FIG Working Week 2012 - Knowing to Manage the Territory, Protect the Environment, Evaluate the Cultural Heritage

Spatial 3D Analysis of Built-up Areas

Oren Gal, Yerach Doytsher

Mapping and Geo-Information Engineering,
Technion, Israel

1 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

Motivation

- 3D GIS.
- Spatial analysis in 3D urban environments.
- Research focused on visibility in open terrains.
- Visibility Problem – fast and accurate.
- Complex and NP-hard computation in 3D:
  - Computer graphics - fast hidden surface removal.
  - Security - art gallery problem, HLS.
  - Robotics - motion planning.

2 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

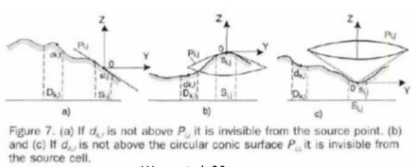
Urban vs. Open Environments

- Open environments can be easily modeled with DEM / DTM.
- Urban environments:
  - Density – computational efficiency.
  - Data with high resolution – complexity.
  - 3D urban model.
  - Different kinds of objects (buildings, roads, vegetation et).
  - Frequently changed (infrastructures).
  - Uncertainly (moving cars / peoples, lighting, weather).

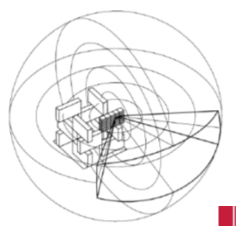
3 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

Previous work

- Viewshed *[Wang et al.96]*
  - Sightline computation and surfaces relationship
    - Limited for DEM models – Open terrains
    - Slow and inefficient
- Openness *[Fisher-Gewirtzman et al. 03]*
  - Urban design environment - DEM
  - Space Openness Index (SOI)



Wang et al. 96



Fisher et al. 2003

4 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

The Problem

- We consider the basic visibility problem in a 3D urban environment, consisting of 3D buildings modeled as 3D box parameterization, $C(\theta,\phi)$ and viewpoint $V(x_0, y_0, z_0)$.

Given:

- A viewpoint $V(x_0, y_0, z_0)$ in 3D coordinates
- Parameterizations of N objects $\sum_{i=1}^{N} C_i(x, y, z = {}^{h_{\max}}_{h_{\min}})$ describing a 3D urban environment model.

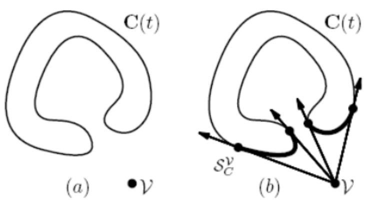
Computes:

- Set of all visible points in $\sum_{i=1}^{N} C_i(x, y, z = {}^{h_{\max}}_{h_{\min}})$ from $V(x_0, y_0, z_0)$.

Concept

- Visibility points can be computed analytically for 2D continuous curve C(t) from viewpoint V:

$$C'(t)\times(C(t)-V)=0$$



Elber et al. 2003

Analytic Solution – Single Object

- Extend this concept for 3D objects:

$$C'(x,y)_{z_{const}} \times (C(x,y)_{z_{const}} - V(x_0, y_0, z_0)) = 0$$

- Solutions to the above equation generate a visibility boundary from the viewpoint to an object.
- Based on basic relations between viewing directions using cross-product characters.

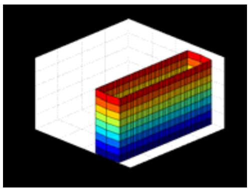
7 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

Building Model - Challenges

- Analytic expressions for visibility points on C
  - Fast Computation.
- Define 3D parameterization for a building
  - Singularity points – approximations.

8 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

3D Building Model



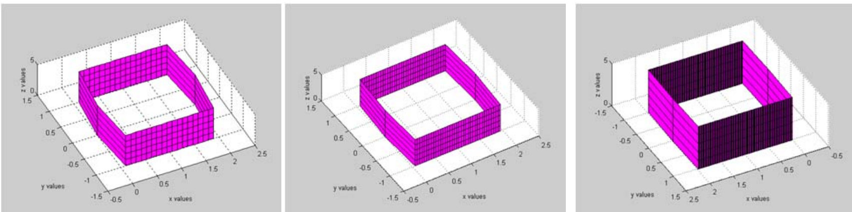
3D Building model

$$x = t,\ y = \begin{pmatrix} x^n - 1 \\ 1 - x^n \end{pmatrix},\ z = c$$

$$-1 \le t \le 1$$

$$n = 350$$

$$c = c + 1$$



(a) (b) (c)

Building models (a) n=50; (b) n=200; (c) n=350

9 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

Analytic Solution – 3D

- Integrating building model and visibility solution:

$$C'(x,y)_{z_{const}} \times (C(x,y)_{z_{const}} - V(x_0, y_0, z_0)) = 0 \rightarrow$$

$$x^n - V_{y_0} - n \cdot x^{n-1}(x - V_{x_0}) - 1 = 0$$

$$x^n + V_{y_0} - n \cdot x^{n-1}(x - V_{x_0}) - 1 = 0$$

$$n = 350$$

$$-1 \le x \le 1$$

10 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

Visibility Computation

- Polynomial equation (n order).
- Visibility boundaries - real roots.
- Faster solution is found by using Matlab zero values algorithm.
- The equations do not depend on Z – roof visibility checking for each object.
- Analytic solution can be found – Visible Boundary Points (VBP).

11 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

Visible Boundary Points (VBP)

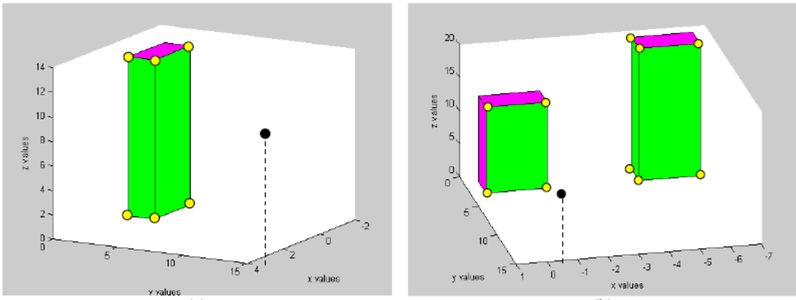
- Based on analytic solution, visibility boundary can be efficiently computed:

Visible Boundary Points (VBP) - we define VBP of the object is as a set of boundary points $j = 1..N_{bound}$ of the visible surfaces of the object, from viewpoint $V(x_0, y_0, z_0)$.

$$VBP_{i=1}^{j=1..N_{bound}}(x_0, y_0, z_0) = \begin{bmatrix} x_1, y_1, z_1 \\ x_2, y_2, z_2 \\ .. \\ x_{N_{bound}}, y_{N_{bound}}, z_{N_{bound}} \end{bmatrix}$$

12 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

Visibility Computation –Example



(a) (b)

Visibility Volume computed with the Analytic Solution. Viewpoint is marked in black, visible parts colored in green, and invisible parts colored in purple. VBP marked with yellow circles - (a) single building; (b) two non-overlapping buildings.

13 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

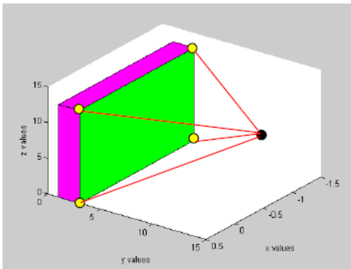
Hidden Surfaces Between Buildings

- Spatial relations between buildings, based on VBP.
- Hidden surfaces between buildings based on intersections of the visible volumes for each object.
- The visible volumes are defined easily using VBP, and are defined, as Visible Pyramids (VP).

14 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

Visible Pyramids (VP)

Visible Pyramid (VP) – we define $VP_i^{j=1..N_{surf}}(x_0, y_0, z_0)$ of the object i as a 3D pyramid generated by connecting VBP of specific surface j to a viewpoint $V(x_0, y_0, z_0)$. Maximum number of for a single object is three.



A Visible Pyramid from a viewpoint (marked as a black point) to VBP of a specific surface

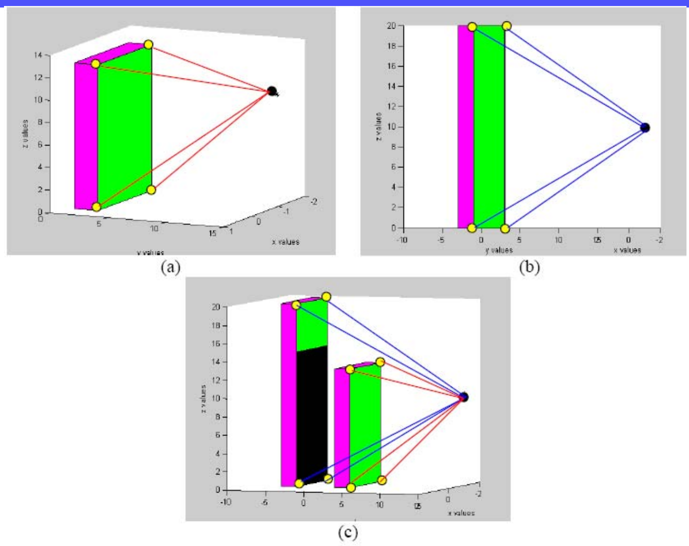
15 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

Hidden Surfaces – VP Intersection

- Invisible parts of the far buildings are computed by:
  - Compute VP for each object.
  - Projection of the closer buildings' VP to the far buildings' VP base.
  - Intersecting the projected part with the far buildings' VP base.

16 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

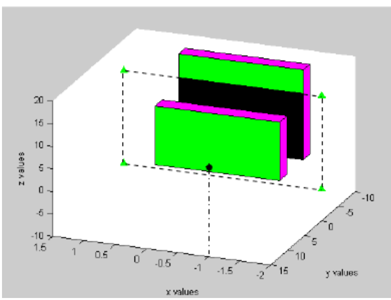
Hidden Surfaces - Demonstration



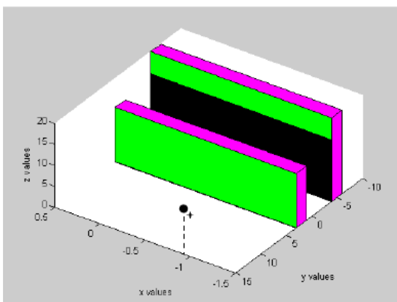
Generating VP – (a) VP_1^1 boundary colored in red arrows; (b) VP_2^1 boundary colored in blue lines; (c) the two buildings - VP_1^1 in red and VP_2^1 in blue, from the viewpoint.

17 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

Hidden Surfaces Between Buildings



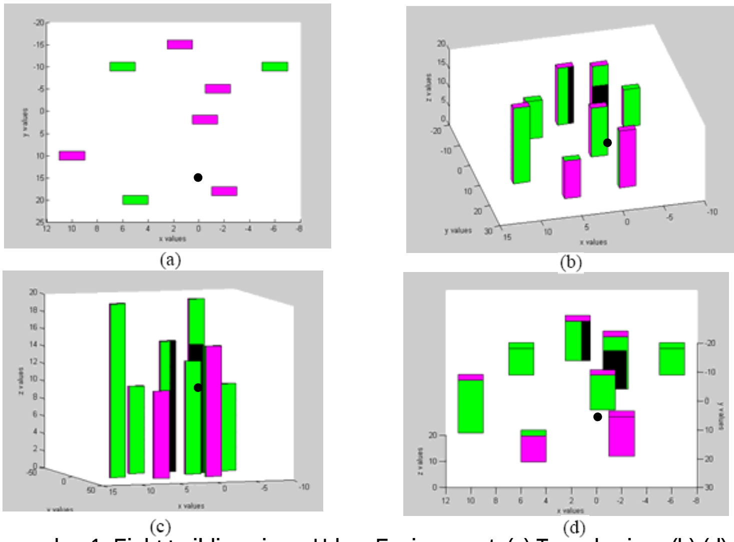
Projection of VP_1^1 to VP_2^1 base plane marked with dotted lines and green triangles



Computing Hidden Surfaces between Buildings by Using the Visible Pyramid colored in black on VP_2^1 Base Plane.

18 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

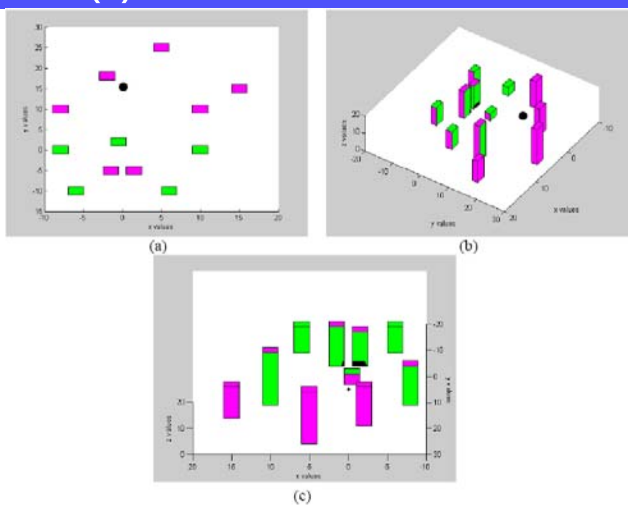
Simulations (1)



Scene number 1: Eight buildings in an Urban Environment, (a) Topside view; (b)-(d) Different views demonstrating the visibility computation using our algorithm. CPU time was 0.18 sec.

19 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

Simulations (2)



Scene number 2: Twelve Buildings in an Urban Environment, where viewpoint is higher than the projected building, (a) Topside view; (b)-(c) Different views demonstrating visibility computation using our algorithm. CPU time was 0.29 sec.

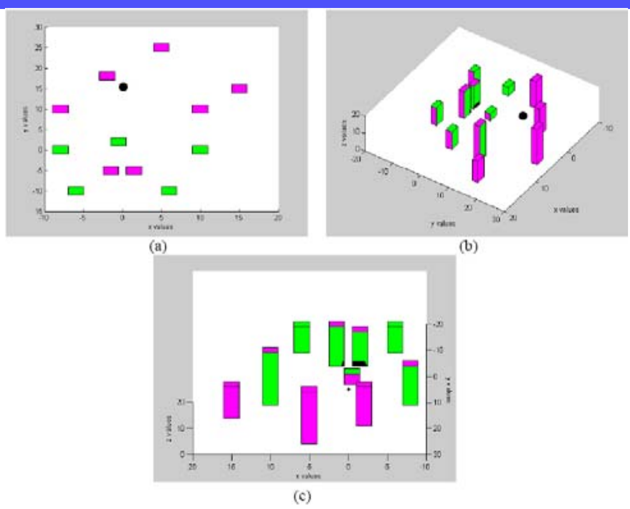
20 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

LOS vs. Proposed Method

- We compare computation time between LOS method and our analytic solution.
- LOS visibility based on line between two point, without other object's points intersection.
- Using *LOS2* Matlab function.
- Scene comparison – scene no. 2.

21 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

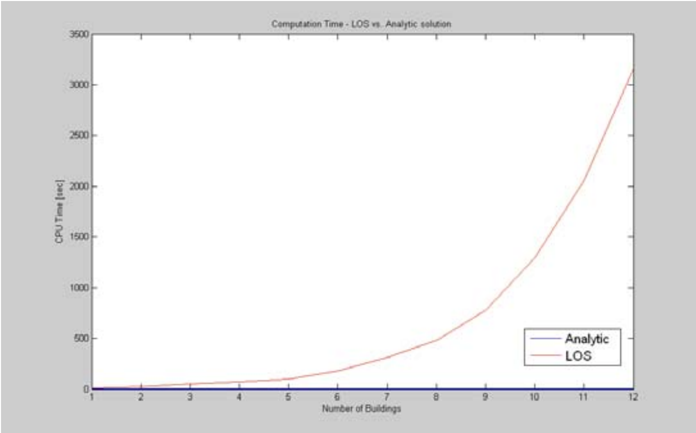
Scene Num. 2



Scene number 2: Twelve Buildings in an Urban Environment, where viewpoint is higher than the projected building, (a) Topside view; (b)-(c) Different views demonstrating visibility computation using our algorithm. CPU time was 0.29 sec.

22 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

Computation Time Comparison



CPU Computation Time of LOS and our algorithm. CPU was measured in the second scene with an increasing number of buildings from one to twelve.
LOS method was 1000 times longer than our algorithm.

23 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

Conclusions

- Efficient algorithm for visibility computation in a built-up environment.
- Modeling basic building structure with mathematical approximating for presentation of buildings' corners.
- Computing hidden surfaces between buildings by using projected surfaces and intersections of the visible pyramids.
- Does not require special hardware, and is suitable for on-line computations.

24 FIG Working Week 2012, Rome, Italy, 6-10/5/2012

Future Work

- Modeling more complex urban environments.
- Generalizing the presented building model.
- Testing algorithm in more complex environments.
- Optimal visibility computation – multi viewpoints.

Thank You