# Study on Pseudo Update of Building Shape in Road Ledger Digital Map

TSUYOSHI TAKAYAMA[1], AKITSUGU OKI[2], HIDEMI FUKADA[3],
YOSHITOSHI MURATA[1], NOBUYOSHI SATO[1], and TETSUO IKEDA[4]
[1] Graduate School of Software and Information Science
Iwate Prefectural University
152-52, Sugo, Takizawa, Iwate020-0193
JAPAN
{takayama, y-murata, nobu-s}@iwate-pu.ac.jp http://www.iwate-pu.ac.jp/
[2] Software Development Center
FUJIFILM Corporation
798, Miyadai, Kaisei, Ashigarakami, Kanagawa258-8538
JAPAN
http://www.fujifilm.co.jp/corporate/aboutus/offices/index.html
[3] Information System Division, General Affairs Department
Morioka Municipal Government
12-2, Uchimaru, Morioka, Iwate020-8530
JAPAN
fukada@city.morioka.iwate.jp http://www.city.morioka.iwate.jp/
[4] School of Administration and Informatics
University of Shizuoka
52-1, Yada, Suruga, Shizuoka, Shizuoka422-8526
JAPAN
t-ikeda@u-shizuoka-ken.ac.jp http://www.u-shizuoka-ken.ac.jp/

*Abstract:* - Recently, in the area of GIS(Geographic Information System), lots of researchers are attracted to efficient update of a digital map data. However, its update cost in the conventional methods is not low. On the other hand, in the case of carrying out some works in a municipal government by employing GIS, the following problem occurs. That is, it is not easy to carry out them because the shape or 'existence or not' of a building on a base map of road ledger generated from GIS is not always relevant to its truth. In our discussion, we limit the scope to building shape. The purpose of our research is to develop an efficient and available update method of building shape with satisfying the following two conditions: (i) it does not contain a data structuring phase from raster images and the update cost in it is not too high, and (ii) it use a map updated more frequently than the basic map which should be updated, and is practically available. Authors investigate semi-automatic and pseudo update of a building by employing an urban design map which has higher update frequency but less precision than the base map of road ledger. In here, 'pseudo' means that we put the major interest to carry out the municipal government works and provide the first priority with the relevance to the truth on the shape or 'existence or not' of a building, even if we sacrifice its precision of the map. In the present paper, we propose its concrete algorithm and evaluate it. According to the evaluation, our proposition has achieved 85.0 % recall and 91.9 % precision. Our proposition has also obtained good result in qualitative evaluation.

*Key-Words:* - GIS(Geographic Information System), update, frequency, recall, precision, evaluation, and municipal government work.

## 1 Introduction

Recently, lots of researchers are attracted to investigating GIS(Geographic Information System)[1]-[3]. Among them, efficient update of a digital map data is one of the important areas. The conventional methods include the one which employs newly taken aerial or satellite images. Klang *et al.* try automatic detections of changes in road database using satellite imagery[4]. Watanabe *et al.* detect building changes using epipolar constraint from aerial images taken at different positions[5]. Nakagawa *et al.* propose an algorithm which detects building change by employing aerial imagery and precise 3-D data[6]. Liu *et al.* detects building change using UltraCamD images which is large format area sensor and existing CAD data[7]. However, the update cost in these methods is not low mainly due to the data structuring phase from these images[8].

Another direction of the conventional method overlaps the information obtained by a field test to an older map, finds some changes, and updates them[6]. However, if we try to detect all of building changes by the field test, we need lots of time and effort.

Yamazawa *et al.* try a method which positions a 1/2500 digital map to a basic one and update it by employing 1/500 digital one[9]. However, we can not say it is practical, because an update frequency of 1/500 digital map is usually lower than 1/2500 one. As a result, even if we update 1/2500 map by employing 1/500 map, we can use it only until next update time of the 1/2500 map(Fig.1).

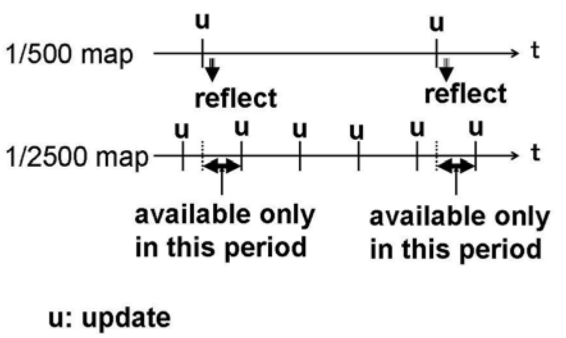


Fig.1 Map update by employing another map with lower update frequency.

If we take into account only the update frequency, it is more practical that we position a 1/500 digital map to a basic one, and update it by employing 1/2500 one.

On the other hand, in the case of carrying out some works in a municipal government by employing GIS, the following problem occurs. That is, it is not easy to carry out them because the shape or 'existence or not' of a building on a base map of road ledger generated from GIS is not always relevant to its truth.

The purpose of our study is to develop an efficient and available update method of building shape with satisfying the following two conditions: (i) it does not contain a data structuring phase from raster images and the update cost in it is not too high, and (ii) it use a map updated more frequently than the basic map which should be updated, and is practically available.

In our discussion, we limit the scope to update of building shape. The present paper tries semi-automatic and pseudo update of a building by employing an urban design map which has higher update frequency but less precision than the base map of road ledger. In here, 'pseudo' means that we put the major interest to carry out the municipal government works and provide the first priority with the relevance to the truth on the shape or 'existence or not' of a building, even if we sacrifice its precision of the map. We also try to reduce the loss of precision in minimum.

The rest of the present paper is organized as follows. In the next section, we describe our proposition. It includes (i) what we should provide for update as input, and (ii) the algorithm for obtaining the updated data. Section 3 introduces our pilot system, and Section 4 is the evaluation of our proposition. Finally in section 5, we conclude the discussion.

# 2 Proposition

In our discussion, we employ 1/500 Shape file which is representative for road ledger map, and 1/2500 DM data representative for urban design map. Hereafter, we use the following notations:

- *1/500Shape(t)*: 1/500 Shape file at the time *t*
- *1/2500DM(t+dt)*: 1/2500 DM data at the time (*t+dt)*
- *1/500 Shape(t+dt)*: 1/500 Shape file at the time (*t+dt).*

We show our proposition of pseudo update process in Fig.2. Its detail is described in the following subsections.

## 2.1 Preprocess Phase

The format of *1/2500DM(t+dt)* is different from the one of *1/500Shape(t)*. Therefore, we put preprocess phase in order to enable to mutually compare with each building shape.

## 2.2 Pseudo Update Phase

We mutually compare each building shape of *1/500Shape(t)* with the one of *1/2500DM(t+dt)* in the same area, and determine whether it should be updated or not. We update only the buildings which should be positively updated.

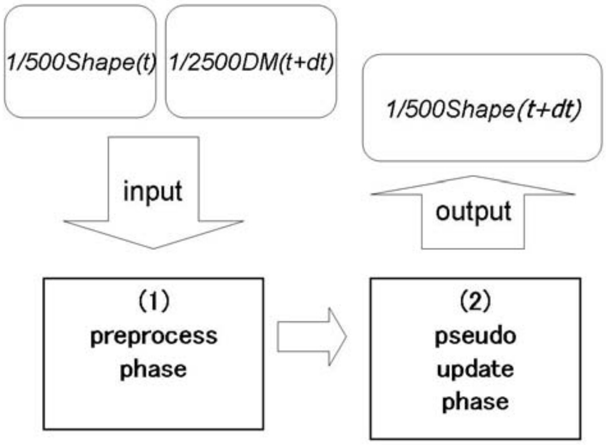


Fig.2 Pseudo update process of building shape in 1/500 digital map.

### 2.2.1 Update pattern of building shape

The paper [8] shows that we can classify update pattern of building shape into seven of the following: 'appearance', 'disappearance', 'mergence', 'split', 'size change', 'shape change', and 'movement'(Fig.3). 'Appearance' means an appearance of a new building on the place in which there has been no building in the past. Conversely, 'disappearance' means a disappearance of a building from the place in which there has been the building in the past. 'Mergence' and 'split' are almost the opposite meaning each other. 'Size change' and 'shape change' sometimes occur in the enlargement or extension of a building. 'Movement' does not happen so many. It is, for example, seen in the movement of a historical building with putting a truck frame under the body of the one. We can replace a single 'movement' into a 'disappearance' and an 'appearance' in a translated position. In this paper, we discuss six patterns except for 'movement'.

### 2.2.2 Algorithm of pseudo update

It is natural that updating a building data by employing lower precision data usually leads to the loss of precision. As we have described in Section 1, it is hopeful to reduce the loss of precision in minimum. Therefore, we

- do not replace all of the building shape in *1/500Shape(t)* into the one in *1/2500DM(t+dt)*,
- investigate whether we should update or not for each building,
- limit the update only to the case which we can conclude with strong confidence that we must update a certain building,
- avoid unnecessary loss of precision, and
- obtain a building data more relevant to the truth.

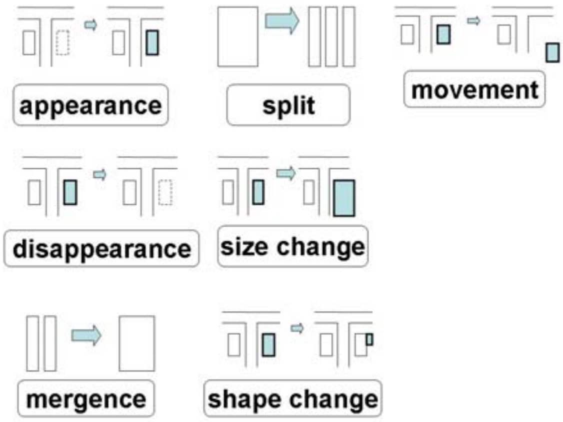


Fig.3 Update pattern of building[8].

First, we define $BS_i$, $gS_i$, $BD_j$, and $gD_j$ as the following:

- $BS_i$: each building in *1/500Shape(t)*. i=1, …, *S-num*. *S-num* is the total number of $BS_i$ .
- $gS_i$: the center of gravity in $BS_i$
- $BD_j$: each building in *1/2500DM(t+dt)*. j=1, …, *D-num*. *D-num* is the total number of $BD_j$ .
- $gD_j$: the center of gravity in $BD_j$

With the above definitions, we propose our algorithm of pseudo update shown in Fig.4.

We describe the detail of Step2 in Fig.5. In here, we define $e_{500}$ and $e_{2500}$ as the following:

- $e_{500}$: maximum error in 1/500 digital map, 0.25 meters length.
- $e_{2500}$: maximum error in 1/2500 digital map, 1.75 meters length. */

We describe the *process1* and *process2* of Fig.5 in more detail.

In the *process1*, we examine whether 'split' occurs or not, and if so, we delete $BS_i$ and adopt the corresponding sequence $BD_u$ , … , $BD_v$(Fig.8).

In the *process2*, we examine whether 'mergence', 'size change' or 'shape change' occurs or not.

Fig.10 shows the algorithm for 'mergence'.

Step1) calculate each $gS_i$ and $gD_j$;
Step2) Based on the distance between $gS_i$ and $gD_j$, we carry out the following two classifications.
$BS_i$ = 0: We should delete it,
1: We should keep it, or
2: We should defer the decision of deletion or keeping.
$BD_j$ = 0: We should not use it for pseudo update,
1: We should use it for pseudo update and adopt it, or
2: We should defer the decision of using or not.
/* The detail of Step2 is described later. */
Step3) Process the building $BS_i$, $BD_j$ whose value is 0 or 1, automatically;
Process the building $BS_i$, $BD_j$ whose value is 2, manually;

Fig.4 Algorithm of pseudo update.

Step2.1) <u>if</u> there are not any $gD_j$ within radius ($e_{500}+e_{2500}$) from $gS_i$ <u>then</u>
Since we can consider that $BS_i$ has been deleted(see Fig.6), we set $BS_i$ to 0.
<u>else</u>
*process1*: examine whether a 'split' occurs or not, and process adequately(described later in Fig.8).
Step2.2) <u>if</u> there are not any $gS_i$ within radius ($e_{500}+e_{2500}$) from $gD_j$ <u>then</u>
Since we can consider that $BD_j$ has been appeared(see Fig.7), we set $BD_j$ to 1.
<u>else</u>
*process2*: examine whether a 'mergence', 'size change', or 'shape change' occurs or not, and process adequately(described later in Fig.10, 12, and 14).

Fig.5 The detail of Step2 in Fig.4.

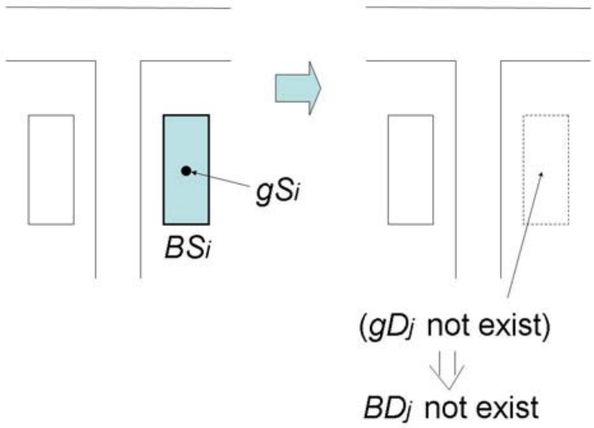


Fig.6 The center of gravity and 'disappearance'.

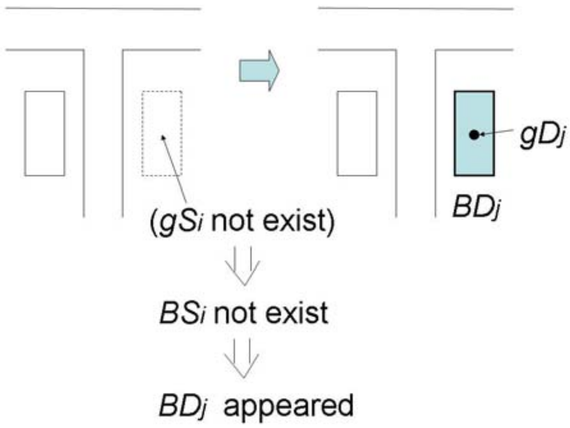


Fig.7 The center of gravity and 'appearance'.

<u>if</u> there are multiple $gD_j$ (say $gD_u$, ... , $gD_v$) within radius ($e_{500}+e_{2500}$) from $gS_i$ <u>then</u>
Since the precision of *1/2500DM(t+dt)* is lower than the one of *1/500Shape(t)*, it is high possibility that a 'split' occurs(See Fig.9). Therefore,
we set $BS_i$ to 0. (We delete $BS_i$)
we set $BD_u$, ... , $BD_v$ to 1. (We adopt them)

Fig.8 The detail of *process1* in Fig.5.

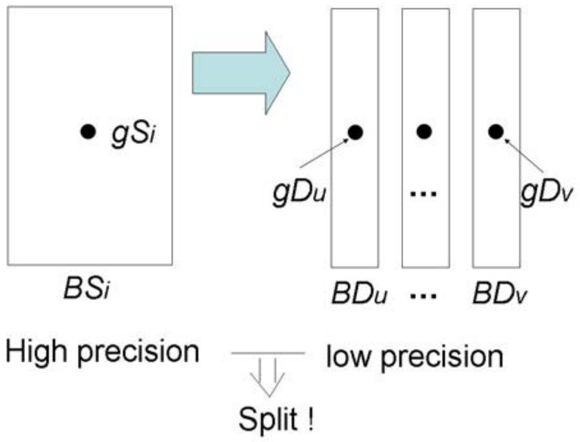


Fig.9 The center of gravity and ‘split’.

> <u>if</u> there are multiple $gS_i$ (say $gS_u$, ..., $gS_v$) within radius ($e_{500}+e_{2500}$) from $gD_j$ <u>then</u>
> Since the precision of *1/2500DM(t+dt)* is lower than the one of *1/500Shape(t)*, it is not easy to distinguish the following two cases(See Fig.11):
> (1) a ‘mergence’ occurs.
> (2) reduction of multiple vertices occurs.
> Therefore,
> we set $BS_u$, ..., $BS_v$ to 2. (We delay the decision of deletion or keeping)
> we set $BD_j$ to 2. (We delay the decision of adoption or not)

Fig.10 The algorithm for ‘mergence’ in *process2*.

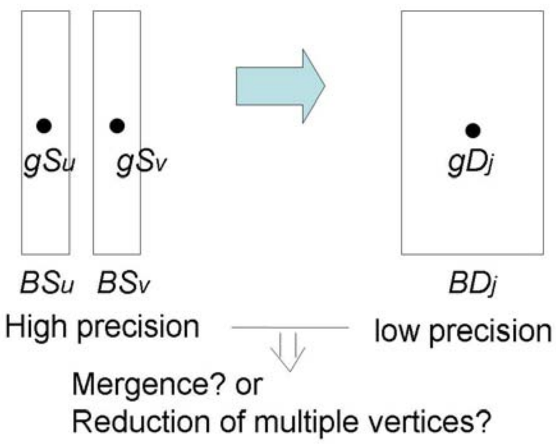


Fig.11 The center of gravity and ‘mergence’.

Next, we show the algorithm for ‘size change’ in Fig.12. We define $S_{big}$, $S_{small}$, $a$, and $b$ as the following:

- $S_{big}$: the area of broader side where we calculate each area of $BS_i$ and $BD_j$ which are mutually counterpart
- $S_{small}$: the area of narrower side where we calculate each area of $BS_i$ and $BD_j$ which are mutually counterpart
- $a$: the length along horizontal axis on the minimum bounding block around $S_{small}$
- $b$: the length along vertical axis on the minimum bounding block around $S_{small}$

With these four variables, Fig.12 shows the algorithm for ‘size change’.

Finally, we show the algorithm for ‘shape change’ in Fig.14. We define $V\_Num(BS_i)$, $V\_Num(BD_j)$ as the following:

- $V\_Num(BS_i)$: the number of vertices in $BS_i$
- $V\_Num(BD_j)$: the number of vertices in $BD_j$

> <u>if</u> $$S_{big} > \frac{(a+e_{500}+e_{2500})(b+e_{500}+e_{2500})}{ab} S_{small}$$
> <u>then</u> Since we can consider the size of the building has changed(See Fig.13),
> we set $BS_i$ to 0. (We delete $BS_i$)
> we set $BD_j$ to 1. (We adopt $BD_j$)

Fig.12 The algorithm for ‘size change’ in *process2*.

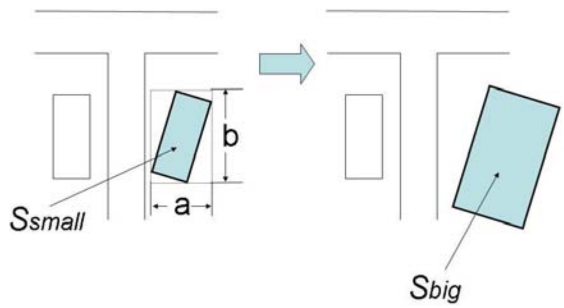


Fig.13 The center of gravity and ‘mergence’.

> <u>if</u> $V\_Num(BD_j) > V\_Num(BS_i)$
> <u>then</u> Since the precision of *1/2500DM(t+dt)* is lower than the one of *1/500Shape(t)*, we can consider the shape of the building has changed(See Fig.15).
> we set $BS_i$ to 0. (We delete $BS_i$)
> we set $BD_j$ to 1. (We adopt $BD_j$)

Fig.14 The algorithm for ‘shape change’ in *process2*.

Tsuyoshi Takayama, Akitsugu Oki,
Hidemi Fukada, Yoshitoshi Murata,
Nobuyoshi Sato and Tetsuo Ikeda

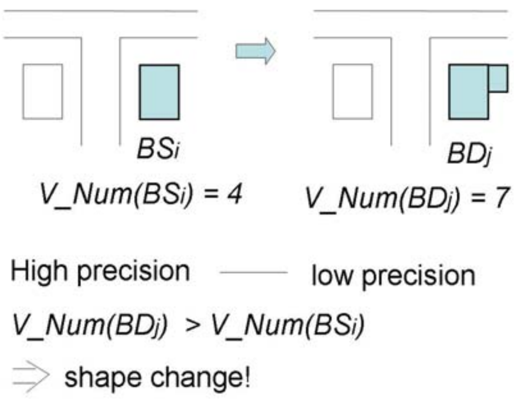


Fig.15 The number of vertices and 'shape change'.

With these two variables, Fig.14 shows the algorithm for 'shape change'. This algorithm can be applicable to appearance or disappearance of hole, too.

Fig.16 in the next page shows the entire flow chart of pseudo update composed of each algorithm described until here.

## 3 Pilot System

We have developed a pilot system. Table 1 shows its implementation environment.

Table 1 Implementation environment of our pilot system

| | |
|---|---|
| Programming language | Visual Basic for Applications 6.3 |
| GIS system | ArcGIS 9.1 |
| Operating System | MS Windows XP Pro SP2 |

Fig.17 shows an example of a map based on the *1/500Shape(t)*. It means that the map is before update.

Fig.18 shows a map based on the *1/2500DM(t+dt)* in the same region. It means that the map is used for update.

We also describe the result of the output from our pilot system. Fig.19 shows the page after pseudo update has finished. It is derived from Fig.17 and Fig.18. Red buildings are determined update or keeping. On the other hand, blue buildings are what we need to determine update or not, manually.

Fig.20 shows the map which has been finished until manual update. We can obtain *1/500Shape(t+dt)* from this map.

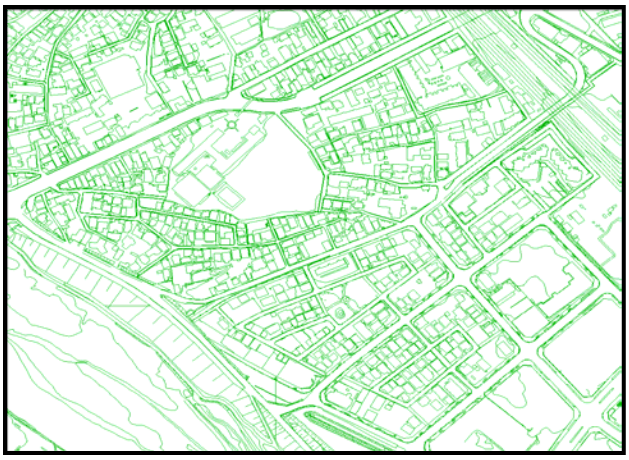

Fig.17 A map based on the *1/500Shape(t)*(before update).

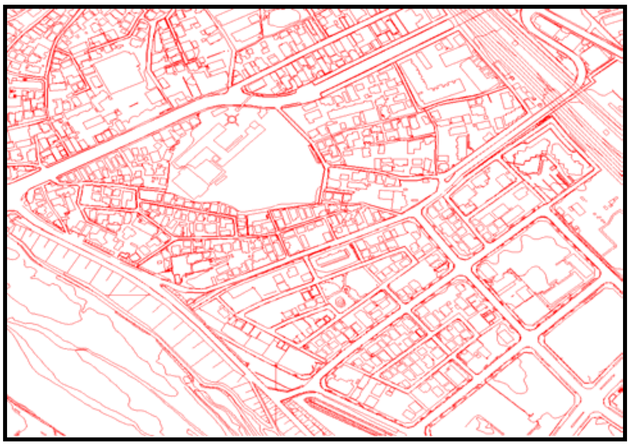

Fig.18 A map based on the *1/2500DM(t+dt)* in the same region as Fig.17(used for update).

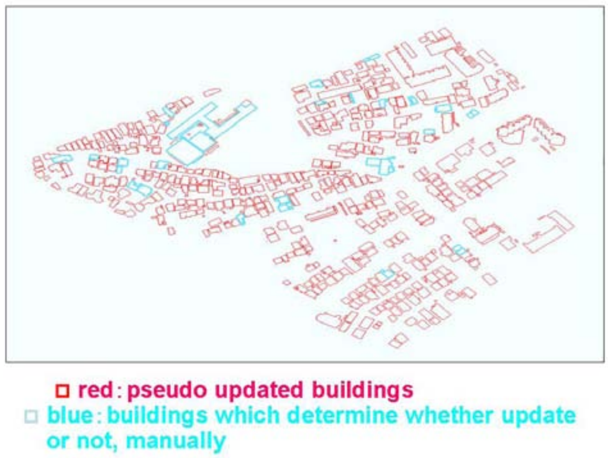


Fig.19 The page after pseudo update has finished.

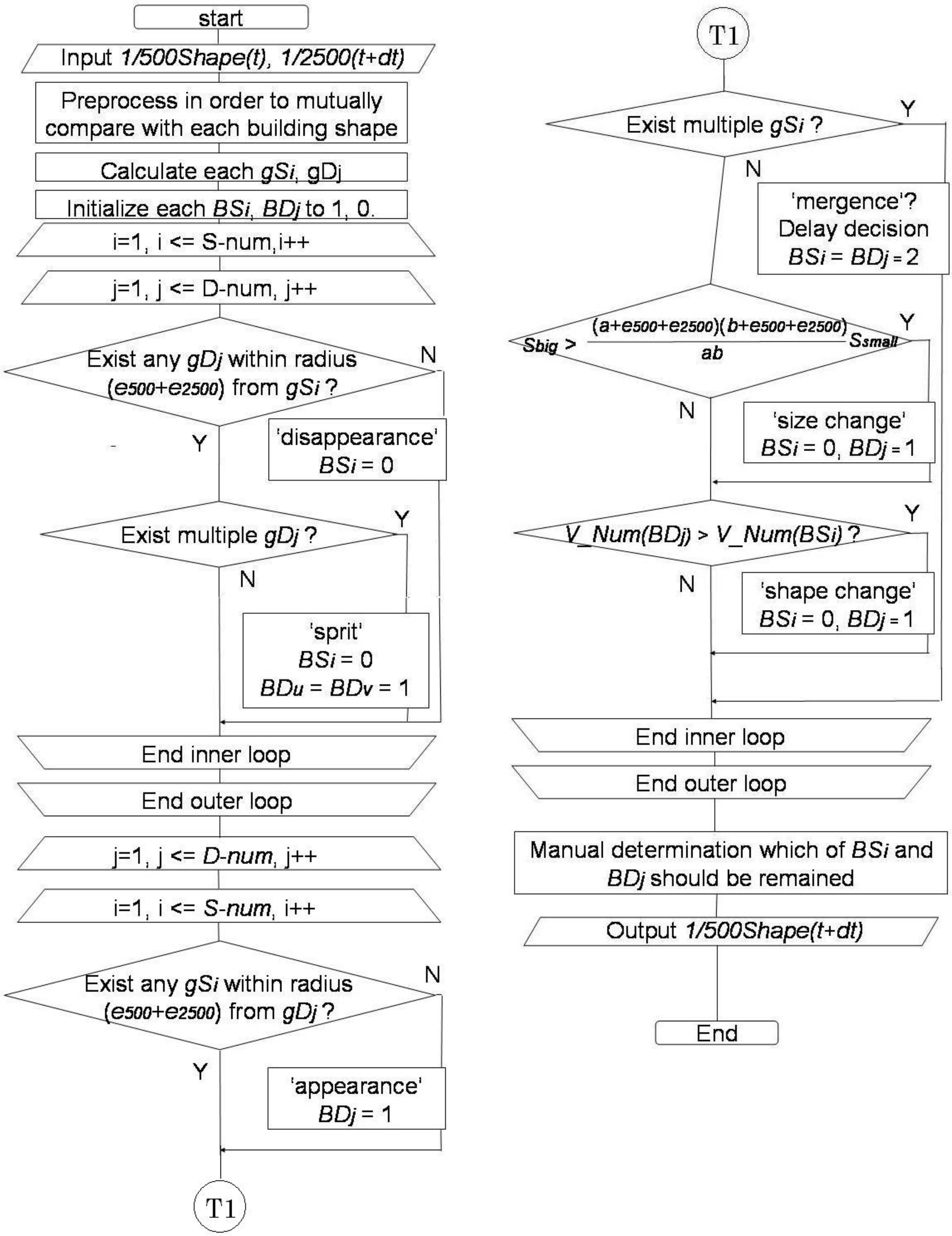


Fig.16 Entire flow chart of pseudo update.

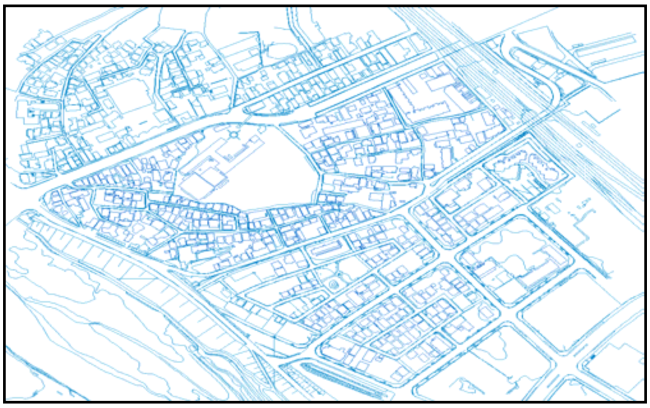

Fig.20 The map which has been finished until manual update(*1/500Shape(t+dt)*).

Table 2 The detail of the five regions

| region | address | The # of buildings in the *1/500 Shape(t)* | The # of buildings in the *1/2500 DM(t+dt)* |
|---|---|---|---|
| $R_1$ | Josai | 317 | 322 |
| $R_2$ | Higashi-Senboku | 534 | 511 |
| $R_3$ | Higashi-Nakano | 258 | 247 |
| $R_4$ | Tenjin | 379 | 347 |
| $R_5$ | Chuo-Douri | 402 | 385 |
| (total) | | 1890 | 1812 |

# 4 Evalution

## 4.1 Quantitative Evaluation

### 4.1.1 Method

As a concrete example, we use 1/500 Shape file and 1/2500 DM data on Morioka-City of Iwate prefecture in our country. Our evaluation is carried out by using total 1890 buildings in the Shape file and 1812 ones in the DM data on the five regions in the Morioka-City. Table 2 shows the detail of the above-mentioned five regions.

We evaluate recall and precision until before manual phase using correct answer determined by visual observation.

### 4.1.2 Result

Table 3 shows the entire results. First, the most right-hand columns show that we have achieved 85.0% recall and 91.9% precision in average of total. Although our algorithm is based on only necessary conditions because it is hard to develop the algorithm based on the necessary and sufficient conditions, both the values of recall and precision are beyond 85.0% together. Furthermore, we can expect to obtain the better evaluation values after the manual determination. According to these facts, we can consider that our method is considered to be feasible.

Table 3 Evaluation result

| | appearance | disappearance | mergence | split | size change | shape change | total |
|---|---|---|---|---|---|---|---|
| A. visual observation (correct answer) | 220 | 389 | 184 | 71 | 95 | 69 | 1028 |
| B. proposition | 205 | 375 | 167 | 63 | 85 | 56 | 951 |
| C. number of correct answer | 192 | 347 | 153 | 56 | 76 | 50 | 874 |
| recall(%) ((C/A)*100) | 87.2 | 89.2 | 83.1 | 78.8 | 80.0 | 72.4 | 85.0 |
| precision(%) ((C/B)*100) | 93.6 | 92.5 | 91.6 | 88.8 | 89.4 | 89.2 | 91.9 |

In order to improve the evaluation value, we will analyze the results in more detail. Table 4 through 9 show recall and precision for each update pattern and region. We can see some variance per update pattern. Especially, we can see the values lower than 80.0(%) in Table 7, which is 'split' pattern. In our algorithm of Fig.16, we detect 'appearance' before than 'split'. It has possibility to fail to detect some splits. We may obtain the better evaluation value if we put 'split' detection before than 'appearance'. We may also be able to realize the similar improvement in the relation between 'disappearance' and 'mergence'.

We can also see the values lower than 80.0(%) in Table 9, which is 'shape change' pattern. In some cases of GIS environment, data of a single building is divided into multiple polygons. In such cases, the algorithm in Fig.14 does not work well. Utilizing 'simplification of polygon' menu has possibility to solve it.

Table 4 Recall and precision in 'appearance'

| | recall(%) | precision(%) |
|---|---|---|
| $R_1$ | 87.2 | 91.1 |
| $R_2$ | 86.5 | 91.8 |
| $R_3$ | 89.4 | 97.1 |
| $R_4$ | 88.6 | 92.8 |
| $R_5$ | 84.6 | 89.1 |

Table 5 Recall and precision in 'disappearance'

| | recall(%) | precision(%) |
|---|---|---|
| $R_1$ | 94.2 | 80.2 |
| $R_2$ | 87.2 | 92.7 |
| $R_3$ | 85.9 | 98.0 |
| $R_4$ | 87.8 | 92.8 |
| $R_5$ | 91.9 | 95.2 |

Table 6 Recall and precision in 'mergence'

| | recall(%) | precision(%) |
|---|---|---|
| $R_1$ | 85.7 | 92.1 |
| $R_2$ | 81.4 | 91.6 |
| $R_3$ | 82.1 | 92.0 |
| $R_4$ | 88.0 | 95.6 |
| $R_5$ | 80.5 | 85.2 |

Table 7 Recall and precision in 'split'

| | recall(%) | precision(%) |
|---|---|---|
| $R_1$ | 80.0 | 94.1 |
| $R_2$ | 73.6 | 94.7 |
| $R_3$ | 100.0 | 100.0 |
| $R_4$ | 75.0 | 90.0 |
| $R_5$ | 87.5 | 93.3 |

Table 8 Recall and precision in 'size change'

| | recall(%) | precision(%) |
|---|---|---|
| $R_1$ | 83.3 | 90.9 |
| $R_2$ | 80.0 | 93.3 |
| $R_3$ | 77.7 | 93.3 |
| $R_4$ | 68.1 | 75.0 |
| $R_5$ | 89.2 | 100.0 |

Table 9 Recall and precision in 'shape change'

| | recall(%) | Precision(%) |
|---|---|---|
| $R_1$ | 85.7 | 83.3 |
| $R_2$ | 75.0 | 90.0 |
| $R_3$ | 77.7 | 100.0 |
| $R_4$ | 70.0 | 77.7 |
| $R_5$ | 68.4 | 92.8 |

## 4.2 Qualitative Evaluation

### 4.2.1 Method

In the evaluation division of fixed property in the Morioka municipal government, we carry out the presentation and hearing concerning our proposition.

### 4.2.2 Result

We have obtained the following comments:

(1) It is difficult to grasp the change of building shape completely in the conventional work manner. It is feasible if we develop a tax evaluation support system for fixed property based on the proposed method.

(2) Since tax imposition error is terribly serious in a municipal government, 'disappearance' is the most important in seven update patterns.

(3) Although precision loss in pseudo update is certainly drawback, it is more important and worthwhile that the shape or 'existence or not' of building is relevant to the truth.

(4) In the experiment of subsection 4.1, the ratio of building which we need to determine update or not manually has been 9.31% in total. It is not low and is hopeful to decrease the ratio. However, it is helpful to be provided which building we should confirm whether update or not, because we come to not need to confirm it for all buildings.

# 5 Conclusion

We have proposed a pseudo update method of building shape in 1/500 shape file which is representative as a road ledger map by employing 1/2500 DM data. According to the evaluation, our proposition has achieved 85.0 % recall and 91.9 % precision in average of total. Our proposition has also obtained good result in qualitative evaluation.

We are planning to several future works. (i) To deal with overlap between updated building and 'road or other building'. (ii) Investigation on reduction of multiple vertices based on the difference in precision. (iii) Investigation on road update. After we have solved these problems, we would like to proceed to entire update of a digital map.

*References:*

[1] J. Komarkova, M. Hub, and M. Ulrich: UMN Map Server and Influence of Selected Security Measurements on Its Response Time, *WSEAS Transactions on Information Science and Applications*, Issue 6, Vol.2, 2005, pp.753-760.

[2] L. Chockalingam: Generating Various Thematic Maps using Remote Sensing and Topographic Data Sources to Map Surficial Features, *WSEAS Transactions on Environment and Development*, Issue 10, Vol.2, 2006, pp.1376-1382.

[3] F. Samadzadegan, S. Saeedi, and M. Hoseini: Automatic Image to Map Registration Based on Genetic Algorithm, *WSEAS Transactions on Signal Processing*, Issue 1, Vol.3, 2007, pp.74-79.

[4] D. Klang: Automatic Detections of Changes in Road Database Using Satellite Imagery, *International Archive of Photogrammetry and Remote Sensing*, Vol.32, 1998, pp.293-298.

[5] S. Watanabe and K. Miyajima: Detecting Building Changes Using Epipolar Constraint from Aerial Images Taken at Different Positions, *Proceedings of the 2001 International Conference on Image Processing*, Vol. 2, Issue, 7-10, 2001, pp.201 – 204.

[6] M. Nakagawa, R. Shibasaki: Building Change Detection Algorithm Using Aerial Imagery and Precise 3-D Data, *Proc. of the 26th Asian Conference on Remote Sensing*, 2005, (CD-ROM).

[7] Z. Liu, P. Gong, et.al: Automated Urban Building Change Detection Using UltraCamD and Existing CAD Data, *MAPPS/ASPRS 2006 Specialty Conference*, 2006.

[8] Z. C. Shi, R. Shibasaki: GIS Database Revision, *Proc. of 19th ISPRS(International Society for Photogrammetry and Remote Sensing) Congress*, 2000, (CD-ROM).

[9] A. Yamazawa, R. Hamajima: Hybrid Map Creating for Integration Type GIS, *Bulletin of Maebashi Institute of Technology*, Vol.7, 2003, pp.13-20.