Improving the activity recognition using GMAF and transfer learning in post-stroke rehabilitation assessment

Issam Boukhennoufa, Xiaojun Zhai*,Klaus D. McDonald-Maier
School of Computer Science and Electronic Engineering
University of Essex
Colchester, United Kingdom
xzhai@essex.ac.uk

Victor Utti, Jo Jackson
School of Sport, Rehabilitation and Exercise Sciences
University of Essex
Colchester, United Kingdom

***Abstract*—An important part of developing a performant assessment algorithm for post-stroke rehabilitation is to achieve a high-precision activity recognition. Convolutional Neural Networks (CNN) are known to give very accurate results, however they require the data to be of a specific structure that differs from the sequential time-series format typically collected from wearable sensors. In this paper, we describe models to improve the activity recognition using the CNN classifier. At first by modifying the Gramian angular field algorithm by encoding all the sensors' channels from a single time window into a single 2D image allows to map the maximum activity characteristics. Feeding the resulting images to a simple 1D CNN classifier improves the accuracy of the test data from 94% for the traditional segmentation approach to 97.06%. Subsequently, we convert the 2D images into the RGB format and use a 2D CNN classifier. This results in increasing the test data accuracy to 97.52%. Finally, we employ transfer learning with the popular VGG_16 model to the RGB images, which yields to improving the accuracy further more to reach 98.53%.**

***Index Terms*—Stroke, GMAF, CNN, Activity recognition, Transfer learning.**

I. Introduction

Rehabilitation after Stoke is a tedious and yet necessary stepping-stone that stroke survivors need to undertake towards recovery. Patients perform an important part of their rehabilitation in an outpatient environment [1] where they are required to carry out their exercises - consisting usually of Activities of Daily Life (ADL) [2] - and record them in order to allow the doctor to monitor and assess their progress. To do so, researchers have come up with applications to help monitor and evaluate the rehabilitation process remotely and without the therapist's involvement using wearable sensors [3]. These devices provide a high level of portability and low-price giving researchers and therapists a variety of possibilities and solutions. In order to implement an intelligent assessment system, these apparatus are used in conjunction with processing algorithms in smartphones, edge-devices or even cloud platforms‘ [4] to obtain a preliminary and objective evaluation. An important part of this assessment system is to perform an accurate Human Activity Recognition (HAR) [5]. HAR is a wide-ranging research that deals with classifying individuals' activities using data collected either remotely, such as from radar or video, or directly from the subject's body using wearable sensors such as using Inertial Measurement Units (IMU) or Electromyography (EMG) sensors.

With the objective to have the most accurate HAR, multiple methods have been investigated starting from 1) conventional signal processing modelling approach that seeks a mathematical relationship between an activity and the different modelling parameters, to 2) machine learning algorithms, that extract pertinent features to allow the model to differentiate and recognise the different activities or to more recently 3) deep learning algorithms that are trained to recognise different patterns to distinguish between the activities.

In the literature, the most employed approaches are done using traditional supervised machine learning algorithms [6]–[9] and the most popular classifiers are: support vector machines, decision-trees, K-nearest neighbor and dynamic hidden Markov models. while these models achieve very good results, but a drawback of these approaches are that they entirely rely on the selection of features, meaning that a poor selection of features will yield to a poor performing HAR model, which will yield in its turn to a poor assessment of the exercises. This is not desirable in post-stroke rehabilitation which requires an accurate evaluation of the execution of the exercises.

In the recent years, with the maturity of the deep learning algorithms, tremendous progress has been achieved in other fields of study namely: computer vision, speech recognition and image classification. One of the models that achieved large success working with images are CNN algorithms. Their architectures are analogous to that of the connectivity pattern of neurons in the numan brain and were inspired by the organisation of the Visual Cortex. [10]–[12]. Many outstanding models that use CNN were developed over time, such as VGG [13] Alex-Net [14] and ResNet [15]. These models can be adapted to be used in other applications without the need for fully re-training them on the new database by employing transfer learning. Transfer learning is used to improve a learner from one domain by transferring information from a related domain [16].

Inspired by these developments, many approaches have been taken in order to adapt time-series data input-structures, to CNN-based algorithms input-requirements in order to ameliorate HAR accuracy. Techniques such as segmentation approaches [17] that take fixed window sizes of data, as well as algorithms to encode the data into images i.e Gramian Angular Field (GMAF) images introduced in [18] are investigated in this work. GMAF has already been used for EEG classification [19] and performed well.

The contribution of this work is that the chunks of the time-series data collected from the different sensors are merged and encoded in an image to allow translate the highest possible number of characteristics in the resulting image. In the first part, two approaches of 2D image-encoding resulting from the GMAF transformation are presented with a comparison with the classical windowing approach when fed to a simple 1D CNN algorithm. This improves the accuracy of the test data from 94% for the traditional segmentation approach to 97.06%. In the second part, the 2D images are converted to the RGB format in order to profit from the pretrained VGG model using transfer learning which yields to improving the accuracy even further to reach 98.53%.

The reminder of the paper is organised as follows: in section II, the dataset utilised is presented with the pre-processing done consisting of the segmentation (subsection II-A) and the encoding of the resulting chunks into images (subsection II-B). After that a description of the classifiers used and the results are presented in section III: The 1D CNN model with 2D images in part III-A, and the 2D CNN model, transfer learning in part III-B. Finally we conclude the paper in the section IV.

II. Dataset and pre-processing

In this work, the smartphone-based recognition of human activities and postural transitions dataset from Reyes-Ortiz et al [20] is used. It contains data from experiments that were carried out with a group of 30 volunteers within an age bracket of 19-48 years who performed a protocol of ADL. In this paper six dynamic activities from the dataset were included: walking, walking up, walking down, sit to stand, stand to sit, laying. The reasons for choosing these activities are that post-stroke patients are required to perform them in their daily lives. In addition, the quantity of data for the different activities are very close allowing to build a more accurate model less prone to bias. Besides some of these activities are very similar and hard to differentiate which will be a good challenge for our algorithms. The data is comprised of tri-axial linear acceleration and 3-axial angular velocity at a constant frequency of 50Hz using the embedded accelerometer and gyroscope in a smartphone. The dataset is organised in two folders the first contains unprocessed raw data and the second contains preprocessed data (denoised and decomposed in different time windows and features). In this work, only the raw data were considered.

A. Data windowing

After the data of the different activities were loaded into different frames of data, each element at a particular time was labeled depending on which activity was performed. After that, a sliding window method has been employed in order to prepare the data for further processing. A sliding window converts sequential data into different chunks of data with a fixed size in order to be used in algorithms that require the data to be of a specific structure. In this work, a sliding window of 4 sec (4 sec × 50 Hz = 200 data elements) was chosen to decompose the dataset into different windows of the same size. The label for each data chunk was chosen to be the label that is most recurrent within the segment. Since the activities were performed sequentially, an activity might be cut when composing the different windows. To remedy to this issue, an overlap of 2 sec was introduced, which means that adjacent windows share 50% of the data. The resulting windows are matrices with fixed sizes 200 × 6 with the six columns corresponding to the triaxial accelerometer and gysroscope. Fig 1 shows how a sliding window operates.

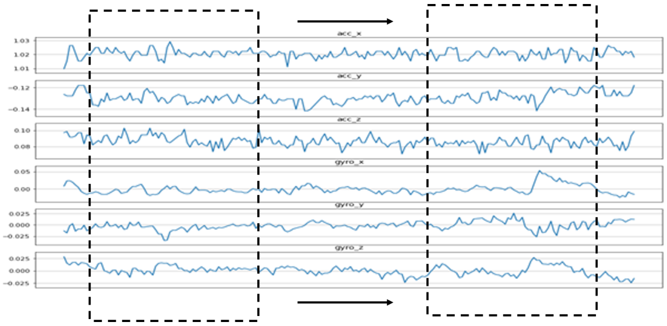


Fig. 1: Sliding window to decompose the dataset.

B. Encoding IMU data into 2D images using Gramian Angular field

A GMAF is a novel technique to encode time series data into images, it employs the polar-coordinates representation of the data written in a matrix form called the Gramian matrix where each element is either the summation (GASF) of the cosines of the angles or difference (GADF). The advantage of using such a mapping is that it maintains the temporal dependency, the reason is time increases as the position shifts from top left to bottom right. The steps to encode the times series data into images using GAF are given bellow:

- First data should be normalised to the range [0,1] using the linear normalisation equation 1:

$$\hat{x_i} = \frac{x_i - min(X)}{max(X) - min(X)} \tag{1}$$

- After that, the resulting time series data is mapped into its polar coordinates representation using equations 2, 3

$$\phi = arccos(\hat{x_i}), -1 \leq \hat{x_i} \leq 1, \hat{x_i} \in X \tag{2}$$

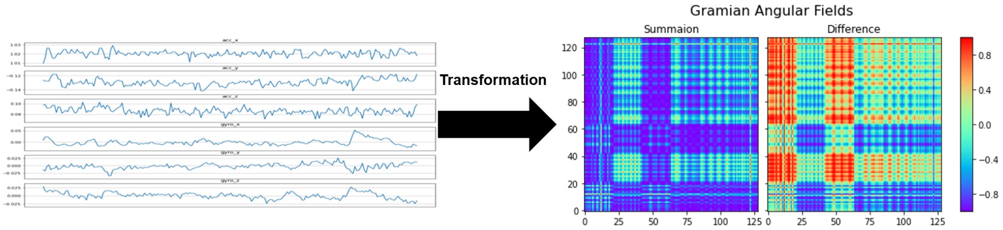


Fig. 2: Encoding a window of the IMU data into gramian images.

$$r = \frac{t_i}{N}, t_i \in \mathbb{N} \tag{3}$$

- Finally we either sum (GASF) or differentiate (GADF) the angles to construct our Gramian matrix as shown in 4, 5 respectively:

$$\begin{aligned} GASF &= cos(\phi_i + \phi_j) \\ &= \hat{X}^T.\hat{X} - \sqrt{I - \hat{X}^2}^T.\sqrt{I - \hat{X}^2} \end{aligned} \tag{4}$$

$$\begin{aligned} GADF &= sin(\phi_i - \phi_j) \\ &= \sqrt{I - \hat{X}^2}^T.\hat{X} - \hat{X}^T.\sqrt{I - \hat{X}^2} \end{aligned} \tag{5}$$

Where I is the unit vector after the transformation to polar coordinates, X the elements of the time series X, and t the time subscript.

The approach taken in this paper is encoding each window of data presented earlier into separate 256 × 256 images. Fig 2 shows an example of a window of data encoded in GASF and GADF.

III. Classification and Experimental Results

In this section we construct the different models for the activity recognition using different classifiers' architectures and different time-series data representations and discuss the results obtained. In subsection III-A we employ the 2D size images from the GMAF transformations (subsection II-B) and the windows from the segmentation (subsection II-A) to feed a 1D based CNN classifier, while in subsection III-B we convert the previous images from (subsection II-B) to RGB format and use a 2D CNN based classifier as well as the VGG_16 pre-trained model employing transfer learning technique and compare the overall results.

A. 2D models

The model used for the classification comprises two 1D CNN layers, supported by a dropout layer for the regularisation of the data, then a pooling layer. The reason for defining two CNN layers is to give the model a good chance of learning the features from the input. In order to avoid over-fitting of the data resulting from the fast learning of the CNNs a dropout layer is utilised. After the CNN, the features are flattened to a 64 nodes vector and goes through a fully connected layer that provides a buffer between the learnt characteristics and the classification. This model uses a standard tuning of 64 parallel feature-maps and a kernel size of 2. The three discussed methods in subsections II-A and II-B were used as inputs to this classifier namely: the windowing method, the GASF and the GADF as shown in Fig 3.

The results of decomposing the dataset are 7474 different windows of data of 200 samples for the six sensors-axes (7474 × 200 × 6). The encoded images resulting from the Gramian transformation are 7474 of 256 × 256 different images. 80% of the data (5980) were used for training the model while 1494 where used for testing. To evaluate the techniques, the model was used in three separate parts, one for each input technique.

The models were trained for 250 epochs on an I7 CPU 6700T 16GB Ram and the results are shown in Fig 4.

- The window-CNN model (Fig 4a) reaches a maximum accuracy of 95.42% for training and 94% for the testing, this model seems less prone to over-fitting as the accuracies seems to stabilise at the same time after 130 epochs at around 94%. This model though starts learning slowly with a training precision of 37.5% and a testing precision of 72.31% at the origin. The average learning time was 740 μs per sample.
- The GASF-CNN model (Fig 4b) reaches a maximum training accuracy of 98.81% and testing accuracy of 97.06% but it seems to start overfitting after 30 epochs. The accuracies seem to stabilise at an accuracy of 98.54% for training and 96.25% for validation. The model also start learning quickly with a training accuracy of 69.46% at the origin and 87.29% for the testing. The average training time was 770 μs by sample.
- Finally, the GADF-CNN model (Fig 4c) reaches a maximum training accuracy of 99.38% and testing accuracy of 97.06%, this model seems to start overfitting after 35 epochs. The accuracies seem to stabilise at 98.43% training and 96.19% for validation were obtained. This model though starts learning very quickly with a training precision of 71.47% and a testing precision of 89.23% at the origin. The average learning time was 820 μs per sample.
- 75 time chunks from the overa11 1494 were miss-classifed in the window-CNN (Fig 4d) model. It confuses 38 walking up activities for walking down and 37 walking down for walking up.
- For the GMAF models, 44 miss-classifications for both models were recorded. The GSAF_CNN (fig 4e) confused 40 walk-ups for walk-downs while the GDAF_CNN (Fig

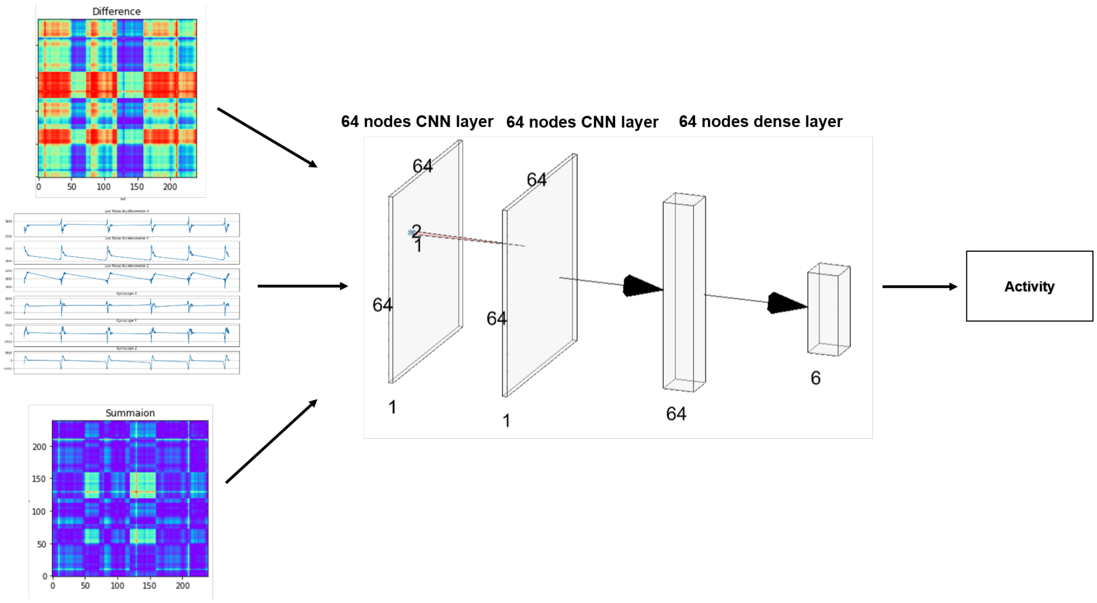
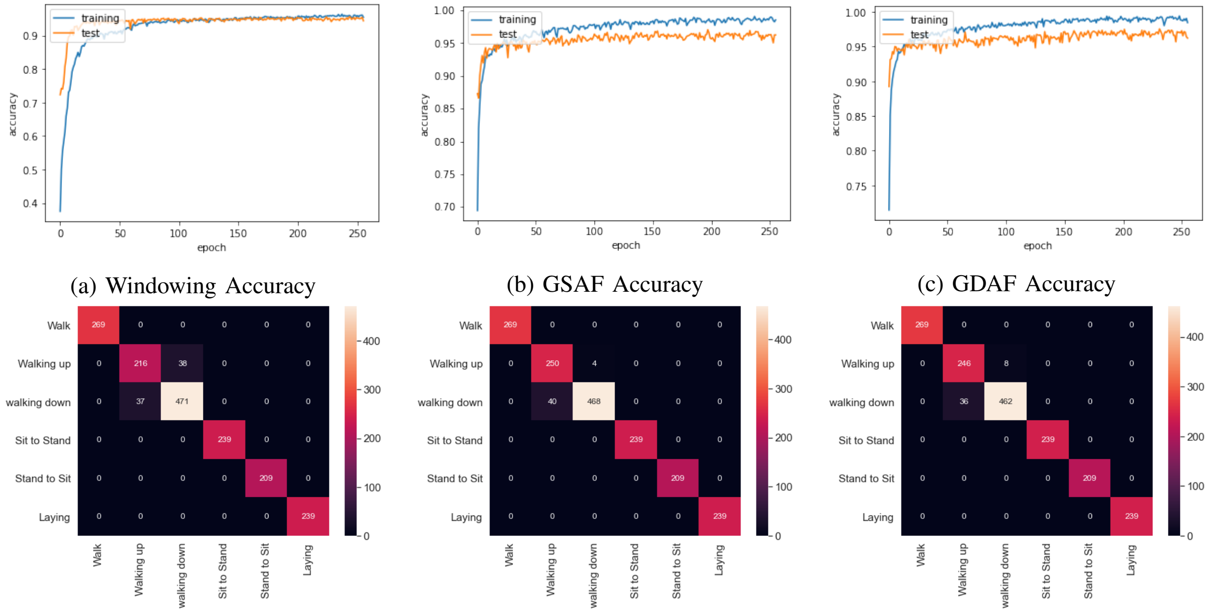


Fig. 3: Classification process for the 2D methods.



(a) Windowing Accuracy

(b) GSAF Accuracy

(c) GDAF Accuracy

(d) Windowing confusion matrix

(e) GSAF confusion matrix

(f) GDAF confusion matrix

Fig. 4: Accuracies and confusion matrices of the different 2D methods

4f) miss-classified 36 walking for walking-down.

B. RGB models

The 2D: 128 × 128 GMAF images were converted to the 128 × 128 × 3 RGB format in order to investigate their performances using:

The first model comprises 2 layers of 2D CNN 64-nodes supported by dropouts to reduce over-fitting. the learned features are flattened and then filtered out through a 64-nodes vector to finally going through the Softmax classification layer. This model uses a standard tuning of 64 parallel feature-maps and a kernel size of 2 × 2.

In the second model transfer learning is used by employing the popular VGG16, which is a 16-layer network built by Oxfords Visual Geometry Group (VGG) [13]. It was pre-trained on 1,000,000 images dataset from ImageNet and achieved state-of-the art results . It contains 16 hidden layers composed of convolutional layers, max pooling. One extra Softmax 6 layers classification layer was added at the top for our classification.

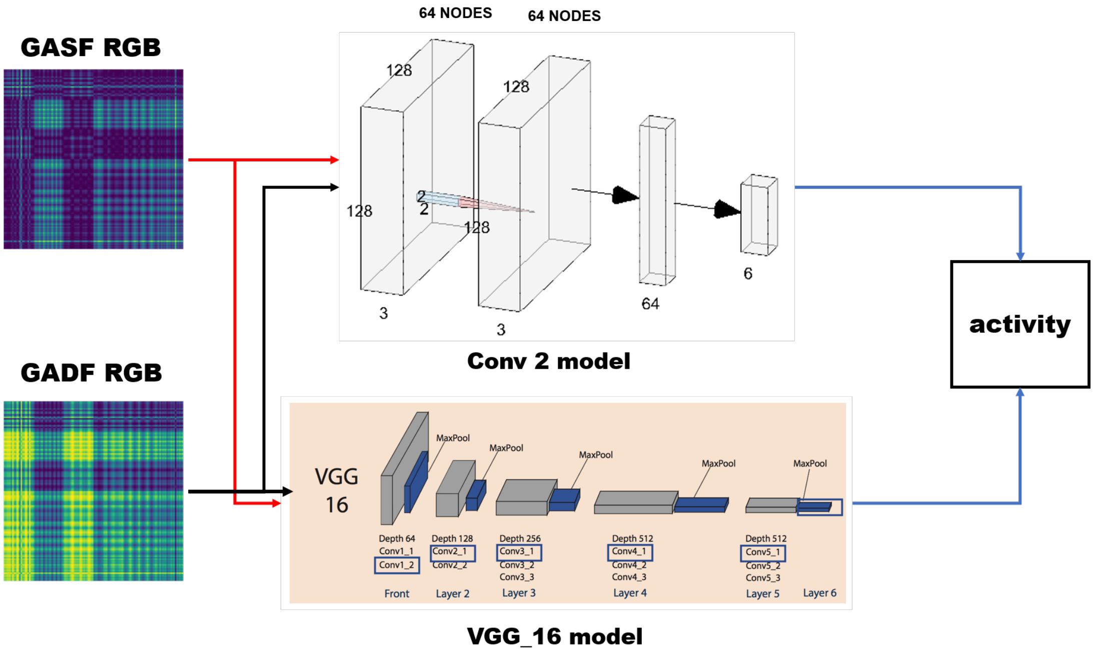
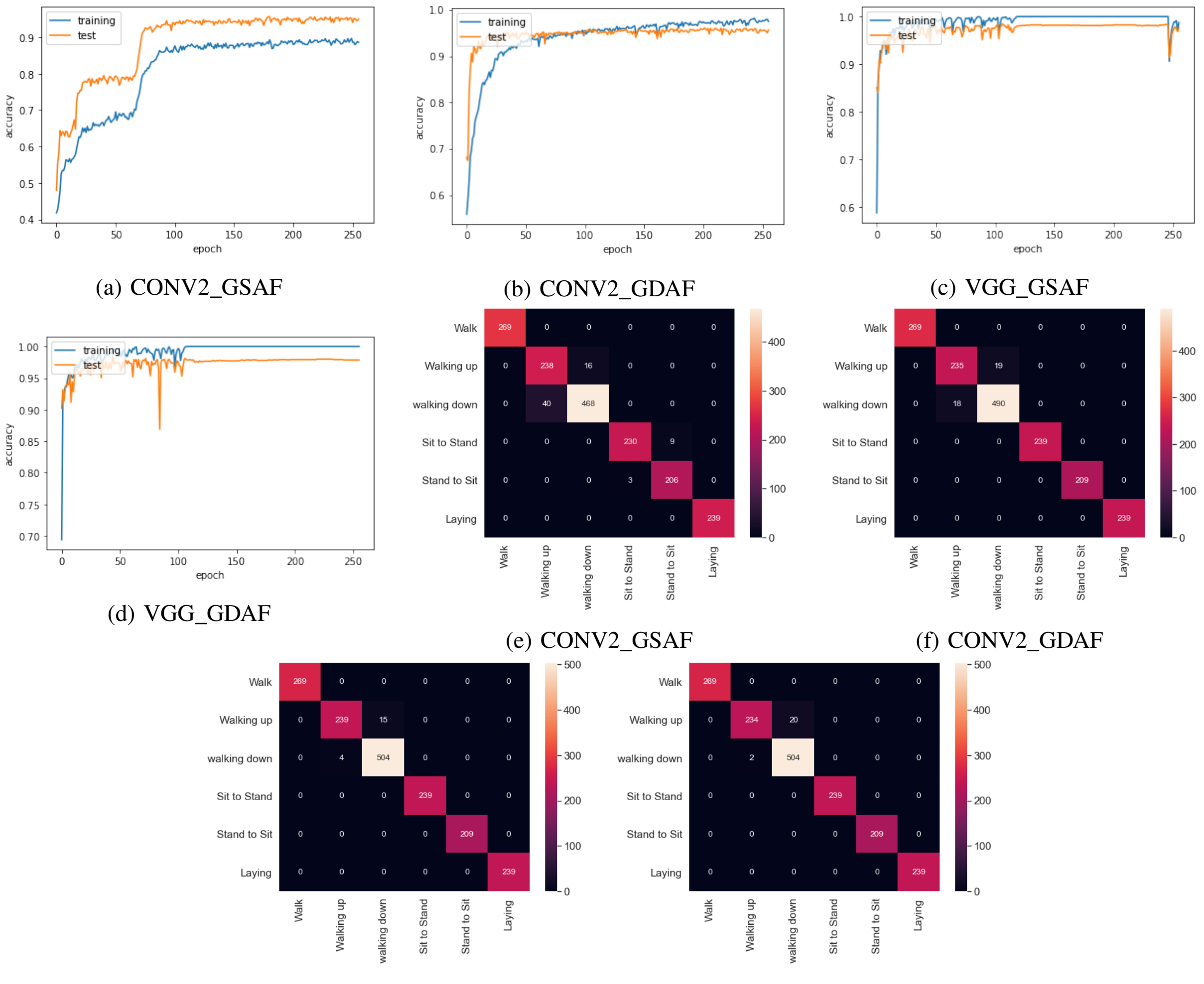


Fig. 5: Classification process for the RGB methods.

As for the 2d models, 80 percent of the data (5980) were used for training the model while the 1494 where used for testing. To evaluate the techniques, the model was used in four separate structures (depending on the two inputs and the two classifiers) as shown in Fig 5. Fig 6 shows the models' accuracies when trained for 250 epochs on Google colab GPU 16GB Ram.

- The CONV2D_GSAF model performs relatively badly (Fig 6a), it reaches a maximum accuracy of 89.53% for training and 95.45% for the testing, this model seems less prone to over-fitting as the accuracies seems to stabilise at the same time after 120 epochs around the accuracies given before. This model though starts learning slowly with a training precision of 41.80% and a testing precision of 47.96% at the origin. The average learning time was 22.33 ms per sample.
- The CONV2D_GDAF model (Fig 6b) reaches a maximum training accuracy of 97.98% and testing accuracy of 97.52% but it seems to start overfitting after 120 epochs. The validation accuracy seem to stabilise at an accuracy of 95.65% while the training keeps increasing above 97.98% . The model also start learning quickly with a training accuracy of 55.88% at the origin and 68.16% for the testing. The average training time was 42 ms by sample.
- The VGG_GSAF model (Fig 6c) reaches a maximum training accuracy of 100% and testing accuracy of 98.46%, this model seems to start overfitting after 115 epochs. The accuracies then decrease to accuracies of 98.73% training and 97.59% for validation. This model though starts learning very quickly with a training precision of 58.86% and a testing precision of 85.08% at the origin. The average learning time was 73 ms per sample.
- Finally, the VGG_GDAF model (Fig 6d) reaches a maximum training accuracy of 100% and testing accuracy of 98.53%, this model seems to stabilise after 120 epochs at 100% training and 97.86% for validation. This model though starts learning very quickly with a training precision of 69.41% and a testing precision of 90.23% at the origin. The average learning time was 79 ms per sample.
- For the 2D CNN models, 68 and 37 miss-classifications were recorded for the 2D_GSAF (Fig 6e) and 2D_GDAF (Fig 6f) models respectively. The first one mostly confuses walking up and down but also some sit to stand and stand to sit activities. the second one is more accurate only miss-classifying some walking up and down activities.
- For the VGG models, 19 and 22 miss-classifications for the VGG_GSAF (6g) and VGG_GDAF (6h) models were recorded respectively. The VGG_GSAF) confused 15 walk-ups for walk-downs while the VGG_GDAF miss-classified 20 walking for walking-down.

To summarise, the four RGB_based models give even better accuracies than the the 2D models. Using GSAF and the CNN_2D improved the accuracy of the windowing method by approximately 1.45% for the validation data, and decreased the training data by 5.89% for training data but took much longer for training. The reason for that is that the windows of data were encoded to 128 × 128 images and then to RGB 128 × 128 × 3 images.
Using GADF and the CNN_2D improved the windowing accuracy 3.56% for the validation data, and 2.52% for training



(a) CONV2_GSAF

(b) CONV2_GDAF

(c) VGG_GSAF

(d) VGG_GDAF

(e) CONV2_GSAF

(f) CONV2_GDAF

(g) VGG_GSAF

(h) VGG_GDAF

Fig. 6: Accuracies and confusion matrices of the different RGB methods

data, nevertheless the required time for training was slower than the GSAF_CNN2D (almost double). On an other hand the VGG models gave the best results overall, it improved the windowing accuracy 4.58% for the training data for both GSAF and GDAF, and 4.46%, 4.53% for the test data accuracy for GSAF and GDAF respectively. The time used for the training was the slowest among all models.

IV. Conclusion

This paper presented three different ways to adapt time series data from IMU sensors to CNN, and benefit from the tremendous accuracies this classifier provided in the other domains, in order to improve the activity recognition process in the assessment of rehabilitation. The contribution of this work consists of the way the data are structured before being fed to the classifier, the six different streams of data coming from the the triaxial gyroscope and the triaxial accelerometer were extracted using a sliding 2D window of a fixed length, these 2D windows were then encoded to different 2D images using GAF transformation which allows to map all the characteristics from the different sensors axes in one image. The accuracy of the test data improved from 94% for the traditional segmentation approach to 97.06%. The 2D images were then converted to RGB in order to profit from some popular pre-trained models using the transfer learning, and it improved the model performance even further to reach 98.53%. As future work, an IoT system based on these models will be implemented to permit real-time monitoring of the data for the post-stroke patients. The models could also be fine-tuned in order to achieve even better precision.

V. Acknowledgements

This work is supported by the UK Engineering and Physical Sciences Research Council through grants EP/R02572X/1, EP/P017487/1 and EP/P015956/1

References

[1] C. for Disease Control, P. (CDC *et al.*, "Outpatient rehabilitation among stroke survivors–21 states and the district of columbia, 2005." *MMWR. Morbidity and mortality weekly report*, vol. 56, no. 20, p. 504, 2007.

[2] O. S. Trialists, "Rehabilitation therapy services for stroke patients living at home: systematic review of randomised trials," *The Lancet*, vol. 363, no. 9406, pp. 352–356, 2004.

[3] S. Patel, H. Park, P. Bonato, and Chan, "A review of wearable sensors and systems with application in rehabilitation," *Journal of neuroengineering and rehabilitation*, vol. 9, no. 1, pp. 1–17, 2012.

[4] S. Patel, H. Park, P. Bonato, L. Chan, and M. Rodgers, "A review of wearable sensors and systems with application in rehabilitation," *Journal of neuroengineering and rehabilitation*, vol. 9, no. 1, pp. 1–17, 2012.

[5] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors," *IEEE communications surveys & tutorials*, vol. 15, no. 3, pp. 1192–1209, 2012.

[6] O. D. Incel, M. Kose, and C. Ersoy, "A review and taxonomy of activity recognition on mobile phones," *BioNanoScience*, vol. 3, no. 2, pp. 145–171, 2013.

[7] X. Su, H. Tong, and P. Ji, "Activity recognition with smartphone sensors," *Tsinghua science and technology*, vol. 19, no. 3, pp. 235–249, 2014.

[8] A. Wang, G. Chen, J. Yang, S. Zhao, and C.-Y. Chang, "A comparative study on human activity recognition using inertial sensors in a smartphone," *IEEE Sensors Journal*, vol. 16, no. 11, pp. 4566–4578, 2016.

[9] Y. Lu, Y. Wei, L. Liu, J. Zhong, L. Sun, and Y. Liu, "Towards unsupervised physical activity recognition using smartphone accelerometers," *Multimedia Tools and Applications*, vol. 76, no. 8, pp. 10 701–10 719, 2017.

[10] M. Egmont-Petersen, D. de Ridder, and H. Handels, "Image processing with neural networks—a review," *Pattern recognition*, vol. 35, no. 10, pp. 2279–2301, 2002.

[11] W. Rawat and Z. Wang, "Deep convolutional neural networks for image classification: A comprehensive review," *Neural computation*, vol. 29, no. 9, pp. 2352–2449, 2017.

[12] G. Lindsay, "Convolutional neural networks as a model of the visual system: past, present, and future," *Journal of Cognitive Neuroscience*, pp. 1–15, 2020.

[13] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[15] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[16] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big data*, vol. 3, no. 1, p. 9, 2016.

[17] T. G. Dietterich, "Machine learning for sequential data: A review," in *Joint IAPR international workshops on statistical techniques in pattern recognition (SPR) and structural and syntactic pattern recognition (SSPR)*. Springer, 2002, pp. 15–30.

[18] Z. Wang and T. Oates, "Imaging time-series to improve classification and imputation," *arXiv preprint arXiv:1506.00327*, 2015.

[19] K. P. Thanaraj, B. Parvathavarthini, U. J. Tanik, V. Rajinikanth, S. Kadry, and K. Kamalanand, "Implementation of deep neural networks to classify eeg signals using gramian angular summation field for epilepsy diagnosis," *arXiv preprint arXiv:2003.04534*, 2020.

[20] J.-L. Reyes-Ortiz, L. Oneto, A. Samà, X. Parra, and D. Anguita, "Transition-aware human activity recognition using smartphones," *Neurocomputing*, vol. 171, pp. 754–767, 2016.