



Spatially Local Coding for Object Recognition

Sancho McCann and David G. Lowe

Laboratory for Computational Intelligence
University of British Columbia
Vancouver, Canada

Overview

Spatial pyramid models:

- Codebooks based on appearance only
- Location accounted for by spatial pooling in the spatial pyramid bins

Spatially local coding:

- Codebooks based on appearance **and** location
- No complicated pooling stage
- Uses linear SVM

We move responsibility for spatial locality to the coding phase.

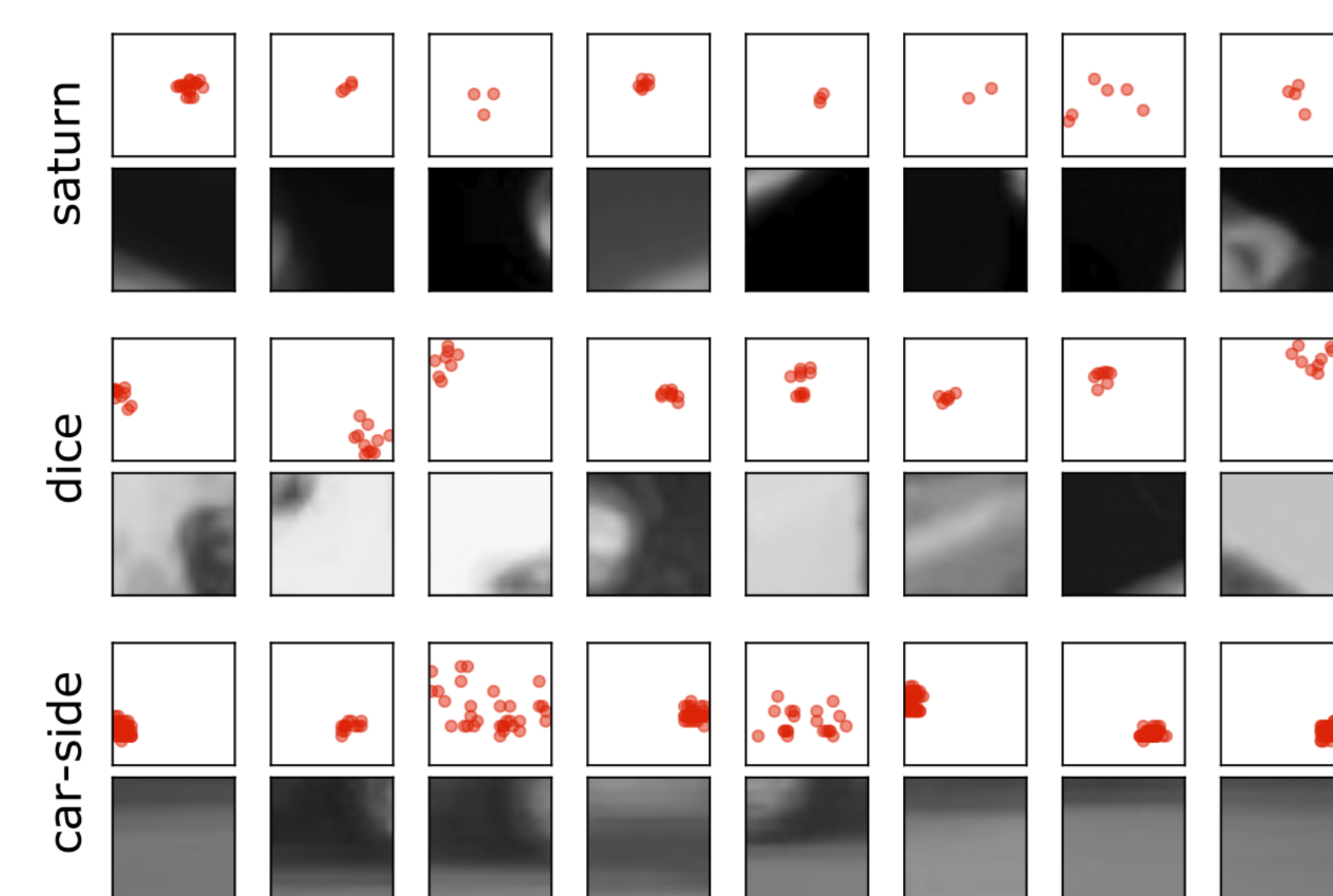
Method	Appearance locality	Spatial locality
Bag-of-features	Coding	-
Spatial pyramid [1]	Coding	Pooling
Sparse coding SPM [2]	Sparse coding	Pooling
LLC [3]	Local coding	Pooling
"Ask-the-locals" [4]	Sparse coding + Pooling	Pooling
Spatially local coding	Coding	Coding

Simply add weighted location dimensions to each descriptor.

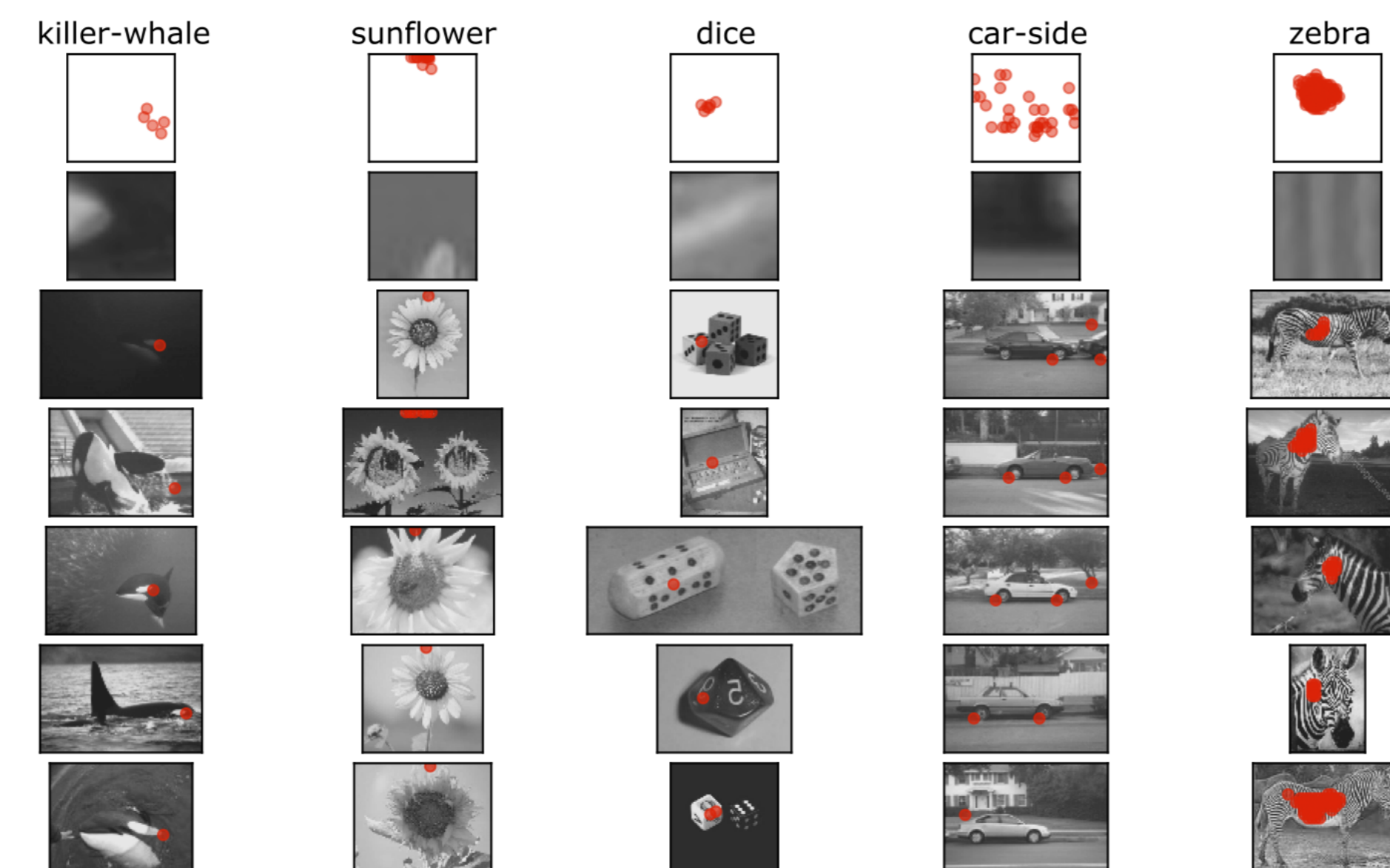
$$\phi_i^{(\lambda)} = [\phi_{i1}, \phi_{i2}, \dots, \phi_{id}, \lambda x_i, \lambda y_i]$$

Features are pooled as in bag-of-words.

Learned features



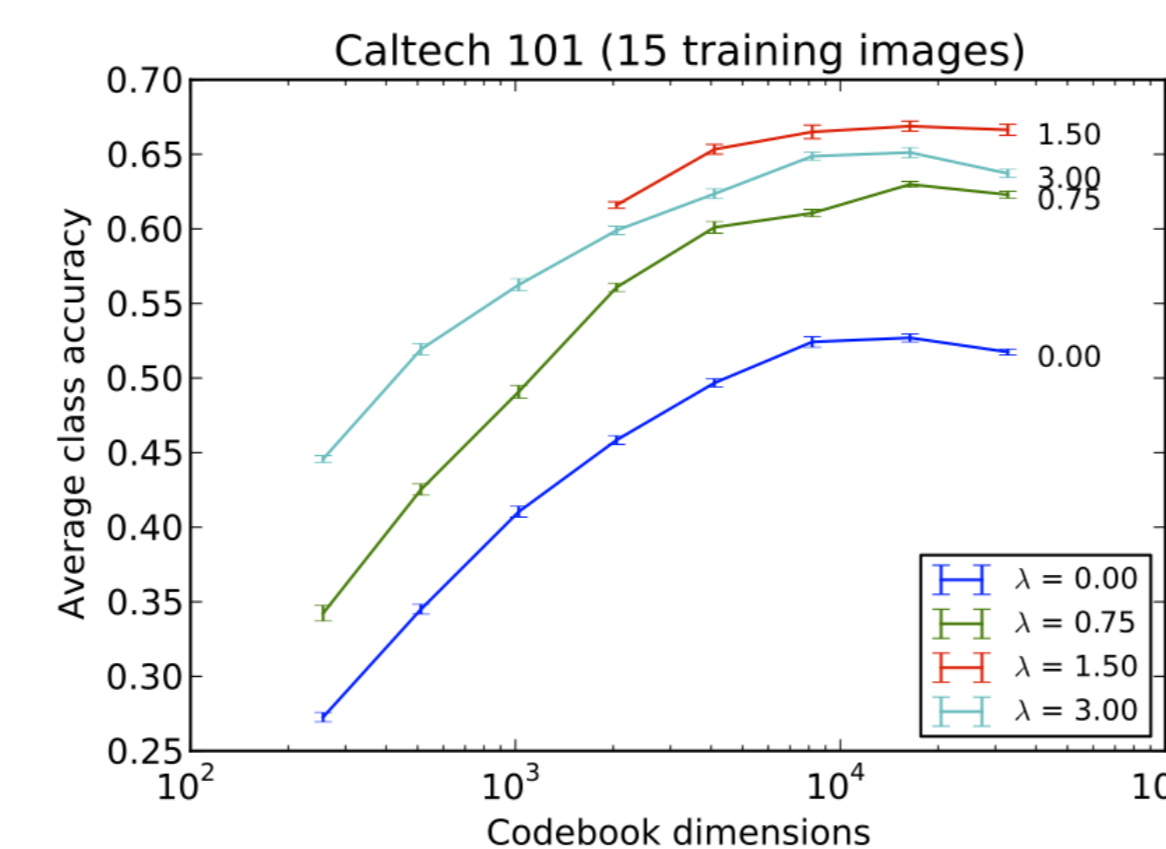
Top 10 features for three of the categories



Feature matches in actual training images

Location weighting

How much should we weight location?



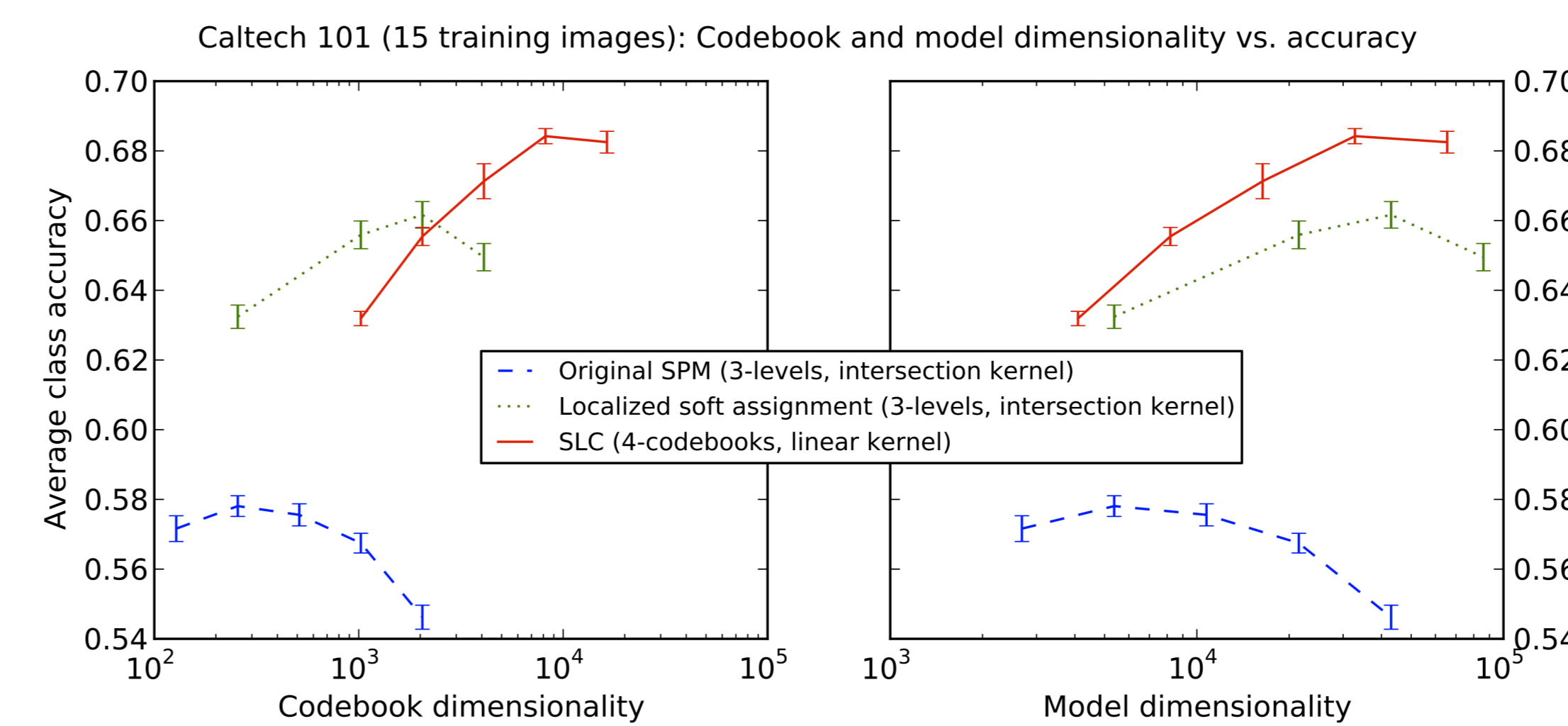
On Caltech 101, it would be optimal to use a location weighting of approximately 1.5.

However, this won't be the case for every dataset, and we can get better performance by combining together multiple codebooks, each with a different location weighting:

Location weighting	Caltech 101 (15 training)
$\lambda = 0.00$	52.4 ± 0.4
$\lambda = 0.75$	61.1 ± 0.2
$\lambda = 1.50$	66.5 ± 0.5
$\lambda = 3.00$	64.9 ± 0.3
4 codebooks combined	68.4 ± 0.2

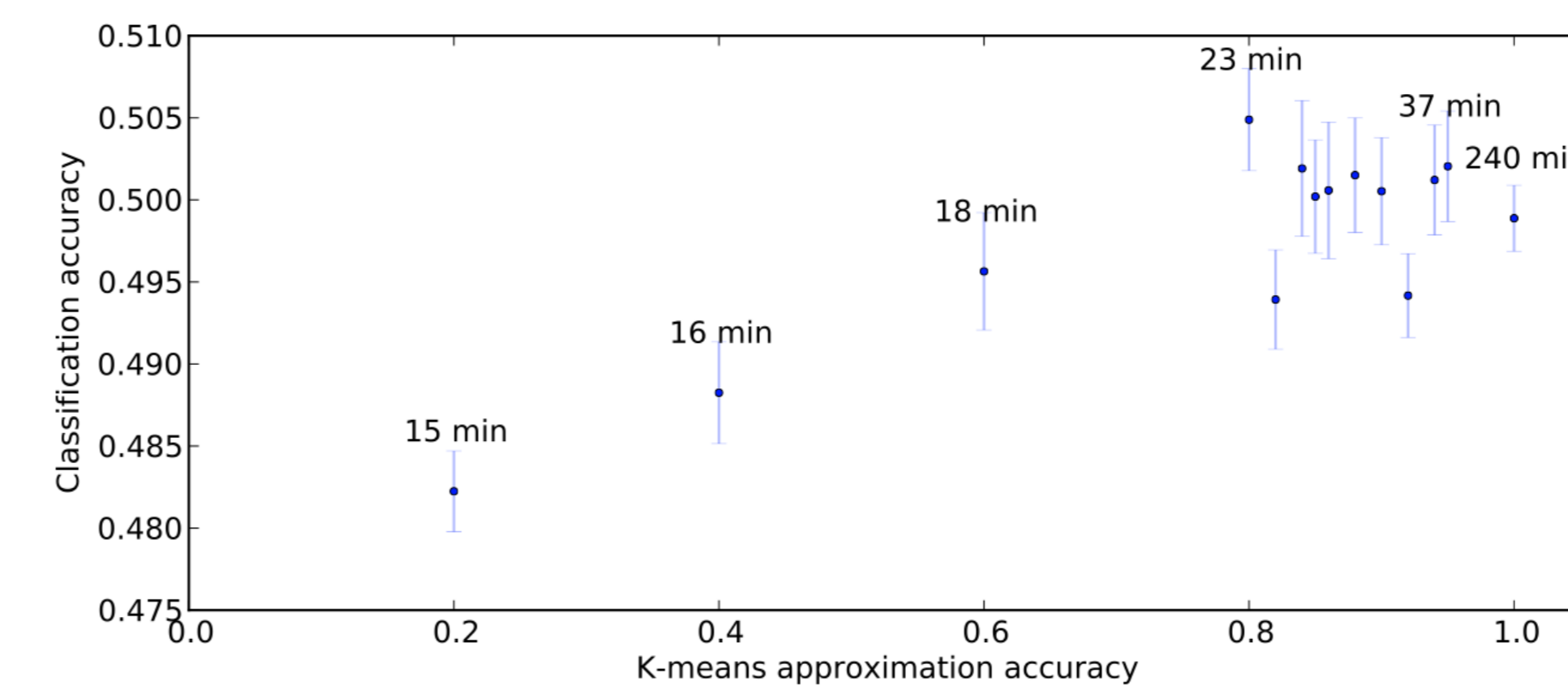
Codebook and model size

Our codebooks use many more codewords than SPM models to achieve optimal performance. However, the resultant model size is comparable:



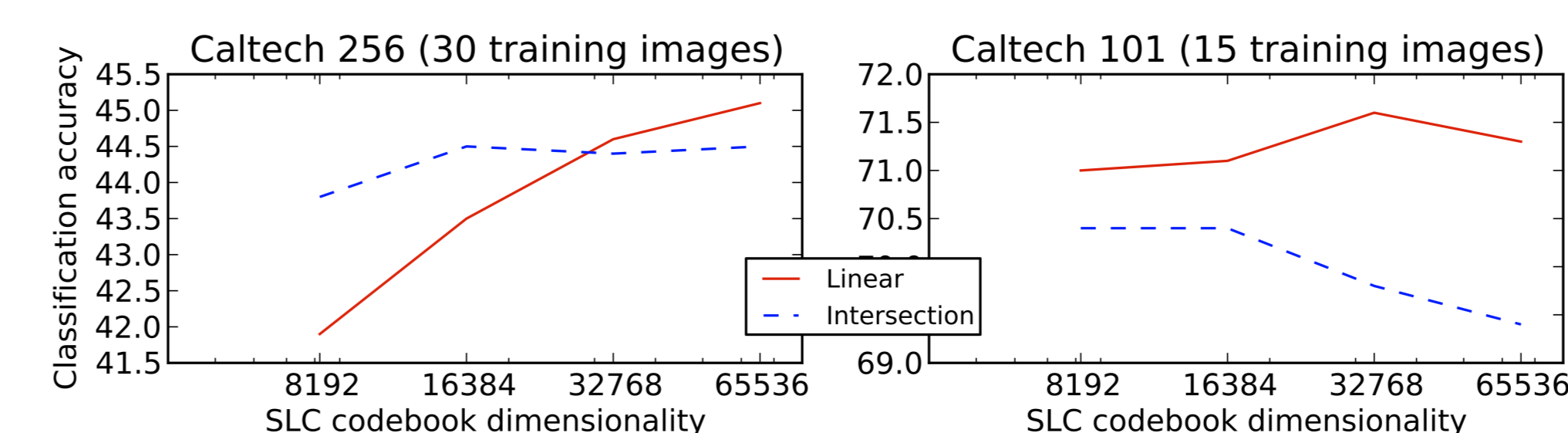
Efficient codebook construction

We use FLANN [5] to do approximate matching during each k-means iteration. This greatly speeds up codebook construction time while having no effect on resultant classification performance:



SVM Training

Our model yields best results with a linear SVM. Only at low dimensionalities on Caltech 256, where performance with both SVM types is low, does the intersection kernel perform better than the linear kernel.



Classification performance

	Cal. 101 (15)	Cal. 101 (30)	Cal. 256 (30)
Single-scale SIFT features Extracted every 8 pixels			
Original SPM [1]	56.4	64.6 ± 0.8	-
Original SPM (ours)	57.8 ± 0.3	65.2 ± 0.4	30.0 ± 0.4
Localized soft assignment [6]	-	74.21 ± 0.81	-
Localized soft assignment (ours)	66.2 ± 0.4	72.2 ± 0.3	37.2 ± 0.2
Sparse coding SPM [3]	67.0 ± 0.45	73.2 ± 0.54	34.02 ± 0.35
Sparse coding SPM [7]	-	71.8 ± 1.0	-
SLC [8192d x 4]	68.4 ± 0.2	75.5 ± 0.4	38.9 ± 0.3
SLC [16384d x 4]	68.3 ± 0.3	75.7 ± 0.4	40.0 ± 0.2
Multi-scale features Extracted every 8 pixels			
LLC (HOG) [3]	65.43	73.44	41.19
Localized soft assignment (SIFT) [6]	-	76.58 ± 0.71	-
SLC [8192d x 4]	71.4 ± 0.4	78.0 ± 0.4	41.8 ± 0.3
SLC [16384d x 4]	72.5 ± 0.3	79.2 ± 0.2	43.4 ± 0.2
SLC [32768d x 4]	70.9 ± 0.4	77.2 ± 0.6	44.3 ± 0.1
Single-scale SIFT features Extracted every 4 pixels			
Boureau et al. [4]	-	77.3 ± 0.6	41.7 ± 0.8
SLC [32768d x 4]	71.6 ± 0.4	79.6 ± 0.8	44.6 ± 0.2
SLC [65536d x 4]	-	-	45.1 ± 0.2
Multi-scale SIFT features Extracted every 4 pixels			
SLC [65536d x 4]	72.7 ± 0.4	81.0 ± 0.2	46.6 ± 0.2

References

- [1] Lazebnik, S., Schmid, C., Ponce, J.: Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. CVPR (2006)
- [2] Yang, J., Yu, K., Gon, Y., Huang, T.: Linear spatial pyramid matching using sparse coding for image classification. In: CVPR. (2009)
- [3] Wang, J., Yang, J., Yu, K., Lv, F., Huang, T., Gong, Y.: Locality-constrained linear coding for image classification. In: CVPR. (2010)
- [4] Boureau, Y.L., Le Roux, N., Bach, F., Ponce, J., LeCun, Y.: Ask the locals: multi-way local pooling for image recognition. In: ICCV. (2011)
- [5] Muja, M., Lowe, D.: Fast approximate nearest neighbors with automatic algorithm configuration. In: VISSAPP. (2009)
- [6] Liu, L., Wang, L., Liu, X.: In Defense of Soft-assignment Coding. In: ICCV. (2011)
- [7] Boureau, Y.L., Bach, F., LeCun, Y., Ponce, J.: Learning mid-level features for recognition. In: CVPR. (2010)