Vision for Graphics

CS280: Computer Vision

Alexei Efros, UC Berkeley, Spring 2017

Weather Forecasting for Dummies™

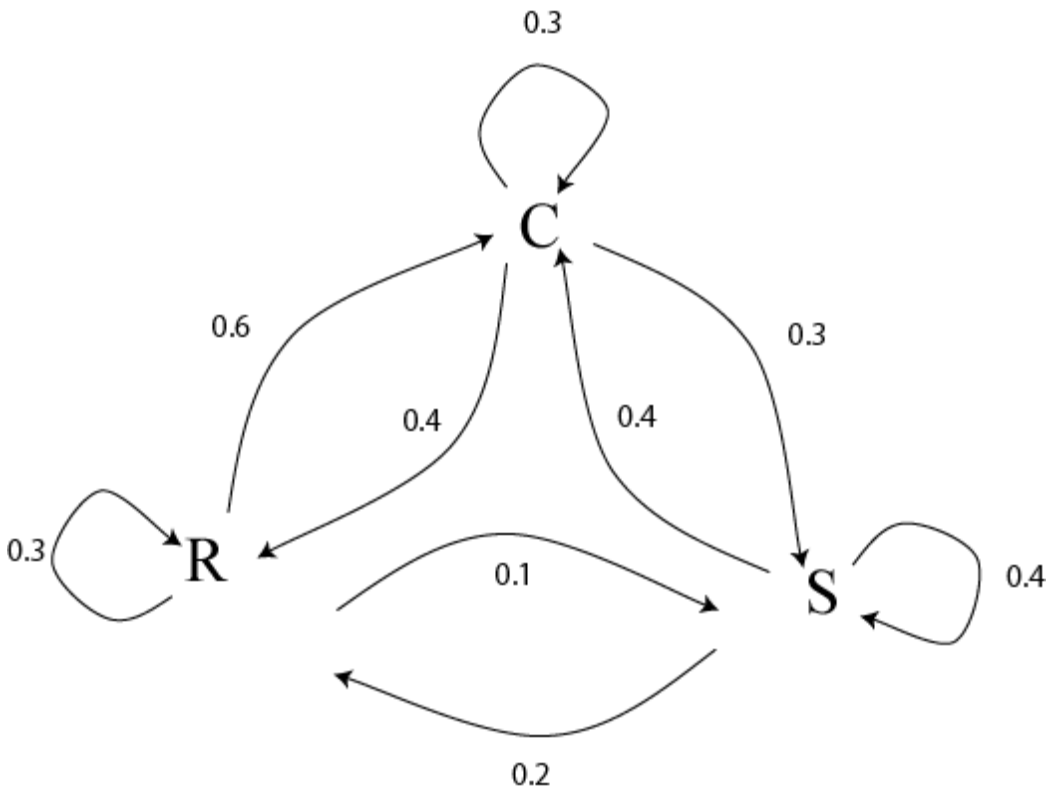
Let's predict weather:

- Given today's weather only, we want to know tomorrow's
- Suppose weather can only be {Sunny, Cloudy, Raining}

The "Weather Channel" algorithm:

- Over a long period of time, record:
  - How often S followed by R
  - How often S followed by S
  - Etc.
- Compute percentages for each state:
  - P(R|S), P(S|S), etc.
- Predict the state with highest probability!
- It's a Markov Chain

Markov Chain



$$\begin{pmatrix} 0.3 & 0.6 & 0.1 \\ 0.4 & 0.3 & 0.3 \\ 0.2 & 0.4 & 0.4 \end{pmatrix}$$

What if we know today and yestarday's weather?

Text Synthesis

[Shannon,'48] proposed a way to generate English-looking text using N-grams:

- Assume a generalized Markov model
- Use a large text to compute prob. distributions of each letter given N-1 previous letters
- Starting from a seed repeatedly sample this Markov chain to generate new letters
- Also works for whole words

WE NEED TO EAT CAKE

Mark V. Shaney (Bell Labs)

Results (using `alt.singles` corpus):

- *"As I've commented before, really relating to someone involves standing next to impossible."*
- *"One morning I shot an elephant in my arms and kissed him."*
- *"I spent an interesting evening recently with a grain of salt"*

Video Textures

Arno Schödl
Richard Szeliski
David Salesin
Irfan Essa

Microsoft Research, Georgia Tech

Still photos

Video clips

Video textures

Problem statement

video clip

video texture

Our approach

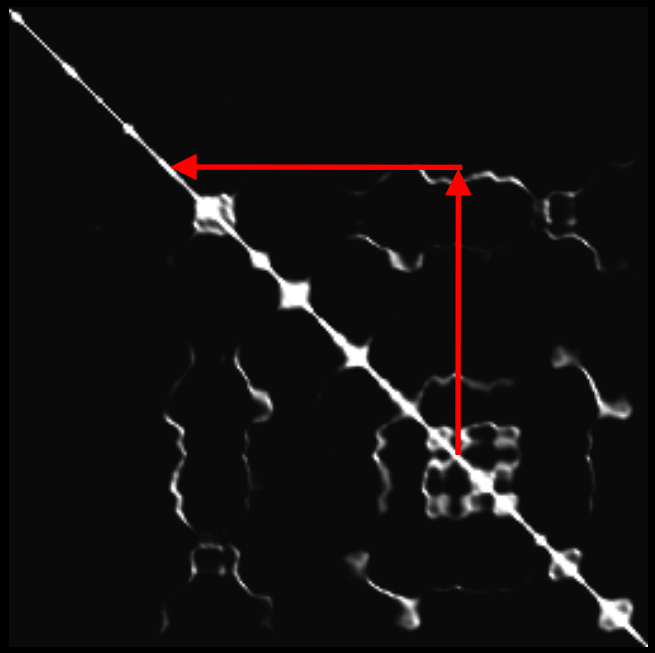
- How do we find good transitions?

Finding good transitions

- Compute L_2 distance $D_{i,j}$ between all frames

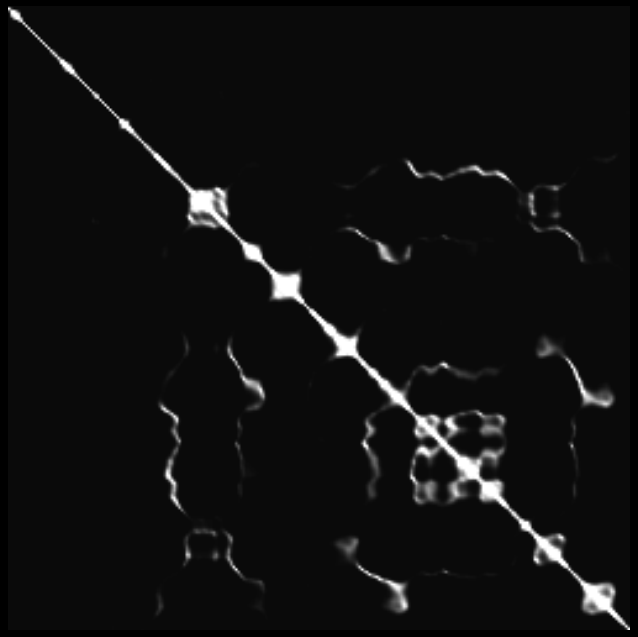
vs. → frame *i*

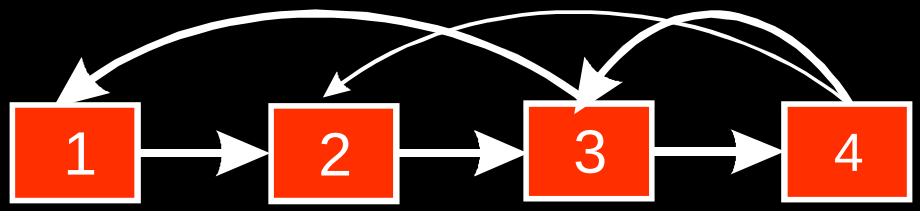
↓ frame *j*



Similar frames make good transitions

Markov chain representation

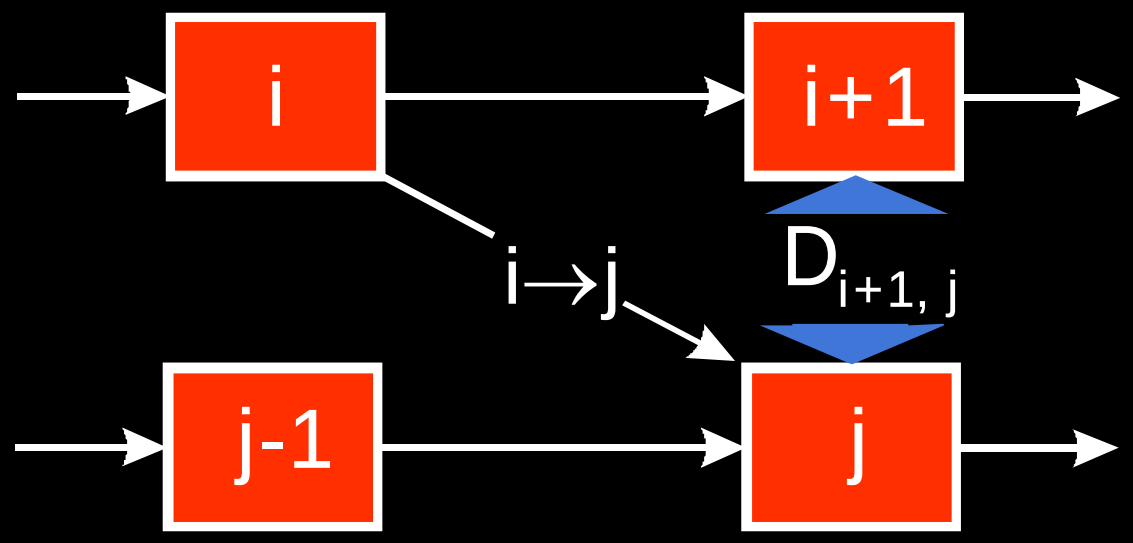




Similar frames make good transitions

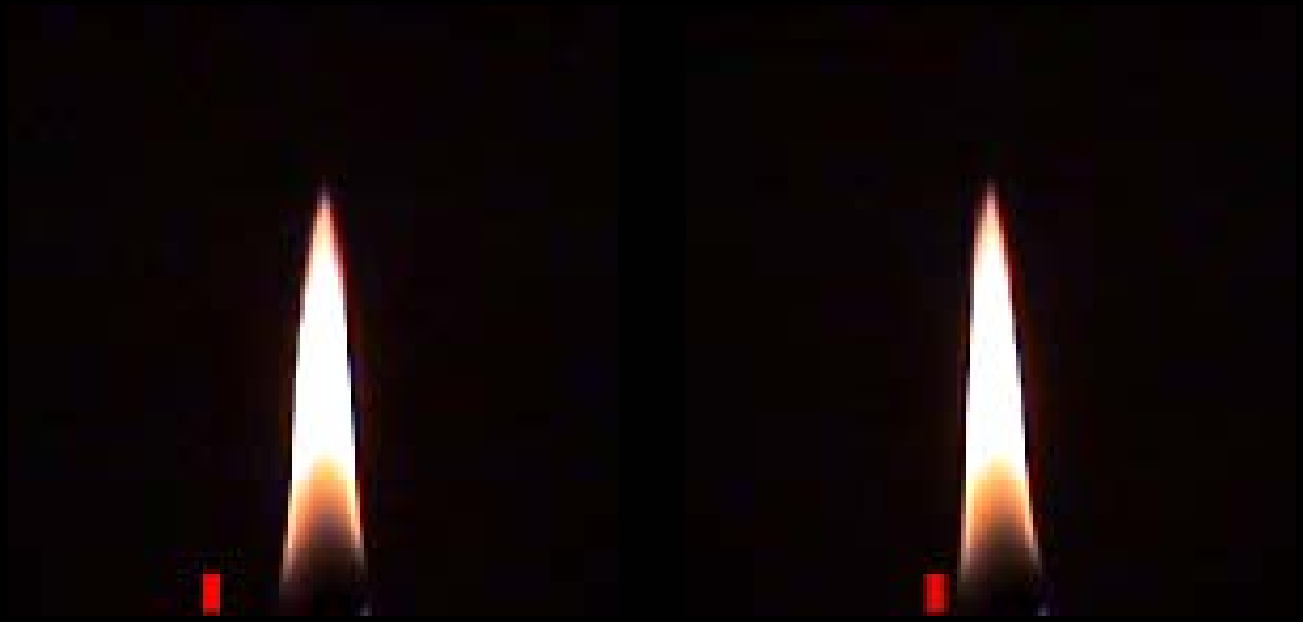
Transition costs

- Transition from i to j if successor of i is similar to j
  - Cost function: $C_{i \to j} = D_{i+1,\, j}$



Transition probabilities

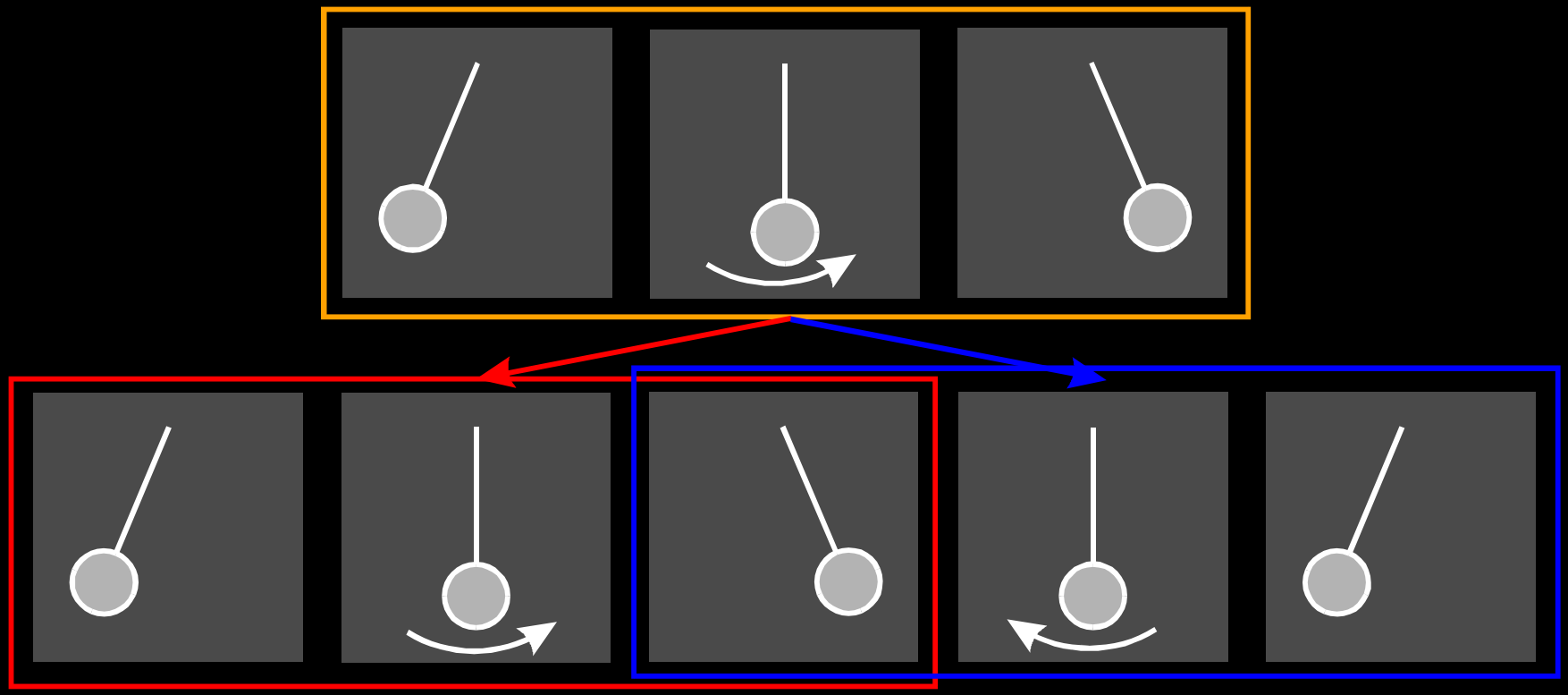
- Probability for transition $P_{i \to j}$ inversely related to cost:
  - $P_{i \to j} \sim \exp\left(- C_{i \to j} / \sigma^2 \right)$



high σ low σ

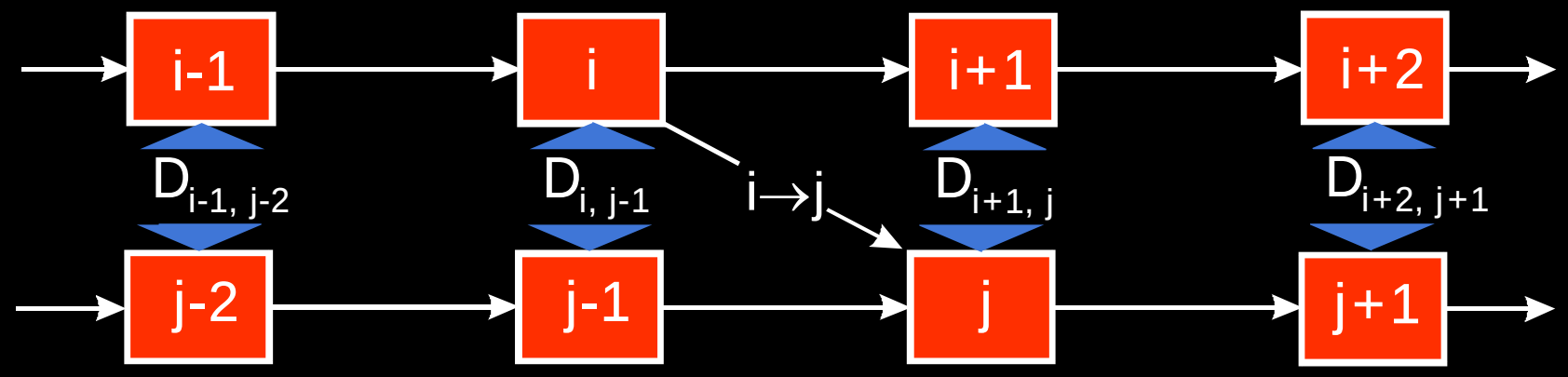
Preserving dynamics

Preserving dynamics



Preserving dynamics

- Cost for transition $i \rightarrow j$
  - $$C_{i \rightarrow j} = \sum_{k=-N}^{N-1} w_k D_{i+k+1,\, j+k}$$

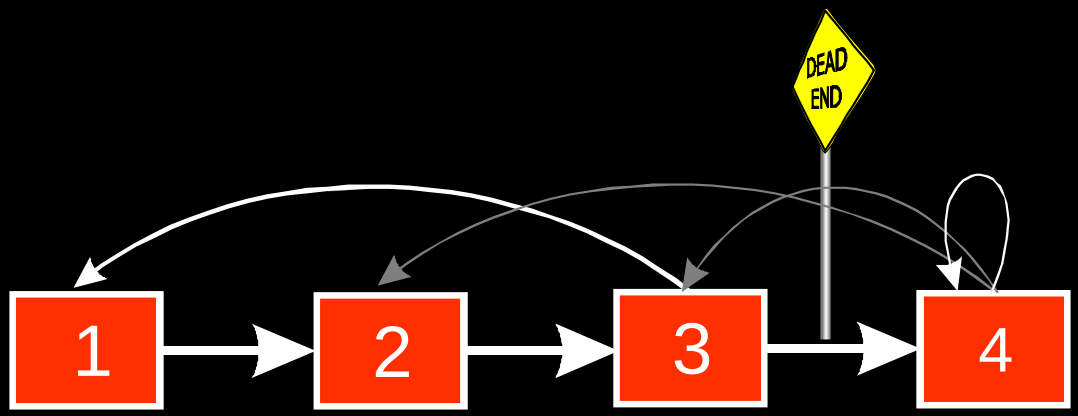


Preserving dynamics – effect

- Cost for transition $i \rightarrow j$
  - $$C_{i \rightarrow j} = \sum_{k=-N}^{N-1} w_k D_{i+k+1,\, j+k}$$

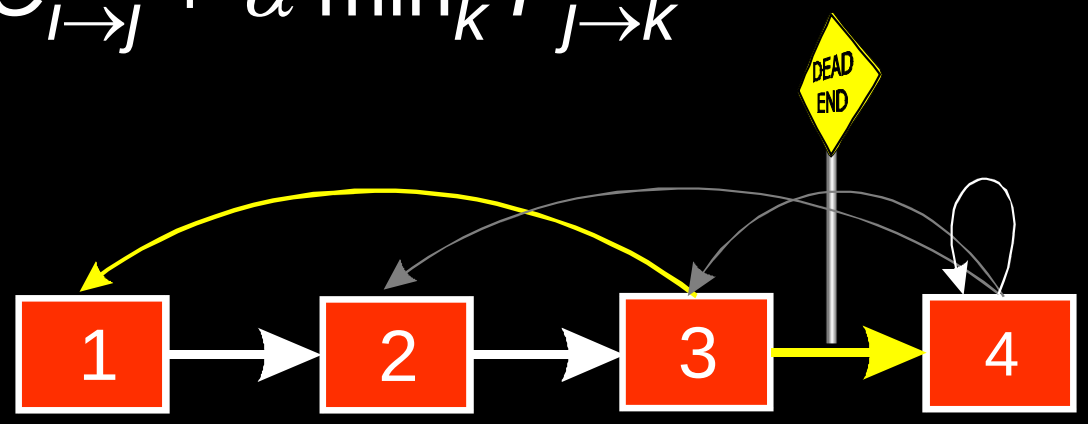
Dead ends

- No good transition at the end of sequence



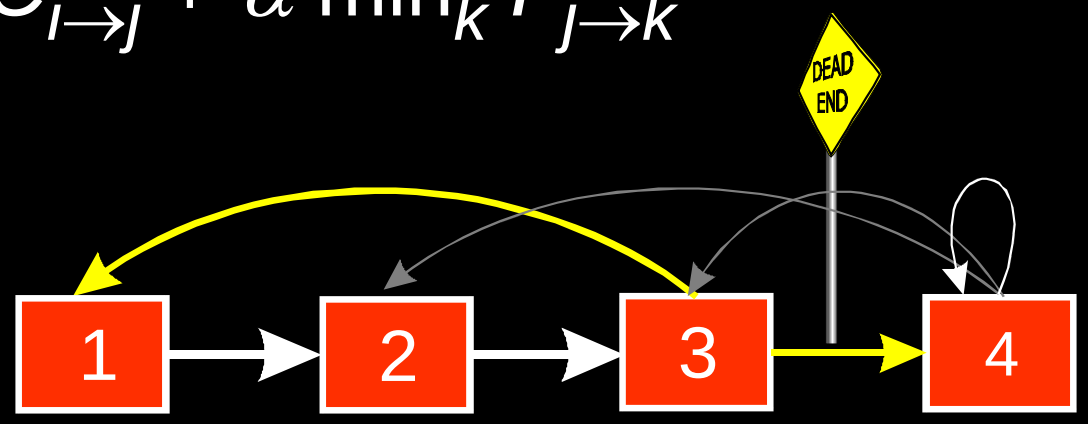
Future cost

- Propagate future transition costs backward
- Iteratively compute new cost
  - $F_{i \to j} = C_{i \to j} + \alpha \min_k F_{j \to k}$



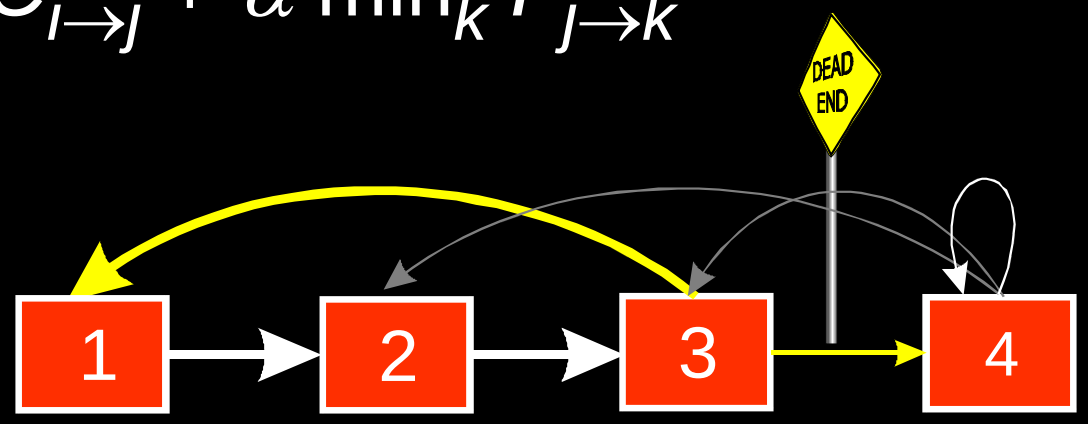
Future cost

- Propagate future transition costs backward
- Iteratively compute new cost
  - $F_{i \to j} = C_{i \to j} + \alpha \min_k F_{j \to k}$



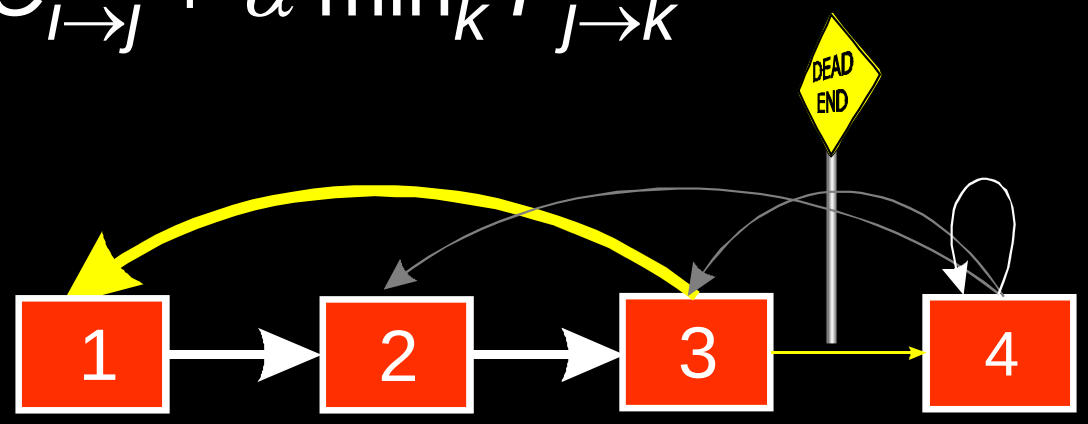
Future cost

- Propagate future transition costs backward
- Iteratively compute new cost
  - $F_{i \to j} = C_{i \to j} + \alpha \min_k F_{j \to k}$



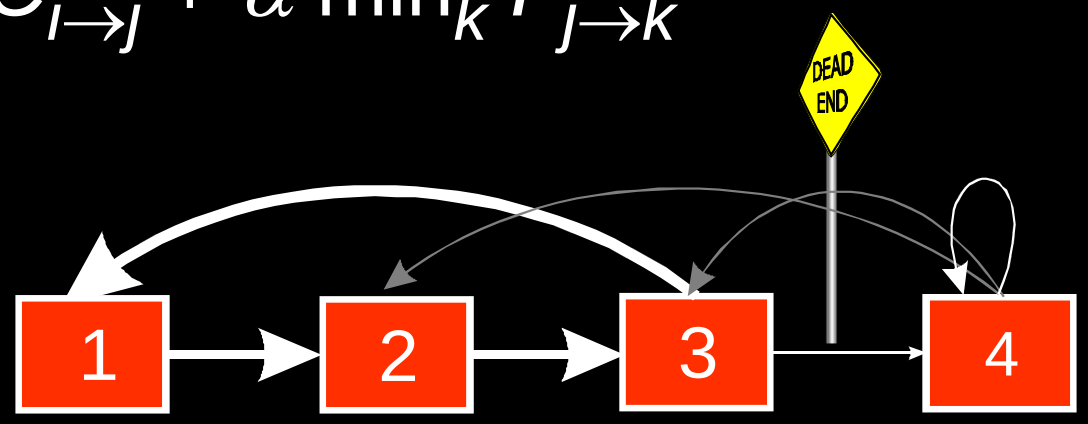
Future cost

- Propagate future transition costs backward
- Iteratively compute new cost
  - $F_{i \to j} = C_{i \to j} + \alpha \min_k F_{j \to k}$



Future cost

- Propagate future transition costs backward
- Iteratively compute new cost
  - $F_{i \to j} = C_{i \to j} + \alpha \min_k F_{j \to k}$
- Q-learning



Final result

Finding good loops

- Alternative to random transitions
- Precompute set of loops up front

Video portrait

- c.f. Harry Potter

Region-based analysis

- Divide video up into regions

- Generate a video texture for each region

User-controlled video textures

slow variable fast

User selects target frame range

Video-based animation

- Like sprites computer games
- Extract sprites from real video
- Interactively control desired motion

©1985 Nintendo of America Inc.

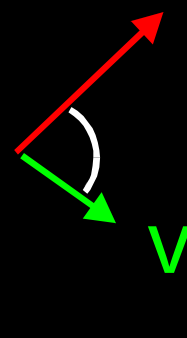
Video sprite extraction

blue screen matting
and velocity estimation

Video sprite control

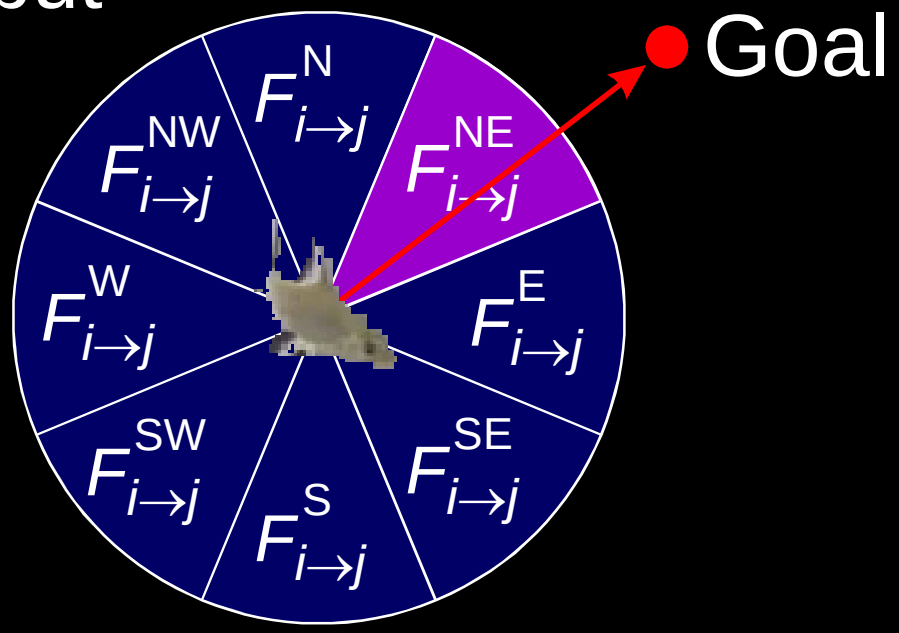
- Augmented transition cost:

$$C^{\text{Animation}}_{i\to j} = \alpha \underbrace{C_{i\to j}}_{\text{Similarity term}} + \beta \underbrace{\text{angle}}_{\text{Control term}}$$



Video sprite control

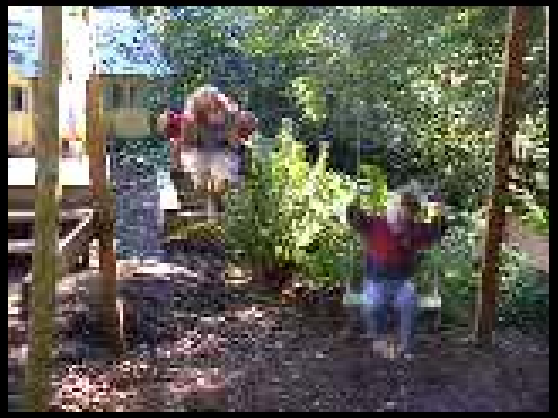
- Need future cost computation
- Precompute future costs for a few angles.
- Switch between precomputed angles according to user input
- [GIT-GVU-00-11]



Interactive fish

Summary / Discussion

- Some things are relatively easy



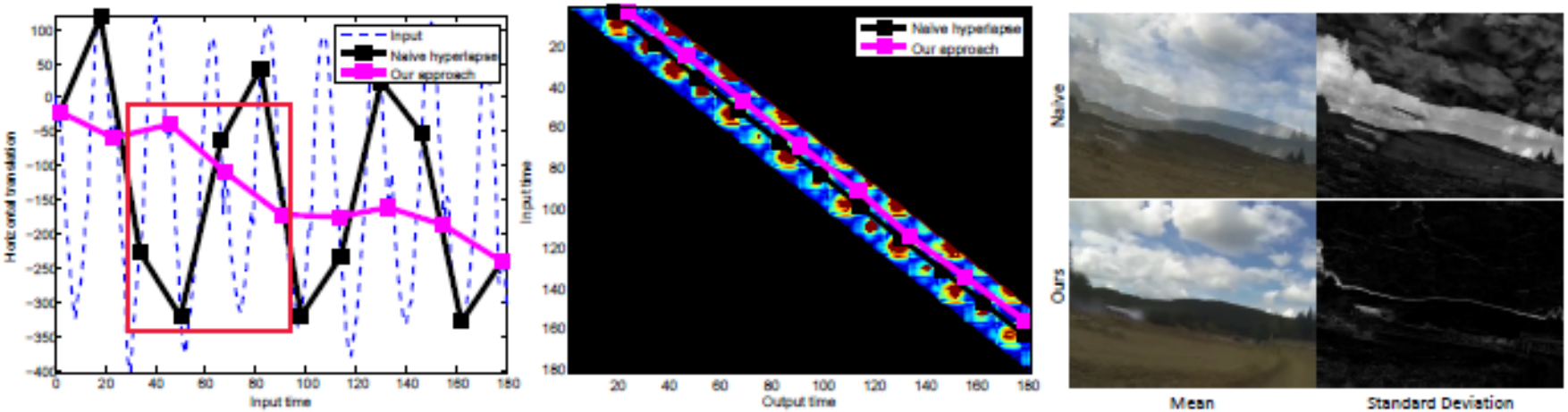
Discussion

- Some are hard

Hyperlapse Videos

Real-Time Hyperlapse Creation via Optimal Frame Selection

Neel Joshi Wolf Kienzle Mike Toelle Matt Uyttendaele Michael F. Cohen
Microsoft Research



https://www.youtube.com/watch?v=Wt_Y04xn84M

“Amateur” by Lasse Gjertsen

http://www.youtube.com/watch?v=JzqumbhfxRo

similar idea:

http://www.youtube.com/watch?v=MsBMG-p1HDM&feature=share&list=PLFFD733D0FF425290

Texture

- Texture depicts spatially repeating patterns
- Many natural phenomena are textures

radishes

rocks

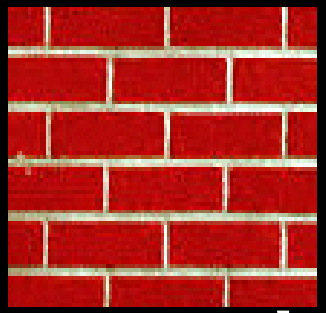
yogurt

Texture Synthesis

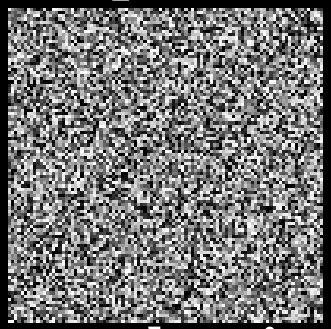
- Goal of Texture Synthesis: create new samples of a given texture
- Many applications: virtual environments, hole-filling, texturing surfaces

The Challenge

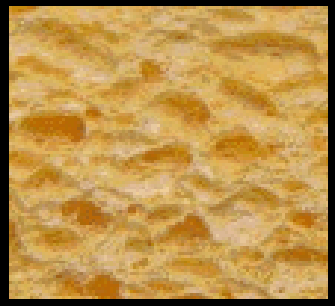
- Need to model the whole spectrum: from repeated to stochastic texture



repeated

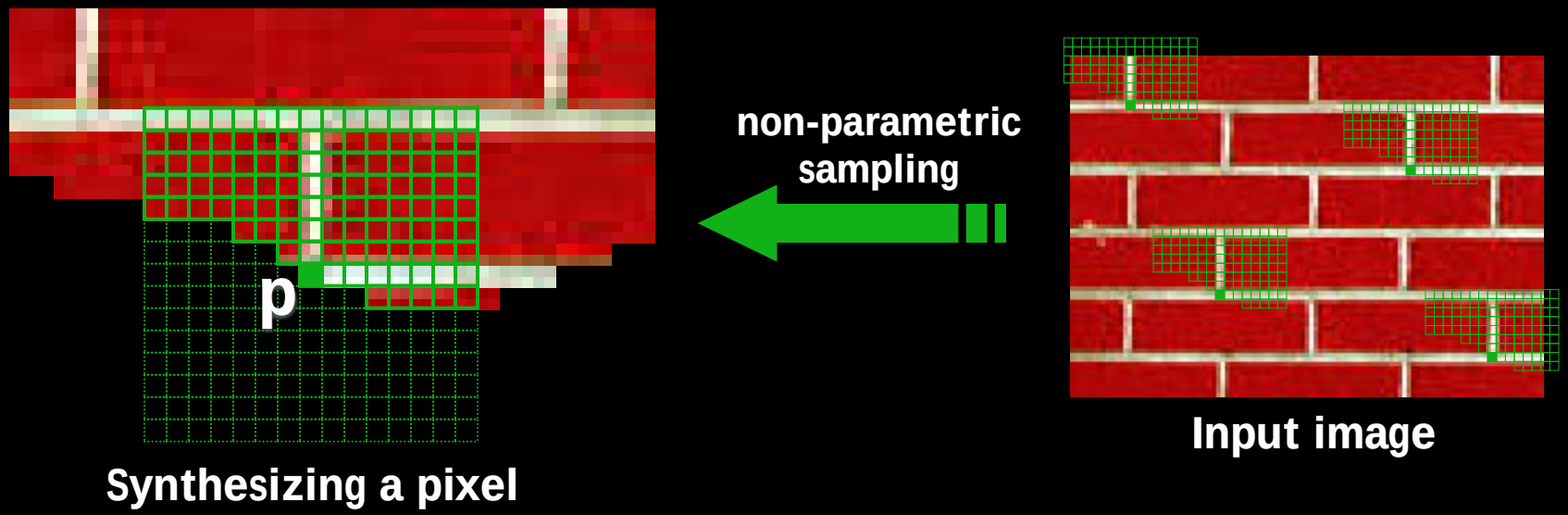


stochastic



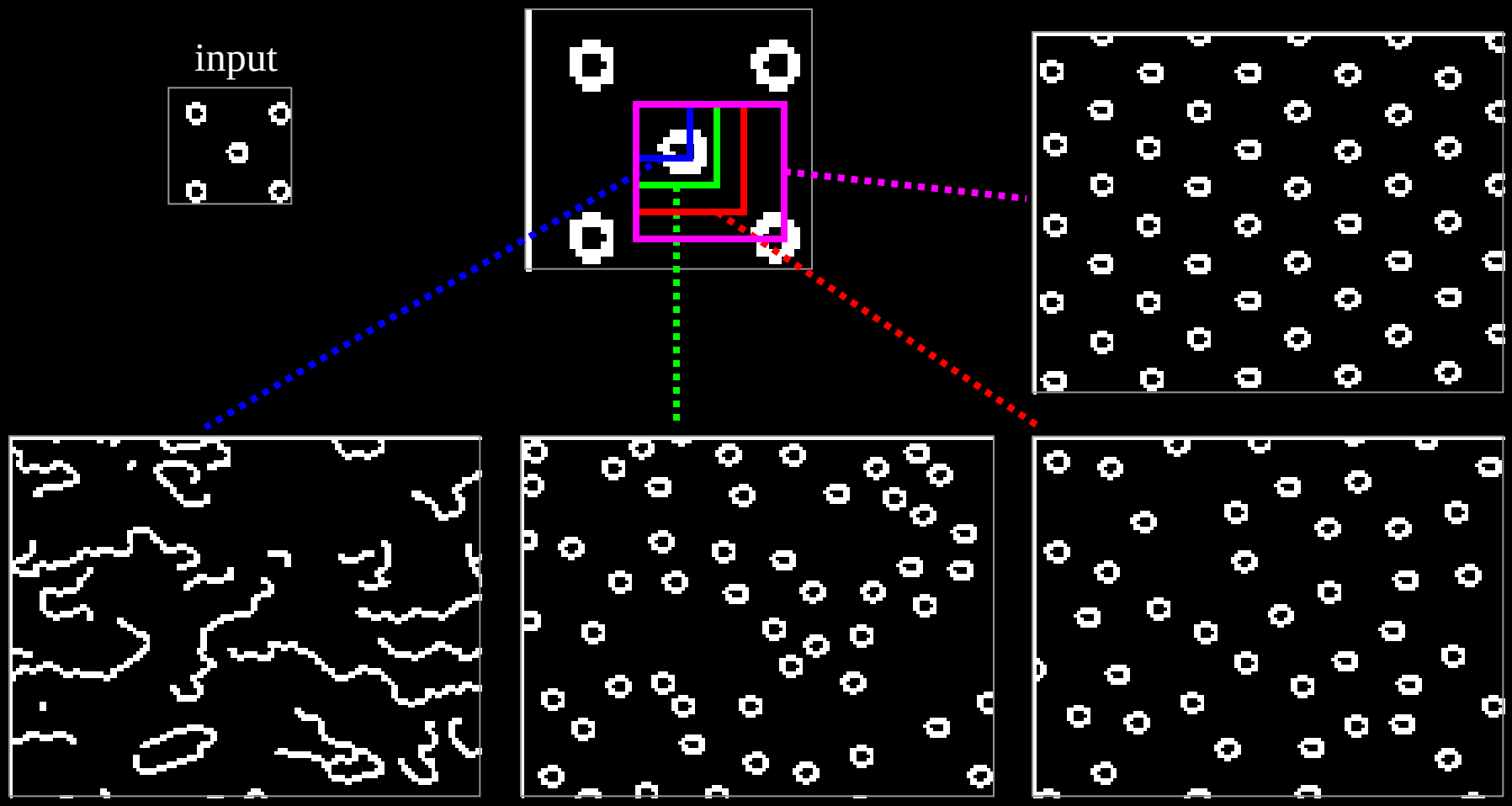
Both?

Efros & Leung Algorithm



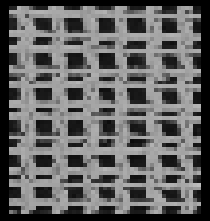
- Assuming Markov property, compute P(**p**|N(**p**))
  - Building explicit probability tables infeasible
  - Instead, we *search the input image* for all similar neighborhoods — that's our pdf for **p**
  - To sample from this pdf, just pick one match at random

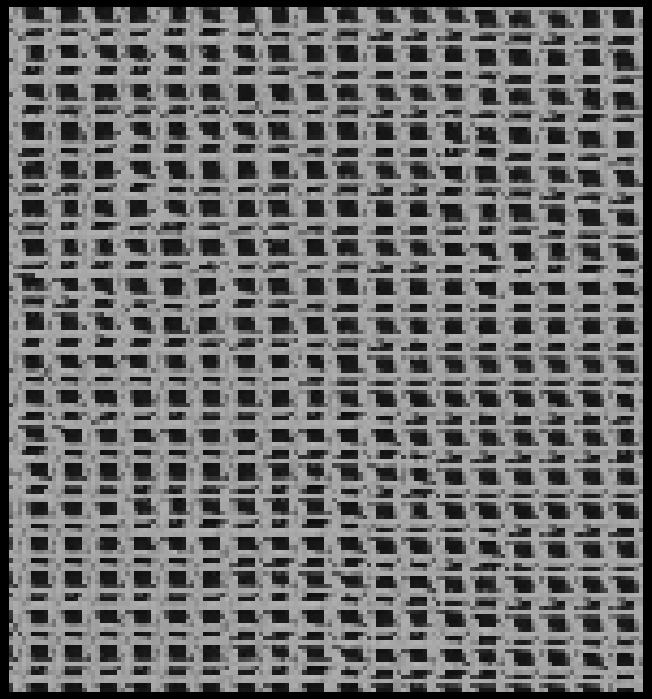
Neighborhood Window



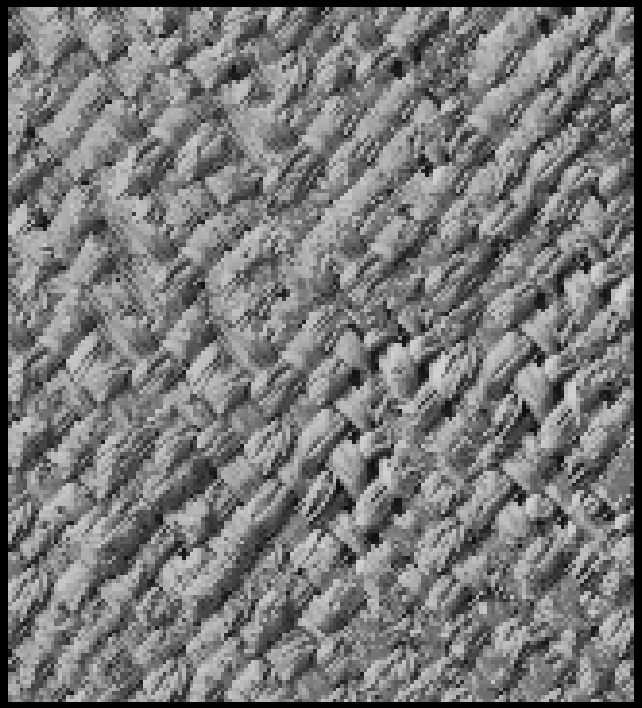
Synthesis Results

french canvas



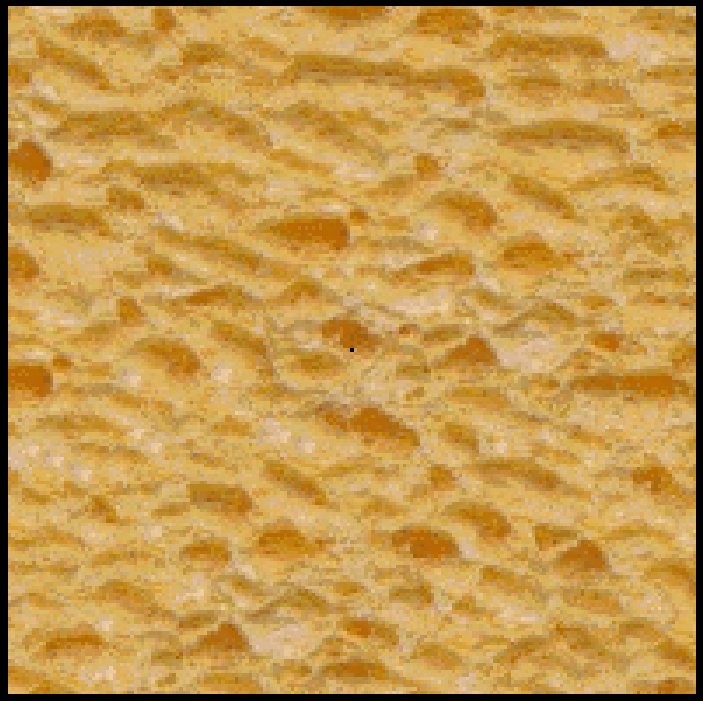


rafia weave

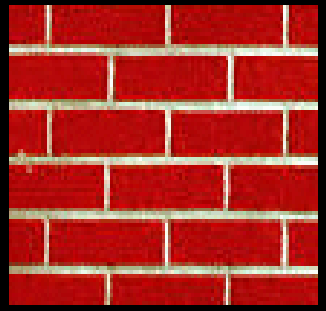


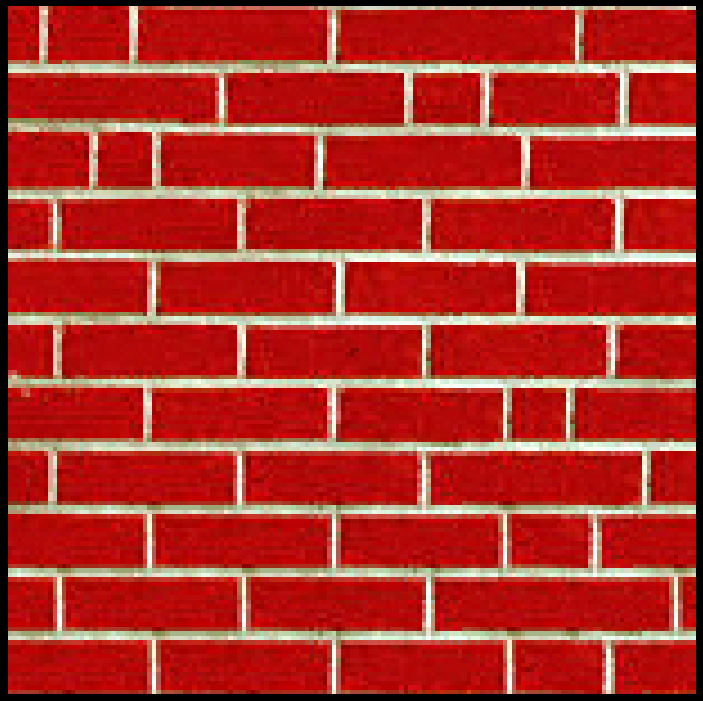
More Results

white bread

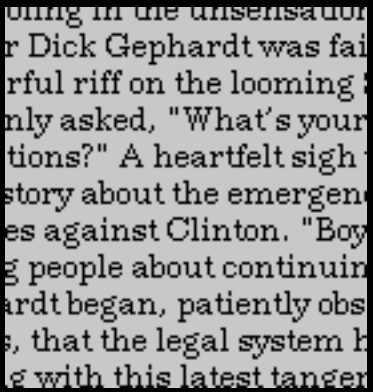


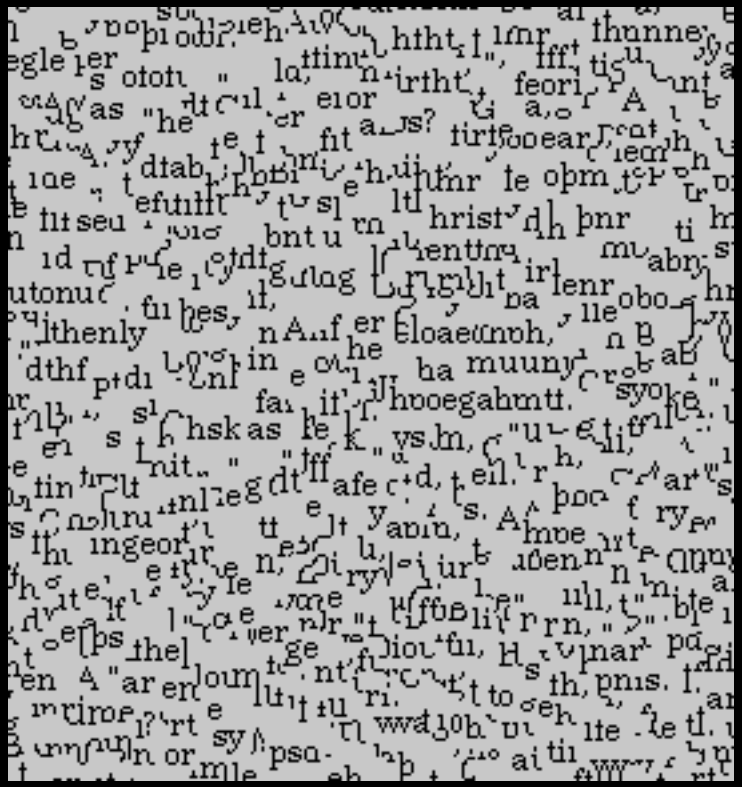
brick wall

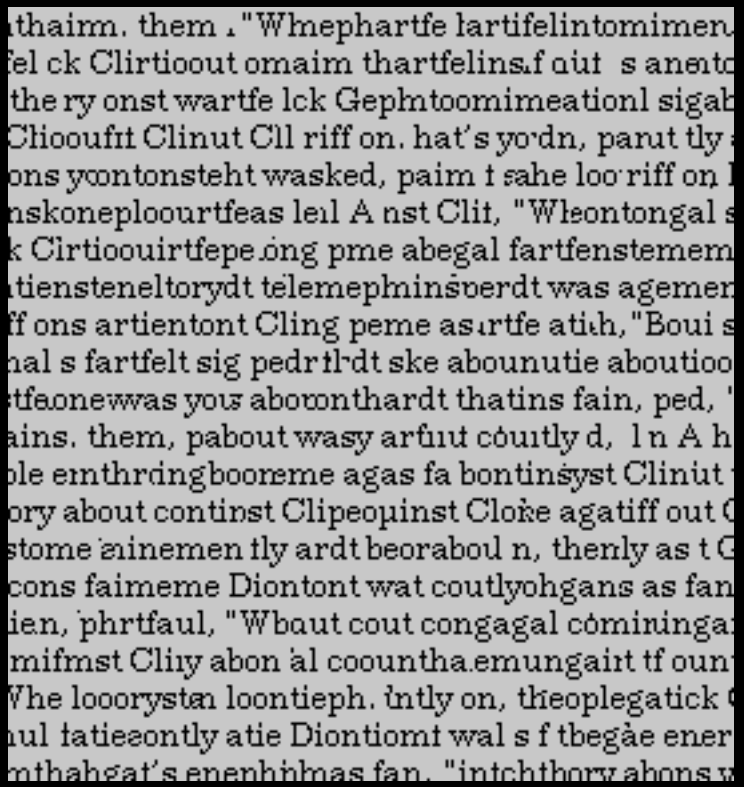




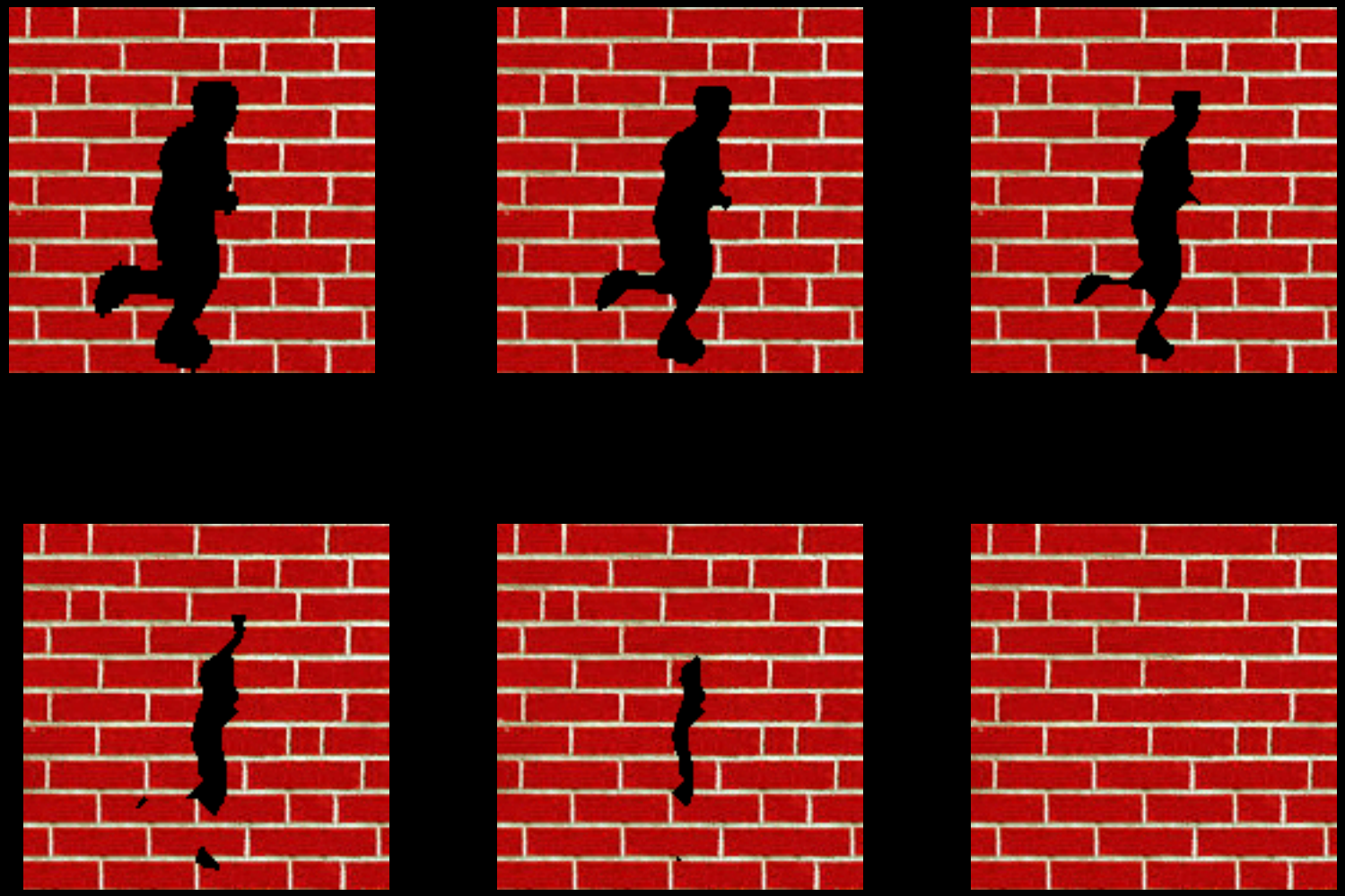
Homage to Shannon



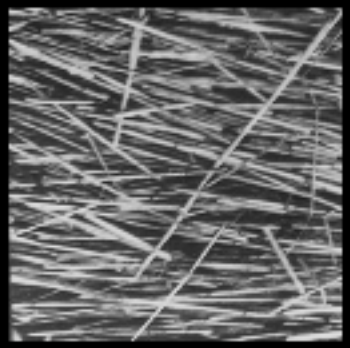


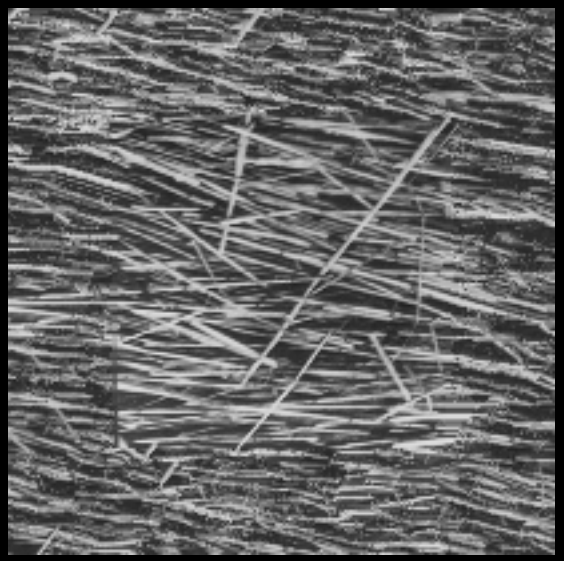


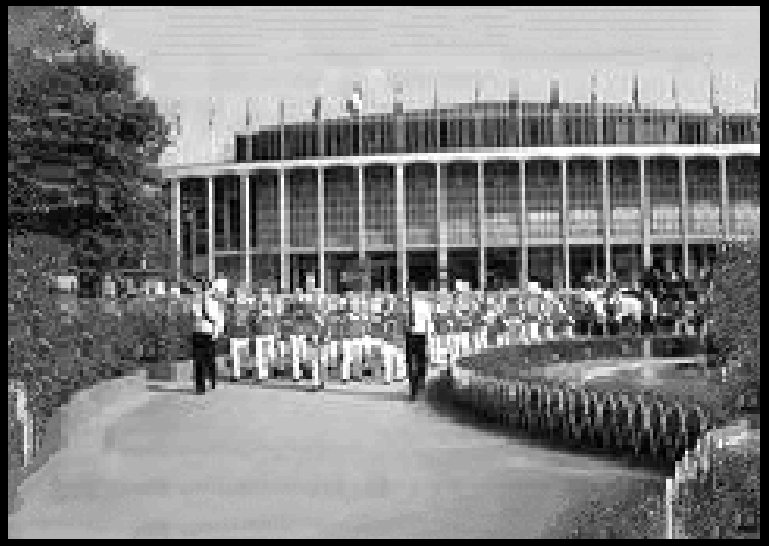
Hole Filling



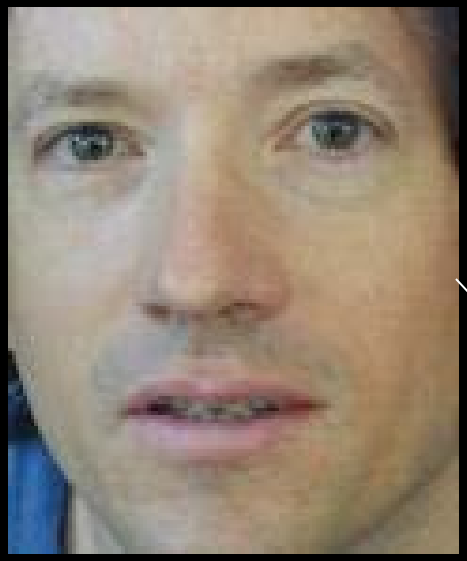
Extrapolation







Texture Transfer

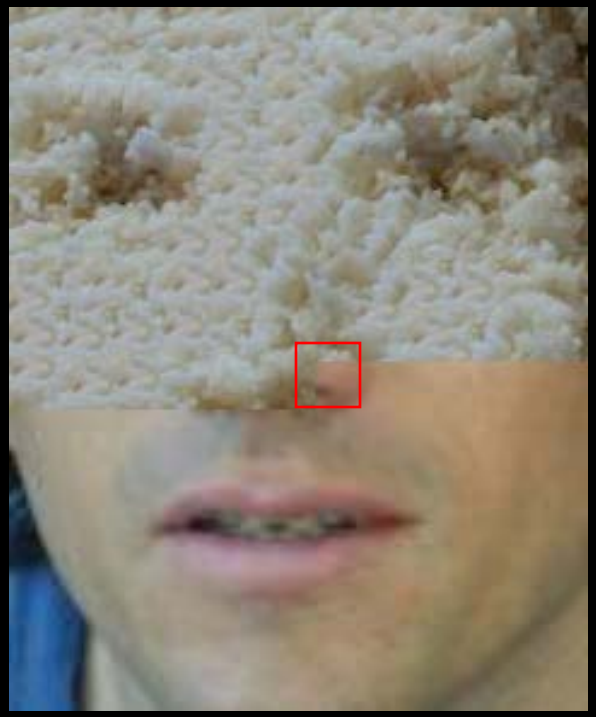


Constraint

Texture sample

Texture Transfer

- Take the texture from one image and “paint” it onto another object



Same as texture synthesis, except an additional constraint:

1. Consistency of texture
2. Similarity to the image being “explained”

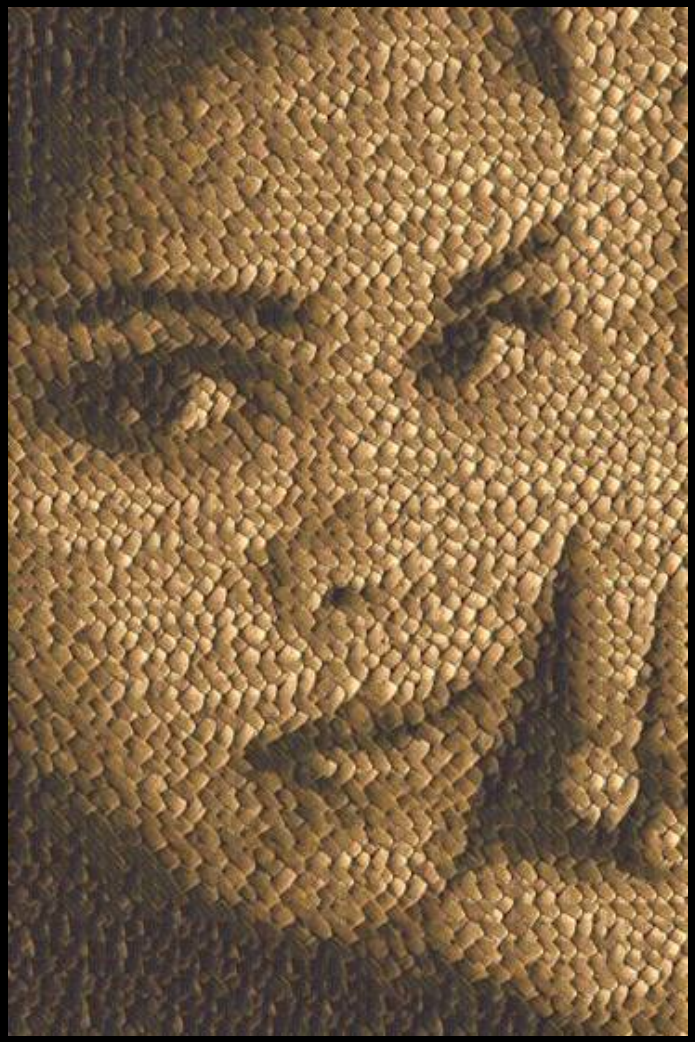


Image Analogies

Aaron Hertzmann[1,2]

Chuck Jacobs[2]

Nuria Oliver[2]

Brian Curless[3]

David Salesin[2,3]

[1]New York University

[2]Microsoft Research

[3]University of Washington

Image Analogies

Goal: Process an image by example

A

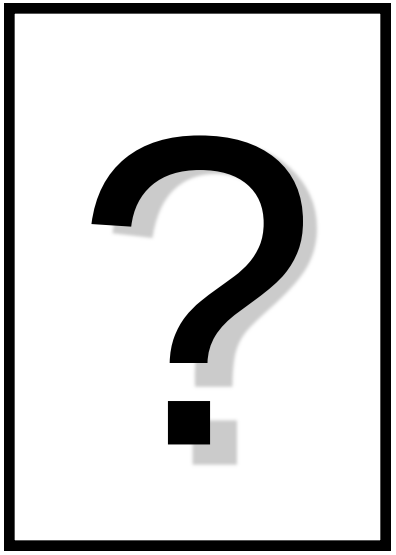
:

A’

::

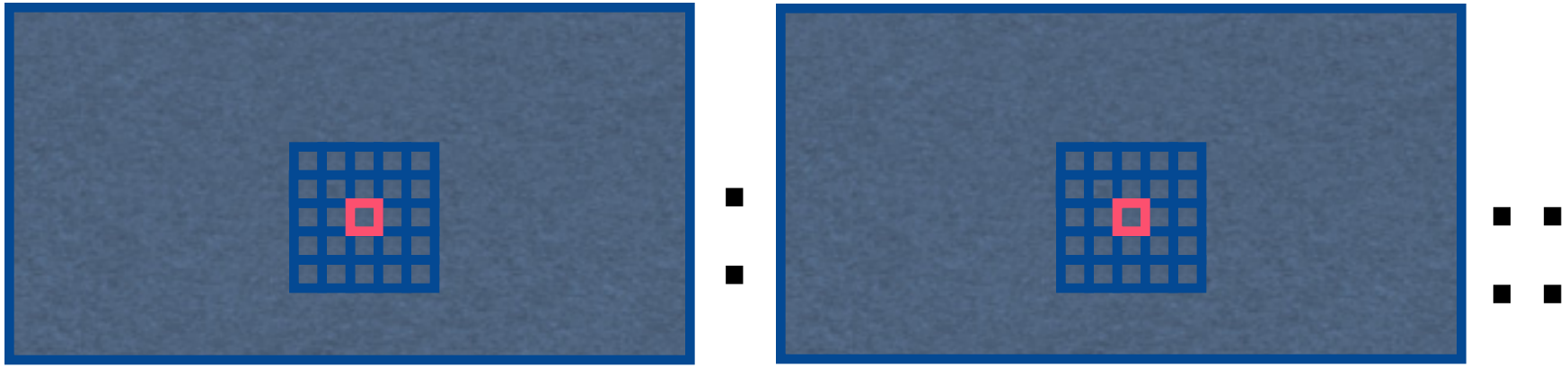
B

:

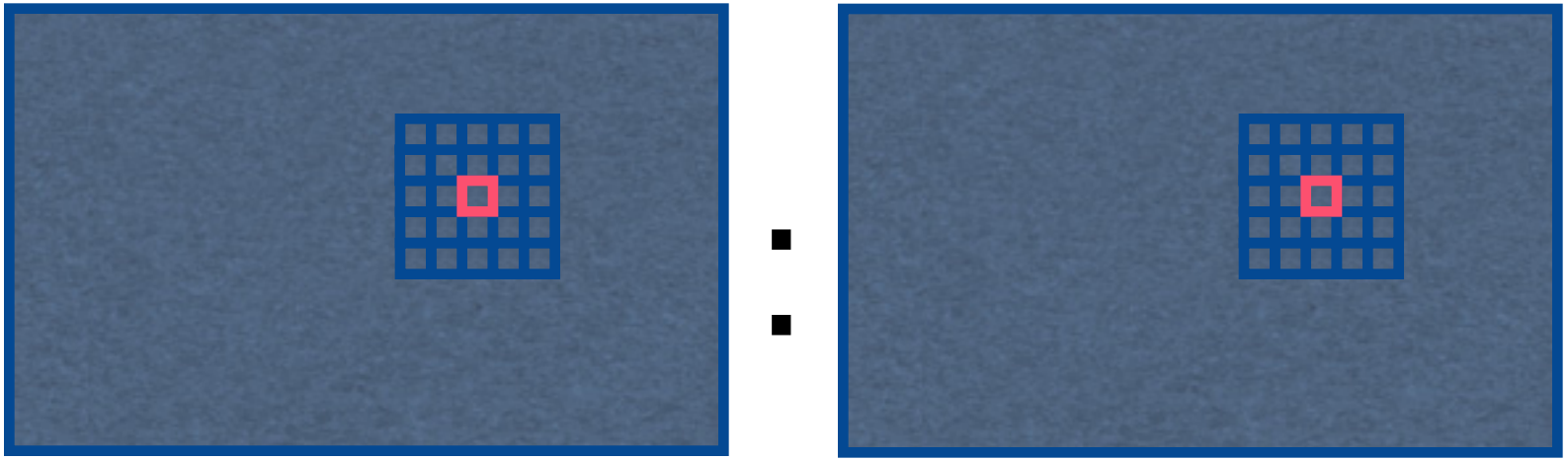


B’

Hertzmann et al. SIGGRAPH 2

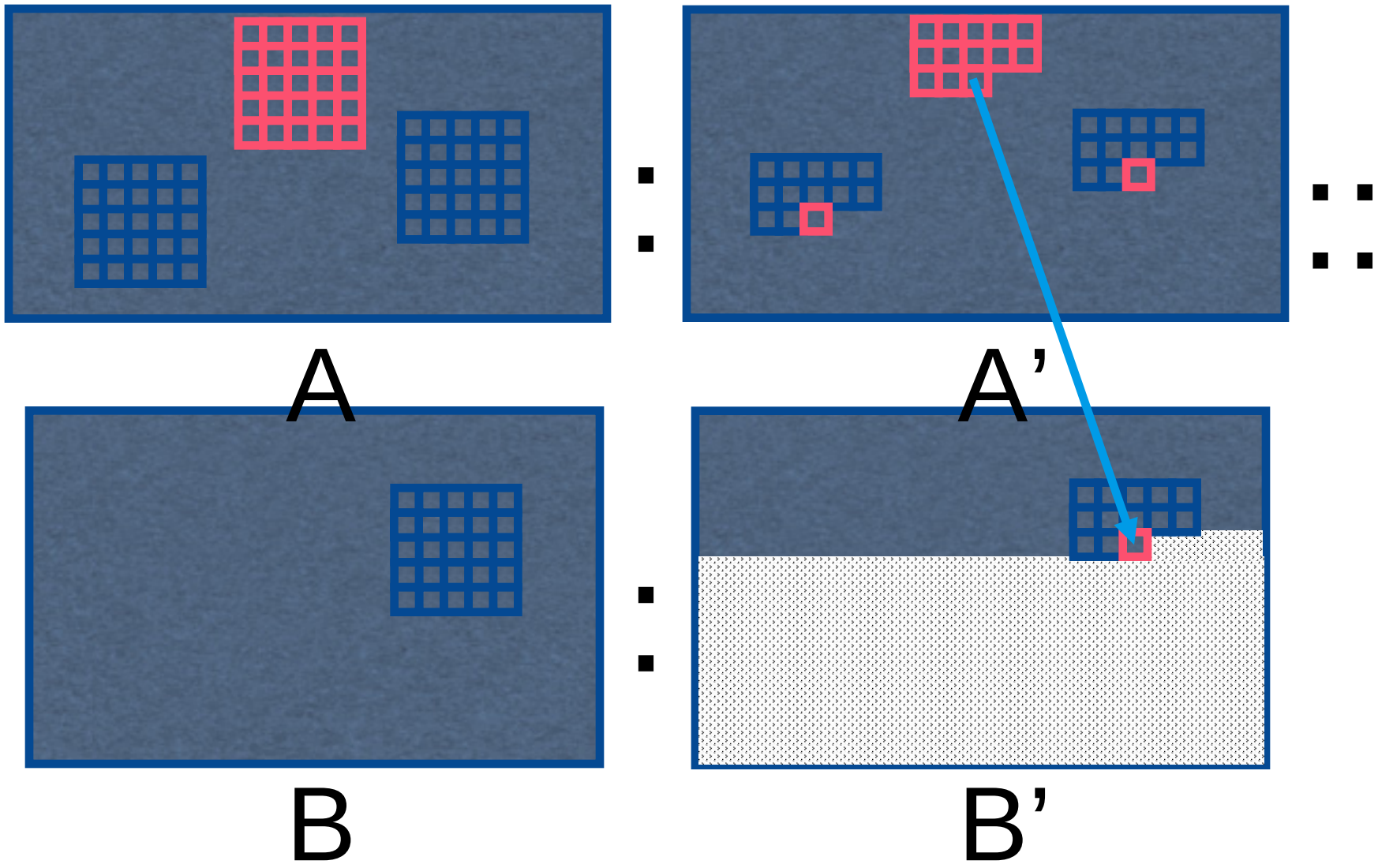


A A’



B B’

Hertzmann et al. SIGGRAPH 2



Hertzmann et al. SIGGRAPH 2

Blur

A

A’

B

B’

Hertzmann et al. SIGGRAPH 2

Superresolution

Hertzmann et al. SIGGRAPH

Texture transfer

(same texture)

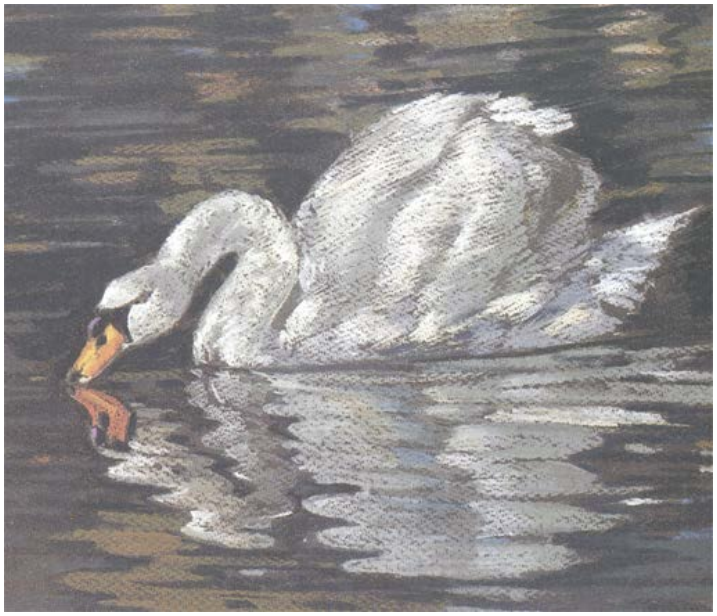
A A’

B

Closer to texture

Closer to photo

B’s



Color transfer

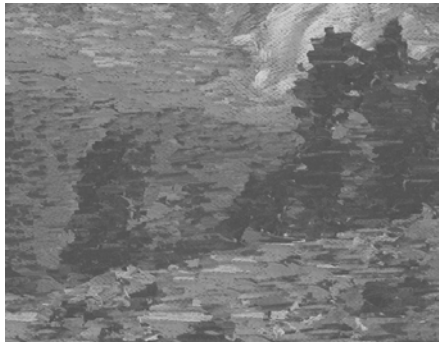
Input photo

Input luminance

Example luminance

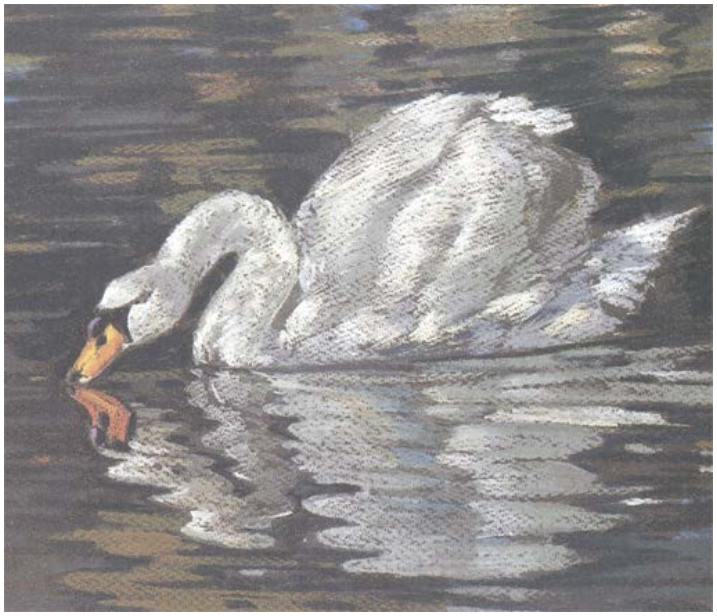
Input colors

+



Output luminance

Output image

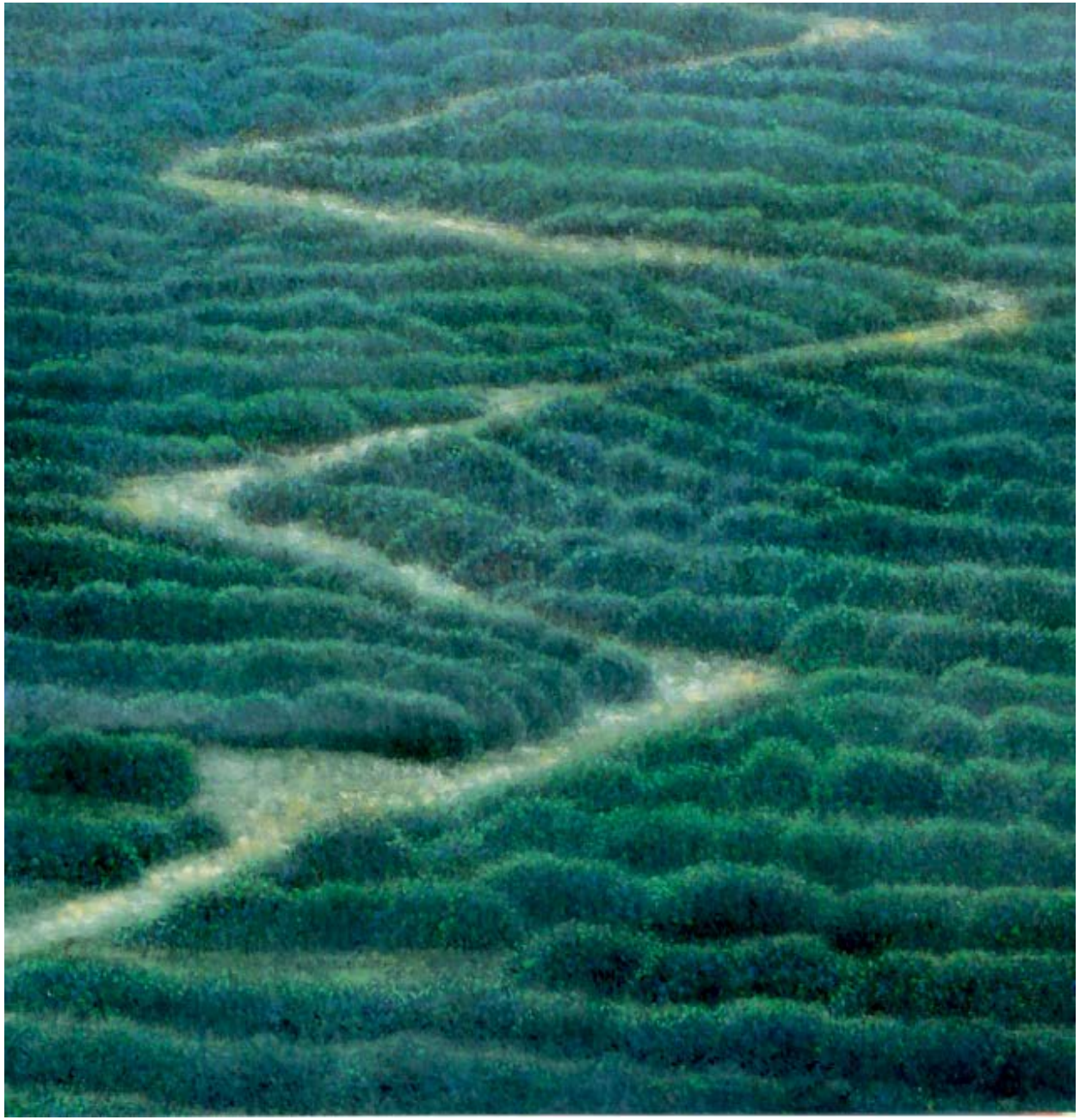


A

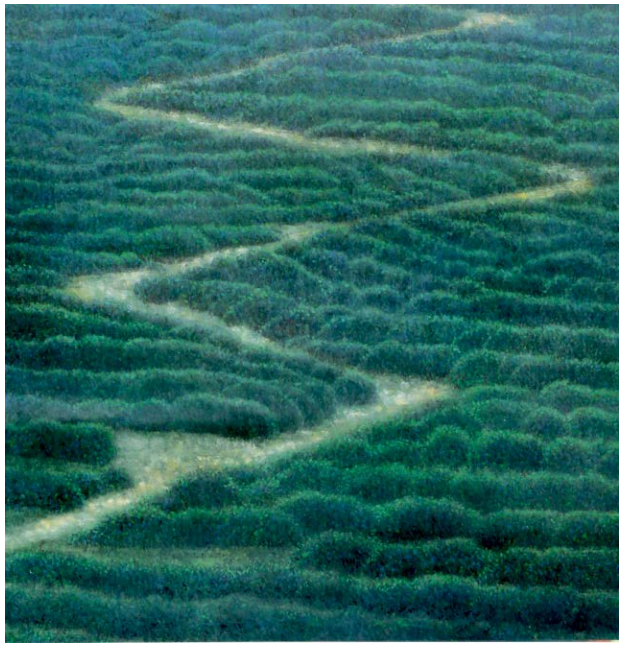
A’

B

B’



:



A

A’

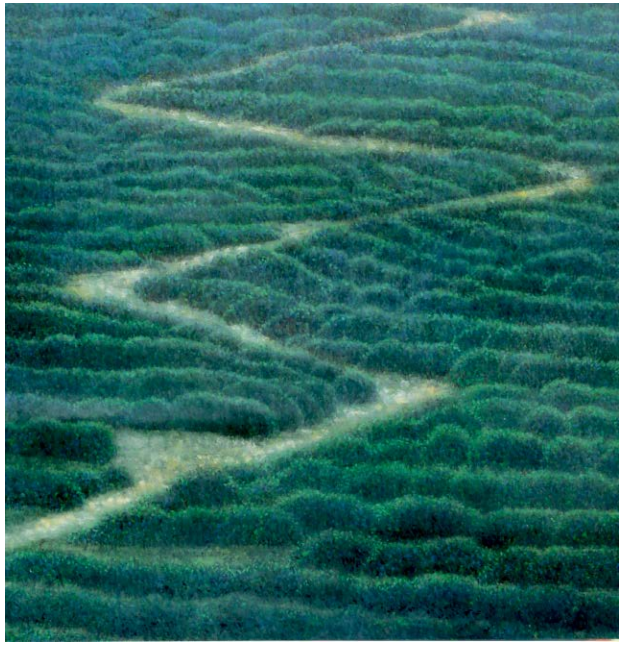
::

:

B

B’

:

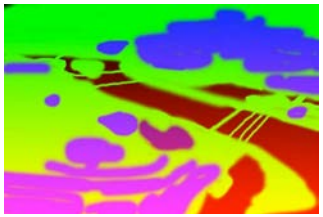


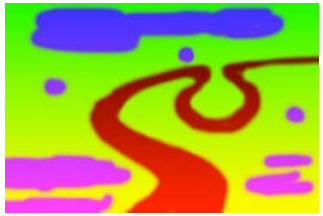
A

A’

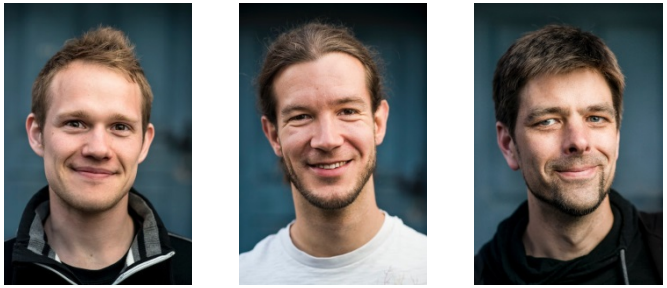
:

:

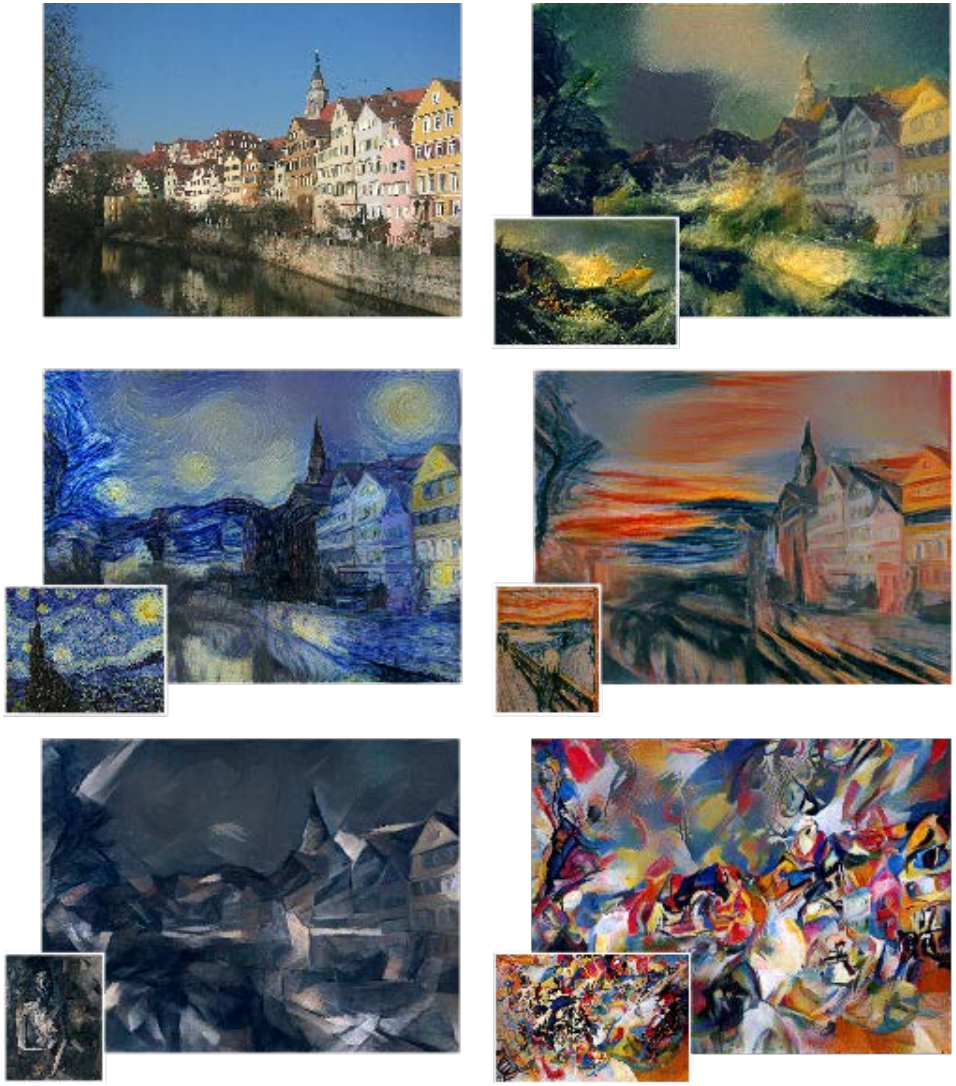


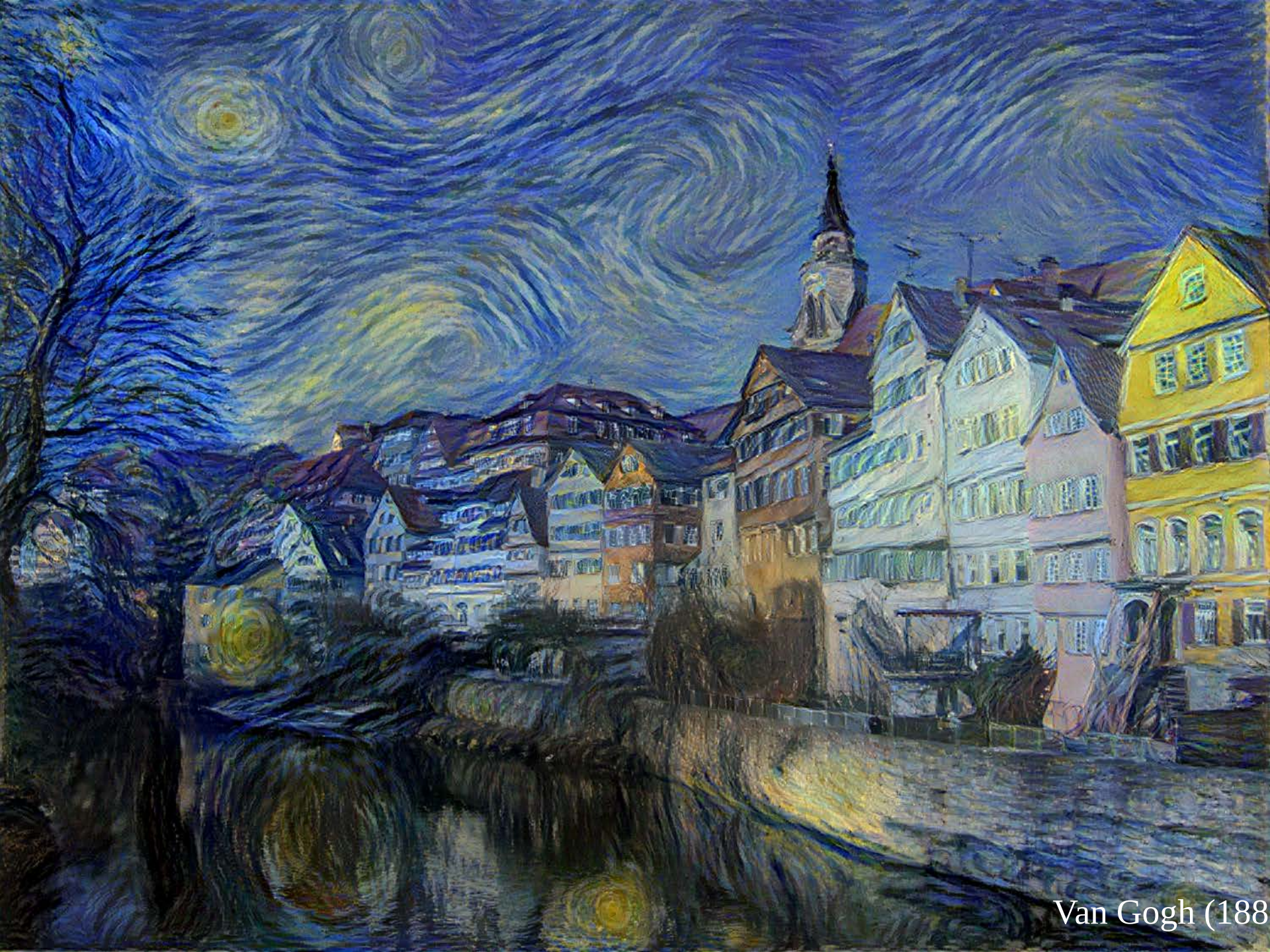


A Neural Algorithm of Artistic Style



Gatys, Ecker, Bethge (arXiv 2015)




Van Gogh (188

Picasso (1910)

Munch (1893)

Turner (1805)

Kandinsky (191

Early Vision Texture Models

Linear filter bank

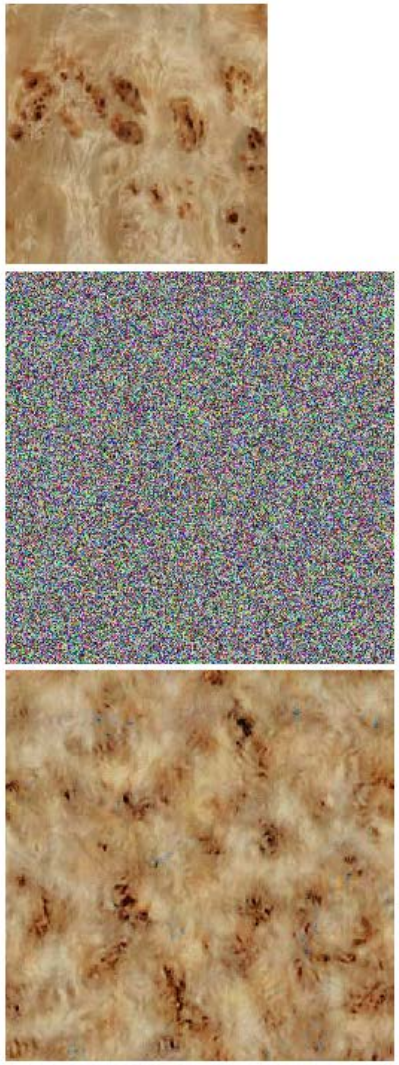
Heeger & Bergen (1995)
Portilla & Simoncelli (2000)

Heeger & Bergen, SIGGRAPH‘95

Start with a noise image as output

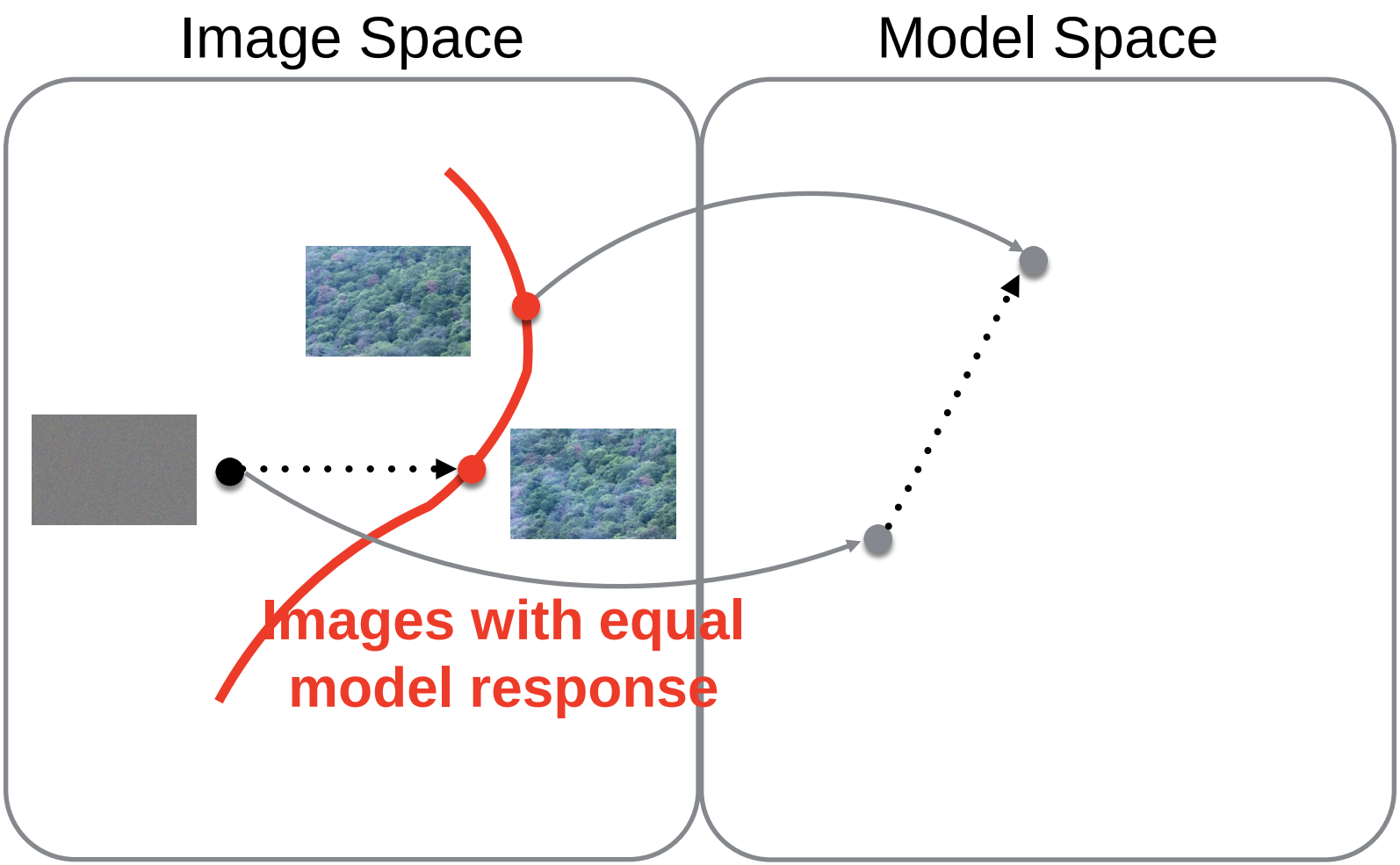
Main loop:

- Match pixel histogram of output image to input
- Decompose input and output images using multi-scale filter bank (Steerable Pyramid)
- Match subband histograms of input and output pyramids
- Reconstruct input and output images (collapse the pyramids)



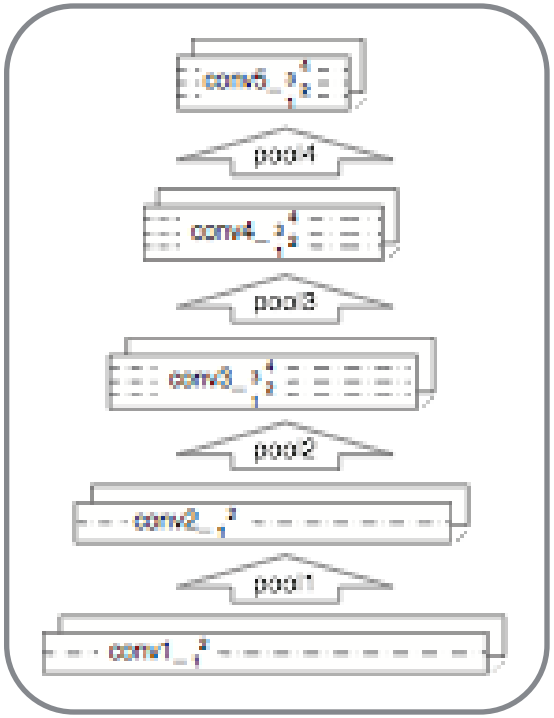
Heeger, Bergen, Pyramid-based texture analysis/synthesis, SIGGRAPH 1995

Texture Synthesis



Portilla & Simoncelli (2000)

Convolutional Neural Network Texture Model

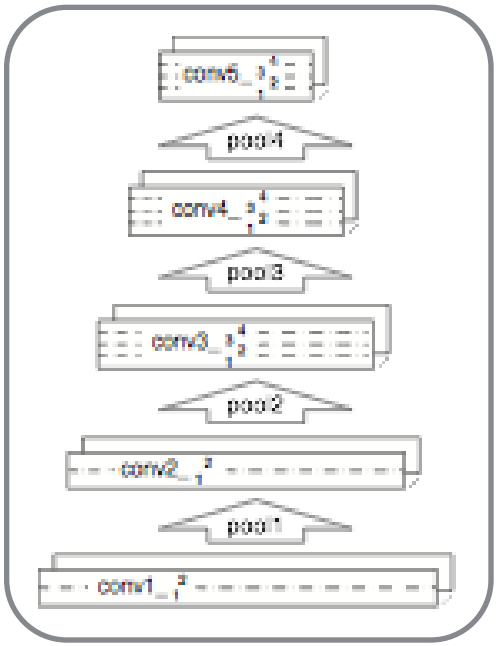


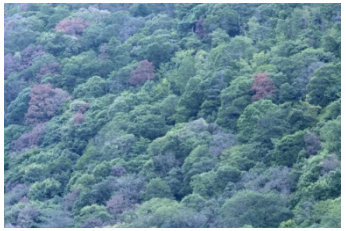
2.3
7.1
3.8
⋮

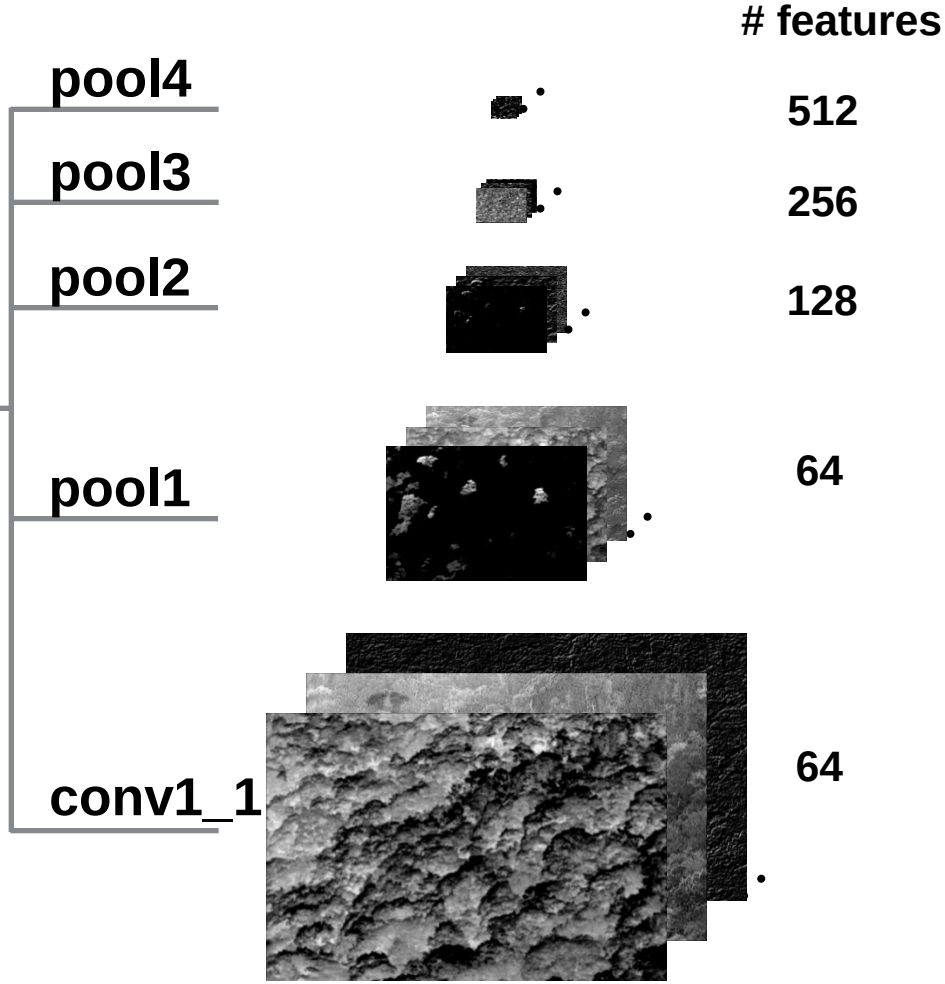
Convolutional Neural Network

Gatys et al. (NIPS 2015)

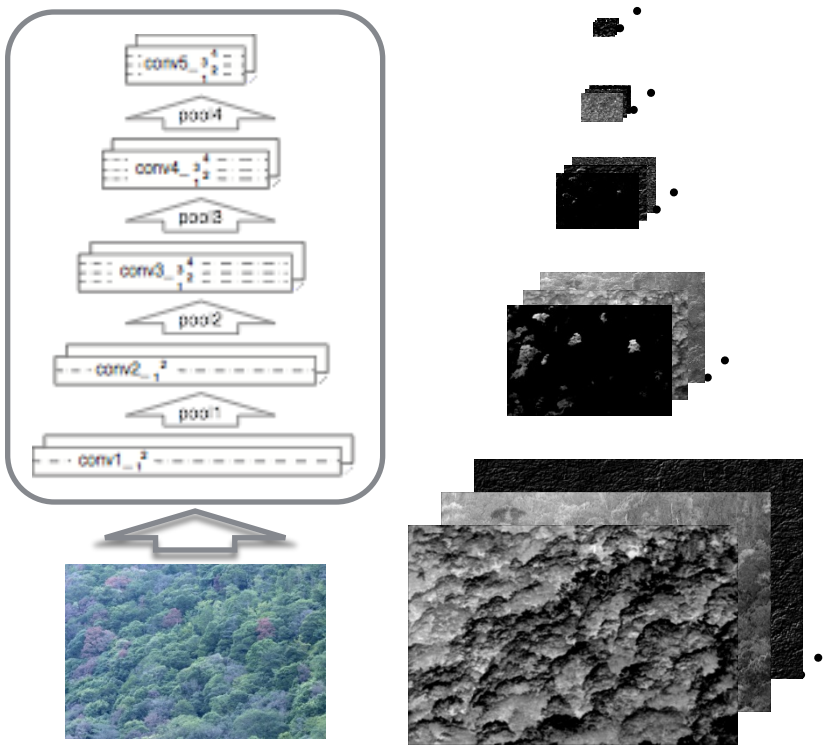
CNN - Multiscale Filter Bank







CNN - Texture Features



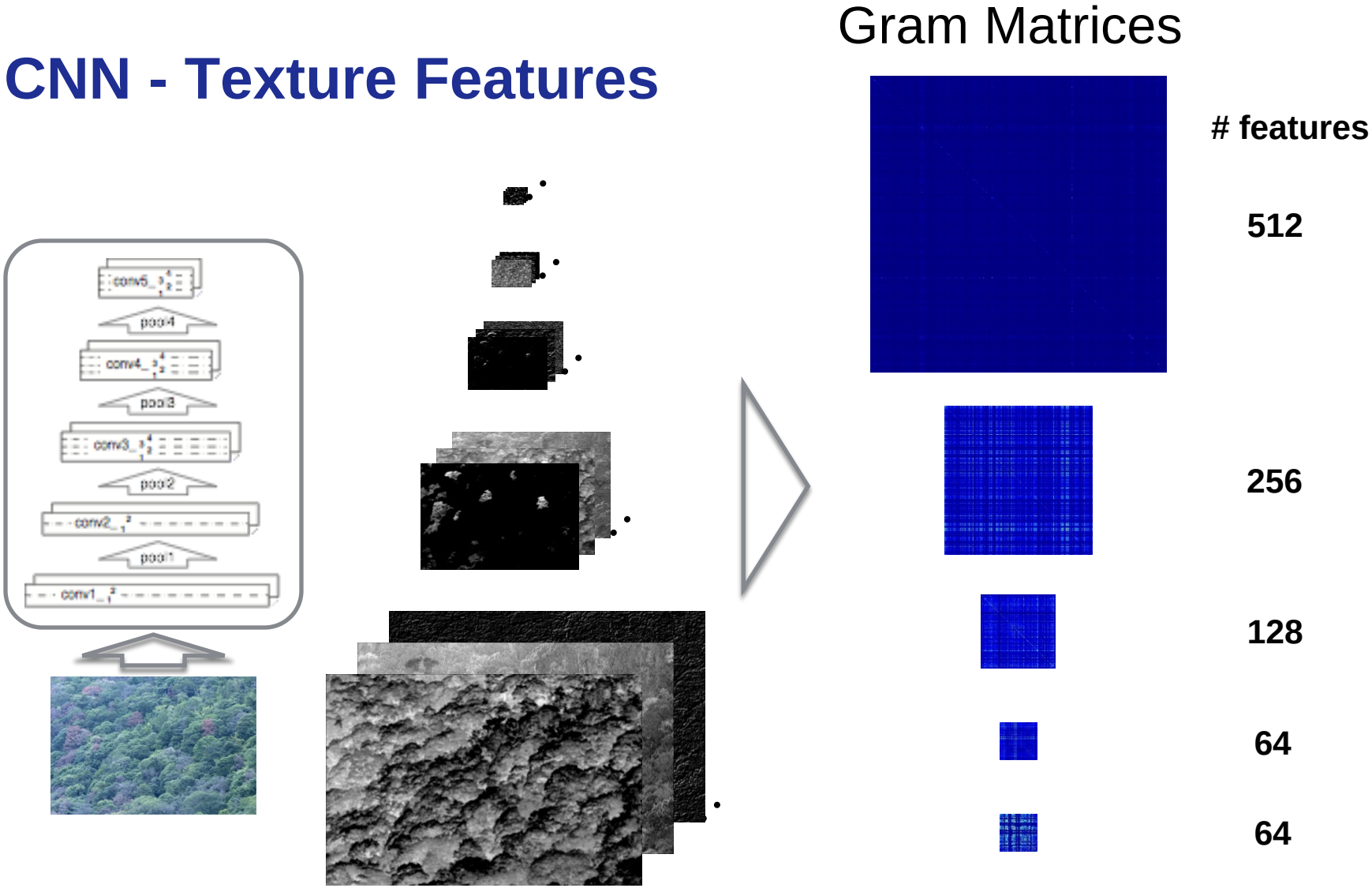
$$F = \left[\bar{f}_1, \bar{f}_2, \bar{f}_3, ..., \bar{f}_N\right]^T$$

$$G = FF^T$$

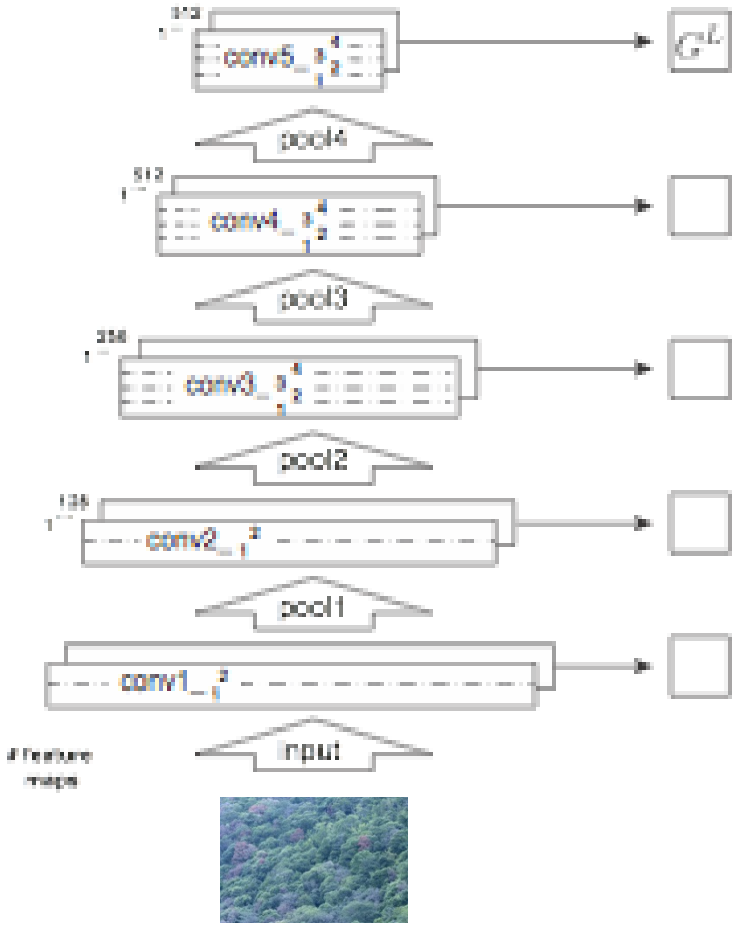
$$= \begin{pmatrix} \langle \bar{f}_1, \bar{f}_1 \rangle & \cdots & \langle \bar{f}_1, \bar{f}_N \rangle \\ \langle \bar{f}_2, \bar{f}_1 \rangle & & \vdots \\ \vdots & \ddots & \vdots \\ \langle \bar{f}_N, \bar{f}_1 \rangle & \cdots & \langle \bar{f}_N, \bar{f}_N \rangle \end{pmatrix}$$

$$\langle \bar{f}_i, \bar{f}_j \rangle = \sum_k F_{ik} F_{jk}$$

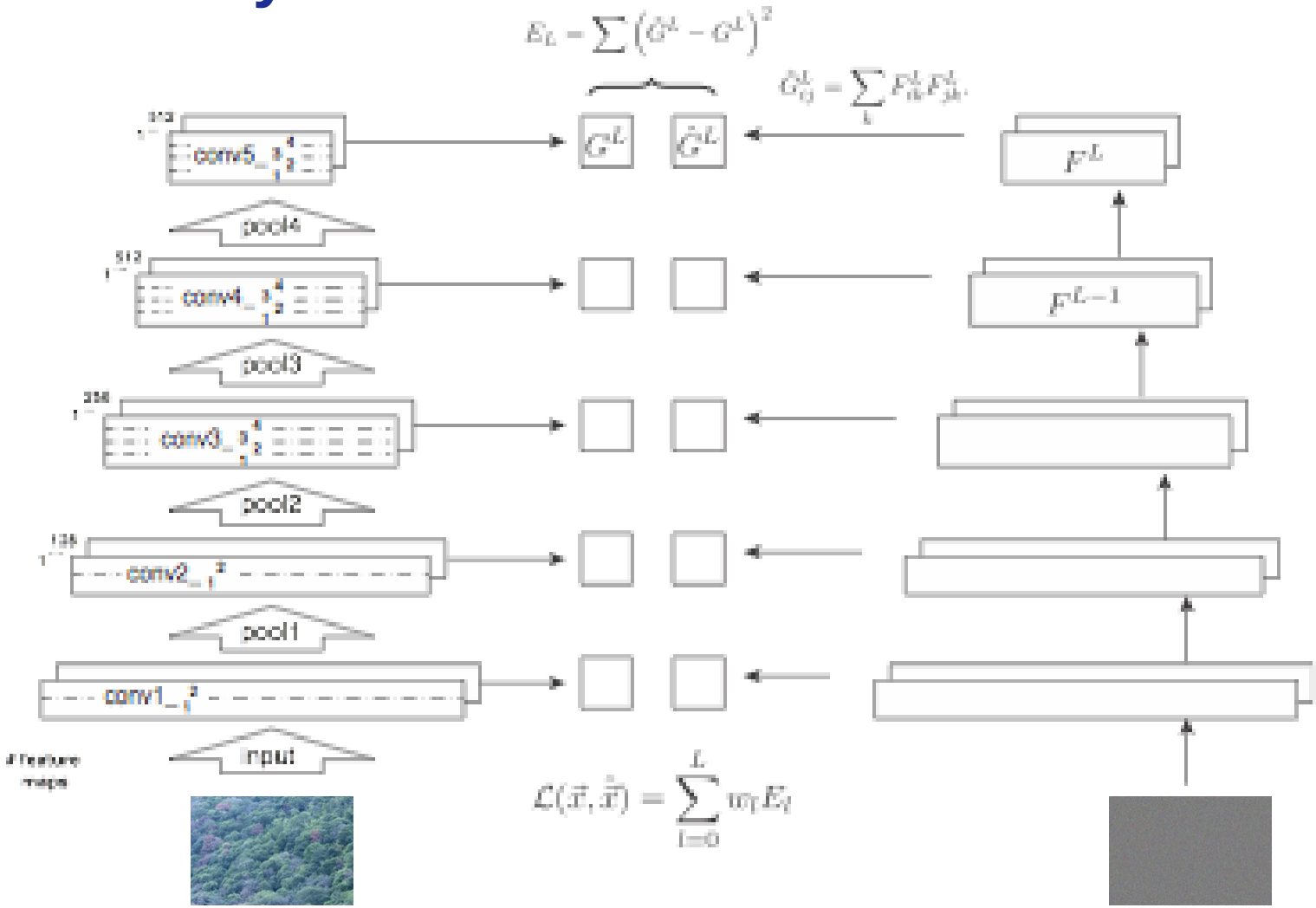
CNN - Texture Features



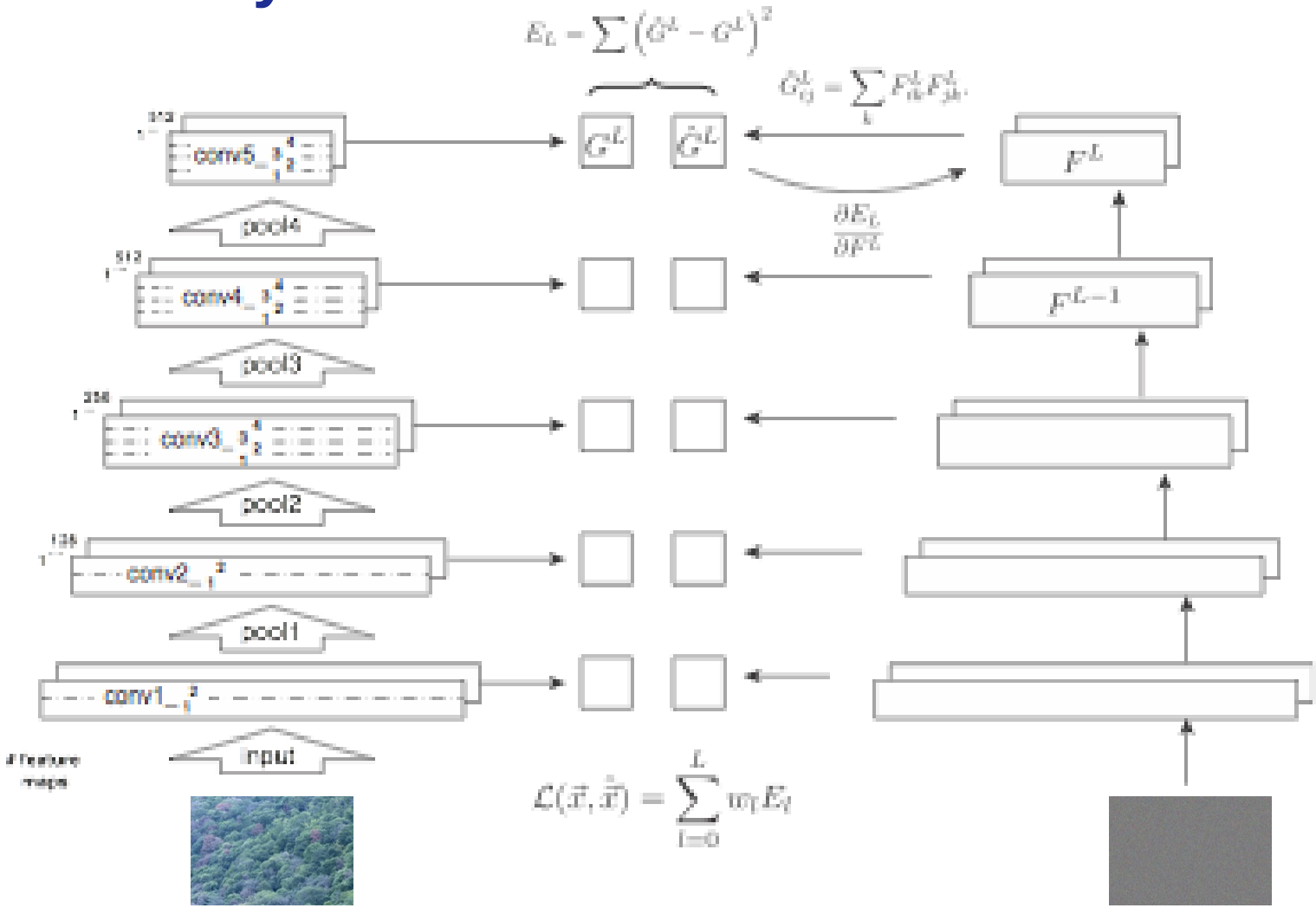
Texture Synthesis



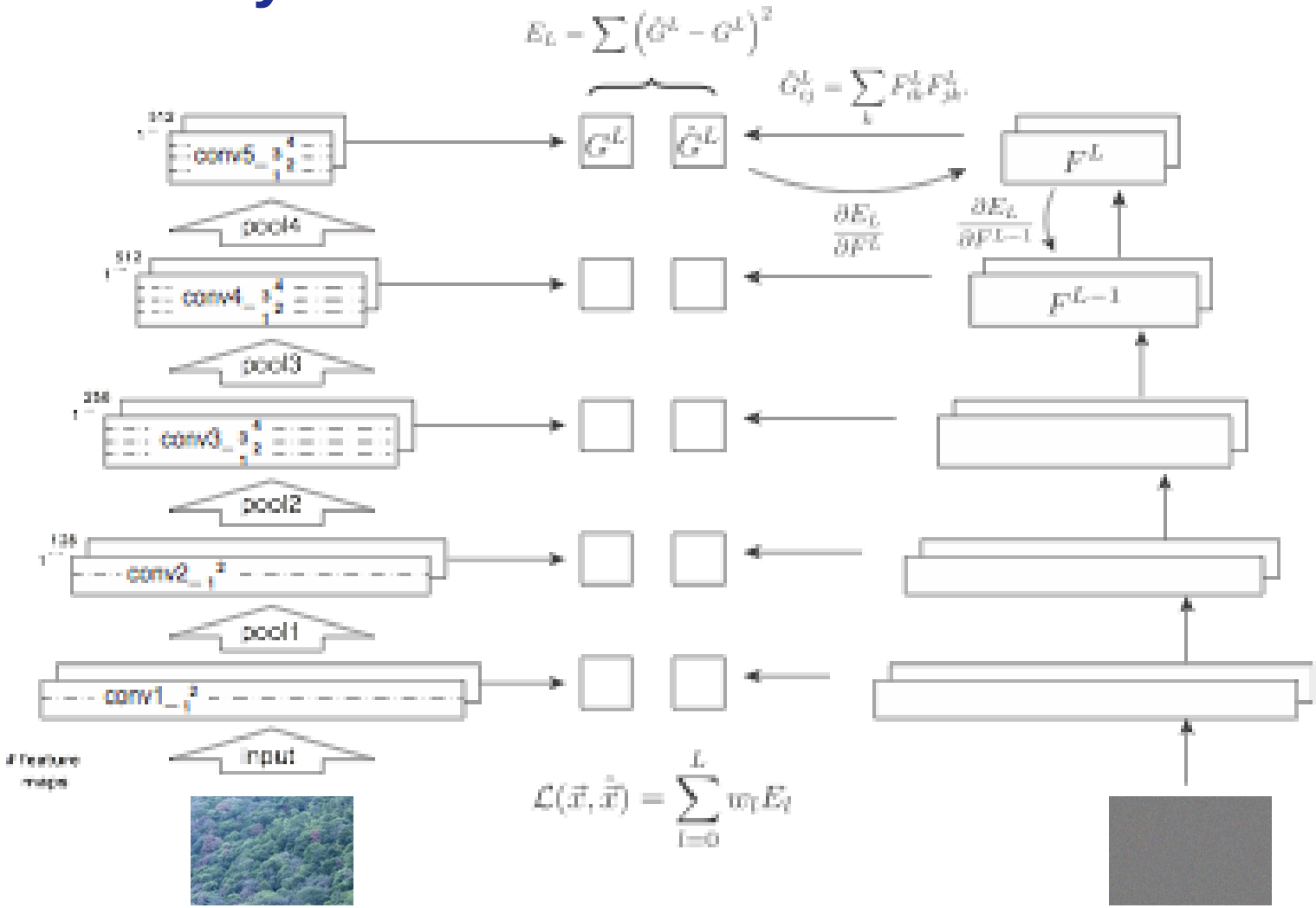
Texture Synthesis



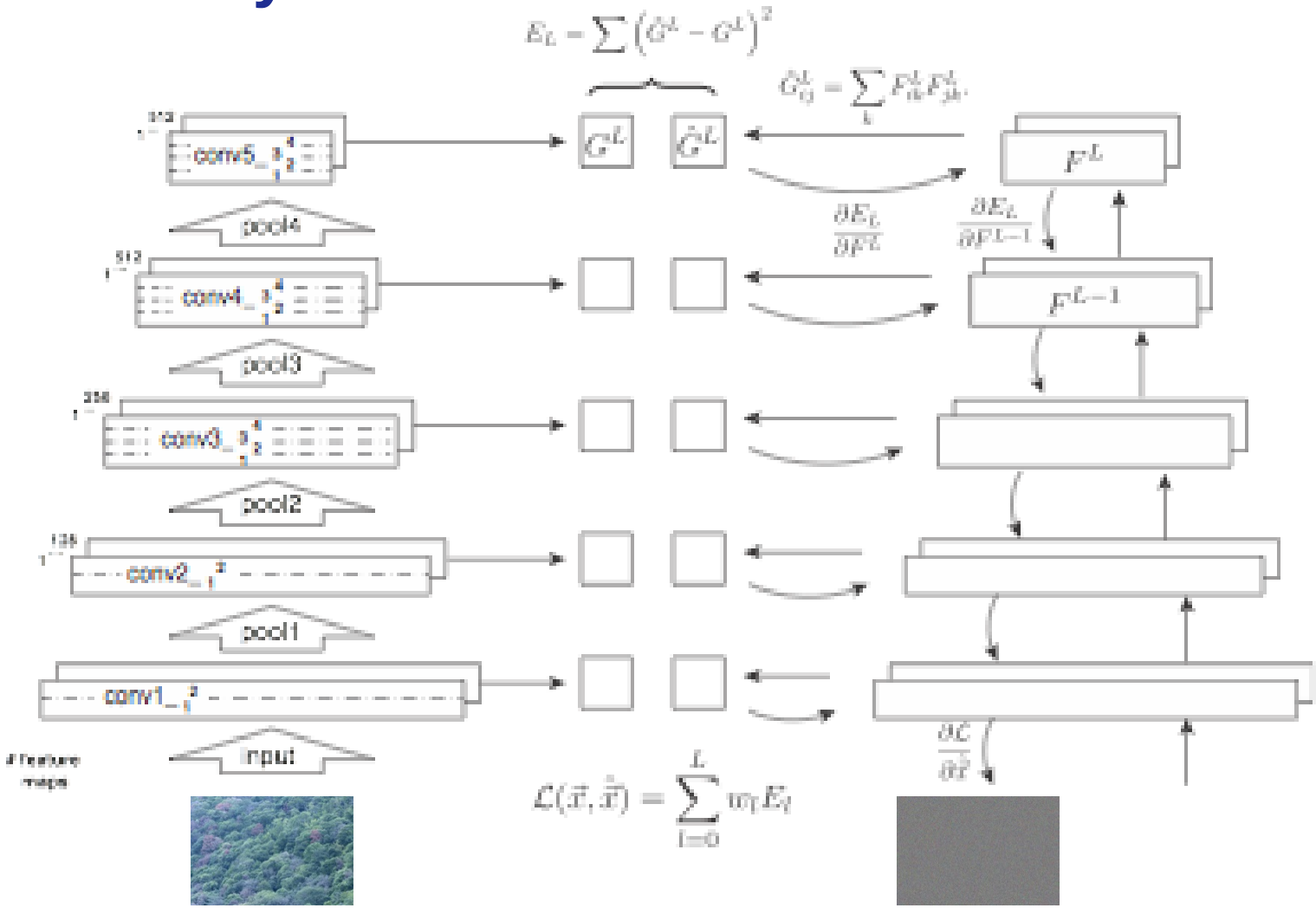
Texture Synthesis



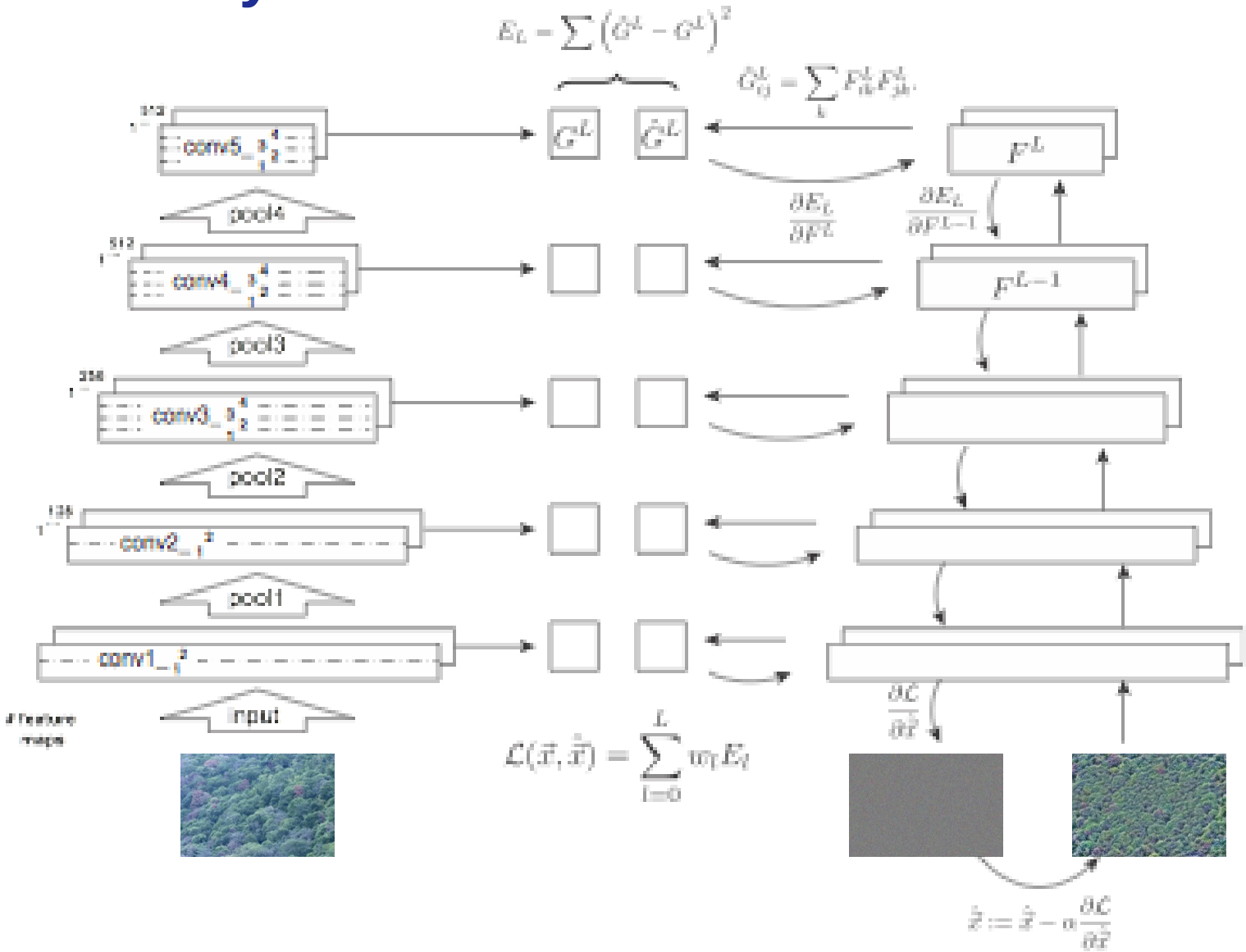
Texture Synthesis



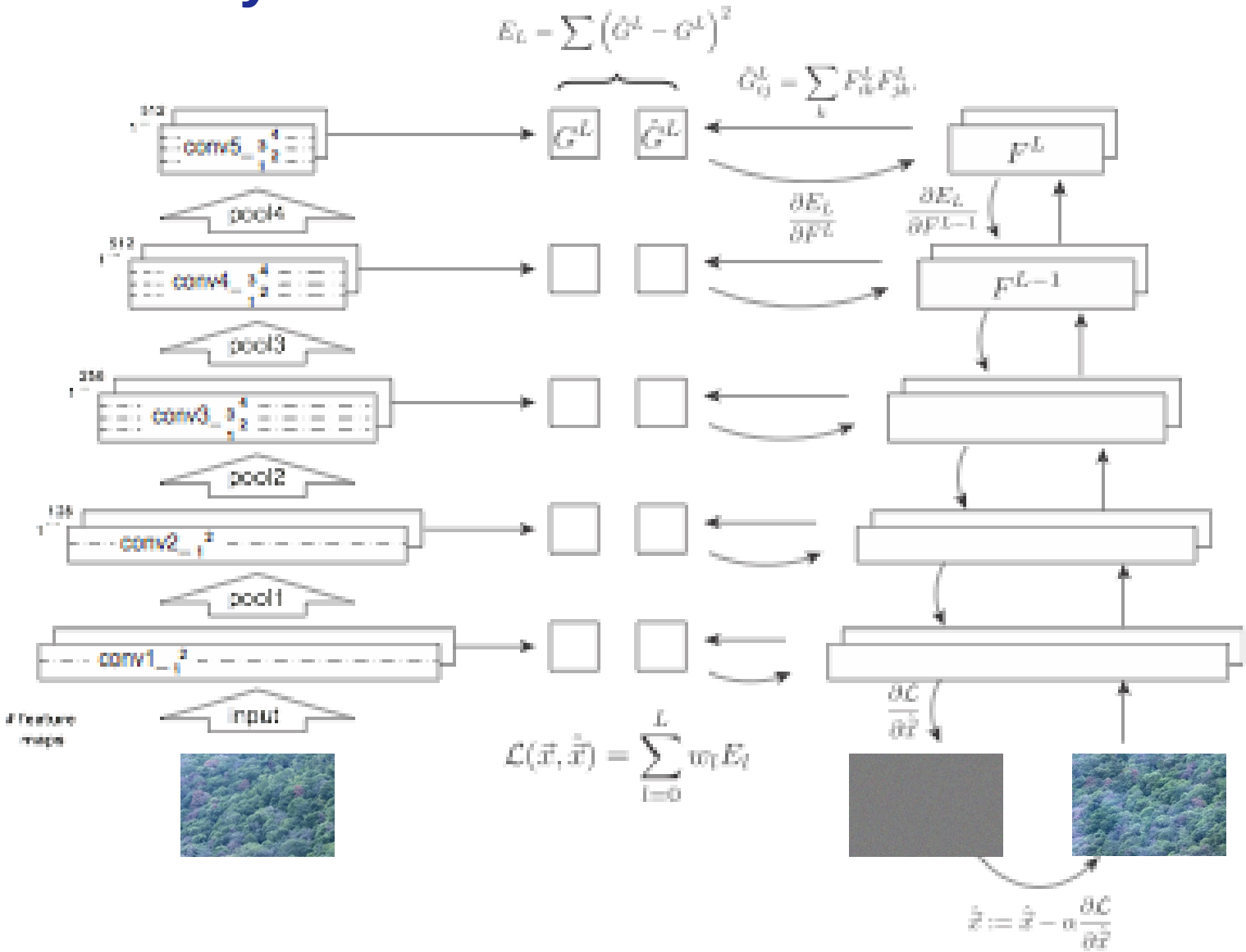
Texture Synthesis



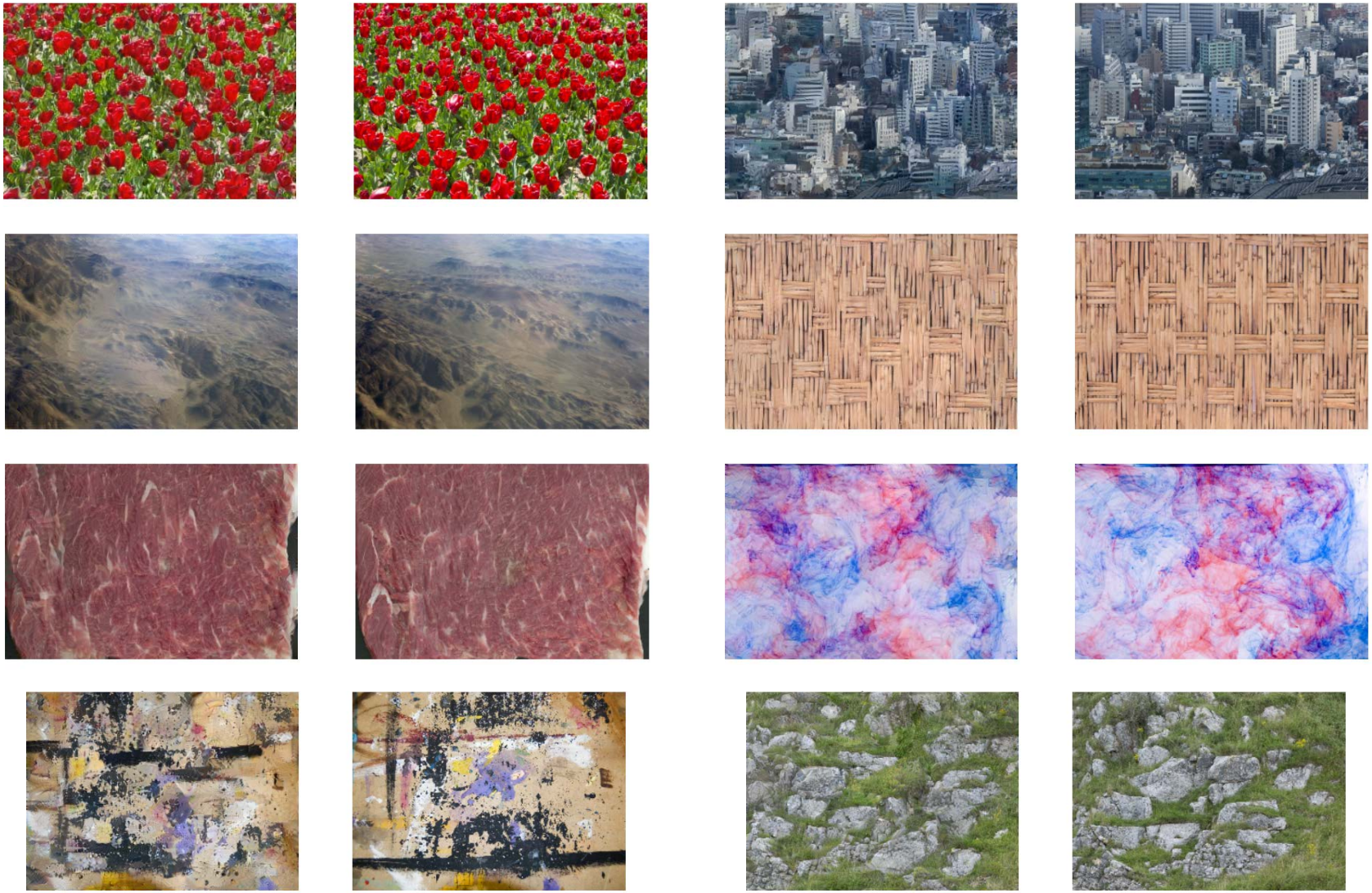
Texture Synthesis



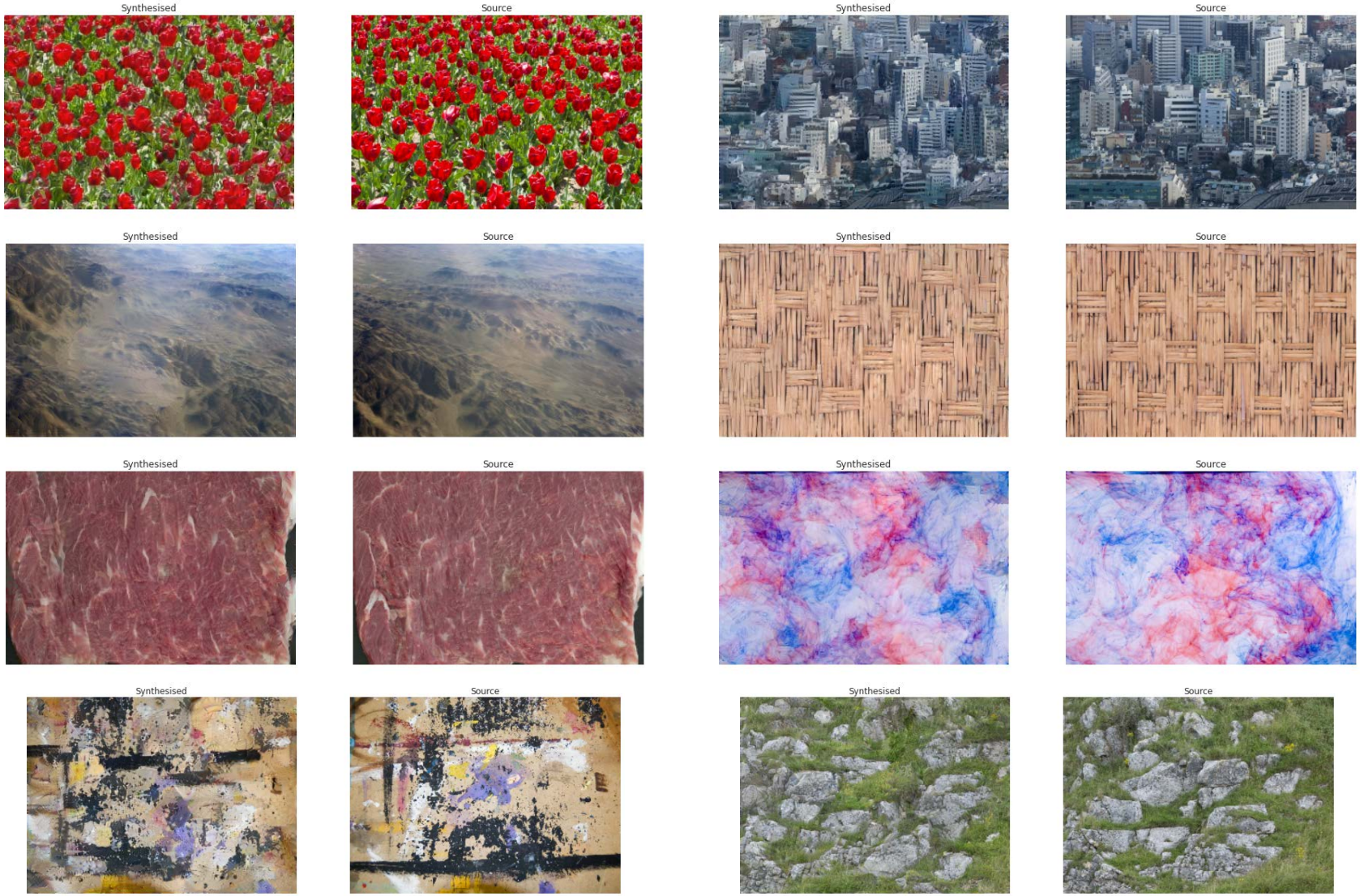
Texture Synthesis



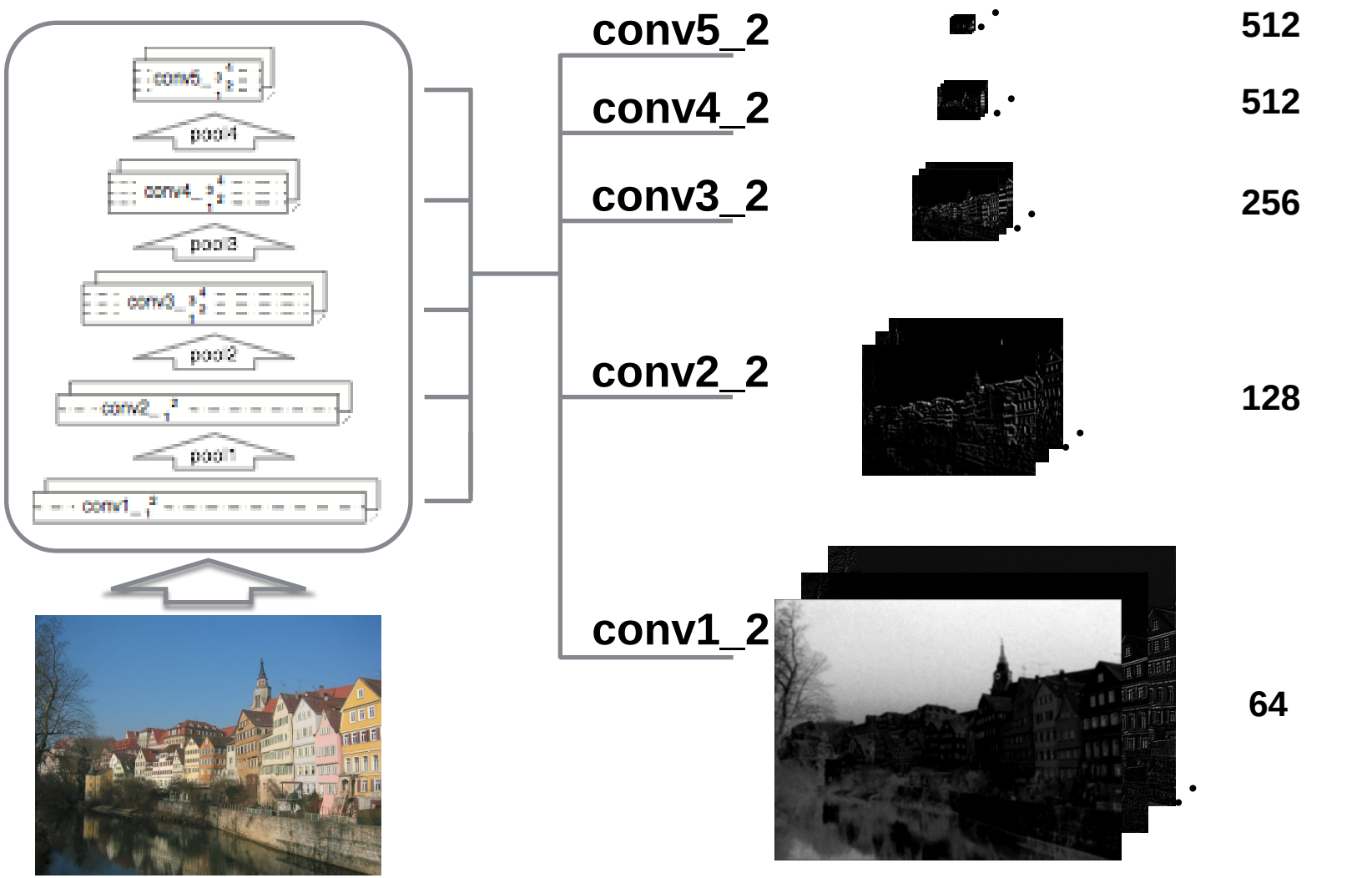
Test Julesz' Conjecture



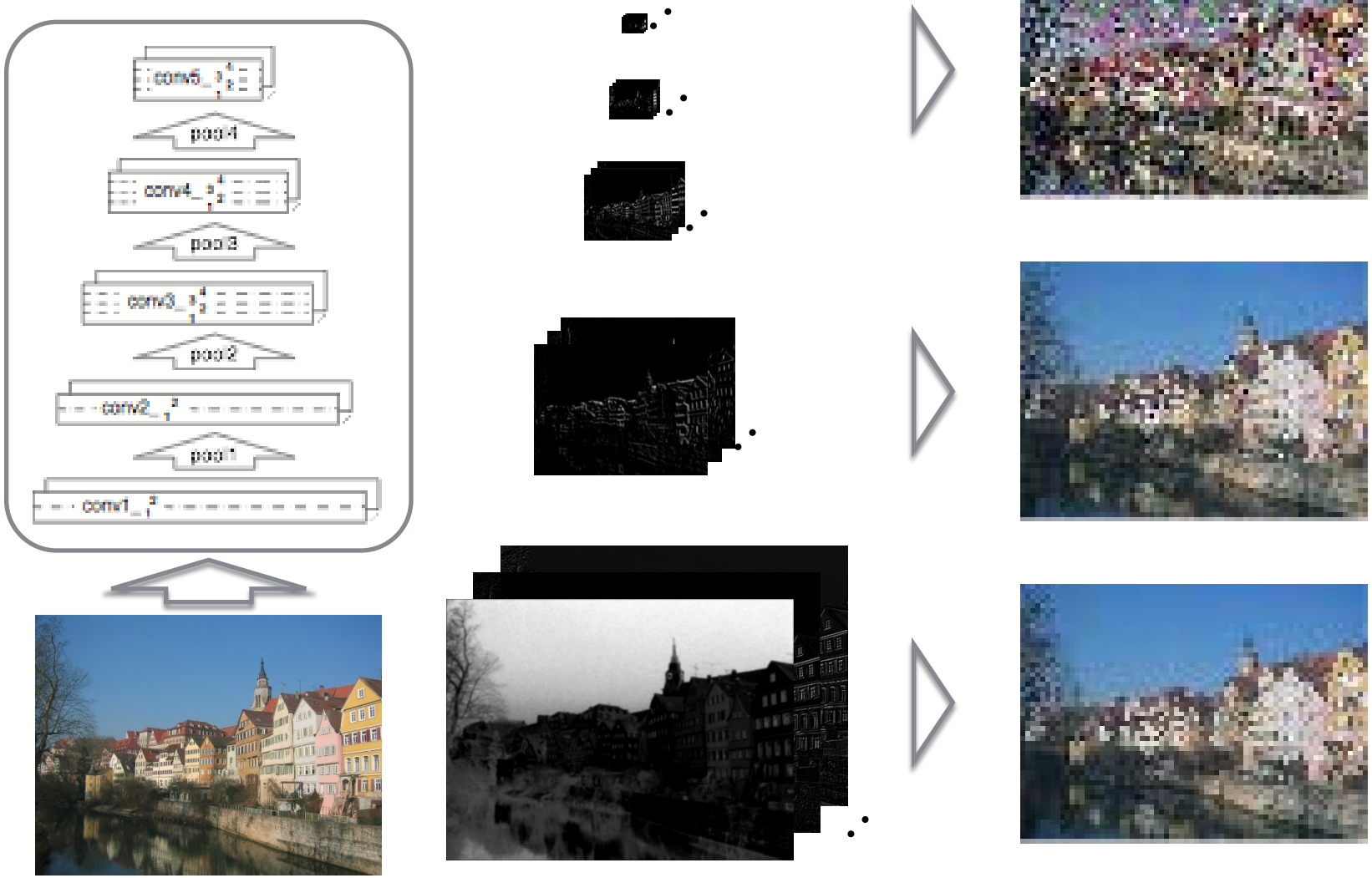
Test Julesz' Conjecture



CNN - Multiscale Nonlinear Filter Bank

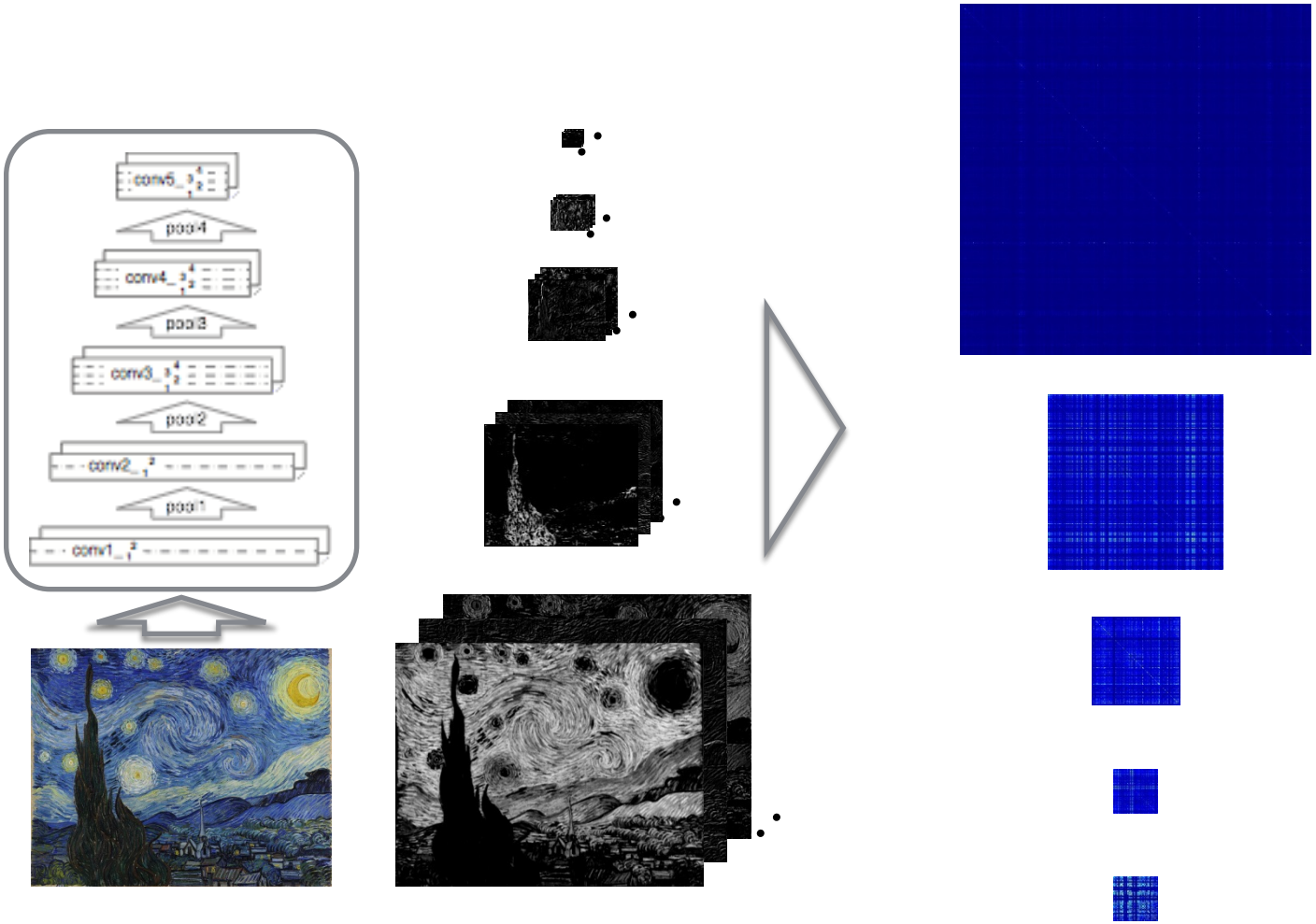


CNN - Image Reconstruction



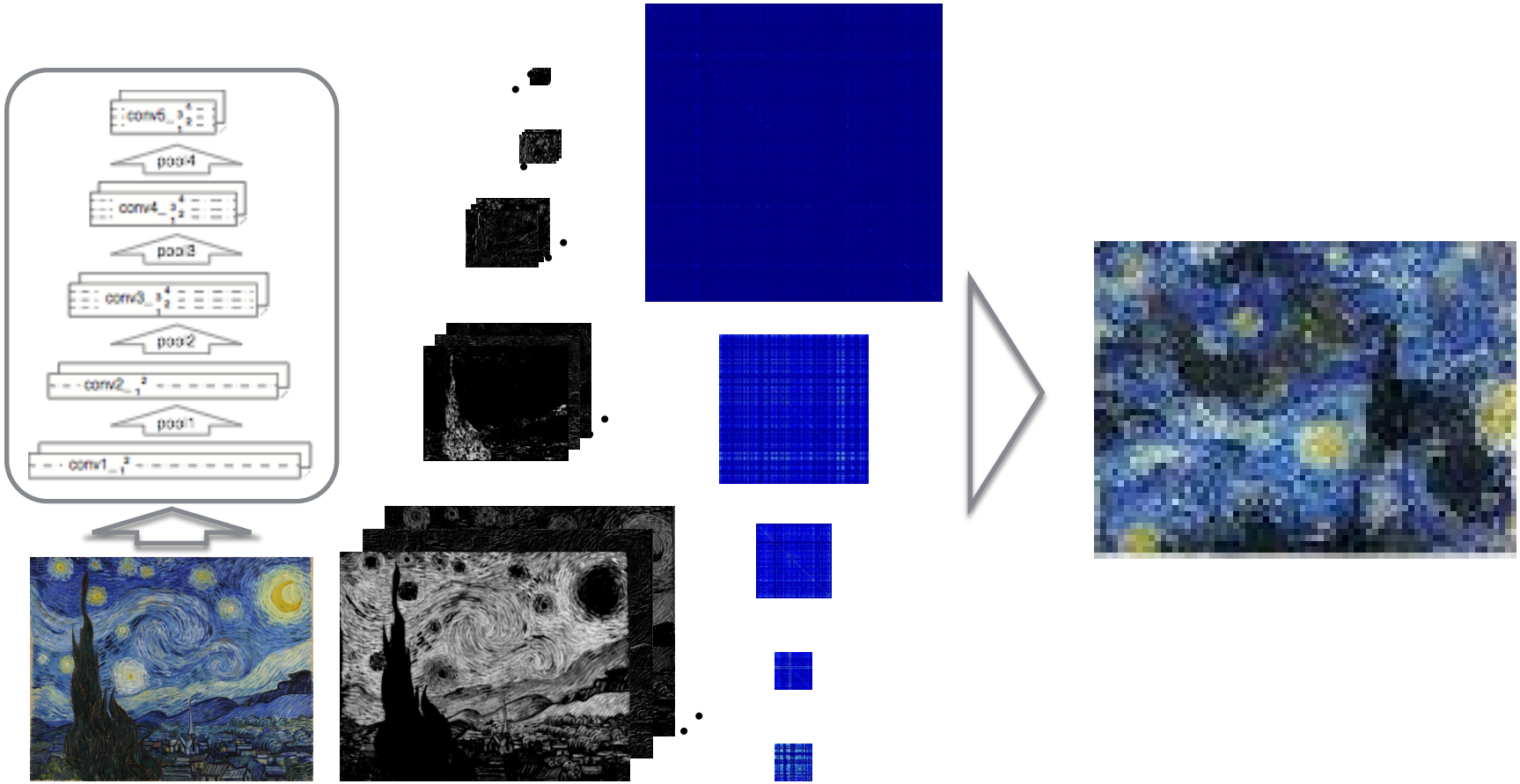
Mahendran & Vedaldi (2014)

CNN - Texture Features



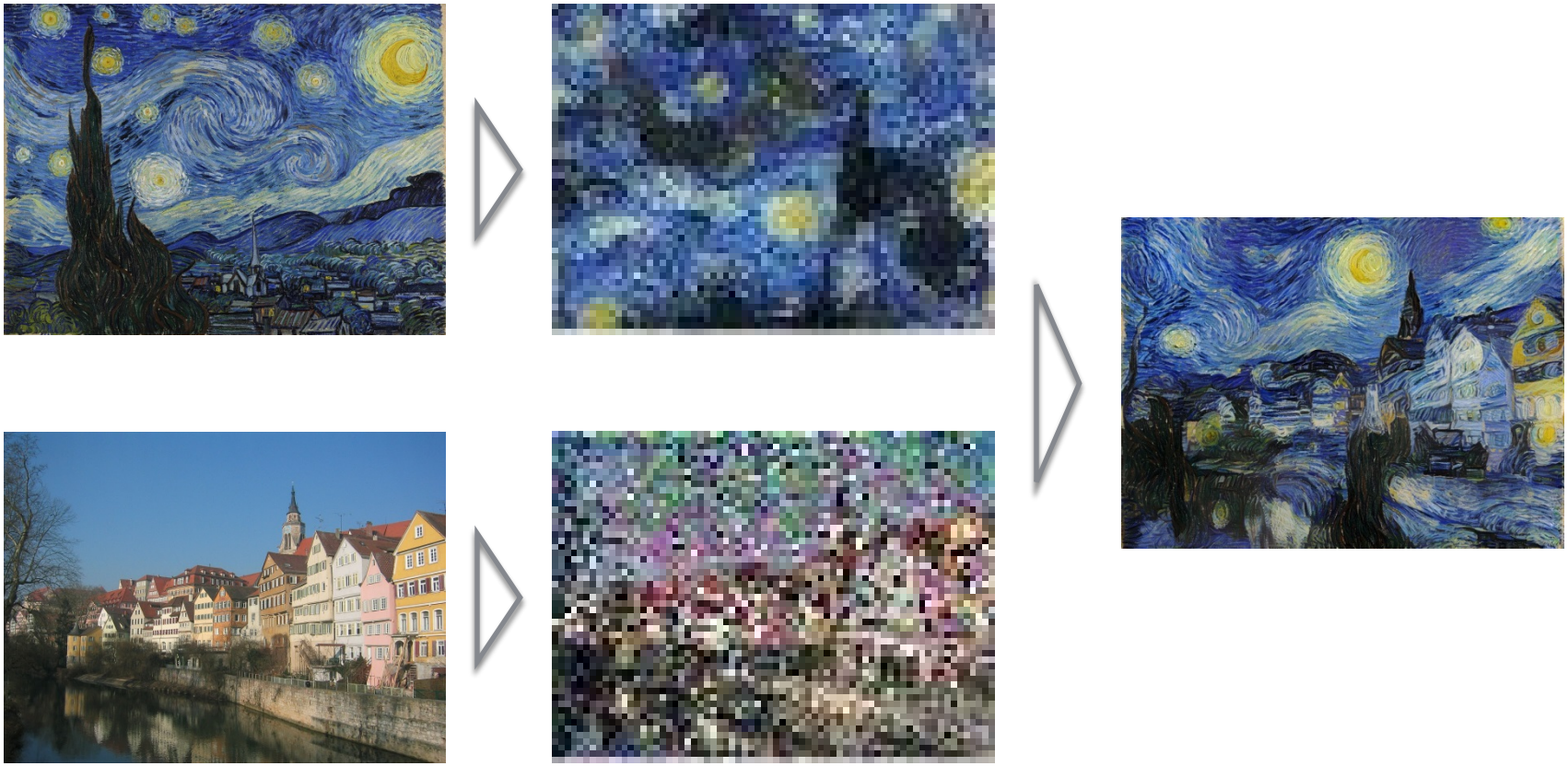
Gatys et al. (NIPS 2015)

CNN - Texture Synthesis

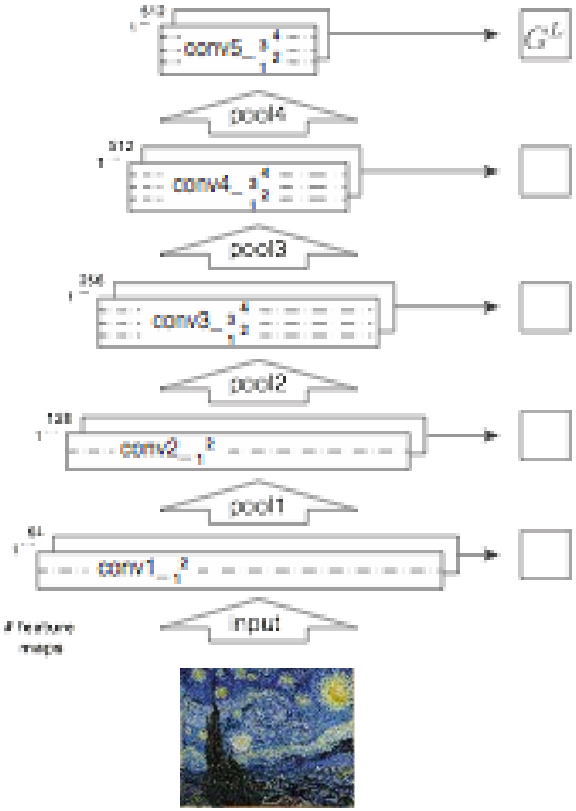


Gatys et al. (NIPS 2015)

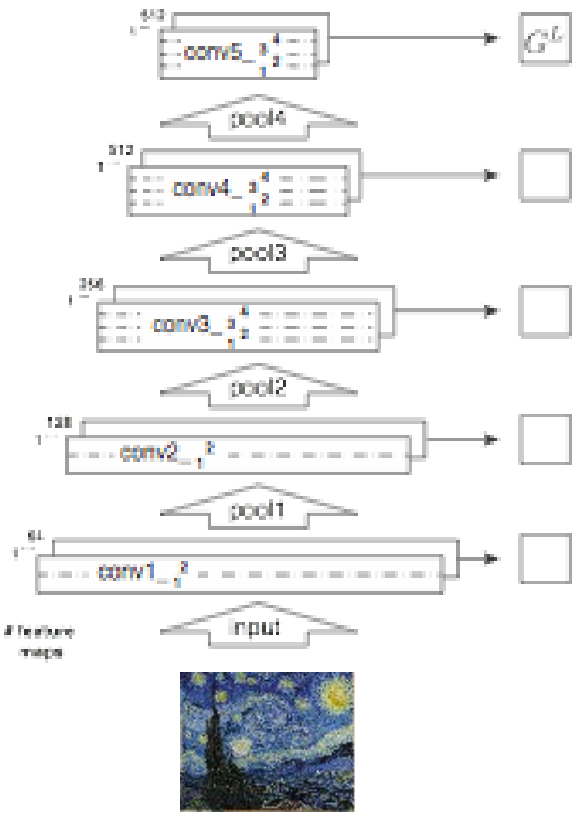
Artistic Style Transfer

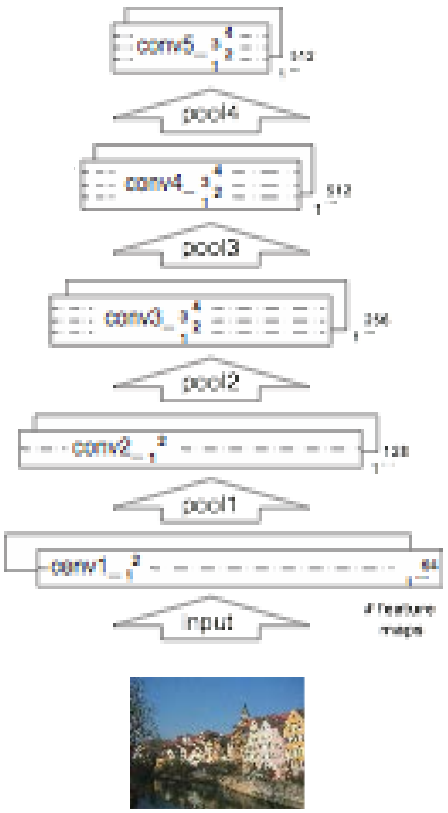


Artistic Style Transfer

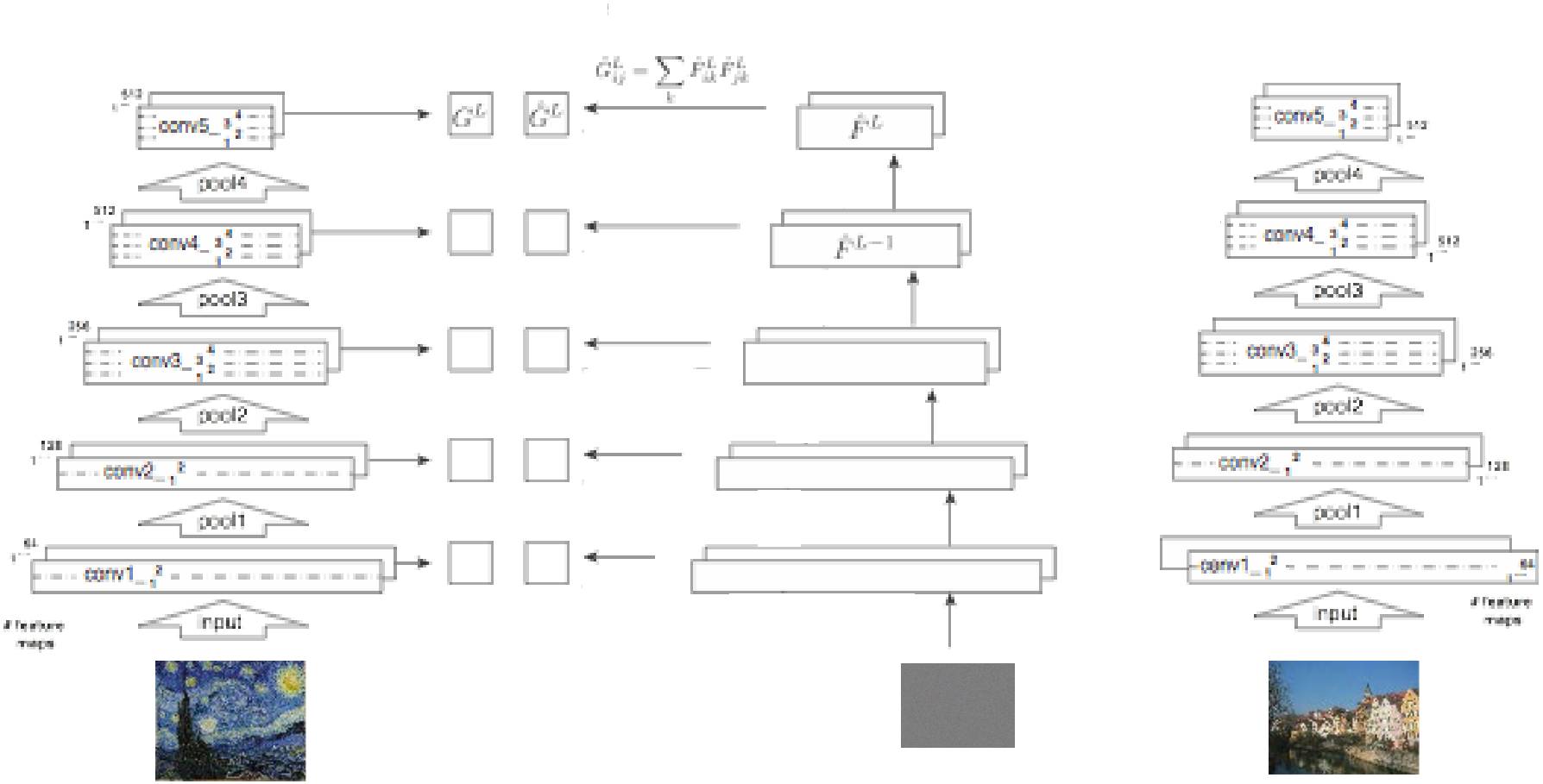


Artistic Style Transfer

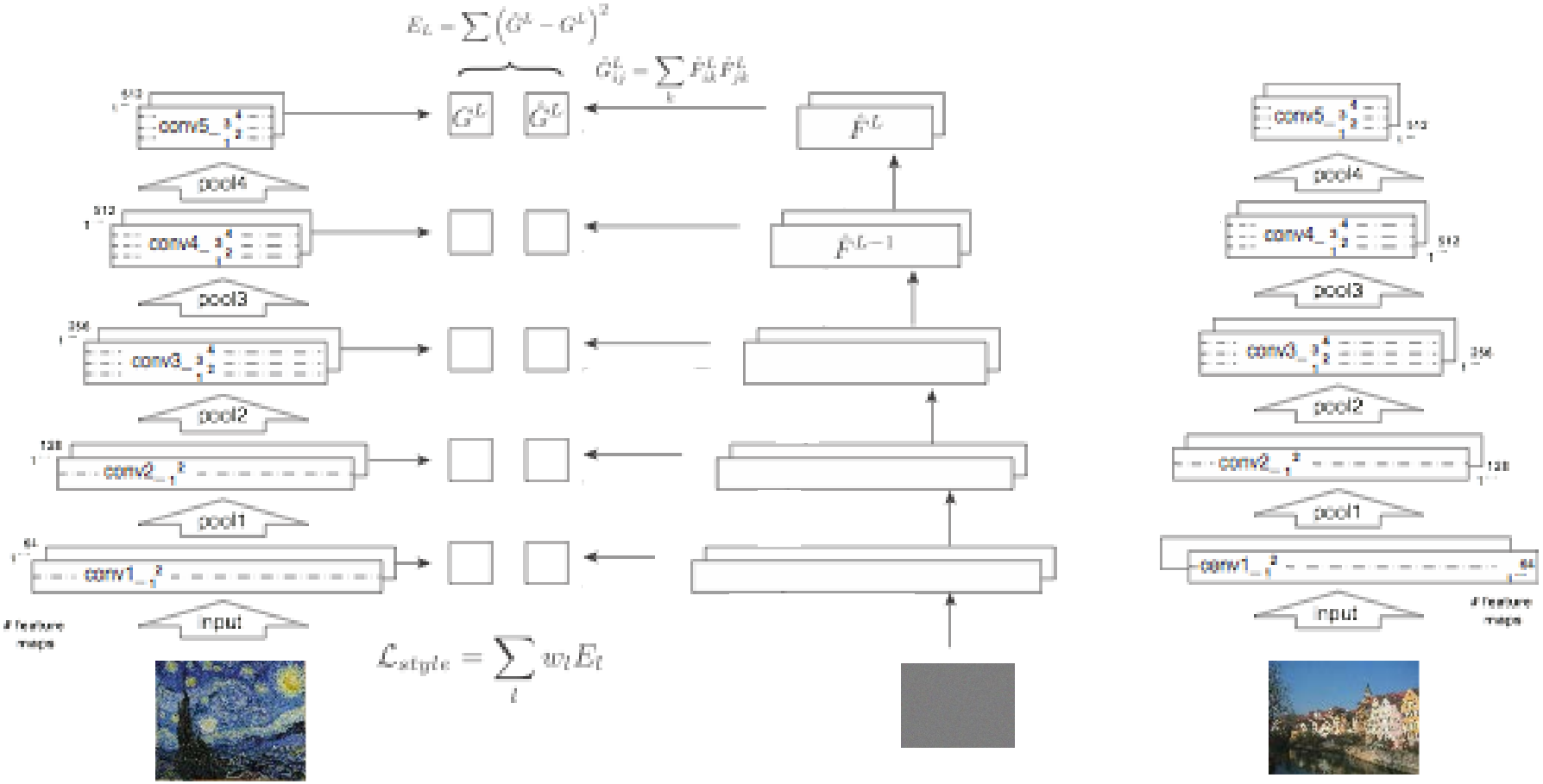




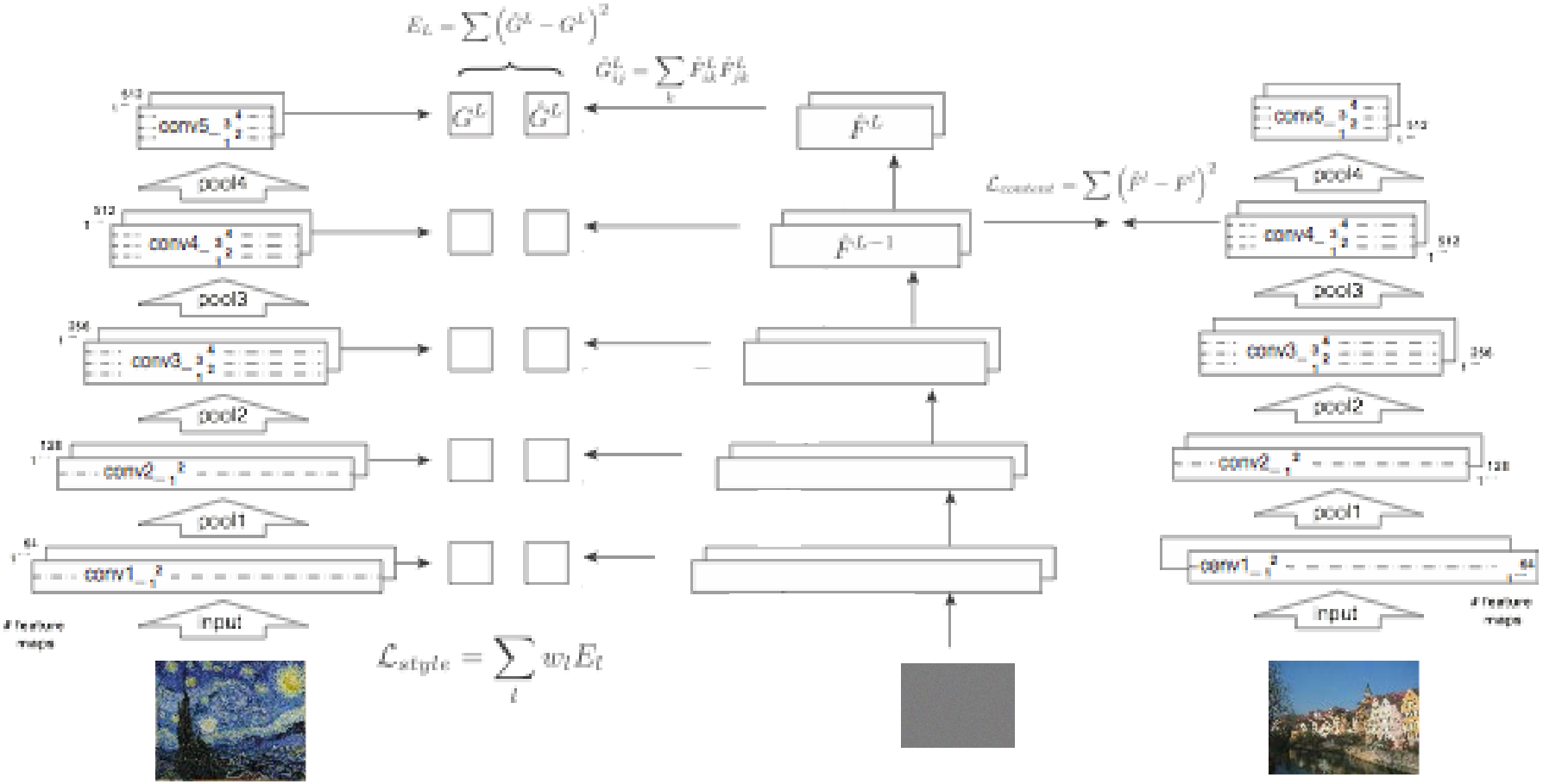
Artistic Style Transfer



Artistic Style Transfer

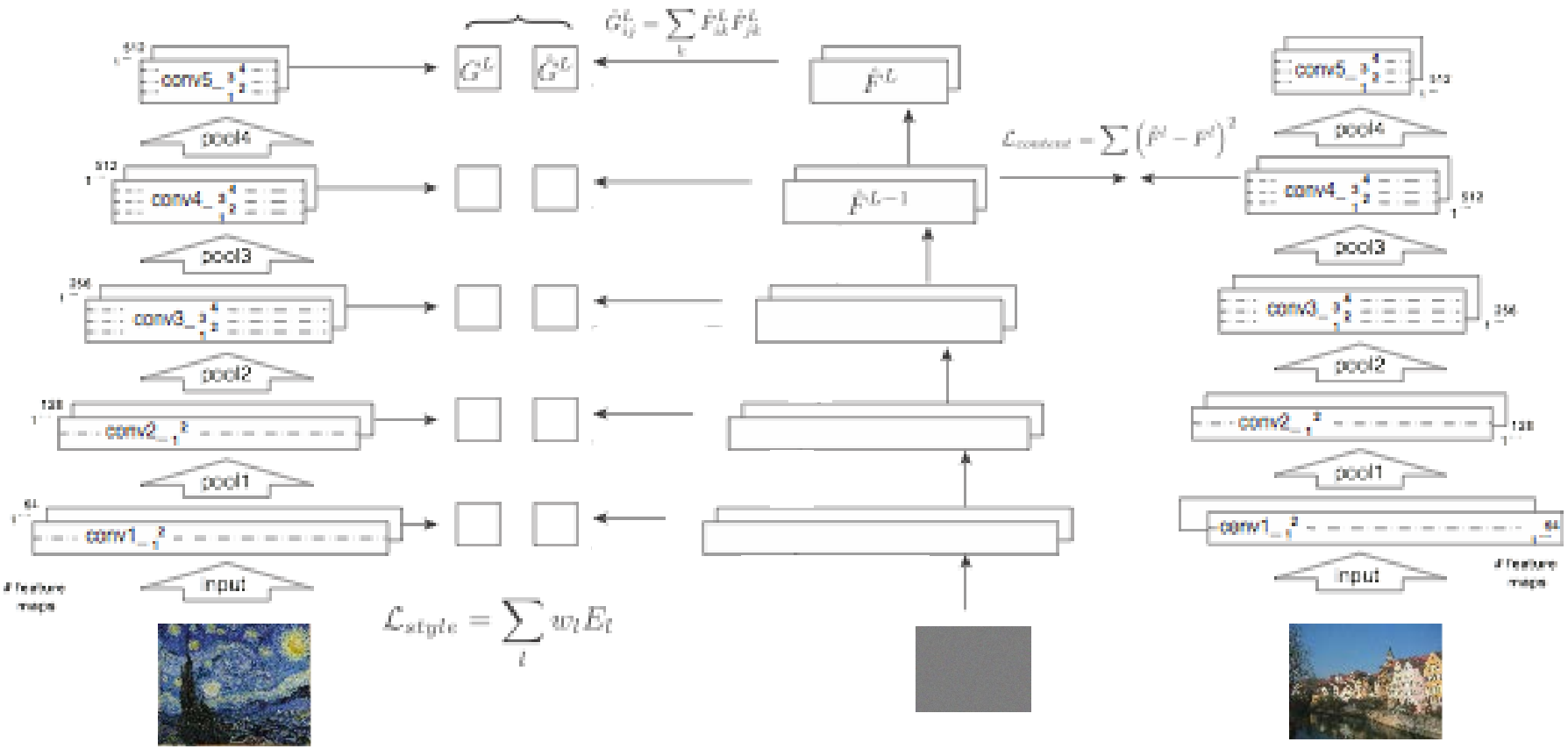


Artistic Style Transfer



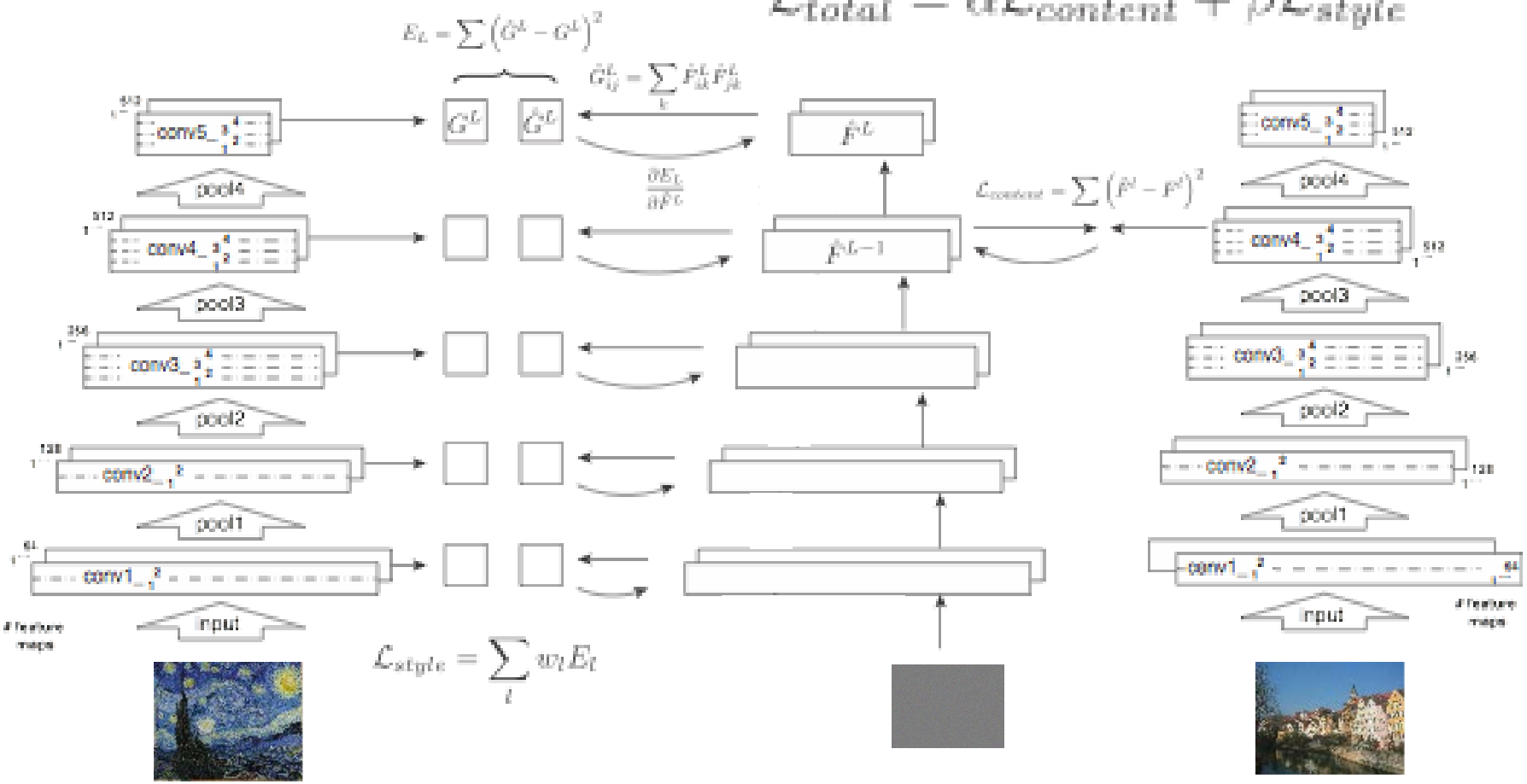
Artistic Style Transfer

$$\mathcal{L}_{total} = \alpha \mathcal{L}_{content} + \beta \mathcal{L}_{style}$$



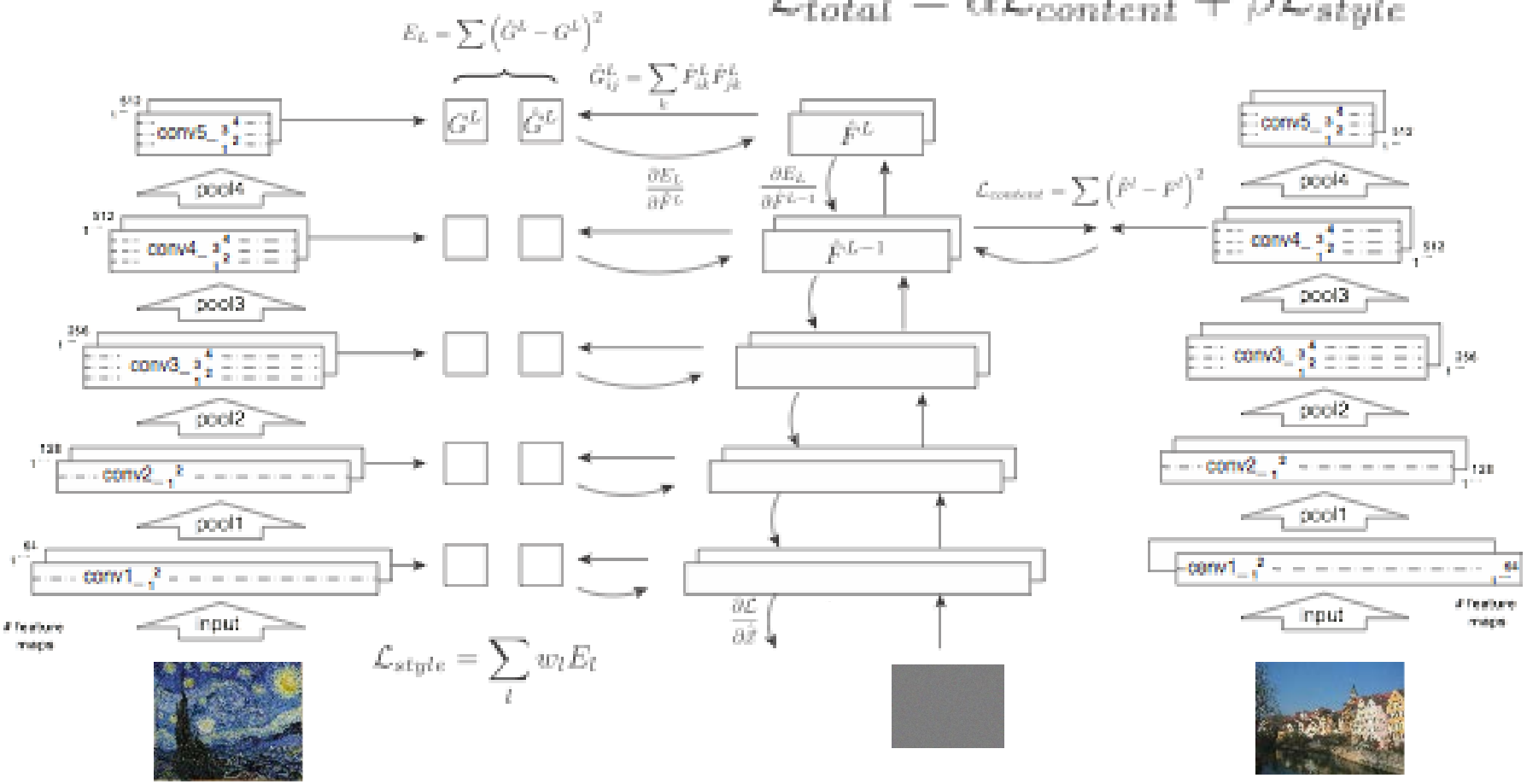
Artistic Style Transfer

$$\mathcal{L}_{total} = \alpha \mathcal{L}_{content} + \beta \mathcal{L}_{style}$$

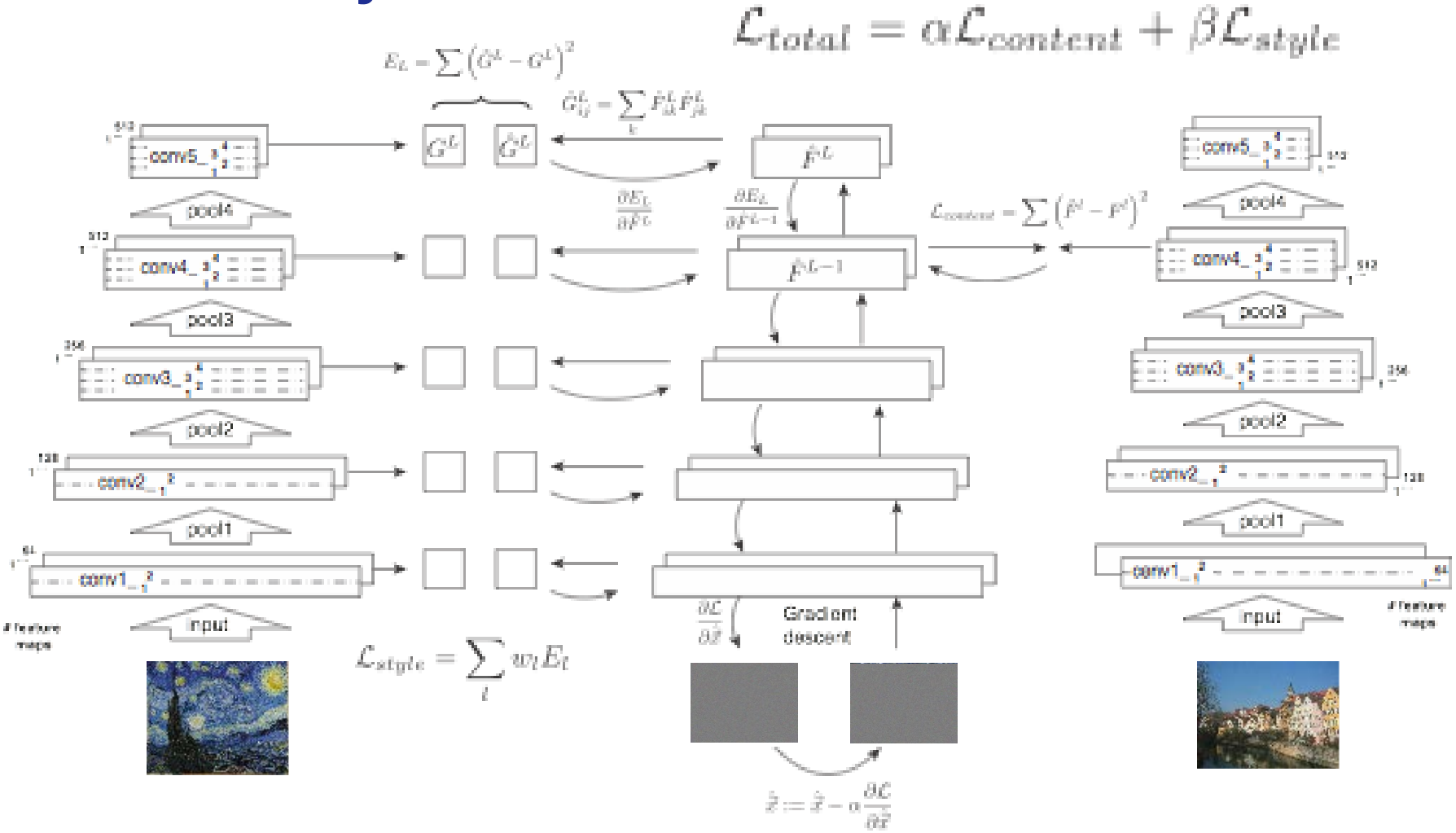


Artistic Style Transfer

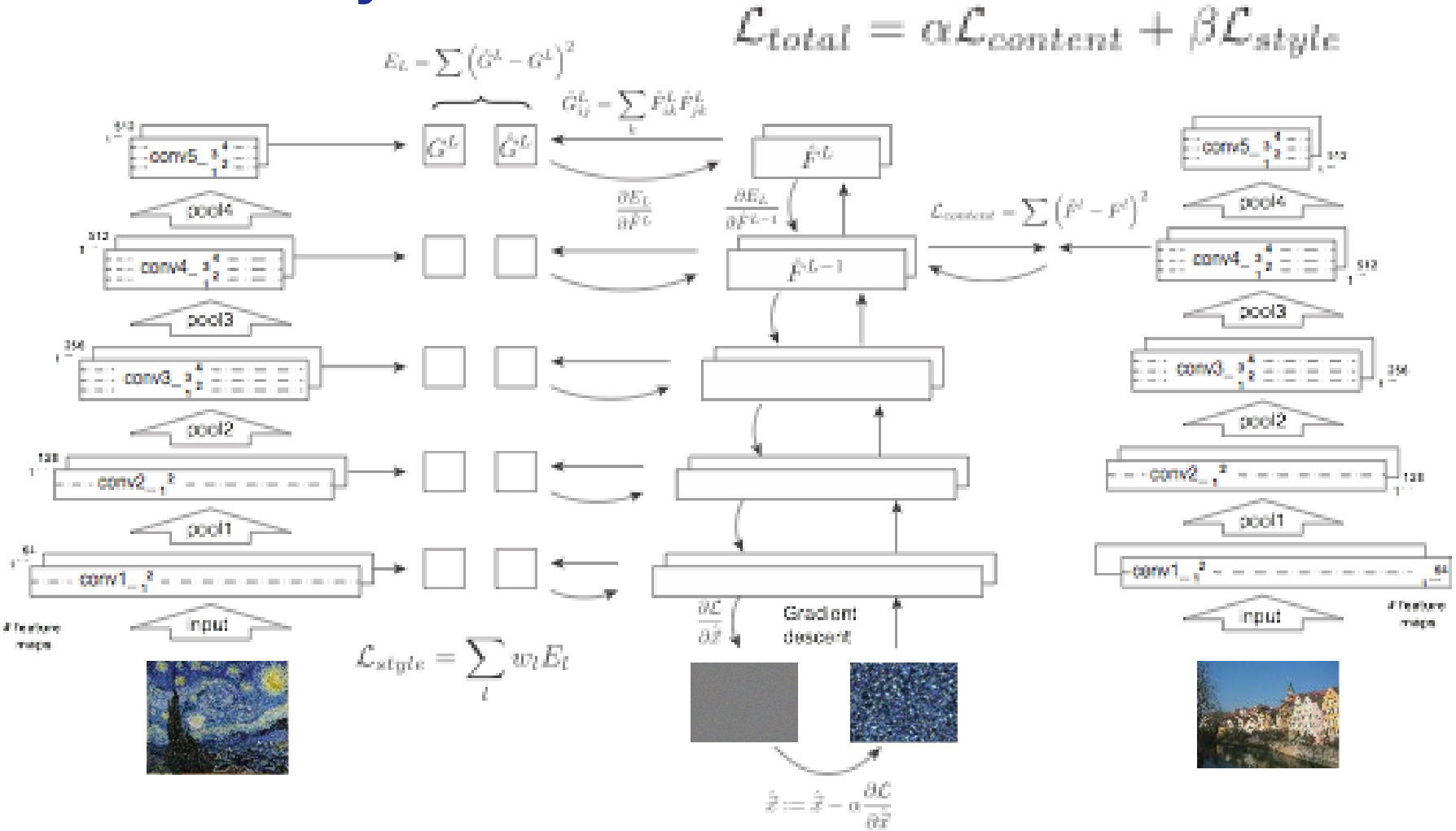
$$\mathcal{L}_{total} = \alpha \mathcal{L}_{content} + \beta \mathcal{L}_{style}$$



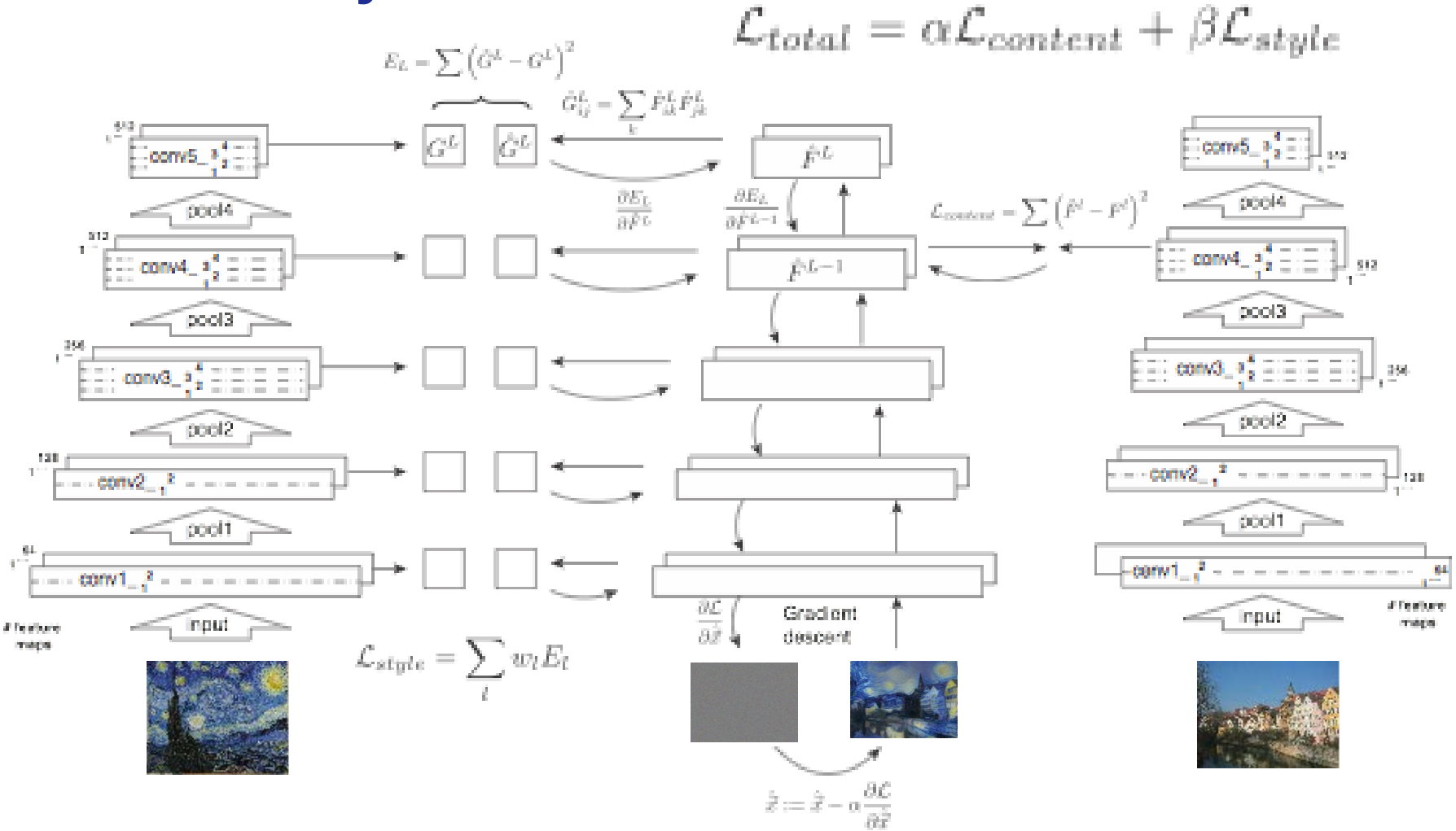
Artistic Style Transfer



Artistic Style Transfer



Artistic Style Transfer



Relative Weighting of Content and Style

1e-4

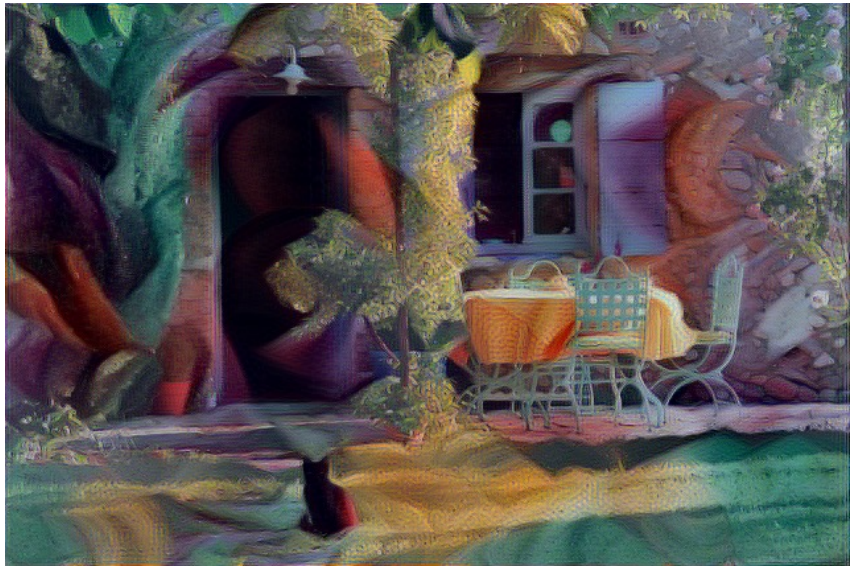
1e-3

1e-2

1e-1

Different Reconstruction Layers

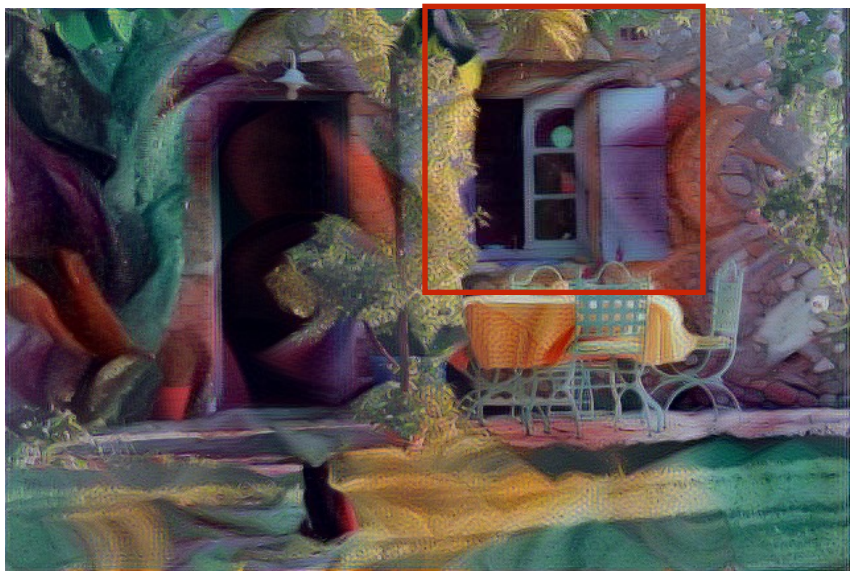
Conv2_2



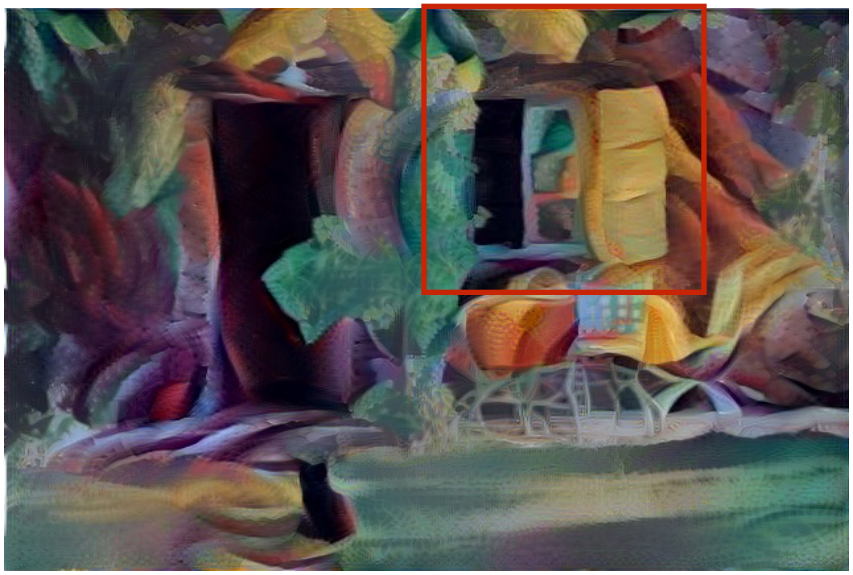
Conv4_2

Different Reconstruction Layers

Conv2_2



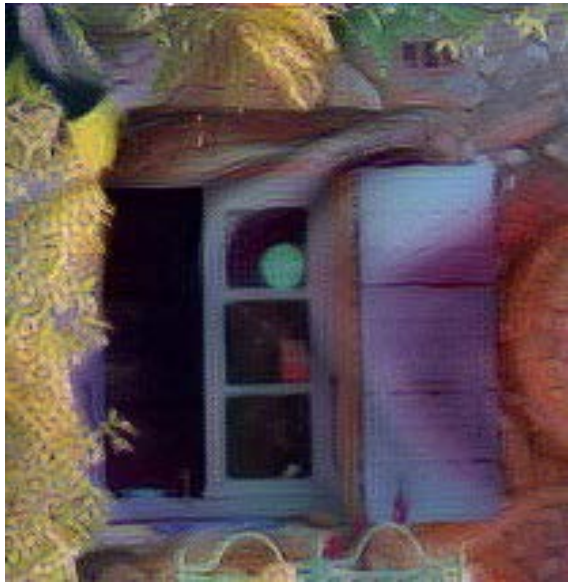
Conv4_2



Different Reconstruction Layers

Original

Conv2_2



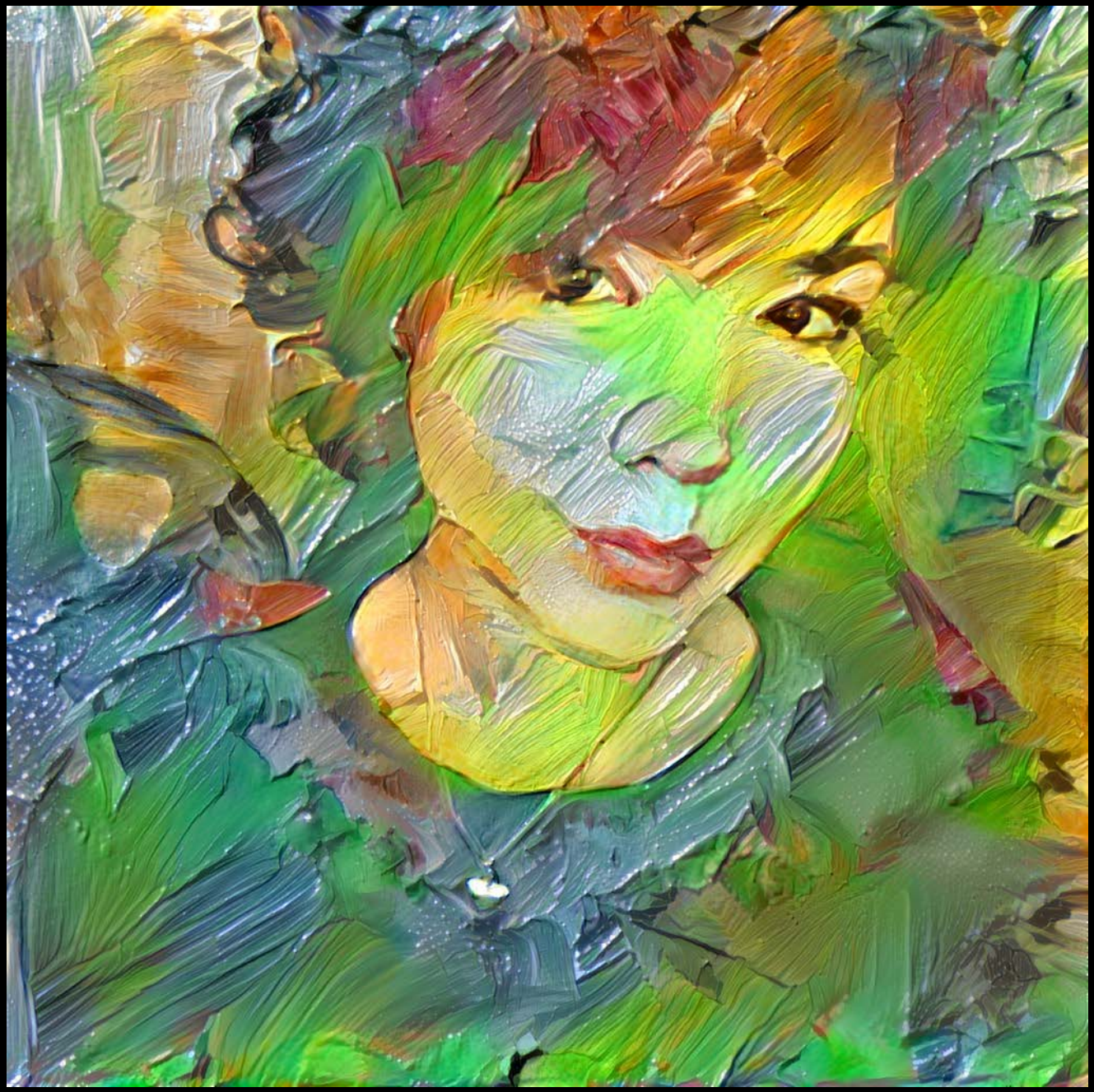
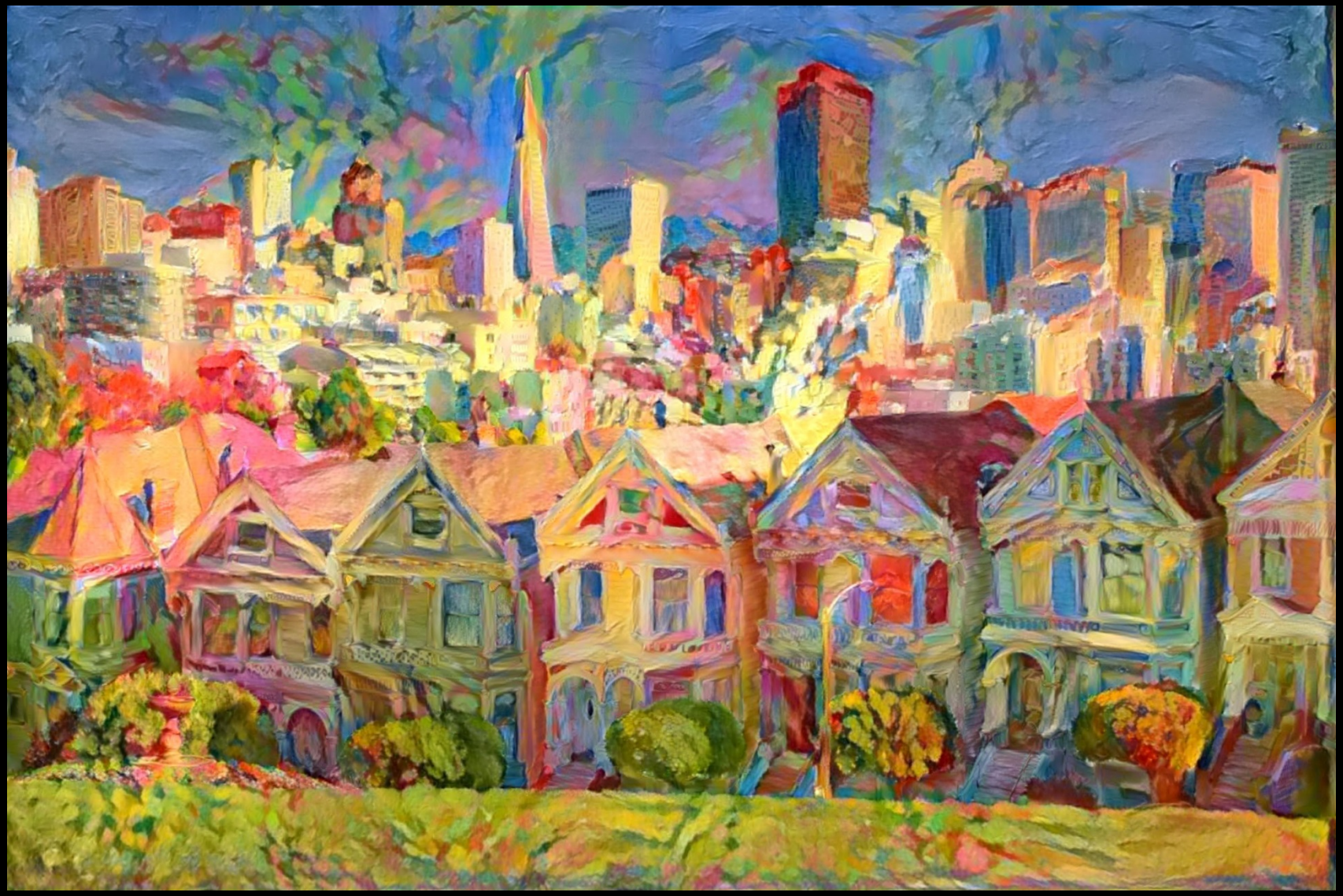
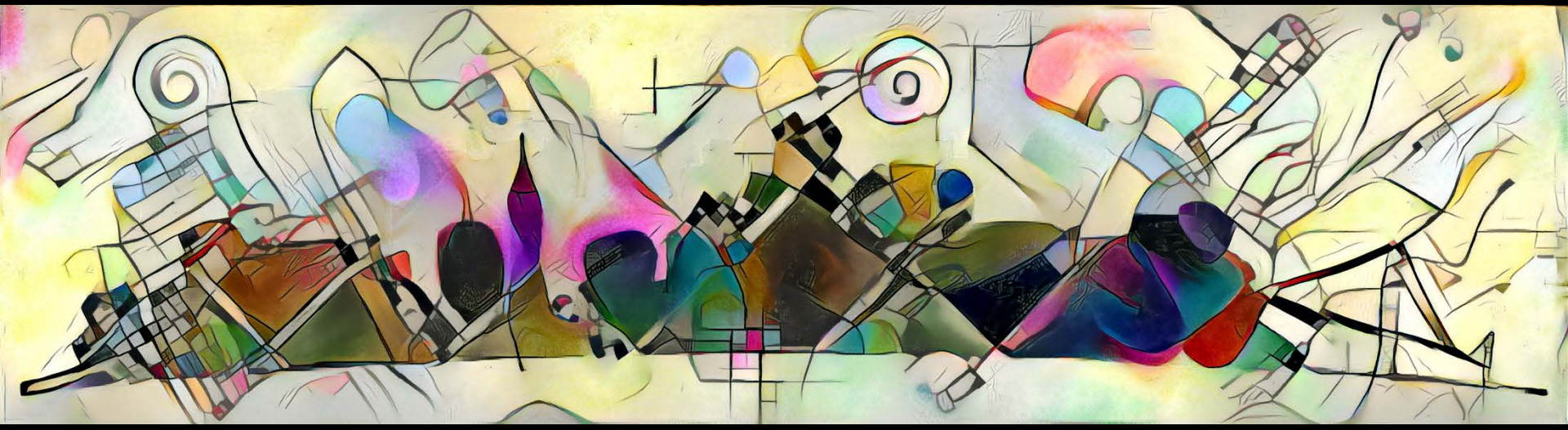
Conv4_2

General Style Transfer

General Style Transfer

The DeepArt Project

DEEPART.io



© Wayne Tilcock