Representation and Coding of Signal Distances

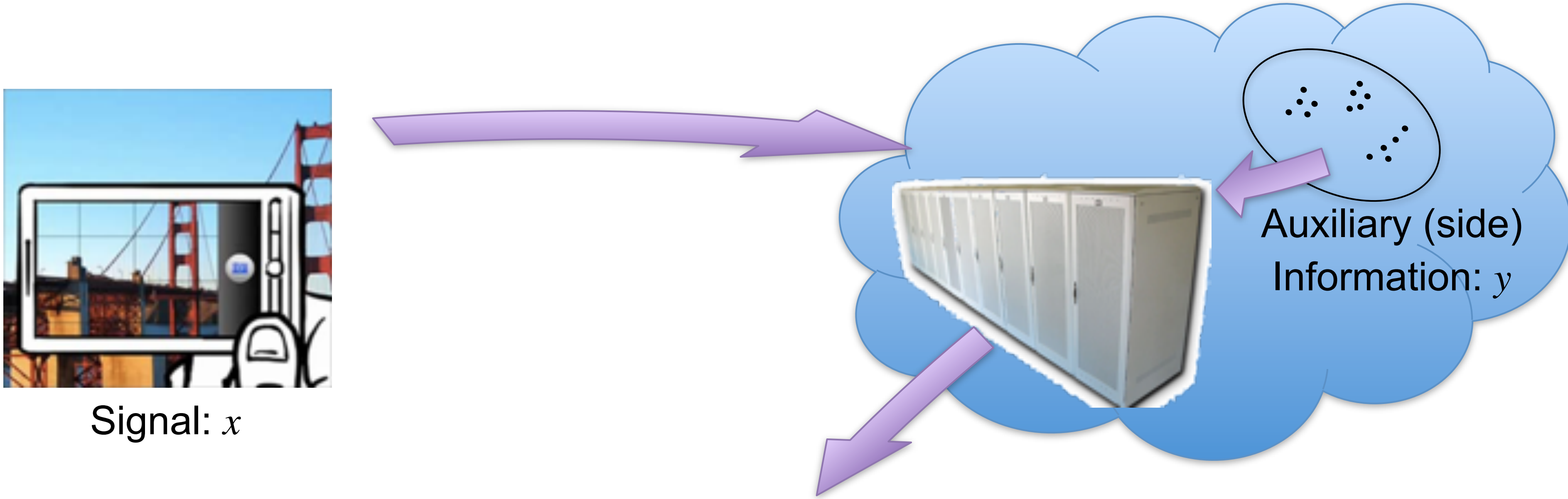
Petros Boufounos

petros@boufounos.com

(Joint work w/ Shantanu Rane)

MITSUBISHI ELECTRIC
Changes for the Better

The Big Picture



Output: $g(x,y)$

Information Scalable Coding:

How to encode signal for general functions $g(\cdot,\cdot)$?

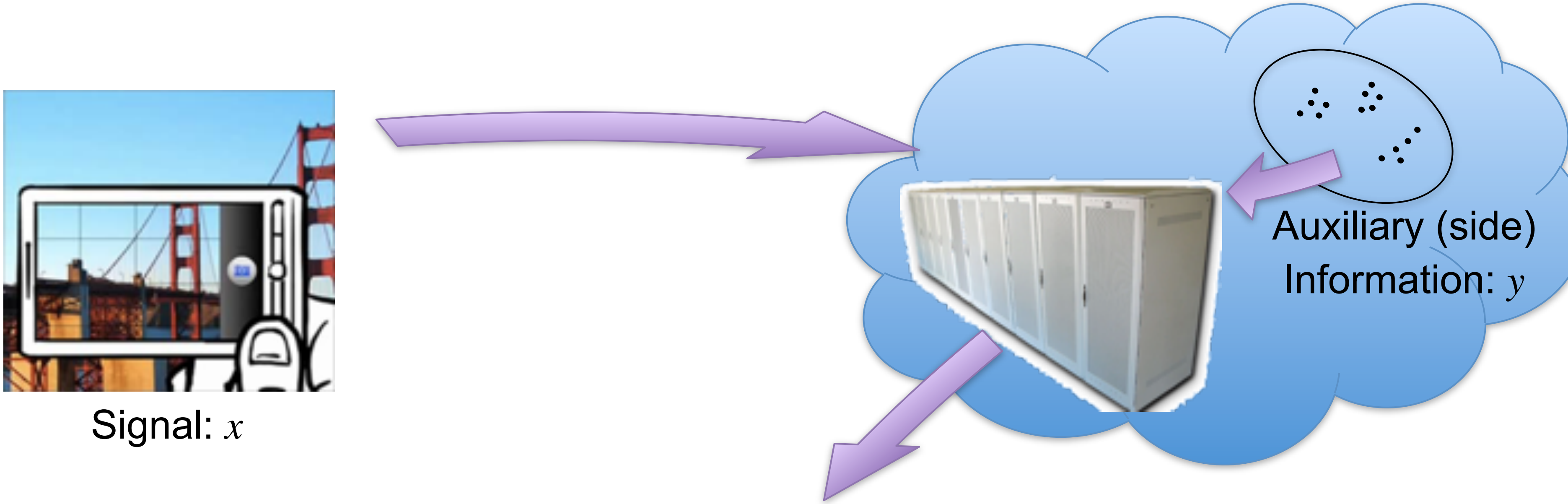
Main questions:

Rate- and computation-**efficient encoding**

Accurate and **efficient computation** of $g(\cdot,\cdot)$

Interaction of encoding and computation

The Big Picture



Output: $g(x,y)$

Example/Special Case:

Function **computes** just **signal**, $g(x,y)=x$,

No auxiliary **information**

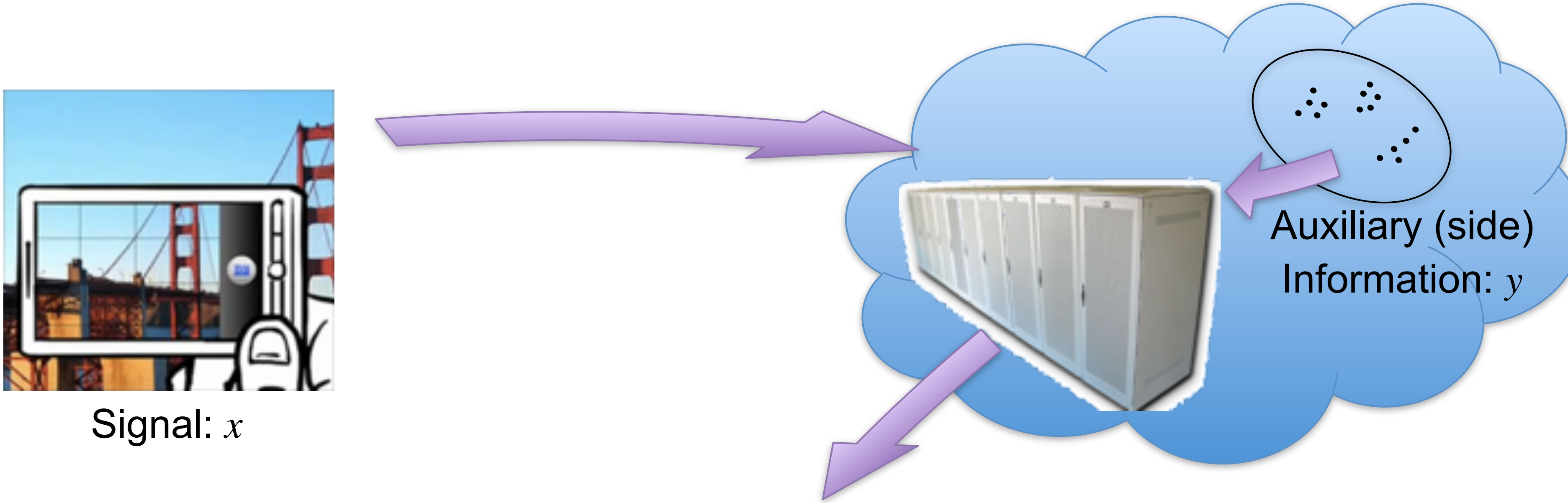
⇒ **Conventional compression**/source coding

Encoding Efficiency: **Rate**

Accuracy of function computation: **Distortion**

Interaction: **Rate/Distortion theory**

The Big Picture



Output: $g(x,y)$

Example/Special Case:

Function **computes** just **signal**, $g(x,y)=x$,

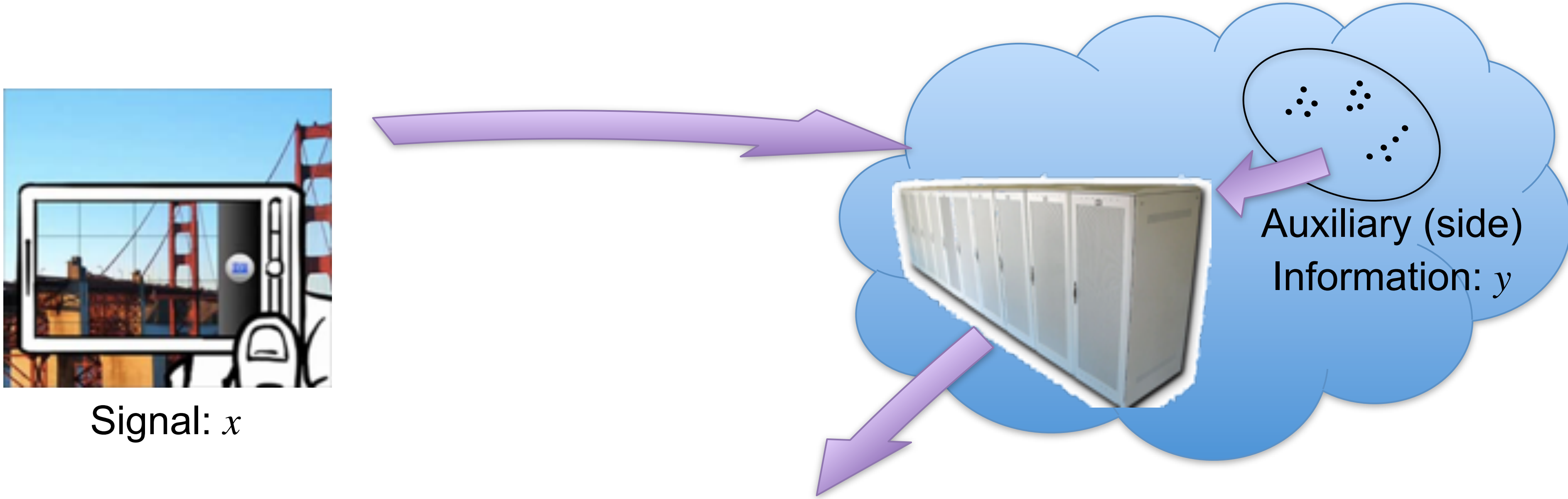
Auxiliary **information: similar signals**

$\Rightarrow$ **Distributed** source **coding** (coding with side information)

Slepian-Wolf coding: discrete source, lossless—no distortion

Wyner-Ziv coding: continuous source, lossy—rate/distortion

The Big Picture



Output: $g(x,y)$

Today

Function computes **functions of signal distances**,

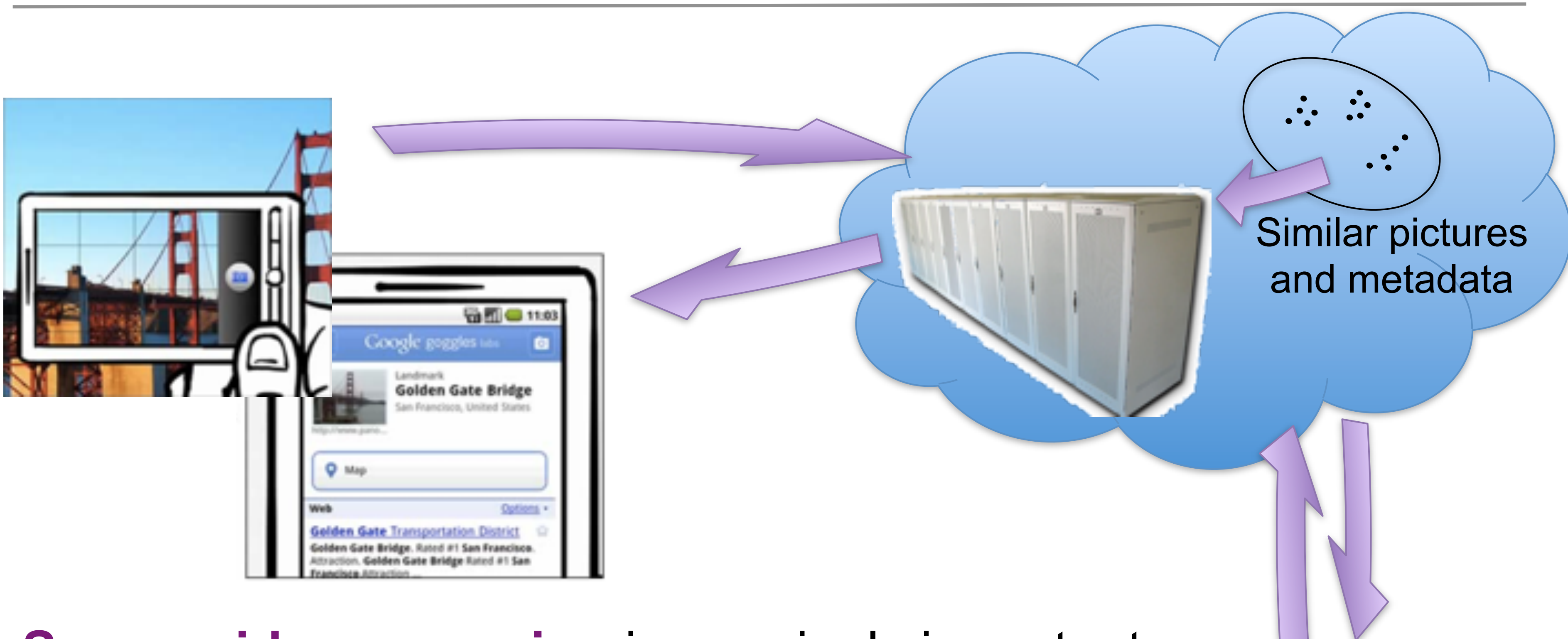
$$g(x,y)=g(\|x-y\|_2),$$

Auxiliary **information: other signals**

$\Rightarrow$ Coding of signal distances

Main tool: **Embeddings**

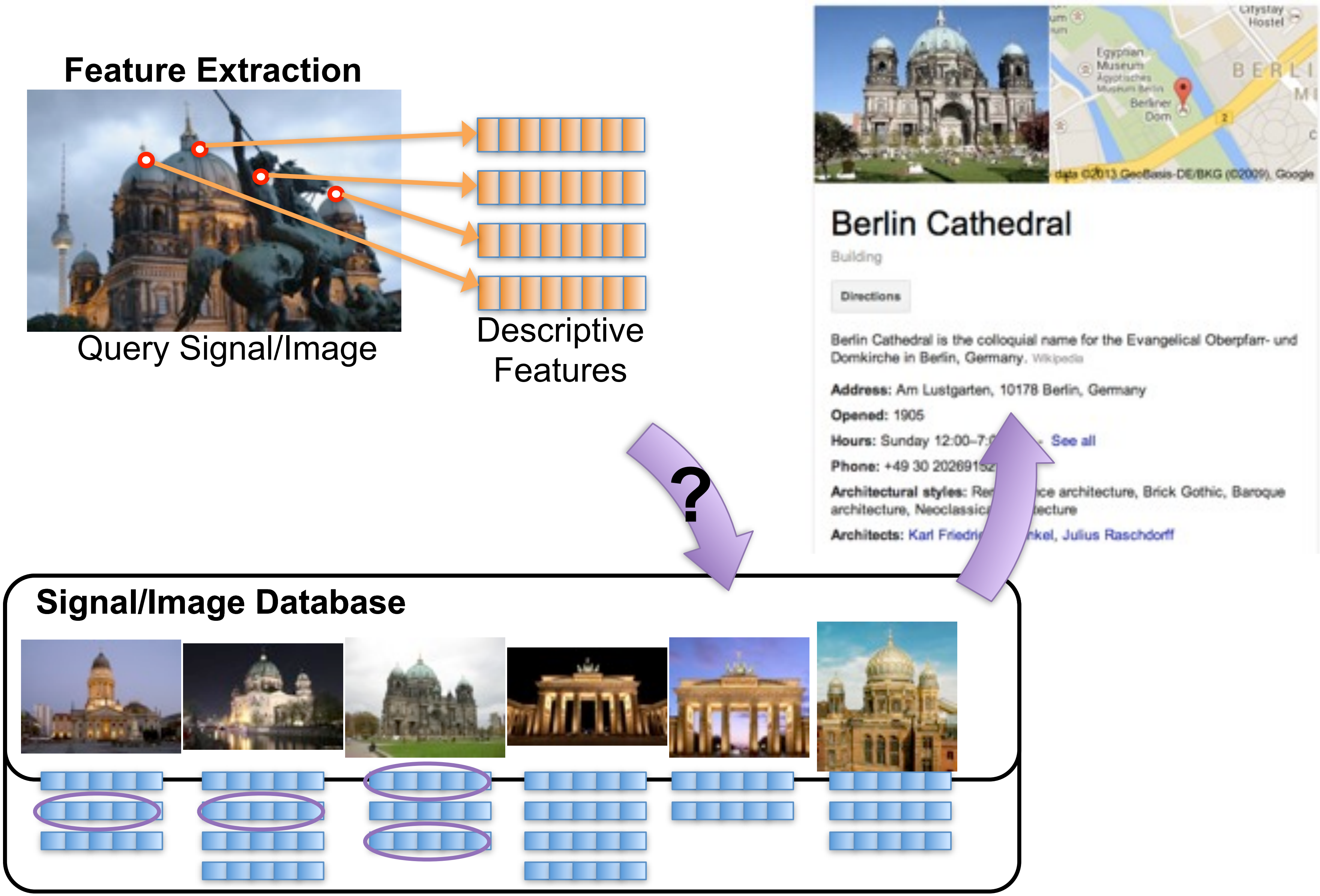
Motivation: Augmented Reality



Server-side processing increasingly important
(e.g. cloud computing, augmented reality)
Compression is necessary
Goal: detection; not image transmission

Q: Should we transmit the signal?
Can we reduce the rate?

Signal/Image-based Retrieval



Detection/Classification Pipeline (typical)

Feature Extraction → Detection/ Classification → "Golden Gate Bridge"

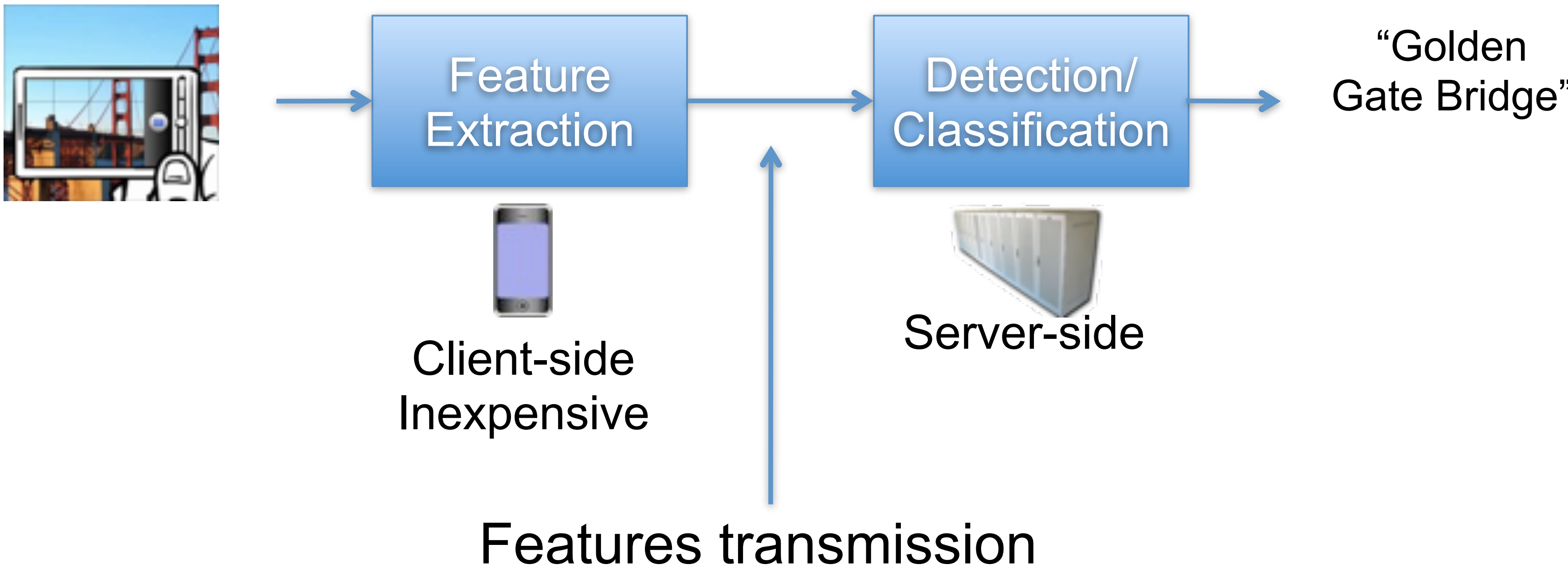
Server-side
Requires reconstruction

Server-side

Image transmission

Detection/Classification: Based on signal geometry

Detection/Classification Pipeline (efficient)



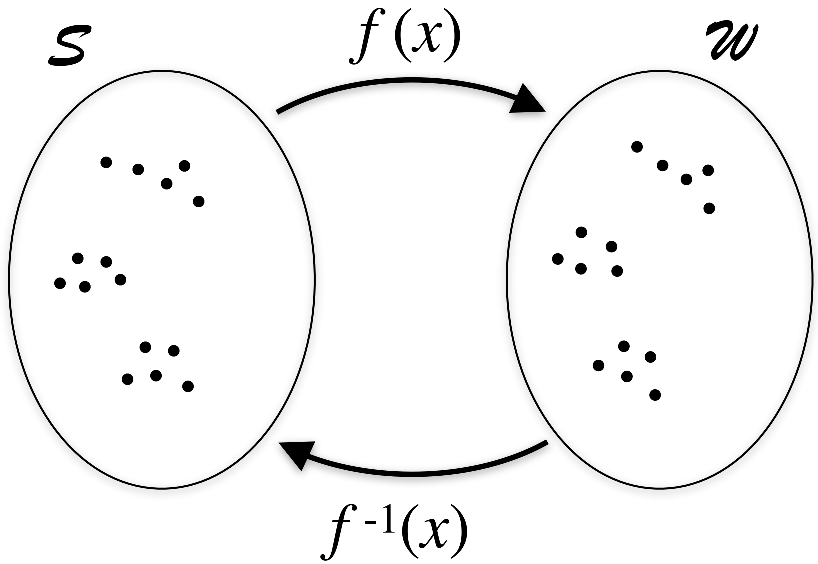
Detection/Classification: Based on signal geometry

Goal: rate-efficient geometry-preserving transmission

GEOMETRY-PRESERVING EMBEDDINGS

Isometric (approximate) embeddings

Original space
high-dimensional and
expensive to work with



Embedding space
lower dimension or
easier to work with
(hopefully)

Transformations that preserve distances

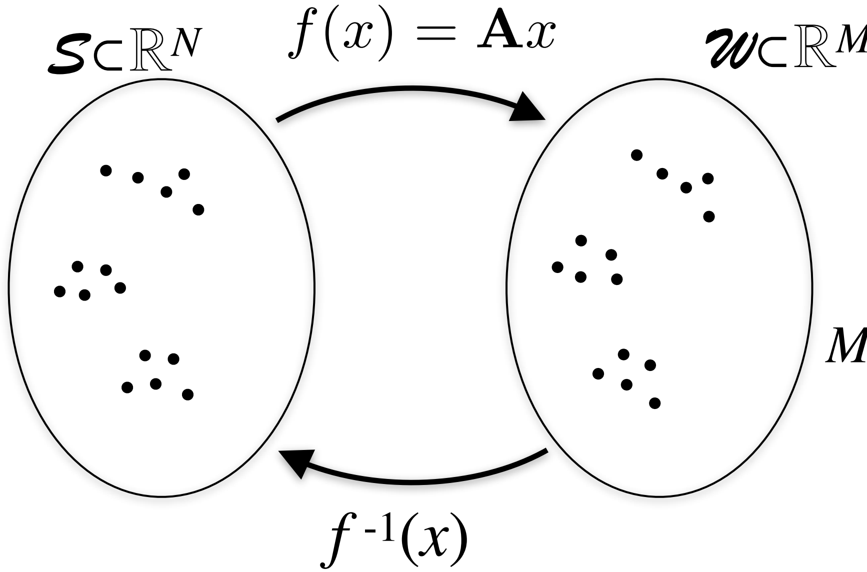
For all x,y in $\mathcal{S}$: $d_{\mathcal{S}}(x,y) \approx d_{\mathcal{W}}(f(x),f(y))$

Johnson-Lindenstrauss embeddings

Original space

Distance metric: ℓ_2

P points in $\mathbb{R}^N$



Embedding space

Embed in $\mathbb{R}^M$

Distance metric: ℓ_2

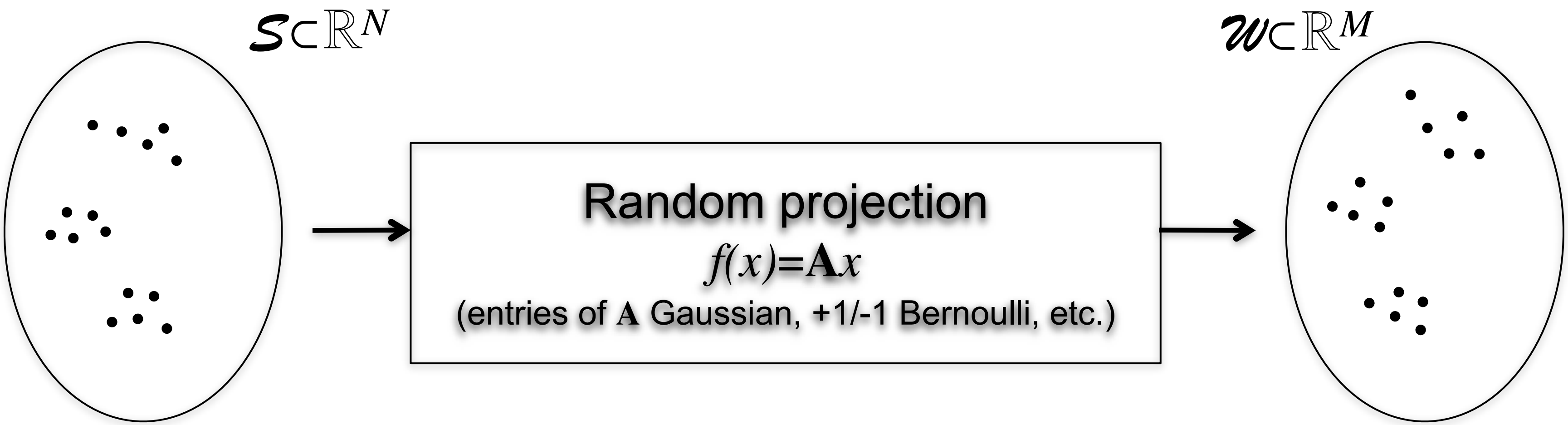
$M = O(\log P)$ dimensions

Transformations that preserve distances

For all x,y in $\mathcal{S}$:

$$(1-\epsilon)\|x-y\|_2^2 \leq \|f(x)-f(y)\|_2^2 \leq (1+\epsilon)\|x-y\|_2^2$$

Johnson-Lindenstrauss embeddings



With overwhelming probability on $\mathbf{A}$, for all x,y in $\mathcal{S}$:

$$(1-\epsilon)\|x-y\|_2^2 \leq \|f(x)-f(y)\|_2^2 \leq (1+\epsilon)\|x-y\|_2^2$$

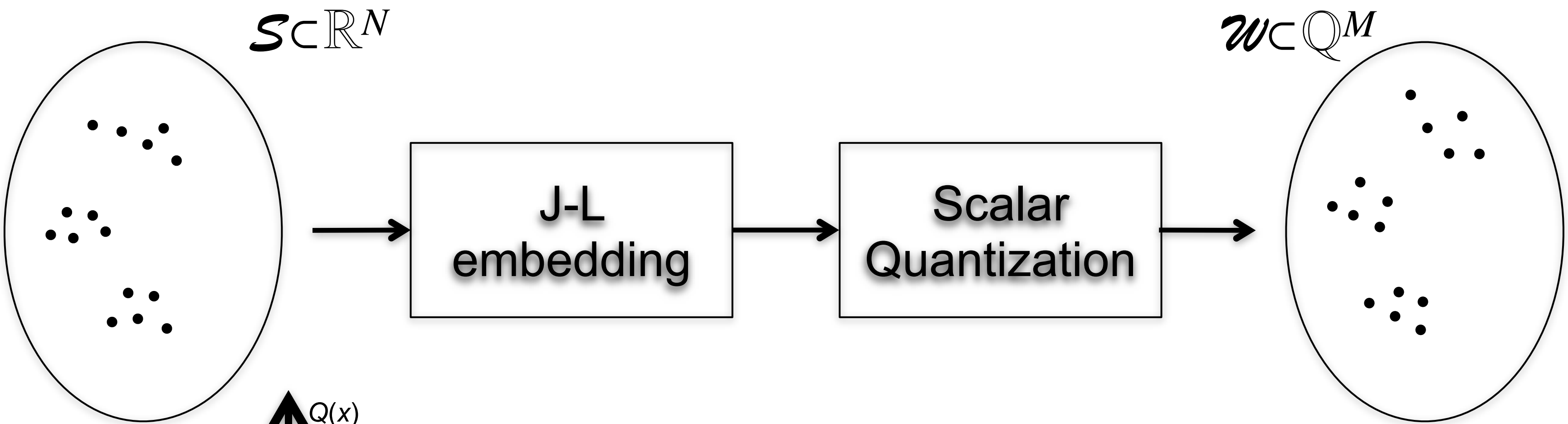
using only $M = O\left(\frac{\log P}{\epsilon^2}\right)$ dimensions

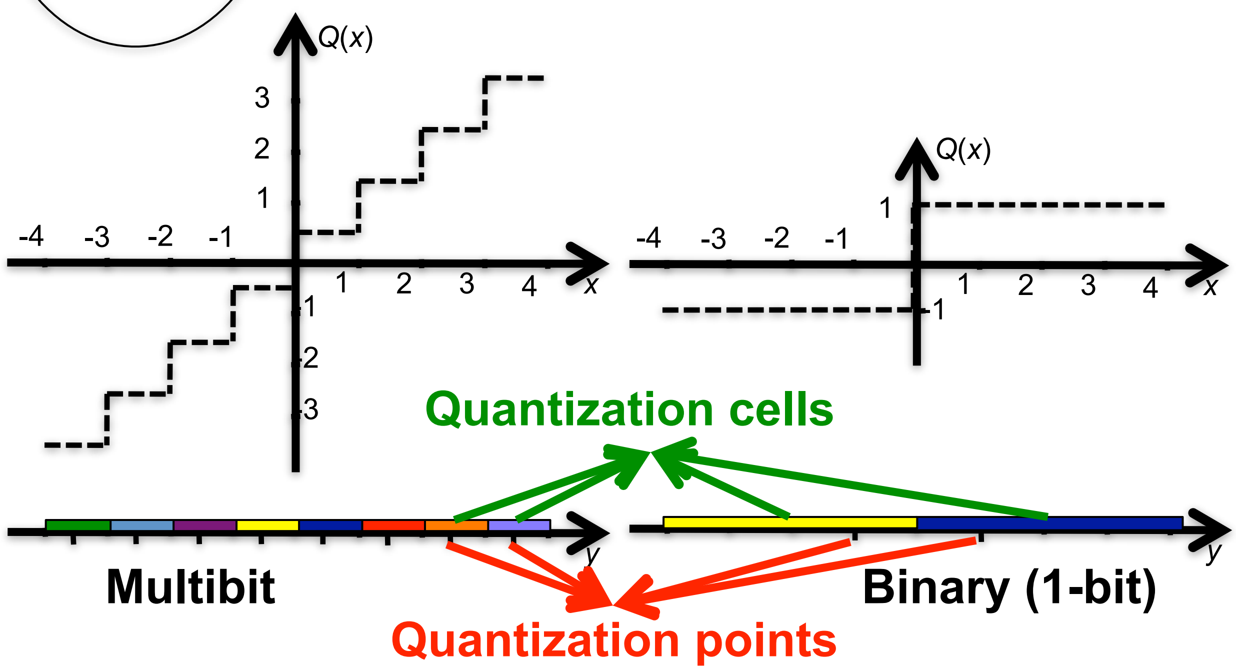
Bound (almost) tight: $M = O\left(\frac{\log P}{\epsilon^2 \log\frac{1}{\epsilon}}\right)$ dimensions necessary

BUT: Quantization is necessary for transmission!

Are J-L Embeddings still appropriate?

Quantized J-L Embeddings





B-bit scalar quantizer with dynamic range $\pm S$

Quantization interval

$$\Delta = S \cdot 2^{-B+1}$$

Binary quantizer (B=1) just preserves sign

Johnson-Lindenstrauss With Quantization [w/ Li, Rane]

Consider $\mathcal{S} \subset \mathbb{R}^N$ containing P points.

We can embed $\mathcal{S}$ in $\mathbb{R}^M$ such that for all x,y in $\mathcal{S}$:

$$
\begin{aligned}
(1-\epsilon)\|x-y\|_2 - 2^{-B+1}S \leq \\
\|Q(f(x)) - Q(f(y))\|_2 \\
\leq (1+\epsilon)\|x-y\|_2 + 2^{-B+1}S
\end{aligned}
$$

using only $M = O\left(\frac{\log P}{\epsilon^2}\right)$ dimensions

and B bits per dimension
(with appropriate normalizations/saturation levels)

Total rate: $R=BM$

Quantized J-L at Fixed Rate

Given total rate: $R=MB$

How to assign B and M? More M or more B?

Larger B, less quantization distortion

$$2^{-B+1}S$$

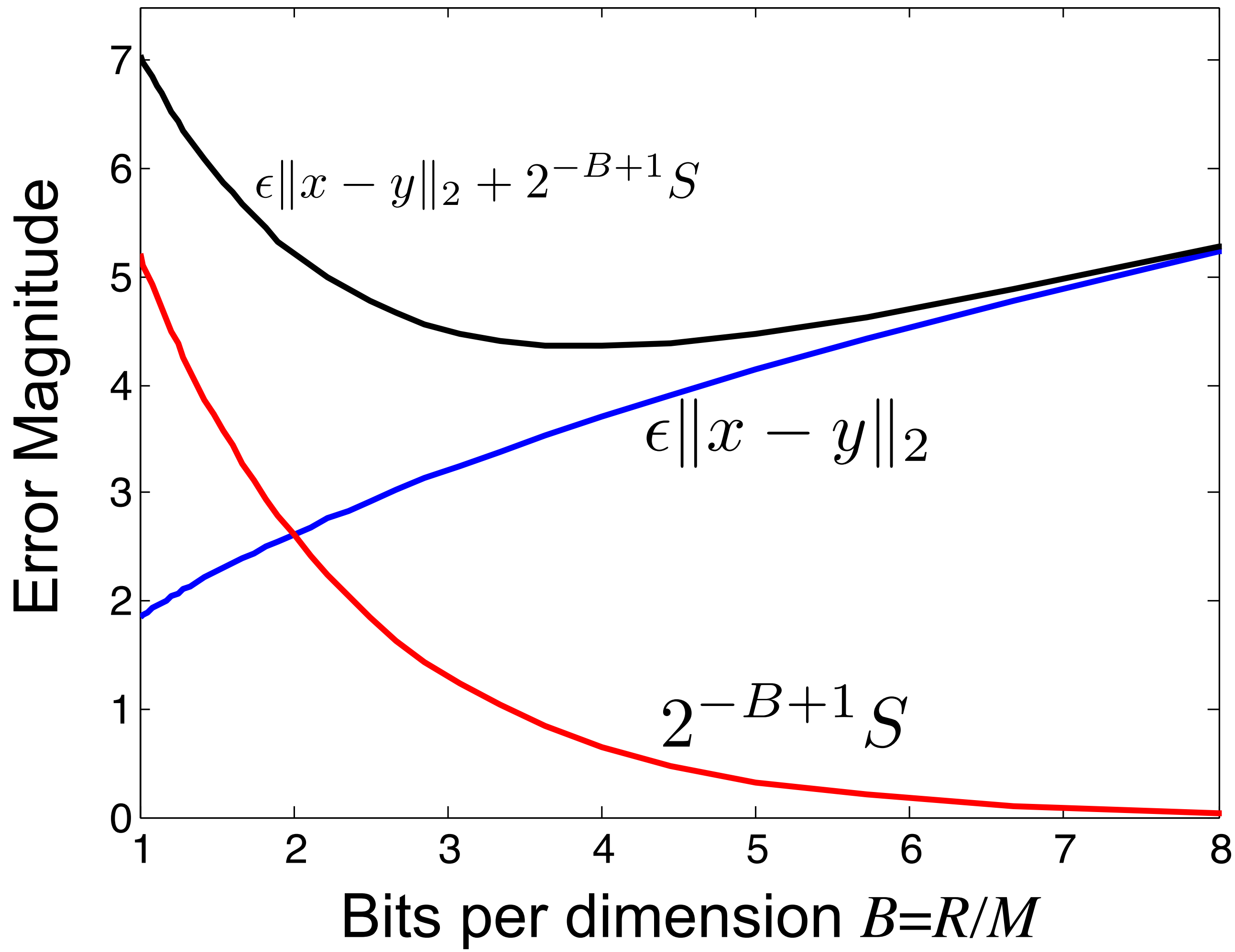
$$(1-\epsilon)\|x-y\|_2 - 2^{-\frac{R}{M}+1}S \leq \|Q(f(x)) - Q(f(y))\|_2 \leq (1+\epsilon)\|x-y\|_2 + 2^{-\frac{R}{M}+1}S$$

Larger M, less J-L type distortion ϵ

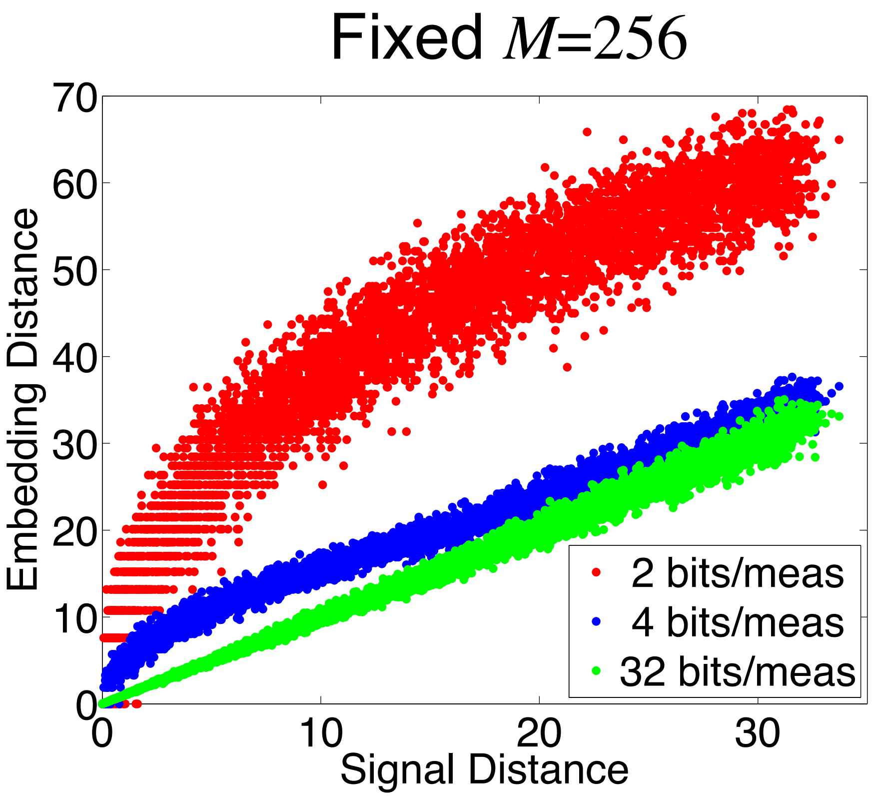
$$\epsilon = O(1/\sqrt{M})$$

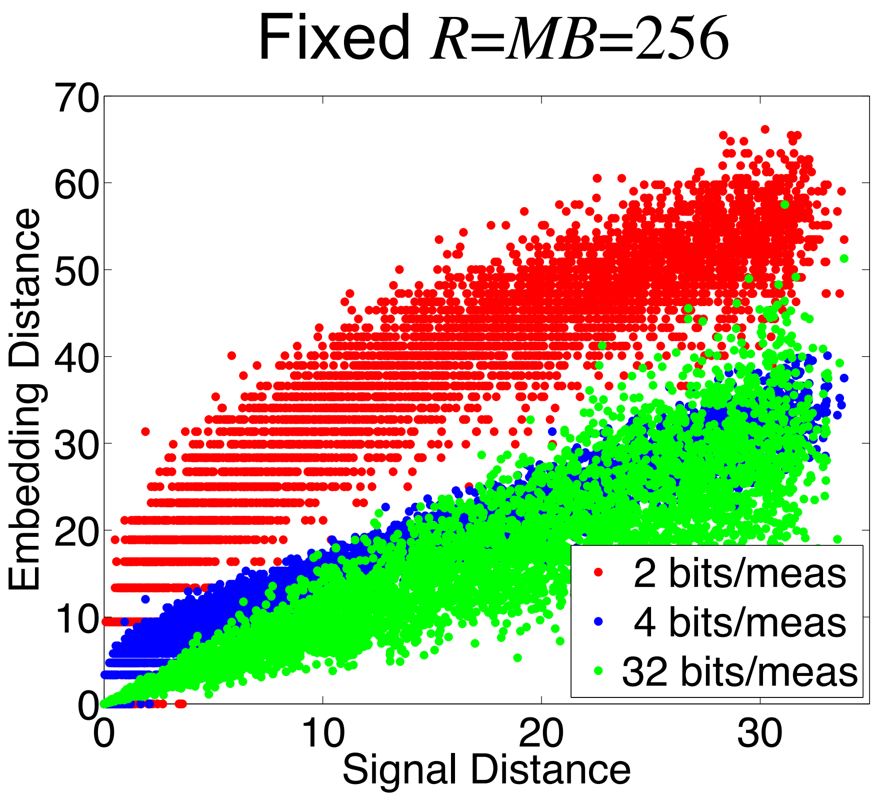
Design tradeoff:
Number of projections vs. bits per projection

Exploring the Design Trade-off



Exploring the Design Trade-off

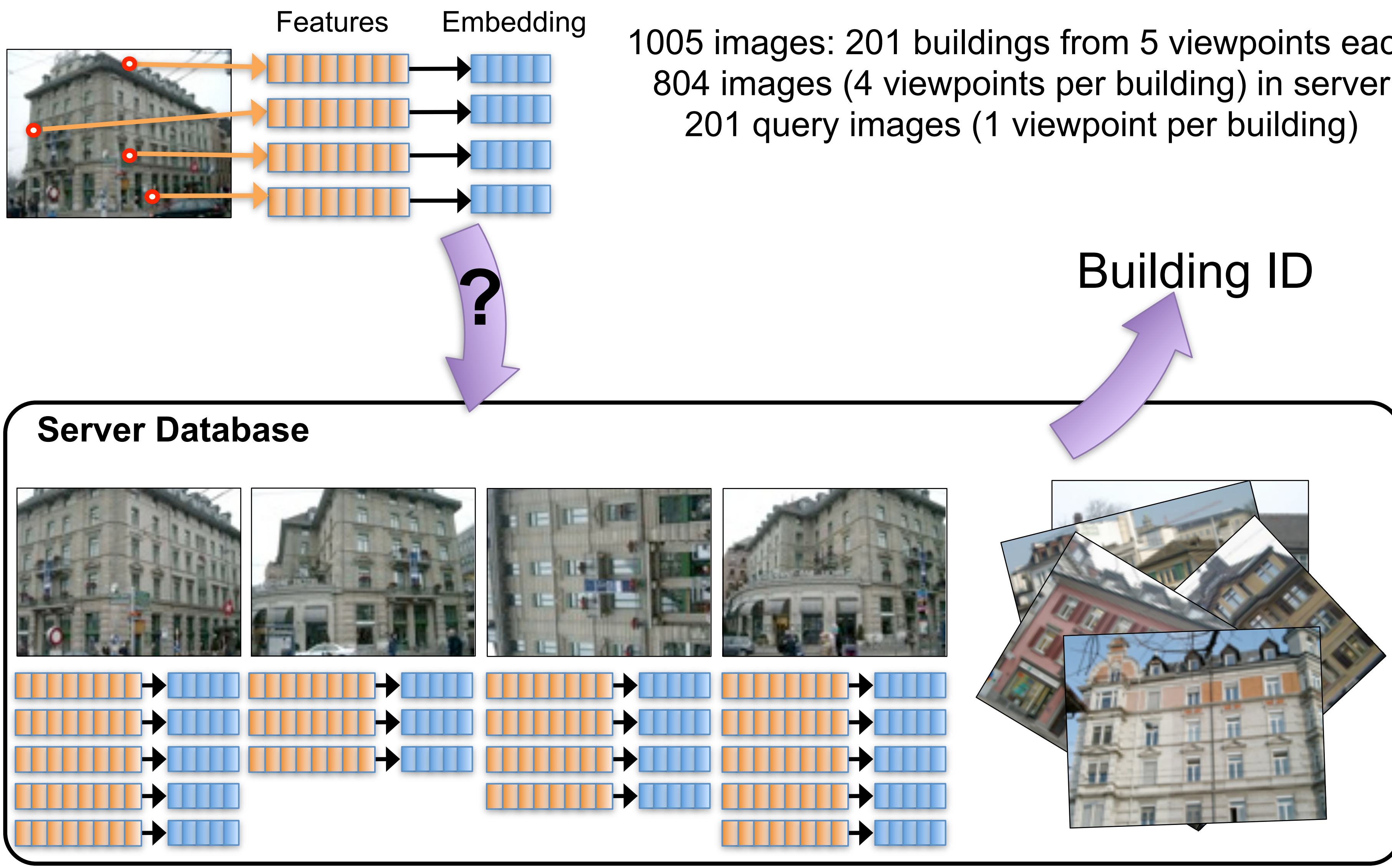




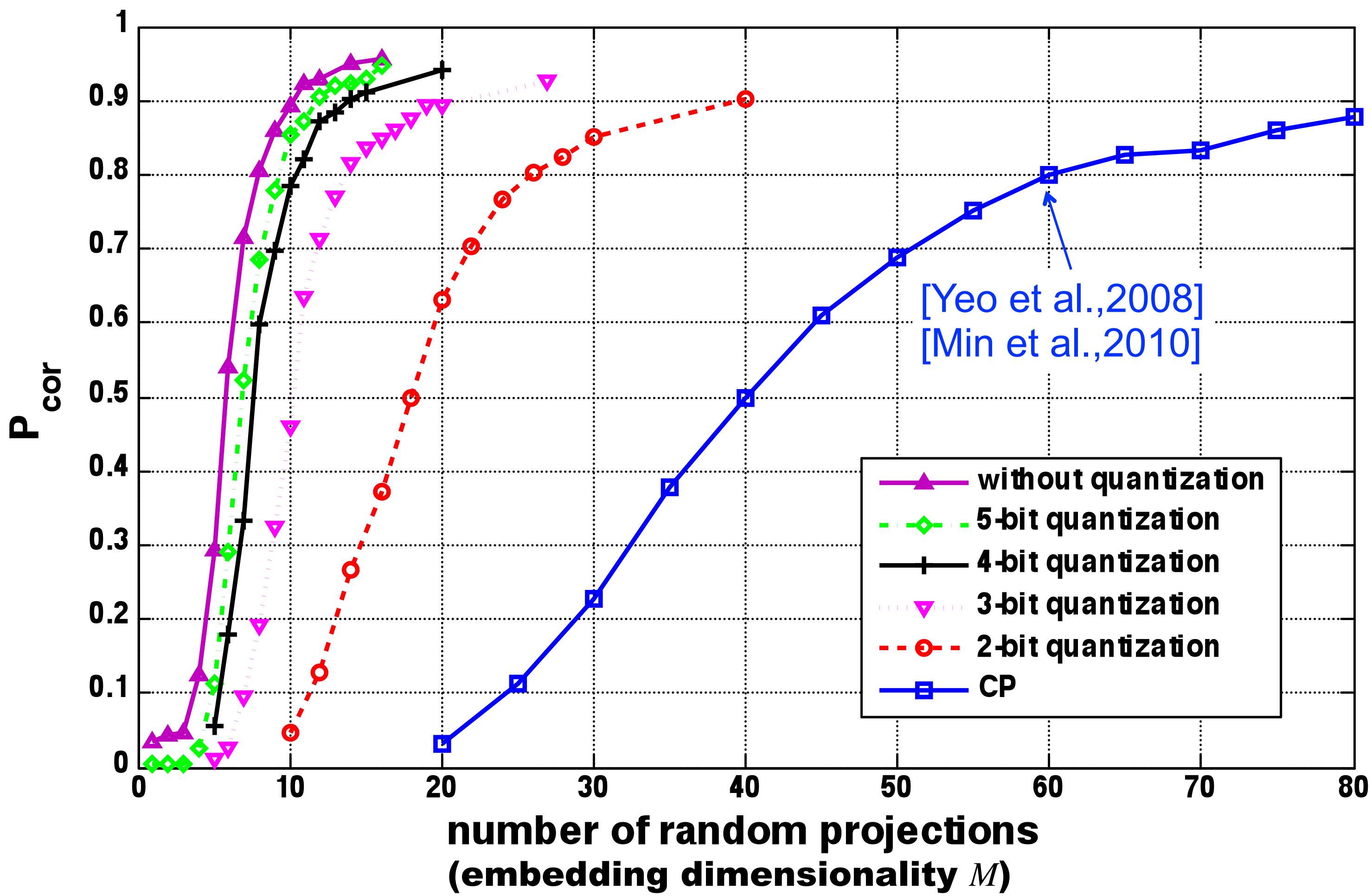
IN PRACTICE

ZuBuD: Zurich Buildings Database

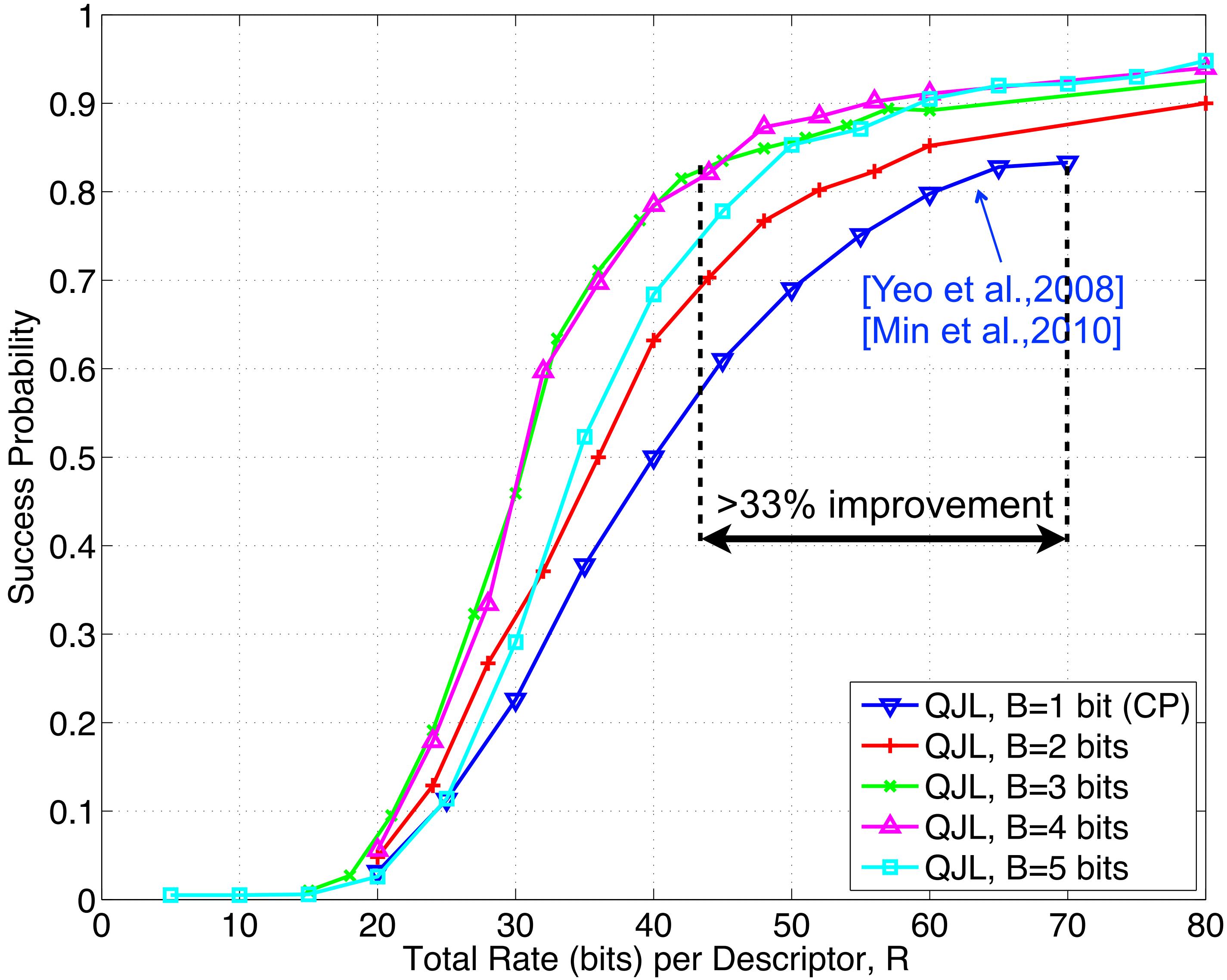
1005 images: 201 buildings from 5 viewpoints each
804 images (4 viewpoints per building) in server
201 query images (1 viewpoint per building)



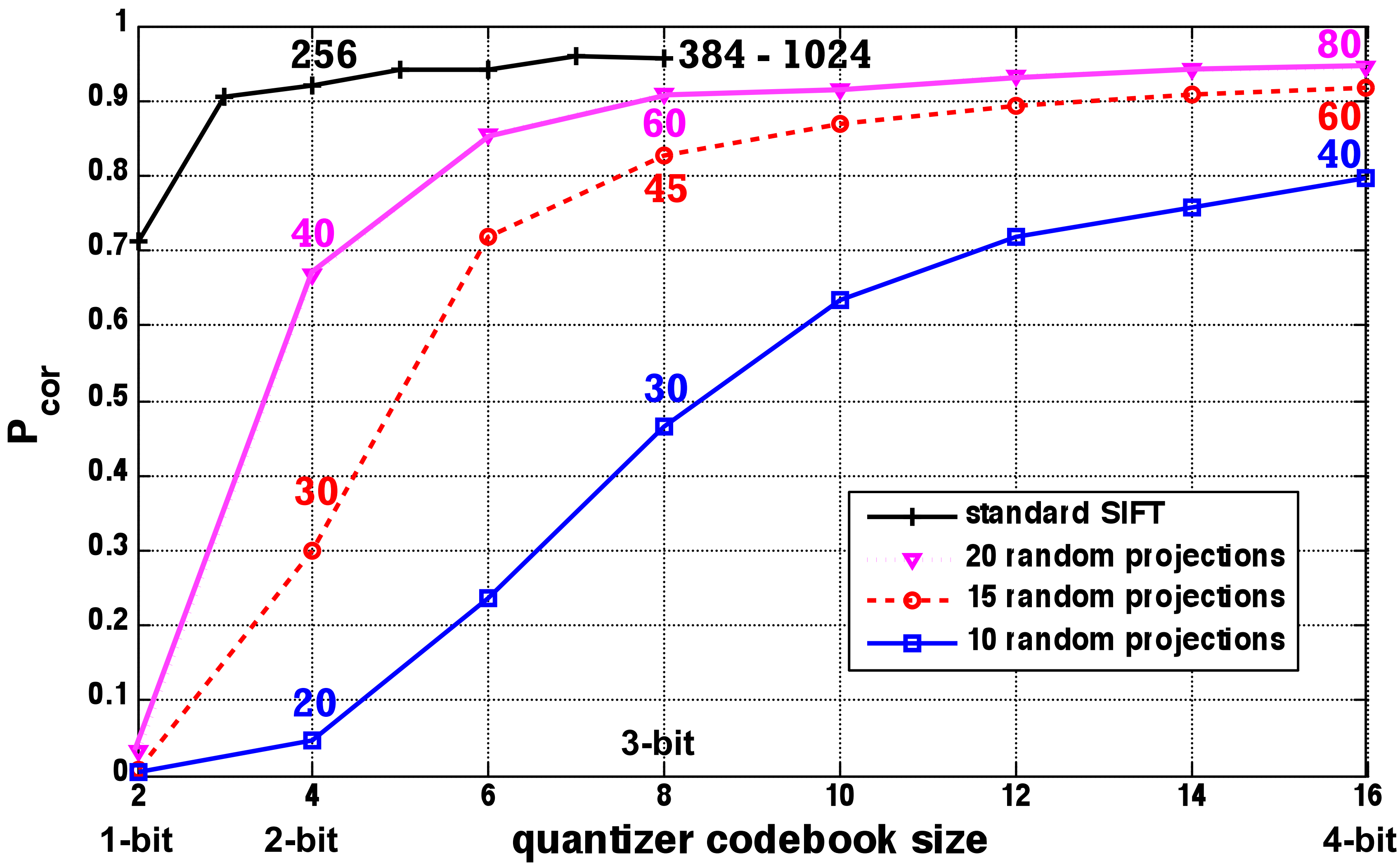
Success Probability



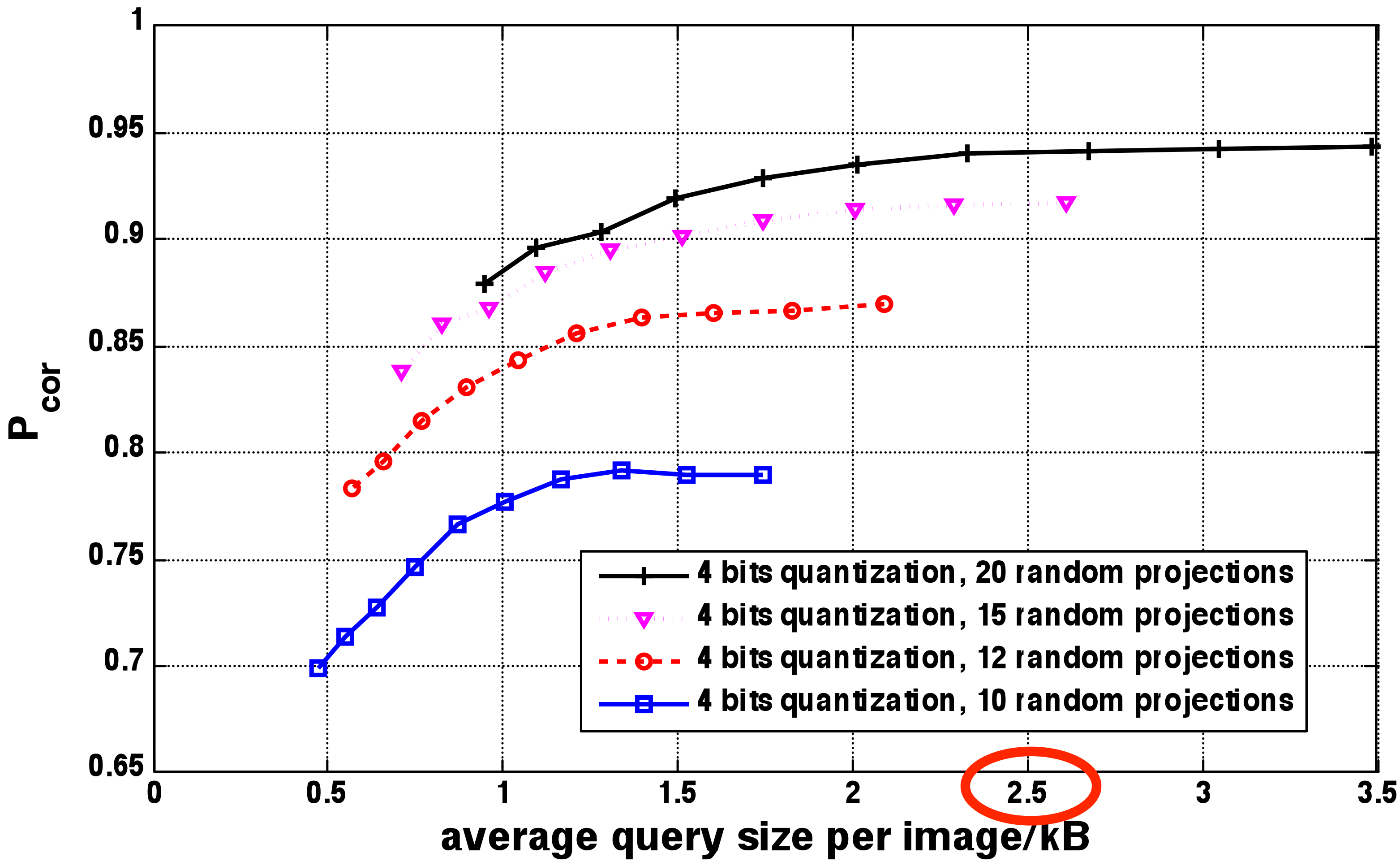
Success Probability with Fixed Rate



Performance Compared to Plain SIFT

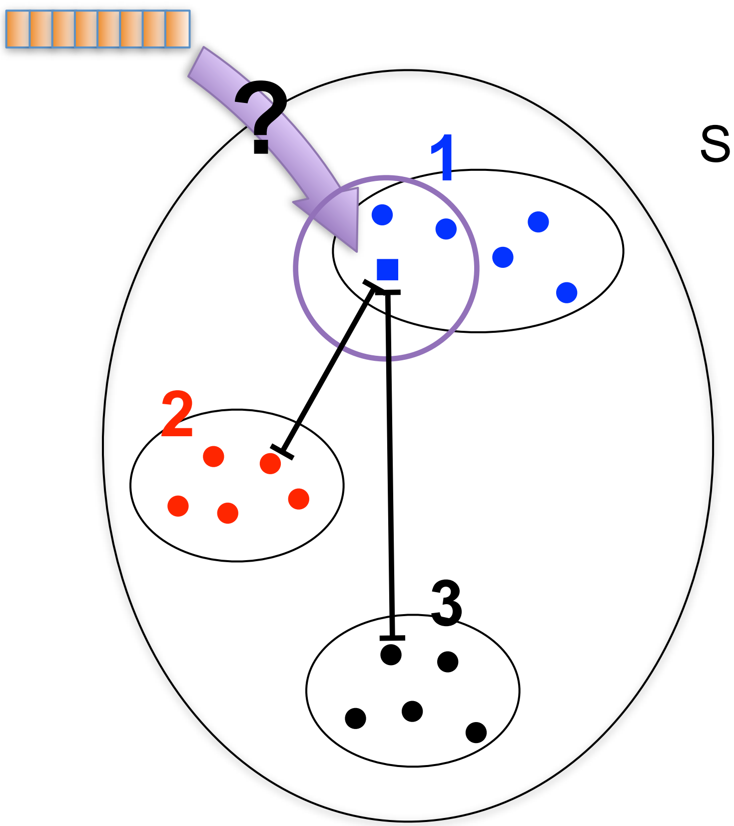


Performance Compared to JPEG



JPEG images at 80 QF need **58.5kB** on average

Information Scalability

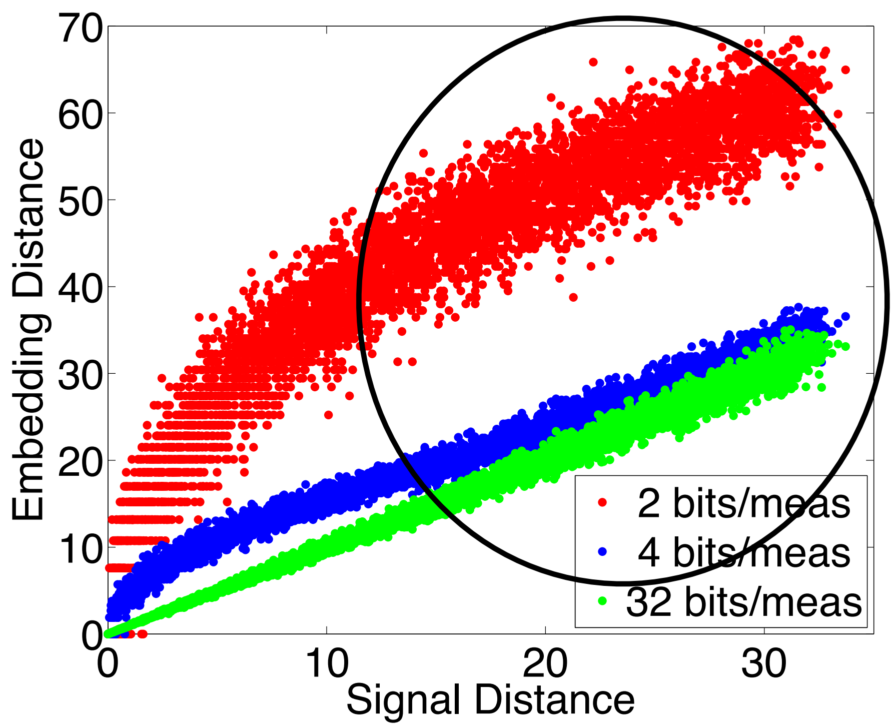


Inference relies on clusters of signals

Large distances not necessary to determine clusters and nearest neighbors

Should not spend bits encoding large distances!

But how?

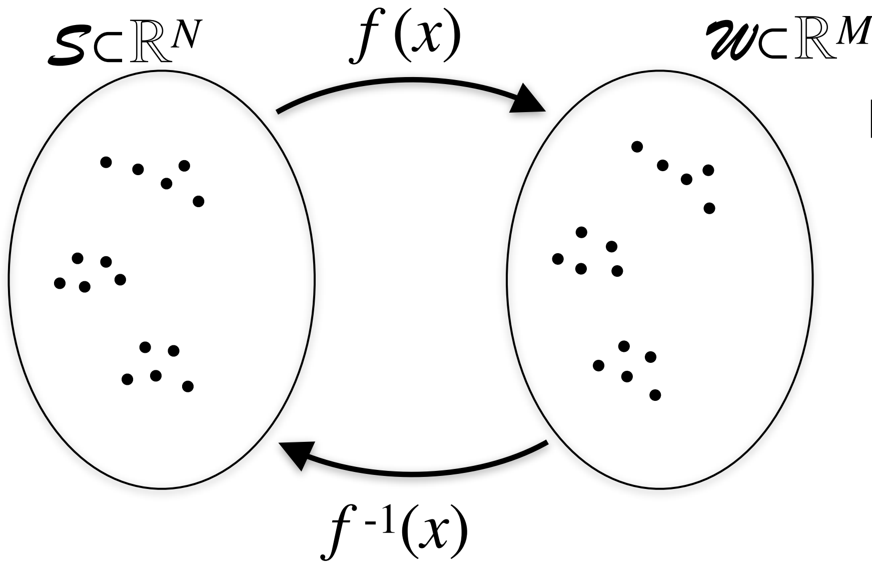


GENERAL EMBEDDING DESIGN

Generalized Embedding Maps

Original space

Distance metric: $d_{\mathcal{S}}$



Embedding space

Embed in $\mathcal{W}$

Distance metric: $d_{\mathcal{W}}$

Assume we can construct a **distance map** $g(\cdot)$

For all x,y in $\mathcal{S}$:

$$
\begin{aligned}
(1-\epsilon)g(d_{\mathcal{S}}(\mathbf{x},\mathbf{x}')) - \delta \leq \\
d_{\mathcal{W}}(f(\mathbf{x}), f(\mathbf{x}')) \\
\leq (1+\epsilon)g(d_{\mathcal{S}}(\mathbf{x},\mathbf{x}')) + \delta
\end{aligned}
$$

Embedding Analysis

For all x,y in $\mathcal{S}$:

$$(1-\epsilon)g(d_{\mathcal{S}}(\mathbf{x},\mathbf{x}')) - \delta \leq d_{\mathcal{W}}(f(\mathbf{x}), f(\mathbf{x}')) \leq (1+\epsilon)g(d_{\mathcal{S}}(\mathbf{x},\mathbf{x}')) + \delta$$

Given two signal **embeddings**, $\mathbf{x}, \mathbf{x}' \Rightarrow f(\mathbf{x}), f(\mathbf{x}')$

What is the distance of the signals $\mathbf{x}, \mathbf{x}'$?

$$d_{\mathcal{W}}(f(\mathbf{x}), f(\mathbf{x}')) \approx g(d_{\mathcal{S}}(\mathbf{x},\mathbf{x}'))$$

$$\Rightarrow \widetilde{d}_{\mathcal{S}} = g^{-1}(d_{\mathcal{W}}(f(\mathbf{x}, f(\mathbf{x}'))$$

Embedding Analysis

For all x,y in $\mathcal{S}$:

$$(1-\epsilon)g(d_{\mathcal{S}}(\mathbf{x},\mathbf{x}')) - \delta \leq d_{\mathcal{W}}(f(\mathbf{x}), f(\mathbf{x}')) \leq (1+\epsilon)g(d_{\mathcal{S}}(\mathbf{x},\mathbf{x}')) + \delta$$

Distance estimate:

$$\widetilde{d}_{\mathcal{S}} = g^{-1}(d_{\mathcal{W}}(f(\mathbf{x}, f(\mathbf{x}'))$$

Given distance estimate, what is the **ambiguity?**

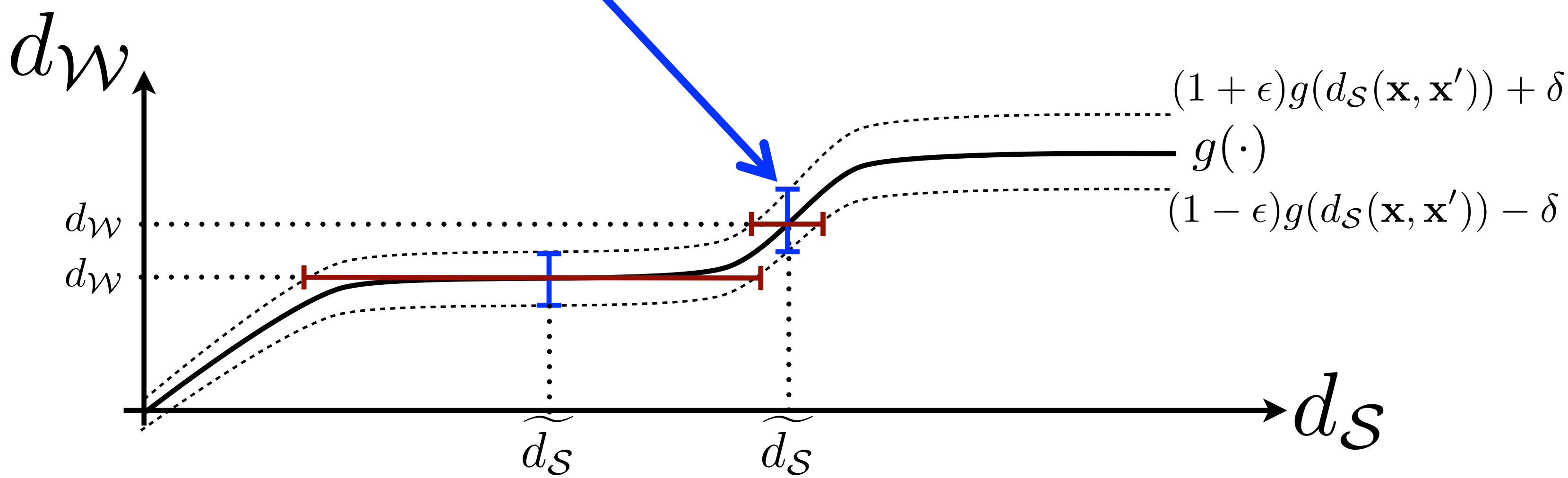
$$\left| d_{\mathcal{S}}(\mathbf{x},\mathbf{x}') - \widetilde{d}_{\mathcal{S}} \right| \lesssim \frac{\delta + \epsilon d_{\mathcal{W}}(f(\mathbf{x}), f(\mathbf{x}'))}{g'(\widetilde{d}_{\mathcal{S}})}$$

Note dependence on slope!

Embedding Analysis

For all x,y in $\mathcal{S}$:

$$(1-\epsilon)g(d_{\mathcal{S}}(\mathbf{x},\mathbf{x}'))-\delta \leq d_{\mathcal{W}}(f(\mathbf{x}),f(\mathbf{x}')) \leq (1+\epsilon)g(d_{\mathcal{S}}(\mathbf{x},\mathbf{x}'))+\delta$$



$$\left|d_{\mathcal{S}}(\mathbf{x},\mathbf{x}')-\widetilde{d_{\mathcal{S}}}\right| \lesssim \frac{\delta+\epsilon d_{\mathcal{W}}(f(\mathbf{x}),f(\mathbf{x}'))}{g'(\widetilde{d_{\mathcal{S}}})}$$

Note dependence on slope!

Embedding Design

Q: Can we design embeddings?

A: Yes. We start with a **random matrix** $\mathbf{A} \in \mathbb{R}^{M \times N}$,

a **periodic function** $h(t) = h(t+1)$

and **random** i.i.d., uniform **dither** $\mathbf{w} \in [0, 1)$

$$\mathbf{y} = h(\mathbf{A}\mathbf{x} + \mathbf{w})$$

Fourier series coefficients of $h(\cdot)$: $\quad H_k$

Also, assume bounded: $\bar{h} = \sup_t h(t) - \inf_t h(t)$

Distance Map

$$\mathbf{A} \in \mathbb{R}^{M \times N} \text{ i.i.d., Gaussian, variance } \sigma^2$$

$$\mathbf{w} \in [0, 1) \text{ i.i.d, uniform}$$

$$h(t) = h(t+1) \qquad \bar{h} = \sup_t h(t) - \inf_t h(t)$$

Fourier series coefficients of $h(\cdot)$: H_k

Theorem (Embedding Design)

Consider a set $\mathcal{S}$ of Q points in $\mathbb{R}^N$, measured using $\mathbf{y} = h(\mathbf{A}\mathbf{x} + \mathbf{w})$, with $\mathbf{A}$, $\mathbf{w}$, and $h(t)$ as above. With failure probability $P_F \leq 2Q^2 e^{-2M\frac{\delta^2}{\bar{h}^4}}$ the following holds

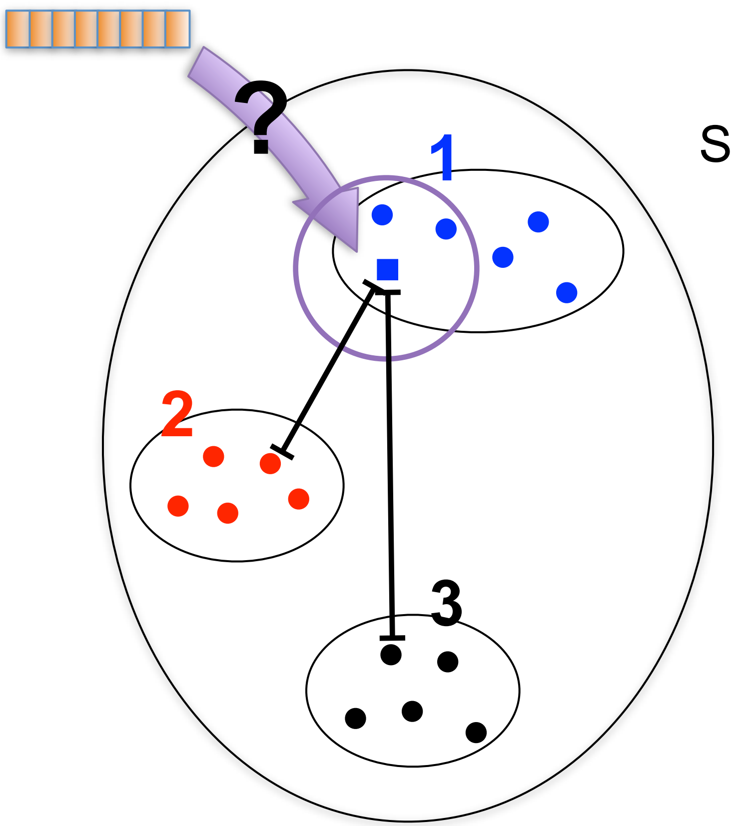
$$g(\|\mathbf{x} - \mathbf{x}'\|_2) - \delta \leq \frac{1}{M}\|\mathbf{y} - \mathbf{y}'\|_2^2 \leq g(\|\mathbf{x} - \mathbf{x}'\|_2) + \delta$$

for all pairs $\mathbf{x}, \mathbf{x}' \in \mathcal{S}$ and corresponding measurements $\mathbf{y}, \mathbf{y}'$.

With distance map:

$$g(d) = 2\sum_k |H_k|^2 \left(1 - e^{-\frac{1}{2}(\sigma d k)^2}\right)$$

Information Scalability

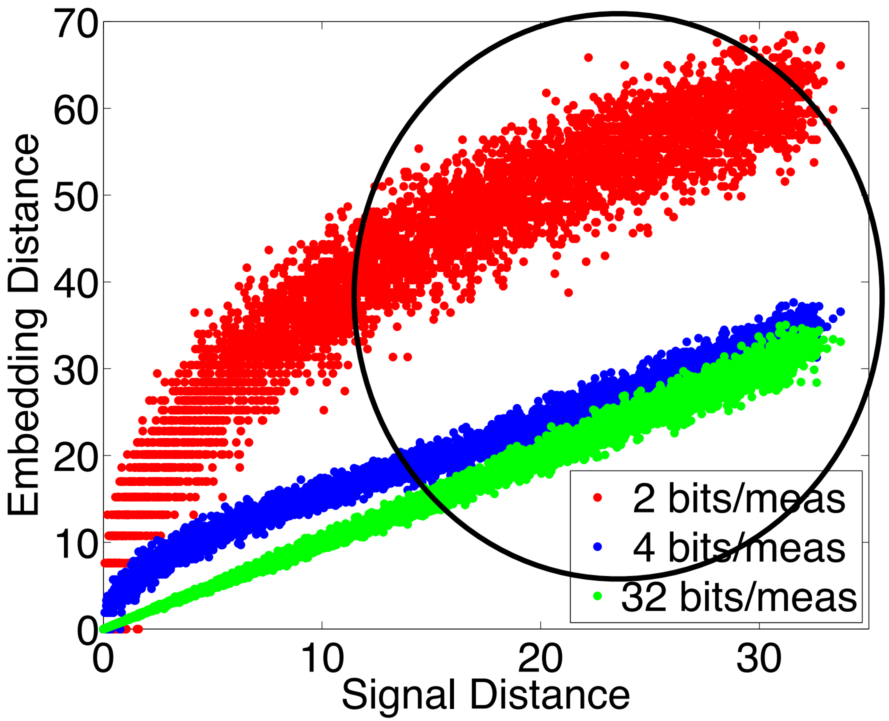


Inference relies on clusters of signals

Large distances not necessary to determine clusters and nearest neighbors

Should not spend bits encoding large distances!

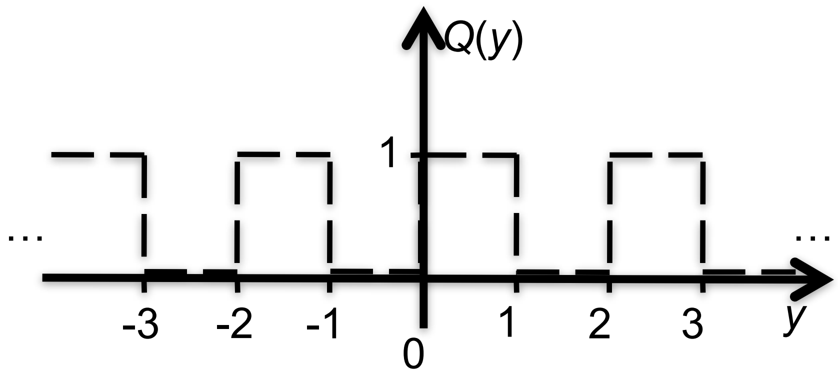
But how?

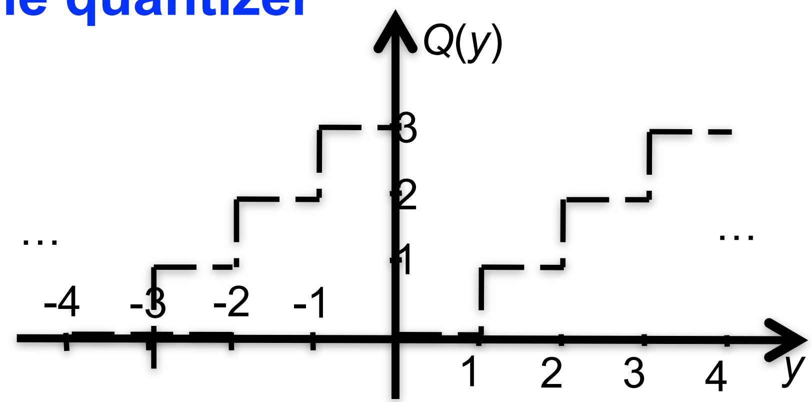


UNIVERSAL QUANTIZED EMBEDDINGS

Rate-Efficient Scalar Quantization

Solution: **Modify the quantizer**





Non-monotonic quantizer: Multiple intervals quantize to same value

(Focus on 1-bit quantizer today)

measurements
(w/ i.i.d. gaussian matrix)

dither
(i.i.d. uniform)

$$q_m = Q\left(\frac{\langle \mathbf{x}, \mathbf{a}_m \rangle + w_m}{\Delta_m}\right), \qquad \mathbf{q} = Q(\Delta^{-1}(\mathbf{A}\mathbf{x} + \mathbf{w}))$$

scalar quantizer
(non-monotonic)

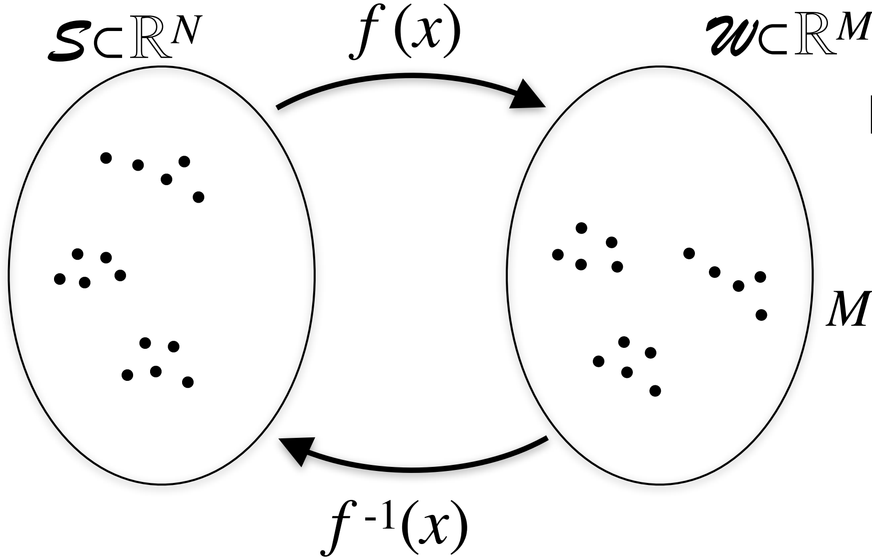
scaling/precision parameter
(Δ_m=Δ, same for all measurements)

Embedding Properties

Original space

Distance metric: ℓ_2

P points in $\mathbb{R}^N$



Embedding space

Embed in $\{0,1\}^M$

Hamming distance

$M = O(\log P)$ dimensions

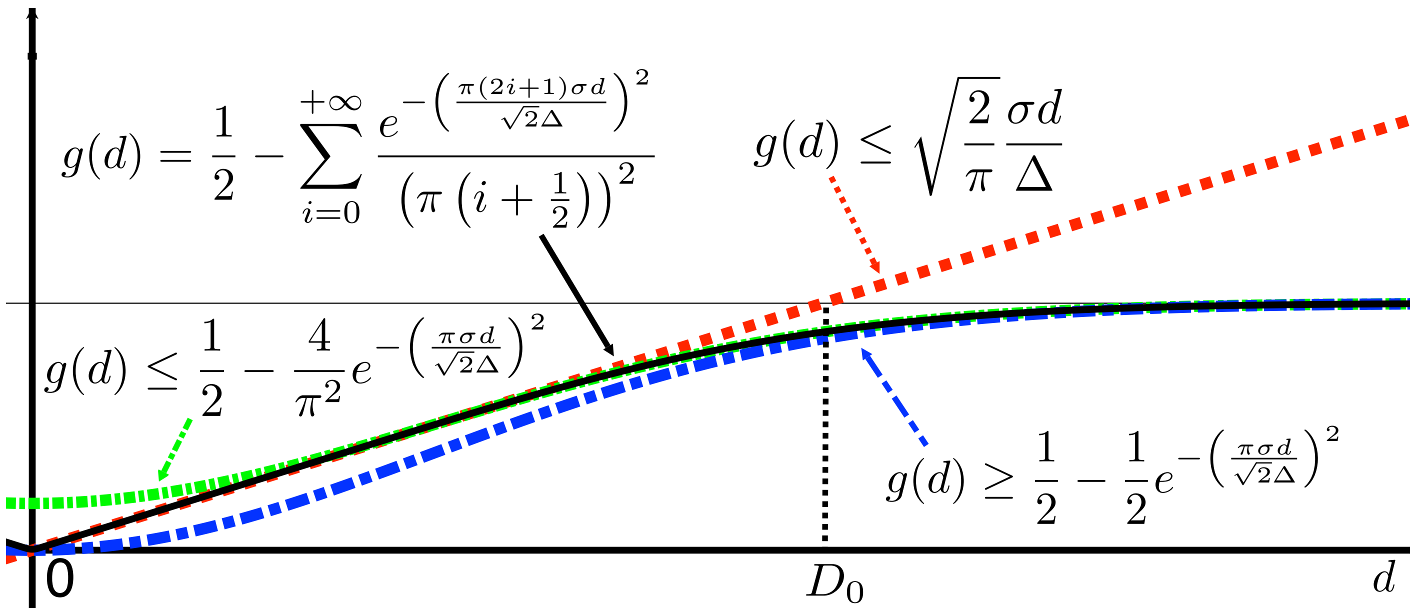
For all x,y in $\mathcal{S}$:

$$g(d) - \delta \leq d_H\left(f(x) - f(y)\right) \leq g(d) + \delta$$

$$g(d) = \frac{1}{2} - \sum_{i=0}^{+\infty} \frac{e^{-\left(\frac{\pi(2i+1)\sigma d}{\sqrt{2}\Delta}\right)^2}}{\left(\pi\left(i+\frac{1}{2}\right)\right)^2}$$

Error Behavior

$$g(d) - \delta \le d_H\left(f(x) - f(y)\right) \le g(d) + \delta$$



Distance estimate: $\widetilde{d} = g^{-1}\left(d_H\left(f(x), f(y)\right)\right)$

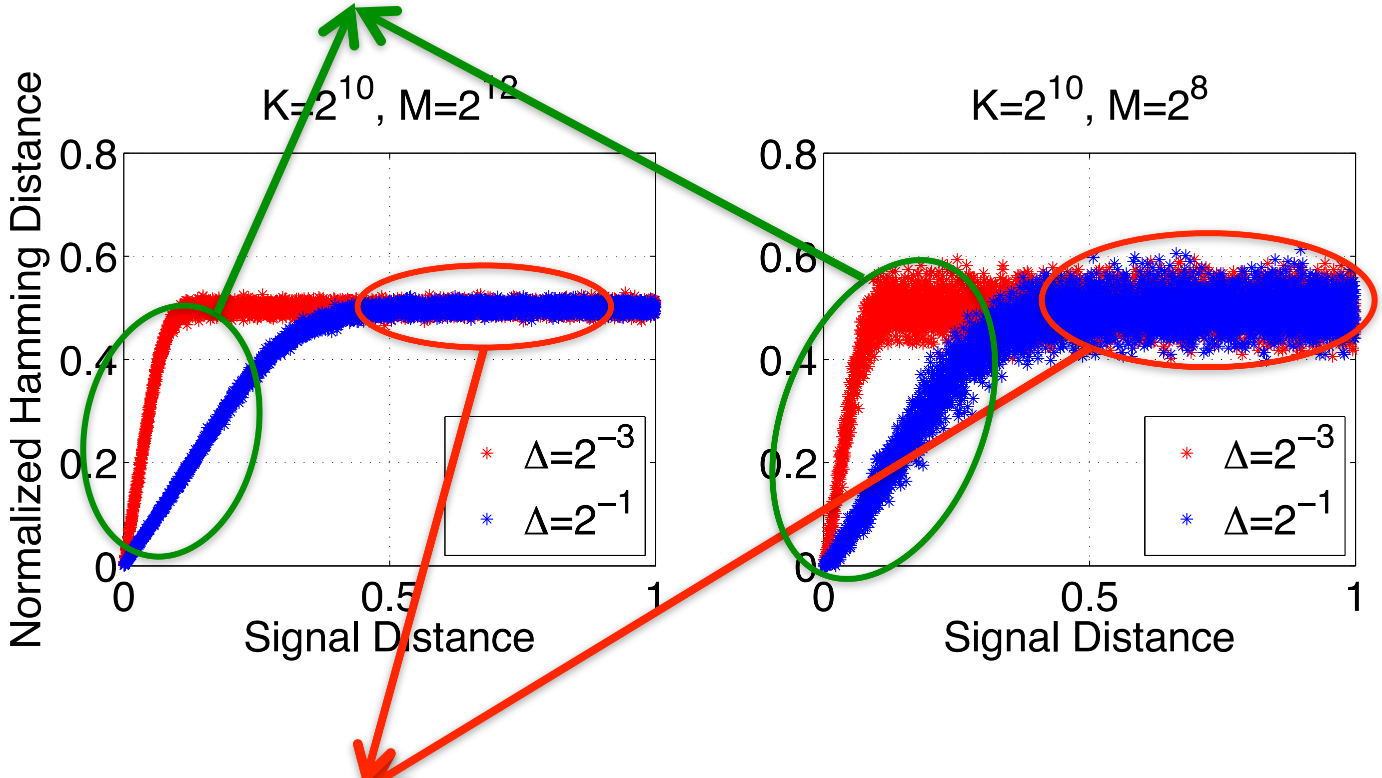
Estimate ambiguity: $\widetilde{d} - \frac{\delta}{g'(\widetilde{d})} \lesssim d \lesssim \widetilde{d} + \frac{\delta}{g'(\widetilde{d})}$

Properties (slope) controlled by choice of Δ

Error Behavior

$$g(d) - \delta \leq d_H\left(f(x) - f(y)\right) \leq g(d) + \delta$$

“Linear” region: $\ell_2 \propto d_H$, slope controlled by Δ

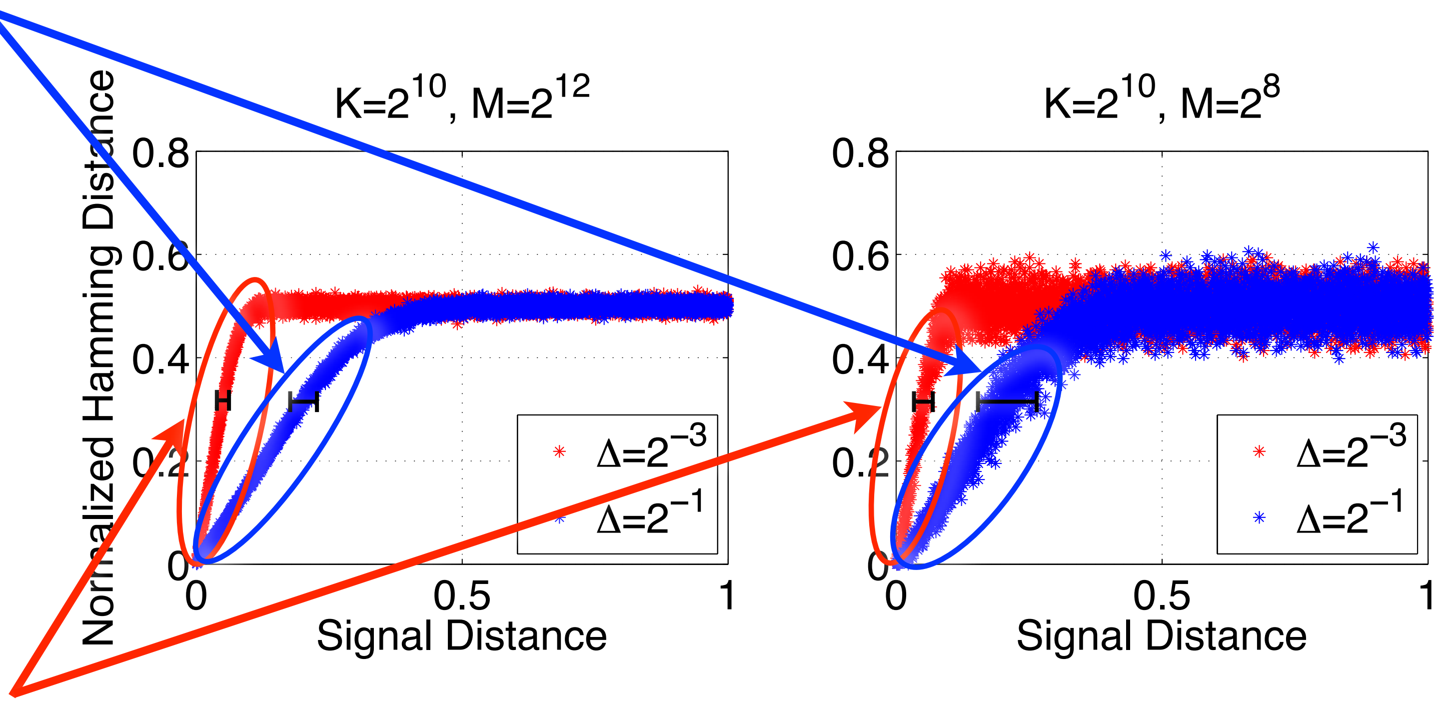


“Flat” region: no distance information

Error Behavior

$$g(d) - \delta \leq d_H\left(f(x) - f(y)\right) \leq g(d) + \delta$$

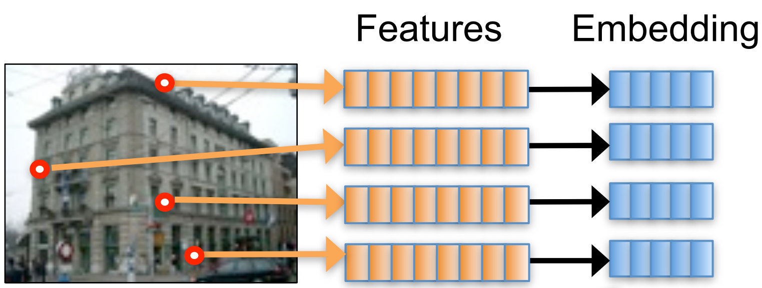
Large Δ: small slope, more ambiguity, preserves larger distances



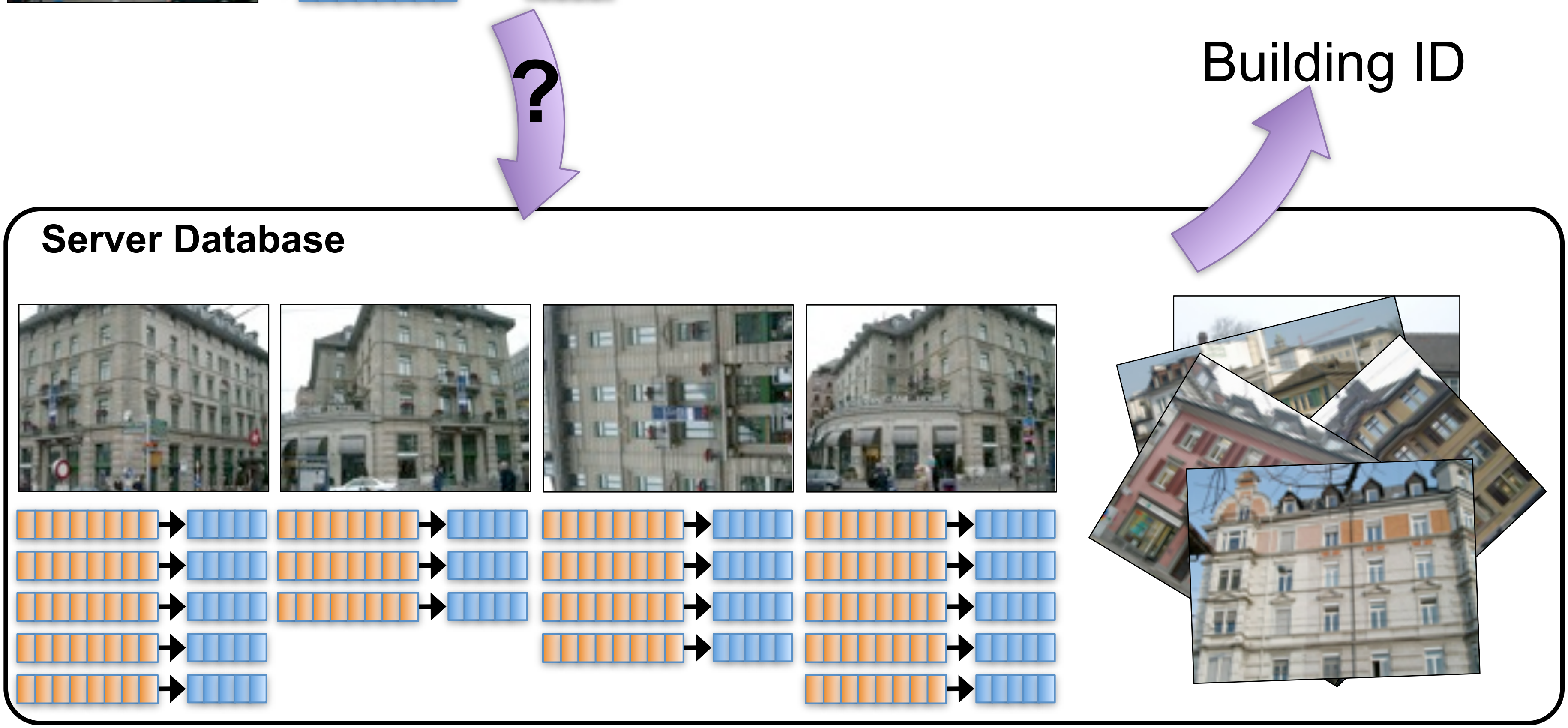
Small Δ: large slope, less ambiguity, preserves smaller distances

IN PRACTICE

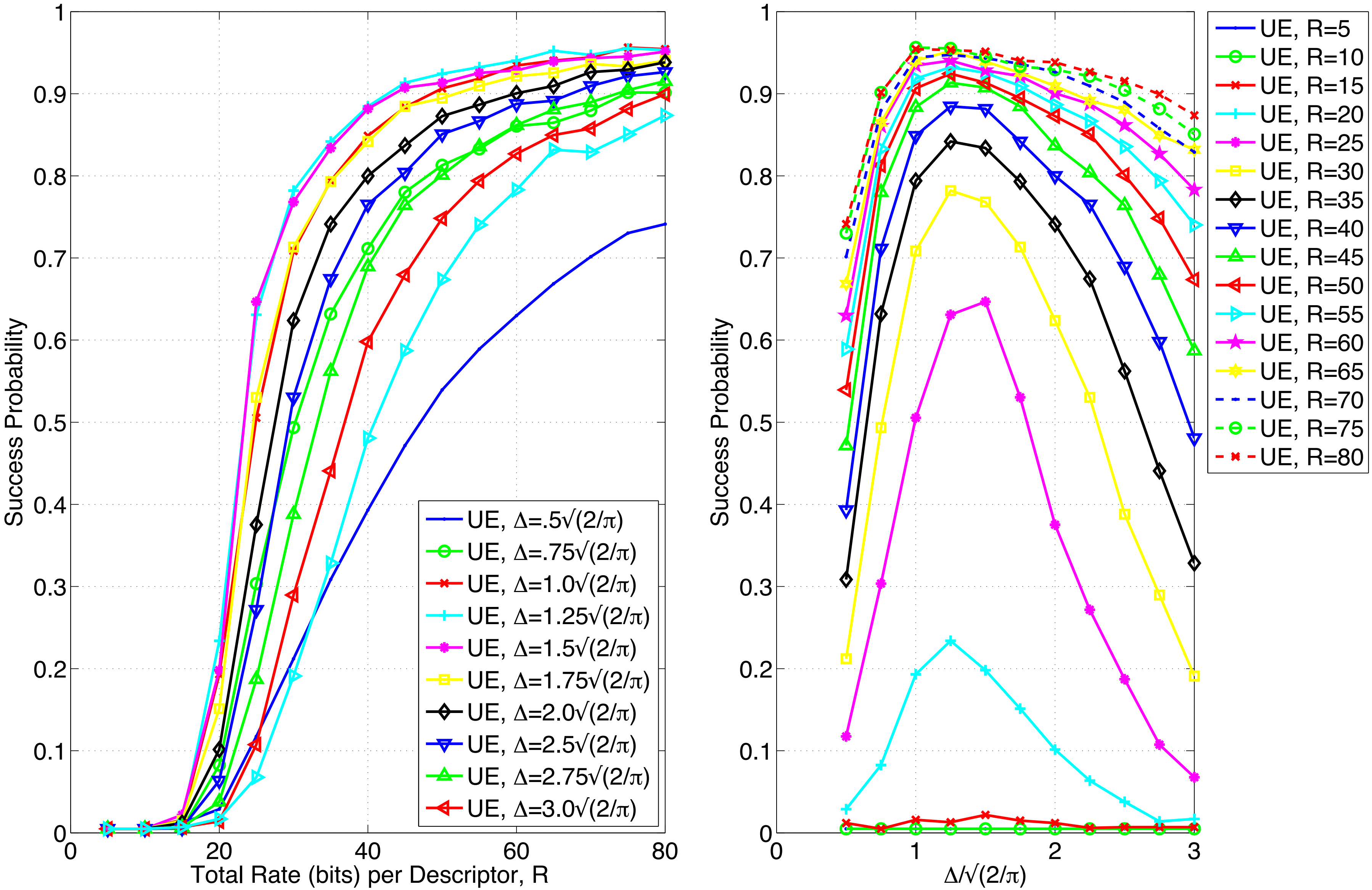
ZuBuD: Zurich Buildings Database



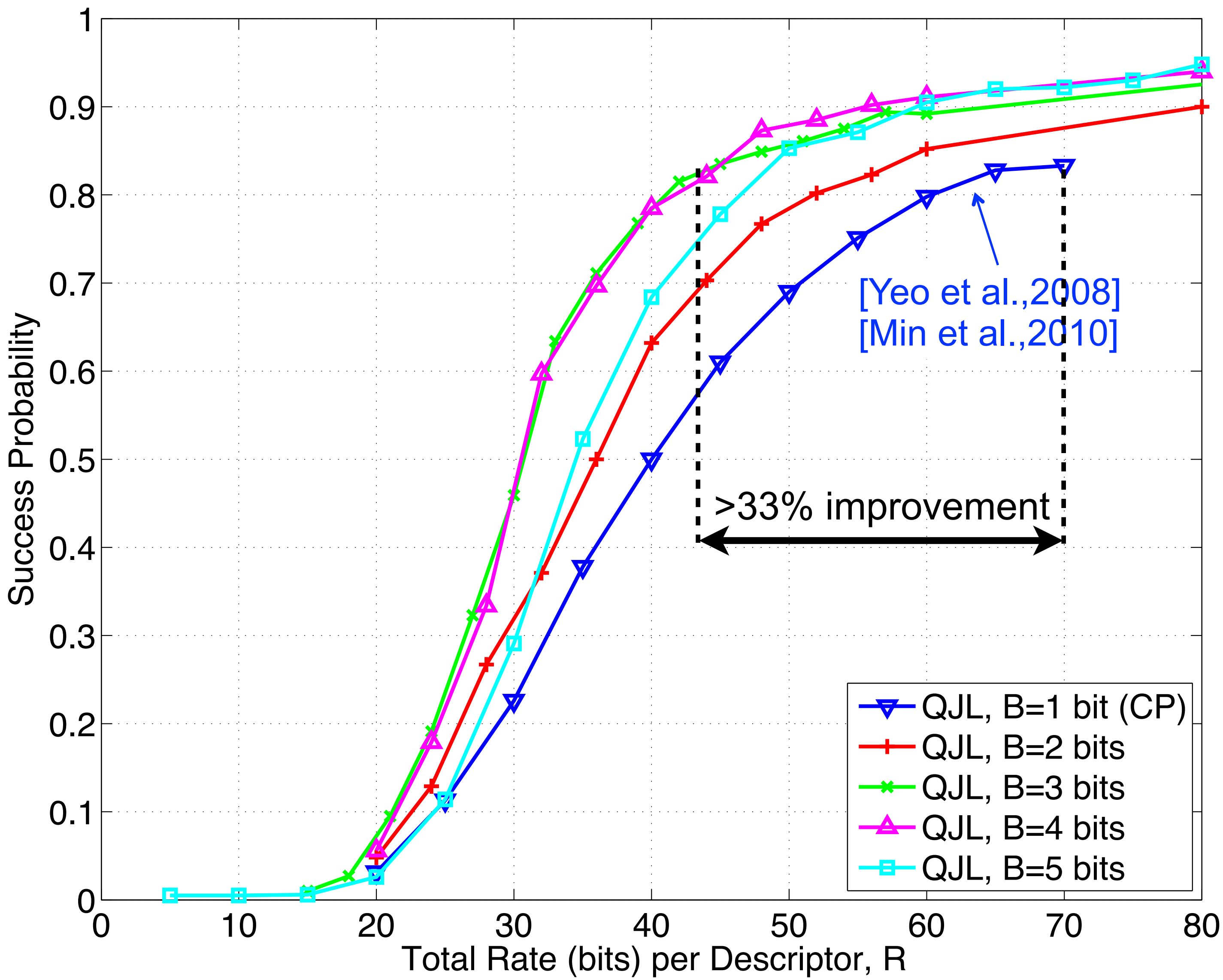
1005 images: 201 buildings from 5 viewpoints each
804 images (4 viewpoints per building) in server
201 query images (1 viewpoint per building)



In practice

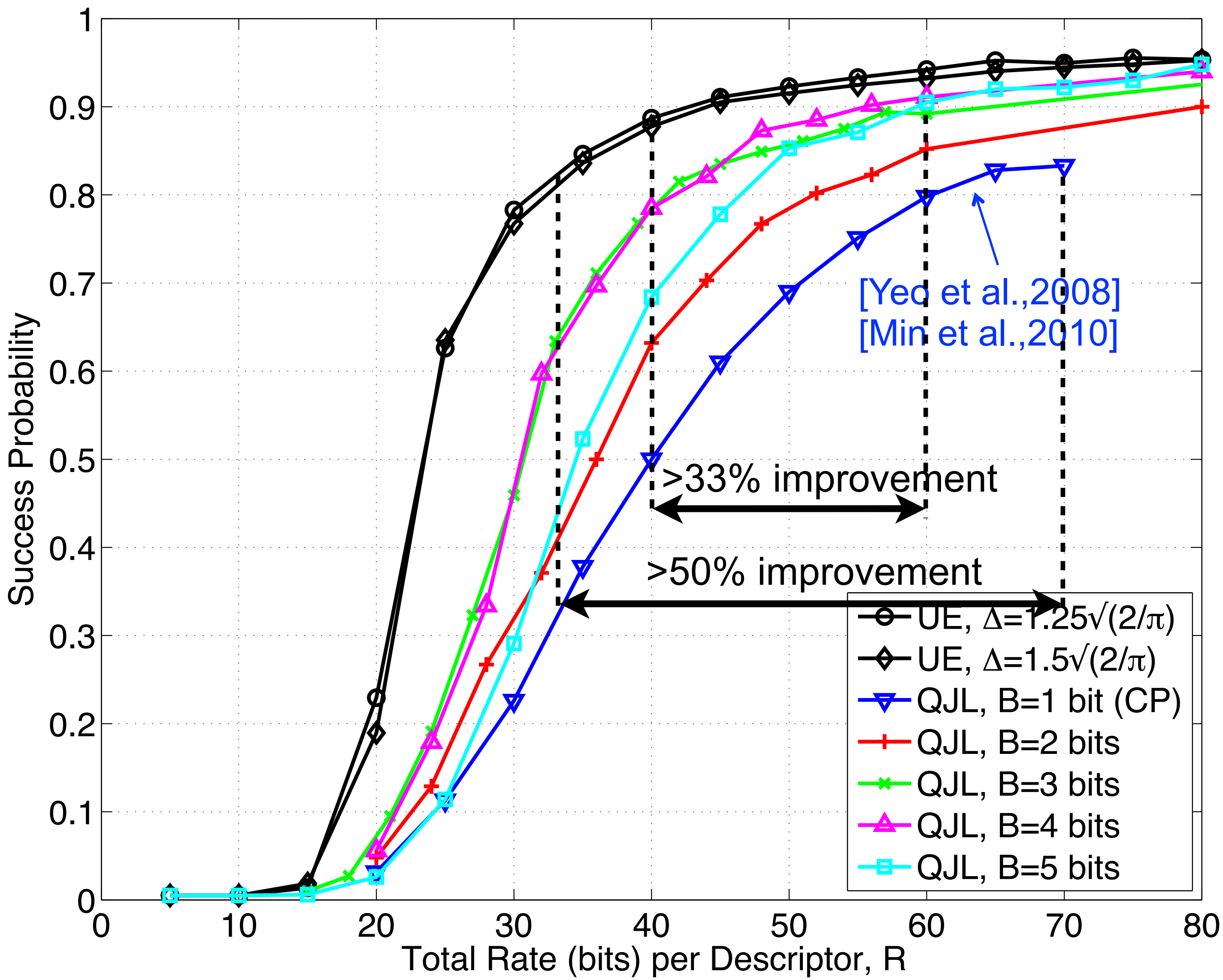

Success Probability
Total Rate (bits) per Descriptor, R
UE, Δ=.5√(2/π)
UE, Δ=.75√(2/π)
UE, Δ=1.0√(2/π)
UE, Δ=1.25√(2/π)
UE, Δ=1.5√(2/π)
UE, Δ=1.75√(2/π)
UE, Δ=2.0√(2/π)
UE, Δ=2.5√(2/π)
UE, Δ=2.75√(2/π)
UE, Δ=3.0√(2/π)
Success Probability
Δ/√(2/π)
UE, R=5
UE, R=10
UE, R=15
UE, R=20
UE, R=25
UE, R=30
UE, R=35
UE, R=40
UE, R=45
UE, R=50
UE, R=55
UE, R=60
UE, R=65
UE, R=70
UE, R=75
UE, R=80

In practice



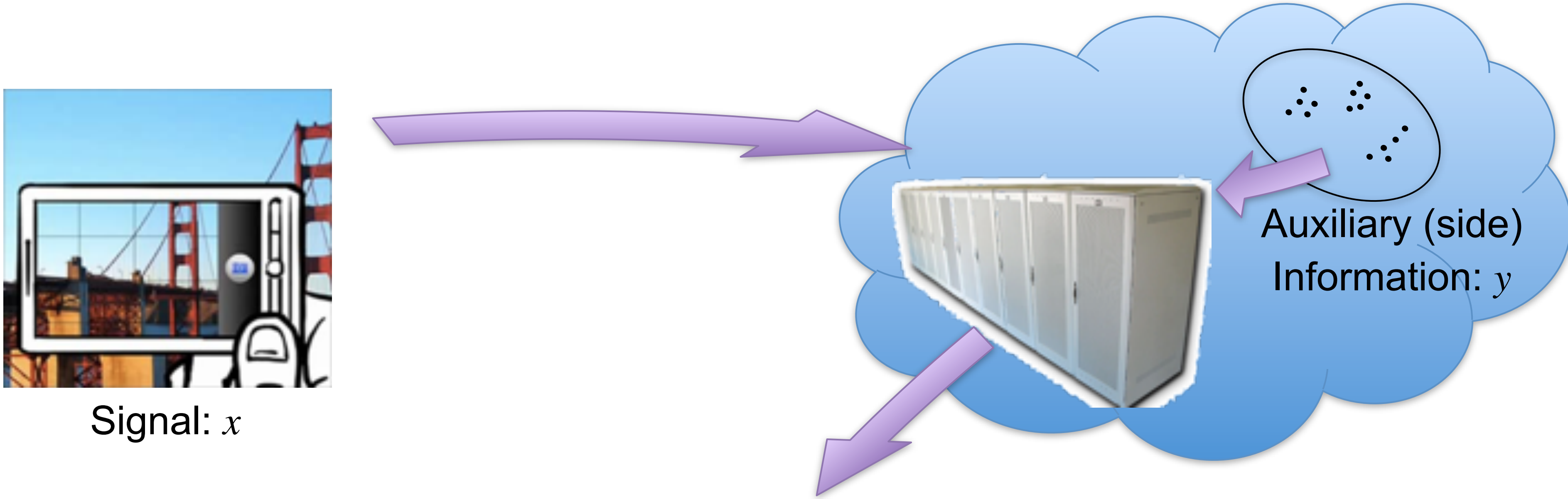
Success Probability
Total Rate (bits) per Descriptor, R
[Yeo et al.,2008]
[Min et al.,2010]
>33% improvement
QJL, B=1 bit (CP)
QJL, B=2 bits
QJL, B=3 bits
QJL, B=4 bits
QJL, B=5 bits

In practice



Success Probability
Total Rate (bits) per Descriptor, R
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
10
20
30
40
50
60
70
80
>33% improvement
>50% improvement
[Yeo et al.,2008]
[Min et al.,2010]
UE, Δ=1.25√(2/π)
UE, Δ=1.5√(2/π)
QJL, B=1 bit (CP)
QJL, B=2 bits
QJL, B=3 bits
QJL, B=4 bits
QJL, B=5 bits

Looking Ahead: The Big Picture



Output: $g(x,y)$

Information Scalable Coding:

Encode/embed the signal for general functions $g(\cdot,\cdot)$?

Preserving whole signal usually not necessary!

Solution known only for some special cases

Summary

- Embeddings a big step towards information scalability
  - Very effective for coding signal distances
  - Very efficient for big data and distributed systems
  - Very promising for quantization and distributed coding

- Embeddings exist for other distance metrics
  - Angle/correlation of signals (1-bit CS and phase embeddings)
  - Edit distance, Earth mover distance [Indyk et al.]
  - How to properly exploit/quantize them?

- General open problems
  - Embeddings/codings for function computation
  - Information scalability in more general inference problems

Questions/Comments?

petros@boufounos.com **http://boufounos.com**