

A Guideline for Integrating Dynamic Areas of Interests in Existing Set-up for Capturing Eye
Movement. - Looking at Moving Aircraft

Maik Friedrich¹

Nele Rußwinkel²

Christoph Möhlenbrink¹

September 21, 2016

8478 words (body)

1 - German Aerospace Center, Braunschweig

2 – Technische Universität Berlin

Author:

Maik Friedrich, maik.friedrich@dlr.de

+49 (531) 295 - 2598

German Aerospace Center

Lilienthalplatz 7

Braunschweig, NI D-38108

1 Abstract

2 Today, capturing the behavior of a human eye is considered a standard method for measuring the
3 information gathering process and thereby gaining insights into cognitive processes. Due to the dynamic
4 character of most task environments there is still a lack of a structured and automated approach for
5 analyzing eye movement in combination with moving objects. In this article, we present a guideline for
6 advanced gaze analysis, called IGDAI (Integration Guideline for Dynamic Areas of Interest). The
7 application of IGDAI allows gathering dynamic areas of interest and simplifies its combination with eye
8 movement. The first step of IGDAI defines the basic requirements for the experimental setup including
9 the embedding of an eye tracker. The second step covers the issue of storing the information of task
10 environments for the dynamic AOI analysis. Implementation examples in XML are presented fulfilling
11 the requirements for most dynamic task environments. The last step includes algorithms to combine the
12 captured eye movement and the dynamic areas of interest. A verification study was conducted, presenting
13 an air traffic controller environment to participants. The participants had to distinguish between different
14 types of dynamic objects. The results show that in comparison to static areas of interest, IGDAI allows a
15 faster and more detailed view on the distribution of eye movement.

16

17 *Keywords:* eye movement, dynamic area of interest, experimental setup, requirements

1 A Guideline for Integrating Dynamic Areas of Interests in Existing Eye Movement 2 Environments. - Looking at Moving Aircraft

3 The interaction and observation of dynamic objects is one of the most common actions of
4 intelligent life. Humans routinely analyse their surroundings to find and react to dynamic objects.
5 Example tasks are manoeuvring in traffic or playing basketball. Within those task environments,
6 analysing the movement of dynamic objects, like a car or a ball, helps us to evaluate our current situation.
7 This enables us to select an action to enhance the outcome of our nearest future, such as braking in front
8 of a traffic light or catching the ball. The knowledge on the information gathered to perform a task
9 therefore increases the insight into cognitive processes. The motion of the eye in combination with the
10 used sources of information is defined as eye movement (Seifert, Rötting, & Jung, 2001). This article
11 describes an Integration Guideline for Dynamic Areas of Interest (IGDAI) for capturing and integrating
12 dynamic objects into an experimental setup, and combining them with eye movement.

13 Over the past decades, human computer interaction analysis has gained in attractiveness due to
14 the fact that the best product does not only have the best functionality but also an intuitive control
15 concept. This also led to an increase in popularity for eye movement analysis, as it is an objective method
16 for measuring the information gathering process of humans interacting with interfaces. Eye movement
17 analysis has been used in a variety of domains, such as reading (e.g., Castelano & Rayner, 2008; Granka,
18 Joachims, & Gay, 2004; Rayner, 1995) or image processing (e.g., Henderson, Weeks, & Hollingworth,
19 1999; McCarthy, Sasse, & Riegelsberger, 2004; Rayner & Pollatsek, 1992). The first level analysis of eye
20 movement data usually focuses on the separation between fixations and saccades. Second level analysis
21 concentrates on summarizing first level results in relation to the task environment, such as the
22 combination of an area of interest (AOI) with distribution of eye movement, scan paths or transition
23 matrices. AOIs are defined regions where a certain information is presented within the task environment
24 (e.g. screen) (Duchowski, 2007).

1 The first level analysis requires an algorithm for separation of raw eye movement data into
2 fixations and saccades. For that matter numerous algorithms are already well documented and can be
3 divided by real-time versus post analysis (Duchowski, 2007) and velocity-, area- or dispersion-based
4 (Salvucci & Goldenberg, 2000). However, in a dynamic task environments smooth pursuits (Reimer &
5 Sodhi, 2006) restrains a clear separation between fixations and saccades.

6 The second level analysis usually includes information about the task or the environment and
7 combines these with the eye movement. A common approach is the application of AOI. Combining AOI
8 with eye movement allows insights into the information gathering process within the environment, such
9 as the meaningfulness and visibility of a status bar integrated into a website. The maximum amount of
10 information gathering is reached by combining all visible information and all actions performed by the
11 human. Gathering this kind of data normally takes a lot of planning and consideration of the task and the
12 environment, especially in the case of a dynamic environment with moving objects.

13 This article focuses on the combination of eye movement with dynamic AOI within artificial
14 environments. Studies combining complex dynamic environments and eye movement occur frequently in
15 the domains of driving (e.g., Akamatsu et al., 2001; Palinko, Kun, Shyrovkov, & Heeman, 2010),
16 marketing (Bulling & Gellersen, 2010; Josephson & Holmes, 2006; Nielsen, 2006) or air traffic
17 management (Bruder, Leuchter, & Urbas, 2003; Landry, Sheridan, & Yufik, 2001; Möhlenbrink,
18 Oberheid, & Werther, 2008). These studies use eye movement within artificial environments that
19 comprise moving objects that are partially or completely controlled by participant actions, such as the
20 view out of a moving car, interaction with web pages or air traffic controller interfaces. The moving
21 objects within these different environments are of importance to the tasks. However, these studies
22 concentrate on the application of dynamic AOI and do not aim to improve the method itself, some of
23 them even do not publish their method. Comments in the conclusions of those studies state that
24 combining eye movement data with dynamic objects is extremely time consuming and specialized only
25 for the described environment (e.g., Papenmeier & Huff, 2010).

1 As described above, regions within the task environment that are a source of information and
2 useful to the eye movement analysis are considered as AOI. Attributes to describe an AOI within a task
3 environment are identifier, shape and position. The identifier is unique for each AOI within a task
4 environment. Shape defines the area of the AOI. Position defines the centre of the shape in relation to the
5 task environment. The static and dynamic AOI can both be defined using identifier, position and shape.
6 The distinction between static and dynamic AOI can be explained using a coordination system as
7 reference for the areas over time. Static AOI have the same position, shape and size in relation to the
8 reference system at all times. Dynamic AOI change their position, form, orientation, or size in relation to
9 the reference system over time. An example for a dynamic task environment is a radar screen because it
10 contains both types of AOIs. A radar screen normally displays a two dimensional top view of airspace
11 sectors. Assuming that the radar screen is fixed over a particular sector, it provides a two dimensional
12 coordination system as reference for AOIs. The radar screen sometimes shows motor highways, rivers or
13 forests, to increase the orientation for the air traffic controller. These areas never move in relation to the
14 screen and therefore represent static AOIs. The radar screen also shows aircraft that move through the
15 sector. The moving aircraft are AOIs that change their position in relation to the screen and are therefore
16 dynamic.

17 Considering the differences between static and dynamic AOIs, an analysis of eye movement data
18 in combination with only static AOIs would not provide a comprehensive view on the used sources of
19 information. The solution is to define dynamic AOIs for moving objects and combine them with eye
20 movement data in post processing. Thereby the separation of the dynamic information and the static
21 background becomes possible and increases the significance of the data analysis.

22 The radar screen example above demonstrates the three main issues for applying dynamic AOIs.
23 The first issue (Set-up requirements) is used to configure the environment to capture the needed
24 information. The second issue (Storage requirements) is used to identify the information that have to be
25 recorded and derive a suitable data structure. The third issue (Data Analysis Methods) is used to match

1 eye movement and AOIs. All three issues depend on each other; this means that the solution to the first
2 issue inflicts the solution to the second and third issue. An overall approach that addresses all issues and
3 is universal for the application to any task is needed for applying dynamic AOI's to any environment.
4 Below an overview of existing approaches for combining eye movement and dynamic environments is
5 provided.

6 **Existing approaches**

7 Research connected to dynamic AOIs ranges from theoretical descriptions (e.g., Bruder et al., 2003) to the
8 application of methods that are customized to the studied task or environment (e.g., Papenmeier & Huff,
9 2010). The three issues identified above are usually covered in the method section by shortly describing
10 the individual solutions. Only a few publications try to give an overall view covering all three issues
11 connected to dynamic AOIs. To assess the different approaches we have to investigate how the three
12 issues are solved.

13 The approaches can be categorized into online and offline. Applying dynamic AOIs online means
14 to match the gaze vector with the dynamic objects within the environment at the time the eye movement
15 is recorded. In contrast, recording dynamic AOIs offline means the matching is made in a post processing
16 step. One example for an online matching of eye movement and dynamic AOIs is presented by
17 Sennersten et al. (2007). They combined an eye tracking system (Tobii) and a game engine (e.g.
18 CryEngine (Crytek GmbH, n.d.)) and transferred the gaze position and direction from the eye tracker
19 online into the virtual environment generated by the game engine. Thereby they cover the Set-up
20 requirements and the Storage requirements. Then they calculate where the eye vector hits the first surface
21 within their scene. The name of this surface is then stored for further analysis. This covers the Data
22 Analysis Method issue. Avoiding the post processing of eye movement and dynamic AOI's by online
23 matching is time efficient. However, playback of the analysis is sometimes difficult because the complete
24 virtual environment, all participant actions and eye vectors have to be stored. This problem is increased if
25 the analysis has to be rerun, in case e.g. that some objects (e.g. windows) are to be removed from the

1 analysis (Papenmeier & Huff, 2010) or the calibration of the eye tracker had a small divergent. The online
2 matching of dynamic AOIs also has the disadvantage of not applying any method for correcting the eye
3 movement data (Hornof & Halverson, 2002; Zhang & Hornof, 2011). Due to the research presented
4 above, the authors believe that at the moment, methods of replay and post correction are necessary in the
5 field of eye movement analysis. Therefore, the offline matching of dynamic AOIs is pursued in the
6 following research.

7 A straightforward approach for realizing offline matching of dynamic AOIs is to use an eye
8 tracking system and a scene camera. Those combined generate videos that include the scene and the
9 calculated eye position, also covering Set-up requirements and Storage requirements. The Data Analysis
10 Method is covered while post processing the dynamic objects. They are identified offline and manually
11 (e.g. Land & McLeod, 2000; Sennersten, 2004), by separated rating of every frame of the combined
12 video. This approach is time consuming and subjective since the position of the eye in relation to dynamic
13 AOIs can be interpreted differently.

14 BeGaze 3.0, an analysis software developed by SMI, also supports offline definition of dynamic
15 AOI. Here the analyst can define key frames with AOI and the software interpolates the positions of the
16 AOIs depending on the number of frames between the key frames. The dynamic AOI can even be stored
17 for repetitive use. However, the interaction between operator and environment changes the most scenarios
18 and creates an individual set of moving objects for each run (and therefore also AOI), therefore the
19 repetitive use of BeGaze is limited. An automated procedure for synchronising video files with eye
20 movement data is presented by Papenmeier and Huff (2010). They present DynAOI which matches the
21 gaze position with three-dimensional models underlying the presented video files. The approach converts
22 the captured eye movement data on a two dimensional screen into a three-dimensional representation of
23 the events within the presented video files. The gaze vector within the three-dimensional representation
24 then is used to determine all objects on its path, with the object then serving directly as AOIs. Huff,
25 Papenmeier, Jahn, and Hesse (2010) use this approach with real time generated videos.

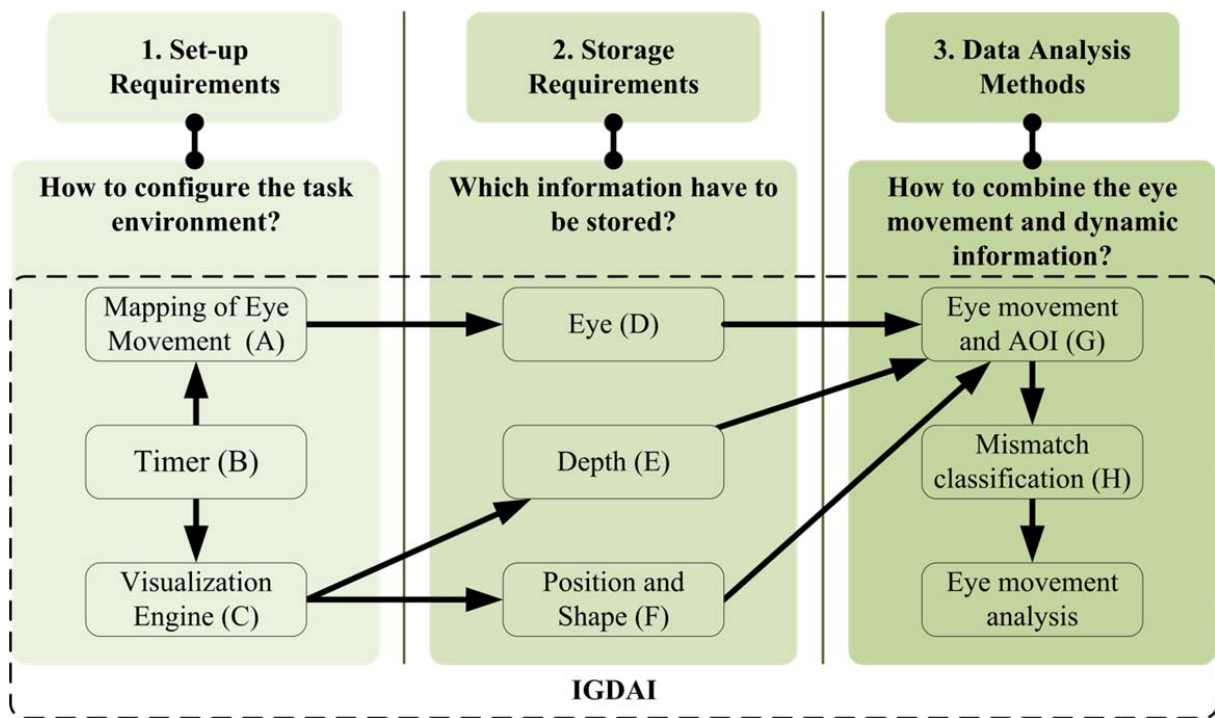
1 Bruder et al. (2003) propose an approach by presenting a set of technical requirements for
2 capturing the dynamic information, covering the Set-up Requirements. The requirements include that eye
3 movement is captured in relation to the screen on which the dynamic objects are presented. They cover
4 the Storage Requirements by suggesting the application of markers, generated by the environment and
5 stored with the eye movement data indicating a change of information on a screen.

6 Another approach for the Data Analysis Method, depending on a synchronized and separate
7 recording of eye movement and dynamic AOIs, is proposed by Zelinsky and Neider (2008) using the
8 shortest distant rule for assigning the eye movement data on a screen to the presented dynamic AOIs.
9 They calculate the distances between each measured eye movement point on the screen to each centre of
10 the visible dynamic AOI. The dynamic AOI that is closest to the eye movement is then classified as
11 “looked at”. This approach can be implemented quickly and provides a good estimation of the results
12 (Gross, Friedrich, & Möhlenbrink, 2010) if bigger dynamic AOIs further away do not cover smaller ones
13 that are nearer to the eye movement. Fehd and Seiffert (2008) extended the shortest distant rule with a
14 location competition analysis that calculates a weight for each moving target in relation to the nearest eye
15 movement. The weights are accumulated over all frames to identify the targets. This approach will lead to
16 a distribution of eye movement data on to dynamic objects but has sometimes difficulties if the
17 participants do not look at any object.

18 **IGDAI**

19 IGDAI was developed as an efficient way to cover all three issues (Set-up, Storage, and Data Analysis) in
20 a structured and flexible manner that is need for each individual eye tracking task environment. Because
21 of the wide variety of different eye tracking tasks and environments, IGDAI does not determine a fixed
22 solution. IGDAI addresses the main issues of integrating dynamic AOIs into an environment for capturing
23 eye movement by defining requirements that need to be fulfilled. Therefore the implementation of the
24 requirements are open and can be adapted to the individual environment but are at the same time

1 interconnected by following IGDAI. IGDAI also includes an example implementation that fulfils the
 2 requirements, to raise understanding for the process.
 3 The guideline begins by defining Set-up requirements (Figure 1 A, B, and C) that concern the process of
 4 data capturing to ensure the needed output information. Requirement D (Figure 1) covers the storage of
 5 the eye movement data with all necessary information. Then Requirement E (Figure 1) accounts for
 6 overlaid visual information and thereby supports the process of post analysis. IGDAI provides a data
 7 structure that can store dynamic information as dynamic AOIs (Figure 1 F). An example of how to store
 8 the necessary information for the post analysis is provided below. Taking all the previous steps into
 9 account, IGDAI describes the combination of eye movement and dynamic AOIs (Figure 1 G and H) with
 10 the depth information. Practical solutions are provided for matching eye movement data with dynamic
 11 AOIs.



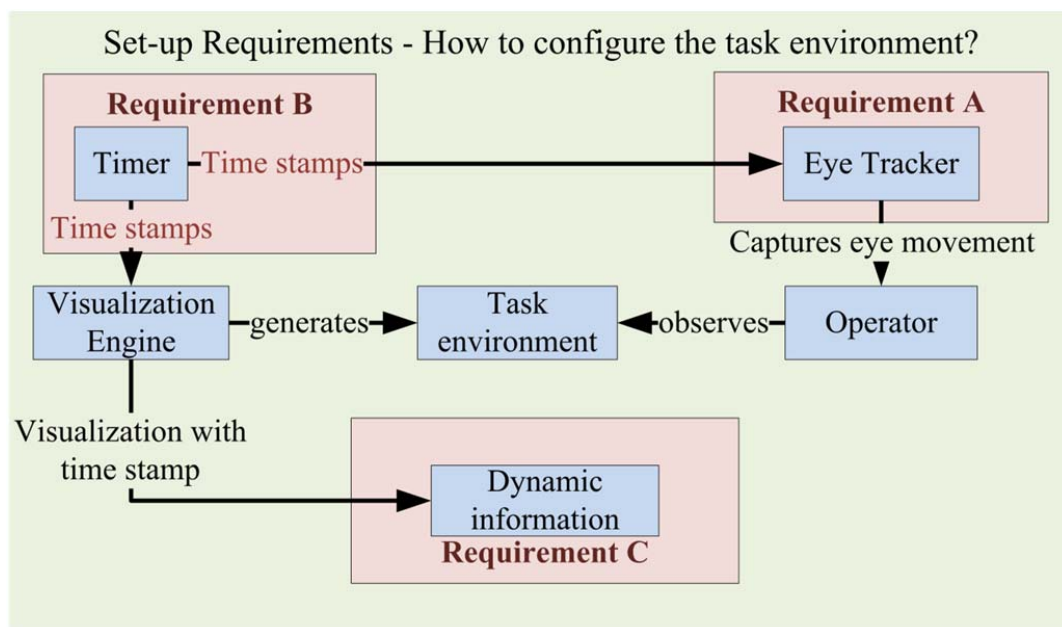
12

13 *Figure 1 IGDAI separated into three issues and seven requirements.*

14

Set-up requirements

1 The Set-up requirements (Figure 1 and Figure 2) are Mapping of Eye Movement (A), Timer (B) and
 2 Visualization Engine (C). Requirement A: The eye tracker shall have the capability to map the eye
 3 movement onto a reference system (e.g. a surface representing the computer monitor). Requirement B:
 4 The capturing of eye movement and dynamic information shall be time synchronized. Requirement C:
 5 The visualization engine shall log the dynamic information on the same reference system as used by the
 6 eye tracker. The main purpose of the Set-up requirements is to harmonize the reference systems between
 7 the eye tracker and visualization engine, in preparation for combining the eye movement data and the
 8 dynamic AOIs.

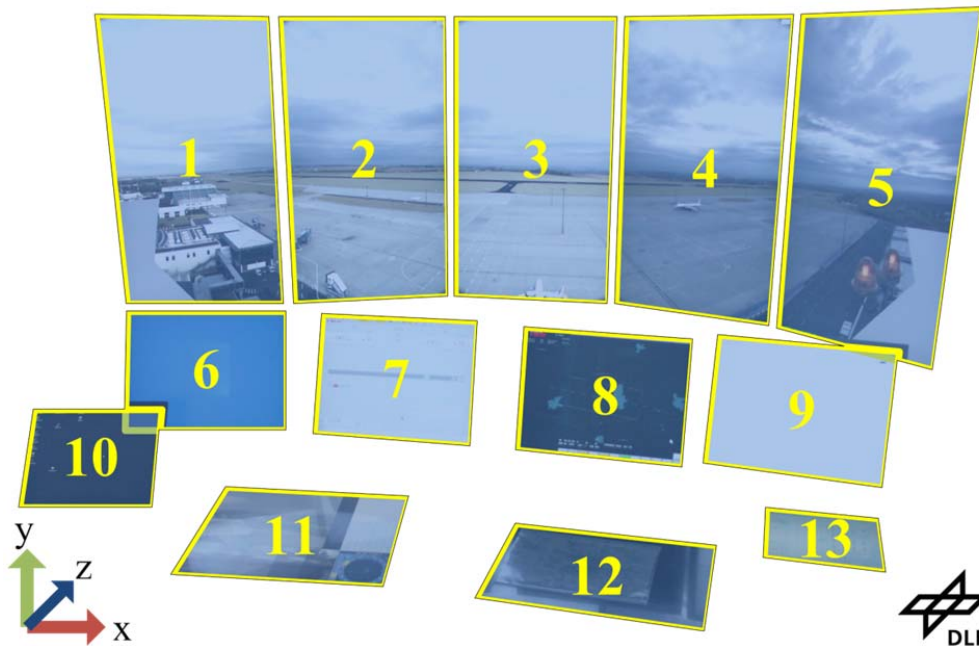


9

10 *Figure 2 Set-up Requirements in relation to capturing of eye movement data and dynamic AOIs.*

11 Figure 2 shows the components of a simplified set-up in relation to the requirements A, B and C, which
 12 we will now discuss in detail. As mentioned above, Requirement A concerns the capturing of the eye
 13 movement data in relation to a reference system. A solution to fulfil the requirement is to use virtual
 14 surfaces as reference systems and map the eye movement onto virtual surfaces that are for example equal
 15 to the screens in the environment. Therefore the eye tracker needs to have the capability of defining
 16 virtual surfaces for mapping the eye movement onto them. These virtual surfaces should be coherent with

1 the actual monitors that present visual information. E.g. Figure 3 shows the virtual surfaces that cover a
 2 multi monitor workplace. The eye movement is determined as the intersection of gaze vector (vector with
 3 the eye position as origin) with the first virtual surface on its path. The result of this requirement is x, y
 4 and screen information of the eye movement.



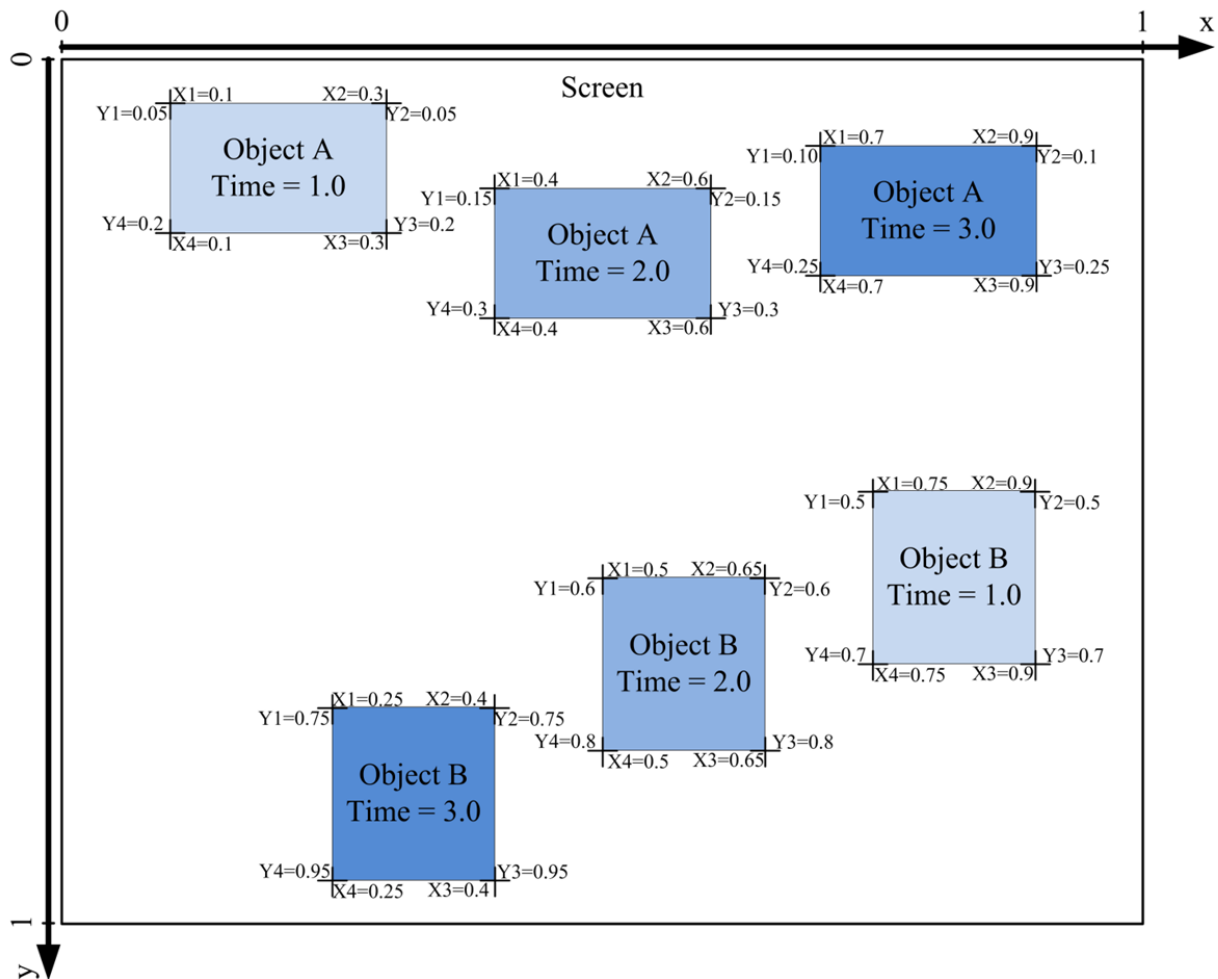
5
 6 *Figure 3 Multiple virtual surface setting for a multi monitor workplace (Friedrich, Möhlenbrink,*
 7 *& Carstengerdes, 2012)*

8 Requirement B concerns the time synchronized capturing of eye movement and dynamic
 9 information. If the eye tracker and the visualization engine are separate systems, which they often are,
 10 their processes of capturing and storing the data need to synchronize. This requirement can be fulfilled
 11 through the application of a timestamp that is used by both systems. The introduction of a timestamp can
 12 be realized in several different ways. Figure 2 shows an example solution where an external timer
 13 provides the timestamp for both systems. An alternative implementation could be that either the eye
 14 tracker or the visualisation engine provides timestamps and distributes them to the respective other
 15 system. Dependent on the implementation the synchronized timestamps there are also different sources
 16 for time lags that also have to be taken in consideration. The three major sources are the processing time

1 of the eye tracker, the time between logging and showing the AOIs on the screen and the duration to
2 synchronise eye tracker and visualisation engine. The resolution of the timestamp should be in the range
3 of milliseconds, otherwise the accuracy of the eye tracker (modern eye tracker reach up to 2000 Hz) is
4 decreased. A rule of thumb, from the author's experience, for two systems is that the time deviation is less
5 than 15 milliseconds for one minute. Therefore, synchronizing every 10 seconds should be enough for a
6 200 Hz eye tracking system. Independent from the implementation, the result of this requirement is that
7 the presented visual information and the captured eye movement are synchronized by timestamps.

8 As Figure 2 shows, Requirement C concerns the generating of dynamic information in relation to
9 the same reference system as the eye movement and with a logging rate. If the virtual surfaces
10 (Requirement A) cover the screens in the task environment, as proposed, the reference systems for the
11 dynamic AOIs should be the same screens. In addition to their identifier, position, shape, orientation and
12 size, dynamic AOIs also have timestamps (Requirement B) depending on the logging rate. The logging
13 rate depends on the context change of the task environment (e.g. 1Hz for an ATC radar workplace or 60
14 Hz for a driving simulator). For example if the task environment changes with 1 Hz, a logging rate of 1
15 Hz is enough because a higher logging rate would not generate more information but only duplicate the
16 already logged one.

17 Figure 4 shows an example of the information needed to fulfil Requirement C. The two objects
18 on the screen change their position once per second and are therefore logged with 1 Hz. The objects move
19 for a total of three seconds. The time is synchronized with the eye movement. The name of the object is a
20 unique identifier that is consistent over time. Each set of X and Y represent a corner of an object. The X
21 and Y values are percentages of their particular axis (e.g. on a screen with 1920 pixel horizontal
22 resolution $X1 = 0.10$ stands for pixel 192).



1

2 *Figure 4 Two dynamic AOIs moving for three seconds in different directions*

3 Requirements A, B and C are the necessary extensions to an existing eye movement environment.
 4 If fulfilled, the set-up captures the eye movement and the dynamic AOIs synchronized and in relation to
 5 the same reference system. This completes the first step (Figure 1) of IGDAI.

6 **Storage requirements**

7 IGDAI sets-up the task environments to record dynamic information, but also prepares the
 8 analysis process by structuring the storage of the dynamic information. The requirements for the storage
 9 of dynamic information are Eye (Figure 1, D), Depth (Figure 1, E) and Position and Shape (Figure 1, F).

10 Requirement D: The eye movement data shall be stored with the timestamp. Requirement E: The dynamic

1 information shall be stored with depth information to distinguish between visible and hidden objects.

2 Requirement F: The position and shape of dynamic information shall be stored as AOIs with a timestamp.

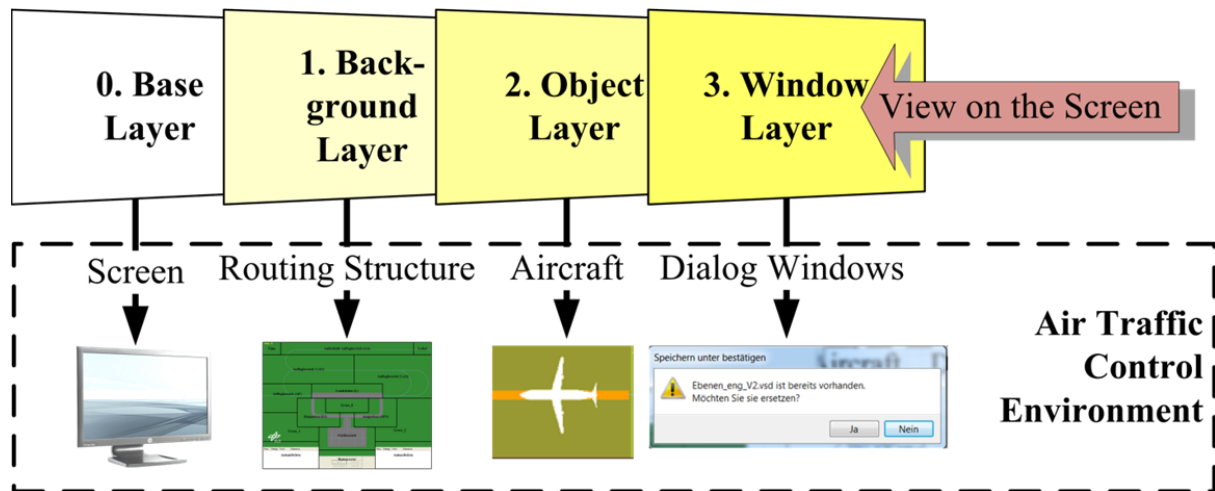
3 Requirement D concerns the storage of the eye movement data with the appropriate timestamp
4 and in relation to the reference system (Requirement A). The result of Requirement A is the information
5 of x, y and the screen of each eye movement point. Requirement D is fulfilled if this information are
6 logged together with the timestamp of the capturing. The information can be logged as a new line into a
7 log file. If stored correctly, this file can be loaded automatically and used for the data analysis.

8 Requirement E concerns the storage of AOIs that overlay each other within a task environment.
9 This requirement can be fulfilled by storing the depth information of every AOI in relation to the
10 reference system (Requirement A and C). In 3D task environments the depth information could be
11 described as distance from the dynamic object to the reference system. The dynamic object with the least
12 depth would be the visual object in the foreground unless it is transparent or semi-transparent. In 2D task
13 environments no depth information is available. IGDAI suggests the usage of a layer model for a 2D task
14 environment to put the dynamic objects in order in relation to the reference system.

15 The layer model orders the dynamic objects by high (nearest to the reference system) and low
16 (furthest away from the reference system), but is independent from the task relevance of the visual
17 information. The lowest layer should contain an AOI for the complete reference system (screen) to secure
18 that each eye movement point can be assigned to at least one AOI. The amount and order of applied
19 layers depends on the task environment and can be redefined for each different task environment.

20 Figure 5 shows an example implementation of the layer model adapted to a 2D air traffic control
21 environment with four different layers. The base layer contains the screen where the visualization is
22 presented. The window layer is reserved for dialog windows that are on top of the other layers. The object
23 layer contains the dynamic objects that change their position in relation to the observer, for example the
24 moving aircraft on a radar screen. The background layer contains static information that does not change

1 their position in relation to the reference system, for example routing structure of an air space or the speed
 2 indicator within a driving environment. The base layer contains the screen.



3
 4 *Figure 5 The layer model used for an air traffic controller task environment*

5 Requirement F is fulfilled by storing the shape and position for every timestamp as a new line
 6 into a log file. The shape of the AOI could be defined by points along a path defining a convex or concave
 7 polygon. As preparation for the data analysis the path should be equally (clockwise or counterclockwise)
 8 for all logged AOIs, because some algorithms for solving the point-in-polygon problem have one or the
 9 other order as restriction, e.g. the Winding number algorithm (Hormann & Agathos, 2001). If a complex
 10 polygon (the path intersects itself) is used special methods have to be applied for the analysis (Galetzka &
 11 Glauner, 2012). The position of the AOI represents the information where to place the shape in relation to
 12 the screen. IGDAI supports Requirement F by providing a XML data structure that accounts for these
 13 assumptions that dynamic information do not change their position or shape over the majority of time and
 14 also integrates the depth information. The XML data structure is optimized for one screen only, which
 15 means that for a multi monitor environment only the screen with actual dynamic AOIs have to be logged.

16 The IGDAI XML data structure separates (Table 1) the shape of an AOI from its movement, size,
 17 or orientation and from the time it is visible. IGDAI XML has the main node types ShapeTemplateList,
 18 UpdateList, and AoiList. ShapeTemplateList contains shape template nodes that describe the shapes (e.g.

1 list of points) that occur during the recording. The UpdateList contains an Update node for each time
 2 interval describing the current position and depth (3D: distance; 2D: layer model) of a dynamic AOI. The
 3 nodes in UpdateList can also contain size or orientation changes. The AoiList contains the link between
 4 the dynamic AOIs and their shapes. The reuse of equal shapes reduces the amount of data stored. Table 1
 5 shows the XML example implementation of an aircraft shape template in combination with AoiList using
 6 the shapes and the update of the screen at second 2.

7 Table 1 ShapeTemplateList, UpdateList, and AoiList implemented in XML.

ShapeTemplateList node for an object with 4 points	AoiList
<pre> <ShapeTemplate id="Object_01"> <Outline NumberPoints="4"> <Polygon> <Point x_rel="-0.05" y_rel="-0.05"/> <Point x_rel=" 0.05" y_rel="-0.05"/> <Point x_rel=" 0.05" y_rel=" 0.05"/> <Point x_rel="-0.05" y_rel=" 0.05"/> </Polygon> </Outline> </ShapeTemplate> </pre>	<pre> <DynamicAoi firstSample="22" AOI_ID="Object_A" template="Object_01" /> <DynamicAoi AOI_ID="Object_B" template="Object_01" /> </pre>
UpdateList node: second 62 from the storing	
<pre> <Update Second="2.0"> <dAOI AOI_ID ="Object A" > <Position x="0.25" y="0.65" Depth="2"/> <Size s="0.25"/> </dAOI> <dAOI AOI_ID ="Object B" > <Position x="0.775" y="0.55" Depth="2"/> <Orientation r="270"/> </dAOI> </Update> </pre>	

8
 9 The following description explains the process of storing dynamic AOIs within the visualisation
 10 engine. For each timestamp the dynamic objects have to be transformed to the reference systems
 11 (Requirement A and C). Then the shapes of the previous timestamp are compared with the current shapes
 12 based on their AOI_IDs. The comparison identifies if the position, orientation or size has changed. If one
 13 of the values has changed the update contains this information. If a new AOI_ID is introduced its shape is

1 compared to all existing nodes in the ShapeTemplateList. If no match is found a new node containing the
2 shape of the new AOI_ID is added to ShapeTemplateList. The AOI_ID is then added to the AoiList and
3 Update node. With Requirement D, E, and F we are now able to combine the dynamic AOIs with the eye
4 movement data.

5 **Data Analysis Methods**

6 In preparation of the eye movement analysis IGDAI (Figure 1) defines a requirement for
7 matching eye movement and AOIs (G). Requirement G: For each timestamp with eye movement data
8 shall a set of AOIs separated by depth be available. The eye movement uses the same timestamp
9 (Requirement A and D) as the stored AOIs. The goal is to assign dynamic AOIs to the eye movement.

10 Algorithm 1 describes pseudocode for automatic matching of eye movement and AOIs. By taking
11 the layer model into account the matching algorithm would start by checking the AOIs in the highest
12 layer (Figure 5, window layer). Checking whether an eye movement point is inside an AOI is a point-in-
13 polygon problem (for complex polygones Galetzka & Glauner, 2012; for simple polygones Taylor, 1994).
14 If the eye movement point does not match with the AOIs on the highest layer, the matching process is
15 continued using the AOIs from the layer directly below (Figure 5, object layer). This process is continued
16 for all layers until an AOI is identified that contains the eye movement point. The result is that each eye
17 movement point is assigned to only one AOI.

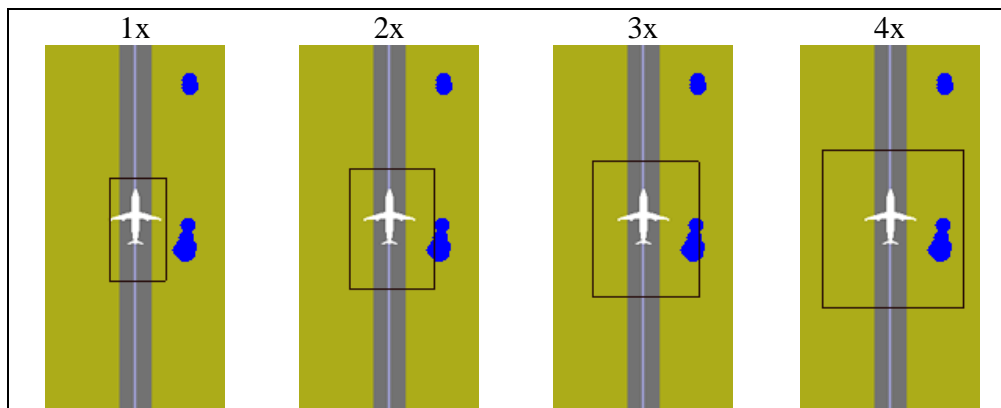
1 Algorithm 1 Merge algorithm to identify the correct AOI for every captured eye movement
 2 point.

```

function Mergealgorithm
allEyePoints = LoadEyePoints()
DynamicAOIList = LoadAOIList()
foreach (eyePoint in allEyePoints){
    time = get_Time (eyePoint) # Determine timestamp for the current eye movement point
    allLayers = get_Layers_From (DynamicAOIList; time) # Determine the current AOIs
    foreach (layer in allLayers){ # Start with the highest layer first
        foreach (polygon in layer){
            if Check_Point_in_Polygon (Polygon, eyePoint) == TRUE
            then {
                polygonIDs.add(Polygon, Distance_to_PolygonCenter(eyePoint))
            }
        }
        if count(polygonIDs) > 0
        then {
            eyePoint.setAOI (Shortest_Distance(polygonIDs))
            break # jump to the next eyePoint
        }
    }
}
  
```

3 IGDAI also accounts for two special cases that arise by matching eye movement to AOIs,
 4 independent from the implementation. Requirement H: The special cases shall be addressed to conduct
 5 the eye movement analysis. The first special case occurs if an eye movement point can be assigned to
 6 multiple AOIs on the same layer. This inhibits a singular classification between eye movement and the
 7 gathered information. This special case will occur for example if a pedestrian covers a traffic sign in a
 8 driving simulation and both AOIs are classified in the same layer. In a 3D task environment the eye
 9 movement point should be assigned to the AOI with the least depth (Requirement E). In a 2D task
 10 environment IGDAI recommends to determine the distance between the eye movement point and the
 11 centre of all matching AOIs. The eye movement point is then classified based on the shortest distance
 12 (Algorithm 1, Shortest_Distance function), assuming that the AOIs surround the visual information
 13 equally centred.

1 The second special case is the mismatch of eye movement and AOI due to the accuracy of eye
2 tracking systems. Figure 6 (1x) shows an example of eye movement aggregated for one second (blue
3 dots) and the nearest AOI (square). The eye movement in Figure 6 (1x) would be classified incorrectly on
4 a lower layer, probably the lowest one, because of the accuracy of the eye tracker system. IGDAI
5 recommends an enlargement of the AOIs if it is possible to determine the degree of accuracy. The
6 accuracy consists of the accuracy of the gaze vector and the accuracy of calibration. The accuracy of the
7 gaze vector should be provided by the eye tracking manufacturer. The accuracy of calibration depends on
8 the calibration of the participant and the stability of the calibration throughout the experiment (e.g. a head
9 mounted eye tracking system could dislocate). Therefore, the degree of accuracy should to be checked in
10 small intervals throughout the capturing of the eye movement, e.g. by calibration points during the
11 experiment. A scale factor for enlargement can be calculated using the tangents of the degree of accuracy
12 (normally expressed in angle), the distance of the participants to the screen and the original size of the
13 AOI on the screen. Figure 6 (1x to 4x) shows the offsetting of the AOI with three different scale factors.
14 If the degree of accuracy cannot be checked during the experiment due to experimental design or task
15 environment constrains, IGDAI recommends using no scale factor (1x) for the eye movement analysis.
16 Even if no scale factor is used, a set of self-selected scale factors can help to determine the quality of the
17 recorded eye movement. This analysis is based on the assumption that each participant must dedicate an
18 average amount of eye movement to the dynamic AOIs to solve the task. A calibration error (e.g. a
19 dislocated eye tracking system) could lead to a strong increase of eye movement data in the perimeter of
20 the dynamic AOIs. A systematically increasing of the dynamic AOIs helps to identify participants with an
21 abnormal increase in assigned eye movement. An individual examination of the eye movement data of
22 these participants should help to decide if their data is valid for further analysis. An example analysis is
23 provided in the result section.



1 *Figure 6 Eye movement data (dots) and dynamic AOIs on the object layer (square) with different*
2 *area sizes.*

3 IGDAI defines requirements for the task environment, provides a structure for recording the
4 dynamic AOIs and then simplifies the post processing by taking into account the previous steps. IGDAI is
5 kept generic to be applicable to a large variety of task environments. An IGDAI implementation was done
6 by the German Aerospace Center (DLR) using a SMI head mounted eye tracker, an air traffic controller
7 simulation (FAirControl) as task environment and the analysing software Eye Tracking Analyzer
8 (EyeTA). The EyeTA was also developed by the DLR and represents the first software that implements
9 IGDAI for the purpose of standardized eye movement analysis with dynamic AOIs¹.

10 **Verification of IGDAI**

11 The feasibility of IGDAI is shown by implementing it into a task environment and showing the
12 effect for the analysis. We selected FAirControl (Möhlenbrink, Werther, & Rudolph, 2007) as a task
13 environment. For the verification, we assume the focus of the study is to identify if the participants gather
14 information by monitoring fixed points or pursuing dynamic objects to generate a decision.

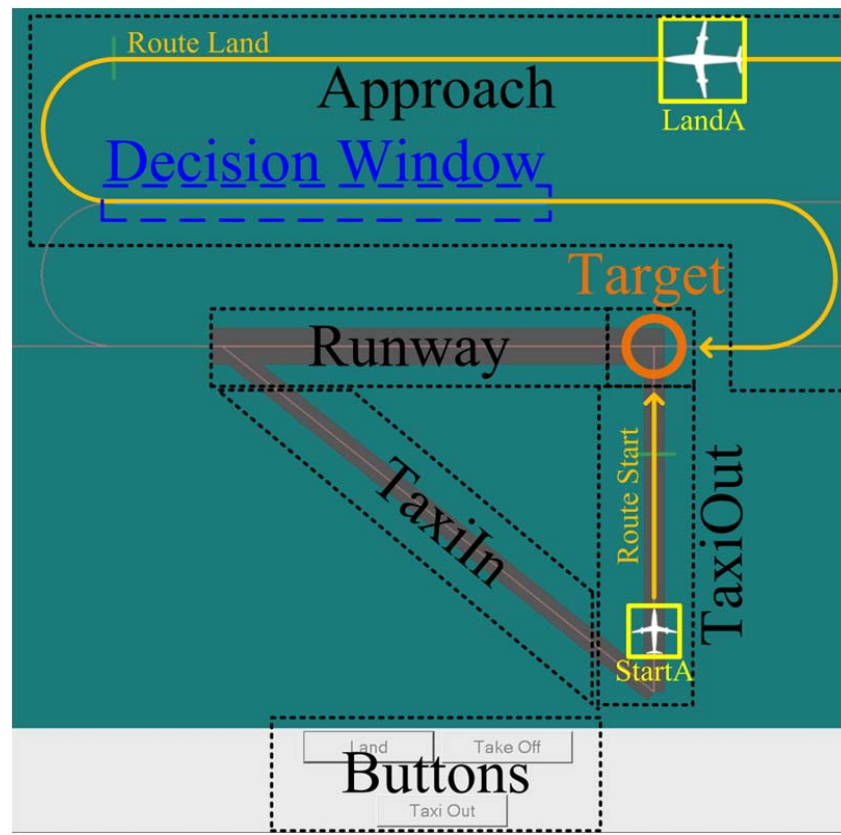
15 **Participants**

¹ This software has to be requested from the author due to DLR restrictions. [A short video presentation is available at dlr.de/fl/Portaldata/14/Resources/dokumente/A_Guideline_for_Integrating_Dynamic_Areas_of_Interests.mp4](https://www.dlr.de/fl/Portaldata/14/Resources/dokumente/A_Guideline_for_Integrating_Dynamic_Areas_of_Interests.mp4)

1 The group of participants consisted of 9 female and 11 male ($N = 20$) students with an average
2 age of 24 years ($SD = 2.12$ years). The participants studied at the Technical University of Berlin. They
3 participated in the experiment in separate sessions and were compensated with 10 €

4 **Material**

5 The task environment to capture the eye movement was set up as proposed by IGDAI and
6 therefore similar to Figure 2. FAirControl represents an airfield from a top view (Figure 7). The airfield
7 consists of a runway and two routes to reach the runway entry point (target circle). FAirControl simulates
8 an aircraft (LandA) landing on Route Land and an aircraft (StartA) taking off on Route Start. The aircraft
9 begin simultaneously on both routes. The time for an aircraft to fly its particular route varies between 9.5
10 and 11 seconds. The distance between the aircraft to reach the entry point vary between 1.5 seconds
11 (LandA first on Target) to -1.5s (StartA first on Target), but was never zero. The speed from the aircraft
12 on Route Land varied between 6.8 and 7.9 cm per second. The AOI of LandA was a square with sides of
13 2.5 cm. The speed from the aircraft on Route Start varied between 0.9 and 1.1 cm per second. The AOI of
14 StartA was a square with sides of 1.5 cm.



1

2 *Figure 7 FAirControl, the task environment that was used for verification of IGDAI*

3 FAirControl was presented in full screen mode on a 22-inch display (17.6-inch wide by 13.7-inch
 4 high) with a resolution of 1280 x 1024 pixels. A SMI RED eye tracker system with a 60 Hz sampling rate
 5 was used to capture the eye movement. The eye tracker warning system made sure that the participants
 6 kept a distance of 65cm and stayed within an area of 50x30 cm to the centre of the screen.

7 The eye tracker system fulfils Requirement A. The timer (Requirement B) was implemented into
 8 FAirControl. Therefore the eye movement data was sent to FAirControl and a timestamp was added. If no
 9 eye movement data was sent or the eye movement point was not on the screen, FAirControl added zero
 10 values to the x and y coordinates in the eye log file (Requirement D). The logging of the dynamic AOIs
 11 were also set to 60 Hz to match the eye movement. FAirControl stored the dynamic AOIs using the layer
 12 model (Requirement E) and the data structure (Requirement F). The chosen layer names and depths of the
 13 layers are equal to Figure 5, except for the window layer that was not used, because FAirControl can be

1 handled without extra dialog windows. The base layer contains the screen. The background layer contains
2 AOIs for approach, runway, both taxiways (in and out), the target, and the buttons (Figure 7). The object
3 layer contains the bounding boxes surrounding the aircraft that moved via both routs.

4 **Procedure**

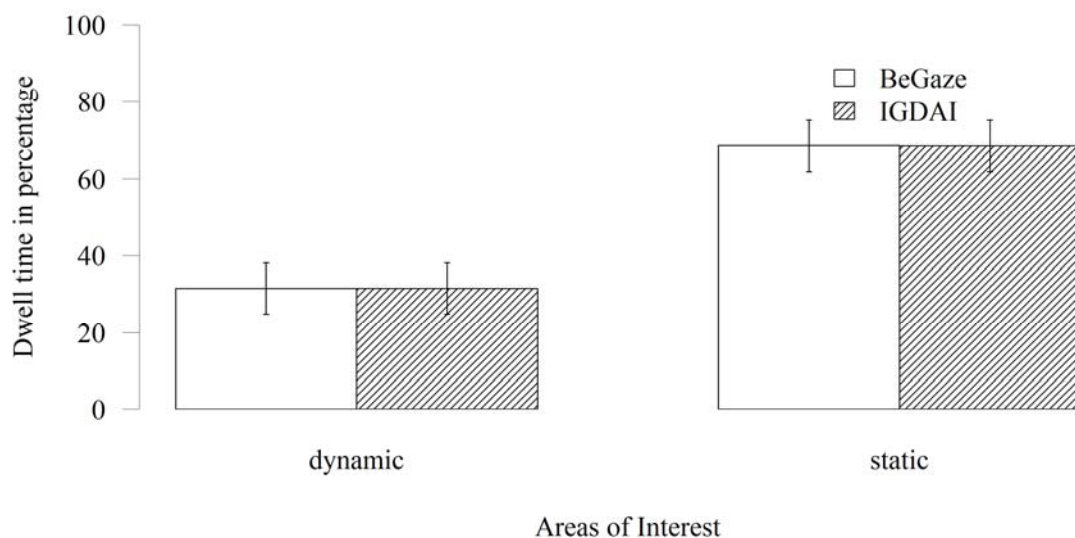
5 After reading the instructions and prior to the experimental session, the eye tracking system was
6 calibrated to each participant. In a 10 minutes training session, participants were familiarized with the
7 FAirControl. The participants were instructed to monitor FAirControl and decide which aircraft would
8 reach the runway entry point (Figure 7, Target) first. They answered by pressing one of the buttons and
9 while the aircraft on Route Land moved through the decision window (Figure 7). After the decision was
10 made each aircraft continued their route and the participants got feedback if they were correct or not.
11 When the aircraft on Route Land reached the start position of Route Start the next cycle began. Every
12 participant assessed 60 randomized cycles with an equally distributed number of aircraft from both routes
13 reaching the runway entry point first. The experiment took approximately 17 minutes to complete.

14 **Results and Discussion**

15 As described above, the IGDAI layer model was used to classify the different types of AOIs used
16 within FAirControl. Due to the problem of the clear distinction between fixation, saccades and smooth
17 pursues in dynamic task environments the data analysis was based on the raw eye movement and the
18 classification by the IGDAI layer model. To verify the impact of IGDAI to the analysis of eye movement,
19 we performed three analyses. The first analysis quantifies the acceleration of data preparation by IGDAI
20 in relation to a manual description. The second analysis quantifies the quality of the recorded eye
21 movement. The third analysis shows the impact of the layer model to quantify the eye movement
22 distribution.

23 The first analysis shows the influence of IGDAI on the AOI preparation time (amount of time
24 needed for generating AOIs) and the quality of the AOIs. The analysis was performed using BeGaze to
25 transcript 5 randomly selected data sets without using IGDAI. The 5 datasets had an average length of

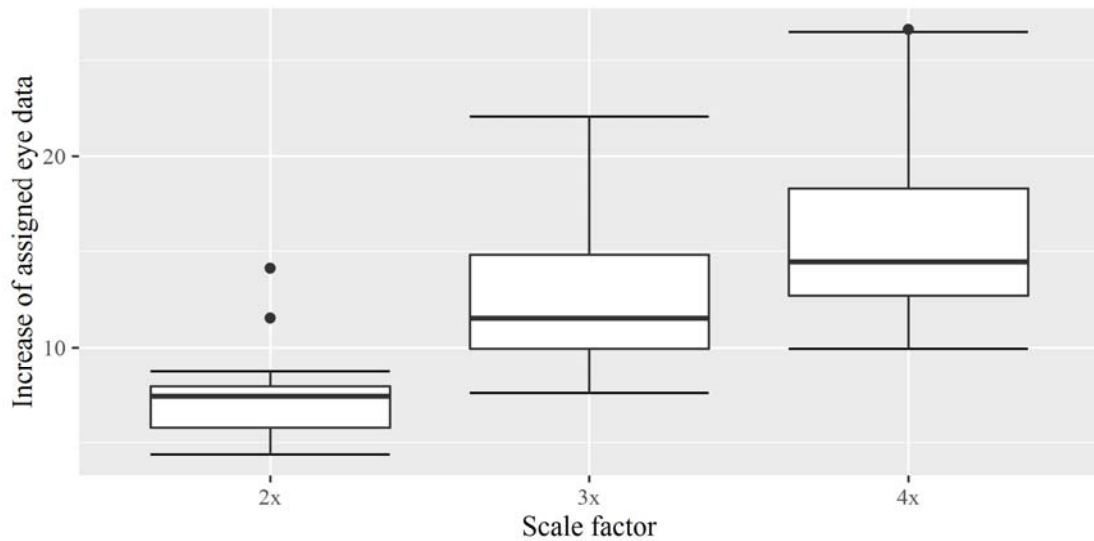
1 17.2 minutes (SD = 0.95) with 2 dynamic (LandA and StartA) and 7 static (Screen, Approach, TaxiOut,
 2 Buttons, Target, Runway, and TaxiIn) AOIs. The AOI preparation time took 35.5 minutes in average (SD
 3 = 4.9). Therefore, the transcription of all 20 data sets would have taken approximately 11 hours. The AOI
 4 preparation time without IGDAI increases linear with the number of participants. The AOI preparation
 5 time with IGDAI was zero, because the capturing of AOIs took place while conducting the study. The
 6 additional effort for implementing IGDAI into the task environment was approximately 6 hours. The
 7 implementation time for IGDAI is a fixed value independent from the number of participants. The
 8 influence of IGDAI on the quality of AOIs was determined by comparing the different dwell times in
 9 percentage. Figure 8 shows that the results from BeGaze and IGDAI are similar. The difference between
 10 the dynamic AOIs is 0.02 percent (BeGaze 31.49%; IGDAI 31.51%). Either dynamic [$t(4) = -0.10807$,
 11 $p < .91$] nor static [$t(4) = 0,14887$, $p < .88$] showed significant differences.



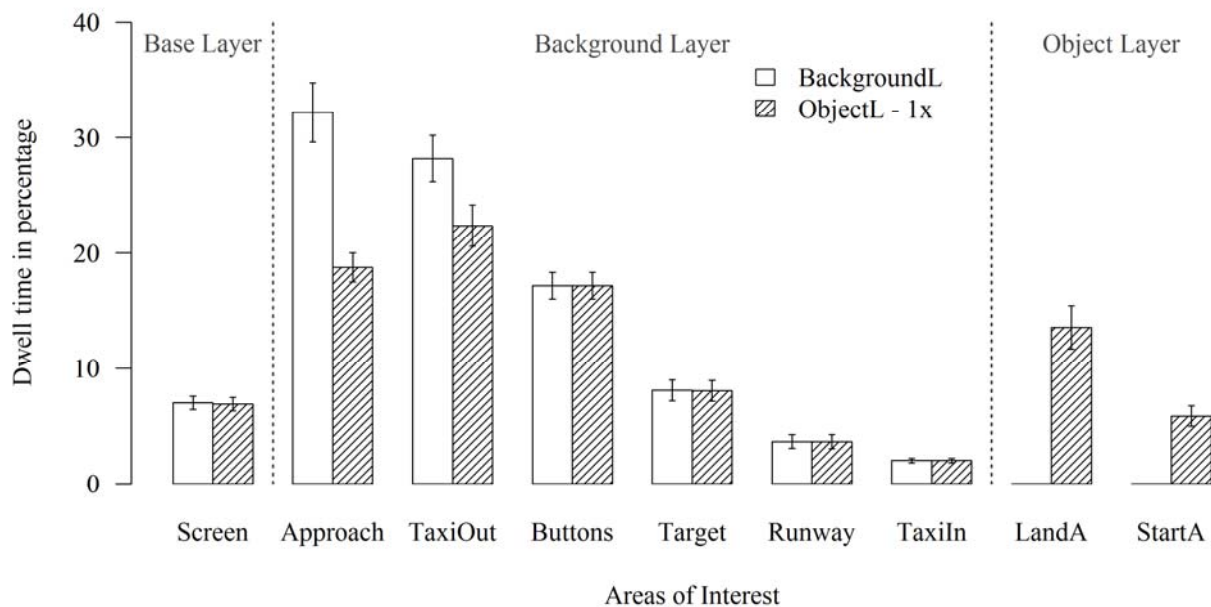
12
 13 *Figure 8 Comparison between dwell time in percentage for 5 participants and static or dynamic*
 14 *AOIs using BeGaze and IGDAI*

15 The second analysis concentrated on the quality of the eye movement data recorded for the
 16 individual participants. First, the amount of valid eye movement per participant (valid eye movement
 17 divided by total number of captured eye movement) was calculated and compared. Eye movement was

1 invalid if the eye tracker could not detect the eyes or the gaze vector was not on the experimental screen.
2 For each participant the validity of the eye movement data reached more than 94% ($M = 97\%$, $SD=1.8$).
3 Second, the quality of the initial calibration was determined by a 9 point verification method. After the
4 calibration of the eye tracker, 91% ($SD=4.5$) of the eye movement data of each participant had an error of
5 1.76° around the actual reference points. Third, because the experimental design did not allow checking
6 the degree of accuracy during the experiment, the alternative analysis proposed by IGDAI is to use a self-
7 selected scale factor (2x, 3x, and 4x) to increase the dynamic AOIs. This leads to an increase in eye data
8 assigned to the dynamic AOIs in relation to the original size (1x). If the increase shows outliers, these
9 participant should be checked individually for calibration errors. Figure 9 shows the increase between 1x
10 and the scale factors in boxplots to identify outliers. An individual examination of both outliers (VP4 and
11 VP18) showed that their percentage of eye movement on dynamic AOIs did not increase at one particular
12 factor, but was above average even for 1x. The analysis with self-selected factors could show that the
13 increase of assigned eye data is similar for all participants, and therefore all participants are comparable
14 on the following analysis.



- 1
- 2 *Figure 9 Increase of assigned eye data to dynamic AOIs in relation to the scale factor 1x for*
- 3 *three self-selected factors.*
- 4 The third analysis demonstrates the impact of the IGDAI layer model to the eye movement
- 5 analysis. As a baseline for this analysis the base and background layer were analysed first and then
- 6 contrasted with the object layer. The dynamic AOIs on the object layer had the original size (1x). Figure
- 7 10 shows the results using IGDAI layers (base and background layer – BackgroundL; BackgroundL plus
- 8 object layer – ObjectL). The bars represent the percentage of eye movement per FAirControl AOI (Figure
- 9 7). The bars for each condition summarize to 100 percent. By using only the BackgroundL the data is
- 10 divided into seven AOIs. When extending the analysis with the object layer the data is divided into nine
- 11 AOIs.



1
2 *Figure 10 Mean plot for percentage of dwell times with standard error (n= 20) separated by*
3 *application of different layers.*

4 Figure 10 shows that the base layer always contains less than 7 percent of the eye movement,
5 independent from the analysis. BackgroundL contains 91.2 percent (Runway + Target + TaxiIn + Buttons
6 = 30.8%; Approach + TaxiOut = 60.362%) of the eye movement. Including ObjectL into the analysis has
7 a significant decreasing effect to the percentage of eye movement data on Approach and TaxiOut [$t(19) =$
8 $9.833, p < .001$]. No other AOIs in the background layer showed a significant increase or decrease when
9 ObjectL was included.

10 The results of the three analyses show an overall view of the possibilities of structured data
11 analysis that IGDAI facilitates. The analysis of distributed eye movement data was increased in detail
12 through the application of the IGDAI layer model. By the use of the new method of structuring the AOIs
13 in a different layer we were able to see the extended distribution of eye movement for the example data.
14 In relation to only concentrating on static AOIs this leads to a detailed insight.

15 **Conclusion**

1 We introduced the guideline IGDAI, which allows an automated classification of eye movement
2 dependent on static or dynamic AOIs. IGDAI enhances the eye movement analysis by simplifying the
3 integration of capturing dynamic AOIs by a step by step approach of integration. IGDAI is also
4 independent from the used eye tracking technology and the task environment because it structures the
5 flow of information that are necessary for eye movement analysis. It can be applied to simple one screen
6 task environments, but also to complex multiple monitor task environments (e.g. Figure 3) or even aircraft
7 cockpits (Sarter, Mumaw, & Wickens, 2007). The requirements (Figure 1) to configure the task
8 environment build the basis for an efficient capturing of the dynamic information and consequently
9 accelerate the data analysis.

10 An important step to understand how participants solve a task is to identify the information
11 gathering sequence. This sequence has to be as accurate as possible because approximations could lead to
12 misinterpretation. With IGDAI the information gathering sequence can objectively be identified. The
13 verification study demonstrated the proper functioning of IGDAI and its influences to various levels of
14 eye movement analysis. It showed that the actual position and size of a dynamic object allows a more
15 detailed analysis than static AOIs alone.

16 Future IGDAI developments should extend the automatic generation of dynamic AOIs to non-
17 artificial task environments. The generic structure of IGDAI allows the adaptation onto those task
18 environments, such as a view through the windscreen while driving in a real environment. Methods (Hsu,
19 S.-H & Huang, 2001) and tools to support the reconstruction of video movements by post processing
20 already exist and only need small adaptation to fulfil the IGDAI requirements, such as the DynAOI
21 generation tool (Papenmeier & Huff, 2010).

22 The results showed that IGDAI lowers the effort of using static and dynamic AOIs as a method of
23 eye movement analysis. The application of dynamic AOI is often associated with effort that is not
24 justified by the results. IGDAI changes that and might lead to dynamic AOIs as a standard tool for eye
25 movement analysis.

1 **Acknowledgments**

2 We thank Hamilton Gross for his help with the first implementation of the IGDAI during his
3 internship at DLR. The authors also thank all the reviewers for their helpful comments on the manuscript.
4 Correspondence concerning this article should be addressed to M. Friedrich, DLR, Lilienthalplatz 7,
5 38108 Braunschweig, Germany (e-mail: Maik.Friedrich@dlr.de).

References

- 1
2 Akamatsu, M., Okuwa, M., Funakawa, M., Kitajima, M., Onuki, M., Life, Q., & Co, M. P.
3 (2001). Whole-View Driving Simulator For Measuring Driving Behavior In Town
4 Environment. In *Driving simulation conference* (pp. 43–50). Valbonne.
- 5 Bruder, C., Leuchter, S., & Urbas, L. (2003). Konzept für automatische Auswertungen von
6 Blickbewegungen bei dynamischen Informationen. *MMI Interaktiv*, 6(Nr. 6), 25–34.
- 7 Bulling, A., & Gellersen, H. (2010). Toward Mobile Eye-Based Human-Computer Interaction.
8 *Ieee Pervasive Computing*, 9(4), 8–12. doi:10.1109/MPRV.2010.86
- 9 Castelhana, M. S., & Rayner, K. E. (2008). Eye movements during reading, visual search, and
10 scene perception: An overview. In *Cognitive and Cultural Influences on Eye Movements*
11 (Vol. 2175, pp. 3–33).
- 12 Crytek GmbH. (n.d.). CryEngine homepage. Retrieved from <http://cryengine.com/>
- 13 Duchowski, A. T. (2007). *Eye Tracking Methodology: Theory and Practice*. New York:
14 Springer.
- 15 Fehd, H. M., & Seiffert, A. E. (2008). Eye movements during multiple object tracking: Where do
16 participants look? *Cognition*, 108(1), 201–209. doi:10.1016/j.cognition.2007.11.008
- 17 Friedrich, M., Möhlenbrink, C., & Carstengerdes, N. (2012). *SESAR- JU D86 Project 06.08.04 -*
18 *Single Remote TWR Ph1 V2 - Validation Report* (00.01.00 ed.). Brussel.
- 19 Galetzka, M., & Glauner, P. O. (2012). A correct even-odd algorithm for the point-in-polygon
20 {(PIP)} problem for complex polygons. *CoRR*, abs/1207.3. Retrieved from
21 <http://arxiv.org/abs/1207.3502>
- 22 Granka, L. A., Joachims, T., & Gay, G. (2004). Eye-tracking analysis of user behavior in WWW
23 search. *Proceedings of the 27th Annual International Conference on Research and*
24 *Development in Information Retrieval SIGIR 04*, 5, 478. doi:10.1145/1008992.1009079
- 25 Gross, H., Friedrich, M. B., & Möhlenbrink, C. (2010). Development of a process for the
26 evaluation of eye-fixation data using dynamic areas of interest. *HMI Laboratory Report*.
27 Braunschweig: DLR. doi:DLR ID IB-112-2009/42
- 28 Henderson, J. M., Weeks, P. A. J., & Hollingworth, A. (1999). The effects of semantic
29 consistency on eye movements during complex scene viewing. *Journal of Experimental*
30 *Psychology Human Perception and Performance*, 25(1), 210–228. doi:10.1037/0096-
31 1523.25.1.210
- 32 Hormann, K., & Agathos, A. (2001). The point in polygon problem for arbitrary polygons.
33 *Computational Geometry: Theory and Applications*, 20(3), 131–144. doi:10.1016/S0925-
34 7721(01)00012-8
- 35 Hornof, A. J., & Halverson, T. (2002). Cleaning up systematic error in eye-tracking data by
36 using required fixation locations. *Behavior Research Methods, Instruments, & Computers :*
37 *A Journal of the Psychonomic Society, Inc*, 34(4), 592–604. doi:10.3758/BF03195487
- 38 Hsu, S.-H., & Huang, C.-L. (2001). Road sign detection and recognition using matching pursuit

- 1 method. *Image and Vision Computing*, 19(3), 119–129.
- 2 Huff, M., Papenmeier, F., Jahn, G., & Hesse, F. W. (2010). Eye movements across viewpoint
3 changes in multiple object tracking. *Visual Cognition*, 18(9), 1368–1391.
4 doi:10.1080/13506285.2010.495878
- 5 Josephson, S., & Holmes, M. E. (2006). Clutter or content? How on-screen enhancements affect
6 how TV viewers scan and what they learn. *Main*, 38(8), 155–162.
7 doi:10.1145/1117309.1117361
- 8 Land, M. F., & McLeod, P. (2000). From eye movements to actions: how batsmen hit the ball.
9 *Nature Neuroscience*, 3(12), 1340–1345.
- 10 Landry, S. J., Sheridan, T. B., & Yufik, Y. M. (2001). A methodology for studying cognitive
11 groupings in a target-tracking task. *Intelligent Transportation Systems, IEEE Transactions*
12 *on*, 2(2), 92–100. doi:10.1109/6979.928720
- 13 McCarthy, J. D., Sasse, M. A., & Riegelsberger, J. (2004). Could I have the menu please? An
14 eye tracking study of design conventions. *People and Computer*, 17(1), 401–414.
- 15 Möhlenbrink, C., Oberheid, H., & Werther, B. (2008). A Model Based Approach to Cognitive
16 Work Analysis and Work Process Design in Air Traffic Control. *Annual Meeting of the*
17 *Human Factors and Ergonomics Society Europe Chapter*. Braunschweig. Retrieved from
18 <http://elib.dlr.de/52438>
- 19 Möhlenbrink, C., Werther, B., & Rudolph, M. (2007). FAirControl: Ein Werkzeug zur
20 Modellierung und Analyse von Verhaltensweisen am Lotsenarbeitsplatz. VDI Verlag.
21 Retrieved from <http://elib.dlr.de/52526/>
- 22 Nielsen, J. (2006). F-Shaped Pattern For Reading Web Content. *Search*. Nielsen Norman Group.
- 23 Palinko, O., Kun, A. L., Shyrovkov, A., & Heeman, P. (2010). Estimating cognitive load using
24 remote eye tracking in a driving simulator. *Proceedings of the 2010 Symposium on*
25 *EyeTracking Research Applications ETRA 10*, 141. doi:10.1145/1743666.1743701
- 26 Papenmeier, F., & Huff, M. (2010). DynAOI: A tool for matching eye-movement data with
27 dynamic areas of interest in animations and movies. *Behavior Research Methods*, 41(1),
28 179–187. doi:10.3758/BRM.42.1.179
- 29 Rayner, K. (1995). Eye movements and cognitive processes in reading, visual search, and scene
30 perception. *Studies in Visual Information Processing*, 6(Elsevier Science B.V.), 3–22.
31 doi:10.1016/S0926-907X(05)80003-0
- 32 Rayner, K., & Pollatsek, A. (1992). Eye movements and scene perception. *Canadian Journal Of*
33 *Psychology*, 46, 342–376.
- 34 Reimer, B., & Sodhi, M. (2006). Detecting eye movements in dynamic environments. *Behavior*
35 *Research Methods*, 38(4), 667–682. doi:10.3758/BF03193900
- 36 Salvucci, D. D., & Goldenberg, J. H. (2000). Identifying Fixations and Saccades in Eye-Tracking
37 Protocols. In *Eye Tracking Research and Applications Symposium* (pp. 71–78). New York:
38 ACM Press.

- 1 Sarter, N. B., Mumaw, R. J., & Wickens, C. D. (2007). Pilots' monitoring strategies and
2 performance on automated flight decks: an empirical study combining behavioral and eye-
3 tracking data. *Human Factors*, 49(3), 347–357. doi:10.1518/001872007X196685
- 4 Seifert, K., Rötting, M., & Jung, R. (2001). Registrierung von Blickbewegungen im
5 Kraftfahrzeug. In T. Jürgensohn & K.-P. Timpe (Eds.), *Kraftfahrzeugführung SE - 14* (pp.
6 207–228). Springer Berlin Heidelberg. doi:10.1007/978-3-642-56721-6_14
- 7 Sennersten, C. (2004). *Eye movements in an Action Game Tutorial*. Lund University, Sweden.
- 8 Sennersten, C., Alfredson, J., Castor, M., Hedström, J., Lindahl, B., Lindley, C., & Svensson, E.
9 (2007). *Verification of an Experimental Platform Integrating a Tobii Eyetracking System*
10 *with the Hifi Game Engine*. Retrieved from <http://www.foi.se/en>
- 11 Taylor, G. (1994). POINT IN POLYGON TEST. *Survey Review*.
12 doi:10.1179/003962694791964997
- 13 Zelinsky, G. J., & Neider, M. B. (2008). An eye movement analysis of multiple object tracking
14 in a realistic environment. *Visual Cognition*, 16(5), 553–566.
15 doi:10.1080/13506280802000752
- 16 Zhang, Y., & Hornof, A. J. (2011). Mode-of-disparities error correction of eye-tracking data.
17 *Behavior Research Methods*, 43(3), 834–842. doi:10.3758/s13428-011-0073-0
- 18