A Neural Approach to Blind Motion Deblurring

Ayan Chakrabarti (TTI-Chicago)
[ayanc@ttic.edu]

**Motion De-Blurring**

- Exposure Interval
  - "Camera Shake"
- Blur Kernel
  - "Latent" Sharp Image
- Noise
  - Fast Exposure
    - Small Aperture
  - Large Aperture
    - Defocus Blur
  - Long Exposure
    - Small Aperture
    - Motion Blur

**Proposed Architecture**

- Goal: Given a blurry patch as input, estimate the sharp version.
- Our approach: Train network to output the complex Fourier coefficients of a restoration filter.

**Current Approaches**

- Kernel Estimate
- Image Estimate
- Observing Image
- Observed Image
- Image Features
- Sharp Features
- Learn Neural Network to Restore Individual Patches independently & in parallel.

**Our Approach**

- State of the Art: Iterative Refinement
  - Accurate, but slow (40-90 mins / image)
  - Sequential, not parallelizable.

Both are based on generative, expert-chosen image priors.
We want something fast and reliable: explore discriminative training!

**Experiments**

Evaluation on [Sun et al., 2013] dataset.

<table>
<thead>
<tr>
<th>Method</th>
<th>Mean</th>
<th>95% UI</th>
<th>Max</th>
<th>Success Rate (%)</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed</td>
<td>3.01</td>
<td>5.76</td>
<td>11.04</td>
<td>92%</td>
<td>65s (GPU)</td>
</tr>
<tr>
<td>Mishra &amp; Ismail (2014)</td>
<td>2.57</td>
<td>4.49</td>
<td>9.31</td>
<td>94%</td>
<td>3min (CPU)</td>
</tr>
<tr>
<td>Sun et al. (2013)</td>
<td>2.38</td>
<td>5.93</td>
<td>5.93</td>
<td>93%</td>
<td>3min (CPU)</td>
</tr>
<tr>
<td>Xu &amp; Xu (2010)</td>
<td>3.65</td>
<td>9.97</td>
<td>65.33</td>
<td>85%</td>
<td>25s (CPU)</td>
</tr>
<tr>
<td>Schuler et al. (2015)</td>
<td>4.55</td>
<td>11.21</td>
<td>20.98</td>
<td>67%</td>
<td>22s (CPU)</td>
</tr>
</tbody>
</table>

**Architecture**

- Used in past for non-blind deblurring
  - [Schuler et al., 2013], [Xu et al., 2014]
  - Fails to train for the general blind case
- [Sun et al., 2015]: classify from discrete set.
- [Schuler et al., 2015] estimate from whole image. Encouraging results, but lag state of the art.

Need to look beyond standard feed-forward architectures.