Monuments Recognition

Yassine Dehbi
yassine.dehbi@etu.enseeiht.fr
yassinedehbi4@gmail.com
Université Internationale de Rabat

2021

Abstract

Due to the vast differences in the construction of different monuments, monument recognition is a difficult challenge in the domain of image classification. Different structure orientations play a significant part in the recognition of monuments. This paper presents a method for categorizing different monuments. For this problem, Deep Convolutional Neural Networks (DCNN) are used. The model is trained using cropped photos of various Moroccan monuments that reflect the country's geographic and cultural diversity. Experiments were conducted on a manually collected dataset that consisted of images of several monuments, each of which had images from various angular perspectives and scales. The trials demonstrate how well the model performs when it is trained on cropped photos of various monuments.
Keywords: Monument recognition, Deep convonutional network, Transfer learning

1. Introduction

Image classification is fundamental yet difficult task in image processing and computer vision. It's been a critical topic in artificial intelligence for decades. Given a test image, the computer should recognize features and predict the corresponding classes or category, i.e digits, cars, animals or an other object that human could recognize. Deep learning solutions have shown their efficiency to solve such complex problems and can even learn to recognize very specific objects in different contexts, i.e low resolution images, scale invariant...
These solutions have been widely used in different domains. In our case, the subject of this paper, which is the recognition of monuments that will be used for an e-tourism application. More precisely, our goal is to replace the panels that carry information about a certain monument, i.e. its name, date of construction, relative history . In this case, the task is a little bit delicate, and requires a deep network to be able to learn the characteristics and features of each monument, and to be able to differentiate them from each other. Fortunately we have a solution called transfer learning. As human beings can learn from what others have learned. Deep neural networks allow the same thing. We take a network already trained to classify objects. And we modify it so that it can learn new objects without learning everything from scratch.

2. Related work

2.1. Dataset

We have at our disposal a dataset of 10 classes, 10 monuments and different images for each one. As shown in Figure 1.

Since we don't have a lot of images per classes, we augment our dataset using Keras image generator by rotating, shifting, resizing and flipping images to generate new ones.

2.2. Choice of Model

Our model is based on the model presented in the paper "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning" [2]. It's a convolutional neural architecture that builds on the Inception family of architectures but incorporates residual connections (replacing the filter concatenation stage of the Inception architecture). The architecture is presented in Figure 2 [1]

Thus, we take the base model pre-trained on ImageNet without including the fully-connected layer at the top of the network and freeze the base model layers. .

Then we add 5 layers:

1. AveragePooling2D
2. Flatten
3. Dense with relu as an activation function
4. Dropout
5. Dense with softmax as an activation function

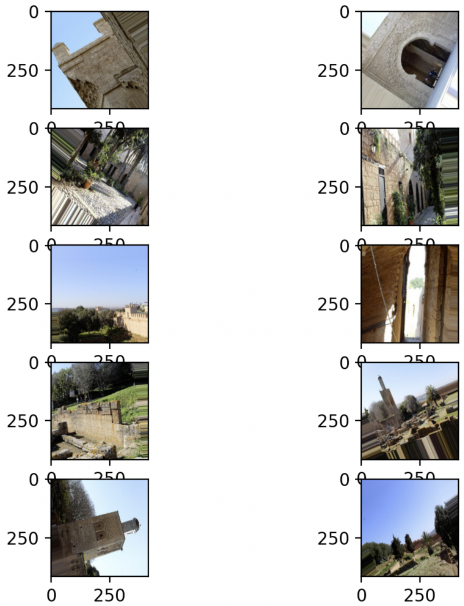

Figure 1: Monuments representation of (10 classes)

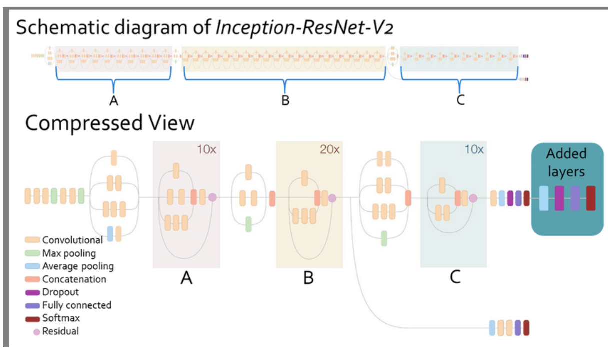

Figure 2: Inception ResNet V2

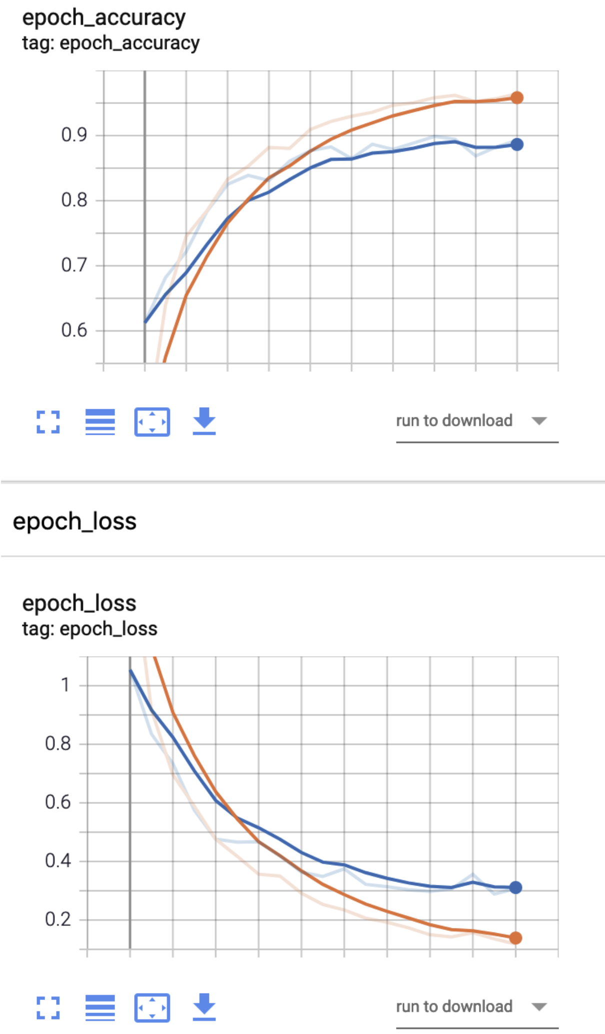

Figure 3: Tensorflow callbacks

2.3. Implementation

We use tensorflow library, we build our model as shown above. Thus for training, we opt for Adam as an optimizer, the categorical cross-entropy for loss function and the accuracy for the metric.(batch size 64, 20 epochs) We obtain results as shown in Figure 4, Table 1 and tensorflow callbacks in Figure

Training acc	Training loss	Test acc	Test loss
0.96	0.11	0.89	0.36

Table 1: Trainig and Test accuracy/loss

```
Epoch 17/20
64/64 [==============================] - 2109s 33s/step - loss: 0.1526 - accuracy: 0.9558 - val_loss: 0.3560 - val_accuracy: 0.8688
Epoch 18/20
64/64 [==============================] - 2069s 32s/step - loss: 0.1491 - accuracy: 0.9500 - val_loss: 0.2890 - val_accuracy: 0.8827
Epoch 19/20
64/64 [==============================] - 2462s 39s/step - loss: 0.1167 - accuracy: 0.9667 - val_loss: 0.3076 - val_accuracy: 0.8926
Epoch 20/20
64/64 [==============================] - 2787s 44s/step - loss: 0.1136 - accuracy: 0.9681 - val_loss: 0.3294 - val_accuracy: 0.8787
##########################################
evaluating model
9/9 [==============================] - 391s 43s/step - loss: 0.2688 - accuracy: 0.8908
[0.2687987685203552, 0.8908382058143616]
```

Figure 4: Training and test accuracy/loss

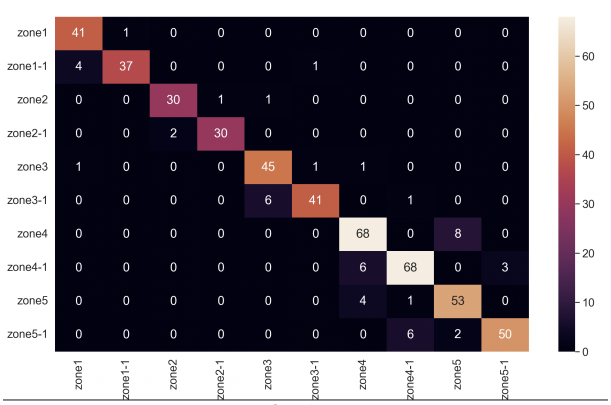

Figure 5: Confusion matrix

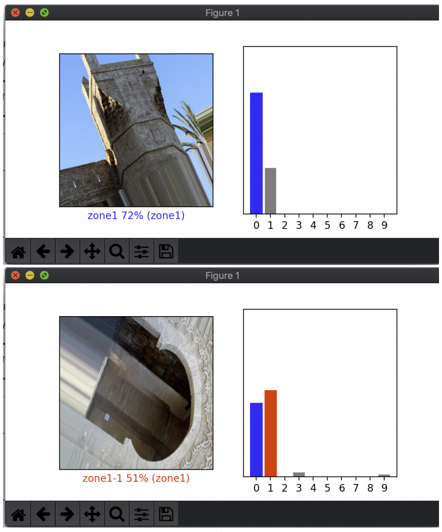

Figure 6: Image and the probability of belonging to each class

2.4. Evaluation

To evaluate our model classification, we use the confusion matrix. We got the results in Figure 5

2.4.1 Notes

- We notice that our model predict the class of our monument.
- Also, zone(i)-1 is a "sub-zone" of zone(i)$(1 \leq i \leq 5)$, i.e this two monuments are in same area and too similar. That's why our model could fail sometimes.

2.5. Application

Finally, given an image of o monuments we predict the probability of belonging to every class. Thus, the class of the monument. Here is some examples in Figure 6

3. Conclusion

In this project and to respond to our task, which is monuments recognition, we implemented a deep CNN model by transfer learning. We have obtained interesting results despite the small dataset and the complexity of the task we have.

References

[1] A shematic diagram of inceptionresnet-v2. https://www.researchgate.net/figure/A-modifiedversionofInceptionResNetV2Szegedy-etal2016wasusedasthe_fig3_327425789.

[2] Rethinking the inception architecture for computer vision. https://arxiv.org/pdf/1602.07261v2.pdf, 2016.