Journal of the Operations Research
Society of Japan
2009, Vol. 52, No. 4, 402-416

WATERMARKING FOR 3D POLYGONS USING WAVELET TRANSFORM AND MODIFIED TRAVELING SALESMAN PROBLEM

Shinichi Murata — *Panasonic Corporation*
Yasunari Yoshitomi — *Kyoto Prefectural University*
Hiroaki Ishii — *Osaka University*
Taro Asada — *Kyoto Prefectural University*

(Received March 22, 2007; Revised January 6, 2009)

Abstract Recent developments in computer technologies and networks have enabled the use of high-quality digital contents, such as electronic documents, images, and audio and video contents. A watermarking technique has recently been developed for the copyright protection of these digital contents. There are few watermarking techniques for three-dimensional (3D) geometric models, such as solid models or polygonal models. In the present paper, we propose an optimization problem for embedding watermarks on 3D polygon models. Then, using a genetic algorithm, we obtain the approximately optimum solutions for an original problem and a partial problem, which has an advantage with respect to security over the corresponding original problem. For embedding and extracting the watermark, we use a simplified method based on the previous study using the wavelet transform. The structure of the optimization problem examined in the present research is the same as that for the Traveling Salesman Problem, except for the objective function and the two additional constraints.

Keywords: Combinatorial optimization, CG, watermarking, wavelet transform, traveling salesman problem, genetic algorithm

1. Introduction

Recent developments in computer technologies and networks have enabled the use of high-quality digital contents, such as electronic documents, images, and audio and video contents, in both business and daily life. Recently, techniques for the copyright protection of digital contents are increasingly demanded, because digital contents may be illegally copied and widely distributed through computer networks without any degradation. A watermarking technique has recently been developed for the copyright protection of digital contents. Watermarking of digital contents is the process of embedding special signals, such as copyright data, into the digital contents so as to be imperceptible and to allow the detection of illegal use of the digital contents over computer networks.

Several techniques have recently been proposed for the watermarking of digital texts, images, and audio and video contents. In contrast, few techniques have been proposed for the watermarking of three-dimensional (3D) geometric models, such as surface models, solid models, or polygonal models. The 3D geometric model is one of the most fundamental techniques in the fields of computer-aided design (CAD), computer-assisted manufacturing (CAM), computer-aided engineering (CAE), and computer graphic (CG) animation, including several types of games. Home video games that use 3D geometric models are becoming more familiar to ordinary people. Therefore, there is a strong need to develop techniques to protect the copyright of the original 3D geometric model and to prevent unauthorized duplication or tampering. However, there has been little research [1, 2, 5, 7–10, 13] on the watermarking of 3D polygonal models. Theoretically, methods of watermarking 3D geometric models can be grouped into two categories: spatial domain methods[1, 2, 7, 8] and frequency domain methods[5, 9, 10, 13]. In previous studies, no special attention has been paid to the optimum distribution of the watermarking on the possible coefficients generated from the individual content.

In the present research, we use a watermarking method [2, 5], categorized as the frequency domain method, in which the wavelet transform (WT) and multi-resolution representation (MRR) of the 3D polygonal model [6, 11] are used to make the embedded watermark imperceptible and to make the control of the geometric error caused by the watermarking reliable. The purpose of the present study is to make an optimization problem for deciding the optimum distribution of the watermarking on the possible coefficients made from the individual content and to find an approximately optimal solution using a genetic algorithm (GA) [3, 4].

2. Requirements of Watermark for 3D Shape Model

The watermark is embedded into a digital content by minutely changing the information on original digital data. The present research focuses on the security of the digital data of the 3D shape model, and the requirements for the watermark are as follows.

(1) Perceptually invisible degradation on the digital content
The quality degradation caused by embedding the watermark must be perceptually invisible and unnoticeable to the user.

(2) High detectability
The embedded watermark can be extracted with the proper algorithm.

(3) Robustness against trial for removal
For instance, in the case that copyright data are embedded in the header part of the 3D model, it may be easy to remove the data. Therefore, the watermark information should be embedded into the geometrical data or topology data. Even if the 3D model is modified by some method, the watermark information should remain. Under 3D geometric operations such as noise addition, affine transformation, and cropping, the watermark information should be difficult to remove.

In general, the requirement listed in (1) might be in tradeoff relationship with that in (2) and/or (3).

3. Multi-resolution Wavelet Decomposition for the 3D Polygon Model

According to references [2, 5], the mathematical formulation of WT for the 3D polygonal model [6, 11] is described in this section. In this theory, it is assumed that all of the faces in a polygon mesh must be triangles. The basic concept is demonstrated in Figure 1. The high-resolution polygon is approximated by a low-resolution polygon through a 4-to-1 subdivision connectivity scheme [6], by which each detailed part is expressed as a wavelet coefficient vector (WCV). This WT technique is used in the watermark-embedding scheme in the present study. We can formulate the WT for the 3D polygon model using the lazy wavelet[11] as:

$$\mathbf{V}^{j+1} = \mathbf{A}^{j+1}\mathbf{V}^{j} \tag{3.1}$$

$$\mathbf{W}^{j+1} = \mathbf{B}^{j+1}\mathbf{V}^{j}, \tag{3.2}$$

where $\mathbf{V}^j$ is the matrix of which the row corresponds to the vertex coordinates at a resolution of j (high resolution), $\mathbf{V}^{j+1}$ is the matrix of which the row corresponds to the vertex

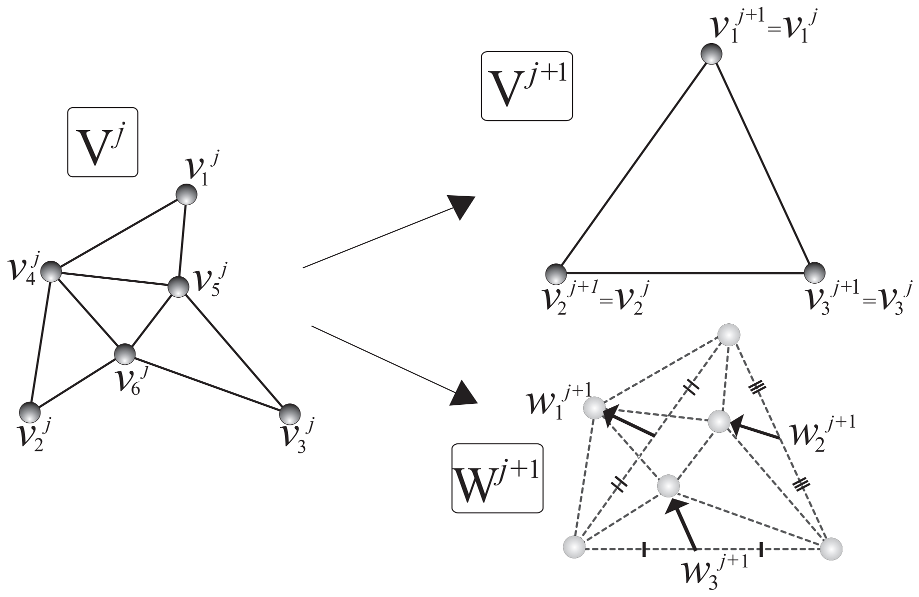

Figure 1: Concept of the WT

coordinates at a resolution of $j+1$ (low resolution), and $\mathbf{W}^{j+1}$ is the matrix of which the row corresponds to a WCV at a resolution of $j+1$.

According to Equations (3.1) and (3.2), the coarsest representation $\mathbf{V}^d$ of the polygon and the corresponding WCVs $\mathbf{W}^d$ can be obtained as:

$$\mathbf{V}^d = \mathbf{A}^d\mathbf{A}^{d-1}\mathbf{A}^{d-2}\ldots\mathbf{A}^2\mathbf{A}^1\mathbf{V}^0 \tag{3.3}$$
$$\mathbf{W}^d = \mathbf{B}^d\mathbf{A}^{d-1}\mathbf{A}^{d-2}\ldots\mathbf{A}^2\mathbf{A}^1\mathbf{V}^0 \tag{3.4}$$

The matrices $\mathbf{A}^{j+1}$ and $\mathbf{B}^{j+1}$ are called analysis filters at resolution level $j+1$. $\mathbf{A}^{j+1}$ is the non square matrix that characterizes the triangle reduction obtained by merging four triangles into one triangle. For the example of the triangles shown in Figure 1, $\mathbf{A}^{j+1}$ is defined as:

$$\mathbf{A}^{j+1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{3.5}$$

$\mathbf{B}^{j+1}$ is the non square matrix and produces the WCVs, each of which starts from the midpoint of the edge in the low-resolution triangle and ends at the vertex, which is lost at the lower resolution (Figure 1). For the example of the triangles in Figure 1, $\mathbf{B}^{j+1}$ is defined as:

$$\mathbf{B}^{j+1} = \frac{1}{2}\begin{bmatrix} -1 & -1 & 0 & 2 & 0 & 0 \\ -1 & 0 & -1 & 0 & 2 & 0 \\ 0 & -1 & -1 & 0 & 0 & 2 \end{bmatrix} \tag{3.6}$$

In contrast, the inverse wavelet transform (IWT) can be formulated as:

$$\mathbf{V}^j = \mathbf{P}^{j+1}\mathbf{V}^{j+1} + \mathbf{Q}^{j+1}\mathbf{W}^{j+1}, \tag{3.7}$$

where $\mathbf{P}^{j+1}$ and $\mathbf{Q}^{j+1}$ are referred to as synthesis filters. The synthesis consists of the following two steps: splitting the triangle of the low-resolution polygon into four sub-triangles by introducing the new vertices at edge midpoints using $\mathbf{P}^{j+1}$, and correcting the position of these new vertices according to the WCVs using $\mathbf{Q}^{j+1}$. By using Equations (3.1), (3.2), and (3.7) recursively, the decomposition and multi-resolution representation of the 3D polygonal models are obtained as demonstrated in Figure 2.

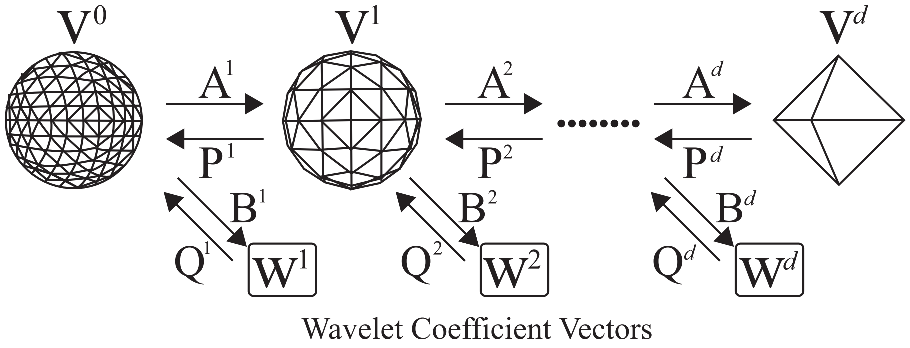

Figure 2: Wavelet based decomposition and multi-resolution representation of the 3D polygonal model

The mathematical relation between the matrices of analysis filter and synthesis filter is given as follows:

$$\left[\mathbf{P}^{j+1}\,\middle|\,\mathbf{Q}^{j+1}\right] = \left[\frac{\mathbf{A}^{j+1}}{\mathbf{B}^{j+1}}\right]^{-1} \tag{3.8}$$

For the example of Figure 1, we can obtain Equation (3.9), according to Equations (3.5), (3.6), and (3.8).

$$\left[\mathbf{P}^{j+1}\,\middle|\,\mathbf{Q}^{j+1}\right] = \frac{1}{2}\left[\begin{array}{ccc|ccc} 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \end{array}\right] \tag{3.9}$$

4. Overview of the Embedding and Extraction Algorithm of the Watermark

The algorithm of the embedding and extracting of the watermark, based on the method by Kanai et al. [2, 5], is explained in this section.

4.1. Outline

The procedure of embedding the watermark is as follows.

1. Original polygon data $\mathbf{V}^0$, the watermark data $\mathbf{B}$ expressed by a binary sequence, and the control parameters for embedding are prepared.
2. The wavelet transform expressed by Equations (3.1) and (3.2) is iteratively applied to $\mathbf{V}^0$ until the resolution level d, so that the WCVs at each resolution and the coarsest approximation $\mathbf{V}^d$ are obtained.
3. The watermark is embedded for the WCVs at the coarsest resolution level, according to the algorithm described below.
4. Then, the watermarked polygon data $\hat{\mathbf{V}}^0$ are obtained through the IWT from the coarsest resolution level to level 0.

The following control parameters for embedding are preset by the user.
k: bit length per WCV that can be embedded in the watermark,
δ: lower bound of the original WCV norm for invisible embedding,
r: highest number of decimal places for embedding,
d: coarsest resolution level of the WT.

4.2. Embedding algorithm

4.2.1. Selective watermark embedding for achieving perceptual invisibility

The existence of the watermark on the 3D polygonal model should be perceptually invisible. The human eye is not sensitive to tiny geometric changes in rough areas. The norm of the WCV indicates the degree of roughness of the small area near the WCV. The larger norm of the WCV means more bumpiness near the WCV. Therefore, in order to embed the watermark invisibly, $\mathbf{w}_i^d$, the norm of which satisfies $\left|\mathbf{w}_i^d\right| > \delta$, is selected as the WCV for embedding the watermark, where δ is the user-specified lower bound for the norm of the WCV for embedding.

4.2.2. Preservation of the watermark under affine transformation

In the present study, the ratio α_i^d of the norm $\left|\mathbf{w}_i^d\right|$ of a WCV to the length l_i^d of the corresponding edge $\overline{\mathbf{v}_a^d\mathbf{v}_b^d}$ is modified in the process of embedding the watermark. This ratio is selected because it is invariant to the affine transformation, which is often performed in CG space. The edge $\overline{\mathbf{v}_a^d\mathbf{v}_b^d}$ on which the start point of the WCV $\mathbf{w}_i^d$ lies at its midpoint is selected as a partner (Figure 3), and the ratio α_i^d defined in Equations (4.1) and (4.2) is modified to $\hat{\alpha}_i^d$ defined in Equations (4.2) and (4.3) by the method described below.

$$\alpha_i^d = \left|\mathbf{w}_i^d\right| \big/ l_i^d \tag{4.1}$$

$$l_i^d = \left|\mathbf{v}_a^d - \mathbf{v}_b^d\right| \tag{4.2}$$

$$\hat{\alpha}_i^d = \left|\hat{\mathbf{w}}_i^d\right| \big/ l_i^d \tag{4.3}$$

The WCV $\mathbf{w}_i^d$ is modified to $\hat{\mathbf{w}}_i^d$ by Equation (4.4). The modified WCV $\hat{\mathbf{w}}_i^d$ is used at the IWT to obtain the watermarked polygon data.

$$\hat{\mathbf{w}}_i^d = (\hat{\alpha}_i^d l_i^d)\mathbf{w}_i^d \big/ \left|\mathbf{w}_i^d\right|, \tag{4.4}$$

where $\hat{\alpha}_i^d$ is obtained by the method described in Section 4.2.3.

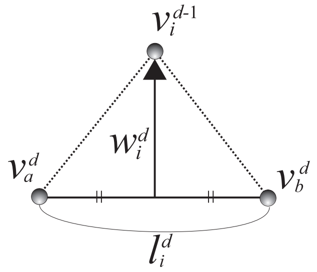

Figure 3: Ratio of the WCV to the partner edge

4.2.3. Embedding a real number value

Watermark embedding is physically performed by modifying the vertex coordinates of the polygon model. Here, the geometric error caused by the watermark embedding should be kept as small as possible under the condition that the watermark is almost perfectly extracted.

Next, a binary sequence $(b_1, b_2, \cdots\cdots, b_k)$ of length k in watermark $\mathbf{B}$ is transformed into the decimal digits β_i^d to embed the signal using Equation (4.5), which is the simplified version of equation obtained from Reference[2, 5].

$$\beta_i^d = b_1 \times 10^r + b_2 \times 10^{r-1} + \cdots + b_k \times 10^{r-k-1} \tag{4.5}$$

Then, the watermarked ratio $\hat{\alpha}_i^d$ is obtained by Equation (4.6)

$$\hat{\alpha}_i^d = \alpha_i^d + \beta_i^d \tag{4.6}$$

4.2.4. Watermark extraction algorithm

First, the WT is iteratively applied to both the watermarked polygon data and the original polygon data until reaching resolution level d, at which the watermark was embedded. The ratios for the watermarked polygon data and the original polygon data are obtained by Equations (4.1) and (4.3), respectively, and their difference is calculated by Equation (4.7).

$$\beta_i^d = \hat{\alpha}_i^d - \alpha_i^d \tag{4.7}$$

Using Equation (4.5), a binary subsequence of length k is extracted from each β_i^d. Using this procedure for the WCVs selected for embedding the watermark at the coarsest level, the watermark is extracted. This extraction algorithm requires the original polygon model.

5. Formulation of Problem

In order to express the watermarked polygon model, we must use all of the information on the connection of vertices for constructing triangles in the original polygon model in CG space. On the other hand, all of the information on the connection of vertices for embedding a watermark can be treated as being independent of that in CG space. Therefore, in order to obtain the watermarked polygon model having small errors in all vertices while maintaining high detectability of the watermark as well as perceptual invisibility, we can formulate an optimization problem as a connection-decision problem of vertices for embedding a watermark. In addition, the connection decision problem of vertices is equivalent to an order-decision problem of vertices under the condition of using the WT in the present study.

The objective function $e(X)$ is a square sum of the coordinate error in all vertices caused by the watermark embedding. The constraint is imposed on the watermark detection rate $d(X)$ in the watermarked polygon data and the relative error on the coordinate of each vertex of the watermarked polygon data. The optimization problem P for minimizing the objective function is formulated as follows:

$$\text{P} \quad \text{Minimize} \quad e(X) \tag{5.1}$$

$$\text{Subject to} \quad d(X) \geq a \tag{5.2}$$

$$(x_i - x_i')^2 + (y_i - y_i')^2 + (z_i - z_i')^2 < \frac{be(X)}{N'} \tag{5.3}$$

$$\sum_P \sum_i \sum_{j \neq i} x_{Pi} x_{Pj} = 0 \tag{5.4}$$

$$\sum_i \sum_P \sum_{Q \neq P} x_{Pi} x_{Qi} = 0 \tag{5.5}$$

$$\sum_{P}\sum_{i} x_{Pi} = N \tag{5.6}$$

$$e(X) = \sum_{i=1}^{N} \{(x_i - x_i')^2 + (y_i - y_i')^2 + (z_i - z_i')^2\} \tag{5.7}$$

$$X = X(x_{ij}), x_{ij} \in \{0, 1\}$$

$$P = 1, 2, \cdots, N,\ Q = 1, 2, \cdots, N,\ i = 1, 2, \cdots, N,\ j = 1, 2, \cdots, N,$$

where (x_i, y_i, z_i) and (x_i', y_i', z_i') express the coordinate of vertex i in the original and watermarked polygon data, respectively, N is the total number of vertices for distributing the watermark in the original polygon data, N' is the total number of vertices whose coordinates are influenced by embedding the watermark, and a and b are constants. When the coordinate of vertex i is selected as the element of row j in making the matrix $\mathbf{V}^0$ expressing original polygon data, an element x_{ij} of a matrix X is equal to 1. Otherwise, x_{ij} is equal to 0. As expressed by Equation (5.4), each vertex is selected only once in giving its coordinate as the element of the row of the matrix $\mathbf{V}^0$. As expressed by Equation (5.5), only one vertex is selected in giving one coordinate as the element of the row of the matrix $\mathbf{V}^0$. Moreover, as expressed by Equation (5.6), all vertices are selected in giving coordinates as the elements of the rows of the matrix $\mathbf{V}^0$. Therefore, Equations (5.4)$\sim$(5.6) express the same restriction as that of the Traveling Salesman Problem. In the present study, the watermark is embedded using the matrix $\mathbf{V}^0$, the elements of which consist of the three-dimensional coordinates of all vertices. The matrix $\mathbf{V}^0$ is decided by the matrix X. In other words, the matrix X decides the order of vertices for which the coordinates correspond to the elements of the row vector in the matrix $\mathbf{V}^0$. Optimizing the matrix X is equivalent to optimizing the matrix $\mathbf{V}^0$. (x_i, y_i, z_i) are given in advance, whereas (x_i', y_i', z_i') are decided by the matrix X.

As described in Section 3, the WT in the present study is equivalent to the operation by which the three vertices forming one triangle are selected in numeric order among the six vertices grouped as the row vectors in the matrix $\mathbf{V}^l (0 \leq l \leq d-1)$ and forming the four triangles. The connection of vertices to form each triangle at the WT and the IWT is decided by the numeric order of the corresponding row in matrix $\mathbf{V}^0$. Accordingly, the number of vertices used for the WT should be a multiple of six, even if the total number of vertices composing the polygon model is not a multiple of six.

In performing the WT for $\mathbf{V}^l (1 \leq l \leq d-1)$, the number of vertices should also be a multiple of six in this research. Therefore, five or fewer vertices from the end of the current numeric order decided by the matrix $\mathbf{V}^0$ might not be used in performing the WT because a number of vertices that is a multiple of six should be adopted. Independently of $\mathbf{V}^0$, all information on the connection of vertices for making triangles in the original polygon model are inherited for expressing the watermarked polygon model in CG space.

We can keep $\mathbf{V}^0$ corresponding to the (approximately) optimum solution and some parameters for calculation secret. Moreover, when the vertex increases, the optimization problem P causes the combination explosion. Accordingly, it is not easy to extract or eliminate the watermark by considering all cases for embedding the watermark. However, since it is fundamentally possible to obtain the optimum solution or the approximately optimum solution of P, the use of P might be not a good strategy for achieving high security by the watermark. As one strategy for improving security by a watermark, a partial problem P' is generated with a feasible solution of P, which defines a solution space of P'. The feasible solution takes the role of a secret key. If the secret key is unknown, it is much more difficult to extract or eliminate the watermark. A partial problem P’ is formulated as follows.

$$
\begin{aligned}
\mathrm{P}' \quad & \text{Minimize} \quad e(s) && (5.8)\\
& \text{Subject to} \quad d(s) \geq a && (5.9)\\
& (x_i - x_i')^2 + (y_i - y_i')^2 + (z_i - z_i')^2 < \frac{be(s)}{N'} && (5.10)\\
& x_{i+s,j} = c_{ij} && (5.11)\\
& \sum_P \sum_i \sum_{j \neq i} c_{Pi} c_{Pj} = 0 && (5.12)\\
& \sum_i \sum_P \sum_{Q \neq P} c_{Pi} c_{Qi} = 0 && (5.13)\\
& \sum_P \sum_i c_{Pi} = N && (5.14)\\
& e(s) = \sum_{i=1}^{N} \{(x_i - x_i')^2 + (y_i - y_i')^2 + (z_i - z_i')^2\} && (5.15)\\
& C = C(c_{ij}),\ c_{ij} \in \{0,1\},\ X = X(x_{ij}),\ x_{ij} \in \{0,1\}\\
& P = 1, 2, \cdots, N,\ Q = 1, 2, \cdots, N,\ i = 1, 2, \cdots, N,\ j = 1, 2, \cdots, N\\
& s = 0, 1, \cdots, N-1, x_{N+k,j} = x_{k,j}, k = 1, 2, \cdots, N-1,
\end{aligned}
$$

where e, d, (x_i, y_i, z_i), (x_i', y_i', z_i'), N, N', a, b, and x_{ij} are the same as those individually described for P. The objective function $e(s)$ and the function $d(s)$ in the constraint are calculated with an integer valuable s, described in Equation (5.11). All values of c_{ij} that satisfy Equations (5.12)∼(5.14) are given in advance. As described by Equations (5.4)∼(5.6), and Equations (5.11)∼(5.14), P’ is a partial problem for P. The matrix X also satisfies Equations (5.4)∼(5.6). The matrix $\mathbf{V}^0$ is decided by the matrix X. In other words, the matrix $\mathbf{V}^0$ is decided by a matrix C and an integer s. As described in Section 3, the WT in the present study is equivalent to the operation by which the three vertices forming one triangle are selected in numeric order among the six vertices grouped as the row vectors in the matrix $\mathbf{V}^l (0 \leq l \leq d-1)$ and forming the four triangles. We randomly choose one matrix as C among the several matrixes that satisfy Equations (5.12)∼(5.14). Therefore, an integer valuable s decides the values of both e and d.

In finding the watermarked WCVs without knowing a feasible solution (matrix C) for P, the number of cases for searching the $\mathbf{V}^0$ is $N!$, where N is the number of vertices of polygon data, whereas that with knowing the matrix C is N. When the third party tries to extract or eliminate the watermark without knowing the matrix C, the calculation cost can be estimated as $(N-1)!$ calculations, as compared with that with knowing the matrix C. Therefore, the matrix C has the role of a secret key.

6. GA Strategy

We use a GA to find an approximately optimum solution for P. We use a GA strategy in which the genotype equals the phenotype. An individual is expressed with a path representation, in which each gene denotes one vertex number. In the present study, a path representation is an order of vertices in which the coordinates correspond to the elements of a row vector in the matrix $\mathbf{V}^0$. Therefore, an individual expressed by a path representation decides a matrix $\mathbf{V}^0$. The fitness function f is defined as

$$f = \frac{1}{e(X)}, \tag{6.1}$$

where $e(X)$ is defined by Equation (5.7). The roulette wheel selection is used. The subtour exchange crossover [12] is used. The most important feature of the present study is the formation of the problem as the modified TSP. We do not focus on the performance of crossover. We choose the subtour exchange crossover because it is relatively easy to make a program using the subtour exchange crossover. The mutation is performed by a procedure in which two vertices are randomly selected and exchanged in the path representation.

We also use a GA to find an approximately optimum solution for P'. We use a GA strategy in which the genotype equals the phenotype. An individual is composed of two parts. The first part expresses a number for distinguishing some matrixes randomly generated and used as C, whereas the second part denotes an integer s defined by Equation (5.11). The first and second parts are expressed in binary notation. The matrix $\mathbf{V}^0$ is decided by a matrix C and an integer s. The fitness function f' is defined as

$$f' = \frac{1}{e(s)}, \tag{6.2}$$

where $e(s)$ is defined by Equation (5.15). The roulette wheel selection is used. Two-point crossover is performed with a pair of parent-individuals with certain probability. The division point is selected randomly. At the division points, the chromosome is divided into three parts, and each part of the chromosome is then combined with that from a different parent in order to obtain a pair of child-individuals. One point mutation is then performed with certain probability. The mutation gene is randomly selected. The allele of the selected gene is exchanged for another.

7. Numerical Experiments

7.1. Condition

The experimental environment was a Dell Dimension 5150C (CPU: Pentium IV 3.0 GHz, memory: 1 GB) running Windows XP as the OS and Visual C++ as the programming language. Three 3D polygon models are prepared, and the method described in Section 4 is used for embedding and extracting watermarks. A monoclonal BMP image having 20×20 pixels is used as a watermark. The WT is performed to level $d = 2$ for models of Norm[14, 15] and Miss Japan[14, 15] and to level $d = 3$ for the model of Robot. As other control parameters for embedding watermarks, $k = 4$, $\delta = 0.8\left|\overline{\mathbf{w}_i^d}\right|$, and $r = 2$ are used. The models Norm, Miss Japan, and Robot have 762, 762, and 2,839 vertices, respectively. For Norm, Miss Japan, and Robot, 762, 762, and 2,838 vertices, respectively, are used for optimization when embedding the watermark. In the present study, the total number of vertices of the model should be a multiple of six. Therefore, in the case of Robot, one vertex having the last number on the original indices is not used for optimizing the embedding of the watermark.

For P, an individual has a number of vertices that equals the number of genes. Therefore, the length of chromosome is 762, 762, and 2,838 for Norm, Miss Japan, and Robot, respectively.

For P', an individual has 15 bits as genes. The first part, composed of the top five bits in binary notation, is a number for distinguishing some randomly generated matrixes and used as C, while the second part composed of the subordinate 10 bits expresses an integer s in binary notation.

As the conditions of the GA for P, the individual of 10, the generation of 5,000, the crossover probability of 0.4, and the mutation probability of 0.2 are used. As the conditions of the GA for P', the individual of 100, the generation of 100, the crossover probability of 0.6, and the mutation probability of 0.1, are used. Here, $a = 0.95$ and $b = 1.2$ are used as parameters for P and P'. The values used in the experiments are decided by trial and error through several preliminary experiments. In addition, in the case of P, the values of GA parameters used in Reference [12] are also taken into consideration.

For the comparison with the performance of the reported method [2, 5], the approximately optimum solution for P' is obtained using the program [16] used in the reported study [2, 5] and the GA with the same condition as that mentioned above. The polygon data of HEAD [16] used in the reported study [2, 5] is used in the comparison. The polygon model has 1,026 vertices. The watermark used in the comparison is the same as that mentioned above. As the control parameters for embedding watermarks, $\varepsilon = 10^{-2}$, $d = 4$ and $k = 5$ which were adopted in References [2, 5] are used. For further information on these parameters, see References [2, 5]. Instead of the Dell Dimension 5150C (CPU: Pentium IV 3.0 GHz, memory: 1 GB), Dell OptiPlex 755 (CPU: Intel Core 2 Duo E6750 2.66 GHz, memory: 2 GB) is used only for the comparison with the conventional method.

7.2. Results and discussions

In this subsection, in order to examine the performances of the proposed techniques, the error in the shape of the watermarked polygon models obtained from the approximately optimum solution for the original and partial problems is first compared with that obtained from 100 feasible solutions generated at random for the original and partial problems. Next, the time to obtain the approximately optimum solution is compared between the original and partial problems.

The changes in the best value in the population for each generation during the experiments are shown in Figure 4. It takes much longer to achieve convergence in the case of P, as compared with P'. The approximately optimum solutions defined as the feasible solution giving the best objective function value during all generations for each problem are used for embedding the watermark for each polygon model.

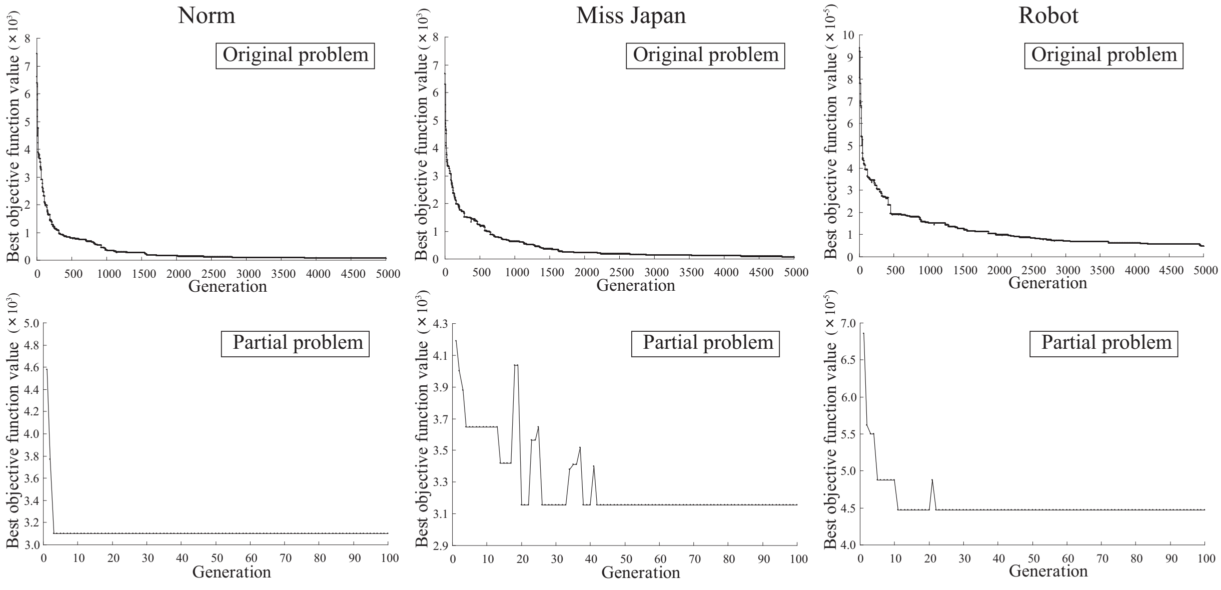

Figure 4: Convergence of the GA for the original and partial problems

Figure 5 shows a comparison among the original model, the model embedded with P (original problem), and the model embedded with P' (partial problem) on the wire frame model, and the model after texture mapping. In Figure 5, each extracted watermark is also shown. The detection rate is 1.0 for the models of Norm and Miss Japan in both cases of P and P'. The detection rate is 1.0 for the model of Robot in the case of P' and is 0.9975 in the case of P. In the GA process, many feasible solutions by which the detection rate is 1.0 are obtained for the model of Robot in the case of P. Among these solutions, we can select the feasible solution that presents the smallest error. In addition, when we use the detection rate as the objective function or one of the objective functions of the multi-objective optimization problem, we may be able to obtain an approximately optimum solution by which the detection rate is 1.0. As shown in Figure 5, the influence of the watermark on the 3D models is undetectable.

Figure 6 shows a comparison of the error in shape caused by the watermark between the approximately optimum solution and the 100 feasible solutions generated at random in both cases of P and P'. As shown in Figure 6, the GA can find an acceptable solution in both cases for P and P'. Moreover, in all three experiments, the approximately optimum solution for P is much better than that for P'. In some cases in which the matrix C used for P' might not be appropriate, the approximately optimum solution for a partial problem could cause visible deterioration of the model. For example, in case of the Norm model, Figure 7 shows the case in which the approximately optimum solution for a partial problem is not acceptable because the deterioration caused by the watermark is visible. In such a case, the matrix C is newly generated and the approximately optimum solution is searched. A reduction in the value of b in Equations (5.3), (5.10) in the range larger than 1.0 might be also effective for avoiding the deterioration shown in Figure 7.

The time to obtain the approximately optimum solution for the three polygon models is shown in Table 1. Much more time is required for P, as compared with P'. It is natural that the approximately optimum solution for the original problem can be better than that for a partial problem and that the time to obtain that for the original problem might be much more than that for a partial problem. Therefore, a tradeoff relationship exists between the quality of the solution and the cost to obtain it. In addition to this general tradeoff relationship, the approximately optimum solution for a partial problem has an advantage in security over that for the original problem. Since it is easier for an expert third party to find an approximately optimum solution to the original problem, as compared to finding an approximately optimum solution to a partial problem without knowing the matrix C. If an approximately optimum solution might be found by a third party, the watermark could be extracted or erased more easily, as compared to the case in which no approximately optimum solution can be found. For this reason, the use of P' as an example of a partial problem has an advantage in security as compared to the use of P.

Table 1: Comparison of time (sec.) to obtain an approximately optimum solution

	Original problem	Partial problem
Norm	1.62×10^4	1.34×10^2
Miss Japan	1.65×10^4	1.50×10^2
Robot	7.31×10^5	7.85×10^2

The detection rate is 1.0 for the model of HEAD in both cases with P' and the conventional method. Figure 8 shows a comparison among the original model, the model embedded with P', and the model embedded with the conventional method on the wire frame model. As shown in Figure 8, the influence of the watermark on the 3D models is undetectable.As

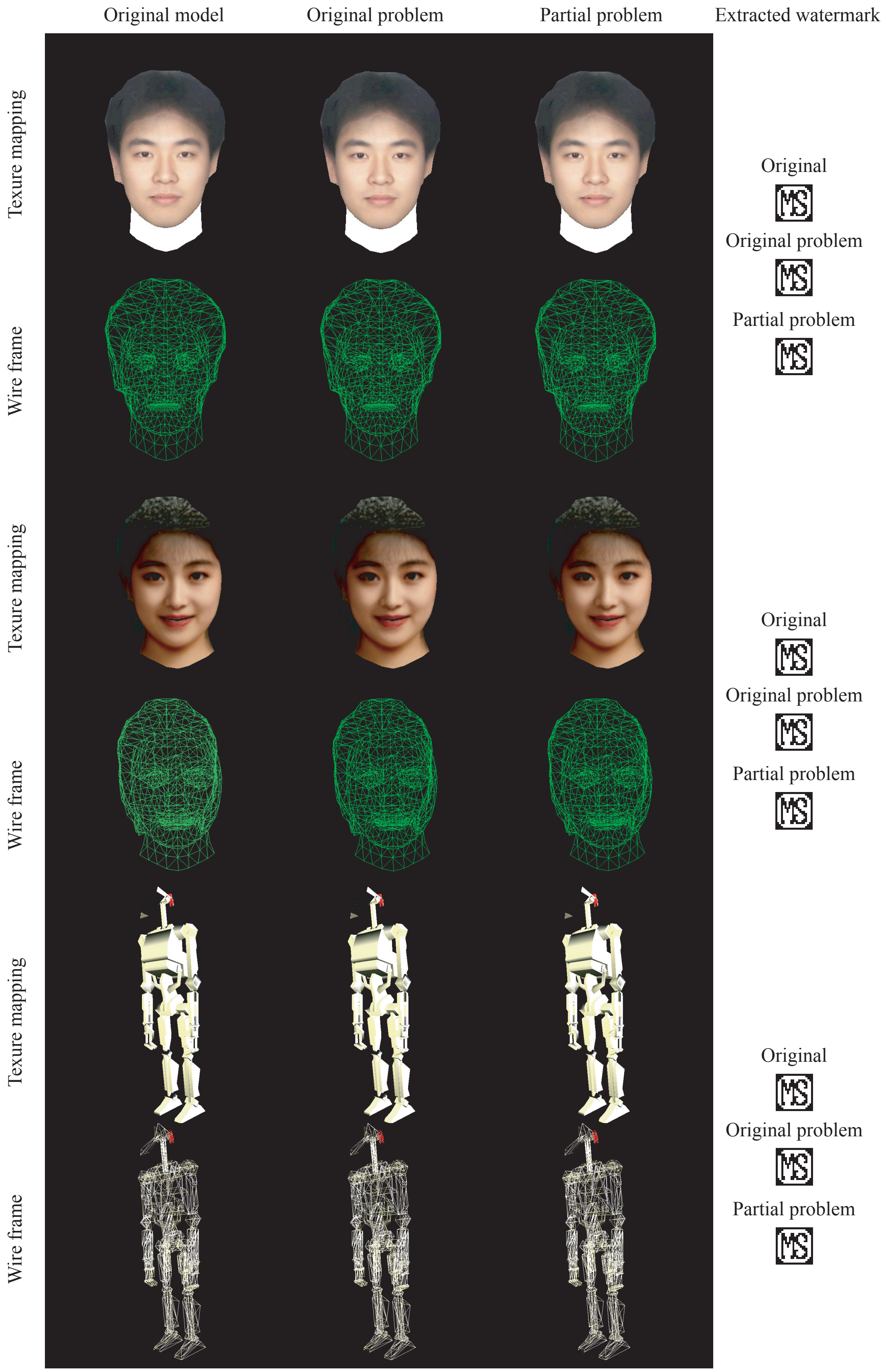

Figure 5: Comparison of the models before and after embedding the watermark. Upper: Norm, Middle: Miss Japan, Lower: Robot

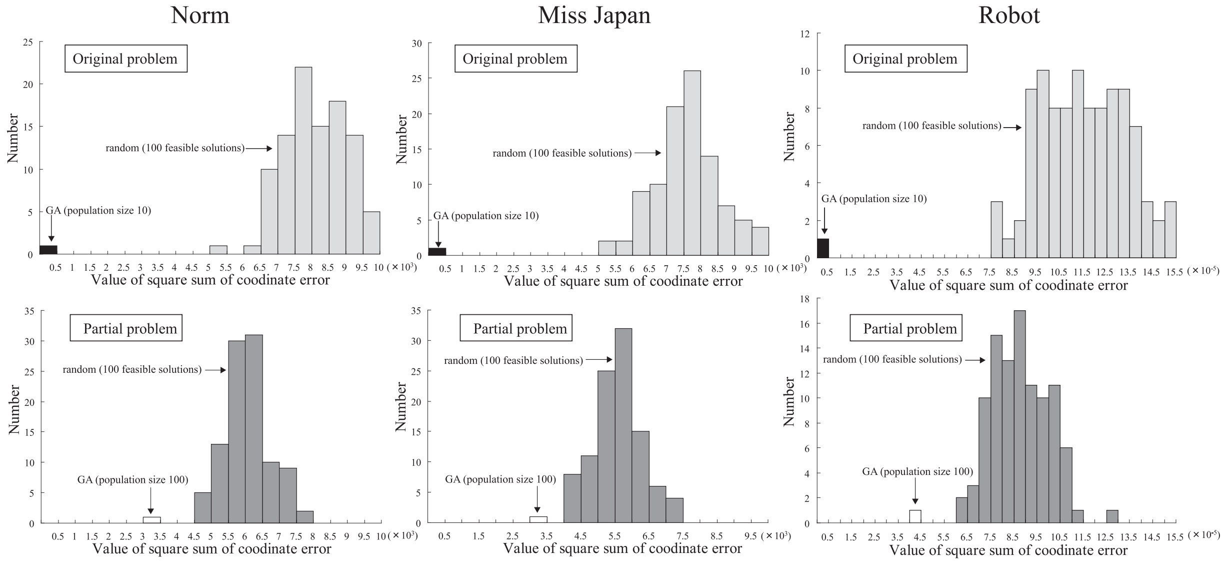

Figure 6: Comparison between the approximately optimum solution and the 100 feasible solutions generated at random

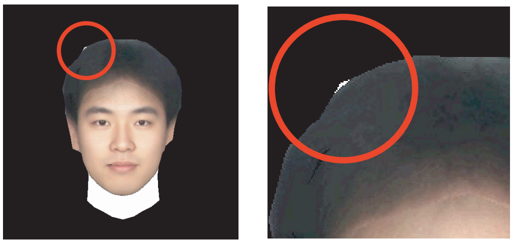

Figure 7: Deterioration of the model

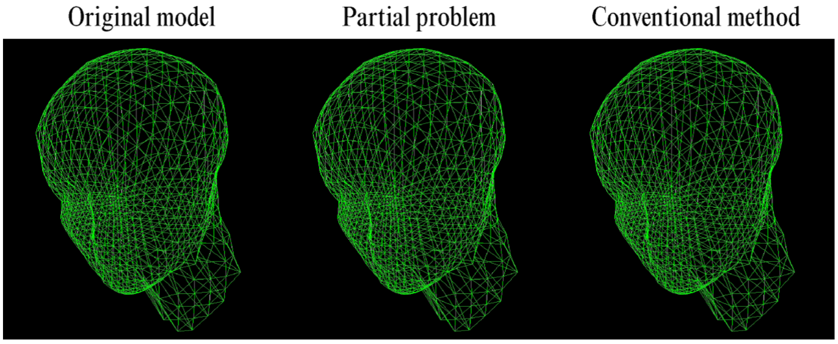

Figure 8: Comparison of the models (HEAD) before and after embedding the watermark

shown in Table 2, the square sum of the coordinate error in all vertices caused by the watermark embedding which is obtained as the best objective function value for P' by GA is much smaller than that obtained by the conventional method. As shown in Table 2, even in an initial population, we can get a good feasible solution with which the square sum of the coordinate error in all vertices caused by the watermark embedding is much smaller than that obtained by the conventional method.

Table 2: Comparison of the square sum of the coordinate error in all vertices caused by the watermark embedding

Partial problem	
(GA)	2.24×10^{-5}
(Best value in an initial population)	2.78×10^{-5}
(Average value in an initial population)	4.70×10^{-5}
Conventional method	7.29×10^{-5}

8. Conclusions

In the present paper, we proposed an optimization problem for embedding a watermark on the 3D polygon model. Then, using a genetic algorithm, we obtained an approximately optimum solution for an original problem and a partial problem, which has an advantage in security over the original problem. The proposed method is based on a previous method that uses the WT for embedding and extracting the watermark. The structure of the optimization problem considered in the present study is the same as that of the Traveling Salesman Problem, except for the objective function and two additional constraints. The effectiveness of the proposed method was confirmed using numerical results.

Acknowledgement

The authors would like to thank Associate Professor H. Okuhara of the Graduate School of Osaka University for his valuable advice on this research. The authors would also like to thank Associate Professor M. Tabuse of Kyoto Prefectural University for his valuable support of this research. The authors would also like to thank Professor S. Kanai and Assistant Professor H. Date of the Graduate School of Hokkaido University for their supplying us a source program and a polygon data which were used in their study [2, 5].

References

[1] O. Benedens: Geometry-based watermarking of 3D models. *IEEE Computer Graphics and Applications*, **19** (1999), 46–55.

[2] H. Date, S. Kanai and T. Kishinami: Digital watermarking for 3D polygons based on wavelet transform. *Journal of the Japan Society of Precision Engineering*, **65** (1999), 1275–1279 (in Japanese).

[3] D. Goldberg: *Genetic Algorithm in Search, Optimization, and Machine Learning* (Addison-Wesley, 1989).

[4] J. H. Holland: *Adaptation in Natural and Artificial Systems* (The Univ. Michigan Press, 1975 and MIT press, 1992).

[5] S. Kanai, H. Date and T. Kishinami: Digital watermarking for 3D polygons using multiresolution wavelet decomposition. *Proceedings of 6th IFIP WG 5.2 GEO-6* (1998), 296–307.

[6] M. Lounsbery: Multiresolution analysis for surfaces of arbitrary topological type. PhD thesis, University of Washington (1994).

[7] R. Ohbuchi, H. Masuda and M. Aono: Watermarking three-dimensional polygonal models. *Proceedings of ACM Multimedia '97* (1997), 261–272.

[8] R. Ohbuchi, H. Masuda and M. Aono: Watermarking three-dimensional polygonal models through geometric and topological modification. *IEEE Journal on Selected Areas in Communication*, **16** (1998), 551–560.

[9] R. Ohbuchi, A. Mukaiyama S. Takahashi: A frequency domain approach to watermarking 3D shapes. *Computer Graphics Forum*, **21** (2002), 373–382.

[10] E. Praun, H. Hoppe and A. Finkelstein: Robust mesh watermarking. *Proceedings of SIGGRAPH'99* (1999), 49–56.

[11] E. J. Stollnitz, T. DeRose and D. H. Salesin: *Wavelets for Computer Graphics: Theory and Applications* (Morgan Kaufmann Pub, 1996).

[12] M. Yamamura, T. Ono and S. Kobayashi: Character-preserving genetic algorithms for traveling salesman problem. *Journal of the Japan Society of Artificial Intelligence*, **6** (1992), 1049–1059 (in Japanese).

[13] K. Yin, Z. Pan, J. Shi and D. Zhang: Robust mesh watermarking based on multiresolution processing. *Computer & Graphic*, **25** (2001), 409–420.

[14] Facial image processing system for human-like "Kansei" agent: Advanced agent homepage, *http://www.hc.ic.i.u-tokyo.ac.jp/project/face/IPA/*

[15] HeikinTool: Average face synthesis tool homepage, *http://www.hc.ic.i.u-tokyo.ac.jp/project/face/HeikinTool/*

[16] Provided by S. Kanai and H. Date

Yasunari Yoshitomi
Department of Informatics and Environmental Science
Kyoto Prefectural University
1-5, Nakaragi-cho, Shimogamo, Sakyo-ku, Kyoto
606-8522 Japan
E-mail: yoshitomi@kpu.ac.jp