# Hardware Compliant Approximate Image Codes

Da Kuang[1], Alex Gittens[2], Raffay Hamid[3]
[1]Georgia Institute of Technology. [2]University of California, Berkeley. [3]Digital Globe Inc.

Image classification frameworks generally consist of (a) extracting local features (*e.g.* SIFT), (b) transforming them into more informative codes, and (c) using these codes for classification (*e.g.*, by using linear SVM). Over the years, several different image encoding techniques have been proposed. As reported in [1], given all things equal, most of these encoding schemes tend to produce impressive yet comparable classification accuracies. At the same time however, they can be computationally expensive. Particularly during the testing phase, their complexity can be a significant proportion of image classification pipeline (see Table 1). This limitation often makes it challenging to use these encoding schemes for large-scale learning problems.

| | Extract | Assign | Encode | Pool | Test |
|---|---|---|---|---|---|
| % Times | 6.77% | 37.76% | **42.50%** | 7.01% | 5.93% |

Table 1: %-times taken by different steps during testing for LLC [2]. Here D = 128, M = 1024, and K = 10.

In this work, we propose an approximate locality-constrained [2] encoding scheme that which is well-suited to efficiently run on modern hardware architectures, and offers significantly better efficiency than its exact counterpart, with comparable classification accuracy.

Our key insight is that for locality-constrained encodings, the set of bases used to encode a point $\mathbf{x}$, can be used equally effectively to encode a group of points similar to $\mathbf{x}$. This observation enables us to approximately encode similar groups of points simultaneously by using shared sets of bases, as opposed to exactly encoding points individually each using their own bases (see Figure 1 for illustration, and Algorithm 1 for an enlisting of our approach). This difference improves our encoding efficiency in two important ways:

- It significantly reduces the number of locality related matrices needed to be factored, from number of points ($\mathcal{O}$(millions)) to number of point-clusters ($\mathcal{O}$(thousands)).
- It lets us view the encoding problem of each point-group as a linear system with a shared left hand side. Solving such a system can be posed as matrix-matrix multiplication that can fully exploit the cache-efficient modern hardware architecture.

These efficiency advantages enable our approximate scheme to achieve a significant speed-up ($\sim 40\times$) over its exact counterpart, while maintaining comparable accuracy. Our accuracy and efficiency results are summarized in Table 2 and Figure 2. The comparisons with state-of-the-art methods are summarized in Table 3.

To summarize, the **main contributions** of our work are:

- A simple yet effective approximate encoding scheme with significant performance gains and similar classification accuracy compared to its exact counterpart.
- A formal approximation analysis of our approach using perturbation analysis of least-square problems.
- A thorough set of empirical analyses to assess the capability of our encoding scheme both from a representational as well as a discriminative perspective.

[1] Ken Chatfield, Victor Lempitsky, Andrea Vedaldi, and Andrew Zisserman. The devil is in the details: an evaluation of recent feature encoding methods. 2011.

[2] Jinjun Wang, Jianchao Yang, Kai Yu, Fengjun Lv, Thomas Huang, and Yihong Gong. Locality-constrained linear coding for image classification. In *IEEE CVPR*, 2010.

This is an extended abstract. The full paper is available at the Computer Vision Foundation webpage.

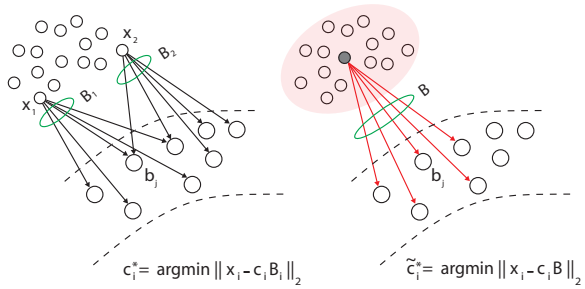


Figure 1: (Left) Locality-constrained encoding [2] finds different sets of bases nearest to each feature to exactly construct its locally-constrained codes. (Right) In contrast, we approximately encode clusters of points simultaneously by using shared sets of bases nearest to the cluster-centroid.

**Algorithm 1** Hardware Compliant Approximate Encoding

**Input**: Image descriptors $\mathbf{X} \in \mathbb{R}^{D\times N}$, codebook $\mathbf{B} \in \mathbb{R}^{D\times M}$, cluster assignment for all the descriptors
**Output**: Approximate image codes $\widetilde{\mathbf{C}} \in \mathbb{R}^{M\times N}$
Form $\mathbf{X}^1, \mathbf{X}^2, \cdots, \mathbf{X}^M$ by gathering the descriptors belonging to each individual cluster
**for** $m = 1$ to M **do**
    Determine $\mathbf{B}^m \in \mathbb{R}^{D\times K}$
    Compute left-hand side $\mathbf{B}^{m\mathrm{T}}\mathbf{B}^m \equiv \mathbf{W}$
    Perform Cholesky factorization: $\mathbf{W} = \mathbf{L}\mathbf{L}^{\mathrm{T}}$
    Compute right-hand side $\mathbf{B}^{m\mathrm{T}}\mathbf{X}^m \equiv \mathbf{Y}$
    Solve $\mathbf{L}\mathbf{Z} = \mathbf{Y}$ for $\mathbf{Z}$, and $\mathbf{L}^{\mathrm{T}}\widetilde{\mathbf{C}}^m = \mathbf{Z}$ for $\widetilde{\mathbf{C}}^m$
**end for**
Normalize each column of $\widetilde{\mathbf{C}}$ to unit $L_2$ norm

| | Accuracy | | Timing | | |
|---|---|---|---|---|---|
| | LLC | Proposed | LLC | Proposed | Speed-Up |
| Caltech-101 | **72.16**±0.7 | **71.35**±0.8 | 512.8 sec. | 12.9 sec. | **39.8**× |
| Caltech-256 | **37.04**±0.3 | **35.69**±0.2 | 1779 sec. | 47.3 sec. | **37.6**× |
| Pascal-07 | **51.95** | **52.90** | 208.0 min. | 4.86 min. | **42.8**× |
| MIT Scenes | **38.30** | **39.91** | 473.3 sec. | 12.8 sec. | **37.0**× |

Table 2: Classification accuracies and timing results for encoding on different data-sets using the exact LLC [2] and the proposed approximate encoding.

| | $(n = 30)$ | | $(n = 15)$ | | | |
|---|---|---|---|---|---|---|
| | PSC | Proposed | Proposed | LcSA | SA | SC |
| **Accuracy** | 76.71 | 76.43 | 72.54 | 71.90 | 71.60 | 74.60 |
| **Run-time** | 0.45 sec | 0.258 sec | 0.258 sec | 1.06 sec | 66.6 sec | 106.4 sec |

Table 3: Classification accuracy and run-time per image for state-of-the-art image coding methods on Caltech-101. The parameters $n$ indicates the number of training images per category.

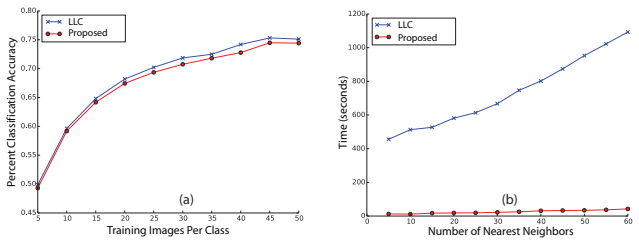


Figure 2: (a) Accuracy with respect to training size. (b) Efficiency comparison.