A Deep Learning Assisted Method for Measuring Uncertainty in Activity Recognition with Wearable Sensors

Ali Akbari[a], Roozbeh Jafari[a,b,c]

[a] Department of Biomedical Engineering, Texas A&M University, College Station, TX, USA
[b] Department of Computer Science and Engineering, Texas A&M University, College Station, TX, USA
[c] Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX, USA
{aliakbari, rjafari}@tamu.edu

***Abstract*—For human activity recognition with wearable sensors, understanding the uncertainty in the classifier decision is necessary to predict sensor failures and design active learning paradigms. Although deep learning models have shown promising results in recognizing human activities from sensor data, it is still challenging to estimate their uncertainty in producing decisions. In this paper, we propose a Bayesian deep convolutional neural network with stochastic latent variables that allows us to estimate both aleatoric (data dependent) and epistemic (model dependent) uncertainties in recognition task. We put a distribution over the latent variables of the model, which are the features that are automatically extracted by the convolutional layers, and show how the inference can be approximated by combining a variational autoencoder with a typical deep neural network classifier. We also leverage Dropout Bayesian neural network to approximate the model uncertainty. The experimental results on a publicly available dataset of human activity recognition with wearable sensors show how each uncertainty (*i.e.*, aleatoric and epistemic) measure is sensitive against different sources of uncertainty namely noisy as well as novel data. Moreover, the uncertainty for the samples that are misclassified by the model is significantly higher on average than the samples that are correctly classified.**

Keywords—activity recognition, uncertainty estimation, active learning, deep learning, wearable sensor

I. INTRODUCTION

Development of wearable technology has provided a great opportunity for understanding human activities of daily living (ADL) by means of various devices such as smartwatches and smartphones, which provides vital information about people and important contextual insight that can enhance the effectiveness of healthcare. Data gathered by wearable sensors could be analyzed by rigorous machine learning models to recognize ADLs [1]. However, understanding what a model does not know or when it is not confident about its decision, known as the confidence or uncertainty of the system, is critical in many cases.

In ADL recognition using wearable sensors, information about the confidence/uncertainty of the recognition system is essential to design active learning paradigms that can adaptively modify themselves to learn complex activities on-the-fly through asking for user's feedback. In this paradigm, the system needs to recognize the data points for which it cannot produce decisions with high confidence to minimize the interaction and the burden on the user. Moreover, wearables suffer from various sources of disturbance such as sensor misplacement and sensor noise, which raises the need for robust algorithms who can detect if they are encountering noisy measurements. This feature is also helpful for data fusion, ensemble classification, as well as novelty detection paradigms for ADL recognition.

In general, there are two types of uncertainties surrounding wearable recognition systems. One is called aleatoric uncertainty, which is data dependent and is related to any noise present in sensor measurements such as electrical or thermal noise, sensor displacement or misplacement, and sensor movements with respect to the body. Such uncertainty cannot be mitigated by increasing the training data. The second type is the epistemic uncertainty which is related to the inability of the model to recognize certain samples due to the lack of sufficient training data. Additionally, this uncertainty increases when the recognition system sees new data that does not know how to deal with. This uncertainty can be mitigated by increasing training data. The epistemic uncertainty is more important for active learning tasks because it can trigger learning and relearning processes while the aleatoric is necessary for detecting sensor malfunctioning scenarios. Although, there are extensive prior studies in ADL recognition that have created powerful machine learning models, there is still a gap in quantifying and distinguishing these uncertainties in the proposed models.

Deep neural networks have contributed to tremendous advances in ADL recognition with wearable sensors as they are able to learn powerful representations from high dimensional data and map them to desired outputs [2]. However, these mappings are often taken blindly and assumed to be accurate, which is not always the case [3]. For ADL recognition tasks, the existing practice has been to apply a Softmax function to the output of the network to create a probability distribution over labels. The output of the Softmax function is often interpreted as an estimate of the true distribution over labels given the input data of the sensors. However, recent works have empirically shown that the Softmax function is often ineffective in producing accurate uncertainty estimations [2,4].

To address this problem, we propose a unified Bayesian deep learning framework for ADL recognition to model the aforementioned uncertainties by considering stochasticity on both the parameters of the neural network and the latent

978-1-7281-0848-3/19/$31.00 ©2019 IEEE

variables served as the features. Our proposed method extracts the features from time series automatically and learns their posterior distribution given the input data through a variational autoencoder (VAE) based framework. To consider the randomness on the model weights, we utilize the Dropout Bayesian network.

In summary the contributions of this paper are as follows:

- We design a unified framework for automatic feature extraction, classification, and estimation of uncertainty of the classifier for human activity recognition.
- We propose a method for quantifying both epistemic and aleatoric uncertainties, and show how each responds to different sources of uncertainty through our experiments.

II. Related Works

Understanding what a model does not know is a critical part of machine learning and deep learning models. In deep learning, the Softmax functions is used at the end of the pipeline to measure probability of the labels; however, it has been shown that it does not always capture model uncertainty correctly [2,4]. A framework has been designed to learn mappings from input data to aleatoric uncertainty in image recognition [4]. The authors compose these together with epistemic uncertainty approximations. However, this model learns aleatoric uncertainty based on the assumption that noisy data is available in training images (*e.g.*, highly textured input images or far objects), which is not always the case in ADL recognition.

Measuring uncertainty is important in active learning tasks for activity recognition. Uncertainty has been used to trigger data annotation inquiry from users to update the classifier [5,6]. In addition, understanding uncertainty due to sensor malfunction is critical when designing data fusion frameworks for robust activity recognition [7,8]. The uncertainty of the classifier has also been used to detect samples of NULL activities [9]. It is essential to decompose and accurately estimate each uncertainty for design of active learning paradigms. Because in one scenario, active learning can be triggered, and in the other scenario, in presence of noisy or malfunctioning sensors, the corresponding data points could be discarded. To the best of our knowledge, there is no study that distinguishes and measures different types of uncertainty in activity recognition.

III. Uncertainty Definition and the Limitaion of Softmax Functions

In this section, we formally define the concept of uncertainty in recognition task and then highlight the limitation of the Softmax function in estimating uncertainty. The concept of recognition uncertainty is coupled with the confidence of the classification model about its decision. More formally, let (x,y) represent a single sample from training data where input data is D-dimensional $x \in R^D$ and the output is N-dimensional $y \in R^N$ corresponding to N different classes. The confidence of the classifier is defined as $p(y_i \mid x)$ where $i \in [1,N]$. To estimate $p(y_i \mid x)$ in neural networks, Softmax activation function is utilized in the last layer while the number of neurons is equal to the number of classes. It has been shown that irrespective of the structure of the data, an input x with a large $l2$ norm always produces higher confidence than that with a lower $l2$ norm [4]. In other words, one can increase the confidence by simply increasing the amplitude of a signal as shown in Figure 1. Based on Figure 1, the signal with doubled amplitude leads to a higher confidence. This means that the Softmax is not a good estimate for confidence of the classifier since increasing the amplitude does not increase the information content and yet Softmax reports additional confidence.

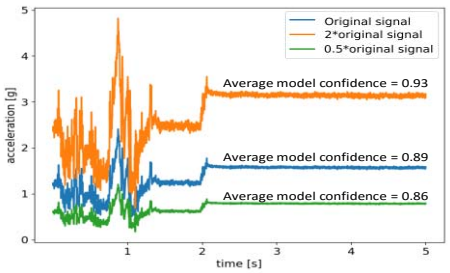

Figure 1. An example that shows increasing the magnitude of the signal increases confidence of Softmax

IV. Methods

In this section, we propose a unified framework for ADL recognition and estimating different types of uncertainties within a deep neural network model.

A. CNN for automated feature extraction and classification

We leverage a CNN for activity recognition with wearable sensors due to the ability of CNN to extract features automatically. The network receives raw signal x as input and maps it to a latent variable z. These latent variables serve as the features that are extracted from a raw input signal. In a typical neural network f, we can write $f=h \circ g$ where $g: x \rightarrow z$ maps raw inputs to a higher level feature space z and $h: z \rightarrow y$ is a discriminative function that maps the features to desired class labels. A deep CNN with multiple layers can extract features from the input signal by leveraging a convolution operation of the signal with a kernel. A kernel can be interpreted as a filter applied to the signal via a convolution operation that can extract a specific pattern or feature anywhere within the signal (*i.e.*, translation invariant). The trainable weights of CNN kernels (W_g) and the classifier (W_h) are learnt through the training of the neural network.

B. Uncertainty estimation

To take into account the epistemic uncertainty, which reflects the model uncertainty, we assume a distribution over the weights of the discriminative function (function h). However, to capture the aleatoric, which is a data-dependent uncertainty, we put a distribution over the latent variable z. By assuming such distributions over the weights and latent variables and treating them as random variables instead of deterministic values, the final label inference can be written as Equation 1.

$$p(y_i|x) = \int p(y_i|W_h, x, z)p(W_h, z|x)dzdW_h \quad (1)$$

where $p(y_i|W_h, x, z)$ is the likelihood function, and it is calculated as the output of the neural network. $p(W_h, z|x)$ is the posterior of weights and latent features given the input data. Since z is independent of W_h and also W_h is independent of input data x, Equation 1 can be written as follows:

$$p(y_i|x) = \int p(y_i|W_h, x, z)p(W_h|D)p(z|x)dzdW_h \quad (2)$$

where D is the whole training dataset used to estimate the W_h. The integral in Equation 2 can be approximated through Monte Carlo estimation as follows:

$$p(y_i|x) = \frac{1}{n}\sum p\left(y_i|\widehat{W_h}, x, \hat{z}\right) \quad \widehat{W_h}\sim p(W_h) \quad \hat{z}\sim p(z|x) \tag{3}$$

In this framework, $p(z|x)$ is used to model the aleatoric uncertainty. $p(W_h)$, on the other hand, is used to model the epistemic uncertainty. Here the problem is how to approximate $p(W_h)$ and $p(z|x)$. Considering the distribution over weights W_h, Dropout variational inference is a practical approach for approximation inference in deep learning models [1]. This inference is done by training a model with dropout after every layer and also applying dropout in testing phase to sample from the approximate posterior. Each sample is passed through the network n times and in every pass, the weights of the network are dropped randomly with the probability of p_{drop}.

To complete the calculation in Equation 3, we still need to estimate the posterior distribution of latent variables $p(z|x)$. Please note that z is an unobserved latent variable while our observation is x. One can leverage Bayesian analysis to calculate $p(z|x)$ directly as $p(z|x) = \frac{p(x|z)p(z)}{p(x)}$; however, this leads to an intractable integral for calculating the denominator. To solve this issue, we approximate $p(z|x)$ with a variational distribution $q(z|x)$ from a Gaussian family. By defining the parameters of $q(z|x)$ such that it is similar to $p(z|x)$, we can use it to perform inference approximation in Equation 3. To achieve this, we could minimize the Kullback-Leibler divergence (D_{KL}) between the two distributions, which leads to Equation 4 [10].

$$\log p(x) - D_{KL}\{q(z|x)||p(z|x)\} = E_{z_i\sim q}[\log p(x|z)] - D_{kl}\{q(z|x)||p(z)] \tag{4}$$

Based on Equation 4 to minimize the KL divergence between *q(z|x)* and *p(z|x)*, we can minimize the right-hand side of this equation. This is the core objective function of a variational autoencoder [9]. The first term on RHS of Equation 4 is the loss of reconstructing input x from the latent variable z and the second term is the divergence between the variational approximation with the prior distribution of z for which we use a standard Gaussian with mean of zero and variance of one.

By training a VAE and using the latent variable z as the features that are fed to the classifier, we can approximate the posterior distribution of features given input data and then calculate the inference in Equation 3. A VAE with convolutional layers in the encoder can extract informative features from a signal in an unsupervised manner. In such a framework, the only concern is to extract features that can retain the structure of the input data to reconstruct it successfully. However, in this paper we are dealing with a supervised classification problem in which the labels are given for certain samples during the training phase. Thus, the features not only should be able to reconstruct the input data but they should also be discriminative enough regarding the classification task given the labels. Based on this intuition, we propose a new architecture of deep neural network as shown in Figure 2 by modifying the typical VAE objective function as follows:

$$L = p(y|x) + E_{z\sim q}[\log p(x|z)] - D_{kl}\{q(z|x)||p(z)] \tag{5}$$

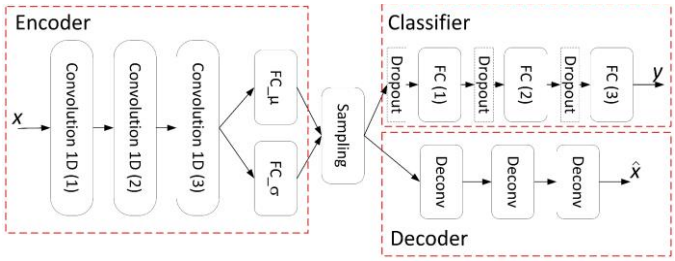

Figure 2. The architecture of the proposed neural network for feature extraction and uncertainty estimation

We add the *p(y|x)* to the typical VAE loss in Equation 4 to produce the new loss function, as shown in Equation 5. This loss guides the VAE to produce latent features that not only can reproduce the input data, but are also discriminated between different class labels. In Figure 2, the encoder, which serves as feature extractor, estimates the mean and standard deviation of a Gaussian distribution that is the approximation of the posterior of the features given data, which is required for calculating Equation 3. The classifier samples from the distribution approximated by the encoder, and maps them to the class labels.

After training the system in Figure 2, the procedure for estimating the label and uncertainty of the classifier is shown in Algorithm 1 where *OneHotEncoding*(.) is a function that returns the one hot encoding of a vector and *std*(.) calculates standard deviation. Every input is passed through the model n times and at each pass *(i)* the weights are dropped randomly and *(ii)* the latent feature is sampled from the distribution approximated by the encoder in VAE. The output of the network is calculated for all n samples, the average of them is taken as the model decision, and the standard deviation is considered as the estimation of uncertainty. Intuitively, for the samples that the classifier is confident about, the generated labels would be more consistent, while for non-confident samples, the classifier will generate distinct labels that leads to higher standard deviations.

The uncertainty calculated by Algorithm 1 is called combined uncertainty as it contains both aleatoric and epistemic uncertainties. To only consider the epistemic uncertainty, we use the mean of z instead of sampling from the approximated posterior. In fact, with this uncertainty, we only consider the randomness of the weights of the model and do not care about the randomness of z. On the other hand, to only calculate the aleatoric uncertainty, we do not drop weights during testing phase. Hence, we ignore model uncertainty and only take into account the data dependent uncertainty over latent variables z.

Algorithm 1 Label and uncertainty estimation

Input: test data x, encoder network g, classifier network h, p_{drop}, number of labels N, parameter n
Initialize *prediction = zeros(n,N)*
Output: classifier decision $\hat{y}$, *Uncertainty*
for *j = 1* **to** *n* **do:**
  Take a sample $z_j \sim g(x)$
  Drop weights of h with the probability of p_{drop}
  $y^j = h(z_j)$ // the output of Softmax function
  prediction [j,:] = OneHotEncoding(y^j)
end for
$\hat{y} = \frac{1}{n}\sum_{j=1}^{n} y^j$
Uncertainty = std(prediction)

In this work, the encoder consists of three convolutional layers with 32, 64, and 128 neurons respectively following by two fully connected (FC) layers with 20 neurons for estimating mean and standard deviation of posterior (FC_μ and FC_σ in Figure 2). The classifier contains three fully connected layers with 64, 128, and 7 neurons. It is worth mentioning that based on our experiments, leveraging less number of layers, below three for each network, did not produce a reasonable accuracy. On the other hand, increasing the number of layers beyond three increases complexity of the model and introduces challenges for the execution of the neural network on wearable devices, while it does not offer significant improvement in the performance.

V. Experimental Results

We analyze the effectiveness of our uncertainty modeling by evaluating its behavior in response to different sources of uncertainty. When presented with novel data, such as those from new classes or new sensors, a good measure of uncertainty should increase. Furthermore, the same should happen when the system deals with noisy measurements. To assess the quality of our uncertainty measurement technique, we investigate the following questions:

- How do different uncertainties quantified in this study (epistemic and aleatoric) behave when dealing with novel data from a different distribution compared to the training set?
- How do noisy sensor measurements affect these uncertainties? With last two questions, we seek to find whether these two types of uncertainties are separable.
- Is there a meaningful change in the uncertainties in data samples that are predicted wrong by the model compared to the ones that are correctly classified?

A. Dataset

The dataset used in this work is a publicly available activity recognition dataset called PAMAP2 which comprises of 18 physical activities measured by three inertial measurement units (IMUs) performed by 9 different subjects. IMUs were worn on three different body parts: wrist of the dominant hand, chest, and ankle of the dominant foot. In this study, we used 7 out of 18 activities, which have most number of samples namely lying down, sitting, standing, walking, running, biking, and rope jumping. We used 3D acceleration and gyroscope sensors that results in 18 axis of data and segmented the data into windows of length 100 (one second as the sampling rate of the sensors is 100Hz) with 50% overlap.

B. Uncertainty Created by Novel Data

In this section, we aim to analyze the behavior of the quantified uncertainties when the system deals with novel data, namely data that is drawn from a different distribution. A good measure of uncertainty must increase when the system is presented with novel unfamiliar data. In activity recognition, the novel data is presented when the user performs an activity for which the system has not been trained, or when the data of a new sensor is fed to the system. Therefore, herein we consider three different scenarios where the results are shown in Figure 3.

In the first scenario (the yellow bar in Figure 3), we test the model with data from the same distribution as the training (non-novel data). In other words, the model is tested on the data of the same sensor and with the same labels as in the training set. In the second scenario (the cyan bar) the model is trained on two sensors and is tested upon the data of the third (new) sensor. In the last scenario (the blue bar), we test the model with the data of new activities while the sensors are the same in the training and testing phase. Unsurprisingly, as Figure 3 shows, all combined (Figure 3-a), epistemic (Figure 3-b) and aleatoric (Figure 3-c) uncertainties increase when the model faces novel data (cyan and blue bars in comparison to the yellow bar). However, this increase is much more significant in the epistemic uncertainty (Figure 3-b) compared to the aleatoric uncertainty (Figure 3-c). This confirms the initial hypothesis about the type of uncertainties and it is in line with [3]. In fact, it shows that the epistemic uncertainty, which considers the model uncertainty, is much more sensitive to the lack of training data compared to the aleatoric uncertainty. By measuring this uncertainty, we can realize if the input data is not familiar for the model. In such case, for example, we can solicit the user to get more information about the activity label and retrain the model.

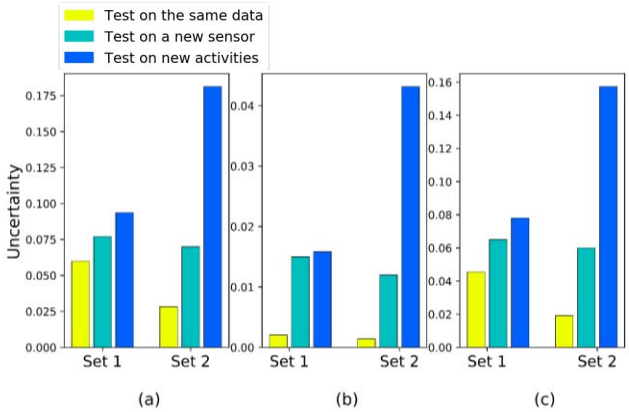

Figure 3 - The uncertainty when the model is tested on novel data. in "Set 1" the training data includes lying, sitting, and standing and the novel data in the test phase includes walking, running, cycling, and rope jumping. in "Set 2" the training and testing activities are exchanged . (a) combined uncertainty; (b) epistemic uncertainty; (c) aleatoric uncertainty

C. Uncertainty Created by Noisy data

In general, we expect the uncertainty to increase in response to the noisy sensor measurement. Thus, the second experiment is devoted to the analysis of the effect of sensor noise on each type of uncertainties quantified in this study as shown in Figure 4. In fact, here we seek to understand whether the uncertainty sourced by the noisy data is distinguishable from the uncertainty of the model sourced by lack of sufficient training data. In order to model the data uncertainty, we add synthetic noise to data where varying level of noise is represented in the horizontal axis of Figure 4. Each axis of raw sensor data is corrupted with a Gaussian noise which has a standard deviation proportional to the absolute value of its average using Equation 6.

$$\tilde{X} = X + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \lambda^2 \Sigma) \tag{6}$$

where Σ is a diagonal matrix containing the absolute value of the mean of each axis and λ is a constant. The value for λ was varied from 0 to 0.5 to model different noise levels, as demonstrated on the X axis of Figure 4. According to the Figure 4, the aleotaric uncertainty (solid red line), which corresponds to the uncertainty caused by noise in the data, increases consistently as the noise magnitude (and power) increases. Epistemic uncertainty (blue

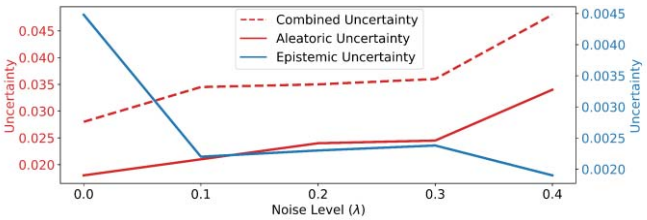

Figure 4 - The uncertainty of the model in presence of different noise levels. X axis corresponds to values for λ in Equation 6. (a) aleatoric uncertainty; (b) epistemic uncertainty; (c) combined uncertainty

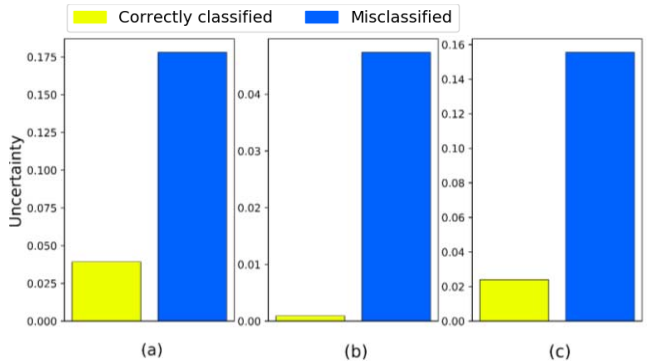

Figure 5- Comparing the uncertainty for the samples that are correctly classified by the model and the samples that are misclassified (a) combined uncertainty; (b) epistemic uncertainty; (c) aleatoric uncertainty

line in Figure 4), contrarily, does not have an increasing pattern as consistent as that of the aleotaric uncertainty, which indicates that the epistemic uncertainty is irrelevant to the uncertainty created by noisy data. Combined uncertainty (dashed red line in Figure 4), similar to aleotaric uncertainty, experiences a steady increase which is desired since the noisy data makes predictions harder. This shows that the uncertainty increases when the system deals with noisy sensor measurements while it can still capture the fact that this uncertainty is not caused by the modeling but by noisy data. This can help the system to identify noisy measurements or sensor misplacements for instance and distinguish them from uncertainty due to lack of training data.

D. Correct vs missclassification

In last two subsections, we showed that the proposed measures of uncertainty are sensitive to different sources of uncertainty and this allows the system to understand which type of uncertainty it is dealing with. The third question to be answered in the experiment section is the comparison between the uncertainties when the output of the model is correct versus when it is not. This experiment is expected to test if there is any meaningful increase in the model uncertainty when the output is not correct. If that is the case, then the system will be able to detect potential errors in general and appropriately intervene.

Figure 5 depicts the aleotaric, epistemic and combined uncertainties, averaged for both correctly and incorrectly classified data samples. According to the figure, the uncertainties in the correctly classified data samples (the yellow bar) are steadily lower than the ones that are misclassified (the blue bar). This indicates that the model renders more mistakes on the data samples on which it is less certain. Therefore, the uncertainties developed here can serve their true purpose and can be used for detecting any potential errors in predictions.

To further investigate the effectiveness of the proposed uncertainty metrics, we compare it to the Softmax output. To do this, based on Figure 5-a we determine 0.05 as a threshold on our combined uncertainty for distinguishing between certain *vs.* uncertain data samples. Surprisingly, 43% of all misclassified samples that are labeled as uncertain by our system (*i.e.*, their uncertainty is above the threshold) have a Softmax output of higher than 0.95. This shows that, the misclassified samples that are even not located close to the decision boundary of the neural network (*i.e.,* Softmax is very certain about them) can be identified as uncertain by our proposed uncertainty metric. In other words, the proposed uncertainty metric is capable of capturing the uncertainty even for the samples that are far from the decision boundaries of the classifier.

VI. CONCLUSION

We proposed a unified deep Bayesian neural network to detect different types of uncertainties for human activity recognition using wearable sensors. Through experimental analysis, we demonstrated how data-dependent and model-dependent uncertainties could be distinguished and measured by the proposed method. Estimation of different kinds of uncertainties is essential in designing active learning paradigms and novelty detection as well as fault detection and data fusion tasks, which are all important in designing effective and personalized activity recognition systems.

VII. ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation, under grants CNS-1734039 and EEC-1648451.

REFERENCES

[1] Guenterberg, E., Ghasemzadeh, H., Loseu, V., & Jafari, R. (2009, June). Distributed continuous action recognition using a hidden markov model in body sensor networks. In International Conference on Distributed Computing in Sensor Systems. Springer, Berlin, Heidelberg.

[2] Gal, Y. and Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In international conference on machine learning, pp. 1050–1059, 2016.

[3] Kendall, A. and Gal, Y. What uncertainties do we need in bayesian deep learning for computer vision? In Advances in neural information processing systems, pp. 5574–5584, 2017.

[4] Subramanya, A., Srinivas, S., and Babu, R. V. Confidence estimation in deep neural networks via density modelling. arXiv preprint arXiv:1707.07013, 2017.

[5] Mannini, A., & Intille, S. (2018). Classifier Personalization for Activity Recognition using Wrist Accelerometers. IEEE JBHI.

[6] Sztyler, T., & Stuckenschmidt, H. (2017, March). Online personalization of cross-subjects based activity recognition models on wearable devices. In Pervasive Computing and Communications (PerCom), 2017 IEEE International Conference on (pp. 180-189). IEEE.

[7] Banos, O., Damas, M., Guillen, A., Herrera, L. J., Pomares, H., Rojas, I., & Villalonga, C. (2015). Multi-sensor fusion based on asymmetric decision weighting for robust activity recognition. Neural Processing Letters, 42(1), 5-26.

[8] Roy, P. C., Abidi, S. R., & Abidi, S. S. (2017). Possibilistic activity recognition with uncertain observations to support medication adherence in an assisted ambient living setting. Knowledge-Based Systems, 133.

[9] D Roggen, S Magnenat, M Waibel, and G Tr¨oster, "Designing and sharing activity recognition systems across platforms: methods from wearable computing," IEEE Robotics and Automation Magazine, vol. 12, pp. 83–95, 2011.

[10] Kingma, D. P. and Welling, M. Auto-encoding variational bayes. arXiv preprint arXiv:1312.611