Towards an Inpainting Framework for Visual Cultural Heritage

Nesreen Hamdallah Jboor
CSE, College of Engineering
Qatar University, Doha, Qatar
nj1700648@student.qu.edu.qa

Abdelhak Belhi
[1]CSE, College of Engineering
Qatar University, Doha, Qatar
[2]DISP Laboratory, Université Lumière
Lyon 2, Lyon, France
abdelhak.belhi@qu.edu.qa

Abdulaziz Khalid Al-Ali
CSE, College of Engineering
Qatar University, Doha, Qatar
a.alali@qu.edu.qa

Abdelaziz Bouras
CSE, College of Engineering
Qatar University, Doha, Qatar
abdelaziz.bouras@qu.edu.qa

Ali Jaoua
CSE, College of Engineering
Qatar University, Doha, Qatar
jaoua@qu.edu.qa

Abstract— **Cultural heritage takes an important part in defining the identity and the history of a civilization or a nation. Valuing and preserving this heritage is thus a top priority for governments and heritage institutions. Through this paper, we present an image completion (inpainting) approach adapted for the curation and the completion of damaged artwork. Our approach uses a set of machine learning techniques such as Generative Adversarial Networks which are among the most powerful generative models that can be trained to generate realistic data samples. As we are focusing mostly on visual cultural heritage, the pipeline of our framework has many optimizations such as the use of clustering to optimize the training of the generative part to ensure a better performance across a variety of cultural data categories. The experimental results of our framework are promising and were validated on a dataset of paintings.**

Keywords— Image Inpainting, Generative Adversarial Networks, Deep Learning, Cultural Heritage

I. Introduction

Cultural heritage assets or artifacts have a very important value as they represent the most effective way to transfer history and knowledge from a generation to another. Protecting and preserving these assets is one of the top priorities for a nation that wants to preserve its moral identity. Regrettably, some of these assets lose their value due to their physical damage where sometimes an important part is missing. Heritage institutions such as museums often ask for the help of art specialists and professional curators to recover these missing regions. Often, the process takes a lot of time and requires abundant financial resources which may be impractical in many cases [1].

Thanks to the recent progress of machine learning, and with data sources becoming available for researchers, tackling such a challenge was never this possible. In fact, many research efforts are dedicated for techniques related to data completion and more specifically the ones used to complete visual data. The term image inpainting is used to describe image completion tasks. Multiple machine learning based techniques were used to address the image inpainting challenge, but with the rise of deep learning, these approaches saw a big leap forward mostly due to the superior performance observed on image reconstruction tasks using auto-encoders and restricted Boltzmann machines. The most important contribution is without a doubt Generative Adversarial networks [2] (GANs) which are nowadays among the best performing generative models for multiple tasks related to computer vision such as super-resolution[3] , unsupervised image generation [4], etc. GANs are also used for unsupervised image inpainting tasks where their performance is considered as state of the art [5].

However, due to the intricacies in addition to the diversity of cultural artwork, it is clear that trying to complete any visually incomplete asset is a tough challenge even for long term human experts. Solving this challenge using computer-based tools is even harder. In this paper, we mainly focus on approaches based on generative adversarial networks which are known for their very good performance for this type of challenges. Most of the approaches focus on completing images from specific visual categories, such as completing faces, human postures, building facades, etc. [5]. However, cultural heritage assets commonly span multiple categories which makes the existing solutions not viable. Through our analysis and experiments, we found that it is rather inefficient to design an inpainting approach based on a single generative model to address several contexts.

Through this paper, we propose a new image inpainting framework for visual cultural data that uses a divide-and-conquer strategy based on clustering. The principle consists of clustering similarly looking cultural images and then training a generative model for each category. When presented with an incomplete image, the system first identifies the category of that image and use the associated GAN for the visual completion process.

This paper is organized as follows. In section two, we present an overview of the literature related to image inpainting and digital curation of visual content. Section three gives some technical details regarding generative adversarial networks which are used in our framework. Following that, section four describes the materials and the methodology we employed to design and implement our cultural image inpainting framework. In section five, we depict and discuss the experimental results of our framework on a large dataset of paintings. Finally, we draw our conclusions and give some perspectives of future work.

II. Related Work

Image inpainting is a set of techniques used in the computer vision field to complete missing areas in images and visual content. Many semantic image inpainting approaches have been proposed [4-11] . These techniques have to ensure that the generated content does not alter the

978-1-5386-7942-5/19/$31.00 ©2019 IEEE

original image context and that the result will look convincing for humans. Among the methods proposed, deep learning-based techniques demonstrated favorable performance.

There are mainly two categories of classical image inpainting techniques. In the first category, we can find techniques that try to generate textures based on the surrounding content to complete the missing region [6]. One of the most notable contributions in this category is *PatchMatch* which uses an approximation of the nearest neighbor to find the adequate patches from the incomplete image [6].

The second category of approaches are ones that leverage large visual databases. These approaches try to match the area surrounding the missing part with the database images assuming that a similar image is indexed in the database. Unfortunately, these kinds of techniques are prone to failure if an image having the same context as in the damaged sample is not indexed in the database. Also, the fact that these techniques rely on large databases is a major drawback [7].

Newer techniques based on deep learning were recently introduced. These techniques are based either on a certain type of convolutional context encoders (autoencoders) [11] or on generative adversarial networks which demonstrated very promising results [4, 5, 8, 12].

It is worth noting that the output of these methods is not always guaranteed to be visually convincing as sometimes these techniques either miss the context of the input or generate blurry images. Some approaches were proposed to mitigate these issues such as the use of Poisson blending and dilated convolutions [4, 8] . In our work, we mostly focus on approaches based on generative adversarial networks.

III. Generative Adversarial Networks

Generative Adversarial Networks (GANs) are a class of unsupervised machine learning models proposed by Goodfellow et al in [2]. These models combine two neural networks that are trained using the minimax concept inspired by game theory. The technique was validated to be one of the most effective at generating near realistic visual data when trained effectively.

The first neural network and the most important one is called the generator G. The other one is called the discriminator D. The generator G learns how to map data from latent space (generally a known probability distribution) to another desired output space (visual in our case). This unknown distribution is denoted p_{data} . The discriminator D, in contrast, is trained to differentiate between real samples and the output of the generator. The networks are trained with different objectives. On the one hand, the generator is trained to fool the discriminator network i.e. increase its failure rate to distinguish between generated and real data samples. On the other hand, the discriminator is trained to predict dataset sample are real and generators output as fake.

Deep convolutional GANs implement convolutional layers without max pooling or fully connected layers. These networks use convolutional strides and transposed convolutions for the downsampling and the upsampling. Other than that, DCGANs rely on the Leaky ReLU activation function instead of ReLU which is often used in CNNs [13].

A. GAN-based image generation

GANs primary function is to generate data samples that are similar to the data that was used to train them. In DCGANs, the generator network is a deconvolutional neural network that takes an input (z) sampled from a multi variate probability distribution. The network then transforms this input until it produces the visual output. Conversely, the discriminator network is a convolutional neural network that takes as input the visual output of the generator and then produces a score which is equivalent to real or fake [2, 5, 13]. Both networks are then connected in a DCGAN. The architecture of the generator part of the DCGAN we used within this paper is outlined in Figure 1.

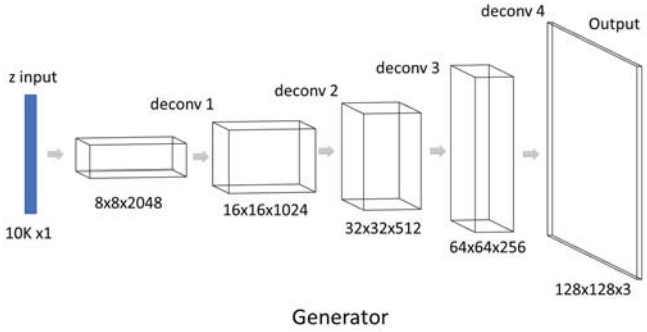


Figure 1. Generator architecture

The discriminator's architecture is nearly similar to the architecture of a normal convolutional neural network and is outlined in Figure 2.

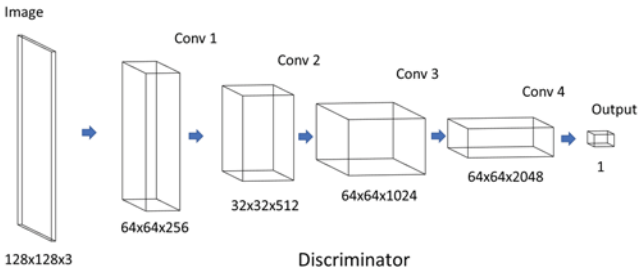


Figure 2. Discriminator architecture

The DCGAN architecture is outlined in Figure 3.

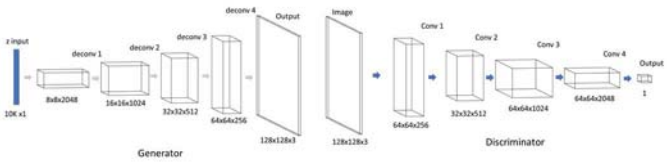


Figure 3 DCGAN architecture

Training the DCGAN consists in optimizing the following loss equation using the backpropagation algorithm.

$$\min_G \max_D V(G,D) = \mathbb{E}_{\mathbf{h}\sim p_{data}(\mathbf{h})}[\log(D(\mathbf{h}))] + \mathbb{E}_{\mathbf{z}\sim p_{\mathbf{Z}}(\mathbf{z})}[\log(1 - D(G(\mathbf{z}))], \quad (1)$$

The training is often reported to be long and difficult to monitor. For example, if the discriminator fails to detect a generated sample, the training process will be stuck.

GANs are not suited for image completion as their output has high chances to be unrelated to what we want to complete. In the following, we present a methodology used to constraint their output.

B. GAN-based image completion

Even though they are mainly developed to generate data, GANs can be used for semantic image inpainting tasks. The visual completion using GANs consists of constraining the output of the generator in order to generate an image that has the same visual characteristics as in the damaged one. The damaged area is then replaced with the associated area from the generated image. The challenge is how to constrain the output of the generator which takes random data as input. For this, the authors of [5] propose a method that combines a contextual loss in addition to the perceptual loss (evaluated by the discriminator). This loss combination is used to perform a backpropagation on the input of the generator (z vector). The goal of the backpropagation optimization is to lower as much as possible this combined loss. As the (z) vector is the only parameter controlling the output after training the DCGAN, the goal is to generate an output (image) that minimizes the loss combination, which will theoretically result in an image that looks similar to the one that we want to complete. In the following, we give details related to the losses evaluation and the process used to perform GAN-based semantic image inpainting [5, 13].

The problem of finding the (z) vector is then treated as an optimization problem that consists of minimizing this combined loss through a backpropagation on the input vector (z) of the generator. Regarding the contextual loss, the authors use the $l2$-Norm as a distance measure between the generated content and the existing content removing the missing areas in both images. The authors stress on the fact that this measure has to be weighted in order to ensure an effective training. The weighting consists of giving high importance to pixels close to the missing regions and less importance to pixels far from those regions.

Deep convolutional generative adversarial networks (DCGAN) demonstrated very good performance in reproducing visual data when trained on the same images that are visually from the same context. However, when we try to diversify the contexts of the training data, these data generation models become inefficient and their visual output will not be convincing even after a long and difficult-to-monitor training [2, 5].

IV. Methodology

In this section, we present the details related to the design and implementation of our cultural image completion framework. Our approach relies on deep convolutional adversarial networks trained to learn how to generate realistic looking artwork. These networks will then be used to perform semantic inpainting following a novel procedure to complete damaged artwork.

A. Data collection and preprocessing

In the following, we present the datasets used to validate our approach on cultural data. It is worth noting that we mostly used paintings for the evaluation of our framework as paintings were the biggest cultural type regarding data samples (more than 140 thousand). The samples collected from WikiArt were the most well-structured.

1) The WikiArt Dataset

Wikiart [14] is an online gallery that hosts thousands of artworks from very known artists across a very wide time span. The assets count is more than 140 thousand. The majority of artwork is available for download. To collect the data, we designed custom harvesting scripts based on the python library *beautifulsoup* to crawl through the Wikiart.org website and download all the data which mainly consists of visual captures associated with metadata (See Figure 4).

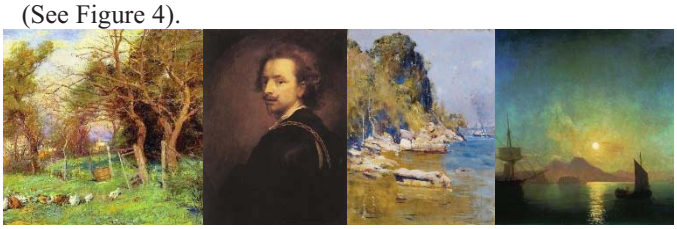


Figure 4. Some paintings from WikiArt

2) The Metropolitan Museum (The MET) Dataset

A dataset of more than 200 thousand artwork was published under the Creative Commons open access license by the Metropolitan Museum of Art, New York [15]. Similarly to WikiArt, the data is mostly images associated with metadata. Collecting the data was based on the CSV file provided by the museum and some custom-made Python scripts (See Figure 5).

Figure 5. Some artwork from the MET

3) The Rijksmuseum Dataset

The Rijksmuseum of Amsterdam, which is often referred to as the Rembrandt museum, is currently opening his collection through a public API [16]. This collection consists mainly of paintings, pottery, etc. The Rijksmuseum dataset is a collection of images associated with metadata. In this work, we mainly focused on paintings. (See Figure 6.).

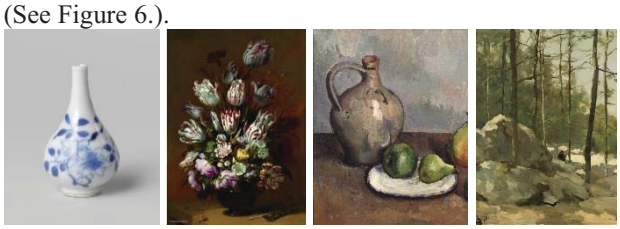


Figure 6. Some artwork from the Rijksmuseum

B. Solution Idea

Instead of relying on a single DCGAN for the completion, our approach uses the divide-and-conquer strategy to train several "*specialized GANs*". We split the task of completing several cultural categories with one GAN, to only a single category per GAN. The question now is how to split the training data across the categories efficiently. To address this matter, one can rely on the human pre-annotated labels, however, as can be seen in Figure 7, having two images from the same labeled category does not always mean they are going to be visually similar.

Figure 7. Paintings from the same annotated category "Baroque" that are not visually similar

To ensure an effective grouping of similarly looking images, our solution leverages global visual features to perform unsupervised clustering using K-means. The training scenario of our framework is as follows: we select the dataset of cultural artwork that we want to use for training. We then compute visual global features of each image using either CNN features, SIFT [17] or SURF [18] features with Bag of Visual Words, etc. Once computed, these global features are clustered using the K-means algorithm [19] with an estimated number of cultural categories as the number of centroids (K). Once all the images have been clustered, a DCGAN is trained for each cluster. The training principle is illustrated in Figure 8.

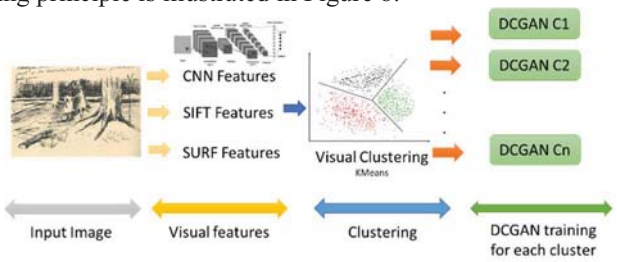


Figure 8. The training of the completion framework

The completion scenario of our framework is as follows. We take an incomplete image of cultural artwork, and based on what visual information is available, we select the best matching cluster as per the last step. Once selected, the generator associated with this cluster is used to generate samples following a semantically constrained generation. The quality of these samples is evaluated using two losses as in the technique proposed in [5].

C. Framework design

To address the visual data completion problem in the cultural context, we designed and implemented a hierarchical framework that combines visual clustering and multiple DCGANs to efficiently and effectively perform an accurate visual completion. Through our analysis, we found that it is rather ineffective to train a single DCGAN for data that has multiple visual contexts.

The training pipeline of our framework has mainly two stages. In the first stage, the training data is clustered using visual features in order to ensure similar visual context in each visual cluster. Afterwards, for each cluster, we train an image generation DCGAN using the same architecture discussed in section 3. For completion, we first try to identify the closest cluster to the image we want to complete. Afterwards, we follow the same semantic inpainting strategy discussed using the DCGAN associated with the cluster. In the following, we give some technical details related to the visual data clustering strategy used in our framework in addition to DCGAN-based image completion.

1) Visual cultural data clustering

Clustering visual data is a critical step for our framework. The results of this step will directly affect the ability of the specialized GANs to generate good looking samples for each category. Our clustering methodology uses a class of global visual features to represent the image with a higher dimensional vector that encodes a special representation of the image in the targeted feature space. The global features are then clustered using the K-means algorithms with an estimation of the visual categories as the number of centroids (K). We use 3 types of known global visual features mainly to evaluate their performance in our context for comparison purposes.

a) CNN features

CNN features are one of the most effective ways to represent the global features of an image to the superior performance of CNNs in image classification tasks in comparison with handcrafted features. To extract these features, we use CNNs pre-trained on the ImageNet challenge. The output of their last convolutional layer was select as the global features array (See Figure 9). In our experiments, we use the following CNNs: VGG16 [20], VGG19 [20], and ResNet50 [21]. Figure 9 shows the features layer used in the VGG16 network.

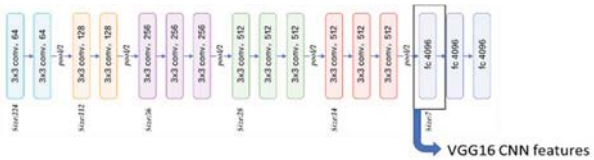


Figure 9. VGG16 CNN features

b) SIFT and SURF Features

SIFT and SURF are patented feature description algorithms. Both algorithms are used for image recognition and classification tasks. SIFT and SURF detect and compute features in images, however, these features are local. To compute global features using these techniques, we use the Bag of Visual Words technique to compute the dictionary of visual words of our training dataset. For each image, we computed the visual words histogram which we considered as the global features of the image.

2) Specilized cultural DCGAN

After the creation of visual clusters from the training data, we must ensure an even distribution of images across clusters. For each cluster, we train a DCGAN following the architecture outlined in Figure 3. Combined, these DCGANs have the ability to generate images similar to the context of the cluster used in their training.

3) Visual cultural data inpainting

For visual data inpainting, in contrast with the semantic inpainting approach presented in [5] our approach leverages the trained clustering model to perform a more accurate inpainting. In fact, in our method, we try to find the closest cluster to the incomplete image, and then assign the completion to the DCGAN that was trained on this cluster using the GAN-based semantic image inpainting approach discussed in section 3. Figure 10 summarizes the completion stage of our framework.

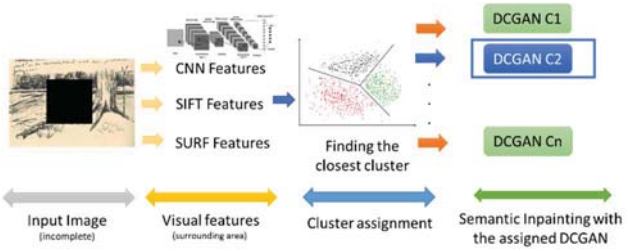


Figure 10. The completion stage of our framework

V. Experimental Results and Discussion

A. Experimental setup

Our approach was implemented with the Python programming language (Version 3.6.3) using the Tensorflow deep learning library (1.9.0) [22]. For experimental tests, we used a machine running the Ubuntu operating system (16.04 LTS) with an Intel Core i5-7600K CPU, 16 GB of RAM and an Nvidia Titan Xp GPU.

Our framework was evaluated against a general DCGAN which was trained on the whole selected dataset. This DCGAN has the same architecture as the one outlined in Figure 3. Table 1 outlines the training and completion hyperparameters that we used to train and validate our approach. It's worth noting that these hyperparameters were also used with the general DCGAN that was trained with the whole dataset.

Train. LR	Z dim	Input size	Epochs	Optimizer	Completion iterations
0.001	10K	128×128	100	Adam	40K

Table 1. Training and completion hyperparameters

B. Experimental results and Discussions

1) Results

The main evaluation criterion is the perceptual quality of the completion. Regarding the specialized GANs, we found that the visual features used for clustering that resulted in the best perceptual quality were CNN features based on the VGG16 CNN. Table 2 outlines some of the visual artwork that was completed with our framework (with clustering) compared to a general DCGAN that was trained with the whole dataset.

One can clearly observe that the completion based on the specialized GANs is more accurate in filling (completing) the missing regions compared to the general GAN which was trained with the whole dataset.

2) Discussion

Through this work, our objective was to design and implement a visual inpainting (completion) framework that can accurately complete missing visual cultural data from multiple categories. Our framework design relies on generative adversarial networks which are used to generate visual content. Since these networks cannot be used directly for inpainting tasks, we used a semantic image inpainting approach based on DCGANs. Following our analysis, we found that training and using a single DCGAN for completion is ineffective especially if the training dataset has data from different visual contexts. As a result, we designed an inpainting framework that improves the visual quality of GAN-based semantic inpainting using a divide-and-conquer strategy. Instead of training a single GAN, we used clustering with the K-means algorithm to categories the training data into similarly looking clusters. For each of these clusters, a GAN is trained. At completion, we rely on the same clustering model to find the closest cluster and perform GAN based semantic inpainting with the GAN that was trained on this cluster. By doing so, we have significantly restricted the visual output context of the GANs. The impact of adding clustering can be easily perceived on the examples of table 2. Our framework still has some room for improvements such as using a simpler architecture for the specialized GANs which will increase the efficiency of our approach.

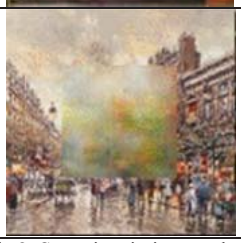
General DCGAN	Specialized GANs + clustering
	
	
	
	
	

Table 2. Some inpainting results

VI. Conclusion

In this paper, we presented an image inpainting framework adapted for visual cultural heritage. Our framework relies on generative adversarial networks which are nowadays considered among the most powerful generative models. These models can be used to perform semantic image inpainting. However, through our analysis, we saw that using a single model with a dataset that has several visual contexts is ineffective. We designed an image inpainting framework, which was validated on cultural data, that can

effectively perform visual completion of different contexts. We relied on the divide-and-conquer strategy and instead of training a single DCGAN, we clustered our training data using the K-means algorithms and trained a DCGAN for each cluster. The resulting clusters have mostly the same visual context which in fact resulted in a better-quality completion. As future work, we plan to optimize the training process by using a simpler architecture for the specialized GANs. We also aim to explore the effect of varying the number of clusters on the performance of the framework, as well as on the quality of the produced images.

ACKNOWLEDGMENT

This publication was made possible by NPRP grant 9-181-1-036 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors.

REFERENCES

[1] A. Belhi, A. Bouras, and S. Foufou, "Towards a hierarchical multitask classification framework for cultural heritage," in *2018 IEEE/ACS 15th International Conference on Computer Systems and Applications (AICCSA)*, 2018, pp. 1-7: IEEE.

[2] I. Goodfellow *et al.*, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672-2680.

[3] C. Ledig *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," *arXiv preprint,* 2017.

[4] S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Globally and locally consistent image completion," *ACM Transactions on Graphics (TOG),* vol. 36, no. 4, p. 107, 2017.

[5] R. A. Yeh, C. Chen, T. Yian Lim, A. G. Schwing, M. Hasegawa-Johnson, and M. N. Do, "Semantic image inpainting with deep generative models," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 5485-5493.

[6] C. Barnes, E. Shechtman, A. Finkelstein, and D. B. Goldman, "PatchMatch: A randomized correspondence algorithm for structural image editing," *ACM Transactions on Graphics (ToG),* vol. 28, no. 3, p. 24, 2009.

[7] J. Hays and A. A. Efros, "Scene completion using millions of photographs," in *ACM Transactions on Graphics (TOG)*, 2007, vol. 26, no. 3, p. 4: ACM.

[8] C. Huang and K. Yoshida, "Evaluations of Image Completion Algorithms: Exemplar-Based Inpainting vs. Deep Convolutional GAN."

[9] A. Fawzi, H. Samulowitz, D. Turaga, and P. Frossard, "Image inpainting through neural networks hallucinations," in *Image, Video, and Multidimensional Signal Processing Workshop (IVMSP), 2016 IEEE 12th*, 2016, pp. 1-5: Ieee.

[10] A. Criminisi, P. Pérez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Transactions on image processing,* vol. 13, no. 9, pp. 1200-1212, 2004.

[11] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2536-2544.

[12] U. Demir and G. Unal, "Patch-Based Image Inpainting with Generative Adversarial Networks," *arXiv preprint arXiv:1803.07422,* 2018.

[13] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434,* 2015.

[14] WikiArt.org. (31-01-2019). *WikiArt.org - Visual Art Encyclopedia.* Available: https://www.wikiart.org/

[15] T. MET. (2019, 31-01-2019). *The Metropolitan Museum of Art.* Available: https://www.metmuseum.org/

[16] T. Mensink and J. Van Gemert, "The rijksmuseum challenge: Museum-centered visual recognition," in *Proceedings of International Conference on Multimedia Retrieval*, 2014, p. 451.

[17] D. G. Lowe, "Object recognition from local scale-invariant features," in *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, 1999, vol. 2, pp. 1150-1157: Ieee.

[18] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *European conference on computer vision*, 2006, pp. 404-417: Springer.

[19] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A k-means clustering algorithm," *Journal of the Royal Statistical Society. Series C (Applied Statistics),* vol. 28, no. 1, pp. 100-108, 1979.

[20] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556,* 2014.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770-778.

[22] M. Abadi *et al.*, "Tensorflow: a system for large-scale machine learning," in *OSDI*, 2016, vol. 16, pp. 265-283.