Scanpath Trend Analysis on Web Pages: Clustering Eye Tracking Scanpaths

DOI:
[10.1145/2970818](https://doi.org/10.1145/2970818)

Document Version
Accepted author manuscript

[Link to publication record in Manchester Research Explorer](#)

Citation for published version (APA):
Eraslan, S., Yesilada, Y., & Harper, S. (2016). Scanpath Trend Analysis on Web Pages: Clustering Eye Tracking Scanpaths. *ACM Transactions on the Web*. <https://doi.org/10.1145/2970818>

Published in:
ACM Transactions on the Web

Citing this paper
Please note that where the full-text provided on Manchester Research Explorer is the Author Accepted Manuscript or Proof version this may differ from the final Published version. If citing, it is advised that you check and use the publisher's definitive version.

General rights
Copyright and moral rights for the publications made accessible in the Research Explorer are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

Takedown policy
If you believe that this document breaches copyright please refer to the University of Manchester's Takedown Procedures [<http://man.ac.uk/04Y6Bo>] or contact uml.scholarlycommunications@manchester.ac.uk providing relevant details, so we can investigate your claim.

Scanpath Trend Analysis on Web Pages: Clustering Eye Tracking Scanpaths

SUKRU ERASLAN, University of Manchester and Middle East Technical University Northern Cyprus Campus

YELIZ YESILADA, Middle East Technical University Northern Cyprus Campus

SIMON HARPER, University of Manchester

Eye tracking studies have widely been used in improving the design and usability of web pages, and in the research of understanding how users navigate them. However, there is limited research in clustering users' eye movement sequences (i.e., scanpaths) on web pages to identify a general direction they follow. Existing research tends to be reductionist which means the resulting path is so short that it is not useful. Moreover, there is little work on correlating users' scanpaths with visual elements of web pages and the underlying source code which means the result cannot be used for further processing. In order to address these limitations, we introduce a new concept in clustering scanpaths called *Scanpath Trend Analysis (STA)* which does not only consider the visual elements visited by all users, but also considers the visual elements visited by the majority in any order. We present an algorithm which automatically does this trend analysis to identify a trending scanpath of multiple web users in terms of visual elements of a web page. In contrast to existing research, the STA algorithm first analyses mostly visited visual elements in given scanpaths, then clusters the scanpaths by using these visual elements based on their overall positions in the individual scanpaths, and then constructs a trending scanpath in terms of these visual elements. This algorithm was experimentally evaluated by an eye tracking study on six web pages for two different kinds of tasks, and therefore 12 cases in total. Our experimental results show that the STA algorithm generates a trending scanpath that addresses the reductionist problem of existing work by preventing the loss of commonly visited visual elements for all the cases. Based on the statistical tests, the STA algorithm also generates a trending scanpath which is significantly more similar to the inputted scanpaths compared to other existing work in 10 out of 12 cases. In the rest of the cases, the STA algorithm still performs significantly better than some of other existing work. This algorithm contributes to behaviour analysis research on the web which can be used for different purposes, for example re-engineering web pages guided by the trending scanpath to improve users' experience or guiding designers to improve their design.

CCS Concepts: •**Human-centered computing** → **Human computer interaction (HCI)**; User models; User studies; Usability testing; Laboratory experiments; •**Information systems** → **World Wide Web**; Web interfaces;

General Terms: Design, Algorithms

Additional Key Words and Phrases: Eye Tracking, Scanpath, Trend Analysis, Clustering, Algorithm

ACM Reference Format:

Sukru Eraslan, Yeliz Yesilada and Simon Harper, 2015. Scanpath Trend Analysis on Web Pages: Clustering Eye Tracking Scanpaths *ACM Trans. Web* V, N, Article A (January YYYY), 36 pages.

DOI: <http://dx.doi.org/10.1145/0000000.0000000>

Authors' addresses: S. Eraslan (corresponding author), School of Computer Science, University of Manchester, Kilburn Building, Oxford Road, Manchester, M13 9PL, United Kingdom, email: eraslans@cs.man.ac.uk and Middle East Technical University, Northern Cyprus Campus, 99738 Kalkanlı, Güzelyurt, Mersin 10, Turkey, email: seraslan@metu.edu.tr; Y. Yesilada, Middle East Technical University, Northern Cyprus Campus, 99738 Kalkanlı, Güzelyurt, Mersin 10, Turkey, email: [yyeliz@metu.edu.tr](mailto:yeliz@metu.edu.tr); S. Harper, School of Computer Science, University of Manchester, Kilburn Building, Oxford Road, Manchester, M13 9PL, United Kingdom, email: simon.harper@manchester.ac.uk.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© YYYY ACM. 1559-1131/YYYY/01-ARTA \$15.00

DOI: <http://dx.doi.org/10.1145/0000000.0000000>

1. INTRODUCTION

Eye tracking studies have commonly been used in improving the design of web pages and evaluating their usability. These studies have also been used in the research of understanding how users interact with web pages [Albanesi et al. 2011; Yesilada et al. 2008]. While users are reading web pages, their eyes make fixations where the eyes become relatively stable. Between fixations, their eyes make quick movements called saccades and the sequences of fixations and saccades define their scanpaths on web pages. Figure 1 shows an example of a web page which is segmented into its visual elements [Akpınar and Yeşilada 2013] and an example of a scanpath (a sequence of saccade-fixation-saccade) on the web page. The user fixations are represented by circles where the larger circles illustrate the longer fixations and the numbers show the sequence. In addition, the saccades are represented by lines.

Fig. 1: A scanpath example

Eye tracking data, particularly scanpaths, can provide us a deep understanding of how people use and interact with web pages. However, in order to be able to process web pages by using scanpath data, we need to bring multiple scanpaths together to identify a general direction followed by users in terms of visual elements of a web page. Otherwise, we only have individual scanpaths which are specific to individual users, and therefore any processing done with those scanpaths will be specific to individuals and will not be representative of multiple users. In the literature, some algorithms exist to address this problem. When we investigate these algorithms, we recognise that they tend to be reductionist¹ which means the generated scanpath is likely to be unacceptably short for further processing. In this context, we classify an algorithm as reductionist if it is likely to lose a shared visual element due to its position in individual scanpaths or it is not tolerant of small deviations in individual scanpaths (especially, directly eliminating the visual element which is visited by most of users) [Eraslan et al. 2016a]. Moreover, these algorithms typically provide a scanpath which is not similar to the inputted scanpaths. Finally, there is little work on correlating users' scanpaths with visual elements of web pages and the underlying source code which is crucial for being able to process pages based on these scanpaths (See Section 3). To address these limitations, we introduce a new concept in clustering scanpaths called *Scanpath Trend Analysis (STA)* which does not only consider the visual elements visited by all users, but also considers the visual elements visited by the majority in any order. This analysis provides a trending scanpath² of multiple users in terms of visual elements of a web page. The trending scanpath term here is referred to as the most commonly followed scanpath which is developed by multiple users [Eraslan et al. 2016b]. The STA also differs from other existing clustering approaches in the context of sequential data clustering, such as discovering interesting places by analysing multiple sequences [Palma et al. 2008], identifying elementary parts within a single sequence [Leiva and Vidal 2013], discovering common sub-sequences by analysing multiple sequences [Lee et al. 2007; Anagnostopoulos et al. 2006]. In this paper, we present an algorithm which automatically conducts this trend analysis to identify a trending scanpath. The trending scanpath preserves the overall positions of the visual elements in the users' scanpaths, thus it deals with the sequentiality of the data.

In STA, we first segment a web page into its visual elements and label them with alphanumeric characters. There are many approaches for segmenting web pages into a set of coherent blocks of visual elements. Any of these existing approaches can be plugged into our algorithm but in this paper we use the improved and extended version of the Vision Based Page Segmentation (VIPS) algorithm. The VIPS algorithm automatically discovers visual elements of web pages and relates them to the underlying source code (See Section 3).

Once the visual elements are discovered, the STA algorithm relates eye tracking data with these visual elements to prepare individual scanpaths in terms of the visual elements. These individual scanpaths are then clustered to construct a trending scanpath. For clustering, the STA algorithm first identifies mostly visited visual elements in the individual scanpaths as trending visual elements by using their occurrence fre-

¹Cambridge Dictionary defines *Reductionism* as “the practice of considering or presenting something complicated in a simple way, especially a way that is too simple” (<http://dictionary.cambridge.org/dictionary/english/reductionism/>). Similar to this definition, we refer to the reductionism term as the practice of an algorithm which oversimplifies complicated scanpaths.

²Cambridge Dictionary defines *Trend* as “a general development or change in a situation or in the way that people are behaving” (<http://dictionary.cambridge.org/dictionary/english/trend/>). In this context, the most commonly followed scanpath is developed by multiple users and it is referred to as a trending scanpath. Even though it does not capture any change over time, it still shows the popularity among the users at a particular time.

quencies and durations. These visual elements are then located in the trending scanpath based on their overall positions in the individual scanpaths (See Section 4).

In order to investigate the validity of the STA algorithm, we conducted an eye tracking study with 40 participants at two different universities in two different countries. In this eye tracking study, we asked users to perform both browsing and searching tasks (See Section 5). With this eye tracking study, we investigated mainly three hypotheses on six web pages for two different kinds of tasks (browsing and searching), and therefore 12 cases in total (See Section 6).

With the searching tasks, it was very clear that certain visual elements had to be visited in a particular order to complete the task. Therefore, our first hypothesis investigated whether or not the STA algorithm provides a trending scanpath that includes the visited visual elements in the expected order. Our results show that on all the web pages used in the study, all the trending scanpaths identified by the STA algorithm include the visited visual elements in the expected order for all of the cases (See Section 6.1).

In our previous work, we showed that most of existing algorithms do not include the visual elements that are visited by all participants in a different order (See Section 3). However, when a particular visual element is visited by all participants, it means that the visual element has important potential for processing [Poole and Ball 2005] so that it should not be ignored just because of its positions in the individual scanpaths. Therefore, our second hypothesis investigated whether the STA algorithm includes all of the visited visual elements by all participants in the resulting trending scanpath or not. Our results show that for all the pages used in the study, the STA algorithm identifies a trending scanpath that includes all the visual elements visited by all the participants for all of the cases (See Section 6.2).

The trending scanpath is also expected to be as similar as possible to the given scanpaths. Therefore, the last hypothesis investigated whether the STA algorithm identifies a trending scanpath which is the most similar to the given scanpaths compared to other existing algorithms in the literature. Our statistical tests show that when we compare the STA algorithm to others, the generated trending scanpath is significantly more similar to the individual scanpaths in 10 out of 12 cases ($p < .0001$). In the rest of the cases, the STA algorithm still performs significantly better than some of other existing algorithms (See Section 6.3).

Finally, even though this eye tracking study shows the validity of the STA algorithm via these hypotheses, it also highlights a number of areas where further studies can be conducted. For example, in this study we had 40 individual scanpaths to identify a trending scanpath, but we do not know if a smaller set of scanpaths could also be used to identify the same trending scanpath. Similarly, in our studies we have used the VIPS algorithm to automatically generate the visual elements, however we do not know what would be the effect of using other segmentation techniques on the STA algorithm. We are planning to conduct further studies to investigate these areas (See Section 7).

2. CONTRIBUTIONS

This study contributes to Eye Tracking research on the web, Human Centred Web and Web Science fields. Moreover, it makes a contribution to the field of Web Accessibility by helping to improve the web experience of users in constrained environments such as small screen device users and visually disabled users. When users access the web by using their small screen devices, they are likely to spend more time and effort to complete their tasks [Ahmadi and Kong 2012]. In addition, visually disabled users typically access the web with their screen readers that follow the source code of web pages. Hence, they have to listen to unnecessary things to reach their targets [Ahmed

et al. 2012]. As trending scanpaths are representative of individual scanpaths, they can be used to re-engineer web pages based on not only the content but also understanding of users' experiences, called "*Experiential Transcoding*", to make them more accessible for users in such constrained environments [Akpınar and Yeşilada 2015; Yeşilada et al. 2013; Brown et al. 2010]. In particular, the visual elements that are not included in the trending scanpath can be removed and a link can be provided to access those visual elements, and/or the visual elements can be re-ordered based on the trending scanpath. For example, if the element J is not fixated by users on the Apple page, it can be removed from the web page by providing a link to access the element. As the size of the web page decreases, the page can be retrieved in a shorter time. This can be useful for small screen device users as these devices typically have limited capacity. Visually disabled users can also benefit from this approach as they do not need to listen the content of the element. Furthermore, if the element F is firstly fixated by users on the Apple page, it can be presented before other elements. This allows small screen device users to directly access the element without a lot of scrolling/zooming. By using this approach, visually disabled users can also directly access the element without listening unnecessary things. The STA algorithm can also be beneficial for web designers for understanding how users interact with web pages. For example, it can be used for usability studies to test whether or not the expected path is followed by users for a particular goal [Albanesi et al. 2011].

3. RELATED WORK

A number of algorithms and visualisation techniques have been introduced to analyse eye tracking scanpaths for different purposes. The visualisation techniques are typically designed for analysing scanpaths in an exploratory and qualitative way [Räihä et al. 2005; Blascheck et al. 2014] whereas the algorithms are mainly designed for comparing a pair of scanpaths quantitatively, computing transition probabilities between web page elements, detecting patterns within scanpaths and identifying a general scanpath for multiple scanpaths [Eraslan et al. 2016a; Holmqvist et al. 2011]. Although some of these algorithms analyse these scanpaths as geometric figures [Jarodzka et al. 2010; Mathôt et al. 2012; Le Meur and Baccino 2013], most of them analyse scanpaths which are represented as a sequence of Areas of Interest (AoIs) [Sutcliffe and Namoun 2012; Le Meur and Baccino 2013; Eraslan et al. 2014]. For example, if a user visits the Apple web page, shown in Figure 1, and fixates the AoI F, then the AoI C, then the AoI E and then the AoI B, the user's scanpath is represented as FCEB.

There are different approaches to generate these AoIs, ranges from using a grid layout directly to analysing fixation distribution over web pages [Privitera and Stark 2000; Santella and DeCarlo 2004; Sutcliffe and Namoun 2012]. However, these approaches typically do not correlate these AoIs with the underlying source code of web pages, and therefore scanpaths cannot be used for further processing of web pages. In order to address this, we use the improved and extended version of the VIPS algorithm which automatically segments web pages into their rectangular visual elements (i.e., AoIs) and relates these visual elements to the underlying source code [Akpınar and Yeşilada 2013]. It creates a tree of visual elements as the result of segmentation process. This tree structure is not the same as a DOM tree, but depends on it. There are more and smaller visual elements in the deeper or higher granularity levels of the tree. Each visual element corresponds to a DOM element and we assign the XPath of the corresponding element. In some cases, this segmentation may combine some inline visual elements into a single block. In such cases, a composite block is created and each inline visual element is pointed to the new composite block. For example, the visual unit `<a><img><br><a><img>` can be a direct child of a `<div>`. In this case, the first `<a>` and `<img>` nodes before the `<br>` node construct a composite block, which

does not have a valid XPath on the page. Therefore, the XPaths of two inline visual elements are mapped to the new composite block. In order to identify which granularity level is mostly preferred by users, a user study was also conducted by Akpınar and Yeşilada [2013]. Based on the ratings given by the users, the fifth granularity level is mostly preferred by users (approximately 74% user satisfaction). Therefore, we use this level to segment web pages.

The Levenshtein Distance (String-Edit) algorithm has been widely used to compare two scanpaths which are represented in terms of AoIs [Privitera and Stark 2000; Josephson and Holmes 2002; Pan et al. 2004; Heminghous and Duchowski 2006; Underwood et al. 2008; Duchowski et al. 2010]. This algorithm calculates a distance (i.e., dissimilarity) between two scanpaths by transforming one scanpath to another one by using the minimum number of operations which are insertion, deletion and substitution [Josephson and Holmes 2002]. For example, the distance between FCEB and GCEB is calculated as one because it is sufficient to substitute F with G or vice versa. In particular, Heminghous and Duchowski [2006] developed an application called iComp which firstly segments an image into its AoIs by using the fixation distribution over the image as suggested by Santella and DeCarlo [2004]. Once user scanpaths are correlated with the AoIs, it then applies the String-edit algorithm to calculate the distances between the scanpaths. However, the substitution costs between all pairs of AoIs may not be the same because the sizes of AoIs and the distances between AoIs may be different [Josephson and Holmes 2002]. To take this into consideration, it has been suggested to use a substitution matrix which stores substitution costs between all pairs of AoIs [Josephson and Holmes 2002; Takeuchi and Habuchi 2007]. Different ways can be used to generate a substitution matrix. Particularly, Takeuchi and Habuchi [2007] suggest to use the Euclidean distances between AoIs to create a substitution matrix. Equation 1 below illustrates how a substitution cost between two AoIs u and v is calculated based on the Euclidean distance where u_1 and u_2 are x and y coordinates of the centre of the Aoi u and α is a type of normalisation parameter [Takeuchi and Habuchi 2007]. Once the substitution matrix is constructed, it can be used with the String-edit algorithm to calculate a distance between two scanpaths.

$$f(\vec{u}, \vec{v}) = \alpha \sum_{i=1}^2 \sqrt{(u_i - v_i)^2} \quad (1)$$

As mathematically formalised in Equation 2 below, a similarity between two scanpaths can be calculated as a percentage (S) based on their String-edit distance [Underwood et al. 2008]. The distance (d) is firstly divided by the length of the longer scanpath (n) to have a normalised score. The normalised score allows preventing possible inconsistencies which can be caused by different lengths. Finally, the normalised score was subtracted from one and then multiplied by 100. Even though the String-edit algorithm can be used for the comparison purpose, the algorithm by itself cannot discover a trending scanpath for multiple scanpaths. Instead of calculating the dissimilarity first between two scanpaths, the Needleman and Wunsch algorithm has also been used to calculate a similarity score between two scanpaths based on a substitution matrix and gap penalty [Needleman and Wunsch 1970; Cristino et al. 2010].

$$S = 100 \cdot \left(1 - \frac{d}{n}\right) \quad (2)$$

To consider transition probabilities between AoIs based on multiple scanpaths, Markov models can be used [West et al. 2006; Chuk et al. 2014; Kang and Landry

2015]. With these models, a matrix can be created to store transition probabilities between AoIs so that it allows to identify which AoIs can be next (or previous) of a particular AoI with their probabilities. However, if these models are considered to discover a trending scanpath, some considerable questions will arise, especially what the start and the end point are, and which probabilities should be considered to construct a trending scanpath [Eraslan et al. 2014]. As also stated in the literature, these models are not able to tell whether or not there is a typical scanpath for multiple scanpaths [Abbott and Hrycak 1990; Josephson 2010].

Some other algorithms can be considered to address the limitations of Markov models. In particular, the Shortest Common Supersequence (SCS) algorithm have been mentioned in the literature which can be used to discover a general scanpath for multiple scanpaths [Räihä 2010]. However, this algorithm has important weaknesses. For example, when the individual scanpaths AWC, AXC, AYC and AZC are available, this algorithm discovers AWCXYZC as a general scanpath. As can be clearly seen from this example, this scanpath is quite long compared to the individual scanpaths. Besides, it is not supported by the majority of the individual scanpaths, for instance, it includes the AoI W but this AoI is included in only one of the individual scanpaths (AWC).

In order to detect patterns in multiple scanpaths, some algorithms have been proposed such as the pattern discovery algorithm of the eyePatterns tool [West et al. 2006], the SPAM algorithm [Ayres et al. 2002; Hejmady and Narayanan 2012] and the T-Pattern (Temporal-Pattern) Detection algorithm [Magnusson 2000; Burmester and Mast 2010; Mast and Burmester 2011]. The algorithm used in the eyePatterns tool is designed to search for exact patterns with minimum 3 elements in individual scanpaths and list them with how many times they are seen in the individual scanpaths and how many individual scanpaths include these patterns. However, this algorithm does not have tolerance to any extra items in scanpaths. For example, although the pattern DEF is included in both of the individual scanpaths CDEF and ADBEF, it cannot be detected because of the extra AoI B. Because of this reason, this algorithm is reductionist which means that the resulting path is likely to be so short that is not useful for further processing of web pages. The SPAM algorithm, which stands for Sequential Pattern Mining, has also been used to detect patterns in multiple scanpaths. It was originally developed for identifying frequent sequential patterns in a sequence database. This algorithm is similar to the algorithm used in the eyePatterns tool in terms of its objective but it has tolerance to extra items in scanpaths. However, it is still reductionist because the order of AoIs in individual scanpaths affects the pattern recognition. In particular, when the individual scanpaths ABCDE, ABDE and ACBED exist, this algorithm detects the patterns ABD and ABE as the longest patterns, even though the AoIs A, B, D and E are included in all of the individual scanpaths. Furthermore, the T-Pattern Detection algorithm, which is a commercial product³, has been applied to detect patterns within individual scanpaths. According to Mast and Burmester [2011], the algorithm requires a series of events as an input. The behaviour type (such as, fixating the AoI E) is considered as an event type, and the behaviour occurrence (such as, fixating the AoI E at 00:05:03) is considered as an event. Two event types become a T-Pattern when they occur more than once in the behaviour sequence in the same order and their occurrences are constantly distributed over time. There are other parameters which control how two event types should be distributed to become a T-Pattern and how detected T-Patterns should be combined with other T-Patterns or event types to provide longer T-Patterns. Since this algorithm has a number of different parameters, the results can be changed based on the adjustments of these parameters.

³<http://www.noldus.com/theme/t-pattern-analysis>

Sutcliffe and Namoun [2012] also focussed on investigating where users concentrate in very early phases of their searches by using a position-based weighted model and a 3x3 grid layout for web page segmentation. In their model, the corresponding AoIs of the first three fixations of users are identified which can be the same AoI. The constant values are then given to these AoIs based on their positions where the first one gets one, the second one gets 0.5 and the third one gets 0.2. Following this, a total value is calculated for each AoI and then the AoIs are sorted by these total values. However, this study did not aim to discover trending scanpaths of users on web pages. Holsanova et al. [2006] used a similar approach to analyse reading paths and reading priorities on newspaper spreads. They defined 10 AoIs on a newspaper spread and then ranked them according to the first visits of the AoIs by five users. However, it is obvious that the same AoI can be visited more than once by a user. Moreover, their approach does not eliminate any AoIs even though they do not get attention from users [Holmqvist et al. 2011]. Because of these reasons, their approach is also not suitable for identifying a trending scanpath.

As discussed above, a number of algorithms has been suggested to detect patterns in eye tracking scanpaths. However, these algorithms can detect more than one pattern for multiple scanpaths. This brings up a number of important problems, such as how multiple patterns can be used for re-engineering web pages, why there are multiple patterns, which factors reveal these multiple patterns. Although web pages can be differently re-engineered for different users, some additional information about users may be needed at the time of access to decide how to re-engineer the web pages. For example, if male and female users have different scan patterns on a particular page, their gender should be identified to decide which pattern should be used for re-engineering the page. However, it may not always be possible to access these information due to privacy reasons. Discovering the representative scanpath for all the group would counteract these problems. There are also several algorithms which can be used to discover only one scanpath for multiple users. The Multiple Sequence Alignment algorithm was one of these algorithms but it was not validated with any study [Hembrooke et al. 2006]. In addition, the Dotplots algorithm was also suggested for hierarchical clustering which can be used to discover one scanpath for multiple scanpaths [Goldberg and Helfman 2010]. In order to create a hierarchical structure, the most similar scanpaths are chosen from the scanpath list by using the Dotplots algorithm, then the merged scanpath of these scanpaths is introduced to the scanpath list and these two scanpaths are removed. This process is repeated until a single scanpath is left which can represent the general scanpath followed by all the users. They propose two ways to create a merged scanpath: (1) the merged scanpath can be created by merging two similar scanpaths by using the Dotplots algorithm (2) the merged scanpath can be assigned to one of the two similar scanpaths. Since the second approach is actually related to selecting one of the scanpaths from the existing scanpaths, it is not suitable for discovering the trending scanpath. Assume that there are 1000 users. In this case, it is not a good idea to select one of them to represent all of them as they typically do not follow exactly the same path. The Dotplots-based algorithm was used by Albanesi et al. [2011] and they called the resulting path a dominant path.

Before we developed our STA algorithm, we developed a preliminary algorithm called eMINE scanpath algorithm [Eraslan and Yesilada 2015; Eraslan et al. 2014; Yesilada et al. 2013]. This algorithm also uses hierarchical clustering. It selects the two most similar scanpaths by using the String-edit algorithm from the scanpath list and then finds their longest common subsequence (LCS). Next, it removes these two scanpaths and then introduces their longest common subsequence to the list. This process is repeated until a single scanpath is left which represents the general scanpath followed by all the users.

Because of hierarchical clustering, these algorithms are also likely to be reductionist [Chiang 2009]. Assume that the individual scanpaths S1:GATACCAT, S2:CTAAAG and S3:GCTATTGCG are available. S1 and S2 can be clustered first and then S1': A - A - - A - can be created. After that, S1' and S3 can be clustered and then S2': - - - A - - - - - can be created. As can be easily recognised from this example, although the AoIs G, C and T are also shared by all the individual scanpaths, they are not included in the resulting path (S2').

Our objective is to discover a trending scanpath for multiple users on a web page. However, as we discuss above, there is limited research in this field and existing research tends to be reductionist which means that the resulting path is likely to be so short, and therefore it not useful for further processing of web pages [Eraslan et al. 2016a]. Specifically, if the resulting path is used to re-engineer a web page, the commonly visited visual elements may be removed from the page because of their positions and/or small deviations in the individual scanpaths. Even though we had developed a preliminary algorithm to address the problem of being reductionist, there was still room for the improvement. These existing algorithms also tend to ignore other aspects of eye tracking data, especially fixation durations. For example, the algorithm used in the eyePatterns tool, the Dotplots-based algorithm and our preliminary algorithm do not take the duration aspect into consideration. However, it is widely accepted that fixation duration is associated with the depth of processing and the ease or difficulty of information processing [Velichkovsky et al. 2002; Follet et al. 2011], and therefore it should be considered. In order to address these limitations, we present our STA algorithm which discovers a trending scanpath of multiple web users in terms of visual elements of a web page.

4. SCANPATH TREND ANALYSIS

In order to analyse individual scanpaths and reveal their trending scanpaths on web pages, we iteratively developed the concept called *Scanpath Trend Analysis (STA)*. This section presents our STA algorithm which automatically carries out this trend analysis. This algorithm has a multi-pass approach. As it is summarised in Pseudocode 1, it has three main stages. The first stage is the preliminary stage where data is prepared for processing and then the following two stages are the main stages of identifying trending visual elements in multiple scanpaths and clustering the scanpaths into a trending scanpath based on the trending visual elements. Each of these stages are explained in detail in the following sections.

Pseudocode 1: Scanpath Trend Analysis (STA)

Input: Eye Tracking Data, Visual Elements

Output: Trending Scanpath

Preliminary Stage: Prepare individual scanpaths in terms of visual elements of a particular web page (Section 4.1)

First Pass: Identify trending visual elements in the individual scanpaths (Section 4.2)

Second Pass: Construct a trending scanpath by using the trending visual elements based on their overall positions in the individual scanpaths (Section 4.3)

return *The trending scanpath*

4.1. Preliminary Stage: Individual Scanpaths

In order to prepare individual scanpaths in terms of visual elements of web pages, we first segment web pages into their visual elements and label them with alphanumeric characters. After this, we export eye tracking data from eye tracking software. The eye

tracking data is then related to these visual elements to create individual scanpaths in terms of the visual elements of the web pages.

Although eye trackers typically record eye movements with high accuracy, they may still have inaccuracy in storing the locations of fixations which they refer to as the degree of accuracy [Tobii Technology AB 2011a]. This can potentially cause problems in identifying a trending scanpath. For instance, some fixations may be located outside of visual elements even though they should have been located in a particular visual element. Figure 2 illustrates an example of a fixation which should have been located on the menu item *iPad*.

Fig. 2: An example of a fixation which should have been located on the menu item *iPad*

In order to ensure that the STA algorithm takes this into account, by using the degree of accuracy provided by eye trackers, we developed an approach to extend the borders of the visual elements. This algorithm uses the formula shown in Equation 3 to calculate an error offset in pixels (E) by using the degree of accuracy (x), the distance between the participant and the screen (D) and the pixels per inch for the screen (P). This offset is then used to extend the visual elements horizontally and vertically.

$$E = \tan(x) \cdot D \cdot P \quad (3)$$

Once the visual elements are extended, the STA algorithm identifies where the fixations are located. When a fixation is located in more than one visual element because of the intersection of the extended visual elements, the fixation is considered to be located in its closest visual element. In order to find the closest visual element, the STA algorithm calculates the distances between the fixation and all the points included in the visual elements. Table I illustrates where the fixations of some users are located on the Apple web page (See Figure 1) with their durations. These are from our eye

Table I: Individual scanpaths of some users on the Apple web page. The numbers in brackets are fixation durations, in milliseconds (ms). As an entire example, the individual scanpath with an ID 5 is fully illustrated.

ID	Individual Scanpaths
1	C [179] B [239] F [399] F [319] F [399] F [1455] F [319] F [618] E [518] ...
2	C [538] F [319] F [279] E [259] F [180] F [279] I [199] F [279] F [239] ...
3	C [383] F [233] C [216] E [233] C [150] C [166] F [217] F [183] F [216] ...
4	F [250] C [300] C [250] E [466] C [250] C [316] C [200] H [350] H [183] ...
5	D [217] C [233] F [333] F [316] I [166] I [333] H [533] G [183] K [300] K [133] K [416] K [516] K [283] C [233] D [183] E [150] D [450] E [216] E [300] E [466] E [383] E [333] E [033] E [083] D [450] K [183] D [300] B [266] B [600]
6	F [233] C [250] C [183] E [233] E [233] H [200] H [416] E [233] D [500] ...
7	F [666] C [167] C [183] C [200] F [250] F [250] C [333] F [166] F [266] ...
8	F [183] F [316] C [266] H [183] G [183] E [250] E [167] H [166] I [316] ...
9	F [150] C [200] C [366] C [183] E [200] F [366] H [266] G [250] F [266] ...
...	...

tracking study that we conducted to evaluate the validity of the STA algorithm (See Section 5).

As an entire example, Table I fully shows the individual scanpath with an ID 5 in terms of the visual elements of the Apple web page with the fixation durations. In this example, the user first fixated the visual element D for 217 ms and then fixated the visual element C for 233 ms. After that, the user moved to the visual element F for 333 ms, then fixated the same visual element for 316 ms, then continued to fixate other elements, and finally fixated the visual element B twice for 266 ms and 600 ms.

Pseudocode 2 shows an overview of how the individual scanpaths are prepared in terms of the visual elements. This stage has a time complexity of $O(n^6)$. It is the slowest stage of the STA algorithm because of the fixations that are located in more than one visual element. The detailed complexity analysis of this part is provided in our external repository (See Open Data Section). Without these fixations, the time complexity of this stage is $O(n^3)$, however, when the STA algorithm detects a fixation that is located in more than one element, it first finds the related elements and then calculates the distance between the fixation and all of the points of those elements. Although the performance of the current implementation of the STA algorithm is not really efficient, its implementation can be optimised in the future.

4.2. First Pass: Trending Visual Elements

Once the individual scanpaths are prepared, the STA algorithm uses a data-driven approach in the first pass to analyse them to identify the trending visual elements by using their occurrence frequencies (i.e., the number of fixations on a visual element) and durations. It means that the algorithm analyses the individual scanpaths (i.e., data recorded by an eye tracker) to determine the threshold values for the occurrence frequencies and durations to decide whether a visual element is trending or not. Hence, the STA algorithm does not use pre-defined threshold values which could be biased in some way. The trending visual elements will be used later to construct a trending scanpath (See Section 4.3).

As can be seen from the individual scanpath with an ID 5 in Table I, visual elements can be repeated in individual scanpaths which can be a consecutive repetition, such as CFF, or a non-consecutive repetition, such as EDE. The consecutive repetition occurs when a user fixates the same visual element more than one time in succession whereas the non-consecutive repetition occurs when a user fixates a visual element (E) and fixates another visual element (ED) and then goes back to the previous visual element (EDE). We refer to each visit of a particular visual element as a visual ele-

Pseudocode 2: An Overview of Preparation of Individual Scanpaths**Input:** A series of fixations for each user, Extended visual elements**Output:** Individual Scanpaths represented in terms of the visual elements and fixation durations

```

foreach user do
  foreach fixation do
    foreach extended visual element do
      | Find the elements where the fixations are located
    end
    if the fixation is located in more than one extended visual element then
      | The closest element is added to the individual scanpath with the fixation duration
    else if the fixation is located in one extended visual element then
      | The element is added to the individual scanpath with the fixation duration
    end
  end
end
return Individual Scanpaths represented in terms of the visual elements and fixation durations

```

ment instance. In the individual scanpath with an ID 5 in Table I, the visual element K is visited two times (...KKKKK...K...). In the first time, the user fixated the visual element K five times in succession, and in the second time, the user fixated only one point in the visual element K and then moved on to another visual element. Hence, there are two instances of the visual element K in the individual scanpath.

Since we are interested in sequential patterns, we need to differentiate these visual element instances. Hence, these instances are numbered by the STA algorithm as shown in Example 4.1 (a numbered version of the individual scanpath with an ID 5 in Table I). In order to take information processing into consideration and make the numbering consistent for all individual scanpaths, the numbering is applied based on the durations of these instances as it is commonly accepted that fixation duration is related to the depth of processing and the ease or difficulty of information processing [Velichkovsky et al. 2002; Follet et al. 2011]. The longest instance of a particular visual element is numbered with one and its next longest instance is numbered with two and then continue for numbering its other instances. For example, the first instance of the visual element K in the individual scanpath shown in Example 4.1 is numbered with one because its duration ($K1 = 300 + 133 + 416 + 516 + 283 = 1648$ ms) is longer than another instance of the visual element K ($K2 = 183$ ms).

EXAMPLE 4.1. *An example of an individual scanpath in terms of the visual element instances of the Apple web page with the fixation durations*

```

D4[217ms] C2[233ms] F1[333ms] F1[316ms] I1[166ms] I1[333ms] H1[533ms] G1[183ms] K1[300ms]
K1[133ms] K1[416ms] K1[516ms] K1[283ms] C1[233ms] D5[183ms] E2[150ms] D2[450ms] E1[216ms]
E1[300ms] E1[466ms] E1[383ms] E1[333ms] E1[033ms] E1[083ms] D1[450ms] K2[183ms] D3[300ms]
B1[266ms] B1[600ms]

```

The STA algorithm then identifies the trending visual element instances for all the individual scanpaths which will be used for the purpose of clustering to construct a trending scanpath. When a particular visual element is fixated by all participants, it means that the visual element has important potential for processing [Poole and Ball 2005]. Because of this reason, when a particular visual element instance is shared by all the individual scanpaths, the STA algorithm considers the instance as a trending visual element instance. The algorithm also considers the occurrence frequencies and durations to identify trending visual element instances as these metrics are associated

with the attention of participants [Poole and Ball 2005]. As illustrated in Pseudocode 3, when a visual element instance is not shared by all participants but gets at least the same attention as the shared instances, the algorithm also identifies the instance as a trending visual element instance. In order to identify these trending instances, the algorithm firstly finds the instances which are shared by all individual scanpaths and then computes the minimum total occurrence frequency and the minimum total duration of the shared instances. When an instance has the same or higher total occurrence frequency than the minimum total occurrence frequency and the same or longer total duration than the minimum total duration, it is defined as a trending visual element instance. Other instances are then removed from the individual scanpaths. Table II illustrates the trending visual element instances of the individual scanpaths which are shown in Table I.

Pseudocode 3: Identification of Trending Visual Element Instances

Input: An instance, Individual Scanpaths in Terms of Visual Element Instances

Output: The marked Instance

if the instance is shared by all individual scanpaths **OR** it gets at least the same attention as the shared instances **then**

 | The instance is marked as a trending visual element instance

else

 | The instance is marked to be removed from the individual scanpaths

end

return The marked instance

Table II: Individual scanpaths with the trending visual element instances after the first pass. The numbers in brackets are fixation durations, in milliseconds. As an entire example, the individual scanpath with an ID 5 is fully illustrated.

ID	Individual Scanpaths with the Trending Visual Element Instances
1	F1 [399] F1 [319] F1 [399] F1 [1455] F1 [319] F1 [618] E2 [518] E2 [498] G1 [180] ...
2	C1 [538] F1 [319] F1 [279] H1 [239] H1 [279] J1 [199] J1 [319] F2 [518] G1 [160] ...
3	F1 [217] F1 [183] F1 [216] F1 [200] F1 [183] C1 [167] C1 [150] C1 [183] I1 [516] ...
4	F2 [250] E3 [466] H1 [350] H1 [183] I1 [300] C1 [200] C1 [250] C1 [117] C1 [200] ...
5	F1 [333] F1 [316] I1 [166] I1 [333] H1 [533] G1 [183] C1 [233] E2 [150] D2 [450] E1 [216] E1 [300] E1 [466] E1 [383] E1 [333] E1 [033] E1 [083] D1 [450] B1 [266] B1 [600]
6	I1 [700] J1 [316] J1 [733] E2 [266] E2 [316] D2 [866] G1 [200] G1 [1749] E3 [283] ...
7	F2 [666] C1 [167] C1 [183] C1 [200] E3 [366] F1 [200] F1 [333] F1 [167] I1 [150] ...
8	F1 [183] F1 [316] B2 [216] B2 [133] B2 [483] B2 [233] F2 [217] F2 [100] I1 [150] ...
9	C1 [200] C1 [366] C1 [183] E3 [200] F2 [366] H1 [266] G1 [250] I1 [183] I1 [383] ...
...	...

As an entire example, Table II shows the full representation of the individual scanpath with an ID 5 where its previous version is shown in Example 4.1. Assume that C2, D2, D3, D4, D5, K1 and K2 in the individual scanpath in Example 4.1 are not defined as the trending visual element instances. In this case, they are removed from the individual scanpath as shown in Table II.

The first pass of the STA algorithm has a time complexity of $O(n^3 \cdot \log n)$.

4.3. Second Pass: Trending Scanpath

In the second pass, a trending scanpath is constructed by clustering the individual scanpaths with the trending visual element instances in the individual scanpaths identified in the first pass (See Section 4.2). These visual element instances are located in the trending scanpath based on their overall positions in the individual scanpaths.

In this pass, the individual scanpaths are abstracted by combining the same visual element instances. Example 4.2 shows the abstracted version of the individual scanpath with an ID 5 in Table II. The total number of occurrences and the total duration for each visual element instance are also stored.

EXAMPLE 4.2. *An example of an abstracted individual scanpath in terms of the trending visual element instances of the Apple web page*

F1 I1 H1 G1 C1 E2 D2 E1 D1 B1

In order to find out the proper positions of the trending visual element instances in the trending scanpath, the STA algorithm calculates a priority value (ψ_i : the priority of the i^{th} instance) for each visual element instance for each individual scanpath with the formula shown in Equation 4 where P_i represents the position of the visual element instance (starting from zero) and L illustrates the length of the individual scanpath. We take the maximum value ($\max_i$) as 1 and the minimum value ($\min_i$) as 0.1. This priority value is a normalised value to prevent the possible effects of the different lengths of the individual scanpaths. Table III shows the priority values (ψ_i) of the visual element instances in the individual scanpath shown in Example 4.2.

$$\psi_i = 1 - P_i \cdot z, \text{ where } z = \frac{\max_i - \min_i}{L - 1} \quad (4)$$

Table III: The priority values (ψ_i) of the visual element instances in the individual scanpath shown in Example 4.2

Visual Element Instance	Priority Value ψ_i where $z = \frac{1-0.1}{10-1}$
F1	$1 - 0 \cdot z = 1.0$
I1	$1 - 1 \cdot z = 0.9$
H1	$1 - 2 \cdot z = 0.8$
G1	$1 - 3 \cdot z = 0.7$
C1	$1 - 4 \cdot z = 0.6$
E2	$1 - 5 \cdot z = 0.5$
D2	$1 - 6 \cdot z = 0.4$
E1	$1 - 7 \cdot z = 0.3$
D1	$1 - 8 \cdot z = 0.2$
B1	$1 - 9 \cdot z = 0.1$

Once these priority values are calculated, the STA algorithm then calculates the total priority value (Ψ) for each visual element instance in all individual scanpaths with the formula shown in Equation 5 where n shows the number of individual scanpaths.

$$\Psi = \sum_{i=1}^n \psi_i \quad (5)$$

The STA algorithm locates the visual element instances in the trending scanpath based on their total priority values where the visual element instance with the highest total priority value is located at the beginning of the trending scanpath. When a trending visual element instance that is not shared by all individual scanpaths has lower

Fig. 3: The Apple web page with its visual elements and the trending scanpath

total priority value than the minimum total priority value of the shared visual element instances, it will not be included in the trending scanpath. In case more than one visual element instance have the same total priority value, the STA algorithm also uses their total duration and total number of occurrences to locate them into the trending scanpath (arranged from the largest values to the smallest values).

When the trending scanpath is constructed, it is not needed to have the numbers of the visual element instances anymore. Therefore, the numbers are ignored and then the trending scanpath is abstracted to remove consecutive repetitions. An example of a trending scanpath of web users on the Apple web page is shown in Example 4.3. This is also illustrated in Figure 3. The STA algorithm can also give a total fixation duration for each visual element in the trending scanpath by keeping the fixation durations until the last step.

EXAMPLE 4.3. *An example of a trending scanpath of users on the Apple web page*
F C I F E H E J D G D E B

The second pass of the STA algorithm has a time complexity of $O(n^3)$.

5. EVALUATION

In order to evaluate the STA algorithm, we conducted an eye tracking study with 40 participants in two different universities: Middle East Technical University Northern Cyprus Campus and the University of Manchester in the UK [Eraslan et al. 2014]. This eye tracking study was also used to evaluate our preliminary simple algorithm, called eMINE scanpath algorithm [Eraslan et al. 2014]. The eye tracking study shows that although the eMINE scanpath algorithm could identify a general scanpath for

multiple users in terms of visual elements of a web page, it still had limitations and there was still room for improvement. Therefore, we developed the STA algorithm and used the same eye tracking study to investigate the validity of the STA. In order to do that, we focused on the following three hypotheses.

H1. When the participants are asked to complete a task by fixating certain visual elements in a particular order, the trending scanpath generated by the STA algorithm includes these visual elements in that order. The eye tracking study conducted was a task based study. Therefore, we expect the outcome to include the sequence of the visual elements that is required for the successful completion of the task.

H2. The trending scanpath generated by the STA algorithm includes all the visual elements fixated by all the participants. This is to ensure that the trend analysis covers all the visual elements that are visited by all the participants. If a visual element is common in all participants' scanpaths, we expect to see that in the trend analysis outcome.

H3. The trending scanpath generated by the STA algorithm is the most similar to the individual scanpaths. We expect to see that the STA algorithm is doing better analysis by showing that the outcome scanpath is more similar to the inputted scanpaths compared to other existing work in the literature.

5.1. Participants

Twenty female and twenty male participated in this study. The majority of these participants were students, along with some academic and administrative staff at the universities. They all used the web daily. Most of these participants (19 participants) were aged between 18-24, then 25-34 group (15 participants) and then 35-54 group (six participants). Moreover, 14 participants completed their high/secondary schools, seven participants had a bachelor's degree, nine participants completed their master's degree and nine participants had their Ph.D. degrees. Furthermore, 35 participants did not have any problem with their vision whereas five participants had corrected vision.

5.2. Equipment and Materials

We recorded eye movements of these participants on the saved versions of the Apple, Babylon, AVG, Yahoo, Godaddy and BBC web pages by using a 17" monitor with the built-in Tobii T60 eye tracker and the screen resolution of 1280 x 1024. The degree of accuracy of the eye tracker is given as typically 0.5 degrees [Tobii Technology AB 2011b]. The data quality can be affected in different conditions [Holmqvist et al. 2012]. However, the degree of accuracy of Tobii T60 eye tracker was tested in different conditions (ideal conditions, large gaze angles, varying head positions, varying illumination and white background) by the Department of Quality Assurance, Hardware Division of Tobii Technology AB [Tobii Technology AB 2011a]. Their tests show that the degree of the accuracy of Tobii T60 eye tracker was stable across the various conditions. As the degree of accuracy is one of the parameters of the STA algorithm, it can be changed by other researchers who want to use this algorithm in their studies.

The web pages used in this study did not include any animated object apart from the Babylon and BBC pages. On the Babylon page, there was an animated arrow to highlight the link for downloading the free version of Babylon. On the BBC page, the content of the top news was changing automatically. As the STA algorithm identifies a trending scanpath on a particular page and we would like to identify the trending scanpaths on the six web pages, the participants were not allowed to use a mouse or keyboard to visit other pages within the websites. In particular, they did not follow the links on the web pages. Figures 3-8 show the web pages with their visual elements which were identified by the extended and improved version of the VIPS

Fig. 4: The Babylon web page with its visual elements

Fig. 5: The AVG web page with its visual elements

Fig. 6: The Yahoo web page with its visual elements

Fig. 7: The Godaddy web page with its visual elements

Fig. 8: The BBC web page with its visual elements

algorithm [Akpınar and Yeşilada 2013] (See also Open Data Section). This VIPS algorithm discovers visual elements of web pages based on a granularity level where smaller visual elements are discovered in the higher levels. As the fifth segmentation granularity level is mostly preferred by users (approximately 74% user satisfaction), we preferred to use this level to segment the six web pages in our experiments.

These web pages were randomly selected from a group of pages that were used in a study that focused on evaluating the extended and improved version of the VIPS algorithm [Akpınar and Yeşilada 2013]. In that study, Akpınar and Yeşilada [2013] used the Visual Complexity Rankings and Accessibility Metrics (VICRAM) framework to investigate the top 100 web pages listed by ALEXA.com⁴. The ViCRAM framework was designed to discover visual complexities of web pages in correlation with users' perception [Michailidou 2010]. When they determined the visual complexities of the web pages, they randomly chose 10 web pages for each of the low, medium and high complexity groups. In order to eliminate the possible effects of the web page complexity on the evaluation of the STA algorithm, we randomly selected two web pages from each complexity group. The Apple and Babylon pages have a low complexity, the AVG and Yahoo pages have a medium complexity, and the Godaddy and BBC pages have a high complexity.

5.3. Procedure

The participants first read the information sheet which explains the main objectives of our study, what happens to their eye tracking data and their rights. After that, they signed the consent form. They then filled in a short questionnaire which aimed to collect their basic demographic information which are gender, age groups and education

⁴<http://www.alexa.com>

levels. Following this, the participants sat in front of the eye tracker and then viewed the six web pages twice for searching (maximum 120 seconds for each page) and browsing in a random order. Thirty seconds were given to the participants for browsing on each web page as used in other studies [Pan et al. 2004; Jay and Brown 2008]. For the browsing tasks, the participants were asked to freely view the web pages without any particular objective, and for the searching tasks, they were asked to find some specific information and/or items on the web pages while their eye movements were being recorded. The searching tasks are as follows:

- (1) Apple web page — See Figure 3
 - (a) Can you locate the link that allows watching the TV ads relating to iPad mini?
 - (b) Can you locate a link labelled *iPad* on the main menu?
- (2) Babylon web page — See Figure 4
 - (a) Can you locate the link that you can download the free version of Babylon?
 - (b) Can you find and read the names of other products of Babylon?
- (3) AVG web page — See Figure 5
 - (a) Can you locate the link which you can download a free trial of AVG Internet Security 2013?
 - (b) Can you locate the link which allows you to download AVG Antivirus Free 2013?
- (4) Yahoo web page — See Figure 6
 - (a) Can you read the titles of the main headlines which have smaller images?
 - (b) Can you read the first item under News title?
- (5) Godaddy web page — See Figure 7
 - (a) Can you find a telephone number for technical support and read it?
 - (b) Can you locate a text box where you can search for a new domain?
- (6) BBC web page — See Figure 8
 - (a) Can you read the first item of Sport News?
 - (b) Can you locate the table that shows market data under the Business title?

The participants were expected to fixate some specific visual elements in a particular order to complete the searching tasks successfully. For instance, on the Godaddy web page, the participants firstly required to fixate the visual element F for the telephone number of the technical support, and then the visual element M for the text box where they could search for a new domain. We expected to see this sequence within the resulting scanpath. Therefore, we created a baseline with these searching tasks to test whether the STA algorithm meets the expectations.

There has been significant research on categorising user tasks on the web. One of the most popular models in this field is G. Marchionini's Search Activities Model which divides the tasks into the lookup, learn and investigation groups [Marchionini 2006]. Our searching tasks are associated with fact finding that is related to the lookup group whereas our browsing tasks are associated with serendipitous browsing which is related to the investigation group.

In order to deal with any possible short term memory effects, the first half of the participants completed the searching tasks on their first visits to the web pages and the browsing tasks on their second visits whereas another half completed the browsing tasks on their first visits and the searching tasks on their second visits. As mentioned above, the web pages were also randomised for each participant. For each transition from one page to another, the previous web browser was closed and a new web browser was opened automatically. Both of the transition and the randomisation of the web pages and tasks allowed to deal with the effects of the previous tasks.

5.4. Pilot Study

Before we started to conduct our eye tracking study, we conducted a pilot study with four users to see if the tasks are clear and the durations are sufficient to complete the tasks. In the pilot study, the users mainly understood and completed their searching tasks within the given time. They also indicated that more than 30 seconds for the browsing tasks could be boring. Therefore, more than 30 seconds could decrease the quality of data for the browsing tasks. Since we did not encounter with any problem regarding the tasks and durations, we did not need to make any changes for our eye tracking study.

6. RESULTS

In the eye tracking study, one male participant changed his seating position while he was reading the web pages, and therefore the eye tracker could not record his eye movements. Another male participant encountered some considerable problems with his eye calibration on the web pages. Hence, we had to remove these two participants from the evaluation of the STA algorithm. In addition, some participants were not able to complete the searching tasks successfully and/or had eye calibration problems. We defined these participants as unsuccessful participants and had to exclude them from our further analysis. Here, we present our major findings in terms of our hypotheses.

6.1. Including Expected Sequences

The participants were asked to complete some searching tasks on six web pages. In order to complete their searching tasks successfully, they needed to locate some specific parts of a given web page in a particular order. For example, on the Apple web page shown in Figure 3, the participants were asked to locate the link which allows them to watch the TV ads, included in the visual element E, and then locate a menu item *iPad*, included in the visual element B. Therefore, the trending scanpath of the Apple web page should include EB. With the searching tasks, we created a baseline to test whether the STA algorithm provides a trending scanpath that includes the expected sequence. We then hypothesised that the trending scanpaths identified by the STA algorithm includes the required sequences for the successful completion of these tasks.

In order to support our first hypothesis *H1: When the participants are asked to complete a task by fixating certain visual elements in a particular order, the trending scanpath generated by the STA algorithm includes these visual elements in that order*, we used the STA algorithm to identify the trending scanpaths of the participants for each of the six web pages for the searching tasks. We were then able to investigate whether or not these trending scanpaths include the expected sequences on the web pages. Table IV shows these trending scanpaths with the expected sequences which should be included in the trending scanpaths⁵.

The results in Table IV shows that all the trending scanpaths include the expected sequences on the web pages. For instance, on the Babylon web page, the participants were asked to locate the link which allows them to download the free version of Babylon and then read the names of other products of Babylon. Since the visual element M is associated with downloading the free version of Babylon and the visual elements P, Q, R and S are related to four other products of Babylon, the expected sequence on the Babylon web page is MPQRS. As can be seen from Table IV, the trending scanpath identified on the Babylon web page for the searching tasks (**MHIRPNPQRS**) includes the expected sequence. Based on these results, we can suggest that our first hypothesis is supported.

⁵The figures which show the trending scanpaths on the six web pages can be downloaded from http://wel-data.cs.manchester.ac.uk/data_files/18.

Table IV: The trending scanpaths identified by the STA algorithm [#: The number of successful participants] & The expected sequences which should be included in the trending scanpaths

Page Name	Expected Sequence	#	Trending Scanpath
Apple	EB	34	I C F H E B G B
Babylon	MPQRS	36	M H I R P N P Q R S
AVG	GI	36	G I
Yahoo	I	32	J I G J I
Godaddy	FM	30	O F O N M L M
BBC	RN	38	L P R P S T R N

Table V: The inclusion of the expected sequences in the resulting paths of the STA and other existing algorithms

Page	STA	eMINE	eyePatterns	Dotplots	SPAM
Apple	✓	✓	✗	✗	✓
Babylon	✓	✓	✗	✗	✓
AVG	✓	✓	✗	✓	✓
Yahoo	✓	✓	✗	✓	✓
Godaddy	✓	✗	✗	✗	✓
BBC	✓	✓	✗	✗	✓
Total	6 out of 6	5 out of 6	0 out of 6	2 out of 6	6 out of 6

To compare the STA algorithm with other existing algorithms in this context, we applied existing appropriate algorithms (the algorithm used in the eyePatterns tool⁶ [West et al. 2006], the Dotplots-based algorithm [Goldberg and Helfman 2010], the SPAM algorithm⁷ [Ayres et al. 2002; Hejmady and Narayanan 2012; Fournier-Viger et al. 2014], and our previous algorithm called eMine⁸ [Eraslan et al. 2014; Yesilada et al. 2013]) to provide the results for all the individual scanpaths on the six web pages for the searching tasks. The SPAM algorithm detected multiple patterns for some pages. In order to be fair, we checked all of its longest detected patterns. There are some other existing algorithms, such as iComp [Heminghous and Duchowski 2006], to analyse user scanpaths but those algorithms were not developed for identifying a general path for multiple users (See Section 3).

Table V shows whether or not the resulting paths of the STA algorithm and other existing appropriate algorithms include the expected sequences for the user tasks on the six web pages. In particular, the eMINE scanpath algorithm includes the expected sequences for the user tasks in 5 out of 6 cases. Apart from the STA algorithm and the SPAM algorithm, the resulting paths of algorithms do not always include those sequences.

6.2. Addressing the Reductionist Problem

Existing research tends to lose the visual elements that are fixated by all users but in a different order [Eraslan et al. 2013; Eraslan et al. 2016a]. This makes existing algorithms reductionist (See Section 3). In order to address this problem, we designed the STA algorithm to ensure that when a particular element is fixated by all users, it will be marked as a trending visual element and included in the trending scanpath. Therefore, our second hypothesis says *H2: The trending scanpath generated by the STA algorithm includes all the visual elements fixated by all the participants.*

⁶<http://sourceforge.net/projects/eyepatterns/>

⁷<http://www.philippe-fournier-viger.com/spmf/>

⁸<http://emine.ncc.metu.edu.tr/software.html>

Table VI: The trending scanpaths identified by the STA algorithm & The visual elements which were fixated by all of the participants [#: The number of successful participants]

Task	Page Name	#	Shared Visual Elements	Trending Scanpath
Searching	Apple	34	B, E, F, I	I C F H E B G B
	Babylon	36	M, P, Q, R, S	M H I R P N P Q R S
	AVG	36	G, I	G I
	Yahoo	32	G, I, J	J I G J I
	Godaddy	30	F, L, M, O	O F O N M L M
	BBC	38	P, R, N	L P R P S T R N
Browsing	Apple	37	C, F, E, G, H	C F I H B G E G
	Babylon	37	M, P, S	M H L I R Q M R S Q P
	AVG	38	G, I	G I G I
	Yahoo	38	G, I, J	I J G J I
	Godaddy	37	M, N, O	O M N O
	BBC	38	L, P, T	L S T P

In order to support our second hypothesis, we checked whether or not the trending scanpaths identified by the STA algorithm for the browsing and searching tasks include the visual elements shared by all the individual scanpaths. Table VI shows these trending scanpaths with the visual elements shared by all the individual scanpaths⁵. Based on the results in Table VI where the shared visual elements are shown in bold in the trending scanpaths, all the trending scanpaths identified by the STA algorithm include the visual elements which were fixated by all the participants. For example, the trending scanpath which was identified for 38 participants on the BBC web page for the browsing task is LSTP where the visual elements L, P and T were fixated by all the participants. These results support our second hypothesis.

Table VII: The rates of the inclusion of the shared visual elements for the STA algorithm and other existing appropriate algorithms on the six web pages for the browsing and searching tasks

Task	Algorithms	Mean	Median	Std. Deviation	Min	Max
Browsing	STA	1.00	1.00	0.00	1.00	1.00
	eMINE	0.46	0.33	0.27	0.33	1.00
	eyePatterns	0.00	0.00	0.00	0.00	0.00
	Dotplots	0.14	0.00	0.22	0.00	0.50
	SPAM	0.82	0.83	0.20	0.60	1.00
Searching	STA	1.00	1.00	0.00	1.00	1.00
	eMINE	0.67	0.58	0.28	0.33	1.00
	eyePatterns	0.00	0.00	0.00	0.00	0.00
	Dotplots	0.26	0.13	0.39	0.00	1.00
	SPAM	0.82	0.88	0.21	0.50	1.00

We also compared the STA algorithm with other existing appropriate algorithms in this context. To achieve this, we calculated the rate of the inclusion of the shared visual elements by dividing the number of shared visual elements in the resulting path by the total number of shared visual elements on a web page. As the SPAM algorithm identified several patterns for some pages, we selected the one with the highest rate. This means that we picked its best match. Table VII shows the mean, median, standard deviation, minimum and maximum values for the rates for the STA algorithm and other algorithms on the six web pages for the browsing and searching tasks. As shown in this table, only the STA algorithm fully addresses the reductionist problem (Mean: 1.00, Std. Deviation: 0.00) whereas the results of other existing algorithms do not always include the shared visual elements (Mean < 1.00, Std. Deviation > 0.00).

6.3. Similarity to the Individual Scanpaths

Since the STA algorithm constructs a trending scanpath based on individual scanpaths, the trending scanpath should be similar to the individual scanpaths. However, it is not possible to have 100% similarity to the individual scanpaths unless all the individual scanpaths are the same. In addition, when the individual scanpaths are completely different from each other in terms of their coverage, such as ABC, KLM and XYZ, it is not possible to identify any trending scanpath. Therefore, the similarities between the trending scanpath and the individual scanpaths are strongly related to the similarities between the individual scanpaths. We expect to see that the median similarity of the trending scanpath to the individual scanpaths is equal or greater than the minimum similarity between the individual scanpaths.

Our third hypothesis says *H3: The trending scanpath generated by the STA algorithm is the most similar to the individual scanpaths*. In order to support our hypothesis, we compared the results of our algorithm with the results of other existing algorithms on the six web pages for the browsing and searching tasks. The SPAM algorithm detected more than one pattern for some pages. In order to be fair, we chose the one with the highest similarities to the individual scanpaths. This means that we picked their best match.

Once the resulting scanpaths were identified, we then calculated the similarities between the resulting scanpaths and the individual scanpaths. These similarities were calculated by using the Levenshtein Distance (String-edit) algorithm as described in Section 3. The standard String-edit algorithm does not take fixation durations into account but this does not cause any problem for our experiments because the existing algorithms, mentioned above, do not include fixation durations in their results. Moreover, these algorithms do not apply any substitution while they are identifying the common scanpaths of users, and therefore no substitution matrix was created and used for similarity calculation.

Tables VIII and IX show the mean, median, standard deviation, minimum and maximum similarities between the individual scanpaths and the resulting scanpaths of our STA algorithm and other existing algorithms on the six web pages. These tables also show the minimum and maximum similarities between the individual scanpaths for each web page. Table VIII presents the results for the browsing tasks and Table IX presents the results for the searching tasks.

In order to investigate whether or not the differences between the results of the STA algorithm and other existing algorithms are statistically significant and not due to chance, we applied some statistical tests [Pallant 2007]. As it has already been stated, the algorithms were applied to the same participants. This means that we had a repeated measures design. Hence, if the distribution of the differences in the similarities to the individual scanpaths was normal, we used the dependent T-Test (one-sided). Otherwise, we used the Wilcoxon Signed Rank test. The T-Test provides t value, which is positive when the STA algorithm has higher similarities, and the Wilcoxon test provides z value. The normality of the results were assessed with the Shapiro-Wilk test.

As usual, the statistical tests were conducted with 95% confidence interval [Pallant 2007]. Therefore, the p value should have been less than 0.05 to illustrate a significant difference between the results of two algorithms. However, when the number of pairwise comparisons increase, the probability of identifying a significant difference due to chance increases [Napierala 2012]. Since the STA algorithm was compared with other four algorithms in a pairwise manner on each web page for both the browsing and searching tasks, the Bonferroni correction method was applied to adjust the p value by dividing it by the number of comparisons ($0.05/4 = 0.0125$) [Napierala 2012]. Hence, in

Table VIII: The minimum and maximum similarities between the individual scanpaths on the web pages for the browsing tasks, and the similarities between the individual scanpaths and the resulting scanpaths of the STA and other existing algorithms [NA: Not applicable as the existing algorithm could not detect any scanpath]

Page Name	Min and Max Similarities Between Individual Scanpaths	Algorithm	Mean	Median	Std. Deviation	Min	Max
Apple	Min: 10.86 Max: 55.81	STA	15.98	15.22	3.42	10.77	24.14
		eMINE	7.25	6.67	2.26	4.48	15.79
		eyePatterns	NA	NA	NA	NA	NA
		Dotplots	NA	NA	NA	NA	NA
		SPAM	9.67	8.89	3.01	5.97	21.05
Babylon	Min: 10.53 Max: 41.67	STA	19.06	18.87	5.02	11.48	34.78
		eMINE	2.31	2.13	0.67	1.41	4.35
		eyePatterns	NA	NA	NA	NA	NA
		Dotplots	NA	NA	NA	NA	NA
		SPAM	4.61	4.26	1.33	2.82	8.70
AVG	Min: 12.50 Max: 51.52	STA	13.18	12.12	5.28	6.67	33.33
		eMINE	6.59	6.06	2.64	3.33	16.67
		eyePatterns	NA	NA	NA	NA	NA
		Dotplots	3.29	3.03	1.32	1.67	8.33
		SPAM	13.18	12.12	5.28	6.67	33.33
Yahoo	Min: 22.22 Max: 73.91	STA	22.57	21.29	6.11	11.11	35.71
		eMINE	4.58	4.26	1.31	2.22	7.14
		eyePatterns	NA	NA	NA	NA	NA
		Dotplots	4.58	4.26	1.31	2.22	7.14
		SPAM	18.34	17.03	5.23	8.89	28.57
Godaddy	Min: 10.87 Max: 57.14	STA	15.83	13.79	9.42	8.00	66.67
		eMINE	3.99	3.45	2.37	2.00	16.67
		eyePatterns	NA	NA	NA	NA	NA
		Dotplots	3.99	3.45	2.37	2.00	16.67
		SPAM	11.98	10.34	7.10	6.00	50.00
BBC	Min: 8.82 Max: 50.00	STA	12.30	11.65	4.06	6.45	22.22
		eMINE	3.23	3.03	1.10	1.61	6.25
		eyePatterns	NA	NA	NA	NA	NA
		Dotplots	NA	NA	NA	NA	NA
		SPAM	9.70	9.09	3.31	4.84	18.75

our tests, when the p value was less than 0.0125, it illustrated a significant difference between the results of two algorithms.

To show the strength of the differences between the results of the STA algorithm and other existing algorithms, the effect sizes were also calculated. The Cohen's d value was calculated as the effect size when the dependent T-test was applied (.2: Small Effect, .5: Medium Effect, .8: Large Effect) whereas the r value was calculated as the effect size when the Wilcoxon test was applied (.1: Small Effect, .3: Medium Effect, .5: Large Effect) [Cohen 1988; Pallant 2007]. Table X illustrates the statistical test results.

Results for the Browsing Tasks

For the browsing tasks (See Table VIII), the algorithm used in the eyePatterns tool could not discover any path for the participants on all of the six web pages. Besides this, the Dotplots-based algorithm could not provide any result on the Apple, Babylon and BBC web pages. As can be seen from Table VIII, the median similarities of the trending scanpaths generated by the STA algorithm to the individual scanpaths are greater than the minimum similarities between the individual scanpaths on the

Table IX: The minimum and maximum similarities between the individual scanpaths on the web pages for the searching tasks, and the similarities between the individual scanpaths and the resulting scanpaths of the STA and other existing algorithms [NA: Not applicable as the existing algorithm could not detect any scanpath]

Page Name	Min and Max Similarities Between Individual Scanpaths	Algorithm	Mean	Median	Std. Deviation	Min	Max
Apple	Min: 15.87 Max: 51.52	STA	18.80	18.07	6.57	7.94	36.84
		eMINE	6.18	5.71	2.36	2.53	12.50
		eyePatterns	NA	NA	NA	NA	NA
		Dotplots	NA	NA	NA	NA	NA
		SPAM	6.18	5.71	2.36	2.53	12.50
Babylon	Min: 14.29 Max: 52.94	STA	20.78	20.67	7.25	9.17	42.11
		eMINE	11.26	11.25	4.50	4.59	26.32
		eyePatterns	NA	NA	NA	NA	NA
		Dotplots	NA	NA	NA	NA	NA
		SPAM	13.52	13.50	5.40	5.50	31.58
AVG	Min: 12.50 Max: 76.92	STA	9.72	7.85	5.33	4.17	28.57
		eMINE	9.72	7.85	5.33	4.17	28.57
		eyePatterns	NA	NA	NA	NA	NA
		Dotplots	9.72	7.85	5.33	4.17	28.57
		SPAM	19.44	15.69	10.67	8.33	57.14
Yahoo	Min: 10.98 Max: 70.00	STA	21.79	20.00	9.92	6.10	44.44
		eMINE	4.77	4.45	2.41	1.22	11.11
		eyePatterns	NA	NA	NA	NA	NA
		Dotplots	4.77	4.45	2.41	1.22	11.11
		SPAM	14.31	13.34	7.24	3.66	33.33
Godaddy	Min: 13.51 Max: 57.14	STA	20.31	20.39	5.05	10.45	28.57
		eMINE	6.38	6.56	1.86	2.99	10.00
		eyePatterns	NA	NA	NA	NA	NA
		Dotplots	3.19	3.28	0.93	1.49	5.00
		SPAM	12.76	13.11	3.72	5.97	20.00
BBC	Min: 10.47 Max: 56.00	STA	22.19	23.33	6.40	9.30	33.33
		eMINE	7.12	7.14	2.76	2.33	16.67
		eyePatterns	NA	NA	NA	NA	NA
		Dotplots	NA	NA	NA	NA	NA
		SPAM	10.68	10.71	4.14	3.49	25.00

majority of the web pages (Apple, Babylon, Godaddy and BBC web pages). This also exists for the SPAM algorithm on the BBC web page. Moreover, the statistical test results (See Table X) present that the STA algorithm performs significantly better with a large effect (apart from the BBC page where the effect size is medium between the STA and SPAM algorithms) compared to other existing algorithms and the probability of the results being a chance is below 0.05%. The only exception to this is the SPAM algorithm on the AVG page. On that page, both the STA and SPAM algorithms provided the same trending scanpath.

Results for the Searching Tasks

For the searching tasks (See Table IX), the algorithm used in the eyePatterns tool again could not discover any path for the participants on all of the six web pages. Besides this, the Dotplots-based algorithm could not provide any result on the Apple, Babylon and BBC web pages. Based on the results in Table IX, we can also suggest that the median similarities of the trending scanpaths of the STA algorithm to the individual scanpaths are greater than the minimum similarities between the individual scanpaths on the web pages, except the AVG web page. This also occurs for the SPAM

Table X: The statistical test results for the browsing and searching tasks [NA: Not applicable as the existing algorithm could not detect any scanpath, NA⁺: Not applicable as the degree of freedom (df) is not related to the Wilcoxon test, =: The resulting paths of the STA algorithm and the existing algorithm have the same similarities to the individual scanpaths, N: The number of successful users, *p<.05, **p<.01, ***p<.0001]

Task	Page	Algorithms	Test	N	df	t or Z Value	Effect Size
Browsing	Apple	STA-eMINE	Wilcoxon	37	NA ⁺	-5.232***	0.61
		STA-eyePatterns	NA	NA	NA	NA	NA
		STA-Dotplots	NA	NA	NA	NA	NA
		STA-SPAM	Wilcoxon	37	NA ⁺	-5.167***	0.60
	Babylon	STA-eMINE	Wilcoxon	37	NA ⁺	-5.303***	0.62
		STA-eyePatterns	NA	NA	NA	NA	NA
		STA-Dotplots	NA	NA	NA	NA	NA
		STA-SPAM	Wilcoxon	37	NA ⁺	-5.303***	0.62
	AVG	STA-eMINE	Wilcoxon	38	NA ⁺	-5.374***	0.62
		STA-eyePatterns	NA	NA	NA	NA	NA
		STA-Dotplots	Wilcoxon	38	NA ⁺	-5.374***	0.62
		STA-SPAM	=	=	=	=	=
	Yahoo	STA-eMINE	T-Test	38	37	22.881***	4.07
		STA-eyePatterns	NA	NA	NA	NA	NA
		STA-Dotplots	T-Test	38	37	22.881***	4.07
		STA-SPAM	Wilcoxon	38	NA ⁺	-5.235***	0.60
	Godaddy	STA-eMINE	Wilcoxon	37	NA ⁺	-5.305***	0.62
		STA-eyePatterns	NA	NA	NA	NA	NA
		STA-Dotplots	Wilcoxon	37	NA ⁺	-5.305***	0.62
		STA-SPAM	Wilcoxon	37	NA ⁺	-5.234***	0.61
	BBC	STA-eMINE	Wilcoxon	38	NA ⁺	-5.374***	0.62
		STA-eyePatterns	NA	NA	NA	NA	NA
		STA-Dotplots	NA	NA	NA	NA	NA
		STA-SPAM	T-Test	38	37	10.771***	0.70
Searching	Apple	STA-eMINE	Wilcoxon	34	NA ⁺	-5.087***	0.62
		STA-eyePatterns	NA	NA	NA	NA	NA
		STA-Dotplots	NA	NA	NA	NA	NA
		STA-SPAM	Wilcoxon	34	NA ⁺	-5.087***	0.62
	Babylon	STA-eMINE	T-Test	36	35	19.072***	1.58
		STA-eyePatterns	NA	NA	NA	NA	NA
		STA-Dotplots	NA	NA	NA	NA	NA
		STA-SPAM	T-Test	36	35	19.248***	1.14
	AVG	STA-eMINE	=	=	=	=	=
		STA-eyePatterns	NA	NA	NA	NA	NA
		STA-Dotplots	=	=	=	=	=
		STA-SPAM	Wilcoxon	36	NA ⁺	-5.233***	0.62
	Yahoo	STA-eMINE	T-Test	32	31	12.550***	2.36
		STA-eyePatterns	NA	NA	NA	NA	NA
		STA-Dotplots	T-Test	32	31	12.550***	2.36
		STA-SPAM	T-Test	32	31	10.963***	0.86
	Godaddy	STA-eMINE	T-Test	30	29	22.804***	3.66
		STA-eyePatterns	NA	NA	NA	NA	NA
		STA-Dotplots	T-Test	30	29	22.436***	4.72
		STA-SPAM	T-Test	30	29	21.121***	1.70
	BBC	STA-eMINE	T-Test	38	37	20.443***	3.06
		STA-eyePatterns	NA	NA	NA	NA	NA
		STA-Dotplots	NA	NA	NA	NA	NA
		STA-SPAM	T-Test	38	37	17.598***	2.14

algorithm on the AVG, Yahoo and BBC web pages. Furthermore, the statistical results tell us that (See Table X) the STA algorithm provides significantly better results with a large effect and the probability of the results being a chance is below 0.05%, except the AVG page. On the AVG page, the eMINE scanpath algorithm and the Dotplots-

based algorithm provided the same result as our STA algorithm. However, the SPAM algorithm provided a path which is more similar to the individual scanpaths compared to other algorithms ($p < .0001$).

Summary

To sum up, our STA algorithm, the eMINE scanpath algorithm, the algorithm used in the eyePatterns tool, the Dotplots-based algorithm and the SPAM algorithm were tested on the six web pages for both the browsing and searching tasks, and therefore 12 cases in total. In order to investigate whether or not our STA algorithm performs significantly better in terms of providing the most similar scanpath to individual scanpaths in comparison with other existing algorithms, we conducted some statistical tests as mentioned above. Our results illustrate that the STA algorithm discovers a trending scanpath which is more similar to the individual scanpaths compared to other existing algorithms in 10 out of 12 cases. In addition, the STA algorithm still performs significantly better than some of other existing algorithms in the rest of the cases. Albeit the SPAM algorithm performed better on the AVG page for the searching task, we can suggest that the overall results support the hypothesis which says our STA algorithm provides a trending scanpath which is more similar to the individual scanpaths compared to other algorithms.

7. DISCUSSION

In this paper, we introduce a new concept in clustering scanpaths called *Scanpath Trend Analysis (STA)* which identifies the most commonly followed path by users on a web page in terms of the visual elements of the page. We refer to this path as a trending scanpath. This work differs from other existing work in that it does not only take the visual elements visited by all users, but also takes the visual elements visited by the majority into consideration. Furthermore, it calculates a priority value for the visual elements to locate them in the resulting path. We present an algorithm to do this trend analysis automatically. This algorithm was experimentally validated with an eye tracking study. The study illustrates that this algorithm is able to identify a trending scanpath of multiple users in terms of visual elements of a web page on different web pages for different tasks.

Before we started our experimental evaluation, we hypothesised that when the participants are asked to complete a task by fixating certain visual elements in a particular order, the STA algorithm provides a trending scanpath that includes these visual elements in that order (H1 - Section 5). Since the participants were required to locate some specific parts of the web pages in the searching tasks, we used these tasks to test our hypothesis. Based on the results presented in Section 6.1, we can suggest that our hypothesis is supported with the eye tracking study. For example, on the BBC page, the participants were asked to read the first item of Sport News, included in the visual element R, and then locate the table that shows market data under the Business title, included in the visual element N. Therefore, we are expecting to see that the trending scanpath identified for this page includes RN. The results show that the trending scanpath identified for 38 participants includes RN, so our hypothesis is supported. These trending scanpaths also include some other visual elements. This is also an expected situation because the participants might locate other visual elements while they were searching for their targets and/or listening and understanding the tasks from the researcher. We also observed that large images attract users even though the users do not have to look at them to complete their tasks. For example, the visual element F of the Apple page includes a large image of the iPad mini. Albeit the element is not part of the searching task, it is seen in the trending scanpath on the Apple page for the searching tasks.

The STA algorithm was designed to address the problem of being reductionist. The main reason of being reductionist is losing the visual elements which are fixated by all participants but in a different order [Eraslan et al. 2016a]. Therefore, we investigated whether or not the trending scanpaths identified by the STA algorithm include the visual elements which were fixated by all of the participants (H2 - Section 5). As presented in Section 6.2, the STA algorithm thoroughly addresses this problem. However, if none of the visual elements is fixated by all participants, this can potentially cause some problems in identifying a trending scanpath. In this case, the web page segmentation should also be re-considered. For example, the segments (namely, visual elements) may be enlarged to cover larger areas on web pages.

A trending scanpath is constructed based on individual scanpaths, so the trending scanpath should be similar to the individual scanpaths (H3 - Section 5). Because it is almost impossible to have the same individual scanpaths, the trending scanpath cannot be 100% similar to the individual scanpaths. Hence, the similarities between the trending scanpaths and the individual scanpaths are obviously associated with the similarities between the individual scanpaths. Our experimental results in Section 6.3 suggest that the STA algorithm discovers a trending scanpath which typically has sufficient similarity to the individual scanpaths. Compared to other existing algorithms, our STA algorithm discovers a trending scanpath which is more similar to the individual scanpaths with a single outlier. On the AVG page, the SPAM algorithm [Hejmady and Narayanan 2012] provides a scanpath which is more similar to the individual scanpaths compared to our STA algorithm for the searching task. There might be various reasons because of the VIPS algorithm [Akpınar and Yeşilada 2013]: (1) the VIPS algorithm might not detect the visual elements efficiently, (2) the source code of the web page might not allow the VIPS algorithm to segment the web page well. In order to investigate the effects of the segmentation granularity level of the VIPS algorithm on the AVG page, we increased the granularity level from five to six. As the results were normally distributed, we applied the dependent T-test. According to the statistical test, our STA algorithm performs significantly better than the SPAM algorithm on the AVG page at the granularity level six ($t(35) = 13.319, p < .0001, d = 0.96$). This result suggests that the segmentation granularity level can potentially have effects on the success of the algorithm. This needs to be further investigated in an extended study.

The String-edit algorithm [Josephson and Holmes 2002] was used to determine a distance between two scanpaths and then calculate the similarity of the scanpaths based on the distance. When the distance increases, the similarity decreases. The String-edit algorithm penalises deletions, so the distances are longer for shorter extracted paths. However, it does not mean that the shorter scanpaths are invaluable. The STA algorithm can also provide short scanpaths. For example, it provides GI on the AVG page for the searching task. However, if a longer scanpath can be generated and an algorithm cannot detect that scanpath, it means that the algorithm is likely to lose some elements which could be in the resulting path. For example, although a particular visual element gets attention from all users, it still may not be included in the resulting paths of other existing algorithms.

In the evaluation of the STA algorithm, we used the absolute values of the numbers of fixations and fixation durations on visual elements. These values might be affected by some individual factors. For example, Rayner [1998] states that shorter fixations are made by bilingual readers in their dominant language. In addition, a study conducted by Pan et al. [2004] suggests that fixation durations are affected by gender. One approach is to normalise these values for each user by using a statistical procedure to remove individual variability from fixation durations [Buscher et al. 2012]. The normalised values can then be used to identify a trending scanpath. However, this brings

up some significant questions. The first question can be “*Does the highest duration on a particular visual element for each user show the same processing?*”. For example, one user may fixate the visual element for 2 seconds to only skim its content, next user may fixate the same visual element for 5 seconds to carefully read its content and another user may concentrate on only a particular part of the visual element for 2 seconds. If these durations are maximum on the visual element for the three users, we should not assume that the information processing is the same. Another question can be “*When two users have the same highest duration, does the duration illustrate the same processing?*”. For example, a web page may consist of a number of visual elements where some of them may include more complicated objects in comparison with others. When users focus on the complicated visual elements, they are likely to make longer fixations to process the elements. If one user spends 2 seconds on the complicated element and another user spends 2 seconds on the element with some simple short text, the information processing should not be counted as the same. Nevertheless, the effect of the normalisation approach needs to be further investigated in another study. As the approach should be applied before the analysis, it does not affect the inner workings of the STA algorithm.

Although we investigated the validity of our STA algorithm and its success over existing algorithms, this study still has some limitations. The majority of our participants were students, despite the fact that our eye tracking study included some academic and administrative staff at the two universities in Cyprus and the UK. Because of this, further studies can be conducted by including different groups of users. Forty people participated to our eye tracking study. More participants can be included to extend the evaluation of the STA algorithm. Moreover, six web pages were used in our eye tracking study but none of these web pages is related to social networking sites and search engines which are popular among web users worldwide⁹. Various web pages from different categories can also be included for more generalisable results. Albeit the STA algorithm has been evaluated with some user tasks (See Section 6.1), the algorithm can be re-investigated with more complex and longer tasks in the future. In such cases, users may firstly wander around web pages a lot, then narrow down and then fixate the expected visual elements. Since the STA algorithm ensures that the commonly visited visual elements are included in the resulting scanpath by preserving their overall positions in individual scanpaths, we expect to see that the algorithm also performs well with more complex and longer tasks. Further studies should be conducted to prove our expectation.

7.1. Scanpath Trend Analysis in Use

In summary, the STA algorithm makes an important and useful contribution to behaviour analysis research on the web which can be used for different purposes. In particular, these trending scanpaths can be used to re-engineer web pages based on not only the content but also understanding and predicting of user experiences to make them more accessible for small screen device users and visually disabled users [Akpınar and Yeşilada 2015; Yesilada et al. 2013; Brown et al. 2010]. For example, Figure 3 shows the Apple page with its visual elements, and the trending scanpath identified with the STA algorithm for that page. As can be seen from this figure, the Apple page has many visual elements and it would be faster to download and navigate if we could remove some of its elements and re-order the rest based on user interactions. Therefore, we need to know which elements are most commonly used and in which order. The trending scanpath is here to answer these questions. Specifically, we can remove the footnote items as they are not seen in the trending scanpath, and pro-

⁹<http://www.alexa.com>

vide a single link to access them. This makes the size of the page smaller such that the page can be downloaded faster. We can also re-order the elements based on their positions in the trending scanpath, and therefore the firstly and commonly used elements can be accessed without a lot of scrolling/zooming when the page is accessed from a small screen device or without spending time on unnecessary clutter when the page is accessed by using a screen reader. Particularly, the visual element I is the third element in the trending scanpath, however it can be accessed in approximately 19 seconds (estimated by aDesigner¹⁰) with a screen reader. By re-ordering the elements, we can change the position of the element to provide a quicker access. Furthermore, we can give more emphasis on the repeating elements. For instance, the element E is seen in the trending scanpath three times, thus we can highlight the element when the page is accessed from a small screen device. Existing approaches tend to be reductionist, thus they can lose commonly used elements [Eraslan et al. 2016a]. If their results are used, some of the most commonly used elements may be lost.

These trending scanpaths can also help web designers, engineers, and practitioners to understand how users interact with web pages and evaluate the usability of web pages. Furthermore, it can be useful for website owners to offer some parts of their websites for companies who want to advertise their products. For instance, the website owner can offer different prices for different visual elements based on their positions in the trending scanpath. For example, the website owner can offer higher prices for integrating the advertisements into the firstly visited visual elements because it is typically desired to attract users' attention as soon as possible they visit the website [Janoschka 2004]. Although existing research claims that web users tend to ignore advertisements on web pages called "*Banner Blindness*" [Yesilada et al. 2008], an eye tracking study conducted by Resnick and Albert [2014] suggests that the locations of advertisements are crucial. Therefore, the trending scanpaths have potential to guide where the advertisements can be located on web pages to get users' attention. Since the STA algorithm also highlights the elements which are not trending, it can also contribute to Information Retrieval (IR) research on the web as IR researchers study on web pages to investigate which parts of the pages should not be provided users to improve the retrieval performance [Yu et al. 2003]. Even though this study focuses on user transactions with web pages, it can be applied to other visual media, especially posters and art drawings, by using a different segmentation approach such as a grid-layout [Underwood et al. 2008]. However, as the STA algorithm is mainly designed for analysing user scanpaths by using a number of heuristics solely based on eye tracking data, it may not be extrapolated to other applications.

8. CONCLUDING REMARKS AND FUTURE WORK

This paper presents a novel multi-pass Scanpath Trend Analysis (STA) algorithm to discover trending scanpaths of multiple web users in terms of visual elements of web pages. This algorithm takes eye tracking data and visual elements of web pages. Since we would have trending scanpaths which can be used for further processing of web pages, we use the VIPS algorithm to segment web pages into their visual elements because it relates these visual elements with the underlying source code [Akpınar and Yeşilada 2013]. The STA algorithm then relates eye tracking data to these visual elements to prepare individual scanpaths. It then uses a data-driven approach to identify the trending visual elements in the individual scanpaths. It finally clusters the individual scanpaths to discover the trending scanpaths by using the trending visual elements based on their overall positions in the individual scanpaths.

¹⁰<http://www.eclipse.org/actf/downloads/tools/aDesigner/>

This paper also presents an eye tracking study that was conducted with 40 participants to investigate the validity of the STA algorithm and its success over existing algorithms. This study suggests that the STA algorithm is able to identify trending scanpaths which address the reductionist problem that other existing work has. Our experimental results also show that the STA algorithm generates a trending scanpath which is more similar to the individual scanpaths compared to other existing work.

This study also highlights a number of areas where further studies can be conducted. For example, in this study we had 40 scanpaths to identify a trending scanpath, but we do not know if a smaller set of scanpaths would also be used to identify the same trending scanpath. Similarly, in our studies we have used the VIPS algorithm to generate the visual elements automatically, however we do not know what would be the effect of using other segmentation techniques with different segmentation granularity levels on the STA algorithm. Furthermore, in our previous work, we showed that the task, gender and familiarity factors are likely to affect these trending scanpaths [Eraslan and Yesilada 2015]. We can conduct further experiments to investigate the effects of these factors by using the STA algorithm. In the literature, some researchers discuss the possible cultural effects on eye movements of users. In particular, according to Chua et al. [2005], Chinese users spend less time on focal objects and make more saccades to the background in comparison with American users. In contrast, other researchers did not find any cultural differences between Chinese and American users in their studies [Evans et al. 2009; Rayner et al. 2009]. Therefore, the cultural effects on scanpath trend analysis need to be further investigated in the future.

ACKNOWLEDGMENTS

We would like to thank our participants for their support. We would also like to thank Dr. Orkun Yetkili for his suggestions about statistical tests.

OPEN DATA

The evaluation data including the materials (web pages, information sheet, consent form and questionnaire) and the individual scanpaths in terms of the visual elements of the web pages can be accessed at http://iam-data.cs.manchester.ac.uk/data_files/18. This link also includes the detailed complexity analysis of the most complicated part of the STA algorithm and the full run of the algorithm for nine scanpaths on the Apple page. The Python implementation of the STA algorithm is also provided.

REFERENCES

- Andrew Abbott and Alexandra Hrycak. 1990. Measuring Resemblance in Sequence Data: An Optimal Matching Analysis of Musicians' Careers. *Amer. J. Sociology* 96, 1 (July 1990), 144–185.
- Hamed Ahmadi and Jun Kong. 2012. User-centric Adaptation of Web Information for Small Screens. *Journal of Visual Languages and Computing* 23, 1 (Feb. 2012), 13–28.
- Faisal Ahmed, Yevgen Borodin, Andrii Soviak, Muhammad Islam, I.V. Ramakrishnan, and Terri Hedgpeth. 2012. Accessible Skimming: Faster Screen Reading of Web Pages. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*. ACM, New York, NY, USA, 367–378.
- Elgin Akpınar and Yeliz Yeşilada. 2015. “Old Habits Die Hard!”: Eyetracking Based Experiential Transcoding: A Study with Mobile Users. In *Proceedings of the 12th Web for All Conference*. ACM, New York, NY, USA, Article 12, 5 pages.
- M. Elgin Akpınar and Yeliz Yeşilada. 2013. Vision Based Page Segmentation Algorithm: Extended and Perceived Success. In *Current Trends in Web Engineering (LNCS)*, Quan Z. Sheng and Jesper Kjeldskov (Eds.), Vol. 8295. Springer International Publishing, 238–252.
- Maria Grazia Albanesi, Riccardo Gatti, Marco Porta, and Alice Ravarelli. 2011. Towards Semi-automatic Usability Analysis Through Eye Tracking. In *Proceedings of the 12th International Conference on Computer Systems and Technologies*. ACM, New York, NY, USA, 135–141.

- Aris Anagnostopoulos, Michail Vlachos, Marios Hadjieleftheriou, Eamonn Keogh, and Philip S. Yu. 2006. Global Distance-based Segmentation of Trajectories. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY, USA, 34–43.
- Jay Ayres, Jason Flannick, Johannes Gehrke, and Tomi Yiu. 2002. Sequential PAttern Mining Using a Bitmap Representation. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '02)*. ACM, New York, NY, USA, 429–435.
- Tanja Blaschek, Kuno Kurzhals, Michael Raschke, Michael Burch, Daniel Weiskopf, and Thomas Ertl. 2014. State-of-the-Art of Visualization for Eye Tracking Data. In *EuroVis - STARs*, Rita Borgo, Ross Maciejewski, and Ivan Viola (Eds.). The Eurographics Association.
- Andrew Brown, Caroline Jay, and Simon Harper. 2010. Using qualitative eye-tracking data to inform audio presentation of dynamic Web content. *New Review of Hypermedia and Multimedia* 16, 3 (2010), 281–301.
- Michael Burmester and Marcus Mast. 2010. Repeated Web Page Visits and the Scanpath Theory: A Recurrent Pattern Detection Approach. *Journal of Eye Movement Research* 3, 4 (2010), 1–20.
- Georg Buscher, Andreas Dengel, Ralf Biedert, and Ludger V. Elst. 2012. Attentive Documents: Eye Tracking As Implicit Feedback for Information Retrieval and Beyond. *ACM Transactions on Interactive Intelligent Systems* 1, 2, Article 9 (Jan. 2012), 30 pages.
- Chung-Han Chiang. 2009. *A Genetic Algorithm for the Longest Common Subsequence of Multiple Sequences*. Master's thesis. National Sun Yat-sen University.
- Hannah Faye Chua, Julie E. Boland, and Richard E. Nisbett. 2005. Cultural variation in eye movements during scene perception. *Proceedings of the National Academy of Sciences of the United States of America* 102, 35 (2005), 12629–12633.
- Tim Chuk, Antoni B. Chan, and Janet H. Hsiao. 2014. Understanding eye movements in face recognition using hidden Markov models. *Journal of Vision* 14, 11 (2014), 1–14.
- Jacob Cohen. 1988. *Statistical power analysis for the behavioral sciences* (second ed.). Hillsdale, NJ: Erlbaum.
- Filipe Cristino, Sebastiaan Mathôt, Jan Theeuwes, and Iain D. Gilchrist. 2010. ScanMatch: A novel method for comparing fixation sequences. *Behavior Research Methods* 42, 3 (2010), 692–700.
- Andrew T. Duchowski, Jason Driver, Sheriff Jolaoso, William Tan, Beverly N. Ramey, and Ami Robbins. 2010. Scanpath Comparison Revisited. In *Proceedings of the 2010 Symposium on Eye Tracking Research and Applications*. ACM, New York, NY, USA, 219–226.
- Sukru Eraslan and Yeliz Yesilada. 2015. Patterns in Eyetracking Scanpaths and the Affecting Factors. *Journal of Web Engineering - Special Issue on "Engineering the Web for users, developers and the crowds"* 14, 4&5 (2015), 363–385.
- Sukru Eraslan, Yeliz Yesilada, and Simon Harper. 2013. Understanding Eye Tracking Data for Re-engineering Web Pages. In *Current Trends in Web Engineering*, Quan Z. Sheng and Jesper Kjeldskov (Eds.). LNCS, Vol. 8295. Springer International Publishing, 345–349.
- Sukru Eraslan, Yeliz Yesilada, and Simon Harper. 2014. Identifying Patterns in Eyetracking Scanpaths in Terms of Visual Elements of Web Pages. In *Web Engineering*, Sven Casteleyn, Gustavo Rossi, and Marco Winckler (Eds.). LNCS, Vol. 8541. Springer International Publishing, 163–180.
- Sukru Eraslan, Yeliz Yesilada, and Simon Harper. 2016a. Eye Tracking Scanpath Analysis Techniques on Web Pages: A Survey, Evaluation and Comparison. *Journal of Eye Movement Research* 9, (1):2 (2016), 1–19.
- Sukru Eraslan, Yeliz Yesilada, and Simon Harper. 2016b. What is Trending in Eye Tracking Scanpaths on Web Pages?. In *Proceedings of the Measuring Behavior 2016*, Andrew Spink, Gernot Riedel, Liting Zhou, Lisanne Teekens, Rami Albatal, and Cathal Gurrin (Eds.). Dublin City University, 360–363.
- Kris Evans, Caren M. Rotello, Xingshan Li, and Keith Rayner. 2009. Scene perception and memory revealed by eye movements and receiver-operating characteristic analyses: Does a cultural difference truly exist? *The Quarterly Journal of Experimental Psychology* 62, 2 (February 2009), 276 – 285.
- Brice Follet, Olivier Le Meur, and Thierry Baccino. 2011. New insights into ambient and focal visual fixations using an automatic classification algorithm. *iPerception: Open-access journal of human, animal, and machine perception* 2, 6 (2011), 592–610.
- Philippe Fournier-Viger, Antonio Gomariz, Ted Gueniche, Azadeh Soltani, Cheng-Wei Wu, and Vincent S. Tseng. 2014. SPMF: A Java Open-Source Pattern Mining Library. *Journal of Machine Learning Research* 15 (2014), 3389–3393.
- Joseph H. Goldberg and Jonathan I. Helfman. 2010. Scanpath Clustering and Aggregation. In *Proceedings of the 2010 Symposium on Eye Tracking Research and Applications*. ACM, New York, NY, USA, 227–234.

- Prateek Hejmady and N. Hari Narayanan. 2012. Visual Attention Patterns During Program Debugging with an IDE. In *Proceedings of the Symposium on Eye Tracking Research and Applications*. ACM, New York, NY, USA, 197–200.
- Helene Hembrooke, Matt Feusner, and Geri Gay. 2006. Averaging Scan Patterns and What They Can Tell Us. In *Proceedings of the 2006 symposium on Eye Tracking Research and Applications*. ACM, New York, NY, USA, 41–41.
- John Heminghaus and Andrew T. Duchowski. 2006. iComp: A Tool for Scanpath Visualization and Comparison. In *Proceedings of the 3rd symposium on Applied perception in graphics and visualization*. ACM, New York, NY, USA, 152–152.
- Kenneth Holmqvist, Marcus Nyström, Richard Andersson, Richard Dewhurst, Halszka Jarodzka, and Joost van de Weijer. 2011. *Eye Tracking: A comprehensive guide to methods and measures*. OUP Oxford.
- Kenneth Holmqvist, Marcus Nyström, and Fiona Mulvey. 2012. Eye Tracker Data Quality: What It is and How to Measure It. In *Proceedings of the Symposium on Eye Tracking Research and Applications*. ACM, New York, NY, USA, 45–52.
- Jana Holsanova, Henrik Rahm, and Kenneth Holmqvist. 2006. Entry points and reading paths on newspaper spreads: comparing a semiotic analysis with eye-tracking measurements. *Visual Communication* 5, 1 (February 2006), 65–93.
- Anja Janoschka. 2004. *Web Advertising: New Forms of Communication on the Internet*. John Benjamins Publishing Company.
- Halszka Jarodzka, Kenneth Holmqvist, and Marcus Nyström. 2010. A Vector-based, Multidimensional Scanpath Similarity Measure. In *Proceedings of the 2010 Symposium on Eye-Tracking Research and Applications*. ACM, New York, NY, USA, 211–218.
- Caroline Jay and Andy Brown. 2008. *User Review Document: Results of Initial Sighted and Visually Disabled User Investigations*. Technical Report. University of Manchester.
- Sheree Josephson. 2010. Using Eye Tracking to See How Viewers Process Visual Information in Cyberspace. In *Visualizing the Web: Evaluating Online Design from a Visual Communication Perspective*, Sheree Josephson, Susan B. Barnes, and Mark Lipton (Eds.). Vol. 1. Peter Lang Publishing Inc., New York, Chapter 5, 99–122.
- Sheree Josephson and Michael E. Holmes. 2002. Visual Attention to Repeated Internet Images: Testing the Scanpath Theory on the World Wide Web. In *Proceedings of the 2002 symposium on Eye Tacking Research and Applications*. ACM, New York, NY, USA, 43–49.
- Ziho Kang and Steven J. Landry. 2015. An Eye Movement Analysis Algorithm for a Multielement Target Tracking Task: Maximum Transition-Based Agglomerative Hierarchical Clustering. *IEEE Transactions on Human-Machine Systems* 45, 1 (Feb 2015), 13–24.
- Olivier Le Meur and Thierry Baccino. 2013. Methods for comparing scanpaths and saliency maps: strengths and weaknesses. *Behavior Research Methods* 45, 1 (2013), 251–266.
- Jae-Gil Lee, Jiawei Han, and Kyu-Young Whang. 2007. Trajectory Clustering: A Partition-and-group Framework. In *Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data*. ACM, New York, NY, USA, 593–604.
- Luis A. Leiva and Enrique Vidal. 2013. Warped K-Means: An algorithm to cluster sequentially-distributed data. *Information Sciences* 237 (2013), 196 – 210. Prediction, Control and Diagnosis using Advanced Neural Computations.
- Magnus S. Magnusson. 2000. Discovering Hidden Time Patterns in Behavior: T-patterns and Their Detection. *Behavior Research Methods, Instruments, and Computers* 32, 1 (2000), 93–110.
- Gary Marchionini. 2006. Exploratory Search: From Finding to Understanding. *Commun. ACM* 49, 4 (April 2006), 41–46.
- Marcus Mast and Michael Burmester. 2011. Exposing Repetitive Scanning in Eye Movement Sequences with T-pattern Detection. In *Proceedings IADIS International conference interfaces and human computer interaction*. Rome, Italy, 137–145.
- Sebastiaan Mathôt, Filipe Cristino, Iain D. Gilchrist, and Jan Theeuwes. 2012. A Simple Way to Estimate Similarity Between Pairs of Eye movement Sequences. *Journal of Eye Movement Research* 5, 1 (2012), 1–15.
- Eleni Michailidou. 2010. *ViCRAM: Visual Complexity Rankings and Accessibility Metrics*. Ph.D. Dissertation. University of Manchester.
- Matthew A. Napierala. 2012. What Is the Bonferroni Correction? *AAOS Now - American Academy of Orthopaedic Surgeons* April (2012), 1–3. <http://www.aaos.org/AAOSNow/2012/Apr/research/research7/>

- Saul B. Needleman and Christian D. Wunsch. 1970. A General Method Applicable to the Search for Similarities in the Amino Acid Sequence of Two Proteins. *Journal of Molecular Biology* 48, 3 (28 March 1970), 443–453.
- Julie Pallant. 2007. *SPSS Survival Manual: A Step By Step Guide to Data Analysis Using SPSS version 15* (4th edition ed.). Maidenhead: Open University Press/McGraw-Hill.
- Andrey Tietbohl Palma, Vania Bogorny, Bart Kuijpers, and Luis Otavio Alvares. 2008. A Clustering-based Approach for Discovering Interesting Places in Trajectories. In *Proceedings of the 2008 ACM Symposium on Applied Computing*. ACM, New York, NY, USA, 863–868.
- Bing Pan, Helene A. Hembrooke, Geri K. Gay, Laura A. Granka, Matthew K. Feusner, and Jill K. Newman. 2004. The Determinants of Web Page Viewing Behavior: An Eye-tracking Study. In *Proceedings of the 2004 symposium on Eye Tracking Research and Applications*. ACM, New York, NY, USA, 147–154.
- Alex Poole and Linden J. Ball. 2005. Eye Tracking in Human-Computer Interaction and Usability Research: Current Status and Future. In *Prospects, Chapter in C. Ghaoui (Ed.): Encyclopedia of Human-Computer Interaction*. Pennsylvania: Idea Group, Inc.
- Claudio M. Privitera and Lawrence W. Stark. 2000. Algorithms for Defining Visual Regions-of-Interest: Comparison with Eye Fixations. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22, 9 (Sept. 2000), 970–982.
- Kari-Jouko Räihä. 2010. Some Applications of String Algorithms in Human-Computer Interaction. In *Algorithms and Applications*, Tapio Elomaa, Heikki Mannila, and Pekka Orponen (Eds.). Lecture Notes in Computer Science, Vol. 6060. Springer Berlin Heidelberg, 196–209.
- Kari-Jouko Räihä, Anne Aula, Päivi Majaranta, Harri Rantala, and Kimmo Koivunen. 2005. Static Visualization of Temporal Eye-Tracking Data. In *Human-Computer Interaction - INTERACT 2005*, MariaFrancesca Costabile and Fabio Patern (Eds.). Lecture Notes in Computer Science, Vol. 3585. Springer Berlin Heidelberg, 946–949.
- Keith Rayner. 1998. Eye Movements in Reading and Information Processing: 20 Years of Research. *Psychological Bulletin* 124 (1998), 372–422.
- Keith Rayner, Monica S. Castelhana, and Jinmian Yang. 2009. Eye Movements When Looking at Unusual/Weird Scenes: Are There Cultural Differences? *Journal of Experimental Psychology: Learning, Memory, and Cognition* 35, 1 (2009), 254 – 259.
- Marc Resnick and William Albert. 2014. The Impact of Advertising Location and User Task on the Emergence of Banner Ad Blindness: An Eye-Tracking Study. *International Journal of Human-Computer Interaction* 30, 3 (2014), 206–219.
- Anthony Santella and Doug DeCarlo. 2004. Robust Clustering of Eye Movement Recordings for Quantification of Visual Interest. In *Proceedings of the 2004 symposium on Eye Tracking Research and Applications*. ACM, New York, NY, USA, 27–34.
- Alistair Sutcliffe and Abdallah Namoun. 2012. Predicting User Attention in Complex Web Pages. *Behaviour & Information Technology* 31, 7 (July 2012), 679–695.
- Haruhiko Takeuchi and Yoshiko Habuchi. 2007. A Quantitative Method for Analyzing Scan Path Data Obtained by Eye Tracker. In *IEEE Symposium on Computational Intelligence and Data Mining*. 283–286.
- Tobii Technology AB. 2011a. *Accuracy and precision, Test report, Tobii T60 Eye tracker*. Technical Report Methodology/Software version: 2.1.1. Tobii Technology AB.
- Tobii Technology AB. 2011b. *Tobii T60 & T120 Eye Tracker User Manual*. Technical Report Revision 4. Tobii Technology AB.
- Geoffrey Underwood, Katherine Humphrey, and Tom Foulsham. 2008. Knowledge-Based Patterns of Remembering: Eye Movement Scanpaths Reflect Domain Experience. In *HCI and Usability for Education and Work*, Andreas Holzinger (Ed.). LNCS, Vol. 5298. Springer Berlin Heidelberg, 125–144.
- Boris M Velichkovsky, Alexandra Rothert, Mathias Kopf, Sascha M Dornhöfer, and Markus Joos. 2002. Towards an express-diagnostics for level of processing and hazard perception. *Transportation Research Part F: Traffic Psychology and Behaviour* 5, 2 (2002), 145 – 156.
- Julia M. West, Anne R. Haake, Evelyn P. Rozanski, and Keith S. Karn. 2006. eyePatterns: Software for Identifying Patterns and Similarities Across Fixation Sequences. In *Proceedings of the 2006 symposium on Eye Tracking Research and Applications*. ACM, New York, NY, USA, 149–154.
- Yeliz Yesilada, Simon Harper, and Sukru Eraslan. 2013. Experiential transcoding: An EyeTracking approach. In *Proceedings of the 10th International Cross-Disciplinary Conference on Web Accessibility*. ACM, 30.
- Yeliz Yesilada, Caroline Jay, Robert Stevens, and Simon Harper. 2008. Validating the Use and Role of Visual Elements of Web Pages in Navigation with an Eye-tracking Study. In *Proceedings of the 17th international conference on World Wide Web (WWW '08)*. ACM, New York, NY, USA, 11–20.

Shipeng Yu, Deng Cai, Ji-Rong Wen, and Wei-Ying Ma. 2003. Improving Pseudo-relevance Feedback in Web Information Retrieval Using Web Page Segmentation. In *Proceedings of the 12th International Conference on World Wide Web (WWW '03)*. ACM, New York, NY, USA, 11–18.