Audio Metric Learning by Using Siamese Autoencoders for One-Shot Human Fall Detection

Audio Metric Learning by using Siamese Autoencoders for One-Shot Human Fall Detection

Diego Droghini, Stefano Squartini, *Senior Member, IEEE,* Emanuele Principi, Leonardo Gabrielli, and Francesco Piazza, *Senior Member, IEEE*

***Abstract*—In the recent years, several supervised and unsupervised approaches to fall detection have been presented in the literature. These are generally based on a corpus of examples of human falls that are, though, hard to collect. For this reason, fall detection algorithms should be designed to gather as much information as possible from the few available data related to the type of events to be detected. The one-shot learning paradigm for expert systems training seems to naturally match these constraints, and this inspired the novel Siamese Neural Network (SNN) architecture for human fall detection proposed in this contribution. Acoustic data are employed as input, and the twin convolutional autoencoders composing the SNN are trained to perform a suitable metric learning in the audio domain and, thus, extract robust features to be used in the final classification stage. A large acoustic dataset has been recorded in three real rooms with different floor types and human falls performed by four volunteers, and then adopted for experiments. Obtained results show that the proposed approach, which only relies on two real human fall events in the training phase, achieves a F_1-Measure of 93.58% during testing, remarkably outperforming the recent supervised and unsupervised state-of-art techniques selected for comparison.**

***Index Terms*—Human Fall Detection, Siamese Neural Networks, One-Shot Learning, Deep Learning, Computational Audio Processing**

NOMENCLATURE

RHF	Real Human Fall.
SHF	Simulated Human Fall.
FAS	Floor Acoustic Sensor.
MFCC	Mel-Frequency Cepstral Coefficient.
k-NN	k-Nearest Neighbors.
SNN	Siamese Neural Network.
MSE	Mean Squared Error.
GMM	Gaussian Mixture Model.
UBM	Universal Background Model.
MAP	Maximum a Posteriori.
GMS	Gaussian Mean Supervector.
SVM	Support Vector Machine.
OCSVM	One-Class Support Vector Machine.
CNN	Convolutional Neural Network.
MLP	Multi-Layer Perceptron.
ReLU	Rectifier Linear Unit.

I. INTRODUCTION

The continuous and unprecedented growth rate of the elderly world population is one of the primary aspects of concern for society and governments. Nowadays, about 8.5% of people in the world are more than 65 years old [1], [2]. Although the average life of the world population is getting longer, elderly people may not necessarily live a healthier life. Indeed, 37.5 million falls require medical interventions and more than 600 thousand are cause of death every year worldwide. In particular, the population segment most affected by this problem is composed of elderly over 65 years that, combined with the growing mobility of the population, are more frequently left alone in their homes without aid in the case of need. Moreover, since falls are the leading cause of death and hospitalizations for older adults, this phenomenon leads to a substantial increase of the cost of healthcare [3], [4].

It is not surprising, thus, that the research community is encouraged, even by governments, to find reliable and performing solutions to minimize the damage caused by the human fall problem. This is also confirmed by the presence in the literature of several contributions dedicated to this specific topic [4]–[9]. In fact, in the past few years, a variety of systems have been presented. One way to divide the methodologies for approaching the fall detection problem is based on the placement of the sensing devices [4]. The main categories are wearable, vision, and environmental, with each category presenting their own advantages and disadvantages. Wearable systems do not suffer from ambient condition, but people may forget to wear them, and they are not operational during the charging time, thus, some people may consider them annoying. Furthermore, a device must be installed on each person to be monitored. An environmental sensor may be used to avoid this kind of problems, but with other limitations. Vision systems, although they are actually environmental sensors, deserve a dedicated category because of many systems proposed in the literature based on them [4]. This category includes several types of sensors like, e.g., cameras for which the major limitations are field-of-view constraints, lighting condition, positioning of multiple cameras and lack of privacy. The environmental sensors category includes several types of sensors. For example, radar doppler based systems used in [10] raise fewer privacy concerns, but they suffer from reflection and blind spots. In particular, for a data-driven system, another aspect that should not be underestimated is the need for a re-training when changing the environment to be monitored or even just some of its components such as the arrangement of furniture as happens in [11]. All this implies that there is no optimal choice, which is instead, a compromise that depends on the type of environment that is monitored as well as on personal sensitivity of the subjects under monitoring.

All authors are with the Department of Information Engineering, Università Politecnica delle Marche, Via Brecce Bianche 12, 60131 Ancona, Italy, e-mail: s.squartini@univpm.it.

From a different point of view, another significant distinction between fall detection systems can be made based on the type and amount of data used for the algorithm development [5]. In fact, the problem can be approached either as supervised or unsupervised based on the availability of data in the hands of the researchers as well as their goals. Most state-of-the-art methods tackle the problem under fully supervised conditions assuming they have enough data for falls. Almost all falls are simulated with professional mannequins [12], [13] or by people with adequate protections [14], [15] that however may not correctly emulate an actual fall. Although this approach leads to accurate results, there is no guarantee that it will generalize well in real situations. Other researchers opt for approaches based on outlier/anomaly detection [16]–[18] because of the large availability of data that can represent normal activity. However, it is challenging to define what "normal activities" are for such approaches, and the risk is to raise several false alarms. Perhaps the situation that most closely approximates reality is a hybrid between the previous ones, in which a large amount of data representing the normality are easily available, with just a few samples of RHF and eventually some related synthetic or simulated data. In these situations, supervised approaches that suffer from strong data imbalance have to apply subsampling [19] or weighting [5] techniques to mitigate this effect. Thus, the need to find an effective way to exploit the few available falls data is evident.

The human fall classification system presented here extends in several regards a work by the same authors [20]. Both works employ a FAS to detect indoor human fall by using only few examples for training. In this work, we depart from the original neural network architecture to improve classification results, as later explained. The training and the evaluation are based on a dataset, that has been largely increased in size and environmental conditions to assess the algorithm on a more complex and realistic scenario. A thorough comparison is provided on the new dataset including supervised [21], unsupervised [22] and one-shot learning [20] techniques previously proposed by the authors.

The outline of the paper is the following: Section 2 presents an overview of the recent literature on fall detection algorithms based on environmental sensors and the principal works related to the techniques employed in this work. Section 3 motivates the proposed approach and presents the contribution of the paper. Section 4 describes the proposed fall detection algorithm. Section 5 describes the experiments conducted to evaluate the performance of the approach. Finally, Section 6 concludes the paper and presents future developments.

II. Related Works

As mentioned above, several fall detection systems have been presented in the literature, the majority of which are based on wearable accelerometers or smart cameras. For further details on these technologies, the reader can refer to the surveys mentioned above [4]–[9]. Here, we focus on solutions employing audio signals.

Among them, Cheffena [23] propose a supervised fall detection algorithm based on smartphone microphones. The falls were performed and recorded from different volunteers with a smartphone placed within 5 m from them. This system may not work when the person is far or in a different room. The author has evaluated different types of features and supervised algorithms, reaching an accuracy of 98% with spectrogram features as the input of an artificial neural network. Popescu et al. [15] proposed a 2-stage threshold-based method using a microphone array. The first step is to compute the energy of the acquired signal. Then, if the value exceeds a threshold, a sound localization is performed to remove possible false alarms. In the end, if the sound was detected from above a specific height, the alarm is removed. The human falls for testing were performed by only one stunt actor falling on a mattress. In [16], Khan et al. present an unsupervised algorithm based on a microphone array of two elements. The algorithm encompasses a source separation and localization block to reduce the effect of background noise. Then, an OCSVM was trained on MFCCs of non-fall events only, in order to distinguish normal sound events from abnormal ones. The authors validated the algorithm using simulated falls of persons only in presence of a television that produced the interfering sound. The results, given in terms of Area Under Curve, are 99.28% and 97.38% without interference and with 75% interference respectively. Collado et al. [24] present a comparison with 7 binary supervised machine learning methods, using 10 standard audio features like the energy of the signal, zero-crossing, spectral centroid, etc. They assessed the performance on a dataset composed of falls performed by a stunt actor. The non-fall class was represented by a human conversation and television background. Due to the strong classes unbalance, they have sub-sampled the non-fall class, getting the same number of instances of the fall class. In this context, a Logistic Regression approach achieved the best results of 93.3% in terms of F_1-Measure. Differently, Irtaza et al. [25] show a Support Vector Machine approach trained on Acoustic-Local Ternary Patterns features. Similarly to [24], the problem of an unbalanced dataset has been faced by undersampling the non-fall class. In this case, the non-fall class is represented by human activity sounds and falling of objects, while they have used human fall sounds recorded with the aid of human subjects.

Several hybrid approaches that use more than one sensor at a time are present in the state-of-the-art. For example, Zigel et al. [13] use an energy-based event detection algorithm in which the floor vibrations are monitored. When an event is detected, vibration and sound features are extracted from these events and classified based on a quadratic classifier that discriminates human fall from other events. The dataset is composed of sounds of dropped objects and SHF by using a mimicking doll. In [26] a solution based on wearable accelerometers and microphones has been proposed. The solution employs empirical rules to detect a fall and validate it combining the sound pressure information utilizing fuzzy logic. The fall instances for training have been performed by volunteer falling on a soft rubber foam mat to cushion the impact of falls.

As shown above, although the literature provides several supervised and unsupervised approaches, no solution has been proposed exploiting one-shot learning for fall detection, to fill

the gap between simulated falls and scarcely available real human falls. One-shot or few-shot methods have been recently revived in other application fields. The Siamese approach was introduced by Bromley et al. [27] for signature verification and later also used in [28] for face verification. Both works are based on a supervised framework. Regarding the one-shot learning approach, the Siamese framework was first employed by Koch et al. [29] for image recognition. In [30], an attention mechanism over a learned metrics is used. In that work, the authors propose so-called Matching Networks trained by showing only a few examples per class for each minibatch in order to mimic the few-shot task by subsampling classes in a meta-learning perspective. In the audio field, one-shot approaches have been rarely used up to now. Lake et al. [31] proposed a hierarchical Bayesian acoustic-based approach to model the way a person learns a word of a new language from a few examples. They use a Hierarchical Hidden Markov model that induces the set of phone-like acoustic units directly from the raw unsegmented speech data in a completely unsupervised manner, identifying segments that should be clustered together and learning a set of phone-like acoustic units for the language. Manocha et al. [32] proposed a method based on Siamese networks for audio Content-based Representations.

A. Motivation and Contribution

As mentioned above, the fall detection task is very challenging due to the difficulty in retrieving examples for human fall modeling. Falls simulated by using a dummy may not represent properly real human falls, because they cannot recreate falls in which arms are used to mitigate the impact. Moreover, the use of protections, such as mattresses, knee pads or foam during the acquisitions of falls performed by volunteers, can significantly modify the samples, especially in the audio field.

The contribution of this work is threefold: first, we introduce a different computational intelligence architecture to improve detection. Then, we augment the dataset presented in [20] to assess fall detection methods in a complex scenario. Finally, we compare the new method with previous methods on the new dataset.

Our previous work [20] was based on twin convolutional neural networks trained as a Siamese neural network. In this architecture, the networks share the last layer used for computing the distance between their outputs, and they are trained by using the contrastive loss as a cost function in order to minimize the distance among positive samples and maximize the distance among negative samples. In this work, we modify the architecture and the training procedure. We exploit a neural network to extract low-dimensionality information that is fed to a classifier. Specifically, a Siamese Convolution Autoencoder (SCAE), composed of twin convolutional autoencoders is employed. Its latent space is forced to learn a metric between sample pairs. This, in turn, requires introducing an additional regularization term. The role of the neural network, thus, is to compress the information into a low-dimensionality space that allows efficient classification, demanded to a k-NN classifier. Furthermore, proper selection of training pairs allows one-shot learning and mapping of simulated falls into real falls, applying a transformation directly into the latent space. The network, thus, learns to generate similar outputs with either real or simulated falls, increasing the reliability of the classifier, leveraging the higher availability of simulated falls.

For what concerns the dataset, the one used in our previous work [20] was composed of recordings of falling objects, daily life sounds and human falls simulated with a manikin. These were recorded in a small empty room with stoneware tile floor. The new recordings have been performed in two additional rooms with different geometry, propagation, and absorption characteristics. These have been selected for their reduced propagation of the waves to the FAS: one room is paved with a fitted carpet floor, while in the other, the FAS is placed beyond a soundproof wall. We also recorded other objects and daily life sound types in addition to those present in the previous dataset. Finally, real human falls were reproduced by volunteers without additional protections. This dataset allows a more exhaustive experimental evaluation of the Siamese approach highlighting its effectiveness in a one-shot learning framework with respect to other state-of-the-art methods.

III. Dataset

The performance of the proposed approach has been evaluated on a corpus of audio events corresponding to falls of several objects and daily life sounds recorded in different conditions and rooms[1]. The dataset used in a previous work [20], created by the authors and hereafter named A3Fall-v1.0, has been extended to form a more complete one. In this section, a detailed description of the extended dataset will be given, from now on, referred to as A3Fall-v2.0.

A. Recording Setup

Since this is an extension of a dataset already created by the authors, the same instrumentation has been used. The recording equipment comprises a Presonus AudioBox 44VSL sound card connected to a laptop and two types of microphones:

- the FAS previously introduced is a special device designed to capture efficiently the audio waves transmitted through the ground. Briefly, it is composed of a membrane in direct contact with the floor. Thus, an inner container amplifies the vibrations which are then captured by a microphone. For further details and in-depth analysis, please refer to [33];
- a linear array of three aerial microphones, not used in this work.

Both the FAS and the aerial microphones are based on AKG BL 400 prepolarized condenser microphones. These have a frequency range of 40-14 kHz ($\pm$10 dB) and 13.5 mV/Pa sensitivity (at 1 kHz). They introduce 1% THD at 115 dB-A and their SNR is 62 dB-A. Signals were sampled at 44.1 kHz with a resolution of 32 bits.

The A3Fall-v1.0 dataset was collected in a room, hereafter named R0, obtained from a cantilever beam structure, thus, particularly suitable for the propagation of acoustic waves.

[1]The data set is available upon request to the authors.

Differently, the recordings of A3Fall-v2.0 have been performed in 2 different rooms with the following characteristics:

- the first is a university auditorium room (R1) in which the flooring is covered with carpet. This makes it particularly suitable for evaluating system performance on surfaces with acoustical behavior that can reduce the impact sound transmitted through the floor and in the air; all the recordings were performed near the auditorium stage in an area of 8×3 m.
- a recording studio (R2) was selected as the second room for its particular characteristics. Here, it was possible to make the acquisitions by placing the sensors in the live room while the audio events were performed in the control room. In particular, the sensors were positioned immediately behind the soundproof wall with the window overlooking the live room. The size of the live room is 5×7 m, while the size of the control room is 3×8 m.

All SHFs were recorded in R0, while all RHFs were recorded in R1 and R2. This simulates a real-world application, with R0 being a laboratory room, where a large number of recordings can be obtained easily; R1 and R2 being deployment rooms, where only a few recordings, including one RHF, can be obtained. A one-shot fall detection system can, thus, be deployed to several rooms without the need to build a dataset as large as it would be required by a completely supervised approach.

B. Composition

As previously mentioned, the A3Fall-v2.0 dataset includes the A3Fall-v1.0. The dataset is therefore composed of recordings realized in 3 different rooms. In Table I the composition of the dataset is summarized. As it can be seen, the same objects used in A3Fall-v1.0 were also used in the new dataset. Moreover, in the R1 and R2 other every-days objects have been recorded, for a total of 12 different object fall classes and 1420 instances. The manikin doll has been used only in R0: here the manikin was dropped 44 times, 31 of which form upright position while in the remaining ones, it was overturned from a chair. In R1 and R2 a total 80 human falls have been performed by 4 people. These falls were performed in different ways: forward, backward and on the side, trying to use the arms to cushion the fall. As in R0, also in R1 and R2 all events were performed from 1, 2, 4 and 6 m away from the FAS. As shown in Table I daily life sounds have been recorded in both rooms, which include: human activities as, i.e., footsteps, human and phone conversation, dragging objects and so on; classic, rock and pop music played from loudspeakers; television shows like newscast and satiric. In this work we did not include pet sounds or noise generated by outdoor events such as sirens, thunders and cars. The SNR was evaluated as the ratio between the power of the signal (falls and daily life) and the noise floor introduced by the recording devices. The average SNR for objects and human falls is 29 dB, while for the daily life sounds is 10 dB.

TABLE I: Composition of the A3Fall-v2.0 dataset.

Class	R0	R1	R2
	Nr. of occurrences		
Basket	64	40	40
Fork	64	40	40
Ball	64	40	40
Book	64	40	40
Bag	64	30	40
Chair	96	40	40
Table	0	40	40
Guitar Slide	0	40	40
Nipper	0	40	40
Keys	0	40	40
Hook	0	40	40
Coat Hook	0	40	40
Manikin Doll	44	0	0
Human Fall	0	40	40
	Total length (s)		
Daily life	2530	9055	5550

IV. Proposed Method

The proposed fall classification system is composed of three main parts that will described in this section. First, the features are extracted from a raw audio file and later they are used to train a Siamese Neural Network. The network embeddings, or latent space, learns a metric that is used by a k-NN classifier to discriminate human falls from non-human falls.

A. Feature Extraction Stage

In the feature extraction stage, the raw audio signals are processed to extract Log-Mel coefficients. Such features have been chosen for their popularity in computational audio analysis [34]–[36]. The first steps for obtaining Log-Mels consist in dividing the signal in frames 40 ms long and overlapped by 20 ms, and applying the Fast Fourier Transform to them. Then, each frame is filtered with a filter-bank composed of 40 triangular filters equally spaced in the mel-space, and the energy of each band is calculated. The final coefficients are obtained by applying the logarithm operator to each energy value. This results in a $40 \times N$ matrix $\mathbf{X}$ that represents the input to the neural network, where N is the number of frames.

B. Metric Learning Stage

The second stage is based on a Siamese Neural Network for learning a non-linear similarity metric. The SNN is directly trained on semantic similarity information and aims at modeling the relationships between classes in order to extract more robust features. The proficiency of a SNN mostly depends on the objective function used to train the network as well as the training set selection strategy. Part of our contribution consists in defining these two aspects.

The proposed neural network architecture, depicted in Fig. 1, consists in a Siamese Convolutional Autoencoder. The Siamese architecture comprises twin convolutional autoencoders that share both the topology and the weights values. As described later, their difference is that in the training phase they are shown two different examples of the training set. Each convolutional autoencoder is composed of an encoder that applies a transformation to the input and projects it into the latent space and a decoder that performs the reverse operation. The encoder, thus, represents a parametric function $\mathbf{S}_e(\cdot) : \mathbb{R}^{40\times N} \rightarrow \mathbb{R}^M$, while the decoder the function $\mathbf{S}_a(\cdot)$:

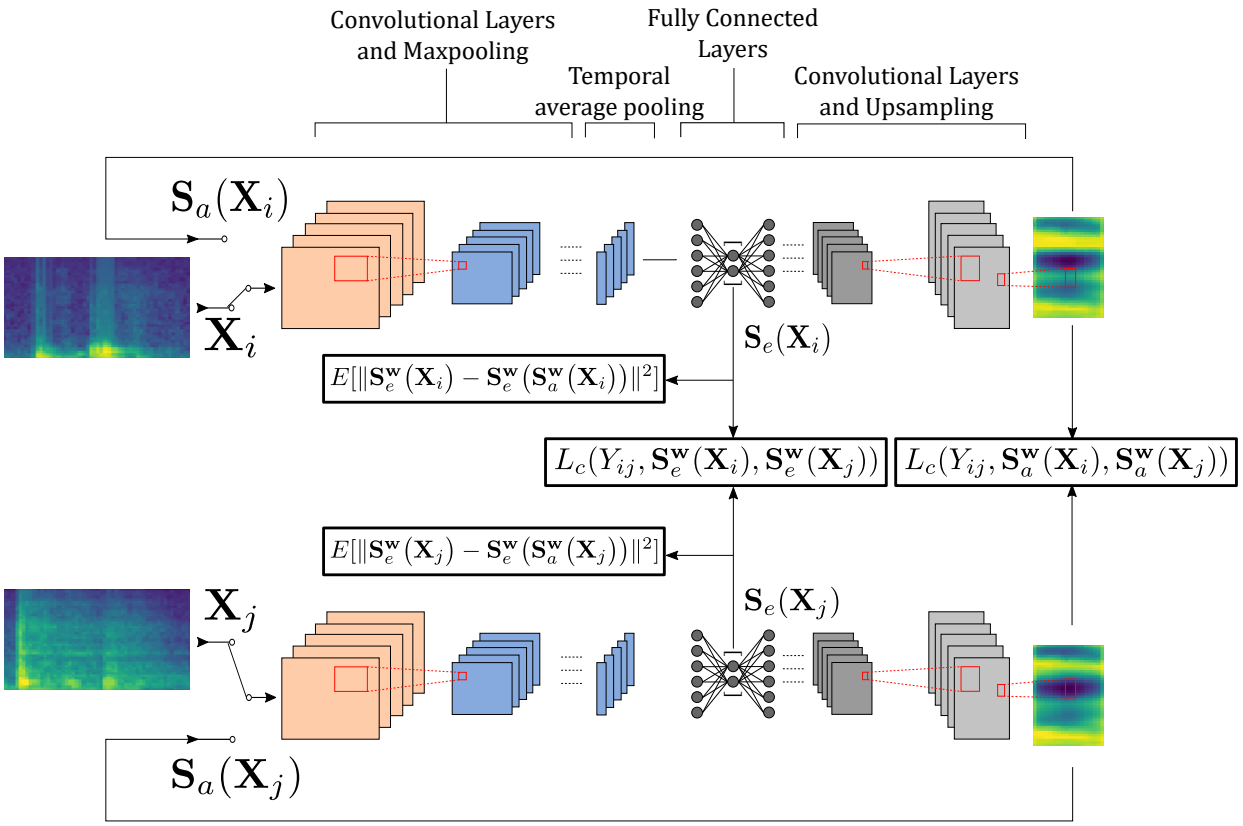


Fig. 1: The architecture of the SCAE network for metric learning and robust feature extraction. The loss terms used for training the network are shown.

$\mathbb{R}^M \rightarrow \mathbb{R}^{40\times N}$, where M is the dimension of the vector at the output of the encoder. The encoder includes convolutional layers alternated with max-pooling layers, followed by fully connected layers, and it ends with a hidden layer representing the mapping of the inputs. Before the fully connected layers, average pooling is applied along the time dimension of the features map related to the last convolutional layer. This makes the network independent of the temporal length of the input signals. The decoder part is mirrored with respect to the encoder.

Denoting with $\mathbf{X}_i$ the Log-Mel matrix extracted from the i-th audio signal and with $\mathcal{I} = \{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_P\}$ the set of Log-Mel matrices used for training, the objective of the SCAE is learning a projection metric $\mathbf{S}_e(\cdot) : \mathbb{R}^{40\times N} \rightarrow \mathbb{R}^M$ from pairs of positive and negative examples $(\mathbf{X}_i, \mathbf{X}_j)$ with $i \neq j$. Each pair is assigned a corresponding label Y_{ij}, whose value is 0 when $\mathbf{X}_i$ and $\mathbf{X}_j$ are from the same distribution (positive example) and is 1 otherwise (negative example). In the training phase, $\mathbf{X}_i$ and $\mathbf{X}_j$ are used respectively as the input to the first and second autoencoder of the Siamese architecture. The two sets $\mathcal{P}$ and $\mathcal{N}$ defined below represent respectively all the pairs of positive and negative examples in the set $\mathcal{I}$:

$$\mathcal{P} = \{(\mathbf{X}_i, \mathbf{X}_j) : i \neq j, \mathbf{X}_i, \mathbf{X}_j \in \mathcal{I} \text{ come from same distribution}\}, \quad (1)$$

$$\mathcal{N} = \{(\mathbf{X}_i, \mathbf{X}_j) : i \neq j, \mathbf{X}_i, \mathbf{X}_j \in \mathcal{I} \text{ come from different distributions}\}. \quad (2)$$

The training of the SCAE is performed on the set of pairs $\mathcal{T} = \mathcal{P} \cup \mathcal{N}$ and it consists in finding a set of weights $\mathbf{w}$ of the SCAE that minimizes the following loss function:

$$\begin{aligned} L^{\mathbf{w}}(Y_{ij}, (\mathbf{X}_i, \mathbf{X}_j)) = {} & L_c(Y_{ij}, \mathbf{S}_e^{\mathbf{w}}(\mathbf{X}_i), \mathbf{S}_e^{\mathbf{w}}(\mathbf{X}_j)) \\ & + L_c(Y_{ij}, \mathbf{S}_a^{\mathbf{w}}(\mathbf{X}_i), \mathbf{S}_a^{\mathbf{w}}(\mathbf{X}_j)) \\ & + E[\|\mathbf{S}_e^{\mathbf{w}}(\mathbf{X}_i) - \mathbf{S}_e^{\mathbf{w}}(\mathbf{S}_a^{\mathbf{w}}(\mathbf{X}_i))\|^2] \\ & + E[\|\mathbf{S}_e^{\mathbf{w}}(\mathbf{X}_j) - \mathbf{S}_e^{\mathbf{w}}(\mathbf{S}_a^{\mathbf{w}}(\mathbf{X}_j))\|^2]. \end{aligned} \quad (3)$$

The first term represents the contrastive loss function [28] calculated at the end of the encoder network and it has the following form:

$$\begin{aligned} L_c(Y_{ij}, \mathbf{S}_e^{\mathbf{w}}(\mathbf{X}_i), \mathbf{S}_e^{\mathbf{w}}(\mathbf{X}_j)) = {} & (1 - Y_{ij})\frac{1}{2}(D^{\mathbf{w}})^2 \\ & + Y_{ij}\frac{1}{2}\{(\max(0, m - D^{\mathbf{w}})\}^2, \end{aligned} \quad (4)$$

$$D^{\mathbf{w}} = \|\mathbf{S}_e^{\mathbf{w}}(\mathbf{X}_i) - \mathbf{S}_e^{\mathbf{w}}(\mathbf{X}_j)\|, \quad (5)$$

where $D^{\mathbf{w}}$ is the Euclidean distance between the two mappings performed by the encoder. The term $m > 0$ is the *margin* that makes pairs from different distributions (i.e., with $Y_{ij} = 1$) contribute only if their distance is greater than m.

The loss function comprises also three terms that include $\mathbf{S}_a^{\mathbf{w}}(\cdot)$, i.e., the output of the autoencoder. The first is the contrastive loss function between the two signals reconstructed at the end of the autoencoder, that is $\mathbf{S}_a(\mathbf{X}_i)$ and $\mathbf{S}_a(\mathbf{X}_j)$. This is possible thanks to the average pooling layer previously mentioned. In fact, by using this layer, the autoencoder reconstructs a signal with fixed time dimension regardless of the length of the inputs, allowing the Euclidean distance $D^{\mathbf{w}} = \|\mathbf{S}_a^{\mathbf{w}}(\mathbf{X}_i) - \mathbf{S}_a^{\mathbf{w}}(\mathbf{X}_j)\|$ to be computed. An alternative solution would have been to equalize the dimension of the inputs by zero-padding them. The reconstruction of the zero-padded portion, however, would have biased the value of the Euclidean distance and have a detrimental effect in the training phase.

The introduction of the temporal average pooling layer prevents the autoencoder to reconstruct the input signal. This is a fundamental feature that forces the autoencoder to engage in robust feature learning. In the proposed method, this behavior has been encouraged by introducing the two MSE terms $E[\|\mathbf{S}_e^{\mathbf{w}}(\mathbf{X}_i) - \mathbf{S}_e^{\mathbf{w}}(\mathbf{S}_a^{\mathbf{w}}(\mathbf{X}_i))\|^2]$ and $E[\|\mathbf{S}_e^{\mathbf{w}}(\mathbf{X}_j) - \mathbf{S}_e^{\mathbf{w}}(\mathbf{S}_a^{\mathbf{w}}(\mathbf{X}_j))\|^2]$ that force the network to produce the same representation when the input is $\mathbf{X}_i$ (respectively $\mathbf{X}_j$) or its reconstruction $\mathbf{S}_a^{\mathbf{w}}(\mathbf{X}_i)$ (respectively $\mathbf{S}_a^{\mathbf{w}}(\mathbf{X}_j)$).

As previously mentioned, another crucial aspect of SNN is the selection of training pairs. The fall detection system should be able to work reliably with as few RHF examples as possible. It is, thus, necessary to train the network to take full advantage of the limited available information. The similarity between SHF and RHF can be exploited, since the cardinality of the SHF set is higher. Several strategies for pairs selection can be envisioned. Let $\mathcal{F}$ be the set of real and simulated human falls, and $\mathcal{O}$ be the set of all other samples, i.e., those coming from the daily life sounds and objects falls distributions. We can compose the following sets:

- $\mathcal{P}_{\mathcal{F}}$, i.e., the positive samples composed only of samples in $\mathcal{F}$;
- $\mathcal{N}_{\mathcal{F}}$, i.e., the negative samples composed of a sample in $\mathcal{F}$ and a sample in $\mathcal{O}$;
- $\mathcal{P}_{\mathcal{O}}$, i.e., the positive samples composed only of samples in $\mathcal{O}$;
- $\mathcal{N}_{\mathcal{O}}$, i.e., the negative samples composed of samples in $\mathcal{O}$ belonging to different distributions.

We can now define four pairs selection strategies as:

- $\mathcal{P}$-$\mathcal{N}$-PAIRS strategy: the network is trained with $\mathcal{P} = \mathcal{P}_{\mathcal{F}} \cup \mathcal{P}_{\mathcal{O}}$ as positive examples and $\mathcal{N} = \mathcal{N}_{\mathcal{F}} \cup \mathcal{N}_{\mathcal{O}}$ as negative examples;
- $\mathcal{P}$-PAIRS strategy: the network is trained with $\mathcal{P} = \mathcal{P}_{\mathcal{F}} \cup \mathcal{P}_{\mathcal{O}}$ and $\mathcal{N} = \mathcal{N}_{\mathcal{O}}$;
- $\mathcal{N}$-PAIRS strategy: the network is trained with $\mathcal{P} = \mathcal{P}_{\mathcal{O}}$ and $\mathcal{N} = \mathcal{N}_{\mathcal{F}} \cup \mathcal{N}_{\mathcal{O}}$;
- NO-PAIRS strategy: the network is trained only with $\mathcal{P} = \mathcal{P}_{\mathcal{F}}$ and $\mathcal{N} = \mathcal{N}_{\mathcal{O}}$.

In the last case the SHF and RHF samples are used only for training the classifier, later introduced.

The best pairs selection strategy should allow the network to learn identifying RHF and SHF as one class, and to project real falls, during normal operation, in the hyper-plane region that was assigned to RHF and SHF. The MSE regularization term should allow the network to learn this mapping in the latent space.

C. Classification Stage

The latent space of the network provides information to a metric-based classifier that discriminates falls from non-falls. Specifically, the entire training set is transformed using the encoder function $\mathbf{S}_e(\cdot)$. Moreover, we apply this transformation also to some instances of SHF previously left out of the training set of the SCAE, thus obtaining a total number of templates for the human fall equal to

$$\mathcal{T}_{hf} = \mathcal{T}_{shf} + \mathcal{T}_{rhf}, \tag{6}$$

with $\mathcal{T}_{shf}$ the number of SHF from R0 and $\mathcal{T}_{rhf}$ the total number of RHF templates selected from R1 and R2 used in SCAE training, two in our case. To train the k-NN classifier, a set of templates composed of $\mathcal{T}_{hf}$ instances has been selected for each other class in order to obtain a balanced training set. Besides, the parameter *K* of the classifier has been set to $\mathcal{T}_{hf}$. Finally, a human fall is detected if there is at least one human fall template in the set of $\mathcal{T}_{hf}$ neighbors related to the sample under test at that moment. This classification technique has been used to reduce the miss rate, which is of greater importance compared to false alarm rate in fall detection applications.

V. Comparative methods

In this section, the methods compared with the proposed work are summarized. The first method is based on a binary SVM. It uses a GMM, trained on a large corpus of audio events with the Expectation Maximization algorithm to model the acoustic space (UBM). Then, for each audio segment, the MAP algorithm is used to calculate a GMS from MFCCs. Further details are given in [21]. This method is employed with two datasets, a balanced training set (simply called SVM from now on) and an unbalanced training set (One-shot-SVM from now on) for direct comparison with the proposed approach. A second comparative method is the unsupervised variation of the previous one based on OCSVM [22]. The third method is the Siamese approach reported in [20], from now on, called *Original Siamese*. It consists of a simple SNN instead of SCAE thus equivalent to the encoded part of the proposed autoencoder architecture, but without the average pooling layer preceding the fully connected layers. In [20], the algorithm was evaluated on a simpler scenario as several human falls were used during training. In this work, the method operates in a one-shot learning framework. Since SHFs were not used in this method, the pairs generation technique consists in the combination of the non-human fall data and the available template of RHF in order to compose the positive $\mathcal{P}$ and negative $\mathcal{N}$ as indicated in Eq. 1 and Eq. 2. Furthermore, a threshold-based classifier is used. A human fall is detected if the sample is mapped within a radius from a real human fall template.

VI. Experiments

This section presents the results of the experimental evaluation. Firstly, we describe the creation of the datasets for each one of the compared methods. Then we present preliminary experiments related to the pair selection strategy. These set of experiments give insights on the embedding of real and simulated falls in the latent space. Finally, the best pair selection strategy is taken, and a random search is performed to optimize the classification performances.

All the experiments have been performed on the dataset described in Section III, but signals have been downsampled to 8 kHz since the majority of their energy is concentrated below 4 kHz, as discussed in [33]. Moreover, their resolution has been reduced to 16 bits. All the following experiments have been conducted with 120000 pairs on average between folds for training the SCAE. Results are expressed in terms of F_1-Measure, calculated from the normalized confusion matrix, cumulative of all the folds. The same metric has also been used to optimize the results shown in Section VI-C. This choice was made to give more weight to false negatives than false positives, as the test set is highly unbalanced, being composed from 6973 non-human fall events and 390 human fall events in total. In particular, since the daily life recordings have been divided into segments of 5 seconds each, the non-fall events are composed of 5275 daily life sounds instances and 1698 object fall events.

A. Data Splitting

Firstly, the A3Fall-2.0 dataset has been split into 5 folds for cross-validation: in particular, the data related to the R0 room without SHFs have been used only for training and used in each fold. Differently, the samples related to R1 and R2 without RHFs have been split into 5 folds with 20% for test and 80% for the training set. Both simulated and real human falls have been treated differently, based on the algorithm under examination:

- SCAE: for the proposed approach, one RHF per room has been randomly selected for each fold and then added to the related training set. Differently, SHFs have been split in 5 folds with 80% for train the SCAE, while the remaining 20% has been left out from the training set of the Siamese network but used only to train the classifier as explained in Section IV-C. The pairs for training the SNN have been generated keeping balanced all the combinations between the classes.
- OCSVM: since this is a completely unsupervised method, both real and simulated human falls have been removed from the training set.
- SVM: since this is a completely supervised method, the RHFs have been split in 5 folds with 20% for test and 80% for train and then added to the respective sets.
- One-shot-SVM: in order to keep this experiment comparable with the proposed method, the same selection carried out for the SCAE has been used for training the SVM, i.e., with just one real human fall sample for each environment to monitor.

TABLE II: Hyper-parameters used in the preliminary experiments, and their value.

Parameter	Value	Parameter	Value
CNN layer Nr.	3	Drop rate	0%
Kernel shape	[4×4, 4×4, 4×4]	CNN Padding	Same
Kernel Nr.	[4, 4, 4]	Batch Size	512
MLP layers Nr.	3	MLP Act.	ReLU[2]
MLP layers dim.	[40, 512, 2]	Optimizers	Adadelta
Max pool shape	[1×2, 2×3, 2×3]	Weight Initializers	Glorot Uniform

- *Original Siamese*: the same sets used for SCAE have also been used for this approach. The only difference is that the SHFs were not used because they are not contemplated by this method.

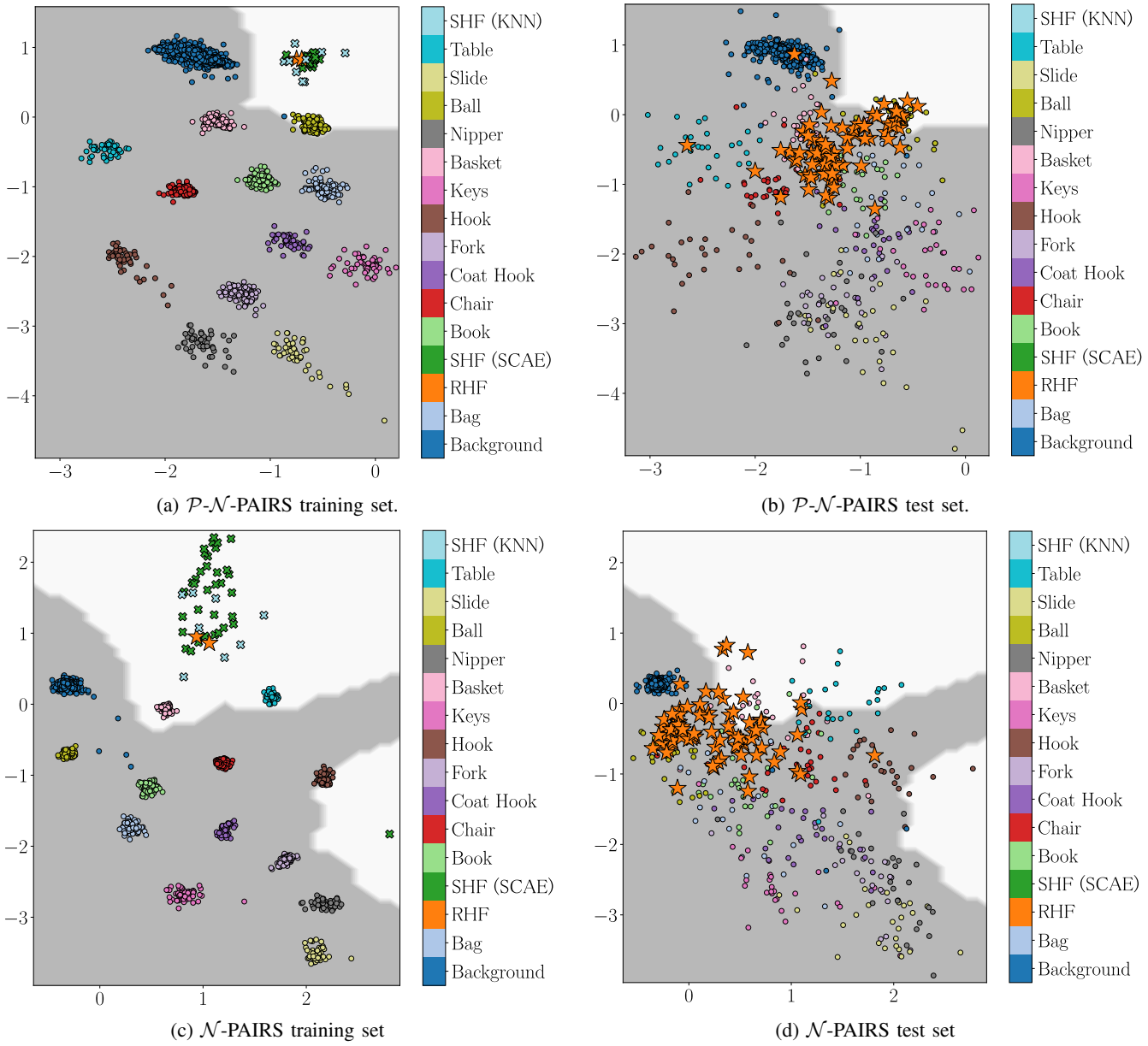
B. Preliminary Experiments

Several strategies have been introduced for pairs selection. Preliminary experiments aimed at studying their influence on the templates generated in the latent space and fed to the classifier. Experiments have been performed with a fixed autoencoder architecture, having a hidden layer composed of 2 neurons to simplify visualization of the mapping between input samples and the latent space. Table II reports the hyper-parameter of that network. Figures 2 and 3 show how training and test samples are encoded by the network, after training is completed, according to the four pairs selection strategies. The mappings in Fig. 2a, Fig. 2c, Fig. 3a and Fig. 3c are then used to train the related k-NN classifier. The final decision boundaries are reported in all figures: the data found in the white area are classified as human fall. Comments related to the four strategies follow:

- by using $\mathcal{P}$-$\mathcal{N}$-PAIRS (Fig. 2(a,b)), the network manages to cluster distributions during training, however, it does not perform equally well with the test set and maps RHFs to a different area. Knowledge of a few human falls is not exploited properly;
- by using the $\mathcal{N}$-PAIRS strategy (Fig. 2(c,d)), the contrastive loss tries to increase the distance between human fall instances and all other classes, but without grouping them together (no positive examples were generated). This results in poor classification performance;
- by using NO-PAIRS (Fig. 3(a,b)), the SCAE spreads the simulated human fall signals in the hyperplane (Fig. 3a), thus, the clustering operated by the classifier leads to too many false alarms as shown in Fig. 3b;
- finally, by using $\mathcal{P}$-PAIRS (Fig. 3(c,d)), the SCAE clusters RHFs and SHFs, thus, learning an efficient representation. The SHFs left out of the SCAE training can be used as additional templates for training the k-NN classifier.

Overall, the last strategy seems to obtain best results. The results for these preliminary experiments are reported in Table III.

[2]In the decoder, an additional CNN layer with $tanh$ activation function has been used to ensure a good reconstruction.



(a) $\mathcal{P}$-$\mathcal{N}$-PAIRS training set.

(b) $\mathcal{P}$-$\mathcal{N}$-PAIRS test set.

(c) $\mathcal{N}$-PAIRS training set

(d) $\mathcal{N}$-PAIRS test set

Fig. 2: Training (a,c) and test (b,d) samples projected in the latent space by the encoder, colored according to their class. RHF samples are shown as a star, some SHF are used for training the SCAE (green crosses, SHF (SCAE)) while some are only used for training the k-NN classifier after passing through the encoder (turquoise crosses, SHF (KNN)). Samples falling in the white area are classified as human falls.

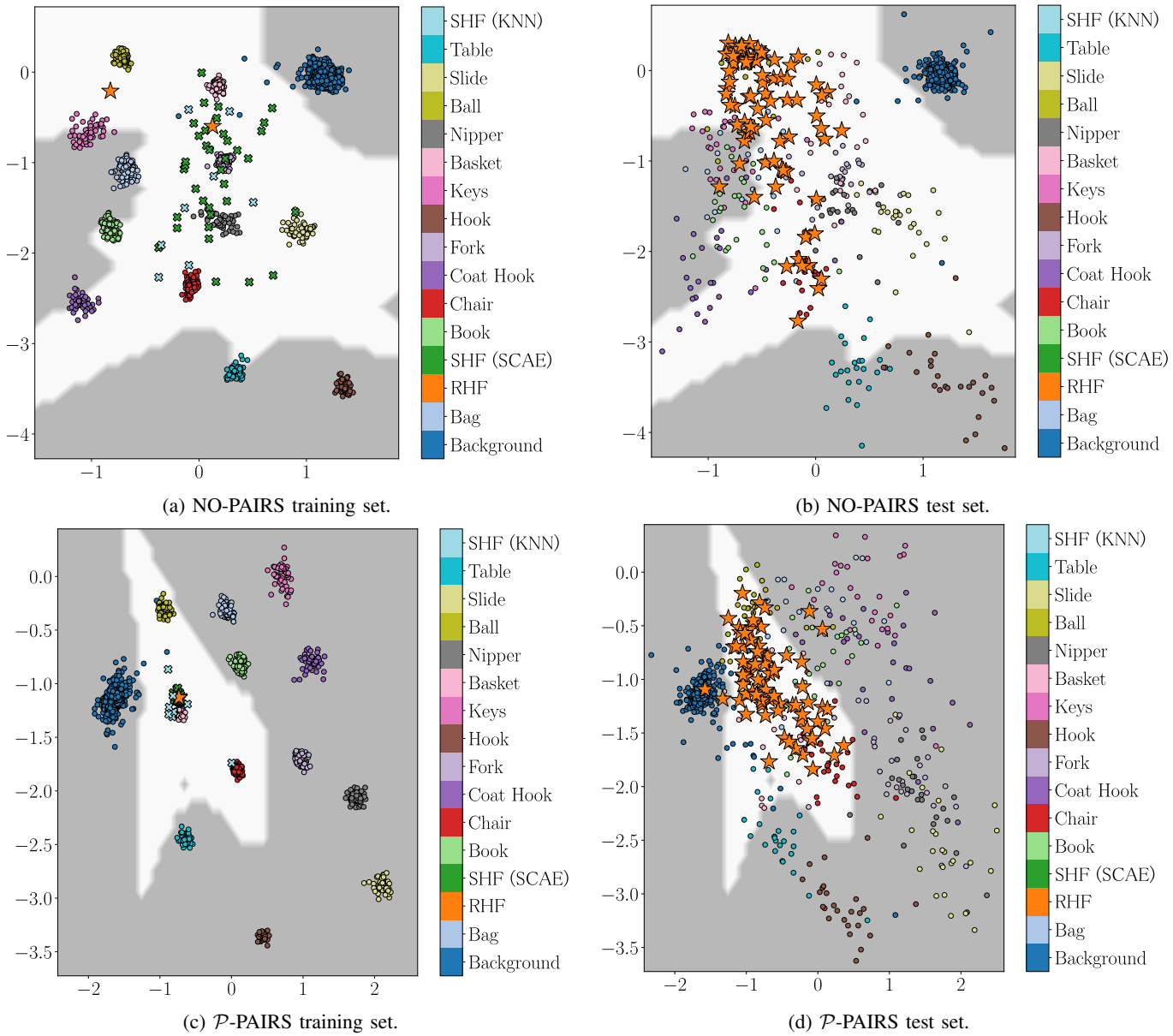
TABLE III: Preliminary F$-$1-Measure results for different pairs generation strategies.

Technique	Result in R1	Result in R2	Overall
$\mathcal{P}$-$\mathcal{N}$-PAIRS	55.17%	64.74%	60.13%
$\mathcal{N}$-PAIRS	76.20%	67.53%	72.05%
NO-PAIRS	91.71%	89.88%	90.97%
***P*-PAIRS**	92.54%	92.54%	**92.54%**

C. Optimized results

Considering the results of the preliminary experiments, a random-search of 50 different configurations was performed, according to Table IV, to optimize the hyper-parameters of the SCAE approach with the $\mathcal{P}$-PAIRS strategy. The same random-search was performed for the *Original Siamese* method, also optimizing the radius of the classifier used with this approach. For the SVM based methods a grid-search strategy has been adopted to optimize the parameters. In particular the parameters assumed values in the ranges $\{2^{-5}, 2^{-3}, \ldots, 2^{15}\}$ for C (SVM) and ν (OCSVM), $\{2^{-15}, 2^{-13}, \ldots, 2^{3}\}$ for γ (both SVM and OCSVM) and $\{1, 2, \ldots, 64\}$ for the number of mixtures of the UBM.

Fig. 4 shows the results obtained for each approach. The completely supervised SVM method is not directly comparable with the others due to different training and test set, but it is reported for completeness. Although the dataset is balanced, the performance is significantly lower compared to the other methods. Moreover, it is evident that using the extremely



(a) NO-PAIRS training set.

(b) NO-PAIRS test set.

(c) $\mathcal{P}$-PAIRS training set.

(d) $\mathcal{P}$-PAIRS test set.

Fig. 3: Training (a,c) and test (b,d) samples projected in the latent space by the encoder, colored according to their class. RHF samples are shown as a star, some SHF are used for training the SCAE (green crosses, SHF (SCAE)) while some are only used for training the k-NN classifier after passing through the encoder (turquoise crosses, SHF (KNN)). Samples falling in the white area are classified as human falls.

TABLE IV: Hyper-parameters optimized in the random-search phase and their range.

Parameter	Range	Distribution
CNN layer Nr.	[1-3]	Uniform
Kernel shape	[1x1-8x8]	Uniform
Kernel Nr.	[1-32]	Uniform
MLP layers Nr.	[1-2]	Uniform
MLP layers dim.	[1-4096]%	Log-uniform
Max pool shape	[0x0-3x3]	Uniform
Drop rate	[0-0.2]%	Uniform

unbalanced dataset for a supervised approach, as the one used for the Siamese network, leads to a very large degradation of the performance. Indeed, the One-Shot SVM reaches an overall F_1-Measure of only 14.72%. In cases where an extremely unbalanced dataset is available, it is better to exploit a completely unsupervised method such as the OCSVM, achieving a score of about 72%. The best performing method is the SCAE that reaches a 93.58% of F_1-Measure, outperforming the *Original Siamese* of 3.25%. The improvement was significant for $p < 0.002$ according to one-tailed z-test [37]. The remarkable results obtained by both the *Original Siamese* and SCAE methods show that the use of Siamese framework is very powerful in this type of scenario, where limited real data is available but simulated data can be exploited. In Table V and Table VI the normalized confusion matrices for the Siamese based approach are reported, showing that the miss rate of the proposed method is less than 4% compared to the *Original Siamese* method. In terms of false alarm rate, it has increased

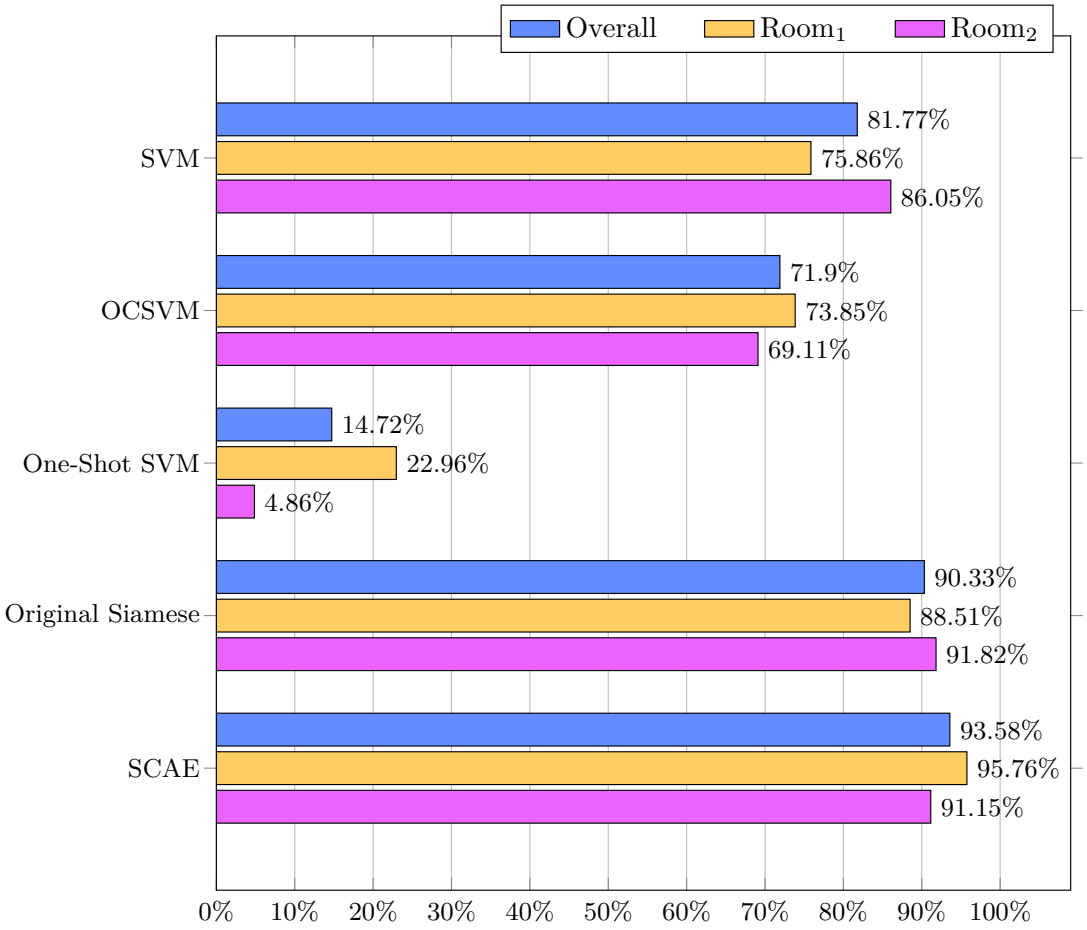


Fig. 4: F_1-Measure results for all compared methods.

TABLE V: Normalized confusion matrix of the *Original Siamese* approach. Absolute values are shown in brackets.

	Human Falls	**Objects**
Human Falls	90% (6283)	10% (690)
Objects	9% (37)	91% (353)

TABLE VI: Normalized confusion matrix of the SCAE approach. Absolute values are shown in brackets.

	Human Falls	**Objects**
Human Falls	91% (6362)	9% (611)
Objects	4% (17)	96% (373)

by 1%, resulting in a good reliability. Since there are many instances of daily life sounds in the dataset, the low number of false alarms indicates that this approach could also be used as a detection system.

VII. Conclusion

This paper described the extension of a previous work by the same authors [20] and its results. Among the novelties, a new data set has been collected starting from the one used in the previous work. The recordings of the original A3Fall-v1.0 dataset have been extended with new events recorded in two new rooms. Moreover, in order to test the system, the dataset was augmented with 80 human falls performed by four actors. In this article, the authors have shown that the proposed method outperforms the other four comparative methods and that the same algorithm may be used not only as a classifier but also as a detector. In this more realistic scenario, the preeminence on the Siamese framework for one-shot learning with respect to conventional methods has been shown. A further improvement in performance has been achieved with an extension of the method previously proposed in [20]. It is composed of 3 stages: Log-Mel feature extraction, metric learning employing a Siamese autoencoder neural network named SCAE and, in the end, a final decision stage based on a k-NN classifier. The network exploits the few information on the real fall by using a particular strategy of pairs generation for the SCAE training. In doing so, the system learns how to transform the available simulated human fall instances to create a more suitable set of templates that can be used to train the final classifier. Although the system seems to be reliable because of the low miss rate, the false alarm rate, of just about 3 false alarms raised every 2 real human falls, may even so be annoying for some users. To reduce this problem, several techniques could be employed. For instance, the system could be extended to include algorithms for fall recovery recognition able to detect whether a person is continuing his normal

activity or if he/she is still lying on the ground.

REFERENCES

[1] "Department of Health and Human Services: World's older population grows dramatically," http://www.who.int/en/news-room/fact-sheets/detail/falls, [Online; accessed 30-Oct-2018].

[2] G. Carone and D. Costello, "Can europe afford to grow old?" *Finance and Development*, vol. 43, no. 3, pp. 28–31, 2006.

[3] "World Health Organization: Falls," http://www.who.int/en/news-room/fact-sheets/detail/falls, [Online; accessed 30-Oct-2018].

[4] M. Mubashir, L. Shao, and L. Seed, "A survey on fall detection: Principles and approaches," *Neurocomputing*, vol. 100, pp. 144–152, 2013.

[5] S. S. Khan and J. Hoey, "Review of fall detection techniques: A data availability perspective," *Medical engineering and physics*, vol. 39, pp. 12–22, 2017.

[6] N. Lapierre, N. Neubauer, A. Miguel-Cruz, A. R. Rincon, L. Liu, and J. Rousseau, "The state of knowledge on technologies and their use for fall detection: A scoping review," *International journal of medical informatics*, 2017.

[7] N. Pannurat, S. Thiemjarus, and E. Nantajeewarawat, "Automatic fall monitoring: a review," *Sensors*, vol. 14, no. 7, pp. 12 900–12 936, 2014.

[8] T. Xu, Y. Zhou, and J. Zhu, "New advances and challenges of fall detection systems: A survey," *Applied Sciences*, vol. 8, no. 3, p. 418, 2018.

[9] N. El-Bendary, Q. Tan, F. C. Pivot, and A. Lam, "Fall detection and prevention for the elderly: A review of trends and challenges." *International Journal on Smart Sensing & Intelligent Systems*, vol. 6, no. 3, 2013.

[10] Q. Wu, Y. D. Zhang, W. Tao, and M. G. Amin, "Radar-based fall detection based on doppler time–frequency signatures for assisted living," *IET Radar, Sonar & Navigation*, vol. 9, no. 2, pp. 164–172, 2015.

[11] T. Liu, H. Yao, R. Ji, Y. Liu, X. Liu, X. Sun, P. Xu, and Z. Zhang, "Vision-based semi-supervised homecare with spatial constraint," in *Pacific-Rim Conference on Multimedia*. Springer, 2008, pp. 416–425.

[12] F. Werner, J. Diermaier, S. Schmid, and P. Panek, "Fall detection with distributed floor-mounted accelerometers: An overview of the development and evaluation of a fall detection system within the project ehome," in *Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2011 5th International Conference on*. IEEE, 2011, pp. 354–361.

[13] Y. Zigel, D. Litvak, and I. Gannot, "A method for automatic fall detection of elderly people using floor vibrations and soundproof of concept on human mimicking doll falls," *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 12, pp. 2858–2867, 2009.

[14] Y. Li, K. Ho, and M. Popescu, "A microphone array system for automatic fall detection," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 5, pp. 1291–1301, 2012.

[15] M. Popescu, Y. Li, M. Skubic, and M. Rantz, "An acoustic fall detector system that uses sound height information to reduce the false alarm rate," in *Proc. of the 30th International Conference of the Engineering in Medicine and Biology Society (EMBC)*, Vancouver, BC, Canada, Aug. 20-25 2008, pp. 4628–4631.

[16] M. S. Khan, M. Yu, P. Feng, L. Wang, and J. Chambers, "An unsupervised acoustic fall detection system using source separation for sound interference suppression," *Signal processing*, vol. 110, pp. 199–210, 2015.

[17] X.-X. Zhang, H. Liu, Y. Gao, and D. H. Hu, "Detecting abnormal events via hierarchical dirichlet processes," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 2009, pp. 278–289.

[18] M. Popescu and A. Mahnot, "Acoustic fall detection using one-class classifiers," in *Proc. of the Annual Int. Conf. of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Minneapolis, MN, USA, 2009, pp. 3505–3508.

[19] E. E. Stone and M. Skubic, "Fall detection in homes of older adults using the microsoft kinect," *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 1, pp. 290–301, 2015.

[20] D. Droghini, F. Vesperini, E. Principi, S. Squartini, and F. Piazza, "Few-shot siamese neural networks employing audio features for human-fall detection," in *Proceedings of the International Conference on Pattern Recognition and Artificial Intelligence*, ser. PRAI 2018. New York, NY, USA: ACM, 2018, pp. 63–69. [Online]. Available: http://doi.acm.org/10.1145/3243250.3243268

[21] D. Droghini, E. Principi, S. Squartini, P. Olivetti, and F. Piazza, "Human fall detection by using an innovative floor acoustic sensor," in *Multidisciplinary Approaches to Neural Computing*, A. Esposito, M. Faudez-Zanuy, F. C. Morabito, and E. Pasero, Eds. Cham: Springer International Publishing, 2018, pp. 97–107.

[22] D. Droghini, D. Ferretti, E. Principi, S. Squartini, and F. Piazza, "A combined one-class svm and template matching approach for user-aided human fall detection by means of floor acoustic features," *Computational Intelligence and Neuroscience*, vol. 2017, 2017, Article ID 1512670.

[23] M. Cheffena, "Fall detection using smartphone audio features," *IEEE journal of biomedical and health informatics*, vol. 20, no. 4, pp. 1073–1080, 2016.

[24] A. Collado-Villaverde, M. D. R-Moreno, D. F. Barrero, and D. Rodriguez, "Machine learning approach to detect falls on elderly people using sound," in *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*. Springer, 2017, pp. 149–159.

[25] A. Irtaza, S. M. Adnan, S. Aziz, A. Javed, M. O. Ullah, and M. T. Mahmood, "A framework for fall detection of elderly people by analyzing environmental sounds through acoustic local ternary patterns," in *Systems, Man, and Cybernetics (SMC), 2017 IEEE International Conference on*. IEEE, 2017, pp. 1558–1563.

[26] P. V. Er and K. K. Tan, "Non-intrusive fall detection monitoring for the elderly based on fuzzy logic," *Measurement*, vol. 124, pp. 91–102, 2018.

[27] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a" siamese" time delay neural network," in *Advances in Neural Information Processing Systems*, 1994, pp. 737–744.

[28] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, San Diego, CA, USA, Jun. 20-25 2005, pp. 539–546.

[29] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *ICML Deep Learning Workshop*, vol. 2, 2015.

[30] O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra *et al.*, "Matching networks for one shot learning," in *Advances in Neural Information Processing Systems*, 2016, pp. 3630–3638.

[31] B. Lake, C.-y. Lee, J. Glass, and J. Tenenbaum, "One-shot learning of generative speech concepts," in *Proceedings of the Annual Meeting of the Cognitive Science Society*, vol. 36, no. 36, 2014.

[32] P. Manocha, R. Badlani, A. Kumar, A. Shah, B. Elizalde, and B. Raj, "Content-based representations of audio using siamese neural networks," *arXiv preprint arXiv:1710.10974*, 2017.

[33] E. Principi, D. Droghini, S. Squartini, P. Olivetti, and F. Piazza, "Acoustic cues from the floor: a new approach for fall classification," *Expert Systems with Applications*, vol. 60, pp. 51–61, 2016.

[34] J. F. Gemmeke, L. Vuegen, P. Karsmakers, B. Vanrumste *et al.*, "An exemplar-based nmf approach to audio event detection," in *Applications of Signal Processing to Audio and Acoustics (WASPAA), 2013 IEEE Workshop on*. IEEE, 2013, pp. 1–4.

[35] A. Mesaros, T. Heittola, A. Eronen, and T. Virtanen, "Acoustic event detection in real life recordings," in *Signal Processing Conference, 2010 18th European*. IEEE, 2010, pp. 1267–1271.

[36] G. Parascandolo, T. Heittola, H. Huttunen, T. Virtanen *et al.*, "Convolutional recurrent neural networks for polyphonic sound event detection," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 6, pp. 1291–1303, 2017.

[37] M. D. Smucker, J. Allan, and B. Carterette, "A comparison of statistical significance tests for information retrieval evaluation," in *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*. ACM, 2007, pp. 623–632.