



# >>> 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



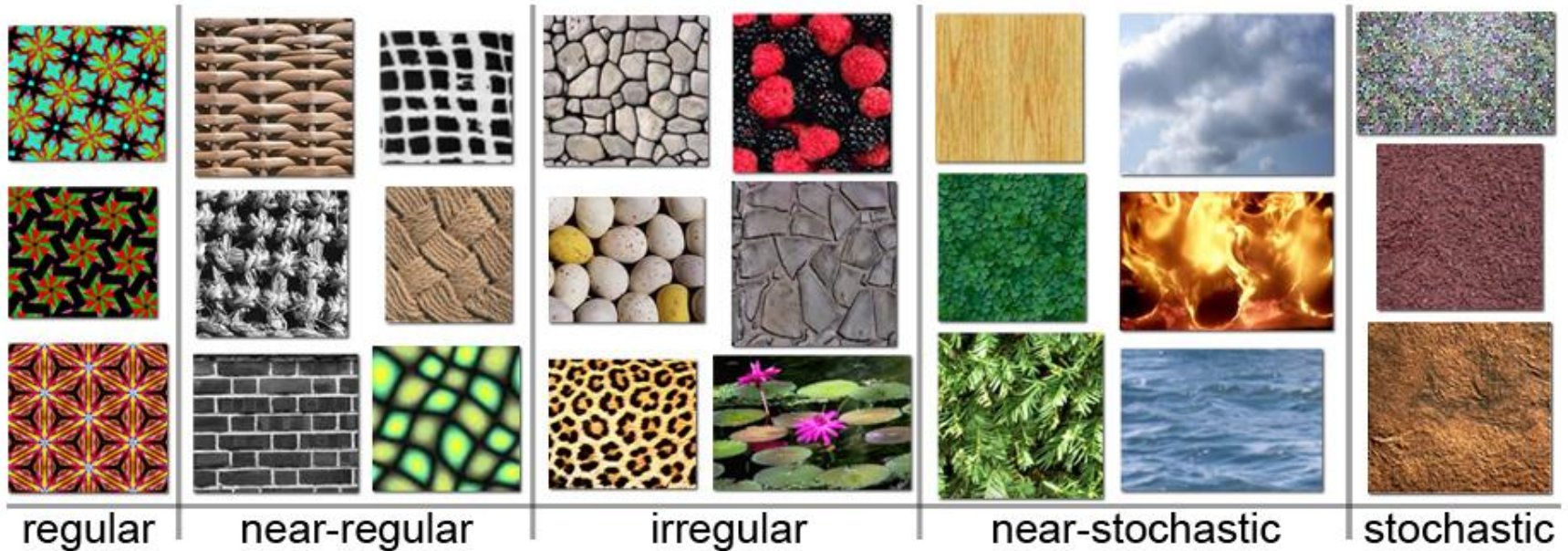
yogurt

# Texture Synthesis

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



# The Challenge



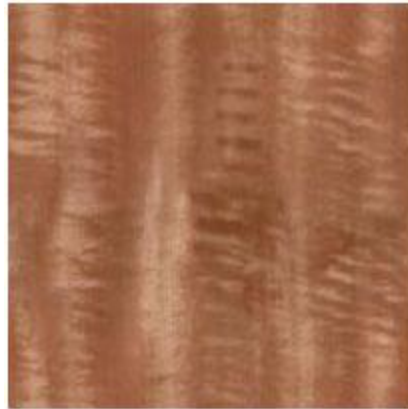
Need to model the whole spectrum: from repeated to stochastic texture



# 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



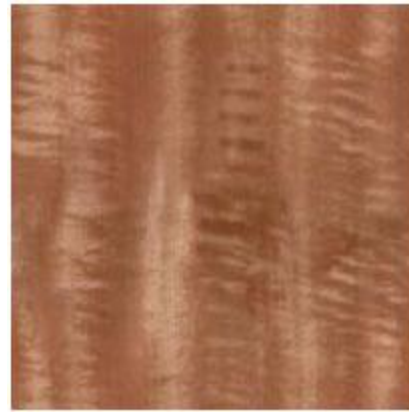
- D. J. Heeger and J. R. Bergen. Pyramid-based texture analysis/synthesis. In *SIGGRAPH '95*.

# One idea: Build Probability Distributions

But it (usually) doesn't work

- Probability distributions are hard to model well

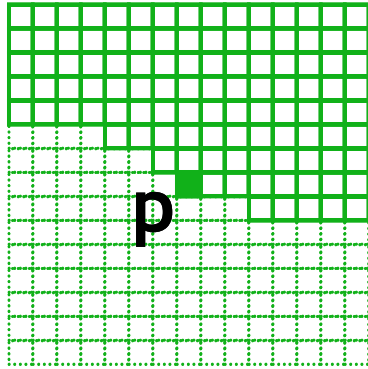
Input



Synthesized

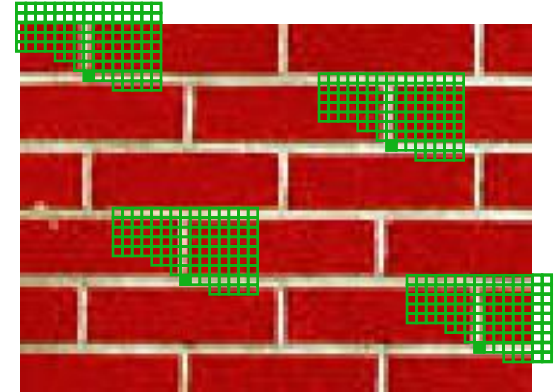


# Another idea: Sample from the image



Synthesizing a pixel

non-parametric  
sampling



Input image

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

# 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?

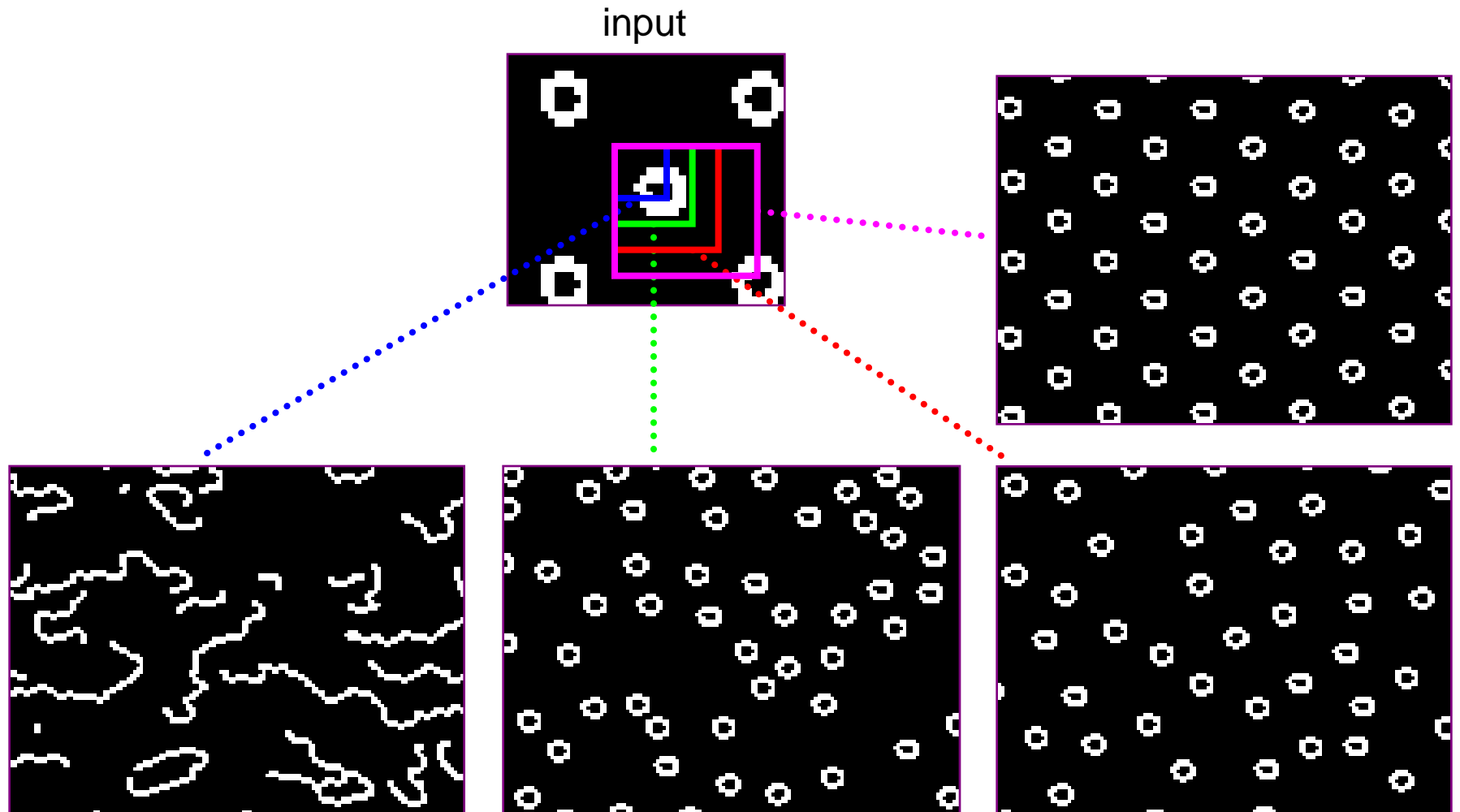


# Idea from Shannon (Information Theory)

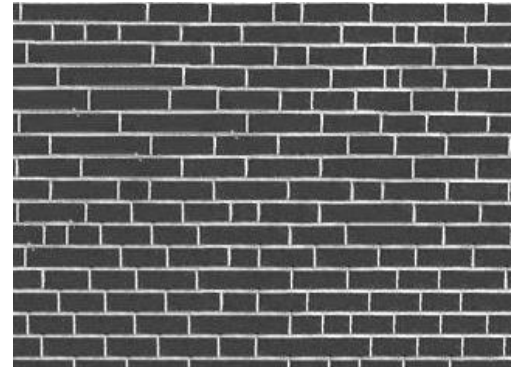
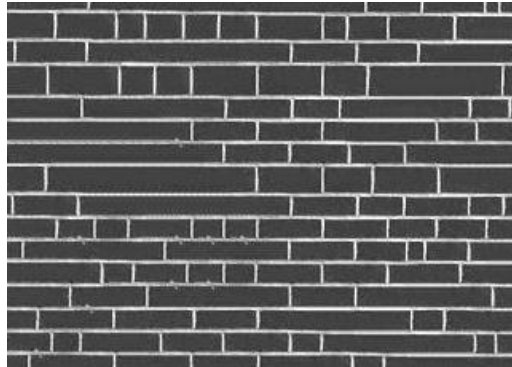
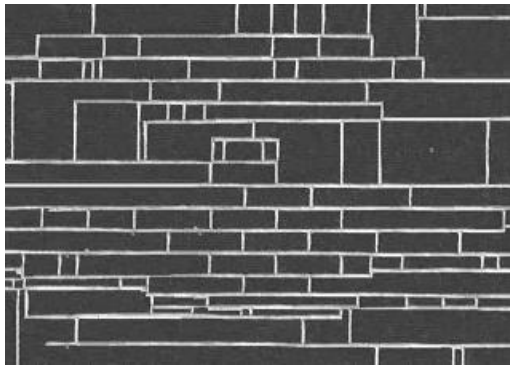
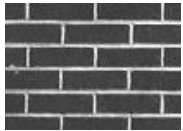
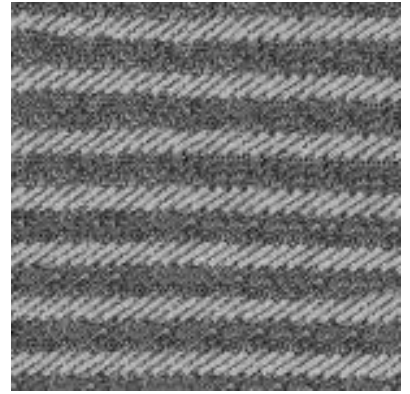
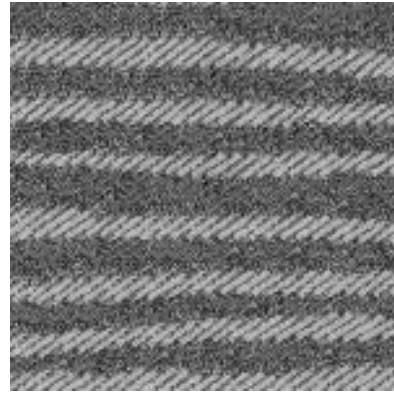
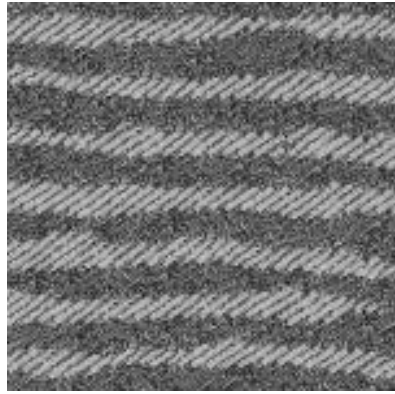
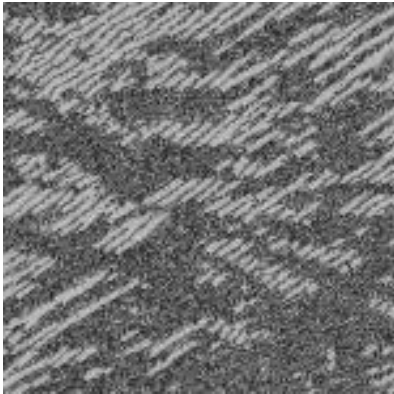
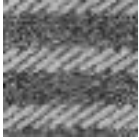
- Generate English-sounding sentences by modeling the probability of each word given the previous words (n-grams)
- Large “n” will give more structured sentences

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

# Size of Neighborhood Window



# Varying Window Size



Increasing window size



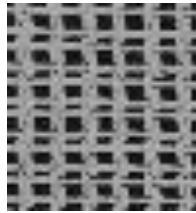
# Texture synthesis algorithm

- 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

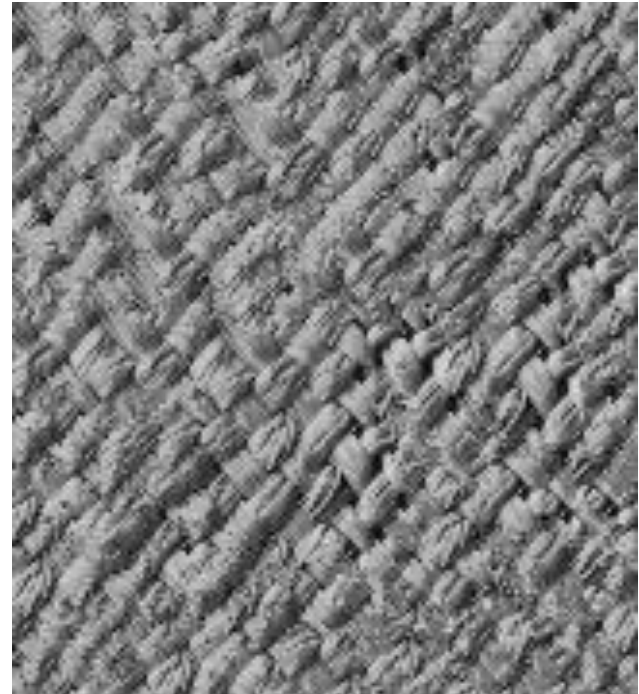
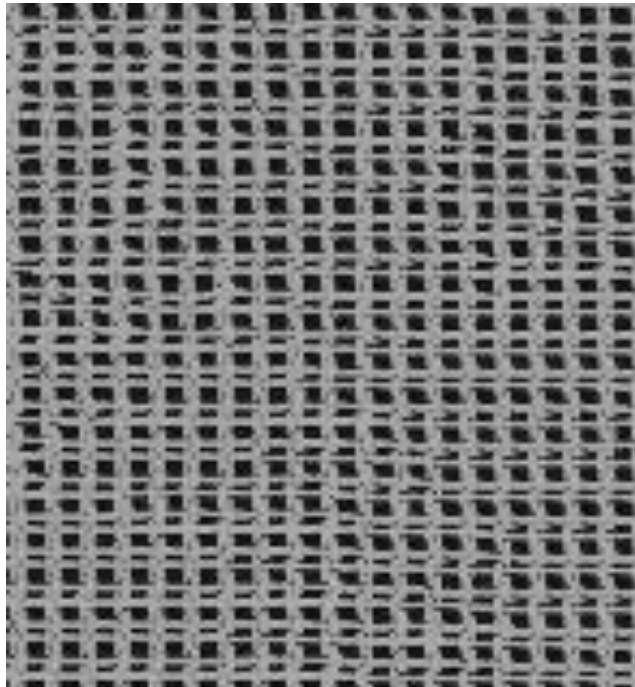


# Synthesis Results

french canvas



rafia weave

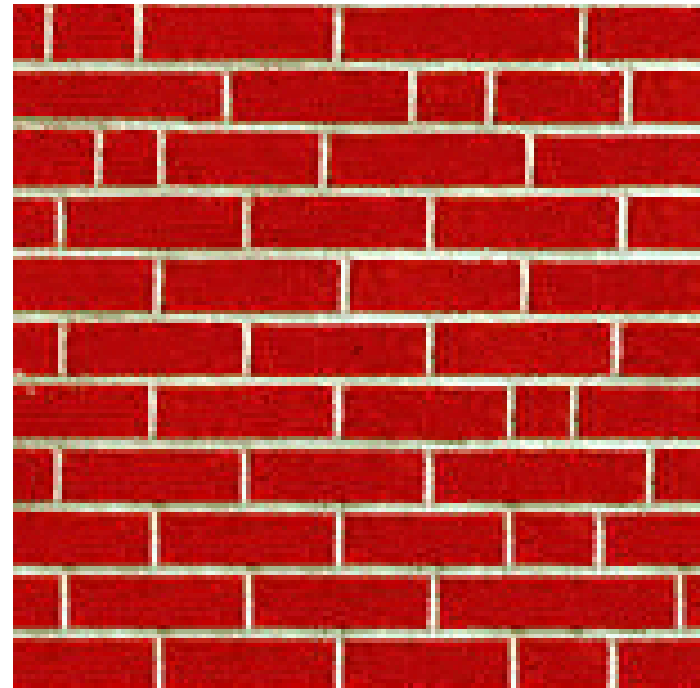
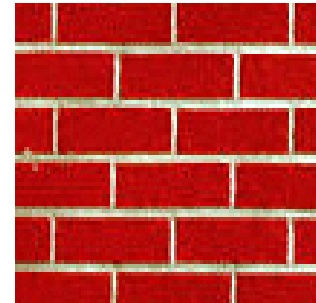


# More Results

white bread



brick wall

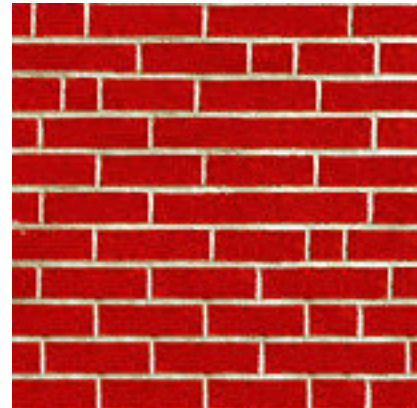
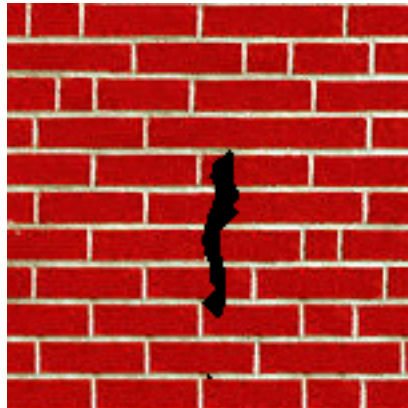


# Homage to Shannon

coming in the unsensational  
r Dick Gephardt was fair  
rful riff on the looming  
nly asked, "What's your  
tions?" A heartfelt sigh  
story about the emergen  
es against Clinton. "Boy  
g people about continuin  
ardt began, patiently obs  
s, that the legal system k  
g with this latest tanger

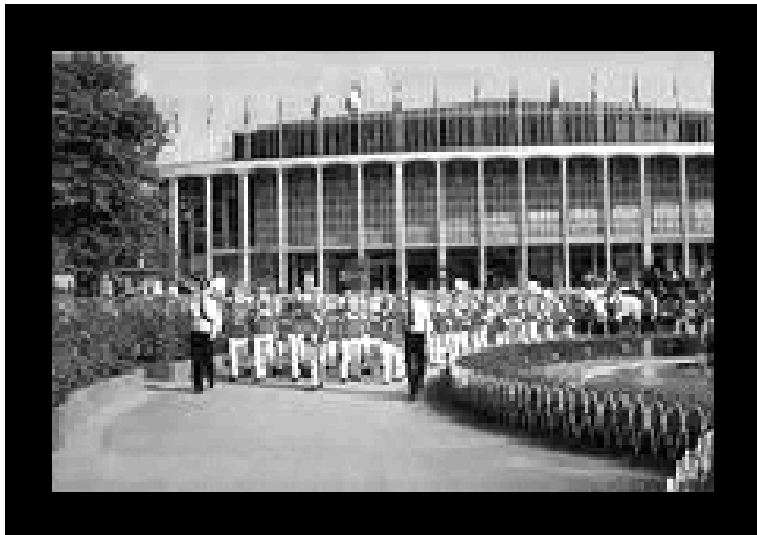
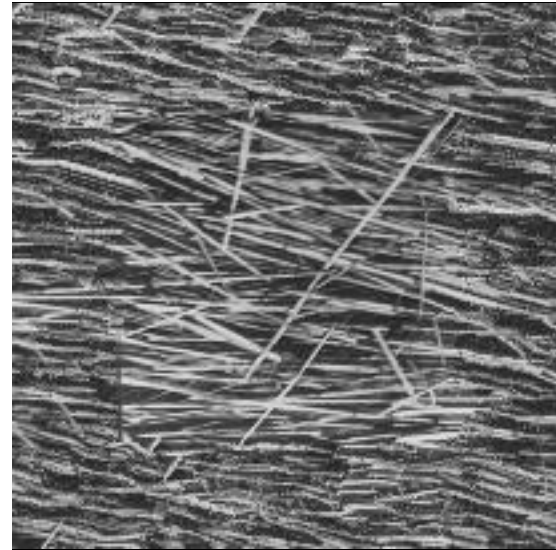
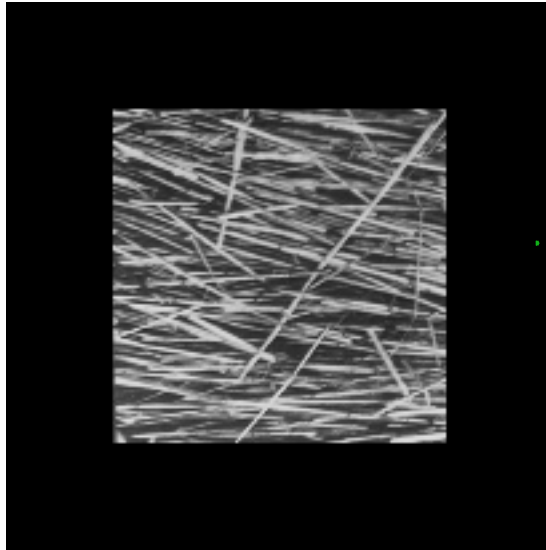
ithaim. them . "Whnephartfe lartifelintomimen  
fel ck Clirtioout omaim thartfelins.f out s anetc  
the ry onst wartfe lck Gephtoomimeationl sigak  
Chiooufit Clinut Cll riff on. hat's yodn, parut tly  
ons youtonsteht wasked, paim t sahe loo riff on l  
nskoneploourtfeas leil A nst Clit, "Wleontongal s  
k Clirtioouirtfepe.ong pme abegal fartfenstemem  
tiensteneltorydt telemephminsverdt was agemer  
ff ons artientont Cling peme as artfe atich, "Boui s  
aal s fartfelt sig pedrtldt ske abounutie aboutioo  
tfaonewwas yow abowonthardt thatins fain, ped, '  
ains. them, pabout wasy arfint couitly d, l n A h  
le emthrdngbooreme agas fa bontinsyst Clinut  
ory about continst Clipeopinst Cloke agatiff out C  
stome zinemen tly ardt beoraboul n, thenly as t C  
cons faimeme Diontont wat coutlyohgans as fan  
ien, phrtfaul, "Wbaut cout congagal comininga  
mifmst Cliiy abon al coountha.emungait tfoun  
Vhe loocrystan loontieph. intly on, tieoplegatick C  
ul fatiecontly atie Diontiomt wal s f tbegae ener  
nthahsat's enenhhbas fan. "intchthorv abonsv

# Hole Filling





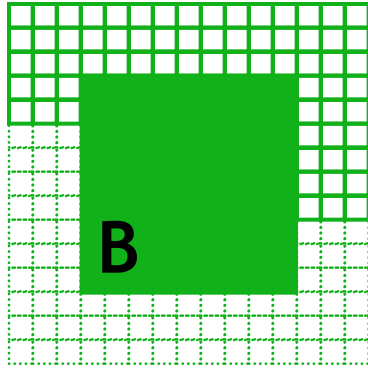
# Extrapolation



# Summary

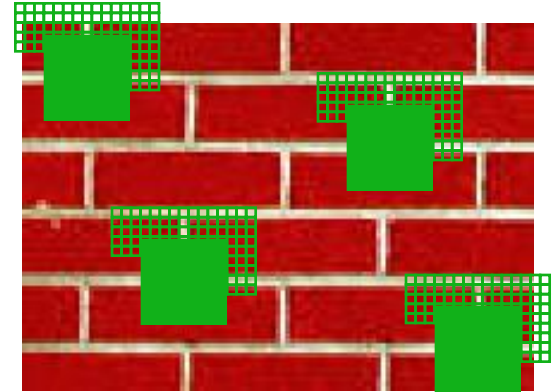
- The Efros & Leung texture synthesis algorithm
  - Very simple
  - Surprisingly good results
  - Synthesis is easier than analysis!
  - ...but very slow

# Image Quilting [Efros & Freeman 2001]



Synthesizing a block

non-parametric  
sampling

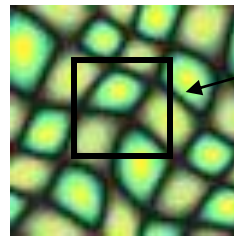


Input image

- Observation: neighbor pixels are highly correlated

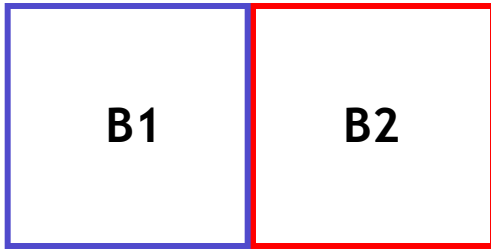
Idea: unit of synthesis = block

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

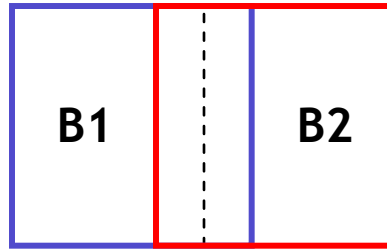


block

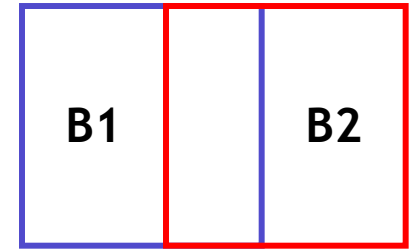
Input texture



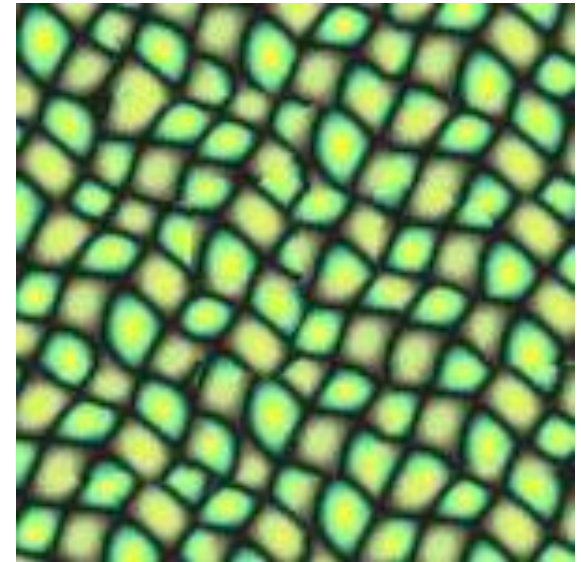
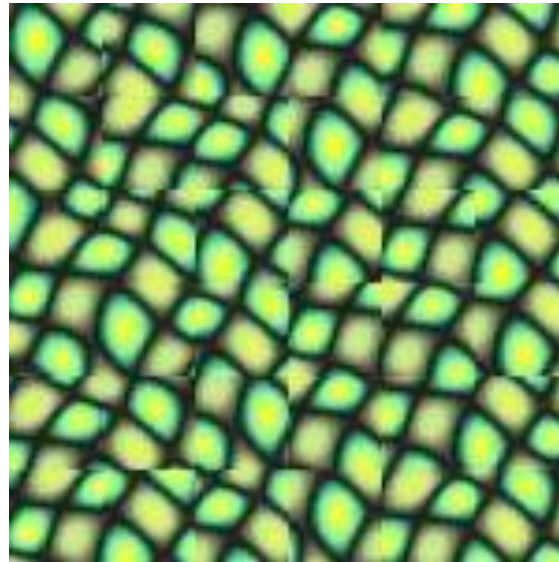
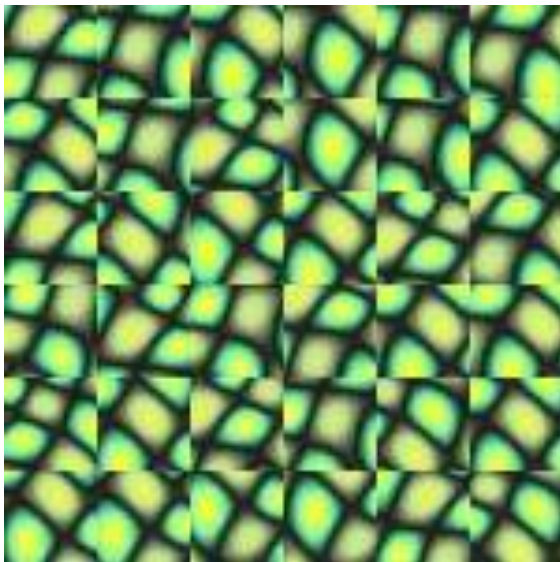
Random placement  
of blocks



Neighboring blocks  
constrained by overlap



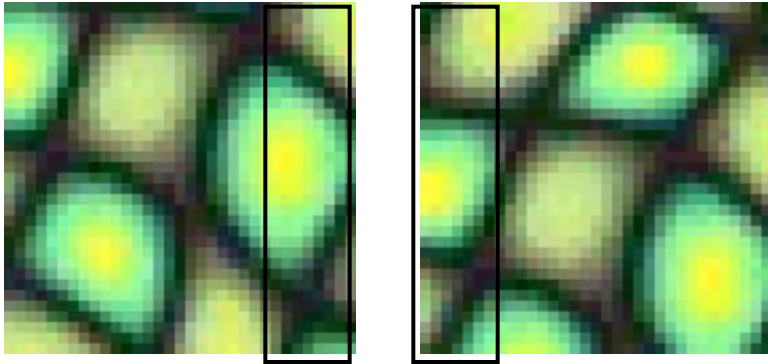
Minimal error  
boundary cut



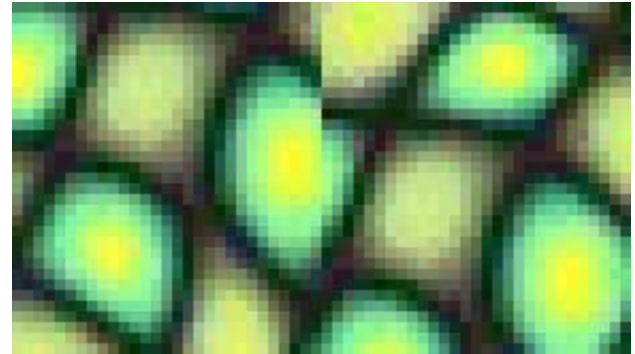


# Minimal error boundary

overlapping blocks

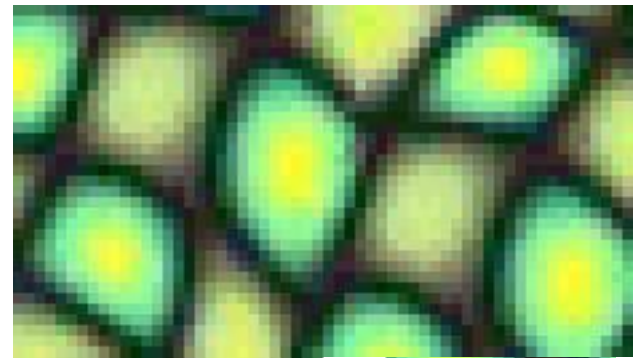


vertical boundary

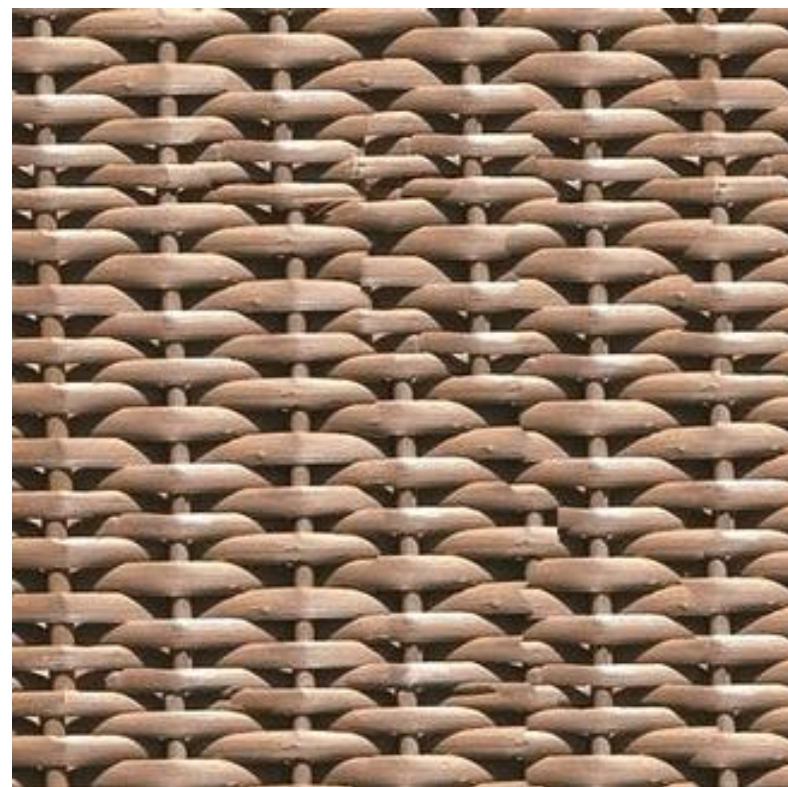
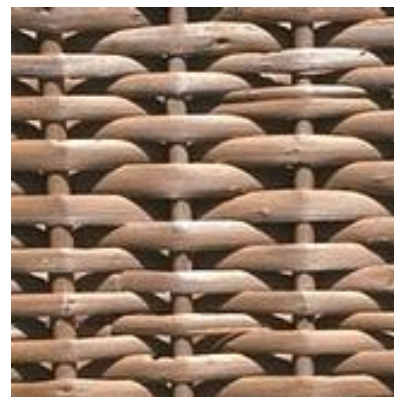


A diagram illustrating the calculation of overlap error. Two vertical rectangular blocks of a heatmap image are shown side-by-side. A large left square bracket is positioned to the left of the blocks. A minus sign is placed between the two blocks. To the right of the minus sign is another large left square bracket. To the right of this second bracket is a large number '2'. To the right of the '2' is an equals sign. To the right of the equals sign is a vertical rectangular block of a heatmap image. A red line is drawn along the right edge of this block, representing the boundary. Two blue arrows point from the top of the two blocks in the first part of the diagram to the two brackets in the second part.

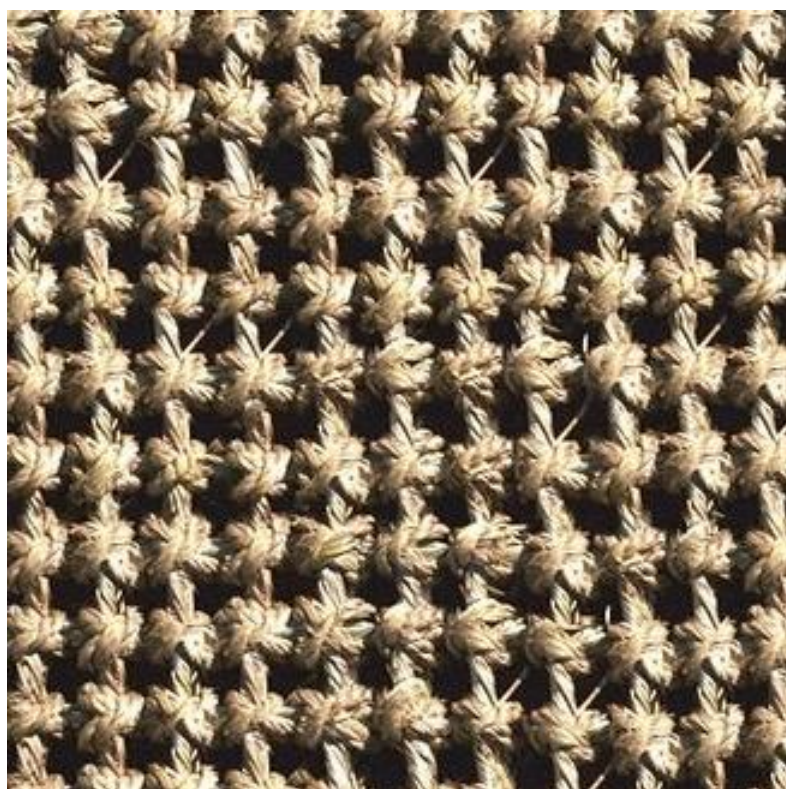
overlap error



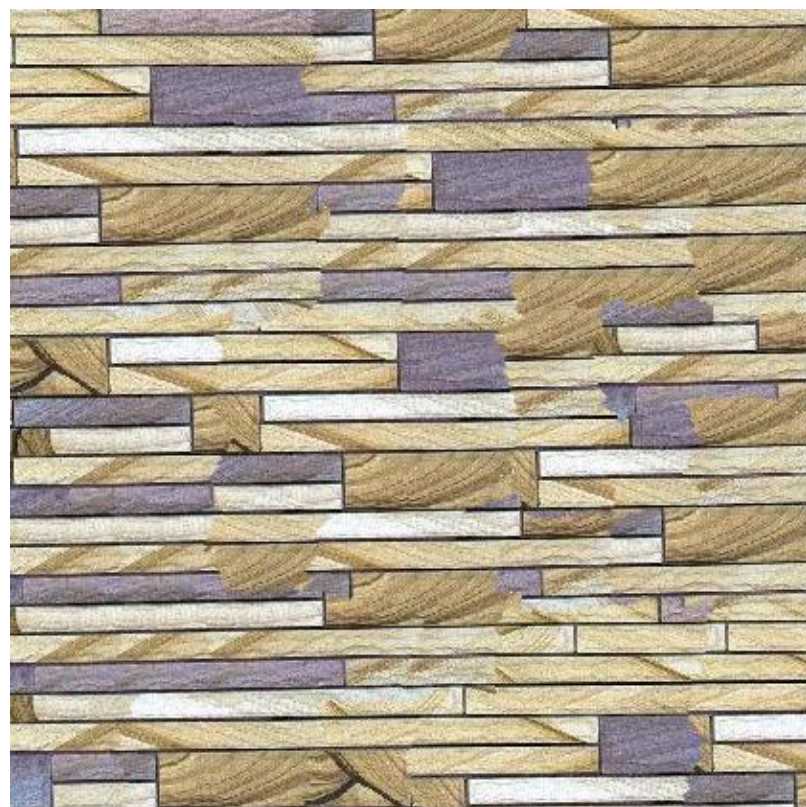
min. error boundary









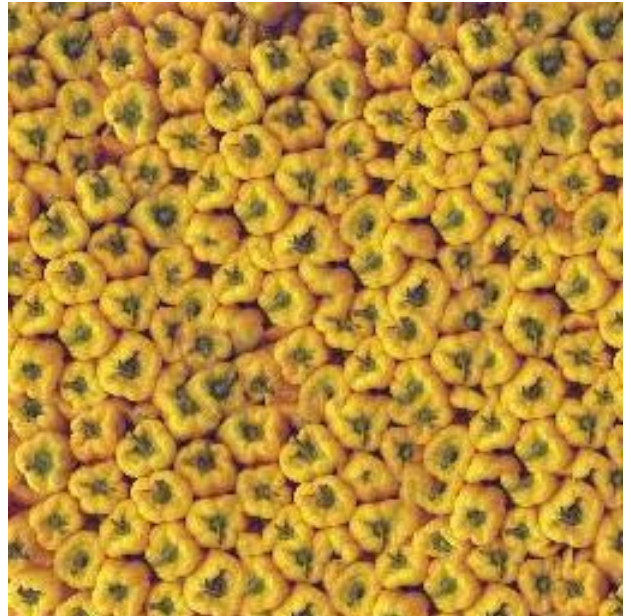
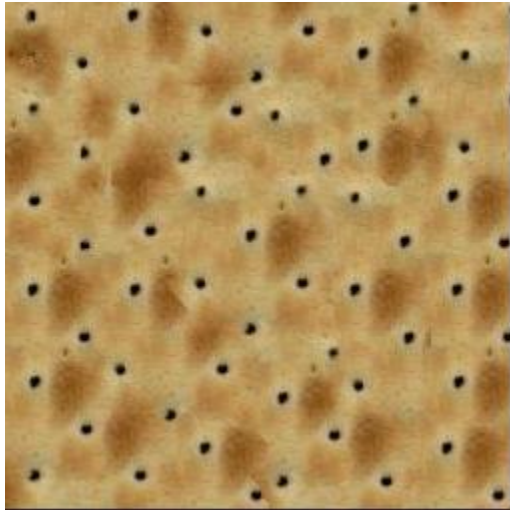
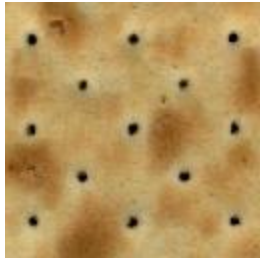




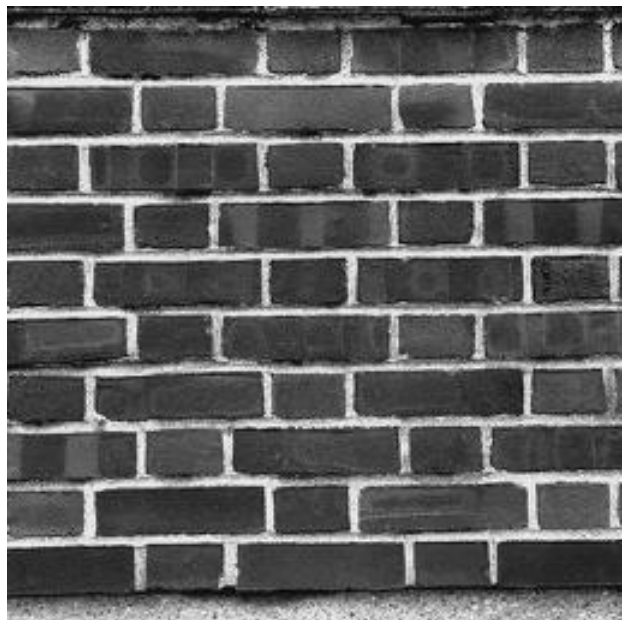








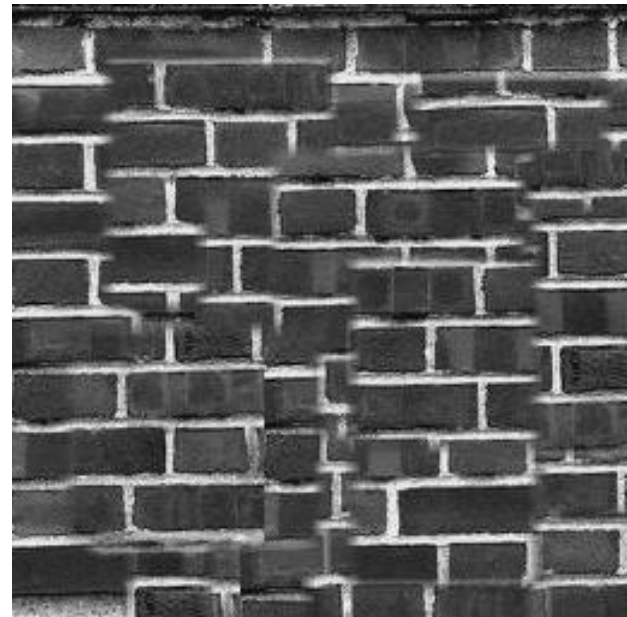




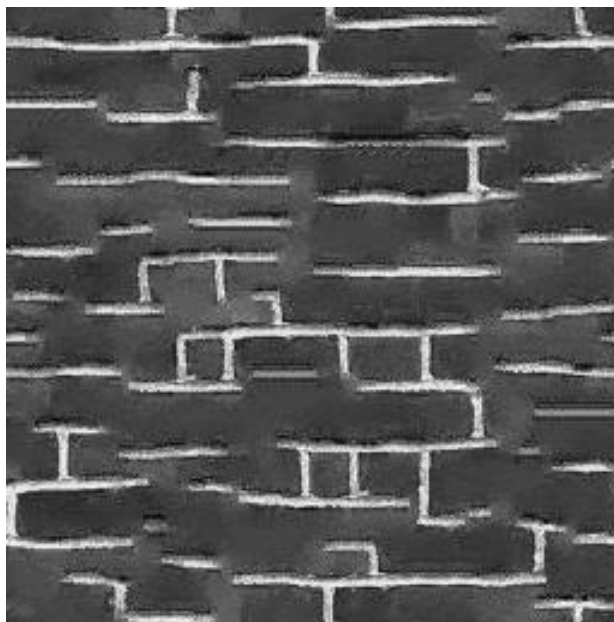
input image



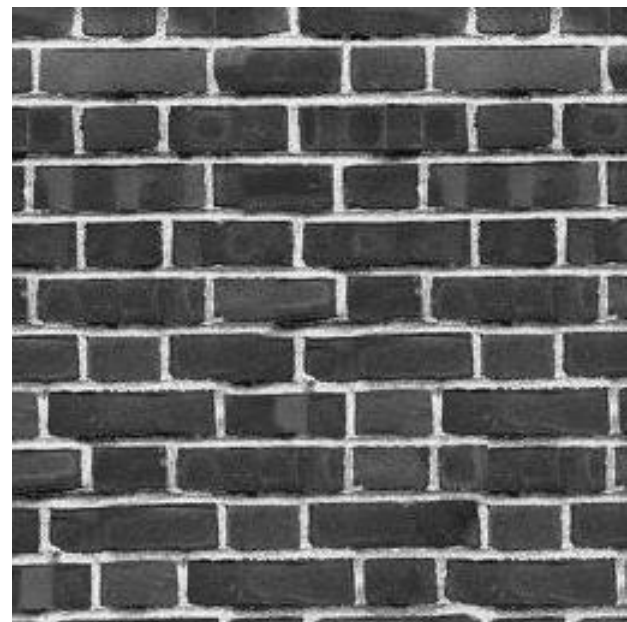
Portilla & Simoncelli



Xu, Guo & Shum



Wei & Levoy



Quilting

... of a visual cortical neuron—the in  
describing the response of that neuro  
ht as a function of position—is perhap  
functional description of that neuron.  
seek a single conceptual and mathem  
describe the wealth of simple-cell recep  
and neurophysiologically<sup>1-3</sup> and inferred  
especially if such a framework has the  
it helps us to understand the functio  
eeper way. Whereas no generic mo  
ussians (DOG), difference of offset C  
rivative of a Gaussian, higher derivati  
function, and so on—can be expecte  
imple-cell receptive field, we noneth

input image

... of a visual cortical neuron—the in  
describing the response of that neuro  
ht as a function of position—is perhap  
functional description of that neuron.  
seek a single conceptual and mathem  
describe the wealth of simple-cell recep  
and neurophysiologically<sup>1-3</sup> and inferred  
especially if such a framework has the  
it helps us to understand the functio  
eeper way. Whereas no generic mo  
ussians (DOG), difference of offset C  
rivative of a Gaussian, higher derivati  
function, and so on—can be expecte  
imple-cell receptive field, we noneth

Portilla & Simoncelli

... of a visual cortical neuron—the in  
describing the response of that neuro  
ht as a function of position—is perhap  
functional description of that neuron.  
seek a single conceptual and mathem  
describe the wealth of simple-cell recep  
and neurophysiologically<sup>1-3</sup> and inferred  
especially if such a framework has the  
it helps us to understand the functio  
eeper way. Whereas no generic mo  
ussians (DOG), difference of offset C  
rivative of a Gaussian, higher derivati  
function, and so on—can be expecte  
imple-cell receptive field, we noneth

Wei & Levoy

... of a visual cortical neuron—the in  
describing the response of that neuro  
ht as a function of position—is perhap  
functional description of that neuron.  
seek a single conceptual and mathem  
describe the wealth of simple-cell recep  
and neurophysiologically<sup>1-3</sup> and inferred  
especially if such a framework has the  
it helps us to understand the functio  
eeper way. Whereas no generic mo  
ussians (DOG), difference of offset C  
rivative of a Gaussian, higher derivati  
function, and so on—can be expecte  
imple-cell receptive field, we noneth

Xu, Guo & Shum

... of a visual cortical neuron—the in  
describing the response of that neuro  
ht as a function of position—is perhap  
functional description of that neuron.  
seek a single conceptual and mathem  
describe the wealth of simple-cell recep  
and neurophysiologically<sup>1-3</sup> and inferred  
especially if such a framework has the  
it helps us to understand the functio  
eeper way. Whereas no generic mo  
ussians (DOG), difference of offset C  
rivative of a Gaussian, higher derivati  
function, and so on—can be expecte  
imple-cell receptive field, we noneth

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.

This section shows a sampling of the duplication of soldiers.



Original photograph



# In-painting natural scenes



# Key idea: Filling order matters

## In-painting Result



Image with Hole



Raster-Scan Order



Onion-Peel  
(Concentric Layers)



Gradient-Sensitive  
Order

# Filling order

Fill a pixel that:

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



# Comparison



Original



With Hole



Onion-Ring Fill



Criminisi

# Comparison



**a**



**b**



Concentric Layers

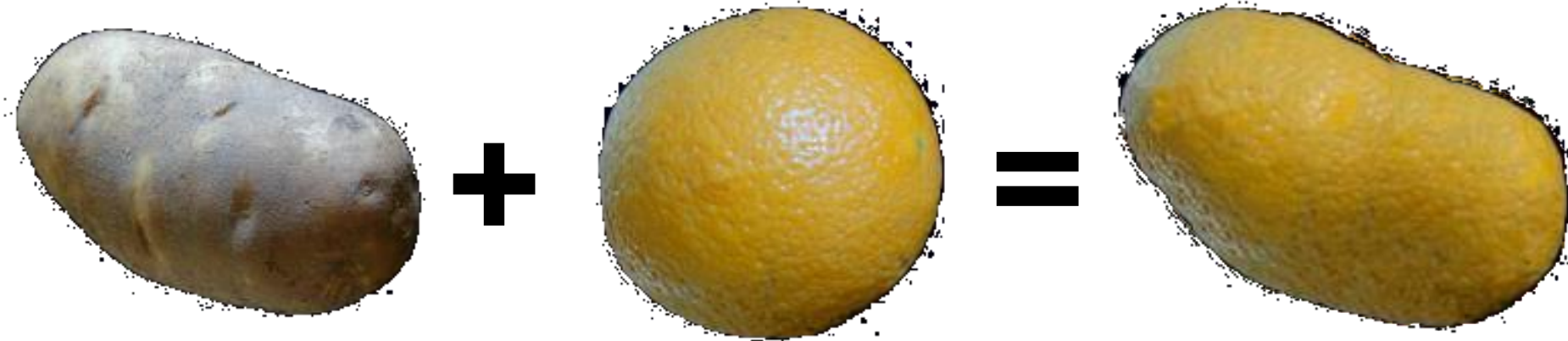


Gradient Sensitive



# Texture Transfer

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



# Texture Transfer



Constraint

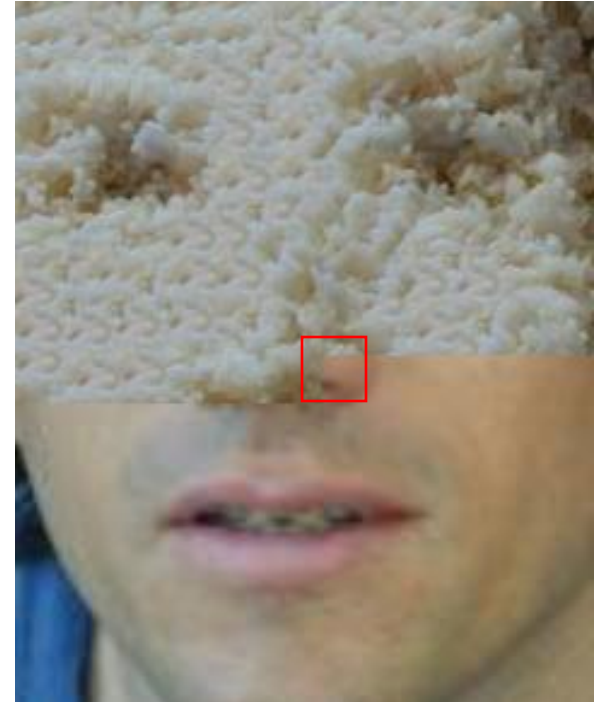


Texture sample



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



source texture



target image



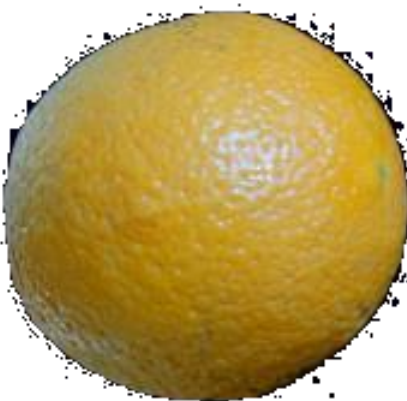
correspondence maps



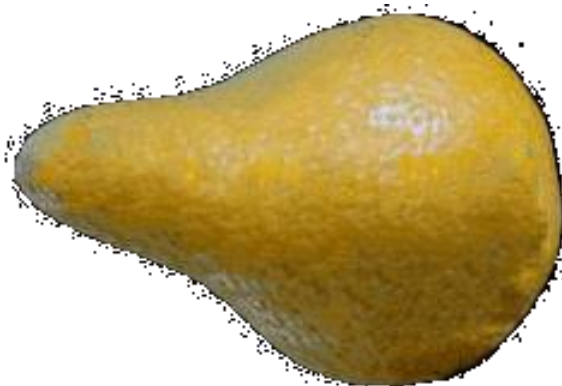
texture transfer result



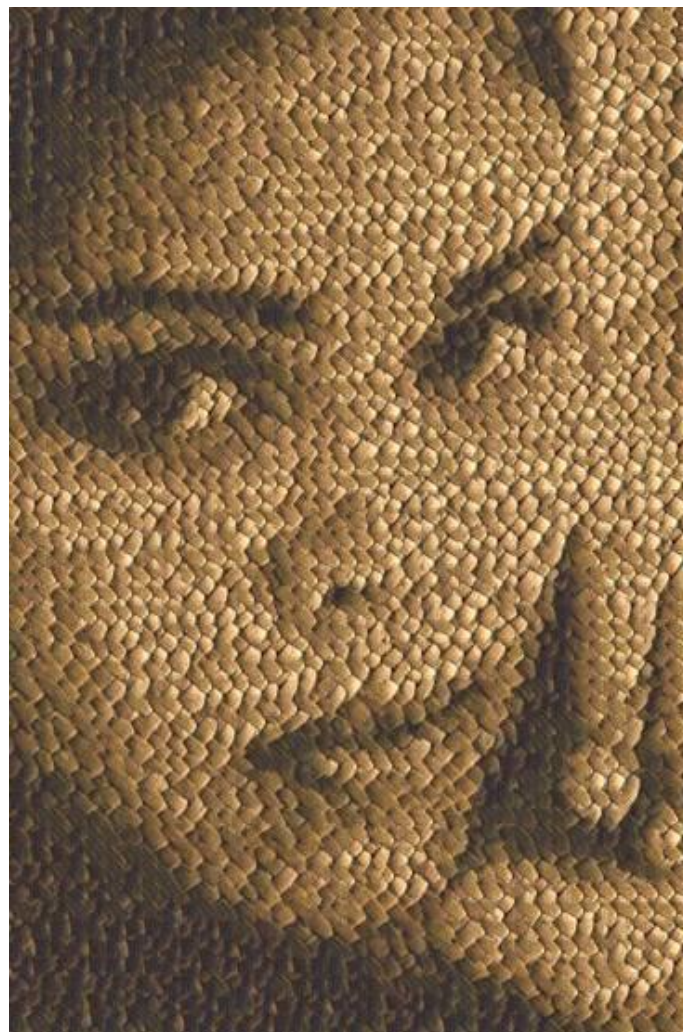
+



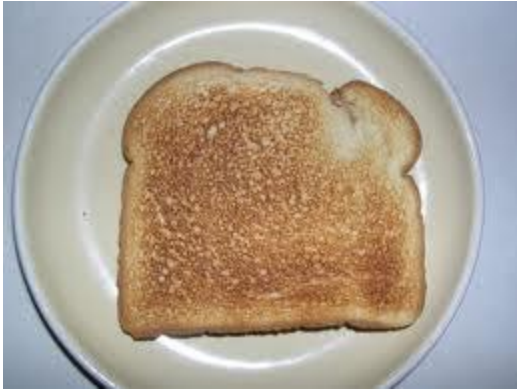
=







# 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'





# Related idea: Image Analogies

Learn filter from examples

A



A'



B



B'



# Blur Filter



**Unfiltered source ( $A$ )**



**Filtered source ( $A'$ )**



**Unfiltered target ( $B$ )**



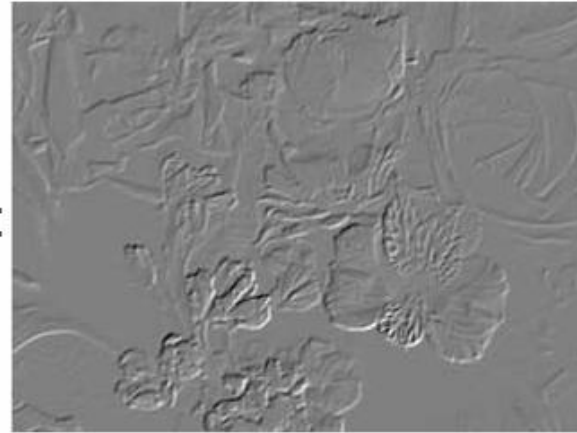
**Filtered target ( $B'$ )**



# Edge Filter



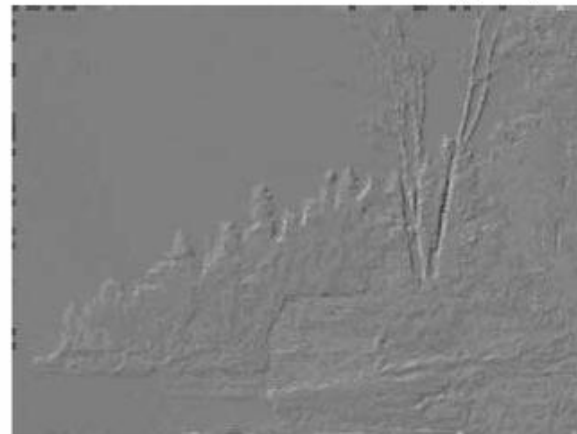
**Unfiltered source ( $A$ )**



**Filtered source ( $A'$ )**

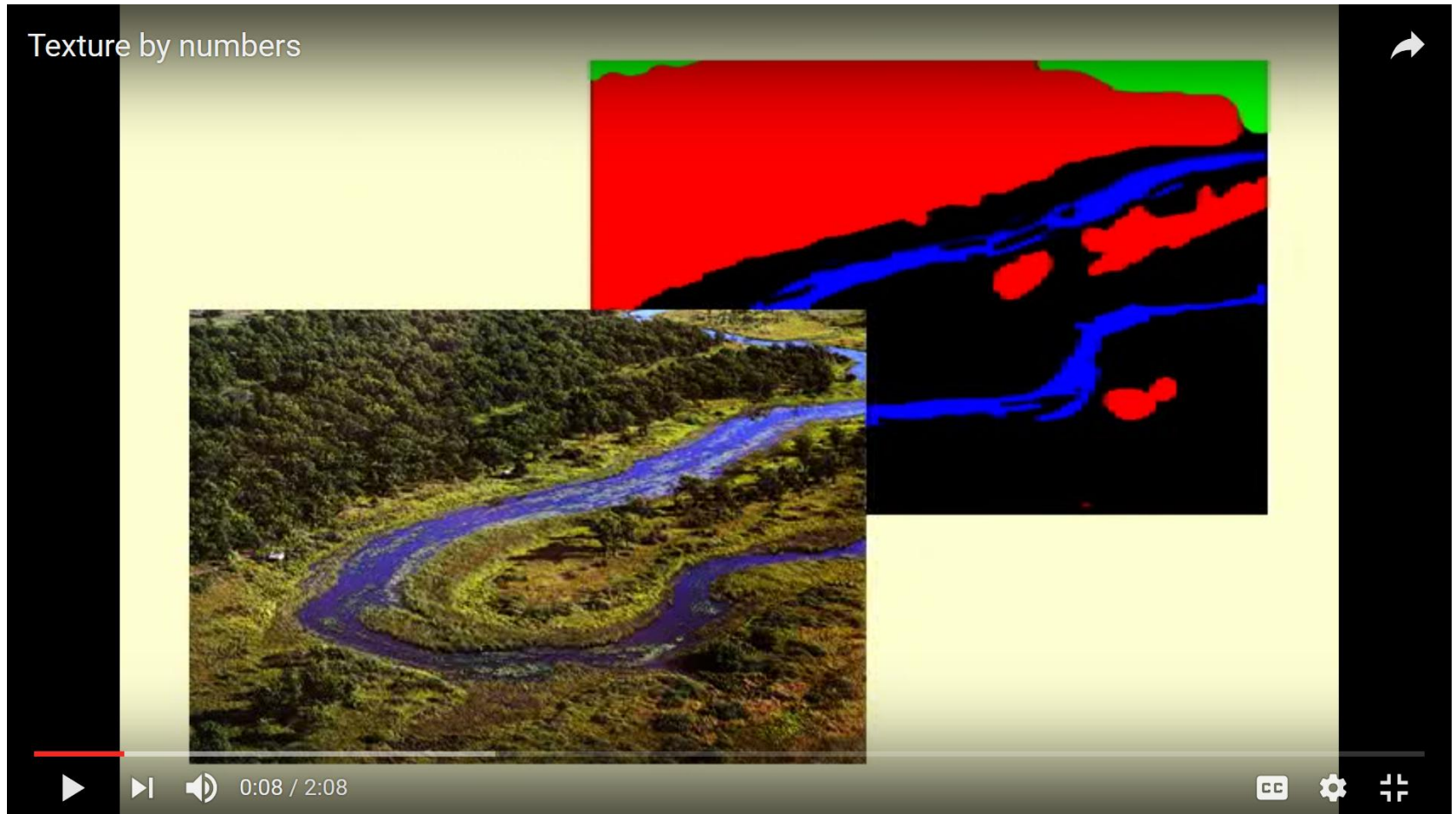


**Unfiltered target ( $B$ )**



**Filtered target ( $B'$ )**

# Texture by numbers



<https://www.youtube.com/watch?v=0F4bpf0Tesk>

# Colorization



**Unfiltered source ( $A$ )**



**Filtered source ( $A'$ )**



**Unfiltered target ( $B$ )**



**Filtered target ( $B'$ )**

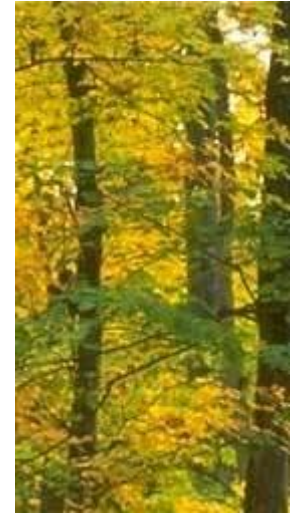
# Super-resolution



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