CIB Student Chapter
Department of Building and Real Estate
The Hong Kong Polytechnic University

A Derivative-free Optimization Approach for Automated As-built 3D Modelling

F Xue
RAP
Dept of REC, HKU

18 August 2017

HKURBANlab
ilab
THE UNIVERSITY OF HONG KONG 香港大學
faculty of architecture 建築學院

Outline

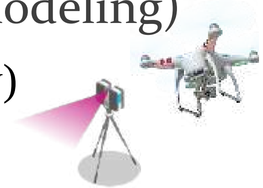
Section 1

BACKGROUND & OPPORTUNITY

1.1 As-built 3D modeling of civil infrastructures

- As-built models [1]
  - Increasingly important for AEC/FM†
    - Construction management
    - Facility management
    - Built env. conservation
    - Business with VR/AR, *etc.*
  - See as-planned, as-designed, as-demolished BIM[2]

- Popular technologies (surface modeling)
  - Photogrammetry (videogrammetry)
  - Point cloud
  - 3D Geographic information system
  - Others (statistical rules, deep learning [3], etc.)



†: Architecture, Engineering and Construction/ Facilities Management

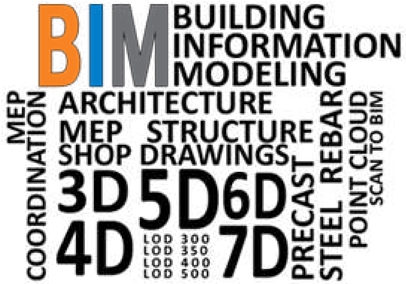
An example of photogrammetry: Kowloon Wall City (Source: patrick-@sketchfab.com)

An example of point cloud: Pompei City (Source: MAP-Gamsau lab, CNRS, France)

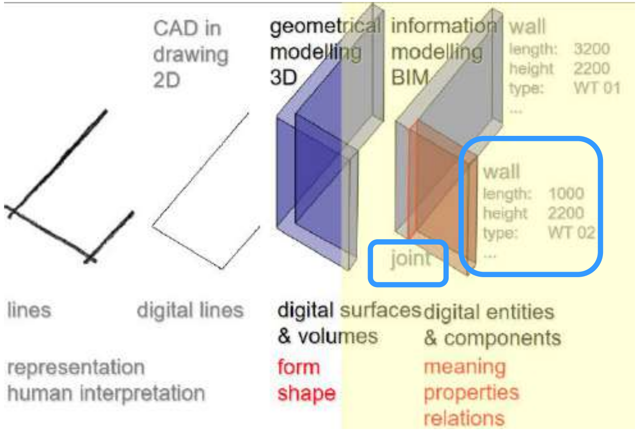
An example of GIS-based: 3D Berlin (Open Data, source: berlin.de)

1.1 Final goal: Semantically rich as-built BIM

- BIM (building information model, narrow sense)[2]
  - The digital representation of physical and functional characteristics of a facility
  - A shared knowledge resource for information about a facility serving as a reliable basis for decisions making
- Two types of semantic information in BIM[2, 4-5]
  - *Attributes* of an individual construction component
    - geometric (*e.g.*, size, position, shape, & textures)
    - non-geometric (*e.g.*, type, material, & functions)
  - *Relationships* between components
    - *E.g.*, dependency, topology, and joints



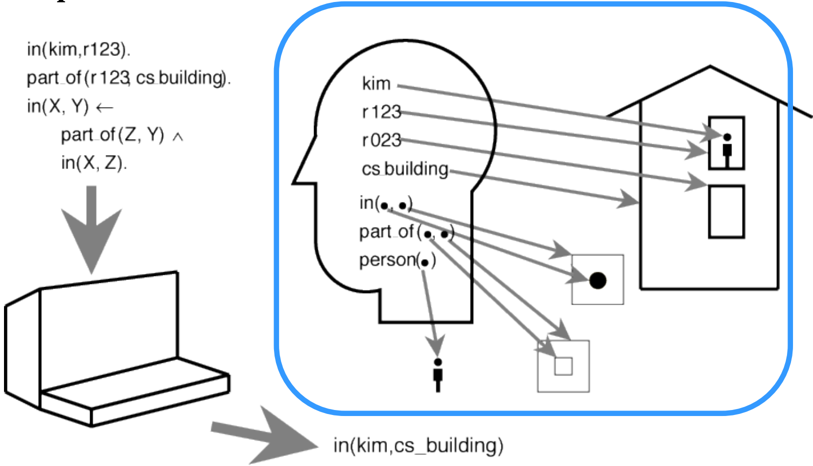
A word cloud of BIM
(Source: advenser.com)



An evolution view of CAD model/BIM [6]

1.1 Final goal: Semantically rich as-built BIM

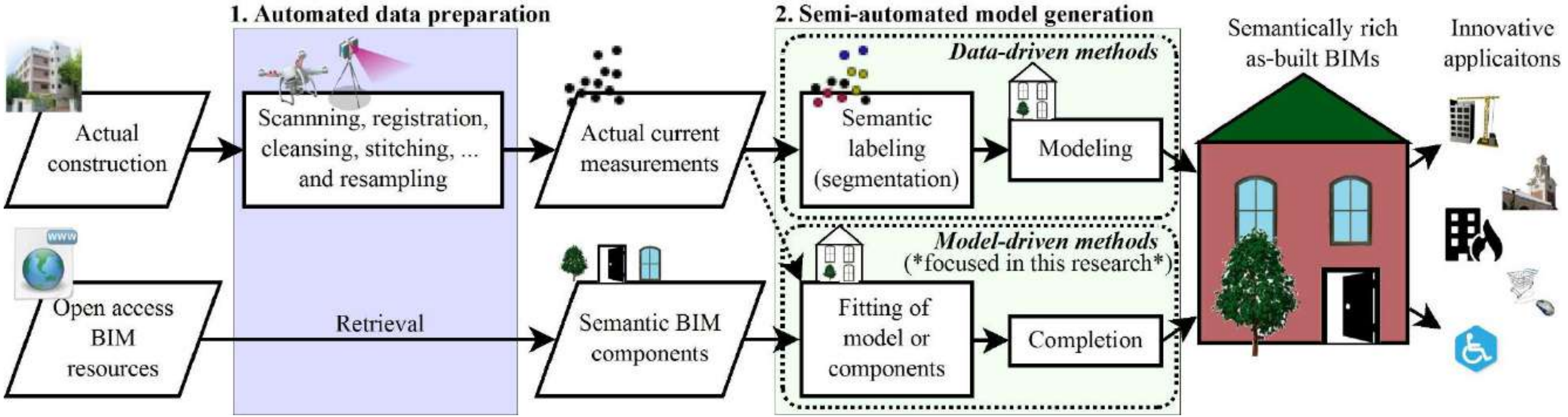
- Semantically rich as-built BIM
  - *Actual* building geometries, *current* function, *real* topology, *etc*.
    - In addition to the surface of the building envelope
  - Advancing the knowledge frontiers of
    - Smart city applications
    - Heathy aging scenarios
    - Robotics and computer vision
    - Artificial intelligence, *etc*.
  - Downward compatibility
    - Methods and technologies *should* also work with as-built 3D building models
  - Relating standards
    - LOD (CityGML level of details[7]), IFC[8], *etc*.



Semantics and artificial intelligence (Source: artint.info)

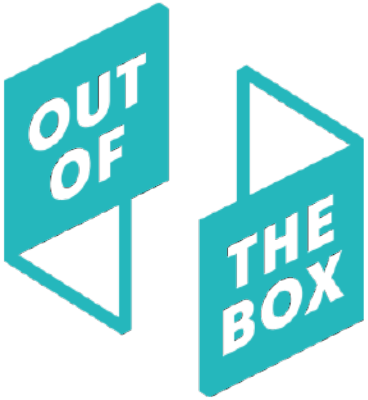
1.2 The (semi-)automated modeling methods

- Two categories of methods (for both)
  - Data-driven: On (pre-processed) point clouds or images
    - Poisson mesh surface, RANSAC planes/spheres, edge detection, image segmentation, *etc*.
  - Model-driven: Recognizing & fitting the known (BIM) components against the data
    - Evolutionary fitting of components, context-based region growing, VR of pipes/ribbons…



1.2 The limitations

- Limitations of existing methods
  - Data-driven
    - Unsatisfactory semantic/abstraction discovery
      - Huge model size
    - Automation level of modeling
      - Tedious and error-prone manual work
    - High requirement on measurement data
      - Expensive equipment
  - Model-driven
    - *Ad-hoc* project setting/ context
      - Poor reusability
- Thinking out of the box
  - Exposing the modeling process to general decision science/OR study



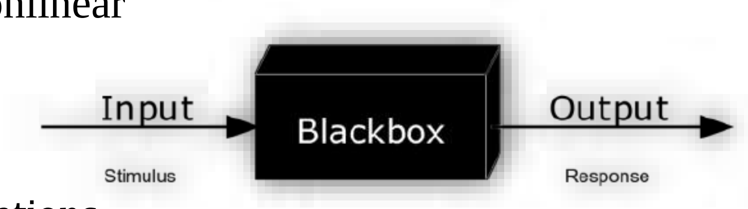
1.3 Derivative-free Optimization in OR†

- Optimization (a.k.a. Mathematical programming)
  - the selection of a *best* element (with regard to some criteria) from *some* set of available alternatives.

$$\max f: \mathbb{R}^n \longmapsto \mathbb{R}$$

An example of optimization

- Nonlinear optimization
  - When *objective function* or some *constraints* are nonlinear
- Derivative-free Optimization (DFO) [9]
  - *Objective function* or *constraints* are *unknown*
    - E.g., model selection, parameter tuning in simulations
    - Especially when function is *very expensive* or *unanalyzable*
  - Challenging (*NP*-hard), but achieved significant success
    - In applied science and engineering such as *molecular biology* and *material sciences*

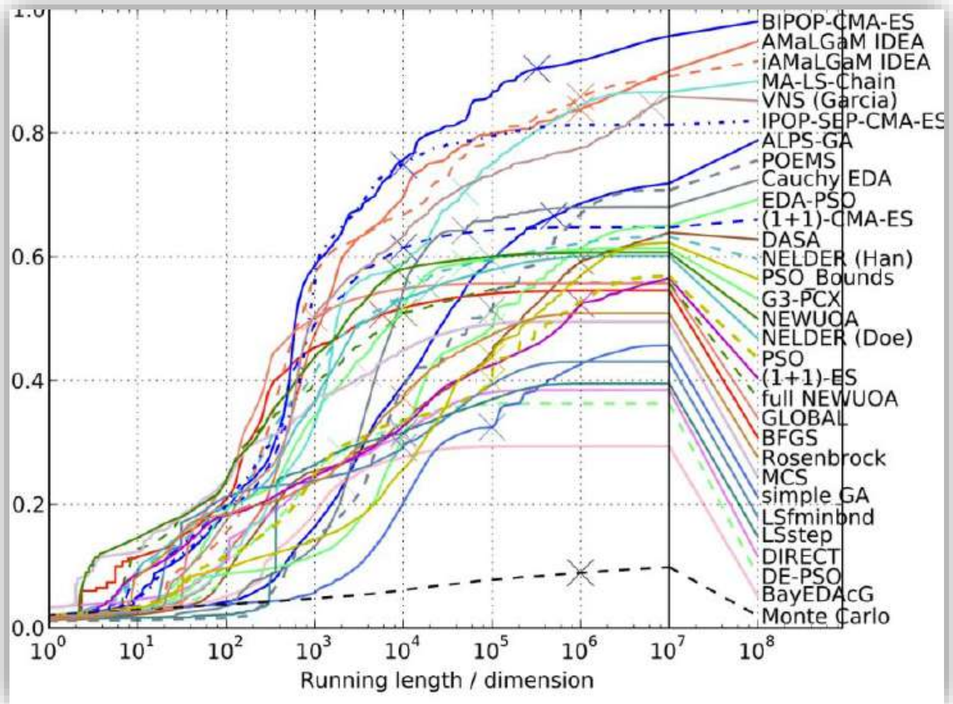


DFO: Manipulating a black-box
(Figure adapted from Wikipedia)

†: Operations Research

1.3 Derivative-free Optimization methods

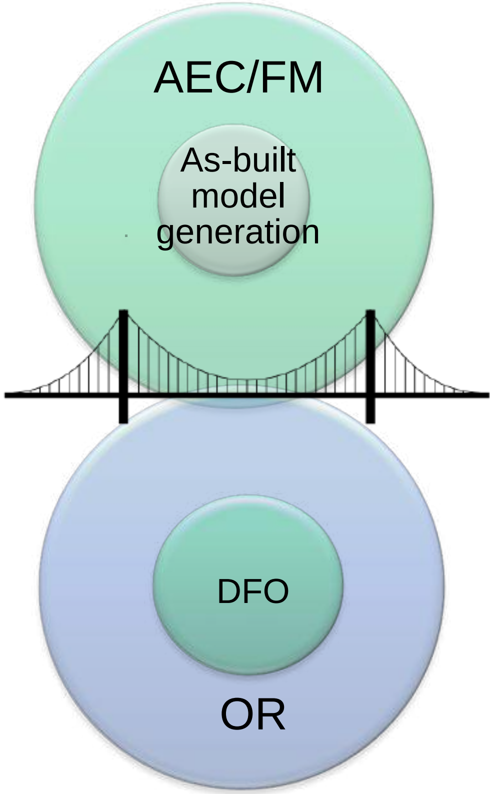
- A long list of off-the-peg algorithms for solving optimization problems as a black-box:
  - Surrogate methods
    - CMA-ES (Covariance matrix adaptation with evolution strategy) [11] and its variants are competitive
  - Trust-region methods
  - Metaheuristics (GA, PSO, VNS, *etc.*)
  - Hyper-heuristics, data mining
  - … and Monte Carlo



Comparison of algorithms for BBOB-2009 (Black-Box Optimization Benchmarking, higher is better) [10]

1.4 An opportunity

- The questions
  - *Can the model generation be generally solved by DFO methods*?
    - *If true, can semantic data be discovered at the same time*?
- If all true, we can
  - Map between a typical problem in AEC/FM and a class of powerful algorithms in OR
    - Also expose as-built model generation to many other nonlinear methodologies
  - Discover semantic (abstraction) information



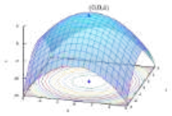
Section 2

AS-BUILT MODELING AS OPTIMIZATION

2.1 A meta-model of as-built 3D modeling

- Given a reference **measurement**, a set of parametric **components**
- A meta-model of constrained optimization is from such a formulation:

$$\max f(X)$$
$$\text{s.t. } C(X) \le 0$$



Meta-model of constrained optimization & its solution space

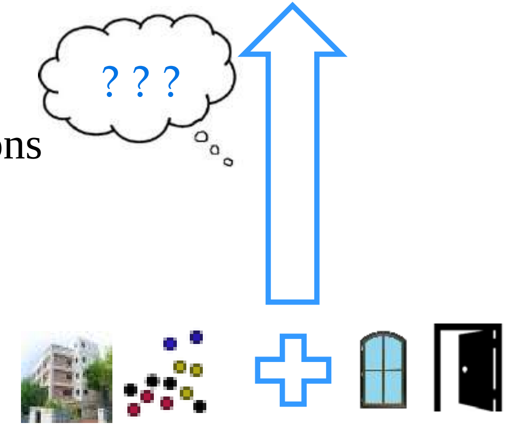
Tells computers:

What to change

What is good

Rules to follow

  - The **variables (X)** are the parameters of the components;
  - The **objective function (f)** is to maximize the similarity (or minimize dissimilarity) between the 3D model (as combinations of the parametric components) and the measurement; and
  - The **constraints (C)** over the variables are the topological relationships between components.
- Meta-: Abstraction
  - from Greek prefix *μετά*-, "beyond"

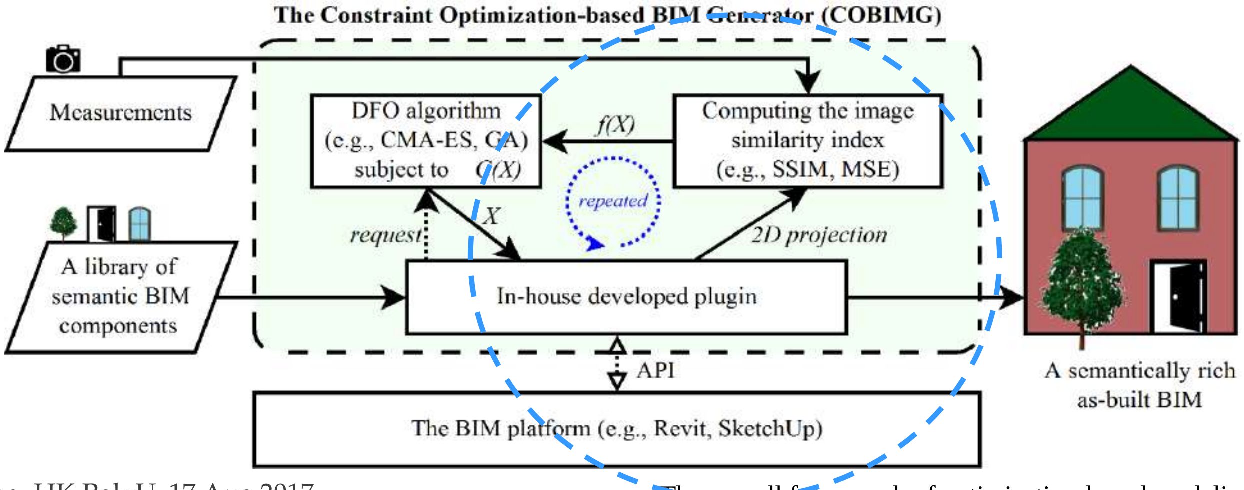


Reference measurements

Parametric (& semantic) components

2.1 The framework: A bird's-eye view

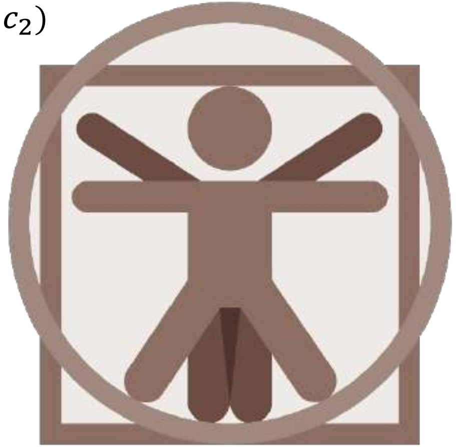
- The full framework (using photo measurement as an example)
  - Input 1: Reference measurements (photos)
  - Input 2: Semantic and parametric components
  - Process: Systematically finding the fittest model by solving meta-model with DFO methods
  - Output: A semantic as-built model



The overall framework of optimization-based modeling

2.1 Formulation of the meta-model

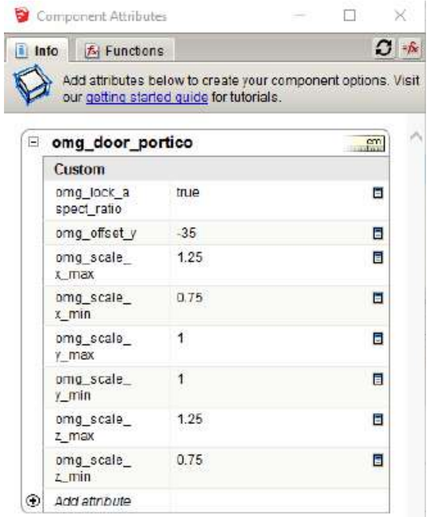
- The variables
  - $X_i = <cl, l, s, r_z>$ for each i-th component instance
    - cl: class, l: location, s: scaling, r_z: rotation-z
- The objective function[12] (similarity between A & Â)
  - $SSIM = structure \cdot luminance \cdot contrast = \frac{(2\mu_{\hat{A}}\mu_{A}+c_1)(2\sigma_{\hat{A}A}+c_2)}{(\mu_{\hat{A}}^2+\mu_{A}^2+c_1)(\sigma_{\hat{A}}^2+\sigma_{A}^2+c_2)}$
  - $Similarity = 1 - \mathrm{MSE} = 1 - \frac{1}{n}\sum_{i=1}^{n}\left(\hat{A}_i - A_i\right)^2$
- The constraints
  - $C(X) = \{C_{\mathrm{I}}(X_i)\} \cup \{C_{\mathrm{R}}(X_i, X_j), i \neq j\}$,
    - C_{I} : about individual component X_i
    - C_{R} : the topological relationships between any (i-th, j-th) components



Restructuring and formulation

2.1 Details of the topological relationships

- Topological relationships
  - Categories
    - Adjacency: ON_TOP_OF, BELOW, NEXT_TO, …
    - Separation: SEPERATED
    - Containment: CONTAINS_ON, CONTAINS_IN
    - Intersection: INTERSECTS_WITH
    - Connectivity: CONNECTS_TO
- Semantic definition
  - Adding properties like *scaling* and *topological relationships* to their SketchUp dictionaries
    - E.g., ON_TOP_OF, BELOW, CONTAINS_ON, *etc.*

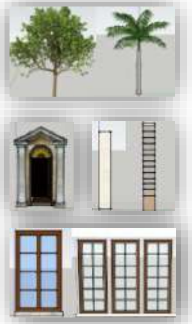


An example of the dictionary of component in SketchUp

2.2 A pilot: A demolished building at campus

- The pilot case
  - A demolished baroque-style two-storey building
    - Once occupied by School of Tropical Medicine and School of Pathology, HKU
  - Input: A photo
- Preparing parametric components
  - Only apparent (>1m) components
    - 1 door portico, 1 tree (unknown type), 2 storeys of walls (...)
    - 5 identical windows on 1/F, 4 on G/F (all unknown types)
  - 7 components were collected from *3D Warehouse* of *SketchUp*
    - With a keyword filter "*baroque*"
    - With limited (3) pairs of conflicting components
    - Adjustment: Removing extra parts, alignment

A historical photo (Source: MTR HKU Station, re-photographed by an Android phone)



- Door portico
- Tree × 2
- Wall × 2
- Windows × 2

(Contributors: Mohamed EL Shahed, Richard, KangaroOz 3D, Yoshi Productions, 3dolomouc, Architect, Ben @ 3D Warehouse)

2.2 The problem

- Meta-model of as a constrained optimization problem
  - Minimize the dissimilarity
    - Between the projected image of model and the input photo
    - Similarity metric is the SSIM
  - With respect to topological constraints
  - Computational functions implemented on SketchUp (2016 Pro) Ruby API
    - Objective function interface
    - Variables as parameters (per component)

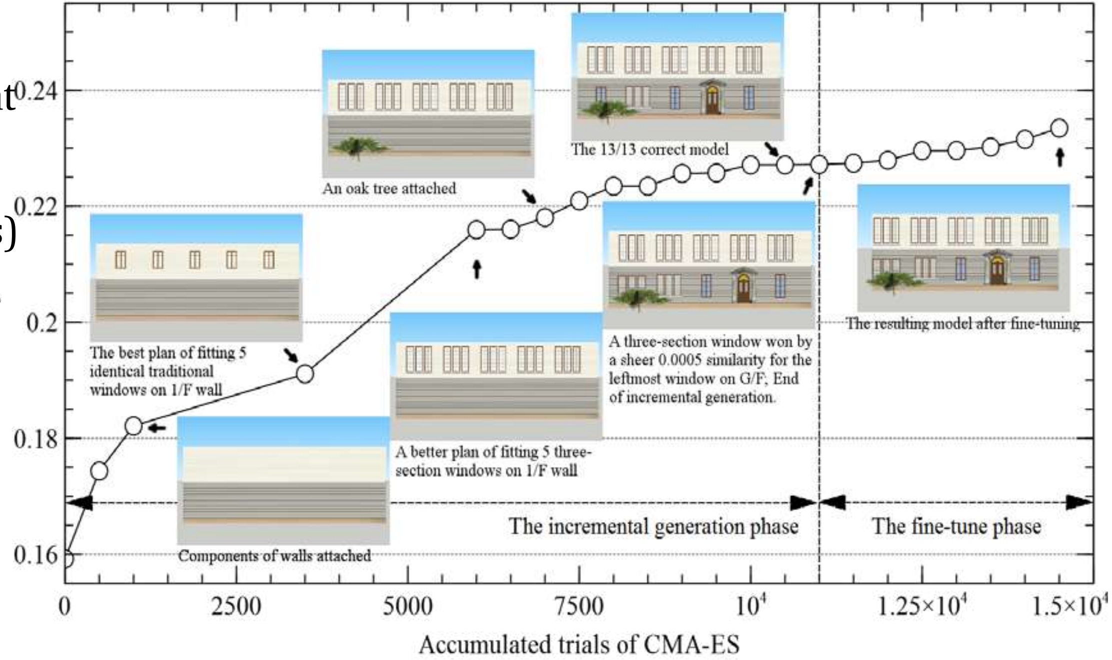
      Manifolds (0, 1) + scaling (*xyz*) + location (*xyz*) + ~~rotation ($\alpha\beta\gamma$)~~ = 4 ~ 6 variables
    - Constraints of topological relationships
  - An invisible *Ground* object is placed at first

min f = SSIM

s.t. Semantic constraints of *position*, *scaling*, and *ABOVE*/ *BELOW*/ *CONTAINS_ON* for each component

2.2 A computational experiment

- Automated generation
  - 200 trials per target component
  - Two phases:
    - Incremental (11,000 trials)
    - Refinement (3,500 trials)
  - The Solver: CMA-ES (C++ code [13] in a Ruby wrapper)
  - Time: 5,012.6s
- Observation
  - Fully automatic
  - Fault-tolerant (see the windows)
  - Semantic/grammar-enhanced

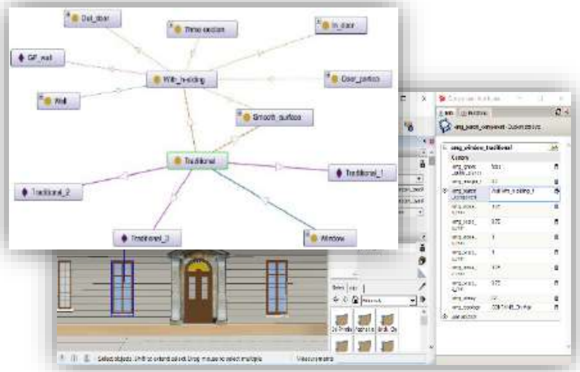


The automated optimization process of the proposed method with annotated SketchUp models in the test with 500 trials for fitting each component (The incremental generation phase: 1~11,000; The fine-tuning phase: 11,001~14,500)

2.2 Results and post-processing

- Obtained
  - The facade in the photo
  - Semantic links
- Post-processing
  - Manual completion
    - Copy & paste
  - Georeferencing and display in 3D

(a) Direct result: The façade in the photo



(b) The semantic links illustrated in Stanford Protégé (Circle denotes a component class and a diamond stands for an instance/object)

(c) Manually completed approximate model (~15 minutes)

(d) Georeferencing and illustration on Google Earth, near MTR Exit A (~5 minutes)

2.2 The number of trials is critical

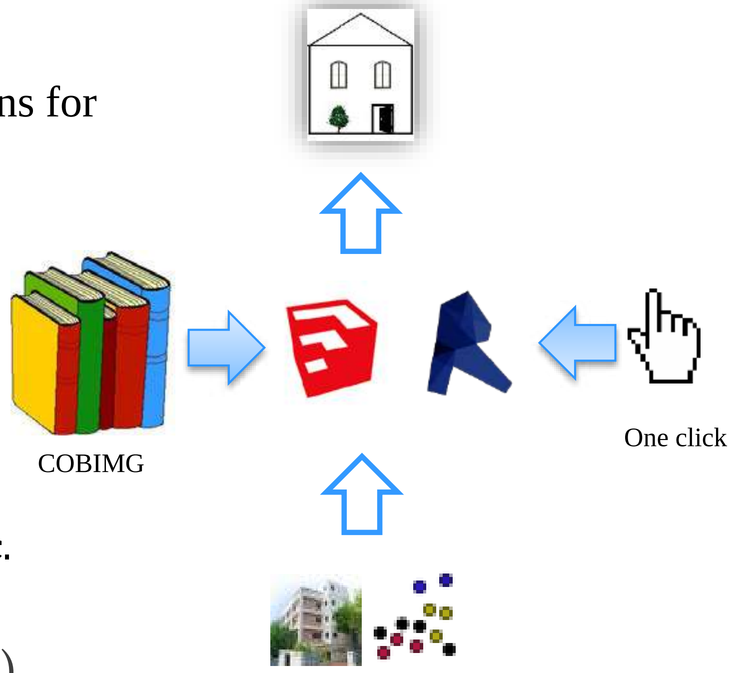
- When increasing the number of trials per component 10 to 1,000
  - both the similarity and overall time cost were *monotonically* increasing
  - Similarity
    - From 0.16 to over 0.24
  - Correct components
    - From 4~5/13 to 12/13
  - Time cost: From 100+s to 10,000+s
    - Over 97% was consumed by BIM environment
      1. Manipulations
      2. Projections

The trends of similarity, overall time cost, and correctly generated components when changing the trails of CMA-ES for fitting each component

Used by	Functions	%time
SketchUp	Manipulating components; 3D to 2D projection	97.67%
Similarity	Computing the image similarity index	1.45%
DFO	Optimizing the parameters of a component	0.00%
System	Reading/writing of temporary image files	0.88%

3.3 COBIMG & live demonstration

- A library COBIMG (constrained optimization-based BIM generator) is under development
  - A shared computational library with specific plugins for
    - SketchUp, Revit (soon), *etc*.
  - Multiple meta-models with various
    - Objective functions
    - Measurement types, and
    - Solving algorithms
  - Multiple modeling options
    - Ontology-guided, free discovery, finetuning, etc.
    - Extended the earlier pilot study
- Demo (Known sum of types for a quick demo)



Section 3

DISCUSSION & FUTURE RESEARCH

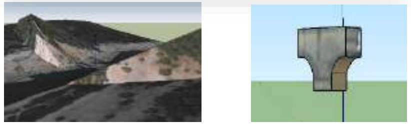
3.1 Discussion

- Meta-modeling of as-built 3D modeling as constrained optimization
  - Pros: General, simple, no explicit object recognition/segmentation (also challenging)
  - Cons: A larger search space (slower), slow full projection, limited by pixels, less accurate
- Semantic definitions of components
  - Pros: Realized ‘grammar’ of components, simplified optimization
  - Cons: Some manual work needed, subject to redefinition from a project to another
- The framework as a whole
  - Pros: High automation, linearly incremental time, reusing components and abstractions, less requirements on equipment, tolerant to errors, (hopefully) semantically rich
  - Cons: Less accurate in geometry, still in its *infancy*
  - Answers to the question: 1) True; 2) Applicable to some relations
    - Semantic recognition/segmentation is another pillar for semantic BIM

3.2 Future research

- Effectiveness
  - More domains (e.g. infrastructures, *etc*.)
  - Advanced DFO methods
  - More objective functions
  - On real BIM/CIM models instead of surface models
- Efficiency
  - Efficient ways of manipulating point clouds (working…)
    - E.g., *k*d-tree, approximate *k*NN, convex hull, planar and object detection
- Extensions
  - Shared component libraries for reusability (e.g., IFC-compatible)
  - Handling other challenging AEC/FM problems

We are still on the way (Source: clipartpanda.com)



The Roman aqueduct Pont du Gard (Source: Wikipedia/ 3Dwarehouse.com)

References

- [1] Volk, R., Stengel, J., and Schultmann, F. (2014). Building Information Modeling (BIM) for existing buildings—Literature review and future needs. Automation in Construction, 38: 109-127.
- [2] National Institute of Building Sciences (NIBS). (2015). National Building Information Modeling Standard. Available at: https://www.nationalbimstandard.org/, retrieved on Mar 8, 2017.
- [3] Patraucean, V. (2016). Deep machine learning: key to the future of BIM? Cambridge: University of Cambridge, Mar 14 2016. Accessed July 15 2016. http://www-smartinfra structure.eng.cam.ac.uk/news/viorica-patraucean-featured-in-infrastructure-intelligence-.
- [4] Eastman, C. M., Teicholz, P., Sacks, R., and Liston, K. (2011). BIM handbook: A guide to building information modeling for owners, managers, designers, engineers and contractors. John Wiley & Sons.
- [5] Belsky, M., Sacks, R., and Brilakis, I. (2016). Semantic enrichment for building information modeling. Computer-Aided Civil and Infrastructure Engineering, 31(4), 261-274.
- [6] Penttilä, H. (2007). Early architectural design and BIM. Computer-Aided Architectural Design Futures (CAADFutures) 2007, 291-302.
- [7] Gröger, G., and Plümer, L. (2012). CityGML–Interoperable semantic 3D city models. ISPRS Journal of Photogrammetry and Remote Sensing, 71, 12-33.
- [8] Venugopal, M., Eastman, C. M., Sacks, R., and Teizer, J. (2012). Semantics of model views for information exchanges using the industry foundation class schema. Advanced Engineering Informatics, 26(2), 411-428.
- [9] Conn, A. R., Scheinberg, K., and Vicente, L. N. (2009). Introduction to derivative-free optimization. MPS-SIAM book series on optimization (Vol. 8). SIAM.
- [10] Auger, A., Finck, S., Hansen, N., and Ros, R. (2010). BBOB 2009: Comparison tables of all algorithms on all noisy functions (PhD thesis, INRIA).
- [11] Hansen, N., and Ostermeier, A. (2001). Completely derandomized self-adaptation in evolution strategies. Evolutionary Computation, 9 (2): 159-195.
- [12] Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing, 13 (4): 600-612.
- [13] libcmaes, an open source library, Version 0.9.5, available at: https://github.com/beniz/libcmaes

Thank You !

HKURBANlab

ilab

THE UNIVERSITY OF HONG KONG 香港大學

faculty of architecture 建築學院