Optical Computing for Fast Light Transport Analysis, 2010

Authors: Matthew O’Toole and Kiriakos Kutulakos
Presented by: Allen Hawkes and Sida Wang
Basic Concept

- T-reconstruct pipeline: set up cameras/light sources, come up with sampling scheme, come up with algorithm to reconstruct T

$$p = Tl$$

- Similar pipeline, estimation of T via “optical computing”
  - Iterative numerical methods for matrix estimation
  - Scene lighting + image capture can replace a matrix-vector in these methods
  - Use this T-estimation as previously seen (scene relighting, light source detection)
Example: Optical Power Iteration

- want to get principal eigenvector of $T$
- use current photo as basis for illumination of next image capture
- iterate until convergence
- final image is principal eigenvector of $T$ [Trefethen and Bau 1997, non-optical]
- only 1 eigenvector, can be slow
Krylov Subspace Methods

- Power iteration after $k$ steps gives a span of vectors
- This span of vectors is the $k$-dim Krylov subspace

**Algorithm 1 The power iteration algorithm.**

**Numerical Implementation:**
- **In:** matrix $T$, iterations $K$
- **Out:** principal eigenvector of $T$

1. $l_1 =$ random vector
2. for $k = 1$ to $K$
3. \[ p_k = T l_k \]
4. $l_{k+1} = p_k / \| p_k \|_2$
5. return $l_{k+1}$

**Optical Implementation:**
- **In:** iterations $K$
- **Out:** principal eigenvector of $T$

1. $l_1 =$ positive vector
2. for $k = 1$ to $K$
3. illuminate scene with vector $l_k$
4. capture photo \& store in $p_k$
5. $l_{k+1} = p_k / \| p_k \|_2$
6. return $l_{k+1}$

span of vectors given initial illumination and $T$ (optical case)
Krylov Subspace Methods

Optical Matrix-vector products for general vectors

- need to multiply $T$ with negative vectors, can’t have negative light
- create a general vector as difference between positive, negative
- positive/negative light can be assigned to 2 different colors, for example

\[
1 = 1^p - 1^n
\]

\[
T 1 = (T 1^p) - (T 1^n).
\]
Krylov Subspace Methods

Symmetric vs. non-symmetric transport matrices

- Convergence of some methods dependent on T-symmetry
- 2 ways to enforce symmetry:
  - optically multiply by a different matrix which is symmetric

\[
\begin{align*}
T^* &= T^t T \\
T^* 1 &= T^t (T 1)
\end{align*}
\]
Transport Acquisition - Optical Arnoldi

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Numerical objective</th>
<th>Step 1</th>
<th>Step 4</th>
<th>Step 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power iteration</td>
<td>estimate principal eigenvector of ( T )</td>
<td>( l_1 = \text{positive vector} )</td>
<td>( l_{k+1} = p_k/|p_k|_2 )</td>
<td>\text{return } l_{k+1} )</td>
</tr>
<tr>
<td>(Section 2.1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arnoldi</td>
<td>compute rank-K approximation of ( T )</td>
<td>( l_1 = \text{non-zero vector} )</td>
<td>( l_{k+1} = \text{ortho}(l_1, \ldots, l_k, p_k) )</td>
<td>\text{return } [p_1 \cdots p_K][l_1 \cdots l_K]^t )</td>
</tr>
<tr>
<td>(Section 3)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Builds rank k approximation of \( T \)

\[
T \approx [p_1 \cdots p_K][l_1 \cdots l_K]^t
\]

\[
T \approx PL
\]
Transport Acquisition - Optical Arnoldi

- Output T is dense, low-rank (good for natural lighting scenes with minimal mirror reflection and sharp shadows)
- reverse of previous methods: eigenvectors of T used to acquire it
- relighting:

\[ p = [d_1 \cdots d_K][l_1 \cdots l_K]^t l \]
**Inverse Transport - Optical GMRES**

\[ l = \arg \min_x \| [p_1 \cdots p_K] [l_1 \cdots l_K]^T x - p \|_2 \]  

Least squares problem

- Difference with Arnoldi’s: Initial illumination vector is a photo \( p \)

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Numerical objective</th>
<th>Step 1</th>
<th>Step 4</th>
<th>Step 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power iteration (Section 2.1)</td>
<td>estimate principal eigenvector of ( T )</td>
<td>( l_1 = \text{positive vector} )</td>
<td>( l_{k+1} = p_k / | p_k |_2 )</td>
<td>\text{return } l_{k+1}</td>
</tr>
<tr>
<td>Arnoldi (Section 3)</td>
<td>compute rank-( K ) approximation of ( T )</td>
<td>( l_1 = \text{non-zero vector} )</td>
<td>( l_{k+1} = \text{ortho}(l_1, \ldots, l_k, p_k) )</td>
<td>\text{return } [p_1 \cdots p_K] [l_1 \cdots l_K]^T</td>
</tr>
<tr>
<td>Generalized minimal residual (Section 4)</td>
<td>find vector ( l ) such that ( p = T l )</td>
<td>( l_1 = \text{target photo } p )</td>
<td>( l_{k+1} = \text{ortho}(l_1, \ldots, l_k, p_k) )</td>
<td>\text{return } [l_1 \cdots l_K] [p_1 \cdots p_K]^T p</td>
</tr>
</tbody>
</table>
Inverse Transport - Optical GMRES
\[ \epsilon_k = \frac{\| T_k - \hat{T} \|_F}{\| \hat{T} \|_F} \]

- \( T \): ground truth matrix
Results - Acquiring Transport Matrices with Optical Arnoldi

(a) 10 iterations for Arnoldi, 200 photos
(b) 20 iterations for Arnoldi, 200 photos
(c) 50 iterations for Arnoldi, 200 photos
(d) Nystrom
spatially - localized

Actual photo
Difference between actual photo and relit image

rank 50
Conclusions

- Built optical methods from existing numerical methods
- Have theoretical bounds on convergence rate
- Methods applied easily to large matrices (i.e. T)

Limitations:

- Optical Arnoldi performs poorly for high-rank matrices
- Optical GMRES inverts only 1 image (at a time)
Pros and Cons

Grade: 1.1

Pros:
- simple, intuitive method requires relatively few photos
- is fast for relighting/inverse transport

Cons:
- leave most technical details in previous works