Multistage Compressed-Sensing Reconstruction of Multiview Images

Maria Trocan *, Thomas Maugey +, Eric W. Tramel #, James E. Fowler *, Béatrice Pesquet-Popescu +

* Institut Supérieur d’Électronique de Paris, France
+ Télécom ParisTech, France
# Mississippi State University, USA

Abstract—Compressed sensing is applied to multiview image sets and the high degree of correlation between views is exploited to enhance recovery performance over straightforward independent view recovery. This gain in performance is obtained by recovering the difference between a set of acquired measurements and the projection of a prediction of the signal they represent. The recovered difference is then added back to the prediction, and the prediction and recovery procedure is repeated in an iterated fashion for each of the views in the multiview image set. The recovered multiview image set is then used as an initialization to repeat the entire process again to form a multistage refinement. Experimental results reveal substantial performance gains from the multistage reconstruction.

I. INTRODUCTION

Many modern applications, such as 3D reconstruction, creation of virtual environments, surveillance systems, and more, require several cameras to record a scene concurrently from different perspectives. In these cases, there is a large amount of correlation between the images representing each viewpoint. Compression, restoration, or other data-processing techniques can make use of this information redundancy to enhance their performance or robustness. Disparity compensation (DC) is commonly used to exploit this redundancy by making a prediction of a current view from other views in the image set. In the case of compression, a DC prediction can be used to calculate a residual between the prediction and the original image. The residual image obtained in this manner is often much more amenable to compression than the original image.

Because multiview data acquisition requires many sensors operating concurrently, the volume of data to be either stored locally or transmitted remotely can be prohibitive in some applications. It is anticipated that such applications can benefit from compressed sensing (CS), a new paradigm which allows signals to be sampled at sub-Nyquist rates and, under certain conditions of sparsity and incoherence [1], be recovered with negligible loss. One common method of CS-based signal acquisition uses a linear projection onto a random basis, a scenario that has been shown to be physically realizable with a single-pixel camera [2]. Recovery of signals sampled in this manner can be achieved via any one of the many proposed CS reconstruction schemes (e.g., [3]).

In this paper, we propose a joint CS reconstruction of multiview image sets by utilizing DC to form predictions which serve as a form of side information to the image reconstruction algorithm. We use the efficient block-based method proposed in [4] as our image-recovery procedure. Experimental results indicate that the proposed method shows promising performance and demonstrates high-quality reconstruction even at very low subsampling rates. We note that a preliminary system we described in [5, 6] used an approach similar to that considered here; however, the system of [5, 6] employed a simpler, two-stage reconstruction. In contrast, the system we propose here adds one or more refinement stages to produce a multistage reconstruction exhibiting substantial improvement in performance over the system of [5, 6].

II. BACKGROUND

One of the main advantages of the CS paradigm is the very low computational burden placed on the encoding process, which requires only the projection of the signal x, of dimensionality N, onto some measurement basis, \( \Phi_{N \times M} \), where \( M \ll N \). The result of this computation is the M-dimensional vector of measurements, \( y = \Phi x \). \( \Phi \) is often chosen to be a random matrix because it satisfies the incoherency requirements of CS reconstruction for any structured signal transform \( \Psi \) with a high probability. In this way, the encoder can also be said to be structure agnostic. We assume \( \Phi \) is also chosen to be orthonormal (\( \Phi^T \Phi = I \)). We define the subsampling rate, or subrate, of the CS scheme as \( M/N \).

This light encoding procedure offloads most the computation of CS onto the decoder. Because the inverse of the projection \( \Phi \) is ill-posed, we cannot directly solve the inverse problem to find the original signal from the given measurements. Instead, the CS paradigm tells us that the correct solution for \( x \) is the sparsest signal which lies in the set of signals that match the measurements [1]; i.e.,

\[
\hat{x} = \arg \min_x \| \Psi x \|_{\ell_0} \quad \text{s.t.} \quad y = \Phi x, \tag{1}
\]

where sparsity is measured in the domain of transform \( \Psi \). However, this \( \ell_0 \)-constrained optimization problem is computationally infeasible due to its combinatorial and non-differentiable nature. Thus, a \( \ell_1 \) convex relaxation is often applied, sacrificing accuracy but permitting the recovery to be implemented directly via linear-programming techniques (e.g., [7–9]). Further relaxations of the optimization have also...
been attempted, such as the mixed \( \ell_1-\ell_2 \) method proposed in [10]. However, all of these schemes still suffer from very long reconstructions times for \( N \) of any practical or interesting size.

Iterative thresholding algorithms have also been proposed as another class of solutions for CS recovery. The most common of these is the iterated hard thresholding (IHT) algorithm (e.g., [11–13]). IHT replaces the constrained optimization formulation with an unconstrained optimization problem via a Lagrangian multiplier and further relaxes the problem by loosening the equality constraint to an \( \ell_2 \)-distance penalty,

\[
\hat{x} = \arg \min_{x} \| \Psi x \|_{\ell_1} + \lambda \| y - \Phi x \|_{\ell_2}.
\]  

(2)

Algorithms of this class recover \( \hat{x} \) by successive projection and thresholding operations. Given some initial approximation \( \hat{x}^{(0)} \) to the transform coefficients \( \hat{x} = \Psi x \), the solution is calculated in the following manner:

\[
\hat{x}^{(i)} = \hat{x}^{(i-1)} + \frac{1}{\gamma} \Psi \Phi^T \left( y - \Phi \Psi^{-1} \hat{x}^{(i-1)} \right), \quad (3)
\]

\[
\hat{x}^{(i+1)} = \begin{cases} 
\hat{x}^{(i)}, & |\hat{x}^{(i)}| \geq \tau^{(i)}, \\
0, & \text{else},
\end{cases} \quad (4)
\]

where \( \gamma \) is a scaling factor, and \( \tau^{(i)} \) is the threshold used at the \( i \)th iteration. Further observation of this process shows us that this procedure is actually a specific instance of a projected Landweber (PL) algorithm [14]. We note that convergence of IHT has been shown in [11].

IHT recovery improves reconstruction speed by at least an order of magnitude and maintains a high degree of accuracy. Reconstruction time can be further reduced by implementing a block-based measurement and recovery procedure, as proposed in [3]. In this technique, \( \Phi \) is applied on a block-by-block basis, while the reconstruction step incorporates a smoothing operation (such as Wiener filtering) into the IHT. By employing blocking, the results in [3] show a reduction of computation time by four orders of magnitude for comparable accuracy versus linear-programming approaches. In [4], this method is referred to as block CS and smoothed PL (BCS-SPL) and is extended via the use of directional transforms. The algorithm in [4] is given as

\[
\text{function } x^{(i+1)} = \text{SPL}(x^{(i)}, y, \Phi_{\text{block}}, \Psi, \lambda) \\
\hat{x}^{(i)} = \text{Wiener}(x^{(i)}) \\
\text{for each block } j \\
\hat{x}_j^{(i)} = \hat{x}^{(i)} + \Phi_{\text{block}}^T (y - \Phi_{\text{block}} \hat{x}^{(i)}) \\
\hat{x}^{(i)} = \Psi \hat{x}_j^{(i)} \\
\hat{x}^{(i)} = \text{Threshold}(\hat{x}^{(i)}, \lambda) \\
\hat{x}^{(i)} = \Psi^{-1} \hat{x}^{(i)} \\
\text{for each block } j \\
x^{(i+1)} = x^{(i)} + \Phi_{\text{block}}^T (y - \Phi_{\text{block}} \hat{x}^{(i)})
\]

Here, \( \Psi^{(i)} = \Phi^T y \). The method uses hard thresholding with a fixed convergence factor \( \lambda \) for all iterations [13], and can be calculated as a function of the number of coefficients used in \( \Psi \) [15].

III. DISPARITY-COMPENSATED CS RECONSTRUCTION

We propose an iterative disparity-compensated algorithm for the reconstruction of multiview images using BCS-SPL. Because multiview images are strongly correlated, we can exploit this redundancy and consider only the DC residual for CS reconstruction. The given method assumes the same context as [4]. Each image in the multiview set, \( x_d \), is acquired using a measurement matrix, \( \Phi_d \), and the decoder is given only the set of observations \( y_d = \Phi_d x_d \) along with each \( \Phi_d \) used. The decoder makes a blind decision on the sparse basis, \( \Psi \), to use.

The algorithm is partitioned into three stages, as can be seen in the block diagram in Fig. 1. In the first, or initial, stage, each image in the multiview set is reconstructed individually from the received set of measurements using BCS-SPL. In the second stage (the “basic” stage), for each image \( x_d \), a prediction, \( x_p \), is created by bidirectionally interpolating the BCS-SPL reconstructions of the closest views, \( x_p = \text{ImageInterpolation}(\hat{x}_{d-1}, \hat{x}_{d+1}) \). Alternatively, the direct reconstruction of the view as obtained from BCS-SPL could be used as the initial prediction. However, we have found that at low subrates, the quality of the final reconstruction is much better when using an interpolation as the initial prediction. Next, we compute the residual \( r \) between the measurements and the projection of \( x_p \) by \( \Phi_d \). This residual in the measurement domain is then reconstructed using BCS-SPL and added back to the prediction to generate a reconstruction, \( \hat{x}_d \).

\( \hat{x}_d \) is further refined in the basic stage by calculating a set of disparity vectors, \( \text{DV}_{d-1} \) and \( \text{DV}_{d+1} \) (the right and left disparity vectors, respectively), via disparity estimation using the reconstructions of the neighboring right and left views from the first stage. These disparity vectors then drive the DC to form a prediction of the current view from these neighboring views. This prediction is substituted for \( x_p \), and the procedure is repeated. This procedure improves the quality of \( \hat{x}_d \) at each iteration by refining the disparity vectors at each step, producing better predictions and therefore producing smoother residuals which are more accurately recovered, leading finally to a more accurate \( \hat{x}_d \). For our implementation, we iterated three times.

Subsequently, one or more bootstrapping stages are performed. A bootstrapping-refinement stage of the algorithm is simply the repetition of the basic stage as described above with the results from the second stage substituted for the references used to drive the DC-CS reconstruction. The stages could conceivably be repeated until there is no significant difference between consecutive passes; however, in our experimental framework described in the next section, we consider only one refinement stage in order to minimize the overall computational complexity of the reconstruction.

We note that, for each view, a different random measurement matrix is used, and the information retained in the different projections has a high probability of being comple-
mentary. Knowing that each view is highly correlated, the performance gains from the refinement iterations is also due to complementary, highly correlated information along the disparity axis. Finally, we note that the system considered in [5,6] used only the initial and basic stages described here; experimental results below, however, demonstrate substantial performance improvement due to the bootstrapping/refinement stage in the multistage reconstruction.

IV. EXPERIMENTAL RESULTS

In our experiments, we used the dual-tree discrete wavelet transform (DDWT) transform [16] as the sparse representation basis, $\Psi$. The performance characteristics of the DDWT within the CS framework has been investigated in [4]. In our results, the direct reconstruction using BCS-SPL (i.e., the output of the initial stage of the algorithm) is referred to as “DDWT.” On the other hand, “DC-DDWT” refers to the results obtained after the basic stage of the proposed method, while “MS-DC-DDWT” refers to the results obtained using a third, bootstrapping stage. The DC prediction for each view is calculated using a block size of $16 \times 16$ pixels with a search window of $32 \times 32$ pixels. For BCS-SPL, a block size of $64 \times 64$ pixels is used as well as 6 levels of DDWT decomposition. All views are acquired with the same subrate, $M/N$.

Figs. 2–6 show the PSNR performance obtained for several $512 \times 512$ images from the Middlebury stereo-image database\(^1\) over the subrate used. All images are rectified and corrected for radial distortion. Because the measurement basis is random, all PSNR results are averaged over five independent trials.

As seen in the figures, the bootstrapping stage yields high-quality results, showing a gain of approximately 1.5 dB to 0.75 dB for high and low subrates, respectively, as compared to using only two stages. For highly textured images (e.g., “Monopoly,” “Aloe”), the last stage greatly improves the final reconstruction quality; for smooth images (e.g., “Plastic”), the gains are more nominal.

V. CONCLUSION

In this paper, we proposed a new method of CS reconstruction for highly correlated multiview image sets. By way of a multistage refinement procedure, we use the performance gains obtained via residual recovery to promote even better performance. The residual recovery was implemented by using DC to create image predictions which were projected into the measurement domain and subtracted from the measurements of the original image. These residuals were then added back to the predictions to get final reconstructions more accurate than direct reconstruction. Repeating the procedure was shown in our results to garner even better PSNR performance.

\(^1\)http://cat.middlebury.edu/stereo/data.html

REFERENCES

Fig. 1. The multistage DC-based reconstruction algorithm.
Fig. 2. Reconstruction quality for “Monopoly” as a function of subrate.

Fig. 3. Reconstruction quality for “Bowling” as a function of subrate.

Fig. 4. Reconstruction quality for “Aloe” as a function of subrate.

Fig. 5. Reconstruction quality for “Baby” as a function of subrate.

Fig. 6. Reconstruction quality for “Plastic” as a function of subrate.