Recognizing Modern Art Movements in Painting with Deep Learning-driven Image Classification

Roberto Daniele Cadili
University of Konstanz
Matriculation no.: 01/906991
roberto-daniele.cadili@uni-konstanz.de

Burak Özturan
University of Konstanz
Matriculation no.: 01/944663
burak.oezturan@uni-konstanz.de

***Abstract*—This paper aims to discuss the construction of a deep learning-driven classifier that is able to accurately recognize modern art movement in paintings. A dataset of 5280 art images for 10 art movements has been fed into a ResNet50 architecture. Accuracy scores of 68% and 64% were obtained on the validation and test set, respectively. These results outperformed the base-line score of about 50% reported in the literature. Notwithstanding satisfactory results, technical limitations and challenges in art image classification contributed to make it extremely hard to obtain high levels of accuracy. Overfitting on training and high loss on validation set represent two of the most relevant concerns. Finally, suggestions for further investigations in the domain of art image classification are proposed.**

Keywords—art movements, classification, deep learning, ResNet 50

I. Introduction

The artistic movement (or style) of a painting is a rich descriptor that captures both visual and historical information about the painting. Correctly identifying the artistic style of a painting is crucial for indexing large artistic databases. Suh a need goes hand in hand with the ever-rising demand of artwork digitalization and content accessibility put forward by art collectors, educators and art enthusiasts. In this paper, we explore the use of deep learning in order to build a style classifier that may help museum curators label art objects automatically, and allow visitors to browse artworks more freely (e.g. by querying for style-related information with just a photo of the artwork).

By constructing a residual neural network with 50 layers, we aim to accurately classify paintings belonging to ten different modern art styles. The paper is composed of four main parts. Firstly, a short review of related works is provided in order to establish the working base-line. Secondly, inherent challenges pertaining art image classification are discussed. Furthermore, the dataset is presented and the necessary preprocessing steps are illustrated. The third part gives a short overview on the implemented ResNet50 architecture and its general mechanics. Finally, the fourth part summarizes the results and performances of ResNet50, and suggests possible ways of improving classification accuracy.

II. Literature Review

Creative artworks have attracted much attention in the AI community, and have generated a flourishing literature that has explored and discussed different approaches to deal with artist or art style recognition problems in visual arts. Details on some pertinent research papers are provided below for illustration purposes. Saleh and Elgammal (2016) used data from Wikiart paintings to classify images by genre, style, and artist using a combination of engineered low-level GIST features and high-level, learned semantic CNN features, and then use a feature vector of high- and low-level features for classification [8]. Florea et al. (2016) assessed the performance of different combinations of image features (e.g. histograms of gradients, spatial envelops, etc.) with different machine learning techniques (e.g. SVM, random forests, etc.). These techniques demonstrated to be unsuited to discriminate between different styles, despite the large size of the dataset and the limited number of labels (12) to predict [3]. Karayev et al (2014) classified artwork from Flickr by style (e.g. Impressionism) and find that a CNN pretrained for ImageNet outperformed models with engineered feature selection. By opting for a pretrained CNN, they achieved a satisfactory accuracy performance, which represents a crucial finding in artwork recognition [5]. Balakrishan et al. discussed the implementation of transfer learning (pretraining on ImageNet) on different fine-tuned architectures (e.g. VGG and ResNet18) in order to classify the most popular artists at the Rijksmuseum of Amsterdam. In their findings, ResNet outperformed VGG, and yielded very high accuracy performances both on train and test set, ranging from 90% to 87%, respectively [1]. Finally, Lecoutre et al. (2017) used a large train set–over 66.000 images–from Wikiart paintings with the goal of classifying images by 24 styles. Those styles mimic very closely some of the classification labels used in this paper. They implemented transfer learning on ImageNet-pretrained deep neural network architectures (e.g. pretrained AlexNet, pretrained ResNet50, retrained ResNet50 and ResNet34) and reached the conclusion that the pretrained ResNet50 model and the retrained one–with a variable number of layers retrained ranging from 1 (the top-layer) layer only to a full retrain–performed better than any other architecture, achieving accuracy of about 50% on the validation set [6] (Fig. 1). Provided that the accuracy on the validation set reached by Lecoutre et al. is taken as base-line for our work, we aim at constructing a model that would outperform this result. Moreover, the findings obtained by Balakrishan et al., and Lecoutre et al. suggest that using a ResNet architecture with transfer learning is likely to yield satisfactory accuracy performances and, thus, we have explored the implementation of such an architecture–pretrained on ImageNet–in the construction of our own network [1][6].

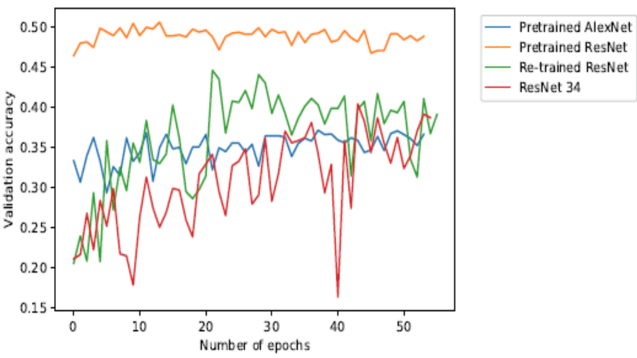

Fig. 1. Base-line comparison (validation set) [6].

III. Challenges in Art Image Classification

Challenges in image classification for paintings can be summarized into three categories [2]. Firstly, while there is abundance of natural images annotated with object, such widespread annotation does not exist for paintings. Secondly, due to lack of annotated art, it is often necessary to learn models from natural images and apply these to paintings. This causes a domain shift problem, given that natural images and paintings can have different low-level statistics. Lastly, paintings can vary considerably in depiction style from photo-realistic representation across artistic movements (e.g. Impressionisms vs. Cubism). Moreover, most modern objects will not be presented in paintings, and some others (e.g. cars, planes) will wave vintage features as opposed to annotated images of modern commercial cars.

Theoretically, a learnt model should ideally adapt well to the new domain, but literature [6] suggests that there is usually a significant drop in performance when a model learnt in one domain is applied to another. Fig. 2 shows an example of different artistic styles, where it is possible to visualize clearly the challenges at hand.

As one can see, there is no easily identifiable visual pattern, which would unequivocally determine to which movement a given painting belong. Color selection is highly variated across styles, and many different colors are used within one single painting. Not even the subjects of paintings can help discriminate the style, for it is possible to observe several common objects (e.g. figure of a man, a woman, a boat) being represented across different movements.

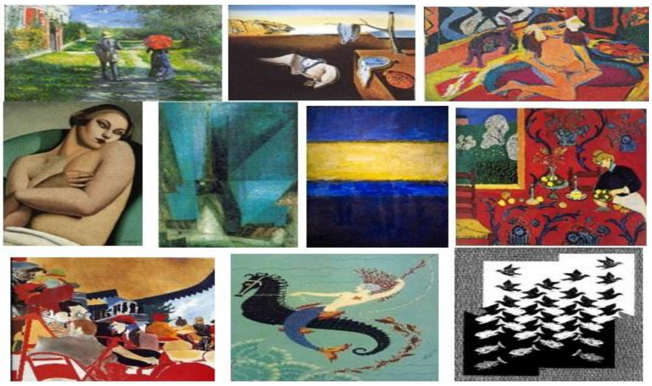

Fig. 2. Paintings for 10 different art styles (from top left to right: Impressionism, Surrealism, Expressionism, Art Deco, Cubism, Abstract Art, Fauvism, Pop Art, Art Nouveau, Optical Art) [9].

IV. Data Description and Preprocessing

The original dataset is taken from “Painter by Numbers” Kaggle Competition [9], where Kaggle collects 103.250 art pieces from Wikipedia and WikiArt.org. After the competition period had ended, Kaggle provided those images with a detailed .csv file which contains information, among others, about the file names of the images, artists–who did the artwork–, title of the painting, style of the painting, genre of the painting, and date–when the painting was created.

The main goal of the original competition was to determine whether two pieces were created by the same artist. For our project, we intend to build an image classifier that is able to recognize modern art movements. There are 136 different art styles in the dataset but we restricted our attention to 10 modern art styles, i.e. 10 classes. Images are not evenly distributed across these 10 classes. Art style names and number of observations are as follows: Impressionism: 10.643, Expressionism: 7013, Surrealism: 4167, Cubism: 1747, Abstract Art: 979, Fauvism: 731, Pop Art: 791, Art Deco: 644, Optical Art: 528, and Art Nouveau: 4.899.

A. Data Preprocessing

As mentioned above, data are not evenly distributed. In order to ensure even distribution of paintings for each style and, therefore, improve accuracy for the classification task, we carried out undersampling to the class with the smallest number of images (Optical Art: 528). Before doing under-sampling, we combined the train and test set provided by Kaggle, so that the total amount of modern art images is augmented to 5280.

Attempts to use all modern art images–32142 in total– were considered, and later abandoned, as doing so did not improve accuracy performances (further details will be discussed in the results section).

The following preprocessing and virtual data augmentation techniques were implemented before training our model:

- *Image resizing:* In the dataset, each image has different shapes, so we resized each image to have the same dimensions of 224 x 224 pixels, which is the standard image size used by the authors of ResNet50.

- *Image cropping:* Images were cropped vertically if the image ratio between the original width and height was smaller than the target image ratio defined in the image resizing.

- *Image normalization:* Images were normalized by dividing all pixel values by the largest pixel value (255).

- *Image flipping:* Images were randomly horizontally flipped to reduce overfitting. Failing to flip may lead the model to find the most obvious features (e. g. facing left or right), and use it as a way of biasedly distinguishing one class from another.

After completing preprocessing, 5280 images (528 for each of the 10 styles) were split into 90% (4276) for train set– 10% (476) of which were used for validation–and 10% (528) for the test set. During the splitting, stratification method for sampling was used to ensure that each art style was fairly represented in the train, validation and test set.

V. Technical Approach

As we already anticipated in the literature review, the technical approach used for our project relies on the construction of an ImageNet-pretrained ResNet50 model, which can be imported from Keras. Details about the implementation and technical aspects of using transfer learning and ResNet50 are provided below.

A. Transfer learning

Transfer learning is a machine learning method where a model developed for a given task is reused as the starting point for another model on a different but strictly related task. It is largely used in deep learning, for pretrained models are often used as the starting point on image recognition tasks given the vast computational and time resources required to develop neural network models. In that sense, transfer learning offers a huge jump in skill that accelerates learning steps [7]. Moreover, Sabatelli et al. (2019) point out that transfer learning is especially convenient when there are a lot of source data–as in the case of ImageNet–, whereas target data are scarce [7]–as in the case of our dataset. Consistent with the literature, in our implementation opting for transfer learning translated into higher accuracy performances compared to other explored experimental implementations (e.g. plain CNN and non-pretrained ResNet50) that did not use transfer learning. However, for our task, it is worth reminding that using ImageNet as a source dataset for training a model before transferring it implicates a domain shift problem. The best-case scenario would entail the possibility of importing a model architecture in Keras that has been pretrained on some sort of "ArtImageNet".

B. *Fine-tuned ResNet 50*

The ResNet architecture was first introduced by He et al. in 2016 [4], as the authors came up with the idea of shortcut connections with the hypothesis that deeper layers should be able to learn something as equal as shallower layers. Moreover, it is assumed that it is easier to optimize the residual mapping than to optimize the original, unreferenced mapping. For that reason, the suggested solution entails copying the activations from shallower layers and setting additional layers to identity mapping. These connections are enabled by shortcut connections, as Fig. 3 shows:

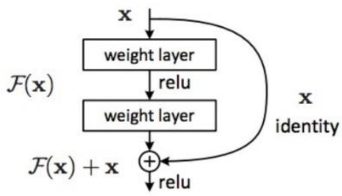

Fig. 3. ResNet shortcut connection [4].

In order to enable these connections (addition operation), one needs to ensure the same dimentions of convolutions through out the network, that's why ResNets have the same 3 x 3 convolutions throughout.

We implemented a ResNet50 architecture in our project, which contains in total 50 convolutional layers, hence its name. ResNet50 starts with a convolutional layer with a filter size of 7 x 7 generating 64 filters, followed by a batch normalization layer, an activation layer and a max-pooling layer. ReLU is used as the activation function for all weight layers, except for the last layer that uses Softmax regression. A fully connected layer ends the network, where the number of neurons corresponds to the number of classes. Our fine-tuned ResNet architecture, did not include the top layers, namely the final pooling and fully connected layer. Instead, a batch normalization, global average pooling and a dropout layer with 0.5 probability were added in order to extract the most important features and reduce overfitting. A dense layer with outputs corresponding to the number of classes and a Softmax regression completed our architectural fine-tuning, and yielded the best accuracy performance on the validation set relative to other experimented architectural layouts.

After exploring many different optimizers and learning rates, the model was compiled using an adaptive Adam optimizer with a learning rate of 0.0001, and was trained on a batch size of 65 units for 50 epochs in order to minimize redundancy in the train set and training time, respectively.

VI. Results

The performance of our model was evaluated employing the accuracy metric. The first finding that stands out is that the accuracy achieved on the test set was 64%, which appears to be a fairly satisfactory score. This is especially meaningful compared to the base-line models and scores obtained on the validation set by Lecoutre et al. (2017), as our implementation outperformed the base-line score by almost 14%. Although it is true that those authors dealt with 24 different art styles, it is worth reiterating that they also had a much larger dataset available.

However, in order to look at the whole picture and conduct a more critical analysis, it is also necessary to investigate the accuracy and loss curves of our model on the train and validation set. Fig. 4 and 5 show train and validation accuracy and loss after 50 epochs, respectively. It is possible to observe how the train curve rapidly and steadily reaches over 90% accuracy after two epochs, and almost 100% accuracy after four. In that sense, the steepness of curve gives away a fundamental problem of overfitting, as our model is able to classify extremely well images on which it has been trained, but shows some considerable weaknesses in its generalization ability. The steady trend of the train accuracy falls slightly only after 11 epochs, and again after 28 epochs at about 95%, but it quickly rebounds after 13 and 30 epochs, converging again towards full accuracy. Less brilliant is, however, the validation accuracy curve, as it behaves quite differently from the train accuracy curve. What stands out at first glance is that the highest achieved level of accuracy is about 68% after 6 epochs, quite far behind the score of 100% obtained for the train set. Moreover, the curve appears to have a quite flat trend–with the exceptions of two distinct falls–, suggesting that each additional epoch marginally changes validation accuracy by a small percentage. Analogously to the trend showed by the train accuracy curve, after 10 and 28 epochs, we observe an abrupt fall in validation accuracy, reaching a value of about 53% at the 12th and 28th epoch. Mimicking the trend of the train accuracy curve, also the validation accuracy curve rapidly rebounds at the 13th and 29th epoch, and it proceeds quite flatly ever after. The same results, interpreted from the perspective of train and validation loss, are showed in Fig 5, where the loss peaks for the validation set are clearly visible.

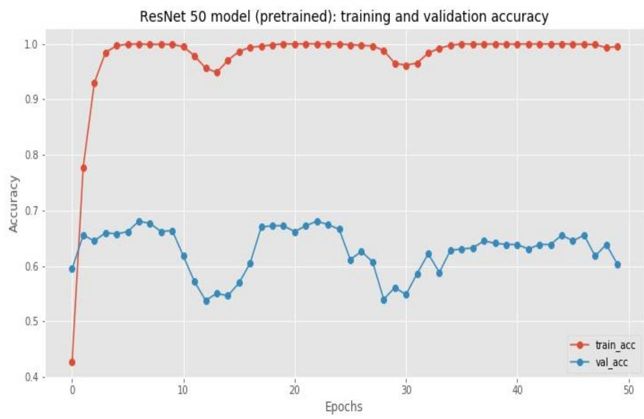

Fig. 4. Training and validation accuracy after 50 epochs.

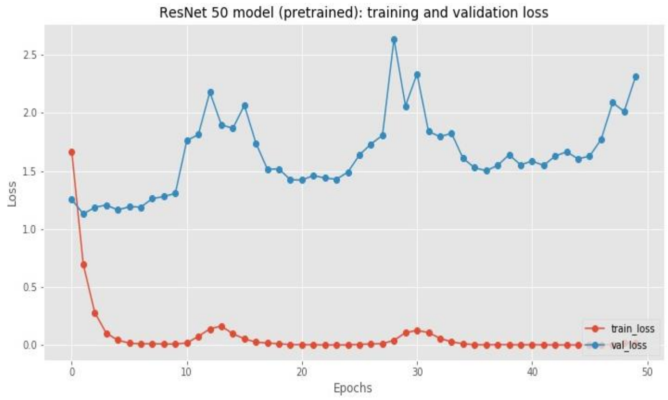

Fig. 5. Training and validation loss after 50 epochs.

Despite substantial actions were taken against overfitting, mainly data preprocessing and augmentation techniques (e.g. cropping, flipping, resizing, normalization), parameter optimization–adaptive and slower learning rate–, and the introduction of architectural changes (e.g. batch normalization and dropout layers), overfitting remains a major concern in our implementation, as it hinders any further improvement on the validation set. Moreover, with the intention of reducing overfitting, it is worth mentioning that we attempted to augment our dataset by including all available images corresponding to our styles. Due to the computational cost of the task, it was necessary to build a data generator that fed the model in small batches. Sadly, however, enlarging the dataset produced very instable results on the validation set, whose accuracy scores ranged from 18% to 60%. In addition to that, data importing times increased considerably, overfitting on the train set was still observed and no significant improvement on test set accuracy could be reported. Because of the above-mentioned reasons, we decided to suspend the use of all available images in favour of class-balanced undersampling.

Finally, the analysis of our results draws attention to the accuracy levels achieved for every style in the test set. The confusion matrix represented in Fig. 6 indicates that Art Deco is the style that was classified with the highest accuracy, reaching a value of 87%. This result is followed by the high accuracy scores achieved by Impressionism and Optical Art with 83% and 85%, respectively. From an art historical perspective, hypotheses about the high accuracy scores obtained for Optical Art and Art Deco might include the preference in both styles for paintings where simple, solid lines, dazzling colours, recognizable objects and well-delimited geometric forms are privileged. These features might have helped the model capitalize on the skills learned during the pretraining, as subjects in these painting styles might approximate more closely objects represented in the ImageNet. Less intuitable seem to be the reasons why Impressionist artworks are being classified with such a high accuracy. A possible hypothesis concerns the preference of Impressionist artists to paint *en plen air* (outdoors) and, hence, reproduce landscapes and elements of nature (e.g. animals, flowers, etc.). Such characteristics might have played a key role in helping the model take advantage of lesson learnt on the vast collection of images representing animals and floral elements contained in ImageNet. Sensibly lower accuracy scores were achieved for paintings belonging to Pop Art, Expressionism, Cubism, Art Nouveau and Abstract Art, ranging between 68% and 51%. It appears unsurprisingly that Cubism and Abstract Art paintings were classified less accurately, as these artistic movements favoured the transfiguration of everyday objects, people or nature into hardly-identifiable, even for human eyes, objects. Hence, the model had very little chance to capitalize on its transferred learning skills, which might explain such low performances. It is, however, very surprising that Art Nouveau paintings were classified with low levels of accuracy, for such a movement is notorious for its elegant lines, light colours and abundance of floral elements, which are widely represented in ImageNet. Furthermore, it appears that the model often predicts wrongly Art Nouveau as Impressionism, for it is very likely being misled by the abundance of floral elements in both styles. Finally, Surrealism and Fauvism were recognised with less than 50% accuracy, which constitutes a quite unsatisfactory result. It is hard to determine the reasons why such styles were classified so poorly, but perhaps the deformed representations of common objects, the unrealistic landscapes, and the surreal collages of animals whose body parts are often the synthesis of different animals–hence the name Surrealism–might have played a crucial role in misleadingly influencing the classifier. Because of such stylistic peculiarities, our classifier might have been unable to correctly recognize objects and style patterns.

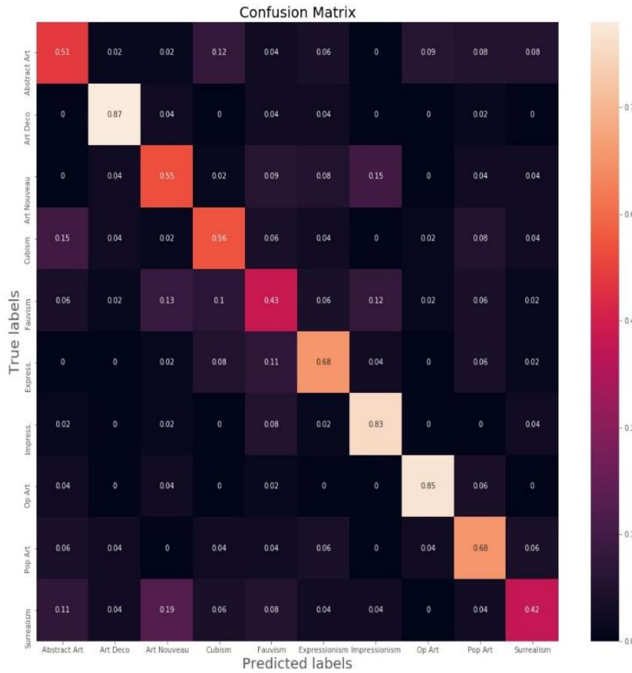

Fig. 6. Confusion matrix for each style.

VII. Conclusion

Building a deep learning-driven classifier that is able to correctly identify different styles in paintings has turned out to be a challenging task, despite the fact that the literature has proven the superiority of ResNet50 architecture in completing such task successfully. Notwithstanding the challenges connected with art image classification, the achieved general level of accuracy on the test set was satisfactory, especially compared to our base-line model. However, there are still unsolved accuracy concerns caused by overfitting on the train set, which hinder any improvement on the validation set, and determine the upward trend of the validation loss curve.

Further investigation may attempt to reduce overfitting by carrying out additional image preprocessing (e.g. blurring using Gaussian noise, segmentation and morphology), exploring advanced augmentation techniques (e.g. conditional GANs– transforming an image from one domain to an image of another domain, or the less computationally intensive neural style transfer), or further fine-tuning and optimizing the model architecture and parameters.

References

[1] T. Balakrishan, S. Rosston, E. Tang, "Using CNN to classify and understand artists from the Rijksmuseum", Univeristy of Stanford, unpublished.

[2] E. J. Crowley, "Visual recognition in art using machine learning", PhD thesis, University of Oxford, 2016, unpublished.

[3] C. Florea, R. Condorovici, C. Vertan, R. Butnaru, L. Florea, and Ruxandra Vranceanu, "Pandora: Description of a painting database for art movement recognition with baselines and perspectives". Proceedings of the European Signal Processing Conference (EUSIPCO), 2016.

[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition". Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770-778.

[5] S. Karayev, M. Trentacoste, H. Han, A. Agarwala, T. Darrell, A. Hertzmann, and H. Winnemoeller, "Recognizing image style". Proceedings of the British Machine Vision Conference (BMVA). BMVA Press, 2014.

[6] A. Lecoutre, B. Negrevergne, F. Yger, "Recognizing art style automatically in painting with deep learning", Journal of Machine Learning Research: Workshop and Conference Proceedings, vol. 80, pp. 1-17, 2017.

[7] M. Sabatelli, M. Kestemont, W. Daelemans, P. Geurts, "Deep Transfer Learning for Art Classification Problems", in Computer Vision – ECCV 2018 Workshops. ECCV 2018. Lecture Notes in Computer Science, vol 11130, L. Leal-Taixé, S. Roth (eds). Springer, Cham, 2019.

[8] B. Saleh and A. Elgammal, "Large-scale classification of fine art paintings: Learning the right metric on the right feature", in International Journal for Digital Art History: Visualizing Big Image Data, vol. II, H. Klinke, L. Surkemper, J. Underhill (eds). Graphentis Verlag e.K., 2016, pp. 71-93.

[9] Dataset: Kaggle, "Painters by numbers", 2016. Available at: https://www.kaggle.com/c/painter-by-numbers/overview [Last accessed on 29.08.2019].

Declaration of Autorship

We hereby declare that the paper submitted is our own unaided work. All direct or indirect sources used are acknowledged as references. This paper was not previously presented to another examination board and has not been published. We both spend comparable time and e ort to this project concerning research, coding, creating the report and the presentation.

Konstanz, 30.08.2019

Place and Date

Roberto Daniele Cadili

Burak Özturan