International Journal of Engineering and Technical Research (IJETR)
ISSN: 2321-0869 (O) 2454-4698 (P) Volume-9, Issue-1, January 2019

Using Convolutional Neural Networks to perform content based image retrieval system

M.H Ramafiarisona, P.A Randriamitantsoa

***Abstract*— Despite being invented close to sixty years ago, artificial neural networks (ANN) remain an area of active research and a powerful tool. Their resurgence in the context of deep learning has led to dramatic improvements in various domains from computer vision. The quantity of available data and the computing power are always increasing, which is desirable to train high capacity models such as Convolutional Neural Networks (CNN). It has been shown that CNN provide a high-level descriptor of the visual content of the image. In this paper, we investigate the use of such descriptors (convolutional neural codes) within the content-based image retrieval (CBIR) application.**

***Index Terms*— Feature extraction, image retrieval, neural network, transfer learning,semantic.**

I. INTRODUCTION

Originally, images were manually annotated with keywords and text-based retrieval systems were utilized. However, due to the rapidly increasing size of image collections, manual annotation became infeasible. Therefore, content-based retrieval systems relying on image content only were developed and are heavily researched within the computer vision community. A separate but related to the image retrieval problem is the problem of image classification. It has been suggested that the features emerging in the upper layers of the CNN learned to classify images can serve as good descriptors for image retrieval. In particular, Krizhevsky et al. have shown some qualitative evidence for that. We measure such performance on Imagenet datasets.

The main problem of implementing and training deep CNN is computational efficiency. Therefore, most implementations use one or more GPUs and took several days. As a result, pre-trained models have become quite popular (that is, the weights of a trained network together with a specification of the network architecture are shared with the community). In the experiments with several standard retrieval benchmarks, we establish that convolutional neural codes perform competitively even when the convolutional neural network has been trained for an unrelated classification task. We also evaluate the improvement in the retrieval performance of convolutional neural codes, when we implement transfer learning technique.

M.H. Ramafiarisona, Telecommunication- Automatic- Signal-Image-Research Laboratory/Doctoral School in Sciences and Technical Engineering and Innovation, University of Antananarivo, Antananarivo Madagascar, +261341654248

P.A. Randriamitantsoa, Telecommunication- Automatic- Signal-Image- Research Laboratory/Doctoral School in Sciences and Technical Engineering and Innovation, University of Antananarivo, Antananarivo Madagascar, +261341034258

II. CONVOLUTIONAL NEURAL NETWORK

A. Neural networks

The prototypical model of neural networks is the L-layer perceptron. Given the output $y^{l-1} \in \mathbb{R}^{m^{(l-1)}}$ of layer $(l-1)$, layer l computes:

$$y_i^{(l)} = f(z^l) = f\left(\sum w_{i,j}^{(l)} y_j^{(l-1)} + w_{0,i}^{(l)}\right),\ 1 \le i \le m^{(l)} \quad (1)$$

Where f is an activation function which is applied component-wise.

B. Layer types

Similar to L-layer perceptrons, convolutional neural networks can be broken down into L layers. Different layer types are used to allow raw images as input, incorporate invariance to noise and distortions and accelerate training.

– Convolutional layer

The convolutional layer is the key-ingredient of a convolutional neural network as it allows handling multichannel images as raw input. If layer l is a convolutional layer, its input is given by $m_1^{(l-1)}$ feature maps $Y_j^{(l-1)}$ from the previous layer, each of size $m_2^{(l-1)} \times m_3^{(l-1)}$. Then, layer l computes $m_1^{(l-1)}$ feature maps as:

$$Y_i^{(l)} = B_i^{(l)} + \sum_{j=1}^{m_1^{(l-1)}} W_{i,j}^{(l)} * Y_j^{(l-1)}, \quad \forall 1 \le i \le m_1^{(l)} \quad (2)$$

Where $B_i^{(l)}$ is a matrix of biases and $W_{i,j}^{(l)}$ is a matrix of eights used as discrete filter. The size $m_2^{(l-1)} \times m_3^{(l-1)}$ of the feature maps $Y_i^{(l)}$ is dependent on the filter size as well as border effects. For $l = 1$, referring to the input image channels as $Y_1^{(0)}, ..., Y_3^{(0)}$, the layer directly operates on the input image.

– Non-linearity and Rectification Layer

A non-linearity layer applies an activation function f component-wise on its input feature maps:

$$Y_i^{(l)} = f(Y_i^{(l-1)}), \quad \forall 1 \le i \le m_1^{(l)} = m_1^{(l-1)} \quad (3)$$

Thus, the output of layer l is given by $m_1^{(l)} = m_1^{(l-1)}$ feature maps of size $m_2^{(l)} \times m_3^{(l)} = m_2^{(l-1)} \times m_3^{(l-1)}$. Common activation functions for convolutional neural networks are the logistic sigmoid, the hyperbolic tangent and the rectified linear unit $f(z) = \max\{0, z\}$, we refer to the layer as rectification layer which can also be interpreted as separate layer.

– Local Contrast Normalization Layer

A local contrast normalization layer aims to create competition among feature maps computed using different filters. Furthermore, contrast normalization layers can also be motivated using results from neuroscience. Krizhevsky et al. use brightness normalization:

$$(Y_i^{(l)})_{r,s} = \frac{((Y_i^{(l-1)})_{r,s})}{(\kappa + \lambda \sum_{j=1}^{m_1^{(l-1)}} (Y_j^{(l-1)})_{r,s}^2)^{\mu}}, \quad \forall 1 \le i \le m_1^{(l)} \qquad (4)$$

– Pooling layer

Average pooling computes the average value within (non-overlapping) windows.

Max pooling computes the maximum value within (non-overlapping) windows.

Pooling has been found to improve convergence and reduce overfitting.

– Fully connected layer

If layer l is a fully connected layer and layer $(l-1)$ one of the above layers, the input feature maps $Y_j^{(l-1)}$ are interpreted as $m_2^{(l-1)}.m_3^{(l-1)}$-dimensional vectors and layer l computes:

$$y_i^{(l)} = f(z_i^{(l)}) = f(\sum_{j=1}^{m_1^{(l-1)}} \sum_{r=1}^{m_2^{(l-1)}} \sum_{s=1}^{m_3^{(l-1)}} w_{i,j,r,s}^{(l)} (Y_j^{(l-1)})_{r,s}), \ \forall 1 \le i \le m_1^{(l)} \qquad (5)$$

III. Content Based Image Retrieval

Content-based image retrieval (CBIR), also known as query by image content (QBIC) is the application of computer vision techniques to image retrieval problem, that is, problem of searching for digital images in large databases [2]. It aims to finding images of interest from a large image database using the visual content of the images. "Content- based" means that the search will analyze the actual contents of the image rather than the metadata such as keywords, tags, and/or descriptions associated with the image. The term 'content' in this context might refer to colors, shapes, textures, or any other information that can be derived from the image itself [3].

In on-line image retrieval, the user can submit a query example to the retrieval system to search for desired images. The system represents this example with a feature vector and the distances (i.e., similarities) between the feature vectors of the query example and those of the image in the feature database are then computed and ranked. Retrieval is done by applying an indexing scheme to provide an efficient way of searching the image database. Finally, the system ranks the search results and then returns the results that are most similar to the query examples [4]. A typical Architecture for CBIR System is illustrated in Figure 1.

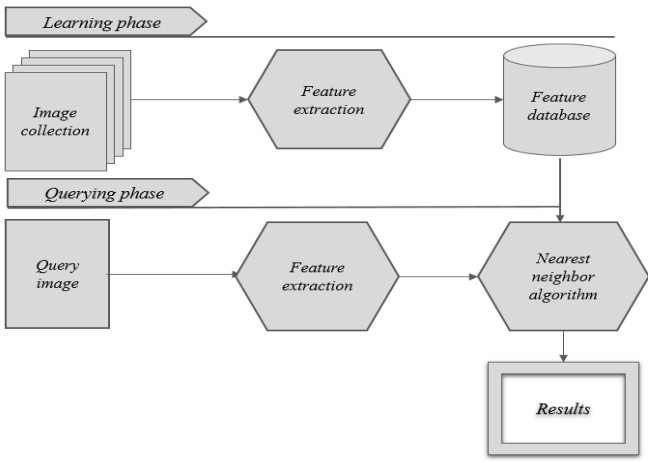

Fig. 1 CBIR system

IV. Approach

A. Using pretrained convolutional neural codes

The model includes five convolutional layers, each including a convolution, a rectified linear (ReLU), and a max pooling transform (layers 1, 2, and 3). At the top of the architecture are three fully connected layers (layer 6, layer 7, layer 8), which take as an input the output of the previous layer, multiply it by a matrix, and, in the case of layers 6, and 7 applies a rectified linear transform. The network is trained so that the layer 8 output corresponds to the one-hot encoding of the class label. The softmax loss is used during training.

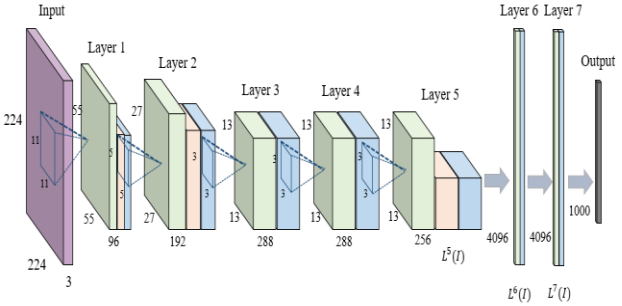

Fig. 2 Architecture used by Krizhevsky and al

A. Transfert learning

A common prescription to a computer vision problem is to first train an image classification model with the ImageNet Challenge data set, and then transfer this model's knowledge to a distinct task. It allows model creation with significantly reduced training data and time by modifying existing rich deep learning models. The concept has a name: Transfer Learning.

The common practice is to truncate the last layer (softmax layer) of the pre-trained network and replace it with our new softmax layer that are relevant to our own problem. Essentially, instead of starting the learning process from a (often randomly initialised) blank sheet, we start from patterns that have been learned to solve a different task

Two common approach may be used: develop model approach and pre-trained model approach. We chose the second approach which consist to:

– Select Source Model. A pre-trained source model is chosen from available models. Many research

institutions release models on large and challenging datasets that may be included in the pool of candidate models from which to choose from. We used Inception-v3 model (Fig. 3).

– Reuse Model. The model pre-trained model can then be used as the starting point for a model on the second task of interest.

– Tune Model. Optionally, the model may need to be adapted or refined on the input-output pair data available for the task of interest.

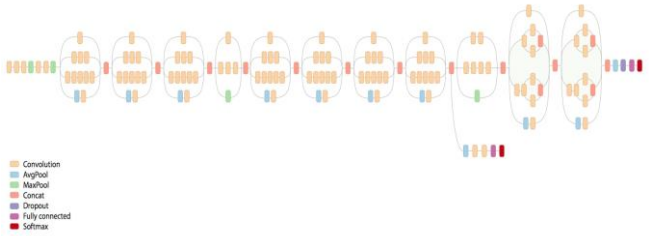

Fig. 3 Schematic diagram of Inception-v3

V. System Evaluation

The system evaluation aims at measuring the performance of the content-based image retrieval system. The testing was performed on a set of 400 images consisting of Chimpanzee, Gorilla, Jaguar, Panthera Tigris, Tiger, Puma, Maki, Indri, Capuchin monkey and Macaque were taken from ImageNet dataset. We used images of animals that look alike on the physical characteristics to highlight the performance of the CNN.

The performance of the system was evaluated by using precision over the first 12 retrieved images and recall. The Precision-Recall curve is a common instrument to visualize and understand the performance of retrieval systems. Furthermore, this curve can be summarized in a single value: Average Precision which can be interpreted as the area under the curve.

$$AP(Z) = \sum_{k=1}^{K} (\mathrm{Re}\, c_k(Z) - \mathrm{Re}\, c_{k-1}(Z) \frac{(\mathrm{Re}\, c_{k-1}(Z) + \mathrm{Pre}_k(Z))}{2}) \qquad (6)$$

The evaluation results show that precision increases when we use transfer learning. The processing time for each retrieval session varied between two seconds. This implies that algorithms under the accuracy search node in our system performs better and find more relevant images. Fig. 3 shows an example of the achieved results when querying the system with a gorilla image.

Classes	Pretrained	Transfer learning
Chimpanzee	0,676	0,830
Gorilla	0,749	0,843
Jaguar	0,690	0,890
Panthera tigris	0,674	0,974
Tiger	0,736	0,916
Puma	0,545	0,916
Maki	0,730	0,833
Indri	0,682	0,882
Capuchin monkey	0,754	0,833
Macaque	0,676	0,750

Table 1 Average precision

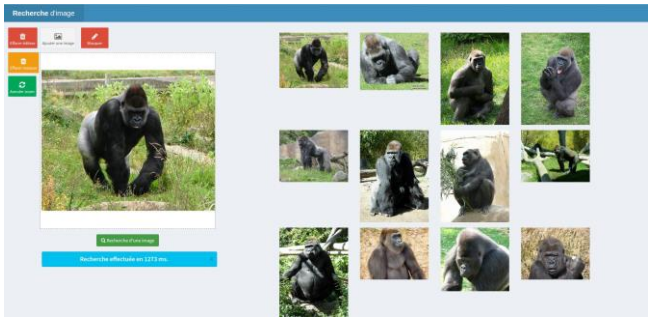

Fig. 3 Example of results

VI. Conclusion and Future Work

In this paper, we have investigated the advance of content based image retrieval using convolutional neural codes. The system has been created by using two approaches: first, using a pre-trained model and second, using transfer learning technique. The test performed on the proposed content based image retrieval system shows the system flexibility and adaptability to the user needs and to the tackled scenario. Future work will regard the extension of this system: first, we plan to enrich our system by using other visual features to extend the collection of usable descriptors. The second improvement is in the research phase where we plan to combine the textual and the proposed approaches to improve the semantic interpretation of the images.

References

[1] Relja Arandjelovic and Andrew Zisserman. « Three things everyone should know to improve object retrieval ». In Computer Vision and Pattern Recognition, Conference on, pages 2911–2918, Providence, Rhode Island, June 2012.

[2] Christopher M. Bishop. « Neural Networks for Pattern Recognition ». Oxford University Press, Inc., New York, New York, 1995.

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. « Imagenet : A largescale hierarchical image database ». In Computer Vision and Pattern Recognition, Conference on, pages 248–255, Miami, Florida, June 2009.

[4] Kevin Jarrett, Koray Kavukcuoglu, Marc'Aurelio Ranzato, and Yann LeCun. « What is the best multi-stage architecture for object recognition ? » In Computer Vision, International Conference on, pages 2146–2153, Kyoto, Japan, September 2009.

[5] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. « ImageNet classification with deep convolutional neural networks ». In Advances in Neural Information Processing Systems, pages 1106–1114, Lake Tahoe, Nevada, December 2012.

[6] James Philbin, Michael Isard, Josef Sivic, and Andrew Zisserman. Descriptor learning for efficient retrieval. In Computer Vision, European Conference on, volume 6313 of Lecture Notes in Computer Science, pages 677–691, Heraklion, Greece, September 2010. Springer.

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

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

[9] 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.

[10] S. Ioffe, C. Szegedy, « Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift», Google Research Lab, 2015.

[11] M. Lew, N. Sebe, C. Djeraba, R. Jain, « Content-based Multimedia Information Retrieval : State of the Art and Challenges », 2006.

[12] S. Thomas, Chang, Shih-Fu, « Image Retrieval : Current Techniques, Promising Directions, and Open Issues », Journal of Visual Communication and Image Representation, 2002.

[13] D. Thomas, K. Daniel, N. Hermann, « Features for Image Retrieval : An Experimental Comparison », Human Language Technology and

Pattern Recognition, Computer Science Department, RWTH Aachen University, Germany, 2007.
[14] D. Kiela, L. Bottou, « Learning Image Embeddings using Convolutional Neural Networks for Improved Multi-Modal Semantics », Microsoft research, 2015.
[15] 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.

M.H. Ramafiarisona, Telecommunication- Automatic- Signal-Image-Research Laboratory/Doctoral School in Sciences and Technical Engineering and Innovation, University of Antananarivo, Antananarivo Madagascar, +261341654248, e-mail: mhramafiarisona@yahoo.fr.
P.A. Randriamitantsoa, Telecommunication- Automatic- Signal-Image- Research Laboratory/Doctoral School in Sciences and Technical Engineering and Innovation, University of Antananarivo, Antananarivo Madagascar, +261341034258