UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

DEVELOPMENT OF METHODOLOGIES TO ANALYZE AND VISUALIZE AIR TRAFFIC CONTROLLERS' VISUAL SCANNING STRATEGIES

A DISSERTATION

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

Degree of

DOCTOR OF PHILOSOPHY

By

SAPTARSHI MANDAL
Norman, Oklahoma
2020

DEVELOPMENT OF METHODOLOGIES TO ANALYZE AND VISUALIZE AIR TRAFFIC CONTROLLERS' VISUAL SCANNING STRATEGIES

A DISSERTATION APPROVED FOR THE
SCHOOL OF INDUSTRIAL AND SYSTEMS ENGINEERING

BY THE COMMITTEE CONSISTING OF

Dr. Ziho Kang, Chair

Dr. Randa Shehab

Dr. Shivakumar Raman

Dr. Theodore Trafalis

Dr. Hairong Song

Acknowledgements

First, I would like to dedicate this dissertation to my mother, Sarbani Mandal, father, Kartick Chandra Mandal, and my sister, Tanni Mandal for their unending love and support. Second, Anamika, thank you for being the better half, literally. Thank you for helping me to overcome many challenges, and you made me a better person. Third, my eternal gratitude to my advisor Dr. Ziho Kang for his constant guidance and support during my graduate studies. I know for sure that your critical attention to details, extensive feedback, and very high work standards have made me a better researcher. Thank you very much for giving the freedom to pursue my own research ideas. I am indebted to you for your motivation and encouragement.

Next, my heartfelt gratitude to my Lab office mate and friend, Dr. Salem Naeeri, for letting me understand what it means to be courageous, persistent, and inclusive in the truest sense, and also for the awesome home-made foods. Ricardo, thanks for providing me with the enthusiasm and energy to explore new domains. Additionally, I'm grateful to Dr. Jerry Crutchfield at the Federal Aviation Administration for providing research opportunities throughout my Ph.D. study.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

This chapter provides the overall organization of the thesis. In this thesis, three different research questions are addressed: (1) how to represent the change in the visual scanning strategy for a dynamic target tracking task; (2) how to simplify visual scanpath representation for a dynamic target tracking task; and (3) how to evaluate the similarity of visual scanning strategies for a dynamic target tracking task.

Chapter 2 addresses the research question of how to represent the evolution of visual scanning strategy for a dynamic target tracking task. In detail, the dynamic directed weighted network (DWN) framework, an extension of the directed weighted network model developed during my masters' research, is adapted to analyze the eye movements of air traffic controllers. In addition, the DWN is used to represent the evolution of visual scanning strategies over time. Furthermore, three dynamic DWN-based measures, i.e. indegree, closeness, and betweenness, were adapted to evaluate target importance and flow of visual attention across time. Lastly, two different visualization approaches, namely the dot plot and the box plot were developed to represent the time evolution of the above-mentioned three measures. The dot plot represents the evolution of the targets that were visually attended over time-intervals, whereas the bar plot enables the comparison of the targets that were visually attended across several participants.

Chapter 3 contains two novel visualization approaches, the order plot and the norm-cube plot, which are built upon the dynamic network framework presented in chapter 2. The order plot visualizes the relative importance of several dynamic targets based on the visual scanning strategy employed by an observer. Moreover, the order plot enables the

representation of the multiple observer's data simultaneously facilitating comparisons of targets that were visually attended among the observers. The norm-cube plot allows the visualization of the three importance measures (i.e. indegree, closeness, and betweenness) simultaneously.

Chapter 4 addresses the research question of how to simplify the visual scanpath representation for a dynamic target tracking task. A novel framework is developed to simplify the representation of the visual scanpaths for a dynamic object tracking and control task (i.e. air traffic control task). The framework first implements a spatial-temporal clustering mechanism to aggregate spatially and temporally neighboring eye fixations, and a time-ordered color scheme is implemented to enhance the visualization of the simplified visual scanpaths. Furthermore, the proposed framework provides two measures, first, to evaluate the amount of the simplification achieved in terms of reduction in constituent eye fixations, and second, the amount of abstraction in terms of shape similarity between the raw visual scanpath and its simplified counterpart. These two developed measures provide the decision-maker to consider better understand the overall visual attentions and provides the researcher with the freedom to choose the parameter values when applying the clustering mechanism.

Chapter 5 addresses the research question of how to evaluate the similarity of visual scanning strategies for a dynamic target tracking task. A novel framework is introduced to evaluate the similarity of scanpath sequences that allows finding the groups of highly similar scanpath sequences. The developed framework consists of two major steps. First, an *n*-gram based distance measure is developed to evaluate the similarity between scanpath sequences. Second, modularity-based network community detection

method is implemented to find groups of highly similar scanpaths. To evaluate the efficacy of the framework, the introduced approach was applied to air traffic control operations, especially when the air traffic controllers issued clearance commands such as clear to land, clear to take-off, and hold short. The results show promise on better analyzing visual scanning strategies of air traffic controllers.

Research Contribution

The thesis includes four major contributions as follows. First, a dynamic network-based framework is developed to model and visualize the dynamic nature of the visual scanning strategies for a dynamic target tracking task. Second, two novel visualization methods are proposed, order plot and norm-cube plot, that shows which targets were more visually attended compared to other other targets based on three measures derived from the network framework. Third, a the spatial-temporal clustering has been adapted to simplify the representation of the visual scanpaths for a dynamic target tracking task. Fourth, *n*-gram and network-based cluster detection methods were adapted to create clusters of similar visual scanpaths.

Chapter 2: Dynamic directed weighted networks and related visualizations.

NOTE: The contents of this chapter has already been published in "Mandal, S., & Kang, Z. (2018). Using Eye Movement Data Visualization to Enhance Training of Air Traffic Controllers : A Dynamic Network Approach. *Journal of Eye Movement Research*, *11*(4)." Therefore, the contents are directly taken from the published manuscript.

Abstract

The Federal Aviation Administration (FAA) forecasted a substantial increase in the US air traffic volume creating a high demand in Air Traffic Control Specialists (ATCSs). Training times and passing rates for ATCSs might be improved if expert ATCSs' eye movement (EM) characteristics can be utilized to support effective training. However, effective EM visualization is difficult for a dynamic task (e.g. aircraft conflict detection and mitigation) that includes interrogating multi-element targets that are dynamically moving, appearing, disappearing, and overlapping within a display. To address the issues, a dynamic network-based approach is introduced that integrates adapted visualizations (i.e. time- frame networks and normalized dot/bar plots) with measures used in network science (i.e. indegree, closeness, and betweenness) to provide in-depth EM analysis. The proposed approach was applied in an aircraft conflict task using a high-fidelity simulator; employing the use of veteran ATCSs and pseudo pilots. Results show that ATCSs' visual attention to multi-element dynamic targets can be effectively interpreted and supported through multiple evidences obtained from the various visualization and associated measures. In addition, we discovered that fewer eye fixation numbers or shorter eye fixation durations

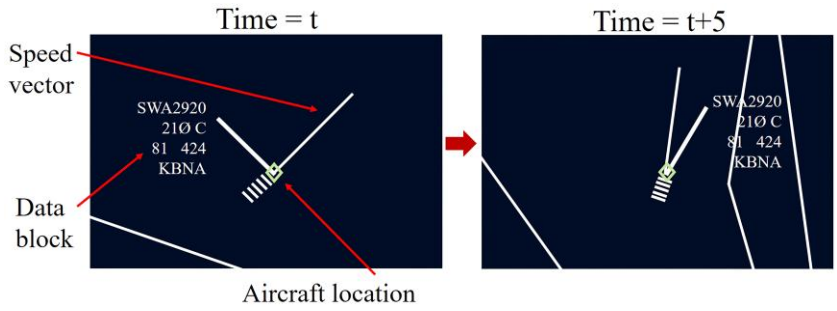
on a target may not necessarily indicate the target is less important when analyzing the flow of visual attention within a network. The results show promise in cohesively analyzing and visualizing various eye movement characteristics to better support training.

2.1 Introduction

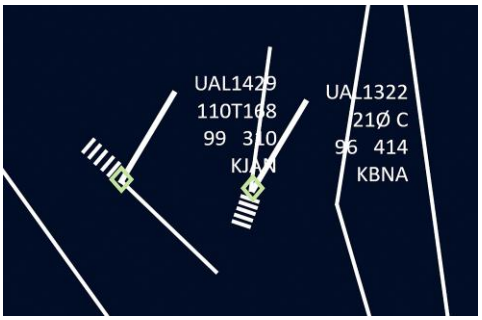
The Federal Aviation Administration (FAA) forecasted a 1.4% annual increase in the US air traffic volume; from currently 43.2 million aircraft to 60.3 million by 2040 (FAA, 2018). However, the currently available number of expert air traffic control specialists (ATCSs) might not be sufficient to handle the anticipated increase of air traffic volume. Additionally, the current training completion time of the air traffic controllers takes many years of intensive training (Hampton, 2016). Therefore, the FAA has been trying to find ways to efficiently train the FAA Academy candidates.

One of the critical tasks of ATCSs is to detect and mitigate possible aircraft conflicts (i.e. possible collisions) through visually scanning the radar screen. The ATCSs look for conflicting situations between aircraft pairs (or groups) to resolve it and guide them to their destination in a safe/timely manner. Thus, the ATCS's task involves a significant amount of visual scanning of the radar display and, subsequently, cognitive processing of the observed information to take necessary actions. Eye-mind hypothesis (Just & Carpenter, 1976) showed there exists high correlation between the eye movement (EM) data and the cognitive process of an observer. Kang and Landry (2014) demonstrated that exposing novice controllers to the visual scanpath of the expert ATCSs improved their overall scanning efficiency by reducing their false positive cases of conflict detection among aircraft. On Similar lines, Rudi, Kiefer and Raubal (2018) demonstrated that visualization of EM data of pilot's working in a cockpit might prove useful for flight

instruction purposes. Therefore, if we could effectively analyze, visualize, and interpret experts' eye movement characteristics, we might be able to use those findings to train the candidates or novices.



(a) Aircraft changing their shape on the radar display



(b) Overlapping aircraft on the radar display

Figure 1: Dynamic aspects of aircraft representation on the radar display of an ATCS: (a) Location of the data blocks changes relative to the aircraft location. (b) Two overlapping aircraft.

Effective analysis of ATCSs' visual scanning process is challenging. The Radar display has, a large number of dynamic targets (i.e. aircraft on radar display) which have dynamic properties (e.g. radar representation of an aircraft can change their shape and position with time). Figure 1 represents the dynamic aspect of the radar representation of aircraft.

Visualization of ATCSs' EM data includes two steps. The first step involves developing a time-ordered mapping between the eye fixations (EFs) and the aircraft on the display. The second step consists of characterization of the developed scanpath sequences.

To develop the mapping function in case of moving and overlapping targets, Dynamic Areas of Interest (AOIs) can be created which are dynamic convex boundaries that fits the moving targets and considers the visual angle accuracy (Kang & Bass, 2014; Kang, Mandal, Crutchfield, Millan, & McClung, 2016; Papenmeier & Huff, 2010). However, as the number of targets increases, visualization of the scanpath sequences becomes challenging using the widely used visualization methods such as point based and AOI based methods (Blascheck, Kurzhals, Raschke, Burch, Weiskopf, & Ertl, 2014) (explained below in Background section), since these methods might create visual clutter when numerous targets are visualized. In addition, air traffic has a dynamic nature, meaning that an aircraft can dynamically move within the radar display for a certain amount of time, and/or two or more aircraft can overlap on one another. As a result, any aircraft's relevance from the visual scanning point of view evolves with time. The prevalent visualization methods are unable to handle this dynamic aspect of the ATCS's visual scanning process.

In addition, other existing pivotal researches (Blascheck, Raschke, & Ertl, 2013; Burch, Beck, Raschke, Blascheck, & Weiskopf, 2014; Goldberg & Kotval, 1999) focus more on EF numbers and EF durations or simpler forms of scanpaths (explained in Background section); however, for a dynamic task such as air traffic control task, how the multiple targets (e.g. aircraft) are observed as a network (using the various EM transition characteristics) can also be important, meaning that even if there were fewer number of EFs or less durations, a target can be considered important if it plays a crucial role in the EM flow among multiple targets or acts as a bridge between two disconnected groups of targets. Furthermore, if we only consider the number of EFs on the aircraft, it might produce an incorrect interpretation about the important aircraft. For example, consider a

case when an aircraft has just entered the radar display. There is high chance that this new entrant aircraft might receive a substantial amount of EF duration, as the ATCSs might want to know about its destination, altitude, and other details; however, this new aircraft might not be important in terms of conflict resolution with the already existing set of aircraft on the display. As a result, the ATCSs will not fixate again on this aircraft, rendering it unimportant in terms of the overall scanning strategy. Zhang, Ren and Wu (2014) provided valuable findings using static networks in the air traffic control domain, but the dynamic aspects and the issues raised above were not addressed.

Therefore, we need an improved analysis framework which will help us (1) develop visualization methods which can represent the EM data with a large number of targets with less visual clutter, (2) find measures that can accommodate the dynamic aspects of the moving targets, and (3) integrate the visualizations and measures for effective analyses and interpretations.

In this paper, we provide several approaches to address the issues raised above. First is to adapt the dynamic network (DNet) approach (Burch et al., 2014) and modify its structural components for visualizing the EM data of the ATCSs. A DNet is a collection of time-ordered static networks. The DNet visualization enables easy handling of a large number of targets, thereby reducing visual clutter. Being a collection of several networks, the DNet can easily represent the evolution of a target's importance over time and the dynamics of visual scanning characteristics. The adapted DNet is aligned with three "vertex importance measures" such as "indegree," "closeness," and "betweenness" (Freeman, 1978) to better determine important targets. Furthermore, two types of normalization procedures (i.e. percent normalization and distance normalization) are introduced that

calculate the relative amount of visual attention given to a target in comparison to the maximum values obtained for a specific task. Finally, we adapted the dot plots and bar plots to either better represent the evolution of the important targets or compare the vertex importance measures among the participants. Note that we will replace the term "vertex" with "target" or "AOI" for easier understanding.

2.2 Background

2.2.1 Eye movement (EM) visualizations: Point based, area of interest (AOI) based, and hybrid

Blascheck et al. (2014) have categorized the various EM visualization methods into point based, AOI based, and hybrid visualizations. Summaries and issues are as follows.

Existing point based visualization methods, e.g. timeline visualization (Kurzhals, Heimerl, & Weiskopf, 2014), scanpath visualization (Goldberg & Helfman, 2010), attention maps (Kurzhals & Weiskopf, 2013), space-time cubes (Kurzhals & Weiskopf, 2013) represent the time-ordered horizontal and vertical coordinates of the EFs occurring on the display. These methods are effective on visualizing the exact EF locations to unravel important regions (in absence of predefined targets) when given static stimuli. However, due to the visual angle error of the eye trackers, it is challenging to map the EFs with small and dynamic multi-element targets making it difficult to apply the point based methods (Mandal, Kang, & Millan, 2016). In addition, our interest is in investigating which moving targets were focused upon rather than the physically fixed area within a display.

On the other hand, existing AOI based visualization methods allows EM analysis based on either pre-defined region or target on the display. The AOI based methods have been categorized into timeline and relational AOI visualizations (Blascheck et al., 2014).

Timeline AOI visualizations such as parallel scanpath (Raschke, Chen, & Ertl, 2012), scarf plot (Kurzhals, Fisher, Burch, & Weiskopf, 2015), and AOI river plot (Burch, Kull, & Weiskopf., 2013) focus on developing effective methods to represent the AOIs that have been fixated upon at various time intervals. However, these methods are challenging to apply for long duration tasks having large number of targets (e.g. twenty or more targets) and frequent EF transitions between them (e.g. air traffic control task).

Relational AOI visualization methods are more appropriate to handle the issues raised above through visualizations using circular heat map transition diagram (Blascheck et al., 2013), transition matrix (Goldberg & Kotval, 1999) and network visualization (Burch et al., 2014). These methods visualize the aggregated EM data by showing the relationship that exists between the AOIS in terms of the EF transitions between them, unlike the timeline approaches. Relational AOI based approaches do not represent the physical location of the AOIs on the display. In detail, in circular heat map visualization (Blascheck et al., 2013), AOIs are represented as segments of a circular layout (using different colors and sizes) and the EF transitions are shown by directed arrows between the circular segments. Transition matrix visualization (Goldberg & Kotval, 1999) represents the EF transitions among the AOIs in a tabular fashion. The most appropriate approach to address the issues of a dynamic task is through the network visualization that shows the AOIs as vertices and the EF transitions between AOIs as the directed edges between the vertices of a network (Burch et al., 2014; Holmqvist, Holsanova, Barthelson, & Lundqvist, 2003; Mandal et al., 2016; Tory, Atkins, Kirkpatrick, Nicolaou, & Yang, 2005).

However, if we try to apply the relational AOI visualization methods, we often run into possible visual clutter issues if there are large number of targets and it can be difficult to

represent all the EM characteristics using the existing network approach. The following subsections 2.2. and 2.3 provide summaries of the DNet mathematical framework and how various EM network characteristics can be integrated based on time intervals.

2.2.2 Mathematical framework of DNet

A DNet is as a sequence of static networks (also called networks), where each constituent network is associated with a time interval (Beck, Burch, Diehl, Weiskopf, 2014). If the total time duration of the collected EM data is divided into T time intervals, then a DNet representing such a data is written as $DynN = \{N_1, N_2, .. N_t, .., N_T\}$, where N_t is the network for time interval t, where $t = 1,2,..,T$.

A network N_t is written as $N_t = (V_t, E_t, M_t)$, where, V_t is the set of vertices (AOIs for the present study), E_t is the set of edges (EF transitions for the present study) between the vertices, and M_t is the adjacency matrix which contains all edge weights (amount of EF transitions between AOI pairs).

The set of vertices is written as $V_t = (v_1, v_2, \dots, v_{m_t})$, where m_t is the number of vertices for time interval t. A network can either have directed or undirected edges, although for EM visualization we only consider directed edges. The set E_t consists of ordered pairs of vertices (v_i, v_j) showing that there exists a directed edge from the vertex v_i towards vertex v_j. Thus, $E_t = \{e_{ij}(t) | v_i, v_j \in V_t, i \neq j\}$. Lastly, the adjacency matrix is written as $M_t = [w_{ij}(t)]_{m_t \times m_t}$, where, $w_{ij}(t)$ is the weight of the edge $e_{ij}(t)$ (Newman, 2004; Newman, 2010).

2.2.3 DNet for EM visualization

Beck et al. (2014) provided an exhaustive list of various DNet visualization approaches representing EM data. Depending on the representation of the time variable,

various visualization approaches have been categorized into two groups: Animation, and timeline visualization. Animation visualization refers to representing a DNet as an animated sequence of networks. Timeline visualization refers to representing a DNet as a sequence of networks in a single image showing the complete sequence of interactions between the targets. In the present work, we have applied the node-link based timeline approach for representing the DNet, because this visualization helps to preserve the mental map and reduces the cognitive load of the observer (Beck, Burch, & Diehl, 2013).

As noted by various researchers (Archambault & Purchase, 2013; Ghani, Elmqvist, & Yi, 2012; Purchase, Hoggan, & Görg, 2006), preserving the mental map (i.e. the abstract structural information layout about a network's elements that an analyst develops in their mind as they visually scan the visualization) helps in tracing the change in vertex properties and edge paths across different time intervals. Additionally, the timeline visualization, using the node-link approach, provides an intuitive and efficient framework for analyzing the change of states of multiple vertices over time (Saraiya, Lee, & North, 2005; von Landesberger et al., 2011).

However, the existing DNet approach only uses the number and duration of EFs on the AOI to measures the AOI importance. As noted in the introduction, these two raw measures may lead to misleading results in case of dynamic targets. Therefore, it is required to consider other target importance measures to address the highlighted issue.

The next section discusses the three target importance measures (from the network science domain) that can be adapted for analysing AOI importance for dynamic scenarios.

2.2.4 Target (or AOI) importance measures

The three most popular target importance measures in a given network are indegree, closeness, and betweenness (Freeman, 1978; Newman, 2004; Opsahl, Agneessens, & Skvoretz, 2010). Mandal et al. (2016) have shown some possibilities in applying the above-mentioned measures to build a basic foundation for the proposed approach in this article. It is noted that we introduce the "time" element ("t") within the three vertex importance measures to consider the dynamicity.

Indegree of a vertex is defined as the sum of all incoming weights to it from all other vertices in the network. For the present study, incoming weights can be interpreted as the incoming EF transitions to a given AOI. Thus, indegree for the j^{th} AOI is given as $I_j = \sum_{k=1}^{m} w_{kj}$ (Newman, 2004), where, w_{kj} is the number of EF transitions from the k^{th}AOI to the j^{th}AOI, and m is the total number of AOIs on the display.

We should note that the indegree measure, shown above, is static in nature. As a result, we modified it to develop the dynamic analogous, where the indegree for an AOI is defined for each of the time interval considered in the DNet framework. Thus, the modified indegree measure for the j^{th}AOI for time interval t is calculated as:

$$I_j(t) = \sum_{\substack{k=1 \\ k \neq j}}^{m_t} w_{kj}(t) \quad (1)$$

Where, $w_{kj}(t)$ is the number of EF transitions coming for the k^{th}AOI to j^{th}AOI and m_t is the number of unique AOIs in the AOI fixation sequence for time interval t. Large indegree value suggests higher importance for an AOI, as it received large number of EFs. Thus, indegree can be interpreted as a measure of direct attention received by an AOI.

However, indegree measure only considers the local structure (direct EF transitions) around a vertex but neglects the global structure of the network (Borgatti, 2005; Opsahl et al., 2010). This issue is addressed by the two measures named "closeness" and "betweenness."

Note that the network science (or graph theory) includes the concept of "outdegree" however, the indegree values and outdegree values are always the same within the EM network since there is no possibility of an inward single EM transition dividing into two or more outward transitions.

Before we understand the two measures, we need to define the concept of distance between AOIs in a network visualizing EM data. In the present case, we define the distance from one AOI (e.g. 'A') to another AOI (e.g. 'B') as the inverse of the number of EF transitions from AOI A towards B. Thus, a large number of EF transitions between AOIs results in a smaller distance between them in terms of the visual scanning strategy.

The closeness of a vertex measures its distance from all other vertices in the network. Thus, higher closeness value for a given vertex means it is easier to access any part of the network from it. In the present study, higher accessibility of an AOI can be interpreted as a greater association (both direct and indirect) with other AOIs in the network. High closeness value suggests that the AOI lies in the central location in terms of the observer's visual scanning strategy. The closeness for the j^{th} AOI is given as $C_j = \sum_{k=1}^{m}(1/d_{jk}{}^{*})$ (Opsahl et al., 2010), where, $d_{jk}{}^{*}$ is the minimum distance from the j^{th}AOI to the k^{th}AOI (if multiple paths exist) and m is the total number of AOIs on the display. Like the previous indegree measure case, the static closeness measure is also modified to

develop its dynamic analogous. Thus, the modified closeness measure for the j^{th} AOI for time interval t is defined as follows:

$$C_j(t) = \sum_{\substack{k=1 \\ k \neq j}}^{m_t} \frac{1}{{d_{jk}}^*(t)} \quad (2)$$

Where, ${d_{jk}}^*(t)$ is the minimum distance from the j^{th}AOI to the k^{th}AOI (if multiple paths exist) and m_t is the number of unique AOIs in the AOI fixation sequence for time interval t.

In a dynamic scenario, there are instances where due to visual scanning strategy of the observer, an AOI despite receiving small amount of direct attention (low indegree measure) and being present in a non-central location (low closeness value) can still play a significant role by connecting (acting as a bridge between) two groups of AOIs. Such an AOI plays a crucial role in controlling the flow of attention among the other AOIs on the display, and this aspect is measured through the concept of betweenness explained below.

Betweenness for the j^{th} AOI is defined as $B_j = \sum_{l=1}^{m} \sum_{k=1, j \neq l \neq k}^{m} \left(SP_{kl}^j / SP_{kl}\right)$ (Opsahl et al., 2010), where SP_{kl} represents the total number of shortest paths (if multiple paths exist) from the k^{th}AOI to the l^{th}AOI, and SP_{kl}^j represents the number of such shortest paths that pass through the j^{th}AOI. Thus, the modified betweenness measure for the j^{th} AOI for time interval t is defined as follows:

$$B_j(t) = \sum_{k=1}^{m_t} \sum_{\substack{l=1 \\ j \neq l \neq k}}^{m_t} \frac{SP_{kl}^j(t)}{SP_{kl}(t)} \quad (3)$$

Where, $SP_{kl}(t)$ represents the total number of shortest paths (if multiple paths exist) from the k^{th}AOI to the l^{th}AOI, and $SP_{kl}^j(t)$ represents the number of such shortest paths which pass through the j^{th}AOI for the time interval t.

2.3. Proposed approach

Figure 2 represents the various steps in the proposed methodology for analyzing ATCS's EM data.

2.3.1 STEP 1. Collect observer's EM and targets' location data

The input for the first step is the simulation experimental data. As output for this step, two types of data are obtained: (a) EM data, that consists of horizontal and vertical coordinates of the EF and its associated fixation duration, and (b) target location data, that consists of the pixel coordinates of the various targets on the display.

2.3.2 STEP 2. Divide the experiment time into several smaller time intervals

The input for step 2 is the time duration (in minutes or hours) for which the experiment has been conducted, and the output are several equal (unequal) sized time intervals which sum to the experimental time duration. For example, consider that the total experiment duration is divided into four equal time intervals (i.e. $T = 4$, see section 2.2). Note that the time intervals can be chosen based on the task characteristics or the researcher's judgment (i.e. fixed or event-based time intervals).

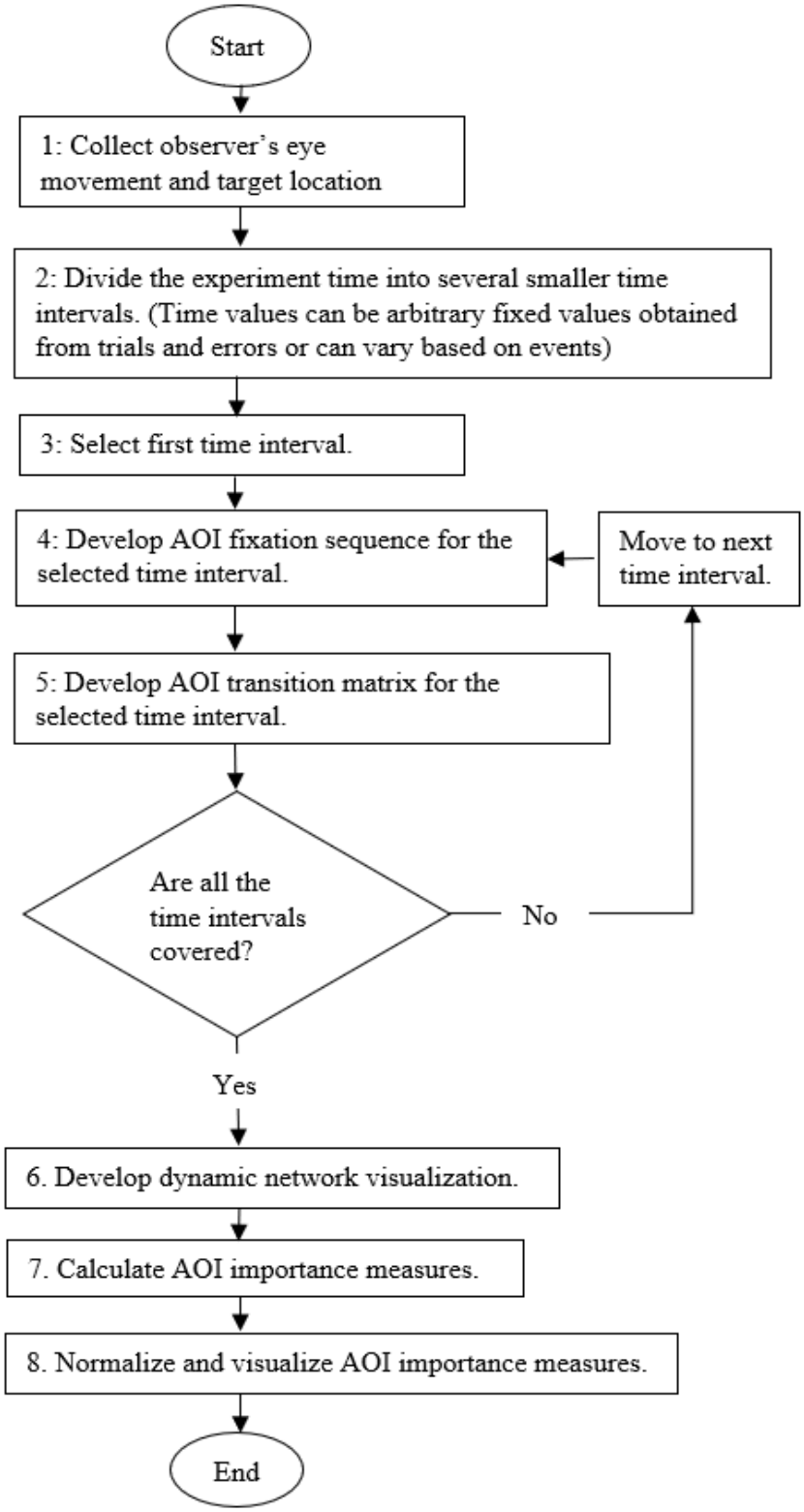


Figure 2: Flowchart showing the various steps of the proposed methodology for analyzing time interval-based EM data.

2.3.3 STEP 3. Select first time interval

The inputs for this step are the various time intervals obtained in the previous step 2. The intervals are indexed and arranged in a time-order sequence. In this step, we start with the first time interval to initiate developing the AOI fixation sequence.

2.3.4 STEP 4. Develop AOI fixation sequence (i.e. scanpath sequence) for the selected time interval

As input, step 4 receives three things, the time interval selected in step 3, and the EM and target location data for this time interval. As output from this step, we obtain the AOI fixation sequence for the time interval considered. The size of the AOI fixation sequence is directly proportional to the number of EFs in the time interval. The AOI fixation sequence is created for the selected time interval by adapting the approach suggested by Kang et al. (2016). Creating the AOI fixation sequence involves mapping the EFs with the AOIs. Only those EFs falling within any of the AOI boundaries are considered for AOI fixation sequence development, else they are ignored. AOIs are coded by assigning uppercase letters followed by lowercase alphabets (i.e. A, B, …, a, b). The developed AOI fixation sequence is the collapsed form of a raw AOI fixation sequence. In the collapsed form of the sequence, multiple consecutive fixations of the same AOI is collapsed to a single fixation case (e.g. AAA is collapsed into A). Thus, a raw AOI fixation sequence AABCC is collapsed to ABC. In addition, an overlapping AOI case is shown in parenthesis with individual constituent AOIs separated by a semi-colon. For example, if an EF falls within the overlapped region of AOIs A and B, it is represented as (A; B). After mapping all the EFs to the AOIs, we obtain a time-ordered sequence of AOIs, that shows which AOIs were fixated and in which order. Table 1 shows a sample AOI fixation sequence for four time intervals in a hypothetical scenario with seven AOIs.

2.3.5 STEP 5. Develop AOI transition matrix for the selected time interval

Step 5 receives the AOI fixation sequence developed in step 4 as input. Afterwards, the AOI fixation sequence is transformed, as per the approach suggested by Noton and Stark (1971), for developing the associated AOI transition matrix, which is the output of this step. The size of the transition matrix depends on the number of unique AOI states (single and overlapped both) in the fixation sequence. The AOI transition matrix shows, in a tabular manner, how many transitions have occurred between various AOIs pairs. For example, Table 2 represents the AOI transition matrix associated with the AOI fixation sequence for time interval 1 in Table 1. The sequence shows there are three EF transitions from AOI A to B highlighted in grey within Table 2. A different AOI transition matrix is required for each time interval; thus, before moving to the next step, we need to create the associated AOI transition matrix for each time interval.

Table 1: Samples AOI fixation sequences having overlaps

Time interval	AOI fixation sequence
1	ABABABEBAEAEACACAEA
2	EAEAEAEABABCBC(D; G)C(D;G)
3	C(D;G)C(D;G)C(D;G)CEAEAEFEF
4	EFEFEFCFC(D;G)C(D;G)FC(D;G)

Table 2. AOI transition matrix developed from the AOI fixation sequence for time interval 1 within Table 1**.**

From AOI	To AOI			
	A	**B**	**C**	**E**
A	0	3	2	3
B	3	0	0	1
C	2	0	0	0
E	3	1	0	0

2.3.6 STEP 6. Develop DNet visualization

The inputs for step 6 are the AOI transition matrix for all the time intervals obtained in step 2. As output from this step, we obtain the DNet representation of the EM data. Developing the DNet involves two steps. First, developing a static network for each of the time intervals considered in STEP 2. Second, arranging the static networks in a time-based order to visualize the DNet. The details of both the steps are given below.

2.3.6.1 Develop static network for each time interval

The static network is developed by adopting the design principles suggested by Mandal et al. (2016). Thus, a network's vertex size is drawn proportional to the number of EFs received by the corresponding AOI and the vertex's color is based on a sequential multihued color scale, where red color means high EF duration and yellow color means low EF duration occurring on the associated AOI. The thickness of an edge between a vertex pair is proportional to the number of EF transitions occurring between the vertices

in the edge's direction. Therefore, for the example shown in Table 1, we have four time intervals, thus the DNet is written as $DynN = \{N_1, N_2, N_3, N_4\}$, where N_i is the static network for time interval i ($i = 1, \ldots, 4$).

2.3.6.2 Visualize the DNet

Before visualizing the DNet, the vertices of the component static networks are to be arranged in a specific layout for mental map preservation. We have used the rectangular grid layout for this purpose; in which, we start from the left bottom corner and move towards the right, ending at the top right corner. We sort the AOI positions using the natural ordering of English letters and putting uppercase AOI groups first and then lowercase AOI groups. The single AOIs are followed by overlapped AOI cases in the grid layout. The overlapped AOI cases are arranged in an increasing order of constituent AOI numbers, i.e. an overlapped AOI case with two AOIs comes before a case with three AOIs. For example, overlapped AOI (A;B) comes before (A;B;C), which comes before (A;B;C;D), and so on.

Consider an AOI fixation sequence that has n unique AOI states (including both single and overlapped AOIs). Therefore, the number of columns (number of AOIs in a single row) in the grid layout will be the smallest integer greater equal to $\sqrt{n}$. Figure 3 represents a sample ordering scheme of six AOIs in a rectangular grid layout.

Once the component static networks are arranged in a grid layout, we visualize the DNet by arranging the static networks in a time-ordered sequence. Figure 4 represents a sample DNet visualization of the EM data for a hypothetical scenario (see section 3.4). Note that the relative location of all AOIs in Figure 4 remains constant in each of the networks corresponding to various time intervals. For example, AOI A is placed at the bottom row first column in all the constituent networks. This constant relative position of

each AOI helps in their navigation across various time intervals and thus helps to preserve the mental map of the observer. The DNet has four time intervals, and the AOI fixation sequences for all these intervals is shown in Table 1. We should note that, due to the dynamic nature of the AOIs and typical scanning strategy of ATCSs, some AOIs may not be fixated upon despite being present on the display. In addition, the overlapped AOI cases arise in the AOI fixation sequence only if they are fixated upon by the ATCSs. Theoretically, there can be many possible overlapped AOI cases as compared to what is observed in real life experimental data. For example, with n unique AOIs, theoretically, we can have $\sum_{r=1}^{n}\binom{n}{r} = \sum_{r=1}^{n}\frac{n!}{r!(n-r)!}$ possible AOIs cases (including both single and overlapped cases) on the display. Although, not all overlapped AOI states will appear on the display and even if they occur not all of them will be fixated upon by the ATCSs. This is also the case with single AOI cases.

Thus, it is computationally expensive and inefficient approach to consider all those AOI states which have not being fixated at all and thus does not appear in the AOI fixation sequence. Consequently, we only consider those AOI states for visualization and analysis which appear at least once in the AOI fixation sequence of the ATCS's. As a result, we ignore those AOIs which, despite being present on the display, received no EFs from the ATCSs. Therefore, not all AOIs appear in the DNet visualization for each of the ATCSs, which results in a change in the relative position of each AOI in the grid structure of the DNet visualization. In addition, for comparing various visual scanning strategies, we only consider the AOI cases which are common to all the AOI fixation sequences of various ATCS's.

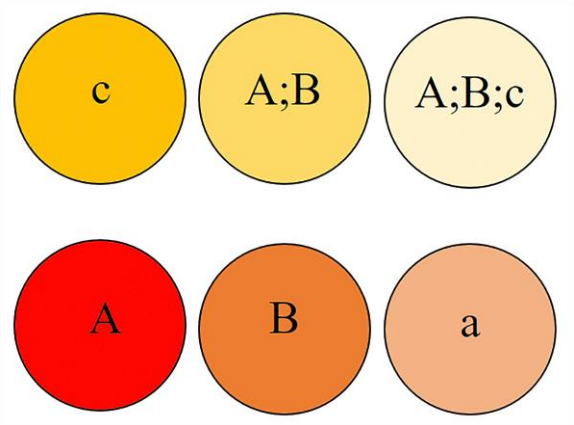


Figure 3: Sample AOI ordering scheme for the grid layout used in designing the networks in the DNet framework.

2.3.7 STEP 7. Calculate target (or AOI) importance measures

Step 7 involves calculating the measures using equations (i), (ii), and (iii). As a result, we provide the DNet as the input to this step, and as output we obtain all three importance measure values for each of the AOIs with respect to all the time intervals considered in the DNet. For example, consider the DNet visualization in Figure 4, where the indegree value of AOI ***B*** changes from 4 to 3 as we move from the time interval 1 to 2 (Figure 4 (a)-(b)). For the given DNet in Figure 4, to calculate the closeness and betweenness value for AOI ***B***, we first demonstrate how to calculate the distance between the AOIs. For example, in the first time interval, the distance from AOI ***B*** to ***A*** is given by $d_{BA}(t) = 1/w_{BA}(t)$ and substituting $t = 1$, we get $d_{BA}(1) = 1/3$.

To calculate the minimum distance between two AOIs consider Figure 4 (a), where there are two EF transition paths from AOI ***B*** to ***E***: Direct path from AOI ***B*** to ***E***, which has edge weight 1; indirect path from AOI ***B*** to ***A*** and then to ***E***, which has edge weights 3 and 3, respectively. Thus, the shortest distance from AOI ***B*** to ***E*** is defined as the path having the minimum distance of $d_{BE}{}^{*}(t) = min\left[d_{BE}(t), \left(d_{BA}(t) + d_{AE}(t)\right)\right]$. For the first time interval $(t = 1)$, we obtain $d_{BE}{}^{*}(1) = min[1/1, (1/3 + 1/3)] = min[1, 2/3] =$

$2/3$. Thus, for the first time interval, the closeness and betweenness value of AOI ***B*** is given by 2.375 and 8, respectively.

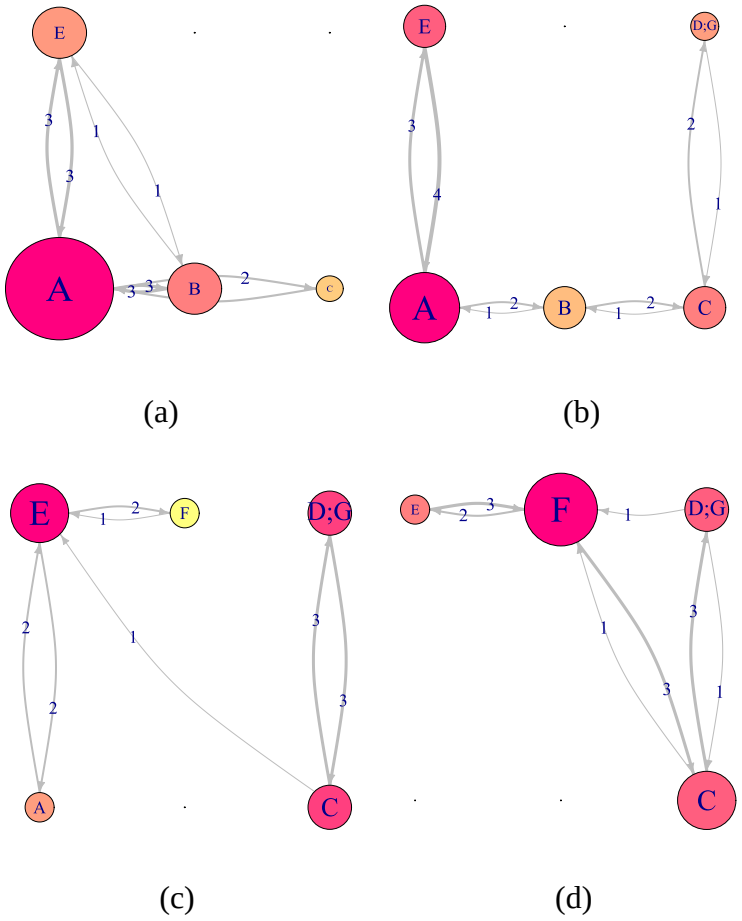


Figure 4: Sample DNet visualization of EM data for the hypothetical scenario described in section 3.4. The DNet consists of four time intervals. The figure shows the important AOIs for each time interval, and how the importance of various AOI's is changing with time. Sequentially: (a) Time interval 1. (b) Time interval 2. (c) Time interval 3. (d) Time interval 4.

2.3.8 STEP 8. Normalize and visualize target (or AOI) importance measures

The last step involves normalizing the calculated measures and subsequently visualizing the normalized measure values. Thus, the obtained AOI importance measures in step 7 acts as input for this step, and as output we obtain the normalized measure values

accompanied by their visualization. Equation (i), (ii), and (iii) shows that the three measures both depend on the number of AOIs and the amount of EFs. In addition, these measures also have different units, as a result, they are incommensurable. For the present dynamic scenario, both the number of AOIs and EFs are not constant for the different time intervals considered in the DNet analysis. In addition, to compare the AOI importance values across various time intervals and across multiple observers, we need to eliminate the units of importance measures, thereby bringing them to a similar scale. To address this issue, we present two normalization options.

2.3.8.1 Percent normalization

Percent normalization refers to dividing an importance measurement of an AOI by the sum of the same measurement of all the AOIs. The percent normalized indegree value of an AOI shows the percentage share of the total number of EFs received by an AOI. Thus, it can be interpreted as the percentage of the total attention attributed to an AOI. The percent normalized indegree value of the j^{th}AOI for time interval t is calculated as:

$$\overline{I_J}(t) = \frac{I_j(t)}{\sum_{j=1}^{m_t} I_j(t)} \quad (4)$$

Where, $\overline{I_J}(t)$ is the percent normalized indegree value for time interval t. We get, $0 \leq \overline{I_J}(t) \leq 1$ and $\sum_{j=1}^{m_t} \overline{I_J}(t) = 1$.

2.3.8.2 Distance normalization

Distance normalization refers to calculating how far away a given importance measurement of an AOI is from the maximum value observed for that measurement within a time interval. The distance normalization process is defined for all the three target importance measures. The distance normalized measure value of an AOI can be interpreted as the relative amount of attention given to an AOI as compared to the maximum amount

of attention given to any AOI. The distance normalized measure for the j^{th}AOI, for time interval t, is calculated as:

$$\widetilde{\emptyset_j}(t) = \frac{\emptyset_j(t) - \min_j \emptyset_j(t)}{\max_j \emptyset_j(t) - \min_j \emptyset_j(t)} \quad (5)$$

Where, $\max_j \emptyset_j(t)$, $\min_j \emptyset_j(t)$ and $\widetilde{\emptyset_j}(t)$ is the maximum, minimum, and distance normalized value of the measure $\emptyset_j(t)$ respectively ($0 \leq \widetilde{\emptyset_j}(t) \leq 1$). $\emptyset_j(t)$ is applicable for representing any of the three target importance measures (i.e. indegree, closeness, and betweenness) whereas percent normalization is useful for only indegree due to manner in which the measures are calculated (see sections 2.2.4 and 2.3.7).

2.3.8.3 Target (or AOI) importance measure visualization

Once the values of the target importance measures are normalized, the last step involves their subsequent visualization. We provide a couple of examples of visualization approaches: Dot plot based on multiple time intervals (for a single observer and single vertex importance measure) and bar plot based on multiple participants (for a single time interval and single vertex importance measure). There can be various ways to visually represent the combinations of importance measures, normalization methods, multiple time intervals, and number of participants. Another example of a bar plot that compares all importance measures of a single participant for a single time frame is also provided in the Results section.

2.3.8.4 Adapted dot plot visualization

Dot plots can better represent the evolution of the visual attention across various time intervals. We have adapted the dot plot to visualize the percent normalized indegree measure for all the time intervals considered in the DNet framework. Using the normalized

measurements helps in comparing the relative significance of AOIs within a single time interval and how an AOI's importance is changing across various time intervals. Figure 5 shows a sample dot plot for the normalized indegree measure for all the AOIs presented in the DNet in Figure 4. In Figure 5, the size of each dot is proportional to the percent normalized indegree value of the AOI for the given time interval.

3.3.8.5 Adapted bar plot visualization

Bar plots can better assist in comparing a single importance measure among multiple participants or multiple importance measures for a single participant. Figure 6 shows an example of the adapted bar plot that compares the indegree measurements among multiple participants through the distance normalization approach.

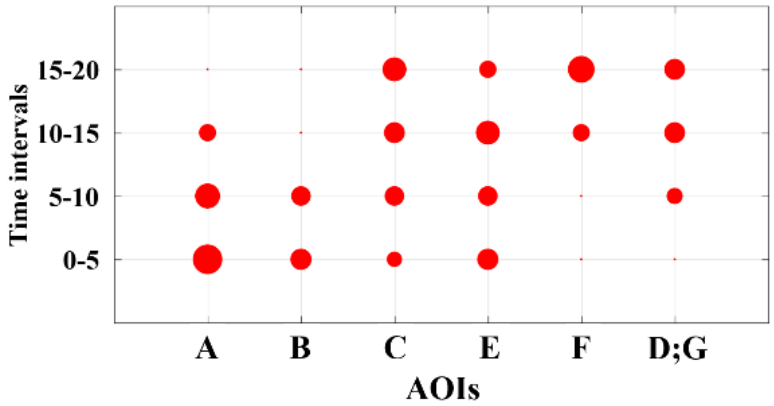


Figure 5: Sample dot plot showing the evolutions of the indegree measurements (for both single and overlapping AOIs) based on time frames using the results shown in Figure 4. Note that the time intervals need not be fixed but can be determined differently (e.g. dividing the time intervals based on events).

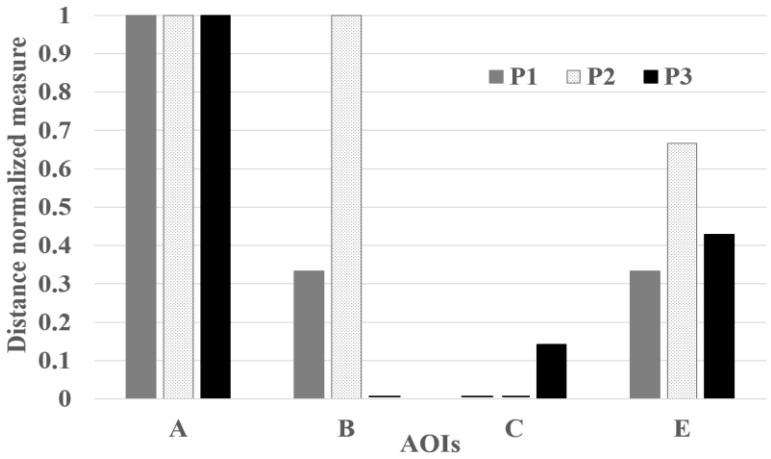


Figure 6: Sample bar plot showing the relative indegree values of various AOIs among multiple observers (i.e. P1, P2, and P3) using the results shown in Figure 4. For this example, the vertical axis represents the distance normalized indegree measure value. Only considering the indegree measure (in this example), AOI A is considered as the important AOI by all participants, whereas AOI B is considered important by only P2.

2.4 Experiment

The proposed approach was implemented into an air traffic control task. The task involved expert ATCSs observing the radar display to detect possible aircraft collisions. Details are as follows.

2.4.1 Participants

Three retired ATCSs (P1, P2 and P3) with over thirty years of experience were participants for the experiment. The ATCSs were recruited with the help of FAA Civil Aerospace Medical Institute (CAMI). Also, three FAA CAMI employees were also involved in the experiment as pseudo-pilots. They made the necessary maneuverings by following the ATCS's voice commands. A simulated radio connection was used as the communication channel between the pseudo-pilots and ATCSs.

2.4.2 Apparatus

Hardware: A 19.83 × 19.83-inch monitor, with a 2048 × 2048-pixel active display area, was used for displaying the simulated air traffic scenarios. Apart from this, an additional monitor

(kept to the right of the simulation monitor) was used to display textual information about the aircraft, e.g. trajectory and possible future conflicts it might encounter (also known as the Enroute Automation Modernization tool). The ATCS's used a keyboard, placed beneath the simulation display, to provide necessary input commands. FaceLab 5 eye tracker with an accuracy in the range of $0.5 - 1.0$ degree visual angle error and sampling rate of 60 Hz (Eyetracking.com, 2017) was used to collect ATCSs' EM data. A threshold of 100 milliseconds was used for defining fixations. The participants were seated within a range of 50-70 cm from the simulated radar display.

Software: A customized ISim software used by FAA CAMI was used for simulating the enroute air traffic scenario (with an update rate of 1 sec). EyeWorks software was used to process the raw eye tracking data collected through FaceLabs eye trackers.

2.4.3 Scenario and task

2.4.3.1 Scenario

For the experiment we used one 20 minutes long simulated enroute air traffic scenario, which was developed and provided by the FAA. This scenario had a total of thirty nine unique aircraft (named as A, B, …, Z, a, b, …, m) with an average of twenty aircraft per frame. The scenario had a minimum of seven and maximum of thirty aircraft per frame. Figure 7 represents a screen capture of the simulated scenario. The scenario update rate was 1 second. The display shows various aircraft and a weather patch in blue color. The aircraft representation shows the direction in which the aircraft is moving (shown by the white line) and data block which contains information about the aircraft's computer code name, altitude, current speed, and destination airport. For example, in Figure 7, the aircraft N7890 (AOI ***U***) is flying at a constant altitude of 19,000 ft (as shown by 190C) with a speed of 402 knots and it going to KGPT airport (i.e. Gulfport–Biloxi International Airport in Mississippi).

2.4.3.2 Task

The task involved controlling the simulated low altitude enroute airspace using the ERAM system. The ATCSs are required to fulfil two objectives: (1) Route the aircraft through the sector within the display and (2) avoid any conflict scenario by preventing loss of separation (vertical and horizontal separation of 1000 ft and 5 knots respectively) between aircraft. To achieve these objectives, the ATCSs gave voice commands (e.g. change is altitudes, speed and direction of the aircraft) to the pseudo pilots, using the simulated radio, for necessary maneuvering of the aircraft.

2.4.4 Data analysis

For the DNet analysis, the 20 minutes simulated scenario was divided manually, on trial and error basis, into four equal time intervals of 5 minutes. This choice was motivated by the fact that we observed on average an aircraft spends around 5 minutes on the radar display. Event based intervals were not applied in this research since many aircraft appeared and disappeared throughout.

Matlab was used to develop the AOI fixation sequence for each of the four time intervals mentioned above. In addition, igraph package (Csardi & Nepusz, 2006) in R software was used to create the dynamic network representation of the AOI fixation sequences and also for AOI importance metric calculation and visualization.

Figure 2 shows there are eight major steps in the proposed methodology. The time complexity (TC) of steps 1-4 for one time interval can be written as $TC_{1-4} = \alpha_1 e_t n_t$, where, e_t and n_t are the number of eye fixations and number of AOIs in time interval t respectively, and α_1 is some positive constant. Due to human physiological limitations the value of e_t cannot increase arbitrability. Therefore, the governing factor of the time complexity for steps 1-4 is the number of AOIs. Thus, we can approximately write $TC_{1-4} =$

$O(n_t)$. Taking the worst case scenario, we can find an upper bound for this value by taking the maximum number of AOIs across all time intervals. Let, n_{max} be the maximum number of AOIs present during any of the time intervals. Thus, we get $TC_{1-4} = \alpha_1 n_{max}$. Similarly, the time complexity of step 5 can be written as $TC_5 = \alpha_2 {n_{max}}^2$, where α_2 is some positive constant. For T time intervals the time complexity of step 1-4 and 5 can be written as $TC_{1-4} = T\alpha_1 n_{max}$ and $TC_5 = T\alpha_2 {n_{max}}^2$ respectively.

As mentioned in section 3.6.2, in a given DNet visualization, the relative location of each AOI remains constant within the various constituent networks for various time intervals. As a result, for drawing the DNet, we need to consider all the unique AOI states that arise in the AOI fixation sequence of an ATCS. Therefore, the time complexity for step 6 can be written as, $TC_6 = T\alpha_3 n_{uniq}$, where, n_{uniq} is the number of unique AOI states (both single and overlapped) in the AOI fixation sequence and α_3 is some positive constant. Step 7 consists of calculating three AOI importance measure. The time complexity for indegree, closeness and betweenness metric is $T\alpha_4 n_{max}$, $T\alpha_5({n_{max}}^2 - n_{max})$, $T\alpha_6 n_{max}(n_{max} - 1)(n_{max} - 2)$ respectively.

Thus, the time complexity for step 7 can be written as $TC_7 = T\alpha_4 n_{max} + T\alpha_5({n_{max}}^2 - n_{max}) + T\alpha_6 n_{max}(n_{max} - 1)(n_{max} - 2)$.

The time complexity of Step 8 can be written as $TC_8 = T\alpha_7 n_{max}$, where, α_7 is some positive constant. Adding all the time complexity for step 1-8, the total time complexity for the overall process can be written as: $TC_{tot} = T\big[\alpha_1 n_{max} + \alpha_2 {n_{max}}^2 + \alpha_3 n_{uniq} + \alpha_4 n_{max} + \alpha_5({n_{max}}^2 - n_{max}) + \alpha_6 n_{max}(n_{max} - 1)(n_{max} - 2) + \alpha_7 n_{max}\big]$.

On simple algebraic reorganization, we get that:

$$TC_{tot} = T[\alpha_6 {n_{max}}^3 + {n_{max}}^2(\alpha_2 + \alpha_5 - 3\alpha_6) + n_{max}(\alpha_1 + \alpha_4 + 2\alpha_6 + \alpha_7 - \alpha_5) + \alpha_3 n_{uniq}].$$

Thus, neglecting the lower order terms of n_{max}, the approx. time complexity of the overall data analysis process reduces to the order of $O(T{n_{max}}^3 + Tn_{uniq})$. Thus, we can see that the number of time interval, maximum number of AOIs within any time interval and the number of unique AOI states in the AOI fixation sequence do impact the processing time of the proposed approach.

2.5 Results

2.5.1 Dynamic graph visualization

Figure 8 represents the DNet visualization of the EM data of one ATCS for the simulated scenario shown in Figure 7. In Figure 8 (a) (i.e. time interval 0-5 minutes), AOI F (i.e. AAL68) and U (i.e. N7890) are the most important AOIs as they both have most EF numbers (circle size) and longest EF duration (circle color). In addition, there are highest EF transitions between AOI F and AOI U (based on the thickness of the link). AOI K (i.e. EJA33), despite having a small number of EFs, has substantially longer EF duration. AOI b and AOI d can also be considered as important AOIs based on how a researcher wants to set the threshold.

For the second time interval (i.e. 5-10 minutes) shown in Figure 8 (b), the important AOIs have changed to AOI G (newly appeared aircraft not shown in Figure 8(a) and AOI K. Notice that AOI F has moved out of the display (see Figure 7(b)) and AOI U is still within the display but AOI U is not visually attended any longer.

Besides, notice that AOI d (i.e. UAL1322) has been receiving consistent visual attention throughout the two time intervals 0-5 minutes and 5-10 minutes. Similarly, the

important AOIs change for the next two time intervals. In the last time interval, the overlapping AOI (i.e. AOI (J;f)) receives much visual attention. As we move across time intervals from 1 to 4, a visible trend is the increase in the complexity of the network, with increase in the number of AOIs and EF transitions among them.

2.5.2 Adapted dot plot

Figure 9 represents the dotplot visualization of the normalized indegree measure for all the AOIs present in DNet shown in Figure 8. For example, in Figure 9 (a), AOI F and U has high importance in first time interval (i.e. 0-5), although, their importance reduces drastically in the subsequent intervals. AOI d receives consistent visual attention throughout the first two intervals. The indegree results are in accordance with the DNet results in Figure 8 since indegree measures the number of EFs received by an AOI.

The adapted dot plot better shows the evolution of important AOIs. Considering AOI K, we see that its importance initially grows as we move from the first time interval to the second, where it reaches its maximum and then starts decreasing for the last two time intervals. Another noticeable fact is that majority of the overlapped AOI cases have significant importance only in one time interval. These trends took more time to identify when observing the DNet.

2.5.3 Adapted bar plot

Figure 10 represents the relative importance of various AOIs present in the first time interval of participant 1 (i.e. P1). Again, AOI ***F*** and ***U*** are important, but also, we can identify that AOI ***b*** (i.e. SWA340) and AOI ***d*** can also be important AOIs when we additionally consider the closeness and betweenness values. We can also observe that AOI ***K*** had small EF numbers during the first interval, however, it can be considered as an

important AOI due to the long EF duration and relatively high closeness value. Furthermore, we can see that the AOI ***Q*** and AOI ***a*** might be an important AOI considering that AOI ***(Q;a)*** also has moderate indegree and closeness values.

Figure 11 shows the measures (i.e. indegree, closeness, and betweenness) visualized based on participants for the first time interval. Although there are slight variations, we can see a consistent trend among the participants. In addition, it becomes more evident that AOI ***Q*** and AOI ***a*** can be important AOIs (considering the values of AOI ***(Q;a)***) in addition to AOI ***F***, AOI ***K***, AOI ***b*** and AOI ***d***. In detail, overlapping AOI ***(Q;a)*** received substantial EF duration (as shown by the vertex color in the DNet visualization) although it received moderate number of EFs on it (as shown by the vertex size in the DNet visualization). On the other hand, comparing the three importance measurements, AOI ***(Q;a)***, in spite of having moderate indegree and closeness value, has insignificant betweenness value.

This might be due to the short lifespan of the overlapped AOI state since AOI ***a*** (i.e. SWA2920 flying at 21000 ft with speed 424 knots overtakes AOI ***Q*** (i.e. N46332 flying at 7000 ft with speed 163 knots very quickly on the display. As a result, it is highly unlikely that it will play a crucial role as a bridge for the flow of attention between other groups of aircraft.

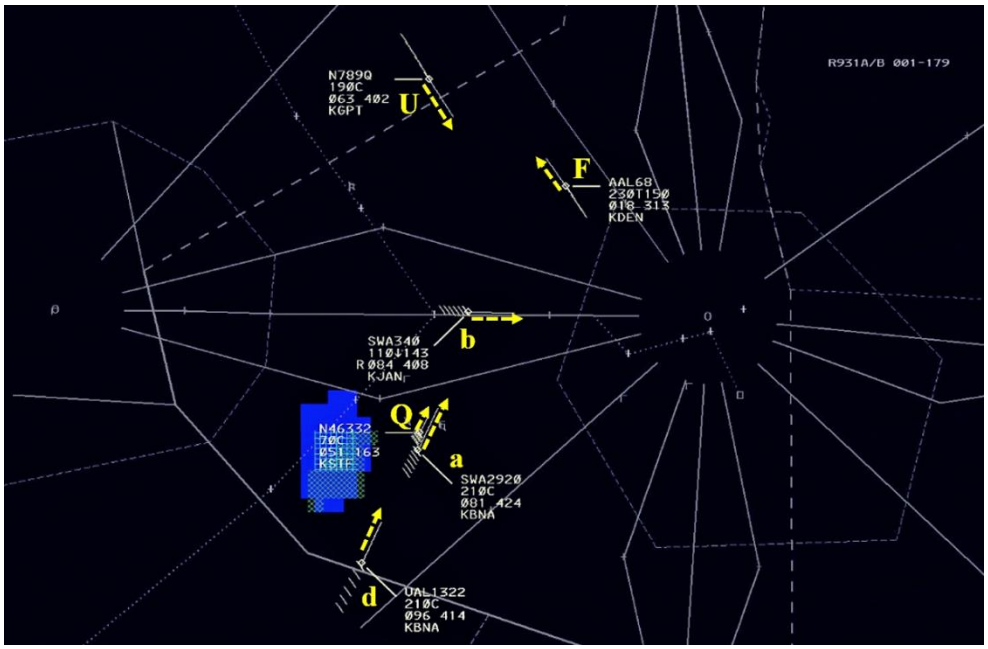
2.6 Discussion

Results obtained from the DNet visualization coupled with multiple time interval visualizations enabled us to identify which are the important targets in each of the time intervals and how their importance evolves as we move across time intervals. Application of the three adapted target importance measures (i.e. indegree, closeness, and betweenness)

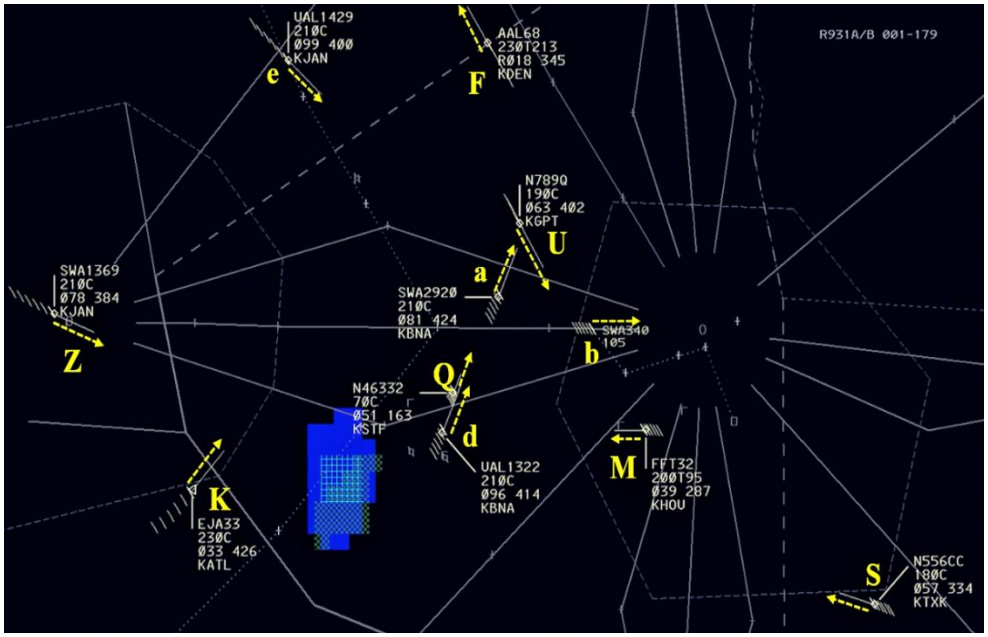
along with the various ways to visualize those values showed that we can better analyze target importance measures that accounts for the interaction among targets.

In addition, we could identify various EM network characteristics of each participant and how the relative importance of various targets differs among them.

DNet was better to visualize the EM flow of the overall network and EF transitions, whereas the adapted dot plot was better to visualize the evolution of importance of each AOI across various time intervals. It is noted that the results could have been different if we had used a different time interval. For example, if the time interval was to 10 minutes, the most important AOI would turn out to be AOI ***d*** and the importance of AOI ***F*** and AOI ***U*** would have been less substantial. However, such results can be misleading since AOI ***F*** and AOI ***U*** were important AOIs during the first 5 minutes.

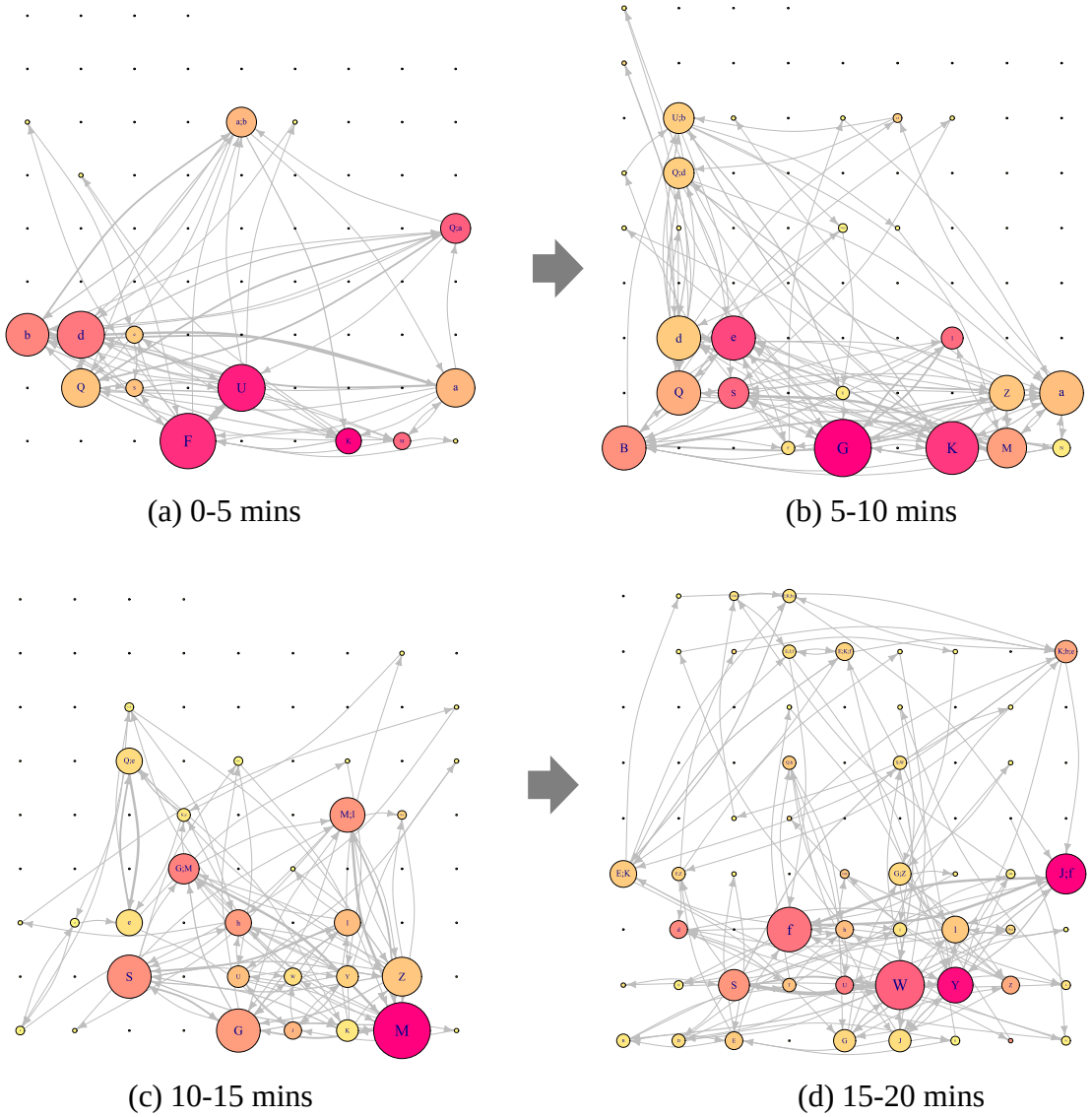


(a) Initial display (i.e. at time 0)



(b) At the end of first time interval (i.e. at 5 minutes): AOI ***F*** is moving out of the display.

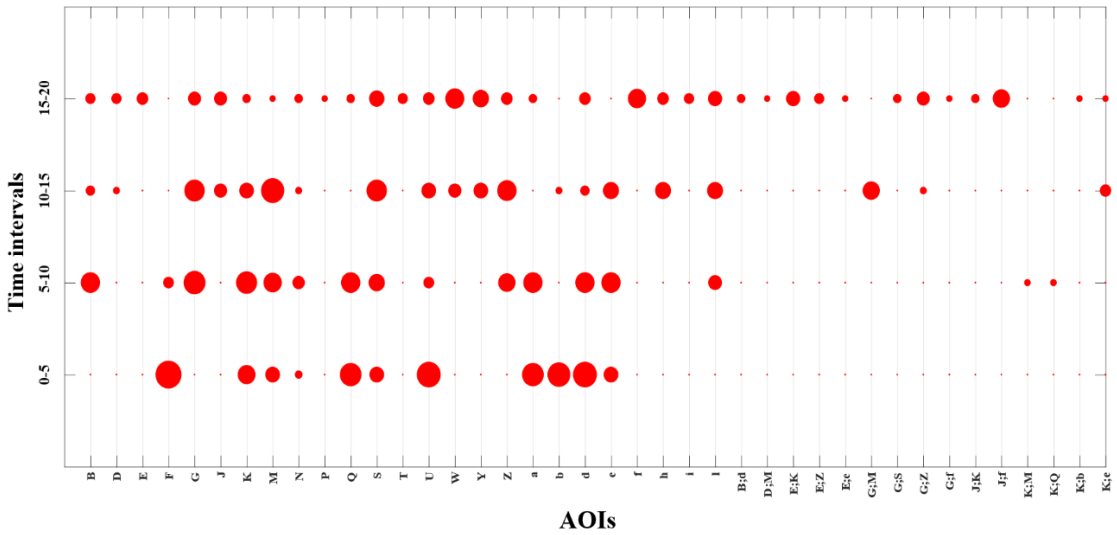
Figure 7: Screen captures at the 0 and 5 minutes. The blue patch shows a weather feature (e.g. thunderstorm) which the aircraft needs to avoid. The characters and arrows in yellow color represent an AOI and the direction of the aircraft, respectively. These characters and arrows were not present during the experiment. Each AOI consists of an aircraft shown as a diamond shape, its direction shown as a vector line, and its associated data block (first line shows the aircraft ID, second line shows its altitude, third line shows its computer ID and speed, and forth line shows its destination). If the altitude changes, the aimed altitude is shown followed by letter "T" and the current altitude. For example, AOI *F* is AAL68 (i.e. American Airlines 68) and flying toward northwest. Its current altitude is 21,300 ft (at 5 minutes) and target altitude is 23,000 ft. Its speed is 345 knots (at 5 minutes) and destination is KDEN (i.e. Denver international airport).



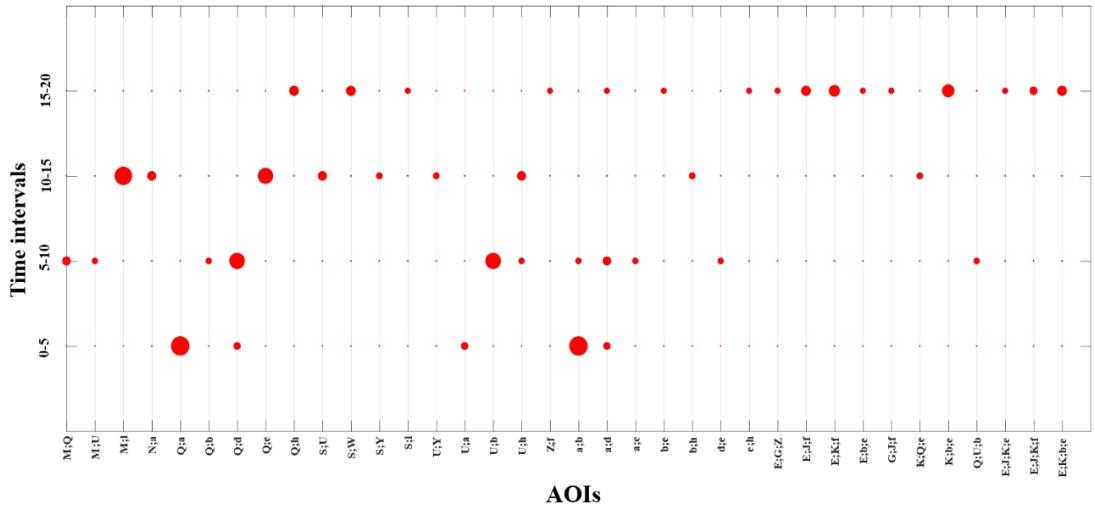
(a) 0-5 mins (b) 5-10 mins

(c) 10-15 mins (d) 15-20 mins

Figure 8: DNet visualisation of the EM data of one ATCS for the simulated enroute air traffic scenario. The figure shows the important AOIs in terms of EF numbers and EF duration for all the four time intervals. It also shows how the importance of various AOIs changes as we move across various time intervals. The relative location of each of the AOIs remains constant across all the static networks associated with different time intervals. For example, AOI *K* is placed at the bottom row sixth column for all the constituent static networks.



(a) Part 1



(b) Part 2

Figure 9: Adapted dot plot visualization of the percent normalized indegree measure value for all AOIs (single and overlapped) present in the DNet visualization in Figure 8.

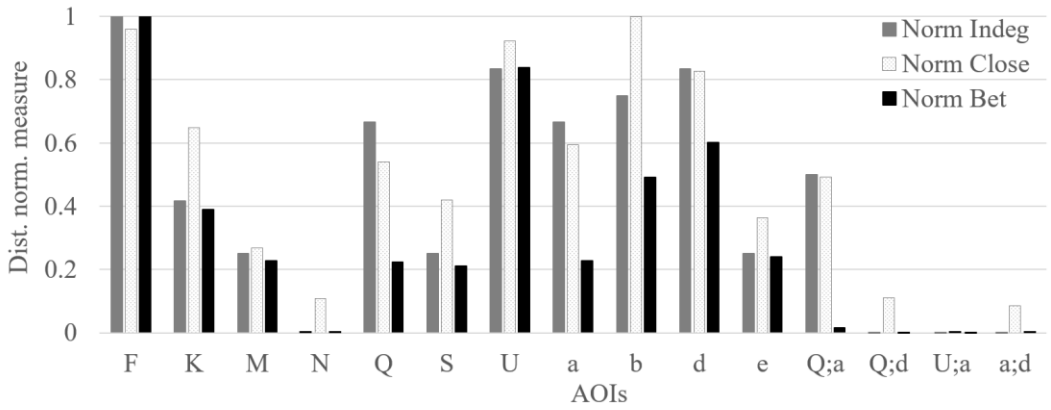


Figure 10: Distance normalized measure value for all AOIs present in the DNet for time interval 0-5 mins for P1 (see Figure 8(a)). "Norm Indeg," "Norm Close," and "Norm Bet" refers to normalized indegree, closeness, and betweenness values, respectively. The vertical axis represents the distanced normalized measure value. The vertical axis ranges from 0 to 1, thus it helps to analyze the relative importance of all three measures at a given time interval.

Selection of time interval thresholds (whether fixed or varied based on events/triggers) can be tricky and depend on the scenario characteristics. The adapted bar plots were better when multiple target importance measures and/or multiple participants' measures were needed to be compared side-by-side. The normalized and adapted measure plots show that, despite receiving a substantial amount of EFs, an AOI may not be significant in terms of the flow of visual attention across the various AOIs within the display as shown by the closeness and betweenness measures. The proposed approach can be useful in increasing the training efficiency of the novice controllers. Novice controllers can know which are the important targets that need to be focused upon and how to move the attention across various targets as the scenario characteristics change. Furthermore, the trainees can better understand which targets are highly correlated for conflict mitigation through observing the EF transition characteristics using the DNet and evaluating the closeness/ between-ness values.

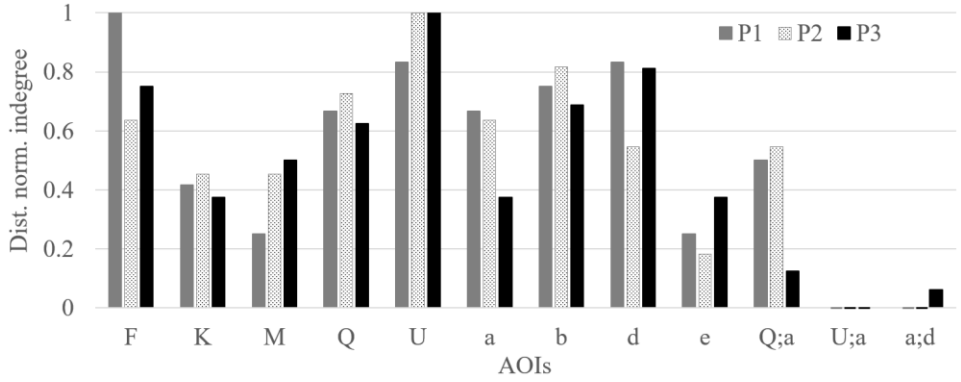
2.7 Conclusion

In this research, we integrated the DNet framework with three target important measures (i.e. indegree, closeness, and betweenness). During the integration, we normalized the measurements and then adapted the dot plot and bar plot to better visualize the outputs. The approach facilitated the understanding of how visual attention occur on the dynamic network (i.e. EM network created from an aircraft conflict task) from various perspectives. The results obtained showed that the traditional approach of using the raw EM data measures (i.e. number and duration of EFs) might be misleading for dynamic scenarios where the targets' lifespan on the display are non-uniform. The proposed approach enabled us to better understand how the observers' attention was devoted to the various targets including the overlapping targets on the display. Also, in case of dynamic targets, to understand target importance we need to also consider which targets are integral in the smooth flow of attention across the various targets within the display.

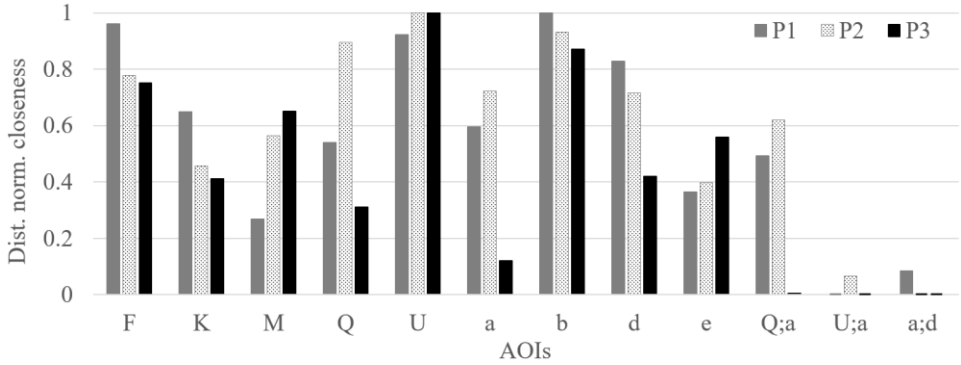
2.8 Limitations and Future work

One challenging issue was determining the time intervals for effective analysis. We have used five minute time intervals for the experiment based on trial and errors and we could have applied time intervals based on specific events (e.g. when a target appears or disappears, or when a verbal control command is issued by an expert ATCS). However, event-based time intervals can create issues if too many events occur within very short times or too few events occur in very long times. Instead, we could try applying shorter time intervals. For example, reducing the 5 minute interval to 1 minute interval). However, we might be burdened with visualizing and analyzing too many out-puts. Therefore, formal

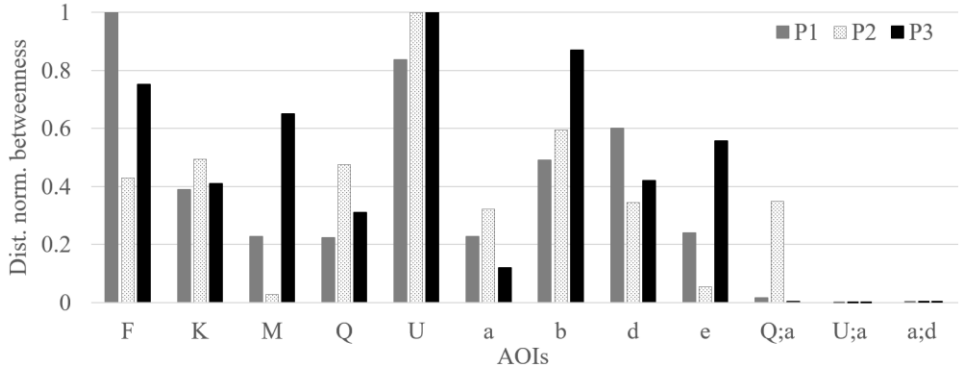
sensitivity analysis needs to be performed as part of a future study to determine an optimal range of values for the intervals.



(a) Distance normalized indegree measure



(b) Distance normalized closeness measure



(c) Distance normalized betweenness measure

Figure 11: Comparison of three ATCS's visual scanning strategy by analyzing the distance normalized importance measure values of AOIs. The Figures (a), (b), and (c) shows the relative importance of AOIs in terms of distance normalized indegree, closeness, and betweenness value (as shown by their zero values in the vertical axis in Figure 10).

In addition, we have not used pre-determined thresholds to identify important AOIs and only identified them if all measures were relatively higher than others. Therefore, a statistical method should be developed that clearly differentiate the important AOIs from the non-important one.

Moreover, as a future study, we can also analyze the variation of the saccade length and eye fixation duration with time and various air traffic scenario characteristics. For example, we can consider how the distance between several aircraft on the display affects the saccade length of the ATCS. This might help us analyze the type of search behaviors undertaken by them, i.e. is the search goal-oriented where they search for targets having similar characteristics or is it a random one.

With the increase in the number of AOIs and EF transition between them, the visual scalability of the DNet visualization gets impacted negatively, as there are more instances of edge crossings in the network representations. Furthermore, increase in the number of time intervals also increases the number of static networks within a DNet framework. This leads to an increase in the cognitive load of the observers as they have to keep track of an AOI across more number of network representations. Thus, in terms of visualization scalability, we can apply various graph simplification processes, also known as graph filtering processes, where the unimportant edges (i.e. edges representing low EF transitions) are not considered for visualization purpose, thus reducing the visual clutter.

Furthermore, it can be valuable to analyze the community structure of AOIs (cluster of AOIs having a high amount of EF transitions between them) and their evolution formed in the network representation of the EF data. Finally, we are planning to develop a mapping scheme between the visual scanning pattern classifications (Mcclung & Kang, 2016) with

the DNet results. This might help in understanding how different visual scanning strategies are related to the overall target's importance.

Ethics and Conflict of Interest

The author(s) declare(s) that the contents of the article are in agreement with the ethics described in http://biblio.unibe.ch/portale/elibrary/BOP/jemr/ethics.html and that there is no conflict of interest regarding the publication of this paper.

Acknowledgements

This research is supported by the Federal Aviation Administration Center of Excellence (Project No. A17-0162) and the FAA Aviation Research Grants Program (Project No. 15-P-0009). The FAA has sponsored this project through the Center of Excellence for Technical Training and Human Performance. However, the agency neither endorses nor rejects the findings of this research. This information is provided in the interest of invoking technical community comment on the results and conclusions of the research. We worked closely with the FAA Civil Aerospace Medical Institute at Mike Monroney Aeronautical Center (AAM-520).

Chapter 3: Novel DWN-based visualization techniques

3.1 Introduction

This chapter provides the details of two novel visualization methods, norm-cube and order plot, that further builds on visualization of the DWN-based target importance measures method introduced in Chapter 2. These two novel methods address the major limitations of the dot plot and bar plot methods, introduced earlier in Chapter 2. For a detailed analysis of the various limitations of the prevalent eye movement visualization method please refer to Chapter 2.

In detail, dot plot method (refer to Chapter 2 for details) visualizes the evolution of target importance for several time intervals but, for only one measure. On the other hand, norm-cube plot (details explained in the following section), provides a way to visualize the overall target importance, considering all three target importance measures.

Bar plot (refer to Chapter 2 for details), although allows representing of target importance for multiple observers, doesn't provide the full picture. Note that, for the enroute traffic scenario there arises many unique overlapped AOI cases in the AOI fixation sequences which are unique for individual ATCs. As a result, it becomes challenging to compare these cases given the bar plot framework. This limitation is addressed by the order plot method (details explained in the following section), which can both handle multiple observer data and can easily represent the relative importance of overlapped AOI cases, and also provides a visual tool to compare across the relative ordering of various targets (AOIs) in the visual scanning strategy of multiple observers.

3.2 Norm-cube plot

Norm-cube plot refers to visualizing the overall importance of targets (AOIs) in terms of all three network-based importance measures, i.e. Indegree, Closeness, and Betweenness (see Chapter 2). Note that, all three importance measures have different dimension and range; therefore, the norm-cube plot uses their normalized values to combine them into a single plot. Refer chapter 2 for details of the measure normalization process.

Figure 12 shows an example of the norm-cube plot for one observer. The three axes represent the normalized values of the three different measures (i.e. Indegree, Closeness, and Betweenness) derived from the directed weighted network model explained in chapter 2. Note that each norm-cube plot caters to only one time segment of analysis, e.g. 0-5 min, 5-10 min etc.

The circles represent various AOIs. The Euclidean distance of any AOI representation is proportional to its overall importance considering all three measures, i.e. Indegree, Closeness, and Betweenness. Besides, to enhance the visualization, the size of each circle is made proportional to its distance from the origin. Furthermore, the color on each AOI follows a multi-hued pattern, where it is proportional to cumulative eye fixation duration on each it. In this, red color represents high eye fixation duration and yellow suggest low eye fixation duration. Furthermore, to reduce visual clutter, AOI names of those AOIs that have very close to origin, i.e. very low overall importance, are not shown (see two yellow circles in Figure 12). Considering all these above-mentioned aspects, it

can be inferred from Figure 12 that AOI ***A*** is the most important target overall, and it has very high eye fixation duration on it.

The major advantage of using the norm-cube plot is its ability to visualize the overall importance of a target (AOI) in terms of the visual scanning strategy of an ATCs. Although the norm-cube plot can be extended to multiple ATC data, however, for many AOI it might lead to visual clutter. Also, different color scheme might be required to denote different ATCs. It can be considered as future research work.

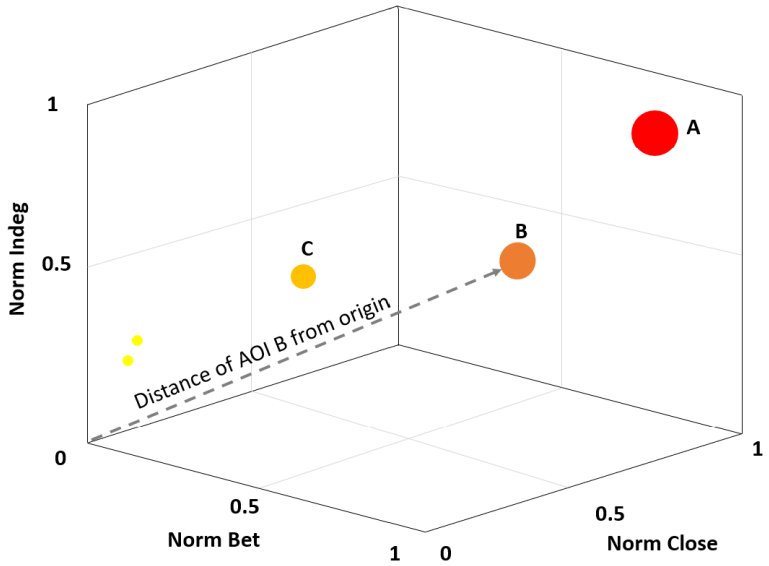


Figure 12: Example of a Norm-cube plot. Norm: normalized, Indeg: indegree, Close: closeness, Bet: betweenness.

3.3 Order plot

Norm-cube plot despite its ability to represent targets' overall importance falls short is visualizing multi-participant data when there are many AOIs on the display. To address the issue of displaying the importance of a large number of targets, and simultaneously provide a visual comparison tool for assessing similarity of visual scanning strategy among ATCs, the order plot has been developed. Order plot refers to visualizing the relative significance of AOIs, in decreasing order of importance, for any given target importance measure out

of the three measures which are indegree, closeness, and betweenness. Figure 13 shows an example order plot for four observers/participants case, where three AOIs are ranked as per decreasing importance for a given target importance measure (indegree for the example) for a single time interval.

Order plot provides a visual tool for comparing the overall scanning strategy of various participants, which in turn can help an analyst to find similarities and/or hidden patterns between various scanning strategies. Note that, for the case of enroute traffic (i.e. where aircraft representations overlap on each other), the singular AOI states are generally common to all the ATCs, however, for the overlapped AOI states it differs across ACTs. The presence or absence of overlapped AOI states in AOI fixation sequence dependent on whether the ATC has fixated on the specific overlapped region or not. Consequently, although order plot allows visual comparison of various singular AOI states importance across ATCs but, it's challenging (even not possible for some cases) to evaluate the importance of every overlapped AOI states across different ATCs.

To further aid the visual comparison of target importance across observers, the size of the targets' representation is made proportional to their relative rank, and they also have been normalized across all the observers. In detail, at the beginning, the maximum and minimum values of targets, in terms of the measure criteria used across all the observers, are selected to determine the maximum and minimum size of the target representations. Subsequently, all the absolute target importance values are then normalized using these threshold values. For example, in Figure 13, the AOI ***A*** for participant/observer P1 has the maximum size and AOI ***C*** for participant/observer P3 has the smallest size. Rest of the

AOI representations are normalized using these to size values. This is similar to the distance normalization approach discussed earlier in Chapter 2 section 2.3.8.

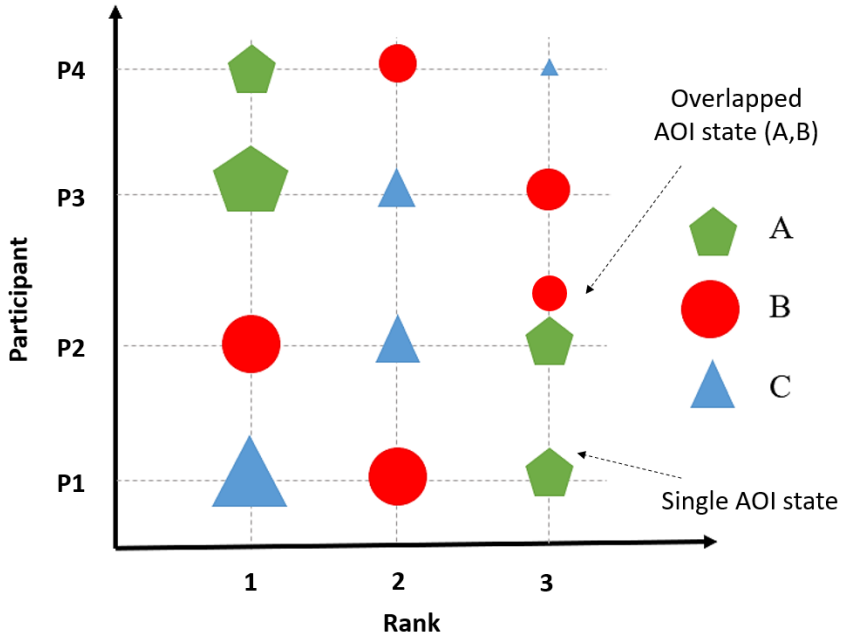


Figure 13: Order plot (indegree measure-based) showing the relative ranking of three AOIs A, B and C for various observers for a given time interval.

3.4 Experimental

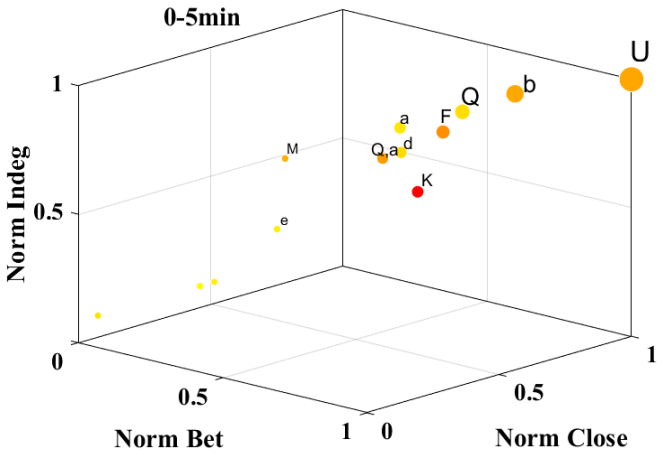
NOTE: The experimental methodology for this chapter remains the same as mentioned in previous Chapter 2. To avoid repetition, the details are not mentioned here.

3.4 Results

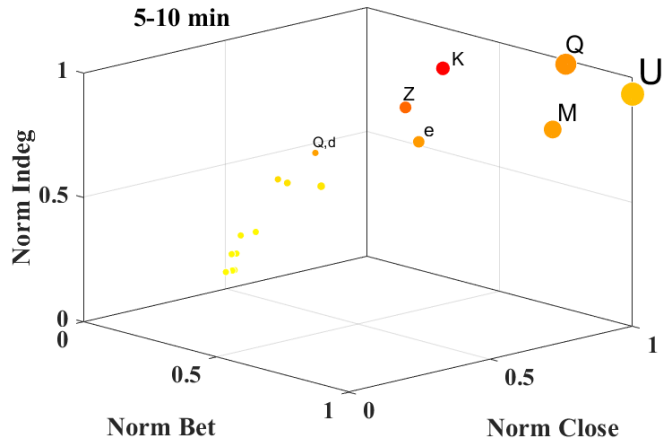
3.4.1 Norm-cube plots

Figure 14 shows the result of norm-cube plot for an ATC for all four time intervals considered for analysis, i.e. 0-5 min, 5-10 min, 10-15 min, and 15-20 min intervals. To see the corresponding network representation of the ATC's eye movement data please refer to Figure 8 in chapter 2.

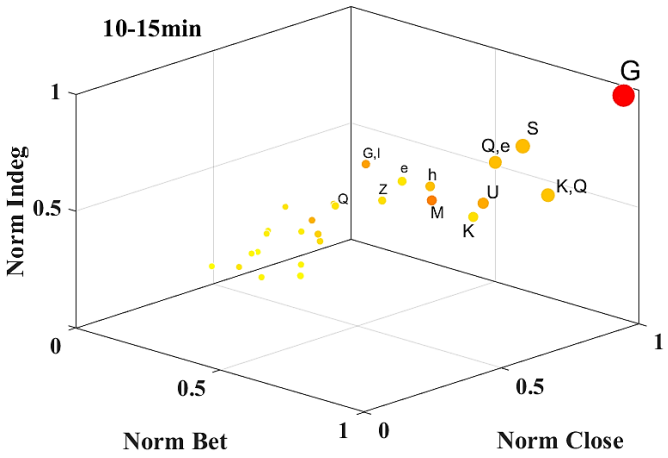
Figure 14 (a) shows that for the first time interval, i.e. 0-5 min interval, AOI ***U*** is the most important target overall. The other very important AOIs are AOI ***b***, ***Q***, and ***F***. For the next time interval, i.e. 5-10 min interval (see Figure 14 (b)), AOI ***U*** remains the most important AOI overall. Also, the overall importance of AOI ***Q*** and ***M*** increases. However, as we move further down the timeline, new AOIs (i.e. AOI ***G***) become more prominent in terms of the visual scanning strategy of the ATC. Several other overlapping AOI states are also observed to be important, e.g. AOI (***Q,e***) and (***K,Q***). As we move further, i.e. 15-20 min interval (see Figure 14 (d)), many AOIs become significant, which shows that due to heavy traffic volume on the radar display the ATC devotes their visual attention to several AOIs.



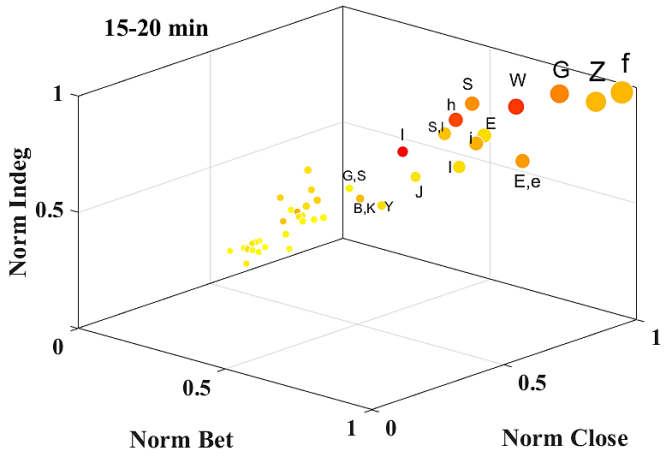
(a) 0-5 min interval



(b) 5-10 min interval



(c) 10-15 min interval



(d) TS 4: 15-20 min

Figure 14: Norm-cube plot for a single ATC showing the overall important AOIs for four different time intervals. Indeg: indegree, Close: closeness, Bet: betweenness.

3.4.2 Order plots

Figure 15 represents the shape and color code used to represent the various AOIs in the order plot visualization method. As the number of AOI increases, this codification scheme helps to reduce visual clutter in the order plot representations. In addition, it also enables a compact representation of the overlapped AOI cases.

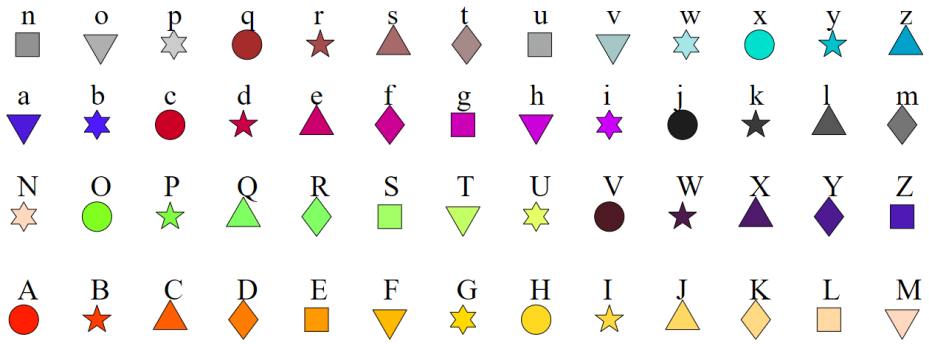
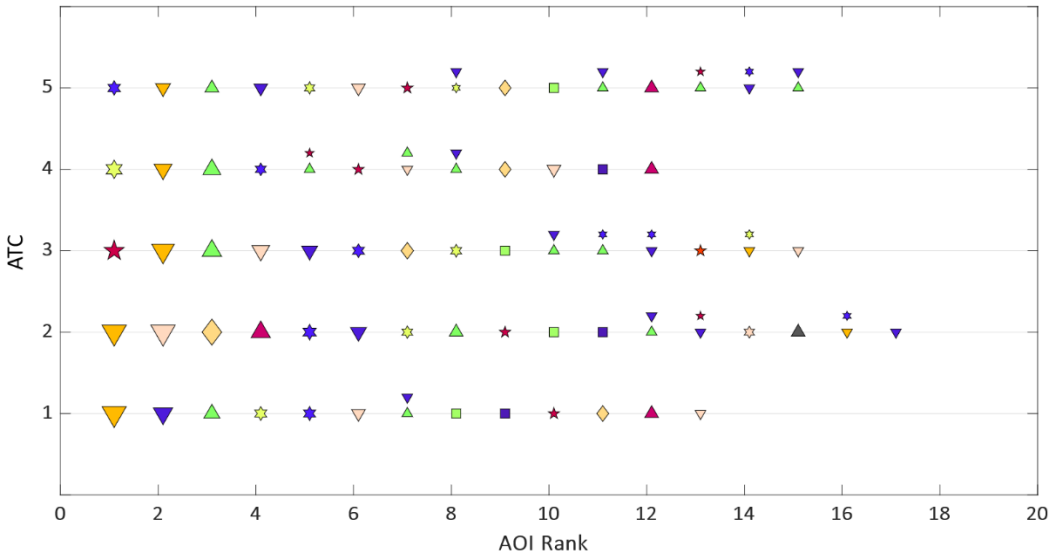


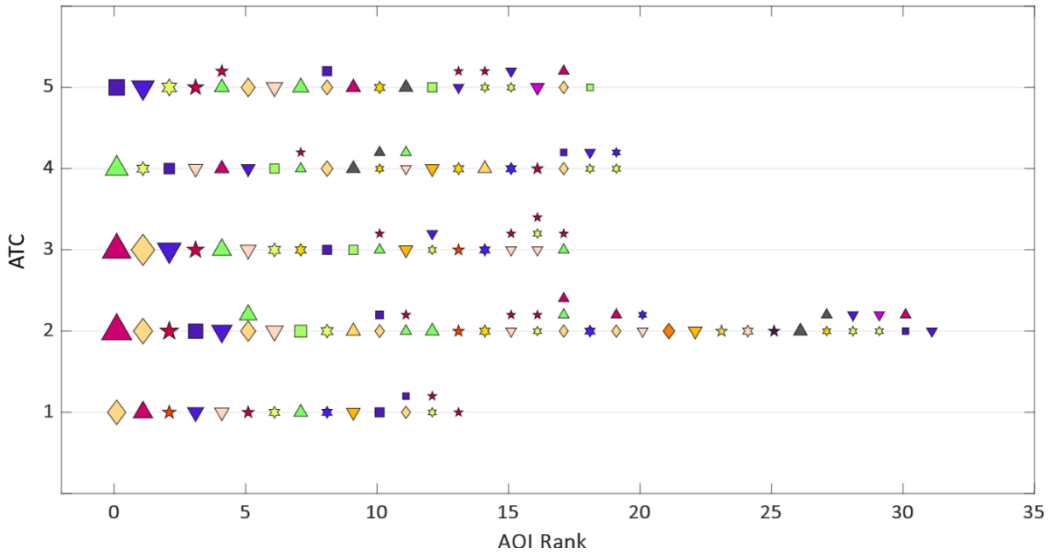
Figure 15: Shape and color code used to represent various AOIs in the order plot.

Figure 16 shows the order plot that represents the relative importance of various AOIs (both singular and overlapped states) for five expert ATCs and for all four time segments. The figure shows the ranking based on the Indegree measure. Note that other importance measures can also be used, i.e. Closeness, Betweenness. Note that, for demonstration purpose, only five expert ATCs data is shown, although, more ATCs data can also be visualized.

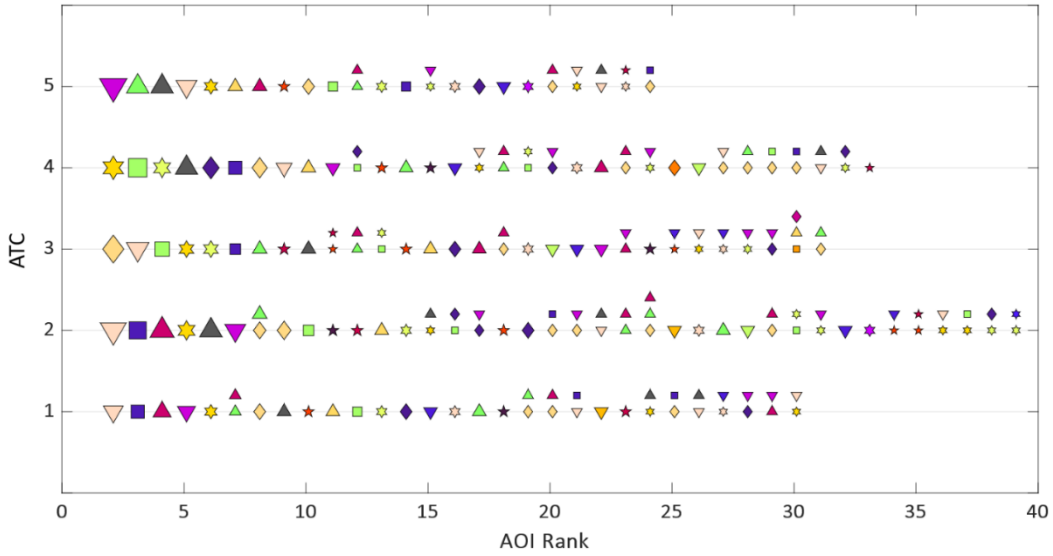
Figure 16(a) shows a clear pattern in the relative importance of AOIs among all the expert ATCs, i.e. AOI ***F*** and ***Q*** are very important in the visual scanning strategy of all the expert ATCs shown. AOI ***F*** holds the rank of either 1 or 2 for all five cases, and AOI ***Q*** holds the rank 3 in four among five cases. Another very significant AOI is ***b***. As a move ahead in the timeline, as shown in Figure 16 (b)-(d), we observe a large number of AOIs (both singular and overlapped cases). Despite a large number of unique AOI states, interestingly, we can still identify some specific AOIs that are significant across various observers, e.g. AOI ***K*** and AOI ***e*** in Figure 16 (b), AOI ***M***, ***I***, and ***P*** in Figure 16(c), and AOI ***l*** and ***S*** in Figure 16(d). Another interesting thing to note is that some specific AOIs appear to be significant both as a singular and overlapped state, e.g. AOI ***G***, ***f*** , and ***l*** (see Figure 16(d)).



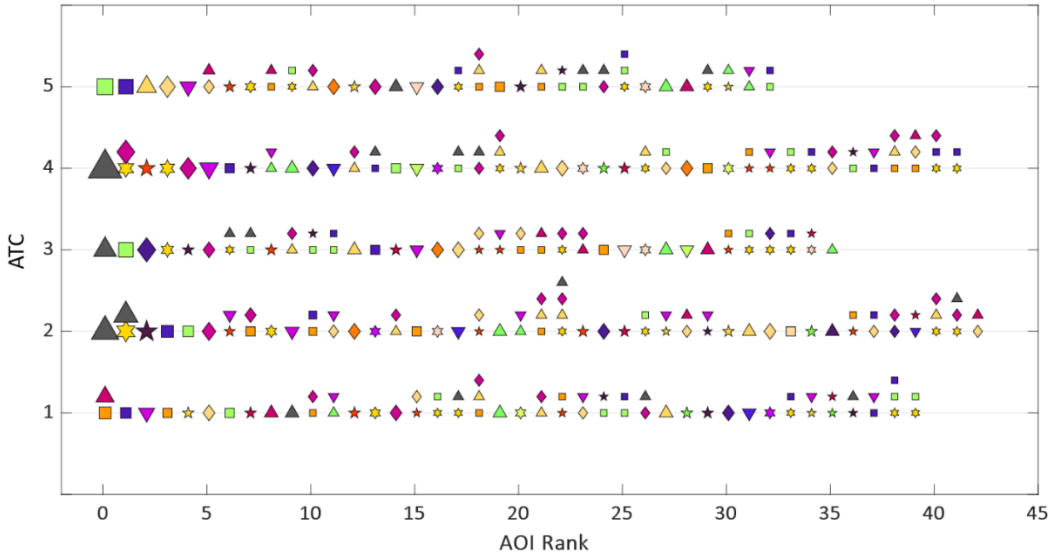
(a) 0-5 min interval



(a) 5-10 min interval



(c) 10-15 min interval



(a) 15-20 min interval

Figure 16: Order plot showing the ranking of various AOIs (singular and overlapped AOI states) for four time intervals in terms of visual scanning strategy of five expert ATCs. The present example only shows the ranking based on the Indegree measure.

3.5 Discussion

In this research, two novel visualization methods, (i) norm-cube, and (ii) order plot, are developed to represent the target importance in terms of the visual scanning strategy of

observers involved in a dynamic target tracking task. These developed frameworks are based on the dynamic DWN-based modelling of eye movement data of observers (ATCs for this specific research).

The norm-cube plot allows visualization of the overall target importance, considering three importance measures simultaneously. This enables the reader to comprehend how important a target is not only in terms of the amount of direct visual attention it receives (i.e. Indegree measure), but also how significant it is as far as the flow of visual attention is concerned, and also what critical role a target plays in forming a bridging link between different groups of targets which otherwise are not connected in terms of scanning strategy employed by the observer. The results from the norm-cube plots show how the evolution of the visual scanning strategy was happening with time as the important targets were changing as we are moving from the first to the last time interval of analysis.

Order plot framework allows visualization of multiple observes data simultaneously, although for one target importance measure only. This is useful as it provides a visual exploration and comparison tool, where the reader can see which targets/AOIs are significant across different observers' visual scanning strategy. The results from the order plot were useful as it provided a very comprehensive view of the relative ranking of the same AOI/ target across the diverse visual scanning strategy. Besides, simple visual exploration also allowed for a clear understanding of which are the very crucial AOIs, as they appear having very row ranks both as a singular state and in overlapping states with other AOIs.

Both these visualization frameworks might provide an easy to use tool to understand which are the important targets to focus upon for different time intervals, and eventually help novice ATCs to effectively modify their visual scanning strategy so that they also attend to these important targets.

3.6 Limitation and Future work

One major limitation of the norm-cube plot is that, given the current format, it can only visualize one observer data. Also, we need multiple norm-cube plots to showcase the evolution of the target importance values. Furthermore, the norm-cube plot being a three-dimensional representation, it becomes difficult to provide visualize it one a two-dimensional paper. As a result, software making it possible to change the angle of view of the norm-cube plot will further enhance its utility in understanding the overall importance of targets more comprehensibly.

On the other hand, the major limitation of the order plot framework is that we can show the relative ranking of the targets based on only one target importance measure. Moreover, as the number of AOIs increases, more shaper and color code scheme needs to be generated; as a result, the reader needs to go back-n-forth between the coding scheme and the plot to understand the relative ordering of the AOIs and also compare the similarity across ATCs.

For future research, in case of the norm-cube plot framework, we can omit insignificant AOIs (i.e. AOIs very close to the origin) by selecting a threshold distance from origin below which AOIs will not be visualized. This will reduce visual clutter for cases where there are many AOIs/targets to visualize. Besides, this will also allow to include multiple observers' data, where a different color scheme can be used to indicate

different observers (e.g. ATCs). This will enable a visual comparison of the visual scanning strategy of multiple observers, where all three target importance measures are incorporated simultaneously.

For the case of order plot, we might consider developing a filtering mechanism that is can show results on the go, i.e. a software tool will allow us to show results dependent on various filtering criteria, e.g. showing the relative position of only selected AOIs across different observers. Many other filtering options can also be included. This will enable improved visual exploration of the results obtained.

Chapter 4: Simplifying Visual Scanpath Representation to Facilitate Training in Dynamic Target Tracking Tasks: Spatial-Temporal Clustering Approach

NOTE: This chapter has been submitted for publication in IEEE Transaction for Human-Machine Systems. Therefore, the contents are directly taken from the submitted manuscript. Although to keep the reference style consistent throughout the dissertation, the in-text and references section has been changed to APA format.

4.1 Introduction

As the saying goes, ‘it takes two to tango; similarly, for safe and efficient air travel management it takes both experienced pilots and expert air traffic controllers (ATCs). Expert ATCs form the major underpinning component in providing both timely and safe navigation of flights, even more so in case of ever-changing and severe weather conditions (e.g. heavy thunderstorms, rapid changes in wind speed and pressure). A recent study by the Federal Aviation Administration (FAA) estimates that by 2040, the air traffic volume will reach 60 million from the current value of 40 million (Federal Aviation Administration, 2018). However, adding to our concern, the study by Hampton (Hampton, 2016) shows that, given the prevalent training approach of ATCs, which takes very long to complete, the projected available workforce of expert ATCs might fall short to maintain the same safety standards and service levels for this anticipated high traffic volume. Thus,

it becomes imperative for the FAA to explore more efficient training methods for their candidates, both in terms of cost and time.

Note that, irrespective of ATCs' workstation, i.e. enroute (very high altitude) or airport tower control (very low altitude), one substantial aspect of their job comprises of visual scanning. Depending on the altitude, the controllers are either monitoring the air traffic by visually scanning the radar display (in case of enroute traffic or for arriving traffic to the airport) or by looking outside the tower window trough naked eyes (in case of looking for the ground traffic in the airport area). Due to this integral nature of the visual scanning activity associated with the air traffic control operations, it becomes essential to analyze and understand their eye movements. This understanding will pave the way towards formalization of the expert's visual scanning strategy and in turn, will help to make the training process of novice ATCs more efficient. Researches in Kang and Landry (2014), Mcclung and Kang (2016), Mandal et al. (2016), and Mandal and Kang (2018) show that eye-tracking technology can facilitate the understanding of expert ATCs' visual scanning strategy while they monitor and control air traffic. In detail, Kang and Landry (2014) showed the potential application of experts' visual scanpath (VS), the time-ordered sequence of eye fixations, as a training tool to increase conflict detection efficiency for novice ATCs. Two other recent studies, Mandal et al. (2016) and Mandal and Kang (2018) demonstrated how a network-based VS analysis and visualization can help us in understanding the cognitive processes of expert ATCs involved in enroute air traffic control tasks. Propelled by these findings, the present study aims to analyze the VSs of expert ATCs involved in airport control tower operation, also known as local controllers (LCs).

LCs are mainly responsible for a smooth flow of traffic, both inbound and outbound from the airport region, e.g. providing landing and takeoff clearances to incoming and outgoing aircraft respectively. Besides, they also provide instructions for both aircraft and other vehicle movements on or across runways. Figure 17 shows an example of a typical LC's viewing field, where the computer screens represent the various radar display (also called 'inside the tower view'). The radar displays are used by LCs to see the incoming and outgoing aircraft which are otherwise invisible to naked eye. Additionally, the LCs investigate the runway region, using a combination of radar display and naked eyes, which is also called 'out of the window view' (shown by the yellow dotted boundary in Figure 17).

The two major challenges in the analysis of LCs' EM data because, first, raw EM data is very complex to visualize, even for a meaningful small duration (e.g. task completion time of 10 or 20 s); and second, in the absence of any predefined order of visual scanning, each LC employ a variety of scanning strategies to manage traffic, making it challenging to extract relevant patterns that can be taught.

Mcclung and Kang (2016) and Palma Fraga et al. (2018) examined the enroute ATCs' VS patterns and classified them into several groups. However, note that the observed VS patterns are very much correlated with the spatial distribution of the targets on the display, i.e. aircraft representations on the enroute radar display. Therefore, change in scenario context may lead to a change in observed VS patterns. Also, unlike the enroute case, the traffic flow is very different for the airport control tower, where aircraft have designated arrival and departure location, and there exist numerous other operations, e.g. aircraft move, line up and wait on in and around the taxiway. Furthermore, our emphasis is on VSs

associated with various clearance commands executed by LCs, e.g. landing, takeoff. This is useful for developing enhanced training materials for novice LCs because, ability to map VSs to the commands will facilitate in our understanding of the context behind the observed visual scanning strategy, which otherwise is very difficult to articulate even for experts.

There exist only a few previous pieces of research that deal with control tower traffic studies and especially analyzing LCs' visual scan paths. The study by Chhaya et al. (2018) talks about developing a better training curriculum for novice controllers, however, they focus on developing better training simulation scenarios. Muthumanickam et al. (2019) analyzed the evolution of expert ATCs' visual scanning strategy, leading to the change in areas of interests on the display, using a hierarchical clustering technique for long hour remote tower simulation studies. Although pertinent in understanding the visual scanning strategy over long durations this analysis doesn't particularly provide many insights in understanding the visual scan paths of expert controllers. Manske and Schier (2015) undertook a control tower simulation study where they evaluated the probability of LCs' fixating on various regions of the display during providing clearance commands. Similarly, Svensson (2015) provided a simple description of LCs' EM data while giving various clearances. Another study by Li et al. (2018) undertook a multiple tower simulation study. These previous studies were limited to the analysis of basic EM measures, e.g. eye fixation count and duration, pupil size, and saccade amplitude. Also note that, in absence of any clear sequential order associated with the visual scanning task of LCs, it is found that variety of scanning strategies exists, giving rise to non-homogenous VS among expert ATCs (Kang & Landry, 2014). Consolidating these observations, we can infer that,

although the basic EM measures are useful in providing a broad understanding of the LCs' visual scanning strategy, in absence of any visual scanpath analysis, the results obtained are not sufficient to facilitate the development of training materials for novice LCs.

In this research, we focus on analyzing the VS of LCs just before giving clearance commands. Moreover, we also propose an intuitive VS representation approach, which abstracts the complex EM data into a simple representation that enables understanding of the underlying VS pattern easily.

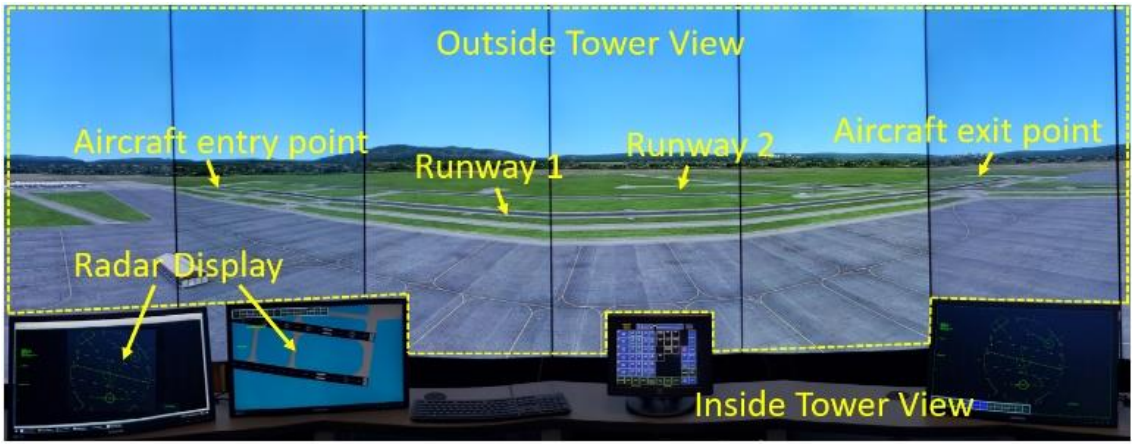


Figure 17: Example figure showing the view of the LCs from inside the airport control tower.

To address these above-mentioned issues, we developed a novel framework for VS simplification using a spatial-temporal clustering method. Moreover, the framework also provides a decision-making tool that facilitates selecting optimal parameter values for the clustering method while considering the scenario context to make meaningful VS simplification. Furthermore, the developed framework also includes a time-ordered color scheme attribute to the simplified Vs which further enhances their visualization.

For validation purposes, we implemented our approach to simplify VSs obtained from a mixed reality airport control tower experiment involving expert LCs.

The rest of the paper is organized as follows. Section 4.2 describes the details of the various algorithms used in the proposed VS simplification framework. Section 4.3 discusses the various steps involved in implementing the framework, from obtaining raw VSs to developing its simplified representation. Section 4.4 talks about the experimental process involved in obtained expert ATCs' raw VS. Section 4.5 and 4.6 provides the results obtained and their discussion respectively.

4.2 Design Concepts

4.2.1 ST-DBSCAN

ST-DBSCAN, an extension of well-know DBSCAN method, is a density-based clustering method to discover arbitrary shaped clusters in data, where a cluster is defined as contiguous regions that have densely packed data points (i.e. high-density regions), and such different clusters are separated with contiguous regions with loosely packed data points (i.e. low-density regions) (Ester, Kriegel, & Sander, 1996). ST-DBCAN, unlike DBSCAN, allows multiple non-commensurable attributes (e.g. spatial and temporal attributes of EM data) to be used as distance measures for clustering (Birant & Kut, 2007). To cluster EM data, ST-DBSCAN method needs three input values: (i) θ_s: spatial distance threshold (in pixels); (ii) θ_t: temporal distance threshold (in milliseconds); and (iii) P_{min}: minimum number of data points to define a dense region.

The algorithm starts with a data sample p and evaluates the total number of data points within the δ-neighborhood of it, i.e. data points within a distance of δ from p. Mathematically, it's the cardinality of the neighborhood set, i.e. $|N_\delta(p)|$, where $N_\delta(p) = \{x: D(p,x) \leq \delta\}$ and $D(p,x)$ is the distance (Euclidean or other metric-based) between point p and x. For two distance attributes (i.e. spatial and temporal), we get $N_\delta(p) =$

$\{N_{\delta=\theta_s}(p) \cap N_{\delta=\theta_t}(p)\}$, where θ_s and θ_t are spatial and temporal distance threshold values respectively. If $|N_\delta(p)| \geq P_{min}$, p is considered a core point and is assigned to a new cluster, if not already been assigned to a preexisting one. Subsequently, all points in $N_\delta(p)$ are assigned to the cluster p belongs. However, if $|N_\delta(p)| < P_{min}$ and it doesn't contain a core point, it's termed an outlier or noise point. Next, the algorithm moves to the next point in the sample and repeats the neighborhood calculation and cluster assignment process mentioned above till all points in the sample are evaluated. In summary, if point p, either core or border point (see Figure 18) belongs to a cluster, then all points density connected to it also belongs to the same cluster. For a more detailed analysis of ST-DBSCAN, refer Birant and Kut (2007).

For our present research, the VS data can be considered a time-ordered sequence of eye fixation points ($P_1, P_2, \ldots, P_N$), where P_i is the i^{th} eye fixation. Each eye fixation having spatial and temporal attributes can be written as $P_i = (S_i, T_i)$, where, $S_i = (x_i, y_i)$ represents the spatial location, with x_i and y_i as the horizontal and vertical position respectively, and $T_i = (t_{s,i}, t_{e,i})$ showing the temporal position, with $t_{s,i}$ and $t_{e,i}$ as the starting and ending time respectively. Thus, the duration of the i^{th} eye fixation is $\Delta t_i = t_{e,i} - t_{s,i}$. Therefore, spatial distance (D_s) and temporal distance (D_t) between the i^{th} and j^{th} eye fixations (where $j > i$) can be written as follows:

$$D_s(i,j) = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} \tag{6}$$

$$D_t(i,j) = |t_{s,j} - t_{e,i}| \tag{7}$$

Using (6) and (7), the spatial neighborhood of eye fixation P_i is $N_{\delta=\theta_s}(P_i) = \{P_j : D_s(i,j) \leq \theta_s, j \neq i\}$, and the temporal neighborhood is $N_{\delta=\theta_t}(P_i) = \{P_j : D_t(i,j) \leq \theta_t, j \neq i\}$. Thus, we get,

$$N_{\delta}(P_i) = N_{\delta=\theta_s}(P_i) \cap N_{\delta=\theta_t}(P_i) \quad (8)$$

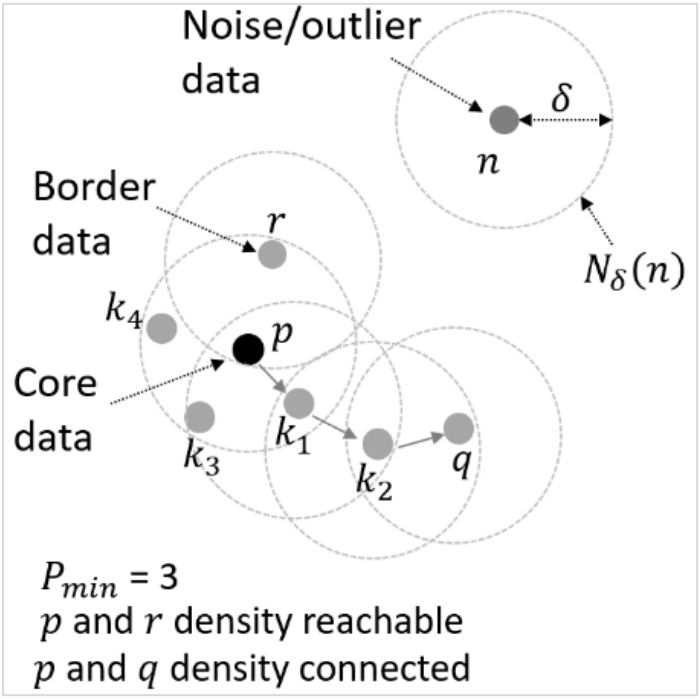


Figure 18: Visual representation of various terminologies associated with ST-DBSCAN.

Figure 18 shows that $\boldsymbol{p}$ is a core data because $|\boldsymbol{N_{\delta}(p)}| \geq \boldsymbol{P_{min}}$; $\boldsymbol{r}$ is a border data because $|\boldsymbol{N_{\delta}(r)}| < \boldsymbol{P_{min}}$ but $\boldsymbol{p} \in \boldsymbol{N_{\delta}(r)}$, where $\boldsymbol{p}$ is a core data; and $\boldsymbol{n}$ is a noise/outlier because $|\boldsymbol{N_{\delta}(n)}| < \boldsymbol{P_{min}}$ and $\forall \boldsymbol{p} \in \boldsymbol{K}, \boldsymbol{p} \notin \boldsymbol{N_{\delta}(n)}$, where $\boldsymbol{K}$ is the set of core data. $\boldsymbol{r}$ and $\boldsymbol{p}$ are density reachable as both the data instances belong to the $\boldsymbol{\delta}$-neighborhood of the other. $\boldsymbol{q}$ and $\boldsymbol{p}$ are density connected because there exists a chain of data instances (i.e. $\boldsymbol{k_1}, \boldsymbol{k_2}$) with $\boldsymbol{p}$ and $\boldsymbol{q}$ as the first and last element (or vice-versa) of the chain respectively and the consecutive elements of the chain are density reachable.

4.2.2 Dynamic Time Warping distance measure

Dynamic time warping (DTW) is a method to calculate the similarity between two temporal sequences of varying lengths (Bemdt & James, 1994; Jeong et al., 2011). This is very useful for the present research as LCs' employ visual scanning strategies which are non-homogenous both in terms of pattern and length (Kang & Landry, 2014; Mcclung & Kang, 2016). Therefore, many previous studies have used DTW to measure the spatial similarity between VSs (Le Meur & Liu, 2015; Li & Chen, 2018). In detail, DTW method employs a non-linear sequence alignment (see Figure 19) to account for the phase difference between two temporal sequences. The left side in Figure 19 shows the linear (one-one) mapping between the various elements of two sequences (shown in grey dotted lines). Unlike this, the right side in Figure 19 shows the non-linear (many-to-many) alignment of elements of the two sequences. DTW method then creates a cost matrix associated with all possible non-linear alignments between all the elements of the two sequences. The cost associated for each such alignment is calculated using the absolute difference between the spatial location of the elements (e.g. Euclidean distance). A warping path in this cost matrix (i.e. starting from upper right corner cell and ending at lower left corner cell), associated with the least traversal cost is selected as the DTW distance between the two given sequences. For example, given two VSs, $A = \{a_1, a_2, \dots, a_N\}$ and $B = \{b_1, b_2, \dots, b_M\}$, the cost matrix $DTW \in \mathbb{R}^{N \times M}$ is evaluated using the recursive formula given below:

$$DTW(i,j) = D(i,j) + min\begin{cases} DTW(i,j-1) \\ DTW(i-1,j) \\ DTW(i-1,j-1) \end{cases} \quad (9)$$

Where, $1 \leq i \leq N$ and $1 \leq j \leq M$, and $D(i,j)$ is the Euclidean distance between a_i and b_j. i.e. i^{th} and j^{th} eye fixation in VSs A and B respectively. Figure 20 shows an example cost matrix between VSs A and B. The warping path starts from the top right corner, i.e. cell (N,M), moves along other cells, and ends at the bottom left corner of the matrix, i.e. cell (1,1). Further, Figure 20 shows the various allowed movements for developing the warping math, e.g. at cell (i,j) only three movement directions are permitted: (i) left horizontal or cell (i,j-1), (ii) down vertical, i.e. cell (i-1,j-1), and (iii) below diagonal, i.e. cell (i-1,j). The direction of movement depends on which of the three cell has the minimum cost. The warping path is created by joining all the cells traversed from start to end, and the DTW distance between VSs A and B is the summation of the cost of each cell forming the warping path. Mathematically, if the warping path is $W = \{w_1, w_2, .. w_L\}$ where, $w_l = DTW(a_i, b_j)$, $1 \leq l \leq L$, and $max(N,M) \leq L \leq (N+M)$ (Liu et al., 2019), the DTW distance between VSs A and B is as follows:

$$DTW(A,B) = \sum_{l=1}^{L} w_l \tag{10}$$

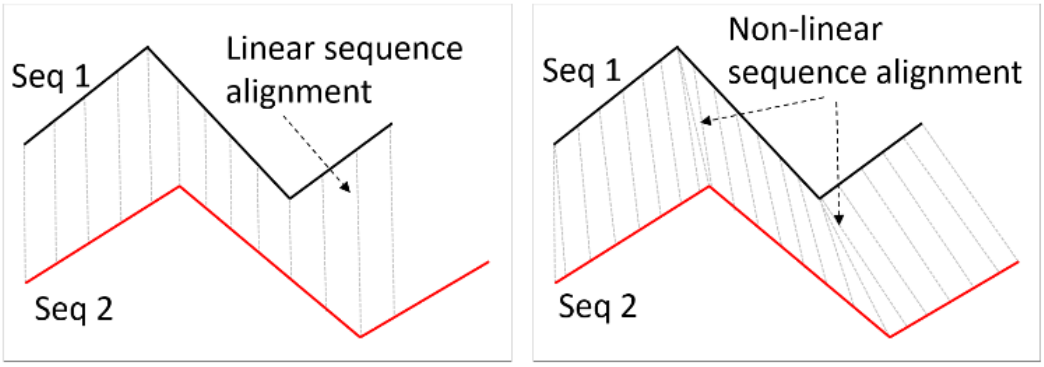


Figure 19: Linear and non-linear sequence alignment process examples. Seq 1 and Seq 2 represent two sequences 1 (black colored) and 2 (red colored). The grey dotted lines show the alignment (either linear or non-linear) between the various elements of the two sequences.

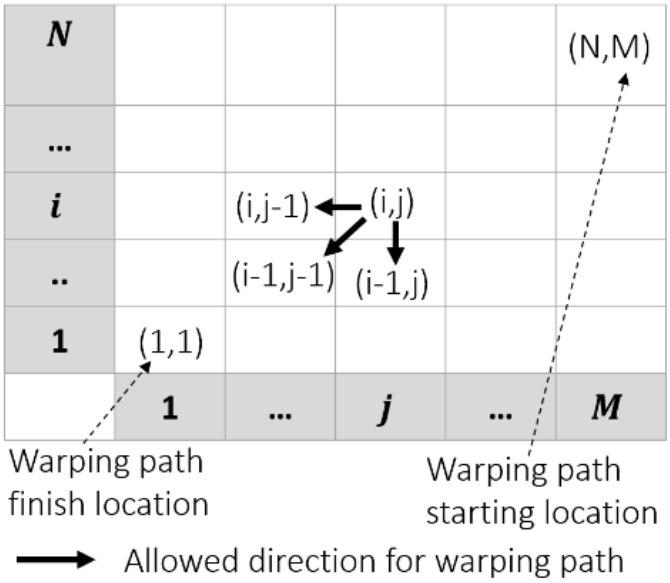


Figure 20: Example of a cost matrix showing the various allowed movements while creating the warping path for evaluating the DTW distance between two VSs of size N and M.

4.3 Methods

Figure 22 shows that methodological flowchart to develop the simplified VS representation from the raw VS of the expert ATCs. The various steps involved are described below in detail.

Step 1: Collect raw VS for various participants

The first step involves collecting the raw VS data of various participants as obtained from commercial eye trackers. The VS for the k^{th} participant can be written as $VS_k = (P_{k,1}, P_{k,2}, \dots, P_{k,N})$, where $P_{k,i}$ is the i^{th} eye fixation for the k^{th} participant. $P_{k,i} = (S_{k,i}, T_{k,i})$, where $S_{k,i} = (x_{k,i}, y_{k,i})$ and $T_i = (t_{k,s,i}, t_{k,e,i})$ are the spatial and temporal position of the eye fixations.

Step 2: Apply spatial-temporal clustering to raw VSs for a given parameter setting

The second step involves applying the spatial-temporal clustering method on the raw VSs, for a given spatial and temporal threshold values, to obtain their simplified

representation. The clustering method transforms a raw VS into its simplified representation, which is a time-ordered sequence of clusters of eye fixations. If $VS_k = (P_{k,1}, P_{k,2}, \dots, P_{k,N})$ represents the raw VS for the k^{th} participant, then its simplified form can be represented as $\widehat{VS_k} = (C_{k,1}, C_{k,2}, \dots, C_{k,M})$, where $C_{k,j}$ is the j^{th} clustered eye fixation point, $1 \leq j \leq M$, and M is the total number of clusters ($M < N$). Each clustered eye fixation point $C_{k,j}$ is an aggregate of its constituent eye fixation points. In detail, if $C_{k,j}$ is the aggregate of w eye fixations, starting from index r i.e. $P_{k,r}, P_{k,r+1}, \dots, P_{k,r+(w-1)}$, then we get the following:

$$C_{k,j} = (\hat{S}_{k,j}, \hat{T}_{k,j}) = P_{k,r} \odot P_{k,r+1} \odot \dots \odot P_{k,r+(w-1)}$$

Where $\odot$ represents the aggregation operation, and $\hat{S}_{k,j}$ and $\hat{T}_{k,j}$ are the spatial and temporal attributes of $C_{k,j}$ respectively. $\hat{S}_{k,j}$ is the average of the spatial attributes of the component eye fixations, and $\hat{T}_{k,j}$ is the cumulative duration of the constituent eye fixations. Mathematically, the aggregation can be written as follows:

$$\hat{S}_{k,j} = (\hat{x}_{k,j}, \hat{y}_{k,j})$$

where, $\hat{x}_{k,j} = \frac{1}{w} \sum_{r=1}^{w} x_{k,r}$ and $\hat{y}_{k,j} = \frac{1}{w} \sum_{r=1}^{w} y_{k,r}$ (11)

$$\hat{T}_{k,j} = (\hat{t}_{k,s,j}, \hat{t}_{k,e,j})$$

where, $\hat{t}_{k,s,j} = t_{k,s,r}$ and $\hat{t}_{k,e,j} = t_{k,e,r+w-1}$ (12)

Step 3: Generate simplified VSs

The output of this step is the simplified VSs, obtained after applying the clustering method described previously.

Step 4: Calculate normalized DTW distance and # of eye fixations for the simplified VSs

Once the simplified VS is obtained, the next step involves two tasks: (i) evaluating how many eye fixations it contains (i.e. aggregated eye fixations), and (ii) how different it is from the raw VS used to generate it. The first task requires calculating the cardinality of the simplified VS, i.e. $|\widehat{VS_k}|$. For the second task, the DTW distance between the raw VS and its simplified representation quantifies how different two VSs are. For this purpose, we use the DTW method described earlier (see Section 4.2).

If we have K participants, the set of all VSs can be written as $VS = \{VS_1, VS_2, \ldots, VS_K\}$. The DTW distance for the k^{th} participant for a given parameter setting Φ can be written as $DTW_{\Phi,\mathrm{k}} = DTW_{\Phi}\left(VS_k, \widehat{VS}_k\right)$ and $DTW_{\Phi} = \{DTW_{\Phi,\mathrm{k}}\}_{k=1}^{K}$ is the set of DTW distance for K participants, for a given setting Φ. To compare across various Φ values (note that different Φ results in simplified VSs of different lengths), it's advisable to use the normalized DTW values, which is written as follows:

$$\overline{DTW}_{\Phi,\mathrm{k}} = \frac{DTW_{\Phi,\mathrm{k}}}{|\widehat{VS_k}|+|VS_K|} \tag{13}$$

Where, $|\widehat{VS_k}| + |VS_K|$ is the sum of the cardinality of the VS and its simplified representation for the k^{th} participant. Subsequently, the average DTW distance, given parameter setting Φ, for K participants can be written as follows:

$$\mu_{\overline{DTW}_{\Phi}} = \frac{1}{K}\sum_{k=1}^{K} \overline{DTW}_{\Phi,\mathrm{k}} \tag{14}$$

Similarly, the average number of eye fixations, given parameter setting Φ, for K participants can be calculated as follows:

$$\mu_{EF_{\Phi}} = \frac{1}{K}\sum_{k=1}^{K} |\widehat{VS_k}| \tag{15}$$

Where, $|\widehat{VS_k}|$ is the cardinality of the simplified VS representation for the k^{th} participant.

Step 5: Plot both average normalized DTW distance and # of eye fixations for all parameter settings

If we consider S and T different numbers of spatial and temporal threshold respectively, then we have $S \times T$ number of different parameter settings for the clustering method. Thus, the parameter set can be written as $\Phi = \{\Phi_l\}_{l=1}^{S\times T}$ where, $\Phi_l = (\theta^s{}_i, \theta^T{}_j)$, where $1 \leq i \leq S$ and $1 \leq j \leq T$. Therefore, we will have one average DTW value ($\mu_{\overline{DTW}_\Phi}$) and one average eye fixation number (μ_{EF_Φ}) for each parameter setting. Hence, this step involves plotting $\mu_{\overline{DTW}_\Phi}$ and μ_{EF_Φ} for all different parameter settings.

Step 6: Select best parameter setting

It's challenging to develop any single analytical function to find the best parameter values (both spatial and temporal) for the clustering method to be applied in any general context. Therefore, one feasible way to find context-specific optimal parameter values is to take the empirical analysis route, i.e. analyze how $\mu_{\overline{DTW}_\Phi}$ and μ_{EF_Φ} change with change in parameter values. Afterwards, we can use domain knowledge to select the optimal parameter values that serve our central purpose, i.e. enhanced understanding of the visual scanning strategy while simplifying the VS representation. Therefore, parameter values that substantially reduces the information content of the simplified VS (i.e. reduces

μ_{EF_Φ}substantially), while keeping it similar to the raw VS (i.e. low $\mu_{\overline{DTW}_\Phi}$) would be a desirable choice.

Step 7: Add time order-based color scheme

Once the best spatial (θ^s) and temporal (θ^t) threshold values are selected, the next step involves adding a time-ordered color scheme to the VS representation. To this purpose, a sequential multihued color scale is developed, where yellow color suggests past eye fixations and red color implies most immediate eye fixations. Adding this color scheme to the simplified VS representation further enhances its interpretability, as it allows the easy distinction of overlapping eye fixations without the observer relying on the eye fixation index numbers, which are otherwise difficult to comprehend for overlapping eye fixations case. Furthermore, a black color border attribute is also added to the clustered eye fixation representation in the simplified VS, to allow easier identification of the aggregated eye fixations (i.e. clustering locations).

Step 8: Visualize simplified VS

The last step involves visualizing the simplified VS, showing the aggregated eye fixations and the saccades joining them. Figure 22 shows an example where a raw VS along with its simplified representation having the time-ordered color scheme has been visualized.

4.4. Experimental Process

4.4.1 Participants

Ten expert ATCs participated in this data collection process. Besides, two FAA employees also participated, one as the pseudo pilot and another as the ground controller.

Although, for the present research, the results of only three expert ATCs have been shown as a proof of concept.

4.4.2 Apparatus

Hardware: Twelve 55" HD (1080p) monitors were used to display the simulation scenario. Tobii Pro Glasses 2 (Tobii Technology, 2020) (with gaze sampling frequency 100 Hz) was used to capture ATCs' gaze data.

Software: MaxSim simulators, developed by Adacel Systems Inc. (Adacel Systems, 2020), was used to develop the simulation scenario. Tobii Pro Lab (Tobii Technology, 2020) was used to process the ATCs' raw EM data. The experiment was conducted at the FAA Civil Aerospace Medical Institute in Oklahoma City, which provided all the above-mentioned hardware and software.

4.4.3 Scenario

One simulated airport control tower scenario of approx. 20 minutes was used for the experiment data collection.

4.4.4 Task

The participants were responsible for checking flight schedules, giving landing clearance commands while ensuring safe separation between incoming aircraft on runway 1 (Figure 17). Furthermore, they were to communicate with the ground controller, who was responsible for ensuring smooth and safe ground traffic flow towards and from the runways. There was one pseudo pilot (FAA employee) who oversaw the task of piloting various aircraft in the scenario and executing all ATCs' voice commands. Communications were conducted via simulated radio connection.

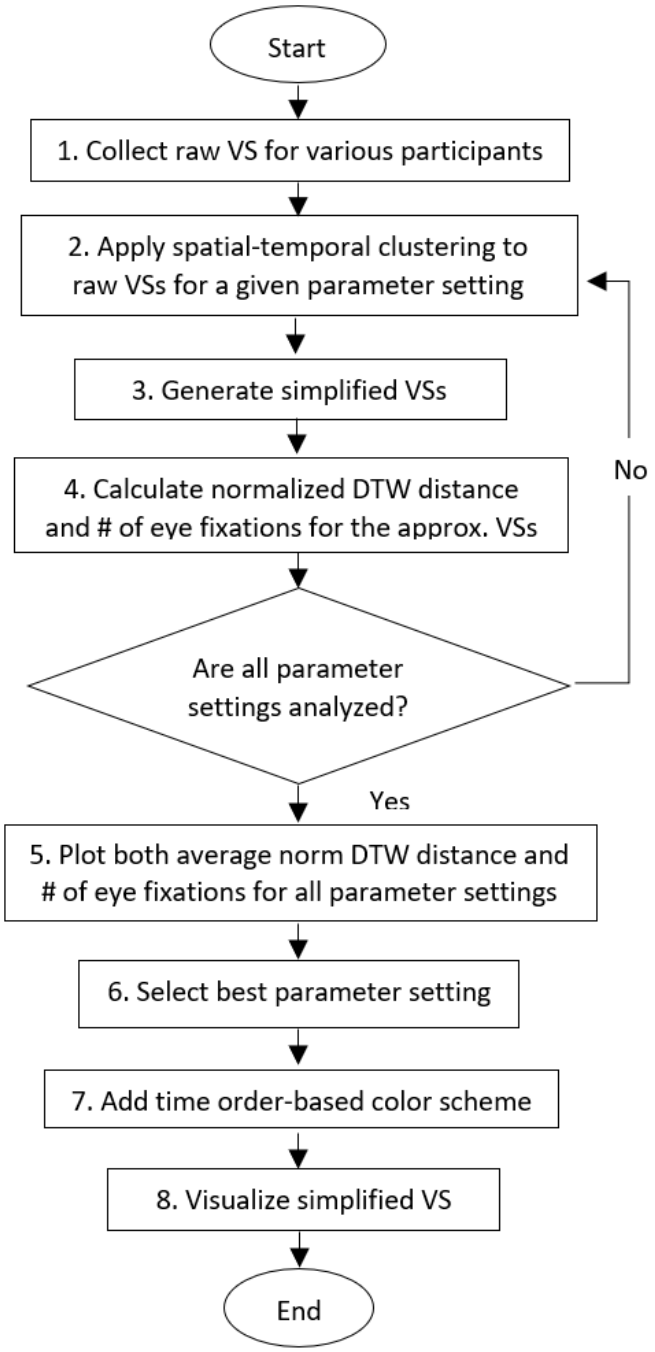


Figure 21: Methodological flowchart showing the steps involved to develop simplified VS representation

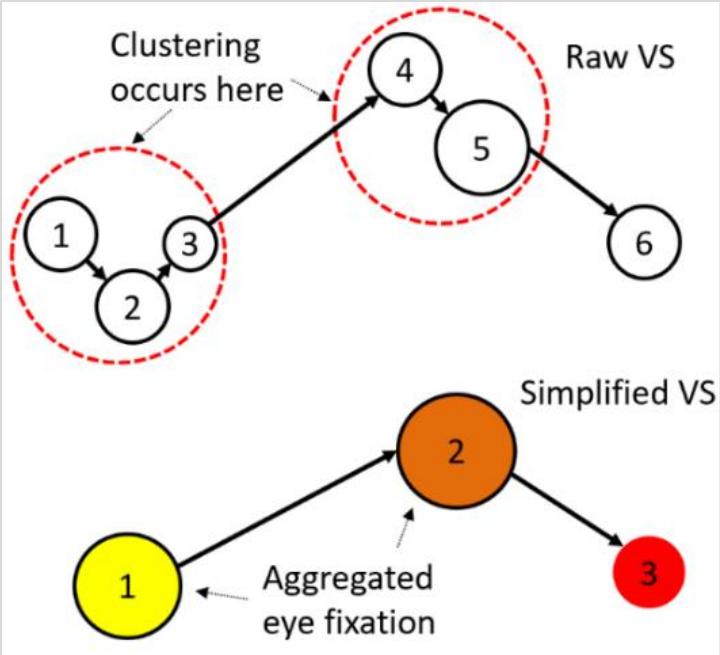


Figure 22: Example of raw VS and its simplified representation using a time-ordered color scheme. The red dotted circles in the raw VS show the spatial-temporal cluster locations. Raw eye fixations 1,2, and 3 were aggregated as 1 in simplified VS, and 4 and 5 in raw VS were aggregated as 2 in simplified VS representation. The black border around the eye fixations 1 and 2 in the simplified VS represents aggregated eye fixations. Absence of black border around eye fixation 3 (bottom figure) suggests it is a direct mapping from the raw VS.

4.4.5 Data Analysis

We've used R (R Team, 2019) to implement the spatial-temporal clustering algorithm (Hahsler & Piekenbrock, 2019) and the DTW (Giorgino, 2009) method. MATLAB 2019 (Mathworks, 2019) was used to develop the VS visualization.

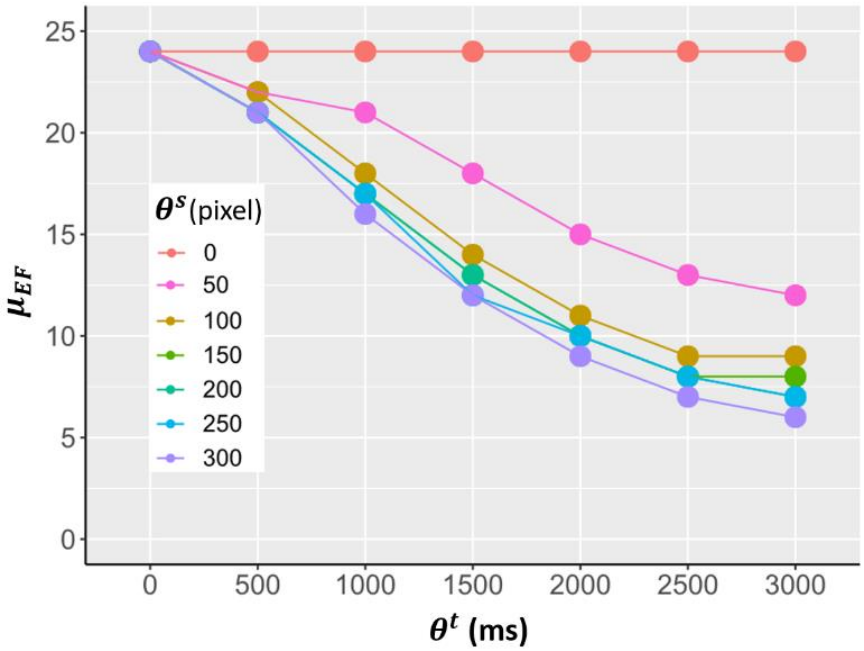
The complexity of the visual scanpath simplification method is $O(STKN^2)$ where K is the number of VSs, S is number of spatial parameters used in the ST-DBSCAN, T is the number temporal parameters used in ST-DVSCAN, and N the maximum length of the VS.

4.5 Results

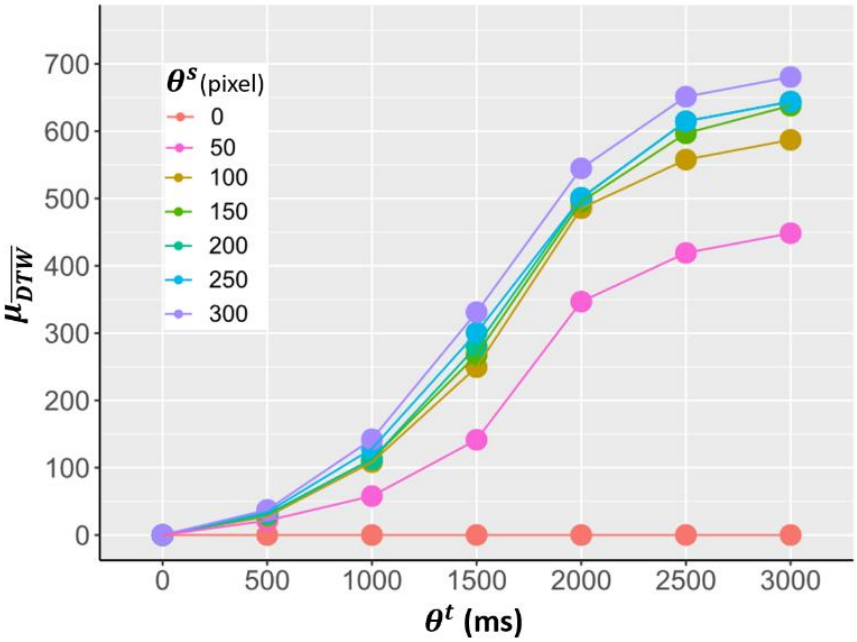
For the present research, we have used seven different values of temporal and spatial thresholds, i.e. $\theta^t = \{0, 500, 1000, 1500, 2000, 2500, 3000\}$ and $\theta^S = \{0, 50, 100, 150, 200, 250, 300\}$. Fig. 8 shows the effect of $\theta^s$and θ^tparameters on μ_{EF} and $\mu_{\overline{DTW}}$ values. Fig. 23 (a) shows that with increasing θ^t and θ^s the μ_{EF} value decreases. Furthermore, for a given θ^tvalue (e.g. 1500 ms), as the θ^schanges from 0 pixel to 50 pixels, the reduction in μ_{EF} is substantial. However, as θ^s further increases (i.e. from 100 to 300 pixels), instead of further reducing rapidly, the different μ_{EF} values seem to band together. Unlike this, Figure 23 (b) shows that, as the θ^tincreases, for a fixed θ^s, the rate of decrease for μ_{EF} remains approximately the same. Unlike this, the $\mu_{\overline{DTW}}$ value increases with increasing θ^t and θ^s. In other words, $\mu_{\overline{DTW}}$ follows a similar trend like μ_{EF} but in the opposite direction, i.e. as θ^s changes from 0 to 50 pixels, there occurs a rapid increase in the $\mu_{\overline{DTW}}$ for all different values of θ^t. Similarly, with further increase in θ^s (i.e. from 100 to 300 pixels), for a given θ^t, the rate of increase of $\mu_{\overline{DTW}}$ reduces significantly, and the values tend to be close to each other (see Figure 23 (b)). Furthermore, as θ^t changes from 500 ms to 2000 ms, the rate of increase of $\mu_{\overline{DTW}}$ is very steep; however, this rate flattens out quickly for any further increase in the θ^t value.

Note that for $\theta^t = 1500$ ms and $\theta^s = 200$ pixel, we get an average 50% reduction in the information contained in the simplified VSs, i.e. Figure 23(a) μ_{EF} = 13 for $\theta^t = 1500$ ms and $\theta^s = 200$ pixel, which is half of μ_{EF}=24 for $\theta^t = 0$ ms and $\theta^s = 0$ pixel (raw VS). Similarly, Figure 23 (b) also shows that the parameter setting of $\theta^t = 1500$ ms and $\theta^s = 200$ pixel results in a considerable simplification of the raw VS ($\mu_{\overline{DTW}}$ = 300), which is roughly 50% of the maximum simplification ($\mu_{\overline{DTW}} \approx 700$).

Notice that, the two intents behind implementing the clustering process are, first, it should lead to a substantial reduction in the information content of the VSs, and second, it should preserve the overall shape of the raw VS (such that observer can comprehend the overall pattern of the ATCs' visual scanning strategy). Combining these two criteria, in conjunction with the above-mentioned results, we choose $\theta^t = 1500$ ms and $\theta^s = 200$ pixel as the optimal parameter setting for the clustering method.



(a) effect of θ^s and θ^t on μ_{EF}

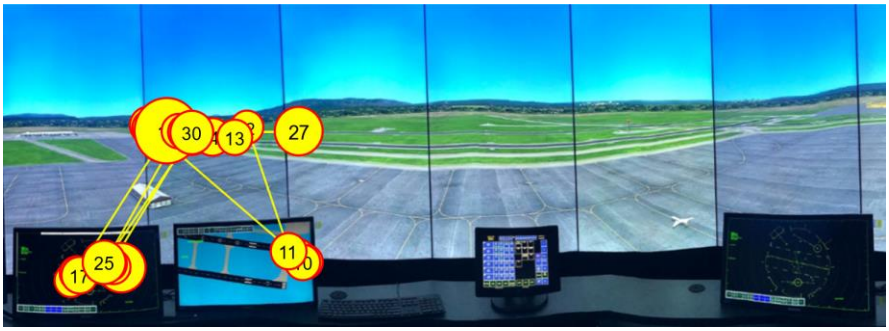


(b) effect of θ^s and θ^t on $\mu_{\overline{DTW}}$

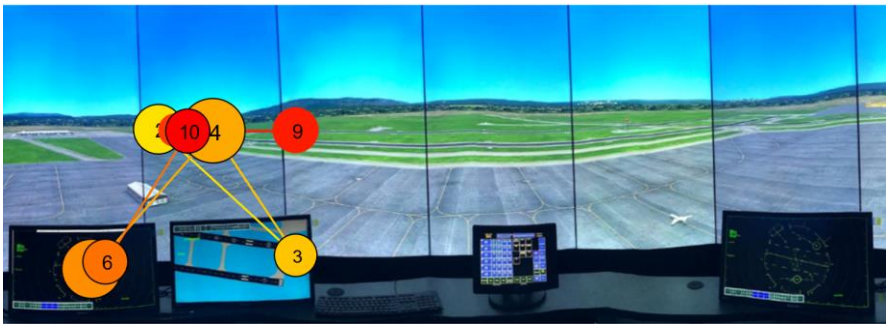
Figure 23: Effect of the spatial ($\boldsymbol{\theta^s}$) and temporal ($\boldsymbol{\theta^t}$) parameters on VS simplification process. (a) Effect on average eye fixation count ($\boldsymbol{\mu_{EF}}$), (b) Effect on average normalized DTW ($\boldsymbol{\mu_{\overline{DTW}}}$).

Figure 24 and 25 show the raw VS and its simplified representation for three ATCs for the clearance commands 'Cleared to land' and 'Cleared to take off' respectively. On visual inspection, one can notice that the spatial-temporal clustering method indeed reduces the visual clutter of the raw VSs while keeping their overall shape intact. For example, Figure 24 (a) and (b) shows that spatial-temporal clustering has indeed reduced the complexity of the raw VS (with simplified VS having only 10 eye fixations, unlike 30 in the raw VS) while keeping it an overall shape similar. Furthermore, the time-ordered color scheme allows us to locate that the ATC start their scanning from the entry point of the runway 1, makes two and movement between this location and the radar displays. Similarly, Figure 25 (a) and (b) shows that the simplified VS has significantly low eye fixations, and it preserves the overall shape of the raw VS, i.e. in both cases, the ATC

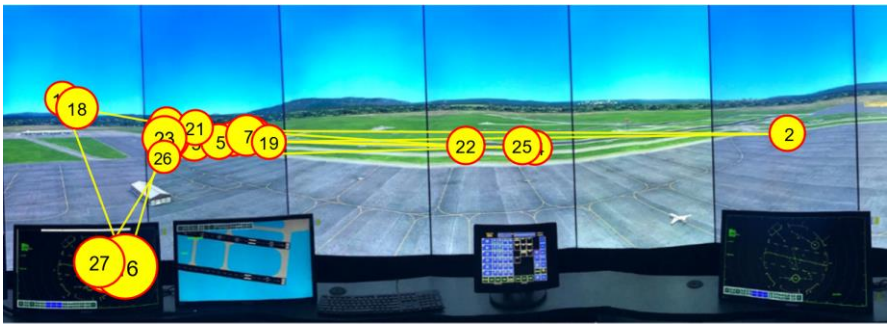
initially focuses at the radar displays, then moves the attention on the entry point on runway 1, and finally at the exit location on the runway 1.



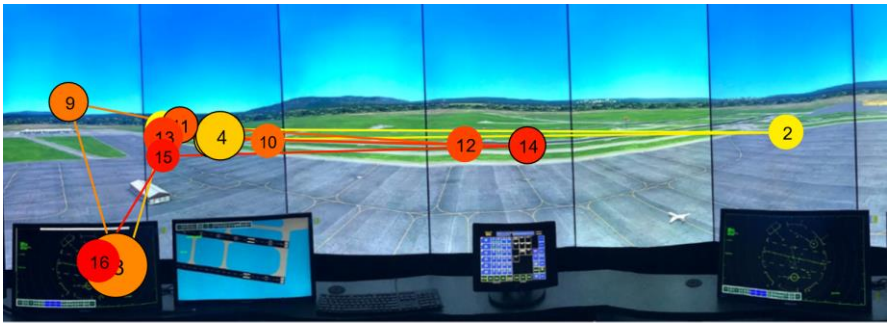
(a) Raw VS of P1



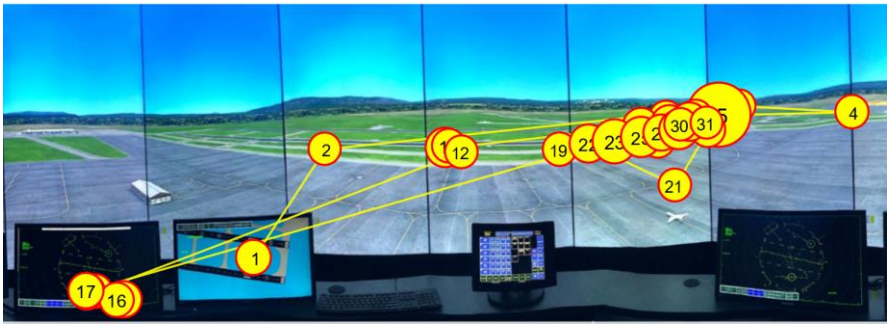
(b) Simplified VS of P1



(c) Raw VS of P2



(d) Simplified VS of P2



(e) Raw VS of P3

(f) Simplified VS P3

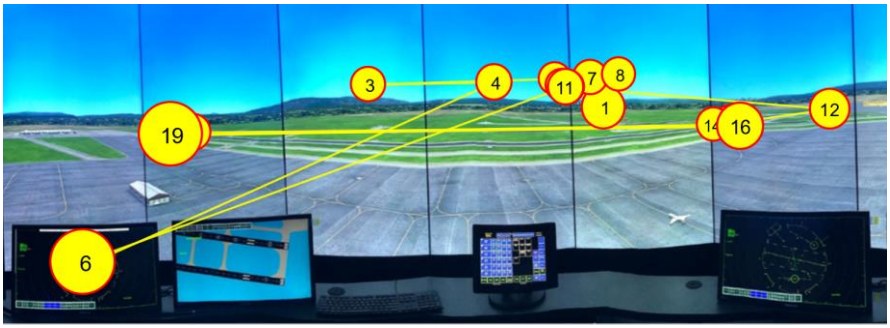
Figure 24: Raw and simplified VSs of three expert ATCs. The VSs correspond to 20 s prior to giving the clearance command 'Cleared to land'. The parameter setting of the clustering method is: $\boldsymbol{\theta^t}$=1500 ms and $\boldsymbol{\theta^s}$= 200 pixels. P1, P2, and P3 indicate ATC's index.

(a) Raw VS of P1

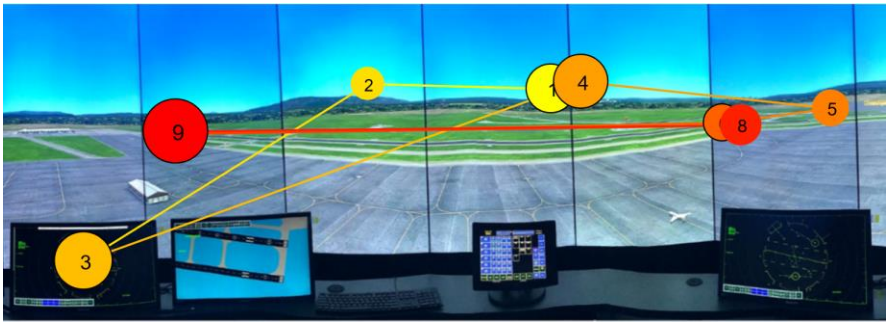
(b) Simplified VS of P1

(c) Raw VS of P2

(d) Simplified VS of P2



(e) Raw VS of P3



(f) Simplified VS of P3

Figure 25: Raw and simplified VSs of three expert ATCs. The VSs corresponds to 20 seconds prior to giving the clearance command 'Cleared to take off'. The parameter setting of the clustering method is: $\boldsymbol{\theta^t}$=1500 ms and $\boldsymbol{\theta^s}$= 200 pixels. P1, P2, and P3 indicates three ATCs.

4.6 Discussion

In this research, a methodology was developed to simplify the representation of complex VSs obtained in dynamic target tracking tasks, e.g. air traffic control tasks. The framework serves two objectives. First, it simplifies the representation of the raw VS by employing a spatial-temporal clustering method. Second, it facilitates the decision-maker to incorporate context-dependent factors for selecting the optimal parameter for the clustering method used the VS simplification. These properties allow for an approximate VS representation that can be used for better training purpose where expert ATCs' visual scanning strategy can be visualized with reduced complexity, thereby enhancing the

understanding of novices while maintaining its significant visual pattern, essential for learning.

4.6.1 VS simplification

The proposed methodology provides two measures to quantify the extent of VS simplification:(i) amount of information in the simplified VS, i.e. the number of eye fixations, and (ii) dissimilarity between the raw and simplified VS, i.e. DTW distance between the raw and simplified VS. Note that, μ_{EF} is found to be inversely proportional to both temporal (θ^t) and spatial (θ^s) thresholds. Unlike this, $\mu_{\overline{DTW}}$ is directly proportional to both θ^t and θ^s. Both these trends are in accordance to the fact that as both the threshold value increases, the associated neighborhood size for each data point (i.e. eye fixations in raw VS) also increases, leading to more distant (both temporally and spatially) eye fixations belonging to the same cluster. As a result, the amount of eye fixations in the simplified VS reduces and simultaneously its final shape becomes more dissimilar to the raw VS it is developed from, leading to increased DTW distance and $\mu_{\overline{DTW}}$. However, note that, the rate of decrease of μ_{EF} and the rate of increase in $\mu_{\overline{DTW}}$ is not constant, but it reduces as the threshold values extend beyond certain values, e.g. for θ^t>2000 ms and θ^s>200 pixels.

Furthermore, extending the threshold values beyond a certain point doesn't serve our purpose of developing a simplified yet meaningful VS. This is because oversimplification (large θ^t and θ^s values) while reducing the amount of information to process (i.e. constituent eye fixations in the simplified VS) will inevitably severely modify the shape of the raw VS, resulting in high DTW distance, thus making it difficult for the observers to learn the actual visual scanning strategy employed by the expert ATCs.

Note that the developed VS simplification methodology is different from two earlier studies in Li and Chen (2018), Li et al. (2017), and Dewhurst et al. (2012). Li and Chen (2018) and Li et al. (2017), unlike the present research that deals with individual VS simplification, provides a methodology to develop representative VS for analysis group viewing pattern. The method in (Li & Chen, 2018) and (Li et al., 2017) is not suitable for the present research as we intend to learn all possible visual scanning strategy exhibited by expert ATCs while giving clearance commands, and not an approximate group representation of experts ATCs. On the other hand, the VS simplification in (Dewhurst et al., 2012) was intended to facilitate VS similarity evaluation, where only saccade direction and amplitude was used for the simplification process. Although, the saccade amplitude is analogous to our spatial threshold, however, (Dewhurst et al., 2012) doesn't consider the temporal attributes for the VS simplification. In addition, no justification was provided for saccadic amplitude threshold of 10% of the length of the diagonal of the display size.

4.6.2 Visualization of simplified VS

The VS results show that incorporation of the novel time-ordered color attribute further enhances the interpretability of simplified VSs: (i) allowing for easier identification of overlapped eye fixations, and (ii) the observer doesn't have to solely rely on the eye fixation indexes to comprehend the time order of the VS representation.

In conclusion, we developed an objective framework for simplifying the representation of raw VSs obtained in dynamic target tracking tasks, e.g. air traffic control tasks. In detail, the developed framework initially implements a spatial-temporal clustering method, aggregating spatially and temporally nearby eye fixations, and then includes a time-ordered color scheme to develop the simplified VS representation. Besides, the

framework enables to evaluate the extent of VS simplification by quantifying two aspects: (i) amount of information reduction, and (ii) similarity between raw and simplified VS. As a result, this simplification reduces the inherent complexity (i.e. the visual clutter) associated with raw VSs and further enhances its visualization. The enhanced representation of expert ATCs' VS might aid the training process of novice ATCs by facilitating easier comprehension of the overall shape of expert ATCs' VS and their associated visual attention flow while controlling complex traffic scenarios.

4.6.3. Limitations and Future Work

One major limitation of the study is the small sample size of expert ATCs used in the experimental data collection. Also, despite the decision-making framework, we still need domain expertise to modify the parameters of the clustering method to develop context-specific meaningful simplification of the VS that is helpful for learning purposes.

For future work, we can perform empirical studies to quantify the effectiveness of the VS simplification process towards the learning of novice ATCs. Besides, the utility of the developed framework in simplifying task-specific longer VSs (e.g. 60 s, 120 s) should be studied. Furthermore, we should also consider other complex tasks, that involves a substantial amount of visual scanning, apart from air traffic control, e.g. driving.

Furthermore, previous studies have shown that novice ATCs who were shown the expert ATCs' visual scanpaths, showed an improvement in their task performance. Besides, the raw visual scanpaths of expert ATCs' are very complex to visualize even for short durations, e.g. 30 s or 60s. Thus, having a simplified and enhanced representation of the expert ATCs' raw visual scanpaths will further enhance the understanding of novice ATCs, and they can be exposed to even long duration visual scanpath patterns. Thus, the

proposed visual scanpath simplification process have potential to be used for increasing the training performance of novice ATCs.

Additionally, evaluating the effectiveness of the proposed method might require performing an experiment where the difference in the performance levels of two groups of novice ATCs can be measured; with one group having the exposure to the simplified representation of the visual scanpath expert ATCs and the other group acting as a control. The task performance can be measured using various key traffic control metrics, e.g. amount of aircraft conflict situations detected and resolved, time taken to detect the conflict between multiple aircraft pairs etc.

Chapter 5: Framework to compare visual scanning strategies for dynamic target tracking tasks: Combining n-gram and network community detection method

5.1 Introduction

Understanding the visual scanning strategy (having underlying patterns if any) of expert air traffic controllers (ATCs) forms the bedrock of developing better training materials for novice ATCs. Besides, improving the current training materials is warranted by the fact that as per the recent projection by the Federal aviation administration (FAA) the air traffic volume will increase substantially from 40 million to 60 million by 2040 (FAA, 2018). However, the available population of expert ATCs might not be sufficient to fulfil the anticipated demand of expert ATCs required to maintain the current levels of safety and efficiency of the air traffic movement.

One significant aspect of ATCs' job comprises of acquiring and processing visual information, either from a radar display (e.g. while managing enroute traffic) or by scanning outside the control tower, depending solely on naked eyes, in case of airport control tower traffic management. Therefore, analyzing expert ATCs' visual scanning strategy, looking for hidden patterns in them, might provide useful insights into the attention deployment strategies employed by expert ATCs.

The three main aspects that give rise to very complex visual scanpaths are: (i) dynamic nature of the targets, changing shape and position with time, being tracked by the ATCs'; (ii) absence of any predefined starting and ending location for attention

deployment; and (iii) existence of significant individual differences among expert ATCs in terms of visual attention deployment strategies (Kang & Landry, 2015; Mcclung & Kang, 2016). Besides, Mcclung and Kang (2016) provided several examples showcasing the heterogeneous nature of expert ATCs' scanpaths, observed while managing enroute air traffic even for a small duration. Moreover, the difficulty associated with developing a single objective framework to characterize scanpaths of the experts into intuitive geometric shapes makes it further challenging to employ the scanpaths directly into developing learning materials. Therefore, it might be useful to search for similarity among the scanpaths; thereby enabling the development of representative scanpaths, that can be used to develop better training materials for novice controllers. Furthermore, the ability to measure the similarity among the scanpaths will enable us to analyze the consistency in the decision-making of individual experts while they manage similar traffic scenarios.

The prevalent scanpath similarity evaluation methods can be broadly categorized into three categories (explained in detail in the background section, see Figure 26): first, methods comparing raw scanpaths; second, methods analyzing similarity between scanpath sequences (SSs); and third, methods evaluating the distance between eye fixation transition matrices.

Methods belonging to the first category uses the exact eye fixation locations and duration to compare scanpaths. However, with increasing target numbers and traffic flow complexity, the raw scanpaths become very diverse. Thus, it becomes challenging to find any substantial similarity among the scanpaths. Therefore, for complex tasks such as air traffic control, these methods are either too strict in their definitions or too complex to implement. On the other hand, methods associated with the third category, that employs

comparison among the transition matrices, completely ignore the order associated with the visual attention switching behavior for the ATCs. Unlike these, the scanpath analysis methods belonging to the second category compare SSs, to evaluate the similarity between the scanning strategy of ATCs. SSs are obtained from the raw scanpath data in two steps: first, dividing the display area into discrete regions or areas of interest (AOIs); and second, mapping the eye fixations with the AOIs (which are represented with character codes, e.g. 'A', 'B' etc.), followed by developing the sequence of characters.

Interestingly, unlike category three methods analyzing transition matrices, these methods incorporate the order of eye fixations while evaluating the similarity between SSs. Besides, by considering eye fixations falling on various locations inside the same AOI as semantically similar, the SS analysis methods cut down heavily the inherent complexity associated with the implementation of the methods using scanpaths directly. Due to the aforementioned reasons, in the present research, we propose similarity evaluation methods that work upon the SSs. Besides, we also require some method that can help us identify homogenous groups of highly SSs, such that the consistency of visual scanning strategy of expert ATCs can be evaluated.

To this purpose, we present a novel framework that accomplishes two objectives: (1) provides two new SS similarity measures, adapted and modified from the n-gram method frequently used in analyzing string sequences; and (2) developed an objective way to identify groups of highly similar SSs by implementing a modularity-based network-community detection method. For the present research, we have focused on analysis on the scanpaths observed before the issuance of clearance commands by expert ATCs. Although,

the proposed methodology can be easily extended to the analysis of scanpaths belonging to other operation also.

The rest of the chapter is organized as follows. Section 5.2 provides the details of the various prevalent SS comparison methods and their limitations. Section 5.3 includes the proposed methodology, explaining the intricacies of the n-gram similarity evaluation and the modularity-based network community detection method. Section 5.4 contains the experimental setup used to collect expert ATCs eye movement data associated with the issuance of clearance commands. The results and associated discussion are included in Section 5.5 and 5.6 respectively. At last, section 5.6 provides the limitations associated with the proposed method and proposed future studies that can address these limitations.

5.2 Background

Figure 26 shows the broad categorization of different methods that are used to evaluate the similarity of scanpaths (and in turn the visual scanning strategies). As we move from group 1 to 3 (see Figure 26), the granularity of analysis decreases.

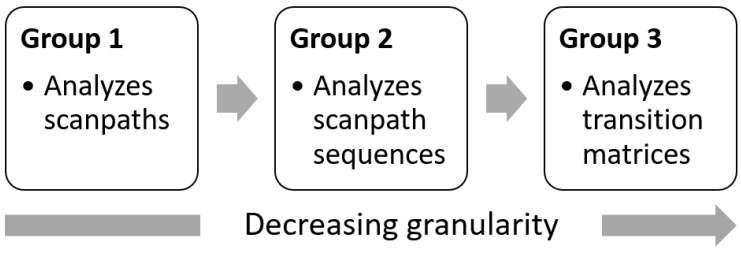


Figure 26: Categorization of methods used to evaluate the similarity of scanpaths.

Fahimi and Bruce (2020) provide a thorough analysis of the pros and cons associated with each such scanpath analysis methods. Moreover, Kang and Landry (2015),

detailed out how these prevalent scanpath analysis methods have some limitations when analyzing scanpath associated with air traffic control operations.

Group 1 methods consist of techniques that directly work on the scanpaths by analyzing the similarity between their shapes, involving the evaluation of the Euclidean distance between them. Mannan distance (Mannan et al., 1996), although, uses Euclidean distance, but doesn't account for the sequential order of the fixations while evaluating the distance between scanpaths. 'Eyeanalysis' approach by Mathôt et al. (2012) implement double mapping maps, which involves mapping every eye fixation in one scanpath to more than one fixation in another scanpath. However, one-to-many eye fixation mapping cases have been noticed (Fahimi & Bruce, 2020). Both the one-to-one and double mapping cases might lead to incorrect results as physical proximity of eye fixation location doesn't entail proximity in the temporal dimension also. Moreover, eye fixations falling on semantically different regions on the display (for a given task under consideration) might appear to be physically proximal if the regions share a common boundary. Therefore, ignoring the sequential order and temporal attribute of the scanpath while evaluating similarity between them is not suitable particularly for dynamic scenarios. On the other hand, MultiMatch (Dewhurst et al., 2012), a vector-based method, needs preprocessing of the raw scanpaths before they are geometrically aligned, followed by their comparison in terms of multiple attributes. Rodrigues and Spalink (2018) showed some simple examples of aggregation schemes used to simplify scanpaths before MultiMatch-based similarity was evaluated between scanpaths. However, Fahimi and Bruce (2020) noted that pre-processing of scanpaths might lead to significant loss of information. For example, scanning strategies associated with localized exploration (having consecutive saccades with small amplitudes)

by experts ATCs might fall on semantically different AOIs (that may be sharing a common border). Therefore, aggregation of these low amplitude saccades, leading to simpler scanpath, may result in loss of crucial task-relevant scanning strategy, that might be crucial from an information gathering standpoint before issuing any clearance command.

Furthermore, the process of shape-based alignment scanpaths is challenging for dynamic scenarios because of two reasons: first, the targets keep changing their position and in turn, it affects scanpath shape as the latter heavily dependent on the distribution of targets over the display area; and second, there is no predefined order for scanning the targets, as a result, scanpath of variable lengths are observed even for a given expert ATC (Kang & Landry, 2015).

Group 3 methods, that analyzes the AOI transition matrices, present a very aggregated level of analysis, where, only the amount of eye fixation transitions between various AOIs pairs are compared. These methods are useful if we are only interested in the broad overview of how the expert ATCs' distribute their visual attention across various AOIs, without being concerned about their visual attention switching behaviour strategy among AOI pairs. Although, there is some merit in using these methods for cases involving conflict resolution between two targets, as the transition order between two targets might be irrelevant from the control point of view (Kang & Landry, 2015).

However, transition matrix comparison methods that utilize pairwise comparison of analogous cell values across different transition matrices, are computationally expensive to apply for scenarios having a large number of AOIs (e.g. airport control tower operations). For example, a scenario having n AOIs will generate an AOI transition matrix of size $n * n$, thereby will require $n * n$ pairwise tests for a given matrix comparison

evaluation (Underwood & Foulsham, 2009). Besides, results from transition matrix comparison, where the whole matrix is treated as a random variable also suffer from sensitivity issues (Kang & Landry, 2010) because, for scenarios having a large number of AOIs, cases with zero transitions between AOIs also become frequent. Moreover, developing sample visual scanning strategies (involving the interrogation of more than two AOIs) from transition matrix is challenging as we no longer preserve the eye fixation order.

Unlike these above-mentioned methods, Group 2 methods that analyze the SSs to compare scanpaths provide a nice sweet spot in terms of the granularity of analysis, preservation of the order visual attention deployment, and enabling the development of sample visual scanning strategies which are semantically similar to each other (useful in developing training materials for novice controllers). Moreover, once the AOIs have been defined using experts' opinion (i.e. expert ATCs, FAA subject matter experts), all eye fixations falling within an AOI boundary (although at different locations) can be treated as semantically equivalent as far an understanding the visual scanning strategy of expert ATCs is involved. Therefore, in the present research, we advocate the use of SS-based similarity evaluation of visual scanning strategies of expert ATCs.

The prevalent SS analysis methods are also known as string-based methods, as the SS can be viewed as a string of characters, where each character represent on AOI. One popular approach named string-edit algorithm (Privitera & Stark, 2000) measures the similarity by evaluating the cost involved in transforming one SS to another by performing operations such as insertion, deletion, and substitution. ScanMatch (Privitera & Stark, 2000) improves upon the limitation of the string-edit algorithm by allowing the inclusion of eye fixation duration and context-specific semantic information in the analysis.

n-gram method preserves the order of the eye fixations when similarities are analyzed among the SSs. Additionally, we can search for longer common subsequences by varying the length of the n-gram. Therefore, n-gram method is more effective when analyzing SSs compared to other text-mining methods such as string-edit algorithm, word embedding, string kernels etc. Note that, being an alignment-free method, the common subsequences identified by n-gram method can occur at different locations within the two SSs. Therefore, the n-gram method facilitates the similarity evaluations among the ATCs' overall visual scanning strategies. The detail working of the n-gram method is explained below.

5.3 Proposed methodology

Figure 27 shows the various steps involved in evaluating the similarity among SSs using the n-gram approach and how these similarity values can be used to detect clusters of homogenous SSs using the graph-based community detection method.

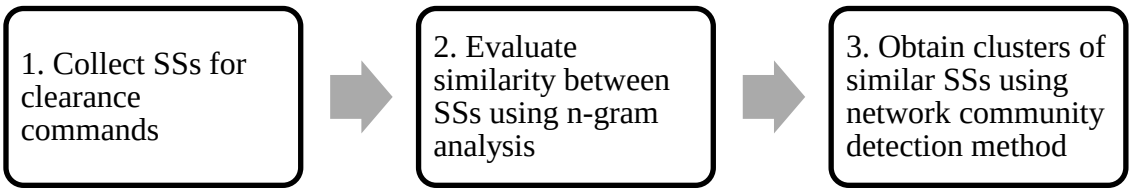


Figure 27: Steps for finding similar visual scanning strategy using n-gram analysis and network community detection method.

Step 1: Collect SSs for clearance commands

The first step involves collecting the SSs of expert ATCs associated with various clearance commands. This is accomplished by performing a high-fidelity simulation experiment, that involves collecting eye movement data of expert ATCs providing clearance commands to control airport control tower traffic. The details of the experimental

process are provided in Section 5.4. The raw eye movement data are mapped to various AOIs that are defined to segregate the ATCs' visual field into discrete regions. The SSs are obtained as the output of this mapping process.

Step 2: Evaluate similarity between SSs using n-gram analysis

Once the SSs are obtained, the next step involves evaluating the n-gram based similarity among them. Before evaluating the similarity values, we need to first understand how to develop the n-gram of a SS.

Let there be L unique AOIs, then the set of AOIs can be represented as $A = \{a_1, a_2, \ldots, a_L\}$. Therefore, the SS (also known as AOI fixation sequence) for the i^{th} ATC can be written as: $S_i = (a_1, \ldots, a_k, \ldots, a_{N_i})$, where N_i is the length of the SS. n-gram of S_i is a subsequence of length n, where $1 \leq n \leq N_i$. Thus, n-gram of length n and starting at position k of the SS S_i can be written as:

$$ng_k(S_i) = (a_k, a_{k+1}, \ldots, a_{k+n-1}) \quad (16)$$

Therefore, the set of all such n-grams, of length n, for S_i can be written as:

$$NG_i = \{ng_k(S_i) | \forall\, k = 1,2, \ldots N_i - n + 1\} \quad (17)$$

Once the set of all n-gram of all the SSs is obtained, the next step involves finding the similarity among the SSs using their n-gram set representation. To this purpose, Maetschke et al. (2010) provided a very intuitive measure, where the similarity is defined as the proportion of n-grams shared by two SSs relative to the length of the smallest SS. Mathematically, the similarity $\emptyset$ between SS S_i and S_j is defined as follows:

$$\phi_{ij} = \frac{|NG_i \cap NG_j|}{\min(|NG_i|, |NG_j|)} \quad (18)$$

However, using the length of the smaller NG set in the denominator (see equation 18) might lead to incorrect interpretation of similarity because this indicates that even when

SSs are of significantly different lengths they will still end up having a similarity value of 1. For example, consider three SSs: (i) S_1: 'ab'; (2) S_2: 'abcd'; and (iii) S_3: 'abcde'. For a 2-gram case, using equation 18, we get $\phi_{13} = \phi_{23} = 1$. Note that, although, S_2 and S_3 are more similar to each other compared to S_1 and S_3, using the approach advocated by Maetschke et al. (2010) leads to the incorrect conclusion that both S_1 and S_2 are equally simialr to S_3. To address this issue and reward similarity between longer SSs, we propose a modified measure where the denominator of equation 18 is changed to the maximum of the cardinality the NG set. Mathematically, it can be written as follows:

$$\phi_{ij} = \frac{|NG_i \cap NG_j|}{\max(|NG_i|,|NG_j|)} \tag{19}$$

On the other hand, the newly proposed relaxed n-gram approach, on the other hand, modifies the numerator of equation 19. For the relaxed n-gram approach, the order of the AOIs in the n-grams are disregarded and thus, any permutation of the constituent elements of n-gram is considered the same while evaluating their intersections across various SSs. For example, consider three n-grams of length 3, i.e. 'abc', 'bca', and 'cba'. In case of the usual definition of n-gram similarity, all three n-gram cases are different, however, as per the relaxed n-gram case, since all three n-grams are a permutation of the constituent elements (i.e. 'a', 'b', and 'c'), thus, they are considered same. This modified definition impacts the value of the numerator of the equation (19), such that, we observe more similar cases between the SSs of expert ATCs.

Step 3: Obtain clusters of similar SSs using network community detection method

After the similarity evaluation between SSs, the next step involves obtaining the cluster of highly similar SSs using the graph-community detection method. This is essential

because currently, we don't have any threshold similarity value that can be used to suggest which SSs are similar and which are not. It depends on the decision-maker on what value to treat as the threshold for the given context. Therefore, to avoid the impact of this subjective decision on the similarity analysis result, we propose the application of an unsupervised clustering method (without requiring the need to explicitly provide the number of clusters required) that uses relative similarity/distance between SSs to divide them into groups of homogenous SSs. To this purpose, we propose using the modularity-based graph-community detection method (Newman & Girvan, 2004; Clauset et al., 2004).

The community detection method involves the following three steps. First, modelling the n-gram similarity values between the SSs as a weighted network, where the nodes represent SSs and the edges between them indicate the n-gram (or relaxed n-gram) similarity between them.

Second, to avoid the chances of having overlapping communities (clusters) and abridge the effect of unrelated SSs on the community detection method, we remove those edges whose weights are less than a chosen threshold, e.g. remove all edges having weights less than 0.3. This weight threshold-based edge removal is analogous to the transformation of the relative distance between objects in the spectral clustering method (clustering method that uses the relative distance/similarity between objects; similar to our case) (Ng et al., 2002 and Shi & Malik, 2000). In spectral clustering approach, the pairwise distance between objects is transformed (using a gaussian kernel method) to an affinity matrix, where objects having stronger ties are further emphasized (leading to a more stronger connection), and on the contrary, weaker ties are diluted. Empirically it has been found that the above-mentioned transformation leads to better segregation of dissimilar objects into

separate clusters. Building upon this idea, in the present research, a weight threshold of 0.3 is chosen. As a result, network edges with weight below 0.3 are removed prior to implementing the network community detection method. This allows for better partitioning of nodes into distinct groups/clusters.

Third, implement modularity-based community detection method on the modified network obtain after removing low weight edges. A brief explanation of the modularity-based network community detection is provided below. For a detailed exposition, please refer to Clauset et al. (2004) and Newman and Girvan (2004).

Modularity is a measure that evaluates the quality of the division process of a network into different communities. Consider a network with n nodes, m edges, and the $\boldsymbol{A} = [A_{ij}]$ is the adjacency matrix. Thus, we get,

$$A_{ij} = \begin{cases} w_{ij}, & i \; and \; j \; connected \\ 0, & othewise \end{cases}$$

The degree of node i can be defined as $k_i = \sum_{j=1}^{n} A_{ij}$. Consider that the network has been divided into communities, and node i belongs to the community c_i, then, the modularity is defined as follows (Brandes et al., 2007; Clauset et al., 2004; Newman & Girvan, 2004; Tantardini et al., 2019):

$$Q = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta(c_i, c_j) \tag{20}$$

Where, δ is a type of indicator function, such that $\delta(u, v) = 1$ if $u = v$, or 0 otherwise. Generally, for connected networks, we get $\frac{-1}{2} \leq Q \leq 1$, and for $Q > 0$ suggest that the observed number of intra-community edges is more than what is expected in case of a random scenario. Besides, $Q > 0.3$ is considered a good indication that a substantial amount of community structure exists in the network under consideration (Clauset et al.,

2004). For the present research, we have used the igraph package (Csardi & Nepusz, 2006) in R to implement the modularity-based community detection method.

5.4 Experiment

NOTE: The experimental design and data collection process are the same as mentioned earlier in Chapter 4. Therefore, to avoid repetition the details are not provided. Although ten expert ATCs have participated in the data collection process, for the present research, the results from only one expert ATC have been shown as a proof of concept.

5.4.1. Data Analysis

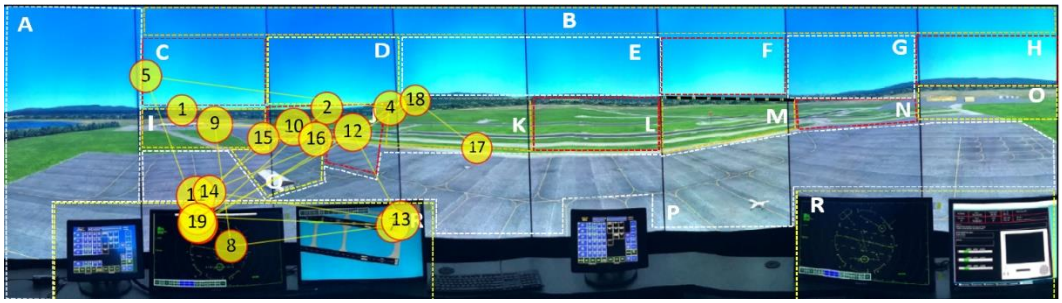
MATLAB 2019 have been used to process raw eye movement data of expert ATCs to produce the VSs visualization and SSs. Subsequently, we've used R software to perform two tasks: (i) develop n-gram and relaxed n-gram similarity method; and (ii) use the similarity results to perform unsupervised clustering using a graph-based community detection method and its subsequent visualization.

The complexity of the n-gram based similarity evaluation method is O($K^2(N - n + 1)^2$) where K is the number of VSs, N is the maximum length of the VSs, and n is the size of the n-gram. The complexity of the modularity-based network detection method is O (K^2) where K is the number of VSs being compared. Thus, the overall complexity of the proposed method is O ($K^2 + K^2(N - n + 1)^2$).

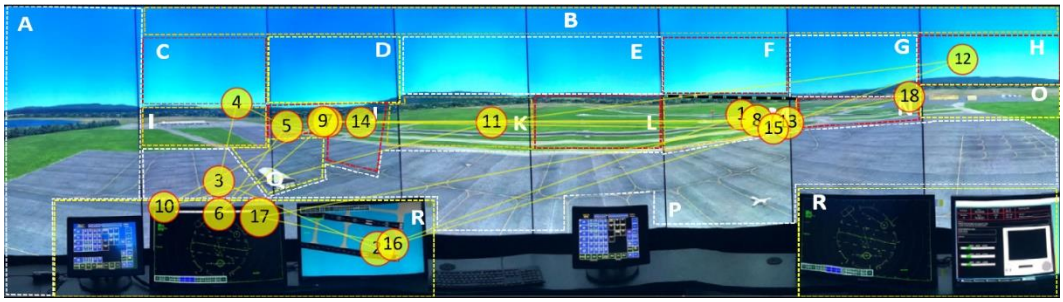
5.5 Results

5.5.1 Visual Scanpath of expert ATCs

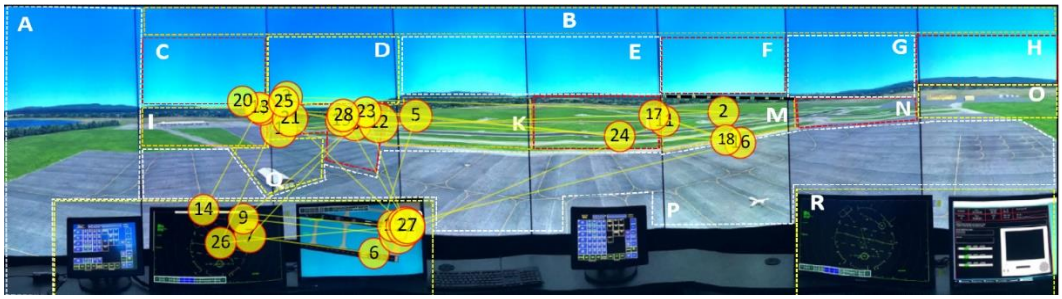
Figure 28 shows examples of different VSs examples associated with three different clearance commands (i.e. CL, CTO, and LW) of an expert ATC. The VSs corresponds to 30 seconds before issuing the clearance command by the expert ATC. In Figure 28, sample 1 and 2 represents two different occurrences of VSs associated with the same clearance command. On simple visual exploration of the VSs plots, both within and between clearance commands, we can infer that there exists a multitude of ways that a singular expert ATC deploys their visual attention before giving clearance commands. Besides, we can see that it is challenging to visually evaluate the similarity among these VSs, and in turn assess the consistency of visual scanning strategy of expert ATCs, even when they are of very for a short duration (i.e. 30 seconds). Therefore, next, we provide the results from the n-gram and relaxed n-gram similarity evaluation process.



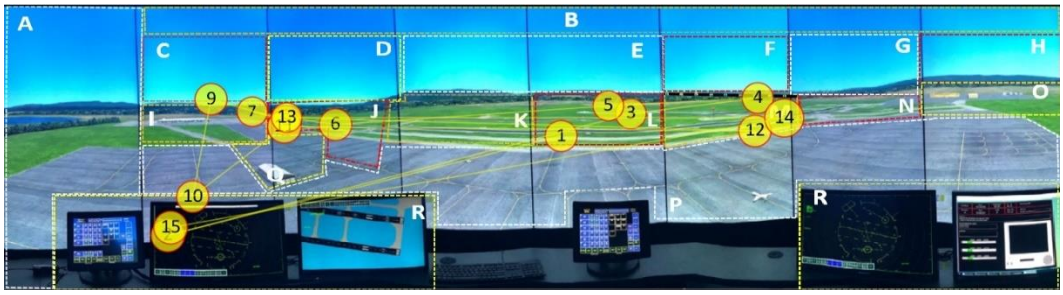
(a) ‘Clear to land’ (CL) - Sample 1



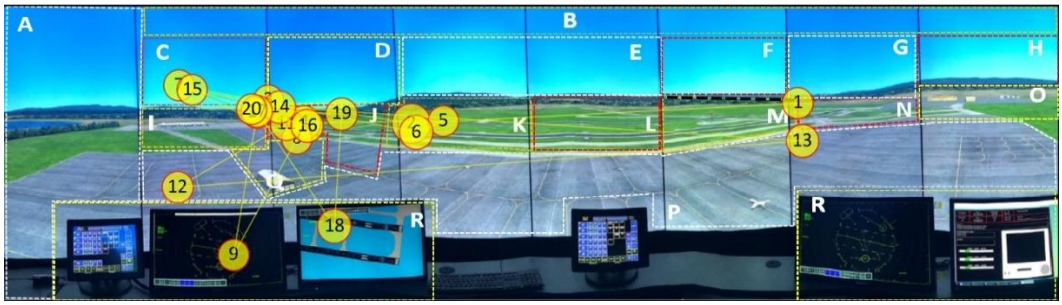
(b) ‘Clear to land’ (CL) - Sample 2



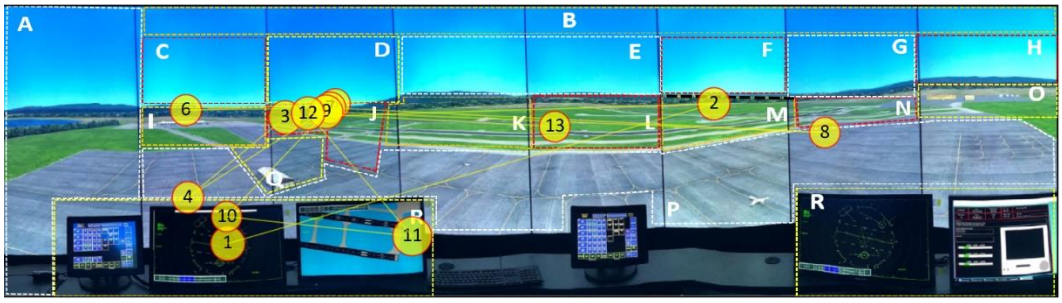
(c) ‘Clear to take off’ (CTO) - Sample 1



(d) ‘Clear to take off’ (CTO) - Sample 2



(e) ‘Line up and wait’ (LW) - Sample 1



(f) ‘Line up and wait’ (LW) - Sample 2

Figure 28: Example VSs associated with three different clearance commands of an expert ATC. The VSs are observed 30 seconds prior to issuing the clearance commands. The characters (in white) show the AOI names used for analysis. The yellow circles and the number on it represent the eye fixations and the eye fixation index respectively.

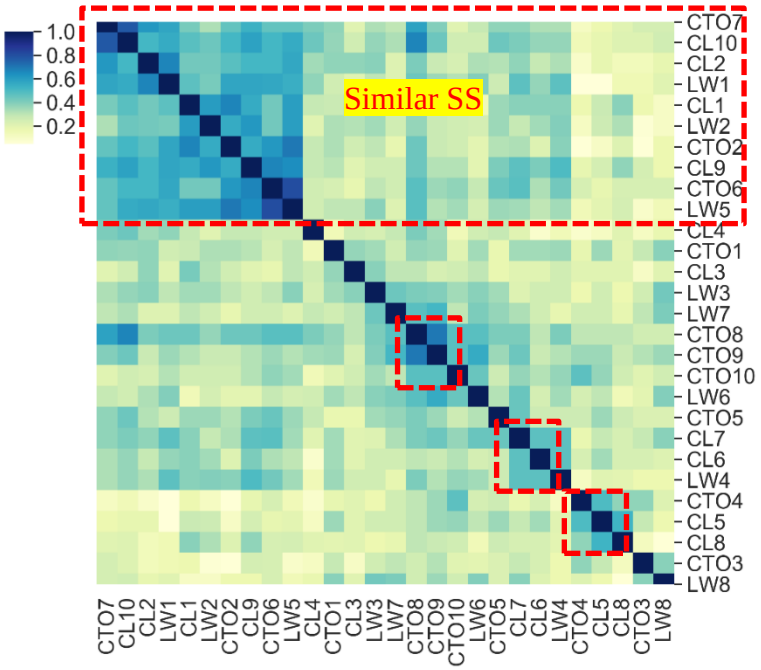
5.5.2 n-gram similarity and heatmaps

The list of raw SSs used for the analysis is provided in the Appendix section, Chapter 5. Figure 29 shows the heatmaps representing the n-gram and relaxed n-gram based similarity values between various instances of SSs associated with three different clearance commands (i.e. CL, CTO, and LW). The row and column names in the heatmaps (Figure 29) indicate both the command name and its sample index. For example, 'CTO7' implies the 7^{th} sample of the VS associated with clearance command 'Clear to take off'.

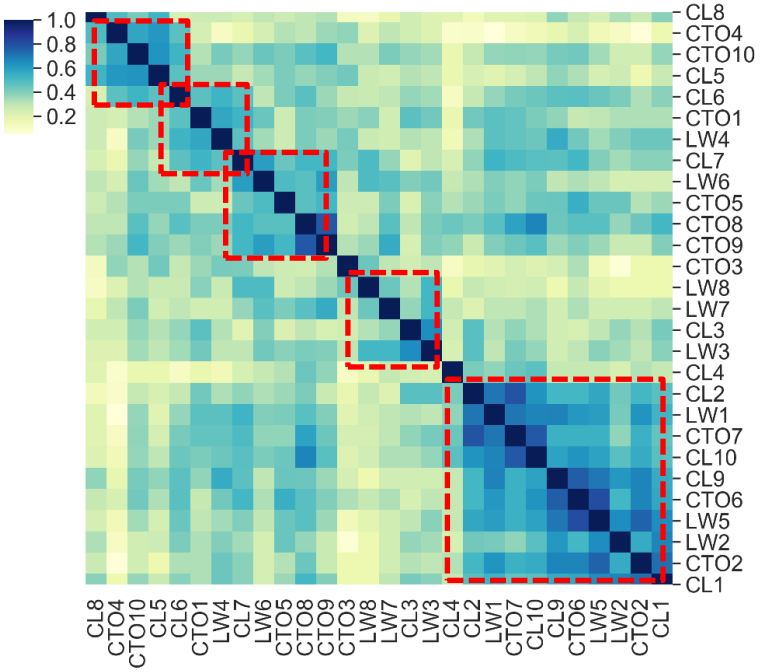
The heatmaps enable easier visual exploration and comprehension of the similarity between SSs. To ensure better comprehension, SSs very similar to each other have been placed nearby on the heatmap representation. The red dotted rectangle in Figure 29 (a), 2-gram similarity heatmap, shows a group of highly similar SSs. Similarly, the various red dotted rectangles in Figure 29 (b), relaxed 2-gram heatmap, indicates the various groups of similar SSs. The diagonal elements in the heatmaps show the self-similarity of SSs, as a result, they are colored blue and indicate a similarity value of 1. Note that, for the same n-gram length, the relaxed n-gram case has more similar SSs compared to the n-gram. This results in a greater number of blue cells in the relaxed n-gram heatmap representations compared to the n-gram. As a result, more clusters which are also bigger (blue colored block structures) can be observed in the relaxed n-gram heatmaps (see Figure 29 (a) and (b)).

On the other hand, if we compare across n-gram of varying lengths, i.e. compare between 2-gram and 3-gram (see Figure 29 (a) and (c)), or relaxed 2-gram and relaxed 3-gram (see Figure 29 (b) and (d)), we can observe that the number of similar SSs reduces, leading to less frequent and smaller size cluster of similar SSs.

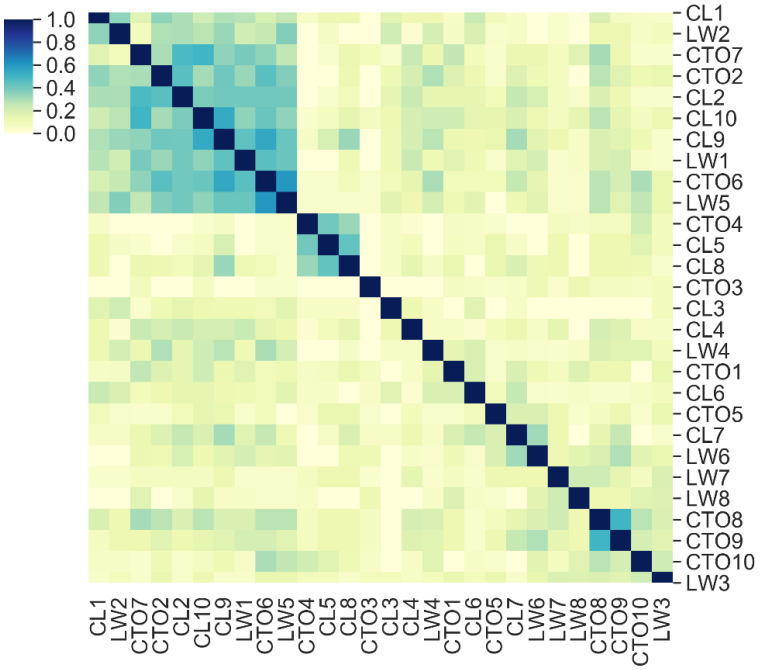
Note that, although, heatmaps allows easy detection of similar SSs and their clusters, however, it depends on the decision-maker on what values they choose as a threshold to define the clusters. Therefore, an objective method, which computes the cluster of similar SSs, without needing the supervision of the decision-maker, would be helpful to validate the developed n-gram based similarity evaluation approach. Thus, next, we provide the results obtained from the network-based community detection method applied to the network developed using both the n-gram and the relaxed n-gram similarity values.



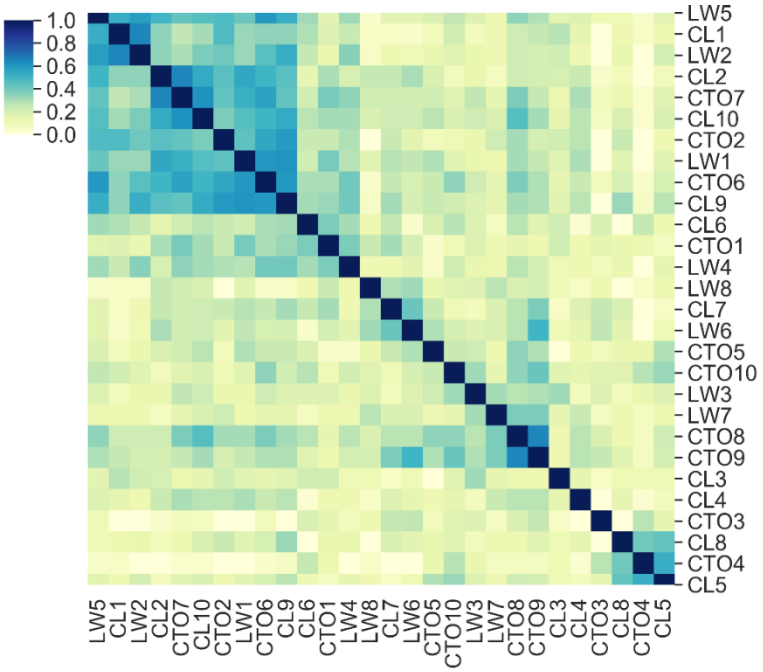
(a) 2-gram similarity heatmap



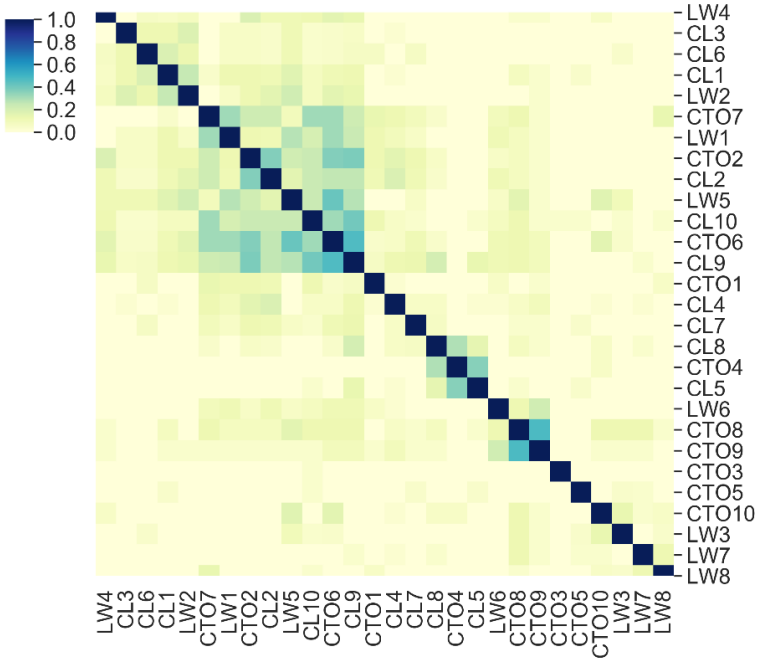
(b) Relaxed 2-gram similarity heatmap



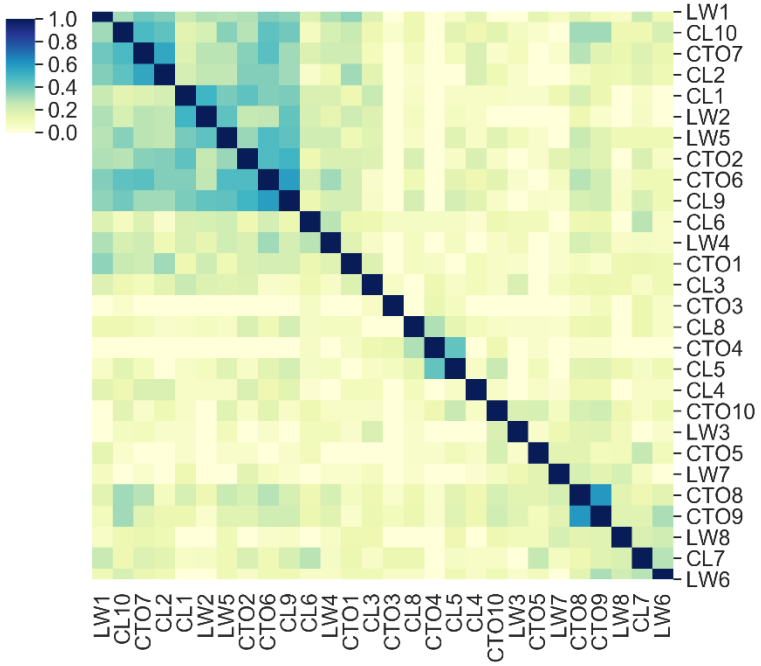
(c) 3-gram similarity heatmap



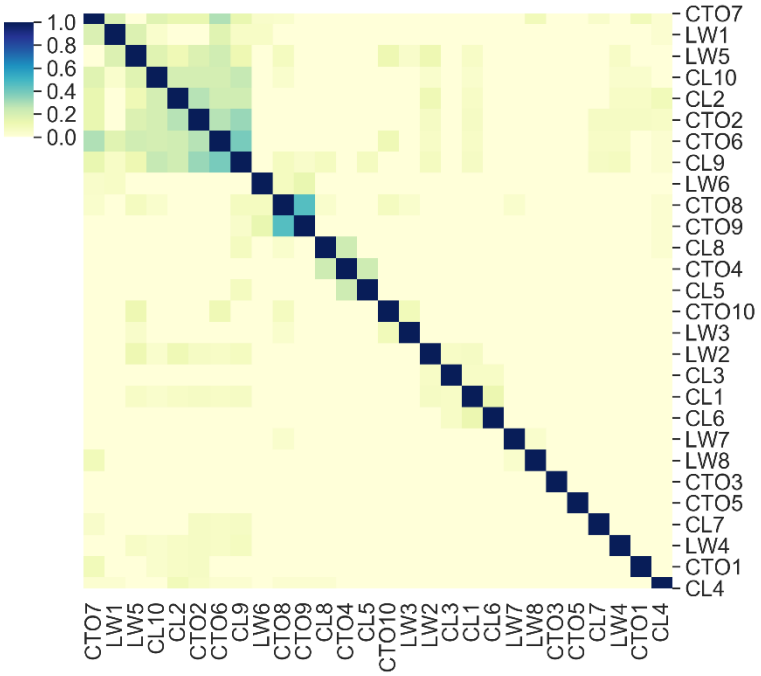
(d) Relaxed 3-gram similarity heatmap



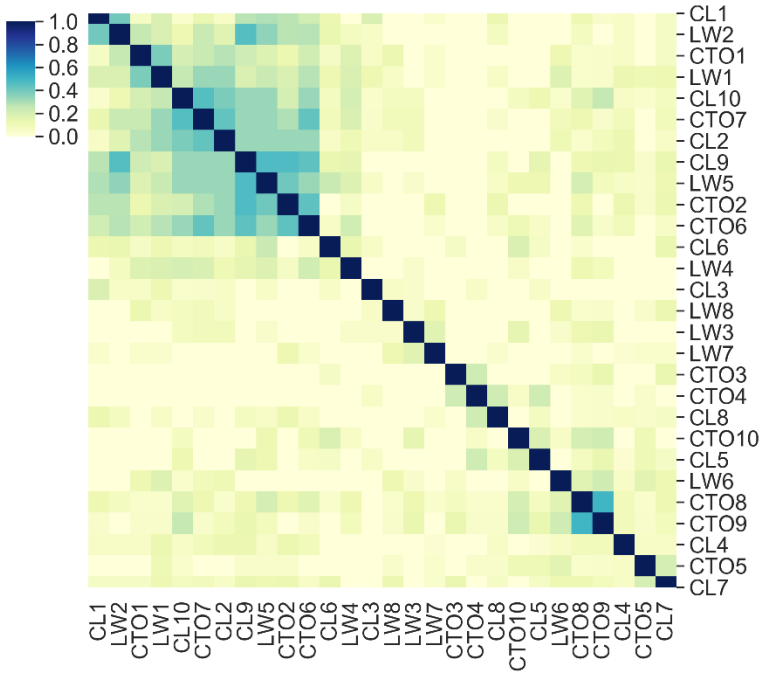
(e) 4-gram similarity heatmap



(f) Relaxed 4-gram similarity heatmap



(g) 5-gram similarity heatmap



(h) Relaxed 5-gram similarity heatmap

Figure 29: Heatmap showing the similarity (both n-gram and relaxed n-gram) between various clearance command related VSs of an expert ATC. The VSs are associated with 30 seconds before issuing the clearance commands.

5.5.3 Network-based visualization of VS similarity

Figure 30 represents the modularity values for various combination of n-gram and relaxed n-gram with both the whole network and threshold-based edge removed case. Note that higher modularity value indicates a better partition of the network into communities. And, the modularity values in Figure 30 show that we obtain a better partition of the network into communities if we remove edges with low weights. Therefore, in the next section, we only present the results for the case where low weight edges have been removed before network community detection was performed.

Moreover, for the case of n-gram and relaxed n-gram, with edges removed, the modularity values are very similar, except for the case of 5-gram, where it experiences a sudden increase in value and reaches approx 0.5. Unlike this, for the relaxed n-gram case

with edges removed, the modularity drops for 5-gram case, however, every other combination sees an improvement in their modularity value with increase in the n-gram length.

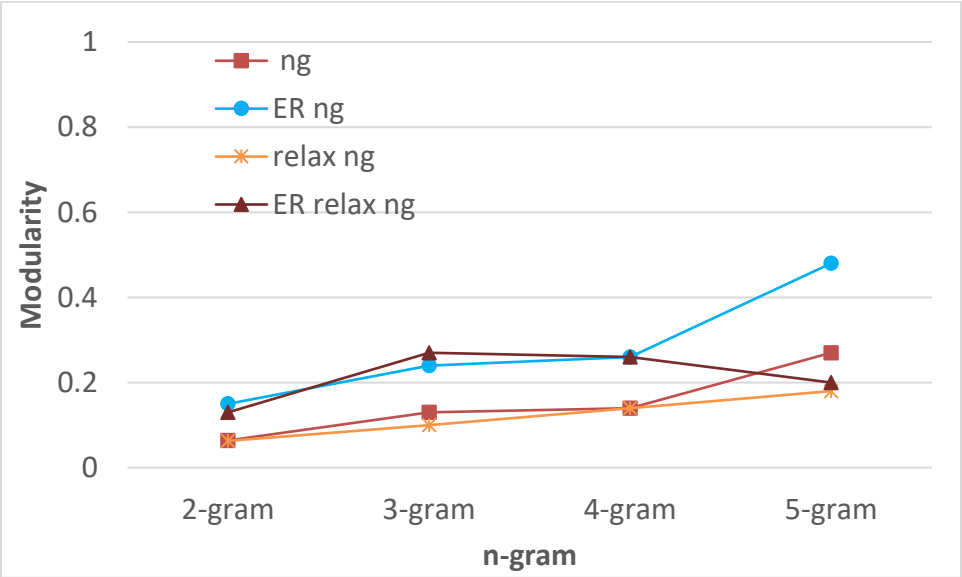


Figure 30: Modularity values for the network-communities obtained for different n-gram lengths. ng: n-gram; relax ng: relaxed n-gram; ER: edge removed.

To aid the understanding of the reader Figure 31 shows an example network-based representation of the similarity between various SSs obtained from applying modularity-based community detection method on the network generated from the n-gram similarity values. In Figure 31, each node represents one instance of a clearance command, e.g. node with label 'CTO9' represents the 9th SS sample observed for the clearance command 'Clear to take off'. Accordingly, CL and LW represents 'Clear to land' and 'Line up and wait' clearance commands respectively. Furthermore, nodes having the same color (e.g. all nodes in pink), except for the nodes colored white, are highly similar to each other. The red dotted circle in Figure 31 shows one such group of highly similar nodes, where each node is colored in pink. Also, similar nodes are joined by an edge between them (shown in grey

color in Figure 31). To aid visual clarity, the edge thickness has been made proportional to the similarity between nodes, i.e. thicker the edge between two nodes, more similar they are to each other.

Moreover, to avoid visual clutter edges having weight less than 0.3 have been removed. Although, any other weight value (e.g. 0.2, or 0.4) can also be chosen as the threshold level for edge removal purpose and depends on the decision-maker.

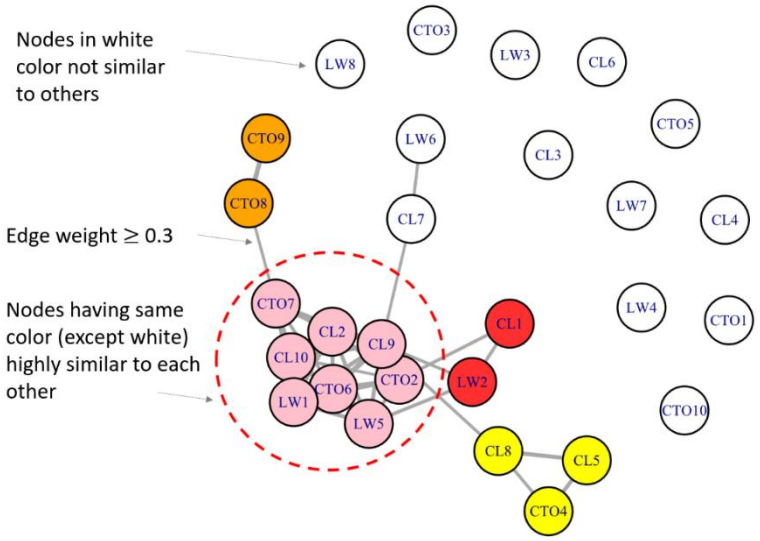
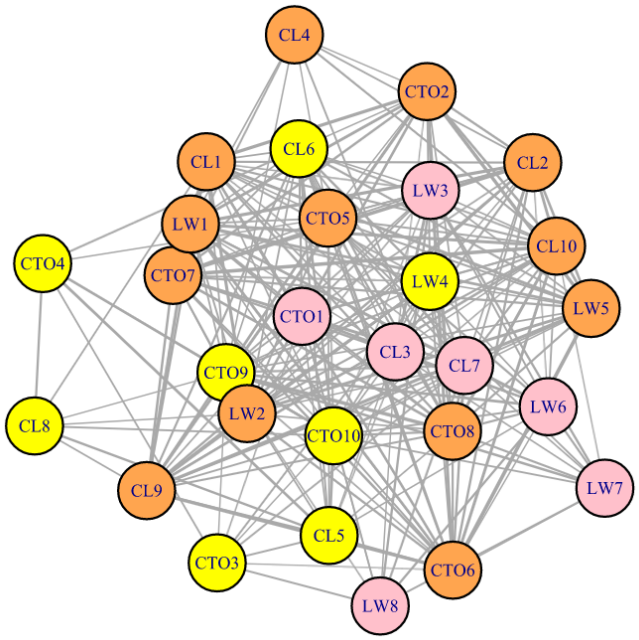


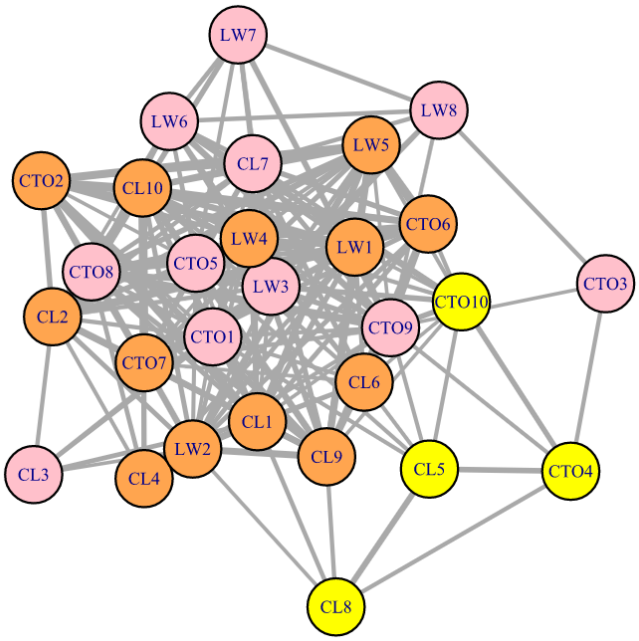
Figure 31: Weighted network-based visualization of expert ATC's visual scanpath similarity.

Figure 32 represents the results of the modularity-based community detection method, showing the similarities between the various SSs for three different clearance commands (i.e. CL, CTO, and LW) for an expert ATC. Figure 32 (a) and (b) indicates that there are three distinct groups of SSs, shown in orange, yellow, and pink color nodes, for both the 2-gram and relaxed 2-gram case. Although, the number of groups is same in both the cases (i.e. general and relaxed 2-gram), the membership of some SSs are different in both the network representations, e.g. CL6 switched from yellow (2-gram) to orange group (relaxed 2-gram).

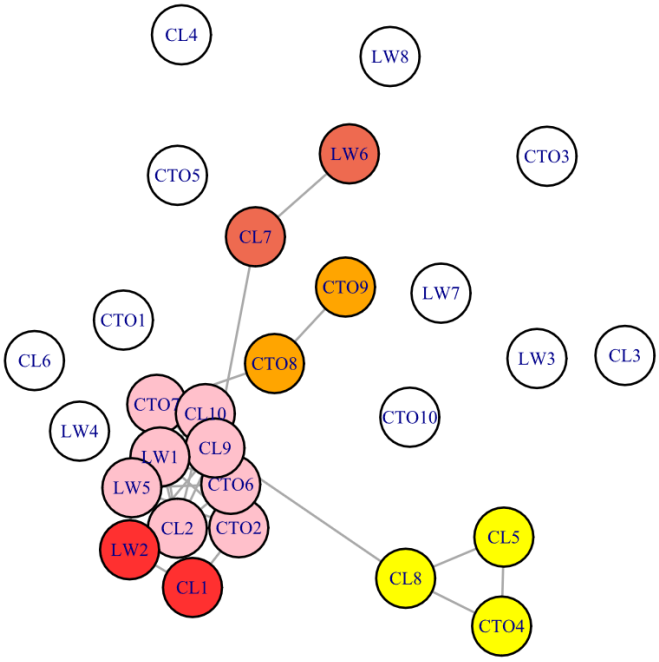
Interestingly, as we move to 3-gram case (see Figure 32 (c) and (d)), the impact of the newly proposed relaxed n-gram approach becomes more prominent because despite having the same number of groups of nodes, the number of constituent nodes is some groups have increased significantly, e.g. the red group. This result is expected for the case of relaxed n-gram method, all permutation of constituent elements of an n-gram is treated to be semantically similar. Another important aspect, which agrees with the results obtained in the heatmap visualization of the n-gram similarity values, that as the length of the n-gram increase, the size of the groups also decreases and there exist many isolated nodes which are not similar to any other nodes. For example, 4-gram has four different groups and 4-gram has only two groups and a large number of island nodes, neither belonging to any group nor connected to any other node (see Figure 32 (e)-(h)). This result is similar to the heatmap representation, where for the case of 5-gram, there is hardly any identifiable block of similar SSs.



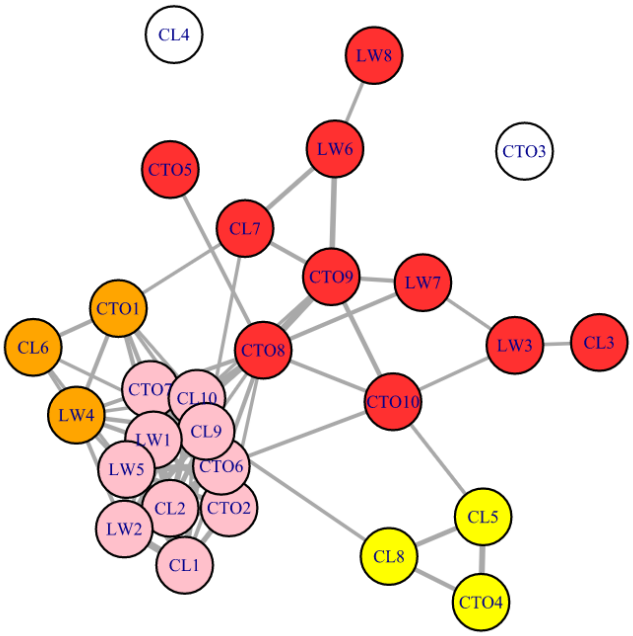
(a) 2-gram similarity



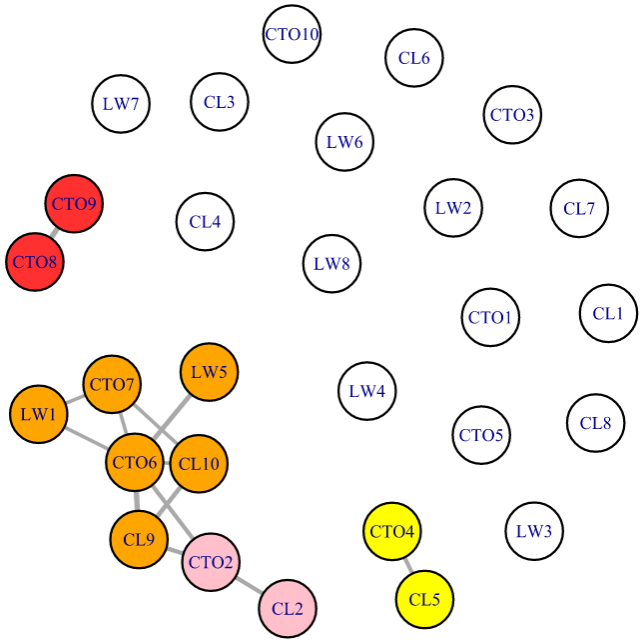
(b) Relaxed 2-gram similarity



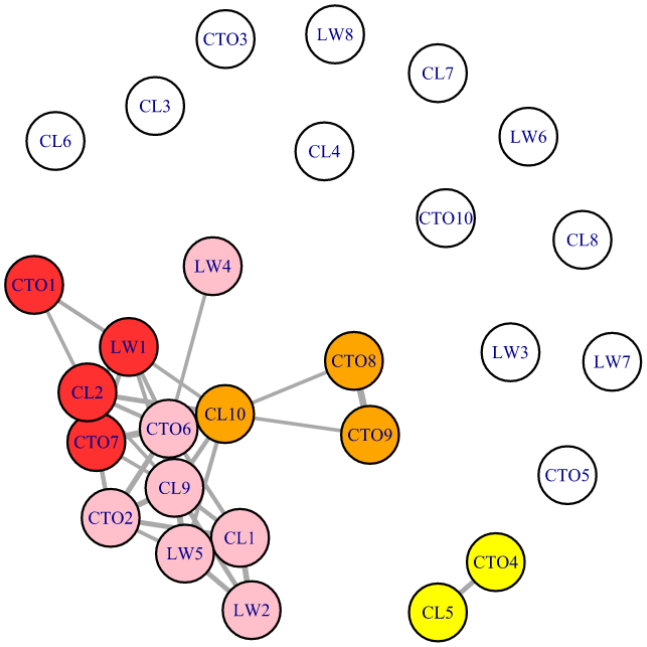
(c) 3-gram similarity



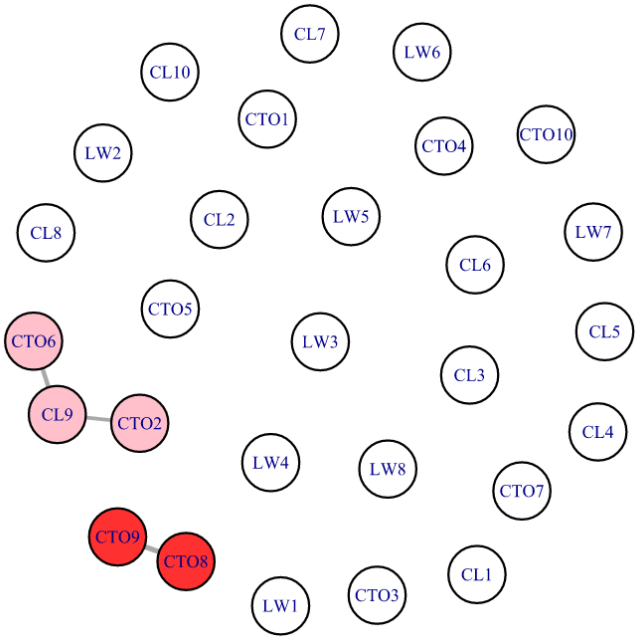
(d) Relaxed 3-gram similarity



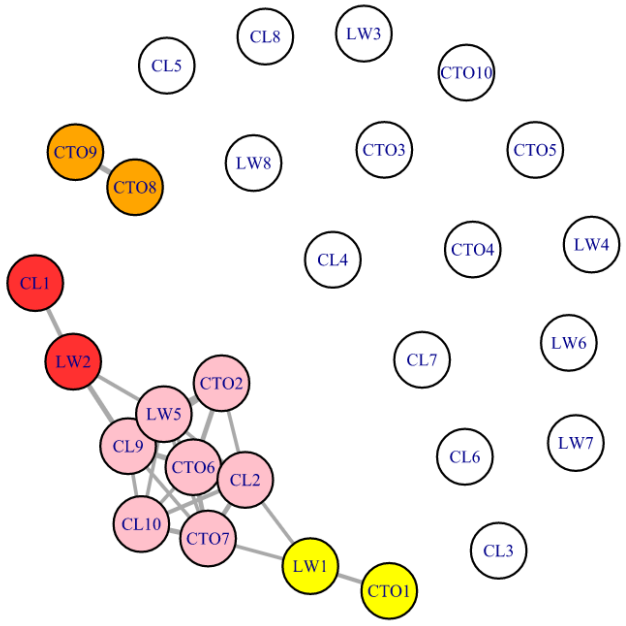
(e) 4-gram similarity



(f) Relaxed 4-gram similarity



(g) 5-gram similarity



(h) Relaxed 5-gram similarity

Figure 32: Network-based representation of similarity between various clearance command related VSs of an expert ATC. Both n-gram and relaxed n-gram based similarity values for various n-gram levels are shown. The VSs are associated with 30 seconds before issuing the clearance commands.

5.6 Discussion

In this research, we developed a framework to evaluate the similarity among SSs obtained in dynamic target tracking and control tasks, e.g. air traffic control tasks. The framework consists of initially evaluating the similarity between SSs using an n-gram based approach. To this purpose, two new n-gram based similarity evaluation methods are proposed: (i) involves computing the number of common n-gram subsequences between the SSs as a proportion of the length of largest SS; and (ii) consists of modifying the traditional n-gram definition with its relaxed version where the order of the constituent elements of the n-gram is ignored, and any permutation of these elements are considered same. Afterwards, the n-gram based similarity values are used to implement a modularity-based network community detection method, which enables identification of highly similar

SSs without explicitly providing the number of groups/clusters being provided by the decision-maker.

The example VSs for the various clearance commands demonstrates that it's indeed challenging to visually explore and identify the commonality among the SSs associated with the clearance commands, even when they are of short duration (i.e. 30 seconds). As a consequence, this justifies the utility of the proposed framework which provides an objective way to discover groups of similar SSs.

The results of the heatmaps, displaying the n-gram (and the relaxed n-gram) similarity values between SSs, show that we were able to identify several cases of very similar SSs. As expected, for small n-gram lengths (2 and 3), there are several cases of high similarity between SSs, and we could locate multiple blocks of homogenous Sss across the diagonal of the heatmap representation. With the increase in the n-gram length (4 and 5), the similarity between SSs decreases and very few blocks of homogenous SSs are visible. Besides this, for a given n-gram length, the relaxed definition led to the higher similarity among SSs as the order of the n-gram elements were ignored. Although, we were able to identify a few instances of similar SSs which were placed along the diagonal of the heatmaps, however, for it's quite cumbersome to locate off-diagonal similar SSs as the number fo SSs compared increases. Moreover, the absence of any context-specific benchmark value for similarity, makes it difficult to identify groups of homogenous SSs. This gap is addressed by the network community detection method, that uses the n-gram similarity scores to determine groups of highly similar SSs using a modularity-based method.

The network community detection and its subsequent visualization allow for easy identification of highly similar SSs, as the node color attribute enable enhanced visual segregation of similar SSs. The community detection results show that as the n-gram length increases, the cluster size of homogenous SSs also shrinks, thereby indicating that the similarity reduces with increasing n-gram length. This is expected, as with longer subsequences, it is less probable that the expert ATC will exhibit similar scanning behavior for different instances of same clearance command and far less for different clearance commands. Interestingly, for a given n-gram length, the size of the SS clusters/communities is larger for the relaxed n-gram case. This is consistent with the heatmap results, as the relaxed n-gram allows for all permutations of n-gram constituent elements to be considered equivalent. Moreover, the network visualization also highlights those particular SSs which are very dissimilar from other SSs, by showing them as isolated nodes. And as expected, the number of isolated nodes increases with the length of the n-gram.

A closer analysis of the SSs clusters will help us understand about the consistency of the visual scanning strategy of expert ATCs while issuing clearance commands. In detail, different instances of the SSs associated with the same clearance command, belonging to the same cluster will indicate that expert ATCs have an inbuilt consistent strategy to deploy their visual attention across the display while they provide similar clearance commands. This might imply a top-down control on the visual attention distribution strategy of expert ATCs. On the other hand, if SSs associated with the same clearance command belongs to difference cluster, and found to be similar to other SSs belonging to different clearance commands, it might indicate that the bottom-up influence

is more substantial, as with time the scenario context evolves even while the same clearance command needs to be issued (e.g. different traffic volume and characteristics near the runway). Although, further in-depth investigation of the scenario attributes for those particular time intervals is warranted before can identify the reason behind the non-homogenous nature of the SSs.

5.7 Limitation and Future Work

One limitation of the present study in the small sample size. Currently, we have only provided an analysis of one expert ATC's case. Moreover, the SSs analyzed were of short duration, i.e. 30 seconds, and they were observed just before the issuance of the various clearance commands. Besides, the present framework evaluates the similarity between SSs by only using n-gram based methods. It's to be noted that, n-gram based methods don't consider the eye fixation duration into account while evaluating similarity..

One limitation of the proposed method is that the similarities were analyzed using all the clearance commands combined for a single ATC. Future research involves implementing the n-gram method based on each clearance command and/or each expertise level (i.e. novices and experts).

The proposed method provides various colors for visualization and those who are color-blinded might not be able to benefit from the proposed approach. To consider those who are color-blinded one method can be using different shape rather than colors. In addition, instead of using multiple colors a single color can be used by implementing lightness contrast.

Therefore, as future work, we should include SSs for more number of expert ATCs. Furthermore, we should also consider analyzing SSs of longer duration (e.g. 60 seconds), and both before and after the clearance commands are issued.

Besides, we should explore similarity evaluation methods that also accommodate the eye fixation duration. Further, in-depth analysis is required, focusing on the underlying scenario contexts responsible for the observed similarity across SSs, and the unearth particular aspect of the SSs (exact subsequences shared across SSs) which led to the similarity among their various manifestations.

Chapter 6: Conclusion

In conclusion, the current research addresses three pertinent questions that will enhance our understanding of the visual scanning strategies of observers involved in a dynamic target tracking task.

The first topic was related to the effective visualizing of the eye movement characterisitics for a dynamic target tracking task. To address the question, a dynamic network-based framework was developed that allows identification of important targets whcih were visually attended and how their importance evolved over time. Moreover, four different visualization methods were introduced: Dot plot, bar plot, norm-cube plot, and order plot. The plots further helped in identifying the important targets that were visually attended. The four types of plots had unique strengths and weaknesses. The proposed methods were applied to an enroute air traffic control tasks to better analyze the visual scanning strategies of expert air traffic controllers.

The second topic was on the visualization of the complex visual scanpaths for a dynamic target tracking task.A novel spatial-temporal clustering method was introduced that allows the simplication of the complex visual scanpaths through aggregating the eye fixations that are both spatially and temporally close to one another. The proposed method was applied in air traffic control operations. The simplified visual scanpath results showed that the overall geometric shape of the visual scanpaths can be retained even if the amount of complexity (constituent eye fixation count) of the visual scanpaths were reduced up to 50%. The reduction of the visual scanpath complexity shows promise on representing clearer visual search strategies of the expert air traffic controllers that might be used to better traing the novice air traffic controllers.

The third research topic was on finding similar visual scanning scanpaths for a dynamic target tracking task. The proposed method computes the *n*-gram based similarities among the visual scanpath sequences, and then implements the modularity-based network community detetion method to cluster the scanpath sequences. The proposed method does not require the researcher to pre-define any parameter values prior to clustering the visual scanpath sequences. The proposed method was applied in air traffic control operations and showed promise on formulating representative visual scanpaths within each cluster.

Chapter 7: Limitations and Future Work

A limitation of the dynamic network-based framework in the absence of a systematic approach to divide the whole experimental time into smaller intervals that can best show the evolution of the eye movement characterisitics. As a future work, the systematic approach needs to be developed based on different experiment times and task types.

Regarding the plots to visualize the eye movements their outputs are currently static, meaning that the users are not able to interact with the plots. Future work involves finding interactive methods to reduce visual clutter, filtering data, and being able to simultaneously show multiple eye movement characteristics and/or outputs obtained from multiple participants to enable more efficient comparisons.

The limitation of the scanpath simplification methodology is that till now only a small sample of data was used to evaluate the proposed method. A larger sample will be used in the future. Furthermore, long duration scanpath sequences (e.g. 60 s or 120 s) will be used to investigate how the the length of scanpath sequnces impact the overall clustering outcomes.

The proposed methods were effective in visualizing the eye movement characteristics and comparing the similarities among the visual scanpath sequences. Future research involves investigating whether the proposed methods might be used to increase the novice performance. In addition, investigations are needed to possibly use the outputs from the proposed methods to evaluate training performance.

References

Adacel Systems, I. (2020). *MaxSim air traffic control simulation*. https://www.adacel.com/maxsim-air-traffic-control-simulation-training

Archambault, D., & Purchase, H. C. (2013). Mental map preservation helps user orientation in dynamic graphs. In W. Didimo, & M. Patrignani (eds), *Graph Drawing. GD 2012, Lecture Notes in Computer Science* (pp. 475–486). Heidelberg, Berlin: Springer, Berlin, Heidelberg.

Beck, F., Burch, M., & Diehl, S. (2013). Matching Application Requirements with Dynamic Graph Visualization Profiles. In *Information Visualisation (IV), 2013 17th International Conference* (pp. 11–18). London, UK: IEEE.

Beck, F., Burch, M., Diehl, S., & Weiskopf, D. (2014). The state of the art in visualizing dynamic graphs. In *Eurographics Conference on Visualization (EuroVis)*(pp. 1–20). Swansea, UK: The Eurographics Association.

Bemdt, D. J., & James, C. (1994). Using Dynamic Time Warping to Find Patterns in Time Series. *ACM SIGKDD Conference on Knowledge Discovery and Data Mining Workshop*, *10*(16), 359–370.

Birant, D., & Kut, A. (2007). ST-DBSCAN: An algorithm for clustering spatial-temporal data. *Data and Knowledge Engineering*, *60*(1), 208–221. https://doi.org/10.1016/j.datak.2006.01.013

Blascheck, T., Kurzhals, K., Raschke, M., Burch, M., Weiskopf, D., & Ertl, T. (2014). State-of-the-Art of Visualization for Eye Tracking Data. In *Eurographics Conference on Visualization (EuroVis)* (pp. 1–20). Swansea, UK: The Eurographics Association.

Blascheck, T., Raschke, M., & Ertl, T. (2013). Circular heat map transition diagram. In *Proceedings of the 2013 Conference on Eye Tracking South Africa - ETSA '13* (pp. 58–61). Cape Town, SA: ACM.

Borgatti, S. P. (2005). Centrality and network flow. *Social Networks*, *27*(1), 55–71.

Brandes, U., Delling, D., Gaertler, M., Robert, G., Hoefer, M., Nikoloski, Z., & Wagner, D. (2007). On Modularity Clustering. *IEEE Transactions on Knowledge and Data Engineering*, *20*(2), 172–188.

Burch, M., Beck, F., Raschke, M., Blascheck, T., & Weiskopf, D. (2014). A dynamic graph visualization perspective on eye movement data. In *Proceedings of the Symposium on Eye Tracking Research and Applications* (pp. 151–158). Safety Harbor, FL: ACM.

Burch, M., Kull, A., & Weiskopf, D. (2013). AOI Rivers for Visualizing Dynamic Eye Gaze Frequencies. In *Proceedings of EuroVis '13 Proceedings of the 15th Eurographics Conference on Visualization* (pp. 151–158). Leipzig, Germany: Eurographs Association & John Wiley & Sons, Ltd.

Chhaya, B., Jafer, S., Coyne, W. B., Thigpen, N. C., & Durak, U. (2018). Enhancing scenario-centric air traffic control training. *AIAA Modeling and Simulation Technologies Conference, 2018, 209959*.

Clauset, A., Newman, M. E. J., & Moore, C. (2004). Finding community structure in very large networks. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, *70*(6), 66111. https://doi.org/10.1103/PhysRevE.70.066111

Csardi, G., & Nepusz, T. (2006). The igraph software package for complex network research. *InterJournal, Complex Systems*, *2006*, 1695. http://igraph.org

Dewhurst, R., Nyström, M., Jarodzka, H., Foulsham, T., Johansson, R., & Holmqvist, K. (2012). It depends on how you look at it : Scanpath comparison in multiple dimensions with MultiMatch, a vector-based approach. *Behavior Research Methods*, *44*, 1079–1100. https://doi.org/10.3758/s13428-012-0212-

Eyetracking.com. (2017). "Powerful eye tracking software developed for researchers." Retrieved March 1, 2017, from http://www.eyetracking.com/Software/EyeWorks

Fahimi, R., & Bruce, N. D. B. (2020). On metrics for measuring scanpath similarity. *Behavior Research Methods*, 1–20.

Federal Aviation Administration. (2018). *Forecasts of IFR aircraft handled by FAA air route traffic control centers FY 2017-2040*. Retrieved from https://www.faa.gov/data_research/aviation/aerospace_forecasts/.

Freeman, L. C. (1978). Centrality in social networks conceptual clarification. *Social Networks*, *1*(3), 215–239.

Ghani, S., Elmqvist, N., & Yi, J. S. (2012). Perception of Animated Node-Link Diagrams for Dynamic Graphs. *Computer Graphics Forum*, *31*(3pt3), 1205–1214.

Giorgino, T. (2009). Computing and visualizing dynamic time warping alignments in R: The dtw package. *Journal of Statistical Software*, *31*(7), 1–24. https://doi.org/10.18637/jss.v031.i07

Goldberg, J. H., & Helfman, J. I. (2010). Visual scanpath representation. In *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications - ETRA '10* (pp. 203-210). Austin, TX: ACM.

Goldberg, J. H., & Kotval, X. (1999). Computer interface evaluation using eye movements: methods and constructs. *International Journal of Industrial Ergonomics*, *24*(6), 631–645.

Hahsler M, Piekenbrock M, D. D. (2019). dbscan: Fast Density-Based Clustering with R. *Journal of Statistical Software*, *91*(1), 1–30. https://doi.org/10.18637/jss.v091.i01

Hampton, M. E. (2016). *Key issues facing FAA's air traffic controller workforce*. Retrieved from https://www.oig.dot.gov/library-item/33382.

Holmqvist, K., Holsanova, J., Barthelson, M., & Lundqvist, D. (2003). Reading or Scanning? A Study of Newspaper and Net Paper Reading. In J. Hyönä, R. Radach, & H. Deubel (Eds.), *The Mind's Eye: Cognitive and Applied Aspects of Eye Movement Research* (pp. 658–670). North Holland: Elsevier B.V.

Jeong, Y. S., Jeong, M. K., & Omitaomu, O. A. (2011). Weighted dynamic time warping for time series classification. *Pattern Recognition*, *44*(9), 2231–2240. https://doi.org/10.1016/j.patcog.2010.09.022

Kang, Z., & Bass, E. J. (2014). Supporting the eye tracking analysis of multiple moving targets: Design concept and algorithm. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC '14)* (pp. 3184–3189). San Diego, CA: IEEE.

Kang, Z., & Landry, S. J. (2015). An Eye Movement Analysis Algorithm for a Multielement Target Tracking Task : Hierarchical Clustering. *IEEE Transactions on Human-Machine Systems*, *45*(1), 13–24.

Kang, Z., & Landry, S. J. (2014). Using Scanpaths as a Learning Method for a Conflict Detection Task of Multiple Target Tracking. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, *56*(6), 1150–1162

Kang, Z., & Landry, S. J. (2010). Capturing and Analyzing Visual Groupings of Multiple Moving Targets in an Aircraft Conflict Detection Task Using Eye Movements. *Proceedings 0f 54th Annual Meeting of Human Factors Ergonomic Society*, 1906–1910.

Kang, Z., Mandal, S., Crutchfield, J., Millan, A., & Mcclung, S. N. (2016). Designs and Algorithms to Map Eye Tracking Data with Dynamic Multielement Moving Objects. *Computational Intelligence and Neuroscience*, *2016*(1), 1–18.

Kurzhals, K., Fisher, B., Burch, M., & Weiskopf, D. (2015). Eye tracking evaluation of visual analytics. *Information Visualization*, *15*(4), 340–358.

Kurzhals, K., Heimerl, F., & Weiskopf, D. (2014). ISeeCube: Visual Analysis of Gaze Data for Video. In *Proceedings of the symposium on Eye tracking research & applications - ETRA '14* (pp. 43–50). Safety Harbor, FL: ACM.

Kurzhals, K., & Weiskopf, D. (2013). Space-time visual analytics of eye-tracking data for dynamic stimuli. *IEEE Transactions on Visualization and Computer Graphics*, *19*(12), 2129–2138.

Le Meur, O., & Liu, Z. (2015). Saccadic model of eye movements for free-viewing condition. *Vision Research*, *116*, 152–164. https://doi.org/10.1016/j.visres.2014.12.026

Li, A., & Chen, Z. (2018). Representative Scanpath Identification for Group Viewing Pattern Analysis. *Journal of Eye Movement Research*, *11*(6), 1–17.

Li, A., Zhang, Y., & Chen, Z. (2017). Scanpath Mining Of Eye Movement Trajectories For Visual Attention Analysis. *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME), July*, 535–540.

Li, W. C., Kearney, P., Braithwaite, G., & Lin, J. J. H. (2018). How much is too much on monitoring tasks? Visual scan patterns of single air traffic controller performing multiple remote tower operations. *International Journal of Industrial Ergonomics*, *67*(May), 135–144. https://doi.org/10.1016/j.ergon.2018.05.005

Liu, Y. T., Zhang, Y. A., & Zeng, M. (2019). Adaptive Global Time Sequence Averaging Method Using Dynamic Time Warping. *IEEE Transactions on Signal Processing*, *67*(8), 2129–2142. https://doi.org/10.1109/TSP.2019.2897958

Maetschke, S. R., Kassahn, K. S., Dunn, J. A., Han, S., Curley, E. Z., Stacey, K. J., & Ragan, M. A. (2010). A visual framework for sequence analysis using n-grams and spectral rearrangement. *Bioinformatics*, *26*(6), 737–744. https://doi.org/10.1093/bioinformatics/btq042

Mandal, S., & Kang, Z. (2018). Using Eye Movement Data Visualization to Enhance Training of Air Traffic Controllers : A Dynamic Network Approach. *Journal of Eye Movement Research*, *11*(4).

Mandal, S., Kang, Z., & Millan, A. (2016). Data visualization of complex eye movements using directed weighted networks: a case study on a multi-element target tracking task. In *Proceedings of the Human Factors and Ergonomics Society 60th Annual Meeting* (pp. 106–110). Washington, DC: Human Factors and Ergonomics Society.

Mannan, S. K., Ruddock, K. H., & Wooding, D. S. (1996). The relationship between the locations of spatial features and those of fixations made during visual examination of briefly presented images. *Spatial Vision*, *10*(3), 165–188.

Manske, P. G., & Schier, S. L. (2015). Visual Scanning in an Air Traffic Control Tower – A Simulation Study. *6th International Conference on Applied Human Factors and Ergonomics (AHFE 2015) and the Affiliated Conferences, AHFE 2015*, *3*, 3274–3279. https://doi.org/10.1016/j.promfg.2015.07.397

Martin Ester, Hans-Peter Kriegel, Jiirg Sander, X. X. (1996). A Density-Based Algorithm

for Discovering Clusters in Large Spatial Databases with Noise. *Proceedings of Second International Conference on Knowledge Discovery and Data Mining*, *2*, 226–231. https://doi.org/10.1016/B978-044452701-1.00067-3

Mathôt, S., Cristino, F., Gilchrist, I. D., & Theeuwes, J. (2012). A simple way to estimate similarity between pairs of eye movement sequences. *Jemr*, *5*(1), 1–15. https://doi.org/10.16910/jemr.5.1.4

Mathworks. (2019). *MATLAB R2019b*.

Mcclung, S. N., & Kang, Z. (2016). Characterization of Visual Scanning Patterns in Air Traffic Control. *Computational Intelligence and Neuroscience, 2016* (1), 1-17.

Muthumanickam, P., Nordman, A., Meyer, L., Boonsong, S., Lundberg, J., & Cooper, M. (2019). Analysis of long duration eye-tracking experiments in a remote tower environment. *13th USA/Europe Air Traffic Management Research and Development Seminar 2019*.

Newman, M. E. J. (2010). *Networks: An introduction.* New York, NY: Oxford University Press Inc.

Newman, M. E. J., & Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, *69*(22), 1–15. https://doi.org/10.1103/PhysRevE.69.026113

Newman, M. E. J. (2004). Analysis of weighted networks. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, *70*(5), 1–9.

Noton, D., & Stark, L. (1971). Scanpaths in Eye Movements during Pattern Perception. *Science, 171*(3968), 308–311.

Ng, A. Y., Jordan, M. I., & Weiss, Y. (2002). On Spectral Clustering: Analysis and an algorithm Andrew. *Advances in Neural Information Processing Systems*, 849–856. https://papers.nips.cc/paper/2092-on-spectral-clustering-analysis-and-an-algorithm.pdf%0Ahhttp://www.treasury.gov.au/ConsultationsandReviews/Consultations/2016/CFFR-Affordable-Housing-Working-Group

Opsahl, T., Agneessens, F., & Skvoretz, J. (2010). Node centrality in weighted networks: Generalizing degree and shortest paths. *Social Networks*, *32*(3), 245–251.

Palma Fraga, R., Kang, Z., & Mandal, S. (2018). Characterization of air traffic controller's visual search patterns and control strategies. *ISER 132nd International Conference, June*.

Papenmeier, F., & Huff, M. (2010). DynAOI: A tool for matching eye-movement data with dynamic areas of interest in animations and movies. *Behavior Research Methods, 42*(1), 179–187.

Privitera, C. M., & Stark, L. W. (2000). *Algorithms for Defining Visual Regions-of-Interest : Comparison with Eye Fixations*. *22*(9), 970–982.

Purchase, H. C., Hoggan, E., & Görg, C. (2006). How Important Is the "Mental Map"? – An Empirical Investigation of a Dynamic Graph Layout Algorithm. In M. Kaufmann & D. Wagner (Eds.), *International Symposium on Graph Drawing GD 2006: Graph Drawing* (pp. 184–195). Berlin: Springer, Berlin, Heidelberg.

Raschke, M., Chen, X., & Ertl, T. (2012). Parallel scan-path visualization. In *Proceedings of the Symposium on Eye Tracking Research and Applications - ETRA '12* (pp. 165–168). Santa Barbara, CA: ACM.

Rodrigues, N., & Spalink, J. (2018). Multiscale Scanpath Visualization and Filtering. *Proceedings of the 3rd Workshop on Eye Tracking and Visualization*.

Rudi, D., Kiefer, P., & Raubal, M. (2018). Visualizing Pilot Eye Movements for Flight Instructors. In *Proceedings of the 3rd Workshop on Eye Tracking and Visualization - ETVIZ'18*. Warsaw, Poland: ACM.

Saraiya, P., Lee, P., & North, C. (2005). Visualization of graphs with associated timeseries data. In *Proceedings of the 2005 IEEE Symposium on Information Visualization, INFO VIS* (pp. 225–232). Minneapolis, MN: IEEE.

Shi, J., & Malik, J. (2000). Normalized Cuts and Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Inteligence*, *22*(8), 888–905. https://doi.org/10.1093/biomet/ass019

Svensson, Å. (2015). *Air traffic controllers' work-pattern during air traffic control tower simulations : A eye-tracking study of air traffic controllers' eye-movements during arrivals* [Linköping University Examiner:]. https://doi.org/10.1108/02686900410549457

Tantardini, M., Ieva, F., Tajoli, L., & Piccardi, C. (2019). *Comparing methods for comparing networks*. 1–19. https://doi.org/10.1038/s41598-019-53708-y

Team, R. C. (2019). *R: A language and environment for statistical computing. R Foundation for Statistical Computing.* (3.6.1). https://www.r-project.org/

Tobii Technology. (n.d.-a). *Tobii Pro Glasses 2*. Retrieved September 20, 2020, from https://www.tobiipro.com/product-listing/tobii-pro-glasses-2/

Tobii Technology. (n.d.-b). *Tobii Pro Lab*. Retrieved September 20, 2020, from https://www.tobiipro.com/product-listing/tobii-pro-lab/

Tory, M., Atkins, M. S., Kirkpatrick, A. E., Nicolaou, M., & Yang, G. Z. (2005). Eyegaze analysis of displays with combined 2D and 3D views. In *Proceedings of the IEEE Visualization Conference 2005* (Vol. 1, pp. 519–526). Minneapolis, MN: IEEE.

Underwood, G., & Foulsham, T. (2009). Saliency and scan patterns in the inspection of real-world scenes : Eye movements during encoding and recognition. *Visual Cognition*, *17*(6/7), 812–834. https://doi.org/10.1080/13506280902771278

von Landesberger, T., Kuijper, A., Schreck, T., Kohlhammer, J., van Wijk, J. J., Fekete, J. D., & Fellner, D. W. (2011). Visual analysis of large graphs: State-of-the-art and future research challenges. *Eurographics Symposium on Geometry Processing*, *30*(6), 1719–1749.

Zhang, J., Ren, J., & Wu, C. (2014). Modeling air traffic controllers' decision making processes with relational complexity network. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)* (pp. 2663–2668). Qingdao, China: IEEE.

Appendix

Chapter 5

Table 3 shows the list of scanpath sequences for three different clearance commands associated with 30 seconds prior to issuing the clearance commands for an expert ATC. CL: 'Clear to land'; CTO: 'Clear to take off'; and LW: 'Line up and wait'.

Table 3: Scanpath Sequence of 30 seconds immediately before issuing clearance command.

CL1:FTBTFBABFTBTABFTRTB CL2:DFTBTFTFBDBFTFTFTRDTFB CL3:FBFTBTRFDRMRBFRBARDB CL4:FDRMBFTFTFTFTATABATFTRTRTRFRFRFTFTF CL5:TDFTABADABABRDB CL6:DABFTBTDTBRCDTDABD CL7:FTATDCDTRDTDAFTFTB CL8:ABABABARMFTFTABR
CTO1:RBRDRTFTFBTDTDB CTO2:RTBTBTBATFTFTBTFTFB CTO3:ABDRDCDRTADT CTO4:BDCRDRBADABABABAD CTO5:DFBABDADFTAFTFDCATAFTDA CTO6:AFTBTFTFTBDTFTBATBADCD CTO7:DTFTFBDRTDTFTFTBDTFTFBTAB CTO8:TDTFTABTBADTFBADRDTFTATRTBAT CTO9:BADRDTFTATRTBATDRTFABATDAD CTO10:BABATBADBADTBADTFRDA
LW1:DTFTRTFTBFTBDTFTFATF LW2:DFTBTFABTFBFTBAF LW3:FTFDRDFBFBRTBADTBRDFRDRA LW4:BDTBTFTDTBATR LW5:AFBTFTBADTFTBTFTBFTBA LW6:ADFATDADTRMDRDTFTFRAT LW7:TABTBRTRDRDRFRDADADMDCTFTAFA LW8:TCDCTFRDRDRDRTDTFDA