Multi-Loss Recurrent Residual Networks for Gesture Detection and Recognition

Igor L. O. Bastos, Victor H. C. Melo, William Robson Schwartz
Sense Smart Laboratory
Universidade Federal de Minas Gerais
{igorlobastos, victorhcmelo, william}@dcc.ufmg.br

Abstract—**Communication through gestures plays a relevant role in human life, in which a non-verbal language is used to propagate information among individuals. To recognize gestures, computers need to represent and interpret human appearance and motion, involving hands, arms, face, head and/or body, in a mathematical sense. Despite the high applicability in different contexts, most gesture recognition approaches in literature are not designed to deal with unsegmented videos. That is, most approaches do not temporally detect when a gesture occurs, which prevents to explore correlations between detection and recognition tasks, besides their application on real-world scenarios. In this sense, we propose the Multi-Loss Recurrent Residual Network (MLRRN), a multi-task based approach that performs both the recognition and temporal detection of gestures at once. It employs a dual loss function which takes into account the class assignment of each frame of a video to a gesture class and also determines the frame interval associated to each gesture. Our model counts with a dual input, gathering information from appearance and human pose on frames, besides bidirectional recurrent layers and residual modules. According to experiments conducted on ChaLearn Montalbano and ChaLearn ConGD datasets, our approach achieves results comparable to state-of-the-art methods considering average temporal Jaccard metric.**

I. Introduction

Gesture recognition (GR) corresponds to a mathematical interpretation of a human motion by a computer device, involving hands, arms, face, head and/or body [1], with applicability in different contexts, such as navigation on virtual environments, development of aid systems for hearing impaired, sign language recognition, surveillance monitoring and biometric validation [2]–[4]. This applicability led GR to be investigated by a wide range of approaches, which vary in terms of features and learning algorithms employed to perform the task [5]–[7].

Spatial and temporal information are key elements for gesture recognition systems, since they represent changes in appearance and motion over time. In addition, the temporal domain provides information about gesture's structured time disposition of events, where the order of them, also referred as sub-actions, is relevant to determine their labels [5].

Despite major advances achieved [8], GR remains a challenging task due to problems such as illumination variation and acquisition conditions, inconsistent behavior among users, cultural gesture specificities, and large vocabularies [9]. In addition, assembling gesture recognition datasets is an expensive task and most of them presents a major issue regarding the absence of unsegmented videos comprising multiple gestures. As a consequence, gesture recognition approaches usually do not handle unsegmented input streams, leading to models that are unsuitable for real-life communication scenarios.

Focusing on the aforementioned issues, we present Multi-Loss Recurrent Residual Network (MLRRN). The approach employs a novel deep architecture for gesture recognition, with focus on the performing of two correlated tasks at once: (i) gesture temporal detection and (ii) gesture recognition. With that, we intend to create a model that deals with unsegmented input videos, beside exploring the complementarity of these tasks. In addition, MLRRN counts with two input modalities, corresponding to video frames and human joints computed over them, representing the pose of the gesture performers. From these input modalities, information is gathered through the employment of different type of layers, such as 2D spatial convolutional, 3D spatiotemporal convolutional and bidirectional-LSTM. The latter is responsible for exploiting information from the gesture temporal well-defined structure. In addition, it captures the long-term dependency that exists on inputs, taking into account past and future information to assign labels to each input frame.

To evaluate MLRRN, tests were conducted on the ChaLearn Montalbano [10] and ChaLearn ConGD [11] datasets, for which the approach achieves 0.919 and 0.621 as average temporal Jaccard, respectively. On ChaLearn Montalbano, MLRRN obtains a slight improvement over state-of-the-art approaches considering this metric.

II. Related Work

Most gesture recognition approaches are based on the extraction/learning of spatiotemporal features from videos [9]. This highlights the importance of two main factors for the recognition of gestures: (i) appearance, which brings information from gesture parameters such as hand configuration, body/facial expression and inflection point [12]; and (ii) motion, which represents the movement executed by the performer [13].

The applicability in several contexts led gesture recognition to be studied by literature work in the last decades. These approaches, initially based on the employment of handcraft spatiotemporal feature descriptors [14]–[16], tend to capture shape, appearance and motion clues, mostly via image gradients and optical flow [5].

Despite good results achieved by handcrafted-based gesture recognition approaches, the advance of GPUs led to a growing trend toward the application of deep neural networks on the task. These approaches are able to efficiently learn representations to characterize gestures and classify them with high accuracies [5], [6], [9], [17], stimulating the development of increasingly complex and effective models, which usually apply spatiotemporal operations to learn features that better distinguish dataset classes [5].

The approach proposed by Duan et al. [18], for instance, considers several input modalities, such as optical flow, RGB, depth and saliency to gather richer information from gesture inputs. Each modality generates a voting representation (considering the classes of the dataset), which are fused to produce the output class. This research achieved accurate outcomes for gesture recognition on ChaLearn IsoGD dataset [11].

Differently from Duan et al. [18], many approaches exploit the strong temporal correlation between sub-events in gesture videos through the employment of recurrent models. These models have achieved state-of-the-art results for most gesture recognition datasets. Molchanov et al. [5], for instance, proposed a model that extracts spatiotemporal features from video clips through spatiotemporal convolutions; propagating such information with the employment of a recurrent layer. Although simplistic, this architecture proved to be effective for recognition of gestures on datasets such as SKIG [19], ChaLearn IsoGD [11] and Multimodal Dynamic Gesture [5].

Aiming at capturing the temporal correlation in gesture videos, Nishida and Nakayama [6] investigated an architecture composed by LSTM layers to handle videos with variable-length gestures. To create a spatiotemporal representation, multiple temporal modalities are fused, which produced a high accuracy outcome on the SKIG dataset [19].

The accurate results achieved with recurrent models led to the development of even more complex approaches based on the employment of bidirectional recurrent layers. The research proposed by Zhang et al. [9], for instance, employs bidirectional LSTM layers to produce a rich representation from video frames, achieving accurate outcomes. In turn, Pigou et al. [13] employed these layers in a model to detect and recognize gestures on unsegmented videos. On their research, residual modules are employed to conserve gradients; important point on the training of deeper networks.

Zhu et al. [20] also invested on an approach to detect and recognize gestures on unsegmented videos. However, instead of producing a single model to do both tasks, Zhu et al. [20] used an isolated temporal recognition network based on Res3D architecture [21] able to produce isolated videos through the recognition of boundaries (transition frames). To poise the two classes employed in this task (boundaries and non-boundaries), a balanced squared hinge loss function is applied. In addition, temporal dilations are included on the convolutional layers of this network to gather contextual information.

Even though the literature approaches present accurate results for gesture recognition, some gaps are still noticed, mainly regarding the existence of very few techniques that consider unsegmented videos, with none of them using detection and recognition tasks to improve the outcomes of each other. Thus, despite similarities with some previously recurrent detection/recognition methods, our approach, named Multi-Loss Recurrent Residual Network (MLRRN), differs from them due to the employment of a single model to perform detection and recognition of gestures, using a dual-output loss to execute these tasks simultaneously. Besides that, the accurate outcomes of MLRRN are also provided by the combination of elements such as our dual modality input, bi-directional LSTM layers and residual modules, which is innovative in relation to literature methods.

III. Handling Unsegmented Videos

Most gesture recognition approaches are designed to handle segmented videos, i.e., a single gesture is presented on the entire video. Despite the existence of accurate methods applied to segmented gesture recognition datasets, such as SKIG [19] and ChaLearn IsoGD [11], they are not suitable to perform gesture recognition on real-life scenarios where there exists a fluid conversation that is less and less dependent of any control over the communication scenario [22]. In this sense, to handle unsegmented gesture videos, two tasks must be considered: (i) gesture detection and (ii) gesture recognition.

Gesture Detection. Determining the start and the end of a gesture in an unsegmented video corresponds to mark the frame interval comprised by this gesture. This task is named gesture detection or gesture temporal detection [5]. Despite simulating a real-life scenario (i.e., unsegmented stream of data), few approaches tackle temporal detection due to its high complexity. The assignment of the label *gesture* or *no-gesture* to each frame is a difficult task since the positive class (i.e., gesture) tends to present high intraclass variation and the negative class (i.e., no-gesture) tends to suffer from the lack of standard postures, producing inconsistent behavior among users and even similarities with the gesture class [5], [13]. In addition, it is important to consider the three temporal phases on the gestures: preparation, nucleus and retraction. The nucleus, core of the gesture, is associated to motion and postures executed by the performer that characterize each gesture. In turn, preparation and retraction are transition phases that regard assuming a posture to start the gesture or going to relaxed postures, respectively. While the nucleus is the discriminative phase [23], the other two phases can be similar for different gesture classes and hence less useful or even detrimental to classification, just representing transitions between no-gesture frames to gesture frames and vice-versa.

Gesture Recognition. Gesture recognition corresponds to the assignment of a label to a gesture sequence. This task has been tackled by several works and is the main task of gesture recognition approaches [5], [17], [22]. Gesture recognition is a typical classification task in which frame sequences are associated to one of the trained classes. Differently from the gesture detection, the complexity of gesture recognition presents two main points: similarity between different classes and possible large number of classes.

IV. Proposed Approach

To perform gesture recognition on unsegmented videos, we propose a model, the *Multi-Loss Recurrent Residual Network (MLRRN)*, that presents three main characteristics:

Recurrent layers: As aforementioned, gestures present a well-structured time disposition of events, making room for an efficient application of recurrent models. Since state-of-art results are mostly achieved by these models and we intend to handle unsegmented videos, recurrent layers are extremely suitable for this task. With recurrent layers, our model is able to extract long-term dependencies and to establish relations between different frames of the input.

Frame-level input: The MLRRN model was developed with a frame-level input, i.e., the input corresponds to one frame of the video per timestep. However, to provide local temporal information, for each input frame, we also provide the previous and the next four frames, producing a 9-frame tensor.

Multi-task (Detection and Recognition): Our model outputs labels for each frame of a video, performing both detection and recognition tasks at once. These tasks present a complementary behavior and when considered in a jointly way, they enhance the outcomes of the other. The detection task, for instance, gives a negative response for no-gesture frames, evidencing that these frames cannot be associated to any class of recognition task. In turn, the recognition task emphasizes transition frames, revealing margins of gestures and no-gesture intervals. Thus, we developed a multi-task model to perform both tasks at once, counting with a dual-loss that weights both detection (binary cross-entropy) and recognition (categorical cross-entropy) responses. Figure 1 illustrates the proposed MLRRN and Sections IV-A and IV-B detail the method.

A. Input Data

The main input of the MLRRN consists of frames of a video sequence. For each time offset (timestep), a next frame from the sequence is fed to the approach. However, instead of using the raw RGB frame, we extract activations from the fully connected layer 7 (*fc7*) of VGG-16 trained on ImageNet to produce a spatial description. This feature contains 4096 dimensions and was reshaped to a 64x64 representation before feeding our model (model spatial input). With VGG-16 activations, we obtained better results than adding spatial layers to the model, leading to a reduction on the number of parameters and a model that is less prone to overfitting, with lower training data requirement and easier convergence.

As showed in Figure 1, we consider a secondary input (model joint input). This input corresponds to human body joints computed with the pose estimation technique proposed by Cao et al. [24], which uses a nonparametric representation to learn to associate appearance of body parts with individuals in images. This information is used to produce a pose signature of individuals that are performing gestures on the video.

B. MLRRN Model

The Multi-Loss Recurrent Residual Networks (MLRRN) is a multi-task architecture that performs two different tasks: gesture temporal detection and gesture classification. Since the inputs of the architecture corresponds to frames of a video, detection and classification labels are generated for each frame, considering both losses of the model, as depicted in Figure 1.

The first point to notice on MLRRN model is the *Spatial Input* which corresponds to VGG-16 activations for every frame. As aforementioned, MLRRN takes into account the previous and subsequent frames, empirically determined as a 9-frame input for every frame of the video. For each new timestep, an 1-frame offset is performed and the current frame is updated, as illustrated in Figure 2. According to this strategy, the first and last four frames are never considered as the main input but are used as auxiliary inputs for adjacent frames.

The second block of MLRRN model is called *Multi-scale Spatial Convolutions*. It corresponds to spatial convolutional layers that consider different filter dimensions. With that, we try to gather information from different scales. In addition, this multi-scale block intends to mitigate the fact that MLRRN performs convolutions over fully-connected activations (reshaped to 64x64) from the VGG-16. With that, we enforce a spatial relation that is spread over the fully-connected representation and, with these convolutions considering different scales, we tend to better capture information from that. Furthermore, it is important to mention that these convolutions are performed over each frame information (activations from current and auxiliary frames) isolated. We do this to increase the capacity of the model at this point, with the aim of producing richer representations from each frame separately. This contributes to provide wealthier information regarding the frame differences along the input frames for each timestep.

The next block is the *Tensor Assembling*, where feature maps obtained from convolutional layers of individual frame responses are concatenated in different ways. With that, a tensor is produced by each combination of these maps, allowing the performing of spatiotemporal convolutions since the maps from different frames represent a time variation on the input. Figure 3 illustrates the combinations that are performed and the tensors produced from MLRRN inputs. One could notice 3- and 9-input tensor combinations, experimentally assembled, from what spatiotemporal (3D) convolutions are performed. At end of this block, responses of convolutions over all tensors are concatenated and propagated through the network.

With the tensor in hand and spatiotemporal convolutions performed, *residual blocks* are employed. These blocks, adapted from the research proposed by Pigou et al. [13], are important to maintain the gradient in a deep network, such as MLRNN. In addition, it allows the employment of operations (mostly convolutions), over representations with lower and higher degree of semantics.

The *bi-recurrent layer* is the next block on our model. This layer presents a crucial importance since it explores the temporal structure of gestures, in which there is an order of events that is relevant to determine their labels. In addition, with this layer it is possible to obtain a response for each frame considering the high dependency that exists on other frames. In the case of MLRRN, a bi-recurrent layer is employed,

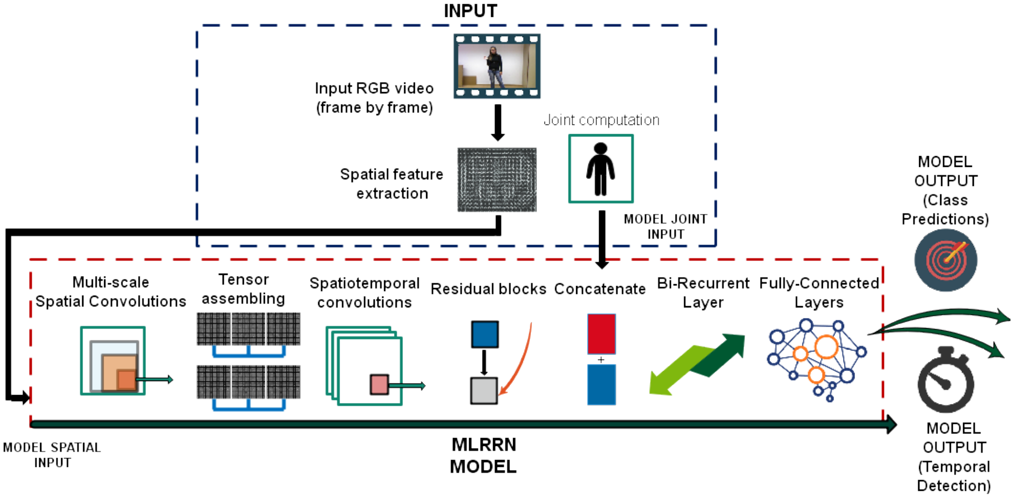
INPUT
Input RGB video (frame by frame)
Spatial feature extraction
Joint computation
MODEL JOINT INPUT
MODEL OUTPUT (Class Predictions)
Multi-scale Spatial Convolutions
Tensor assembling
Spatiotemporal convolutions
Residual blocks
Concatenate
Bi-Recurrent Layer
Fully-Connected Layers
MODEL SPATIAL INPUT
MLRRN MODEL
MODEL OUTPUT (Temporal Detection)

Fig. 1. Illustration of the proposed MLRRN approach.

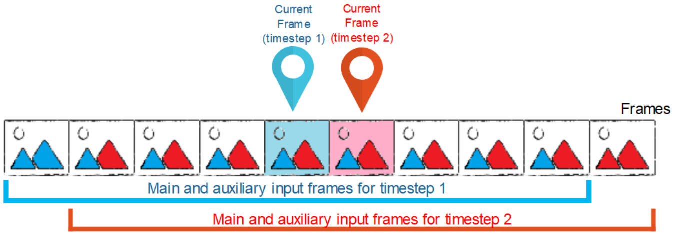
Current Frame (timestep 1)
Current Frame (timestep 2)
Frames
Main and auxiliary input frames for timestep 1
Main and auxiliary input frames for timestep 2

Fig. 2. Input of MLRRN for different timesteps.

gathering information from previous and future frames. According to our tests, bi-LSTM layers presented better results than vanilla RNNs or GRUs, which is expected due to the long-term dependency that exists between a frame and its previous and future instances. Finally, the secondary input of the model (human joints) is concatenated to the representation obtained from residual blocks, being both fed to this recurrent layer.

At end, the MLRRN model presents a stack of *fully connected-layers* that act over the recurrent representation. These layers result in two output layers, one responsible for the class prediction (output is the label of a frame considering gesture classes) and the other corresponds to temporal detection (determines if a frame is part of a gesture or not). The complete MLRRN model is showed in Figure 4. One can notice that the outputs of the model are a binary and a (n+1)-class softmax layers, which apply binary and categorical cross entropy loss functions, respectively. The binary output corresponds to temporal detection, classifying frames as gesture and no-gesture. The other output has its dimensionality associated to the number of gesture classes of the dataset added by one. This addition of an extra class regards the existence of the no-gesture class also for this output. However, this class receives no-weight on the training of the model.

V. Experimental Results

To evaluate MLRRN, experiments are conducted on ChaLearn Montalbano [10] and on ChaLearn CongGD [11] datasets. For both, detection and recognition of gestures are performed following the standard evaluation protocols.

Experimental Setup. Most parameters of the MLRRN architecture (shown in Figure 4), such as the choice for a bidirectional LSTM layer, the employment of residual modules and activation function of layers, were determined by tests on the validation set of the ChaLearn Montalbano [10]. However, since ChaLearn ConGD [11] is a complex dataset, containing more videos and gesture classes, the architecture depicted on Figure 4 was adjusted before conducting experiments on it, with the insertion of one extra residual block and the increment of the number of feature maps in some layers. Finally, the output of the softmax layer responsible for the gesture recognition had its size adjusted in order to contemplate the classes of the ChaLearn ConGD and the no-gesture class.

To train the model shown in Figure 4, the learning rate was set experimentally to 0.0001. All convolutional layers employ ReLU activation (shown in blue), except for some on residual blocks (shown in green), which employ ELU. LSTM and fully-connected layers employ sigmoid activation (shown in purple). The model evaluated on ChaLearn Montalbano contains 53.1Mi parameters while the one evaluated on ChaLearn ConGD contains 60.2Mi parameters, both trained on a NVIDIA GeForce 1080Ti.

The evaluation of MLRRN considered the average temporal Jaccard metric, which takes into account both the accuracy of the network responses and the overlap between the responses and ground-truth annotations. It is defined by

$$Amplitude(J_{s,n}) = \frac{A_{s,n} \cap B_{s,n}}{A_{s,n} \cup B_{s,n}}, \tag{1}$$

where A$_{\mathrm{s,n}}$ denotes the ground truth of gesture n at sequence s and B $_{\mathrm{s,n}}$ is the prediction of such gesture at sequence n.

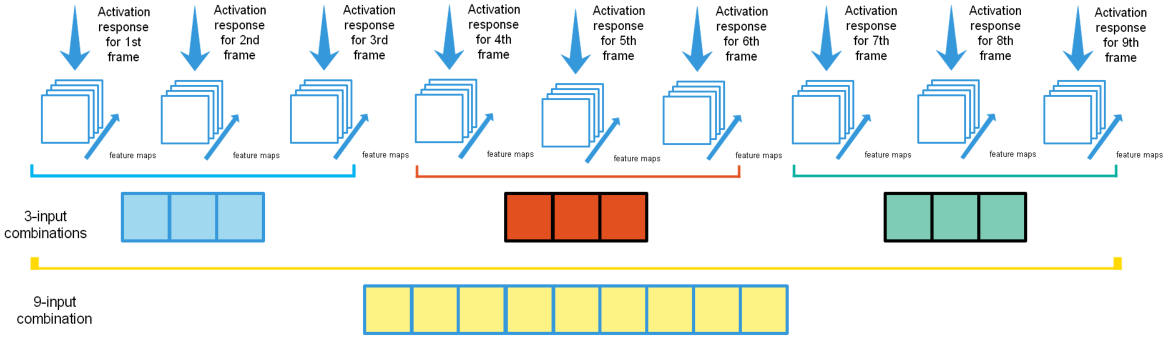
Activation response for 1st frame
Activation response for 2nd frame
Activation response for 3rd frame
Activation response for 4th frame
Activation response for 5th frame
Activation response for 6th frame
Activation response for 7th frame
Activation response for 8th frame
Activation response for 9th frame
feature maps
3-input combinations
9-input combination

Fig. 3. Assembling of tensors on MLRRN architecture.

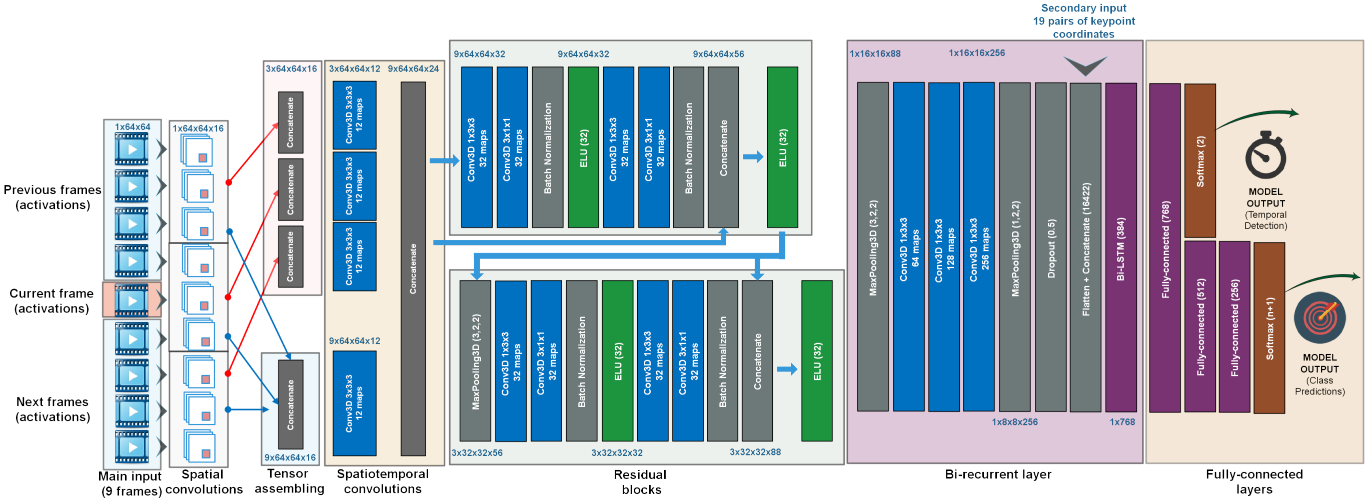

Fig. 4. MLRRN architecture. Layers employ ReLU (blue), ELU (green), sigmoid (purple) and softmax activations (brown).

Datasets. ChaLearn Montalbano [10] is a public dataset composed by 970 RGB, depth and user-segmented videos, simulating a continuous recognition scenario. Each video contains multiple gestures, resulting in more than $14,000$ from 20 Italian sign gesture categories executed by 20 performers. This dataset employs a protocol with mutually exclusive training, validation and testing subsets. All videos are annotated with the beginning and end of each gesture, besides the class they belong. These gestures are separated by intervals of frames in which the performers relax, being associated to none of the 20 classes of the dataset (no-class frames).

ChaLearn CongGD [11] is a public dataset comprising 249 different gesture classes and more than $56,000$ gesture performances disposed into $22,535$ videos. Similarly to ChaLearn Montalbano, this dataset presents a standard evaluation protocol with mutually exclusive training, validation and testing subsets. ChaLearn ConGD is considered a challenging dataset due to the variability introduced by a high number of classes, performers, backgrounds and illumination constraints. Additionally, the gesture recognition in this dataset requires an initial temporal detection of gestures. Differently from ChaLearn Montalbano, ChaLearn ConGD dataset is annotated with no intervals (no-class frames) between gestures, with a 1-frame difference between gestures.

A. Evaluation on ChaLearn Montalbano

Since MLRRN relies in frame-level inputs, it is not necessary to perform any adjustment on the length of the data. For each frame, a RGB input tensor is assembled along the joint response of the technique of Cao et al. [24]. Moreover, once MLRRN presents a bidirectional recurrent layer, it considers previous and future frames on the response produced for every input frame. An important point on this approach is the batch size, since it must be large enough to provide information that reflects long-term dependency that exists between frames. However, the larger the batch size, the larger must be the number of parameters of this recurrent layer, leading to problems such as higher time to train and training data requirement, proclivity to overfitting and struggling convergence. On ChaLearn Montalbano, a batch size of 50 is used, along *Adam* optimizer with a learning rate value of 0.0001.

Since ChaLearn Montalbano comprises 20 gesture classes and one non-gesture class, it was necessary to train a 21-class classification model for the recognition task. For the detection task, this model acts as a binary classifier, outputting labels that indicate whether a frame is part of a gesture or

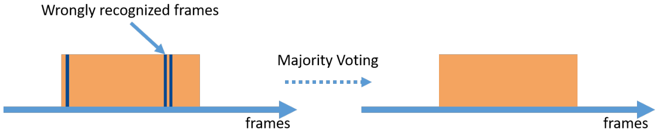

Fig. 5. Post processing on MLRRN response. Colors indicate the label of the frame.

TABLE I
AVERAGE TEMPORAL JACCARD SCORE ON CHALEARN MONTALBANO DATASET.

	Approach	Jaccard Score
Results	MRF, KK, PCA, HOG [14]	0.827
	AdaBoost, HOG [15]	0.834
	Multi-scale DNN [25]	0.870
	TempConv + LSTM [13]	0.906
	3DCNN + ConvLSTM [20]	0.915
Our Results	MLRNN	0.914
	MLRRN + 3-size mask	0.916
	MLRRN + 5-size mask	**0.919**
	MLRRN + 7-size mask	0.912
	MLRRN + 9-size mask	0.908

not. Based on that, we evaluated our approach on the test subset of ChaLearn Montalbano, which provided frame-level recognition accuracy of 96.87%. This accurate result led to a high average temporal Jaccard response of 0.914 considering the output of the recognition task of the model.

We also executed a post-processing on MLRRN recognition output, performing a majority voting around each frame response. To do that, masks with different sizes were employed, making the label of each frame to be the most common response of the own frame and its neighbors. The employment of this post-processing regards the frame-level output of MLRRN what makes the approach sensible to wrongly recognized frames, as depicted in Figure 5, where colors indicate the class label of a frame. One could notice incorrect responses (purple) among the frames of a gesture (represented in orange).

Table I shows the results of the proposed MLRRN and state-of-art approaches on ChaLearn Montalbano [10] considering the conventional and post-processed outputs with different mask sizes. According to the results, it is possible to see that larger masks tend to degrade the Jaccard response of the model, since transition zones (between different gestures and gesture to non-gesture frames) are corrupted. With a 5-size mask, we achieved state-of-art results, outperforming the method proposed by Zhu et al. [20]. It is important to remark that the research proposed by Molchanov et al. [5] achieved a higher Jaccard score on this dataset. However, since that approach uses ground-truth annotations to perform gesture detection, their outcomes cannot be compared to ours.

B. Evaluation on ChaLearn ConGD

The evaluation of MLRRN on ChaLearn ConGD presented few changes in comparison to ChaLearn Montalbano. Most of them, as aforementioned, are related to the increment of the model capacity to handle a more complex dataset and the increase on the number of classes for classification, which goes to 250. In addition, since ChaLearn ConGD contains shorter videos, the batch size was experimentally set to 40. *Adam* optimizer was employed along a learning rate of 0.00015.

TABLE II
AVERAGE TEMPORAL JACCARD SCORE ON CHALEARN CONGD DATASET.

	Approach	Jaccard Score
Results	Two-Stream ConvNets + Ensemble learning [26]	0.5307
	Faster-RCNN + Heterogeneous networks [12]	0.5950
	Faster RCNN+ C3D [27]	0.6103
	TS1-Res3D + Multiply Fusion [20]	0.6435
	TS1-Res3D + Average Fusion Fusion [20]	0.7163
Our Results	MLRNN	0.5627
	MLRRN + 5-detection classes	0.6204
	MLRRN + 5-detection classes + 5-mask	0.6231

Since ChaLearn ConGD presents a 1-frame distance between different gestures, the impact of the detection task was mitigated. MLRRN achieved a frame-level recognition accuracy of 73.23%. In turn, the model reached a temporal Jaccard score of 0.5627, as showed in Table II with results of other state-of-art approaches on the ChaLearn ConGD. We performed an additional evaluation on this dataset with the aim of enhancing the outcomes of MLRRN, in which we assigned five different labels for classes on the detection task. For instance, classes 0-49 are assigned to label 1, classes 50-99 to label 2, and so on. With that, the detection task could determine intervals related to these classes, gathering information about their lenght and transitions that exist between them. As a consequence, a significant improvement on temporal Jaccard score was noticed, evidencing the impact of detection task for the proper recognition of gestures.

On the ChaLearn ConGD, we can notice similar problems as the ones found on ChaLearn Montalbano, such as the existence of some noise between frames outputs of a class. Besides that, since the annotation of this dataset does not include no-gesture frames, the performance of MLRRN is deteriorated with our model acting in a similar way to a standard classifier, with no contribution coming from the detection task. However, with the assigning of five classes for detection, the impact of this multi-task was greatly enhanced, even this 5-class separation being performed with a very simple criterion.

C. Ablation Study of MLRRN

Since MLRRN is composed by several components, an ablation study shows to be valuable. Table III presents outcomes obtained with the ablation evaluation of MLRRN on ChaLearn Montalbano for recognition and detection tasks. The results obtained with this study justified our choices on the assembling of the final MLRRN architecture. From this study, some points need to be highlighted, as: (i) the huge impact of recurrent layers, indicating the importance of temporal information and disposition of events for gesture recognition, (ii) the complementarity of appearance and skeleton inputs and (iii) detection and recognition losses, with improvements obtained from the combination of them. Detection outcomes do not correspond to the final response of the approach. Presenting them intends to highlight the improvement, even for this complementary task, obtained with our multi-task strategy.

TABLE III
ABLATION STUDY PERFORMED ON CHALEARN MONTALBANO DATASET.

	Approach variation	Jaccard Score (recognition)
Results	Only appearance input	0.830
	Only skeleton input	0.659
	No residual blocks (skip-connections removed)	0.881
	No recurrent layers	0.547
	Single-directional recurrent layers	0.861
	Only recognition task (no detection)	0.806
	Full model	0.919
	Approach variation	**Jaccard Score (detection)**
Results	Only detection task (no recognition)	0.949
	Full model	0.982

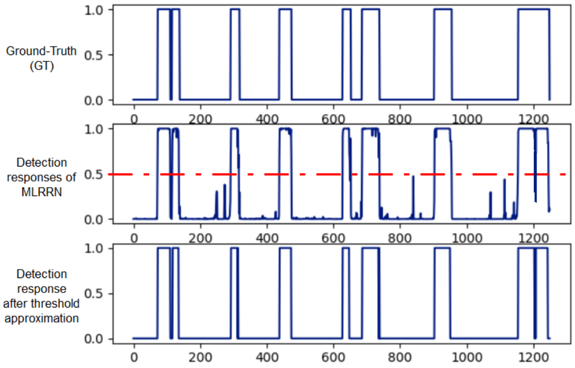

Fig. 6. Detection responses of MLRRN. Dashed red line represents the threshold used to approximate responses.

VI. DISCUSSION

Results on ChaLearn Montalbano and ChaLearn ConGD evidenced the high performance of MLRRN and the positive impact of the employment of correlated tasks. The superiority on ChaLearn Montalbano, for which we achieved state-of-art performance, is greatly related to the gesture disposition of this dataset and their annotations, which provide no-gesture frame intervals between the gesture instances. On ChaLearn ConGD, the performance of MLRRN is mitigated mostly due to annotations. On that, frames associated to relaxing postures of performers that should be annotated as no-gesture frames, are still annotated as part of gesture intervals.

In addition to the quantitative results showed, it is interesting to notice, in a qualitative way, how the method performed for both tasks. Figure 6 depicts the response of the temporal detection task for a video of ChaLearn Montalbano compared to the ground-truth response. In this figure, low-responses (next to 0) indicate the absence of gesture and high-responses (next to 1) indicate the presence of a gesture. One can notice the existence of some noise on the detection, which is filtered by the employment of an empirical threshold (0.5) used to approximate the responses to 0 or 1. It is interesting to notice that abrupt transitions between non-gesture and gesture frames are well-detected by the model in most cases in this dataset.

The detection and recognition tasks produced accurate outcomes for most videos in the dataset. However, the last frames of some gestures were predicted as no-class frames, what produced a shortening effect on recognition. This point can be associated to the similarity, in terms of appearance and motion, between frames of retraction phase (final part of a gesture) and relaxing phase (post gesture), with this latter being associated to no-class frames. In addition, the lack of no-class standard postures and inconsistent behavior of performers could have contributed to this outcome, illustrated on Figure 7.

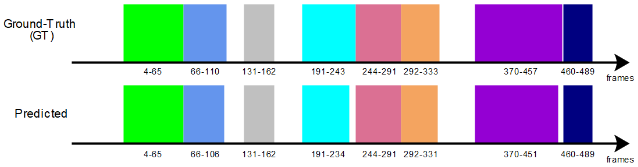

Fig. 7. Shortening effect on the recognition of gestures by MLRRN.

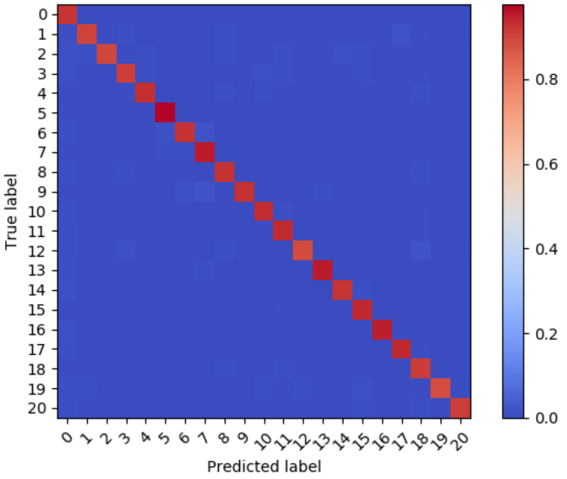

Fig. 8. Frame-level confusion matrix of MLRRN on ChaLearn Montalbano. Indexes represent different dataset classes. Class-0 represents *no-gesture*.

On the evaluation of the proposed MLRRN, we could notice that, even on ChaLearn Montalbano for which the approach presented very accurate results, some frames belonging to all classes of the dataset are recognized as class-0 (i.e., no-gesture class). This result is mostly associated to the retraction frames of each gesture, since these frames are similar, in terms of appearance and even motion, to the relaxing postures that are common on post-gesture frames. Figure 8 depicts the frame-level confusion matrix of MLRRN responses on ChaLearn Montalbano, for which the approach presented more than 96% of accuracy. It is possible to see that for almost all classes, some frames are predicted as no-gesture (class 0).

Finally, we performed a cross-dataset test, where we used a model trained on ChaLearn Montalbano to act over videos of ChaLearn ConGD. Since the class-recognition labels make no sense in this scenario, we only qualitatively verified whether the detection task was able to produce reasonable results. Figure 9 depicts the detection task on videos of ChaLearn ConGD. For that, we manually annotated the preparation, nucleus and relaxing phases of gestures. According to the results, the model presents high responses for the nucleus part of the gestures, with oscillating responses on preparation and retraction and low responses on the relaxing postures, annotated on ChaLearn Montalbano as the non-gesture class. Even though not completely accurate, the results suggest that the trained model is able to indicate the separation between

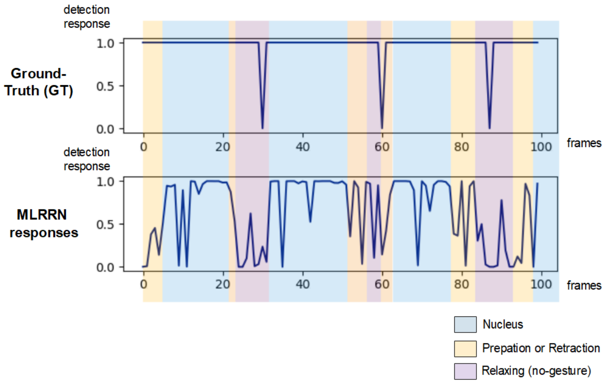

Fig. 9. Detection responses of MLRRN on ChaLearn ConGD.

gestures in a different dataset, which is promising once that experiment emulates conditions similar to real-life scenarios.

Since the model trained on ChaLearn ConGD presents no impact on detection task due to the lack of annotations regarding the no-gesture frames, this detection task makes no sense to be evaluated on ChaLearn Montalbano dataset.

VII. Concluding Remarks

This paper presented a gesture detection/recognition approach, the Multi-Loss Recurrent Residual Network (MLRRN), based on the application of multi-task, residual blocks and recurrent layers. Experiments were conducted on ChaLearn Montalbano, for which the approach achieved state-of-art performance, and on ChaLearn ConGD datasets, both evaluated considering the average temporal Jaccard metric. Even not surpassing the outcomes of methods such as the one proposed by Zhu et al. [20] on ChaLearn ConGD, the method presented accurate responses, with tests indicating how correlated tasks could enhance outcomes of each other. On this dataset, a more sophisticated separation of detection classes, instead of our 5-class simple strategy on ChaLearn ConGD, could improve the results of MLRRN. Finally, since our model handles unsegmented input streams and needs no detection/pre-processing steps before the recognition of gestures, it presents a straight forward applicability in real-life scenarios, which comprises a fluid communication.

Acknowledgments

The authors would like to thank the National Council for Scientific and Technological Development – CNPq (Grants 311053/2016-5 and 438629/2018-3), the Minas Gerais Research Foundation – FAPEMIG (Grants APQ-00567-14 and PPM-00540-17), the Coordination for the Improvement of Higher Education Personnel – CAPES (DeepEyes Project). This study was financed in part by the Coordenacao de Aperfeicoamento de Pessoal de Nivel Superior - Brasil (CAPES) - Finance Code 001.

References

[1] S. Mitra and T. Acharya, "Gesture recognition: A survey," *Transactions Systems, Man and Cybernetics Part C*, vol. 37, no. 3, pp. 311–324, 2007.

[2] V. Pavlovic, R. Sharma, and T. Huang, "Visual interpretation of hand gestures for human-computer interaction: A review," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 677–695, 1997.

[3] S. Xu and Y. Xue, "A long term memory recognition framework on multi-complexity motion gestures," in *ICDAR*, 2017, pp. 201–205.

[4] H. Zhou and Q. Ruan, "A real-time gesture recognition algorithm on video surveillance," in *8th International Conference on Signal Processing*, vol. 3, 02 2006.

[5] P. Molchanov, X. Yang, S. Gupta, K. Kim, S. Tyree, and J. Kautz, "Online detection and classification of dynamic hand gestures with recurrent 3d convolutional neural networks," in *Proceedings of 2016 IEEE CVPR*, 2016, pp. 4207–4215.

[6] N. Nishida and H. Nakayama, "Multimodal gesture recognition using multi-stream recurrent neural network," in *7th Pacific-Rim Symposium on Image and Video Technology - Volume 9431*, 2016, pp. 682–694.

[7] C. Cao, Y. Zhang, Y. Wu, H. Lu, and J. Cheng, "Egocentric gesture recognition using recurrent 3d convolutional neural networks with spatiotemporal transformer modules," in *2017 IEEE ICCV*, vol. 00, 2018, pp. 3783–3791.

[8] D. Wu, L. Pigou, P. Kindermans, N. D. Le, L. Shao, J. Dambre, and J. Odobez, "Deep dynamic neural networks for multimodal gesture segmentation and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 8, pp. 1583–1597, 2016.

[9] L. Zhang, G. Zhu, P. Shen, J. Song, S. A. Shah, and M. Bennamoun, "Learning spatiotemporal features using 3dcnn and convolutional lstm for gesture recognition," in *IEEE ICCV*, 2017.

[10] S. Escalera, X. Barao, J. Gonzalez, M. Bautista, M. Madadi, M. Reyes, V. Ponce-Lopez, H. Escalante, J. Shotton, and I. Guyon, "ChaLearn LAP Challenge 2014: Dataset and Results," in *ECCV*, 2014.

[11] J. Wan, S. Z. Li, Y. Zhao, S. Zhou, I. Guyon, and S. Escalera, "Chalearn looking at people rgb-d isolated and continuous datasets for gesture recognition," in *2016 IEEE CVPR Workshops*, 2016, pp. 761–769.

[12] H. Wang, P. Wang, Z. Song, and W. Li, "Large-scale multimodal gesture recognition using heterogeneous networks," in *ICCV Workshops*. IEEE Computer Society, 2017, pp. 3129–3137.

[13] L. Pigou, M. V. Herreweghe, and J. Dambre, "Gesture and sign language recognition with temporal residual networks," in *ICCV Workshops*. IEEE Computer Society, 2017, pp. 3086–3093.

[14] J. Y. Chang, "Nonparametric gesture labeling from multi-modal data," in *Computer Vision - ECCV 2014 Workshops*. Springer International Publishing, 2015, pp. 503–517.

[15] C. Monnier, S. German, and A. Ost, "A multi-scale boosted detector for efficient and robust gesture recognition," in *ECCV Workshops*, 2015.

[16] I. L. O. Bastos, M. F. Angelo, and A. Loula, "Recognition of static gestures applied to brazilian sign language (libras)," in *SIBGRAPI*, 2015.

[17] I. L. O. Bastos, V. H. C. Melo, G. R. Goncalves, and W. R. Schwartz, "Mora: A generative approach to extract spatiotemporal information applied to gesture recognition," in *15th International AVSS*, 2018.

[18] J. Duan, S. Zhou, J. Wan, X. Guo, and S. Z. Li, "Multi-modality fusion based on consensus-voting and 3d convolution for isolated gesture recognition," *CoRR*, 2016.

[19] L. Liu and L. Shao, "Learning discriminative representations from rgb-d video data," in *IJCAI*, 2013, pp. 1493–1500.

[20] G. Zhu, L. Zhang, P. Shen, J. Song, S. Shah, and M. Bennamoun, "Continuous gesture segmentation and recognition using 3dcnn and convolutional lstm," *IEEE Transactions on Multimedia*, 9 2018.

[21] D. Tran, J. Ray, Z. Shou, S.-F. Chang, and M. Paluri, "Convnet architecture search for spatiotemporal feature learning," *CoRR*, vol. abs/1708.05038, 2017.

[22] Y. Song, D. Demirdjian, and R. Davis, "Continuous body and hand gesture recognition for natural human-computer interaction," *ACM Transactions on Interactive Intelligent Systems*, vol. 2, p. 5, 03 2012.

[23] D. M. Gavrila, "The visual analysis of human movement: A survey," *Computer Vision and Image Understanding*, vol. 73, pp. 82–98, 1999.

[24] Z. Cao, T. Simon, S. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *2017 IEEE CVPR*, 2017.

[25] N. Neverova, C. Wolf, G. W. Taylor, and F. Nebout, "Moddrop: Adaptive multi-modal gesture recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 8, pp. 1692–1706, 2016.

[26] H. Wang, P. Wang, Z. Song, and W. Li, "Large-scale multimodal gesture segmentation and recognition based on convolutional neural networks," in *ICCV Workshops*, 2017.

[27] Z. Liu, X. Chai, Z. Liu, and X. Chen, "Continuous gesture recognition with hand-oriented spatiotemporal feature," in *ICCV Workshops*, 2017.