Deep Multimodal Fusion: Combining Discrete Events and Continuous Signals

Héctor P. Martínez
Institute for Digital Games
University of Malta, 2080
Msida, Malta
hector.p.martinez@um.edu.mt

Georgios N. Yannakakis
Institute for Digital Games
University of Malta, 2080
Msida, Malta
georgios.yannakakis@um.edu.mt

ABSTRACT

Multimodal datasets often feature a combination of continuous signals and a series of discrete events. For instance, when studying human behaviour it is common to annotate actions performed by the participant over several other modalities such as video recordings of the face or physiological signals. These events are nominal, not frequent and are not sampled at a continuous rate while signals are numeric and often sampled at short fixed intervals. This fundamentally different nature complicates the analysis of the relation among these modalities which is often studied after each modality has been summarised or reduced.

This paper investigates a novel approach to model the relation between such modality types bypassing the need for summarising each modality independently of each other. For that purpose, we introduce a deep learning model based on convolutional neural networks that is adapted to process multiple modalities at different time resolutions we name *deep multimodal fusion*. Furthermore, we introduce and compare three alternative methods (convolution, training and pooling fusion) to integrate sequences of events with continuous signals within this model. We evaluate deep multimodal fusion using a game user dataset where player physiological signals are recorded in parallel with game events. Results suggest that the proposed architecture can appropriately capture multimodal information as it yields higher prediction accuracies compared to single-modality models. In addition, it appears that pooling fusion, based on a novel *filter-pooling* method provides the more effective fusion approach for the investigated types of data.

Categories and Subject Descriptors

H.1.2 [**Information Systems**]: User/Machine Systems—*Human factors*; I.2.1 [**Artificial Intelligence**]: Applications and Expert Systems—*Games*

ICMI'14, November 12–16, 2014, Istanbul, Turkey.

http://dx.doi.org/10.1145/2663204.2663236.

General Terms

Algorithms, Experimentation, Human Factors

Keywords

Deep Learning; Multimodal Fusion; Sequence Classification; Sequence Fusion; Auto-encoders; Convolutional Neural Networks; Pooling Method; Physiology; Behaviour

1. INTRODUCTION

The different sources of information that form a multimodal dataset are usually presented on a variety of formats related to their abstraction level. We can distinguish two particular formats that are common across multiple domains: *continuous signals* and sequences of *discrete events*. Continuous signals are often captured by hardware devices such as cameras, microphones and physiological sensors that monitor a human or the environment at a high and fixed sampling rate and a low abstraction level. On the other hand, sequences of discrete events are processed by software (e.g. a logged action in a virtual environment) or annotated by experts (e.g. a facial action unit), they are more infrequent than continuous recordings but they hold more abstract and complex information that characterises the user's behaviour or context. Capturing the relation between these two streams of data can provide valuable information that is not visible on each modality independently. For example, an increment in skin conductance (SC) can indicate an increment in arousal but a facial action unit indicating a smile or a frown can connect that heightened state of arousal to excitement or anger. However, integrating automatically these signals is not trivial due to their different formats and abstraction levels. Most studies resort to feature- or model-level fusion [16], i.e. the signals are reduced independently prior to their fusion (see [21, 12] among others). This process, however, neglects the low-level interactions among modalities which leads to models of lower expressivity.

In this paper we introduce *deep learning* as a general methodology for *data-level fusion*. Rather than reducing each modality independently (feature-level fusion) our approach generates a set of multimodal features, thereby, maintaining core properties of the dissimilar signals and resulting to fused models of higher accuracy. While deep learning has been applied for automatic feature extraction [8, 9, 13] the integration of multiple modalities with different time resolutions has been far from trivial. Towards this aim this paper introduces an augmented convolutional neural network (CNN) that integrates signals at different layers according

to their sampling rate. CNNs reduce the time resolution of their input signals as they pass through convolution and pooling layers, increasing the abstraction level of the information and, thereby, making them the ideal model for the fusion task at hand.

Based on the CNN model, we evaluate three different approaches to fuse sequences of discrete events with continuous signals. The first approach, namely *convolution fusion*, consists of transforming the sequence of events into a *pulse signal* that is one when the event is occurring and zero otherwise. This signal can be simply introduced as the input to a convolutional layer. The second approach, namely *training fusion*, utilises the fused signal that is created when combining information from the original continuous signals and the sequence of discrete events. This fused signal is produced by a single-layer CNN that models the effect discrete events have on the continuous signals and vice-versa. In this approach, the sequences of events are not directly added to a convolutional layer, but they are used to stir the training process. The third approach, namely *pooling fusion*, introduces the sequence of discrete events into a pooling layer resulting in a novel *filter-pooling* method that attenuates the outputs of a convolutional layer within time intervals when events are not occurring. This method complements training fusion by highlighting the patterns detected by a CNN when the interaction among modalities is potentially occurring.

We evaluate and compare the three approaches on a dataset that includes a physiological signal (skin conductance) and two different sequences of events (collecting pellets and colliding with enemies in a 3D prey/predator game). After extracting multimodal features with the CNN, we use a single-layer perceptron to predict six different affective states. The results of this initial study show that convolutional neural networks can be adapted to create multimodal features with a high predictive power. Pooling fusion appears to be the most promising fusion approach. Even though the filter-pooling layers hinder information when events do not occur, when combined with standard average-pooling layers the pooling fusion approach outperforms the other two fusion approaches and any single-modality modelling attempts. Convolution fusion also yields high prediction accuracies but it shows lower robustness. Finally, the training fusion does not appear to be able to capture information specific to both modalities within the space of topologies explored.

2. RELATED WORK

Sequences of discrete events and continuous signals are often found together in many different domains but this combination is particularly prominent in human-computer interaction studies. The interaction with the computer is often characterised by a series of events that give a context to multiple other modalities that generate continuous streams of data. Successfully fusing context with other modalities appears to be the key to automatically understand user experience [1].

In this domain, McQuiggan et al. [12] used statistical summarisation (i.e. extracted averages, standard deviations and other simple statistical features) to reduce continuous signals such as heart rate and skin conductance. In addition, different events arising from the interaction with a 3D learning environment where processed with ad-hoc metrics such as the time spent on a task (duration of the event or time between events) or a list of locations visited. On a similar basis, statistical summarisation was used to reduce several physiological features and game events in [10, 21]. Although these simple approaches yield models of acceptable prediction accuracies, the exact relation among modalities remains largely unknown. The method proposed in this paper attempts to improve the accuracy and expressiveness of these models by exploiting the low temporal relation among modalities.

Data-level fusion of events and continuous signals has been achieved before using frequent sequence mining [11]; however, this approach requires transforming the continuous signals into sequences of events. For instance, in [11] a skin conductance signal is converted into a sequence of significant increments and decrements of the signal. This transformation requires ad-hoc preprocessing which discards part of the information present in the raw signal. Deep learning approaches investigated in this paper, on the other hand, are defined for continuous signals and therefore do not require ad-hoc preprocessing.

Deep learning has already been applied to a number of human-computer interaction tasks such as facial expression recognition [17], speech recognition [8] and modelling of physiological signals [9]; however, all these studies develop methods tailored for single-modality atemporal or continuous data, and therefore are not suited for series of events that are central to this paper. Likewise, some studies have explored the fusion of dissimilar modalities that are either atemporal (e.g. text and images [19], images and sound [14]) or well synchronised (e.g. video and sound [13]). The main approach followed in these studies consists of processing each modality individually and combining them at a higher representation level. By doing so, however, the low level interactions between modalities may be hindered, which, in turn, signals the necessity of new deep learning approaches to data level multimodal fusion. This paper proposes alternative deep fusion mechanisms for eliminating the drawbacks of current multimodal fusion practice reported above.

3. METHOD

In order to facilitate the extraction of information across sequences of discrete events and continuous signals, we define a new architecture for convolutional neural networks to handle inputs at different time resolutions. As CNNs are naturally suited for continuous signals but no discrete events, we propose three different approaches to integrate sequences of events with other continuous modalities. In this section we describe the basics of CNNs and detail the proposed enhancements. In addition, we briefly describe the modelling mechanism used to test the prediction power of the variant CNNs.

3.1 Convolutional Neural Networks

Convolutional (or time-delay) neural networks [8] are feed-forward neural networks designed to deal with large input spaces as those seen in image classification tasks. CNNs are constructed by stacking alternatively *convolutional layers* and *pooling layers*. A convolutional layer consists of a number of neurons that process sequentially consecutive patches of the input signal, i.e. this layer convolves a set of neurons along the temporal dimension of the input signal. Each neuron defines one local feature which is extracted at every position of the input signal; the resulting values cre-

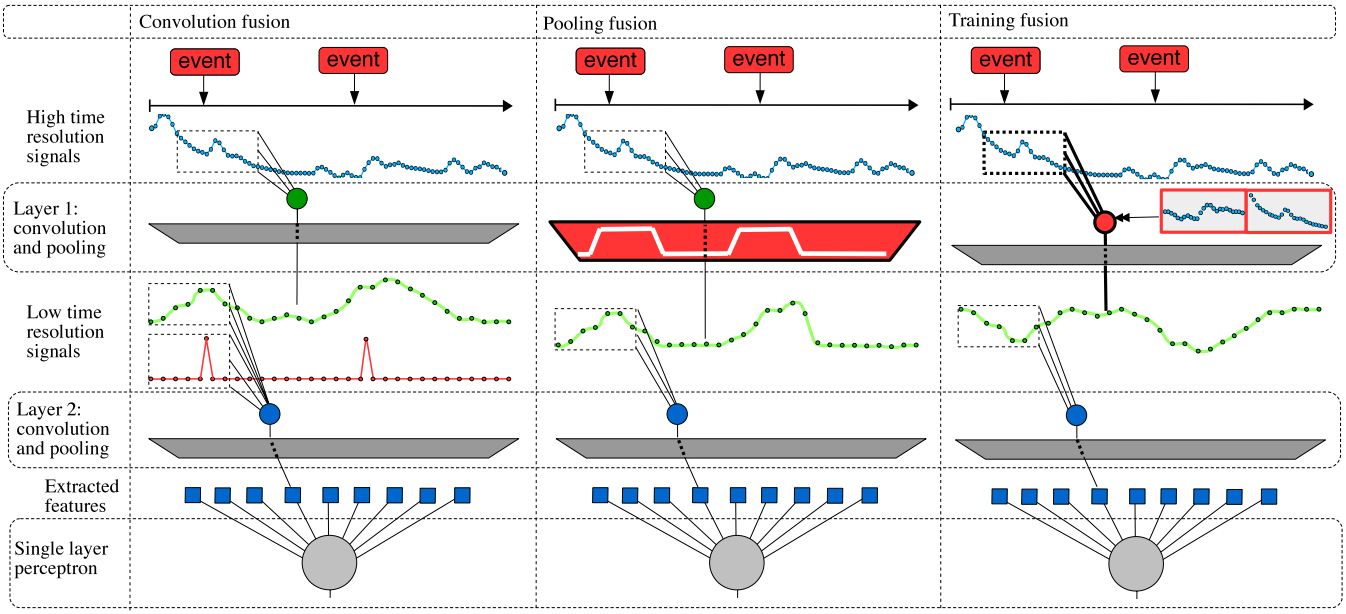


Figure 1: Three approaches to deep multimodal fusion via convolutional neural networks. The example CNNs illustrated present two layers with one neuron each. The first convolutional layer receives as input a continuous signal at a high time resolution, which is further reduced by a pooling layer. The resulting signal (feature map) presents a lower time resolution. The second convolutional layer can combine this feature map with additional modalities at the same low resolution. In the convolution fusion network (left figure), the two events are introduced at this level as a pulse signal. In the pooling fusion network (middle figure), the events are introduced as part of the first pooling layer, resulting in a filtered feature map. Finally, in the training fusion network (right figure), the events affect the training process of the first convolutional layer, leading to an alternative feature map.

ate a new signal referred to as *feature map*. A pooling layer reduces the dimensionality of the feature maps generated by a convolutional layer. It typically applies a simple statistical function (e.g. average or maximum value) to non-overlapping patches of the feature maps. By stacking several convolutional and pooling layers, CNNs can effectively reduce the input signals to a small set of features. The experimenter must define the topology of the network including the number of neurons and the number of inputs for each neuron in each convolutional layer, and the number of inputs and function used by the pooling layer. Once those parameters are fixed, the neurons can be automatically trained to minimise the loss of information in the feature extraction process.

The pooling layers perform a very simple processing task, and thus the extracted features are mainly defined by the convolutional layers, and more concretely, their neurons. By analysing the weights of each neuron, one can derive the characteristics of the input that every feature is capturing. When the input to the neuron is a 1-dimensional signal (e.g. skin conductance), the weights of the neuron can be plotted in a temporal order to reveal the input patterns that yield higher output values (e.g. see [9]).

We train each convolutional layer using *denoising auto-encoders* [5, 4]. This approach consists of feeding the outputs of the convolutional layer into a *decoder* that reconstructs the original input signal; by means of gradient-descent, the weights of each neuron are adjusted iteratively to achieve a minimal reconstruction error, i.e. a minimal discrepancy between the signal reconstructed by the decoder and the original input signal [9]. The neurons of each convolutional layer are trained patch-wise, i.e. by considering the input at each position (one patch) in the sequence as one example. This allows faster training than training convolutionally, but may yield translated versions of the same filter. This is the same idea as the process followed by PCA but with the advantage of bypassing the linearity assumption [20]. Alternative training methods such as restricted Boltzmann machines present a more complex theoretical background but have not produced better results in practice. Thus, we opt for the simplest method.

3.1.1 *Dealing with multiple time resolutions*

Standard CNN architectures process all inputs at the first layer after all data samples have been reduced to the same dimensions. Then the neurons of the first convolutional layer scan the input signals sample-by-sample along time. When the signals present different number of samples per unit of time, however, this process is not applicable as the scanning process requires a different pace for each modality.

The CNN architecture can be adapted by allowing modalities with different sampling rates to feed into different layers (see Figure 1). Consequently, the signals with a higher sampling rate are processed earlier by a convolutional and a pooling layer that transform them into a feature map with the same time resolution as the signals with lower sampling rates. Alternatively, one could simply undersample all the signals to the lowest sampling rate (which can be seen as a form of a pooling layer). However, using also a convolutional layer can potentially reduce the information loss because consecutive outputs of a convolutional layer are very similar when entering the pooling layer. In addition, the

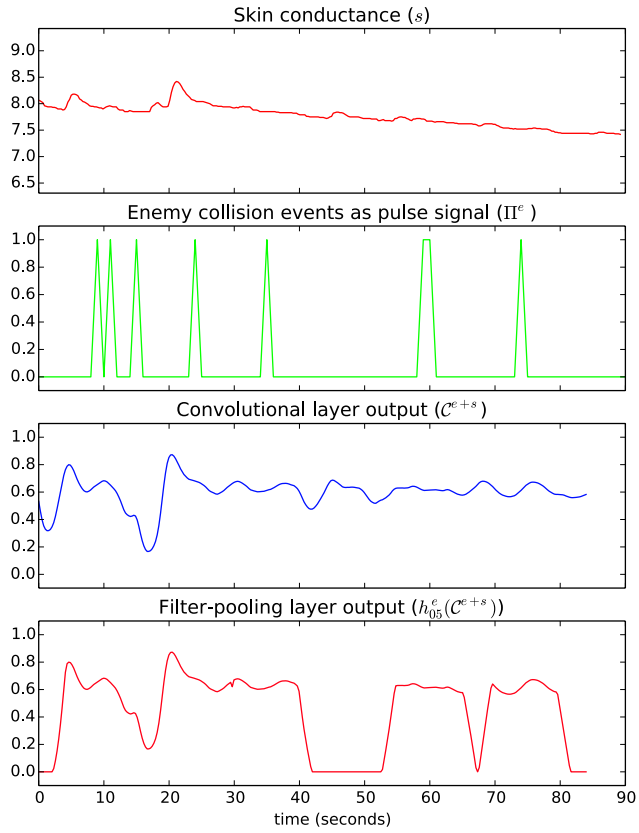


Figure 2: Signals captured in one game through the three approaches for fusion. The top figure shows the raw skin conductance signal. The second figure depicts the sequence of enemy collision events. The third figure shows the output of a convolutional layer with one neuron applied to the SC signal. The bottom figure depicts the output of a filter-pooling layer applied to the previous convolutional layer.

convolutional layers increase the complexity of the information as the time resolution is reduced, which may facilitate modality fusion at the same abstraction level. Consider for instance the example of skin conductance and pellet collection events; matching one of those events with the exact value of SC on an exact millisecond might not provide interesting information. Alternatively, if SC is first processed by a convolutional layer, it may detect spikes on the signal caused by a pellet collection event.

A disadvantage of using this architecture is that global relations might be missed (e.g. if the final game score is related to the maximum value of the SC). Nevertheless, we expect that the local temporal relations among the modalities are more informative in most domains.

3.1.2 *Convolution fusion*

With the architecture presented above, the simplest solution to introduce a sequence of events is to transform it into a continuous signal at the appropriate sampling rate. We choose a pulse signal that equals one only when the event was recorded and zero otherwise (see Figure 1 and Figure 2).

3.1.3 *Training fusion*

Training fusion is inspired by the idea that the continuous signals may present specific patterns when discrete events are occurring, that are not that common otherwise. On that basis, the events are used to select specific parts of the continuous signals. A new subset of data is created by extracting patches within a given time window centred at each event (in this paper the window is 5, 10 and 15 seconds). The resulting dataset is used to train a single-layered CNN that is specialised on the patterns of variation in the continuous signals around the events. We expect that, to some degree, this model characterises both the changes that are produced in the continuous signals when events occur and the variations in the continuous signals that prompt events.

Once trained, the single-layered CNN is convolved over the raw continuous signal to generate a new set of signals based on the variation patterns seen around the events (see Figure 2). As we are using a CNN to further process the resulting signals, this approach can be interpreted as an alternative training of the first layer of the network (see Figure 1).

3.1.4 *Pooling fusion*

An alternative approach to introduce discrete event information into the CNN is through the pooling layers. This is achieved by defining a new pooling method (denoted as *filter-pooling*) that attenuates the input signal (i.e. the output of the previous convolutional layer) around a sequence of events given. The result is a new signal where the patterns within a given window (in this paper 5, 10 and 15 seconds) centred at the events are highlighted (see Figure 2 and Figure 1)). This fusion approach makes multimodal patterns more salient for the next layer of the CNN. In this paper we use a simple filter that equals 1 within the time window selected and decays linearly to zero (or increases linearly from zero) within two seconds after (or before) the window.

3.2 Modelling

Once the CNN is trained as described in the previous section, it can be used to produce a number of multimodal features (see Figure 1). We test whether these features are more informative than single-modality features by training a single layer perceptron (SLP) to predict a series of target outputs. As the dataset used in this paper contains pairwise preferences indicating different affective states, we employ a *preference learning* variant of backpropagation to train the SLP (see Section 3.2.1). In addition, as the CNN creates features that capture any characteristic of the input signals regardless of the prediction target, *automatic feature selection* (see Section 3.2.2) is an essential process towards distinguishing which of those features can assist the creation of the prediction model. These experiments are performed using the *Preference Learning Toolbox* [2].

3.2.1 *Preference Learning*

Preference learning [3] is a subfield of machine learning that deals with the problem of learning orders. When these orders are specified as pairs of data samples, the learning problem consists of creating a model that outputs higher values for the samples preferred on each pair and lower for the non-preferred. Backpropagation [18] on its basic form optimises an error function iteratively across a number of epochs by adjusting the parameters of a neural network (i.e.

value of each weight and threshold) proportionally to the derivative (gradient) of the error with respect to the parameter. In this paper we use the *regularised least squares* error function, designed for learning from pairwise preferences [15].

3.2.2 Feature Selection

Feature selection (FS) consists of a search scheme to test alternative combinations of input features and a heuristic to determine their relevance. Opposed to other dimensionality reduction methods such as principal component analysis [20] and *Fisher's projection* [7] that project the feature space into a space of lower dimensionality, FS eliminates dimensions (features) from the original space maintaining the physical meaning of the inputs to the model. We consider that this is a key feature for multiple domains, as it is necessary to analyse the mappings captured by the models learned.

In this paper we used *sequential forward feature selection* (SFS) [21, 6], a bottom-up search procedure where one feature is added at a time to the current feature set. The feature to be added is selected from the subset of the remaining features such that the new feature set generates the maximum value of an objective function over all candidate features for addition. The objective function used in this paper is the training accuracy of an ANN model on the evaluated dataset.

4. DATASET

The dataset used to evaluate the proposed methodology was gathered during an experimental game survey in which 36 participants played four pairs of different variants of the same video-game. The test-bed game named *Maze-Ball* is a 3D prey/predator game that features a ball inside a maze controlled by the arrow keys. The goal of the player is to maximise her score in 90 seconds by collecting a number of pellets scattered in the maze while avoiding enemies that wander around. A number of eight different game variants were presented to the players. The games were different with respect to the virtual camera profile used which determines how the virtual world is presented on screen. We expected that different camera profiles would induce different experiences and affective states which would, in turn, reflect on the physiological state of the player making possible to predict the players' affective self-reported preferences using information extracted from their physiology. Each participant played one pair of variants in both orders and other two pairs with different game variants. The games played by each participant are assigned in such a way that, in total, 4 preference instances should be obtained for each pair of the game variants in both orders (2 preference instances per playing order).

Skin conductance was recorded during each game session using the IOM biofeedback device at 31.25Hz. While other off-the-self devices offer higher sampling rates, previous studies have shown that accurate predictors of affect can be constructed for this device. Following previous work on training convolutional models of affect from SC [9], we further reduced the signal by 8 after applying a mean filter of the same length, to produce approximately 2.5 recordings per second. In addition, pellet collection and collision with enemies are logged. After each pair of games, the players filled-in a questionnaire reporting which of the games induced higher levels of *anxiety*, *challenge*, *excitement*, *fun*, *frustration* and *relaxation*. The answers "none of the two'' and "both equally'' are also included in their options (4 alternative forced choice questionnaire). The details of the Maze-Ball game design and the experimental protocol followed are already well reported in the literature and can be found in [21, 10].

5. RESULTS

In order to evaluate the proposed methodology, we compare single and multiple modality features created with CNNs. As seen in previous work [9] the topology of the network is critical to create accurate prediction models, however, the exhaustive evaluation of all combinations is intractable. After preliminary experiments with the single modalities, we fix the topology of the CNN to two layers with 5 neurons each; each neuron has 20 inputs for each input feature map (either an input modality or the output of the previous pooling layer). The learning rate, number of epochs and corruption level are adjusted for each experiment independently.

The outputs of the CNNs are used as inputs for the single layer perceptron. For each experimental condition, we run SFS 10 times, and for each of those runs we evaluate the best set of features using the 3-fold cross validation accuracy of a SLP. This procedure may hinder partially the generality of the created models, as the average cross-validation accuracy is used to guide the feature selection search. However, the comparison between single and multiple modalities, and across fusion approaches is fair because all experiments follow the same procedure.

Significance in this paper is calculated using a two-tailed unpaired t-test.

5.1 Fusing Skin Conductance with Pellet Collection

The best models that rely on training fusion applied to pellet collection events and SC ($\mathcal{C}^{p+s}$) outperform the single-modality models built on SC (s) only on *anxiety* and *frustration* (see Figure 3a). This suggests that this approach captures low amount of event-specific information. This is also supported by the small differences across trained neurons (see Figure 4). As it can be observed, the weights of the five neurons are very similar for the alternative time windows (5, 10 and 15). In fact, the difference with respect to a model built on the complete dataset (i.e. using every interval in the SC signal, hence ignoring events) is also very small.

Interestingly, the convolution fusion approach that introduces pellet collection events as a pulse signal ($\Pi^p + s$) yields prediction accuracies significantly higher than $\mathcal{C}^{p+s}$ only for *challenge* and *fun* (p-value < 0.01), and only higher than the single-modality approach for *challenge* and *anxiety* (p-values < 0.01) as depicted in Figure 3a which indicates that this approach did not fuse physiological and contextual information satisfactorily either. Models based only on the event signal (Π^p) yield poor prediction accuracies which indicates that either the pellet collection event is not relevant for prediction on its own, or that the nature of the pulse signal complicates the feature extraction task for convolutional neural networks.

When the $\mathcal{C}^{p+s}$ signals are combined with a filter-pooling layer ($h^p_{t_w}(\mathcal{C}^{p+s})$), we observe significant improvements (p-values < 0.01) in prediction accuracy for all affective states except *frustration* with respect to the unfiltered version (see

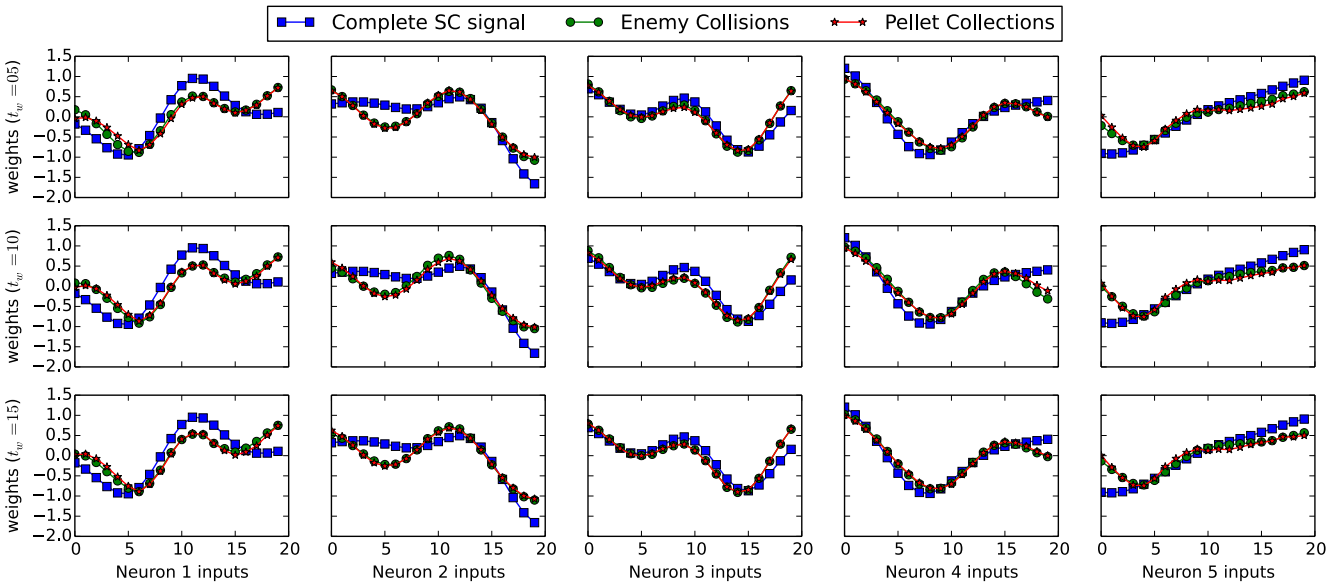


Figure 4: Example models built with the training fusion approach. Each frame shows the weight values of the same neuron from three independent CNNs: one trained with patches around pellet collection events, another trained with patches around enemy collision events and the last one trained with the complete SC signal. Each row of frames corresponds to the 5 neurons in the same network, trained using a time window t_w centred at the given event.

Figure 3b). Furthermore, these models outperform the other approaches significantly in all affective states except *fun* (no significant difference compared to single-modality), and *challenge* and *fun* (no significant difference compared to $\Pi^p + s$). This suggests that incorporating the contextual information as a filter-pooling layer, facilitates the fusion process when compared to the use of convolution fusion.

Finally, when the filtered $\mathcal{C}^{p+s}$ signals are combined with the raw SC signal into the same model ($h^p_{t_w}(\mathcal{C}^{p+s}) + s$), further improvements are observed, notably for *frustration* and *anxiety* (p-value < 0.01), and *fun* (p-value < 0.05) (see Figure 3c). This suggests that the information carried by the SC signal in intervals when no events are triggered is also relevant for prediction of particular states. It thus appears that the combination of filter-pooling and standard average-pooling methods can provide a mixture of single and multiple modality signals that yield more accurate multimodal predictors.

The time window values used to define the filter in the pooling layer signals $\mathcal{C}^{p+s}$ do not show a consistent variation across affective states. This pattern was expected as the physiological manifestations of each emotion may be salient within different time intervals around the events.

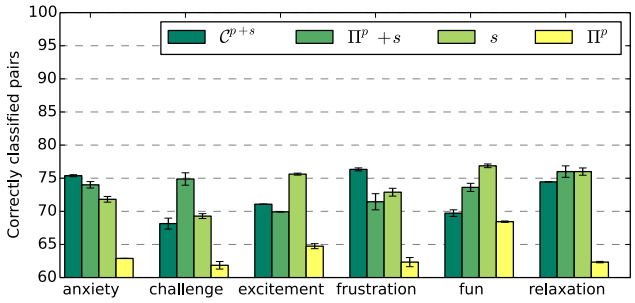
5.2 Fusing Skin Conductance with Enemy Collision

Predictors using enemy collision events as a pulse signal together with the raw SC signal ($\Pi^e + s$) show significantly higher accuracies (p-values < 0.01) than predictors using the $\mathcal{C}^{e+s}$ signals or a single modality (s or Π^e) for *fun*, *excitement*, *frustration* and *anxiety*, and no difference in the remaining two affective states (see Figure 5a). As observed by the heightened prediction accuracies of several models based on Π^e, it appears that the pulse signal based on enemy collisions yields more informative features than the pellet collection signal, which also contributes to more accurate convolution fusion models. As also seen in the previous set of experiments, training fusion models ($\mathcal{C}^{e+s}$) do not consistently outperform the single skin conductance signal. In fact, the differences between the neurons trained using SC values around enemy collision events and pellet collection events are minimal (Figure 4). Therefore, it appears that the training fusion approach captured only a minimal amount of information about the events in this dataset.

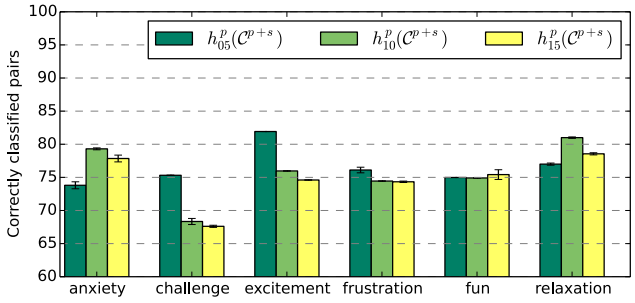
Adding a filter-pooling layer, improves the accuracy of models based on $\mathcal{C}^{e+s}$ in all affective states (p-values < 0.01), and it even results in accuracies higher than $\Pi^e + s$ for *anxiety* (p-value < 0.01), *challenge* (p-value < 0.01) and *relaxation* (p-value $= 0.03$) as depicted in Figure 5b. These results further validate the appropriateness of using the filter-pooling layer as a method to fuse sequences of events with continuous signals.

Similarly to the experiments with pellet collection events, adding the raw SC as an additional input ($h^e_{t_w}(\mathcal{C}^{e+s}) + s$) yields further improvements (see Figure 5c). These improvements lead to accuracies above (p-values < 0.02) or equal to $\Pi^e + s$ in all affective states, making this combination the most robust method for fusion identified across all experiments of this paper. As before, the optimal time window is dependent on the affective state.

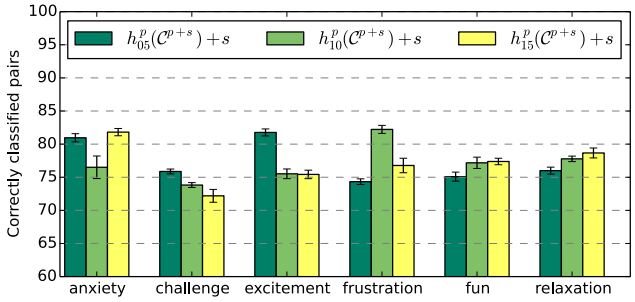
When comparing the results for the same time window value, we can observe some accuracy decrements when the raw signal is added (e.g. compare $h^e_{05}(\mathcal{C}^{e+s})$ to $h^e_{05}(\mathcal{C}^{e+s}) + s$ for *excitement*). This is caused by the fixed topology used in both experiments that forces the second model to ignore information from $h^e_{05}(\mathcal{C}^{e+s})$ in favour of the added s signal. While the new information introduced by s is in general relevant, some of the ignored information may have been key for the prediction. A simple and sensible solution consists of increasing the number of neurons as the number of inputs



(a) Single modalities (s and Π^p), convolution fusion ($s + \Pi^p$) and training fusion ($\mathcal{C}^{p+s}$)

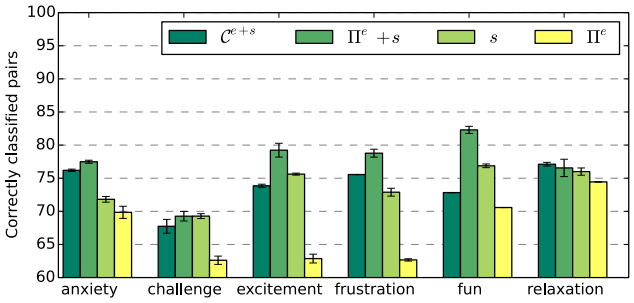


(b) Pooling fusion ($h^p_{t_w}(\mathcal{C}^{p+s})$)

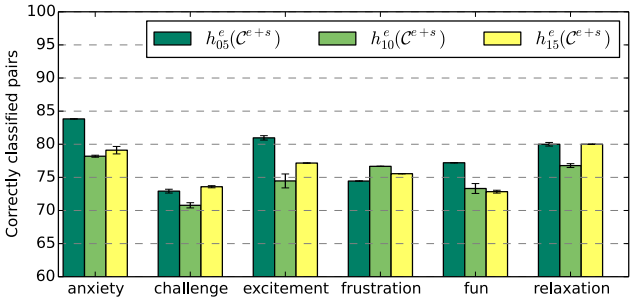


(c) Pooling fusion combined with raw SC ($h^p_{t_w}(\mathcal{C}^{p+s}) + s$)

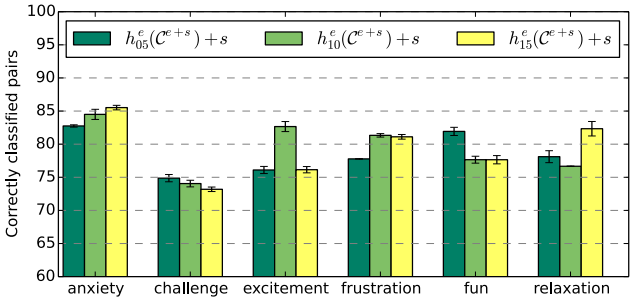
Figure 3: Pellet collection events and skin conductance. Average classification accuracy of 10 single-layer perceptrons trained on a set of features selected using sequential feature selection. The pool of available features is created by convolutional neural networks that take one or more of the following signals as input: a SC signal (s), a pulse signal representing the pellet collection events (Π^p), a fusion signal produced by a single-layered CNN ($\mathcal{C}^{p+s}$) and a fusion signal produced by a CNN with a filter-pooling layer ($h^p_{t_w}(\mathcal{C}^{p+s})$) with time window t_w.



(a) Single modalities (s and Π^e), convolution fusion ($s + \Pi^e$) and training fusion ($\mathcal{C}^{e+s}$)



(b) Pooling fusion ($h^e_{t_w}(\mathcal{C}^{e+s})$)



(c) Pooling fusion combined with raw SC ($h^e_{t_w}(\mathcal{C}^{e+s}) + s$)

Figure 5: Enemy collision events and skin conductance. Average classification accuracy of 10 single-layer perceptrons trained on a set of features selected using sequential feature selection. The pool of available features is created by convolutional neural networks that take one or more of the following signals as input: a SC signal (s), a pulse signal representing the enemy collision events (Π^e), a fusion signal produced by a single-layer CNN ($\mathcal{C}^{e+s}$) and a fusion signal produced by a CNN with a filter-pooling layer ($h^e_{t_w}(\mathcal{C}^{e+s})$) with time window t_w.

increase. This will naturally allow the CNN to extract more patterns from each modality.

6. CONCLUSIONS

In this paper we introduce deep learning as a method for fusing sequences of discrete events and continuous signals. In particular, we define a convolutional neural network architecture to integrate modalities with different time resolutions and propose three different approaches to integrate information from the continuous signals and the sequences of events. We evaluate these using a dataset that contains game events and physiological signals.

The first fusion approach, *convolution fusion*, consists of injecting events as a pulse signal into a convolutional layer. This method yielded high prediction accuracies specially when the events appeared to be informative as a standalone modality. Nevertheless, models based on a single-modality could outperform them, suggesting that the convolutional

network cannot extract all the relevant information associated with the events.

The second fusion approach, *training fusion*, is based on training a single-layer convolutional network to distinguish the effects of events in the continuous signals. This method appears to fail to capture information across modalities as—regardless of the events presented and time interval used—the resulting CNNs found similar patterns in the dataset studied. The prediction accuracies also suggest that this method is weaker than convolution fusion.

The third method, *pooling fusion*, adds information about events through a pooling layer that filters the output of a convolutional layer connected to a continuous signal. This method boosts the prediction accuracies of training fusion models suggesting the successful integration of information from continuous signals and the sequences of events. As a side effect of this pooling method, patterns found on the continuous signal when no events occur are attenuated. However, we show that a combination of this new pooling method with standard average-pooling yields models that can combine single modality and multimodality information leading to more accurate predictions than any of the other methods evaluated. It appears that the filter-pooling layer is responsible for the entire improvement on accuracy, and therefore it is likely to yield the same results if used independently of training fusion. Future work will investigate this hypothesis and explore the new space of parameters introduced by this pooling method, including the type of filter and the window size.

While our experiments are restricted to unidimensional signals, the architecture introduced in this paper is general and can be applied to other multimodal datasets including, for instance, video recordings. Future work will focus on validating these methods in such datasets as well as extending the current evaluation to a larger number of modalities, prediction models, feature selection mechanisms and network topologies.

7. ACKNOWLEDGMENTS

This research was supported, in part, by the ILearnRW (project no: 318803) FP7 ICT EU project.

8. REFERENCES

[1] L. Barrett, B. Mesquita, K. Ochsner, and J. Gross. The experience of emotion. *Annual review of psychology*, 58:373, 2007.

[2] V. E. Farrugia, H. P. Martínez, and G. N. Yannakakis. The preference learning toolbox. Technical Report IDG-2014-01, Institute of Digital Games, University of Malta, 2014.

[3] J. Fürnkranz and E. Hüllermeier. *Preference learning*. Springer, 2010.

[4] G. Hinton and R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

[5] G. E. Hinton and R. S. Zemel. Autoencoders, minimum description length, and helmholtz free energy. In *Advances in neural information processing systems (NIPS)*, 1994.

[6] A. Jain and D. Zongker. Feature selection: evaluation, application, and small sample performance. *IEEE transactions on pattern analysis and machine intelligence*, 19(2):153–158, 1997.

[7] W. Krzanowski. The performance of fisher's linear discriminant function under non-optimal conditions. *Technometrics*, 19(2):191–200, 1977.

[8] Y. LeCun and Y. Bengio. Convolutional networks for images, speech, and time series. In *The handbook of brain theory and neural networks*, volume 3361. Cambridge, MA: MIT Press, 1995.

[9] H. P. Martínez, Y. Bengio, and G. N. Yannakakis. Learning deep physiological models of affect. *Computational Intelligence Magazine, IEEE*, 9(1):20–33, 2013.

[10] H. P. Martínez and G. N. Yannakakis. Genetic search feature selection for affective modeling: a case study on reported preferences. In *Proceedings of international workshop on Affective interaction in natural environments (AFFINE)*, pages 15–20. ACM, 2010.

[11] H. P. Martínez and G. N. Yannakakis. Mining multimodal sequential patterns: a case study on affect detection. In *Proceedings of International Conference on Multimodal Interfaces (ICMI)*, pages 3–10. ACM, 2011.

[12] S. Mcquiggan, B. Mott, and J. Lester. Modeling self-efficacy in intelligent tutoring systems: An inductive approach. *User Modeling and User-Adapted Interaction*, 18(1):81–123, 2008.

[13] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng. Multimodal deep learning. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 689–696, 2011.

[14] P. O'Connor, D. Neil, S.-C. Liu, T. Delbruck, and M. Pfeiffer. Real-time classification and sensor fusion with a spiking deep belief network. *Frontiers in neuroscience*, 7, 2013.

[15] T. Pahikkala, E. Tsivtsivadze, A. Airola, J. Järvinen, and J. Boberg. An efficient algorithm for learning to rank from preference graphs. *Machine Learning*, 75(1):129–165, 2009.

[16] M. Pantic and L. Rothkrantz. Toward an affect-sensitive multimodal human-computer interaction. *Proceedings of the IEEE*, 91(9):1370–1390, 2003.

[17] S. Rifai, Y. Bengio, A. Courville, P. Vincent, and M. Mirza. Disentangling factors of variation for facial expression recognition. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2012.

[18] D. Rumelhart. *Backpropagation: theory, architectures, and applications*. Lawrence Erlbaum, 1995.

[19] N. Srivastava and R. Salakhutdinov. Multimodal learning with deep boltzmann machines. In F. Pereira, C. Burges, L. Bottou, and K. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 2222–2230. Curran Associates, Inc., 2012.

[20] S. Wold, K. Esbensen, and P. Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1):37–52, 1987.

[21] G. Yannakakis, H. Martínez, and A. Jhala. Towards affective camera control in games. *User Modeling and User-Adapted Interaction*, 20(4):313–340, 2010.