JOHNS HOPKINS UNIVERSITY

Dictionaries – Mixtures of Gaussians

– Mini-Epitomes.

Alan Yuille
JHU

Local Image Patches

- Analyze properties of local image patches.
- Get a lot of image patches.
- Apply techniques:

(i) Dictionaries – *k*-means.

(ii) Mini-epitomes.

JOHNS HOPKINS UNIVERSITY

Extreme Sparsity **Matched Filters**

Set of basis function: $\{b_i(x)\}$

Represent each image by one basis function only

$$E[\alpha] = \sum_x \left| I(x) - \sum_i \alpha_i b_i(x) \right|^2$$ with constant only one $\alpha_i \neq 0$

Algorithm estimate $\hat{\alpha} = \arg\min E[\alpha]$

Set $\hat{\alpha}_i = \arg\min \sum_x |I(x) - \alpha_i b_i(x)|^2 = \arg\min \sum_x I(x) b_i(x)$ ⬅ $\sum \{b_i(x)\}^2 = 1$

Choose $\hat{i} = \min_i \sum_x |I(x) - \hat{\alpha}_i b_i(x)|^2$ ➡ Set $\alpha_{\hat{i}} = \hat{\alpha}_i$

$\alpha_j = 0$ otherwise

$$\sum \{b_i(x)\}^2 = 1$$

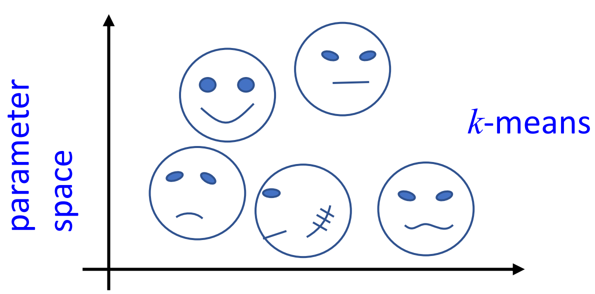
Minimize $\quad E[b,\alpha] = \frac{1}{|\Lambda|}\sum_{\mu\in\Lambda}\sum_x \left\{ I^\mu(x) - \sum_i \alpha_i^\mu b_i(x) \right\}^2$

with constraint that only one α_i^μ is non-zero for each μ

How to minimize?

➡ Convert this to k-means clustering

Requires normalizing each image $I^\mu(x) \to \dfrac{I^\mu(x)}{\sqrt{\sum_x \{I^\mu(x)\}^2}}$ so that $\sum_x \{I^\mu(x)\}^2 = 1$



➡ Implies that the best $\alpha_i^\mu = 1$

Supplement: k-means Algorithm

- Deterministic k-means
  1. Initialize a partition $\{D_a^0 : a = 1, \ldots, k\}$
     - E.g. Randomly choose points x and put them into set, $D_1^0, D_2^0, \ldots, D_k^0$- so that all datapoints are in exactly one set
  2. Compute the mean of each cluster D_a, $m_a = \frac{1}{w_a} \sum_{x \in D_a} x$
  3. For $i = 1, \ldots, N$, compute $d_a(x_i) = |x_i - m_a|^2$
     - Assign x_i to cluster D_a
       s.t. $a^* = \arg\min \{d_a(x_i), \ldots, d_k(x_i)\}$
  4. Repeat steps 2 & 3 until converge

Supplement: k-means Algorithm

- Soft version of k-means: The EM algorithm
  - A 'softer' version of k-means – the Expectation-Maximization (EM) algorithm.
  - Assign datapoints x_i to each cluster with probability $(P_1, \ldots, P_k)$

1. Initialize a partition
   - E.G. randomly choose k points as centres $m_1, m_2, \ldots, m_k$
2. For $j = 1, \ldots, N$
   - Compute distances $d_a(x_j) = |x_j - m_a|^2$
   - Compute the probability that x_j belongs to D_a: $P_a(x_j) = \frac{1}{(2\pi\sigma_a^2)^{d/2}} e^{-\frac{1}{2\sigma_a^2}(x_j - m_a)^2}$
3. Compute the mean and variance for each cluster

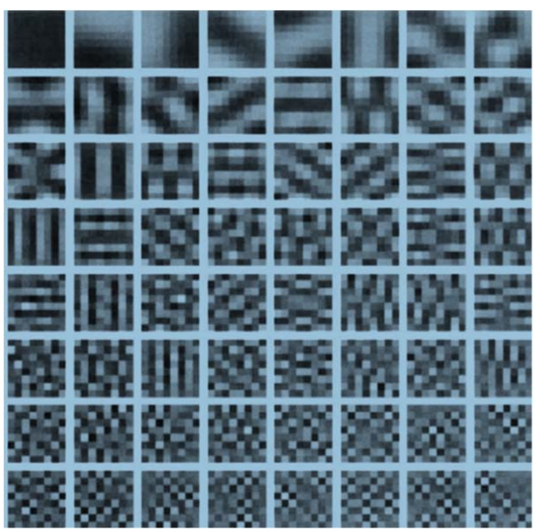
$$m_a = \tfrac{1}{|D_a|}\sum_{x\in D_a} xP_a(x) \qquad \sigma_a^2 = \tfrac{1}{|D_a|}\sum_{x\in D_a} (x-m_a)^2 P_a(x)$$

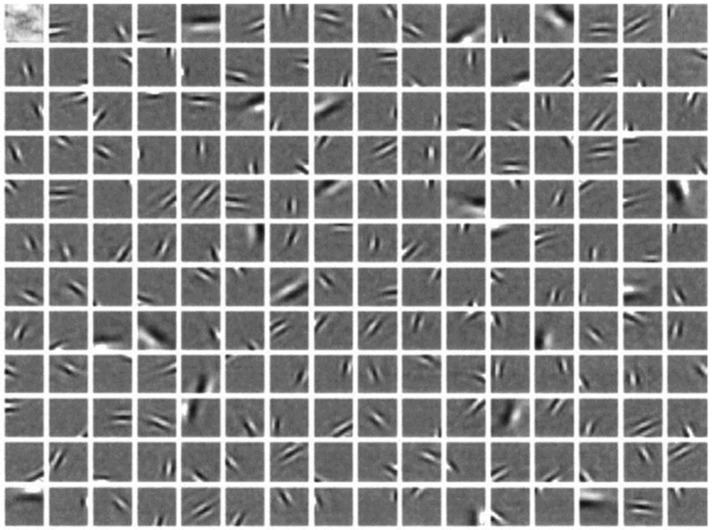
4. Repeat steps 2 & 3 until convergence

Recall PCA & Sparsity

- Shift-invariance arises both in PCA and Sparsity.

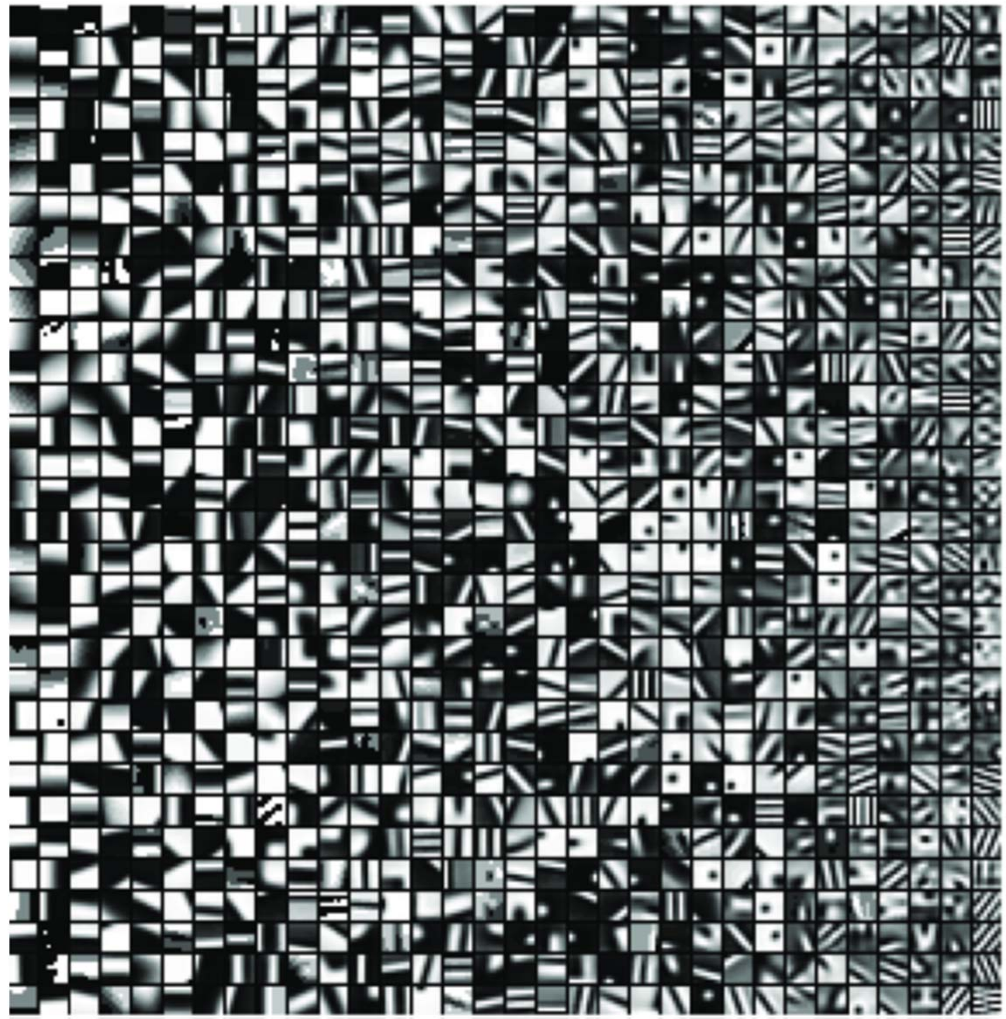
- Are we wasting bases by encoding spatial translation?





Full Sparsity

- Dictionaries of patches:
- Cluster – *k*-means.



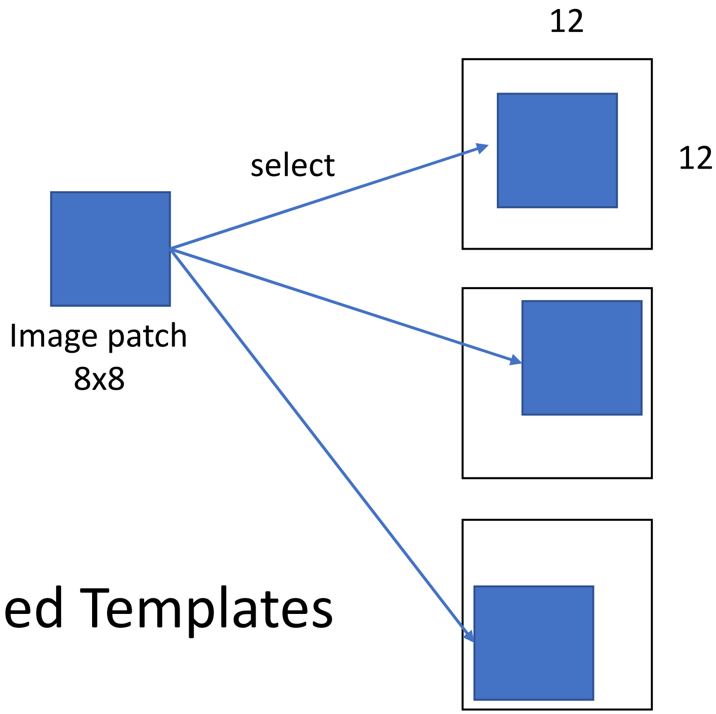
Modeling Shift

- A variants of image patches.
- Mini-Epitomes (G. Papandreou et al. CVPR 2014)

- An attempt to deal directly with shift-invariance.

JOHNS HOPKINS UNIVERSITY

Mini-Epitomes



This is like an extension of Matched Templates

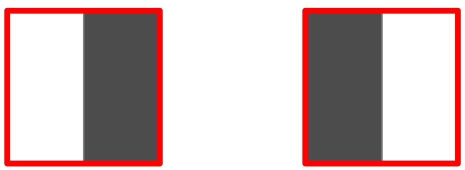
But with smarter patches

Can be learnt by the EM algorithm: extending k-means

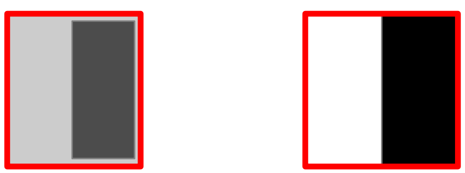
Sources of Redundancy in Patch Dictionaries

1. Same pattern, different position

2. Same pattern, opposite polarity (x2 redundancy)

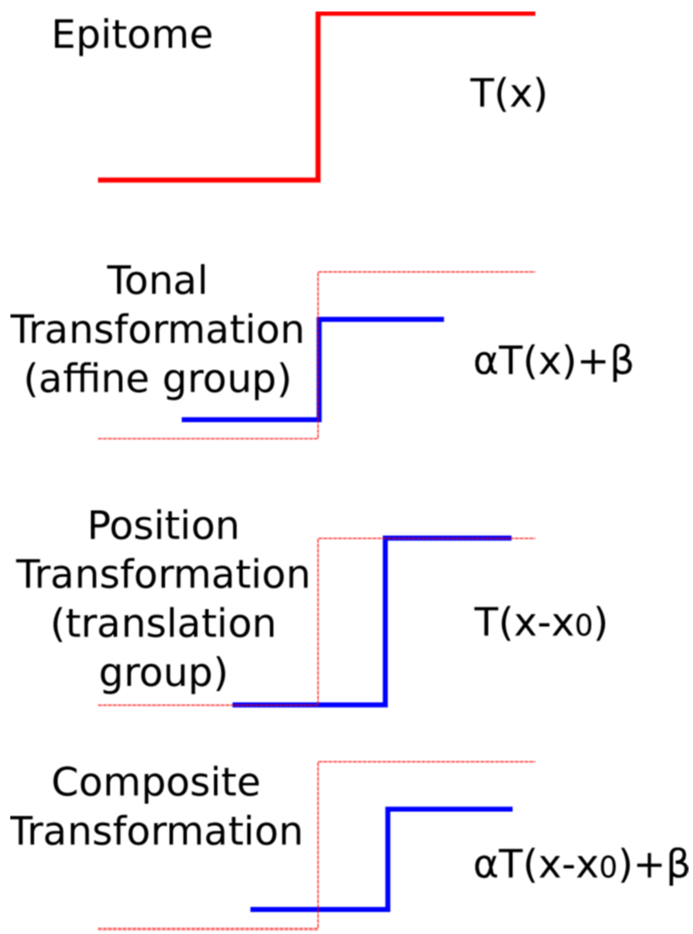


3. Same pattern, different contrast



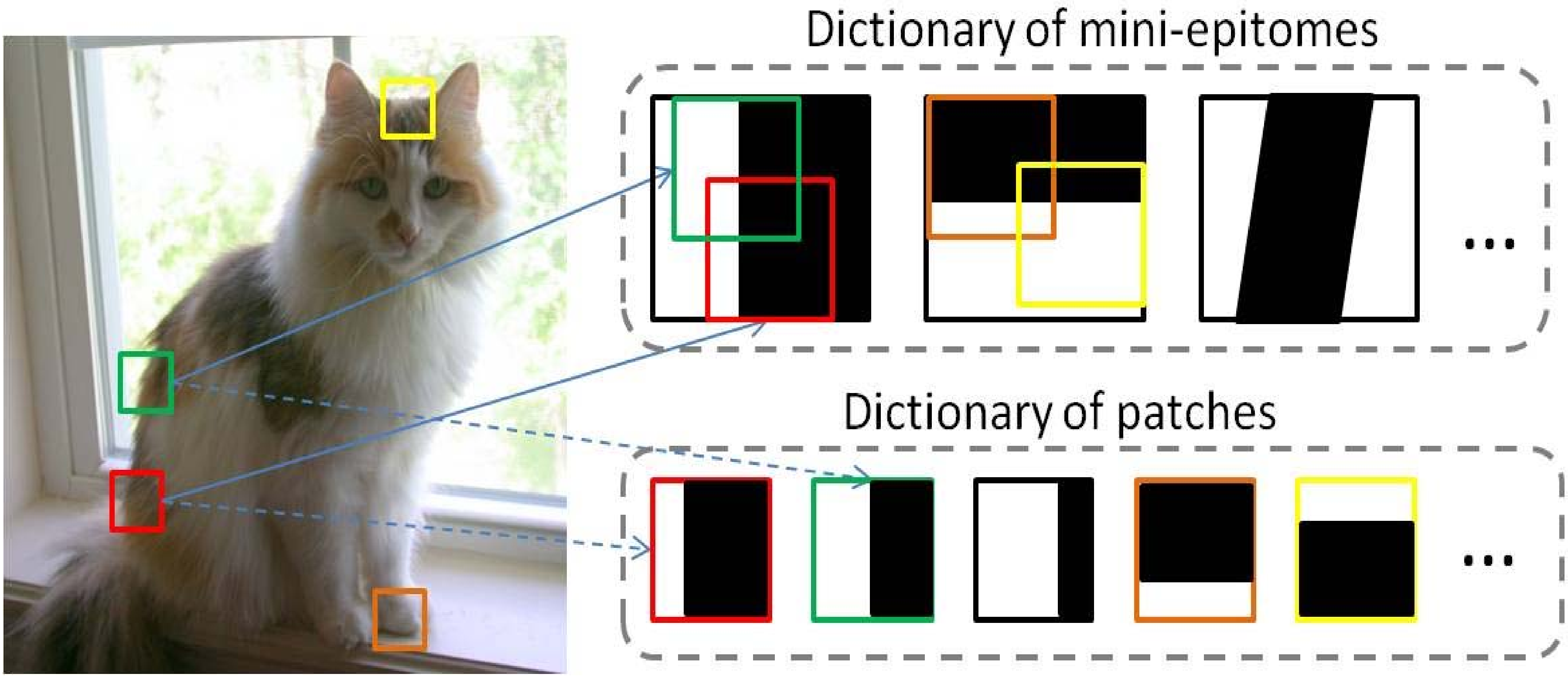
→ Our work: Build less redundant epitomic dictionaries

The Epitome Data Structure



Epitomes: Jojic, Frey, Kannan, ICCV-03

Dictionary of Mini-Epitomes



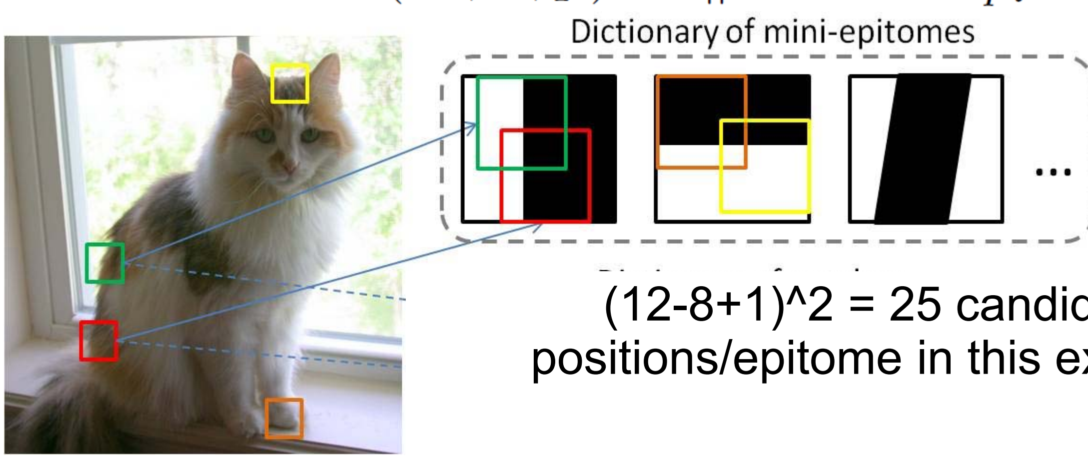
G. Papandreou, L.-C. Chen, A. Yuille (CVPR-14)

“Modeling Image Patches with a Generic Dictionary of Mini-Epitomes”

JOHNS HOPKINS UNIVERSITY

Epitomic Patch Matching

1. We have K mini-epitomes (say patch size is 8x8 pixels and mini-epitome size is 12x12 pixels).
2. For each patch $\mathbf{x}_i$ in the image and each mini-epitome k = 1:K, find the patch at position p in the epitome which minimizes the reconstruction error (whitening omitted): $R^2(\mathbf{x}_i; k, p) = \|\mathbf{x}_i - \alpha_i \mathbf{T}_p \boldsymbol{\mu}_k\|^2$

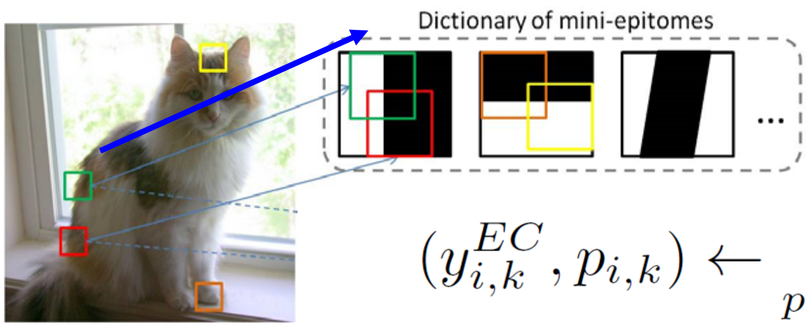


(12-8+1)^2 = 25 candidate positions/epitome in this example

3. Algorithms: Exact search (GPU, <0.5 sec/image) or ANN or dynamic programming algorithm.

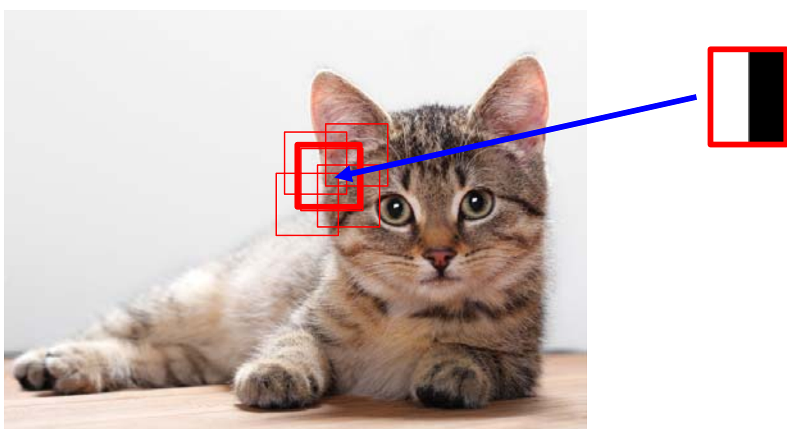
Epitomic Match vs. Max Pooling

1. Position search equivalent to *epitomic convolution*:



$$(y_{i,k}^{EC}, p_{i,k}) \leftarrow \max_{p \in \mathcal{N}_{epitome}} \frac{\mathbf{x}_i^T \boldsymbol{\nu}_{k,p}}{\|\boldsymbol{\nu}_{k,p}\|}$$

2. Epitomic convolution is an image-centric alternative to convolution followed by “max-pooling”:

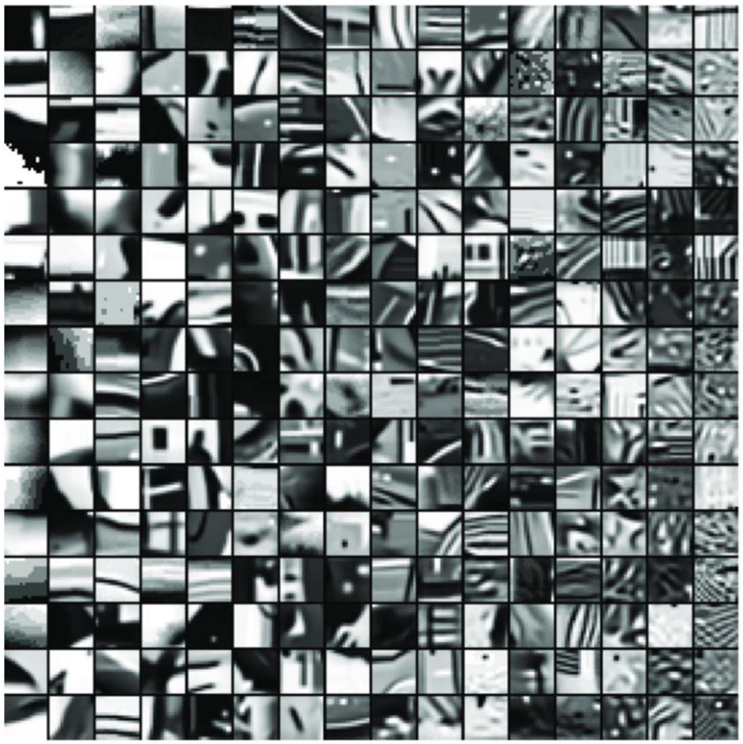


$$(y_{i,k}^{MP}, p_{i,k}) \leftarrow \max_{p \in \mathcal{N}_{image}} \mathbf{x}_{i+p}^T \boldsymbol{\mu}_k$$

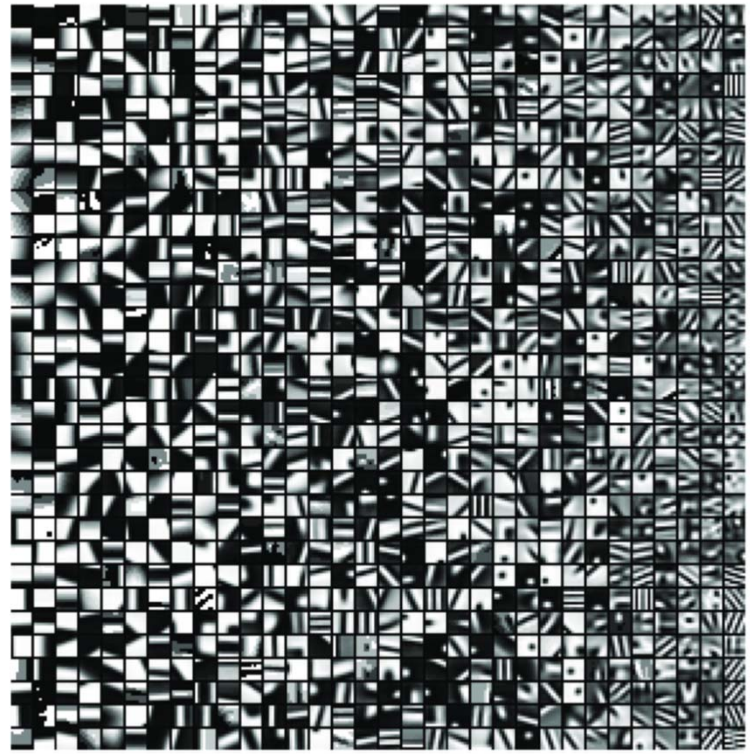
* It is much easier to define image prob models based on EC than MP

* Evaluation in discr. tasks underway

A Generic Mini-Epitome Dictionary



Epitomic dictionary
256 mini-epitomes (16x16)



Non-Epitomic dictionary
1024 elements (8x8)

Both trained on 10,000 Pascal images

Epitomic Dictionary Learning

Unsupervised training. Generative model:

1. Select mini-epitome k with prob $P(l_i = k) = \pi_k$
2. Select position p within epitome uniformly
3. Generate the patch $\mathbf{x}_i$(whitening not shown here):

$$P(\mathbf{x}_i | l_i, p_i) = \mathcal{N}(\mathbf{x}_i; \alpha_i \mathbf{T}_{p_i} \boldsymbol{\mu}_{l_i}, \sigma^2 \mathbf{I})$$

→ Max likelihood, hard EM – essentially epitomic adaptation of K-Means.

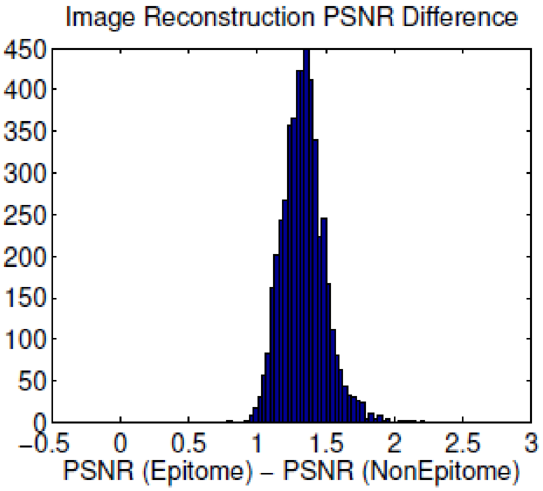
→ Faster convergence using diverse initialization of mini-epitomes by epitomic adaptation of K-Means++.

→ Mini-batch K-Means for very large-scale (to try).

Evaluation on Image Reconstruction

Original image

Epitome reconstr. (PSNR: 29.2 dB)

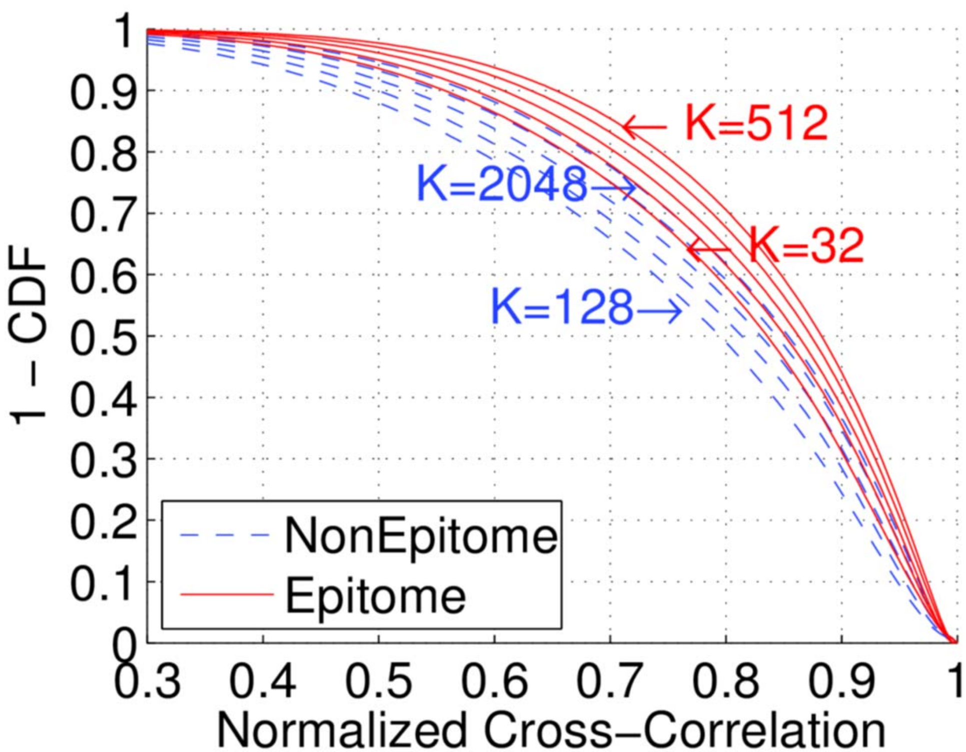


Improvement over non-epitome

Universal dictionary?

- Can a limited number of patches, or mini-epitomes “accurately” model most image patches that appear in a large set of images?
- Accurate, means normalized cross-correlation of 0.8 or higher. Perceptually the patches look similar (image patch and closest dictionary element).
- What is the set of all possible 8x8 image patches?

Epitome Benefit in Reconstruction



1. Mini-Epitome dictionary with 64 elements =

   Non-epitome dictionary with 2048 elements (8x better/ param)

2. NCC better than 0.8 for 70% of image patches