

Artistic Style Transfer

Industrial AI Lab.
Prof. Seungchul Lee

Artistic Style Transfer







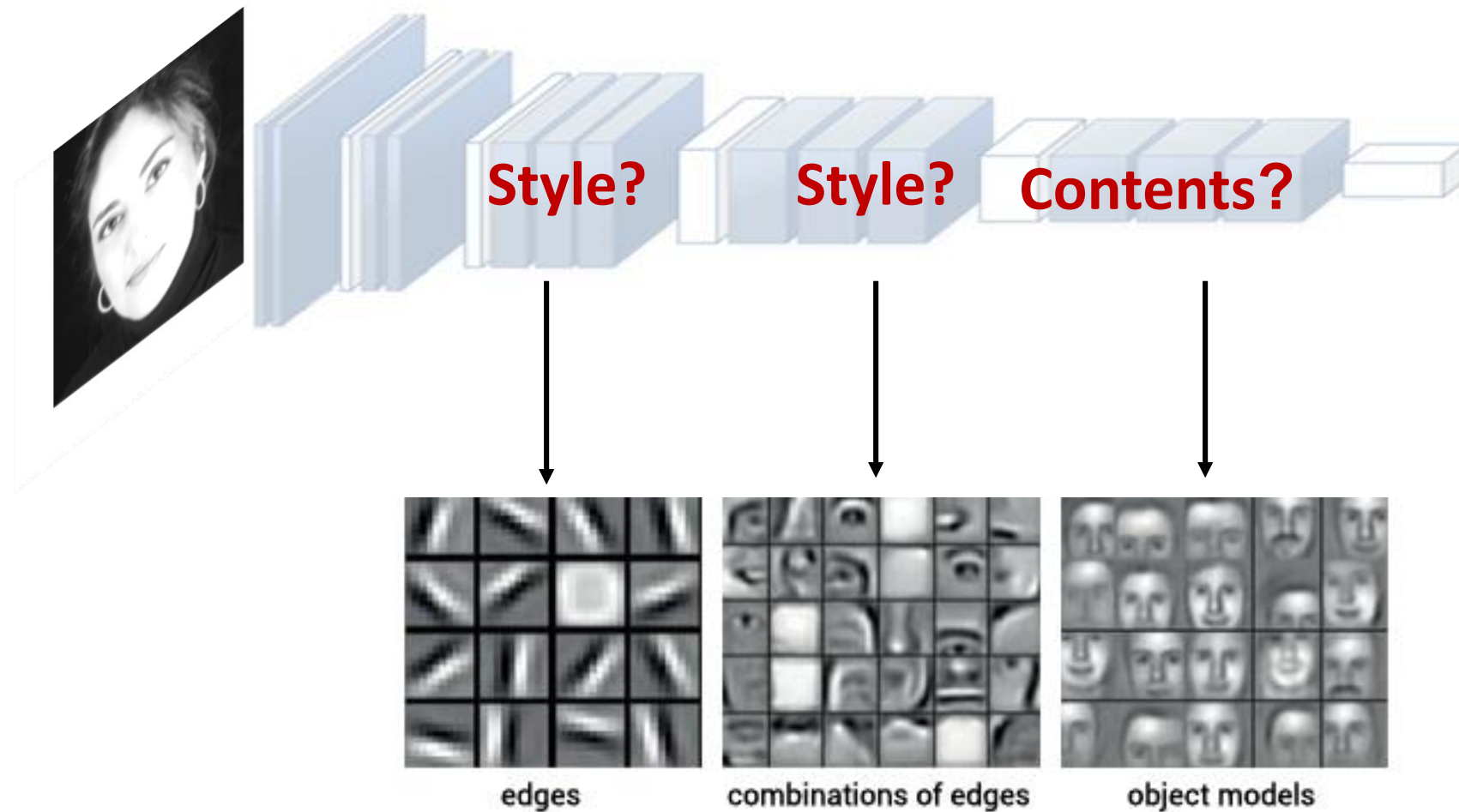
Vincent van Gogh



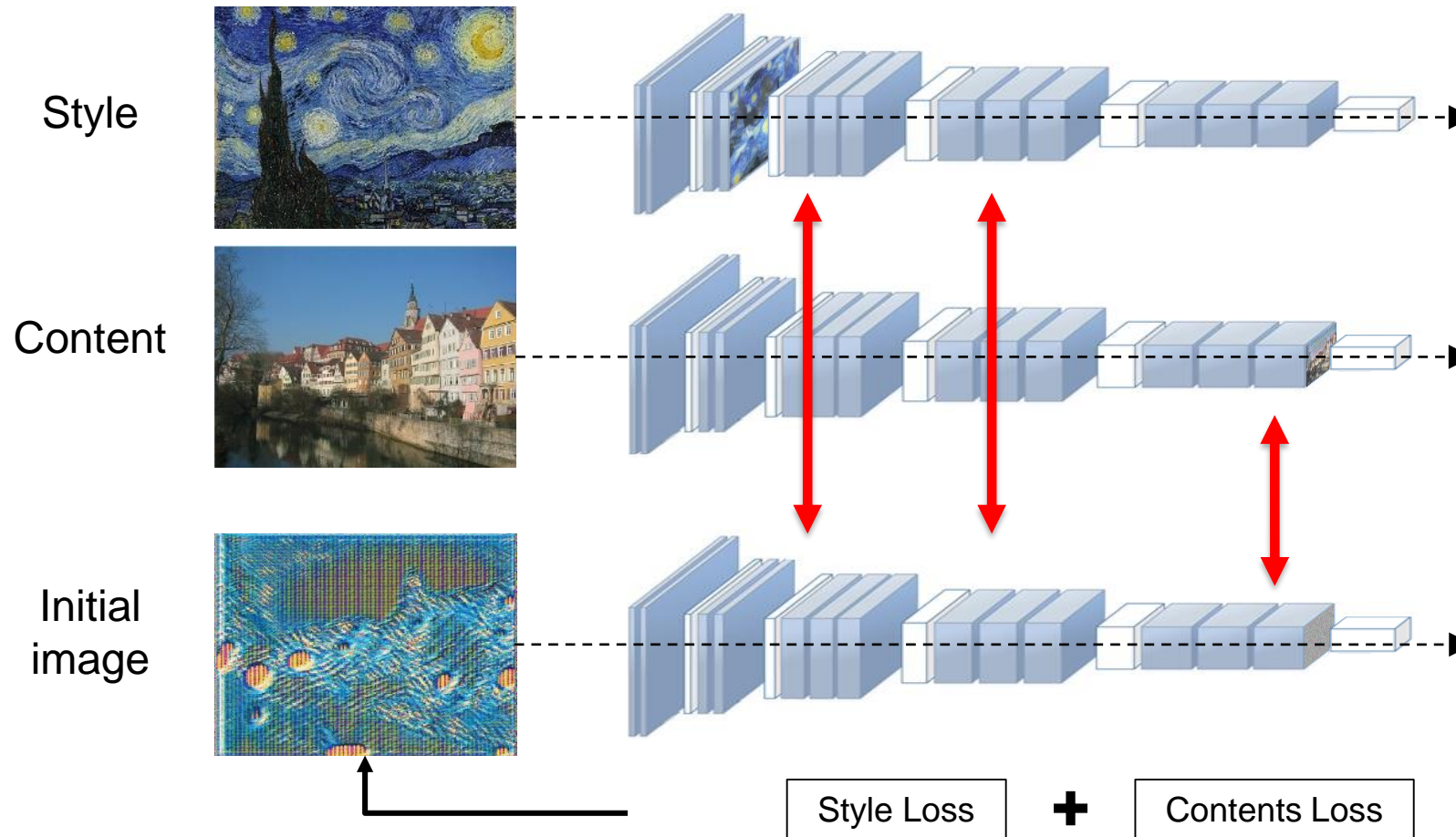
Pablo Picasso

Convolutional NN

- Hierarchical feature representation

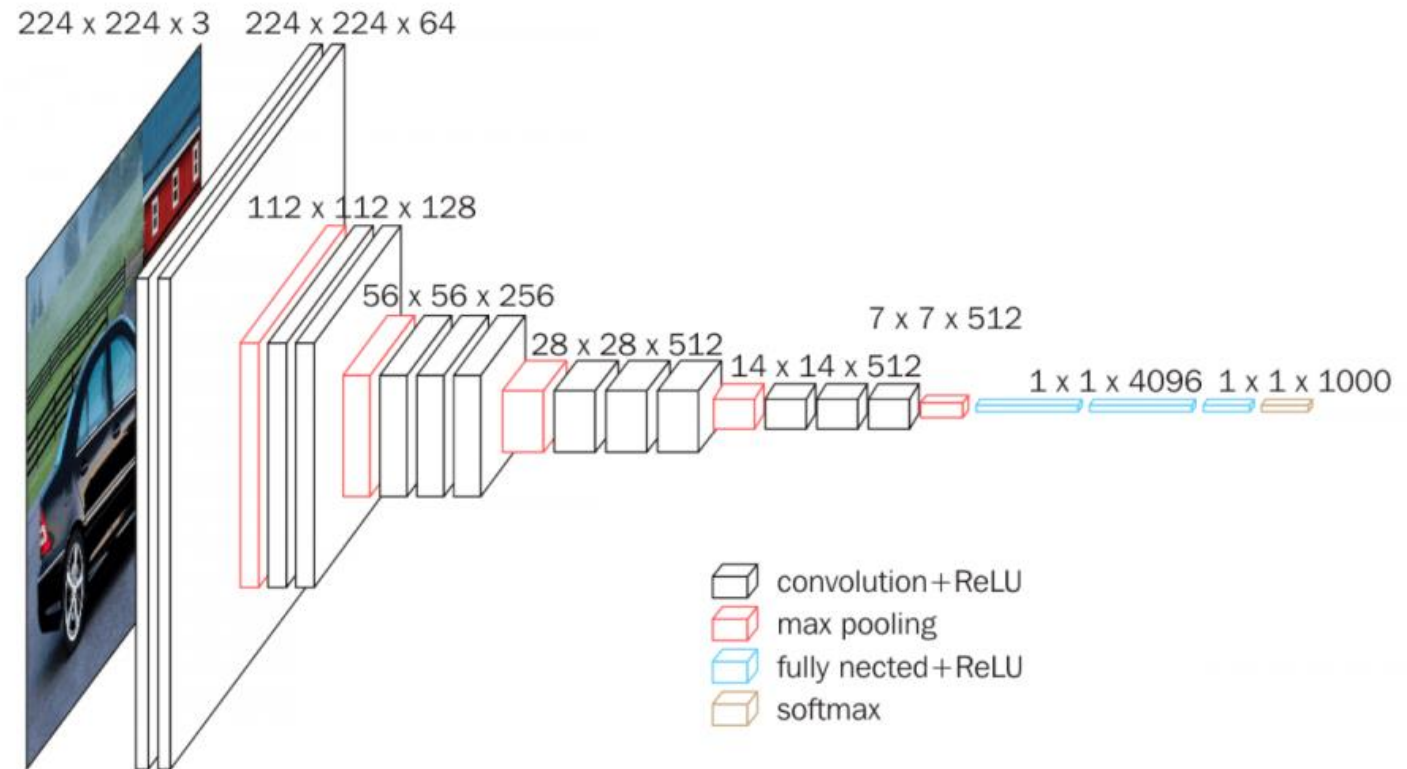


Artistic Style Transfer

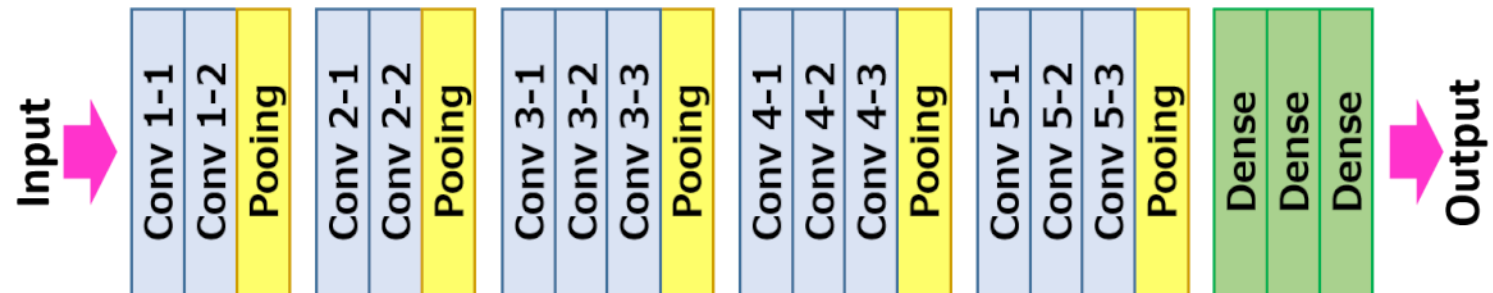


Implementation

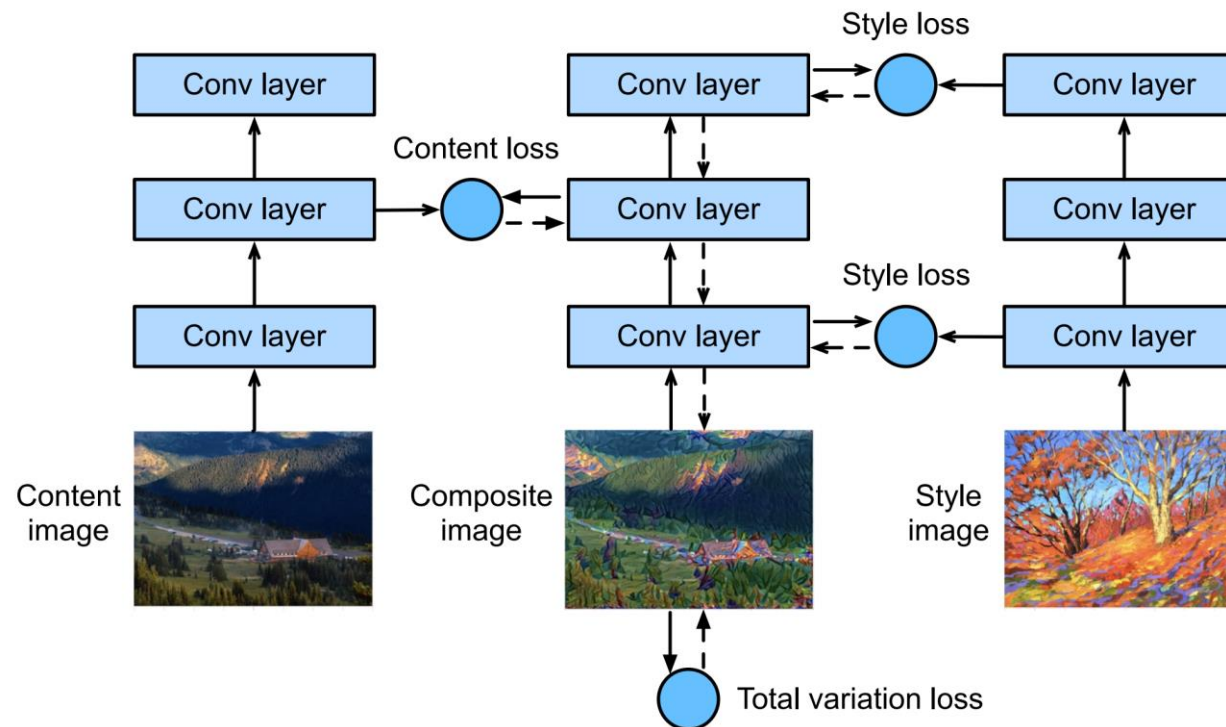
- VGG16



VGG-16



- Composite image is the only variable that needs to be updated
- We select a pre-trained CNN to extract image features
- The output of certain layers to use as content features or style features



- Content loss is used to make the composite image approximate the content image as regards content features

```
def get_loss_content(gen_layer, ref_layer):  
    loss = tf.reduce_mean(tf.square(gen_layer - ref_layer))  
    return loss
```

- Style loss is used to make the composite image approximate the style image in terms of style features
 - Gram matrix

```
def get_gram_matrix(conv_layer):  
    channels = conv_layer.get_shape().as_list()[3]  
    conv_layer = tf.reshape(conv_layer, (-1, channels))  
    gram_matrix = tf.matmul(tf.transpose(conv_layer), conv_layer)  
    return gram_matrix / ((conv_layer.get_shape().as_list()[0]) * channels)  
  
def get_loss_style(gram_matrix_gen, gram_matrix_ref):  
    loss = tf.reduce_mean(tf.square(gram_matrix_gen - gram_matrix_ref))  
    return loss
```



Total Variance Loss

- Sometimes, the composite images we learn have a lot of high-frequency noise, particularly bright or dark pixels.
- One common noise reduction method is total variation denoising.

$$\sum_{i,j} |x_{i,j} - x_{i+1,j}| + |x_{i,j} - x_{i,j+1}|$$

```
def get_loss_TV(conv_layer):  
    loss = tf.reduce_mean(tf.abs(conv_layer[:, :, 1 :, :] - conv_layer[:, :, :-1, :])) ...  
        + tf.reduce_mean(tf.abs(conv_layer[:, 1 :, :, :] - conv_layer[:, :, -1, :, :]))  
    return loss
```