Putting Compression into CS

Lawrence Carin
Department of Electrical & Computer Engineering
Duke University
Durham, NC

Joint work with Lihan He
Notation

- Assume we are interested in measuring a signal $\mathbf{x} \in \mathbb{R}^N$
- It is assumed that $\mathbf{x} = \Psi \theta$ where $\Psi$ is an $N \times N$ matrix with columns corresponding to an orthonormal basis, and $\theta$ is a compressible $N$-dimensional vector
- Rather than directly measuring $\mathbf{x}$, we measure $\mathbf{v} = \Sigma \mathbf{x}$ where $\Sigma$ is an $M \times N$ dimensional real matrix, ideally with $M \ll N$
- The projection measurements may be expressed as $\mathbf{v} = \Sigma \Psi \theta = \Phi \theta$, where $\Phi$ is an $M \times N$ dimensional matrix
Problems With Traditional CS Inversion

- In practice the signal is not exactly sparse, and therefore there will be loss in the reconstruction, and one may be interested in the estimated “error bars”
- Rather than desiring simply a “point” estimate for $\theta$, desire a posterior distribution that quantifies uncertainty
- Since degree of compressibility generally not known \textit{a priori}, this measure of confidence in inversion may also define when enough CS measurements $M$ have been taken
- $\ell_1$-based inversion does not exploit information we generally know about $\theta$, and by accounting for such structure one may anticipate that fewer CS measurements may be required to achieve a particular accuracy
Bayesian View of $\ell_1$-Based CS Inversion

- A standard linear regression problem for the unknown coefficients $\theta$ may be expressed as

\[ \mathbf{v} = \Phi \theta + \epsilon \]

with components of $\epsilon$ assumed drawn i.i.d. from $\mathcal{N}(0, \alpha_0^{-1})$

- The likelihood function for the coefficients $\theta$ is

\[ p(\mathbf{v}|\theta) = \mathcal{L}(\theta; \mathbf{v}) = \mathcal{N}(\mathbf{v}|\Phi \theta, \alpha_0^{-1} \mathbf{I}) \]

- Laplace prior widely employed to impose belief that $\theta$ is sparse

\[ p(\theta) = \frac{\gamma}{2} \exp[-\gamma \sum_{i=1}^{N} |\theta_i|] \]
MAP Estimate for Transform Coefficients

- The posterior density function for the coefficients $\theta$ satisfies

$$p(\theta|v) \propto p(v|\theta)p(\theta)$$

- If we let the approximate solution $\hat{\theta}$ correspond to the coefficients that maximize the log-posterior, under above assumptions

$$\hat{\theta} = \arg \min_{\theta} \left[ ||v - \Phi \theta||_2^2 + \beta ||\theta||_{\ell_1} \right]$$

with $\beta = \gamma/\alpha_0$.

- The standard $\ell_1$-regularized CS inversion may be viewed as a MAP approximation to fully Bayesian analysis.
Why Use a MAP Approximation?

- *If* one *a priori* knows the sparseness $S$, and the signal is truly sparse, $\ell_1$-based inversion is exact
- *Direct* evaluation of the posterior $p(\theta|v)$ using a Laplace prior is not tractable

But ....

- In practice the signal is not exactly sparse, and there will always be loss in reconstruction
- Would like to quantify “error bars”
- Modern hierarchical Bayesian inference techniques *are* computationally efficient
Hierarchical Representation of Laplace Prior

Consider the following hierarchical model for $\theta$:

\[
\theta \sim \mathcal{N}(0, \text{diag}^{-1}(\alpha))
\]

\[
\alpha \sim \prod_{i=1}^{N} \text{InvGamma}(\alpha_i; 1, \gamma_j/2)
\]

\[
\gamma \sim \prod_{i=1}^{N} \text{Gamma}(\gamma_i; a_0, b_0)
\]

The marginal density function for $\theta$ may be shown to satisfy

\[
p(\theta|\gamma) = \prod_{i=1}^{N} \text{Laplace}(\theta_i; \sqrt{\gamma_i})
\]

Efficient inference: conjugate-exponential structure
Sparse Linear Regression

- Assume that the $M$ CS measurements $\mathbf{v}$ may be represented as $\mathbf{v} = \Phi \theta$, where $\theta$ are *compressible* transform coefficients.
- Let $\theta_S$ represent $\theta$ with the $N - S$ smallest coefficients set to zero.
- Let $\theta_e$ represent $\theta$ with the $S$ largest coefficients set to zero.

$$\mathbf{v} = \Phi \theta = \Phi \theta_S + \Phi \theta_e = \Phi \theta_S + \epsilon$$

- May also add measurement noise in addition to the error $\epsilon$. 

Lawrence Carin
Putting Compression into CS
Bayesian Lasso

\[ \mathbf{v} = \Phi \mathbf{\theta} = \Phi \mathbf{\theta}_S + \Phi \mathbf{\theta}_e = \Phi \mathbf{\theta}_S + \epsilon \]

Impose Laplace prior via hierarchical structure:

\[ \mathbf{v} \sim \mathcal{N}(\Phi \mathbf{\theta}_S, \alpha_0^{-1} \mathbf{I}) \]
\[ \mathbf{\theta}_S \sim \mathcal{N}(0, \text{diag}^{-1}(\alpha)) \]
\[ \alpha \sim \prod_{i=1}^{N} \text{InvGamma}(\alpha_i; 1, \gamma_i/2) \]
\[ \gamma \sim \prod_{i=1}^{N} \text{Gamma}(\gamma_i; a_0, b_0) \]
Fast Inference

- Hierarchical model may look complicated, but it is actually simple
- Each consecutive term in the hierarchy in the conjugate-exponential family
- Fast inference via variational Bayesian analysis
- Yields a full posterior density function on all model parameters
Structure in Wavelet Transform

- Baraniuk et al. introduced concept of “high” and “low” states as well as persistence across scales of the strength of wavelet-transform coefficients
- Represented in terms of a hidden Markov tree
DCT-based JPEG compression is most widely used standard in image-compression technology

DCT performed separately on contiguous 8 × 8 blocks in image

A tree structure may be constituted almost identical to that used for wavelets (Z. Xiong, O.G. Guleryuz and M.T. Orchard, “A DCT-Based Embedded Image Coder”, SPL, 1996)

The exact same tree structure used for wavelets may be used for JPEG-like DCT decomposition
### JPEG Tree Structure

![JPEG Tree Structure Diagram]

<table>
<thead>
<tr>
<th></th>
<th></th>
<th>4</th>
<th>5</th>
<th></th>
<th></th>
<th>16</th>
<th>17</th>
<th>20</th>
<th>21</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>16</td>
<td>17</td>
<td>20</td>
<td>21</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>12</td>
<td>13</td>
<td>18</td>
<td>19</td>
<td>22</td>
<td>23</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>11</td>
<td>14</td>
<td>15</td>
<td>24</td>
<td>25</td>
<td>28</td>
<td>29</td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>33</td>
<td>36</td>
<td>37</td>
<td>48</td>
<td>49</td>
<td>52</td>
<td>53</td>
<td></td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>35</td>
<td>38</td>
<td>39</td>
<td>50</td>
<td>51</td>
<td>54</td>
<td>55</td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>41</td>
<td>44</td>
<td>45</td>
<td>56</td>
<td>57</td>
<td>60</td>
<td>61</td>
<td></td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>43</td>
<td>46</td>
<td>47</td>
<td>58</td>
<td>59</td>
<td>62</td>
<td>63</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Fig. 1.* An $8 \times 8$ DCT block can be treated as a depth-3 tree of coefficients.
Wavelet/JPEG-DCT Zero Trees

- Henceforth refer to $\theta_S$ as $\theta$ for notational simplicity, recognizing that it is $S$-sparse.
- In a wavelet/JPEG-DCT decomposition, it is well known that if a coefficient is zero, it is likely that its four “children” coefficients will also be zero.
- This manifests “zero trees”.
- Implies that there is important structure in the form of the transform coefficients, which we would like to exploit.
- Would like to impose this prior information within our prior $p(\theta)$.
Prior for Zero Trees

The $i$th transform coefficient, at scale $s$, is assumed drawn from a “spike-slab” prior

$$\theta_i \sim (1 - \pi_i)\delta_0 + \pi_i \mathcal{N}(0, \alpha_s^{-1})$$

The two terms in the above mixture model correspond to zero and non-zero states of a given wavelet coefficient.

Each scale $s$ has its own “precision” $\alpha_s$.

Wish to impose Markovian relationships on the $\pi_i$ between parent and children coefficients.
Hierarchical Representation for Wavelet Coefficients

\[ \mathbf{v} | \theta, \alpha_n \sim \mathcal{N}(\Phi \theta, \alpha_n^{-1} \mathbf{I}) \]
\[ \theta_{s,i} \sim (1 - \pi_{s,i}) \delta_0 + \pi_{s,i} \mathcal{N}(0, \alpha_s^{-1}) \]
\[ \pi_{s,i} = \begin{cases} 
\pi_r, & \text{if } s = 1 \\
\pi_0, & \text{if } 2 \leq s \leq L, \theta_{pa(s,i)} = 0 \\
\pi_1, & \text{if } 2 \leq s \leq L, \theta_{pa(s,i)} \neq 0 
\end{cases} \]
\[ \alpha_n \sim \text{Gamma}(a_0, b_0) \]
\[ \alpha_s \sim \text{Gamma}(c_0, d_0), \quad s = 1, \ldots, L \]
\[ \pi_r \sim \text{Beta}(e_{r0}, f_{r0}) \]
\[ \pi_0 \sim \text{Beta}(e_{0s0}, f_{0s0}), \quad s = 2, \ldots, L \]
\[ \pi_1 \sim \text{Beta}(e_{0s1}, f_{0s1}), \quad s = 2, \ldots, L \]
In addition to the hierarchical two-state Markovian model for the wavelet/JPEG-DCT coefficients, we must also build a prior for the scaling/DC coefficients.

- The scaling/DC coefficients are small in quantity relative to the rest of the coefficients.
- Scaling/DC coefficients are typically not sparse.
- The scaling coefficients are assumed drawn i.i.d. from a Gaussian with precision $\alpha_{sc}$.
- The $\alpha_{sc}$ is drawn from a Gamma distribution with set hyper-parameters.
Fast Bayesian Inference

- Important to note that consecutive levels in the wavelet- and DCT-based models are in the conjugate-exponential family.
- In this case, local posterior inference is analytic, characterized by updating hyper-parameters.
- Use Markov chain Monte Carlo (MCMC) sampling, with mean inverted signal typically converging very quickly (around 100 iterations).
- Yield a posterior distribution on the underlying transform coefficients, and therefore we can quantify accuracy.
Statistical CS Inversion
Exploiting Structure in Transform Coefficients
Example CS Inversion Results
Conclusions

(a) Original image
(b) Format 1: cutting
(c) Format 2: resizing
(d) Format 3: hybrid CS

Lawrence Carin
Putting Compression into CS
CPU Time: Wavelet-Based CS

[Graph showing CPU time vs. number of measurements for various algorithms]

- TSW-CR
- BP
- VB BCS
- fast-BCS
- OMP
- StOMP-CFAR
- LARS/Lasso
- TV
Example Inferred Wavelet Coefficients

Truth
TSW–CS
BP
VB BCS
fast–BCS
OMP
StOMP–CFAR
LARS/Lasso
TV

Wavelet Coefficient Index

0 2000 4000 6000 8000 10000 12000 14000 16000
Statistical CS Inversion
Exploiting Structure in Transform Coefficients
Example CS Inversion Results
Conclusions

Lawrence Carin
Putting Compression into CS
Five Image Classes
Statistical CS Inversion
Exploiting Structure in Transform Coefficients
Example CS Inversion Results
Conclusions

Lawrence Carin
Putting Compression into CS
### Average Performance, $N = 2000$

<table>
<thead>
<tr>
<th>Class</th>
<th>Algorithm</th>
<th>TSW-CS</th>
<th>VB BCS</th>
<th>fast-BCS</th>
<th>OMP</th>
<th>StOMP-CFAR</th>
<th>LARS/Lasso</th>
<th>TV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flowers</td>
<td>MEAN</td>
<td>0.2296</td>
<td>0.3075</td>
<td>0.3921</td>
<td>0.4082</td>
<td>0.3465</td>
<td>0.2870</td>
<td>0.2908</td>
</tr>
<tr>
<td></td>
<td>STD</td>
<td>0.0736</td>
<td>0.0914</td>
<td>0.1251</td>
<td>0.1313</td>
<td>0.1038</td>
<td>0.0868</td>
<td>0.0840</td>
</tr>
<tr>
<td>Cows</td>
<td>MEAN</td>
<td>0.1631</td>
<td>0.2162</td>
<td>0.2663</td>
<td>0.2776</td>
<td>0.2391</td>
<td>0.1988</td>
<td>0.2053</td>
</tr>
<tr>
<td></td>
<td>STD</td>
<td>0.0556</td>
<td>0.0683</td>
<td>0.0920</td>
<td>0.0963</td>
<td>0.0758</td>
<td>0.0649</td>
<td>0.0609</td>
</tr>
<tr>
<td>Buildings</td>
<td>MEAN</td>
<td>0.2178</td>
<td>0.2814</td>
<td>0.3559</td>
<td>0.3723</td>
<td>0.3173</td>
<td>0.2618</td>
<td>0.2655</td>
</tr>
<tr>
<td></td>
<td>STD</td>
<td>0.0606</td>
<td>0.0778</td>
<td>0.1031</td>
<td>0.1063</td>
<td>0.0878</td>
<td>0.0735</td>
<td>0.0722</td>
</tr>
<tr>
<td>Urban</td>
<td>MEAN</td>
<td>0.2003</td>
<td>0.2556</td>
<td>0.3254</td>
<td>0.3389</td>
<td>0.2885</td>
<td>0.2377</td>
<td>0.2463</td>
</tr>
<tr>
<td></td>
<td>STD</td>
<td>0.0252</td>
<td>0.0341</td>
<td>0.0475</td>
<td>0.0492</td>
<td>0.0376</td>
<td>0.0321</td>
<td>0.0309</td>
</tr>
<tr>
<td>Office</td>
<td>MEAN</td>
<td>0.2360</td>
<td>0.3164</td>
<td>0.3969</td>
<td>0.4187</td>
<td>0.3553</td>
<td>0.2920</td>
<td>0.2958</td>
</tr>
<tr>
<td></td>
<td>STD</td>
<td>0.0448</td>
<td>0.0642</td>
<td>0.0780</td>
<td>0.0842</td>
<td>0.0708</td>
<td>0.0592</td>
<td>0.0579</td>
</tr>
</tbody>
</table>
Statistical CS Inversion
Exploiting Structure in Transform Coefficients
Example CS Inversion Results
Conclusions

DCT, BlockSize=8

Image size: 128 x128

Wavelet, DecompositionLevel=3

DCT & Wavelet
Summary

- Significant structure is exploited in JPEG and JPEG2000
- This same structure may be exploited in CS inversion
- Have developed a Bayesian CS inversion algorithm that infers underlying signal and structure
- Framework yields a full posterior density function on underlying signal (error bars)
- State-of-the-art CS-inversion quality based on a large set of natural images