ICIC Express Letters
Volume **11**, Number **5**, May **2017**
ICIC International ©2017 ISSN 1881-803X
pp. 939–945

A VISUALIZATION PLATFORM FOR SCALING-INVARIANT BOUNDARY IMAGE MATCHING

Seongwoo Moon, Sanghun Lee and Yang-Sae Moon*

Department of Computer Science
Kangwon National University
1 Kangwondaehak-gil, Chuncheon-si, Gangwon-do 24341, Korea
{ swmoon; sanghun }@kangwon.ac.kr; *Corresponding author: ysmoon@kangwon.ac.kr

Received November 2016; accepted February 2017

Abstract. *In this paper, we present a visualization platform that supports the scaling invariance in boundary image matching. Once converting image boundaries into time-series, we can use efficient time-series matching techniques and perform the boundary image matching fast even for large-scale image databases. In this boundary image matching, supporting the scaling invariance is very important to retrieve similar images much more correctly. In this paper, we design and implement the scaling-invariant boundary image matching as a client-server visualization platform. The client converts a query image into a time-series, sends it to the server with scaling ranges, and finally visualizes the returned results. The server performs the scaling-invariant matching efficiently on the boundary time-series database with multidimensional indexes. Experimental results show that our visualization platform performs the scaling-invariant matching very correctly by representing the matching results through various charts and graphs.*

Keywords: Scaling invariance, Boundary image matching, Boundary time-series, Scaling-invariant matching

1. **Introduction.** In recent years, there have been many research efforts in visual data mining which enables data analysts or naive users to intuitively understand the data mining results on a variety of image, text, and time-series data [8]. Visualization is a very useful mechanism that represents meaningful information or implicit knowledge through images, charts, graphs, or animations for the more intuitive and easier understanding. Through the visualization, we can improve efficiency of recognition, understanding, analysis, and process of raw data and data mining mechanisms [8, 13].

Boundary image matching [3] is a novel image matching approach that identifies similar images by using time-series extracted from image boundaries. By using time-series rather than image boundaries, we can apply efficient time-series matching techniques to the image matching, and accordingly, we can handle the large-scale image databases in the simple time-series domain rather than the complex image domain. In particular, we can use lower-dimensional transformations and multidimensional indexes [1, 2] for improving the matching performance. Recently, there has been an interesting approach that supports the scaling invariance in boundary image matching, that is, it identifies similar images even though they are arbitrary scaled from original images [9]. Using the scaling-invariant boundary image matching, we can retrieve the data images whose shapes are similar to the query image even though their horizontal or vertical scales are different. For the scaling-invariant matching [9], we construct the scaled time-series using scaling factors and exploit those scaled time-series to support the scaling-invariant property in the boundary image matching. Figure 1 shows how we convert an original time-series to a scaled time-series, and Figure 2 depicts an example of scaling-invariant distances between query and data images. Readers are referred to [9] for the detailed working mechanism of how to get scaled time-series and how to perform scaling-invariant boundary image matching.

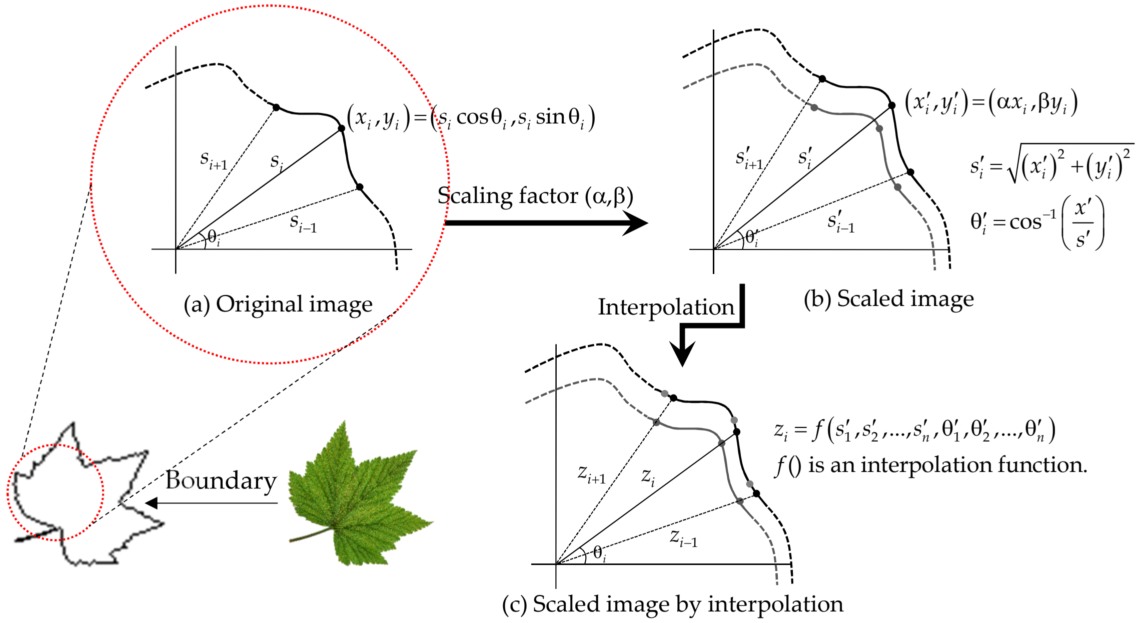


FIGURE 1. Scaling techniques in scaling-invariant boundary image matching

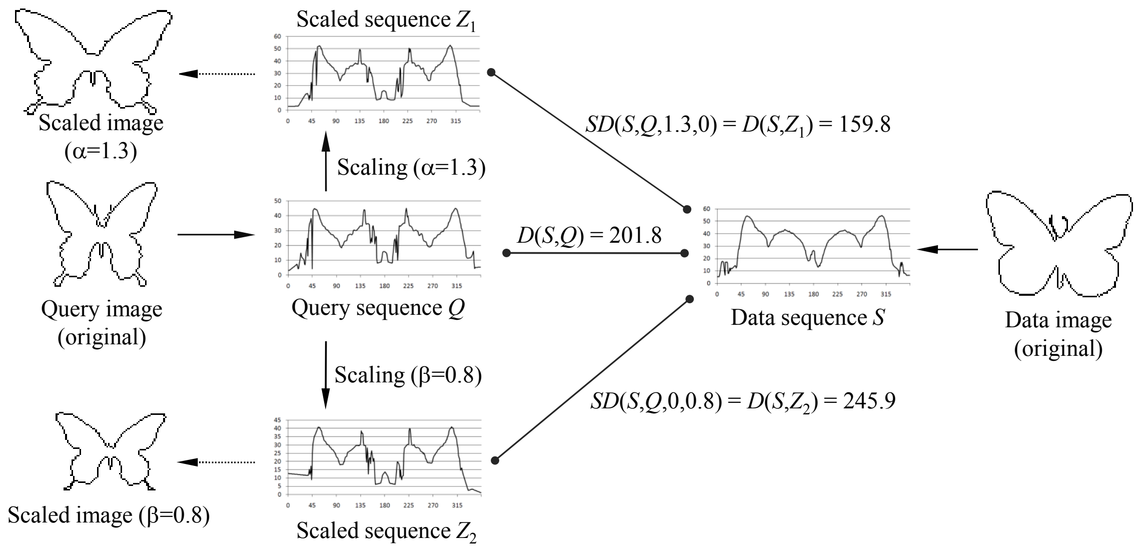


FIGURE 2. Scaling-invariant distances between query and data images

In this paper, we design and implement a visualization platform that intuitively represents the result of scaling-invariant boundary image matching [9]. The previous solution provides the matching result in a plain text form, which is very difficult for the users to understand what the result means. To solve this problem, in this paper we visualize the scaling-invariant matching result through various charts and graphs, which are much easier to understand and to get important insights behind the matching result. The proposed platform uses an image chart to display the scaling-invariant similar images, and we can easily understand why those returned images are similar to the given query image. The platform also provides line and polar charts which enable us to intuitively compare the boundaries of query and returned data images. It further presents a distance graph, which shows scaling-invariant distances from a query to k-nearest neighbor (k-NN) data time-series, and using the graph we can intuitively identify which parameter values are suitable for k in the k-NN queries.

The rest of this paper is organized as follows. Section 2 describes the related work on time-series matching and boundary image matching. Section 3 presents the proposed visualization platform that supports scaling-invariant boundary image matching. Finally, Section 4 concludes the paper and explains the future work.

2. **Related Work.** *Time-series matching* is the problem of finding the data time-series similar to the given query time-series [1, 8]. In this paper, we use the *range search* of time-series matching, which retrieves the data time-series whose distances to the query time-series are within the tolerance ϵ. As the similarity model, we use the Euclidean distance [1, 9] since it can be easily understood in visual representation. Our platform can also be easily extended to k-NN queries instead of range queries and to the dynamic time warping (DTW) [10] distance instead of the Euclidean distance.

In recent years, there have been a few efforts on visualization of time-series matching results [8]. First, [5] visualizes the process of computing DTW distances, so we can intuitively understand how the DTW distance is computed between two time-series. Second, [12] visualizes time-series data as various spiral shapes by varying spiral thickness and color degrees [8]. These two approaches, however, focus on *pure* time-series rather than *image* time-series.

Boundary image matching has been studied as a representative application of time-series matching [3, 9, 7, 11], and there have also been a few efforts on visualization of the boundary image matching [6, 8]. First, [6] visualizes the process of *naive* boundary image matching without supporting any transformation. It uses line and polar charts to display the results of range and k-NN search queries for naive image time-series data. Second, [8] presents a visualization tool for *rotation-invariant* boundary image matching [7, 11]. It visually displays the rotation-invariant property using image, line, and polar charts. These two solutions, however, do not support the scaling invariance, which is the major technical contribution of this paper.

3. **Visualization Platform of Scaling-Invariant Boundary Image Matching.**

3.1. **Working framework.** We design a scaling-invariant boundary image matching platform as a client-server model like in Figure 3. First, the client converts a query image given by a user into a time-series by the centroid contour distance (CCD) method [3] and sends it to the server with the tolerance ϵ and scaling ranges of $[\alpha_s, \alpha_e]$ and $[\beta_s, \beta_e]$ [9]. Second, the server constructs a range query using the query time-series and scaling ranges, evaluates the query on the multidimensional index, and returns the matching result to the client. Finally, the client visualizes the returned result using various charts and graphs.

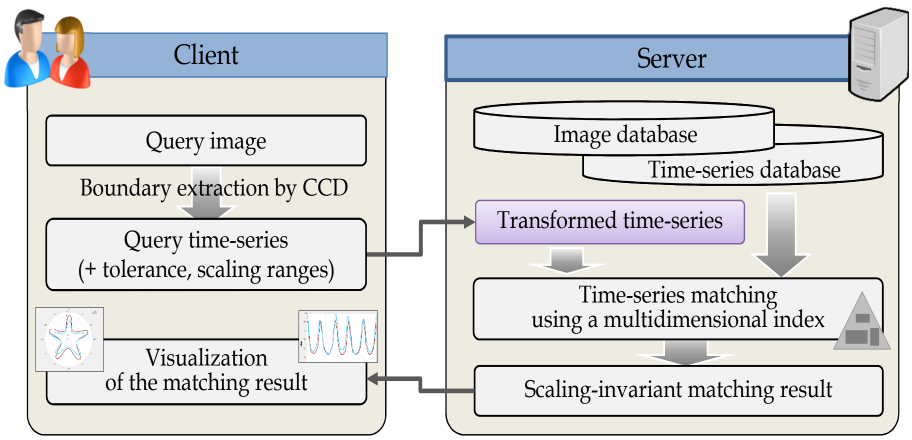


FIGURE 3. Working framework of the scaling-invariant boundary image matching

3.2. **Implementation of scaling-invariant visualization platform.** The experimental environment of the implemented client-server platform is as follows. In the client, we use C# language and MSChart of Microsoft Visual Studio 2010 on Windows 7 operating system. In the server, we use GNU C language on Cent OS 5.9 Linux operating system. We crawl 10,000 images from the Web [6] and use total 90,000 images, which consist of 10,000 original crawled images and 80,000 their scaled images. For time-series matching, we extract a time-series of length 360 from each image by the CCD method. We use an R*-tree [1, 3] as a multidimensional index and piecewise aggregate approximation (PAA) [4] as a lower dimensional transformation of using the R*-tree. In the experiment, we set the tolerance ϵ to 70 and the scaling range of α and β to $[0.75, 1.25]$.

Figure 4 shows an initial screenshot of the implemented scaling-invariant matching platform. We choose an input image using the "Open" button of Part Ⓐ and use the "Search" button to execute the matching process. The "Option" button of Part Ⓐ browses Part Ⓓ to get the tolerance ϵ and scaling ranges of α and β. Using Part Ⓑ, we provide a list of scaling-invariant similar images received from the server, where each row represents a resulting data image by its rank, identifier, and distance from the query image. Finally, Part Ⓒ displays graphs or charts for visualizing the matching result.

Figure 5 shows the visualization result, where we use a starfish image as a query image. Figure 5(a) shows the result of *naive* boundary image matching, and Figure 5(b) shows that of *scaling-invariant* boundary image matching. As shown in Figure 5(a), the naive matching displays wrong images having similar sizes, and it cannot find very similar images having different scales. On the other hand, in Figure 5(b), the scaling-invariant matching identifies a variety of scaled, similar images very correctly.

Figure 6 shows a line chart that displays data and query images using their time-series. By selecting a data image from Part Ⓑ, we can compare its time-series with the query time-series in the line chart of the same window frame. As shown in the figure, we can intuitively understand how query and data time-series are different or similar with each other.

Figure 7 shows the polar charts that visualize boundaries of data and query images, where we also use starfish images. In Figure 7(a), we can easily recognize that the retrieved data image is scaled *horizontally* from the query image. In Figure 7(b), we can also easily know that the returned data image is scaled *vertically* from the query image. Likewise, by comparing data and query images using the polar chart, we can intuitively understand how and why those two images are similar or different with each other.

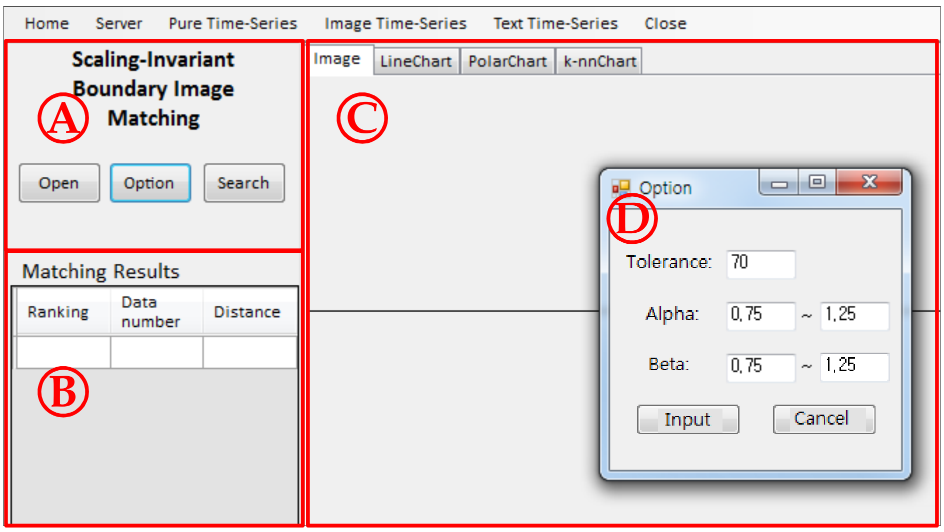
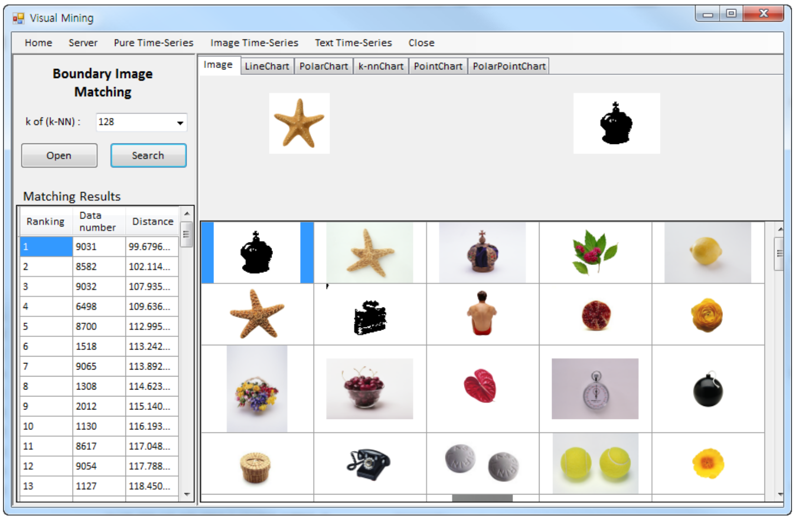
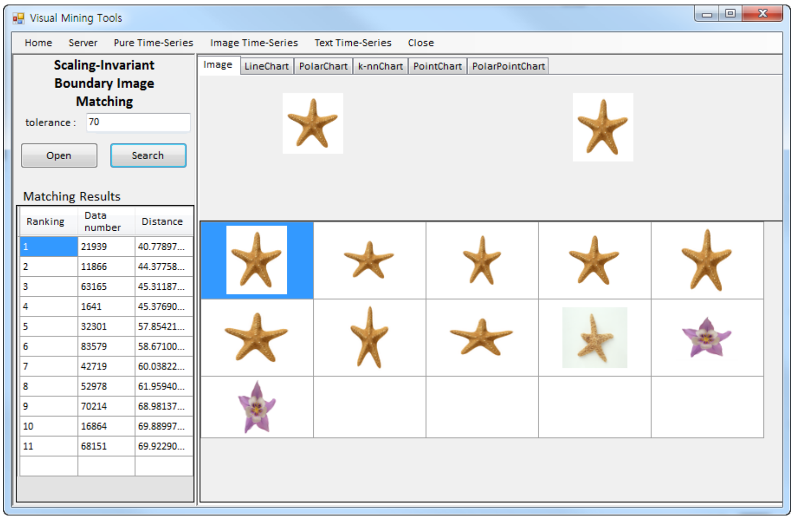


FIGURE 4. Initial screenshot of the scaling-invariant visualization platform



(a) Naïve matching (no scaling invariance).



(b) Scaling-invariant matching.

FIGURE 5. Visualization results of naive and scaling-invariant matching

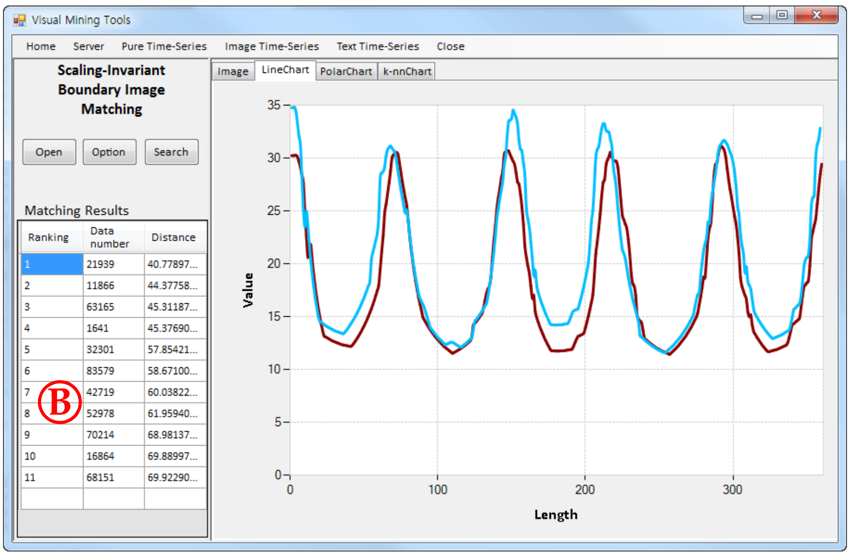
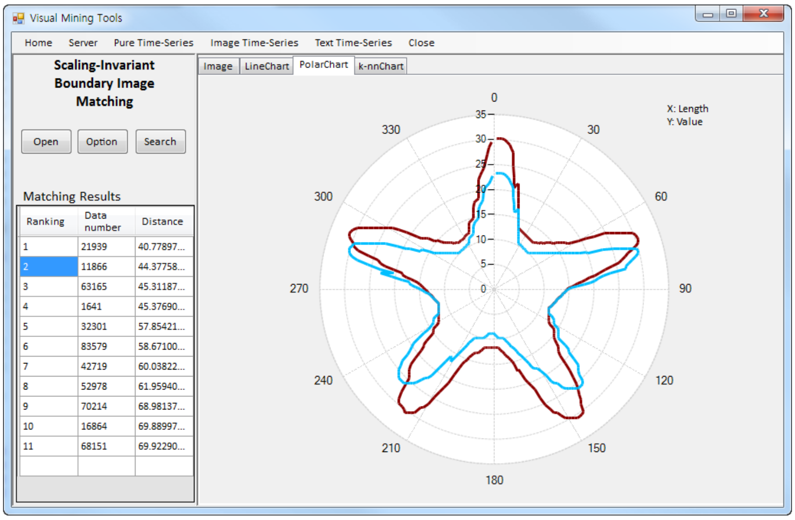
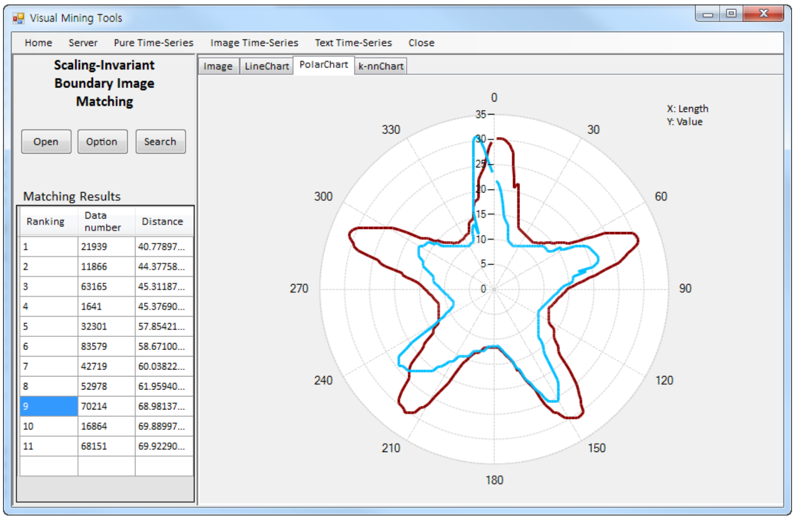


FIGURE 6. Comparison of query and data time-series using a line chart



(a) Starfish image and its horizontally scaled image.



(b) Starfish image and its vertically scaled image.

FIGURE 7. Comparison of query and data image boundaries using a polar chart

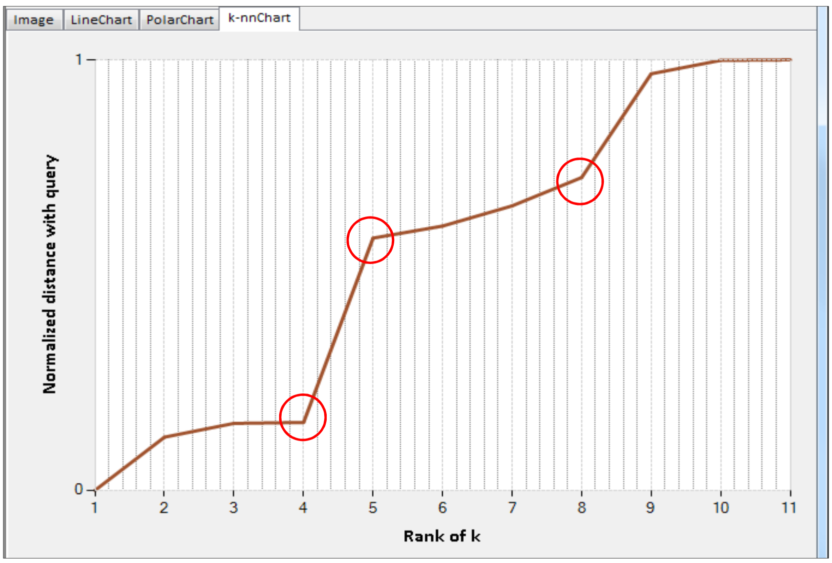


FIGURE 8. Scaling-invariant distances from a query to k-NN data time-series

Finally, Figure 8 visualizes scaling-invariant distances from a query time-series to k data time-series retrieved as scaling-invariant similar ones. Using this graph, we can intuitively understand the relative change of scaling-invariant distances between query and data time-series. Also, we can use this graph to determine an optimal (or near optimal) k to identify rapid changing points of the distances. For example, in Figure 8, we can choose 4, 5, or 8 as the k value since there are rapid distance changes in those three values compared with other values.

4. **Conclusions and Future Work.** In this paper, we have designed and implemented a visualization platform for scaling-invariant boundary image matching. The platform provides four visualization types: image comparison frame, line chart, polar chart, and k-NN distance graph. Using these various visualization tools, we can easily recognize the scaling-invariant matching result, and we can also intuitively understand why and how such results are produced. In particular, as we can see in visualizing the scaling-invariant images, the proposed visualization platform efficiently solves the scaling-invariant problem of naive boundary image matching. As the future work, we will consider the visualization of symmetric-invariant boundary image matching, and we will try to apply the DTW distance [2] to rotation-, scaling-, or symmetric-invariant boundary image matching.

Acknowledgment. This work was supported by Institute for Information & Communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (No. R7117-16-0214, Development of an Intelligent Sampling and Filtering Techniques for Purifying Data Streams).

REFERENCES

[1] R. Agrawal, C. Faloutsos and A. Swami, Efficient similarity search in sequence databases, *Proc. of the 4th International Conference on Foundations of Data Organization and Algorithms*, Chicago, IL, pp.69-84, 1993.

[2] K. P. Chan, A. W. C. Fu and C. T. Yu, Haar wavelets for efficient similarity search of time-series: With and without time warping, *IEEE Trans. Knowledge and Data Engineering*, vol.15, no.3, pp.686-705, 2003.

[3] B.-S. Kim, Y.-S. Moon, M.-J. Choi and J. Kim, Interactive noise-controlled boundary image matching using the time-series moving average transform, *Multimedia Tools and Applications*, vol.72, no.3, pp.2543-2571, 2014.

[4] S. Lee, B.-S. Kim, M.-J. Choi and Y.-S. Moon, An approximate multi-step k-NN search in time-series databases, *Advances in Computer Science and Its Applications*, vol.279, pp.173-178, 2013.

[5] S.-J. Lee, J.-S. Lee, H. Cho and W.-S. Han, A visualization tool for ranked subsequence matching in time-series databases, *Journal of KIISE: Databases*, vol.38, no.2, pp.92-103, 2011.

[6] S. Moon, S. Lee, B.-S. Kim and Y.-S. Moon, Visualization tool for boundary image matching based on time-series data, *Proc. of DASFAA Workshops*, Hanoi, Vietnam, pp.286-292, 2015.

[7] Y.-S. Moon and W.-K. Loh, Triangular inequality-based rotation-invariant boundary image matching for smart devices, *Multimedia Systems*, vol.21, no.1, pp.15-28, 2015.

[8] S. Moon, S. Lee, B.-S. Kim and Y.-S. Moon, A visualization tool for rotation-invariant boundary image matching, *ICIC Express Letters*, vol.10, no.2, pp.309-315, 2016.

[9] Y.-S. Moon, B.-S. Kim, M. S. Kim and K.-Y. Whang, Scaling-invariant boundary image matching using time-series matching techniques, *Data and Knowledge Engineering*, vol.69, no.10, pp.1022-1042, 2010.

[10] T. Rakthanmanon et al., Searching and mining trillions of time series subsequences under dynamic time warping, *Proc. of the 18th International Conference on Knowledge Discovery and Data Mining*, Beijing, China, pp.262-270, 2012.

[11] M. Vlachos, Z. Vagena, P. S. Yu and V. Athitsos, Rotation invariant indexing of shapes and line drawings, *Proc. of ACM Conference on Information and Knowledge Management*, Bremen, Germany, pp.131-138, 2005.

[12] M. Weber, M. Alexa and W. Muller, Visualizing time-series on spirals, *Proc. of International Conference on IEEE Symposium on Information Visualization*, San Diego, CA, pp.7-13, 2001.

[13] *Wikipedia*, http://en.wikipedia.org/wiki/Visualization.