Image Based Coding of Spatial Probability Distribution on Human Dynamics Data

Hideaki KIMATA^{†a)}, Senior Member, Xiaojun WU[†], and Ryuichi TANIDA[†], Members

SUMMARY The need for real-time use of human dynamics data is increasing. The technical requirements for this include improved databases for handling a large amount of data as well as highly accurate sensing of people's movements. A bitmap index format has been proposed for high-speed processing of data that spreads in a two-dimensional space. Using the same format is expected to provide a service that searches queries, reads out desired data, visualizes it, and analyzes it. In this study, we propose a coding format that enables human dynamics data to compress it in the target data size, in order to save data storage for successive increase of real-time human dynamics data. In the proposed method, the spatial population distribution, which is expressed by a probability distribution, is approximated and compressed using the one-pixel one-byte data format normally used for image coding. We utilize two kinds of approximation, which are accuracy of probability and precision of spatial location, in order to control the data size and the amount of information. For accuracy of probability, we propose a non-linear mapping method for the spatial distribution, and for precision of spatial location, we propose spatial scalable layered coding to refine the mesh level of the spatial distribution. Also, in order to enable additional detailed analysis, we propose another scalable layered coding that improves the accuracy of the distribution. We demonstrate through experiments that the proposed data approximation and coding format achieve sufficient approximation of spatial population distribution in the given condition of target data size.

key words: human dynamics data, probability distribution, scalable coding, non-linear mapping, multimedia database

1. Introduction

Traditionally, human dynamics data has been widely used to clarify and analyze the economic and cultural activities of people in the real world. Human dynamics data mainly consists of human location information, age and gender. For example, the data obtained from the Japanese census is human dynamics data. In recent years, the means for acquiring human dynamics data has become mechanized to the point that near real-time acquisition is now possible. The use of mobile devices is attracting attention as a means of acquiring human location information. In addition to mobile devices acquiring their own location information from GPS and providing it to services, mobile communication base stations are also used to acquire rough location information for each device. Currently, real-time human dynamics data via mobile communication is beginning to be provided while simultaneously giving consideration to privacy. It is expected

to be used for traffic congestion prediction and to examine evacuation measures [1]–[4]. The supposed application scenario of our study is searching queries in database of real-time human dynamics data and analyze, which is similar to the Terada's study [3].

When human dynamics data are collected in real time, the amount of data becomes enormous because they are continuously collected at short time intervals, so studies on large-scale storage and database technology for enabling such storage are currently underway [5]–[10]. In particular, for multi-dimensional data, an indexing method that can be read out and utilized at high speed in consideration of the 2D (two-dimensional) spatial structure on positions is being investigated. However, in order to store such a huge amount of data with attributes of time for real-time human dynamics data, it is also necessary to compress the data in a desired size. Therefore, not only indexing but also suitable data formats are required. In this paper, we propose a coding format that enables the spatial distribution to fit in the target data size. In the proposed method, the spatial distribution is represented by a probability distribution, and the probability distribution is converted into image data. The data are expressed in the one-byte format normally used for image coding. When expressing with this format, we apply two kinds of approximation for spatial distribution: accuracy of probability and precision of spatial location. Under these approximations, the amount of information and the number of codes can be controlled by parameters so that the data size confirms to the desired size. We also propose a layered structure that can be provided while updating the spatial distribution of people's positions by applying the scalable data structure used for image coding.

2. Related Work

There has been much prior research on speeding up database indexes. There are many efforts to structure data according to data categories, such as by using a k-d tree. Many studies utilizing the bitmap index method have been conducted on multi-dimensional data associated with 2D space [5]–[10]. A bitmap is an image format, and since images are two-dimensional data, they are used as vector representations of data and the method to store them. In the bitmap index method, multi-dimensional data is defined in 2D space, and when data is queried, it is assumed that appropriate data is searched, read, processed, and presented. Especially in the Nagarkar's study [6], to achieve fast search of requested

Manuscript received January 8, 2021.

Manuscript revised April 22, 2021.

Manuscript publicized June 24, 2021.

[†]The authors are with NTT Media Intelligence Laboratories, Nippon Telegraph and Telephone Corporation, Yokosuka-shi, 239-0847 Japan.

a) E-mail: kimata@m.ieice.org

DOI: 10.1587/transinf.2021PCP0001

data from multi-dimensional data, hierarchical bitmap indexing was proposed. And in the Wu's study [8], data format and indexing by image format for fast query was proposed and its efficiency was evaluated. Such a use case is suitable for services that visualize and present geographically widespread data, such as human dynamics data. We also suppose such a use case for real-time human dynamics data. In our use case, successive human dynamics data at short time intervals are collected and stored, and then some of them are queried and extracted.

In this study as well, we assume that the data is saved in the form of images because population of concatenated regions is queried. The efficiency of using image format for such case has been demonstrated in the studies [6], [8]. The efficiency of compression efficiency by using image coding is evaluated in this study.

Generally, the format of the data stored in the database should be the normally used data structure. For sensor data, several methods based on combination of run length encoding and differential encoding have been proposed. Some business intelligence tool also uses run length coding [11]. For images, there are several parameters, e.g., bit depth, and bit depth is often expressed in one byte (eight bits). In this study as well, the data is represented as a one-pixel one-byte image. Conversion from higher dynamic range data to one-byte is called tone mapping. Various tone mapping methods have been proposed. We discuss efficiency of them to apply for human dynamics data in this study. Since this conversion derives 256 level value, it can be treated as quantization. Quantization is a coding process. In the image coding, quantization parameter is used to control data size. We study how to control data size by setting appropriate parameter for human dynamics data.

In this study, for controlling the number of codes, we utilize a scalable structure. The rough data is presented and then additional detailed information is presented [12], [13]. Conventionally, scalable coding has been utilized for the transmission of image and acoustic data. For natural images, the spatial or SNR scalability function prepares standard resolution or quality images in the base layer and adds an extension layer on top of them to provide high-resolution or high-quality images. In this study, we apply the general concept of scalable coding on images in order to give the data a controllable structure.

We have studied image-based representation for human dynamics data in our own prior work [14]. In our previous study, we converted population data that has a high dynamic range into one-byte image data with non-linear mapping and evaluated the efficiency as image data for visualization. In the present work, we introduce probability distribution for representing the spatial distribution of human dynamics in order to support additionally analysing statistics of human dynamics data, for instance, comparing current population distribution in 2D space with past distribution [4]. We also evaluate the efficiency for approximation of distribution in terms of controlling data size and the accuracy of distribution in this study.

Fig. 1 Maximum population in a mesh by prefecture

Fig. 2 Number of meshes with same population range

3. Characteristics of Human Dynamics Data

The characteristics of actual human dynamics data are as follows. Human dynamics data sensed by mobile devices are acquired at 10-minute intervals [1], [3], but in this study, as we do not consider this time constraint, we used the open data of the Japanese census for our evaluation. Specifically, we used the statistical data of the address registration information for the area mesh of 500 square meters in 2010 [15]. From the latitude and longitude information of the whole of Japan, the vertical direction (latitude) is 5150 mesh and the horizontal direction (longitude) is 4966 mesh, which constitutes a two-dimensional space of vertical 5150 and horizontal 4966. In addition, assuming that processing can be performed on a regional basis, statistical processing is performed on a prefectural basis.

The mesh population distribution for each prefecture is shown in Fig. 1, which depicts the maximum value of the population in a mesh. The prefecture ID on the horizontal axis is the prefecture number determined by the Statistics Bureau. The maximum population ranges from about 2,000 to 13,000, which varies greatly from prefecture to prefecture. Figure 2 shows histograms of the demographics from Tokyo and Miyagi prefectures. The horizontal axis is the population standardized in 255 steps, and the vertical axis is the number of meshes. Any mesh with zero population is outside this statistic. In both cases, there are many meshes with a small population, but in Tokyo, there are a significant number of meshes with a larger population.

4. Approximation and Coding of Human Dynamics Data

4.1 Probability Distribution Expression

Human dynamics statistics include data such as each person's address, age, gender. Of these, we only use location information in this study. We define population information for two-dimensional mesh position as follows. Considering the two-dimensional space $V^2 \subset \mathbb{Z}^2$ specified by the combination $\varphi = (x, y)$ of two sets of integer variables (x, y) that specify the two-dimensional mesh position, the mapped elements of the set are variables that indicate whether or not a significant value exists at each discrete position in space V^2 . In this study, the size of a mesh is determined by the grid square code [16]. In the grid square code, grids are defined from the 1st to the 6th level, and the size of mesh is different. In the space V^2 , we can construct probability distribution $P(X_\varphi)$ with the variable X_φ . Here, by properly creating a set for space V^2 , the probability distribution can be set for each population data in a desired specific region. By multiplying the probability by the total number of population, the actual population value is obtained.

4.2 Approximation of Accuracy

We developed a method for expressing the probability distribution $P(X_\varphi)$ in a two-dimensional space. In the proposed method, we approximate the accuracy of the probability by mapping the values of the probability to 256 levels of discrete values including 0, which can be expressed in one byte. In addition, under this approximation we control the amount of information and the data size.

First, the values that can be taken as probability values and the correspondence to the 256 levels of values are defined. The probability value 0 has a one-to-one correspondence with the value 0. The probability distribution of the two-dimensional mesh can be expanded by

$$P(X_\varphi) = g(q)q(x, y), \quad q = \left\{0, \frac{1}{hmax}, \dots, 1\right\} \quad (1)$$

where $hmax$ is the maximum value of the population. $q(x, y)$ is a function that returns the discrete value q according to variable X_φ , and $g(q)$ is a function that returns the probability value linearly according to q . As shown in Fig. 3 (a), q can take $(hmax + 1)$ values in units of $1/hmax$. We approximate this in the form of using $q_e(x, y)$, $q_e = \{0, 1/255, \dots, 1\}$ with $1/255$ as the unit, as

$$P_{me}(X_\varphi) = g(q)q_e(x, y), \quad q_e = \left\{0, \frac{1}{255}, \dots, 1\right\} \quad (2)$$

Here, let us consider an approximation method for maximizing the amount of information in the probability distribution $P_{me}(X_\varphi)$. This can be approximated to maximize entropy. The entropy of probability distribution $P_{me}(X_\varphi)$ can be expanded as follows.

Fig. 3 Approximated expression of population distribution

$$H = - \sum_{X,Y} P_{me}(X_\varphi) \log(P_{me}(X_\varphi)) \\ = g(q) \left(- \sum_{X,Y} q_e(x, y) \log(q_e(x, y)) \right) - \log(g(q)) \quad (3)$$

To maximize Eq. (3), $-\sum_{X,Y} q_e(x, y) \log(q_e(x, y))$ should be maximized. $-\sum_{X,Y} q_e(x, y) \log(q_e(x, y))$ has the same form as the entropy when q_e is regarded as a probability distribution. Therefore, in order to maximize the amount of information, q_e should be close to a uniform distribution.

We consider the form of the function q_e . As shown in Fig. 3 (a), assuming that the probability distribution is mortar-shaped, the width of x that has the same value is large in the part where the value of the function q is small. In that part, the population is small in the actual population distribution, and there are many meshes, as shown in Fig. 2. In order to make the function q_e closer to a uniform distribution, we utilize the relationship of the mesh to the population distribution. As shown in Fig. 3 (b), the value of function q_e non-linearly increases more finely in the part where the population distribution is small, which results in a greater number of meshes.

As non-linear processing, we apply tone mapping in image processing to natural images. Here, we consider two types of tone mapping. One is for the purpose of suppressing overexposure in captured images, which we call white suppression (WS) method in this study. By using WS tone mapping, we aim to more finely approximate the part where the value of function q is small while maintaining the approximation accuracy of the part where the value of function q is large. The other tone mapping is for improving the brightness of dark parts in the image, which we call black enhancement (BE) method in this study. By using BE tone mapping, we aim to more accurately approximate only the areas where the value of function q is small.

First, we describe the WS method. So-called overexposure occurs when the signal strength is too high in an image taken by a camera. This is caused by a process for capturing a fixed-bit image signal in a suppressed form. There are various approaches for this type of tone mapping, but in this work, we utilize the equation proposed by Reinhard et al. [17],

$$L_s = \frac{L_w(1 + L_w/L_{white}^2)}{1 + L_w} \quad (4)$$

which transforms the real-space luminance L_w into the scale

Fig. 4 Characteristics of WS non-linear mapping

luminance L_s . L_{white} is the minimum value of white and can take infinity. In this work, in contrast to captured images, infinite values are not necessarily expressed, so the maximum value is set. By setting the maximum value, we can define the inverse mapping process analytically. Equation (5) is used for mapping processing:

$$t(q; \alpha) = \frac{q(1 + \alpha q)}{\alpha + q} \quad (5)$$

where function $t(q)$ maps the input value q of 0 to 1 to the output value $t(q)$ of 0 to 1. Using the function $t(q)$, function q_e is converted into:

$$q_e(x, y) = t(q)q(x, y) \quad (6)$$

and function $g(q)$ can be modified as in Eq. (7). Function $t^{-1}(q_e)$ is a function of inverse tone mapping processing and follows Eq. (8).

$$g(q)t^{-1}(q_e) \quad (7)$$

$$t^{-1}(q_e; \alpha) = \frac{1}{2\alpha} \left(q_e - 1 + \sqrt{(q_e - 1)^2 + 4\alpha^2 q_e} \right) \quad (8)$$

Here, parameter α has the same value in a series of tone mapping and inverse tone mapping. That is, α is the only parameter required for transformation and inverse transformation. Figure 4 shows an example of the WS tone curve. We assume that parameter α has a value from $\min \alpha$ to 1, where $\min \alpha$ is a certain positive minimum value. The parameter $\alpha = 1$ corresponds to a linear mapping.

On the other hand, in the BE method, a tone mapping function is defined by applying gamma correction to an image signal. Equation (9) is used for the tone mapping processing.

$$s(q; \beta) = q^\beta \quad (9)$$

Equation (10) is used for the inverse tone mapping process.

$$s^{-1}(q_e; \beta) = q_e^{1/\beta} \quad (10)$$

Then, in Eq. (6), function $s(q)$ is used instead of function $t(q)$, and in Eq. (7), function $s^{-1}(q_e)$ is used instead of function $t^{-1}(q_e)$. Note that β is a parameter in Eq. (9) and (10). It is also assumed that parameter β has a value from $\min \beta$ to 1, where $\min \beta$ is a certain positive minimum value. The parameter $\beta = 1$ corresponds to a linear mapping. Figure 5 shows an example of the BE tone curve. The difference between the characteristics of the WS method and the BE method is shown in Fig. 6, which depicts tone curves when

Fig. 5 Characteristics of BE non-linear mapping

Fig. 6 Comparison of WS and BE methods

the definite integral value of $t^{-1}(q_e)$ and $s^{-1}(q_e)$ from $q_e = 0$ to $q_e = 1$ is the same, where the definite integral is 0.23, and parameters are $\alpha = 0.015$ and $\beta = 0.3$. We can see that the WS method has a more gentle tone curve in the part of q_e that is large.

Next, we present a method for determining the minimum value of parameter α or β . In order to aim for the form of a function in which the value of function q_e increases non-linearly and finely in the part where the population distribution is small and many meshes exist, the population (excluding zero) is divided into 255 stages, the distribution of the reciprocals of the number of meshes is calculated, and the parameters are estimated by fitting the function of $t^{-1}(q_e)$ or $s^{-1}(q_e)$ to the distribution. In our estimation, we use the Levenberg-Marquardt algorithm [18], [19] to find the parameters that minimize the least squares of the difference for the distribution. Parameter α or β can be determined for each region (i.e., prefecture).

The value $q_e(x, y)$ obtained after the above tone mapping process is multiplied by 255 and converted into an integer $val(x, y)$. All $val(x, y)$ are compressed as an image. At this time, by controlling parameter α or β , the degree of approximation and the entropy of the spatial distribution represented by $q_e(x, y)$ are controlled. The data size of $val(x, y)$ to be encoded is also controlled.

4.3 Coding Flow of Spatial Probability Distribution

The flow of the distribution coding process is shown in Fig. 7. First, linear scale conversion is performed so that the maximum value of the probability value is one. Then, non-linear tone mapping is performed, where parameter α or β is set for each region. The obtained $val(x, y)$ is encoded as a two-dimensional image. When setting parameter α or β for each region, $region(x, y)$ is registered to specify the region information for each mesh as supplemental data. The region information is also encoded as a two-dimensional image.

We use lossless coding for compression. As addressed in our previous study [14], the autocorrelation coefficient is smaller than that of a natural image. For this reason, we propose using PNG (Portable Network Graphics), which is usually used for compressing two dimensional graphics.

4.4 Accuracy Scalable Coding of Human Dynamics Data

$q_e(x, y)$ represents an approximation of the probability distribution. The approximation accuracy may not always be high, so we propose a mechanism to further improve the approximation accuracy by adding one byte of data to $q_e(x, y)$. We call this accuracy scalable coding. It has a layer structure similar to SNR scalable coding for natural images and features a two-layer structure consisting of a base layer and an extension layer as shown in Fig. 8. In the case of encoding an image signal, addition and subtraction are performed between layers to obtain a signed value, and that value is then encoded. In this study, since the dynamic range of the original data is large (see Fig. 1), the difference by addition and subtraction is also large. Expressing such a large value with a sign increases the amount of code, making it difficult to express it in one byte. Therefore, we use the difference $d(x, y)$ of multiplication and division instead of addition and subtraction, as shown in Eq. (11). The difference value is converted to $vald(x, y)$ using the decimal value of the top three significant figures and is expressed in one byte. Equation (11) is calculated in mesh units. The probability distribution $P_{md}(X_\varphi)$ when adding an extension layer is expressed by Eq. (12). The operator $*$ is the Hadamard product.

$$d(x, y) = \frac{P(X_\varphi)}{P_{me}(X_\varphi)} \quad (11)$$

$$P_{md}(X_\varphi) = d(x, y) * P_{me}(X_\varphi) \quad (12)$$

When accuracy scalable coding is used, $vald(x, y)$ indicating the difference is additionally encoded as an image for an enhancement layer.

Fig. 7 Coding Flow of spatial probability distributions

Fig. 8 Layered structure of accuracy scalable coding

5. Spatial Distribution with Different Mesh Sizes

5.1 Mesh Size Relationship

The size of the two-dimensional mesh is arbitrary. We show the relationship of the spatial distribution between different mesh sizes and examine the application to compression coding. The grid square code defines mesh sizes. For instance, a 3rd level mesh has a size of about 1000 meters and a 4th level mesh has a size of about 500 meters.

Here, the set of mesh positions based on the 3rd level mesh is V_1^2 , and the probability distribution $P(X_{\varphi_1})$ is constructed, where variable X_{φ_1} is defined for $\varphi_1 = (x_1, y_1) \in V_1^2$. Then, we consider a two-dimensional space specified by a set of two-dimensional real values $V_{R1}^2 \subset \mathbb{R}^2$, define a function f_{R1} on this space, and set $f_{R1}(x_1, y_1) = P(X_{\varphi_1})$. Similarly, we set the mesh positions based on the 4th level mesh to V_2^2 , and define the variable X_{φ_2} corresponding to $\varphi_2 = (x_2, y_2) \in V_2^2$ and the probability distribution $P(X_{\varphi_2})$. And a two-dimensional space specified by a set of two-dimensional real values is set to $V_{R2}^2 \subset \mathbb{R}^2$, a function f_{R2} is defined, and $f_{R2}(x_2, y_2) = P(X_{\varphi_2})$ is set. Figure 9 shows the relationship between functions f_{R1} and f_{R2} in consideration of the positional relationship of the regional mesh at this time. At the blue circle, $f_{R1}(x_1, y_1)$ has a value of $P(X_{\varphi_1})$, and at the red circle, $f_{R2}(x_2, y_2)$ has a value of $P(X_{\varphi_2})$.

5.2 Approximation of Distribution of Lower Level Mesh from Upper Level Mesh

We next examine the approximation of the distribution in terms of the precision of spatial location. The approximated distribution is generated from upper level meshes. For example, the approximated distribution of the 4th level mesh is generated from the distribution of the 3rd level mesh. We consider a method of interpolating $f_{R1}(x_1, y_1)$ to obtain an approximate value of $f_{R2}(x_2, y_2)$. An interpolation method to enlarge an image is applied. When enlarging the image, the image signal of the target pixel is interpolated from the peripheral pixels. Typical interpolation processing includes nearest neighbor interpolation, bilinear interpolation, and bicubic convolution interpolation [20]. In this study we use bicubic convolution interpolation and calculate the value of position (x, y) in the (x_1, y_1) coordinate system with the

Fig. 9 Relationship of 3rd and 4th level mesh

following kernel function for horizontal direction, where c_x is a parameter.

$$h(x; c_x) = \begin{cases} (c_x + 2)|x|^3 - (c_x + 3)|x|^2 + 1 & (0 \leq |x| < 1) \\ c_x|x|^3 - 5c_x|x|^2 + 8c_x|x| - 4c_x & (1 \leq |x| < 2) \\ 0 & (2 \leq |x|) \end{cases} \quad (13)$$

The kernel function for vertical direction, $h(y; c_y)$, is similarly calculated with parameter c_y . These parameters are estimated by minimizing the difference from the actual distribution using the Levenberg-Marquardt algorithm [18], [19].

5.3 Spatial Scalable Coding of Upper and Lower Level Meshes

We propose a compression coding that combines the distribution of the upper and lower level meshes, which we call spatial scalable coding. We consider a configuration in which $f_{R1}(x_1, y_1)$ of the upper level mesh is encoded first, and then a part of $f_{R2}(x_2, y_2)$ of the lower level mesh is encoded.

When coding a natural image, the amount of bits is reduced by encoding two different spatial resolution images together, instead of encoding the images independently. There are two main techniques for spatially scalable coding of natural images. One is a configuration in which an image signal is filtered by a spatial frequency component and divided by a frequency domain, as in the still image coding method JPEG2000 [21]. The signal corresponding to the low frequency component is treated as a low resolution image. The other is to prepare a low resolution image and a high resolution one, like the video coding method H.264 SVC [22], and when encoding the high resolution image, the image signal obtained by enlarging the low resolution one is referenced. However, in this study, we prepare the values of the upper level mesh and the lower level mesh, and examine the configuration in which a part of the values of the lower level mesh is calculated by using the values of the upper level mesh.

Since probability values are set for functions $f_{R1}(x_1, y_1)$ and $f_{R2}(x_2, y_2)$ obtained from the distribution of the 3rd level mesh and the 4th level mesh, there is

Fig. 10 Data arrangement of spatial scalable coding

the relationship between them:

$$4f_{R1}(x_1, y_1) = \sum_{i=0}^1 \sum_{j=0}^1 f_{R2}(x_2 + i, y_2 + j) \quad (14)$$

Therefore, after value $f_{R1}(x_1, y_1)$ of the 3rd level mesh is obtained, we can calculate one of the four values $f_{R2}(x_2 + i, y_2 + j)$, $i, j \in \{0, 1\}$ of the corresponding 4th level meshes. For example, in Fig. 10 (a), the value of the white circle position in the 4th level mesh can be calculated from the corresponding black mesh in the 3rd level and blue, red, green meshes in the 4th level. To encode these values, we first encode the values of the 3rd level meshes and next encode the values of subset1, subset2, and subset3 of the 4th level meshes sequentially. To control the data size during the encoding, sequential encoding of subsets in the 4th level meshed is terminated. This is similar to the technique that recursively divides the lower frequency in JPEG2000.

6. Experimental Evaluation and Results

6.1 Evaluation of Non-Linear Mapping of Distribution

First, we evaluate the performance of the method when nonlinearly mapping the population distribution to 1-byte discrete values. KL divergence (Kullback-Leibler divergence) is used to measure the similarity because KL divergence is commonly used for evaluation of the probability distribution. $P_{eval}(X_\varphi)$ is the probability distribution to be evaluated.

$$KL = - \sum_{X_\varphi} P(X_\varphi) \log \left(\frac{P(X_\varphi)}{P_{eval}(X_\varphi)} \right) \quad (15)$$

We begin by evaluating how well the method expresses the spatial distribution of the population. First, the results of the parameters used for nonlinear mapping are discussed. Figure 11 shows an example of the result of fitting a 4th level mesh to the reciprocal of the number of meshes using Eq. (8) of the WS method. We can see that it was fitted close to the part with a small population. Figure 12 (a) shows the obtained values of parameter α . These values ranged from 0.005 to 0.08, and highly non-linear mapping parameters were obtained. Figure 12 (b) shows the obtained values of parameter β as results of fitting by the BS method. These values ranged from 0.1 to 0.6, and mapping parameters with high non-linearity were also obtained.

Next, we evaluate the approximating of the probability distribution of the population distribution of each prefecture.

Fig. 11 Non-linear mapping to reciprocal of number of meshes

Fig. 12 Estimated parameter of non-linear mapping

Fig. 13 KL divergence of space probability distribution

Fig. 14 Entropy of space probability distribution

Figure 13 shows the KL divergence when the values of parameters α and β were obtained by fitting. We can clearly see that the non-linear WS and BE methods had a smaller KL divergence than the linear method. Depending on the prefecture, the KL divergence was slightly different between the WS and BE methods.

Figure 14 shows the entropy of the population distribution by function q_e in the WS method. The non-linear mapping had higher entropy than the linear mapping. This shows that the non-linear mapping makes information more uniformly assigned to each 255 levels.

The same tendency was obtained in the result of non-linear mapping to the 3rd level mesh.

6.2 Evaluation of Encoding Control and Compression Method

When parameter α of the WS method is changed from the minimum value to 1, it changes continuously from non-linear to linear. Utilizing this, Fig. 15 shows the degree of deterioration of the approximation and the tendency of the change in the code amount when parameter α was changed in the 4th level mesh. The value obtained for each prefecture was set for parameter α , and the value calculated according to Eq. (16) in 11 steps was used.

$$\alpha = \frac{1 - \min \alpha}{10} * i + \min \alpha, \quad i \in \{0, \dots, 10\} \quad (16)$$

Fig. 15 Relationship of data size and KL divergence and mean square error

We utilized PNG for the encoding. Since the population value of the 4th level mesh we used was within 16 bits at the maximum, the code amount when a grayscale 16-bit image was encoded with PNG was 745,816 bytes.

These results demonstrate that the code amount can be controlled by controlling parameter α . We can also see that the degree of deterioration of the approximation increased as the amount of code decreased.

We evaluated efficiency of using PNG. In Fig. 15, we show the results when GZIP compression method, which is based on run length coding, was applied. We can see that PNG outperforms GZIP in terms of compressed data size. This is because the population data has correlation in spatial direction and spatial prediction of PNG effectively utilizes the correlation.

We also evaluated similarity when actual population number is decoded. We show the results on mean square error in Fig. 15 (b). We also can see the similar tendency to the results on KL-divergence. From this result, according to the proposed control scheme, approximation of either the probability distribution or decoded values can be controlled.

Figure 16 shows the decoded picture when α is equal to $\min \alpha$ or 1. Intensity on green color means the value of $q_e \times 255$. We can see high intensity in the area whose population is high shown in Fig. 16 (a). Comparing Fig. 16 (b) and Fig. 16 (c), we can see more pixels of high intensity in the whole picture when α is equal to $\min \alpha$. This is because the number of high intensity pixels is increased by non-linear mapping.

6.3 Evaluation of Accuracy Scalable Coding

Next, for the accuracy scalable coding, we evaluate the performance when the probability distribution was updated in the extension layer. Figures 17 and 18 respectively show the frequency of differences between the WS and BE method extension layers in the 3rd level mesh. In both methods, the difference from 0 to 2 times was dominant. There were no differences over 2.5 times in the WS method, but some were seen in the BE method. This is because the tone curve function in the WS method is designed to suppress cases where the value is large and the width of quantization is narrower than that in the BE method.

The KL divergence when the enhancement layer was

(a) Whole picture ($\alpha = \min \alpha$) (b) Tokyo area ($\alpha = \min \alpha$) (c) Tokyo area ($\alpha = 1$)**Fig. 16** Decoded pictures **Fig. 17** Numbers of residual value in enhancement layer for WS method **Fig. 18** Numbers of residual value in enhancement layer for BE method

added had a value less than 0.00001. We found that a distribution close to the actual probability distribution could be reproduced by adding an enhancement layer. In the nonlinear mapping using the WS method, the code amount when

Fig. 19 Approximation from upper level mesh **Fig. 20** KL divergence of spatial scalable coding

$val(x, y)$ was compressed by PNG when parameter α was set to the minimum value was 324,603 bytes. Since the amount of the base $val(x, y)$ was 527,340 bytes, the total of the base layer and the enhancement layer was 851,943 bytes. The amount of code was about 100 bytes more than the amount of code when a grayscale 16-bit image was encoded with PNG. This can be considered as a layer division loss in the scalable coding.

6.4 Approximation from Upper Level Mesh

Next, we evaluate the interpolation process that interpolates the distribution of the 3rd level mesh to obtain an approximation of the distribution of the 4th level mesh. Figure 19(a) shows the parameters of the interpolation function of $f_{R1}(x_1, y_1)$ obtained for each prefecture. We can see many values that ranged from -0.5 to -2.0 . Since a negative value is often used in natural images, we consider that similar values were obtained here. Figure 19(b) shows the values of KL divergence. A slight decrease in KL divergence was confirmed by bicubic convolution interpolation.

6.5 Evaluation of Spatially Scalable Coding

Figure 20 shows the KL divergence when a part of the 4th level mesh was calculated and obtained using the 3rd level mesh. For the 4th level mesh, we utilized the results of non-linear mapping using the WS method and compared it with the case where the value in the 4th level mesh was directly coded. Degradation from direct coding was less than 0.002 bits. The total code amount for three subsets in the 4th level mesh was $139845 + 103740 + 138591 = 382176$ bytes. Since the code amount of the upper mesh was about 174000 bytes, this corresponds to about 2.2 times the code amount.

6.6 Discussion on Data Storage

Here, we explain how to determine the coding parameters. When the target data size is set, it is first necessary to determine the mesh level. For example, the code amount of the 3rd level mesh is on the order of less than about 170 kilobytes, and the code amount of the 4th level mesh is on the order of about 430 to 500 kilobytes. Therefore, if you want to make it smaller than 430 kilobytes, it is necessary to select a 3rd level mesh. Next, the non-linear parameters can be adjusted within the level so that the approximation of the probability distribution is most accurate within the range that fits in the target data size. Then, for the additional data prepared for further improvement of approximation, encoding can be performed using two methods, accuracy scalable coding and spatial scalable coding. To improve the accuracy, a code amount similar to that of the code amount at the same level is further required. For generating a distribution from a 3rd level mesh to a 4th level mesh, up to about three times the code amount of the 3rd level mesh can be applied.

The proposed method has some limitation on controlling the data size. There is a gap of data size between 3rd level mesh and 4th level mesh. As described above, the data size of the 3rd level mesh is about 170 kilobytes, and the minimum size of the 4th level mesh is about 430 kilobytes. There is a gap from 170 to 430 kilobytes. Therefore, for instance, if we want to set the data size to less than about 300 kilobytes, the 3rd level mesh should be selected.

7. Conclusion

In this study, we proposed a data representation and encoding method to fit the target data size when registering human dynamics data in the database. We examined the data structure for expressing data in the form of images, and proposed a method for expressing the spatial distribution in one byte. We proposed two methods for approximating the distribution. The first method features a non-linear mapping to data with a large dynamic range. Experiments showed that non-linear mapping made it possible to construct data with a large amount of information, and it could be approximated in a form close to the real distribution from the viewpoint of the amount of information. The second is a spatial scalability method. It has a mechanism to express subsets of the lower level mesh. We also proposed a mechanism to control the approximation and code amount by setting specific parameters.

In addition, we proposed accuracy scalable coding to improve the accuracy. By adding enhancement layer of the same data size to the base layer, the proposed method enables accuracy improvement, which was confirmed through experiments.

As stated above, in this study, we proposed a method for expressing human dynamics data with a probability distribution and demonstrated its effectiveness. Since human dynamics data are expected to be collected and used in real

time in the future, this method, which can express a small amount of data, should be effective for real-time distribution. Attribute information such as age information in the corresponding mesh is often added to human dynamics data. As such, representation and coding of attribute information is a topic for future study. In addition, the applicability to other than human dynamics data is also a topic for future research.

References

- [1] M. Terada, T. Nagata, and M. Kobayashi, "Population Estimation Technology for Mobile Spatial Statistics," NTT DOCOMO Technical Journal, vol.14, no.3, pp.10–15, Jan. 2013.
- [2] W. Loibl and J. Peters-Anders, "Mobile phone data as source to discover spatial activity and motion patterns," GIForum 2012: Geovisualization, Society and Learning, Publisher: Wichmann Verlag, Jan. 2012.
- [3] M. Terada, H. Akatsuka, T. Nagata, and S. Nakanishi, "Avoiding Tokyo Bay Aqua Line Congestion Using Traffic Congestion Forecasting AI," NTT DOCOMO Technical Journal, vol.21, no.2, pp.27–35, Oct. 2019.
- [4] L. Pappalardo and F. Simini, "Data-driven generation of spatio-temporal routines in human mobility," Data Mining and Knowledge Discovery, vol.32, no.3, pp.787–829, May 2018.
- [5] T. Zäschke, C. Zimmerli, and M.C. Norrie, "The PH-Tree - A Space-Efficient Storage Structure and Multi-Dimensional Index," Proc. 2014 ACM SIGMOD International Conference on Management of Data, pp.397–408, June 2014.
- [6] P. Nagarkar, K.S. Candan, and A. Bhat, "Compressed spatial hierarchical bitmap (cSHB) indexes for efficiently processing spatial range query workloads," Proc. VLDB Endowment, vol.8, no.12, pp.1382–1393, Aug. 2015.
- [7] Y. Tong, J. She, B. Ding, L. Chen, T. Wo, and K. Xu, "Online minimum matching in real-time spatial data: experiments and analysis," Proc. VLDB Endowment, vol.9, no.12, pp.1053–1064, Aug. 2016.
- [8] X. Wu, M. Kitahara, and A. Shimizu, "Fast Query Method for Visualization of Large Spatial Data by Using Image-based Structure," Proc. 15th Forum on Information Technology, no.3, pp.195–198, Sept. 2016 (in Japanese).
- [9] F. Li, L. Chen, Y. Zeng, A. Kumar, X. Wu, J.F. Naughton, and J.M. Patel, "Tuple-oriented Compression for Large-scale Mini-batch Stochastic Gradient Descent," Proc. 2019 ACM SIGMOD International Conference on Management of Data, pp.1517–1534, June 2019.
- [10] K. Park, "A hierarchical binary quadtree index for spatial queries," Wireless Networks, vol.25, pp.1913–1929, Jan. 2019.
- [11] R. Wesley and P. Terlecki, "Leveraging compression in the tableau data engine," Proc. 2014 ACM SIGMOD International Conference on Management of Data, pp.563–573, June 2014.
- [12] M.T.H. Khan and S. Han, "An Interface between 3D Modeling Tool and Scientific Data Format," International Journal of Machine Learning and Computing, vol.2, no.2, pp.168–172, April 2012.
- [13] G. Kaur and N. Aggarwal, "Exploiting Hierarchical Structure of XML Data Using Association Rule Analysis," International Journal of Machine Learning and Computing, vol.2, no.3, pp.274–277, June 2012.
- [14] H. Kimata, X. Wu, and R. Tanida, "Hierarchical and Compact Bitmap Based Data Structure of Human Dynamics Data for Visualization," Proc. 2020 The 4th International Conference on Graphics and Signal Processing, pp.98–102, June 2020.
- [15] 2010 Population Census, Statistics Bureau of Japan, <http://www.stat.go.jp/english/data/kokusei/index.html>
- [16] A. Sato, S. Nishimura, and H. Tsubaki, "World Grid Square Codes: Definition and an example of world grid square data," Proc. 2017 IEEE International Conference on Big Data (Big Data),

- pp.4238–4247, Dec. 2017.
- [17] E. Reinhard, M. Stark, P. Shirley, and J. Ferwerda, “Photographic tone reproduction for digital images,” *ACM Transactions on Graphics*, vol.21, no.3, pp.267–276, July 2002.
 - [18] K. Levenberg, “A Method for the Solution of Certain Non-linear Problems in Least Squares,” *Quarterly of Applied Mathematics*, vol.2, no.2, pp.164–168, July 1944.
 - [19] D.W. Marquardt, “An Algorithm for Least-Squares Estimation of Nonlinear Parameters,” *Journal of the Society for Industrial and Applied Mathematics*, vol.11, no.2, pp.431–441, June 1963.
 - [20] R. Keys, “Cubic convolution interpolation for digital image processing,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol.29, no.6, pp.1153–1160, Dec. 1981.
 - [21] A. Skodras, C. Christopoulos, and T. Ebrahimi, “The JPEG 2000 Still Image Compression Standards and Beyond,” *IEEE Signal Process. Mag.*, vol.18, no.5, pp.36–58, Sept. 2001.
 - [22] H. Schwarz, D. Marpe, and T. Wiegand, “Overview of the Scalable Video Coding Extension of the H.264/AVC Standard,” *IEEE Trans. Circuits Syst. Video Technol.*, vol.17, no.9, pp.1103–1120, Sept. 2007.

Hideaki Kimata received the B.E. and M.E. degrees in applied physics, and the Ph.D. degree in electrical engineering respectively from Nagoya University, Nagoya, Japan, in 1993, 1995, and 2006. He joined Nippon Telegraph and Telephone Corporation (NTT) in 1995, and has been involved in the research and development of video coding, realistic communication, computer vision, and video recognition based on machine learning (deep learning). He is currently a Senior Research Engineer and also the

Supervisor at NTT Media Intelligence Laboratories. He is a Chair of Technical Committee on Image Engineering of the Institute of Electronics, Information and Communication Engineers of Japan.

Xiaojun Wu received the Ph.D in Informatics from Kyoto University, 2005. Working in NTT Media Intelligence Labs from 2005. Member of IEICE. Research interest includes dynamic data modeling of human activities and visualizations.

Ryuichi Tanida received the B.E. and M.E. degrees in electrical engineering from Kyoto University in 2001 and 2003, respectively. Since joining NTT in 2003, He has been engaged in research on video coding and its application technology. He is currently a Senior Research Engineer at NTT Media Intelligence Laboratories. He is a member of IEICE.