

## AUTOMATIC GENERATION OF 3-D ANCIENT BUILDING MODELS BASED ON A DIGITAL MAP

Kenichi Sugihara

Dept. of Information Media, Gifu Keizai University, 5-50 Kitagata-chou Ogaki-city Gifu-Pref, Japan  
sugihara@gifu-keizai.ac.jp

Commission VI, WG VI/4

**KEY WORDS:** Automatic Generation, GIS, CG, 3-D ancient building model, 3-D city model, Pagoda, Ancient gate

### ABSTRACT:

3-D ancient city models including temples, pagodas and ancient gate are important in archaeological research and in facilitating “Public Involvement”. In order to carry out public projects smoothly or to encourage public participation, it is necessary to disclose information about public projects such as archaeological excavations or urban planning to the public. In Japan, public agencies publicize the results of excavations or urban planning by showing floor plans and side views and by explaining with papers difficult to understand. To facilitate “Public Involvement”, 3-D models simulating a real city or an ancient city by a 3-D CG can be of great use. However, enormous time and labour has to be consumed to create these 3-D models, using 3D modelling software such as 3ds Max or SketchUp. For example, when manually modelling a temple by Constructive Solid Geometry (CSG), one must follow these laborious modelling steps: **1)** generation of primitives of appropriate size, such as box, prism or polyhedron that will form parts of a building **2)** Boolean operation among these primitives to form the shapes of parts of a building such as making holes in a building body for doors and windows **3)** rotation of parts of a building **4)** positioning of parts of a building **5)** texture mapping onto these parts.

In order to automate these laborious steps, we are proposing the GIS and CG integrated system that automatically generates 3-D building models from building polygons (building footprints) on a digital map. Since most building polygons’ edges meet at a right angle (orthogonal polygon), a complicated building polygon can be partitioned into a set of rectangles. The integrated system partitions orthogonal building polygons into a set of rectangles and places rectangular roofs and box-shaped building bodies on these rectangles. In order to partition an orthogonal polygon, we proposed a useful polygon expression (RL expression) and a partitioning scheme that is used in deciding from which vertex a dividing line (DL) is drawn. In this paper, we propose a new scheme for partitioning building polygons. We also show the process of creating a basic gable roof model and a complicated temple top roof.

## 1. INTRODUCTION

### 1.1 General Introductions

Based on building polygons or building footprints on digital maps or an ortho image shown in Figure 1, we propose a GIS and CG integrated system that automatically generates 3-D building models. A 3-D urban model as shown in Figure 2 is an important information infrastructure that can be utilized in several fields, such as, urban planning and landscape evaluation,

and archaeological research by 3-D models for restoration of ancient cities.

However, enormous time and labour has to be consumed to create these 3-D models, using 3D modelling software such as 3ds Max or SketchUp. For example, when manually modelling a house with roofs by Constructive Solid Geometry (CSG), one must follow these laborious modelling steps:

**1)** generation of primitives of appropriate size, such as box, prism or polyhedron that will form parts of a house **2)** Boolean



Figure 1. Building polygons on an ortho image



Figure 2. An automatically generated 3-D urban model

operation among these primitives to form the shapes of parts of a house such as making holes in a building body for doors and windows **3)** rotation of parts of a house **4)** positioning of parts of a house **5)** texture mapping onto these parts.

In order to automate these laborious steps, we proposed the GIS and CG integrated system that automatically generates 3-D building models from building polygons on a digital map (Sugihara, 2005). As shown in Figure 1, most building polygons' edges meet at a right angle (orthogonal polygon). A complicated orthogonal polygon can be partitioned into a set of rectangles. The integrated system partitions orthogonal building polygons into a set of rectangles and places rectangular roofs and box-shaped building bodies on these rectangles. In order to partition an orthogonal polygon, we proposed a useful polygon expression (RL expression) and a partitioning scheme that is used in deciding from which vertex a dividing line (DL) is drawn (Sugihara, 2006). In this paper, we propose a new scheme for partitioning building polygons. We also show the process of creating a basic gable roof model and a complicated temple top roof.

## 1.2 Related Work

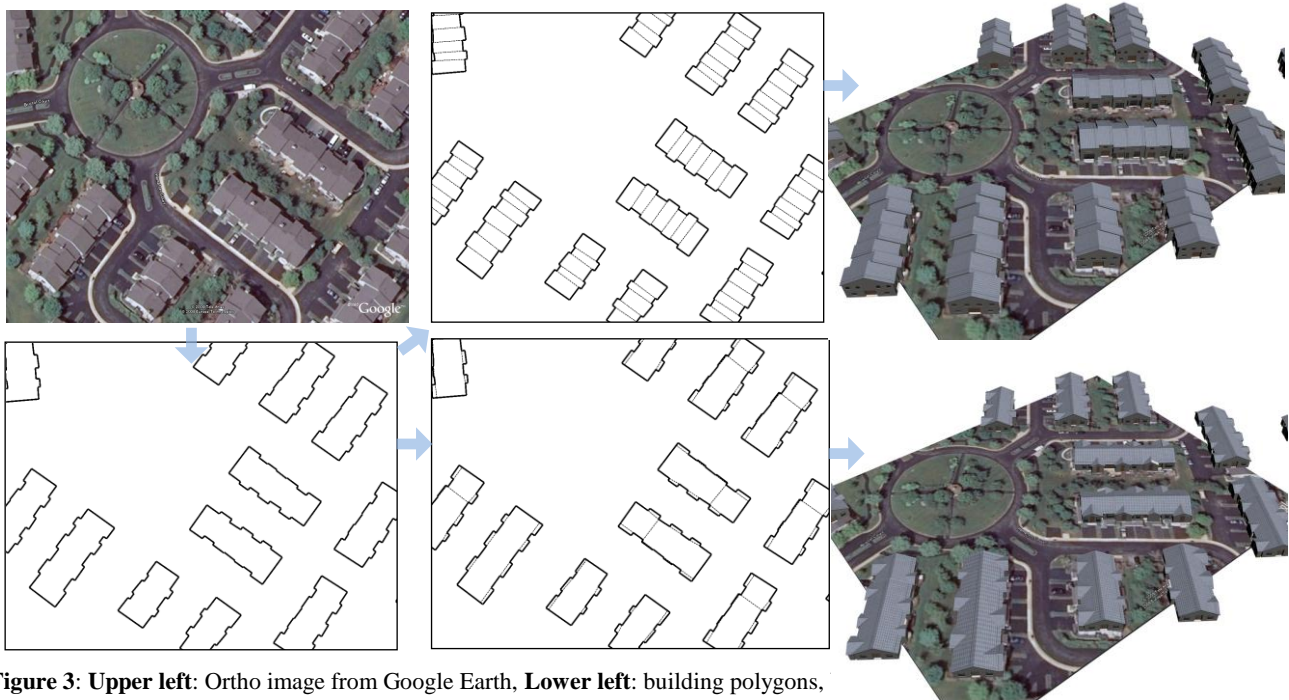
Since 3-D urban models are important information infrastructure that can be utilized in several fields, the researches on creations of 3-D urban models are in full swing. Procedural modelling is an effective technique to create 3-D models from sets of rules such as L-systems, fractals, and generative modelling language.

Müller et al.(2006) have created an archaeological site of Pompeii and a suburbia model of Beverly Hills by using a shape grammar with production rules. They import data from a GIS database and try to classify imported mass models as basic shapes in their shape vocabulary. If this is not possible, they use a general extruded footprint together with a general roof obtained by a straight skeleton computation (Aichholzer et al.,1995).

The straight skeleton can be used as the set of ridge lines of a building roof, based on walls in the form of the initial polygon (Aichholzer et al.,1996). However, the roofs created by the straight skeleton are limited to hipped roofs or gable roofs with their ridges parallel to long edges of the rectangle into which a building polygon is supposed to be partitioned, since shorter edges disappear while longer edges remain in shrinking process. There are many roofs whose ridges are vertical to a long edge of the rectangle as shown in an ortho image of Figure 3, and these roofs cannot be created by the straight skeleton. The straight skeleton treats a building polygon as a whole and forms a seamless roof so that it cannot place individual roof independently on partitioned polygons. The 3-D roofs shown in lower right of Figure 3 can be created by the straight skeleton. Figure 3 also shows the different partitioning schemes (separation prioritizing or shorter DL prioritizing) which are selected by an attribute data of the building polygon manually stored at first in our system. To create a various shape of 3-D roofs, building polygons are to be partitioned into sets of individual rectangles.

Laycock et al. (2003) have combined the straight skeleton method and polygon partitioning in the following steps; **1)** Partition the polygon into a set of rectangles by horizontal and vertical lines from all reflex vertices. **2)** Construct the straight skeleton and grow an axis aligned rectangle (AAR) from each of the lines. **3)** For each AAR, collect the rectangles which are interior to AAR and union them to obtain an exterior boundary. **4)** Assign a roof model to each exterior boundary. Merge the roof models.

This method seems effective in independently choosing a roof model for each rectangle and merging the roof models for polygons with a small number of vertices. However, for polygons with a large number of vertices, implementation of partitioning along all DLs (dividing lines from all reflex vertices) often results in an unnecessarily large number of rectangles and collecting and merging steps become so cumbersome that they don't succeed in doing this.



**Figure 3:** **Upper left:** Ortho image from Google Earth, **Lower left:** building polygons, **middle:** Separation prioritizing scheme partitions into a set of rectangles, **L middle:** Shorter DL prioritizing scheme, **U Right:** Ridges of roofs vertical to a long edge, **L Right:** 3-D building models by straight skeleton



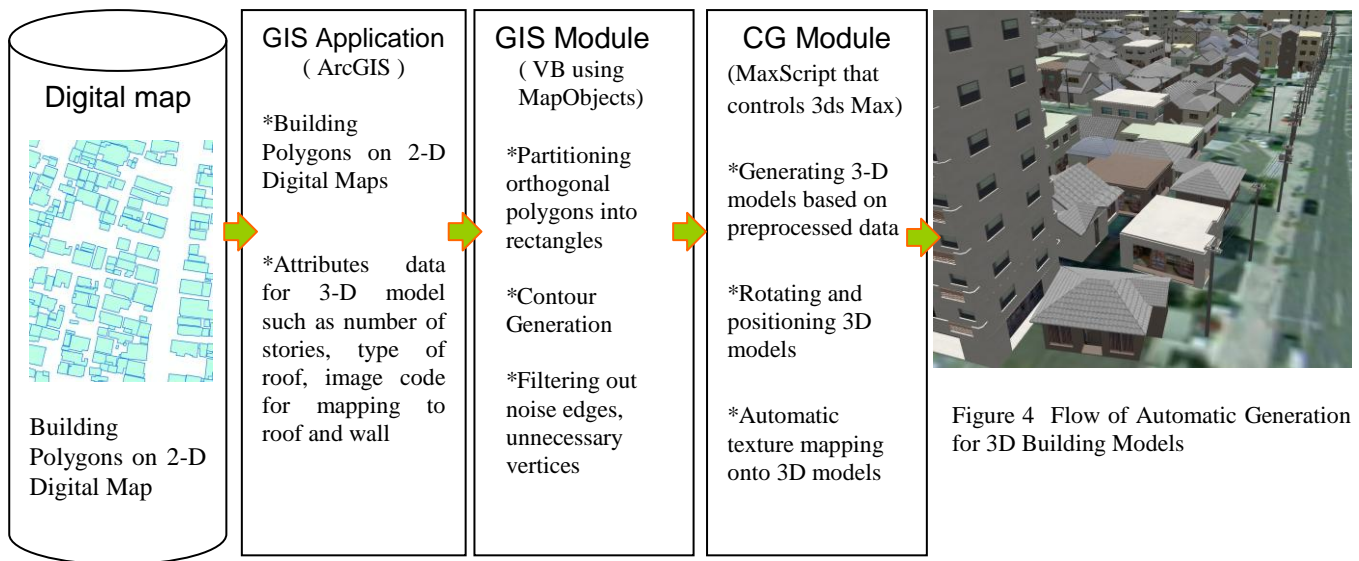


Figure 4 Flow of Automatic Generation for 3D Building Models

In our system, one has an option to choose partitioning scheme; prioritizing separation or prioritizing shorter DL. Our system tries to select a suitable DL for partitioning or a suitable separation, depending on the RL expression of a polygon, the length of DLs and the edges of a polygon.

### 1.3 Flow of Automatic Generation

The automatic generation system consists of GIS application (ArcGIS, ESRI Inc.), GIS module and CG module as shown in Figure 4. The source of a 3-D urban model is a digital map that contains building polygons linked with attributes data such as the number of stories and the type of roof. The GIS module pre-processes building polygons on the digital map. Pre-process includes filtering out a ‘short edge’ (a short edge that is between long edges of almost the same direction) and unnecessary vertices of a building polygon, partitioning orthogonal building polygons into sets of rectangles, generating inside contours for positioning walls and glasses of a building and exporting the coordinates of polygons’ vertices and attributes of buildings. The attributes of buildings consist of the number of stories, the image code of roof, wall and the type code of roof (flat, gable roof, hipped roof, oblong gable roof, gambrel roof, Mansard roof and so forth). The GIS module has been developed using 2-D GIS software components (MapObjects, ESRI).

The CG module receives the pre-processed data that the GIS module exports, generating 3-D building models. CG module has been developed using Maxscript that controls 3-D CG

software (3ds Max, Autodesk Inc). In case of modelling a building with roofs, the CG module follows these steps:

- 1) generation of primitives of appropriate size, such as boxes, prisms or polyhedra that will form the various parts of the house
- 2) Boolean operation on these primitives to form the shapes of parts of the house, for examples, making holes in a building body for doors and windows
- 3) rotation of parts of the house
- 4) positioning of parts of the house
- 5) texture mapping onto these parts according to the attribute received
- 6) copying the 2nd floor to form the 3rd floor or more in case of building higher than 3 stories.

## 2. POLYGON PARTITIONING

### 2.1 Proposed Polygon Expression

At map production companies, technicians are drawing building polygons manually with digitizer, depending on aerial photos or satellite imagery as shown in Figure 5. This aerial photo and digital map (Figure 1) also show that most building polygons are orthogonal polygons. An orthogonal polygon can be replaced by a combination of rectangles. When following edges of a polygon clockwise, an edge turns to the right or to the left by 90 degrees. Therefore, it is possible to assume that an orthogonal polygon can be expressed as a set of its edges’ turning direction.

We proposed a useful polygon expression (RL expression) in specifying the shape pattern of an orthogonal polygon

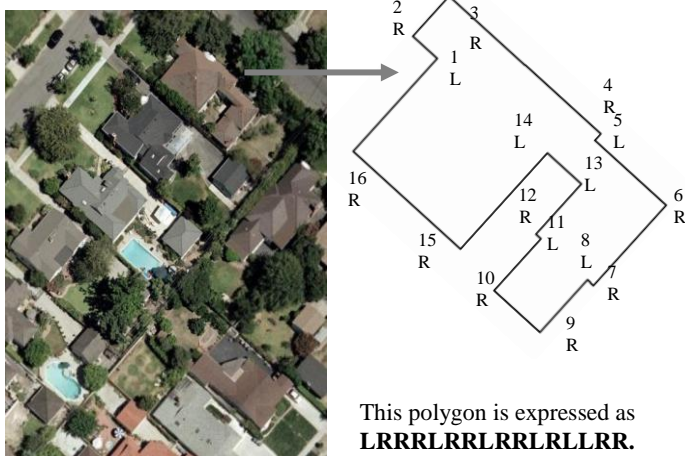


Figure 5 Satellite imagery and building polygon

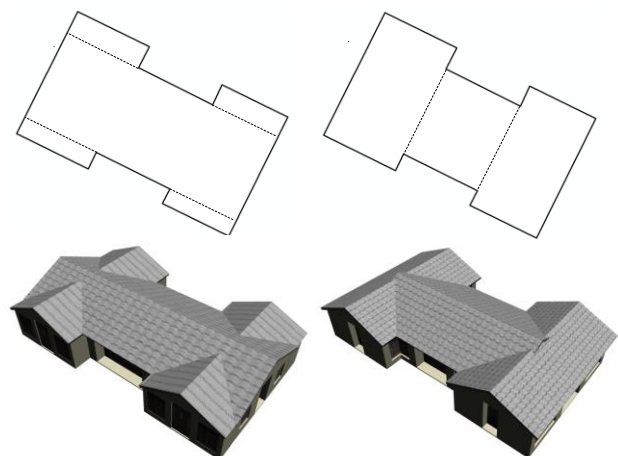


Figure 6 Polygons and 3-D house models by different partitioning schemes

(Sugihara,2005). For example, a polygon with 16 vertices shown in Figure 5 is expressed as a set of its edges' turning direction; LRRRLRRLRRLRLLRR where R and L mean a change of an edge's direction to the right and to the left, respectively. The number of shapes that a polygon can take depends on the number of vertices of a polygon.

The advantage of this RL expression is as follows.

- (1) RL expression specifies the shape pattern of an orthogonal polygon without regard to the length of its edges.
- (2) This expression decides from which vertex a dividing line (DL) is drawn.

## 2.2 Partitioning Scheme

The more vertices a polygon has, the more partitioning scheme a polygon has, since the interior angle of a 'L' vertex is 270 degrees and two DLs(dividing lines) can be drawn from a 'L' vertex.

We proposed the partitioning scheme that gives higher priority to the DLs that divide 'fat rectangles' (Sugihara,2006). A 'fat rectangles' is a rectangle close to a square. Our proposed partitioning scheme is similar to Delaunay Triangulation in the sense that Delaunay Triangulation avoids thin triangles and generates fat triangles. However, our proposal did not always result in generating plausible and probable 3-D building models with roofs. For example, the right model of Figure 6 generated by 'prioritizing fat rectangle' scheme indicates that a rectangle cut off by a DL is wider than a body rectangle, resulting in that a branch roof is higher than a main roof, where a 'branch roof' is a roof that is cut off by a DL and extends to a main roof. A 'main roof' is a roof that is extended by a branch roof.

In our new proposal, among many possible DLs, the DL that satisfies the following conditions is selected for partitioning.

- (1) A DL that cuts off 'one rectangle'.
- (2) Among two DLs from a same 'L' vertex, a shorter DL is selected to cut off a rectangle.
- (3) A DL whose length is shorter than the width of a 'main roof' that a 'branch roof' is supposed to extend to.

Our newly proposed partitioning scheme as shown in the left of

Figure 6 leads to a plausible and probable 3-D house model: a main roof is higher than a branch roof.

## 2.3 Partitioning Process

Figure 7 shows the partitioning process of an orthogonal building polygon into a set of rectangles. Stage 2 in Figure 7 shows an orthogonal polygon with all possible DLs shown as thin dotted lines and with DLs that satisfy condition (1), shown as thick dotted lines. The example of a branch roof is shown as the rectangle formed by vertices 6,7,8,9 cut off by DL.

The reason we set up these conditions is that like breaking down a tree into a collection of branches, we will cut off along 'thin' part of branches of a polygon. Therefore, we propose a scheme of prioritizing the DL that cuts off a branch roof, based on the length of the DL. Since each roof has the same slope in most multiple-roofed buildings, a roof of longer width is higher than a roof of shorter width and 'probable multiple-roofed buildings' take the form of narrower branch roofs diverging from a wider and higher main roof. Narrower branch roofs are formed by dividing a polygon along a shorter DL and the width of a branch roof is equal to the length of the DL.

In the partitioning process as shown in Figure 7, the DLs that satisfy the mentioned conditions are selected for partitioning. By cutting off one rectangle, the number of the vertices of a body polygon is reduced by two or four. After partitioning branches, the edges' lengths and RL data are recalculated to find new branches. Partitioning continues until the number of the vertices of a body polygon is four. After being partitioned into a set of rectangles, the system places 3-D building models on these rectangles.

## 2.4 How to partition branches

The vertices of a polygon are numbered in clock-wise order as shown in Figure 7. Here, how the system is finding 'branches' is as follows. The system counts the number of consecutive 'R' vertices (=  $n_R$ ) between 'L' vertices. If  $n_R$  is two or more, then it can be a branch. One or two DLs can be drawn from 'L' vertex in a clockwise or counterclockwise direction, depending on the length of the incident edges of 'L' vertex.

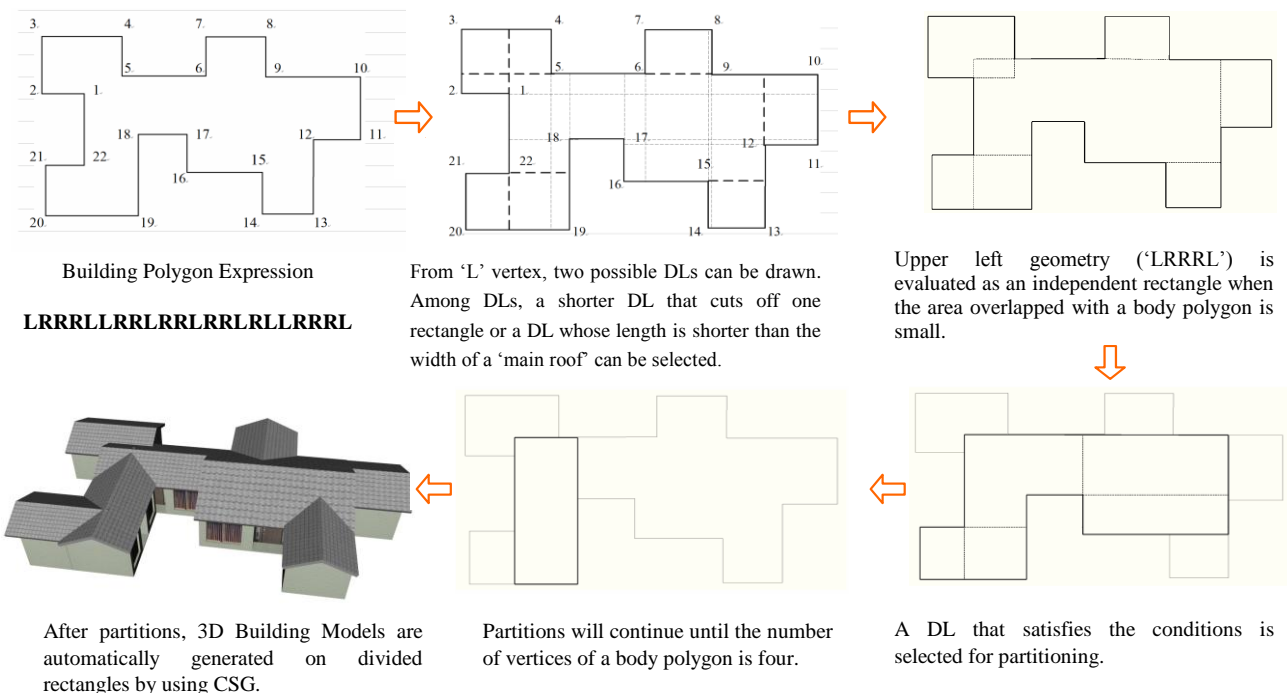
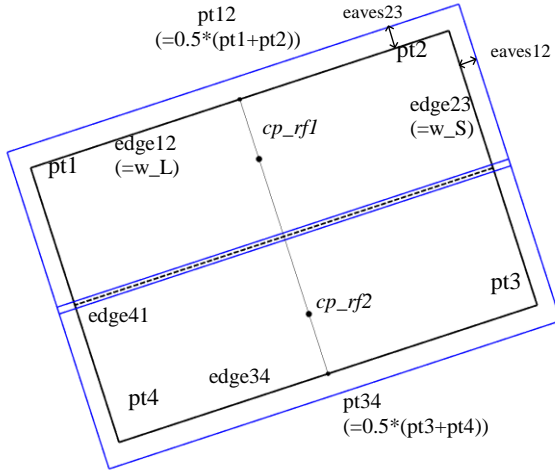


Figure 7 Partitioning process of orthogonal building polygon into a set of rectangles



$$\text{ratio}_s = 0.25 - \frac{0.5 \times (\text{eaves23} \times \cos \theta + \text{rf\_offs} \times \sin \theta)}{w_S} + \frac{\text{thick\_rf} \times \sin \theta}{w_S}$$

$$\text{cp\_rf1} = (1.0 - \text{ratio}_s) \times \text{pt12} + \text{ratio}_s \times \text{pt34}$$

$$\text{cp\_rf2} = \text{ratio}_s \times \text{pt12} + (1.0 - \text{ratio}_s) \times \text{pt34}$$

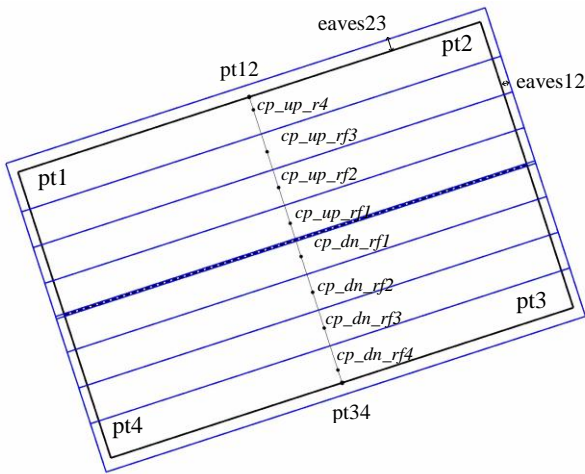
Here, 'thick\_rf' is a thickness of a roof board. 'eaves23' is the length of eaves in a direction of edge23.  $\theta$  is an angle of a roof slope with a horizontal plane. 'rf\_offs' is the offset of a roof board from a prism as shown Figure 9.

Figure 8 Ground plan of a gable roof and explanation of parameters of a gable roof

### 3. HOW TO CREATE ROOF MODELS

The integrated system partitions orthogonal building polygons into a set of rectangles and places rectangular roofs and box-shaped building bodies on these rectangles. The positioning of parts of a building is implemented in following steps. After measuring the length and the direction of the edges of the partitioned rectangle, the edges are categorized into a long edge and a short edge. The vertices of the rectangle are numbered clockwise with the upper left vertex of a long edge being numbered 'pt1' as shown in Figure 8.

In 3ds Max we use for the creation of 3D models, each building part or primitive has its own control point ('cp') and local

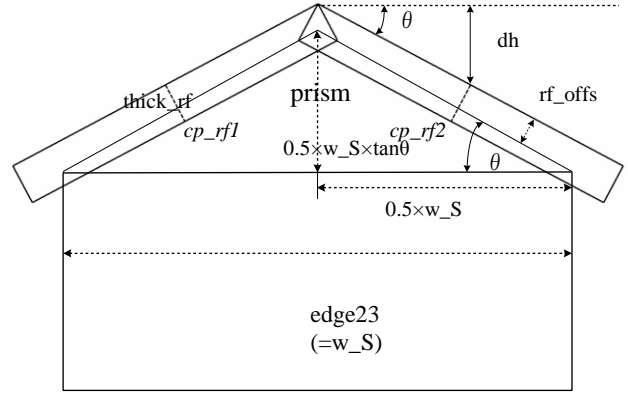


$$\text{ratio}(i) = \frac{n\_div + i}{2 \times n\_div} - \frac{0.5 \times (\text{eaves23} \times \cos \theta + \text{rf\_offs} \times \sin \theta)}{w_S} + \frac{\text{thick\_rf} \times \sin \theta}{w_S}$$

$$\text{cp\_dn(up)\_rf}(i) = (1.0 - \text{ratio}(i)) \times \text{pt12}(\text{pt34}) + \text{ratio}(i) \times \text{pt34}(\text{pt12})$$

$n\_div$  is the number of roof boards into which a roof is divided into.  $\text{cp\_dn\_rf}(i)$  and  $\text{cp\_up\_rf}(i)$  are  $i$ -th control points of thin roof boards.

Figure 11 Ground plan of a temple top roof



The width of a roof board is as follows.  
 $\text{wid\_rfb} = \text{side23L} + \text{eaves23} + \text{rf\_offs} \times \tan \theta$   
 Here,  $\text{side23L} = 0.5 \times w_S \times \sqrt{1 + \tan^2 \theta}$

The height of a roof board is as follows.

$$\text{hei\_rf} = \text{st\_heit} - 0.5 \times (\text{side23L} + \text{eaves23} + \text{rf\_offs} \times \tan \theta) \times \sin \theta - \text{thick\_rf} \times \cos \theta + \text{rf\_offs} / \cos \theta + 0.5 \times \tan \theta \times w_S$$

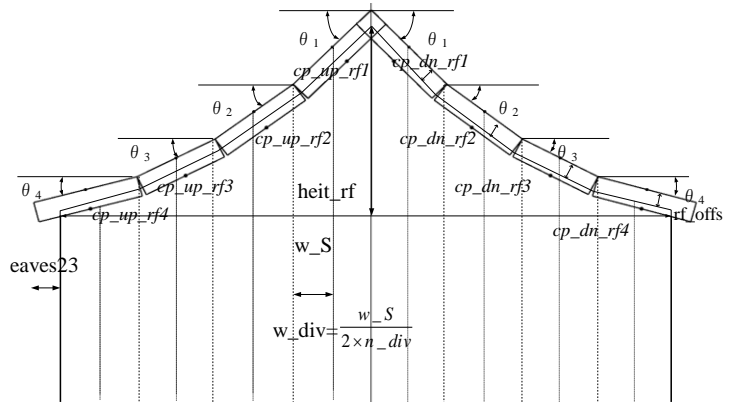
Here, 'st\_heit' is start height as follows.

$$\text{st\_heit} = (\text{floor-to-floor height}) \times (\text{the number of stories})$$

Figure 9 Front view of a gable roof and explanation of parameters of a gable roof

coordinates that control its position and direction. The position of a 'cp' is different in each primitive. For placing building parts properly, their 'cp's are placed at the point that divides edge12 and edge23 at an appropriate ratio. For example, a prism is used for the construction under roof boards. The 'cp' of a prism lies in one of the vertex of the base triangle in an upright position when a prism is newly created.

The top of a gable roof consists of two roof boards (two thin boxes). Since the 'cp' of a box lies in a center of a base, it is placed on the point that divides the line through pt12 and pt34 at the ratio shown in ground plan (Figure 9). The height of the 'cp's of two roof boards is shown in the front view of a gable roof (Figure 10).



The gradient of each thin roof boards  $\theta_i$  is  
 $\theta_i = \arctan \frac{\text{heit\_rf} \times (\text{rat\_heit}(i) - \text{rat\_heit}(i-1))}{w_S \div n\_div}$

Here,  $\text{rat\_heit}(i)$  is the  $i$ -th ratio to 'heit\_rf' that is given by sampling catenary at a constant interval.

The height of  $i$ -th roof board ( $\text{heit\_rfb}(i) : i=1$  to  $3$ ) is as follows.  $i=4$ ; the width of the roof board is longer by eaves23.

$$\text{heit\_rfb}(i) = 0.5 \times \text{heit\_rf} \times (\text{rat\_heit}(i) - \text{rat\_heit}(i-1)) + (\text{rf\_offs} - \text{thick\_rf}) \times \cos \theta_i$$

$$\text{heit\_rfb}(4) = \text{heit\_rf} \times \text{rat\_heit}(4) - 0.5 \times (w\_div \div \cos \theta_4 + \text{eaves23}) \times \sin \theta_4 + (\text{rf\_offs} - \text{thick\_rf}) \times \cos \theta_4$$

Figure 12 Front view of a temple top roof



Also, the other complicated shapes of roofs, such as a temple roof, consist of many thin roof boards. Figure 14 and Figure 15 show automatically generated 3-D models with complicated shapes of roofs restoring a Japanese ancient temple and a pagoda. The 'cp' s of thin roof boards consisting of a temple top roof are also placed on the point that divides the line through pt12 and pt34 at the ratio shown in ground plan (Figure 11). The front view of a temple top roof (Figure 12) reveals the height of the 'cp' s of thin roof boards.

#### 4. APPLICATION AND CONCLUSION

Here are examples of 3-D models created by the automatic generation system. Figures from 13 to 18 show photos and automatically generated models of ancient gate and temple.

For residents, citizens or even students as well as the researchers, a 3-D urban model is quite effective in understanding what was built, what image of the town were or what if this alternative plan is realized. This model can act as a simulator to realize alternative ideas of ancient building reconstruction or urban planning virtually.

In this paper, we propose a new scheme for partitioning building polygons and show the process of creating a basic gable roof model. By applying polygon partitioning algorithm, the system divides polygons along the thin parts of its branches. We also present 3-D models restoring a Japanese ancient temple and a pagoda, based on the generation process of a basic gable roof.

Future work will be directed towards the development of methods for:

- 1) the creation of general shape of roofs by a straight skeleton computation based on general shape of building polygons.
- 2) 3D reconstruction algorithm to generate any objects in 3-D urban model by using Computer Vision that defines the geometry of the correspondences between multiple stereo views and leads to 3D reconstruction.

#### References from Journals:

Aichholzer, Oswin; Aurenhammer, Franz; Alberts, David; Gärtner, Bernd (1995) A novel type of skeleton for polygons, *Journal of Universal Computer Science* 1 (12): 752–761.

Müller, P., Wonka, P., Haegler, S., Ulmer, A., Luc van Gool, L. V. (2006) Procedural modeling of buildings, *ACM Transactions on Graphics* 25, 3, 614–623.

Parish, Y. I. H., and Müller, P. (2001) Procedural modeling of cities In *Proceedings of ACM SIGGRAPH 2001*, ACM Press, E. Fiume, Ed., New York, 301–308.

#### References from Other Literature:

Aichholzer, Oswin; Aurenhammer, Franz (1996) Straight skeletons for general polygonal figures in the plane, *Proc. 2nd Ann. Int. Conf. Computing and Combinatorics (COCOON '96)*: 117–126

R. G. Laycock and A. M. Day (2003) Automatically Generating Roof Models from Building Footprints, *WSCG posters proceedings*, 346-351

Kenichi SUGIHARA (2005) Automatic Generation of 3-D Building Model from Divided Building Polygon, *ACM SIGGRAPH 2005, Posters Session, Geometry & Modeling*

Kenichi SUGIHARA (2006) Generalized Building Polygon Partitioning for Automatic Generation of 3-D Building Models, *ACM SIGGRAPH 2006, Posters Session Virtual & Augmented & Mixed Reality & Environments*



Figure 13 Photo of the south gate of the Heijo shrine, called Suzaku mon

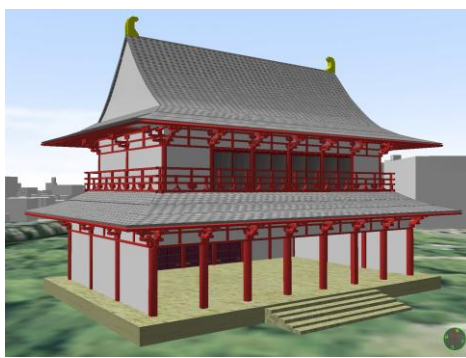


Figure 14 Automatically generated model of ancient gate called Suzaku mon



Figure 15 The parameters (the slope of roofs, the standard interval of pillars and so on) are changed



Figure 16 Photo and automatically generated model of five stories pagoda



Figure 17 Photo of miniature model of the Mino Kokubunji temple

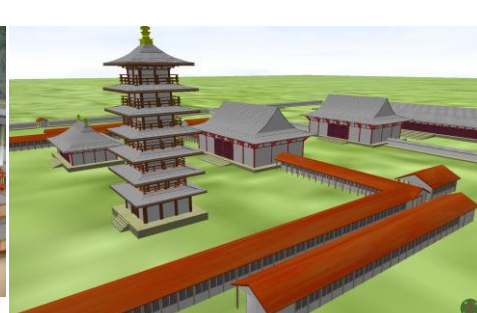


Figure 18 Automatically generated model of the Mino Kokubunji temple