J Inf Process Syst, Vol.19, No.6, pp.730~744, December 2023
https://doi.org/10.3745/JIPS.02.0207

ISSN 1976-913X (Print)
ISSN 2092-805X (Electronic)

JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

Egocentric Vision for Human Activity Recognition Using Deep Learning

Malika Douache[1,2,*] and Badra Nawal Benmoussat[1]

Abstract

The topic of this paper is the recognition of human activities using egocentric vision, particularly captured by body-worn cameras, which could be helpful for video surveillance, automatic search and video indexing. This being the case, it could also be helpful in assistance to elderly and frail persons for revolutionizing and improving their lives. The process throws up the task of human activities recognition remaining problematic, because of the important variations, where it is realized through the use of an external device, similar to a robot, as a personal assistant. The inferred information is used both online to assist the person, and offline to support the personal assistant. With our proposed method being robust against the various factors of variability problem in action executions, the major purpose of this paper is to perform an efficient and simple recognition method from egocentric camera data only using convolutional neural network and deep learning. In terms of accuracy improvement, simulation results outperform the current state of the art by a significant margin of 61% when using egocentric camera data only, more than 44% when using egocentric camera and several stationary cameras data and more than 12% when using both inertial measurement unit (IMU) and egocentric camera data.

Keywords

Convolutional Neural Network, Deep Learning, Egocentric Vision (or First-Person Vision), Human Activity Recognition, Image Classification, Inertial Measurement Unit (IMU)

1. Introduction

Thanks to advancements in wearable technology, the human activity recognition from egocentric vision often mentioned as first-person vision, provides much potential research. These advancements give the possibility to detect the surroundings and the subject's activities from his viewpoints. Fig. 1 presents some examples of wearable equipment.

In a process that can be used in a variety of settings, as well as mobile/ambient assisted life, assistant for personal health care, interaction between humans and computers, industrial environments, observation systems and smart buildings, the task of this type of recognition is to identify what action is being performed in a given egocentric video segment. We can also use it to locate visitors to a cultural or outside natural site, analyze their activity automatically to better understand their preferences, inform them about where they are and what they can view.

※ This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (http://creativecommons.org/licenses/by-nc/3.0/) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Manuscript received May 27, 2022; first revision August 18, 2022; accepted September 9, 2022.

* **Corresponding Author:** Malika Douache (malika.douache@univ-usto.dz)
[1] Automation, Vision and Intelligent Systems Control Laboratory, University of Sciences and Technology of Oran Mohamed-Boudiaf (USTOMB), Oran, Algeria (malika.douache@univ-usto.dz, badranawal.benmoussat@univ-usto.dz)
[2] National Institute of Telecommunications, Information and Communication Technologies (INTTIC), Oran, Algeria (malika.douache@univ-usto.dz)
Current affiliation for author, Malika Douache, is National Higher School of Telecommunications, Information and Communication Technologies "ENSTTIC", Oran, Algeria.

Copyright© 2023 KIPS

Fig. 1. Examples of wearable equipment.

For this purpose, a variety of methodologies has been developed. They may be classified into two categories: machine learning algorithms and neural network techniques [1,2]. In summary, the first category includes decision trees, support vector machine, hidden Markov models, and k-nearest neighbor method. The second one includes artificial neural network, recurrent neural network (RNN), and convolutional neural network (CNN) which is the most widely used deep learning algorithm. Owing to the appearance of big data and increasingly powerful computing components, the power and data-intensive deep learning algorithms have overtaken most other methods.

That is why, in this paper, we use the deep learning to egocentric human activity recognition. It tends to work well with a significant volume of data, while more traditional machine learning models, which are powerful programming tools allowing in particular, the recognition of images by automatically attributing to each one provided as input, a label corresponding to its membership class, stop improving after a saturation point.

Our investigations were in the context of kitchens; in particular, on the Carnegie Mellon University Multi-Modal Activity (CMU-MMAC) database [3]. We are collecting data from only the ego-vision videos, where some subjects have been captured cooking five different recipes: brownies, pizza, sandwich, salad, and scrambled eggs (Fig. 2).

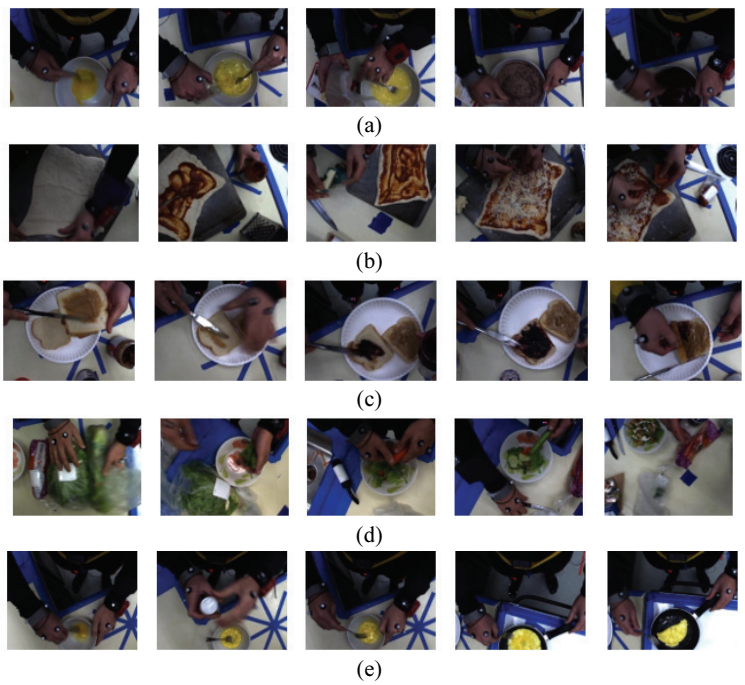

Fig. 2. Egocentric video database frames from the CMU-MMAC dataset (http://kitchen.cs.cmu.edu/main.php): preparing brownie (a), pizza (b), sandwich (c), salad (d) and scrambled eggs (e).

Our proposed method is simple and efficient. It allows the increase of the accuracy obtained in the state-of-the-art approaches, for the same database [3], compared to either methods using only the egocentric videos, or those combining egocentric videos and inertial measurement units (IMUs).

This document is organized in the following way: Section 2 discusses the state of the art, Section 3 resumes the methods used, and Section 4 provides a description of the dataset utilized. Sections 5 and 6 present experimentations and future objectives respectively. Finally, Section 7 gives some conclusions.

2. Related Work

Widely studied in previous research, egocentric action recognition uses a variety of sensor modalities. In this section, we give a non-exhaustive summary of previously published works in a chronological order.

In [4], the authors used a wearable camera and IMUs from the CMU-MMAC database [3] to investigate first-person perception. They conducted a supervised and unsupervised temporally segmenting of human motion into actions and classify activity. Fathi et al. [5], showed that combined modeling of activities, actions, and objects improves performance than when they are analyzed separately. Later in [6], by using two new datasets including egocentric videos of daily activities and gaze, they showed improvements in action recognition rates and gaze prediction accuracy compared to state-of-the-art approaches. In [7], the authors develop several models of daily activities based on object-centric representations.

Afterward, Ryoo and Matthies [8] were looking into multichannel kernels as a way to combine global and local motion data, describing a new activity learning/recognition approach that takes temporal structures presented in first-person activity videos into account. Next, Song et al. [9] used Google Glass to create an egocentric video dataset called LENA (Life-logging EgoceNtric Activities). They used LENA to evaluate the state-of-the-art activity recognition and looked at how popular descriptors performed in egocentric activity recognition. Later in [10], the authors use a bi-linear maximum margin model to find the appropriate camera important factors to maximize action prediction accuracy. Ryoo et al. [11] introduced a model for temporally pool features in order to recognize egocentric actions with [10], using the CMU-MMAC database [3]. In [12], the authors evaluated how different egocentric cues (such as gaze, the presence of hands, objects, and head movement) can be employed to perform the task.

Thereafter, Ma et al. [13] created a deep learning architecture that enables them to combine various egocentric-based features to identify actions. Otherwise, Song et al. [14] to solve the egocentric activity recognition challenge, suggested combining video and temporal improved sensor characteristics using the Fisher kernel framework, proposing, in [15], a multimodal multi-stream deep learning system that uses both video and sensor data. Singh et al. [16] proposed CNNs for classification of wearer's actions, by recording hand stance, head motion, and saliency map utilizing egocentric cues. Moreover, in [17], the authors explored CNN and temporal segment networks, using hands movements and what object is being manipulated for analyzing first-person action.

Furthermore, Khalid et al. [18] begin by surveying all existing egocentric datasets. The authors then include the Swain's distance into a dynamic time warping method and utilize it to construct an algorithm that employs visual lifelogs to automatically classify daily activities. Singh et al. [19] used improved dense trajectories to solve the difficulty of recognizing egocentric actions.

In another field, with the procedure being then repeated for the duration of the video, Liu et al. [20] applied a beam search to recognize the fluent item in each frame concurrently. Possas et al. [21] developed a model-free reinforcement learning technique for learning energy-aware rules that maximize the use of low-energy cost predictors while maintaining competitive accuracy levels. They demonstrated that a policy developed on an egocentric dataset may efficiently tradeoff energy expenditure and accuracy by utilizing the synergy between motion and vision sensors. Li et al. [22] introduced a revolutionary deep model for simultaneous gaze estimation and egocentric action identification.

In [23], the authors developed a spatial attention method that allows the CNN to pay attention to regions containing objects that are connected to the activity, doing this before using them for spatiotemporal encoding of video with a long short-term memory (LSTM). Later in [24], they proposed long short-term attention as a technique for focusing on features from relevant spatial parts while attention is followed smoothly over a video sequence. In [25], a multi-modal fusion architecture has been proposed. It has been trained from beginning to end to outperform individual modalities and late fusion of modalities. Thereafter, Lu and Velipasalar [26], by employing 10 videos representing five different subjects (two videos per subject) for training and testing, developed and implemented a genetic algorithm-based method for optimizing multiple parameters of their network architecture autonomously and simultaneously. They used the CMU-MMAC database [3].

On the other hand, Diete and Stuckenschmidt [27] investigate the transfer of deep learning models in vision to models for activity recognition and object detection by combining inertial and video features. In [28], the authors deal with the issue of egocentric action anticipation. The rolling-unrolling (RU) LSTM was presented as a learning architecture for anticipating actions from egocentric videos. In the same context, Rodin et al. [29] proposed ideas on how to improve the quality of predictions and reviewed the current approaches for action anticipation from egocentric video. By introducing and benchmarking different changes based on some objectives cited in their paper, they propose to extend the RU-LSTM model [28].

Besides, Min and Corso [30] presented a probabilistic method for integrating human gaze to spatio-temporal attention to recognize egocentric activity. In another issue, Ragusa et al. [31] proposed a new dataset named "MECCANO", establishing the egocentric human-object interaction (EHOI) detection task and conducting baseline experiments to demonstrate the dataset's potential, while the dataset was focused to exploring EHOIs in an industrial setting.

In [2], the authors used first-person camera data, from the CMU-MMAC database [3] and, by considering only three actions (or recipes; brownies, scrambled eggs, and sandwiches) performed a deep learning to extract and recognize features. This was instead of considering all five actions present in this database.

The methods used in this research are described in the following section.

3. Methods Used

In this work, we exploit a specific type of deep learning, which is the CNN. It is considered one of the most efficient deep learning algorithms because of its performance in image classification and action recognition [32]. The following is a quick description of this latter.

3.1 Convolutional Neural Network

A deep CNN model is composed of a limited number of processing layers that could learn different characteristics of input data (for example, image). Description of these different layers is shown in Fig. 3.

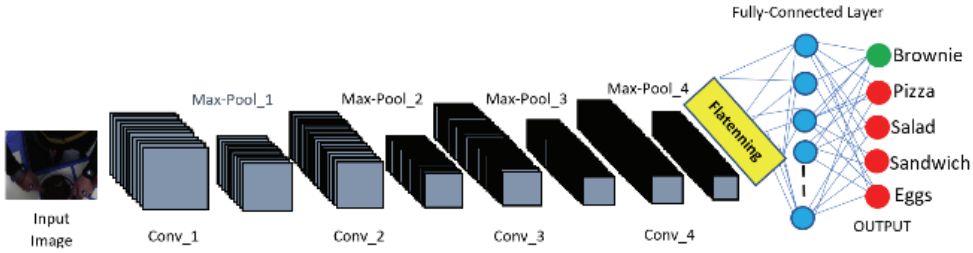

Fig. 3. Example of a CNN processing for image classification of a brownie preparation.

3.1.1 Layers in CNN

With each one executing different functions to translate one volume, a CNN is composed from three basic layers, convolutional layer, pooling layer, and flattening and fully connected layers.

Convolutional layer

The most crucial layer in any CNN design is the convolutional layer. It is composed of a set of convolutional kernels (also known as filters) that are convolved with the input image (N-dimensional metrics) by a simple mathematical convolution, to produce an output feature map.

The convolutional layer is characterized by the following hyperparameters: the first one is the size and the number of filters. The second one is the stride value "S" with which we drag the window corresponding to the filter on the image, and the third one is the zero-padding "P". In this last hyperparameter, and with the padding of pixels being necessary in order to accentuate the input image's border size information, we add a black (shades of gray = 0) outline to the input image with a layer of thickness "P" pixels. However, the border side features are erased away too rapidly if no padding is used.

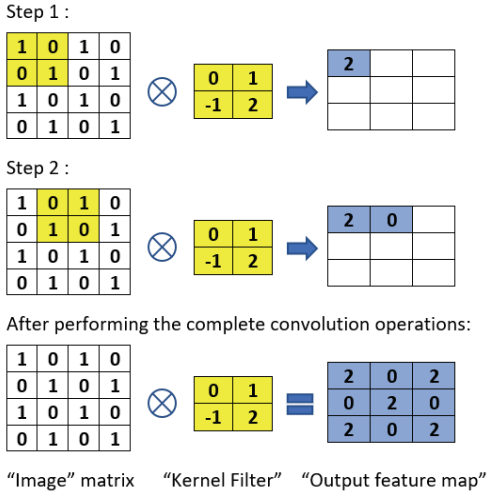

Fig. 4. Example of 2D convolution with no padding to the input image and a kernel stride of 1.

Fig. 4 presents an example of 2D convolution with no padding to the input image and a kernel stride of 1, while Fig. 5 is showing an example of 2-D convolution with zero-padding, "P" = 1 for the input image and a kernel stride of 3.

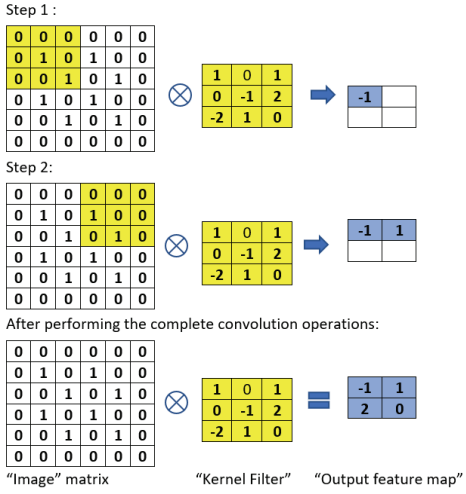

Fig. 5. Example of 2D convolution with zero-padding P=1 for the input image and a kernel stride of 3.

Pooling layer

After convolution operations, the pooling layer is utilized to sub-sample the output feature maps in order to reduce the convolved feature size. This is useful for obtaining dominant features that are invariant in terms of position and rotation [33].

This layer has two hyperparameters: The pool size "F", used to split the image into square cells of size F×F pixels and the stride value, which is defined as a vector, containing two positive integers [a b], with "a" representing the vertical step size and "b" representing the horizontal step size. The stride can be set as a scalar when this layer is created to utilize the same step size value for both vertical and horizontal dimensions. An example of a max pooling technique is illustrated in Fig. 6. The pooling operated in this case replace all values in the cell of 2×2 size by the max value in the mask.

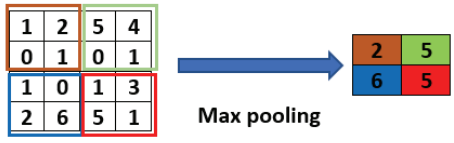

Fig. 6. Example of a max pooling technique with stride value of 2 and pool size F=2.

Flattening and fully connected layers

Flattening and fully connected layers are the last part of every CNN architecture (Fig. 7). Flattening is converting the data into a 1-dimensional array to inject them into the fully connected layers. The term, fully connected, means that every neuron inside a layer is linked to every neuron from the preceding one. This final layer of fully connected layers and the output of the CNN is the classifier, where each neuron assigns to the image a probability value of belonging to one class among the remaining possible classes.

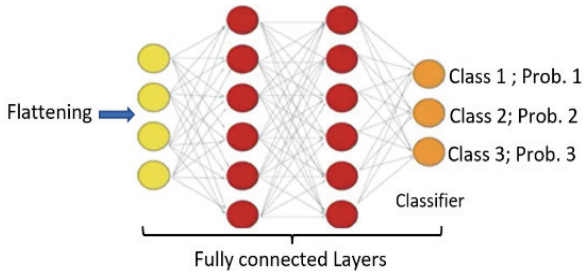

Fig. 7. Example of flattening and fully connected layers.

3.1.2 ReLU activation function

The rectifier linear unit (ReLU) activation function (Fig. 8) is widely used in convolutional neural networks, between the convolutional and pooling layer, because it requires less computation load compared to other activation functions used in this field.

In our proposed CNN deep learning method, we employ the ReLU activation function presented by Eq. (1):

$$f(x)_{ReLU} = max(0, x). \tag{1}$$

3.1.3 Epoch

An epoch is defined as one cycle during the entire training dataset. Although there is no guarantee that increasing the number of epochs will improve the network convergence, generally, it takes several epochs to make the training CNN. It is a way to review the previous data and readjust parameters of the training model.

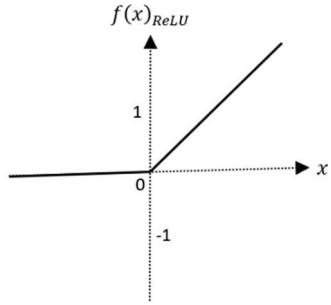

Fig. 8. ReLU activation function.

3.1.4 Evaluation metric

As a metric for evaluation, we employ recognition accuracy. It explains how the model works in all classes. This metric may be beneficial when all classes are equally important. It is the ratio between the number of correct predictions and the total number of predictions.

$$Acuuracy = \frac{True_{positive}+True_{negative}}{True_{positive}+True_{negative}+False_{positive}+False_{negative}}. \tag{2}$$

The dataset utilized in this paper is described in the following section.

4. The Dataset Utilized

In our experiments, inclusive a multimodal measure about the human activity of persons executing actions related to food preparation and cooking, we use the CMU-MMAC database [3]. With 25 subjects have been captured preparing five different recipes—sandwich, salad, brownie, pizza, and scrambled eggs (Fig. 2), this database was generated in the Motion Capture Lab at Carnegie Mellon University.

Video, audio, motion capture, and inertial measurement were recorded using cameras, microphones, a Vicon motion capture system, and wired/Bluetooth IMUs, respectively. A BodyMedia and an eWatch were employed as wearable gadgets. The detailed characteristics of each equipment are given in [3]. In addition to an auxiliary dataset including anomalous situations being available, the database includes a main dataset where subjects are cooking five recipes. In this context, three subjects are cooking while some atypical situations occur (falling dishes, fire and smoke, distractions, etc.).

In the proposed method, we are exploiting only the first-person video from the main dataset, for each subject cooking the five different recipes cited above.

In next section, we present our experimentations.

5. Experimentations

In this section, before the specifications of the training options for the proposed CNN model being given, we firstly describe materials and software employed in this study, the pre-processing of the CMU-MMAC dataset, as well as the architecture of the suggested CNN model for deep learning. Next, results and discussions follow, respectively.

5.1 Materials and Software

We present in the following the characteristics of the computer, the digital calculation and programming platform, plus the video file frame extraction tool [34] used in this work. Table 1 summarizes these characteristics.

Table 1. Materials and software used

Computer	Digital calculation and programming platform	Free Video to JPG converter [34]
Processor: Intel Core i7-8750H CPU @2.20 GHz 2.21 GHz RAM: 16.0 Go Operating system: Window 10, 64 bits	MATLAB	Editor: DVDVideoSoft Limited v5.0.101 build 201 (last version)

5.2 Pre-processing of CMU-MMAC Dataset

From the CMU-MMAC database [3], we generate a new one containing five labels or classes called, sandwich, salad, brownie, pizza and scrambled eggs, according to the five prepared recipes.

In each label, we put the ego-videos of the different subjects, which have been recorded cooking. Then, we use the Free Video to JPG converter [34], which is a software, dedicated to frames extraction from

videos (Fig. 9). With the total number of frames obtained from each ego-video providing us with the necessary amount of information, the process conducts a video temporal sampling, where the sampling period is a parameter to be chosen. In our case, we take one frame in every half second.

Finally, we arrive to recognize the activity being carried out and therefore which recipe is being prepared by using the proposed CNN, illustrated in the next section, to classify every test input image in its corresponding class.

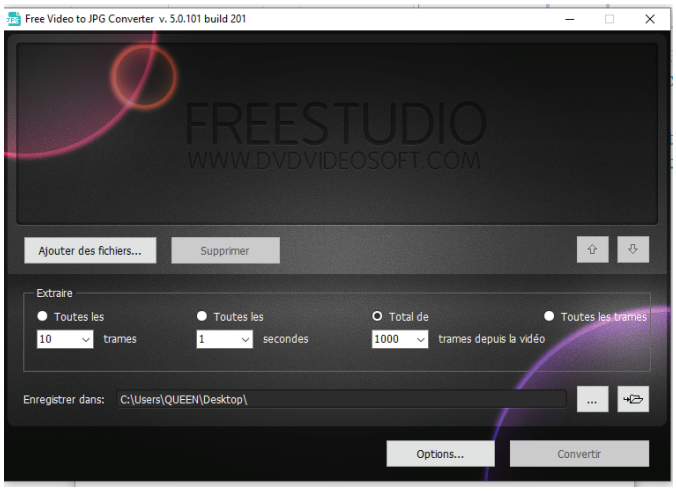

Fig. 9. Free Video to JPG Converter interface (https://www.dvdvideosoft.com/products/dvd/Free-Video-to-JPG-Converter.htm).

5.3 Architecture of the Proposed Deep Learning CNN Model

Fig. 3 illustrates the architecture of our proposed deep learning CNN model. It is composed from four convolutional layers and four max pooling layers. Table 2 summarizes the corresponding hyperparameters values.

Table 2. Architecture details of the proposed deep learning CNN model

Layer	Parameter
First convolution layer	32 filters used, filter size = 8×8, zero-padding P=2
Second convolution layer	64 filters used, filter size = 3×3, zero-padding P=2
Third convolution layer	128 filters used, filter size = 5×5, zero-padding P=2.
Fourth convolution layer	256 filters used, filter size = 5×5, zero-padding P=2.
Max pooling layers 1, 2, 3, and 4	pool size [2 2], stride [3 3].

5.4 Specifications of the Training Options for the Proposed Deep Learning CNN Model

Using Stochastic Gradient Descent with Momentum, we create a set of options for training the network [35,36]. This method helps the network to accelerate gradient vectors in the right directions and avoid local minima.

Before giving the corresponding values of the training options in Table 3, we present some definitions [35]:

- Initial learning rate is a positive scalar, if it is too low, the training will take a long time. Whereas, if this one is too high, then training may produce unsatisfactory results.
- Size of the mini-batch utilized for each training iteration, defined as a positive integer.
- A mini-batch is a subset of the training set used to calculate the loss function's gradient and adjust the weights.
- Shuffle: Data shuffle option, which might be ones that follow:

  "once": Before training, shuffle the training and validation data once.

  "never": Do not shuffle the data.

  "every-epoch": Before each training epoch shuffle the training data and before each network validation, shuffle the validation data. To prevent discarding the same data every epoch, set the shuffle option to "every-epoch".
- Validation Frequency is a positive integer that represents the frequency of network validation in number of iterations.

Table 3. Training options values of the proposed method

Variable	Value
Momentum	0.9
Initial learning rate	0.02
Max epochs	20, 30, 40, 50, and 60
Size of mini-batch	64 observations at each iteration
Shuffle	Option: "every-epoch"
Validation frequency	20
Percentage of training image	90%, 85%, and 80%
Percentage of test images	10%, 15%, and 20%

5.5 Results

To test our CNN deep learning model performances, we chose to consider, for the preparation of each recipe, one, two, three, four, five and six subjects. For each case, we varied the number of epochs considering 20, 30, 40, 50, 60 and 70 epochs and taking different percentages (%) of training images equal to 80, 85 and 90. The accuracy of egocentric activity recognition of these different cases is shown in Table 4.

These results will be discussed in the next section.

5.6 Discussion

One can see from table 4 that a maximum accuracy of 99.41% is reached for thirty epochs in the case of one subject per recipe with a percentage of training images equal to 90%. Then, for the remaining considered cases, the maximum of this rate varies between 96.45% and 99.13% according to the epochs number choices which range from 30 to 70 epochs and the considered percentage of training images.

This means that our proposed method almost allows solving the variability problem in action executions. Indeed, subjects, in CMU-MMAC database [3], were not provided instructions on how to conduct the recipe [4]. In our proposed method, we almost got over this problem.

Our model remains competitive and efficient whatever the number of considered subjects and for all the recipes present in this database [3]. For comparison purpose of the proposed method, we selected related works, which used the CMU-MMAC dataset and the same evaluation metric, the accuracy rate.

Table 4. Accuracy of the proposed method in various cases

	Training images (%)	Accuracy (%)					
		Epoch=20	Epoch=30	Epoch=40	Epoch=50	Epoch=60	Epoch=70
One subject per recipe (5 videos)	90	96.46	**99.41**	98.82	97.84	97.05	98.82
	85	97.04	**98.42**	98.22	97.41	96.74	98.03
	80	95.71	96.60	**98.52**	97.63	97.93	97.93
Two subjects per recipe (10 videos)	90	97.02	98.65	97.29	98.38	**98.92**	98.65
	85	96.48	98.11	**98.74**	97.29	98.20	97.66
	80	95.61	97.84	97.03	97.70	98.24	**98.65**
Three subjects per recipe (15 videos)	90	97.30	**99.13**	98.07	99.04	98.46	97.30
	85	94.29	97.05	96.28	**98.65**	97.43	97.43
	80	94.27	96.15	96.29	96.25	**98.08**	96.73
Four subjects per recipe (20 videos)	90	95.37	96.94	97.65	97.43	**98.29**	98.15
	85	94.01	96.77	97.34	**98.19**	97.53	97.43
	80	93.94	95.58	96.19	96.51	96.86	**97.72**
Five subjects per recipe (25 videos)	90	95.77	**98.78**	97.80	97.57	98.72	98.03
	85	95.36	97.18	97.84	**98.22**	97.64	97.68
	80	93.59	94.90	96.38	96.35	97.28	**97.39**
Six subjects per recipe (30 videos)	90	94.25	97.87	97.96	98.24	**98.42**	97.77
	85	94.93	97.09	97.03	97.40	97.28	**97.65**
	80	93.72	95.46	96.10	**96.45**	96.38	96.43

The bold font indicates the best performance in each case.

As can be observed in Table 5, the accuracy of the proposed method is outperforming that of [4] by 41.61%. Comparing with [10], using egocentric camera data only for both cases, and using data from egocentric camera with multiple static cameras, the given accuracy is less than ours by 61.49% and 44.79%, respectively. Finally, considering [26], using the genetic algorithms, the given accuracy rate is less than ours by 12.77%.

The proposed method is already better in terms of accuracy as shown in Table 5. It presents a global recognition without any exception or constraints. As it is shown in table 4, we considered 5, 10, 15, 20, 25 and 30 videos representing one, two, three, four, five and six different subjects in each recipe preparation from the five recipes presented in the database used. The proposed method works without any exception, while maintaining a maximum of accuracy rate between 96.45% and 99.41% depending on the considered case. If there were other recipes, we could integrate them into our proposed method and recognize them very easily without any problem or obligation.

Table 5. Comparison of proposed method versus different approaches using CMU-MMAC dataset

Sensor modality	Year	Method	Accuracy (%)
Egocentric camera and IMU	2009	Temporal segmentation and activity classification [4]	57.80
Multiple static cameras + egocentric camera data	2014	A bi-linear max margin model [10]:	
		-Using egocentric camera data only	37.92
		-Using data from the multiple static cameras and egocentric camera data	54.62
Egocentric camera and IMU	2019	Genetic algorithm [26]	86.64
Egocentric camera only (proposed)	2023	Deep learning CNN (proposed)	**99.41**

The bold font indicates the best performance in each case.

The proposed algorithm is simple and easy to apply, it consists of taking a few frames by sampling an egocentric video only, making a classification, and recognizing the activity in question. The sampling done every 0.5 seconds allows getting closer to real-time activity recognition.

On the other hand, Soran et al. [10] used both egocentric and multiple static cameras to perform their method, studies in [2], [4], and [26], for example, used both the first-person camera data and IMU to extract actions from different activities before applying their proposed methods. The computation load of the proposed method is almost that required by the deep learning algorithm.

6. Future Objectives

In future research, using the same CMU-MMAC database, the following situations are targeted:

- Anomalous situations, which can occur (fire and smoke, falling dishes, distractions, etc.). Here, using a deep learning to detect such cases could be useful for intervention to help or rescue.
- Predefined situations, where the subjects follow a weekly cooking program. Hence, while knowing the recipe cooked today, one can anticipate that of tomorrow. This in turn can be useful to check the availability of all necessary ingredients, or simply make a reminder. This could be achieved using RNN model.

Other suggestions for future works are to use the MECCANO dataset, to investigate human-object interactions in the industrial context. To detect the current action in a production chain is a matter, and also to then anticipate the next one. Thus, a checking by recognition process using RNN deep learning could be conducted, and a decision is made. If the next anticipated action is executed correctly, no intervention is needed, otherwise, an error message is triggered to rectify or resume the action.

7. Conclusion

We have presented a simple and efficient classification method using egocentric camera data only from the CMU-MMAC database. The data-used reduction accelerates the process of human activities recognition. Then, we extract frames by temporal sampling of egocentric videos by taking one frame every half second to get closer to real-time activity recognition. After that, we prepared a new database containing five labels according to the five prepared recipes of database [3]. On this new database, we applied a classification using our proposed deep learning CNN algorithm.

The exploitation of this algorithm proved its effectiveness in recognizing the activities in question with a very satisfactory accuracy equal to 99.41% when one subject was performing the five recipes. We have a maximum of accuracy varying between 96.45% and 99.13% when many subjects were preparing these recipes each in his or her own way. It is important to notice that proposed method remains effective with whatever egocentric video data and the manner in which the subjects carry out their activities. The accuracy of the proposed method exceeds [4] by 41.61%. When compared to [10], the accuracy provided is less than ours by 61.49% and 44.79% in the cases of data from egocentric cameras alone and from egocentric cameras combined with multiple static cameras, respectively. In consideration of [26], the accuracy rate provided by the genetic algorithms is 12.77% lower than ours.

Acknowledgement

The data used in this paper was obtained from kitchen.cs.cmu.edu and the data collection was funded in part by the National Science Foundation (Grant No. EEEC-0540865).

References

[1] C. Jobanputra, J. Bavishi, and N. Doshi, "Human activity recognition: a survey," *Procedia Computer Science*, vol. 155, pp. 698-703, 2019. https://doi.org/10.1016/j.procs.2019.08.100

[2] T. Alhersh, H. Stuckenschmidt, A. U. Rehman, and S. B. Belhaouari, "Learning human activity from visual data using deep learning," *IEEE Access*, vol. 9, pp. 106245-106253, 2021. https://doi.org/10.1109/ACCESS. 2021.3099567

[3] Carnegie Mellon University, 'CMU-Multimodal Activity (CMU-MMAC) database," 2010 [Online]. Available: http://kitchen.cs.cmu.edu/main.php.

[4] E. H. Spriggs, F. De La Torre, and M. Hebert, "Temporal segmentation and activity classification from first-person sensing," in *Proceedings of 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, Miami, FL, 2009, pp. 17-24. https://doi.org/10.1109/CVPRW.2009.5204354

[5] A. Fathi, A. Farhadi, and J. M. Rehg, "Understanding egocentric activities," in *Proceedings of 2011 International Conference on Computer Vision*, Barcelona, Spain, 2011, pp. 407-414. https://doi.org/10.1109/ ICCV.2011.6126269

[6] A. Fathi, Y. Li, and J. Rehg, "Learning to recognize daily actions using gaze," in *Computer Vision – ECCV 2012*. Heidelberg, Germany: Springer, 2012, pp. 314-327. https://doi.org/10.1007/978-3-642-33718-5_23

[7] H. Pirsiavash and D. Ramanan, "Detecting activities of daily living in first-person camera views," in *Proceedings of 2012 IEEE Conference on Computer Vision and Pattern Recognition*, Providence, RI, 2012, pp. 2847-2854. https://doi.org/10.1109/CVPR.2012.6248010

[8] M. S. Ryoo and L. Matthies, "First-person activity recognition: What are they doing to me?," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Portland, OR, 2013, pp. 2730-2737. https://doi.org/10.1109/CVPR.2013.352

[9] S. Song, V. Chandrasekhar, N. M. Cheung, S. Narayan, L. Li, and J. H. Lim, "Activity recognition in egocentric life logging videos," in *Computer Vision – ACCV 2014 Workshops*. Cham, Switzerland: Springer, 2014, pp. 445-458. https://doi.org/10.1007/978-3-319-16634-6_33

[10] B. Soran, A. Farhadi, and L. Shapiro, "Action recognition in the presence of one egocentric and multiple static cameras," in *Computer Vision - ACCV 2014*. Cham, Switzerland: Springer, 2015, pp. 178-193. https://doi.org/10.1007/978-3-319-16814-2_12

[11] M. S. Ryoo, B. Rothrock, and L. Matthies, "Pooled motion features for first-person videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, 2015, pp. 896-904. https://doi.org/10.1109/CVPR.2015.7298691

[12] Y. Li, Z. Ye, and J. M. Rehg, "Delving into egocentric actions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, 2015, pp. 287-295. https://doi.org/10.1109/CVPR. 2015.7298625

[13] M. Ma, H. Fan, and K. M. Kitani, "Going deeper into first-person activity recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, 2016, pp. 1894-1903. https://doi.org/10.1109/CVPR.2016.209

[14] S. Song, N. M. Cheung, V. Chandrasekhar, B. Mandal, and J. Liri, "Egocentric activity recognition with multimodal fisher vector," in *Proceedings of 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Shanghai, China, 2016, pp. 2717-2721. https://doi.org/10.1109/ICASSP.2016.7472171

[15] S. Song, V. Chandrasekhar, B. Mandal, L. Li, J. H. Lim, G. Sateesh Babu, P. P. San, and N. M. Cheung, "Multimodal multi-stream deep learning for egocentric activity recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, Las Vegas, NV, 2016, pp. 24-31. https://doi.org/10.1109/CVPRW.2016.54

[16] S. Singh, C. Arora, and C. V. Jawahar, "First person action recognition using deep learned descriptors," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, 2016, pp. 2620-2628. https://doi.org/10.1109/CVPR.2016.287

[17] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool, "Temporal segment networks: towards good practices for deep action recognition," in *Computer Vision – ECCV 2016*. Cham, Switzerland: Springer, 2016, pp. 20-36. https://doi.org/10.1007/978-3-319-46484-8_2

[18] E. A. Khalid, A. Hamid, A. Brahim, and O. Mohammed, "A survey of activity recognition in egocentric lifelogging datasets," in *Proceedings of 2017 International Conference on Wireless Technologies, Embedded and Intelligent Systems (WITS)*, Fez, Morocco, 2017, pp. 1-8. https://doi.org/10.1109/WITS.2017.7934659

[19] S. Singh, C. Arora, and C. V. Jawahar, "Trajectory aligned features for first person action recognition," *Pattern Recognition*, vol. 62, pp. 45-55, 2017. https://doi.org/10.1016/j.patcog.2016.07.031

[20] Y. Liu, P. Wei, and S. C. Zhu, "Jointly recognizing object fluents and tasks in egocentric videos," in *Proceedings of the IEEE International Conference on Computer Vision*, Venice, Italy, 2017, pp. 2924-2932. https://doi.org/10.1109/ICCV.2017.318

[21] R. Possas, S. P. Caceres, and F. Ramos, "Egocentric activity recognition on a budget," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, 2018, pp. 5967-5976. https://doi.org/10.1109/CVPR.2018.00625

[22] Y. Li, M. Liu, and J. M. Rehg, "In the eye of beholder: joint learning of gaze and actions in first person video," in *Computer Vision – ECCV 2018*. Cham, Switzerland: Springer, 2018, pp. 619-635. https://doi.org/10. 1007/978-3-030-01228-1_38

[23] S. Sudhakaran and O. Lanz, "Attention is all we need: nailing down object-centric attention for egocentric activity recognition," 2018 [Online]. Available: https://arxiv.org/abs/1807.11794.

[24] S. Sudhakaran, S. Escalera, and O. Lanz, "LSTA: long short-term attention for egocentric action recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Long Beach, CA,2019, pp. 9954-9963. https://doi.org/10.1109/CVPR.2019.01019

[25] E. Kazakos, A. Nagrani, A. Zisserman, and D. Damen, "EPIC-fusion: audio-visual temporal binding for egocentric action recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, Seoul, South Korea, 2019, pp. 5492-5501. https://doi.org/10.1109/ICCV.2019.00559

[26] Y. Lu and S. Velipasalar, "Autonomous human activity classification from ego-vision camera and accelerometer data," 2019 [Online]. Available: https://arxiv.org/abs/1905.13533.

[27] A. Diete and H. Stuckenschmidt, "Fusing object information and inertial data for activity recognition," *Sensors*, vol. 19, no. 19, article no. 4119, 2019. https://doi.org/10.3390/s19194119

[28] A. Furnari and G. M. Farinella, "Rolling-unrolling LSTMs for action anticipation from first-person video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 11, pp. 4021-4036, 2021. https://doi.org/10.1109/TPAMI.2020.2992889

[29] I. Rodin, A. Furnari, D. Mavroeidis, and G. M. Farinella, "Scene understanding and interaction anticipation from first person vision," in *Proceedings of the 1st Workshop on Smart Personal Health Interfaces co-located with 25th International Conference on Intelligent User Interfaces*, Cagliari, Italy, 2020, pp. 78-83.

[30] K. Min and J. J. Corso, "Integrating human gaze into attention for egocentric activity recognition," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, Waikoloa, HI, 2021, pp. 1069-1078. https://doi.org/10.1109/wacv48630.2021.00111

[31] F. Ragusa, A. Furnari, S. Livatino, and G. M. Farinella, "The MECCANO dataset: understanding human-object interactions from egocentric videos in an industrial-like domain," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, Waikoloa, HI, 2021, pp. 1568-1577. https://doi.org/ 10.1109/wacv48630.2021.00161

[32] A. Ghosh, A. Sufian, F. Sultana, A. Chakrabarti, and D. De, "Fundamental concepts of convolutional neural network," in *Recent Trends and Advances in Artificial Intelligence and Internet of Things*. Cham, Switzerland: Springer, 2020, pp. 519-567. https://doi.org/10.1007/978-3-030-32644-9_36

[33] S. Saha, "A comprehensive guide to convolutional neural networks: the ELI5 way," 2018 [Online]. Available: https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53.

[34] DVDVideoSoft, "Free Video to JPG converter," 2022 [Online], Available: https://www.dvdvideosoft.com/products/dvd/Free-Video-to-JPG-Converter.htm.

[35] MathWorks, "TrainingOptionsSGDM: training options for stochastic gradient descent with momentum," c2023 [Online]. Available: https://fr.mathworks.com/help/deeplearning/ref/nnet.cnn.trainingoptionssgdm.html;jsessionid=4a2aaa96a2ed0eec48f9cfd48951.

[36] S. Shi, "On the hyperparameters in stochastic gradient descent with momentum," 2021 [Online]. Available: https://arxiv.org/abs/2108.03947.

Malika Douache **https://orcid.org/0000-0002-3482-3895**

She received the State Engineer Diploma in Telecommunications in 2000 and the master's degree in telecommunications, information and communication technologies option, in 2007, from National Institute of Telecommunications, Information and Communication Technologies (INTTIC), Oran, Algeria. Her current position is a master assistant, since 2010 and until now, at INTTIC. She is also, a Ph.D. candidate at Faculty of Electrical Engineering, University of Sciences and Technology of Oran Mohamed-Boudiaf (USTOMB), Algeria. Her current research interests include egocentric vision, artificial intelligence, technologies of information and communication.

Badra Nawal Benmoussat **https://orcid.org/0000-0001-8584-0911**

She received the Diploma of Electrical Engineering Degree in 1996, the master's degree in Signals and systems in 2000 and the Ph.D. degree in Signals filtering and reconstruction in 2006 from the University of Sciences and Technology of Oran Mohamed-Boudiaf, Algeria. She obtained a postdoctoral diploma, habilitation of conducting research, in Electronics Engineering in 2010. Actually, she is an associate professor at Automation Department, University of Sciences and Technology of Oran Mohamed-Boudiaf. Her research includes egocentric vision, artificial intelligence, mental activities recognition, and brain-computer interfaces.