

A long short-term memory convolutional neural network for first-person vision activity recognition

Girmaw Abebe^{1,2} and Andrea Cavallaro¹

¹Centre for Intelligent Sensing, Queen Mary University of London

²Technical Research Centre for Dependency Care and Autonomous Living, UPC-BarcelonaTech

¹{g.abebe, a.cavallaro}@qmul.ac.uk
²girmaw.abebe@upc.edu

Abstract

Temporal information is the main source of discriminating characteristics for the recognition of proprioceptive activities in first-person vision (FPV). In this paper, we propose a motion representation that uses stacked spectrograms. These spectrograms are generated over temporal windows from mean grid-optical-flow vectors and the displacement vectors of the intensity centroid. The stacked representation enables us to use 2D convolutions to learn and extract global motion features. Moreover, we employ a long short-term memory (LSTM) network to encode the temporal dependency among consecutive samples recursively. Experimental results show that the proposed approach achieves state-of-the-art performance in the largest public dataset for FPV activity recognition.

1. Introduction

First-person vision (FPV) activity recognition using wearable cameras is beneficial for assisted living [26, 36, 37], activity tracking [1, 3, 23, 24], life-logging and summarization [4, 5, 12, 17]. Activities of interest can be proprioceptive (e.g. walking) [3, 36, 37], person-to-object interactions (e.g. cooking) [8, 21] or person-to-person interactions (e.g. handshaking) [22, 25].

Proprioceptive activities are defined based on the full- or upper-body motion of the subject. Examples include *Run*, *Walk*, *Go upstairs* (cardiovascular activities) as well as *Sit* and *Stand* (states). However, unlike traditional third-person vision (TPV) problems, in FPV the body of the subject does not appear in the video. Moreover, because of the mounting position of the camera and the motion involved, FPV is characterized by outlier motions, motion blur and self-occlusions (Fig. 1). Effective encoding of the global motion



Figure 1: Sample frames showing some of the challenges in proprioceptive activity recognition in first-person vision: (a) outlier motions; (b) motion blur; and (c) self-occlusions.

is therefore necessary to accurately recognize proprioceptive activities.

Prior works on proprioceptive activity recognition used domain-specific handcrafted motion features [3, 17]. These features exploit motion magnitude and direction in time and frequency, and are often tailored to a specific problem. Because of the success of convolutional neural networks (CNNs) in image-based problems such as object recognition [6], deep frameworks have also been used for video-based activity recognition [16, 31, 34]. However, the recognition performance is still unsatisfactory due to the difficulty associated with the additional temporal dimension [16].

The recognition performance of deep frameworks may improve with the integration of handcrafted features [26, 31, 34]. Examples include the integration of learned spatio-temporal features with dense trajectory features [31, 34] and with optical flow features [26]. Deep frameworks for video-based activity recognition for TPV often discard camera motion [34] and are therefore ineffective to encode global motion for FPV. Moreover, existing recurrent neural networks are designed to encode only short-term motion dynamics [7]. The long-term temporal dependency among ac-

tivities has not been exploited yet.

In this paper we present a long short-term memory (LSTM) convolutional neural network for the continuous recognition of proprioceptive activities in FPV (Fig. 2). We employ two global motion streams: the mean grid optical-flow and the movement of the intensity centroid. We propose a global motion representation that encodes the dynamics in a video sample (*intra-sample encoding*) by scaling and translating its time-frequency motion representation (i.e. its spectrogram). A stacked spectrogram representation is derived for each global motion stream to enable 2D convolutions for learning and extracting global motion features. This approach reduces the number of network parameters and therefore the complexity. Importantly, the stacked spectrogram representation enables transfer learning from existing CNN models trained on large image datasets, such as ImageNet [6]. Moreover, the LSTM network exploits the long-term temporal dependency among different activities (*inter-sample encoding*). We validate the proposed framework against state-of-the-art temporal encoding methods on the largest public dataset of proprioceptive activities. The software of the proposed framework is available at <http://www.eecs.qmul.ac.uk/~andrea/fpv-lstm.html>.

The paper is organized as follows. Section 2 covers existing approaches that employ learning for video representation. Section 3 presents the proposed framework. Section 4 describes the dataset used for validation and the experimental results. Finally, Section 5 concludes the paper.

2. Related work

This section covers methods that use 3D convolutions to learn spatio-temporal features [18, 24, 31], apply temporal pooling across frame-level CNN features [9, 16, 26, 29, 34, 35], apply recursive networks [7, 18, 35], or employ ranking functions to encode temporal dependencies [10, 11].

3D convolution-based networks were proposed for videos as a direct mapping of the 2D convolutions for images. 3D convolution networks are complex and require larger datasets for training [31]. Two-stream networks encode appearance and temporal information separately using 2D convolutions [9, 18, 29, 34]. This approach requires effective temporal pooling to summarise motion.

Deep frameworks for the recognition of human activities in FPV mainly address object-interactive activities. These frameworks learn local hand-motions and objects using multi-stream networks [19, 29]. In addition to the spatial and temporal streams, Singh *et al.* [29] proposed an *ego-stream* consisting of complementary 2D and 3D convnets with a class score fusion. Similarly, Ma *et al.* [19] proposed appearance- and motion-based streams for concurrent object, action and activity recognition. The appearance stream consists of two segmentation and localization

sub-streams to segment hands and localise the object of interest. However, proprioceptive activities are mainly distinguished through global motion and therefore require an effective framework that captures short-term dynamics.

Existing TPV methods are not effective for FPV as they often discard global motion by subtracting the mean from the optical flow [28] or by using a homography matrix [34]. Poleg *et al.* [24] learned motion features from a volume of grid optical flow in FPV [24]. Ryoo *et al.* [26] used OverFeat [27] and Caffe [15] CNNs as spatial feature extractors. The histogram and the sum of time-series gradient provide multi-resolution temporal encoding on the frame-level CNN features. The histogram pooling encodes motion more effectively than the sum pooling [1]. However, each gradient pooling doubles the feature dimension and thus increases the computational cost.

Due to the difficulty associated with learning spatio-temporal features from raw images, early deep-learning frameworks for video-based activity recognition hardly outperformed frame-level inference [14, 16]. Stacks of optical flow vectors replacing raw images in a two-stream network can help addressing this problem [28]. Moreover, a network with two parallel streams can encode spatial and temporal information separately [9, 18, 29, 34, 35]. However, temporal pooling across frame-level CNN features does not encode fine details of the motion data, which are discarded by consecutive convolutions.

Fernando *et al.* [10, 11] proposed a video representation using rank pooling functions that order the frames chronologically. Recurrent neural networks (e.g. LSTM) learn the motion dynamics and are temporally deep and therefore increase the amount of data required for training when employed together with CNNs [7, 20, 35]. As a result, these networks are often applied to encode only short-term dynamics, e.g. 0.64 seconds [7].

3. Proposed method

Let $\mathcal{C} = \{A_c\}_{c=1}^C$ be a set of C activities and $\mathbf{V} = (V_1, \dots, V_n, \dots, V_N)$ be N temporally ordered activity samples, $V_n = (f_{n,1}, f_{n,2}, \dots, f_{n,l}, \dots, f_{n,L})$, that might overlap. Each activity sample contains L frames.

We aim to recognise the activity A_c^n taking place in V_n by encoding the short-term (*intra-sample*) and the long-term (*inter-sample*) dependency with the preceding T samples: $V_{n-1}, V_{n-2}, \dots, V_{n-T}$.

3.1. Intra-sample temporal encoding

Intra-sample encoding exploits the global motion dynamics in V_n using a CNN with 2D convolutions only. We encode the global motion between a pair of frames using two motion information sources, namely the frame-level mean of the grid optical flow, $J_{n,l}$, and the velocity of the intensity centroid, $K_{n,l}$, $l \in [1, L - 1]$ [3].

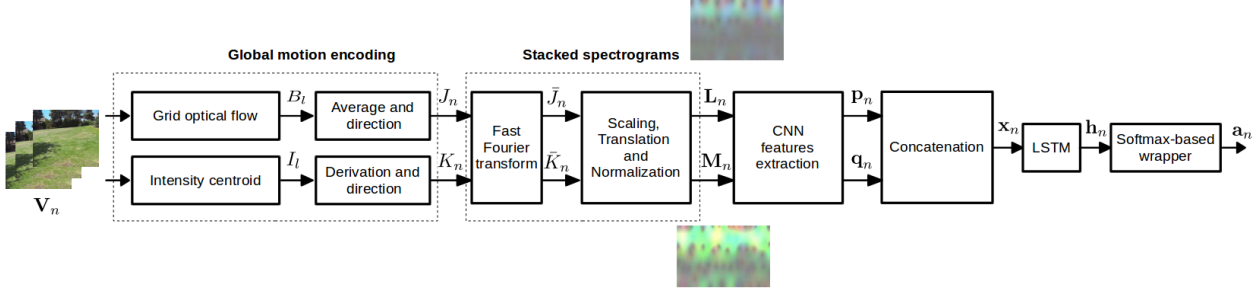


Figure 2: The proposed method for the recognition of proprioceptive activities in first-person videos.

We compute the grid optical flow for each subsequent pair of frames, $B_{n,l}$, $l \in [1, L-1]$, that contains horizontal, x , and vertical, y , components as¹ $B_l = B_l^x + jB_l^y$. We use the Horn-Schunk method [13] because its global smoothness assumption fits our problem of classifying ego-motion. We chose a grid representation to reduce the computational cost when encoding the global motion.

Let g be the number of grids in the horizontal and vertical dimensions of B_l . We compute the corresponding mean of optical flow as $J_l^x = (1/g^2) \sum B_l^x$ and $J_l^y = (1/g^2) \sum B_l^y$.

The intensity centroid of a frame, f_l , which is H -pixels high and W -pixels wide, is derived from the first-order image moments, \mathcal{M}_{pq} , where $p, q \in \{0, 1\}$. \mathcal{M}_{pq} is calculated as the weighted average of all the intensity values in f_l as $\mathcal{M}_{pq}^l = \sum_{r=1}^H \sum_{c=1}^W r^p c^q f_l(r, c)$. Similarly to [3], we compute the velocity of the intensity centroid, $K_l = K_l^x + jK_l^y$, from the first-order derivative of the centroids as $K_l^x = I_{l+1}^x - I_l^x$ and $K_l^y = I_{l+1}^y - I_l^y$, where $I_l^x = \mathcal{M}_{01}^l / \mathcal{M}_{00}^l$ and $I_l^y = \mathcal{M}_{10}^l / \mathcal{M}_{00}^l$.

The final global motion data of V_n consists of $J_n = (J_1, J_2, \dots, J_{L-1})$ and $K_n = (K_1, K_2, \dots, K_{L-1})$. We encode the intra-sample dynamics from J_n and K_n using a frequency-domain analysis, and we employ time-frequency representation (spectrogram) of the motion data. The spectrogram contains the frequency response magnitude of the global motion at different frequency bins. We apply the fast Fourier transform (FFT) on each axial component of J_n and K_n .

In order to encode high-level CNN features from the spectrograms with 2D convolutions only, we stack the spectrograms of the horizontal, vertical and direction components of J_n and K_n , into 3-channel motion representations, \mathbf{L}_n and \mathbf{M}_n , respectively. The direction spectrogram is included to exploit its discriminating characteristics (note that this differs from Ng *et al.* [35] who filled the third channel with zeros).

The three spectrogram components from J_n are computed as $\bar{J}_n^x = \mathcal{F}(J_n^x)$, $\bar{J}_n^y = \mathcal{F}(J_n^y)$, $\bar{J}_n^\theta = \mathcal{F}(J_n^\theta)$, where

¹For simplicity we drop the subscript n .

$\mathcal{F}(\cdot)$ represents the magnitude of the fast Fourier transform and $J_n^\theta = \arctan(J_n^y / J_n^x)$. The stack of spectrograms from J_n becomes $\bar{\mathbf{J}}_n = (\bar{J}_n^x, \bar{J}_n^y, \bar{J}_n^\theta)$. Similarly, $\bar{K}_n^x = \mathcal{F}(K_n^x)$, $\bar{K}_n^y = \mathcal{F}(K_n^y)$ and $\bar{K}_n^\theta = \mathcal{F}(K_n^\theta)$ are computed from K_n , where $K_n^\theta = \arctan(K_n^y / K_n^x)$. The stack of spectrograms from K_n becomes $\bar{\mathbf{K}}_n = (\bar{K}_n^x, \bar{K}_n^y, \bar{K}_n^\theta)$. Similarly to [7], we apply scaling, translation and normalization on $\bar{\mathbf{J}}_n$ and $\bar{\mathbf{K}}_n$ to limit their values in $[0, 255]$ and obtain \mathbf{L}_n and \mathbf{M}_n , respectively. If α and τ are the scaling and translation factors, \mathbf{L}'_n is computed as follows: $\mathbf{L}'_n = \alpha * \bar{\mathbf{J}}_n + \tau$; $\mathbf{L}''_n = \max(0, \mathbf{L}'_n)$ and $\mathbf{L}_n = \min(255, \mathbf{L}''_n)$. Similarly \mathbf{M}'_n is computed from $\bar{\mathbf{K}}_n$ followed by \mathbf{M}''_n and \mathbf{M}_n . The scaling, translation and normalization operations facilitate transfer learning from image datasets, e.g. using CNN models that are pre-trained on ImageNet [6].

We employ CNN models to extract high-level global motion features from the low-level spectrogram representations \mathbf{L}_n and \mathbf{M}_n . This improves the generalizing capability of the features. Since the motion is represented as a stacked spectrogram, it is possible to employ a sequence of 2D convolution filters to extract the high-level intra-sample features. In addition to the benefit of transfer learning, our 2D CNN-based approach reduces the number of network parameters and hence the amount of data required for training. This is useful as FPV datasets are much smaller than TPV datasets (e.g. Sports-1M [16]).

As a feature extractor we use GoogleNet [30], whose inception module contains multiple convolution outputs of different filter sizes in parallel rather than in cascade. It is a common practice to utilise ImageNet-trained models on other data such optical flow images [7]. Though the spectrogram information is different from the ImageNet dataset on which GoogleNet is trained, the scaling, translation and normalization operations followed by stacking help to mimic RGB image characteristics. The plausibility of inception features can be improved by retraining, ideally, the whole network or the last layers using spectrograms. We extract the inception features \mathbf{p}_n and $\mathbf{q}_n \in \mathbb{R}^D$, from \mathbf{L}_n and \mathbf{M}_n , respectively. We combine these features as $\mathbf{x}_n = (\mathbf{p}_n, \mathbf{q}_n)^T$, where $(\cdot)^T$ represents the transpose operation.

The concatenated intra-sample inception feature, $\mathbf{x}_n \in \mathbb{R}^{2D}$, encodes the temporal evolution of motion magnitude and direction inside a segment, which is later used as an input to the inter-sample temporal encoding.

3.2. Inter-sample temporal encoding

We aim to exploit the long-term temporal relationships among consecutive samples (i.e. inter-sample encoding) to improve inference. To this end we use a recurrent neural network (RNN) that uses previous hidden information, $\mathbf{h}_{n-1} \in \mathbb{R}^\nu$, to estimate the current hidden state, $\mathbf{h}_n \in \mathbb{R}^\nu$, where ν is the number of neurons in the hidden layer.

The vanishing and exploding gradient problems in basic RNNs hinder learning long-term temporal dependencies. A *vanishing* gradient happens when it becomes zero due to consecutive multiplications of small gradient values across T temporal indices. This phenomenon incorrectly signals optimal learning of the network parameters. An *exploding* gradient happens when it becomes too large to minimize due to its consecutive multiplications across temporal indices. This may saturate the weights at the high level and incorrectly signal a high discriminative capability.

To overcome the vanishing and exploding gradient problems, we employ an LSTM network that uses three additional gates: forget, input and output. These gates act as switches for monitoring the information flow from the current input, \mathbf{x}_n , and previous hidden state, \mathbf{h}_{n-1} , to the current hidden state, \mathbf{h}_n , via the cell state, \mathbf{c}_n .

The forget gate, \mathbf{f}_n , helps to discard less useful information from the previous cell state, \mathbf{c}_{n-1} , as

$$\mathbf{f}_n = \sigma(W_{xf}\mathbf{x}_n + W_{hf}\mathbf{h}_{n-1} + \mathbf{b}_f), \quad (1)$$

where $\sigma(\cdot)$ represents the *sigmoid* activation function and \mathbf{b}_f is the bias in the forget gate.

The input gate, \mathbf{i}_n , weights the candidate cell information, $\bar{\mathbf{c}}_n$, to be the current state of the cell, \mathbf{c}_n , as

$$\mathbf{i}_n = \sigma(W_{xi}\mathbf{x}_n + W_{hi}\mathbf{h}_{n-1} + \mathbf{b}_i), \quad (2)$$

$$\bar{\mathbf{c}}_n = \phi(W_{xc}\mathbf{x}_n + W_{hc}\mathbf{h}_{n-1} + \mathbf{b}_c), \quad (3)$$

$$\mathbf{c}_n = \mathbf{f}_n \odot \mathbf{c}_{n-1} + \mathbf{i}_n \odot \bar{\mathbf{c}}_n, \quad (4)$$

where $\phi(\cdot)$ represents the *tanh* activation function, \odot is an element-wise product, \mathbf{b}_i and \mathbf{b}_c represent the input gate and the memory cell biases, respectively.

The output gate, \mathbf{o}_n , evaluates the cell information, \mathbf{c}_n , to predict \mathbf{h}_n as

$$\mathbf{o}_n = \sigma(W_{xo}\mathbf{x}_n + W_{ho}\mathbf{h}_{n-1} + \mathbf{b}_o), \quad (5)$$

$$\mathbf{h}_n = \mathbf{o}_n \odot \phi(\mathbf{c}_n), \quad (6)$$

where \mathbf{b}_o represents the bias in the output gate.

The weight parameters W_{hf} , W_{hi} , W_{hc} and $W_{ho} \in \mathbb{R}^{\nu \times \nu}$ describe the relationship between the previous hidden state, \mathbf{h}_{n-1} , and the remaining states, \mathbf{f}_n , \mathbf{i}_n , \mathbf{c}_n and

$\mathbf{o}_n \in \mathbb{R}^\nu$, respectively, where ν represents the number of neurons used in each of the states. The parameters W_{xf} , W_{xi} , W_{xc} and $W_{xo} \in \mathbb{R}^{\nu \times 2D}$ describe the relationship between $\mathbf{x}_n \in \mathbb{R}^{2D}$ and the remaining states.

Finally, an output projection wrapper is applied that provides the activity prediction vector, $\mathbf{a}_n \in \mathbb{R}^C$, for V_n as

$$\mathbf{a}_n = \frac{e^{W_{ha}\mathbf{h}_n}}{\sum_{c=1}^C e^{W_{ha}\mathbf{h}_n}}, \quad (7)$$

using the softmax normalization and $W_{ha} \in \mathbb{R}^{C \times \nu}$ is the wrapping matrix.

The class with the maximum score in \mathbf{a}_n is the winning class, A_c^n .

4. Experiments

In this section we describe the setting of parameters used in the intra-sample and the inter-sample encoding stages, the state-of-the-art methods selected for comparison with the proposed approach and the dataset used for validation.

4.1. Methods under comparison

We compare the proposed approach against four state-of-the-art video representation methods, namely C3D (3D convolutional networks [31]); TDD (trajectory-pooled deep descriptors [34]); VD (VideoDarwin [10]); and TGP (time-series gradient pooling [26]).

C3D employs 3D convolutions to learn spatio-temporal features and hence requires large datasets for training. TDD is a highly discriminative video representation that encodes both spatial and temporal streams. Unlike dense representations [32], TDD exploits a trajectory representation [33] that takes into account the camera motion. VD uses ranking functions and handles handcrafted or CNN features [10]. TGP is used as a baseline method, which employs histogram and sum pooling of each time-series feature element.

All experiments are conducted with 100 iterations and the average performance of the iterations is reported as a final recognition result.

4.2. Dataset and train-test split


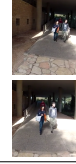


HUJI² is the largest public dataset for FPV activity recognition [24]. A head-mounted camera is used for collection. We used the 15 hours (h) subset that contains the following activities: *Go upstairs*, *Run*, *Walk*, *Sit/Stand* and *Static* (see Table 1).

Approximately 7.5 h of video (50% of the dataset) is collected from publicly available YouTube videos. Examples include *Go upstairs* sequences with significant illumination changes and *Run* sequences that contain outlier mo-

²<http://www.vision.huji.ac.il/egoseg/videos/dataset.html>

Table 1: Number of video segments and their duration per activity in the HUJI dataset [24]. Note the class imbalance between *Run* (47%) and *Walk* (7%). (#: number; min: minutes. ‘% of total’ is the ratio of the duration of an activity to the total duration of the dataset).

	Go upstairs	Run	Walk	Sit/Stand	Static	Total
Segments (#)	13	13	19	26	14	85
Duration (min)	151	409	62	143	104	869
% of total	17	47	7	16	12	100

	Go upstairs	Run	Walk	Sit/Stand	Static
Key frames					

tions (e.g. other runners). Since we are interested in activities produced by full- or upper-body motion, we discarded videos where the subject travels by car or by bus, or rides a bicycle. We merge *Sit* and *Stand* as single *Sit/Stand* state since they may both involve large head motion when the subject is mostly stationary. *Static* is included as a reference as it does not contain a significant head and/or body motion.

We applied equal decomposition of the sequences to train and test sets. Among the 44 sequences in the dataset, the first 22 videos are used for testing and the remaining 22 videos are used for training. Due to the class imbalance problem shown in Table 1, we select the second half for training as it contains an equivalent number of samples among activities.

The performance measures to evaluate the recognition performance are precision, \mathcal{P} , recall, \mathcal{R} , and F-score, \mathcal{F} :

$$\mathcal{P} = 100 \frac{tp}{tp + fp}, \quad (8)$$

$$\mathcal{R} = 100 \frac{tp}{tp + fn}, \quad (9)$$

$$\mathcal{F} = 100 \frac{2\mathcal{P}\mathcal{R}}{\mathcal{P} + \mathcal{R}}, \quad (10)$$

where tp is the number of true positives, fp is the number of false positives and fn is the number of false negatives. We first evaluate \mathcal{P} and \mathcal{R} per class and finally report their average as the overall system performance. The overall \mathcal{F} is computed from the averaged \mathcal{P} and \mathcal{R} .

4.3. Parameters

We set the length of an activity sample to 3 seconds, i.e. $L = 90$ frames for the 30 fps frame rate in the HUJI dataset. We also resize the videos to a resolution of 320×240 . For grid optical flow computation, we set the number of grids as $g = 100$ in each of the horizontal and vertical dimensions. We apply the FFT on each window of 15 frames in the sample with a stride of one frame to generate the spectrograms

in the intra-sample encoding. The scaling factor of $\alpha = 16$ and translation of $\tau = 128$ are used in the stacked spectrogram representation.

We use inception-v3 trained on ImageNet [6] to extract the CNN or inception features on the spectrogram images. The inception-v3 reaches the top-5 error rate of 3.46% on ImageNet. We extracted the features from the next-to-last layer of inception-v3, i.e. ‘pool_3 : 0’, which provides $D = 2,048$ dimensional high-level global motion feature.

For the LSTM network, we focus on its simplicity due to the limited dataset size and high dimensional feature input. Thus, we use only a single hidden layer, which contains $\nu = 128$ neurons trained with a batch size of 100 and with 80 epochs. We set the recursive duration to contain $T = 20$ samples and the learning rate to be 0.01.

For VD, similarly to [3] we use as input for the ranking functions concatenated histograms of motion magnitude and direction with 15 and 36 bins, respectively. This results in a feature vector with $D = 102$. We use the C3D model pre-trained on the Sports-1M dataset [16] for C3D feature extraction. For an activity sample of $L = 90$ frames, $D = 4,096$ long C3D feature is extracted from the sixth layer ($fc6 - 1$). Average pooling of the C3D features extracted for each chunk of 16 frames is performed for the final C3D representation of the activity sample.

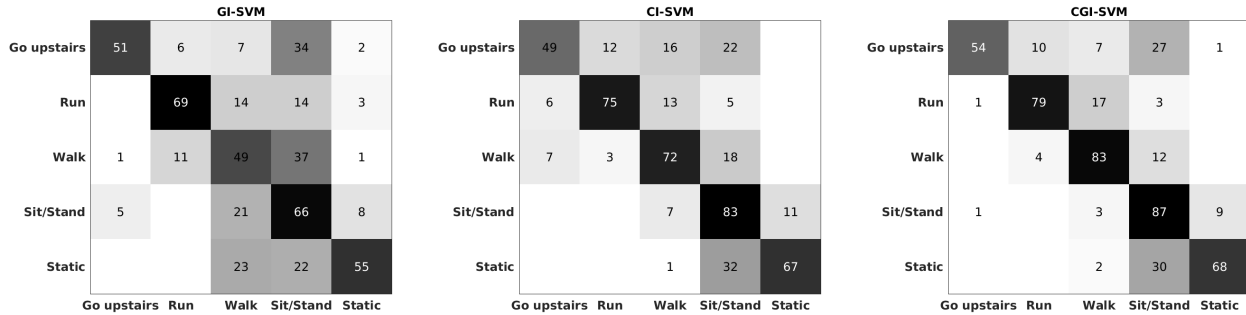
We use the TDD model pre-trained on the UCF101 dataset. Appearance features are extracted from *conv4* and *conv5* layers of the spatial stream, and motion feature are extracted from *conv3* and *conv4* layers of the temporal stream. Each layer provides $D = 512$ long feature vector. We also apply both spatio-temporal and channel normalizations. The final TDD feature for an activity sample is $D = 4,096$ feature vector.

TGP is derived by applying sum and histogram pooling on the gradient of frame-level inception features. TGP provides $D = 12,288$ dimension from input of $D = 2,048$ inception feature. A one-vs-all support vector machine (SVM) classifier is used to validate the state-of-the-art methods and the proposed intra-sample encoding.

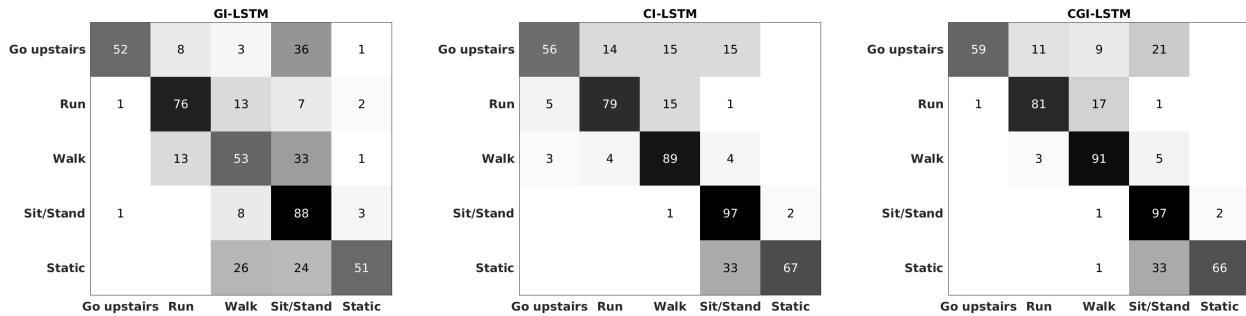
4.4. Discussion

The confusion matrices in Figure 3 evaluate the combination of inception features from grid optical flow data and the movement of the intensity centroid. The performance of both intra-sample and inter-sample encoding are improved when grid inception (GI) and centroid inception (CI) features are concatenated (i.e. CGI). Misclassifications of *Go upstairs* to *Sit/Stand* happen when subjects *Stand* to take a rest during *Going upstairs*. Moreover, *Run* and *Walk* are sometimes misclassified due to the similarity of their motion dynamics. The stationary nature of the subject during *Sit/Stand* and *Static* also causes misclassifications.

Table 2 compares the performance of state-of-the-art



(a) Intra-sample temporal encoding: inception features + SVM



(b) Inter-sample temporal encoding: inception features + LSTM

Figure 3: Performance improvement by combining grid-based (GI) and centroid-based (CI) inception features in (a) intra-sample and (b) inter-sample temporal encoding.

Table 2: Average per-class recall, \mathcal{R} , precision, \mathcal{P} , and F-score, \mathcal{F} , of existing methods compared with the proposed intra-sample encoding framework. An SVM classifier is used for all methods. (Proposed*: the SVM output after intra-sample encoding without inter-sample encoding.)

Method	$\mathcal{P}(\%)$	$\mathcal{R}(\%)$	$\mathcal{F}(\%)$
TGP [26]	57	61	59
VD [10]	59	62	61
C3D [31]	64	65	65
TDD [34]	63	73	68
Proposed*	70	74	72

methods with the proposed intra-sample encoding. The features from all methods are validated on an SVM classifier. The concatenation of inception-features extracted from the grid-based and centroid-based stacked spectrograms (Proposed*) outperforms existing methods. VD [10] has the smallest feature dimension ($D = 102$), but achieves slightly higher performance than the baseline TGP [26], which employs the histogram and sum pooling of the time-series gradient of the inception features. TDD [34] uses multi-stream handcrafted and learned features, and is second to the proposed framework.

Table 3 evaluates the LSTM-based inter-sample encoding of the proposed framework and its effectiveness when

applied on state-of-the-art methods. The proposed framework achieves the best performance in the majority of the activities, i.e. *Run* (81%), *Walk* (91%) and *Sit/Stand* (97%); but it is inferior to TDD [34] and C3D [31] in recognising *Go upstairs* and *Static*, respectively. This is because our framework uses motion only, while TDD and C3D also include spatial (appearance) information. For example, *Go upstairs* can be better distinguished using the appearance features of the staircases. In addition, the recognition of the *Static* state can exploit the appearance information of similar indoor environments available in the dataset.

5. Conclusion

We proposed a long short-term memory convolutional neural network that recognises human activities from first-person videos. The network uses a global motion representation that enables the encoding of temporal information with a 2D CNN. In addition to its simplicity, the proposed representation enables transferring knowledge from image datasets and reduces the need of large problem-specific training data. On top of the CNN-based intra-sample temporal encoding, we proposed an LSTM-based encoding of long-term temporal dependencies among samples. Results showed the benefit of the combination of grid-based

Table 3: Per-class recall performance, $\mathcal{R}(\%)$, with and without the proposed LSTM-based inter-sample encoding.

	Without LSTM					With LSTM				
	Go upstairs	Run	Walk	Sit/Stand	Static	Go upstairs	Run	Walk	Sit/Stand	Static
TGP [26]	52	34	82	57	81	52	34	83	60	84
VD [10]	54	67	46	73	70	55	71	55	89	89
C3D [31]	67	74	73	57	53	63	68	74	47	94
TDD [34]	68	76	95	52	72	70	71	86	83	39
Proposed	54	79	83	87	68	59	81	91	97	66

and centroid-based motion features as well as of the intra-sample and the inter-sample temporal encoding strategies.

The proposed network can be used in multi-modal problems using a fusion strategy that accounts for the independent discriminative characteristics of feature groups [2].

As future work, we plan to reduce the dimensionality of the input to the LSTM network by simplifying the temporal encoding complexity to speed-up the training phase.

Acknowledgment

G. Abebe was supported in part by the Erasmus Mundus Joint Doctorate in Interactive and Cognitive Environments, which is funded by the EACEA Agency of the European Commission under EMJD ICE FPA no 2010-2012.

References

- [1] G. Abebe and A. Cavallaro. Hierarchical modeling for first-person vision activity recognition. *Neurocomputing*, 267:362–377, June 2017. [1](#), [2](#)
- [2] G. Abebe and A. Cavallaro. Inertial-Vision: cross-domain knowledge transfer for wearable sensors. In *Proc. of International Conference on Computer Vision Workshops (ICCVW)*, Venice, Italy, October 2017. [7](#)
- [3] G. Abebe, A. Cavallaro, and X. Parra. Robust multi-dimensional motion features for first-person vision activity recognition. *Computer Vision and Image Understanding (CVIU)*, 149:229–248, 2016. [1](#), [2](#), [3](#), [5](#)
- [4] Y. Bai, C. Li, Y. Yue, W. Jia, J. Li, Z.-H. Mao, and M. Sun. Designing a wearable computer for lifestyle evaluation. In *Proc. of Northeast Bioengineering Conference (NEBEC)*, pages 93–94, Philadelphia, USA, March 2012. [1](#)
- [5] N. Caprani, N. E. O’Connor, and C. Gurrin. Investigating older and younger peoples’ motivations for lifelogging with wearable cameras. In *Proc. of IEEE International Symposium on Technology and Society (ISTAS)*, 2013. [1](#)
- [6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255, Miami, USA, June 2009. [1](#), [2](#), [3](#), [5](#)
- [7] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2625–2634, Boston, USA, June 2015. [1](#), [2](#), [3](#)
- [8] A. Fathi. *Learning Descriptive Models of Objects and Activities from Egocentric Video*. PhD thesis, Georgia Institute of Technology, 2013. [1](#)
- [9] C. Feichtenhofer, A. Pinz, and A. Zisserman. Convolutional two-stream network fusion for video action recognition. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1933–1941, Las Vegas, USA, June 2016. [2](#)
- [10] B. Fernando, E. Gavves, J. Oramas, A. Ghodrati, and T. Tuytelaars. Rank pooling for action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 39(4):773–787, 2017. [2](#), [4](#), [6](#), [7](#)
- [11] B. Fernando, E. Gavves, J. M. Oramas, A. Ghodrati, and T. Tuytelaars. Modeling video evolution for action recognition. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5378–5387, Boston, USA, June 2015. [2](#)
- [12] S. Hodges, E. Berry, and K. Wood. SenseCam: A wearable camera that stimulates and rehabilitates autobiographical memory. *Memory*, 19(7):685–696, October 2011. [1](#)
- [13] B. K. P. Horn and B. G. Schunck. Determining optical flow. *Artificial Intelligence*, 17(1-3):185 – 203, 1981. [3](#)
- [14] S. Ji, W. Xu, M. Yang, and K. Yu. 3D convolutional neural networks for human action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 35(1):221–231, 2013. [2](#)
- [15] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: convolutional architecture for fast feature embedding. In *Proc. of ACM International Conference on Multimedia*, pages 675–678, Florida, USA, November 2014. [2](#)
- [16] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1725–1732, Ohio, USA, June 2014. [1](#), [2](#), [3](#), [5](#)
- [17] K. M. Kitani, T. Okabe, Y. Sato, and A. Sugimoto. Fast unsupervised ego-action learning for first-person sports videos. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3241–3248, Colorado, USA, June 2011. [1](#)
- [18] Q. Li, Z. Qiu, T. Yao, T. Mei, Y. Rui, and J. Luo. Action recognition by learning deep multi-granular spatio-temporal video representation. In *Proc. of ACM on International Conference on Multimedia Retrieval*, pages 159–166, Amsterdam, The Netherlands, October 2016. [2](#)
- [19] M. Ma, H. Fan, and K. M. Kitani. Going deeper into first-person activity recognition. In *Proc. of IEEE Conference*

- on *Computer Vision and Pattern Recognition (CVPR)*, pages 1894–1903, Las Vegas, USA, June 2016. 2
- [20] S. Ma, L. Sigal, and S. Sclaroff. Learning activity progression in LSTMs for activity detection and early detection. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1942–1950, Las Vegas, USA, June 2016. 2
- [21] K. Matsuo, K. Yamada, S. Ueno, and S. Naito. An attention-based activity recognition for egocentric video. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 565 – 570, Columbus, USA, June 2014. 1
- [22] S. Narayan, M. S. Kankanhalli, and K. R. Ramakrishnan. Action and interaction recognition in first-person videos. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 526 – 532, Columbus, USA, June 2014. 1
- [23] H. Pirsiavash and D. Ramanan. Detecting activities of daily living in first-person camera views. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2847 – 2854, Providence, USA, June 2012. 1
- [24] Y. Poleg, A. Ephrat, S. Peleg, and C. Arora. Compact CNN for indexing egocentric videos. In *Proc. of IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–9, New York, USA, March 2016. 1, 2, 4, 5
- [25] M. S. Ryoo and L. Matthies. First-person activity recognition: What are they doing to me? In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2730 – 2737, Portland, USA, June 2013. 1
- [26] M. S. Ryoo, B. Rothrock, and L. Matthies. Pooled motion features for first-person videos. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 896–904, Boston, USA, March 2015. 1, 2, 4, 6, 7
- [27] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. In *Proc. of International Conference on Learning Representations (ICLR)*, Banff, Canada, April 2014. 2
- [28] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in Neural Information Processing Systems*, pages 568–576, Montreal, Canada, December 2014. 2
- [29] S. Singh, C. Arora, and C. V. Jawahar. First person action recognition using deep learned descriptors. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2620–2628, Las Vegas, USA, June 2016. 2
- [30] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9, Boston, USA, June 2015. 3
- [31] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3D convolutional networks. In *Proc. of IEEE International Conference on Computer Vision (ICCV)*, pages 4489–4497, Santiago, Chile, December 2015. 1, 2, 4, 6, 7
- [32] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu. Dense trajectories and motion boundary descriptors for action recognition. *International Journal of Computer Vision (IJCV)*, 103(1):60–79, 2013. 4
- [33] H. Wang and C. Schmid. Action recognition with improved trajectories. In *Proc. of IEEE International Conference on Computer Vision (ICCV)*, pages 3551–3558, Sydney, Australia, December 2013. 4
- [34] L. Wang, Y. Qiao, and X. Tang. Action recognition with trajectory-pooled deep-convolutional descriptors. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4305–4314, Boston, USA, June 2015. 1, 2, 4, 6, 7
- [35] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici. Beyond short snippets: Deep networks for video classification. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4694–4702, Boston, USA, June 2015. 2, 3
- [36] K. Zhan, S. Faux, and F. Ramos. Multi-scale conditional random fields for first-person activity recognition on elders and disabled patients. *Pervasive and Mobile Computing*, 16, Part B:251–267, January 2015. 1
- [37] H. Zhang, L. Li, W. Jia, J. D. Fernstrom, R. J. Scabassi, Z.-H. Mao, and M. Sun. Physical activity recognition based on motion in images acquired by a wearable camera. *Neurocomputing*, 74(12):2184–2192, June 2011. 1