This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2883905, IEEE Internet of Things Journal

Exploiting Emerging Sensing Technologies Towards Structure in Data for Enhancing Perception in Human-centric Applications

Murat Ozatay, *Student Member, IEEE,* and Naveen Verma, *Member, IEEE*

***Abstract*—Structure in data can be leveraged to enhance learning. In many perception tasks, the embedded signals arising from physical processes of interest naturally have structure of high semantic relevance. However, traditional forms of *remote sensing* (e.g., vision) preserve such structure only in limited ways. This paper examines how embedded, form-fitting sensing, referred to as *physically-integrated (PI) sensing*, can preserve such structure in richer ways. While the analysis is agnostic to the particular technology for PI sensing, for which a range of options is emerging, especially driven by the Internet of Things (IoT), a particular emerging technology called Large-Area Electronics (LAE) is considered. Using synthetic data from 3D modeling and rendering of human-activity scenes, LAE-based PI sensing and vision-based remote sensing are emulated and perception systems are formed, showing: (1) enhanced data-efficiency of learning models based on PI sensing; (2) potential for selective deployment of PI sensors in new perception tasks, thanks to robust ranking of their value in such tasks; (3) enhanced data-efficiency of learning models based on vision sensing, by integrating PI sensing; (4) efficient mapping of PI-sensing features across perception tasks to enhance transferability of learning.**

***Index Terms*—Internet of Things, physically-integrated sensing, activity detection, artificial intelligence, machine learning, large-area electronics.**

I. Introduction

WHILE learning-based perception systems have achieved great success in many practical cyber applications, their expansion to physical applications, pervading our living and working environments, has been much more limited. This has particularly been the case in human-interactive systems, where learning and/or adaptation must be achieved within timescales and at levels of robustness compatible with human activities [1]. The potential challenges that have been noted include increased statistical diversity of data due to the relatively unconstrained nature of physical processes, noise in those processes, and noise in the sensing devices involved [1], [2]. Autonomous vehicles are one prominent example of a physical application, in which we note that a *confluence of sensing technologies* have been required, along with corresponding algorithmic solutions, to address such challenges [3], [4].

This work was supported in part by C-BRIC, one of six centers in JUMP, a Semiconductor Research Corporation (SRC) program sponsored by DARPA.

The authors are with the Department of Electrical Engineering, Princeton University, Princeton, NJ, 08544 USA (e-mail: mozatay@princeton.edu; nverma@princeton.edu).

The focus of this paper is to explicitly explore how emerging technologies for embedded sensing might impact learning systems, particularly for human-activity applications. A range of technologies for embedded sensing are emerging, especially driven by the Internet of Things (IoT) [5]. These enable what we refer to in this paper as *physically-integrated (PI) sensing*, where sensors are directly coupled with physical objects [6]. This is in contrast with more traditional *remote sensing* (e.g., vision). The key attribute of PI sensing is that it aims to strongly preserve structure in sensor data *based on the interactions between physical objects* (this is somewhat different than tactile sensing, which also involves direct coupling with physical objects). For instance, in human-activity applications, PI sensing follows the insight that the ways we interact with objects around us, say something about our activities and underlying intentions. Assigning sensors to specific objects enables invariant and semantically-relevant structure in data, based on our interactions with these objects. On the other hand, in remote sensing, indirect coupling with embedded signals (e.g., changing objects in an imager's field of view), requires first detecting what embedded signals are being sensed and then forming these into semantic features. For instance, deep convolutional neural networks (CNNs) do this by first applying correlation filters within each layer, and then hierarchically composing outputs into features through subsequent layers.

This paper examines how structure provided by PI sensing overcomes the need for such ground-up learning, over a large space of physical activities and associated embedded signals. However, PI sensing can potentially require the deployment of a large number of sensors, and further such deployment must not be disruptive to the natural interactions between objects, to ensure that structure arising from such interactions is preserved. To address the realism of PI sensing, a particular technology called Large-Area Electronics (LAE) is considered. LAE has the ability to form large numbers and diverse types of sensors, integrated in sheets that can be large (square meters), thin (micrometers), and highly conformal [6]. A dataset for human-activity detection is synthesized from 3D modeling and rendering software, emulating vision sensing as well as PI sensing, from demonstrated LAE systems. This enables analyses comparing learning systems based on each type of sensing, as well as approaches to minimize the number of PI sensors, by combining PI and vision sensing and by selecting which PI sensors to deploy.

The main contributions of this paper are: (1) the creation of a simulation framework for evaluating different sensing

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2883905, IEEE Internet of Things Journal

technologies, namely LAE-based PI sensing and vision-based remote sensing; (2) analyses, based on this framework, of important metrics surrounding perception. Specifically, the analyses contributed are as follows:

- **Data efficiency of learning.** The number of examples required to train a model is a critical concern, particularly in human-interactive systems. This analysis evaluates how the structure provided by PI sensing can enable simple linear models for activity detection, requiring much fewer training examples, as compared to CNNs required for vision sensing. Further, the ability to employ very simple features based on the sensor data is demonstrated.
- **Robust ranking of PI sensors.** Given the cost of sensor deployment in PI sensing, it is useful to understand the relative value of each PI sensor within perception tasks, so that the number of deployed sensors can be optimized. Further, it is useful to assess how consistent that relative value is across tasks, so that the optimal deployment in new tasks can be robustly predicted from the start. Significant diversity in relative value across sensors and consistency in relative value across different human-activity-detection deployments is demonstrated, suggesting the potential for selective deployment of PI sensors.
- **Integration of PI and vision sensing.** Particularly observing that significant benefits from PI sensing can be derived from the deployment of a relatively small number of sensors, the value such deployments can provide when combined with vision sensing is explored. This can enable increased data efficiency while further simplifying sensor deployment. Gains in data efficiency, relative to baseline vision sensing, are demonstrated in accordance with the relative ranking of PI sensors, thus enabling designer trade-offs between scale of sensor deployment and data efficiency.
- **Transfer learning with PI sensing.** Structure relating human activity with object interactions is expected to be preserved across different human-activity-detection deployments, suggesting significant potential for transfer learning with PI sensing. Thus, the ability to efficiently infer how objects in a new deployment map to those in a previous deployment, where perception models have already been trained, can be highly beneficial. Algorithms for such mapping are explored, and the ability to rapidly learn such mapping is demonstrated.

The remainder of this paper is organized as follows. Section II provides background and related work on sensors for activity detection. Section III gives background information on LAE technology for PI sensing. Section IV describes the methodologies for synthetic dataset generation. Section V discusses the experimental procedure and provides experimental results for the analyses listed above, followed by discussions. Finally, Section VI concludes.

II. Background and Related Work

Human-activity detection, and associated sensing technologies, have been of great interest for some time [7], [8].

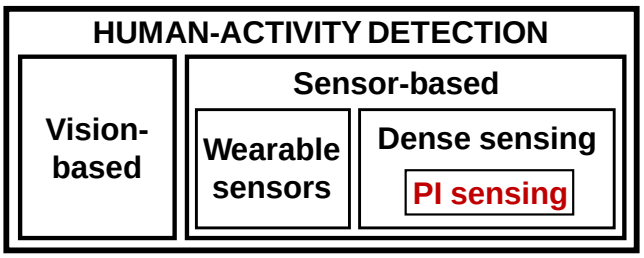


Fig. 1. Classification of activity recognition systems in terms of the type of sensors and the way they are deployed.

Previous works have classified the sensing technologies, as shown in Fig. 1 [5]: (1) vision-based activity detection; and (2) sensor-based activity detection. In this paper, we adopt the slightly different classification and terminology of remote- versus PI-based sensing, respectively, to distinguish based on what we see as the critical distinction: namely, how the two categories of sensors couple with embedded signals, yielding corresponding structure in the sensor data.

Most recently, vision-based sensing has most dominated in human-activity detection. This has included both video monitoring of humans [9]–[15], as well as, recently, still-image human-activity detection, based on analyzing human poses and interactions with objects in a scene [16].

However, for the past 15 years, a growing number of works have emerged employing sensor-network-based human-activity detection [5], [8]. In some of these works, sensors are attached to a human, namely *wearable sensors* [17]–[19], while, in other works, sensors have been attached to objects in the environment, namely *dense sensors* [20]. For instance, [20] uses an RFID sensor network, which includes RFID tags with accelerometer attached to objects for tracing object movement to detect human activity. Going further, [21]–[24] combine wearable and dense sensors, by using an RFID glove/bracelet worn by humans and RFID tags attached to objects. In [21], it is shown that user-object interactions potentially offer a powerful way to infer activities.

Of course, vision- and sensor-based human-activity detection are not mutually exclusive. [25] combines vision-based human tracking with RFID-based object tracking to improve the estimation of high-level interactions between people and objects, for application domains such as retail, home-care, workplace-safety, manufacturing, and other. Similarly, [26] proposes a dynamic Bayesian network model which combines RFID and video data to automatically learn object models and recognize activities.

Importantly, such integrative sensing in human-activity settings raises concerns surrounding privacy protection, simultaneously with energy-aware task management, and optimal resource allocation. [27] proposes a dynamic privacy protection model ensuring privacy even over large volumes of data transmission for resource constrained devices. [28] uses a reinforcement-learning-based resource-allocation approach to achieve optimal allocation in complex networking environments. [29] proposes an algorithm to reduce the total energy

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2883905, IEEE Internet of Things Journal

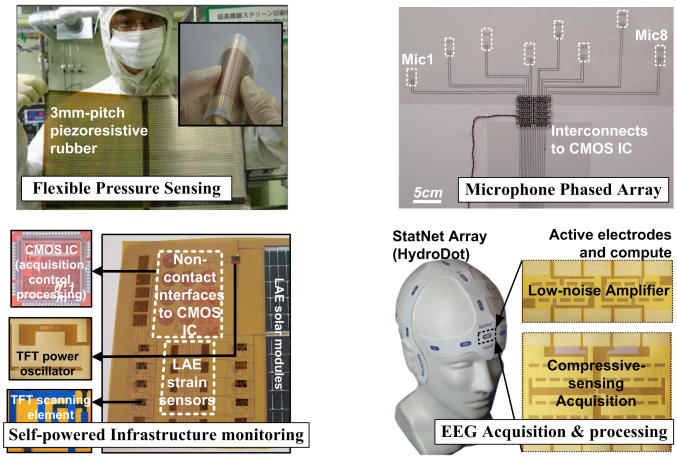
cost of mobile heterogeneous embedded systems via energy-aware task assignment to heterogeneous cores and mobile clouds.

While previous research has shown distinct gains by introducing sensor-based activity detection, the technologies employed and, accordingly, the features derived from these have been limited. For instance, requiring users to wear gloves/bracelets can be restrictive to scale and diversity, due to obtrusiveness of the hardware and sensing of only objects being handled, as well as to the interaction modalities acquired, due to need for close proximity and readout of specific state variables such as object acceleration. This work is motivated by emerging technologies for PI sensing, which provide greater scale, reduced obtrusiveness, and greater sensing cross-section (i.e., not limited to close-proximity objects). This is ultimately essential for preserving the targeted structure in sensor data, given that human interactions span many objects across their different activities, are otherwise hindered by obtrusive technologies, and extend over the length scales humans move about. The analysis that follows in this work is based on such an emerging technology for PI sensing, described next.

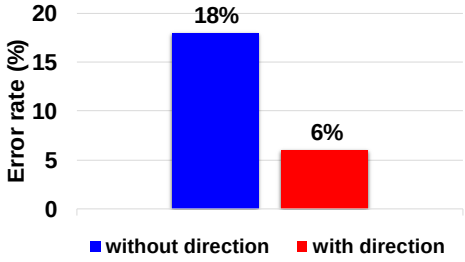
III. Large-Area Electronics (LAE) for PI sensing

Large-Area Electronics (LAE) is a technology based on processing thin films of materials (10's to 100's of nanometers thick) at low temperatures, in the range of 200°C (as compared to conventional silicon electronics, used for integrated circuits, which requires processing at over 1000°C). Low temperature enables compatibility with a broad base of materials and fabrication methods, leading to a wide range of transducers for sensing and energy harvesting, which can be integrated on substrates such as plastic, paper, glass, etc. that can be large and conformal. LAE is thus emerging as a platform technology for diverse and expansive (square meter) arrays of form-fitting sensors [30]–[33]. Fig. 2a provides an illustration of some such sensors and form factors that have been achieved [34]–[37].

Fig. 2b shows the potential value such sensors bring to perception, by preserving physical structure of the embedded processes within the sensed signals. Here, a microphone phased array is used for enhancing audio-based activity detection, by providing source-directionality features in addition to audio features. Specifically, the audio features employed include the first 12 mel-frequency cepstral coefficients (discarding 0^{th} coefficient), zero crossing, energy, energy entropy, harmonic ratio, fundamental frequency, spectral entropy, and spectral roll-off [38], while the directionality feature includes the audio-source angle from a plane centered at and normal to the surface on which the array is mounted. The performance of the system is tested using ESC-10 dataset, which consists of a labeled set of 400 environmental sound recordings equally balanced between 10 classes [39]. During recording phase, each class of sounds is played from its unique location with respect to microphone array resulting in 10 different sound source locations. As seen in Fig. 2b, a detector based on SVM classification for the feature vector yields substantially higher performance with the directionality feature.



(a) Examples of demonstrated systems [34]–[37].



(b) Ability for LAE sensing systems to exploit physical structure of embedded signals.

Fig. 2. Illustration of Large-Area-Electronics (LAE) sensing systems.

LAE, as a commercial technology today, is used for solar cells and flat-panel displays, with fabrication being performed on $10m^2$ glass substrates and the industry moving to flexible plastic substrates. There has been growing commercial interesting in expanding LAE to diverse embedded-sensing applications, with a range of LAE-based sensing systems recently being demonstrated [40].

While the early stage of deployment of LAE in such applications leaves many practical questions still to be answered, a major driver often cited is the potential for LAE to achieve large-scale, form-fitting sensing at very low cost points. Recent studies focused on ramping up manufacturing of flexible electronics propose projections suggesting competitive cost models will be achieved in the near future [41], and in fact specific manufacturing pathways for flexible electronics, such as roll-to-roll (R2R) processing, open up the potential for extremely aggressive cost models. For instance, RFID tags have been in production using R2R processing for more than 20 years, and the number of tags sold per year has increased with a steady decrease in the price per tag (e.g., 100M tags sold in 2004 at $1.15 per tag and 10B tags sold in 2016 at $0.057 per tag) [42]. All of this potential has motivated an advanced manufacturing initiative, at a level of investment of $120M by the U.S. government and industry, to foster an industry around flexible electronics [43].

With such activity around LAE, this paper ground the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2883905, IEEE Internet of Things Journal

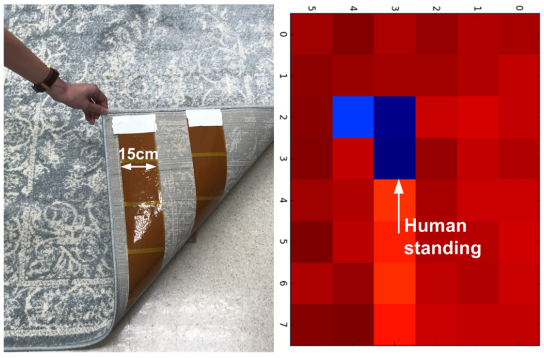
realism of PI sensing by considering two systems based on LAE technology, both of which have been experimentally demonstrated. An overview of these is provided below.

A. Floor-based Human-location Sensors

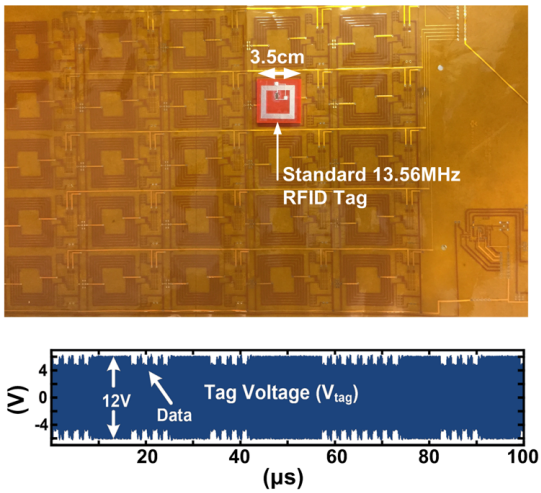
An LAE system for floor-based human-location sensing is considered. It is based on extended-range capacitance sensing, presented in [44]. To sense the locations of humans in a large room, a thin (50μm-thick) array of metal electrodes is integrated in the flooring. The proximity of humans above each electrode changes the electrode's self-capacitance (i.e., by changing the distance of coupling to electric charge carried by humans), which can be measured by low-power electronics. Capacitance sensing presents a number of advantages for floor-integrated sensing. Most notably: (1) thin metal electrodes can be easily and cheaply integrated over large areas; and (2) absence of mechanical transduction (deformation, strain) enhances reliability. A challenge with capacitance sensing is stray coupling of electrodes to extraneous charge, not associated with humans. While typical self-capacitance sensing arrays select and drive one electrode at a time, the system in [44] also actively drives the de-selected electrodes in a manner that mitigates such stray coupling. This enhances the forward projected capacitance, maintaining high sensitivity at increased distances. Fig. 3a shows the floor-based sensors integrated within a carpet tile, along with the sensor-readout map showing human position.

B. RFID-reader Array

An LAE system for object detection and localization is considered. It is based on an array of RFID readers, which are integrated in a thin (50μm-thick) sheet and which can be individually selected using row/column control signals. Each RFID reader in the array, when selected, provides power to a standard 13.56 MHz (ISO14443) passive RFID tag, via near-field inductive coupling. The tag transmits its code by modulating the electrical load it presents, which is then demodulated by the RFID reader array. The RFID reader array can cover surfaces such as floors, tables, counters, etc., enabling the detection and localization of tagged objects on the surfaces. Such a system exploits the low cost and scalability of passive RFID tags, which can be readily and unobtrusively deployed on a large number of objects. Fig. 2b shows a version of the RFID reader array, implemented as a flexible sheet, along with an oscilloscope recording of the tag-side voltage waveform arising from load modulation.



(a) Floor-based human-location sensing system.



(b) RFID-reader array system.

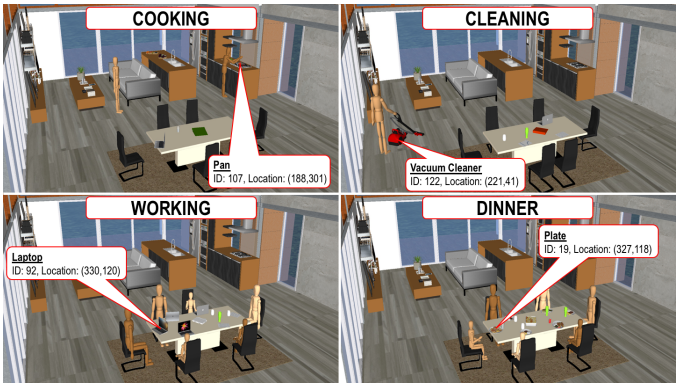
Fig. 3. Experimentally-demonstrated PI sensing systems considered in this work.

IV. Synthesized Human-Activity Detection (SHAD) Dataset

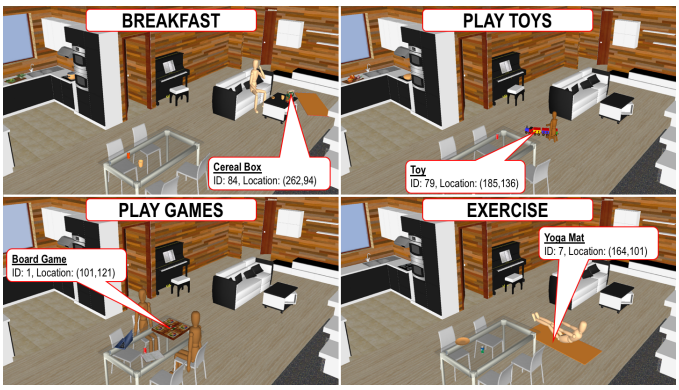
To emulate both vision and PI sensing, we synthesize a dataset, referred to as the Synthesized Human-Activity Detection (SHAD) dataset, corresponding to human-activity scenes. This is achieved using the 3D modeling and rendering software SketchUp [45]. SketchUp provides a library of objects, which can be selected and placed within images. Scene construction by object placement in this way enables emulation of PI sensing, while scene rendering enables emulation of vision sensing. Depending on their complexity, human activities can be categorized into different levels [7]: (1) gestures; (2) atomic actions; (3) human-to-object or human-to-human interactions; (4) group actions; (5) behaviors; and (6) events. During construction of the SHAD dataset, we focus on human-to-object interactions. Two home environments are constructed, referred to as SHAD-1 and SHAD-2, respectively, and scenes are generated for ten human-activity classes: *breakfast, dinner, work, clean, cook, exercise, play toys, play games, recreation, no activity*. SHAD-1/2 each consists of 20k scenes (2k per class). Each scene is rendered as a 400×400 color-pixel image, and consists of 126/87 household objects and 9/6 humans. Fig. 4 shows sample images from each home environment for four human-activity classes, along with annotations of object locations and codes (as would be derived from RFID tags). The approach to scene construction and details of the SHAD-1/2 datasets are provided below.

For each of the home environments, location distributions are defined for 25/26 groups of household objects (plates/bowls, cups, cans, games, small appliances, etc.). For each object in each home environment, ten different distributions were defined, for the ten different human-activity classes. The dataset is synthesized for each home environment by performing Monte Carlo sampling from the object-location

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2883905, IEEE Internet of Things Journal



(a) First home environment (SHAD-1).



(b) Second home environment (SHAD-2).

Fig. 4. Images synthesized in SketchUp from Monte Carlo sampling of object-location distributions.

distributions. Scene construction by sampling object locations in this ways enables emulation of PI sensing, while scene rendering enables emulation of vision sensing.

A. Scene Creation

In this section, we present details of the scene-creation methodology, given in Algorithm 1. There are four main steps taken for scene creation:

1. For each home environment and each human-activity class, the probability of whether or not to place a household human/object is defined.
2. For each home environment, each human-activity class, and each household human/object a probability density function (PDF) is defined for the location the human/object can take. For object locations, the PDF employed is conditional on the sampled locations of humans, such that a human sitting at the dining table for dinner would cause appropriate positioning of dishes and cutlery. All object-location PDFs also capture some probability of random clutter. For illustration, Fig. 5 shows conditional PDFs for various objects (such as plate, cup, can, bowl, board game, notebook, laptop, etc.) on the dining table in SHAD-1 for the *dinner* and *play games* classes. In Figure 5b, the aim of placing objects at region 2 and 3 is to create clutter.
3. Scenes are constructed by placing humans/objects via Monte Carlo sampling of the PDFs.
4. The sampled location of each object is checked to identify whether it results in overlap of humans/objects. If overlap is found, either re-sampling of a new location is performed for the corresponding objects, or, depending on the object, removal from the scene is performed.

Algorithm 1 Human-Activity Detection Dataset Generation

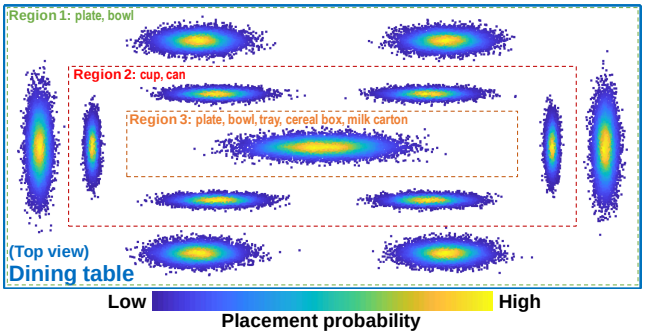
Require: $Class$, ID, $E_{human-obj|Class}$ (conditional distribution of human/object existence in the scene), $L_{human|Class}$, $L_{obj|Class,loc_{humans},rot_{humans}}$ (conditional distribution of human and object locations), $R_{human|Class,loc_{human}}$, $R_{obj|Class,loc_{obj}}$ (conditional distribution of human and object rotations)

Ensure: $SceneImage$, $LocID$

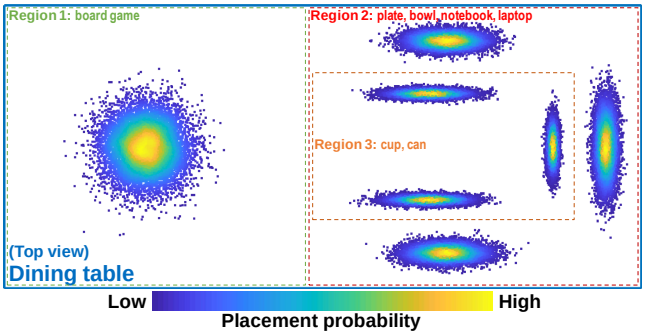
for each human **do**
  sample from $E_{human|Class}$
  if human exists in the scene **then**
    sample location loc_{human} from $L_{human|Class}$
    sample rotation rot_{human} from $R_{human|Class,loc_{human}}$
    if no overlap with other humans **then**
      place human to location loc_{human} with rotation rot_{human}
      store ID_{human} and loc_{human} in key-value storage $LocID$
    end if
  end if
end for
for each object **do**
  sample from $E_{obj|Class}$
  if object exists in the scene **then**
    sample location loc_{obj} from $L_{obj|Class,loc_{humans},rot_{humans}}$
    sample rotation rot_{obj} from $R_{obj|Class,loc_{obj}}$
    if no overlap with other humans and objects **then**
      place object to location loc_{obj} with rotation rot_{obj}
      store ID_{object} and loc_{obj} in key-value storage $LocID$
    else
      if object is crucial in the scene **then**
        go back to Step 13
      end if
    end if
  end if
end for
store image in $SceneImage$
return $SceneImage$, $LocID$, $Class$

We note that such a synthetic dataset depends on how the PDFs are defined, and that different PDFs could be more or less realistic in various ways. Nonetheless, the problem of activity inference remains valid with such a dataset, i.e., where the aim is to determine the underlying distribution from observed samples of the distribution. Furthermore, the way our PDFs are defined explicitly asserts structure, i.e., by using conditional PDFs for object locations based on sampled human locations. Nonetheless, we believe our experiments remain

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2883905, IEEE Internet of Things Journal



(a) Distributions for *dinner* class.



(b) Distributions for *play games* class.

Fig. 5. Distributions of object locations in SHAD-1 on the dining table for two different classes.

valid, where the focus is on exploring how such presumed structure can be exploited.

B. Differences between SHAD-1 and SHAD-2

Although SHAD-1 and SHAD-2 are synthesized using the same approach, there are notable differences between them. First, SHAD-1 (126 objects with emulated RFID tags and up to 9 humans) has more objects and humans compared to SHAD-2 (87 objects with emulated RFID tags and up to 6 humans). In addition to more total household objects, SHAD-1 also has higher diversity of the objects placed, in terms of their shapes/colors and also their locations. For example, *breakfast*/*dinner* activities can be performed in 3 different regions (dinning table, sofa, and kitchen), while, for SHAD-2, only 2 regions are possible (dining table and sofa). On the other hand, the PDFs used in SHAD-2 represent more random clutter of objects in the scenes. Nonetheless, considering these differences, we believe that SHAD-1 presents somewhat greater inference complexity than SHAD-2.

V. Experiments

In this section, we describe the experimental procedure, and present as well as discuss experimental results for the four analyses conducted, focusing on: data-efficiency of learning, robust ranking of PI sensors, integration of PI and vision sensing, and transfer learning with PI sensing.

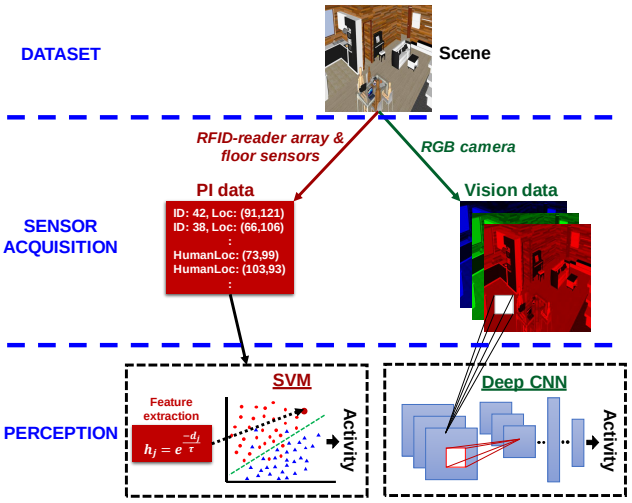


Fig. 6. Illustration of data-efficiency experiments using SHAD datasets.

A. Data-efficiency of Learning

This analysis employs different learning models for activity detection using PI and vision sensing, where the structure in PI sensor data is exploited towards a simple, linear model. Fig. 6 illustrates the overall approach taken for the experiment. For PI sensing, a linear support-vector machine (SVM) is applied to human/object-locations features, which are derived by emulating the LAE floor-based capacitance-sensing and RFID reader-array systems. For vision sensing a CNN is applied to images of the human-activity scenes.

Linear SVM and feature extraction. A linear SVM (with penalty parameter of 1.0) is used to create binary classifiers for all pairs of classes, and multi-class decisions are derived via majority rule voting as implemented by the scikit-learn Python package [46]. A feature is computed for every object, corresponding to the object's proximity to the nearest human. For the j^{th} object, the corresponding feature h_j is calculated as

$$h_j = e^{-\frac{d_j}{\tau}}, \tag{1}$$

where d_j is the absolute distance (in inches) between object j and the nearest human, and τ is a decay factor. If there is no human in the scene or if the object does not appear in the scene, h_j is set to 0. h_j thus yields higher feature values for objects close to a human while ensuring that features are bounded in $[0, 1]$. This follows the insight that relevant interactions tend to occur with nearby objects.

Deep CNN architecture. The CNN used is adapted from the AlexNet base architecture [47], and is implemented using Keras with TensorFlow backend [48], [49]. The network contains 5 convolutional layers followed by 3 fully-connected layers as listed in Table I. The first, second, and fifth convolutional layers are followed by batch normalization and max pooling layers. The first convolutional layer has 48 kernels of size $11\times11\times3$ with a stride of 4 pixels. The second convolutional layer has 128 kernels of size $5\times5\times48$, while the third and forth convolutional layers have 192 kernels of size $3\times3\times128$ and $3\times3\times192$, respectively, and the fifth convolutional layer

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2883905, IEEE Internet of Things Journal

has 128 kernels of size 3×3×192. Zero-padding is used in all convolutional layers. The first 2 fully-connected layers have 1024 neurons each, and the last layer derives 10 outputs via a softmax operation. We apply dropout of 0.5 between fully-connected layers. Activation function selection and the initialization of weights and biases are given in Table I. For training, stochastic gradient decent (SGD) is employed with momentum of 0.9, decay of 0.0005, and batch size of 40 samples.

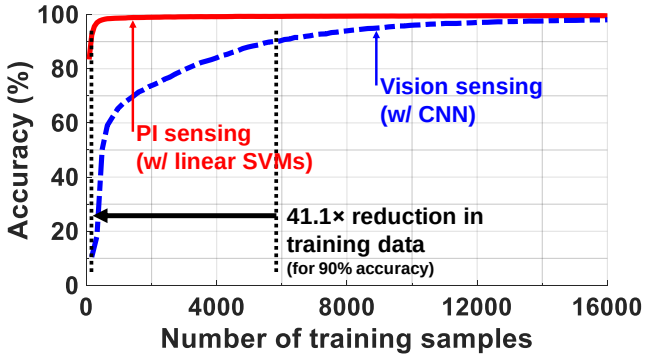
Algorithm 2 Data Efficiency of Learning Analysis

Require: $X_{PI,train}$, $X_{PI,test}$ (feature extracted PI data), $X_{V,train}$, $X_{V,test}$, $y_{PI,train}$, $y_{PI,test}$, $y_{V,train}$, $y_{V,test}$ (labels), $TotalSamples$

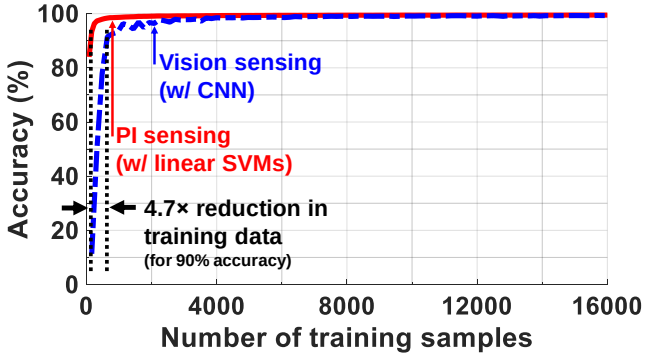
Ensure: acc_{PI}, acc_V

initialize $StepSize$ and $TotalSteps$
$step \leftarrow 1$
for $step \leq TotalSteps$ **do**
  $nTrain \leftarrow step \times StepSize$
  $X_{train} \leftarrow X_{PI,train}(1:nTrain,:)$
  $y_{train} \leftarrow y_{PI,train}(1:nTrain)$
  $modelPI \leftarrow train.SVM(X_{train}, y_{train})$
  $y_{pred} \leftarrow predict.SVM(modelPI, X_{PI,test})$
  $acc_{PI}(step) \leftarrow accuracy(y_{pred}, y_{PI,test})$
  $step \leftarrow step + 1$
end for
initialize $StepSize$ and $TotalSteps$
$step \leftarrow 1$
for $step \leq TotalSteps$ **do**
  $nTrain \leftarrow step \times StepSize$
  $X_{train} \leftarrow X_{V,train}(1:nTrain,:,:,:)$
  $y_{train} \leftarrow y_{V,train}(1:nTrain)$
  if step == 1 **then**
    $modelV \leftarrow train.CNN(X_{train}, y_{train})$
  else
    $modelV \leftarrow retrain.CNN(X_{train}, y_{train}, modelV)$
  end if
  $y_{pred} \leftarrow predict.CNN(modelV, X_{V,test})$
  $acc_V(step) \leftarrow accuracy(y_{pred}, y_{V,test})$
  $step \leftarrow step + 1$
end for
return acc_{PI}, acc_V

For training/validation/testing, each home-environment dataset is split into training (80%) and testing sets (20%), with balanced classes. During hyperparameter tuning, 25% of the training set is used as a validation set. To evaluate the detection performance versus number of training samples with PI/vision sensing using the SVM/CNN, we start with 80/160 training samples and add 80/160 samples incrementally, at each step re-evaluating the performance as shown in Algorithm 2. For vision sensing with the CNN, at each step we re-train the CNN from the previous step, instead of creating a new model. For SHAD-1, we use a constant number of 50 epochs for CNN training at each step. For SHAD-2, we use 100, 80, 50, and 20 epochs at steps 1-7, 8-50, 51-80, 80-100, respectively. Initial learning rates are set to 0.01 and 0.0004 for SHAD-1 and SHAD-2, and each training/testing sequence



(a) SHAD-1 ($\tau = 300$).



(b) SHAD-2 ($\tau = 100$).

Fig. 7. Data-efficiency comparison of PI and vision sensing for the two home environments.

is repeated 10 times with randomly shuffled training/testing datasets to average the obtained results. Fig. 7 shows the testing accuracy of PI and vision sensing. For both home environments, PI sensing shows substantial gains in data efficiency, with accuracy achieving the 90% point with 41.1× and 4.7× less training data respectively (i.e., at 139/132 samples, compared to 5717/623 samples, for SHAD-1/2). We also see that the accuracy of vision sensing converges much more slowly for SHAD-1 than SHAD-2, owing to the greater expected complexity of SHAD-1.

B. Robust Ranking of PI Sensors

This analysis employs *Fisher score* as a metric for measuring the discriminative power of each feature in PI sensing [50]–[52]. The Fisher score is derived for the proximity feature h_j associated with each object, and then similar objects (e.g., plates, cups, etc.) are grouped together. Then, the average Fisher score for each group is computed, and groups of objects are sorted according to average Fisher scores. This enables assessment of the value different objects bring to human-activity detection, ultimately directing how to judiciously deploy of PI sensing.

Fig. 8 shows the ranking of object features based on the group-averaged Fisher score, for the two datasets considered. As seen there is significant diversity in the Fisher score, with a relatively small number of objects having considerably higher relative discriminative power. Analyzing this further,

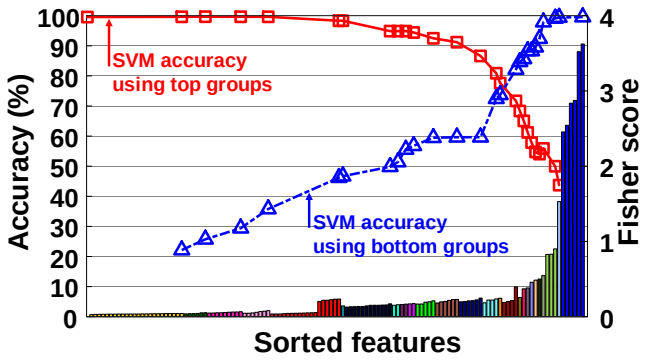
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2883905, IEEE Internet of Things Journal

TABLE I
DEEP CNN ARCHITECTURE.

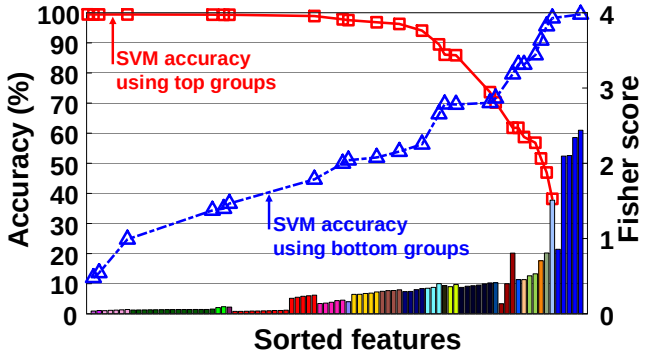
Layer type	# Kernels	Kernel Size	Initializers: Kernel	Initializers: Bias	Activation	Strides	Padding	Pool size	Drop rate	Output shape
Input	-	-	-	-	-	-	-	-	-	400×400×3
Convolution	48	11×11×3	$\mathcal{N}(0, 0.01)$	0	ReLU	4×4	same	-	-	100×100×48
Batch normalization	-	-	-	-	-	-	-	-	-	100×100×48
Max pooling	-	-	-	-	-	-	-	2×2	-	50×50×48
Convolution	128	5×5×48	$\mathcal{N}(0, 0.01)$	1	ReLU	1×1	same	-	-	50×50×128
Batch normalization	-	-	-	-	-	-	-	-	-	50×50×128
Max pooling	-	-	-	-	-	-	-	2×2	-	25×25×128
Convolution	192	3×3×128	$\mathcal{N}(0, 0.01)$	0	ReLU	1×1	same	-	-	25×25×192
Convolution	192	3×3×192	$\mathcal{N}(0, 0.01)$	1	ReLU	1×1	same	-	-	25×25×192
Convolution	128	3×3×192	$\mathcal{N}(0, 0.01)$	1	ReLU	1×1	same	-	-	25×25×128
Batch normalization	-	-	-	-	-	-	-	-	-	25×25×128
Max pooling	-	-	-	-	-	-	-	2×2	-	12×12×128
Flatten	-	-	-	-	-	-	-	-	-	18432×1
Dense	1024	18432×1	$\mathcal{N}(0, 0.01)$	1	ReLU	-	-	-	-	1024×1
Dropout	-	-	-	-	-	-	-	-	0.5	1024×1
Dense	1024	1024×1	$\mathcal{N}(0, 0.01)$	1	ReLU	-	-	-	-	1024×1
Dropout	-	-	-	-	-	-	-	-	0.5	1024×1
Dense	10	1024×1	$\mathcal{N}(0, 0.01)$	0	Softmax	-	-	-	-	10×1

we trained SVMs with features from top-k/bottom-k groups (using the full training set), to observe the impact on detection accuracy. To do this, we start from $k = 1$ and increment k at each step until all features are used. The resulting accuracies are shown in Fig. 8, along with the object Fisher-score rankings. First, we see by incrementally removing low-ranking objects (red line), that a significant number of objects can be removed from PI sensing, without degrading accuracy. Second, we see by incrementally adding higher-ranking objects (blue line), that a significant increase in accuracy is observed due to small number of objects. Specifically, for the SHAD-1/2 datasets, an accuracy of 90% is achieved with the only 32/28 top-ranked features, as compared to 114/80 bottom-ranked features.

Importantly, we see that there is substantial consistency in the value of features from different objects across the two datasets SHAD-1/2. This suggests that evaluating the value of PI sensing on objects in one deployment enables reliable assessment of how to deploy PI sensing on objects in another deployment. Specifically, Table II provides a sorted list of the top 10 objects whose features yield the highest Fisher score from each dataset. As seen, 5 out of the top 6 object groups are the same. Furthermore, there are 3 groups with similar purpose of use (e.g., vacuum and handheld vacuum have similar purpose as dust pan and floor sweeper, balance ball has similar purpose as pilates ball). Thus, we observe that the relative value of similar objects has substantial consistency across the different human-activity detection systems, suggesting that in a restricted deployment of PI sensing, the optimal deployment of sensors can be readily predicted from the beginning. This indicates that the structure provided by PI sensing also has the potential to enhance transferability of learning across deployments.



(a) SHAD-1 dataset.



(b) SHAD-2 dataset.

Fig. 8. Fisher-score ranked features from objects in PI sensing and SVM accuracy of top/bottom groups. Each accuracy data point shows the testing accuracy of an SVM employing features from the right/left side of the data point.

C. Integration of PI and Vision Sensing

This analysis explores the value of combining PI and vision sensing. Specifically, PI sensing has the potential to preserve structure in sensor data towards enhanced efficiency of learning, while vision sensing incurs lower cost of sensor deployment. Combining the two has the potential to enable designer knobs for trading off these factors. For this, the CNN architecture applied to vision sensing is modified by

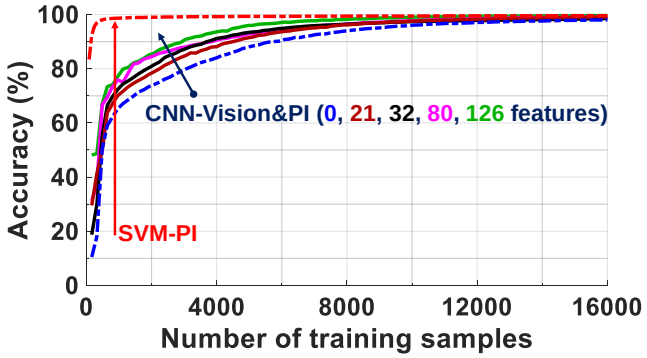
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2883905, IEEE Internet of Things Journal

TABLE II
TOP-10 OBJECT GROUPS WITH RESPECT TO FISHER SCORE.

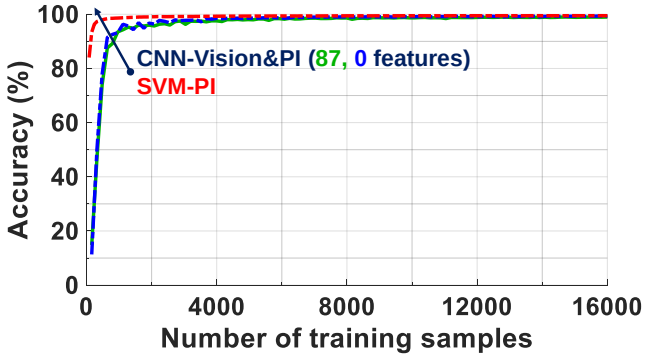
SHAD-1		SHAD-2	
#	Group	#	Group
1	**chair**	**1**	**chair**
2	**guitar**	**2**	**guitar**
3	**cutting board**	**3**	**yoga mat**
4	**yoga mat**	4	*pilates ball*
5	*vacuum*	**5**	**cutting board**
6	**broom**	**6**	**broom**
7	milk carton	7	*dust pan*
8	easel	8	board game
9	*handheld vacuum*	9	*floor sweeper*
10	*balance ball*	10	bowl

concatenating the features extracted from PI sensing with the output of the second fully-connected layer. The resulting concatenated feature vector is then provided to the last CNN layer. To expose the designer knob, various models are trained, scaling the numbers of PI sensors employed based on the rankings obtained in Sec. V-B.

For testing, the procedure outlined in Sec. V-A is applied. Fig. 9 shows the testing accuracy when combining PI and vision sensing for the two home environments. As seen in Fig. 9a, adding more features from PI sensing in SHAD-1 clearly and substantially improves data efficiency, even when adding a small number of PI-sensing features. This suggests that structure can be exploited even with modest sensor-deployment costs, and motivates further research in learning models combining features from the two types of sensors. Fig. 9b shows that adding PI-sensing features in SHAD-2 has somewhat less impact, as the simpler dataset already benefits from more rapid learning convergence.



(a) SHAD-1 dataset.



(b) SHAD-2 dataset.

Fig. 9. Testing accuracy when combining features from PI sensing with vision sensing.

D. Transfer Learning with PI Sensing

This experiment explores the gains that can be derived from transfer learning with PI sensing. Generally, the effectiveness of transfer learning depends on the similarity of statistics between source- and target-domain data. Given our use of a synthesized dataset based on specified PDFs, the focus of transfer-learning experiments is not on the ultimate effectiveness of transfer learning, but rather on the efficiency with which object codes (i.e., from RFID tags) can be mapped from source-domain data to target-domain data. The motivation behind this is that pre-programming or careful assignment of codes to objects in a new human-activity detection system deployment is not feasible. Instead, if the data statistics are similar thanks to the strong structure, then rapid automatic learning of the mapping between target- and source-domain object codes has the potential to enable substantial gains from transfer learning. To explore this, mapping is performed from SHAD-2 to SHAD-1, followed by model transfer.

Feature-space mapping. To map source- and target-domain features, we use the Informed Feature-Space Remapping (IFSR) method [53]. This involves first deriving *a vector of meta-features*. For each feature, one meta-feature is derived for each target-domain class. Given class c and feature j, the meta-feature vector is formed from meta-features $M_{j,c}$, corresponding to the expected value of the j^{th} feature within the c^{th} class. Next, a *similarity matrix* $\mathbf{S}$ is created by computing a similarity score between each source feature and target feature pair. The similarity score $S_{i,j}$ between the i^{th} source feature and the j^{th} target feature is the negative Euclidean distance between their corresponding meta-feature vectors, given by Eq. 2.

$$S_{i,j} = -\sqrt{\sum_{c=1}^{K} |M_{i,c} - M_{j,c}|^2}, \tag{2}$$

where K is the number of distinct classes. Finally, the feature mapping to target features is established by mapping the source feature that exhibits maximal similarity to the given target feature. As a result, a one-to-many mapping is created from the source feature space to the target feature space. We note that objects not occurring in the available target-domain data all lead to zero-valued meta-features, preventing proper similarity assessment; thus, such object features are explicitly not mapped to any source-domain feature, and are assigned a feature value of zero during model learning and transfer.

Model transfer. After feature-space mapping, transfer learning can be employed to enhance testing accuracy with fewer training samples. For this, we use an Adaptive SVM (A-SVM) model [54], and extend it to our multi-class problem. An A-SVM aims to learn a decision vector w_s by training a linear SVM using all of the source data, and uses w_s to regularize

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2883905, IEEE Internet of Things Journal

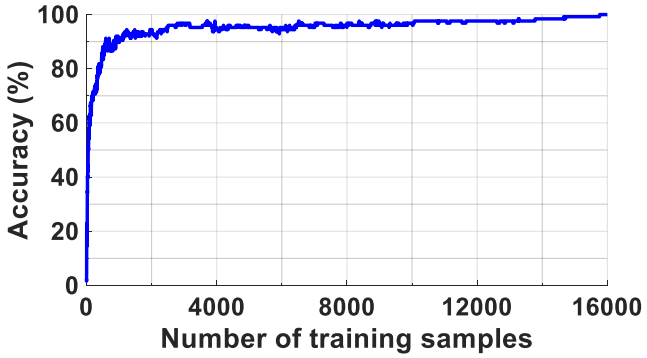
learning of a decision vector w using all of the available target data. This is done by changing regularization term of a classic SVM objective function from $\|w\|^2$ to $\|w - \Gamma w_s\|^2$ as shown in Eq. 3.

$$\min_{w,b} \|w - \Gamma w_s\|^2 + C \sum_{i=1}^{N} max(0, 1 - y_i(w^T \mathbf{x}_i + b)), \quad (3)$$

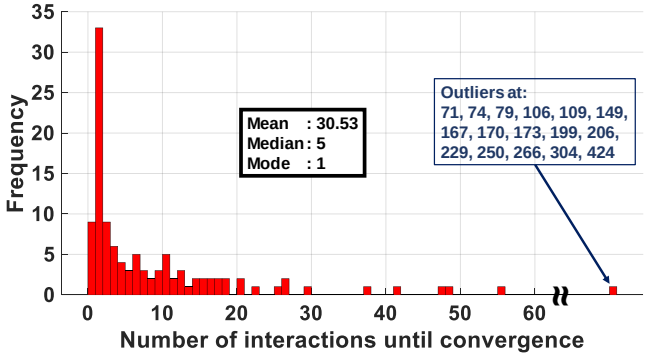
where Γ controls the level of transfer while C controls the weight of the loss function, and N is the number of training samples.

Fig. 10a shows the accuracy of the learned mapping from SHAD-2 (source) to SHAD-1 (target) features, with respect to the number of target domain features used for mapping. Here, to assess accuracy, the mapping learned after using the entire target-domain training dataset is assumed to be the true mapping. In this analysis, the feature-extraction parameter τ is set to 100 for both target and source data during feature-space mapping. However, mapping accuracy represents only one metric for assessing the data efficiency of mapping. In particular, PI sensor data is sparse in the sense that human activities involve interactions with a small number of the total objects. Thus, because objects not present in the scene yield feature values of zero, in fact many of the feature values will be zero. Thus, an alternate metric is the number of times an object is interacted with before its feature is correctly mapped to that from the source domain. This requires defining when an object is interacted with, which we do by saying that proximity within 4 feet of a human is designated as an interaction. From this, Fig. 10b shows a histogram of the number of interactions before correct mapping of a feature is achieved. As seen, the majority of objects require very few interactions (median: 5, mode: 1). However, a few objects require a rather large number of interactions. The primary reason for this is the sparsity of PI data, which is observed to cause notable variance in the meta-feature values until the number of target-domain training samples increases adequately (i.e., because meta-feature computation involves an average taken over the number of available target-domain training samples).

Fig. 11 shows the target domain transfer accuracy for three different scenarios: (1) when there is no transfer (SVM-Target); (2) when there is transfer from source data using the feature mapping and A-SVM (A-SVM); and (3) when there is transfer from source data using a pre-determined feature mapping, i.e., using all target data for mapping (A-SVM-Pre). For this analysis, Γ and C are set to 1 and 100, respectively. As seen, despite the high data-efficiency of learning to begin with (SVM-Target), data-efficiency is improved with transfer learning, when there is very little target data (less than 100). Further, the difference between the A-SVM and A-SVM-Pre cases shows that there is further room to enhance efficiency with transfer learning by focusing on increasing the efficiency of feature mapping.



(a) Mapping accuracy versus training samples.



(b) Histogram of number of interactions in target domain with object before correct feature mapping is achieved.

Fig. 10. Data efficiency of feature space mapping performance for PI sensing.

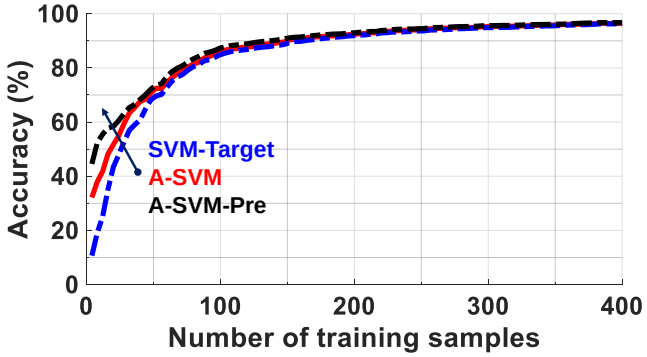


Fig. 11. Testing accuracy with transferring learning from SHAD-2 (source) to SHAD-1 (target).

VI. Conclusion

With sensors being the main source of data in perception systems and subsystems, this paper evaluates how structure enforced in the sensor data by the sensing technology itself can enhance learning algorithms. Specifically, the focus is on physically-integrated (PI) sensing, where sensors are directly coupled to embedded signals expressing human interaction with objects, in contrast to remote sensing, where sensors are not directly coupled to the embedded signals. The structure enforced by PI sensing follows from the assertion that human interactions are indicative of human activities and intentions. By starting with demonstrations of PI sensing and emerging PI-sensing technologies, namely based on Large-Area Electronics (LAE), a simulation environment is developed that emulates specific, demonstrated PI sensors and conventional

vision sensors, to create two Synthesized Human-Activity Detection (SHAD-1/2) datasets. Analysis is performed to evaluate: (1) the data efficiency of learning; (2) the ability to predict which PI sensors offer the greatest discriminative value; (3) the prospects of integrating PI and remote sensing; and (4) the prospects of transfer learning with PI sensing across smart-home deployments. Results show that PI sensing substantially enhances data efficiency of learning and that high discriminative power can be achieved with a small, but select, set of PI sensors, potentially addressing the costs of PI-sensor deployment. Results also show that integration of PI and remote (vision) sensing yields a further trade-off between data efficiency and sensor-deployment cost, and suggests this as a promising area for exploring new sensor-fusion algorithms. Finally, given the likelihood of high transferability of models for human-activity detection across smart-home deployments, this work exposes the potential and some of the challenges with mapping PI-sensing features across deployments for transfer learning.

ACKNOWLEDGMENT

The authors thank Y. Mehlman, P. Kumar, and S. Hsia for their inputs on the LAE sensing technologies. This work was supported in part by C-BRIC, one of six centers in JUMP, a Semiconductor Research Corporation (SRC) program sponsored by DARPA.

REFERENCES

[1] S. Vishwakarma and A. Agrawal, "A survey on activity recognition and behavior understanding in video surveillance," *The Visual Comput.*, vol. 29, no. 10, pp. 983–1009, Oct. 2013.

[2] J. A. Stankovic, "Research directions for the internet of things," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 3–9, Feb. 2014.

[3] E. Frazzoli, M. A. Dahleh, and E. Feron, "Real-time motion planning for agile autonomous vehicles," *J. Guidance, Control, Dynamics*, vol. 25, no. 1, pp. 116–129, Jan. 2002.

[4] M. Gerla, E. K. Lee, G. Pau, and U. Lee, "Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds," in *IEEE World Forum Internet Things (WF-IoT)*, Mar. 2014, pp. 241–246.

[5] L. Chen, J. Hoey, C. D. Nugent, D. J. Cook, and Z. Yu, "Sensor-based activity recognition," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 42, no. 6, pp. 790–808, Nov. 2012.

[6] M. Ozatay, L. Aygun, H. Jia, P. Kumar, Y. Mehlman, C. Wu, S. Wagner, J. C. Sturm, and N. Verma, "Artificial intelligence meets large-scale sensing: Using large-area electronics (lae) to enable intelligent spaces," in *IEEE Custom Integrated Circuits Conf. (CICC)*, Apr. 2018, pp. 1–8.

[7] M. Vrigkas, C. Nikou, and I. A. Kakadiaris, "A review of human activity recognition methods," *Frontiers in Robotics and AI*, vol. 2, p. 28, 2015.

[8] D. Guan, T. Ma, W. Yuan, Y.-K. Lee, and A. M. J. Sarkar, "Review of sensor-based activity recognition systems," *IETE Tech. Rev.*, vol. 28, no. 5, pp. 418–433, 2011.

[9] J. Aggarwal and Q. Cai, "Human motion analysis: A review," *Comput. Vision and Image Understanding*, vol. 73, no. 3, pp. 428 – 440, 1999.

[10] A. F. Bobick, "Movement, activity and action: the role of knowledge in the perception of motion," *Philosoph. Trans. Roy. Soc. London B: Biological Sciences*, vol. 352, no. 1358, pp. 1257–1265, 1997.

[11] D. Gavrila, "The visual analysis of human movement: A survey," *Comput. Vision and Image Understanding*, vol. 73, no. 1, pp. 82 – 98, 1999.

[12] V. Krüger, D. Kragic, A. Ude, and C. Geib, "The meaning of action: a review on action recognition and mapping," *Advanced Robotics*, vol. 21, no. 13, pp. 1473–1501, 2007.

[13] R. Poppe, "A survey on vision-based human action recognition," *Image and Vision Computing*, vol. 28, no. 6, pp. 976 – 990, 2010.

[14] J. Aggarwal and M. Ryoo, "Human activity analysis: A review," *ACM Computing Surveys (CSUR)*, vol. 43, no. 3, pp. 16:1–16:43, Apr. 2011.

[15] G. Cheng, Y. Wan, A. N. Saudagar, K. Namuduri, and B. P. Buckles, "Advances in human action recognition: A survey," *CoRR*, 2015. [Online]. Available: http://arxiv.org/abs/1501.05964

[16] Y. Zhang, L. Cheng, J. Wu, J. Cai, M. N. Do, and J. Lu, "Action recognition in still images with minimum annotation efforts," *IEEE Trans. Image Process.*, vol. 25, no. 11, pp. 5479–5490, Nov. 2016.

[17] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors," *IEEE Commun. Surveys Tut.*, vol. 15, no. 3, pp. 1192–1209, 2013.

[18] D. Wang, E. Candinegara, J. Hou, A. H. Tan, and C. Miao, "Robust human activity recognition using lesser number of wearable sensors," in *Int. Conf. Security, Pattern Anal., Cybern. (SPAC)*, Dec. 2017, pp. 290–295.

[19] G. Ding, J. Tian, J. Wu, Q. Zhao, and L. Xie, "Energy efficient human activity recognition using wearable sensors," in *IEEE Wireless Commun. Networking Conf. Workshops (WCNCW)*, Apr. 2018, pp. 379–383.

[20] M. Buettner, R. Prasad, M. Philipose, and D. Wetherall, "Recognizing daily activities with rfid-based sensors," in *Proc. 11th Int. Conf. Ubiquitous Computing*. New York, NY, USA: ACM, 2009, pp. 51–60.

[21] M. Philipose, K. P. Fishkin, M. Perkowitz, D. J. Patterson, D. Fox, H. Kautz, and D. Hahnel, "Inferring activities from interactions with objects," *IEEE Pervasive Computing*, vol. 3, no. 4, pp. 50–57, Oct. 2004.

[22] D. J. Patterson, D. Fox, H. Kautz, and M. Philipose, "Fine-grained activity recognition by aggregating abstract object usage," in *9th IEEE Int. Symp. Wearable Comput. (ISWC)*, Oct. 2005, pp. 44–51.

[23] T. Gu, S. Chen, X. Tao, and J. Lu, "An unsupervised approach to activity recognition and segmentation based on object-use fingerprints," *Data and Knowledge Eng.*, vol. 69, no. 6, pp. 533 – 544, 2010.

[24] D. Wyatt, M. Philipose, and T. Choudhury, "Unsupervised activity recognition using automatically mined common sense," in *Proc. 20th Nat. Conf. Artificial Intelligence*, vol. 1. AAAI Press, 2005, pp. 21–27.

[25] N. Krahnstoever, J. Rittscher, P. Tu, K. Chean, and T. Tomlinson, "Activity recognition using visual tracking and rfid," in *7th IEEE Workshops Appl. Comput. Vision*, vol. 1, Jan. 2005, pp. 494–500.

[26] J. Wu, A. Osuntogun, T. Choudhury, M. Philipose, and J. M. Rehg, "A scalable approach to activity recognition based on object use," in *11th IEEE Int. Conf. Comput. Vision (ICCV)*, Oct. 2007, pp. 1–8.

[27] K. Gai, K. R. Choo, M. Qiu, and L. Zhu, "Privacy-preserving content-oriented wireless communication in internet-of-things," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 3059–3067, Aug. 2018.

[28] K. Gai and M. Qiu, "Reinforcement learning-based content-centric services in mobile sensing," *IEEE Network*, vol. 32, no. 4, pp. 34–39, July 2018.

[29] K. Gai, M. Qiu, and H. Zhao, "Energy-aware task assignment for mobile cyber-enabled applications in heterogeneous cloud computing," *J. Parallel Distrib. Comput.*, vol. 111, no. C, pp. 126–135, Jan. 2018.

[30] S. Wagner, S. P. Lacour, J. Jones, P. I. Hsu, J. C. Sturm, T. Li, and Z. Suo, "Electronic skin: architecture and components," *Physica E: Low-dimensional Syst. and Nanostructures*, vol. 25, no. 2-3, pp. 326 – 334, 2004.

[31] N. Verma, Y. Hu, L. Huang, W. Rieutort-Louis, J. Robinson, T. Moy, B. Glisic, S. Wagner, and J. Sturm, "Enabling scalable hybrid systems: Architectures for exploiting large-area electronics in applications," *Proc. IEEE*, vol. 103, no. 4, pp. 690–712, Apr. 2015.

[32] T. Someya, B. Pal, J. Huang, and H. E. Katz, "Organic semiconductor devices with enhanced field and environmental responses for novel applications," *MRS Bulletin*, vol. 33, pp. 690–696, July 2008.

[33] T. Someya and T. Sekitani, "Printed skin-like large-area flexible sensors and actuators," *Procedia Chemistry*, vol. 1, no. 1, pp. 9–12, Sept. 2009.

[34] T. Someya, T. Sekitani, S. Iba, Y. Kato, H. Kawaguchi, and T. Sakurai, "A large-area, flexible pressure sensor matrix with organic field-effect transistors for artificial skin applications," *Proc. Nat. Academy of Sciences*, vol. 101, no. 27, pp. 9966–9970, 2004.

[35] J. Sanz-Robinson, L. Huang, T. Moy, W. Rieutort-Louis, Y. Hu, S. Wagner, J. C. Sturm, and N. Verma, "Large-area microphone array for audio source separation based on a hybrid architecture exploiting thin-film electronics and cmos," *IEEE J. Solid-State Circuits*, vol. 51, no. 4, pp. 979–991, Apr. 2016.

[36] Y. Hu, L. Huang, W. S. A. Rieutort-Louis, J. Sanz-Robinson, J. C. Sturm, S. Wagner, and N. Verma, "A self-powered system for large-scale strain sensing by combining cmos ics with large-area electronics," *IEEE J. Solid-State Circuits*, vol. 49, no. 4, pp. 838–850, Apr. 2014.

[37] T. Moy, L. Huang, W. Rieutort-Louis, C. Wu, P. Cuff, S. Wagner, J. C. Sturm, and N. Verma, "An eeg acquisition and biomarker-extraction system using low-noise-amplifier and compressive-sensing circuits based on flexible, thin-film electronics," *IEEE J. Solid-State Circuits*, vol. 52, no. 1, pp. 309–321, Jan. 2017.

[38] T. Giannakopoulos and A. Pikrakis, "Chapter 4 - audio features," in *Introduction to Audio Analysis*, T. Giannakopoulos and A. Pikrakis, Eds. Oxford: Academic Press, 2014, pp. 59 – 103.

[39] K. J. Piczak, "Esc: Dataset for environmental sound classification," in *Proc. 23rd ACM Int. Conf. Multimedia*. New York, NY, USA: ACM, 2015, pp. 1015–1018.

[40] "Flexible hybrid electronics manufacturing innovation institute (FHE-MII)," http://manufacturing.gov/fhe-mii.html.

[41] A. Nathan, A. Ahnood, M. T. Cole, S. Lee, Y. Suzuki, P. Hiralal, F. Bonaccorso, T. Hasan, L. Garcia-Gancedo, A. Dyadyusha, S. Haque, P. Andrew, S. Hofmann, J. Moultrie, D. Chu, A. J. Flewitt, A. C. Ferrari, M. J. Kelly, J. Robertson, G. A. J. Amaratunga, and W. I. Milne, "Flexible electronics: The next ubiquitous platform," *Proc. IEEE*, vol. 100, no. Special Centennial Issue, pp. 1486–1517, May 2012.

[42] N. Palavesam, S. Marin, D. Hemmetzberger, C. Landesberger, K. Bock, and C. Kutter, "Roll-to-roll processing of film substrates for hybrid integrated flexible electronics," *Flexible Printed Electron.*, vol. 3, no. 1, p. 014002, 2018.

[43] "Nextflex," https://www.nextflex.us.

[44] Y. Hu, L. Huang, W. Rieutort-Louis, J. Sanz-Robinson, S. Wagner, J. C. Sturm, and N. Verma, "12.2 3d gesture-sensing system for interactive displays based on extended-range capacitive sensing," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers (ISSCC)*, Feb. 2014, pp. 212–213.

[45] Trimble, Inc., "SketchUp Make." [Online]. Available: https://www.sketchup.com

[46] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[47] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Advances in Neural Inform. Process. Syst.*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105.

[48] F. Chollet *et al.*, "Keras," https://keras.io, 2015.

[49] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. [Online]. Available: https://www.tensorflow.org/

[50] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," in *Proc. Advances in Neural Inform. Process. Syst.*, Y. Weiss, B. Schölkopf, and J. C. Platt, Eds. MIT Press, 2006, pp. 507–514.

[51] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification (2nd Edition)*. New York, NY, USA: Wiley-Interscience, 2000.

[52] Q. Gu, Z. Li, and J. Han, "Generalized Fisher Score for Feature Selection," *ArXiv e-prints*, Feb. 2012.

[53] K. D. Feuz and D. J. Cook, "Transfer learning across feature-rich heterogeneous feature spaces via feature-space remapping (fsr)," *ACM Trans. Intell. Syst. Technol.*, vol. 6, no. 1, pp. 3:1–3:27, Mar. 2015.

[54] Y. Aytar and A. Zisserman, "Tabula rasa: Model transfer for object category detection," in *Int. Conf. Comput. Vision*, Nov. 2011, pp. 2252–2259.

Murat Ozatay (S'17) received the B.Sc. degree in electrical and electronics engineering from Middle East Technical University, Ankara, Turkey in 2015, and the M.A. degree in electrical engineering from Princeton University, Princeton, NJ in 2017, where he is currently pursuing his Ph.D. degree.

He is a member of Verma Lab. His research focuses on bringing together algorithms and insights for learning with technologies and systems for advanced sensing. His primary research interests include machine learning, artificial intelligence, Internet-of-Things, and the design of VLSI systems.

Naveen Verma (M'09) received the B.A.Sc. degree in electrical and computer engineering from the UBC, Vancouver, Canada in 2003, and the M.S. and Ph.D. degrees in electrical engineering from MIT in 2005 and 2009, respectively.

Since July 2009, he has been with the Department of Electrical Engineering at Princeton University, where he is currently a Professor. His research focuses on advanced sensing systems, exploring how systems for learning, inference, and action planning can be enhanced by algorithms that exploit new sensing and computing technologies. This includes research on large-area, flexible sensors, energy-efficient statistical-computing architectures and circuits, and machine-learning and statistical-signal-processing algorithms.

Prof. Verma has served as a Distinguished Lecturer of the IEEE Solid-State Circuits Society, and currently serves on the technical program committees for ISSCC, VLSI Symp., DATE, and IEEE Signal-Processing Society (DISPS). Prof. Verma is recipient or co-recipient of the 2006 DAC/ISSCC Student Design Contest Award, 2008 ISSCC Jack Kilby Paper Award, 2012 Alfred Rheinstein Junior Faculty Award, 2013 NSF CAREER Award, 2013 Intel Early Career Award, 2013 Walter C. Johnson Prize for Teaching Excellence, 2013 VLSI Symp. Best Student Paper Award, 2014 AFOSR Young Investigator Award, 2015 Princeton Engineering Council Excellence in Teaching Award, and 2015 IEEE Trans. CPMT Best Paper Award.