Deep 3D Flow Features for Human Action Recognition

Athanasios Psaltis
Centre for Research and Technology Hellas
at.psaltis@iti.gr

Georgios Th. Papadopoulos
Centre for Research and Technology Hellas
papad@iti.gr

Petros Daras
Centre for Research and Technology Hellas
daras@iti.gr

***Abstract*—The present work investigates the use of 3D flow information for performing Deep Learning (DL)-based human action recognition. Generally, 3D flow fields include rich and fine-grained information, regarding the motion dynamics of the observed human actions. However, despite the great potentials present, 3D flow has not been widely used, mainly due to challenges related to the efficient modeling of the flow information and the addressing of the respective computational complexity issues. In this paper, different techniques are investigated for incorporating 3D flow information in DL action recognition schemes. In particular, a novel sequence modeling approach is introduced, which combines the advantageous characteristics for spatial correlation estimation of Convolutional Neural Networks (CNNs) with the increased temporal modeling capabilities of Long Short Term Memory (LSTM) models. Additionally, an extended CNN-based deep flow model is proposed that extracts features from both the spatial and temporal domains, by applying 3D convolutions; hence, modeling the action dynamics within consecutive frames. Moreover, for compact and efficient 3D motion feature extraction, the combined use of CNNs with a 'flow colorization' approach is adopted. The proposed methods significantly outperform similar DL and hand-crafted 3D flow approaches, and compare favorably with most skeleton-based techniques in the currently most challenging public dataset, namely the NTU RGB-D.**

***Index Terms*—Action recognition, 3D flow, Deep Learning**

I. Introduction

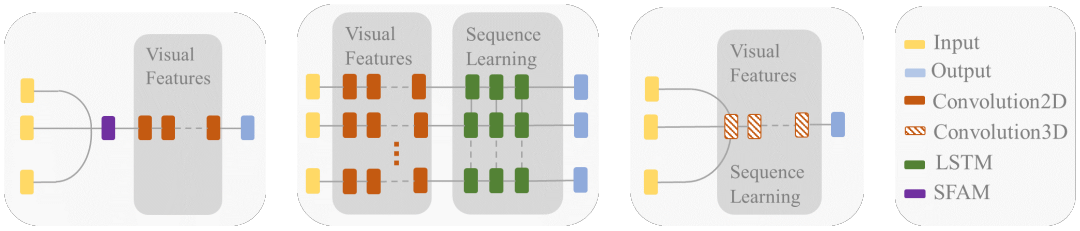
Human action recognition has recently attracted a lot of attention due to its very wide set of possible application fields, ranging from surveillance and robotics to gaming and e-learning. Several researchers have devoted increased resources for accomplishing reliable solutions [1]. However, despite the huge plethora of 2D approaches that have already been proposed, 3D action recognition in the general case constitutes an open research topic of wide interest in the field of computer vision.

For reaching robust action recognition performance, several challenges need to be addressed, including, among others, the difference in the appearance of the individuals, difference in the execution of the same action by different subjects, different action execution speed, *etc*. [2]. Initially, research on human action recognition focused on designing appearance-based representations, based only on the processing of RGB information, due to the availability of large RGB datasets for training and evaluation purposes [3]. However, with the recent introduction of corresponding large-scale 3D resources, such as the NTU RGB-D dataset [4], significant boost has been given in the field. The provided depth maps include a great wealth of information and can significantly improve the RGB-based performance; hence, shifting the analysis focus to the 3D space.

According to the type of input stream, RGB-D action recognition methods are roughly divided into the following main types: a) surface (depth), b) skeleton-tracking, and c) flow ones. Surface methods make use of only the captured depth maps (or the computed surface normal vectors) for estimating a representation of the human subject pose and subsequently modeling the action dynamics [5]–[7]. On the other hand, skeleton-tracking methods make extensive use of domain knowledge, regarding the appearance and the topological characteristics of the human body, relying on the tracking of human body parts over time. This is the most popular category of methods, where the aim is to produce discriminative representations of the tracked human skeleton [8]–[11]. Moreover, flow-based methods have also been explored, which combine depth with RGB information for estimating more discriminative representations (namely 3D flow fields) that enable the focus of the analysis procedure on the areas where motion has been observed. Munaro *et al*. [12] introduce a grid-based motion descriptor, by estimating correspondences between point-clouds belonging to consecutive frames. Histograms of local 3D motion are used in [13], taking into account spatio-temporal interest points. Furthermore, Fanello *et al*. [14] present an effective real-time system for one-shot action modeling and recognition, using histograms of 3D flow. In [15], Papadopoulos *et al*. introduce a set of local/global-level 3D flow descriptors, which incorporate spatial and surface information in the flow representation, while efficiently encoding the global motion characteristics in a compact way.

The recent trend in the computer vision field, the so-called 'Deep Learning' (DL) paradigm, relies on the use of data-driven methods for automatically learning optimal features. The latter has shown outstanding performance in multiple image analysis tasks, including object detection [16], [17], concept detection [18], [19] and image classification [20],

978-1-5386-7021-7/18/$31.00



a) b) c)

Fig. 1: Examined architectures for modeling visual feature correlations and learning sequential motion patterns: a) SFAM scheme , b) proposed CNN-LSTM-based, and c) proposed C3D-based.

[21]. Until now, DL schemes for action recognition have in principle focused on the use of skeleton-tracking data, *i.e*. they make extensive use of domain knowledge (employed skeleton-tracker), and relatively straight-forward implementations. In [4], a Part-aware Long Short-Term Memory (P-LSTM) model is proposed as an extension of the conventional LSTM, targeting the learning of correlations among different body parts. A tree structure spatio-temporal design of a LSTM is proposed in [22], which models spatial dependencies among joints and temporal correlations among frames at the same time. Huang *et al*. [23] incorporate the Lie group structure into a deep network architecture to learn more appropriate Lie group features for 3D action recognition. Moreover, a new class of LSTM network, termed as the Global Context-Aware Attention LSTM (GCA-LSTM), for 3D action recognition is introduced in [24], which is able to selectively focus on the informative joints in the action sequence with the assistance of global contextual information.

Not surprisingly, DL techniques have also been applied to 3D flow action-recognition problems. In [25], a representation based on the processing of 3D flow fields is introduced, termed Scene Flow Action Map (SFAM), which encodes the flow sequence into a single action template. However, the latter has been designed based solely on the use of 2D CNNs, aiming at estimating complex patterns along the spatial dimensions.

In this context, the main research contribution of this work comprises the design of several spatio-temporal architectures for efficiently modeling detailed motion patterns encoded in multiple adjacent frames, either by combining the increased temporal modeling capabilities of LSTMs with the advantageous characteristics for spatial correlation estimation of CNNs or by applying 3D convolutions directly to spatio-temporal data.

In this paper, the problem of 3D flow-based human action recognition using DL techniques is investigated. The main contributions of this work are summarized as follows:

- **A novel sequence modeling approach for efficiently modeling fine-grained, yet compact and discriminative, spatio-temporal features** for robustly encoding 3D human action dynamics, by combining the advantageous characteristic for spatial correlation estimation of CNNs with the increased temporal modeling capabilities of LSTMs.
- **An extended CNN-based deep flow model** that extracts features from the both spatial and temporal domains, by applying 3D convolutions; hence, modeling the action dynamics in multiple consecutive frames.
- **A new processing and representation scheme of 3D flow information** that leverages the learning and discrimination capabilities of CNNs for human action recognition, coupled with a 'flow colorization' approach.

The proposed approaches significantly outperform similar DL and hand-crafted 3D flow methods, and compare favorably with most skeleton-based techniques in the currently most challenging public dataset, namely the NTU RGB-D [4].

The remainder of the paper is organized as follows: The proposed 3D flow action recognition method is presented in Section II. Experimental results are discussed in Section III and conclusions are drawn in Section IV.

II. Action Recognition Realization

Human actions inherently include a temporal dimension (the so called 'action dynamics'), the capturing and encoding of which is of paramount importance for achieving robust recognition performance. Despite the fact that information streams containing a great wealth of information are available (*e.g*. 3D flow), they have not received particular attention so far in the context of DL-based 3D action recognition methods. In this respect, new methodologies for processing and representing flow information are described in this section. One of the major challenges, regarding 3D flow estimation, concerns the corresponding computational requirements, which have hindered its widespread use so far. However, computationally efficient 3D flow estimation algorithms have recently been introduced with satisfactory flow computation accuracy. In this work, the algorithm of [26] has been employed, which exhibits a processing rate equal to 24 frames per second (fps).

The SFAM architecture, which is presented in Fig. 1a, is currently the only approach that utilizes 3D flow estimations, while exploiting the increased computational capabilities of DL. The latter encodes a video sample into a single dynamic image, by taking into account consecutive 3D flow fields. It can be seen that the core part of the SFAM is based solely on the use of 2D CNNs, focusing on the spatial domain analysis. This raises the question whether such a template matching approach is capable of modeling the complex spatio-temporal structure of human actions. To this end, CNN-LSTM and 3D CNNs architectures are considered in this work. The proposed methods are depicted in Fig. 1b and Fig. 1c, respectively. The CNN-LSTM architecture involves the use of Convolutional

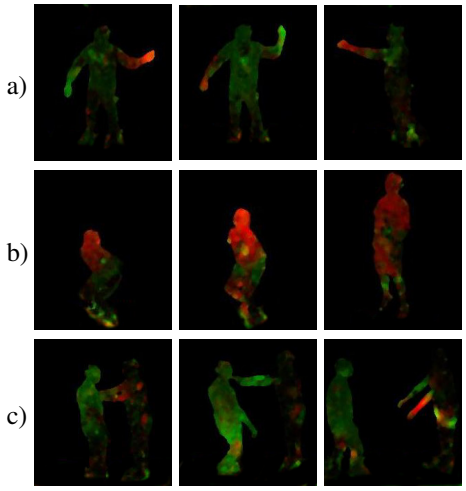


Fig. 2: Exemplary colorized frames of 3D flow for actions: a) 'Throw', b) 'Jump up' and c) 'Pushing other person'.

Neural Network (CNN) layers for per frame feature extraction and the use of Long Short Term Memory (LSTM) neural architectures for temporal sequence modeling. The overall architecture comprises two sub-networks, a CNN network for feature extraction and a LSTM network for modeling correlations across time steps. On the other hand, the 3D convolution architecture is based on the C3D model [27], which is able to learn directly from spatio-temporal data, trained in an end-to-end manner; hence, modeling spatial and temporal correlations at the same time.

A. 3D flow processing and representation

For computing a discriminant 3D flow representation for each video frame, while taking advantage of the DL paradigm, CNNs are employed, due to their increased ability in modeling complex patterns at multiple scales along the spatial dimension [28]. In order to fully leverage the learning and discrimination capabilities of CNNs in typical visual analysis tasks, the common practice of utilizing CNNs pre-trained using large-scale datasets and fine-tuning them using the particular data of the application at hand has also been followed in this work. However, in order to enable the use of pre-trained CNNs (*i.e.* models that receive as input RGB information) in the development of the proposed 3D flow representation, an appropriate transformation of 3D flow to RGB-like information is required.

For achieving this, a 'flow colorization' approach is followed, similar to the one introduced by Eitel *et al*. in [29]. In particular, at every pixel location and for each of the X, Y, Z axes, the absolute value of the corresponding flow vector component is linearly normalized in the interval $[0, 255]$, taking into account the respective minimum and maximum values that have been measured in the whole dataset and for all pixel positions. In this way, a colorized 3-channel RGB image is estimated for every frame 3D flow field, which can in turn be provided as input to any conventional CNN that has been trained using RGB data. In Fig. 2, examples of colorized flow fields of consecutive frames for different types of actions are given (green, red and blue colors are used for X, Y, and Z axes respectively). For example, for the 'Pushing other person' action the green color indicates that there is an intense motion in both arms towards the X direction, which is the case when someone pushes another person away. On the other hand, the red color is predominant in the 'Jump' action, where the person moves along the Y direction. Following the proposed flow colorization approach, 3D flow patterns of increased complexity are estimated along the layers of the originally RGB trained CNN.

B. Sequence modeling approach (CNN-LSTM)

Regarding the CNN-LSTM architecture, the proposed flow representation is eventually computed by considering the features from the last CNN convolutional layer. In the current implementation, the 'ResNet-34' [30] model is used and the last convolutional layer is fed to the LSTM in order to perform the classification. LSTM networks [31], a particular type of RNNs, have been extensively used, among others, for human action recognition, due to their efficiency in modeling time evolving processes [32], [33]. The latter aims at modeling the temporal succession of its internal states representation for best explaining the input data; hence, encoding the action dynamics. To this end, LSTMs have also been used in this work for realizing human action recognition. For every action instance, a constant number of T frames is uniformly sampled. For each frame, flow features are extracted from the CNN network as described in Section II-A, and features from the last convolutional layer subsequently are provided as input to the LSTM model. In the current work, the individual NN architectures are selected to be independently trained for simplicity purposes; thus, during training the CNN model parameters are not updated, while the LSTM weights are adapted following the back-propagation operation.

In order to examine the fundamental functionality of a LSTM in depth, let $\mathbf{X(t)}$ be an input sequence and $\mathbf{P(t)}$ the corresponding target output. A LSTM then maps $\mathbf{X(t)}$ to $\mathbf{P(t)}$ through a series of intermediate representations [31]:

$$\mathbf{I(t)} = \sigma[\mathbf{W_{xi}X(t)} + \mathbf{W_{hi}H(t-1)} + \mathbf{B_i}] \quad (1)$$

$$\mathbf{F(t)} = \sigma[\mathbf{W_{xf}X(t)} + \mathbf{W_{hf}H(t-1)} + \mathbf{B_f}] \quad (2)$$

$$\mathbf{O(t)} = \sigma[\mathbf{W_{xo}X(t)} + \mathbf{W_{ho}H(t-1)} + \mathbf{B_o}] \quad (3)$$

$$\mathbf{G(t)} = \tanh[\mathbf{W_{xc}X(t)} + \mathbf{W_{hc}H(t-1)} + \mathbf{B_c}] \quad (4)$$

$$\mathbf{C(t)} = \mathbf{F(t)C(t-1)} + \mathbf{I(t)G(t)} \quad (5)$$

$$\mathbf{H(t)} = \mathbf{O(t)}\tanh[\mathbf{C(t)}] \quad (6)$$

$$\mathbf{P(t)} = \mathbf{W_{hp}H(t)} + \mathbf{B_p} \quad (7)$$

$\sigma(.)$ is a non-linear scaling factor. $\mathbf{C(t)}$ is the 'internal memory' of the LSTM and the gates $\mathbf{I(t)}$, $\mathbf{F(t)}$ and $\mathbf{O(t)}$ control the degree to which the memory accumulates new input $\mathbf{G(t)}$, attenuates its memory and influences the hidden layer output $\mathbf{H(t)}$, respectively. The LSTM is parametrized by the learnable weight $\mathbf{W}$ and biases $\mathbf{B}$ matrices. These weights are used to direct the operation of the gates and they depend solely on the current and the previous time step. From the above equations (1)-(4), it can be observed that a LSTM

unit encodes the temporal patterns between two consecutive frames, namely between frames t and $t-1$. Additionally, the LSTM network has relative few parameters to update during the learning phase, since the input $\mathbf{W_x}$ and the recurrent $\mathbf{W_h}$ weight matrices are shared across time. Moreover, the developed LSTM network is trained to predict the observed action class at every video frame, while for estimating an aggregated probability $\mathbf{P(t)}$ for each action for the entire video sequence, simple averaging of all corresponding probability values of all frames is performed. Multi-layer LSTMs are used in this work for efficiently encoding more long-term correlations in the input data.

C. Template matching approach (3D convolution)

Compared to 2D convolutions, 3D convolutions have the ability to model temporal dependencies and correlations, due to the employed 3D filters and pooling operators. As stated in [27], 3D convolutions are applied to both the spatial and temporal domain, while 2D convolutions operate only along the spatial dimensions. It has been proven that 2D convolutions do not fully exploit the temporal information of the input signal. On the other hand, 3D convolutions are shown to be advantageous in modeling also temporal, apart from only spatial, characteristics [27], [34]. In addition, the 3D pooling operation further reduces the size of the input data, while preserving the encoded motion patterns and removing irrelevant information. For the 3D convolution, both feature maps and kernels have a temporal dimension, and the convolution also needs to slide along that direction. The shape of the kernel is $d \times h \times w$ where d, represents the kernel temporal depth, h and w are the height and width, respectively. Compared with the LSTM described in the Section II-B, the receptive field of a 3D convolution layer takes into account a variable number of adjacent frames; thus, learning wider(*i.e.* across the whole action) temporal patterns.

For modeling the correlations among the sequentially selected 2D colorized motion frames (Section II-A), the 3D ConvNet (C3D) network [27] is employed to learn the chromatic changes, edge orientations and, hence, the encoded motion patterns. The C3D consists of 8 convolutional layers with $3 \times 3 \times 3$ kernels that operate spatio-temporally on the RGB sequence, interleaved by 5 max-pooling layers. At the top of the model, two fully connected layers of 4096 neurons each are followed by a softmax layer, the latter with neurons equal to the number of the supported action classes.

III. Experimental results

In this section, experimental results, as well as comparative evaluation, from the application of the proposed 3D action recognition methods are presented. For the evaluation, the 'NTU RGB+D' [4] dataset was used, *i.e.* the currently broadest and by far most challenging publicly available one (a set of 60 action types are supported). Taking into account the videos' duration in the dataset, a set of 60 frames were uniformly selected for feature extraction, which roughly corresponds to one third of the average number of frames per action. Prior to feature extraction, simple depth thresholding techniques based on skeleton tracking information were used for maintaining only the subjects' silhouettes.

A. CNN-LSTM implementation details

Concerning the CNN-LSTM architecture, for motion feature extraction, the following procedure was applied at every frame: a) 3D flow fields were extracted at the Kinects's depth resolution (512×424), b) a square (400×400) region positioned at the frame center was selected, c) the aforementioned region was down-sampled to size 300×300 pixels, and d) a (224x224) patch was randomly cropped; it needs to be noted that the employed CNN input dimension was equal to 224×224. Therefore the CNN-LSTM receives input clips of size $3 \times 60 \times 224 \times 224$.

Regarding implementation details, the proposed LSTM network consisted of three layers with 2048 units, while the 'Torch[1]' scientific computing framework and a Nvidia Tesla K40 GPU were used. Zero-mean Gaussian distribution with standard deviation equal to 0.01 was used to initialize all NN weight and bias matrices. All class predictions were passed through a softmax operator (layer) to estimate a probability distribution over the supported actions. Stochastic Gradient Descent (SGD) was used during training, along with a multinomial logistic loss function. The batch size was set equal to 256, while the momentum value was equal to 0.9. Weight decay with value 0.0005 was used for regularization. For single-modality analysis the training procedure lasted 80 epochs. An adaptive learning rate approach [35] was followed during training, which proved to speed up the learning process while solving the problem of exploding gradients.

B. C3D implementation details

Unlike the previous spatio-temporal architecture, the 3D ConvNet architecture required a different pre-processing approach, mainly due to the computational complexity of the C3D model. Following the same procedure, the extracted features were center-cropped (400×400), down-sampled (300×300) and randomly cropped (224×224) to induce spatial and temporal jittering. During training, clips were down-scaled to 112×112 to match the C3D input requirements. Each video sequence was split into 16-frame clips with 8 frames overlap, resulting in a slightly higher performance and reducing the computational burden. Therefore, the C3D receives input clips of size $3 \times 16 \times 112 \times 112$.

Regarding the C3D model, the 'Keras[2]' deep learning framework with 'Tensorflow[3]' backend was used for experimentation on two Nvidia Tesla K40 GPUs. The model was trained with SGD, using mini-batches of 60 clips with $lr = 3e^{-3}$. The lr was divided by 5 every 4 epochs and the model required 30 epochs to converge.

	Method	Accuracy	
		Cross-subject	Cross-view
a)	1 Layer LSTM (256)	49.28%	50.15%
	1 Layer LSTM (512)	51.23%	53.48%
	1 Layer LSTM (1024)	53.49%	56.85%
	1 Layer LSTM (2048)	56.54%	58.91%
	2 Layer LSTM (2048)	58.62%	60.43%
	3 Layer LSTM (2048)	59.85%	61.83%
b)	Proposed Conv3D-Flow	73.27%	79.64%
c)	Local flow [15]*	34.33%	37.42%
	Global flow [15]*	48.09%	52.44%
	SFAM [25]	57.36%	59.14%
	LieNet [23]	61.37%	66.95%
	P-LSTM [4]	62.9%	70.3%
	ST-LSTM [22]	69.2%	77.7%
	GCA-LSTM [24]	**74.4%**	**82.8%**

TABLE I: Action recognition results: a) LSTM scheme parameterization, b) C3D-based approach and c) Comparative evaluation. Methods with an asterisk '*' follow the hand-crafted approach, while underlined ones indicate skeleton-based schemes.

C. Evaluation

In Table I, quantitative action recognition results are given in the form of the overall classification accuracy, *i.e*. the percentage of all action instances that were correctly classified. Concerning the proposed two-step spatio-temporal scheme, a set of variant NN architectures are evaluated (group 'a' of experiments in Table I). It can be observed that the introduced 3-Layer CNN-LSTM approach exhibits the highest overall performance. This is mainly due to the more complex feature representations that the developed LSTM encodes in deeper layers. Examining the behavior of the CNN-LSTM scheme in more details, it is shown that the performance is maximized by increasing the number of hidden units as well as the number of stacked LSTM layers. Further analysis of the findings suggests that the depth of the network, in terms of additional layers, is more important than the number of memory cells in a given layer to model an action. Although these approaches, which combine the advantageous characteristics for spatial correlation estimation of CNNS with the increased temporal modeling capabilities of LSTMs, show promising performance in the task of 3D human action recognition, they do not employ an end-to-end model and require separate computation of the 3D flow representation and the estimation of temporal dependencies of adjacent frames. In addition, the LSTM network may be often susceptible in the presence of noise in the input signal, as it only models correlations between the two consecutive frames t and $t-1$ (1)-(4). On the other hand, the proposed C3D approach achieves remarkably improved performance over the CNN-LSTM architectures (group 'b' of results in Table I), due to the increased receptive field and the 'shared' spatio-temporal weight matrices (as reported in Section II-C), highlighting the strength of template matching approaches in modeling spatio-temporal patterns.

For providing a better insight, the action recognition confusion matrix obtained from the proposed best performing scheme (C3D-based) is given in Fig. 3. It can be observed that the proposed network boosts the performance of actions that contain whole-body motions, such as 'standing up', 'sitting down', 'throw', 'pickup', 'wear jacket', 'jump up', *etc*. Additionally, actions with similar poses or subtle motions are shown to be hard to distinct. For example, the 'writing' action is misclassified as 'reading' or 'playing with phone/tablet'. It needs also to be highlighted that the proposed action representation does not make use of domain specific knowledge (*i.e*. the same flow analysis methodology can be applied with any other type of objects being present in the examined scene, *e.g*. chair). The latter demonstrates the increased discrimination capabilities of the proposed flow representation scheme.

The proposed methods are comparatively evaluated with a set of approaches that make use of 3D flow or skeleton information. In particular, the performance of both hand-crafted (local flow [15] and global flow [15]) and DL (SFAM [25], LieNet [23], P-LSTM [4], ST-LSTM [22] and GCA-LSTM [24]) methods is reported (group 'c' of experiments in Table I). The recognition performance of literature approaches is indicated as reported in [15] and [24]. Only the method of SFAM [25] was implemented by the authors of this work. Overall, from the presented results (Table I), it can be seen that the proposed method exhibits improved performance, compared to the other 3D flow methods. In particular, the introduced CNN-LSTM method surpasses the SFAM approach (*i.e*. the best-performing literature work) by at least 2.5%, while the C3D approach by at least 16% in both the 'CrossSubject' and 'CrossView' setups. This justifies the fundamental claim of the current work that for achieving robust action recognition results, design of truly spatio-temporal schemes is required. Concerning the comparison with the skeleton-based approaches, the proposed best performing method (namely C3D) surpasses most well-known literature methods, demonstrating the rich and discriminative properties of the 3D flow modality. Only the method of GCA-LSTM [24] exhibits increased performance; however, the latter technique incorporates a recurrent attention mechanism for performance improvement. On the contrary, the focus of this work was to propose new means for exploiting the 3D flow information more efficiently, while such attention mechanisms could also be additionally incorporated for further performance improvement. Moreover, it needs to be highlighted again that the proposed flow representation exhibits a significant advantageous characteristic, since it includes rich and fine-grained information, regarding the motion dynamics of the observed human actions, while however retaining generality, *i.e*. it can support any type of object, does not incorporate domain specific information (*e.g*. output of a skeleton-tracker) and it is able to generalize across different evaluation scenarios.

[1]http://torch.ch/
[2]https://keras.io/
[3]https://www.tensorflow.org/

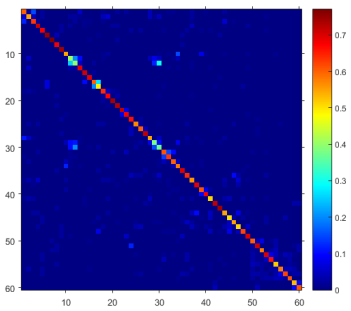


Fig. 3: The confusion matrix obtained from the application of the proposed C3D-based scheme.

IV. Conclusions

In this paper, the problem of 3D flow human action recognition using DL techniques was investigated. In particular, independent CNN and LSTM architectures (comprising the proposed composite one) were selected for simplicity purposes. Complementary, a template matching approach was presented that learns spatio-temporal features from videos, by applying 3D convolutions; thus, modeling the action dynamics within consecutive frames. Moreover, a new processing and representation scheme that utilizes 3D flow information was introduced, while further exploits the learning and discrimination capabilities of DL for human action recognition. The proposed methods were experimentally shown to outperform similar DL and hand-crafted 3D flow approaches, and compare favorably with most skeleton-based techniques in the currently most challenging public dataset, without using domain-specific information (Section III-C). Future work includes the investigation of including domain-specific knowledge in the introduced flow representation and its more efficient combination with skeleton-tracking data.

Acknowledgment

The work presented in this paper was supported by the European Commission under contract H2020-700367 DANTE.

References

[1] G. Cheng, Y. Wan, A. N. Saudagar, K. Namuduri, and B. P. Buckles, "Advances in human action recognition: A survey," *CoRR*, vol. abs/1501.05964, 2015.

[2] S. Herath, M. Tafazzoli Harandi, and F. Porikli, "Going deeper into action recognition: A survey," *CoRR*, vol. abs/1605.04988, 2016.

[3] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *NIPS*. IEEE, 2014, p. 568576.

[4] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "NTU RGB+D: A large scale dataset for 3d human activity analysis," in *CVPR*, 2016.

[5] O. Oreifej and Z. Liu, "HON4D: Histogram of oriented 4D normals for activity recognition from depth sequences," in *CVPR*, 2013.

[6] X. Yang and Y. Tian, "Super normal vector for human activity recognition with depth cameras," *IEEE TPAMI*, vol. PP, no. 99, pp. 1–1, 2016.

[7] H. Rahmani, A. Mahmood, D. Q. Huynh, and A. S. Mian, "Histogram of oriented principal components for cross-view action recognition," *IEEE TPAMI*, vol. 38, no. 12, pp. 2430–2443, 2016.

[8] M. E. Hussein, M. Torki, M. A. Gowayyed, and M. El-Saban, "Human action recognition using a temporal hierarchy of covariance descriptors on 3D joint locations," in *IJCAI*, 2013.

[9] Y. Wu, "Mining actionlet ensemble for action recognition with depth cameras," in *CVPR*, 2012.

[10] L. Xia, C.-C. Chen, and J. K. Aggarwal, "View invariant human action recognition using histograms of 3D joints," in *CVPRW*, 2012, pp. 20–27.

[11] M. Jiang, J. Kong, G. Bebis, and H. Huo, "Informative joints based human action recognition using skeleton contexts," *Signal Processing: Image Communication*, vol. 33, pp. 29–40, 2015.

[12] S. Michieletto M. Munaro, G. Ballin and E. Menegatti, "3d flow estimation for human action recognition from colored point clouds," *BICA*, 2013.

[13] M. B. Holte, B. Chakraborty, J. Gonzalez, and T. B. Moeslund, "A local 3-D motion descriptor for multi-view human action recognition from 4-D spatio-temporal interest points," *J-STSP*, vol. 6, no. 5, pp. 553–565, Sept 2012.

[14] S. R. Fanello, I. Gori, G. Metta, and F. Odone, "Keep it simple and sparse: real-time action recognition.," *JMLR*, vol. 14, no. 1, pp. 2617–2640, 2013.

[15] G. Papadopoulos and P. Daras, "Human action recognition using 3d reconstruction data," *IEEE TCSVT*, vol. PP, no. 99, pp. 1–1, 2017.

[16] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov, "Scalable object detection using deep neural networks," in *CVPR*, 2014.

[17] W. Ouyang, X. Wang, X. Zeng, S. Qiu, P. Luo, Y. Tian, H. Li, S. Yang, Z. Wang, C.-C. Loy, et al., "Deepid-net: Deformable deep convolutional neural networks for object detection," in *CVPR*, 2015.

[18] C. Gan, N. Wang, Y. Yang, D.-Y. Yeung, and A. G. Hauptmann, "DevNet: A deep event network for multimedia event detection and evidence recounting," in *CVPR*, 2015.

[19] H. Fang, S. Gupta, F. N. Iandola, R. K. Srivastava, L. Deng, P. Dollár, J. Gao, X. He, M. Mitchell, J. C. Platt, C. L. Zitnick, and G. Zweig, "From captions to visual concepts and back," *CoRR*, vol. abs/1411.4952, 2014.

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE TPAMI*, vol. 37, no. 9, pp. 1904–1916, 2015.

[21] K. Simonya and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *ICLR*, 2015.

[22] Jun Liu, Amir Shahroudy, Dong Xu, and Gang Wang, "Spatio-temporal LSTM with trust gates for 3D human action recognition," in *ECCV*. Springer, 2016, pp. 816–833.

[23] Z. Huang, C. Wan, Probst T, and L. Van Gool, "Deep learning on lie groups for skeleton-based action recognition," *CoRR*, vol. abs/1612.05877, 2016.

[24] J. Liu, G. Wang, P. Hu, L. Y. Duan, and A. C. Kot, "Global context-aware attention lstm networks for 3d action recognition," in *CVPR*, July 2017, pp. 3671–3680.

[25] P. Wang, W. Li, Z. Gao, Y. Zhang, C. Tang, and P. Ogunbona, "Scene flow to action map: A new representation for rgb-d based action recognition with convolutional neural networks," *CVPR*, 2017.

[26] M. Jaimez, M. Souiai, J. Gonzalez-Jimenez, and D. Cremers, "A primal-dual framework for real-time dense RGB-D scene flow," in *ICRA*, 2015.

[27] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *ICCV*, 2015, pp. 4489–4497.

[28] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012, pp. 1097–1105.

[29] A. Eitel, J. T. Springenberg, L. Spinello, M. Riedmiller, and W. Burgard, "Multimodal deep learning for robust rgb-d object recognition," in *IROS*, 2015.

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[31] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[32] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *CVPR*, 2015.

[33] S. Yeung, O. Russakovsky, N. Jin, M. Andriluka, G. Mori, and L. Fei-Fei, "Every moment counts: Dense detailed labeling of actions in complex videos," *IJCV*, 2017.

[34] G. Varol, I. Laptev, and C. Schmid, "Long-term temporal convolutions for action recognition," *IEEE TPAMI*, vol. 40, no. 6, pp. 1510–1517, 2018.

[35] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *CVPR*, 2014.