Urban scene modeling from airborne data

Yannick VERDIE

Thesis Advisors: Florent Lafarge and Josiane Zerubia
INRIA Sophia Antipolis, Titane/Ayin Teams

October 15, 2013
3D urban reconstruction
3D urban reconstruction

- Geometry
3D urban reconstruction

- Geometry
- Radiometry
3D urban reconstruction

- Geometry
- Radiometry
- Semantics

Building
Ground
Park
Applications

• Applications for 3D urban reconstruction

- Radio planning
- Movie
- Computer game
- Online services
- Urban planning
- Drone planning
Problem statement

Airborne Acquisition -> Lidar data -> Urban scene modeling

Meshes from Multi-View Stereo
Airborne data

- Point cloud
  - Accurate
  - Not dense
  - Incomplete

Lidar data

Meshes from Multi-View Stereo

- Mesh with triangular facets
  - Complete surface
  - Dense
  - Potential defects
Requirements

5 criteria:
- Geometric accuracy
Requirements

5 criteria:
- Geometric accuracy
- Semantic-aware
Requirements

5 criteria:
- Geometric accuracy
- Semantic-aware
- Low complexity
Requirements

5 criteria:
- Geometric accuracy
- Semantic-aware
- Low complexity
- Scalability

Size of the scene

small  large
Requirements

5 criteria:
- Geometric accuracy
- Semantic-aware
- Low complexity
- Scalability
- Automatic
Surveys

3 major surveys:

- Modeling the Appearance and Behavior of Urban Spaces
  
  [Vanegas et al. Eurographics 09]

- A survey of Urban Reconstruction
  
  [Musialski et al. Eurographics 12]

- Structure-Aware Shape Processing
  
  [Mitra et al. Eurographics 13]
Surveys

“Modeling the Appearance and Behavior of Urban Spaces”

[Vanegas et al., Eurographics 09]

procedural modeling
Surveys

“A survey of Urban Reconstruction”

[Musialski et al., Eurographics 12]

broad overview of the literature on urban reconstruction
SURVEYS

“Structure-Aware Shape Processing”

[Mitra et al., Eurographics 13]

structures to enhance, regularize and manipulate existing meshes
Overview of existing methods

Primitive-based building reconstruction

[ Zebedin et al., ECCV 08]

[ Toshev et al., CVPR 09]

[ Chauve et al., CVPR 10]
Overview of existing methods

- **Primitive-based building reconstruction**
  - [Zebedin et al., ECCV 08]
  - [Toshev et al., CVPR 09]
  - [Chauve et al., CVPR 10]

- **Structure-aware building modeling**
  - [Pauly et al., Siggraph 08]
  - [Mehra et al., Siggraph Asia 09]
  - Zhou and Neumann, CVPR12]
Overview of existing methods

Primitive-based building reconstruction

Structure-aware building modeling

Automatic large scale urban reconstruction

still many unsolved problems
**Contribution**

General pipelines for MVS and LiDAR data
Outline

① Introduction
② Semantic labeling
③ Object Reconstruction: parametric-based object detection
④ Object Reconstruction: mesh-based object reconstruction
⑤ Conclusion and future work
Outline

① Introduction

② Semantic labeling

③ Object Reconstruction: parametric-based object detection

④ Object Reconstruction: mesh-based object reconstruction

⑤ Conclusion and future work
Semantic labeling

What is important?

In many cases, majority of urban scenes can be explained by 3 classes of objects:

- Ground
- Buildings
- Trees
Semantic labeling for Lidar

Input data

Semantic Labeling

Object reconstruction
Semantic labeling for Lidar

Classes for Lidar data

Ground  Trees  Buildings  Clutters

need for geometric features that discriminate the classes
Discriminative geometric features for Lidar

Local non-planarity $\beta_p$
Elevation $\beta_e$
Local dispersion $\beta_s$
Local linearity $\beta_l$

Combine the features to discriminate the classes
## Confidence functions for Lidar

<table>
<thead>
<tr>
<th>Category</th>
<th>High non-planarity $\beta_p$</th>
<th>Low non-planarity $\beta_p$</th>
<th>High elevation $\beta_e$</th>
<th>Low elevation $\beta_e$</th>
<th>Low local dispersion $\beta_s$</th>
<th>Elevation $\beta_e$</th>
<th>Local linearity $\beta_l$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Trees</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Ground</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Buildings</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Clutters</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Confidence functions for Lidar

\[
D(x_i) = \begin{cases} 
D_{\text{tree}}(x_i) & \text{if } x_i = \text{tree} \\
D_{\text{ground}}(x_i) & \text{if } x_i = \text{ground} \\
D_{\text{building}}(x_i) & \text{if } x_i = \text{building} \\
D_{\text{clutter}}(x_i) & \text{if } x_i = \text{clutter}
\end{cases}
\]

\[
\arg \min_x \sum D(x_i)
\]

Need for spatial consistency
Energy minimization over a Markov Random Field

\[ U(x) = \sum D(x_i) + \lambda \sum_{\{i, j\} \in E} V_{ij}(x_i, x_j) \]

\[ D(x_i) = \begin{cases} 
D_{\text{tree}}(x_i) & \text{if } x_i = \text{tree} \\
D_{\text{ground}}(x_i) & \text{if } x_i = \text{ground} \\
D_{\text{building}}(x_i) & \text{if } x_i = \text{building} \\
D_{\text{clutter}}(x_i) & \text{if } x_i = \text{clutter} 
\end{cases} \]

Potts model:

\[ V_{ij}(x_i, x_j) = \delta(x_i \neq x_j) \]

Spherical neighborhood such as \( \{i, j\} \in E \iff \|i - j\|_2 < r \)

Optimisation with graph-cut and alpha-beta swap
[Boykov et al, PAMI 2001]
Semantic labeling for Lidar data

Visual reference from Google map

- Building
- Vegetation
- Ground
- Clutter

Close-up
Semantic labeling for MVS

- Input data
- Semantic Labeling
- Object reconstruction

MVS
Semantic labeling for MVS

Difference with semantic labeling for Lidar data

- Regroup facets into "f-clusters"
  - Tractable
  - Enforce local coherency
Semantic labeling for MVS

Difference with semantic labeling for Lidar data

- Compute f-clusters
- *Building* class splits in two sub-classes
Semantic labeling for MVS

Difference with semantic labeling for Lidar data

- Compute f-clusters
- Building class splits in two sub-classes

Low non-planarity $\beta_p$

Low verticality $\beta_v$

High elevation $\beta_e$

Low non-planarity $\beta_p$

High verticality $\beta_v$
Semantic labeling for MVS

Difference with semantic labeling for Lidar data

- Compute f-clusters
- *Building* class splits in two sub-classes
- **Correction rules**

![Rule 1](image1)
![Rule 2](image2)
![Rule 3](image3)
Semantic labeling

- Roof
- Facade
- Vegetation
- Ground

MVS data

Classification

Close-up
Outline

① Introduction
② Semantic labeling
③ **Object Reconstruction: parametric-based object detection**
④ Object Reconstruction: mesh-based object reconstruction
⑤ Conclusion and future work
Object Reconstruction for Lidar

Focus on tree detection and reconstruction from Lidar
Parametric-based object detection

Objective:
• Localize and reconstruct simple objects

Buildings are too complex structures
Trees can be approximated by simple shapes
Parametric-based object detection

Objective:
- Localize and reconstruct simple objects
- Detection in large scenes

Thousands of simple objects (e.g. trees) are in the scene

Use Marked Point Processes (MPP)
Marked Point Processes

Preliminary:

- A point process describes random configurations of points (of unknown size) in a continuous bounded set K.

- A marked point process is a point process where each point is associated with a parametric objects.
Marked Point Processes

Previous work:

[Lacoste et al., PAMI05]
[Perrin et al., EMMCVPR05]
[Ge et al., CVPR09]
Marked Point Processes

Requirements:
1) **Simple parametric objects**

Object characterized by a limited number of parameters

- **Line-segment**
  - 5 parameters

- **Ellipse**
  - 5 parameters

- **Cone, Ellipsoid, Semi-ellipsoid**
  - 7 parameters
Marked Point Processes

Requirements:
1) Simple parametric objects

2) Energy measuring the quality of a configuration of objects

Standard form of energy:

$$\forall x \in S, U(x) = \sum_{p_i \in x} D(p_i) + \sum_{p_i \sim p_j} V(p_i, p_j)$$

with $\sim$ the symmetric neighborhood relationship such as:

$$p_i \sim p_j = \{(p_i, p_j) \in x^2 : i > j, ||p_i - p_j||_2 < \epsilon\}$$
Marked Point Processes

Requirements:
1) Simple parametric objects

2) Energy

3) Minimization method
   - Unknown number of objects
   - Minimize non-convex energy

Use Reversible-Jump Monte Carlo Markov Chain (RJ-MCMC) [Green 1995]
Optimization method

RJ-MCMC:

- Sequential algorithm with a two-step update mechanism
  
a) Proposition step
  - New configuration is proposed from a proposal density (kernel)
  - New configuration must be close to the current one

b) Acceptance step depending on
  - Random variable
  - Energy variation
  - Stochastic relaxation
Optimization method

RJ-MCMC:

• **Sequential** algorithm with a two-step update mechanism
  
a) Proposition step
  - New configuration is proposed from a proposal density (kernel)
  - New configuration must be close to the current one (local perturbation)

b) Acceptance step depending on
  - Random variable
  - Energy variation
  - Stochastic relaxation

Focus on improving performance of RJ-MCMC
Marked Point Processes

Requirements:

1) Simple parametric objects

2) Energy

3) Minimization method
   - unknown number of objects
   - minimize non-convex energy

Novel optimization method based on RJ-MCMC
   - Exploit two properties for a faster optimization
   - Exploit GPU capability
(1) Parallelization

The Markovian property in the energy: $\Pr(x_i \mid x) = \Pr(x_i \mid x_{\mathcal{N}(i)})$

the blue object and the red object can be updated by MCMC at the same time.
(1) Parallelization

- 4-cell subdivision scheme generates 4 mic-sets \( \{ \text{cell width} = \text{diameter of the neighborhood relation} + \text{maximal length of a move} \} \).

- Mic-Set: a set of Mutually Independant Cells
(2) Non-uniform point densities

How many Birds?
(2) Non-uniform point densities

How many Birds?

Low probability
(2) Non-uniform point densities

How many Birds?

High probability
(2) Non-uniform point densities

We proposed
- quadtree data partitioning for 2D space.
- octree data partitioning for 3D space.

Compatible with the parallelization scheme (1)
Novel optimization method

1-Initialize $X_0 = x_0$ and $T_0$ at $t = 0$;
2-Compute a space-partitioning tree $\mathcal{K}$;
3-At iteration $t$, with $X_t = x$,
   - Choose a mic-set $S_{mic} \in \mathcal{K}$ and a kernel type $t \in \mathcal{T}$ according to probability $\sum_{c \in S_{mic}} p_{c,t}$
   - For each cell $c \in S_{mic}$,
     - Perturb $x$ in the cell $c$ to a configuration $y$ according to $Q_{c,t}(x \rightarrow \cdot)$
     - Calculate the Green ratio
       
       $$R = \frac{Q_{c,t}(y \rightarrow x)}{Q_{c,t}(x \rightarrow y)} \exp \left( \frac{U(x) - U(y)}{T_t} \right)$$

   - Choose $X_{t+1} = y$ with probability $\min(1, R)$, and $X_{t+1} = x$ otherwise
2D Ellipsoidal objects

10800 objects detected, 269 sec (image size: 8Mpixels)
2D Ellipsoidal objects

Time to converge

Energy vs. Time (s)

- our sampler without PT
- our sampler with PT
- RJMCMC
- MBD
- DDMCMC
- Parallel tempering

269s 1078s
Lempitsky and Zisserman, NIPS2010

Input image | Ground Truth | Our result
--- | --- | ---

| cell17 | 209 | 202.9 | 194.1 | 213 |
| cell18 | 184 | 184.6 | 175.9 | 185 |
| cell19 | 187 | 192.2 | 180.1 | 188 |
| cell20 | 169 | 174.1 | 170.4 | 169 |
| cell21 | 147 | 148.6 | 144.4 | 149 |
| cell22 | 184 | 182.6 | 176.5 | 184 |
| cell23 | 159 | 158.3 | 157.6 | 161 |
| RMSE | 1.93 | 4.71 | 9.21 | - |
GPU occupancy

Average time of detection per object (ms)

Image size (#pixels x 10^6)
3D Objects

• Three parametric objects
  ▪ 7 parameters
Three parametric objects

New energy formulation

\[
D(x_i) = \frac{1}{|C_{x_i}|} \prod_{p_c \in C_{x_i}} \gamma(d(p_c, \partial x_i))
\]

\[
V(x_i, x_j) = \beta_1 V_{\text{overlap}}(x_i, x_j) + \beta_2 V_{\text{competition}}(x_i, x_j)
\]

\(d(p_c, \partial x_i)\) is a distance measuring the coherence of the point \(p_c\) with respect to the object surface \(\partial x_i\)

penalizes the overlapping between objects

favors area with similar type of objects
3D Objects

Evolution of the configuration

![Graph showing energy change over time with images depicting the configuration evolution.](image-url)
3D Objects

30k trees in 96min (3.7km² / 12.8M points)
3D Objects

5,400 trees in 53 min (1 km² / 2.3 M points)
3D Objects

Details on cropped area

Visual reference from Google map
Outline

① Introduction
② Semantic labeling
③ Object Reconstruction: parametric-based object detection
④ **Object Reconstruction: mesh-based object reconstruction**
⑤ Conclusion and future work
Mesh-based object reconstruction

Focus on the building reconstruction from MVS

Facades

Roofs
Contributions on building reconstruction from MVS

We propose

• **Multiple Level of Details (LOD)**
  - Definition of the City Geography Markup Language
  - LOD1 – Building as “blocks model, without any roof structures or textures”
  - LOD2 – Building with “differentiated roof structures”
  - LOD3 – Building as “architectural model with detailed wall and roof structures”
  - Visually more appealing
  - More adapted to certain urban applications

[Kolbe et al., 2005]
Contributions on building reconstruction from MVS

We propose

- Multiple Level of Details (LOD)
- **Efficient plane regularization**
  - Predominant in urban environment
  - Support the LOD scheme
  - Efficient on large scale

Existing solutions un-adapted: accurate but too slow for our application

[Zhou and Neumann, CVPR12]

[GLOBFIT]

[Li et al., Siggraph11]
Contributions on building reconstruction from MVS

We propose
- Multiple Level of Details (LOD)
- Efficient plane regularization
- Efficient Binary Space Partitioning (BSP)
  - Exact geometry for BSP is costly (slow)

use a new discrete formulation
Contributions on building reconstruction from MVS

We propose

- Multiple Level of Details (LOD)
- Efficient plane regularization
- Efficient Binary Space Partitioning (BSP)

- Advantages:
  - reconstruct with exact geometry only a subset of cells
  - the plane regularization limits the number of different planes (lower BSP complexity)
Building reconstruction from MVS data

Labeling

Surface extraction

Plane hypothesis from roof and facade f-clusters

Plane regularization

Discrete space partitioning
Building reconstruction from MVS data

Plane regularization:
• 4 pairwise relationships controlled with two parameters $\varepsilon$ and $d$
Building reconstruction from MVS data

Plane regularization:
• 4 pairwise relationships controlled with two parameters $\varepsilon$ and $d$
  - Parallelism

$P_1$ and $P_2$ are $\varepsilon$-parallel if $|\mathbf{n}_1 \cdot \mathbf{n}_2| \geq 1 - \varepsilon$
Building reconstruction from MVS data

Plane regularization:
• 4 pairwise relationships controlled with two parameters $\varepsilon$ and $d$
  ▪ Parallelism
  ▪ Orthogonality

$P_1$ and $P_2$ are $\varepsilon$-parallel if $|\mathbf{n}_1 \cdot \mathbf{n}_2| \geq 1 - \varepsilon$
Building reconstruction from MVS data

Plane regularization:
• 4 pairwise relationships controlled with two parameters $\varepsilon$ and $d$
  ▪ Parallelism
  ▪ Orthogonality
  ▪ $Z$-symmetry

$P_1$ and $P_2$ are $\varepsilon$-$Z$-symmetric if $|n_1 \cdot n_z| - |n_2 \cdot n_z| \leq \varepsilon,$
Building reconstruction from MVS data

Plane regularization:

- 4 pairwise relationships controlled with two parameters $\epsilon$ and $d$
  - Parallelism
  - Orthogonality
  - Z-symmetry
  - Coplanarity

$P_1$ and $P_2$ are $d$-$\epsilon$-coplanar if they are $\epsilon$-parallel and

$$|d_\perp(c_1, P_2) + d_\perp(c_2, P_1)| < 2d,$$
Building reconstruction from MVS data

Plane regularization:
• 4 pairwise relationships
• Groups of parallel planes
Building reconstruction from MVS data

Plane regularization:
• 4 pairwise relationships
• Groups of parallel planes
• 2-step strategy:
  1) **Orientation correction**: propagate orthogonality and z-symmetry relationships from large groups to smaller

  The barycenter of each group is fixed

Before orientation correction

After orientation correction
Building reconstruction from MVS data

Plane regularization:

• 4 pairwise relationships
• Groups of parallel planes
• 2-step strategy:
  1) Orientation correction: propagate orthogonality and z-symmetry relationships from large groups to smaller

2) Position correction: merge co-planar groups
Building reconstruction from MVS data

Plane regularization:
• 4 pairwise relationships
• Groups of parallel planes
• 2-step strategy:
  1) Orientation correction: propagate orthogonality and z-symmetry relationships from large groups to smaller

  2) Position correction: merge co-planar groups

Converge very fast (no data refitting)
Thousand of planes in few seconds
Building reconstruction from MVS data

1. Labeling
2. Plane hypothesis from roof and facade f-clusters
3. Plane regularization
4. Surface extraction
5. Discrete space partitioning
Building reconstruction from MVS data

Discrete space partitioning:
• Volumetric occupancy grid
Building reconstruction from MVS data

Discrete space partitioning:
• Volumetric occupancy grid
• Binary Space Partitioning (BSP)

Red volume = a path in the BSP

? selection of the right path
Building reconstruction from MVS data

Surface extraction:

- The targeted surface is at the boundary between inside and outside volumes

Surface = { + | + | + | }

A min-cut formulation is used to label the volumes. Optimized with min-max flow [Boykov and Kolmogorov, PAMI04]
Building reconstruction from MVS data

Surface extraction:

- The targeted surface is at the **boundary** between **inside** and **outside** volumes

- **Min-cut formulation:**

\[
C(S) = \sum_{c_k \in C_{out}} V_{c_k} g(c_k) + \sum_{c_k \in C_{in}} V_{c_k} (1 - g(c_k)) + \beta \sum_{e_i \in S} A_{e_i}
\]

- \(V_{c_k}\) is the discrete volume of cell \(c_k\)
- \(g(c_k)\) is function of the ratio of inside anchors of cell \(c_k\)
- \(\sum_{e_i \in S} A_{e_i}\) is the discrete area of the resulting surface
Building reconstruction from MVS data

Surface extraction:

• The targeted surface is at the boundary between inside and outside volumes

• Min-cut formulation:

The boundary of the inside volume represents the targeted surface
Building reconstruction from MVS data

Surface extraction:

- The targeted surface is at the **boundary** between **inside** and **outside** volumes

- **Min-cut formulation:**

  The boundary of the inside volume represents the targeted surface

  A control on the sets of planes composing the BSP gives different LODs
Various buildings: (LOD)

Various buildings: 170k facets, ~3min
Experiments

Geometric accuracy (Hausdorff distance) and structure awareness

LOD 1

LOD 2 without plane regularization

VSA

QEM

RMS = 0.88
RMS = 0.48
RMS = 0.43
RMS = 0.65
RMS = 0.82

[Garland and Heckbert, Siggraph97]

[Cohen-Steiner et al., Siggraph04]
Experiments

Robustness assessment
Experiments

Robustness assessment

MVS point cloud data

Point set structuring

MVS-based mesh generation

Our method

LOD1

LOD2

[Lafarge and Alliez, Eurographics13]
Choice of the input: mesh or point cloud

Airborne Lidar data

Smooth DEM

Primitive-based building reconstruction

[Lafarge and Mallet, IJCV11]

Our method

LOD1

LOD2
Experiments

Large scale experiments (scalability)

Paris, 7th district: 11M facets, ~2hours
Outline

① Introduction
② Semantic labeling
③ Object Reconstruction: parametric-based object detection
④ Object Reconstruction: mesh-based object reconstruction
⑤ Conclusion and future work
Contribution summary

Applicative contributions

- Two pipelines for Lidar and MVS data
  - Semantical and structural enhancement of purely-geometric meshes
  - Geometrically accurate reconstruction and visually convincing
  - Scalable and adapted to wide range of applications
Contribution summary

Methodological contributions

- Sampler for Marked Point Processes (MPP) using a parallel scheme
  - Exploit GPU architecture
  - Outperforms current samplers for MPP

- Efficient Binary Space Partitioning (BSP)
  - Rely on a discrete energy formulation for fast approximation
Limitations

Urban labeling
• Classes of objects limited

Sampler for Marked Point Processes
• Efficient only when performed on large scenes for small objects

Building reconstruction
• Piecewise-planar buildings
• Primitive dependant
• Discrete formulation misses details (empty cells)
Future Work

Extensions:

Urban labeling
  • Add more classes of interest for a better labeling (bridge, water,…)

Building reconstruction
  • Generalize the BSP for other primitives (spheres, cylinder,…)
  • Complete LOD3 representation with facade modeling
    • Use data regularization
    • Grammar rules for façade
Future Work

Future directions:

Multiple source of data
  • Use multiple source of data together (terrestrial and aerial Lidar, MVS, images,...)

Functional analysis
  • Combine structure-aware techniques with semantic understanding of urban scenes
THANK YOU

Acknowledgment:

Acute3D, InterAtlas, IGN, Tour du valat, Victor Lempitsky, Roger W. Ehrich, Qian-Yi Zhou