Multi-label Classification of Artworks into their Represented Icon(s)

Ludovica Schaerf
11925329

Liberal Arts and Sciences

Bachelor of Science

Amsterdam University College

Submitted on June, 3^{rd} 2020,
Word Count: 8700 words

Student Email	ludovica.schaerf@gmail.com
Supervisor	Dr. Giovanni Colavizza
Supervisor Affiliation	University of Amsterdam
Supervisor Email	g.colavizza@uva.nl
Reader & Tutor	Prof. Breanndán Ó Nualláin
Reader Affiliation	Amsterdam University College
Reader & Tutor Email	o@uva.nl

The link to the repository containing the code written for this Capstone is this. The information to navigate around the repository is in the README.

Acknowledgements

First and foremost, I would like to thank my Supervisor, Giovanni Colavizza, for his enthusiastic, continuous and attentive guidance, for pushing me to always achieve more and for acknowledging my progress. Secondly, I would like to thank my Reader, Breanndán Ó Nualláin, as he taught me to program and to enjoy the field, and, especially, as he offered me the incredible learning opportunity of interning at his Start-Up. My grateful thanks are also extended to my dad, Marco Schaerf, and his computer, that, unlike my laptop, ran all of my models without complaining.

I would also like to extend my thanks to everyone at ALCOR lab, and especially Edoardo Alati and Fiora Pirri, who taught me about Computer Vision with incredible affection and availability.

Finally, I wish to thank my family and friends for their support and their proof-reading throughout my study.

Abstract

As the field of Computer Vision develops, new applications of its techniques need to be investigated. One of these is the automatic recognition of icons represented in artworks. To the best of our knowledge, this task has not yet been comprehensively tackled using Deep Learning classification methods on a dataset of classic, modern, and contemporary artworks. To fill this gap, this paper uses different multi-label and transfer learning techniques to classify the artworks in the Tate Collection Dataset ([25]) into their content. In this paper, artworks will be classified into 15 general icons based not only on the recognition of physical elements in an artwork but abstract concepts such as Religion & Belief as well. In Appendix 1, we show some preliminary results on 141, more specific, classes using hierarchical methods. The best model, CNN-RNN, achieves a mean per-class precision of 0.57 and a per-class accuracy of 0.83.

The automatic generation of iconographical metadata will contribute to enrich the indexation of artworks, allowing to search paintings by content rather than only by artist, style and period, as most websites offer now. On a broader level, these experiments will allow us to trace some conclusions on the ability of current machine learning techniques to recognise the content of artistic depictions: Are the models able to extrapolate the manner in which objects are depicted in art? Can they understand how specific abstract themes and concepts are represented?

Keywords— Deep Learning, Iconography, Paintings Classification, Digital Humanities, Machine Intelligence

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

In 2015, Deep Residual Networks outperformed human annotators ability at recognising the content of images, ending up first at the ImageNet Large Scale Visual Recognition Challenges (ILSVRC, [1, 7]). The achievement constituted a leap forward in computer vision, allowing the machine learning community to tackle increasingly complex tasks. This breakthrough raised various followup questions, including whether machines can recognise the content of artworks, and, in particular, paintings. These are more challenging compared to pictures. Firstly, recognition of artistic iconography requires human-level synthetic ability and knowledge of cultural-sociological context and aesthetic representational tools ([6]). Further, representation in paintings involves a greater level of abstraction and variation. ([32]).

Figure 1.1: An example of an object represented at different levels of abstraction. *From left to right*: A picture of a guitar, Caravaggio's Suonatore di liuto (1595), and Picasso's Guitar (1920)

Take as an example Figure 1.1. The representation of the object in the three images varies incredibly. The picture on the left has few elements of variance, pertaining to the natural variation in the object types, its exposure, pose and inclination. Already in Caravaggio's depiction, the variance increases: some parts are voluntarily

shaded to create a theatrical effect, chiaroscuro on the instrument is sketched but not completed. In Picasso, the depiction is much more abstract: it is fragmented and bi-dimensional, hardly recognisable to the rookie eye.

In order to tackle the problem introduced above, we propose a comparative study on the applicability of deep learning techniques, including the aforementioned Deep Residual Networks, to cross-domain paintings classification. Furthermore, we assess the performance of a number of pre-trained models to generate features for the classification task. The feature-extraction models have been chosen among the top competitors at the ILSVRC, while the classifier architectures have been selected due to their state-of-the-art performance or their simplicity.

We classify paintings into the icon(s)[1] they depict. The Subject index ([23]), that this classification experiment uses as target, was introduced by the Tate to allow searching by content in their digitised archive ([24]). The index is an adaptation of the already existing classification system of Iconclass, a hierarchical thesaurus that organises artworks into 'what they are about' ([14, 29]). A classification using the Subject index is complex, as the thesaurus is inherently multi-label and hierarchical. Furthermore, some of the classes of the Subject index, together with containing few images, represent rather abstract concepts, such as Emotions, Concepts and Ideas, and are, thus, presumably harder to classify.

In this paper, we aim to evaluate the performance of the best model in terms of its practical utility at automatically producing metadata and to analyse its results on the basis of the following topics: The difficulty level of the different iconographical classes; The difficulty level of different styles and artistic periods; And, more specifically, the difficulty level of classifying 20^{th} and 21^{st} century abstract art[2]. The paper will begin by providing an overview of the existing literature. It will then proceed to elucidate the methods used for classification and present their results. This will be done by comparing the performance of the different models, analysing

[1]The term icon is here used in a broad sense to refer to a depiction of physical and abstract subjects in an artwork

[2]The term abstract art is used to mean 'art that does not attempt to represent an accurate depiction of a visual reality but instead use shapes, colours, forms and gestural marks to achieve its effect', as defined by the Tate ([25])

the patterns in the predictions of the best model and treating separately the case of abstract art. Finally, the results of these different classification techniques are discussed in terms of the expressive power and limitations of modern machine learning techniques, highlighting which feature extractors are able to best represent aesthetic and abstract concepts.

1.1 Motivation

Nowadays, numerous efforts are being put into place by museums and galleries to digitise and create metadata about their artwork collection. Digitisation, however, naturally outpaces the metadata production, thus creating an imbalance in production time. To bridge to this gap, the aim of this paper is to automatically generate iconographical annotations, thus shifting the role of the archivist. In fact, with the aid of automatically generated tags, the archivist would have to annotate only a part of the corpus and, for the remaining part, only check on the automatic predictions. In addition, this categorisation structure can, at a later stage, be used as an index to retrieve artworks. Since icon metadata is still rather rare in digitised art collections, its expansion will give access to users, such as curators or scholars, to a broader selection of art, organised by their icons. Such an organisation will enable consumers to make new, unprecedented, comparisons among artworks.

Furthermore, this paper tests the performance of various multi-label classification algorithms, together with different feature extractor methods, on an under-explored task. Such an application gives the possibility to spot previously unseen limits of the algorithms. As hypothesised by [6], this analysis might highlight a tendency in appearance-based models to over-fit to one kind of image, a photograph.

Finally, this paper fosters a broader reflection on the ability of machines to recognise artistic depictions of objects. In fact, the super-human ability demonstrated by deep learning architectures at the ImageNet Large Scale Visual Recognition Challenges (ILSVRC, [1]) suggests that machines can understand the content of images. In this paper, we reflect on whether the same can be concluded about artworks and

we introduce what such an understanding implies for the complexity of abstract and aesthetic concepts that machines can learn.

Chapter 2

Related Work

In the last decade, the field of Computer Vision has witnessed significant technical development ([5]). This has prompted researchers to investigate possible applications of these new techniques to visual arts. In particular, researchers were encouraged by the success of feature learning methods, which automatically extract higher-level semantic features[1] from input images. A special interest has been devoted to the classification of paintings based on these learned features. Considering that classification efforts heavily rely on the available metadata of the paintings, these researches were mainly circumscribed to the classification of artist, style, genre and year (metadata that is available in almost all published datasets).

Already in the first decade of the century, the earliest attempts were made at classifying paintings. These first investigations, however, only dealt with low to middle-level features[2] (such as colour, light, edges). In fact, [34] classified paintings into five artistic genres based on solely technical features. They used both high-quality museum paintings and variable resolution ones retrieved from the Web, and extracted the features according to colour, texture, and edges. The authors experimented with different classification techniques, obtaining the best performance with *AdaBoosted J48*, which achieved an average accuracy of 0.7 among the 5 classes. Subsequently, [18] classified artworks into 9 artists and 3 schools of art. The features, this time,

[1] When referring to high-level or semantic-level features, this paper means features that represent shapes and objects in computer images and that extract semantic information about the objects, these are typically associated with Deep Learning methods.

[2] When referring to low-level and middle-level features, this paper means features that contain information about basic visual properties of a computer image, these include colour, gradient orientation, pixel intensity ...

were extracted using Chebyshev transform, Wavelet transform, Colour transform, FFT, and Edge detection in addition to the standard low-level features. The most significant ones were selected using Fisher scores and finally used for classification. The resulting accuracy was 0.77 for painters and 0.91 for the school of art. The success of these classification efforts raised the question of whether these low-level features could indeed be enough to discriminate on artworks.

Around the year of 2014, with the wave of success of computer vision techniques, researchers suddenly drifted away from feature engineering and moved towards feature learning. One of these first attempts was the artist attribution experiment that used the PigeoNet architecture, a neural network comprising 5 CNNs and 3 FNNs (Feed-forward Neural Networks, [28]). The network was trained on the Rijksmuseum public dataset and obtained a mean class accuracy of roughly 0.76 for the 100 artists ([28]). Although the results were comparable to those achieved by low-level features, these semantic level features showed more promise as their performance was proved to increase faster with larger datasets.

With the success of ResNet and other object recognition networks, researchers began to include pre-trained weights at the top of their networks, in order to transfer the knowledge learned by these pre-trained to their task ([7]). One of the first instances of this trend is an investigation by [10], who used AlexNet and ResNet trained on ImageNet with 20 layers of fine-tuning to classify paintings into their style. The use of transfer learning and deep re-tuning allowed, respectively, to overcome the issue of the sparsity of the data and of the heterogeneity of tasks (of style recognition instead of content recognition), obtaining an overall best accuracy of 0.62 with ResNet, over 25 classes. Along the same line, [15] conducted a more comprehensive evaluation of different pre-trained models for painting classification. In fact, they used four pre-trained architectures (VGG19, Inception-V3, Xception and ResNet50) and assessed their performance on three different tasks (attribution of author, material and artistic category), both fine-tuning and not. ResNet50 outperformed all other architectures after fine-tuning, while simpler networks, like VGG19, had the highest off-the-shelf accuracy.

Autor	Classification Task	Feature Level	Year
Zujovic et al.	genre	low	2009
Tzouveli et al.	content	low / intermediate	2009
Shamir et al.	artist & school of art	low / intermediate	2010
Saleh et al.	influence	semantic	2014
van Noord et al.	artist	semantic	2015
Lecoutre et al.	style	semantic + TL	2017
Elgammal et al.	style	semantic	2018
Gonthier et al.	content	semantic	2019
Sabatelli et al.	material, artist & genre	semantic + TL	2019

Figure 2.1: A summary table of the research context. The table presents the investigations mentioned in this section, highlighting their task, feature level and year. The table is ordered by year of publication and level of extracted features. The feature level low refers to, for instance, the extraction of colour, light, and edges; the intermediate level encompasses various techniques, such as Chebyshev transform, Wavelet transform, Colour transform, FFT, and Edge detection; the semantic level feature indicates deep learning methods in general and + TF stands for the use of Transfer Learning and pre-trained weights.

Departing from the well-investigated tasks, [16] addressed the problem of automatically finding influences and connections between artists. They produced high-level semantic features and evaluated them at the task of painting style classification. The features were then used to assess the similarity to other paintings, based also on a ground truth of temporal sequence. They discovered that semantic-level features perform best at the task and found some new, plausible, correlations. In 2018, [2] together with classifying artistic style, analysed the extracted features in correlation to concepts in art history. To be able to interpret the results, the model was forced to maintain a low number of dimensions. They used principal component analysis (PCA) and showed that 10 modes contain 95% of the variation. Retaining the 3

modes of highest variation, they also found that the Pearson correlation coefficient (PCC) of the modes with time and with Wölfflin's artistic concepts[3] was high ([2]).

Despite the efforts at classifying paintings are numerous, to the best of our knowledge, there is no generalised attempt to classify paintings into their content using high-level features. [27] attempted a semantic classification of cultural content, specialising in medieval icon art. However, their analysis was specific to Byzantine iconography, which has low variability of characteristics and a well defined iconographical code. The features were extracted from the machine-learned segmentation of the image and from the colour, density, length, and form of the segments and classified using fuzzy description logics (DLs), that explicitly encode the semantic interpretation. The classification obtained, on average, a precision of 0.8 and a recall of 0.7. Differently from [27], this paper adopts semantic-level features and the heterogeneous collection of the Tate for content classification. In 2019, [4] also implemented an object recognition algorithm for paintings. Despite similar to this paper in intention, the latter focuses on both recognising and localising objects in artworks, using R-CNN (Region-CNN) and MIT (Multiple Instance Learning). The detection is evaluated on a dataset of only classical paintings, achieving an average precision above 0.6. The examination carried out by this paper is, hence, a generalisation of the two previous attempts, that introduces a wider variety of training techniques and a more diversified dataset.

[3]These concepts have been introduced by Wölfflin in 1915 in his famous book 'Fundamental Concepts of Art History', where he delineates 5 notions (and their antitheses) of artistic style: 1. linear-painterly, 2. place-recession, 3. closed-open form, 4. multiplicity-unity, 5. absolute-relative clarity

Chapter 3

Methods

In light of the great performance achieved by Convolutional Neural Networks (CNNs) at ILSVRC ([5]), this paper limits the comparative study to high-level semantic features extracted by deep CNN's. To provide an understanding of the classification methods used by this paper and the reasons why they have been chosen, this section is made up of two components: An introduction to transfer learning, with a particular attention devoted to VGGNet, InceptionV3, ResNet; And an exploration of two multi-label classification techniques, with a description of the baseline model.

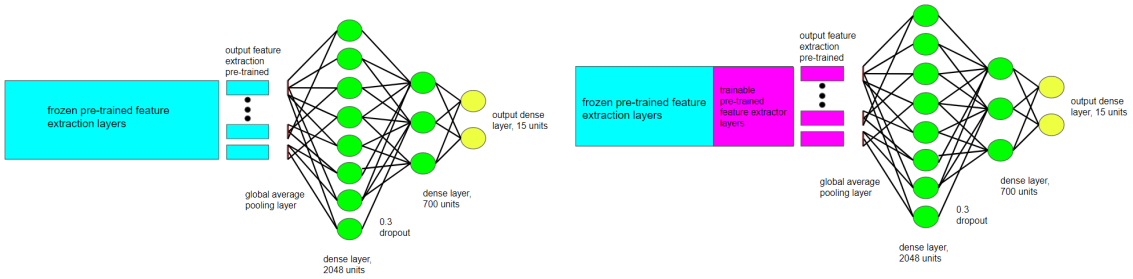
In addition to the information in this section, Appendix 1 contains the description and exploration of two hierarchical methods that we implemented in order to classify at the second level of Tate's Subject Index, that will be further explained in the following section.

3.1 Transfer Learning

Transfer learning (TL) is a deep learning technique used, predominantly, to improve the performance of a model when the training data is scarce. Formally speaking, in a supervised learning setting with an input target space X, an output space y and a function $F : X \rightarrow y$ that minimises the expectation of a given loss, TL adds a second input space X', the source space, and a second output space y'. Normally, at least one of $X \neq X'$ and $y \neq y'$ is true. The goal of TL is to find a better F' that exploits the source data and, possibly, the target data ([5]).

Given the scarcity of annotated paintings data, this paper assesses whether using

transfer learning can improve the model performance. This is done by adding weights that have been pre-trained on ImageNet as source data. These are used in substitution, or as activation, to the feature extraction phase of our classification model. We test, therefore, whether there is a substantial similarity, in terms of extracted features, between the pictures in ImageNet and the target data of paintings.



(a) Off-the-shelf model architecture

(b) Fine-tuned model architecture

Figure 3.1: A visualisation of the model architectures both in the case of fine-tuning and not. As visible in the picture, only the feature extractor layers of the pre-trained are used, while the classification is added by us. The classification is made up of a global average pooling layer, two hidden fully connected layers and an output layer.

Among the top competitors of ILSVRC, this paper uses three radically different architectures: VGGNet, InceptionV3, and ResNet ([19, 22, 7]). The saved architectures and their weights are retrieved from the Tensorflow 2.1 `tf.keras.applications` module. The weights are imported without including the fully connected layers, which we substituted with a global average pooling layer, two dense hidden layers, with dropout after the first, and a dense output layer of the size of the classification classes (Figure 3.1a). The pre-trained models are tested with and without fine tuning. The fine-tuning involves the last 1/2 of the pre-trained network layers for reasons of limited GPU power. This means that the last 1/2 of the convolutional layers are trained, while the first 1/2 is frozen (Figure 3.1b). These specific networks have been chosen due to the great diversity in their structures, as such diversity allows this paper to test a wide variety of extracted features.

3.1.1 VGGNet 16

VGGNet is a simple network with 16 convolutional layers and a constant 3x3 stride. The first three blocks consist of two convolutional layers and one max pooling layer,

while the last three have one extra convolutional layer in each block ([19]). Each convolutional layer uses a rectified-linear activation unit, *ReLU*. The strength of the architecture resides in the large receptive field that is achieved, despite the small filter size, by using three convolutional layers in a row. This way, the network learns semantic level features, maintaining the amount of parameters low (the convolutional blocks have in total 14 million parameters).

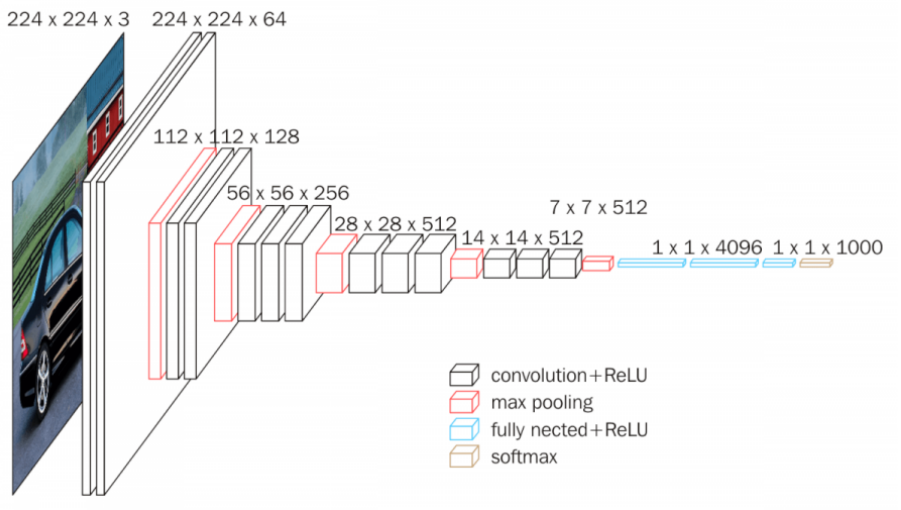


Figure 3.2: An illustration of the architecture of VGGNet 16. Retrieved from: researchgate.net.

3.1.2 Inception V3

InceptionV3 is a largely paralleled network that consists of 11 inception modules. Each module computes concurrently a 1x1, 3x3 and 5x5 convolution and a 3x3 pooling. The results are concatenated and passed to the next module. In order to reduce the size of the convolutional layers, both the 3x3 and the 5x5 convolutional layers are first passed through a bottleneck 1x1 convolution ([22]). Thanks to the addition of the bottleneck, the complete network (excluding the fully connected layers) has 21 million parameters.

3.1.3 ResNet 50

ResNet is an extremely deep cascading network that learns via micro residual modules. In particular, each residual module learns, subsequently, a 1x1 bottleneck, a

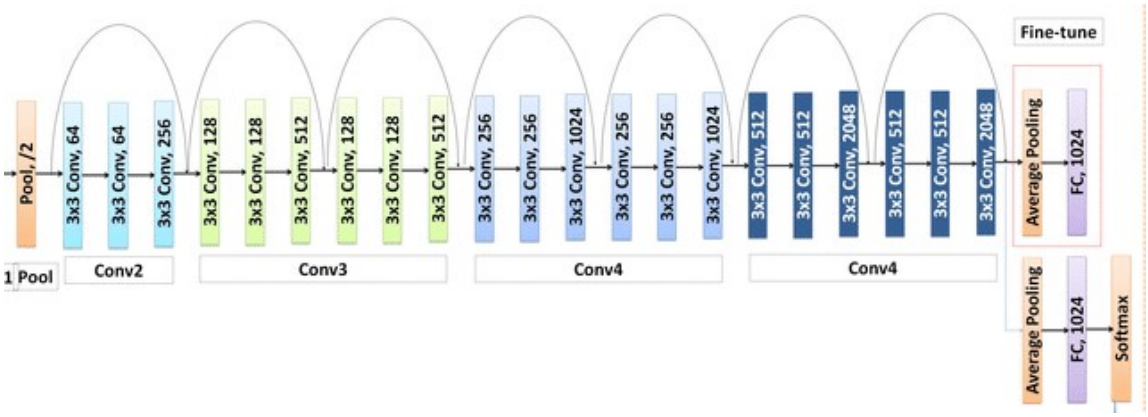


Figure 3.3: An illustration of the architecture of Inception V3. Retrieved from: Medium.

3x3 and another 1x1 bottleneck residual and it adds it to a skip connection (an identity mapping). The use of residuals and identity mappings is introduced to avoid the vanishing gradient problem and to allow the architecture to adapt to the complexity of the task or data. This is achieved, effectively, by learning to use only some portions of its representational capacity ([7]). The network used by this paper is a 50 layer deep variation of ResNet with 23 million parameters excluding the fully connected layers.

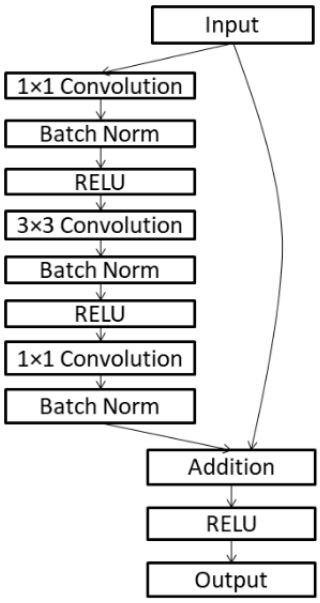


Figure 3.4: An illustration of the architecture of ResNet 50. Retrieved from: researchgate.net.

3.2 Multi-label Classification

Classification into the subjects represented in an image is, intrinsically, multi-label. In fact, paintings most often contain more than one subject, making the classification inherently multi-class multi-label, as shown in Figure 3.5. In terms of multi-label

Figure 3.5: Multi-label paintings. As shown in picture, each image has one or more tags.

classification algorithms, there are multiple approaches to modelling and their performance varies largely depending on their application. The approaches include graph based models, Recurrent Neural Networks (RNNs), and implicitly modelled label dependencies using attention mechanisms [12, 31, 30].

Since the labels of Tate's Subject Index, the target of the classification, are designed to have a limited number of labels, only the strictly necessary ones, and a minimal semantic overlap between labels, we focus here on the simplicity and scalability of the network rather than its capacity to extract complex relations between labels. For this reason, we evaluate the performance of the Naive approach and of the CNN-RNN architecture ([30]). The first method is an adaptation of multi-class neural network classifiers and, therefore, the simplest multi-label classification method. The second has been chosen among state-of-the-art multi-label algorithms because of its scalability and efficiency. Differently from graph based architectures that model label relationships explicitly in a network, this RNN model with attention learns label sequences implicitly, thus reducing the output space (which is combinatorial in the Naive version) and preserving flexibility in the dependencies. These label sequences

determine which label combinations are more probable, aiding the classification in cases where one of the labels is easily recognisable and the other can be deduced by the presence of the first. For example, when the algorithm spots People, it will be more likely to understand whether Society or Work & Occupations are present.

3.2.1 Naive Multi-label Classification

The Naive approach treats each label in isolation, as a binary classification problem. In fact, the multi-label problem is converted into a set of binary problems. With respect to multi-class classifications, this model uses a different loss and a different activation function to the output node. In order to treat each output as an independent Bernoulli distribution, the categorical cross-entropy is substituted by a binary cross-entropy:

$$H'_y(y) = -\frac{1}{N}\sum_{i=1}^{N} y_i \times \log(p(y_i)) + (1 - y_i) \times log(p(1 - y_i)) \tag{3.1}$$

Where y is the single label (1 or 0) and $p(y)$ is the predicted probability of the image being 1 for all N images.

In terms of the final activation function, instead of the standard *softmax*, a *sigmoid* with a threshold at 0.5 is used, since it allows to penalise each output node individually.

$$h(x) = \frac{1}{1 + e^{-x}} \tag{3.2}$$

The output of the *sigmoid* is a value between 0 and 1, and it is centred at 0.5.

The Naive model performs well on multi-label classifications where there is low to no dependency between labels. In addition, its performance can be greatly increased using TL.

As a baseline, this paper uses a CNN architecture comprising four convolutional layers with a number of nodes 64, 64, 128, 128 respectively, with *Glorot* initialisation, $ReLU$ activation, 3x3 stride, batch normalisation and dropout of 0.2 and max pooling after each convolutional layer ([8, 21, 11, 13]).

3.2.2 CNN-RNN Multi-label Classification

Despite being computationally efficient and easy to implement, the Naive method fails to explicitly model the co-occurrence of labels, sometimes leading to an impossible, or unlikely, combination of predicted labels. In fact, in some cases, such as the aforementioned classes of People and Society, it would be useful for the model to understand their relationship. CNN-RNN has been proven to obtain leading-edge performance by modelling a joint image-label embedding and label co-occurrence ([30]).

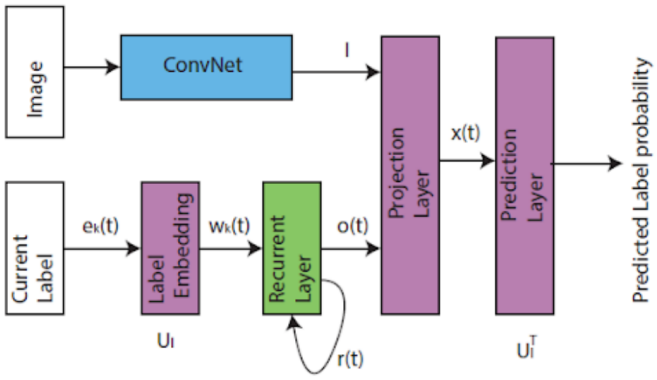


Figure 3.6: An illustration of the architecture of CNN-RNN [30]

Specifically, at the first iteration of each RNN cycle, the CNN module with attention extracts high-level features and predicts the most probable label. Successively, the model employs an LSTM (Long Short Term Memory) as RNN to predict the next applicable label. This is repeated until the algorithm cannot find any candidate prediction paths. The next applicable label is computed by modelling correlations among labels, by iterating over the possible prediction paths of all the labels. Here, a prediction path is a sequence of labels $(l_1, l_2, l_3, ..., l_N)$, where the probability of each label l_t depends on the image I and the labels that have already been predicted $l_1, ..., l_{t-1}$. The paths are found using label embedding and k-nearest neighbour and selected using Beam search. The latter maximises the posterior probability of the prediction path given the input image:

$$l_1, ..., l_k = \arg\max_{l_1,...,l_k} P(l_1, ..., l_k|I) \tag{3.3}$$

where k is the length of the prediction path.

It is trained with *softmax* as final activation, *categorical cross entropy loss*, *rmsprop* optimisation.

Chapter 4

Dataset

This paper adopts the publicly-available Tate Collection ([25]) for its experiments. It contains metadata and image URLs of over 70,000 artworks owned by the Tate jointly with the National Galleries of Scotland. Basic metadata is available in CSV format and some more elaborate metadata (including the Subject index) in separate JSON files. The images that were retrievable from the URL are 40,000 and these were available both in thumbnail format (256x256 pixels circa) and in HD (1500x1500 pixels circa). The artworks are organised in 6 subsections: A, AR, D, N, P, T, of which section D has been discarded, as it mostly contains black and white pencil drawings. The filtered dataset contains 25.000 images.

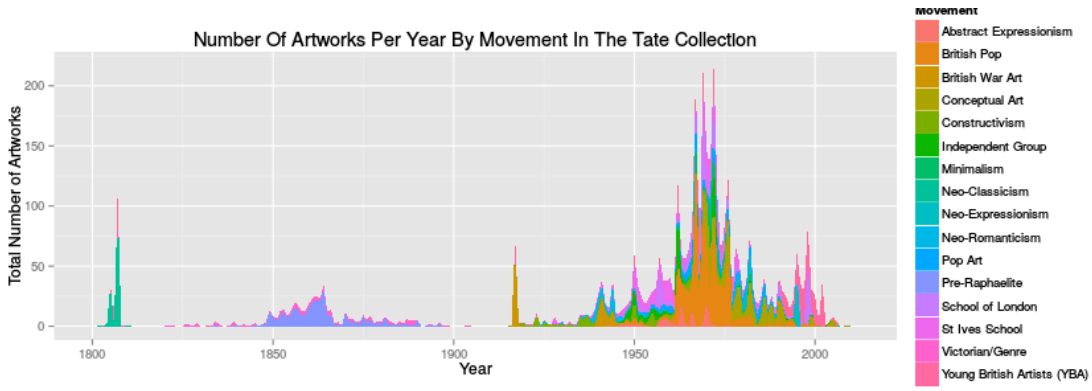


Figure 4.1: Distribution over time and style of the paintings in the Tate Dataset. As visible in the plot, the paintings are made between the XIX-XXI century and belong mostly to British art movements. Retrieved from robmyers.

The collection has been published on GitHub as part of the 5-year project of 'digital access, participation and learning with archives' ([24]) that started in 2012. The metadata has been dynamically updated until October 2014, when the GitHub re-

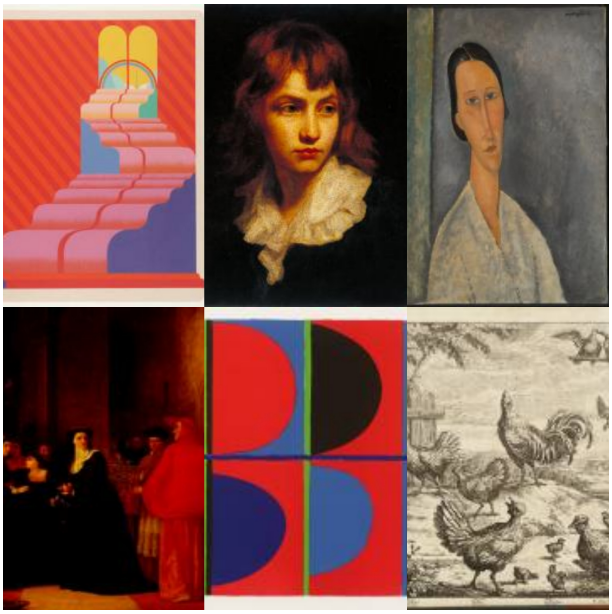


Figure 4.2: Example images from the Tate Dataset. *From top left to bottom right*: Map No. III Ariadne's Thread by Jens Lausen; Master William Opie by John Opie; Madame Zborowska by Amedeo Modigliani; Cardinal Bourchier Urges the Widow of Edward IV to Let her Son out of Sanctuary by John Zephaniah Bell; Red, Blue, Green, After by Sir Terry Frost; Cock; Hen; Chicken; Dove, by Francis Barlow. Already from these images, the variety of paintings is clear.

pository stopped being maintained. The project, Archives & Access, has been funded by the Heritage Lottery Fund (HLF) and Tate. It focused on the digitisation of the Tate Collection (which is currently the world's largest archive of British art) and the development of interactive activities at Tate.

4.1 Target data

The metadata concerning the subject(s) of the artworks (the Subject Index) is used by this paper as target for the classification. The index is extracted from a subsection of the JSON files corresponding to each artwork.

As one can see from the example in Figure 4.3, the subject metadata is extremely detailed. It is structured in the form of a tree in which each applicable node has a variable number of children and the children can also have children. In addition, each node of the tree has a corresponding numeric ID. Empirically, we observed that the largest depth is three (considering the root as depth zero). The nodes at depth

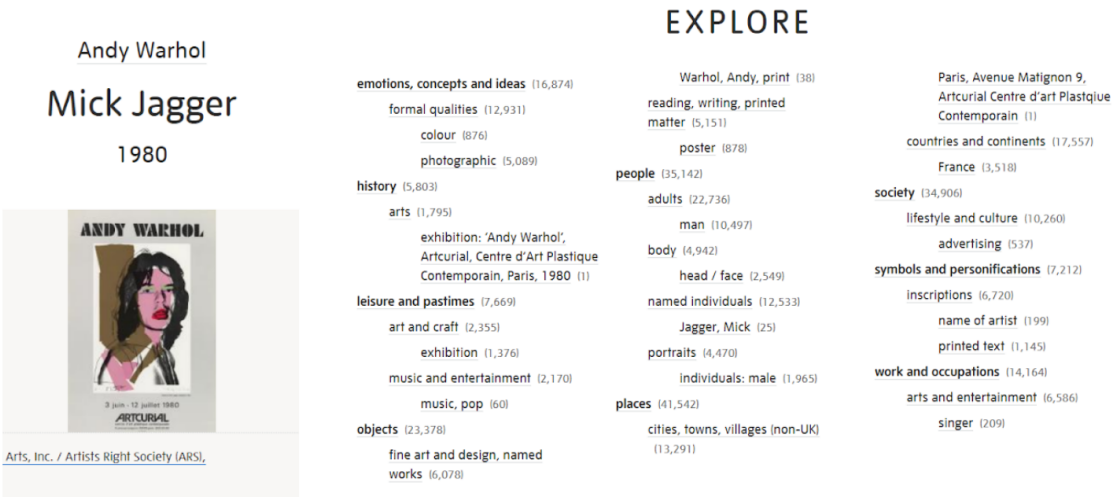


Figure 4.3: On the right, an example of the Subject index and, on the left, the corresponding artwork. Retrieved from: Tate.

one denote different domains contained in the image. In the instance above, the subsets under People indicate that the image is a portrait of a male individual, Mick Jagger. Exceptionally, the category Places does not indicate a visible place in the image, rather, it encodes the text on the bottom of the poster. When making the array of labels, the class Group/Movement has been disregarded as it contains only one image. Each target array $y_i \in N$, where N are the images, is stored and passed to a data-handler in the form $y = [y_1, ..., y_i, ..., y_C], i \in 0, ..., C$ where C is the number of classes and y_i is 1 if the image contains that class and 0 otherwise.

4.2 Dataset Statistics

The dataset is, in general, sparse. At depth one, the 15 classes have a number of entries that ranges from a minimum of 1,352 to a maximum of 38,358 images. The table below is aimed at giving an impression of the distribution of artworks per class and their overlap with other classes:

Table 4.1: Dataset Description

Class Name	Number of images	Number of Sub-classes (depth 2)	Percentage of images with that class in training set	Percentage of images with that class in test set	Class with highest overlap, images overlapping
1. People	38358	13	58%	56%	Objects 7160
2. Objects	16322	21	38%	38%	People 7160
7. Nature	24857	18	49%	46%	People 6952

Table 4.1: Dataset Description

Class Name	**Number of images**	**Number of Sub-classes (depth 2)**	**Percentage of images with that class in training set**	**Percentage of images with that class in test set**	**Class with highest overlap, images overlapping**
6. Society	10295	15	29%	28%	People 5539
10. Work and Occupations	4951	14	16%	16%	People 3956
4. Architecture	15426	14	32%	32%	Nature 6063
13. Leisure and Pastimes	2641	5	9%	9%	People 2010
8. Emotion, Concepts and Ideas	8318	3	24%	23%	People 3303
11. Symbols & Personifications	2819	8	9%	9%	Objects 1200
9. Interiors	1978	5	7%	8%	Objects 1458
5. Abstraction	9322	2	29%	29%	Emotion, ... 2663
12. Religion & Belief	2862	10	7%	7%	People 1735
3. Places	17353	5	26%	26%	Nature 5282
14. History	1352	3	5%	5%	Society 889
15. Literature and Fiction	2385	3	8%	9%	People 1467
Total	159240	139	-	-	-

As one can see from the table, the overlap between classes is significant. The most striking example is the class Leisure and Pastimes, that overlaps with People almost 80% of the time. Each image, on average, has 6 classes and very rarely has only one. The classes at depth two and three are additionally sparse and the number of classes mushrooms. In fact, at level three, the classes are 2442, with a minimum of 1 and a maximum of 7145 entries. The average number of entries is 15 images and the median is barely 2.

Chapter 5

Results

5.1 Experimental Setup

In our experiments, we use the Tensorflow 2.1 deep learning framework. In terms of pre-processing, the images are resized to a 224x224x3 input shape, normalised between 0 and 1 and passed to the model with batch size 56. The data is split into training, validation and test set using a random shuffle (which maintained approximately the same distributions in all sets). The training set contains 18000 images, the validation 3000 and the test 4000. We ran all the models for 60 epochs.

In addition, in order to penalise the model more heavily when miss-classifying an instance that pertains to a class with fewer training images, we ran the off-the-shelf models including custom class weights to the loss. The class weights have been computed as follows:

$$w_i = \frac{\sum_{j=1}^{N} n_j}{n_i} \tag{5.1}$$

Where w_i is the weight associated with class i, n_i is the number of training images of class i and N is the number of classes.

5.1.1 Evaluation Metrics

The predicted labels are computed by applying a 0.5 threshold to the predictions of the model (1 if $p'_i > 0.5$, 0 otherwise, where p' is the predicted array) and are evaluated against the ground-truth labels. The adopted metrics include overall accuracy

(O-A), F1 score (O-F1), precision (O-P), and recall (O-R). Additionally, the predictions on the test set are also evaluated using the per class accuracy (P-A) and per class precision (P-P), recall (P-R) and F1 (P-F1). The overall metrics are computed as an average among all samples in the set (micro average), while the per class metrics refer to the average grouped by each class (macro average). These metrics have been chosen to give a comprehensive understanding of the performance of the model and to provide an indication of the performance of each class. These are computed using the `sklearn.metrics` package, that treats the labels as a collection of binary problems, one for each class.

5.2 Model Comparison

Table 5.1 shows the results of the classification experiments previously introduced.

Model	**O-A**	**O-F1**	**O-P**	**O-R**	**P-A**	**P-F1**	**P-P**	**P-R**
Baseline	0.83	0.27	0.37	0.25	-	-	-	-
VGG16 no tuning	0.85	0.51	0.60	0.48	0.71	0.32	0.39	0.27
VGG16 class weights	**0.86**	0.50	**0.62**	0.48	0.71	0.32	0.37	0.34
VGG16 fine tuning	0.85	**0.52**	0.58	**0.51**	0.71	0.34	0.41	0.31
Inception V3 no tuning	0.82	0.37	0.50	0.34	0.73	0.34	0.40	0.29
Inception V3 class weights	0.83	0.40	0.51	0.35	0.74	0.33	0.42	0.27
Inception V3 fine tuning	0.85	0.49	**0.62**	0.44	0.72	0.32	0.37	0.29
ResNet50 no tuning	0.81	0.42	0.47	0.42	0.71	0.34	0.36	0.33
ResNet50 class weights	0.83	0.42	0.52	0.39	0.72	0.34	0.38	0.31
ResNet50 fine tuning	0.84	0.47	0.57	0.43	0.71	0.31	0.35	0.28
CNN-RNN	0.82	**0.52**	0.57	**0.51**	**0.83**	**0.54**	**0.57**	**0.51**

Table 5.1: Results Table. The results are organised according to the pre-trained used and the training method. *No tuning* refers to the models trained with the convolutional layers of the pre-trained frozen, *fine tuning* is the model in which 1/2 of the convolutional layers are re-trained and *class weights* is the model, with no tuning, that is trained with a weight associated to each class. Finally, the CNN-RNN model uses VGG as pre-trained.

Looking at Table 5.1, the model that performs best overall is the CNN-RNN implemented using VGG as pre-trained. The performance improvement is clearly demonstrated in the per-class metrics, which surpass the metrics of the other models by more than 15%. However, the CNN-RNN model does not perform better than the Naive approach on the overall metrics, indicating that the predictive power is not as

varied among classes when compared to the other methods. In fact, while the other architectures do not predict accurately on numerous classes, CNN-RNN has a precision below 25% only for Literature & Fiction and History. The great performance of this method provides evidence of the utility of modelling label-to-label correlations next to the image-to-label ones for our task.

When comparing the baseline model[1] to the models that use transfer learning, the addition of ImageNet as source data has proven to be valuable, obtaining a significant improvement in all cases. In fact, all the pre-trained have improved over the baseline model by, at least, 10% on O-F1, O-P, O-R; reaching a 25% improvement in the case of O-P with VGG and InceptionV3. Although the architectures of the baseline model and of the pre-trained ones are not identical, in broad terms, this improvement indicates that VGG, InceptionV3, and ResNet generalise sufficiently well on artistic images. These results are not surprising as already [3] had shown the substantial improvement of using transfer learning on related tasks. Similarly to [3]'s results, VGG16 has proven to obtain consistently the best off-the-shelf performance, while ResNet50, which performed extremely well in [3, 10], was the least interesting in our case.

In terms of the results achieved by fine-tuning the model, Table 5.1 shows that InceptionV3 and ResNet benefited greatly from re-training the second half of the convolutional layers. The improvement, in both cases, reaches 10% in O-P. O-R and 0-F1 have been consistently improved for all pre-trained. This result, once again, confirms the investigations by [3, 10], who witnessed a considerable improvement when fine-tuning ResNet. The minimal improvement in VGG suggests that the features extracted are already largely applicable to content classification on artworks and do not need fine-tuning.

The improvement induced by the implementation of custom class weights is modest but consistent, indicating that the additional focus on lesser represented classes aids, to some extent, the training.

[1]The baseline model is trained from scratch with initial weights randomly sampled from a Gaussian distribution

Figure 5.1: A comparison between the different approaches in terms of their precision on the validation set over the first 60 epochs. We observe that off-the-shelf VGG with class weights and InceptionV3 with fine-tuning perform best, while the performance of both off-the-shelf and with class weights ResNet have the lowest precision.

When comparing the learning curves in Figure 5.1, we observe that the improvement rate generally settles after 20 epochs and decreases radically already after the first 7-8 epochs. In the case of ResNet and InceptionV3, fine-tuning overtakes their counterparts in around 25 epochs. All the models show an oscillating growth over the epochs, indicating that the learning rate was too high. Some experiments will be put in place in the future with a lower or a custom learning rate to test whether this change would result in a gain in performance.

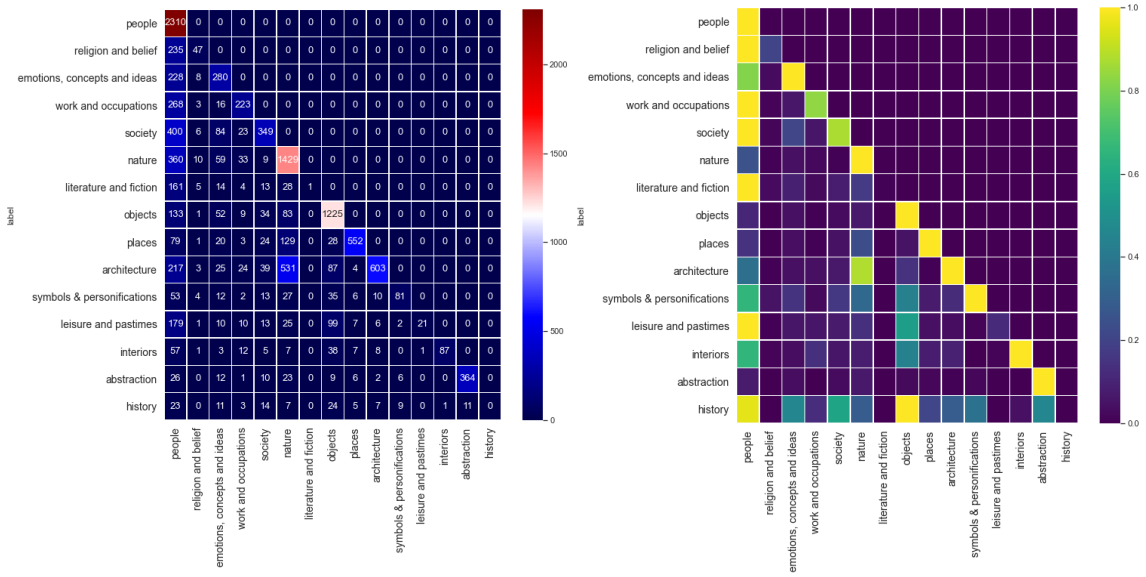
In sum, these findings seem to indicate that the features extracted by VGG16 are the most adequate for a content classification of artworks. Furthermore, the little improvement of fine-tuning on this pre-trained model shows that the features extracted in the last layers of the network are already satisfactory for artwork classification.

5.2.1 Error Analysis

Below CNN-RNN predictions will be elucidated based upon performance variability across classes and artistic movements. We then proceed to determine the model's accountability in the subsequent paragraphs.

Although the accuracy of the method suggests that the predictions are, in most cases, correct and reliable, the variation in the performance among classes is significant, raising the questions of: What classes are easier to classify? Which ones are the most problematic and why?

Setting aside the class Abstraction for a now as it will be discussed more in detail in the next section, the confusion matrices in Figure 5.2 show that the classes of People (0.73 precision), Nature (0.74 precision), Objects (0.58 precision), and Architecture (0.67 precision) display few errors in the predictions, both in absolute terms and especially relatively to the number of test samples in the class. An acceptable performance is obtained, remarkably, also for Emotions, Concepts & Ideas (0.51 precision), while the remaining classes are often mistaken. On these classes, we identify the most important mis-classifications on our test set: these classes are frequently wrongfully labelled as People. This seems to imply that the concerns expressed in the Introduction were correct: Classes which do not refer to a specific, identifiable,



(a) Absolute Confusion Matrix Heat-map of CNN-RNN

(b) Relative Confusion Matrix Heat-map of CNN-RNN

Figure 5.2: A heat-map visualisation of a confusion matrix of the predictions of CNN-RNN. Being a multi-label classification, the confusion matrix is computed by adding 1 if the prediction of label i of the image j is 1 and the ground-truth is also 1. When the ground-truth is 1, but it has not been predicted, to all of the mis-predictions are added $1/n_j$ where n_j is the number of mis-predictions on the image j. The normalisation is done row-wise, by dividing each entry in a row by the maximum of that row. NB: The classes are not in any specific order.

physical entity but rather to abstract, less clear-cut, concepts are harder to discern for an algorithm. Although it is still to be determined whether this factor was actually at play, or whether this imbalance was due to these classes having only a couple of hundred samples in the training set, it is safe to say that the predictions on these classes are not reliable and a lot more work has to be put into improving these predictions.

As aforementioned, the class predictions made by the CNN-RNN model demonstrate variability, thus a closer examination of specific cases could be indicative of where and why the algorithm fails.

In Figure 5.3, the predictions are rather accurate. In fact, the drawings on the middle and bottom right column are either fully correct or almost. In the other cases the most apparent elements are successfully recognised, while the algorithm fails to detect both the less clear depictions of some physical elements (an example is Architecture in Study for 'Moving Day' and Objects in A girl seated on rocks in a wood); and more abstract elements (such as Emotion, Concepts & Ideas and Symbols & Personifications). The errors of the first type are understandable because the elements could be considered difficult to recognise even for human understanding. The second type of error manifests a difficulty of this algorithm to accurately differentiate among the more abstract classes and, possibly, to individuate what, in an artwork, is linked to that class. Finally, it appears that images that are more similar in style to the source data of our pre-trained are more accurately predicted, i.e. artwork 4 and 6 of Figure 5.3, thus indicating that the performance of the model decreases as the style of the painting becomes more abstract. This would imply that there is an almost linear downwards progression over time, with a significant decrease caused by the formation of the Avantguardes[2].

Figure 5.5,5.4 represent the precision respectively by artistic movement and over time. The figure provides evidence that the progression over time does not fully satisfy the above hypothesis, as there appears to be an undulatory movement with two peaks at 1740-50s and 1860's. Focusing on the decrease from from 1860's onwards,

[2]For more information about the Avantguardes mentioned in this paragraph you may refer to The Story of Art.

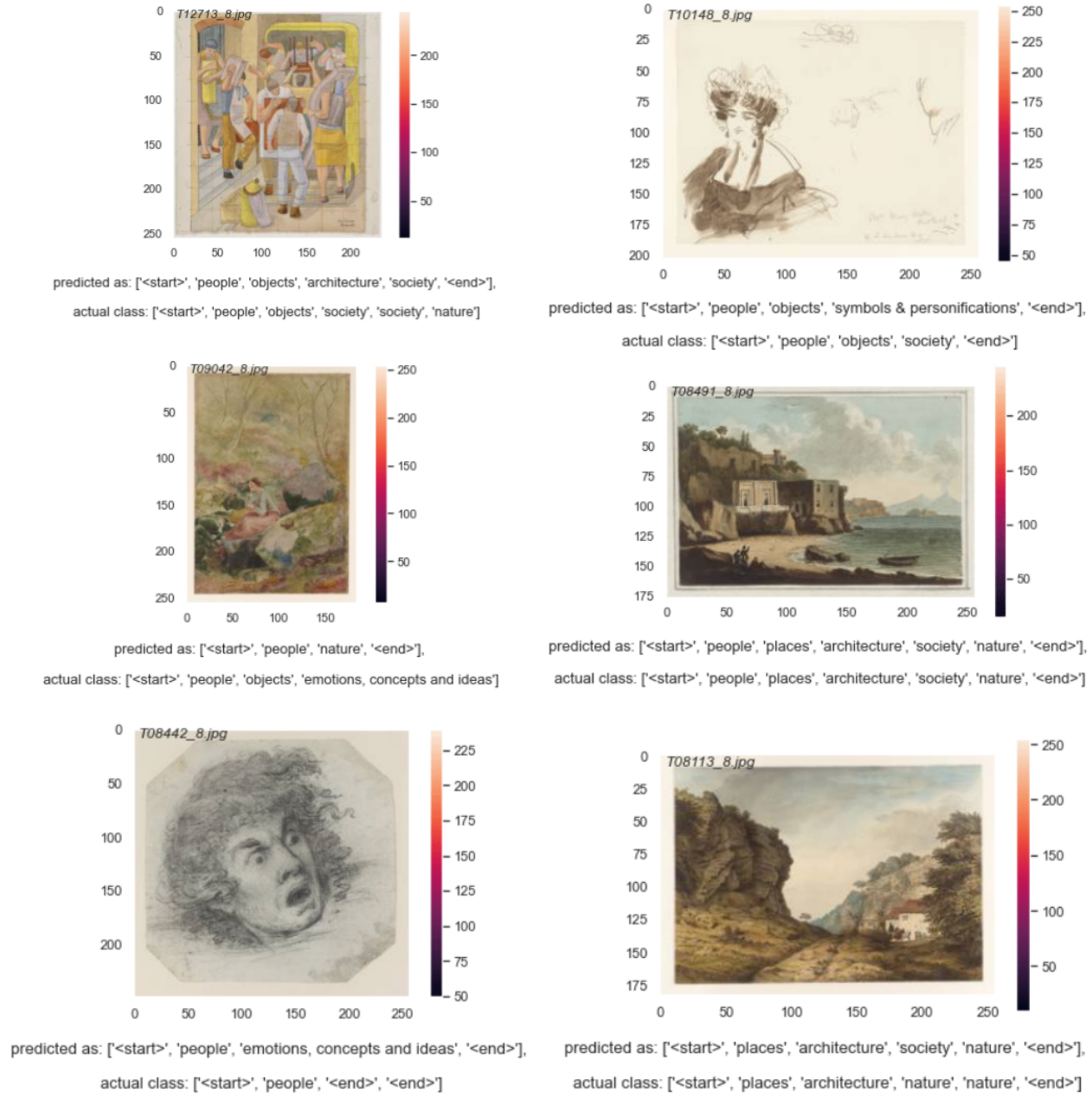


Figure 5.3: A collection of predictions by CNN-RNN. *From left to right, top to bottom:* 1. William Roberts, Study for 'Moving Day', 2. Sir Edwin Henry Landseer, Portrait of Mrs Henry Wells of Redleaf and Sketches of a Man's Head in Profile, Leaning on his Hand, 3. John William Inchbold, A Girl Seated on Rocks in a Wood, 4. John 'Warwick' Smith, From Pausilipo 5. Sir Nathaniel Dance-Holland, Study of a Head with an Expression of Horror, 6. Samuel Hieronymous Grimm, Cresswell Crags, Derbyshire

it is plausible that the following transformations in style caused difficulties for the algorithm:

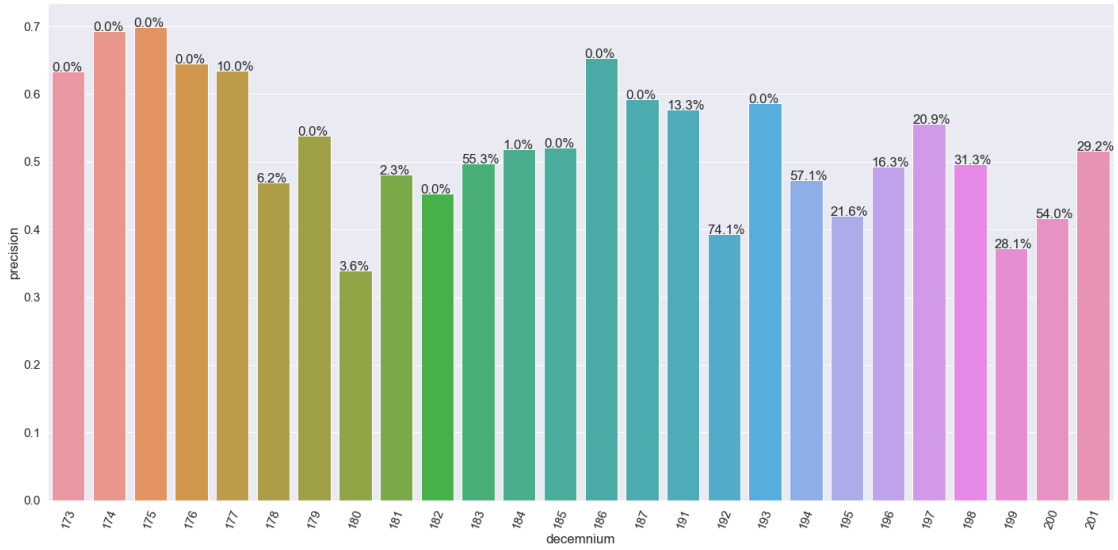


Figure 5.4: A bar plot with confidence intervals of the precision per year of CNN-RNN. The percentage on each bar represents the coverage of the movement data for that decennial.

- the shift from a truthful depiction of reality to a general depiction of an impression, around the years of 1860-70s, mostly with Impressionism ([26]).
- the shift towards the simplification to elementary geometrical shapes, around the years of 1890-1920s, predominantly with Cubism ([26]).
- the shift towards anti-naturalistic colours, around the years of 1890-1920s, predominantly with Expressionism ([26]).

Through these movements, the representation has drifted away from the external and approached the internal, expressing the inner with the outer ([9]). The simplification in the shapes and the use of un-natural colours may have mislead the algorithm, that, for instance, encountered more difficulties at classifying the artworks 1 and 3 of Figure 5.3 than the others.

In order to understand whether this assumption is valid, we plotted the performance by movement in Figure 5.5. We have to acknowledge that the information by movement in the metadata was scarce and biased towards more recent artworks, its coverage is plotted in Figure 5.4. Although the good performance of artworks that belong to a realistic movement, such as Pre-Raphaelite artworks and Baroque ones, confirms the impression that a more faithful representation of reality would be easier

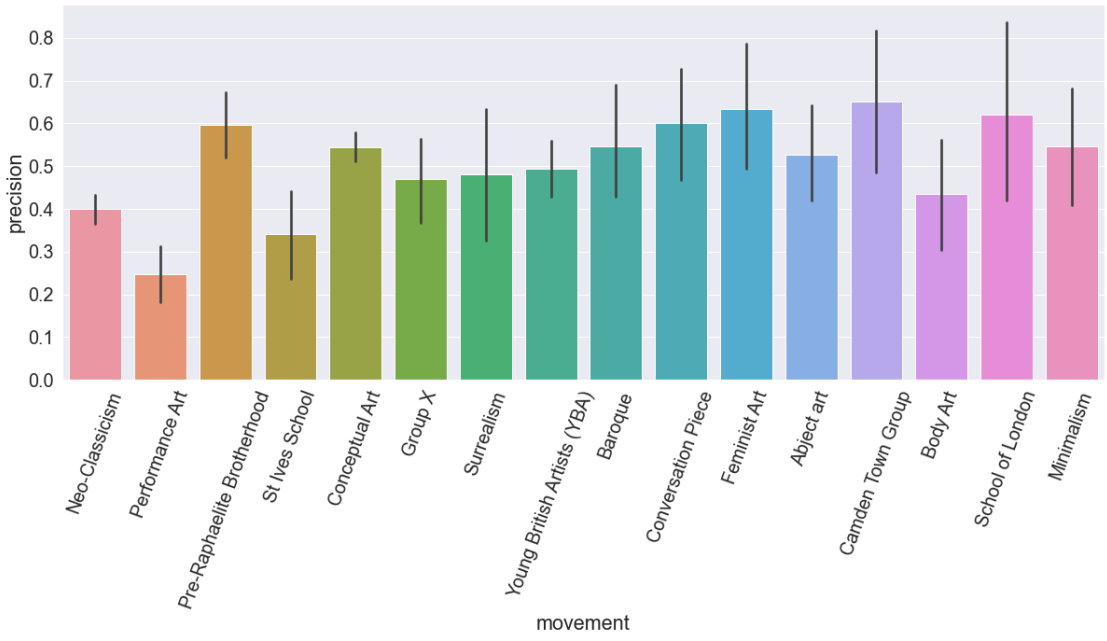


Figure 5.5: A bar plot with confidence intervals of the precision per movement of CNN-RNN. Only the movements that appear more than 10 times in the test set are kept.

to classify, the low precision of the realistic Neo-Classicism and the high precision on the Camden Town Group, a post-impressionist style, show that this assumption is not always satisfied. Overall, the hypothesis may be valid, but more investigation has to be carried out to conclude with confidence whether the abstraction in style creates additional hardship for the algorithm. Furthermore, the fact that the demarcation is not very clear indicates that CNN-RNN is able to generalise remarkably well on different artistic movements and periods.

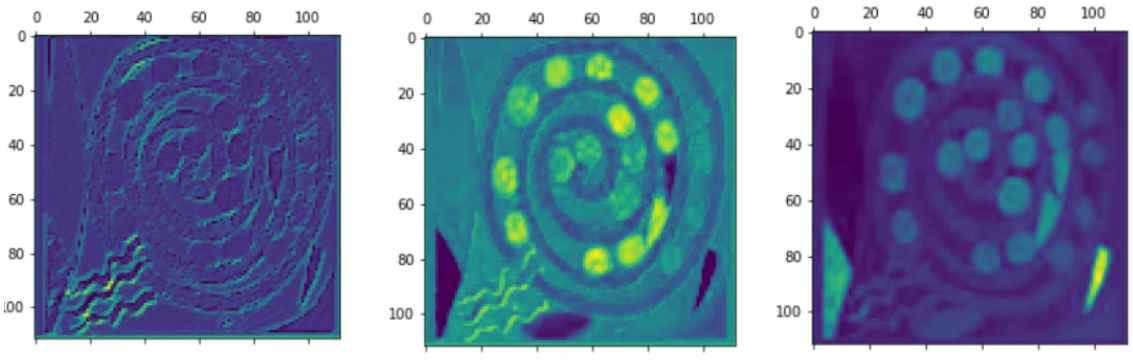


Figure 5.6: Some example of activation maps of VGG16 on a painting of a shell. One can clearly see how the rightmost activation is reacting to triangles, the middle to circular shapes and the last to the curls of the shell.

In order to validate our findings, we visualised some activation maps, saliency maps, and filters for VGG. This is important as we would like to be able to account for the reasons behind the predictions of our algorithms. Although neural networks are still generally considered as 'black boxes' ([5]), there are methods to understand which filters are being used, how the image is reacting to them, and what areas of the images are contributing most to the prediction. The Figure 5.7 is obtained using Activation Maximisation, which produces an image that maximises the output activations of the filter. In this way it is possible to obtain a rough idea of what sort of patterns the filter is looking at. The image is obtained using the package `tf-keras-vis`. To understand how the image reacted to the filters, we used activation maps, the outputs of each convolutional layer before the rectified linear units. These indicate which areas of the image reacted to the filter applied at that layer. Finally, to look at which areas of the image were the most influential for the prediction, we visualised a saliency map proposed by [20] and implemented in `tf-keras-vis`, which uses the gradients to determine the focus areas in input regions that, if changed, create the most change in the output. In this way, the saliency highlights the areas that contribute the most to the output. The areas where the most important features were extracted were visualised using GradCAM, an algorithm proposed by [17] and implemented in `tf-keras-vis`. The algorithm uses the gradients of the output and back-propagates them into the last convolutional layer. This identifies the areas of the last convolution that are most influential for the output.

As one can see in Figure 5.6, the image is correctly reacting to the filters learned by VGG16. Furthermore, Figure 5.7 reveals that the filters in the convolutional layers of VGG16 are rather complex. Take as an example the activation on the curls of the shell and the filters on the bottom row of Figure 5.7, the patterns extracted are unusual and not geometric. Such filters show that, together with the typical Gabor-like and blob-like filters that are often learned on ImageNet ([2]), VGG16 also individuates more design-like shapes, which turn out to be useful for artwork classification.

Looking at Figure 5.8, the algorithm correctly recognised the presence of a Person and an Object in the picture. The saliency map and the GradCAM visualisations

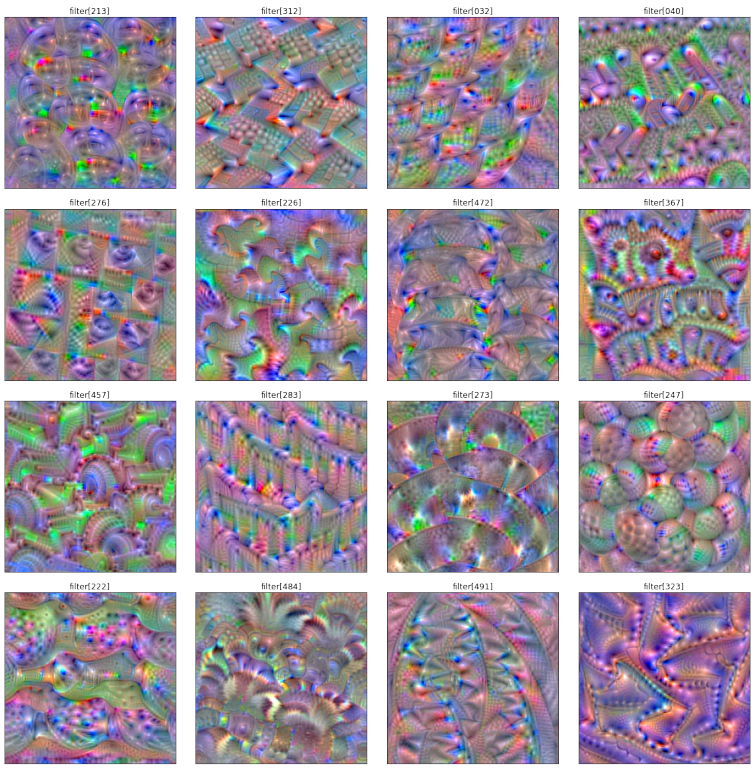
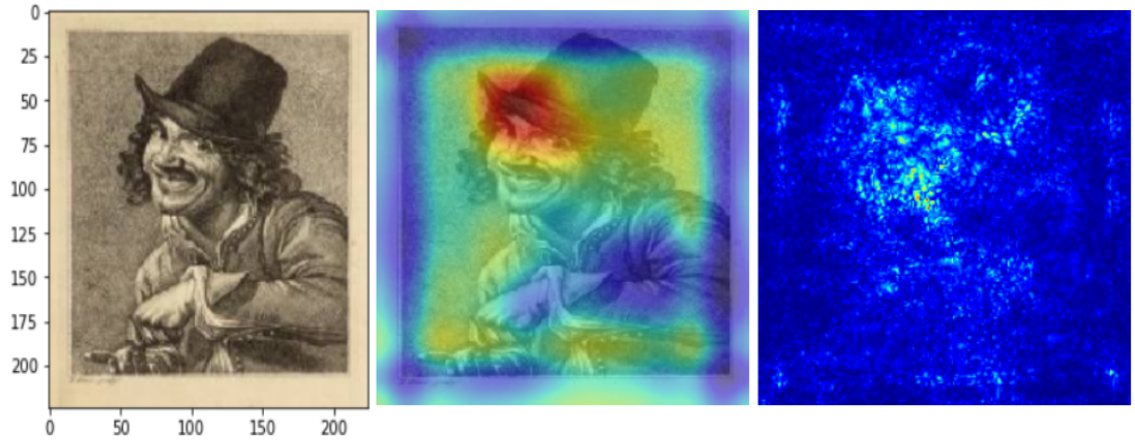


Figure 5.7: A visualisation of 16 of the filters of VGG16, obtained using the Activation Maximisation algorithm.

both seem to validate this finding: the focus is located between the face and the hat. While it is clear why the hat could be recognised as an Object, it is interesting here how the algorithm does not pay any attention to some features that are very distinctively human, such as the hand; but focuses uniquely on the face. It is plausible that the algorithm can successfully extract faces as a feature and associates them with a Person. It is also plausible that the algorithm was able to connect the facial expression of the man in the etching with Emotions, Concepts & Ideas; as, after all, the face does clearly manifest emotions. The results on these visualisations are positive, and it appears that the algorithm is looking at the right spots to make the right conclusions.

5.2.2 The Case of Abstraction

Looking more specifically into the case of Abstract Art: How does the class Abstraction perform compared to the other classes? How well can the algorithm determine




```
predicted as: ['people', 'objects', 'emotions, concepts and ideas']
actual class: ['people', 'objects']
```

Figure 5.8: *On the left*, a etching by Francis Place of a man wearing a hat, with its actual class and its predicted one using off-the-shelf VGG with class weights displayed underneath. *On the right*, the saliency map made through back-propagation from the output layer; *in the middle*, a visualisation of the attention of the final convolutional layer computed using the GradCAM algorithm. As one can see, the attention is correctly placed over the face and the hat, which are successfully recognised by the algorithm.

whether an abstract artwork is only conceptual or whether the intent is representational? In the last case, how well can it recognise the representation?

If we look back at Figure 5.2, in the heat-map on the right, Abstraction shows good performance both in terms of confusion for other classes and with other classes. The algorithm predicts that an artwork is Abstract with precision: 0.43, recall: 0.68, and f1: 0.52, meaning that Abstraction is among the classes that are most successfully recognised. The class is mistaken mainly for People and Nature (for reference, this is the information on Figure 5.2 following the row Abstraction), while History is the only class that is wrongfully predicted as Abstract (following the column Abstraction). Furthermore, in Figure 5.5, Minimalism, a form of Abstract Art, is among the most successfully recognised movements, while Performance Art, whose artworks are also often classified as Abstraction, is the worst performing. Although this highlights a deficiency in the algorithm, the poor performance can be

explained by the intrinsic difficulty of capturing the complexity of Performance Art in one picture.

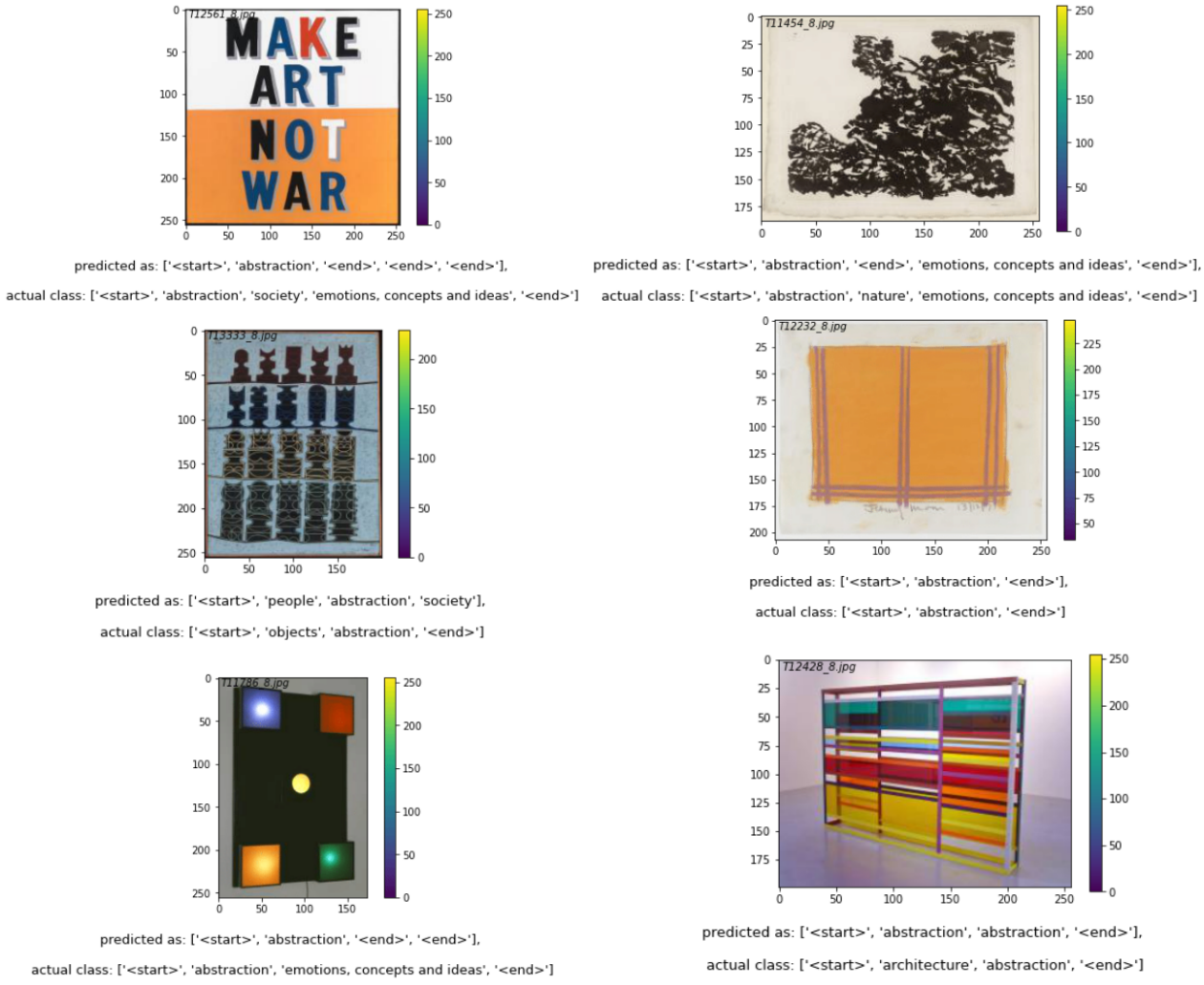


Figure 5.9: A collection of predictions related to abstraction by CNN-RNN. *From left to right, top to bottom:* 1. Bob and Roberta Smith, 'Make Art Not War', 2. Alexander Cozens, 'A High Foreground, That Is to Say, a Large Kind of Object, or More than One. Near the Eye.', 3. Anwar Jalal, 'Shemza Chessmen One', 4. Jeremy Moon, 'Drawing', 5. Stephen Willats, 'Visual Field Automatic No.1', 6. Liam Gillick, 'Returning to an Abandoned Plant'.

Reconnecting to the modifications in style mentioned in the previous section, [9] identifies in Abstraction the culmination of these movements, the 'dissolution of matter and pure composition'. In this sense, Abstraction is presented as the absence of the outer world and the complete dominance of the inner, which was achieved around the years of 1920-50s. Intuitively, this should constitute a challenge for our

algorithm, especially when it comes to recognising the representational subject and the conceptual intent of the artwork.

When looking at Figure 5.9, the artworks 2,3, and 6 are representative of the hardship hypothesised above in individuating the subject of the abstract representation. In fact, the model could connect these artworks to Abstraction, but it could not recognise that People and Nature were depicted. Artwork 1, and 5 demonstrate how arduous it can be for an algorithm to detect the conceptual and symbolic meaning of an artwork. In the first case, the algorithm would have to be able to process text in order to understand the societal impact of the artwork, while in the second, where the artist creates a game in which the spectator is impelled to create order out of the random flashing of the lights ([24]), it is impossible to detect the concept based only on the image.

In sum, the model predicts with good confidence whether the artwork is Abstract and whether it represents Emotions, Concepts & Ideas, but it seems almost aleatory when it comes to predicting the content of the abstract representation and its conceptual meaning.

Chapter 6

Discussion

In the previous section, we have analysed the performance of CNN-RNN and attempted to locate and understand its errors. The results discussed thus far bring us to the following interrogatives:

- What is the practical utility of this model and what are its limitations?
- What can these results tell us about the expressive power and limitations of current Deep Learning (DL) methods?

Regarding the practical utility of the proposed models, and, specifically, of CNN-RNN; the ample limitations in its predictions suggest that the model cannot be used without supervision to generate automatic tags. Nonetheless, a viable option remains to adopt an iterative procedure: We train the model on the entirety of the currently annotated corpus, predict on the newly digitised artworks, appoint a person to correct the mislabels, and periodically feed these newly annotated artworks back into the training. Taking into account that the performance is not going to be reliable *per se* until the size of the training data is increased significantly, this method of 'suggestion and correction' will greatly speed up the annotation process, allowing to produce more metadata in the same time-frames. It is hard to predict how many annotations will be needed before the predictions are reliable and can be put in production. This will be interesting for future investigations.

Currently, the biggest practical limitations concern lack of data and the class imbalance: classes with fewer training samples are less reliably classified, and the samples are not enough for the algorithm to learn to discern the content of very abstract

representations. Moreover, being a hierarchical thesaurus, more work has to be put into classification at other levels, especially with regards to the growing number of classes and decreasing amount of samples per class.

The second point is more subtle. In broad terms, the difference in performance among classes seems fundamental to gain some insight into the power and limitations of this method and, by extension, of DL. As aforementioned, the classes People, Objects, Places, Architecture, Nature, Interiors, Abstraction, and Emotions, Concepts & Ideas are recognised with satisfactory precision; while Work and Occupations, Symbols & Personifications, Religion & Belief, Leisure & Pastimes, History, and Literature & Fiction are not. In hindsight, this indicates that a DL algorithm is capable of discerning the physical content of most depictions, while, in most cases, not the abstract content of the same.

According to Panofsky, who conceived Iconclass[1], there are three layers of meaning in an artwork, which are at the basis of iconography:

1. "primary or natural subject matter, constituting the world of artistic motifs"
2. "secondary or conventional subject matter, constituting the world of images, stories, and allegories"
3. "iconographical interpretation in a deeper sense (Iconographical synthesis)" [29]

When looking at these layers, we can identify the recognition of primary or natural subject matter with the recognition of the physical classes discussed above. The second layer, on the other hand, is not straightforwardly comparable to a task in our classification. A good indication of whether the algorithm acknowledges this second layer could be the correct recognition of some of the abstract classes, such as Symbols & Personifications, but could also be found in the correct classification of the content of an abstract artwork. This step is not yet fully achieved, but it is plausible that with more training data it would be. The last layer requires understanding of the cultural-historical aspect of the depiction. We believe that this will be close to impossible to determine in a DL algorithm, especially as long as such algorithms

[1]Iconclass is the classification system which inspired the Subject Index used in this paper.

remain mostly black boxes. Panofsky himself admits that most artists are not aware of this layer of their artworks.

OBJECT OF INTERPRETATION	ACT OF INTERPRETATION	EQUIPMENT FOR INTERPRETATION	CONTROLLING PRINCIPLE OF INTERPRETATION
I–**Primary** or **natural** subject matter - (A) factual, (B) expressional, constituting the world of artistic motifs.	**Pre-iconographical description** (and pseudo-formal analysis).	**Practical experience** (familiarity with **objects** and **events**).	History of **style** (insight into the manner in which, under varying historical conditions, **objects** and **events** were expressed by **forms**).
II–**Secondary** or **conventional** subject matter, constituting the world of **images, stories** and **allegories.**	**Iconographical analysis** in the narrower sense of the word.	**Knowledge of literary sources** (familiarity with specific **themes** and **concepts**).	History of **types** (insight into the manner in which, under varying historical conditions, specific **themes** or **concepts** were expressed by **objects** and **events**).
III–**Intrinsic meaning** or **content, constituting the world of 'symbolical values'.**	**Iconographical interpretation** in a deeper sense (**Iconographical synthesis**).	Synthetic **intuition** (familiarity with the **essential tendencies of the human mind**), conditioned by personal psychology and **'Weltanschauung'.**	History of **cultural symptoms** or **'symbols'** in general (insight into the manner in which, under varying historical conditions, **essential tendencies of the human mind** were expressed by specific **themes** and **concepts**).

HISTORY OF TRADITION

Figure 6.1: An illustration of Panofsky's the three layers of meaning. Retrieved from [29]

Figure 6.1 is interesting as it helps us determine, based on what was stated above, the current state of DL. As mentioned before, our algorithm was able to discern the first layer. According to this table, such a recognition indicates that the algorithm has sufficient 'familiarity with the objects' (that, in this case, has been learned during the pre-training on ImageNet), and was able to extrapolate, in most circumstances, 'the manner in which [...] objects and events were expressed by forms'. Although the second layer was not fully achieved, some aspects, such as the decent recognition of abstract styles and of conceptual art, are encouraging. For this reason, we believe that the second layer is at an arm's distance from being achieved. This, according to the table, would signify 'familiarity with specific themes and concepts', and understanding of how these are 'expressed into objects and events'. The lack of data was possibly the biggest limitation to the achievement of this second layer. With regards to the third, an intuition into the 'essential tendencies of the human mind' and how these are 'expressed by themes and concepts' seems to be unattainable

with the current techniques of DL, therefore, exhibiting an intrinsic limitation of the state-of-the-art DL methods.

Chapter 7

Conclusions

The aim of this study was to delineate to what extent machines can recognise the content of artworks. This was done by assessing the performance of various multi-label classification models, that use transfer learning in the feature extraction phase, on the Tate Dataset. The best results were achieved by CNN-RNN, which used VGG as pre-trained and modelled label-to-label correlations. The model obtained a per-class accuracy of 0.57, which is comparable to the result obtained by [4], only on classical paintings. The investigations showed that VGG is the best performing feature extractor for content classification for artworks, confirming the conclusion by [3] that VGG, without fine-tuning, generalises best on artworks.

Although the predictions on some classes (such as People, Objects, Places, Architecture and Nature) were satisfactory and manifested understanding of most aesthetics representational tools, the overall performance was compromised by the scarcity of the data (as the filtered dataset contains barely 24'000 artworks) and the significant class imbalance (some classes have more than 10 times as many samples as others). In fact, classes such as Literature & Fiction and History obtained extremely poor results, illustrating that the algorithm was unable to extract and understand the socio-cultural context of the artworks.

An interesting case was the performance of the classes Abstraction and Emotions, Concepts & Ideas. Despite the precision being lower than some physical classes, the number of false negatives for the two classes was very low, and they were often correctly predicted in conjunction, to delineate when the artwork is predominantly

conceptual. This phenomenon could, in our opinion, indicate that the ability of these machines to comprehend themes and concepts is not too far down the road.

Despite this feeling of optimism for the future of the application of these algorithms to artworks, some elements, such as the little improvement that was achieved during training, the poor performance of the majority of abstract classes and the difficulties encountered with Performance Art, inform us that the limitations of such algorithms are still considerable and a lot more work has to be done to achieve the desired confidence to use the algorithm in practice.

As to future research, the main areas that need further elaboration are the implementation of adequate methods to overcome class imbalance, data scarcity, and to classify at different levels of depth in the thesaurus. In particular, the following techniques should be tested: Custom learning rate, oversampling for under-represented classes, data augmentation, and cross-validation to increase the training size. Some experiments should be carried out providing as input to the CNNs not only the image but also related metadata, such as artist, year, and movement, in the form of embedding.

Appendix A

A.1 Hierarchical Classification

A.1.1 Methodology

Subjects that appear in paintings can be defined at different levels of specificity. One can say that an image contains a person, but also a female figure in her 20s. For this reason, exploration has to be carried out on how to classify at different levels, possibly using the information of the previous level. A viable approach to the classification is to classify first on the general level, and subsequently delve into a deeper level, using a hierarchical or cascading architecture. For instance, the algorithm must first determine whether the figure contains a person, then proceed to determine the gender.

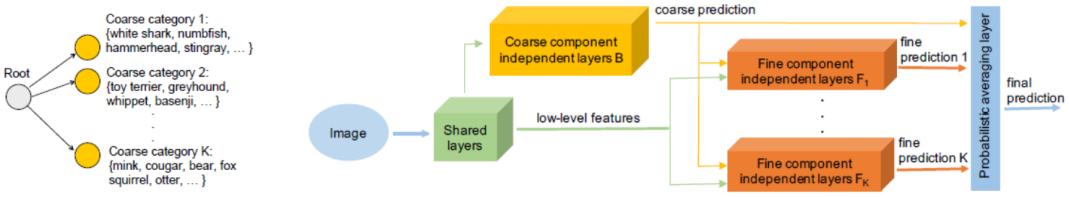


Figure A.1: An illustration of the architecture of HD-CNN. [33].

In terms of the hierarchical classification techniques, this paper tests the performance of HD-CNNs (Hierarchical Deep CNN's, [33]) and of a naive cascading architecture. The two models are compared to a baseline obtained with the same model architec-

ture of Figure 3.1, only substituting the output layer of 15 units with 141, which is the number of classes at this level of specificity.

HD-CNN is an architecture composed of a coarse classifier, single fine classifiers for each coarse prediction, and a probabilistic averaging layer as in Figure A.1 ([33]). The coarse classifier serves as a threshold and as weights for the subsequent fine classification. This architecture has excellent performance but does not allow flexibility when passing from the coarse to the fine layer. In fact, if an error is made in the prediction on the coarse layer, the fine layer will automatically be mis-classified.

For this reason, we implemented a less strict algorithm, which we will call Naive Cascading algorithm. In this model, the prediction of each level of classification is passed to the following level in the form of an embedding. This way, the algorithm is informed about a prediction made on the coarse layer, but still has the opportunity to recover from an error on the subsequent level. The architecture is trained altogether using as loss the sum of the losses of each classification level. A lot more work has to be put into improving this architecture as the multiple levels create problems that affect the gradient flow, which, in turn, makes the training unstable.

A.1.2 Preliminary Results

Table A.1: Results Table at Level 2

Model	**O-A**	**O-F1**	**O-P**	**O-R**
Baseline	**0.96**	**0.07**	**0.06**	0.09
Cascading	0.53	0.06	0.03	**0.56**
Hierarchical	0.90	0.05	0.04	0.07

As one can see in Table A.1, the models at this level are at a very early stage of development and require a lot more work.The above Table A.1 depicts the metrics that were computed on the test set after the second epoch, thus improvement is to be expected in the following epochs, especially concerning the Cascading and Hierarchical algorithms. Unfortunately, due to the excessive memory the models demanded, they could not be run for longer.

In general terms, the Table A.1 shows that the baseline model performs best after

very few iterations, while the Cascading architecture seems to be improving only the O-R, demonstrating an imbalance towards a positive prediction. The poor performance of the two hierarchical models indicates that the knowledge of the prediction on the coarse level is not crucial, at least at the beginning of training.

In the future, optimising both the Hierarchical and Cascading algorithm should be tackled, particularly regarding memory use.

References

[1] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li and L. Fei-Fei, 'Imagenet: A large-scale hierarchical image database.', *2009 IEEE conference on computer vision and pattern recognition*, no. 1, pp. 248–255, 2009 (cit. on pp. 1, 3).

[2] A. Elgammal, B. Liu, D. Kim, M. Elhoseiny and M. Mazzone, 'The shape of art history in the eyes of the machine', vol. 9, no. 1, 2018 (cit. on pp. 7, 8, 31).

[3] M. Fiorucci, M. Khoroshiltseva, M. Pontil, A. Traviglia, A. Del Bue and S. James, 'Machine learning for cultural heritage: A survey.', *Pattern Recognition Letters*, vol. S0167865520300532, no. 1, 2020. [Online]. Available: `https://doi.org/10.1016/j.patrec.2020.02.01` (cit. on pp. 23, 40).

[4] N. Gonthier, Y. Gousseau, S. Ladjal and O. Bonfait, 'Weakly supervised object detection in artworks', *Leal-Taixé & S. Roth (Eds.), Computer Vision – ECCV 2018 Workshops*, vol. 11130, no. 1, pp. 692–709, 2019. [Online]. Available: `https://doi.org/10.1007/978-3-030-11012-3_53` (cit. on pp. 8, 40).

[5] I. Goodfellow, Y. Bengio and A. Courville, 'Deep learning.', *MIT press*, 2016 (cit. on pp. 5, 9, 31).

[6] P. Hall, H. Cai, Q. Wu and T. Corradi, 'Cross-depiction problem: Recognition and synthesis of photographs and artwork', *Computational Visual Media*, vol. 1, no. 2, pp. 91–103, 2015 (cit. on pp. 1, 3).

[7] K. He, X. Zhang, S. Ren and J. Sun, 'Deep residual learning for image recognition.', *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, no. 1, pp. 770–778, 2016. [Online]. Available: `https://doi.org/10.1109/CVPR.2016.90` (cit. on pp. 1, 6, 10, 12).

[8] S. Ioffe and C. Szegedy, 'Batch normalization: Accelerating deep network training by reducing internal covariate shift', *arXiv preprint arXiv:1502.03167*, 2015 (cit. on p. 14).

[9] W. Kandinsky, *Concerning the spiritual in art*. Courier Corporation, 2012 (cit. on pp. 29, 34).

[10] A. Lecoutre, B. Negrevergne and F. Yger, 'Recognizing art style automatically in painting with deep learning.', vol. 16, no. 1, 2017 (cit. on pp. 6, 23).

[11] Y. LeCun *et al.*, 'Lenet-5, convolutional neural networks', *URL: http://yann. lecun. com/exdb/lenet*, vol. 20, p. 5, 2015 (cit. on p. 14).

[12] C.-W. Lee, W. Fang, C.-K. Yeh and Y.-C. Frank Wang, 'Multi-label zero-shot learning with structured knowledge graphs', in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1576–1585 (cit. on p. 13).

[13] V. Nair and G. E. Hinton, 'Rectified linear units improve restricted boltzmann machines', in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 807–814 (cit. on p. 14).

[14] E. Posthumus, 'Brill iconclass ai test set', 2020 (cit. on p. 2).

[15] M. Sabatelli, M. Kestemont, W. Daelemans and P. Geurts, 'Deep transfer learning for art classification problems.', *L. Leal-Taixé & S. Roth (Eds.), Computer Vision – ECCV 2018 Workshops*, vol. 11130, no. 1, pp. 631–646, 2019. [Online]. Available: `https://doi.org/10.1007/978-3-030-11012-3_48` (cit. on p. 6).

[16] B. Saleh, K. Abe, R. S. Arora and A. Elgammal, 'Toward automated discovery of artistic influence.', *Multimedia Tools and Applications*, vol. 75(7), no. 1, pp. 3565–3591, 2016. [Online]. Available: `https://doi.org/10.1007/s11042-014-2193-x` (cit. on p. 7).

[17] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh and D. Batra, 'Grad-cam: Visual explanations from deep networks via gradient-based localization', *International Journal of Computer Vision*, vol. 128, no. 2, pp. 336–359, Oct. 2019, ISSN: 1573-1405. DOI: `10.1007/s11263-019-01228-7`. [Online]. Available: `http://dx.doi.org/10.1007/s11263-019-01228-7` (cit. on p. 31).

[18] L. Shamir, T. Macura, N. Orlov, D. M. Eckley and I. G. Goldberg, 'Impressionism, expressionism, surrealism: Automated recognition of painters and schools of art.', *ACM Transactions on Applied Perception*, vol. 7(2), no. 1, pp. 1–17, 2010. [Online]. Available: `https://doi.org/10.1145/1670671.1670672` (cit. on p. 5).

[19] K. Simonyan and A. Zisserman, 'Very deep convolutional networks for large-scale image recognition.', *ArXiv:1409.1556 [Cs].*, no. 1, 2015. [Online]. Available: `http://arxiv.org/abs/1409.1556` (cit. on pp. 10, 11).

[20] K. Simonyan, A. Vedaldi and A. Zisserman, 'Deep inside convolutional networks: Visualising image classification models and saliency maps', *arXiv preprint arXiv:1312.6034*, 2013 (cit. on p. 31).

[21] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever and R. Salakhutdinov, 'Dropout: A simple way to prevent neural networks from overfitting', *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014 (cit. on p. 14).

[22] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna, 'Rethinking the inception architecture for computer vision', in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826 (cit. on pp. 10, 11).

[23] Tate, *Archives & access project: (subject) index to the soul: Opening up tate's archives: By darragh o'donoghue – behind the scenes*. [Online]. Available: `http://www.tate.org.uk/about-us/projects/transforming-tate-britain-archives-access/archives-access-project-subject-index` (cit. on p. 2).

[24] ——, *Archives & access project: (subject) index to the soul: Opening up tate's archives: By darragh o'donoghue – behind the scenes*. [Online]. Available: `http://www.tate.org.uk/about-us/projects/transforming-tate-britain-archives-access/archives-access-project-subject-index` (cit. on pp. 2, 17, 35).

[25] Tategallery, *Tategallery/collection*, Feb. 2018. [Online]. Available: `https://github.com/tategallery/collection` (cit. on pp. iii, 2, 17).

[26] V. Terraroli, *Lezioni di storia dell'arte: Il Mediterraneo dall'antichità alla fine del Medioevo*. Skira, 2001, vol. 1 (cit. on p. 29).

[27] P. Tzouveli, N. Simou, G. Stamou and S. Kollias, 'Semantic classification of byzantine icons', *IEEE Intelligent systems*, no. 2, pp. 35–43, 2009 (cit. on p. 8).

[28] N. Van Noord, E. Hendriks and E. Postma, 'Toward discovery of the artist's style: Learning to recognize artists by their artworks', *IEEE Signal Processing Magazine*, vol. 32, no. 4, pp. 46–54, 2015 (cit. on p. 6).

[29] R. Van Straten, 'Panofsky and iconclass', *Artibus et historiae*, pp. 165–181, 1986 (cit. on pp. 2, 37, 38).

[30] J. Wang, Y. Yang, J. Mao, Z. Huang, C. Huang and W. Xu, 'Cnn-rnn: A unified framework for multi-label image classification.', *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, no. 1, pp. 2285–2294, 2016. [Online]. Available: `https://doi.org/10.1109/CVPR.2016.251` (cit. on pp. 13, 15).

[31] Y. Wei, W. Xia, M. Lin, J. Huang, B. Ni, J. Dong, Y. Zhao and S. Yan, 'Hcp: A flexible cnn framework for multi-label image classification', *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 9, pp. 1901–1907, 2015 (cit. on p. 13).

[32] N. Westlake, H. Cai and P. Hall, 'Detecting people in artwork with cnns', in *European Conference on Computer Vision*, Springer, 2016, pp. 825–841 (cit. on p. 1).

[33] Z. Yan, H. Zhang, R. Piramuthu, V. Jagadeesh, D. DeCoste, W. Di and Y. Yu, 'Hd-cnn: Hierarchical deep convolutional neural networks for large scale visual recognition.', *2015 IEEE International Conference on Computer Vision*

(ICCV), no. 1, pp. 2740–2748, 2015. [Online]. Available: https://doi.org/10.1109/ICCV.2015.314 (cit. on pp. 42, 43).

[34] J. Zujovic, L. Gandy, S. Friedman, B. Pardo and T. N. Pappas, 'Classifying paintings by artistic genre: An analysis of features & classifiers', in *2009 IEEE International Workshop on Multimedia Signal Processing*, IEEE, 2009, pp. 1–5 (cit. on p. 5).