AN EFFICIENT FIRST-ORDER METHOD FOR $\ell_1$ COMPRESSION OF IMAGES

Joachim Dahl†, Jan Østergaard‡, Tobias Lindstrøm Jensen‡, Søren Holdt Jensen‡.

†Anybody Technology A/S
Niels Jernesvej 12
Aalborg, Denmark

‡Aalborg University
Department of Electronic Systems
Niels Jernesvej 12, Aalborg, Denmark

ABSTRACT
We consider the problem of lossy compression of images using sparse representations from overcomplete dictionaries. This problem is in principle easy to solve using standard algorithms for convex programming, but often the large dimensions render such an approach intractable. We present a highly efficient method based on recently developed first-order methods, which enables us to compute sparse approximations of entire images with modest time and memory consumption.

Index Terms— Basis pursuit, sparse approximations, image compression, convex optimization, first-order optimization algorithms.

1. INTRODUCTION
Recently there has been tremendous resurgence of interest in sparse estimation techniques for signal processing based on $\ell_1$ heuristics, see e.g., the recent issue devoted to compressive sampling [1]. The theory is by now well-established and much is known about cases where the $\ell_1$ heuristic coincides with the solution to the otherwise intractable minimum cardinality solution, see [2] and references therein.

Perhaps the best known method for sparse reconstruction is the so-called basis pursuit algorithm, which solves the problem

$$\begin{align*}
\min_{z} & \quad \|z\|_1 \\
\text{subject to} & \quad \|Dz - y\|_2 \leq \delta
\end{align*}$$

where $D \in \mathbb{R}^{M \times N}$ represents an overcomplete dictionary ($N > M$), $z \in \mathbb{R}^N$ is the variable, and $y \in \mathbb{R}^M$ is the signal we wish to decompose into (as few as possible) dictionary elements or basis vectors of $D$.

In this paper, we consider efficient large-scale implementations of a variation, namely the $\ell_1$ compression problem

$$\begin{align*}
\min_{z} & \quad \|z\|_1 \\
\text{subject to} & \quad \|Dz - y\|_2 \leq \delta
\end{align*}$$

where $\delta > 0$ is a given fidelity or reconstruction error. For a single orthogonal basis (i.e., $D^T D = I$) the $\ell_1$-compression problem can be rewritten as

$$\begin{align*}
\min_{z} & \quad \|z\|_1 \\
\text{subject to} & \quad \|z - D^T y\|_2 \leq \delta,
\end{align*}$$

which can be solved very efficiently even for very large instances (due to the simple constraints) using a plethora of methods, e.g., standard primal-dual interior-point methods [3]. Many primal-dual interior-point algorithms have proven worst-case iteration complexity bound $O(\log(1/\epsilon))$, meaning that less than $O(\log(1/\epsilon))$ iterations are required to produce an $\epsilon$-optimal solution $z_\epsilon$ such that $\|z_\epsilon\|_1 - \|z^\star\|_1 < \epsilon$.

When $D$ is not orthogonal the dimensions of typical images are far too large for primal-dual interior-point implementations of the $\ell_1$ compression problem. This is because each iteration requires the solution of an $N \times N$ dense system of linear equations.

Alternatively, we could apply standard large-scale gradient or sub-gradient methods with $O(1/\epsilon^2)$ iteration complexity, where each iteration is much cheaper compared to primal-dual interior-point methods. However, the slow convergence rate renders such an approach unattractive.

Instead, in this paper, we apply recently developed first-order methods developed by Nesterov in a series of papers [4, 5] which achieve remarkable $O(1/\epsilon)$ iteration complexity. These and related modern gradient methods have recently been surveyed by Tseng [6] in the context of minimization of sufficiently smooth functions with $O(\sqrt{1/\epsilon})$ iteration complexity. Nesterov’s method has recently been applied to a series of large-scale problems [7, 8] including total-variation image restoration [9, 10], but the application of $\ell_1$ compression with overcomplete dictionaries causes additional difficulties and has so far not been studied.

The remaining paper is structured as follows: in Sec. 2 we discuss Nesterov’s method tailored specifically for the $\ell_1$ compression problem. In Sec. 3 we give numerical examples of $\ell_1$ compression with several bases via Nesterov’s method for full-size images. Finally, we give discussions in Sec. 4.

2. NESTEROV’S METHOD FOR $\ell_1$ COMPRESSION
Nesterov’s method is an efficient first-order method for solving saddle-point problems of the form,

$$\min_{x \in Q_1} \max_{y \in Q_2} u^T Ax$$

where $Q_1$ and $Q_2$ are bounded convex sets. From the definition of the induced norm

$$\|x\|_p = \max_{\|y\|_q \leq 1} u^T x,$$

where $\|x\|_q$ is the dual-norm (i.e., $\|x\|_p = (\|x\|_q^p)^{1/p}$), the $\ell_1$ compression problem can be written as such a saddle-point problem,

$$\min_{\|Dz - y\|_2 \leq \delta} \max_{\|z\|_1 \leq 1} u^T z.$$

Since the dictionary $D$ is generally overcomplete, the primal set $\{z \mid \|Dz - y\|_2 \leq \delta\}$ is not bounded, and we cannot immediately
use Nesterov’s method. Instead we assume that the overcomplete
dictionary is comprised of a set of orthogonal (inverse) transforms
\( D_1^{-1}, D_2^{-1}, \ldots, D_K^{-1} \),
where \( z_i \in \mathbb{R}^M, i = 1, \ldots, K \) with \( K = N/M \) denote the different
subblocks of the vector of transform coefficients. To make the primal
set bounded we introduce the arbitrary bound
\[
\|(z_2, \ldots, z_K)\|_2 \leq \gamma
\]
which does not alter the solution provided \( \gamma \) is chosen sufficiently
large. The resulting problem
\[
\min_{x \in Q_1} \max_{u \in Q_2} z^T u
\]
where
\[
Q_1 = \{ z \mid \|D(z) - y\|_2 \leq \delta, \| (z_2, \ldots, z_K)\|_2 \leq \gamma \}
Q_2 = \{ u \mid \|u\|_{\infty} \leq 1 \}
\]
fits the framework for Nesterov’s method, but it appears difficult to
solve it efficiently in this form; the complexity of \( Q_1 \) makes the
subproblems 2) and 3) in Nesterov’s method (discussed later in the
paper) as difficult to solve as the original \( \ell_1 \) compression problem.
Instead we introduce a transform pair,
\[
W = \begin{bmatrix}
D_1 & -D_1D_2^{-1} & \cdots & -D_1D_K^{-1} \\
0 & I & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & I
\end{bmatrix}
\]
\[
W^{-1} = \begin{bmatrix}
D_1^{-1} & D_2^{-1} & \cdots & D_K^{-1} \\
0 & I & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & I
\end{bmatrix}
\]
meaning that
\[
W(z) = \left( D_1(z_1 - \sum_{i=2}^{K} D_i^{-1}(z_i)), z_2, \ldots, z_K \right)
\]
\[
W^{-1}(z) = \left( \sum_{i=1}^{K} D_i^{-1}(z_i), z_2, \ldots, z_K \right).
\]
We next introduce a change of variables,
\[
x = W^{-1}(z),
\]
and with a slight abuse of notation denote \( W(x), W^{-1}(x) \) and
\( W^{\text{adj}}(x) \) by \( Wx, W^{-1}x \) and \( WTx \), respectively, for the benefit
of a simpler notation in the following. We can now rewrite the
compression problem in a convenient form, that is
\[
\min_{x \in Q_1'} \max_{u \in Q_2} u^T Wx
\]
where
\[
Q_1' = \{ x \mid \| x_1 - y \|_2 \leq \delta, \| (x_2, \ldots, x_K)\|_2 \leq \gamma \}
Q_2 = \{ u \mid \|u\|_{\infty} \leq 1 \}
\]
which is easier to solve using Nesterov’s method; in particular, the
solutions to subproblems 2) and 3) in the following can be found
analytically with complexity \( O(N) \). The associated dual problem of
(3) is
\[
\max_{u \in Q_2} \min_{x \in Q_1'} -\delta \| (W^T u)_1 \|_2 + y^T (W^T u)_1 - \gamma \| (W^T u)_2 \|_2 
\]
where \( (W^T u)_1 \) denotes the first block of \( (W^T u) \) and \( (W^T u)_2 \)
denotes the remaining the \( K - 1 \) blocks, partitioned conformly with
\( x \) and \( z \).
In the following we review the steps in Nesterov’s method for
solving (3). To both sets \( Q_1' \) and \( Q_2 \) we associate so-called
prox-functions. As a prox-function for \( Q_1' \) we choose
\[
d_1(x) = (1/2)\|x_1 - y\|_2^2 + (1/2) \sum_{i=2}^{K} \|x_i\|_2^2.
\]
The prox-function is upper bounded by
\[
\Delta_1 := \max_{x \in Q_1'} d_1(x) = \frac{\gamma^2 + \delta^2}{2}.
\]
Similarly, as a prox-function for \( Q_2 \) we choose
\[
d_2(u) = (1/2)\|u\|_2^2
\]
and upper bound
\[
\Delta_2 := \max_{u \in Q_2} d_2(u) = \frac{KM}{2}.
\]
As a smooth approximation for \( f(x) = \| W(x) \|_1 \) we use
\[
f_\mu(x) = \max_{u \in Q_2} \{ u^T Wx - \mu d_2(u) \}
\]
which bounds \( f(x) \) as
\[
f_\mu(x) \leq f(x) \leq f_\mu(x) + \mu \Delta_2,
\]
i.e., for \( \epsilon = \frac{\mu}{2\Delta_2} \) we have an \( \frac{1}{2} \)-approximation of \( f(x) \). Furthermore,
it can be shown that \( f_\mu(x) \) has Lipschitz continuous derivatives with
constant \( L_\mu = \frac{\mu}{2} \| W \|_2^2 \).
Nesterov’s first-order method for minimizing the strongly convex
function \( f_\mu(x) \) with Lipschitz continuous derivatives is given below.

**Nesterov’s method**

Given a feasible \( x^{[0]} \) and a smoothing parameter \( \mu = \epsilon/(2\Delta_2) \).

For \( k \geq 0 \)

1. Evaluate \( g^{[k]} = \nabla f_\mu(x^{[k]}) \).
2. Find
\[
u^{[k]} = \arg \min_{x \in Q_1'} \{ (x - x^{[k]})^T g^{[k]} + \frac{L_\mu}{2} \|x - x^{[k]}\|_2^2 \}.
\]
3. Find
\[
u^{[k]} = \arg \min_{x \in Q_1'} \{ L_\mu d_1(x) + \sum_{i=0}^{k} \frac{i+1}{2} \|x - x^{[i]}\|_2^2 \}.
\]
4. Update \( x^{[k+1]} = \frac{2}{k+3} x^{[k]} + \frac{k+1}{k+3} x^{[k]} \).
As a feasible starting point we can use \( x[0] = (y, 0, \ldots, 0) \). We terminate the algorithm when the duality gap

\[
\|Wx\|_1 + \delta\| (W^Tu)_1 \|_2 - y^T (W^Tu)_1 + \gamma \| (W^Tu)_{2\ell} \|_2 \leq \epsilon.
\]

Nesterov shows in [4] that it takes at most

\[ N_{it} = 4\|W\|_2 \sqrt{\Delta_1 \Delta_2} \cdot \frac{1}{\epsilon} \]

iterations for the algorithm to produce an \( \epsilon \)-optimal solution. Using the fact that the transforms \( D_i \) are orthogonal \( (D_i^{-1} = D_i^T) \), the norm \( \|W\|_2^2 \) is easily found by considering the possible solutions \( \lambda \) to the characteristic equation,

\[ WW^T x = \lambda x \]

which shows that \( WW^T \) has only two distinct eigenvalues,

\[ \lambda = \frac{(K+1)^2 \pm \sqrt{(K+1)^2 - 4}}{4} \]

i.e., \( \|W\|_2^2 = \frac{(K+1)^2 + \sqrt{(K+1)^2 - 4}}{4} \). Substituting the values of \( \Delta_1 \), \( \Delta_2 \) and \( \|W\|_2^2 \), we get the iteration complexity bound for \( \ell_1 \) compression,

\[ N_{it} = \left( \frac{(K+1)^2 + \sqrt{(K+1)^2 - 4}}{4} \right)^{1/2} \left( N(\gamma^2 + \delta^2) \right)^{1/2} \cdot \frac{1}{\epsilon}. \]

The three different sub-problems in Nesterov’s method are easily solved. In step 1) we evaluate \( \nabla f_i(x) = W^T u \) where we find \( u \) as the solution to

\[
\min_{\|u\|_1 \leq 1} \langle \mu/2 \rangle u \|_2^2 - u^T W x
\]

which has a closed-form solution for the \( i \)th component (not to be confused with the \( i \)th sub-vector),

\[ u_i = \min\{1, \frac{\langle (Wx)_i \rangle}{\mu} \} \text{sign}((Wx)_i). \]

By simple changes of variables the solution to steps 2) and 3) can both be found as the solution to simple quadratically constrained problems

\[
\begin{align*}
\text{minimize} & \quad w^T w - 2\epsilon^T w \\
\text{subject to} & \quad w^T w \leq \delta^2
\end{align*}
\]

with solution \( w = \min\{1, \delta/\|\epsilon\|_2\} \epsilon \).

A remaining issue is to find a bound on \( \gamma \). For \( K = 1 \) we obviously have that \( \|z_1\|_2 \leq \|y\|_2 \). For \( K = 2 \) the same bound can be used for \( \|z_2\|_2 \), where the bound can only be reached if \( z_1 = 0 \) at optimality. In a similar fashion, we can argue that \( \|z_2, \ldots, z_K\|_2 \leq \|y\|_2 \).

3. NUMERICAL EXPERIMENTS

We implemented the algorithm in the Python programming language, and for orthogonal transforms we used the discrete cosine transform in FFTW [11] and a 2D wavelet library [12]. For reproducibility we have made the developed code for \( \ell_1 \) compression available for download [13].

3.1. Quality of the approximation

An issue often reported in large-scale implementations of sparse \( \ell_1 \) approximations is that a high accuracy is required to achieve a sparse solution. In the first experiment we compare the quality of the solution as a function of \( \epsilon \) with that of MOSEK [14], a state-of-the-art commercial convex optimization solver based on a primal-dual interior point algorithm. Thus, by using a primal-dual interior point method we get a very accurate reference solution, but we are only able to solve problems with small images.

For different values of \( \epsilon \) we randomly generate 100 8bit images of dimensions 32 \( \times \) 32 with values between 0 and 255 and we then compute the solution to (3) using Nesterov’s method with a value \( \delta \) corresponding to PSNR=40dB, and we compare the obtained solution with that of MOSEK.

In the examples we use an overcomplete dictionaries formed by two orthogonal transforms; a discrete Haar wavelet transform with two decomposition levels, and a discrete Symlet4 wavelet transform also with two decomposition levels.

<table>
<thead>
<tr>
<th>( \epsilon \cdot KM )</th>
<th>10(^{-3} )</th>
<th>5 ( \cdot ) 10(^{-4} )</th>
<th>10(^{-4} )</th>
<th>5 ( \cdot ) 10(^{-5} )</th>
<th>10(^{-5} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>NEST ( 1027.7 )</td>
<td>995.7</td>
<td>963.2</td>
<td>961.7</td>
<td>959.5</td>
<td></td>
</tr>
<tr>
<td>MOSEK ( 956.3 )</td>
<td>( 1102.0 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1. Average number of non-zero transform coefficients out of a total of 2 \( \cdot \) 32\(^2 \) = 2048 coefficients.

Table 1 shows the average number of non-zero transform coefficients for Nesterov’s method and MOSEK, and for comparison we also show the average number of non-zero transform coefficients needed by the standard matching pursuit algorithm [15] to achieve PSNR=40dB. To count the number of transform coefficients produced by MOSEK and Nesterov’s method we need to truncate the result in order to remove very small coefficients; this truncation resulted in degradation of the reconstruction of approximately 0.3dB. From the table we see that for moderately high accuracies the sparsity of the solution computed with Nesterov’s method is comparable to the MOSEK solution, but as \( \epsilon \) increases the number of nonzero coefficients also increases.

In the table we specify a relative accuracy normalized by the number of variables; a similar normalization of accuracy by the number of variables is used in most practical optimization solvers including MOSEK.

3.2. Large-scale studies

In the next example we illustrate the performance of Nesterov’s method for two standard 512 \( \times \) 512 grey-scale test-images, namely LENA and BOAT. In the example we show the effect of increasing the number of bases.

Table 2 shows the number of iterations used by Nesterov’s method and the resulting \( \|z\|_1 \) objective as well as the number of non-zero coefficients of the solution after the truncation of non-zero coefficients (called \( \|z\|_0 \)). For comparison we also show the number of non-zero coefficients of the solution produced by the standard matching pursuit using two bases. The bases used in the experiment were: \( D_1 \): DCT, \( D_2 \): Symlet8 pyramid transform with 6 levels, \( D_3 \): Symlet8 standard transform with 4 levels, and \( D_4 \): Symlet16 standard transform with 3 levels, \( i.e. \), MP2 and NE2 used \( D_1 \) and \( D_2 \), NE3 used \( D_1 \), \( D_2 \), and \( D_3 \), etc.
<table>
<thead>
<tr>
<th></th>
<th>It</th>
<th>|z|_1</th>
<th>|z|_0</th>
</tr>
</thead>
<tbody>
<tr>
<td>MP2</td>
<td>38949</td>
<td>6247.1</td>
<td>38949</td>
</tr>
<tr>
<td>NEST2</td>
<td>3882</td>
<td>5229.9</td>
<td>56205</td>
</tr>
<tr>
<td>NEST3</td>
<td>4043</td>
<td>4848.4</td>
<td>61379</td>
</tr>
<tr>
<td>NEST4</td>
<td>4244</td>
<td>4402.3</td>
<td>55240</td>
</tr>
</tbody>
</table>

Table 2. Sparsity of solutions \( z \) for large images for \( \text{PSNR}=40 \text{dB} \) with \( \epsilon = 10^{-3} K M \). Top rows show results for standard LENA test image, and bottom rows show results for standard BOAT test image. NEST\( K \): Nesterov’s method with \( K \) bases, MP\( K \): matching pursuit with \( K \) bases. “It” is the number of iterations used.

From the example we observe that the \( \ell_1 \) measure decreases as we add redundancy to the dictionary by increasing the number of bases. Similarly, the compression rate defined as the ratio of the number non-zero transform coefficients and total number of transform coefficients generally increases as we use more bases.

From the table we observe the somewhat discouraging phenomenon that even though the \( \ell_1 \) objective decreases as we use more bases the cardinality of the solution does not decrease correspondingly. Also, in these examples matching pursuit is actually superior to \( \ell_1 \) compression in terms of sparsity, but the number of iterations is significantly higher for the matching pursuit method; the standard matching pursuit method requires \( K+1 \) transforms per iteration, whereas Nesterov’s method requires an evaluation of \( W(x) \) and \( W_{\text{opt}}(x) \) amounting to a total of \( 3K \) transforms, so the overall complexity of Nesterov’s method is much lower than matching pursuit (due to a much lower number of iterations).

4. DISCUSSIONS

In this paper, we developed an efficient first-order scheme for \( \ell_1 \) compression of images using incomplete dictionaries. The overcomplete dictionaries are formed by combining orthogonal transforms (i.e., they work as operators) and thus have modest memory requirements.

Our approach of combining transforms is by no means unique. The only requirement is that the augmented transform is invertible, so that we can perform the variable change in (2). For example, for two orthogonal transforms \( D_1 \) and \( D_2 \) a straightforward approach is to define the transform pair as

\[
W^{-1} = \frac{1}{\sqrt{2}} \begin{bmatrix} D_1^{-1} & D_2^{-1} \\ -D_1^{-1} & D_2^{-1} \end{bmatrix}, \quad W = \frac{1}{\sqrt{2}} \begin{bmatrix} D_1 & -D_1 \\ D_2 & D_2 \end{bmatrix},
\]

resulting in a lower iteration complexity bound since \( \|W\|_2 = 1 \). Similarly we can augment the transform for any \( K = 2^l \). The concept presented in this paper works for any \( K \geq 1 \), however.

We have demonstrated that \( \ell_1 \) compression for large-scale image coding is in fact tractable; this has so far not been the case with traditional approaches based either on large-scale subgradient schemes, or interior-point with iterative large-scale linear equation solvers.

One experimental observation we have made is that \( \ell_1 \) compression using multiple bases often distributes the transform coefficients evenly across several bases, which is obviously suboptimal seen from a coding perspective. One approach to alleviate this problem is to solve several \( \ell_1 \) compression problems with different weights on the transform coefficients in the cost function, as proposed by Candès et al. in [16]. Preliminary experiments using this scheme show promising results, e.g., using the reweighting for the compressing LENA with two bases (cf., Table 2) reduces the number of transform coefficients from 56205 to 32959, which is better than matching pursuit in terms of coding gain. The price paid for this substantial improvement is that we must solve several \( \ell_1 \) compression problems; in this particular case we solved 6 problems. It is therefore expected that (improved) \( \ell_1 \) compression algorithms for images will be more efficient than matching pursuit both in terms of complexity and coding gain.

Studies [17] show that the resulting bit-rate is largely proportional to the sparsity of the solution, but more detailed coding considerations would be an important topic for future studies; coding of sparse transforms is largely undeveloped [18], whereas state-of-the-art coding for wavelet transforms (e.g., [19]) is highly sophisticated.

5. REFERENCES