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/TMM.2017.2726187, IEEE Transactions on Multimedia

Deep Temporal Multimodal Fusion for Medical Procedure Monitoring using Wearable Sensors

Edgar A. Bernal*[1], Xitong Yang*[2], Qun Li*[3], Jayant Kumar[4], Sriganesh Madhvanath*[5], Palghat Ramesh[4] and Raja Bala[4]

[1]United Technologies Research Center [2]University of Maryland, College Park [3]Microsoft Corporation [4]PARC, A Xerox Company [5]Conduent Labs

***Abstract*—Process monitoring and verification have a wide range of uses in the medical and healthcare fields. Currently, such tasks are often carried out by a trained specialist, which makes them expensive, inefficient, and time-consuming. Recent advances in automated video- and multimodal-data-based action and activity recognition have made it possible to reduce the extent of manual intervention required to effectively carry out process supervision tasks. In this paper, we propose algorithms for automated egocentric human action and activity recognition from multimodal data, with a target application of monitoring and assisting a user perform a multi-step medical procedure. We propose a supervised deep multimodal fusion framework that relies on concurrent processing of motion data acquired with wearable sensors and video data acquired with an egocentric or body-mounted camera. We demonstrate the effectiveness of the algorithm on a public multimodal dataset and conclude that automated process monitoring via the use of multiple heterogeneous sensors is a viable alternative to its manual counterpart. Furthermore, we demonstrate that the application of previously proposed adaptive sampling schemes to the video processing branch of the multimodal framework results in significant performance improvements.**

***Index Terms*—Wearable sensors, medical procedures, action and activity recognition, multimodal fusion, deep learning, hand localization, egocentric vision, deep temporal fusion.**

I. Introduction

EFFECTIVE process monitoring and validation are critical to ensuring the quality of a wide variety of healthcare and pharmaceutical procedures. For instance, there exist numerous protocols for self-administration of medications including insulin, epinephrine and bronchodylators associated with treatments whose success largely relies on the degree of process compliance exhibited by the patient. The negative implications of this common practice are manyfold: some studies show that diabetic patients have an average theoretical knowledge of the self-insulin injection guidelines as laid out by the American Diabetics Association (ADA) [4][3] of under 70% [69]; other research shows that once the patients are left to carry out the procedure without supervision, their actual self-injection protocol performance scores are in the 60% range [67]. Another healthcare field that benefits from strict adherence to established guidelines is pharmacy compounding, which comprises processes involving the preparation of personalized medications for patients by pharmacy technicians. In this scenario too, verification of the degree of adherence to the guidelines is usually performed manually by direct inspection of the end product by a pharmacist. This time-consuming process requires the pharmacist entering and exiting the clean room, which may in turn lead to medication errors [10]. Less intrusive monitoring methodologies based on offline review of video and imagery of the compounding process have been proposed [42]. However, such approaches still require a large degree of human intervention, and cannot correct in real-time a serious error made by the person carrying out the task.

Recent advances in automated video- and multimodal-data-based action and activity recognition have made it possible to reduce the extent of manual intervention required to effectively carry out process supervision tasks [14][39][40][53][30][57][12]. Wearable cameras enjoy the benefit of continuously monitoring the user's activities on-the-go from an intimate perspective. For example, *Microsoft Hololens*, *Google Glass* and *Taser* body cams allow the recording of the surrounding environment from a perspective that closely resembles that of the user. An added advantage of wearable video acquisition devices is that they often come paired with or can be seamlessly tethered to an increasing variety of wearable sensors that are capable of collecting data describing the behavior and actions of the user in a richer manner than what is possible with video data alone. For instance, smart watches from *Apple* and *Samsung* are equipped with accelerometers, gyroscopes and compasses. Action classification from multimodal first-person data is a relatively new area of research that is beginning to gain attention [64].

In this paper, we focus on the problem of egocentric human action and activity recognition from multimodal data, with a target application of monitoring and assisting a user perform a multi-step task. While we illustrate the proposed methodology in the context of a specific activity, namely, insulin self-injection, we point out that the algorithms are general enough to be applicable to monitoring and assistance of other multi-step tasks. Joint processing of multimodal data acquired by simultaneous use of video and motion sensors can lead to a decrease in uncertainty in decision-making tasks relying on the acquired data, particularly when compared to scenarios where only one data modality is available. The synergistic processing of multiple types of data in support of decision-making processes is termed multimodal data fusion [5], and a

*Work carried out while at PARC, a Xerox Company

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/TMM.2017.2726187, IEEE Transactions on Multimedia

variety of approaches including early and late fusion schemes have been proposed. We believe that existing fusion schemes may fail to effectively model temporal dependencies across multimodal data that are inherently temporal in nature, such as video, audio, and motion sensor data. While temporal models for data fusion have been proposed in the past (see *e.g.*, [64]), they have traditionally relied on memoryless or short-term memory frameworks which fail to learn long-term cross-modal dependencies from extended experience where temporal gaps between significant events are highly variable.

The main contributions of this paper can be summarized as follows:

- A deep, supervised end-to-end hierarchical multimodal data fusion scheme for egocentric action classification using data from wearable sensors. The approach exploits temporal dependencies across time-varying sequences corresponding to the different data modalities. Our scheme is a type of early fusion in that the fusion occurs prior to assigning class labels; however one that attempts to explicitly capture temporal sequence behavior and correlations. The fusion is hierarchical in the sense that correlation between features in each data modality is minimized before the fusion takes place.
- Extensive experimental validation that demonstrates that the proposed approach significantly boosts action classification performance relative to unimodal and state-of-the-art fusion approaches on a multimodal Insulin Self-Injection (ISI) Dataset [35].
- Empirical demonstration of improved action classification performance brought about by the incorporation to the proposed approach of the hand localization and efficient video sampling algorithms first introduced in [35]. The motivation behind the adaptive sampling approach is that regions of high saliency in the video, and in particular those in the neighborhood of where the user's hands are located, are more relevant to action classification tasks than other regions.

The proposed framework relies on a novel end-to-end deep learning architecture, where the outputs of individually optimized unimodal representation branches utilizing Convolutional Neural Networks (CNNs) are temporally fused with a Long-Short-Term Memory (LSTM) network [27]. In contrast to previously proposed deep-learning-based multimodal fusion schemes which rely on atemporal architectures, the proposed approach is capable of learning high-level temporal correlations among the different data modalities via the use of the LSTM network. We believe this is the first end-to-end deep multimodal fusion framework proposed in the literature for a process monitoring application, where both the feature extraction and the multimodal data fusion are effected using a deep learning network. Consequently, an advantage of our proposed approach relative to existing systems is that the time-consuming and laborious fine-tuning of features to improve robustness is obviated. While there exist other deep frameworks for multimodal feature learning, many of them are unsupervised in nature [66][47][65]; as a majority of recent empirical evidence suggests, learning features in a task-oriented, supervised manner based on a discriminative objective results in better classification performance [16].

This paper is organized as follows: Sec. II gives an overview of the existing literature on related research topics; Sec. III describes the proposed deep end-to-end hierarchical multimodal data fusion scheme for egocentric action classification; Sec. IV provides the experimental validation of the proposed methods; lastly, we conclude and provide a brief discussion on implementational considerations and future work in Sec. V.

II. Related Work

A. Video-Based Action and Activity Recognition

The majority of the literature on video-based activity recognition describes methods that rely on third-person camera views; for a detailed review of algorithms relying on shallow architectures and hand-engineered features, including an overview of existing datasets, we refer the reader to [12]. More recently, deep architectures that perform unsupervised feature learning for action recognition and event detection from video streams have been proposed [41][11][71]. Supervised deep frameworks based on feedforward networks have also been proposed [61][29][21]. All these models, being static in nature, have been shown to have less than optimal performance, comparable to that achieved by methods relying on naïve frame-by-frame processing [29]. Temporal deep models that explicitly exploit the dynamic nature of the video data have also been proposed, most of them based on Recurrent Neural Networks (RNNs). Of particular relevance is the method from [7] which combines CNNs and RNNs to achieve human action recognition; the method, however, performs recognition based solely on video streams.

Interest in analysis of egocentric video data [28] with applications including action and activity recognition [56][18][74][19][51], object recognition and scene understanding [20][55][54][43], and event summarization [44][76][35] has recently surged. In order to address the challenges brought about by the limited computational power as well as the inherent ego-motion associated with head- or body-mounted cameras, much of the work addressing egocentric-video-based action and activity recognition exploits the significance of hand positions [35], hand-object interactions [70][46][18], and user attention [45] and gaze [19][48]. Closely related to aspects of our work is that of [74] where saliency models are used as filters to the sampling process in a standard action recognition pipeline. A variety of sampling schemes are proposed, including biologically inspired masks, an analytical mask based on structure tensor, and an empirical mask based on eye-tracking data. A significant body of existing work relies solely on video data as a singe data source. As mentioned earlier, [64] proposes a framework for action recognition that relies on multiple modalities of data; however, not all of the video on which the method relies is egocentric in nature, which makes it less amenable to analysis on-the-go.

B. Healthcare and Medical Applications of Egocentric Video

In contrast to traditional applications of ubiquitous healthcare which mainly monitor health status, video data collected

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/TMM.2017.2726187, IEEE Transactions on Multimedia

by wearable cameras provides a vast amount of temporally and spatially continuous information which enables monitoring of complex healthcare procedures. Applications are numerous, ranging from daily living assistance for sick, impaired or elderly citizens, to training and compliance monitoring in medical procedures. Recognizing the relationship between lifestyle and health conditions, life-logging-based approaches to recognizing sedentary behavior [31], analyzing dietary practices [49], and studying the effectiveness of egocentric video as a memory aid [60] have been proposed; none of these approaches, however, involve automated analysis of the acquired video. In contrast, the authors of [63] study the problem of automatically issuing reminders about actions that users might forget.

C. Wearable Sensors in Healthcare

Recent technological advances in wearable-sensor-based Wireless Sensor Networks (WSN) have enabled the design of low-cost, intelligent, and lightweight medical sensor nodes that can be strategically placed on the human body to monitor various physiological vital signs for extended periods of time and provide real-time feedback to the user and medical staff. WSNs have been widely used for emergency response, industrial automation, military surveillance, and environmental and agricultural monitoring, with applications in healthcare being the most promising [52]. In contrast to traditional sensor networks that are carefully designed and deployed predeterminately, WSNs can be deployed in an ad-hoc manner leading to improved robustness, efficiency and increase in spatial coverage, as well as improved patient comfort [73]. WSNs enable home healthcare, also known as remote healthcare [1], or ubiquitous healthcare (u-healthcare) [22], and the closely related field of mobile health (m-health) [8][22]. Ubiquitous healthcare is an emerging technology whereby patients can monitor their health without visiting the hospital or clinic; at the same time, hospitals can provide patients with efficient medical services through computerized medical information and resources.

D. Multimodal Data Fusion

Multimodal fusion refers to the integration of multiple data modalities, their associated features, or the intermediate decisions in order to perform an analysis task [5]. Generally speaking, there are two types of fusion depending on the level at which the fusion takes place: early or feature-level fusion and late or decision-level fusion. In naïve early fusion schemes, features from different modalities are concatenated before the learning takes place. The Multiple Kernel Learning (MKL) method [37] initially proposed as a systematic approach to combine features in the context of multi-class classification tasks, has been recently extended to support multimodal inputs [38][79] and can be thought of as a more elaborate form of early fusion. In contrast, late fusion schemes learn a separate model for each data modality and combine the per-modality decisions to achieve a consolidated decision. Recently, recognizing that heterogeneous data leads to decisions with possibly diverse levels of confidence, the authors of [77] proposed a method that achieves feature scale invariance in the presence of unnormalized decision vectors across modalities. For more extensive treatments on multimodal data fusion, we refer the reader to [5] and [36]. Given the scope of the paper, the focus of the remainder of this review will be on literature related to multimodal fusion of data obtained with wearable sensors as well as on deep learning approaches to fusion.

Most of the existing work in the area of wearable data fusion focuses on healthcare applications such as remote patient monitoring [6], and fall detection [78]. Previous efforts have explored the fusion of motion data with data from specialized sensors including hand tracking data from an ultrasonic positioning system in maintenance tasks [68] and depth data from time of flight sensors to detect falls [24]. Fusion of egocentric video and motion sensor data has been studied in the context of localization [32][33] and navigation [2]. More closely related to our work is [64] which proposes a method for automatic segmentation and classification of human action from multiple sources of video including third-person RGB and IR cameras and a wearable camera, audio from five directional microphones, and data from multiple wearable units including accelerometers, gyroscopes, magnetometers, and thermometers, among others. The multiple modalities are fused in different ways with an atemporal model based on k-Nearest Neighbors outperforming the other fusion approaches including a temporal framework based on Hidden Markov Models (HMMs). Deep learning based fusion approaches have also been proposed in the literature, however most of them rely on atemporal deep architectures such as Deep Belief Networks [66], Deep Autoencoders [47], and Deep Boltzmann Machines [65]. Also, while fusion is typically performed with a deep network, hand-engineered features are usually extracted from the data streams [65][62][47][17]. The authors of [17] use an LSTM in a fusion framework to perform classification of non-linguistic vocalization. They too extract hand-engineered shape and appearance features from the video signal and low-level features from the audio signal, which are concatenated before being fed to the LSTM for classification. In contrast, our proposed temporal fusion approach extracts deep features from each data modality in a fully automated manner, and performs the fusion at an intermediate layer of the LSTM for improved shared representation learning across data modalities. It is consequently more robust to high data dimensionality and can capture long-term temporal trends and correlations across the different data streams. A recent example of a deep multimodal data fusion framework for action recognition that does not rely on hand-engineered features is [50]; although the proposed method includes temporal modeling of the data, the fusion is performed with atemporal feedforward networks, more specifically CNNs, so the temporal modeling only involves the already fused representation. Lastly, and although not used for fusion, the deep framework for action recognition proposed in [15] shares commonalities with the video feature extraction stage in our approach since both approaches cascade a CNN for visual extraction and an LSTM for video sequence learning.

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/TMM.2017.2726187, IEEE Transactions on Multimedia

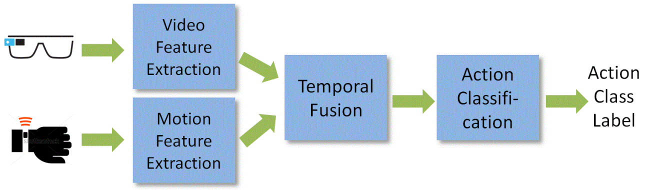


Fig. 1. High-level overview of proposed multimodal fusion framework.

III. Multimodal-Data-Based Action Recognition

We propose a novel end-to-end deep learning framework for temporal data fusion to recognize human actions and activities by combining multiple sensing modalities. In this paper, we have applied the proposed framework to the fusion of egocentric video with data from a wearable wrist sensor corresponding to a set of actions of interest. We do not make any assumptions regarding the duration of the actions or the lengths of the video and sensor data segments. In other words, these may be of variable length. Given input sequences from two modalities, we first extract clip-level features (sequence representation) and minimize the correlation between features within each modality. Learning features with reduced within-modality correlation will lead to better learning of features that exploit cross-modality correlations [62]. Then a fused state (joint representation) is learned to minimize the action classification error. Fig. 1 includes a high-level overview of the proposed approach in the form of a block diagram. We now proceed to describe each of the illustrated steps in detail.

A. Video Data Feature Extraction

The goal of this step is to compute concise representations of the input egocentric video data that are amenable to the action classification task. We provide two alternatives to this step. In both cases, feature extraction is accomplished by using a two-step process inspired by the long-term recurrent convolutional network (LRCN) framework first proposed in [15]. In the first step, purely visual features are extracted from individual video frames using a deep convolutional network (CNN) as the one introduced in [34], where the features are the 4096-dimensional vectors corresponding to the activation of the last hidden layer in the network. The second step aggregates the visual features temporally across multiple frames using a long-short-term memory (LSTM) network. To this end, incoming videos are partitioned into 32-frame clips with a 16-frame temporal stride. Visual features are temporally aggregated across the frames in a clip with an LSTM network with 256 hidden states. The 256-dimensional vector corresponding to the hidden state of the LSTM at the end of each clip serves as the clip sequence representation. The weights of the networks involved are determined via an offline supervised training procedure aimed at classification of labeled training samples from the dataset of interest.

The differences between the two alternative approaches to the video data feature extraction lie on the type of sampling implemented. In the first variant, the full video frame is fed to the two-stage feature extraction process. We refer to this exhaustive sampling method of the incoming video as Dense Sampling (DS). In the second approach, a variation on one of the adaptive frame sampling strategies proposed in our previous work [35] is introduced. The assumption is that for activities involving hand-eye coordination, the location and motion of the user's hands provide critical cues on the action being performed. The approach operates as follows: in an offline stage, a customized hand detector is trained based on color data from pixels identified as belonging to the hand in an initial video segment capturing a predetermined hand gesture. The *a priori* knowledge about the nature of the hand gesture is exploited to accurately detect hand pixels throughout the duration of the gesture sequence. Once trained, the detector can be used to perform pixel-level hand localization in an on-line stage as new video is being processed. For every incoming video frame, hand detection is first performed at the pixel level to produce a binary mask with active pixels corresponding to locations identified as skin pixels. The centroid pixel location $\mathbf{C}$ of the binary hand mask is then computed. For each frame for which a hand centroid is computed, a rectangular sub-window centered around $\mathbf{C}$ and of size 160×200 pixels is cropped from the video frame and fed to the two-stage feature extraction process. Note that this approach is similar to the Hand-based Adaptive Sampling (HAS) scheme proposed in [35] is introduced, except that the original HAS scheme also performs soft sampling, or selective feature extraction, on the sub-window centered at the hand location. For frames for which no hand is detected, linear interpolation between temporal neighboring hand centroid locations is performed in order to maintain the frame rate of the incoming video stream, which is necessary to preserve synchronization between the multiple data modalities.

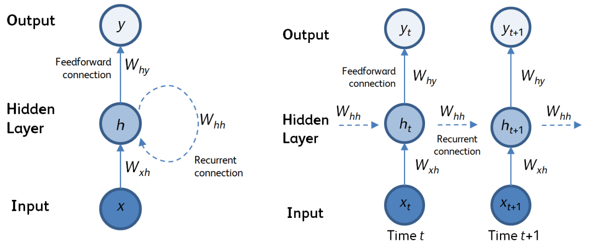
B. Motion Data Feature Extraction

In order to compute concise representations of the input motion data that are amenable to the action classification task, the data in each motion channel (three acceleration and four orientation channels are available) is first smoothed by applying temporal median filtering and normalized independently by subtracting the mean and scaling by the standard deviation. The resulting sequences are fed to a one-dimensional CNN with two convolutional layers and two fully connected layers. As in the video feature extraction step, data corresponding to a 32-frame clip is concatenated before being passed to the CNN, which means that the 32×7 matrix is input to the CNN for each clip since seven data channels are available. Each of the rows of this matrix is convolved with the 1D kernels along the temporal dimension. The last hidden layer of the CNN yields a 252-dimensional vector that is used as the feature representing the sensor data for each motion stream corresponding to a 32-frame time period. As before, the weights of the CNN are determined via an offline supervised training procedure aimed at classification of labeled training data from the dataset of interest. Prior to feeding the output of sequence representation of the motion data stream to the temporal fusion stage, the motion and video data sequence representations are temporally aligned.

C. Temporal Fusion

As stated, temporal fusion is achieved via the application of an LSTM network to the visual and motion features extracted from overlapping 32-frame clips. Before describing the proposed architecture, we provide a brief overview of the theory behind recurrent neural networks.

Recurrent Neural Networks: RNNs store and update an internal state that has information about the past behavior of the network, which makes them more suitable to model sequential data than traditional feedforward networks. During the forward pass, for each point in the input sequence, feedforward connections convey current activation information while recurrent connections transmit information from activations in the previous time step. This *modus operandi* leads to a dual representation of a recurrent neural network, as illustrated in Fig. 2. In the figure, x_t, h_t and y_t denote the input, state and output values at time t, respectively, and W_{ij} denotes the weights in the connections between layer i and j, where the layers can be the input, hidden or output layer. In the figures, the continuous-line arrows illustrate traditional feedforward connections, while the dotted-line arrows illustrate recurrent connections between the various RNN elements. The folded view of an RNN (see Fig. 2(a)) is a compact representation of a network whose temporal depth may be arbitrarily large since it is determined by the length of the input sequence. In contrast, the unfolded view (see Fig. 2(b)) explicitly depicts the temporal structure of the network. Specifically, given an input sequence $x_1,\ldots,x_T$, a traditional RNN computes an output sequence $y_1,\ldots,y_T$ by maintaining an internal state sequence $h_1,\ldots,h_T$ according to the expressions $h_t = \sigma(W_{xh}x_t + W_{hh}h_{t-1})$ and $y_t = W_{hy}h_t$, where $\sigma(\cdot)$ denotes the non-linear activation function of the neurons in the hidden layer. Note that these expressions assume that there are no bias units present in the network.



(a) (b)

Fig. 2. (a) Folded and (b) unfolded views of an RNN.

LSTMs are a type of RNN in that they contain recurrent connections in addition to feedforward connections. In contrast to traditional RNNs which are comprised of neurons, an LSTM network is composed of layers of memory blocks, with possibly various degrees of connectivity among themselves. A memory block contains cells with self-connections, which maintain and update an internal state c_t in a process dictated by the actions of gates, as illustrated in Fig. 3. The input gate modulates the influence of the input x_t on the cell state; the output gate modulates the influence of the cell state on the output of the memory block h_t; and the forget gate determines the degree to which the current cell state is maintained across time. The gate activation functions are usually sigmoid functions, while the input activation function is usually a tanh. No activation function is used within the cell. Unlike traditional RNNs, which are often unable to learn long-term dependencies in the data due to vanishing and exploding gradients [9], LSTMs have been shown to be able to reason from data spanning extended periods of time [23] thanks to the action of the multiplicative gates. The input to output mapping in a memory block takes place according to the following expressions:

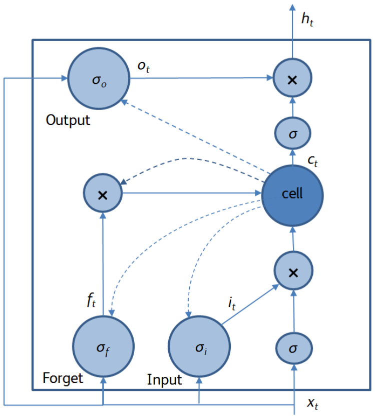


Fig. 3. Schematic representation of a memory block with a single memory cell.

$$i_t = \sigma_i(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1}) \tag{1}$$

$$f_t = \sigma_f(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1}) \tag{2}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \varphi(W_{xc}x_t + W_{hc}h_{t-1}) \tag{3}$$

$$o_t = \sigma_o(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t) \tag{4}$$

$$h_t = o_t \odot \varphi(c_t) \tag{5}$$

where i_t, o_t and f_t denote the outputs of the input, output and forget gates, respectively, $\sigma_i(\cdot)$, $\sigma_o(\cdot)$ and $\sigma_f(\cdot)$ denote the activation functions of the input, output and forget gates, respectively, $\varphi(\cdot)$ denotes a non-linear function, usually a tanh, the operator $\odot$ denotes element-wise multiplication, and W_{jk} denotes the weights in the connections between elements j and k in the memory block. Again, for simplicity it is assumed that no bias units are present in the memory block.

Proposed Network Architecture: While deep feedforward networks aimed at learning various levels of feature hierarchies have been used successfully since the mid-2000s in a wide range of tasks including vision and natural language processing, deep RNNs have only been recently proposed in the field of speech recognition [26][25][58]. It has been shown empirically that deep networks are better at learning efficient multimodal representations which capture high-level associations between data modalities [62]. This is especially true when the cross-modality learning takes place at deep layers placed on top of layers that learn per-modality representations; this is because, in such configurations, early layers may eliminate within-modality patterns that would otherwise become dominant in shallow fusion schemes given that those patterns are often stronger than cross-modality correlations. We note that most of the existing work dealing with deep fusion relies on atemporal networks. We propose a deep temporal fusion scheme with early recurrent layers devoted to the learning of within-modality representations, and deep recurrent layers devoted to joint modality learning. For illustration purposes, consider the three different LSTM network architectures illustrated in Fig. 4; for compactness, only folded views are included. The LSTM network in Fig. 4(a) has one fully connected hidden layer with two memory blocks. An LSTM of this type would be capable of modeling the temporal behavior of the input signal and producing an output signal that is a function of the temporal behavior of the input signal. The LSTM in Fig. 4(b) has one partially connected hidden layer with four memory blocks. An LSTM of this type would be capable of independently modeling the temporal behavior of two input signals, and producing an output signal that is a function of the independent temporal behavior of the signals. Lastly, the LSTM in Fig. 4(c) has one partially connected hidden layer with four memory blocks, and one fully connected hidden layer with two memory blocks. The first layer of this LSTM would be capable of independently modeling the temporal behavior of two input signals, while the second layer would be capable of jointly modeling the temporal behavior of the signals; consequently, an LSTM of this type would be able to produce an output signal that is a function of the joint temporal behavior of the signals. This hierarchical fusion method enables the model to better capture temporal patterns across data modalities since the activations of the first hidden layer are devoid of within-modality feature correlations which, when present, may prevent the learning of cross-modality correlations.

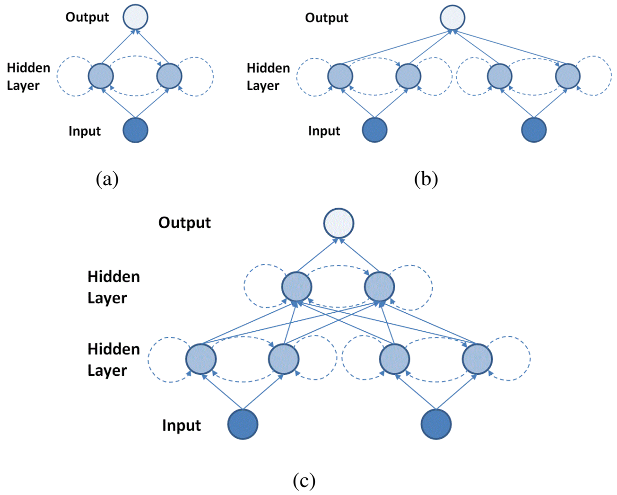


Fig. 4. Different LSTM architectures.

The LSTM network in the proposed system has two hidden layers, one that is fully connected for each modality, and one that is fully connected across modalities, as illustrated at a high-level in Fig. 5; consequently, the architecture of the LSTM used in the fusion process is similar to the architecture illustrated in Fig. 4(c). The first hidden layer has 128 memory blocks per modality, and it maps the input features (256- and 252-dimensional for video and motion, respectively) to 128-dimensional activations. The second hidden layer has 128 memory blocks and is fully connected to the first layer. Intuitively speaking, the first hidden layer of the temporal fusion network learns temporal trends present in each modality independently and minimizes within-modality correlations, and the second hidden layer learns temporal correlations across modalities and effectively performs temporal data fusion. We refer to the latter as the multimodal hidden layer. This approach is in contrast with that of [17] where features are concatenated before being input to the LSTM network and can be considered an example of an early fusion scheme as in Fig. 4(a), where the input to the LSTM combines both video and motion features.

Formally, the temporal fusion stage in the proposed approach operates as follows: let x_t^{vid} and x_t^{mot} denote the video and motion feature representation vectors for the clip at time t, and let h_t^{fus} denote the temporal fusion feature at time t. We next describe how the mapping between inputs x_t^{vid} and x_t^{mot} and output h_t^{fus} takes place in the network. The per-modality branches of the LSTM take the feature representations of each modality, x_t^{vid} and x_t^{mot}, and produce temporal representations h_t^{vid} and h_t^{mot} according to:

$$i_t^{mod} = \sigma(W_{xi}^{mod} x_t^{mod} + W_{hi}^{mod} h_{t-1}^{mod} + W_{ci}^{mod} c_{t-1}^{mod} + b_i^{mod}) \tag{6}$$

$$f_t^{mod} = \sigma(W_{xf}^{mod} x_t^{mod} + W_{hf}^{mod} h_{t-1}^{mod} + W_{cf}^{mod} c_{t-1}^{mod} + b_f^{mod}) \tag{7}$$

$$c_t^{mod} = f_t^{mod} \odot c_{t-1}^{mod} + i_t^{mod} \odot \varphi(W_{xc}^{mod} x_t^{mod} + W_{hc}^{mod} h_{t-1}^{mod} + b_c^{mod}) \tag{8}$$

$$o_t^{mod} = \sigma(W_{xo}^{mod} x_t^{mod} + W_{ho}^{mod} h_{t-1}^{mod} + W_{co}^{mod} c_{t-1}^{mod} + b_o^{mod}) \tag{9}$$

$$h_t^{mod} = o_t^{mod} \odot \varphi(c_t^{mod}) \tag{10}$$

where $mod \in \{vid, mot\}$ denotes the modality of the branch and h_t^{mod} are the output of each branch, or equivalently, the per-modality representations at time t. In Eqs. 6-10, c_t^{mod} denotes the internal state, i_t^{mod}, o_t^{mod} and f_t^{mod} denote the outputs of the input, output and forget gates, respectively; b_c^{mod}, b_i^{mod}, b_o^{mod} and b_f^{mod} denote the bias of the gate cell and the input, output and forget gates, respectively; $\sigma(\cdot)$ denotes the logistic activation function, $\varphi(\cdot)$ denotes a non-linear function, usually a tanh, and W_{jk}^{mod} denotes the weights in the connections between elements j and k in the memory block in the branch corresponding to modality mod. Note that in Eqs. 6-10, it has been assumed that $\sigma_i(\cdot)$, $\sigma_o(\cdot)$ and $\sigma_f(\cdot)$ from Eqs. 1-5 are all the same function, namely $\sigma(\cdot)$.

The fusion layer of the LSTM network takes in the concatenated activations of each of the per-modality branches in the first hidden layer and produces the fused representation

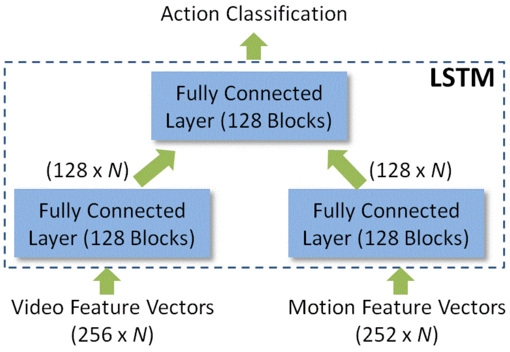

Action Classification
LSTM
Fully Connected Layer (128 Blocks)
(128 x N)
(128 x N)
Fully Connected Layer (128 Blocks)
Fully Connected Layer (128 Blocks)
Video Feature Vectors (256 x N)
Motion Feature Vectors (252 x N)

Fig. 5. Temporal fusion via a multi-layer LSTM.

h_t^{fus}. Specifically, let $x_t^{vidmot} \triangleq h_t^{vid} \| h_t^{mot}$ denote the concatenated activation vector; then h_t^{fus} is obtained according to the following mappings effected in the fusion layer:

$$i_t^{fus} = \sigma(W_{xi}x_t^{vidmot} + W_{hi}h_{t-1}^{fus} + W_{ci}c_{t-1}^{fus} + b_i) \tag{11}$$

$$f_t^{fus} = \sigma(W_{xf}x_t^{vidmot} + W_{hf}h_{t-1}^{fus} + W_{cf}c_{t-1}^{fus} + b_f) \tag{12}$$

$$c_t^{fus} = f_t^{fus} \odot c_{t-1}^{fus} + i_t^{fus} \odot \varphi(W_{xc}x_t^{vidmot} + W_{hc}h_{t-1}^{fus} + b_c) \tag{13}$$

$$o_t^{fus} = \sigma(W_{xo}x_t^{vidmot} + W_{ho}h_{t-1}^{fus} + W_{co}c_{t-1}^{fus} + b_o) \tag{14}$$

$$h_t^{fus} = o_t^{fus} \odot \varphi(c_t^{fus}) \tag{15}$$

where c_t^{fus} denotes the internal state, i_t^{fus}, o_t^{fus} and f_t^{fus} denote the outputs of the input, output and forget gates, respectively; b_c, b_i, b_o and b_f denote the bias of the gate cell and the input, output and forget gates, respectively; and W_{jk} denotes the weights in the connections between elements j and k in the memory block. As before, it has been assumed that $\sigma_i(\cdot)$, $\sigma_o(\cdot)$ and $\sigma_f(\cdot)$ from Eqs. 1-5 are all the same function, namely $\sigma(\cdot)$.

Complexity Analysis: Let N_i^{vid} and N_i^{mot} denote the length of the input video and motion features, respectively, N_c the number of classes in the classification task, and N_h^{vid}, N_h^{mot} and N_h^{fus} the number of memory blocks in the hidden layers for the video, motion and fusion sections of the network. Once trained, the computational complexity of forward pass (at inference) of the fusion framework is linear in the number of connections in the network. There are $O(4[N_i^{vid}N_h^{vid} + N_i^{mot}N_h^{mot}])$ edges between the input layer and both sections of the first hidden or per-modality layer, $O(4[N_h^{vid} + N_h^{mot}]N_h^{fus})$ between the first and the second hidden, or fusion layer, and $O(N_h^{fus}N_c)$ edges between the fusion layer and the output layer. Note that the input to an LSTM memory block affects four different elements in it, hence the constant multiplicative factor in the expressions above. The total number of recurrent connections is $O(4[(N_h^{vid})^2 + (N_h^{mot})^2 + (N_h^{fus})^2])$ and the total number of internal connections within the memory blocks is $O(3[N_h^{vid} + N_h^{mot} + N_h^{fus}])$. Consequently, and ignoring constant factors, the overall complexity of the fusion stage is $O(N_i^{vid}N_h^{vid} + N_i^{mot}N_h^{mot} + [N_h^{vid} + N_h^{mot}]N_h^{fus} + (N_h^{vid})^2 + (N_h^{mot})^2 + (N_h^{fus})^2)$, where we have assumed that $N_c \ll N_i^{vid}, N_i^{mot}, N_h^{vid}, N_h^{mot}, N_h^{fus}$. Furthermore, if $N_k^{mod} = O(N)$ for all $mod \in \{vid, mot, fus\}$ and all $k \in \{i, h\}$, then the complexity of the fusion stage can be expressed as $O(N^2)$.

D. Action Classification

Inference regarding the class to which incoming data streams correspond can be performed in different ways. In the case where the end-to-end deep nature of the framework is desired, the last layer in the LSTM network can be configured as a soft-max layer with as many units as classes in the classification task, N_c. Alternatively, the activations of the last hidden layer of the LSTM network can be used as features that are input to a traditional classifier, such as an SVM. In practice, we found both alternatives to have comparable performance.

IV. Experiments

A. Datasets

The ISI dataset includes video data acquired with a *Google Glass* wearable camera and motion data acquired with an Invensense motion wrist sensor of a number of subjects performing seven actions related to an insulin self-injection activity. The seven steps in self-injection of insulin, each of which is treated as an action class, are:

1) Sanitize hand
2) Roll insulin bottle
3) Pull air into syringe
4) Withdraw insulin
5) Clean injection site
6) Inject insulin
7) Dispose needle

Fig. 6 shows a sample image from the dataset. A total of 8 subjects (4 female, 4 male) with different skin colors and ages simulated performing the self-injection activity. Subjects were instructed only on the sequence of steps to be taken and were not coached on how to perform a given step. Prior experience varied widely. Three locations with different lighting and background conditions were used, and objects and their arrangement, as well as sitting geometry were allowed to vary freely. As evidence of inter-subject variability, durations of the video segments for a given action varied by a factor of 2 or more (e.g., 5s vs. 10s, or 13s vs. 29s); furthermore, optical flow motion analysis reveals that mean motion magnitude varies on average by a factor of 4 and up to a factor of 6 across subjects for a given action. For some of the videos, subjects were asked to exaggerate head and body motion patterns in order to simulate health conditions that can lead to tremors or other uncontrollable movements. A total of 25 different videos were acquired, and each video was manually partitioned into seven segments corresponding to the seven actions in the procedure, for a total of 175 segments. Of the 25 videos, 11 have corresponding motion sensor data.

Fig. 6. Sample video frame from ISI dataset

B. Video-Based Action Recognition – Results

In [35], we demonstrated that performing adaptive sampling on the incoming video by giving preference to areas surrounding the location of the hands effectively improved action classification performance independently of the choice of video feature representation. Imprevements in action recognition accuracy were empirically verified on classifiers based on dense trajectories [75], SIFT3D [59], and stacked convolutional independent subspace analysis (SC-ISA) [41] features. We also studied the dependence of the action recognition performance on the sampling budget, that is, the number of samples involved in the feature extraction process. We observed that adaptive sampling schemes achieved the best classification performance using roughly one third of the total number of descriptors possible.

In order to quantify the computational gain brought about by the proposed sampling methods, in [35] we timed the execution of the standard dense trajectory approach (*i.e.*, with the DS scheme) and the proposed sampling approaches on a 30 fps, 8 second video. In addition to the DS and HAS approaches described in Sec. III-A, we also implemented the Image-Center-based Soft Sampling (ICS) scheme which samples preferentially near the image center, and the Random Sampling (RS) scheme which randomly selects locations across the video frame for feature extraction. Table I contains the results. Execution time was measured in seconds on a Windows 7 machine with 16GBytes of RAM and an Intel i7 2.80GHz processor. The implementation was done in Matlab R2013b.

	HAS	ICS	RS	DS [75]
Time	0.0483	0.0394	0.0374	0.1217

TABLE I
EXECUTION TIMES IN SECONDS (PER FRAME) FOR DIFFERENT SAMPLING SCHEMES.

Note that the performance gains documented in Table I are observed in spite of the asymptotical complexity of the different sampling algorithms being equivalent. To see this, assume that the number of pixels in the image is $O(N)$ and that the feature extraction process has linear complexity $O(N)$ (this can be thought of as a best-case scenario as the complexity of the features considered in [35] is worse than $O(N)$). Then, the complexity of the analysis on the densely sampled input is $O(N^2)$. Assume that the complexity of a random number generator is $O(1)$, which is often the case; then, the complexity of the RS method would be $O(N)+O(kN^2)$ for some constant $0 \leq k \leq 1$ being indicative of the fraction of pixels being considered after the randomized downselection. Asymptotically, this is equivalent to $O(N^2)$. Similarly, the complexity of the ICS and HAS methods would be $O(k_{i1}N)+O(k_{i2}N^2)$ for constants $0 \leq k_{i1}, k_{i2} \leq 1$ where $i \in \{\text{ICS,HAS}\}$, and where k_{i1} is indicative of the size of the sub-window where the sampling takes place and k_{i2} is indicative of the fraction of pixels being considered by each of the methods; again, this complexity is asymptotically is equivalent to $O(N^2)$. Consequently, while all sampling algorithms have asymptotic complexity that can be expressed as $O(N^2)$, there are actual verifiable performance differences in practice that can be captured by measuring execution times on standard equipment.

C. Multimodal-Data-Based Action Recognition – Results

Of the 11 video/motion data pairs corresponding to 77 segment/motion pairs, we use 56 segment/motion data pairs for training and 21 segment/motion data pairs for testing. After performing the sliding window approach described in Sec. III-A for feature extraction, the total number of clip-level data pairs available in the data set is 1431 (or, equivalently, close to 46 thousand video frame-sensor data pairs), with approximately 70% of the data being used for training. We implemented the network in the form of a Sequential model in Keras [13]. In order to avoid potential overfitting issues, a dropout layer with a 50% dropout unit fraction was introduced after each LSTM layer. We measure the classification performance of the different methods in terms of clip- and segment-level mean average precision (mAP). To illustrate the difference between the performance metrics, note that the fundamental data unit that can be processed with the proposed fusion network corresponds to video and motion data within a video clip with a predetermined frame length – 32 frames in our case. Consequently, decisions can be made either at the clip level or at the segment level, where a segment includes multiple clips (the number of which is determined by the length of the segment). For each clip, the network produces a 7-dimensional probability vector, each entry of which denotes the probability that the action contained in the incoming multimodal data stream corresponds to the index of the entry within the vector. In order to obtain segment-level classification accuracy numbers, multiple clip-level outputs are averaged, and the action with the highest average probability entry (averaged across clips in the segment) is selected as the action corresponding to the incoming data stream. We first compare the performance of the framework relative to naïve early and late fusion schemes, and then compare it with that of more advanced fusion schemes.

1) Performance Comparison against Naïve Fusion Schemes: Table II illustrates performance results of the proposed method when compared to single-modality as well as the traditional early and late fusion approaches at both the clip and the segment level. Dense features were extracted from the video data stream. Single-modality results (see rows 1 and 2 in table) were obtained by implementing a linear SVM on the feature representations obtained by each of the described feature extraction branches. In the late fusion case (see row 3), the final decision was computed as

Method		Classification Accuracy (clip)	Classification Accuracy (segment)
Video Only		0.53	0.76
Motion Only		0.44	0.57
Late Fusion	$\omega = 0.25$	0.41	0.67
	$\omega = 0.50$	0.58	0.81
	$\omega = 0.75$	0.56	0.81
Early Fusion		0.63	0.81
Temporal Fusion		**0.89**	**0.95**

TABLE II
CLASSIFICATION RESULTS WITH FUSED DATA AND DENSE SAMPLING

Method		Classification Accuracy (clip)	Classification Accuracy (segment)
Video Only		0.63	0.86
Motion Only		0.44	0.57
Late Fusion	$\omega = 0.25$	0.47	0.81
	$\omega = 0.50$	0.63	0.86
	$\omega = 0.75$	0.65	0.86
Early Fusion		0.70	0.95
Temporal Fusion		**0.90**	**1.00**

TABLE III
CLASSIFICATION RESULTS WITH FUSED DATA AND ADAPTIVE SAMPLING

$c_{out} = \omega \cdot c_{vid} + (1-\omega) \cdot c_{mot}$, where $0 \leq \omega \leq 1$ denotes the fusion coefficient, c_{out} denotes the fused decision, and c_{vid} and c_{mot} denote the decision of the video and the motion data processing branches used in the single-modality approach, respectively. Three different values for ω were tested, as indicated in the table. In the early fusion case (see row 4), features extracted in the video and motion data extraction stages were concatenated and an SVM classifier trained based on the resulting features.

Table III illustrates performance results of the proposed method when adaptive sampling is applied to the video data feature extraction process. The benefits of the adaptive sampling technique are apparent, with significant improvements in performance across the board, whenever video data processing is involved in the decision-making process.

2) Performance Comparison against State-of-the-Art Fusion Schemes: Multiple Kernel Learning (MKL) was initially introduced as an extension of the single-kernel SVM to incorporate multiple kernels in the classification task [37]. More recently, MKL has been shown to outperform traditional early fusion frameworks [38][79]. The authors of [79] introduced several variants of MKL in the context of a classification task with multimodal data. The different versions of MKL stem from the type of norm used in the optimization of the fused kernel: the ℓ_1- and ℓ_∞-norm regularization formulations result in solutions with sparse kernel coefficients, while the solution to the ℓ_2-norm regularization formulation is smooth. Out of all the MKL variants proposed, we found the performance of the ℓ_2 version to be the most effective. We also implemented the unimodal MKL framework from [72] on each of the modalities and performed linear late fusion in order to compare the effectiveness of the MKL approach at the different stages of the classification process. Table IV shows the results, obtained from analysis of features extracted with the adaptive sampling scheme. It can be seen that, while the clip-level results are comparable for both methods (see rows 1 and 2), the early MKL fusion framework achieves better aggregate segment-level results, which indicates that the confidence of the decisions made based on learning from the combined modalities is higher than that achieved by combining the individual decisions. It can also be seen that neither of the fusion techniques is on par with the proposed approach (see row 3).

Method	Classification Accuracy (clip)	Classification Accuracy (segment)
MKL (Early) [79]	0.71	0.95
MKL (Late) [72]	0.71	0.86
Temporal Fusion	**0.90**	**1.00**

TABLE IV
CLASSIFICATION RESULTS WITH MKL FUSED DATA AND ADAPTIVE SAMPLING

As stated, one of the limitations of late fusion is that the confidence of the output of the different classifiers may be largely dissimilar across modalities. The authors of [77] formulate the late fusion problem as a rank minimization task that alleviates that limitation, particularly when the decisions of the individual classifiers are somewhat consistent. To this end, the confidence score vectors obtained by each classifier are converted into a pairwise relationship matrix, each entry of which describes the relative confidence between the scores of two test samples. Then, leveraging the assumption that decisions across classifiers are somewhat consistent, each relationship matrix is decomposed into two terms, one rank-2 decision matrix that is common across all classifiers, and a sparse matrix containing classification errors. The classification scores are recovered from the reconstructed low-rank decision matrix. We applied the rank-minimization late fusion scheme to decision scores from classifiers trained on features extracted with the adaptive sampling scheme, and recorded the results in Table V (see row 1). It can be seen that, while the clip-level performance is not particularly good, the segment-level performance is (comparatively) strong, particularly in the context of the quality of the per-clip decisions. We also evaluated the performance of the rank-minimization algorithm applied to decision vectors obtained with the unimodal MKL algorithm from [72]; Table V (see row 2) shows the results. This strategy outperforms rank-minimization at the clip level but does not perform as well at the segment level. Lastly, an empirical upper bound for linear late fusion schemes can be determined by assuming the optimal algorithm will be able to find, for each set of sample-level decisions $c_{vid}^{(i)}$ and $c_{mot}^{(i)}$, a value for $\omega^{(i)}$ that will result in $c_{out}^{(i)} = \omega \cdot c_{vid}^{(i)} + (1-\omega) \cdot c_{mot}^{(i)}$ that matches the ground truth for test sample *i*. The upper bound for clip- and segment-level classification performance for all linear late fusion methods can be found in Table V (see row 3). We note that the proposed algorithm performs better than either algorithm both at the clip- and the segment-levels

Method	Classification Accuracy (clip)	Classification Accuracy (segment)
Rank-Minimization [77]	0.63	0.95
Rank-Minimization with MKL [77][72]	0.69	0.90
Linear Late Fusion - Upper Bound	0.79	0.86
Temporal Fusion	**0.90**	**1.00**

TABLE V
CLASSIFICATION RESULTS WITH LATE FUSION AND ADAPTIVE SAMPLING

(see row 4).

D. Discussion

The results from the multimodal fusion approach show that, for the most part and as expected, fusion classification approaches outperform approaches relying on individual modalities. It can also be seen that the video data provides features that are more relevant to the action classification task than the features provided by the motion data. This occurs to the extent that if the late fusion coefficients are not selected carefully (*e.g.*, if not enough weight is provided to the video branch in the late fusion scheme), the performance of the fused approach is inferior to that of the system relying on video data alone. As the appropriate post-inference fusion weight is selected, the performance of the late fusion framework approaches that of its early fusion counterpart. This result highlights the fact that the classifier trained on concatenated features automatically learns to assign larger importance to the more informative dimensions of the combined feature set. The significant performance gap between the proposed fusion scheme and existing approaches indicates that our framework can effectively learn temporal patterns of behavior across data modalities that may be overshadowed by the dominance of an individual modality. It is apparent that the modalities used have largely different statistical properties which are difficult to exploit jointly with shallow models. While there seems to be a semantic correlation between the modalities, as evidenced by the improved results of the different fusion approaches, the proposed deep model is better at capturing the high-level association between the modalities than the existing shallow approaches. Lastly, the experimental findings on adaptive sampling on both the proposed approach and on more standard video-based action recognition pipelines indicate that feature descriptors derived from spatiotemporal regions in close vicinity of the hands are critical for recognizing common human tasks involving hand-eye coordination in egocentric video.

V. CONCLUSIONS

We have proposed a novel deep learning framework for fusion of multimodal temporal data. Unlike existing deep approaches for data fusion, the method is deep end-to-end, which means that the multiple stages can be optimized simultaneously and there is no need for tuning of hand-engineered features of the different modalities. Also, unlike existing fusion methods, it explicitly models temporal sequence behavior and correlations across modalities. Temporal fusion is achieved by implementing a hierarchical model where correlations between features within each modality are minimized prior to the fusion taking place. In this manner, the system is capable of better learning temporal trends across modalities, which would be otherwise drowned out by high intra-modality correlations. We demonstrated the performance of the proposed method on a public multimodal dataset and showed that it achieves marked improvements in action classification over traditional and state-of-the-art fusion approaches. We also demonstrated additional gains in performance by combining the proposed approach with slight modifications of a previously introduced [35] adaptive sampling scheme for egocentric action and activity recognition, which reenforces the notion that certain parts of the video provide more relevant information to automated decision-making tasks than others. In particular, both hand presence and location seem to provide important cues to automated action classification tasks.

We envision the proposed fusion framework supporting usage scenarios wherein the multimodal input data can be streamed to a cloud (or even to a connected smartphone serving as a local host,) analysis is done offline, and fed back within only a few minutes of completion of the action. One such scenario is the automated evaluation of self-injection practices of diabetics as part of public health studies. In this setting, study participants use the wearable equipment (wearable camera and wrist motion sensor) while injecting themselves with insulin, either in their own homes or at a clinic. The procedure only lasts a few minutes, well short of the battery lifespan of a typical wearable video-acquisition device (the Google Glass supports approximately 30 minutes of continuous video recording.) Data collected from these devices is analyzed using the proposed fusion technique after the fact to measure process compliance. With subsampling at the point of video capture, the battery life of the video acquisition device may be extended to cover multiple sessions per user. The expectation is that as devices continue to get more capable and power-efficient, more of the processing can be performed locally in the device with ever decreasing reliance on cloud resources. Developing fusion algorithms capable of supporting real-time, in-device inference and feedback is a promising direction for future research.

REFERENCES

[1] A. P. Abidoye, N. A. Azeez, A. O. Adesina, K. K. Agbele, and H. O. Nyongesa, “Using wearable sensors for remote healthcare monitoring system,” *J. Sensor Technology*, vol. 1, no. 2, pp. 22–28, 2011.

[2] A. Amanatiadis, A. Gasteratos, and D. Koulouriotis, “An intelligent multi-sensor system for first responder indoor navigation,” *Measurement Science and Technology*, vol. 22, no. 11, p. 114025, 2011. [Online]. Available: http://stacks.iop.org/0957-0233/22/i=11/a=114025

[3] American Association of Diabetes Educators, “Strategies for insulin injection therapy in diabetes self-management,” Apr 2011.

[4] American Diabetes Association, “Insulin administration,” *Diabetes Care*, vol. 27, no. suppl 1, pp. S106–S107, 2004.

[5] P. K. Atrey, M. A. Hossain, A. E. Saddik, and M. S. Kankanhalli, “Multimodal fusion for multimedia analysis: a survey,” *Multimedia Systems*, vol. 16, no. 6, pp. 345–379, 2010.

[6] A. Babakanian and A. Nahapetian, “Data fusion for movement visualization in a remote health monitoring system,” in *Mobile Computing, Applications, and Services*. Springer Berlin Heidelberg, 2012, vol. 95, pp. 32–40.

[7] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt, “Sequential deep learning for human action recognition,” in *Proceedings of the Second International Conference on Human Behavior Unterstanding*. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 29–39. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-25446-8_4

[8] O. Banos, C. Villalonga, M. Damas, P. Gloesekoetter, H. Pomares, and I. Rojas, “Physiodroid: Combining wearable health sensors and mobile devices for a ubiquitous, continuous, and personal monitoring,” *The Scientific World Journal*, vol. 2014, 2014.

[9] Y. Bengio, P. Simard, and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, Mar 1994.

[10] A. Beso, B. Franklin, and N. Barber, “The frequency and potential causes of dispensing errors in a hospital pharmacy,” *Pharmacy World and Science*, vol. 27, no. 3, pp. 182–190, 2005. [Online]. Available: http://dx.doi.org/10.1007/s11096-004-2270-8

[11] B. Chen, J.-A. Ting, B. Marlin, and N. de Freitas, “Deep learning of invariant spatio-temporal features from video,” in *NIPS 2010 Deep Learning and Unsupervised Feature Learning Workshop*, 2010. [Online]. Available: http://www.cs.ubc.ca/ nando/papers/nipsworkshop2010.pdf

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

[13] F. Chollet, “keras,” https://github.com/fchollet/keras, 2015.

[14] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie, “Behavior recognition via sparse spatio-temporal features,” in *2nd Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, Oct 2005, pp. 65–72.

[15] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, T. Darrell, and K. Saenko, “Long-term recurrent convolutional networks for visual recognition and description,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 2625–2634.

[16] A. Dosovitskiy, J. T. Springenberg, M. A. Riedmiller, and T. Brox, “Discriminative unsupervised feature learning with convolutional neural networks,” *CoRR*, vol. abs/1406.6909, 2014. [Online]. Available: http://arxiv.org/abs/1406.6909

[17] F. Eyben, S. Petridis, B. Schuller, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, “Audiovisual classification of vocal outbursts in human conversation using long-short-term memory networks,” in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2011, pp. 5844–5847.

[18] A. Fathi, A. Farhadi, and J. Rehg, “Understanding egocentric activities,” in *Computer Vision (ICCV), 2011 IEEE International Conference on*, Nov 2011, pp. 407–414.

[19] A. Fathi, Y. Li, and J. M. Rehg, “Learning to recognize daily actions using gaze,” in *Proceedings of the 12th European Conference on Computer Vision - Volume Part I*, ser. ECCV’12, vol. 7572. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 314–327.

[20] A. Fathi, X. Ren, and J. Rehg, “Learning to recognize objects in egocentric activities,” in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, June 2011, pp. 3281–3288.

[21] C. Gan, N. Wang, Y. Yang, D.-Y. Yeung, and A. G. Hauptmann, “Devnet: A deep event network for multimedia event detection and evidence recounting,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 2568–2577.

[22] Y. E. Gelogo and H.-K. Kim, “Integration of wearable monitoring device and android smartphone apps for u-healthcare monitoring system,” *International Journal of Software Engineering and Its Applications*, vol. 9, no. 4, pp. 195–202, 2015.

[23] F. A. Gers, N. N. Schraudolph, and J. Schmidhuber, “Learning precise timing with LSTM recurrent networks,” *J. Mach. Learn. Res.*, vol. 3, pp. 115–143, Mar 2003.

[24] M. Grassi, A. Lombardi, G. Rescio, M. Ferri, P. Malcovati, A. Leone, G. Diraco, P. Siciliano, M. Malfatti, and L. Gonzo, “An integrated system for people fall-detection with data fusion capabilities based on 3d tof camera and wireless accelerometer,” in *Sensors, 2010 IEEE*, Nov 2010, pp. 1016–1019.

[25] A. Graves, N. Jaitly, and A. R. Mohamed, “Hybrid speech recognition with deep bidirectional LSTM,” in *2013 IEEE International Conference on Automatic Speech Recognition and Understanding (ASRU)*, Dec 2013, pp. 273–278.

[26] A. Graves, A. Mohamed, and G. E. Hinton, “Speech recognition with deep recurrent neural networks,” *CoRR*, vol. abs/1303.5778, 2013. [Online]. Available: http://arxiv.org/abs/1303.5778

[27] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997. [Online]. Available: http://dx.doi.org/10.1162/neco.1997.9.8.1735

[28] T. Kanade and M. Hebert, “First-person vision,” *Proceedings of the IEEE*, vol. 100, no. 8, pp. 2442–2453, Aug 2012.

[29] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, “Large-scale video classification with convolutional neural networks,” in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, June 2014, pp. 1725–1732.

[30] Y. Ke, R. Sukthankar, and M. Hebert, “Efficient visual event detection using volumetric features,” in *2005 Tenth IEEE International Conference on Computer Vision*, vol. 1, Oct 2005, pp. 166–173.

[31] J. Kerr, S. J. Marshall, S. Godbole, J. Chen, A. Legge, A. R. Doherty, P. Kelly, M. Oliver, H. Badland, and C. Foster, “Using the sensecam to improve classifications of sedentary behavior in free-living settings,” *American Journal of Preventive Medicine*, vol. 44, no. 3, pp. 290–296, 2016.

[32] M. Kourogi and T. Kurata, “A method of personal positioning based on sensor data fusion of wearable camera and self-contained sensors,” in *Proceedings of IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, July 2003, pp. 287–292.

[33] ——, “Personal positioning based on walking locomotion analysis with self-contained sensors and a wearable camera,” in *Proceedings of the 2Nd IEEE/ACM International Symposium on Mixed and Augmented Reality*, ser. ISMAR ’03. Washington, DC, USA: IEEE Computer Society, 2003, pp. 103–112. [Online]. Available: http://dl.acm.org/citation.cfm?id=946248.946806

[34] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. Burges, L. Bottou, and K. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

[35] J. Kumar, Q. Li, S. Kyal, E. A. Bernal, and R. Bala, “On-the-fly hand detection training with application in egocentric action recognition,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, June 2015, pp. 18–27.

[36] D. Lahat, T. Adali, and C. Jutten, “Multimodal data fusion: An overview of methods, challenges, and prospects,” *Proceedings of the IEEE*, vol. 103, no. 9, pp. 1449–1477, Sep 2015.

[37] G. R. G. Lanckriet, N. Cristianini, P. Bartlett, L. E. Ghaoui, and M. I. Jordan, “Learning the kernel matrix with semidefinite programming,” *J. Mach. Learn. Res.*, vol. 5, pp. 27–72, Dec. 2004. [Online]. Available: http://dl.acm.org/citation.cfm?id=1005332.1005334

[38] G. R. G. Lanckriet, T. De Bie, N. Cristianini, M. I. Jordan, and W. S. Noble, “A statistical framework for genomic data fusion,” *Bioinformatics*, vol. 20, no. 16, pp. 2626–2635, Nov. 2004. [Online]. Available: http://dx.doi.org/10.1093/bioinformatics/bth294

[39] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, “Learning realistic human actions from movies,” in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2008, pp. 1–8.

[40] I. Laptev, “On space-time interest points,” *Int. J. Comput. Vision*, vol. 64, no. 2-3, pp. 107–123, Sep. 2005.

[41] Q. Le, W. Zou, S. Yeung, and A. Ng, “Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis,” in *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2011, pp. 3361–3368.

[42] D. Lebel, M. Thibault, and J. Bussières, “Asynchronous validation and documentation of sterile compounding in a hospital pharmacy,” *The Canadian Journal of Hospital Pharmacy*, vol. 63, no. 4, pp. 323–327, 2010.

[43] Y. J. Lee, J. Ghosh, and K. Grauman, “Discovering important people and objects for egocentric video summarization,” in *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2012, pp. 1346–1353.

[44] Z. Lu and K. Grauman, “Story-driven summarization for egocentric video,” in *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2013, pp. 2714–2721.

[45] K. Matsuo, K. Yamada, S. Ueno, and S. Naito, “An attention-based activity recognition for egocentric video,” in *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, June 2014, pp. 565–570.

[46] T. Mccandless and K. Grauman, “Object-centric spatio-temporal pyramids for egocentric activity recognition,” in *Proceedings of the British Machine Vision Conference*, 2013, pp. 30.1–30.11.

[47] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, “Multimodal deep learning,” in *International Conference on Machine Learning (ICML)*, Bellevue, USA, June 2011.

[48] K. Ogaki, K. Kitani, Y. Sugano, and Y. Sato, “Coupling eye-motion and ego-motion features for first-person activity recognition,” in *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, June 2012, pp. 1–7.

[49] G. O’Loughlin, S. J. Cullen, A. McGoldrick, S. O’Connor, R. Blain, S. O’Malley, and G. D. Warrington, “Using a wearable camera to increase the accuracy of dietary analysis,” *American journal of preventive medicine*, vol. 44, no. 3, pp. 297–301, 2016.

[50] F. J. Ordez and D. Roggen, “Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition,” *Sensors*, vol. 16, no. 1, p. 115, Jan 2016. [Online]. Available: http://www.mdpi.com/1424-8220/16/1/115

[51] H. Pirsiavash and D. Ramanan, “Detecting activities of daily living in first-person camera views,” in *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2012, pp. 2847–2854.

[52] V. Potdar, A. Sharif, and E. Chang, “Wireless sensor networks: A survey,” in *2009 International Conference on Advanced Information Networking and Applications Workshops*, May 2009, pp. 636–641.

[53] A. Prest, C. Schmid, and V. Ferrari, “Weakly supervised learning of interactions between humans and objects,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 3, pp. 601–614, Mar 2012.

[54] X. Ren and C. Gu, “Figure-ground segmentation improves handled object recognition in egocentric video,” in *2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2010, pp. 3137–3144.

[55] X. Ren and M. Philipose, “Egocentric recognition of handled objects: Benchmark and analysis,” in *2009 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, June 2009, pp. 1–8.

[56] M. Ryoo and L. Matthies, “First-person activity recognition: What are they doing to me?” in *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2013, pp. 2730–2737.

[57] S. Sadanand and J. J. Corso, “Action bank: A high-level representation of activity in video,” in *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2012, pp. 1234–1241.

[58] H. Sak, A. W. Senior, and F. Beaufays, “Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition,” *CoRR*, vol. abs/1402.1128, 2014. [Online]. Available: http://arxiv.org/abs/1402.1128

[59] P. Scovanner, S. Ali, and M. Shah, “A 3-dimensional sift descriptor and its application to action recognition,” in *Proceedings of the 15th International Conference on Multimedia*. New York, NY, USA: ACM, 2007, pp. 357–360.

[60] A. R. Silva, S. Pinho, L. M. Macedo, and C. J. Moulin, “Benefits of sensecam review on neuropsychological test performance,” *American journal of preventive medicine*, vol. 44, no. 3, pp. 302–307, 2016.

[61] K. Simonyan and A. Zisserman, “Two-stream convolutional networks for action recognition in videos,” *CoRR*, vol. abs/1406.2199, 2014. [Online]. Available: http://arxiv.org/abs/1406.2199

[62] K. Sohn, W. Shang, and H. Lee, “Improved multimodal deep learning with variation of information,” in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, Eds. Curran Associates, Inc., 2014, pp. 2141–2149. [Online]. Available: http://papers.nips.cc/paper/5279-improved-multimodal-deep-learning-with-variation-of-information.pdf

[63] B. Soran, A. Farhadi, and L. Shapiro, “Generating notifications for missing actions: Don’t forget to turn the lights off!” in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec 2015, pp. 4669–4677.

[64] E. Spriggs, F. De la Torre, and M. Hebert, “Temporal segmentation and activity classification from first-person sensing,” in *2009 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, June 2009, pp. 17–24.

[65] N. Srivastava and R. Salakhutdinov, “Multimodal learning with deep boltzmann machines,” *Journal of Machine Learning Research*, vol. 15, pp. 2949–2980, 2014. [Online]. Available: http://jmlr.org/papers/v15/srivastava14b.html

[66] N. Srivastava and R. R. Salakhutdinov, “Learning representations for multimodal data with deep belief nets,” in *2012 ICML Representation Learning Workshop*, 2012.

[67] T. S. G. Stacciarini, A. E. Pace, and V. J. Haas, “Insulin self-administration technique with disposable syringe among patients with diabetes mellitus followed by the family health strategy,” *Revista Latino-Americana de Enfermagem*, vol. 17, no. 4, pp. 474 – 480, Aug 2009.

[68] T. Stiefmeier, G. Ogris, H. Junker, P. Lukowicz, and G. Troster, “Combining motion sensors and ultrasonic hands tracking for continuous activity recognition in a maintenance scenario,” in *Wearable Computers, 2006 10th IEEE International Symposium on*, Oct 2006, pp. 97–104.

[69] A. Surendranath, B. Nagaraju, G. Padmavathi, S. C. Anand, P. Fayaz, and G. Balachandra, “A study to assess the knowledge and attitude of insulin self administration among patuents with diabetes mellitus,” *Asian journal of pharmaceutical and clinical research*, vol. 5, no. 1, p. 63, Jan 2012.

[70] D. Surie, T. Pederson, F. Lagriffoul, L.-E. Janlert, and D. Sjlie, “Activity recognition using an egocentric perspective of everyday objects,” in *Ubiquitous Intelligence and Computing*, ser. Lecture Notes in Computer Science, J. Indulska, J. Ma, L. Yang, T. Ungerer, and J. Cao, Eds. Springer Berlin Heidelberg, 2007, vol. 4611, pp. 246–257. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-73549-6_25

[71] G. W. Taylor, R. Fergus, Y. LeCun, and C. Bregler, “Convolutional learning of spatio-temporal features,” in *Proceedings of the 11th European Conference on Computer Vision: Part VI*, ser. ECCV’10. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 140–153. [Online]. Available: http://dl.acm.org/citation.cfm?id=1888212.1888225

[72] M. Varma and B. R. Babu, “More generality in efficient multiple kernel learning,” in *Proceedings of the International Conference on Machine Learning*, June 2009, pp. 1065–1072. [Online]. Available: http://research.microsoft.com/apps/pubs/default.aspx?id=172472

[73] U. Varshney, “Pervasive healthcare and wireless health monitoring,” *Mob. Netw. Appl.*, vol. 12, no. 2-3, pp. 113–127, Mar. 2007.

[74] E. Vig, M. Dorr, and D. Cox, “Space-variant descriptor sampling for action recognition based on saliency and eye movements,” in *Proceedings of the 12th European Conference on Computer Vision - Volume Part VII*, ser. ECCV’12. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 84–97.

[75] H. Wang, A. Kläser, C. Schmid, and C. Liu, “Dense trajectories and motion boundary descriptors for action recognition,” *International Journal of Computer Vision*, vol. 103, no. 1, pp. 60–79, May 2013.

[76] J. Xu, L. Mukherjee, Y. Li, J. Warner, J. M. Rehg, and V. Singh, “Gaze-enabled egocentric video summarization via constrained submodular maximization,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 2235–2244.

[77] G. Ye, D. Liu, I. H. Jhuo, and S. F. Chang, “Robust late fusion with rank minimization,” in *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2012, pp. 3021–3028.

[78] W.-J. Yi, O. Sarkar, S. Mathavan, and J. Saniie, “Wearable sensor data fusion for remote health assessment and fall detection,” in *2014 IEEE International Conference on Electro/Information Technology (EIT)*, June 2014, pp. 303–307.

[79] S. Yu, T. Falck, A. Daemen, L.-C. Tranchevent, J. A. Suykens, B. De Moor, and Y. Moreau, “L2-norm multiple kernel learning and its application to biomedical data fusion,” *BMC Bioinformatics*, vol. 11, no. 1, pp. 1–24, 2010. [Online]. Available: http://dx.doi.org/10.1186/1471-2105-11-309