UNIVERSITAT DE BARCELONA

>>> IMAGE PROCESSING AND COMPUTATIONAL PHOTOGRAPHY

SESSION 7: SYNTHESIZE

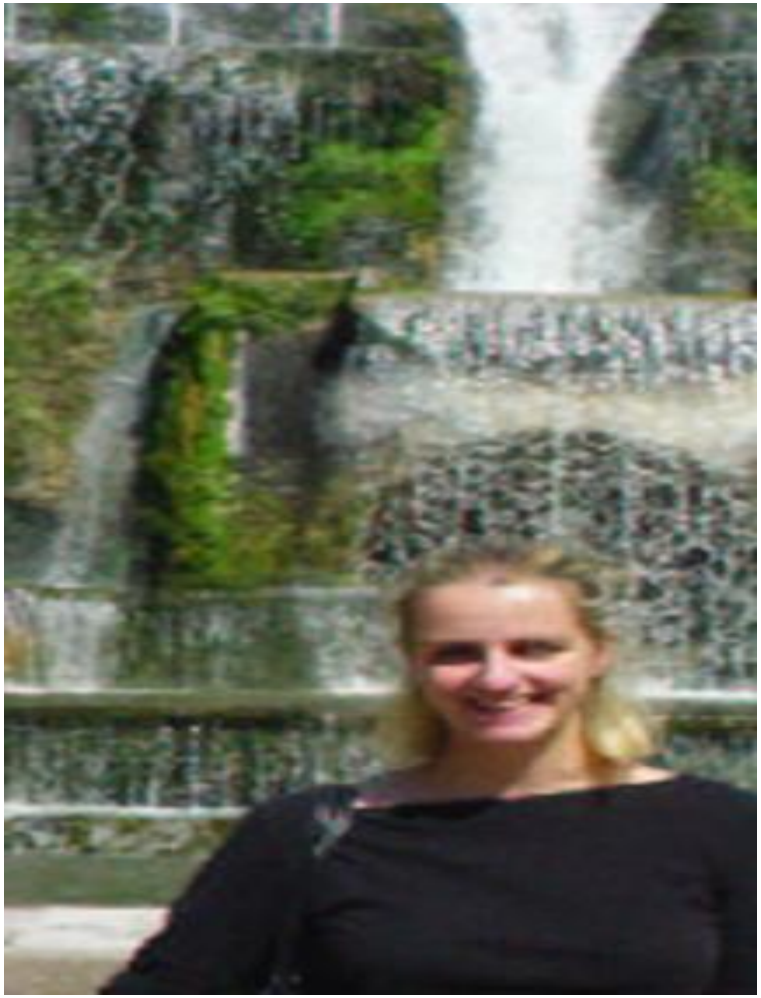
Oriol Pujol & Simone Balocco

Last class: finding boundaries

- Intelligent scissors
  - Good boundary has a low-cost path from seed to cursor
  - Low cost = edge, high gradient, right orientation

- GrabCut
  - Good region is similar to foreground color model and dissimilar from background color
  - Good boundaries have a high gradient
  - Optimize over both

Last class: Cut someone out to make more of them



But what if we want less of somebody?

Today's Class

- Texture synthesis and hole-filling

Texture

- Texture depicts spatially repeating patterns
- Textures appear naturally and frequently


radishes


rocks

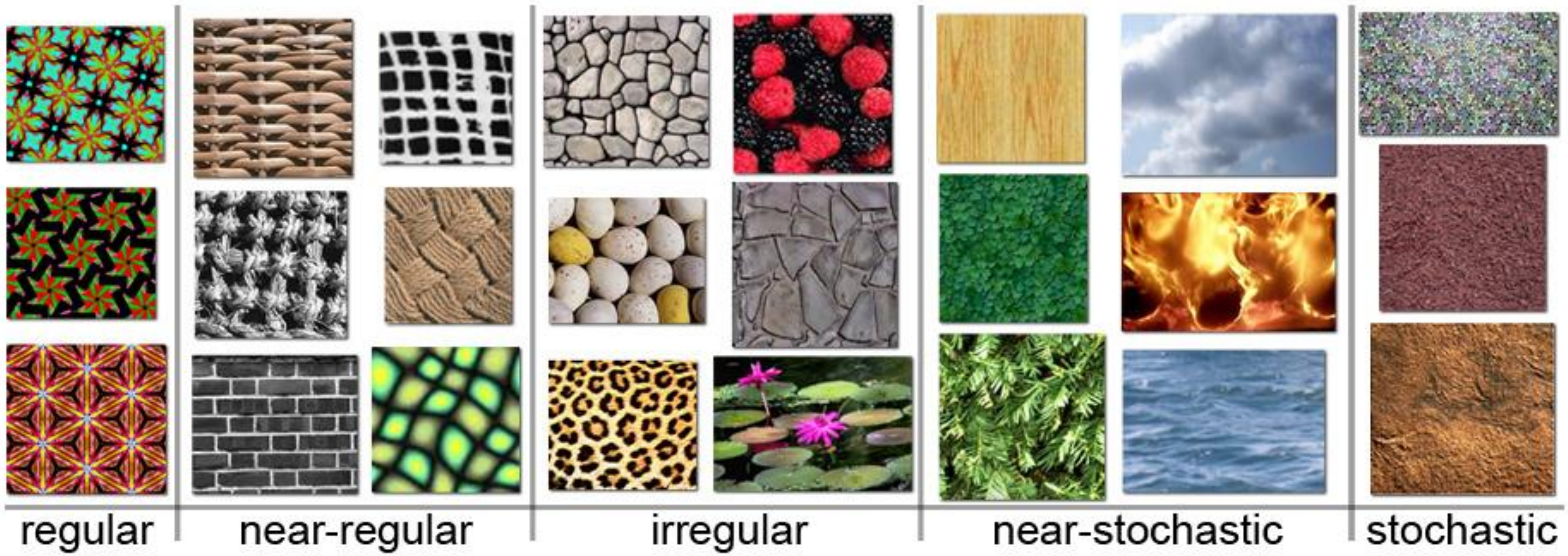

yogurt

Many slides from James Hays

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

One idea: Build Probability Distributions

Basic idea

1. Compute statistics of input texture (e.g., histogram of edge filter responses)
2. Generate a new texture that keeps those same statistics

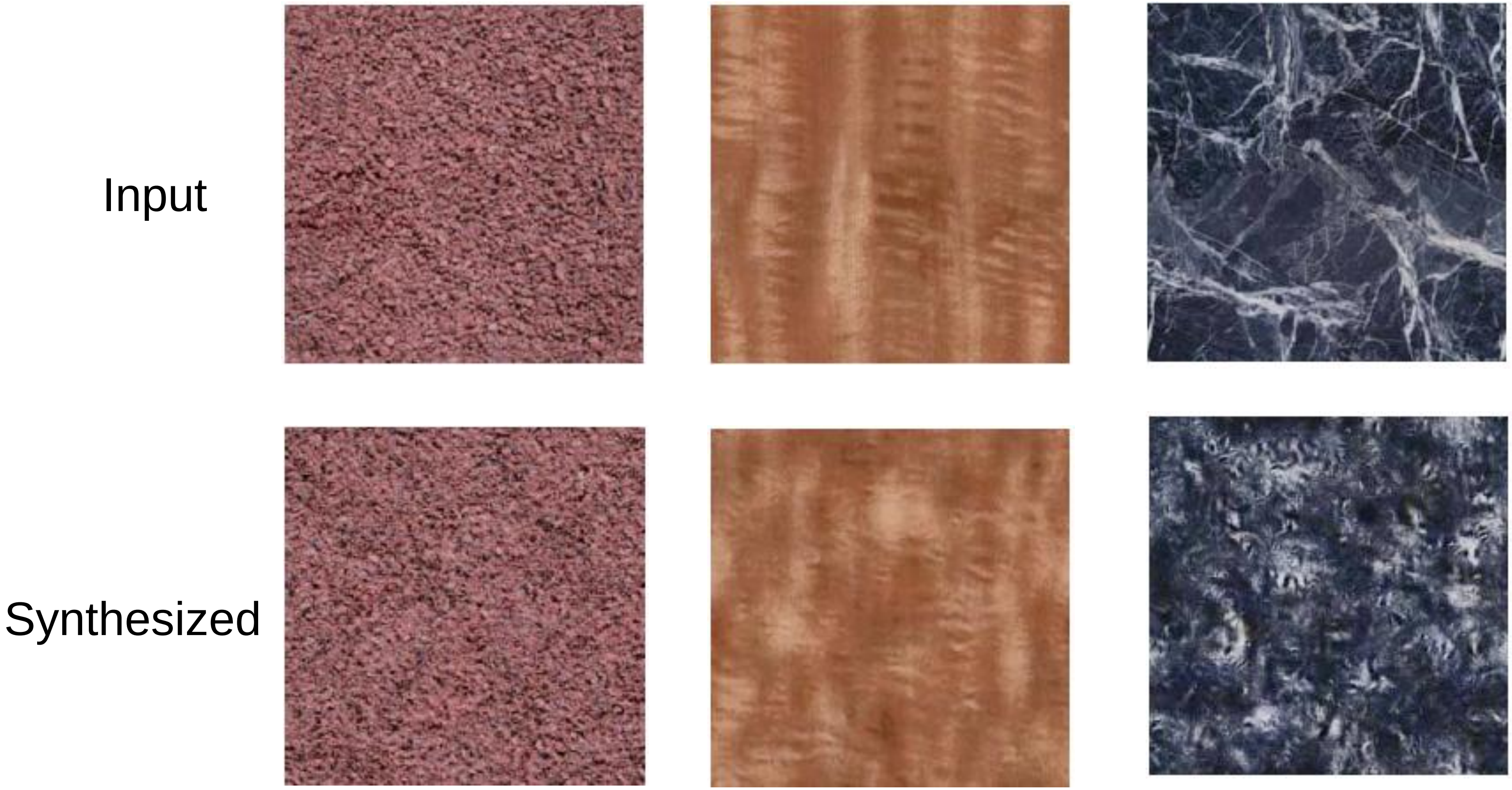


- D. J. Heeger and J. R. Bergen. Pyramid-based texture analysis/synthesis. In *SIGGRAPH '95.*

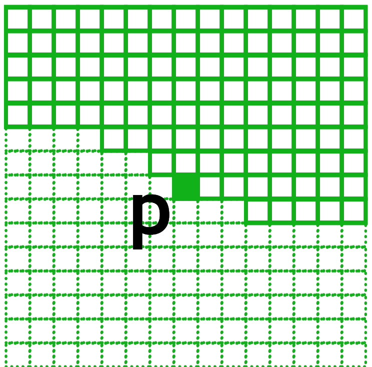
One idea: Build Probability Distributions

But it (usually) doesn't work

- Probability distributions are hard to model well

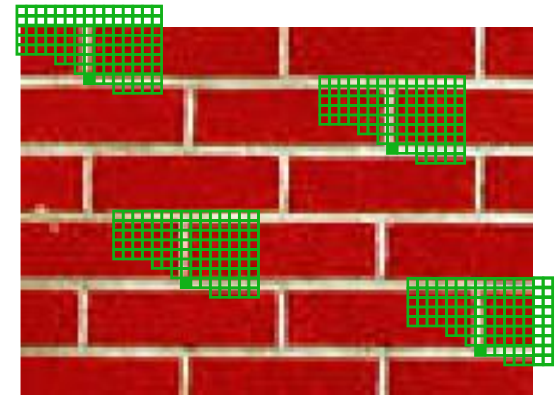


Another idea: Sample from the image



Synthesizing a pixel

non-parametric sampling



Input image

- 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

Efros and Leung 1999 SIGGRAPH

Details

- How to match patches?
  - Gaussian-weighted SSD (more emphasis on nearby pixels)
- What order to fill in new pixels?
  - “Onion skin” order: pixels with most neighbors are synthesized first
  - To synthesize from scratch, start with a randomly selected small patch from the source texture
- How big should the patches be?

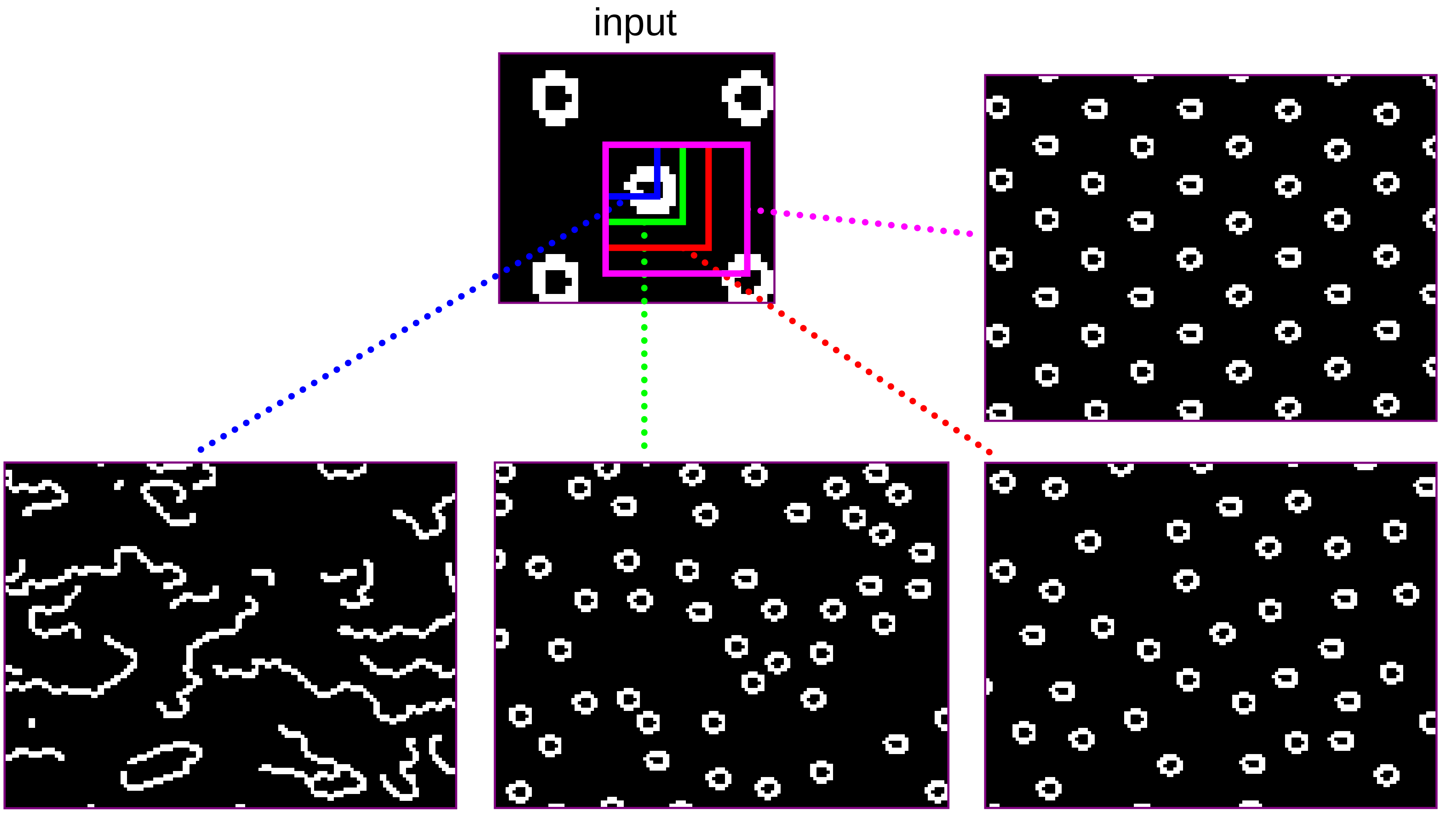
Idea from Shannon (Information Theory)

- Generate English-sounding sentences by modeling the probability of each word given the previous words (n-grams)

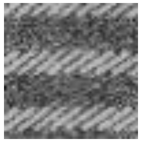
- Large "n" will give more structured sentences

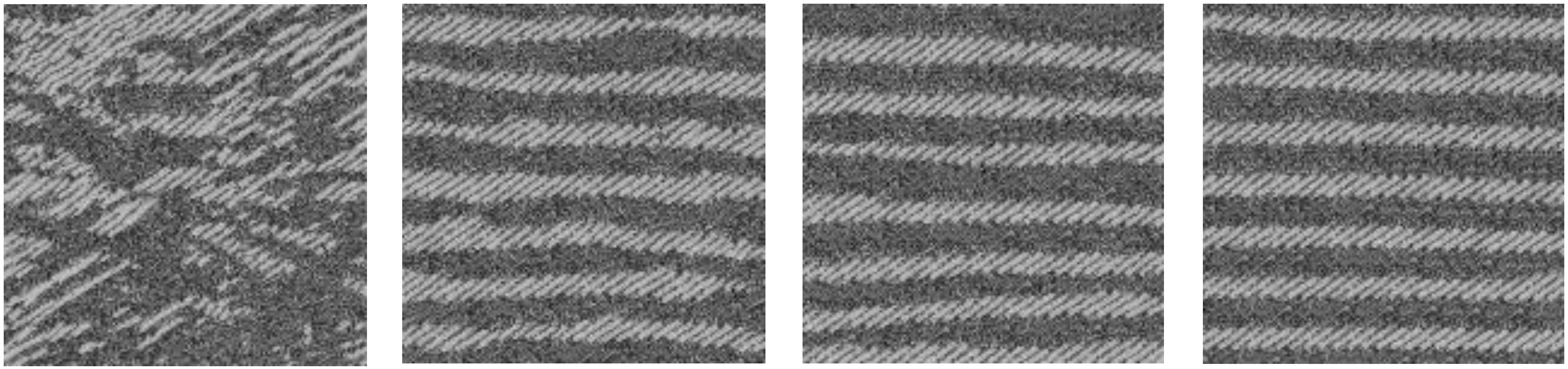
"I spent an interesting evening recently with a grain of salt."

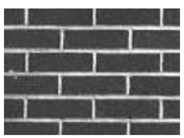
Size of Neighborhood Window

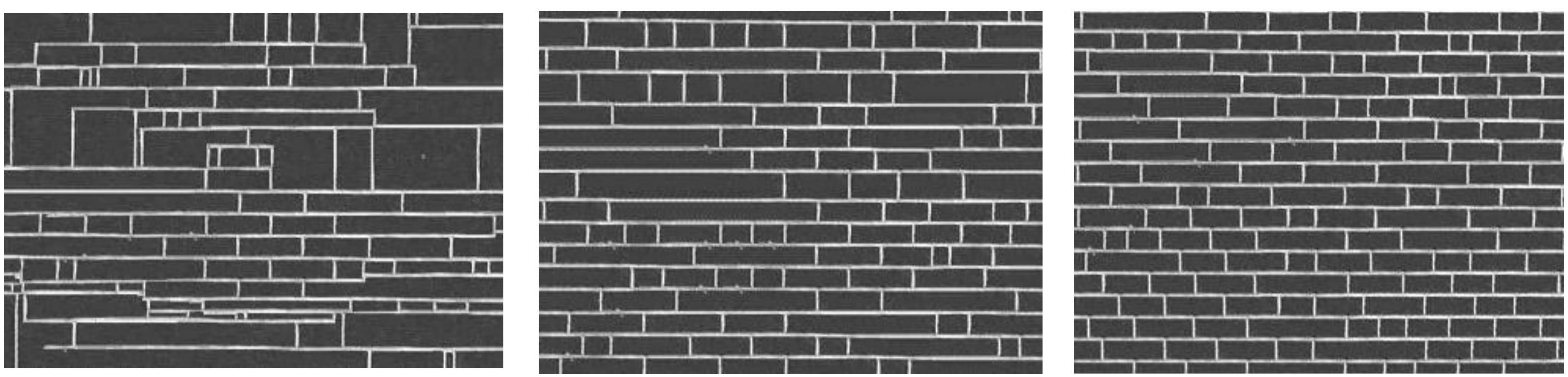


Varying Window Size









Increasing window size

Texture synthesis algorithm

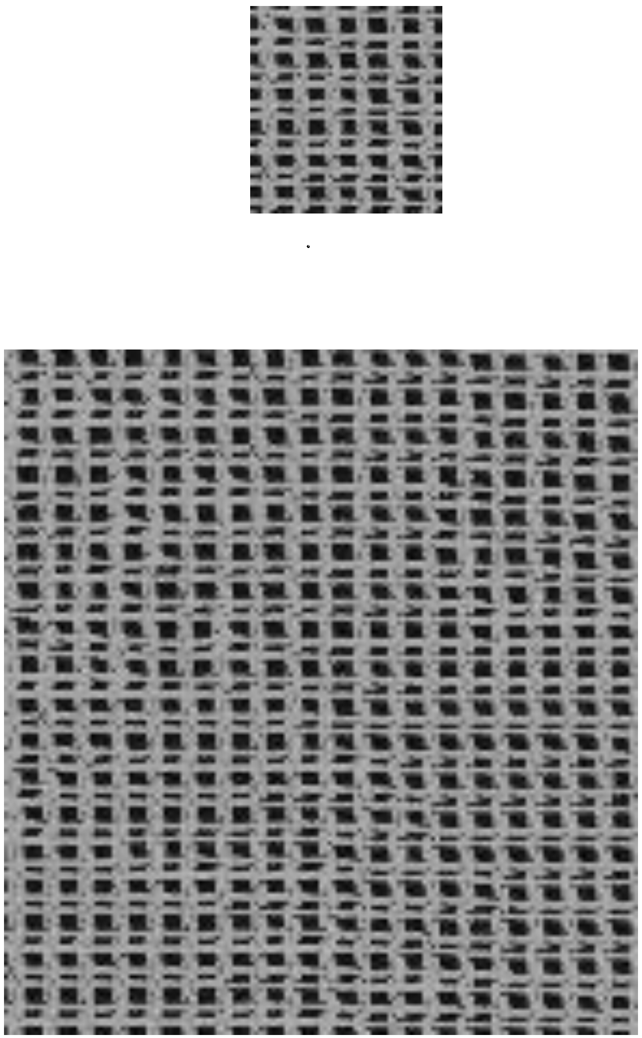
- While image not filled
  1. Get unfilled pixels with filled neighbors, sorted by number of filled neighbors

  1. For each pixel, get top N matches based on visible neighbors
      2. - Patch Distance: Gaussian-weighted SSD

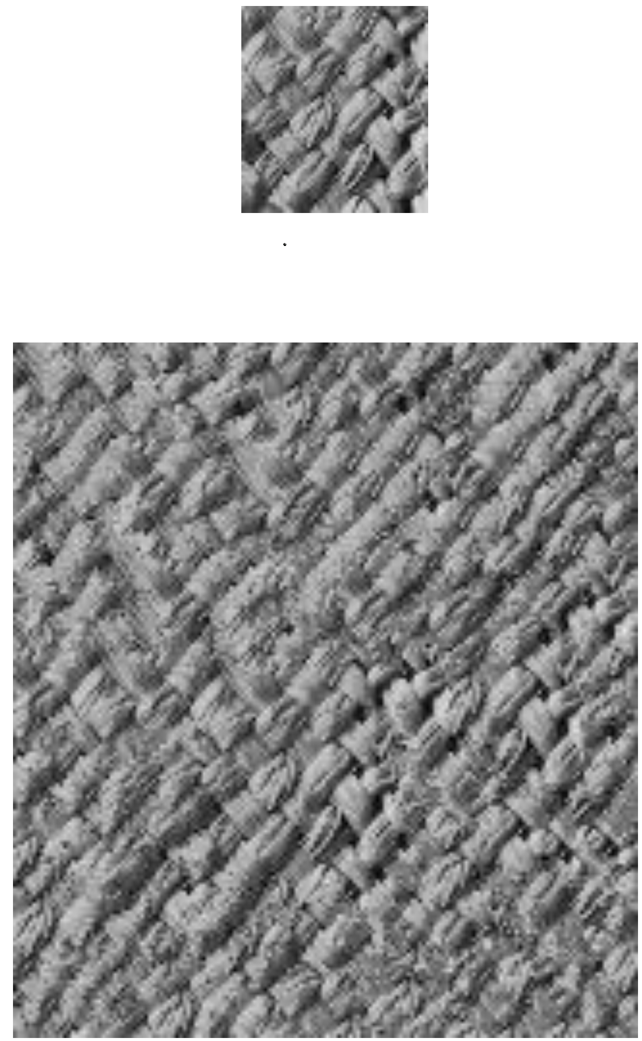
  1. Randomly select one of the matches and copy pixel from it

Synthesis Results

french canvas



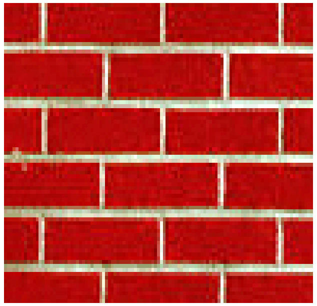
rafia weave

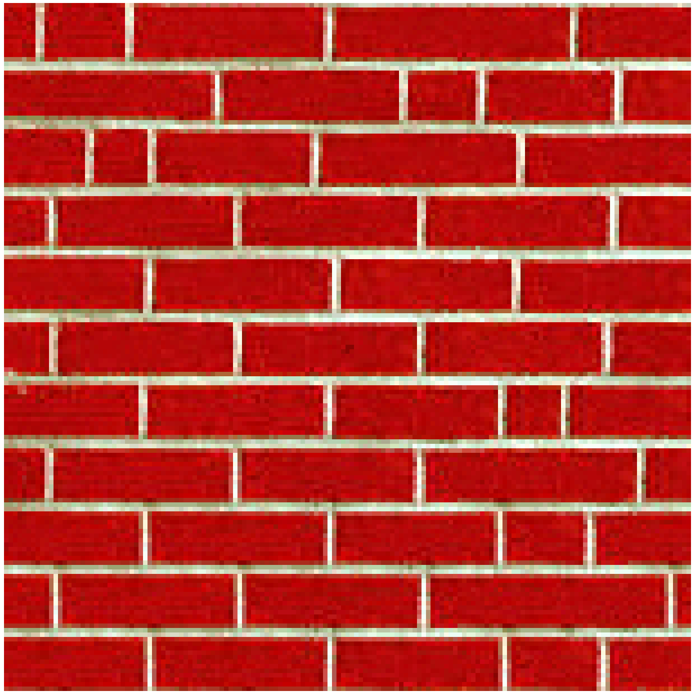


More Results

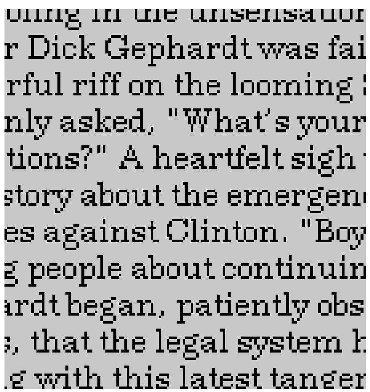
white bread

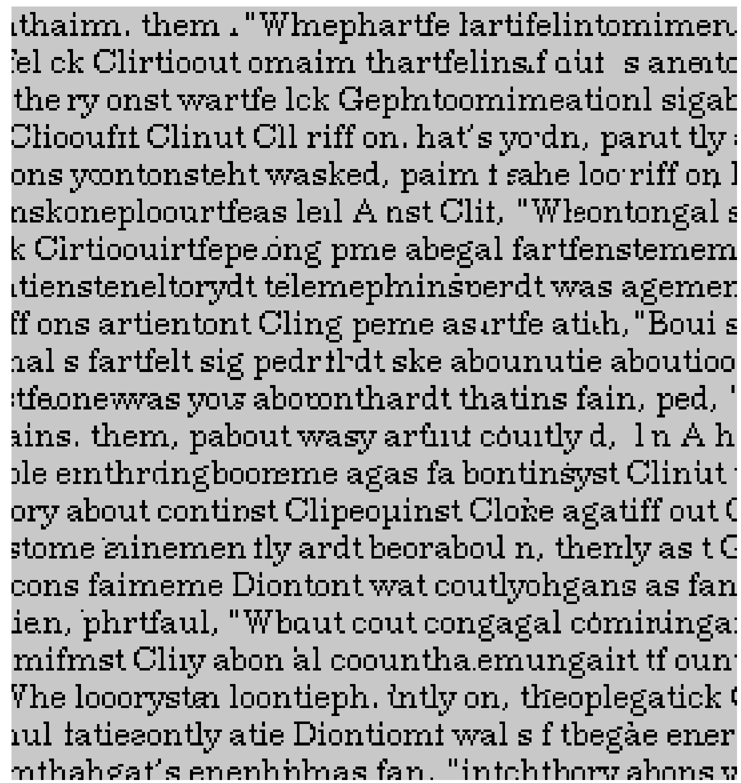
brick wall



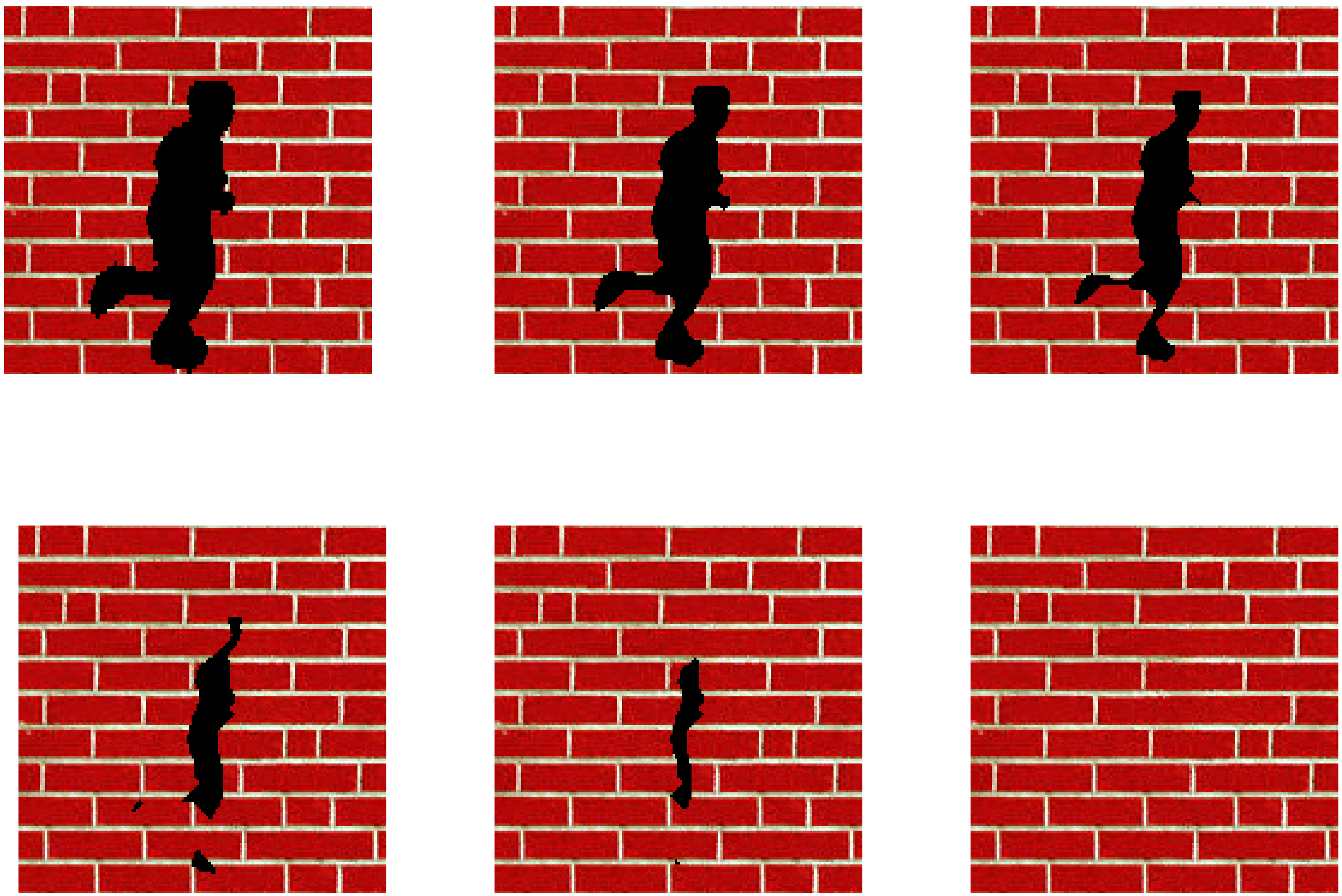


Homage to Shannon

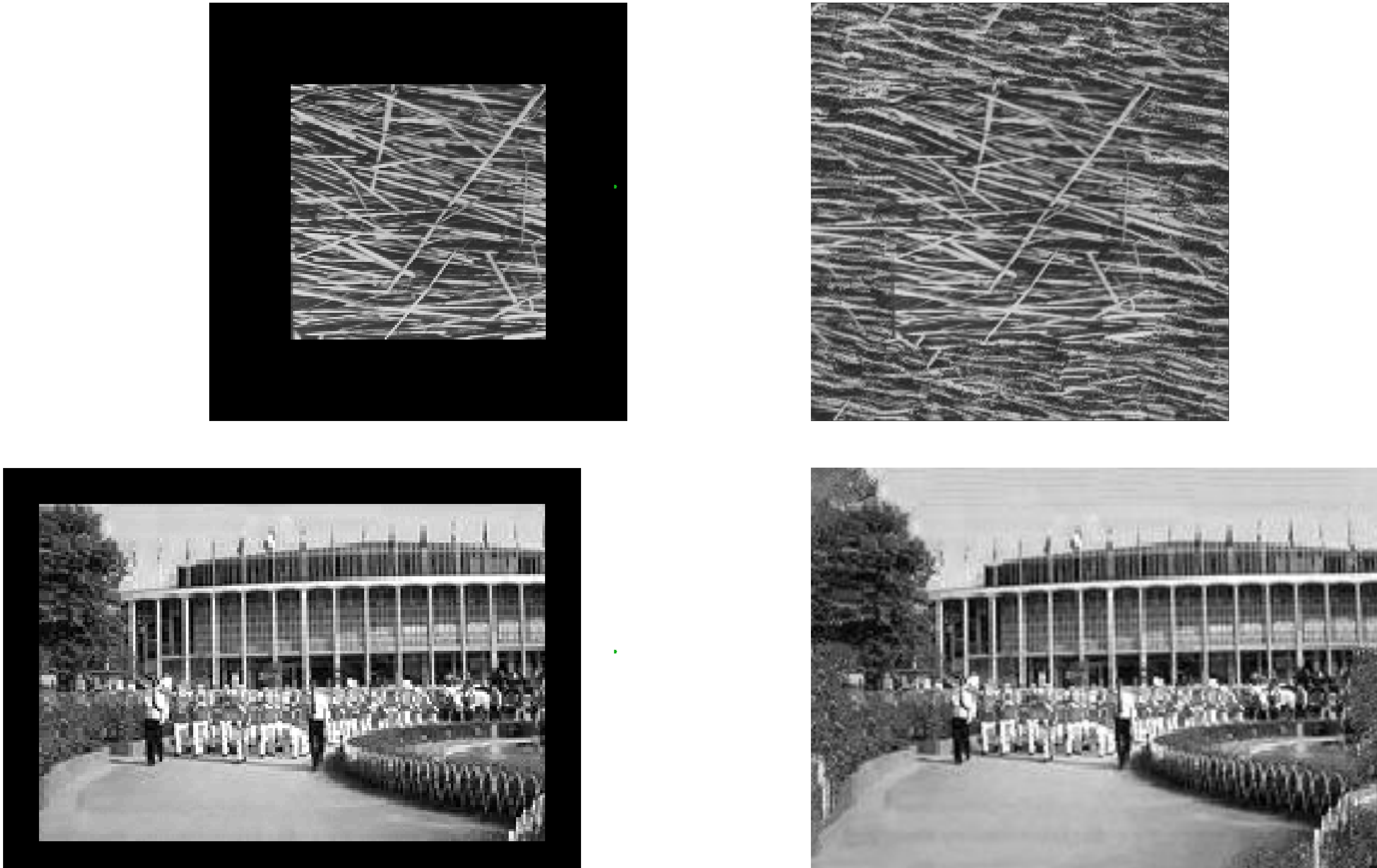




Hole Filling



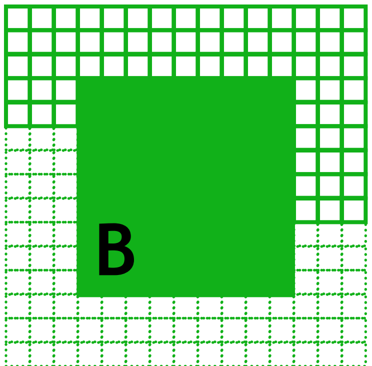
Extrapolation



Summary

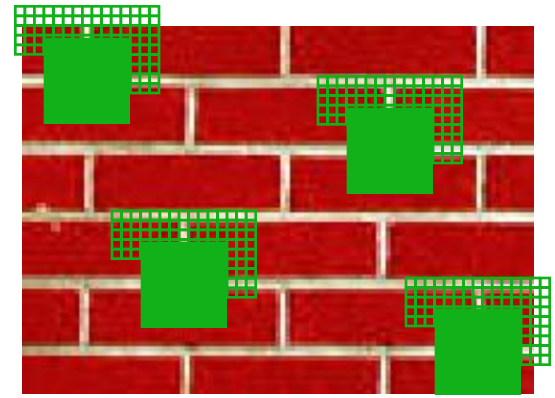
- The Efros & Leung texture synthesis algorithm
  - Very simple
  - Surprisingly good results
  - Synthesis is easier than analysis!
  - …but very slow

Image Quilting [Efros & Freeman 2001]



Synthesizing a block

non-parametric
sampling

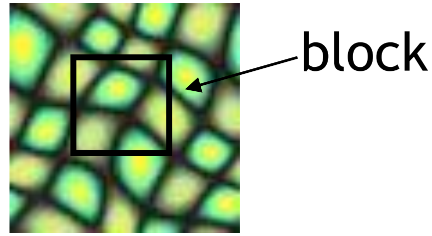


Input image

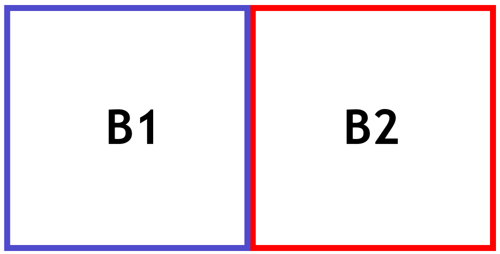
- <u>Observation:</u> neighbor pixels are highly correlated

<u>Idea:</u> unit of synthesis = block

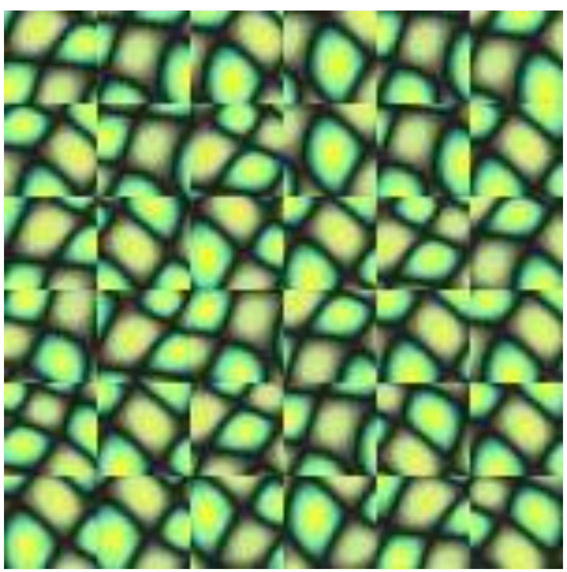
- Exactly the same but now we want P(**B**|N(**B**))
- Much faster: synthesize all pixels in a block at once

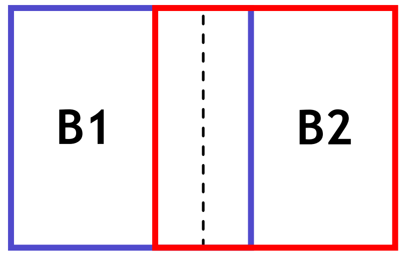


Input texture

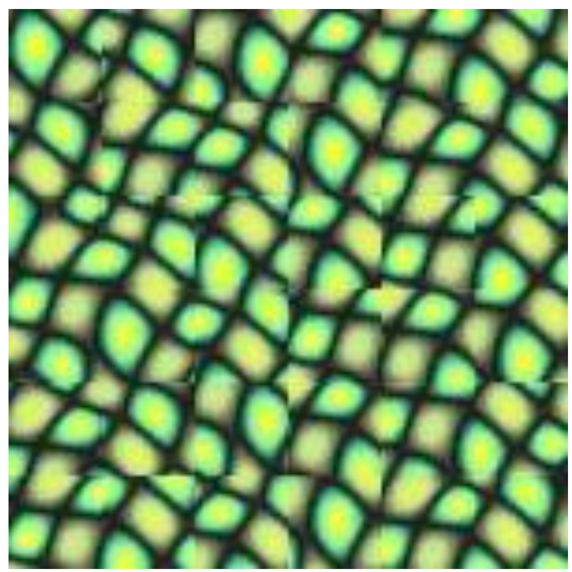


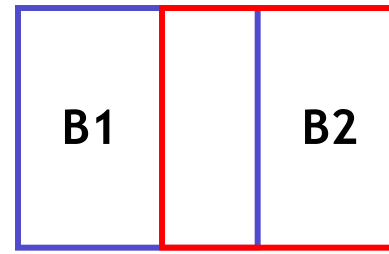
Random placement of blocks



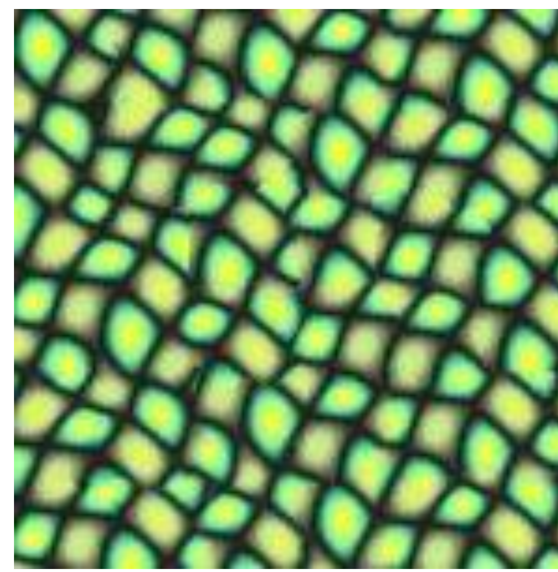


Neighboring blocks constrained by overlap

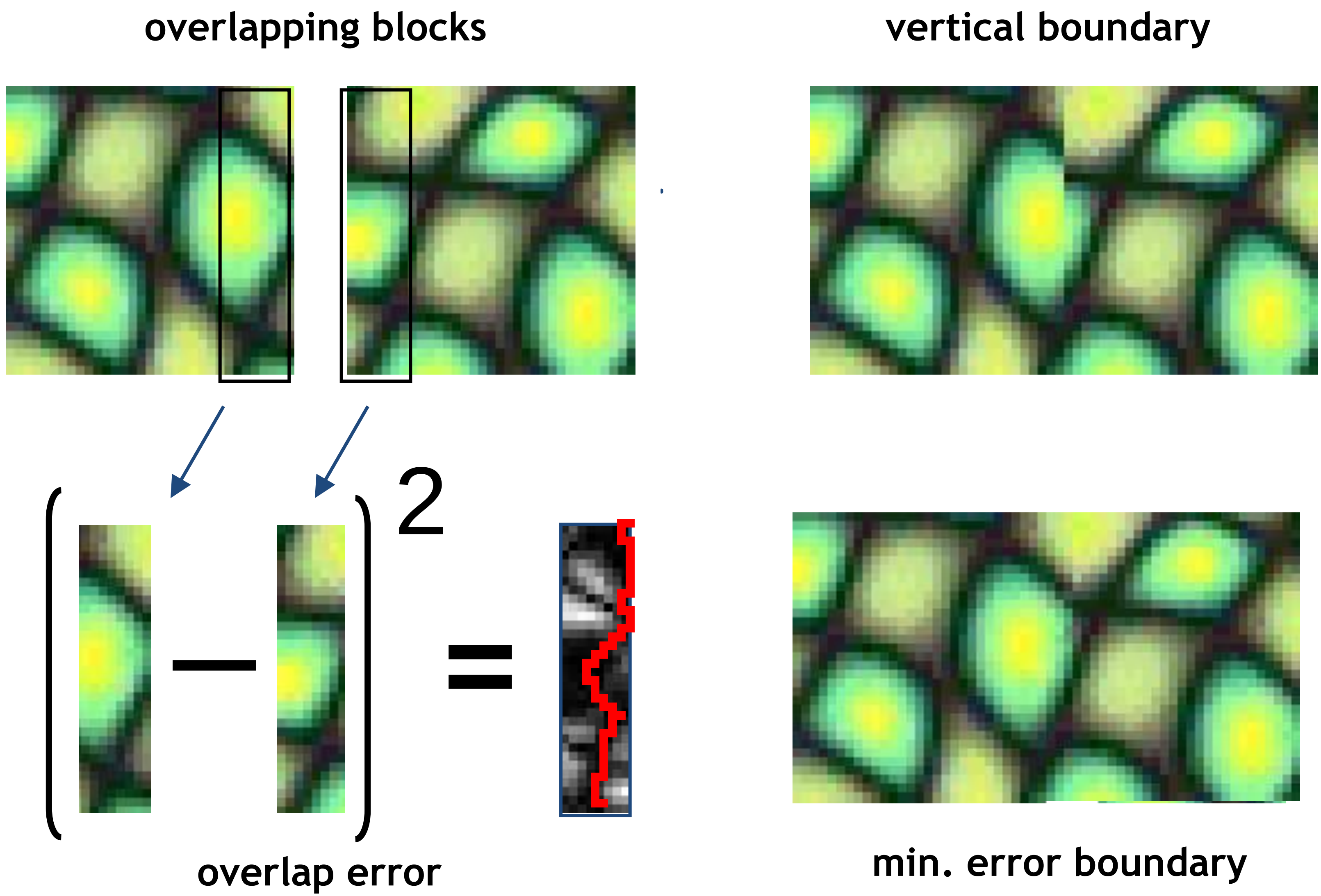


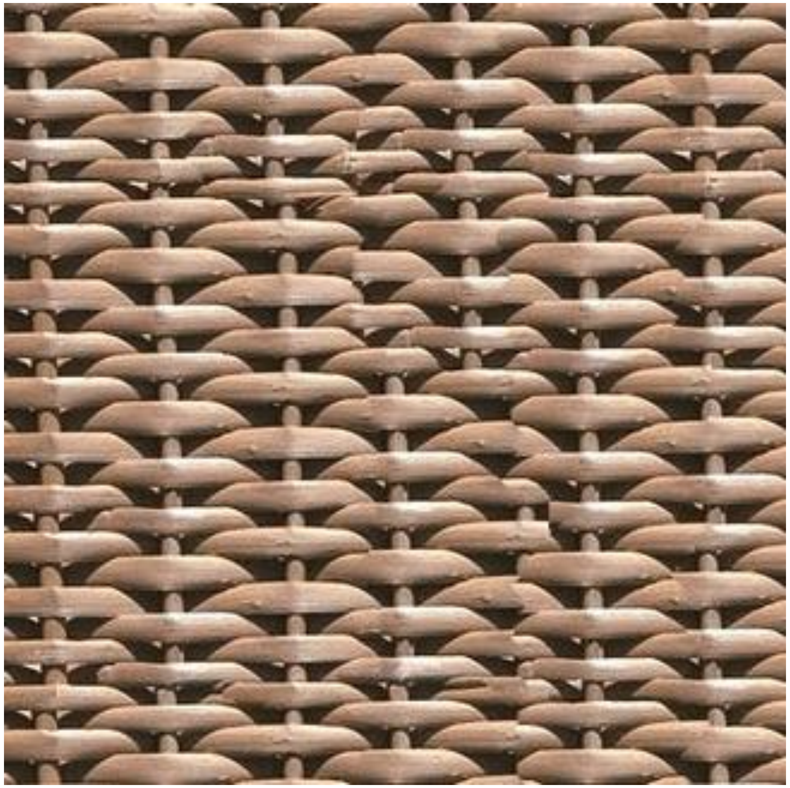


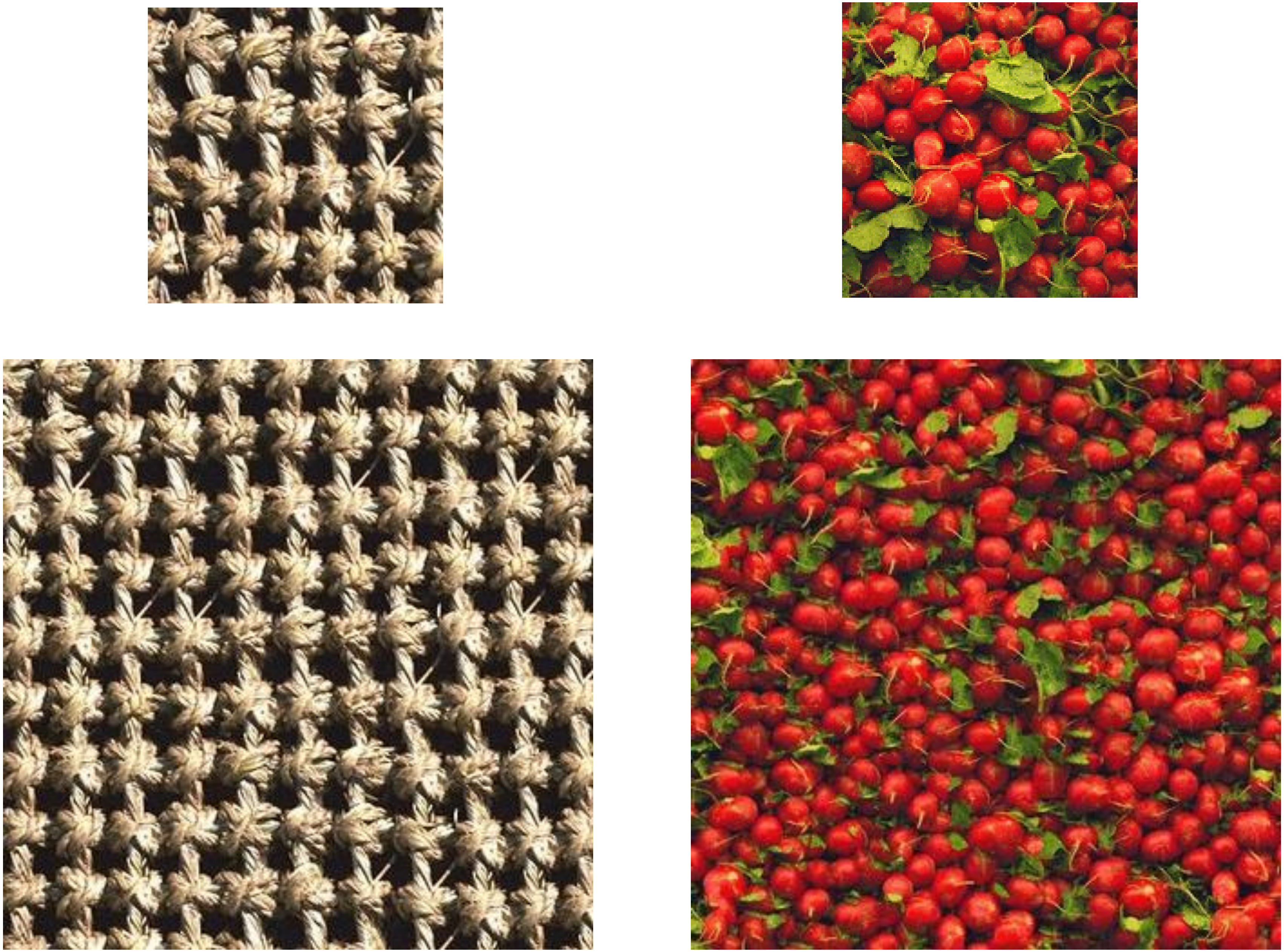
Minimal error boundary cut

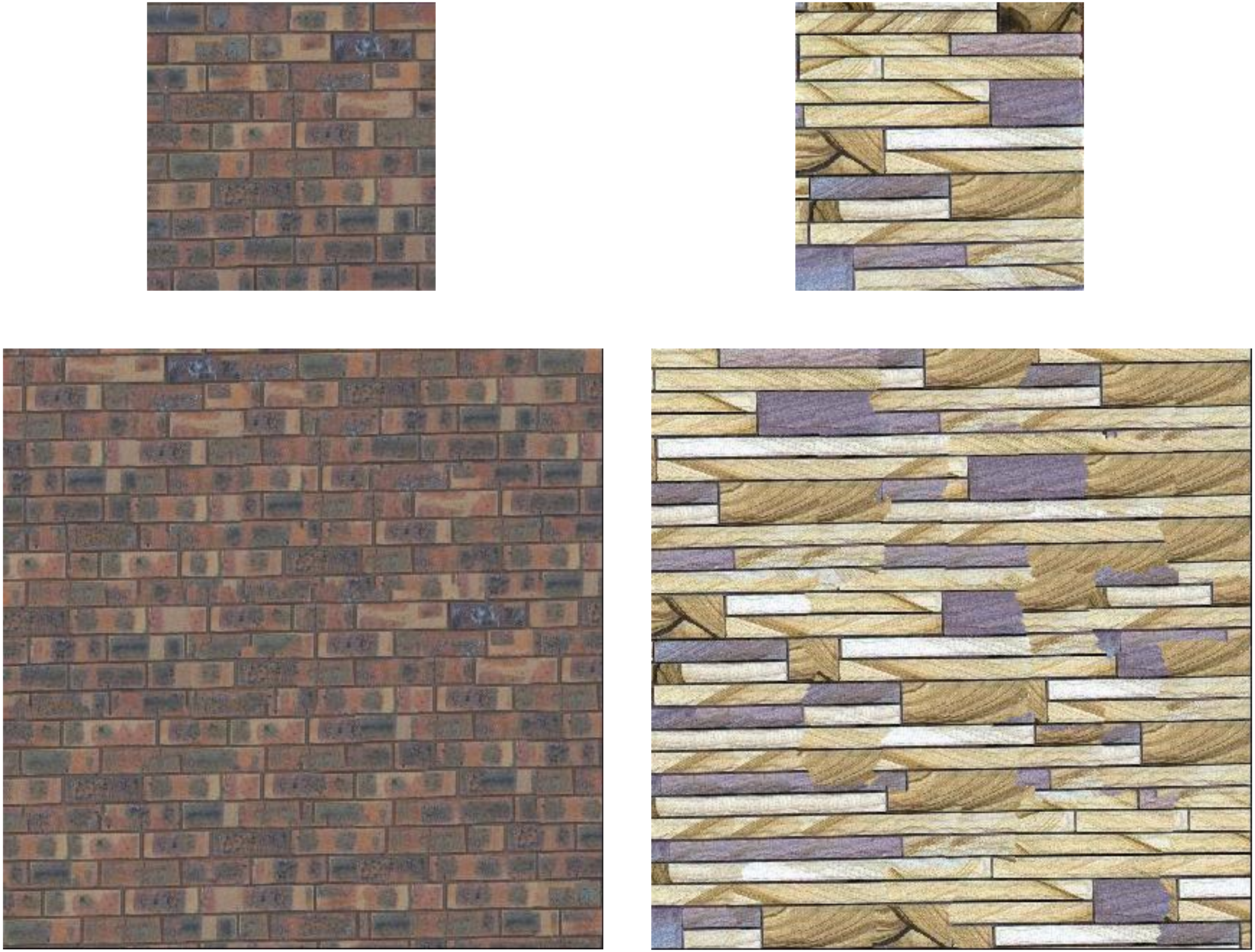


Minimal error boundary



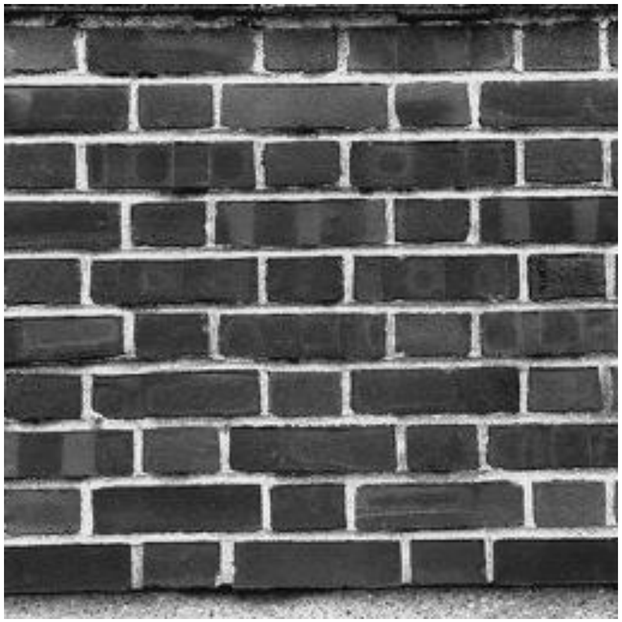






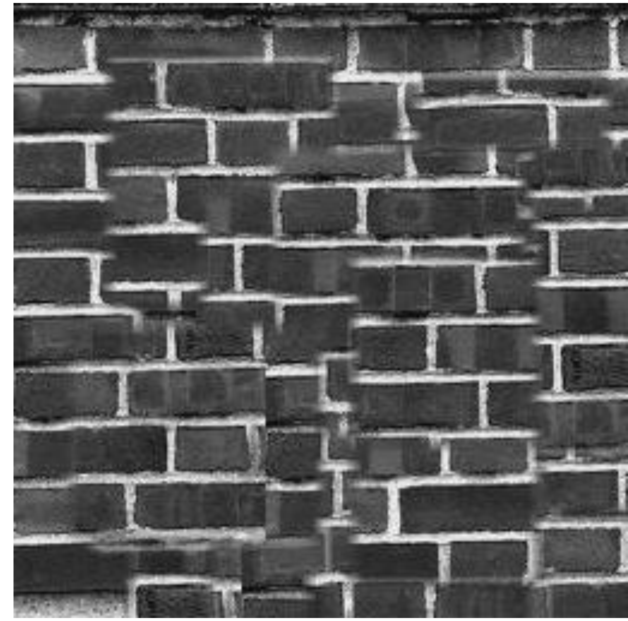

College Inn


College Inn

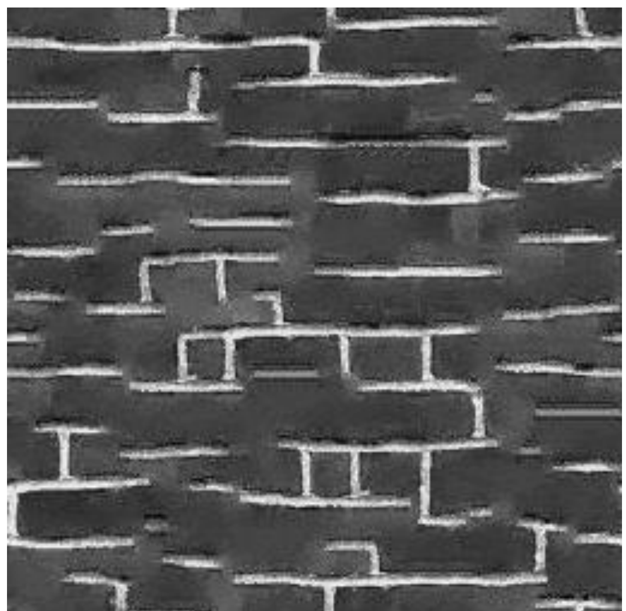


input image

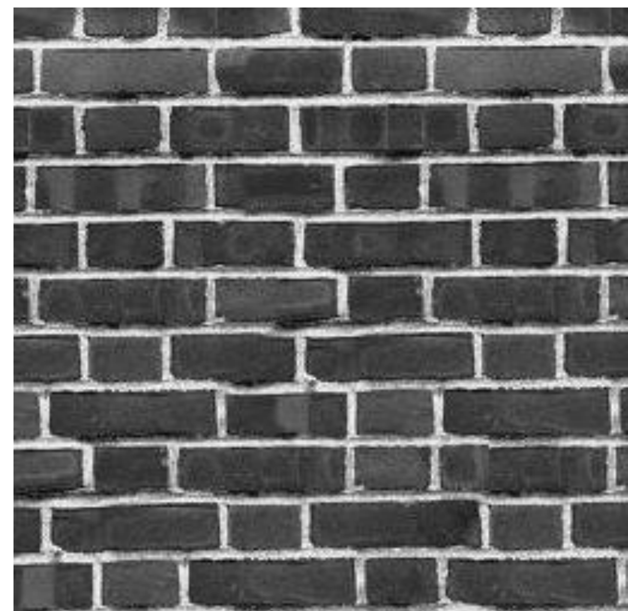
Portilla & Simoncelli



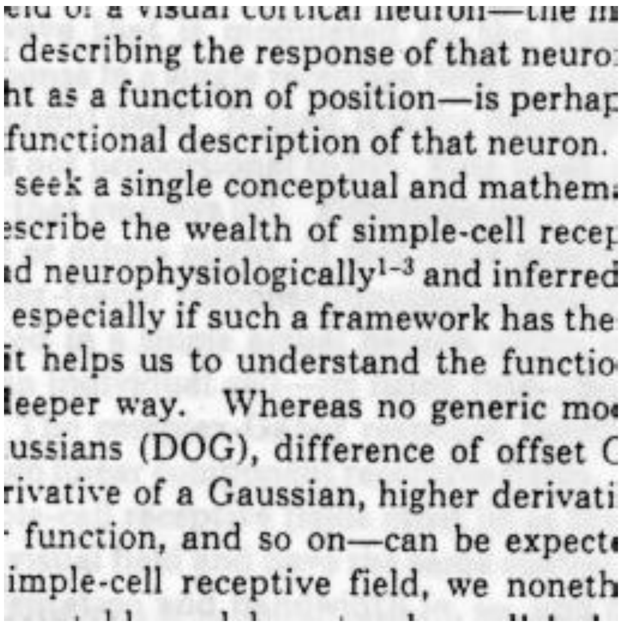
Xu, Guo & Shum



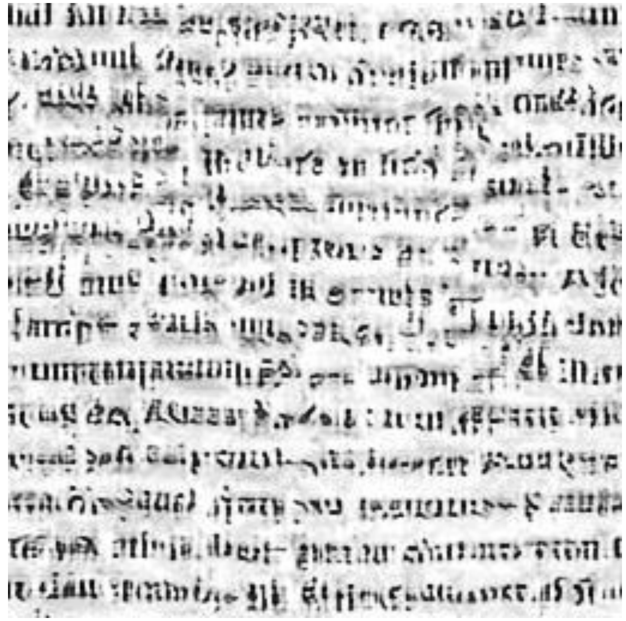
Wei & Levoy



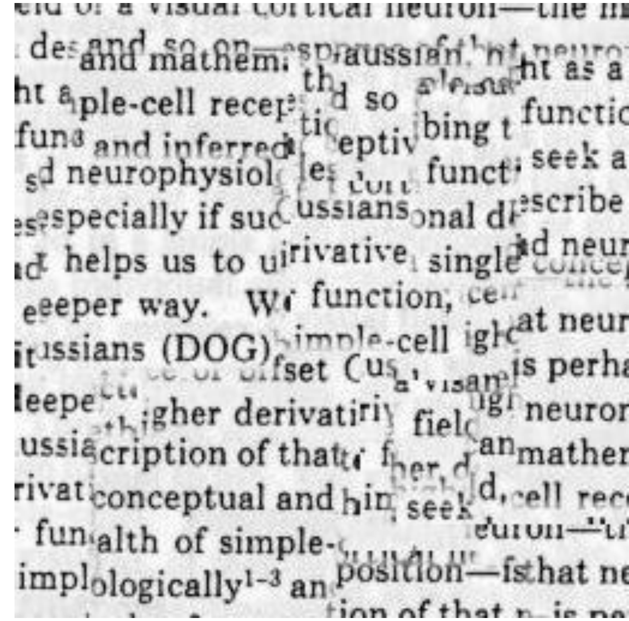
Quilting



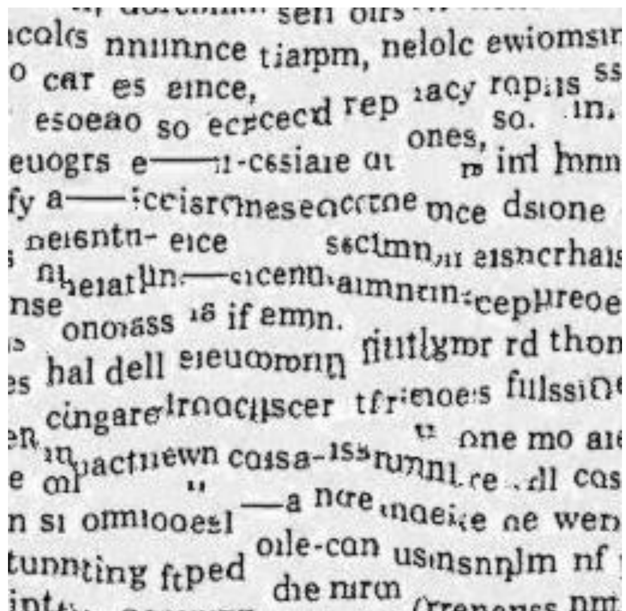
input image



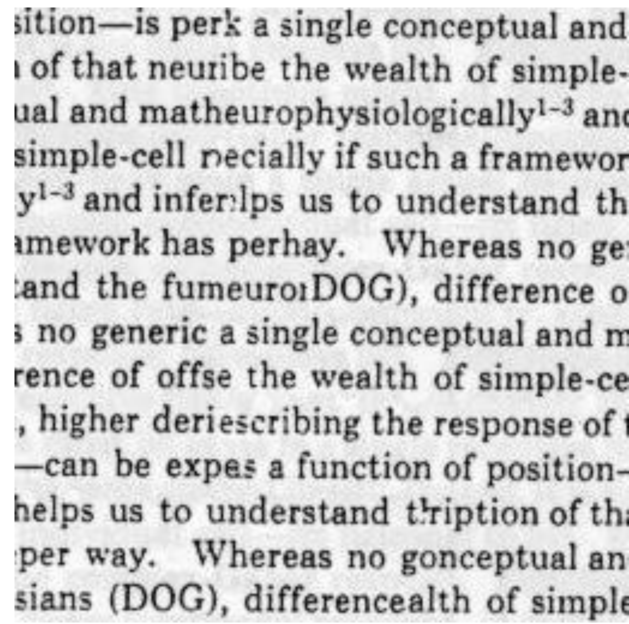
Portilla & Simoncelli



Xu, Guo & Shum

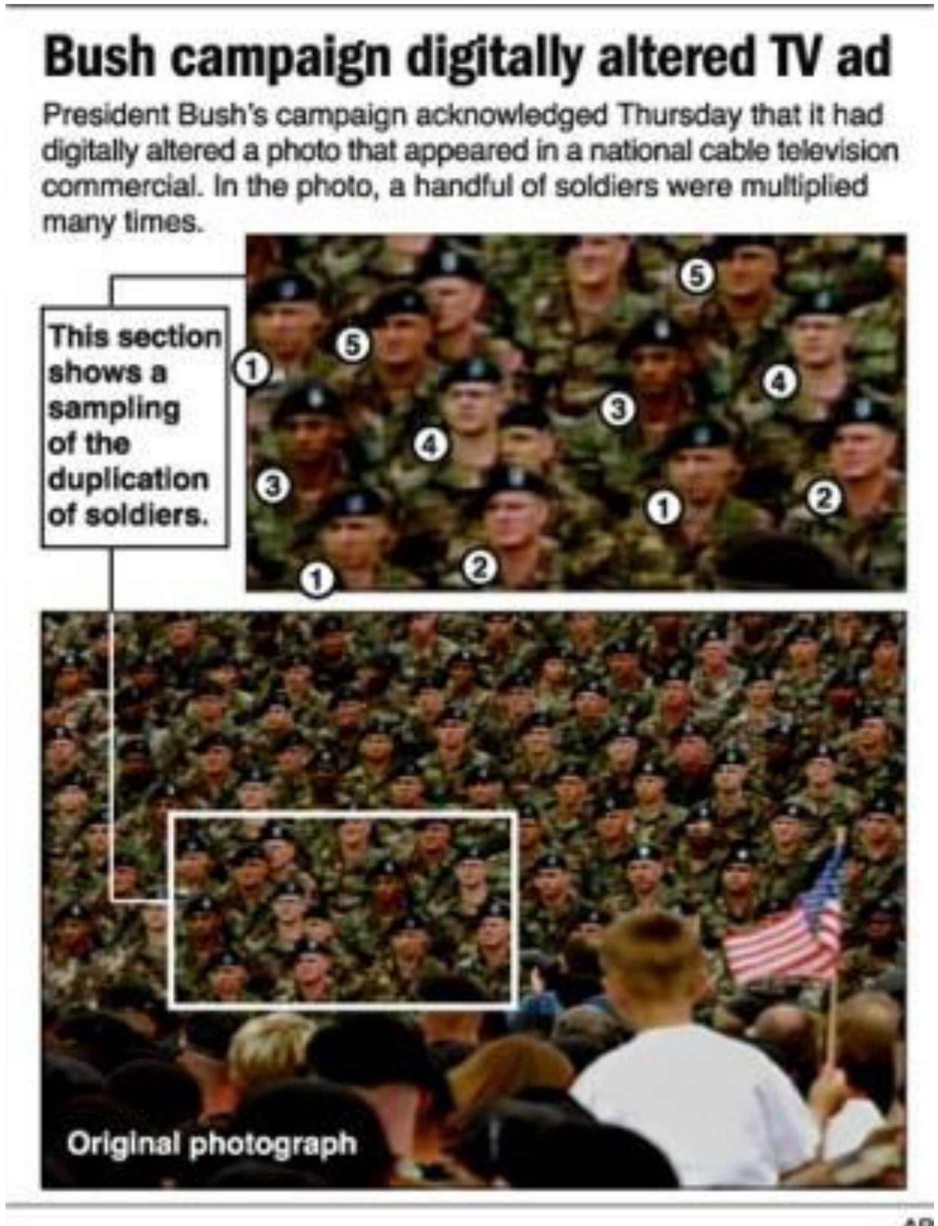


Wei & Levoy



Quilting

Political Texture Synthesis!


Bush campaign digitally altered TV ad

President Bush's campaign acknowledged Thursday that it had digitally altered a photo that appeared in a national cable television commercial. In the photo, a handful of soldiers were multiplied many times.

In-painting natural scenes

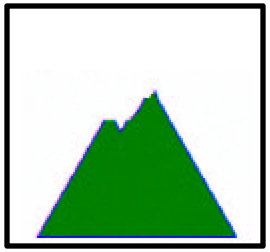
Criminisi, Perez, and Toyama. "Object Removal by Exemplar-based Inpainting," Proc. CVPR, 2003.

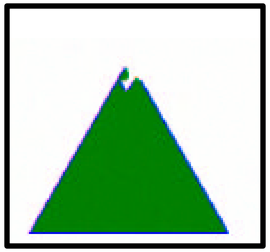
Key idea: Filling order matters

In-painting Result


Image with Hole


Raster-Scan Order


Onion-Peel
(Concentric Layers)


Gradient-Sensitive
Order

Criminisi, Perez, and Toyama. "Object Removal by Exemplar-based Inpainting," Proc. CVPR, 2003.

Filling order

Fill a pixel that:

1. Is surrounded by other known pixels
2. Is a continuation of a strong gradient or edge

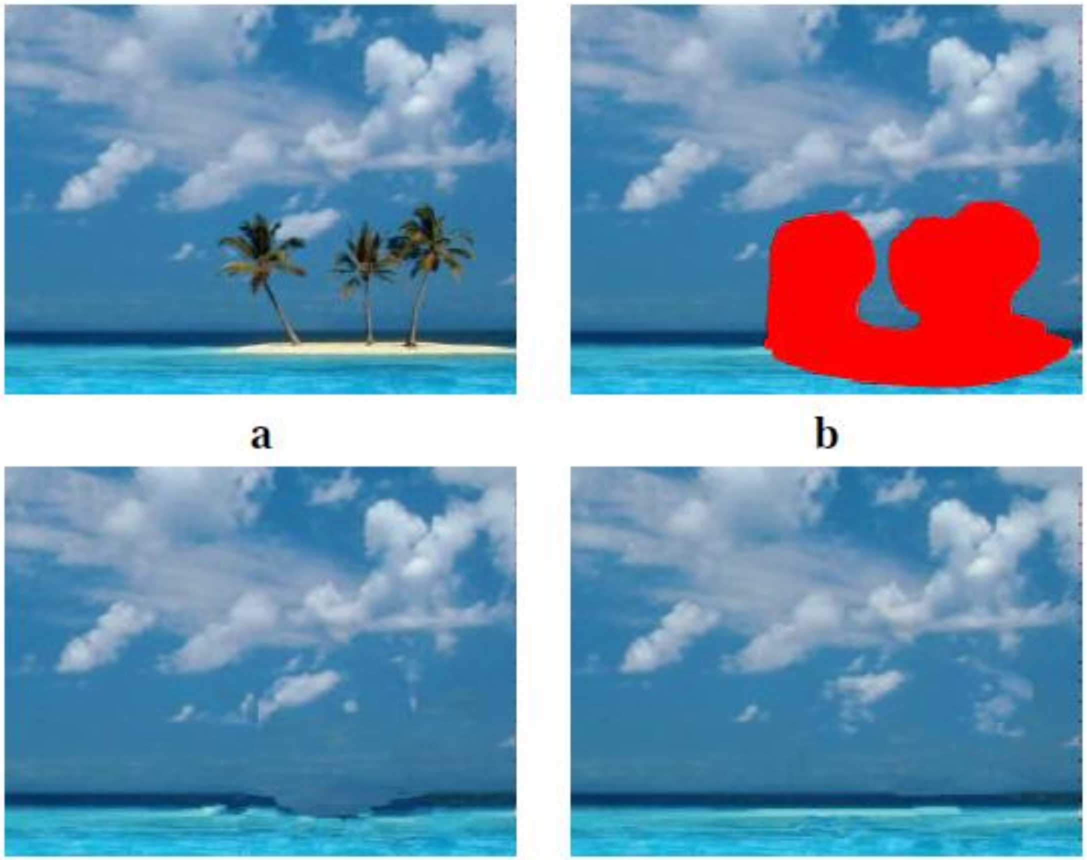
Criminisi, Perez, and Toyama. "Object Removal by Exemplar-based Inpainting," Proc. CVPR, 2003.

Comparison

Original With Hole Onion-Ring Fill Criminisi

Criminisi, Perez, and Toyama. “Object Removal by Exemplar-based Inpainting,” Proc. CVPR, 2003.

Comparison

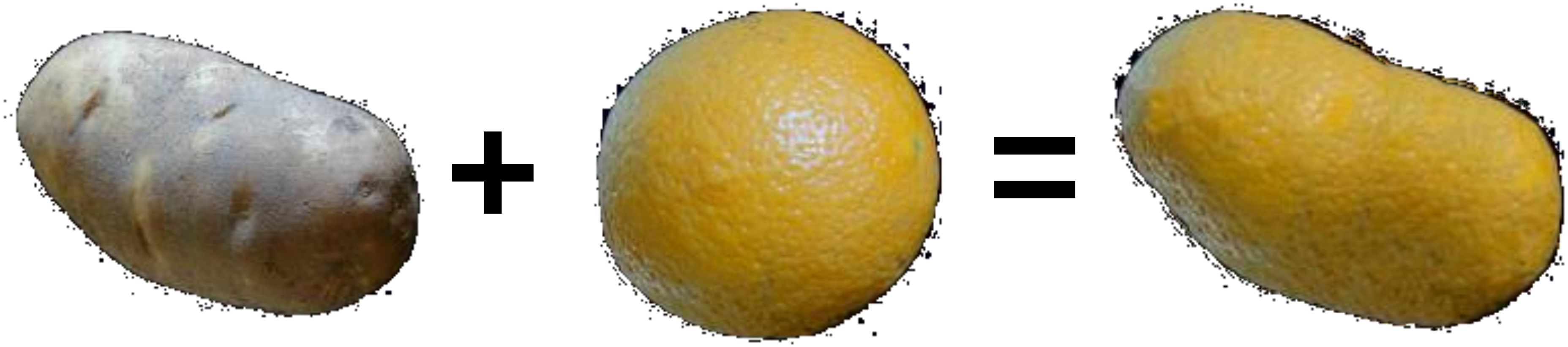


Concentric Layers

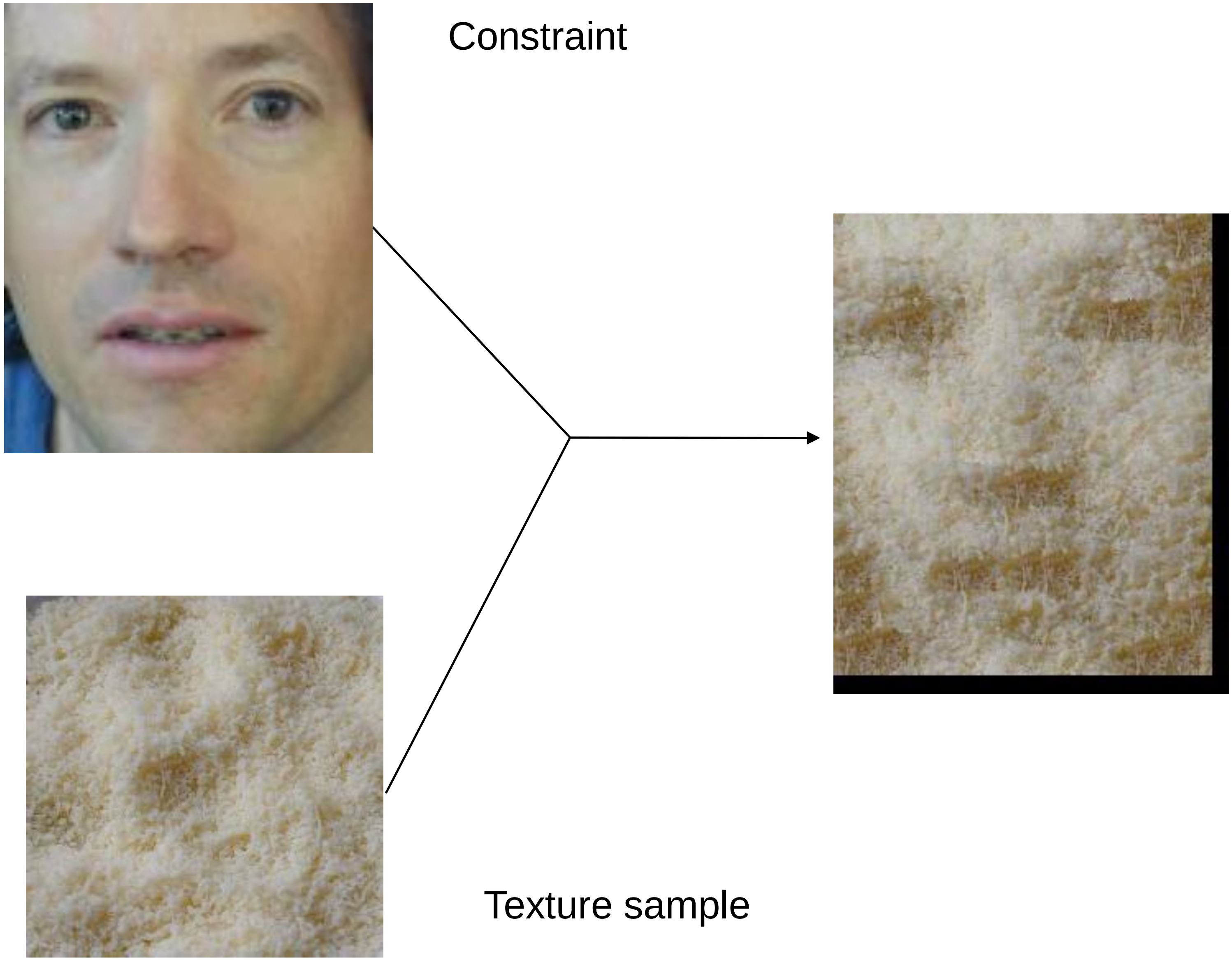
Gradient Sensitive

Texture Transfer

- Try to explain one object with bits and pieces of another object:



Texture Transfer



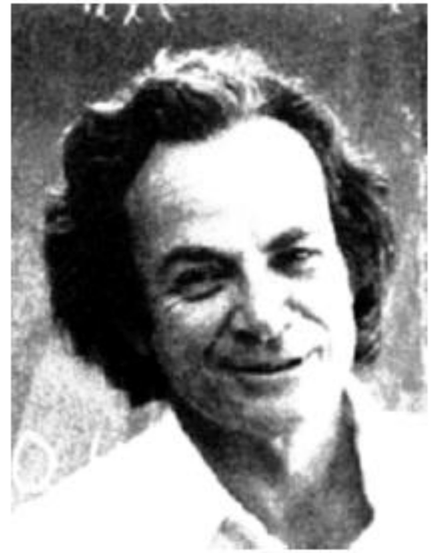
Texture Transfer

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. Patches from texture should correspond to patches from constraint in some way

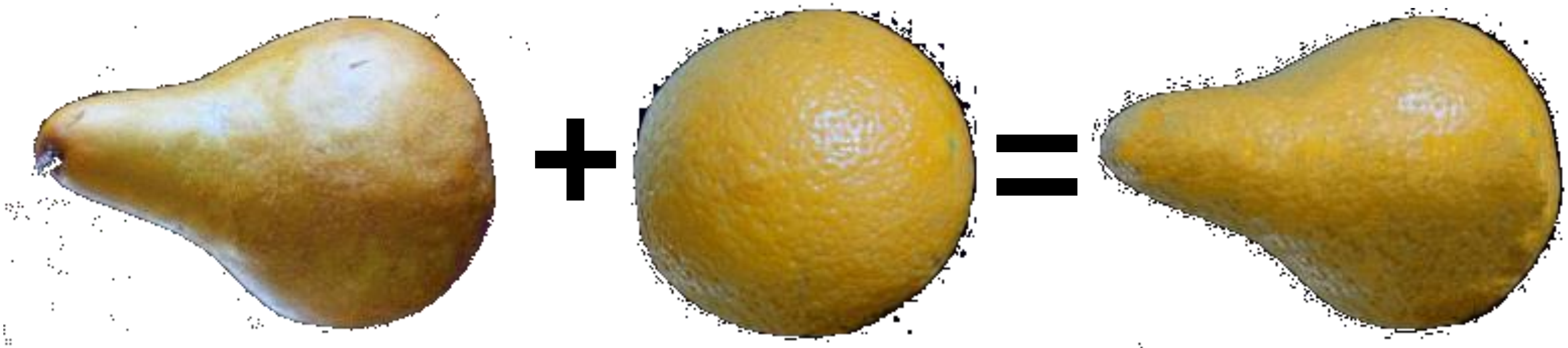
source texture



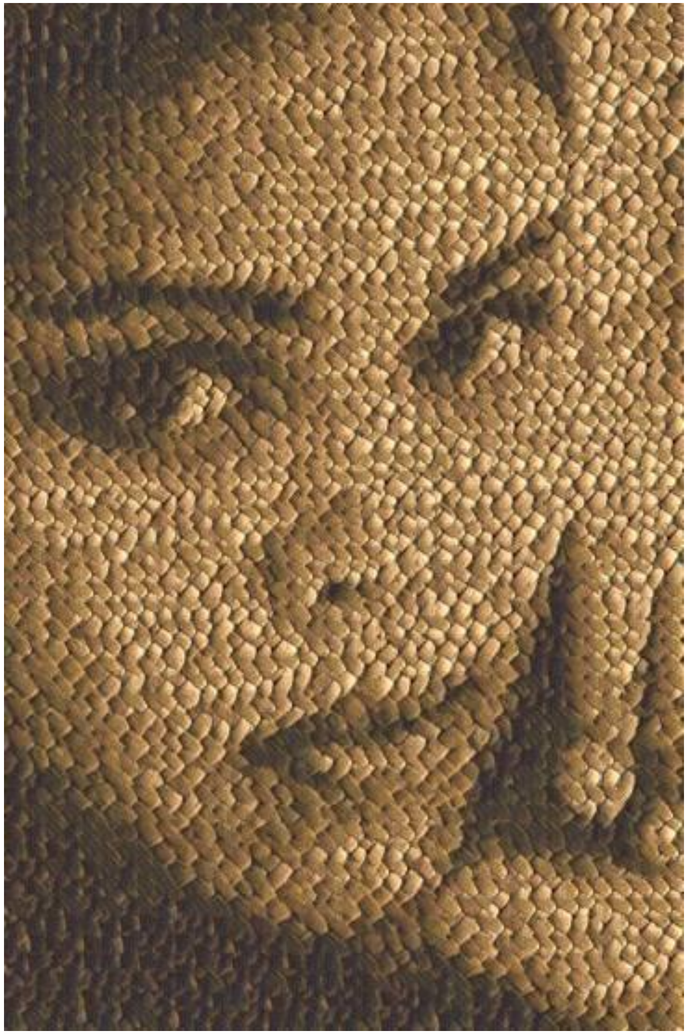
target image

correspondence maps

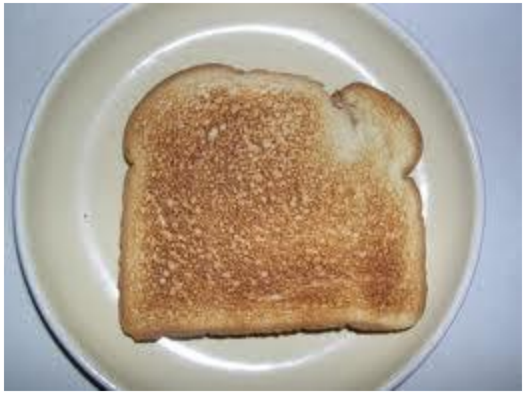
texture transfer result







Make your own sacred toast



+

?

Image analogies

- Define a similarity between A and B
- For each patch in B:
  - Find a matching patch in A, whose corresponding A' also fits in well with existing patches in B'
  - Copy the patch in A' to B'
- Algorithm is done iteratively, coarse-to-fine

Related idea: Image Analogies

Learn filter from examples

A

A'

B

B’

Image Analogies, Hertzmann et al. SG 2001

Related idea: Image Analogies

Learn filter from examples

A

A'

B

B’

Image Analogies, Hertzmann et al. SG 2001

Blur Filter

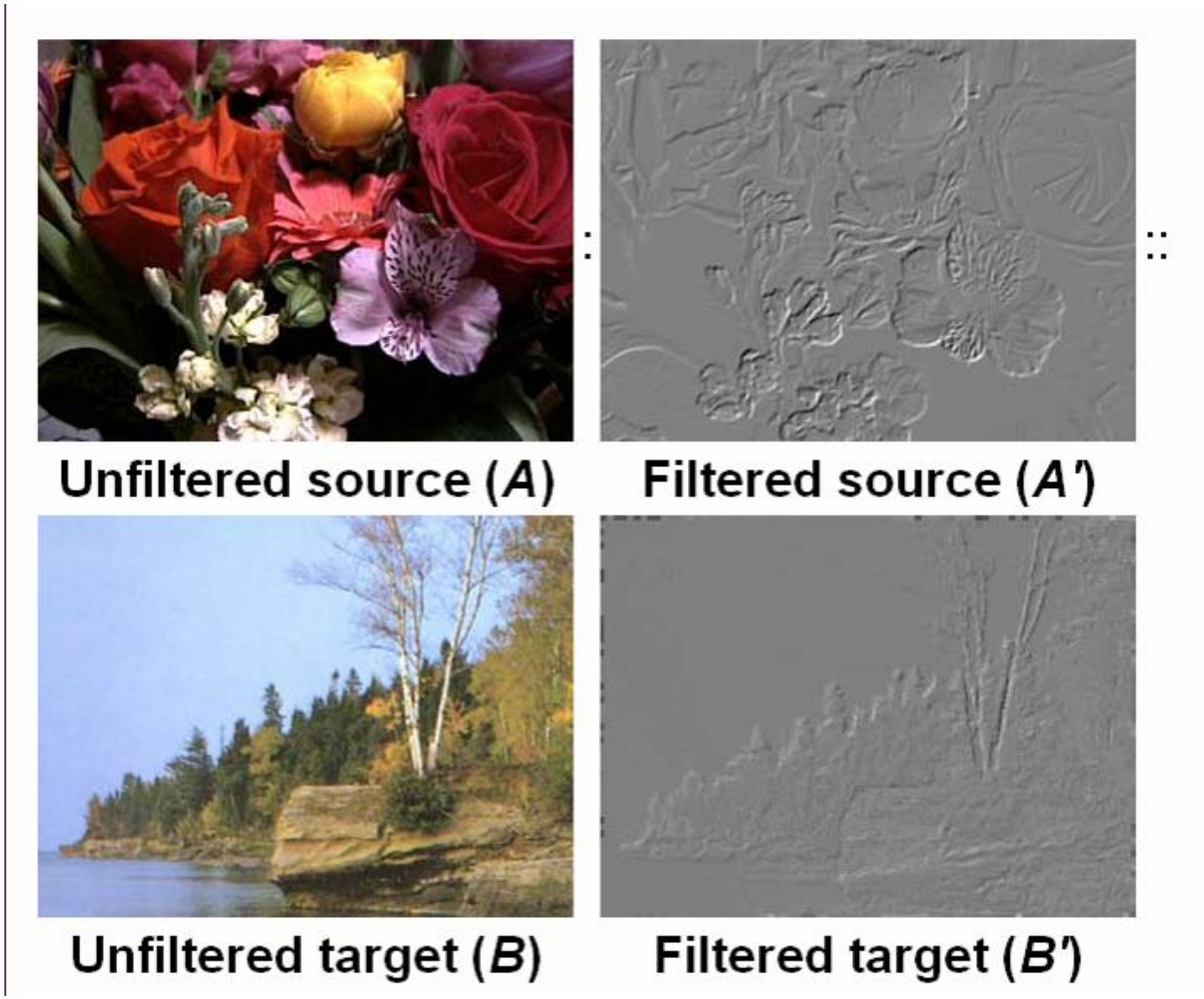
Unfiltered source (A)

Filtered source (A')

Unfiltered target (B)

Filtered target (B')

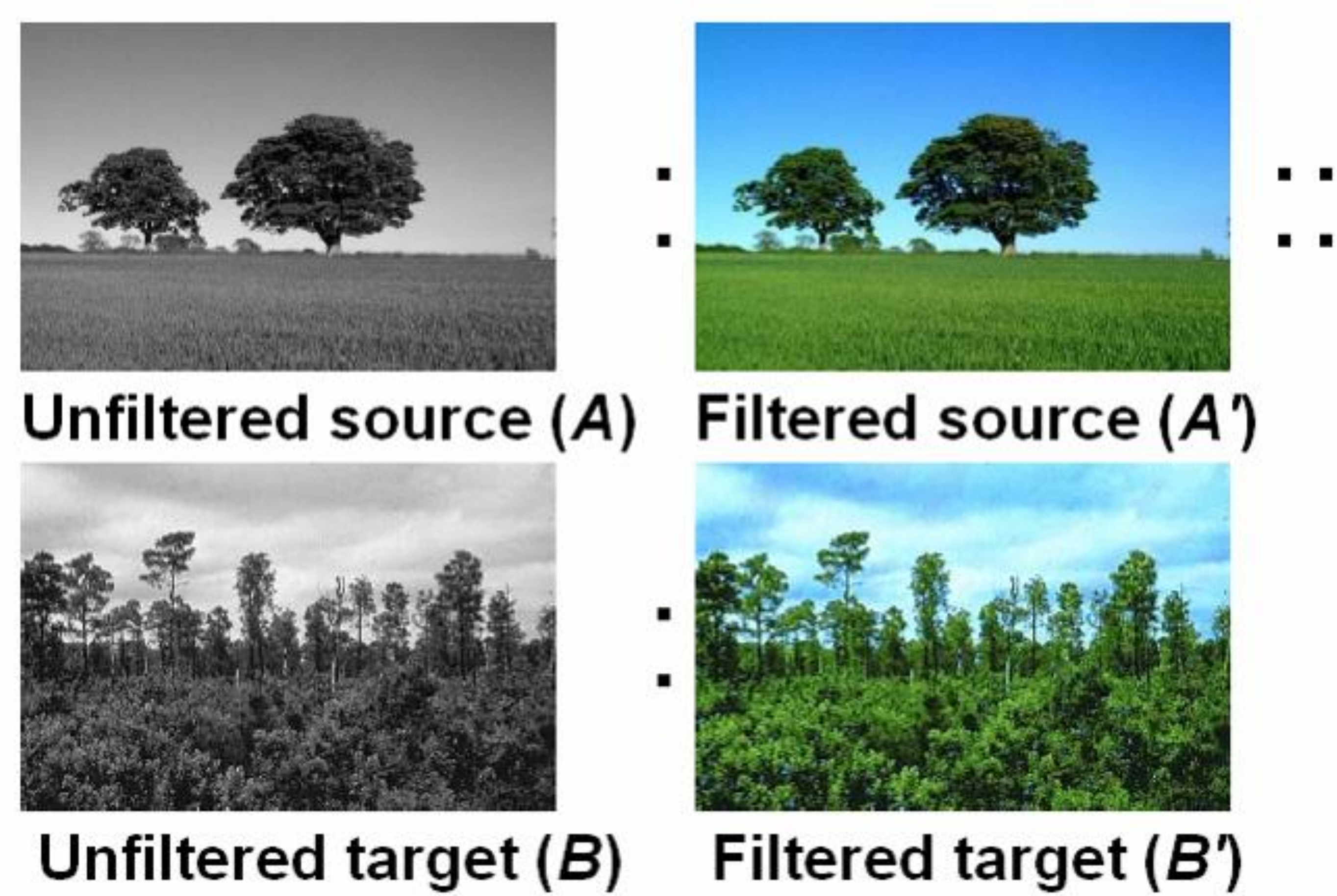
Edge Filter


Unfiltered source (A)
Filtered source (A')
Unfiltered target (B)
Filtered target (B')

Texture by numbers

https://www.youtube.com/watch?v=0F4bpf0Tesg

Colorization



Super-resolution

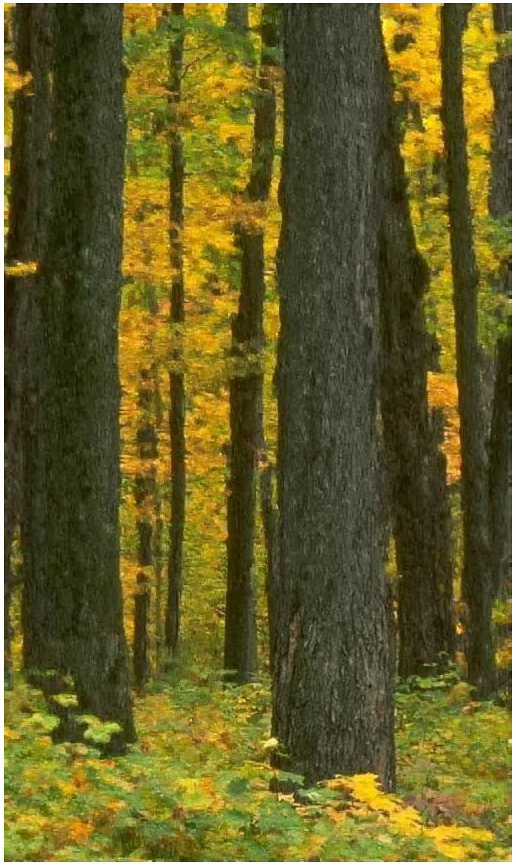
A

A’

Will it work?

Super-resolution (result!)

B



B'

Things to remember

- Texture synthesis and hole-filling can be thought of as a form of probabilistic hallucination
- Simple, similarity-based matching is a powerful tool
  - Synthesis
  - Hole-filling
  - Transfer

Next class

- Blending and gradient-domain methods