Visual Rhythm-based Convolutional Neural Networks and Adaptive Fusion for a Multi-stream Architecture Applied to Human Action Recognition

Helena de Almeida Maia*[§], Marcelo Bernardes Vieira[†] and Helio Pedrini*
*Institute of Computing, University of Campinas (UNICAMP)
Campinas - SP - Brazil
Email: {helena.maia,helio}@ic.unicamp.br
[†]Department of Computer Science, Federal University of Juiz de Fora (UFJF)
Juiz de Fora - MG - Brazil
Email: marcelo.bernardes@ufjf.edu.br

***Abstract*—In this work, we address the problem of human action recognition in videos. We propose and analyze a multi-stream architecture containing image-based networks pre-trained on the large ImageNet. Different image representations are extracted from the videos to feed the streams, in order to provide complementary information for the system. Here, we propose new streams based on visual rhythm that encodes longer-term information when compared to still frames and optical flow. Our main contribution is a stream based on a new variant of the visual rhythm called Learnable Visual Rhythm (LVR) formed by the outputs of a deep network. The features are collected at multiple depths to enable the analysis of different abstraction levels. This strategy significantly outperforms the handcrafted version on the UCF101 and HMDB51 datasets. We also investigate many combinations of the streams to identify the modalities that better complement each other. Experiments conducted on the two datasets show that our multi-stream network achieved competitive results compared to state-of-the-art approaches.**

I. Introduction

Over the past few years, a large amount of video data has been produced and released due to the easy access to both devices for capturing new data such as video cameras and mobiles, and streaming platforms such as YouTube for sharing. Since the analysis of this large amount of data by human operators may be stressful and may involve sensitive content, automatic procedures are needed to address related problems.

The problem addressed in this work is the Human Action Recognition (HAR) in videos that aims to classify the action being performed by one or more actors. The understanding of human activity has several relevant applications, such as intelligent surveillance [1], [2], human-computer interaction [3], [4] and smart home security [5]–[8]. Similar to other video-based problems, HAR faces some challenges related to difficult scene conditions (e.g., occlusions and lighting changes) which affect how the actions are seen in the video. It also presents specific challenges, for example, the similarity among different classes (e.g., walking and running) and the various ways of performing the same action.

Deep networks have been widely explored for HAR. However, the high cost of video-based networks and the absence of datasets as large as image-based ones have led the researchers to explore image networks for the problem. Following this trend, we propose an image-based network inspired by the two-stream architecture [9], which is an important method that gave rise to a variety of state-of-the-art approaches [10]–[12]. This architecture has two parallel networks working with different image modalities: RGB which represent static appearance and optical flow that encodes short-term motion. Its central idea is to explore the strengths of these modalities by combining their respective stream outputs. Our main objective is to provide complementary information for the streams in order to capture new aspects of the actions.

Our first contribution is a **multi-stream framework based on visual rhythm (VR)**. VR consists of a compact 2D representation of the video constructed by the concatenation of frame-level features (also called “slices”). In contrast to RGB and optical flow, it represents the entire video in a single image, being a longer-term modality. The second contribution is a **novel method to construct VRs** called Learnable Visual Rhythm (LVR). LVR is based on Convolutional Neural Networks (CNNs) for feature extraction and so, it is able to capture complex patterns in the frames, achieving superior results than handcrafted VRs. In our research, we also investigated three **adaptive fusion** methods with trainable parameters to exploit the strengths of each modality in different scenarios, and carried out an **extensive analysis** of individual and combined performance of the streams. However, the adaptive fusion and part of the extensive analysis are not covered by this text due to the page limit. We suggest the reading of Chapter 7 of the Ph.D. thesis [13] for a detailed study.

II. Proposed Method

Our basic method consists of a three-stream architecture composed of the spatial, temporal and VR-based streams (Figure 1). Each stream consists of an image-based CNN. The first two are based on the two-stream network [9], [15]. The spatial

[§]This work relates to a Ph.D. thesis.

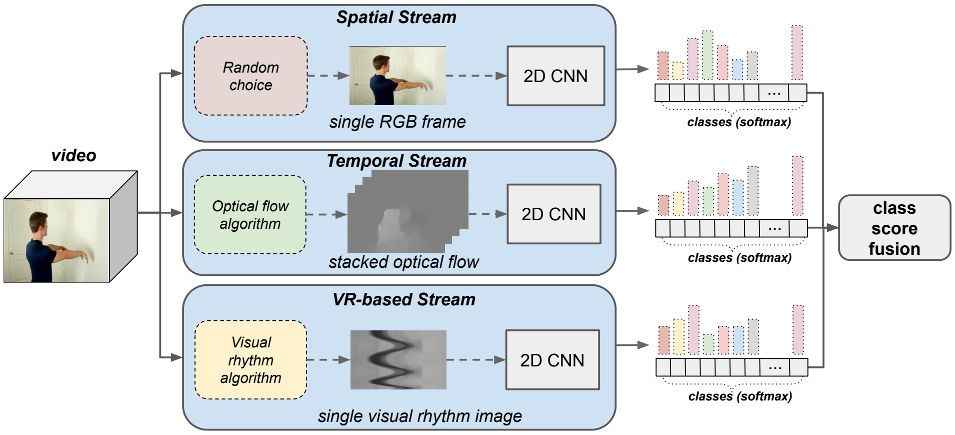


Fig. 1. Overview of our three-stream basic method for action recognition. Adapted from Concha *et al*. [14].

stream performs object recognition using a single RGB frame randomly selected from the video. Thus, it captures elements that compose the appearance of the action, for instance, the usual scenario, objects involved, among others. For instance, recognizing a guitar may help to recognize a video from the PlayingGuitar class. However, a green grass field or even the equipment may not be sufficient to distinguish between a CricketBowling and a CricketShot video. Therefore, some dynamics are needed to complement the spatial information. This is carried out by the temporal stream that receives 10 pairs of consecutive optical images in the form of a 20-channel image representing the motion information. The 20-channel image is referred to as a stack of optical flow images.

The third stream is the main contribution of our work and is based on a long-term feature, the visual rhythm. We proposed different approaches to it detailed in Subsections II-A and II-B. Each stream is individually trained and their m-dimensional score vectors are fused during the test stage using a weighted average with fixed weights, where m is the number of classes.

A. Adaptive Visual Rhythm

Our first strategy for the VR-based stream is called Adaptive Visual Rhythm (AVR). For the AVR, the rhythms are handcrafted images based on the operations proposed by Souza *et al*. [16]. These operations produce two types of slices per frame defined as the average of the columns/rows intensities (Figure 2). The slices of a fixed direction are concatenated to form the horizontal- and vertical-mean VRs.

We propose a method for adaptively deciding the best VR direction for each action according to the predominant movement. It is based on the following observation. Consider, without loss of generality, horizontal-mean slices. If an object moves orthogonally to the slice direction, it is very likely that the mean color of the corresponding column remains the same (Figure 3a). However, a horizontal movement affects the average color of all columns spanned by the object (Figure 3b). Therefore, **movements parallel to the slice direction tend to produce more distinctive patterns**. For estimating the predominant direction of movement, we use the Lucas-Kanade point-tracker. The absolute horizontal and vertical displacement estimated by the tracker are accumulated over the frames, and the highest value defines the rhythm direction.

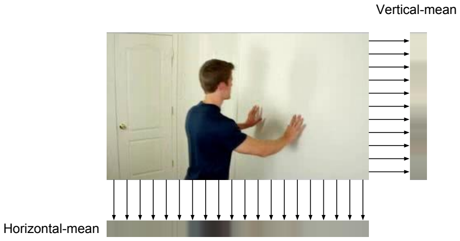
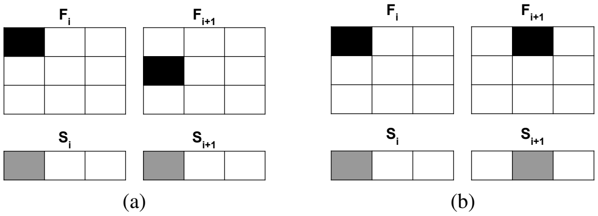
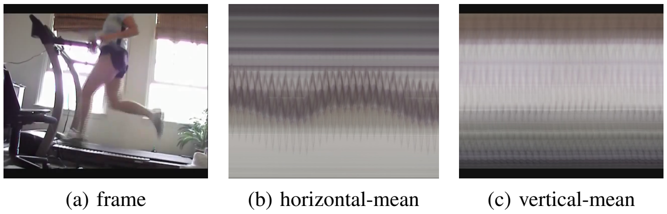


Fig. 2. Examples of horizontal- and vertical-mean slices extracted from Concha *et al*. [14]. The slices are defined as the average of columns/rows. They were resized for illustration purposes.



(a) (b)

Fig. 3. Moving objects considering two consecutive frames and horizontal-mean slices. Parallel movement is better captured in the slice. Extracted from Concha *et al*. [14].



(a) frame (b) horizontal-mean (c) vertical-mean

Fig. 4. Example of frame and rhythms from a Kinetics video of the “running on treadmill” class. The horizontal-mean rhythm presents a wavy pattern that better characterizes the action. Extracted from Maia *et al*. [17].

Figure 4 shows a frame and the visual rhythms extracted from a Kinetics [18] video of the “running on treadmill” class. This action is predominantly horizontal due to the leg motion. For this reason, the horizontal rhythm presents more relevant patterns for the classification. As can be observed in the example, the horizontal rhythm contains a wavy pattern that represents the leg movements, whereas the vertical one is composed of quite homogeneous lines.

B. Learnable Visual Rhythm

Image-based networks have achieved great results for image classification, describing objects and appearance. For this reason, they are useful to build frame-level descriptors. In this approach, we use a 2D CNN as the operation to produce VR slices. This stream is called Learnable Visual Rhythm (LVR), thanks to its trainable operation. The LVR is also composed of a second CNN that predicts the action from the produced VR. Each CNN is an Inception V3 network pre-trained on ImageNet, and only the second is fine-tuned on each video dataset. Figure 5 illustrates the LVR stream.

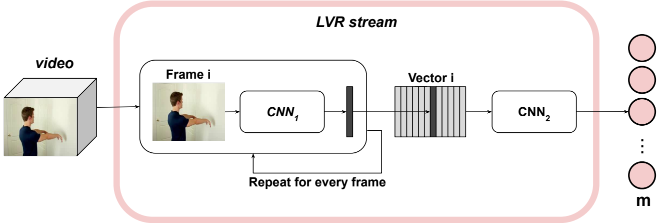


Fig. 5. LVR stream composed of two stacked CNNs. The first one computes an 1D descriptor for each video frame. The second one predicts the action based on the 2D concatenation of the descriptors. The final LVR output is an m-dimensional score vector. Adapted from Maia *et al.* [19].

We consider three distinct points of the Inception to extract slices, called LVR_0, LVR_1 and LVR_2 (Figure 6). This enables the analysis of different abstraction levels, from the lowest LVR_0 to the highest LVR_2. Average pooling layers and reshape method are used to reduce the size of the intermediate images and collapse them into one dimension. The connections between the original layers are maintained, therefore, the extra pooling layers do not affect the results of the following ones. The resulting feature vectors are normalized using min-max normalization, which helps mapping them into a grayscale image. For matching the second CNN input dimension, we apply an adaptive average pooling along the vertical axis and a resize method along the horizontal one. This is done due to the fact that the rhythm height is much greater than its width (about 10 times greater).

The rationale for using intermediate outputs is that networks trained for image classification tend to be invariant to the object position and poses at the latter layers, but this information is rather relevant to distinguish actions. The former outputs, on the other hand, represent less refined information. Since the three outputs can be computed in a single forward propagation through the network, they can be combined at a minimum extra computational cost for the extractions. A detailed scheme of the extraction process is given in Figure 6.3 of the thesis [13].

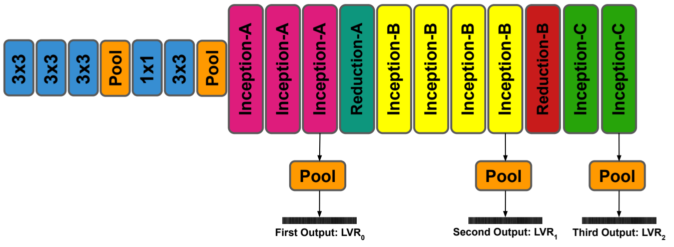


Fig. 6. An illustration of the three distinct positions considered to extract the features in the Inception architecture. Each position represents a different level of abstraction. These features compose the LVR_0, LVR_1 and LVR_2 rhythms.

The main purpose of the second network is to find temporal patterns in the rhythm images. As the first CNN, it is initialized with ImageNet weights, but fine-tuned on the video dataset. The training and testing processes consider a single abstraction level for the inputs at a time, that is, LVR_0, LVR_1 and LVR_2 are trained/tested separately.

III. Results

Table I shows the stream results on UCF101 [20] and HMDB51 [21] and includes different VR-based approaches. In the table, RGB* stands for spatial stream, whereas OF is the temporal one. The VR-based approaches are identified by their respective acronyms. We can see that OF outperforms the others on both datasets. Among the VR approaches, LVR_1 presents the best accuracies, considerably exceeding the AVR, which suggests that deep features may be more representative than handcrafted ones. These results also indicate the superiority of the intermediate depth in providing good descriptors for action recognition. Since the Inception is originally trained for the object recognition problem, latter layers may produce features more invariant to poses and positions. Hence, the classification CNN that works with LVR_2 images might face difficulties in capturing the temporal evolution and distinguishing the actions. LVR_0, on the other hand, may lack information about the scene structure. This last observation is reinforced by the fact that the handcrafted AVR outperforms LVR_0. Moreover, compared to the other depths, this image undergoes a significant reduction in size to feed the CNN.

TABLE I
Individual Results. Cells on bold represent the highest accuracies (%).

Modality	UCF101	HMDB51
RGB*	86.61 ± 0.09	51.77 ± 2.15
OF	**86.95 ± 0.64**	**59.91 ± 0.43**
AVR	64.74 ± 0.63	39.63 ± 0.60
LVR_0	63.64 ± 0.57	35.06 ± 1.34
LVR_1	81.26 ± 0.30	51.94 ± 1.01
LVR_2	78.75 ± 0.57	45.53 ± 1.53

We have tested many combinations of the streams using the weighted average to find the best setting for our multi-stream network. The results are presented in Tables 6.3, 6.4 and 6.5 of the thesis. For the combinations, we assigned weights 2, 3 and 1 for the spatial, temporal and any VR approach, respectively. The experiments show that all combinations outperform individual versions, suggesting that the streams complement each other in some levels, even combinations of VR-based approaches. Most combinations using only the temporal stream outperform those using only the spatial, however, both together surpass the others. We also noted that the AVR tends to better complement the spatial stream, whereas the LVR achieves higher scores combined with the temporal one. A possible explanation is that the feature computed in the spatial network is already embedded into the LVR images, since the feature-extractor CNNs are very similar to the spatial stream, and so the LVR were not able to contribute much to it. Finally, the contribution of the LVR_0 is usually lower, in line with individual results.

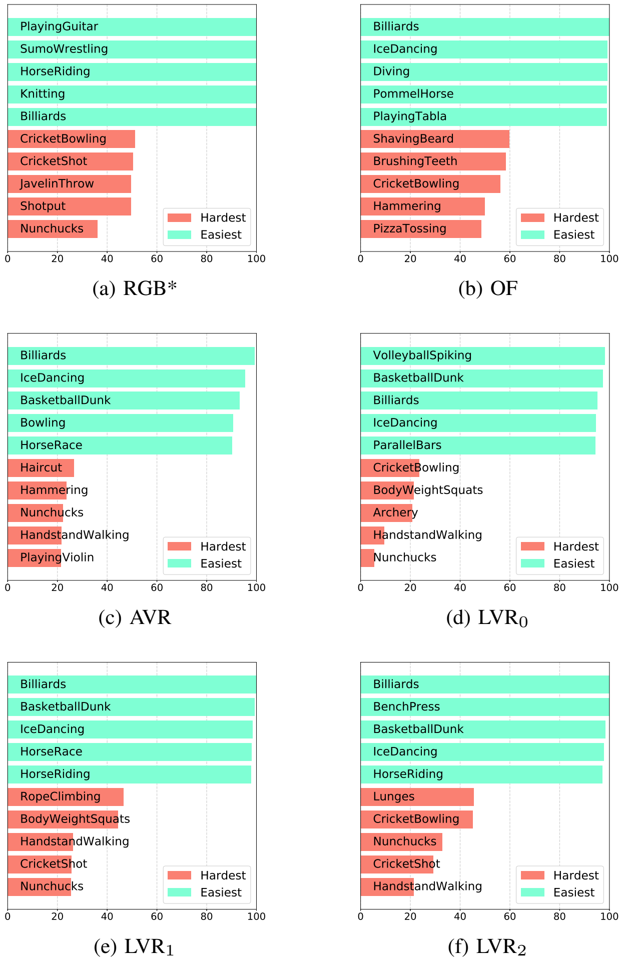
From these combinations, we selected the five-stream RGB* + OF + AVR + LVR_0 + LVR_1 which presented the highest accuracy on HMDB51 and results very similar to the best combination on UCF101 (a difference of only 0.06%). The four-stream RGB* + OF + AVR + LVR_1 also presented satisfactory results at a lower computational cost, since it contains fewer streams. In Table II, which is a short version of Table 6.6 from the thesis, we compare these combinations with some state-of-the-art approaches. Table 6.6 also shows other methods [10], [11], [22]–[25] that achieved higher scores by pre-training the network on a considerable larger dataset, the Kinetics [18]. Here, we show only the methods with protocol similar to ours. We can see that the five-stream method achieved the 6th best accuracy on UCF101 and the 5th best on HMDB51.

TABLE II
COMPARISON OF ACCURACY RATES (%) ON UCF101 AND HMDB51 DATASETS. CELLS ON BOLD REPRESENTS THE HIGHEST ACCURACIES.

Method	UCF101	HMDB51
Two-stream + SVM [9]	88.0	59.4
Two-stream + LSTM [26]	88.6	—
TDD + iDT [27]	91.5	65.9
KVMDF [28]	93.1	63.3
Two-stream fusion + iDT [29]	93.5	69.2
Two-stream TSN [30]	94.0	68.5
Three-stream TSN [30]	94.2	69.4
Recurrent hybrid network [31]	93.2	71.8
L^2STM [32]	93.6	66.2
Three-stream [33]	94.1	70.4
STP [34]	94.6	68.9
TLE [12]	95.6	71.1
Two-stream LTC + iDT [35]	92.7	67.2
Gated TSN [36]	94.5	—
Four-stream + iDT [37]	96.0	**74.9**
Heterogeneous two-stream [38]	94.4	67.2
Two-stream Choquet [39]	92.9	65.9
TEA [40]	**96.9**	73.3
LVR (four-stream)	94.3	70.7
LVR (five-stream)	94.4	71.0

IV. ANALYSIS

In this section, we analyze the behavior of the streams inspired by the analysis of the Kinetics paper [18]. Figure 7 shows the 5 easiest and 5 hardest classes for each modality on UCF101. To generate these lists, we considered the class recall obtained during test and averaged over the splits.

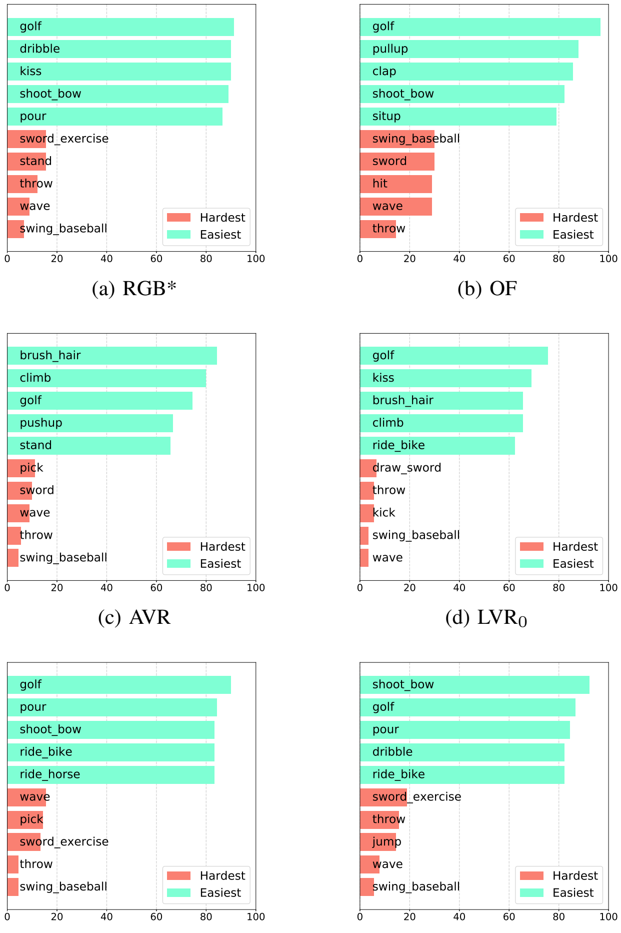


(a) RGB*
(b) OF
(c) AVR
(d) LVR_0
(e) LVR_1
(f) LVR_2

Fig. 7. List of the 5 easiest (highest scores) and 5 hardest (smallest scores) classes for each modality on UCF101. We use the stream recall per class (horizontal axis) averaged over all splits.

Only the “Billiards” class appears in all lists as an easy class. None of the classes are in the easiest list of one stream and the hardest list of another one. Concerning the RGB* stream, its easiest classes generally involve specific objects such as “PlayingGuitar” and “HorseRiding”. On the other hand, its hardest classes present common scenarios (“CricketShot” and “CricketBowling”, “JavelinThrow” and “Shotput”) that may have affected the appearance recognition. “CricketBowling” and “Nunchucks” reached lower scores in most streams. The majority of the easiest classes of the OF list present large and characteristic movements involving the entire body (for instance, “IceDancing”), in contrast with the hardest ones that contain more subtle motions (for instance, “BrushingTeeth”).

In addition to “Billiards”, other two classes are present in the easiest list of every VR-based modality: “BasketBallDunk” and “IceDancing”. “IceDancing” is also common to the OF stream, perhaps because it requires temporal information to be distinguished. A significant number of videos from the "BasketBallDunk" class have between 51 to 100 frames. It was expected that the reshape process, regarding the Inception input size, would distort the rhythm, causing a negative impact in the scores, but it was not the case. A possible explanation is that the CNN learned the distortion, since the majority of the "BasketBallDunk" clips fall in the same length interval ($[51, 100]$). which implies that many short videos were available during the training stage. However, it is not the only factor, because these streams do not achieve good results in other classes predominantly short such as "JumpingJack". Considering the hardest classes, "Nunchucks" and "HandStandWalking" are common to the four VR-based streams.



(a) RGB*

(b) OF

(c) AVR

(d) LVR$_0$

(e) LVR$_1$

(f) LVR$_2$

Fig. 8. List of the 5 easiest (highest scores) and 5 hardest (smallest scores) classes for each modality on HMDB51. We use the stream recall per class (horizontal axis) averaged over all splits.

The top 5 hardest and easiest classes on HMDB51 are shown in Figure 8. The "golf" class was effectively recognized by every stream, whereas all of them achieved low scores for the classes "swing_baseball", "throw" and "wave". Each hardest list contains exactly one class involving swords ("sword", "sword_exercise" and "draw_sword"). In fact, the pair "sword_exercise" and "draw_sword" achieved high confusion rates for every stream, which might indicate some ambiguity in these two classes. There is a significant overlap between the hardest/easiest list from the RGB*, LVR$_1$ and LVR$_2$ streams. A complementarity analysis is required to assess whether these three streams can be combined or are redundant.

Tables III and IV show the pairwise complementarity on UCF101 and HMDB51, respectively, defined as

$$Comp(c_i, c_j) = 1 - \frac{\#\text{ of common errors}}{\#\text{ of } c_i \text{ errors}}, \qquad (1)$$

where c_i and c_j represent two streams. The complementarity rates indicate the potential accuracy for the pair after the insertion of the second stream. However, two factors can influence the real combination accuracy. First, the ability of the fusion method to select the best output, which can negatively affect the final accuracy. Second, since the fusion method combines score vectors and not labels, it can find the correct prediction from incorrect ones, positively affecting the accuracy. Therefore, the complementarity measure is used for analytical purposes only. It is worth mentioning that if c_j complements c_i much more than the opposite, i.e. $Comp(c_i, c_j) \gg Comp(c_j, c_i)$, the stream c_i is not being very useful for the pair. For this reason, we use a harmonic mean of the two values for the analysis.

TABLE III
PAIRWISE COMPLEMENTARITY $Comp(c_i, c_j)$ ON UCF101 DATASET AVERAGED OVER THE SPLITS. THE STREAM IN THE ROW CORRESPONDS TO c_i AND c_j IS THE STREAM IN THE COLUMN. THE COMPLEMENTARITY IS NONCOMMUTATIVE.

	Complementarity (↑)					
	RGB*	OF	AVR	LVR$_0$	LVR$_1$	LVR$_2$
RGB*	0.000	0.631	0.376	0.272	0.355	0.365
OF	0.640	0.000	0.299	0.369	0.525	0.541
AVR	0.772	0.738	0.000	0.463	0.663	0.656
LVR$_0$	0.722	0.755	0.441	0.000	0.598	0.609
LVR$_1$	0.579	0.683	0.395	0.308	0.000	0.450
LVR$_2$	0.594	0.701	0.398	0.343	0.463	0.000

Table III shows that all streams present more than 0.272 of complementarity rate on UCF101. We can see that the pair RGB* and OF presents a balanced complementarity with harmonic mean $HM(\text{RGB*}, \text{OF}) = 0.635$. The LVR versions present higher complementarity rates combined with the OF stream rather than in the RGB* combinations, reaching $HM(\text{OF}, \text{LVR}_2) = 0.611$, whereas the AVR better complements the RGB* ($HM(\text{RGB*}, \text{AVR}) = 0.506$). The pairs involving the RGB*, LVR$_1$ and LVR$_2$ were able to complement each other with harmonic means of approximately 0.45.

On HMDB51 (Table IV), all pairs achieved at least 0.184 of complementarity rate. In contrast to UCF101, the OF stream complements the RGB* more than the other way around ($HM(\text{RGB*}, \text{OF}) = 0.358$). The RGB* contribution to OF was similar to LVR$_1$ and LVR$_2$. The LVR$_1$ complementarity rates were higher than the other VR-based to both the RGB* and OF streams. As on UCF101, the pairs containing the RGB*, LVR_1 and LVR_2 present good scores, despite their similar behavior on the easiest/hardest classes analysis. In conclusion, we can see that every pair presents a promising contribution on both datasets. Even the combinations between VR-based streams achieved good scores, although the LVR_0 shows inferior results.

TABLE IV

Pairwise complementarity $Comp(c_i, c_j)$ on HMDB51 dataset averaged over the splits. The stream in the row corresponds to c_i and c_j is the stream in the column. The complementarity is noncommutative.

	Complementarity (↑)					
	RGB*	OF	AVR	LVR_0	LVR_1	LVR_2
RGB*	0.000	0.416	0.230	0.184	0.268	0.246
OF	0.314	0.000	0.203	0.205	0.312	0.308
AVR	0.399	0.470	0.000	0.239	0.384	0.366
LVR_0	0.387	0.491	0.267	0.000	0.361	0.369
LVR_1	0.300	0.439	0.244	0.186	0.000	0.272
LVR_2	0.294	0.448	0.239	0.214	0.288	0.000

V. Conclusions

Human action recognition is a challenging and attractive problem due to the wide range of possible applications. Although much effort has been made in this research field, there is no generic methodology for solving the problem and many questions remain open. The perception of the problem itself evolves as new datasets are released.

Throughout this text, we presented our research achievements for HAR in videos. The central issue of any application involving video analysis is the definition of a proper spatiotemporal representation that describes the event of interest. We explored deep learning strategies for this task that learns complex visual patterns from data. To minimize the high training cost of video-based deep networks, we follow the trend of exploring non-trainable elements from traditional methods in image-based ones. Thus, we used handcrafted inputs that encode the input video in an image form. Our proposed architecture is based on the multi-stream architecture [9], exploring complementary image modalities. In addition to the original spatial and temporal streams, here we introduced new ones that work with visual rhythms. Visual rhythms handle different video lengths and encode long-term information.

Our first approach, the AVR, was part of a collaborative project. The corresponding stream receives the horizontal-mean or vertical-mean rhythm as input, which represents the movement of objects by means changes in intensity over time. We proposed a method to adaptively decide the best direction for each video.

The handcrafted AVR evolved into a learnable one (LVR) composed of a feature-extractor and classification CNNs. We showed that the LVR achieves higher scores in both the individual and the combined scenarios. We also showed a comparison of the proposed methods against state-of-the-art approaches. Although our method achieves competitive results compared to those pre-trained only on ImageNet, we are behind those pre-trained on both ImageNet and Kinetics.

We analyzed the stream performance regarding the datasets classes and assessed how much the streams contribute to the combinations using the pairwise complementarity rate. The VR-based streams were able to provide complementary information for the spatial and temporal streams. The results suggest that the AVR tends to better complement the spatial stream, whereas the LVR achieves higher scores combined with the temporal one. Furthermore, the VR approaches were able to complement each other as well.

VI. Publications

The following nine papers have been published since the beginning of our research:

- **VR-based approaches:** The original AVR [14] described in Subsection II-A was published in the 17th IEEE International Conference on Machine Learning and Applications (ICMLA 2018, *Qualis B1*). An extended AVR [17] was published as a chapter of the Deep Learning Applications book (DLAPP 2020), which is composed of expanded versions of ICMLA 2018 selected papers. The LVR [19] described in Subsection II-B was published in ICMLA 2019 (*Qualis B1*). A strategy exploring optical flow rhythms [41] was recently published in the Journal of Visual Communication and Image Representation (JVCI 2021, Impact Factor of 2.479 and *Qualis A2*).
- **Multi-stream improvements:** A data augmentation technique called Symmetric Extension was proposed specifically for VR inputs and published in the International Conference on Computational Science and Its Applications (ICCSA 2021, *Qualis B1*) [42] and an extended version in the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISAPP 2020, *Qualis B1*) [43]. We also proposed different fusion methods based on fuzzy integrals for the two-stream architecture. It was published in VISAPP 2020 [39] (*Qualis B1*).
- **Other related papers:** We published a survey on VR that includes different terminologies, applications (beyond HAR), and strategies. This survey was published in Neurocomputing (2020, Impact Factor of 4.438 and *Qualis A1*). A siamese architecture for tracking [44] was published in VISAPP 2020 (*Qualis B1*).

In addition, an extension of the VISAPP work [39] about fusion methods [45], a paper related to adversarial attacks on the temporal stream [46], and a survey on video stabilization (which may support HAR) [47] have been recently submitted.

Acknowledgments

The authors would like to thank FAPESP (grant #2017/09160-1), CNPq, CAPES and FAPEMIG for the financial support, and NVIDIA for the donation of a GPU as part of the GPU Grant Program.

References

[1] W. Sultani, C. Chen, and M. Shah, "Real-world Anomaly Detection in Surveillance Videos," in *CVPR*, 2018, pp. 6479–6488.

[2] S. Ji, W. Xu, M. Yang, and K. Yu, "3D Convolutional Neural Networks for Human Action Recognition," *TPAMI*, vol. 35, no. 1, pp. 221–231, 2013.

[3] I. Gori, J. K. Aggarwal, L. Matthies, and M. S. Ryoo, "Multitype Activity Recognition in Robot-Centric Scenarios," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 593–600, Jan. 2016.

[4] M. S. Ryoo and L. Matthies, "First-Person Activity Recognition: Feature, Temporal Structure, and Prediction," *IJCV*, vol. 119, no. 3, pp. 307–328, Sep. 2016.

[5] S. M. Amiri, M. T. Pourazad, P. Nasiopoulos, and V. C. Leung, "Non-intrusive Human Activity Monitoring in a Smart Home Environment," in *International Conference on e-Health Networking, Applications and Services*. IEEE, 2013, pp. 606–610.

[6] B. Kwolek and M. Kepski, "Human Fall Detection on Embedded Platform Using Depth Maps and Wireless Accelerometer," *Computer Methods and Programs in Biomedicine*, vol. 117, no. 3, pp. 489–501, 2014.

[7] G. Leite, G. Silva, and H. Pedrini, "Fall Detection in Video Sequences Based on a Three-Stream Convolutional Neural Network," in *ICMLA*. IEEE, 2019, pp. 191–195.

[8] A. C. Sintes, "Learning to Recognize Human Actions: from Hand-crafted to Deep-learning Based Visual Representations," Ph.D. dissertation, Departament de Matemàtiques i Informàtica, Universitat de Barcelona, Barcelona, Spain, 2018.

[9] K. Simonyan and A. Zisserman, "Two-Stream Convolutional Networks for Action Recognition in Videos," in *NIPS*, 2014.

[10] J. Carreira and A. Zisserman, "Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset," in *CVPR*. IEEE, 2017.

[11] J. Zhu, Z. Zhu, and W. Zou, "End-to-End Video-level Representation Learning for Action Recognition," in *ICPR*. IEEE, 2018, pp. 645–650.

[12] A. Diba, V. Sharma, and L. Van Gool, "Deep Temporal Linear Encoding Networks," in *CVPR*, 2017.

[13] H. Maia, "Visual Rhythm-based Convolutional Neural Networks and Adaptive Fusion for a Multi-stream Architecture Applied to Human Action Recognition," Ph.D. dissertation, Institute of Computing, University of Campinas, Campinas, Brazil, 2020.

[14] D. Concha, H. Maia, H. Pedrini, H. Tacon, A. Brito, H. Chaves, and M. Vieira, "Multi-Stream Convolutional Neural Networks for Action Recognition in Video Sequences Based on Adaptive Visual Rhythms," in *ICMLA*. IEEE, 2018.

[15] L. Wang, Y. Xiong, Z. Wang, and Y. Qiao, "Towards Good Practices for very Deep Two-Stream Convnets," *arXiv preprint arXiv:1507.02159*, 2015.

[16] M. R. Souza, "Digital Video Stabilization: Algorithms and Evaluation," Master's thesis, Institute of Computing, University of Campinas, Campinas, Brazil, 2018.

[17] H. Maia, D. Concha, H. Pedrini, H. Tacon, A. Brito, H. Chaves, M. Vieira, and S. Villela, "Action Recognition in Videos Using Multi-Stream Convolutional Neural Networks," in *DLAPP*. Springer, 2020.

[18] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman, "The Kinetics Human Action Video Dataset," *arXiv preprint arXiv:1705.06950*, 2017.

[19] H. Maia, M. Souza, A. Santos, H. Pedrini, H. Tacon, A. Brito, H. Chaves, M. Vieira, and S. Villela, "Learnable Visual Rhythms Based on the Stacking of Convolutional Neural Networks for Action Recognition," in *ICMLA*. IEEE, 2019.

[20] K. Soomro, A. R. Zamir, and M. Shah, "UCF101: A Dataset of 101 Human Actions Classes from Videos in the Wild," *arXiv preprint arXiv:1212.0402*, 2012.

[21] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "HMDB: A Large Video Database for Human Motion Recognition," in *ICCV*, 2011.

[22] Y. Bo, Y. Lu, and W. He, "Few-Shot Learning of Video Action Recognition Only Based on Video Contents," in *WACV*, March 2020.

[23] V. Choutas, P. Weinzaepfel, J. Revaud, and C. Schmid, "PoTion: Pose MoTion Representation for Action Recognition," in *CVPR*, 2018.

[24] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A Closer Look at Spatiotemporal Convolutions for Action Recognition," in *CVPR*, 2018, pp. 6450–6459.

[25] J. Wang, A. Cherian, F. Porikli, and S. Gould, "Video Representation Learning Using Discriminative Pooling," in *CVPR*, 2018, pp. 1149–1158.

[26] J. Y.-H. Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond Short Snippets: Deep Networks for Video Classification," in *CVPR*, 2015, pp. 4694–4702.

[27] L. Wang, Y. Qiao, and X. Tang, "Action Recognition with Trajectory-Pooled Deep-Convolutional Descriptors," in *CVPR*, 2015, pp. 4305–4314.

[28] W. Zhu, J. Hu, G. Sun, X. Cao, and Y. Qiao, "A Key Volume Mining Deep Framework for Action Recognition," in *CVPR*. IEEE, 2016, pp. 1991–1999.

[29] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional Two-stream Network Fusion for Video Action Recognition," in *CVPR*, 2016, pp. 1933–1941.

[30] L. Wang *et al.*, "Temporal Segment Networks: Towards Good Practices for Deep Action Recognition," in *ECCV*. Springer, 2016.

[31] S. Yu, Y. Cheng, L. Xie, Z. Luo, M. Huang, and S. Li, "A Novel Recurrent Hybrid Network for Feature Fusion in Action Recognition," *JVCIR*, 2017.

[32] L. Sun, K. Jia, K. Chen, D. Y. Yeung, B. E. Shi, and S. Savarese, "Lattice Long Short-Term Memory for Human Action Recognition," in *ICCV*, 2017, pp. 2166–2175.

[33] H. Wang, Y. Yang, E. Yang, and C. Deng, "Exploring Hybrid Spatio-Temporal Convolutional Networks for Human Action Recognition," *MTA*, 2017.

[34] Y. Wang, M. Long, J. Wang, and P. S. Yu, "Spatiotemporal Pyramid Network for Video Action Recognition," in *CVPR*. IEEE, 2017.

[35] G. Varol, I. Laptev, and C. Schmid, "Long-Term Temporal Convolutions for Action Recognition," *TPAMI*, vol. 40, no. 6, pp. 1510–1517, 2018.

[36] J. Zhu, W. Zou, and Z. Zhu, "Two-stream Gated Fusion Convnets for Action Recognition," in *ICPR*. IEEE, 2018.

[37] H. Bilen, B. Fernando, E. Gavves, and A. Vedaldi, "Action Recognition with Dynamic Image Networks," *TPAMI*, 2017.

[38] E. Chen, X. Bai, L. Gao, H. C. Tinega, and Y. Ding, "A Spatiotemporal Heterogeneous Two-stream Network for Action Recognition," *IEEE Access*, vol. 7, pp. 57 267–57 275, 2019.

[39] A. C. S. Santos, H. A. Maia, M. R. Souza, M. B. Vieira, and H. Pedrini, "Fuzzy Fusion for Two-stream Action Recognition," in *VISAPP*. INSTICC, 2020.

[40] Y. Li, B. Ji, X. Shi, J. Zhang, B. Kang, and L. Wang, "TEA: Temporal Excitation and Aggregation for Action Recognition," in *CVPR*, 2020.

[41] A. Brito, M. Vieira, S. Villela, H. Tacon, H. Chaves, H. Maia, D. Concha, and H. Pedrini, "Weighted Voting of Multi-Stream Convolutional Neural Networks for Video-Based Action Recognition using Optical Flow Rhythms," *JVCIR*, 2020.

[42] H. Tacon, A. Brito, H. Chaves, M. Vieira, S. Villela, H. Maia, D. Concha, and H. Pedrini, "Human Action Recognition Using Convolutional Neural Networks with Symmetric Time Extension of Visual Rhythms," in *ICCSA*. Springer, 2019.

[43] H. Tacon, A. Brito, H. L. Chaves, M. B. Vieira, S. M. Villela, H. A. Maia, D. T. Concha, and H. Pedrini, "Multi-stream Architecture with Symmetric Extended Visual Rhythms for Deep Learning Human Action Recognition," in *VISAPP*, 2020, pp. 351–358.

[44] H. Chaves, K. Ribeiro, A. Brito, H. Tacon, M. Vieira, A. Cerqueira, S. Villela, H. Maia, D. Concha, and H. Pedrini, "Filter Learning from Deep Descriptors of a Fully Convolutional Siamese Network for Tracking in Videos," in *VISAPP*. INSTICC, 2020.

[45] H. Maia, M. Souza, A. S. Santos, J. Bobadilla, M. Vieira, and H. Pedrini, "Early Stopping for Two-Stream Fusion Applied to Action Recognition," in *Springer Book of VISAPP 2020*, 2020, [Submitted].

[46] V. C. Lobo-Neto, H. de Almeida Maia, M. R. e Souza, J. C. M. Bobadilla, and H. Pedrini, "Direct Optical Flow Attacks in a Two-Stream Network for Robustness Evaluation," *Computer Vision and Image Understanding*, 2021, [Submitted].

[47] M. Souza, H. Maia, and H. Pedrini, "Survey on Digital Video Stabilization: Concepts, Methods and Challenges," *ACM Computing Surveys*, 2021, [Submitted].