Journal of Mobile Multimedia, Vol. 11, No. 3&4 (2015) 281-295

PERFORMANCE EVALUATION OF HIGH-SPEED COMMUNICATION METHOD BY COMPACTIFICATION OF DESIGN DATA

SHUJI SHOJI and AKIO KOYAMA

Yamagata University, Japan

shoji@specsystem.co.jp, akoyama@yz.yamagata-u.ac.jp

A large number of isomorphic polygons are included in design data of large scale integrated circuit. Fast searching and classification for isomorphic polygons in these design data are able to apply to compactification of design data. Compactification of design data has a merit for communication via networks. So, we use expression method using a geometric invariant feature value for polygon. This method does not be affected for rotation and reduced scale. The Geometric Hashing method is known widely as an object recognition method using geometric invariant feature amount which expresses feature of shape. However, this method has drawbacks which increase computational complexity and memory usage amount with increasing of feature points. To solve these issues, we propose a fast and high accuracy search method for isomorphic polygon and apply to compactification of design data. From evaluation results of the proposed method, we verified that the proposed method can compact the design data by performing fast search and classification for isomorphic polygons, and reduce the communication quantity drastically.

Key words: Isomorphic polygon, geometric invariant feature value, geometric hashing method, high-speed communication

1 Introduction

In recent years, design data of Large Scale Integrated circuit (LSI) has become large scale, because of progress of high density, miniaturization of LSI, and appearance of various circuit patterns [1]. The design data include a large amount of polygons. A combination of polygons becomes parts and they are compacted by layers [2][3]. However, there are many cases that isomorphic polygons or patterns are used in plural parts. Furthermore, there are flat design data which do not have layer or parts. Since the flat design data include many isomorphic patterns and polygons, a degradation of data handling performance occurs. Therefore, it is important to compact the large scale design data. Compactification of design data can contribute to reduce communication and memory usage amounts.

Since a large scale design data includes huge number of polygons, it is difficult to search isomorphic polygons rapidly by conventional linear search methods because a search of isomorphic polygons needs shape comparison process. In this research, we consider an expression method by a feature value which does not receive influence of geometric conversion for features of polygon in order to reduce amount of shape comparison process.

As expression method using a geometric invariant feature value [4][5][6] for features of shape, the Geometric Hashing (GH) method is well known [7]. The GH method forms a model by extracting feature points in an image. The invariant feature amount is calculated by selected 2 base points in feature points included to the model. The invariant feature amount is recorded to a hash table with the model and base pair. Since these processing is repeated to all feature points in all images, computational complexity and memory usage amount increase depending on the number of extracting feature points. To solve this problem, Locally Likely Arrangement Hashing (LLAH) method was proposed [8][9][10]. However, since computational complexity which depends on the number of feature points and register to the hash table occur, memory usage amount increases.

Here, we consider polygons as an image. Each feature point replaces vertexes of polygons and searching of isomorphic polygons is performed by similarity transformation of GH method. When the number of feature points (vertices) which is included the image (polygon) of a model is n, combination of two base points is ${}_{n}P_{2}$. Moreover, computational complexity of register is $O(n^3)$. Recently, there are design data with the number of polygon over one hundred million points. In this case, all the number of vertices are beyond several hundred million points. The proposed method does not have limitation for the number of polygons or vertices of objected data. However, in GH method, when the combination of base of n vertices supposes ${}_{n}P_{2}$, since computational complexity and memory usage amount become huge, fast searching cannot realize. In LLAH method, for n vertices, when referenced neighbourhood points are k, computational complexity at entry of invariant feature value for similarity transformation of a model becomes $O({}_{k}C_{3}nN)$. Where, N is the number of polygons. Although LLAH method is improved than GH method, it is difficult to apply to the design data with several hundred million points.

In the proposed method, feature amount of polygon is expressed by one feature value [11]. This feature value is invariant feature value which is not affected of parallel shift, rotation and reduced scale. In this value, computational complexity is a few and the number of entry times is only one per one polygon. Namely, the number of invariant feature value does not depend on the number of vertexes n, and it is decided by the number of polygons N. By hash table generated at N-invariant feature values, fast search of polygon group with same invariant feature value can realize. By searching and classification of isomorphic polygons can realize quickly and strictly by performing detail shape comparison for selected polygons at this grouping for feature value. From classification results of this method, the design data is compacted and the compacted design data is used by application system. Furthermore, use of compacted design data drastically reduces not only memory usage amount but also communication amount. Therefore, high speed communication can be realized.

The rest of paper is organized as follows. In section 2, we introduce a related work. In section 3, we explain the proposed method. In section 4, we describe performance evaluations. Finally, some conclusions are given in section 5.

2 Related Work

2.1 GH Method

A GH method [7] forms a model by extracting several feature points. Arbitrary 2 points (P_i, P_j) from feature points included in the model are selected as a base pair and a base vector P_iP_j generates. When the vector P_iP_j sets as x axis and an origin point sets as middle point between P_i and P_j, the vector P_iP_j shows positive direction of x axis. In this time, by setting as 1 for length of P_iP_j, feature points except P_i and P_j in the model are projected in this coordinate system and quantized at the small divided area. Feature points except P_i and P_j, the model and pair of the base point (P_i, P_j) are stored to a bin (small area) which configures a hash table. This process repeats regarding all of feature points in the model.

Figure 1 shows an example. In model M constructed by feature points (P_1, P_2, P_3, P_4, P_5), this is an example when points (P_1, P_2) are selected as a base pair. The base vector P_1P_2 sets as x axis and a middle point between P_1 and P_2 sets as an origin point, and remaining feature points (P_3, P_4, P_5) are projected to a rectangular coordinates system. Projected feature points and (M, (P_1, P_2)) are stored to a bin.

By storing multiple images as each model to the hash table by calculating invariant feature amount, it is possible to consider this hash table as search image database. When a similar image for a certain image Q is searched from this database, feature point of image Q is extracted, and base and invariant feature amount are matched for all feature points. Then this is voted to the bin in the hash table. As a result of vote, a model which gets most number of votes becomes most similar image. Moreover, a model group where the number of votes exceeds a constant value is considered as similar images.

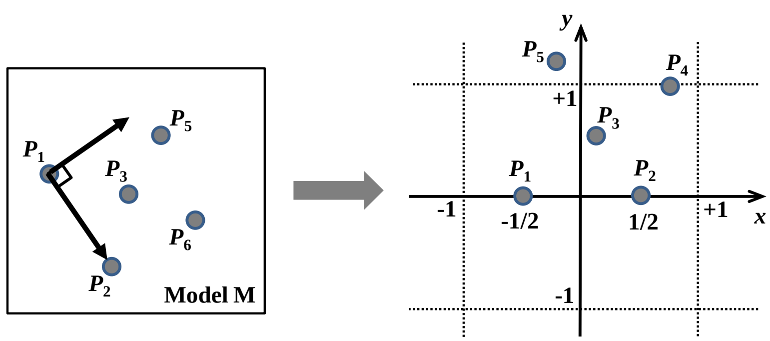


Figure 1 Projection to the rectangular coordinates system by base P_1P_2.

3 Proposed Method

3.1 Geometric Invariant Feature Value of Polygon

A feature value of polygon is an invariant feature value which does not affect geometric transformation like translation, rotation and reduced scale. An invariant feature value is same value at the isomorphic polygon. However, even if an invariant feature value is same, there is a case which is not an isomorphic polygon.

a) Target Polygon

A vertex coordinate which is configuring polygon is integer values with sign. In the proposed method, all polygons included in data become target without distinguishing hierarchical structure or size. The isomorphic polygon is the similar polygon. There is a shape which is not considered to be an isomorphic polygon even if an outline of polygon is same. The proposed method considers that left and right polygons in figure 2 are not to be the isomorphic polygon. In upper figure of figure 2, although outlines are same, vertex coordinates are different. In lower figure of figure 2, although outlines are same, the number of vertices is different. When such polygons exist, as pre-processing, shaping by geometric logic operations is required.

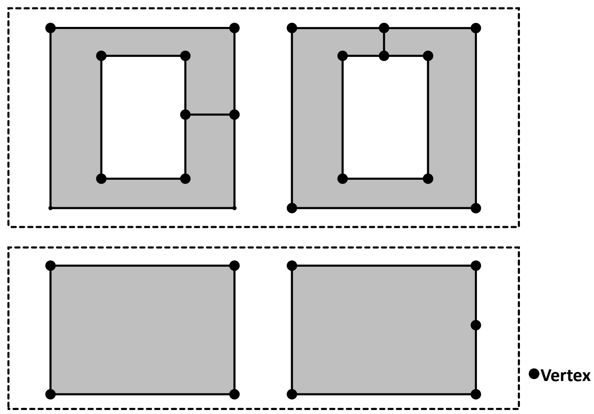


Figure 2 An example of polygon which are different of isomorphic polygon.

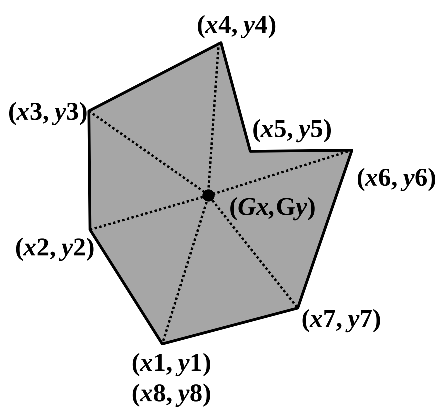


Figure 3 An example of polygon with eight vertices.

b) Invariant Feature Value

A calculation procedure of invariant feature value for polygon is as follows. Let consider a polygon with n vertices. Figure 3 shows an example of the polygon with eight vertices.

An equation (1) shows a distance summation between each vertex and a center of gravity (distance summation of a center of gravity).

$$GL = \sum_{i=1}^{n-1} \sqrt{(x_i - G_x)^2 + (y_i - G_y)^2} \qquad (1)$$

An equation (2) shows a periphery distance.

$$PL = \sum_{i=1}^{n-1} \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \qquad (2)$$

An equation (3) shows an invariant feature value.

$$GF = \frac{GL}{PL} \qquad (3)$$

Since the invariant feature value is ratio of distance summation of a center of gravity and outline distance, it is not affected for parallel shift, rotation and reduced scale.

3.2 Fast Search Method of Isomorphic Polygon

a) Procedure

First, the invariant feature value is calculated for N target polygons. Prepare a hash function with hash key which is an invariant feature value, and calculate a group ID which is a hash value. Using group ID as the hash value, the polygon ID number is stored into the hash table. Then, after shape comparison process is performed in same group, sub groups are formed. These sub groups become classified polygons. Moreover, when a searching process is performed, same procedure is carried out.

b) Calculation of Invariant Feature Value

For N polygons included to the data, the invariant feature values GF(i) are calculated by equation (4).

$$GF(i) = \frac{GL(i)}{PL(i)} \qquad (i = 1,2,\cdots,N) \qquad (4)$$

Moreover, N invariant feature values are normalized by equation (5).

$$GF_n(i) = \frac{(GF(i) - GF_{min})}{(GF_{max} - GF_{min})} \qquad (i = 1,2,\cdots,N) \qquad (5)$$

Where,

$$GF_{min} = \min(GF(1), GF(2), \cdots, GF(N))$$
$$GF_{max} = \max(GF(1), GF(2), \cdots, GF(N))$$

c) Calculation of Group ID

An index for accessing to isomorphic polygon group is determined by a normalized invariant feature value. This index is the group ID. The equation (6) is hash function with hash key which is normalized invariant feature value. A group ID is calculated as hash value. When the group ID is determined by this function, similarity in the group can determine.

$$G_{ID}(i) = GF_n(i) \cdot E \qquad (i = 1,2,\cdots,N) \qquad (6)$$

Where, E is integer coefficient of normalized invariant feature value. This coefficient has great meaning for uniqueness of the polygon included isomorphic polygon group. The uniqueness of the group ID means that each polygon belongs to one kind group. Namely, when E is large, the uniqueness of group ID becomes high. Meanwhile, when E is small, the uniqueness of group ID becomes low. When the uniqueness of group ID is low, low similarity polygons belong to same group. Meanwhile, when the uniqueness of group ID is high, high similarity polygons belong to same group. Namely, when the uniqueness is high, processing quantity of shape comparison is reduced (describe later). The coefficient E determines processing amount of shape comparison.

d) Shape Comparison

For the polygon group which was classified in hash table, strict classification is performed by shape comparisons in detail. Table 1 shows items by this processing.

Although a vertex number comparison does not have meaning for isomorphic polygon search, it can verify different shapes which have same invariant feature value. An invariant feature value comparison can perform strict classification by using a real number before shape comparison. An area comparison is effective for classification of congruent polygon, and it can perform strict classification before shape comparison. A shape comparison processing for congruent polygon compares side length. In the case of similar polygon, side length and periphery distance are compared. Figure 4 shows hash table and database structure of classification results by shape comparison.

e) An Example of Classification Results

Figure 5 shows polygon samples for classification. Invariant feature values of all polygons are same and calculated group ID is as m. Polygon ID 1 and 2 are similar polygon. Although ID 3 is look like to ID 1 and 2, it is not similar relation.

Figure 6 shows a hash table and database of classification results for figure 5.

Table 1. Items by shape comparison.

Vertex number comparison
Invariant feature value comparison (real value)
Area comparison (in the case of search and classification by congruent polygon)
Shape comparison (each side length comparison)

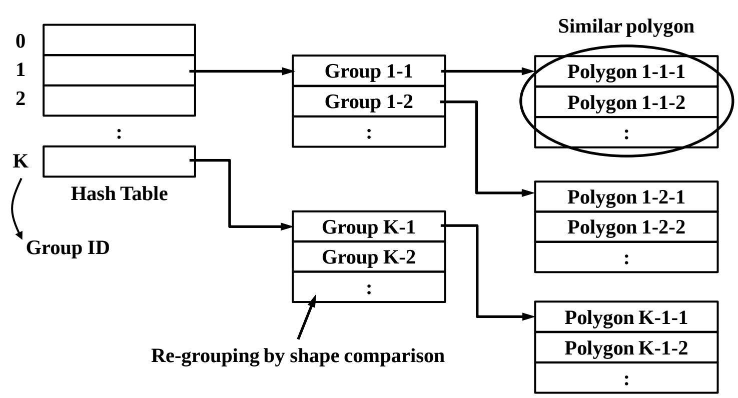


Figure 4 Hash table and database structure of classification results by shape comparison.

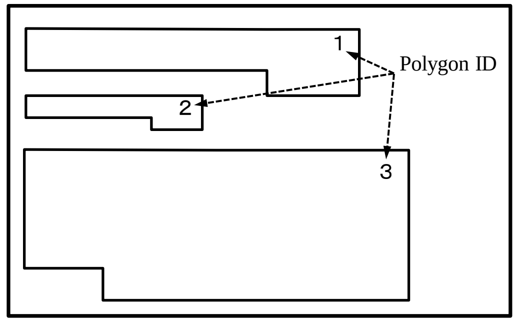


Figure 5 Polygon samples for classification.

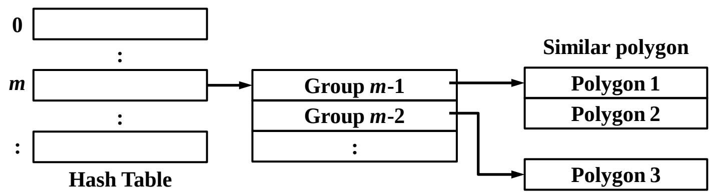


Figure 6 Database structure of classification result for sample data.

3.3 High Speed Transmission Considering Confidentiality of Design Data

a) Outline

Since design data need high confidentiality, it cannot copy or refer easily. In the case of referring, a user can refer the design data in local machine environment by receiving restriction. These restrictions increase operation cost. So, we propose server-client environment as shown in figure 7 to relax the restriction. A high speed communication is provided by using the compacted design data. Moreover, confidentiality of the design data is considered by access restriction of access library.

The server classifies design data requested from clients to isomorphic polygons by the proposed method. Then, compactification is performed. The compacted design data are stored to memory of server. The server sends the compacted design data to the client. The client receives the design data and stores to own memory. After that, the access of the same design data at the client is performed to the design data in own memory. When the referring finished, client's memory is released. The server releases the memory when all referring were nothing.

Why the compacted design data do not save to hard disk is that compact processing of proposed method is enough fast. This is due to hard disk capacity and confidentiality. Furthermore, high speed communication enables by avoiding hard disk access. Figure 8 shows an access model of design data by proposed server and client environment.

From 1 to 11 in figure 8 shows a flow from access request of the design data by client application until acquire of the data. From 12 to 17 in figure 8 shows a flow from next access request of the design data until acquire of the data. This means that access request of the design data between the server and client does not occur from second time.

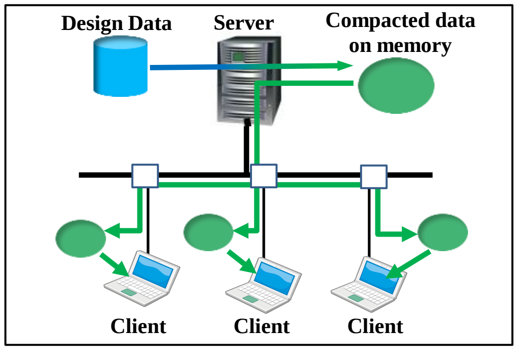


Figure 7 Access of design data between clients and server.

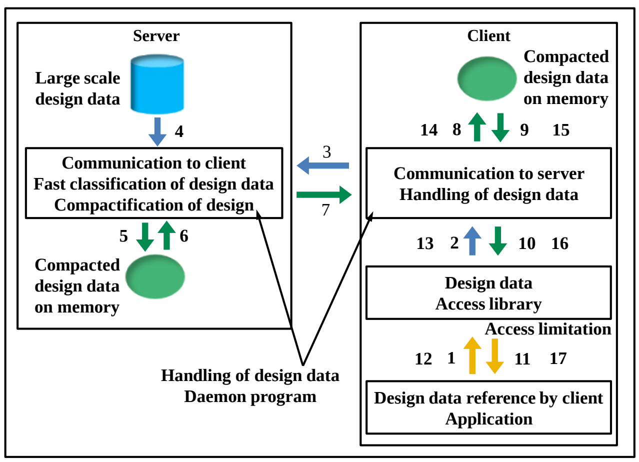


Figure 8 An access model of design data between client and server.

b) Compactification of Design Data

After classification by fast search of isomorphic polygon, compactification of the design data is performed. Vertex coordinates of polygon are defined to the number of types of classified polygons. For referring, polygon numbers and location information are defined. Figure 9 shows an example of structure of compacted design data.

-1	**Polygon counts m**					
Polygon No.1	**Vertices n1**	x_1	y_1	**…**	$x_{(n1-2)}$	$y_{(n1-2)}$
…	**…**	**…**	**…**	**…**	**…**	**…**
Polygon No.m	**Vertices nm**	x_1	y_1	**…**	$x_{(nm-2)}$	$y_{(nm-2)}$
Reference polygon No.1	**Group 1 counts k1**					
location x_1	**location** y_1	**Ctrl**	**Mirror**	**Scale**	**Rotate**	
…	**…**	**…**	**…**	**…**	**…**	
location x_{k1}	**location** y_{k1}	**Ctrl**	**Mirror**	**Scale**	**Rotate**	
Reference polygon No.2	**Group 2 counts k2**					
location x_1	**location** y_1	**Ctrl**	**Mirror**	**Scale**	**Rotate**	
…	**…**	**…**	**…**	**…**	**…**	
location x_{k2}	**location** y_{k2}	**Ctrl**	**Mirror**	**Scale**	**Rotate**	
…	**…**					
…	**…**	**…**	**…**	**…**	**…**	
Reference polygon No.m	**Group m counts km**					
…	**…**	**…**	**…**	**…**	**…**	

Figure 9 Structure of compacted design data.

Value -1 means that definition of polygon continues. Polygon counts m is the definition number of classified polygons. From polygon No.1 to No.m is definition of classified polygon. Vertices n1 to nm of polygon definition is the number of vertices of each polygon, and from (x1, y1) to (xn1-2, yn1-2) is definition of vertex coordinates. Here, a vertex coordinate is defined at a relative coordinate from a starting point. Since a coordinate of starting and ending points is (0, 0), the definition is no need. Therefore, coordinate definition of the number of vertices -2 continues. From reference Polygon No.1 to No.x are polygon number of references and values are from 1 to m. The number of definitions of each reference group is defined Group 1 counts k1 to km, and location information of reference polygon is defined (place x, place y), Ctrl Mirror, Scale and Rotate.

(place x, place y): Absolute coordinates of location

Ctrl: Show control information of Mirror, Scale and Rotate.

Bit 0: Mirror output existence (default : mirror OFF)

Bit 1: Scale ratio output existence (default : 1.0)

Bit 2: Rotation angle output existence (default : 0.0)

Mirror: Mirror processing

1: The mirror according to the X-axis

2: The mirror according to the Y-axis

Scale: Scale value

Rotate: Rotation angle (1 degree to 359 degrees)

c) Transfer Procedure

The design data for transmission are compacted design data. Transmission and receiving tasks with multi-port are put in a server and clients. High speed transmission is realized by these tasks. First, a client requests reference of design data to the server. The server sends division transmission summary of the design data to the client and both machines prepare to transmission and receiving. The division transmission summary includes size of all data, the number of division, division size and the number of concurrent communication ports. When the client prepared receiving, it requests beginning of transmission. The server sends division data to the client. When the server finished all transmission, it sends transmission finish notice. Figure 10 shows a detail sequence.

The division data has own location information (division data number) in all data, and the receiving task in the client can judge a stored location of received data instantly. As the results, since the client does not need an exclusion control for stored and merge processing of all data, high speed communication can be realized. Figure 11 shows the receiving task and relation of all data after receiving.

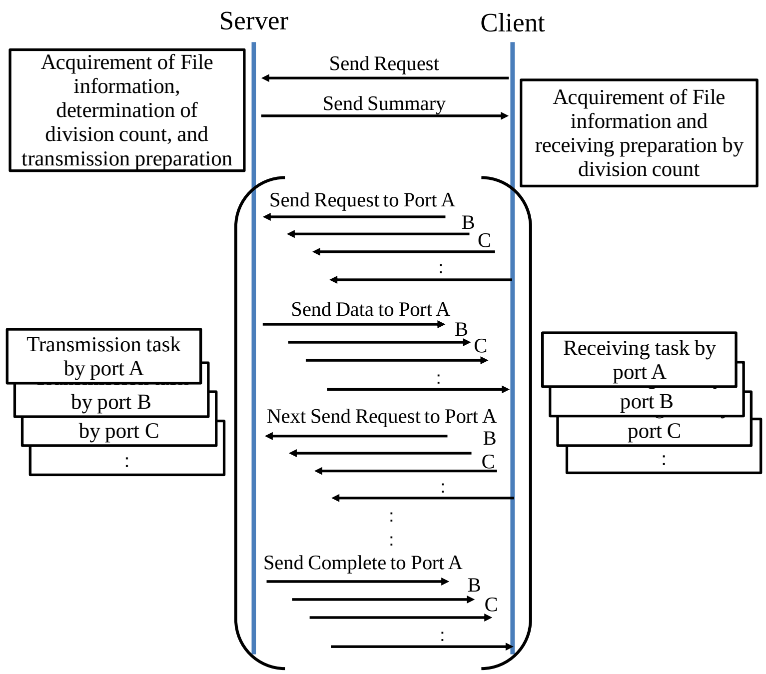


Figure 10 A transmission procedure of compacted design data.

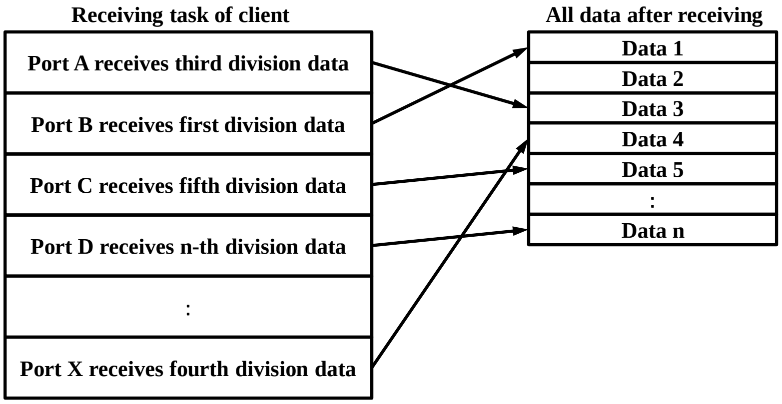


Figure 11 Receiving task and relation of all data after receiving.

4 Performance Evaluation

4.1. Outline

To evaluate the performance and usefulness of the proposed method, we prepared the polygon data as shown in table 2. For prepared data, search and classification of congruent polygons and similar polygons execute. Integer coefficients E used in this evaluation are as follows.

Search and classification of congruent polygon: E=1,000,000

Search and classification of similar polygon: E=1,000

Since the search and classification of similar polygons include a calculation of scale coordinates, an accuracy of invariant feature value drops than a congruent polygon. Therefore, the integer coefficient E of similar polygon is set smaller. We verified the integer coefficient value E by preparing another data. As the results, even if E sets to low value than above E, we verified that the search results do not change. So we used E value which described above as the evaluation.

Next, we evaluate compression rate by compacted design data based on classification results. Moreover, we evaluate communication speed regarding before compact and after compact.

Table 2 Data list for evaluation.

Data No.	1	2	3	4	5	6
Number of polygons (N)	380,106	2,082,669	4,090,373	7,118,380	11,347,748	15,420,367
Number of vertices (*V*)	2,855,008	15,556,614	29,389,334	59,757,164	102,561,374	140,823,000

4.2. Experimental Results

Table 3 shows a comparison of computational complexity for each method. Here, for the GH and LLAH methods, we compare the computational complexity of invariant feature value at stored to the hash table. In this evaluation, we assume that the number of polygons N is the number of images (or the number of model) and the number of vertices n in one polygon is the number of feature points. Where, k ($k < n$) is the neighbourhood points referred by LLAH method. Although the conventional methods depend on the number of vertices n (the number of feature points n), the proposed method does not depend on the number of vertices. Namely, computational complexity becomes O(N). Therefore, there is big difference for computational complexity between the proposed method and conventional methods.

Table 4 and table 5 show the results of search and classification for isomorphic polygons which include congruent and similar polygons. The number of polygon kinds is shown in table 4 and the processing time is shown in table 5. From table 2, table 4, table 5 and figure 12, the number of isomorphic polygon kinds increase linearly with increasing of the number of polygons. The processing time increases linearly with increasing of the number of polygons. There is big difference for processing time between data 3 and data 4. This is because increasing rate of the number of polygons increases largely from data 4. Moreover, the processing time between congruent polygon and similar polygon increases largely from data 4. This is because calculation time of shape comparison is increasing largely with increasing of the number of polygons.

Next, based on classification results, compactification of design data carried out. Table 6 shows this results and table 7 shows compression rate of the data. As shown in these tables, evaluation data could be compacted to less than 20% by using the proposed method. Here, despite the number of kinds of the isomorphic polygon is small, the data size of classification of similar polygon for congruent polygon is large. This is because though shape definition is decreased in the case of similarity, the definition of scale factor which was unneeded at congruent is needed.

Table 3 Comparison of computational complexity for each method.

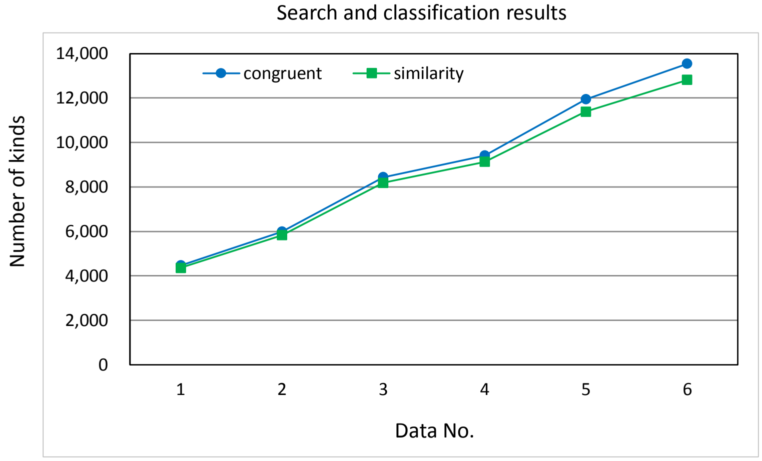
GH method	LLAH method	Proposed method
$O(n^3N)$	$O({}_kC_3nN)$	$O(N)$

Table 4 Search and classification results of isomorphic polygons (number of polygon kinds).

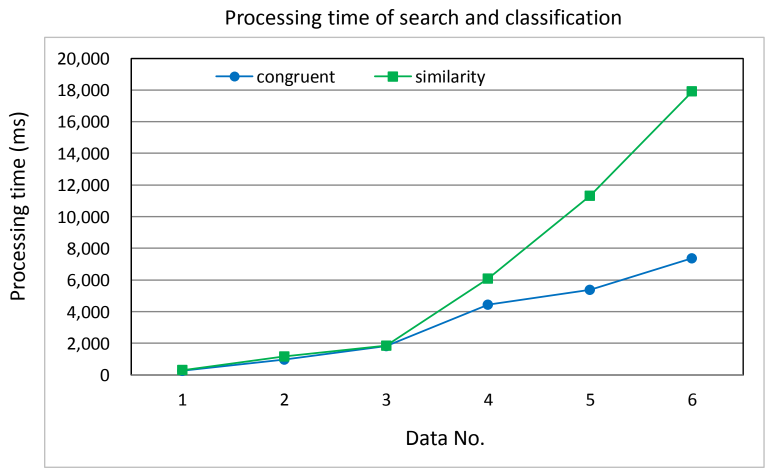
Data No.	1	2	3	4	5	6
Number of congruent polygon kinds	4,471	5,985	8,440	9,414	11,944	13,540
Number of similar polygon kinds	4,360	5,828	8,191	9,125	11,380	12,814

Table 5 Search and classification results of isomorphic polygons (processing time).

Data No.	1	2	3	4	5	6
Congruent polygon (ms)	250	968	1,826	4,431	5,382	7,363
Similar polygon (ms)	296	1,154	1,856	6,084	11,310	17,909



(a)



(b)

Figure 12 Search and classification results of similar and congruent polygons.

Table 6 Data compactification by search and classification of isomorphic polygon.

Data No.	1	2	3	4	5	6
Evaluation data (Byte)	24,360,584	132,792,320	251,476,404	506,531,072	865,882,320	1,188,266,092
Congruent polygon (Byte)	4,279,903	21,194,104	41,411,342	72,625,679	118,251,005	158,762,505
Similar polygon (Byte)	4,701,921	21,254,784	41,509,564	72,801,372	118,465,763	159,097,398

Table 7 Data compression rate by search and classification of isomorphic polygon.

Data No.	1	2	3	4	5	6
Congruent polygon (%)	17.57	15.96	16.47	14.34	13.66	13.36
Similar polygon (%)	19.30	16.01	16.51	14.37	13.68	13.39

Next, transfer evaluation of compacted design data carried out. Table 8 and figure 13 show these results. We verified that the transmission time could be decreased less than one-fourth by using the proposed method. Furthermore, evaluation of concurrent transmission using multi-ports carried out by using evaluation data 6. Table 9 and figure 14 show these results. This experiment evaluates that how much time is decreased by compactification of design data. As shown in figure 14, when the number of ports is large, namely high network load, compactification of design data is effective.

Table 8 Comparison of transmission time by data compactification.

Data No.	1	2	3	4	5	6
Evaluation data (ms)	312	1,919	3,760	7,472	12,215	16,459
Congruent polygon (ms)	78	281	561	983	1,326	2,337
Similar polygon (ms)	78	265	562	998	1,467	2,402

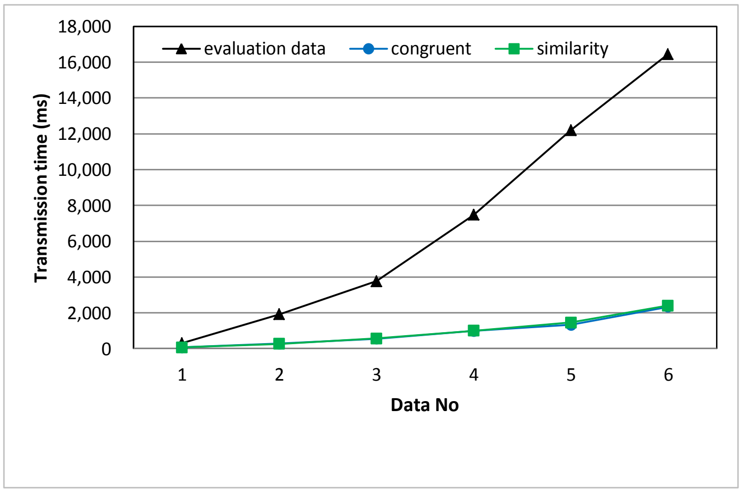


Figure 13 Comparison of transmission time by data compactification.

Table 9 Transmission speed comparison by multi-ports concurrent transmission using No. 6 evaluation data.

Number of ports	Average of 1 port	Average of 2 ports	Average of 3 ports	Average of 4 ports
Evaluation data (ms)	16,459	25,275	32,437	40,400
Congruent polygon (ms)	2,337	2,428	3,757	4,119
Similar polygon (ms)	2,402	2,699	3,417	4,208

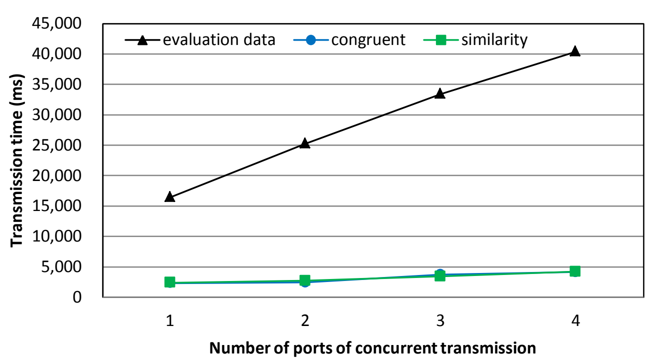


Figure 14 Comparison of transmission time by concurrent transmission of multi-ports.

5 Conclusions

In this paper, we proposed fast search and classification method of isomorphic polygons and fast transmission method of the design data. The proposed method expresses the polygon as invariant feature value. Although this method is an application method of GH method, it drastically decreased computational complexity which was drawback of GH method, and realized fast transmission. In the proposed method, since shape comparison is performed after register to the hash table, simple comparison of processing time cannot carry out. However, since search and classification of large scale polygon data with over one hundred million vertices is carried out at ten few seconds, usefulness of the proposed method showed.

From the experimental results, we verified as follows.

- Evaluation data could be compacted to less than 20% by using the proposed method.
- Transmission time could be decreased less than one-fourth by using the proposed method.

As the future work, we would like to confirm an effectiveness of confidentiality, and consider an adaptive decision method of an integer coefficient E.

References

1. IEICE knowledge base, group 10 (integrated circuit) – Part 1 (basic configuration and design technology), 2010. http://www.ieice-hbkb.org/files/10/10gun_01hen_01.pdf (accessed 2015-04-13).
2. GDSII Stream Format Manual, February 1987. http://bitsavers.trailing-edge.com/pdf/calma/GDS_II_Stream_Format_Manual_6.0_Feb87.pdf (accessed 2015-04-13).
3. OASISTM - Open Artwork System Interchange Standard, SEMI Draft Document 3626 2003/04/23 http://www.wrcad.com/oasis/oasis-3626-042303-draft.pdf (accessed 2015-04-13).
4. Bishop, C. M., Pattern Recognition and Machine Learning, Springer-Verlag, 2006.
5. Motoda, H. et. al., Pattern Recognize and Machine Learning, First volume, Springer Japan, 2007.
6. Motoda, H. et. al., Pattern Recognize and Machine Learning, Second volume, Springer Japan, 2008.
7. Wolfson, H. J. and Rigoutsos, I., Geometric Hashing: An Overview, IEEE Computational Science & Engineering, 1997, vol.4(4), 10-21.
8. Iwamura, M., Kise, K. and Iwamura, M., Fast Camera-Based Document Image Retrieval Using Local Arrangements of Feature Points, IEICE Transaction, 2006, vol.J89-D(9), 2045-2054.
9. Horimatsu, A., Iwamura, M. and Kise, K., Fast Affine Invariant Shape Recognition Using Uniqueness of Arrangement of Points, IEICE Technical Report, 2009, vol.108(432), 127-132.
10. Iwamura, M., Iwamura, M. and Kise, K., On Accuracy and Speed of Object Recognition Based on Local Arrangements of Feature Points, IEICE Technical Report, 2006, vol.2006(93), 49-56.
11. Shoji, S. and Koyama, A., High –Speed Communication with Confidentiality by Compactification of Design Data, Proc. of NBiS2015, 2015, 124-131.