technologies

Article

Artwork Style Recognition Using Vision Transformers and MLP Mixer

Lazaros Alexios Iliadis [1], Spyridon Nikolaidis [1], Panagiotis Sarigiannidis [2] and Shaohua Wan [3] and Sotirios K. Goudos [1,*]

[1] ELEDIA@AUTH, School of Physics, Aristotle University of Thessaloniki, 541 24 Thessaloniki, Greece; liliadis@physics.auth.gr (L.A.I.); snikolaid@physics.auth.gr (S.N.)
[2] Department of Informatics and Telecommunications Engineering, University of Western Macedonia, 501 00 Kozani, Greece; psarigiannidis@uowm.gr
[3] School of Information and Safety Engineering, Zhongnan University of Economics and Law, Wuhan 430073, China; shwanhust@zuel.edu.cn
* Correspondence: sgoudo@physics.auth.gr

Abstract: Through the extensive study of transformers, attention mechanisms have emerged as potentially more powerful than sequential recurrent processing and convolution. In this realm, Vision Transformers have gained much research interest, since their architecture changes the dominant paradigm in Computer Vision. An interesting and difficult task in this field is the classification of artwork styles, since the artistic style of a painting is a descriptor that captures rich information about the painting. In this paper, two different Deep Learning architectures—Vision Transformer and MLP Mixer (Multi-layer Perceptron Mixer)—are trained from scratch in the task of artwork style recognition, achieving over 39% prediction accuracy for 21 style classes on the WikiArt paintings dataset. In addition, a comparative study between the most common optimizers was conducted obtaining useful information for future studies.

Keywords: vision transformers; computer vision; deep learning; artistic style recognition

check for updates

Citation: Iliadis, L.A.; Nikolaidis, S.; Sarigiannidis, P.; Wan, S.; Goudos, S.K. Artwork Style Recognition Using Vision Transformers and MLP Mixer. *Technologies* **2022**, *10*, 2. https://doi.org/10.3390/technologies10010002

Academic Editor: Pietro Zanuttigh

Received: 18 November 2021
Accepted: 23 December 2021
Published: 28 December 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Deep Learning (DL) has become the dominant paradigm in the field of Computer Vision (CV). ImageNet Challenge (ILSVRC) has been proven to be a driving force for many novel neural network architectures, which have prevailed in the CV research community [1]. However, there are many CV tasks that include the classification of more abstract visual forms, like clouds in the sky [2], abstract images [3] and paintings [4].

The artistic style (or artistic movement) of a painting is a descriptor that contains valuable data about the painting itself, providing, at the same time, a framework of reference for further analysis. In this context, artistic style recognition is an important task in CV taking into consideration that authentic artwork carries a high value (aesthetic, historic and economic) [4]. Artwork style recognition, artist classification and other CV tasks related to paintings had been studied before the "DL revolution" [5–7].

A great deal of research work has been done in this field leading to many impressive results [4–12]. Although many techniques have been deployed, the architectures based on Convolutional Neural Networks (CNNs) have prevailed. Nevertheless, the current technology appears to have reached a plateau in model performance, highlighting the need for new designs. In recent months, there have been a plethora of new DL-based CV models that perform really well on many tasks. However, these models have not been tested, to the best of our knowledge, on the artwork style recognition task.

Transformers have become the dominant architecture in use in the field of Natural Language Processing (NLP), outperforming previous models in various tasks [13]. Transformers are based on the attention mechanism. Attention allows to derive information from any state of a given text sequence. Introducing the attention layer, it is possible to access all previous states and weigh them according to a learned measure of relevancy to the current token, providing sharper information about far-away relevant tokens [13]. Transformers have proven that the recurrence is unnecessary.

In addition, modified architectures have been successfully applied on object detection tasks [14]. Until recently in the task of image recognition, attention has been complementary to convolutions. Extending the idea of attention only mechanisms to CV, a few modifications to the basic transformer architecture are required [15].

Another recent proposal in DL-based CV is MLP Mixer [16]. MLP Mixer is based solely on multi-layer perceptrons (MLPs) and does not make use of either convolutions or attention mechanisms. MLP Mixer may be proven to be a valid alternative to many CV tasks, where there are not so many training data or where the available hardware does not support more expensive (in computational terms) architectures.

Following [17], we present the motivation for this work, along with the letter contributions and the organization of the rest of this work.

1.1. Motivation

The guiding motivation for this research is twofold. On the one hand, there is a need to test the newly proposed DL models on more complicated CV tasks and to demonstrate their applicability. On the other hand, artwork style recognition is a complex problem that needs to be studied further from the research community, since it poses interesting questions about aesthetics, artistic movements, the connection between different styles etc.

1.2. Contribution

The main contributions of this work are as follows

- We propose Vision Transformers as the main ML method to classify artistic style.
- We train Vision Transformers from scratch in the task of artwork style recognition, achieving over 39% prediction accuracy for 21 style classes on the WikiArt paintings dataset.
- We conduct a comparative study between the most common optimizers obtaining useful information for future studies.
- We compare the results compared with MLP Mixer's performance on the same task, examining in this way two very different DL architectures on a complex pattern recognition framework.

To the best of the authors knowledge, this is the first time that Vision Transformers have been applied to the specific problem. The results obtained in this work provide a minimum benchmark for future studies regarding the application of ViT and MLP Mixer in the artwork style recognition task and possibly to other CV tasks, which may include a diverse set of training images.

1.3. Organization of the Paper

The rest of the paper is organized as follows: Section 2 briefly describes related work. In Section 3 Transformers, Vision transformers, MLP Mixer and the basic information about DL optimizers are discussed and details about WikiArt paintings dataset are provided. We elaborate and present the numerical results in Section 4. Finally, Section 5 concludes this work.

2. Related Work

ML and DL techniques have been successfully deployed in the task of Artistic Style recognition. In [4], researchers conducted a comprehensive study of CNNs applied to the task of style classification of paintings and analyzed the learned representation through correlation analysis with concepts derived from art history. In [8–12], many DL and Image Processing techniques are deployed in order to improve accuracy. The advantages and disadvantages of these methods are presented in Table 1 following the presentation in [18].

Another field of study is the Image Style Transfer. Gatys et al. [19], by separating and recombining the image content and style of images, managed to produce new images that combine the content of an arbitrary photograph with the appearance of numerous well-known artworks. Many modifications and optimizations have been proposed since [20,21]. However, in style transfer, it is necessary to separate style from content as much as possible, whereas, in artistic style recognition, the description of the content is used as an additional feature [8].

An active research area is the use of a Generative Adversarial Network (GAN) for conditional image synthesis (ArtGAN) [22,23]. The proposed model is capable of creating realistic artwork as well as generate compelling real world images.

Vision Transformers have gained much research interest. The first model based solely on attention is ViT [15], while [16] introduces MLP Mixer. To the best of our knowledge, this is the first time that ViT and MLP Mixer are implemented on the task of artistic style classification.

Table 1. Artwork style recognition based on DL methods.

Paper	Advantages	Disadvantages
Elgammal A., et al. [4]	• Study of many CNN architectures • Interpretation and representation	No comparison with previous works
Lecoutre A., et. al. [8]	• Comprehensive methodology • Plenty techniques used	Full analysis is provided only for Alexnet
Bar Y., et. al. [21]	• Combination of low level descriptors and CNNs	test only one CNN architecture
Cetinic E., et. al. [10]	• Fine-tuning • Analysing image similarity	No interpretation
Huang X., et. al. [11]	• Two channels used; the RGB channel and the brush stroke information	No interpretation
Sandoval C., et al. [12]	• Novel two stage approach	Only pre-trained models

3. Materials and Methods

3.1. Vision Transformers

Vision Transformer (ViT) was proposed as an alternative to convolutions in deep neural networks. The model was pre-trained on a large dataset of images collected by Google and later fine-tuned to downstream recognition benchmarks. A large dataset is necessary in order to achieve state of the art results.

The main architecture of the model is depicted in Figures 1 and 2. ViT processes 2D images patches that are flattened in a vector form and fed to the transformer as a sequence. These vectorized patches are then projected to a patch embedding using a linear layer, and position embedding is attached to encode location information. In addition, at the beginning of the input, a classification token is attached to the transformer. The output representation corresponding to the first position is then used as the global image representation for the image classification task.

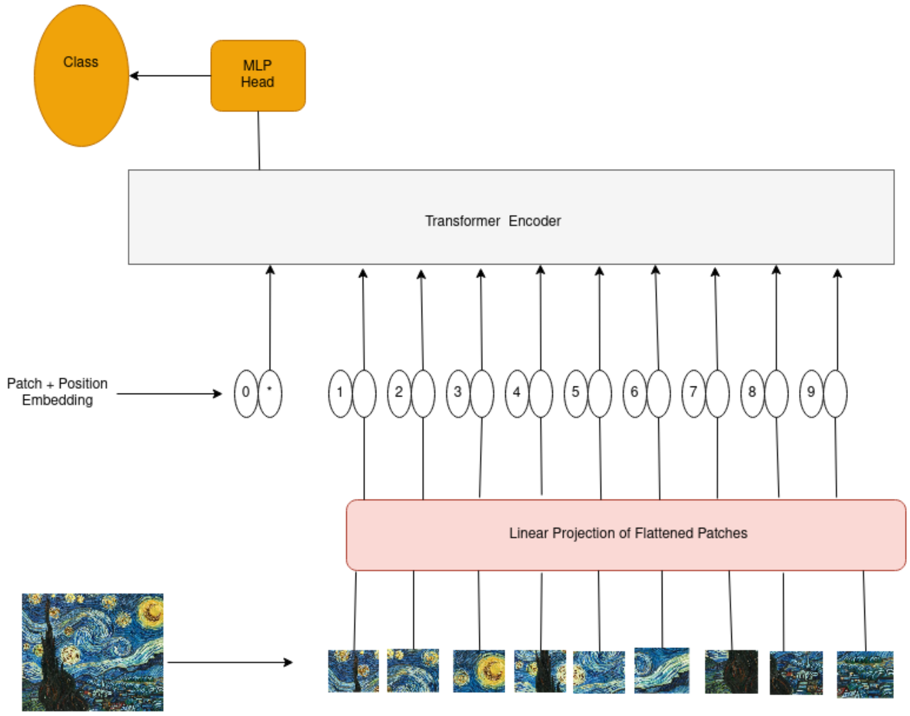


Figure 1. ViT model overview.

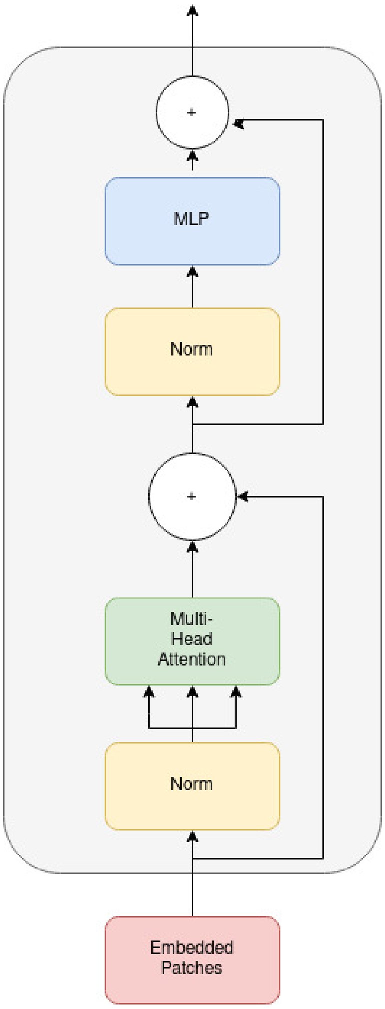


Figure 2. Transformer encoder: The illustration is inspired by [15].

3.2. MLP Mixer

MLP Mixer was recently introduced [16] as a CV model based solely on MLPs, without using either convolutions or attention mechanisms. The core idea behind MLP Mixer is to separate in a clear way using only MLPs, the per-location (channel-mixing) operations and the cross-location (token-mixing) operations.

Mixer takes as input a sequence of S non-overlapping image patches, each one projected to a desired hidden dimension C, thus obtaining a matrix $X \in \mathbb{R}^{S \times C}$. If the original input image has resolution (H, W), and each patch has resolution (P, P), then the number of patches is $S = \frac{H \cdot W}{P^2}$. All patches are linearly projected with the same projection matrix. Mixer consists of multiple layers of identical size, and each layer consists of two MLP blocks.

- The token-mixing MLP: it acts on columns of X and is shared across all columns.
- The channel-mixing MLP: it acts on rows of X and is shared across all rows.

Unlike ViT, MLP Mixer does not use position embeddings because the token-mixing MLPs are sensitive to the order of the input tokens.

3.3. Optimizers

Hyper-parameter optimization is a crucial part of DL training process. Image classification is usually considered a Supervised Learning task. In this framework, given a dataset, the learning algorithm is trained in such a way to minimize a suitably chosen cost function $\mathcal{L}$. An optimizer is needed to achieve the minimum of this function. For an extensive review of the most common used optimizers in DL, one may refer to [24]. Here the weights' update rule for each method is provided. In order to evaluate each optimizer "equally", none of the "tricks" that are proposed in the literature were used.

In the following, θ_t means the weight at step t, η_t is the learning rate, and v_t is the momentum. The rest of the parameters are explained in [24–26].

1. Stochastic Gradient Descent: Stochastic Gradient Descent (SGD) is one of the most used optimizers. SGD allows to update the network weights per each training image (online training).

$$\theta_{t+1} = \theta_t - \eta_t \nabla \mathcal{L}(\theta_t) \tag{1}$$

2. Momentum Gradient Descent: SGD may lead to oscillations during training. The best way to avoid them is the knowledge of the right direction for the gradient. This information is derived from the previous position, and, when considering the previous position, the updating rule adds a fraction of the previous update, which gives the optimizer the momentum needed to continue moving in the right direction. The weights in the Momentum Gradient Descent (MGD) are updated as

$$\begin{aligned} &v_0 = 0, \quad v_{t+1} = \gamma v_t + \nabla \mathcal{L}(\theta_t) \\ &\theta_{t+1} = \theta_t - \eta_t \nabla \mathcal{L}(\theta_t) \end{aligned} \tag{2}$$

3. Adam: Adam has been introduced as an algorithm for the first-order gradient-based optimization of stochastic objective functions, based on adaptive estimates of lower-order moments [25]. Adam has been established as one of the most successful optimizers in DL.

$$\begin{aligned} &m_0 = 0, \quad v_0 = 0 \\ &m_{t+1} = \beta_1 m_t + (1 - \beta_1) \nabla \mathcal{L}(\theta_t) \\ &v_{t+1} = \beta_2 v_t + (1 - \beta_2) \nabla \mathcal{L}(\theta_t)^2 \\ &b_{t+1} = \frac{\sqrt{1 - \beta_2^{t+1}}}{1 - \beta_1^{t+1}} \\ &\theta_{t+1} = \theta_t - a_t \frac{m_{t+1}}{\sqrt{v_{t+1} + \epsilon}} b_{t+1} \end{aligned} \tag{3}$$

4. AdaMax: AdaMax is a generalisation of Adam from the l_2 norm to the l_∞ norm [25].
5. Optimistic Adam: Optimistic Adam (OAdam) optimizer [26] is a variant of the ADAM optimizer. The only difference between OAdam and Adam is the weight update,

$$\theta_{t+1} = \theta_t - 2\frac{\eta_t}{\sqrt{v_{t+1} + \epsilon}} b_{t+1} + \frac{\eta_t}{\sqrt{v_t + \epsilon}} b_t \tag{4}$$

6. RMSProp: Using some Adaptive Gradient Descent Optimizers leads, in some cases, the learning rate to decrease monotonically because every added term is positive. After many epochs, the learning rate is so small that it stops updating the weights. The RMSProp method proposes

$$\begin{aligned} &v_0 = 1, \quad m_0 = 0 \\ &v_{t+1} = \rho v_t + (1-\rho)\nabla\mathcal{L}(\theta_t)^2 \\ &m_{t+1} = \gamma m_t + \frac{\eta_t}{\sqrt{v_{t+1} + \epsilon}} \\ &\theta_{t+1} = \theta_t - m_{t+1} \end{aligned} \tag{5}$$

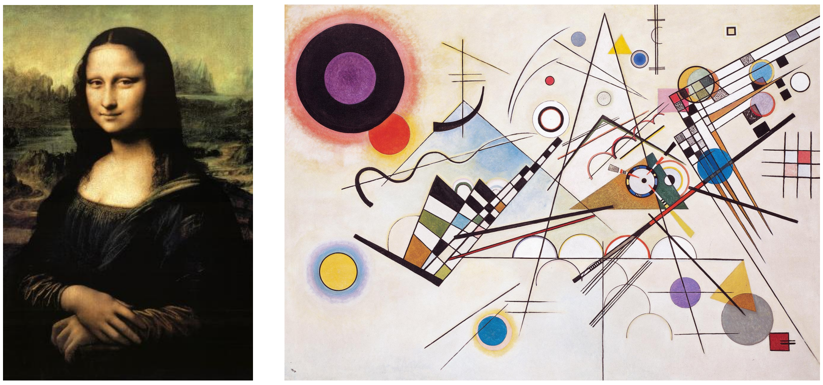
3.4. WikiArt Dataset

The dataset used in this work is called WikiArt paintings dataset and has been collected from the ArtGan [22] GitHub repository website (https://github.com/cs-chan/ArtGAN/tree/master/WikiArt%20Dataset, accessed on 20 July 2021) with its respective metadata.

The dataset contains 81,446 images tagged with one corresponding style among the 27 following styles: Abstract Expressionism (2782 images), Action Painting (92 images), Analytical Cubism (110 images), Art Nouveau (4334 images), Baroque (4241 images), Color Field Painting (1615 images), Contemporary Realism (481 images), Cubism (2235 images), Early Renaissance (1391 images), Expressionism (6736 images), Fauvism (934 images), High Renaissance (1343 images), Impressionism (13,060), Mannerism (Late Renaissance) (1279 images), Minimalism (1337 images), Naive Art/Primitivism (2405 images), New Realism (314 images), Northern Renaissance (2552 images), Pointillism (513 images), Pop Art (1483 images), Post Impressionism (6451 images), Realism (10,733 images), Rococo (2089 images), Romanticism (7049 images), Symbolism (4528 images), Synthetic Cubism (216 images) and Ukiyo-e (1167 images).

The WikiArt dataset is highly unbalanced. To avoid some of the issues that may follow, Action Painting and Pointillism classes were dropped. In addition, Analytical Cubism and Synthetic Cubism classes were incorporated into the Cubism class and similarly Contemporary Realism and New Realism were transferred into Realism class. The resulting dataset was comprised of 21 classes and 80,835 images in total. The train, validation and test sets were 60%, 20% and 20% of the whole dataset respectively.

In Figure 3, two samples of the dataset are shown, highlighting the diversity of the artwork style.



(**a**) (**b**)

Figure 3. Samples from the WikiArt dataset. (**a**) Mona Lisa; (**b**) Composition VIII.

4. Results and Discussion

4.1. Experiments

Two types of experiments were conducted: In the first, a comparative study between the most common used optimizers in DL was conducted as part of ViT training process. In the second experiment, MLP Mixer was trained from scratch on the same task, drawing in this way useful information for future studies regarding complex CV problems. The dataset was pre-processed as described in the previous section.

4.1.1. ViT: Optimizers' Performance

For the purposes of this study, ViT is comprised of eight heads, the dropout is set to 0.2, and the image size is 256 × 256. The training was performed on a single NVIDIA RTX 2070 GPU, for 100 epochs. The experiments were conducted using Python DL libraries Pytorch and Sci-kit learn, along with the "classical" ones like Matplotlib, Pandas and Numpy.

Since many optimizers are frequently used in different CV tasks, in the first experiment a comparative study between the most common used ones, was conducted. The scope of this experiment is to evaluate ViT' s performance on the task of artistic style recognition, setting in the same time a minimum benchmark for future studies. Five-fold cross-validation was used. The results are shown in Table 2.

Table 2. ViT performance in artwork style recognition.

Optimizer	Accuracy
Adam	39.89%
Adamax	39.42%
Optimistic Adam	39.71%
SGD	39.28%
MGD	39.31%
RMSProp	38.97%

From Table 2 it follows that the Adam optimizer outperforms the others, achieving over 39% prediction accuracy. The test loss for the best optimizer is depicted in Figure 4. The loss is calculated on test set and it shows how well the model is doing for this set. The loss function that is used is negative log-likelihood. It is clear that that the test procedure was stable enough, and it is depicted that the model learns through training. Table 3 provides a better visualization of the results since it shows the prediction accuracy per class. However, it should be noted that the difference in the results is not significant between the different optimizers.

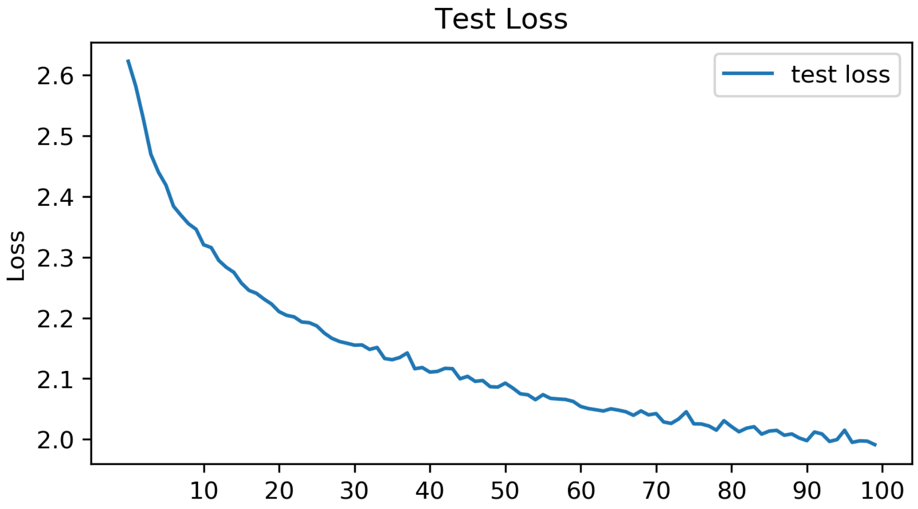


Figure 4. ViT: Adam training test loss.

Table 3. ViT accuracy per class.

Class	Accuracy %
Abstract Expressionism	29.6
Art Nouveau	25.3
Baroque	48.3
Color Field Painting	65.5
Cubism	21.3
Early Renaissance	34.1
Expressionism	28.0
Fauvism	17.5
High Renaissance	8.5
Impressionism	65.0
Mannerism Late Renaissance	18.1
Minimalism	49.4
Naive Art / Primitivism	15.4
Northern Renaissance	6.1
Pop Art	14.6
Post Impressionism	30.6
Realism	57.7
Rococo	45.3
Romanticism	37.4
Symbolism	28.4
Ukiyo-e	68.3

Table 3 shows that the model performs really well on the Ukiyo-e class, the Color Field Painting class, Impressionism and the Realism class and also achieves a good performance in Baroque and Rococo.

4.1.2. MLP Mixer Performance

In the second experiment MLP Mixer, a recent DL architecture that does not use either convolutions or attention, was trained from scratch on artwork style recognition. The scope of this experiment is to compare the two advancements in CV architectures and to determine if a lighter model, like MLP Mixer, is able to achieve good results on a complex CV task.

For the purposes of this study, MLP Mixer has a depth equal to 8, the dropout is set to 0.3, and the image size is 256 × 256. The training was also performed on a single

NVIDIA RTX 2070 GPU, for 100 epochs. The experiments were conducted using Python DL libraries Pytorch and Sci-kit learn, along with the "classical" ones like Matplotlib, Pandas and Numpy.

The test loss in the test set is shown in Figure 5. The results show that the learning process was not stable. Perhaps more work on the estimation of model's hyper-parameters is needed. However, the MLP Mixer achieves a prediction accuracy similar to that of ViT Table 4. This result indicates that a lighter architecture can be applied in such complicated tasks. For a better visualization of the results, a matrix showing the accuracy per class is built Table 5.

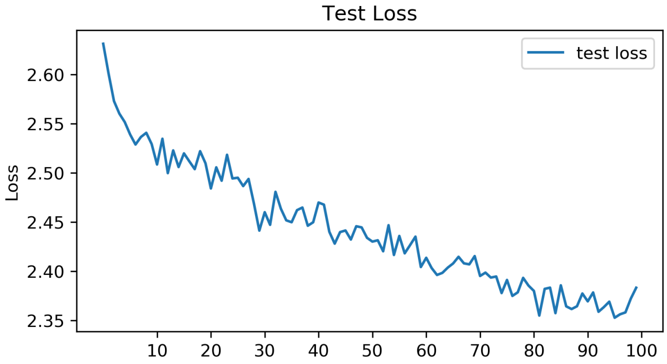


Figure 5. MLP Mixer: test loss.

Table 4. MLP Mixer performance in artwork style recognition.

Model	Accuracy
MLP Mixer	39.59%

Table 5. MLP Mixer accuracy per class.

Class	Accuracy %
Abstract Expressionism	29.0
Art Nouveau	34.3
Baroque	46.2
Color Field Painting	64.9
Cubism	20.4
Early Renaissance	34.1
Expressionism	28.0
Fauvism	16.8
High Renaissance	23.1
Impressionism	62.0
Mannerism Late Renaissance	15.1
Minimalism	46.6
Naive Art / Primitivism	25.5
Northern Renaissance	5.3

Table 5. *Cont.*

Class	Accuracy %
Pop Art	22.1
Post Impressionism	30.4
Realism	44.3
Rococo	44.4
Romanticism	36.2
Symbolism	28.2
Ukiyo-e	60.9

Table 5 shows that the model performed well on the Ukiyo-e class, Color Field Painting class and Impressionism class and also achieved good performance in Baroque, Realism and Rococo. ViT and MLP Mixer learned different classes better, and a further investigation of the learned parameters is set as a future goal.

5. Conclusions

In this paper, the Vision Transformers ViT and MLP Mixer were successfully applied on the WikiArt dataset in the artistic style recognition task. ViT was trained from scratch in the WikiArt dataset achieving over 39% accuracy for 21 classes, thus, setting a minimum benchmark in accuracy prediction for future studies. In addition, a comparative study was conducted among the most common used optimizers, which showed that training with the Adam optimizer and Optimistic Adam optimizer resulted in better performance. Using the above results, MLP Mixer was trained from scratch, performing close to ViT in terms of prediction accuracy. As suggested by our experiments and literature, the use of larger datasets with richer resources should improve the accuracy of the models.

Future work on this subject will be focused on improvements on the models' hyper-parameters through parametric studies and other experiments. Variations of the models that were used here may provide better results, especially with the combination of other CV techniques. In addition, the creation of a larger dataset will provide a better overview of the tested models' prediction accuracy.

Author Contributions: Conceptualization, S.K.G. and L.A.I.; methodology, L.A.I.; software, L.A.I.; validation, S.N., P.S., S.W. and S.K.G; formal analysis, L.A.I. and S.K.G.; investigation, S.N., P.S. and S.W.; resources, S.N.; data curation, L.A.I.; writing—original draft preparation, L.A.I.; writing—review and editing, S.N., P.S., S.W. and S.K.G.; visualization, L.A.I.; supervision, S.K.G.; project administration, S.K.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: WikiArt dataset was downlowded from the following Github repository https://github.com/cs-chan/ArtGAN/tree/master/WikiArt%20Dataset and it was accessed on 20 July 2021.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CNN	Convolutional Neural Networks
CV	Computer Vision
DL	Deep Learning
GAN	Generative Adversarial Network
MLP	Multi-Layered Perceptron
NLP	Natural Language Processing
ViT	Visual Transformer

References

1. Waseem, R.; Zenghui, W. Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review. *Neural Comput.* **2017**, *29*, 2352–2449. [CrossRef]
2. Olejnik, A.; Borecki, M.; Rychlik, A. A simple detection method of movement of clouds at the sky. In Proceedings of the SPIE 11581, Photonics Applications in Astronomy, Communications, Industry, and High Energy Physics Experiments, Wilga, Poland, 14 October 2020; p. 1158111. [CrossRef]
3. Stabinger, S.; Rodríguez-Sánchez, A. Evaluation of Deep Learning on an Abstract Image Classification Dataset. In Proceedings of the 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), Venice, Italy, 22–29 October 2017; pp. 2767–2772. [CrossRef]
4. Elgammal, A.; Liu, B.; Kim, D.; Elhoseiny, M. The Shape of Art History in the Eyes of the Machine. In Proceedings of the AAAI, Palo Alto, CA, USA, 2–7 February 2018.
5. Johnson, C.R.; Hendriks, E.; Berezhnoy, I.J.; Brevdo, E.; Hughes, S.M.; Daubechies, I.; Li, J.; Postma, E.; Wang, J.Z. Image processing for artist identification. *IEEE Signal Process. Mag.* **2008**, *25*, 37–48. [CrossRef]
6. Altenburgera, P.; Kämpferb, P.; Makristathisc, A.; Lubitza, W.; Bussea, H.-J. Classification of bacteria isolated from a medieval wall painting. *J. Biotechnol.* **1996**, *47*, 39–52. [CrossRef]
7. Li, C.; Chen, T. Aesthetic Visual Quality Assessment of Paintings. *IEEE J. Sel. Top. Signal Process.* **2009**, *3*, 236–252. [CrossRef]
8. Lecoutre, A.; Negrevergne, B.; Yger, F. Recognizing Art Style Automatically in Painting with Deep Learning. In Proceedings of the Ninth Asian Conference on Machine Learning, Seoul, Korea, 15–17 November 2017; pp. 327–342.
9. Bar, Y.; Levy, N.; Wolf, L. Classification of Artistic Styles Using Binarized Features Derived from a Deep Neural Network. In *Lecture Notes in Computer Science Proceedings of the ECCV Workshops, Zurich, Switzerland, 6–7 September 2014*; Springer: Cham, Switzerland, 2014.
10. Cetinic, E.; Lipic, T.; Grgic, S. Fine-tuning Convolutional Neural Networks for Fine Art Classification. *Expert Syst. Appl.* **2018**, *114*, 107–118. [CrossRef]
11. Huang, X.; Zhong, S.; Zhijiao, X. Fine-Art Painting Classification via Two-Channel Deep Residual Network. In *Lecture Notes in Computer Science, Proceedings of the Advances in Multimedia Information Processing, Harbin, China, 28–29 September 2017*; Springer: Cham, Switzerland, 2017. [CrossRef]
12. Sandoval, C.; Pirogova, E.; Lech, M. Two-Stage Deep Learning Approach to the Classification of Fine-Art Paintings. *IEEE Access* **2019**, *7*, 41770–41781. [CrossRef]
13. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.; Kaiser, L.; Polosukhin, I. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17), Long Beach, CA, USA, 4–9 December 2017; Curran Associates Inc.: Red Hook, NY, USA, 2017; pp. 6000–6010.
14. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-End Object Detection with Transformers. In *Lecture Notes in Computer Science, Proceedings of the Computer Vision – ECCV 2020, Glasgow, UK, 23–28 August 2020*; Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M., Eds.; Springer: Cham, Switzerland, 2020; Volume 12346. [CrossRef]
15. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Gelly, S. An Image is Worth 16 × 16 Words: Transformers for Image Recognition at Scale. In Proceedings of the ICLR 2021: The Ninth International Conference on Learning Representations, Vienna, Austria, 3–7 May 2021.
16. Tolstikhin, I.O.; Houlsby, N.; Kolesnikov, A.; Beyer, L.; Zhai, X.; Unterthiner, T.; Yung, J.; Keysers, D.; Uszkoreit, J.; Lucic, M.; et al. MLP-Mixer: An all-MLP Architecture for Vision. *arXiv* **2021**, arXiv:2105.01601.
17. Amin, F.; Choi, G.S. Advanced Service Search Model for Higher Network Navigation Using Small World Networks. *IEEE Access* **2021**, *9*, 70584–70595. [CrossRef]
18. Amin, F.; Ahmad, A.; Sang Choi, G. Towards Trust and Friendliness Approaches in the Social Internet of Things. *Appl. Sci.* **2019**, *9*, 166. [CrossRef]
19. Gatys, L.; Ecker, A.; Bethge, M. Image Style Transfer Using Convolutional Neural Networks. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 2414–2423. [CrossRef]
20. Gatys, L.A.; Ecker, A.S.; Bethge, M.; Hertzmann, A.; Shechtman, E. Controlling Perceptual Factors in Neural Style Transfer. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 June 2017; pp. 3730–3738. [CrossRef]

21. Johnson, J.; Alahi, A.; Fei-Fei, L. Perceptual Losses for Real-Time Style Transfer and Super-Resolution. In *Lecture Notes in Computer Science, Proceedings of the Computer Vision—ECCV 2016, Amsterdam, The Netherlands, 11–14 October 2016*; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Springer: Cham, Switzerland, 2016; Volume 9906. [CrossRef]
22. Tan, W.R.; Chan, C.S.; Aguirre, H.E.; Tanaka, K. ArtGAN: Artwork synthesis with conditional categorical GANs. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; pp. 3760–3764. [CrossRef]
23. Tan, W.R.; Chan, C.S.; Aguirre, H.E.; Tanaka, K. Improved ArtGAN for Conditional Synthesis of Natural Image and Artwork. *IEEE Trans. Image Process.* **2019**, *28*, 394–409. [CrossRef] [PubMed]
24. Choi, D.; Shallue, C.; Nado, Z.; Lee, J.; Maddison, C.; Dahl, G. On Empirical Comparisons of Optimizers for Deep Learning. *arXiv* **2020**, arXiv:1910.05446.
25. Kingma, D.; Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, 7–9 May 2015.
26. Daskalakis, C.; Ilyas, A.; Syrgkanis, V.; Zeng, H. Training GANs with optimism. In Proceedings of the International Conference on Learning Representations, Vancuver, BC, Canada, 30 April–3 May 2018. Available online: https://openreview.net/forum?id=SJJySbbAZ (accessed on 20 September 2021).