RestoreNet

Quantifying Image Restoration of World War II Photos

Emmanuel Diaz

Abstract

In this project, we look at an approach towards restoring degraded images from the World War II era. These target photos have certain artifacts like “photo creasing” that develop due to natural factors in handling photos and our goal is to devise a method that attempts to restore these images back to their original form. We collected a dataset of World War II images and selected degraded images to analyse features of the degradation process, like Signal-to-Noise Ratio and Structural Similarity Index, in order to further understand reversing this process. Using a Generative Adversarial Network called “Deep Generative Prior”, we show how an inpainting restoration is sensitive to the training data and observe the difficulties of proving a ‘good’ restoration both subjectively and quantitatively.

Introduction

The field of digital image restoration has been developed for more than 30 years with its foundations in signal processing techniques. These techniques helped remove noise from ‘degraded’ images which are images that have some form of reduced quality from their original version. However, these image processing techniques have their bounds in how much lost information from degradation may be recovered. The recent advancements of neural networks and collections of huge image datasets have led many researchers to believe that neural networks hold some potential in proper image restoration.

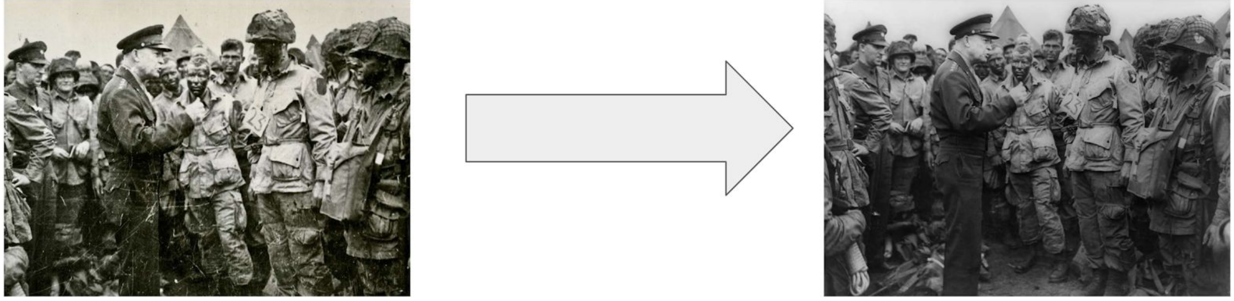
Previous Work

Researchers began looking at different types of neural network architectures to find the best in restoring images. It began with the use of Recurrent Neural Networks (RNNs)[1] which takes the degraded image as input and performs a feedback loop, further restoring the image in each network pass. This method has shown perceptually good results on small dimensional test images, however RNNs suffered from inherent biases of the initial conditions, causing this method not to generalize well beyond a few examples. The next network challenger up to the task was the Convolutional Neural Network (CNN) Autoencoder[2] which use CNNs as feature extractors and encode the convolutions to a small dimension latent variables then deconvolutional layers learns the upscale process for these variables, giving a ‘cleaned’ image. CNN Autoencoder proved exceptional perceptive results for denoising and inpainting but its complexity and optimization left researchers to keep exploring new methods.

A breakthrough in image restoration came through with the application of Generative Adversarial Networks (GANs)[3] which gave rise to many facets in image processing and computer vision like DeepFakes[4]. GANs are constructed using two networks, the Generator and Discriminator, where the Generator creates constructed samples from the distribution of the training set and the Discriminator classifies these generated samples as being ‘real (could be from the training set)’ or ‘fake’ (not likely in training set). These two networks have contrasting objectives as the Generator wants to create ‘realistic’ samples to fool the Discriminator, and the Discriminator wants to detect as many ‘fake’ samples that the Generator creates. This adversarial battle leads each network to achieve better loss each iteration and once certain epochs are reached, the training is terminated and the Generator is kept to create similar examples from the training set.

Problem

It is widely known that old photos converted from film to digital might have experienced some degradation in the process. World War II photos are no exception and properly restoring these images will benefit us in preserving historical documents. We will investigate if using GANs to approach the ‘image inpainting’ problem, where sections of the image are obstructed and we have to ‘fill in the blanks’, will give us not only a perceptual improvement in quality but also a quantification of how well the restoration occurred.



Data Description

To approach this problem, we start by investigating the population of World War II images. The data collected includes ~3,000 digital images of events from World War II scraped from ww2db.com[5]. The images were separated between three categories: Pre-War, Mid-War, and Late-War.

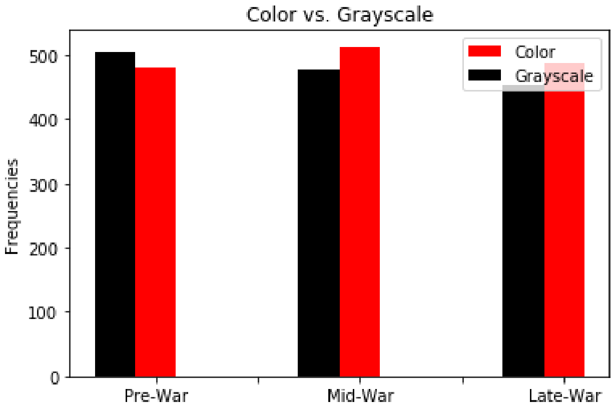


Fig (1)

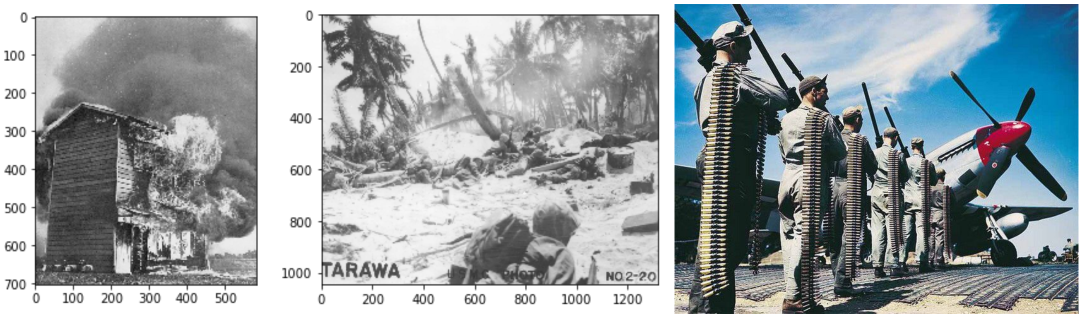


Fig (2)-(4)

Similar amounts of color and grayscale images were collected for each time period in order to have enough data to represent both subsets of the population in further analysis/training. Fig (2)-(4) are sample images from the collected image dataset, respectively from the three chronological time periods. The population we are seeking to refine and learn is the collection of World War II photos and by analysing a sample of these photos, we hope to gain insight behind the degradation process.

In finding similarities between images, I used a blob detector to classify similar images based on number of invariant points and amount of clustering between points. This choice of feature was decided due to the selective inpainting that occurs later, where we want the GAN to learn the restoration among selected regions in the image. Finding similar images based on similar invariant points may give the network the ability to learn among ‘cluttered’ versus ‘spacious’ images. The Difference of Gaussian blob detector was applied to each image and clustered using KMeans.

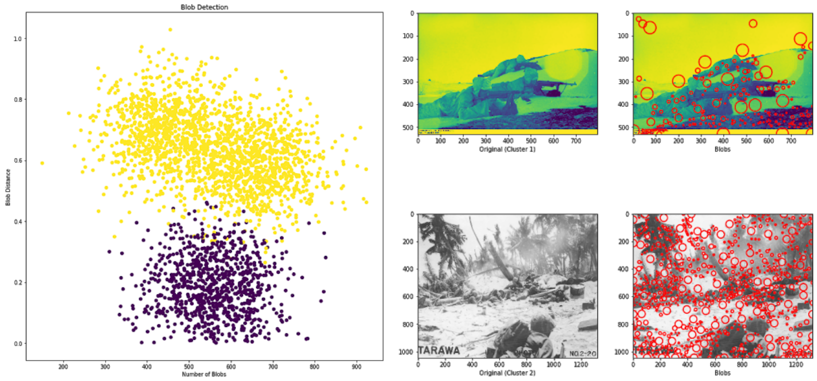


Fig (5)-(6)

Two clusters were identified in this feature space, with a rough ratio of 60:40 between cluster populations. We want to see if using a smaller subset of our sample to learn features will have greater limitations on our restoration than using the entire sample. We will see later that our model for performing restoration uses a supplemental type of learning known as ‘prior’, which can allow the network to learn beyond the data given. If our results show that combining smaller domain dataset sizes with this technique creates approximate or improved results compared to using the entire dataset, we could show that domains with smaller data populations could have practical restorations.

Methods

Quantifiable Metric

To start measuring how much ‘better’ we are getting upon restoration stages, we can quantify this with a measure, namely a combination of Peak Signal-to-Noise Ratio and Structural Similarity Index (See Appendix for definitions).

For iteration t, and chosen iteration ε

Maximize $PSNR + \frac{max(0,t-\varepsilon)*SSIM}{t-\varepsilon}$ over degraded region

We can use PSNR for early iterations for its approximation qualities then adding SSIM for later iterations when PSNR may taper off, for a finer measure of image quality near the end.

Choosing the model

In approaching a GAN architecture, we chose “Deep Generative Prior”[6] which is based on the BigGAN[7] layers/data-loaders as well as “Deep Image Prior”[8], an approach for image restoration learned from parameterizing the network based on the image rather than learning the parameters of the images on the network. Through the use of learning the ‘prior’, we will observe how this model performs using our limited domain dataset size.

Pre-Processing

Given a target image that is of size $h \times w$, it is loaded into the model where we apply a degradation transformation. To perform this, we use the previously defined binary mask $Mask(x, y)$ and receive the output $Target'(x, y) = Target(x, y) \cdot Mask(x, y)$. For the block inpainting, we crop a square in the center of the image of sides $min(h, w)$. For the selective mask inpainting, the mask applied is chosen based on the region to be restored (the creases). The selected regions of the image are where the network is designated to inpaint.

Training and parameter selection

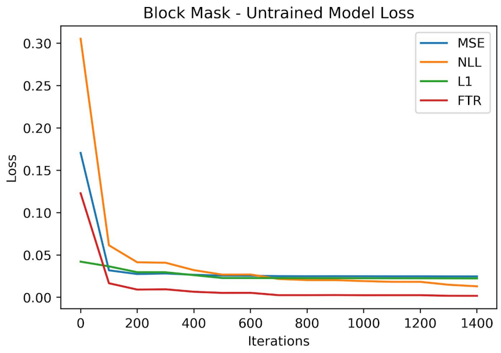
A generator is trained for each input image on the cluster closest to the input image in the blob detector. For block and selective inpainting, we have resolution=256x256 with 2000 iterations at 5 stages. After hyperparameter choosing, we choose

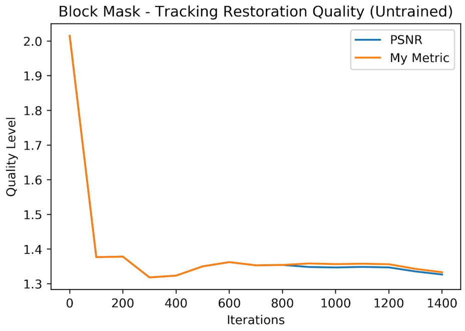
Parameters	Stage 1	Stage 2	Stage 3	Stage 4	Stage 5
Num Features	10	8	8	8	8
Learning Rate	1	1	0.5	0.2	0.1
MSE Weights	1	1	1	1	10
Generator Learning Rate	5e-5	5e-5	1e-5	1e-5	1e-6
Latent Z Learning Rate	2e-3	1e-3	2e-4	2e-5	1e-5
Optimizer	Adam	Adam	Adam	Adam	Adam

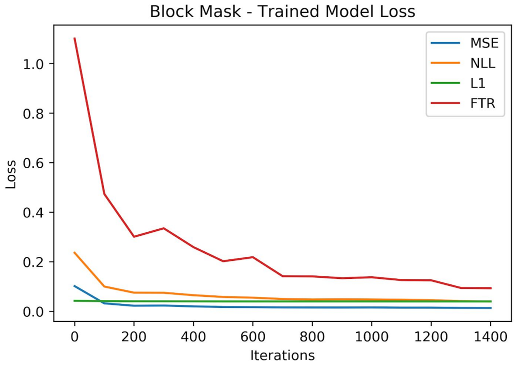
Results

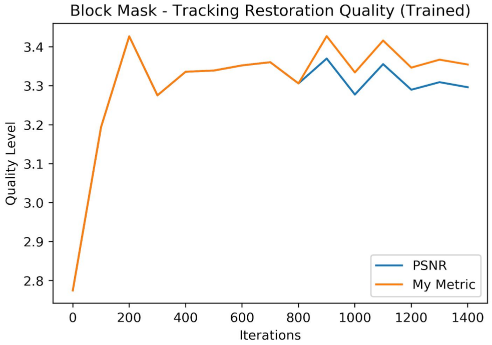
After implementing the model and performing trained/untrained variants, the following results were found by looking at the losses from the model as well as the output restorations. Three inpainting types were used to create these results: Block, Large Selective, and Fine Selective. Block inpaints a center square crop of the image, Large Selective is a custom mask with large inpainting area, and Fine Selective is a custom mask with smaller inpainting area.

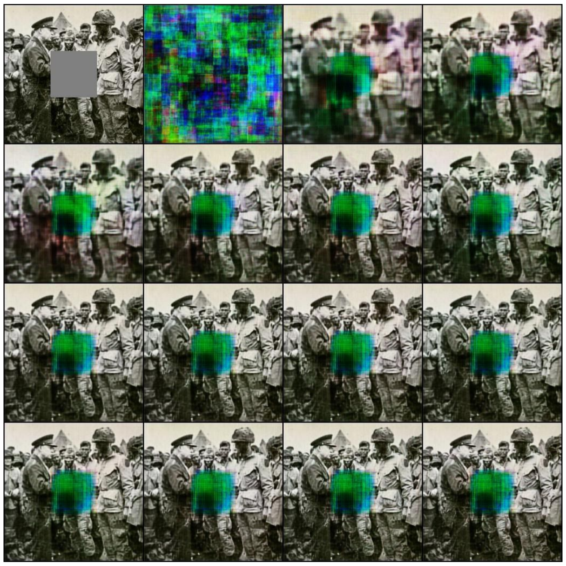
Block Inpainting Model











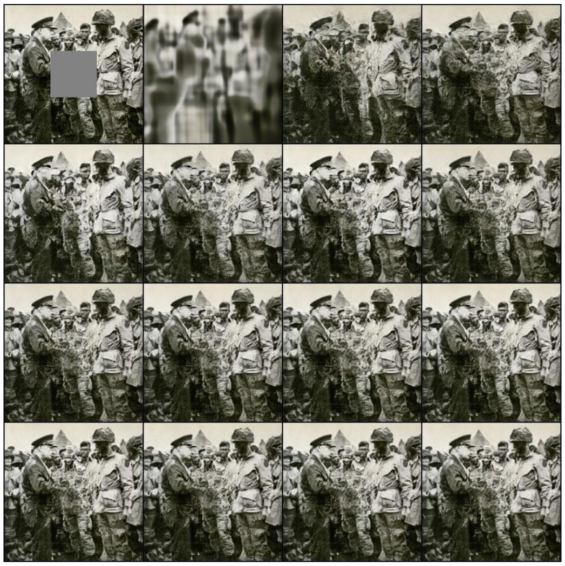
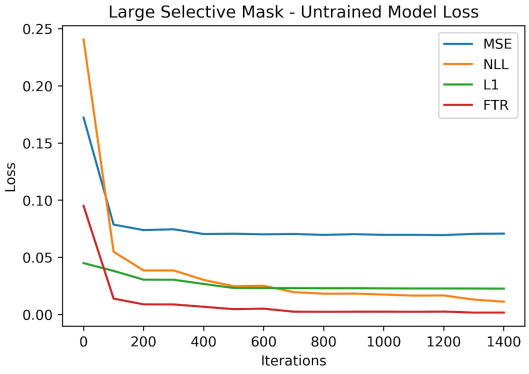
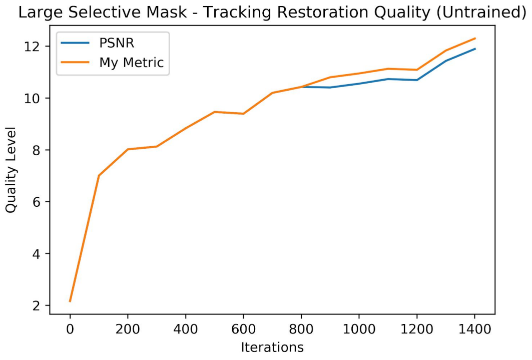
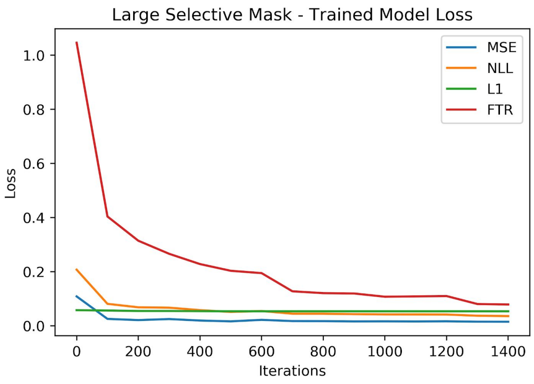


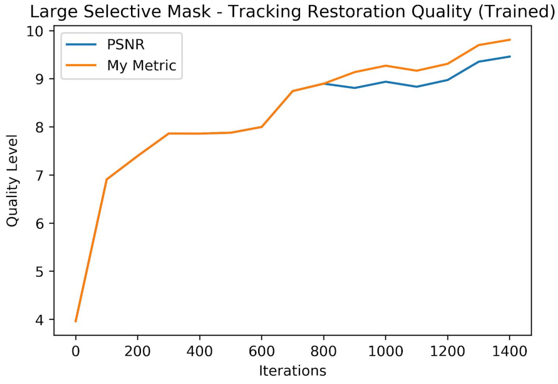
Fig (7): Random Gen - Metric: +6.9027 Fig (8): Trained Gen - Metric: +11.59994

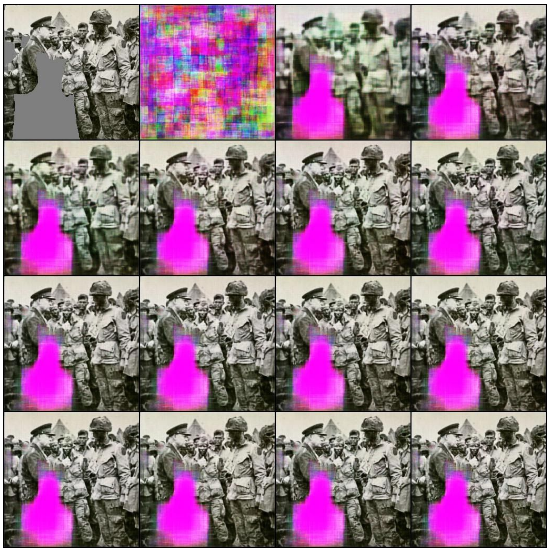
Large Selective Mask Model


Large Selective Mask - Untrained Model Loss


Large Selective Mask - Tracking Restoration Quality (Untrained)


Large Selective Mask - Trained Model Loss


Large Selective Mask - Tracking Restoration Quality (Trained)



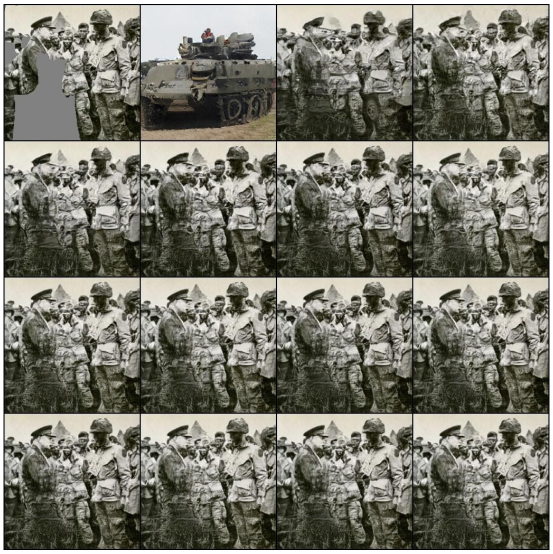
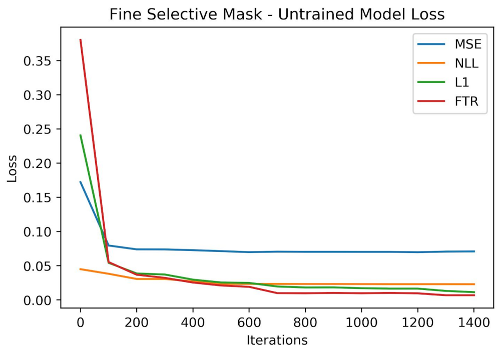
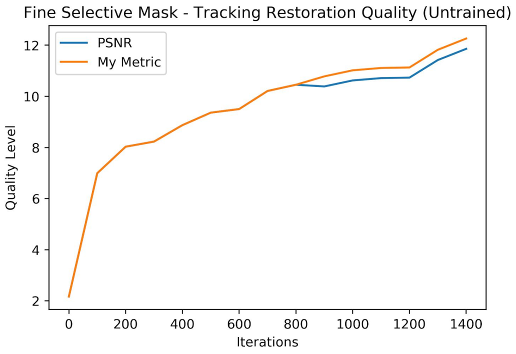
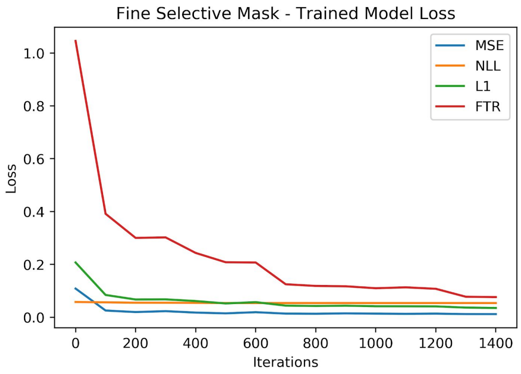


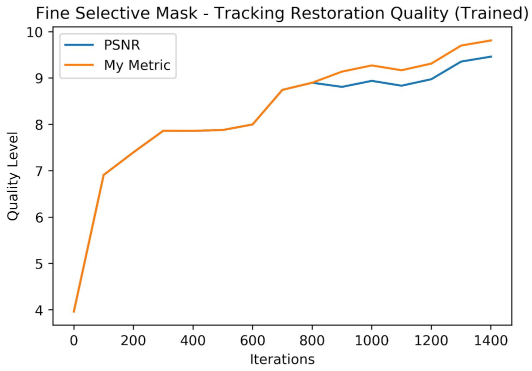
Fig (9): Random Gen - Metric: +4.3359 Fig (10): Trained Gen - Metric: +12.2342

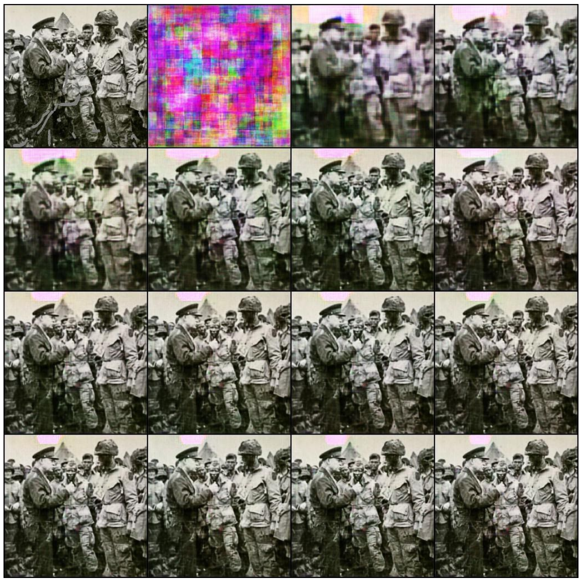
Fine Selective Untrained Model











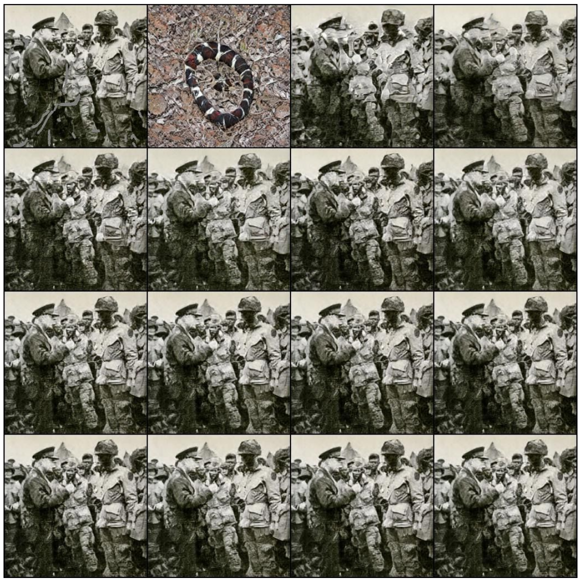


Fig (11): Random Gen - Metric: +9.3471

Fig (12): Trained Gen - Metric: +16.2342

Discussion

The results seen so far show some promise in using the trained weights over the 'random generator' method. As we see in the first iterations between figure (7) and (8), figure (7) starts with a random image prior and does not converge to a semantically improved restoration as it had no prior knowledge of how to generate the occluded region. In figure (8), we see that the first iteration has aspects of an image learned before and this converges towards a visually better restoration. The calculated metric clearly shows the improvement in restoration in this respect.

Looking at the large selective results where masking is a large area masking is applied over known degraded regions, we find the results to be slightly similar. Between figure (9) and (10), it is clear that random generators do not produce any valuable restoration in the inpainting regions. Figure (10) shows more localized improvement on the creases than figure (8).

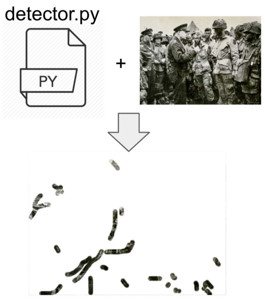
The fine selective regions prove better results from all the trained weight inpaintings. Figure (12) has finer crease restoration that matches the surrounding scene compared to figure (10) which has a non-smooth inpainted region.

In previous work[6], DGP has shown to provide great results for large missing regions due to its intuition of spatial coherence. In figure (8), we see perceptually acceptable estimates of the original image, even cleaner than figure (10). Since the block inpainting occurs all in one localized area of the image, the network has a better sense of what to fill in into that region rather than the large selective area, where the regions are not square nor properly localized. However, it is clear that the fine selective masking proves the best results, but manually detecting these degraded regions is inefficient compared to the masking method used for block inpainting. Finding the best way to select the degraded regions in the image will allow for better restorations.

In practice, we should combine the use of masking with respect to spatial coherence while performing it efficiently, that is selecting larger regions of the image where degradation is known to occur then performing a proper masking like block inpainting in those regions. Through these results, the pipeline for restoring images can be streamlined into the network.

The limitations seen in this approach mainly regard the amount of data collected. Even with three thousand photographs, the network has a difficult time getting good context on the data population. Scaling the dataset to a higher representative quantity would give the network more information on the spatial regions in the image.

In the future, we would like to extend this investigation into different restoration objectives like super-resolution, denoising and colorization. These different restoration tasks would create an even more general purpose restoration method for World War II images. We would also like to create an automated selective mask creator that can detect the creases and segment them to make a mask for the selective mask GAN process. This would speed up the process of inputting images into the network for restoration.



References

[1] *On learning optimized reaction diffusion processes for effective image restoration*, Y Chen, W Yu, T Pock, 2016 https://arxiv.org/pdf/1503.05768v2.pdf

[2] *Image Restoration Using Convolutional Auto-encoders with Symmetric Skip Connections,* Xiao-Jiao Mao and Chunhua Shen and Yu-Bin Yang, 2016, https://arxiv.org/abs/1606.08921

[3] Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A.C., & Bengio, Y. (2014). Generative Adversarial Networks. *ArXiv, abs/1406.2661*.

[4] Nguyen, T.T., Nguyen, C.M., Nguyen, D.T., Nguyen, D.T., & Nahavandi, S. (2019). Deep Learning for Deepfakes Creation and Detection. *ArXiv, abs/1909.11573*.

[5]Chen, C. P. (n.d.). *World War II Database*. Lava Development LLC, Retrieved from https://ww2db.com/photo.php

[6]*Exploiting Deep Generative Prior for Versatile Image Restoration and Manipulation*, Pan, Xingang and Zhan, Xiaohang and Dai, Bo and Lin, Dahua and Loy, Chen Change and Luo, Ping, 2020 https://arxiv.org/pdf/2003.13659.pdf

[7] BigGAN - https://github.com/ajbrock/BigGAN-PyTorch

[8]Ulyanov, D., Vedaldi, A., & Lempitsky, V.S. (2018). *Deep Image Prior*. 2018

IEEE/CVF Conference on Computer Vision and Pattern Recognition, 9446-9454.

Appendix

Peak Signal-to-Noise Ratio:

A measure of reconstruction quality, typically used in lossy compression applications. It is defined as a ratio between the maximum signal power and the amount of noise in the image.

Noise is described as the Mean Squared Error of between the original and an approximation (the reconstructed image). Given an original image I and reconstruction J, we get the MSE as

$$MSE = \frac{1}{3 \cdot m \cdot n} \sum_{x=0}^{m} \sum_{y=0}^{n} \sum_{c=0}^{2} [I(x, y, c) - J(x, y, c)]^2$$

and the maximum signal power as

$$MAX = 2^8 - 1 = 255$$

because there are 8 bits per sample. Thus our PSNR is

$$PSNR = 20 \cdot log_{10}(\frac{MAX}{\sqrt{MSE}})$$

We want our PSNR to be as high as possible, that means we want to decrease our MSE loss as much as possible.

Structural Similarity Index:

A full reference metric that predicts perceived quality, this metric takes two windows between the reconstructed image and degraded image and compares based on 3 qualities using windows x and y in each respective image

- Luminance is described between two windows as $l(x, y) = \frac{2\mu_x\mu_y + c_1}{\mu_x{}^2 + \mu_y{}^2 + c_1}$
- Contrast is describe as $c(x, y) = \frac{2\sigma_x\sigma_y + c_2}{\sigma_x{}^2 + \sigma_y{}^2 + c_2}$
- Structure is described as $s(x, y) = \frac{\sigma_{xy} + c_3}{\sigma_x\sigma_y + c_3}$

where μ_x and μ_y are the mean of the respective windows, $c_1 = (0.01 \cdot 255)^2$, $c_2 = (0.03 \cdot 255)^2$ and $c_3 = \frac{c_2}{2}$ and σ_x, σ_y are the standard deviations of the respective windows.

Our SSIM is calculated as $SSIM = l(x, y) \cdot c(x, y) \cdot s(x, y)$