Using Deep Learning to detect Facial Expression from front camera: Towards students' interactions analyze

N. El Bahri[1,2*], *Z. Itahriouan*[2], *S. Brahim Belhaouari*[3] *and A. Abtoy*[1]

[1]TIMS team, Abdelmalek Essadi University, Tetuan, Morocco
[2]SED Laboratory, Private University of Fez, Fez, Morocco
[3]ICT Team, Hamad Bin Khalifa University, Doha, Qatar

Abstract. The recent advancement of Artificial Intelligence (AI) affords ambition to exploit this revolution in multiple fields. Computer-assisted teaching and learning creates a very important area of AI application. Consequently, this last will be able to revolutionize this field. In research conducted by our laboratory, we are interested to explore AI trends to teaching and learning technologies. As part of this, we aim to study learner's behaviors in education and learning environment, thus we aim to analyze the student through the front camera, as a first step we intend to develop a model that classify face's images based on deep learning and Convolutional Neural Networks (CNN) in particular. Model development of images classification can be realized based in several technologies, we have chosen for this study to use IBM solutions, which are provided on the cloud. This paper describes the training experiment and the model development based on two alternatives proposed by IBM where the goal is to generate the most precise model. It presents a comparative study between the two approaches and ends with result discussing and the choice of the accurate solution for deployment in our teaching and learning system.

Keywords. Deep Learning, Artificial Neural Network, Computer Vision, Emotion Detection

1 Introduction

Nowadays the smartest software programs can be based on AI systems because they perform tasks that can be considered intelligent human tasks. Those systems are designed to process: facial recognition, Speech recognition, control systems, Natural language processing, Problem solving Learning, business analytics, Planning and pattern machine [1], [2].

Knowledge engineering is a core part of AI research [3]. this last is based on several AI techniques such as augmented intelligence, deep learning algorithms, approaches of thinking that can help people for [4]: decision making, Fraud Detection, Handling Repetitive Jobs, organizing and managing data in banking and financial institution sectors, identifying the numerous numbers of medical applications and helping the patient to know about the side effects of different medicines and also behaves as personal digital care.

The current progresses well known in AI is mainly a result of advancement in artificial neural networks (ANN) in machine learning. ANN are collections of interconnected nodes that work together to transform input data to output data, its vision is to create artificial intelligence by building machines whose architecture simulates the computations in the human brain [5].

Computer vision is image processing that can help in creating applications that make tasks that resemble to human visual system functions [6] [7], face image processing is an important application field of Computer vision.

IBM Watson is the AI service offer of IBM. It proposes multiple deep learning approaches to process computer vision problems. The goal of this paper is to find experimentally the best approach to deal with facial expression classification.

First, we start by presenting some AI concepts applied to computer vision. We also present deep learning concept and convolutional neural networks. Secondly, we give an overview on the experimental study features and properties. Finally, we present results of experiments to decide which approach offer the accurate model to deploy it in student's emotion detection applications.

2 Computer Vision

Computer vision (CV) is the science responsible for the study and application of methods that enable computers to understand the content of an image This interpretation involves the extraction of certain characteristics which are important for a given aim, a system of visual inspection

[*] Corresponding author: nisserine.elbahri@etu.uae.ac.ma

requires a data entry (image), normally obtained by sensors, cameras or videos, and further processing these data in order to transform them into the desired information [8].

Neural Networks are typically formed by back propagation (BP) and stochastic gradient descent (SGD) to find weights and biases that reduce the loss function in order to map arbitrary inputs to target outputs as closely as possible [9]. The BP algorithm refers only to the gradient calculation method, while the SGD algorithm is used to perform training using this gradient [10].

Deep Convolutional Neural Networks (CNNs) are a specialized kind of artificial neural networks that use convolution in place of general matrix multiplication in at least one of their layers.[10] The CNNs consist of many layers. Such a feature allows them to compactly represent highly nonlinear and varying functions [11]. CNNs involve many connections, and the architecture is typically comprised of different types of layers, including convolution, pooling and fully connected layers, and realize form of regularization [12].

Computer vision (CV) is the science responsible for the study and application of methods that enable computers to understand the content of an image This interpretation involves the extraction of certain characteristics which are important for a given aim, a system of visual inspection requires a data entry (image), normally obtained by sensors, cameras or videos, and further processing these data in order to transform them into the desired information [8].

Neural Networks are typically formed by back propagation (BP) and stochastic gradient descent (SGD) to find weights and biases that reduce the loss function in order to map arbitrary inputs to target outputs as closely as possible [9]. The BP algorithm refers only to the gradient calculation method, while the SGD algorithm is used to perform training using this gradient [10].

Deep Convolutional Neural Networks (CNNs) are a specialized kind of artificial neural networks that use convolution in place of general matrix multiplication in at least one of their layers.[10] The CNNs consist of many layers. Such a feature allows them to compactly represent highly nonlinear and varying functions [11]. CNNs involve many connections, and the architecture is typically comprised of different types of layers, including convolution, pooling and fully connected layers, and realize form of regularization [12].

3 Experiment

In order to build the most accurate model to identify learner's emotions from his face expression, we have realized two experiments of image classification using IBM Watson. The first one uses IBM Watson Visual Recognition « custom model » as solution 1 and the second uses IBM Watson Studio « Watson Machine Learning model » as solution 2. The purpose is to compare their results with each other, to find out which model gives the best accuracy. We mention that both solutions proposed by IBM are executed as a service on the cloud on graphic processors (GPUs), which simplifies the realization of the experiments by reserving the necessary performance as needed.

3.1 Experiment 1: IBM Watson Visual Recognition

As a first step, we have created a custom model using IBM Watson visual recognition to classify face expressions from front camera pictures by classifying the face into an emotional state as clearly shown in figure1. The emotional state indicates sadness or happiness and even anger or frustration and also neutral state.

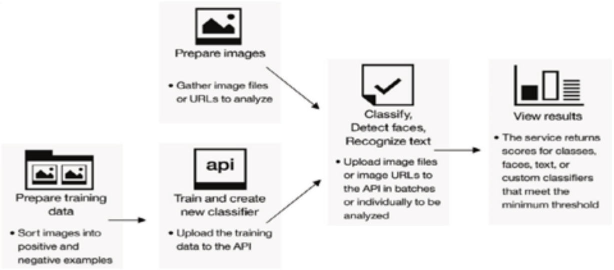
Prepare images
• Gather image files or URLs to analyze
Classify, Detect faces, Recognize text
• Upload image files or image URLs to the API in batches or individually to be analyzed
View results
• The service returns scores for classes, faces, text, or custom classifiers that meet the minimum threshold
Prepare training data
• Sort images into positive and negative examples
api
Train and create new classifier
• Upload the training data to the API

Fig. 1. Overview of the process for using Watson Visual Recognition service[13]

3.1.1. Creating custom model

We have created customized visual classifiers that go beyond the built-in images' classes configured in Watson Studio Visual Recognition tool. Then we added six positive classes (disappointed, sad, satisfied, concentrated, Angry, and Neutral) as shown in the next figure:

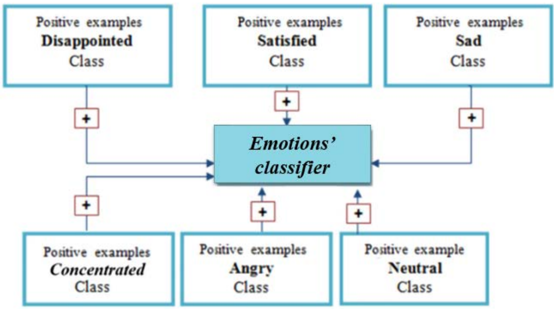
Positive examples Disappointed Class
Positive examples Satisfied Class
Positive examples Sad Class
+
Emotions' classifier
Positive examples Concentrated Class
Positive examples Angry Class
Positive example Neutral Class

Fig. 2. structure of training data for the "Emotions'" custom model

3.1.2. Custom model Training

To train our CNN model, we have used a set of images datasets named "MMA FACIAL EXPRESSION" [14]. The dataset includes thousands of webcam usual facial expressions images, including happiness, sadness, anger, neutrality, disgust, fear and surprise. We divided dataset as follows:

Positive examples collect a set of 500 color images in each class. The dimension of all images is 32x32 where the extension is jpg. All these images have been compressed in a specific archive.

3.1.3. Model test

The test dataset contains images of 32x32 where the extension is jpg. Obviously, images were not used in training process. Each image of the test dataset already has its corresponding class.

3.2 Experiment 2: IBM Watson Studio

We have used the same classes and their images used in the previous model training to train this model, after the first step of data preparation, the second step is the creation of a Modeler flow including all setting of each layer as shown in the figure 3 and Table1. The next step is the creation of a deep learning experiment that aims to train the model by using experiment builder in Watson Studio with the previous data. Finally, the last step will be the model deployment and the accuracy test for some images in json format or using a web service after its configuration.

3.2.1. Deep neural network architecture

The advantage of the second experiment is that we can customize the architecture of the neural network to use for training. As Neural Networks are built from multiple layer and can be designed in different architectures [15], we have used modeler flow tool to design the Convolutional Neural Network to use in our experiment. Figure 3 shows the Neural Network Modeler Architecture:

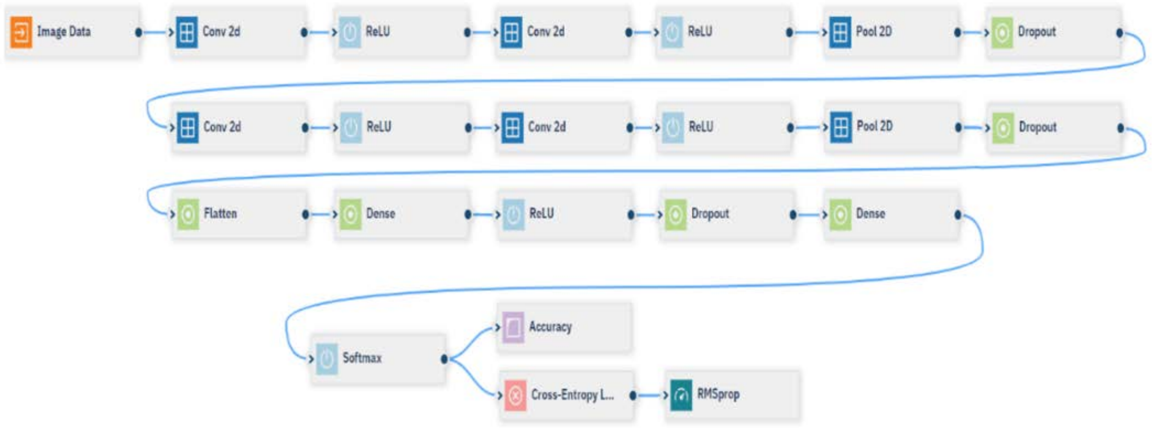

Fig. 3. Nodes that represent the different layers of a Neural Network that are connected to create a flow

The following table shows all components and configuration of the CNN:

Table 1. Modeler flow CNN configuration

Image data		**Conv 2d**		**RelU**	**Pool 2d**		**Dropout**	**Conv 2d**	
Image height	32	***Number of filters***	3	***Negative slope***	***Kernel row***	2	***Probability***	***Number of filters***	64
Image width	32	***Kernel row & col***	3		***Kernel col***	2		***Kernel row & col***	3
Channels	3	***Stride row & col***	1					***Stride row & col***	1
Tensor Dimensionality	Channels last	***Bias***	0		***Stride row***	2		***Bias***	0
Classes	6	***Weight constraint & regularizer***	null	0	***Stride col***	2	0,25	***Weight constraint & regularizer***	null
Data format	Python Pickle								
Epochs	100	***Weight : LR multiplier decay multiplier***	1		***Pooling function***	MAX		***Weight : LR multiplier decay multiplier***	1
Batch Size	32	***Bias : LR multiplier decay multiplier***	1					***Bias : LR multiplier decay multiplier***	1

Pool 2d		**Dense**		**Dropout**	**Dense**		**Accuracy**		**Cross-Entropy Loss**	**RMSprop**	
Kernel row	2	***#nodes***	512	***Probability***	***#nodes***	6	***Phase***	***Top K***	***Loss weight***	***Learning rate***	***Decay***
Kernel col	2	***Initialization***	Glorot_uniform		***Initialization***	Glorot_uniform					
		Bias	0		***Bias***	0					
Stride row	2	***Weight constraint & regularizer***	null		***Weight constraint & regularizer***	null					
Stride col	2	***Weight : LR multiplier decay multiplier***	1	0,5	***Weight : LR multiplier decay multiplier***	1	Train	1	0,3	0,0001	0,000001

3.2.2 Monitoring training progress and results

We can monitor the progress of a training where accuracy is represented in function of iterations (epochs), figure 4 represents accuracy variation during training process:

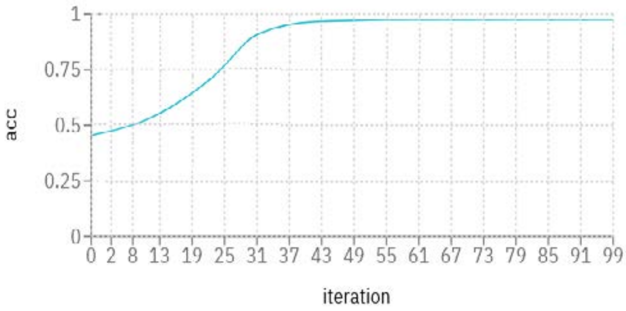

Fig. 4. Learning curve for the accuracy in training process

Precision increases with iterations and it stabilizes near 1 after about 50 epochs. This means that the learning process went well and the predictions made using validation data are excellent.

Another monitoring technic offered by Watson machine learning service is loss monitoring. This tool represents loss rate in function of learning iterations. The following figure shows loss variation during our training process:

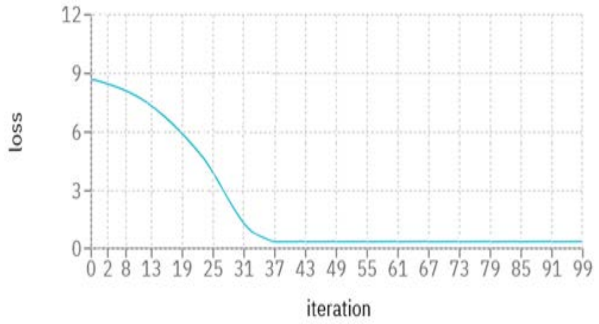

Fig. 5. loss decrease curve in training process

As shown in previous figure, loss decreases with iterations and it stabilizes near 0 after about 40 epochs. This parameter confirm that learning process went well and that our model is ready to make accurate predictions.

3.2.3 Deployed model test:

We have used the same color images with a dimension of 32x32 where the extension is jpg, to test the custom model. To measure efficiency of the model compared to custom model in experiment 1, a specific algorithm has been developed (Details are in the next part).

4 Results and discussion

Both generated models have been implemented as a web service. Comparative tests were based on request sent to the service. The output of prediction tests returns an array of confidence scores for the six classes "Disappointed class / Angry class/ Sad class/Neutral class/Satisfied Class / Concentrated Class". The scores values indicate how well the input matches each class according to generated model. For each test image the class considered as predicted is the class with the highest score with a value near 1 or the class with a higher percentage compared to others. While either the other classes have scores of 0 or a very small number or a low percentage as clearly illustrated in Table 2. The following table show a sample example of 11 test pictures. The table illustrates prediction results made by both models for the same example images:

[*] Corresponding author: nisserine.elbahri@etu.uae.ac.ma

Table 2. The image class accuracy for each model

Images	Watson Machine Learning model						Custom model						image class
	SAD	ANG	CON	NEU	SAT	DIS	SAD	ANG	CON	NEU	SAT	DIS	
image1	0,01	0,02	0,13	0,01	0,82	0,01	0,07	0,03	0,11	0,02	0,72	0,05	**SAT**
image2	0,07	0,03	0,61	0,13	0,09	0,07	0,03	0,05	0,57	0,09	0,20	0,06	**CON**
image3	0,02	0,93	0,01	0,00	0,01	0,03	0,06	0,80	0,02	0,01	0,01	0,10	**ANG**
image4	0,03	0,04	0,06	0,02	0,70	0,15	0,07	0,09	0,13	0,05	0,61	0,07	**SAT**
image5	0,49	0,12	0,03	0,05	0,01	0,30	0,52	0,14	0,03	0,70	0,04	0,20	**SAD**
Image6	0,03	0,00	0,03	0,84	0,01	0,08	0,05	0,02	0,07	0,73	0,10	0,03	**NEU**
image7	0,31	0,16	0,04	0,08	0,01	0,40	0,17	0,28	0,07	0,06	0,02	0,40	**DIS**
image8	0,10	0,76	0,01	0,01	0,03	0,09	0,13	0,71	0,02	0,01	0,03	0,10	**ANG**
image9	0,68	0,10	0,01	0,08	0,01	0,12	0,63	0,14	0,01	0,10	0,01	0,11	**SAD**
image10	0,02	0,03	0,80	0,13	0,01	0,01	0,03	0,05	0,78	0,09	0,05	0,00	**CON**
Image11	0,06	0,07	0,01	0,05	0,00	0,81	0,04	0,11	0,02	0,03	0,02	0,78	**DIS**

By analysing table 2 we can confirm that also for test after deployment, both models give acceptable results.

To decide which approach is better using an experimental evaluation, we have developed an algorithm that systematically makes this decision. Below the evaluation algorithm steps:

For all images, maximum class probability should be that of the intended class:

$$P(C\ max) = P(C\ Intended)$$

If condition is satisfied, we can go to step 2. Otherwise, we must retry learning process again.

- If a. is true, we calculate two measures:

$$Precision\ 1 = \sum_{i=1}^{n} P_i(C_{max}) \big/ n$$

Where:
i: image index
n: number of images
Pi(C): certainty of prediction for Cass C.
Cmax: class with maximum predicted value

Precision 1 can be considered as an average calculated value to make decision between both proposed approaches. We should calculate this value for every one of the proposed experiments and obviously the best approach is that one with maximum precision value.

Precision 1 is practically a good indicator to make decision, but it does not take under consideration values for other classes a part correct class with Maximum value. We assume that a good prediction is that which gives maximum value for correct class and a very low value for other classes. To create a measure that take this under consideration, we have defined another measure (Precision 2) which is illustrated in the following expression:

$$Precision\ 2 = \frac{\left(\sum_{i=1}^{n} \frac{\left(\sum_{j=1}^{m} P_i(C_{max}) - P_i(C_j)\right)}{m-1}\right)}{n} \quad where\ Cj \neq Cmax$$

Where:
i: image index
j: class index
n: number of images
Pi(Cj): certainty of prediction for class Cj.
Cmax: class with maximum predicted value

Precision 2 calculates an average value of difference between correct predicted value and predicted values of other incorrect classes. The purpose is that whenever the predicted value for correct class increases and predicted value for incorrect class decreases, the precision increases consequently. Therefore, to decide which model is better based on Precision 2, we must choose that one with maximum value of this measure.

As every precision parameter measures different metrics, both precisions are important to take under consideration for decision making. Table 3 represents results obtained by calculating Precision 1 and Precision 2.

Table 3. Precision values for different Models.

Watson Machine Learning model Precision1	*Custom model Precision1*	*Watson Machine Learning model Precision2*	*Custom model Precision2*
97%	96,7%	96,3%	95,8%

Table 3 shows that both models give good results in terms of predictions, as precision 1 values are high for both models. Precision 2 confirm that as its value is high for both models. Both measures of Watson machine learning model are better than the custom model measures. This means that this model can make relatively best predictions for our use case. Consequently, model generated in Watson machine learning will be deployed in the solution we are developing in order to detect student face expression from front camera.

5 Conclusion

In this paper, we have presented a brief literature review of computer vision evolution, specifying its algorithms and techniques with a glimpse into deep learning and convolutional neural networks. The first goal of this work was to train the Visual Recognition Service Custom Model and the Neural Network Modeler to classify webcam images based on facial expression. The second one was to compare predictions results accuracy of each model and to decide which model gives the best precision. Therefore, we have developed an algorithm that make decision automatically. According to comparison results, we have found that the difference between the two models results is not very large. We have found also that the difference is minimal in the level of the same image if we compare results image by image. Moreover, we have developed two measures of prediction precision to make the more accurate decision. The model developed using Neural Network Modeler have shown more precision in tests according to both precision values. This allows us to conclude for sure that the model generated using neural network modeler is better. Finally, we have decided to use this last as a model. However, from the front view of the subject we have just pointed to the person's emotional state from the face expression detection and not the personality type or psychological state either. Thus, this last will be the subject of our future research.

References

[1] W. R. Reitman, *Artificial Intelligence Applications for Business: Proceedings of the NYU Symposium, May, 1983*. Intellect Books, 1984.

[2] A. Pannu, « Artificial Intelligence and its Application in Different Areas », vol. 4, nº 10, p. 6, 2015.

[3] Edward A. Feigenbaum, « Knowledge engineering », stanford university stanford, califonia, USA, 1982.

[4] « FSA: Applying AI Techniques to the Familiarization Phase of Financial Decision Making ». https://info.computer.org/csdl/magazine/ex/1987/03/04307089/1e7ugbD1yms (consulté le 19 septembre 2020).

[5] E.-J. Lee, Y.-H. Kim, N. Kim, et D.-W. Kang, « Deep into the Brain: Artificial Intelligence in Stroke Imaging », *J Stroke*, vol. 19, nº 3, p. 277-285, sept. 2017, doi: 10.5853/jos.2017.02054.

[6] R. Szeliski, *Computer Vision*. London: Springer London, 2011. doi: 10.1007/978-1-84882-935-0.

[7] H. Amakdouf, A. Zouhri, M. El Mallahi, A. Tahiri, D. Chenouni, et H. Qjidaa, « Artificial intelligent classification of biomedical color image using quaternion discrete radial Tchebichef moments », *Multimed Tools Appl*, vol. 80, nº 2, p. 3173-3192, janv. 2021, doi: 10.1007/s11042-020-09781-x.

[8] J. F. S. Gomes et F. R. Leta, « Applications of computer vision techniques in the agriculture and food industry: a review », *Eur Food Res Technol*, vol. 235, nº 6, p. 989-1000, déc. 2012, doi: 10.1007/s00217-012-1844-2.

[9] D. Floreano, P. Dürr, et C. Mattiussi, « Neuroevolution: from architectures to learning », *Evol. Intel.*, vol. 1, nº 1, p. 47-62, mars 2008, doi: 10.1007/s12065-007-0002-4.

[10] K. G. Kim, « Book Review: Deep Learning », *Healthc Inform Res*, vol. 22, nº 4, p. 351, 2016, doi: 10.4258/hir.2016.22.4.351.

[11] M. Dalto, J. Matusko, et M. Vasak, « Deep neural networks for ultra-short-term wind forecasting », in *2015 IEEE International Conference on Industrial Technology (ICIT)*, Seville, mars 2015, p. 1657-1663. doi: 10.1109/ICIT.2015.7125335.

[12] A. Ferreira et G. Giraldi, « Convolutional Neural Network approaches to granite tiles classification », *Expert Systems with Applications*, vol. 84, p. 1-11, oct. 2017, doi: 10.1016/j.eswa.2017.04.053.

[13] M. Adan, « Unit 14. Introduction to IBM Watson Visual Recognition », p. 31, 2018.

[14] « MMA FACIAL EXPRESSION ». https://kaggle.com/mahmoudima/mma-facial-expression (consulté le 19 septembre 2020).

[15] N. L. W. Keijsers, « Neural Networks », in *Encyclopedia of Movement Disorders*, K. Kompoliti et L. V. Metman, Éd. Oxford: Academic Press, 2010, p. 257-259. doi: 10.1016/B978-0-12-374105-9.00493-7.

* Corresponding author: nisserine.elbahri@etu.uae.ac.ma