2018.12.15

Image Style Transfer with CNN

Image Style Transfer Using Convolutional Neural Networks

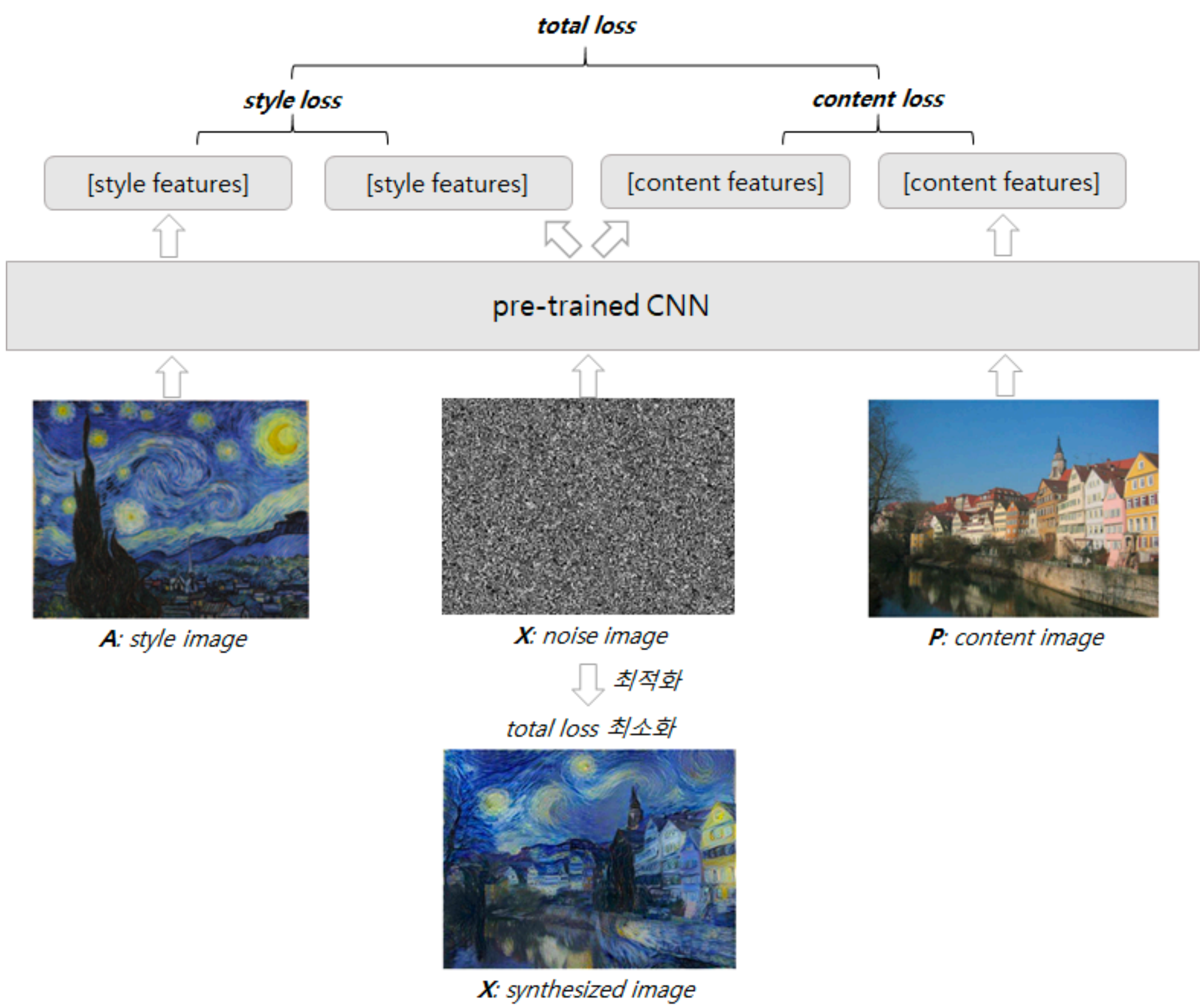
Leon A. Gatys
Centre for Integrative Neuroscience, University of Tübingen, Germany
Bernstein Center for Computational Neuroscience, Tübingen, Germany
Graduate School of Neural Information Processing, University of Tübingen, Germany
leon.gatys@bethgelab.org

Alexander S. Ecker
Centre for Integrative Neuroscience, University of Tübingen, Germany
Bernstein Center for Computational Neuroscience, Tübingen, Germany
Max Planck Institute for Biological Cybernetics, Tübingen, Germany
Baylor College of Medicine, Houston, TX, USA

Matthias Bethge
Centre for Integrative Neuroscience, University of Tübingen, Germany
Bernstein Center for Computational Neuroscience, Tübingen, Germany
Max Planck Institute for Biological Cybernetics, Tübingen, Germany

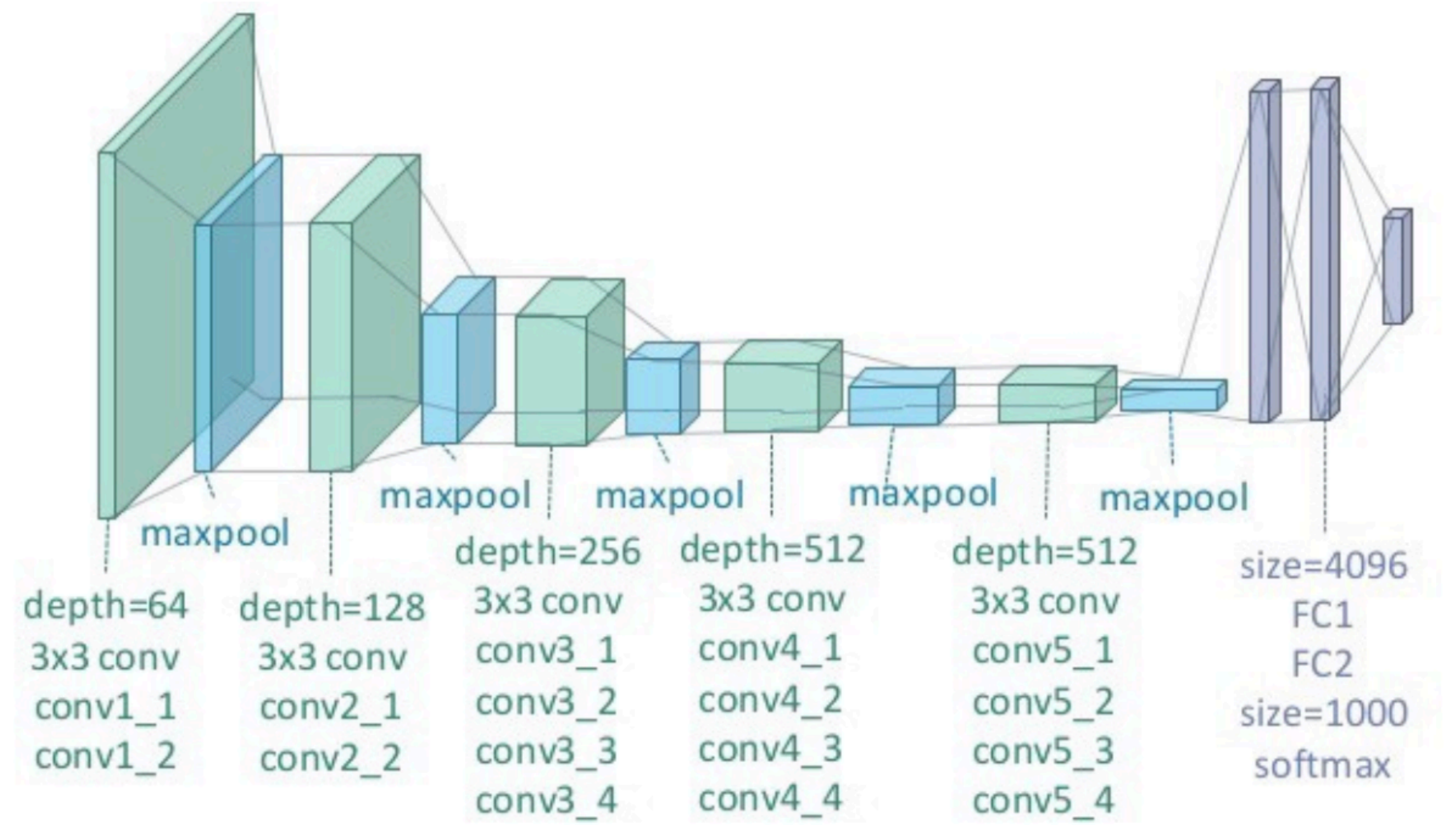
***P**: content image*   ***A**: style image*   ***X**: synthesized image*

$$
\begin{aligned}
loss \;=\;& distance(style(reference) - style(new)) + \\
& distance(content(original) - content(new))
\end{aligned}
$$



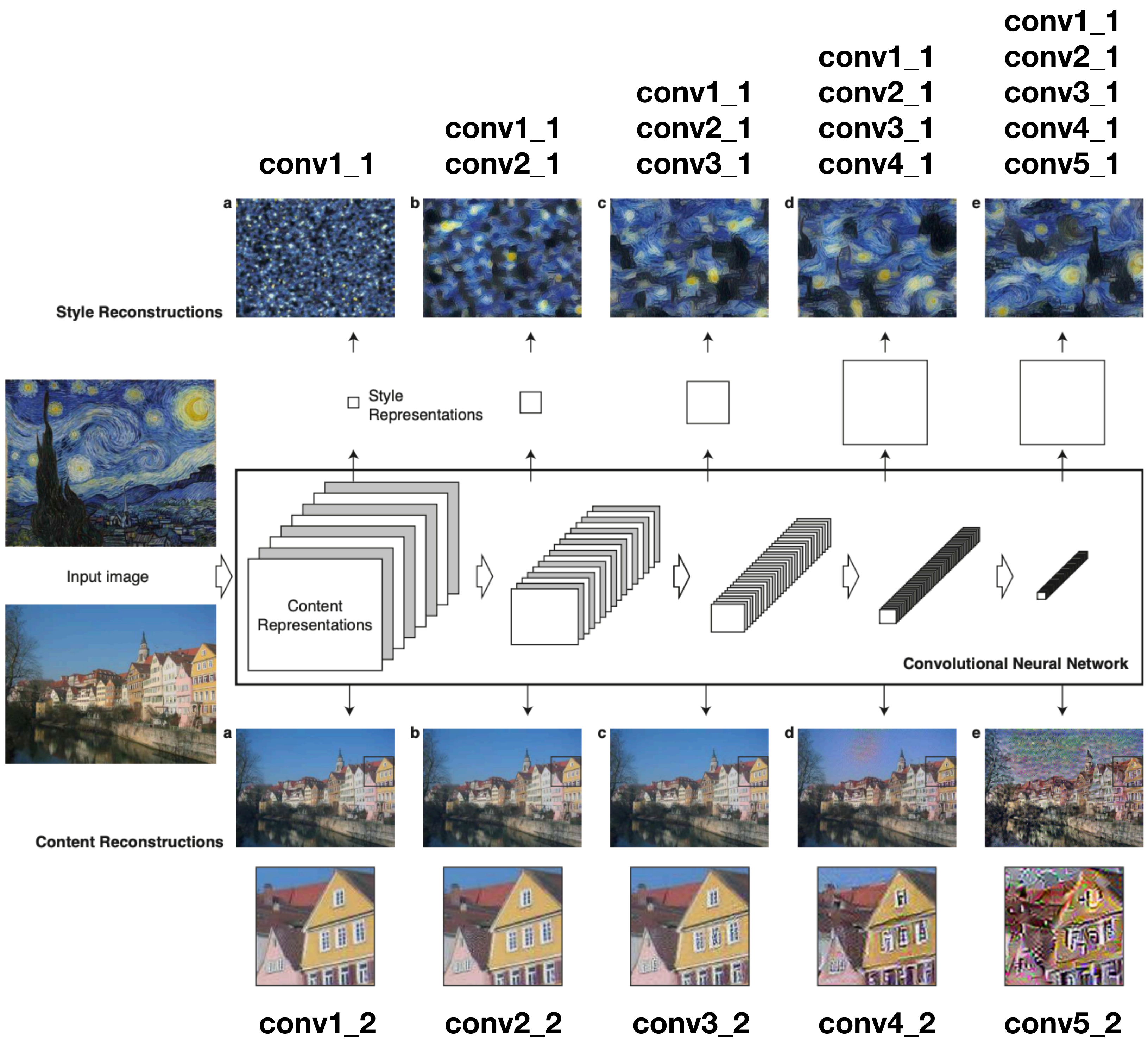
https://www.popit.kr/neural-style-transfer-%EB%94%B0%EB%9D%BC%ED%95%98%EA%B8%B0/

모델: VGG19

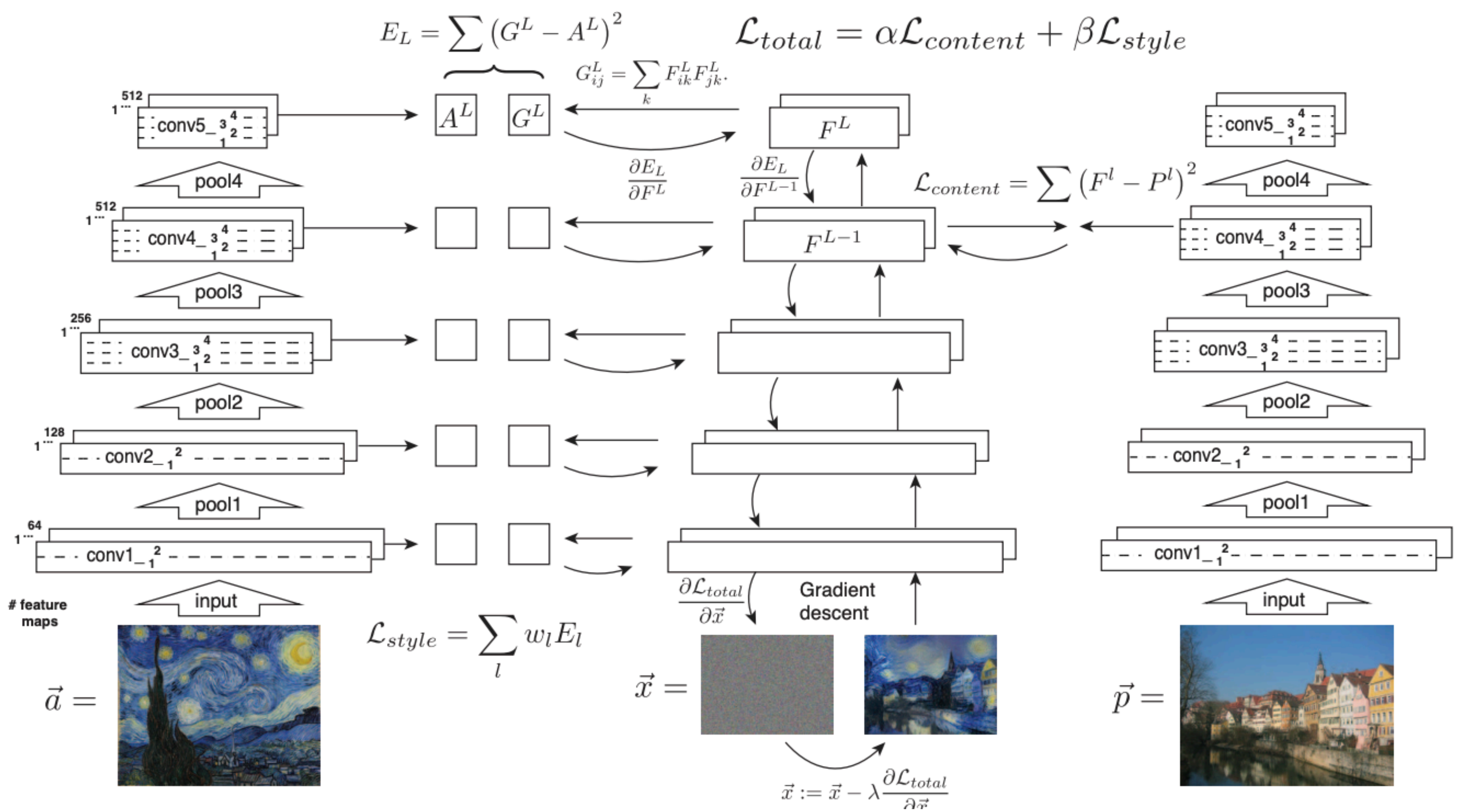


https://www.slideshare.net/ckmarkohchang/applied-deep-learning-1103-convolutional-neural-networks

두 가지 방법의 Reconstruction



$$\mathcal{L}_{\text{total}}(\vec{p},\vec{a},\vec{x}) = \alpha\mathcal{L}_{\text{content}}(\vec{p},\vec{x}) + \beta\mathcal{L}_{\text{style}}(\vec{a},\vec{x})$$



Deep image representation

Content

High-level layer의 feature map 사용
→ 이미지의 **전역적인 특성**을 보유
→ **Content 복원**에 필수적인 요소들만 가져옴

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

Style

Style(Texture)은 공간적인 정보와 **무관**해야 함
→ 각 layer의 **feature map 사이의 상관관계**

Gram matrix → correlation을 얻어냄

$$G_{ij}^{l} = \sum_{k} F_{ik}^{l} F_{jk}^{l}.$$

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

$$\mathcal{L}_{\text{style}}(\vec{a}, \vec{x}) = \sum_{l=0}^{L} w_l E_l,$$

Style Transfer

$$\boxed{\mathcal{L}_{\text{total}}(\vec{p}, \vec{a}, \vec{x}) = \alpha \mathcal{L}_{\text{content}}(\vec{p}, \vec{x}) + \beta \mathcal{L}_{\text{style}}(\vec{a}, \vec{x})}$$

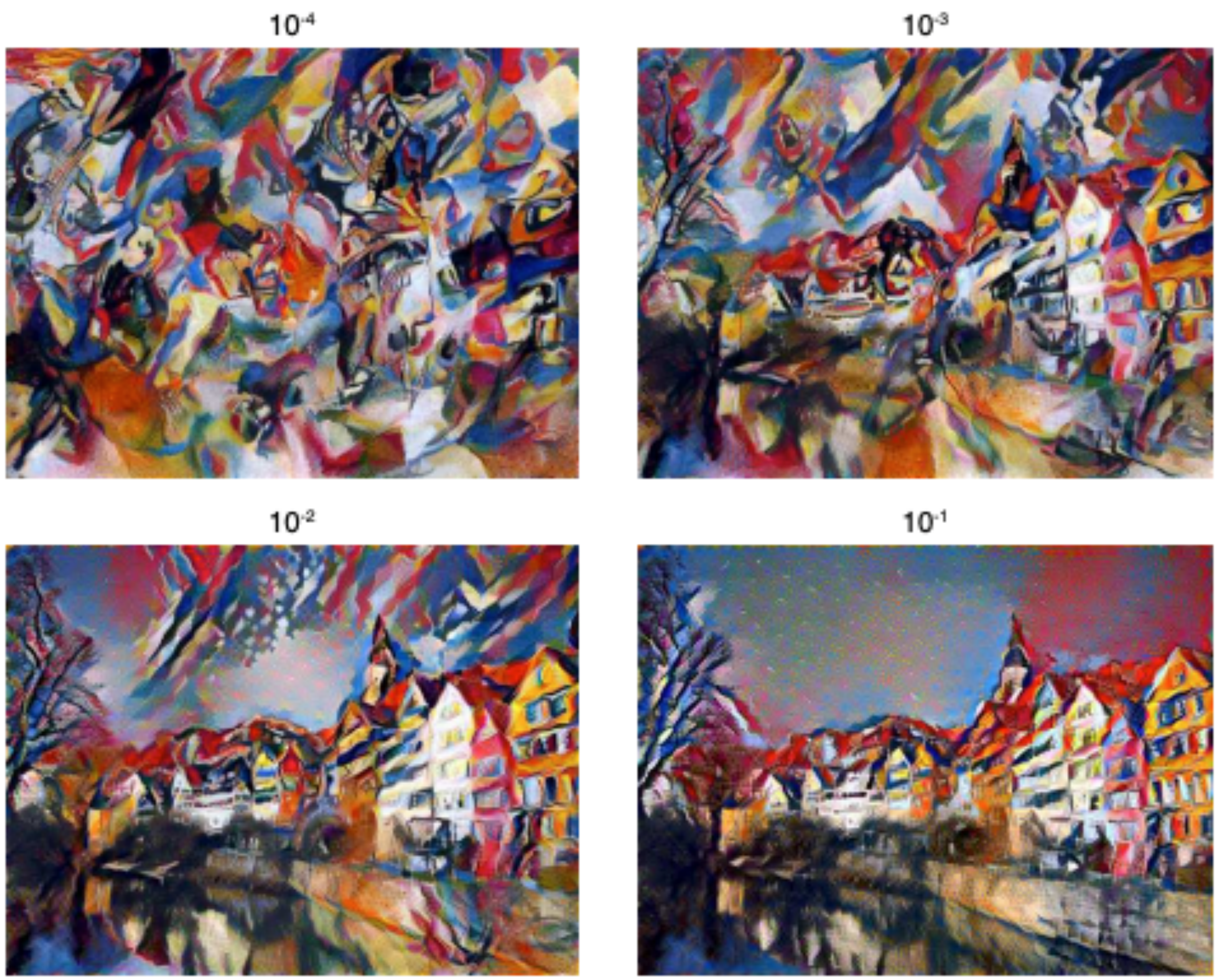
gradient descent
with **L-BFGS**

Limited-memory-**B**royden–**F**letcher–**G**oldfarb–**S**hanno algorithm

최적화하려는 영역을 **2차 함수**로 가정
2차 도함수 대신에 Hessian 행렬 정보를 이용

variable의 갯수가 많을 때 **가볍다**

Content와 Style matching의 Trade-off



▲ α/β의 비율에 따른 이미지 변화

Content 복원을 위한 layer 선택

깊은 레이어에서
content reconstruction을 할 수록
이미지의 *detailed pixel information*와
*texture*는 사라진다

경사하강법을 위한 초기화 방법

A: content 이미지
B: style 이미지
C: 화이트 노이즈 이미지

→ 큰 영향은 없다