Efficient Domain Adaptation for Painting Theme Recognition

Mihai Badea, Corneliu Florea, Laura Florea, Constantin Vertan
Image Processing and Analysis Laboratory, University "Politehnica" of Bucharest

***Abstract*— In this paper we approach the problem of scene recognition in paintings. We tackle this task with the aid of Convolutional Neural Networks and a large database consisting of around 80,000 paintings. The main purpose is to identify an efficient method to enlarge the database by domain transfer from photographic content to artistic content. Thus, we discuss the practical capabilities of a recent method of domain transfer from photographs to paintings while augmenting the employed database and aid the learning of difficult styles. We propose a set of improvements to increase the feasibility of the domain transfer in the context of large databases.**

I. Introduction

Mankind has long surpassed its basic needs for survival, as it has begun to ask deeper and more philosophical questions. It is as Pablo Picasso once said, "Art is the lie that enables us to realize the truth". In time, that art started asking deeper questions, and its representation of life became less realistic and more abstract. It is only natural to test our computer vision systems in human situations, since performance in mundane tasks is getting closer to human level.

Probably, the current most powerful tools of the moment in various computer vision problems are the Deep Convolutional Neural Networks. They have brought a huge leap in the field, with incredible results. Their rise is mainly credited to an increase in processing power brought by the development of CUDA technology and to constant increase in available image data. Thus, with the aid of GPU acceleration, highly parallel architectures have gained significant momentum. At this moment, CNNs are the state of the art in tasks such as object detection and recognition, semantic segmentation, etc.

Yet, the CNN are of use only in the context of an adequate database. A continuous effort to digitize paintings has led to the creation of consistent databases. such as WikiArt[1], which contains well over 80,000 paintings spanning across multiple styles (art movements), genres (the main theme of the paintings) and time periods. In this paper we are interested recognizing the scene of the painting (genre). Style information, which also carries information about the abstraction level was also studied, as we are interested in the influence of abstraction over the discriminative capabilities of the networks.

Related work. Research in automated painting analysis is by no means a new thing. Recent years have seen quite an interest in this particular field, as emphasized in the review of Bentowska and Coddington [2]. One main difference with the work we have conducted is that most papers in the past have studied discrimination between styles or artists. With a theme more similar to ours, Crowley and Zisserman have studied object detection in the YourPaintings dataset [4]. Some research in scene recognition has already been conducted in the past. One such endeavor is in [1]where it is studied a small databases divided into only 5 classes, with a total of 1500 images. Similar to this approach, Condorovici et al. [3], which addressed the problem on a database of 500 images. The former two papers examine the problem in a more classical manner, with a feature and classifier system.

The development of the WikiArt database has made this issue far more interesting. With a considerable larger database, Saleh and Elgammal have explored various features and metric learning to improve the similarity measure between paintings [11]. By contrast, Tan et al. [13], approach the problem by employing CNN. The chosen architecture, AlexNet [10], is initialized on the ImageNet database and then used to separate between different scenes and styles.

Recently, the work of Gatys et al [8] proposes a novel algorithm which separates the content and the style of an image, performing style transfer between different images. This innovation has been used, up to this point, mainly for creative purposes.

We have tackled the issue of scene recognition in paintings using CNNs. The presence of multiple inhomogeneous styles is a worthwhile impediment in a task otherwise simple for humans. To combat this issue, active learning and a optimized version of the style transfer method mentioned earlier [7]. In this paper we further investigate the transfer method. In the standard implementation it takes around 20 minutes for a new pseudo painting to be produced. While this is a short time compared to a living painter, it is inefficient for pattern recognition purposes, where the database size matters. Given the 80,000 paintings from WikiArt, to produce an additional 30,000 pseudo paintings, which may have an impact, one needs 400 days. Thus efficientization methods are needed for practical applications.

The remainder of the paper is organized in the following manner: the second section presents the style transfer algorithm and its optimization, section presents the used databases in-depth and the forth describes the conducted experiments. The paper is ended with a discussion regarding the results.

[1] http://www.wikiart.org/

II. Style Transfer and Optimization

A. Algorithm

The algorithm [8] is based on the hierarchical structure of CNN architectures. Basically, these types of networks are a series of filters which extract increasingly complex information on each layer. Lower level layers are usually used to extract low level features (edges, corners, basic shapes), while higher layers are composed of more complex filters, good at detecting precise objects or other high level features [16]. Gatys et al. [8] suggested the idea that style and content can be separated using the nature of the aforementioned layer hierarchy. In other words, the content of an image is well represented in high level layers, while lower level layers are more representative for style.

The aim of the algorithm is to create a new image with the content of an image and the style of another one. To achieve this, a white noise image is initialized and then undergoes an optimization process until it matches the desired content and style. The target function is a composite of two different loss functions, one for content and one for style.

The loss function for content matching uses feature map activations. The input image and the content image are processed in the network up to different desired levels, and the squared error between the activations of their feature maps is calculated. Thus, for the desired image $\vec{p}$ and the image to be optimized, $\vec{x}$, the loss function in layer l is defined as follows:

$$\mathcal{L}_{content}(\vec{p}, \vec{x}, l) = \frac{1}{2} \sum_{i,j} (F_{ij}^l - P_{ij}^l)^2, \tag{1}$$

where F_{ij}^l is the activation in layer l of filter i at position j. Comparing activation maps is not however adequate for matching styles. If the same error was used, the source and destination images would be matched at pixel level. For this purpose, an approach using Gram matrices has been used. The function to be minimized is the mean-squared distance between the different Gram matrices E_l of the source and destination images, $\vec{a}$ and $\vec{x}$:

$$\mathcal{L}_{style}(\vec{a}, \vec{x}) = \sum_{l=0}^{L} w_l E_l, \tag{2}$$

where w_l are a set of weights corresponding to each layer taken into account. Matching the Gram matrices means that response to specific filters has to matched and not the content. Thus a superior level of abstraction can be reached. With the defined equations, the compound loss is a weighted sum described by:

$$\mathcal{L}_{total}(\vec{p}, \vec{a}, \vec{x}) = \alpha \mathcal{L}_{content}(\vec{p}, \vec{x}) + \beta \mathcal{L}_{style}(\vec{a}, \vec{x}). \tag{3}$$

B. Optimization

The whole process implies multiple passes of the output image through the large VGG19 network [12] and a time consuming optimization process. For these reasons, it is not practical to create many outputs to be used in data augmentation.

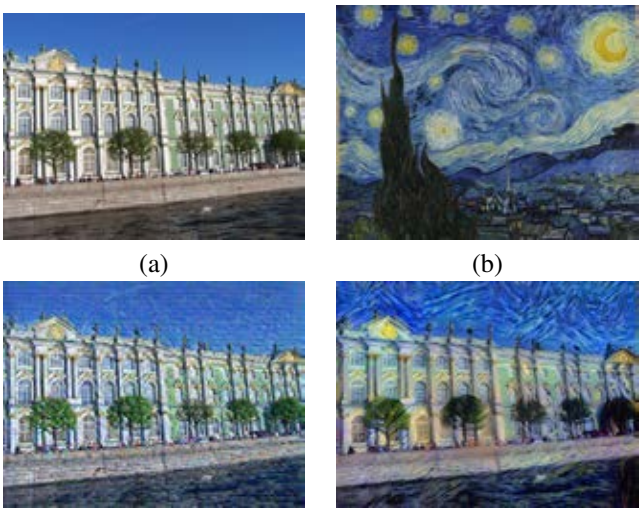


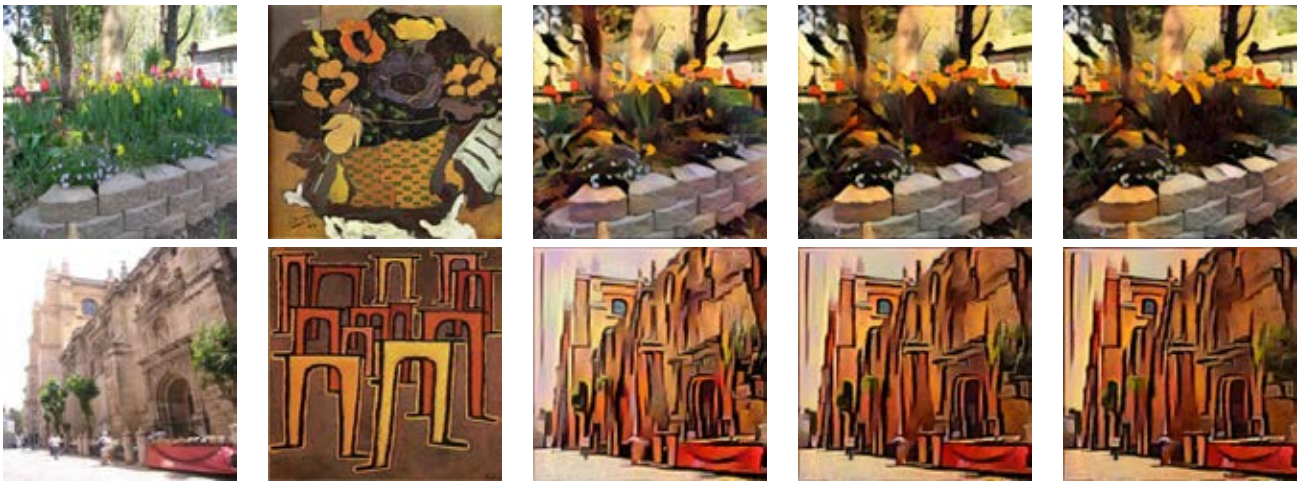
Fig. 1. Comparison of styling process when using ResNet and VGG networks: (a) photograph taken from SUN database, (b) painting, (c) results using ResNet, and respectively with (d) VGG-19.

A first attempt in trying to optimize the process was to change the network. Our implementation is based on publicly available code[2]. The typically used networks are VGG19 and VGG16, as they were the deepest at the moment of publication of original algorithm and thus enforced multiple verifications according to eq. (1). We have tested an alternative in the form of the ResNet-50 network [9]. This architecture has significantly fewer parameters than the usual ones, yet its significant depth bear promises. Unfortunately, the results were less than satisfactory, because while the content was correctly represented, the style was far from resembling any desired output. Our explanation is that the shortcut connection specific to ResNet, that permitted much more efficient use of parameters, allowed images to pass too fast with actual enforcing the blending process.

An alternative, which turned to be an efficient way reduce the time needed for transfer, is to limit the optimization process. If left until convergence, transfers usually take more than 450 iterations. However, the Stochastic Gradient Descent used, has the initial steps larger, while the final ones are smaller, to refine the performance. Thus we anticipate that using the initial part may suffice.

A study was conducted on the influence of the number of iterations on the output. Figure 2 shows that there are aesthetic differences between the versions that are noticeable by the human eye. Yet the similarities are evident and the 100 iterations version should be tested for practical uses. Data from experiment 2 in table II was obtained with 100 iterations. The results shows that the later part of the convergence is not needed for transfer purposes and the *performance of the transfer is kept although the time is less than one forth.*

[2]https://github.com/fzliu/style-transfer



(a) (b) (c) (d) (e)

Fig. 2. Influence of the number of iterations on the end result. (a) is the original photo, while (b) is the original painting (c),(d) and(e) are the output when the process runs for 100 iterations, 250 iterations, 500 iterations, respectively. By zooming in, one will find some "ringing" artifacts in images produced after small number of iterations.

III. Databases

Although there are multiple databases which portray digitized paintings, we have chosen WikiArt, since it has, to our knowledge, the widest range of paintings. As an auxiliary database, we have used the SUN database [15]. We have chosen certain sets of photos to augment some WikiArt classes, with the aid of the presented style transfer algorithm.

A. WikiArt

The WikiArt paintings database contains an approximate 100,000 samples out of which 80,000 are annotated precisely in terms of genre. The database has labels for scene types, named genre (45 classes), for artistic styles (27 classes) and artists (well over 1000 classes). Many of the scenes types were not well represented (¡200 examples), which lead to their inclusion in a collector class, "Others". This operation has lead to a significant decrease in the number of classes, from 45 to 26. Even now there are labelling issues, with some arguable annotations: there is a "Genre paintings" class, which works more or less like a collector class, with no solid scene type. Also, some paintings in the "Literary" class can be considered as "Landscapes" when referring to the scene.

Other works ([11],[13]) preferred keeping only the 10 most significant classes: Abstract paintings, Cityscape, Genre painting, Illustration, Landscape, Nude painting, Portrait, Religious painting, Sketch and Study, Still Life.

B. SUN database

The auxiliary database is composed of over 130,000 photographs, grouped into a total of 899 classes. Although it features a large number of samples, only some classes are of use for our experiments, so their impact on the size of the database is not as large as expected.

IV. Experiments

Our main focus in the course of experiments was to study the various ways to improve the genre recognition performance and the impact of different styles on the end results. Besides using CNNs for classification, we have employed more traditional approaches, with the use of features: a HOG pyramid [5], a LBP pyramid [14] and DeCAF [6], which uses the first layers of AlexNet as a feature extractor. All these methods have been then used as input for SVM classifiers.

As far as CNNs go, our main network was a ResNet-34 [9], but tests on the significantly smaller AlexNet were also conducted. The ResNet-34 networks were not initialized on ImageNet (they were trained from scratch to allow further domain adaptation) and used to learn both the 26 class case, and the 10 class one. Details about this part can be found in our previous work [7][3]. The results are summarized in Table I. It shows that the ResNet-34 is top performer in the genre recognition problem and, thus, the remaining experiments will continue with it. Some of the misclassified paintings are featured in Figure 3. A few of the paintings are understandably hard for the classifier, although some fairly simple examples are left out.

The next step was represented by testing the impact of styled photographs when it comes to augmenting three classes of the database (Cityscape, Flower paintings, Marina). After creating new samples from photographs, an abridged version of the database was created, as to keep the number of psuedo paintings similar to the one of paintings and training times low. The condensed databases feature a maximum 250 (where possible) from each of the classes.

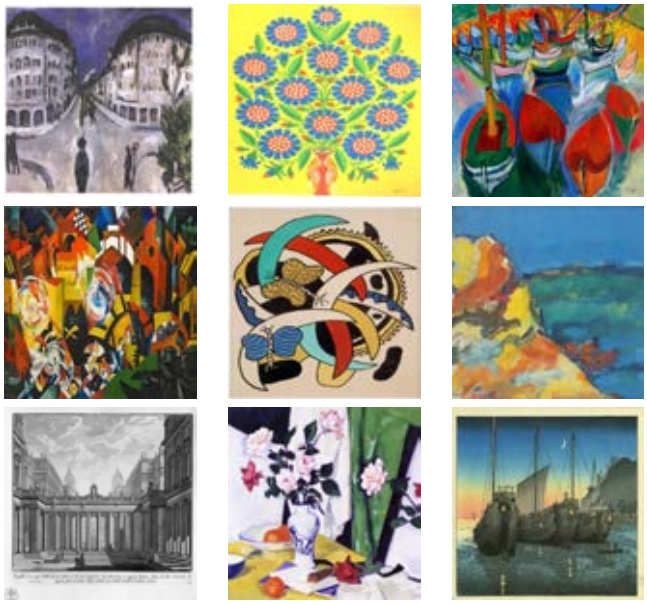
The problem was approached in two different ways: (1) new samples from random combinations of photographs and

[3]Available at http://imag.pub.ro/pandora/pandora_publications.html

TABLE I

COMPARISON BETWEEN DIFFERENT METHODS OF SCENE RECOGNITION

Method	**No. classes**	**No. images**	**Test ratio**	**Acc** (%)
[13] AlexNet - scratch	10	63.691	n/a	69.29
[13] CNN- finetune				**74.14**
[7] *ResNet 34 - scratch*			20%	*73.74*
[7] pHoG + SVM	26	79,434	20%	44.37
[7] pLBP + SVM				39.58
[7] DeCAF + SVM				59.05
[7] AlexNet - scratch				53.02
[7] *ResNet 34 - scratch*				**61.15**



Cityscape Flower paintings Marina

Fig. 3. Examples of misclassified paintings from the training set

paintings (of the same class); (2) paintings which were incorrectly classified at the end of the training process are chosen as style sources. Experiment (1), reported in prior work [**?**] assume full convergence (450-550 iterations). Experiment (2) assumes partial convergence by only 100 iterations. Direct comparison between (1) and (2) shows insignificant reduction in performance.

The set styled with random paintings and the one styled with hard examples are extremely overlapped in regard with the photographs from the SUN database. Table II shows that the result do not improve noticeably when both sets are used together. This is an indicator of how repeated content may not be of any use in further augmentation of the database.

V. CONCLUSION

We have proven the practical potential of the style transfer algorithm to work as domain adaptation between a photograph database and a painting database. The tradeoff which led to *significantly decreased processing time* is well worth the marginal loss in image quality as the transfer is still effective. Large numbers of styled photographs may lead to increased performance, and more independence in regard to the style of the paintings.

ACKNOWLEDGMENT

The work was supported by a grant of the Romanian National Authority for Scientific Research and Innovation, CNCS UEFISCDI, number PN-II-RU-TE-2014-4-0733.

REFERENCES

[1] S. Agarwal, H. Karnick, N. Pant, and U. Patel. Genre and style based painting classification. In *WACV*, pages 588–594, 2015.
[2] A. Bentkowska-Kafel and J. Coddington. Computer vision and image analysis of art. In *SPIE*, 2010.
[3] R. Condorovici, C. Florea, and C. Vertan. Painting scene recognition using homogenous shapes. In *ACIVS*, pages 262–273, 2013.
[4] E. Crowley and A. Zisserman. The state of the art: Object retrieval in paintings using discriminative regions. In *BMVC*, 2014.
[5] P. Dollar, R. Appel, S. Belongie, and P. Perona. Fast feature pyramids for object detection. *IEEE Trans. on PAMI*, 36(8):1532–1545, 2014.
[6] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML*, 2014.
[7] C. Florea, M. Badea, L. Florea, and C. Vertan. Painting genre recognition by deep neural networks and domain transfer. Technical report, LAPI UPB, 2016. Submitted for evaluation.
[8] L. Gatys, A. Ecker, and M. Bethge. A neural algorithm of artistic style. In *CVPR*, 2016.
[9] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.
[10] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105, 2012.
[11] B. Saleh and A. Elgammal. Large-scale classification of fine-art paintings: Learning the right metric on the right feature. In *International Conference on Data Mining – Workshops*, 2015.
[12] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.
[13] W. R. Tan, C. S. Chan, H. Aguirre, and K. Tanaka. Ceci n'est pas une pipe: A deep convolutional network for fine-art paintings classification. In *ICIP*, 2016.
[14] A. Vedaldi and B. Fulkerson. Vlfeat: An open and portable library of computer vision algorithms. In *ACM MM*, pages 1469–1472, 2010.
[15] J. Xiao, J. Hays, K. Ehinger, A. Oliva, and A. Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*, pages 3485–3492, 2010.
[16] M. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *ECCV*, 2014.

TABLE II

GENRE RECOGNITION ACCURACY, WHEN THE NETWORK WAS TRAINED WITH FEW EXAMPLES PER CLASS AND WITH THE NEURAL STYLE TRANSFER [8]. FOR OTHER CLASSES, IN ALL THREE EXPERIMENTS WE HAVE KEPT A MAXIMUM OF 250 PAINTINGS/CLASS .

Exp. No.	**Class**	**Transferred**		**Testing**	**Recogn. images**
		Random	From hard examples		On given classes
1	Cityscape	262	0	764	287
	Flower paint.	180	0	252	141
	Marina	229	0	259	145
2	Cityscape	0	250	764	257
	Flower paint.	0	230	252	115
	Marina	0	250	259	133
3	Cityscape	262	250	764	264
	Flower paint.	180	180	252	102
	Marina	229	229	259	133