图像缩放与补全

彭思达

计算摄影学第九讲

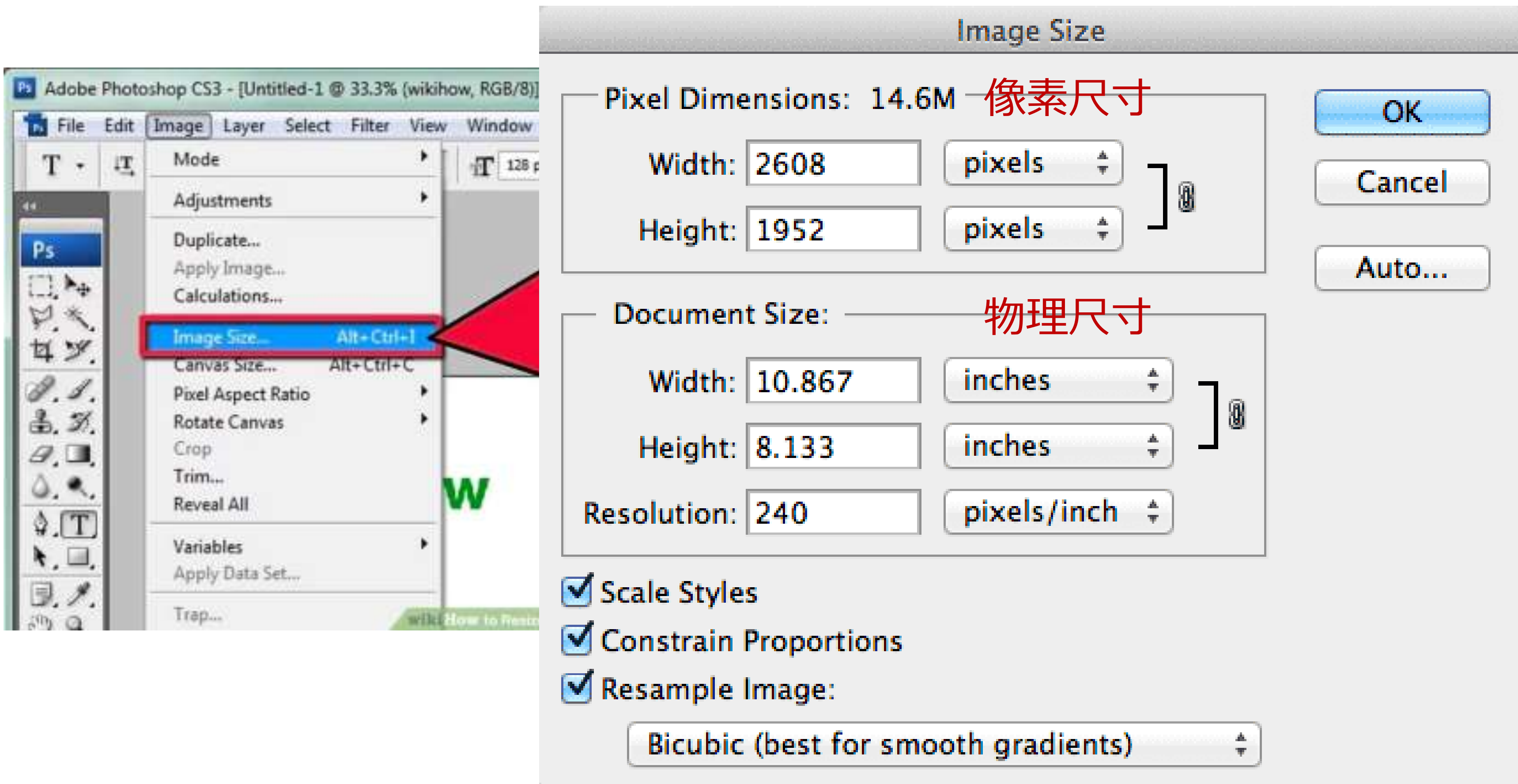
Outline

- Image resizing （图像缩放）

- Image completion （图像补全）

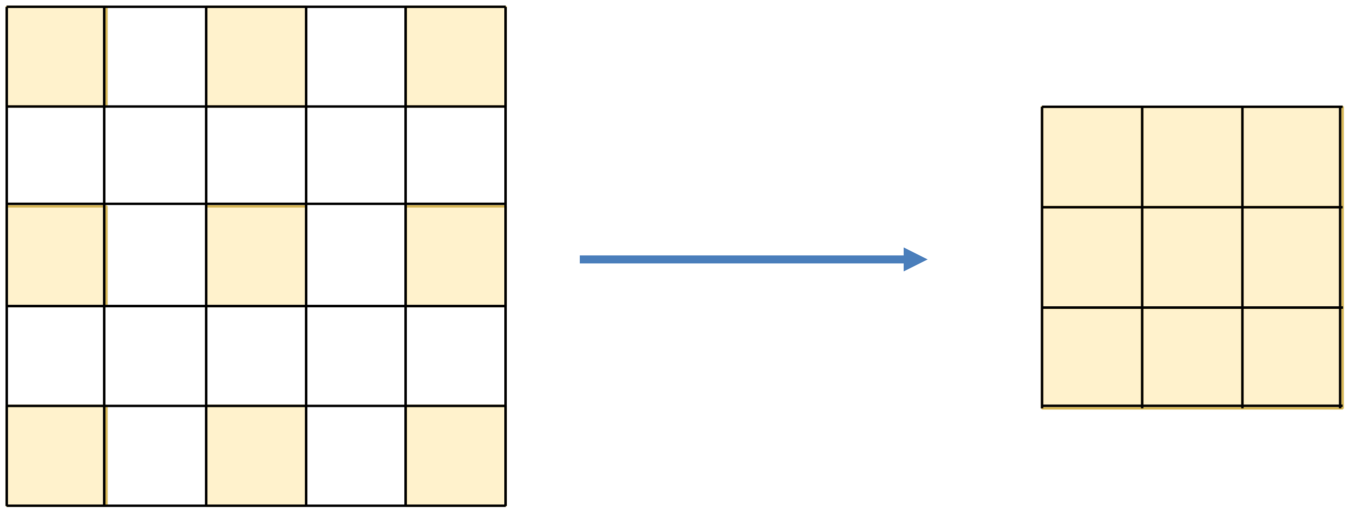
Image resizing

Change image size / resolution in Photoshop



Sampling

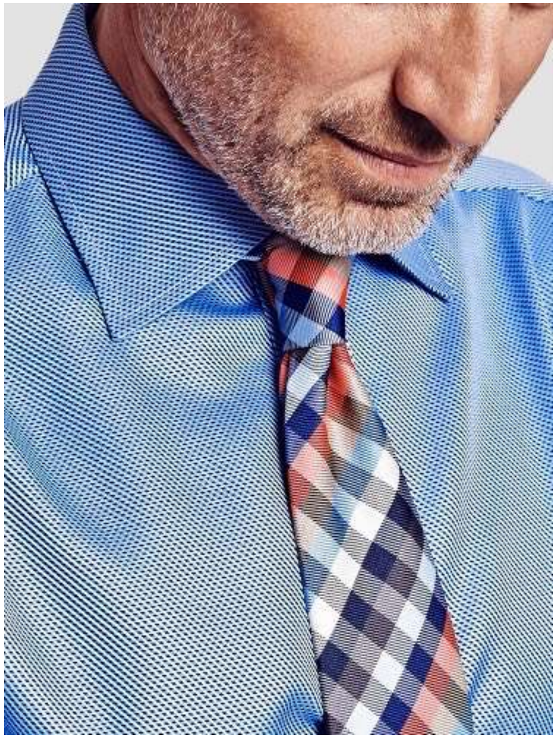
Reducing image size – down-sampling



Is sampling really so easy?

Moiré Patterns in Imaging

[mwɑ:]



lystit.com

Skip odd rows and columns

Slides borrowed from Lingqi Yan (UCSB)

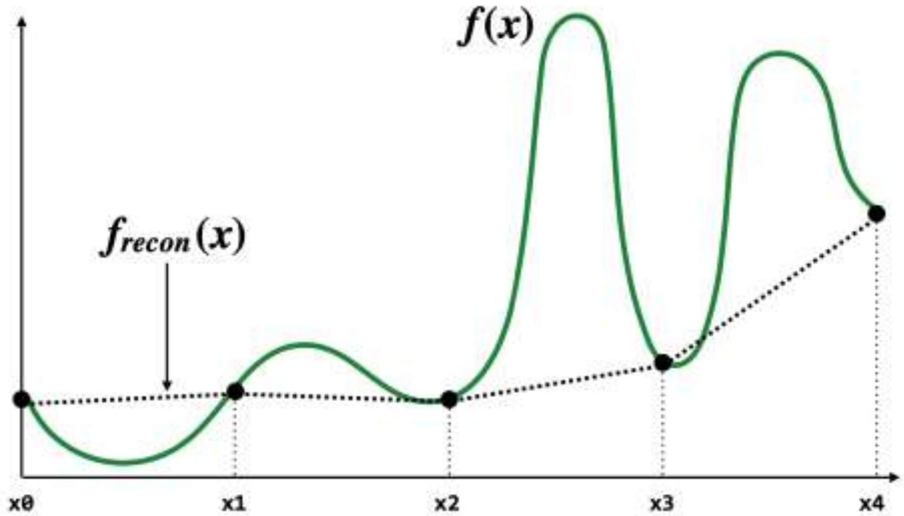
Is sampling really so easy?

Wagon Wheel Illusion (False Motion)

Slides borrowed from Lingqi Yan (UCSB)

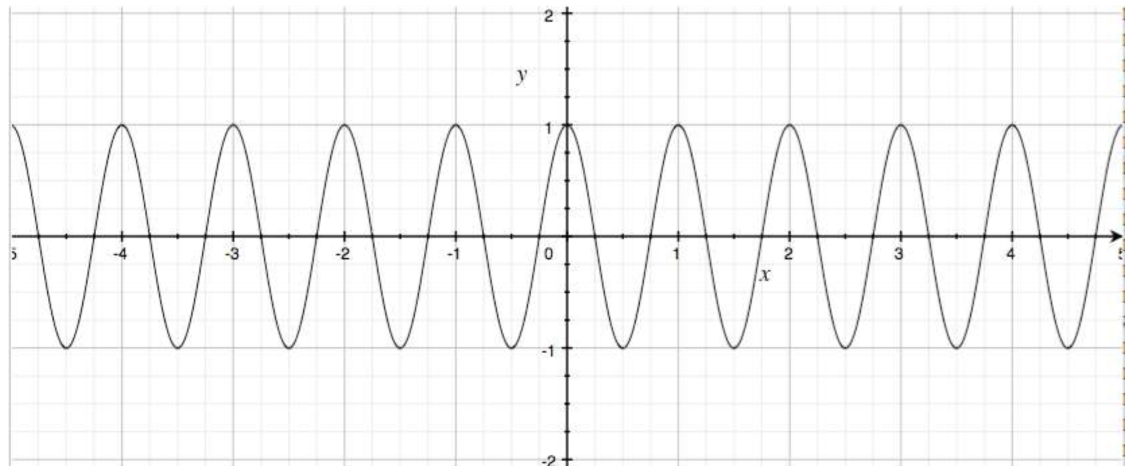
Aliasing

- Aliasing - artifacts due to sampling
- Why does aliasing happen?
  - Signals are changing too fast but sampled too slow

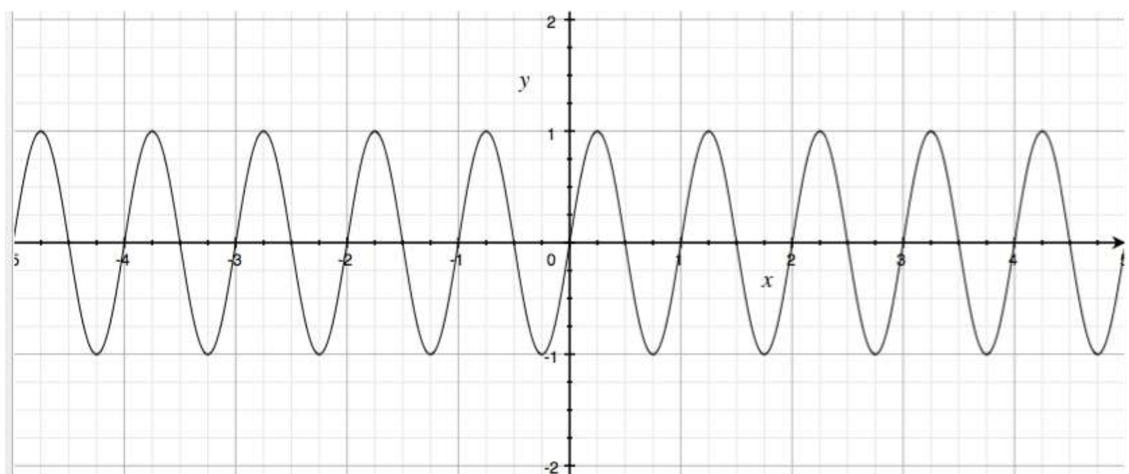


How to mathematically describe the changing speed of a signal?

Sines and Cosines



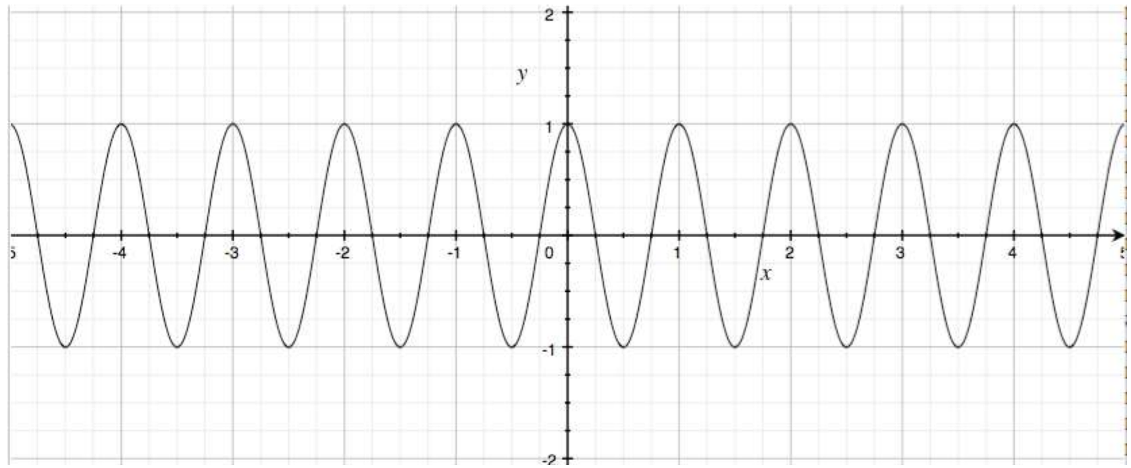
$\cos 2\pi x$



$\sin 2\pi x$

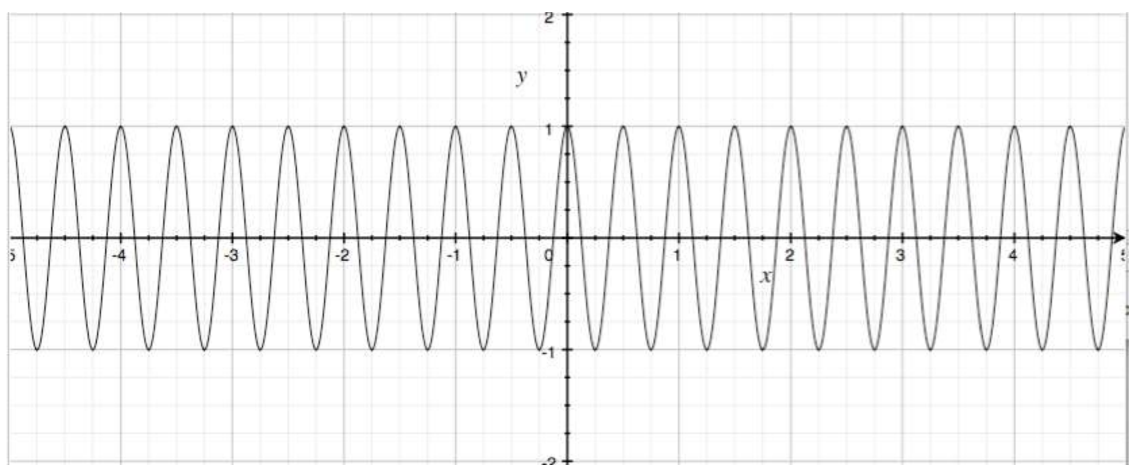
Frequencies $\cos 2\pi f x$

$$f = \frac{1}{T}$$



$\cos 2\pi x$

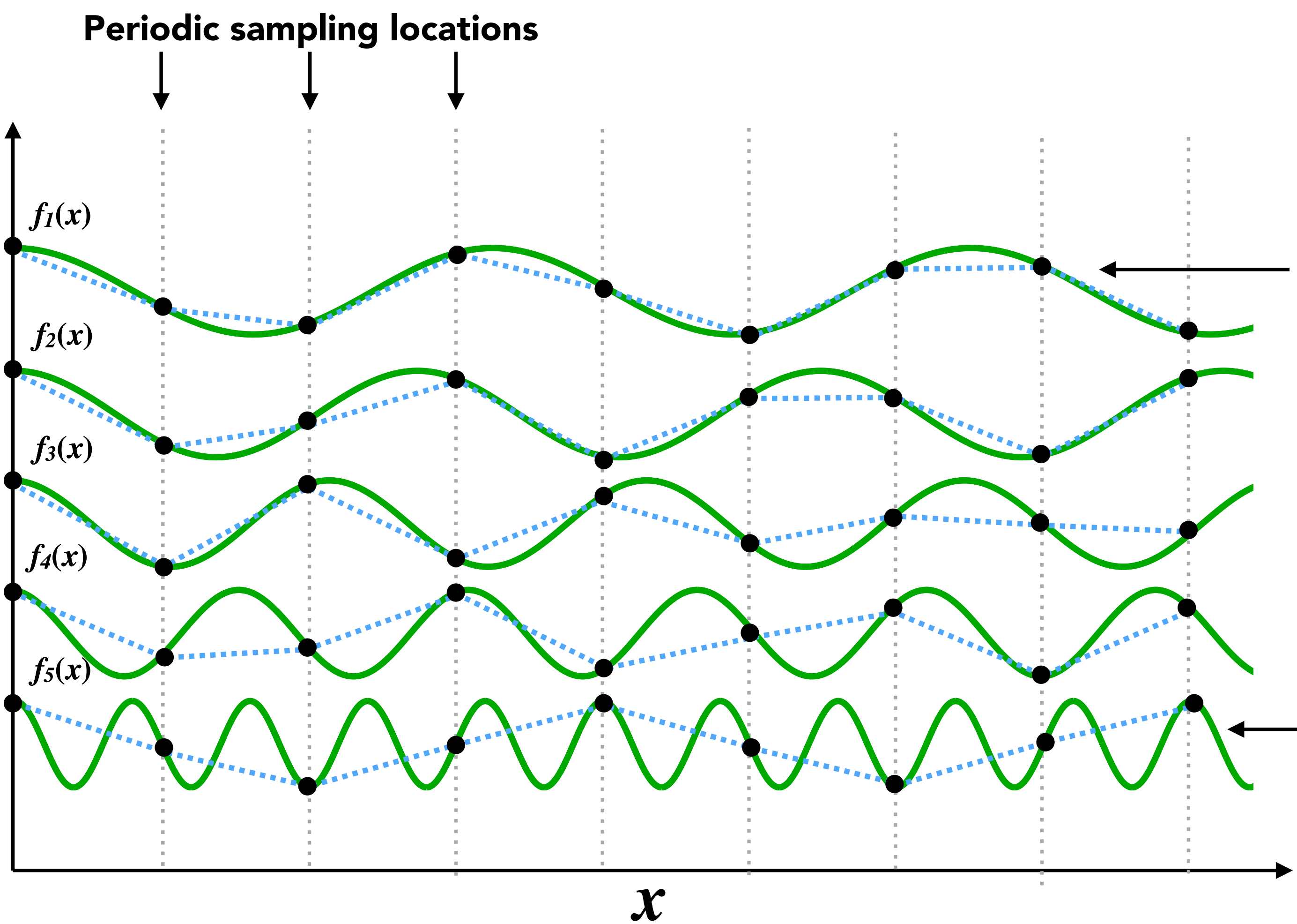
$$f = 1$$



$\cos 4\pi x$

$$f = 2$$

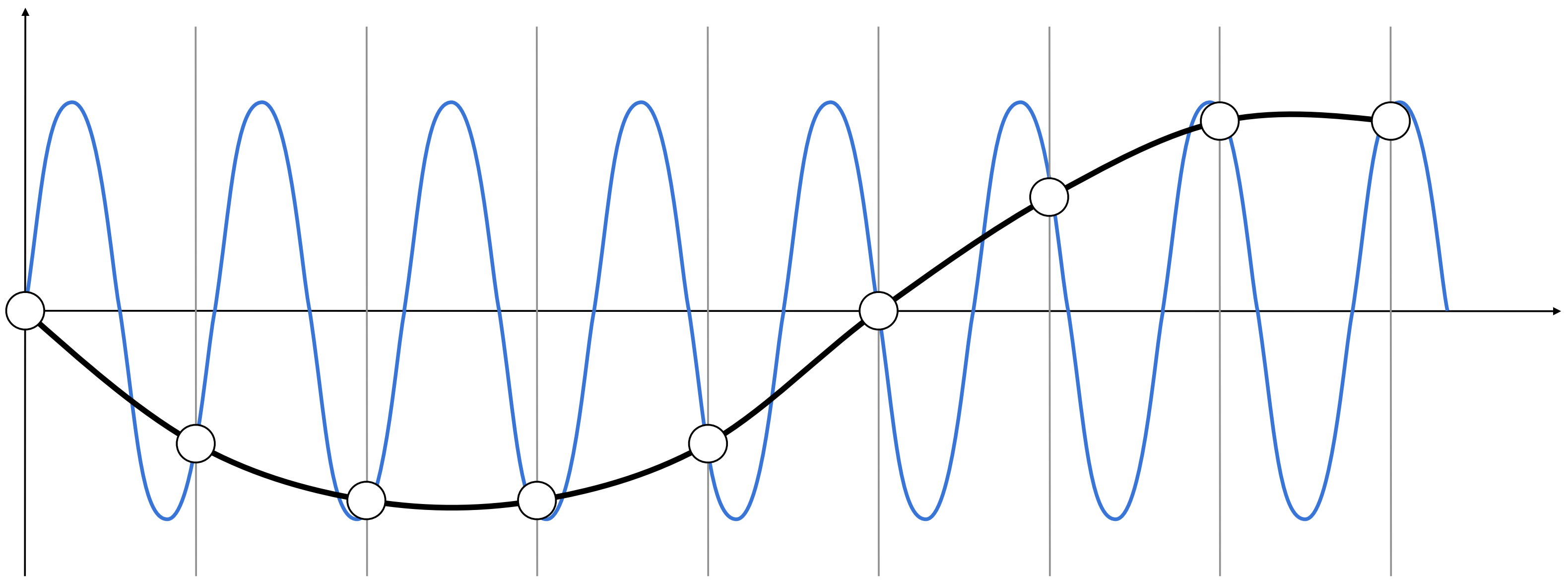
Higher Frequencies Need Faster Sampling



Low-frequency signal: sampled adequately for reasonable reconstruction

High-frequency signal is insufficiently sampled: reconstruction incorrectly appears to be from a low frequency signal

Undersampling Creates Frequency Aliases



High-frequency signal is insufficiently sampled: samples erroneously appear to be from a low-frequency signal

Two frequencies that are indistinguishable at a given sampling rate are called "aliases"

What are the frequencies of arbitrary signals

Fourier, Joseph (1768-1830)

French mathematician who discovered that any periodic motion can be written as a superposition of sinusoidal and cosinusoidal vibrations. He developed a mathematical theory of heat in *Théorie Analytique de la Chaleur (Analytic Theory of Heat)*, (1822), discussing it in terms of differential equations.

Fourier was a friend and advisor of Napoleon. Fourier believed that his health would be improved by wrapping himself up in blankets, and in this state he tripped down the stairs in his house and killed himself. The paper of Galois which he had taken home to read shortly before his death was never recovered.

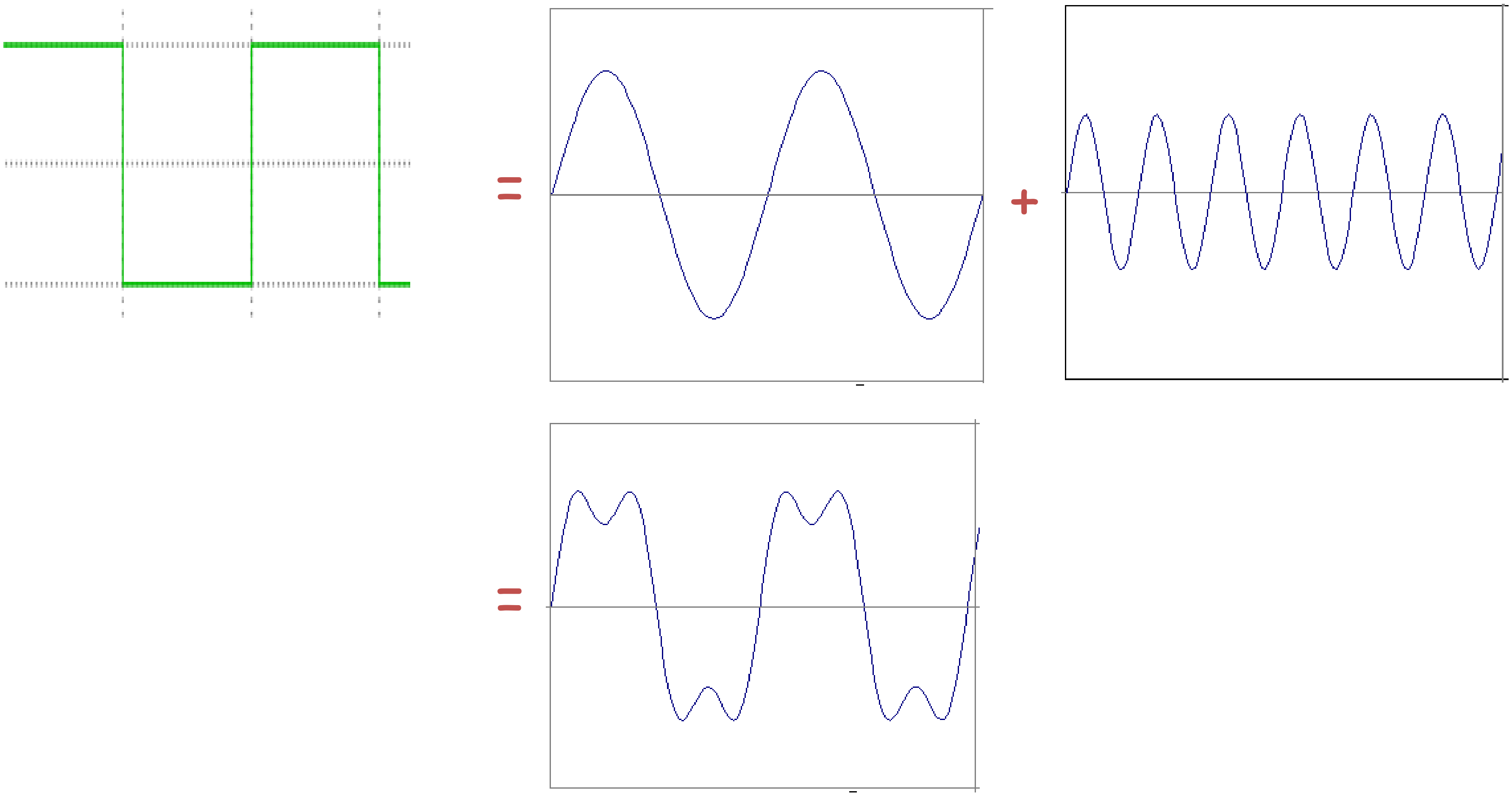
SEE ALSO: Galois

Additional biographies: MacTutor (St. Andrews), Bonn

Slides: Hoiem, Efros, and others

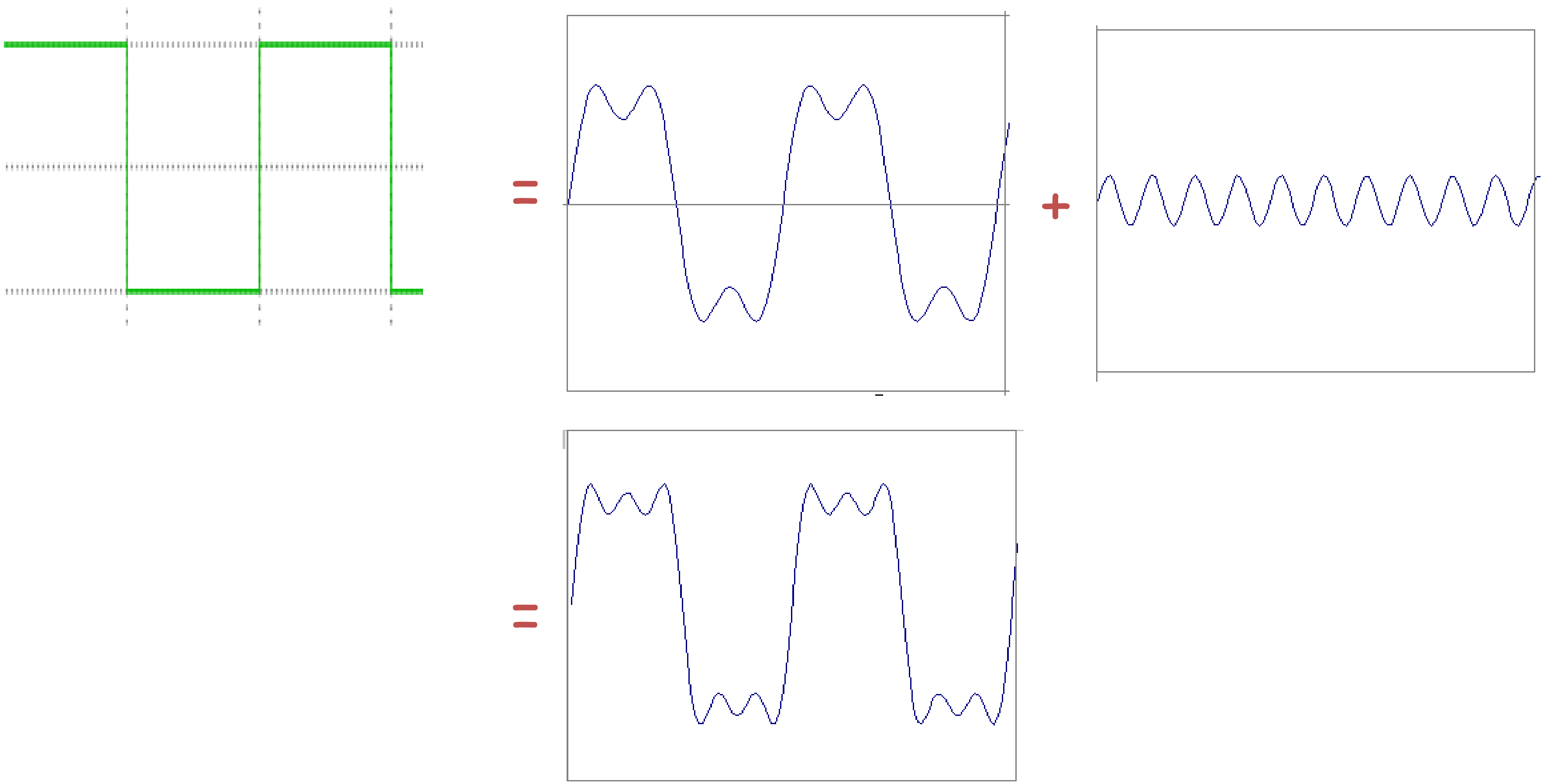
Fourier Transform

- Represent a function as a weighted sum of sines and cosines.



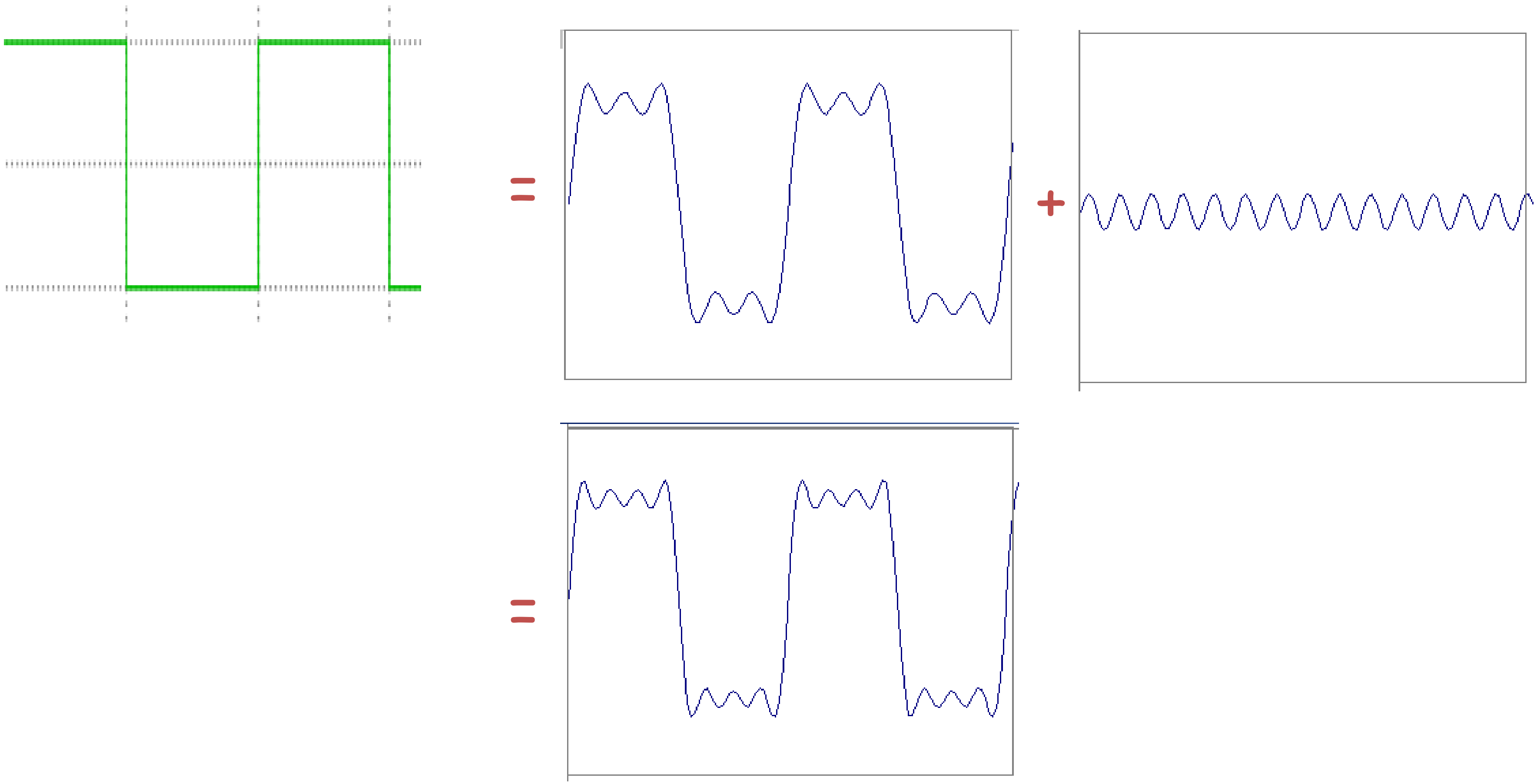
Fourier Transform

- Represent a function as a weighted sum of sines and cosines.



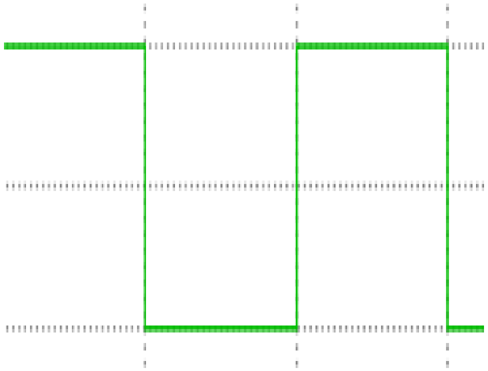
Fourier Transform

- Represent a function as a weighted sum of sines and cosines.

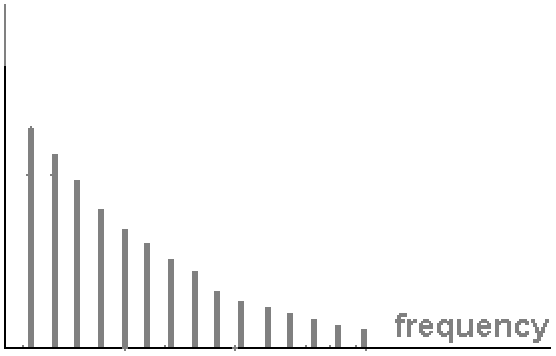


Fourier Transform

- Represent a function as a weighted sum of sines and cosines.

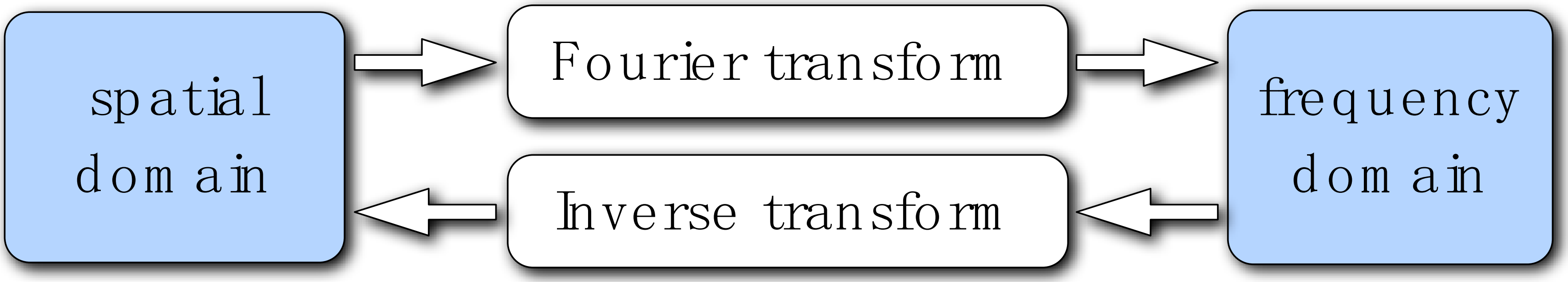


$$= \frac{A}{2} + \frac{2A\cos(t\omega)}{\pi} - \frac{2A\cos(3t\omega)}{3\pi} + \frac{2A\cos(5t\omega)}{5\pi} - \frac{2A\cos(7t\omega)}{7\pi} + \cdots$$



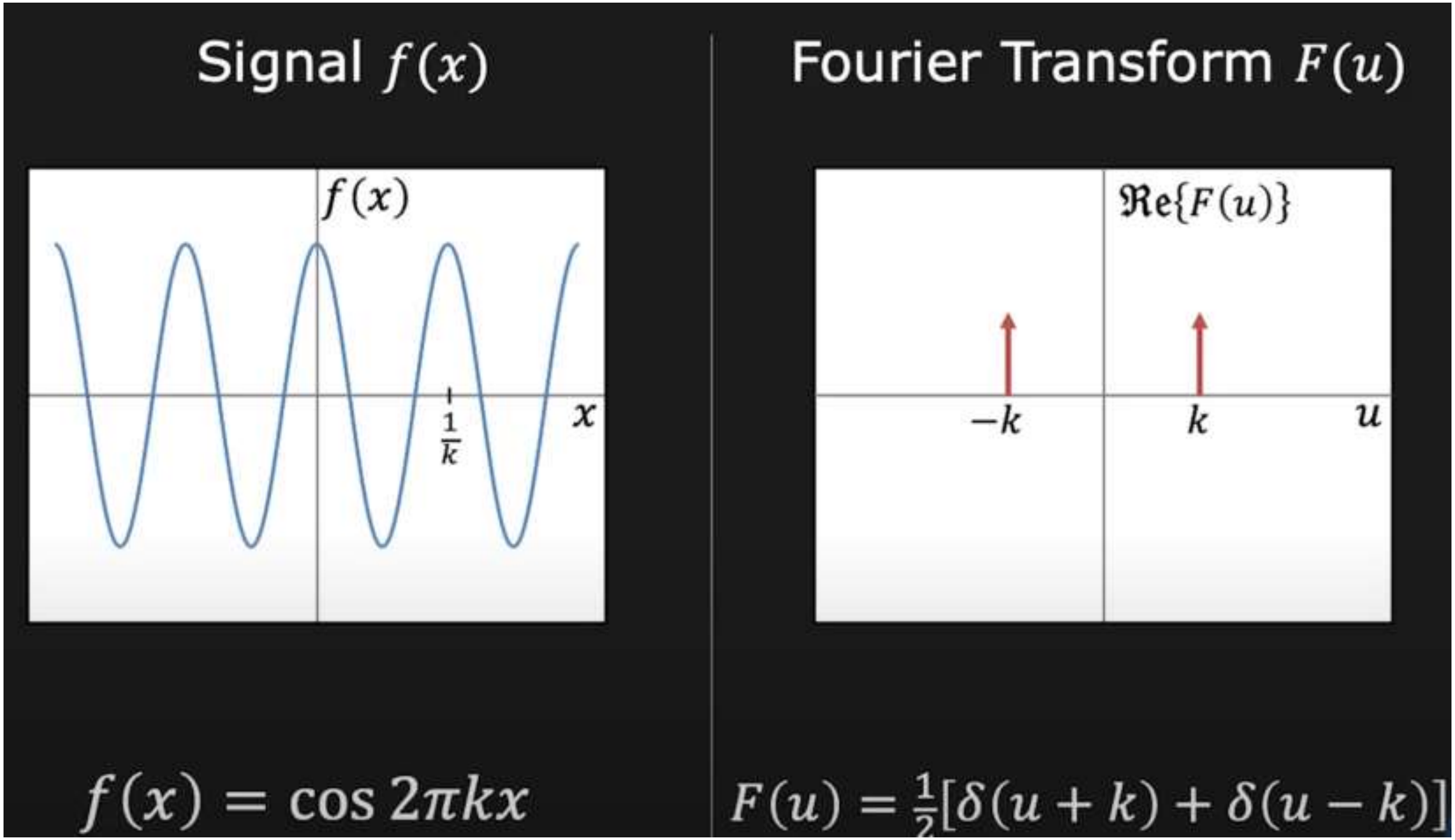
Fourier Transform

- $F(u) = \int_{-\infty}^{\infty} f(x) e^{-i2\pi ux}\, dx$
- $f(x) = \int_{-\infty}^{\infty} F(u) e^{i2\pi ux}\, du$
- x: space, u: frequency, $e^{i\theta} = cos\theta + isin\theta$, $i = \sqrt{-1}$



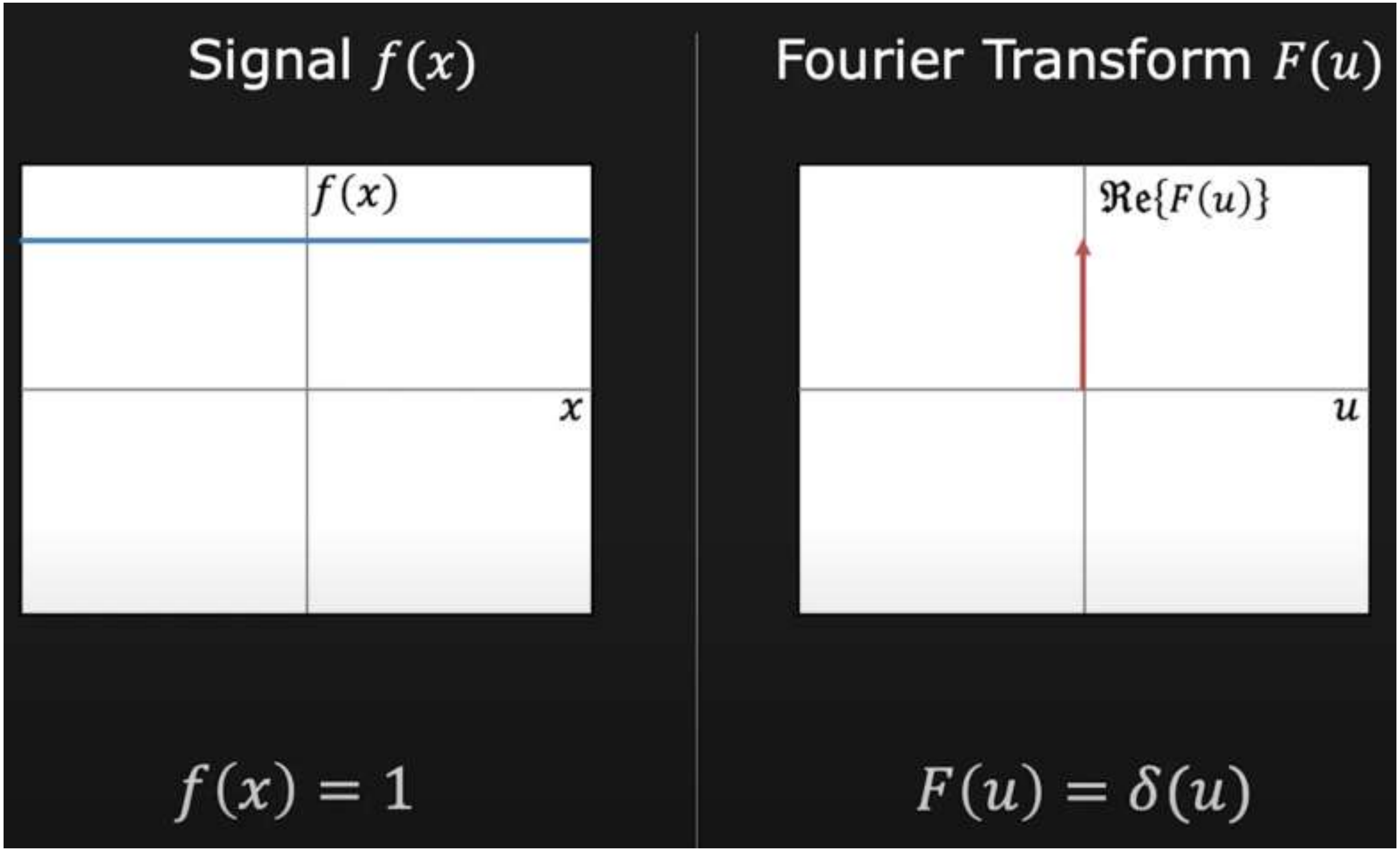
Slides borrowed from Lingqi Yan (UCSB)

Fourier Transform of sinusoids



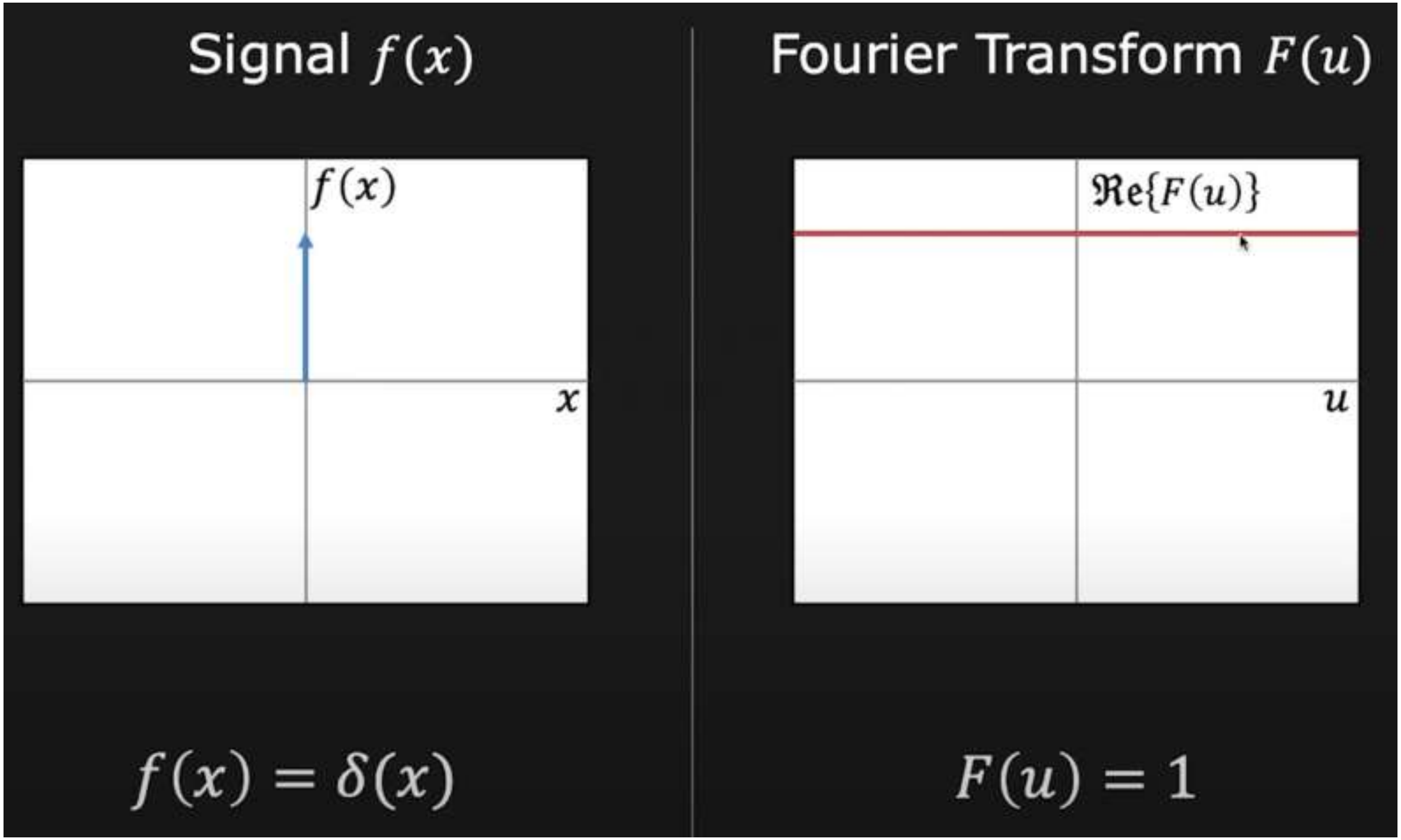
Source: fpcv.cs.columbia.edu

Fourier Transform of constant function



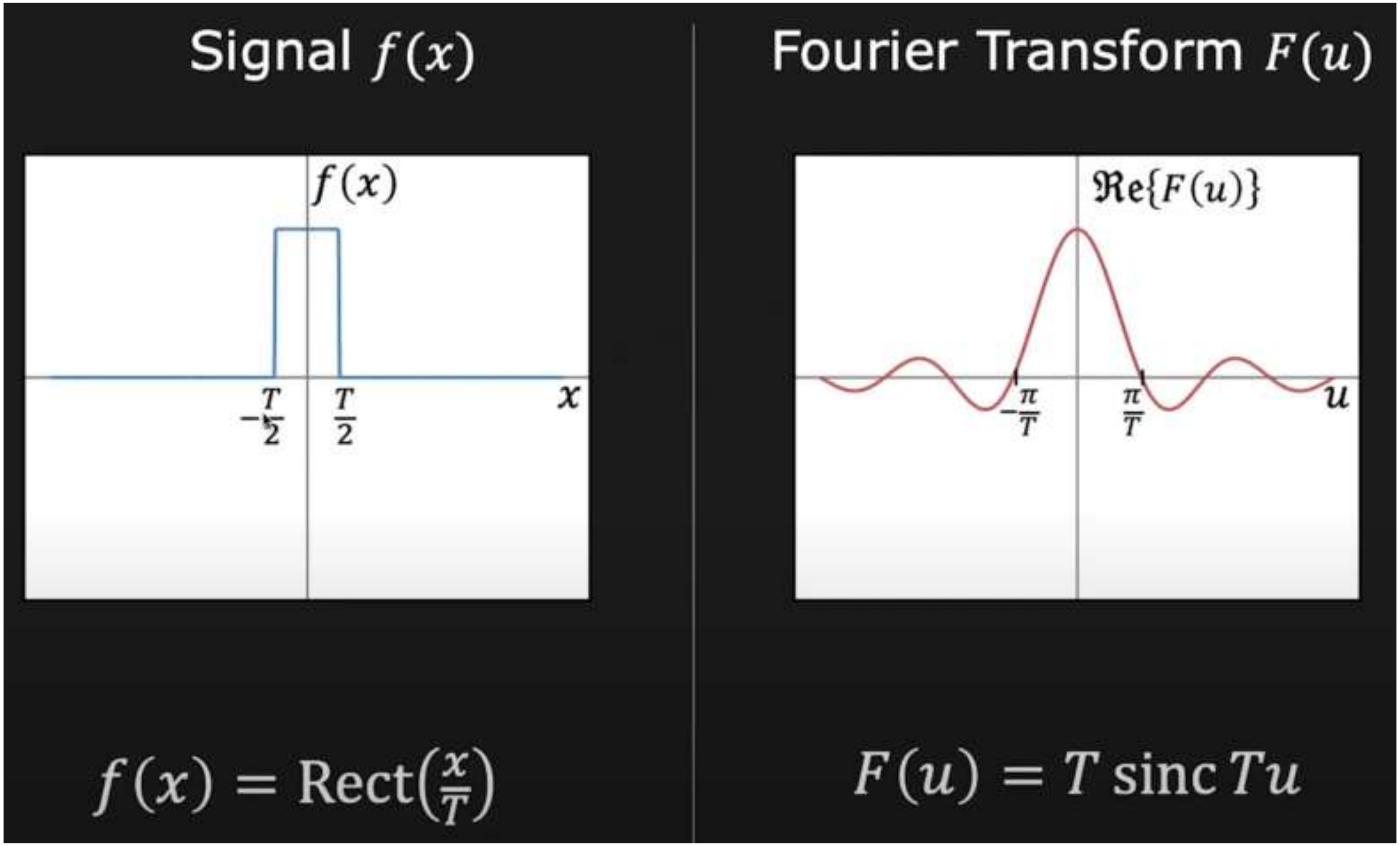
Source: fpcv.cs.columbia.edu

Fourier Transform of Dirac function



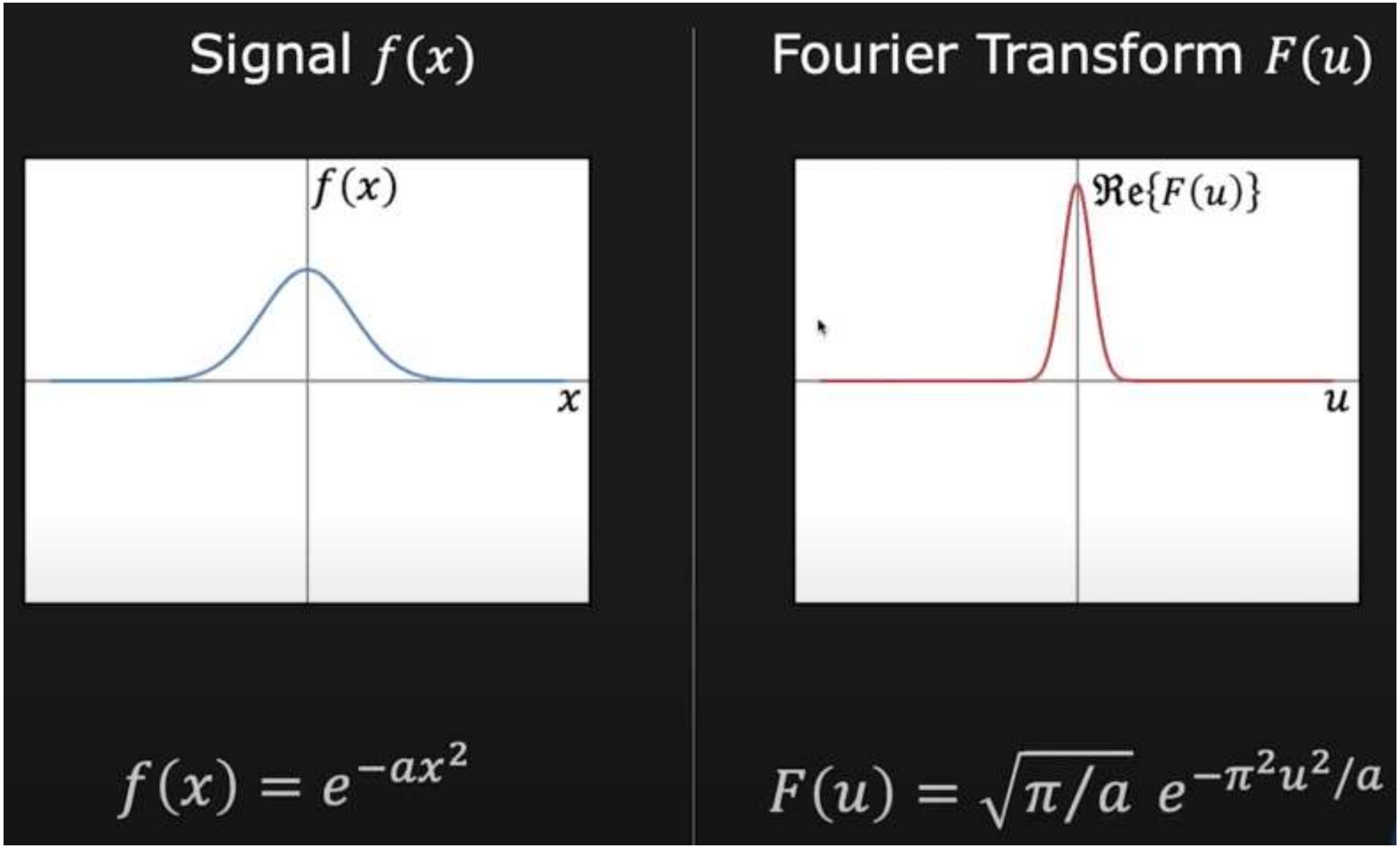
Source: fpcv.cs.columbia.edu

Fourier Transform of box function



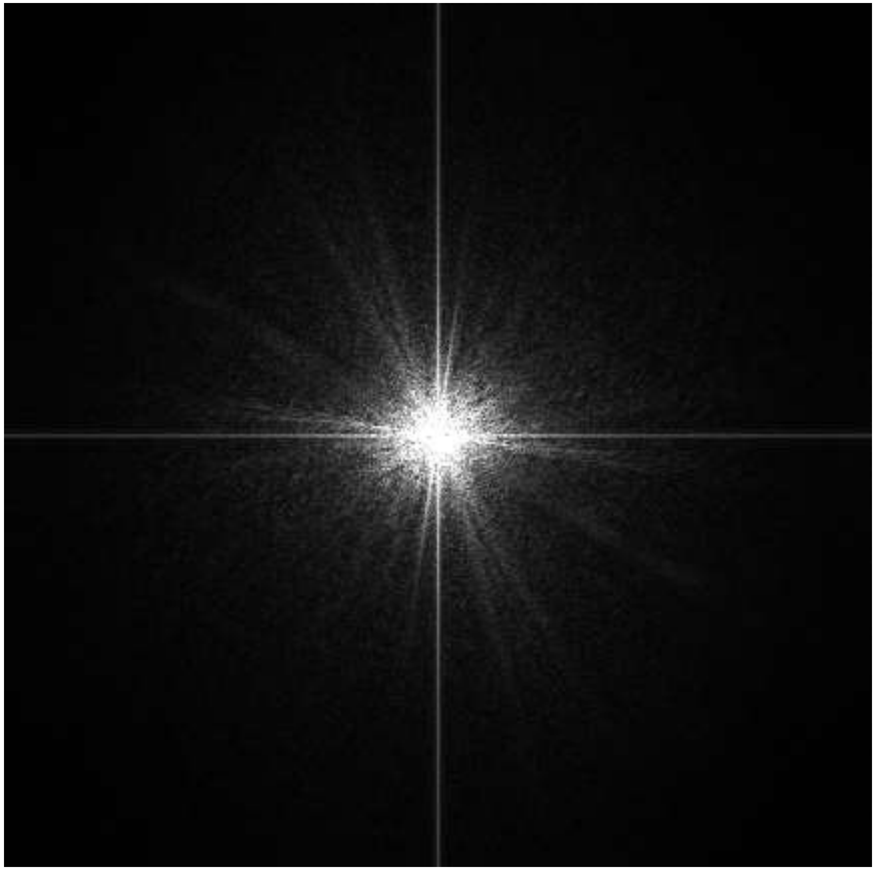
Source: fpcv.cs.columbia.edu

Fourier Transform of Gaussian function



Source: fpcv.cs.columbia.edu

Visualizing frequency content of images



Source: fpcv.cs.columbia.edu

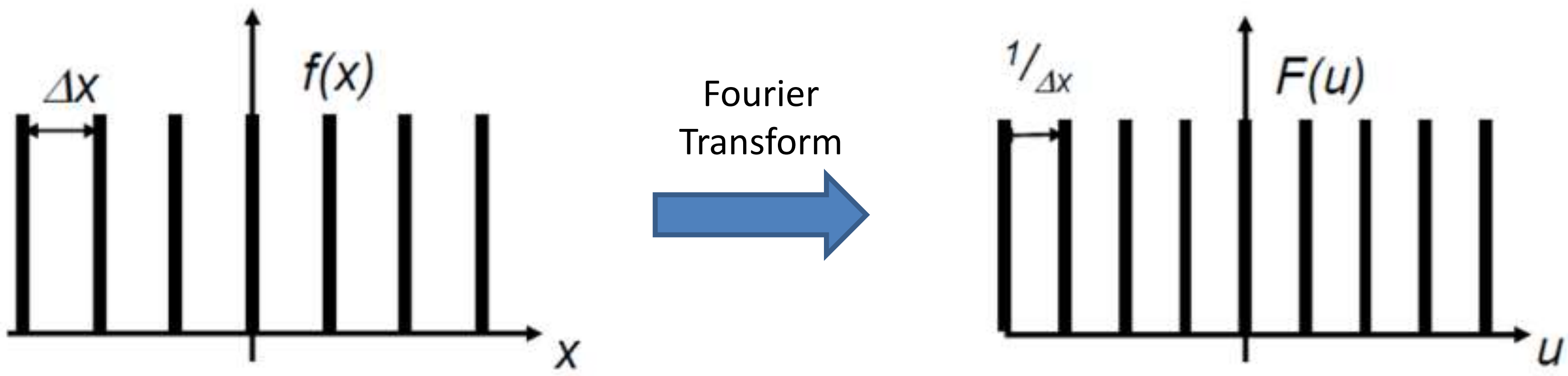
Convolution Theorem

Spatial Domain		Frequency Domain
$g(x) = f(x) * h(x)$ Convolution	$\longleftrightarrow$	$G(u) = F(u)\,H(u)$ Multiplication
$g(x) = f(x)\,h(x)$ Multiplication	$\longleftrightarrow$	$G(u) = F(u) * H(u)$ Convolution

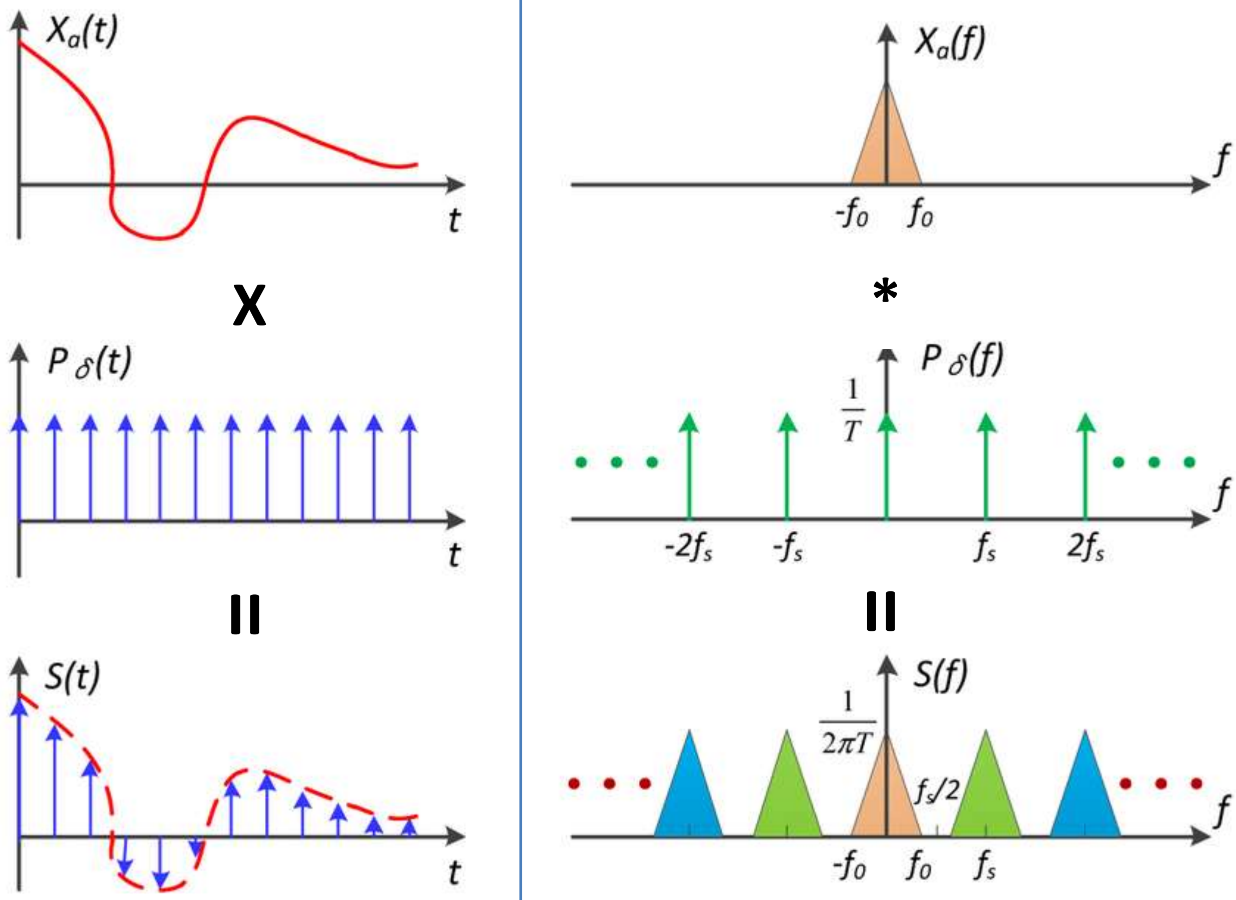
Source: fpcv.cs.columbia.edu

Sampling

- Sampling a signal = multiply the single by a Dirac comb function



Sampling = Repeating Frequency Contents



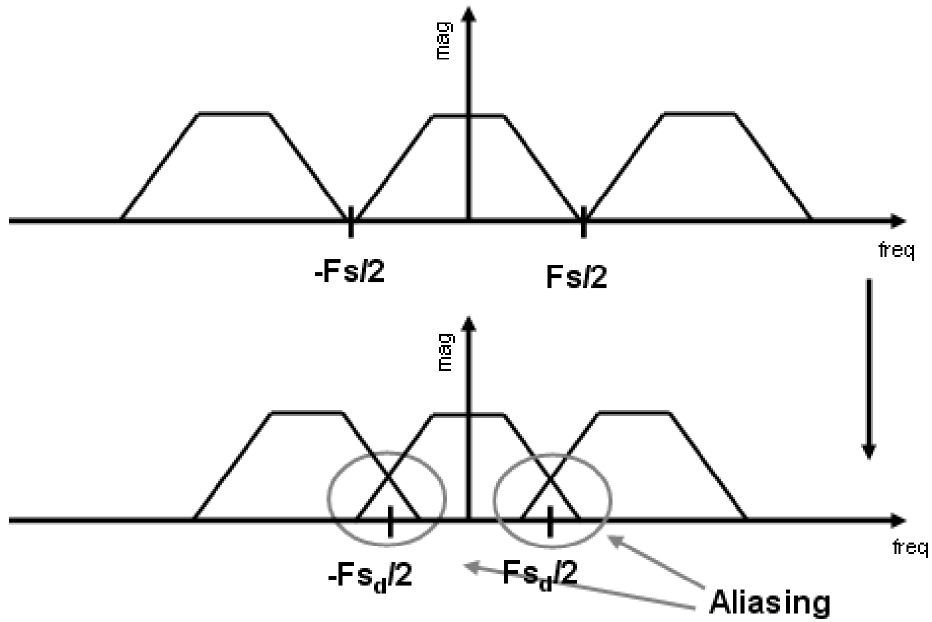
Spatial domain

Fourier domain

Aliasing = Mixed Frequency Contents

Dense sampling:

Sparse sampling:



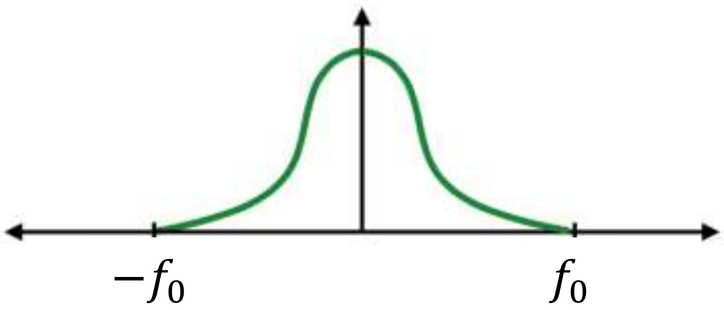
How can we reduce aliasing?

Option 1: Increasing sampling rate

How large is enough?

Nyquist-Shannon theorem

- Consider a band-limited signal: has no frequencies above f_0



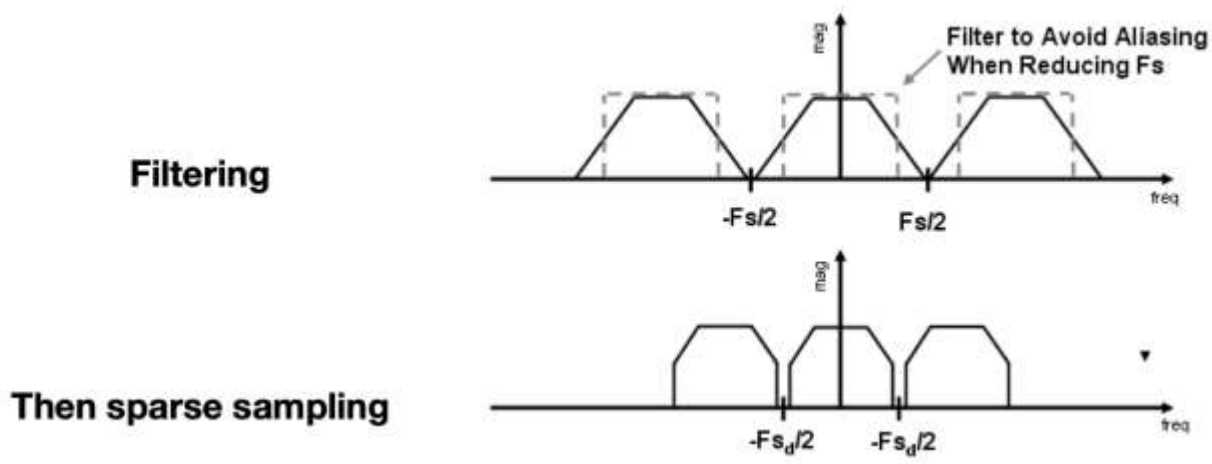
- The signal can be perfectly reconstructed if sampled with a frequency larger than $2f_0$

How can we reduce aliasing?

Option 1: Increasing sampling rate

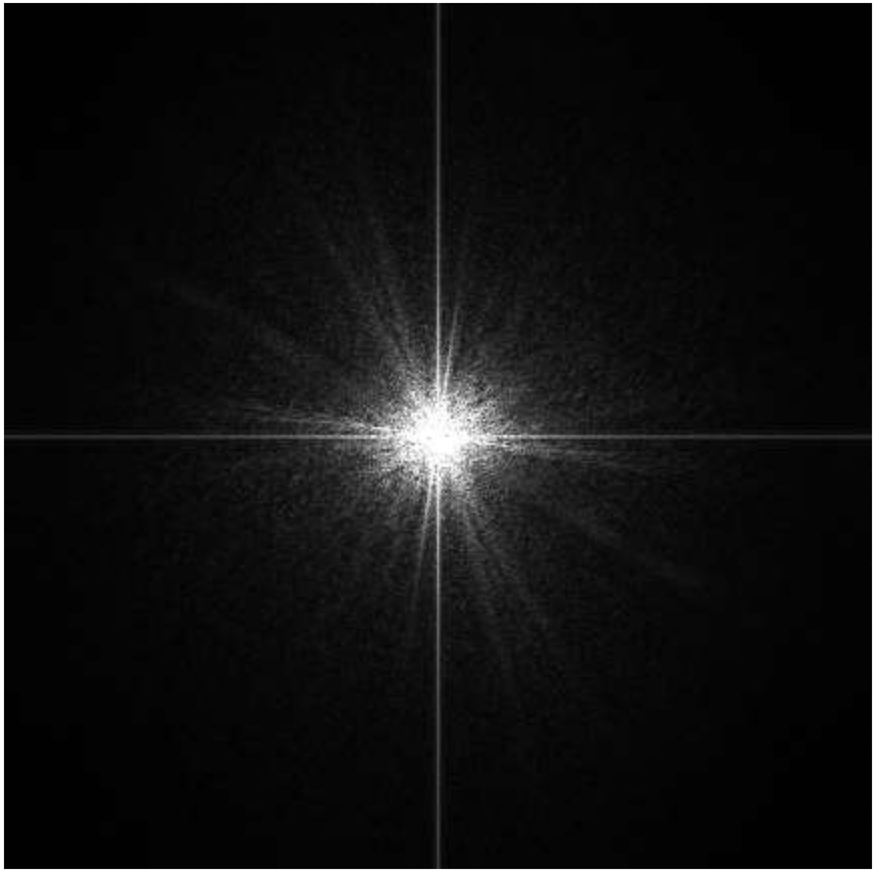
Option 2: Anti-aliasing

Filtering out high frequendcies before sampling



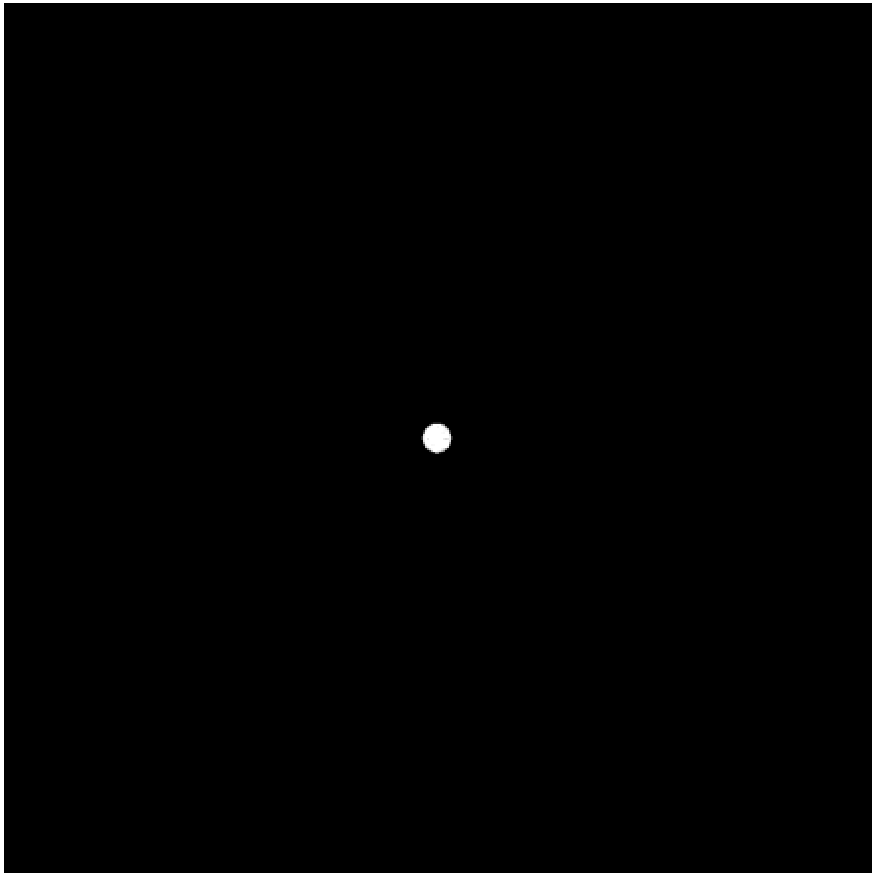
Filtering = Getting rid of certain frequency contents

Visualizing Image Frequency Content



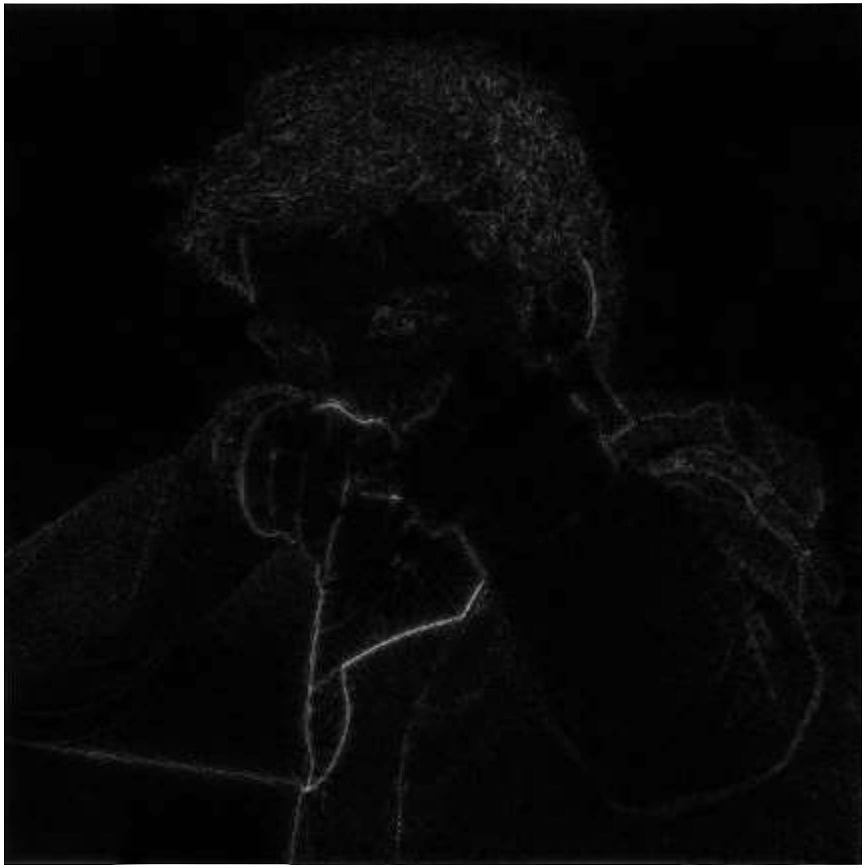
Slides borrowed from Lingqi Yan (UCSB)

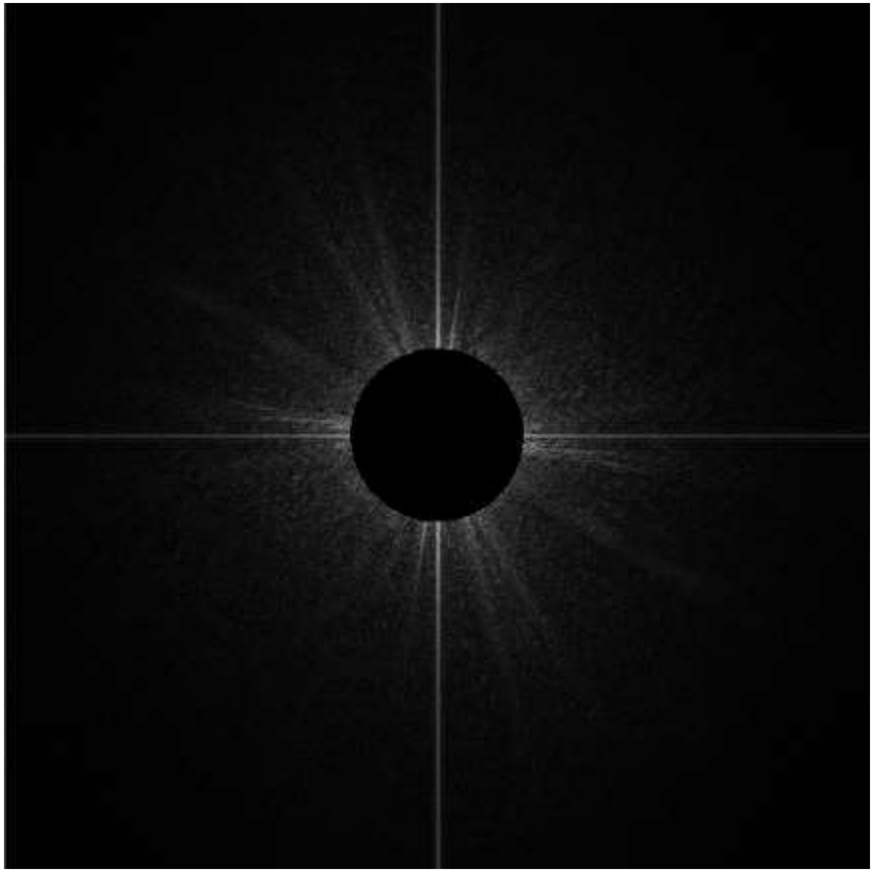
Filter Out High Frequencies (Blur)



Low-pass filter

Filter Out Low Frequencies Only (Edges)



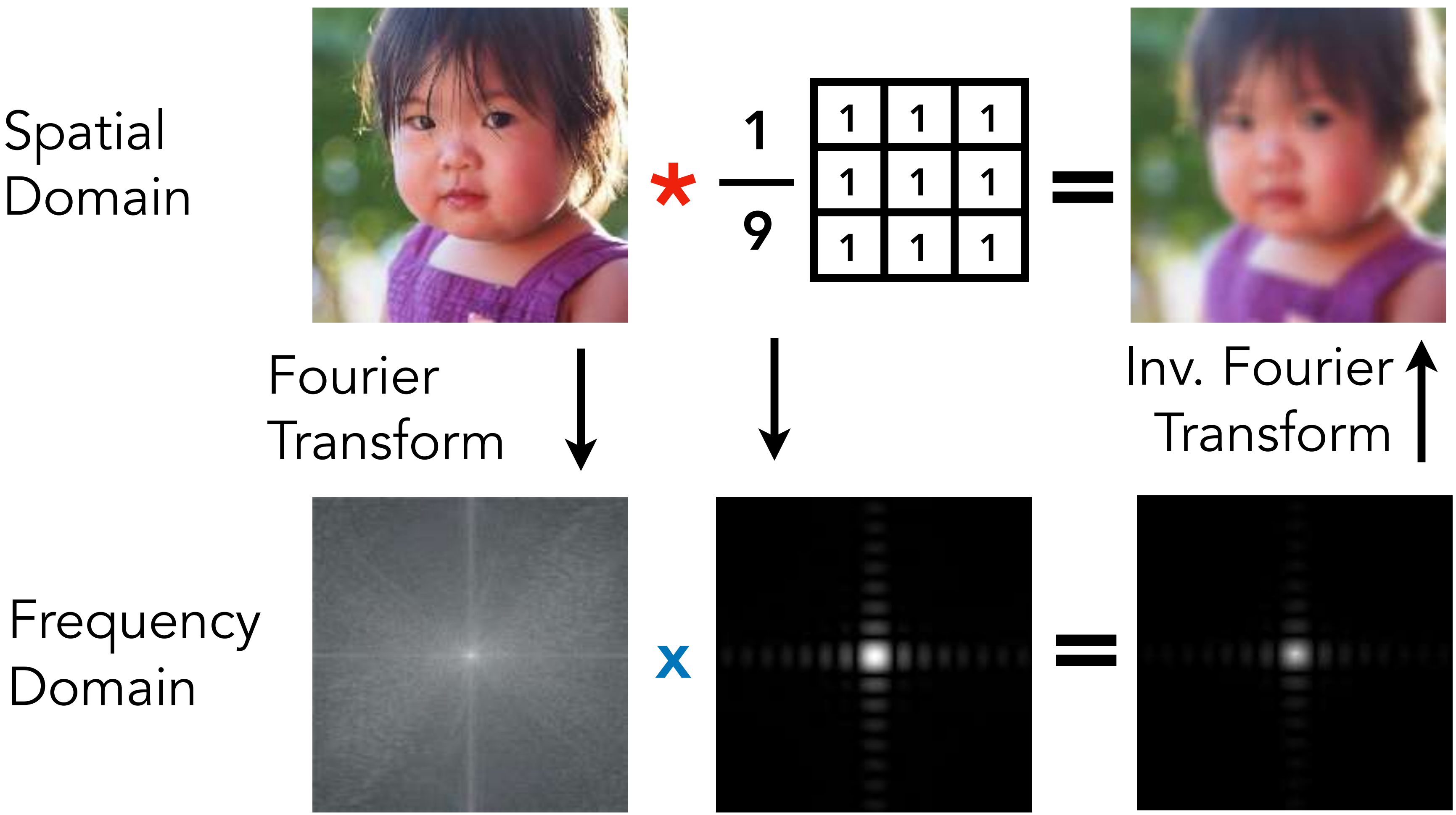


High-pass filter

Filtering = Convolution

(= Averaging)

Convolution Theorem

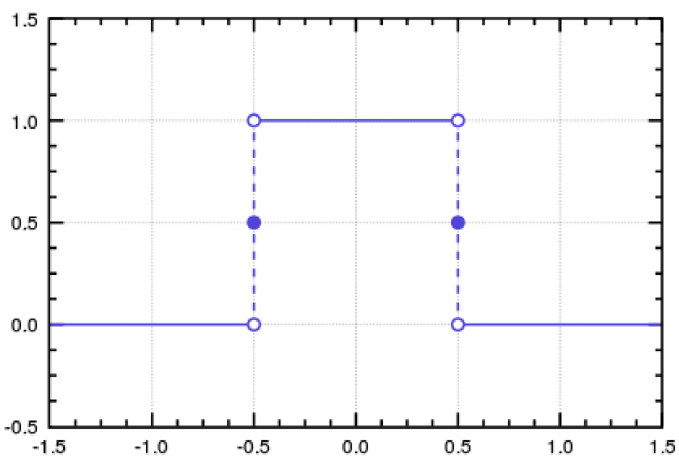


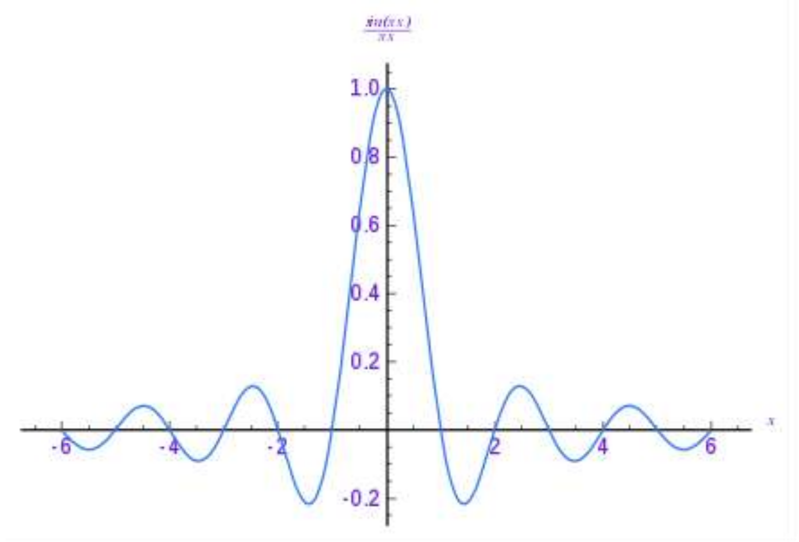
Average Filter

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Average Filter

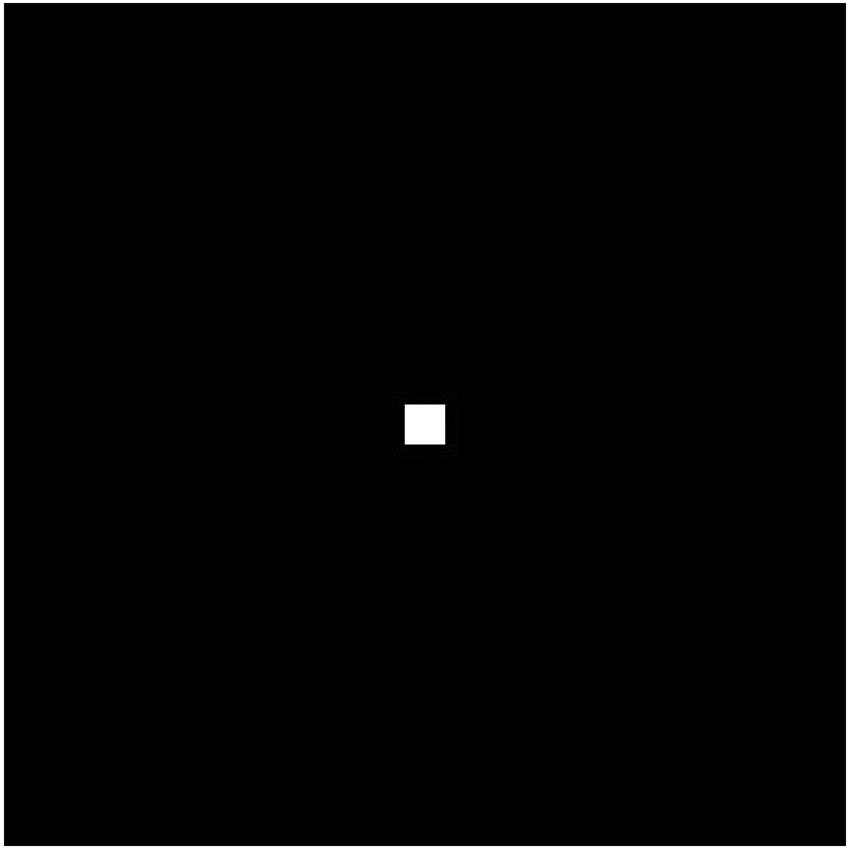
What is the Fourier transform of a rectangular function?

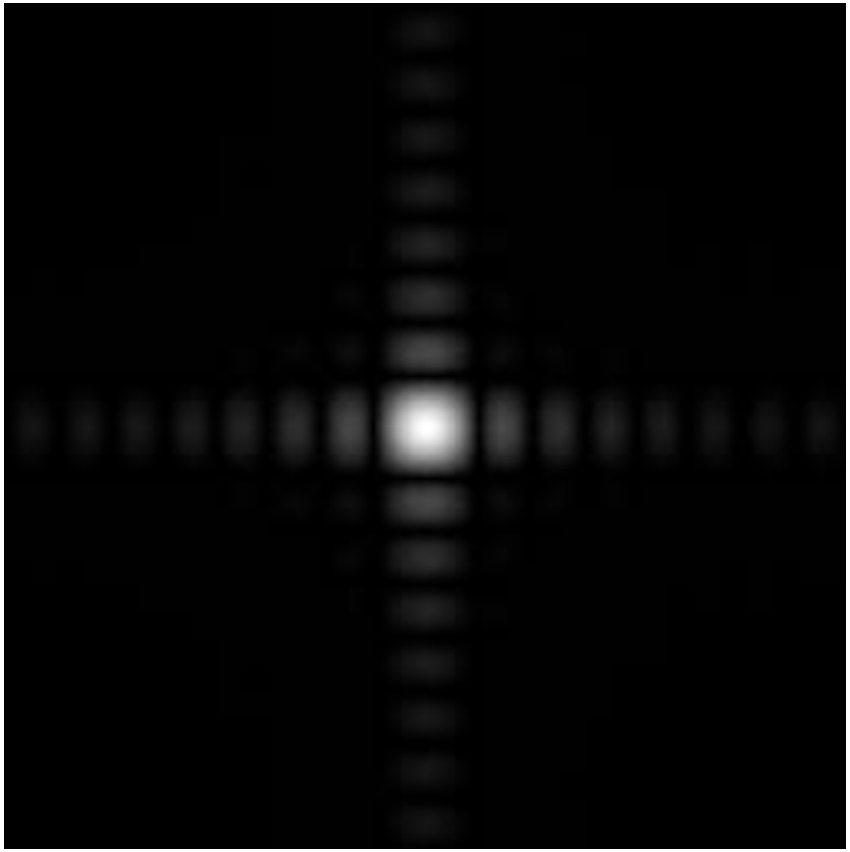




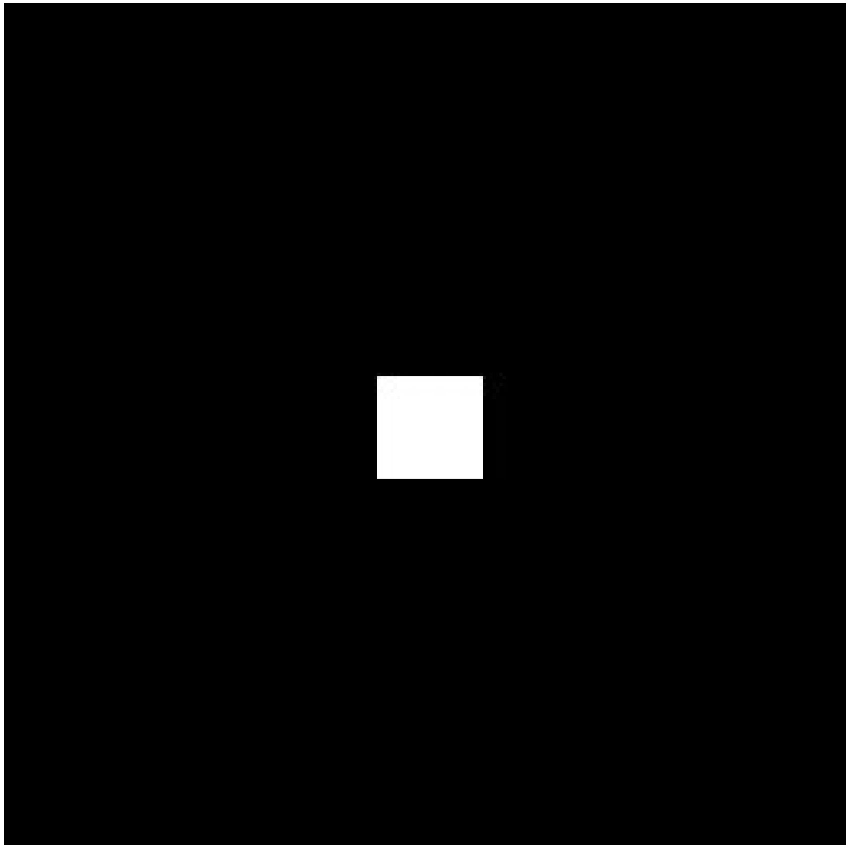
$$\mathrm{sinc}(x) = \frac{\sin(\pi x)}{\pi x}.$$

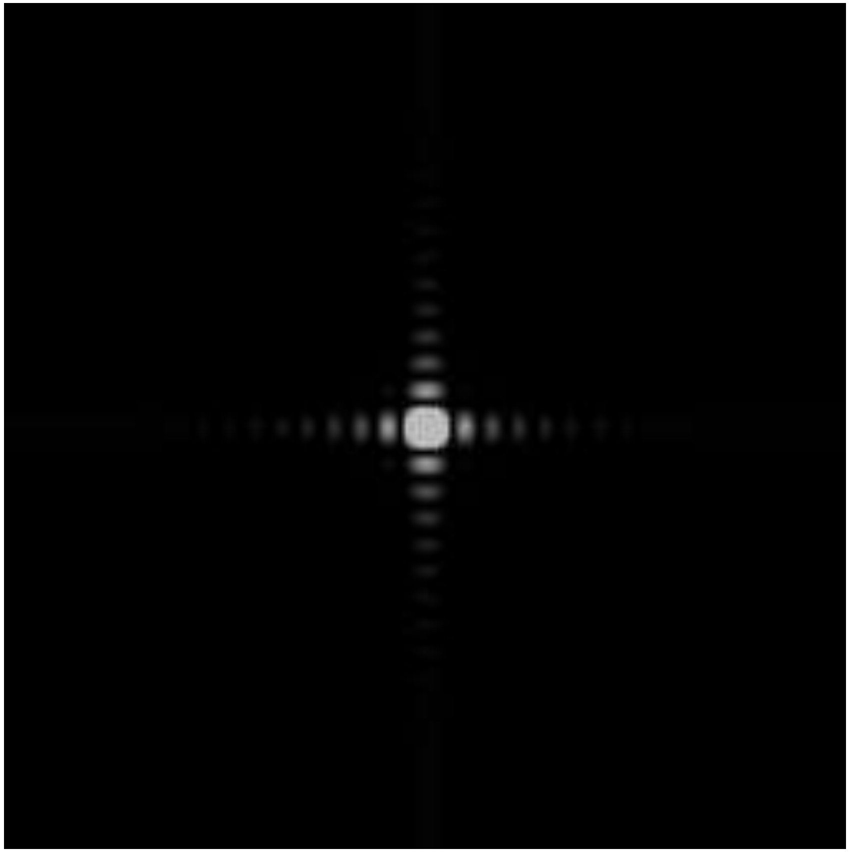
Average Filter = low-pass filter





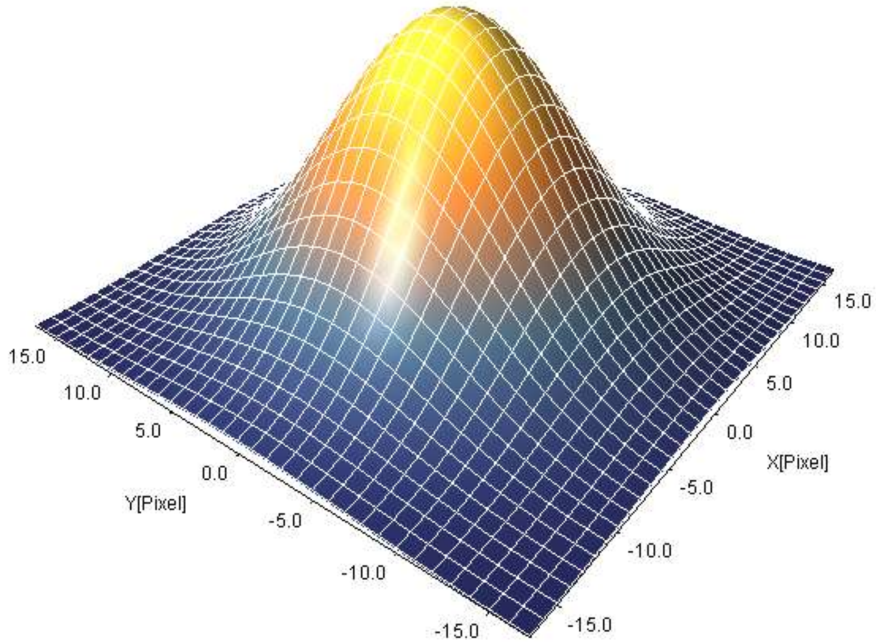
Wider kernel = lower frequency





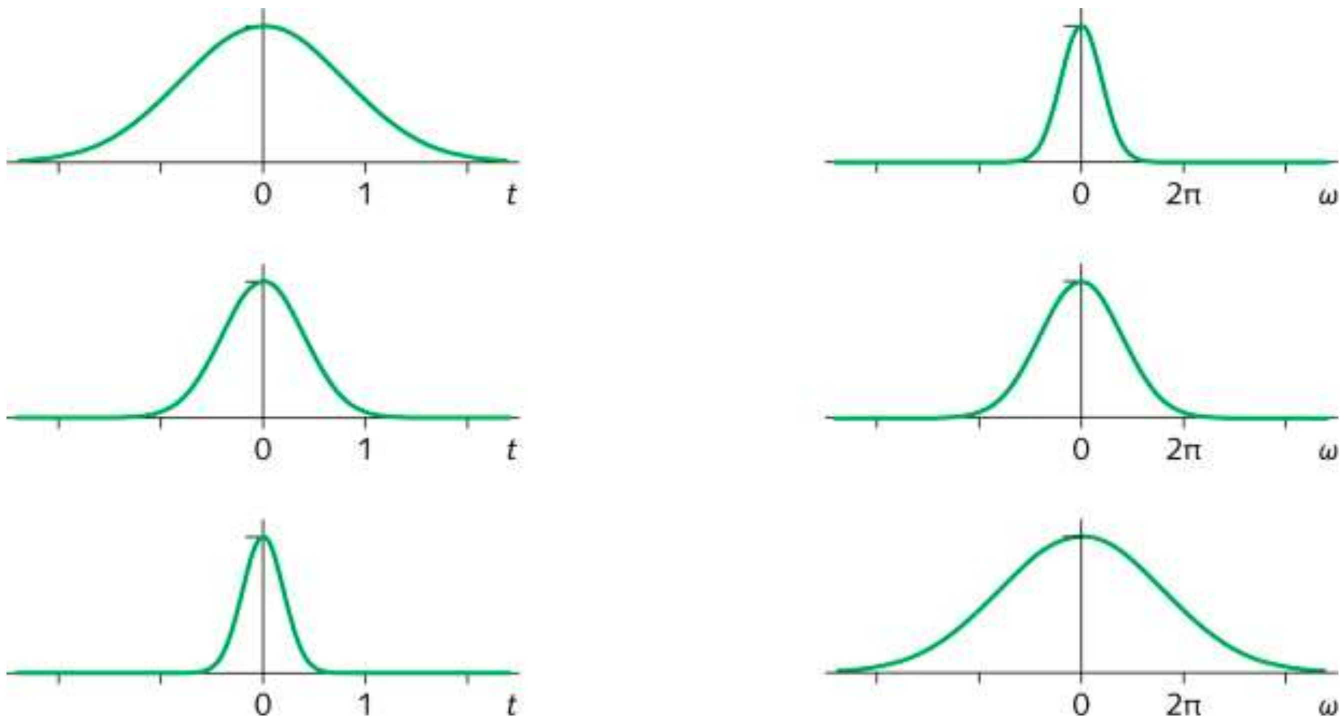
Slides borrowed from Lingqi Yan (UCSB)

Gaussian filter



$$f(x,y) = A \exp\left(-\left(\frac{(x-x_o)^2}{2\sigma_X^2} + \frac{(y-y_o)^2}{2\sigma_Y^2}\right)\right).$$

Guanssian filter

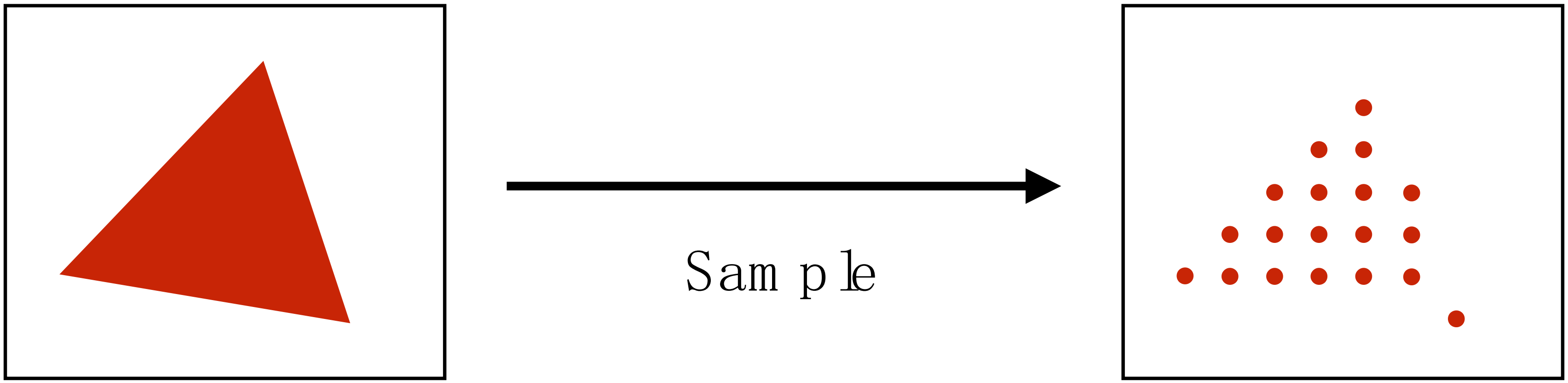


Steps for anti-alisaing

1. Convolve the image with low-pass filters (e.g. Average filter or Gaussian)

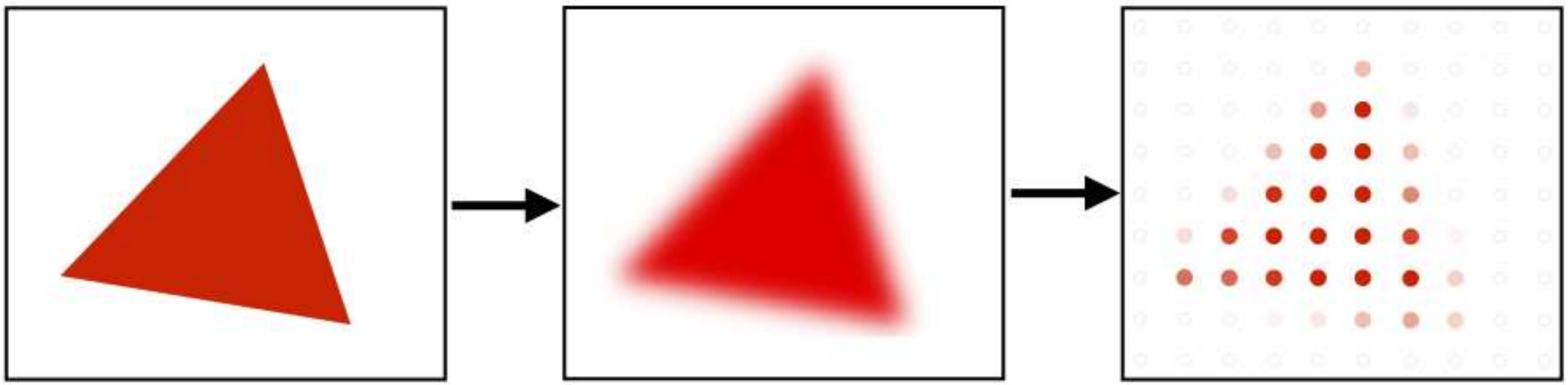
2. Sample it with a Nyquist rate

Regular Sampling



Note jaggies in rasterized triangle
where pixel values are pure red or white

Antialiased Sampling



Pre-Filter

(remove frequencies above Nyquist)

Sample

Note antialiased edges in rasterized triangle where pixel values take intermediate values

Antialiasing

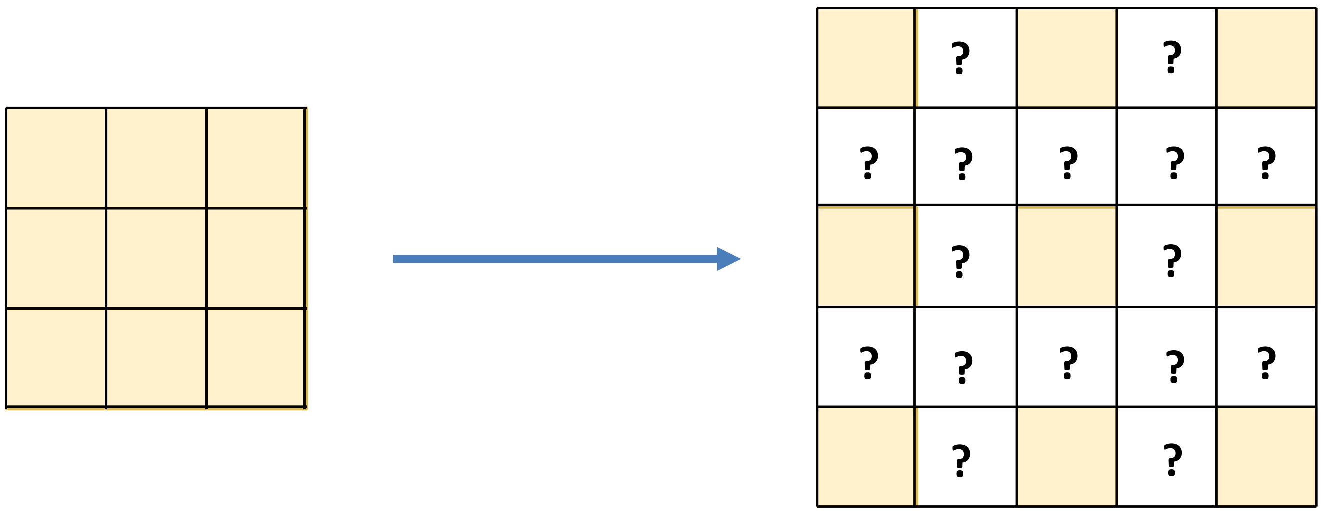
Today

- Image processing basics.
- Image sampling.
- **Image magnification.**

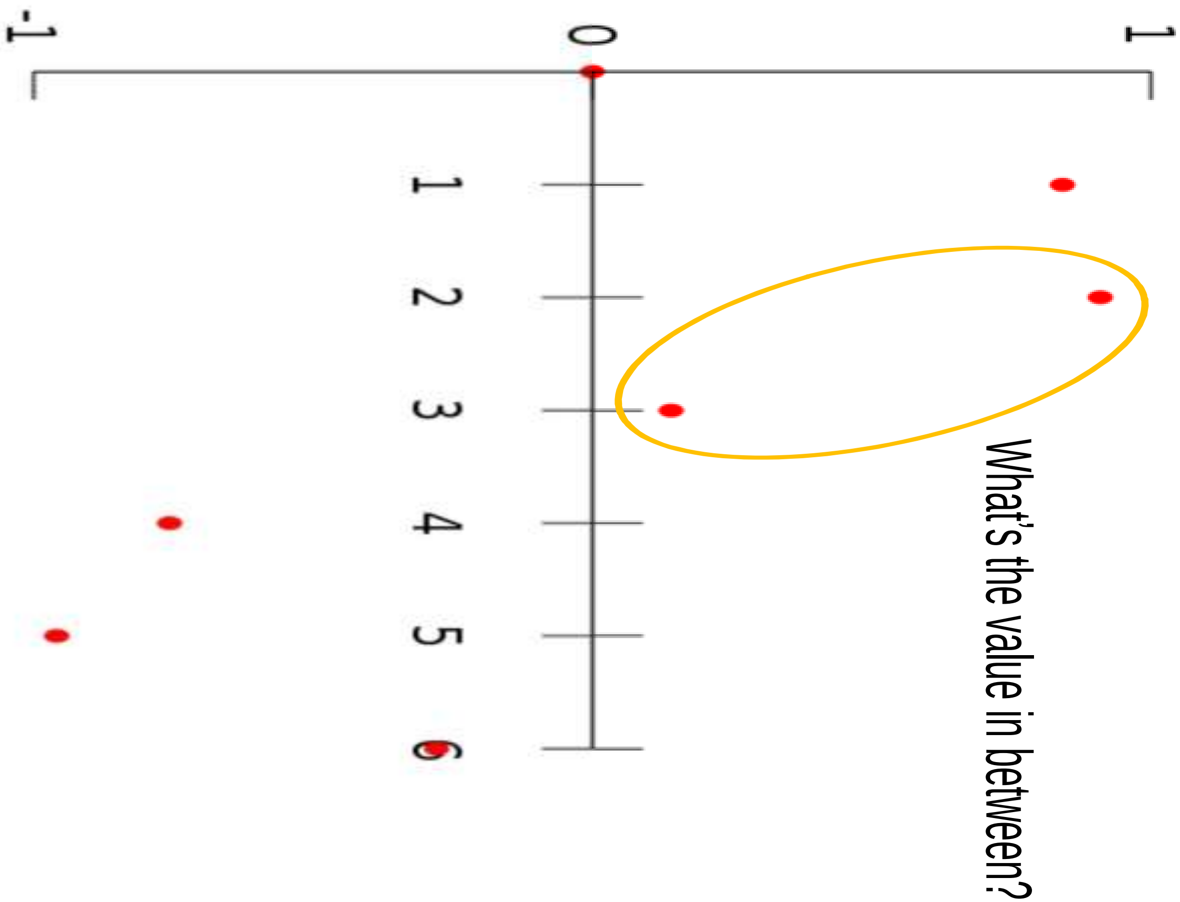
Image magnification

Image magnification

Inverse of down-sampling (up-sampling)



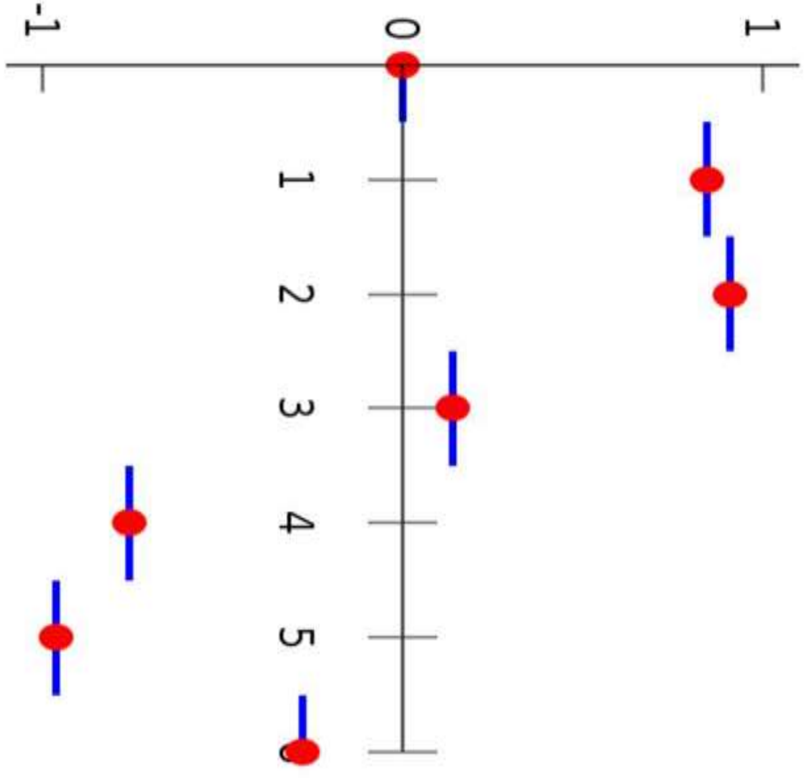
Interpolation



Nearest-neighbor interpolation

Not continuous

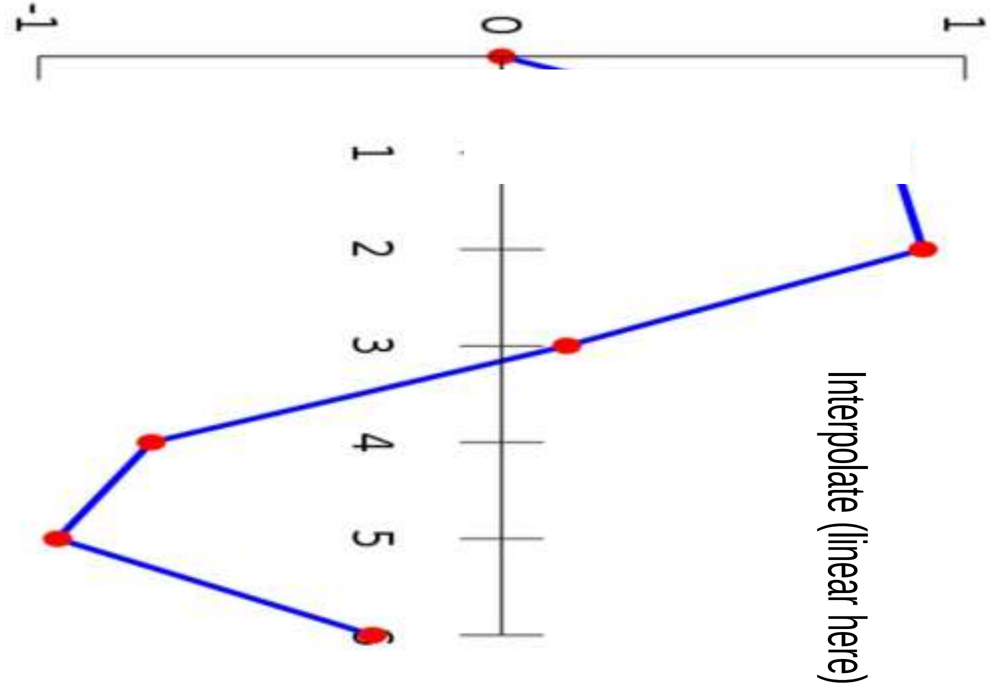
Not smooth



Linear interpolation

Continous

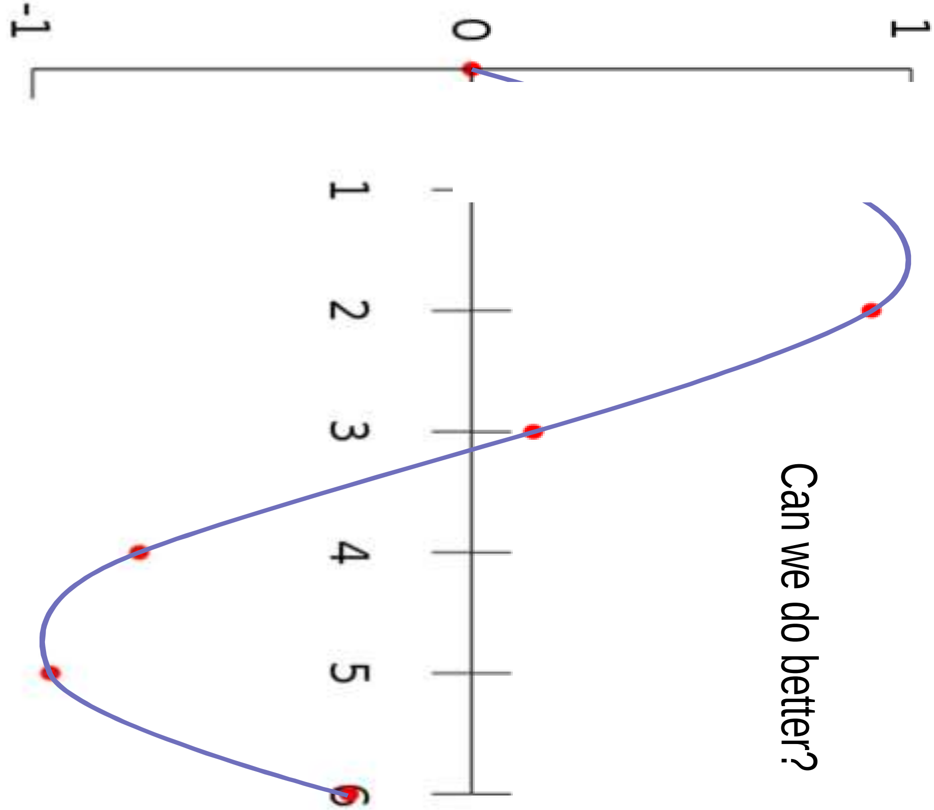
Not smooth



Cubic interpolation

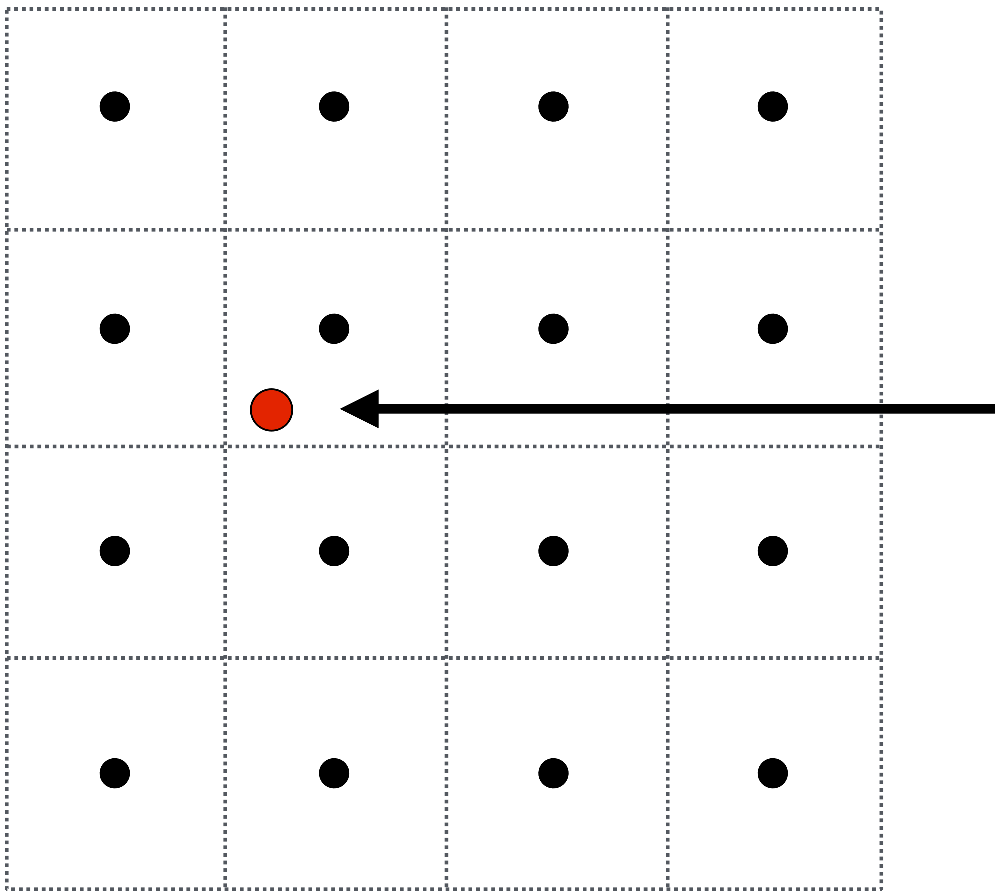
Continous

Smooth



For each interval: $y = ax^3 + bx^2 + cx + d$

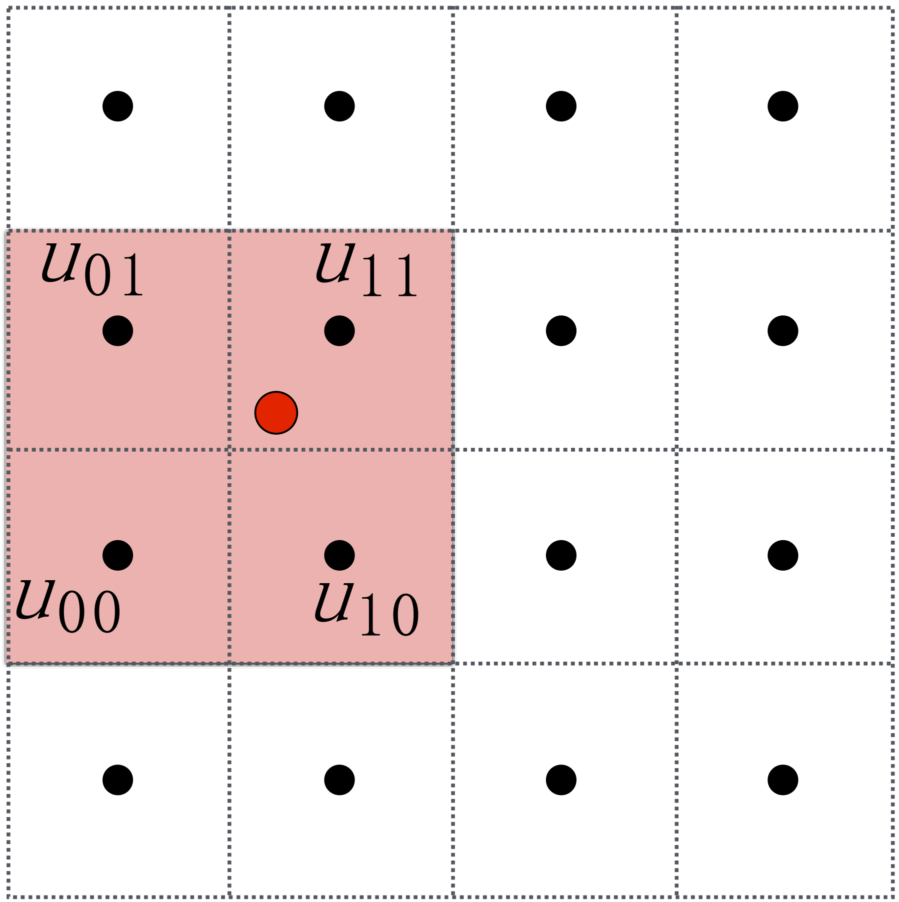
Bilinear Interpolation



Want to sample texture value f(x,y) at red point

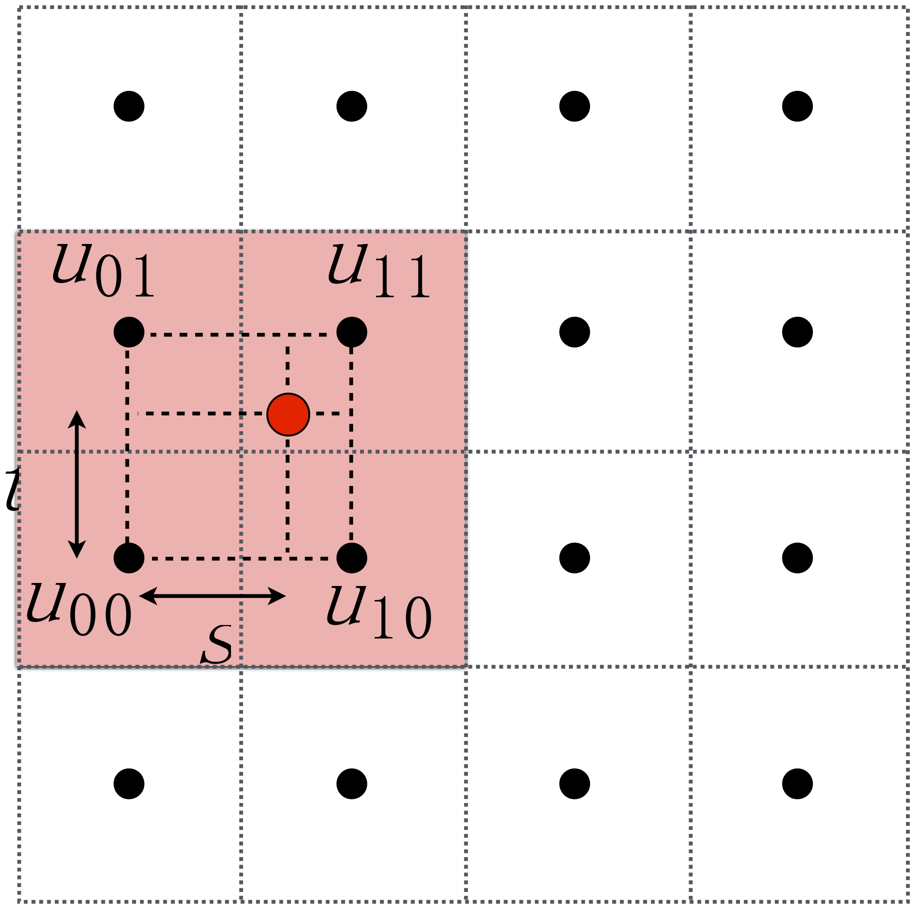
Black points indicate texture sample locations

Bilinear Interpolation



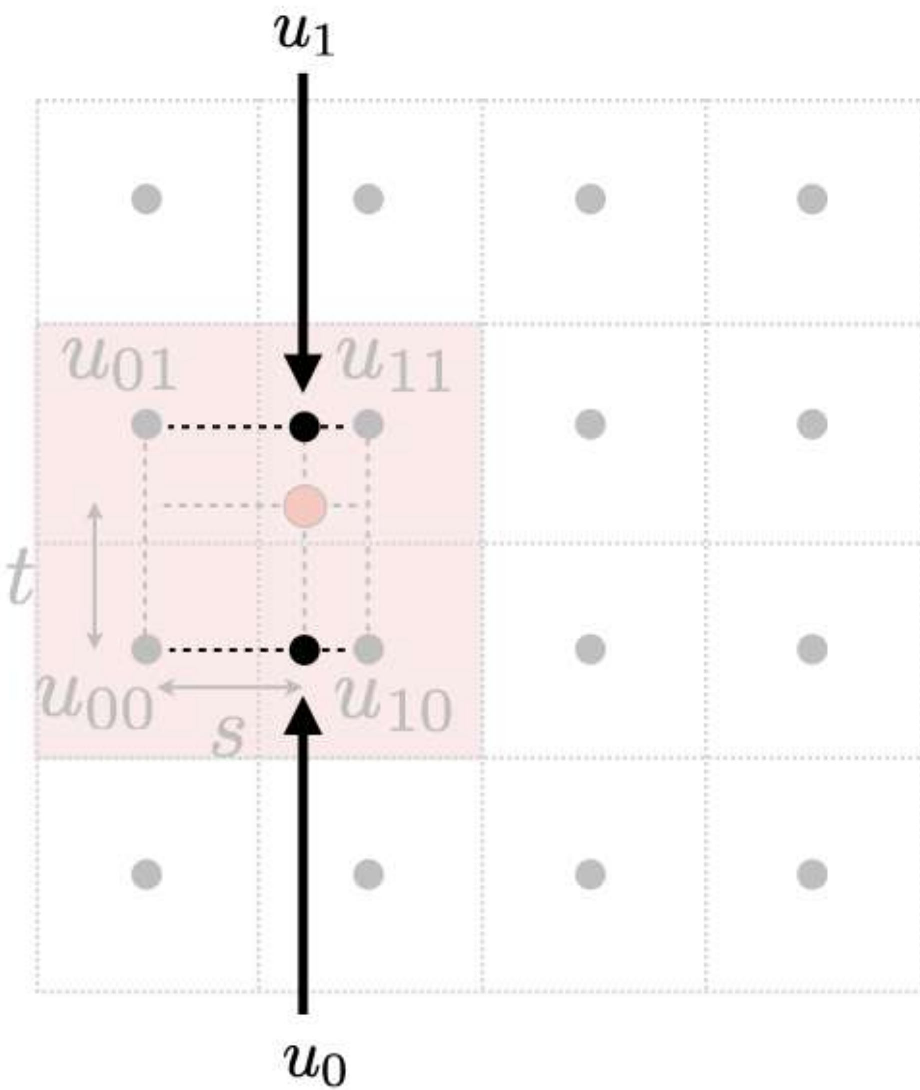
Take 4 nearest sample locations, with texture values as labeled.

Bilinear Interpolation



And fractional offsets, (s,t) as shown

Bilinear Interpolation



Linear interpolation (1D)

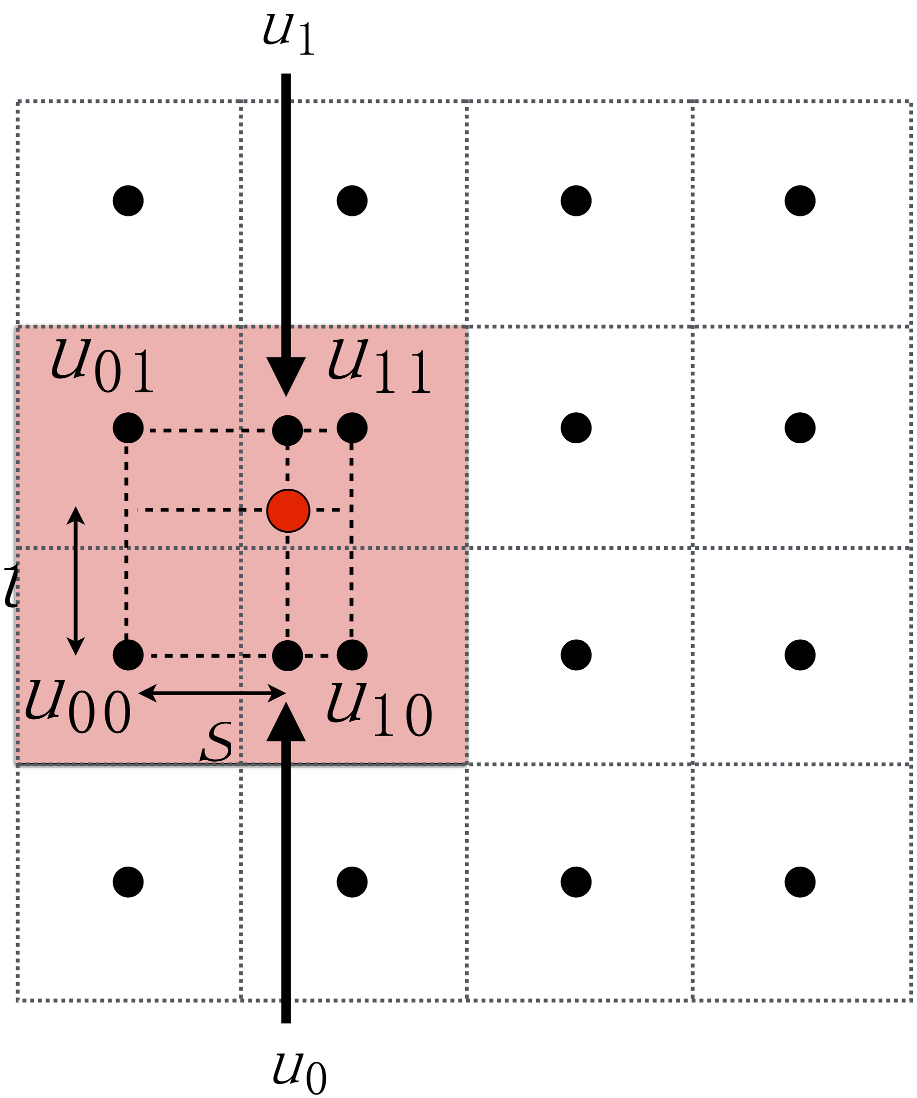
$$\text{lerp}(x, v_0, v_1) = v_0 + x(v_1 - v_0)$$

Two helper lerps (horizontal)

$$u_0 = \text{lerp}(s, u_{00}, u_{10})$$

$$u_1 = \text{lerp}(s, u_{01}, u_{11})$$

Bilinear Interpolation



Linear interpolation (1D)

$$\text{lerp}(x, v_0, v_1) = v_0 + x(v_1 - v_0)$$

Two helper lerps

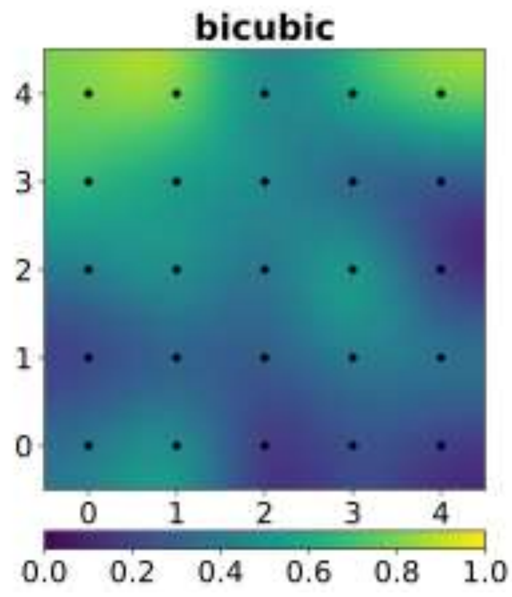
$$u_0 = \text{lerp}(s, u_{00}, u_{10})$$

$$u_1 = \text{lerp}(s, u_{01}, u_{11})$$

Final vertical lerp, to get result:

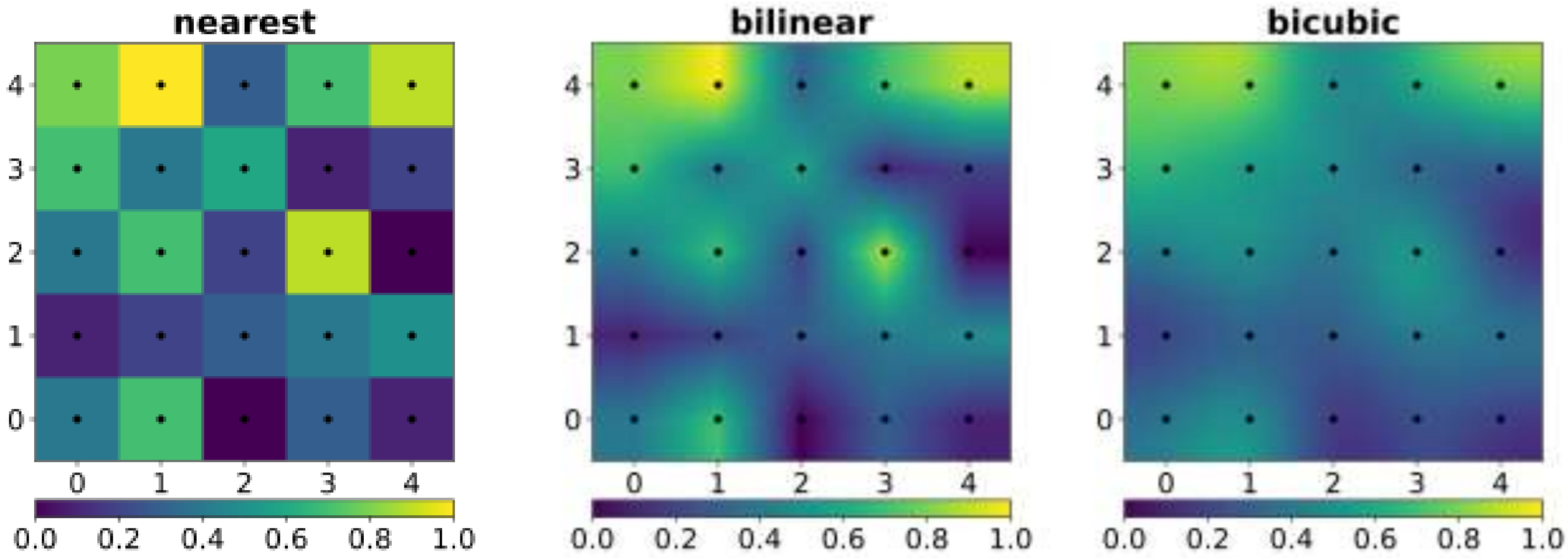
$$f(x, y) = \text{lerp}(t, u_0, u_1)$$

Bicubic Interpolation



$$p(x,y)=\sum_{i=0}^{3}\sum_{j=0}^{3}a_{ij}x^iy^j.$$

Comparison



Comparison

Generally bilinear is good enough

Nearest

Bilinear

Bicubic

Super-Resolution

Original

Bi-Cubic

Super-Resolution

How to change aspect ratio?

Challenge

Changing aspect ratio causes distortion

Cropping may remove important contents

Seam Carving for Content-Aware Image Resizing

Shai Avidan
Mitsubishi Electric Research Labs

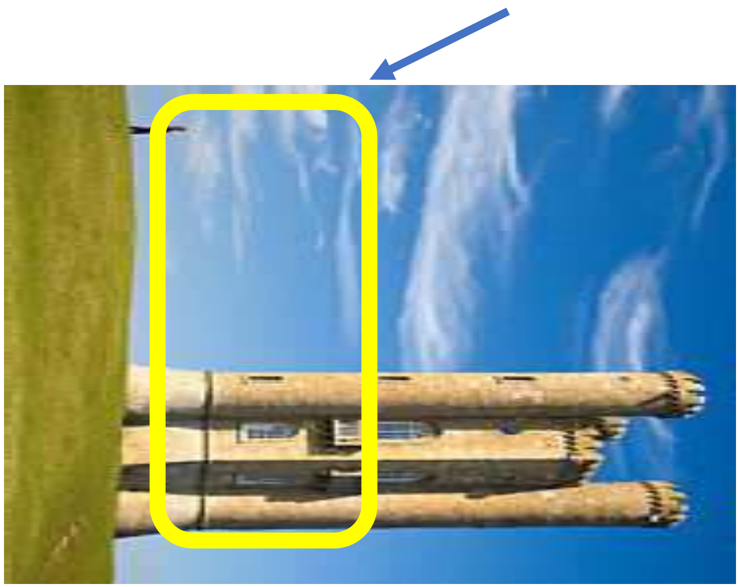
Ariel Shamir
The Interdisciplinary Center & MERL

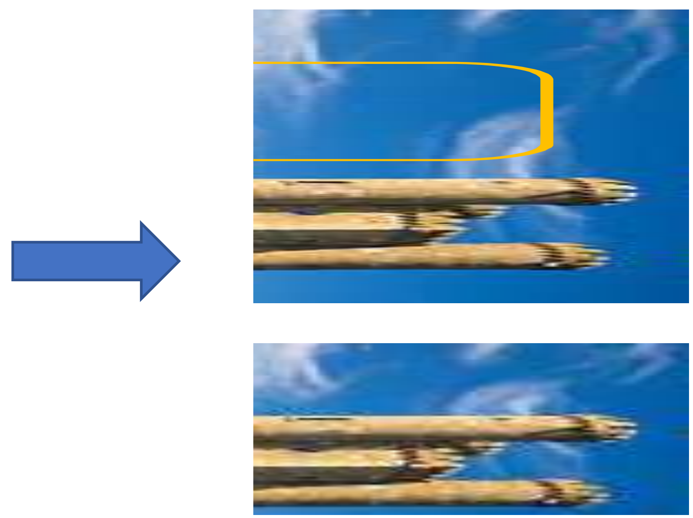
Content-aware resizing

Basic idea

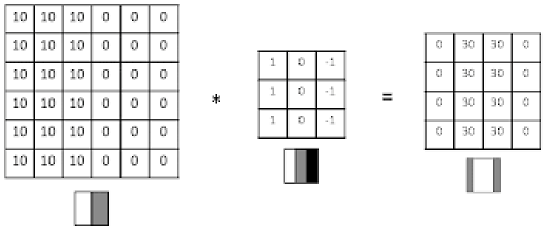
Problem statement: we need to remove n pixels from each row

Basic idea: remove unimportant pixels



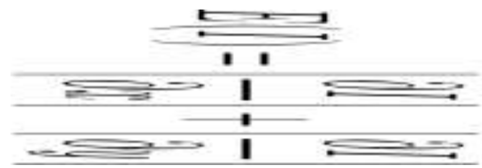


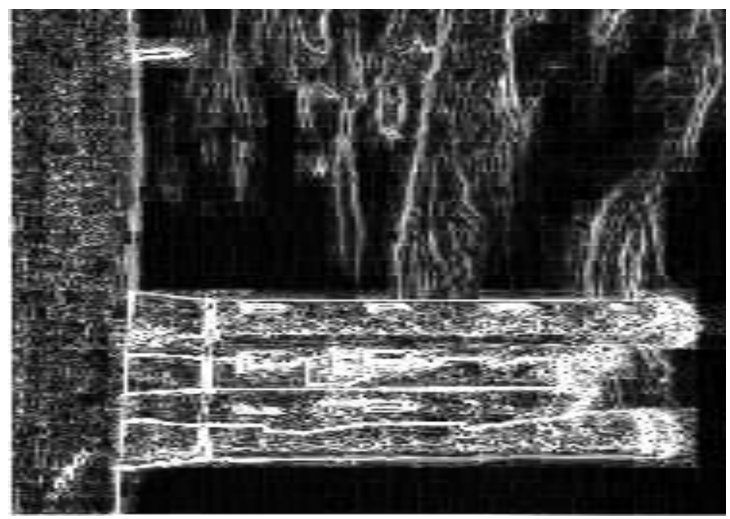
Importance of pixel



How to measure importance of a pixel?

- A simple idea – edges are important
- Edge energy:





Greedy algorithm

Remove pixels or columns with the smallest energy?

Least-energy pixels

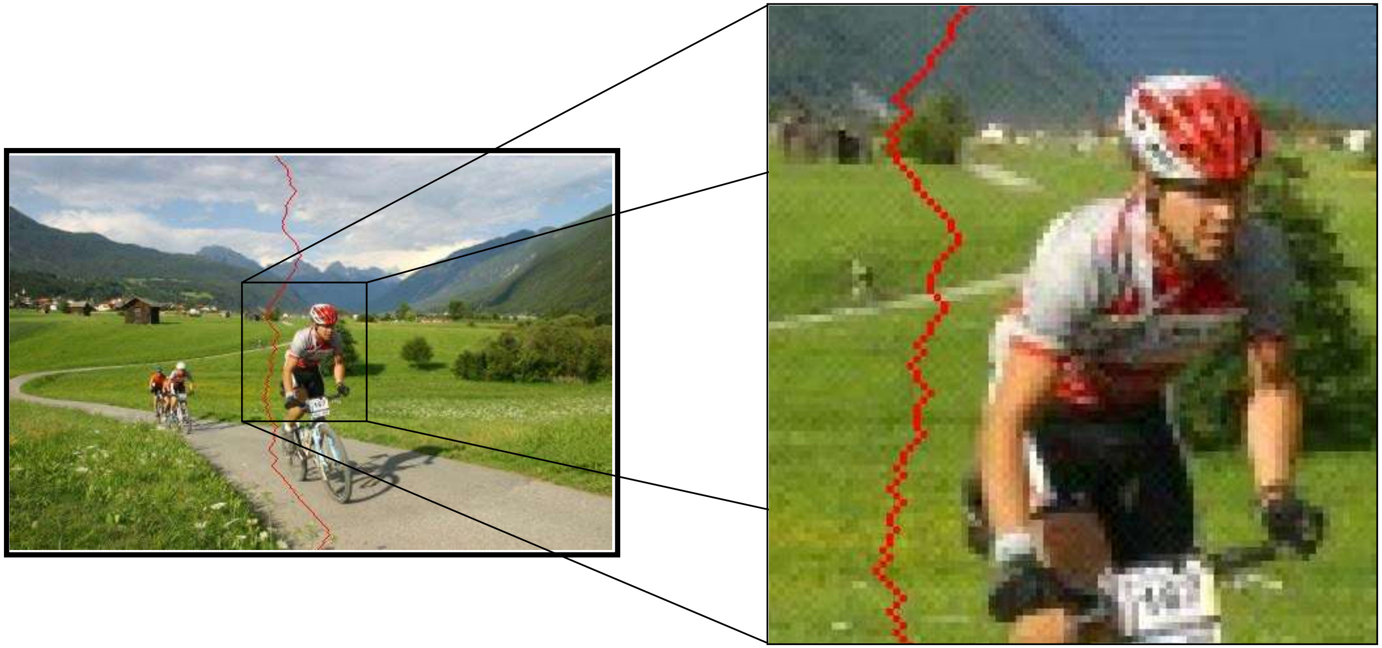
Greedy algorithm

Remove pixels or columns with the smallest energy?

Least-energy columns

Seam carving

Find connected path of pixels from top to bottom
of which the edge energy is minimal



Finding the seam?

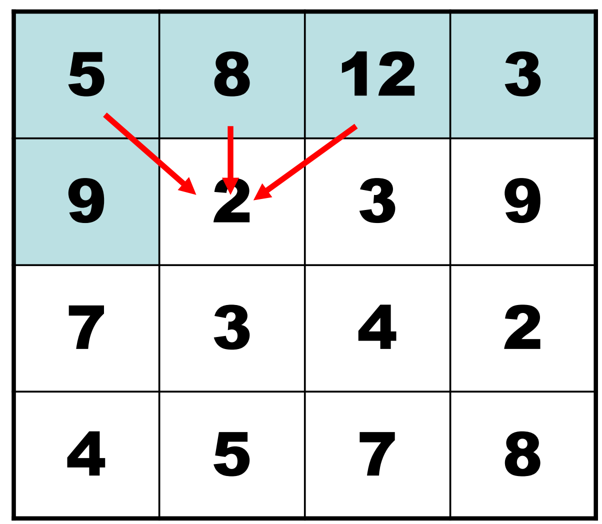
Finding the seam?

Going from top to bottom

- If M(i,j) = minimal energy of a seam going through (i,j)
- Then:

$$\mathbf{M}(i,j) = E(i,j) + \min\big(\mathbf{M}(i-1,j-1), \mathbf{M}(i-1,j), \mathbf{M}(i-1,j+1)\big)$$

- Solved by dynamic programing



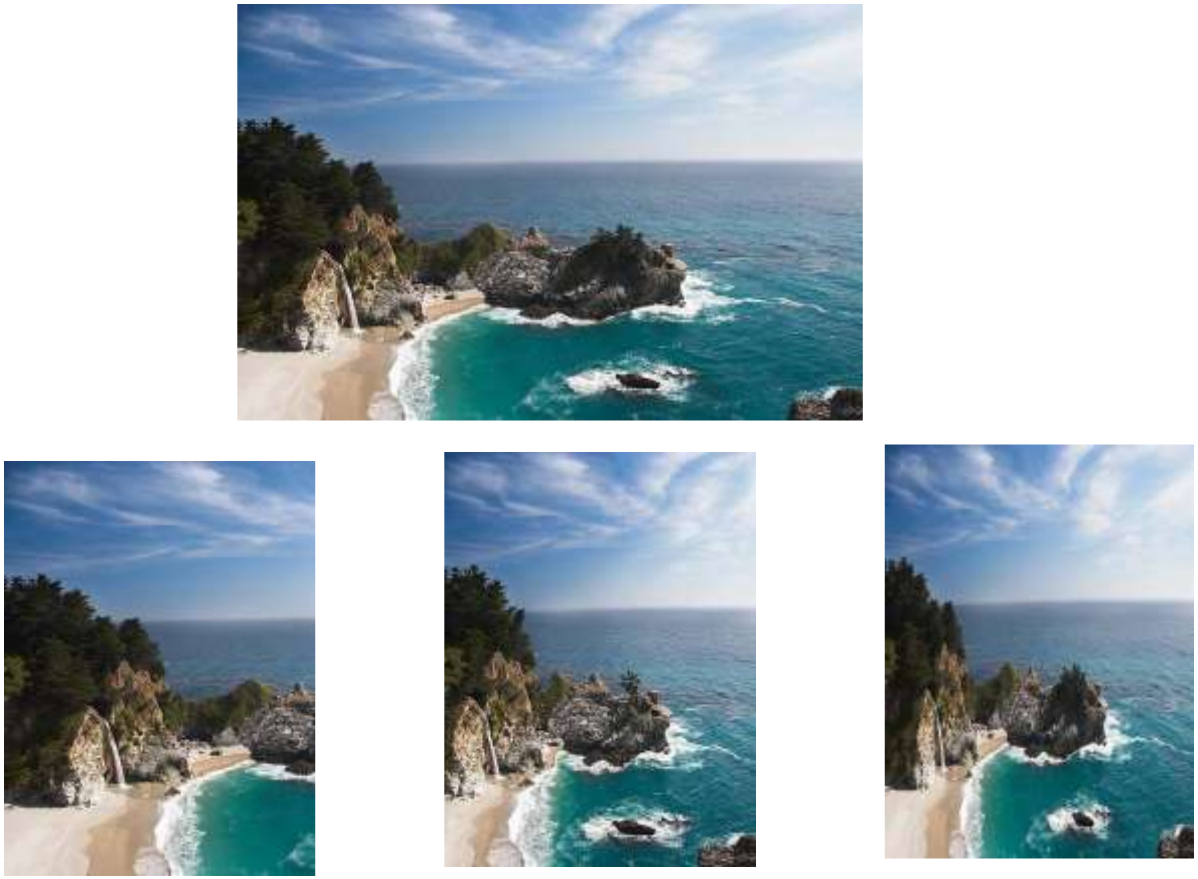
Results

Original

Seam Carving

Scaling

Results



Cropping *Seams* *Scaling*

Can we enlarge an image?

Seam insertion

Find k seams to insert

Then interpolate pixels

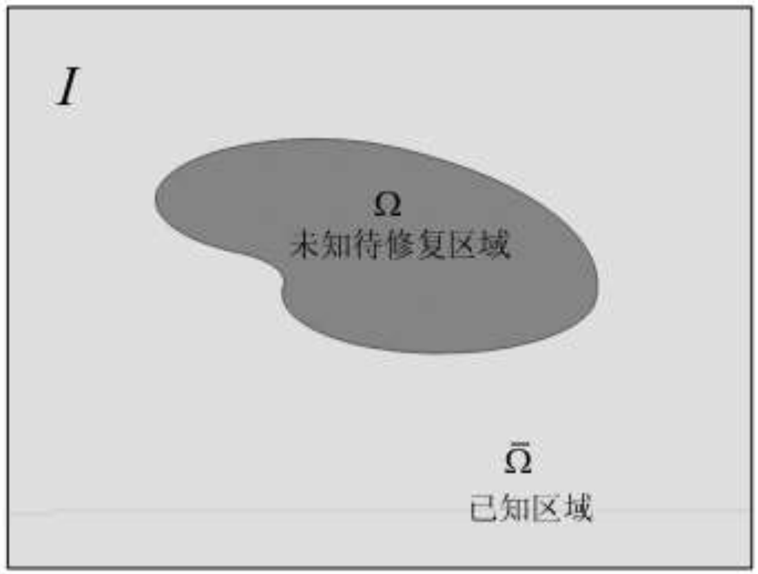
Questions?

Image completion

Restoration

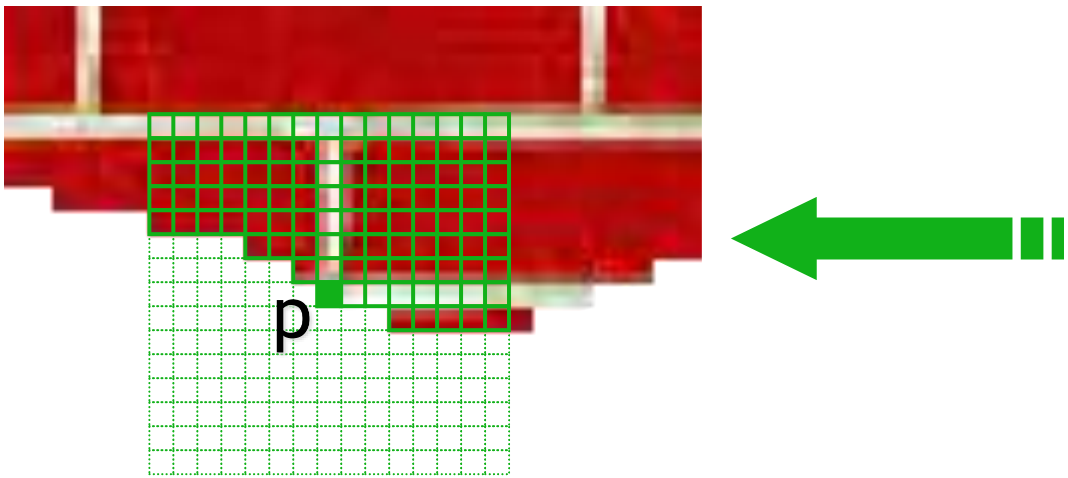
Object
removal

Problem statement

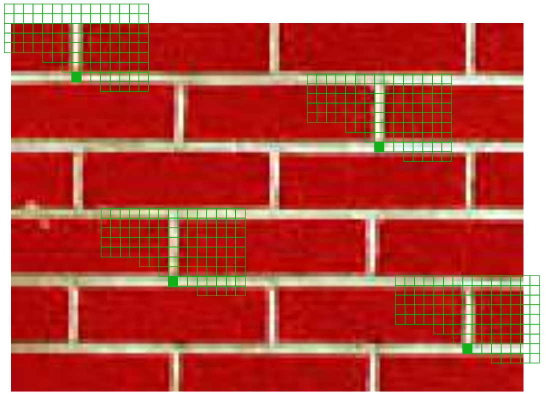


I 代表待修复图像，I 中深色区域 Ω 代表受损区域，也就是需要修补的区域，其余部分 $\overline{\Omega} = I - \Omega$ 为已知区域。Completion 即根据已知区域 $\overline{\Omega}$ 修复未知区域，得到重建区域 Ω'，使得修复后的图像 $I' = \Omega' \cup \Omega$ 在视觉上自然

Examplar-based methods



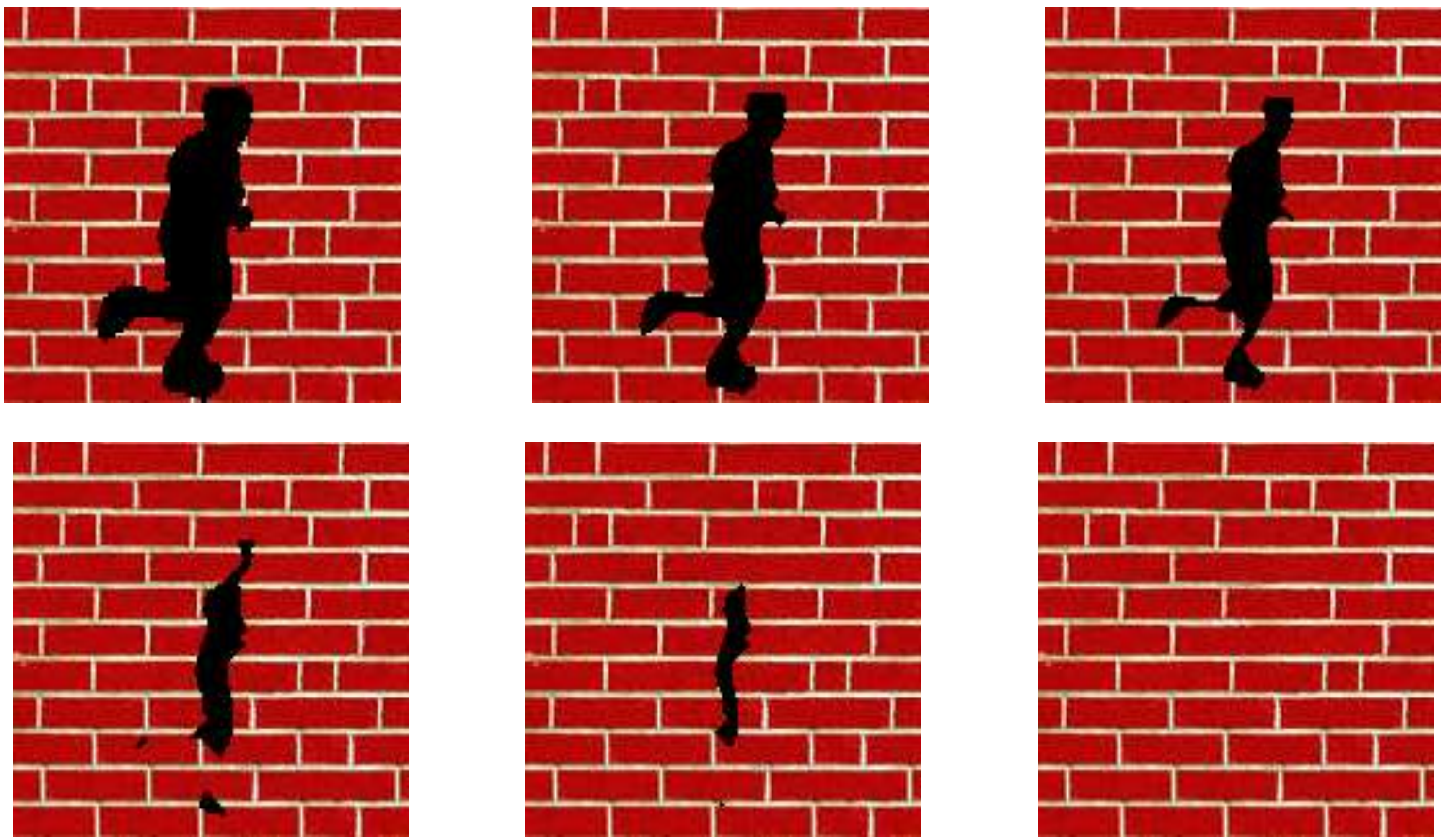
空洞的边界



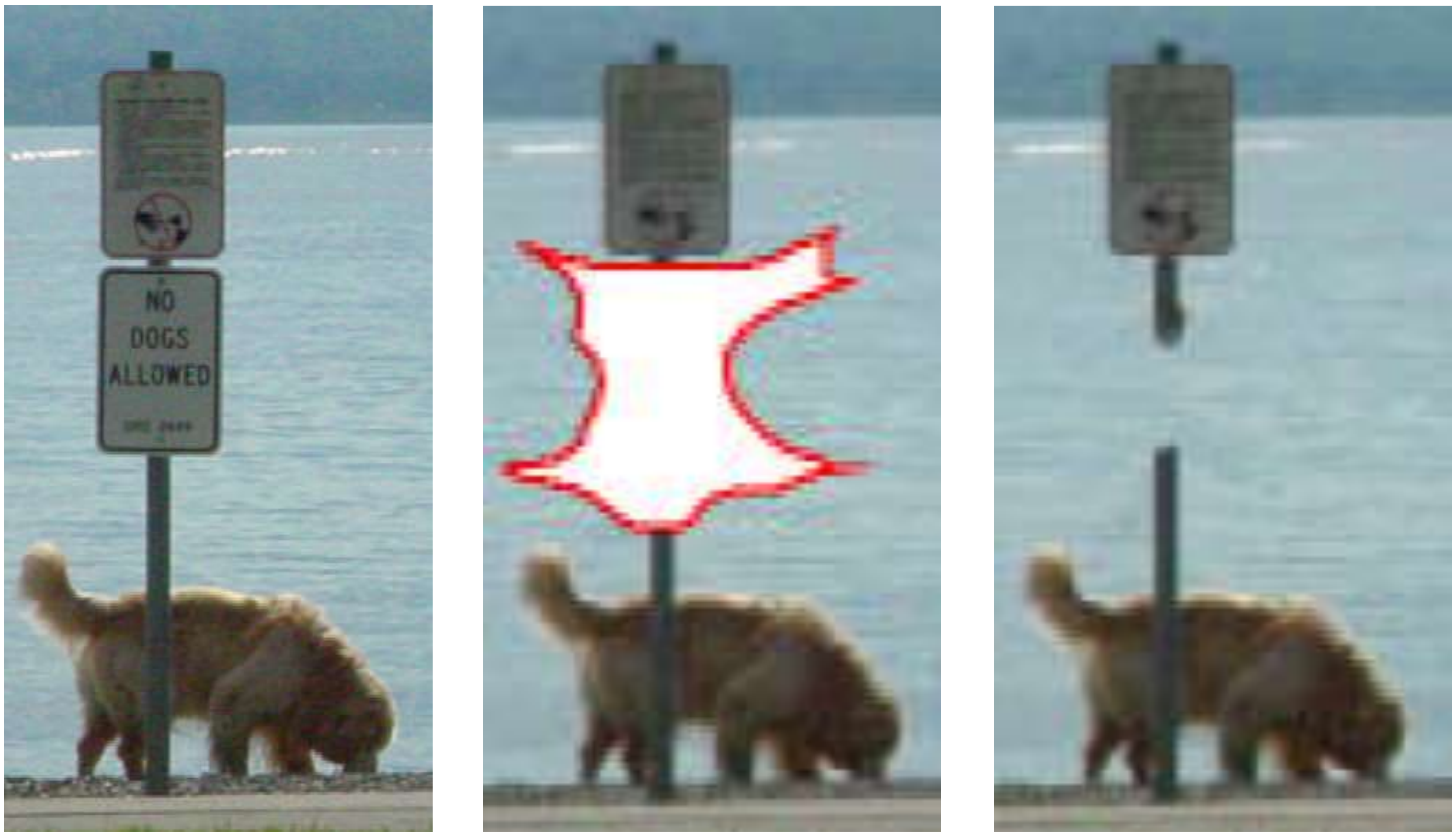
已知的样本区域

Examplar-based methods

“剥洋葱”



The order matters!



基本问题: 结构保持

如何保持结构连续？

带交互的补全算法

Image Completion with Structure Propagation

J. Sun, L. Yuan, J. Jia, and H. Shum

SIGGRAPH 2005

带有优先级的填充策略

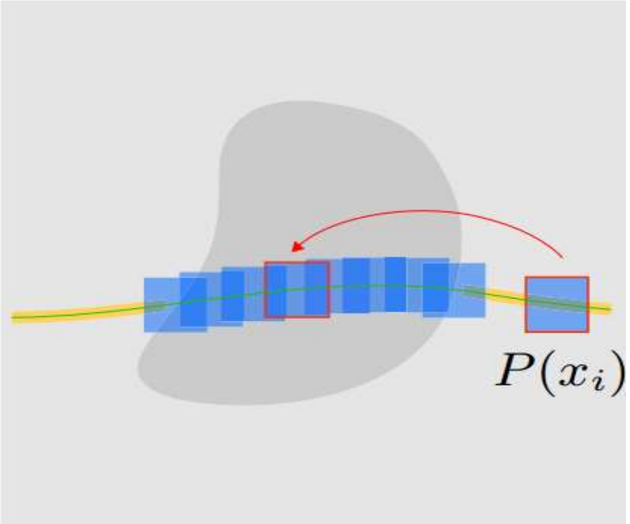
算法概览

1. 用户输入：用户在空洞区域以及已知图像区域勾画结构线,
2. 结构补全：该算法在已知图像区域采样, 通过优化一个目标能量来决定如何将样本填充被结构线覆盖的空洞区域
3. 纹理补全：补全剩余区域的纹理

目标能量

对于每一个锚点 p_i 我们找到一个标签 $x_i \in \{1, 2, ..., N\}$ 对应于其中的一个样本块，

将样本块 $P(x_i)$ 复制到 p_i 的位置如下图所示。



能量函数定义如下:

$$E(X) = \sum_{i \in v} E_1(x_i) + \sum_{(i,j) \in \varepsilon} E_2(x_i, x_j),$$

$$E_1(x_i) = k_s E_s(x_i) + k_i E_I(x_i).$$

$E_s(x_i)$， $E_I(x_i)$ 和 $E_2(x_i, x_j)$ 分别表示结构，边界和一致性约束。

目标能量

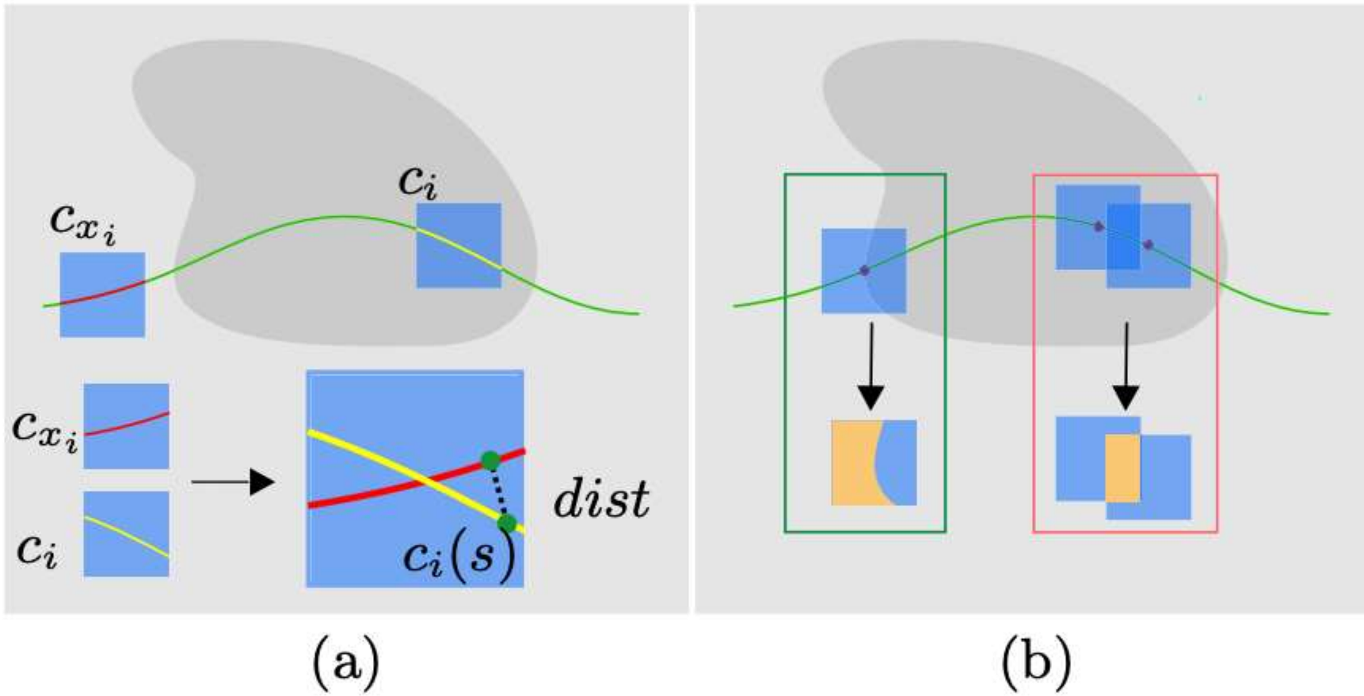
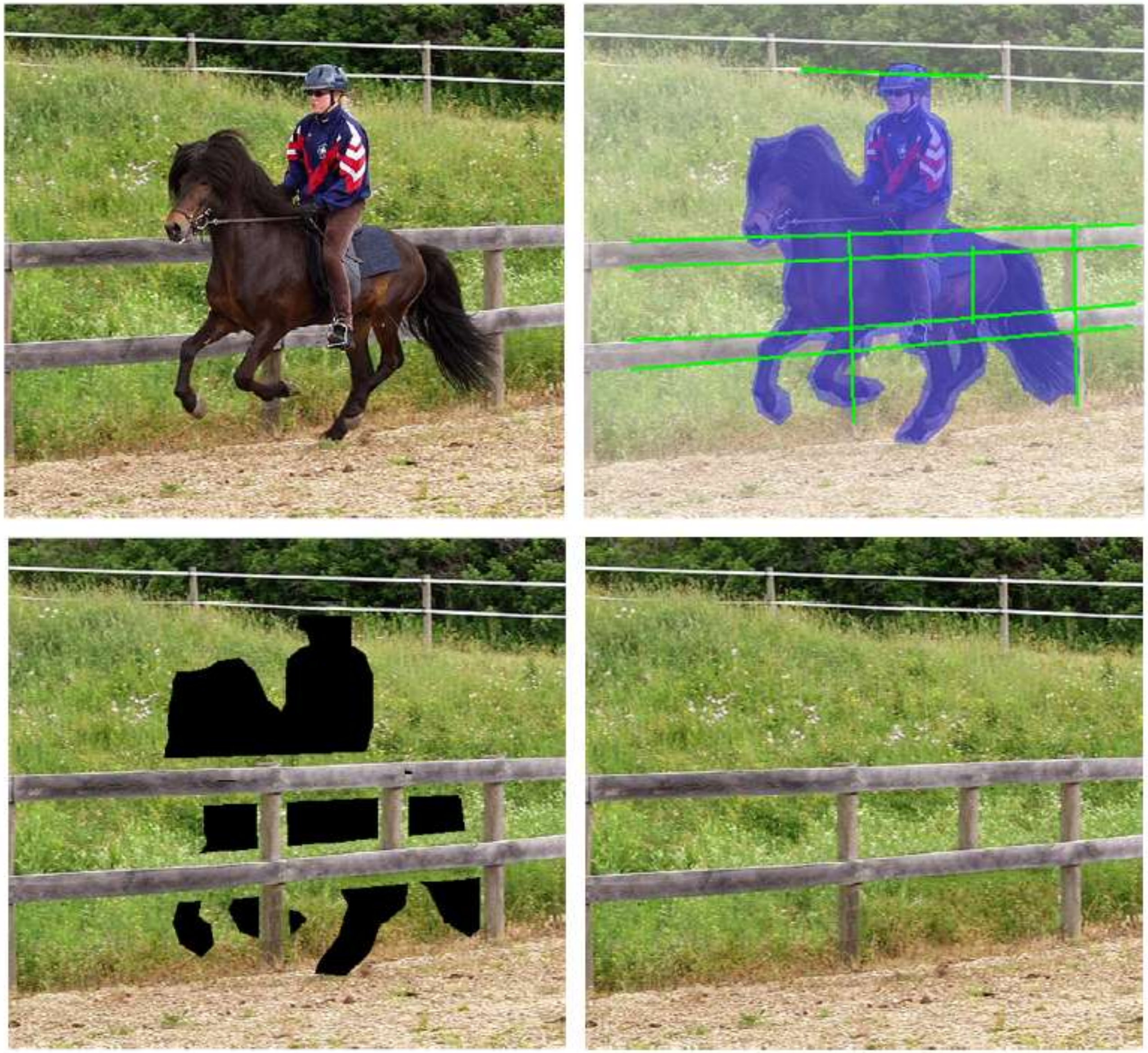


Figure 3: Energy terms for structure propagation. (a) Curve segments c_{x_i} (red) in the source patch, and curve segments c_i (yellow) in the target rectangle. $E_S(x_i)$ measures the structure similarity between c_{x_i} and c_i. dist is the shortest distance (black dotted line) from point $c_i(s)$ on segment c_i to segment c_{x_i}. (b) The green box shows the cost $E_I(x_i)$ on the boundary of the unknown region. The red box shows the cost $E_2(x_i, x_j)$ for neighboring patches.

实验结果

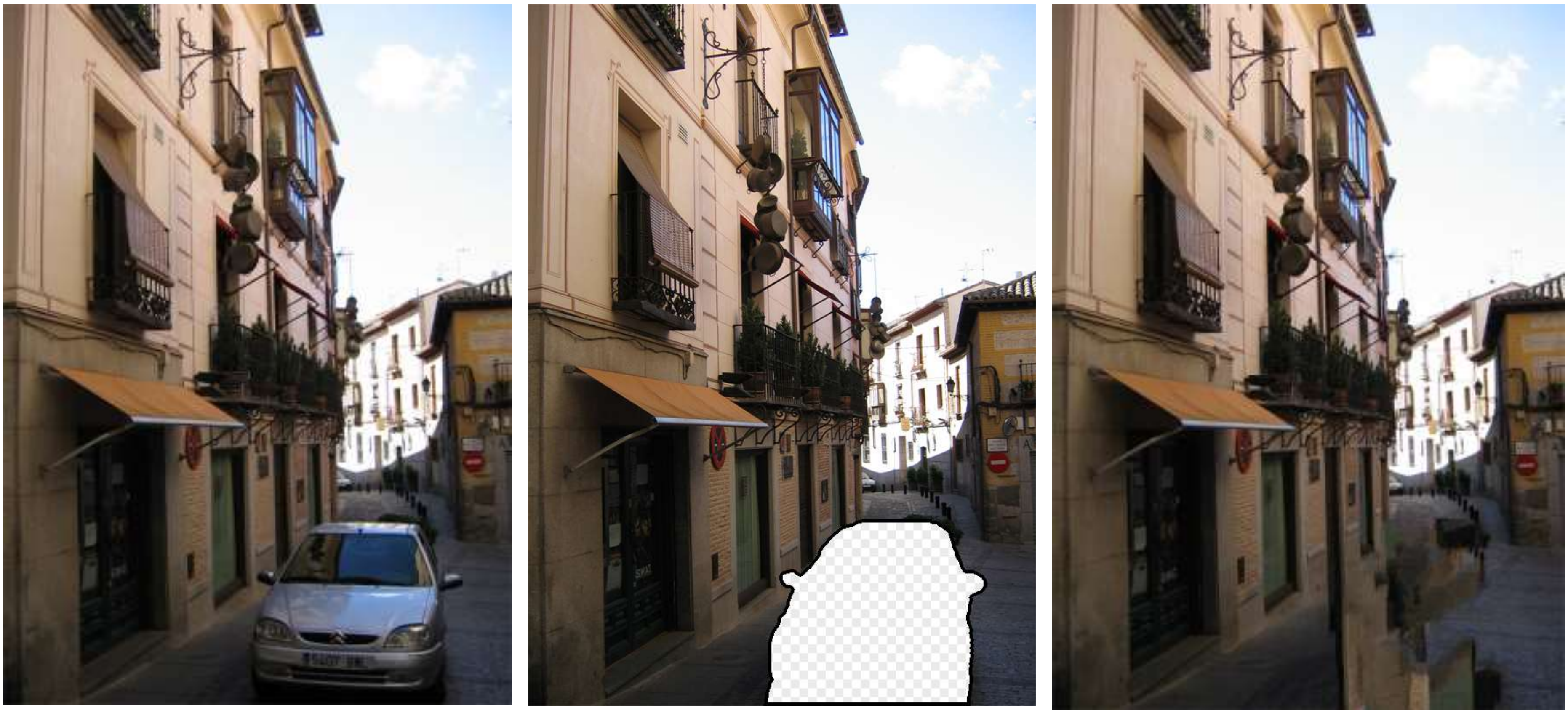
实验结果比较

实验结果



实验结果比较

Using local pathces may be insufficient



Criminisi et al. result

利用更多大数据

Scene Completion Using Millions of Photographs

James Hays and Alexei A. Efros

SIGGRAPH 2007

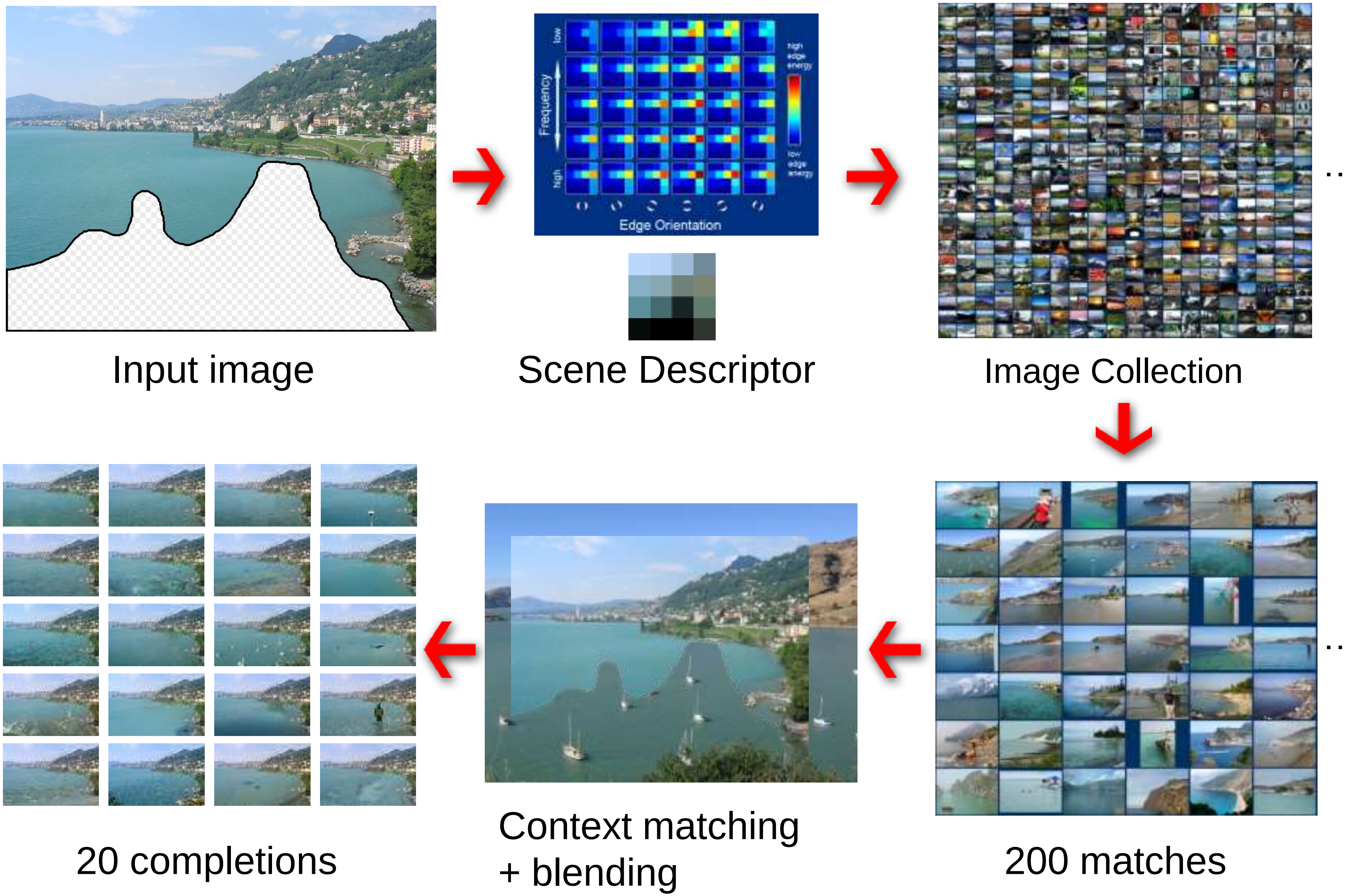
Scene Matching for Image Completion

Data

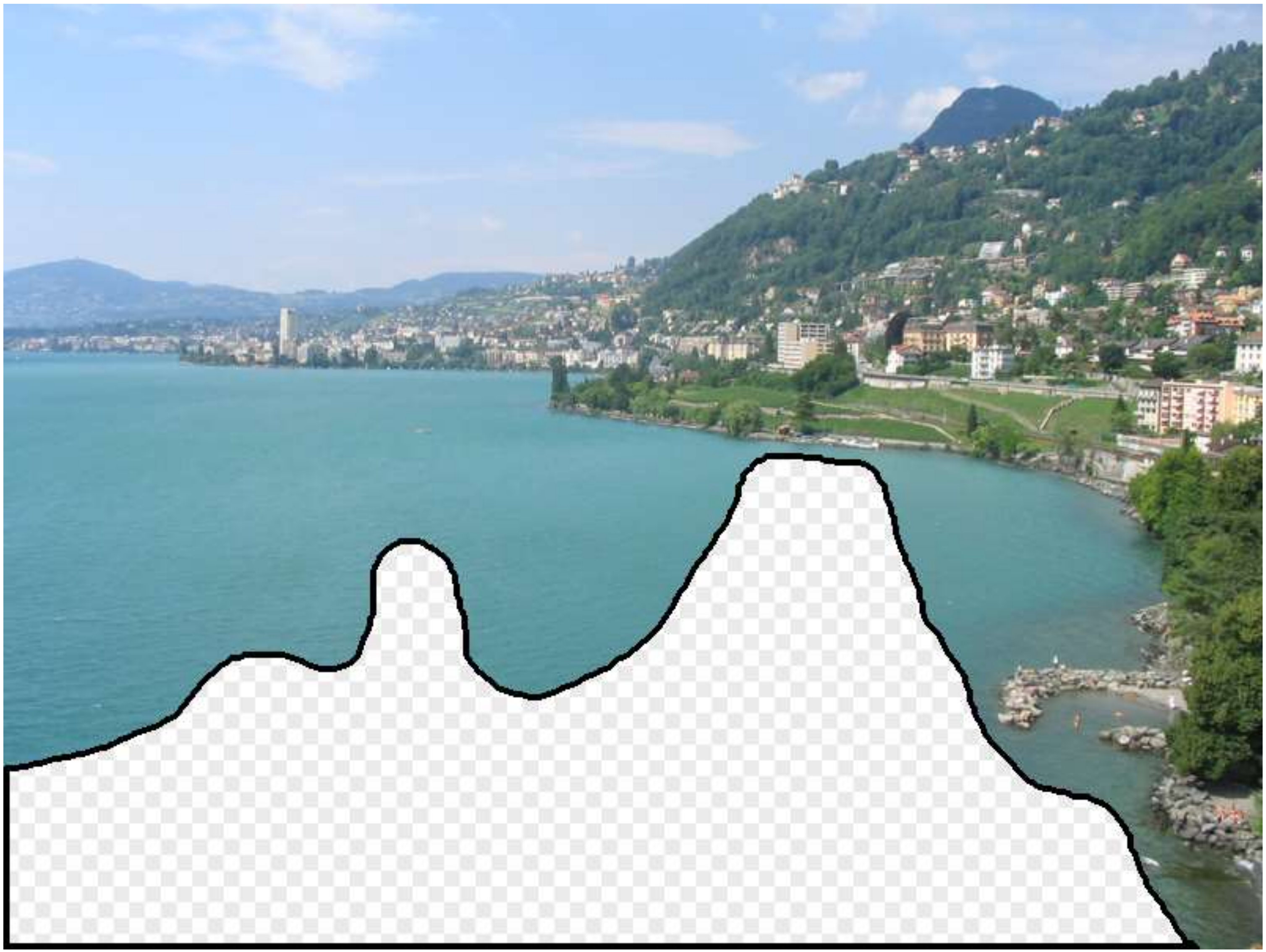
2.3 Million unique images from Flickr groups and keyword searches.

Scene Completion Result

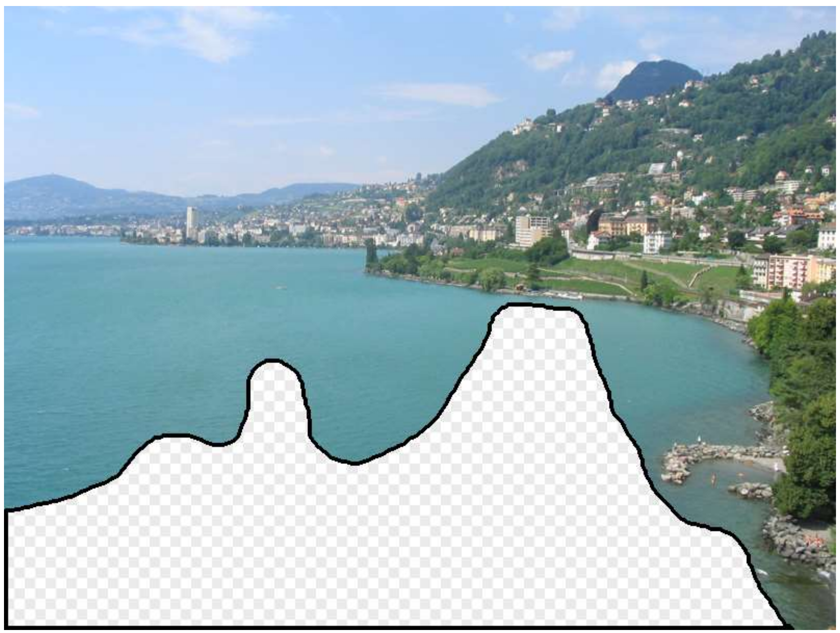
The Algorithm

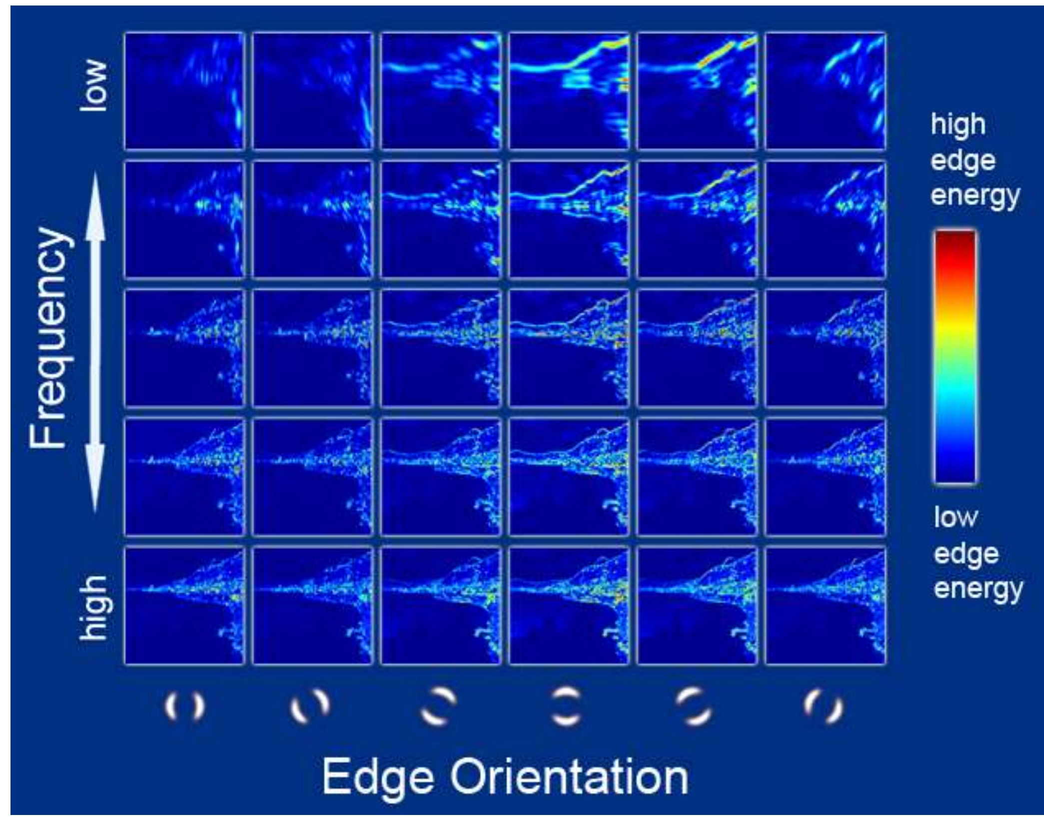


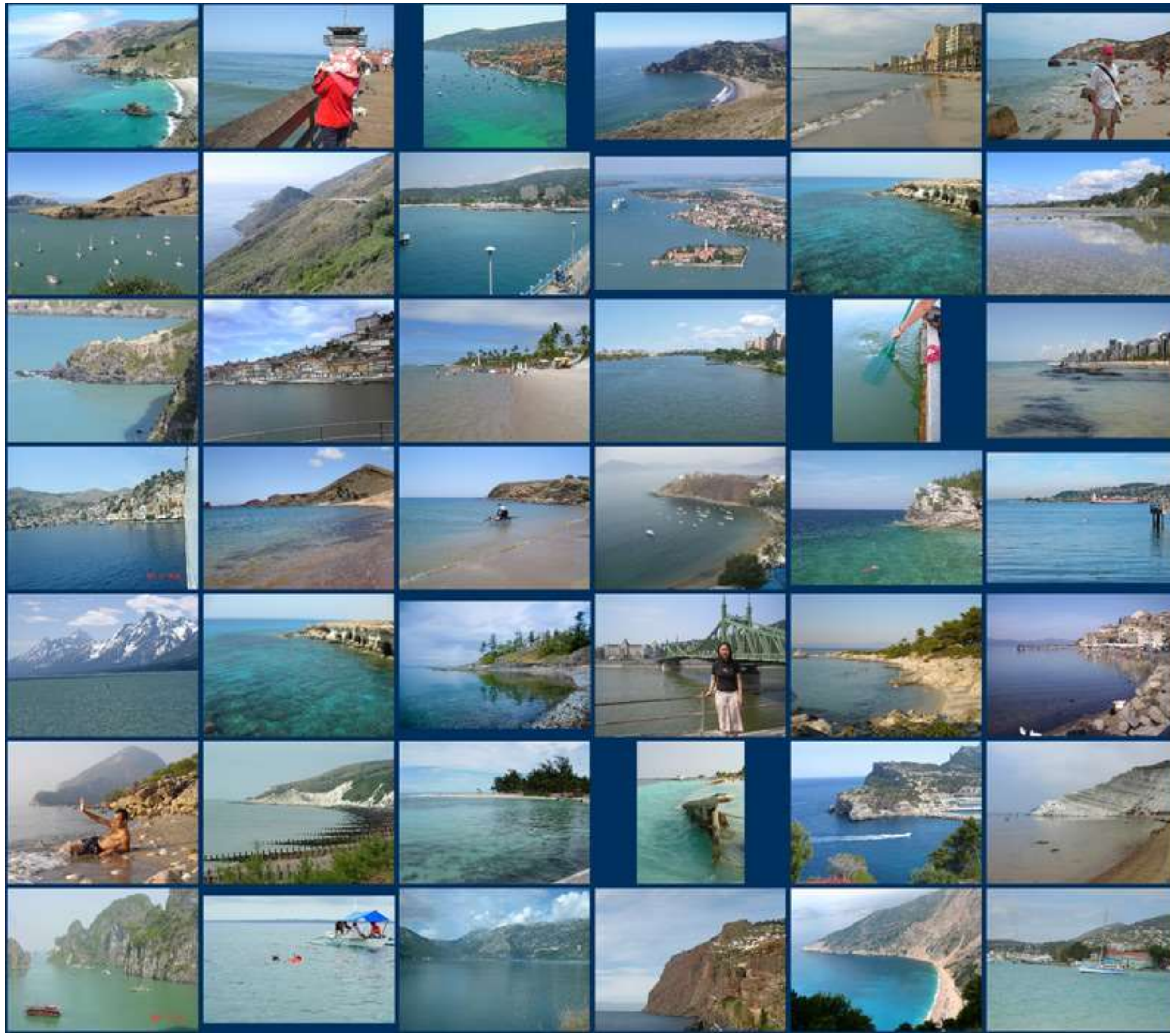
Scene Matching



Scene Descriptor

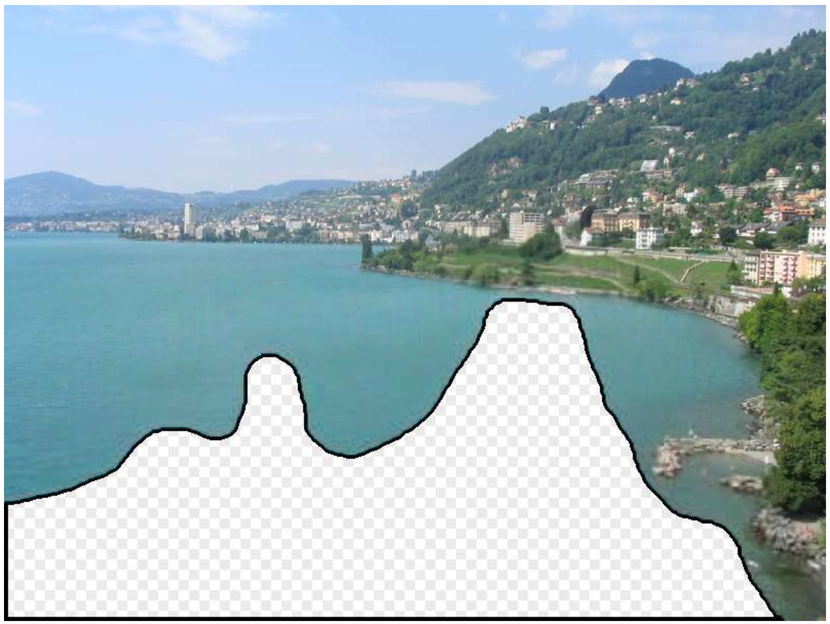




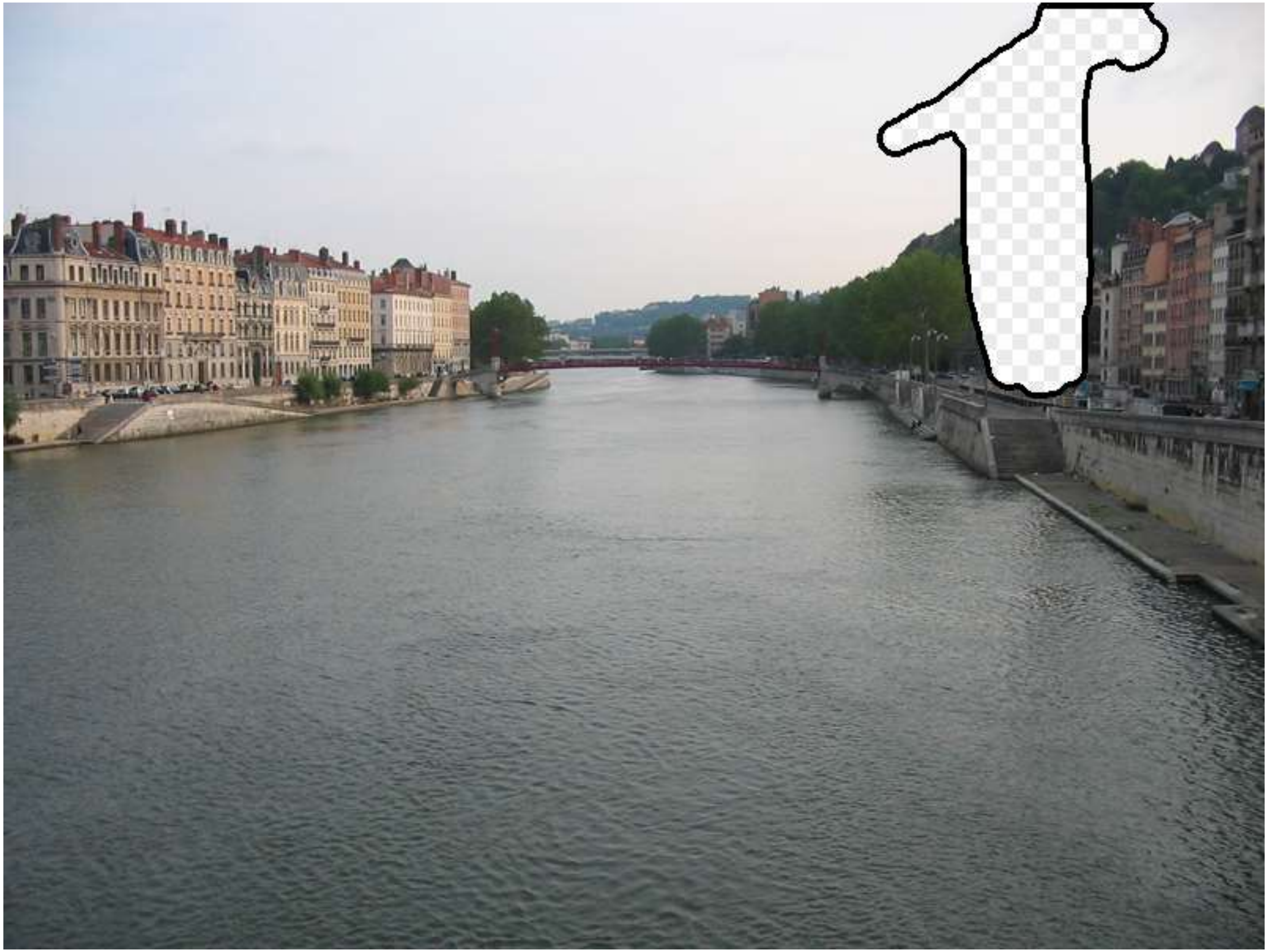


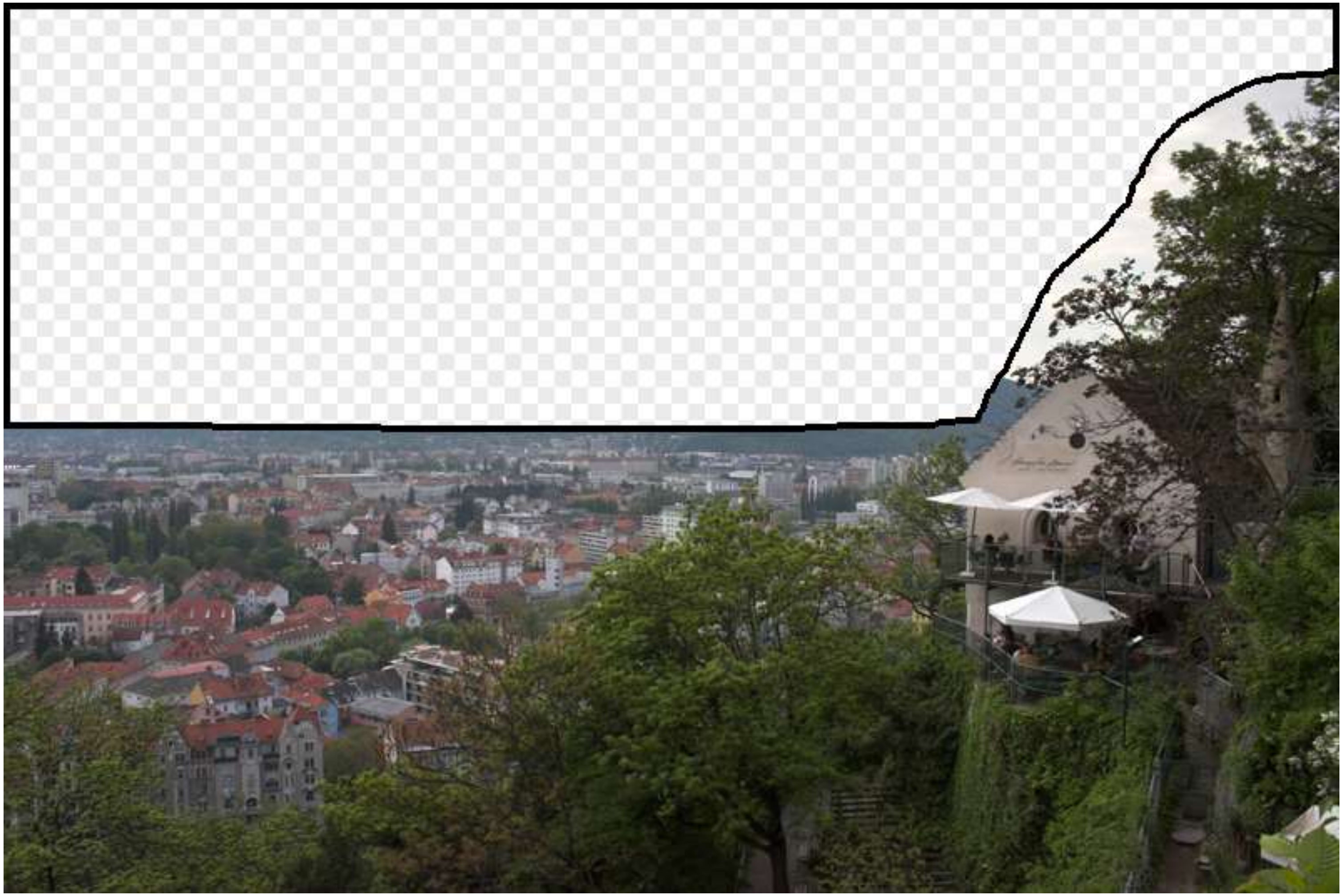
… 200 total

Context Matching







Using deep learning?