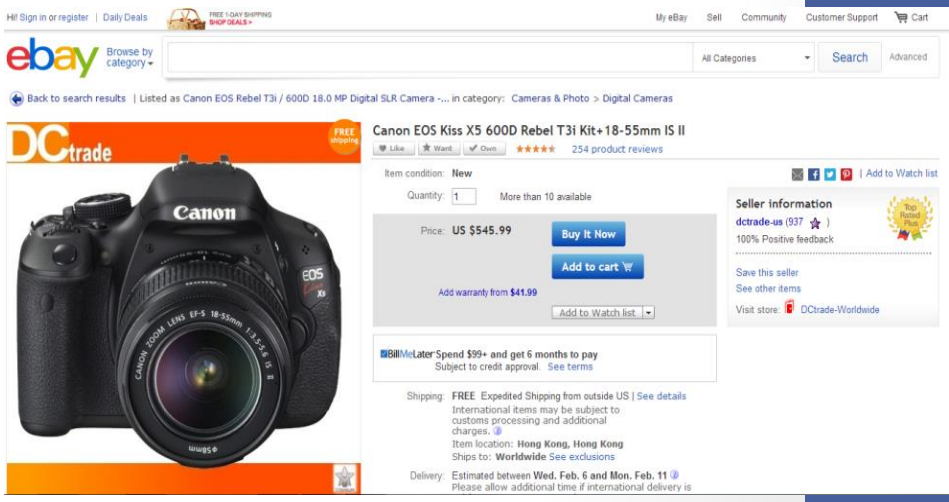
A consumer level 3D object scanning device using Kinect for web-based C2C business

Geoffrey Poon, Yu Yin Yeung and Wai-Man Pang

Caritas Institute of Higher Education

Introduction

- Internet shopping is popular
- C2C / auction websites
  - E-bay and Taobao
- Existing C2C sites
  - Textural
  - 2D still images



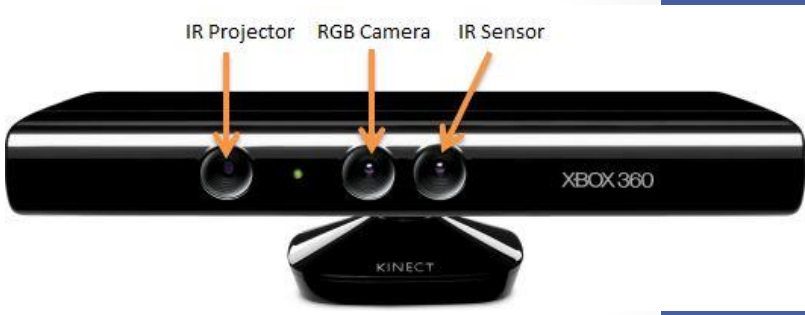
E.g. An auction of a camera
Several pictures are acquired for the camera

Project Goal

- Use latest 3D technology
  - Understand better the condition of auction product
- Problems
  - Affordable solution of 3D content creation
  - 3D visualization of product on web-based environment

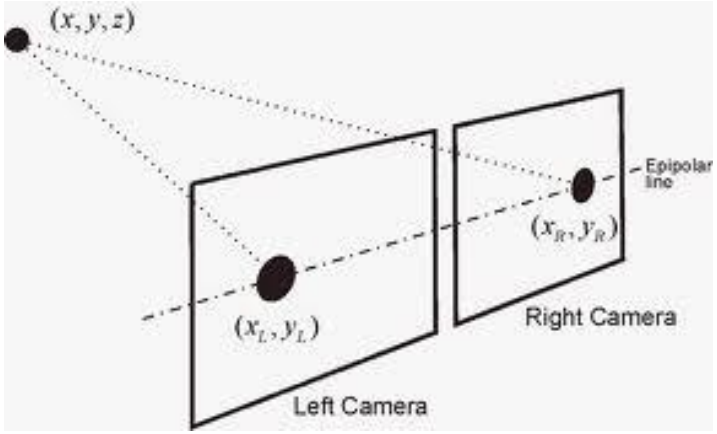
Project Goal

- 3D scanner with low cost camera
- Kinect
  - Around USD$150
  - Provide depth data robustly
- Fast and simple scanning steps
- Automatic 3D reconstruction



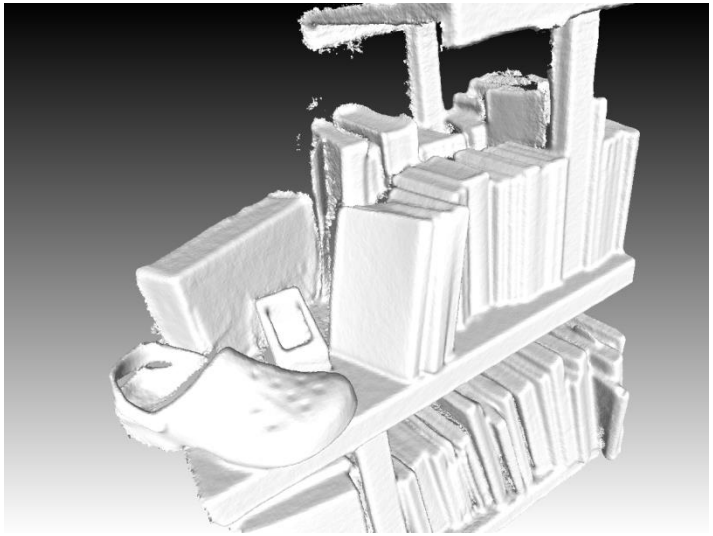
3D Reconstruction System

- To create 3D model from set of 2D images from the real world
- A common approach:
  - Reconstructed from stereo views
  - Similar to how human eyes and brain works



3D Reconstruction using Kinect

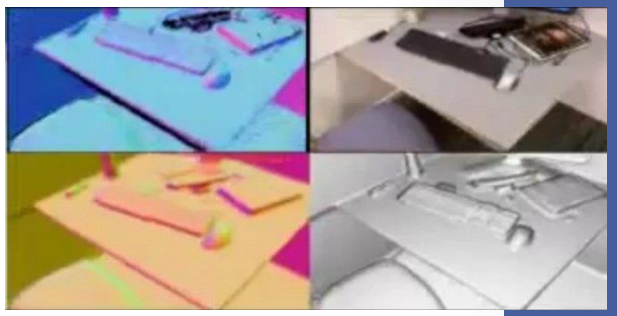
- ReconstructME



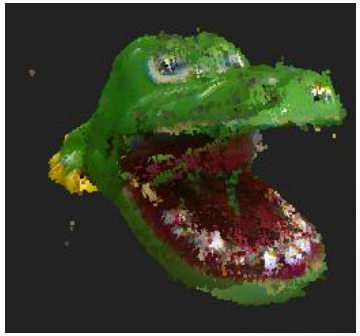
  - 3D model without texture;
  - Caused the capture lost easily.

3D Reconstruction using Kinect

- KinectFusion (Microsoft)
  - Colored 3D model from reconstruction;
  - Depends heavily on GPU



- 3D Scan 2.0
  - Lower quality Colored 3D model from reconstruction;
  - Support linux only.

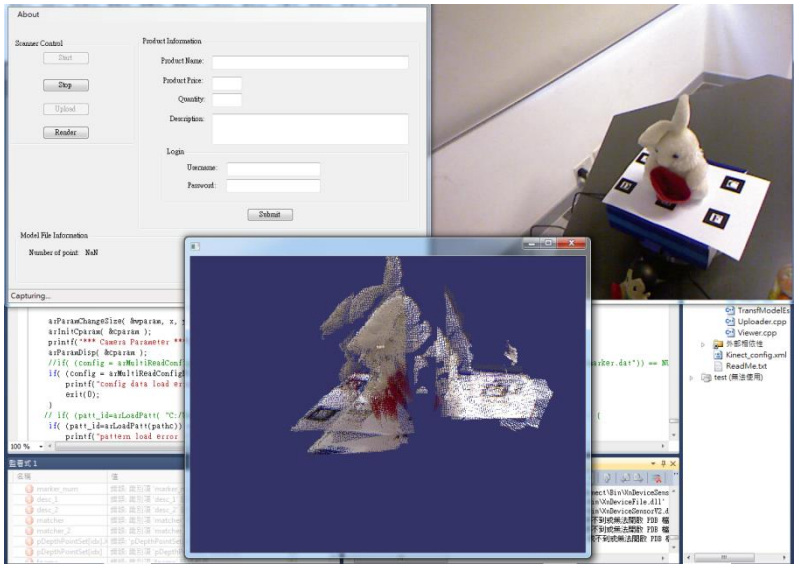


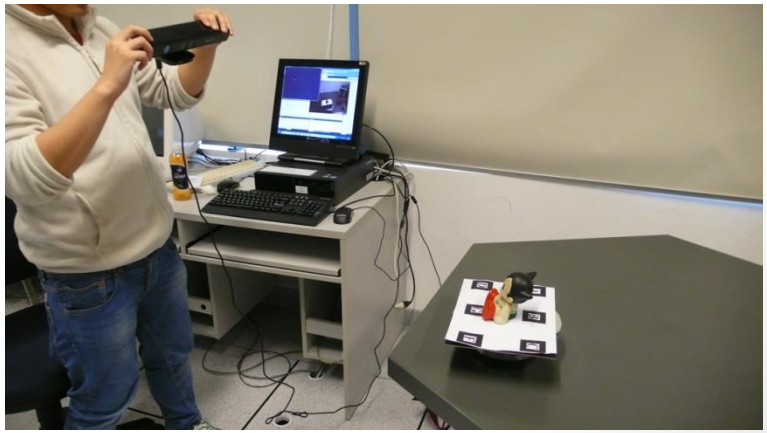
Overview of Our Approach

- Three major steps in 3D model creation
  - Capturing
  - Point Cloud Processing
    - Background removal & orientation matching
  - Point Cloud Registration

Model Capturing

- OpenNI SDK to obtain RGBD-data from Kinect
  - Point clouds are formed from RGBD-data
- Multiple frames from different views





Post-processing

- Our first problem:

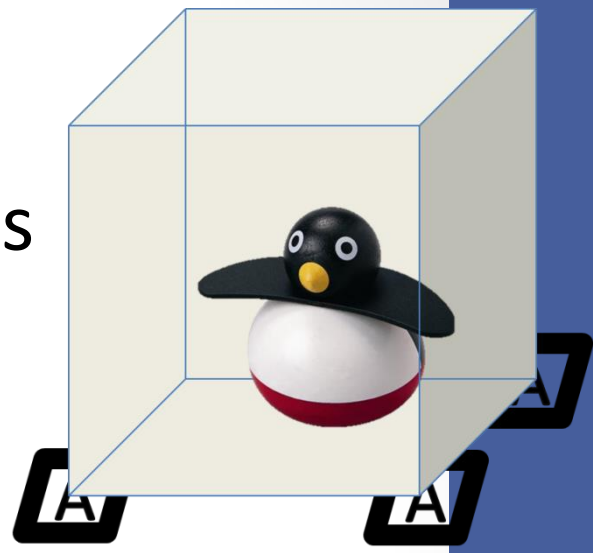
How to remove the background/noise ?

Post-processing

- Second problem:

How to align point clouds captured at different view point ?

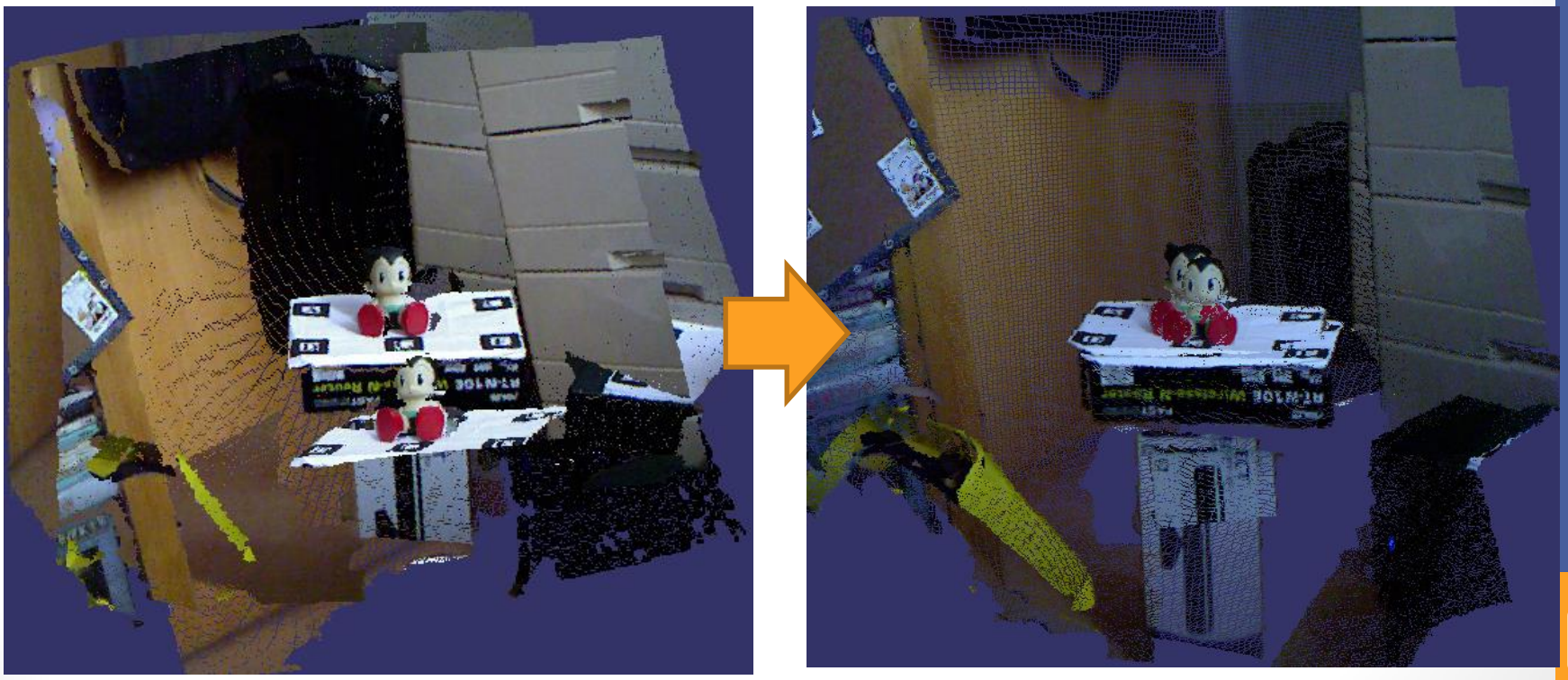
Processing Point Cloud : Views Alignment



- To align points from different views
  - Marker-based detection
    - Fast response
  - ARToolKit is used
    - Markers is extracted from image captured
    - Estimate orientations from different views
    - Match the orientations of from different views

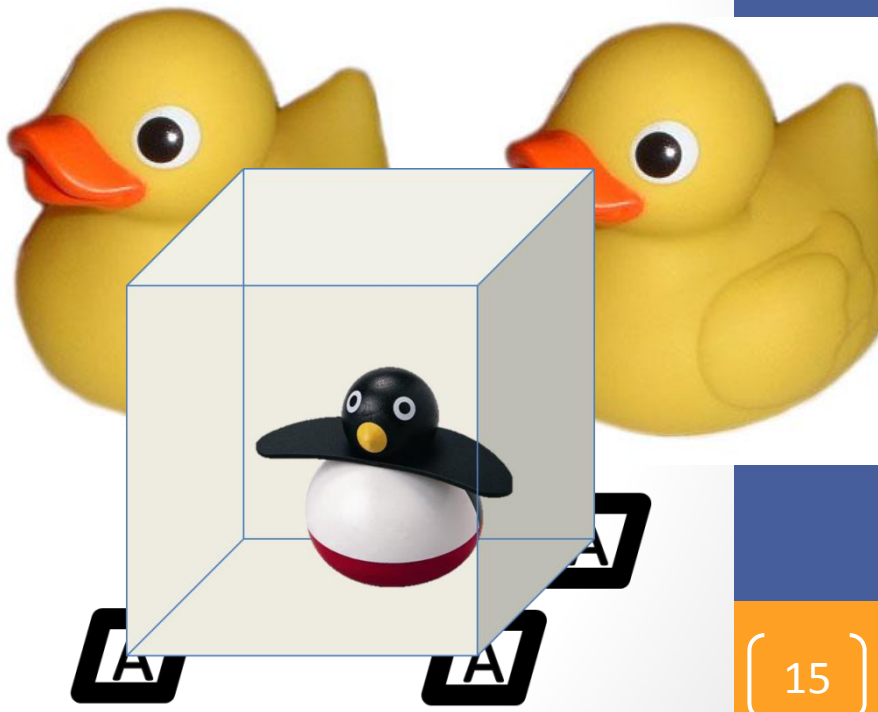
Processing Point Cloud : Views Alignment

- Result after alignment using markers



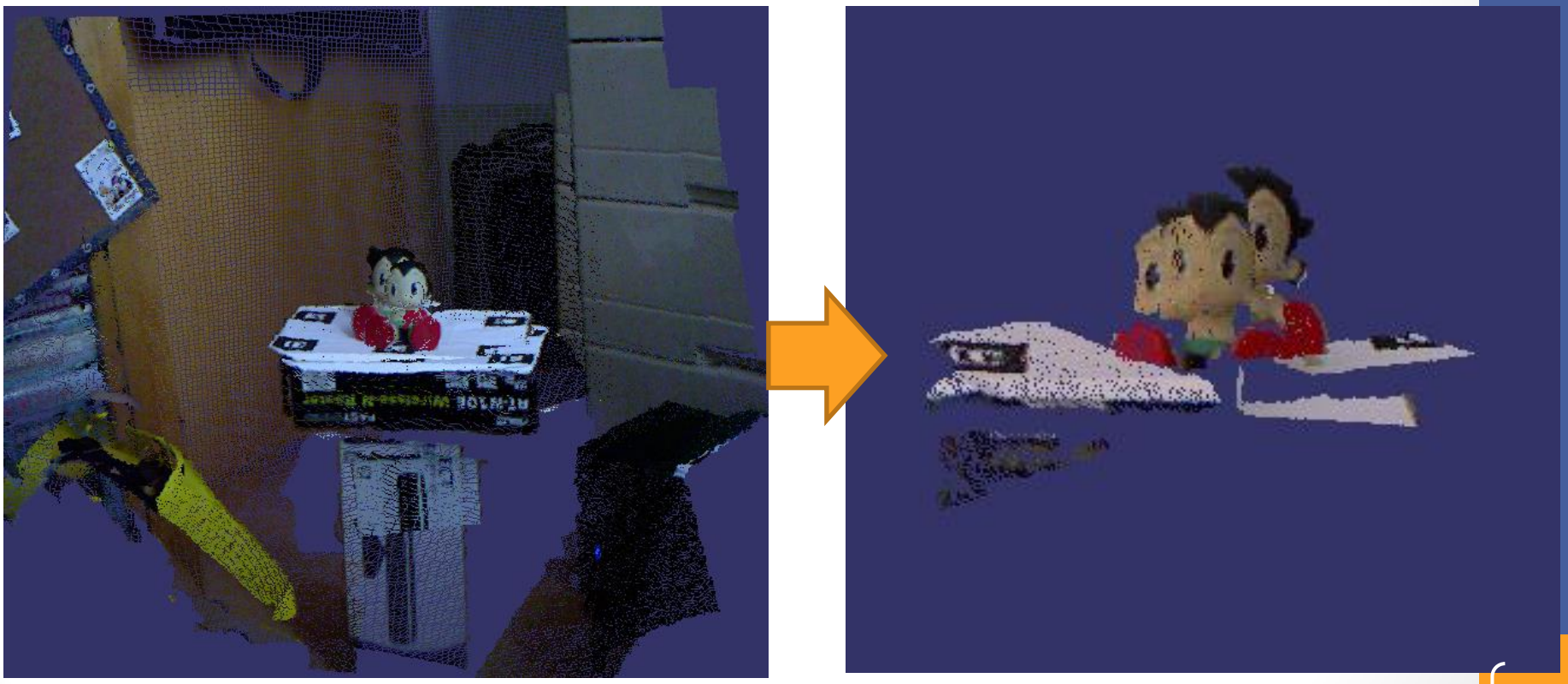
Processing Point Cloud: Background Removal

- Markers are used to define interested region
- Points outside are removed
- Left points within the markers.
- 90% of background can be removed



Processing Point Cloud: Background Removal

- Unwanted background is removed

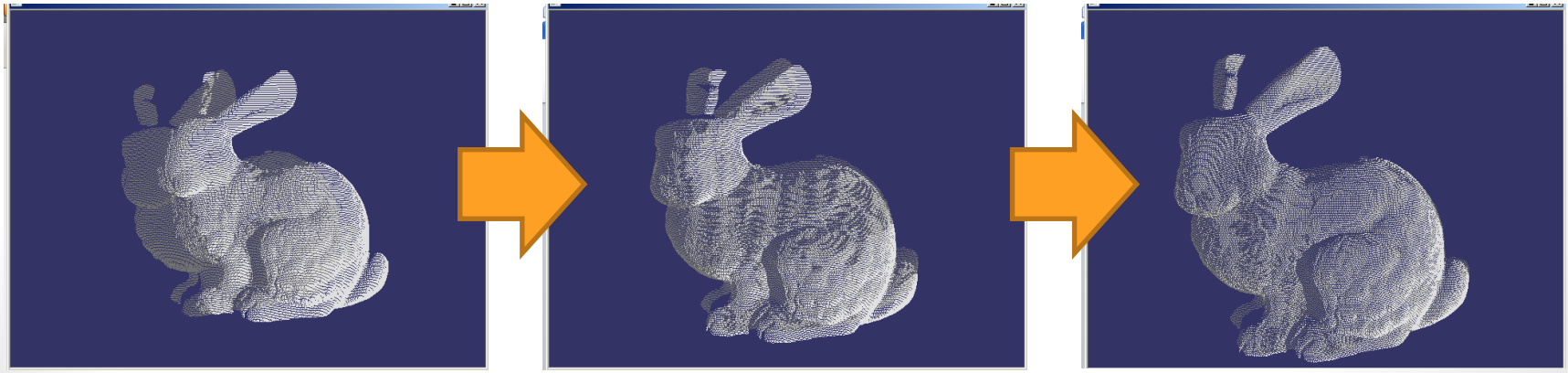


Processing Point Cloud

- However, a closer look…
- View alignment is still not satisfactory !
  - Large error exists in marker-based detection

Point Cloud Registration

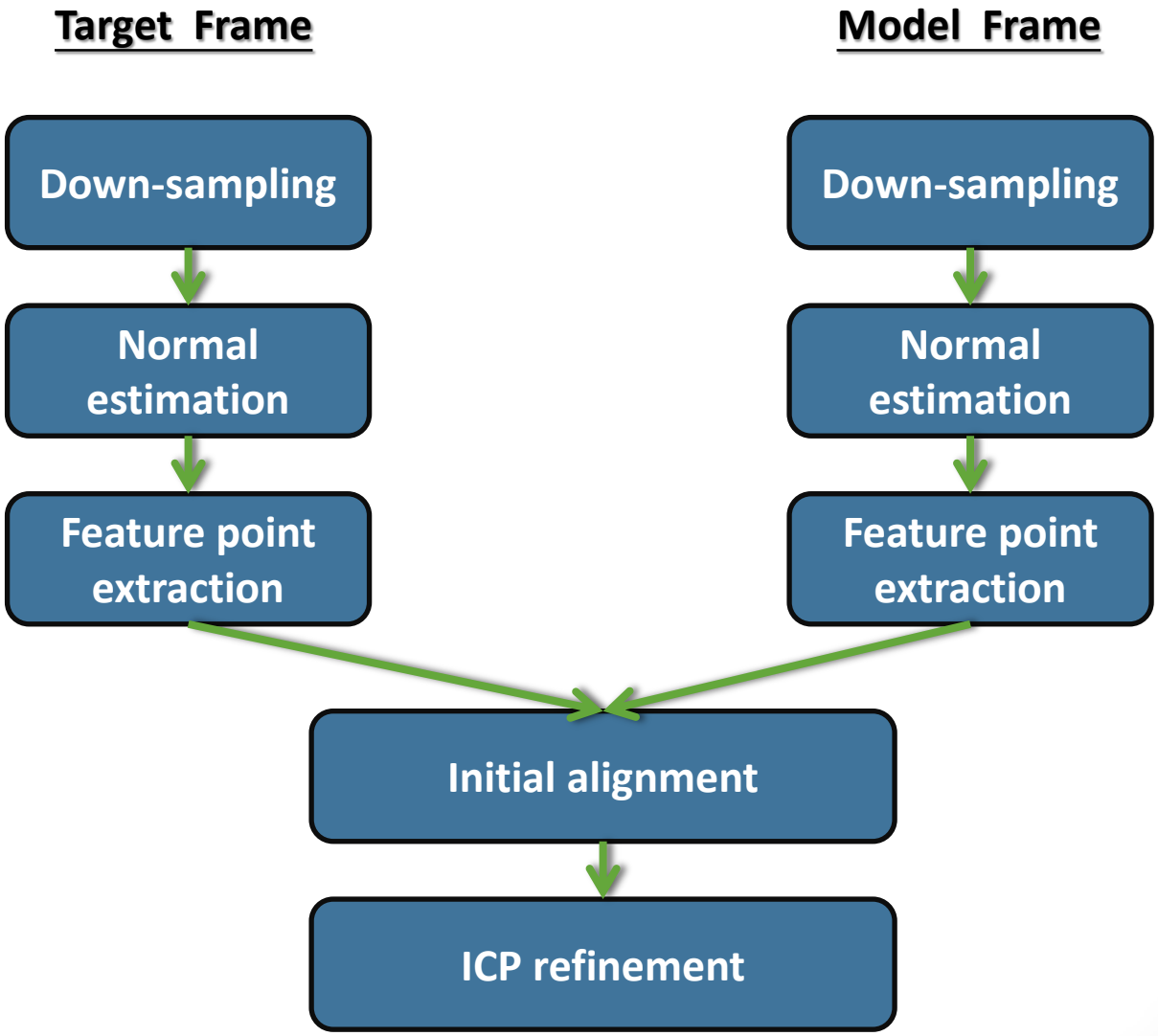
- Refine alignment of different views
- ICP (Iterative Closest Point)
  - Initial alignment
    - Transformation between frames
  - Iterative process



Point Cloud Registration

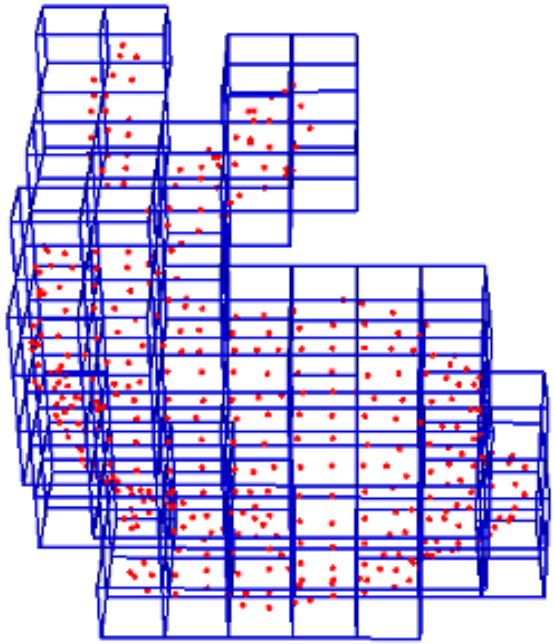
- Pure ICP is slow
- Reduce number of points
  - Down-sampling using voxel-grid
- Create a better initial alignment
  - Matching of feature points

Point Cloud Registration



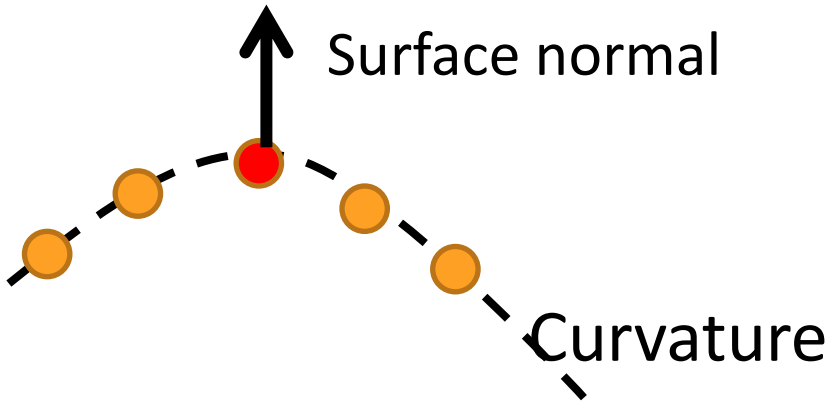
Down-sampling

- First step in the point cloud registration
  - Improve speed on point cloud alignment
  - Obtain the better features value on huge point cloud
- Voxel-grid down-sampling
  - Dividing the point cloud into the grids
  - Use centroid of points in each grid as the sample



Computation of Initial Alignment

- To extract key points
  - Base on large curvature value
  - Estimate using Principal Component Analysis (PCA)
    - involves neighboring points

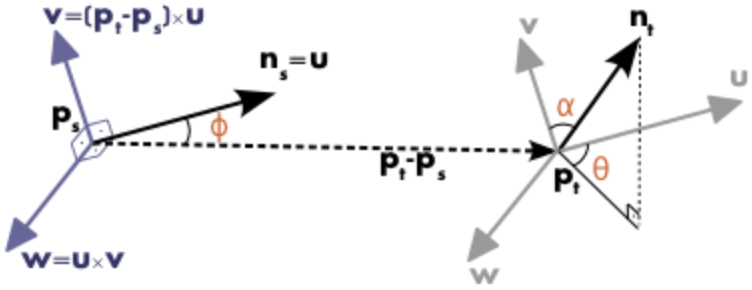


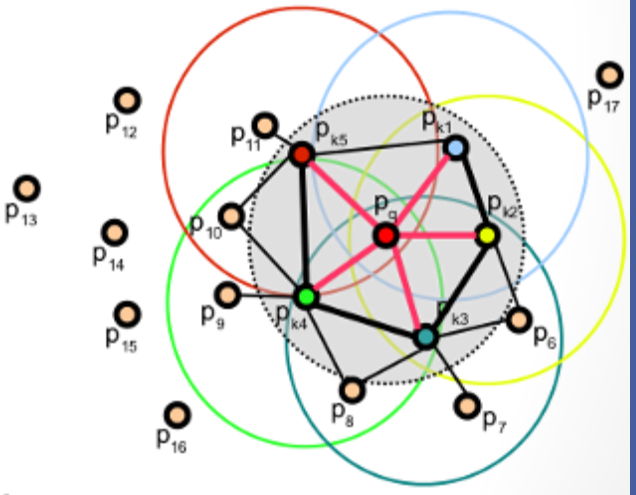
Computation of Initial Alignment

- Matching the corresponding feature key points between 2 frames
  - Fast Point Feature Histogram (FPFH) as the feature descriptor
  - KdTree(KNN search implementation with OpenCV)
- FPFH
  - 3 angles $\alpha,\emptyset,\theta$ and distance d

$$\alpha = v \cdot n_t \qquad \emptyset = u \cdot \frac{(p_t - p_s)}{d}$$

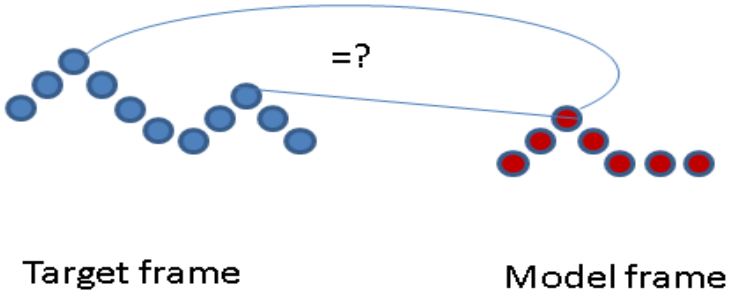
$$\theta = \arctan(w \cdot n_t, u \cdot n_t) \qquad d = \|P_t - P_s\|_2$$





Computation of Initial Alignment

- Estimated the transformation between key pairs from 2 frames
  - Incorrect point pairs may be form by taking the 1 nearest point.



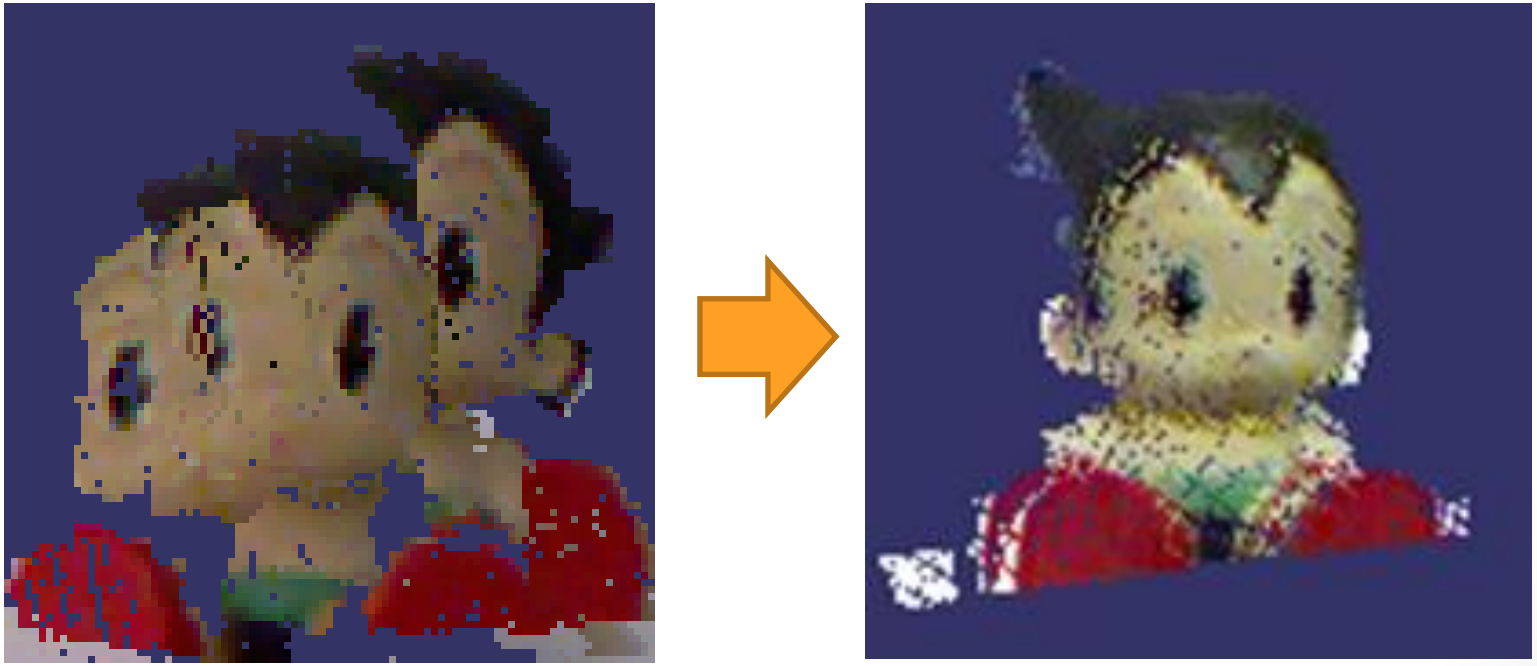
  - Taking the best samples using RANSAC
    - Take 3 sample points for each iteration,
    - Form the point pairs by Kdtree,
    - Estimate the Transformation by SVD and apply it,
    - The 3 pairs with the minimum error is the best model.
  - Compute the Transformation on best model by SVD

Point Cloud Registration

- Refine the alignment
  - Iterative Closet Point (ICP)
    - Extract the good key points from initial guess
    - Matching the corresponding key points between 2 frames (using XYZ)
    - Estimated the transformation between key pairs from 2 frames
      - Outlier Removal using RANSAC
      - SVD
    - Iterative until reach the acceptable error

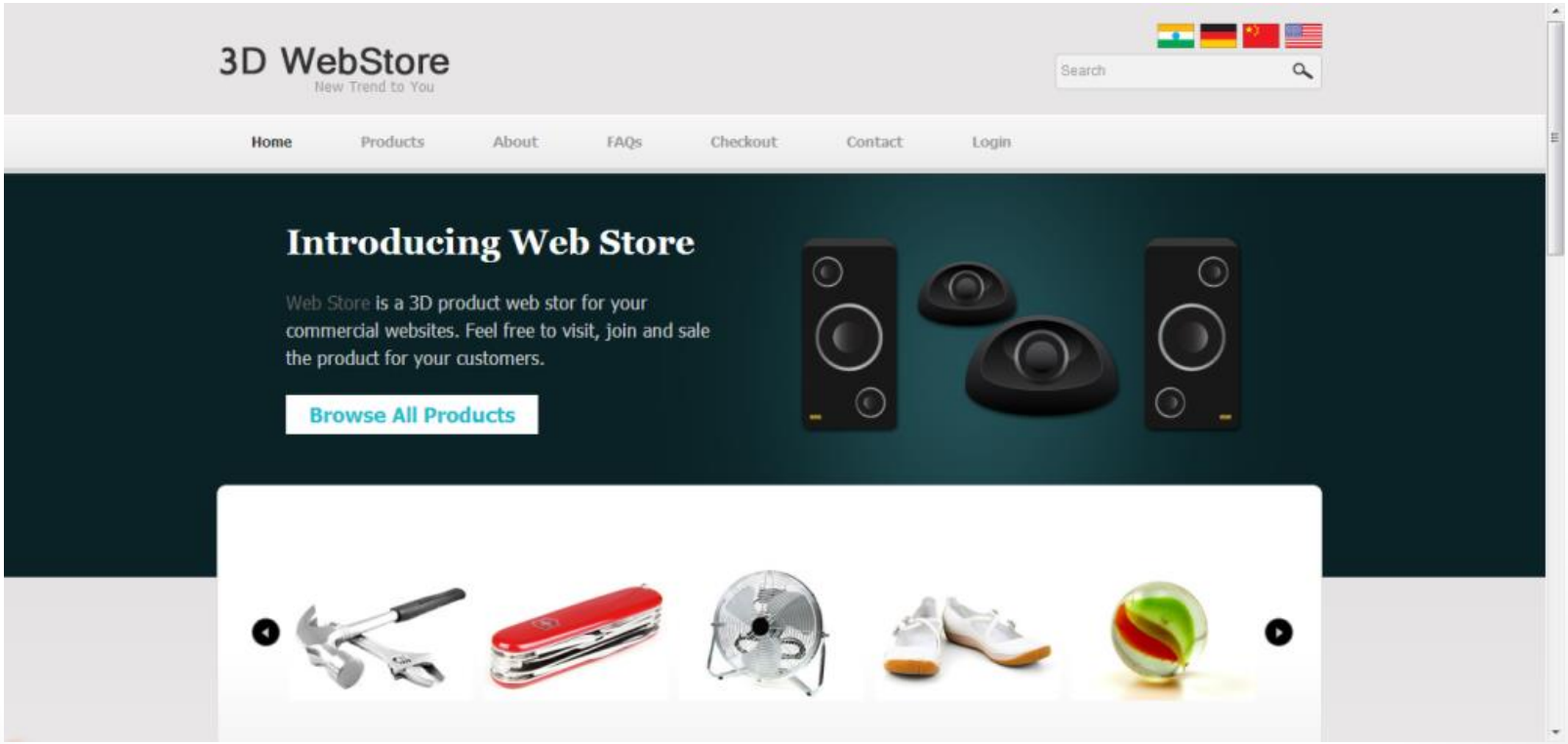
Point Cloud Registration

- The alignment is refined after using point cloud registration



3D Point Cloud

- After the registration, the 3D point cloud model can be upload to web host by using cURL



3D Point Cloud Format

.ASC files for point cloud storage

For example:

123.000	534.123	534.143	255	255	123	One point
54.000	67.123	12.143	10	20	30	
54.000	67.123	12.143	10	20	30	
……						

(X, Y, Z and R, G, B data)

Rendering point cloud model

- WebGL
  - JavaScript API based on Open GL ES 2.0
  - Web browser without any plugins
  - XB Point Stream WebGL

Result

Movie of Capturing Process

Result : 3D Scanning

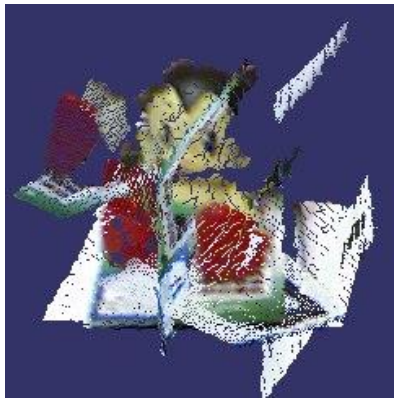
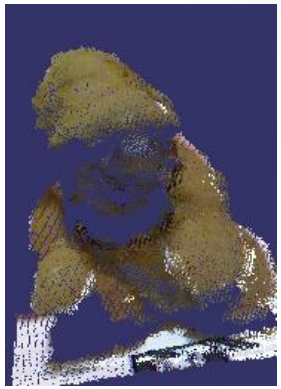
Test Model- Astroboy			
Number of total Points	247254	10089	10089
Size	9.34MB	387KB	424KB
Number of frame	1	3	3
Match	No	No	Yes
Noise/background	More	less	less

Result : 3D Scanned Products

			
Number of total Points	98024	44560	10089
Number of frames	8	6	4
Time*	3mins	1.5mins	57sec
Number of Iteration	560	390	160

***The testing is performed with an Intel i5-3.4GHz CPU**

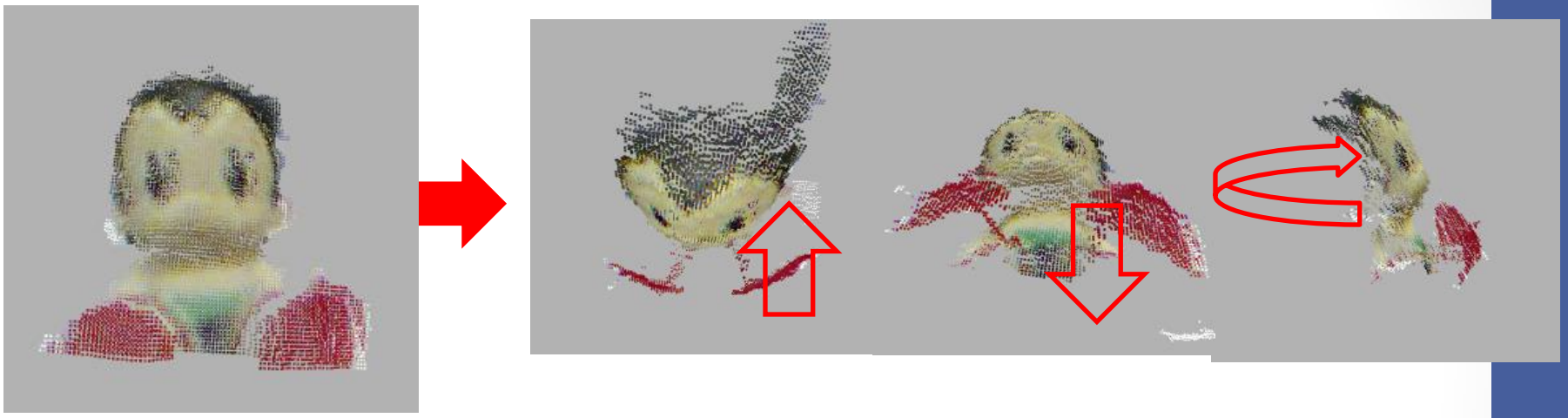
Result : Reconstruction Accuracy

			
Environment	Symmetric	Non-symmetric	Under strong light intensity
Minimized Error	0.002204m	0.002143m	0.003506m

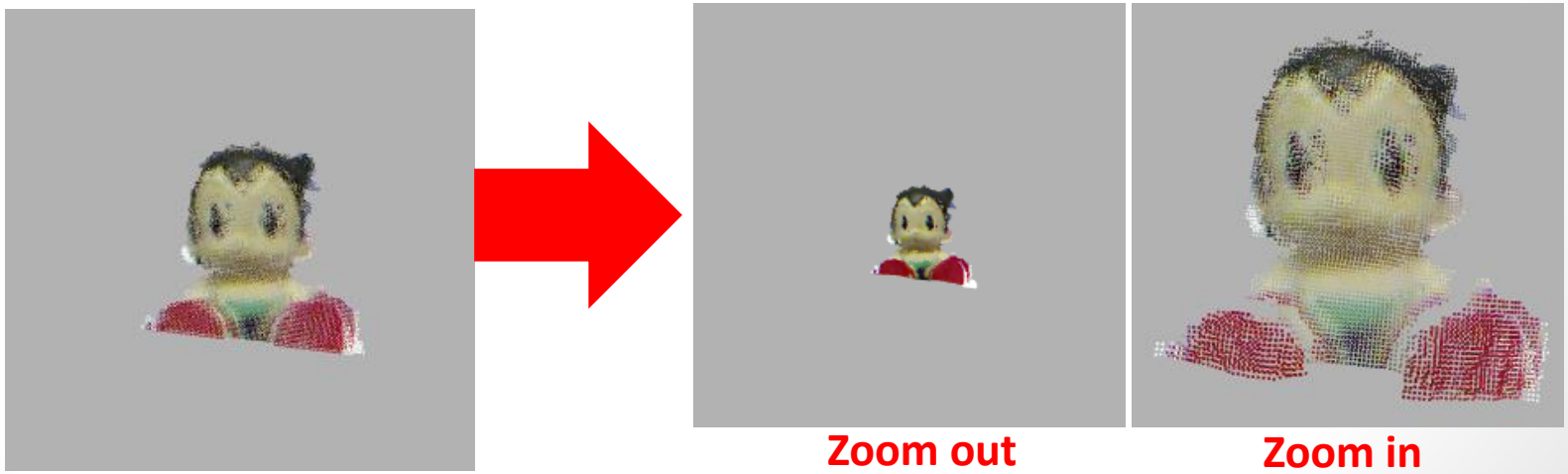
- **Symmetric (Shape containing similarity)**
  - Limitation on the curvature evaluation by PCA
- **Under strong light intensity**
  - infrared sensor will be affected by strong light confuse to the infrared reflection
- **Non-symmetric**
  - reconstructed better 3D feature and more matched points

Result : Control of views in browser

Mouse move to control rotation



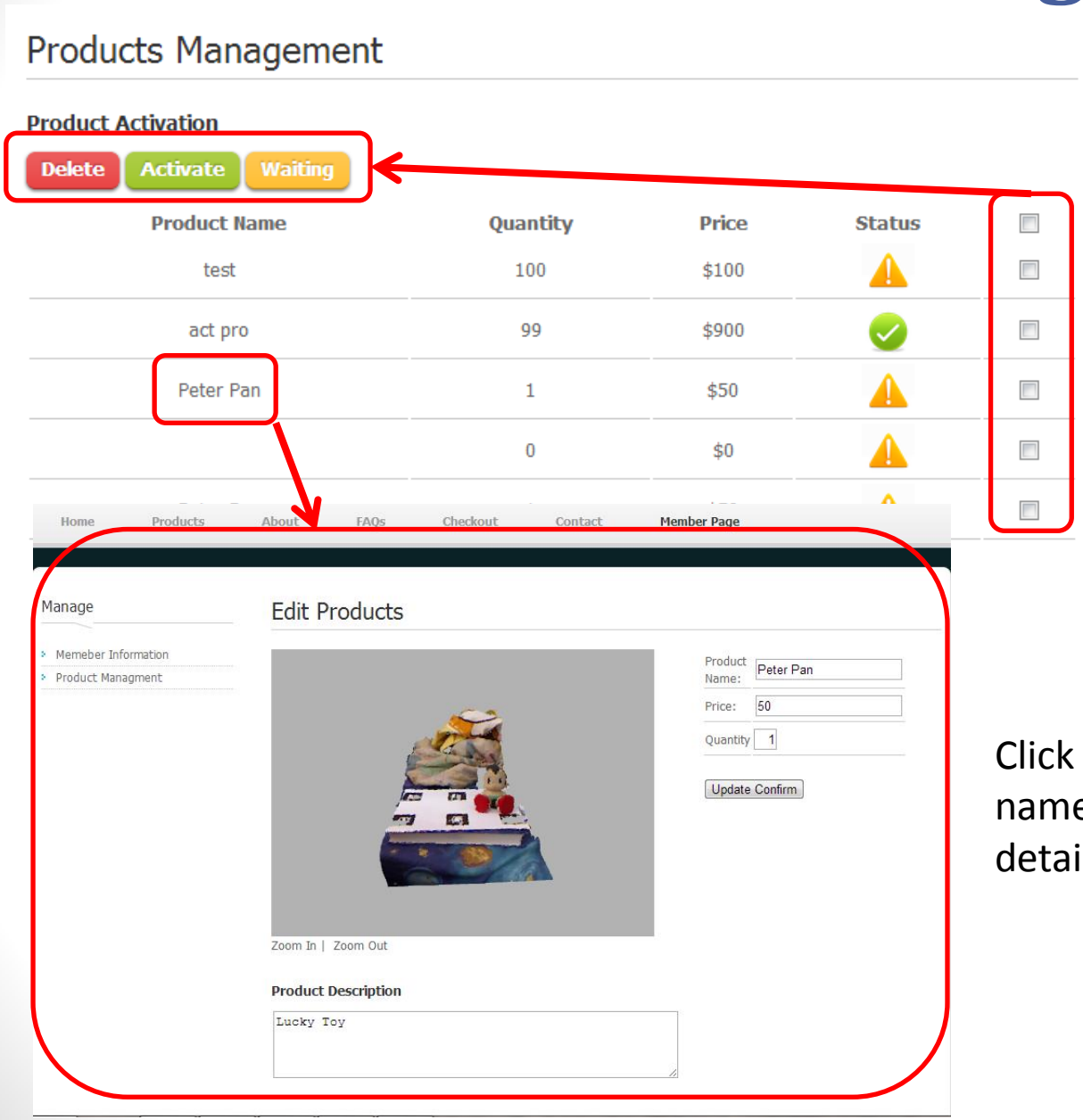
Mouse scroll to control zooming



Zoom out

Zoom in

Result : Products Management



Activate the product status for public

Click the product name for edit product detail

Limitation

- The target object size is limited
  - Corresponding to AR marker size
- The target object cannot be transparent or translucency object
- The target scan distance is limited
  - The range about 0.6meter to 3 meter
- The Kinect resolution low quality
  - The large resolution only 640 * 480

Future Works

- Optimize the current algorithm to reduce post-processing time
- Increase matching accuracy
- Using ECE(Euclidean Cluster Extraction) for background noise remove
- Compress the 3D file size for increase store using

Conclusion

- Low-cost 3D reconstruction system
  - Using Kinect costs just USD$150
  - Point with color
  - Simple object extraction
- Web-enabled 3D rendering of product
  - 3D rendering of point cloud in web browsers
- A comprehensive system
  - Easy to use 3D content creation
  - New 3D experience for online shoppers

Thank you