Predicting Photography Aesthetics with CNNs

Abhishek Chowdhry, Pratik Chhajer
2015CSB1002@iitrpr.ac.in, 2015CSB1025@iitrpr.ac.in
Indian Institute of Technology Ropar

Abstract

Separating aesthetically pleasing images from collection of large number of images can be a tedious task. Automatic image aesthetic assessment is a challenging artificial intelligence task. In this paper we investigate the aesthetic image classification problem which involves automatic classification of an image into low or high aesthetic quality. Using machine learning techniques, for a given image, we aim to classify whether it is aesthetically pleasing or not on a scale of 1(Very Bad) to 5(Excellent). We propose a novel approach which involves learning ensembles of convolutional neural networks for classification.

Introduction

Automatic classification of images based on aesthetics is a potentially significant research problem to to its application in various fields which require visual experience. They can be used as a ranking criteria in image retrieval systems, in the field of image editing and the ratings can even be used by users to separate their images based on aesthetics. Due to its various applications in multiple fields it is a widely studied problem

The problem however is not very straightforward as image aesthetics may be rated differently by different people. An image which is aesthetically pleasing for one individual may not be aesthetically pleasing for another. Due to these variations it is very difficult to design a highly accurate model for image classification. The aesthetic quality of an image depends on various factors. Some of them include color harmony, good lighting, object emphasis, etc. Some of the recent efforts in image classification are on the basis of a combination of a large number of these basic factors by using more than one convolution neural network to extract various features and then using a combination of their outputs to make the prediction.

To train our model, we used the AADB dataset[4] containing 1000 coloured images each of size 256 X 256. Each image was rated by several users on a scale of 0.0 to 1.0, 0.0 being the lowest score and 1.0 being the highest. The aesthetic value for each image was taken to be the average of all the user ratings which were provided to us in the image name. We further converted this rating to integral value between 1 to 5 by partitioning the continuous line [0,1] into 5 equal segments. The actual sample distribution of the images for different aesthetic values is given below.

Figure 1: An image with low aesthetic quality(left) and an image with high aesthetic quality(right)

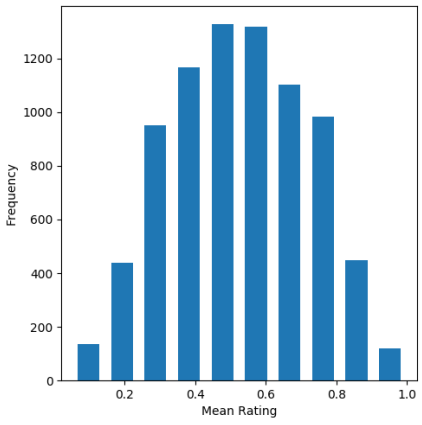

Figure 2: Distribution of images in the AADB dataset

It is clear from the above figure that the distribution of aesthetic scores to images is highly unbalanced. We could not drop any images and were forced to use all the images because the size of the dataset was already too small. We

also could not switch to some other available datasets like the AVA dataset with a large number of images because the size of the images in those datasets were large and we did not have significant computational resources to operate on them. We also could not afford to decrease the resolution of those images as they could have effected the aesthetics. So we used this dataset only and due to the uneven distribution of images in the dataset, the final model that we obtained was biased towards the images with the aesthetic scores in the range of [0.3,0.7] in the actual distribution(in the range of 2-4 in the converted distribution).
We divided the dataset into 2 sets of size 900 and 100 for training and testing purposes respectively. We then trained our CNN model on the training datasets. The exact implementation details are given in the Method section.

Related Work

Several people have worked on this problem in the past decade. Some of the general methods adopted to solve this problem are given below:

CNN for classification: CNN based methods have been used on images to classify them into different aesthetic categories. People have shown that they can achieve high accuracies with fine tuning of different parameters in CNN's like the number of hidden layers, the number of hidden layer units in each layer, the learning rate, etc.[1][7][8] Some people have also use patches of images instead of the entire images and have shown that they can result in good performance[4]. But people have shown examples where taking patches of images instead of the entire image can also result in a miserable performance. In our model we have tried this method of Convolution Neural Networks in order to classify the images into the described 5 aesthetic ranges.We have used an ensemble of CNN's instead of a single CNN in order to achieve good performance.

Models using attributes of images: Recent works have tried to use high level attributes of images/different features in images in order to classified them aesthetically. But there are two problems with usage of these attributes[3][4][6]. First is that these these high level attributes need to be chosen so that they represent all the selected features of the image. Secondly despite of choosing many important attributes needed for classification, we may still miss many features of an image that may be important for classifying that image. Or in other words there may ignore some aspects of an image that are important to the overall aesthetics of an image. So the attributes chosen may work well for one set of images, but may not give very good results when tested with another set of images. We have not used any kind of image attributes in our model.

Models using content of images: Some previous works have also tried to use image content in order to predict the image aesthetics. People have tried tried to predict the photographic object and scene categories in order to predicts the aesthetics of the image[5]. Many works have shown that image content are very useful for image aesthetics. Some works have also assumed the category of the image before predicting the aesthetic quality of that image[4]. But this is not always the case. Predicting the image content is in itself a difficult and error prone task and the category/content of images are often not available. All that is available is raw data. We have not use any kind of image content in order to classify the image in our model.

Method

After we had obtained the training and test data(900 and 100 images respectively), we realized that the data is insufficient for our training purpose. We tried using several parameters for very simple one convolution layer neural networks but the model always overfitted the training data after 5-10 epochs. So we concluded that this can problem can only be avoided by using more training images. Since we didn't had more training data, so we tried data augmentation on the images given to us. For each image given to us in the dataset we generated 3 more images by rotating it clockwise. So we got 4 images for every image in the dataset. After that to further increase the size of the datsaset, we flipped all the 4 images we had obtained to obtain 4 more images. In this way we obtained 8 images for every image in the dataset as shown below:

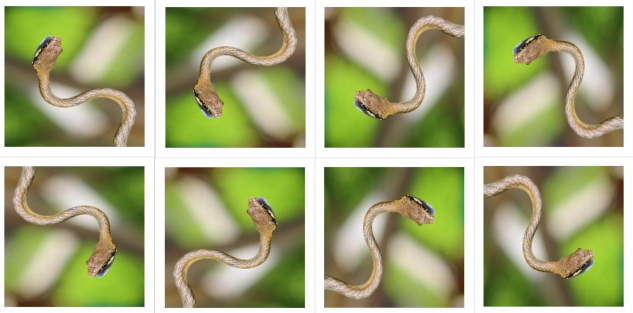

Figure 3: For every image in the dataset we obtained 4 new images by rotating and flipping the original image

The rotating of images worked in our case because the task was not of object detection or matching but was that of aesthetic quality prediction. The aesthetic quality of an image remains same even after the image is rotated or flipped. So this helped us to expand our training and testing data. The training data expanded from 900 images to 7200 images and the testing data expanded from 100 to 800 images.
We further took 10% of the images from training data(720 images) and kept it as validation data. So the training data finally contained 6480 images. We shuffled the data and fed it to our neural network.
After the data was made available to us, we started training a 2 layer convolutional network with 32 and 64 filters of size 3 X 3 and with ReLU activation function and max-pooling after every convolutional layer. In the end we had a fully connected layer of 128 nodes with ReLU activation function and softmax function at the output nodes. We use

the cross entropy error as the error function and performed back-propagation. But our model still overfitted the training data after about 15-20 epochs. So we realized that our model is too complex for such a small amount of training data and decreased the size of or model drastically. When we reached a model with 10 and 15 filters in the hidden/convolutional layers and 64 nodes in the fully connected layer, we realized that the overfitting was still happening but after a comparatively large number of epochs. So we tried to implement Dropout in order to reduce the overfitting in the model. We tried with some initial dropout probabilities and found that dropout probabilities of 0.5 worked well. So we arrived on the model given below:

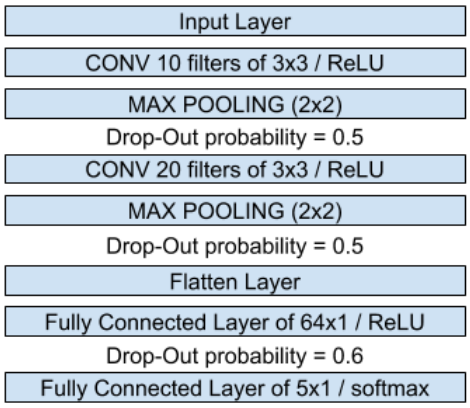

Figure 4: CNN-1

Dropout to a large extent avoided overfitting even after many epochs and helped us obtain a good model. As the dataset we were training on was biased towards the middle aesthetic ratings in terms of number of images, the model that we obtained was also biased. We observed this by watching the predictions of the model as even for the images having aesthetic rating 1 predicted a rating of 5 and vice-versa.

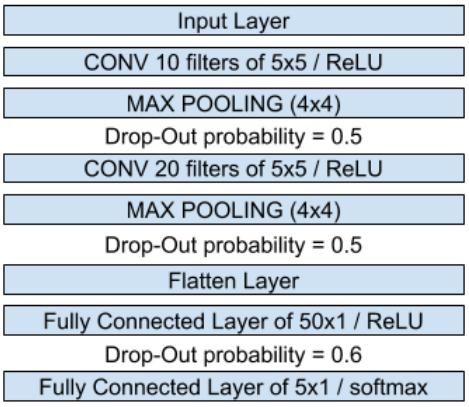

Figure 5: CNN-2

By tweaking with the model a little by increasing the size of the filters to 5 X 5, by maxpooling over a window of size 4 X 4 and by decreasing the size of fully connected layer to 50, we obtained another CNN with similar performance on the dataset. But the errors made by this CNN were different from the first one as shown on the left. We then realized that we can use an ensemble of CNN's to increase our accuracy. We decided to make 3 different CNN's each having its own bias and then combine their outputs to reduce the overall bias in the output. We then started to tweak with the CNN by adding other layers to increase the complexity of the function that it can represent and at the same time reduced the number of nodes in each layer to decrease the complexity of the function it can represent. We did both at the same time in order to prevent overfitting and under fitting. We finally obtained another CNN that gave us similar accuracy compared to the other 2 CNN's. It is shown below:

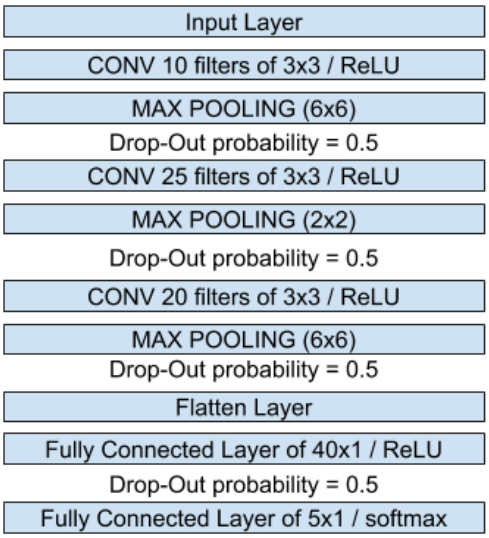

Figure 6: CNN-3

We finally obtained 3 different models by running each CNN for 100 epochs. While running each CNN for 100 epochs, we saved that model which gave maximum accuracy on the validation set. This was done to avoid overfitting. So after we had obtained the final models, we classified the test images using each of the 4 CNN's. The predicted class(1-5) for each test image was then taken to be the majority of classes predicted by all the 3 CNN's. If there was no majority present, then the predicted class was taken to be the class predicted by the the CNN having maximum accuracy individually on the test set. In this way we have implemented an ensemble of 3 CNN's where we take the majority of the three. This helps in eliminating any kind of bias in the final ensemble model. This is because the bias in any CNN is random and hence they cancel out each other when we take the majority. We have presented this final ensemble as our model.

Results and Discussion

The final accuracies of all the three CNN's individually on the testing set are given in the table given below:

Model	Accuracy
CNN 1	0.3523
CNN 2	0.3637
CNN 3	0.4137
Ensemble	0.4225

Accuracies on the test dataset

Also the final accuracy of the ensemble is **42.25%** which is greater than any of the 3 CNN's. This is because the ensemble eliminates the bias in any model towards any class by taking the average of all the models. It does this as it takes the majority of the class labels predicted by any one class. By eliminating any bias it gives the best unbiased results of all the 3 CNN's on any unseen set. That is why it gives the best results on the test set as compared to any one CNN model.

		Predicted				
	Classes	1	2	3	4	5
	1	1	49	14	0	0
	2	3	159	113	13	0
Actual	3	1	62	107	14	0
	4	0	39	69	63	13
	5	0	16	27	29	8

Figure 7: Confusion Matrix for ensemble

If we look at the above confusion matrix, we can see that our model is biased towards the middle aesthetic values. This is due to the presence of more image samples in that range compared to other ranges. As a result the CNN that we train get more training examples having aesthetic values in the range 2-4 and so as a result the final model that we obtain is biased towards these values.

The current **state of the art** accuracy achieved by any model on 5 class classification of the AADB dataset is **67.82%**[4]. The model which achieves this has used many high level image attributes for classification. Those attributes need to be chosen very carefully, so that they reflect all the features of the image. We also need to make sure that that the selected attributes represent all the important features of the image. As a result it takes a lot of time, effort and research to compute those attributes. Due to limited resources and time we could not try extracting those attributes out of the image. As a result our accuracy is less compared to the state of the art accuracy.

Future Work

In future we would like to extend our work to increase the prediction accuracy. There are many possibilities of future work. We can increase the input image size from 256x256 to some larger image so as to capture more information. Here we used 8000 images, which can be increased to some 40,000-50,000 images given proper GPUs are available for this massive computation. We can also try to work on different patches of image and then find prediction on each patch and then find prediction for complete image. There are different ways in which we can choose patches like random picking or center focused, etc.. Here we are aiming to solve problem from pure machine learning approach, but involvement of computer vision can be a great aid in making our models input more robust. We can extract much more information from images like blur, daylight, contrast, brightness etc... Using these to add attributes as input to our model can improve learning of our model.

Acknowledgements

We would like to express my special thanks of gratitude to our professor Dr. Narayanan C Krishnan. We would also like to thank the Teaching Assistants for this course Mr. Sanatan Sukhija and Ms. Akanksha Paul. Without their constant guidance this project would not have been possible. All that we have done is only due to such supervision and assistance and I would not have been bale to complete this project otherwise.

References

[1] Kao, Yueying, Chong Wang, and Kaiqi Huang. "Visual aesthetic quality assessment with a regression model." Image Processing (ICIP), 2015 IEEE International Conference on. IEEE, 2015.

[2] Lu, Xin, et al. "Rapid: Rating pictorial aesthetics using deep learning." Proceedings of the 22nd ACM international conference on Multimedia. ACM, 2014.

[3] Lu, Xin, et al. "Rating image aesthetics using deep learning." IEEE Transactions on Multimedia 17.11 (2015): 2021-2034.

[4] Kong, Shu, et al. "Photo aesthetics ranking network with attributes and content adaptation." European Conference on Computer Vision. Springer International Publishing, 2016.

[5] Wu, Yaowen, Christian Bauckhage, and Christian Thurau. "The good, the bad, and the ugly: Predicting aesthetic image labels." Pattern Recognition (ICPR), 2010 20th International Conference on. IEEE, 2010.

[6] Marchesotti, Luca, et al. "Assessing the aesthetic quality of photographs using generic image descriptors." Computer Vision (ICCV), 2011 IEEE International Conference on. IEEE, 2011.

[7] Jin, Bin, Maria V. Ortiz Segovia, and Sabine Ssstrunk. "Image aesthetic predictors based on weighted CNNs." Image Processing (ICIP), 2016 IEEE International Conference on. Ieee, 2016.

[8] Qian, Chen. "Photo aesthetics evaluation system: an application of CNN and SVM."

Link to source code: **Predicting-Photography-Aesthetics-with-CNNs**