

# Exploratory Study with Eye Tracking Devices to Build Interactive Systems for Air Traffic Controllers

Michael Traouré, Christophe Hurter

► **To cite this version:**

Michael Traouré, Christophe Hurter. Exploratory Study with Eye Tracking Devices to Build Interactive Systems for Air Traffic Controllers . HCI-Aero 2016, International Conference on Human-Computer Interaction in Aerospace, Sep 2016, Paris, France. <hal-01366998>

**HAL Id: hal-01366998**

**<https://hal-enac.archives-ouvertes.fr/hal-01366998>**

Submitted on 14 Feb 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Exploratory study with eye tracking devices to build interactive systems for Air Traffic Controllers

**Michael Traoré**

LII - ENAC  
7 Avenue Edouard Belin  
31055, Toulouse, France  
traore.s.michael@gmail.com

**Christophe Hurter**

LII - ENAC  
7 Avenue Edouard Belin  
31055, Toulouse, France  
Christophe.hurter@gmail.com

## ABSTRACT

While the mouse is the main input device for interacting with different screens, many alternatives do exist. In this article, we report our exploratory study with the usage of eyes as a new input device for Air Traffic Control systems. Our investigations, based on a user-centered design, include a study of the activity, a classification of interaction techniques based on eye tracking systems, and finally a working prototype with the evaluations of the developed interaction techniques. Our goal is to investigate gaze usages as a means of interaction, and give recommendations for future development of Air Traffic Control systems.

## Keywords

Ethnographic studies, New technologies, Multi-modal interaction

## INTRODUCTION

The activity of air traffic control is complex. Operators use many interactions techniques with dedicated IT systems to ensure the traffic flow with respect of security and optimization concerns [].

Air traffic activities is constantly evolving. Air traffic controllers, responsible for ensuring the safety and the fluidity of air traffic, have to deal with more and more information that could cause a significant increase in their workload. However, whether in control tower, approach or even in En-Route control center, air traffic controllers have many tools in addition to the radar screen [12, 16, 17]. Their workspace is thus the scene of an accumulation of interaction devices (mainly mice and touchscreens).

In this paper, we study the possibilities offered by an alternative interaction devices: eye tracking systems. These devices are usually used in the field of air traffic control as a means of gaze analysis and not as a modality of interaction. Due to their recent democratization and their improving performances, their potential usages have changed. Eye tracker device, as a data acquisition device is able to determine the location of the eye movements of an

individual. It can also support the activity of air traffic controller by providing new interaction techniques based on user gaze. The idea however is not to replace existing pointing devices (e.g. touch or mouse), but to leverage Air Traffic Controller activity by the mean of alternative solutions.

There are two main types of eye tracker: the head mounted, and table mounted eye tracker. The first has the advantage of being more accurate because it is fixed on the head of the user with the camera very close to the eyes. But it is expensive and invasive. Meanwhile, the table mounted eye have the advantage of being cheaper (only 90 € for the model used for this study), but it faces many problems with calibration and accuracy issues. Eye tracker systems are obviously not perfect tools and cannot, for now, claim to fully replace the mouse device. However, they may be used in specific cases where the speed of movement of the pointing area would be appreciated [5, 25].

In the first part of this paper, we analyze the activity of Air Traffic Control, and then we present a structured state of the art that allowed us to identify effective ways to instrument ATC activities. Finally, we detail the interactions we have designed and their preliminary assessments.

## METHODOLOGY

We studied ATC activity using the user-centered design methods [3]: contextual observations, production of work scenarios, brainstorming, prototyping and validation.

### Contextual observations

Seminal previous works provide a description of the activity of air traffic control: paper strip usages, augmented environment (Hurter et al. [12]), Air Traffic Controller externalization (Letondal et al. [16]). To extend these studies with eye tracker systems, we conducted a total of 7 observation and interview sessions at ENAC (National School of Civil Aviation) simulators with 8 controllers having a wide range of career (tower, en-Route, small air field) and experience experiences (students, just graduated Air Traffic Controller, Controllers with more than 20 year of practice). During 15 hours of observations, our goal was to understand their tasks and the types of interactions they performed in their day work. We have defined three user profiles: controller students, instructors, and air traffic controllers in office. Furthermore, air traffic controls carry

out they work within 3 main contexts: tower, approach and en-route. In the context of the control tower, controllers are responsible for the aircraft management in the airport vicinity (landing, takeoff, taxiing). In the approach context, controllers have to coordinate aircraft in nearby lowland airports areas, few minutes after takeoff or before their landings. Finally, in the en-route context, controllers deal with aircraft in the higher areas (where the planes are mostly cruising).

In the following, we briefly describe the controller tasks within the three previously defined contexts. The activity of en-route and approach controllers [16] is mainly to maintain a safe distance between aircraft, and to guide them with traffic fluidity concerns. For this, the airspace is divided into sectors, each sector being the responsibility of one controller pair. When a flight goes through an area, controllers guide the pilot by giving him or her orders (clearances) for heading, speed or altitude, until the flight reaches an adjacency sector where other controllers will be in charge of this aircraft. In a typical environment, two controllers are seated in front of a control position, specially designed to support their job. A traditional position includes a set of main subsystems: two radar screens (one for each controller), paper strips shared by both controllers (for systems using the strips) displayed on a horizontal table.

### Limiting Factors

With current systems, controllers are able to perform many different types of interaction. This allows in particular to obtain aircraft details, to setup the radar configuration or to share information with other controllers. During the observation sessions, we extracted some controllers frequent interactions with existing systems:

- *pan and zoom to set the radar view configuration,*
- *entering a clearance,*
- *distance computation between planes using the alidade,*
- *display of the separation distance between aircraft,*
- *display of velocity speed vectors (position of the aircraft in 3, 6, or 9 minutes).*

We have identified a number of limiting factors for these interactions. Although navigation is available in the radar image (pan and zoom), it still does not allow to see the calling planes located outside of the displayed area. This view configuration is done by clicking on the buttons located in a context menu. Other features are accessible via a toolbar or a shortcut menu. This is for example the case of the alidade (measuring the distance between two aircraft), which is performed by selecting it on the tool bar on the left of the radar display (Figure 1) and then plotting on the map the line representing the distance to assess the distance using a drag with the left mouse button.

In conclusion, many interaction requires the mouse device with numerous manipulations (reach the mouse, find the menu location, select the appropriated item, perform selections and mouse drags...). In our study, we chose to

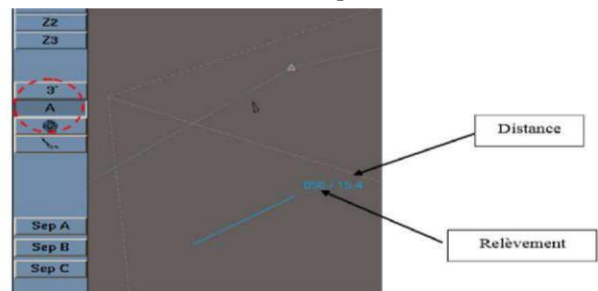
investigate possible gaze based integrations to simplify existing interactions.

### Scenarios

Following our observing sessions, and our interviews, we transcribed our notes, and extracted frequent scenarios. For each of them, we added one positive and negative key points. As such, we defined nine scenarios where the use of an eye tracker could improve Air Traffic Controller activity:

- *give clearance to an aircraft,*
- *indicate a dense traffic area,*
- *pass a strip to a colleague,*
- *deal with the first call of an aircraft,*
- *sequence aircraft (i.e. landing sequence),*
- *get information on a flight,*
- *find aircraft location on the radar screen from its strip.*

Using these working scenarios, brainstorming sessions were carried out with five experienced air traffic controllers and one researcher in HCI (Human Computer Interaction).



**Figure 1. Using the Alidade tool "A" in order to assess the distance and the heading between two points**

### NEEDS AND REQUIREMENTS

In situ observations, and the brainstorming session allowed us to synthesize needs (N) and requirements (R) to improve air traffic controller activities with an eye tracker.

#### N1: Get Information Related To an Aircraft

In all observed scenarios, controllers have to know information on a given flight. This information is usually related to its heading, flight level, speed, or distance between it and another aircraft. In current systems, this information is available on the radar screen (label close to the aircraft location), in context menus, or via the toolbar (at the bottom of the radar screen).

#### N2: Easily Navigate In Radar Image

Air traffic controllers sometimes deal with planes that are not yet visible on their radar screen depending on the view setting (pan and zoom). They have developed the reflex to look at the screen of their side colleague who has a wider radar view configuration. Since zooming and panning the image require multiple interactions (context menu or pressing on the buttons) with existing systems, they are not fast to operate.

**N3: Free the Hands**

The controllers work with one paper strip for each aircraft. They usually hold these strips in their hands when they are dealing with the corresponding aircraft. In addition, other tools such as mouse or pen are often in their other hand in order to communicate with colleagues (identify planes, transmit strips).

**N4: Ensure Collaboration between Controllers**

Air traffic controllers work together in front of their workstation. The strips move between controllers; they can be analyzed and annotated by each of them.

**R1: Respect the Work Methodology**

In the field of air traffic control, any equipment or methodology change have a very high cost in terms of

development and verification to ensure the safety requirements. In general, users are also very reluctant to any change that creates a learning phase which can be detrimental (increased cognitive workload, and new identified and unidentified risk factors for the activity). Using a new device must not alter the controller's working methodology, but simplify it.

**STATE OF THE ART**

During our study, we collected and structured many interaction techniques based on user gaze. Our classification is focused on two main areas: the use of eye tracking alone, and its combination with other devices (Table 1).

Interaction Modality		Tasks							
		Target selection	Pan	Zoom	Scroll	Navigation through a menu	writing <sup>new</sup>	Interaction <sup>new</sup>	
Mon 0	Dwell Time	[1, 7, 15, 23, 26, 32, 33]					[33]	[18, 19, 36]	[2]
	Gaze gestures	[7, 15]							
	Eye Blinking					[4]		[20]	
Multi	Mouse	Magic Pointing	[6,8]						
		Classic mouse	[32]	[27]	[27]	[30]	[34]		[30]
		Ninja cursors	[25]						
	Keyboard	[1,33]	[30]	[30]		[33]	[14]		
	Gestures	Feet			[10]				
		Facial muscles						[23]	
	Smartphones / digital tablet	Touch screens	[19]	[24, 27, 28, 29, 30]	[24, 27, 28]	[29, 30, 31]	[25]	[24]	
	Sensors		[27]	[27,30]					[30]
	Voice	[23]							[22]

**Table 1. Classification of interaction techniques using eye tracking**

### **The Eye Tracker and its Limitations**

There are two types of eye tracker [20] the wearable one (mounted on the head of the user, called Head Mounted Eye Tracker: HMET) and the table mounted eye tracker (located on a table or one a screen). Head mounted eye tracker has two cameras, one that observes the user's eyes and the other the environment of the user. This system always sees the user eyes even if the he or she turns the head. Unlike the head mounted eye tracker, the table mounted eye tracker possess one or more cameras placed in a fixed location close to the screen. This setting can restrict the user with limited head movements. However, some table mounted eye tracker [6, 24] are equipped with systems capable of detecting head movements. Table mounted eye trackers are more used than head mounted eye trackers since they are considered more natural and suitable for daily usage [24]. Nevertheless, this type of eye tracker is facing a recurring limitation related to the aperture angle with which the user can observe a scene. In some cases, the corners of the screen are difficult to observe [4]. Therefore, despite high prices, head mounted eye trackers remain the best solution, especially in a changing and / or confined environment. Baldauf et al. [2] suggest, for example, to use a bicycle helmet on which is fixed an eye tracker to analyze the activity of a cyclist and allow it to view additional information according to its surroundings (augmented reality). However, head mounted eye trackers suffer from the fact that it is intrusive and can hinder user field of view [14], which could be also a limitation in an ATC environment.

### **Usage of a solely Eye Tracker**

In many cases, eye tracker, whether worn or remote, is sufficient to allow interaction between a user and a system. Thus in this section, will be presented different modalities of interaction using the eye tracker as the only input device.

#### *The Dwell Time Method*

The dwell time method consists in selecting a target on a screen by only fixing the gaze for a certain period of time [1, 18, 26, 33]. One can have either a single feedback when selecting the target, or two distinct stages [19]: at first, one can highlight an item and then later select it (a second feedback can then be used). If the user moves his or her eyes before the selection period, then the target is not selected. Some users do not like the presence of a feedback when hovering over a target but only in the selection of the latter [18]. For this kind of task, the fixation time is the most common method. Despite its benefits, several limitations appear. First, the size and spacing of targets strongly influence user performance. Based on Fitts' law [9], Roel Vertegaal [32] shows that it is difficult to make a selection with small size targets. In the air traffic control radar image, the main targets (aircraft) are large enough to allow the use period.

#### *Gaze gestures*

The gaze gestures are sometimes a good alternative for target selection. According to researchers at the University of San Marcos [15], this selection method is faster than the dwell time since the saccadic eye movements are more natural to the user. These researchers offer two selection techniques. In the first one, named Saccade Offset Selection, selection occurs when the gaze lands on the target while with the second technique, called Instantaneous Saccade Selection, the system predicts where the gaze will land. It would be interesting to test this method as an alternative to the dwell time method in the ATC domain but it requires complex settings to efficiently tune this algorithm.

### **The Eye Tracker and the Mouse**

The intention of a user can often be difficult to interpret (i.e. "Midas Touch effect" described by Jacob [13]). Jacob says that the user can watch part of the interface without wanting to activate it immediately. One solution to this problem is to use eye movements with other modalities of interaction. The mouse was the first device coupled to the gaze. Many studies [6, 8, 35] have chosen to promote the association of eye tracker and mouse in order to benefit from the advantages of each: the speed of the eyes and the accuracy of the hand.

The Magic Pointing [35] is one of the first techniques using the combination of eye and mouse. The idea is to use the eyes to roughly move the cursor on the target and then use the mouse to accurately define the selection. However, certain limitations have been highlighted [6, 8], including the fact that mouse movements performed by the user can change the cursor position determined by the gaze.

Unlike Zhao et al. [35] which deal with only one button on the screen, Yamato [34] proposes to combine the eye tracker and the mouse to control several ones. The operation is done in two steps: move the view of the selected button and confirm the selection by a mouse click. The results show that the combination of the eye tracker and mouse is much more effective than single mouse for targets exceeding 3 cm<sup>2</sup>. Otherwise, it is the only mouse that wins. Ninja Cursors [25] is a technique where several pointers are displayed on a screen with one of them chosen by the gaze as the active pointer. Handling multi-cursor with the gaze surpasses the use of a single cursor if the targets are of a reasonable size and they are distant from the current position. Combining the mouse and the gaze can be interesting in an ATC system since controllers already use the mouse device as the main interaction method.

### **The eye tracker and devices with buttons**

The devices with buttons such as the keys of a keyboard induce deliberate movements without ambiguity. This feature helps counter the Midas Touch Effect [13]. Ware

[33] proposed the joint use of the dwell time and buttons to make the selection more efficient at the cost of a learning phase to synchronize user eyes movements with the hand. Such a solution is possible in the ATC environment since such device does not occupy too much space.

### The Eye tracker and the Gestures

The combination of an eye tracker and actions performed with the feet were previously studied [10]. For example, to navigate through a map, the user looks at the place where he or she wants to zoom in, then operates the pedal forward or backward to control the zoom speed. The head movements can also be used as interaction means for selecting targets as they involve many muscles whose movements can be recognized by electromyography (EMG) [21]. This solution seems viable because air traffic controllers are already using a foot switch.

### The Eye Tracker and Touch Devices

Today, mobile devices such as smartphones and tablets are very powerful and have many sensors (proximity, light, gyroscope, accelerometer, GPS ...). Combined with these sensors, eye trackers can offer many opportunities for interaction [27, 30, 31]. The combination of sight and touch also solves the Midas Touch Effect [13] while avoiding unintended selections [28]. Stellmach and Dachsel show that zoom in and out can also be performed by a hand gesture [27]. In addition, tactile devices also permit the construction of more complex interactions such as Drag'n Drop or Cut & Paste [29]. This solution is possible in ATC systems without paper strips with a tactile device in front of the controller.

### The Eye tracker and the Voice

To overcome the eye tracker precision issue, it is possible to associate gaze and voice [22, 23, 54]. The use of these two types of entries completely frees the user hands and allows him or her to communicate without constraint.

### SYSTEM DESCRIPTION

After the brainstorming, we built prototypes that were evaluated during a design walkthrough session with 3 confirmed air traffic controllers. We did not reduce the solution space in order to get a maximum of ideas from air traffic controllers. Finally, we made a prototype covering all the prototyped interactions.

This prototype was developed in C# which allows to deal easily with the libraries of our eye tracker (The EyeTribe). We used the software bus IVY [16] for the communication between the different modules (Figure 2).

### I1: Selection Methods

The selection of an aircraft is the first task an air traffic controller shall achieve before interacting with it. Thus, it is mainly N1 and N4 needs. To do this, different techniques of interaction are possible and are shown in the following.

### I1-1: The mouse (Existing Technic)

We kept the functioning of this interaction which is already implemented in existing systems, in order to compare the eye tracker and the single mouse. (Figure 5)

### I1-2: The Dwell Time

The Dwell Time is a technique that allows to change the state of a target (aircraft in this case) when looked for a certain period. (Figure 3)

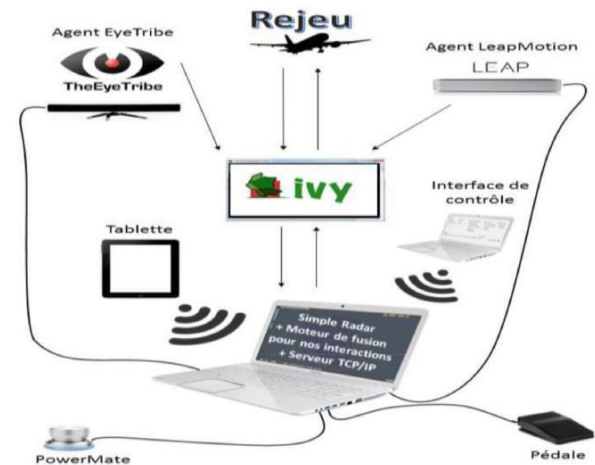


Figure 2. The architecture of the prototype



Figure 3. Selection with dwell time



Figure 4. The different states of an aircraft

Our prototype has two buttons, one for the validation and the other one to cancel a previous validation. The appearance and size of these buttons have been modified following the remarks during the design walkthrough sessions. The buttons are positioned in areas normally without traffic (Figure 3). The user must look at a plane: it will switch to blue after a first timer (dwell time, 200ms). Then the user simply look at the "validate" button for a certain period to select the aircraft (it changes to yellow). To cancel the selection of this aircraft, the user must again

look at it, then it turns red, and if the user looks at the cancel button for a certain period, the airplane is deselected (Figure 4).

#### *11-3: Buttons*

During the brainstorming session, it emerged that the best option to select an aircraft is to use an external device (button, keyboard or pedal ...). That's why we decided to include this interaction in the final prototype.

#### *11-4: Pedal*

This interaction is almost identical to the previous one, except that it offers the user a button that plays both the role of the "validate" and the "cancel".

### **12: Function Selection**

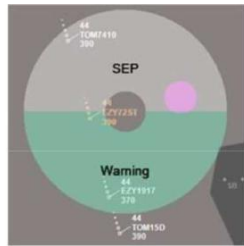
The methods of the function selection we implemented, respond to the needs to obtain information on a plane, and free the hands (N1 and N3).

#### *12-1: Pulldown Menu*

This interaction is similar to the one existing in the current systems. With a click on an aircraft, users have access to a context menu offering several features. This interaction has been developed for the purpose of comparison with the new methods of selection that we propose (Figure 5).



**Figure 5. Pulldown menu**



**Figure 6. Pie Menu**

#### *12-2: Pie Menu and Dwell Time*

The advantage of the pie menu is that it appears at the exact location where the user has selected the second aircraft. Hence, the user can directly select one of the menu items. However, following the remarks of some users, we have increased the delay to 500ms before opening the pie menu. Moreover, in order to respond to the need to keep all the parts of the radar image visible, we opted for transparent colors (Figure 6).

#### *12-3: Pie Menu and Gaze Gestures*

In this interaction, the user must move his or her gaze from one of the menu items to the outside of the pie menu. This gaze gesture launches the warning function on the selected planes.

### **13: Zoom Methods**

These interactions respond to the need to freely navigate through the radar image.

#### *13-1: Pointer centered Scroll*

This interaction exists in some ATC interfaces (mainly interfaces without strip called "stripless"). The user acts on

the mouse wheel to zoom in or out. The zoom center is then located at the mouse pointer. Also, it will help us to compare the effectiveness of our new interactions.

#### *13-2: Gaze centered Scroll*

As before, this zoom technique is based on the mouse, but this zoom is not centered on the mouse pointer anymore, but rather on the user's gaze location on the screen. In both techniques, the zoom returns to its original level by clicking with the mouse wheel.

#### *13-3: Touchpad*

With the usual gestures made on a touch pad, the controllers can perform forward zooming (Pinch open) and backward zooming (Pinch close) centered on his / her gaze.

#### *13-4: Leap Motion*

The Leap Motion is a device that detects the movements and aspects of the user's hands. We therefore used it to achieve an interaction that is to approach or pull away the palm of the hand to zoom in or out. When removing the hand from the Leap Motion's detection area, the zoom will return to its original level.

### **14: Pan Methods**

Sometimes it is necessary for the controller to observe the planes that are not necessarily displayed on the controller screen. To do this, the user must perform lateral movements (i.e. Pan) of the radar screen. These interactions respond to the need to navigate through the radar image (N2).

#### *14-1: The Mouse*

A first technique, which often exists in some air traffic control systems, is to make lateral movements with the mouse. The user clicks on the map then performs the movements of "drag" to move it.

#### *14-2: Touchpad*

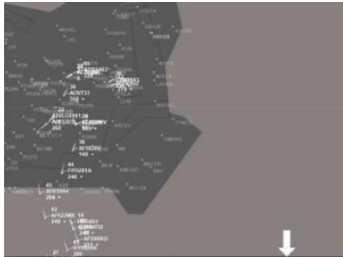
This technique does not use the gaze but given the presence of the touchpad in the final prototype, we wanted to know if the users appreciated the fact to navigate through the map with their fingers (as we can do it in Google Map Application for Smartphone for example).

#### *14-3: SmartPan*

Our goal was to use the eye tracker to observe directly a calling aircraft that would be out of the controller's screen. For this, we have designed a system with voice recognition to detect the corresponding call sign from calling aircraft. If the calling plane is not within the area displayed on the screen, the system automatically displays an arrow indicating the direction toward this aircraft. The user then has to look at the arrow (which gets yellow thanks to the eye tracker), and a pan is automatically processed toward the calling aircraft. This aircraft will be displayed in a different color for easy identification (Figure 7). Once the information is taken by the controller, he or she just need to look in the opposite direction of the arrow and the Pan



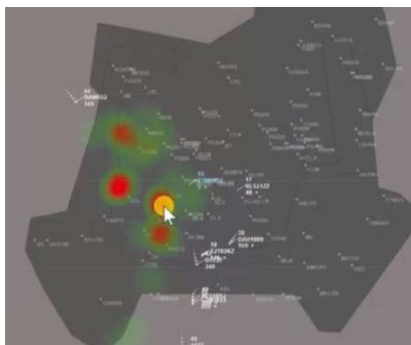
returns to its initial value. In user testing, we simulate a calling plane with the control software.



**Figure 7. SmartPan: the arrow indicates a calling aircraft outside the radar image.**

### 15: Heat map

The heat map is already widely used in other areas and allows ATC to analyze the gaze of a user by recording the history of the recorded gaze locations. These locations are then transcribed in the form of a color gradient indicating regions of interest on which the user focused on. When controllers interact with each other to discuss a situation, they often have to share information from the radar screen and to highlight aircraft to clarify the situation. The idea here is to use the heat map as a support to discussions and to solve conflicting aircraft situations (N4). When one of the two controllers do not immediately identify the area his or her colleague is talking about, one can press a button or a key to display a heat map tracing the gaze history (about 20 seconds of past gazes) of the areas observed (Figure 8). The controllers can then have an overview of the specific situation and find a solution to address it.



**Figure 8. Heat map showing the most viewed areas + improved magic pointer display the mouse cursor inside the most viewed area**

### 16: Improved Magic Pointing

This interaction takes its root from the heat map computation. Its goal is to display the mouse pointer at a suitable location when the controller has to use it and thus avoid too long mouse drag to reach the suitable pointing target. To do so, just before the controller touches the mouse, the algorithm computes the heat map. When the user moves the mouse pointer, the system will display the mouse pointer location at denser gaze location (Figure 8).

## EVALUATION OF THE PROTOTYPE

The evaluation was made to cover the 3 different user profiles defined in this study: controller students, instructors, and air traffic controllers in office. We conducted the tests with a sample of 15 people with 5 people per profile. The eye tracker is a new device for these people, and it has been necessary to conduct a familiarization phase with them. This phase consisted in introducing the device, then calibrate it according to the user profile. Then, during the training phase, we proposed a game having no connection with the ATC in order to train to use the different developed interaction techniques. In this game, different geometrical shapes followed random trajectories on the screen. The task was to select a moving star and assign it with a green color. In addition to this work, we proposed to the user to zoom and move the picture. These tasks allowed to use the different types of interaction we developed. After this training, we proceed to the evaluation of each of the developed techniques within three tasks:

The first task was to change the state of two aircraft in a conflict situation into "Warning". This task involved the use of a selection method and the warning function. The second task was to give a clearance (change of altitude or speed) to an aircraft located in a dense traffic area. This task required the use of a zoom technique. The third task was to give a clearance to an aircraft located outside the area displayed on the screen. We had the opportunity to test a method of pan. And finally, the last task was to show the recorded gaze location on the screen via a heat map.

Each of these tasks has been performed with each of the proposed interactions including the techniques used in the current ATC systems. At the end of each task, the user had to give their opinions by filling a questionnaire asking a qualitative evaluation of each of the interaction techniques.

The different remarks from these questionnaires have enabled us to establish a table of user preferences (Table 2). The heat map was generally appreciated, especially by the instructors. Indeed, they found that it would be very interesting during the training courses of controllers in the sense that this heat map would help them observe and correct the visual path of apprentice. It is also a good help for the other controllers who will be able to know if their colleague is aware or not of a conflicting situation. The zoom centered method on the gaze using the mouse wheel was the method that received the most positive comments from users.

Following user tests, many interactions have been improved and new ones were created to best meet the needs of air traffic controllers. To select an aircraft with only the eyes, we designed another solution. When the plane is gazed, two small labels appear on its side. The



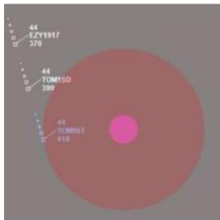
controller have just to look at the action he or she wants to perform (Figure 9).

Regarding the selection of functions, the pie menu was not easy to master, either with the selection of items via the dwell time or worse through gaze gestures. In stressful situations, involuntary actions could take place. To address this, we proposed a validation of pie menu items via a button or keyboard. We also added a visual feedback on the gazed item in order to meet users' needs during the first test session.

To compensate the loss of calibrations noticed during the tests, we implemented an algorithm called Bubble Cursor [11]. This algorithm consisting in computing the nearest neighbor of the user gaze location. Thus, even if the eye is not perfectly on one of the aircraft in the control area, the nearest aircraft is still selected (Figure 10).



**Figure 9. Aircraft selection using dwell time. The controllers have just to look at the action he or she wants to perform**



**Figure 10. Bubble cursor selecting the closest aircraft**

## DISCUSSION

### Performance

The eye tracker used for this study was a low cost eye tracker (€90). This involved limitations in terms of accuracy and range, which hindered the designed interactions.

It turned out during our study that the eye tracker used still lacked more accuracy on wide screens (over 23 inches). It is therefore recommended to use a device with better performance for critical situations. Indeed, the mobility of controllers in their workspace requires the use of an eye tracker that can maintain its current calibration. In fact, calibration is time consuming when redundantly performed. A way around this accuracy problem is to use the bubble cursor method to select the nearest target. Also, showing feedback from the gaze position on the screen

allows users to correct this inaccuracy; but in this case, the interaction becomes unnatural.

### Relevance

The designed interactions allow to perform different types of tasks. Among the developed interactions, the most appreciated selection method was the one using a foot switch. This can be explained by the desire to minimize involuntary actions related to eye movements. In fact, one of the major challenge is to avoid Midas Touch Effect inherent to Eye tracker systems. It is always better to validate a critical action with another device, such as a switch. Thus, eye tracker based interactions, such as the dwell time, does not seem appropriate to critical systems. Regarding navigation methods like the Pan and Zoom, two interactions were mostly appreciated: the gaze centered scroll and the SmartPan. The gaze centered scroll was much appreciated as a technique close to the existing one. The only difference is that the zoom is gaze centered. In contrast, zoom via touchpad or leap motion did not receive as many good score because they imposed to add another input device in an already cluttered workspace. The SmartPan was also well received because it offers an innovative feature. Although it essentially depends on the gaze, this interaction was appreciated because it allows to perform a non-critical task. The different types of proposed menus such as the pie menu with dwell time and the pie menu with gaze gestures, received both positive and negative scores depending on the user. The comments were partly on the selection time that seemed too long or too short for some others. The good compromise would be to make it editable by the users. Furthermore, eye movements did not seem natural to some users. In fact, this technique increases the risk of Midas touch effect. In addition, the heat map was appreciated since it does not change the current working method since the user has the option to display it at will.

## CONCLUSION AND PERSPECTIVES

In this article, we present an exploration of interactions offered by eye tracker systems in air traffic control environments. This study was carried out with close involvement of air traffic controllers during all design phases (in situ observation, brainstorming, design, prototyping, design walkthrough, development, testing). We developed new interaction techniques. The most appreciated techniques were the gaze centered scroll, the SmartPan, the selection combining the gaze and a switch, and the heat map. The 2 main design recommendations emerging from this study are, first, to validate the actions by a third device for critical task; and second, to favor non penalizing interactions in case of poor detection of the eye tracker. As an example we can quote the magic pointing [35] which allows to gain speed without disturbing the current task.

Regarding the perspectives, one can mention the following: the implicit calibration of the eye tracker, and using the eye tracker to train controllers. First, the current calibration phase is time consuming; it would be very interesting to find new faster calibration methods. Then, several instructor controllers have expressed interest for the eye tracker as a means of analyzing the working methods of air controller apprentices. This could in fact facilitate the teaching of good working methods. Finally, we plan to quantitatively assess all the techniques developed to validate their performance and extend the design rules in remote tower environments.

## BIBLIOGRAPHY

- Alonso R., Causse M., Vachon F., Parise R., Dehais F., Terrier P. Evaluation of head-free eye tracking as an input device for air traffic control. *Ergonomics*, vol. 56, no. 2, pp. 246-255, 2013.
- Baldauf M., Fröhlich P., Hutter S. KIBITZER: a wearable system for eye-gaze-based mobile urban exploration. In Proc. *AH '10*, ACM (2010).
- Beaudouin-Lafon M., Mackay W. Prototyping tools and techniques. In *The human-computer interaction handbook*, Julie A. Jacko and Andrew Sears (Eds.). L. Erlbaum Associates Inc., 1006-1031.
- Biswas P., Langdon P. A new interaction technique involving eye gaze tracker and scanning system. In Proc. *ETSA '13*, ACM (2013), 67-70.
- Blanch R., Ortega M. Rake cursor: improving pointing performance with concurrent input channels. In Proc. *CHI '09*, ACM (2009), 1415-1418.
- Drewes H., Schmidt A. The MAGIC Touch: Combining MAGIC-Pointing with a TouchSensitive Mouse. In Proc. *INTERACT '09*.
- Dybdal M., Agustin J., Hansen P. Gaze input for mobile devices by dwell and gestures. In Proc. *ETRA '12*, ACM (2012), 225-228.
- Fares R., Downing D., Komogortsev O. Magic-sense: dynamic cursor sensitivity-based magic pointing. In *CHI EA '12*, ACM (2012), 2489-2494.
- Fitts, Jones, Milton. Eye movements of aircraft pilots during instrument-landing approaches. In *Aeronautical Engineering Review* 9(2), 4950, 24 29.
- Göbel F., Klamka K., Siegel A., Vogt S., Stellmach S., Dachselt R. Gazesupported foot interaction in zoomable information spaces. In *CHI EA '13*, ACM (2013), 3059-3062.
- Grossman T., Balakrishnan R. The bubble cursor: enhancing target acquisition by dynamic resizing of the cursor's activation area. In Proc. *CHI '05*. ACM (2005), 281-290.
- Hurter C., Lesbordes R., Letondal C., Vinot J., Conversy S. Strip-TIC: exploring augmented paper strips for Air Traffic Controllers. In *AVI 2012*, 225-232.
- Jacob R., Karn K. Eye tracking in Human-Computer Interaction and usability research: Ready to deliver the promises. In J. Hyönä, R. Radach, & H. Deubel (Eds.), *The mind's eye: Cognitive and applied aspects of eye movement research*, 2003, pp. 573-605.
- Johansen S., Agustin J., Skovsgaard H., Hansen J., Tall M. Low cost vs. high-end eye tracking for usability testing. In *CHI EA '11*, ACM (2011), 1177-11 82.
- Komogortsev V., Ryu Y., Koh D., Gowda S. Instantaneous saccade driven eye gaze interaction. In Proc. *ACE '09*, ACM (2009), 140-147.
- Letondal C., Hurter C., Lesbordes R., Vinot J., Conversy S. Flights in my hands: coherence concerns in designing StripTIC, a tangible space for air traffic controllers. In Proc. *CHI '13*. ACM (2013), 2175-2184.
- Mackay W., Fayard A., Frobert L., Médini L. Reinventing the familiar: exploring an augmented reality design space for air traffic control. In Proc. *CHI '98*. ACM Press, 558-565.
- Majaranta P., Aula A., Rähä K. Effects of feedback on eye typing with a short dwell time. In Proc. *ETRA '04*, ACM (2004), 139-146.
- Majaranta P., Rähä K. Twenty years of eye typing: systems and design issues. In Proc. *ETRA '02*, ACM (2002), 15-22.
- Mardanbegi D., Hansen D. Mobile gaze-based screen interaction in 3D environments. In Proc. *NGCA '11*, ACM (2011).
- Mateo J., Agustin J., Hansen J. Gaze beats mouse: hands-free selection by combining gaze and emg. In *CHI EA '08*, ACM (2008), 3039-3044.
- Merchant S., Schnell T. Applying eye tracking as an alternative approach for activation of controls and functions in aircraft. In Proc. *DASC*, 2000.
- Miniotas D., Šnakov O., Tugoy I., MacKenzie I. Speech-augmented eye gaze interaction with small closely spaced targets. In Proc. *ETRA '06*, ACM (2006), 67-72.
- Nagamatsu T., Yamamoto M., Hiroshi S. MobiGaze: development of a gaze interface for handheld mobile devices. In *CHI EA '10*, ACM (2010), 3349-3354.
- Rähä K., Šnakov O. Disambiguating ninja cursors with eye gaze. In Proc. *CHI '09*. ACM (2009), 1411-1414.
- Sibert E., Jacob R. Evaluation of eye gaze interaction. In Proc. *CHI '00*, ACM (2000), 281-288.
- Stellmach S., Dachselt R. Investigating gaze-supported multimodal pan and zoom. In Proc. *ETRA '12*, ACM (2012), 357-360.
- Stellmach S., Dachselt R. Look & touch: gaze-supported target acquisition. In Proc. *CHI '12*, ACM (2012), 2981-2990.
- Stellmach S., Dachselt R. Still looking: investigating seamless gaze-supported selection, positioning, and manipulation of distant targets. In Proc. *CHI '13*. ACM (2013), 285-294.
- Stellmach S., Stober S., Nürnberger A., Dachselt R. Designing gazesupported multimodal interactions for the exploration of large image collections. In Proc. *NGCA '11*. ACM (2011).
- Turner J., Alexander J., Bulling A., Schmidt D., Gellersen H. Eye Pull, Eye Push: Moving Objects between Large Screens and Personal Devices with Gaze and Touch
- Vertegaal R. A Fitts Law comparison of eye tracking and manual input in the selection of visual targets. In Proc. *ICMI '08*, ACM (2008), 241-248.
- Ware C., Mikaelian H. An evaluation of an eye tracker as a device for computer input2. In Proc. *CHI '87*, ACM (1986), 183-188.
- Yamato M., Inoue K., Monden A., Torii K., Matsumoto K. Button selection for general GUIs using eye and hand together. In Proc. *AVI '00*, ACM (2000), 270-273.
- Zhai S., Morimoto C., Ihde S. Manual and gaze input cascaded (MAGIC) pointing. In Proc. *CHI '99*, ACM (1999).
- Zhao X., Guestrin E., Sayenko D., Simpson T., Gauthier M., Popovic M. Typing with eye-gaze and tooth-clicks. In Proc. *ET RA '12*.
- Hurter C., Conversy S., Gianazza D., Telea A. Interactive image-based information visualization for aircraft trajectory analysis. In *Transp. Res. Part C Emerg. Technol.*, vol. 47, Part 2, p. 207 227, oct. 2014.