Proceduralization of Urban Models

Ilke Demir
Purdue University
10.14.15
Who Am I?

- PhD candidate at Purdue
- Uncomfortable researcher
- Woman in CS
- Blissful Pixarian!
- Dancer
- Musician
- Gamer
Who Am I?

- PhD candidate at Purdue
- Uncomfortable researcher
- Woman in CS
- Blissful Pixarian!
- Dancer
- Musician
- Gamer
PROCEDURAL MODELING
Modeling buildings, plants, clouds, cities, worlds; using an underlying system of rules.
PROCEDURAL MODELING

• Procedural Modeling of Cities
  [Parish and Müller, 2001]
PROCEDURAL MODELING

- Procedural Modeling of Cities [Parish and Müller, 2001]
- Instant Architecture [Wonka et al., 2003]
PROCEDURAL MODELING

• Procedural Modeling of Cities [Parish and Müller, 2001]

• Instant Architecture [Wonka et al., 2003]

• Procedural Modeling of Buildings [Müller et al., 2006]

...and streets, parcels, roads, et
PROCEDURAL MODELING: 
THE GRAMMAR

\[ G = \{ S, \Sigma, N, R \} \]
PROCEDURAL MODELING: THE GRAMMAR

\[ G = \{ S, \Sigma, N, R \} \]

Starting symbol, Terminals, Non-terminals, Rules
PROCEDURAL MODELING: A GRAMMAR

\[ G = \{ \text{Building}, \{\text{wall, win, }\ldots\}, \{\text{Building, Roof, }\ldots\}, <R> \} \]
PROCEDURAL MODELING: A GRAMMAR

\[ G = \{ \text{Building, } \{\text{wall, win, ...}\}, \{\text{Building, Roof,...}\}, <R> \} \]
PROCEDURAL MODELING:
A GRAMMAR

\[ G = \{ \text{Building,} \{\text{wall, win, } \ldots\}, \{\text{Building, Roof, } \ldots\}, <R> \} \]

\[ R: \]
- Building -> Roof (MidFloor)* Base
- Roof -> roofPl (chimney | roofWin)*
  | Roof roofPl
- MidFloor -> wall (win wall)*
- Base -> (wall)+
PROCEDURAL MODELING: AN INSTANCE

- Building -> Roof (MidFloor)* Base
- Roof -> roofPl (chimney | roofWin)*
  | Roof roofPl
- MidFloor -> wall (win wall)*
- Base -> (wall)+
PROCEDURAL MODELING: AN INSTANCE

- Building -> Roof (MidFloor)* Base
- Roof -> roofPl (chimney | roofWin)* | Roof roofPl
- MidFloor -> wall (win wall)*
- Base -> (wall)+
PROCEDURAL MODELING

Procedural Representation (Grammar) $\rightarrow$ Well-known $\rightarrow$ Geometric Model
PROCEDURAL MODELING

Well-known

Procedural Representation (Grammar) <-> Geometric Model
PROCEDURAL MODELING

Procedural Representation (Grammar) \rightarrow Well-known \rightarrow Geometric Model

RESEARCH!
MOTIVATION

• Urban reconstruction, manual modeling and crowd-sourced models: available.

• New content synthesis, rendering and compaction: hard.

• Repetitions: exploitable (for procedural representation and completion)

• Models contain semantic information, but discovering such influences is hard.
PROCEDURALIZATION

• Converting existing urban models into procedural representation.
PROCEDURALIZATION

• Converting existing urban models into procedural representation.

Using inverse procedural modeling methods,
PROCEDURALIZATION

• Converting existing urban models into procedural representation.

Using inverse procedural modeling methods on, meshes, point clouds, textured models
Converting existing urban models into procedural representation.

Using inverse procedural modeling methods on, meshes, point clouds, textured models of, buildings and cities.
WHY PROCEDURALIZE?

Proceduralization for Grammar Creation

Proceduralization for Editing

Proceduralization for Reconstruction
WHY PROCEDURALIZE?

Proceduralization for Grammar Creation
Let’s Organize...

- Who? Why? What?
- Previous Work
- Our Research
- City Proceduralization
  - Dissimilarity Clustering
  - Proceduralization
  - Results
- What next?
- Have fun!
STYLE GRAMMARS FOR INTERACTIVE VISUALIZATION OF ARCHITECTURE [ALIAGA ET AL., 2007]

- Map images to a simple model
- Subdivide into features
- Create a grammar instance
- Apply to a new model
- Render with tex or stylized
A CONNECTION BETWEEN PARTIAL SYMMETRY AND INVERSE PROCEDURAL MODELING [BOKELOH ET AL., 2010]

- Grammar extraction using dockers and docking sites
- Operational on both polygonal and point set input
- Limited decomposition, no reconstruction/completion
- No (hierarchical) rules, only “glueable” components
• A method for bringing artistic control to PM
• Find the best derivation by maximizing the likelihood of an instance based on user input. (Bayesian inference)
• Use rjMCMC to span the derivation space.
Proceduralization of Buildings at City Scale
Ilke Demir, Daniel Aliaga, Bedrich Benes,
International Conference on 3D Vision,
Dec 10, 2014, Tokyo, Japan.
PROCEDURALIZATION FRAMEWORK

Geometric Input

Segments

Label

Grouping

Construct

Parse Tree

Grammar

Extract

Segment
PROCEDURALIZATION FRAMEWORK

**Geometric Input:** Buildings

- **Segments:** Components
  - **Hierarchical clustering**

- **Grouping:** Simplified Components
  - **Construct**

- **Parse Tree:** Hierarchical Tree
  - **Neumann Partitioning**
  - **Grammar:** CFG

- **Split planes**
PROCEDURALIZATION FRAMEWORK

Geometric Input: Buildings

Hierarchical clustering

Grouping: Simplified Components

Segments: Components

Hierarchical Partitioning

Parse Tree: Hierarchical Tree

Neumann partitioning

Grammar: CFG

Proceduralization for Grammar Creation
A semi-automatic segmentation adaptive to segmentation granularity

A proceduralization method to convert the segments and their patterns into a grammar, and exploit them for completion

A synthesis method that uses the procedural representation to directly edit the point cloud, while preserving the structure
COUPLED SEGMENTATION AND SIMILARITY DETECTION FOR ARCHITECTURAL MODELS
ILKE DEMIR, DANIEL ALIAGA, BEDRICH BENES
SIGGRAPH 2015

- A set-cover formulation for architectural model segmentation
- A novel combinatorial optimization to couple segmentation and similarity detection
- A geometric approach to reduce the search-space for a combinatorial optimization
INFERRING HIERARCHICAL SPLIT GRAMMARS FOR SYNTHESIS OF ARCHITECTURAL MODELS
ILKE DEMIR, DANIEL ALIAGA, BEDRICH BENES (SUBMITTED)

- A novel inverse procedural modeling approach, without any template grammar;
- A structure discovery method to parse a collection of building elements into a split tree.
- An interactive procedural engine for local and global style-preserving synthesis and editing of 3D architectural models.
SUMMARY

b) Hierarchical Clustering

c) Grammar $(T, N, R, S)$

d) Color-Coded Components

e) Synthesized and Simplified Procedural Model
CONTRIBUTIONS

• Any building => Procedural representation

• Hierarchical de-instancing to organize the city based on similarity

• Hierarchical clustering to organize the city based on structure
APPROACH

Model => Components => Features => S. Tree => T. Graph => Rules => Grammar
DE-INSTANCING

• Provides flexible and visually coherent component labeling,

• Supports control over alphabet size
DE-INSTANCING

• Find split planes
  - move near plane
  - render top-down
  - calculate der. of pix.
• Find comp. orientation
  - render the comp. cont.
  - fit&extrude or. rect.
• Extract features
  - img proc. for windows
  - comp. props.
• Cluster hierarchically
  - k-means, k(h)=$(1/2)^h \cdot n$
  - repr. is closest to mean
• Provides control over the generated rules by changing weights of edges.

• Based-on-need grammar extraction is available.
PROCEDURALIZATION

- Select a simp. level
- Create the terminal graph
  - vertices = simplified bdgs
  - edges = nbrs in a thresh.
  - weights = F
- (std, mean, freq)

\[ D(A, B) = \frac{w_m \bar{d}_{AB} + w_s \sigma_{AB}}{N_{AB}} \]

- Apply Neumann-clustering recursively
- Obtain the parse tree of the city instance
RESULTS

Rendering:
Allows existing content to be renderable by procedural renderers.

Compression:
Allows model compression without loss in visual content.
RESULTS

• reduces geo and tex by 2 to 21 times
• produces 177 - ~1000 terminals & 5 - ~100 nonterminals.
• simp+proc take 2-4 hrs for 3000 to 6000 bdgs (180km2)
• provides interactive editing and novel synthesis
RESULTS - VIDEO

Simplified & Proceduralized
New York
**RESULTS**

**Localization**: provides model based location estimation, based on the same feature computation.

*Figure 12. Localization*. Given a) a street view, b) which is rectified, our system extracts features and c) finds the viewing location to within one of our locations, and d) can render a synthetic view. Other potential locations are e.g. See text for more details.
What Next?

- Extending city proceduralization:
  - Co-segmentation for component coherency, inter-building similarity, and compression.
  - Learn a per-type grammar from the segments.
  - Learn also the constraints and extents, to improve the editing and semanticality of the synthesis.
- Apply the proceduralization framework to other domains.
- Create an infinite database of buildings.
THANKS!
Got Feedback?

Talk about it! #GHC15 #proceduralization @ilkedemir

Say hi! ilke@pixar.com idemir@purdue.edu

Rate and review! http://ddut.ch/ghc15
(GHC 2015 in app store)


REFERENCES

RESULTS (ICCV)

Consensus model (CM) improves the reconstruction and point cloud.
RESULTS (ICCV)
RESULTS - VIDEO (SIGGRAPH)
RESULTS - VIDEO (SIGGRAPH)