Introduction to Computer Graphics

– Image Processing (2) –

June 18, 2020

Kenshi Takayama

Texture Synthesis

Scenario 1: Removal of objects in images

Original

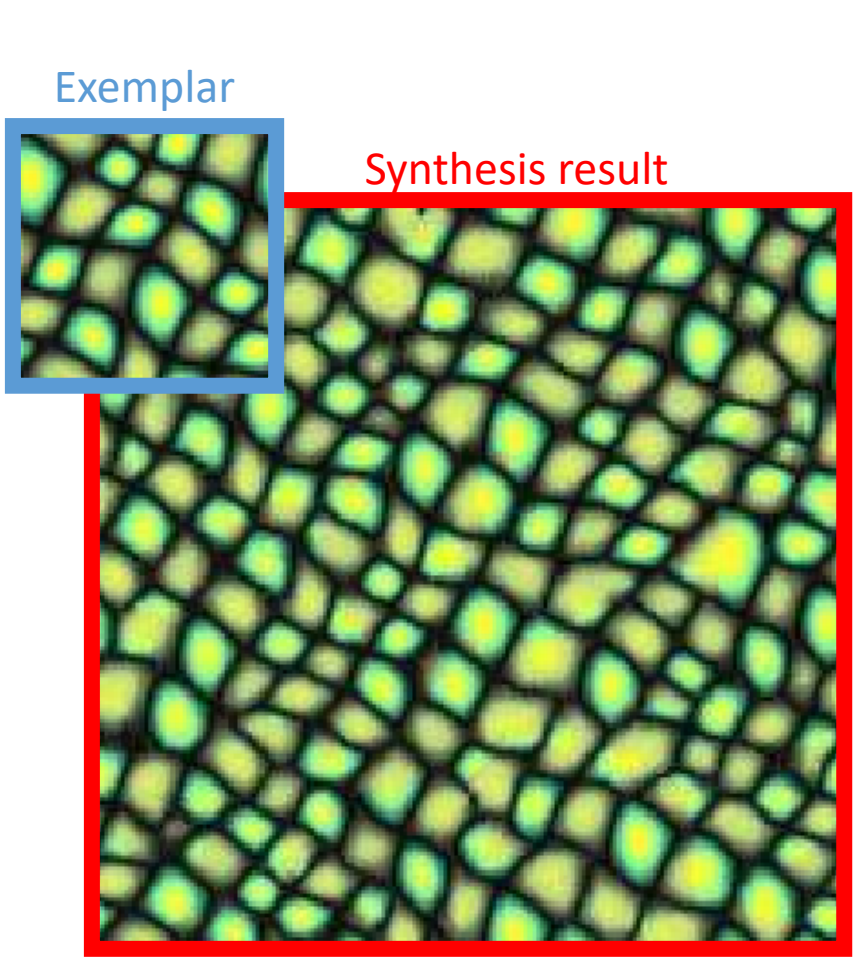
Mask

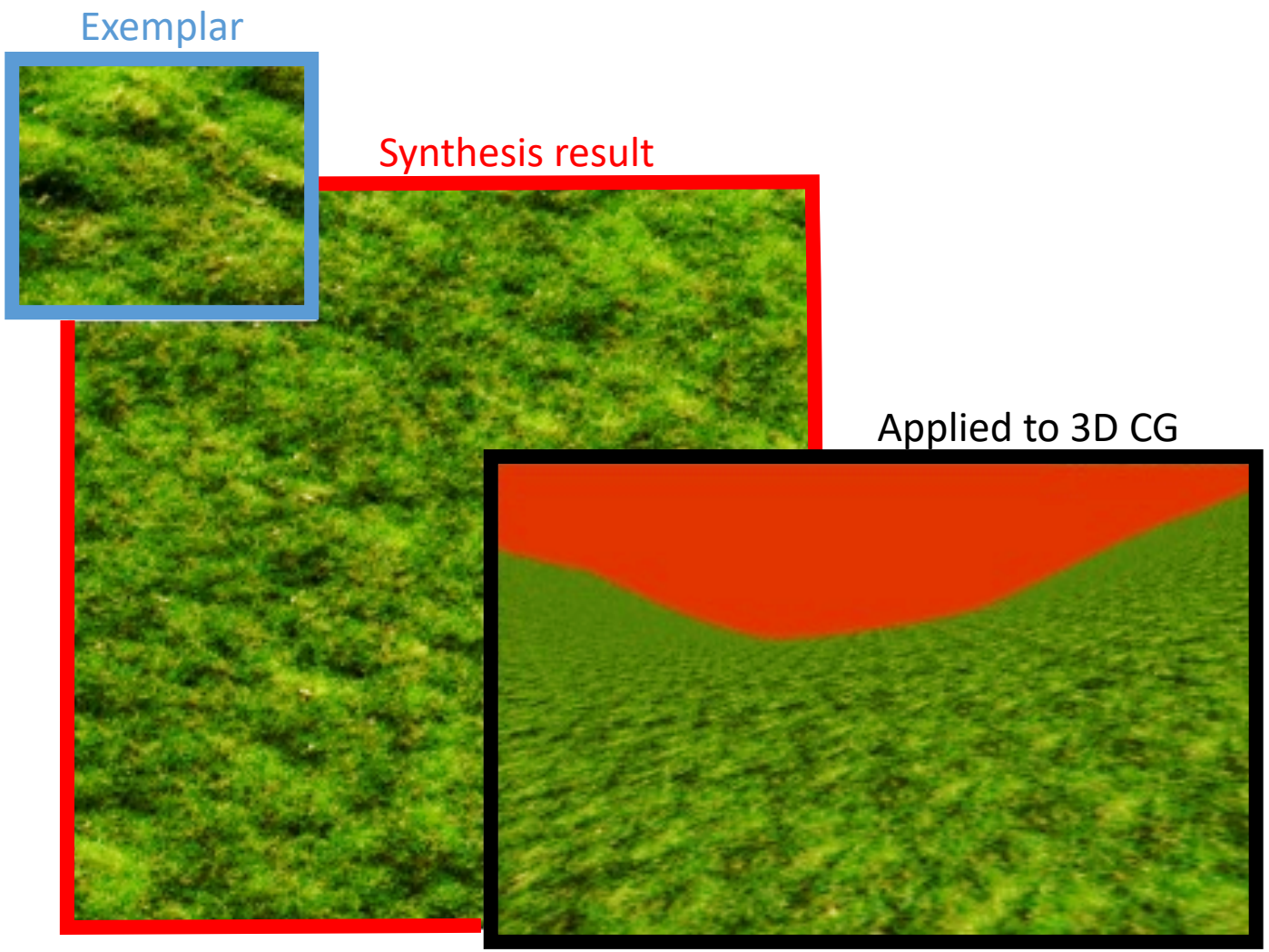
Synthesis result

- Bit different problem setting than “image cloning”

PatchMatch: a randomized correspondence algorithm for structural image editing [Barnes SIGGRAPH09]

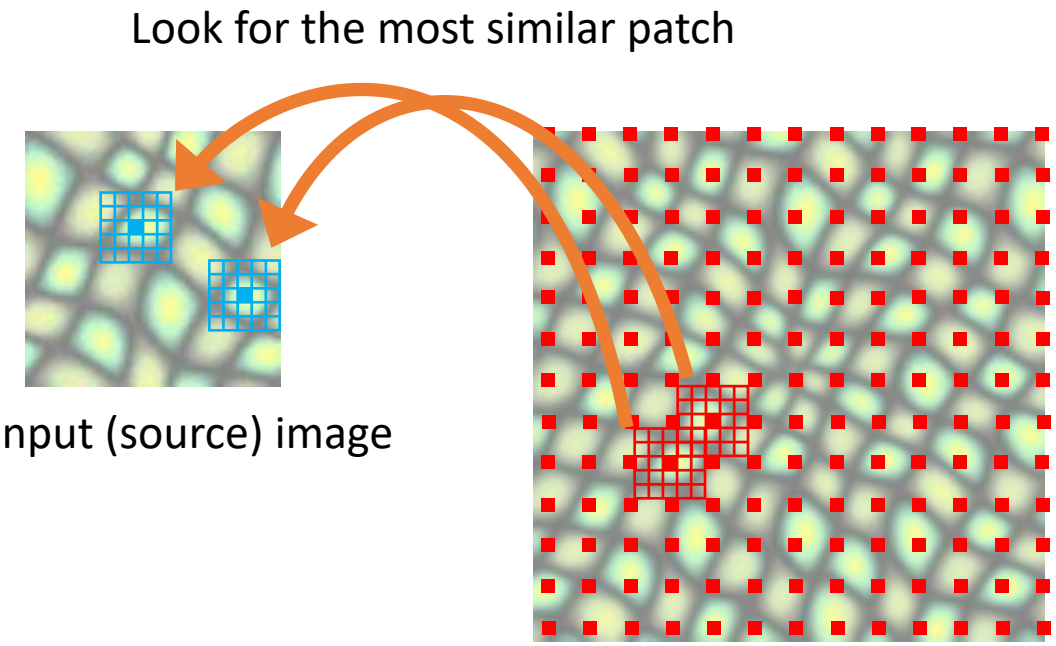
Scenario 2: Synthesis of large texture image





Fast texture synthesis using tree-structured vector quantization [Wei SIGGRAPH00]
http://robburke.net/mle/wang/

Similarity between input & output images [Kwatra05]



Sum of per-pixel differences squared

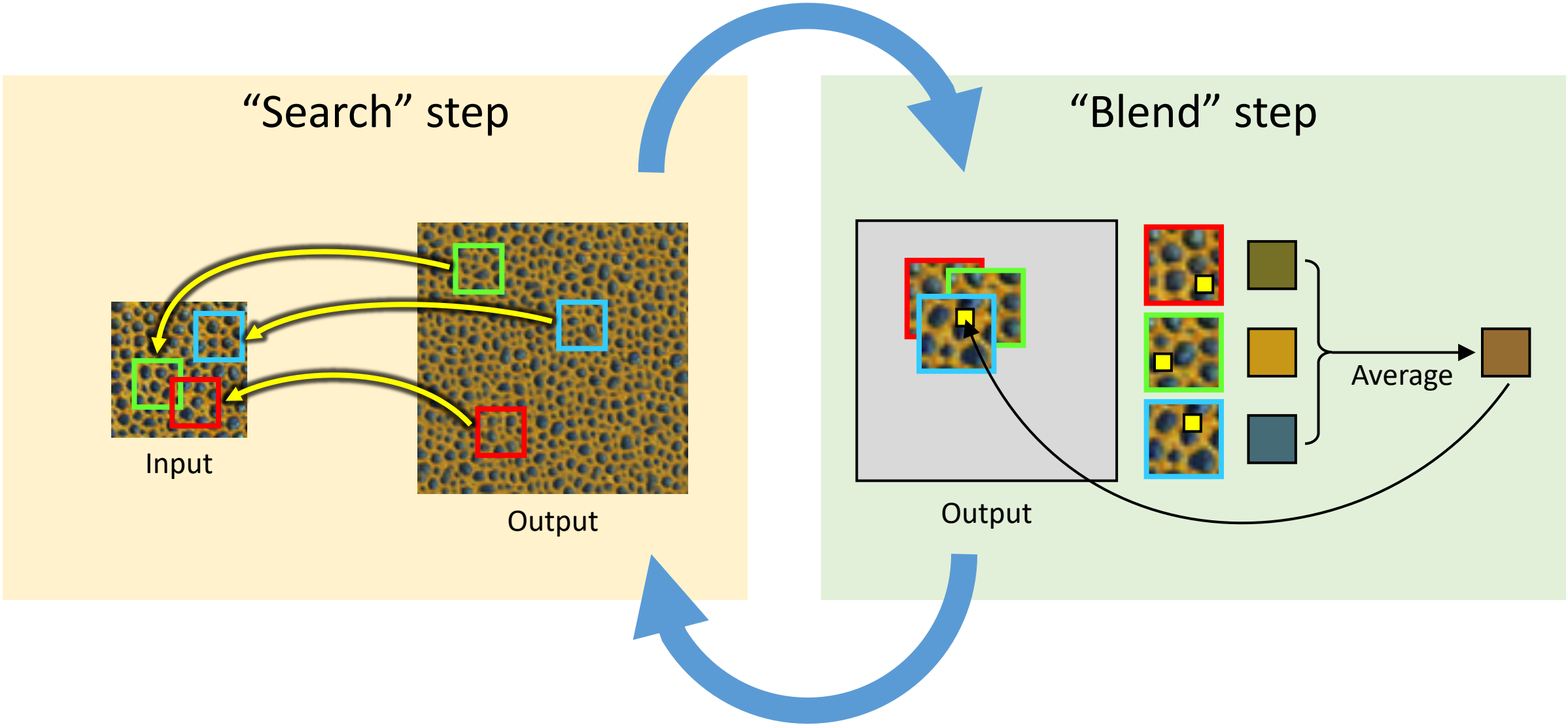
$$D(S,T) = \sum_{t \subset T} \min_{s \subset S} \|s - t\|^2$$

patches

- Want to find T which minimizes D
- Direct solution seems infeasible
  ➔ iterative computation

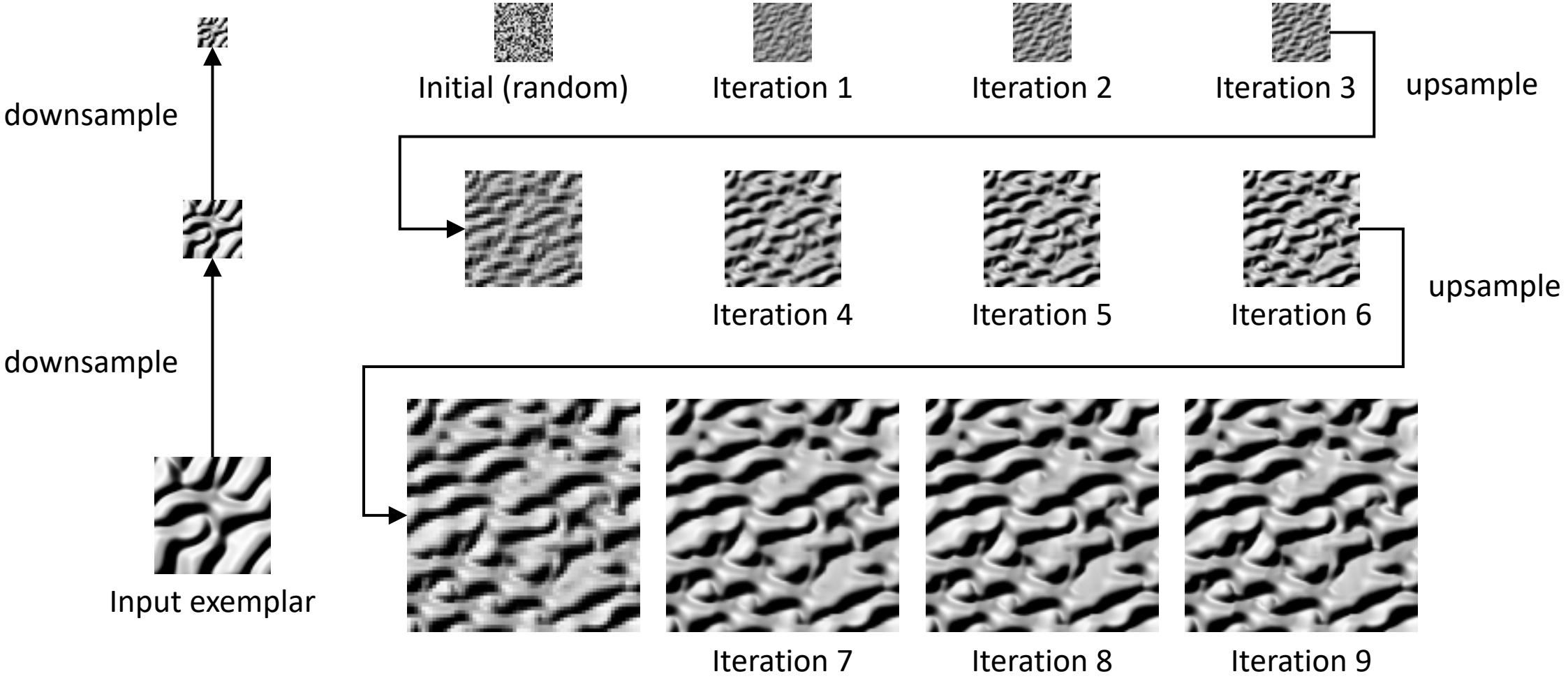
Texture optimization for example-based synthesis [Kwatra SIGGRAPH05]

Optimization by iterative computation [Kwatra05]



Texture optimization for example-based synthesis [Kwatra SIGGRAPH05]

Multiresolution synthesis

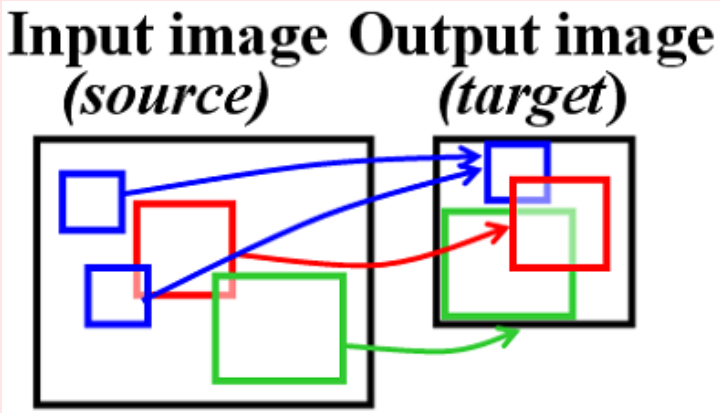


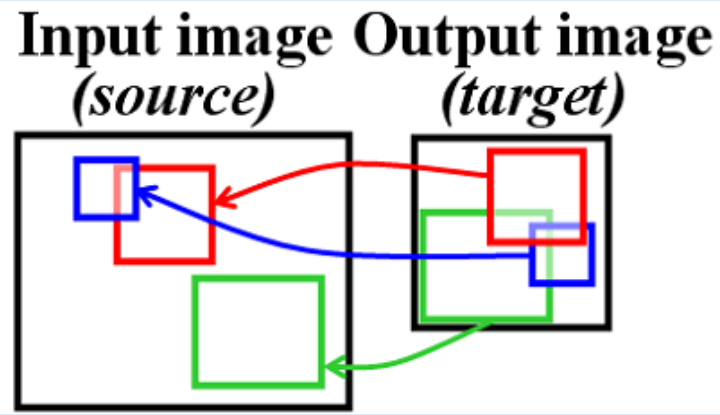
Bidirectional similarity [Simakov08; Wei08]

$$D(S,T) = \sum_{s \subset S} \min_{t \subset T} \|s - t\|^2 + \lambda \sum_{t \subset T} \min_{s \subset S} \|s - t\|^2$$

Completeness term

Coherence term



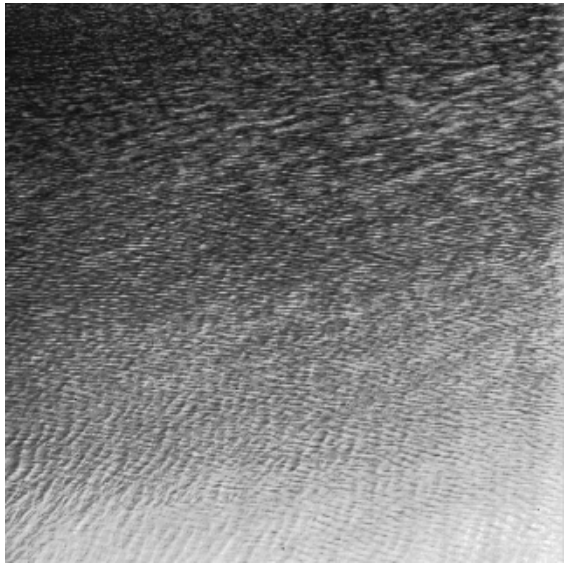


Inverse texture synthesis [Wei SIGGRAPH08]
Summarizing Visual Data Using Bidirectional Similarity [Simakov CVPR08]

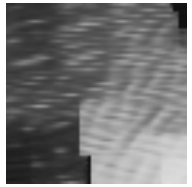
Effect of Completeness/Coherence terms

problem a.k.a.
“Image Summarization”

Input image

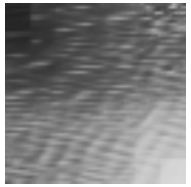


Output image



Completeness only

$$\sum_{s \subset S} \min_{t \subset T} \|s - t\|^2$$



Bidirectional

Coherence only

$$\sum_{t \subset T} \min_{s \subset S} \|s - t\|^2$$

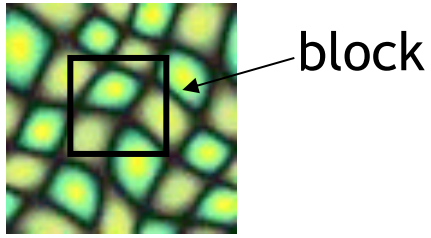
Bidirectional

Inverse texture synthesis [Wei SIGGRAPH08]

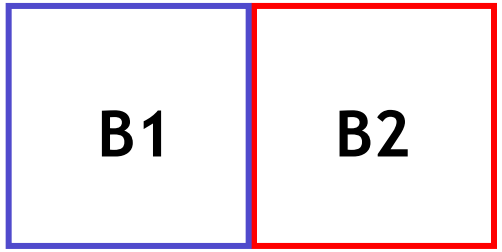
Texture synthesis via stitching of patches

(briefly)

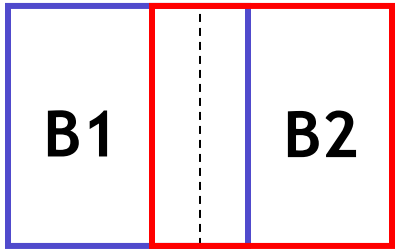
Image Quilting [Efros01]



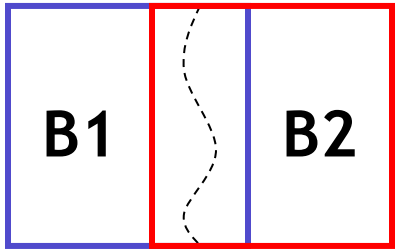
Input texture



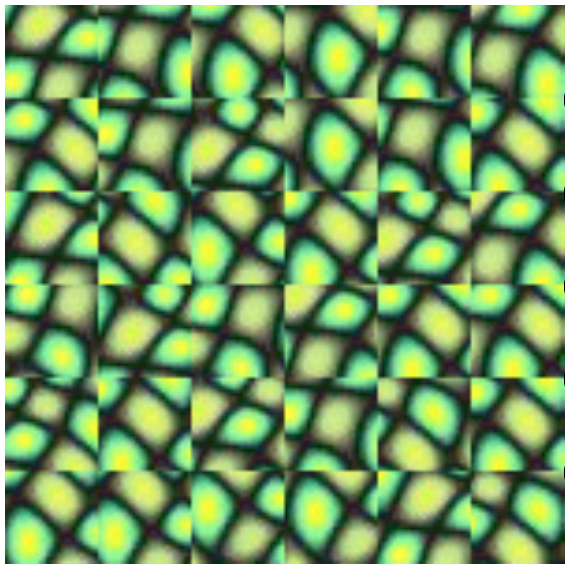
Random placement of blocks

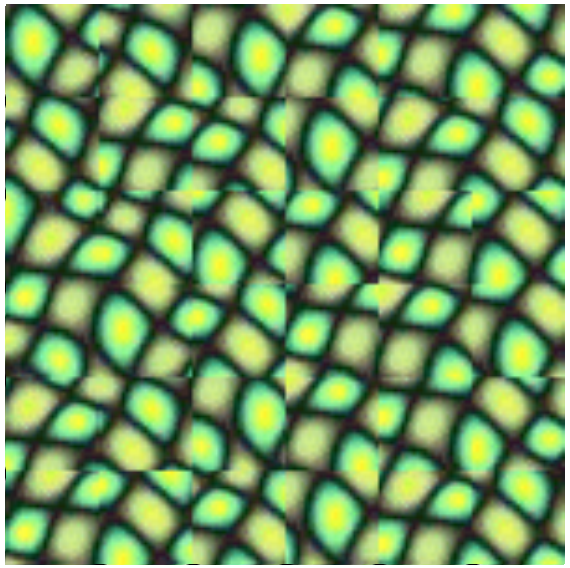


Neighboring blocks constrained by overlap



Minimal error boundary cut





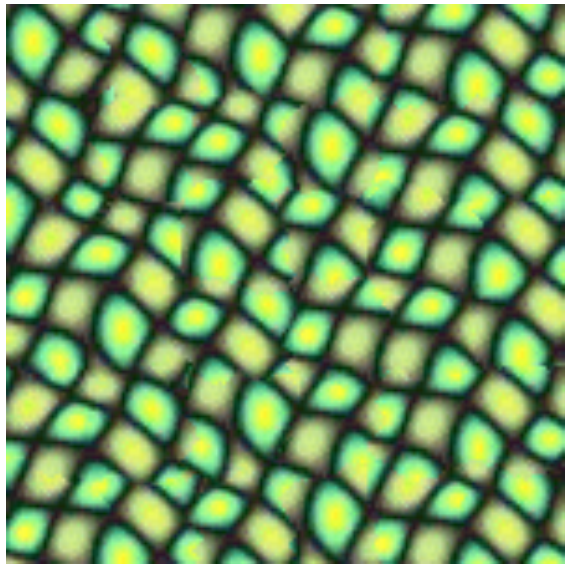


Image quilting for texture synthesis and transfer [Efros SIGGRAPH01]

Image Quilting [Efros01]

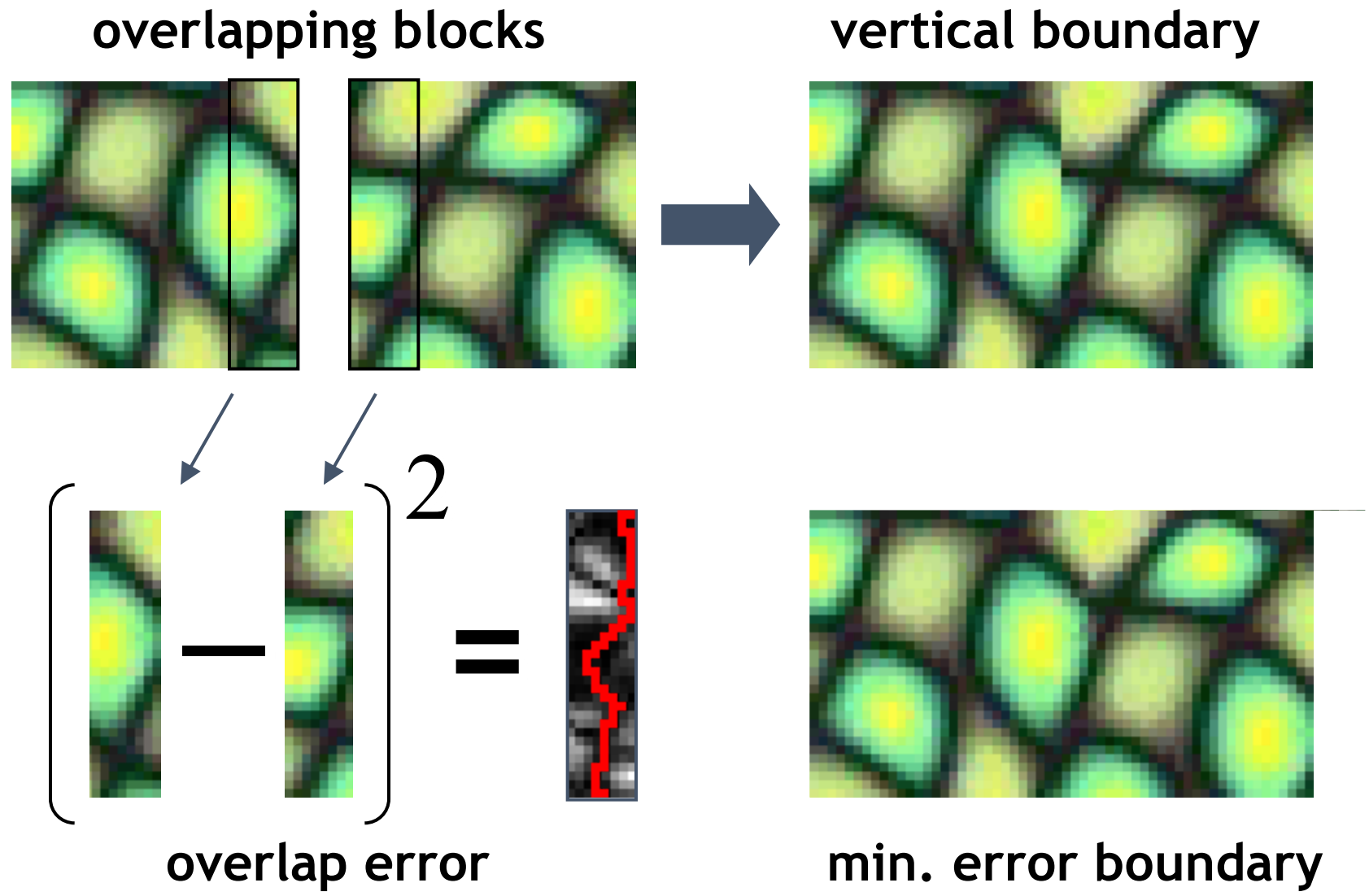
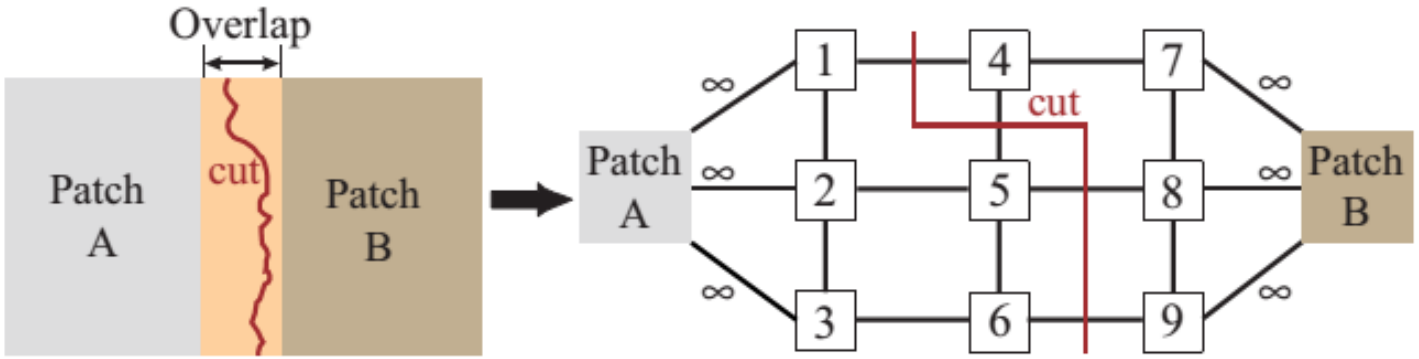
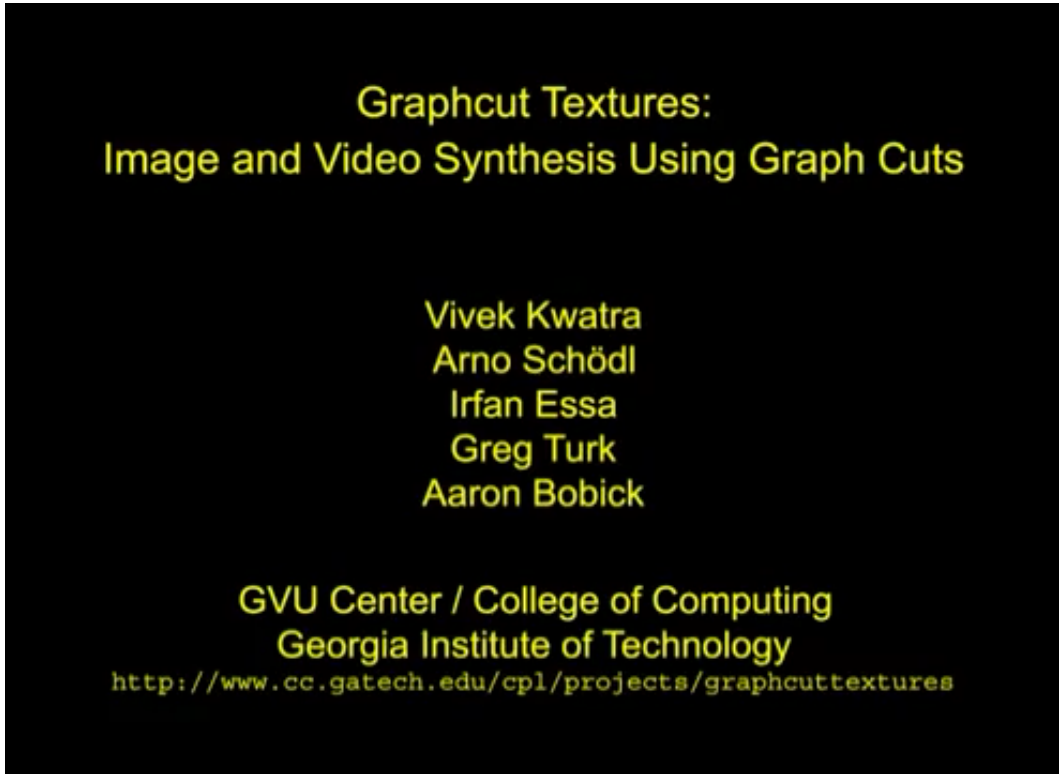


Image quilting for texture synthesis and transfer [Efros SIGGRAPH01]

Graphcut Textures [Kwatra03]





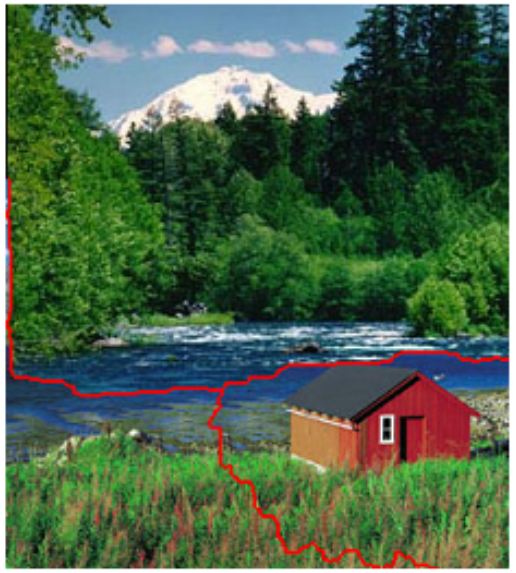
https://www.youtube.com/watch?v=Ya6BshBH6G4

- Formulate the best seam between patches as a minimum-cost cut in a graph

Graphcut Textures: Image and Video Synthesis Using Graph Cuts [Kwatra SIGGRAPH03]

Graphcut Textures [Kwatra03]



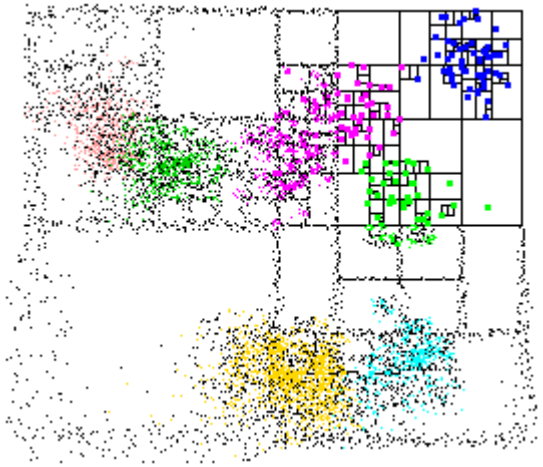


Graphcut Textures: Image and Video Synthesis Using Graph Cuts [Kwatra SIGGRAPH03]

Acceleration techniques for nearest neighborhood search

Technique #1:
Spatial data structure + dimensionality reduction

- 5x5 neighbor pixels each with RGB channels
  ➔ 75D vector



- Nearest neighbor search in high dimensional space
  ➔ Acceleration using k-d tree

- k-d tree performs poorly when dimensionality is too high
  ➔ Dimensionality reduction using
  Principal **C**omponent **A**nalysis

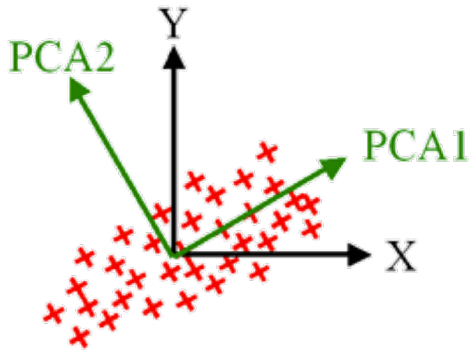
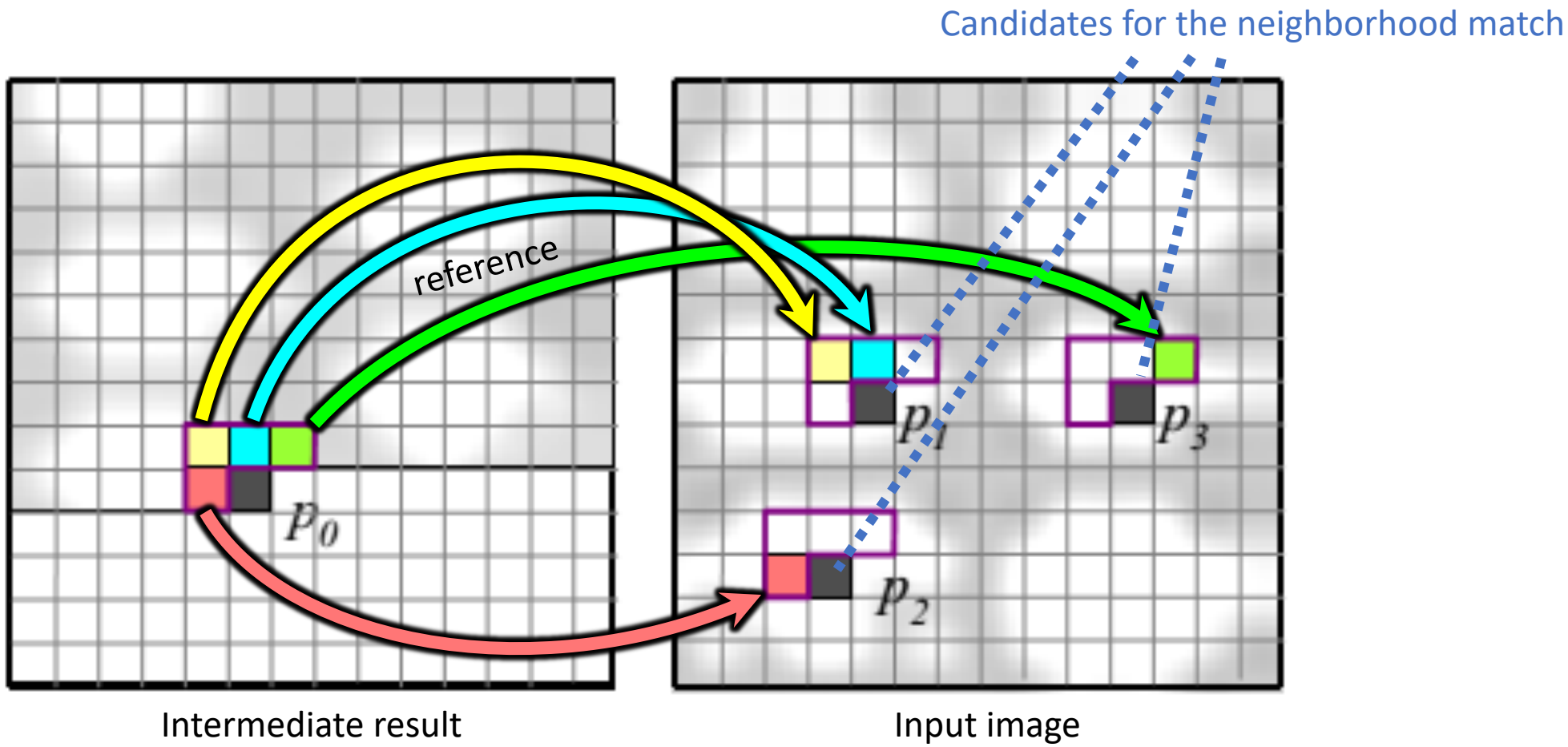


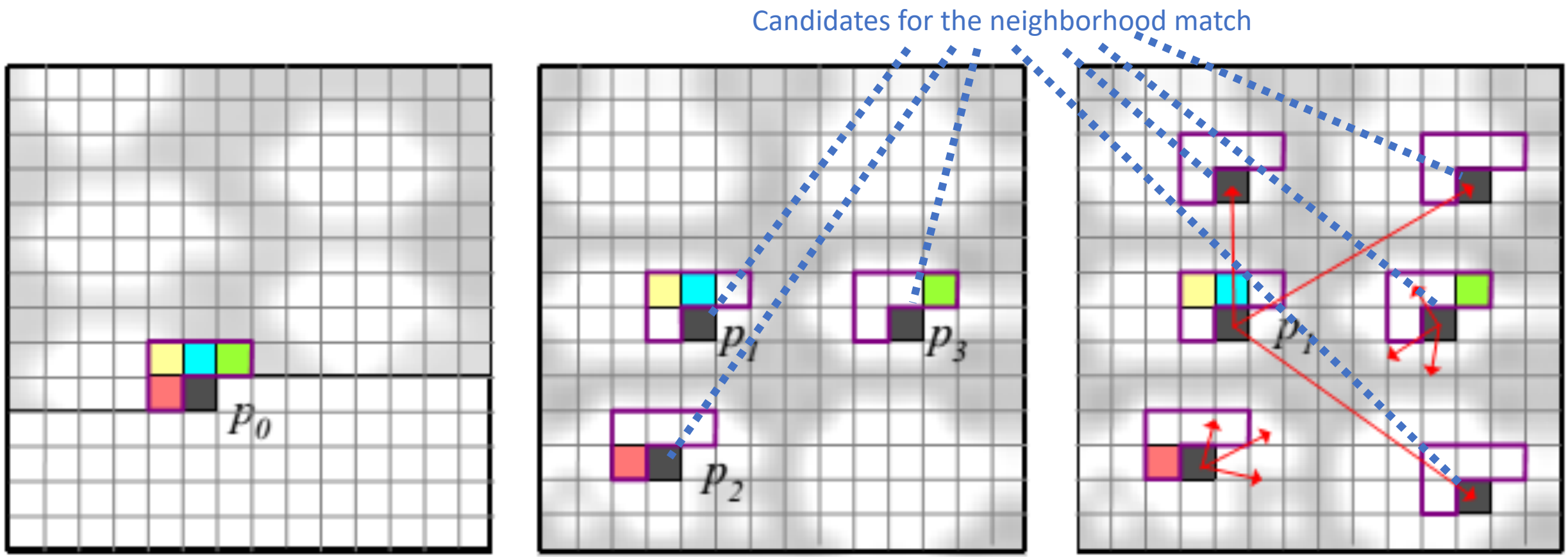
Image analogies [Hertzmann SIGGRAPH01]
https://www.cs.umd.edu/~mount/ANN/

Technique #2: k-coherence [Tong02]



Synthesis of bidirectional texture functions on arbitrary surfaces [Tong SIGGRAPH02]

Technique #2: k-coherence [Tong02]



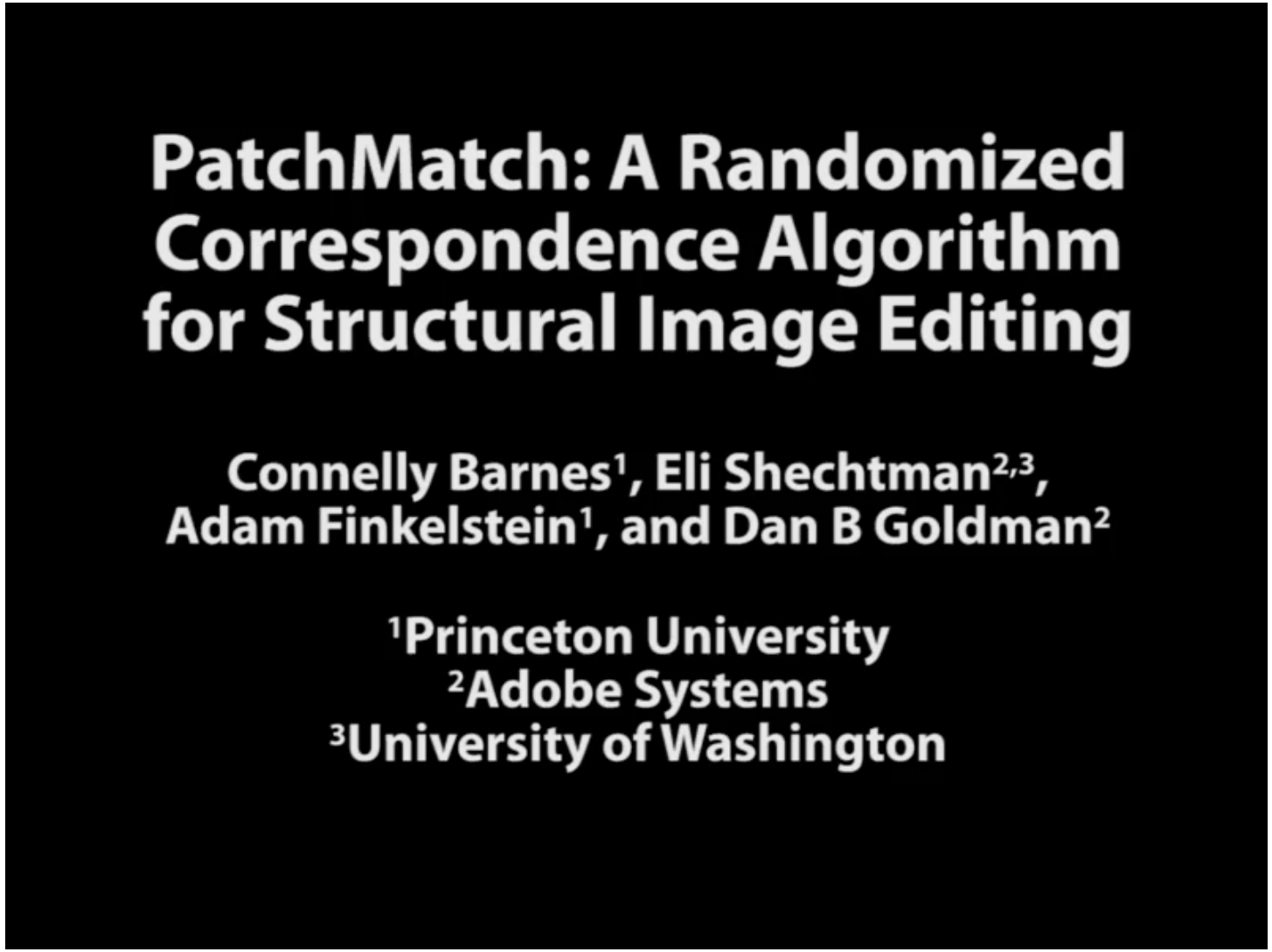
Intermediate result

Input image

Precomputed set of
k similar neighborhoods

Synthesis of bidirectional texture functions on arbitrary surfaces [Tong SIGGRAPH02]

Best bet: PatchMatch [Barnes09]

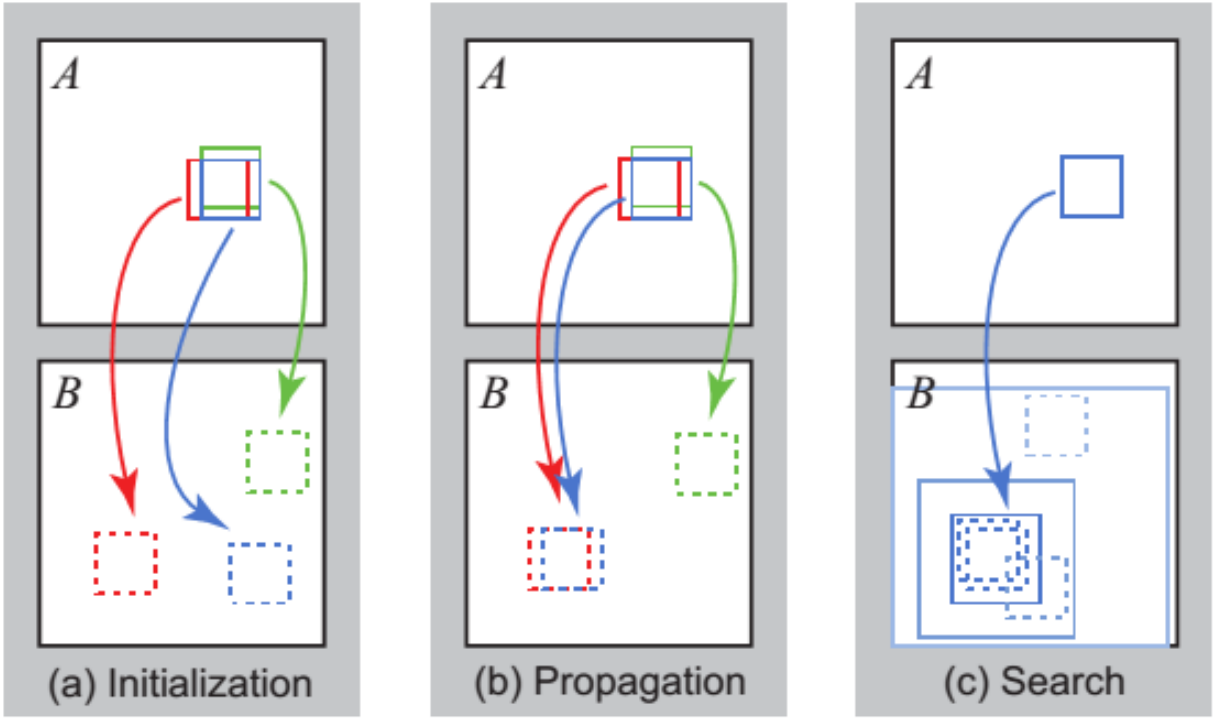


https://www.youtube.com/watch?v=dgKjs8ZjQNg

PatchMatch: a randomized correspondence algorithm for structural image editing [Barnes SIGGRAPH09]

Best bet: PatchMatch [Barnes09]

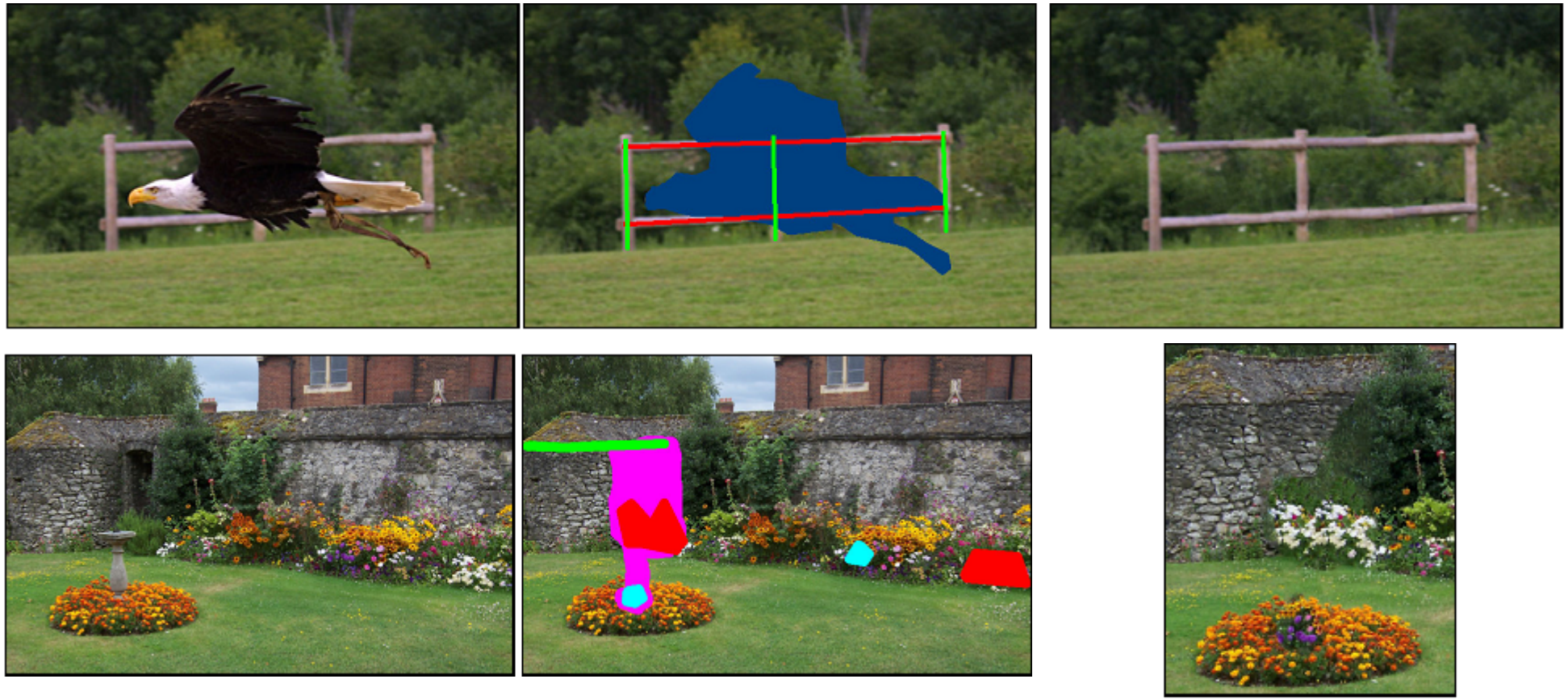
- Randomly initialize matches
- Update matches in scanline order
  - **Propagation:**
    Accept either left or above match if it's better than the current match
  - **Random Search:**
    Try a few random matches; accept if it's better than the current match
- Demo



PatchMatch: a randomized correspondence algorithm for structural image editing [Barnes SIGGRAPH09]

Extensions & applications

Synthesis control by limiting the search space



- Output pixels with markers only match with input pixels with the same markers

Image completion with structure propagation [Sun SIGGRAPH05]

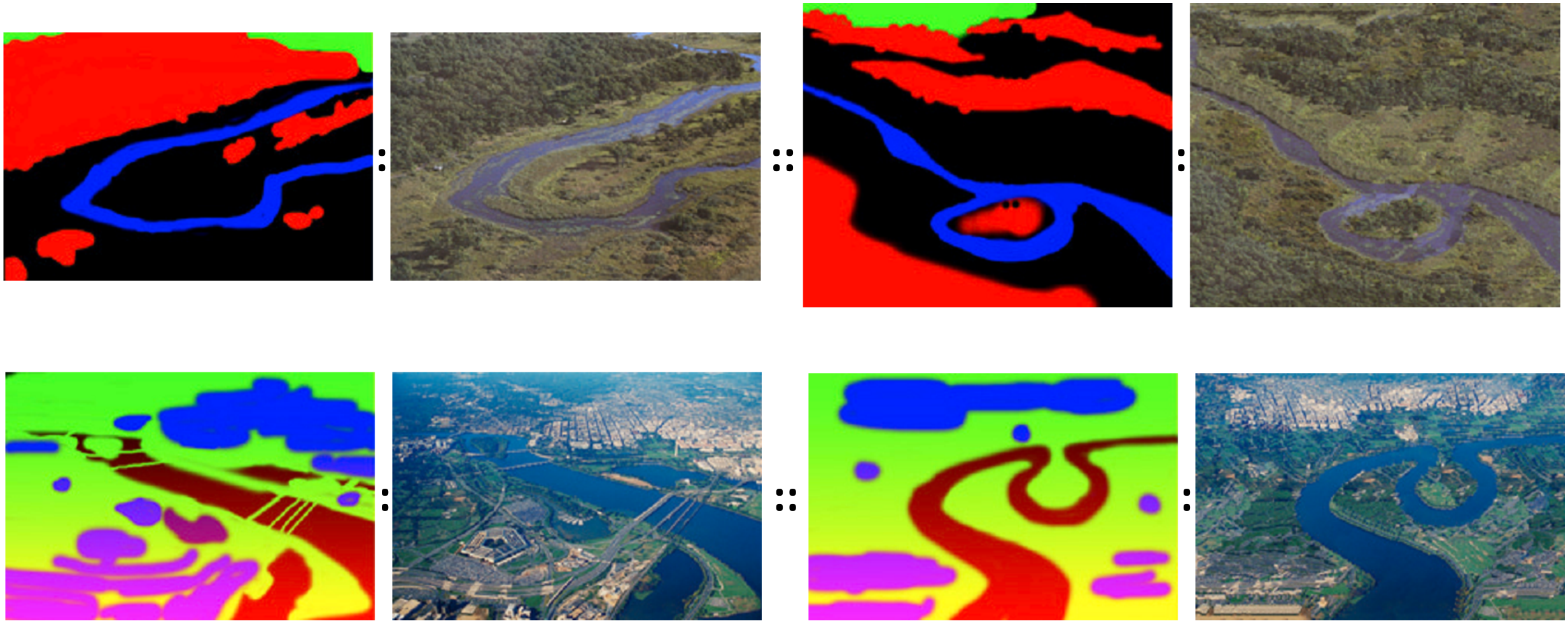
Image Analogies [Hertzmann01]

- Simulate arbitrary image filters using texture synthesis

- Variety of applications possible with this formulation

Image analogies [Hertzmann SIGGRAPH01]

Image Analogies – Texture by Numbers



A A’ B B’

Image analogies [Hertzmann SIGGRAPH01]

Image Analogies – Super Resolution

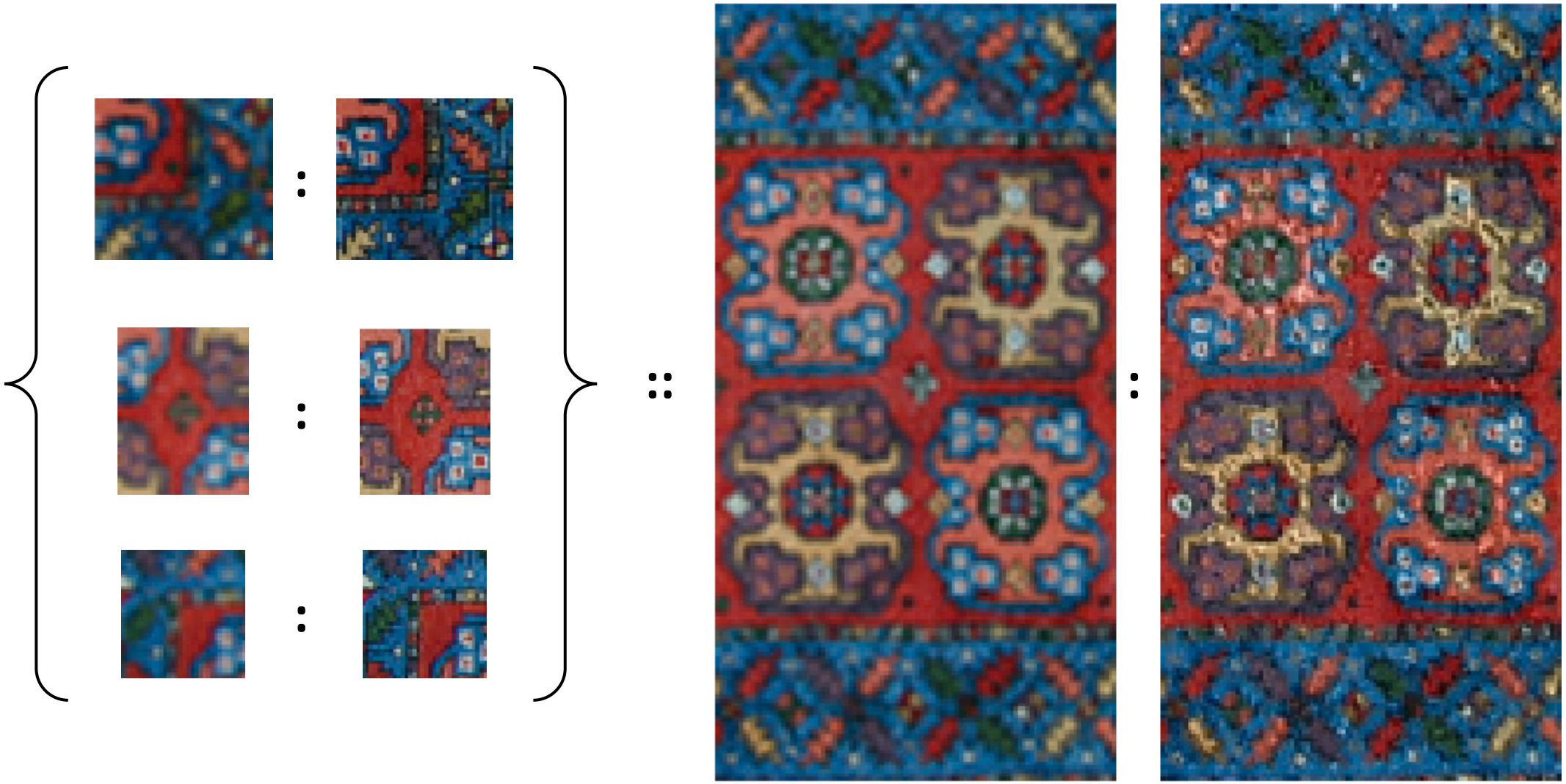
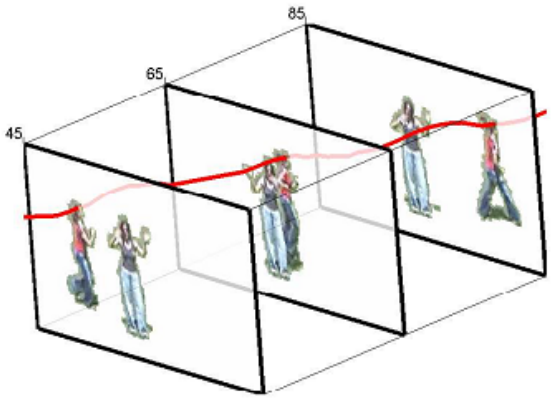


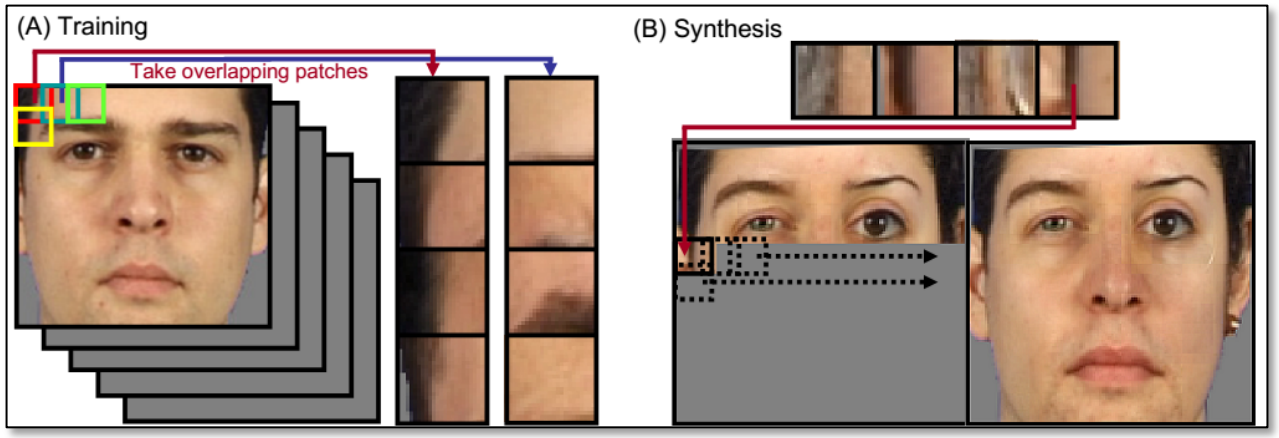
Image analogies [Hertzmann SIGGRAPH01]

Removal of objects in videos



Space-time video completion [Wexler CVPR04]

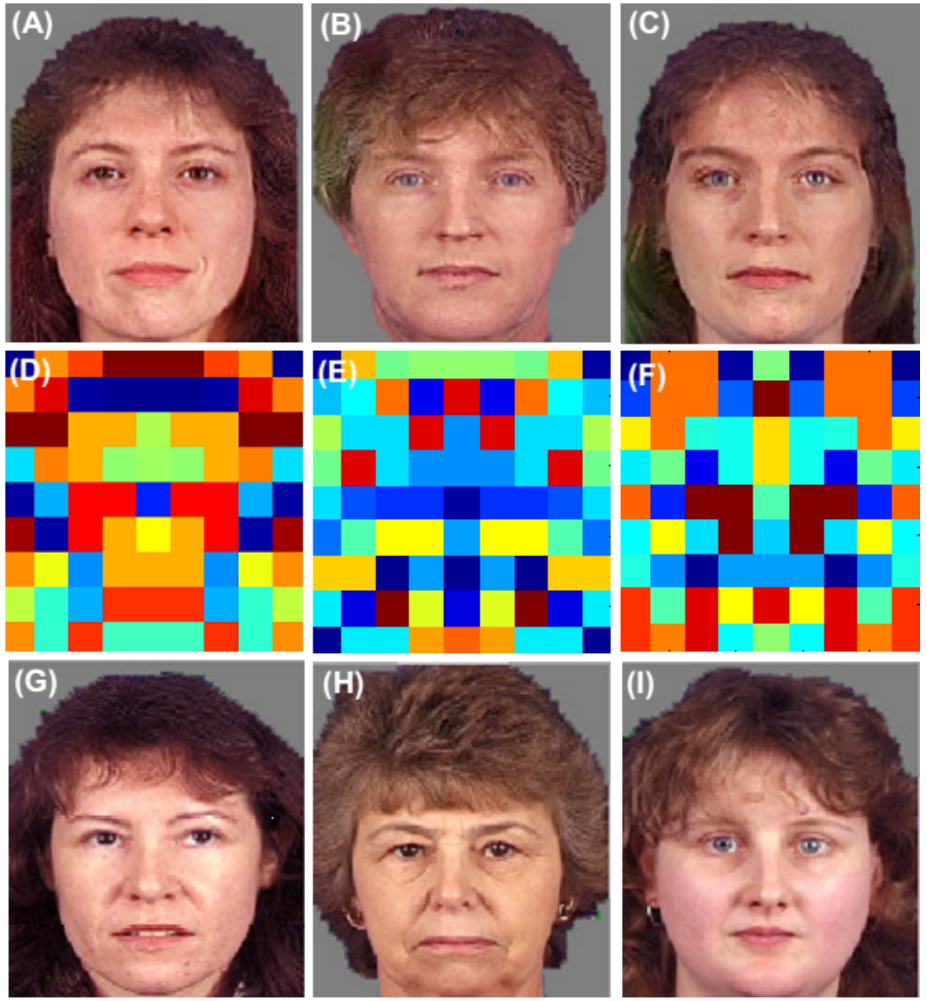
Random synthesis of face images [Mohammed09]



Naïve synthesis from face images with positional alignment

Parametric model for “average faces”

Synthesis result



Training images closest to synthesis results

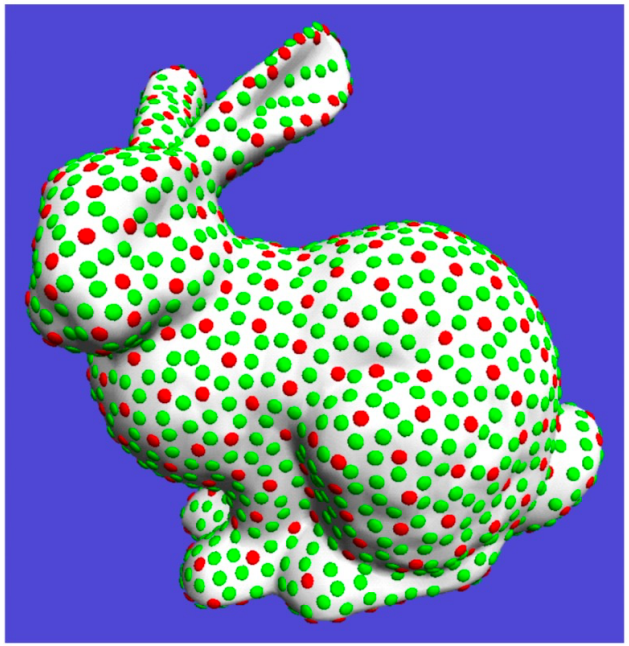
Visio-lization: generating novel facial images [Mohammed SIGGRAPH09]

Random synthesis of structured images [Risser10]

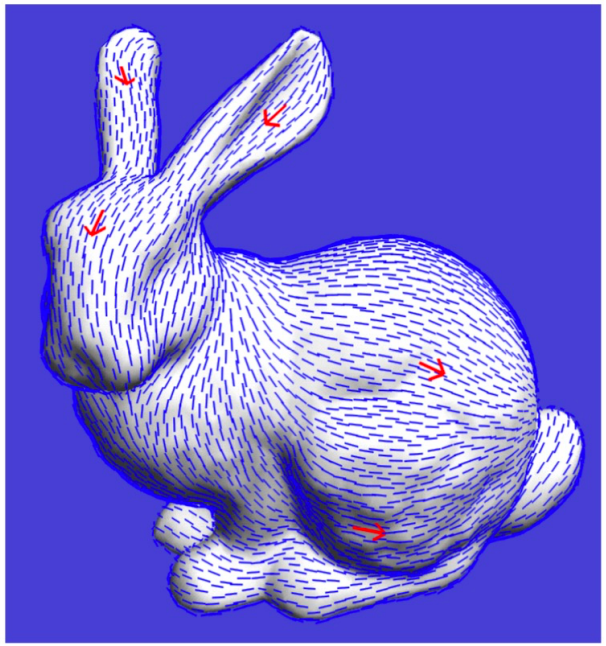
Synthesizing Structured Image Hybrids [Risser SIGGRAPH10]

Texture synthesis for 3D graphics

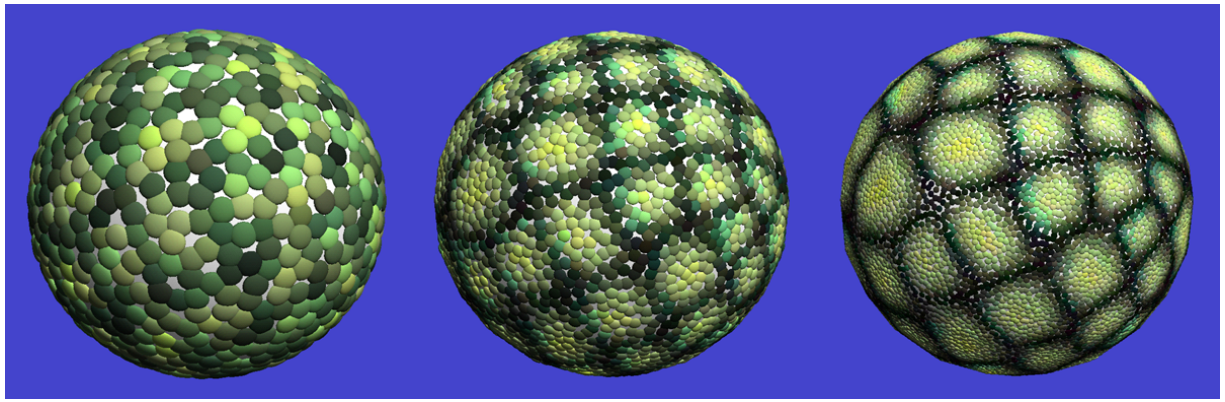
On-surface texture synthesis [Wei01; Turk01]



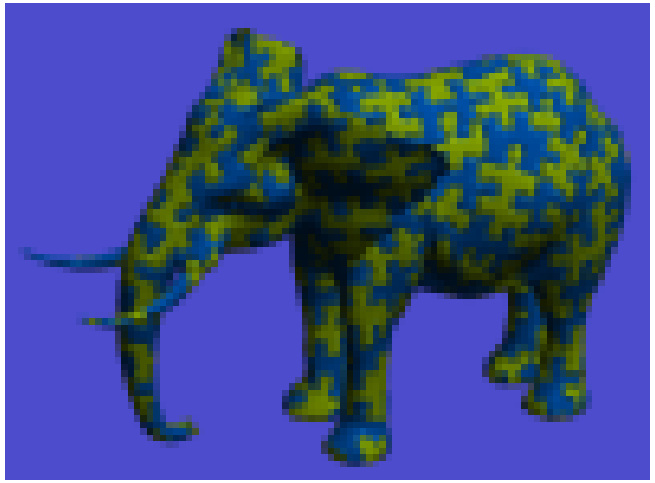
Uniform sample points



Vector field



Multiresolution synthesis



- Fundamentally equivalent to synthesizing texture images over UV parameter space

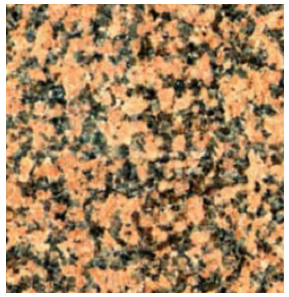
Texture synthesis over arbitrary manifold surfaces [Wei SIGGRAPH01]
Texture synthesis on surfaces [Turk SIGGRAPH01]

Solid textures

- Represent texture as 3D volume (e.g. voxel) of RGB
  - RGB color directly obtained from XYZ coord
    ➔ easy to use!

$$marble(x,y,z) = colormap(\sin(x + \text{noise}(x,y,z)))$$

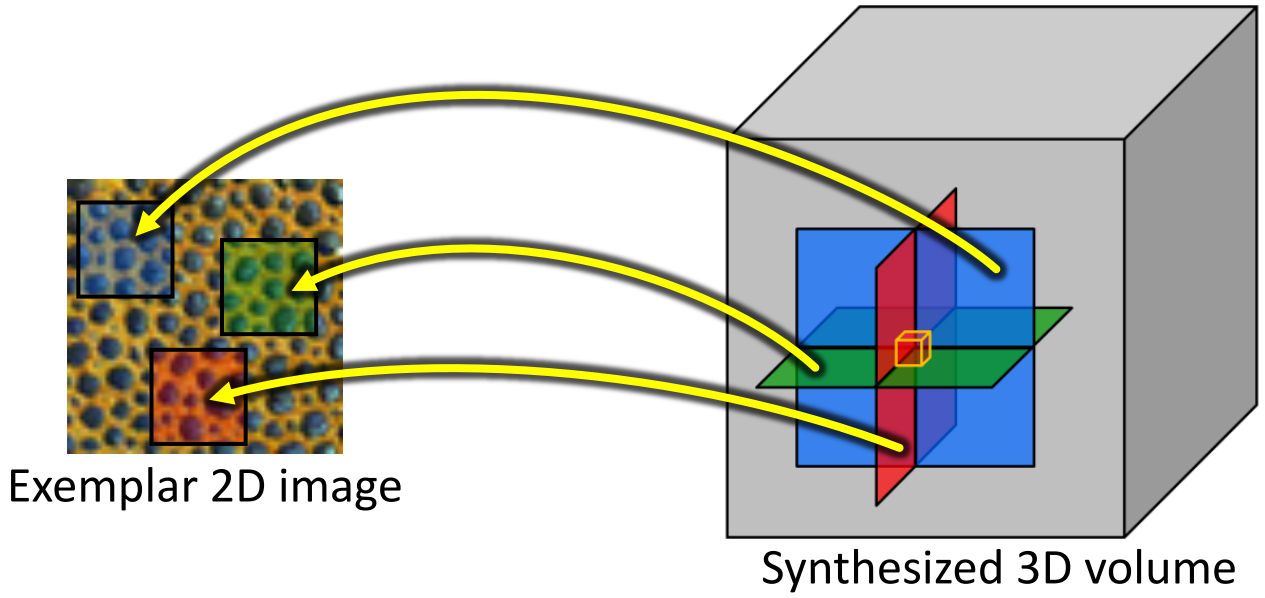
- Early methods
  - Combine noise functions, tweak parameters
  - Automatic example-based synthesis using statistical approaches
    - Limited to noise-like textures



An image synthesizer [Perlin SIGGRAPH85]
Pyramid-based texture analysis and synthesis [Heeger SIGGRAPH95]

Solid texture synthesis by optimization

- Almost straightforward generalization of 2D version [Kwatra05] to 3D



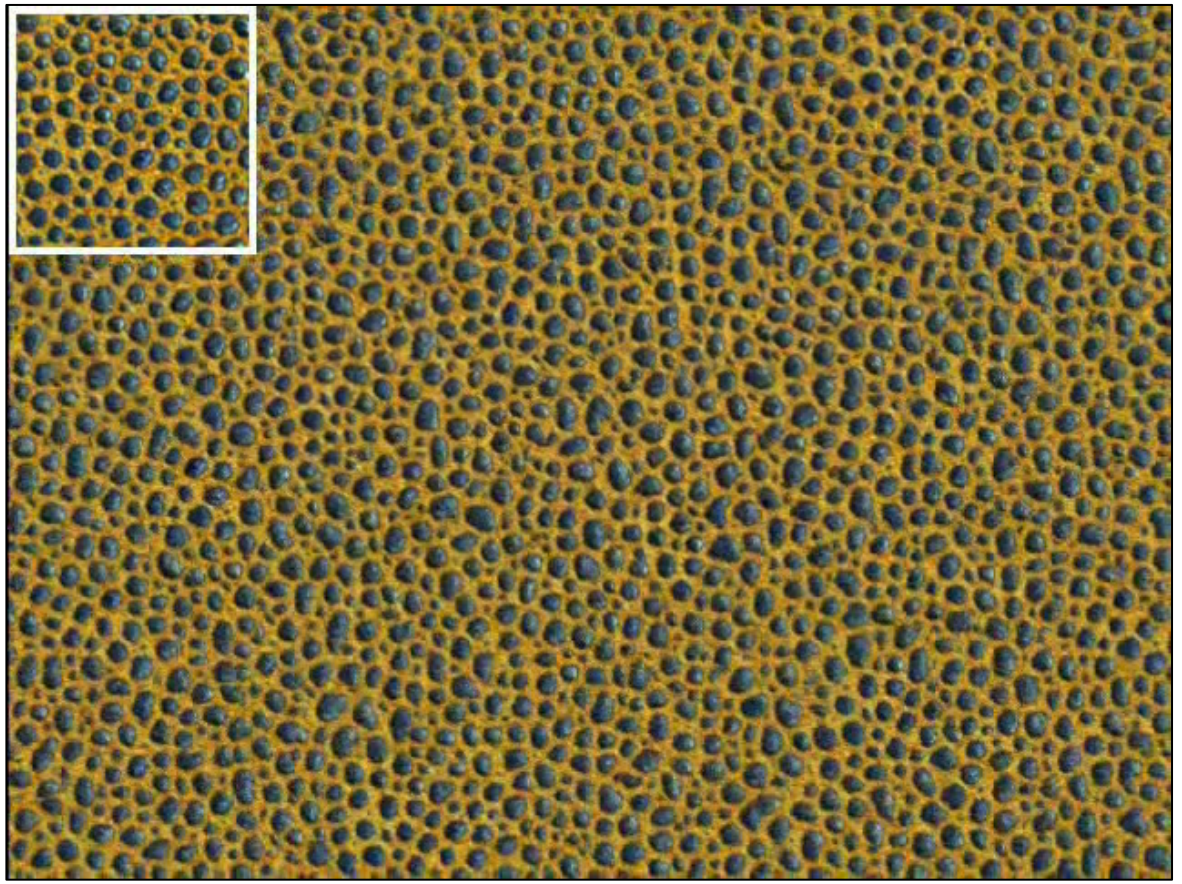
(Some tricks needed for better quality)

Solid texture synthesis from 2d exemplars [Kopf SIGGRAPH07]

Fast on-demand synthesis using GPU parallelism [Lefebvre05]

- Basic idea similar to [Kwatra05]
  - Key technique: precomputation + parallel independent processing

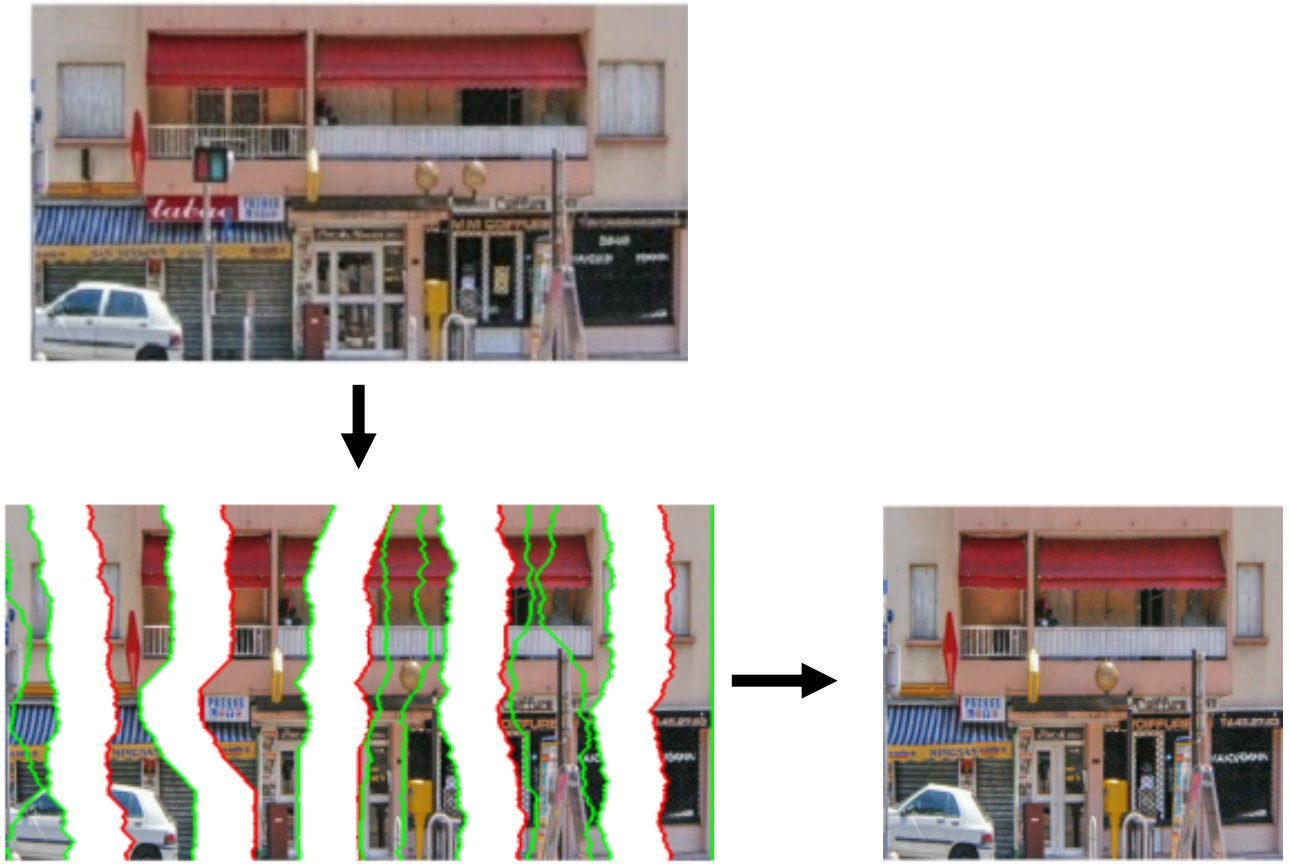
- Synthesize only when drawing
  = Reduced memory consumption
  ➔ suited for games



Parallel controllable texture synthesis [Lefebvre SIGGRAPH05]

On-demand synthesis specific to façade images [Lefebvre10]

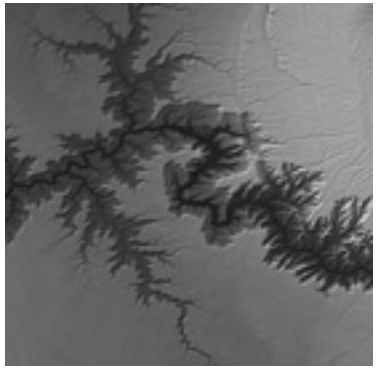
- Precompute horizontal/vertical seams ➔ combine at runtime on GPU



By-example Synthesis of Architectural Textures [Lefebvre SIGGRAPH10]

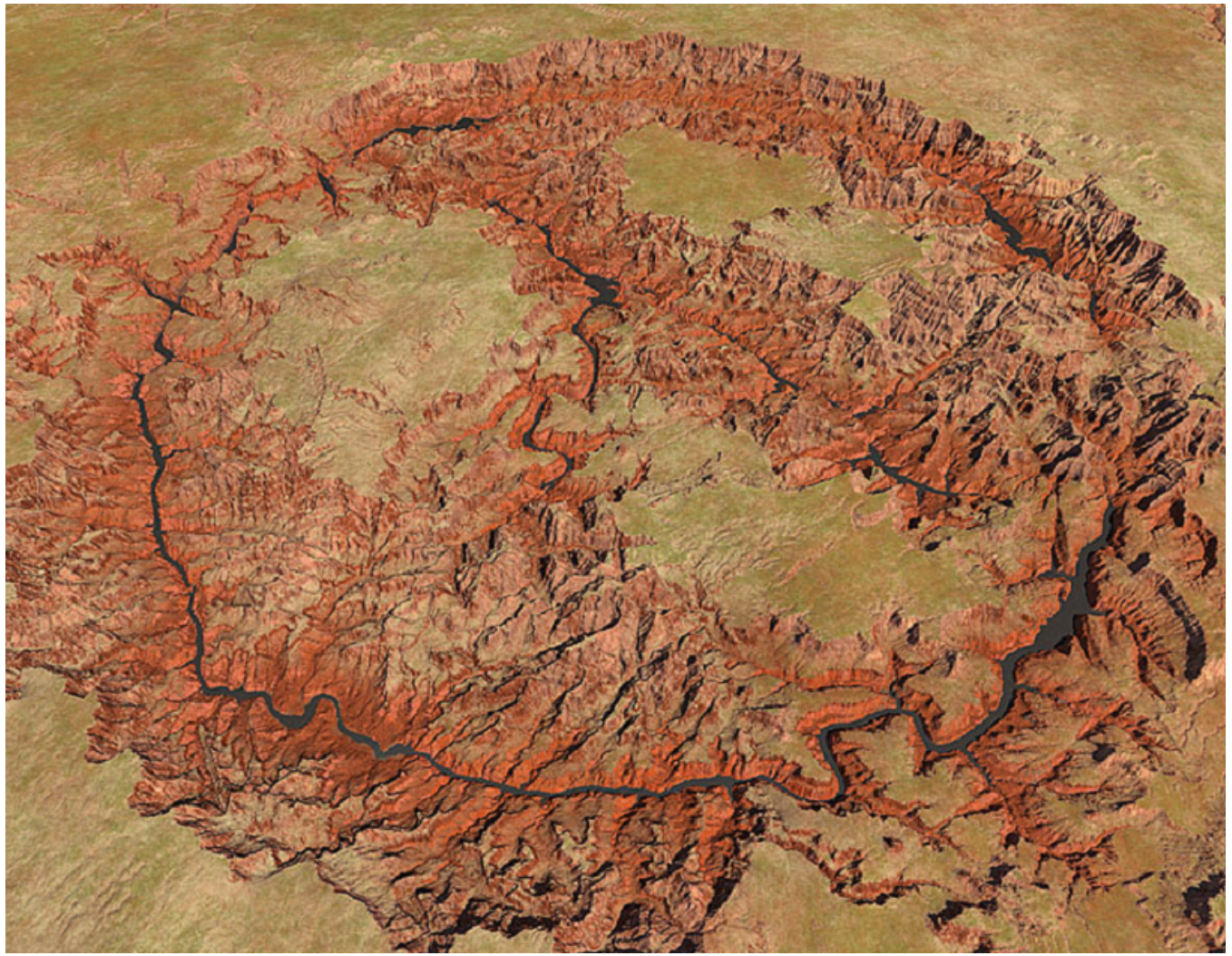
Applications of texture synthesis outside image processing

Terrain (height field) synthesis [Zhou07]



Geographical data

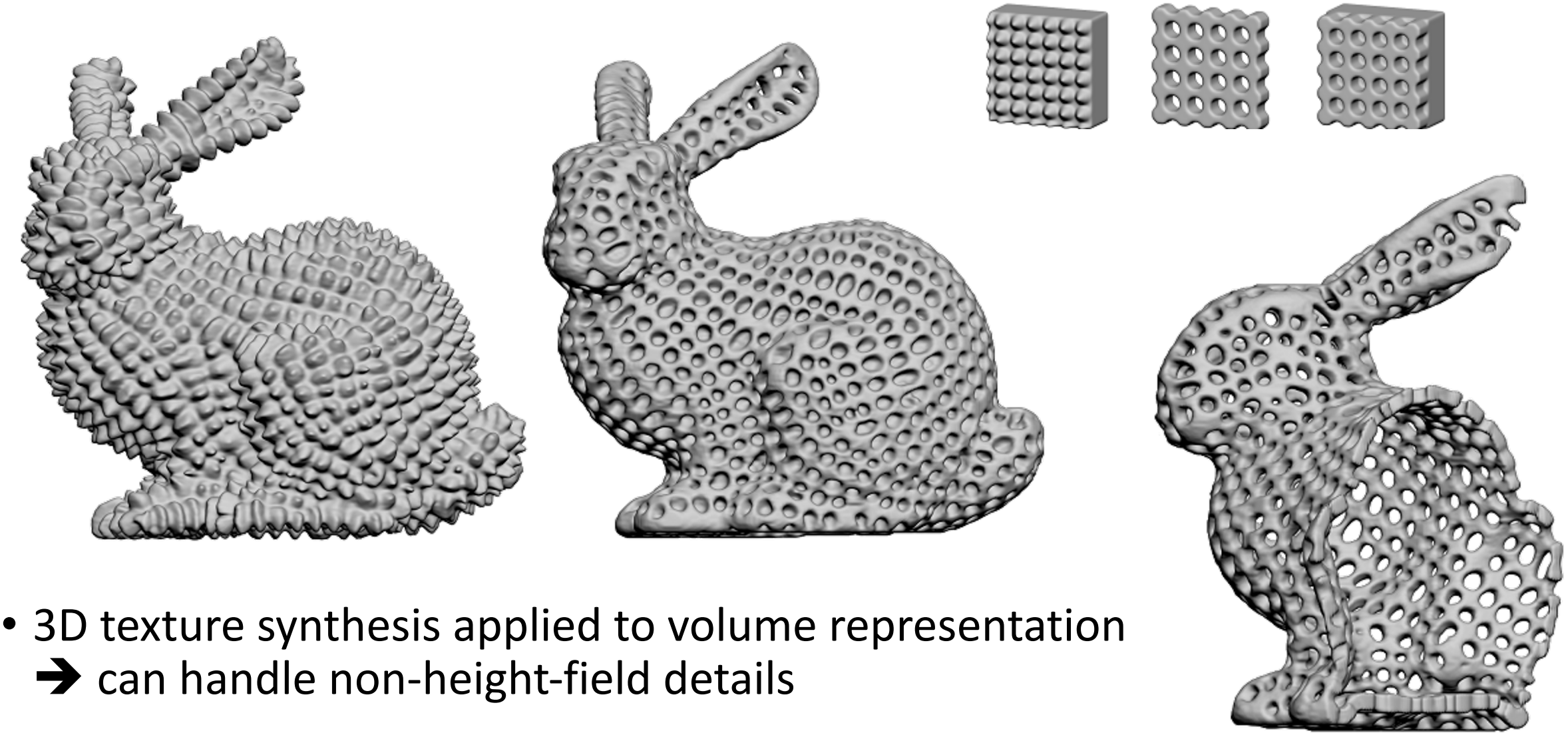
User’s sketch



Synthesis result

Terrain synthesis from digital elevation models [Zhou TVCG07]

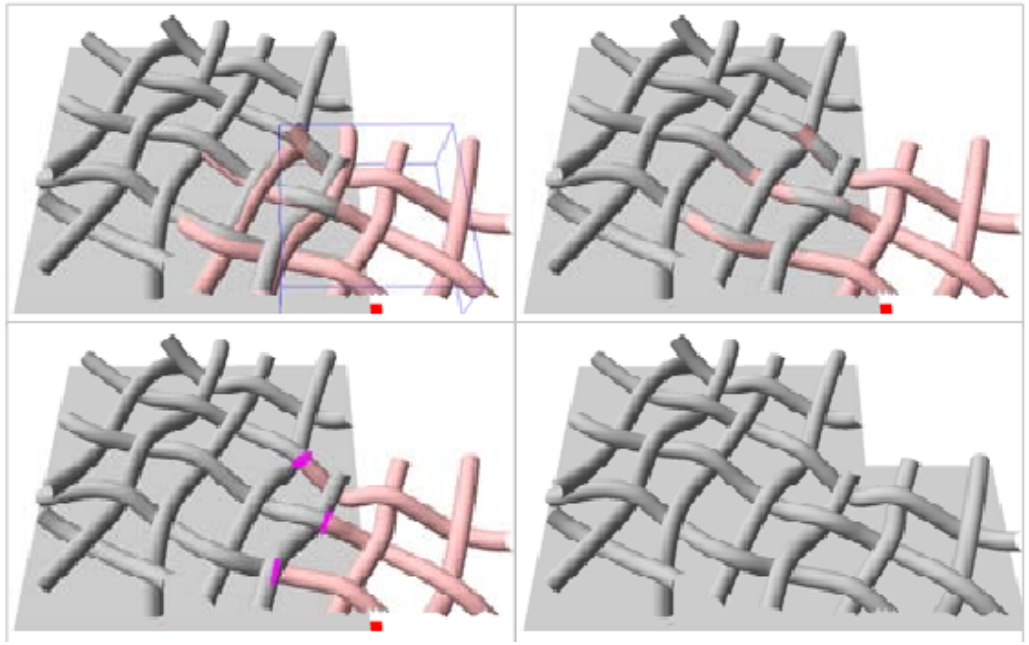
Synthesis of surface details [Bhat04]

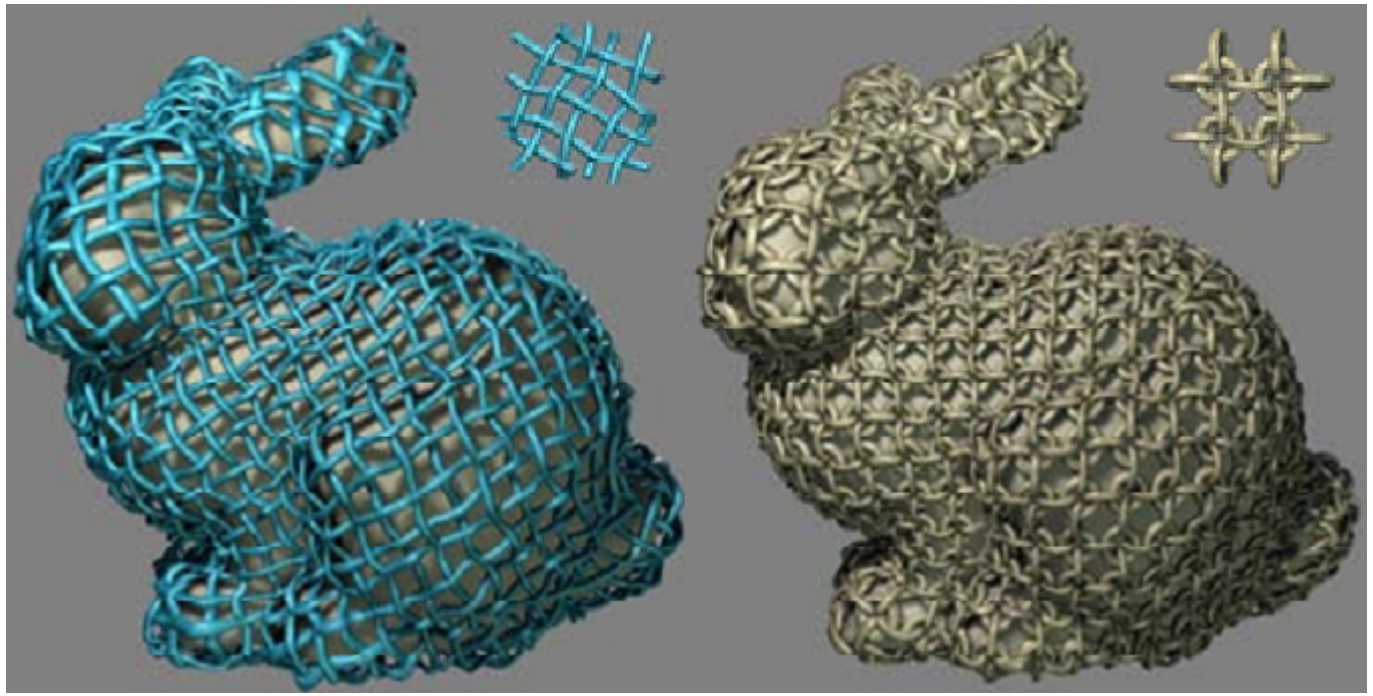


- 3D texture synthesis applied to volume representation
  ➔ can handle non-height-field details

Geometric texture synthesis by example [Bhat SGP04]

Mesh Quilting [Zhou06]

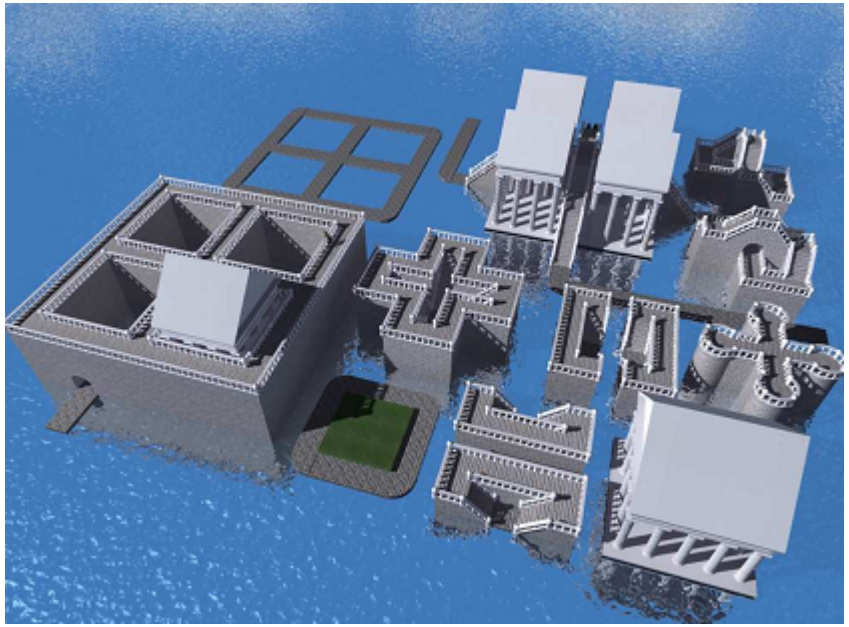


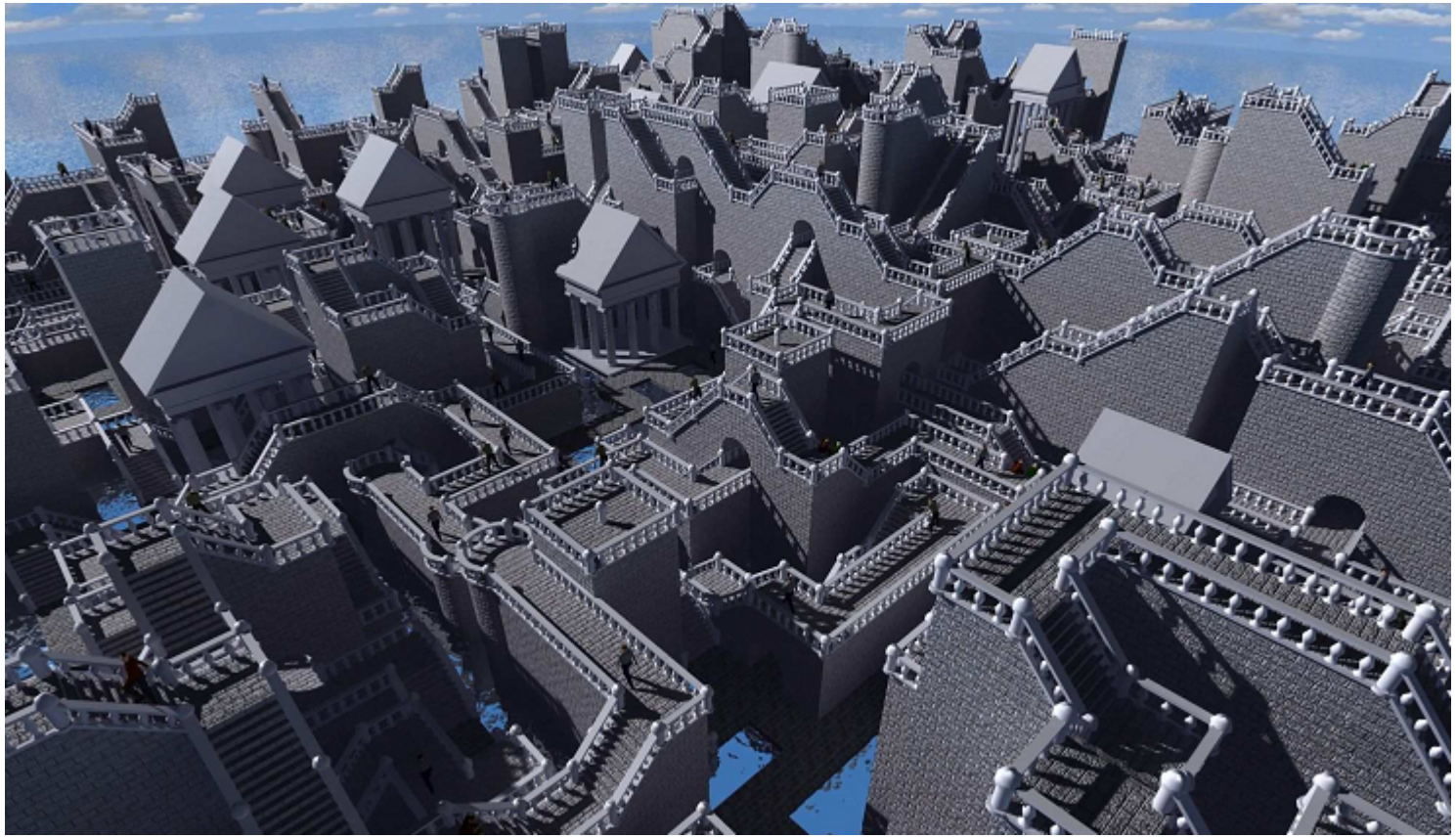


- Careful stitching of neighboring triangle meshes

Mesh quilting for geometric texture synthesis [Zhou SIGGRAPH06]

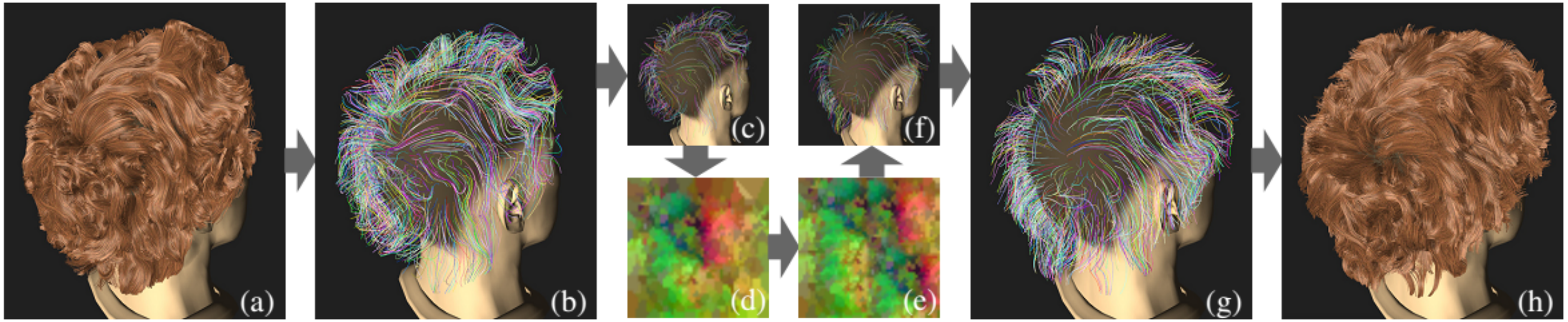
Synthesis of architectural models [Merrell07]





Example-based model synthesis [Merrell I3D07]

Hair synthesis [Wang09]



- Single hair strand = 3D polyline with N vertices = 3N dim vector
  ➔ Regarding this as color, apply texture synthesis

Example-based hair geometry synthesis [Wang SIGGRAPH09]

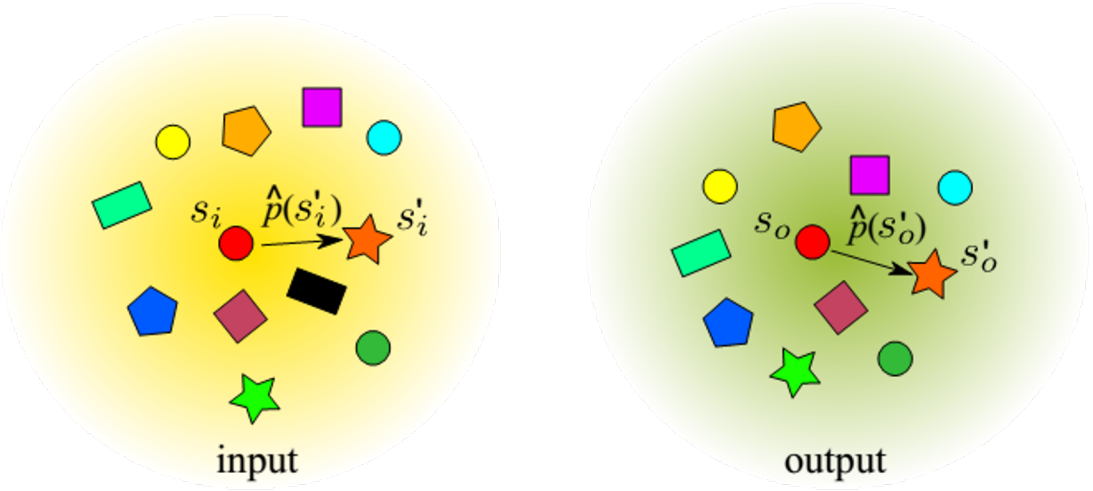
Synthesis of artistic vortices for fluids [Ma09]

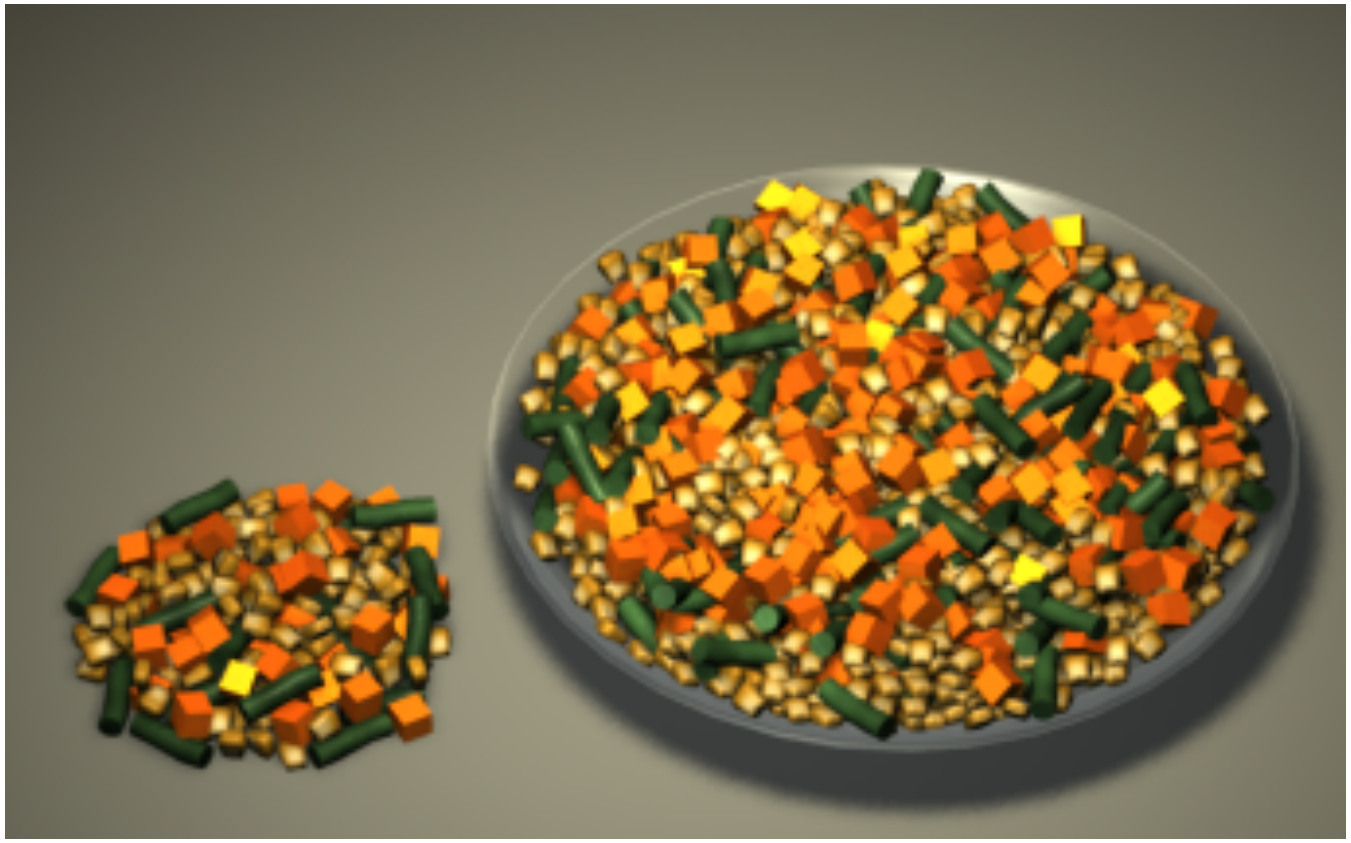
- Synthesize detailed vortex velocity field along input low-res velocity field
  - Regarding 2D/3D velocity vector as colors, apply texture synthesis

Motion field texture synthesis [Ma SIGGRAPHAsia09]

https://www.youtube.com/watch?v=TKOA_PBdt30

Synthesis of element arrangement [Ma11]





- Define similarity between distributions of sample points

- Optimization algorithm similar to [Kwatra05]

Discrete Element Textures [Ma SIGGRAPH11]

Pointers

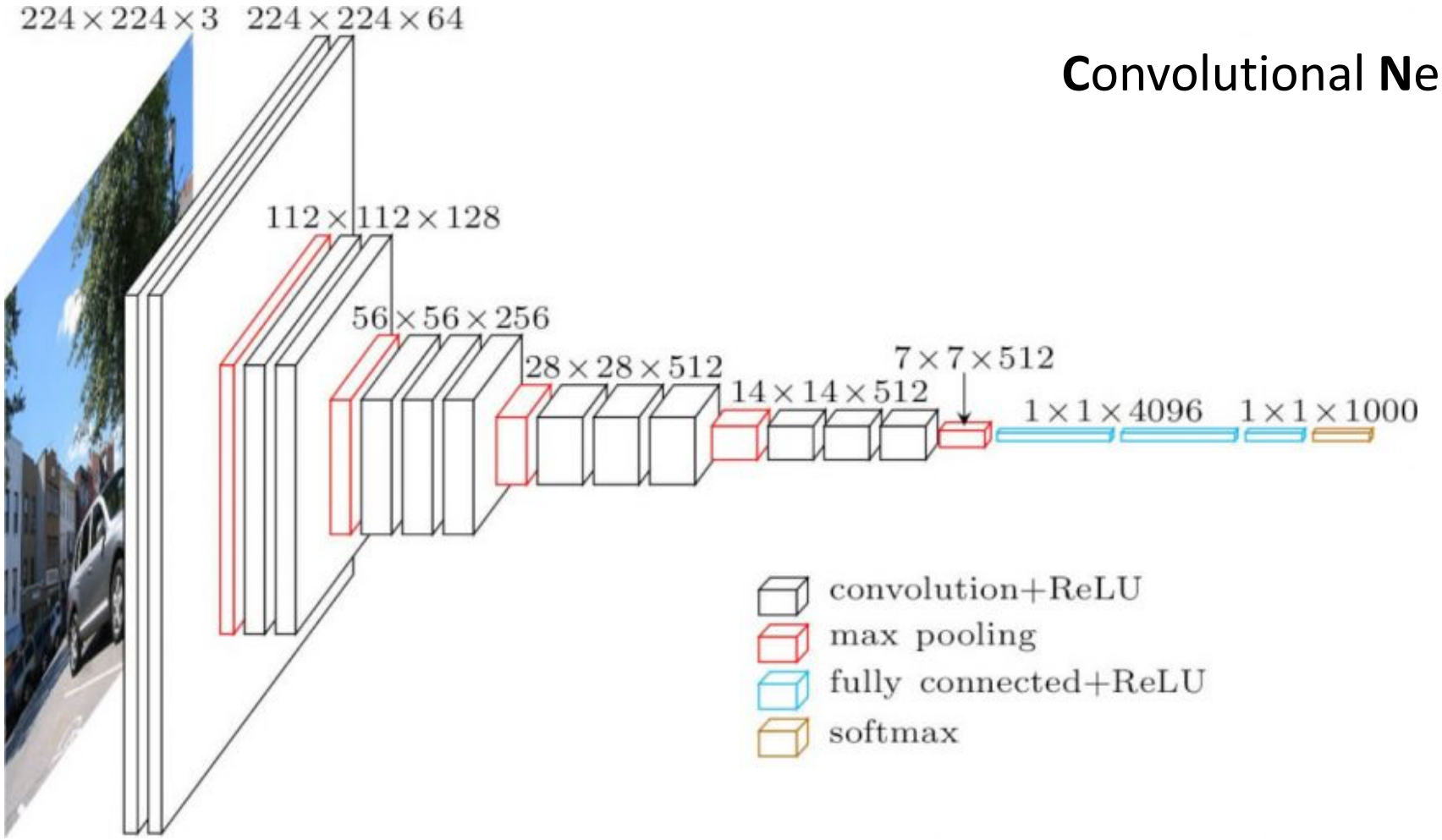
- Existing implementations
  - http://www2.mta.ac.il/~tal/ImageCompletion/ImageCompletion1.01.zip
  - http://www.cs.princeton.edu/gfx/pubs/Barnes_2009_PAR/patchmatch-2.1.zip
  - http://research.nii.ac.jp/~takayama/cggems12/cggems12.zip

- Surveys
  - State of the art in example-based texture synthesis [Wei EG09STAR]
  - Solid-Texture Synthesis; A Survey [Pietroni CGA10]

Extra: Texture synthesis based on **deep learning**

(Basics) VGG: CNN-based image classifier

Convolutional **N**eural **N**etwork

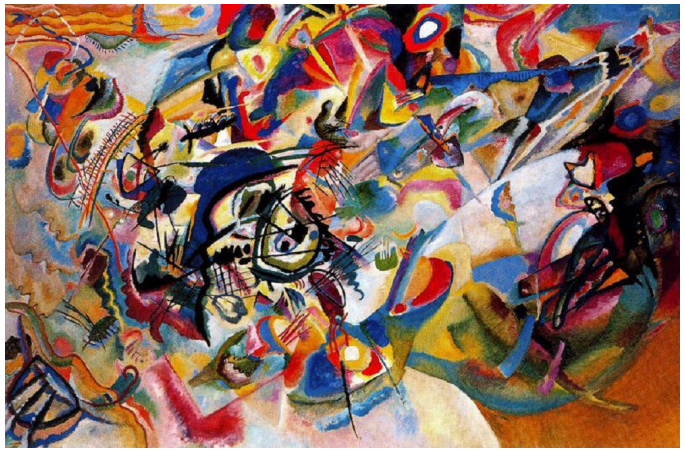


https://neurohive.io/en/popular-networks/vgg16/

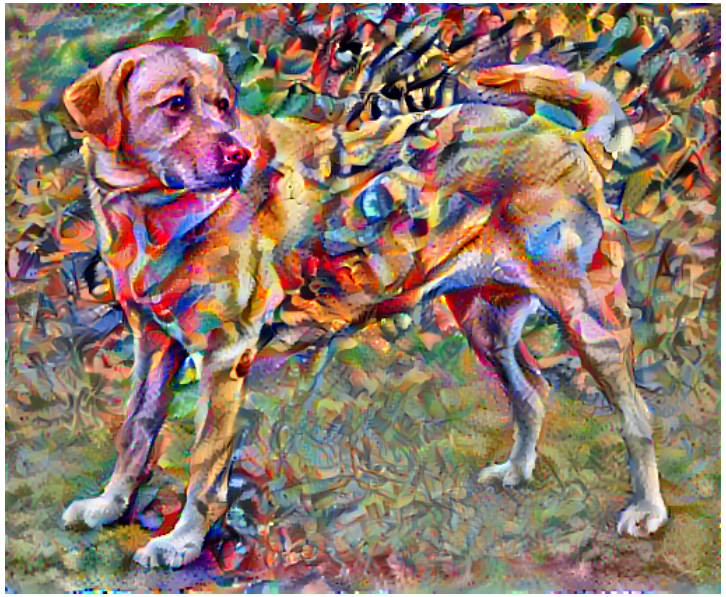
Very Deep Convolutional Networks for Large-Scale Image Recognition [Simonyan arXiv14]

(Basics) Neural style transfer

Content



Style



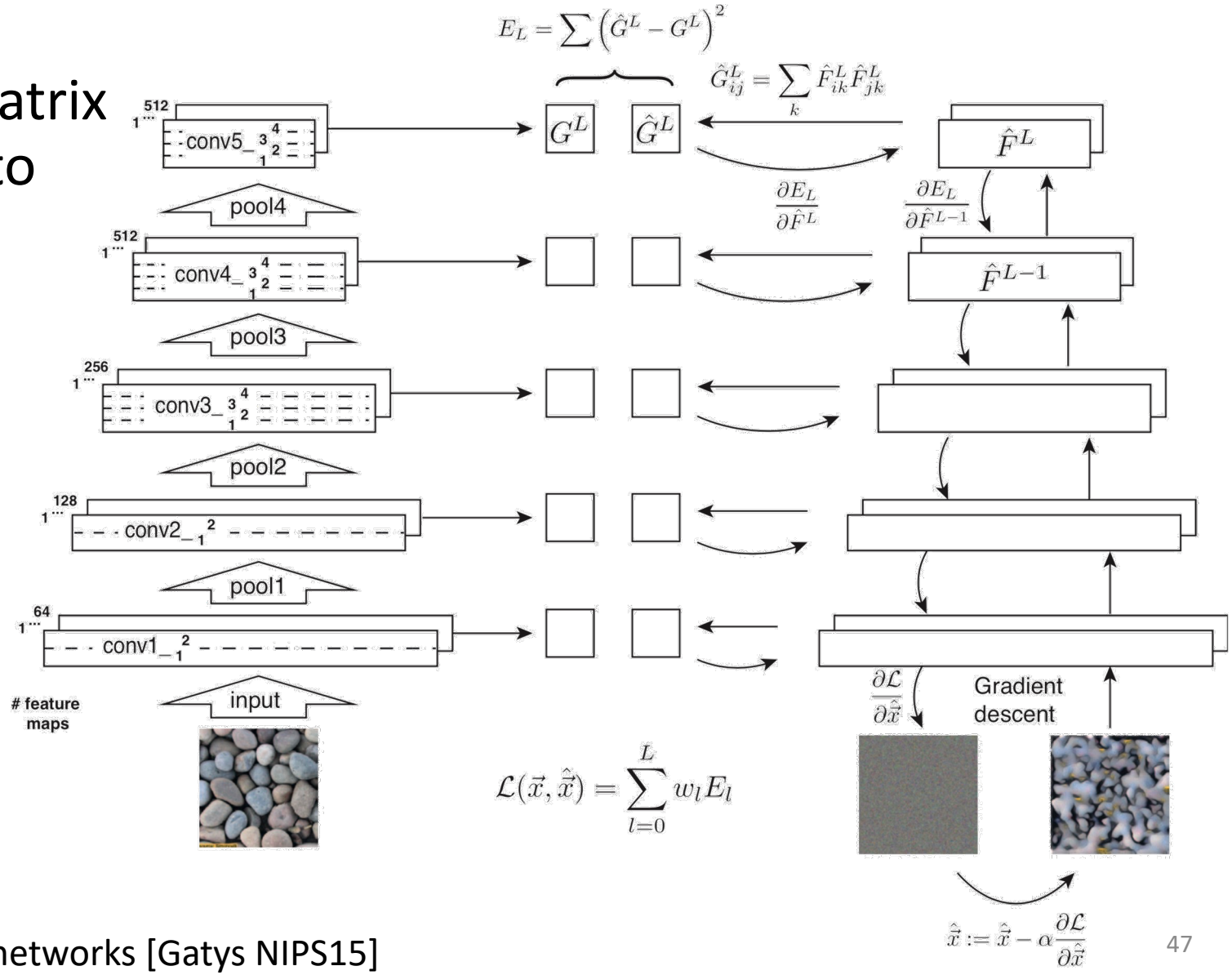
https://www.tensorflow.org/tutorials/generative/style_transfer

- Gram matrix: correlation between different feature channels output by VGG
- Start with noise, update pixels (with gradient descent) s. t. its feature responses will be close to those of the Content image, while its Gram matrices will be close to those of the Style image

A Neural Algorithm of Artistic Style [Gatys arXiv15]

Texture synthesis based on CNN

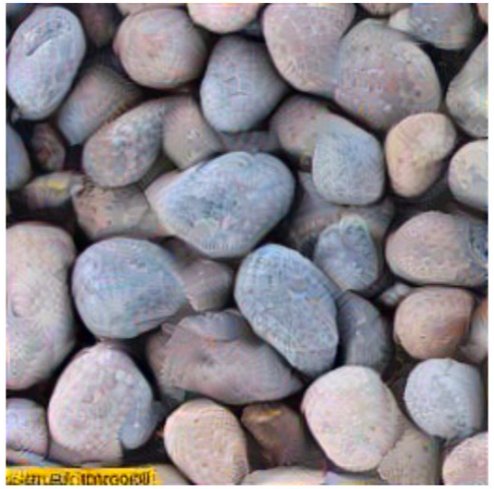
- Optimize s. t. the Gram matrix of the output gets closer to that of the exemplar



Texture synthesis using convolutional neural networks [Gatys NIPS15]

Texture synthesis based on CNN

pool4

original

Texture synthesis using convolutional neural networks [Gatys NIPS15]

Deep Learning intro for CG people

- Deep learning: a crash course (by Andrew Glassner)
  - SIGGRAPH 2018/2019 Courses
    - https://dl.acm.org/doi/abs/10.1145/3305366.3328026
  - https://www.youtube.com/watch?v=r0Ogt-q956I