Visual Object Perception in Unstructured Environments

Changhyun Choi
Robotics Ph.D. Program
Interactive Computing, College of Computing
Georgia Institute of Technology

Prof. Henrik I. Christensen (Advisor)
Prof. James M. Rehg
Prof. Irfan Essa
Interactive Computing
College of Computing
Georgia Institute of Technology

Prof. Anthony Yezzi
Electrical and Computer Engineering
Georgia Institute of Technology

Prof. Dieter Fox
Computer Science and Engineering
University of Washington
Outline

- Introduction
- State of the art
- Remaining challenges
- Motivations
- Thesis statement
- Approaches
- Conclusions & Future work
Introduction

- Pick-and-place task
- Robots moving from *controlled* settings to *unstructured* environments
- **Robust** object perception is crucial
Problem Formulation

3D Object Models

Camera → Image → Pose Estimation & Tracking → Object ID & Pose
Early Object Perception

Convert 2D photo to 3D model
- perspective projection
- edge detection, line fitting
- 6-DOF transform
[Roberts 65]

**Categorical 3D** shape models: GC, geon
- viewpoint invariant
- textureless objects
[Binford 71, Brooks 83, Biederman 85, Dickinson et al. 92, ...]

**Exact 3D** shape models: polyhedron or CAD
- viewpoint invariant
- textureless objects
[Lowe 87, Thompson and Mundy 87, Huttenlocher and Ullman 90, ...]

**Exemplar 2D** appearance model: 2D templates
- viewpoint dependent
- textured objects
[Murase and Nayar 95, Ohba and Ikeuchi 97, Black and Jepson 98, ...]

**Categorical 2D** appearance model:
- spatial models with local appearance features
- viewpoint dependent
- textured objects with clutter and occlusion
[Lowe 99, Mikolajczyk and Schmid 04, Fei Fei et al. 06, Fergus et al. 07, ...]
State of the Art

[Collet et al., IJRR’11]
- SiftGPU feature
- Sparse 3D keypoint models
- Iterative Clustering
- Require well textured objects
- Cannot handle textureless objects

[Hinterstoisser et al., PAMI’11]
- Template Matching
- Combine image gradients and surface normals
- Can handle untextured objects
- Require large amount of templates (e.g. 2000)
- Coarse pose estimation
- Produce jitter noises in pose estimates

[Aldoma et al., ICCV workshop’11]
- Table-top assumption
- Object segmentation + CVFH/kernel descriptors
- Require planar background
- Hard in cluttered environment

[Lai et al., AAAI’11]

[Klein et al., BMVC’06]
- Particle Filter
- Arbitrary shaped object
- Require a given starting pose
- Do not address challenging cases
Remained Challenges

1. Object with and without Textures
2. Background Clutter
3. Object Discontinuities
4. Real-time Constraints
Challenge 1: Texture

- Textured objects
  - Photometric: color, keypoints, edges or textures from surfaces
- Textureless objects
  - Geometric: point coordinates, surface normals, depth discontinuities

Handling both *textured* and *textureless* objects
Challenge 2: Clutter

Controlled environments

Unstructured environments

Difficulties = Degree of Clutter

- False measurements
- False pose estimates
- Stuck in local minima
- No table-top assumption
Challenge 3: Discontinuities

- Ideal vs Reality
  - Occluded by other objects, human, or robots
  - Object goes out of the camera’s field of view
  - Blurred in images
- Re-initialization problem
Challenge 4: Real-time

- Constrained by timing limitations
- Scarcely see real-time state-of-the-art
Definition and Scope 1

Model-based Visual Object Perception in Unstructured Environments

- 3D mesh models
- Object instance recognition
- 6-DOF pose estimation and tracking
- Monocular or RGB-D
- Cluttered & obj. discontinuities
Definition and Scope 2

Visual features: Photometric & Geometric

- Photometric image formation in 2D
- Intensity, color, edges from texture, keypoint descriptors, ...
- 3D geometric shapes
- Depth points, edges from geometric shapes, line segments, planes, normals, ...

photometric image formation in 2D
intensity, color, edges from texture, keypoint descriptors, ...
3D geometric shapes
depth points, edges from geometric shapes, line segments, planes, normals, ...
Motivations

Known 3D object model was strong assumption

3D object models have been accumulated on the Internet!
Motivations

Google 3D warehouse
(about 2.5 million models)
3D modeling will be a **trivial** task with **Kinect**!
Motivations

Depth sensors are everywhere!

24 million Kinects sold

Occipital, Inc

Google Project Tango

Apple + PrimeSense
3D modeling will be a trivial task even with a mobile phone!
Motivations

• Promising in Robotics
  • exist in 3D space
  • interact with 3D world
  • 3D data is significant information for robots

• Advantages
  • Foreground object segmentation is trivial
  • Employ various geometric features from (3D models and 3D scene depth)
Thesis Statement

• To close the loop between the geometric era of early computer vision and the currently dominating appearance age, both photometric and geometric features need to be considered.

• The combination of these features enables object perception algorithms not only to be more effective but also to handle an increased spectrum of objects.

• Two theoretical frameworks using multiple pose hypotheses based on combined features are contributed in this thesis.

• These new frameworks are robust to significant clutter and occlusions, and are therefore efficacious solutions for visual object perception in unstructured environments.
Approaches

- **2D Visual Information (Monocular Camera)**
  - Combining Keypoint and Edge Features
    [ICRA'10, ICRA'11, IJRR'12]
  - Extending to Textureless Objects
    [IROS'12]

- **3D Visual Information (RGB-D Camera)**
  - Voting-based Pose Estimation using Pair Features
    [ICRA'12, IROS'12]
  - Object Pose Tracking
    [IROS'13]
Related Work

• **Edge-based** approaches
  - Cheap to extract edges (real-time)
  - Applicable to textureless objects
  - Not distinctive enough
  - Might be stuck in local minima

• **Keypoint-based** approaches
  - Good for initialization
  - Invariant to scale and rotation
  - Only applicable to textured objects
  - Computationally expensive

**Complementary Combining**

[2D Monocular > Combining Keypoint and Edge Features [ICRA’10, ICRA’11, IJRR’12]]

[Harris, 92] [Drummond, PAMI’02]

[Lowe, IJCV’04] [Gordon, 06] [Collet, IJRR’11]
Overview

For several decades, methods which employ natural features have been proposed. Researchers have focused on tracking using natural features, specifically markers. However, this approach has been regarded as a major limitation. Artificial markers are attached to the object or environment. One of the easiest ways is through the use of fiducial markers.

In the generality of object shapes, salient edges are automatically identified during an offline stage. Dull edges are usually invisible in CAD models or at least manually well designed models. Our system can handle any form of polygon mesh model. To achieve this, we have employed a number of different approaches. Differential methods for pose tracking is presented. A strategy for using keypoint for pose initialization and image which are suitable for robust wide baseline matching are employed.

The main contributions of this paper include:

- While the former models are based on 3D and augmented reality have all addressed this as a model for edge-based tracking, the latter models match with keypoints in an image which can be efficiently computed while the latter models match with keypoints in an image which can be efficiently computed.

- The edge-based tracking sometimes drift because of edge ambiguity. The proposed system monitors the tracking scheme. Since the edge-based tracking requires a manual pose initialization which is inconsistent. Experimental results demonstrate our system's results and occasionally reinitialize when the tracking results are inconsistent. The keypoint-based tracking solution unlike most of the previous ones has the advantage of not requiring any manual initialization and being able to run in real-time.

- The keypoint features are also computationally expensive descriptors which maintain local and partial viewpoint. But the keypoints require relatively computationally cheap to compute and computationally cheap. Since the edge is usually easy to detect in a grayscale image.

- To project the model close to the object by comparing projected CAD model edges to edges less 3D model-based real-time tracking system. It tracks an object recognition and tracking solution, unlike most of the previous systems which are inconsistent. Experimental results demonstrate our system's results and occasionally reinitialize when the tracking results are inconsistent.

- When the track gets off, it is triggered by describing edge-based tracking. The combination of these two complete the initial pose estimation. This pose estimate serves as an initial estimate for edge-based tracking. The keypoint-based tracking is used to maintain 3D features. For an in-depth study of the different template-based methods, we refer the interested reader to the survey.

As robots move from industrial to daily environments, the most important problem robots face is to recognize objects and estimate DOF pose parameters in less-constrained environments. For the last decade, computer vision, robotics, and intelligence areas have employed a number of different approaches.

Contribution of This Paper

- The paper presents the RAPiD style tracking methods have used simplified and more efficient methods provides an efficient and robust tracking solution. The combination of these two complements each other.

- The edge features are easy to detect in a grayscale image. The keypoint features are also computationally expensive descriptors which maintain local and partial viewpoint. But the keypoints require relatively computationally cheap to compute and computationally cheap. Since the edge is usually easy to detect in a grayscale image.

- To project the model close to the object by comparing projected CAD model edges to edges less 3D model-based real-time tracking system. It tracks an object recognition and tracking solution, unlike most of the previous systems which are inconsistent. Experimental results demonstrate our system's results and occasionally reinitialize when the tracking results are inconsistent.

- The keypoint features are also computationally expensive descriptors which maintain local and partial viewpoint. But the keypoints require relatively computationally cheap to compute and computationally cheap. Since the edge is usually easy to detect in a grayscale image.
• Original CAD models are too complex.
• Most edges in CAD do not appear in the real edge image.
• We should simplify in some way.
Salient Edges

\[
I(\text{edge}_i) = \begin{cases} 
1 & \text{if } |\mathbf{n}_1 \cdot \mathbf{n}_2| \leq \tau_s \\
0 & \text{otherwise}
\end{cases}
\]

- Use face normal vectors
- Automatically determine salient edges which are more likely to be visible in images
our approach
edges
model rendering

keypoint only
Limitation

- **Single** pose hypothesis
  - **Wrong** prior pose → not converging to global optimum
- **Ambiguous** edges
- **Stuck in** local minima
  - Highly cluttered environment
  - Occlusions
- **Multiple** pose hypotheses
- **Particle Filtering**
Related Work

Particle Filtering using Edges

[Isard, IJCV’98] Condensation in 2D

[Pupilli, ICPR’06] PF for 3D edge-based tracking

[Klein, BMVC’06] PF for complex object tracking

[Teuliere, ICRA’10] Multiple edge correspondences
Contributions

- Given starting pose
- Gaussian random walk
- No re-initialization
- Initialization
- AR(1) state dynamics
- Auto re-initialization
Initialization

• Given 2D-3D keypoints correspondences
• Randomly choose a set of minimum correspondences
• Solve PnP problem to estimate candidate poses
• Weights proportional to inlier ratio of remaining correspondences
• Importance sampling
\[ X_t = X_{t-1} \cdot \exp(A_{t-1} + dW_t \sqrt{\Delta t}), \]
\[ A_{t-1} = a \log(X_{t-2}^{-1} X_{t-1}) \]

- Instead of Gaussian random walk models
- Linear prediction based on previous states
- Propagate particles more effectively
Re-initialization

\[ \hat{N}_{eff} = \frac{1}{\sum_{i=1}^{N}(\tilde{\pi}(i))^2} \]

Effective number of particle size

![Graph showing the effective number of particles over frames](image)
Experiments

The synthetic image sequence of the “Book” object: complex background case

The synthetic image sequence of the “Car door” object: complex background case

The “Book” object

The real image sequence of the “Teabox” object

The real image sequence of the “Car door” object

The “Car door” object

Single vs. Multiple pose hypotheses

with vs. without AR state dynamics

Reinitialization exp.
Ours vs. BLORT

SYNTHETIC

The synthetic image sequence of the “Book” object: simple background case

The synthetic image sequence of the “Teabox” object: complex background case

REAL

The real image sequence of the “Book” object

The real image sequence of the “Cup” object

Book  
Teabox  
Book  
Cup
Robotic Assembly
Robotic Assembly
Approaches

• 2D Visual Information (Monocular Camera)
  • Combining Keypoint and Edge Features [ICRA’10, ICRA’11, IJRR’12]
  • Extending to Textureless Objects [IROS’12]

• 3D Visual Information (RGB-D Camera)
  • Voting-based Pose Estimation using Pair Features [ICRA’12, IROS’12]
  • Object Pose Tracking [IROS’13]
Edges (or boundaries) are preferred for textureless objects.

- From CAD model to Edge templates
- Efficient chamfer matching [Liu, CVPR’10]
- Coarse 3D pose estimation from 2D chamfer matching results
- Annealing Process after Initialization
Edge Templates

CAD model
Results of Chamfer Matching algorithm
Approaches

• 2D Visual Information (Monocular Camera)
  • Combining Keypoint and Edge Features
    [ICRA’10, ICRA’11, IJRR’12]
  • Extending to Textureless Objects
    [IROS’12]

• 3D Visual Information (RGB-D Camera)
  • Voting-based Pose Estimation using Pair Features
    [ICRA’12, IROS’12]
  • Object Pose Tracking
    [IROS’13]
Overview
• Noise
• Occlusions
• Clutter
Contributions

• Exploiting objects’ *boundary* information
• B2B, S2B, and L2L features
• Better for *planar* objects
• *Sparser* primitives
• More *efficient*
Flowchart

3D RGB-D > Voting-based Pose Estimation using Pair Features [ICRA’12, IROS’12]

- Object CAD Model
- Model Point Clouds
- Pair Feature Extraction
- Hash Table

- 3D Sensor
- Scene Point Clouds
- Pair Feature Extraction
- Voting Procedure
- Object Poses

Yellow -> Offline
Blue -> Online
Geometric Primitives

Surface Points

Boundary Points

Surface & Boundary

Lines
Pair Features

3D RGB-D > Voting-based Pose Estimation using Pair Features [ICRA'12]

Pair Features

S2S (Drost et al.)

B2B

S2B

L2L

\[ f_1 = \| \mathbf{m}_i - \mathbf{m}_r \|_2 \]
\[ = \| \mathbf{d} \|_2 \]
Object Learning

CAD model

Hash Table

3D RGB-D > Voting-based Pose Estimation using Pair Features [ICRA'12]
Flowchart

3D RGB-D > Voting-based Pose Estimation using Pair Features [ICRA’12, IROS’12]
Why Voting?

- Low dimensional pair features: 3D or 4D
- One scene pair feature $\rightarrow$ Many model pair features
- Self symmetric regions
- Noise
- Background clutter
- Voting procedure to overcome the *ambiguities*
- Maximum votes $\rightarrow$ the most likely pose hypothesis
The voting-based pose estimation can reliably estimate poses of the object when there are multiple objects and the background is highly cluttered. The upper row has multiple objects sampled from our test objects. The scan of the lower row contains multiple Circuit Breaker objects.

Two example scenes from the real scans.

From left to right: Results using the four features. The displayed scan of the upper row contains multiple objects in which four objects are false positives. Similarly to the results for the synthetic data, the planar area of the Clamp object caused several false pose estimates. As shown in the lower row, we also tested the four features in the scan which has multiple Circuit Breaker objects. For comparison, we rendered top six pose hypotheses obtained for each feature. Although in general, all features provide good performance for this object as the directions of line segments become unstable for short distances as in Fig. 9. Two selected examples using the B2B feature might be reported.

The normal on its surface is well detected. The Teabox object is well captured since the Cup object exhibits various daily objects as in Fig. 9. Two selected examples using the B2B feature might be reported. It is well suited for our small target objects in Fig. 9. Microsoft released an infrared pattern-based stereo Micro sensor. It is designed for short range scans and thus it can be applied to this mid-range sensor. To verify this, we can find the Cup object in a normal on its surface.

So far, all of the 3D data were obtained via the Mitsubishi line segments. Our algorithm provides reliable 3D point clouds between 0 to 1 meters and line segments. It is designed for activity recognition. Our algorithm can be applied to this mid-range sensor. To verify this, we choose new test objects since previous test objects are too small to be scanned using Kinect. Thus we select three normal on its surface. It is well suited for our small target objects in Fig. 6. Our algorithm can be applied to this mid-range sensor. To verify this, we choose new test objects since previous test objects are too small to be scanned using Kinect. Thus we select three normal on its surface.
TABLE I
AVERAGE NUMBERS OF PAIR FEATURES IN THE SYNTHETIC SCENE
DATASET AND RELATIVE PROCESS TIME.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Number of Features</th>
<th>Relative Process Time(^\dagger)</th>
</tr>
</thead>
<tbody>
<tr>
<td>S2S [23]</td>
<td>23040000 (= 4800 x 4800)</td>
<td>3.21</td>
</tr>
<tr>
<td>B2B</td>
<td>2616953 (≈ 1618 x 1618)</td>
<td>1.00</td>
</tr>
<tr>
<td>S2B</td>
<td>7689280 (≈ 4800 x 1602)</td>
<td>1.20</td>
</tr>
<tr>
<td>L2L</td>
<td>121058 (≈ 348 x 348)</td>
<td>1.03</td>
</tr>
</tbody>
</table>

\(^\dagger\) The fastest method, B2B, is shown as one.

- Our pair features are **sparser** and **faster**.
Exploiting Color Info.

- Industrial parts
  - Low texture or textureless
  - Boundary information is useful

- Daily objects
  - Rich color and texture information
  - Exploit both color and depth information
Color Point Pair Feature

\[ f_1 = \| \mathbf{p}_i - \mathbf{p}_j \|_2 = \| \mathbf{d} \|_2 \]

- **PPF** (Drost et al.): 4 dimensional
- **CPPF** (proposed): 10 dimensional
Why Color Points?

- **Point Pair Feature**
  - Objects having rich variations in surface normals
  - **Inefficient** for planar or self-symmetric objects
  - **False matching** from background clutter

- **Color Point Pair Feature**
  - **Prune** potentially false matches based on **color similarity**
  - HSV color space
  - More **efficient** because unnecessary votes are skipped

To prune unnecessary feature matching
Parallel Implementation

- parallel NVIDIA Thrust lib
  - reduction
  - counting
  - partition
  - binary search
  - sorting
  - ...

3D RGB-D > Voting-based Pose Estimation using Pair Features [IROS’12]
3D RGB-D > Voting-based Pose Estimation using Pair Features [IROS’12]

Test Objects

- Clorox
- Flash
- Kuka Mug
- Milk
- MVG Book
- Orange Juice
- Pringles
- Starbucks Mug
- Tide
- Wrench
Performance Evaluation

Dataset

Gaussian noise

MOSI

SOMI

Real cluttered scenes

Compared Approaches

Hinterstoisser et al., PAMI’11 with/without ICP

Papazov et al., IJRR’12

Drost et al., CVPR’10
To simulate the noise of RGB-D cameras, Gaussian noise is added in the direction of the camera ray. From left to right: $\sigma = 0, 2, 4, 6, 8, 10$ mm, which are 6 different standard deviations. For statistically meaningful results, 50 different test clouds were generated with random rotations for each object, as some of "Clorox" scenes are shown in Figure 5.7. Thus the total number of tested point clouds for 10 test objects are $10 \times 6 \times 50 = 3000$.

Figure 5.9 presents recognition rates with respect to the six different Gaussian noise. As we would expect, the recognition rates of the five approaches decrease as the noise level increases. Our approach and Drost et al. (2010) report similar performance in "Clorox", "Flash", and "Starbucks Mug", and Drost et al. (2010) even shows slightly better recognition rate in "Flash". In "Kuka Mug" and "Starbucks Mug", Hinterstoisser et al. (2012b) is better or at least comparable to both our approach and Drost et al. (2010).

Except these cases, our approach reports the best recognition performance in general, and slightly worse performance is shown by Drost et al. (2010). Hinterstoisser et al. (2012b) shows moderate recognition rates, while Papazov et al. (2012) reports the worst recognition performance overall. According to the results of Hinterstoisser et al. (2012b) with and without the ICP, the ICP refinement helps to increase the recognition rate, but only for smaller Gaussian noise levels. As the standard deviation increases, the additional ICP process even worsens due mainly to the false point data association on the noisy point cloud. It is worth noticing that Hinterstoisser et al. (2012b) is relatively less affected by the Gaussian noise. It is because Hinterstoisser et al. (2012b) relies on template matching in which 2D templates are matched against the input scene image. Since the Gaussian noise is added only to the depth channel, the RGB channels are not affected at all, and thus the approach relatively less degenerates. However, our approach still shows better recognition rates than Hinterstoisser et al. (2012b) in most cases.

The results are also shown as precision-recall curves in Figure 5.8. The curves were generated by varying the threshold value on the score of the pose estimates: the number of votes for both our approach and Drost et al. (2010), the visibility term $\mu_V$ which is the ratio of the model surface area matched to the scene in Papazov et al. (2012), and the template matching score in Hinterstoisser et al. (2012b). According to the precision-recall graphs, our approach outperforms other approaches in most cases with some minor exceptions. In both mug objects, Hinterstoisser et al. (2012b) shows better or comparable precision and recall to our approach. In "Orange Juice", Hinterstoisser et al. (2012b) with the ICP shows slightly better precision with yet less recall. In most cases, Papazov et al. (2012) shows poor recall, mostly lower than 30%.
Results: Gaussian noise

Precision-recall curves of the noise experiment.
Cluttered Scenes
Pose Estimation Results

Hinterstoisser et al. without [1] and with ICP [2], Papazov et al. [3], Drost et al. [4], and ours [5]
Approaches

• 2D Visual Information (Monocular Camera)
  • Combining Keypoint and Edge Features [ICRA’10, ICRA’11, IJRR’12]
  • Extending to Textureless Objects [IROS’12]

• 3D Visual Information (RGB-D Camera)
  • Voting-based Pose Estimation using Pair Features [ICRA’12, IROS’12]
  • Object Pose Tracking [IROS’13]
Motivations

• Posterior p.d.f. as a set of weighted particles
• Slow frame rate due to a serial likelihood evaluation of particles
• Inherently parallel algorithm
• each particle weight update is independent of other updates

To parallelize the time-consuming likelihood evaluation
Contributions

- Rich features from RGB-D channels (colors, points, normals)
- Frame Buffer Object (FBO) in OpenGL & CUDA OpenGL interoperability
- Multiple object rendering
Related Work

[Isard, IJCV’98] Condensation in 2D

[Montemayor, SIGGRAPH’04] Simple PF on GPU

[Klein, BMVC’06] Fast PF using GPU shader


Employ rich features: depth, normals, and color
3D RGB-D > Object Pose Tracking [IROS'13]

Likelihood Evaluation
Designing an efficient and robust likelihood function is the crux of the computer graphics pipeline. Hence, we can efficiently measure the likelihood of each association of a point in the rendered object with a point in the model. The likelihood function can be expressed as:

$$p(Z_t | X_t^{(n)}, M_t) = \prod_{(i,j) \in A} p(z_t^{(i)} | X_t^{(n)}, m_t^{(j)})$$

where $A = \{(i, j) | \text{proj}(x(z_t^{(i)})) = \text{proj}(X_t^{(n)} \cdot x(m_t^{(j)}))\}$

This expression captures the likelihood that a point in the rendered object, $z_t^{(i)}$, is associated with a point in the model, $m_t^{(j)}$, given the RGB-D scene, $X_t^{(n)}$, and the model, $M_t$. The likelihood is computed over all possible associations in set $A$. This formulation allows for the efficient and robust estimation of object poses in real-time tracking applications.
Distance functions

\[ d_e(x_1, x_2) = \begin{cases} 
\|x_1 - x_2\| & \text{if } \|x_1 - x_2\| \leq \tau \\
1 & \text{otherwise}
\end{cases} \]

Euclidean distance

\[ d_n(n_1, n_2) = \frac{\cos^{-1}(n_1^T n_2 - 1)}{\pi} \]

Normal distance

\[ d_c(c_1, c_2) = \|c_1 - c_2\| \]

Color distance
Likelihood Evaluation

\[ p(Z_t | X_t^{(n)}, M_t) = \prod_{(i,j) \in A} p(z_t^{(i)} | X_t^{(n)}, m_t^{(j)}) \]

\[ p(z_t^{(i)} | X_t^{(n)}, m_t^{(j)}) = \exp^{-\lambda_e \cdot d_e(x(z_t^{(i)}), X_t^{(n)} \cdot x(m_t^{(j)}))} \]
\[ \cdot \exp^{-\lambda_n \cdot d_n(n(z_t^{(i)}), X_t^{(n)} \cdot n(m_t^{(j)}))} \]
\[ \cdot \exp^{-\lambda_c \cdot d_c(c(z_t^{(i)}), c(m_t^{(j)}))} \]
3D models on the Web

IKEA LACK Side Table
This is a model of the LACK Side Table available from IKEA. The color of this model is Birch Effect.
For more information, visit: http://www.ikea.com/us/en/catalog/products/481560470

IKEA Karlstad Canapea 2 locuri
For more information, visit: http://www.ikea.com/us/en/catalog/products/392539141

bookcase IKEA EXPEDIT 149x79 cm
For more information, visit: http://julien.perret.googlepages.com/sketchup3dcomponents

IKEA Malm Queen Platform Bed with Nightstands
For more information, visit: http://www.ikea.com/us/en/catalog/products/481560470
Herman Miller Aeron Chair

The Aeron chair didn’t end up in the Museum of Modern Art’s permanent collection just because it looks cool. Although it does, its looks are only the beginning. Aeron accommodates both the sitter and the environment: it adapts naturally to virtually every body, and it’s 94% recyclable. Even if it’s black, it’s green.

For more information, visit:
https://www.hermanmiller.com/
Conclusions

- Contributed toward robust object perception in unstructured environments

Four challenges

- object perception regardless of the degree of texture
- highly cluttered backgrounds
- object discontinuities
- real-time constraints
- combined photometric and geometric features
- multiple pose hypotheses frameworks
- combined pose estimation and tracking
- parallelized on GPU
To close the loop between the geometric era of early computer vision and the currently dominating appearance age, both photometric and geometric features need to be considered.

The combination of these features enables object perception algorithms not only to be more effective but also to handle an increased spectrum of objects.

Two theoretical frameworks using multiple pose hypotheses based on combined features are contributed in this thesis.

These new frameworks are robust to significant clutter and occlusions, and are therefore efficacious solutions for visual object perception in unstructured environments.
Future Work

- Object model adaptation
- Object modeling
- Multi-object tracking
- Scalable object perception
- Object categorization
Thank You
Backup Slides
Voting Scheme I

\[ s_i = T_{s \rightarrow g}^{-1} R_x(\alpha) T_{m \rightarrow g} m_i \]

- **2D accumulator space:** \((m_r, \alpha)\)
- **S2S, B2B, and S2B share the same transform**
Voting Scheme II

\[
l_i^s = T_{s \rightarrow g}^{-1} T_{x}(\tau) R_{x}(\alpha) T_{m \rightarrow g} l_i^m
\]

- 3D accumulator space: \((o^m, \alpha, \tau)\)
Real Scan: $S_2B$ feature
3D RGB-D > Voting-based Pose Estimation using Pair Features [IROS’12]

Dataset: MOSI
Results: MOSI

Hinterstoisser et al. without [1] and with ICP [2], Papazov et al. [3], Drost et al. [4], and ours [5]
3D RGB-D > Voting-based Pose Estimation using Pair Features [IROS’12]

Dataset: SOMI
Hinterstoisser et al. without [1] and with ICP [2], Papazov et al. [3], Drost et al. [4], and ours [5].