Accelerating a learning–based image processing pipeline for digital cameras

Local, Linear and Learned ($L^3$) pipeline

Qiyuan Tian and Haomiao Jiang

Department of Electrical Engineering
Stanford University

GPU Technology Conference, San Jose
March 17, 2015
Digital camera sub-systems

Focus control

Exposure control

Lens, aperture and sensor

Pre-processing
- dead pixel removal
- dark floor subtraction
- structured noise reduction
- quantization
- etc.

RAW image

Image processing pipeline

Display image

Transform the sensor data into a display image
Standard image processing pipeline

- Requires multiple algorithms
- Each algorithm requires optimization
- Optimized only for Bayer (RGB) color filter array (CFA)
Opportunity

Extra sensor pixels enable new CFAs that improve sensor functionality and open new applications

Challenge
- Customized image processing pipeline
- Speed and low power
L³ image processing pipeline

Local, Linear and Learned (L³)
- Combines multiple algorithms into one
- Rendering is simple, fast and low-power
- Uses machine learning to optimize the class transforms for any CFA
Classify pixels

RAW image

Center pixel color

Intensity

Contrast

Class
Center pixel color: red
Intensity: high
Contrast: flat

“Local” pixel values (local patch)
Retrieve and apply transforms

RAW image

“Linear” transforms

R  G  B

Class
Center pixel color: red
Intensity: high
Contrast: flat

Weighted summation

Learned table of linear transforms

Intensity

Contrast

Rendered R, G, B values
Table-based architecture suits GPU

- Independent calculation for each pixel
- Simple weighted summation

Thus well-suited for parallel rendering using GPU
GPU implementations

Render one pixel \((i, j)\)
- Calculate class index
- Retrieve transforms
- Weight sum

Table of transforms

Constants, e.g. CFA pattern
GPU acceleration results

- GPU: NVidia GTX 770 (1536 kernels, 1.085 GHz)
- CPU: Intel Core i7-4770K (3.5 GHz)
- CUDA/C programming
Potential speed improvement
Use shared memory and registers

Specialized image signal processor (ISP)

$L^3$ ISP
L³ processing

Pre-processing

RAW Image

Local Patch Classification

Classification Map

Transform Application

Display image

Table of Transforms

```
<table>
<thead>
<tr>
<th>Class</th>
<th>R</th>
<th>G</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class1</td>
<td>☐</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>Class2</td>
<td>☐</td>
<td>☑</td>
<td>☐</td>
</tr>
<tr>
<td></td>
<td>☑</td>
<td>☐</td>
<td>☐</td>
</tr>
</tbody>
</table>
```

"Learn" the transforms

GPU
Locally linear transform

- Globally nonlinear for an entire image
- 480 linear transforms in total
Learn the locally linear transform for each class

Local RAW values \( A \)  
Linear transform \( x \)  
Desired RGB values \( b \)
Solve the transform

\[ A \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = b \]

minimize \[ \|Ax - b\|^2 + \|\Gamma x\|^2 \]

ridge regression
Training data from camera simulation

- Simulate any camera designs
- Various training scenes, illuminants and luminances
- Registered and desired RGB images
Learned transforms

- Accounts for spatial and spectral correlation
- Accounts for sensor and photon noise
Advantages of learning

- Adapts to any application and scene content

  Consumer Photography  Document Digitization  Industrial Inspection  Pathology  Endoscopy

- Adapt to any CFA

  Bayer  RGBW  RGBX  RGBCMY  Medical
Solve RGBW rendering

In dark scene
- Two f-stops gain

In bright scene
- Same performance

Simulation conditions
Exposure: 100 ms
F-number: f/4

Tian et al. 2014
Smooth transition from dark to bright

Scene Luminance

Bayer

RGBW

Tian et al. 2014
Compare RGBW CFA designs

Bayer

Parmar & Wandell, 2009

Aptina CLARITY+

Kodak

Wang et al., 2011

Simulation conditions
Luminance: 1 cd/m²
Exposure: 100 ms
F-number: f/4

Tian et al. 2014
Five-band camera prototype

RGB Cyan Orange
4×4 super-pixel

Tian et al. 2015
$L^3$ solves five-band prototype rendering

Tian et al. 2015
### GPU acceleration results

<table>
<thead>
<tr>
<th>Results</th>
<th>GPU</th>
<th>CPU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image (1280×720)</td>
<td>0.062s (16 fps)</td>
<td>12.4s</td>
</tr>
<tr>
<td>Video (1280×720×1800)</td>
<td>163.2s (11 fps)</td>
<td></td>
</tr>
</tbody>
</table>

- **GPU**: NVidia GTX 770 (1536 kernels, 1.085 GHz)
- **CPU**: Intel Core i7-4770K (3.5 GHz)
- **CUDA/C programming**

Tian et al. 2015
L³ learning

Novel Camera

Camera Calibration

Calibrated Parameters

Multispectral Scenes

ISET camera Simulation

Simulated RAW Image

Supervised Learning

Desired RGB Images

Table of Transforms

L³ processing

Novel Camera

Pre-processing

RAW Image

Local Patch Classification

Classification Map

Transform Application

GPU

Display image

Table of transforms
Local, linear and learned pipeline (L³) summary

- Table-based rendering architecture is ideal for GPU acceleration

- Machine learning automates image processing for any CFA and scene content

Rethink image processing pipeline
Acknowledgement

Advisors
  Brian Wandell, Joyce Farrell

Group members
  Henryk Blasinski, Andy Lin

Stanford collaborators
  Francois Germain, Iretiayo Akinola

Olympus collaborators
  Steven Lansel, Munenori Fukunishi

End

Thanks for your attention!

Questions?

Contacts
  qytian@stanford.edu
  hjiang36@stanford.edu
Potential speed improvement

• Local vs Global
  • L3 is locally linear: can use local memory to speed up
  • Locality in memory: writing output as RGBRGB is faster than writing as image plane

• Device based optimization
  • CFA pattern and other parameters are fixed: Constant Memory & no need to pass in
  • Symmetry and other properties

• CUDA, GLSL, FPGA, Hardware
  • L3 rendering is based on linear transforms and can be implemented with shaders or hardware circuits to achieve further acceleration