

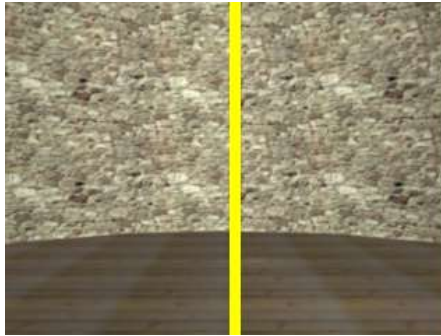
# Visual Hull Construction in the Presence of Partial Occlusion

**Li Guan, Sudipta Sinha, Jean-Sebastien Franco, Marc Pollefeys**  
**Department of Computer Science**  
**UNC-Chapel Hill**

{lguan, ssinha, franco, marc}@cs.unc.edu

# The Problem

- **For Silhouettes (Background Subtraction)**



- **For Visual Hull**



*The subject is not completely located in the constructed visual hull.*

Original Data from [www.mpi-sb.mpg.de/departments/irg3/kungfu/index.html](http://www.mpi-sb.mpg.de/departments/irg3/kungfu/index.html)

# Previous Work

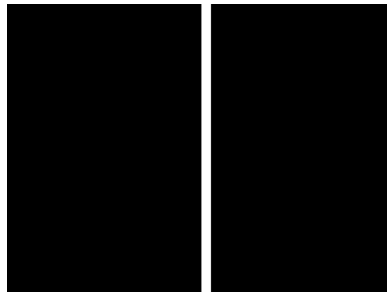
- **Occlusion-free environment**
  - **Theory**
    - Baumgart '74, Laurentini '94 etc.
  - **Volumetric visual hulls**
    - Martin et al. '83, Szeliski '93 Matusik et al. '00 etc.
  - **Polyhedral visual hull**
    - Matusik et al. '01, Franco et al. '03 etc.
  - **Other advanced representations**
- **Self-occlusion**
  - **Space Carving**
    - Kutulakos et al. '99 etc.
  - **Graph cut**
    - Vogiatzis et al. '05 etc.

# The Idea

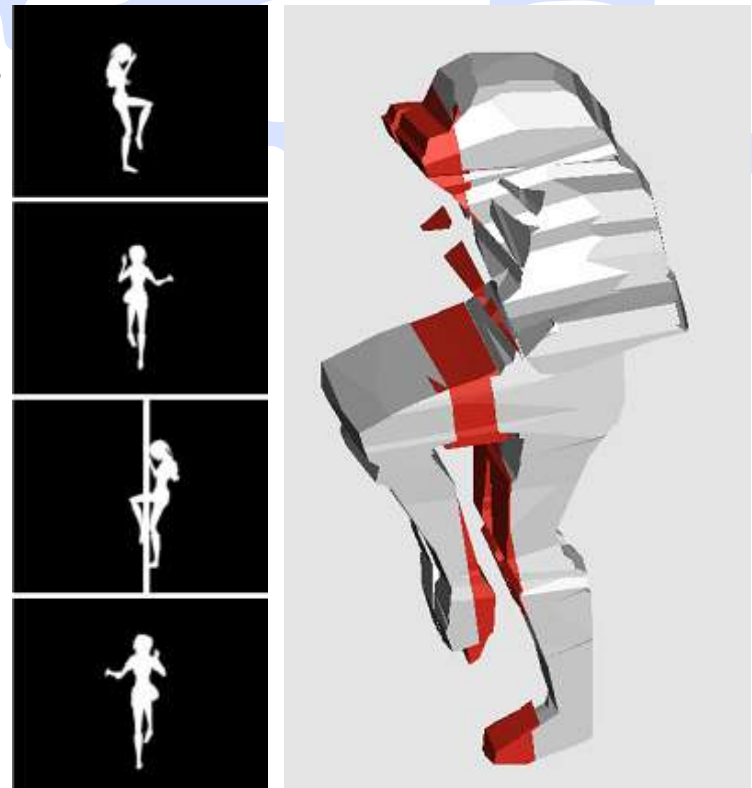
- **Occlusion Mask (similar to projection matrix)**

## Assumptions

- **Static Occluders**
- **Occluders in front of all moving objects**



- **Extended Silhouette**



# Outline

- **Occlusion Mask Extraction**
- **Main algorithm**
- **Algorithm Analysis**
- **Examples**
- **Summary**

# Occlusion Mask Extraction

- **For every camera**
- **Occluder boundary extraction using observation videos**
- **Independent of calibration**
- **On-line updating system**



# Foreground Blob

- **Brostow et al. Motion Based Decompositing of Video, *ICCV*, '99**



**Construction Site Video**

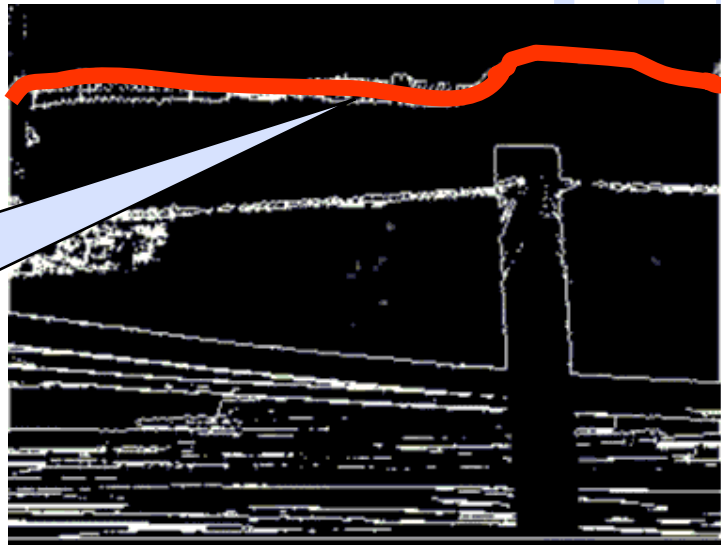


**Cumulated Silhouette Image (CSI)**

# Extend CSI to find occluder boundary

- No need to accumulate pixels inside a foreground silhouette (Foreground boundary is enough)
- Delete pixels inside foreground region in other frames
- Only a portion of the foreground silhouette boundary is effective—Effective Boundary

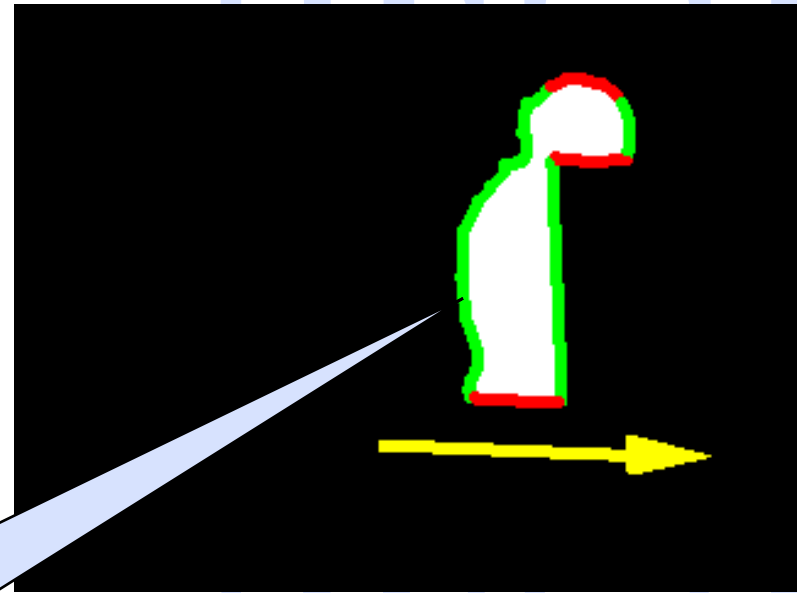
Region above which foreground object never reached—the **Phantom Boundary**





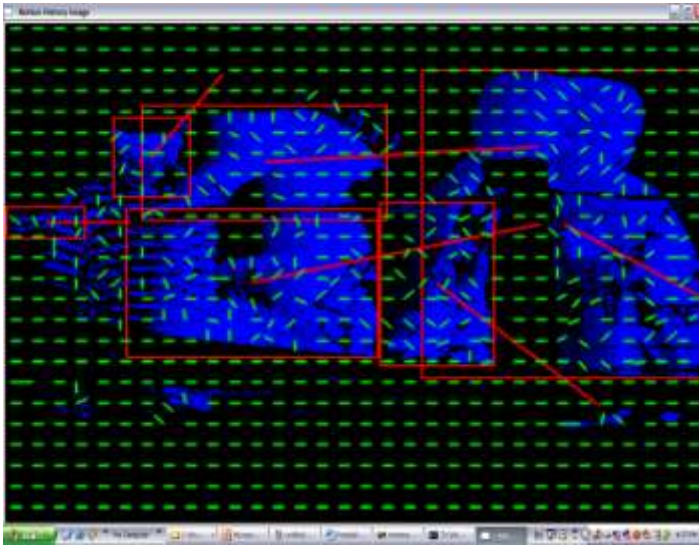
# Effective Boundary

- The *frontal parts* of the moving object silhouette in both the motion and its reverse direction

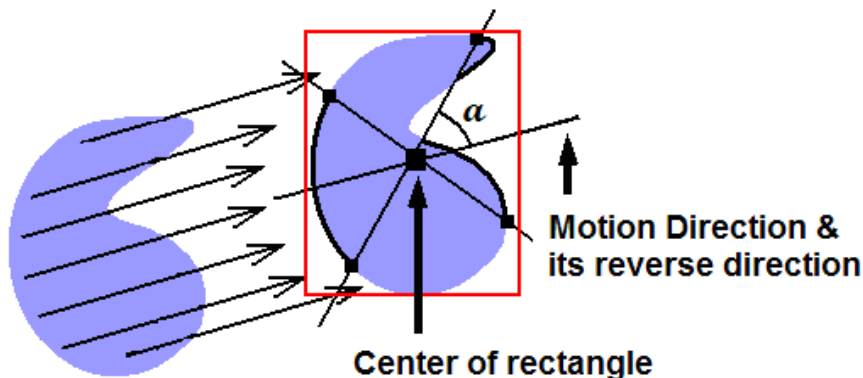


Orthogonal to the  
motion direction

# Our Approach



- **Silhouette Extraction: Using Blob (Statistical Background Subtraction)**
  - Background model
  - Shadow model
- **Motion Direction: Using Motion History Image (MHI)**
  - An image whose pixel intensity is a function of the recency of motion in a sequence.
  - Encode temporal information
  - Determine moving direction of the blob



# Result



**Cumulated Silhouette Image  
(CSI)**



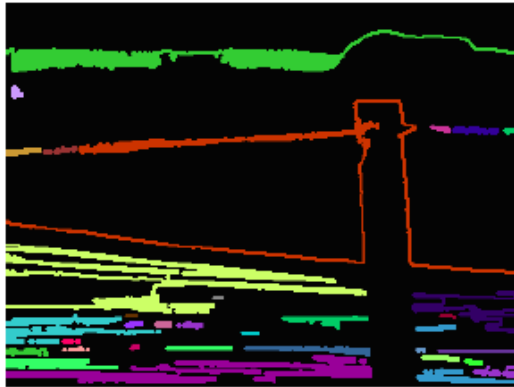
**Cumulated Effective Boundary Image  
(CEBI, after Frame No. 274)**

# Refinement

- **CSI: smooth, but cannot distinguish Phantom from real occluder boundary**
- **CEBI: real boundary of the occluder, but not smooth**

# Refinement

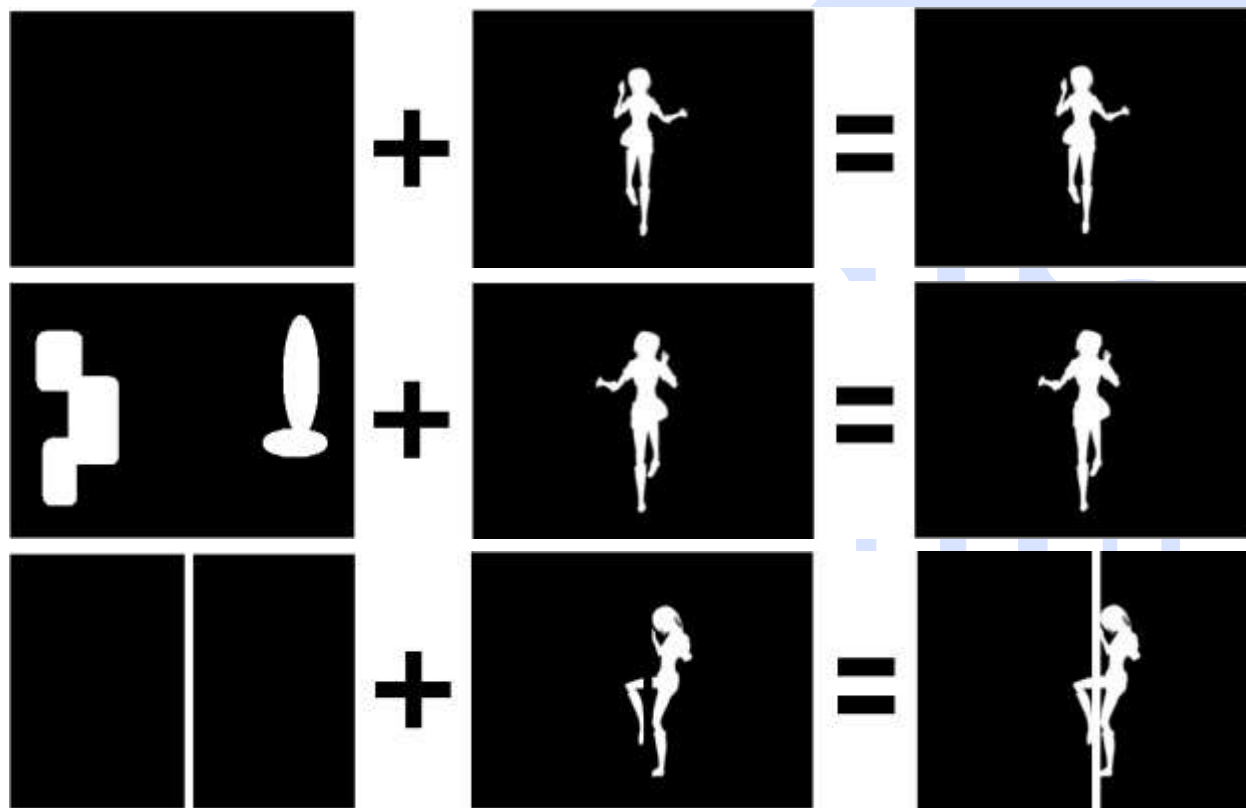
- Use **CEBI** boundary pixels to vote for boundaries in **CSI**



- **Floodfilling**



# Extended Silhouette



$$S_k^E = \begin{cases} S_k^P & \text{when } \partial S_k^P \cap \partial S_k^O = \emptyset \\ S_k^P \cup S_k^O & \text{otherwise} \end{cases}$$

# Algorithm

---

## Algorithm 1 General Visual Hull Construction Algorithm

---

- 1: Occlusion mask extraction for every camera view;
  - 2: Camera calibration;
  - 3: To form “extended silhouette”  $S_k^E$  of  $P$  in view  $k$ ,  $k \in [1, n]$ ,  $n$  is the number of camera views, according to silhouette boundary overlapping;
  - 4: To construct visual hull  $H_{with\ occluder}$  with all extended silhouettes, *i.e.*  $H_{with\ occluder} = \bigcap C_k(S_k^E)$ ,  $k \in [1, n]$ .
-

# Is the visual hull correct?

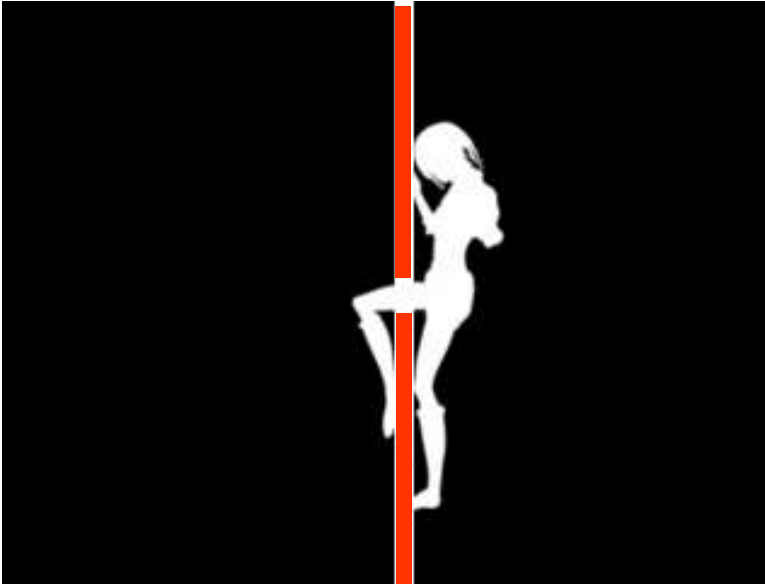
Four-view case, one view with occlusion.



$$H^4_{\text{traditional}} \subseteq H^4_{\text{occlusion-free}} \subseteq H^4_{\text{with occluder}} \subseteq H^3_{\text{without occluder}}$$

**The real object is in our visual hull.  
It is better that simply discarded the occluded views.**

# Minimum Visual Hull



$H^4$   
with occluder



$H^3$   
without occluder

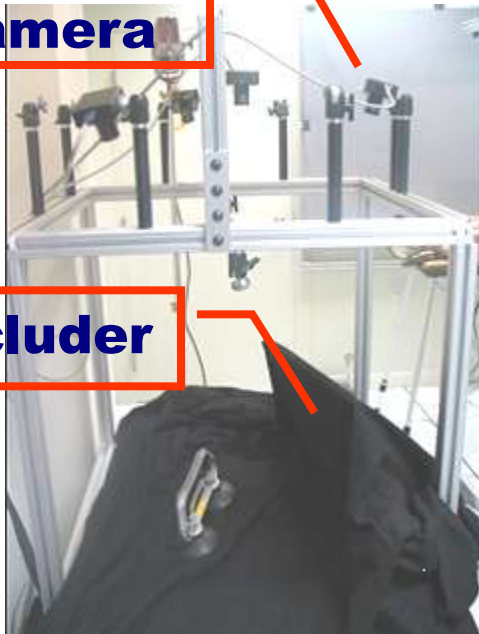


$H^4$   
with occluder



# Example 1. Lab Data

Occluded Camera



Occluder

4 PointGrey DragonFly  
640x480 color cameras  
at 15 fps



(a) *view 1*



(b) *view 2*



(c) *view 3*



(d) *view 4*



(e) *silhouette 1*



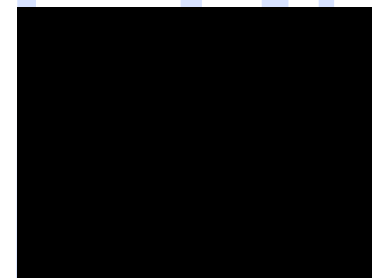
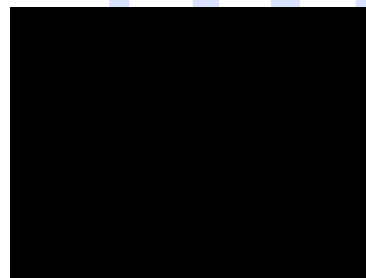
(f) *silhouette 2*



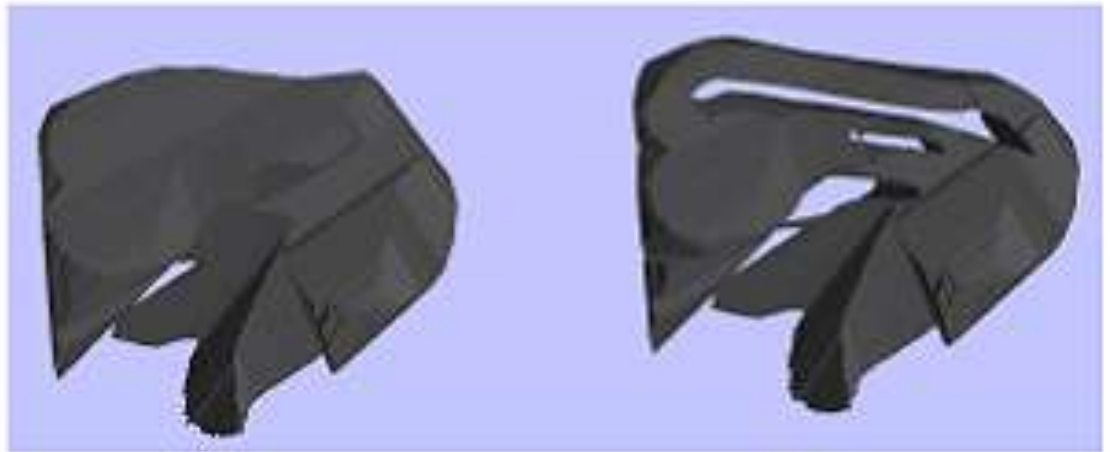
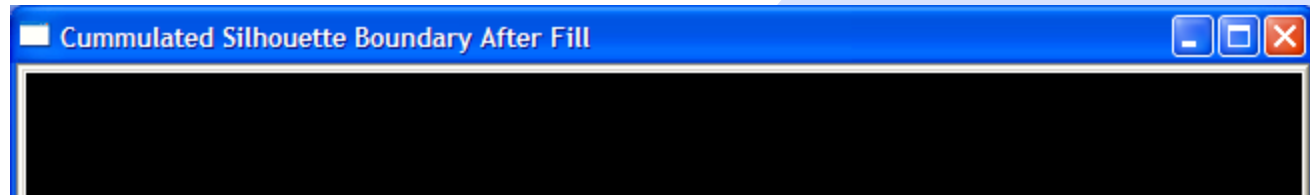
(g) *silhouette 3*



(h) *silhouette 4*



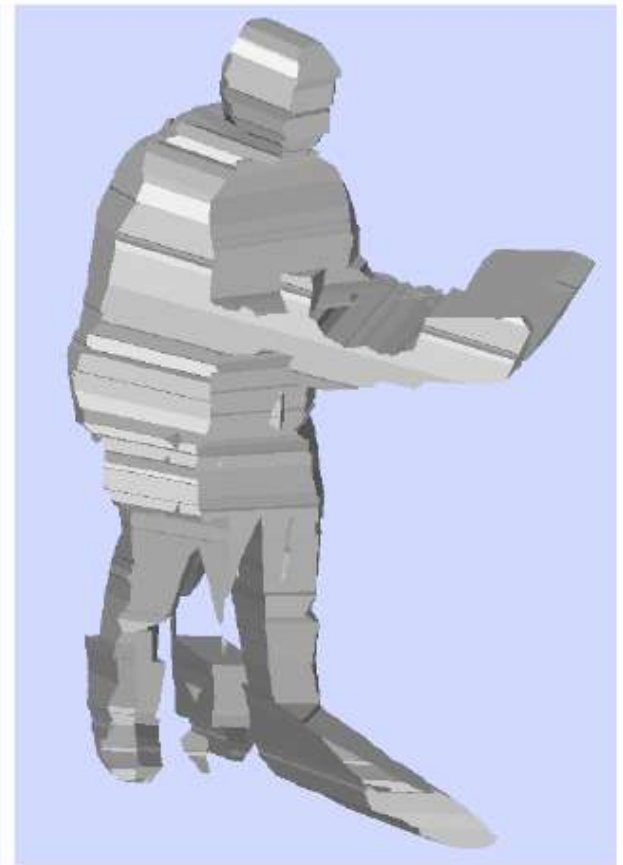
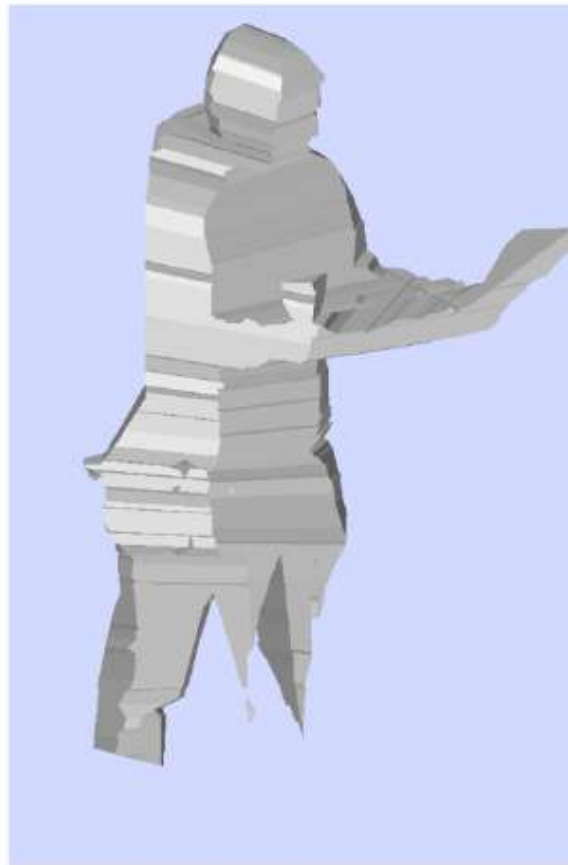
# Example 1. Lab Data (cont.)



Extended silhouette  
our result

$H^{n-1}$  v.s.

# Example 2. Real Office Data



Original method  
result

Our

# Summary

- Contribution:
  - **Extended Silhouette** for visual hull at occlusions
  - **Effective Boundary** for robust occluder mask extraction
  - **Minimum visual hull** in the presence of occlusion.
- Limitation & Future work:
  - Static occluder as a single layer closest to the camera
  - The occluder boundary extraction algorithm

**Thank you, any questions?**