ImageCube: A Browser for Image Collections Associated with Multi-Dimensional Datasets

Yunzhu Zheng
Ochanomizu University
Tokyo, Japan
Email: yunzhu@itolab.is.ocha.ac.jp

Ai Gomi
Ochanomizu University
Tokyo, Japan
Email: gomiai@itolab.is.ocha.ac.jp

Takayuki Itoh
Ochanomizu University
Tokyo, Japan
Email: itot@is.ocha.ac.jp

***Abstract*—Image browsing techniques thus become increasingly important for overview and retrieval of particular images in large-scale collections. At the same time, there are various sets of images which are associated with multi-dimensional or multivariate datasets. We believe that image browsing for such datasets should be inspired from multi-dimensional data visualization techniques. This paper presents ImageCube, a scatterplot-like browser for image collections associated with multi-dimensional datasets. ImageCube locates a set of images into a display space assigning a pair of dimensions to X- and Y-axes. It suggests preferable pairs of dimensions by applying Kendall's rank correlation and Entropy on the display space, so that users can easily obtain interesting visualization results. This paper presents a case scenario that a user finds a preferable car from an image collection by using ImageCube.**

***Keywords*-Visualization, Image browser, Multidimensional data.**

I. Introduction

With the rapid development of the imaging technologies over the recent years, advanced visualization techniques for thousands of pictures are making big progress. At the same time, now we can obtain various sets of images which are associated with multi-dimensional or multivariate datasets via Internet. For example, we can obtain the images of recipes which have a variety of nutritional value, those of cars which have a variety of performance values, and those of medical which have a variety of diagnosis value, based on our specific requests. We think that image browsing techniques featuring multidimensional data visualization techniques are useful to explore such kinds of image datasets. Especially, multidimensional data visualization techniques are useful to explore and analyze features and structures in the datasets, including clusters, outliers or correlations. We believe it is interesting to explore and analyze such features and structures of multidimensional values assigned to images while browsing the images themselves.

Among various information visualization techniques, scatterplots is one of the most popular and widely-used visual representations for multidimensional data due to its simplicity, and visual clarity. Scatterplots visualize multidimensional datasets by assigning data dimensions to graphical axes and rendering data cases as points in the Cartesian space defined by the axes. This approach has been widely used in visualization technique such as Rolling the Dice [2], which presents new interactive methods to explore multidimensional data. It applies a metaphor of rolling a dice, since as they implement an animation mechanism as they transform one scatterplot representation into another one by rotating a cubic space.

This paper proposes "ImageCube", an image browser featuring a multidimensional data visualization technique which is similar to Rolling the Dice. ImageCube can show the visualization through interactively selecting two dimensions from the multidimensional datasets of images applying scatterplot. Our implementation of ImageCube assists the dimension selection operations by suggesting interesting pairs of dimensions based on their correlations and entropies. ImageCube is helpful for users to obtain qualitative visualization results to explore and analyze features and structures of multidimensional values assigned to images.

We tested ImageCube with images of recipes and those of cars. Visualization results in the paper demonstrate that ImageCube efficiently supports users to obtain insightful visualizations.

II. Related Work

There have been various image browsers related to ImageCube. Also, there have been various techniques for multidimensional data visualization related to ImageCube. This section introduces several related techniques.

A. Image Browser

Overview and retrieval of image collections are important issues for their owners. Image browsing is therefore an active research topic and therefore recently many novel image browsers have been presented. This section categorizes the image browsers into two groups: browsers for structured/unstructured sets of images.

Many of image browsers for structured sets of images suppose that images forms trees or graphs. PhotoMesa [1] is one of the most famous image browsers for structured sets of images. It divides a window space by nested set of rectangles to represent the hierarchy of images, and then packs them in each of the rectangular subspaces.

On the other hand, many of other image browsers for unstructured sets of images scatter the images onto 2D/3D spaces like scatterplots. Several techniques applies dimension reduction techniques such as MDS (Multi Dimensional Scaling) or PCA (Principal Component Analysis) to locate the images [8] so that similarly looking images are placed closer on the display spaces. Several others directly assigns two or three values associated to the images to the axes of 2D/3D spaces [4].

MIAOW [3] is a hybrid technique that forms hierarchy while it assigns three values to axes of a 3D space. It divides the images according to latitudes, longitudes, and times which the images are taken, while assigns the three values to axes of a 3D space. The mechanism is somewhat similar to ImageCube; however, MIAOW just treats latitude, longitude, and time as three dimensional values. ImageCube presented in this paper is a more generalized image browser.

B. Multi-Dimensional Data Visualization

There have been various multi-dimensional data visualization techniques, including Parallel Coordinates, VisDB, and Worlds within Worlds. Some other techniques apply heatmaps or glyphs to represent multi-dimensional value. Meanwhile, Scatterplots is one of the most popular techniques to visualize multi-dimensional data. Many scatterplots implementations directly assigns two or three of the dimensions to axes of the visualization spaces, while others apply dimension reduction techniques. Scatterplot matrix is often used for overview of scatterplots selecting arbtrary pairs of dimensions; however, it requires very large display spaces if number of dimensions is large. If users do not want to use such large display spaces for scatterplots, they may need to interactively switch the pairs of dimensions to understand correlations between the dimensions. Rolling the Dices [2] is one of the novel techniques to assist the interactive selection of dimensions for scatterplots.

Dimension analysis is helpful to obtain fruitful knowledge from multi-dimensional data visualizations. Sips et al. [7] presented a view selection technique of multi-dimensional data visualization by applying the dimension analysis. Nagasaki et al. [5] presented a correlation-based dimension selection technique for scatterplots-based visualization of credit card fraud data. The correlation-based strategy is also useful to reorder the dimensions and improve the readability of Parallel Coordinates and scatter plots matrices [6].

III. Browsing Image Collections with Multi-Dimensional Values

This section presents our image browser ImageCube, which represents image collections by scattering the images. Specified arbitrary two dimensions, ImageCube calculates positions of images in a display space by assigning the two dimensions to axes of the display space. It then groups the images based on their positions, and selects a representative image for each group. ImageCube initially displays the representative images, and provides a user interface to click the images so that other images in the same group with the clicked images are displayed in another window space. Dimension selection may be a problem for usability. ImageCube supports a function to recommend interesting pairs of dimensions so that users can easily select them.

A. Definition of Input Images

We suppose an image set $I = \{i_1, i_2, ..., i_n\}$ as input information, where i_i is the i-th image, and n is the number of images. We also suppose that an image $i_i = \{v_{i1}, v_{i2}, ..., v_{im} n_i, u_i\}$, where v_{ij} is the j-th value of the i-th image, m is the number of dimensions, a_i is the name of the i_i, and u_i is the URL or path of the i_i. Our implementation consumes input data files which describe names of the dimensions, multidimensional values, name of the images, and URL or path of the images. It then automatically generates and displays the selection menu from the input information, which are used to select arbitrary two dimensions as X- and Y-axes. Also, it calculates c_{ij}, the coordinate value of the j-th dimension of the i-th image, which is used as positions of images.

B. Multidimensional Visualization

ImageCube represents two dimensions in a single visualization as many scatterplots techniques do. When a user specifies the p-th dimension as the X-axis, and the q-th dimension as the Y-axis, ImageCube places the i-th image at (c_{ip}, c_{iq}). ImageCube senses click operations of users to display detailed information of the particular images on demand.

ImageCube supports a smooth dimension selection, by rotating the display space, like Rolling the Dice [2] supports. Suppose that the p-th dimension is assigned to the X-axis, and the q-th dimension to the Y-axis. When a user specifies the r-th dimension as the X-axis, ImageCube temporarily assigns the r-th dimension to the Z-axis, and therefore ImageCube places the i-th image at (c_{ip}, c_{iq}, c_{ir}). ImageCube then rotates the display space along the Y-axis, so that the XZ-plane gets the XY-plane, and the XY-plane gets the XZ-plane. Figure 1(Left) shows a capture of the rotation process.

C. Recommendation of Dimension Pairs

ImageCube automatically generates the selection menu featuring buttons of dimensions for X- and Y-axes. Here, a major challenge is how to easily get fruitful visualization results from multidimensional datasets according to user's needs. Therefore ImageCube provides a mechanism to recommend interesting pairs of dimensions so that users can easily select them. Current our implementation shows the

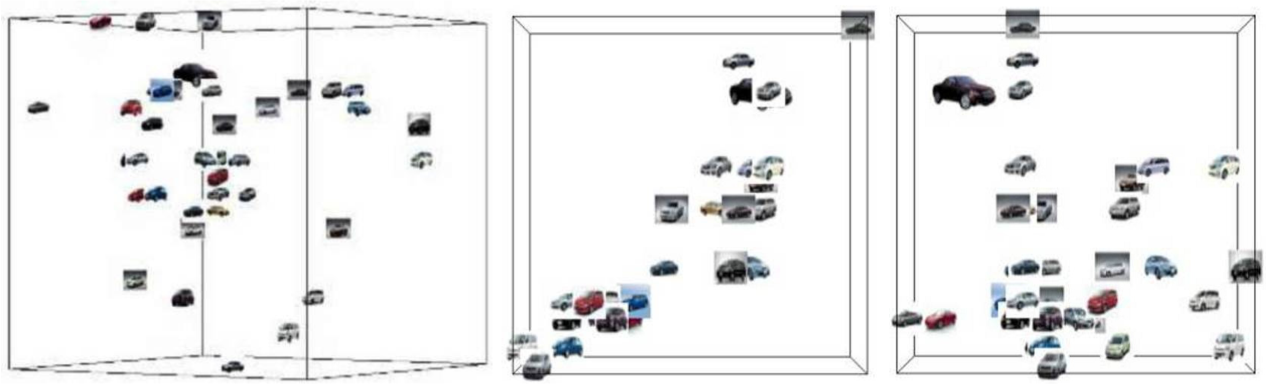

Figure 1. (Left) Image redeployment by rotation function of XY- and XZ-planes. (Center) A visualization result with a high-correlation pair of dimensions. (Right) A visualization result with a high-entropy pair of dimensions.

recommended pairs by coloring corresponding dimensions on the selection menu.

To realize the mechanism, we need to analyze the numerical features between arbitrary two dimensions. Current our implementation calculates the following two kinds of numerical features between arbitrary pairs of dimensions to obtain interesting visualization results:

- Kendall's rank correlation to obtain results which images are regularly aligned. See Figure 1(Center).
- Entropy to obtain results which images are evenly distributed. See Figure 1(Right).

Kendall's rank correlation: Correlation analysis is useful for multidimensional datasets visualization techniques, as discussed in Section II-B We apply Kendall rank correlation for the correlation analysis. The Kendall rank (τ) denotes the similarity of the orderings of the datasets ranked by each of the quantities, defined as follows:

$$\tau = \frac{4P}{n(n-1)} - 1 \tag{1}$$

which is supposed as following.

- If the agreement between the two rankings is perfect (i.e., the two rankings are the same) the coefficient has value 1.
- If the disagreement between the two rankings is perfect (i.e., one ranking is the reverse of the other) the coefficient has value -1.
- If X and Y are independent, then we would expect the coefficient to be approximately zero.

ImageCube calculates the Kendall rank correlation for every possible pair of dimensions, and suggests the pairs which bring high correlations.

Entropy: ImageCube measures the distribution and randomness of the visualization results by applying Entropy. ImageCube internally divides the display space into N_s rectangular subspaces, and count the number of images p_i in the i-th subspace. ImageCube calculates the sum of Entropy E_{sum} in the subspaces as follows:

$$E_{sum} = \sum^{N_s} \left(\frac{p_i}{N} \log \frac{p_i}{N}\right) \tag{2}$$

ImageCube calculates the Entropy for every possible pair of dimensions, and suggests the pairs which bring high randomness.

D. Overlap Reduction

The simple image location strategy described in Section III-B easily causes overlap of images on the display space, while displaying large-scale image collections. To improve the understanding and usability, ImageCube reduces the number of displaying images by a clustering based on their positions on the display space. It generates groups of images that locate inside a constant radius of circles, and selects a representative image for each group. It initially displays only the representative images, and other images in the group of a representative image s will be displayed in another space, when a user clicks one of the representative images.

E. User Interface

Figure 2 shows our implementation of ImageCube. It features a drawing area in the left side of the window, and three tabs with user interface widgets in the right side of the window. The first tab in Figure 2 (Left) features widgets for setting viewing and drawing options.

The second tab in Figure 2 (Center) features buttons for selection of dimensions for assigning to X- and Y-axes. When a dimension is selected for the X-axis, labels of several buttons are colored in red, which denotes that higher rank correlation is obtained when the colored dimensions are selected for the Y-axis. Similarly, when a dimension is selected for the Y-axis, labels of several buttons are colored in red, because they are recommended for the X-axis. The labels may be also colored in yellow, which denotes that higher Entropy is obtained when the colored dimensions

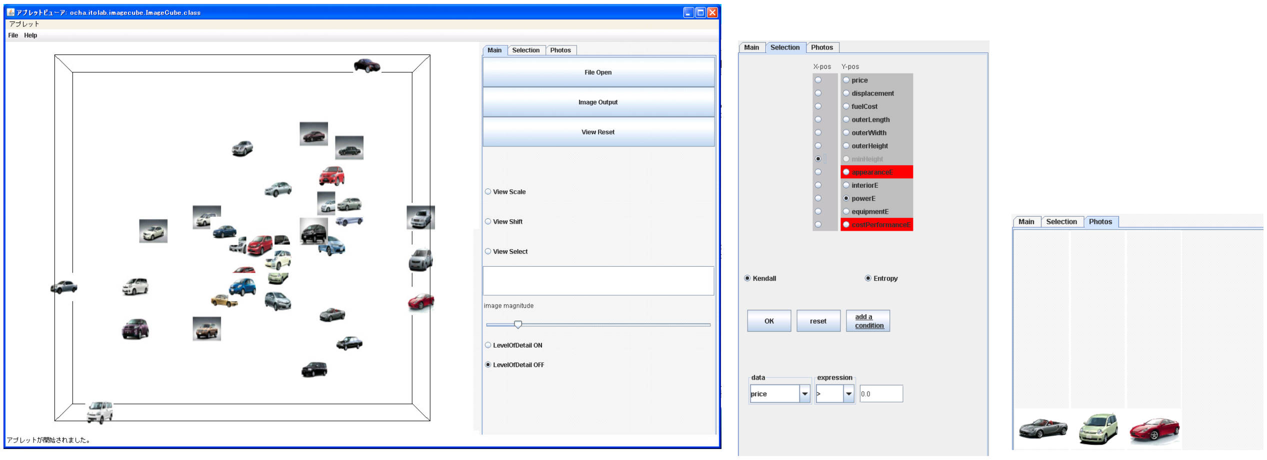

Figure 2. Our implementation of ImageCube. (Left) A drawing area, and a tab featuring widgets for setting viewing and drawing options. (Center) A tab featuring buttons for selection of dimensions for assigning to X- and Y-axes. (Right) A tab featuring a panel to display the images belonging to the clusters of clicked representative images.

are selected. This tab also features widgets for specifying the ranges of values. Images are not displayed if the values assigned to them are out of the specified ranges.

The third tab in Figure 2 (Right) features a panel to display the images belonging to the clusters of clicked representative images.

IV. Example

We implemented ImageCube on Java 1.6.0 and JOGL (Java binding for OpenGL). We tested ImageCube on Lenovo ThinkPad T510 (CPU 2.4GB, RAM 2GB) with Windows XP SP3.

This section shows results of ImageCube applying 100 images collected from Japanese automobile catalog Web site [1] . We parsed HTML files introducing particular cars, to extract specifications and evaluations of the cars, and download their images. Consequently we obtained a 12 dimensional dataset including the following specification and evaluation values: 1) price, 2) displacement of the engine, 3) fuel cost, 4) outer length, 5) outer width, 6) outer height, 7) height of floor, 8) user evaluation of appearance, 9) user evaluation of interior design, 10) user evaluation of engine power, 11) user evaluation of equipments, and 12) user evaluation of cost performance.

We also constructed an image collection from Japanese recipe Web site [2]. We parsed HTML files introducing particular recipes, and consequently we obtained a 7 dimensional dataset including the following values: 1) time required to cook, 2) energy, 3) amount of vegetable, 4) amount of salt, 5) amount of Calcium, 6) amount of fat, and 7) amount of Iron.

[1] http://autos.yahoo.co.jp/

[2] http://www.recipe.nestle.co.jp/recipe/

This section introduces a use case scenario with the image collection of cars. First of all, we were interested in correlations between prices and other variables. We checked correlation and Entropy between them, and found that displacement and fuel costs had relatively high correlations with the price. Figure 3(1) shows an example that price is assigned to the X-axis, and displacement is assigned to the Y-axis. The example denotes that they are nearly proportional; luxury sedans are relatively expensive, and station wagons are relatively low cost in this collection. Figure 3(2) shows an example that price is assigned to the X-axis, and fuel cost is assigned to the Y-axis. Equipment evaluation also had high correlations with the price, but the distribution of the images was not linear. Figure 3(3) shows an example that price is assigned to the X-axis, and equipment evaluation is assigned to the Y-axis. It denotes that equipment evaluation increases proportional to the price of low-price cars, but it becomes flat if the price is higher. On the other hand, it was our surprise that appearance evaluation was not correlated to the price. Figure 3(4) shows an example that price is assigned to the X-axis, and appearance evaluation is assigned to the Y-axis, where it seems that expensive cars do not always obtain higher evaluations of appearance.

Be derived from the above surprising result, we were interested in what impact to the evaluation of appearance. Figure 3(5)(6) show examples that outer length or outer width is assigned to the X-axis, and appearance evaluation is assigned to the Y-axis. They denote that outer length or outer width is not well correlated with the evaluation of appearance. Actually, these pairs of dimensions had relatively higher Entropies. On the other hand, Figure 3(7) shows an example that height of floor is assigned to the X-axis, and appearance evaluation is assigned to the Y-axis. The result briefly denotes that appearance evaluations are better if floors are lower, which looks a common trend both for wagons and sedans. Actually, correlation of height of floor to appearance evaluation was relatively higher than others.

Finally, we checked what impacts to the evaluation of cost performance, and found that appearance evaluation was one of them. Figure 3(8) shows an example that appearance evaluation is assigned to the X-axis, and cost performance evaluation is assigned to the Y-axis. This high correlation denotes that appearance is very important for the user evaluation of cost performance. Again, it looks a common trend both for wagons and sedans.

This example demonstrates that we can discuss the trend of multi-dimensional values associated to the collections of images while looking at the images themselves. Actually, we could discuss what impacts to the evaluation of appearance and cost performance of cars, while looking at various values as well as designs of cars. We think this analysis tool can be applied to various fields dealing with images; for example, collections of medical images associated with medical checkup values, and collections of facial and cosmetic images associated with evaluations of subjects.

V. Conclusion and Future Work

This paper presented ImageCube, a scatterplot-like browser for image collections associated with multi-dimensional datasets. The paper described technical components of ImageCube, and introduced a use case scenario with a real image collection of cars on the Web.

Our current implementation of ImageCube supports a user interface to specify ranges of values to be displayed, however, our user case did in Section 4 did not use it. Moreover, the implementation does not suppose that tags or keywords are not associated to images. We would like to enhance them so that we can flexibly retrieve or narrow down interested images. Another our interest is implementation of dimension reduction schemes, and representations of three or more dimensions. We expect these bring us more various visualization results. Finally, we would like to test ImageCube with larger datasets including more images and higher-dimensional values, and subjectively evaluate with experimental users.

References

[1] B. B. Bederson, PhotoMesa: A Zoomable Image Browser Using Quantum Treemaps and Bubblemaps, *Symposium on User Interface Software and Technology*, 71-80, 2001.

[2] N. Elmqvist, P. Dragicevic, J. Fekete, Rolling the Dice: Multidimensional Visual Exploration using Scatterplot Matrix Navigation, *IEEE transactions on Visualization and Computer Graphics*, 14(6), 1141-1148, 2008.

[3] A. Gomi, T. Itoh, MIAOW: A 3D Image Browser Applying a Location- and Time-Based Hierarchical Data Visualization Technique, *Advanced Visual Interface (AVI10)*, 225-232, 2010.

[4] H. Horibe, T. Itoh, PhotoSurfing: A 3D Image Browser Assisting Association-Based Photo-graph Browsing, *NICOGRAPH International*, 2009.

[5] A. Nagasaki, T. Itoh, M. Ise, K. Miyashita, A Correlation-based Hierarchical Data Visualization Technique and Its Application to Credit Card Fraud Data, *1st International Workshop on Super Visualization (in conjunction with the 22nd ACM International Conference on Supercomputing)*, 2008.

[6] W. Peng, M. O. Ward, E. A. Rundensteiner, Clutter Reduction in Multi-Dimensional Data Visualization Using Dimension Reordering, *IEEE Symposium on Information Visualization*, 89-96, 2004.

[7] M. Sips, B. Neubert, J. P. Lewis, P. Hanrahan, Selecting Good Views of High-Dimensional Data Using Class Consistency, *Computer Graphics Forum*, 28(3), 831-838, 2009.

[8] J. Yang., J. Fan, D. Hubball, Y. Gao, H. Luo, W. Ribarsky, M. Ward, Semantic Image Browser: Bridging Information Visualization with Automated Intelligent Image Analysis, *IEEE Visual Analytics in Science and Technology*, 191-198, 2006.

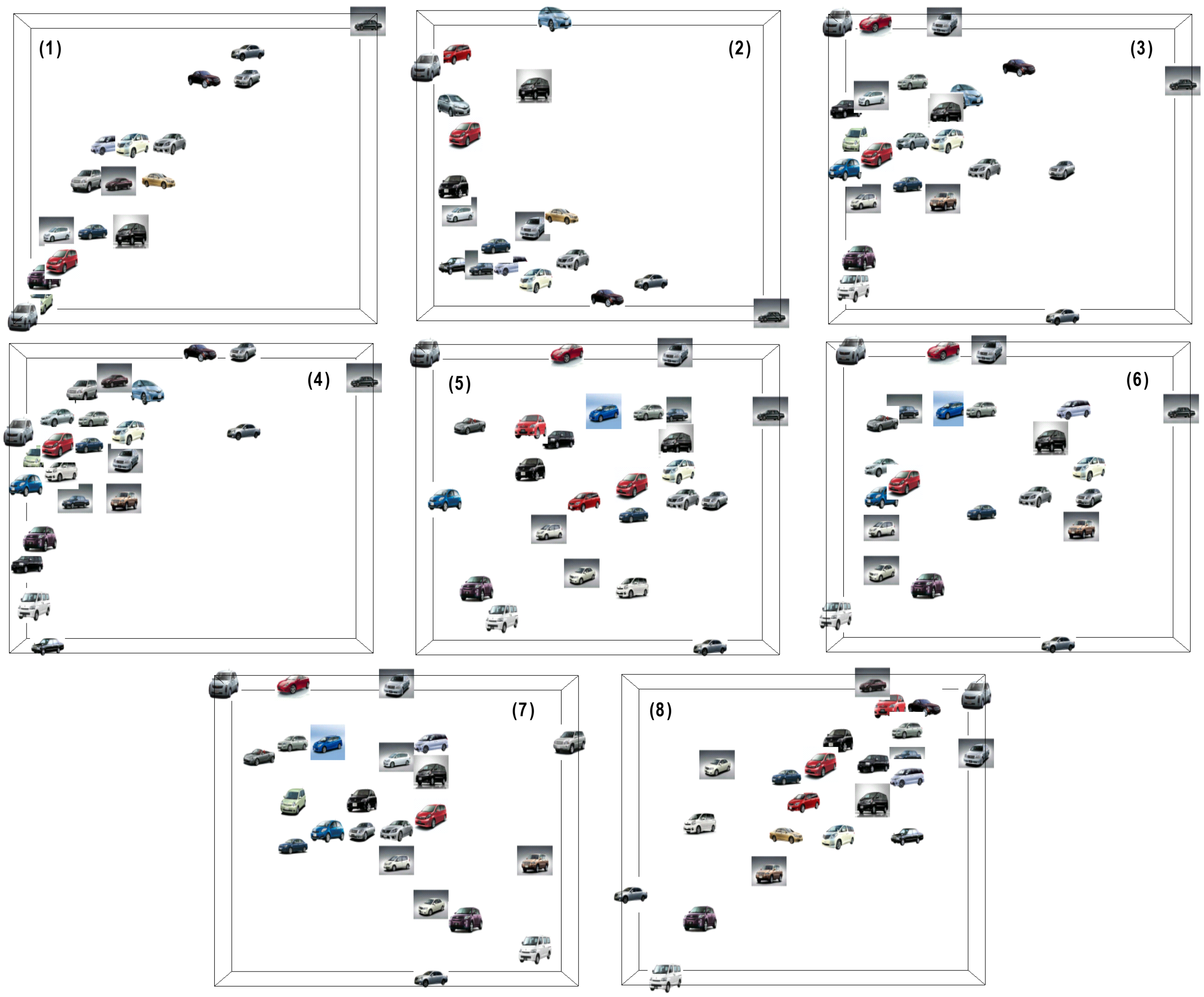

Figure 3. Examples with an image collection of cars. (1) Price for X-axis, and displacement for Y-axis. (2) Price for X-axis, and fuel cost for Y-axis. (3) Price for X-axis, and equipment for Y-axis. (4) Price for X-axis, and appearance evaluation for Y-axis. (5) Outer length for X-axis, and appearance evaluation for Y-axis. (6) Outer width for X-axis, and appearance evaluation for Y-axis. (7) Height of floor for X-axis, and appearance evaluation for Y-axis. (8) Appearance evaluation for X-axis, and cost performance evaluation for Y-axis.