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
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:**
  \[ g = DHf + n, \]
  \[ H = H_1 \otimes H_2 \]
  \[ D = D_1 \otimes D_2 \]

- **SVD:**
  \[ D_1 H_1 = U_1 [\Sigma_1 | 0] V_1^T \]
  \[ D_2 H_2 = U_2 [\Sigma_2 | 0] V_2^T \]

  \[ \rightarrow \min \left\| \left( \left[ \begin{array}{c} [\Sigma_1 | 0] \otimes [\Sigma_2 | 0] \\ \sqrt{\lambda} I \end{array} \right] \right) y - \left( \begin{array}{c} z \\ 0 \end{array} \right) \right\|_2^2 \rightarrow f = \text{Vect}(V_2 \text{Mat}(y)V_1^T) \]

- 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.

  Each processor $P_i$ computes an SVD of dimension:
  \[ \frac{f}{\sqrt{Np}} \times N \times kN \]

  $N \times N$: dimension of the LR image
  $k$: decimation factor
  $Np$: number of processors
Performance Results

<table>
<thead>
<tr>
<th>HR image size</th>
<th>Execution time (s) MATLAB serial</th>
<th>Execution time (s) pMATLAB</th>
</tr>
</thead>
<tbody>
<tr>
<td>256 x 256</td>
<td>0.0347</td>
<td>0.0067</td>
</tr>
<tr>
<td>512 x 512</td>
<td>0.3397</td>
<td>0.0399</td>
</tr>
<tr>
<td>1024 x 1024</td>
<td>1.96</td>
<td>0.2480</td>
</tr>
<tr>
<td>2048 x 2048</td>
<td>18.62</td>
<td>1.85</td>
</tr>
<tr>
<td>4096 x 4096</td>
<td>211</td>
<td>20.44</td>
</tr>
<tr>
<td>8192 x 8192</td>
<td>2471</td>
<td>219</td>
</tr>
<tr>
<td>16384 x 16384</td>
<td>13864</td>
<td>1982</td>
</tr>
</tbody>
</table>

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

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

• References: