Country Classification Using House Photos

Meltem TOKGOZ
Hacettepe University
21527381
meltemtokgoz@hacettepe.edu.tr

Enes Furkan CIGDEM
Hacettepe University
21526877
enescigdem@hacettepe.edu.tr

Asma AIOUEZ
Hacettepe University
21504074
asma.aiouvez@hacettepe.edu.tr

Abstract

Home designs vary from country to country and when we talk about housing, we should refer to both modern and traditional styles. You can come across a picture of a house taken by someone anywhere in the world and you may wonder where it has been taken from. In this project, we tried to find out which country the photo of a house was taken from. In short, we worked on the problem of classification according to where the photographs were taken.

We used our own World dataset for this project. This dataset contains over 4000 pictures for 15 different countries. In our project, we collected our data from the Flickr [1] , Pinterest [3], and Google Photos [2]. We first tested our data with a single layer neural network and then with convolutional neural networks (CNN). We used ResNet18 and AlexNet models when implementing CNN in our project. In accordance with the results, we applied some methods to increase the accuracy and we got the best accuracy with ResNet18.

1. Introduction

Recognizing home photos and classifying them by country is a quite difficult problem. Because the houses in many countries in the modern world are similar to each other. Beside that, there are some features to distinguish these houses. For example, each country's climate, people's lifestyle and culture are different. This gives us some hints on the architecture of the houses in that country. From this point of view, especially the design of traditionally styled houses begins to change from a country to another. The main problem here is that the houses in the same continent are very similar to each other. For example as shown in *Figure 1*, in the Asian continent, traditionally styled houses of some countries such as South Korea, Japan, Indonesia and Malaysia are very similar. This factor complicates the solution of the problem. In addition, many factors such as the shooting angle, light, shadow and seasonal differences affect the solution of this problem.

Figure 1. Example of similar data

Since this is an image classification problem, there are many algorithms and methods used in its solution. K-nearest neighbors, logistic regression, support vector machine and convolutional neural networks are some of these solutions. Especially in recent years, CNN is a successful algorithm preferred to solving problems in this area.

Figure 2. Example of similar data

In our study, we deal with the problem of classification according to the country where the house pictures were

taken from. For this, we tried to solve the problem by training different convolutional neural network models. We created the world dataset that we use for training and testing with photos we collected from applications such as pinterest flickr and google photos. Detailed information about our image dataset is given in section 3.

We used AlexNet and ResNet18 deep neural network models that were trained on our train data for the image classification problem on the data set we created.After applying the algorithms with different parameters and methods, we compared the results.The hyper-parameters and the different methods we use at this stage are discussed in detail in sections 4 and 5.

2. Related Work

There have been many works and researches on image classification. We have found that many researches have been done on and however, we are especially focused on the studies related to the location estimation with image.We were interested in the ones that used deep learning techniques as that be the method that well be using to train our model.

Deep Learning is the most popular approach in developing artificial intelligence for machines to perceive and understand the world. Image that causes deep learning to become widespread recognition study [10] large dataset ImageNet [7] has received successful results. Instead of handmade attributes that make this structure, which consists of sequential convolution processes more effective than previous studies, it is learned from the raw data itself that the hierarchical attributes suitable for the problem are appropriate.Another important improvement ResNet [9], aimed to learn the residual information between layers in order not to lose the flow of information between sequential convolution layers.

[15] Study of 15 cities identified from turkey and images collected from this cities image classification work done with deep learning.

[11], [14] , [18], [20] used deep learning-based methods to determine where an image was taken.

These [19] [4] [16] [21] [5] [6], [8] , [12] , [13] are some of the other relevant studies we have investigated for our project.

3. Dataset

In our project, we collected house pictures of 15 countries from different continents of the world. Approximately 300 images were collected for each country. And we've obtained about 4000 images in total. The countries which we collect picture data for our project are as follows: Turkey, Russia, Indonesia, England, France, Germany, Australia, Iceland, New Zealand, Italy, USA, Ireland, South Africa, South Korea and Malaysia. *Figure 3* shows some examples of data for the 15 countries in our data.We collected our data from the Flickr [1] , Pinterest [3], and Google Photos [2] in this project.

Figure 3. Dataset

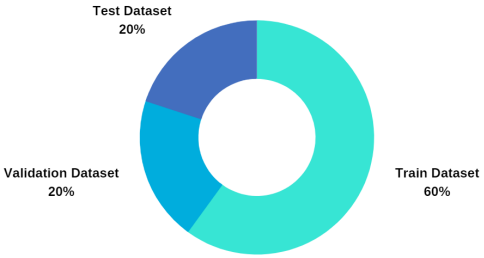

Figure 4. Dataset distribution

While collecting the data we have encountered two problems. First, the number of house image data was quite low for some countries and we had a hard time make it equal for all countries. The second problem was that we needed to make sure that the image data we collected is correct for each country. In order to solve these problems, we gathered our data set, which was created in external images of houses that were taken from different angles and shapes. We have named our image data set World. When applying the single layer neural network algorithm, we observed that the low number of our data affects our success. We applied data augmentation while applying convolutional neural networks which is our main algorithm.

4. Methodology

One way to approach this problem is by using deep learning techniques. In particular, we used Convolution Neural networks. CNNs are one of the widely used supervised learning methods in image recognition applications of machine learning. The network which happens to reduce the number of parameters we need to learn, has sequentially ordered layers, namely: convolution layers, pooling layers and fully connected layers. When we input an image to the network, the computer is rather faced with a matrix of numbers unlike from the human perspective. It first looks for basic characteristics such as boundaries, edges and curvatures. The deeper we move on the network, the more complex and subjective these characteristics get.

4.1. Overview of the model

4.1.1 Convolutional layer

This layer comes always first. It receives the input image and using filters, convolutions are produced. The mentioned operation is analogous to identifying boundaries and colors on images. Next, non-linearity is applied to give it more detection power.

4.1.2 Pooling layer

The purpose of this layer is to reduce the size of the image it receives by getting rid of some non-informative features and also refrain from dealing with features that have been identified in previous convolutional operations.

4.1.3 Fully connected layer

After completion of series of convolutional and pooling layers, it is essential to attach the output information from the convolutional networks to a fully connected layer. This process leads to an N dimensional vector, where N is the number of countries we have on in dataset.

At first, we trained our model from scratch but since our dataset size is not sufficiently large we thought it's a good idea to use some pre-trained models, ones that were trained on very large datasets like ImageNet[17]. We employed two models, ResNet18 which is showed in *Figure 4*. The second pre-trained model is AlexNet[10], which happens to be the first popular CNN model *Figure 5*.

Because CNN works better with large datasets, we tried to improve the model by some transfer learning[9] techniques like and feature extraction.

4.2. Data Augmentation

Before importing the pre-trained methods, we first process the data we have in hands. In order to avoid overftting and increase the variety of data seen during training hence get better generalization results. In addition to the fact that the size of the data we collected is quite small, we attempted to increase it by creating new data based on modification of the existing data. The adjustments made include flipping the images horizontally, cropping them and at the end we normalized them.

4.3. Feature Extraction

For both of the pre-trained models that we have used, we treated the ConvNets as a fixed feature extractor by freezing all the network except the final layer whose parameters will get updated, for it to fit our data classes. It's a good practice not to touch the first hidden layers since the characteristics that get extracted there tend to be universal like edges, shapes and colors.

Layer Name	Output Size	ResNet-18
conv1	$112 \times 112 \times 64$	7×7, 64, stride 2
conv2_x	$56 \times 56 \times 64$	3×3 max pool, stride 2 $\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$
conv3_x	$28 \times 28 \times 128$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$
conv4_x	$14 \times 14 \times 256$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$
conv5_x	$7 \times 7 \times 512$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$
average pool	$1 \times 1 \times 512$	7×7 average pool
fully connected	1000	512×1000 fully connections
softmax	1000	

Figure 5. ResNet18 Architecture

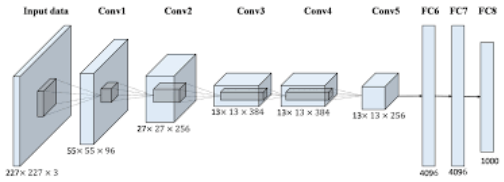

Figure 6. AlexNet Architecture

5. Experimental Results

In this part, we explain our experiments. We discuss and analyze results which we got from experiments as well. Moreover, we used the accuracy metric of classifications and confusion matrix metric for evaluation the results which we we obtained. We tried two method for experiments. First one is learning from scratch, and the other one is feature extraction.

5.1. Learning from Scratch

Firstly, we used Alexnet and Resnet18 models as scratch versions. It means that there is no pre-trained model weights or biases or feature extraction by trained weights where it starts to learn random initialized weights and biases. We did the training process by using maximum 30 epochs number due to time issue, and tried some learning rates 0.1 - 0.001. However, we could not get high accuracies and good confusion matrices. This situation made us look for new ways to improve our results.

The confusion matrix where is showed as *Figure 7* is the best of Alexnet.However some countries are never predicted like France,Iceland.There are many conflictions and mispredictions.
As shown by *Figure 8* , although Resnet18 model gave good predictions comparing to Alexnet on some countries, including Australia, Indonesia and USA, most of the predictions were wrong since there is no time and enough data.

Method	Epoch	Learning Rate	Accuracy
Alexnet	15	0.1	15%
Alexnet	30	0.1	20%
Resnet-18	15	0.01	10%
Resnet-18	30	0.01	32%

Table 1. Scratch learning accuracy scores

In scratch learning, these confusion matrices is obtained.

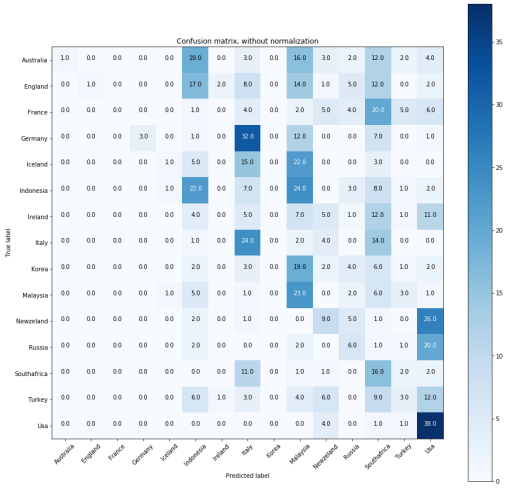

Figure 7. Alexnet - Scratch Learning best confusion matrix

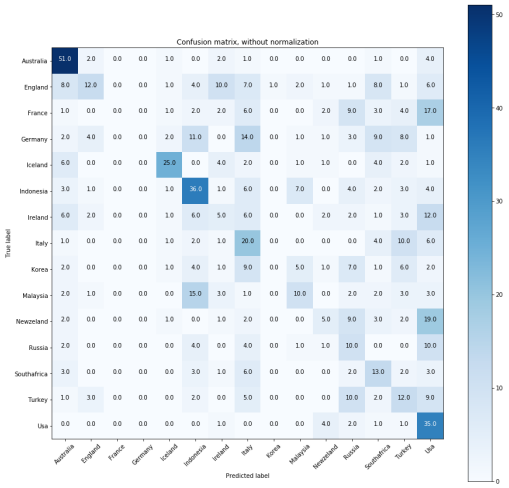

Figure 8. Resnet-18 Scratch Learning best confusion matrix

5.2. Feature Extraction

Scratch learning did not give good results. Thus, we decided to use feature extraction which is a transfer learning method in order to improve our results. We freeze layers of the network for using them as a fixed feature extractor then we re-trained the network to classify our classes of the models. 30 epoch number is used as maximum epoch number and 0.1 - 0.001 learning rates are used as hyper-parameters. Likewise we used in Scratch learning, accuracy scores and confusion matrices is much better comparing to scratch part results though. We used Alexnet and Resnet18 models which are already pretrained on the ImageNet dataset. Highest accuracy scores are obtained by this method and our best accuracy is obtained with Resnet18 model due to fact that Resnet18 has more layers than Alexnet.

Method	Epoch	Learning Rate	Accuracy
Resnet-18	10	0.1	81%
Resnet-18	10	0.01	86%
Resnet-18	20	0.1	85%
Resnet-18	20	0.01	87%
Resnet-18	30	0.01	89%
Resnet-18	30	0.01	87%
Alexnet	10	0.1	76%
Alexnet	10	0.01	75%
Alexnet	20	0.1	77%
Alexnet	20	0.01	75%
Alexnet	30	0.1	75%
Alexnet	30	0.01	81%

Table 2. Feature extraction accuracy scores

In Feature extraction , these confusion matrices is retrieved.

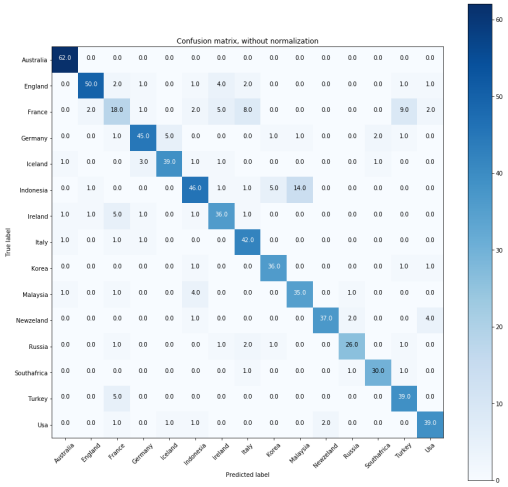

Figure 9. Alexnet confusion matrix

This is the best confusion matrix of Alexnet.There is barely confliction between Indonesia and Malaysia countries.In France class, Turkey and Italy classes are the reasons of confusions.

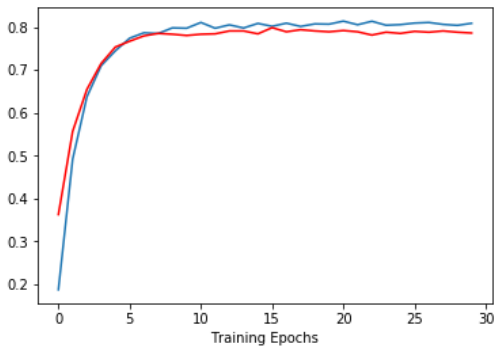

Figure 10. Best accuracy score plotting

Figure 10 shows the accuracy vs epoch graph. The red line is is for the test whereas the blue line is for the train. The accuracy is increasing when the epoch is going up .

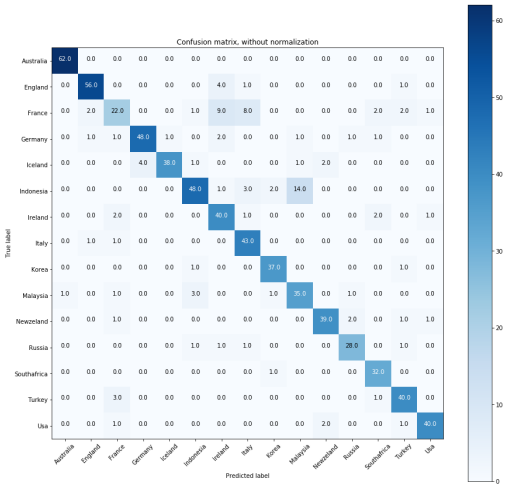

Figure 11. Resnet-18 confusion matrix

Figure 11 is the best confusion matrix for Resnet18 model among all of the matrices. The matrix contains some confusion between Indonesia and Malaysia countries due to fact that the house architectures of these two countries have similarities, but the amount of confusion is decreased. France class have confusions as well. However, there are less mispredictions compared to Alexnet model.

Figure 12 shows the accuracy scores of each class. According to this graph, highest accuracy score we have got is in Australia class with 100% and the lowest accuracy obtained is in France class with 46.80% . France class has probably mislabeled dataset and many similarities with other classes.

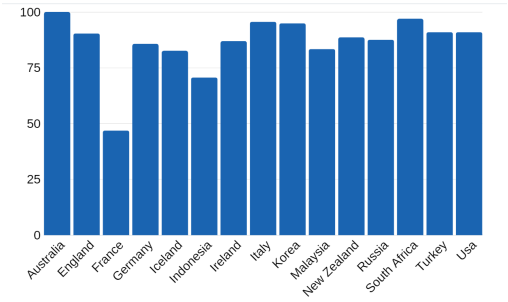

Figure 12.

6. Example Outputs

As a result of our study, some sample results we have obtained for our data are shown in *Figure 13,14,15,16* .

Figure 13. Example Output 1

Predict Class : South Africa

Correct Class : South Africa

Figure 14. Example Output 2

Predict Class : Indonesia

Correct Class : Malaysia

Figure 15. Example Output 3

7. Conclusion

As a wrap up, we initially collected our dataset which consists of images of houses of different architectures for 15 countries from around the world. The dataset was not big enough to obtain good results hence we applied data augmentation on the existing data while training the models. Training is conducted by using scratch and both of the pre-trained models, Alexnet and Resnet-18. We got low accuracy scores from scratch learning so we changed the strategy and used feature extraction which is a transfer learning method and we obtained quite good results.

Another way to improve the model is by applying some more advanced changes on the pre-trained models, like adding/removing layers but up to some extent in order not to face overfitting. Moreover, working on more complex architecture may lead to a better outcome. Finally, varying the dataset by increasing the number of countries and consequently expanding the dataset size.

References

[1] *Flickr*. `https://www.flickr.com/`.

[2] *Google Photos*. `https://www.google.com.tr/imghp?hl=tr`.

[3] *Pinterest*. `https://tr.pinterest.com`.

[4] Where your photo is taken: Geolocation prediction for social images.

[5] Y. Avrithis, G. Tolias, and Y. Kalantidis. Feature map hashing: sub-linear indexing of appearance and global geometry. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 231–240. ACM, 2010.

[6] S. Cao and N. Snavely. Graph-based discriminative learning for location recognition. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 700–707, 2013.

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. Ieee, 2009.

[8] J. Hays and A. A. Efros. Large-scale image geolocalization. In *Multimodal Location Estimation of Videos and Images*, pages 41–62. Springer, 2015.

[9] K. He, X. Zhang, S. Ren, and J. S. r. Deep residual learning for image recognition. *CVPR*, pages 770–778, 2016.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks, 2012. Curran Associates, Inc.

[11] S. Lee, H. Zhang, and D. J. Crandall. Predicting geo-informative attributes in large-scale image collections using convolutional neural networks. In *Applications of Computer Vision (WACV), 2015 IEEE Winter Conference on*, pages 550–557. IEEE, 2015.

[12] Y. Li, N. Snavely, and D. P. Huttenlocher. Location recognition using prioritized feature matching. In *European conference on computer vision*, pages 791–804. Springer, 2010.

[13] T.-Y. Lin, S. Belongie, and J. Hays. Cross-view image geolocalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 891–898, 2013.

[14] T.-Y. Lin, Y. Cui, S. Belongie, and J. Hays. Learning deep representations for ground-to-aerial geolocalization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5007–5015, 2015.

[15] Y. E. Özköse, T. A. Yilikoğlu, L. Karacan, and A. Erdem. Finding location of a photograph with deep learning. In *2018 26th Signal Processing and Communications Applications Conference (SIU)*, pages 1–4. IEEE, 2018.

[16] K. Tang, M. Paluri, L. Fei-Fei, R. Fergus, and L. Bourdev. Improving image classification with location context. In *Proceedings of the IEEE international conference on computer vision*, pages 1008–1016, 2015.

[17] A. Torralba, R.Fergus, and W. T. Freeman. Tiny images, 2007.

[18] N. Vo, N. Jacobs, and J. Hays. Revisiting im2gps in the deep learning era. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 2640–2649. IEEE, 2017.

[19] T. Weyand, I. Kostrikov, and J. Philbin. Planet-photo geolocation with convolutional neural networks. In *European Conference on Computer Vision*, pages 37–55. Springer, 2016.

[20] S. Workman, R. Souvenir, and N. Jacobs. Wide-area image geolocalization with aerial reference imagery. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3961–3969, 2015.

[21] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464, 2018.