Storage-Efficient Quasi-Newton Algorithms for Image Super-Resolution

Diego A. Sorrentino and Andreas Antoniou

Dept. of Elec. and Comp. Eng.
University of Victoria
Victoria, BC, Canada
Email: dsorrent@ieee.org
aantoniou@ieee.org

16th International Conference on Digital Signal Processing
July 5–7, 2009

University of Victoria

Overview

- Greyscale super-resolution algorithm (Farsiu, Robinson, Elad, and Milanfar, *IEEE Trans. Image Processing, 2004*)

University of Victoria

Overview

- Greyscale super-resolution algorithm (Farsiu, Robinson, Elad, and Milanfar, *IEEE Trans. Image Processing, 2004*)
- Color super-resolution algorithm (Farsiu, Elad, and Milanfar, *IEEE Trans. Image Processing, 2006*)

Overview

- Greyscale super-resolution algorithm (Farsiu, Robinson, Elad, and Milanfar, *IEEE Trans. Image Processing, 2004*)
- Color super-resolution algorithm (Farsiu, Elad, and Milanfar, *IEEE Trans. Image Processing, 2006*)
- Efficient implementation of grayscale SR algorithm using quasi-Newton Algorithms (Sorrentino and Antoniou, ISCAS 2008)

Overview

- Greyscale super-resolution algorithm (Farsiu, Robinson, Elad, and Milanfar, *IEEE Trans. Image Processing, 2004*)
- Color super-resolution algorithm (Farsiu, Elad, and Milanfar, *IEEE Trans. Image Processing, 2006*)
- Efficient implementation of grayscale SR algorithm using quasi-Newton Algorithms (Sorrentino and Antoniou, ISCAS 2008)
- Efficient implementation of color SR algorithm using quasi-Newton Algorithms (Sorrentino and Antoniou, CCECE 2009)

Overview

- Greyscale super-resolution algorithm (Farsiu, Robinson, Elad, and Milanfar, *IEEE Trans. Image Processing, 2004*)
- Color super-resolution algorithm (Farsiu, Elad, and Milanfar, *IEEE Trans. Image Processing, 2006*)
- Efficient implementation of grayscale SR algorithm using quasi-Newton Algorithms (Sorrentino and Antoniou, ISCAS 2008)
- Efficient implementation of color SR algorithm using quasi-Newton Algorithms (Sorrentino and Antoniou, CCECE 2009)
- Storage-efficient implementations using quasi-Newton Algorithms

University of Victoria

Overview

- Greyscale super-resolution algorithm (Farsiu, Robinson, Elad, and Milanfar, *IEEE Trans. Image Processing, 2004*)
- Color super-resolution algorithm (Farsiu, Elad, and Milanfar, *IEEE Trans. Image Processing, 2006*)
- Efficient implementation of grayscale SR algorithm using quasi-Newton Algorithms (Sorrentino and Antoniou, ISCAS 2008)
- Efficient implementation of color SR algorithm using quasi-Newton Algorithms (Sorrentino and Antoniou, CCECE 2009)
- Storage-efficient implementations using quasi-Newton Algorithms
- Simulation results and comparisons

Overview

- Greyscale super-resolution algorithm (Farsiu, Robinson, Elad, and Milanfar, *IEEE Trans. Image Processing, 2004*)
- Color super-resolution algorithm (Farsiu, Elad, and Milanfar, *IEEE Trans. Image Processing, 2006*)
- Efficient implementation of grayscale SR algorithm using quasi-Newton Algorithms (Sorrentino and Antoniou, ISCAS 2008)
- Efficient implementation of color SR algorithm using quasi-Newton Algorithms (Sorrentino and Antoniou, CCECE 2009)
- Storage-efficient implementations using quasi-Newton Algorithms
- Simulation results and comparisons
- Conclusions

University of Victoria

Motivation

- In our ISCAS 2008 and CCECE 2009 papers, we have shown that efficient quasi-Newton implementations of the Farsiu et al. grayscale and color super-resolution algorithms based on the Broyden, Fletcher, Goldfarb, and Shanno (BFGS) method lead to *improved speed and reconstruction quality*.

Motivation

- In our ISCAS 2008 and CCECE 2009 papers, we have shown that efficient quasi-Newton implementations of the Farsiu et al. grayscale and color super-resolution algorithms based on the Broyden, Fletcher, Goldfarb, and Shanno (BFGS) method lead to *improved speed and reconstruction quality*.
- Unfortunately, these implementations require a large amount of storage.

Motivation

- In our ISCAS 2008 and CCECE 2009 papers, we have shown that efficient quasi-Newton implementations of the Farsiu et al. grayscale and color super-resolution algorithms based on the Broyden, Fletcher, Goldfarb, and Shanno (BFGS) method lead to *improved speed and reconstruction quality*.
- Unfortunately, these implementations require a large amount of storage.
- In the present paper, we explore the implementations of the Farsiu et al. algorithms in terms of the so-called *memoryless* and *limited-memory* BFGS algorithms (*MBFGS* and *LBFGS*, respectively).

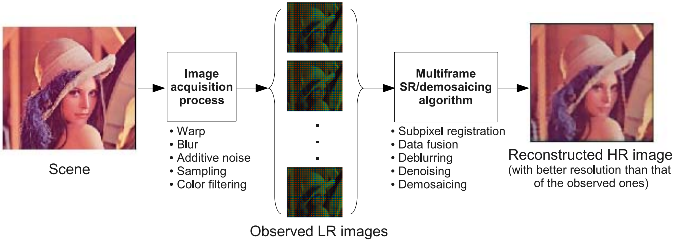
University of Victoria

Motivation

- In our ISCAS 2008 and CCECE 2009 papers, we have shown that efficient quasi-Newton implementations of the Farsiu et al. grayscale and color super-resolution algorithms based on the Broyden, Fletcher, Goldfarb, and Shanno (BFGS) method lead to *improved speed and reconstruction quality*.
- Unfortunately, these implementations require a large amount of storage.
- In the present paper, we explore the implementations of the Farsiu et al. algorithms in terms of the so-called *memoryless* and *limited-memory* BFGS algorithms (*MBFGS* and *LBFGS*, respectively).
- As will be demonstrated, these algorithms require *reduced and controllable storage* while retaining the *speed advantage* as well as the *improved reconstruction quality* associated with the BFGS algorithm.

University of Victoria

- Super-resolution can be used to fuse several blurry, undersampled, and noisy low-solution images of a scene, referred to as *frames*, into a high-quality high-resolution image.



Frame Acquisition Models

- After lexicographical reordering, grayscale and color low-resolution frame sets can be represented by

$$\mathbf{y_k} \quad = \quad \mathbf{DHF_k x} + \boldsymbol{\nu}_\mathbf{k}, \quad k = 1, \ldots, N$$

and

$$\mathbf{y_{i,k}} \quad = \quad \mathbf{D_i H F_k x_i} + \boldsymbol{\nu}_\mathbf{i,k}, \quad k = 1, \ldots, N \quad i = R, G, B$$

respectively, where

$\mathbf{F_k}$ is the warp matrix
$\mathbf{H}$ is the blur matrix
$\mathbf{D}$ is the downsampling matrix
$\mathbf{D}_i$ are the downsampling plus color filtering matrices
$\boldsymbol{\nu}$, $\boldsymbol{\nu}_i$ are additive noise vectors
$\mathbf{y}_k$, $\mathbf{y}_{i,k}$ are the kth LR Frame and ith color component of the kth LR frame

Iterative Reconstruction

- An HR image is reconstructed from the observed LR frames by means of an iterative optimization-based model inversion, as follows:

Iterative Reconstruction

- An HR image is reconstructed from the observed LR frames by means of an iterative optimization-based model inversion, as follows:
  - For grayscale SR:

$$\hat{\mathbf{x}}_{\mathbf{G}} = \underset{\mathbf{x}}{\text{ArgMin}}\,[f_G(\mathbf{x})] = \underset{\mathbf{x}}{\text{ArgMin}}\,[J_0(\mathbf{x}) + \lambda J_1(\mathbf{x})]$$

University of Victoria

Iterative Reconstruction

- An HR image is reconstructed from the observed LR frames by means of an iterative optimization-based model inversion, as follows:
  - For grayscale SR:

$$\hat{\mathbf{x}}_{\mathbf{G}} = \underset{\mathbf{x}}{\text{ArgMin}}\,[f_G(\mathbf{x})] = \underset{\mathbf{x}}{\text{ArgMin}}\,[J_0(\mathbf{x}) + \lambda J_1(\mathbf{x})]$$

  - For color SR:

$$\hat{\mathbf{x}}_{\mathbf{C}} = \underset{\mathbf{x}}{\text{ArgMin}}\,[f_C(\mathbf{x})] = \underset{\mathbf{x}}{\text{ArgMin}}\,[J_0(\mathbf{x}) + \lambda J_1(\mathbf{x}) + \lambda' J_2(\mathbf{x}) + \lambda'' J_3(\mathbf{x})]$$

  where

  $J_0(\mathbf{x})$ is the data fidelity term (L_1-norm distance measure)
  $J_1(\mathbf{x})$ is the luminance spatial regularization (total variation-based)
  $J_2(\mathbf{x})$ is the chrominance spatial regularization (Tikhonov-based)
  $J_3(\mathbf{x})$ represents the intercolor dependencies (RGB edge consistency)

University of Victoria

- Farsiu et al. obtain $\hat{\mathbf{x}}_{\mathbf{G}}$ and $\hat{\mathbf{x}}_{\mathbf{C}}$ by using a steepest-descent algorithm with fixed step size, which has the following disadvantages:

Iterative Reconstruction *Cont'd*

- Farsiu et al. obtain $\hat{\mathbf{x}}_{\mathbf{G}}$ and $\hat{\mathbf{x}}_{\mathbf{C}}$ by using a steepest-descent algorithm with fixed step size, which has the following disadvantages:
  - Slow convergence (takes an indirect path to the minimizer)

University of Victoria

Iterative Reconstruction *Cont'd*

- Farsiu et al. obtain $\hat{\mathbf{x}}_\mathbf{G}$ and $\hat{\mathbf{x}}_\mathbf{C}$ by using a steepest-descent algorithm with fixed step size, which has the following disadvantages:
  - Slow convergence (takes an indirect path to the minimizer)
  - Is susceptible to numerical ill-conditioning

University of Victoria

Iterative Reconstruction *Cont'd*

- Farsiu et al. obtain $\hat{\mathbf{x}}_{\mathbf{G}}$ and $\hat{\mathbf{x}}_{\mathbf{C}}$ by using a steepest-descent algorithm with fixed step size, which has the following disadvantages:
  - Slow convergence (takes an indirect path to the minimizer)
  - Is susceptible to numerical ill-conditioning
  - Requires empirical determination of the step size

University of Victoria

Iterative Reconstruction *Cont'd*

- Farsiu et al. obtain $\hat{\mathbf{x}}_\mathbf{G}$ and $\hat{\mathbf{x}}_\mathbf{C}$ by using a steepest-descent algorithm with fixed step size, which has the following disadvantages:
  - Slow convergence (takes an indirect path to the minimizer)
  - Is susceptible to numerical ill-conditioning
  - Requires empirical determination of the step size
- By contrast, quasi-Newton algorithms offer the following advantages:

University of Victoria

Iterative Reconstruction *Cont'd*

- Farsiu et al. obtain $\hat{\mathbf{x}}_{\mathbf{G}}$ and $\hat{\mathbf{x}}_{\mathbf{C}}$ by using a steepest-descent algorithm with fixed step size, which has the following disadvantages:
  - Slow convergence (takes an indirect path to the minimizer)
  - Is susceptible to numerical ill-conditioning
  - Requires empirical determination of the step size
- By contrast, quasi-Newton algorithms offer the following advantages:
  - Fast convergence (they take a more direct path to the minimizer)

University of Victoria

Iterative Reconstruction *Cont'd*

- Farsiu et al. obtain $\hat{\mathbf{x}}_{\mathbf{G}}$ and $\hat{\mathbf{x}}_{\mathbf{C}}$ by using a steepest-descent algorithm with fixed step size, which has the following disadvantages:
  - Slow convergence (takes an indirect path to the minimizer)
  - Is susceptible to numerical ill-conditioning
  - Requires empirical determination of the step size
- By contrast, quasi-Newton algorithms offer the following advantages:
  - Fast convergence (they take a more direct path to the minimizer)
  - Better performance in numerically ill-conditioned problems such as the super-resolution problem

University of Victoria

Iterative Reconstruction *Cont'd*

- Farsiu et al. obtain $\hat{\mathbf{x}}_\mathbf{G}$ and $\hat{\mathbf{x}}_\mathbf{C}$ by using a steepest-descent algorithm with fixed step size, which has the following disadvantages:
  - Slow convergence (takes an indirect path to the minimizer)
  - Is susceptible to numerical ill-conditioning
  - Requires empirical determination of the step size
- By contrast, quasi-Newton algorithms offer the following advantages:
  - Fast convergence (they take a more direct path to the minimizer)
  - Better performance in numerically ill-conditioned problems such as the super-resolution problem
  - The step size is adaptively recalculated by the algorithm in each iteration

University of Victoria

Quasi-Newton-Based Reconstruction

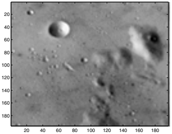
- Quasi-Newton algorithms involve the following basic steps:
  - $\hat{\mathbf{x}}_{n+1} = \hat{\mathbf{x}}_n + \beta_n \mathbf{d}_n$
  - $\mathbf{d}_n = -\mathbf{S}_n \mathbf{g_n}$
  - $\mathbf{g_n} = \nabla f(\mathbf{x}_n)$
  - $\mathbf{S}_n \approx \mathbf{H}^{-1}(\hat{\mathbf{x}}_\mathbf{n})$
  - $\beta_n = \text{ArgMin}_\beta\, f(\hat{\mathbf{x}}_n + \beta \mathbf{d}_n)$ is found using an inexact line search.

Algorithm	Search direction	Storage requirements
BFGS	$\mathbf{d_{n+1}}(\mathbf{x_0}, \mathbf{x_1}, \ldots, \mathbf{x_{n+1}}, \mathbf{g_0}, \mathbf{g_1}, \ldots, \mathbf{g_{n+1}})$	Large
LBFGS	$\mathbf{d_{n+1}}(\mathbf{x_{n-M+1}}, \ldots, \mathbf{x_{n+1}}, \mathbf{g_{n-M+1}}, \ldots, \mathbf{g_{n+1}})$	Moderate
MBFGS	$\mathbf{d_{n+1}}(\mathbf{x_n}, \mathbf{x_{n+1}}, \mathbf{g_n}, \mathbf{g_{n+1}})$	Low

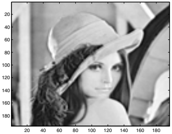
University of Victoria

Simulation Results

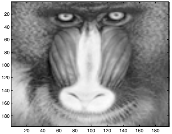
- From 3 different grayscale 192×192-pixel images, we generated 9 LR sets with different amounts of blur and noise:



‘Moon’



‘Lena’



‘Baboon’

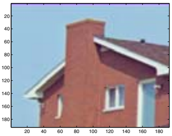
LR Set Type	SNR (dB)	Blur σ (pixels)
a	24	1
b	24	3
c	18	1

$N = 16$, images obtained by applying uniform translational warping, Gaussian blur, a downsample factor of 4, and additive Gaussian noise.

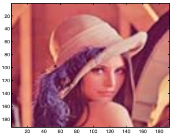
University of Victoria

Simulation Results, *Cont'd*

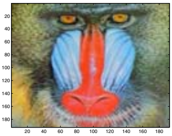
- Also, from 3 different color 192×192-pixel images, we generated 12 LR sets with different amounts of blur and noise:



'House'



'Lena'



'Baboon'

LR Set Type	SNR (dB)	Blur σ (pixels)
a	30	1
b	30	3
c	18	1
d	18	3

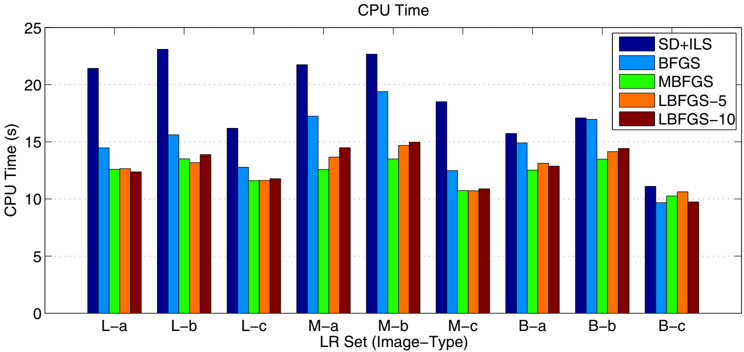
$N = 10$, images obtained by applying uniform translational warping, Gaussian blur, a downsample factor of 4, Bayer color filtering, and additive Gaussian noise.

University of Victoria

Simulation Results, *Cont'd*

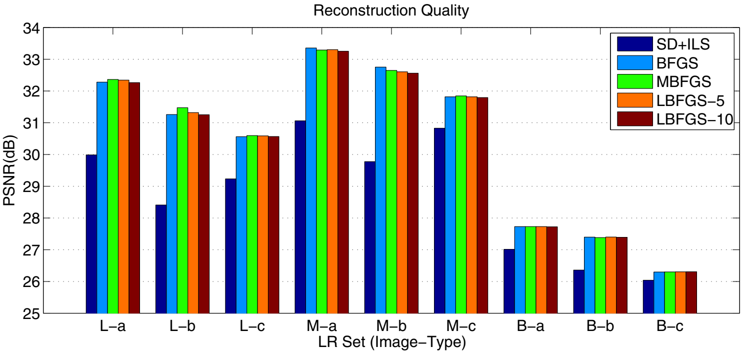
- For each one of the 9 LR grayscale sets and the 12 LR color sets, we performed reconstruction using
  - the steepest-descent algorithm with β adaptively selected using an inexact line search.
  - the BFGS algorithm
  - the MBFGS algorithm
  - the LBFGS algorithm with $M = 5$, and another with $M = 10$

- CPU time for grayscale super-resolution:



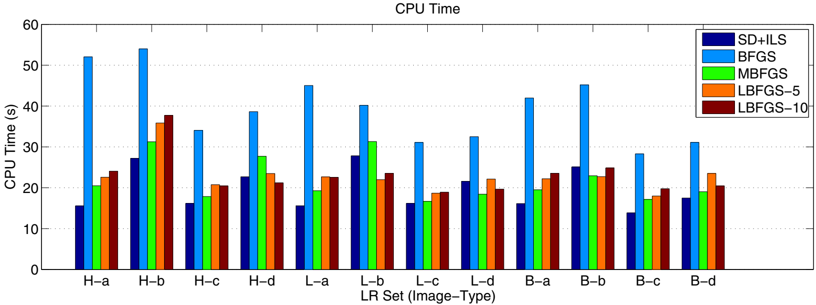
University of Victoria

- Reconstruction quality for grayscale super-resolution:



Simulation Results, *Cont'd*

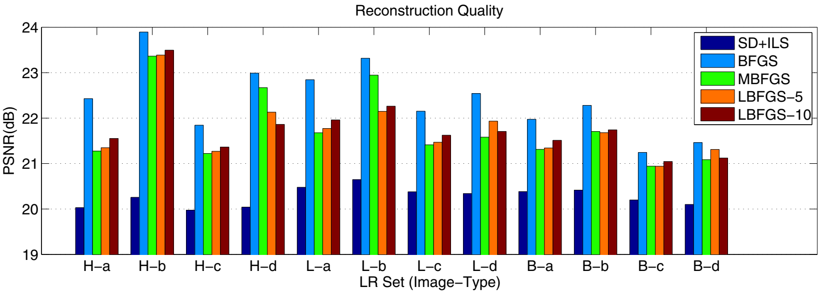
- CPU time for color super-resolution:



University of Victoria

Simulation Results, *Cont'd*

- Reconstruction quality for color super-resolution:



Conclusions

- The grayscale and color super-resolution algorithms of Farsiu et al. were implemented in terms of two *storage-efficient* quasi-Newton algorithms.

Conclusions

- The grayscale and color super-resolution algorithms of Farsiu et al. were implemented in terms of two *storage-efficient* quasi-Newton algorithms.
- The *MBFGS and LBFGS algorithms outperform the steepest-descent algorithms* while requiring negligible or limited additional storage.

University of Victoria

Conclusions

- The grayscale and color super-resolution algorithms of Farsiu et al. were implemented in terms of two *storage-efficient* quasi-Newton algorithms.
- The *MBFGS and LBFGS algorithms outperform the steepest-descent algorithms* while requiring negligible or limited additional storage.
- For grayscale super-resolution, the *MBFGS algorithm offers the best overall performance* in terms of speed, reconstruction quality, and storage requirements.

University of Victoria

Conclusions

- The grayscale and color super-resolution algorithms of Farsiu et al. were implemented in terms of two *storage-efficient* quasi-Newton algorithms.
- The *MBFGS and LBFGS algorithms outperform the steepest-descent algorithms* while requiring negligible or limited additional storage.
- For grayscale super-resolution, the *MBFGS algorithm offers the best overall performance* in terms of speed, reconstruction quality, and storage requirements.
- For color super-resolution, the *LBFGS offers a meaningful compromise* between the amount of storage and computational effort required and the quality of the reconstruction.

University of Victoria

Thank you for your attention.