Information Fusion 101 (2024) 101984

Contents lists available at ScienceDirect

Information Fusion

journal homepage: www.elsevier.com/locate/inffus

Full length article

Spatio-temporal visual learning for home-based monitoring

Youcef Djenouri [a,b,c,*], Ahmed Nabil Belbachir [d], Alberto Cano [e], Asma Belhadi [f]

[a] *University of South-Eastern Norway, Kongsberg, Norway*
[b] *NORCE Norwegian Research Center, Oslo, Norway*
[c] *IDEAS, NCBR, Warsaw, Poland*
[d] *NORCE Norwegian Research Center, Grimstad, Norway*
[e] *Virginia Commonwealth University, Richmond, VA, USA*
[f] *OsloMet University, Oslo, Norway*

ARTICLE INFO

Keywords:
Hybrid Deep Learning
Home Based Monitoring
Spatio-temporal visual learning

ABSTRACT

This paper introduces a novel concept for Home-based Monitoring (HM) that enables robust analysis and understanding of activities towards improved caring and safety. Spatio-Temporal Visual Learning for HM (STVL-HM) is a novel method that learns from sensor data that is jointly represented in space and time in order to robustify the HM process. We propose a hybrid model based on a Convolution Neural Network (CNN) and Transformers. The CNN first learns the visual spatial features from various sensor data. The learned visual features are then fed into the transformer, which captures temporal information by observing the sensor status at various timestamps. STVL-HM has been tested using Kinetics-400, the real use case of human activity recognition scenario for HM data. The results reveal the clear superiority of the STVL-HM compared to the recent baseline HM solutions.

1. Introduction

Home-based Monitoring (HM) is the process of monitoring a person's health or well-being from their own home rather than in a clinical setting [1–3]. This type of monitoring can be especially useful for people who have chronic health conditions or who are recovering from a disease or injury, as it allows them to receive care and support while maintaining their independence. There are many different types of home-based monitoring that can be used to track a person's health and well-being: (1) Remote patient monitoring [4]: This involves using technology to remotely monitor a patient's vital signs, such as blood pressure, heart rate, and oxygen levels. This information can be transmitted to a healthcare provider who can monitor the patient's condition and intervene if necessary. (2) Telehealth [5–7]: This involves using video conferencing or other technologies to allow patients to communicate with healthcare providers from their own homes. This can be especially useful for people who have mobility issues or who live in remote areas, and (3) Wearable technology [8–10]: This involves using devices such as fitness trackers or smartwatches to monitor a person's activity levels, sleep patterns, and other health metrics. Deep learning can support home-based monitoring in automatizing the analysis of data collected from various sensors and devices, such as wearable sensors, health monitors, and IoT devices [11–13]. By using deep learning algorithms, it is possible to detect patterns and anomalies in the activities that might indicate deviations and changes in a person's health or well-being [14,15]. For example, a deep learning model could be trained to analyze the data collected from a wearable sensor that tracks a person's heart rate, breathing rate, and other vital signs. The model could be used to identify patterns in the data that suggest the person is experiencing a health issue, such as an irregular heartbeat or breathing difficulties. The model could then alert healthcare providers or family members to intervene and provide care. Sensor fusion is the process of combining data from multiple sensors in order to obtain a more accurate and complete picture of a system or environment [16,17]. This technique is commonly used in home-based monitoring to gather data from different types of sensors and devices in order to provide a more comprehensive view of a person's health and well-being [18,19]. For example, sensor fusion could be used to combine data from a wearable heart rate monitor, a blood pressure monitor, and a pedometer to provide a more complete picture of a person's cardiovascular health. By combining data from these different sensors, it is possible to detect patterns and anomalies in the data that might not be visible using a single sensor alone. Most HM-based solutions from sensor fusion are recently developed, including:

1. **Computer vision-based solutions**: Computer vision-based solutions can be used for HM to analyze data from visual sensors, such as cameras or motion sensors [20,21]. This can provide

[*] Corresponding author at: University of South-Eastern Norway, Kongsberg, Norway.
E-mail address: youcef.djenouri@usn.no (Y. Djenouri).

https://doi.org/10.1016/j.inffus.2023.101984
Received 16 April 2023; Received in revised form 4 August 2023; Accepted 23 August 2023
Available online 25 August 2023

valuable information about a person's movements, posture, and activities, which can be used to assess their health and well-being. For example, computer vision algorithms can be used to analyze video data from a camera to detect changes in a person's gait or posture, which might indicate mobility issues or balance problems. This can be particularly useful for monitoring elderly or disabled individuals who may be at risk of falls or other injuries. Computer vision-based solutions can also be used to monitor a person's activities of daily living, such as cooking, cleaning, and self-care activities. By using deep learning algorithms to analyze video data, it is possible to detect patterns in a person's activities that might indicate changes in their health or well-being. Solutions based on convolution neural networks [22], generative adversarial networks [23,24], and autoencoders [25] are examples of these solutions. CNNs and autoencoders are both widely used for learning temporal features. The choice between the two depends on the nature of the data, the specific objectives of the task, and the characteristics of the features we want to capture. CNNs are designed to be translation invariant, meaning they can recognize patterns regardless of their location in the input. This property is beneficial in home-based monitoring tasks, where the timing of events may vary, and the model needs to recognize temporal patterns at different time points. Autoencoders may not inherently possess this translation invariance property.

2. **Sequence-based solutions**: Sequence-based representation for human-based monitoring involves analyzing time series data, such as sensor data or physiological data, to extract meaningful information about a person's movements, activities, and behaviors over time. This can be used to assess their health and well-being, and to detect changes that might indicate a health problem or other issue. For instance, extracting relevant features from the time series data, such as the mean, standard deviation, or frequency components of the signal. These features can then be used as inputs to a machine learning model to detect patterns or changes in the data. Solutions based on recurrent neural networks [26], long-short term memory [27], and transformers [28] are examples of these solutions. Transformers and LSTM (Long Short-Term Memory) are both powerful architectures for home based monitoring. However, transformers have gained popularity and shown advantages over LSTM in learning temporal features [29–31]. Indeed, in home-based monitoring, the temporal relationships between different data points (e.g., physiological signals, time series data) can span over extended periods. Transformers, with their attention mechanism, are more adept at modeling such long-range dependencies compared to LSTMs. In addition, as the amount of data in home-based monitoring increases, Transformers can be more easily scaled to handle larger datasets, while LSTMs may become computationally expensive and less feasible.

All the above solutions are effectively used in real-world applications of HM and achieved promising results in monitoring human behaviors in real time. However, these solutions suffer from accuracy when handling similar human behaviors like drinking and eating. We assume that taking both spatial and temporal information features in the learning can radically improve the HM performances. Different hybrid architectures can be combined for learning spatial and temporal information including LSTM and GNN, however the hybrid CNN-transformer architecture is well-suited for handling multimodal data, such as combining image data (from cameras) with textual data (e.g., from smart home assistants or environmental sensors). CNNs handle visual data effectively, while transformers excel at sequential and textual data processing. Home based monitoring involves processing both image and textual data, the CNN-Transformer is therefore more suitable. We believe that a hybrid CNN and transformer network for home-based monitoring will be a potential architecture that can be used to process and analyze data from various sensors and devices commonly found in a home environment. Motivated by the success of Convolution Neural Networks (CNN) in learning the spatial visual features, and transformers in learning the temporal features, we propose a hybrid CNN and transformer based model to learn spatio-temporal interconnections between sensor readings for addressing the HM challenges. The main contributions of this research work are given in the following:

1. We present a novel concept for solving HM systems called Spatio-Temporal Visual Learning for Home-based Monitoring (STVL-HM). It learns from both spatial and temporal data sensors to improve the HM process.
2. We develop a hybrid model based on CNN and transformers. The CNN first learns the spatial features from the different sensor data. The learned features will be then injected into the transformer where the temporal information is captured by observing the sensor status at different timestamps.
3. We evaluate and analyze the performance of STVL-HM in a real-life use case of human activity recognition scenario for HM data compared with the baseline HM-based solutions. The results reveal the superiority of STVL-HM, where the achievement of 95% accuracy performance has been observed.

2. Home-based monitoring

Home-based monitoring refers to the use of devices, sensors, or other tools to monitor a person's health or well-being in their own home or a non-clinical setting. This type of monitoring is often used to help people manage chronic conditions, recover from an illness or injury, or simply to stay healthy. Home-based monitoring has many applications across different areas, including healthcare, wellness, and lifestyle. Here are some examples of applications of home-based monitoring:

1. Chronic Disease Management: Home-based monitoring can help patients with chronic diseases, such as diabetes, hypertension, and heart disease, to better manage their conditions. Patients can use devices to monitor their vital signs, such as blood pressure, blood sugar levels, or oxygen saturation, and share the data with their healthcare providers. Providers can use the data to adjust treatment plans and medication dosages.
2. Post-acute care: After being discharged from the hospital, patients can use home-based monitoring to track their recovery progress and prevent readmissions. Devices can monitor vital signs and symptoms, such as fever or pain, and alert healthcare providers of any changes. Providers can provide remote guidance and adjust treatment plans as needed.
3. Elderly Care: Home-based monitoring can help elderly individuals to age in place and maintain their independence. Devices can monitor activities of daily living, such as walking, bathing, or eating, and detect falls or emergencies. Caregivers or family members can receive alerts and provide assistance as needed.
4. Mental Health: Home-based monitoring can help individuals with mental health conditions, such as depression or anxiety, to monitor their symptoms and improve their overall well-being. Apps can track mood, sleep patterns, or stress levels, and provide personalized insights and recommendations.
5. Wellness and Lifestyle: Home-based monitoring can also help individuals to improve their overall health and well-being. Wearable devices can track physical activity, sleep, and nutrition, and provide personalized feedback and coaching. Apps can also monitor environmental factors, such as air quality or temperature, and provide recommendations for improving the indoor environment.

Human Activity Recognition (HAR) is one of the important steps of the home-based monitoring. It has the goal to recognize human activities. This study focuses on human-activity recognition in home-based monitoring setting. In the following, we will give a formal description of the human activity recognition problem:

Consider the set of human activities $\mathcal{A}$, a set of sensors $\mathcal{S}$. The data collected from $\mathcal{S}$ is defined in time windows $\mathcal{W}$. HAR problem can be considered as the following optimization problem:

$$\min_{\theta} \sum_{y^{\mathcal{W}}} \mathcal{L}(y^{\mathcal{W}}, y^{*\mathcal{W}}, \theta) \tag{1}$$

where,

- θ is the parameters of the model to be optimized.
- $y^{\mathcal{W}}$ is the set of activities of each time window in $\mathcal{W}$.
- $y^{*\mathcal{W}}$ is the predicted activity returned by the trained model.
- $\mathcal{L}(\cdot)$ is the loss function used in the learning process.

Human activity recognition is a complex task and needs the investigation of advanced deep learning architectures for several reasons:

1. **Variability**: Human behavior is highly variable and can differ from person to person and even from instance to instance for the same person. For example, walking can vary based on the individual's stride length, walking speed, and walking style.
2. **Sensor precision**: Sensor data can be noisy and contain artifacts that make activity recognition challenging. Different sensors can also vary in their accuracy and precision, which can affect the quality of the data and the accuracy of the recognition.
3. **Ambiguity**: Certain activities can be ambiguous and difficult to distinguish from one to another. For example, walking and jogging can have similar sensor data patterns, making it challenging to differentiate between them.

3. Related work

STVL-HM is a hybrid model which considers the benefits of the best models for solving HM tasks. Existing works can be roughly grouped into two families: human activity recognition and spatio-temporal learning. In the following, we will give insights into using STVL-HM compared to studies belonging to both families.

3.1. Human activity recognition

A novel differential spatio-temporal LSTM (DST-LSTM) technique is used to propose a new HAR system based on a pedal wearable device [32]. The pedal wearable device is integrated in the tongue area of the overall footwear to collect dorsum pedis pedal musculoskeletal response data. The DST-LSTM is designed to classify five common activity states based on the collected data: standing, sitting, floor walking, down the stairs, and up the stairs. The dynamic discrepancy knowledge of the collected information is used to construct a new Long short-term memory unit. Multi-head graph attention networks are also used to extract the spatial correlation features of the collected data. Li et al. [33] developed a methodology for recognizing a single user's daily behavior that can adaptively constrain detector vibration throughout human activities in multitenant smart home situations. For HAR, they suggested a data stream analysis method associated with the input frequency-inverse structure. This method is used to assess the contribution of a specific type of sensor to a specific type of activity recognition. They then constructed a spatial matrix based on the arrangement of sensing devices for context awareness and data noise reduction. Finally, they proposed a HAR algorithm for daily behavior recognition that is based on a longer time convolutional neural network and multi-environment sensor data. Islam et al. [34] debuted a new streaming video blower that concentrates on acquiring multiscale and multidimensional frames for quick action detection. The method's ultimate goal is to create a meaningful action recognition module by recognizing five multiple kinds of action capture operators and thoroughly evaluating their impacts on movement prediction over short and long time periods. It gathered motion-type information across the entire clip using a multi-differentiation modeling approach. Wu et al. [35] designed an efficient method for capturing relationships within skeleton action scenes by decomposing the spatio-temporal graph model efficiently. In particular, for spatial simulation, they presented a movement relational classifier that extends the channel dimension to improve modeling of motion local patches as a supplement to traditional physical adjacency relationships. To better fit the data characteristics, an enhanced user defined tangible relationship model is also proposed. They developed an effective multi-focus temporal information seeking strategy for temporal modeling that aggregates data from various temporal stretches and adjacent regions. Ahn et al. [36] proposed a deep learning transformer capable of representing two pass functionalities as a distinguishable vector. First, frames are output as global grid tokens and skeletons are output as joint location tokens from the input video and skeleton sequences, respectively. These tokens are then combined to form multi-class tokens, which are then fed into the designed transformer. The encoder includes a full spatio-temporal attention module as well as a suggested zigzag spatio-temporal attention module. Yang et al. [37] created a hybrid network by combining graph convolution neural network and convolution neural network. It not only makes good use of structural information, but it also accurately models complex relationships between interframe joints. It investigated the relevant impact of convolution neural network and graph convolution neural network, and proposed a new gluing unit to assist the elegant integration of both convolution neural network and graph convolution neural network feature extraction modules while exploiting contextual information.

3.2. Spatio-temporal learning

The Spatio-Temporal Graph Convolutional Neural Network was proposed [38], which models interactions as a graph and eradicates the need for aggregation methods. A linear function was suggested to incorporate social network interactions within the data structure. The paths of nodes in the graph are simulated as a temporal graph to substitute the aggregation layers. A weighted adjacency matrix is generated, in which the kernel function quantifies the influence of nodes. Ali et al. [39] presented a graph deep learning model for extracting influential spatio-temporal features from path data collected to nearby correlated data. The data are represented by an undirected network, with each node depicting a wind station. Each node in a long short-term memory network harvests temporal information. A customizable graph convolutional deep learning architecture was inspired by the concentrated first-order estimation of spectral graph convolutions and used the recovered temporal features to predict the time series of the whole network nodes. It also captured both spatial and deep spatial features of the correlated data with similar spatial information. Deng et al. [40] proposed a convolutional adversarial network on a spatio-temporal graph. To anticipate regularity in the input data, a spatio-temporal generator is created, and a spatio-temporal discriminator is generated to check the correctness of the input sequence data. There are strong correlations between neighboring data points in both the spatial and temporal dimensions. As a result, a new module is presented that employs the graph convolutional gated recurrent unit to aid the GAN components composed by the generator and the discriminator in gaining knowledge the spatio-temporal elements of the input data. After adversarial training, the generator and discriminator can be used as detectors independently, with generator modeling regular evolution shapes and the discriminator supplying recognition requirements that differ with spatio-temporal variables. Wang et al. [41] explored the new concept of multivariate correlation-aware multi-scale prediction and suggested an emergent solution which is a spatio-temporal graph convolutional network with feature correlation. Given

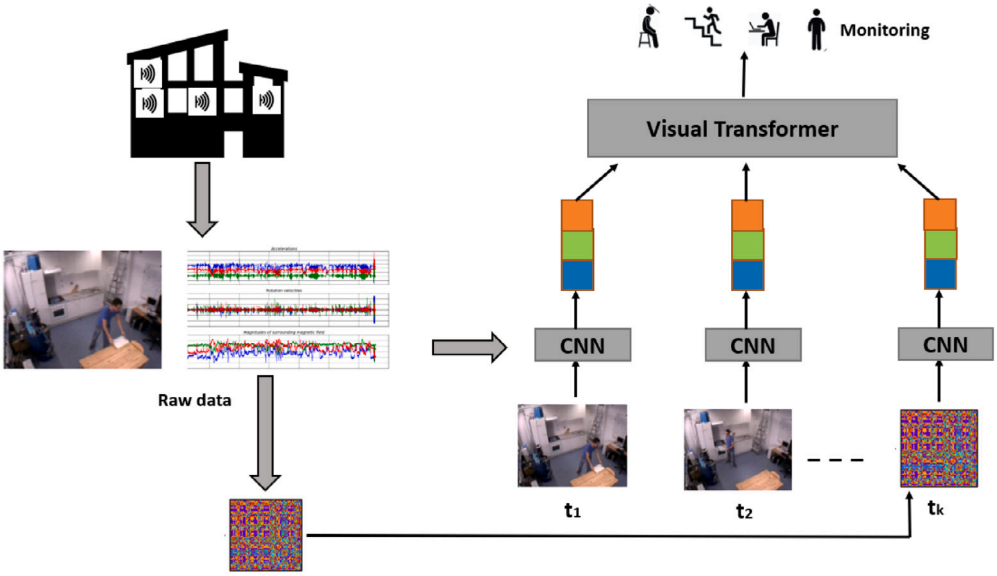

Fig. 1. STVL-HM concept: The sensor data is first converted to a set of graphs of networks. Each graph represents the spatial features of the sensor locations with different timestamps. Hybrid GNN and LSTM model is trained, where GNN is first performed to learn the spatial features from each graph, and LSTM is then executed to learn the temporal features from a sequence of graphs.

a graph, a coarse-grained graph is created based on topology and similarity among the nodes. Temporal learning is then provided for dealing with both fine and coarse-grained network data. A cross-scale graph convolution neural network is presented in the spatial domain to contemporaneously learn and fuse multi-scale spatial variables. For successfully capturing intra- and inter-scale temporal correlations in the temporal domain, a cross-scale temporal network built of structured attention is created.

3.3. Discussion

To our knowledge, existing works for HM suffer from the vanishing gradient problem and do not capture long dependency. We propose a hybrid convolution neural network and transformer model to learn spatio-temporal interconnections between sensor readings, inspired by the achievement of the hybrid combination of convolution neural network and transformers in handling spatio-temporal data. The detailed principles of the designed concept are described in the following section.

4. STVL-HM design: Spatio-temporal visual learning for home-based monitoring

4.1. Principle

The process begins with raw sensor data, which could originate from various sources such as cameras, LiDAR, or other sensors. These raw data streams are then preprocessed and converted into a series of frames. Each frame represents a collection of images captured at different timestamps, effectively encapsulating the temporal aspect of the data. To exploit the visual information within each image, a CNN is applied to learn the spatial features. The CNN's convolutional layers analyze local patterns and structures, enabling it to recognize objects, textures, and visual patterns within the images. This step is crucial for extracting high-level visual features that can help understand the scene and objects captured by the sensors. Once the spatial features have been learned by the CNN, the temporal aspects of the data are addressed using a transformer model. The transformer model is known for its superior ability to capture long-range dependencies and temporal patterns within sequential data. In the context of STVL-HM, the transformer is employed to learn temporal relationships between consecutive frames in the sequence. By doing so, the model gains an understanding of how the visual information evolves over time. Our hybrid CNN-Transformer architecture is a two-step network. Both models are then jointly trained using a large dataset of annotated sensor data. During training, the model optimizes its parameters to effectively fuse spatial and temporal features, allowing it to generate robust representations of the input data. The end result is a sophisticated model that can efficiently capture both static and dynamic aspects of the visual scenes captured by the sensors. Different number of epochs/iterations are used for both CNN and transformer architectures. The CNN is first trained using the CNN epochs, the extracted features will be injected and training using the transformer epochs. Fig. 1 visually illustrates the overall architecture of STVL-HM, showcasing the flow of data from raw sensor inputs through the frame generation, CNN-based spatial feature extraction, and finally, the transformer-based temporal feature learning.

4.2. Data collection

Home-based monitoring involves collecting data from individuals in their own homes. The type of data collected will depend on the specific purpose of the monitoring. We will integrate several strategies for collecting data, including:

1. **Wearable sensors**: We will use fitness trackers or smartwatches to collect data on physical activity, heart rate, sleep, and other vital signs. These sensors can be worn continuously, providing a continuous stream of data.
2. **Remote monitoring devices**: We will use blood pressure monitors, glucose monitors, or pulse oximeters to provide real-time data on vital signs and other health indicators.
3. **Smart home devices**: We will use smart home devices, such as smart scales to collect data on weight, body composition, and other user behavior indicators. These devices can be integrated with other monitoring systems, allowing for a more comprehensive view of an individual's behavior.

At the end of this step, we will have a heterogeneous data composed of time series, and images captured in different timestamps $(t_1, t_2, \dots, t_n)$. At each timestamp, we have the representative data $\hat{d}(t)$

captured by the different sensors. We also used the MinMax normalization method to normalize the captured data. It is crucial step for better learning process. The value 1 is assigned as the highest feature value and the value 0 as the lowest value. The binary equivalents of each value of 0 and 1 are calculated. For each data image $d_i \in \hat{d}(t)$, the normalization is determined as follows:

$$Normalize(d_i) = \{x', \forall x \in d_i, \text{MinMax}(x, d_i)\} \tag{2}$$

where,

$$\text{MinMax}(x, d_i) = \frac{x - \min(d_i)}{\max(d_i) - \min(d_i)} \tag{3}$$

$\min(d_i)$, and $\max(d_i)$ are the minimum and the maximum values of all data values in I_i.

4.3. Spatial visual learning

This step aims to capture the visual features of each data in $d(t)$ using CNN. The process of computing visual features with a CNN can be broken down in the following: Let d_i be the normalized input data, which can be represented as a three-dimensional tensor of size (H, W, C), where H is the height, W is the width, and C is the number of channels. For instance, if the captured data is an image then it will directly injected into the tensor. However, if it is a time series then a conversion is required before injecting it to the tensor. We used the Gramian Angular Fields (GAF) [42] to encode the time series as images. GAF encapsulates the correlation structures and uses the output to generate 2D images. A time series signal is presented in GAF as a polar coordinated system, and the angles of every data point are transformed into matrices. Let K be a set of learnable kernels of size (K_h, K_w, C, O), where K_h and K_w are the kernel height and width, and O is the number of output channels. Each kernel K_j in K is convolved with the input data d_i to produce a feature map F_l, where l ranges from 1 to O. This operation is defined as:

$$F_l = K \bigodot d_i \bigoplus b_l \tag{4}$$

where $\bigodot$ denotes the convolution operation, $\bigoplus$ is a matrix addition, b_l is a learnable bias term for the lth filter, and the output feature map f_l is of size (H', W', 1), where H' and W' are the height and width of the output feature map. The output feature map F_l is then passed through a non-linear activation function g, which introduces non-linearity into the model. We use the commonly known activation function, named ReLU, and which is defined as:

$$g(x) = \max(0, x) \tag{5}$$

where x is the input to the activation function. After each convolutional layer, a pooling layer is often used to downsample the output feature map. We used the most commonly pooling operation which is max-pooling. It aims to extract the maximum value within a pooling window of size (F'_h) from the input feature map. This operation is defined as:

$$F'_i = max_{j=1}^{F'_h} F_{i+(j-1)s} \tag{6}$$

where F'_i is the output of the pooling operation at ith position, s is the stride (i.e., the distance between adjacent pooling windows), and $F_{i+(j-1)s}$ is the input feature map value at position $i + (j-1)s$.

At the of this step, we have the set of data features F', each element $F'_i \in F'$ represent the visual features of the image I_i.

4.4. Temporal learning

To detect temporal dependencies in the series of spatial embeddings produced by the CNN network, we design a transformer-based network. The most popular deep learning algorithms for sequential data are RNNs. Traditional RNNs can only detect long-term dependencies because of gradient vanishing or explosion problems. Because they fixed this issue, variations like LSTM's (Long Short-Term Memory) [43] and GRU's (Gated Recurrent Units) [44] were considered the most popular. Although LSTM and GRU perform equally well on a variety of tasks, GRU's structure becomes less complex and can be trained more quickly, hence it was chosen for this study. The main drawback of LSTM, and GRU is the computationally expensive which are considered more complex than traditional RNNs and require more computational resources to train. Another important factor is the limited context of LSTMs and GRUs where they have a fixed memory size and are only able to capture a limited amount of context from the input sequence. This can lead to difficulties in modeling long-term dependencies in the data. To overcome these issues, we propose in this section a transformer-based network. Transformer is a type of neural network architecture that are able to effectively capture long-range dependencies in sequential data. The key innovation of the Transformer is the self-attention mechanism, which allows the model to attend to different parts of the input sequence when making predictions. In the following, we present our adaptation of the transformer for analyzing the sequence of features F' derived in the previous step:

1. **Patch Embeddings**: In natural language processing and computer vision, one of the key challenges is effectively handling high-dimensional input data. To tackle this issue, a powerful technique known as patch-based embeddings has emerged, which has proven to be highly successful in various deep learning models. The process begins by taking each input feature $F'_i \in F'$ and dividing it into a set of non-overlapping patches. Each patch is essentially a small, localized portion of the input feature that captures specific patterns and details. By breaking down the input into these patches, we gain the ability to extract valuable information from different parts of the data. Following this patch division step, a linear projection is applied to each patch, effectively flattening it into a vector representation. This projection step is crucial as it allows the model to transform the patch data into a format suitable for further processing. The resulting set of patch embeddings, denoted as $\mathcal{P}$, now represents the input data in a more structured and compact manner. These patch embeddings $\mathcal{P}$ are then fed into the transformer encoder as input sequences.
2. **Encoder**: The Transformer encoder consists of a stack of several identical layers, each of which contains multi-head attention and feedforward neural networks. The multi-head attention mechanism allows the model to attend to different parts of the input patches $\mathcal{P}$ and capture long-range dependencies. Assume we have a query matrix Q, a key matrix K, and a value matrix V. The multi-head attention operation can be represented as:

$$MultiHead(Q, K, V) = Concat(head_1, head_2, head_3) \times W^O \tag{7}$$

where $head_i$ is the output of the ith attention head:

$$head_i = Attention(Q \times W_i^Q, K \times W_i^K, V \times W_i^V) \tag{8}$$

and $Attention$ is the scaled dot-product attention function:

$$Attention(Q, K, V) = Softmax\left(\frac{Q \times K^T}{\sqrt{d}}\right) \times V \tag{9}$$

Note that W_i^Q, W_i^K, and W_i^V are learnable parameter matrices for the ith attention head, and W^O is the learnable output projection matrix. The feedforward neural network (FFN) is an essential component of the transformer's encoder layer. It is responsible for introducing non-linearity and transforming the output of the self-attention mechanism within each layer. The FFN is typically applied separately to each token in the input sequence $\mathcal{P}$. Note that in our transformer, we will not use the decoder since the output in home-based monitoring will be a fixed-length vector.

3. **Positional Encoding**: By incorporating positional information into the input patches, our model can effectively understand the spatial relationships between different image features. To achieve this, we introduce positional encodings, which serve as an additional set of learned embeddings representing the spatial location of each patch within the image. The process of adding positional encodings begins with generating unique positional vectors for each patch in the image. These positional vectors are designed in a way that captures the relative positions and spatial distances between patches. The positional encodings are then added to the original patch embeddings before being fed into the transformer encoder. To elaborate on how this works, consider an image divided into an $N \times N$ grid of patches. Each patch is represented by an embedding vector that encodes its visual features. Additionally, for each patch, we generate a positional encoding vector that encodes its spatial location. These positional encodings are designed to complement the visual features and are learned along with the rest of the model during the training process.
4. **Output Head**: The encoder, which is responsible for processing the input sequence and creating contextual representations, generates a set of hidden states representing the input tokens. These hidden states need to be transformed into meaningful predictions or probabilities for the home-based monitoring tasks. The output head typically consists of a fully connected layer, also known as a dense layer, which connects every neuron in the previous layer (the final hidden states) to every neuron in the output layer. This fully connected layer is followed by the softmax function. The softmax function takes the raw output scores and converts them into a probability distribution, making it easier to interpret the model's predictions as probabilities.

4.5. Loss function

To train our hybrid CNN-Transformer model, we combine the cross-entropy loss and the MSE loss into a hybrid loss function $hybridloss(y, \hat{y})$. We introduce a hyperparameter λ that controls the trade-off between the two loss components. The hybrid loss is defined as:

$$hybridloss(y, \hat{y}) = (1 - \lambda) \cdot crossentropy(y, \hat{y}) + \lambda \cdot mse(y, \hat{y}) \tag{10}$$

Where:

- y represents the ground truth labels.
- $\hat{y}$ represents the predicted values from the model.
- $crossentropy(y, \hat{y})$ is the cross-entropy loss between the ground truth labels y and the predicted values $\hat{y}$. This loss is commonly used for classification tasks.
- $mse(y, \hat{y})$ is the mean squared error loss between the ground truth labels y and the predicted values $\hat{y}$. This loss is commonly used for regression tasks.
- λ is a hyperparameter that controls the trade-off between the two loss components. It determines the relative importance of the cross-entropy loss and the mean squared error loss in the overall hybrid loss function.

By using a hybrid loss function, the model can benefit from both the advantages of cross-entropy loss for classification and the advantages of mean squared error loss for regression. The hyperparameter λ allows the model designer to adjust the balance between the two losses based on the specific requirements of the task at hand. For example, setting $\lambda = 0$ would make the model solely rely on cross-entropy loss (ideal for pure classification tasks), while $\lambda = 1$ would make it rely solely on mean squared error loss (ideal for pure regression tasks). Values of λ between 0 and 1 create a trade-off between the two loss components, making it suitable for tasks that have characteristics of both classification and regression.

Table 1
Hyperparameters for STVL-HM.

Hyperparameter	Value
Batch size	32
Learning rate	0.0005
Dropout rate	0.15
Number of convolutional layers	5
Number of transformer layers	3
Kernel sizes	[3, 5, 7]
Weight regularization	L2 (0.0001)

5. Performance evaluation

A robust set of experiments has been carried out to evaluate the performance of STVL-HM solution. A case study of human activity recognition has been analyzed in this experiment.

5.1. Experimental setting

We used the Kinetics human action video dataset, a large-scale video dataset designed for human activity recognition research. It contains more than 650,000 video clips of human actions, each lasting around 10 s, covering 400 action classes. The actions are diverse and include various human activities such as sports, cooking, and playing musical instruments, among others. The videos were collected from YouTube and filtered to ensure that they only contained human actions. The dataset was first released in 2017 and has since been updated with new releases. We used the latest version, Kinetics-700 [45], which was released in 2019 and includes 700 action classes. It has become a standard benchmark for evaluating human activity recognition algorithms. The videos in the Kinetics dataset are labeled with their action class, and each action class has at least 400 video clips. The dataset also includes a train–validation–test split, with around 240,000, 20,000, and 40,000 video clips, respectively. The Kinetics dataset has been widely used in the computer vision community to develop and evaluate human activity recognition models. It has been used as a benchmark dataset for several large-scale action recognition challenges.

To evaluate the performance of STVL-HM, several factors have been take into account, including:

1. **Accuracy**: This is a measure of how well the solution correctly identifies the human activity being performed. This measure is determined by model accuracy. It is often the most important metric for evaluating the performance of a HAR model. Higher accuracy means that the model is able to correctly identify the activity being performed more often, and lower accuracy indicates that the model is making more incorrect predictions.

2. **Recognition rate**: This refers to the ability of the solution to perform well on new, unseen data. It refers to the proportion of instances in the dataset that are correctly classified by a human activity recognition model. It is often used as a measure of the model's performance and is typically expressed as a percentage. For example, if a model correctly identifies 100 activities out of a total of 150 instances in the test set, the recognition rate would be 67%. The higher the recognition rate, the better the performance of the model.

In addition, we used the greedy optimization for hyperparameter tuning [46] which involves iteratively selecting and updating hyperparameters to find the best-performing model on a validation dataset. This approach can be computationally expensive but provides a simple and effective way to optimize model performance. Table 1 shows the optimal hyperparameters of the STVL-HM used in the experiments.

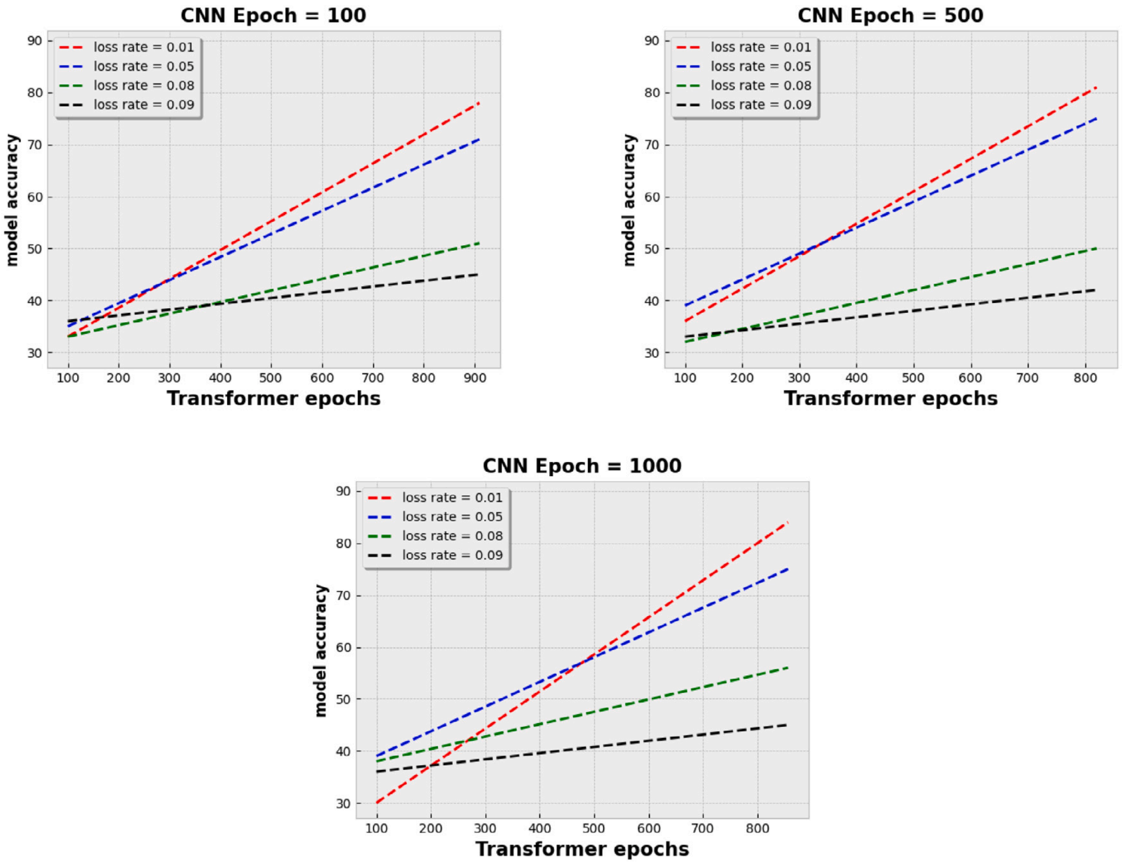

Fig. 2. Accuracy performance of STVL-HM for different CNN epochs (from 100 to 1000), different transformer epochs (from 100 to 1000), and for different loss rate {0.01, 0.05, 0.08, 0.09}.

5.2. Baselines

We used these two recent baseline solutions for experimental comparisons:

1. **DST-LSTM** [32]: It is intended to classify five common activity states: standing, sitting, floor walking, down the stairs, and up the stairs. The collected information's dynamic discrepancy knowledge is used to build a new Long short-term memory unit. The spatial correlation features of the collected data are also extracted using multi-head graph attention networks.

2. **Hybridnet** [37]: It not only makes good use of structural information, but it also accurately models complex interframe joint relationships. It investigated the relevant impact of convolution neural network and graph convolution neural network feature extraction modules while exploiting contextual information, and proposed a new gluing unit to assist the elegant integration of both convolution neural network and graph convolution neural network feature extraction modules.

5.3. Accuracy performance

Several experiments were carried out to evaluate the accuracy performance of the STVL-HM method. We varied both the number of epochs in transformer, and CNN from 100 to 1000, and we also varied the loss rate from 0.01 to 0.09. Fig. 2 reported the obtained results. We observed that the model accuracy is increased while enhancing the number of epochs of transformer, and CNN. For instance, when the number of epochs in transformer is set to 100, the number of epochs in CNN is set to 100, and the loss rate is set 0.01, the model accuracy is less than 33%, however when the number of epochs in the transformer is set to 1000, the number of epochs in CNN is set to 1000 and the loss rate is set to 0.01, the model accuracy is greater than 84%. Furthermore, the results indicates when increasing the loss rate, the model accuracy decreased, where the model accuracy is high for loss rate equal to 0.01, and 0.05, and low for 0.08 and 0.09. From these results, we will set the following parameters in the remaining of the experiments: Transformer's epoch is set to 1000, CNN's epoch is set to 1000, and the loss rate is set to 0.01.

5.4. Recognition rate performance

Several experiments has been performed to evaluate the recognition rate performance. Two different tests have been carried out. We compare the STVL-HM with baseline human activity recognition systems including DST-LSTM, and Hybridnet. We varied the percentage of selected input features from 10% to 100%, and the percentage of training data from 20% to 100%. Fig. 3 reported the obtained results. From these results, we can observe a clear superiority of the STVL-HM compared to the baseline methods (DST-LSTM, and Hybridnet), whatever the scenario used in the experiments. These results confirm the applicability of the proposed methods in recognizing human activities from unseen observations. These results are achieved thanks to the efficient combination of both transformers, and CNN in learning the spatio-temporal visual information from different data sensors highly correlated. STVL-HM can be used as alternative data fusion solutions, where a sequence of images is generated and trained from multi-data sources.

6. Conclusion

This paper introduces a novel model for Home-based Monitoring (HM) that empowers examination, monitoring, and comprehension of

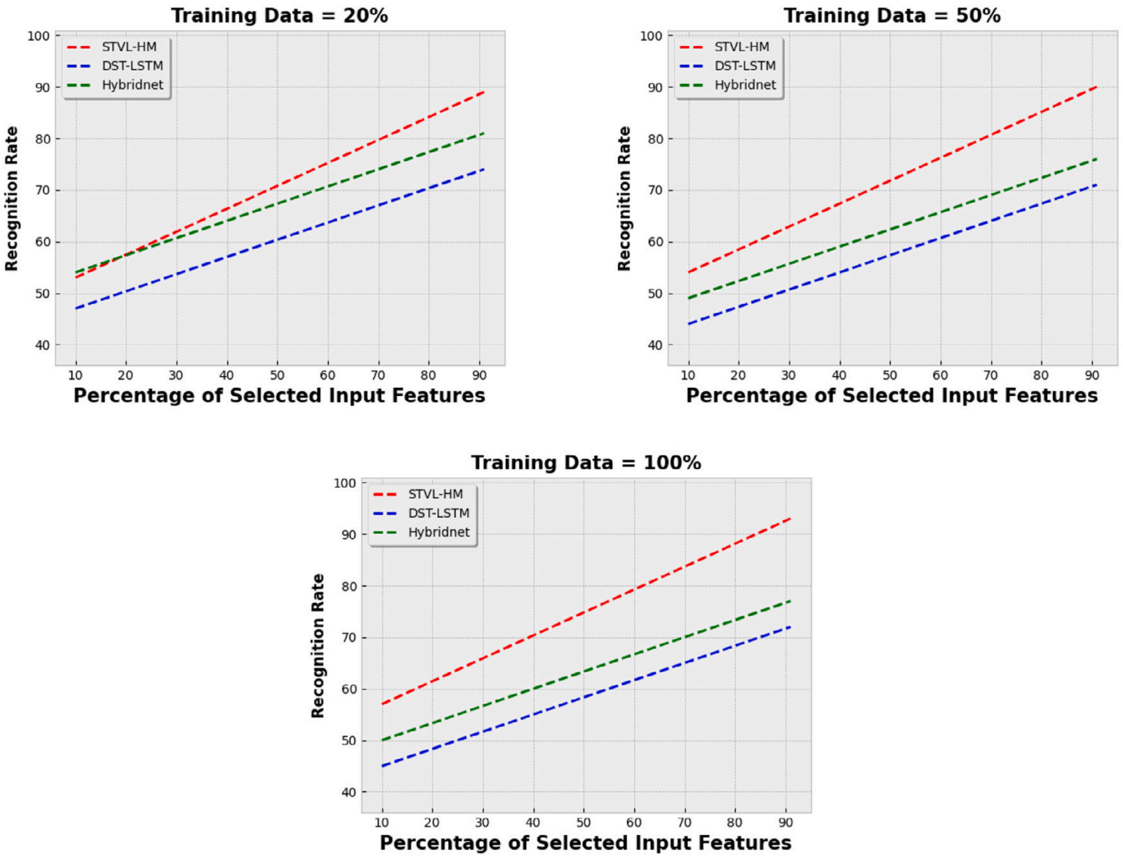

Fig. 3. Recognition rate performance of STVL-HM compared to baseline methods (DST-LSTM, and Hybridnet) for different data sizes.

home-based activities. Spatio-Temporal Visual Learning for HM (STVL-HM) is a new technique for robustifying the HM process by learning from sensor data that is collectively represented in space and time. We propose a CNN and Transformers hybrid model. The CNN begins by learning visual spatial features from sensor data. The visual features that have been learned are then injected into the transformer, which encapsulates temporal data by observing the sensor status at different timestamps. Kinetics-700 (the real use case of human activity recognition scenario for HM data) was used to test STVL-HM. The results show that the STVL-HM is clearly superior to the most recent baseline HM solutions. Future research should focus on improving spatio-temporal visual learning by integrating different optimization such as hyperparameter optimization [47–49]. Also, we would like to explore other applications for hybrid combination of CNN and transformers such as related to remote sensing [50–52], and traffic prediction [53–55] in order to capture both the temporal and the spatial dependencies from such complex data. Testing STVL-HM with other use cases including home elderly care [56], home-gait monitoring [57], and human behavior analysis [58] is also in our future agenda.

Declaration of competing interest

We declare that there is no conflicts of interests.

Data availability

Data will be made available on request.

Acknowledgments

This work is supported by the project Next Generation 3D Machine Vision with Embedded Visual Computing and co-funded under the grant number 325748 of the Research Council of Norway.

Declarations

Ethical and informed consent for data used: No ethical issue for data used.

References

[1] B. Dittakavi, D. Bavikadi, S.V. Desai, S. Chakraborty, N. Reddy, V.N. Balasubramanian, B. Callepalli, A. Sharma, Pose tutor: An explainable system for pose correction in the wild, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 3540–3549.

[2] A. Piau, K. Wild, N. Mattek, J. Kaye, Current state of digital biomarker technologies for real-life, home-based monitoring of cognitive function for mild cognitive impairment to mild Alzheimer disease and implications for clinical care: Systematic review, J. Med. Internet Res. 21 (8) (2019) e12785.

[3] P. Tiwari, A. Lakhan, R.H. Jhaveri, T.-M. Gronli, Consumer-centric internet of medical things for cyborg applications based on federated reinforcement learning, IEEE Trans. Consum. Electron. (2023).

[4] C. Chakraborty, A. Kishor, Real-time cloud-based patient-centric monitoring using computational health systems, IEEE Trans. Comput. Soc. Syst. 9 (6) (2022) 1613–1623.

[5] E. Rajkumar, A. Gopi, A. Joshi, A.E. Thomas, N. Arunima, G.S. Ramya, P. Kulkarni, P. Rahul, A.J. George, J. Romate, et al., Applications, benefits and challenges of telehealth in India during COVID-19 pandemic and beyond: A systematic review, BMC Health Serv. Res. 23 (1) (2023) 1–15.

[6] L. Walton, K. Courtright, G. Demiris, E.F. Gorman, A. Jackson, J.G. Carpenter, Telehealth palliative care in nursing homes: A scoping review, J. Am. Med. Dir. Assoc. 24 (3) (2023) 356–367.

[7] R. Chengoden, N. Victor, T. Huynh-The, G. Yenduri, R.H. Jhaveri, M. Alazab, S. Bhattacharya, P. Hegde, P.K.R. Maddikunta, T.R. Gadekallu, Metaverse for healthcare: A survey on potential applications, challenges and future directions, IEEE Access 11 (2023) 12765–12795.

[8] A. Bannis, S. Pan, C. Ruiz, J. Shen, H.Y. Noh, P. Zhang, IDIoT: Multimodal framework for ubiquitous identification and assignment of human-carried wearable devices, ACM Trans. Internet of Things (2023).

[9] A. Omidvar, J. Kim, A novel theoretical model for predicting the individuals' thermal sensations base d on air temperature and biomarkers measured by wearable devices, Build. Environ. 232 (2023) 110050.

[10] P.N. Srinivasu, G. JayaLakshmi, R.H. Jhaveri, S.P. Praveen, Ambient assistive living for monitoring the physical activity of diabetic adults through body area networks, Mob. Inf. Syst. 2022 (2022) 1–18.

[11] A. Sujith, G.S. Sajja, V. Mahalakshmi, S. Nuhmani, B. Prasanalakshmi, Systematic review of smart health monitoring using deep learning and artificial intelligence, Neurosci. Inform. 2 (3) (2022) 100028.

[12] T.D. Pereira, N. Tabris, A. Matsliah, D.M. Turner, J. Li, S. Ravindranath, E.S. Papadoyannis, E. Normand, D.S. Deutsch, Z.Y. Wang, et al., SLEAP: A deep learning system for multi-animal pose tracking, Nature Methods 19 (4) (2022) 486–495.

[13] H.D.M. Ribeiro, A. Arnold, J.P. Howard, M.J. Shun-Shin, Y. Zhang, D.P. Francis, P.B. Lim, Z. Whinnett, M. Zolgharni, ECG-based real-time arrhythmia monitoring using quantized deep neural networks: A feasibility study, Comput. Biol. Med. 143 (2022) 105249.

[14] L. Wang, Y. Zhou, R. Li, L. Ding, A fusion of a deep neural network and a hidden Markov model to recognize the multiclass abnormal behavior of elderly people, Knowl.-Based Syst. 252 (2022) 109351.

[15] V. Shenbagalakshmi, T. Jaya, Application of machine learning and IoT to enable child safety at home environment, J. Supercomput. 78 (8) (2022) 10357–10384.

[16] Y. Celik, S. Stuart, W.L. Woo, E. Sejdic, A. Godfrey, Multi-modal gait: A wearable, algorithm and data fusion approach for clinical and free-living assessment, Inf. Fusion 78 (2022) 57–70.

[17] S. Kandanaarachchi, H. Ochiai, A. Rao, Honeyboost: Boosting honeypot performance with data fusion and anomaly detection, Expert Syst. Appl. 201 (2022) 117073.

[18] D.K. Sah, K. Cengiz, Y. Alshehri, N. Alnazzawi, N. Ivković, et al., Early alert for sleep deprivation using mobile sensor data fusion, Comput. Electr. Eng. 102 (2022) 108228.

[19] Z. Ma, Y. Geng, S. Nie, H. Ji, X. Yan, H. Liao, SNIF-DFA: A signal processing and information fusion method for smart Gua Sha device, IEEE Sens. J. 22 (24) (2022) 24176–24185.

[20] A. Krishnan, S. Das, M. Bhattacharjee, Flexible piezoresistive pressure and temperature sensor module for continuous monitoring of cardiac health, IEEE J. Flex. Electron. (2023).

[21] F. Herold, P. Theobald, T. Gronwald, N. Kaushal, L. Zou, E.D. de Bruin, L. Bherer, N.G. Müller, Alexa, let's train now!—A systematic review and classification approach to digital and home-based physical training interventions aiming to support healthy cognitive aging, J. Sport Health Sci. (2023).

[22] H. Sadreazami, M. Bolic, S. Rajan, Contactless fall detection using time-frequency analysis and convolutional neural networks, IEEE Trans. Ind. Inform. 17 (10) (2021) 6842–6851.

[23] R.S.N. Noella, J. Priyadarshini, Diagnosis of Alzheimer's, Parkinson's disease and frontotemporal dementia using a generative adversarial deep convolutional neural network, Neural Comput. Appl. (2022) 1–10.

[24] Q. Liu, X. Meng, F. Shao, S. Li, Supervised-unsupervised combined deep convolutional neural networks for high-fidelity pansharpening, Inf. Fusion 89 (2023) 292–304.

[25] M. Panagiotou, A. Zlatintsi, P. Filntisis, A. Roumeliotis, N. Efthymiou, P. Maragos, A comparative study of autoencoder architectures for mental health analysis using wearable sensors data, in: 30th European Signal Processing Conference, EUSIPCO, IEEE, 2022, pp. 1258–1262.

[26] A. Habib, C. Karmakar, J. Yearwood, Domain agnostic post-processing for QRS detection using recurrent neural network, IEEE J. Biomed. Health Inf. (2023).

[27] D. Fan, X. Yang, N. Zhao, L. Guan, Q.H. Abbasi, Exercise monitoring and assessment system for home-based respiratory rehabilitation, IEEE Sens. J. 22 (19) (2022) 18890–18902.

[28] E. Lan, Performer: A novel PPG-to-ECG reconstruction transformer for a digital biomarker of cardiovascular disease detection, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2023, pp. 1991–1999.

[29] M. Gehrig, D. Scaramuzza, Recurrent vision transformers for object detection with event cameras, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 13884–13893.

[30] A. Metin, A. Kasif, C. Catal, Temporal fusion transformer-based prediction in aquaponics, J. Supercomput. (2023) 1–25.

[31] A. Mehrish, N. Majumder, R. Bharadwaj, R. Mihalcea, S. Poria, A review of deep learning techniques for speech processing, Inf. Fusion (2023) 101869.

[32] H. Wu, Z. Zhang, X. Li, K. Shang, Y. Han, Z. Geng, T. Pan, A novel pedal musculoskeletal response based on differential spatio-temporal LSTM for human activity recognition, Knowl.-Based Syst. 261 (2023) 110187.

[33] Y. Li, G. Yang, Z. Su, S. Li, Y. Wang, Human activity recognition based on multienvironment sensor data, Inf. Fusion 91 (2023) 47–63.

[34] M.S. Islam, K. Bakhat, M. Iqbal, R. Khan, Z. Ye, M.M. Islam, Representation for action recognition with motion vector termed as: SDQIO, Expert Syst. Appl. 212 (2023) 118406.

[35] C. Wu, X.-J. Wu, T. Xu, Z. Shen, J. Kittler, Motion complement and temporal multifocusing for skeleton-based action recognition, IEEE Trans. Circuits Syst. Video Technol. (2023).

[36] D. Ahn, S. Kim, H. Hong, B.C. Ko, STAR-transformer: A spatio-temporal cross attention transformer for human action recognition, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2023, pp. 3330–3339.

[37] W. Yang, J. Zhang, J. Cai, Z. Xu, HybridNet: Integrating GCN and CNN for skeleton-based action recognition, Appl. Intell. 53 (1) (2023) 574–585.

[38] R.W. Liu, M. Liang, J. Nie, Y. Yuan, Z. Xiong, H. Yu, N. Guizani, STMGCN: Mobile edge computing-empowered vessel trajectory prediction using spatio-temporal multi-graph convolutional network, IEEE Trans. Ind. Inform. (2022).

[39] A. Ali, Y. Zhu, M. Zakarya, Exploiting dynamic spatio-temporal graph convolutional neural networks for citywide traffic flows prediction, Neural Netw. 145 (2022) 233–247.

[40] L. Deng, D. Lian, Z. Huang, E. Chen, Graph convolutional adversarial networks for spatiotemporal anomaly detection, IEEE Trans. Neural Netw. Learn. Syst. 33 (6) (2022) 2416–2428.

[41] S. Wang, M. Zhang, H. Miao, Z. Peng, P.S. Yu, Multivariate correlation-aware spatio-temporal graph convolutional networks for multi-scale traffic prediction, ACM Trans. Intell. Syst. Technol. 13 (3) (2022) 1–22.

[42] Z. Wang, T. Oates, et al., Encoding time series as images for visual inspection and classification using tiled convolutional neural networks, in: Workshops At the Twenty-Ninth AAAI Conference on Artificial Intelligence, vol. 1, AAAI Menlo Park, CA, USA, 2015, pp. 1–7.

[43] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780.

[44] K. Cho, B. Van Merriënboer, D. Bahdanau, Y. Bengio, On the properties of neural machine translation: Encoder-decoder approaches, 2014, arXiv preprint arXiv:1409.1259.

[45] L. Smaira, J. Carreira, E. Noland, E. Clancy, A. Wu, A. Zisserman, A short note on the kinetics-700-2020 human action dataset, 2020, arXiv preprint arXiv:2010.10864.

[46] E.K. Sahin, S. Demir, Greedy-AutoML: A novel greedy-based stacking ensemble learning framework for assessing soil liquefaction potential, Eng. Appl. Artif. Intell. 119 (2023) 105732.

[47] Y. Djenouri, G. Srivastava, J.C.-W. Lin, Fast and accurate convolution neural network for detecting manufacturing data, IEEE Trans. Ind. Inform. 17 (4) (2020) 2947–2955.

[48] Y. Djenouri, M. Comuzzi, Combining apriori heuristic and bio-inspired algorithms for solving the frequent itemsets mining problem, Inform. Sci. 420 (2017) 1–15.

[49] T. Mezair, Y. Djenouri, A. Belhadi, G. Srivastava, J.C.-W. Lin, A sustainable deep learning framework for fault detection in 6G industry 4.0 heterogeneous data environments, Comput. Commun. 187 (2022) 164–171.

[50] X. Meng, N. Wang, F. Shao, S. Li, Vision transformer for pansharpening, IEEE Trans. Geosci. Remote Sens. 60 (2022) 1–11.

[51] N. Wang, X. Meng, X. Meng, F. Shao, Convolution-embedded vision transformer with elastic positional encoding for pansharpening, IEEE Trans. Geosci. Remote Sens. 60 (2022) 1–9.

[52] H. Shen, X. Meng, L. Zhang, An integrated framework for the spatio–temporal–spectral fusion of remote sensing images, IEEE Trans. Geosci. Remote Sens. 54 (12) (2016) 7135–7148.

[53] Y. Djenouri, A. Belhadi, J.C.-W. Lin, A. Cano, Adapted k-nearest neighbors for detecting anomalies on spatio–temporal traffic flow, IEEE Access 7 (2019) 10015–10027.

[54] T. Mezair, Y. Djenouri, A. Belhadi, G. Srivastava, J.C.-W. Lin, Towards an advanced deep learning for the internet of behaviors: Application to connected vehicles, ACM Trans. Sensor Netw. 19 (2) (2022) 1–18.

[55] Y. Djenouri, A. Belhadi, E.H. Houssein, G. Srivastava, J.C.-W. Lin, Intelligent graph convolutional neural network for road crack detection, IEEE Trans. Intell. Transp. Syst. (2022).

[56] C. Odabasi, F. Graf, J. Lindermayr, M. Patel, S.D. Baumgarten, B. Graf, Refilling water bottles in elderly care homes with the help of a safe service robot, in: 2022 17th ACM/IEEE International Conference on Human-Robot Interaction, HRI, IEEE, 2022, pp. 101–110.

[57] H. Abedi, A. Ansariyan, P.P. Morita, A. Wong, J. Boger, G. Shaker, AI-powered non-contact in-home gait monitoring and activity recognition system based on mm-Wave FMCW radar and cloud computing, IEEE Internet Things J. (2023).

[58] A. Belhadi, Y. Djenouri, G. Srivastava, D. Djenouri, J.C.-W. Lin, G. Fortino, Deep learning for pedestrian collective behavior analysis in smart cities: A model of group trajectory outlier detection, Inf. Fusion 65 (2021) 13–20.