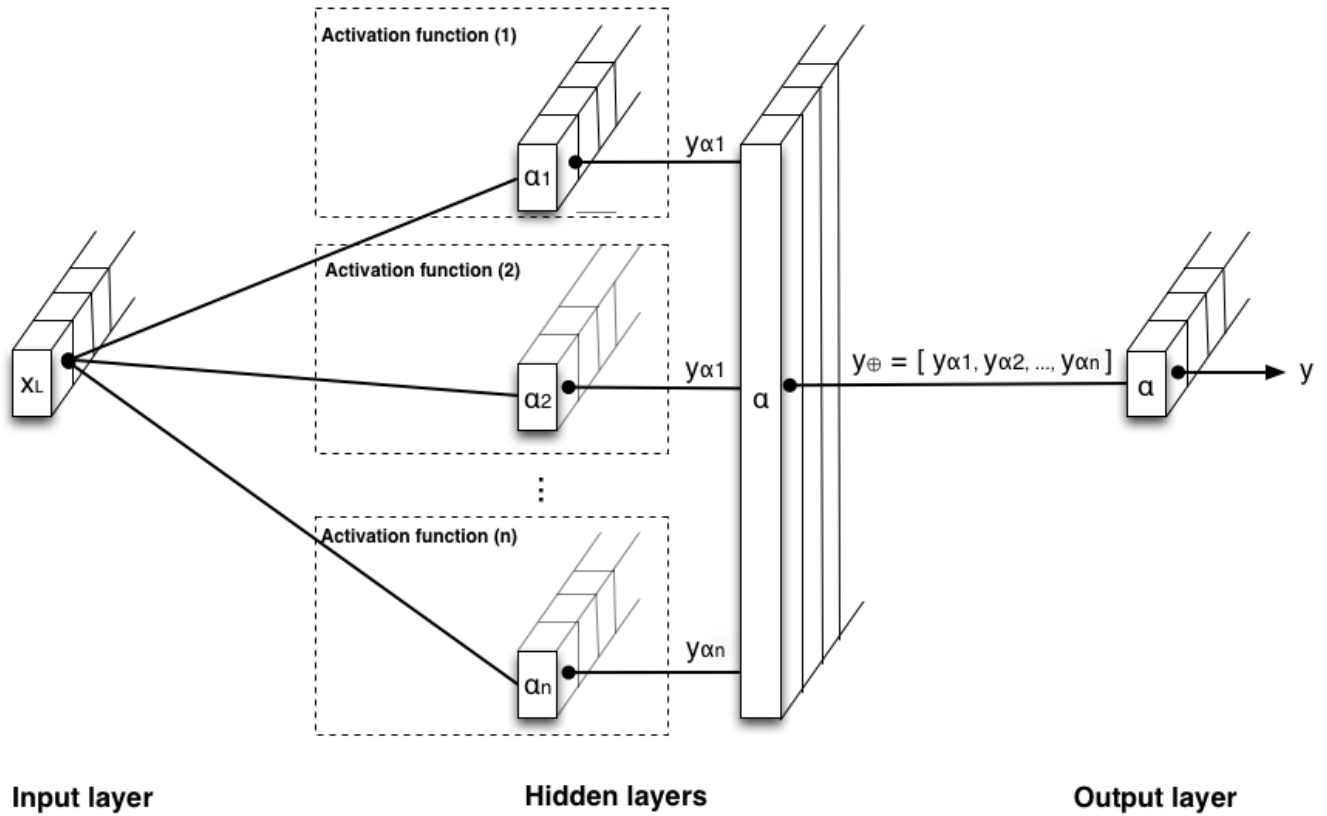


Figure 3. Graphic representation of the activation function fusion proposed in this method. Where X_L represents the semi-labelled data retrieved, α stands for the activation functions, y depicts the activation functions' outcome and y_{\oplus} its concatenation.

$$\begin{aligned}
 y_{\alpha 1} &= \alpha_1(X_L) \\
 y_{\alpha 2} &= \alpha_2(X_L) \\
 y_{\alpha n} &= \alpha_n(X_L)
 \end{aligned}
 \tag{2}$$

Where y_{α} are outcomes from the classification for the activity labels provided in X_L . Hence:

$$y_{\oplus} = \{y_{\alpha 1}y_{\alpha 2} : y_{\alpha 1} \in X_L, y_{\alpha 2} \in X_L\}
 \tag{3}$$

Assuming that there is no restriction between activation functions, this method can assemble a large number of paired activation functions (e.g., tanh & softmax, elu & softplus, relu & linear). In this paper, we have empirically explored the impact of tuple activation function by permuting nine different functions (i.e., tanh, softmax, elu, softplus, softsign, relu, sigmoid, hard_sigmoid, and linear).

Next, we present the evaluation of our approach on one publicly available human behaviour dataset, where we explore the research questions: To what extent does the activation function fusion benefit transfer learning activity recognition models?

3.1. Dataset description

OPPORTUNITY is an open-access collection of sensing data gathered in a realistic environment (i.e., kitchen). It's built upon ubiquitous sensing technologies (i.e., motion data) available in wearable and mobile devices to showcase the capabilities of sensing motion in activity recognition tasks. In this context, we addressed this study to utilise motion technology, given its widely available in daily use technology such as watches and mobile phones; which is of interest to the research community.

We utilised the OPPORTUNITY activity recognition dataset [28], in which data was collected from four subjects performing scripted ADL (Activities of Daily Living) in an adapted home environment that simulates a kitchen. Our particular interest with this dataset is because of the rich overlap of sensing data from heterogeneous sensing devices (i.e., wearable, kitchen appliances, and environmental devices) in which acceleration, inertial measurement unit (IMU), and binary sensors were considered. In this context, each participant completed five repetitions of 17 different activities, such as grooming, relaxing, making/drinking coffee from a cup, preparing/eating a sandwich, and executing a simple-movement drill. The average length

of each activity lasted from 2 to 5 seconds (i.e., 107 ± 44 data points) with a sampling frequency of 30Hz. Overall, the dataset consisted of approximately 6 hours of data recorded. We included data from ambient sensors (e.g., dishwasher doors), kitchen appliances (e.g., cutlery), and on-body sensors which captured the subject's performance from different views simultaneously.

3.2. Data preparation

Given the interest of this study, we have focused on motion sensing from tri-axial sensors as we study the benefits of ADL's recognition in a wearable system [28].

The acceleration signal is synthesised by extracting the output voltage, which is mapped from the ax , ay , and az axes as they are orthogonal to the decomposition of the actions performed. Due to the magnitude of the acceleration with no directional information, the acceleration is orientation independent [5].

Imputation methodology based on the previous sample was implemented for handling missing data in the dataset. Feature data was segmented in window of 500ms with 50% overlap. The selection of features was based on previous studies that have proven them to be effective modelling a NN to classify ADL. Features such as location aren't included since the activities performed were conducted in a single environment (i.e., kitchen).

3.3. Experiment setup

We first evaluated the performance of our NN using the wearable sensing data points from the previously described dataset. We explored the benefit of activation function fusion by permuting the eight activation functions available in the Keras API.

Since NN's activation function computes in a particular manner (i.e., implementing respective neuron algorithms) by finding patterns in a given dataset (i.e., $y = \alpha((w_1x_1, w_2x_2, \dots, w_nx_n) + b)$). In this paper, we leverage the NN by treating the different activation function outcomes (i.e., $y\alpha_n$) as new inputs, therefore, we can compute a set of outcomes from each of the activation functions (i.e., $y \oplus = y\alpha_1 || y\alpha_2 || \dots || y\alpha_n$). In order to showcase our methodology, we conducted our experiments by permuting the different activation function in tuples consisting of two activation functions.

The dataset was pre-processed using Matlab 2018b (as detailed in Section 3.2). The NN is implemented in TensorFlow using Keras to build our models. We have empirically settled our NN with a Rectified linear unit (ReLU) as the activation function of the input layer, and Sigmoid activation function of the output layer. Our NN is trained for 500 epochs under a fully connected stack which implements two hidden layers; one of which

consists of three vertical levels and implements our activation function fusion approach.

In this experiment, we used the leave-one-out-cross-validation method for each user, in which ADL1, ADL2, ADL3, and Drill sessions defined our source dataset, and ADL4, and ADL5 session define our target dataset as advised by the OPPORTUNITY's authors [28]. The process is conducted for each of the four subjects.

4. Results

To establish the feasibility of activation function fusion, we considered the scenario in which the kitchen appliances and wearable sensing devices are available simultaneously and data has been labelled.

To answer our research question (i.e., To what extent does activation function fusion benefit transfer learning activity recognition models?), we randomly trained our activity recognition model with 100, 75, 50, and 25% of labelled data from each view in order to simulate four scenarios in which different levels of fusion data are available (subsequently referred to as fusion data). If the activation function fusion concept proposed as part of our methodology is feasible, we would expect an improvement in the activity recognition model as more fusion data is provided.

To validate the improvement of our methodology, we built a conventional NN implementing a single hidden layer over the ReLU as the activation function, so we can compare the performance of our approach. Results are presented in Table 1.

Subject	Accuracy
S1	0.59455
S2	0.60114
S3	0.31173
S4	0.60025

Table 1. Benchmark results for the activity recognition model utilising a single hidden layer NN and 100% of labelled data.

Given the motivation of this paper which relies on maximizing the potential of the labelled data, next, we highlight results based on the 25% of fusion data.

Tables 2 to 5, show the 10 highest activity recognition model's achieved when using 25% of the fusion data. For example, in Table 2 results show that fusion of "hard sigmoid" and "linear" activation functions have an improvement of 15.17% compared to the subject benchmark accuracy (i.e., 59.45%). Moreover, an increase in the activity recognition of 2.74% is observed when coping with the activity recognition model with 25% (i.e., 71.88% accurate) compared to 100% (i.e., 74.62% accurate) of fusion data.

Similarly, in Table 3, we can observe that the highest achievement concurs with the activation function

Activation functions ($\alpha_1 + \alpha_2$)		Amount of fused data			
		100%	75%	50%	25%
tanh	elu	0.70184	0.71972	0.71021	0.73012
tanh	softplus	0.70919	0.73304	0.71896	0.72416
softmax	relu	0.74128	0.72403	0.75143	0.73557
elu	softplus	0.70450	0.74952	0.73824	0.73278
softplus	relu	0.74344	0.72809	0.73431	0.74635
softplus	linear	0.72086	0.74952	0.73976	0.74115
softsign	relu	0.71579	0.71465	0.72898	0.72822
relu	linear	0.72568	0.74838	0.73989	0.73494
sigmoid	linear	0.71528	0.73811	0.74115	0.73976
hard_sigmoid	linear	0.71883	0.73735	0.75105	0.74623

Table 2. Tuples of the ten highest activity recognition model results for Subject 1, which benchmark accuracy is 0.59455.

tuple of Subject 1. For Subject 2 the benchmark is 60.14%, results for "hard sigmoid" and "linear" activation functions is 74.94%, representing a 14.83% of improvement. Moreover, a positive increase in accuracy of 3.72% is also observed when using 25% of data compared to 100% of fusion data.

As presented in Table 4, for Subject 3 the improvement achieved by our approach rises the activity recognition model by 24.13% when adopting "sigmoid" and "hard_sigmoid" as a activation function tuple. This result represents an improvement of 24.14% when comparing against the subject benchmark (i.e., 31.27%) and using only the 25% of fusion data.

As presented in Table 5, for Subject 4 the improvement achieved is 0.9% when comparing the benchmark (i.e., 60%) against the model's outcome of fusing the "softsign" and "sigmoid" activation function tuple.

As it can be observed across all tables, a similar accuracy level is achieved regardless of the dataset used, which represents a positive benefit for recognising activities in scenarios with constrained access to data. Full results are presented in Appendix A.

5. Discussion and Conclusion

In view of the ubiquitous technology available when undertaking activities of daily living, the underlying activity recognition techniques need to leverage the variability of these technology resources to achieve the highest accuracy of activity recognition, especially when data is partially available. In this paper, we have proposed a novel approach to maximise the accuracy of activity recognition given access to partially labelled, heterogeneous sensor data.

The objective of our methodology was to maximise the accuracy of human activity classification where limited labelled data are available. Unlike other similar studies that explore the feasibility of fusing sensor data, we proposed an elegant approach in which a NN

processes data implementing a collection of different activation functions as part of its hidden layers. The hidden layer is built dynamically, which makes this approach flexible to adopt as many activation functions as wanted. To showcase the feasibility of this study, however, we limited the number to two activation functions.

Since a NN's activation function computes in a particular manner by finding patterns in a given dataset, in this paper, we leverage the NN by treating the different activation function outcomes (i.e., $y\alpha_n$) as new inputs. In this way, we can compute a concatenated set of outcomes from each of the activation functions (i.e., $y\oplus = y\alpha_1 || y\alpha_2 || \dots || y\alpha_n$).

Our results have demonstrated that the fusion of activation functions can perform well and is feasible for human activity recognition utilising inertial sensor devices worn by a user or embedded in kitchen appliances.

We evaluated our methodology using OPPORTUNITY, which is an open-access dataset in which subjects perform ADL while wearing sensor devices and interacting with smart kitchen appliances. The empirical study shows an improvement in activity recognition accuracy, compared to traditional approach across four different subjects. Note that there is notable variability in the results of Subjects 3, which as reported by other authors [7] is due to missing data.

As part of our analysis, in Tables 2 to 5, it was observed that the highest achieved accuracy when using only 25% of activation function fusion data outperformed results compared to when using 100% of labelled data in a conventional shallow network (i.e., increase in accuracy of 2.7%, 3.7%, 4.8%, and 0.9% across the activity recognition of four subjects). We hypothesise that, given the benefit of extracting the pattern from a different perspective (i.e., activation functions) tuples of data becomes more specific as more

Activation functions ($\alpha_1 + \alpha_2$)		Amount of fused data			
		100%	75%	50%	25%
tanh	relu	0.72048	0.71592	0.72632	0.73811
tanh	linear	0.71592	0.72036	0.74115	0.74699
softmax	softplus	0.70501	0.73900	0.71845	0.73177
softmax	relu	0.74052	0.71211	0.74509	0.74572
elu	softplus	0.72251	0.72340	0.71807	0.73697
elu	linear	0.72936	0.74077	0.72682	0.73989
softplus	linear	0.73748	0.75117	0.72099	0.73367
relu	hard_sigmoid	0.71224	0.71490	0.71148	0.73202
sigmoid	linear	0.72162	0.74153	0.73101	0.74217
hard_sigmoid	linear	0.71224	0.72809	0.74750	0.74940

Table 3. Tuple of the ten highest activity recognition model results for Subject 2, which benchmark is 0.60114.

Activation functions ($\alpha_1 + \alpha_2$)		Amount of fused data			
		100%	75%	50%	25%
tanh	softmax	0.48548	0.53393	0.50438	0.49753
tanh	softsign	0.11731	0.52454	0.49829	0.52074
tanh	sigmoid	0.50907	0.44236	0.54217	0.51541
elu	softsign	0.30273	0.31198	0.55764	0.52632
elu	linear	0.43120	0.45529	0.43348	0.48332
softplus	sigmoid	0.50881	0.47229	0.29030	0.54179
softplus	linear	0.42498	0.36271	0.47736	0.50032
softsign	hard_sigmoid	0.48104	0.47102	0.48434	0.48434
relu	sigmoid	0.48129	0.44857	0.41978	0.48878
sigmoid	hard_sigmoid	0.50425	0.52403	0.51427	0.55308

Table 4. Tuple of the ten highest activity recognition model results for Subject 3, which benchmark is 0.31173.

Activation functions ($\alpha_1 + \alpha_2$)		Amount of fused data			
		100%	75%	50%	25%
tanh	elu	0.60355	0.60317	0.60266	0.60736
tanh	softplus	0.59632	0.61205	0.60355	0.60583
tanh	softsign	0.59975	0.60583	0.60342	0.60685
tanh	relu	0.59797	0.59962	0.60774	0.60723
tanh	sigmoid	0.60634	0.59949	0.60089	0.60913
tanh	linear	0.59987	0.60393	0.59391	0.60621
softmax	hard_sigmoid	0.59797	0.60926	0.59290	0.60495
elu	linear	0.59252	0.59924	0.60355	0.60786
softplus	softsign	0.60254	0.60533	0.59759	0.60609
softsign	sigmoid	0.60634	0.60659	0.60114	0.60380

Table 5. Tuple of the ten highest activity recognition model results for Subject 4, which benchmark is 0.60025.

data is adopted. Thus, 25% of data benefits the models by building a more general model.

Given the positive results and the rich opportunities that this approach opens, in the future, we will

investigate the extent to which activation function fusion can be extended to three, four or more functions in the tuple, as well as the characteristics of the data which can affect performance of the model.

Similarly, we will extend our investigation to consider data collected in naturalistic conditions, in which unpredictable human behaviour presents a paramount challenge in the field.

References

- [1] G. Abebe and A. Cavallaro. 2017. Inertial-Vision: Cross-Domain Knowledge Transfer for Wearable Sensors. In *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*. 1392–1400. <https://doi.org/10.1109/ICCVW.2017.165>
- [2] Mohammad Arif Ul Alam and Nirmalya Roy. 2017. Unseen Activity Recognitions: A Hierarchical Active Transfer Learning Approach. In *Proceedings - International Conference on Distributed Computing Systems*. <https://doi.org/10.1109/ICDCS.2017.264>
- [3] Ling Bao and Stephen S. Intille. 2004. Activity Recognition from User-Annotated Acceleration Data. https://doi.org/10.1007/978-3-540-24646-6_1
- [4] Andreas Bulling, Ulf Blanke, and Bernt Schiele. 2014. A tutorial on human activity recognition using body-worn inertial sensors. *Comput. Surveys* (2014). <https://doi.org/10.1145/2499621>
- [5] Wan Yu Deng, Qing Hua Zheng, and Zhong Min Wang. 2014. Cross-person activity recognition using reduced kernel extreme learning machine. *Neural Networks* (2014). <https://doi.org/10.1016/j.neunet.2014.01.008>
- [6] Caiwen Ding, Siyu Liao, Yanzhi Wang, Zhe Li, Ning Liu, Youwei Zhuo, Chao Wang, Xuehai Qian, Yu Bai, Geng Yuan, Xiaolong Ma, Yipeng Zhang, Jian Tang, Qinru Qiu, Xue Lin, and Bo Yuan. 2017. CIRCNN: Accelerating and compressing deep neural networks using block-circulant weight matrices. In *Proceedings of the Annual International Symposium on Microarchitecture, MICRO*. <https://doi.org/10.1145/3123939.3124552> arXiv:1708.08917
- [7] Pouya Foudeh, A. Khorshidtalab, and Naomie Salim. 2017. Testing and analysis of the proposed data driven method on the opportunity human activity dataset. <https://doi.org/10.1145/3018009.3018011>
- [8] Louise Francis. 2014. Unsupervised learning. In *Predictive Modeling Applications in Actuarial Science: Volume I: Predictive Modeling Techniques*. <https://doi.org/10.1017/CB09781139342674.012>
- [9] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT Press. <http://www.deeplearningbook.org>.
- [10] Simon Haykin. 2009. *Neural Networks and Learning Machines 3rd Edition*. <https://doi.org/978-0131471399> arXiv:1312.6199v4
- [11] Netzahualcoyotl Hernandez, Muhammad Asif Razaq, Chris Nugent, Ian McChesney, and Shuai Zhang. 2018. Transfer Learning and Data Fusion Approach to Recognize Activities of Daily Life. In *Proceedings of the 12th EAI International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth '18)*. ACM, New York, NY, USA, 227–231. <https://doi.org/10.1145/3240925.3240949>
- [12] Oswaldo Ludwig Junior, David Delgado, Valter Gonçalves, and Urbano Nunes. 2009. Trainable classifier-fusion schemes: An application to pedestrian detection. In *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*. <https://doi.org/10.1109/ITSC.2009.5309700>
- [13] Bahador Khaleghi, Saiedeh N. Razavi, Alaa Khamis, Fakhreddine O. Karray, and Mohamed Kamel. 2009. Multisensor data fusion: Antecedents and directions. In *3rd International Conference on Signals, Circuits and Systems, SCS 2009*. <https://doi.org/10.1109/ICSCS.2009.5412296>
- [14] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger (Eds.). Curran Associates, Inc., 1097–1105. <http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks>
- [15] Gert Lanckriet, Nello Cristianini, Peter Bartlett, Laurent El Ghaoui, and Michael I Jordan. 2004. Learning the Kernel Matrix with Semi-Definite Programming. *The Journal of Machine Learning Research* (2004).
- [16] Seon Woo Lee and Kenji Mase. 2002. Activity and location recognition using wearable sensors. <https://doi.org/10.1109/MPRV.2002.1037719>
- [17] Xinwang Liu, Lei Wang, Jianping Yin, En Zhu, and Jian Zhang. 2013. An efficient approach to integrating radius information into multiple kernel learning. *IEEE Transactions on Cybernetics* (2013). <https://doi.org/10.1109/TSMCB.2012.2212243>
- [18] Xinwang Liu, Jianping Yin, Lei Wang, Lingqiao Liu, Jun Liu, Chenping Hou, and Jian Zhang. 2013. An adaptive approach to learning optimal neighborhood kernels. *IEEE Transactions on Cybernetics* (2013). <https://doi.org/10.1109/TSMCB.2012.2207889>
- [19] Magdi Mahmoud and Yuanqing Xia. 2014. Multi-Sensor Data Fusion. In *Networked Filtering and Fusion in Wireless Sensor Networks*. <https://doi.org/10.1201/b17667-9>
- [20] H. B. Mitchell. 2012. *Data fusion: Concepts and ideas*. <https://doi.org/10.1007/978-3-642-27222-6>
- [21] Chigozie Nwankpa, Winifred Ijomah, Anthony Gachagan, and Stephen Marshall. 2018. Activation Functions: Comparison of trends in Practice and Research for Deep Learning. (2018), 1–20. arXiv:1811.03378 <http://arxiv.org/abs/1811.03378>
- [22] Henry Friday Nweke, Ying Wah Teh, Uzoma Rita Alo, and Ghulam Mujtaba. 2018. Analysis of Multi-Sensor Fusion for Mobile and Wearable Sensor Based Human Activity Recognition. In *Proceedings of the International Conference on Data Processing and Applications (ICDPA 2018)*. ACM, New York, NY, USA, 22–26. <https://doi.org/10.1145/3224207.3224212>
- [23] S. Oniga and J. Sütő. 2014. Human activity recognition using neural networks. In *Proceedings of the 2014 15th International Carpathian Control Conference (ICCC)*. 403–406. <https://doi.org/10.1109/CarpathianCC.2014.6843636>

- [24] Krzysztof Patan. 2019. Neural Networks. In *Studies in Systems, Decision and Control*. https://doi.org/10.1007/978-3-030-11869-3_2 arXiv:cond-mat/9705270
- [25] Shyamal Patel, Hyung Park, Paolo Bonato, Leighton Chan, and Mary Rodgers. 2012. A review of wearable sensors and systems with application in rehabilitation. <https://doi.org/10.1186/1743-0003-9-21>
- [26] Kevin L. Priddy and Paul E. Keller. 2009. *Artificial Neural Networks: An Introduction*. <https://doi.org/10.1117/3.633187>
- [27] Parisa Rashidi and Diane J. Cook. 2009. Keeping the resident in the loop: Adapting the smart home to the user. *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans* (2009). <https://doi.org/10.1109/TSMCA.2009.2025137>
- [28] Daniel Roggen, Alberto Calatroni, Mirco Rossi, Thomas Holleczeck, Kilian Förster, Gerhard Tröster, Paul Lukowicz, David Bannach, Gerald Pirkl, Alois Ferscha, Jakob Doppler, Clemens Holzmann, Marc Kurz, Gerald Holl, Ricardo Chavarriaga, Hesam Sagha, Hamidreza Bayati, Marco Creatura, and Jose Del R. Millàn. 2010. Collecting complex activity datasets in highly rich networked sensor environments. In *INSS 2010 - 7th International Conference on Networked Sensing Systems*. <https://doi.org/10.1109/INSS.2010.5573462>
- [29] Kaushik Sinha. 2014. Semi-supervised learning. In *Data Classification: Algorithms and Applications*. <https://doi.org/10.1201/b17320>
- [30] Huan Song, Jayaraman J. Thiagarajan, Prasanna Sattigeri, Karthikeyan Natesan Ramamurthy, and Andreas Spanias. 2017. A deep learning approach to multiple kernel fusion. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*. <https://doi.org/10.1109/ICASSP.2017.7952565>
- [31] Maja Stikic, Diane Larlus, and Bernt Schiele. 2009. Multi-graph based semi-supervised learning for activity recognition. In *Proceedings - International Symposium on Wearable Computers, ISWC*. <https://doi.org/10.1109/ISWC.2009.24>
- [32] Zhelong Wang, Donghui Wu, Raffaele Gravina, Giancarlo Fortino, Yongmei Jiang, and Kai Tang. 2017. Kernel fusion based extreme learning machine for cross-location activity recognition. *Information Fusion* (2017). <https://doi.org/10.1016/j.inffus.2017.01.004>
- [33] Dapeng Zhu, Xiaohua Yi, Yang Wang, Kok Meng Lee, and Jiajie Guo. 2010. A mobile sensing system for structural health monitoring: Design and validation. *Smart Materials and Structures* (2010). <https://doi.org/10.1088/0964-1726/19/5/055011>

F. Appendices

Activation functions ($\alpha_1 + \alpha_2$)		Amount of fused data				Difference between 100 and 25%
		100%	75%	50%	25%	
tanh	softmax	0.71756	0.72302	0.70843	0.71122	-0.00634
tanh	elu	0.70184	0.71972	0.71021	0.73012	0.02828
tanh	softplus	0.70919	0.73304	0.71896	0.72416	0.01497
tanh	softsign	0.74978	0.2941	0.71528	0.70666	-0.04312
tanh	relu	0.70146	0.71921	0.72340	0.71389	0.01243
tanh	sigmoid	0.71059	0.71668	0.72378	0.70082	-0.00977
tanh	hard_sigmoid	0.70095	0.71744	0.70564	0.71046	0.00951
tanh	linear	0.72226	0.71668	0.73633	0.71642	-0.00584
softmax	elu	0.72822	0.70818	0.71401	0.70260	-0.02562
softmax	softplus	0.70247	0.71414	0.71046	0.72086	0.01839
softmax	softsign	0.73114	0.72099	0.72479	0.70843	-0.02271
softmax	relu	0.74128	0.72403	0.75143	0.73557	-0.00571
softmax	sigmoid	0.71655	0.73583	0.71224	0.72074	0.00419
softmax	hard_sigmoid	0.69816	0.2941	0.72175	0.71351	0.01535
softmax	linear	0.72353	0.72403	0.72226	0.72251	-0.00102
elu	softplus	0.70450	0.74952	0.73824	0.73278	0.02828
elu	softsign	0.71439	0.72682	0.71604	0.72403	0.00964
elu	relu	0.69271	0.71769	0.73443	0.70970	0.01699
elu	sigmoid	0.71224	0.72416	0.71477	0.70742	-0.00482
elu	hard_sigmoid	0.71211	0.73938	0.71883	0.71097	-0.00114
elu	linear	0.73228	0.71528	0.72416	0.72086	-0.01142
softplus	softsign	0.71186	0.72758	0.74027	0.71173	-0.00013
softplus	relu	0.74344	0.72809	0.73431	0.74635	0.00291
softplus	sigmoid	0.70564	0.71541	0.70805	0.70438	-0.00126
softplus	hard_sigmoid	0.71148	0.73025	0.7215	0.71769	0.00621
softplus	linear	0.72086	0.74952	0.73976	0.74115	0.02029
softsign	relu	0.71579	0.71465	0.72898	0.72822	0.01243
softsign	sigmoid	0.70526	0.72226	0.72619	0.71084	0.00558
softsign	hard_sigmoid	0.70336	0.71566	0.71680	0.71592	0.01256
softsign	linear	0.71972	0.73152	0.74572	0.72086	0.00114
relu	sigmoid	0.71706	0.73088	0.71376	0.70526	-0.0118
relu	hard_sigmoid	0.70628	0.71389	0.72479	0.70399	-0.00229
relu	linear	0.72568	0.74838	0.73989	0.73494	0.00926
sigmoid	hard_sigmoid	0.70729	0.70793	0.71414	0.70120	-0.00609
sigmoid	linear	0.71528	0.73811	0.74115	0.73976	0.02448
hard_sigmoid	linear	0.71883	0.73735	0.75105	0.74623	0.02740

Table 1. Full results of the tuples for the activity recognition models for Subject 1. Complementary information of Table 2.

Activation functions ($\alpha_1 + \alpha_2$)		Amount of fused data				Difference between 100 and 25%
		100%	75%	50%	25%	
tanh	softmax	0.71845	0.71325	0.71275	0.70919	-0.00926
tanh	elu	0.70717	0.71617	0.72010	0.70679	-0.00038
tanh	softplus	0.73418	0.72796	0.75460	0.70615	-0.02803
tanh	softsign	0.71211	0.71668	0.71947	0.69803	-0.01408
tanh	relu	0.72048	0.71592	0.72632	0.73811	0.01763
tanh	sigmoid	0.72784	0.71414	0.71490	0.70577	-0.02207
tanh	hard_sigmoid	0.69081	0.71668	0.71059	0.70958	0.01877
tanh	linear	0.71592	0.72036	0.74115	0.74699	0.03107
softmax	elu	0.72416	0.73989	0.70640	0.71034	-0.01382
softmax	softplus	0.70501	0.73900	0.71845	0.73177	0.02676
softmax	softsign	0.73570	0.72023	0.70159	0.70729	-0.02841
softmax	relu	0.74052	0.71211	0.74509	0.74572	0.0052
softmax	sigmoid	0.73240	0.71744	0.71858	0.71389	-0.01851
softmax	hard_sigmoid	0.70120	0.29410	0.29410	0.70197	0.00077
softmax	linear	0.71782	0.73469	0.72327	0.71909	0.00127
elu	softplus	0.72251	0.72340	0.71807	0.73697	0.01446
elu	softsign	0.71262	0.72175	0.71465	0.73215	0.01953
elu	relu	0.71528	0.71160	0.72454	0.70514	-0.01014
elu	sigmoid	0.72936	0.74724	0.72048	0.73075	0.00139
elu	hard_sigmoid	0.70564	0.72238	0.72150	0.70070	-0.00494
elu	linear	0.72936	0.74077	0.72682	0.73989	0.01053
softplus	softsign	0.71262	0.71110	0.7116	0.71021	-0.00241
softplus	relu	0.72365	0.71896	0.71579	0.72746	0.00381
softplus	sigmoid	0.59074	0.72315	0.73253	0.71110	0.12036
softplus	hard_sigmoid	0.71934	0.72200	0.72264	0.71198	-0.00736
softplus	linear	0.73748	0.75117	0.72099	0.73367	-0.00381
softsign	relu	0.71097	0.74192	0.73088	0.70552	-0.00545
softsign	sigmoid	0.70501	0.71338	0.72911	0.71756	0.01255
softsign	hard_sigmoid	0.72061	0.71934	0.70983	0.71477	-0.00584
softsign	linear	0.73519	0.74914	0.71756	0.72961	-0.00558
relu	sigmoid	0.71237	0.72809	0.71237	0.70120	-0.01117
relu	hard_sigmoid	0.71224	0.71490	0.71148	0.73202	0.01978
relu	linear	0.72150	0.73240	0.72048	0.71642	-0.00508
sigmoid	hard_sigmoid	0.59696	0.72074	0.72327	0.71148	0.11452
sigmoid	linear	0.72162	0.74153	0.73101	0.74217	0.02055
hard_sigmoid	linear	0.71224	0.72809	0.74750	0.74940	0.03716

Table 2. Full results of the tuples for the activity recognition models for Subject 2. Complementary information of Table 3.

Activation functions ($\alpha_1 + \alpha_2$)		Amount of fused data				Difference between 100 and 25%
		100%	75%	50%	25%	
tanh	softmax	0.48548	0.53393	0.50438	0.49753	0.01205
tanh	elu	0.46151	0.55041	0.48244	0.45086	-0.01065
tanh	softplus	0.41509	0.44883	0.45694	0.40964	-0.00545
tanh	softsign	0.11731	0.52454	0.49829	0.52074	0.40343
tanh	relu	0.43855	0.50526	0.39556	0.42663	-0.01192
tanh	sigmoid	0.50907	0.44236	0.54217	0.51541	0.00634
tanh	hard_sigmoid	0.54052	0.47876	0.53304	0.44857	-0.09195
tanh	linear	0.40089	0.45732	0.47432	0.40520	0.00431
softmax	elu	0.39417	0.34103	0.47800	0.42549	0.03132
softmax	softplus	0.39062	0.47914	0.39226	0.41332	0.0227
softmax	softsign	0.50717	0.42942	0.44084	0.43957	-0.0676
softmax	relu	0.38732	0.41002	0.40025	0.47204	0.08472
softmax	sigmoid	0.41382	0.41484	0.45441	0.45986	0.04604
softmax	hard_sigmoid	0.44223	0.41218	0.38288	0.32796	-0.11427
softmax	linear	0.54990	0.36880	0.46265	0.42727	-0.12263
elu	softplus	0.57299	0.40736	0.50146	0.42777	-0.14522
elu	softsign	0.30273	0.31198	0.55764	0.52632	0.22359
elu	relu	0.42980	0.50704	0.48205	0.38503	-0.04477
elu	sigmoid	0.40977	0.53114	0.47381	0.30932	-0.10045
elu	hard_sigmoid	0.52188	0.45529	0.45694	0.34547	-0.17641
elu	linear	0.43120	0.45529	0.43348	0.48332	0.05212
softplus	softsign	0.39429	0.49918	0.47242	0.46252	0.06823
softplus	relu	0.40190	0.42105	0.42346	0.35549	-0.04641
softplus	sigmoid	0.50881	0.47229	0.2903	0.54179	0.03298
softplus	hard_sigmoid	0.39632	0.49017	0.52619	0.46963	0.07331
softplus	linear	0.42498	0.36271	0.47736	0.50032	0.07534
softsign	relu	0.52505	0.48903	0.41281	0.40406	-0.12099
softsign	sigmoid	0.51592	0.46049	0.46367	0.46100	-0.05492
softsign	hard_sigmoid	0.48104	0.47102	0.48434	0.48434	0.0033
softsign	linear	0.47229	0.49461	0.42676	0.46569	-0.0066
relu	sigmoid	0.48129	0.44857	0.41978	0.48878	0.00749
relu	hard_sigmoid	0.49093	0.46075	0.44249	0.47229	-0.01864
relu	linear	0.45580	0.48725	0.46164	0.43640	-0.0194
sigmoid	hard_sigmoid	0.50425	0.52403	0.51427	0.55308	0.04883
sigmoid	linear	0.48561	0.45225	0.36576	0.36766	-0.11795
hard_sigmoid	linear	0.49245	0.35675	0.44667	0.42207	-0.07038

Table 3. Full results of the tuples for the activity recognition models for Subject 3. Complementary information of Table 4.

Activation functions ($\alpha_1 + \alpha_2$)		Amount of fused data				Difference between 100 and 25%
		100%	75%	50%	25%	
tanh	softmax	0.60013	0.60812	0.60545	0.60355	0.00342
tanh	elu	0.60355	0.60317	0.60266	0.60736	0.00381
tanh	softplus	0.59632	0.61205	0.60355	0.60583	0.00951
tanh	softsign	0.59975	0.60583	0.60342	0.60685	0.0071
tanh	relu	0.59797	0.59962	0.60774	0.60723	0.00926
tanh	sigmoid	0.60634	0.59949	0.60089	0.60913	0.00279
tanh	hard_sigmoid	0.60254	0.61002	0.61129	0.58998	-0.01256
tanh	linear	0.59987	0.60393	0.59391	0.60621	0.00634
softmax	elu	0.60545	0.59746	0.60444	0.60165	-0.0038
softmax	softplus	0.60647	0.60342	0.60241	0.60254	-0.00393
softmax	softsign	0.60380	0.60292	0.60241	0.60152	-0.00228
softmax	relu	0.60025	0.60025	0.59493	0.59772	-0.00253
softmax	sigmoid	0.60672	0.60571	0.59569	0.60292	-0.0038
softmax	hard_sigmoid	0.59797	0.60926	0.59290	0.60495	0.00698
softmax	linear	0.60051	0.59899	0.59797	0.60216	0.00165
elu	softplus	0.60013	0.60254	0.5986	0.60203	0.0019
elu	softsign	0.60837	0.60393	0.60634	0.60279	-0.00558
elu	relu	0.60203	0.60482	0.60355	0.60063	-0.0014
elu	sigmoid	0.60621	0.59899	0.59949	0.60368	-0.00253
elu	hard_sigmoid	0.60495	0.61091	0.60583	0.60266	-0.00229
elu	linear	0.59252	0.59924	0.60355	0.60786	0.01534
softplus	softsign	0.60254	0.60533	0.59759	0.60609	0.00355
softplus	relu	0.59949	0.60241	0.60089	0.60127	0.00178
softplus	sigmoid	0.60190	0.60317	0.60406	0.60304	0.00114
softplus	hard_sigmoid	0.60254	0.60101	0.60431	0.60216	-0.00038
softplus	linear	0.59569	0.60406	0.59290	0.59696	0.00127
softsign	relu	0.60659	0.60431	0.60888	0.60355	-0.00304
softsign	sigmoid	0.60634	0.60659	0.60114	0.60380	-0.00254
softsign	hard_sigmoid	0.60520	0.59784	0.60330	0.59467	-0.01053
softsign	linear	0.60038	0.6038	0.59962	0.59759	-0.00279
relu	sigmoid	0.60419	0.60837	0.60558	0.60038	-0.00381
relu	hard_sigmoid	0.60254	0.60457	0.60127	0.60076	-0.00178
relu	linear	0.59759	0.59391	0.60038	0.60025	0.00266
sigmoid	hard_sigmoid	0.60507	0.60710	0.60216	0.59911	-0.00596
sigmoid	linear	0.60076	0.60178	0.60596	0.59341	-0.00735
hard_sigmoid	linear	0.60368	0.60507	0.59822	0.60317	-0.00051

Table 4. Full results of the tuples for the activity recognition models for Subject 4. Complementary information of Table 5.