Matching CAD Object Models in Semantic Mapping

Sven Albrecht, Thomas Wiemann,
Martin Günther, Joachim Hertzberg

May 9, 2011

UNIVERSITÄT OSNABRÜCK

Introduction

Setting:

- system for model-based object recognition
- 3D point clouds of office environment
- CAD model database available

Result

Hybrid Semantic Map, identified objects replaced by CAD models

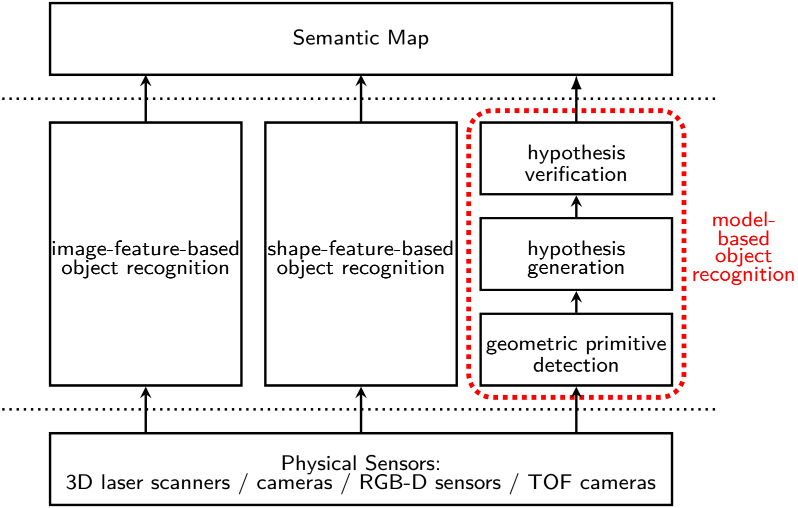
Model-based object recognition

- most object recognition approaches appearance-based
  - extract image / shape features
  - train classifier on labeled training data set
- model-based object recognition:
  - obtain a declarative, *structural* object model, consisting of planar patches and their spatial interrelations
  - extract planar patches and spatial relations from point cloud and match to structural model
  - verify by matching CAD model into point cloud
  - extends semantic labelling of coarse, large-scale structures (walls, floor, ceiling) to medium-scale objects (furniture)

Furniture Domain

- CAD models of many object classes are widely available (e. g., Google 3D Warehouse)
- we focus on furniture detection, because:
  1. availability of accurate CAD models especially strong
  2. most furniture features large planar surfaces
  3. due to rigidness of furniture, spatial relations clearly defined
  4. medium scale objects with sufficient point density

Embedding in general semantic mapping framework



Complementary Approaches

advantages of model-based object recognition:

- re-uses the information contained in CAD models
- instead of one classifier per object, only one detector for each type of geometric primitive necessary
- no classificator training, no labeled training sets required
- to add new object class, just add new CAD model; future: even automatic on-line retrieval conceivable
- recognized CAD model provides additional information (e.g., filling up occlusions)

not applicable when:

- object non-rigid
- labeled training data available, but no CAD model
- object does not consist of clearly identifiable geometric primitives

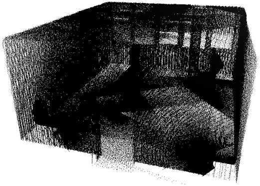
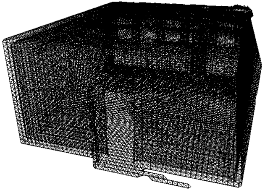
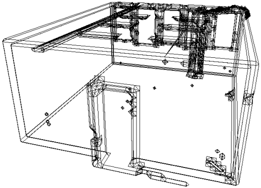
Approach

1. reconstruct surface planes, extract geometric relations
2. semantic classification of planes, initial pose estimation
3. model surface sampling + CAD model fitting

Environment Surface Reconstruction

plane extraction algorithm:

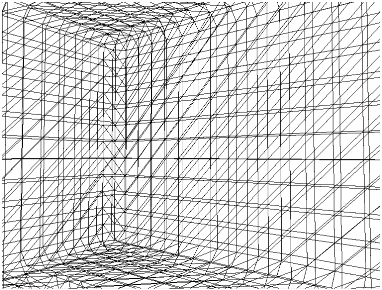
1. generate consistent 3D point cloud
2. convert to triangle mesh using optimized marching cubes
3. extract connected planar regions using region growing

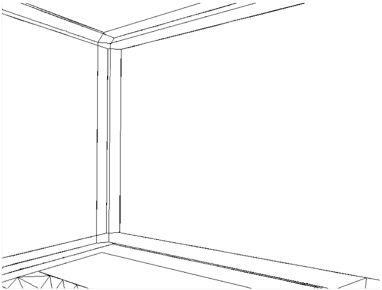
source code available at
`http://lssrtoolkit.sourceforge.net/`

Example of polygonalization algorithm

marching cubes output



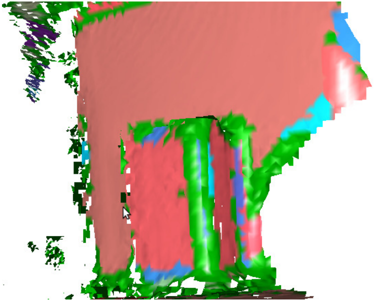
after plane detection



Example: extracted furniture planes

closable shelf (sliding doors) in front view

filing cabinet



Extracting geometric primitives and spatial relations

Extracted planes and spatial relations are added as individuals to OWL-DL ontology

Planar patches, as individuals:

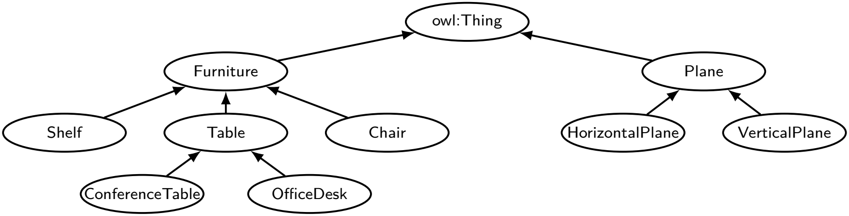
- class: HorizontalPlane, VerticalPlane or simply Plane, depending on surface normal
- attributes: size; height above ground (based on centroid); bounding box

spatial relations added as properties if distance of centroids projected on ground plane $<$ threshold:

- *isAbove*: added between 2 HorizontalPlanes
- *isPerpendicular*: added between a HorizontalPlane and a VerticalPlane

Object Hypothesis Generation

- structural object models represented in OWL-DL ontology, combined with SWRL rules
- OWL-DL reasoner is used to generate hypotheses of possible object locations and initial pose estimation

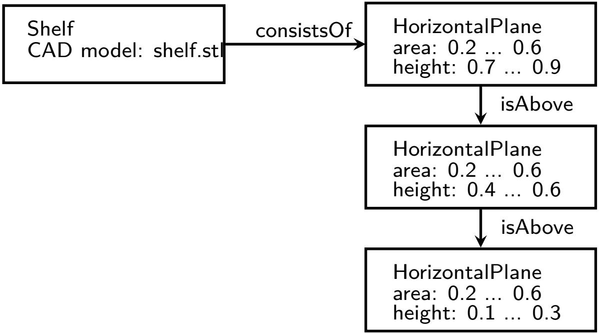


SWRL example

- definitions of furniture classes in the ontology conditions that are used to classify the planes into possible furniture instances
- simple example: a table consists of a horizontal plane with size of at least $1\,\mathrm{m}^2$, and a height between $0.65\,\mathrm{m}$ and $0.85\,\mathrm{m}$

$$\begin{aligned} \mathit{Table}(?p) \leftarrow\ & \mathit{HorizontalPlane}(?p) \wedge \mathit{hasSize}(?p, ?s) \\ & \wedge \mathit{swrlb} : \mathit{greaterThan}(?s, 1.0) \wedge \mathit{hasPosY}(?p, ?h) \\ & \wedge \mathit{swrlb} : \mathit{greaterThan}(?h, 0.65) \wedge \mathit{swrlb} : \mathit{lessThan}(?h, 0.85) \end{aligned}$$

Ontology fragment representing a shelf



Initial Pose Estimation

- calculate axis-parallel bounding boxes and center points of the constituting planes
- center point of one predefined plane (e. g., the table top) is used to anchor the position
- for some objects (like chairs), intrinsic orientation defined by configuration of planes
- for others (like tables), apply PCA to approximate orientation

CAD model matching

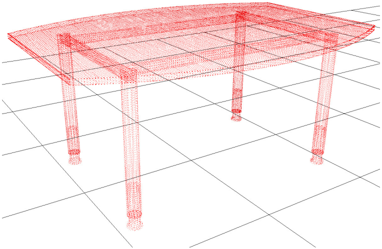
CAD model matching:

- sample CAD model surface to produce an artificial point cloud
- place surface sampling at initial pose estimation
- use standard ICP (restricted to local surrounding of pose estimation) to match the model
- average correspondence error already gives rough estimate of matching quality

Hypothesis verification:

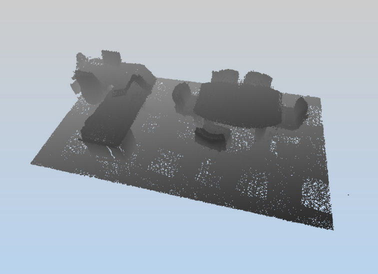
- discretize model into voxels
- check how many of these voxels contain data points in the final pose
- reject/accept hypothesis based on ratio of filled vs. empty voxels

Example Sampling

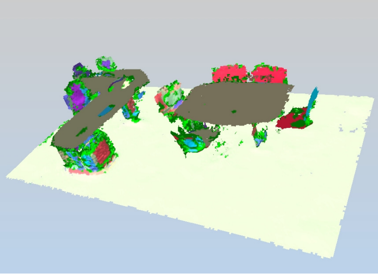


Experiment: Plane Extraction (1)

- large connected surfaces (floor, walls, ceiling) are recognized
- also smaller structures like tabletops, backrests of chairs
- total runtime: $\leq 4\,\mathrm{s}$ on standard Intel Quad Core processor, including normal estimation

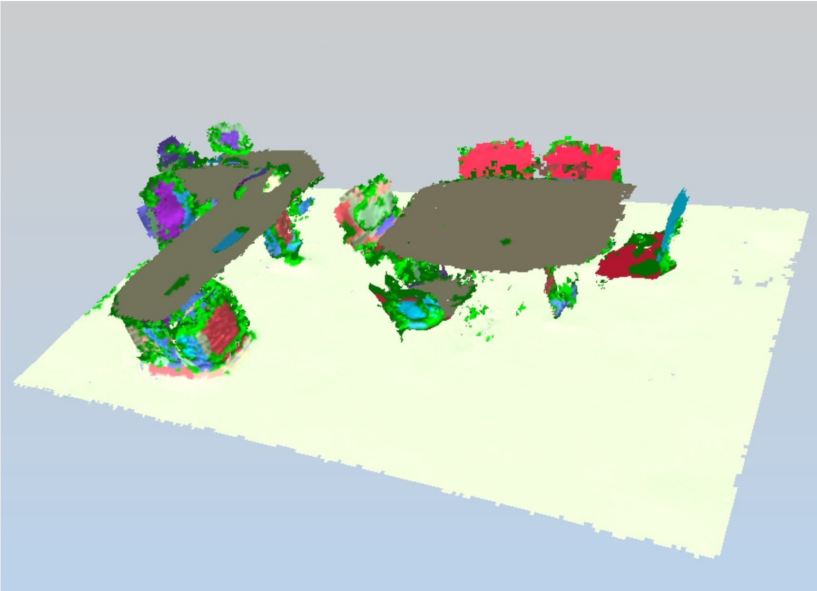


input point cloud

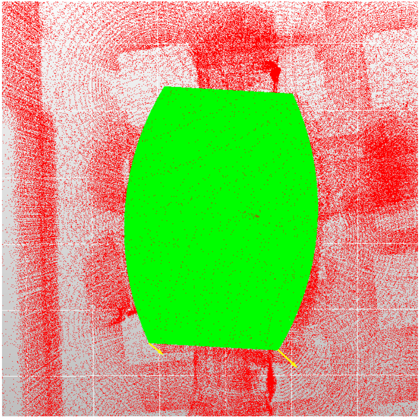


extracted planes

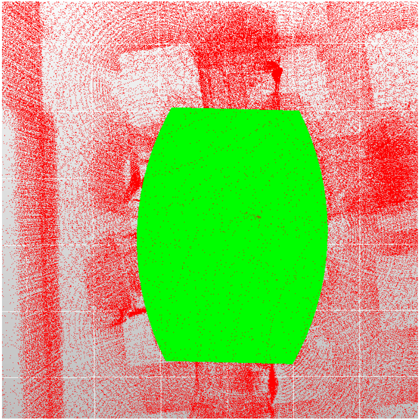
Experiment: Plane Extraction (2)



CAD Model Matching

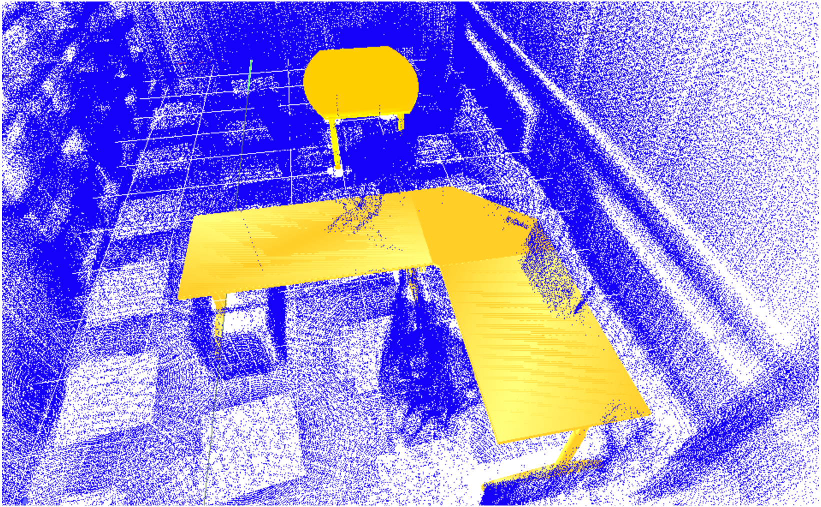


before ICP matching



after ICP matching

Final Result



Future Work

1. alternative representation formalisms to OWL-DL (particularly Statistical Relational Models)
2. improve hypothesis verification error function
3. automate extraction of structural models for hypothesis generation from CAD models