**Artist Recognition for Fine Art Paintings with Deep Learning**

CS688: Pattern Recognition (Fall 2017)

Bhavika Tekwani (btekwani), Jaecheon Jeong, Melanie Baker

**Abstract**

**Dataset**

We used the Wikiart dataset released by the authors of the ICIP2016 paper on deep convolutional networks for fine-art paintings classification [5]. The dataset in its entirety is 27 GB and contains over x images spanning y artists and 27 genres.For our artist classification task, we use only the Impressionism genre (13060 images). Furthermore, to quickly prototype several methods and iterate on them, we limit all our experiments to 15 artists chosen in no particular order. For each artist, we select a set of 40 images. Therefore, our dataset for this task contains 600 images.

The complete list of artists is as follows:

|  |  |
| --- | --- |
| **Artist** | **Class Label** |
| adam-baltatu | 0 |
| alfred-sisley | 1 |
| antoine-blanchard | 2 |
| arkhip-kuindzhi | 3 |
| armand-guillaumin | 4 |
| auguste-rodin | 5 |
| berthe-morisot | 6 |
| camille-pissarro | 7 |
| childe-hassam | 8 |
| claude-monet | 9 |
| Constantin-artachino | 10 |
| cornelis-vreedenburgh | 11 |
| edgar-degas | 12 |
| edouard-manet | 13 |
| eugene-boudin | 14 |

**Setup**

Hardware:

We did not use any specialized hardware for running our experiments. Available memory ranged from 4 GB - 8 GB on i5 and i7 processors. Although we used a deep learning approach, we did not use GPUs to train our convolutional neural networks.

Software:

We used Python 3 for all our experiments. Some libraries we used for machine learning & deep learning include *pandas*, *scikit-learn, numpy, OpenCV, Tensorflow, Keras & PyTorch*.

**Literature Review**

A literature review of several gave us insight into what features and methods have worked best for similar tasks. Those papers are below with our general findings.

A. Blessing and K. Wen’s paper was written in 2010, which explains why a deep learning network was not part of their architecture. The author’s initially implemented their solution using Naive Bayes to create their features, but found it wasn't sufficient enough and they went to an SVM. The main thing we got out of this was different features (ex. SIFT, HOG, color histograms, etc.) we could focus our algorithm on if we didn't implement our solution as a CNN [1].

In B. Saleh and A. Elgammal’s paper, the authors’ goal was to develop an algorithm that was “able to make aesthetic-related semantic-level judgments, such as predicting a painting’s style, genre,and artist, as well as providing similarity measures optimized based on the knowledge available in the domain of art historical interpretation”. The main takeaway, since our main goal is to identify paintings by artist, is that for Artist Classification, their CNN that learned using the Information-Theoretic Metric Learning (ITML) trained the fastest and had the best accuracy [2].

Y. Hong and J. Kim’s paper was published in 2017 and is very similar in our original approach to the artist classification problem. It talks about each of the different CNN architectures they came up with and their results. With CNNs doing so well in image recognition, it's clear this is the more modern approach and we think if we each developed a CNN we could improve our success on this task [3].

A. Elgammal’s paper shows another option for building a learning architecture if we didn't want to use python and focus more on classifying the more general painting style. This person was able to use the MATLAB image processing libraries and create an SVM that got up to 60% classification on painting style. It's worse than a CNN, but if we had time we could do an experiment [4].

After a review of the above, we conclude that our approach would be unique in how we architect our CNN and the features we could use for the SVMs & Random Forests. We also want to try different learning rates for the deep nets and see the effect on classification accuracy per class.

**Approach**

1. Preprocessing
2. Feature Engineering
3. Experiments

**Results**

**Conclusions**

**References**

[1] A. Blessing and K. Wen, Using machine learning for identification of art paintings,

Tech. Rep., 2010.

[2] B. Saleh and A. Elgammal, Large-scale classification of fine-art paintings: Learning

the right metric on the right feature, ArXiv Prepr. ArXiv150500855, 2015.

[3] Y. Hong and J. Kim, Art Painting Identification using Convolutional Neural

Network, Int. J. Appl. Eng. Res., vol. 12, no. 4, pp. 532–539, 2017.

[4] Creating Computer Vision and Machine Learning Algorithms That Can Analyze

Works of Art - MATLAB & Simulink. [Online]. Available:

<https://www.mathworks.com/company/newsletters/articles/creating-computer-vision-and-machine-learning-algorithms-that-can-analyze-works-of-art.html>. [Accessed: 18-Sep-2017].

[5] ICIP2016, Wei Ren Tan, Chee Seng Chan, Hernan E. Aguirre and Kiyoshi Tanaka, Ceci n'est pas une pipe: {A} deep convolutional network for fine-art paintings classification. <http://web.fsktm.um.edu.my/~cschan/doc/ICIP2016.pdf>