ViSTA: Visualisation of Scanpath Trend Analysis (STA)

Şükrü Eraslan, Serkan Karabulut, Mehmet Can Atalay, Yeliz Yeşilada

Middle East Technical University Northern Cyprus Campus,
99738 Kalkanlı, Güzelyurt, Mersin 10 Turkey
{seraslan,karabulut.serkan,atalay.can,yyeliz}@metu.edu.tr

Abstract. Eye tracking plays a key role in understanding user behaviours and usability studies. We have previously proposed an algorithm called STA (Scanpath Trend Analysis) that analyses multiple individual scanpaths on a web page to discover their trending path in terms of the areas of interest (AOIs) of the page. It provides a good understanding of how users interact with web pages in general. Our extensive previous work shows that this algorithm provides the most representative path of multiple users as its result is more similar to individual scanpaths in comparison with the results of other algorithms. However, its current implementation has no graphical user interface and provides a sequence of characters that represent AOIs. Some platform and modules should also be installed in advance to run it. To address these limitations, this paper presents a web-based visualisation tool for the STA algorithm called ViSTA. This tool allows researchers and practitioners to visualise individual scanpaths on a particular web page with gaze plots, visually draw AOIs, apply the STA algorithm, and visualise the result of the algorithm with an AOI graph. Our user evaluation shows that the workload is lower with the ViSTA tool compared to the current implementation of the STA algorithm.

Keywords: Eye Tracking, Areas of Interest, Web Pages, Trending Path, Web-based, User Interface

Scanpath Trend Analysis (STA)'in Görselleştirilmesi

Özet. Göz izleme teknolojisi insan davranışlarını anlama ve kullanılabilirlik çalışmalarında etkin bir rol oynamaktadır. Yaptığımız önceki çalışmalarda STA (Scanpath Trend Analysis) algoritmasını önerdik. Bu algoritma bir sayfa üzerindeki birçok kişiye ait tarama güzergahını analiz ederek, sayfa üzerindeki ilgi alanları cinsinden popüler olan tarama güzergahını çıkarıyor. Bu güzergah kullanıcıların ilgili web sayfası ile genel olarak nasıl etkileşim kurduğunu anlamayı sağlıyor. Önceki çalışmalarımız gösteriyor ki literatürdeki benzer algoritmalar ile karşılaştırıldığı zaman, STA algoritmasının sonucu bireysel tarama güzergahlarına daha çok benzerlik gösteriyor. Fakat, STA algoritmasının görsel bir ara yüzü yoktur ve sonucu ise her karakteri sayfa üzerindeki belirli bir alanı ifade eden alfanumerik bir karakter dizisi şeklindedir. Bu uygulamanın çalıştırılabilmesi için ise bazı platform ve modüllerin önceden yüklenmesi gerekmektedir. Bu bildiri, STA algoritmasının var olan uygulamasının sınırlamalarını ortadan kaldıran web tabanlı görsel bir uygulama sunmaktadır. Bu görsel uygulama ile araştırmacılar ve pratisyenler bireysel tarama güzergahlarını bakış grafiği isimli teknik ile görselleştirebilecek, görsel olarak sayfa üzerindeki ilgi alanlarını belirleyebilecek, STA algoritmasını uygulayabilecek, ve bu algoritmanın sonucunu da yine ilgi alanları haritası isimli teknik ile görselleştirilebilecektir. Yaptığımız kullanıcı değerlendirmeleri bu görsel uygulamanın STA algoritmasının uygulanması için gerekli iş yoğunluğunu azalttığını göstermektedir.

Anahtar Kelimeler: Göz İzleme, İlgi Alanları, Web Sayfaları, Popüler Güzergah, Web Tabanlı, Kullanıcı Ara Yüzü

1 Introduction

Eye tracking has been commonly used for better understanding of how users interact with web pages, especially which areas are frequently used and which paths are usually followed in terms of these areas to complete certain tasks. Based on the analysis of eye tracking data, web pages can be further processed to support users in constrained environments for completing their tasks. In particular, web pages can be re-engineered for these users to allow them to directly access the most frequently used areas without being distracted by other inappropriate areas. This is especially useful for visually disabled users who access the web with their screen readers and have to listen clutter [13].

While users are interacting with web pages, their eyes make quick movements called saccades. Their eyes also make fixations on certain points. The series of saccades and fixations show their scanpaths on these pages. There are different techniques to visualise individual scanpaths [2] (Section 2). The most popular technique for scanpath visualisation is a gaze plot that represents saccades and fixations with straight lines and circles respectively. The radius of a circle is directly proportional to the duration of a fixation and each circle is numbered to show the sequence. Figure 1 shows an example of a gaze plot on the home page of the Babylon website.

To conduct scanpath analysis on the web, web pages are typically divided into their areas that can interest or attract users (AOIs) and then fixations are represented with their corresponding AOIs. For example, Figure 1 shows how the home page of the Babylon website is automatically divided into its AOIs with the extended Vision-based Page Segmentation (VIPS) algorithm [1]. If a user looks at the AOIs M, H and H respectively, then his/her path will be represented as MHH. After representing scanpaths in terms of AOIs, they are further analysed for different purposes, especially calculating a similarity score between two scanpaths, computing transition probabilities between AOIs, detecting patterns in scanpaths, or discovering a single representative scanpath of multiple users [4].

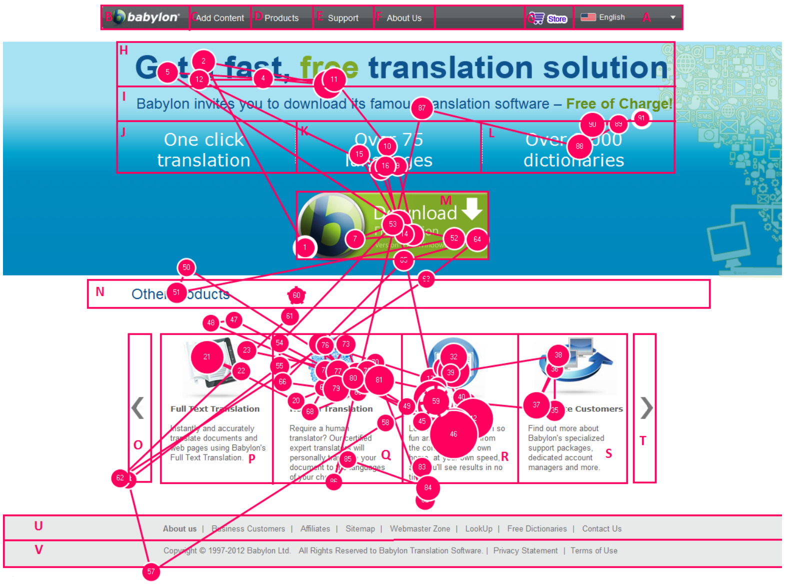

Figure 1. A gaze plot on the home page of the Babylon website segmented by the extended VIPS algorithm [1]

Even though there are different algorithms which aim to detect patterns in multiple scanpaths or discover a representative scanpath for a group of scanpaths, their results typically have low similarities to individual scanpaths [4]. Compared to these algorithms, Scanpath Trend Analysis (STA) discovers the most representative path of multiple users on a web page as its result is the most similar to individual scanpaths [5] (Section 2). However, the current implementation of the STA algorithm[1] has no visual user interface and its output is only a sequence of alphanumeric characters that represent AOIs (e.g., ABACD). This algorithm is novel in this field and is being increasingly used. To understand and interpret the output, researchers and practitioners should remember which alphanumeric character corresponds to which AOI. If a visual user interface is developed for the STA algorithm and its output is also visualised on a corresponding page, then we eliminate the burden of remembering and understanding the sequence of letters. Furthermore, some platform and modules are also needed to be installed in advance to run the current implementation.

To address the limitations of the current implementation of the STA algorithm and make it more usable and functional, this paper presents the first web-based visualisation tool for this algorithm called Visualisation of Scanpath Trend Anlaysis or shortly ViSTA (Section 3). This tool does not need any specific platform and modules installed. It allows users to upload an eye tracking dataset and select one of the web pages included in the dataset for further analysis. Since gaze plots are commonly used to visualise scanpaths, this tool visualises the in-

[1] STA's current implementation: `https://github.com/SukruEraslan/sta`

dividual scanpaths on the selected page by using gaze plots [2]. It then allows users to visually draw AOIs on the page and apply the STA algorithm for discovering the trending path based on the drawn AOIs. The trending path is then visualised by using an AOI graph. Since an AOI graph shows both AOIs and transitions between these AOIs, it would be easier to recognise which AOIs are trending among users and in which order these AOIs are visited.

The ViSTA tool can be used by eye tracking researchers, usability evaluators, data scientists and psychologists. Our user evaluation shows that the STA algorithm can be used with less workload with this tool (Section 4). Although it is mainly designed for visualising the STA algorithm, other such algorithms can also be integrated into this tool. Similarly, other visualisation techniques can also be integrated so that the outputs of the algorithms can be visualised in different ways (Sections 5 and 6).

2 Related Work

A large number of techniques have been proposed in the literature to visualise eye tracking data on 2D and 3D visual stimuli in context or not in context [2]. Heat maps and gaze plots have been commonly used in eye tracking research for visualisation. Heat maps use spatial information of eye tracking data, specifically x, y and possibly z coordinates of fixations. These maps use different colours to differentiate which areas are commonly used and which areas are rarely used by users. For example, Tobii Studio Software [11] generates heat maps which show the commonly used areas with the red colour and the rarely used areas with the green colour. These maps can be generated based on different features, such as fixation counts or fixation durations. Although the most commonly used areas can easily be recognised with heat maps, these maps do not take sequential information into consideration. Specifically, heat maps do not allow to determine in which order these areas are used. Even though gaze plots use both spatial and sequential information as illustrated in Figure 1, they will overlap each other when there are many scanpaths. Therefore, the visualisation of many scanpaths with gaze plots at the same time would be a problem because it would be difficult to analyse them. There are also some visualisation techniques which are designed to visualise eye tracking data based on given AOIs [2]. In particular, each AOI can be used as a node (represented with circles) in a graph and directed edges (represented with lines) between these nodes are used to illustrate transitions between AOIs [2]. The sizes of nodes can be determined based on different features, such as fixation count or dwell time [2]. The thickness of edges can also be determined based on different features, such as the number of transitions [8, 2]. This visualisation technique is called an AOI graph [2].

Even though some of these visualisation techniques have a web-based interface (such as, [7]), most of them do not have a web-based interface which would allow researchers and practitioners to directly access and use them in their studies without installing extra platform and modules on their computers. There are also different algorithms available to process and analyse eye tracking data, especially scanpaths [4]. However, the results of these algorithms are typically not visualised with any visualisation technique on the web platform, with some exceptions [12, 3]. For example, the ScanGraph tool[2] finds similarities between individual scanpaths based on one of three methods (Levenshtein, Needleman-Wunsch and Damerau-Levenshtein) and produces a graph where similar scanpaths are connected to each other [3].

To the best of our knowledge, none of the algorithms which discover a representative scanpath for multiple scanpaths has a tool to visualise their outputs on the web platform. This paper presents a web-based visualisation tool for the STA algorithm which is able to identify the most representative path for multiple users [5]. The STA algorithm is a multi-pass algorithm with three main stages which are responsible for preparing individual scanpaths, identifying trending AOIs and constructing the trending path with the trending AOIs respectively [5]. The first stage represents each fixation with its corresponding AOI and each individual scanpath is represented a series of AOIs. The second stage identifies trending AOIs to be used for the construction of the trending path. If a particular AOI gets at least the same attention as the fully shared AOIs in terms of the total fixation duration and the total number of fixations, it is considered as trending AOI. The last stage constructs the trending scanpath by combining the trending AOIs based on their overall positions in the individual scanpaths. The full description of the STA algorithm can be found in [5].

3 ViSTA

The ViSTA tool[3] can be used to upload an eye tracking dataset, visualise individual scanpaths with gaze plots, visually draw AOIs on stimuli, apply the STA algorithm to discover the trending scanpath and then visualise the trending path with an AOI graph. As illustrated in Figure 2, the ViSTA tool is comprised of three main modules explained below which are as follows: (1) Data Processing, (2) Algorithmic Processing and (3) Visualisation Processing. This tool was mainly developed by using jQuery which is a JavaScript library. The current implementation of the STA algorithm was also slightly modified by using Flask[4] to run it as a server which accepts POST requests from the ViSTA tool.

3.1 Data Processing

The first module is responsible from taking eye tracking data and converting this data into an internal storage format for further analysis. There should be a different data file for each participant that contains the details of their fixations (index, timestamp, duration, x and y coordinates and stimuli name). These data files should include web page links as stimuli name since the ViSTA tool is currently designed for scanpath analysis on the web. This tool accesses each page

[2] ScanGraph: `www.eyetracking.upol.cz/scangraph`
[3] ViSTA: `http://iam.ncc.metu.edu.tr/vistatool/demo/index.html`
[4] Micro web framework for Python: `http://flask.pocoo.org/`

by using their links to retrieve some information, especially their titles. When the data files are uploaded and they are converted into an internal storage format, this module becomes ready to send the data to the next module for further processing.

In this tool, two algorithms are currently available: (i) the gaze plot algorithm which further processes the data to visualise individual scanpaths with gaze plots; and (ii) the STA algorithm which further processes the data to identify the trending scanpath. Since the ViSTA tool firstly visualises individual scanpaths with gaze plots, the algorithm parameter is set to the gaze plot algorithm internally here so that individual scanpaths are visualised with gaze plots in the first round.

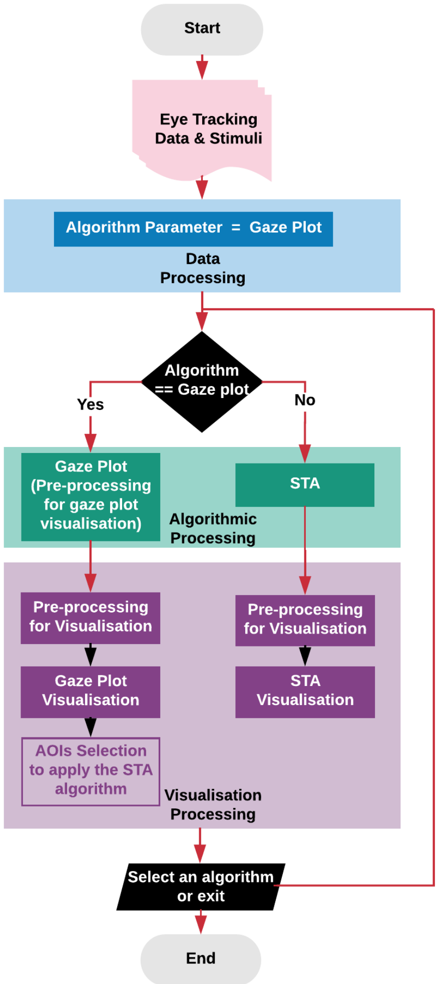

Figure 2. The work flow of the ViSTA tool [Selection of AOIs is needed when the STA algorithm will be applied.]

3.2 Algorithmic Processing

When the Algorithmic Processing module takes the data, it further processes the data based on the setted algorithm parameter. If the algorithm parameter is set to the gaze plot algorithm, this module further processes the data to be able to visualise individual scanpaths with gaze plots. However, if the algorithm parameter is set to the STA algorithm, this module further process the data to discover the trending path. Therefore, the algorithm processing modules either prepares the data to be visualised with gaze plots or applies the STA algorithm and prepares the result to be visualised with an AOI graph based on the algorithm parameter.

3.3 Visualisation Processing

The Visualisation Processing module takes the results of the algorithms and visualises them with their visualisation techniques by using vis.js[5]. In particular,

[5] Dynamic, browser based visualisation library: `http://visjs.org/`

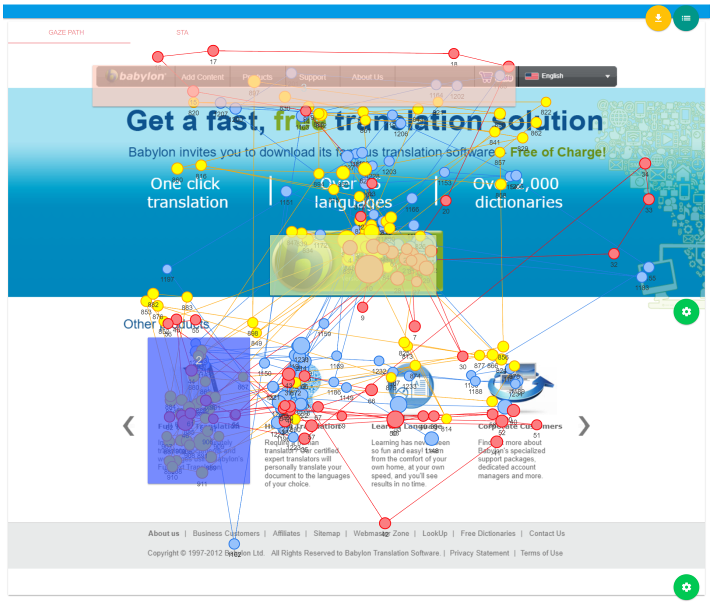

Figure 3. Identification of AOIs on the home page of the Babylon website with gaze plots

when the results of the gaze plot algorithm are received, researchers and practitioners should firstly select which participants' scanpaths on which web pages will be visualised with gaze plots. By default, all the participants are selected. When the participants and the web page are selected, this tool takes a screen shot of the page with html2canvas[6] by using its link and the individual scanpaths are visualised on the page with gaze plots. Figure 3 shows how the ViSTA tool visualises individual scanpaths on the home page of the Babylon website with gaze plots. Researchers and practitioners can then visually draw AOIs to be used with the STA algorithm. If no AOI is drawn, the STA algorithm can not be selected for further processing the data because this algorithm discovers the trending scanpath in terms of the AOIs. As an example, Figure 3 also shows three AOIs identified by a user on the home page of the Babylon website.

When the STA algorithm is selected after the visualisation of individual scanpaths with gaze plots and the result of the STA algorithm is received, this module firstly pre-processes the result for visualisation, and then visualises it by using an AOI graph. An example of this visualisation is shown in Figure 4. Since the AOI that includes the link to download the free version of Babylon is the most frequent AOI in the trending path, its node is larger than other two AOIs' nodes.

[6] Screenshots with JavaScript: `https://html2canvas.hertzen.com/`

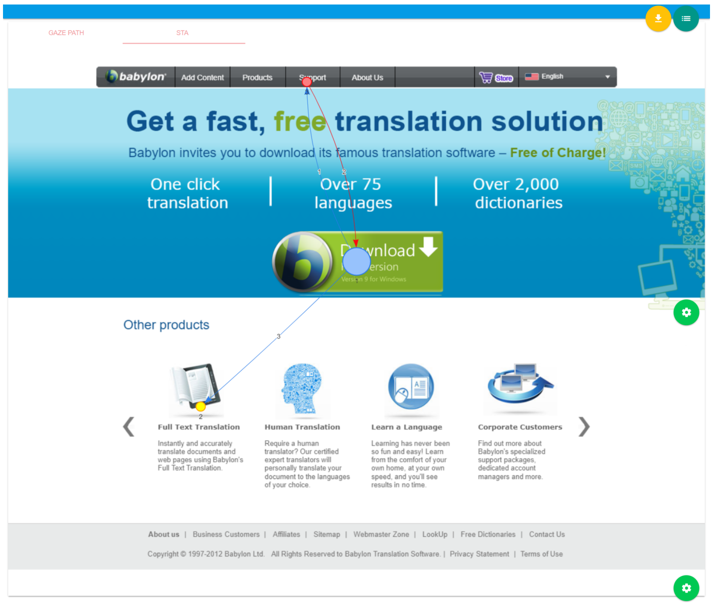

Figure 4. Visualisation of the trending scanpath discover by the STA algorithm on the home page of the Babylon website

4 Evaluation

In order to assess the effectiveness of the VISTA tool, a user study was conducted with NASA Task Load Index (TLX)[7]. This metric is widely used to perform workload assessments for different user interfaces and is composed of six attributes: Mental Demand, Physical Demand, Temporal Demand, Performance, Effort and Frustration. Each attribute is assessed by the participants by giving a value between 1 and 20. Higher values represent negative scores. In this study, our research question was "Does the VISTA tool have better TLX attributes in comparison with the current implementation of the STA algorithm?"

4.1 Methodology

A detailed description of the dataset used in the user study can be found in [5]. In this user study, we used a subset of that dataset. First of all, we randomly divided the dataset into two halves called dataset A and dataset B. We then asked our participants to apply the current implementation of the STA algorithm with one half of the dataset and the VISTA tool with the other half to identify their trending paths on a particular web page. A half of the participants used the current implementation of the algorithm and then the VISTA tool whereas another half used the VISTA tool and then the current implementation of the algorithm. Besides this, a half of the participants applied the VISTA tool with the

[7] NASA TLX: `https://humansystems.arc.nasa.gov/groups/tlx/`

dataset A and the current implementation of the algorithm with the dataset B whereas another half applied the VISTA tool with the dataset B and the current implementation of the algorithm with the dataset A. Therefore, we ensured that all the cases were counterbalanced to deal with any possible familiarity and order effects. When the participants identified the trending scanpath for a particular dataset, they were given a printed copy of the page to draw the trending path on the page and then they were asked to fill in the NASA TLX Form[8].

Participants: The study was conducted at Middle East Technical University Northern Cyprus Campus with four female and eight male students. The participants were between the age of 18-24, apart from one of them who was between the age of 25-34. All of the participants were daily web users. They were also asked to assess their computer skills by using a 5-point Likert scale. The mean value was 3.42 with the standard deviation 1.00.

Procedure: The participants firstly read the information sheet to understand the main objectives of the study and their rights and then they signed a consent form. After that, they were asked to complete a short questionnaire to collect their basic demographic information (gender, age groups, web usage, general computer skills, and departments). A training session was then given to them to illustrate how they can use the current implementation of the STA algorithm and the VISTA tool. This training session mainly used a sample dataset (not the one used in the evaluation). After the training session, they started their evaluation session.

Materials: The original eye tracking dataset includes six web pages with varying level of visual complexities. We used the home page of the Babylon website for the training sessions, and the home page of the Apple website for the evaluation sessions.

4.2 Results

Figure 5 shows the comparison of the current implementation of the STA algorithm and the VISTA tool based on the mean values of the NASA TLX attributes where the error bars illustrate the standard deviation. Besides this, Table 1 shows a detailed descriptive analysis of the NASA TLX attributes for the current implementation of the STA algorithm and the ViSTA tool.

The evaluation shows that the ViSTA tool needs less mental demand, physical demand and temporal demand in comparison with the current implementation of the STA algorithm. The participants also stated that they needed less effort to complete the given task with the ViSTA tool. Moreover, they were less frustrated when they used the ViSTA tool. However, they stated that they performed slightly better (better satisfied with their performance in achieving the task) with the current implementation of the STA algorithm. Based on the evaluation, we can suggest that the VISTA tool has a lower workload in terms of mental demand, physical demand, temporal demand, effort and frustration level compared to the current implementation of the STA algorithm.

[8] `https://humansystems.arc.nasa.gov/groups/TLX/downloads/TLXScale.pdf`

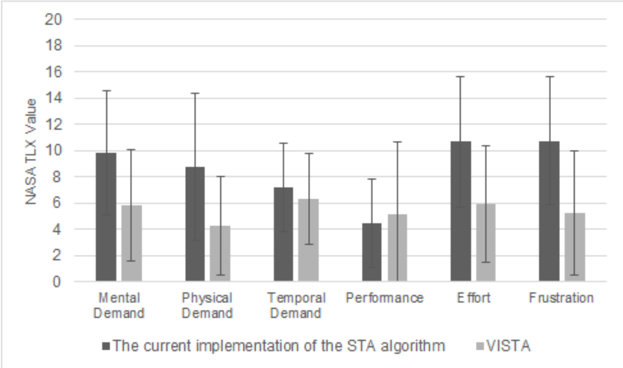

Figure 5. NASA TLX attributes for the current implementation of the STA algorithm and the ViSTA tool

Tablo 1. Mean, median, standard deviation, maximum and minimum values of the NASA TLX attributes for the current implementation of the STA algorithm and the ViSTA tool

	NASA TLX Attribute	Mean	Median	Std. dev.	Max	Min
The current implementation of the STA algorithm	**Mental Demand**	9.83	9.50	4.71	17	3
	Physical Demand	8.75	10.00	5.58	18	2
	Temporal Demand	7.17	8.00	3.35	12	2
	Performance	4.50	3.50	3.37	12	1
	Effort	10.67	12.50	4.98	17	2
	Frustration	10.75	10.50	4.90	18	2
ViSTA	**Mental Demand**	5.83	4.50	4.24	14	1
	Physical Demand	4.25	3.00	3.79	14	1
	Temporal Demand	6.33	5.50	3.47	12	1
	Performance	5.17	2.50	5.54	17	1
	Effort	5.92	4.50	4.44	15	1
	Frustration	5.25	3.50	4.73	17	1

5 Discussion

The ViSTA tool provides a web-based interface for the STA algorithm. It allows researchers and practitioners to directly access and use the STA algorithm for their studies without installing extra platform and modules on their computers. Similar to other algorithms, the STA algorithm can also be used by different researchers in different fields. Therefore, it should not be expected that all of these researchers easily manage to download and run their implementations on their computers. With the ViSTA tool, researchers and practitioners can easily upload their eye tracking datasets and visually draw their AOIs to apply the STA algorithm. When the trending path is generated with the algorithm, it is visualised with an AOI graph so that researchers can easily interpret the trending path without the need for remembering which alphanumeric character represents which AOI as they will see the trending AOIs and the order in which the AOIs are used. Our evaluation shows that the VISTA tool decreases the workload of using the STA algorithm. Specifically, the tool scores better in

TLX attributes in terms of mental demand, physical demand, temporal demand, effort and frustration level compared to the current implementation of the STA algorithm. We evaluated the ViSTA tool with 12 people who were mostly not experienced in eye tracking research and studies. In the future, we can conduct further studies with more, and more experienced users for detailed feedback.

The current version of the VISTA does not include automatic AOI detection based on fixation clusters or the source code of web pages. The tool allows researchers to visually draw their own AOIs on the fly. They may detect AOIs by using different techniques and then they can draw the AOIs in the VISTA tool. As the tool has an open architecture, it can be easily extended in the future to support automatic AOI detection. For example, the VIPS algorithm has been extended and implemented as a web service [1], so we can use this web service to automatically segment web pages into their areas based on their source code and visual representation. Appropriate clustering algorithms can also be implemented as part of this tool to allow the identification of AOIs by clustering fixations [10]. Another visualisation level can also be added to the ViSTA tool for AOIs which will be shared by other components of the tool.

The ViSTA tool currently has only the STA algorithm, apart from the generic algorithm to pre-process the data for visualising individual scanpaths with gaze plots. However, it has an open architecture which means that other algorithms can easily be integrated. For example, the SPAM algorithm has been used to identify sequential patterns in eye tracking data [6]. It can also be integrated into this tool and its output can also be visualised with an AOI graph. However, some studies need to be conducted.

We are still improving this tool by exploring appropriate algorithms and visualisation techniques to make this tool more functional and usable. In the future, we are planning to add different visualisation techniques to the Visualisation Module (see Figure 2), such as a time plot [9]. As a consequence, when a particular algorithm provides an output, a list of appropriate visualisation techniques can be listed to be selected for visualising the output. The current version of the ViSTA tool takes a screen shot of a web page by using its link for visualisation. However, we are planning to allow researchers and practitioners to directly upload their visual stimuli so that they can use this tool for other visual stimuli which are not on the web. The visualisation of the outputs of the algorithms can also be animated to make them more informative. For example, when the output of the STA algorithm is visualised with an AOI graph, its nodes and edges can be shown one at a time based on the time sequence which allows researchers to focus on a particular trending AOI at a time during analysis. The ViSTA tool can also be improved in a way that a particular algorithm can be applied to different groups of participants (at the moment, the STA algorithm is only applied to one group) and the results of these groups can be visualised on the same visual stimulus for a comparison purpose. Additionally, an account system can be created such that the users of the tool can have their workbenches. This would allow them to store their datasets and revisit their workbenches.

6 Conclusion

The main aim of this paper is to introduce the first web-based interface of the STA algorithm. The STA algorithm is novel in this field and it is being increasingly used in different studies. Its visualisation, called ViSTA, will allow researchers and practitioners to easily use the STA algorithm and understand its output directly without the need of remembering the names of AOIs and relating the AOIs to the output in their minds. Since there is a limited number of web-based tools for analysing eye tracking data, our tool also makes an useful contribution to eye tracking research. This tool is open to be further improved by integrating other algorithms and visualisation techniques.

References

1. Akpınar, M.E., Yeşilada, Y.: Vision Based Page Segmentation Algorithm: Extended and Perceived Success, pp. 238–252. Springer International Publishing, Cham (2013)
2. Blascheck, T., Kurzhals, K., Raschke, M., Burch, M., Weiskopf, D., Ertl, T.: Visualization of eye tracking data: A taxonomy and survey. Computer Graphics Forum 36(8), 260–284 (2017)
3. Dolezalova, J., Popelka, S.: Scangraph: A novel scanpath comparison method using visualisation of graph cliques. Journal of Eye Movement Research 9(4) (2016)
4. Eraslan, S., Yesilada, Y., Harper, S.: Eye tracking scanpath analysis techniques on web pages: A survey, evaluation and comparison. Journal of Eye Movement Research 9(1) (2015)
5. Eraslan, S., Yesilada, Y., Harper, S.: Scanpath trend analysis on web pages: Clustering eye tracking scanpaths. ACM Trans. Web 10(4), 20:1–20:35 (Nov 2016)
6. Hejmady, P., Narayanan, N.H.: Visual attention patterns during program debugging with an ide. In: Proceedings of the 2012 Symposium on ETRA. pp. 197–200. ETRA '12, ACM, New York, NY, USA (2012)
7. Herman, L., Popelka, S., Hejlova, V.: Eye-tracking analysis of interactive 3d geovisualization. Journal of Eye Movement Research 10(3) (2017)
8. Holmqvist, K., Holsanova, J., Barthelson, M., Lundqvist, D.: Reading or scanning? A study of newspaper and net paper reading., pp. 657–670. The mind's eye: cognitive and applied aspects of eye movement research, Elsevier (2003)
9. Räihä, K.J., Aula, A., Majaranta, P., Rantala, H., Koivunen, K.: Static Visualization of Temporal Eye-Tracking Data, pp. 946–949. Springer Berlin Heidelberg, Berlin, Heidelberg (2005)
10. Santella, A., DeCarlo, D.: Robust clustering of eye movement recordings for quantification of visual interest. In: Proceedings of the 2004 Symposium on ETRA. pp. 27–34. ETRA '04, ACM, New York, NY, USA (2004)
11. Tobii Technology AB: Tobii Studio^{TM} 2.X User Manual (Sep. 2010). Tobii Technology AB (2010)
12. Topić, G., Yamaya, A., Aizawa, A., Martínez-Gómez, P.: Fixfix: Fixing the fixations. In: Proceedings of the Ninth Biennial ACM Symposium on ETRA. pp. 319–320. ETRA '16, ACM, New York, NY, USA (2016)
13. Yesilada, Y., Harper, S., Eraslan, S.: Experiential transcoding: An eyetracking approach. In: Proceedings of the 10th International Cross-Disciplinary Conference on Web Accessibility. pp. 30:1–30:4. W4A '13, ACM, New York, NY, USA (2013)