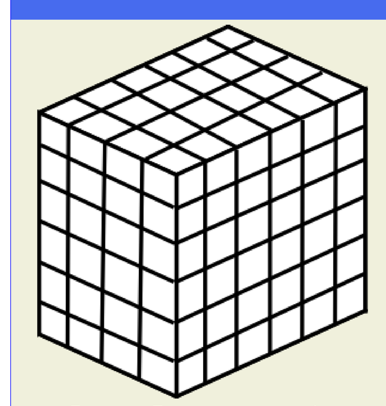
Ray Markov Random Fields for Image-Based 3D Modeling: Model and Efficient Inference

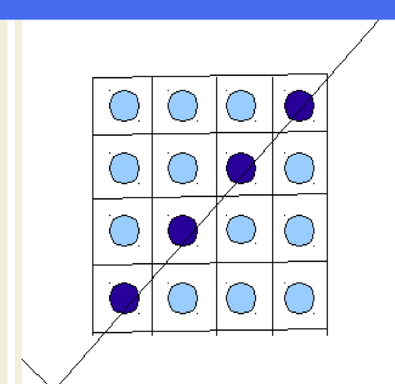
Shubao Liu and David B. Cooper
LEMS, Brown University
sbliu@lems.brown.edu, cooper@lems.brown.edu

Core Ideas

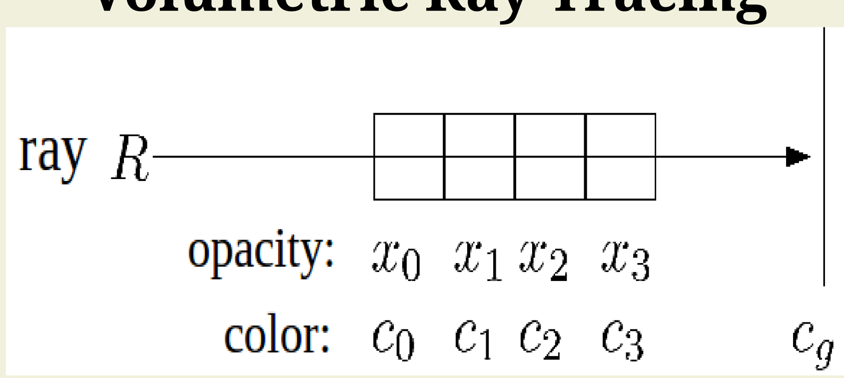
* **Optimization-based multi-view 3D reconstruction based on the idea of inversing the ray tracing process.**
* Uses MRF to model the multi-view image-formation process, where occlusion is accurately modeled.
* Each ray creates a clique, consisting of 100-1000 voxels the ray passes through.
* The RayMRF clique proposed is unusual because it constitutes a large number of random variables(voxels).
* Developed a highly efficient algorithm for 3D shape estimation based on loopy belief propagation and dynamic programming --- compact belief propagation.

Inverse Ray Tracing


3d volumetric representation


2d ray-volume intersection

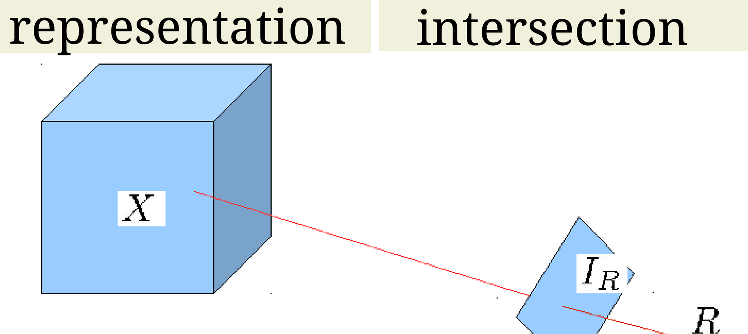
Volumetric Ray Tracing



$X_R = \{x_0, x_1, x_2, x_3; c_0, c_1, c_2, c_3, c_g\}$

$\times$ denotes any possible value

$$\Upsilon(X_R) = \begin{cases} c_0, & X_R = 1, \times, \times, \times; \quad c_0, \times, \times, \times, \times \\ c_1, & X_R = 0, 1, \times, , \times; \quad \times, c_1, \times, \times, \times \\ c_2, & X_R = 0, 0, 1, \times; \quad \times, \times, c_2, \times, \times \\ c_3, & X_R = 0, 0, 0, 1; \quad \times, \times, \times, c_3, \times \\ c_g, & X_R = 0, 0, 0, 0; \quad \times, \times, \times, \times, c_g \end{cases}$$



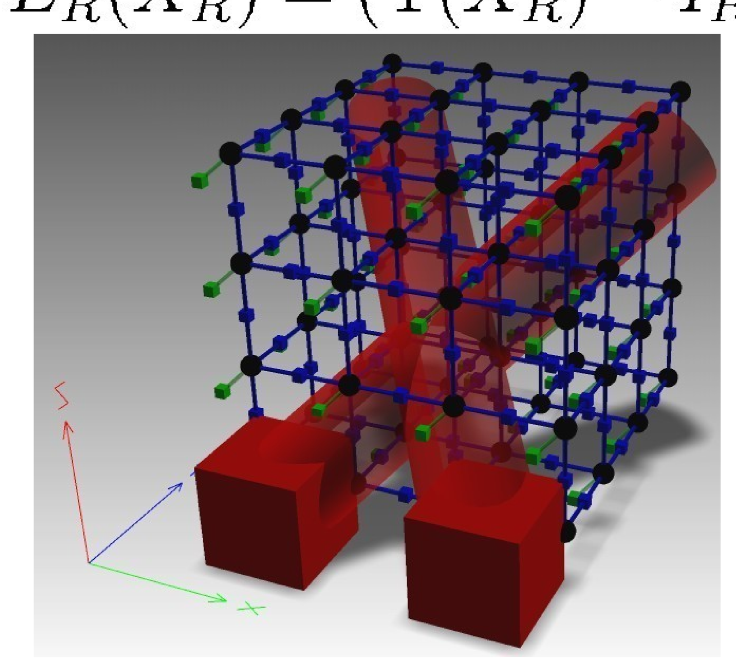
Analysis by Synthesis

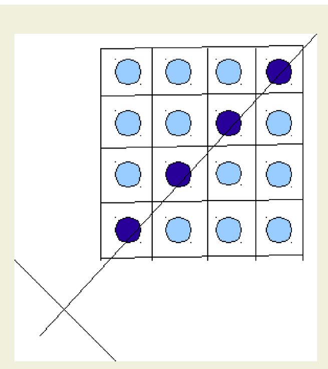
* projected pixel value from ray tracing: $\Upsilon(X_R)$
* observed pixel value: I_R
* We want to minimize their difference: $\min_X \sum_R ||\Upsilon(X_R) - I_R||^2$

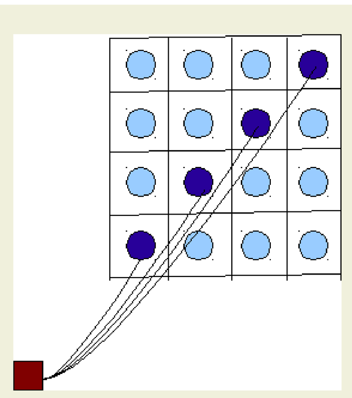
Ray Markov Random Fields

ray clique energy:

$$E_R(X_R) = (\Upsilon(X_R) - I_R)^2$$



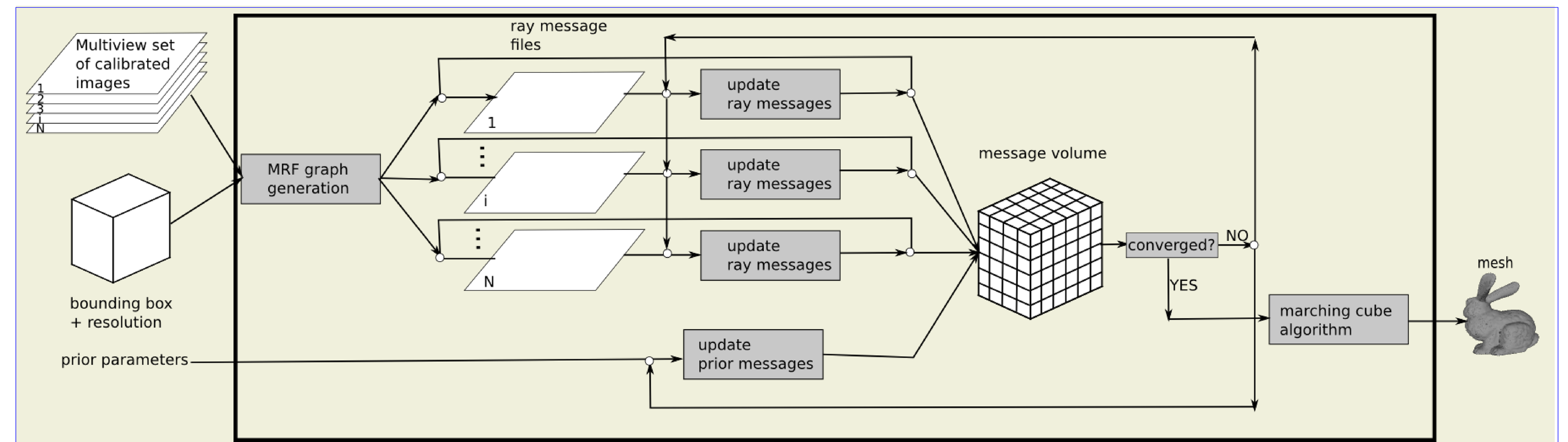

voxel-ray intersection


factor graph representation of the ray energy

Regularization / Prior: $E_u(x_i) = \begin{cases} \alpha_u, & x_i = 0 \\ 0, & x_i = 1 \end{cases}$

$$E_p(x_i, x_j) = \begin{cases} 0, & x_i = x_j; \\ \alpha_p, & x_i \neq x_j. \end{cases}$$

$$E = \sum_{R \in \mathcal{R}} E_R(X_R) + \sum_{\langle i,j \rangle \in \mathcal{P}} E_p(x_i, x_j) + \sum_{i \in \mathcal{U}} E_u(x_i)$$



system diagram

Compact Belief Propagation

Need to compute: $X^* = \arg_X \min E$

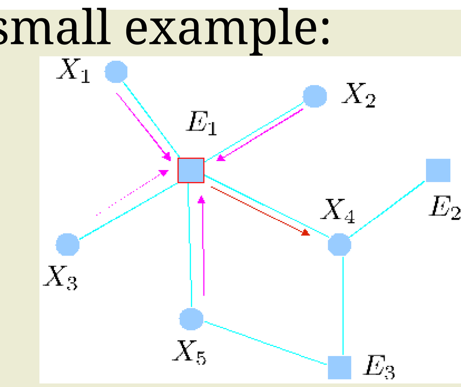
Loopy Belief Propagation:

In general:

$$m_{E_m \to x_i}(x_i) = \min_{x_k \in \mathcal{X}(E_m) \setminus \{x_i\}} \{E_m(\mathcal{X}(E_m)) + \sum_{x_k \in \mathcal{X}(E_m) \setminus \{x_i\}} m_{x_k \to E_m}(x_k)\}$$

a **small** optimization problem!

small example:



$$m_{F_1 \to x_4}(x_4) = \min_{x_1, x_2, x_3, x_5} \left\{ \begin{array}{l} F_1(x_1, x_2, x_3, x_4, x_5) + \\ m_{x_1 \to F_1}(x_1) + m_{x_2 \to F_1}(x_2) + \\ m_{x_3 \to F_1}(x_3) + m_{x_5 \to F_1}(x_5) \end{array} \right\}$$

BIG OBSTACLE:
For ray clique, the **small** combinational optimization is **LARGE**!
2^(1000) combinations for each ray-clique!

SOLUTION:
Explore the **compactness** of the clique energy!
Go back to the toy example: $E_R(x_0, x_1, x_2, x_3)$
(Note: observe the reduction in each color box below.)

$$m_{R \to x_1}(x_1 = 0) = \min_{x_0, x_2, x_3} \left\{ \begin{array}{l} E_R(x_0, 0, x_2, x_3) + \\ m_{x_0 \to R}(x_0) + m_{x_2 \to R}(x_2) + m_{x_3 \to R}(x_3) \end{array} \right\}$$

$$= \min \left\{ \begin{array}{l} E_R(0,0,0,0) + m_{x_0 \to R}(0) + m_{x_2 \to R}(0) + m_{x_3 \to R}(0) \\ E_R(0,0,0,1) + m_{x_0 \to R}(0) + m_{x_2 \to R}(0) + m_{x_3 \to R}(1) \\ E_R(0,0,1,0) + m_{x_0 \to R}(0) + m_{x_2 \to R}(1) + m_{x_3 \to R}(0) \\ E_R(0,0,1,1) + m_{x_0 \to R}(0) + m_{x_2 \to R}(1) + m_{x_3 \to R}(1) \\ E_R(1,0,0,0) + m_{x_0 \to R}(1) + m_{x_2 \to R}(0) + m_{x_3 \to R}(0) \\ E_R(1,0,0,1) + m_{x_0 \to R}(1) + m_{x_2 \to R}(0) + m_{x_3 \to R}(1) \\ E_R(1,0,1,0) + m_{x_0 \to R}(1) + m_{x_2 \to R}(1) + m_{x_3 \to R}(0) \\ E_R(1,0,1,1) + m_{x_0 \to R}(1) + m_{x_2 \to R}(1) + m_{x_3 \to R}(1) \end{array} \right\}$$

2^(4-1)=8 terms

In general: 2^(N-1) terms

$$= \min \left\{ \begin{array}{l} E_R(0,0,0,0) + m_{x_0 \to R}(0) + m_{x_2 \to R}(0) + m_{x_3 \to R}(0) \\ E_R(0,0,0,1) + m_{x_0 \to R}(0) + m_{x_2 \to R}(0) + m_{x_3 \to R}(1) \\ E_R(0,0,1,\times) + m_{x_0 \to R}(0) + m_{x_2 \to R}(1) + \min\{m_{x_3 \to R}(0), m_{x_3 \to R}(1)\} \\ E_R(1,0,\times,\times) + m_{x_0 \to R}(1) + \min\{m_{x_2 \to R}(0), m_{x_3 \to R}(1)\} + \min\{m_{x_3 \to R}(0), m_{x_3 \to R}(1)\} \end{array} \right\}$$

reduced to 4 terms; In general: N terms. And these terms can be reused when computing the message sent from the ray-clique to the other voxels on the ray.

SUMMARY

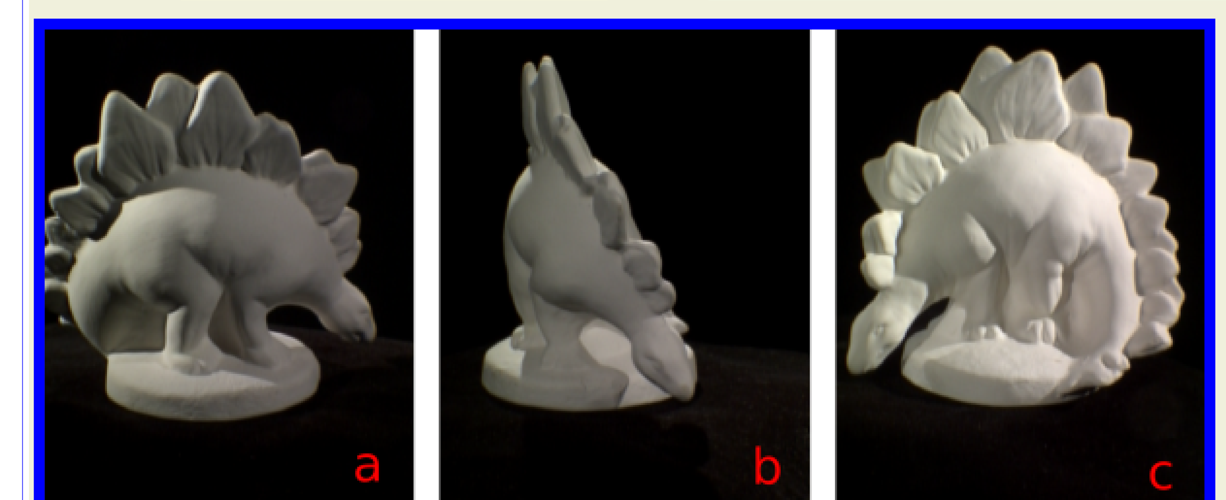
* **compact belief propagation = loopy belief propagation + dynamic programming.**
* Reduces the computational cost from **exponential to linear**, where N is the number of random variables in the clique.

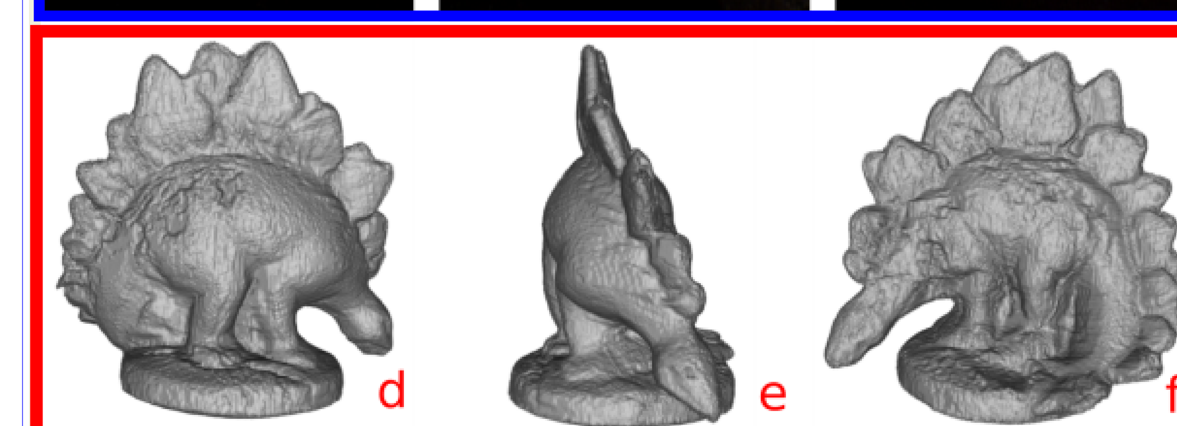
Experimental Results

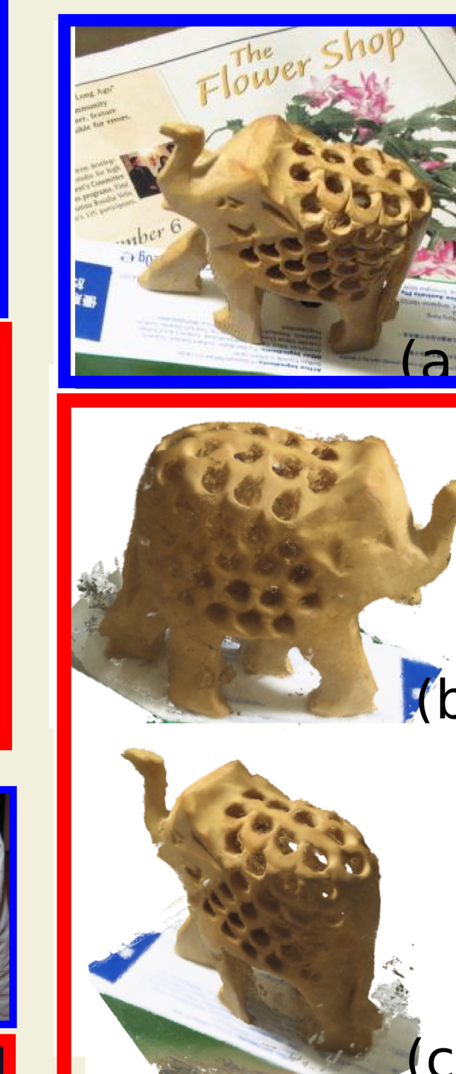
* **Accurate 3D reconstruction**: Comparable to the state-of-the-art with our current proof-of-concept implementation.
* **Photo-realistic**
* **Very general**: Handles objects of arbitrary topology, any camera configuration, with background and/or foreground clutter.
* **Automatic**: Only needs the user to specify a bounding box (This can be relaxed in the future.)
* Summary: A system for automatic reconstruction from hand-held camera or multiple stationary cameras.

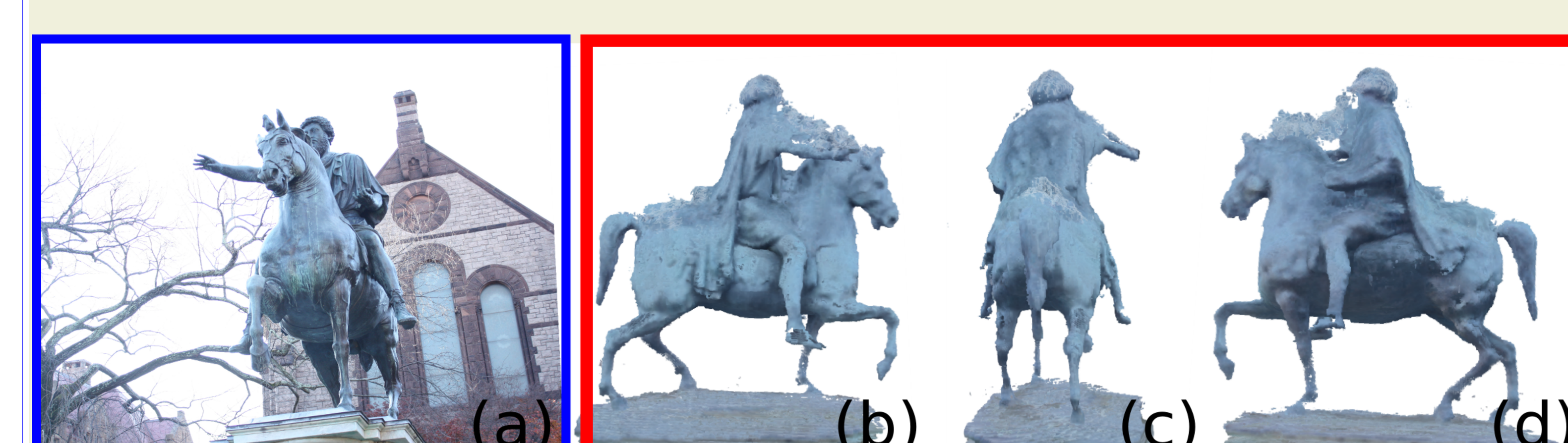
input images

rendered 3d reconstruction









Acknowledgement
The support of National Science Foundation under grant IIS-0808718 is gratefully acknowledged.