Supplementary Material for On Compact Codes for Spatially Pooled Features

1. Relationship between $\mathbf{K}'$ and $\mathbf{C}'$

We briefly prove the bound on $\mathbf{K}$ we presented in the paper:

$$||\mathbf{K} - \mathbf{K}'||_F \leq O' + M' \left(\frac{1}{c}\right)^{\frac{1}{4}}.$$

Recall that $\mathbf{K} = \mathbf{C}\mathbf{C}^\top$ and $\mathbf{K}' = \mathbf{C}'\mathbf{C}'^\top$, and since $\mathbf{C}$ and $\mathbf{C}'$ are symmetric, $\mathbf{K} = \mathbf{C}^2$ and $\mathbf{K}' = \mathbf{C}'^2$. Note that the Frobenius norm satisfies subadditivity and submultiplicativity properties (Meyer, 2001), i.e.,

$$||A + B||_F \leq ||A||_F + ||B||_F, \text{ and} \quad (1)$$
$$||AB||_F \leq ||A||_F ||B||_F. \quad (2)$$

Thus, we have

$$\begin{aligned} ||\mathbf{K} - \mathbf{K}'|| &= ||\mathbf{C}^2 - \mathbf{C}'^2|| \quad (3)\\ &= ||(\mathbf{C} - \mathbf{C}')\mathbf{C} + \mathbf{C}'(\mathbf{C} - \mathbf{C}')|| \\ &\leq ||(\mathbf{C} - \mathbf{C}')\mathbf{C}|| + ||\mathbf{C}'(\mathbf{C} - \mathbf{C}')|| \\ &\leq ||(\mathbf{C} - \mathbf{C}')||||\mathbf{C}|| + ||\mathbf{C}'||||(\mathbf{C} - \mathbf{C}')|| \\ &\leq ||(\mathbf{C} - \mathbf{C}')||||\mathbf{C}|| + ||\mathbf{C}' - \mathbf{C}||^2 + \\ &\quad + ||\mathbf{C}||||(\mathbf{C} - \mathbf{C}')|| \\ &= ||(\mathbf{C} - \mathbf{C}')|| \left(||(\mathbf{C} - \mathbf{C}')|| + 2||\mathbf{C}||\right) \\ &= \mathcal{O}(||(\mathbf{C} - \mathbf{C}')||) \end{aligned}$$

where all the $||.||$ are the Frobenius norms, and where in the last line we assumed that $||(\mathbf{C} - \mathbf{C}')||$ is sufficiently small and $||\mathbf{C}||$ is constant w.r.t. c. Thus, we can expect that the approximation quality of $\mathbf{K}'$ will be similar than $\mathbf{C}'$, and we know that the quality of the kernel approximation $\mathbf{K}'$ will determine the accuracy of the final classifier, which we will also empirically show in the experiments.

2. Note on the Metric in the Coding Space

In our derivation linking recent coding strategies (Coates & Ng, 2011) to Nyström sampling, we noted that the approach taken when considering doing subsampling of the coding matrix $\mathbf{C}$ to form a new code matrix to which we want to apply linear SVM is given by:

$$\mathbf{K}' = \mathbf{C}'\mathbf{C}'^\top = \mathbf{E}\mathbf{W}^{-1}\mathbf{E}^\top\mathbf{E}\mathbf{W}^{-1}\mathbf{E}^\top = \mathbf{E}\mathbf{\Lambda}\mathbf{E}^\top$$

where $\mathbf{E}$ is a matrix with the coded feature vector stacked as rows, and covers various encoding techniques. However, most of the work in the literature does not consider the square matrix $\mathbf{\Lambda}$ which arises when derivating the new kernel matrix from a Nyström sampling point of view. This is a square $c \times c$ matrix, where c is the dictionary size. Furthermore, the matrix is symmetric and PSD, which means that we can interpret it as a metric in the c dimensional coding space. This could also be interpreted as an skewed regularization term for the classifiers, and with enough training data, the effect of this regularization term may be ignorable.

We further note that, if the selected columns when doing Nyström sampling are orthogonal, $\mathbf{\Lambda}$ is going to be diagonal, and as a consequence the effect of not considering it will be negligible, as it would act as a per dimension standard deviation normalization, which is typically done before the linear SVM regardless. Even though uniformly sampling columns of the original coding matrix $\mathbf{C}$ (that is, to randomly select samples from our training set as dictionary elements) yields good performance (Coates & Ng, 2011), methods such as Orthogonal Matching Pursuit perform better, specially for small values of c. This can now be partially explained by the fact that, since most of these methods did not consider $\mathbf{\Lambda}$, the gap between the implementation of Nyström sampling and methods without $\mathbf{\Lambda}$ is artificially closed by selecting samples that were dissimilar, yielding a closer to diagonal $\mathbf{\Lambda}$ matrix.

3. Dataset Description

We describe the experimental settings on the four classification benchmarks: CIFAR-10, STL-10, TIMIT and WSJ. The CIFAR-10 dataset and the STL dataset both contain image data, with the former focusing on large labeled examples and the latter on large unsupervised images with a small amount of labeled examples. TIMIT is a speech database consisting of read digits

that contains two orders of magnitude more training samples than the other datasets, and the largest output label space as it has phone states as the output, and WSJ is a corpus with roughly five times more data than TIMIT, and consists of read sentences of the Wall Street Journal corpus.

CIFAR-10 and STL-10 The two datasets both contain 10 object classes. We rescale the STL-10 dataset so that the image sizes for both datasets are 32×32, and then follow the state-of-the-art pipeline defined in Coates & Ng (2011): dense 6x6 local patches with ZCA whitening are extracted with stride 1, and thresholding coding with $\alpha = 0.25$ is adopted for encoding. The codebook is trained with OMP-1. The features are then average-pooled on a 2×2 grid to form the global image representation.

TIMIT The TIMIT data is collected from speech streams using a 25-ms Hamming window with a 10-ms fixed frame rate. We represent the speech using first- to 12th-order Mel frequency cepstral coefficients (MFCCs) and energy, along with their first and second temporal derivatives. The training set consists of 462 speakers, with a total number of frames in the training data of size 1.1 million. The development set contains 50 speakers, with a total of 120K frames, and is used for cross validation. Results are reported using the standard 24-speaker core test set consisting of 192 sentences with 7333 phone tokens and 57920 frames.

The data is normalized to have zero mean and unit variance. All experiments used a context window of 11 frames. This gives a total of $39 \times 11 = 429$ elements in each feature vector. We used 183 target class labels (i.e., three states for each of the 61 phones), which are typically called "phone states", with a one-hot encoding.

The pipeline adopted is otherwise unchanged from the previous dataset. However, we did not apply pooling, and instead coded the whole 429 dimensional vector with a dictionary found with OMP-1, with the same parameter α as in the vision tasks. The competitive results with a framework known in vision adapted to speech has been recently reported in Vinyals & Deng (2012).

WSJ All experiments were conducted on the 5000-word speaker independent WSJ0 (5k-WSJ0) task (Paul & Baker, 1992). The training material from the SI84 set (7077 utterances, or 15.3 hours of speech from 84 speakers) is separated into a 6877-utterance training set and a 200-sentence cross-validation (CV) set. Evaluation was carried out on the Nov92 evaluation data with 330 utterances from 8 speakers. The features and pipeline is exactly the same as we used for TIMIT. However, the phone labels were derived from the forced alignments generated using a 2818 8-mixture tied-state cross-word tri-phone GMM-HMM speech recognition system trained with maximum likelihood criterion.

References

Coates, A and Ng, A. The importance of encoding versus training with sparse coding and vector quantization. In *ICML*, 2011.

Krizhevsky, A. Learning multiple layers of features from tiny images. Technical report, 2009.

Meyer, CD. *Matrix Analysis and Applied Linear Algebra*. SIAM, 2001.

Paul, DB and Baker, JM. The design for the Wall Street Journal-based CSR corpus. In *ICSLP*, 1992.

Vinyals, O and Deng, L. Are sparse representations rich enough for acoustic modeling? In *Interspeech*, 2012.