CAPSULE TRANSFORMER NETWORK FOR DYNAMIC HAND GESTURE RECOGNITION USING MULTIMODAL DATA

Alexandre Lebas, Rim Slama, Hazem Wannous

CAPSULE TRANSFORMER NETWORK FOR DYNAMIC HAND GESTURE RECOGNITION USING MULTIMODAL DATA

Alexandre Lebas[†] *Rim Slama*[⋆] *Hazem Wannous*[†]

[†] IMT Nord Europe, University of Lille, CNRS UMR 9189 - CRIStAL F-59000 Lille, France
[⋆]CESI LINEACT, UR 7527, Lyon, 69100, France

ABSTRACT

In recent years, deep learning techniques have achieved remarkable success in video analysis and more especially in action and gesture recognition. Even though convolutional neural networks (CNNs) remain the most widely used models, they have difficulty in capturing the global contextual information involving spatial and temporal domains or inter-modality due to the local feature learning mechanism. This paper introduces a Capsule Transformer Network, which composed of a frame capsule module for extracting hand features and a gesture transformer module for modeling the temporal features and recognizing the dynamic gesture. Spatial attention is ensured through the capsule module to enhance the spatial information of the hand image, while the transformer module guarantees temporal attention through gesture sequence. We propose to use multimodal data, including RGB, depth and IR data, which improves the accuracy of our approach as it better captures the 3D structure of the hand and can distinguish between similar hand gestures. Testing on two datasets, Briareo and SHREC17, the proposed approach outperforms or equals previous methods.

***Index Terms*—** hand gesture recognition, capsule network, transformer , multi-modal data

1. INTRODUCTION

The increase in low-cost RGB-D devices has resulted in more datasets that offer multimodal data, including infrared, depth, and RGB data for each gesture, leading to more research on multimodal methods. To ensure that the proposed method can work in the presence of dramatic and fast light changes, recent works use these light-invariant modalities [1]. To tackle hand gesture recognition tasks using these different modalities, it is necessary to develop a resilient algorithm that can accurately identify hand gestures from RGB-D videos [2, 3]. In recent years, the field of computer vision has seen a significant shift in the way researchers approach image and video understanding tasks. While Convolutional Neural Networks (CNNs) have been the standard approach for many years [4, 5], they face limitations such as invariance due to pooling and the inability to understand the relationship of spatial features between convolutional layers. The introduction of Capsule Networks has been seen as a promising solution to these problems. While Capsule Networks are a relatively new approach, they have already shown interesting results in hand gesture recognition applications [6, 7]. In these works, Capsule Networks have proven their ability to extract more relevant spatial features of the image and understand the hierarchical relationships between these features. Besides, the same is observed regarding temporal feature modelling. Initially, recurrent neural networks such as LSTMs are the most used to capture motion dynamics [8]. However, since the appearance of transformers by Vaswani et.al [9] several works in the literature adopted transformers for video analysis [10, 1]. Authors of [11, 1] recently proposed transformer-based approaches in order to tackle the action and hand gesture recognition tasks. Motivated by these observations regarding spatial and temporal features evolution, we present in this paper a spatio-temporal approach taking advantage of both capsule network and transformers within multimodal data. The first component helps to create links between spatial features and to have more relevant features with routing by agreement. The second one has as a role to incorporate temporal attention which enables the analysis of how the spatial features of the hand evolve over time. This is critical for dynamic hand gesture recognition, as the same hand shape may represent different gestures depending on its motion. By capturing the temporal evolution of hand gestures, the proposed approach can more accurately predict the intended gesture. To valid our approach, two benchmarks are used : Briareo [12] and SHREC17 [13]. The main contributions of this paper are: (i) We introduced a spatiotemporal capsule-transformer based approach, which extracts efficiently spatiotemporal information from dynamic hand gestures. (ii) We studied different modalities and fusion strategies and their impact on the proposed approach. (iii) Competitive results are achieved on two benchmarks. The code of the proposed approach is available in this link.

2. OUR APPROACH

As shown in Figure 1, our model is composed of three main components. The first one is the feature extraction module. It

is based on ResNet and CapsNet. The second is the temporal feature analysis, which contains the proposed transformers encoder. The third component is ensuring the classification step. This latter is composed of one dense layer which predicts the probability distribution. In the following feature extraction and temporal analysis modules are detailed.

2.1. Spatial feature module

Capsule network was originally presented in [14] that introduced capsules inside a CNN. In their architecture, a group of neurons that captures the parameters of a particular feature is considered as a capsule entity. After the low-level feature capsules have been computed, they can be aggregated by the dynamic routing algorithm to form higher-level capsules that represent more abstract features.

We used this CapsNet [14] to ensue a spatial attention to enhance the spatial information of the hand image. In this Capsnet, we replace the convolutions layers before the primaries caps by a pretrained model, the Resnet18 [15], which is pretrained on ImageNet. The spatial feature for each image of a clip is computed as follows: $RS_{\mathrm{i}}(x) = R_0(x) \oplus ... \oplus R_{\mathrm{j}}(x) \oplus ... \oplus R_{\mathrm{n}}(x)$ Where $\oplus$ denotes the operation of convolution, RS_{i} is the extracted feature for the clip i and R_{j} is the extractd feature for the image j of the clip i.

Then, we adopt the Capsule Network (CapsNet) architecture, which provides a more efficient way to represent the spatial relationships and hierarchies between the features. The CapsNet uses dynamic routing to learn how to combine the features of different images in a way that captures their mutual dependencies and provides a more robust and accurate representation of the hand. To begin the process, the scalar input values generated by the CNN network are converted to primary capsules within the Capsule Network architecture. The PrimaryCaps layer comprises of N primary capsules, with each capsule consisting of dp-dimensional vectors that encode spatial features derived from the previous convolutional operation. Through this transformation, the input data is converted into a more structured representation that allows for improved hierarchical learning. To scale and constrain the output vectors of the Capsule Network, a non-linear squashing function is applied at the output layer. This function is designed to ensure that the magnitude of the output vector is scaled to a value between zero and one, depending on the length of the vector. By doing so, the output vector can effectively convey meaningful information to the downstream layers of the neural network. Additionally, the non-linear nature of the squashing function facilitates the handling of output vectors with varying lengths and helps preserve the spatial relationships between the features. The formula for the squash function can be expressed using the following equation: $x = \frac{\|x\|^2}{1+\|x\|^2} \frac{x}{\|x\|}$

where x is the vector that we squash. The next layer in the Capsule Network architecture is the dense caps layer, which is responsible for further processing the input data. Initially, an affine transformation is applied to the input vectors using the equation: $\hat{u}_{\mathrm{jli}} = W_{\mathrm{ij}}$. $\hat{u}_{\mathrm{jli}}$ represents the predicted output vector for the dense caps layer, W_{ij} is the weight matrix, and u_{i} is the capsule obtained at the output of the primary capsule layer. This affine transformation allows the dense caps layer to learn more complex and abstract representations of the input data, capturing higher-level features and patterns.

The next step in the Capsule Network is the routing by agreement, which serves to ensure that the output vectors of the dense capsules are properly aligned and combined to form an accurate and comprehensive representation of the input data. Once all the spatial features for each image of the hand have been extracted, the features are concatenated and flattened in preparation for the final processing step.

2.2. Temporal attention module

In our model, inspired from [1], we only use the transformer encoder. This part corresponds to the analysis of the temporal features. First a positional encoding is applied to the spatial features extracted by our Capsnet using this equation:

$$PE_{(\mathrm{pos,2i})} = sin(\frac{pos}{10000^{\frac{2i}{d_{\mathrm{model}}}}})$$
$$PE_{(\mathrm{pos,2i+1})} = cos(\frac{pos}{10000^{\frac{2i+1}{d_{\mathrm{model}}}}})$$

Where pos is the current frame, i is the number of frames in one clip and d_{model} is the number of spatial features for one image. It is a way to incorporate temporal information on the order of the frame and to obtain information on the movement of the hand. The definition of one transformer encoder is: $T(x) = Norm(x + FC(muHead(x)))$

Norm means the normalisation layer, FC is a fully connected layers. muHead corresponds to the multi head attention. This equation shows that we keep the original encoded features of the hand and the temporal features extract by the multi head attention as :

$$muHead(x) = (Attention_1(x) \oplus ... \oplus Attention_{\mathrm{n}}(x)).W^{\mathrm{O}}$$

Where W^{O} is a linear projection from our features to change the representation subspace of our features. The used attention function is defined as: $Attention_{\mathrm{i}}(x) = softmax(\frac{Q_{\mathrm{i}}K_{\mathrm{i}}}{\sqrt{d_{\mathrm{k}}}})V_{\mathrm{i}}$. Where Q is the queries, K is the keys, V is the values, and d_{k} is the number of keys and queries. In practice we compute K, Q and V with a fully connected Layer which take in input x the features of all hand in the clip after the positional encoding. The attention block allow us to analyse the movement of the hand encoded by the positional encoding and extract temporal features which help us to classify the gesture.

Then, an average pooling is used to get only one frame and reduce the number of features. Finally, we have a Fully Connected layers Y to predict the gesture.

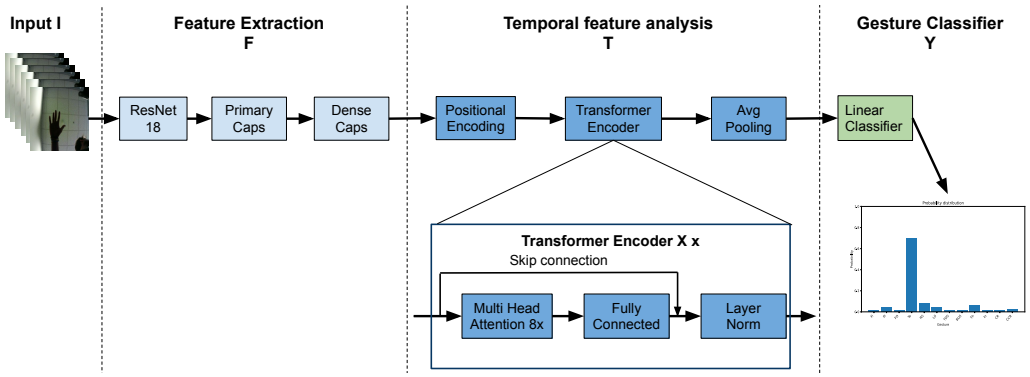

Fig. 1. Overview of the proposed method.

3. EXPERIMENTS AND RESULTS

This section provides details on the dataset used, the hyperparameters of the model, and the implementation particulars. Subsequently, the outcomes associated with various modalities and fusion strategies, as well as a comparison with state-of-the-art approaches, are presented.

3.1. Dataset

Briareo dataset [12] contains dynamic hand gestures used for recognition in automotive settings context. It provides RGB images, depth maps, infrared intensities, and 3D hand joints modalities. In addition, we compute the surface normals like it was presented in [16]. The dataset consists of 12 gesture performed by 40 subjects, each repeating 3 times.
SHREC17 dataset [13] contains 2800 videos of dynamic hand gestures. It comprises 14 classes, which are performed using one finger or the entire hand, thereby offering a total of 28 gestures if the two modalities are considered separately.

3.2. Training model

We employed a fixed-length clip approach of 40 frames for both datasets. To normalize the input data, except for the surface normals which are already normalized and contained in the range [0, 1], each input data was individually normalized to obtain zero mean and unit variance input. The frames were then resized to 224x224 pixels, and random rotation between -15 and 15 degrees was applied for data augmentation to avoid overfitting. Subsequently, random crop was used.We used the ResNet-18 architecture for initialization, with weights pre-trained on ImageNet [15], while the other parts of the architecture were trained from scratch. The categorical cross-entropy loss was minimized using the Adam optimizer, with a learning rate of 1e-4, a scheduler, weight decay of 1e-4, dropout of 0.1 for transformers, and an 8-sample minibatch. The model was trained for 500 epochs, following the official dataset splits. The PyTorch deep learning framework was utilized for training, and the hardware configuration comprised an Intel i9 9900k processor, Nvidia RTX Titan, Nvidia GeForce RTX 3090, and 64 GB of RAM.

3.3. Ablation study

In this study, we investigate two fusion strategies: feature concatenation and probability fusion. Our fusion approach involves the consideration of two or more modalities, specifically RGB, Depth, Normal or IR, with the aim of achieving better results than using a single modality.
Fusion by feature concatenation (FF): After computing spatial and temporal features for each modality, we make fusion in the feature level by concatenating them. Then, three fully connected layers are applied for gesture classification. The weights of the networks trained on individual modalities are frozen for training this fusion strategy.
Fusion by probability (FP) : It is a late fusion strategy that can be defined as: $Y = 1/N. \sum_i^N Y_{\mathrm{i}}$ Where Y is the calculated probability, N is the number of classifier used classifiers and Y_{i} is the probability given by the i^{th} tested classifier.

Table 1 presents the results of both unimodal and multimodal data on the Briareo dataset, whith color, depth IR and computed normasl. Analysis of results using different combinations of these modalities show that he probability fusion (PF) strategy is the more accurate giving best results. We notice that this strategy is also requires less training time. Besides, the best result, regarding selected modality, is given by the PF applied to IR and normals for 97.57%. As shown in table 2, for SHREC17 dataset, depth modality gives better results than normals. The PF applied to bith modalities helps to improve results with a significant increase in accuracy of about 5% for 14 and 28 gestures.

3.4. Comparison with state of the art

We present in Table 3 a comparison of our proposed multimodal method with state-of-the-art approaches. Our approach outperforms state-of-the-art accuracy with 97.57% accuracy

modality	Single modality Accuracy	
Color (C)	93.75	
Depth (D)	90.28	
IR	**97.20**	
Normal (N)	92.36	
modality	PF	FF
C + D	**95.83**	92.36
C + IR	**97.20**	**97.20**
C + N	**94.44**	75
D + IR	95.83	**96.53**
D + N	**96.18**	93.06
IR + N	**97.57**	97.20
C + D + IR	**96.18**	95.14
C + D + IR	**95.49**	93.06
C + IR + N	**97.22**	96.88
D + IR + N	**96.52**	96.18
C + D + IR + N	**96.88**	96.18

Table 1. Our results on Briareo using different modalities.

modality	PF 14 gestures	PF 28 gestures
Depth	83.33	79.29
Normal	82.38	78.33
Depth + Normal	**88.69**	**84.04**

Table 2. Our results on SHREC'17 using different modalities.

Approach	Briareo	SHREC17
C3D [12] IR	87.5	-
D'Eusanio et al.[1] N+IR	97.2	-
D'Eusanio et al.[1] N+D	-	**89.40(14) 88.93(28)**
D'Eusanio et al. [17] D+IR	92.0	-
Chen et al. [18] D+C	94.1	-
Ohn-Bar et al [19] D	-	83.85(14) 76.53(28)
Oreifej et al [20] D	-	78.53(14) 74.03(28)
Key frames [13] D	-	82.90(14) 71.90(28)
Ours (IR+N)	**97.57**	-
Ours (N+D)	-	88.69(14) 84.04(28)

Table 3. Comparison with state of the art approaches

on the Briareo dataset using PF on infrared data and surface normals. For SHREC17 dataset, we compare our approach with the methods that used Depth or image data. We computed the results of D'Eusanio et al [1] on this dataset using the same hyperparameters as they proposed in their work. Our approach achieves comparable results with their approach for 14 gestures, but we observe a decrease in accuracy for 28 gestures. Notably, the fusion of modalities greatly improves the accuracy of our approach. We note that SHREC17 dataset is unbalanced in terms of frame number per sequence which lead to more challenging recognition process.

3.5. Real time application and computational time

We have developed an online hand gesture recognition system for real time application by deploying our approach on on the NVIDIA Jetson TX2 device. The program has been constructed utilizing several software components, including Python 3.8, PyTorch 1.10, Torchvision 0.11.1, and OpenCV 4. As shown in Fig. 2, the developed application is able to capture, visualize and recognize hand gestures trained from the Briareo dataset using an RGB camera. To ensure optimal performance, we propose capture RGB data with a resolution of 640x480 at 30 frames per second, selectively ignoring some frames during the capture process. We also conducted a comparative analysis of the execution time of our approach with D'Eusanio et al. [1] regarding Briareo dataset. As shown in Table 4. The results indicate that our approach offers better execution times. Additionally, the number of parameters in our approach is lower, making it more suitable for deployment in real-time applications.

Fig. 2. Visualization of the developed application.

Approach/Modality	C	D	IR	N
D'Eusanio et al.[1]	295	280	281	330
Ours	203	208	203	205

Table 4. Running Time in ms on Briareo dataset

4. CONCLUSION

In this paper, we present a novel Capsule Transformer Network for video-based hand gesture recognition. The proposed network leverages the benefits of Capsule Networks, which enable the creation of stronger connections between spatial features compared to traditional CNNs, resulting in improved features for the transformer's input which ensured the temporal attention through the gesture sequence. Additionally, Capsule Networks can better capture and process information about symmetrical gestures, which are sometimes confused by CNNs. We evaluate the proposed method using multiple modalities of data from an RDB-D camera, and test different fusion strategies. Our approach achieves state-of-the-art results by extracting features at the image level and aggregating them temporally using the transformer. In future work, we plan to further develop our model for deployment in real-life HCI contexts.

5. REFERENCES

[1] Andrea D'Eusanio, Alessandro Simoni, Stefano Pini, Guido Borghi, Roberto Vezzani, and Rita Cucchiara, "A transformer-based network for dynamic hand gesture recognition," in *2020 International Conference on 3D Vision (3DV)*. IEEE, 2020, pp. 623–632.

[2] Rim Slama, Wael Rabah, and Hazem Wannous, "Str-gcn: Dual spatial graph convolutional network and transformer graph encoder for 3d hand gesture recognition," in *2023 IEEE 17th International Conference on Automatic Face and Gesture Recognition (FG)*. IEEE, 2023, pp. 1–6.

[3] Mohamed Akremi, Rim Slama, and Hedi Tabia, "Spd siamese neural network for skeleton-based hand gesture recognition," in *17th International Conference on Computer Vision Theory and Applications VISAPP 2022)*. SCITEPRESS-Science and Technology Publications, 2022, pp. 394–402.

[4] Munir Oudah, Ali Al-Naji, and Javaan Chahl, "Hand gesture recognition based on computer vision: a review of techniques," *journal of Imaging*, vol. 6, no. 8, pp. 73, 2020.

[5] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[6] Osman Güler and İbrahim Yücedağ, "Hand gesture recognition from 2d images by using convolutional capsule neural networks," *Arabian Journal for Science and Engineering*, vol. 47, no. 2, pp. 1211–1225, 2022.

[7] Theo Voillemin, Hazem Wannous, and Jean-Philippe Vandeborre, "2d deep video capsule network with temporal shift for action recognition," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021, pp. 3513–3519.

[8] Liang Zhang, Guangming Zhu, Lin Mei, Peiyi Shen, Syed Afaq Ali Shah, and Mohammed Bennamoun, "Attention in convolutional lstm for gesture recognition," *Advances in neural information processing systems*, vol. 31, 2018.

[9] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[10] Matej Králik and Marek Šuppa, "Waveglove: Transformer-based hand gesture recognition using multiple inertial sensors," in *2021 29th European Signal Processing Conference (EUSIPCO)*. IEEE, 2021, pp. 1576–1580.

[11] Lianqing Zheng, Jie Bai, Xichan Zhu, Libo Huang, Chewu Shan, Qiong Wu, and Lei Zhang, "Dynamic hand gesture recognition in in-vehicle environment based on fmcw radar and transformer," *Sensors*, vol. 21, no. 19, pp. 6368, 2021.

[12] Fabio Manganaro, Stefano Pini, Guido Borghi, Roberto Vezzani, and Rita Cucchiara, "Hand gestures for the human-car interaction: The briareo dataset," in *International Conference on Image Analysis and Processing*. Springer, 2019, pp. 560–571.

[13] Quentin De Smedt, Hazem Wannous, Jean-Philippe Vandeborre, Joris Guerry, Bertrand Le Saux, and David Filliat, "Shrec'17 track: 3d hand gesture recognition using a depth and skeletal dataset," in *3DOR-10th Eurographics Workshop on 3D Object Retrieval*, 2017, pp. 1–6.

[14] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton, "Dynamic routing between capsules," *Advances in neural information processing systems*, vol. 30, 2017.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[16] Paul J Besl and Ramesh C Jain, "Invariant surface characteristics for 3d object recognition in range images," *Computer vision, graphics, and image processing*, vol. 33, no. 1, pp. 33–80, 1986.

[17] Andrea D'Eusanio, Alessandro Simoni, Stefano Pini, Guido Borghi, Roberto Vezzani, and Rita Cucchiara, "Multimodal hand gesture classification for the human–car interaction," in *Informatics*. MDPI, 2020, vol. 7, p. 31.

[18] Huizhou Chen, Yunan Li, Huijuan Fang, Wentian Xin, Zixiang Lu, and Qiguang Miao, "Multi-scale attention 3d convolutional network for multimodal gesture recognition," *Sensors*, vol. 22, no. 6, pp. 2405, 2022.

[19] Eshed Ohn-Bar and Mohan Trivedi, "Joint angles similarities and hog2 for action recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2013, pp. 465–470.

[20] Omar Oreifej and Zicheng Liu, "Hon4d: Histogram of oriented 4d normals for activity recognition from depth sequences," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 716–723.