Art Style Recognition with Self-Trained Ensemble of AutoEncoding Transformations

Akshay Joshi
2581346
s8akjosh@stud.uni-saarland.de

Ankit Agrawal
2581532
s8anagra@stud.uni-saarland.de

Sushmita Nair
2581308
s8sunair@stud.uni-saarland.de

Abstract— **The artistic style of a painting is a rich descriptor that catches both visual and chronicled data about artistic creation. Accurately distinguishing the artistic style of paintings is significant for indexing large artistic databases. Artistic feature classification has received little to no attention in computer vision research. In this paper, we investigate the use of self-supervised learning methods to solve the problem of recognizing complex artistic styles and outperforming existing state-of-the-art approaches by achieving an overall accuracy gain of 20% on a highly class imbalanced WikiArt dataset consisting of 27 classes.**

I. INTRODUCTION

Art style recognition and classification are gaining recent interest because of the large amount of visual artistic data made available by art museums that have been digitalized and indexed into three categories according to their artist, style, and genre. Style classification is easily perceived by human but the properties like sentiment and emotion are difficult to be modeled in a computational way.

In this paper, we deal with the problem of classifying paintings by its style (e.g. Impressionism, Baroque, etc.). This task is quite challenging as the dataset is highly class imbalanced and also due to a large amount of both inter and intra-class variations i.e. there are different personal styles present in the same art style as well as similar style in many different classes. To understand the idea of these challenges, some examples taken from the WikiArt dataset are reported in Figure 1.

The rest of the paper is organized as follows: In Section II, we look into related works that have been done in this domain. We discuss about how the data is acquired including some of the augmentations that have been used in Section III. Further, in Section IV we describe our approach followed by various baselines we used to assess the performance of our approach in Section V. In Section VI, we present experimental results and compare them against the baselines. Finally, we conclude with inferences and future directions.

II. RELATED WORK

A lot of research has been done in the domain of paintings categorization mainly for styles, genres, and artists. In this section, we will discuss some work done in the field of art style recognition and classification.

Several publications have addressed this problem using pre-computed features such as color histograms, special organisation and lines descriptions ([1], [2]), or directly training features from the images itself ([3], [4]). Good results

Fig. 1. Four examples of images of the WikiArt datasets. The top two images represent the problem of variation found within a style and the bottom two represents similarity between different styles.

were achieved using pre-computed features using multi-task learning and dictionary learn by ([2], [4]), addressed the problem using a variation of the same neural network and managed to achieve better results with a fully automatic procedure. More recently, [5] suggested that the ResNet50 is the best performing CNN model with an accuracy of 62% on WikiArt data. The results obtained in this study show that self-supervised learning methods outperform other CNN methods used with even less computation and data required compared to other methods.

III. DATA ACQUISITION

To train our models, we use the Wikipaintings dataset, a large image dataset collected from WikiArt which is made publicly available following the experimental protocol used by Tan et al., 2016. The resulting dataset contains more than 80,000 fine-art paintings for a total of 27 styles: Abstract Art, Abstract Expressionism, Art Informel, Art Nouveau (Modern), Baroque, Color Field Painting, Cubism, Early Renaissance, Expressionism, High Renaissance, Impressionism, Magic Realism, Mannerism (Late Renaissance), Minimalism, Naive Art (Primitivism), Neoclassicism, Northern

Renaissance, Pop Art, PostImpressionism, Realism, Rococo, Romanticism, Surrealism, Symbolism, and Ukiyo-e.

Out of a total of 81,446 images, for the ResNet[6] baseline setup, we took 40,000 images for training, 20,168 images for validation and 8,616 images as test set. Similarly, for EnAET[7] setup, we used only a subset of data having a total of 30,183 images, among which 58% is used to training i.e. 17,567 images, 4000 images for validation, and 29% for test i.e 8,616 images. We've used the same test set for all models to obtain comparable results.

The dataset here is high class imbalanced which can be seen from the class-wise image distribution in Figure 2. To handle the problem of class imbalance, different data augmentation techniques have been used such as spacial transformations (Projective, affine, similarity, and Euclidean transformations) and non-spatial transformations (color, contrast, brightness, and sharpening transformations). The usages of these augmentations are explained clearly in the next section.

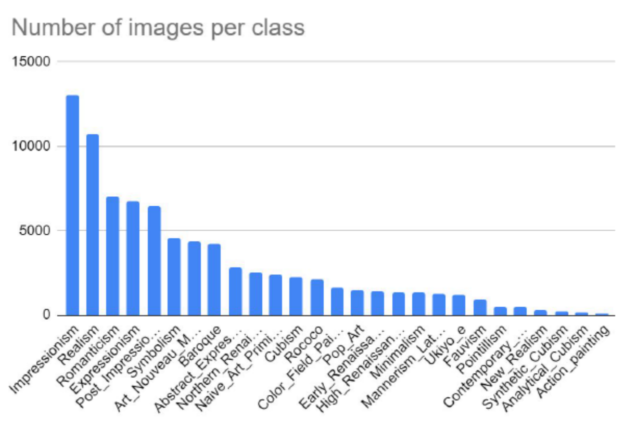

Fig. 2. Distributions of number of images available for each of the 27 styles within the WikiArt dataset.

IV. METHODOLOGY

Self-supervised representations of the images play a crucial role in exploring the data variations across various transformations. In Supervised approaches, to handle class imbalance problem in various domains such as Medical Diagnosis, Disease Prediction and Art Style recognition, vanilla data augmentation techniques are applied to labeled data. Unlike these approaches, we can instead weakly train a semi-supervised model without any major reliance on labeled data.

In this approach published by the authors at Maple Research[7], employ unsupervised/weakly supervised data augmentation techniques to explore various spatial and non-spatial transformations and their effects on the unlabeled data. On verifying the model performance across different benchmarks it is evident that self-supervised representations learned from an ensemble of transformations can enact a crucial role in significantly enhancing semi-supervised models.

The framework to recognize rare and exotic art styles is summarized as follows:

- To train a semi-supervised model, an ensemble of both spatial and non-spatial transformations from both labeled and unlabeled data are used in a self-supervised setting.
- These set of AutoEncoding transformations are used as a regularization network by learning robust features across different image transformations and further improve the consistency of label predictions for transformed images $t_k(x)$ by minimizing their KL divergence with the original images x.

A. Model

In the Semi-supervised learning paradigm, instead of pre-training the model, the proposed method devises network of AETs as a regularizer which compliments the Semi-supervised learning loss to train classifiers.

The approach further enforces two aspects to enrich the final semi-supervised label predictions by accomplishing the following criteria:

- Consistent Predictions: To maximize the prediction consistency even when the classification boundary to confidently predict labels is overlapping. The Mean Teacher model is used where weights of a teacher model is updated with an exponential moving average of the weights from all the student models.

$$\Theta'_\tau = \alpha\Theta'_{\tau-1} + (1-\alpha)\Theta_\tau$$

- Confident Predictions: To further improve the boundary between classes to achieve highly confident label predictions MixUp[7] is employed to train a model with the linear combination of the inputs and their corresponding outputs.

Due to different applied transformations and augmentations, there could be a difference between the features extracted from original and transformed images. Thus, the transformation decoders D_k could recover the corresponding transformations as long as the encoded features from the encoders capture the intricate details of fine visual structures in artistic images. The AutoEncoding Transformations can self-train a good feature representation which could be used along with the encoded features by an efficient semi-supervised classifier to explore an ensemble of spatial and non-spatial transformations.

B. Framework

For each and every image x, five different transformations are applied, namely: Projective, Affine, Similarity, Euclidean a combination of Color + Contrast + Brightness + Sharpness.

As shown in Figure 3, the model is split into three parts: an Encoder E, Classifier C, and a collection of Decoders D_k for different transformations. The original input image x and all of its corresponding transformations $t_k(x)$ are loaded to

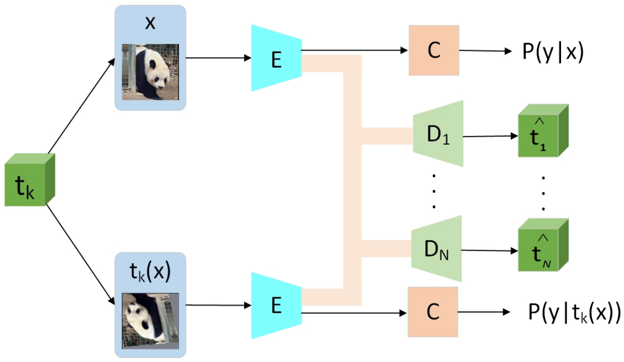

Fig. 3. Confusion matrix plot of ResNet50. Y-axis specifies the true labels and x-axis specifies predicted labels.

respective encoders. Both the Encoders and Classifiers are arranged in a Siamese configuration (shared model weights).

The framework uses Wide ResNet-28-2 network which consists of four blocks. Among which the last block is used as Classifier C, while the Encoder E constitutes the other three blocks of the network. Similarly, all the Decoders D_k use the same network architecture as that of the Classifier. But, the decoders are not configured in a Siamese configuration. The encoded representations of both original and transformed images are concatenated together and are fed to the decoders D_k. Each decoder predicts the parameters of corresponding transformation t_k. Finally, the classifiers C use the encoded representations from the encoder and generates appropriate label predictions for original and transformed images.

C. Loss Functions

- Model Loss: The idea is to minimize a linear combination of SSL and AET loss to train a classifier over the network weights theta.

$$\min_{\Theta} \mathcal{L}_{SSL} + \sum_{k=1}^{N} \lambda_k \mathcal{L}_{AET_k}$$

- AutoEncoding Transformation Loss: Compute the Mean-Squared Error between the predicted transformation and the sampled transformation.

$$\mathcal{L}_{AET_k} = \mathbb{E}_{x,t_k} \|D\left[E(x), E(t_k(x))\right] - t_k\|^2$$

- Kullback-Leibler Divergence Loss: To make persistent predictions across different transformations, the approach is to minimize KL divergence between the predicted label on an original image x and predicted label on a transformed image t(x).

$$\mathcal{L}_{KL} = \mathbb{E}_{x,t} \sum_{y} P(y|x) \log \frac{P(y|x)}{P_t(y|x)}$$

- Semi-supervised Loss: Any loss function which could be used to train a semi-supervised classifier can be utilized. But, in this framework, MixMatch[7] Loss is employed.

$$\begin{cases} \mathcal{L}_{\mathcal{X}'} = \mathbb{E}_{(x,y)\in\mathcal{X}'} H(y, f(x,\Theta)) \\ \mathcal{L}_{\mathcal{U}'} = \mathbb{E}_{(u,q)\in\mathcal{U}'} \|f(u,\Theta) - q\|^2 \\ \mathcal{L}_{mix} = \mathcal{L}_{\mathcal{X}'} + \lambda_{\mathcal{U}'} \mathcal{L}_{\mathcal{U}'} \end{cases}$$

D. Training Hyperparameters

Exhaustive Grid Search is performed to obtain optimal values for the following hyperparameters: Epochs, LR for ADAM optimizer (Backbone network), LR1 for SGD optimizer (AET regularizer network), KL Lambda value to maintain consistency, KLambda are the warm factor values for different transformations in AET network, MaxLambda are the hyperparameters for different transformations in AET network, Data Portion, Batch Size and Beta to maintain consistency in MixMatch.

The final hyperparameters that were utilized are 100 training epochs, LR of 0.002, LR1 of 0.1, Batch Size of 128 images, KL Lambda = 1.0, KLambda = (10, 7.5, 5, 2, 0.5), MaxLambda = (1, 0.75, 0.5, 0.2, 0.05), Data Portion = 1.0 and Beta = 75. The restriction for using a larger batch-size was the amount of volatile memory available in a single P100 GPU.

V. BASELINES

In order to evaluate the results of the trained model, we implemented two baselines - ResNet50, and ResNet50 with data augmentation.

A. ResNet50

The ResNet50 architecture contains a convolution layer with 64 7*7 kernels with stride 2 followed by max pooling with stride 2. Then there are 4 groups of blocks such that each block is a 3-layer bottleneck block. At the end, there is a fully connected layer with output neurons corresponding to the number of classes. Rectified Linear Unit (ReLU) is used as the activation function for all weight layers, except for the last layer that uses softmax regression. This forms a total of 50 convolutional layers.

Of the 65,168 WikiArt training images available to us, we use 45,000 images as train set and 20,168 images as validation set to train the model. Input images are first resized to 224x224 to simplify computation and preserve overall image structure. We then zero center the images and normalize them. Though our WikiArt dataset contains 65,168 training images, half of the classes contain less than 1,300 images per class. Hence we chose to use a model pre-trained for object recognition on ImageNet. During the training phase, we fine-tuned the entire model on our WikiArt dataset to obtain task specific features.

B. ResNet50 with Data Augmentation

To compare the performance of EnAET which leverages a wide range of image transformations we retain the same network and training setup as above but augment the dataset with the following transformations - random horizontal flip, rotation, translation, scaling and color-jitter.

Horizontal image flip happens with a probability of 0.5, rotation is randomly drawn between 0° to 90°, horizontal and vertical translations are drawn between 0 and image width and between 0 and image height respectively, scaling factor is drawn between 1 to 2. Color-jitter consists of randomly modifying contrast, brightness and saturation of the input image independently.

VI. Experiments and Results

In this section we aim to answer two scientific questions - ***Q1:*** How well does EnAET perform on our dataset compared to the baselines and why? ***Q2:*** How much training data does EnAET require compared to the baselines?

A. Performance Comparison and Analysis

To answer the first question we look at the overall classification accuracy of the three models on the test set in Table I. The three models were evaluated on a test set consisting of 8616 images. We can see that the classification accuracy of the baseline ResNet50 model is the least at 50.1% and with data augmentation the accuracy is increased slightly to 55%. On the other hand, EnAET performs significantly better with an accuracy of 82.61%. Observing the test accuracy curve for different epochs in Figure 4, we can see that EnAET starts to perform better than the baseline ResNet50 after epoch 40. It achieves the best accuracy scores after epoch 90.

To understand why EnAET performs so well in style classification, let us look at some drawbacks of the ResNet50 model performance. The confusion matrix of ResNet50 test predictions is presented in Figure 5. On examining the misclassifications in the vertical direction, it can be observed that images from many classes are often misclassified as the following five classes - Impressionism, Realism, Expressionism, Post Impressionism and Romanticism. Unsurprisingly, these are the most populated classes i.e, classes with the most number of training images. This means that precision of these highly populated classes are low as shown in Table II. On the other hand, on examining the least populated classes in the horizontal direction we can see that 70% of Action painting images are wrongly predicted as Abstract expressionism. 60% of Analytical cubism and Synthetic cubism images are falsely predicted as its bigger cousin Cubism. 30% of New Realism images are misclassified as Impressionism and 20% of them are misclassified as Realism. 30% of Contemporary Realism images are misclassified as Realism. This means that recall of these less populated classes are low as shown in Table II. These classes have less than 500 images each in the train set. This shows that our imbalanced dataset is degrading the performance of the baseline ResNet. Even on augmenting our dataset, the accuracy only slightly improves but this problem persists.

TABLE I
Comparison of test accuracies of EnAET with baselines

ResNet50	**ResNet50 with data augmentation**	**EnAET**
50.1%	55%	82.61%

Contrary to different Supervised methods which utilize various image augmentation strategies, EnAET overcomes the class imbalance problem by weakly training a semi-supervised model using meager amounts of labelled data and also employs unsupervised data augmentation techniques to explore various spatial and non-spatial transformations and their explicit effects on the unlabeled data. Further, crucial visual features in highly ambiguous classes are mastered by employing ensemble of AutoEncoding transformations as model regularizers.

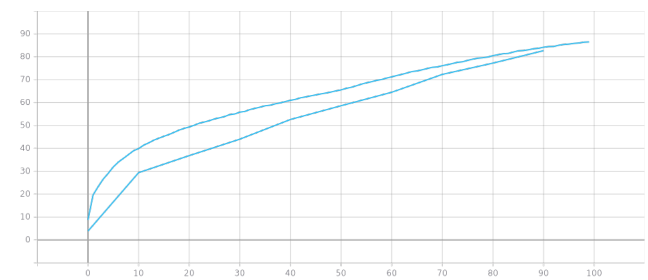

Fig. 4. Test accuracies of EnAET from ephoch 1 to 100

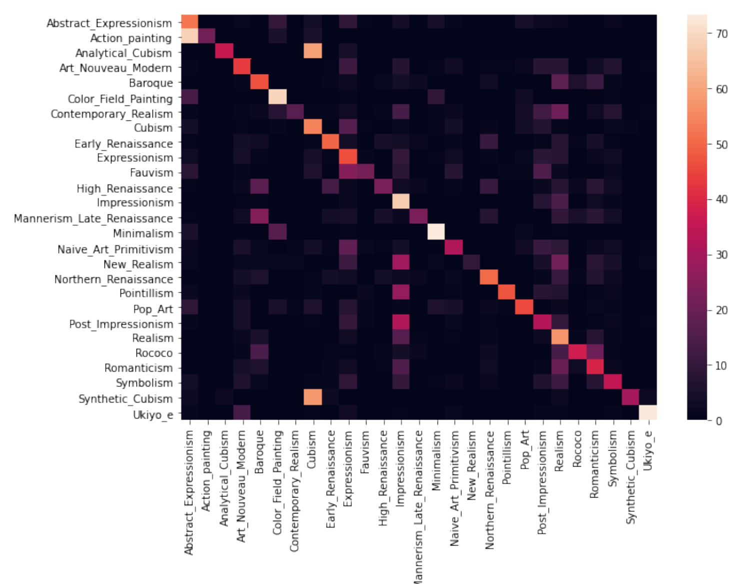

Fig. 5. Confusion matrix plot of ResNet50. Y-axis specifies the true labels and x-axis specifies predicted labels.

B. Experiment with data portions

To answer the second question of how much data the EnAET model requires to produce good results, we first note

TABLE II

PER-CLASS PRECISION AND RECALL FOR THE MOST POPULATED CLASSES AND LEAST POPULATED CLASSES.

Style	Precision	Recall
Impressionism	34%	68%
Realism	28%	57%
Expressionism	30%	46%
Post Impressionism	34%	33%
Romanticism	34%	40%
Action Painting	66%	21%
Analytical Cubism	73%	36%
Synthetic Cubism	62%	30%
New Realism	86%	10%
Contemporary Realism	47%	17%

TABLE III

TEST ACCURACIES OF EnAET WITH DIFFERENT TRAINING DATA PORTIONS COMPARED WITH BASELINES.

Model	Test Accuracy	% of training data used
EnAET	45.63%	8%
EnAET	82.61%	33%
ResNet50	50.1%	100%
ResNet50 with data augmentation	55%	100%

that the EnAET model is trained on 21,567 images which is only 33% of the data used for the baseline ResNets.

We then experimented by using 5,391 training images which is only 8% of the data used for training the baselines. In Table III, the model trained on this train set is compared with other models. We can see that with just about 8% of total training images used for the baselines, EnAET's performance is comparable to the baseline ResNet50 though it is not better. Therefore, EnAET requires very little training data i.e. between 10-33% of the data required for ResNet50.

VII. CONCLUSION

We clearly observe that style classification performance of the self-supervised EnAET model is better than the baseline methods. Our approach successfully overcomes the problems arising with highly imbalanced classes. It also requires only a fraction (10 - 33%) of the training data required for the baseline models.

As future work, deeper encoder networks could be used to better represent image data into feature vectors. Additionally, distributed batch norm can be utilized to fasten the long model training routines. As alternative approaches we could explore contrastive learning methods which could be useful for high resolution image datasets similar to WikiArt and one-shot/few-shot approaches since we have very few images in several classes.

REFERENCES

[1] Florea, Corneliu, Razvan George Condorovici, Constantin Vertan, Raluca Butnaru, Laura Florea and Ruxandra Vrânceanu. “Pandora: Description of a painting database for art movement recognition with baselines and perspectives.” 2016 24th European Signal Processing Conference (EUSIPCO) (2016): 918-922.

[2] Gaowen Liu, Yan Yan, Elisa Ricci, Yi Yang, Yahong Han, Stefan Winkler, and Nicu Sebe. 2015. "Inferring painting style with multi-task dictionary learning." In Proceedings of the 24th International Conference on Artificial Intelligence (IJCAI'15). AAAI Press, 2162–2168.

[3] Saleh, Babak Elgammal, Ahmed. (2015). "Large-scale Classification of Fine-Art Paintings: Learning The Right Metric on The Right Feature." 10.11588/dah.2016.2.23376.

[4] Tan, Wei Ren Chan, Chee Seng Aguirre, Hernan Tanaka, Kiyoshi. (2016). Ceci n'est pas une pipe: "A Deep Convolutional Network for Fine-art Paintings Classification." 10.1109/ICIP.2016.7533051.

[5] Lecoutre, A., Négrevergne, B., Yger, F. (2017). Recognizing Art Style Automatically in Painting with Deep Learning. ACML.

[6] He, Kaiming Zhang, Xiangyu Ren, Shaoqing Sun, Jian. (2016). Deep Residual Learning for Image Recognition. 770-778. 10.1109/CVPR.2016.90.

[7] Wang, Xiao Kihara, Daisuke Luo, Jiebo Qi, Guo-Jun. (2019). EnAET: Self-Trained Ensemble AutoEncoding Transformations for Semi-Supervised Learning.