A High-Speed and Low-Complexity Lens Distortion Correction Processor for Wide-Angle Cameras

Won-Tae Kim, Hui-Sung Jeong, Gwang-Ho Lee, Tae-Hwan Kim

Korea Aerospace Univ., Republic of Korea
Background

◆ Image sensing systems with a wide-angle camera

More images

Lens distortion

Barrel distortion

Distorted image

◆ Lens Distortion Correction (LDC) processor is required for the image sensing systems.

- High correction speed for the real-time operation
- Low-complexity implementation for the miniaturized low-cost systems
LDC Process

◆ Backward-mapping-based LDC process

- Step 1: Backward mapping
- Step 2: Interpolation

\[ P' = P \times \text{Scaling factor} \]

\[
\text{CIS Data} = \text{DIS Data}_0 \times b \times d + \text{DIS Data}_1 \times a \times d + \text{DIS Data}_2 \times b \times c + \text{DIS Data}_3 \times a \times c
\]
Proposed LDC Processor

**Low-complexity backward mapping**

- 7-stage pipeline
- Stage 1 to 4: Backward mapping
- Stage 5 to 7: Bilinear interpolation

---

**Conventional scaling factor**

\[ s_n = a + b \times r_n^2 + c \times r_n^4 \quad (r_n^2 = u^2 + v^2) \]

**Applying the incremental method**

\[ s_{n+1} - s_n = (r_{n+1}^2 - r_n^2) (b+c(r_{n+1}^2 + r_n^2)) \]

\[ r_{n+1}^2 - r_n^2 = 2u+1 = \{u,1\} \]

\[ r_{n+1}^2 + r_n^2 = 2r_n^2 + 2u+1 = r_n^2 << 1 + \{u,1\} \]

**Proposed scaling factor**

\[ s_{n+1} = s_n + \{u,1\}(b+c(r_n^2 << 1 + \{u,1\})) \]

---

**Architecture Comparison**

<table>
<thead>
<tr>
<th>Architecture</th>
<th>This work</th>
<th>[2]</th>
</tr>
</thead>
<tbody>
<tr>
<td>24 bit × 24 bit multiplier</td>
<td>0</td>
<td>7</td>
</tr>
<tr>
<td>24 bit × 16 bit multiplier</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>19 bit × 12 bit multiplier</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>16 bit × 16 bit multiplier</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>12 bit × 11 bit multiplier</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>11 bit × 7 bit multiplier</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

---

Proposed LDC Processor

- Efficient memory interface

Read operations for the bilinear interpolation have the spatial locality.
Results

◆ Implementation results

<table>
<thead>
<tr>
<th></th>
<th>This work</th>
<th>[1]</th>
<th>[2]</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CMOS technology (µm)</strong></td>
<td>0.11</td>
<td>0.18</td>
<td>0.18</td>
</tr>
<tr>
<td><strong>Equivalent gate count</strong></td>
<td>17223</td>
<td>44992</td>
<td>13917</td>
</tr>
<tr>
<td><strong>Correction speed (Mpixels/s)</strong></td>
<td>205</td>
<td>140</td>
<td>40</td>
</tr>
<tr>
<td><strong>Max. freq (MHz)</strong></td>
<td>370</td>
<td>200</td>
<td>200</td>
</tr>
<tr>
<td><strong>DIS memory size (byte)</strong></td>
<td>4M</td>
<td>4 x 4M</td>
<td>4M</td>
</tr>
<tr>
<td><strong>Maximum supported frame size</strong></td>
<td>2048 x 2048</td>
<td>1024 x 1024</td>
<td>1024 x 1024</td>
</tr>
<tr>
<td><strong>Power consumption (mW)</strong></td>
<td>9.77</td>
<td>27.86</td>
<td>12.68</td>
</tr>
<tr>
<td><strong>FOM (Kpixels/s)</strong></td>
<td>11.9</td>
<td>3.1</td>
<td>2.8</td>
</tr>
</tbody>
</table>

1) The smallest 2 input NAND cell is counted as one.
2) 8-bit gray scale, Frame size : 2048 x 2048
3) External memory power (0.11µm tech. process, single-port synchronous memory, time units : 1 ns)
4) Figure of Merit (FOM) : Correction speed(Mpixels/s)/Equivalent gate count(K)

4 times more efficient than references in terms of FOM

◆ Chip layout and demonstration result

![Chip Layout](image1)

![FPGA Demo System](image2)

**Demonstration Result**
