Undergraduate Research Opportunities Program Report

Deep Face Analysis

Submitted by

Zhecan Wang

Franklin. W. Olin College of Engineering

Supervised by

Prof. Ashraf Kassim

National University of Singapore

May 5th 2017

ACKNOWLEDGEMENT

Foremost, I would like to express my deepest gratitude to my supervisor and mentor, Professor Ashraf Kassim and Dr. Shengtao Xiao for their selfless support and mentoring along this arduous but extraordinary and rewarding journey, which would definitely become one of my precious memory for lifelong benefits. During this semester project, Dr. Shengtao had profoundly inspired my way of thinking and behaving. I am really grateful to Dr. Shengtao Xiao for his painstaking efforts in training me for the best aiming to become not only an excellent student but also a qualified researcher, building my strong confidence and solid technical competencies for taking more challenging jobs. Last but not least, I would also like to express my gratefulness to my home school, Olin College of Engineering and Department of ECE at National University of Singapore, for providing me with this great exchange opportunity to undertake my research module in the world-leading university.

CONTENTS

ABSTRACT

This project mainly focuses on deep face analysis which includes two parts: photo realistic face texture generation and face bounding box refinement. Photo realistic face texture generation uses neural style transfer to transfer realistic detailed features from real face to synthesized face by using deep convolutional neural network. The research mainly involves modifying the structure of deep learning networks to leverage between performance and efficiency. This research also includes experimenting with different activation functions, cost functions, layer structures and preprocessing steps; Bounding box refinement functions for face recognition, a very limiting factor for face detection. Improving bounding box refinement involves experimenting different preprocessing methods like augmentation and deep learning network structures. Some other important approaches also include experimenting with different optimization methods, dynamic learning rates and etc.

1. INTRODUCTION

Neural style transfer has been developed quickly recently but mostly in general picture style transfer. It hasn't been explored much on human face style transfer yet. Furthermore, the speed of neural style transfer is also a very important limitation. The transfer generally takes too much time to conduct in live. We explore different deep learning network structures and their effects on human face style transfer trying to achieve a better tradeoff between the accuracy and speed. Most current face detection algorithms like Haar Cascades Face Detection are very limited by face recognition methods such as face bounding box refinement. The inaccuracy of bounding box could cause the search region off target. Throughout the experiments, we realize the different poses profoundly affect the results of bounding box prediction. Locating a face is very easy but the main difficulty is to perfectly extract the face from noised background while leaving enough space around the target so the face detection algorithm could detect the facial landmarks accurately.

2. PHOTO REALISTIC FACE TEXTURE GENEARATION

2.1. General Neural Style Transfer

The general neural style transfer transfers the style of one image to another while maintaining the content of the second image. There are two major challenges in this process. First, extracting image representations to well represent both the semantic style source and content source is very difficult. Second, the transfer, combination part is also crucial and challenging.

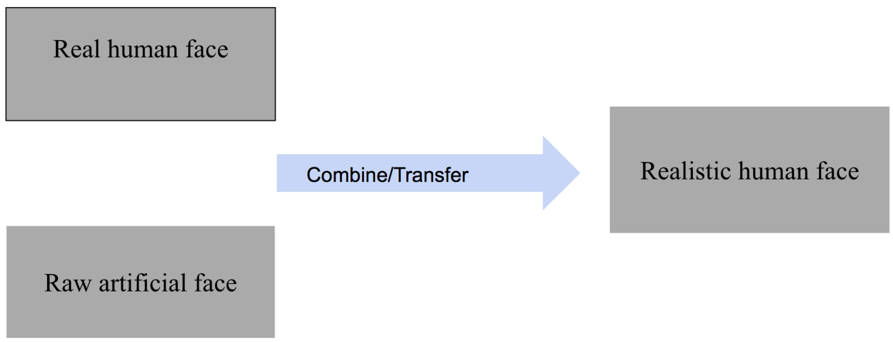


Fig 1. Systematic Diagram for Neural Style Face Transfer

2.1.1. Extraction, Representation

Image representations are extracted by convolutional neural network in higher dimensioned layers. Feature maps extracted from images are used as image representations for both style and content sources. For instance, as in the structure map, Fig 3, the network may use layer conv2_2 extracted from the content source as content representation and conv1_1, con2_1, conv3_1, conv4_1 and conv5_1 layers extracted from the style source as style representation.

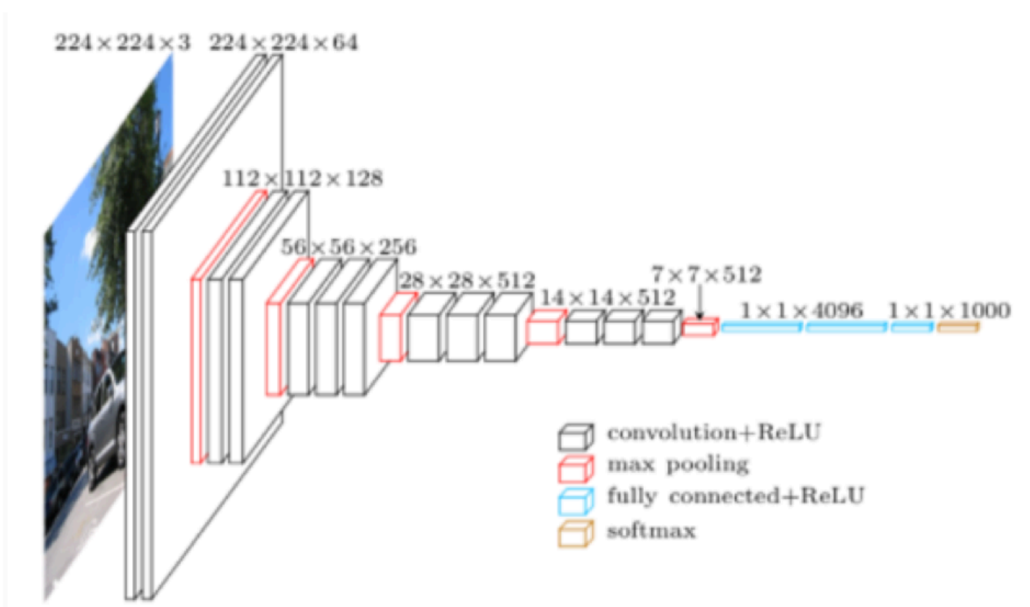


Fig 2. Structure of VGG 16 Network

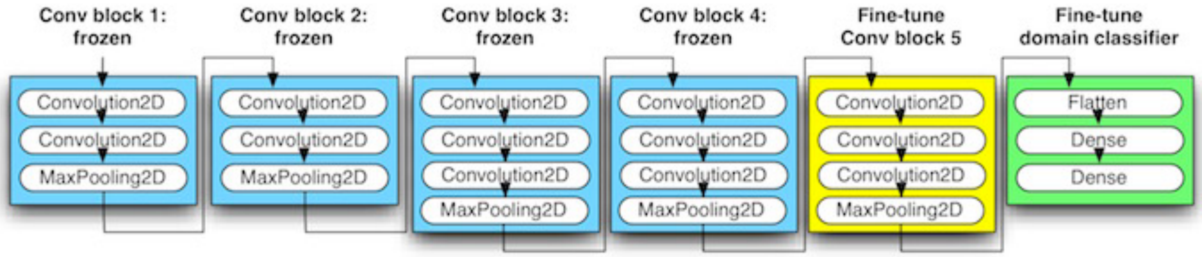


Fig 3. Structure of VGG 16 Network

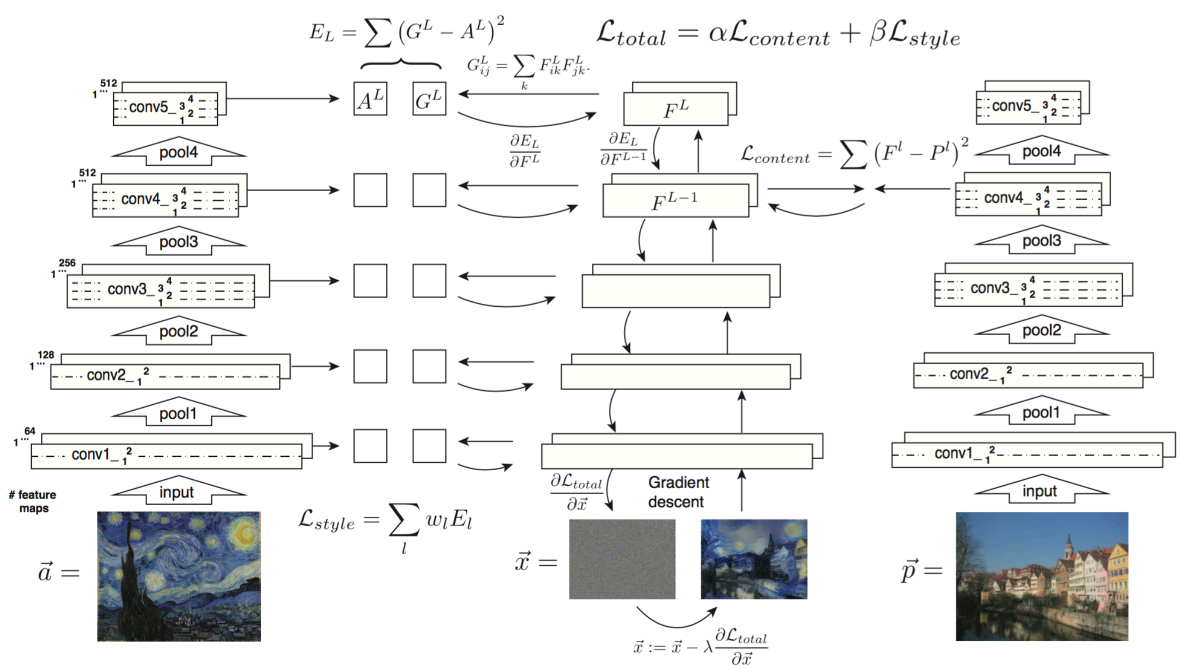


Fig 4. Overview Structure of Neural Style Transfer

2.1.2. Combination, Transfer

The transfer could use two inputs to initialize, white noise image or content source. In our results shown in Fig 16, the difference between these two results from different initialization methods is very minimal. After initialization, the algorithm uses both content loss and style loss to conduct standard backpropagation to readjust weights in order to adjust the input images in a pixel-wise matter. The adjustments would try to reach a point with global minimum total loss.

2.1.2.1. Content Loss

Let $\vec{p}$ and $\vec{x}$ be the original image and the image that is generated, and P^l and F^l their respective feature representation in layer l. We then define the square-error loss between the two feature representations

$$L_{content}(\vec{p},\vec{x},l) = \frac{1}{2}\sum_{i,j}\left(F_{ij}^{l} - P_{ij}^{l}\right)^2 \quad (1)$$

2.1.2.2. Style Loss

The author of [1] uses a feature space designed to capture texture information, the correlations between the different filter responses. These feature correlations are given by the Gram matrix $G^{l} \epsilon\ R^{N_l \times N_l}$, where G_{ij}^{l} is the inner product between the vectorised feature maps i and j in layer l:

$$G_{ij}^{l} = \sum_{k} F_{ik}^{l} F_{jk}^{l} \quad (2)$$

By including the feature correlations of multiple layers, we obtain a stationary, multi-scale representation of the input image, which captures its texture information but not the global arrangement.

Let a and x be the original image and image that is generated, and A^{l} and G^{l} their respective style representation in layer l. The contribution of layer l to the total loss is then:

$$E_l = \frac{1}{4N_l^2 M_l^2}\sum_{ij}\left(G_{ij}^{l} - A_{ij}^{l}\right)^2 \quad (3)$$

$$L_{style}(\vec{a},\vec{x}) = \sum_{l=0}^{L} w_l E_l \quad (4)$$

where w_l are weighting factors of the contribution of each layer to the total loss.

2.1.2.3. Total Loss

Total loss is a combination between the content loss and the style loss.

$$L_{total}(\vec{p},\vec{a},\vec{x}) = \alpha L_{content}(\vec{p},\vec{x}) + \beta L_{style}(\vec{a},\vec{x}) \quad (5)$$

Where α and β are the weighting factors for content and style reconstruction, respectively.

2.2. Experiments

2.2.1. Sample for Experiments

2.2.1.1. General Picture

2.2.1.1.1. Content Source Picture

Fig 5. Black and White Young Man   Fig 6. Neckarfront in Tuebingen

Fig 7. Window

2.2.1.1.2. Style Source Picture

Fig 8. Starry Night

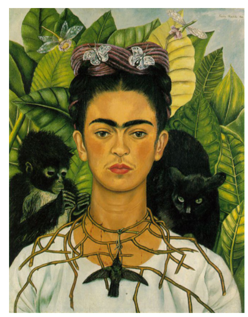


Fig 9. Frida Kahlo

Fig 10. Composition VII by Wassily Kandinsky

Fig 11. Jesuiten III by Lyonel Feininger

2.2.1.2. Human Face

2.2.1.2.1. Content Source Picture

Fig 12. Synthesized Face

2.2.1.2.2. Style Source Picture

Fig 13. Real Face 1

Fig 14. Real Face 2

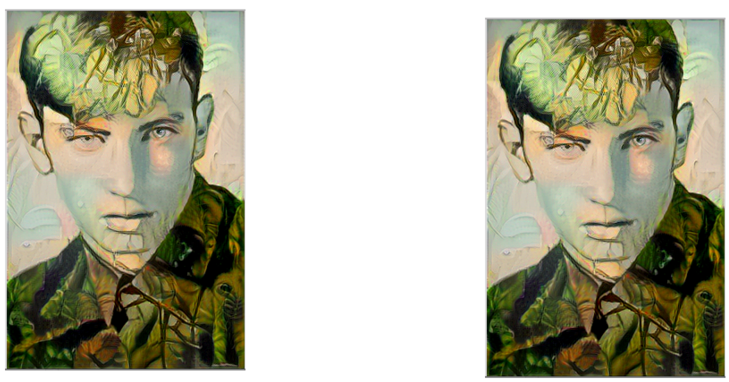
2.2.2. Neural Style Transfer Sanity Testing

A very important sanity and logical checking suggested by the author, Ph. D student, Shunsuke Saito [2] through email conversation is to check the result by using the same image as both content and style source. Logically the output should be almost the same as the input without much artificial defects. If so, then this sanity check proves the transfer does maintain the content information and style information consistently without extenuating information or adding exterior information. The following figures present the output of sanity check by using Fig 5 black and white young man picture as both content and style source.

Fig 15. Sanity Check Output

2.2.3. Initialization Method

As mentioned previously, we experiment with two initialized inputs, the white noise and the content source. Surprisingly, because the backpropagation is heavily computed per pixel wise. After around 30 iterations, the difference between the results from both initialization methods is very minimal as shown in Fig 16.

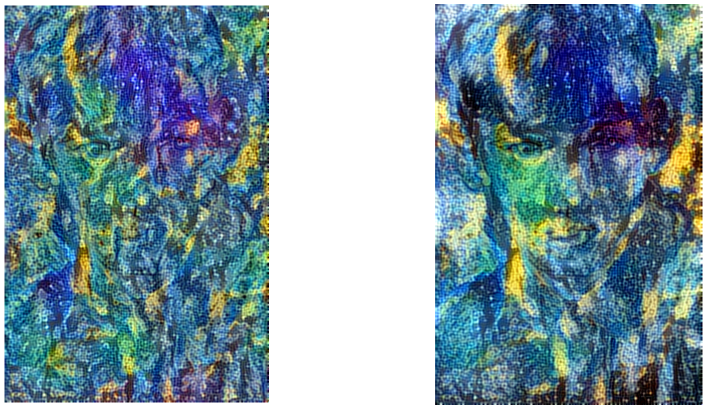


Initialized by White Noise Initialized by Content Source

Fig 16. Using Fig 5 Black and White Young Man as Content Source and Fig 9 Frida Kahlo as Style Source

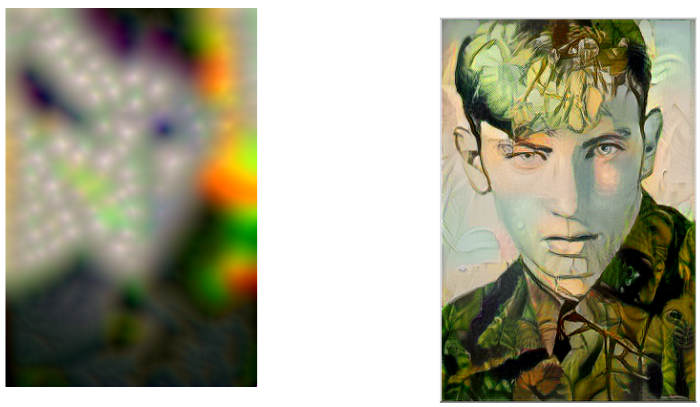
2.2.4. Preloading Weights

We choose to start experiments with both VGG 16 and VGG 19 models. Both convolutional neural networks are medium size compared with 100 layers' tiramisu neural network. However, they are still computationally expensive to train from scratch in order to perform well. Thus we experiment preloading the VGG16 model with pre-trained weights from ImageNet data set. We compare the results from pre-trained model and model with random initialized weights.



With Random Initialized Weights With Preloading Weights

Fig 17. Output after 10 Iterations using Fig 5 Black and White Young Man as Content Source and Fig 8 Starry Night as Style Source



With Random Initialized Weights With Preloading Weights

Fig 18. Output after 10 Iterations using Fig 5 Black and White Young Man as Content Source and Fig 9 Frida Kahlo as Style Source

Based on Fig 17 and Fig 18 above, the pred-trained model could produce more finalized output within limited iteration and is more efficient than the model with random initialized weights.

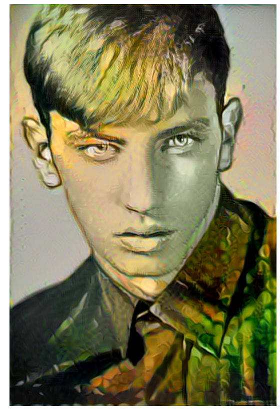
2.2.5. VGG16 and VGG 19

The results form VGG 19 are slightly more smooth, realistic and detailed than VGG16. The results from the 16 layers VGG are slightly noisy around boundaries.

2.2.6. Max Pooling vs Average Pooling

Average pooling results in more smooth output pictures than the max pooling in general.

max pooling



average pooling

Fig 19. Output after 10 Iterations using Fig 5 Black and White Young Man as Content Source and Fig 9 Frida Kahlo as Style Source

2.2.7. Content and Style Constant Weighting

In total loss function for backpropagation in equation (5), there are two constant parameters for both constant and style loss respectively. The ratio between these two constants greatly affects the output of transfer.

1:10000

1:1000

1:100

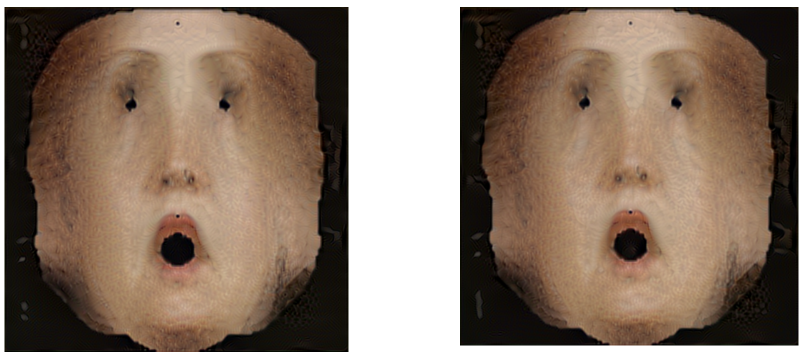
1:10

Fig 20. Relative weighting of matching content and style of the respective source images using Fig 6 Neckarfront in Tuebingen as content source and Fig 10 Composition VII as style source.

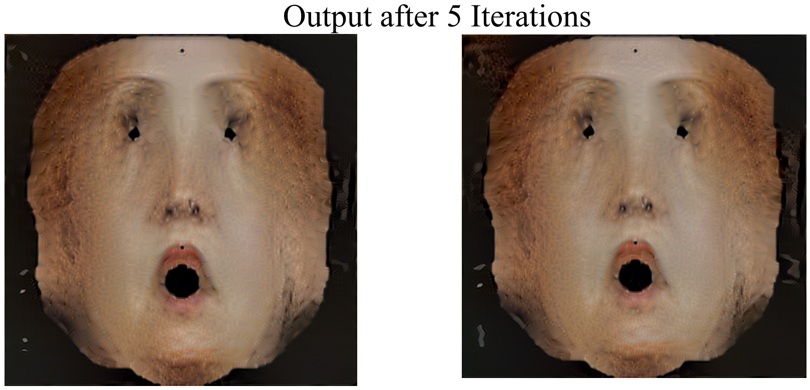
The ratio α/β between matching the content and matching the style increases from top left to bottom right. A high emphasis on the style effectively produces a texturized version of the style image (top left). A high emphasis on the content produces an image with only little stylization (bottom right). In practice one can smoothly interpolate between the two extremes.

The result is much more obvious on picture transfer but not so obvious on face because both the content face source and style face source are relatively similar.

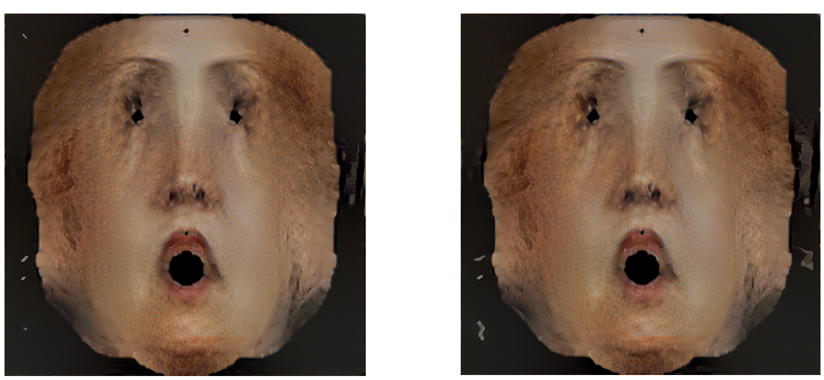
1:2000 1:6000



Output after 5 Iterations



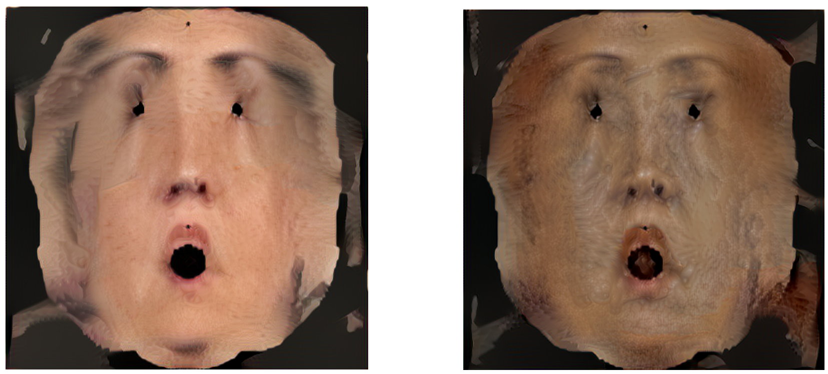
Output after 10 Iterations



Output after 200 Iterations

Fig 21. Output Using Fig 12 Sythesized Face as Content Source and Fig 14 Real Face 1 as Style Source

2.2.8. Similarity between Content and Style Source



Using Fig 14 Real Face 1 as Style Source   Using Fig 13 Real Face 2 as Style Source

Fig 22. Both Outputs Using Fig 12 Sythesized Face as Content Source

As shown from Fig 22, we also find that similar color tone between content source and style source results in better combined output for human face. If the difference of color tone between content source and style source is too extreme, the output results in some artificial defects.

2.2.9. Simplifying Network Structure

In order to improve the efficiency of the neural style transfer, attempting to decrease the complexity of network structure is an intuitive approach while trying to maintain reliable results.

The followings are outputs from experiments where some layers of VGG 16 network are deleted on purpose.

Output after 30 iterations. Only keeping lower layers, conv1_1, conv2_1, conv3_1 in the network

Output after 30 iterations. Only keeping conv1_1, conv1_2, conv2_1, conv2_2, conv3_1, conv3_2, conv4_1, conv4_2 in the network

Fig 23. Using Fig 5 Black and White Young Man as Content Source and Fig 9 Frida Kahlo as Style Source

From above Figure 23, we could compare the outputs from simplified models with the outputs in Figure 19 from VGG16 model. Apparently, decreasing the number of layers profoundly decreases the quality of transfer output. The content would still be maintained mostly but the style is barely transferred.

2.2.10. Different Layers for Style and Content Representative

Choosing different layers for content feature extractor or style feature extractor will also end up having different effects

2.2.10.1. Different layers for style representative

We find matching the style representations up to higher layers in the network preserves local images structures an increasingly large scale, leading to a smoother and more continuous visual experience. Thus, the visually most appealing images are usually created by matching the style representation up to high layers in the network.

Output after 20 iterations, only keeping higher level style layers, conv4_1, conv5_1 as style representatives

Output after 20 iterations, only keeping lower level style layers, conv1_1, conv2_1 as style representatives

Output after 20 iterations with layers conv1_1, conv2_1, conv3_1, conv4_1 and conv5_1 all as style representatives.

Fig 24. Using Fig 5 Black and White Young Man as Content Source and Fig 9 Frida Kahlo as Style Source

As observed from above Figure 24, a wider range of style layers results in a smoother and more continuous visual experience. Thus for the final version of neural style transfer, we match the style features in layers conv1_1, conv2_1, conv3_1, conv4_1 and conv5_1 of the network as style representation.

Output with layer conv1_1, conv2_1, conv3_1, conv4_1 and conv5_1 as style representative

Output with layer conv1_2, conv2_1, conv2_2, conv3_1, conv3_2, conv3_3, conv3_4, conv4_1, conv4_2, conv4_3, conv4_4,conv5_1, conv5_2, conv5_3, conv5_4 as style representative

Fig 25. Output after 30 Iterations using Fig 12 Synthesized Face Photo as Content Source and Fig 14. Real Face 2 as Style Source.

As above Figure 25, the output with more style representative layers achieve more smooth result after the same iterations.

2.2.10.2. Different layers for content representative

Matching the content representation in different layers of the network have different effect. As in Fig 26, matching the content on layer 'conv2_2' preserves much of the fine structure of the original photograph and the synthesized image looks as if the texture of the painting is simply blended over the photograph (middle). When matching the content on layer 'conv4_2' the texture of the painting and the content of the photograph merge together such that the content of the photograph is displayed in the style of the painting (bottom). Both images were generated with the same choice of parameters ($\frac{\alpha}{\beta} = 1\times10^{-3}$)[1].

Conv2_2

Conv4_2

Fig 26. Output using Fig 7 window as Content Source and Fig 11 Jesuiten III as Style Source

2.2.10.3. Using convX_2 layers as style representative

As in Fig 4, Structure of VGG 16 Network, the original implementation of the algorithm only focuses on using combinations of the first convolutional layers in each block as the style representatives. We try to experiment using the second convolutional layers in the blocks to check the performance.

Fig 27 Output after 20 iterations, using layers Conv1_2, Conv2_2, Conv3_2, Conv4_2 and Conv5_2 as style representative, Conv4_3 as content representative and using Fig 5 Black and White Young Man as Content Source and Fig 9 Frida Kahlo as Style Source

From the above Fig 27, we could see, comparing with the results in Figure 16, the result seems more abstract and artistic.

2.2.11. Different weighting combination for style layers

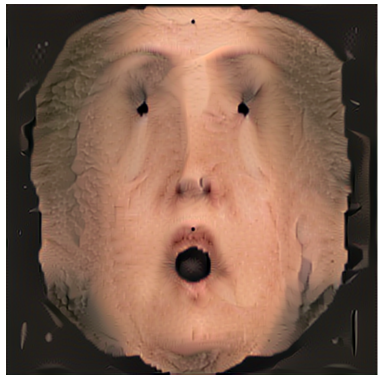
From previous section, the experiments suggest using only higher layers or only lower layers as style representative do not result in comparatively great performance output. Nevertheless, using different layers ranging from lower level to higher but weighting them differently may provide more information about how layers' impact on the overall performance for the style transfer.

Output with Weighting 0.1, 0.1, 0.2, 0.3 and 0.3 Respectively

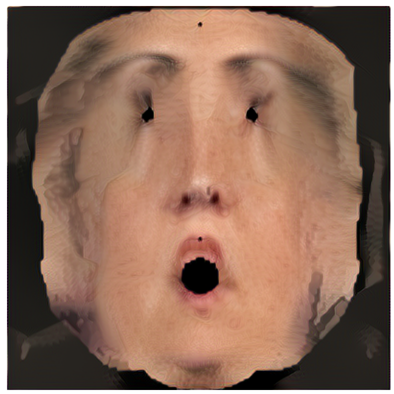
Output with Weighting 0.3, 0.3, 0.2, 0.1 and 0.1 Respectively

Fig 28 Output after 20 iterations using Fig 5 Black and White Young Man as Content Source and Fig 9 Frida Kahlo as Style Source with Layers conv1_1, conv2_1, conv3_1, conv4_1 and conv5_1 as Style Representatives

From Fig 28, heavier weighting on lower layers may cause the output closer to keep original structure information from the content source and heavier weighting on higher layers may cause the output to be more abstract, artistic and even have some distortion on the face in this case. Surprisingly, when weights for different style layers are small, the difference is more clear but when the weights exceed certain amount, the difference is very minimum.

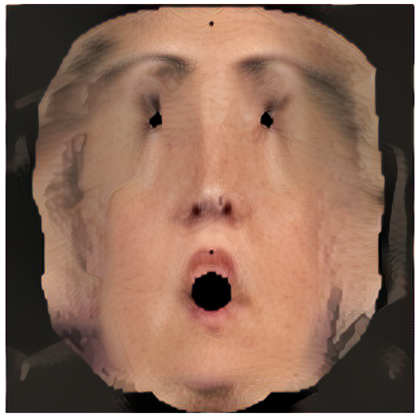


Output after 5 Iterations

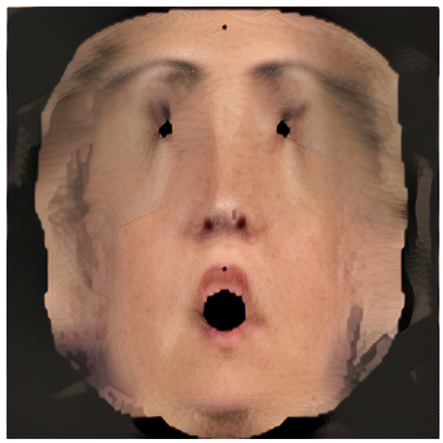


Output after 200 Iterations

Fig 29 Output using Fig 12 Synthesized Face Photo as Content Source and Fig 14 Real Face 2 as Style Source with Weighting, 400, 300, 200, 100, 100, 100, 100, 100, 100, 100, 100, 100,100,100, 100, 100 respectively. The network uses Layers conv1_1, conv1_2, conv2_1, conv2_2, conv3_1, conv3_2, conv3_3, conv3_4,conv4_1, conv4_2, conv4_3, conv4_4, conv5_1, conv5_2', conv5_3, conv5_4 as Style Representatives.



Output after 5 Iterations



Output after 200 Iterations

Fig 30 Output using Synthesized Face Photo as Content Source and Real Face 1 as Style Source with Weighting, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 200, 300,400,500, 600, 800, respectively. The network uses Layers conv1_1, conv1_2, conv2_1, conv2_2, conv3_1, conv3_2, conv3_3, conv3_4,conv4_1, conv4_2, conv4_3, conv4_4, conv5_1, conv5_2', conv5_3, conv5_4 as Style Representatives.

2.3. Conclusion

Even though the output of neural style face transfers still results in some artifacts inevitably but comparing with Fig 12 and Fig 30, we could easily observe the synthesized face is much improved and closer to real face after the transfer. It has more realistic details. Based on the observation, we could believe neural style transfer is a valid direction for further implementation for photo realistic face texture generation. However, we still face the difficulty of leveraging between the accuracy of performance and efficiency of the transfer.

2.4. Future Work

i. Further implement Fast Neural Style Transfer to increase the efficiency of the transfer [3].
ii. Potentially adding a Generative Adversarial Network(GAN) on top of the last layer of the model to fine tune face after preprocessing.

iii. Improving the mask procedure to allow the content source and style source have better preprocessed alignment before neural transfer.
iv. Experimenting more different sources of the content and style pairs to validate current results and conclusions.
v. Figuring out reasons for why less artifacts show up in lower style layer heavy weighting transfer comparing with higher style layer heavy weighting transfer. And also they happen less often in transfer with more different color tone between content and style comparing with the one with similar color tone.

3. FACE BOUNDING BOX REFINEMENT

This part of the research is a part of preprocessing procedure that could be used for submission in 2nd Facial Landmark Localization Competition. The goal of this part is to create an algorithm which could effectively predict bounding boxes including the target human faces from photos. This challenge aims at testing the ability of current systems for fitting unseen subjects, independently of variations in pose, expression, illumination, background, occlusion, and image quality.

3.1. Data Selection

Training data contains the facial images and their corresponding annotation (.pts file). Because the final official testing set includes both frontal faces and profile faces so I use training images of frontal images from dataset like FDDB, AFLW databases and previous datasets from 300W 1st Facial Landmark Localization Competition like LFPW, AFW, HELEN, and XM2VTS[4][5][6][7]. Of course, some profile side faces from dataset, Menpo39_Train are also mixed in the total training set.

3.1. Pre-processing method

3.1.1. Data Extracting and Filtering

Because the facial landmarks annotations are provided in the training set so I calculate the minimum and maximum values in the landmarks for both x and y axis. This allows me to get the longest edges for both edges. Then I pick a maximum value out of them as the bounding box edge for grand truth. The x, y value of the center point is obtained by averaging all the x and y values from the facial landmarks'. Thus the finalized ground truth for each image contains 3 values, x value of center point, y value of center point and edge length of bounding box.
After reshaping the images to uniform input size of 256 by 256 pixels and standard normalizing the data, 7 augmentation methods are carried out in the pipeline in order to make the algorithm more robust. The augmentation methods include

i. Scale: scaling the data to mimic the case of extreme big and small target headshot images
ii. Rotate: mimic the situation when the face is not center and tilted in the frame
iii. Brightness and contrast: make the algorithm more general to different lighting and color tone situations

iv. Translation
v. Gray: some of the testing photos are in black and white base
vi. Blur: mimic the situation when the resolution of the image is very low
vii. Combinations: varied combinations of methods above

3.2. Model Structure

There are two main experimental model structures in this research. The reason of designing these two models is to leverage between efficiency and accuracy.

3.2.1. Small Net

The small net consists of 5 blocks and each contains a convolutional layers following by a max pooling layer. The ending part consists of two fully connected layers with relu activation and each follows by a dropout layer. The last output dense layer uses linear activation function.

3.2.2. Big Net

The big net is almost the same as the small net except adding 8 more convolutional layers in the existing 5 blocks.

3.3. Training

In order to make sure the algorithm converge around global minimum point instead of local minimum, I used dynamic learning rate with mean square error for pixel predictions, stochastic gradient decent optimizer, decay of 1e-6, momentum of 0.9 and batch learning. The final optimized learning rate adjusting steps is obtained by experiments with combination of multiple learning rate decay.

3.4. Result

The benchmark goal of developing this algorithm is to make it better than the existing commercial bounding box prediction algorithm like Haar Cascades Face Detection in the pipeline. In the following figures, the green box is predicted by the existing algorithm, Haar Cascades Face Detection and the blue bounding box is predicted and finalized by this new trained algorithm.

3.4.1. Improvement Cases

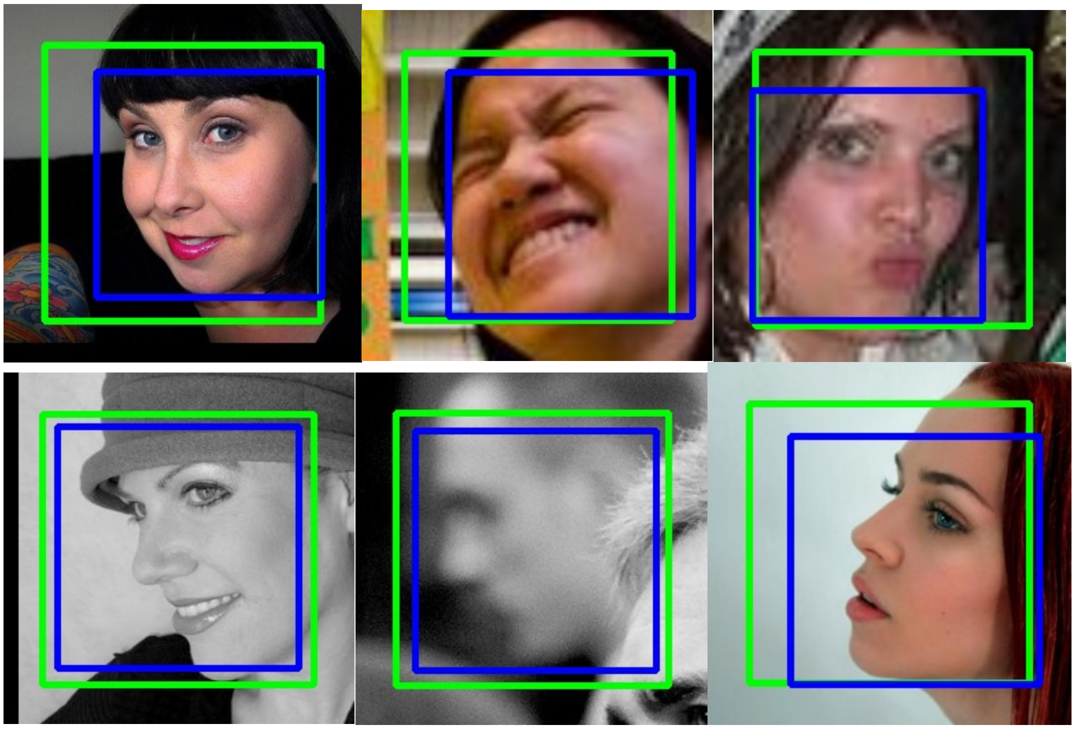
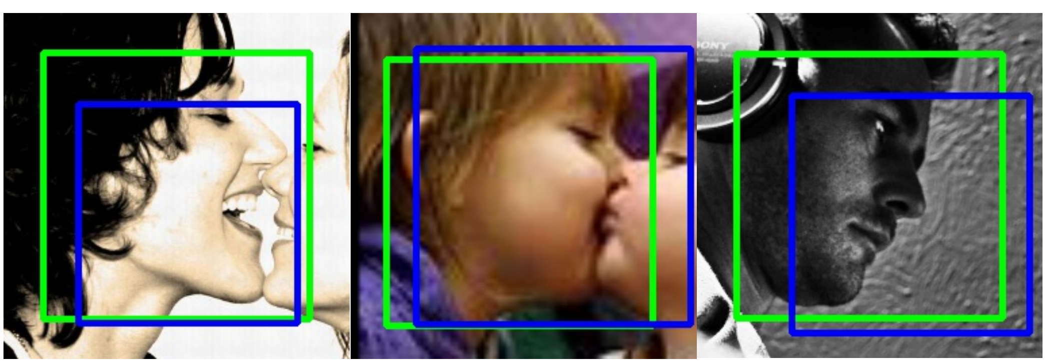


Fig 31. Improvement Cases in which the trained algorithm perform better than Haar Cascade Face detection algorithm

Generally, the training model could improve the accuracy of bounding box prediction. It could help to further narrow down the localization of the target face quite efficiently not matter of frontal or side face profiles.

3.4.2. Defected Cases



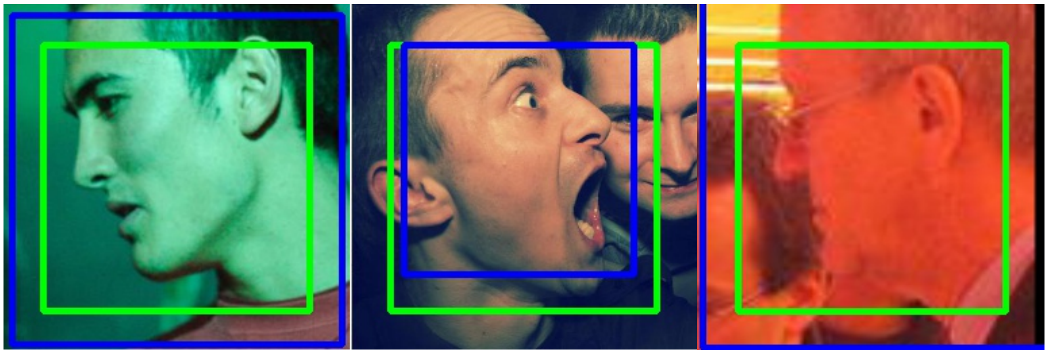


Fig 32. Defected Cases in which the trained algorithm perform worse than Haar Cascade Face detection algorithm

However, the algorithm still can be improved to be more stable in some edge cases. These edge cases mainly include photos where target face is companied by other faces, with low resolution or extreme lightings.

The code repository could be found from here(https://github.com/ZhecanJamesWang/Facial_Localization_Orientation_Detection)

3.5. Conclusion

As mentioned earlier, throughout the research we realize pose could greatly affect the accuracy of the face detection in unrestricted environment. The trained algorithm could perform better than the off self, commercial Haar Cascade Face Detection algorithm in many cases but still fails in some edge cases. Thus it needs more implementation to improve it stability in performance which is a very important factor when evaluating the face recognition algorithm.

3.6. Future Work

i. Implement more straightforward evaluation method to analyze the prediction results like IOU evaluation method.
ii. Previously experimented with using the algorithm to predict key facial points around tip of noise and chin in order to localize the bounding box but did not end up with great performance. Maybe could continue the implementation to improve the predictions.

REFERENCES

[1] Gatys, L.A., Ecker, A.S., Bethge, M.: A neural algorithm of artistic style. arXiv preprint arXiv:1508.06576 (2015)

[2]S. Saito, L. Wei, L. Hu, K. Nagano, and H. Li. Photorealistic facial texture inference using deep neural networks. arXiv preprint arXiv:1612.00523, 2016.

[3] J. Johnson, A. Alahi, and F. Li. Perceptual losses for real-time style transfer and super-resolution. In European Conference on Computer Vision (ECCV), pages 694–711. Springer, 2016.

[4] S. Zafeiriou, et. al. "The Menpo Facial Landmark Localisation Challenge", CVPRW, 2017.

[5] C. Sagonas, E. Antonakos, G, Tzimiropoulos, S. Zafeiriou, M. Pantic. 300 faces In-the-wild challenge: Database and results. Image and Vision Computing (IMAVIS), Special Issue on Facial Landmark Localisation "In-The-Wild". 2016.

[6] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, M. Pantic. 300 Faces in-the-Wild Challenge: The first facial landmark localization Challenge. Proceedings of IEEE Int'l Conf. on Computer Vision (ICCV-W), 300 Faces in-the-Wild Challenge (300-W). Sydney, Australia, December 2013.

[7] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, M. Pantic. A semi-automatic methodology for facial landmark annotation. Proceedings of IEEE Int'l Conf. Computer Vision and Pattern Recognition (CVPR-W), 5th Workshop on Analysis and Modeling of Faces and Gestures (AMFG 2013). Oregon, USA, June 2013.