Neural style transfer

KIM JAE MIN
AD20216801

Neural style transfer

Introduction

Style transfer consists in generating an image with the same "content" as a base image, but with the "style" of a different picture (typically artistic).

- **style loss**

The style loss is where the deep learning keeps in --that one is defined using a deep convolutional neural network. Precisely, it consists in a sum of L2 distances between the Gram matrices of the representations of the base image and the style reference image, extracted from different layers of a convnet (trained on ImageNet). The general idea is to capture color/texture information at different spatial scales (fairly large scales --defined by the depth of the layer considered).

- **content loss**

The content loss is a L2 distance between the features of the base image (extracted from a deep layer) and the features of the combination image, keeping the generated image close enough to the original one.

- **total variation loss**

The total variation loss imposes local spatial continuity between the pixels of the combination image, giving it visual coherence.

Neural style transfer

Description

Transfering the style of a reference image to target image using gradient descent.

Target image

Reference image

Neural style transfer

Description

Transfering the style of a reference image to target image using gradient descent.

Generated image

Neural style transfer

Functions

def preprocess_image(image_path):

Util function to open, resize and format pictures into appropriate tensors

def deprocess_image(x):

Util function to convert a tensor into a valid image

Neural style transfer

Style transfer functions

def gram_matrix(x):

The gram matrix of an image tensor (feature-wise outer product)

def style_loss(style, combination):

The "style loss" is designed to maintain the style of the reference image in the generated image.
It is based on the gram matrices (which capture style) of feature maps from the style reference image and from the generated image

def content_loss(base, combination):

An auxiliary loss function designed to maintain the "content" of the base image in the generated image

def total_variation_loss(x):

The 3rd loss function, total variation loss, designed to keep the generated image locally coherent

Neural style transfer

Model

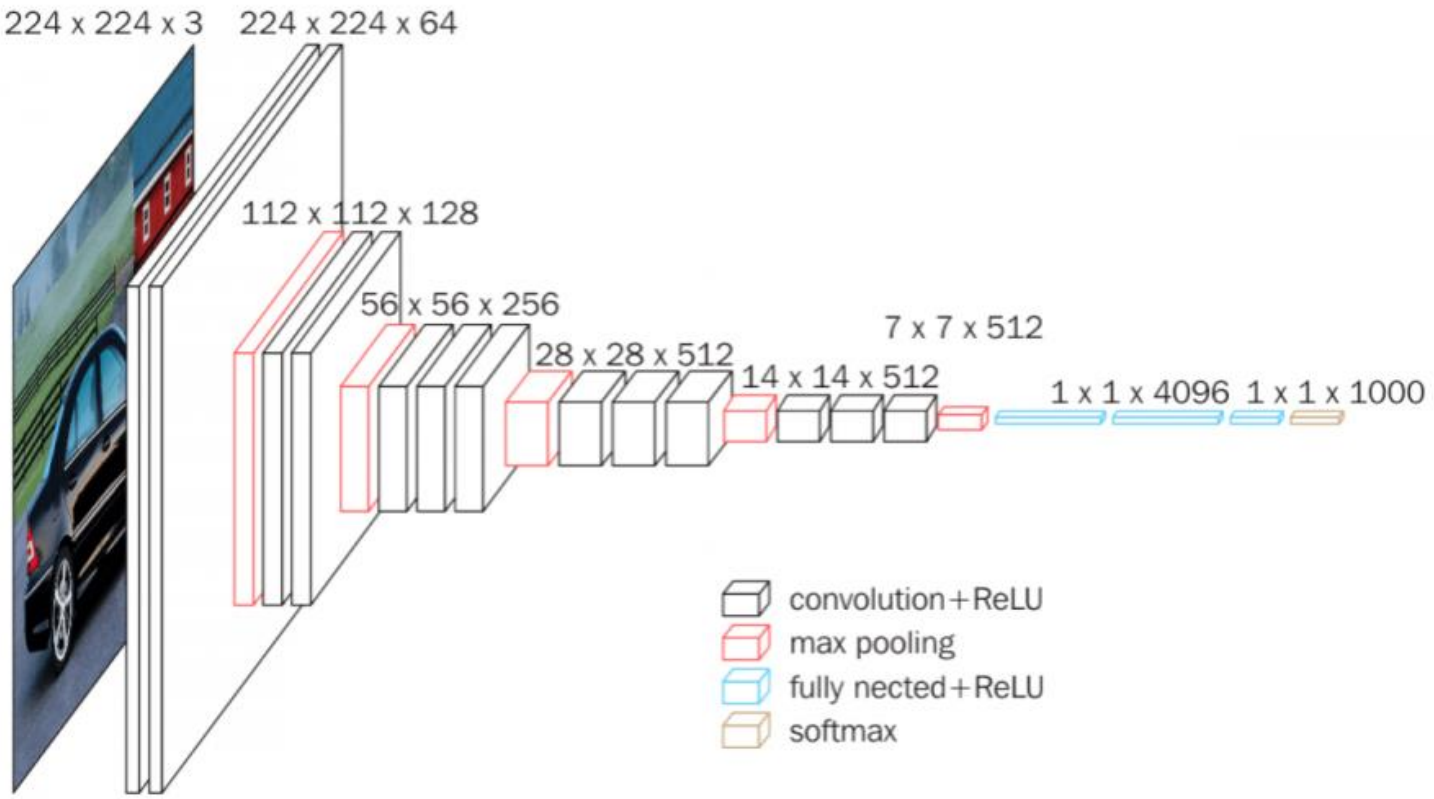
VGG19 algorithm or VGGNet

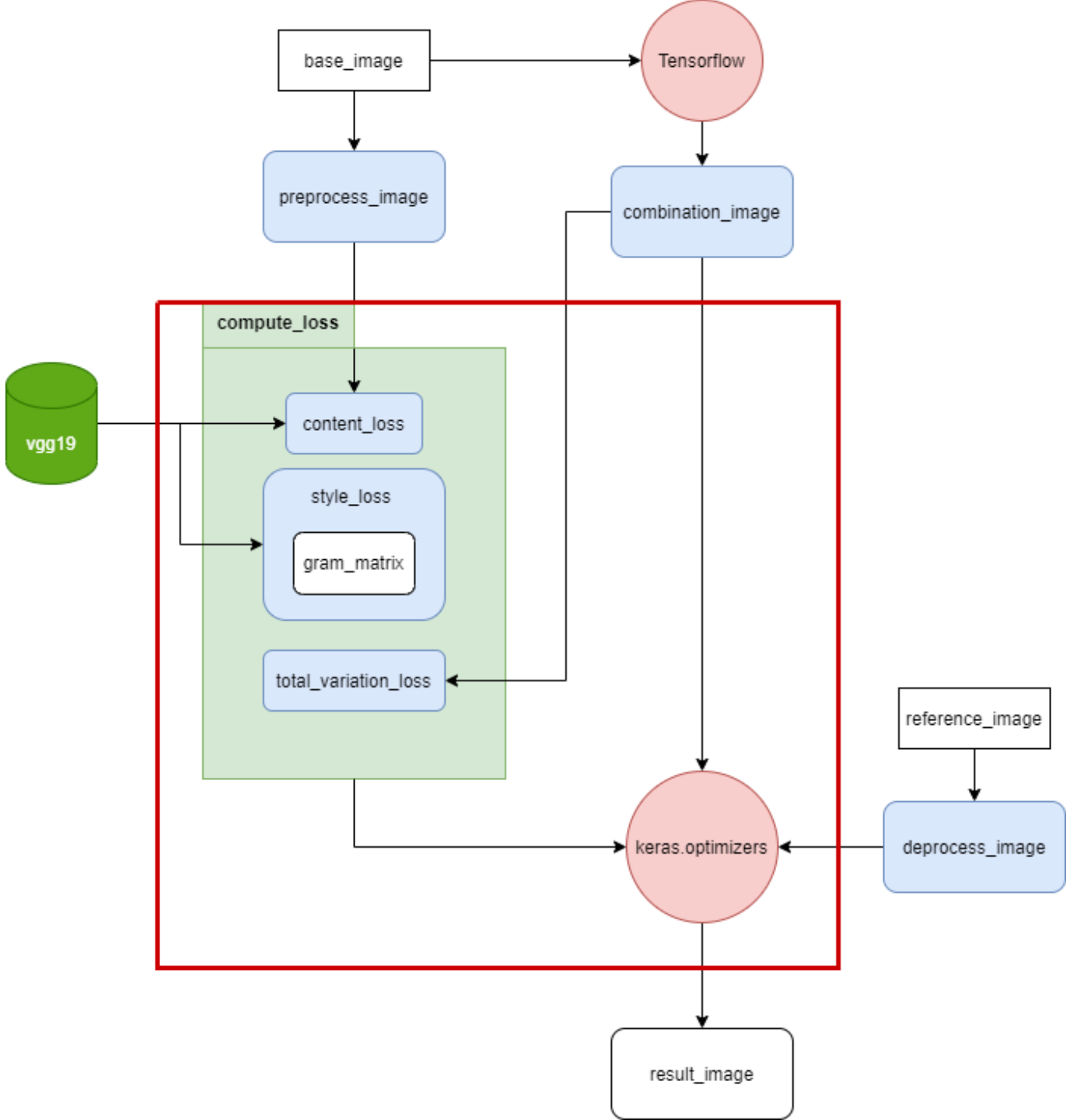
- VGGNet is a model developed by VGG, a research team at Oxford University, and it is the model that won the 2014 ImageNet image recognition contest.

- VGGNet means a model consisting of 16 or 19 layers.

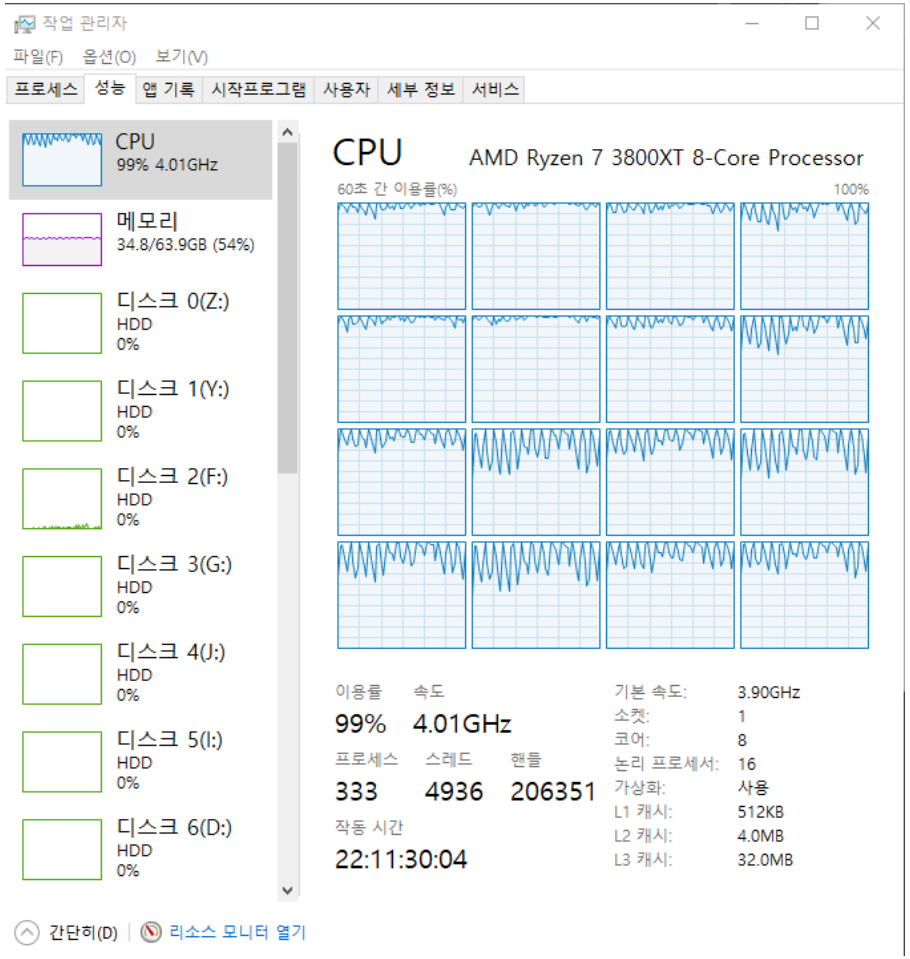
Neural style transfer

VGG16 algorithm of VGGNet




base_image
Tensorflow
preprocess_image
combination_image
compute_loss
vgg19
content_loss
style_loss
gram_matrix
total_variation_loss
reference_image
keras.optimizers
deprocess_image
result_image

Neural style transfer



Spec:

CPU: AMD Ryzen 7 3800XT 8Core, 16 Threads
RAM: 64G
Train Mode : CPU Mode

Train Time:

Start time : 21:04
End time : Next day 01:51

The Train Time: about 6 hours

Neural style transfer

Iteration 100: loss=2700.52
Iteration 200: loss=2140.55
Iteration 300: loss=1888.79
Iteration 400: loss=1738.28
Iteration 500: loss=1635.57
Iteration 600: loss=1559.19
Iteration 700: loss=1499.00
Iteration 800: loss=1449.74
Iteration 900: loss=1408.50
Iteration 1000: loss=1373.33
Iteration 1100: loss=1342.80
Iteration 1200: loss=1315.90
……
……

Iteration 3700: loss=1055.29
Iteration 3800: loss=1051.24
Iteration 3900: loss=1047.40
Iteration 4000: loss=1043.75

Neural style transfer

Train result

Iteration 100

Iteration 4,000

Neural style transfer

Try other train result

Target image

Reference image(Cow picture of LEE JUNG SUB)

Neural style transfer

Train result

Generated image Iteration 4,000

Neural style transfer

The End

Thanks !