Model of attention for a virtual agent able to interact with a user through body movements

Panagiotis Mavridis

Master research Internship

Research Master Thesis

Model of attention for a virtual agent able to interact with a user through body movements

Author:
Panagiotis Mavridis

Supervisors:
Elisabetta Bevacqua
Pierre de Loor
ENIB-CERV

Abstract

This report presents the work during my internship concerning the creation of a model of attention for a virtual agent capable of interacting with a human through body movements. Attention is defined as the process of filtering on the intelligent entities senses. Depending on the stimuli the attention is characterized as bottom up (external stimuli) or top down. Different models of attention argue on the time a perception filter acts on attended or unattended information or introduce the notion of attentional capacity and argue that the filtering relates to the semantic meaning of the perceived information. Also, specific behaviour expressed by body movements, gestures and gaze is related to attention that is an intrinsic human mechanism. To create believable virtual agents that resemble human, implementation of attention mechanisms is needed. The existing computational models are grouped into methods that perform either visual search to find points of interest or semantic processing or implement overt behaviour or combine methods that are used for expressive embodied conversational agents. Based on the previous work, we propose some ideas about the conception of our model. We think that a combination of bottom up and top down method along with a vision field filtering for the agent that depend on its head orientation in order to show a more believable behaviour. A module of Attention was created as a part of a multi-module architecture virtual agent architecture. The modules communicate through messages. The Attention module consists of a vision field filter, a motion calculation, a state machine, a strategy pattern. The vision field filters whether the user is visible or not to the agent. Then the motion calculation takes into account the visible parts of the user that move in order to calculate a level of attention for the agent. Then depending on the input from the vision filter and the motion calculation a state is calculated and an action (message is sent) is taken.

Contents

List of Figures

1 Introduction

The main goal of this work is to propose a computational model of attention for an autonomous virtual agent capable of real-time dynamic affective bodily communication with a human. We aim at introducing this model in a wider virtual agent architecture which is being implemented within the French project Ingredible (www.ingredible.fr) founded by the ANR (French National Research Agency). We believe that in order to show more natural and human-like behaviour the virtual agent should be endowed with attentional capabilities. In fact human behaviour is strongly connected with attention [Mundy and Newell, 2007].

Randal Hill in his article states the reasons concerning the importance of attention. There is exceed information in the visual field for the human perceptual system to process [Hill, 1999]. For this particular reason the human perception system uses an attention mechanism to ignore the extra information and select only the valuable information [Chun and Wolfe, 2005, Wolfe, 2006]. Similarly, introducing this type of behaviour for an autonomous virtual agent can help to discard less important data, which suits well the computational capacity restrictions when dealing with this data.

According to Peters et al. [Peters et al., 2011], there are at least two major reasons for agents to be attentive to their environment. Firstly, for aesthetic reasons. In order for agents to become more human-like and give the illusion of credibility about their intentions and attention, a set of behaviour should be attached to them. This kind of behaviour is the orientation of their gaze, body language (gestures) and facial expressions that is already intrinsic to human beings. It is also the way we are used to perceive it from other intelligent forms of life. For example, in our everyday life, we are used to look someone that we want to pay attention to while he speaks and we are interested on what he says. At the same time our body direction is towards the speaker and not opposed to him and our facial expressions or head movements, as we nod, show that we are following the train of his expressed thoughts. On the contrary when we are not interested our attention level and thus behaviour shows that we are more distracted or even less interested. So we might look away or even walk way to show we are not interested.

Secondly, there are also functional reasons. Virtual agents, have a certain plan that consists of goals that they want to achieve. In order to do this, they should orient their senses with compliance to these goals. Their internal modes should be updated with information relevant to these goals. For instance one can take as an example an agent that has to find a particularly coloured or shaped object in a set of objects. In order to find the particular color the agent should take into account the color of every object and try match it to the goal color. If found then the goal is achieved and he can move to the next goal.

The next section introduces the notion of attention from an interdisciplinary point of view. Different types of attention stages, models and attentional behaviour are discussed. The third section consists of the state of the art on computational attention models for virtual agents, grouped with respect to the methodology they follow. We will also discuss the advantages and the limits of these models according to our project requirements.

Then a brief project context including technical details is discussed. A next session discusses the general project architecture. Afterwards, we focus on the Attention model architecture that was developed during the internship. The model is not yet evaluated but some preliminary experimentation on the expected values and responses has been done. Some snapshots of the current version of te application, while these lines were written, are presented on the experimentation section. As a conclusion we provide some insight on what could be developed more, our constraints and our

expectations.

2 The interdisciplinary point of view for Attention

2.1 Defining attention

As an intrinsic human capability, attention has been studied in several humanistic domains such as Psychology, Neuroscience and Linguistics [Wolfe, 2006, Chun and Wolfe, 2005, Mundy and Newell, 2007, Coull et al., 2000, Posner, 1980, Posner and Rothbart, 1992, Poggi, 2001, Goertzel et al., 2010, James, 1890, Ratey, 2001, Theeuwes, 1991]. Thus, there are several definitions that apply to this notion from different points of view.

According to psychologist and philosopher William James [James, 1890], attention "...is the taking possession of the mind, in clear and vivid form, of one out of what seem several simultaneously possible objects or trains of thought...".

Later on, John Ratey [Ratey, 2001], attention involves a number of processes including filtering out and balancing multiple perceptions while also attaching emotional significance to them. This definition comparing to William James, who proposes a perceptional process, also clarifies the balancing and the filtering that takes place during the process of attention. To understand attention from this point of view, let us think of an example. We consider someone listening to the music with an mp3 player and walking to the street. He might not be able to listen to other people but he can watch the traffic light when it turns green, in order to pass the zebra crossing and also can balance his walk, in order not to fall. He could be able to do all the above while still listening to the music.

It is also common for experts of attention, to describe it with an analogy. This analogy wants attention to act as a spotlight [Chun and Wolfe, 2005] (as seen in figure 1). The spotlight focuses on a certain visual space the same way attention focuses our awareness on a subset of what's going on in our head or in our environment. This analogy is completing and illustrating very well the two previous definitions. It also illustrates the experience we live while we concentrate on reading a book and we are absorbed by the world of the book. For our eyes there is only the book pages and for our mind there is nothing but the world the book describes that we envision.

Figure 1: The spotlight analogy.

To sum up, attention is defined as the notion that refers to how we process, filtering in or out, specific information present in our environment or mind.

2.2 Classifications of attention

Let's imagine that we are watching TV. Our senses and mind are concentrated on the act of watching the TV show. As defined in the previous sections, attention is used to filter the input from our senses in order to discard any data perceived and not relevant to what we are doing. At the same time, while watching the show there is a possibility that an external stimuli such as a high pitch voice from the neighbours might distract us from the film and gain our attention and curiosity. Also, it is likely that we have left some water to boil and we do not want it to be spilled while we watch TV. Suddenly, we remember that we have to switch it off. This implies that there is not just one form of attention.

Attention can be classified according to different approaches such as stimuli, orientation, or the senses involved.

Attention according to stimuli can be characterized as either bottom up or top down. To be more precise, bottom up attention is driven by an external factor and not relevant to our current context of actions. It is an involuntary process, thus also called passive. For example, an odour that catches someone's attention, a high pitch voice or sound, an interesting car or a beautiful person that passes by the street. It could be an object that drives the attention, or a particular point of interest. Another terminology used for the same notion is "exogenous" [Mayer et al., 2004, Theeuwes, 1991, Coull et al., 2000, Chun and Wolfe, 2005], from Greek exo, meaning "outside". Also, it is considered to be reflexive and automatic and is caused by a sudden change in the context (periphery). This often results in a reflexive reaction. Since exogenous cues are typically presented in the periphery, they are referred to as "peripheral cue". Exogenous orienting can be observed when individuals are aware that the cue will not remain reliable.

On the other hand, top down attention is driven from a center of cognition and is task based. It is the process one makes in order to drive his interest from his thoughts to an external object, space, act or person. It is a voluntary process, thus also called active. Other terminology used for the same notion is endogenous [Mayer et al., 2004, Theeuwes, 1991, Coull et al., 2000, Chun and Wolfe, 2005], from Greek endo, meaning "within" or "internally". Simply stated, endogenous orienting occurs when attention is oriented according to an observer's goals or desires, allowing the focus of attention to be manipulated by the demands of a task. In order to have an effect, endogenous cues must be processed by the observer and acted upon purposefully. These cues are frequently referred to as central cues. This is because they are typically presented at the center of a display, where an observer's eyes are likely to be fixated. Central cues, such as an arrow or digit presented at fixation, tell observers to attend to a specific location.

Attention can also be classified by its orientation as *covert* and *overt*. This categorization has mostly to do with the behaviour one can show to expresses attention [Posner, 1980]. To this extent, overt orientation of attention characterizes someone that uses gaze and thus head orientation to express his attention state. It is the act of selectively attending to an item or location over others by moving the eyes to point in that direction [Posner, 1980]. Overt orienting can be directly observed in the form of eye movements.

On the other hand, covert orienting of attention does not express overtly the shift of attention. It is the act of mentally changing one's focus without altering gaze direction [Posner, 1980]. It is simply a non-observable or perceivable change in the internal attention state. It has the potential

to affect the output of perceptual processes by governing attention to particular items or locations, but on the same time separates it from the information processed by the senses. It seems that visual covert attention is a mechanism to quickly scan the field of view for interesting locations and thus conserving sensing resources at the same time. To make things clear we can consider the following example. We can imagine someone that is aware of the presence of someone else and and has his attention on him but not showing it. This way one can use his senses to watch a different field that needs also attention and conserve valuable cognitive resources.

Another classification of attention is based on the senses involved, that are the visual, auditory, olfactory, and haptic senses. Visual, auditory, olfactory and haptic having to do with eyes, ears, nose and touch respectively.

Visual attention is the most studied approach. There are at least two metaphors when it comes to visual attention. They are analogies of the actual processes that are thought to be taking place for the visual attention. The fist one is the metaphor of the spotlight that has already been mentioned and is already a way to perceive the notion of attention [Chun and Wolfe, 2005, Theeuwes, 1991]. The other is the metaphor of the zoom lens [Eriksen and Yeh, 1985] (as seen in figure 2). The zoom lens inherits all the properties of the spotlight metaphor, however it differs on the fact that has additional properties. The zoom lens, as known from the cameras, can zoom to a specific area and thus change the focus in a scene. This way we can perceive a scene with different levels of detail with respect to our zoom level choice. One can understand this metaphor by thinking of a scene with actors and a spotlight focusing on them. While the spotlight moves the focus to both actors, the zoom lens can fix the attention and the zoom level to choose only one of the actors. This way the other actor is filtered out of the visual frame and cannot be perceived.

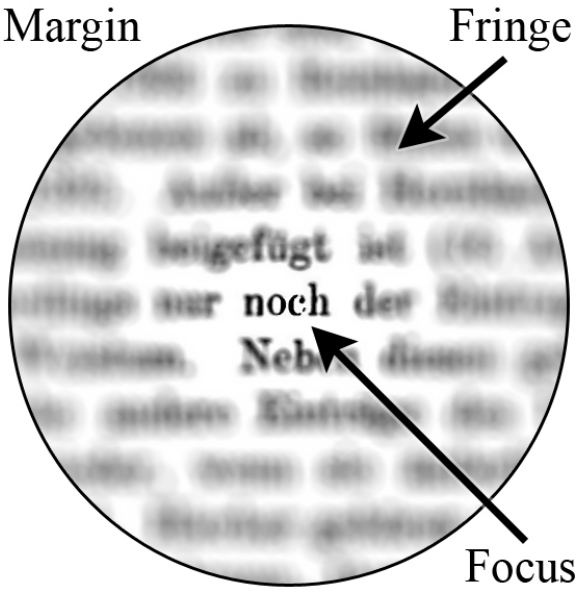

Figure 2: The zoomlens analogy of attention.

With respect to collaboration there is also the joint attention. Joint attention is the type of attention that connects two or more persons with an act or object. Social behaviour and social skills are related to joint attention acquisition [Mundy and Newell, 2007]. It is also not known whether

joint attention is part of attention or a separate mechanism required to develop attention skills.

2.3 Models of Attention

Concerning cognitive science models of attention there are several models proposed. Each one trying to decrypt and formalize the way attention mechanisms works and which act of the mechanism of attention comes first.

The early selection model of attention, of Broadbent [Broadbent, 1954], states that stimuli are filtered, or selected to be attended to, at an early stage during processing. The filter in Broadbent's model can be seen as the selector of relevant information based on basic features (the filter procedure can be seen in figure 3). Those features can be color, pitch, or direction of stimuli. A preattentive store holds the information of stimuli. Then it is the filter that takes part and information with similar characteristics pass through the filter.

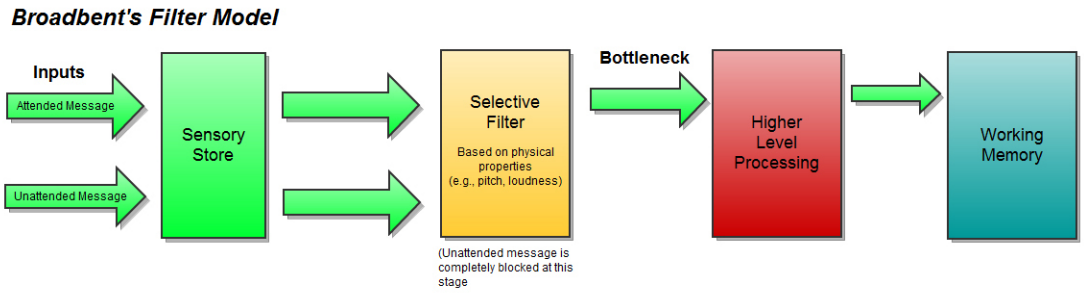

Figure 3: The Broadbent model of a attention.

On the other hand, late selection models oppose to the idea of early filtering and suggest that information is selected after processing for meaning [Deutsch and Deutsch, 1963]. According to this opposing models, all information is attended to and thus processed, despite ones intention or not. Information inputs are processed equivalently, until semantic encoding and analysis can be performed. This notion implies that internal decisions of stimuli relevance must be made, before allowing it to gain conscious awareness.

In order to extend Broadbent's filter, Anne Treisman, proposed the attenuation model [Treisman, 1969, Klein, 1996] (it can be seen in figure 4). This theory supports an early-selection filter. However the difference is, that the filter also attenuates stimuli presented to the unattended channel. If a threshold is passed, info from the unattended channel will leak through the filter and could be attended. As the unattended channel includes weak attention to information, to gain conscious awareness this information must surpass a threshold, which Treisman believed was determined by the information meaning.

Deutsch and Deutsch and Norman where inspired by the cocktail party effect to propose their model of memory selection. The cocktail party effect is an example of how unattended information can gain one's attention [Deutsch and Deutsch, 1963, Conway and Cowan, 2001] (can be seen in figure 5). The cocktail party effect suggests an event where someone is at a social gathering and has a conversation with friends or colleagues. While attending to this conversation, in an instance, he hears someone in a different conversation mentioning his name. The act of hearing an information relevant to him grasped his attention. This unattended-to information somehow

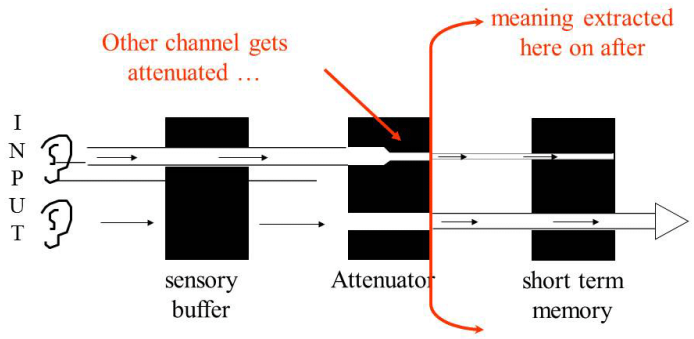

Figure 4: The late selection model of attention from Deutsch, Deutsch and Norman.

gained his attention and was processed beyond its physical characteristics, for its meaning, that is his name in this example. The model proposed share ideas with Broadbent's model. However, attended and unattended information pass through the filter, to a second stage of selection on the basis of semantic characteristics of the information. Therefore, there is a second mechanism that decides what information is attended to.

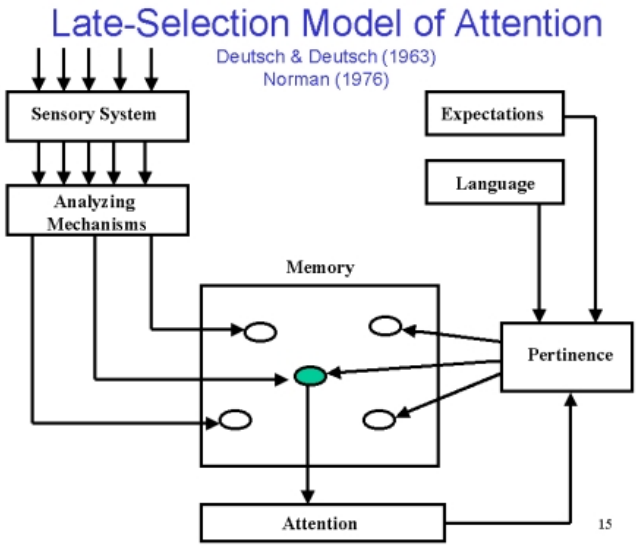

Figure 5: The late selection model of a attention from Deutsch, Deutsch and Norman. Inspired by the cocktail party effect.

Additional research proposes the notion of a movable filter. The multimode theory of attention combines previous models into one unified theory [Johnston and Heinz, 1978]. Within this model, switching from physical and semantic features depending on the person's needs is done. As a basis for selection it yields advantages and disadvantages. The stimulus information will be attended, through an early filter based on sensors, then as complexity increases, semantic analysis is involved, in order to compensate for the limited capacity of attention.

To fill in the gap of the known models, Daniel Kahneman proposed the attentional capacity

model, approaching attention by its division and not selection mechanisms [Globerson, 1983]. Attention is described there as a resource that requires energy or mental effort. The greater the complexity of a mental task, the greater the effort needed (the model can be seen in figure 6). He believes that three basic conditions are required for proper completion of a task. Those are the combination of attentional capacity, momentary mental effort, and the appropriate allocation policy of the attentional capacity. His model is also compatible with the top down orientation of attention. In order to direct attention appropriately, one must attend to relevant information, while neglecting irrelevant information to prevent becoming distracted.

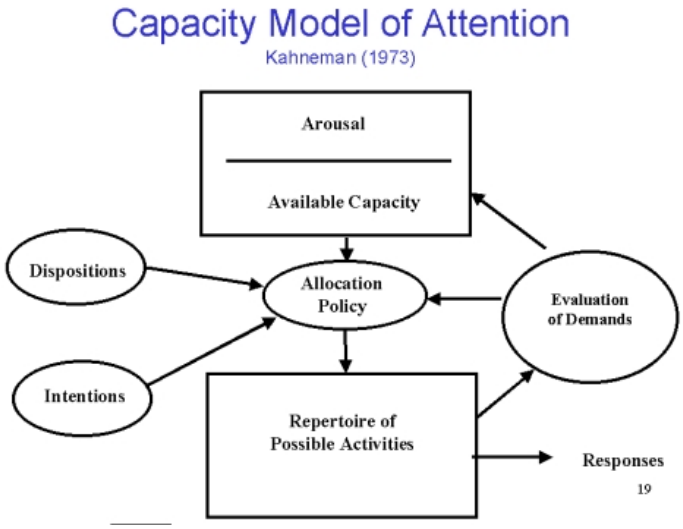

Figure 6: The late selection model of a attention from Deutsch, Deutsch and Norman.

2.4 Behavior and attention

During communication between two or more people, a person's attentional intention is related to his behaviour [Poggi, 2001]. While people communicate, they try to exchange ideas with the other participants through verbal and non verbal ways. These ideas are formed into communicative signs that fall into two classes. First class refers to what one wants to communicate and how he is planning to do it. The other class, refers to the abstract or concrete event and acts one communicates about. To achieve communication the speaker uses gestures, facial expressions and overt attention signs that differ from culture to culture and from one place to the other. According to specific culture or location specific overt behaviour, gaze orientation or gestures can be translated to attentive or not attentive behaviour. Many of these indicators are also inter-cultural and common. It is reasonable, to connect focus of someone's gaze on a person or object as attentive behaviour or to link lack of interest with fatigue signs and inattentive behaviour. On the same side, deictic signs, such as gestures can also give information on the speaker's beliefs so as to communicate what is his perception of the world he imagines. As a translation of attentive behaviour also stand the small words someone says. Those words such as "hmmm", "aha" etc are often used as back channelling feedback, to give the speaker the idea that the other person has understood and is attentive.

3 State of the art models of Attention

Previous computation models of attention can be grouped according to the stimuli of the agent as top-down, bottom-up or both. This a very general classification and is mostly useful for psychological purposes. For this reason, we introduce another classification for the existing work. Previous models can be classified into (1) saliency points methods that try to find important points in a scene; (2) semantic information methods that use or attach semantic information to a scene; (3) two party communication methods that tries to compute the level of interest of the user while interacting with a virtual agent; and (4) attentional methods for virtual agents that try to implement agents capable of orienting their attention towards points of interest in their perception field and expressing their intentions with a certain behaviour or action.

3.1 Salient points methods

The first group of classification one should consider is the methods to find interesting or salient points in a scene for an agent to focus. Known and discussed members for the group are the saliency maps, the salient objects, petters combined (top-down and bottom-up) and visual field simulation.

The saliency maps are very well known to find the most interesting points in order to driver an agent's attention. Peters et al. [Peters et al., 2011, Itti et al., 2003] present them properly in their survey. They are used as a bottom up technique and thus driven by external cause. They are used to drive attention and interest for a virtual agent in an unknown environment (virtual or real) where input is taken from a web camera. These methods are used in frame by frame processing by actually creating a 2d grey-scale map representing the areas most possible for the viewer to pop out. This 2d map is created by a superposition of information of lower level filters that combined give the saliency map [Itti et al., 2003]. The filters are applied to each frame of the video and can be the Gabor, colour, luminance, intensity, depth or motion. Each point from a filtered frame, points to one specific location in the saliency map. Then from the map an attention threshold is calculated in order to find high activity locations in the saliency map [Itti et al., 2003]. These predicted locations are probably the most informative and interesting of the scene and can thus help reduce the complexity of visual search on a complex scene. This way an agent can be oriented rapidly towards relevant parts of the incoming visual input.

The limit of this approach and equivalent approaches [Peters and O'Sullivan, 2003, Itti et al., 2003] is the complexity of the computation that remains high. Another limitation is that they cannot be used for object recognition. Also, extra semantic information is needed to recognize what is really attended in the scene. Because of their high computational complexity other methods that use less filters, are preferred.

Another model that can belong to this group is that proposed by Kokkinara et al. [Kokkinara et al., 2011]. They propose a method to extend an already existed attention mechanism [Oyekoya et al., 2009] and integrate it to the Second Life (SL) platform. Because it applies to a virtual world there is meta-data information on the world which can be used to properly calculate saliency. The model they suggest in their work computes a saliency summation of particular characteristics for the objects on the virtual world. Then the gaze and head orientation are driven to the object with the maximum saliency. For every object a proximity based on the euclidean distance between the object and the user is calculated and then fitted within a Gaussian curve. Then eccentricity is also calculated in order to understand if the object is on the periphery of the vision of the user. Also velocity is calculated with respect to the user's vision, objects with higher

velocity are more likely to be attended. Another parameter calculated is the rotation speed of an object, as objects with higher rotation rate are more likely to be attended. The sum of all the above salience is calculated and the combined saliency is used to determine which object will be regarded either with eye or head fixation. Also a combined behaviour of eye and head fixation might be used if needed.

The method was evaluated by a sample of participants that had to choose between the default SL avatars and a more advanced enriched with the attentional model. It appeared from the results that the former was preferred and performed better for the participants. It was judged as more natural and credible. To avoid high computational complexity this approach does not take into account some information like colour or luminance that saliency maps favour.

In [Peters and Itti, 2007], Peters and Itti propose a combined top-down with bottom-up model to find points of interest in a series of frames on a video. They claim and prove that while combined with a bottom-up model a top down approach can improve the performance and the results received. The top-down module of the approach consists of a learning module that works in two stages in order to associate the most important information, the frame-structure, of an image with special locations on it. Those locations are carefully chosen to be task-relevant, under the current task performed. A set of videos with users that have are playing games, with information about their gaze, is used to train the module. In this way an empiric model can be built based on real world information of real users. Also for each image a vector with the gist identity of the image is taken, mostly a low dimensional feature vector. For the bottom up module method similar to [Itti et al., 2003] with saliency maps is used. The method was properly evaluated and performed very well. The possible limitations of this method is the need of a proper training set as a supervised method. That means that the training set should be ideal, which is not easy and also that the method will give the agent a tendency to perform good only on similar tasks.

In order for the agent to be able to perceive his environment and points of interest, an interesting implementation simulating the human field of view for attention is proposed in [Kim et al., 2005]. It is implemented in a cognitive architecture named Soar. The model proposes a segmentation of the field of view in three zones. A 190 degree top-down view is considered. In this field there is a central arc with 30 degrees of high attention alert and thus maximum perception probability for an object or person to be perceived. Medium attention alert is connected with a deviation of 30 degrees on the left and another 30 degrees on the right of the central arc. The remaining 100 degrees are equally distributed to the remaining area of the visual field. The side view of the agent attention is modelled as 90 degrees field of attention where the central 30 degrees receive the most attention (high alert) and the rest 60 degrees receive lesser attention (medium alert). The model is combined with a saliency method [Nothegger et al., 2004] to calculate possible saliency depending on certain objects properties. The limits of the method are mostly linked to the bottom-up principle of the method as there is not semantic information attached from the saliency just possible saliency and interest.

3.2 Semantic information methods

In order to combine semantic information on virtual world or to create semantic information to label actions there is another group of consideration. This group contains methods that use or attach semantic information on the information processed. The members of this group are the object based recognition methods and the spatiotemporal processing.

The object based recognition methods are another group for consideration [Peters et al., 2011,

Viola and Jones, 2001, Raidt et al., 2005]. They can be used either as a top-down or bottom up method. Viola and Jones [Viola and Jones, 2001] propose an object recognition method that is used to recognise objects from images. The proposed implementation is based on three important contributions. The first is called "Integral Image" and is a method to extract fast features (called rectangular features) from the image of concern to be used on the next steps. The second is the use of a learning method to create rapid classifiers that learn to determine a small set of important features in images that hold the property we are interested in detecting. The last one is a set of more sophisticated classifiers that get incrementally more complex, to rule out the background regions of images. This way more computational resources can be devoted on the valuable areas of the image. Their method has been used with success into face recognition and provided results as fast as other well known methods. Concerning the limits of this object recognition method, is the bond of the system to information given on the training session. It is not autonomous enough to be used to find general objects and has to be trained each time for the object or feature we are interested. This problem makes it difficult to use in a virtual agent.

A work related to the saliency maps, but with the goal to give semantic information to a set of frames from a processed video, is the spatiotemporal processing [Peters et al., 2011, Oikonomopoulos et al., 2006, Rapantzikos et al., 2005]. In order to enhance the results and enrich them with space and time related information the existing saliency map approach methods can be used to acquire space and time points of interest. To achieve this, instead of only the current frame information, a group of frames is processed in every step to correlate the frames with the information attached to them. This way even considering a small amount of frames to save computational resources one can find the points of interest that respect both time and space constraints. Actions such as walking, running or even more abstract information not categorized can be extracted. For instance, in a security department, a suspicious behaviour in an area not expected to have activity, could be recognized. The limitations of these methods still have to do with the high computational load. Every saliency map computation in a frame is multiplied by the depth of the group of frames they process each step and thus makes it a lot more considerable as a load.

3.3 Two-party conversation methods

The third group consists of implementations, that model the level of the interest on a two party conversation for the participants, a user and an agent, and the appropriate behaviour of the agent according to these.

In [Peters et al., 2005] Peters et al., propose a finite state machine model to evaluate dual party conversation. The model suggests a finite state machine approach to express the state of a listener and a speaker in an engaging conversation. Each of the two agents has its attention implemented with a finite state machine with two states, labelled "look" at and "look away". For their conversation three states (phases) are defined. The "establish", the "maintain" and the "close" phase. A single finite state machine with three states, one for each phase, keeps track in which phase the participants are. In the establish phase the agent perceives the attention of the second party that is interested to initiate an conversation. Then with the involvement of the two agents to a conversation, the maintain phase begins and continues till an end phase arrives. While the two entities communicate, both attention levels are elevated for the scope of the conversation and the lack of interest means the end of the conversation. To implement the above finite state machines, the HPTS++ definition language that provides tools for describing multi-agent systems

in finite state machines, is used. By perceiving the gaze of the speaker and the listener a change in the state of the finite state machines can occur and thus on the state of the conversation. For the implementation of the above state machines three algorithms are needed. One of them is used to compute the perceived gaze information and to calculate the interest level using the direction of eyes (gaze). The other two compute the probabilities of changing state for the Listener and the Speaker state machines. The simulation is being used in a probabilistic manner in order to simulate a possible conversation and not to be used to perceive a real one. There are limits that are attached to this approach. According to the author, only the gaze was taken into account while new state machines could be created for gestures, facial expressions and the emotional state of the agent.

To improve this model Peters et al. [Peters et al., 2009] proposed another way to estimate the level of the users interest (LIU) in a two party conversation with an embodied conversational agent (ECA). A simple web camera is used to capture the user gaze and associate it with the level of interest he has. The level of interest of the agent (LIA) is associated with the LIU in order to create an engagement chart. This engagement chart consists of 9 values which are the possible values of the Cartesian product(none, very low, low, average, quite strong, strong some of them are repeated-quite strong, low,very low as the interest level falls) of the corresponding values for the user and the agent. In order to assess the user's level of interest they use a number of metrics. These metrics are the directness and level of attention, the virtual attention objects (VAO) and the level of interest (LIU). The first correlates the user's head orientation with gaze in a way to assess his interest. A head that has an orientation away from the camera but a gaze that is looking the camera is assessed with a lower value of interest. The second assesses users attention to objects on the scene and keeps track of every one object if it is looked. Even the agent is considered a VAO. Then for the LIU, a LIU is attached to every VAO of the scene set. Also a LIA for the agent is calculated. The agent responds according to a level of attention that matches that of the user. When the user is more motivated for the interaction and thus looks the agent then the agent's attention rises to the highest level. If their interaction also contains an object then the LIA can be high while the user pays attention to the object and not the agent. The system was implemented in two modules (a gaze detector module and a shared Attention module) that communicate via *Psyclone* connection-a blackboard system for use in large scale multi-modal Artificial Intelligence systems. The approach is limited by the dependence to lighting conditions, the quality of the web camera and the lack of a perfect model for gaze detection with a single camera.

3.4 Attention methods applied on virtual agents

The last group consists of implementations of existing agents capable of orienting their attention with respect to interesting points on their perception field and to express their internal attention level with a certain behaviour or action. Some members of this group, are the GazeNet method [Trias et al., 1996], the virtual military pilot [Hill, 1999] and a collaboration agent named MAX [Pfeiffer and Wachsmuth, 2008].

In [Trias et al., 1996] Trias et al. propose a model of decision networks to integrate behaviours or Virtual Agents and Avatars, named PatNets. This work also proposes a special module to model the gaze of an agent named GazeNets. The information of GazeNets, is modelled and represented in the PatNet system that is a finite state machine implemented as a directed graph. To show the use of the method they suggest a hide and seek game with a virtual seeker and multiple virtual hiders. The GazeNet system drives the virtual seeker's gaze in order to find the virtual hiders. There is not only a central mechanism that decides but four different parallel running modules that

collaborate to create a consistent gaze behaviour for the agent. The agent can either attend on an interesting point on his environment (attract) or avoid obstacles on his way (avoid), or perform visual search to find a hider (visual search) or look spontaneously because an object or action drew his attention (visual pursuit). Particularly, the agent performs attract when an interesting point is found and *attracts* his attention. As a prechosen behaviour, he will interleave his gaze between this point and the ground. visual search in order to find a hider in the environment. If an obstacle is found on the way the *avoid* behaviour will help him to fix the gaze towards it. It happens mostly if there is an immediate proximity with the obstacle. While walking in an uncertainty path, the agent, can again perform *visual search* to find a new interesting point. If a new object appears suddenly then the agent will perform the *visual pursuit* behaviour and fix his gaze towards the new object, even if it is moving. The advantage of the method is that multiple simultaneous behaviours are allowed and can be implemented on the agent using different parallel modules that can drive different behaviours responsible for walking, gaze or head orientation. Later on the implementation is also used in [Chopra Khullar and Badler, 2001] by Chopra and Badler, to model the autonomous behaviour of attention on virtual agents. A possible limit is that the system works only on a virtual environment. In order to add a human in the system his virtual representation should be introduced in the virtual world. Therefore, the virtual agents would be able to perceive him.

Randal Hill in [Hill, 1999] uses a hybrid model of bottom up and top down attention to simulate a virtual pilot for helicopters to be used for a military application of pursuit. The virtual pilot is implemented in Soar which is a cognitive architecture. The pilot uses a combined saliency model based on object properties (both saliency map filter information and object properties such as distance, shape and color) for the bottom-up attention. It is useful in order to shift the attention of the agent to a new object. On the other hand in order to achieve certain goals in the simulation environment the top down attention is used. This is done by explicit goal operators on the Soar architecture. Before the real attention procedure a pre-attention procedure takes place. The pre-attention procedure, acts as a primitive action to group objects that have same properties (proximity, similarity, motion). In this way it limits the resolution and the quantity of the objects to be perceived at the same time and valuable computational resources can be saved. Also, on each cycle of preattention, new objects without detailed information are perceived and information for the existing objects is updated. The grouping can happen both in a top down and a bottom up attention procedure, depending either on the environment or on agent's goals set. Following the preattention procedure, the attentive procedure focuses on a visual target according to the zoom lens paradigm of Eriksen and Yuh [Eriksen and Yeh, 1985]. The advantages of resource saving are obvious and the result outstanding in performance. Because the objects that need to be attended simultaneously can not be limited in number, the computational load can not be controlled yet. Another limit is the present vision system of the pilot. It is not limited like the human vision field. In [Kim et al., 2005] Kim et al., use a more realistic field of view in order to simulate the vision cone of human beings that could be used for the virtual pilot. In addition to this the pilot uses a covert model of attention so that shifts in his attention are not affecting his gaze.

MAX [Pfeiffer and Wachsmuth, 2008] is an ECA that the authors have enriched with joint attention in order to participate in conversations with other agents or users, with a more human-like manner. Joint attention is very important when it comes to communication and common objects, interests or persons are present in the environment. It is also very useful as a property when there is a need for alignment and cooperation with another agent or user.

Pfeiffer and Wachsmuth state Kaplan's [Kaplan and Hafner, 2006] 5 requirements for an agent

to achieve proper joint attention.

R1: The tracking of the other agent's attentional behaviour by gaze monitoring.

R2: Switch between situation context and agent perception.

R3: Recognize the attentional intention of the interlocutor.

R4: Instant reaction for credibility and simultaneity.

R5: Use of an overt behaviour that should be easily recognized by the interlocutor - non ambiguous.

MAX uses both top down and bottom up attention processes to implement joint attention. He is implemented in a cognitive architecture that Max is named CASEC and is based on BDI (Belief, Desire, Intention) instead of Soar (Virtual Pilot etc.). The BDI paradigm wants an agent cognition and actions to be driven by his beliefs, desires and intentions. MAX, in order to drive intentional tasks has a mental state that is updated with the completion state of every task. He also has a model to keep track of partner's information, the perceived attention state of the user-partner(such as gaze), for the collaboration task. The information on the environmental context is also perceived in order to adapt to the environmental process and to embrace the situation context. This is achieved by bottom up attention used to relate objects on the environment with an importance (saliency) for the current situation. The agent should be overt in order to express himself to the interlocutor.

The attention detection is achieved by an eye tracking device. The gaze tracking respects a cone of attention of 2.5 degrees (that resembles human field of vision) to perceive human focus of attention and histograms are used to understand if human is paying attention (a threshold of 400-600ms focus on a fixed point is used). The partner model, perceives the interlocutor's state that is in compliance with the interpretation of his overt behaviour (belief). Also, intention is perceived for the intentions of the partner and his desire. This triad models the partner perception of the virtual agent (respecting the BDI architecture). Another interesting property of MAX is the attention manipulation. MAX is capable of intentional gaze, deictic gestures, and verbal expression in order to manipulate the interlocutor's attention level. In order to drive his partners attention, he uses intentional actions to persuade and manipulate him. The Agent's mental state is also modelled with the BDI paradigm.

The limits of this approach are mostly linked to the architecture and the fact that the method has not yet been evaluated by human participants. A preloaded cognitive system can overload the system while can be a bottleneck in an existing system. Also, every architecture comes with limitations on its environment.

4 Discussion on our Attention model

The context of the Ingredible project consists of a two party non-verbal interaction between a user and a virtual agent. The virtual agent has a very simple representation like a mannequin without facial expression and gaze behavior. It represents a magician who performs three tricks in front of an audience (the user). While the virtual agent performs an act the user might lose interest or give corrective instructions such as "continue faster" or "stop", for such a reason the agent should pay attention to the user while performing its trick.

To start the interaction the agent has to understand that the user is present. During the performance the user is the only point of interest for the agent. So we do not need to model different attentive behaviour such as those implemented in GazeNet [Trias et al., 1996]. Moreover

the user is continuously tracked by an Xbox kinect device so there is no need for saliency maps or saliency object methods to find new points of interest.

The agent has two main goals: to perform its show and pay attention to the user. For such a reason both a top-down and a bottom-up Attention model is needed. On the one hand while performing his trick the agent should focus from time to time on the user to check that he is still there (top-down attention). On the other hand, the agent's attention could be drawn by some user's behaviour (bottom-up attention). For example if the user shows boredom and try to go away the agent should stop its trick and call the user back (similarly to MAX [Pfeiffer and Wachsmuth, 2008]). If the user asks for repetition the agent should be able to repeat its trick.

To estimate the user's interest level, we plan to take into account the user's head and body orientation as described in [Peters et al., 2005, Peters et al., 2009] and his gestural behaviour. For example if the user asks for repetition we could presume that he is very interested in the agent's performance.

We also think that dividing the vision field into three zones of alertness (as in [Kim et al., 2005]) to drive the agent's attention is also useful. The segmentation on the vision field can make the agent behaviour more believable and human-like. With this capability the agent would be unable to fully perceive the user when he stands behind his back, so according to the agent head and body orientation, its perception of the user changes.

5 Project Context

The project as already discussed puts a user and a virtual agent to the same environment to communicate through gestures and body movements. The agent is a human resembling manequin with neither face expressions nor eyes nor talking ability. The user is also not using voice to command or communicate with the agent. The user and the agent are coupled in a variety of different magician scenarios context. For this reason the user is continuously tracked and each movement of the user and the agent are properly modelled. In the following sections we explain the scenarios and the technical details of the project, the architecture and the main picture of the whole project. Moreover, we present in detail the architecture of the Attention module and how it is built and then we present a preliminary demonstration and experimentation of the module. In the end we present a conclusion commenting on our early results and what could be possibly improved and what could be a probable future research.

5.1 Project scenario

The desired context on which the project will be tested suggests that the agent will perform a magician show that consists of three magical tricks. In the beginning, the agent introduces itself and shows to the user that it is a magician that will perform a show and starts with the tricks. The first trick proposes that the agent will make a rabbit appear from an empty hat. There is no real hat or rabbit in the context but the agent will make the proper gestures to show that expressivity. The second trick is to make a box disappear and last trick is to make a very long scarf to appear from a jacket. Again the agent will not have a real or virtually designed box or long scarf but will make the gestures as if some virtual objects were all the time present in the scene. The order of the tricks is off course irrelevant. The user interpretation in this scenario consists of suggestions and corrective movements on the agent. The user will also express his interest with his postures

and movements. On the mean time, it has to interpret the user's moves and postures and follow a strategy to try to attract again the user. None of these scenarios are ready yet so our module was tested in more simplified scenarios. The user and the agent are interacting and the agent pays attention to the user whenever the user appears in the scene or performs a movement.

5.2 Technical details of the systems

In the system that is proposed and will be used the user will be continuously tracked. To achieve this a camera system is needed. For this reason either a kinect or an 8 camera system is used. For the kinect device the OpenNI Library is used that provides a skeleton of the user. In any case a 15 joint skeleton including the head, the shoulders, elbows, torso, hips, knees, hands and feet of the user is provided. Also the project is built up on different modules that need to communicate data between each other. In order to do this google's protobuf library has been properly adapted in order to handle the messaging. Protocol Buffers are a method of serializing structured data. As such, they are useful in developing programs to communicate with each other over a wire or for storing data. The method involves an interface description language that describes the structure of some data and a program that generates from that description source code in various programming languages for generating or parsing a stream of bytes that represents the structured data.

They are considered lighter than equivalent JSON or XML serialised messages so that they can be quickly exchanged and processed. There are three types of these messages used in the project. These are the generic, the skeleton and the gesture messages. The generic is used to exchange general data that is not predefined between the modules. Every generic consists of a number of features attached to it. Then there is the skeleton where the user coordinate and rotation information is stored and finally the gesture message where a set of characteristics for the user's gestures are stored.

5.3 Kinect OpenNI library

The OpenNI organization is an industry-led, not-for-profit organization formed to certify and promote the compatibility and interoperability of Natural Interaction (NI) devices, applications and middleware.

As a first step towards this goal, the organization has made available an open source framework the OpenNI framework which provides an application programming interface (API) for writing applications utilizing natural interaction. This API covers communication with both low level devices (e.g. vision and audio sensors), as well as high-level middleware solutions (e.g. for visual tracking using computer vision).

5.4 Ingredible Communication framework

In general the Communication framework is responsible for message creating, sending, receiving and synchronisation between the modules. It has been written in several languages (Python, Java, C sharp, C++) in order to make it easier for researchers to add their code to the project. Any module should have the file with the network configuration of every other module in order to be able to receive and send a message to it. The first thing to be done is to initialise the communication between a module and the modules that it wants to receive messages from. Then an event handler handles each message depending on the type of the message and the sender. Afterwards it is

possible for the module to broadcast messages to the framework so that the other modules can receive.

5.4.1 Generic message

A generic message is a message that can consist of different features. A feature is a set of values and a name and can be an integer, a float or a string. The generic message is built of one or more different features attached to it.

```
message Generic {
repeated Feature features = 1;
}
message Feature {
optional string name = 1;
optional int32 intValue = 2;
optional float floatValue = 3;
optional string stringValue = 4;
}
```

5.4.2 Skeleton message

The skeleton message consists of the geometrical properties of the user that is tracked. It is related to the OpenNI framework. On the skeleton the OpenNI framework gives info on fifteen different joints of the human (as seen in figure 7). These info include the rotation on each angle of the user and the position with respect to the OpenNI framework kinect coordinates. The different joints provided are the head, the neck, torso, left and right shoulder, hip, elbow, hand, knee and foot. To send a skeleton a root bone should be set and then the skeleton build the skeleton with respect to this root bone.

```
message Skeleton{
optional Bone root_bone = 1;
enum SkeletonType {
NORMALIZED = 0;
KINECT = 1;
ARTTRACK = 2;
MOVEN = 3;
}
optional SkeletonType skeleton_type = 2 [default = NORMALIZED];
}

message Bone{
optional string name = 1;
repeated Bone children = 2;
optional Rotation rotation = 3;
optional Position position = 4;
}
```

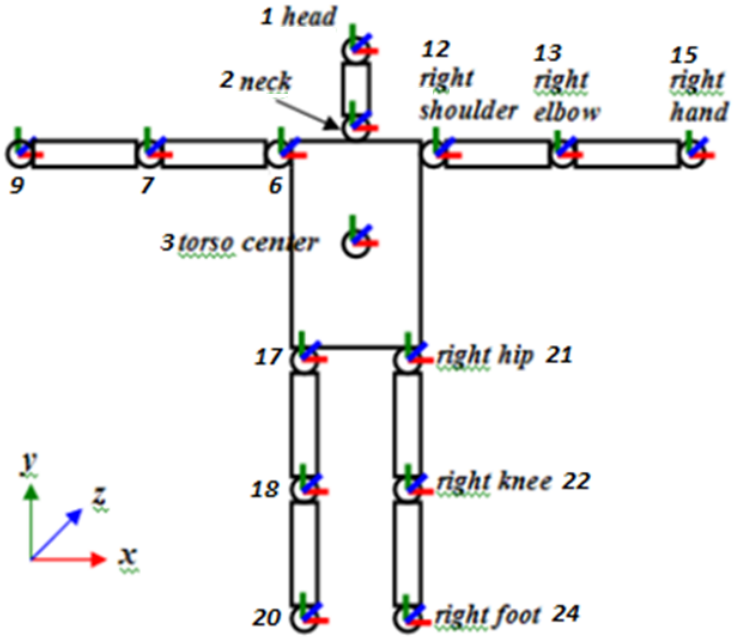

Figure 7: The OpenNI kinect joints of the skeleton and the coordinate system.

```
message Rotation{
optional float euler_x = 1;
optional float euler_y = 2;
optional float euler_z = 3;
optional float quaternion_w = 4;
optional float quaternion_x = 5;
optional float quaternion_y = 6;
optional float quaternion_z = 7;
}
message Position{
optional float x = 1;
optional float y = 2;
optional float z = 3;
}
```

5.4.3 Gesture message

A gesture message is given by the analysis module of the project. It is a recognised move of the user's hands and posture coupled with certain fluidity, speed and amplitude values. An example of a gesture message is following:

```
message Gesture{
     optional string name = 1; //ASCII
     optional int32  amplitude = 2;  //amplitude
```

```
    optional int32  fluidity = 3;  //fluidity
    optional int32  speed = 4;  //speed
}
```

6 Project architecture and algorithms

The project consists for the time being of six modules. As it is an ongoing research project (www.ingredible.fr) that lasts 4 years, this number and the specifications of the project might change (the modules without the Attention module can be seen in figure 8).

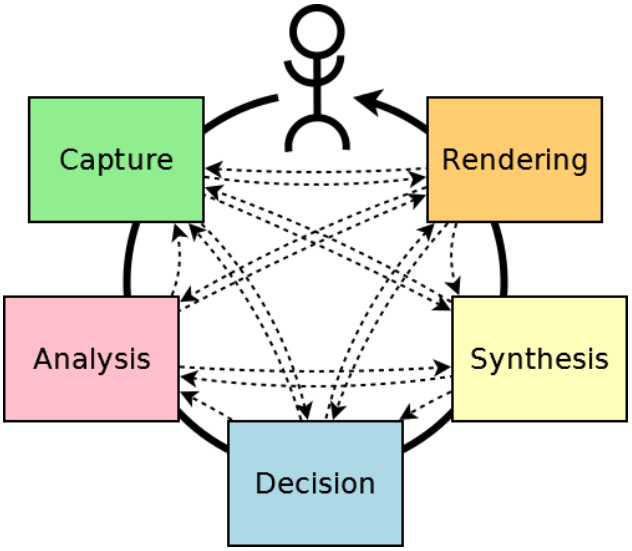

Figure 8: The current project modules and their communication without the Attention module.

6.1 Project modules

As already mentioned the project consists of different responsibility modules. These modules are the Capture, Analysis, Attention, Decision, Synthesis and finally the Rendering. Every module communicates with one or more other modules in order to receive or send messages with either a generic, or a skeleton or a gesture message.

- **Capture**

  The Capture module responsibility is to track the user. The module's calculations handle the video from the kinect or the 8-camera system in order to capture the user and interpret it to a skeleton. It then communicates a skeleton message through the communication system meant to be processed and filtered by the Analysis module.

- **Analysis**

  The Analysis module responsibility is to receive a series of skeletons and process them to give an interpretation of them and calculate some various quantities. The exact interpretation that the module does is to filter and resend the skeleton received from the capture to recognise the

user gestures and characterise them with a certain level of characteristics such as amplitude, fluidity, speed and also send generic messages concerning the speed, the amplitude and the acceleration of the user hand and feet movements to the other modules.

- **Attention**

  The Attention module responsibilities are to calculate the attention level of the agent according to the user's actions. It receives from the Analysis module a skeleton with fifteen bones containing the user's coordinates with respect to a coordinate system. Moreover, the analysis sends generic messages with various info on the user hand and feet movements such as speed, amplitude and acceleration. It also receives from the Synthesis module the head position and rotations for all axis of the agent. With this information the attention level of the agent can be calculated and depending on the calculations, messages can be sent to the other modules. The Attention sents generic messages regarding whether the head position of the agent, that represents also the eye regard of the agent, should be towards the user or not.

- **Decision**

  The Decision module is responsible for the cognitive part of the agent. The agent's intentions and goals to perform the magic tricks are guided through this module. There is an independent from the user list of goals that the agent wants to achieve and sub goals in order to achieve these goals. During the interaction a memory and decision system of each move of the agent is needed. It has to guide the body of the agent to show it's intentions. The messages of the Decision are addressed to the Synthesis module.

- **Synthesis**

  The Synthesis module is responsible for the assembly of the agent decisions. It takes into account the valid configurations available for the agent's skeleton and it calculates for every received message from the Decision and the Attention module the corresponding angles and the positions of the agent's skeleton in order to be represented and sent to the Rendering module.

- **Rendering**

  The Rendering is responsible for the representation of the agent in the screen. It receives messages only from the synthesis that have to do only with the illustration of the agent to a virtual world. It is occupied with the geometry calculations to place the agent in the three dimensional environment.

7 Attention module architecture

This is the module that we are going to focus on the present report. The responsibilities of the Attention module is to calculate the level of attention of the agent. To achieve it, there is a need to interpret the messages received from analysis and synthesis. These messages concern the user's and agent's actions. The module is performing an abstraction between the sensor's of the real world, like the kinect and the agent's sensing capabilities, that is the agent's head position in the virtual world, with respect to the user's position. To explain this further, imagine a two dimensional painting of a human looking at you. One can always get the impression that the represented person is looking

at him. In a thee dimensional representation this is not always the case. The person represented in the three dimensional representation, in the project's case, the agent, is not always giving the impression of looking at you. This is why there is a need for this separation of worlds in order to be realistic. A user moving but not visible from the agent is not valuable user's information for the agent. The present module is helping to achieve this abstraction.

The Attention module consists of smaller processing units. Each unit is responsible for a particular action for the behalf of the attention. Those units are:

- *the vision field model*,
- *the skeleton joint vision filtering*,
- *the motion detection after vision filtering*,
- *the state machine*,
- *the strategy pattern*,
- *the adaptation component*.

7.1 Vision field model

The vision field model is based on the observation of [Kim et al., 2005] about the alertness level on a human simulating vision system depending on the angle of an object or person with respect to this field (as shown in Figure 9). The segmentation he proposed consists of three different alertness levels for the a top-view vision field and two different alertness levels for the side-view field. He also proposes some angles for these segmentation fields. Trying to extend this work, we propose a new model, based on geometry and the use of different filter functions. Instead of having three fixed levels of alertness - high, medium, low - we assume that is feasible to have a mathematical value corresponding to these discrete values but in a continuous spectrum and normalised from zero to one. This is achieved by the filter functions.

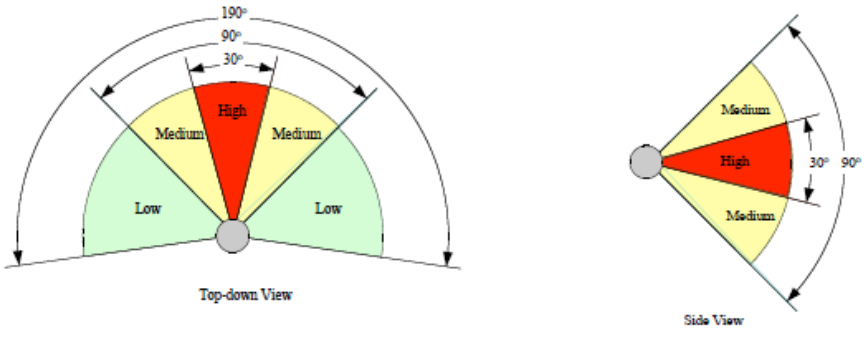

Figure 9: The [Kim et al., 2005] vision field.

7.1.1 Geometry and trigonometry model

The mathematical model works as following. The agent head rotation, position and the user position are used to calculate the angles for the agent's regard with respect to the user position. Considering that the agent has no eyes we use a single vector for his eye regard. The line between the user and the agent and the vector of the eye regard of the agent form an angle. This is the angle of the relative angle between the eye regard of the agent and the user.

The most important step in order to couple the agent's and user's coordinates is to bring them to the same coordinate system (a representation of the two worlds can be seen in figure 10). The coordinates for the user is kinect skeleton coordinates but for the agent the coordinates are Unity3d coordinates. In order to associate the two different systems in a unified coordinate system we have to consider the same scale. We need to have a value that is constant to transform from the virtual world coordinate system to the real world coordinate system. That constant is the agent's height value. We have a predefined height value for the agent so we know exactly how to transform it in the real world. The distance from the feet of the agent till its head in the virtual world divided by the real height of the agent is the scale used to represent the agent in the virtual environment. It is the way we used to transform from virtual world units to the real world units. After putting the two entities in the same coordinate system we consider the head orientation of the agent or more precisely the center of regard of the agent as one of the important values. The real world z-axis rotation of the agent's head gives its regard vector.

Afterwards we can see that the position of the user in the coordinate system and the position of the agent in the coordinate system form a line (as seen in figure 11). The unary vector (vi) that represents this line and points to the user forms an angle between the agent's eyes' regard and the position of the user. Using the internal product of these two vectors we can find the angle of the point of interest and the user. The formulas used to calculate this angle are give below.
Coordinates of the non unary vector (vi):

$$X_X o = (UserX - AgentX),$$

$$Y_Y o = (UserY - AgentY),$$

$$lengthofvi = \sqrt{X_X o^2 + Y_Y o^2},$$

$$cosine = cos(agentrotation) * X_X o/lengthofvi + sin(agentrotation) * Y_Y o/lengthofvi,$$

$$angle = cosine * 180/pi,$$

That angle is the angle that is fed to the filter functions to compute the agent's alertness to the particular point of interest (the agent's vision field is formed by the red lines that can be seen in figure 12). If the absolute value of the angle is out of the vision field then the point of interest can not be sensed. Or else it is sensed and a particular value according to the following filter functions is given.

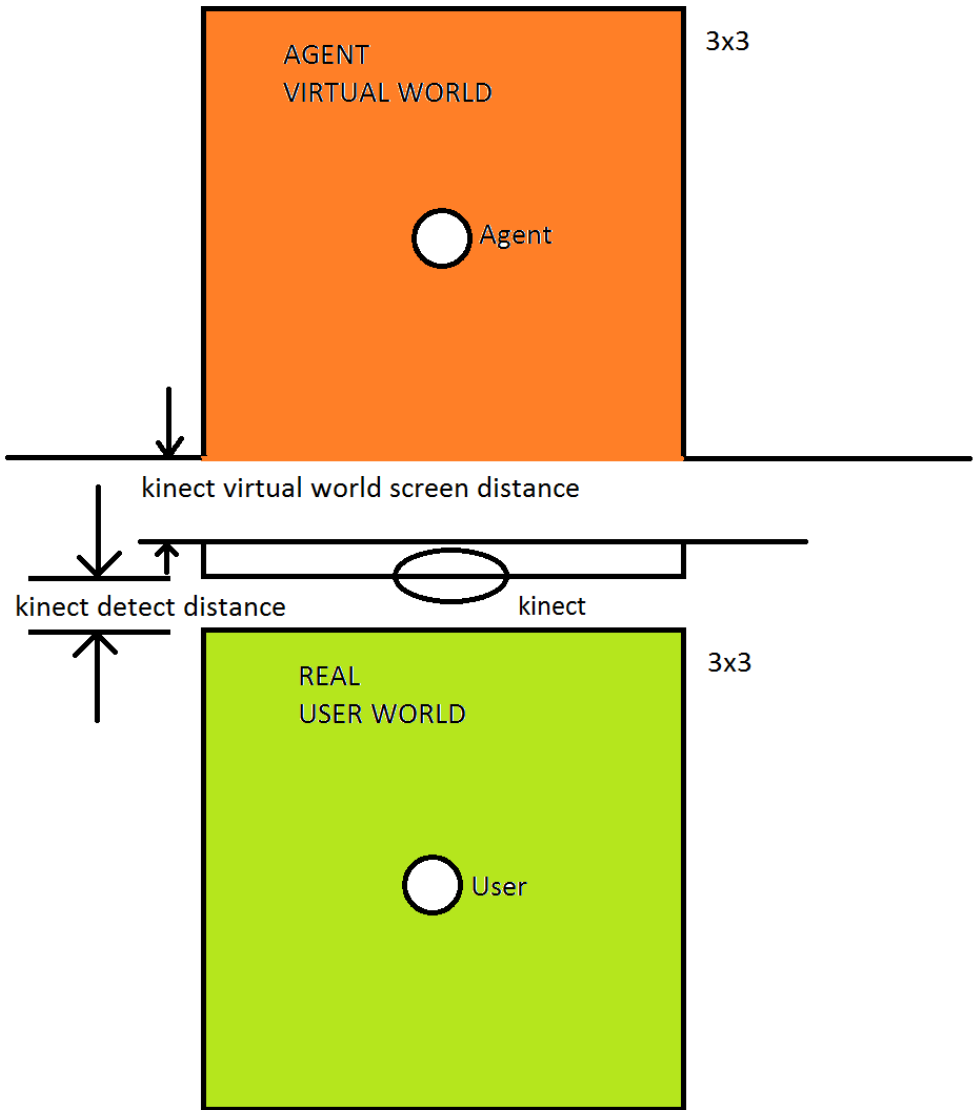

Figure 10: The virtual world and the real world abstraction.

7.1.2 Filter functions

The functions we thought of using are all following the idea that the bigger the angle the smaller the alert level for the agent, respecting the model in [Kim et al., 2005]. Four different functions are proposed. One is a Gaussian filter based on the non-normalised Gauss distribution function. The second is a convex function based on the exponential. The third is a concave function based on the logarithmic function and last is a triangular semi-linear function.

We assume that the functions are symmetric on the y-axis. However it is not proven in biology or neuroscience that our attention is symmetric to the view angle. That means that if a vision field is 160 degrees wide it is not sure that it will be distributed evenly on the left and on the right of a theoretical center of attention. In addition to this, all functions are monotonous on each of the two symmetric spaces. They can be parametrised in order to work with the angles we want.

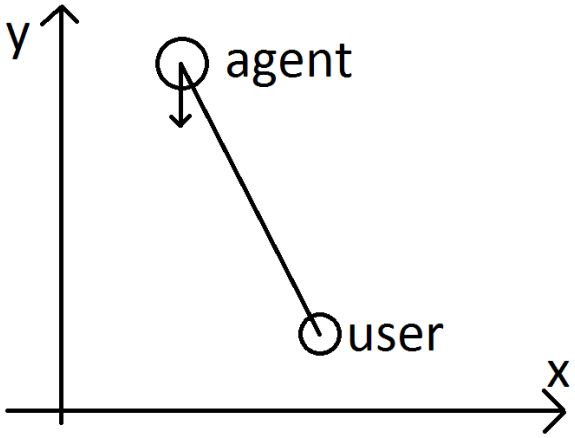

Figure 11: The angle between the agent regard (head rotation and position) and the user position.

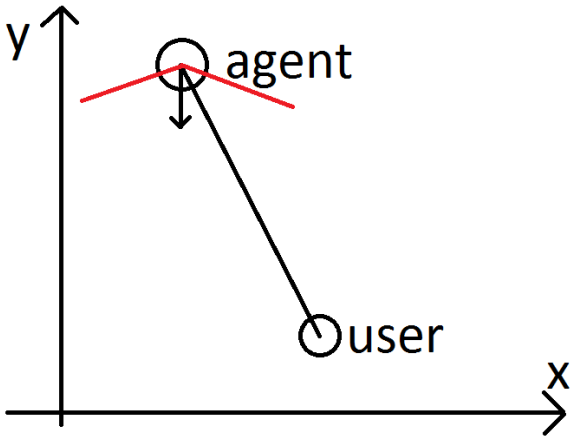

Figure 12: The vision field of the agent drawn with respect to the regard of the agent.

For the project we are using a 160 degree wide vision field to simulate the top view field of the virtual agent in contrast with the 190 degree of [Kim et al., 2005]. The reason why we use a more narrow vision field is double. Firstly, our agent does not have eyes to simulate the gaze. Secondly, there is not a very solid scientific reason to use the 190 degrees as field width. On the contrary it seems quite unnatural using the 190 degree vision field because some of the objects at fair invisible angles will be visible while they shouldn't. We keep the same value for the side view field that is 90 degrees. Every angle on this field view is related to a different alertness level calculated by the filter function.

For the filter functions there is not very specific reasons to consider only one function as the most suitable. We propose the four functions for experimentation and for completeness. Considering each function has different formula can help on different applications where agent attention is needed to behave differently. Also a different function can be used for the top-down vision field and the side-view vision field. There is not a concrete scientific reason to stick on the use of the same

function for both. In addition to this they are very important because they are needed to calculate the attention level for the agent in combination with the motion values received from the analysis module. Because most of the phenomena in nature are following a normal distribution that is close to the gauss distribution, we thought that it would be more suitable to use it for a filter function. However, for research and test reasons more functions and a combination of these can be used as shown and described below in figure 13.

7.1.3 Different propositions for the filter functions

The form of the *gauss filter function* is characterised by its deviation. The gaussian filter used is not normalised as the normal distribution function. The formula used is: f(x)=$e^{-0.5(angle/deviation)^2}$ In order to have a parametrised gauss filter function we have to give it a different deviation. For the gauss function it is known that 66.7 percent of the values are between the - and the + (s=deviation), that 90 percent are in between -2s to 2s and that -3s to 3s is the 99.7 percent of the values. This way we can adapt the function with the correct deviation for the corresponding angles that we want to be inside this space.

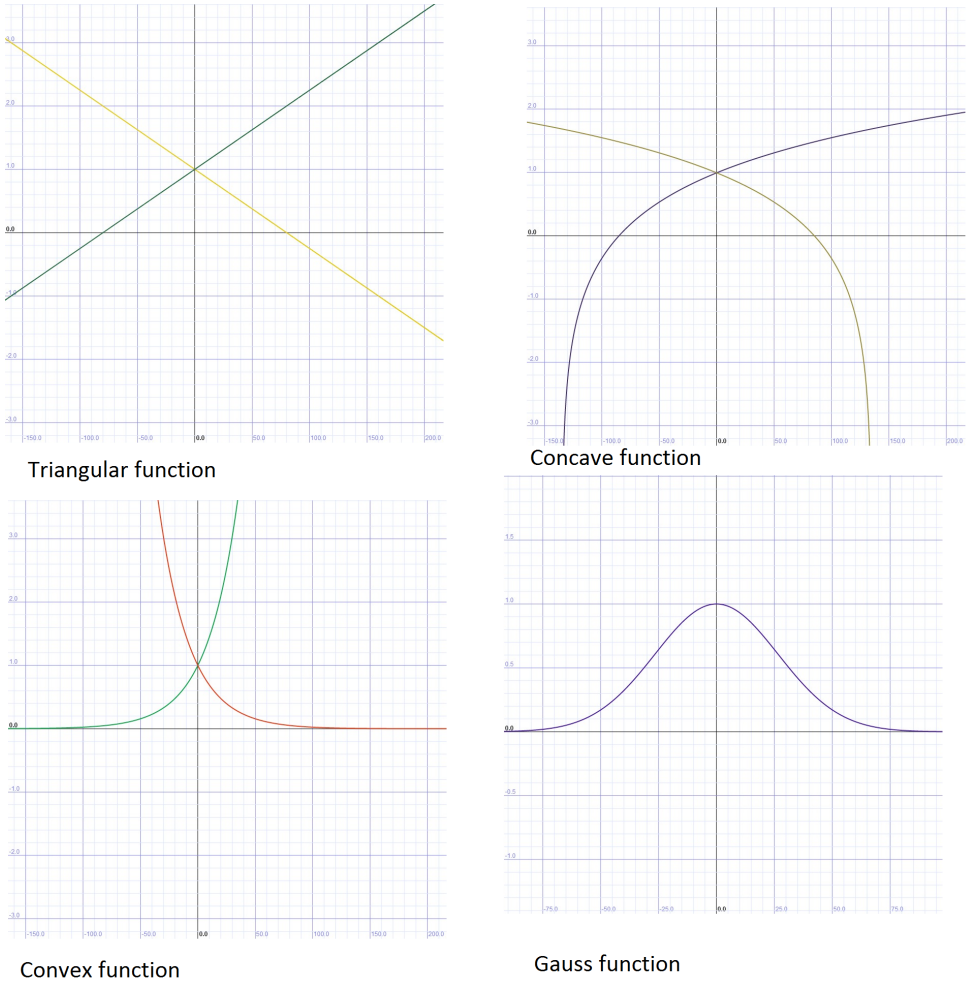

Figure 13: The different proposition for filter functions.

Using the same logic and keeping the constraint of having a monotonous function we propose a *concave function* based on the logarithmic function. In order to achieve this we have to use two logarithmic functions. 1) $f(x) = log((x + 135)/30)$ for $x < 0$, $x > -80$ and 2) $f(x) = log((-x + 135)/30)$ for $x > 0$, $x < 80$

The formula for the convex function is $f(x) = e^{x/27}$ for $x < 0$, $x > -80$ and $f(x) = e^{-x/27}$ for $x > 0$, $x < 80$ The value on the denominator of x is used as the deviation to the gauss filter formula and helps to parametrize the width of the desired vision field.

The formula used for the triangular function is $f(x) = (x + 80)/80$ for $x < 0$, $x > -80$ and $f(x) = -(x - 80)/80$ for $x > 0$, $x < 80$.

7.2 Skeleton joint vision filtering

It is already mentioned that for the model we receive a skeleton from the Analysis module. This skeleton contains information for the coordinates of the torso, the head, the shoulder, the hips, elbows, hands, knees and feet of the user. Not all of the user parts are visible at all times. In order to be aware of how much of the user is visible at every frame we pass every joint of the skeleton from the vision field and the function filter calculation that was explained in the previous step. The joints are fifteen so we get fifteen values of the alertness on each joint of the user. Most important of them for an interaction are the head, the neck, the torso, hand and feet of the user. Every joint's coordinate is associated with an angle with respect to the agent's regard vector. These angles then are associated with a different value of alertness as calculated by the filter functions. Then as these alertness values are a normalised value for each joint from zero to one they are stored to a matrix. Afterwards the values calculated for the hand and feet are used to calculate the value of attention from the vision field.

7.3 Skeleton joint motion detection after vision filtering

After the vision filtering of every joint we have to see if the visible parts are moving with respect to the agent. If the user is moving hands to attract the agent's attention then we have to inform the synthesis module to pay attention to the user by sending a message. The calculation that is done is the following:

A multiplication of current alertness value of every moving joint with the weighted average of the same joint motion parameters (amplitude, speed, acceleration). First we calculate a weighted average for every motion parameter of every moving joint (hand and foot) of the user. To do this we assume three different weights u1, u2, u3 that will be the weights for this average. We assume that speed is very important for attention based on [Kokkinara et al., 2011] and we give less importance to acceleration and amplitude. This way we use a u1 = 0.7 value for speed a u2 = 0.2 value for amplitude and a u3 = 0.1 value for acceleration (we can parametrise this values for furter experimentation). The sum of these values is 1 to form the average. The formula for this calculation is: $BodyPartMotionAttention = 0.7 * speed + 0.2 * amplitude + 0.1 * acceleration.$ Then, in order to calculate the attention value of the agent, we calculate another weighted average using four weight values namely w1, w2, w3, w4 each one needed for a unique joint (hand and foot). If we want to give more importance to the hands we have to use a value more than 0.25 that should have been the normal for a classic average. The value chosen for every hand is 0.4 and for every foot it is the half of the remaining value (0.1) to keep the sum of all weight values to one in order to form the weighted average. Formula for the calculation: $Attention = 0.4 * lhand + 0.4 *$

$rhand + 0.1 * lfoot + 0.1 * rfoot$. Depending on the user's persistence and speed on the movement we can have three different messages proposing 3 different situations. Move only head, move main body with head, move completely towards user. Move can include a combination of translation and rotation.

7.4 State machine and strategy method pattern

In order to codify the context on the non-verbal communication we propose a state machine and a strategy method to decide for the actions on each state. The states we recognised for the project context concerning the Attention module are seven. They can be seen in figure 14.

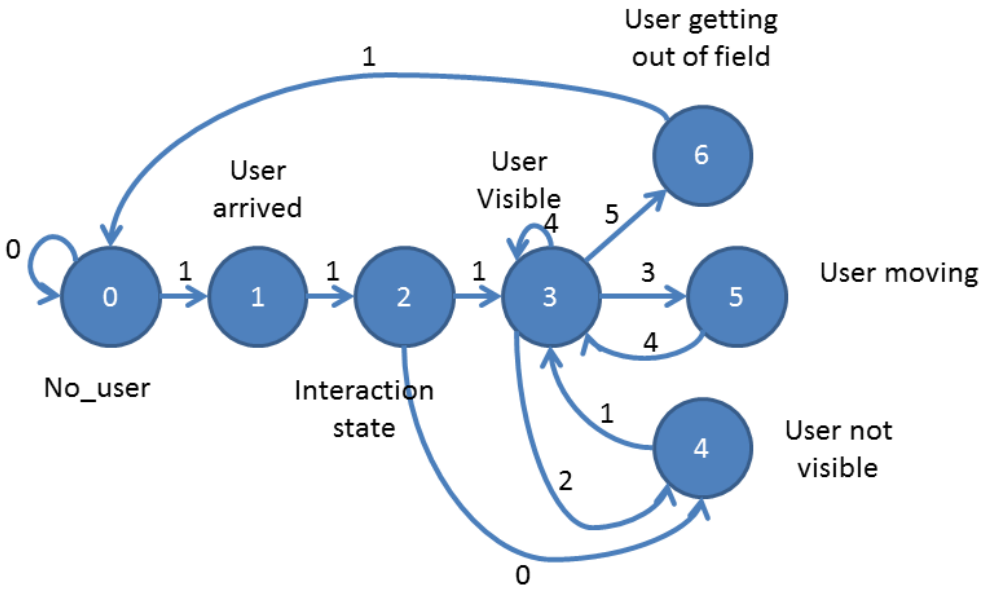

Figure 14: Finite state machine showing the current agent Attention module states and transitions.

- *0: Zero State*

  The agent is unaware of a user because either the user has not entered his vision field or the user is not yet present. It is the initial state before the start of the interaction. If a user is sensed on the vision field then we move from this state to the user present state.

- *1: User present State*

  The agent is now aware of the presence of the user and looks at him. The agent initiates the interaction and waits input from the user. By initiating the interaction the agent moves to the next state of the interaction state.

- *2: Interaction State*

  The user is known that had been visible before a while. He is either visible or not visible. If he is visible then we can move to the visible state. If not we can have to move to the not-visible state.

- *3: User visible state*

  If the user is visible we have to know if he is moving or not. If while in our vision field the user moves a bit then we have to send a look at generic message to the communication framework.

- *4: User not visible state*

  The agent is not looking at the user. So that the user is out of the vision field. If the user is out of the vision field for a long time, we can propose that the agent interleaves between the user and the task at regular intervals in order to reassure that the user is there. Also to check if he is moving or not.

- *5: User moving state*

  The user is inside the vision field of the agent and he is moving. When in this state we want the agent to be aware of the movements on hand of the user in order to react to his movements.

- *6: User escaping vision field/interaction state*

  If the user is trying to escape from the conversation by moving more than a threshold value in a certain amount of time we can assume that he is starting to loose interest and try to attract his attention.

- *7: User not moving state*

  The user is either inside the vision field or partly inside the vision field and the visible parts of the user are not moving. In this case there is no need to look directly towards the user. To reduce the number of states, this state, because of the visibility of the user is omitted and we use the state 3 instead.

7.5 Adaptation component

We mentioned on the user not visible state that the agent can check back at the user at regular intervals. These regular intervals do not have to be fixed in time. We propose an adapted interval time that can be changed with respect to a feedback value. A simple way to do this is to check if the user is moving or not when the agent is looking at the user. If the user is moving then probably the agent should have checked earlier and so a decrease in the time interval is needed. On the opposite case that the user is not moving the agent can increase the interval of time check. We can achieve a dynamic threshold of the interval and provide correction through this simple control system. Each time we can increase and decrease with a fixed value of time. Later, on the conclusion we mention a different way, for future research, avoiding the fixed value in order to have a more realistic behaviour.

7.6 Messages to other modules

The module has to sent messages to some of the other modules. Generally in the present communication framework of the project a message is broadcasted and any module can listen to it. In order for the other modules to listen to it they have to be able to interpret it. The Attention module sends various messages to the platform. Some of them are not yet implemented by the synthesis. The messages sent have generally a certain form. They are generic with 2 features. The command

and the receiver feature. The command holds the information for the type of command or information sent to the agent. The receiver holds the information for the recipient of the information. It is useful if there are more than one agents available. An example of the creation of this generic message is shown below:

```
Generic generic = new Generic();
Feature receiver = new Feature();
Feature command = new Feature();
this.generic.features.Add(receiver);
this.generic.features.Add(command);
```

7.6.1 Look at user message

A "look at user" generic message is addressed to the synthesis module and proposes that the agent should move its regard towards the user's face with a certain speed. It is usually sent to move the head of the agent towards something that attracted his attention. The speed can vary depending on the speed, acceleration and amplitude of a user's detected move. It can respect a fixed threshold in order to give different speed levels. The levels can be continuous if we give a continuous value or discrete. If it is a possible configuration according to a threshold the model can provide three different discrete behaviours. Either the turn of the head, or the turn of half of the body, or the complete turn of the body towards the user in the most attentive case. An example of how his message is written for the communication is given below. The Astrid mentioned in the message is one of the two agents shown in figure 19 and will be used to test our agent.

```
this.receiver.stringValue = "COMMAND";
this.command.name = "LookAt";
this.command.stringValue = "Human";
this.command.floatValue = 5f;
this.receiver.name = "Astrid";
this.command.intValue = 1;
```

7.6.2 Stop Looking at user message

A "Stop looking at user" generic message is addressed to the synthesis module and proposes that the agent should stop paying attention to the user. It is usually sent after the agent has seen the user and there is no need to look at him again. After the reception of this message the agent can continue its performance or look around. An example of how his message is written for the communication is given below:

```
this.receiver.name = "Astrid";
this.command.intValue = 0;
```

7.6.3 Move head to user

This is the first threshold of level of response of attention for the agent. While movement is detected from the user's point of view then the agent has to rotate only its head towards the user with a certain speed. The speed can be variable and can depend on the speed of the user's movement.

This message is not yet interpreted by the synthesis and thus we omit to send it for the time being. An example of how his message is written for the communication is given below:

```
this.receiver.stringValue = "COMMAND";
this.command.name = "HeadLookAt";
this.command.stringValue = "Human";
this.command.floatValue = 5f;
this.receiver.name = "Astrid";
this.command.intValue = 2;
```

7.6.4 Move body to user

This is a second threshold of level of response of attention for the agent. While the movement of the user is more intense the agent in order to show natural behaviour should consider moving not only his head but also partly his body towards the user. Again the speed of these moves are proportional or analogous to the user's speed of motion. This message is not yet interpreted by the synthesis and thus we omit to send it for the time being. An example of how his message is written for the communication is given below:

```
this.receiver.stringValue = "COMMAND";
this.command.name = "BodyLookAt";
this.command.stringValue = "Human";
this.command.floatValue = 5f;
this.receiver.name = "Astrid";
this.command.intValue = 3;
```

7.6.5 Shift completely towards the user

This is a third threshold of level of response of attention for the agent. While the movement of the user is even more intense the agent in order to show natural behaviour should consider moving not only his head and body but to stop what it is doing and regard completely towards the user. The speed of these moves are proportional or analogous to the user's speed of motion. Again this message is not yet interpreted by the synthesis and thus we omit to send it for the time being. An example of how his message is written for the communication is given below:

```
this.receiver.stringValue = "COMMAND";
this.command.name = "CompletelyLookAt";
this.command.stringValue = "Human";
this.command.floatValue = 5f;
this.receiver.name = "Astrid";
this.command.intValue = 4;
```

7.6.6 Perform attentive behaviour

While interacting with the agent. It is possible that the user might loose interest. We do not calculate the user's interest but we do can understand if the user is trying to get distant from the agent and it falls to the agent's attention. Then the agent can stop its performance and try to attract

the user performing an attracting behaviour. The most important is to stop the performance. An example of how his message is written for the communication is given below:

```
this.receiver.stringValue = "COMMAND";
this.command.name = "Attract";
this.command.stringValue = "Human";
this.command.floatValue = 5f;
this.receiver.name = "Astrid";
this.command.intValue = 1;
```

7.6.7 User in Vision field

The Attention module has to inform the other modules that the user is visible. If the user is not visible the decision module can not take a contradictive decision by omitting this fact. In this case it is urgent to always inform the system of the user visibility. An example of how his message is written for the communication is given below:

```
this.receiver.stringValue = "INFO";
this.command.name = "UserInField";
this.command.stringValue = "Human";
this.receiver.name = "Astrid";
this.command.intValue = 1;
```

8 Algorithm of the model

Below there is a high level pseudocode of the algorithm used to calculate the attention level of the agent using a set of procedures. Each procedure is then more detailed with another pseudocode.

```
Attention Computation{
while input not over{
get user skeleton
update graphs
get agent skeleton
update graphs
get user speed, amplitude, acceleration of hands and feet motion
update graphs
calculate agent alertness for each skeleton bone
pass skeleton bones motion values from vision filter
calculate agent state
provide strategy for state
send messages to other modules if needed
}
end of while
}
end of Attention computation

calculate agent alertness for each skeleton bone{
```

```
for root bone till last bone
Alertness[bone]= Visionfunction(bone coordinates,agent skeleton)
nextbone
}
end of calculate agent alertness for each skeleton bone

pass skeleton bones motion values from vision filter{
for hand and feet
//calculate weighted average or maximum
Attention = 0.4*lefthandMotionvalues + 0.4*righthandMotionvalues +
0.1*leftfootMotionvalues + 0.1*rightfootMotionvalues
//instead of average a max calculation can be used
max calculation
Max(lefthandMotionvalues,righthandMotionvalues,leftfootMotionvalues,rightfootMotionvalues)
}
end of pass skeleton bones motion values from vision filter

calculate agent state{
if noUser then state = 0
if UserinVisionField then state = 1
if UserInteracting then state = 2
if UserVisible then state = 3
if UserNotVisible then state = 4
if UserMoving then state = 5
If UserNotMoving then state = 3
If UserMovingOut then state = 6
}
end of calculate agent state

calculate motion values for each{
for every hand and feet
Motionvalue= 0.7*speedvalue + 0.2*amplitude + 0.1*acceleration
}
end of calculate motion values for each

provide strategy for state{
if state=0 then do nothing/look around
if state=1 then show attentive behaviour
if state=2 then interact with user
if state=3 then stop looking at user, stop interleaving timer
if state=4 then start counting for the interval of interleaving between task and user
if state=5 then look at user
if state=6 then stop all actions perform attentive behaviour towards user
}
end of provide strategy for state
```

```
interleaving between task and user{
Start timer
if UserNotVisible then continue timing
if UserVisible then stop timing
if time=criticalTime and UserMoving then lookAtUser and decreaseTime of timer
if time=criticalTime and UserNotMoving then increaseTime of timer
}
end of interleaving between task and user

user out of vision field{
Start timer
if user moves more than 1 m in 2 seconds then state=6 stop timer
}
end of user out of vision field
```

9 Experimentation

In order to test the program we did some various simple tests to check the code and the expected states, the messages and the behaviour of the agent. These test include a fixed agent position test, a fixed user position test and a full interaction between user and agent test.

9.1 Various experimentation tests

- **Fixed agent position test**

  The agent's position and head rotation is kept fixed while the user is moving. That results into a perturbation of the agent's alertness to each of the users position. The below figure 15 can show how the alertness level changes for the agent in time with respect to the user's position.

- **Fixed user and agent position test** The user position is kept fixed while the agent's head rotation changes. With head rotation, agent's regard vector changes. The perturbation on the values of the graph that can be shown below show how the agent's rotation changes the result on its alertness from 0 to 1 (see figure 15.

- **Full interaction test with attentive agent** The agent is looking constantly at the user or at least the user is in the agent's vision field and the user is moving at times. We can expect that moving of the user will result to the agent sensing of the user movement while he is in its vision field. The motion graph will give some normalised values from 0 to 1 for this movement of the user. In the figure 16 below the results of such an interaction can be seen.

9.2 Unity experimentation

During the development of the Attention module there was a need to check it in every step. To see if the geometrical model is calibrated with he visual model and how it can be corrected. In order to test the model during the different phases and the different parts of it we used some extra

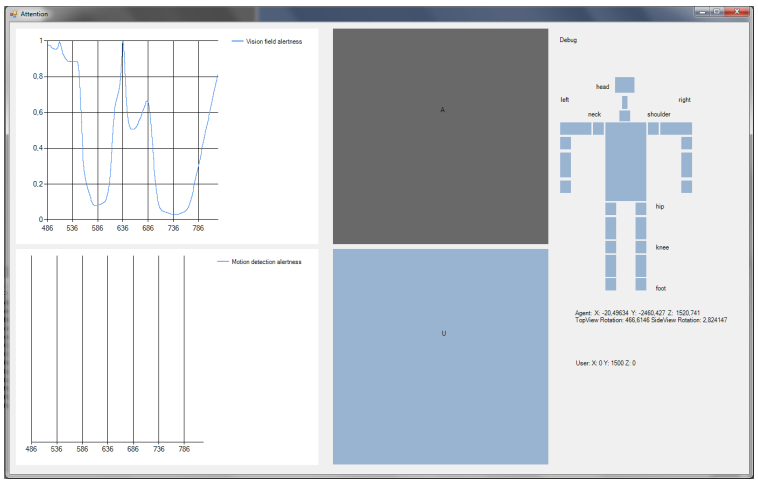

Figure 15: This figure show how the user and the agent interact in time in 2 different scenarios. First scenario that the user is changing positions and we take into account only his head position. Second scenario is that the user stays still and the agent rotates each head. The motion of the user body parts is not taken into account in this image that is why only one graph is drawn.

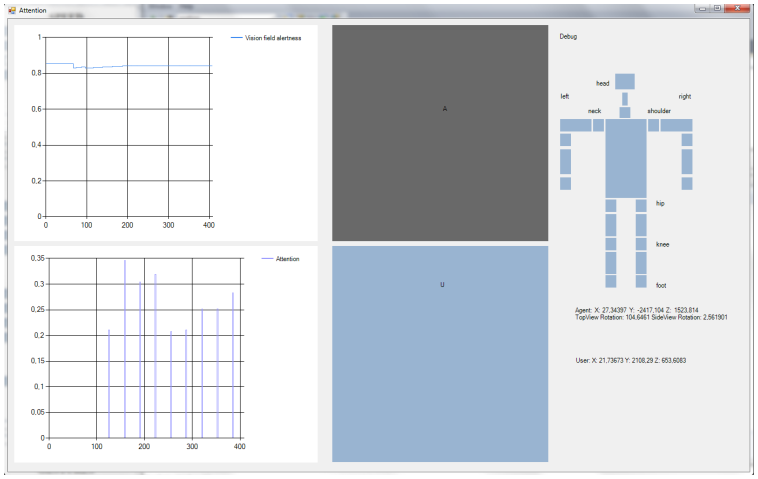

Figure 16: In this figure we can see the agent and the user moving freely. All the values of speed, amplitude and acceleration are used to compute the agent's level of attention (bottom up attention).

applications made in CSharp on Unity3d. Unity3d offers a complete environment to use in order to visualize 3d objects.

For the first experimentation an application with 2 balls was created in order to represent the heads of the user and the agent (as seen in figure 17). With the rotation of the agent's head, its alertness value was changing. The same was seen if the agent was still and the user was changing positions.

Again, for the second experimentation an application with the two balls was used. This time the user head was driven by the kinect input and there was a clear separation between the real and the virtual world (see figure 18).

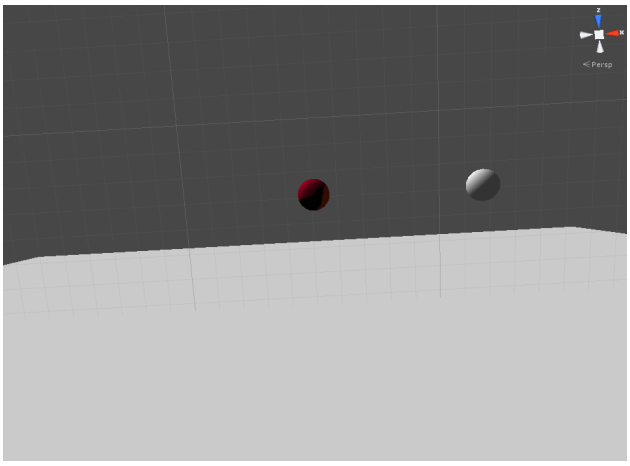

Figure 17: The user's head is represented by the grey ball.

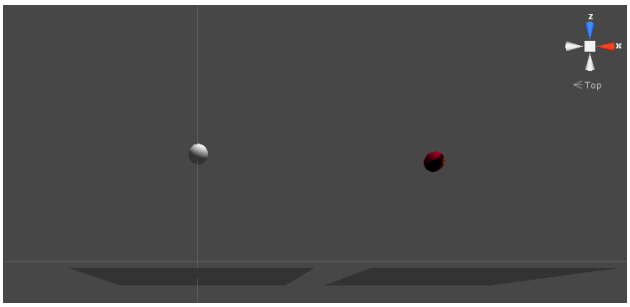

Figure 18: The user's head is represented by the grey ball again. The two worlds are separated to show the impossibility of jumping from one world to the other.

In the last experimentation phase that is not complete, we use one of the two virtual agents that can be seen in figure 19. The two agents are meant to be used in another project were the control can be passed from one agent to the other. For the moment, we use only one of the agents (either the female one named Astrid or the male one named Toto) in order to test that the agent can attend to the user with the *look at* message. What we do to test the Attention module is to run all the modules of the project and see how the agent reacts to the user input. What we should see is a response of the agent for each user gestures.

10 Conclusion

In this master research report we studied and presented the notion of attention from an interdisciplinary point of view. Moreover we discussed about the importance of providing a virtual agent with an attentional behaviour in order to make its behaviour more believable and human-like. Previous work on computational models of attention has been described. We also, describe the ideas on the conception of our model and what we implemented so far.

The idea of the whole project containing a set of modules that interact together to build the virtual agent and its capabilities have been introduced and discussed. In this context the part and

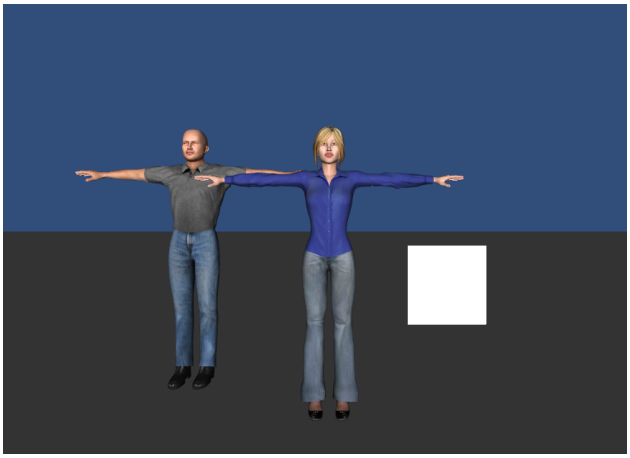

Figure 19: The two virtual agents in Unity3d. The female or the male can be used to test the module of Attention.

responsibilities of the main module of this report, the Attention module has also been presented.

The Attention module consists of a set of different responsibility parts. It is the vision field part that consists of a geometry modelling for the vision field and a probability modelling for the filter functions.

During the work there were some constraints that we needed to address. Because the present report has been written for an internship that lasted 5 months while the project is an ongoing project that has a duration of 4 years, not all of the modules were completely finished. At the same time, there was a need of team work and communication to solve technical errors not related to our module. This affected the research and testing capabilities. Some of the proposition even implemented can not yet be tested. Also, some other need to be tested more in order to see where there is space for enhancement. Moreover, because of the inability to track with the use of kinect the rotation of the user's head with a very good precision we do not use it to interpret the user's level of interest as planned. Besides, we thought that the modelling of the user's interest part in mostly a responsibility for the Decision module in order to take it into account for the agent's goals.

For the future work, i believe that an offline and an online classifier can be added in order to change the shift of attention between the task and the user. The agent can be trained to use different interleaving times in order to switch between these two tasks from different datasets. Then with an online algorithm he could be able to choose between these times in order to fit them into a better value. Ideal would be to have different initial times in a database and then cross fit them to produce new values.

Another idea is to research on the speed, amplitude, acceleration values so that they are not considered as same importance values to form the motion value needed for the motion recognition part. For the time being they were all included to form a motion value for the moving parts (each hand and foot). But probably the max value among these can be more important than the average value.

Moreover a fatigue coefficient can be added. If the agent is interacting with a user for a more extended time it is more natural to expect his attention alertness to be falling. A coefficient can be co-calculated in order to alter the vision field spectrum of the agent. A multiplication of a more narrow triangular function with any of the filter functions mentioned above can be used to achieve

it. The research interest is how to change the vision field with respect to the fatigue of the agent in real time and in a believable way.

Finally, the detection of the user keeping distant from the agent and thus moving out of the scene can be further researched. A more sophisticated way instead of just using fixed thresholds of time and distance can be used. For the project needs and specifications the simple check can be probably enough, but in a different context it might be proven inadequate. We could incorporate the same idea proposed for the interleaving between the user and the task as mentioned above. One can think of a simplified classifier that associates the user moving out of the interaction with his interest level to give a more realistic result for the agent behaviour.

11 Acknowledgements

The work was founded by the ANR INGREDIBLE project: ANR-12-CORD-001 http://www.ingredible.fr. I would like to personally thank my supervisor, Elisabetta Bevacqua, for her experience in creating Virtual Agents and for her help to create them in Unity for test use.

References

[Broadbent, 1954] Broadbent, D. (1954). The role of auditory localization in attention and memory span. *Journal of Experimental Psychology*, 47:191–196.

[Chopra Khullar and Badler, 2001] Chopra Khullar, S. and Badler, N. I. (2001). Where to look? Automating attending behaviors of virtual human characters. *Autonomous agents and multi-agent systems*, 4(1-2):9–23.

[Chun and Wolfe, 2005] Chun, M. M. and Wolfe, J. (2005). Visual attention. *Blackwell Handbook of Sensation and Perception*, pages 272–310.

[Conway and Cowan, 2001] Conway, A. R. A. and Cowan, N. (2001). The cocktail party phenomenon revisited: The importance of working memory capacity. *Psychonomic Bulletin and Review*, 76(8):331–335.

[Coull et al., 2000] Coull, J. T., Frith, C. D., Büchel, C., and Nobre, a. C. (2000). Orienting attention in time: behavioural and neuroanatomical distinction between exogenous and endogenous shifts. *Neuropsychologia*, 38(6):808–19.

[Deutsch and Deutsch, 1963] Deutsch, J. A. and Deutsch, D. (1963). Attention: Some theoretical considerations. *Psychological Review*, 70:80–90.

[Eriksen and Yeh, 1985] Eriksen, C. W. and Yeh, Y. Y. (1985). Allocation of attention in the visual field. *Journal of experimental psychology. Human perception and performance*, 11(5):583–97.

[Globerson, 1983] Globerson, T. (1983). Mental capacity, mental effort, and cognitive style. *Developmental Review*, 107(3):292–302.

[Goertzel et al., 2010] Goertzel, B., Lian, R., Arel, I., de Garis, H., and Chen, S. (2010). A world survey of artificial brain projects, Part II: Biologically inspired cognitive architectures. *Neurocomputing*, 74(1-3):30–49.

[Hill, 1999] Hill, R. (May 1999). Modeling perceptual attention in virtual humans. *Proceedings of the 8th Conference on Computer Generated Forces and Behavioral Representation, SISO, Orlando, Fla*, pages 563–573.

[Itti et al., 2003] Itti, L., Dhavale, N., and Pighin, F. (2003). Realistic avatar eye and head animation using a neurobiological model of visual attention. In *Proceedings of the SPIE 48th annual international symposium on optical science and technology*, pages 64–78, San Diego, USA.

[James, 1890] James, W. (1890). *The Principles of Psychology*, chapter 11, page 404. Henry Holt and Co.

[Johnston and Heinz, 1978] Johnston, W. A. and Heinz, S. P. (1978). Flexibility and capacity demands of attention. *Journal of Experimental Psychology 107*, 107(1):420–435.

[Kaplan and Hafner, 2006] Kaplan, F. and Hafner, V. (2006). The challenges of joint attention. *Interaction Studies*, 7(2):135–169.

[Kim et al., 2005] Kim, Y., Hill, R. W., and Traum, D. (2005). A computational model of dynamic perceptual attention for virtual humans. In *Proceedings of the 14th Conference on Behavior Representation in Modeling and Simulation*, Universal City, CA.

[Klein, 1996] Klein, R. M. (1996). Attention: Yesterday, today, and tomorrow. *The American Journal of Psychology*, 109(1):139–150.

[Kokkinara et al., 2011] Kokkinara, E., Oyekoya, O., and Steed, A. (2011). Modelling selective visual attention for autonomous virtual characters. *Computer Animation and Virtual Worlds*, 22(4):361–369.

[Mayer et al., 2004] Mayer, A. R., Dorflinger, J. M., Rao, S. M., and Seidenberg, M. (2004). Neural networks underlying endogenous and exogenous visualspatial orienting. *NeuroImage*, 23(2):534 – 541.

[Mundy and Newell, 2007] Mundy, P. and Newell, L. (2007). Attention, Joint Attention, and Social Cognition. *Current directions in psychological science*, 16(5):269–274.

[Nothegger et al., 2004] Nothegger, C., Winter, S., and Raubal, M. (2004). Selection of salient features for route directions. *Spatial Cognition and Computation*, 4(2):113–136.

[Oikonomopoulos et al., 2006] Oikonomopoulos, A., Patras, I., and Pantic, M. (2006). Spatiotemporal salient points for visual recognition of human actions. *IEEE Transactions on Systems, Man and Cybernetics - Part B*, 36(3):710–719.

[Oyekoya et al., 2009] Oyekoya, O., Steptoe, W., and Steed, A. (NY 2009). A saliency-based method of simulating visual attention in virtual scenes. *In Proceedings of the Virtual Reality Software and Technology. ACM Press*, pages 199–206.

[Peters et al., 2009] Peters, C., Asteriadis, S., and Karpouzis, K. (2009). Investigating shared attention with a virtual agent using a gaze-based interface. *Journal on Multimodal User Interfaces*, 3(1-2):119–130.

[Peters et al., 2011] Peters, C., Castellano, G., Rehm, M., André, E., Raouzaiou, A., Rapantzikos, K., Karpouzis, K., Volpe, G., Camurri, A., and Vasalou, A. (2011). Fundamentals of Agent Perception and Attention Modelling. *Emotion-Oriented Systems*, pages 293–319.

[Peters and O'Sullivan, 2003] Peters, C. and O'Sullivan, C. (2003). Bottom-up visual attention for virtual human animation. In *16th International Conference on Computer Animation and Social Agents (CASA)*, pages 111–117. IEEE Computer Society.

[Peters et al., 2005] Peters, C., Pelachaud, C., Bevacqua, E., Mancini, M., and Poggi, I. (2005). A model of attention and interest using gaze behavior. In Panayiotopoulos, T., Gratch, J., Aylett, R., Ballin, D., Olivier, P., and Rist, T., editors, *Intelligent Virtual Agents*, volume 3661 of *Lecture Notes in Computer Science*, pages 229–240. Springer Berlin Heidelberg.

[Peters and Itti, 2007] Peters, R. J. and Itti, L. (2007). Beyond bottom-up: Incorporating task-dependent influences into a computational model of spatial attention. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE.

[Pfeiffer and Wachsmuth, 2008] Pfeiffer, N. and Wachsmuth, I. (2008). Toward alignment with a virtual human - achieving joint attention. In Dengel, A., Berns, K., Breuel, T., Bomarius, F., and Roth-Berghofer, T., editors, *KI 2008: Advances in Artificial Intelligence*, volume 5243 of *Lecture Notes in Computer Science*, pages 292–299. Springer Berlin Heidelberg.

[Poggi, 2001] Poggi, I. (2001). Mind markers. *The Semantics and Pragmatics of Everyday Gestures. Berlin Verlag Arno Spitz*.

[Posner, 1980] Posner, M. I. (1980). Orienting of attention. *Quarterly Journal of Experimental Psychology*, 32(1):3–25.

[Posner and Rothbart, 1992] Posner, M. I. and Rothbart, M. K. (1992). Attentional mechanisms and conscious experience. *The Neuropsychology of Consciousness*.

[Raidt et al., 2005] Raidt, S., Bailly, G., and Elisei, F. (2005). Basic components of a face-to-face interaction with a conversational agent: Mutual attention and deixis. In *Proceedings of the 2005 Joint Conference on Smart Objects and Ambient Intelligence: Innovative Context-aware Services: Usages and Technologies*, sOc-EUSAI '05, pages 247–252, New York, NY, USA. ACM.

[Rapantzikos et al., 2005] Rapantzikos, K., Avrithis, Y., and Kollias, S. (2005). Handling uncertainty in video analysis with spatiotemporal visual attention. In *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 05)*, Reno, Nevada.

[Ratey, 2001] Ratey, J. (2001). *A User's Guide to the Brain*, chapter 3, page 114. New York: Pantheon Books.

[Theeuwes, 1991] Theeuwes, J. (1991). Exogenous and endogenous control of attention: the effect of visual onsets and offsets. *Perception and psychophysics*, 49(1):83–90.

[Treisman, 1969] Treisman, A. (1969). Strategies and models of selective attention. *Psychological Review*, 76(3):282–299.

[Trias et al., 1996] Trias, T. S., Chopra, S., Reich, B. D., Moore, M. B., Badler, N. I., Webber, B. L., and Geib, C. W. (1996). Decision networks for integrating the behaviors of virtual agents and avatars. *Proceedings of the IEEE 1996 Virtual Reality Annual International Symposium*, pages 156–162.

[Viola and Jones, 2001] Viola, P. and Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 511–518.

[Wolfe, 2006] Wolfe, J. M. (2006). Guided Search 4.0 Current Progress With a Model of Visual Search. *Journal of Vision*, pages 99–120.