MODELISATION OF SEMANTIC IMAGE RETRIEVAL SYSTEM USING DEEP LEARNING

Ramafiarisona Hajasoa Malalatiana[1] – Randriamitantsoa Paul Auguste[2]

[1]PHD, TASI, ED-STII, Antananarivo, Madagascar
[2]Thesis director TASI, ED-STII, Antananarivo, Madagascar

ABSTRACT

A major challenge in CBIR systems is the semantic gap that exists between the low level visual information captured by imaging devices and high level semantic information perceived by human. The efficacy of such systems is more crucial in terms of feature representations that can characterize the high-level information completely. In this paper, we propose an approach based on convolutional neural networks, a model inspired by biological models that learn from large volumes of data and on computationally intensive architectures.

Keywords: *Machine learning, recognition, neural network, CNN, Image.*

1. INTRODUCTION

1.1 Neural network concept

- Formal neuron

The formal neuron is the main component of an artificial neural network. It is the mathematical modeling of a biological neuron, so a formal neuron receives input variables from other neurons. At each of the inputs is associated a weight ω_{ij} representative of the connection force. Weights can be positive or negative.

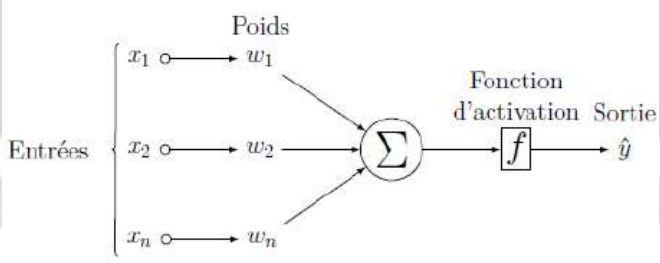

Fig - 1: Formal neuron

The value of the output $\hat{y}$ is given by:

$$\hat{y} = f\left(\sum_{i=1}^{n} x_i \omega_i\right) \quad (1)$$

- Network of artificial neurons

An artificial neural network is an interconnection of several formal neurons. The interconnection between neurons follows a well-defined architecture. This architecture describes the topology of a neural network.
There are two classes of neural network according to their topology:
- forward-propagating networks or feedforward networks where the propagation of information is in a single direction from the input to the output.
- recurrent networks or feedback networks where information spreads in both directions.

- Multilayer networks with forward information propagation

Multilayer networks with forward information propagation or multilayer perceptron are organized in layers. Each layer is composed of neurons and each neuron receives its inputs from the neurons of the upstream layer

and sends its outputs to the neurons of the next layer. The first layer is called the input layer and the last output layer. The layers between these two layers are called hidden layers.

- Back propagation algorithm

Backpropagation is a method that calculates the error gradient for each neuron in a neural network, from the last layer to the first. The purpose of the backpropagation algorithm is to iteratively converge to an optimized configuration of synaptic weights.

- Output layer connection - hidden layer

The updating of the weights between the hidden layer and the output layer is given by the formula (2)

$$\Delta_p \omega_{kj} = \eta \delta_k^p \hat{y}_j^p \quad (2)$$

With:

$$\delta_k^p = \frac{\partial E^p}{\partial v_k^p} = -e_k^p f'(v_k^p) \quad (3)$$

- Input layer connection – hidden layer

The update of the weights between the input layer and the hidden layer is given by (4), we have:

$$\Delta_p \omega_{ji} = \eta \delta_j^p \hat{y}_i^p \quad (4)$$

With:

$$\delta_j^p = -f'(v_j^p) \sum_k \delta_k^p \, \omega_{kj} \quad (5)$$

2. CONVOLUTIONAL NEURON NETWORK (CNN)

Unlike a classical neural network, the CNN layers have neurons arranged in 3 dimensions: width, height and depth.

We use three main types of layers to build CNN architectures: Convolutional Layer, Pooling Layer, and Full-Connected Layer as a fully connected layer. We will stack these layers to form a complete CNN architecture.

2.1 CNN learning

CNN is usually trained in a supervised manner. Unsupervised learning is used only if there is a lack of labeled data. In a multilayer standard neural network, we can defuse the error to a layer j by the following expression:

$$\delta_j^l = \frac{\partial E}{\partial x_j^l} \quad (6)$$

Where:

$$x_j^l = \sum_k \omega_{jk}^l y_k^{l-1} + b_j^l \quad (7)$$

And $y_k^{l-1} = f(x_j^l)$, f is the activation function.In the case of a CNN, we have a convolution operation:

$$x_{i,j}^{l+1} = \omega^{l+1} * f(x_{i,j}^l) + b_{i,j}^{l+1} \quad (8)$$

Which give us:

$$x_{i,j}^{l+1} = \sum_{i'} \sum_{j'} \omega_{a,b}^{l+1} f(x_{i-a,j-b}^l) + b_{i,j}^{l+1} \quad (9)$$

2.2 CNN architecture

Couche	*Entrée*	*F*	*S*	*K*	*P*	*Sortie*
Convolution + ReLU	229 × 229 × 3	7	2	64		112 × 112 × 64
Maxpooling	112 × 112 × 64	3	2			56 × 56 × 64
Convolution + ReLU	56 × 56 × 64	3	1	128	1	56 × 56 × 128
Max pooling	56 × 56 × 64	2	2			28 × 28 × 128
Convolution + ReLU	28 × 28 × 128	3	1	256	1	28 × 28 × 256
Max pooling	28 × 28 × 256	2	2			14 × 14 × 512
Convolution + ReLU	14 × 14 × 512	3	1	512	1	14 × 14 × 512
Max pooling	7 × 7 × 512	2	2			7 × 7 × 512
Convolution + ReLU		3	1	1024		7 × 7 × 1024
Average pooling		7	7			1 × 1 × 1024
Fully connected	1 × 1 × 1024			1000		1 × 1 × 1000
Softmax	1 × 1 × 1000			1000		1 × 1 × 1000

Fig - 2: Proposed architecture

For our paper, we started from the GoogleNet architecture. In terms of computational power, our hardware is quite limited, so we will keep only 10 layers among the 22 layers of the "Inception-v3" architecture, and then organized them to reduce computing time during the learning phase and to obtain a fairly consistent accuracy rate. We therefore see in the table opposite the proposed CNN architecture.

3. IMAGE RETRIEVED BY CNN

3.1 Architecture

A user chooses a request image. The index is calculated via the CNN for the unknown image. The system measures the similarity of the unknown index with the indexes of the database. The system sends the best images in the semantic sense of the similarity measure.

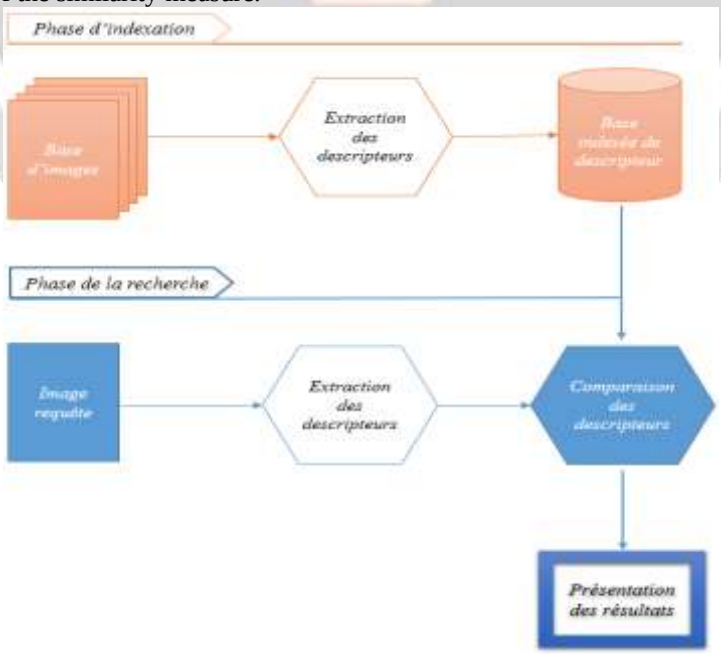

Fig - 3: Architecture of SRIC

3.2 Choosing the image database

The base used for this study is the ImageNet database. ImageNet is a database of more than 15 million high resolution images in 22,000 categories. Of these 22,000 classes, we only retained 24 classes for our tests and a basic set of 1650 images.

Fig - 4: Example of images from the Imagenet database

4. RESULTS

Comparing the responses of a system for a query with the ideal responses allows us to evaluate the following two metrics: precision and recall.
The goal is to find images relevant to a query, and therefore useful for the user. The quality of a system should be measured by comparing the responses of the system with the ideal responses that the user hopes to receive.
In our case, we obtained a rate of 52.38% for recall and 91.66% for accuracy.
For a query image named "jaguar.jpg" for example, we obtain the result represented by the figure 5. From here, we can already observe that the system does not take into account the low level characteristics such as the colour and texture like in global searching with simple signatures of image.

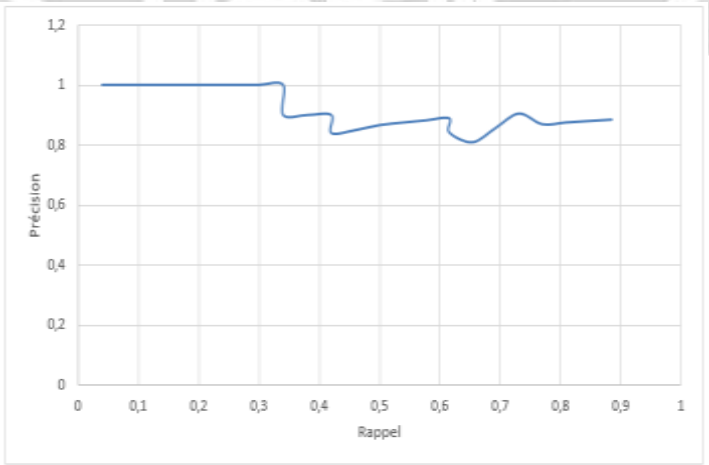

Fig - 5: Recall and Precision curve

For our image search system, the results are sorted in order of decreasing similarity.

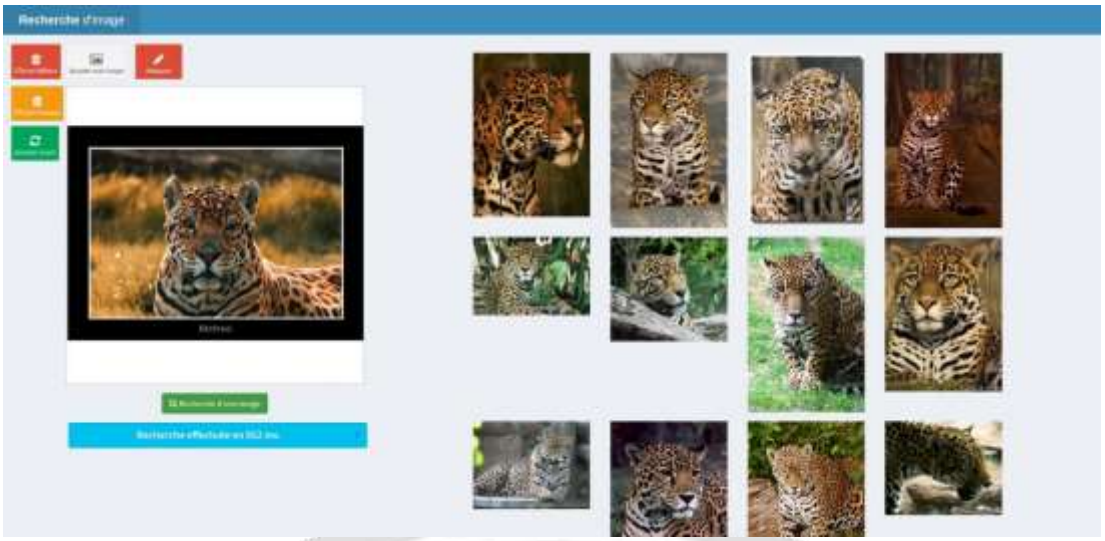

Fig - 6: Resultant images of the "jaguar" search

5. CONCLUSION

Recent advances due to the availability of large data volumes, graphics processors, and advances in optimization have significantly improved the performance of convolutional neural networks and point to many applications in machine vision. In our case, we applied it in a system of image search by the content. To do this, we used a pre-trained CNN to automatically extract the characteristic vectors of images from our database during an offline phase. In a second phase called search line, the user submits an image as a request; the system extracts the descriptors in the same mode as during the first indexing phase. Thus, the descriptors of the request image will be compared to all previously stored descriptors to bring back the most similar images to the request.

6. REFERENCES

[1] B. Krose, P. Smagt, « An introduction to neural network », University of Amsterdam, 1997.

[2] A. Krizhevsky, I. Sutskever, G. E. Hinton, « ImageNet Classification with Deep Convolutional Neural Networks », University of Toronto, 2012.

[3] I. Goodfellow, A. Courville, « Deep Learning », MIT Press book, 2016.

[4] Fei-Fei Li, A. Karpathy, J. Johnson, « CS231n : Convolutional Neural Networks for Visual Recognition », http://cs231n.stanford.edu/, Stanford Vision Lab, Juin 2017.

[5] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, Salakhutdinov, « Dropout : A Simple Way to Prevent Neural Networks from Overtting », Department of Computer Science, University of Toronto, 2014.

[6] D. C. Ciresan, U. Meier, J. Masci, L. M. Gambardella, J. Schmidhuber, « Flexible, High Performance Convolutional Neural Networks for Image Classification », Proceedings of the Twenty-Second international joint conference on Artificial Intelligence, 2013.

[7] T. Kato, « Database architecture for content-based image retrieval », Science and Technology, International Society for Optics and Photonics Hall, 2000.