International Conference on Computer Vision
and Pattern Recognition (CVPR), 2013

Fast Convolutional Sparse Coding

Hilton Bristow,[1,3] Anders Eriksson[2] and Simon Lucey[3]
[1]Queensland University of Technology, Australia
[2]The University of Adelaide, Australia [3]CSIRO, Australia
{hilton.bristow, simon.lucey}@csiro.au, anders.eriksson@adelaide.edu.au

Abstract

*Sparse coding has become an increasingly popular method in learning and vision for a variety of classification, reconstruction and coding tasks. The canonical approach intrinsically assumes independence between observations during learning. For many natural signals however, sparse coding is applied to sub-elements (*i.e. *patches) of the signal, where such an assumption is invalid. Convolutional sparse coding explicitly models local interactions through the convolution operator, however the resulting optimization problem is considerably more complex than traditional sparse coding. In this paper, we draw upon ideas from signal processing and Augmented Lagrange Methods (ALMs) to produce a fast algorithm with globally optimal subproblems and super-linear convergence.*

1. Introduction

Sparse dictionary learning algorithms aim to factorize an ensemble of input vectors $\{\mathbf{x}\}_{n=1}^N$ into a linear combination of overcomplete basis elements $\mathbf{D} = [\mathbf{d}_1, \ldots, \mathbf{d}_K]$ under sparsity constraints. One of the most popular forms of this algorithm attempts to solve,

$$\begin{aligned} \arg\min_{\mathbf{d},\mathbf{z}} \quad & \frac{1}{2}\sum_{n=1}^{N} ||\mathbf{x}_n - \mathbf{D}\mathbf{z}_n||_2^2 + \beta||\mathbf{z}_n||_1 \\ \text{subject to} \quad & ||\mathbf{d}_k||_2^2 \leq 1 \text{ for } k = 1 \ldots K, \end{aligned} \tag{1}$$

where β controls the L_1 penalty, and the inequality constraint on the columns of $\mathbf{D}$ prevent the dictionary from absorbing all of the system's energy. This problem is also known as *basis pursuit* [5] or *LASSO* [19] and has proven useful in a variety of perceptual classification [24], reconstruction [4], and coding tasks, and numerous papers have been devoted to finding fast exact and approximate solutions to this problem, significantly the works of [1, 11, 13].

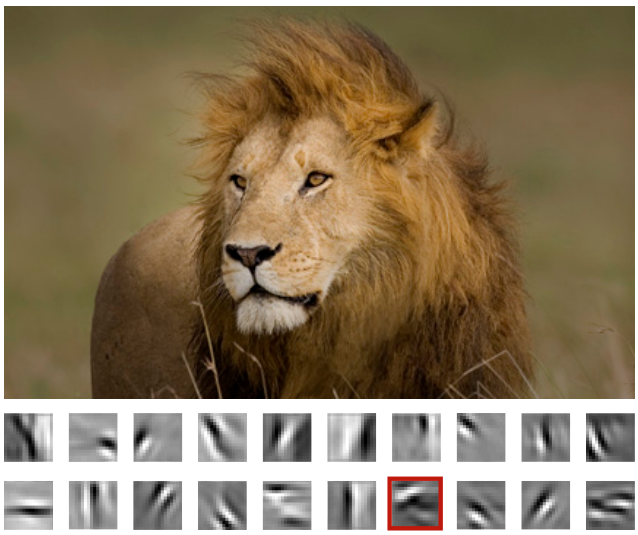

Figure 1. A selection of filters learned from an unaligned set of lions. The spatially invariant algorithm produces expression of generic Gabor-like filters as well as specialized domain specific filters, such as the highlighted "eye".

Sparse coding has a fundamental drawback however, as it assumes the ensemble of input vectors $\{\mathbf{x}_n\}_{n=1}^N$ are independent of one another, *i.e.* the components of the bases are arbitrarily aligned with respect to the structure of the signal.

This independence assumption, when applied to natural images, leads to many basis elements that are translated versions of each other. *Convolutional* sparse coding attempts to remedy this shortcoming by modelling shift invariance directly within the objective,

$$\begin{aligned} \arg\min_{\mathbf{d},\mathbf{z}} \quad & \frac{1}{2}||\mathbf{x} - \sum_{k=1}^{K} \mathbf{d}_k * \mathbf{z}_k||_2^2 + \beta \sum_{k=1}^{K} ||\mathbf{z}_k||_1 \\ \text{subject to} \quad & ||\mathbf{d}_k||_2^2 \leq 1 \text{ for } k = 1 \ldots K. \end{aligned} \tag{2}$$

Now $\mathbf{z}_k$ takes the role of a sparse feature map which, when convolved with a filter $\mathbf{d}_k$ and added over all k, should approximate the input signal $\mathbf{x}$: a full signal

(*e.g.* image or audio sequence) rather than independent patches as in the objective of Equation (1). Like traditional sparse coding the estimated sparse basis $\{\mathbf{d}_k\}_{k=1}^K$ will be of a fixed spatial support. However, unlike traditional sparse coding the input signal $\mathbf{x}$ and the sparse feature maps $\{\mathbf{z}_k\}_{k=1}^K$ are of a different and usually much larger dimensionality. Note also that we assume there is only a single signal $\mathbf{x}$ in our formulation in Equation (2); it is trivial in our proposed formulation to handle multiple signals each of varying length. We persist with the single signal assumption throughout the derivation of our approach for the sake of clarity and brevity.

Notation: Matrices are always presented in uppercase bold (*e.g.*, $\mathbf{A}$), vectors are in lower-case bold (*e.g.*, $\mathbf{a}$) and scalars in lower-case (*e.g.*, a). A 2D convolution operation is represented as the $*$ operator. The matrix $\mathbf{I}_D$ denotes a $D \times D$ identity matrix, and $\otimes$ denotes the Kronecker product operator. A $\hat{}$ applied to any vector denotes the Discrete Fourier Transform (DFT) of a vectorized signal $\mathbf{a}$ such that $\hat{\mathbf{a}} \leftarrow \mathcal{F}(\mathbf{a}) = \mathbf{F}\mathbf{a}$, where $\mathcal{F}()$ is the Fourier transforms operator and $\mathbf{F}$ is the $D \times D$ matrix of complex basis vectors for mapping to the Fourier domain for any D dimensional vectorized image/signal. We have chosen to use a Fourier representation in this paper due to its particularly useful ability to represent convolutions as a Hadamard product in the Fourier domain. Additionally, we take advantage of the fact that $\text{diag}(\hat{\mathbf{z}})\hat{\mathbf{a}} = \hat{\mathbf{z}} \odot \hat{\mathbf{a}}$, where $\odot$ represents the Hadamard product, and diag() is an operator that transforms a D dimensional vector into a $D \times D$ dimensional diagonal matrix. Commutativity holds with this property such that role of filter $\hat{\mathbf{z}}$ or signal $\hat{\mathbf{a}}$ can be interchanged. Any transpose operator ^{T} on a complex vector or matrix in this paper additionally takes the complex conjugate in a similar fashion to the Hermitian adjoint [17]. With respect to Equation (2) – the objective of central interest in this paper – we will often omit filter indices (*e.g.* $\mathbf{d}_k$ refers to the kth filter and $\mathbf{z}_k$ refers to the kth filter response) when referring to the variables being optimized. In these instances we assume that $\mathbf{d} = [\mathbf{d}_1^T, \ldots, \mathbf{d}_K^T]^T$ and $\mathbf{z} = [\mathbf{z}_1^T, \ldots, \mathbf{z}_K^T]^T$. When we employ Fourier notation $\hat{\mathbf{z}}$ on the super vector $\mathbf{z}$, then $\hat{\mathbf{z}} = [\mathcal{F}(\mathbf{z}_1)^T, \ldots, \mathcal{F}(\mathbf{z}_K)^T]^T$ is implied.

Prior Art: The idea that sparse part-based representations form the computational foundations of visual systems has been supported by Olshausen and Field [15, 16] and many other studies [8, 20, 21]. Neurons in the inferotemporal cortex respond to moderately complex features which are invariant to the position of stimulus within the visual field. Based on this observation, Hashimoto proposed a model which allowed features to be shifted by a given amount within each image patch [8]. The resulting features were complex patterns rather than the Gabor-like features obtained by sparse coding. [23] and [7] extended the approach to a more general set of linear transforms applied to each patch with similar results. The idea of truly shift-invariant or "convolutional" sparse coding was first proposed by Lewicki and Sejnowski for discrete 1D time-varying signals [12], and later generalized to images by Mrup *et al.* [14].

Zeiler *et al.*'s work in convolutional sparse coding was motivated by the study of deconvolutional networks [26, 27], which are closely related to the seminal works of Lecun on convolutional networks [9, 10]. Zeiler *et al.* proposed to solve the objective in Equation (2) through an alternation strategy where one solves a sequence of convex subproblems until convergence. The approach alternates between solving the subproblem $\mathbf{d}$ given a fixed $\mathbf{z}$, and the subproblem $\mathbf{z}$ given a fixed $\mathbf{d}$. A drawback to this strategy however, is the computational overhead associated with both subproblems. The introduction of convolution necessitates the use of gradient solvers for each subproblem, with linear convergence properties dramatically affecting the convergence properties of the overall algorithm.

Zeiler further introduced an auxiliary variable, $\mathbf{t}$, to separate the convolution from the L_1 regularization (allowing for an explicit and efficient solution to $\mathbf{t}$ using soft thresholding). Instead of enforcing the equality constraint $\mathbf{z} = \mathbf{t}$ explicitly, the authors add a quadratic term $\frac{\mu}{2}||\mathbf{z} - \mathbf{t}||_2^2$ to penalize violations. This quadratic penalty can be reinterpreted as a trust region constraint $||\mathbf{z} - \mathbf{t}||_2^2 \leq \epsilon$ where $\epsilon \propto \mu^{-1}$. In order to satisfy the equality constraint, μ must be increased arbitrarily large, which simultaneously forces the new estimate of $\mathbf{z}$ to be within ϵ of $\mathbf{t}$ and increases numerical error. The optimal value of subproblem $\mathbf{d}$ requires solution of a QCQP. To avoid this added computational burden, Zeiler normalizes the solution to an unconstrained minimization, and while this tends to work in practice, it is an approximation not guaranteed to converge to the global minima of the original constrained objective (see Figure 2).

Similar to Zeiler's method is FISTA, from the family of proximal gradient methods [1]. It is a well known iterative method capable of solving L_1 regularized least squares problems with quadratic convergence properties. For FISTA to approach the L_1-min however, $\beta \to 0$. Augmented Lagrange methods (ALMs) – such as the method of multipliers (ADMM) we use – have similar quadratic convergence properties under more modest conditions [2, 25] and through their capacity

to compose functions, present fast, scalable and distributed solvers.

Contributions: We make four specific contributions in this paper:

- We advocate the use of Alternating Direction Method of Multipliers (ADMMs) approach, over the traditional continuation method, for separating the L_1 penalty from the convolutional component of the objective using auxilliary variables. We argue that an algorithmic speedup can be obtained by applying an ADMM approach to the objective as a whole rather than the L_1 subproblem alone.
- We demonstrate that the convolution subproblem can be solved efficiently and explicitly in the Fourier domain; outperforming conventional gradient solvers that use spatial convolution. By incorporating this approach within an ADMM optimization strategy the inequality constraints on the norm of the dictionary elements can be satisfied exactly by scaling the solution to an isotropic problem through the introduction of an additional auxillary variable.
- We propose a quad-decomposition of the objective into subproblems that are convex and can be solved efficiently and explicitly without the need for gradient or sparse solvers. As a result, we demonstrate an improvement in the computational efficiency of convolutional sparse coding over canonical methods (*i.e.* Zeiler *et al.*).
- Finally, we present a convolutional sparse coding library, which can plug-and-play directly into existing image and audio coding applications: `hiltonbristow.com/software`.

2. Problem Reformulation

Our proposed approach to solving convolutional sparse coding involves the introduction of two auxiliary variables $\mathbf{t}$ and $\mathbf{s}$ as well as the posing of the convolutional portion of the objective in the Fourier domain,

$$\begin{aligned} \arg\min_{\mathbf{d},\mathbf{s},\mathbf{z},\mathbf{t}} \quad & \frac{1}{2D}||\hat{\mathbf{x}} - \sum_{k=1}^{K} \hat{\mathbf{d}}_k \odot \hat{\mathbf{z}}_k||_2^2 + \beta \sum_{k=1}^{K} ||\mathbf{t}_k||_1 \\ \text{subject to} \quad & ||\mathbf{s}_k||_2^2 \leq 1 \text{ for } k = 1 \dots K \\ & \mathbf{s}_k = \mathbf{\Phi}^T \hat{\mathbf{d}}_k \text{ for } k = 1 \dots K \\ & \mathbf{z}_k = \mathbf{t}_k \text{ for } k = 1 \dots K. \end{aligned} \tag{3}$$

$\mathbf{\Phi}$ is a $D \times M$ submatrix of the Fourier matrix $\mathbf{F} = [\mathbf{\Phi}, \mathbf{\Phi}_\perp]$ that corresponds to a small spatial support of the filter where $M \ll D$. Fourier convolution is not exactly equivalent to spatial convolution due to the different manner in which boundary effects are handled. Ignoring these for the moment (see *Boundary Effects* on mitigating these differences) the objective in Equation (3) is equivalent to the original objective in Equation (2).

In our proposed Fourier formulation $\hat{\mathbf{d}}_k$ is a D dimensional vector like $\hat{\mathbf{x}}$ and $\hat{\mathbf{z}}_k$, whereas in the original spatial formulation in Equation (2) $\mathbf{d}_k \in \mathbb{R}^M$ is of a significantly smaller dimensionality to $M \ll D$ corresponding to its small spatial support. We enforce the smaller spatial constraint through the auxiliary variable $\mathbf{s}$, which now becomes separable (in terms of variables) to the convolutional component of the objective. In a similar spirit to Zeiler *et al.*'s original approach, we also separate the L_1 penalty term from the convolutional component of the objective using the auxiliary variable $\mathbf{t}$.

Augmented Lagrangian: In this paper we propose to handle the introduced equality constraints through an augmented Lagrangian approach [3]. The augmented Lagrangian of our proposed objective can be formed as,

$$\begin{aligned} \mathcal{L}(\mathbf{d}, \mathbf{s}, \mathbf{z}, \mathbf{t}, \boldsymbol{\lambda}_\mathbf{s}, \boldsymbol{\lambda}_\mathbf{t}) = & \\ & \frac{1}{2D}||\hat{\mathbf{x}} - \sum_{k=1}^{K} \hat{\mathbf{d}}_k \odot \hat{\mathbf{z}}_k||_2^2 + \beta||\mathbf{t}||_1 \\ & + \boldsymbol{\lambda}_\mathbf{s}^T(\mathbf{s} - [\mathbf{\Phi}^T \otimes \mathbf{I}_K]\hat{\mathbf{d}}) + \boldsymbol{\lambda}_\mathbf{t}^T(\mathbf{z} - \mathbf{t}) \\ & + \frac{\mu_\mathbf{s}}{2}||\mathbf{s} - [\mathbf{\Phi}^T \otimes \mathbf{I}_K]\hat{\mathbf{d}}||_2^2 \\ & + \frac{\mu_\mathbf{t}}{2}||\mathbf{z} - \mathbf{t}||_2^2 \\ \text{subject to} \quad & ||\mathbf{s}_k||_2^2 \leq 1 \text{ for } k = 1 \dots K. \end{aligned} \tag{4}$$

where $\boldsymbol{\lambda}_\mathbf{p}$ and $\mu_\mathbf{p}$ denote the Lagrange multiplier and penalty weighting for the two auxiliary variables $\mathbf{p} \in \{\mathbf{s}, \mathbf{t}\}$ respectively. Augmented Lagrangian Methods (ALMs) are not new to learning and computer vision, and have recently been used to great effect in a number of applications [3, 6]. Specifically, the Alternating Direction Method of Multipliers (ADMMs) has provided a simple but powerful algorithm that is well suited to distributed convex optimization for large learning and vision problems. A full description of ADMMs is outside the scope of this paper (readers are encouraged to inspect [3] for a full treatment and review), but they can be loosely interpreted as applying a Gauss-Seidel optimization strategy to the augmented Lagrangian objective. Such a strategy is advantageous as it often leads to extremely efficient subproblem decompositions. Del Bue *et al.* [6] recently applied this approach with success to a variety of bilinear forms common to computer vision (*e.g.* structure from motion, photo-

metric stereo, image registration, etc.). A full description of our proposed algorithm is presented in Algorithm 1. We detail each of the subproblems following:

Subproblem z:

$$\begin{aligned}\mathbf{z}^* &= \arg\min_{\mathbf{z}} \mathcal{L}(\mathbf{z};\mathbf{d},\mathbf{s},\mathbf{t},\boldsymbol{\lambda}_\mathbf{s},\boldsymbol{\lambda}_\mathbf{t}) && (5)\\ &= \mathcal{F}^{-1}\{\arg\min_{\hat{\mathbf{z}}} \frac{1}{2}||\hat{\mathbf{x}}-\hat{\mathbf{D}}\hat{\mathbf{z}}||_2^2 + \\ &\quad \hat{\boldsymbol{\lambda}}_\mathbf{t}^T(\hat{\mathbf{z}}-\hat{\mathbf{t}}) + \frac{\mu_\mathbf{t}}{2}||\hat{\mathbf{z}}-\hat{\mathbf{t}}||_2^2\} && (6)\\ &= \mathcal{F}^{-1}\{(\hat{\mathbf{D}}^T\hat{\mathbf{D}}+\mu_\mathbf{t}\mathbf{I})^{-1}(\hat{\mathbf{D}}^T\hat{\mathbf{x}}+\mu_\mathbf{t}\hat{\mathbf{t}}-\hat{\boldsymbol{\lambda}}_\mathbf{t})\} && (7)\end{aligned}$$

where $\hat{\mathbf{D}} = [\text{diag}(\hat{\mathbf{d}}_1),\ldots,\text{diag}(\hat{\mathbf{d}}_K)]$. Although the size of the matrix $\hat{\mathbf{D}}$ is $KD \times KD$, it is sparse banded, and an efficient variable reordering exists (see Figure 3) such that the optimal $\mathbf{z}^*$ can be found as the solution to D independent $K \times K$ linear systems.

Subproblem t:

$$\begin{aligned}\mathbf{t}^* &= \arg\min_{\mathbf{t}} \mathcal{L}(\mathbf{t};\mathbf{d},\mathbf{s},\mathbf{z},\boldsymbol{\lambda}_\mathbf{s},\boldsymbol{\lambda}_\mathbf{t}) && (8)\\ &= \arg\min_{\mathbf{t}} \frac{\mu_\mathbf{t}}{2}||\mathbf{z}-\mathbf{t}||_2^2 + \boldsymbol{\lambda}_\mathbf{t}^T(\mathbf{z}-\mathbf{t}) + \beta||\mathbf{t}||_1 && (9)\end{aligned}$$

Unlike subproblem $\mathbf{z}$, the solution to $\mathbf{t}$ cannot be efficiently computed in the Fourier domain, since the L_1 norm is not rotation invariant. Solving for $\mathbf{t}$ first requires projecting $\hat{\mathbf{z}}$ and $\hat{\boldsymbol{\lambda}}_\mathbf{t}$ back into the spatial domain. Since the objective in Equation (9) does not contain any rotations of the data, each element of $\mathbf{t} = [t_1,\ldots,t_D]^T$ can be solved independently,

$$t^* = \arg\min_t \beta|t| + \lambda(z-t) + \frac{\mu}{2}(z-t)^2 \qquad (10)$$

where the optimal solution for each t can be found efficiently using the shrinkage function,

$$t^* = \text{sgn}\left(z+\frac{\lambda_\mathbf{t}}{\mu_\mathbf{t}}\right)\cdot\max\left\{|z+\frac{\lambda_\mathbf{t}}{\mu_\mathbf{t}}|-t,0\right\} \quad . \qquad (11)$$

Subproblem d:

$$\begin{aligned}\mathbf{d}^* &= \arg\min_{\mathbf{s}} \mathcal{L}(\mathbf{d};\mathbf{s},\mathbf{z},\mathbf{t},\boldsymbol{\lambda}_\mathbf{s},\boldsymbol{\lambda}_\mathbf{t}) && (12)\\ &= \mathcal{F}^{-1}\{\arg\min_{\hat{\mathbf{d}}} \frac{1}{2}||\hat{\mathbf{x}}-\hat{\mathbf{Z}}\hat{\mathbf{d}}||_2^2 + \\ &\quad \hat{\boldsymbol{\lambda}}_\mathbf{s}^T(\hat{\mathbf{d}}-\hat{\mathbf{s}}) + \frac{\mu_\mathbf{s}}{2}||\hat{\mathbf{d}}-\hat{\mathbf{s}}||_2^2\} && (13)\\ &= \mathcal{F}^{-1}\{(\hat{\mathbf{Z}}^T\hat{\mathbf{Z}}+\mu_\mathbf{s}\mathbf{I})^{-1}(\hat{\mathbf{Z}}^T\hat{\mathbf{x}}+\mu_\mathbf{s}\hat{\mathbf{s}}-\hat{\boldsymbol{\lambda}}_\mathbf{s})\} && (14)\end{aligned}$$

where $\hat{\mathbf{Z}} = [\text{diag}(\hat{\mathbf{z}}_1),\ldots,\text{diag}(\hat{\mathbf{z}}_K)]$. In a similar fashion to subproblem $\mathbf{z}$, even though the size of the matrix $\hat{\mathbf{Z}}$ is $KD\times KD$, a similar variable reordering exists such that finding the optimal $\mathbf{d}^*$ involves solution to D independent $K \times K$ linear systems.

Subproblem s:

$$\begin{aligned}\mathbf{s}^* &= \arg\min_{\mathbf{s}} \mathcal{L}(\mathbf{s};\mathbf{d},\mathbf{z},\mathbf{t},\boldsymbol{\lambda}_\mathbf{s},\boldsymbol{\lambda}_\mathbf{t}) && (15)\\ &= \arg\min_{\mathbf{s}} \frac{\mu_\mathbf{s}}{2}||\hat{\mathbf{d}}-[\boldsymbol{\Phi}^T\otimes\mathbf{I}_K]\mathbf{s}||_2^2 + \\ &\quad \hat{\boldsymbol{\lambda}}_\mathbf{s}^T(\hat{\mathbf{d}}-[\boldsymbol{\Phi}^T\otimes\mathbf{I}_K]\mathbf{s}) \\ &\quad \text{subject to } ||\mathbf{s}_k||_2^2 \le 1 \text{ for } k=1\ldots K \ . && (16)\end{aligned}$$

In its general form, solving Equation (16) efficiently is problematic as it is a quadratically constrained quadratic programming (QCQP) problem. Fortunately due to the kronecker product with the identity matrix $\mathbf{I}_K$ it can be broken down into K independent problems,

$$\begin{aligned}\mathbf{s}_k^* &= \arg\min_{\mathbf{s}_k} \frac{\mu_\mathbf{s}}{2}||\hat{\mathbf{d}}_k-\boldsymbol{\Phi}^T\mathbf{s}_k||_2^2 + \hat{\boldsymbol{\lambda}}_{\mathbf{s}k}^T(\hat{\mathbf{d}}_k-\boldsymbol{\Phi}^T\mathbf{s}_k) \\ &\quad \text{subject to } ||\mathbf{s}_k||_2^2 \le 1 \quad . && (17)\end{aligned}$$

Further, since $\boldsymbol{\Phi}$ is orthonormal (ignoring the $\sqrt{D}$ scaling factor) projecting the optimal solution to the unconstrained problem (see Figure 2) can be found efficiently through,

$$\mathbf{s}_k^* = \begin{cases} ||\tilde{\mathbf{s}}_k||_2^{-1}\tilde{\mathbf{s}}_k, & \text{if } ||\tilde{\mathbf{s}}_k||_2^2 \ge 1 \\ \tilde{\mathbf{s}}_k, & \text{otherwise}\end{cases} \qquad (18)$$

where,

$$\tilde{\mathbf{s}}_k = (\mu_\mathbf{s}\boldsymbol{\Phi}\boldsymbol{\Phi}^T)^{-1}(\boldsymbol{\Phi}\hat{\mathbf{d}}_k + \boldsymbol{\Phi}\hat{\boldsymbol{\lambda}}_{\mathbf{s}k}) \quad . \qquad (19)$$

Finally, the solution to Equation (19) can be found very efficiently using

$$\tilde{\mathbf{s}}_k = \mathcal{M}\Big\{\frac{1}{\mu_\mathbf{s}}\sqrt{D}^{-1}\big(\mathcal{F}^{-1}\{\hat{\mathbf{d}}_k\} + \mathcal{F}^{-1}\{\hat{\boldsymbol{\lambda}}_{\mathbf{s}k}\}\big)\Big\} \qquad (20)$$

where $\mathcal{F}^{-1}\{\}$ is the inverse FFT and $\mathcal{M}\{\} : \mathbb{R}^D \to \mathbb{R}^M$ is a mapping function that preserves only $M \ll D$ active values relating to the small spatial structure of the estimated filter. As a result one never needs to actually construct the sub-matrix $\boldsymbol{\Phi}$ in order to estimate $\mathbf{s}_k$.

Lagrange Multiplier Update:

$$\boldsymbol{\lambda}_\mathbf{t}^{(i+1)} \leftarrow \boldsymbol{\lambda}_\mathbf{t}^{(i)} + \mu_\mathbf{t}(\mathbf{z}^{(i+1)} - \mathbf{t}^{(i+1)}) \qquad (21)$$

$$\boldsymbol{\lambda}_\mathbf{s}^{(i+1)} \leftarrow \boldsymbol{\lambda}_\mathbf{s}^{(i)} + \mu_\mathbf{s}(\mathbf{d}^{(i+1)} - \mathbf{s}^{(i+1)}) \qquad (22)$$

Penalty Update: Superlinear convergence of the ADMM may be achieved if $\mu^{(i)} \to \infty$ [18]. In practice, we limit the value of μ to avoid poor conditioning and numerical errors. Specifically, we adopt the following update strategy:

$$\mu^{(i+1)} = \begin{cases} \tau\mu^{(i)} & \text{if } \mu^{(i)} < \mu_{max} \\ \mu^{(i)} & \text{otherwise}\end{cases} \qquad (23)$$

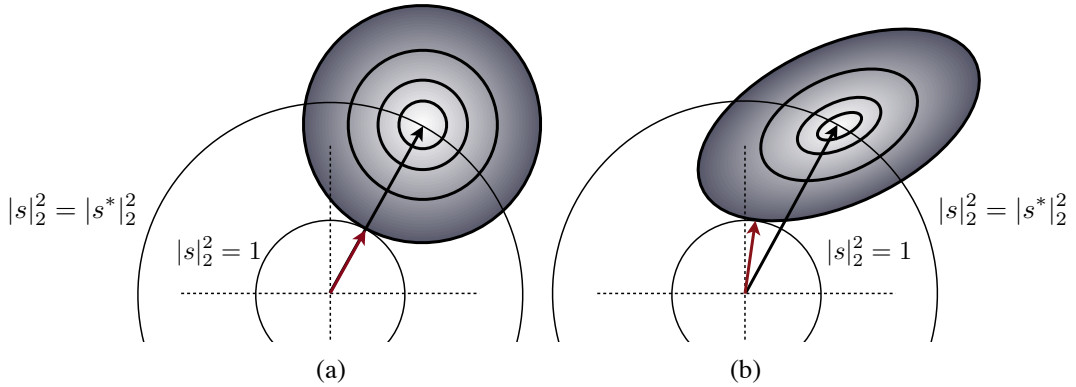

Figure 2. (a) Solving the isotropic ridge regression problem of Equation (17) with a trust region (red arrow) is equivalent to projecting the optimal unconstrained solution (black arrow) onto the unit sphere. (b) The same cannot be said for the general case of anisotropic problems, where projection of the unconstrained solution is different to the trust region solution.

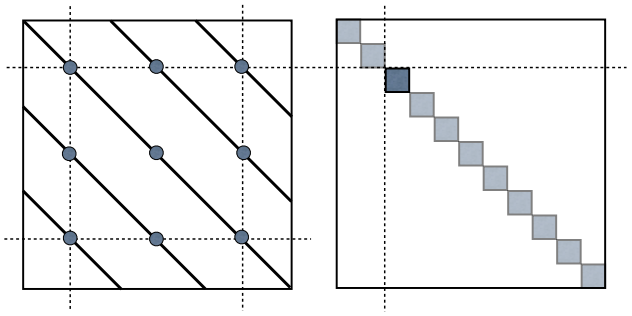

Figure 3. Variable reordering in the sparse banded systems of subproblems $\mathbf{s}$ and $\mathbf{z}$. Each output pixel is dependent only on the K filters, thus each distinct set of K pixels can be found by reordering and solving a $K \times K$ linear system.

We observed good convergence for $\tau \in [1.01\ 1.5]$, and $\mu_{max} = 1.0e^5$. Larger values of τ enforced the equality constraints quickly, but at the expense of primal feasibility. Smaller values of τ led to slow, linear convergence. Adaptive strategies can also be entertained to balance the rate of primal and dual convergence [22].

Boundary effects: Zeiler *et al.* state that one reason for avoiding convolution in the Fourier domain is the boundary effects that are introduced. Circular convolution, implied by the Fourier convolution theorem, assumes periodic extension of the signal. To determine the degree to which boundary effects were corrupting our solution, we replaced circular convolution in our method with symmetric convolution which assumes symmetric extension of the signal across the boundaries - a more reasonable real-world assumption for natural signals such as images and speech. The resulting filters learned not only looked qualitatively similar to those from circular convolution, but the final objective value was also comparable (within a 1×10^{-3} margin of error).

We surmise that the large size of the input images (100×100 or larger) compared to the support of the

Algorithm 1 Convolutional Sparse Coding using Fourier Subproblems and ADMMs

Intialize $\mathbf{z}^{(0)}$, $\mathbf{t}^{(0)}$, $\mathbf{s}^{(0)}$, $\boldsymbol{\lambda}_{\mathbf{s}}^{(0)}$, $\boldsymbol{\lambda}_{\mathbf{t}}^{(0)}$.
Perform FFT $\mathbf{z}^{(0)}$, $\mathbf{t}^{(0)}$, $\mathbf{s}^{(0)}$, $\boldsymbol{\lambda}_{\mathbf{s}}^{(0)}$, $\boldsymbol{\lambda}_{\mathbf{t}}^{(0)} \rightarrow \hat{\mathbf{z}}^{(0)}$, $\hat{\mathbf{t}}^{(0)}$, $\hat{\mathbf{s}}^{(0)}$, $\hat{\boldsymbol{\lambda}}_{\mathbf{s}}^{(0)}$, $\hat{\boldsymbol{\lambda}}_{\mathbf{t}}^{(0)}$.
$i = 0$
repeat
  Solve for $\hat{\mathbf{d}}^{(i+1)}$ using Eqn. (14) given $\hat{\mathbf{z}}^{(i)}, \hat{\mathbf{s}}^{(i)}, \hat{\boldsymbol{\lambda}}_{\mathbf{s}}^{(i)}$.
  Perform inverse FFT $\mathcal{F}^{-1}\{\hat{\mathbf{d}}^{(i)}\} \rightarrow \mathbf{d}^{(i+1)}$.
  Solve for $\tilde{\mathbf{s}}^{(i+1)}$ using Eqn. (20) given $\mathbf{d}^{(i+1)}$.
  Preserve only the M local pixels in $\mathbf{d}_k^{(i+1)}$ and scale by $\frac{1}{\sqrt{D}}$ to estimate $\tilde{\mathbf{s}}_k^{(i+1)} = \frac{1}{\sqrt{D}}\mathcal{M}\{\mathbf{d}_k^{(i+1)}\}$ - Eqn. (16) - for all $k = 1 \ldots K$.
  Project $\tilde{\mathbf{s}}_k^{(i+1)}$ onto the isotropic trust region constraint to estimate $\mathbf{s}_k^{(i+1)}$ for all $k = 1 \ldots K$.
  Solve for $\hat{\mathbf{z}}^{(i+1)}$ using Eqn. (7) given $\hat{\mathbf{t}}^{(i)}, \hat{\boldsymbol{\lambda}}_{\mathbf{t}}^{(i)}, \hat{\mathbf{d}}^{(i+1)}$
  Perform inverse FFT $\mathcal{F}^{-1}\{\hat{\mathbf{z}}^{(i+1)}\} \rightarrow \mathbf{z}^{(i+1)}$.
  Solve for $\mathbf{t}^{(i+1)}$ using Eqn. (9) given $\mathbf{z}^{(i+1)}, \boldsymbol{\lambda}_{\mathbf{t}}^{(i)}$
  Update Lagrange multiplier vector $\boldsymbol{\lambda}_{\mathbf{t}}^{(i+1)} \leftarrow \boldsymbol{\lambda}_{\mathbf{t}}^{(i)} + \mu_{\mathbf{t}}(\mathbf{z}^{(i+1)} - \mathbf{t}^{(i+1)})$
  Update Lagrange multiplier vector $\boldsymbol{\lambda}_{\mathbf{s}}^{(i+1)} \leftarrow \boldsymbol{\lambda}_{\mathbf{s}}^{(i)} + \mu_{\mathbf{s}}(\mathbf{d}^{(i+1)} - \mathbf{s}^{(i+1)})$
  Perform FFT on $\mathbf{t}$, $\mathbf{s}$, $\boldsymbol{\lambda}_{\mathbf{s}}$, $\boldsymbol{\lambda}_{\mathbf{t}} \rightarrow \hat{\mathbf{t}}$, $\hat{\mathbf{s}}$, $\hat{\boldsymbol{\lambda}}_{\mathbf{s}}$, $\hat{\boldsymbol{\lambda}}_{\mathbf{t}}$.
  $i = i + 1$
until $\mathbf{z}, \mathbf{s}, \mathbf{d}, \mathbf{t}$ has converged

filters being learned (12×12) results in negligible contributions from pixels affected by boundary effects to the overall objective. For large support filters, the Discrete Fourier Transform (DFT) can be replaced by the Discrete Cosine Transform (DCT) which diagonalizes symmetric convolution, and like a DFT can be applied extremely efficiently [17]. Kavukcuoglu *et al.* point out, however, that even spatial convolution introduces boundary effects since the boundary pixels have less contributions than inner pixels, so no convolution method is truly exempt from boundary effects.

Complexity Analysis: Here we briefly analyse the complexity properties of our method versus current first order methods (*i.e.* Zeiler *et al.*). We consider the cost of evaluating a single iteration of subproblems. Both methods are linear in the number of training examples, so we assume a single example for clarity. The cost of our method is dominated by the solution to the linear systems arising from the $\mathbf{z}$ and $\mathbf{d}$ subproblems.

$$\underbrace{K^3D}_{\text{Linear systems}} + \underbrace{KDlog(D)}_{\text{Fourier transforms}} + \underbrace{KD}_{\text{Soft thresholding}} \tag{24}$$

$$= \mathcal{O}(K^3D) \tag{25}$$

Because Zeiler's method is iterative within the $\mathbf{z}$ and $\mathbf{d}$ subproblems, the cost of convolution is multiplicative with the cost of the conjugate gradient method. Although our method uses a more costly direct solver, the contribution of convolution is *additive* to the overall complexity.

$$\underbrace{KD}_{\text{Conjugate gradients}} \times \underbrace{KDM}_{\text{Spatial convolution}} + \underbrace{KD}_{\text{Soft thresholding}} \tag{26}$$

$$= \mathcal{O}(K^2D^2M)\ , \tag{27}$$

where M corresponds to the support of the filter. Our method has better asymptotic performance than Zeiler's under very mild conditions: $K < DM$. We show following that our algorithm not only has good asymptotic properties, but is also fast in practice.

3. Experiments

We show three key results of our algorithm: (i) it has faster convergence than current state of the art methods, (ii) it consistently converges to local minima of equal quality to these existing methods, and (iii) that when applied to natural images our method produces structured Gabor-like and higher complexity filters.

We compare our method to the underlying learning method of Zeiler *et al.* [27]. We use their "Fruit" dataset consisting of 10 images, apply local contrast normalization and select random 50×50 subimages from each. We perform two experiments, first holding the number of training examples fixed whilst varying the number of filters learned, then visa versa. We use the relative residual between two iterations as the stopping criteria. The results are shown in Figure 4.

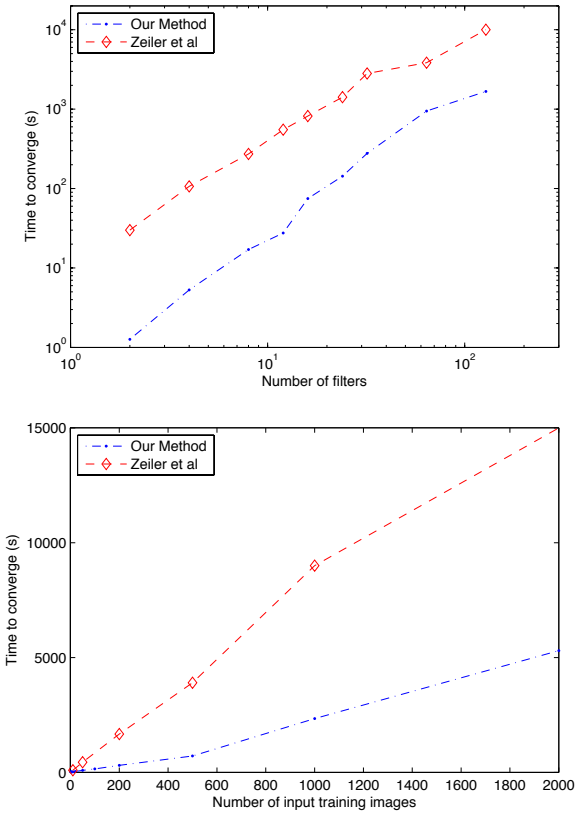

Figure 4. Time to convergence as a function of (left) the number of filters learned with fixed number of input images, and (right) the number input images with fixed number of filters.

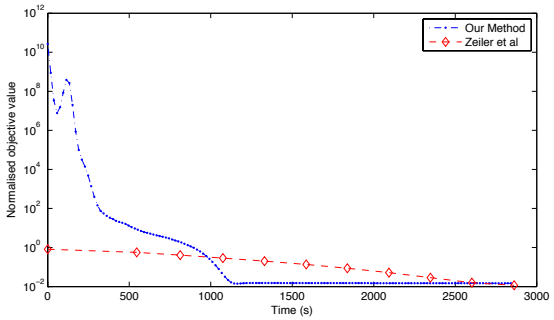

Figure 5. Comparison of objective when learning 64 filters from the experiments of Figure 4. Our objective starts at a much larger value than Zeiler *et al.*'s, due to added Lagrange multipliers and penalty terms, but quickly converges to a good solution.

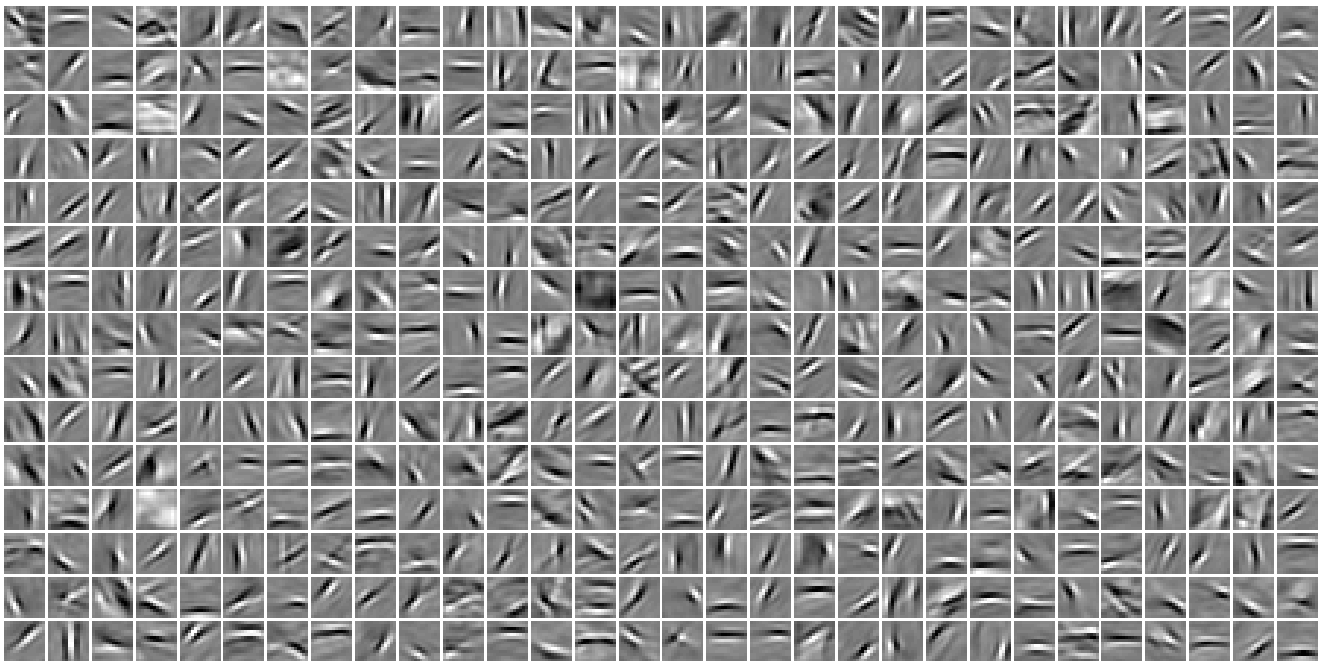

Figure 6. A representative set of 450 filters learned on a laptop using a collection of whitened natural images. The set contains Gabor-like components, as well as more expressive centre-surround and cross-like components which appear since the spatially-invariant learning strategy produces less spatially shifted versions of the filters.

Our method (blue line) consistently converges nearly an order of magnitude faster than Zeiler *et al.*'s method (red line) with respect to the number of filters being learned. This is largely due to two contributing factors: (i) the direct methods we use to solve each subproblem have super-linear convergence properties, whereas the conjugate gradient method employed by Zeiler *et al.* is limited to linear convergence, and (ii) convolution in the Fourier domain involves a simple Hadamard product whereas convolutions must be explicitly recomputed for each iteration of conjugate gradients.

Our method also has better scalability for increasing number of input examples. We do not use any batching techniques during training: learning is performed jointly across the entire training set.

In Figure 5 we show a representative trial from the Experiments of Figure 4. System variables $\mathbf{d}$, $\mathbf{s}$, $\mathbf{z}$ and $\mathbf{t}$ are randomly initialized with Gaussian noise. The initial Lagrange multipliers are set to zero. Our method starts at a much larger objective value, due to the additional Lagrange multiplier and penalty terms. The objective quickly decreases however, as these terms vanish and the equality constraints are satisfied. The overall curve of our objective is typical of ADMMs: steep convergence to begin with, followed by flat-lining and minimal convergence beyond that point.

Applying sparse coding algorithms to the original Olshausen and Field dataset [15] has become a standard "sanity check", to ensure that the method is capable of producing Gabor-like oriented edge filters. We do the same by dividing the original 512×512 pixel images into 16 subimages, for a total training set of 160 images each of 128×128 pixels. A visualization of the learned filters is presented in Figure 6.

While our convolutional algorithm produces some Gabor-like responses, it also has a greater expression of non-Gabor filters which are tailored more towards the semantics of the dataset. Figure 1 shows a compelling example of this artifact, with one of the filters clearly synthesizing an "eye" feature from a set of unaligned lions.

4. Conclusions

We presented a method for solving convolutional sparse coding problems in a fast manner through quad-decomposition of the original objective into subproblems that have an efficient parameterization in the Fourier domain. These components working in union allow us to solve the rotation invariant L_1 subproblem for each index independently using soft thresholding, and transform the quadratically constrained filter update equation to an unconstrained isotropic system. As filter support size increases, the appeal of Fourier convolution becomes more apparent, and where boundary effects are problematic the Fourier transform can be seamlessly replaced by the Discrete Cosine Transform.

Acknowledgements

This research was supported under the Australian Research Council's Discovery Early Career Researcher Award (project DE130101775) and Future Fellowship Award (project FT0991969).

References

[1] A. Beck and M. Teboulle. A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems. *SIIMS*, 2009. 1, 2

[2] D. Bertsekas, A. Nedic, and A. Ozdaglar. *Convex Analysis and Optimization*. Athena Scientific, 2003. 2

[3] S. Boyd. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Foundations and Trends in Machine Learning*, 2010. 3

[4] E. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *Information Theory*, 2006. 1

[5] S. S. Chen, D. L. Donoho, and M. a. Saunders. Atomic Decomposition by Basis Pursuit. *SIAM Review*, 2001. 1

[6] A. Del Bue, J. Xavier, L. Agapito, and M. Paladini. Bilinear Modelling via Augmented Lagrange Multipliers (BALM). *PAMI*, Dec. 2011. 3

[7] J. Eggert, H. Wersing, and K. Edgar. Transformation-invariant representation and NMF. *Neural Networks*, 2004. 2

[8] W. Hashimoto and K. Kurata. Properties of basis functions generated by shift invariant sparse representations of natural images. *Biological Cybernetics*, 2000. 2

[9] K. Kavukcuoglu, P. Sermanet, Y.-l. Boureau, K. Gregor, M. Mathieu, and Y. LeCun. Learning convolutional feature hierarchies for visual recognition. *NIPS*, 2010. 2

[10] Y. LeCun and L. Bottou. Gradient-based learning applied to document recognition. 1998. 2

[11] H. Lee, A. Battle, R. Raina, and A. Ng. Efficient sparse coding algorithms. *NIPS*, 2007. 1

[12] M. Lewicki and T. Sejnowski. Coding time-varying signals using sparse, shift-invariant representations. *NIPS*, 1999. 2

[13] J. Mairal, F. Bach, and J. Ponce. Online learning for matrix factorization and sparse coding. *JMLR*, 2010. 1

[14] M. Morup, M. N. Schmidt, and L. K. Hansen. Shift invariant sparse coding of image and music data. *JMLR*, 2008. 2

[15] B. Olshausen and D. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 1996. 2, 7

[16] B. A. Olshausen and D. J. Field. Sparse Coding with an Overcomplete Basis Set: A Strategy Employed by V1? *Science*, 1997. 2

[17] A. V. Oppenheim, A. S. Willsky, and with S. Hamid. *Signals and Systems (2nd Edition)*. Prentice Hall, 1996. 2, 6

[18] R. Rockafellar. Monotone operators and the proximal point algorithm. *SICON*, 1976. 4

[19] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society*, 1996. 1

[20] S. Ullman. *High-Level Vision: Object Recognition and Visual Cognition*. MIT Press, 1996. 2

[21] E. Wachsmuth, M. W. Oram, and D. I. Perrett. Recognition of objects and their component parts: responses of single units in the temporal cortex of the macaque. *Cerebral Cortex*, 1994. 2

[22] S. Wang and L. Liao. Decomposition Method with a Variable Parameter for a Class of Monotone Variational Inequality Problems. *JOTA*, 2001. 5

[23] H. Wersing, J. Eggert, and K. Edgar. Sparse coding with invariance constraints. *ICANN*, 2003. 2

[24] J. Yang, K. Yu, and Y. Gong. Linear spatial pyramid matching using sparse coding for image classification. *CVPR*, 2009. 1

[25] J. Yang and Y. I. N. Zhang. Alternating direction algorithms for in compressive sensing. Technical report, 2010. 2

[26] M. Zeiler and R. Fergus. Learning Image Decompositions with Hierarchical Sparse Coding. Technical report, 2010. 2

[27] M. Zeiler, D. Krishnan, G. Taylor, and R. Fergus. Deconvolutional networks. *CVPR*, 2010. 2, 6