Kronecker Products-based Regularized Image Interpolation Techniques

Blas Trigueros (presenter)

Ricardo H. Castañeyra

Juan Valera

Domingo Rodriguez

University of Puerto Rico at Mayaguez

9/23/2009

HPEC 2009 Workshop

Problem Formulation

- **In this work, we present a parallel implementation of an image interpolation algorithm based on the *Tikhonov* regularization technique, for the restoration of a (high resolution) HR image from a (low resolution) LR noisy image.**

- **Previous work:**
  - **Direct interpolation (cubic spline, adaptive spline…)**
  - **Regularization functional-based interpolation [1].**
  - **Discrete Cosine Transform (DCT) based techniques [2].**

- **Solution approach:**
  - **Parallel implementation performed using pMATLAB based on the regularized interpolation technique proposed by Li Chen *et al*. [3].**
  - **Data parallelism is employed to reduce execution time by partitioning the image into overlapping subimages.**
  - **Overlapping technique is introduced to avoid edge distortions.**

Implementation Model

- **Observation model:**

$$g = DHf + n,$$

$$H = H_1 \otimes H_2 \qquad D = D_1 \otimes D_2$$

f: high-resolution image
g: low-resolution image
D: decimation matrix
H: low-pass filter (blurring)
n: additive noise

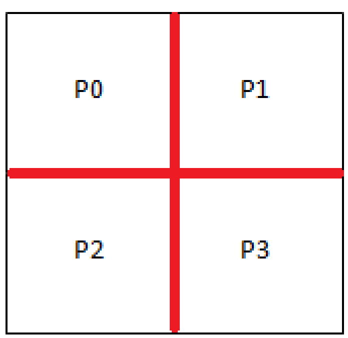
SVD:

$$\left.\begin{aligned} D_1H_1 &= U_1[\Sigma_1 \,|\, 0]V_1^T \\ D_2H_2 &= U_2[\Sigma_2 \,|\, 0]V_2^T \end{aligned}\right\} \longrightarrow \; min \left\| \begin{bmatrix} ([\Sigma_1 \,|0] \otimes [\Sigma_2 \,|0]) \\ \sqrt{\lambda} I \end{bmatrix} y - \begin{bmatrix} z \\ 0 \end{bmatrix} \right\|_2^2 \longrightarrow f = Vect(V_2 Mat(y) V_1^T)$$

  - **This algorithm takes advantage of SVD and Kronecker products to reduce the computational cost of the regularized solution.**
  - **But the SVD computation grows as O(N^3), prohibitive for large matrices.**

- **Parallel Implementation:**
  - **In our implementation of the algorithm, the image is partitioned into subimages, and each processor computes a portion of the final result.**

Overlapping zone



Each processor *Pi* computes an SVD of dimension: $\frac{1}{\sqrt{Np}} \times N \times kN$

***NxN*: dimension of the LR image**
***k*: decimation factor**
***Np*: number of processors**

Performance Results

HR image size	Execution time (s) MATLAB serial	Execution time (s) pMATLAB
256 x 256	0.0347	0.0067
512 x 512	0.3397	0.0399
1024 x 1024	1.96	0.2480
2048 x 2048	18.62	1.85
4096 x 4096	211	20.44
8192 x 8192	2471	219
16384 x 16384	13864	1982

Table 1: Execution time results of the interpolation algorithm in MATLAB and pMATLAB (4 processors, 2x2 mapping)

(a) Figure 1: (b)
(a) LR image, (b) HR image.

✓Computation time is reduced by using distributed arrays on pMATLAB for the SVD operation.

- **References:**

  - ***[1] Julie Kamm and James G. Nagy, "Kronecker product and SVD approximations in image restoration," Linear Algebra and its Applications, Vol. 284, pp. 177-192, Jan 1998***
  - ***[2] Yoshinori Abe, Youji Iuguni, "Image restoration from a downsampled image by using DCT," Signal Processing, 87, pp. 2370-2380, Mar 2007***
  - ***[3] Li Chen and Kim-Hui Yap, "Regularized Interpolation Using Kronecker Product for Still Images," IEEE International Conference on Image Processing, Vol. 2, pp. 1014-17, Sep 2005.***