WYSIWYG Computational Photography via Viewfinder Editing

Jongmin Baek\textsuperscript{13}  Dawid Pają\textsuperscript{2}  Kihwan Kim\textsuperscript{2}  Kari Pulli\textsuperscript{2}  Marc Levoy\textsuperscript{3}

\textsuperscript{1}Dropbox, Inc.  \textsuperscript{2}NVIDIA Research  \textsuperscript{3}Stanford University
Enhance photography via computation

Computational Photography

Deferred computation
This project

- Make computational photography WYSIWYG!

**Edit on a “live” viewfinder**

feedback for user & application

user input
Desired user experience

Without edit

With edit
1. Viewfinder editing
2. Appearance-based camera control
3. Implementation and results

This talk

Outline
Viewfinder editing

User selects a region

Find all similar regions (in each frame)

User specifies an edit

Apply the edit onto the regions (in each frame)
Overview for region selection

User strokes → Subsequent Viewfinder frames → Patches in each frame → Lookup → High-dimensional data structure → Store → Processing → Compute response
+ Multi-scale texture lookup

Red: patches matched from the finest scale.
Green: patches matched from the medium scale (among those remaining)
Blue: patches matched from the coarsest scale (among those remaining)
+ Spatial filtering

[Gastal and Oliveira, SIGGRAPH 2011]
+ Temporal filtering
In action...
Comparison of methods (ms)

- Ours
- Li et al., 2010
- Bie et al., 2011
- Xu et al., 2009
- Farbman et al., 2010
- Li et al., 2008
- An and Pellacini, 2008
- Chen et al., 2012

Per-frame runtime on multi-core x86 machine
Comparison of methods (ms)

<table>
<thead>
<tr>
<th>Method</th>
<th>Total</th>
<th>Per frame</th>
<th>Precomputation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours</td>
<td>30 fps</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Li et al., 2010</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bie et al., 2011</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Xu et al., 2009</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Farbman et al., 2010</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Li et al., 2008</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>An and Pellacini, 2008</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chen et al., 2012</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Chen et al., 2012
An and Pellacini, 2008
Li et al., 2010
Farbman et al., 2010
Li et al., 2008
An and Pellacini, 2008
Chen et al., 2012

0 10 20 30 40 50 60 70 80 90 100
1. Viewfinder editing

2. Appearance-based camera control

3. Implementation and results

This talk
HDR Metering: Status Quo

- Determine the **# of exposures** and **duration of each exposure** based on the **scene content**.

- What if the camera is now WYSIWYG?

- Determine the **# of exposures** and **duration of each exposure** based on the **displayed content**.

  - Take each of short, medium long exposures if and only if necessary.
  - Minimize capture time, leading to less motion blur, handshake, etc.
Evaluation (synthetic)

[Hasinoff et al., CVPR 2010]
135 ms total exposure

[Proposed]
135 ms total exposure
Evaluation (synthetic)

[Hasinoff et al., CVPR 2010]  [Proposed]

135ms total exposure
Evaluation (dynamic scene)

[Hasinoff et al., CVPR 2010]
85 ms total exposure

[Proposed]
27 ms total exposure
Evaluation (local edits)

[Without edit]
17 ms total exposure

[With edit]
96 ms total exposure
1. Viewfinder editing

2. Appearance-based camera control

3. Implementation and results

This talk
Platform

- NVIDIA developer tablet

- Tegra 3 SoC
- Cortex-A9 CPU
- ULP GeForce GPU
- 30-fps camera with FCam API
System pipeline

Viewfinder stream
Latest N frames

Registration + Blending

HDR composite

Generate edit maps

Edit masks

Tonemap + Edit

Appearance-based metering
Results (Local tonal edit)

Viewfinder interaction
Results (Local tonal edit)

Viewfinder interaction
Results (focal stack composition)

Viewfinder interaction

Input  Reference  Output
Summary

• Design and implementation of Viewfinder Editing
  • Algorithmic advances.
  • Reclaim WYSIWYG experience.

• Useful for Computational Photography
  • Better camera control for better image quality

• Possible in today’s mobile devices
  • Harness computes in CPU, GPU
Questions?