PREDICTING THE PIECES OF FURNITURE IN LIVING ROOMS

By Mohammed Ali, Aybuke Yalciner and Hatice Acar

In this paper, we introduce some method of image classification tasks - that is classifying the images - and compare them. We compare the accuracy of the algorithms based on the original images. Our dataset has 10 different classes. We aim to evaluate the method that we choose to get more accuracy.

1 Introduction

Home furniture is an essential part of home decor and furnishings in living rooms of People around the world. People are so keen on decorating their living rooms with pieces of furniture and these types of furniture are similar in most cultures. Having a Smart Home requires recognizing all pieces of furniture in the home to give homeowners more control and some features in their dream homes. So we decided to make the prediction on some 10 main furnitures of any living room such as Sofa, Armchair, Chair, Coffee table, Lamp, Television, Television table, Carpet, Vases and Pictures. To start the project, we firstly collect our dataset and store all photos in separate folders according to their names like in Figure 1.

Figure 1: Sofa

In this study, we propose to learn how we can classify the images and how can we evaluate the accuracy of the classification. We compare some algorithms to decide the algorithm that we will use through the project. The rest of the papers as follows. We briefly review the related studies in Sec. 2. In Sec. 3 we describe the methodologies to be employed and preprocessing. In Sec. 4, we mention about implementations and compare the results. And finally, We describe the future works to evaluate the results in Sec 5.

2 Related Works

Recently there have been several studies done in the image classification based on the convolutional neural network [1,2] and based on a multilayer neural network [3,4]. And also there are some well-known image classification problems that inspire us such as handwritten image classification [5,10,11], iris recognition that uses convolutional neural network [6], Flower Categorization using Deep Convolutional Neural Networks [7, 8], cat or dog classification. [9] This paper is related to convolutional neural network (CNN) and why we choose it. In this project, We use the a very large dataset that is collected by us. This dataset has 10 different classes (Sofa, Armchair, Chair, Coffee table, Lamp, Television, Television table, Carpet, Vases and Pictures.)

3 Employed Methodology

3.1 Preprocessing the dataset

Dataset is collected from various resources and web pages by using some python web scraping libraries such as BeautifulSoup and manually as well. Collected data were not clean, random, with various sizes and different extensions. Preprocessing the Dataset has done by using 2 different libraries to do it OpenCV and PIL. Dataset is standardized to one extension with same pixel sizes for each photo in the dataset.

3.2 Algorithm

There are many machine learning and deep learning algorithms are using to classify the images and make good predictions. Some researches are done to choose the right algorithms to do so. We decided to use Convolutional Neural Network[10,14,16,17] and Multilayer perceptron[15]

3.3 Choosing the right framework

In deep learning, there are some machine learning and deep learning frameworks such as Caffe, Keras, Tensorflow, Torch, and Theano. We were not familiar with any of them. We started reading and researching about choosing one of these frameworks that will be convenient for our dataset.

4 Implementation

We planned to use three algorithms to run and evaluate our project. There are K nearest neighbor algorithm, Convolutional Neural Networks and Multilayer Perceptron. of the We eliminated KNN because it responded too late and now we are working with Convolutional Neural Networks algorithm and Multilayer Perceptron algorithm.

4.1 Convolutional neural networks algorithm

In this study, to choose the correct algorithm for this task, we implement a convolutional neural network algorithm using Keras library. A convolutional neural network (CNN) [10, 14, 16, 17] is a specific type of artificial neural network that uses perceptrons, a machine learning unit algorithm, for supervised learning, to analyze data. CNN's apply to image processing, natural language processing and other kinds of cognitive tasks. A convolutional neural network is also known as a ConvNet. Like other kinds of artificial neural networks, a convolutional neural network has an input layer, an output layer and various hidden layers. Some of these layers are convolutional, using a mathematical model to pass on results to successive layers. This simulates some of the actions in the human visual cortex. CNN's are a fundamental example of deep learning, where a more sophisticated model pushes the evolution of artificial intelligence by offering systems that simulate different types of biological human brain activity. At the end of this task, we come up with accuracy is 80.33 in percentage.

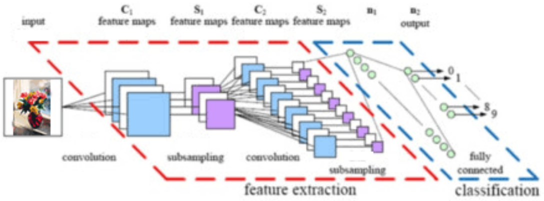

Figure 2: Typical Structure of Convolutional Neural Network

4.2 Multilayer perceptron algorithm

In this study, to choose the correct algorithm for this task, we also implement a multilayer perceptron algorithm using Keras library. Multilayer neural networks solve the classification problem for nonlinear sets by employing hidden layers, whose neurons are not directly connected to the output. The additional hidden layers can be interpreted geometrically as additional hyper-planes, which enhance the separation capacity of the network. The typical structure of multilayer neural network shown in figure 3. The training occurs in a supervised style. The basic idea is to present the images to the network; calculate in the forward direction the output of each layer and the final output of the network. For the output layer the desired values are known and therefore the weights can be adjusted as for a single layer network; in the case

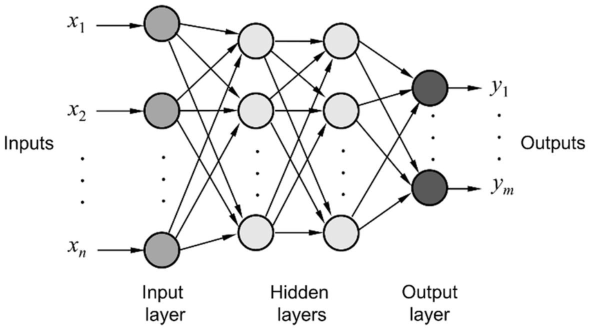

Figure 3: Typical structure of multilayer neural network

of the backpropagation algorithm according to the gradient descent rule. [15, 18]. Our multilayer neural network has 1 input layer (that has 768 nodes), 2 hidden layers (hidden layers have 522 nodes and 351nodes respectively) and the output layer. (that has 10 nodes) [19] At the end of this task, the accuracy is 56.43.

4.3 Selecting the correct algorithm

Fully connected neural networks are good enough classifiers, however they aren't good for feature extraction. Before the emergence of CNN's, the state-of-the-art was to extract explicit features from images and then classify these features. CNNs are trained to identify and extract the best features from the images for the problem at hand. That is their main strength. The latter layers of a CNN are fully connected because of their strength as a classifier. Thats why we choose the convolutional neural network as our method. We see that for equivalent complexity, CNN are better than MLPs at predicting missing details in upsampled images. When we compare CNN and KNN we see that CNN produce faster results.We didnt get the result from KNN anyway because it takes too much time because we have too much data so we eliminate it.

5 Experimental evaluation

Thus far we are in the midst of developing our model for image classification. We will continue with convolutional neural network anymore with 80.33 in percentage with training datasets but there are some overfitting issues when we are testing with new images.
To get more accuracy, we plan to learn more image processing technicals. Maybe we should organize the dataset more.

6 References

[1]https://www.researchgate.net/publication2752-57620-Image-Classification-Using-Convolutional-Neural$_{N}etworks$
[2]https://www.ijser.org/researchpaper/Image-Classification-Using-Convolutional-Neural-Networks.pdf
[3]https://pdfs.semanticscholar.org/b79c/8768-1f6f3b3fd11577-ce2a37991f6dca03d4.pdf
[4]https://www.researchgate.net/publication-224120227-Hand-Gesture-Recognition-using-Neural-Networks
[5]http://yann.lecun.com/exdb/mnist/
[6]https://www.researchgate.net/publication-325014195-Convolutional-Neural-Network-Based-Feature$_{E}xtraction_{f}or_{I}RIS_{R}ecognition$
[7]https://arxiv.org/ftp/arxiv/papers/1708-1708-03763.pdf
[8]https://www.iis.sinica.edu.tw/papers/song-19840-F.pdf
[9]https://sites.ualberta.ca/ bang3/files-DogCat-report.pdf
[10]https://mohitatgithub.github.io/2018-03-28-MNIST-Image-Classification-with-CNN--Keras/
[11]https://mohitatgithub.github.io/2018-03-28-MNIST-Image-Classification-with-CNN--Keras/
[12]https://keras.io/
[13] https://en.wikipedia.org/wiki/Keras
[14]https://en.wikipedia.org/wiki/Convolutional-neural-network
[15]https://en.wikipedia.org/wiki/Multilayer-perceptron
[16]https://ip.cadence.com/uploads/901/cnn$_{w}p - pdf$
[17]https://medium.com/@sidereal/cnns-architectures-lenet-alexnet-vgg-googlenet-resnet-and-more-666091488df5
[18]http://deeplearning.stanford.edu/tutorial-supervised-MultiLayerNeuralNetworks/
[19]https://www.heatonresearch.com/2017/06/01-hidden-layers.html