Deep Learning in Low Resolution Image Recognition

Mrs. Jyoti S. Patil
Vishwakarma Institute of Information Technology, Kondhwa Pune.

Abstract – Currently deep learning is the widely adopted machine learning techniques by computer vision and signal processing communities. Deep Learning, in particular, the Convolutional Neural Networks (CNN), are the most impressive classifiers widely used for image recognition in recent years. The key advantages of CNN over the traditional machine learning algorithms are discussed in this paper which includes representation learning at different layers, translation and scale invariance, and parallel implementation with efficient use of GPUs, stochastic gradient descent optimization with a new regularization technique called as ‘dropout’.

INTRODUCTION

The Computer Vision and Signal Processing communities strongly rely on machine learning approach for object recognition. In the initial days of machine learning, the models were trained with a relatively small labelled datasets (thousands of images) to prevent overfitting and then tested on a considerably small test set. However, having realized the importance of ‘BIG DATA’ analysis and the shortcomings of small image datasets [1], we need a machine learning model with large learning capacity. A model that can learn about thousands of objects from millions of images making it useful in different recognition tasks like image recognition, speech recognition, medical diagnosis and numerous other social and business problems. A Machine Learning with big data can be thought of a human learning from the huge collection of life experiences. With a ‘BIG DATA’ of past life experiences, a person can learn to deal with the unfamiliar situations in the present and future.

In traditional Machine Learning Algorithms, hand-crafted features are chosen by programmer to train the model, making the model training for ‘BIG DATA’ a ‘Big Challenge’. Here a large amount of a time, effort and creativity is sacrificed on engineering the good features with a very little focus on execution of the machine learning algorithms. In contrast, in deep learning algorithms the feature engineering is done automatically [2] making it possible to deal with ‘BIG DATA’ in much simpler and faster way. Feature engineering is very difficult, time-consuming task along with the need for programmer to be domain expert. On the other hand, Deep learning algorithms are more accurate machine learning algorithms since they learn the features from data representation itself, and thereby minimizing the need for programmer based feature engineering.
Computer vision algorithms coupled with machine learning have been applied successfully to wide range of recognition tasks in the high-resolution images such as face recognition, signature detection, etc. However, analysing all kinds of images at high resolution for object recognition becomes computationally expensive. For example, in Radio Astronomy, sky is studied in the radio range of electromagnetic spectrum. The different objects in galaxy clusters such as stars, relics, halos, etc.[3], can be viewed using high resolution telescopes giving cosmologists observational inputs for their theories. However, the use of high resolution telescopes in such detection process is very expensive and time consuming and is an unavailable entity. Hence, in such case, low resolution wide survey data (images) is used to find the candidate objects from the galaxy clusters which can be further studied by high resolution telescopes. Similarly, in Biomedical image Processing, the damage detection such as cancer detection is viewed using high-resolution images, but analysing large amount of tissues at high resolution is computationally too expensive and therefore, the potential damage areas are first detected at low resolution and then analysed at high resolution. [4]

II. LOW RESOLUTION IMAGE UNDERSTANDING WITH DEEP CONVOLUTIONAL NEURAL NETWORK

The statistical image features in low resolution images, are less distinct and so applying the same traditional machine learning algorithms for object detection used in high resolution images will be of not much use. For such images, the local features extracted by combination of basic units such as group of pixels will be more meaningful in image recognition tasks. Such feature learning without any human intervention is the key advantage of deep learning algorithms. One such particular type of deep, feedforward networks is Convolutional Neural Networks (ConvNets). [2, 5] The ConvNets are much easier to train and generalize as compared to other neural networks due to full connectivity between their adjacent layers.

Convolutional Neural Networks is the kind of Neural Network architecture that received a remarkable success practically in the period when the neural networks were out of favour [2], due to their ability to learn the useful features automatically [4]. Early 2000s, is considered as the phase when ConvNets were widely been adopted by the Computer Vision and Signal Processing Communities. They have been applied successfully to different image recognition tasks like handwritten digit recognition [9], face detection [10], traffic sign detection [7], and segmentation of biological images [8].etc.

III. KEY FEATURES OF CONVOLUTIONAL NEURAL NETWORKS

The Convolutional Neural Network is a deep-learning architecture with a fully-connected; multilayer stacks each consisting of a set of neurons. During training, each layer transforms its input (starting with the raw input), computes the non-linear input-output mappings to reach the more abstract level for classification of input image pattern. With such multiple non-linear layers, say, a depth of 5 to 20, a system can easily implement the intricate functions of its input[2] making them sensitive to minute image details, more specifically needed in low resolution images. The successful implementation of ConvNets considers following key factors.

A) Automatic Feature Learning
In traditional machine learning algorithms, image features hidden in the matrix of pixels and presented in the form of number arrays does not make any sense. They are limited in their ability to process 2D images in their raw form (pixel values) [2]. Careful feature engineering and a considerable domain expertise is needed to transform the raw image into a suitable internal representation or a feature vector from which a classifier would learn to classify the input patterns.

On the other hand, ConvNets can automatically learn features for the corresponding datasets. At each layer the set of neurons constitute a feature map [4]. At each position, the different neural units in different feature maps will compute the different types of features. In an image, the input comes in the form of an array of pixel values, and the learned features in first layer of representation typically represents presence or absence of elementary visual features such as oriented edges, endpoints, corners, etc[4]. These features are then combined in the second layer to form the particular arrangements for identifying the motifs. The third layer assembles different motifs into the larger combinations that correspond to the parts of familiar objects, and subsequent layers would detect the objects as combination of these parts [14]. For example, in face recognition task using ConvNets, very first layer in the stack is fed with pixel values of different dark and bright pixels in the image (face). It then learns to identify visual features such as edges and other simple shapes in images. These features enables second layer in the net to identify more complex shapes and objects like eyes, ears, nose, etc. On combining such objects and shapes the subsequent layers are in position to define a clear face [11, 12, 14] as shown in Fig. 1.

![Figure 1. Each successive layer in a CNN uses features in the previous layer to learn more complex features.](image)

A full deep ConvNet architecture consists of input layer, output layer and a large number of hidden layers in between them, in contrast to shallow learners with a single hidden layer. The initial layers in hidden layers constitutes of repetitive stacks for Convolutional layers, Sub-Sampling (Pooling) layers (called as feature extractor layers), ReLU layers (rectified
linear unit). It is then followed by one or more fully connected layers (classifier) and softmax probabilities (normalization in range of 0 and 1) layer as the output layer that identifies the class for input images [13] as shown in Fig 2.

The convolutional layers and subsampling layers in ConvNets are inspired by the connectivity patterns between neurons present in animal visual cortex. The set of feature maps in convolutional layers are formed when different local patches in image present in previous layer is convolved with a bunch of kernels (preserving spatial correlation) more specifically called as the filter banks (each of size \(3\times3\) or \(5\times5\)). The result is then sub-sampled (using \(2\times2\) window and max pooling function) followed by ReLU layer having non-linear activation function forming the input for the next layer which can be next convolutional layer or a fully connected layer [6,16].

![Figure 2. Learning hierarchy of visual features in CNN architecture](image)

### B) Fast classification of input data

The traditional machine learning algorithms uses statistical features of the input images (computed by a programmer) to learn the classification process, making it computationally slow and expensive. Instead, the CNN builds the features by applying linear convolutions to images [4]. In a convolutional layer, a single feature map shares the same filter bank (a \(3\times3\) or \(5\times5\) kernel each with similar weight values). However, the different feature maps uses different filter banks [2] enabling parallel and hence faster implementation of entire process[6].
Moreover, the motifs in the image can be ‘stationary’, means if they appear in one part of the image(a local patch) they could also appear anywhere else in the image(another local patch). Hence, sharing same filter weights enables detecting the same patterns of features (such as edges, curves) in different parts of the image[2]. In addition, the ongoing research on fast hardware implementation of CNN architectures with efficient use of GPUs (graphic processing units) [15], enables having deeper nets.

C) Translation, Scale, Rotation and Reflection Invariance

The hand-crafted features in traditional machine learning algorithms give a very poor classification performance when images undergo variations such as illumination, pose, background, occlusions etc. On the other hand, ConvNet architectures are insensitive to irrelevant variations in input images such as variations in position, orientation, illumination of the objects, presence of partial occlusions, etc[2]. Such a structural invariance to translation is possible because in the convolutional layer each feature map shares the same weight vectors, and hence the same output will be produced on convolving the weight vectors with the image patch and the same image patch moved/shifted by few pixels[4]. Moreover, the subsampling layer reduces the dimensions of the representation by computing the maximum of local patches (using 2*2 window) in a feature map thereby making the system scale invariant(probably called as equivariant to scale and translation) [2]. However, CNN is not structurally invariant to rotations and reflections. But they can be generalized to exploit rotation and reflection invariance using generalization techniques namely group equivariant CNN (G-CNN)[17]. Thus, deep ConvNets remain insensitive to large irrelevant variations such as background, pose, lighting and surrounding objects but are highly sensitive to minute image details [2].

D) Classification Error (Loss Function) Minimization

The classification error corresponds to the difference between the actual score of class and the predicted score of the class. In most of the machine learning classifiers a threshold is set for the weighted sum of feature vector components to check for the class of input images. However, in ConvNets a loss function defines the difference between true class and the predicted class for test images. The loss function minimization is possible through different optimization algorithms such a gradient descent algorithm[2]. The knob adjustment for different parameters enables the classification error minimization and hence are called as the ‘HyperParameters’. The key hyperparameters in CNN includes: number of different layers, repetitive number of similar layers, number of neurons in each layer, number of filters, filter size, shared weights, pooling, etc[2]. For each set of hyperparameters the gradient descent algorithm checks the amount by which the error would increase or decrease(forward and backpropagation). A positive gradient vector indicates larger difference between the true class and predicted class for images, while the negative gradient descent corresponds to low difference between the true class and predicted class. A set of hyperparameters tuned to obtain minimum error corresponds to the ‘good set of parameters’ used to model the convolutional architecture that works efficiently for test set of images [4]. However, to address the overfitting problem [21], a new regularization technique called ‘drop-out’ is used [18, 20].
Drop-out drops the random units (from both hidden and visible layers) in the neural network along with their input and output connections temporarily as shown in fig[3]. In dropout, the loss function is minimized stochastically under a noise distribution[2,19] calling the optimization technique as stochastic gradient descent (SGD). The dropout may be turned off during the testing phase.

Considering all these factors, ConvNets can give more faithful way for low resolution image recognition in comparison to other machine learning techniques.

IV. CONCLUSION

The success of Convolutional Neural Networks has brought a revolution in the way Machine learning thinks about different recognition and detection tasks. The successful implementations of ConvNets architectures with 10 to 20 layers, hundreds of millions of weights, and billions of connections between them have made the recognition task for ‘low resolution’ and ‘large scale’ images easier. The advances in hardware, software and parallelization algorithms have reduced the training times for such a large ConvNets to a few hours which otherwise would require few weeks for training.

REFERENCES