

A Visualization Technique Using Loop Animations

Takao Ito^{1(✉)} and Kazuo Misue²

¹ Department of Computer Science, University of Tsukuba,
Tennodai, Tsukuba 305-8573, Ibaraki, Japan
ito@vislab.cs.tsukuba.ac.jp

² Faculty of Engineering, Information and Systems, University of Tsukuba,
Tennodai, Tsukuba 305-8573, Ibaraki, Japan
misue@cs.tsukuba.ac.jp

Abstract. Animation extends the dimensions of spaces for visual representations by adding a temporal dimension. Although various techniques using animation have been proposed, most of them use a temporal dimension in the representation space only for representing temporal data. To represent quantitative data, an animation technique is proposed. A short animation played repeatedly is called a “loop animation.” The frequency of loop animations can represent quantitative data. Four variations of loop animations were designed and an experiment was conducted to evaluate their accuracy in representing quantitative data. In the experiment, loop animations were compared with static visual representations. The results showed that animations, in which either marks were spun or sizes were vibrated, were more accurate than visual representations using brightness and area.

Keywords: Visualization · Loop animation

1 Introduction

Visualization is often used for analyzing and presenting data. It helps in understanding complex data in analyzing tools. By using this in websites, viewers can understand the given data in an easier manner.

It is evident that a large number of various visual representations are required. Many techniques to analyze large-scale and complex data are also required. Attractive representations are required to present data to viewers. In order to meet the increasing demand, extending spaces for visual representations is necessary.

Animation is one of the most effective methods to extend design spaces for visual representations. It extends the dimensions of spaces for visual representations by adding a temporal dimension that can attract viewers. Although various techniques using animation have been proposed, animations are used in most techniques to represent motions or changes [1, 10]. In other words, the techniques use a temporal dimension in the representation space only for representing temporal data.

In this research, a technique to represent quantitative data by using short repeating animations is proposed. We call this technique “Loop Animation.” It is proposed that the frequency of cycles or changes in the sizes in loop animations can represent quantitative data. If animations can represent quantity, the dimensions of spaces for visual representations are extended.

Frequency is used to represent quantitative data in the real world, but usage and accuracy of such representations have not been studied enough. For example, a certain battery charger represents state-of-charge by the frequency of the blinking of a lamp. Access lamps of HDD or LAN flash when devices are working. Users of such devices interpret the degree of working from the frequency of blinking. However, it is not understood what kind of representations can show quantitative data more accurately.

The goal is to establish methods to represent quantitative data using animations. First, variations of loop animations based on traditional static visual variables were designed. Then, an experiment to investigate the accuracy of their representations for quantitative data was conducted. In the experiment, loop animations were compared with static visual representations.

2 Related Work

In the present study, we investigated the techniques for representing quantitative data to extend the dimensions of spaces for visual representations. First, we investigated the accuracy of representing quantitative data with traditional visual representations, and then, we study animations in visualization.

2.1 Static Visual Representations and Their Accuracy

Some studies have examined the accuracy of static visual representations for representing quantitative data. Bertin [2] showed that the position, size, and color of a mark are able to represent the quantity. Cleveland and McGill [3] showed that the position is more accurate for representing quantitative data than the length. Mackinlay [8] described that the length and orientation of a mark are also able to represent quantity relatively precisely.

There are some static visual representations that can represent quantity. However, accuracy of animations for representing quantitative data has not been investigated so far.

2.2 Usage of Animations in Visualization

Animations are often used in information visualization. In particular, animated transitions are used in several visualization systems such as ScatterDice [4] and DiffAni et al. [10]. It is known that animation is an efficient method to represent category data. Munzner [9] described that the direction and frequency of spin can represent categorical data.

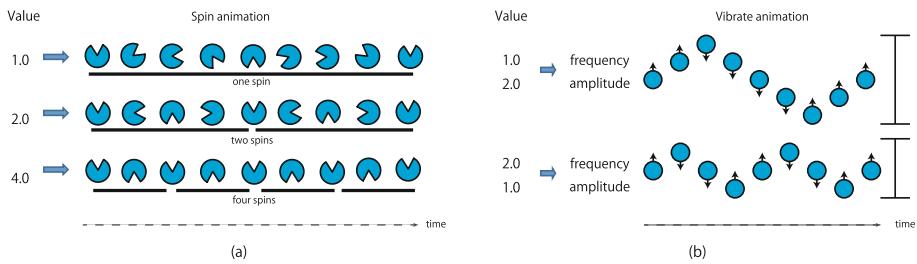


Fig. 1. Examples of loop animations. (a) Values are encoded to the frequency of spins. (b) Values are encoded to the amplitude and frequency of vibrations.

Several researchers have studied various features of animations. Archambault et al. [1] compared the effects of animation and small multiples. Heer and Robertson [5] studied the effects of animated transitions in statistical data graphics. Huber and Healey [6] showed that humans are able to distinguish the presence or absence of a small group of target elements that flicker at a rate different from background elements.

Although some investigations have been carried out on animation techniques for visualization, animation techniques to represent quantitative data have not been researched.

3 Representing Quantitative Data with Loop Animations

Herein, we introduce some examples to represent quantitative data with loop animations. Figure 1(a) illustrates examples of representation methods with loop animations where the marks spin. These loop animations represent quantitative data based on the frequency of the spin according to the quantitative value. Figure 1(b) illustrates some examples where the marks vibrate. These loop animations represent two quantitative values based on the amplitude and frequency of vibration. We believe that the frequency and amplitude of loop animations can represent quantitative data.

It was assumed that loop animations are used together with existing visualization techniques. Visualization designers often consider the use of Bertin's static visual variables for designing visual representations. Therefore, using Bertin's static visual variables as the basis, we designed four variations of loop animations that can represent quantitative data (Table 1).

In Table 1, each row shows a variation of loop animations, and Bertin's static visual variables used for loop animations are written in the column named "Visual Variable." Examples of each variation are drawn in the column named "Example." A Spin animation changes the orientation of a mark according to time. A Blink animation changes the color of a mark according to time. A Vibrate animation changes the position or size of a mark according to time. A Morph animation changes the level of morphing of the shape or texture of a mark according to time.

Table 1. Variations of loop animations

Variation	Visual Variable	Example
Spin	Orientation	 spin of a cut circle
Blink	Value, Color	 blink of value blink of color(hue)
Vibrate	Position, Size	 vertical vibrate horizontal vibrate size vibrate
Morph	Shape, Texture	 shape morph texture morph

4 Experiment to Evaluate Accuracy

Accuracy, i.e., how correctly the values can be interpreted from representations, is one of the important aspects of visual representations of quantitative data. Therefore, an experiment was conducted to compare traditional static visual representations with loop animation in terms of accuracy of representing quantitative data.

4.1 Tasks

If the variance of values read from a certain visual representation is low, it can be said that the visual representation has high accuracy. Based on such an idea, a task was designed in which participants read values from visual representations.

In the tasks, five marks were shown on the screen (Fig. 2). The physical parameter of each mark, like the size and brightness of the mark, and the frequency of loop animations, was controlled by a value. This value is called “given value.” The physical parameters were proportional to the given values. The given value of the first mark from the left was 0, and the given value of the second mark was 1. Given the values of the third to fifth marks were random values between 0 and 3, a circle was drawn below the fourth mark. It was assumed that loop animations are used in the environment where many marks are drawn, like in Figs. 5 and 6. Therefore, five marks were shown.

The participants were told that the first mark represented 0 and the second mark represented 1. They answered a value represented by the fourth mark by referring to the first and second marks. Values read from representations are called “judged values.” The participants used the slider bar under the marks to answer.

The range of the slider bar varied from 0 to 3 with 0.1 steps. The participants finished the task by clicking the “OK” button after controlling the slider bar.

There was no time limit set for answers. The participants were told to answer as soon as and as correctly as possible.

4.2 Visual Representations Used in the Experiment

Length, Slope (Tilt), Area, and Density (Brightness) were used as static visual representations (Table 2). Table 3 shows the ranking of the static visual representations for quantitative data provided by Cleveland et al. and Mackinlay. Although this ranking was made by considering various psychological aspects, it is useful as an index of accuracy. Comparison targets were narrowed down to reduce the load of participants. Length was chosen as the delegate of representations, which have high accuracy. The accuracy of Angle and Slope was almost the same [8]. Therefore, Angle was removed from the comparison targets. Volume was excluded from the comparison targets because it is hardly used. Brightness was chosen because it is the most accurate parameter among the color-related representations: Hue, Saturation, and Brightness.

Spin, Blink of Brightness (Value), Vertical Vibrate, Horizontal Vibrate, Size Vibrate, and Shape morph were used as loop animations. The simplest possible variation among the various alternatives was chosen in each type of loop animations. Finally, the variations enclosed with red rectangles shown in Table 1 were used in the experiment. Cut circles were used for Spin. The brightness of gray scale was used for Blink. Circles were used for Vertical Vibrate and Horizontal Vibrate, and squares were used for Size Vibrate. The morph of between a circle and a square was used for Morph. For any animation, “stop” was assigned to 0, and the given values controlled the frequency of animations. For Spin, the frequency was double the given value; for other animations, the frequency was a given value.

Table 2. The list of static visual represents used in the experiment. In figures of Example column, given values fo marks were 0, 0.5, 1, 1.5, 2, and 2.5 sequentially from the left.

Represent	Example	Encode
Length	· · · · · · ·	The length of a bar is proportional to the given value.
Slope(Tilt)	↑ ↗ ↘ ↓ ↙ ↖	The tilt of a bar is proportional to the given value. A bar is vertical when the value is 0, and the angle of tilt is 30° when the value is 1.
Area	· · ■ ■ ■ ■	The area of a rectangle is proportional to the given value.
Density(Brightness)	● ● ● ● ● ●	The brightness of a circle is proportional to the given value. The color is black when the value is 0, and the color is white when the value is 3.

Table 3. Ranking of static visual variables to represent quantitative values. Underlined variables were used in the experiment.

Rank	Visual variable
1	Position
2	<u>Length</u>
3	Angle
4	Slope(Tilt)
5	<u>Area</u>
6	Volume
7	Density(Brightness)
8	Color saturation
9	Color hue

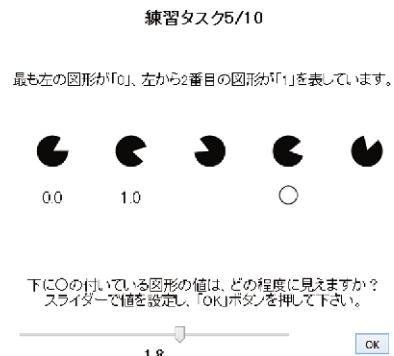


Fig. 2. An example of a task screen. The participants selected the value by the slider bar and clicked “OK” button to finish the answer.

For loop animations, there are many variations in marks used in animations, color, and the speed of animation. Such slight differences affect accuracy. To extensively investigate the effects by experiments, the participant were forced with a heavy load. Therefore, experiment conductors selected comparison targets through preliminary surveys of variations of loop animations. Various visual representations were surveyed by using Iv Studio [7], which is a development environment for visualization methods. Iv Studio has a dataflow visual language for designing visual representations, and it supports the implementation and tuning of visual representations interactively. For example, 12 marks were tried for Spin, and animations of hue, saturation, and brightness were tried for Blink. The speed of animations was tuned so that the animations could be read easily and participants do not get tired.

4.3 Procedure

Fourteen students from the Department of Computer Science, University of Tsukuba, participated in the present experiment. Out of these, 8 participants were members of a laboratory whose research field was Information Visualization.

A set consists of 10 tasks for 10 visual representations. The order of visual representations in a set was decided at random to remove the order effect. The given values of the third, fourth, and fifth marks varied from 0.2 to 2.8, and these values were decided at random.

The operations of tasks were explained to the participants; then, one set was given to them for practice. During practice, they were explained how to read each visual representation. For example, they were told that for Density (Brightness), the brighter color represented higher values and in Area, values were represented by the area. After the practice, the participants performed three sets of tasks,

rested for 3 min, and then again performed three sets of tasks. In other words, six sets (60 tasks) were carried out. Finally, the participants were told to answer a questionnaire about the free description. It was assumed that the experiment would take about 30 min.

In the experiment, the visual representations implemented by Iv Studio as Lua's scripts¹ were exported, and the scripts were run on web pages with `lua.vm.js`². The Web browser was Mozilla Firefox 42.0. The monitor used in the experiment was Eizo ColorEdge CG277.

4.4 Result

The result of the experiment is shown as scatter plots (Fig. 3). In these charts, the given values of the fourth marks are represented by the x-axis, and the judged values by the y-axis. Answers were considered “wrong” when the participants answered with a value bigger than 1.0 when the given value was smaller than 1.0, or when they answered with a value smaller than 1.0 when the given value was bigger than 1.0. Twenty five answers were considered wrong, and were excluded from the result.

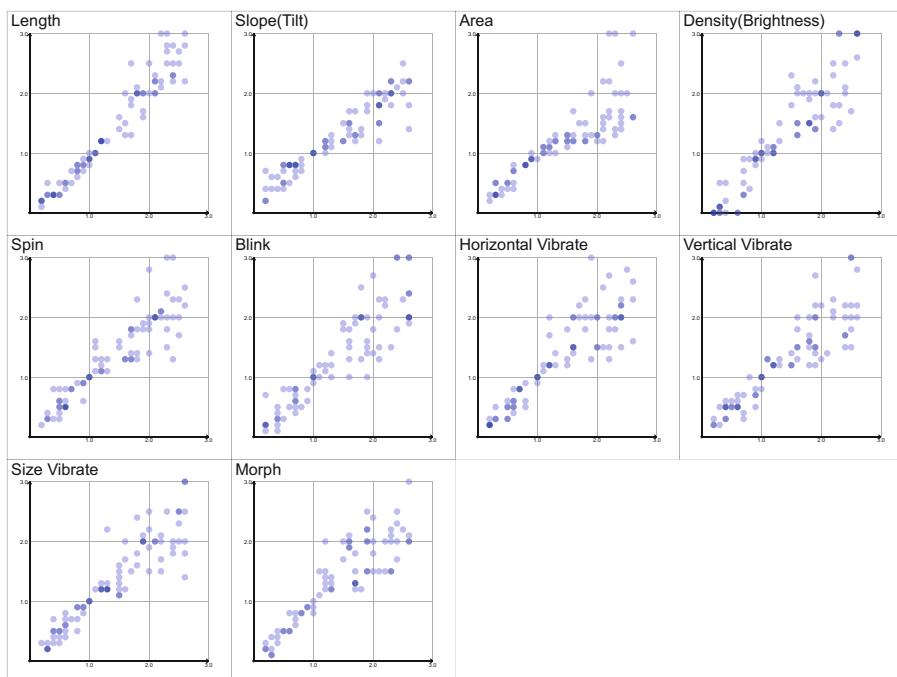


Fig. 3. Result of the experiment

¹ <http://www.lua.org/>.

² <https://kripken.github.io/lua.vm.js/lua.vm.js.html>.

Two indexes were used for computing the variance of answers. The smaller the values of indexes are, the more accurate the visual representation is.

Error of mean square obtained from the given values was used as the first index. Computation was done for the first index $e1$ with Eq. 1, where n is the number of answers for a visual representation; $v = (v_1, v_2, \dots, v_n)$ is a sequence of the given values, and $j = (j_1, j_2, \dots, j_n)$ is a sequence of the judged values.

$$e1 = \frac{\sum_{k=1}^n (v_k - j_k)^2}{n} \quad (1)$$

Next, the distortion of the sense that humans receive was considered. It is known that the relationship between the magnitude of a physical stimulus and its perceived intensity or strength is based on the power laws (Stevens' power law; Eq. 2).

$$E(I) = kI^a \quad (2)$$

Using this law, we computed the second index $e2$ (Eq. 3).

$$e2 = \frac{\sum_{k=1}^n (E(v_k) - j_k)^2}{n} \quad (3)$$

Coefficients of the power trendline were taken as k and a in Eq. 2. In Table 4, k , a , $e1$, and $e2$ are shown.

The outcome of static visual representations was analyzed to validate the experimental results. It was found that Length and Slope (Tilt) were more accurate than Area and Density (Brightness). This result matched that of a previous work [3,8]. In particular, the order of $e1$ was almost the same as that of the previous work (Table 3). Therefore, it can be concluded that the present experiment is valid.

Loop animations were compared with static visual representations. The results showed that Spin and Size Vibrate were more accurate than Area and Density (Brightness) in both $e1$ and $e2$. Other loop animations were as accurate as Area and Density (Brightness).

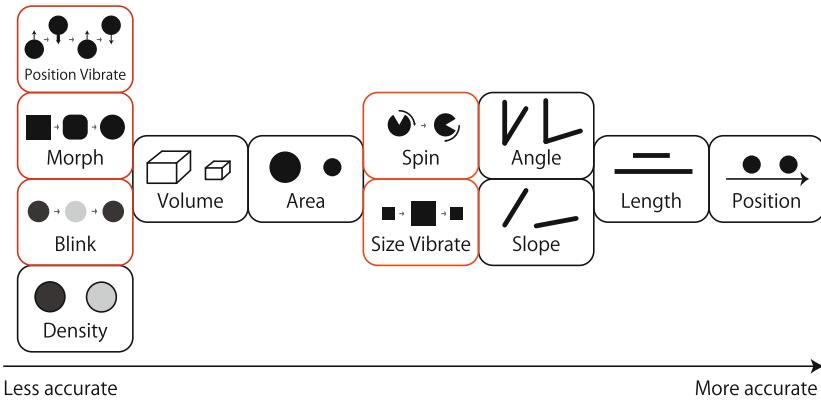
In loop animations, Spin was the most accurate representation, followed by Size Vibrate. The accuracy of Blink, Position Vibrate (Vertical Vibrate and Horizontal Vibrate), and Morph was low.

With regard to k and a , loop animations were closer to 1 than Area and Density (Brightness). The results showed that the distortion of representations by the frequency of loop animations was low.

From the above results, it can be concluded that loop animations can be used to represent quantitative data instead of Area or Density (Brightness). Furthermore, it was found that Spin and Size Vibrate can represent quantitative data more accurately than Area and Density (Brightness). Accuracy ranking of visual representations obtained by considering loop animations and static visual representations is shown in Fig. 4.

Table 4. Result of the experiment

Represent	k	a	$e1$	$e2$
Length	0.950	1.112	0.057	0.048
Slope(Tilt)	1.010	0.739	0.088	0.038
Area	0.906	0.811	0.189	0.104
Density(Brightness)	0.816	1.285	0.116	0.114
Spin	0.967	0.947	0.092	0.081
Blink	0.903	1.005	0.130	0.117
Horizontal vibrate	0.973	0.947	0.114	0.104
Vertical vibrate	0.923	0.923	0.121	0.095
Size vibrate	0.958	0.960	0.097	0.086
Morph	0.938	1.029	0.119	0.114

**Fig. 4.** Accuracy ranking of visual representations for quantitative data. Red visual representations are loop animations. (Color figure online)

4.5 Discussion

Spin was more accurate than other loop animations because it was easy to compare the speed of spin. In order to read quantitative values from the frequency or speed of loop animation, length of the cycles of animations has to be read. Because it was easy to decide the start point of one cycle, it was thought that among our tasks, Spin was the easiest to perform. In Blink and Position Vibrate, the start point of one cycle was not clear; hence, their accuracy was low. It was evident that more accurate representations can be designed by improving the wave to afford clarity to the standard of one cycle.

Many participants commented that it looked like that marks were working together in Blink and Position Vibrate. We believe that it is one of the reasons for the low accuracy of Blink and Position Vibrate. In future, more research should be carried out on the influence of neighboring marks.

Two reasons for low accuracy have been given; however, the reason for low accuracy of Morph has not been determined yet. Representations of Morph used in the experiment were similar to those of Size Vibrate, and a participant commented that Morph's start point of one cycle was clearer than that of Size Vibrate, and therefore, Morph was easy to read. However, the accuracy of Morph was low. More investigation needs to be conducted to determine the reason for the low accuracy of Morph.

The variation of Blink also needs to be investigated more. In order to simplify, only brightness was used for Blink. However, it was evident that hue and saturation affect the readability of Blink. Therefore, it is possible to design more accurate representations by considering hue and saturation.

5 Use Cases

In this section, two examples of visualization using loop animations are shown. Sample videos of these examples are available from the first author's web site³.

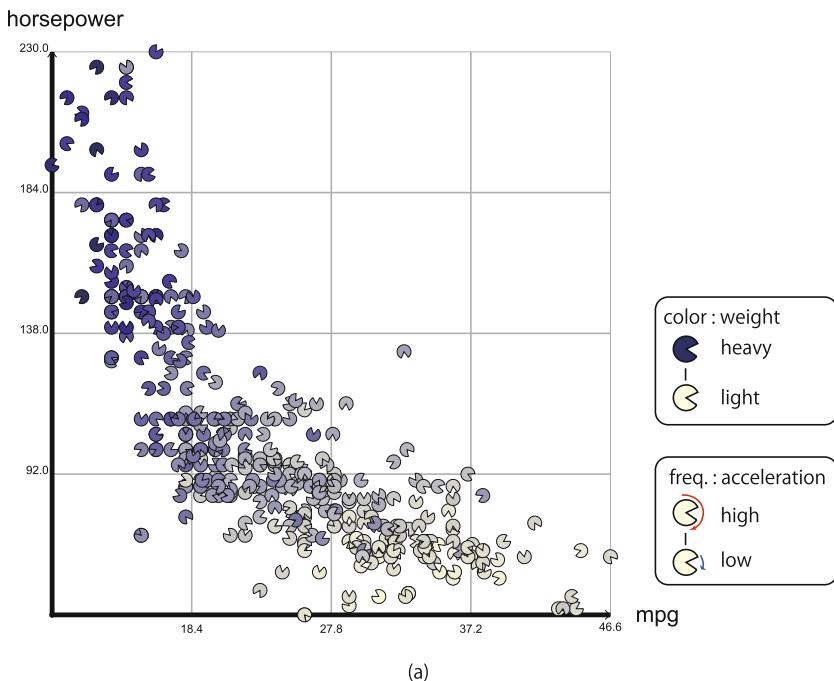


Fig. 5. An example of visualization with loop animations. Each cut circle spins. (Color figure online)

³ <http://www.vislab.cs.tsukuba.ac.jp/members/ito/loopanim/>.

Figure 5(a) illustrates the data of cars⁴ that contain 406 speculations about these cars. In this figure, X positions represent fuel economy, Y positions represent horsepower, and colors of the cut circles represent weight. In addition, each cut circle spins. The frequency of the spins represents acceleration. This chart was shown to six viewers, and they were asked about the kind of information they could read. The viewers found a trend that cars with higher horsepower had higher acceleration. They also found a trend that horsepower affected acceleration more than the weight. In this way, it was found that the viewers could read trends of data from loop animations.

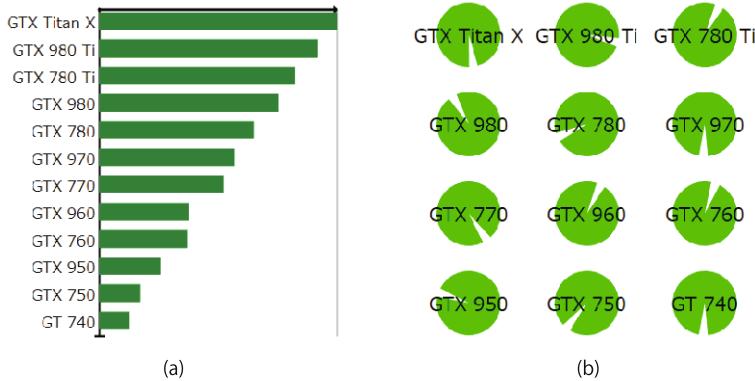


Fig. 6. Processing power of GPUs. (a) The processing power is illustrated by a bar chart. (b) The processing power is represented by the speeds of spins.

Figure 6 illustrates processing power of 12 GPUs (graphics processing unit)⁵. Figure 6(a) is a bar chart where x-axis represents processing power (GFLOPS). Such bar charts are often seen on web articles; the charts in Fig. 6 show that displays of GPUs' names are small and white space in the right-bottom is large. In Fig. 6(b), processing powers are represented by spin animations. Spin animations allow visualization designers to use space freely, and hence, designers can display GPUs' name in large fonts. Representation by Area also allows the same layout; spin animations can represent data more accurately. Additionally, spin animations can convey the mean of processing power to the viewer intuitively. It was evident that Fig. 6(b) is suitable for web advertisements where designers want to attract viewers. In this way, loop animations could be used to increase the flexibility of the design and extend the possibility of designing new visual representations.

6 Conclusion

Usage of loop animations was proposed as a method to represent quantitative data. First, four variations of loop animations were obtained. These were

⁴ <http://stat-computing.org/dataexpo/1983.html>.

⁵ https://en.wikipedia.org/wiki/List_of_Nvidia_graphics_processing_units.

designed based on traditional static visual variables so that loop animations could be used along with other visual variables. Then, an experiment was conducted wherein the accuracy of representations for quantitative data was evaluated. As a result, it was found that Spin and Size Vibrate loop animations were more accurate than Area and Brightness. With the help of use cases, it was found that loop animations could be used to represent the trends of data and increase the flexibility of the design.

References

1. Archambault, D., Purchase, H.C., Pinaud, B.: Animation, small multiples, and the effect of mental map preservation in dynamic graphs. *IEEE Trans. Vis. Comput. Graph.* **17**(4), 539–552 (2011)
2. Bertin, J.: *Semiology of Graphics: Diagrams, Networks Maps*. University of Wisconsin Press, Madison (1984)
3. Cleveland, W.S., McGill, R.: Graphical perception: theory, experimentation, and application to the development of graphical methods. *J. Am. Stat. Assoc.* **79**(387), 531–554 (1984)
4. Elmqvist, N., Dragicevic, P., Fekete, J.D.: Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. *IEEE Trans. Vis. Comput. Graph.* **14**(6), 1141–1148 (2008)
5. Heer, J., Robertson, G.: Animated transitions in statistical data graphics. *IEEE Trans. Vis. Comput. Graph.* **13**(6), 1240–1247 (2007)
6. Huber, D.E., Healey, G.C.: Visualizing data with motion. In: *Visualization, 2005 VIS 2005*, pp. 527–534. IEEE (2005)
7. Ito, T., Misue, K., Tanaka, J.: A development environment for InfoVis system using dataflow visual language. In: *IPSJ SIG Technical report, HCI, 2015-HCI*, vol. 162(22), pp. 1–8 (2015). (in Japanese)
8. Mackinlay, J.: Automating the design of graphical presentations of relational information. *ACM Trans. Graph.* **5**(2), 110–141 (1986)
9. Munzner, T.: *Visualization Analysis and Design*. AK Peters/CRC Press, Boca Raton (2014)
10. Rufiange, S., McGuffin, J., DiffAni, M.: Visualizing dynamic graphs with a hybrid of difference maps and animation. *IEEE Trans. Vis. Comput. Graph.* **19**(12), 2556–2565 (2013)