Electronics and Computer Engineering Dept.

Technical University of Crete

A real time selective rendering algorithm based on spatial cognition

Alexandros Zotos

A thesis submitted in fulfillment of the requirements for the degree of Master of Science in Electronic and Computer Engineering.

Department of Electronic and Computer Engineering.

Laboratory of Distributed Multimedia Information Systems and Applications *MUSIC*

CHANIA 2009

Abstract

One of the central goals in computer graphics is to produce the best *perceived* image in the least amount of time. Advanced rendering techniques such as ray-tracing and global illumination improve image quality, but at a commensurate cost. In many cases, we end up spending significant effort improving details the viewer will never notice. If we can find a way to apply rendering effort selectively to a small number of perceptually significant regions in a given scene, we can improve the perceived quality of synthetic real-time imagery without paying the full computational price.

We use the term *selective rendering* to refer to rendering systems that allow a choice of flexible computation within the rendering framework. Techniques that we consider as selective rendering do not necessarily need to exploit visual attention. The human visual system (HVS) provides us with our visual sensory input, but while the HVS is good, it is not perfect. If we wish to invoke, in a virtual environment, the same perceptual response in viewers as if they were actually there in the real scene, then only those parts of an environment that are perceptually important to a viewer at any point of time need to be calculated at the highest quality. The remainder of the image can be calculated at a much lower quality and with much less computational expense, without the user being aware of this quality difference.

In order to produce photorealistic images from a human rather than a machine point of view, perception principles have been incorporated into rendering algorithms. In order to economize on rendering computation, *selective rendering* guides high level of detail to specific regions of a synthetic scene and lower quality to the remaining scene, without compromising the level of information transmitted. Scene regions that have been rendered in low and high quality can be combined to form one complete scene. According to previous research efforts, such decisions may be guided by predictive attention modeling, gaze or task-based information.

We propose a novel selective rendering approach which is *task* and *gaze* **independent**, simulating cognitive creation of spatial hypotheses. Scene objects are rendered in varying polygon quality according to how they are associated with the context (schema) of the scene. Experimental studies in synthetic scenes have revealed that consistent objects which are expected to be found in a scene can be rendered in lower quality without affecting

information uptake. Exploiting such expectations, inconsistent items which are salient require a high level of rendering detail in order for them to be perceptually acknowledged. The contribution of this thesis is an innovative x3D-based selective rendering framework based on memory schemata and implemented through metadata enrichment.

Using classic findings from memory research in which schemata are used to explain memory processes, this work examines whether computationally more expensive scenes containing higher polygon counts and thus greater visual fidelity, add to the functional realism of an immersive virtual environment displayed on a stereo capable, head tracked Head Mounted Display, or whether they are they merely more aesthetically pleasing than their lower polygon count counterparts.

Declaration

The work in this thesis is original and no portion of the work referred to here has been submitted in support of an application for another degree or qualification of this or any other university or institution of learning.

Signed: Date:

Alexandros Zotos

Acknowledgements

To Gitsako..

'Give me a lever long enough and a fulcrum on which to place it, and I shall move the world'.

Archimedes

'Δός μοι πά στώ και ταν γάν κινήσω'

Αρχιμήδης

Publications

2009 - **Zotos, A.,** Mania, K., Mourkoussis, N. (2009). A Schema-based Selective Rendering Framework. *ACM Siggraph Symposium on Applied Perception in Graphics and Visualization 2009*, 85-92, Chania, Crete, Greece.

2009 - **Zotos, A.,** Mania, K., Mourkoussis, N. (2009). A Selective Rendering Algorithm based on Memory Schemas. Poster, *ACM Siggraph 2009*, New Orleans, USA.

2008 - Mania, K., Mourkoussis, N., **Zotos, A**. (2008). Selective Rendering based on Perceptual Importance of Scene Regions. *IEEE conference on Systems, Man and Cybernetics 2008*, 1596 – 1601, Singapore.

Contents

List of Figures

List of Tables

Computer Graphics: 1. Graphics implemented through the use of computers. 2. Methods and techniques for converting data to or from graphic displays via computers. 3. The branch of science and technology concerned with methods and techniques for converting data to or from visual presentation using computers.

American National Standard for Telecommunications, (ANST 2003).

Chapter 1

Introduction

The power of images is summarized by Confusious' saying 'One picture is worth a thousand words'. It is, therefore, hardly surprising that computer graphics has advanced rapidly in the last 50 years. This has led to an exponential growth in the power of computer hardware and the increased complexity and speed of software algorithms. In turn, this has created the ability to render highly realistic images of complex scenes in ever decreasing amounts of time.

The realism of rendered scenes has been increased by the development of lighting models, which mimic the distribution and interaction of light in an environment. These methods ensure, to a high degree, the physical accuracy of the images produced and give what is known as a photo-realistic rendering (Ward Larson and Shakespeare 1997).

Photo-realism, (Ferwerda 2003), is when an image produces the same visual response as the actual scene it is trying to re-create, for example the images in Figure 1.

Figure 1: Some examples of photo-realistic images (Ward Larson and Shakespeare 1998).

One major goal of virtual reality and computer graphics is achieving these photo-realistic images at real-time frame rates. On modern computers creating such images may take a huge amount of computational time, for example the images shown in Figure 1 took over 3 hours to render on a 1GHz Pentium processor. Although improvements in basic rendering hardware and algorithms have produced some remarkable results, it is still computationally demanding, to date, to render highly realistic imagery in real-time, especially in relation to specular effects (Chalmers and Cater 2002).

A key issue in the design of virtual environments then, is to tailor visual fidelity to fit the needs of the application. The term visual fidelity refers to the degree to which visual features in the VE conform to visual features in the real environment (Mania et al 2005). Visual fidelity mediates the mapping from the real world environment to a computer generated one (Waller et al 1998).

Ferwerda (2001) discussed three main standards of realism related to visual fidelity: physical; photorealism; and functional realism. The criterion for physical realism is that it provides the same visual stimulation as the real world scene. The synthetic image has to be an accurate point-by-point representation of the spectral radiance values of the real scene. This could be considered an overkill however, as the algorithms required are computationally expensive, current display devices are limited in capability for displaying such scenes, and the human visual system is limited in its ability to perceive all the intricacies displayed. Photorealism on the other hand produces the same visual response as the real scene even though actual physical energy depicted differs from the real scene. It generally refers to the scene being indistinguishable from a photograph of the scene. This approach enables creators of computer generated scenes to relax visual precision by taking advantage of the limitations of the human visual system. Functional realism requires transmission of the same visual information as the real world scene in the sense that the users are able make the same visual judgments and perform the same visual tasks with similar efficiency. Realism, here, refers to the fidelity of information that the image communicates.

Flight simulators provide an example of synthetic scenes which, although not physically accurate nor photorealistic, are functionally realistic since they transmit the same visual information to users as if flying a real plane. The goal is to engineer a simulator of high functional realism so that positive transfer of training is achieved when training in the simulators, meaning that task performance is better in the real-world compared to training in the simulator.

Inarguably, one of the central goals in computer graphics is to produce the best *perceived* image in the least amount of time. Advanced rendering techniques such as ray-tracing and global illumination improve image quality, but at a commensurate cost. In many cases, we end up spending significant effort improving details the viewer will never notice. If we can find a way to apply our effort selectively to the small number of regions a viewer attends in a given scene, we can improve the perceived quality without paying the full computational price.

We use the term *selective rendering* to refer to rendering systems that allow a choice of flexible computation within the rendering framework. In order to economize on rendering computation, selective rendering guides high level of detail to specific regions of a synthetic scene and lower quality to the remaining scene, without compromising the level of

information transmitted. Scene regions that have been rendered in low and high quality can be combined to form one complete scene. Techniques that we consider as selective rendering do not necessarily need to exploit visual attention. The human visual system (HVS) provides us with our visual sensory input, but while the HVS is good, it is not perfect. If we wish to invoke, in a virtual environment, the same perceptual response in viewers as if they were actually there in the real scene, then only those parts of an environment that are perceptually important to a viewer need to be calculated at the highest quality. The remainder of the image can be calculated at a much lower quality and with much less computational expense, without the user being aware of any quality difference.

According to previous research efforts, selective rendering may be guided by predictive attention modeling, gaze or task-based information. Previous research suggested rendering in high quality the fovea region based on gaze information (McConcie and Loschky 1997; Watson et al. 1997). Gaze-dependent rendering can economize on computational complexity by rendering in high resolution only those parts of an image that are at the focus of attention. However, such an approach might encounter difficulties of maintaining updates of the multi-resolution display without disturbing the visual processing after a fast (~4ms) eye saccade.

Following a different approach, it has been proposed to assign selective high quality rendering in the visual angle of the fovea (2 degrees) centered on the users' task focus (Cater et al. 2003). In this manner, inattentional blindness may be exploited to accelerate rendering by reducing quality in regions unrelated to the task focus. This approach, however, cannot be applied when there is no overt task to be conducted and even when there is, we cannot predict exactly where each task-relevant saccade will land. Haber et al. (2001) suggested rendering the informative areas of a scene in varying quality based on saliency models, where the most salient areas are rendered at higher quality. Such bottom-up visual attention models do not always predict attention regions in a reliable manner (Marmitt and Duchowski 2002). Indeed, Land (1999) argues that salience models do not account for any fixations when carrying out a real-world task such as making a cup of tea. Correlation between actual human and artificially presented scanpaths was found to be much lower than predicted. Moreover, we have no generally accepted model of comparing scanpaths.

We propose a novel selective rendering approach which is *task* and *gaze* **independent**, simulating cognitive creation of spatial hypotheses. Scene objects are rendered in varying polygon quality according to how they are associated with the context (schema) of the

scene. Experimental studies in synthetic scenes have revealed that consistent objects which are expected to be found in a scene can be rendered in lower quality without affecting information uptake. Exploiting such expectations, inconsistent items which are salient require a high level of rendering detail in order for them to be perceptually acknowledged. The contribution of this thesis is an innovative x3D-based selective rendering framework based on memory schemata and implemented through metadata enrichment (Zotos, Mania, Mourkoussis 2009).

Of particular interest to this thesis, in order to apply selective rendering techniques, is the role that memory schema plays in the memory process. Schemata are knowledge structures of cognitive frameworks based on past experience which facilitate the interpretation of events and influence memory retrieval (Brewer et al. 1981).

The assumption is that an individual's prior experience will influence how he or she perceives, comprehends and remembers new information in a scene. Schema consistent information is what is expected to be found in a given context. Minsky (1975) suggested that expectation based on prior knowledge or experience already existent in memory, can trigger the brain to pre-load existing spatial knowledge when encountering a familiar scene. Inferences of absent elements of a space are said to occur when memory performance after exposure to a space contains spatial information that was not actually there but was expected to be there based on past associations (Mania et al 2005). In other words, information slots which have not been filled with perceptual information are filled by default assignment based on stereotypical expectations (Kuipers 1975). Such an effect was demonstrated by the results from Kuiper's 1975 study: upon glancing upon a clock on a wall, hands may be assigned during a memory test even though no hands where actually present on that particular clock. In a computer graphics content, hands are rendered in lower quality since they are 'expected' to be found, exploiting such preconceptions to save rendering computation. Schema research has generally shown that memory performance is frequently influenced by schema-based expectations. We will present a novel selective rendering system that will exploit schema theory by identifying the perceptual importance of scene regions. Objects that have been rendered in low and high quality are incorporated in a scene based on schema expectations. The rendered quality of these objects will change in real time, dependent on user navigation and interaction. The selective rendering framework is based on metadata enrichment and X3D technologies (X3D).

The **Selective Renderer Software Engineering Framework to be presented** has one major objective, to economize on rendering computation. This will be accomplished by initially producing objects of varying polygon count based on whether an object is consistent with the content of the scene. We assign high quality of polygon count to inconsistent objects in the scene, e.g. objects which are salient in a given scene and lower quality to consistent objects, e.g. objects which are expected to be found in a given context.

The **Selective Renderer Software Engineering Framework** is evaluated against a scene including objects of high polygon count utilizing memory recognition after exposure to the scene displayed on a stereo capable, head tracked Head Mounted Display (HMD).

1.1 Contribution

As safety and cost issues in relation to training in real world task situations are becoming increasingly complex, simulators have proven to be the most successful arena for production of synthetic environments. Within simulators such as flight simulators, flight crew can train to deal with emergency situations, can gain familiarity with new aircraft types and learn airfield specific procedures. It is argued that training, for instance, in a simulator with maximum fidelity would result in transfer equivalent to real-world training since the two environments would be indistinguishable. However, there is always a trade-off between visual/interaction fidelity and computational complexity.

Therefore, it is essential to have robust and yet efficient techniques in order to assess the fidelity of VE implementations comprising of computer graphics imagery, display technologies and 3D interaction metaphors across a range of application fields. In order to save on rendering computation of complex scenes comprising of millions of polygons selective rendering algorithms allocate varied levels of quality to selected parts of a scene based on the simulation's goals. Scene regions that have been rendered in low and high quality can be combined to form one complete scene. According to previous research efforts, such decisions may be guided by predictive attention modeling, gaze or task-based information. An entirely novel approach which is gaze and task independent, thus offering greater generality is proposed here. The basic premise is centered on identifying the relative 'importance' of objects in a scene towards communicating a specific context or 'schema' and assigning varied levels of rendering quality in terms of polygon count, shadow accuracy and texture resolution to objects according to the association of each object to the schema in context.

Building on previous studies which create a hierarchy of objects in a scene communicating a specific schema, an object-based rendering engine would be developed. This engine would render objects in a scene according to their association with the schema in context, incorporating experimental results. For instance, objects which are consistent with a given schema could be recalled better because of this association with the schema in context, therefore, they would not require a high amount of rendering computation.

The contribution of this thesis is an innovative x3D-based selective rendering framework based on memory schemata and implemented through metadata enrichment. We exploit the X3D's capability of defining metadata information in order to describe the level of consistency that every object entails in relation to the specific context of a scene's region.

1.2 Thesis Outline

This thesis is divided into a number of sections.

Chapter 2: Technical Background

This chapter introduces a set of fundamental terms in computer graphics starting with defining light and its properties, light energy, photometry and radiometry. Subsequently, computer graphics illumination models are analyzed. The following sections are focused on visual perception and its application to computer graphics rendering.

Furthermore, this chapter illustrates the complex variety of tools and equipment required to create, view and interact with immersive virtual environments. It provides background information regarding the key technologies used in this study and an overview of the technologies necessary to display and interact with immersive virtual environments.

Chapter 3: Memory Schemata

This chapter provides background information regarding the memory schema theory and discusses in detail previous research in cognitive psychology exploited in this thesis in order to form a novel selective rendering framework.

Chapter 4: A Selective Rendering Framework

Chapter 4 introduces the technical requirements of the Selective Rendering Framework which exploits the schema memory theory in order to economize in rendering computation (Mania, Mourkoussis, Zotos 2008). An innovative x3D-based selective rendering framework based on memory schemata and implemented through metadata enrichment is presented.

Chapter 5: Implementation of a Selective Rendering System

In this chapter, a real time selective rendering application implemented through the *Selective Rendering Framework* (Zotos, Mania, Mourkoussis, 2009 a & b) is described in detail. Objects that have been rendered in low and high quality are incorporated in a scene based on schema expectations. The rendered quality of these objects will change in real time, dependent on user navigation and interaction.

Chapter 6: Evaluation

In this chapter, the real-time Selective Framework presented in Chapter 5 is evaluated against a fully-fledged high polygon count computer graphics scene, employing memory recognition tasks.

Chapter 7: Conclusions and Future Work

Finally, the results and contributions of this thesis are presented. Future work, unveiled by relevant conclusions is suggested.

"What you see is news, what you know is background, what you feel is opinion"
Lester Markel

Chapter 2

Technical Background

A virtual environment (VE) is a computer simulated scene which can be interactively manipulated by users. Typical scenes used in such environments generally comprise of geometric models, shades, images and lights which are converted into final images through the rendering process. The rendering process must be conducted in real time in order to

provide scenes which are updated in a reacting to user interaction. An immersive virtual environment (IVE) perpetually surrounds the user within the VE. The first sections of this chapter present a set of fundamental terms of computer graphics. The third section proceeds by describing the shortcoming of previous selective rendering approaches. Finally, the last sections analyze the complex variety of tools and technologies which have been used in this thesis.

2.1 Computer Graphics Rendering

As described in Chapter 1 there is a great need for realistic rendering of computer graphical images in real time. The term 'realistic' is used broadly to refer to an image that captures and displays the effects of light interacting with physical objects occurring in real environments looking perceptually plausible to the human eye, e.g. as a painting, a photograph or a computer generated image (Figure 2). When eating in seafood restaurants, if a meal smells like fish, it is not good fish. The same principle applies in computer graphics: if it looks like computer graphics, it is not good computer graphics (Birn 2000).

Figure 2: The goal of realistic image synthesis: an example from photography.

There are no previously agreed-upon standards for measuring the actual realism of computer-generated images. In many cases, physical accuracy may be used as the standard to be achieved. Perceptual criteria are significant rather than physics based simulations to the point of undetermined 'looks good' evaluations. Ferwerda (2003) proposes three levels of realism requiring consideration when evaluating computer graphical images. These are *physical* realism, in which the synthetic scene is an accurate point-by-point representation of the spectral radiance values of the real scene; *photorealism*, in which the synthetic scene produces the same visual response as the real scene even if the physical energy depicted from the image is different compared to the real scene; and

finally *functional realism*, in which the same information is transmitted in real and synthetic scenes while users perform visual tasks targeting transfer of training in the real world (Ferwerda 2003). Physical accuracy of light and geometry does not guarantee that the displayed images will seem real. The challenge is to devise real time rendering methods which will produce perceptually accurate synthetic scenes in real – time. This chapter describes how different rendering models produce their realistic graphical images.

2.1.1 The Physical Behavior of the Light

Light is one form of *electromagnetic radiation*, a mode of propagation of energy through space that includes radio waves, radiant heat, gamma rays and X-rays. One way in which the nature of electromagnetic radiation can be pictured is as a pattern of waves propagated through an imaginary medium. The term 'visible light' is used to describe the subset of the spectrum of electromagnetic energy to which the human eye is sensitive. This subset, usually referred to as the *visual range* or the *visual band* consists of electromagnetic energy with wavelengths in the range of 380 to 780 nanometers, although the human eye has very low sensitivity to a wider range of wavelengths, including the infrared and ultraviolet ranges. The range of visible light is shown in Figure 3. As shown, the wavelength at which the human eye is most sensitive is 555 nm.

In the field of computer graphics three types of light interaction are primarily considered: *absorption, reflection and transmission*. In the case of absorption, an incident photon is removed from the simulation with no further contribution to the illumination within the environment. Reflection considers incident light that is propagated from a surface back into the scene and transmission describes light that travels through the material upon which it is incident and can then return to the environment, often from another surface of the same physical object. Both reflection and transmission can be subdivided into three main types:

Specular: When the incident light is propagated without scattering as if reflected from a mirror or transmitted through glass.

Diffuse: When incident light is scattered in all directions.

Glossy: This is a weighted combination of diffuse and specular.
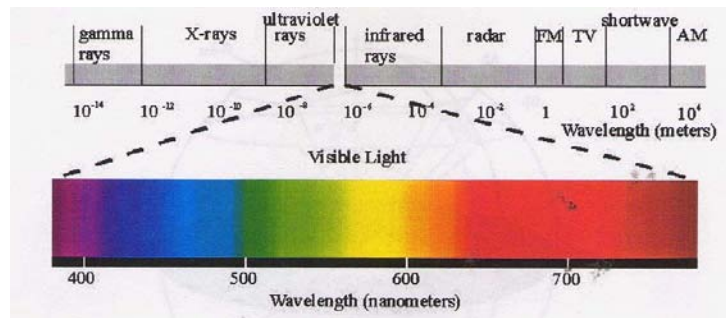
Figure 3: The visible portion of the electromagnetic spectrum.

Most materials do not fall exactly into one of the material categories described above but instead exhibit a combination of specular and diffuse characteristics.

In order to create shaded images of three dimensional objects, we should analyze in detail how the light energy interacts with a surface. Such processes may include emission, transmission, absorption, refraction, interference and reflection of light (Palmer 1999).

- **Emission** is when light is emitted from an object or surface, for example the sun or man-made sources, such as candles or light bulbs. Emitted light is composed of photons generated by the matter emitting the light; it is therefore an intrinsic source of light.

- **Transmission** describes a particular frequency of light that travels through a material returning into the environment unchanged as shown in Figure 4. As a result, the material will be transparent to that frequency of light. Most materials are transparent to some frequencies, but not to others. For example, high frequency light rays, such as gamma rays and X-rays, will pass through ordinary glass, but the lower frequencies of ultraviolet and infrared light will not.
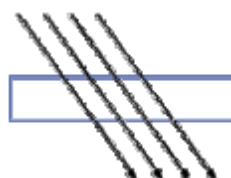
Figure 4: Light transmitted through a material.

- **Absorption** describes light as it passes through matter resulting in a decrease in its intensity as shown in Figure 5, i.e. some of the light has been absorbed by the

object. An incident photon can be completely removed from the simulation with no further contribution to the illumination within the environment if the absorption is great enough.
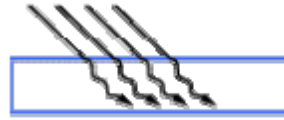
Figure 5: Light absorbed by a material.

- **Refraction** describes the bending of a light ray when it crosses the boundary between two different materials as shown in Figure 6. This change in direction is due to a change in speed. Light travels fastest in empty space and slows down upon entering matter. The refractive index of a substance is the ratio of the speed of light in space (or in air) to its speed in the substance. This ratio is always greater than one.
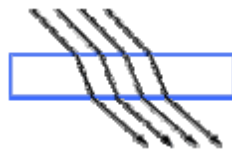
Figure 6: Light refracted through a material.

- **Interference** is an effect that occurs when two waves of equal frequency are superimposed. This often happens when light rays from a single source travel by different paths to the same point. If, at the point of meeting, the two waves are in phase (the crest of one coincides with the crest of the other), they will combine to form a new wave of the same frequency, however the amplitude of this new wave is the sum of the amplitudes of the original waves. The process of forming this new wave is called *constructive interference* (Flavios 2004). If the two waves meet out of phase (a crest of one wave coincides with a trough of the other), the result is a wave whose amplitude is the difference of the original amplitudes. This process is called *destructive interference* (Flavios 2004). If the original waves have equal amplitudes, they may completely destroy each other, leaving no wave at all. Constructive interference results in a bright spot; destructive interference produces a dark spot.

- **Reflection** considers incident light that is propagated from a surface back into the scene. Reflection depends on the smoothness of the material's surface relative to the wavelength of the radiation (ME 2004). A rough surface will affect both the relative direction and the phase coherency of the reflected wave. Thus, this characteristic determines both the amount of radiation that is reflected back to the first medium and the purity of the information that is preserved in the reflected wave. A reflected wave that maintains the geometrical organization of the incident radiation and produces a mirror image of the wave is called a *specular reflection*, as can be seen in Figure 7.

Figure 7: Light reflected off a material in different ways, from left to right, specular, diffuse, mixed, retro-reflection and finally gloss (Katedros 2004).

2.1.2 Computer Graphics Illumination Models

An illumination model computes the color at a point in terms of light directly emitted by the light source(s). A *local illumination model* calculates the distribution of light that comes directly from the light source(s). A *global illumination* model additionally calculates reflected light from all the surfaces in a scene which could receive light indirectly via intereflections from other surfaces. Global illumination models include, therefore, all the light interaction in a scene, allowing for soft shadows and color bleeding that contribute towards a more photorealistic image. The rendering equation expresses the light being transferred from one point to another (Kajiya 1986). Most illumination computations are approximate solutions of the rendering equation:

$$I\ (x,y) = g(x,y)\ [\ \varepsilon(x,y) +\_\ S\ p(x,\ y,z)\ I(y,z)\ dz\ ]$$

where

x,y,z are points in the environment,

I(x,y) is related to the intensity passing from y to x,

g(x,y) is a 'geometry' term that is 0 when x,y are occluded from each other and 1

otherwise,

p(x,y,z) is related to the intensity of light reflected from z to x from the surface at y, the integral is over all points on all surfaces S.

ε(x,y) is related to the intensity of light that is emitted from y to x.

Thus, the rendering equation states that the light from *y* that reaches *x* consists of light emitted by *y* itself and light scattered by *y* to *x* from all other surfaces which themselves emit light and recursively scatter light from other surfaces. The distinction between *view-dependent* rendering algorithms and *view-independent* algorithms is a significant one. *View-dependent* algorithms discretise the view plane to determine points at which to evaluate the illumination equation, given the viewer's direction, such as ray-tracing (Dingliana 2004, Glassner A.S.1995). *View-independent* algorithms discretise the environment and process it in order to provide enough information to evaluate the illumination equation at any point and from any viewing direction, such as radiosity.

Bouknight (1970) introduced one of the first models for local illumination of a surface. This included two terms, a *diffuse* term and an *ambient* term. The diffuse term is based upon the Lambertian reflection model, which makes the value of the outgoing intensity equal in every direction and proportional to the cosine of the angle between the incoming light and the surface normal. The ambient term is constant and approximates diffuse inter-object reflection. Gouraud (1971) extended this model to calculate the shading across a curved surface approximated by a polygonal mesh. His method calculated the outgoing intensities at the polygon vertices and then interpolated these values across the polygon as shown in Figure 8 (middle).

Phong (1975) introduced a more sophisticated interpolation scheme where the surface normal is interpolated across a polygon and the shading calculation is performed at every visible point, as shown in Figure 8 (right). He also introduced a *specular* term. Specular reflection is when the reflection is stronger in one viewing direction, i.e. there is a bright spot called a *specular highlight*. This is readily apparent on shiny surfaces. For an ideal reflector, such as a mirror, the angle of incidence equals the angle of specular reflection.

Although this model is not physically based, its simplicity and efficiency make it still the most commonly used local reflection model.
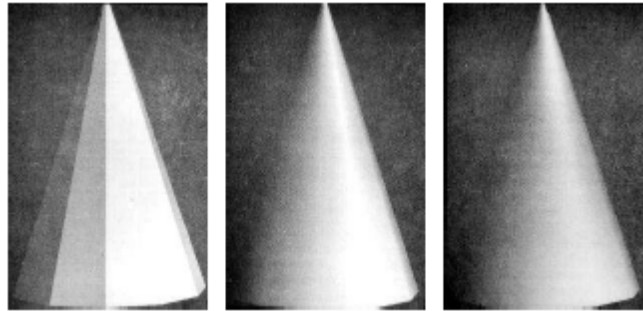
Figure 8: The differences between a simple computer generated polyhedral cone (left), linearly interpolated shading to give appearance of curvature (Gouraud Shading). Note Mach bands at edges of faces (middle) and a more complex shading calculation, interpolating curved surface normals (Phong Shading). This is necessary to eliminate Mach Bands (right).

A global illumination model adds to the local illumination model, the light that is reflected from other non-light surfaces to the current surface. A global illumination model is physically correct and produces realistic images resulting in effects such as color bleeding and soft shadows. When measured data is used for the geometry and surface properties of objects in a scene, the image produced should then be theoretically indistinguishable from reality. However, global illumination algorithms are also more computationally expensive.

Global illumination algorithms produce solutions of the rendering equation proposed by Kajiya (1986):

$$L_{out} = L_E + \int L_{In}\, f_r \cos(\theta)\, d_\theta$$

where Lout is the radiance leaving a surface, LE is the radiance emitted by the surface, LIn is the radiance of an incoming light ray arriving at the surface from light sources and other surfaces, ¦r is the bi-directional reflection distribution function of the surface, q is the angle between the surface normal and the incoming light ray and d_q is the differential solid angle around the incoming light ray.

The rendering equation is graphically depicted in Figure 9. In this figure LIn is an example of a direct light source, such as the sun or a light bulb, L'In is an example of an indirect light source i.e. light that is being reflected off another surface, R, to surface S. The light seen by the eye, Lout, is simply the integral of the indirect and direct light sources modulated by the reflectance function of the surface over the hemisphere Ω.
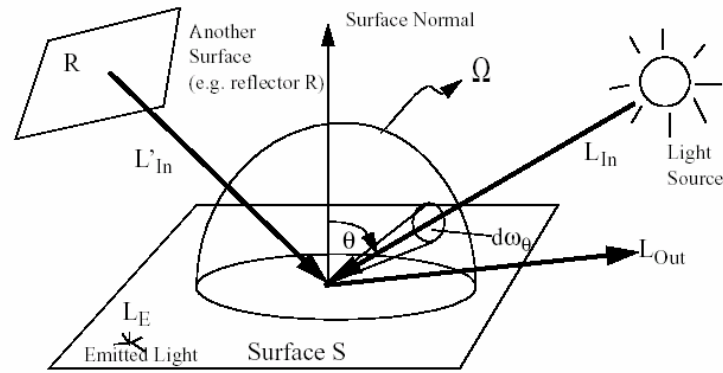
Figure 9: Graphical Depiction of the rendering equation (Yee 2000).

The problem of global illumination can be seen when you have to solve the rendering equation for each and every point in the environment. In all but the simplest case, there is no closed form solution for such an equation so it must be solved using numerical techniques and therefore this implies that there can only be an approximation of the solution (Lischinski 2003). For this reason most global illumination computations are approximate solutions to the rendering equation.

The two major types of graphics systems that use the global illumination model are *radiosity* and *ray tracing*. In this work, we utilized radiosity rendering, however, we will refer to ray-tracing in order to emphasize algorithmic differences which guided us to our choice of radiosity.

2.1.3 Ray tracing

Ray tracing is a significant global illumination algorithm which calculates specular reflections (view dependent) and results in a rendered image. Rays of light are traced from the eye through the centre of each pixel of the image plane into the scene, these are called *primary rays*. When each of these rays hits a surface it spawns two child rays, one for the reflected light and one for the refracted light. This process continues recursively for each child ray until no object is hit, or the recursion reaches some specified maximum depth. Rays are also traced to each light source from the point of intersection. These are called *shadow rays* and they account for direct illumination of the surface, as shown in Figure 10. If a shadow ray hits an object before intersecting with the light source(s), then the point under consideration is in shadow. Otherwise, there must be clear path from the point of intersection of the primary ray to the light source and thus a local illumination model can be applied to calculate the contribution of the light source(s) to that surface point.

The simple ray tracing method outlined above has several problems. Due to the recursion involved and the possibly large number of rays that may be cast, the procedure is inherently expensive. Diffuse interaction is not modeled, nor is specular interaction, other than that by perfect mirrors and filters. Surfaces receiving no direct illumination appear black. In order to overcome this, an indirect illumination term, referred to as *ambient light*, is accounted for by a constant ambient term, which is usually assigned an arbitrary value (Glassner 1989). Shadows are hard-edged and the method is very prone to aliasing. The result of ray tracing is a single image rendered for a particular position of the viewing plane, resulting in a view – dependent technique.

In ray tracing each ray must be tested for intersection with every object in the scene. Thus, for a scene of significant complexity, the method rapidly becomes impracticable. Several acceleration techniques have been developed, which may be broadly categorized into two approaches: reducing the number of rays and reducing the number of intersection tests. Hall and Greenberg noted that the intensity of each ray is reduced by each surface it hits, thus the number of rays should be stopped before any unnecessary recursion to a great depth occurs (Hall and Greenberg 1983). Another approach, which attempts to minimize the number of ray object intersections, is *spatial subdivision*. This method encloses a scene in a cube that is then partitioned into discrete regions, each of which contains a subset of the objects in the scene. Each region may then be recursively subdivided until each sub-region (voxel or cell) contains no more than a preset maximum number of objects.

Several methods for subdividing space exist. Glassner (1984) proposes the use of an *octree*, e.g. a structure where the space is bisected in each dimension, resulting in eight child regions. This subdivision is repeated for each child region until the maximum tree depth is reached, or a region contains less than a certain number of objects. Using such a framework allows for *spatial coherence*, i.e. the theory that similar objects in a scene affect neighboring pixels. Rays are traced through individual voxels, with intersection tests performed only for the objects contained within, rather than for all the objects in the scene. The ray is then processed through the voxels by determining the entry and exit points for each voxel traversed by the ray until an object is intersected or the scene boundary is reached.
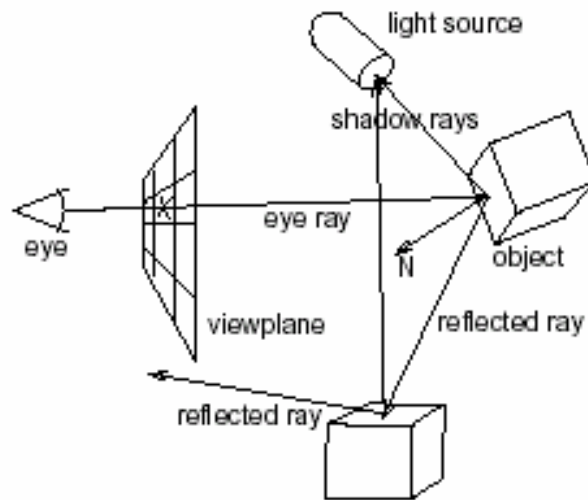
Figure 10: Ray tracing.

Ray tracing was not employed in this work because of the extreme computational demand of this method when real time scenes are produced. In order to achieve interactive frame rates, a view independent rendering framework is needed. Recent research explore methods for real time ray tracing (Mortensen, Yu, Khanna, Tecchia, Spanlang, Marino, Slater 2008).

2.1.4 Radiosity

Radiosity calculates diffuse reflections in a scene and results in a finally divided geometrical mesh. The scenes produced in this work have been rendered using radiosity calculations. The *radiosity* method of computer image generation has its basis in the field of thermal heat transfer (Goral et al. 1984). The heat transfer theory describes radiation as the transfer of energy from a surface when that surface has been thermally excited. This encompasses both surfaces that are basic emitters of energy, as with light sources and surfaces that receive energy from other surfaces and thus have energy to transfer. The thermal radiation theory can be used to describe the transfer of many kinds of energy between surfaces, including light energy.

As in thermal heat transfer, the basic radiosity method for computer image generation makes the assumption that surfaces are diffuse emitters and reflectors of energy, emitting and reflecting energy uniformly over their entire area. Thus, the radiosity of a surface is the rate at which energy leaves that surface (energy per unit time per unit area). This includes the energy emitted by the surface as well as the energy reflected from other surfaces in the scene. Light sources in the scene are treated as objects that have self emittance.
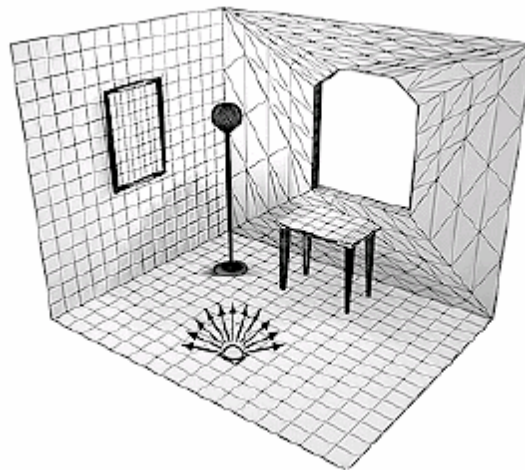
Figure 11: Radiosity (McNamara 2000).

The environment is divided into surface patches, Figure 11, each with a specified reflectivity and between each pair of patches there is a *form factor* that represents the proportion of

light leaving one patch (*patch i*) that will arrive at the other (*patch j*) (Siegel and Howell 1992).

Thus the radiosity equation is:

$$B_i = E_i + r_i S B_j F_{ij}$$

Where:

B_i = Radiosity of *patch I*

E_i = Emissivity of *patch i*

r_i= Reflectivity of *patch i*

B_j = Radiosity of *patch j*

F_{ij} = Form factor of *patch j* relative to *patch i*

Where the form factor, F_{ij}, is the fraction of energy transferred from *patch i* to *patch j* and the reciprocity relationship (Siegel and Howell 1992) states:

$$A_j F_{ji} = A_i F_{ij}$$

Where A_j and A_i are the areas of patch j and i respectively, as shown in Figure 12.
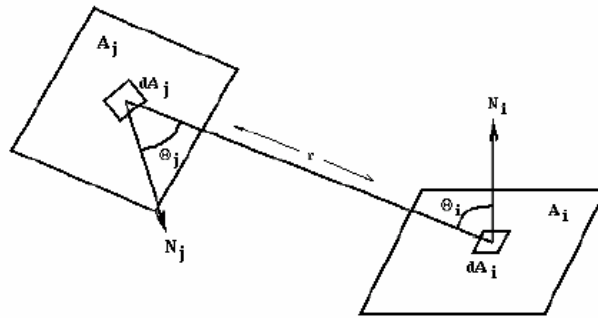
Figure 12: Relationship between two patches (Katedros 2004).

As the environment is closed, the emittance functions, reflectivity values and form factors form a system of simultaneous equations that can be solved to find the *radiosity* of each patch. The radiosity is then interpolated across each of the patches and finally the image can then be rendered.

The basic form factor equation is difficult even for simple surfaces. Nusselt (1928) developed a geometric analog that allows the simple and accurate calculation of the form factor between a surface and a point on a second surface. The *Nusselt Analog* involves placing a hemispherical projection body, with unit radius, at a point on the surface Ai. The second surface, Aj, is spherically projected onto the projection body, then cylindrically projected onto the base of the hemisphere. The form factor then may be approximated by the area projected on the base of the hemisphere divided by the area of the base of the hemisphere, as shown in Figure 13.

Cohen and Greenberg (1985) proposed that the form factor between each pair of patches could also be calculated by placing a hemi-cube on each patch and projecting the environment on to it as defined by the Nusselt Analog. Each face of the hemicube is subdivided into a set of small, usually square ('discrete') areas, each of which has a precomputed delta form factor value, as shown in Figure 14. When a surface is projected onto the hemicube, the sum of the delta form factor values of the discrete areas of the hemicube faces which are covered by the projection of the surface is the form factor between the point on the first surface (about which the cube is placed) and the second surface (the one which was projected). The speed and accuracy of this method of form factor calculation can be affected by changing the size and number of discrete areas on the faces of the hemicube.
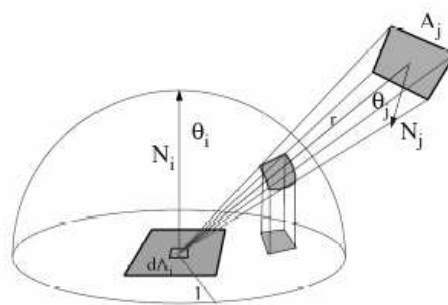
Figure 13: Nusselt's analog. The form factor from the differential area dAi to element Aj is proportional to the area of the double projection onto the base of the hemisphere (Nusselt 1928).
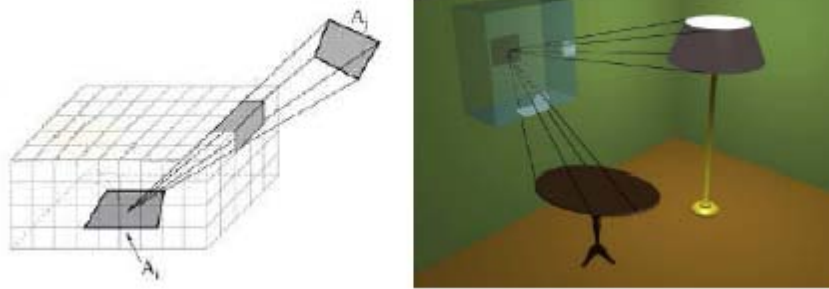
Figure 14: The hemicube (Langbein 2004).

Radiosity assumes that an equilibrium solution can be reached; that all of the energy in an environment is accounted for, through absorption and reflection. It should be noted that because of the assumption of only perfectly diffuse surfaces, the basic radiosity method is **viewpoint independent**, i.e. the solution will be the same regardless of the viewpoint of the image. The diffuse transfer of light energy between surfaces is unaffected by the position of the camera. This means that as long as the relative position of all objects and light sources remains unchanged, the radiosity values need not be recomputed for each frame. This has made the radiosity method particularly popular in architectural simulation, targeting high-quality walkthroughs of static environments. Figure 15 demonstrates the difference in image quality that can be achieved with radiosity compared to ray tracing.

However, there are several problems with using the hemicube radiosity method. It can only model diffuse reflection in a closed environment; it is limited to polygonal environments; it is prone to aliasing and has excessive time and memory requirements. Also, only after all the radiosities have been computed in the scene is the resultant image displayed. There is a form factor between each pair of patches, so in an environment with N patches, $N2$ form factors must be stored. For a scene of moderate complexity this will require a vast amount of storage and as the form factor calculation is non-trivial the time taken to produce a solution can be extensive. This means that the user is unable to alter any of the parameters of the environment until the entire computation is complete. Then once the alteration is made, the user must once again wait until the full solution is recomputed.

The visual quality of the rendered images in radiosity also strongly depends on the method employed for discretizing the scene into patches. A too fine discretization may give rise to artefacts, while with a coarse discretization, areas with high radiosity gradients may appear (Gibson and Hubbold 1997). To overcome these problems, the discretization should adapt to

the scene. That is, the interaction between two patches should account for the distance between them as well as their surface area. In other words, surfaces that are far away are discretized less finely than surfaces that are nearby. These aspects are considered by the adaptive discretization method proposed by Languénou et al. (1992). It performs both discretization and system resolution at each iteration of the shooting process, which allows for interactivity. Gibson and Hubbold (1997) demonstrated another solution for this problem by presenting an oracle that stops patch refinement once the difference between successive levels of elements becomes perceptually unnoticeable.

Progressive refinement radiosity (Cohen et al. 1988) works by not attempting to solve the entire system simultaneously. Instead, the method proceeds in a number of passes and the result converges towards the correct solution. At each pass, the patch with the greatest unshot radiosity is selected and this energy is propagated to all other patches in the environment. This is repeated until the total unshot radiosity falls below some threshold. Progressive refinement radiosity generally yields a good approximation to the full solution in far less time and with lesser storage requirements, as the form factors do not all need to be stored throughout. Many other extensions to radiosity have been developed and a very comprehensive bibliography of these techniques can be found in (Ashdown 2004).

Figure 15: The difference in image quality between ray tracing (middle) and radiosity (right hand image).

2.2 Visual Perception in Computer Graphics

In recent years, research in realistic image synthesis has included perceptually based rendering, considering aspects of the Human Visual System (HVS) to produce faster or more realistic computer graphics images. The basic goal of realistic rendering is to create images perceptually indistinguishable from real scenes. Since the human observer judges the fidelity and quality of the resulting images, the perceivable differences between the appearance of a computer graphics image and its real world counterpart should be minimized. Thus, visual perception issues are clearly involved in realistic rendering and should be considered at various stages of computation, rendering and displaying (Chalmers, A., McNamara, A., Troscianko 2000). This specific research direction has gained much attention of the computer graphics research community (Greenberg, 1999), motivated by the progress in physiology, psychophysics and psychology in providing computational models of the HVS. In this section, principles of human visual perception that are employed in this thesis are going to be reviewed. Also, research that utilizes principles of human visual perception towards efficient computer graphics algorithms and image quality metrics is going to be mentioned. It is useful to demonstrate that such knowledge is invaluable for the progress of the computer graphics field.

2.2.1 Human Visual Perception

The ability of a person or animal to detect fine spatial pattern, therefore, resolve detail in an image, is expressed as *visual acuity* (Bruce et al. 1996.). The human eye is more sensitive to intermediate changes in brightness as opposed to gradual or sudden changes. Acuity decreases with increase in distance. At any instant, the human eye samples a relatively large segment of the optic array (the *peripheral* field) with low acuity and a much smaller segment (the *central*, or *foveal* field) with high acuity. Saccadic movements are rapid jumps of eye position in order to focus to an object. Smooth and saccadic eye movements shift this high-acuity segment about rapidly so that acute vision over a wide angle is achieved. If the distance of an object from the observer changes, *convergence* movements keep it fixated by the foveal field of both eyes. The Human Field of Vision (FoV) is normally around 200 degrees. Please note that the Head Mounted Display utilized in this work allows a FoV of 50 degrees diagonal and the scenes are produced in stereo.

Perceptual constancy refers to the fact that we perceive a surface as having a constant appearance despite changes in the spectral composition of light reflected from it.

Depth perception is the ability to see the world in three dimensions. *Binocular disparity* (horizontal disparity is the distance between the two views of the two eyes) and *pictorial cues* provide depth perception information. Animals with overlapping visual fields have stereoscopic information available to them from a comparison of the images obtained at the two eyes. Each eye sees a slightly different view of the world due to the horizontal distance between the two eyes. Objects at different distances will appear to move together, or apart, reflecting the horizontal disparity between the two views. The human brain fuses the two separate views into one that is interpreted as being in 3D. The two images are called a *stereo pair*. Figure 16 shows the geometry of binocular vision. Most theories of stereo vision have argued that in analyzing binocular disparity the visual system must determine which parts of one eye's image correspond to particular parts in the other eye's image.
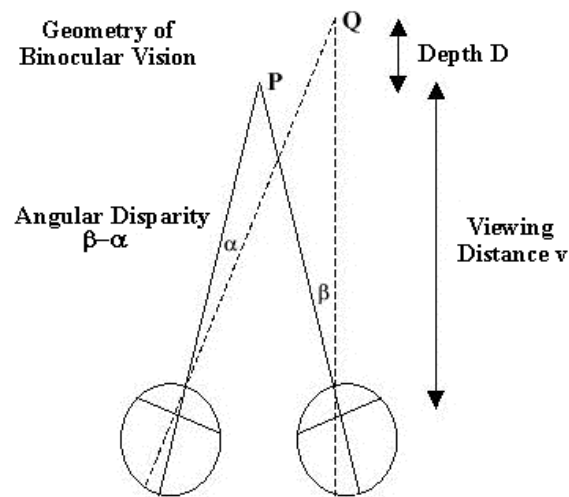
Figure 16: Geometry of binocular vision.

In Figure 16, the eyes fixate at point P. Point Q further away than P is imaged on non-corresponding or disparate points in the two eyes. The disparity (β-α) produced by point Q is measured by the discrepancy between the two image locations in degrees of visual angle. Disparity is proportional to depth, D, but inversely proportional to the squared viewing distance (Bruce et al. 1996.). For instance, if the image of Q was located 6 degrees away from the fovea in one eye but 5 degrees away in the other, then Q produced an angular disparity of 1 degree. Disparity increases with the amount of depth but decreases rapidly with increasing viewing distance.

In computer graphics/VE displays various techniques exist for providing different images to each eye including glasses with polarized filters and holography. Some of these techniques

make possible true 3D images that occupy space, rather than being projected on a single plane. These displays can provide an additional depth cue: Closer objects actually are closer just as in real life so that the viewer's eyes focus differently on different objects, depending on each object's proximity.

2.2.2 Perceptually Driven Rendering

Since global illumination solutions are costly in terms of computation, there are good prospects for their efficiency improvement by focusing computation on those scene features which can be readily perceived by the human observer under given viewing conditions. The features that are below perceptual visibility thresholds can be simply omitted from the computation without causing any perceivable difference in the final image appearance (Chalmers, McNamara et al. 2000). For instance, exploiting the poor color spatial acuity of the HVS, Meyer and Liu (1992) developed an adaptive image synthesis algorithm that uses an opponent's processing model of color vision comprising chromatic and achromatic color channels. They achieved a modest saving in computational effort and showed, using a psychophysical experiment, that decreasing the number of rays used to produce the chromatic channels had less of an effect on image quality than reducing the number of rays used to create the achromatic channels. This was the first work to attempt to minimize the computation of color calculations. Subsequently, the research community has incorporated more complex models of the HVS into the rendering algorithms (Myszkowski 1998), (Bolin et all 1998).

Another technique that addresses the problem of mapping real-world illumination to a display is *tone mapping* which takes advantage of the HVS sensitivity to relative luminance rather than absolute luminance. If the luminance of a light source is increased 10 times, the human viewer does not perceive that the actual brightness has increased 10 times. The actual relationship is logarithmic meaning that the sensitivity of the eye decreases rapidly as the luminance of the source increases. Tumblin & Rushmeier (1993) were the first who attempted to match the brightness of a real scene (vast luminances and contrast ratio) to the brightness of the computer graphics image on a typical display (limited display luminance and contrast ratio). They attempted to recreate the same perceptual response for a human viewer looking at a display as the response in the real world. For a complete review, see (Chalmers, McNamara et all 2000).

Generally, using validated visual models that could predict image fidelity, researchers can work towards greater efficiency and speed in rendering, knowing that resulting images will still be faithful visual representations to the human eye (McNamara 2000).

2.2.3 Perceptually based Image Quality Metrics

Current global illumination algorithms usually rely on energy-based metrics of solution errors, which do not necessarily correspond to the *visible* improvements of the image quality (Lischinski et al 1994.). Ideally, the development of perceptually-based error metrics which can control the accuracy of every light interaction between surfaces, are desired. This can be done by predicting the visual impact those errors may have on the perceived fidelity of the rendered images. Another approach is to develop a perceptual metric which operates directly on the rendered images. If the goal of rendering is just a still frame, then the image-based error is adequate. In the case of view-independent solutions, the application of the metric becomes more complex because a number of 'representative' views should be chosen. In practice, instead of measuring the image quality in absolute terms, it is much easier to derive a relative metric that predicts the perceived differences between a pair of images (Rushmeier et al 1995).

It is commonly known that a common mean squared-error metric usually fails in such a task. An example of advanced image fidelity metric that incorporates a complex HVS model is the Visible Differences Predictor (VDP) (Daly, 1993). Based on previous human experimentation, the VDP predicts many characteristics of human perception. The VDP metric when applied in global illumination computation provides a summary of the algorithm performance as a whole rather than giving a detailed insight into the work of its particular elements. The VDP gets as input a pair of images and it generates as output a map of probability values that characterize if these differences could be perceived by a human observer. Another perceptually based metric is the Sarnoff Visual Discrimination Model that focuses on modeling the physiology of the visual pathway (Lubin, 1995).

2.2.4 Simplification of Complex Models

Previous research has investigated how complex detail of geometrical models can be reduced without altering the viewer's recognition of the models. Maciel and Shirley's visual navigation system used texture mapped primitives to represent clusters of objects maintaining high and approximately constant frame rates (Maciel and Shirley 1995). Their approach works by ensuring that each visible object or a cluster that includes it, is drawn in each frame for the cases where there are more unoccluded primitives inside the viewing frustum than can be drawn in real-time on a workstation. Their system also supports the use of traditional *Level-Of-Detail* (LOD) representations for individual objects and supports the automatic generation of a certain type of LOD for objects and clusters of objects. The system supports the concept of choosing a representation from among those associated with an object that accounts for the direction from which the object is viewed. The system as a whole can be viewed as a generalization of the level-of-detail concept, where the entire scene is stored as a hierarchy of levels-of-detail that is traversed top-down to find a good representation for a given viewpoint. However, their system does not assume that visibility information can be extracted from the model and thus it is especially suited for outdoor environments.

Luebke et al. (2000) presented a polygonal simplification method which uses perceptual metrics to drive the local simplification operations, rather than the geometric metrics common to the other algorithms in this field. Equations derived from psychophysical studies which they ran determine whether the simplification operation will be perceptible; they then only perform the operation if its effect is judged imperceptible. To increase the range of the simplification, they use a commercial eye tracker to monitor the direction of the user's gaze allowing the image to be simplified more aggressively in the periphery than at the centre of vision. Luebke and Hallen (2001) extended this work by presenting a framework for accelerating interactive rendering produced on their psychophysical model of visual perception. In their user trial, four subjects did not perform better than chance in perceiving a difference between a rendering of a full-resolution model and a rendering of a model simplified with their algorithm in 200 trials; thus showing that perceptually driven simplification can reduce model complexity without any perceptually noticeable visual effects (Figure 17).
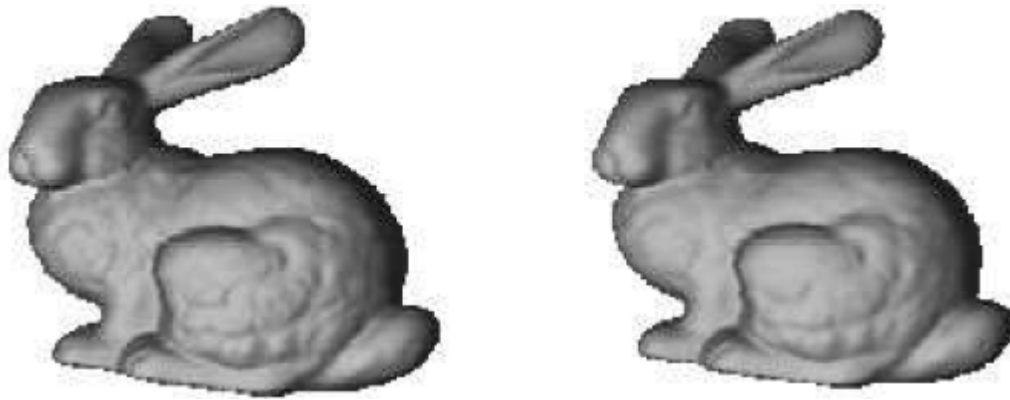
Figure 17: The original Stanford Bunny model (69,451 faces) and a simplification made by Luebke and Hallen's perceptually driven system (29,866 faces). In this view the user's gaze is 29° from the centre of the bunny (Luebke and Hallen 2001).

Marvie et al. (2003) propose a new navigation system, which is built upon a client-server framework. With their system, one can navigate through a city model, represented with procedural models that are transmitted to clients over a low bandwidth network. The geometry of the models that they produce is generated on the fly and in real time at the client side. Their navigation system relies on several different kinds of pre-processing, such as space subdivision, visibility computation, as well as a method for computing some parameters used to efficiently select the appropriate level of detail of the objects in the scene. Both deciding on the appropriate level of detail and the visibility computation are automatically performed by the graphics hardware.

Krivánek et al. (2003) devised a new fast algorithm for rendering the depth-offield effect for point-based surfaces. The algorithm handles partial occlusion correctly, does not suffer from intensity leakage and it renders depth-of-field in presence of transparent surfaces. The algorithm is new in that it exploits the level-of-detail to select the surface detail according to the amount of depth-blur applied. This makes the speed of the algorithm practically independent of the amount of depth-blur. Their proposed algorithm is an extension of the Elliptical Weighted Average (EWA) surface splatting (Heckbert 1989). It uses mathematical analysis to increase the screen space EWA surface splatting to handle depth-of-field rendering with level-of-detail.

In our system, varied polygon count versions of objects are created and stored with the help of a polygon count tool that diminishes the number of polygons of an object to the point that it is recognizable. The parameters of the polygon count tool follow:

- **Vert Percent**: The modified object's vertex count as a percentage of the overall number of vertices in the original mesh. Adjusting this setting also alters the Vert Count value.

- **Vert Count**: This is the total number of vertices in the modified object. This control is used to set the maximum number of vertices in the output mesh. Adjusting this setting also alters the Vert Percent value.

- **Max Vertex**: This parameter displays the vertex count from the original mesh that is applied.

The polygons of each object were reduced on an individual basis until it was felt that the object contained the minimum number of polygons required to retain the object's recognizable shape. This was confirmed by pilot studies.

2.3 Selective Rendering

One of the central goals in computer graphics is to produce the best *perceived* image in the least amount of time. Advanced rendering techniques such as ray-tracing and global illumination improve image quality, but at a commensurate cost. In many cases, we end up spending significant effort improving details the viewer will never notice. If we can find a way to apply our effort selectively to the small number of regions a viewer attends in a given scene, we can improve the perceived quality without paying the full computational price.

We use the term *selective rendering* to refer to rendering systems that allow a choice of flexible computation within the rendering framework. In order to economize on rendering computation, *selective rendering* guides high level of detail to specific regions of a synthetic scene and lower quality to the remaining scene, without compromising the level of information transmitted. Techniques that we consider as selective rendering do not necessarily need to exploit visual attention. The human visual system (HVS) provides us with our visual sensory input, but while the HVS is good, it is not perfect. If we wish to invoke, in a virtual environment, the same perceptual response in viewers as if they were actually there in the real scene and at the same time allow for real time interaction, then only those parts of an environment that are cognitively significant by a viewer at any point of time need to be calculated at the highest quality. The remainder of the image can be calculated at a much lower quality and with much less computational expense, without the user being aware of this quality difference.

2.3.1 Gaze based Selective Rendering

The new generation of eye-trackers replaces the unsatisfactory situation of former laboratory studies which demanded subject's immobilisation, prohibited speech, darkness, recalibration of the method after every blink etc. by ergonomically acceptable measurement of gaze direction in a head-free condition. The new methods are fast and increasingly non-invasive. This means that eye tracking no longer interferes with the activities the participant is trying to carry out. As gaze-direction is the only reliable (albeit not ideal) index of the locus of visual attention there is an almost open-ended list of possible applications of this methodology of eye tracking.

Gaze-contingent processing can be used for enhancing low-bandwidth communication, firstly by an appropriate selection of information and channels and, secondly, by transmission with high resolution of only those parts of an image which are at the focus of

attention. In this way also low-bandwidth channels can be optimally exploited, e.g. in Virtual Reality applications. There is, however, a general problem on the way to realization of most of these applications as '*not every visual fixation is filled with attention because our attention can be directed inward, on internal transformation of knowledge*' (Challis et al. 1996) and '*without attention there is no conscious perception*' (Mack and Rock 1998). This is why knowledge of the actual limits of human visual processing of information is needed to fully take advantage of this exploitation.

Osteberg's research builds on human vision work that indicates the human eye only processes detailed information from a relatively small part of the visual field (Osterberg 1935). Watson et al. (1997a; 1997b) proposed a paradigm for the design of systems that manage level of detail in virtual environments. They performed a user study to evaluate the effectiveness of high detail insets used in head-mounted displays. Subjects performed a search task with different display types. Each of these display types was a combination of two independent variables: peripheral resolution and the size of the high level of detail inset as shown in Figure 18. The high detail inset they used was rectangular and was always presented at the fine level of resolution. The level of peripheral resolution was varied at three possible levels; fine resolution (320x240), medium resolution (192x144) and coarse (64x48). There were three inset sizes; the large inset size was half the complete display's height and width, the small inset size was 30% of the complete display's height and width, the final size was no inset at all.
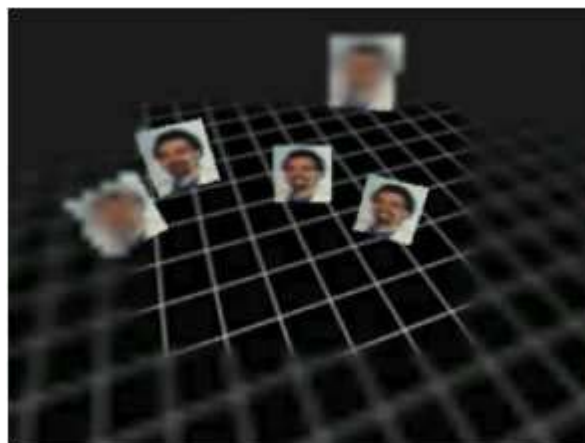
Figure 18: Experimental environment as seen with the coarse display (Watson et al. 1997b).

Their results showed observers found their search targets faster and more accurately for the fine resolution no inset condition, however it was not significantly better than the fine resolution inset displays with either medium or coarse peripheral resolutions. Thus

peripheral level of detail degradation can be a useful compromise in order to achieve desired frame rates. Watson et al. are continuing to work on measuring and predicting visual fidelity for simplifying polygonal models (Watson et al. 2000; 2001).

McConkie and Loschky (1997; 2000) and Loschky et al. (1999; 2001) instructed observers to examine complex scenes displayed on an eye linked multiple resolution display, which produces high visual resolution only in the region to which the eyes are directed (Figure 19). Image resolution and details outside this 'window' of high resolution are reduced. This significantly lowers bandwidth as many interactive single-user image display applications have prohibitively large bandwidth requirements. Their recent study measured viewers' image quality judgments and their eye movement parameters. They found that photographic images filtered with a window radius of 4.1 degrees produced results statistically indistinguishable from that of a full high-resolution display.

This approach did, however, encounter the problem of keeping up with updating the multi-resolutional display after an eye movement without disturbing the visual processing. The work has shown that the image needs to be updated after an eye saccade within 5 milliseconds of a fixation otherwise the observer will detect the low resolution. These high update rates were only achievable by using an extremely high temporal resolution eye tracker and by pre-storing all possible multi-resolutional images that were to be used.
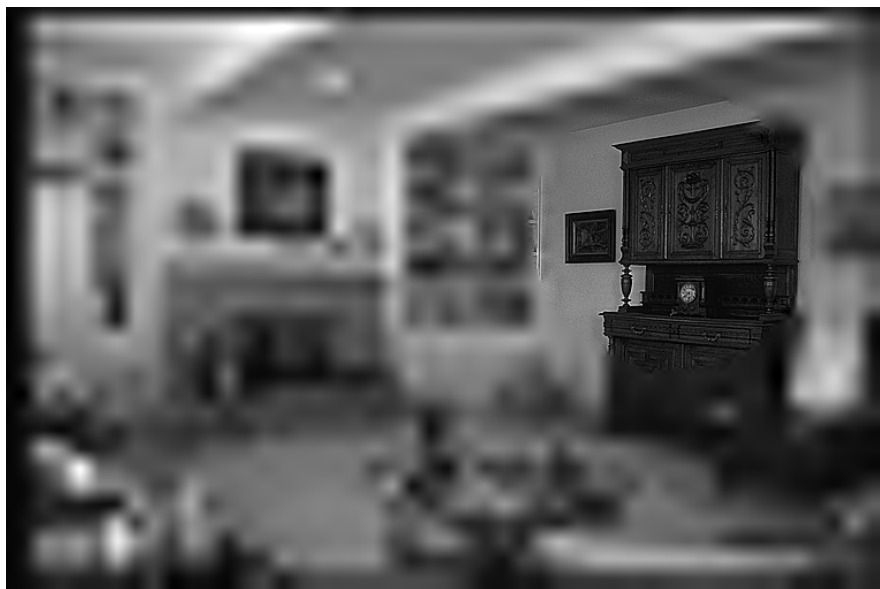
Figure 19: An example of McConkie's work with an eye linked multiple resolution display (McConkie and Loschky 1997).

2.3.2 Selective Rendering based on Saliency Models

Saliency models determine what is visually important within the whole scene. They are based on the idea first advanced by Koch and Ullman (1985) of the existence, in the brain, of a specific visual map encoding for local visual conspicuity. The purpose of the *saliency map* is to represent the conspicuity, or saliency, at every location in the visual field by a scalar quantity and to guide the selection of attended locations, based on the spatial distribution of saliency (Wooding 2002). A combination of the feature maps provides bottom-up input to the saliency map, modeled as a dynamical neural network. The system developed by Itti et al. (1998) attempts to predict given an input image in which *Regions Of Interest* (*ROIs*) in the image attention is drawn automatically and unconsciously towards them.

This biologically inspired system, takes an input image which is then decomposed into a set of multi-scale neural *feature maps* that extract local spatial discontinuities in the modalities of color, intensity and orientation (Itti et al. 1998). All feature maps are then combined into a unique scalar *saliency map* that encodes for the salience of a location in the scene irrespective of the particular feature that detected this location as conspicuous. A winner-takes-all neural network then detects the point of highest salience in the map at any given time and draws the focus of attention towards this location. In order to allow the focus of attention to shift to the next most salient target, the currently attended target is transiently inhibited in the saliency map. This is a mechanism that has been extensively studied in human psychophysics and is called the *inhibition-of-return* (Itti and Koch 2001). The interplay between winner-takes-all and inhibition-of-return ensures that the saliency map is scanned in order of decreasing saliency by the focus of attention and generates the model's output in the form of spatio-temporal attentional scanpaths, as shown in Figure 20.
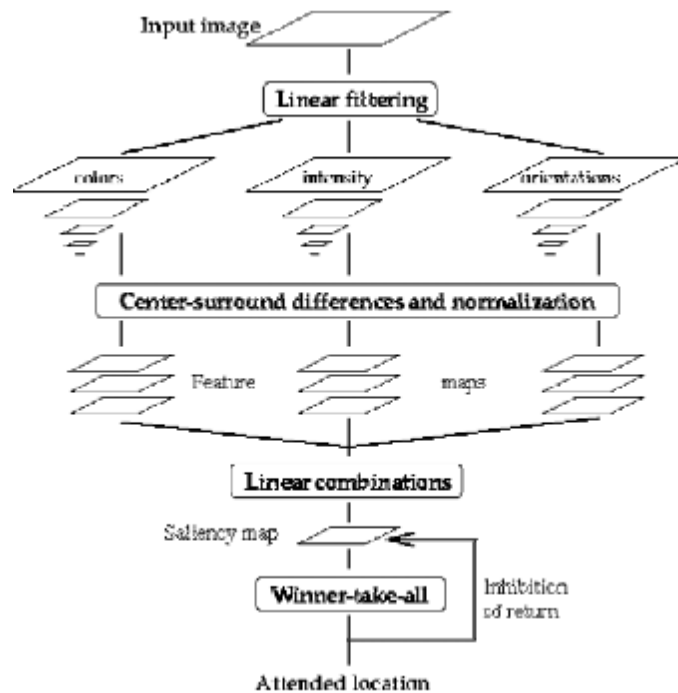
Figure 20: How the saliency map is created from the feature maps of the input image (Itti 2003a).

Marmitt and Duchowski (2002) looked at the fidelity of using such a model of visual attention to predict the visually salient features in a virtual scene. Itti et al.'s model (1998) had been previously shown to be quite accurate over still natural images (Privitera and Stark 2000); however it was not clear how well the model generalizes to the dynamic scene content presented during virtual reality immersion. Marmitt and Duchowski's analysis showed that the correlation between actual human and artificially predicted scan-paths was much lower than predicted. They hypothesis that the problem may lie in the algorithm's lack of memory, for each time the algorithm is run, as far as it is concerned, it is presented with a completely new image. In contrast, a human has already seen most parts of the image and is therefore free to distribute visual attention to new areas, even though, according to the algorithm's saliency map, the areas may appear to the model as less interesting. Therefore such systems as Yee et al. (2001) and Haber et al. (2001) propose may need some more refining to foster closer correspondence to actual human attentional patterns for use with rendering dynamic virtual reality scenes.

2.3.3 Task based Selective Rendering

Following a different approach, it has been proposed to assign selective high quality rendering in the visual angle of the fovea (2 degrees) centered on the users' task focus (Cater et all 2003). In this manner, intentional blindness may be exploited to accelerate rendering by reducing quality in regions unrelated to the task focus (Figure 21).
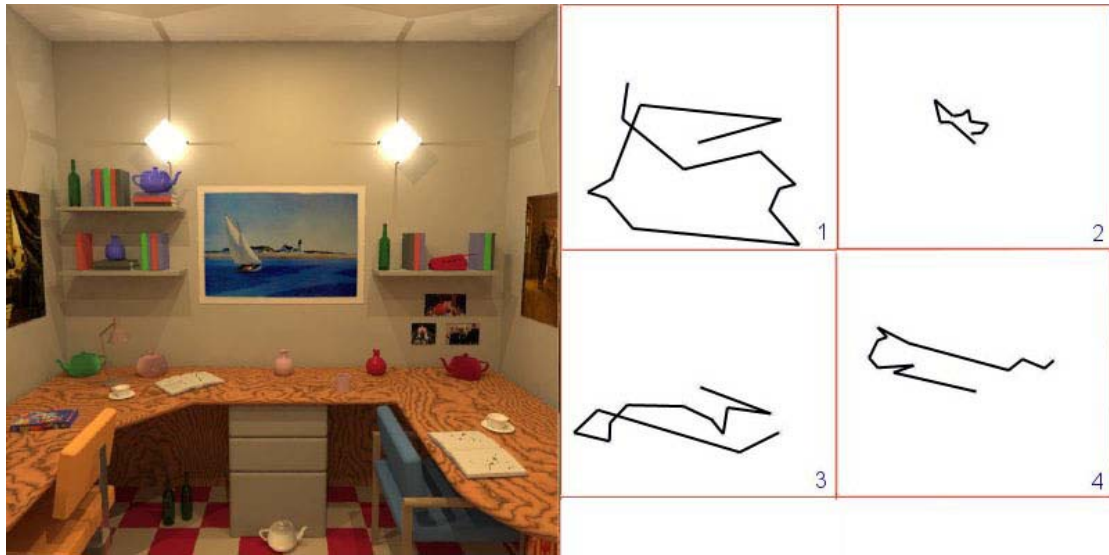
Figure 21: Effects of a task on eye movements. Eye scans for observers examined after different task instructions; 1. Free viewing, 2. Remember the central painting, 3. Remember as many objects on the table as you can, 4. Count the number of books on the shelves.

Cater et al. showed, that conspicuous objects in a scene that would normally attract the viewer's attention in an animation are ignored if they are not relevant to the task at hand. This failure of the human to see unattended items in a scene, is known as *inattentional blindness*.

The human eye is physically incapable of capturing a moving scene in full detail. Humans sense image detail only in a 2 degrees foveal region, relying on rapid eye movements, or saccades, to jump between points of interest. Human's brain then reassembles these glimpses into a coherent, but inevitably imperfect, visual percept of the environment. In the process, human literally lose sight of the unimportant details. Cater et al. demonstrated how properties of the human visual system, in particular *inattentional blindness*, can be exploited to accelerate the rendering of animated sequences by applying **a** priori knowledge of a viewer's task focus.

A controlled experimental setting demonstrated how human subjects will consistently fail to notice degradations in the quality of image details unrelated to their assigned task, even when these details fall under the viewers' gaze. Based on these observations, a perceptual rendering framework created that combines predetermined task maps with spatiotemporal contrast sensitivity to guide a progressive animation system which takes full advantage of image-based rendering techniques.

This approach, however, cannot be applied when there is no overt task to be conducted and even when there is, we cannot predict exactly where each task-relevant saccade will land.

2.3.4 Schema based Selective Rendering

Based on the above, a new approach is needed which is *task* and *gaze* **independent**, simulating cognitive creation of spatial hypotheses. Our approach does not rely on saliency model predictions which by themselves, even if correct, could be computationally demanding when operating in real-time. Saliency models are proven to be wrongly predicting the focus of attention in a variety of cases. In order to optimize rendering computation and produce photorealistic images from a human rather than a machine point of view, we incorporate information about how cognitive hypotheses are formed in a novel selective rendering framework. The memory schema theory analyzed in Chapter 3 describes how humans have the amazing capacity of merging perceptual information as seen through the eyes with previous preconceptions while making sense of real-world surroundings. In this manner, such integrated information offers the ability to recognize the context of spaces, e.g. entering a school, an office or a hospital, after a very brief exposure to that space and recognition of familiar objects associated with each context. In our system, scene objects are rendered in varying polygon quality according to how they are associated with the context (schema) of the scene.

Figure 22: Real time rendering behavior of a trashcan. Left, low quality of the trashcan is loaded. Right, high quality of the same model is loaded.

Experimental studies in synthetic scenes have revealed that consistent objects which are expected to be found in a scene can be rendered in lower quality without affecting information uptake (Mourkoussis, Mania et all. 2009). Since expected, such objects could be rendered in lower quality because of our previous knowledge, that, for instance, every office probably includes desks, office chairs and computers. Exploiting such expectations, inconsistent items which are salient require a high level of rendering detail in order for them to be perceptually acknowledged. A novel schema-based selective rendering approach is going to be analyzed in Chapter 4.

.

2.4 Virtual Reality Technology

2.4.1 Mainstream Simulation Technology

The popular view of Virtual Reality (VR) is one in which a headset or some form of eyeglasses is worn to produce a totally convincing replacement for reality. Films such as the Jonny Mnemonic (1997) have popularized such impressions, leading to the common belief that when wearing a VR headset, all five senses are somehow catered for, providing the full spectrum of human perceptual feedback. Such technology currently remains within the realms of science fiction, while existing technologies, such as those used for the research presented in Chapter 4, will be discussed in a later section.

While Virtual reality and simulation have been used almost interchangeably throughout this thesis, it is perhaps worth making a distinction between the two for clarity. It could be argued that while VR is indeed a form of simulation, it is a form that attempts to simulate not only the external environment, but also the close personal space of the user.

Mainstream simulation, such as Air Traffic Control (ATC) or Flight simulation on the other hand only attempts to model the external environment, the personal space of the user remains in reality. Certain applications lend themselves to simulation very well; Air Traffic Controllers (tower) for example work with computer systems that may be faithfully recreated within the simulator. The view from the tower window can be generated by a dedicated image generator (IG) or cluster of parallel computers and displayed using the most convenient display systems.

The operator of the simulator operates the computer systems in a way identical to a real-world control tower; there is no relevant benefit for modeling the student's personal space artificially. The operator has a continual frame of reference based in reality (Figure 23). This may account for the slightly lower incidence of 'Sim-sickness' encountered within such simulators when compared to their fully immersive VR counterparts.

Figure 23: An impression of an ATC tower simulator (using 'rear-projected' LCD), courtesy of Adacel Canada Ltd.

A dedicated and vastly expensive IG generates the external view in this simulation system and is able to create highly realistic images as shown in Figure 24.
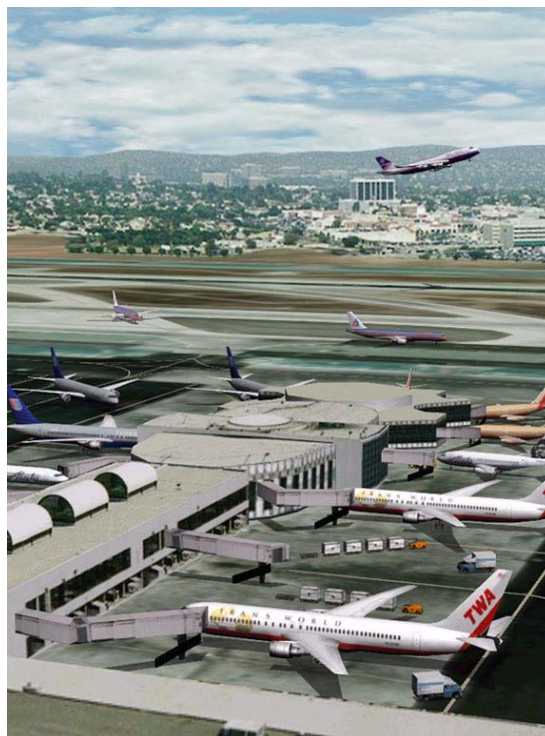
Figure 24: A near photorealistic view from a simulated control tower window, courtesy of Adacel Canada Ltd

It is worth noting that since the majority of the scene remains static with regard to the operator's viewpoint, only the geometry of the aircraft and ground vehicles needs to be recalculated and displayed in real-time.

2.4.2 Immersive VR Systems

There are however other applications for which such simulation is unsuitable. For an astronaut or diver for example, unlike a pilot or ATC controller, their personal space changes and becomes part of the external environment. Simulations for such applications must therefore include a model of the user's personal space and may not be displayed on such systems as those shown above since there is no static viewpoint. Instead, a display that blocks vision of the outside world and allows the user to see only the computer-generated environment, ideally including the personal space, is required.

An early version of such a system was created in the 1960's by Ivan Sutherland (Mania 2001), known as a CAVE and was a forerunner to modern Head Mount Displays (HMDs). An HMD, is worn on the users head and consists of one or frequently two individual display units, that are located approximately an inch in front of the user's eyes and in some cases a shield that blocks periphery vision. An example of an HMD is shown in Figure 25, as used in the research presented in Chapter 5 comprising of a Rockwell Collins headset and a 3 degrees-of-dreedom (dof) head tracker.

Figure 25 shows an HMD of the type used in this research, with no periphery shielding.

HMDs such as this may be coupled with a head-tracking device, which provides feedback to the computer system relating to the users orientation. The use of such devices allows the display to be updated to show the part of the environment relevant to the view direction of the user, providing the ability to look around the environment freely. There are currently two types of available tracking device that may be employed according to their capability of

tracking rotational movement (3 dof) and rotational as well as translational movement (6 dof).

The 3 dof tracker allows the viewpoint to be rotated around the x, y and z axis, which provides the ability to orient the viewing direction left or right, up or down, or twist it laterally. When using a 3 dof tracker only rotation is accounted for with no provision for translating the viewpoint through the environment except on pre-defined paths, so the application of such devices will typically require the subject to remain seated in a static location.

By comparison, 6 dof devices track not only the rotational position of the head, but also the physical location in space. 6 dof devices therefore allow the environment to not only be scrutinized by rotating the head, but also navigated by physically walking or moving around. It should be noted however that since the HMD will obscure the users normal view of the physical environment and because of the attached cabling, this type of interaction with the environment must be carefully thought out and does not lend itself to every application.

Since the HMD is mounted directly onto the users head, weight is of paramount importance to usability. For this reason most, if not all current HMD systems employ LCD displays. Because each display is small in terms of physical dimensions, the resolution of HMD devices is somewhat limited, typically being 640-480 (VGA) or 800-600 (SVGA). While more recent advances have lead to the production of HMD systems with a resolution of 1024-768 (XGA), environments viewed through most HMD's will appear blocky or 'pixelated', giving an unrealistic image. It is also typical in low-end devices to have a colour depth limited to 16 bits-per-pixel, which may also cause the displayed scene to appear a little unrealistic, no matter what the technology used to render the image.

Perhaps the most significant limitation of existing HMD systems however, is the lack of periphery information. While some systems employ periphery shielding, this is simply a black mask intended to prevent light from the physical world from entering the users FoV. In the real-world periphery vision exists and is used, perhaps unconsciously, in many ways. Since it is normal to perceive periphery vision, its removal must play a huge role in preventing a feeling of immersion within a VE viewed via an HMD, since most such devices present the view as if looking through an oblong tube, giving the impression of tunnel vision. It is understood that research into curved or tiled LCD systems, which would allow periphery vision to be catered for is continuing.

However, the fact that typical HMDs provide two individual LCD screens, one for each eye, provides a significant advantage over other displays systems. By providing separate, slightly varying images to each display (dual channel), binocular disparity may be used to give the impression of realistic 3D effects. Such a technique was employed during the evaluation procedures of the selective rendering framework, presented in Chapter 6.

One clear difference between a simulator in which the close personal space of a user is not modeled and one in which it is, is the way in which the immediate environment is perceived in relation to the users own body. A simulator such as that shown in Figure 23 faithfully recreates the normal working environment of the user, physical aspects of the body, such as arms and hands, can still be seen and used in the normal way. Within a system employing an HMD however, the limbs are no longer visible, with the result that it is difficult to operate any external systems that may rely on physical contact (such as a mouse or keyboard). Many methods exist to try to compensate for this, including the use of 6 dof trackers fitted to the hand and haptic user interface devices. Using the feedback provided by these trackers, a virtual representation of the hand may be included within the simulation. Despite this, manipulation of real-world objects within a VE presented through an HMD is difficult and the manipulation of virtual objects and related 3D User Interface issues are still the subject of much debate (3DUI).

In our work, VEs were presented in stereo at xVGA resolution on a Kaiser Electro-optics Pro-View 30 Head Mounted Display with a Field-of-View comprising 50 degrees diagonal. An Intersense Intertrax2, three degree of freedom tracker was utilized for rotation.

2.5 Extensible 3D Graphics (X3D)

In this section we will analyze the technical framework of the X3D language utilized for the implementation of the Selective Rendering Framework presented in this thesis. X3D is a royalty-free open standard file format and run-time architecture to represent and communicate 3D scenes and objects using XML. It is an ISO ratified standard that provides a system for the storage, retrieval and playback of real time graphics content embedded in applications, all within an open architecture to support a wide array of domains and user scenarios.

X3D has a rich set of componentized features that can tailored for use in engineering and scientific visualization, CAD and architecture, medical visualization, training and simulation, multimedia, entertainment, education and more.

The development of real-time communication of 3D data across all applications and network applications has evolved from its beginnings as the Virtual Reality Modeling Language (VRML) to the considerably more mature and refined X3D standard.

2.5.1 X3D Specifications

The X3D specifications are a highly detailed set of technical documents that define the geometry and behavior capabilities of VRML using the Web-compatible tagsets of XML. Scene graphs, nodes and fields (in X3D terminology) correspond to documents, elements and attributes (in XML terminology). As part of the Web3D Consortium, the X3D working Group has designed and implemented the next-generation X3D graphics specification (www.web3D.org/x3d). Detailed feedback and modifications by an active user community improved these results throughout the review process.

It is particularly important to note that XML benefits are numerous: XML has customized metalanguages for structuring data, is easily read by both humans and computer systems, has validatable data constraints and so on. XML is license free, platform independent and well supported (Bos 2001). Together these qualities ensure that the VRML ISO standard has been extended to functionally match the emerging family of next-generation XML-based Web languages. X3D is now part of that Web-compatible information infrastructure.

The original VRML specification was written to stand alone as a single document. While this made for a simpler reference document, the result was not easily modifiable. Growing from VRML97 to X3D 3.0 took many years of work. Because the X in X3D stands for Extensible,

there are now multiple specification documents that govern the coherent evolution and diverse capabilities of X3D. Each can be developed and extended independently, allowing annual updates that document the stable growth of X3D.

It is interesting that the primary functionality of nodes and fields in X3D graphics are specified in a technology-neutral way that is independent of any particular file encoding or programming-language binding. The X3D Abstract Specification remains the governing reference on how the X3D scene graph works. In this way, each file-format encoding and language binding is expected to remain interoperable, compatible and functionally equivalent with the rest.

Each of these specification documents has been developed through an open, collaborative process by volunteer working-group members in the nonprofit Web3D Consortium. Many of these participants are industry experts who are supported by their companies. Each year, proposed new functionality is implemented (at least twice), evaluated and specified in formal draft specifications that go to the ISO for final review and ratification. Because all X3D development, implementation and evaluation is done within the Web3D Consortium process, the overall process is relatively rapid for production of an international standard. The annual update cycle keeps this work in step with commercial product development and the always-growing capabilities of the 3D-graphics industry.

The various file encodings and language bindings consistently implement the common- core functionality of the X3D Abstract Specification. The Humanoid Animation (H-Anim) specification is also supported. Figure 26 (the honeycom diagram) illustrates how these specifications relate to each other employing a honeycomb diagram.
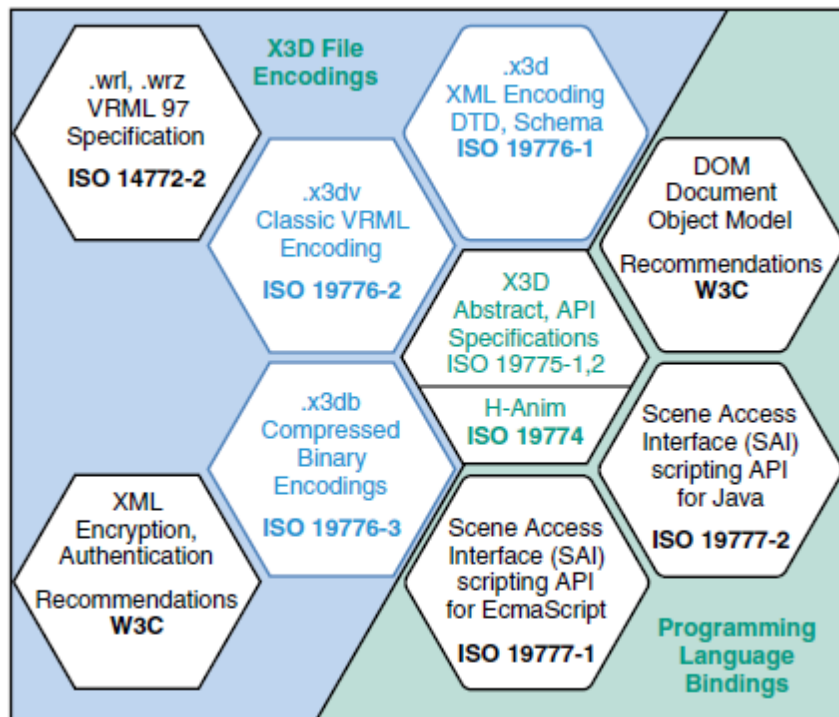
Figure 26: The family of X3D specifications includes multiple file encodings and programming-language bindings, all mapping to the same common functionality defined by the abstract specifications.

The World Wide Web Consortium (W3C) Document Object Model (DOM) is another language-neutral API designed for processing XML documents. It uses string-based accessor methods to get and set both element and attribute values. It can also be used to build and modify X3D scenes that are written in the XML encoding. However, because string-based DOM performance is usually too slow for the demands of real-time, interactive 3D graphics, DOM is rarely used directly in combination with a rendering X3D browser. Nevertheless, DOM and other XML-related APIs can be quite useful for creating, reading, or modifying X3D scenes separately from an X3D browser.

2.5.2 File Structure

The X3D scene graph is usually presented in a file, using either the .x3d XML encoding or the .x3dv classic VRML encoding. The same-graph structure is well defined and remains consistent in each encoding. Each encoding imposes its own requirements on the syntax and layout of the common representation of information.

The top-level structure of X3D files includes the following elements:

- The file header
- The X3D header statement
- The profile statement
- The component statement (optional, multiple)
- The META statement (optional, multiple)
- The X3D root node (implicit in Classic VRML encoding)
- The X3D scene graph child nodes (multiple)

In this section we will focus on the analysis of the X3D nodes that are requisite for the implementation of the selective rendering algorithm. Of particular interest to this thesis are the Switch Nodes and the Proximity Nodes.

2.5.3 Switch Node

Switch is a Grouping node that only renders one (or zero) child at a time. Like the other X3DGroupingNodes, Switch can contain zero or more X3DChildNodes as children.

The Switch node allows the scene author to switch one at a time among various sets of children nodes. Switch is used for animating geometries, providing customized level of detail and maintaining scene optimization at run time. Unlike the LOD node, the choice of the child node to use is directly controlled by the author and the user through the initial setting (and subsequent animation events) that change the *whichChoice* field. Either one or no child node is displayed by the Switch at any given moment.

The *whichChoice* field is the index of active child choice, counting from 0 for the zeroth child. The default value whichChoice="-1'' means no selection from the children and thus no rendering and whichChoice="0'' means to select the initial child.

The field definitions for SwitchNode are shown in Figure 27. The Node syntax is shown Figure 28.

Type	accessType	Name	Default	Range	Profile
SFInt32	inputOutput	whichChoice	−1	[−1,∞)	Immersive
MFNode	inputOutput	children	[]	[X3DChildNode]	Interchange
SFVec3f	initializeOnly	bboxCenter	0 0 0	(− ∞,∞)	Interchange
SFVec3f	initializeOnly	bboxSize	−1 −1 −1	[0,∞) or −1 −1 −1	Interchange
SFNode	inputOutput	metadata	NULL	[X3DMetadataObject]	Core

Figure 27: Field Definitions for Switch Node.

```
XML Syntax (.x3d)

<Switch DEF="MySwitchNode"
   whichChoice="-1"
   bboxCenter="0 0 0"
   bboxSize="-1 -1 -1">
<Group/>
   <Group/>
<Group/>
</Switch>
```

Figure 28: Node Syntax for Switch.

2.5.4 Proximity Sensor Node

The ProximitySensor node detects changes in viewer position and orientation relative to a author-defined box deliminating the active volume of the sensor. Events are sent whenever the viewer enters or exits the box. This box is not visibly rendered but can nevertheless be adjusted by parent transformations. The sensor also reports changes in viewer position, orientation and centerOfRotation when the user is within the sensed box. Sensing and sending this information to other animation nodes via ROUTE statements allows a scene to detect (and react to) user exploration within a scene.

The field definitions for the ProximitySensor are shown in Figure 29. The Node syntax is shown in Figure 30.

Type	accessType	Name	Default	Range	Profile
SFVec3f	inputOutput	center	0 0 0	$(-\infty, \infty)$	Interactive
SFVec3f	inputOutput	size	0 0 0	$[0, \infty)$	Interactive
SFBool	inputOutput	enabled	true		Interactive
SFBool	outputOnly	isActive			Interactive
SFTime	outputOnly	enterTime			Interactive
SFTime	outputOnly	exitTime			Interactive
SFVec3f	outputOnly	centerOfRotation_changed			Immersive
SFRotation	outputOnly	orientation_changed			Immersive
SFVec3f	outputOnly	position_changed			Immersive
SFNode	inputOutput	metadata	NULL		Core

Figure 29: Field Definitions for ProximitySensor.

```
XML Syntax (.x3d)

<ProximitySensor DEF="MyProximitySensor"
    center="10 20 30"
    size="40 40 40"/>
```

Figure 30: Node Syntax for ProximitySensor.

2.5.5 Metadata Node

Metadata is the use of data to explain other data. In other words, the information provided in a document might have further descriptions that explain the origin or context of the information itself.

An X3D metadata node can be used to provide such information about any node in an X3D scene graph and is placed as a child of the node that it describes. Metadata nodes are persistent, meaning that their values remain available and accessible after loading. Metadata nodes do not affect the visible rendering of a scene.

X3D documents are a growing and important part of the World Wide Web. Metadata strategies become important when X3D scenes might get reused, repurposed or improved.

The metadata nodes include MetadataDouble, MetadataFloat, MetadataInteger, MetadataString and MetadataSet. Each node can contain an array of named values, along with an optional reference that governs naming and semantic conventions. In each case, the data value is strongly typed to match the name of the node.

```
                        XML syntax (.x3d)

<MetadataSet DEF="MyMetadataSet"
      name="Metadata_name"
      reference="Standard 1.2.3c">
      <MetadataString  USE="MyMetadataString" containerField="value"/>
      <MetadataInteger USE="MyMetadataInteger" containerField="value"/>
      <MetadataFloat    USE="MyMetadataFloat" containerField="value"/>
      <MetadataFloat    name="coefficients" value="3.141592653 2.7128 1 0" containerField="value"/>
</MetadataSet>
```

Figure 31: Node Syntax for MetadataSet Node.

An X3D Hello World Example code and its representation in a browser follows:

```xml
<Scene>
  <Transform translation='-5 0 0'>
   <Shape DEF='DefaultShape'>
    <Box DEF='DefaultBox' size='2 2 2'/>
    <Appearance>
     <Material diffuseColor='1 0.2 0.2'/>
    </Appearance>
   </Shape>
  </Transform>
  <Transform translation='-2.5 0 0'>
   <Shape>
    <Cone DEF='DefaultCone' bottom='true' bottomRadius='1' height='2' side='true'/>
    <Appearance>
     <Material diffuseColor='0.2 1 0.2'/>
    </Appearance>
   </Shape>
  </Transform>
  <Transform translation='0 0 0'>
   <Shape>
    <Cylinder DEF='DefaultCylinder' bottom='true' height='2' radius='1' side='true'
top='true'/>
    <Appearance>
     <Material diffuseColor='0.2 0.2 1'/>
    </Appearance>
   </Shape>
  </Transform>
  <Transform translation='2.5 0 0'>
   <Shape>
    <Sphere DEF='DefaultSphere' radius='1'/>
    <Appearance>
     <Material diffuseColor='1 1 0.2'/>
    </Appearance>
   </Shape>
  </Transform>
  <Transform translation='4 0 0'>
   <Shape>
    <Text DEF='DefaultText' string='"hello" "X3D!"'>
     <FontStyle DEF='DefaultFontStyle'/>
    </Text>
```

Figure 32: A typical hello world .x3d example.
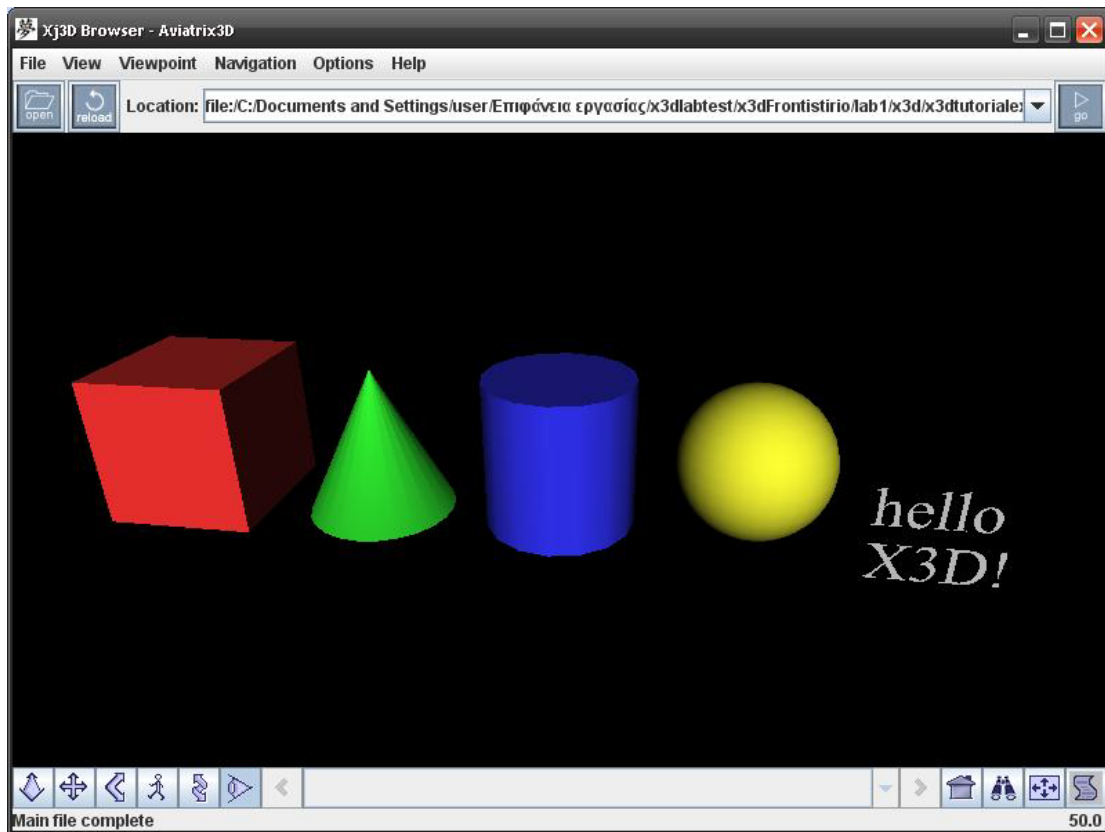
Figure 33: Viewing the x3d file using an x3d Browser (Xj3D Browser).

Figure 33 presents the five primitive types of the X3D. A box, a cone, a cylinder, a sphere and a text node are rendered using Xj3D Browser (section 2.6). The X3D code (Figure 32) of these primitive types is used in order to produce more complex X3D objects. The X3D code above, defines an X3D Box Node, an X3D Cone Node, an X3D Cylinder Node, an X3D Sphere Node and an X3D Text Node. Various fields of these nodes have also been defined, such as the size, the height, the radius etc.

2.5.6 Route Connection and Animation

The basic animation principle in X3D is simple: most field values can be modified at run time. Changes to field values are called events, each consisting of a timestamp and the update value. Some nodes have field sources that produce events and then are connected via a ROUTE to fields in target nodes. Therefore, a behavior can be defined as the act of changing one or more values within an X3D scene graph to animate 3D content.

From a browser perspective, the process of animation is accomplished by repeatedly recomputing the scene graph and then drawing a new view frame (or snapshot) in the output window. Changing views are achieved by modifying the position and orientation of objects in the scene. As one might expect, the faster an object moves in 3D coordinates relative to the camera viewpoint, the faster the motion is perceived by the user. Such animation of positions and orientations is common and works well. The same event-routing principles apply to animating any other aspect of the 3D scene: scale, color, applying texture images and even changing the active viewpoint.

Most X3D animations follow a consistent pattern, making them easy to build and understand. The basic idea is that a sensor or trigger first activates a clock node, usually a TimeSensor. The TimeSensor then sends fraction values to an interpolator, which in turn computes output values that are sent to modify the desired parameter. At each step an event is passed from one node's output field to another node's input field via a ROUTE. This process is straightforward.

Figure 34 shows the basic animation sequence. This is a common design pattern and is applicable to the majority of event-generation tasks using interpolator and sequencer nodes. The initializing TouchSensor (or similar node) that triggers the animation is optional if the TimeSensor is in loop mode. This pattern can be repeated and connected many times. Thus, sophisticated behavior responses are usually a combination of simple interpolation chains, creating a chain of activity and response. As shown in Figure 34, each arrow represents a strictly typed event, passed by a ROUTE connection, each defining output-to-input relationships between pairs of nodes. There are several details that deserve attention. The next section explains how Interpolator nodes generate a continuous stream of floating-point animation values and then describes how to build an animation chain which connects events from animation producers to animation consumers via pair-wise ROUTE definitions.

A ROUTE connects the output field of one node to the input field of another node. Ordinarily, nodes in a scene graph have any number of possible outputs. A ROUTE passes events of interest. The type of the output field must exactly match the type of the corresponding input field. This requirement is called *strict typing*. It ensures that the value of an event being delivered to the destination node is properly usable.

This approach allows authors to set up the cause-and-effect relationships which are essential to declarative programming. No imperative "do this, then do that" programming is required. Established relationships are allowed to proceed in parallel. As the simulation clock proceeds and users interact with a scene, values are generated by some nodes and sent via ROUTE connections as time-stamped events to other nodes.

This chain of events continues until no new events are generated. The full process is called an *event cascade,* commencing and completing within a single time step. A rendering update of the user's view is performed after the event cascade is complete.
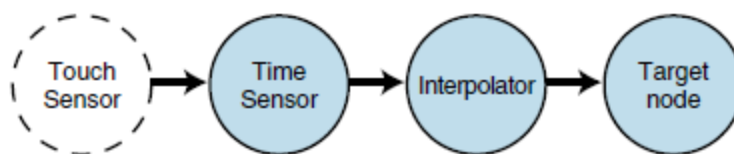
Figure 34: Event-animation chain connecting trigger, clock, event-producer node and event-consumer node. Depending on an author's design, trigger nodes (such as TouchSensor) are not always needed.

Because events are not generally passed unless a ROUTE connects two nodes, this approach is computationally efficient. That is important because such quick calculations permit high frame rate and a more interactive response to users. More than one ROUTE may connect a single output field to other nodes. This technique is called *multiple fan out*. More than one ROUTE may also expose a single input field to the output of more than one event-producing node. This technique is called *multiple fan in*. This approach is error prone and often nondeterministic, however, because one of the input events may override another if they both occur during the same timestamp. Such overloading is usually noticed as flip-flopping or unpredictable rendering in the scene.

2.6 X3D Browsers

X3D browsers are software applications that can parse (read) X3D scenes and then render (draw) them, not only showing 3D objects from varying viewpoints but also enabling object animation and user interaction. Sometimes referred to as players or viewers, X3D browsers are often implemented as plugins that work as an integrated part of a regular hypertext markup language (HTML) web browser (such as Mozilla Firefox or Internet Explorer). X3D browsers can also be delivered as standalone or embedded applications that present X3D scenes for user viewing.

Figure 35 shows a representative example of the software architecture typically used in a browser. The descriptions in this paragraph follow the blocks in a counterclockwise order (starting at the upper left).
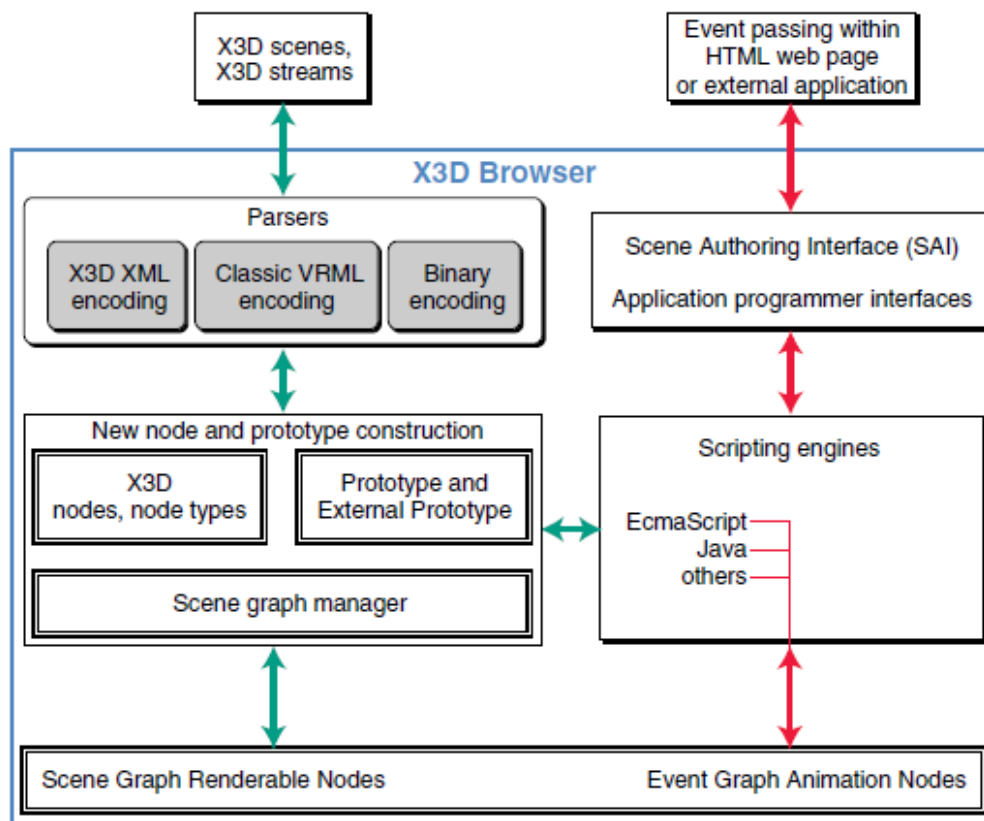
Figure 35: Example software architecture for an X3D browser.

X3D scenes are usually files that are read (or written) by the browser. Parsers are used to read the various file-format encodings available. Nodes are then created and sent to a scene-graph manager, which keeps track of defined geometry, appearance, locations and orientations. The scene-graph manager repeatedly traverses the scene-graph tree of X3D

Nodes to draw output image frames at a rapid rate. This process rapidly redraws precisely calculated perspective-based images as the user's point of view and objects of interest change. The event graph also keeps track of all animation nodes, which are computationally driven to generate and pass value-change events into the scene graph. Events received by the scene graph can change any aspect of the rendered image. Further extending the animation nodes are scripts, which can send or receive events and also generate (or remove) geometry in the scene. Scripts encapsulate programming code from other languages, usually either ECMAScript (formerly known as JavaScript) or Java. The Scene Authoring Interface (SAI) defines how these application programming interfaces (APIs) work, allowing authors to create scripting code that can work across different operating systems and different browser software. Finally (ending at the upper-right corner of the diagram), HTML web pages and external applications can also be used to embed X3D plugin browsers, which appear to users as live, interactive 3D images on the page.

2.7 Scene Access Interface (SAI)

X3D is not only a 3D file format. Instead, it is a powerful markup language that allows (in combination with Script nodes) to create full-featured 3D applications. Sometimes it is not sufficient to write the complete application in X3D - e.g. when you need to integrate the 3D rendering into other software components. For these cases, there are standard means to control the 3D scene from external software components. These are the older External Authoring Interface (EAI), defined by the VRML standard and the newer Scene Access Interface (SAI), defined by the X3D standard. Both interfaces allow external software components to completely control the scene graph, i.e. by adding, modifying or removing nodes or by sending and receiving events.

The most important functions used in this thesis are presented in Table 1. These concern the parsing and the rendering of the X3D Document, the 'listening' of the events and the modification of the X3D fields.

Node	Method Summary
X3DBrowser	X3DScene **createScene**(ProfileInfo profile, ComponentInfo[] components) void **addBrowserListener**(BrowserListener l) void **removeBrowserListener**(BrowserListener l) void **startRender**() void **stopRender**()
X3DBrowserFactory	X3DComponent **createX3DComponent**(Map params)
X3DViewpointNode	float **getFieldOfView**() void **getPosition**(float[] val) void **setFieldOfView**(float val) void **setPosition**(float[] val)
X3DNode	X3DField **getField**(String name) X3DFieldDefinition[] **getFieldDefinitions**() X3DMetadataObject **getMetadata**() **setMetadata**(X3DMetadataObject node)
SwitchNode	int **getWhichChoice**() void **setWhichChoice**(int val)
ProximityNode	void **getCenterOfRotation**(float[] val) void **getOrientation**(float[] val) void **getPosition**(float[] val) void **setCenterOfRotation**(float[] val) void **setOrientation**(float[] val) void **setPosition**(float[] val)
MetadataSet	void **getValue**(X3DNode[] val) void **setValue**(X3DNode[] val)
MetadataString	void **getValue**(X3DNode[] val) void **setValue**(X3DNode[] val)

X3DField	**addX3DEventListener**(X3DFieldEventListener l) X3DFieldDefinition **getDefinition**() **removeX3DEventListener**(X3DFieldEventListener l) void **setUserData**(Object data)
X3DEvent	String **getName**() double **getTimeStamp**() ConstField **getValue**()

<div align="center">**Table 1: SAI Java Api methods wich have been used in current thesis.**</div>

2.8 Chapter Summary

This chapter introduced a set of fundamental terms of computer graphics starting with defining light energy and its properties and light propagation. Subsequently, computer graphics illumination models were analyzed. The following sections were focused on visual perception and its application to computer graphics rendering.

Most importantly, this chapter presented the shortcoming of previous selective rendering approaches. The need of a new approach which is gaze and task independent as well as excluding saliency model predictions is needed. In this thesis we present a novel selective rendering approach which exploits previous spatial knowledge and preconceptions through schemata in order to simulate cognitive processes rather than physics.

Furthermore, this chapter illustrated the complex variety of tools and equipment required to create, view and interact with immersive virtual environments. It provided background information regarding the key technologies used in this study and an overview of the technologies necessary to display and interact with immersive virtual environments. Also, the X3D framework was presented.

We will proceed with analyzing the schema memory processes and explain how these are going be exploited in order to devise a new selective rendering approach.

Psychology: The study of the human mind; The study of the human behavior; The mental characteristics of a particular individual

Chapter 3

Memory Schemata

While Chapter 2 introduces the basic principles of computer graphics and provides technical information relating to the complexities of VE generation, this chapter provides background information regarding the memory schema theory. The schema theory forms the basis of the research conducted.

3.1 Memory Schemata

This thesis uses an approach based on classic findings from memory research, in which schemata are used to explain memory processes. Schemata are knowledge structures of cognitive frameworks based on past experience which facilitate the interpretation of events and influence memory retrieval (Minsky 1975, Brewer and Treyens 1981).

Schemata provide a mechanism for integrating new experiences into old and are thus considered to be an effective memory tool. Schema theories propose that perception, language comprehension and memory are processes that involve the interaction of new episodic information with old, schema consistent information (Brewer and Treyens 1981, Mania et al 2005). Schema consistent spatial elements are expected to be found in a given spatial context, for instance into an academic's office. Minsky suggested that expectation based on prior knowledge or experience is already existent in memory, can trigger the brain to pre-load certain objects when encountering a familiar scene (Minsky 1975). Inferences of absent elements of a space are said to occur when memory performance after exposure to a space contains spatial information that was not actually there but was expected to be there based on past associations (Mania et al 2005). In other words, information slots which have not been filled with perceptual information are filled by default assignment based on stereotypical expectations (Kuipers 1975). Such an effect was demonstrated by the results from Kuiper's 1975 study: upon glancing a clock on a wall, hands may be assigned during a memory test even though no hands where actually present on that particular clock. Hands are expected to be found in the clock, therefore, this knowledge is retrieved and combined with perceptual information acquired at the time of exposure. In a computer graphics content, hands are rendered in lower quality since they are 'expected' to be found, exploiting such preconceptions to save rendering   computation.

It is possible that this integration is so complete that episodic information may be indistinguishable from schema consistent information, although this is not always the case.

Schema research has generally shown that memory performance is frequently influenced by schema-based expectations and that an activated schema can aid retrieval of information in a memory task (Minsky 1975). Brewer & Treyens (1981) proposed five different ways in which schemata might influence memory performance:

- By determining which objects are looked at and encoded into memory (e.g. fixation time).
- By providing a framework for new information.
- By providing information which may be integrated with new episodic information.
- By aiding the retrieval process.
- By determining which information is to be communicated during recall.

Schema research in a real world setting has demonstrated enhanced memory performance for consistent items, but interestingly has also shown the opposite, i.e. greater memory performance for inconsistent items. An overview of both sets of findings is discussed here.

Consistent items are those that are expected to be found in a given context. Pichet's & Anderson's 1966 schema model claimed that inconsistent items are ignored and thus predicted better memory performance for schema consistent items. The consistency effect as demonstrated by Brewer & Treyens (1981) also supported the concept that memory performance is better for consistent items. Schema expectancy, i.e. how likely an object is to be found in a scene, was positively correlated with recall and recognition scores after a short exposure to a scene. Participants were taken into what they assumed was an academic's office and were later tested for memory of the objects in the room with drawing recall, written recall or verbal recognition. Schema expectant objects were positively correlated with recall and recognition. Expected objects were also inferred in recall, an effect which demonstrated the ability of the office-schema information to influence place memory.

There is also evidence to support the inconsistency effect. The inconsistency effect contends that schema inconsistent information for an episodic event will be easily accessible and therefore would provoke better memory performance (Lampinien et al 2001). Rojahn and Pettigrew (1992) reviewed of a large number of independent schema studies, some of which supported the consistency effect and others the inconsistency effect. Their analysis indicated a slight overall memory advantage towards schema inconsistent information. Meta-analysis of their findings however, was thought to show that variations in favor of consistent versus inconsistent objects (or vice versa) could be due to a number of methodological variations such as, methods of presentation of the stimuli, instructions, total number of objects in the rooms, time of exposure to the environment and type of memory task.

3.2 Memory Recognition Experimental Methods

In this section, we will present a complete experimental procedure of memory recognition. In order to examine the role of memory schema within a real-world place, an experimental environment based upon a 'typical' office was created (Figure 36). Participants were exposed to the office for a short period of time (Brewer & Treyens 1981).

Figure 36: photograph of the original experimental 'office' used in the Brewer & Treyens experiment.

Based upon the supposition that knowledge about rooms is organised into a 'room frame' in memory and that this room frame contains information about the room that one may be nearly certain about before entering (Minsky 1975), seven items within the test office designed by Brewer and Treyens (1981) were designated room-frame items: wall, floor, ceiling, door, doorknob, lights and light switch. These frame items were chosen since they are objects that any room is almost certain to contain, regardless of room function. In addition to the frame objects, the test room also contained objects that were specifically intended to give the impression that it was indeed an office such as a typewriter, desk and calendar. However, it also included certain objects such as human skull that are inconsistent with the office schema and deliberately omitted certain objects such as books that would almost certainly be expected. The objects selected for inclusion or omission within the test environment were based upon a preliminary study of twentyeight test participants, split into two groups of 14, each given a booklet containing 70 objects. The first test group were asked to rate these objects in terms of *Saliency* or how noticeable a given item was or would be if it were included within an office. The second group were asked to rate how likely it would be to find each object within an office in order to provide an indication of *Schema-expectancy*.

A group of 86 test participants were asked to take part in the experiment, but were not initially informed what this experiment would consist of. Upon arrival they were individually taken into the test office described above and asked to have a seat and wait while the experimenter checked on the progress of the previous test participant. Only one seat was empty since the others contained various objects. After 35 seconds the experimenter reappeared and asked the participant to follow him to a nearby seminar room where they were informed that the real nature of the experiment was to test their memory of the office. The initial 86 test participants were divided into 3 groups, one given a written recall test, one a verbal recall test and one drawing recall test, each test subgroup presented with the test in groups of one or two, within a constant post exposure time of approximately one minute. The results of each written, verbal and drawn test were examined to check for evidence of an active room schema. The initial subterfuge whereby the test participants were unaware of the real experimental objective is an important factor; if they had been aware the experiment was based on memory recall, it is likely that each would have deliberately paid attention to the room with regard to its contents and specifically noted each item, thereby invalidating the results with regard to the influence of schemata in memory.

When considered overall, the results from this experiment provide strong support for the action of schemata in memory for places, with clear evidence for the integration of episodic and schema-based knowledge. In the various types of recall test for example there were a number of objects recalled as being present which were not actually there. Such inferred objects received high schema-expectancy scores during the preliminary research. Books, which one would expect to find in an office and which therefore can be considered of high schema-expectancy, were not present in the room. Out of the 81 non-frame objects presented to the test participants during the recall tasks, they were the 16th most frequently recalled item. The fact that books and other high expectancy items that were not present but still frequently recalled can only be explained, Brewer hypothesises, by the existence of an active room schema. The recall of such items provides strong evidence that schema-based information about offices is integrated with the episodic information related to the experimental room (Brewer & Treyens 1981).

Table 2 provides a breakdown of the responses given by the 30 participants who were given the written recall test, listed in order of recall frequency. For example, 29 participants rememberd that there was a desk in the room.

Objects + Frequency		Objects + Frequency		Objects + Frequency		Objects + Frequency	
Chair (next to desk)	29	Clown light switch	8	Cardboard boxes	2	Hole in wall	1
Desk	29	Coffee pot	8	Computer cards	2	Home coming button	1
# Wall	29	Skull	8	Papers on bulletin board	2	Magazines	1
Chair (in front of desk)	24	Mobile	7	* Pens	2	Sugar	1
Poster (of chimp)	23	Road sign	6	Pot (for cactus)	2	Paper (on desk chair)	1
# Door	22	Calendar	6	Solder	2	Papers (on shelf)	1
Table (worktable)	22	Doll	5	Vacuum tube	2	* Pencil holder	1
Shelves	21	Jar of creamer	5	* Window	2	* Pencils	1
# Ceiling	16	Pipe	5	Wires	1	Picnic basket	1
Table (with coffee)	15	Postcards	5	* Ball	1	* Pliers	1
Skinner Box	14	Tennis racket	5	* Brain (additional)	1	Saucer	1
Childs chair	12	Fan	4	Brick	1	* Scissors	1
# Floor	12	Coloured patterns on lights	4	Computer Surveys (on floor)	1	* Screws	1
# Light Switch	12	Piece of Bark	4	* Curtains	1	Teaspoon	1
Toy Top	12	* Filing cabinet	3	* Decals on wall	1	* Telephone	1
Brain	11	Frisbee	3	* Desk (additional)	1	Umbrella	1
Parts (on worktable)	11	Coffee jar	3	# Doorknob	1	Wrench	1
Poster on ceiling	10	Poster (additional)	3	Eraser	1		
* Books	9	Screwdriver	3	Air Vent	1		
# Ceiling light	9	Snoopy picture	3	* Glass plate (on desk)	1		
Poster of food	9	Rotary switches	3	Globe	1		
Typewriter	9	Cactus	2	* Lamp	1		
Bulletin board	8	* Coffee Cup	2	* Nails	1		

Table 2: Responses given by participants (# denotes Frame object, * denotes inferred object).

The interaction of new episodic information and existing schema-based information is largely born out by the results obtained from this experimental data, with high schema-expectancy objects (such as books) being recalled even when not present. Low schema-expectancy objects by comparison (such as the ball) were not recalled as frequently, due to the lack of schemata facilitating their recall.

3.3 Relevance to Virtual Reality

Based on the conclusions of the study presented in the previous section, Brewer and Treyens 1981 concluded that schemata play an important role in place memory of real-world environments. The objective of a Virtual Environment is to replicate as closely as possible the real-world environment simulated. It is, therefore, of great interest to assess behavioural fidelity of VEs with regard to the existence of memory schemata and to explore the effect of rendering quality on memory recognition. If memory schemata play a similar role such as in real world studies, we could conclude that the VE simulated is of high functional and behavioral fidelity. Previous methods for ascertaining such immersion rely on asking fairly emotive and open-ended questions post-exposure, such as "how realistic did you feel the

environment was" or "Were you convinced you were actually there" (Slater, 1999). Such questions may be answered dependent on users' interpretation of the question. Post-exposure tests for memory recognition on the other hand may be conducted providing quantitative data which can be analyzed utilyzing standard statistical tests, providing a potentially more accurate indication.

Research investigating how schema theories apply to real versus virtual memories has also supported both contentions, i.e. support exists favouring the consistency effect related to recognition scores, but also exists favouring the inconsistency effect related to confidence ratings. Despite the schema/non schema dichotomy, it is clear that schemata do play an important role in encoding and retrieving information from real-world scenes. Moreover, since the same dichotomy exists in the real world, it can be concluded that synthetic scenes could induce similar responses as real world scenes (Flannery & Walles 2003).

Mania et al (2005) aimed to determine whether the effect of schemata on recognition memory would be analogous to the real-world study by Brewer and Treyens 1981. They explored the effect of shadow detail on memory recognition aiming to identify whether consistent or inconsistent items provoke better memory performance in synthetic worlds. They considered schema theory relevant to their goal of exploring functional fidelity since it establishes the degree to which the appropriate schema has been activated in the real scene and its VE equivalent. Schema activation occurs when schema consistent or inconsistent information either positively or negatively correlate with memory performance, hence influencing place memory. The closer the similarity between the two schema activations, the higher the functional fidelity of the VE irrespectively of the level of visual or interaction fidelity per se (Mania et al 2005). This allows various aspects of the VE system to be varied in order to investigate how these variations impact on memory performance.

Mania et al (2005) conducted an experiment to explore the effect of object type and rendering quality on object memory recognition in a room (an academics office). The computer graphics simulation, rendered using radiosity, was displayed on a HMD utilising stereo imagery and head tracking. 36 participants across three condition of varied rendering quality of the same space were exposed to the computer graphics environment and complete a memory recognition task. The simulations consisted of an office displayed on stereo capable head-tracked Head Mounted Display (HMD). The high-quality and mid-quality conditions included a pre-computed radiosity simulation of an office (with 80% and 40% radiosity iterations respectively). The low-quality condition consisted of a flat shaded version

of the same office (no shadows). The findings of the study support the consistency effect as revealed in the Brewer and Treyens 1981 study. Results revealed that objects expected to be located in the scene (schema consistent) were more likely to be recognized than unexpected ones (inconsistent). The consistency effect was revealed both for objects that were present as well as absent in the scene and was independent of viewing condition., ie. rendering quality. They also found that participants assigned overall higher confidence ratings (confidence about responses) for consistent compared to inconsistent items, irrespective of viewing condition. Total object recognition was better for the mid-quality condition compared to the low-quality one. The presence of shadow, though, did not affect recognition of consistent or inconsistent objects, therefore low quality of rendering was adequate for better memory recognition of consistent objects

Another experimental study employed a more extreme set of rendering types: wireframe with added color and full radiosity (Troscianko et al. 2007, Mourkoussis et al. 2009) (Figure 1). The proportion of inconsistent/consistent objects was varied and object recognition tests ensured that all objects were easily recognized in all conditions. The results showed a significant interaction between rendering type, object type and consistency ratio. This suggests that inconsistent objects are only preferentially remembered if the scene looks "normal" or if there are many such objects in an "abnormal" scene such as in the wireframe condition. It was also shown that memory performance is better for the inconsistent objects in the radiosity rendering condition compared to the wireframe condition. We conclude that memory for objects can be used to assess the degree to which the context of a VE appears close to expectations. Schema-based expectations and related prior knowledge could be exploited in order to economize on rendering computation related to the polygon count of individual objects found in a scene.

In Chapter 4, we will start presenting the implementation of a Selective Rendering Framework by employing the consistency effect and assuming that the lower polygon count quality assigned to consistent items in the scene will not affect memory recognition when compared to a high polygon count rendering solution. In this sense, prior knowledge and preconceptions will compensate for the missing polygons of consistent objects.
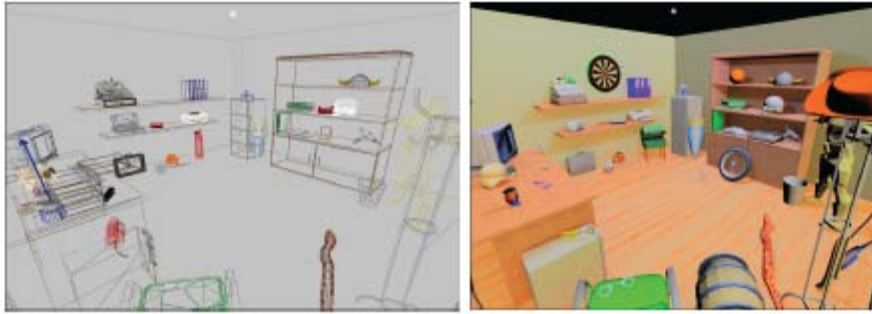
Figure 37: Schema memory experimental studies (Troscianko et al. 2007, Mourkoussis et al. 2009).

3.4 Chapter Summary

This chapter provided background information regarding the memory schema theory. Experimental studies in synthetic scenes have revealed that consistent objects which are expected to be found in a scene can be rendered in lower quality without affecting information uptake taking advantage of such expectations, whereas inconsistent items which are salient would require a high level of rendering detail in order for them to be perceptually acknowledged. Therefore, by exploiting schema theory, it is possible to reduce computational complexity, producing scenes from a cognitive point of view without affecting information uptake and resulting in an entirely novel and interdisciplinary approach which is gaze, task and saliency-model independent. The next chapter introduces a rendering framework which exploits schema theory and its benefits.

A software framework, in computer programming, is an abstraction in which common code providing generic functionality can be selectively overridden or specialized by user code providing specific functionality

Chapter 4

A Selective Rendering Framework

While the previous chapter provided background information regarding the memory schema theory, this chapter will introduce a Selective Renderer Framework (Zotos, Mania, Mourkoussis 2009) which exploits schema theory in order to economize in rendering computation.

4.1 Introduction

We present a novel selective rendering system that will exploit schema theory by identifying the perceptual importance of scene regions. Objects that have been rendered in low and high polygon count quality are incorporated in a scene based on schema expectations. The rendered quality of these objects will change in real time, dependent on user navigation and interaction.
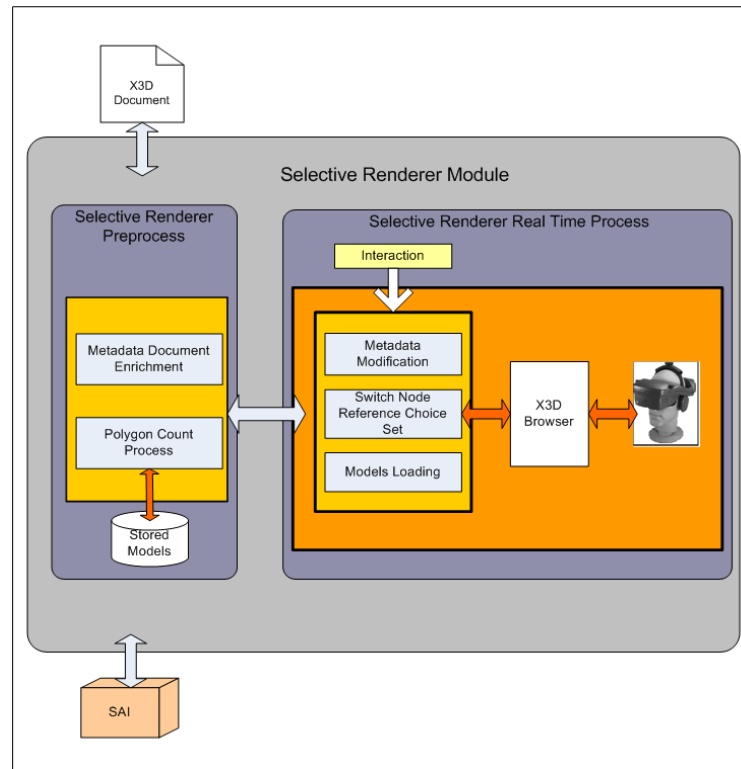
Figure 38: The Selective Renderer Architecture.

The **Selective Renderer Software Engineering Framework** has one major objective, to economize on rendering computation. Our goal is to showcase that by exploiting schema theory, it is possible to produce scenes from a cognitive point of view without affecting the information uptake and at the same time reduce computational complexity. This will be accomplished by initially producing objects of varying polygon count based on whether an object is consistent with the content of the scene. We assign high quality of polygon count to inconsistent objects in the scene, e.g. objects which are salient in a given scene and lower quality to consistent objects, e.g. objects which are expected to be found in a given context.

The **Selective Renderer Framework** (Figure 38) is responsible of modifying and enriching an X3D document with appropriate metadata information. Such a document describes the geometry of a given scene in relation to whether an object is consistent or inconsistent. It

exploits this type of information in real-time in order to apply differing polygon quality to objects included in specific scene's regions. The architecture of the selective renderer module comprises of two main components: the S**elective Renderer Preprocess** component and the **Selective Renderer Real Time** component. Input to the system is an X3D document that describes the geometry of the scene. Validation and modification of the X3D document is achieved via the Scene Access Interface (SAI) which is a JAVA API that provides all the appropriate functions for the processing of the document.

4.2 The Selective Renderer Preprocess Component

The **Selective Renderer Preprocess** component is responsible for the pre-processing and enrichment of the X3D document with congruous metadata information available during the real-time process. During the pre–process phase, varied polygon count versions of the scene's objects are created. Consistent objects which are expected to be found in a scene are rendered in lower quality to the point of being recognizable whereas inconsistent items which are salient would require a high level of rendering. Every X3D node that describes the geometry of an object for which different versions have been created, is replaced by a *switch* node that points to varied polygon quality objects through inline nodes. The *WhichChoice* field of every *switch* node guides which version of the object should be rendered. Thus, by altering the value of this field, the appropriate polygon count version of each object model can be loaded in real-time, as the user navigates any 3D scene.

The next sub sections analyze the functionality of the Polygon Count Process Component which creates different polygon count versions of the objects and the Metadata Document Enrichment Component which is responsible for the metadata enrichment of the X3D document.

4.2.1 The Polygon Count Process Component

During the **Polygon Count Process** stage, varied polygon count versions of objects are created and stored with the help of a polygon count tool that diminishes the number of polygons of an object to the point that it is recognizable. Consistent objects which are expected to be found in a scene are rendered in lower quality whereas inconsistent items which are salient would require a high level of rendering. Thus, the appropriate polygon count version of each object model can be loaded in real-time, while the user navigates the 3D scene.

Chapter 5 includes details and examples related to the polygon count process when building a specific scene depicting an apartment. An example is presented below (Figure 39).
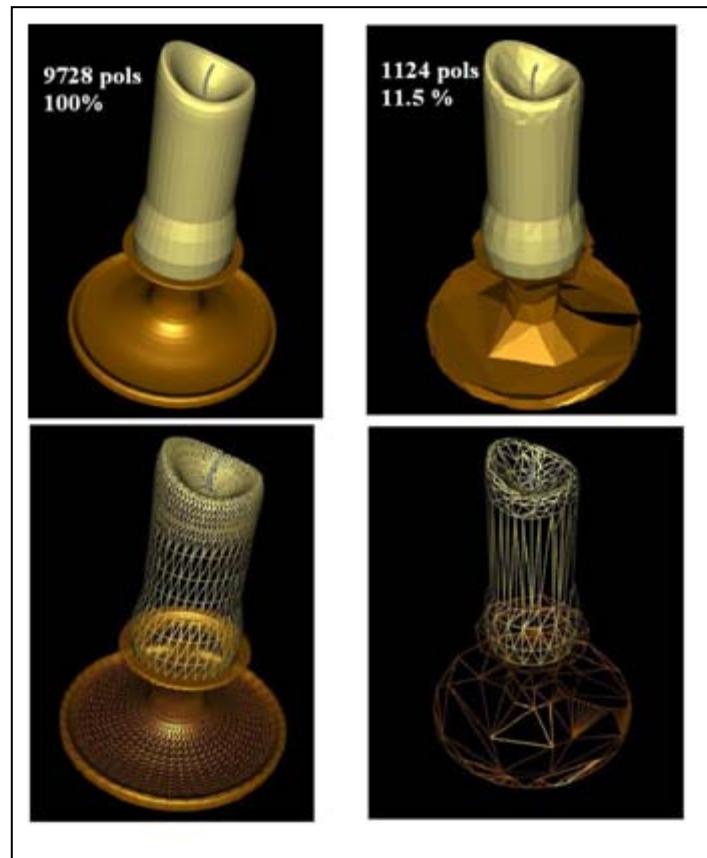
Figure 39: Two polygon versions of a candle model which has been used in our scene. Left up is the original model and right up is the diminished version. Wireframe editions of the objects appear at the bottom. High poly edition consists of 9728 polygons and low poly edition consists of 1124 polygons.

To retain consistency of appearance throughout the scene, the polygon count of each object is reduced on an individual basis until it is felt that the object contains the minimum number of polygons required to communicate the object's recognizable shape.

Scenes are created in which all objects communicated the same perceived visual quality per condition, regardless of polygon count per object, as determined by detailed pilot studies. Participants were required to recognize different diminished versions of the same object. For each object, it was decided that the lowest polygon count version which was easily recognized by the participants should be retained.

4.2.2 The Metadata Document Enrichment Component

Metadata is the use of data to explain other data. An X3D metadata node can be used to provide such information about any node in an X3D scene graph and is placed as a child of the node that it describes. Metadata nodes are persistent, meaning that their values remain available and accessible after loading. The Metadata Document Enrichment stage exploits the X3D's capability of defining metadata information for every node of an X3D document. Metadata information describes the level of consistency that every object entails in relation to the specific context of a scene's region (Figure 40).
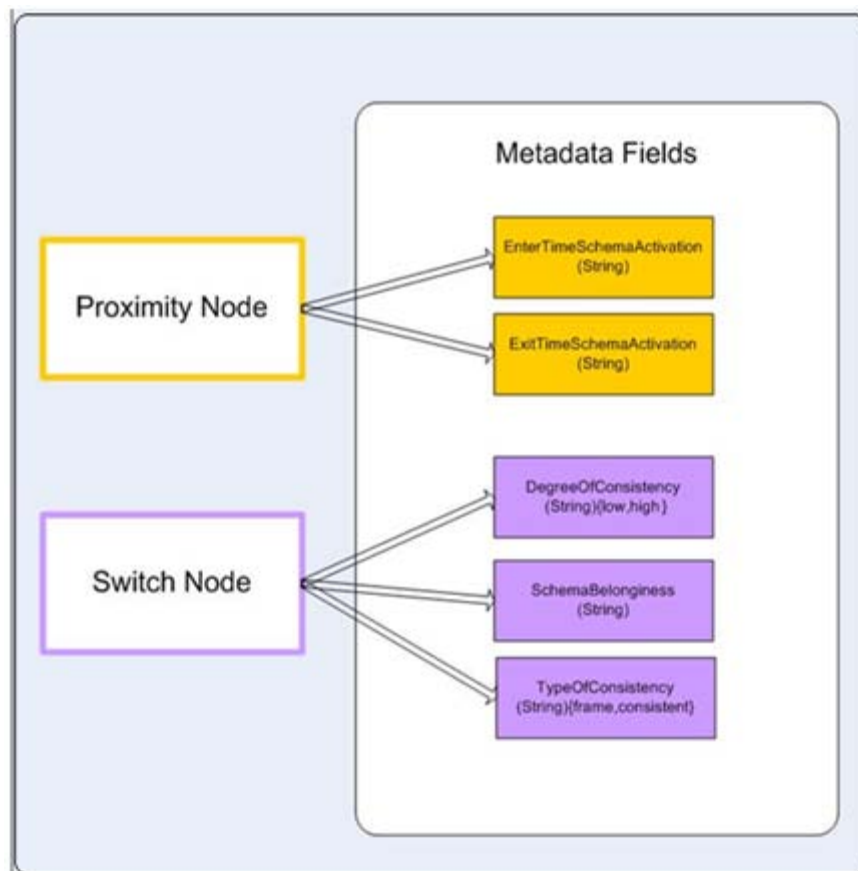
Figure 40: Metadata fields that Selective Renderer Framework defines.

Metadata information is defined for every *Proximity* node. This information describes which schema(s) is activated when the avatar enters a specific region (*enterTimeSchemaActivation* meta-field) as well as when it exits that region (*exitTimeSchemaActivation* meta-field). Similarly, metadata information is defined for every *Switch* node. This information describes the level of consistency of each version of the object in relation to the context of the scene (*degreeOfConsistency* meta-field, 2 levels of consistency: low, high), the area description that corresponds to the positioning of the objects (*schemaBelonginess* meta-field) and

whether or not this object is a frame object, e.g. wall, ceiling, lights fixture, door, etc (*typeOfConsistency* meta-field).

We assume that a scene, i.e. a house, comprises of areas of varied context such as a kitchen, an office and other sub-areas. Similarly, these sub-areas can also be associated with sub - schemata: an academic office area may contain a desk sub-area and a bookcase subarea. It is crucial to devise a method that can incorporate information related to the level of consistency of objects in each sub-area, in the X3D input document. At the time that a sub-area is 'activated' as the user navigates through it, information concerning the level of consistency or inconsistency of objects will define the polygon quality version of the objects in that area which is loaded and visualized. A *Proximity* Node is created for every area and sub-area in order to track user interaction in these regions. Metadata information is attached to every proximity node which describes whether objects are consistent or inconsistent in relation to the context of each sub-area. When the navigation avatar approaches a scene's region, this action is acknowledged and by exploiting relevant metadata information it is feasible to know which knowledge schema should be activated. 'Activation' will occur when the user navigates closely to each area.

The next step of the Metadata Document Enrichment stage is to devise a method that can relate the activated 'memory schema' with the appropriate polygon quality version of each object. The X3D's capability of defining metadata information makes it possible to implement such a strategy. *Switch* nodes indicate which version of the object should be rendered depending on user navigation. For example, let's assume an area *areaA* that activates the memory schema *schemaA*. This area contains sub-area *subAreaB* that activates the memory schema *subSchemaB*. An object of the scene *Obj* is consistent in relation to the *area*, however, it could be inconsistent in relation to the *subAreaB* (Figure 41)*.*
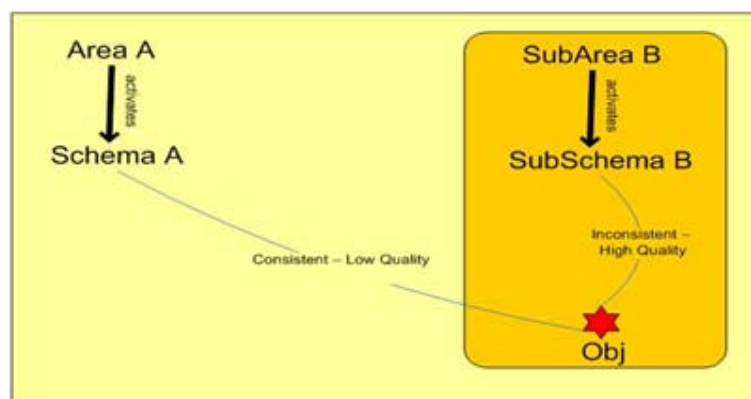
Figure 41: Example of an object that belongs to different schemata with different levels of consistency.

The Metadata Document Enrichment component will define the metadata information attached to the *proximity* nodes that represent *areaA* and *subAreaB* by associating these areas with the memory schema to be activated. So, *areaA* should activate *schemaA* and *subAreaB* should activate *subSchemaB* (Figure 42).

```
<Transform DEF="ProxArea" translation="-0.1614 0.0 -2.053">
  <ProximitySensor DEF="RoomProximity" size="4.613 2.768 4.843">
    <MetadataSet DEF="MetadataSet_Example" containerField="metadata">
      <MetadataString name="enterTimeSchema" value="SubSchemaB"/>
      <MetadataString  name="exitTimeSchema" value="SchemaA"/>
    </MetadataSet>
  </ProximitySensor>
</Transform>
```

Figure 42: Metadata example code of a *proximity* sensor. Schema A is activated when the avatar exits the *proximity* sensor. SubSchema B is activated at the entrance.

Metadata information is also included in the *switch* node of the object *Obj* (Figure 43). This information should associate each version of the object with the appropriate level of consistency in relation to each context.

High *degreeOfconsistency* could be related to schemaA or *low degreeOfconsistency* could be activated by subSchemaB. If an object is consistent, a low quality polygon version of this object will be loaded. If the same object is inconsistent in relation to a different sub-area, then a high quality version of the same object will be loaded. Eventually, by tracking which 'schema' or context is activated by user interaction, the appropriate polygon quality version of the object *Obj* is loaded and visualized.

```
<Switch DEF="Obj" whichChoice="0">
  <MetadataSet DEF="MetadataSet_Obj" containerField="metadata">
    <MetadataSet containerField="value" >
      <MetadataString   name="consistencyType" value="consistent"/>
      <MetadataString   name="degreeOfConsistency" value="low"/>
      <MetadataString   name="schemaBelonginess" value="SubSchemaB"/>
    </MetadataSet>
    <MetadataSet containerField="value" >
      <MetadataString   name="consistencyType" value="consistent"/>
      <MetadataString   name="degreeOfConsistency" value="high"/>
      <MetadataString   name="schemaBelonginess" value="SchemaA"/>
    </MetadataSet>
  </MetadataSet>
</Switch>
```

Figure 43: Metadata example code of a *switch* node. Object Obj belongs to schemaA with high degree of consistency and to SubSchemaB with low degree of consistency.

The metadata enrichment procedure of the X3D document can also be achieved with the support of a Graphical User Interface (GUI) that our implementation provides (Figure 44) in order to automatically produce X3D metadata information.

Figure 44: Screen shot of the Graphical User Interface.

Figure 44 presents a screen shot of the Graphical User Interface. The user should perform several steps (Figure 45) in order to enrich the X3D document with the appropriate metadata information.
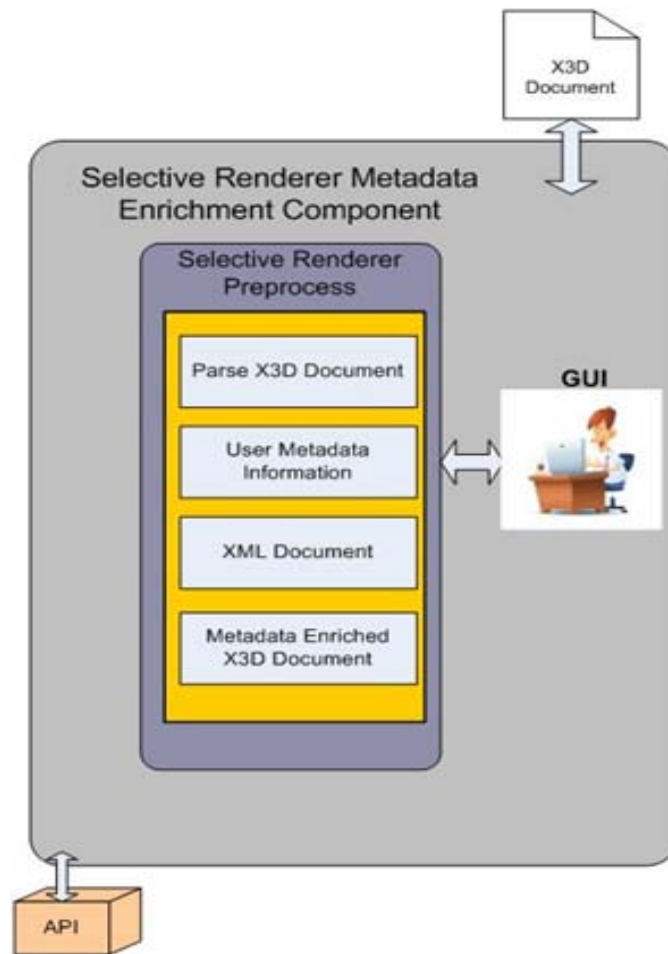
Figure 45: The steps of the metadata enrichment process.

In detail, the user should parse the X3D document (input), select which X3D nodes need to be enriched with the metadata information and fill in the corresponding form (Figure 44). There are two forms. For each *SwitchChoice* the user can define metadata information about *schemaBelonginess* and degreeOfConsistency metadata fields. Similarly, for each proximity node the user can define metadata information about enterTimeSchema and exitTimeSchema metadata fields. Finally, the user should save his choices and produce the metadata enriched X3D document.

The Selective Renderer Framework can accommodate any X3D scene as long as relevant object metadata are included. Furthermore, it can support more complicated cases such as, for example, a scene's region which includes many sub-areas which are activated by user interaction simultaneously (Figure 46).
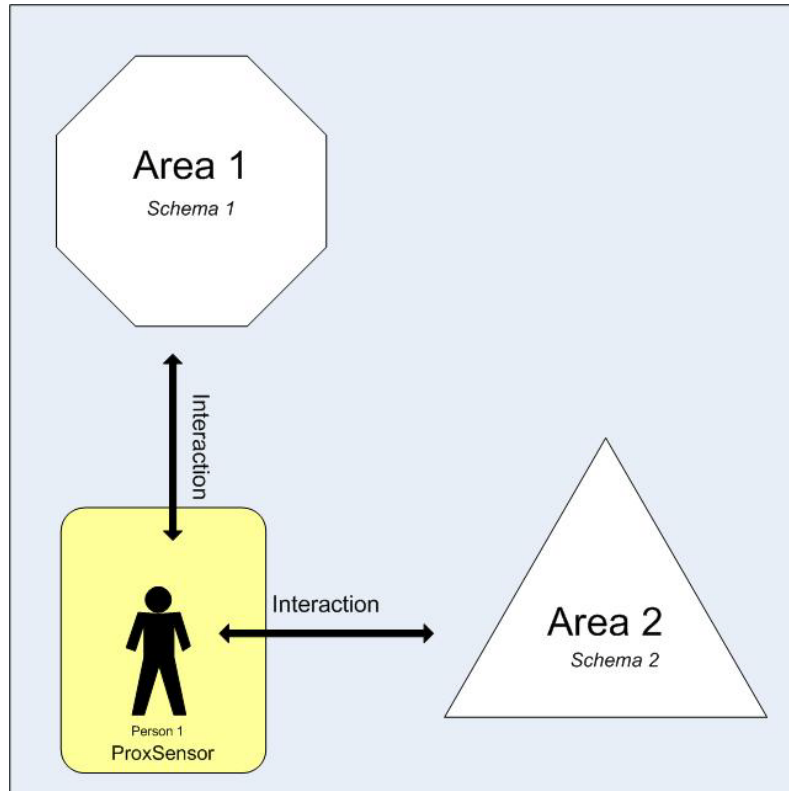
Figure 46: A scene's region which includes many sub-areas which are activated by user interaction simultaneously.

Figure 46 describes a case where the user activates at the same time more than one areas. Activation occurs when the user enters the boundaries of the *proximity* area. When the user enters the *proximity* area, the context (the schema) of the area 1 and area 2 should be activated. In this case, metadata information should be defined for the *Proximity* node which will trigger both the schema 1 and the schema 2 (Figure 47).

```
<ProximitySensor DEF="ProxSensor">
    <MetadatSet DEF="MetaSetExample" containerField="Metadata">
        <MetadataString name="enterTimeSchema" value="Schema1"/>
        <MetadataString name="enterTimeSchema" value="Schema2"/>
    </MetadatSet>
</ProximitySensor>
```

Figure 47: Metadata information that activates at the same time two schemata.

Figure 47 presents the X3D code defined for the *Proximity* node. Two *MetadataString* nodes were defined. When the user enters the *proximity* area (*enterTimeSchema*, *value*) , the schema1 and the schema2 should be activated.

Therefore, the Selective Renderer Framework is able to handle any case when appropriate metadata has been defined.

4.3 The Selective Renderer Real Time Process Component

The **Selective Renderer Real Time Process** component is responsible for the 3D rendering of the synthetic scene as well as the user navigation and the real-time processing of varied polygon quality objects. Rendering and navigation are achieved with the help of the Xj3D browser while users are immersed in the synthetic scene through a stereo capable, head-tracked Head Mounted Display (HMD). At this stage, the previously modified and metadata enriched document of the pre-process stage is exploited.

Several methods have been implemented that can track user navigation when a *proximity* node is activated (*enterTime*, *exitTime*) and update the corresponding metadata information attached to each node. In this manner, decisions are taken concerning the memory schema (context) activated and therefore, which polygon count version of the object should be loaded. Consequently, *switch* nodes that have been enriched with metadata information relevant to the activated schema are detected.

In particular, some of the most important functions worth mentioning are the following:

- *void senseProximitySensors(Event event)*(Figure 48). When a *proximity* node is activated, this method 'listens' to the event and decides which schema should be activated.

- *void activateSwithNodes(String schema*(Figure 49))*. This method loads the corresponding polygon count versions of the objects that are related to the activated schema.

- *void setViewpoinFromTracker(TcpClient tcpclient)* (Figure 50)*. This method communicates with the tracker, gets the orientation values and converts them to radians. Then, it sets current orientation of the avatar.

- *void setPositionFromMouse(Evente event)* (Figure 50)*. This method gets events from the mouse and moves the avatar. If the user presses the right button, the avatar moves backwards, otherwise, if the user presses the left button the avatar moves forward.

```
void senseProximitySensors (Event event)
{
            if(event.hasType("enterTime"))
            {
                        Proximity activatedProximity = event.getSource();
                        MetadataSet metaSet = activatedProximity.getMetadataSet();
                        for(MetadataString metaString : metaSet.getMetadataStrings())
                        {
                                    if(metaString.getName().equals("enterTimeSchema"))
                                    {
                                                String enterSchema = metaString.getValue();
                                                activateSwithNodes (enterSchema);
                                    }
                        }
            }

            if(event.hasType("exitTime"))
            {
                        Proximity activatedProximity = event.getSource();
                        MetadataSet metaSet = activatedProximity.getMetadataSet();
                        for(MetadataString metaString : metaSet.getMetadataStrings())
                        {
                                    if(metaString.getName().equals("exitTimeSchema"))
                                    {
                                                String exitSchema = metaString.getValue();
                                                activateSwithNodes (exitSchema);
                                    }
                        }
            }

}
```

Figure 48: The *senseProximitySensors* method. This method 'listens' to the event and decides which schema should be activated.

```
void activateSwithNodes(String schema)
{
            for(SwitchNode switchNode : x3dScene.getAllSwitchNodes())
            {
                        MetadataSet metaSet = switchNode.getMetadataSet();
                        for(MetadataSet metadataSet : metaSet.getMetadataSet())
                        {
                                    for(MetadataString metaString : metadataSet.getMetadataStrings())
                                    {
                                                if(metaString.getName().equals("schemaBelonginess"))
                                                {
                                                            String schemaBelonginess = metaString.getValue();
                                                            if(schemaBelonginess.equals(schema))
                                                            {

            switchNode.setWhichChoice(metadataSet.getIndex()) //load the polygon count version
                                                            }
                                                }
                                    }
                        }
            }
}
```

Figure 49: The a*ctivateSwitchNodes* method. This method loads the corresponding polygon count versions of the objects that are related to the activated schema.

```
void setViewpoinFromTracker(TcpClient tcpclient)
        {
                    //Convert euler to radians
                    double heading=tcpclient.getTrackerCoordinates();
                    double attitude=tcpclient.getTrackerCoordinates();
                    double bank=tcpclient.getTrackerCoordinates();


                    heading=heading*Math.PI/180;
                    attitude=attitude*Math.PI/180;
                    bank=bank*Math.PI/180;

                    double x,y,z,w,angle;
                    double c1 = Math.cos(heading/2);
                    double s1 = Math.sin(heading/2);
                    double c2 = Math.cos(attitude/2);
                    double s2 = Math.sin(attitude/2);
                    double c3 = Math.cos(bank/2);
                    double s3 = Math.sin(bank/2);
                    double c1c2 = c1*c2;
                    double s1s2 = s1*s2;
                    w =c1c2*c3 - s1s2*s3;
                    x =c1c2*s3 + s1s2*c3;
                    y =s1*c2*c3 + c1*s2*s3;
                    z =c1*s2*c3 - s1*c2*s3;
                    angle = 2 * Math.acos(w);
                    double norm = x*x+y*y+z*z;
                    if (norm == 0.000) { // when all euler angles are zero angle =0 so
                                    // we can set axis to anything to avoid divide by zero

                                    x=1;
                                    y=z=0;
                    } else {
                                    norm = Math.sqrt(norm);
                    x /= norm;
                    y /= norm;
                    z /= norm;
                    }
                    //System.out.println(x+" "+y+" "+z+" "+angle);
                    //System.out.println(angle*180/Math.PI);
                    Vector orientationVector=new Vector();
                    float xf=(float) x;
                    float yf=(float) y;
                    float zf=(float) z;
                    float anglef=(float) angle;
                    orientationVector.add(xf);
                    orientationVector.add(yf);
                    orientationVector.add(zf);
                    orientationVector.add(anglef);

                    viewPoint.setOrientationorientationVector)

        }

        void setPositionFromMouse(Evente event)
        {
                    Vector currentPositionOfAvatar = viewPoint.getCurrentPosition();
                    if(event.hasType("rightClick"))
                    {
                                    currentPositionOfAvatar.setX(- (Xj3dBrowser.xAxis.position)) //move backwards
                                    currentPositionOfAvatar.setY(- (Xj3dBrowser.yAxis.position)) //move backwards
                    }
                    else if(event.hasType("rightClick"))
                    {
                                    currentPositionOfAvatar.setX((Xj3dBrowser.xAxis.position)) //move forward
                                    currentPositionOfAvatar.setY((Xj3dBrowser.yAxis.position)) //move forward
                    }

                    viewPoint.setCurrentPositon(currentPositionOfAvatar);
        }
}
```

Figure 50: The Set orientation and position methods. These methods communicate with the tracker and the mouse so the user can navigate through the scene.

By identifying the degree of consistency of every object of the scene when a sub-area is entered, it is possible to change its level of detail in real-time. An example of a model that changes its level of detail real time is presented below.
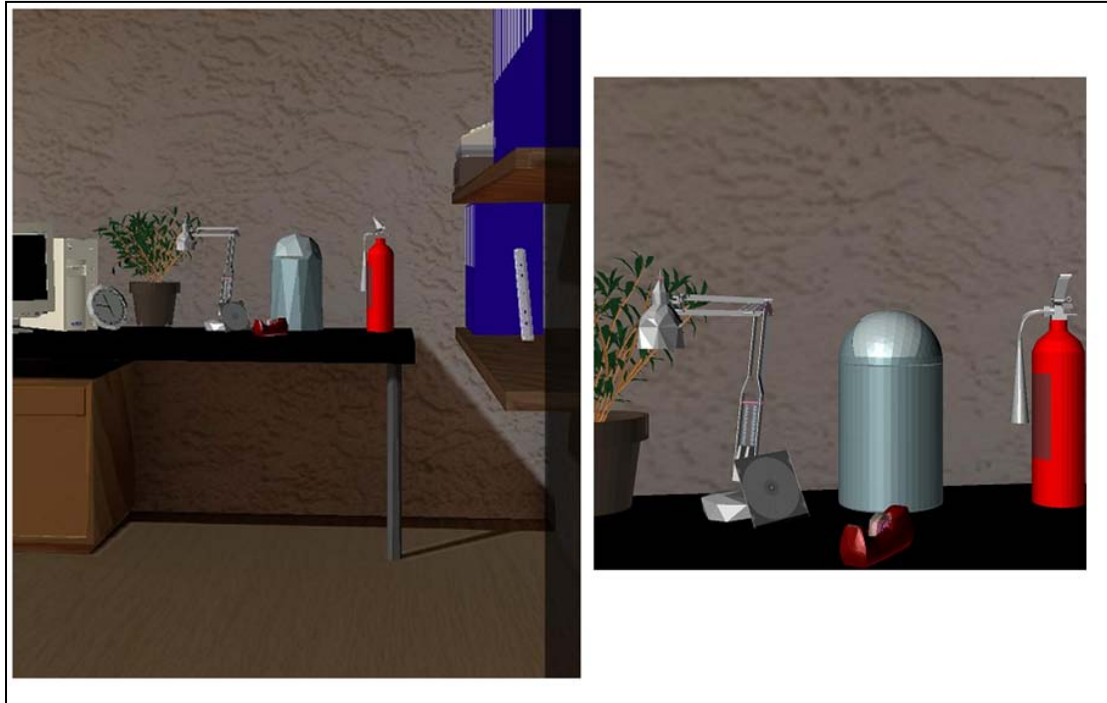
Figure 51: Real time rendering behavior of a trashcan. Left, the low quality of the trashcan is loaded. Right, the high quality of the same model is loaded.

Figure 51 describes a scenario when an object is consistent for the office schema and inconsistent for the desk schema. The object requires high polygon count rendering when the desk schema is activated and respectively, low polygon count rendering when the office schema is activated. Activation occurs when the users enters the defined boundaries of each area. This alteration is achieved through the metadata enrichment of the *Switch* node which contains the different versions of the same object. More specifically, the object in question is a trashcan. The trashcan is considered to be consistent for the office schema and inconsistent for the desk schema. The X3D code in Figure 52 defines a *Switch* node which points to two versions of a trashcan model. Two *metadataSet* nodes have also been defined which determine the degree of consistency for each version of the trashcan. The low polygon count version is linked to the office schema. Similarly, the high polygon count version is linked to the desk schema (*schemaBelonginess*).

```
<Switch DEF="Switch_TrashCanDefinition"  containerField="children" whichChoice="1">
   <Inline DEF="Inline_TrashCan_v1_Definition"  url="inlineNodes/trashCan_v1.x3d"/>
   <Inline DEF="Inline_TrashCan_v2_Definition"  url="inlineNodes/trashCan_v2.x3d"/>


<MetadataSet  containerField='metadata' DEF="MetadataSet_TrashCanDefinition">
   <MetadataSet  containerField='value' DEF="MetadataSetTrashCan_Part1Definition" >
      <MetadataString DEF="consistencyTypeDefinitionTrashCan_Part1" name="consistencyType" value="consistent" conta:
      <MetadataString DEF="degreeOfConsistencyDefinitionTrashCan_Part1" name="degreeOfConsistency" value="high" cont
      <MetadataString DEF="schemaBelonginessDefinitionTrashCan_Part1" name="schemaBelonginess" value="OfficeSchema"
   </MetadataSet>
   <MetadataSet  containerField='value' DEF="MetadataSetTrashCan_Part2Definition" >
      <MetadataString DEF="consistencyTypeDefinitionTrashCan_Part2" name="consistencyType" value="consistent" conta:
      <MetadataString DEF="degreeOfConsistencyDefinitionTrashCan_Part2" name="degreeOfConsistency" value="low" conta
      <MetadataString DEF="schemaBelonginessDefinitionTrashCan_Part2" name="schemaBelonginess" value="DeskSchema" cc
   </MetadataSet>
</MetadataSet>
</Switch>
```

Figure 52: *Switch* node metadata information of a trashcan model. Trashcan is consistent (high degree of consistency) for the office schema and inconsistent (low degree of consistency) for the desk schema.

When the office *proximity* sensor is activated, the trashcan is rendered in low quality since this object is consistent for the office schema and requires low polygon quality. When the desk *proximity* sensor is activated, the trashcan is dynamically rendered in high quality since this object is inconsistent for the desk schema and requires high rendering quality (Figure 53). A similar situation in relation to various objects in the scene will be presented and analyzed in the next chapter which describes a complete implementation of a selectively rendered scene.
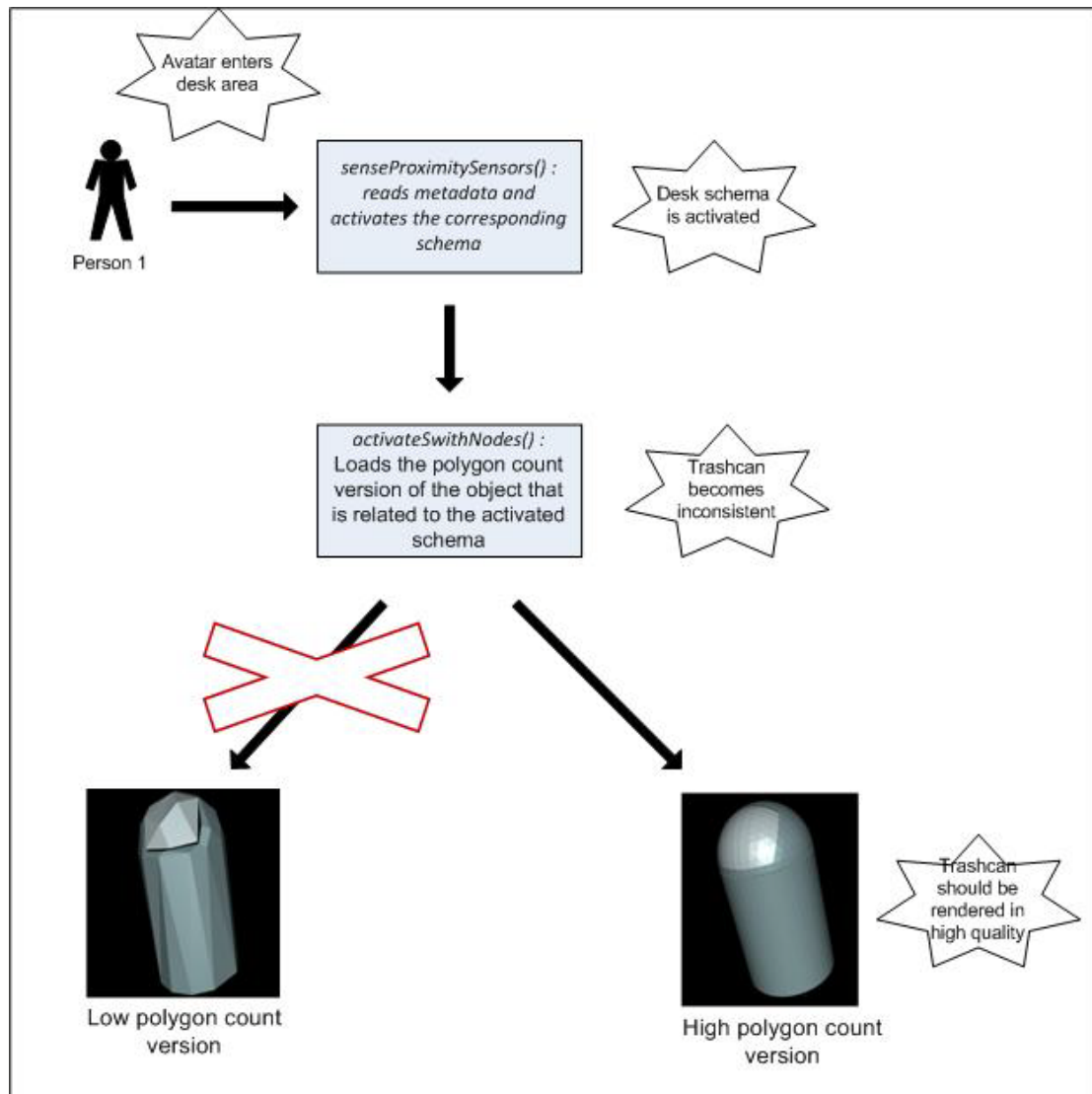
Figure 53: The real time behavior of the trashcan.

4.4 Chapter Summary

In this chapter we introduced the Selective Renderer Framework by analyzing the basic software components to be implemented. In the following chapter we are going to present how 3D scenes can be selectively rendered using the Selective Renderer Framework presented. A real time selective rendering application based on the *Selective Rendering Framework* (Zotos, Mania, Mourkoussis, 2009) is presented in detail.

In scientific research, an experiment (Latin: ex- periri, "to try out") is a method of investigating causal relationships among variables

Chapter 5

Implementation of a Selective Rendering System

In this chapter, a selective rendering system based on the *Selective Rendering Framework* (Chapter 4) is described in detail. Experimental design issues which are requisite in order to selectively display the scene utilized are also analyzed.

The Selective Renderer Framework was utilized to implement a selectively rendered 3d scene which was populated by consistent and inconsistent objects. An apartment scene was chosen

because it includes various areas and contexts. The selectively rendered scene will be evaluated against a fully-fledged rendering solution in order to ascertain that it induces similar spatial memory recognition performance compared to a scene comprising of objects of high polygon count. We can then conclude that computational savings resulted in similar spatial performance as when immersed in a computationally-heavy rendered solution. In this section, we will describe how the scene was assembled employing the Selective Renderer Framework described in Chapter 4 as well as the technical framework of interaction.

5.1 The 3D scene

Visual Content

In order to explore the effect of memory schemata, a 3d scene depicting an apartment was created which contained both consistent and inconsistent objects. A preliminary study was designed aiming to define the type and the degree of consistency of objects which could be found in an apartment scene.

In order to define which objects were consistent and which objects were inconsistent in relation to the context of the scene (type of consistency), a set of questionnaires were designed that contained a list of objects asking participants to "*Rate each object for how likely the object would to be appear in a room like this."* The methodology was similar to the one described in the *Brewer & Treyens, 1981* paper.

Pilot questionnaires were created containing a list of objects related to three areas in the apartment:

- A postgraduate's office, including
    - a desk sub area
    - a bookcase sub area
- A lounge area including
    - a sofa sub area
- A kitchen area

Each area and sub area represents a unique memory schema reflecting varied scene context. Six questionnaires were designed, each dedicated to an area in order to define which objects were either consistent or inconsistent. A sample questionnaire is presented below (see Appendix A for detailed questionnaires):

Section D: Schema expectancy questionnaire

Please read below:

In the following pages you will find a list of objects. Following each object there is a scale from 1 to 6. We would like you to use this scale in order to rate each object for how likely it is to appear in an academics office. As an example if you are in a restaurant carrying out this task you would probably give an object such as table a high number and an object such as bed a low number. When making your rating ignore whether or not the object is actually present in this current room. A response of 1 is defined as "extremely **unlikely** that the object would be **found** in a room like this", and a response of 6 is defined as "extremely **likely** that the object would be **found** in a room like this."

Please circle as you feel appropriate

Alien	not likely	1	2	3	4	5	6	likely
Android	not likely	1	2	3	4	5	6	likely
Barbeque	not likely	1	2	3	4	5	6	likely
Basketball	not likely	1	2	3	4	5	6	likely
Bicycle	not likely	1	2	3	4	5	6	likely

Figure 54: Sample questionnaire of an academics office.

Instructions were administered to participants (Figure 54) rating each object on a list in relation to how likely it was that the object could be found in the specific area of interest with a rating of 1 representing 'not likely' and 6 representing 'likely'. Six questionnaires were administered as follows:

- o A **kitchen** questionnaire (kitchen schema) which contained a list of 71 objects.

- o A **lounge** questionnaire (lounge schema) which contained a list of 75 objects associated with
    - o a **sofa** questionnaire (sofa sub-schema) which contained a list of 10 objects.

- o An **office** questionnaire (office schema) which contained a list of 45 objects associated with
    - o a **desk** questionnaire (desk sub-schema) which contained a list of 15 objects
    - o a **bookcase** questionnaire (bookcase sub-schema) which contained a list of 15 objects.

25 undergraduate, postgraduate and PhD students (15 male – 10 female) provided input. Objects were categorized according their expectant rating. Objects with a rating ranging between 3 and 4 were described as **neutral**, whereas objects with ratings between 4 and 6 where described as **expectant** and those with ratings between 1 and 3 as **non-expectant** accordingly.

- o 1<= *Non-expectant* (inconsistent) <= 3
- o 3< **Neutral** <4
- o 4<= **Expectant** (consistent) <=6

In this manner, we assumed that objects which were rated above 4, were considered as consistent or very likely to be found in the area in question. Similarly, objects which were rated under 3, were considered as inconsistent or very unlikely to be found in the area in question.

In summary:

In the kitchen area, objects were categorized as follows:

- o 29 non expectant objects (e.g. *a shovel* – 1.1 rate of consistency)
- o 4 neutral objects (e.g. *a phone* – 3.7 rate of consistency)
- o 38 expectant objects (e.g. *plates* – 5.8 rate of consistency)

In the lounge area, objects were categorized as follows:

- o 34 non expectant object (e.g. *a hammer* – 1.17 rate of consistency)
- o 9 neutral objects (e.g. *a hat* – 2.3 rate of consistency)
- o 23 expectant objects (e.g. *a tv* – 5.6 rate of consistency)

In the sofa sub area (over the sofas), objects were categorized as follows:

- o 5 non expectant objects (e.g. *a bottle of wine* – 2 rate of consistency)
- o 1 neutral objects (e.g. *keys* – 3.8 rate of consistency)
- o 4 expectant objects (e.g. *a remote control* – 5.6 rate of consistency)

In the office area, objects were categorized as follows:

- o 12 non expectant objects (e.g. *a trumpet* – 1.1 rate of consistency)
- o 2 neutral objects (e.g. *a poster* – 3.5 rate of consistency)
- o 31 expectant objects (e.g. *a computer mouse* – 5.8 rate of consistency)

In the desk sub area (over the desk), objects were categorized as follows:

- o 6 non expectant objects (e.g. *a fire extinguisher* – 1.1 rate of consistency)
- o 9 expectant objects (e.g. *a cd* – 4.4 rate of consistency)

In the bookcase sub area, objects were categorized as follows:

- o   13 non expectant objects (e.g. *a laptop* – 1.8 rate of consistency)
- o   1 neutral object (e.g*. a pencil* – 3.1 rate of consistency)
- o   1 expectant objects (e.g. *books* – 6 rate of consistency)

5.2 Creating the Selectively Rendered Scene

The next step was to create a complete 3d scene that combined both consistent and inconsistent items. Objects which were highly consistent or inconsistent in relation to the context of each area were randomly chosen from questionnaire results. 3d models were created or downloaded from 3D models' repositories (3d repositories) and were placed in the scene with the help of an industry-standard 3D modeling software (3d studio Max 2007).

The scene (Figure 55) included a kitchen, a lounge and an office area and comprised of consistent and inconsistent objects relevant to each context. The office area included two sub-regions, the desk area and the bookcase area. Similarly, the lounge area included a sofa area. The scene was designed so that it contained objects of varying level of scene consistency based on the area and sub-area they were placed. The scene consisted of basic modules such as walls, floors, ceilings, doors, etc (frame objects). According to *Brewer & Treyens, 1981, "the room frame contains the information about rooms that one can be nearly certain about before encountering a particular room".*

Figure 55: 3d scene - top view screen shot.

In particular, excluding frame objects, the kitchen area contained 17 consistent and 6 inconsistent objects.

Figure 56: Screen shot of the kitchen area.

The lounge area contained 17 consistent and 4 inconsistent objects. 5 out of the 17 consistent objects revolve to inconsistent when the sofa sub-schema is activated. For example, a bottle of wine is consistent in relation to the lounge area but, at the same time, it is inconsistent in relation to the sofa area according to the pilot study.

Figure 57 Screen shot of the lounge area and the sofa sub area.

The office area contained 25 consistent and 1 inconsistent objects. 16 out of the 25 consistent objects revolve to inconsistent when the sub-schema is activated. For example, a fire extinguisher is consistent in relation to the office area but, at the same time, it is inconsistent in relation to the desk area according to the pilot study.

Figure 58: Screen shot of the office area and the desk sub area.

Figure 59: Screen shot of the office area and the bookcase sub area.

5.2.1 Radiosity Solution

Radiosity calculates diffuse reflections in a scene and results in a finely divided geometrical mesh. Heat transfer theory describes radiation as the transfer of energy from a surface when that surface has been thermally excited. This encompasses both surfaces that are basic emitters of energy, as with light sources and surfaces that receive energy from other surfaces and thus have energy to transfer.

In order to simulate light propagation, the radiosity algorithm was utilized in order to render the scene. In order to apply the radiosity solution, the following parameters should be defined:

- o **Number of iterations**. The maximum number of Radiosity iterations. The radiosity engine bounces rays around the scene and distributes energy on surfaces. Between each iteration, the engine measures the amount of variance (noise between surfaces) that was computed. As the number of algorithmic iterations of surface light propagation increases, it improves the radiosity's shading accuracy and polygon count.

- o **Min and max mesh size**. Max mesh size is the size of the largest faces after adaptive subdivision. Faces are not divided smaller than the minimum mesh size.

- o **Initial quality %.** This parameter sets the quality percentage at which to stop the Initial Quality stage, up to 100%. For example, if the initially quality is set to 80%, the result is a radiosity solution that is 80% accurate in relation to total energy distribution. A goal of 80 to 85% is usually sufficient for good results.

- o **Global subdivision settings**. This parameter turns on the radiosity mesh for the entire scene.

- o **Regather indirect illumination**. In addition to recalculating all the direct lighting, the algorithm recalculates the indirect lighting at each pixel by regathering illumination data from the existing radiosity solution. Using this option, the most accurate, artifact-free images can be produced, adding, however, a considerable amount of rendering time.

- o **Photometric lights**. Photometric lights use photometric (light energy) values that enable the user to more accurately define lights as they would be in the real world. The user can create lights with various distribution and color characteristics, or import specific photometric files available from lighting manufacturers.

Radiosity algorithms display view-independent diffuse inter-reflections in a scene assuming the conservation of light energy in a closed environment. The surfaces of objects are divided into patches or elements. Despite transmitting energy to others, a patch will also reflect the energy from other meshes that arrives on its surface into the scene. These processes will be iterated until energy equilibrium in the close space is achieved. Radiosity produces color-bleeding effects from one surface to another, shades inside the shadow area and creates soft-edge shadow with penumbrae along shadow boundaries. All of these results imitate the physical propagation of light in the real environment. The number of algorithmic iterations of surface light propagation as increased, it improves the radiosity's shading accuracy and polygon count.

Figure 60 and Figure 61 show that the radiosity scene consisted of a higher polygon count compared to the initial flat-shaded geometry used as input. This is predictable since the radiosity algorithm subdivides the initial scene in finely-meshed regions in order to apply shadow information and physical energy rendering. In the lounge area (Figure 60, right), the regions which require high level of shadow information such as the areas under the sofas and the table have been subdivided in finer polygon meshes compared to other floor regions.
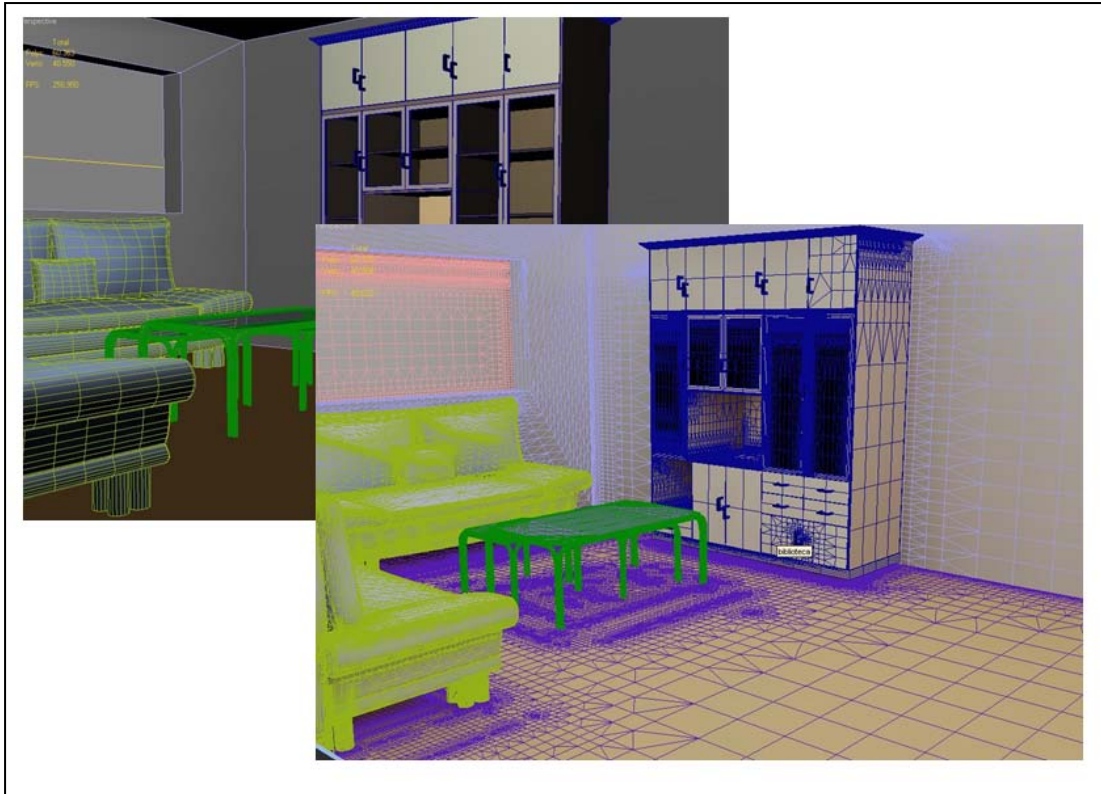
Figure 60: Left: The lounge with highlighted polygons before applying the radiosity solution. Right: The lounge with highlighted polygons after applying the radiosity solution.
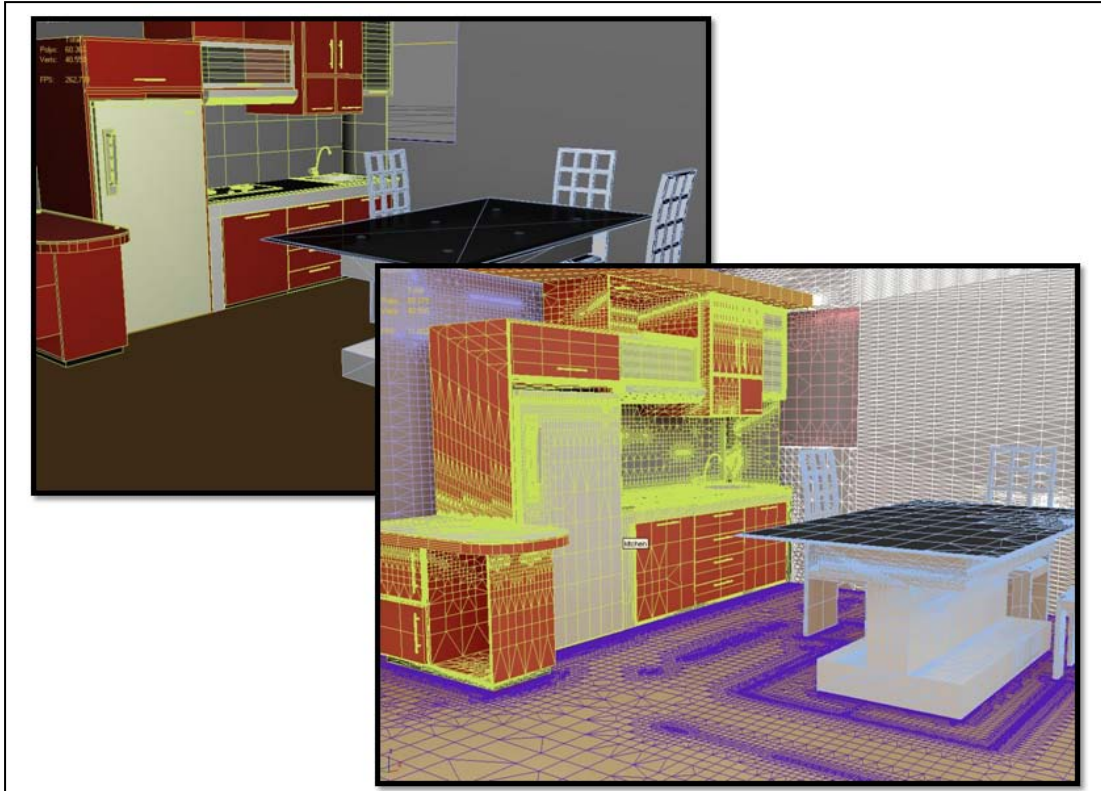
Figure 61 Left: The kitchen with highlighted polygons before applying the radiosity solution. Right: The kitchen with highlighted polygons after applying the radiosity solution.

Figure 62(left), Figure 63(left) and Figure 64((left)) show the areas of the scene before applying the radiosity solution. The same figures (right) present the areas of the scene after applying the radiosity solution. Shadows and simulation of physical energy propagation (right) make the scene more realistic.
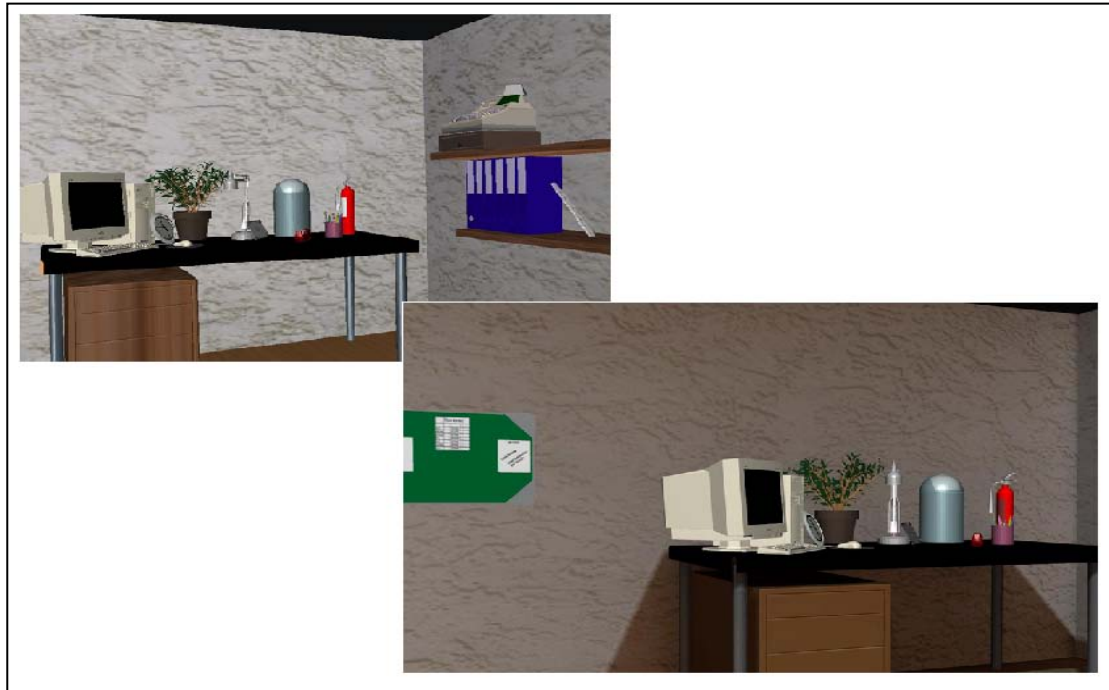
Figure 62: Left: The office area before applying the radiosity solution. Right: The office area after applying the radiosity solution. Shadows and simulation of physical energy propagation (right) make the scene more realistic.

Figure 63: : Left: The kitchen before applying the radiosity solution. Right: The kitchen after applying the radiosity solution. Shadows and simulation of physical energy propagation (right) make the scene more realistic.

Figure 64: Left: The lounge before applying radiosity solution. Right: The lounge after applying radiosity solution. Shadows and simulation of physical energy propagation (right) make the scene more realistic.
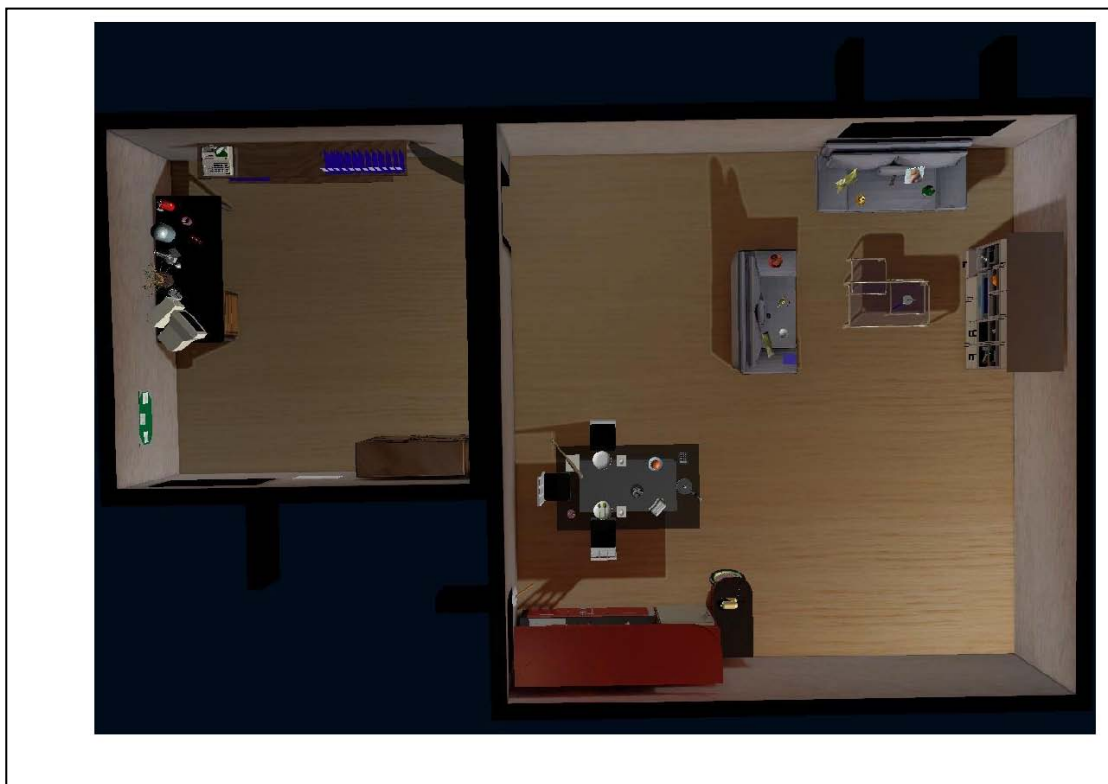
Figure 65: Top view of the final scene. Three main areas: the kitchen area, the lounge area and the office area. The lounge area includes the sofa sub area and the office area includes the desk sub area and the bookcase sub area.

5.3 Variations of Polygon Count

As analyzed in Chapter 4, consistent objects which are expected to be found in a scene are rendered in lower quality to the point of being recognizable whereas inconsistent items which are salient would require a high level of rendering. Therefore, the next step in order to assemble the final scene is to produce models of varying polygon count related to the consistent objects since inconsistent objects require only high rendering quality.
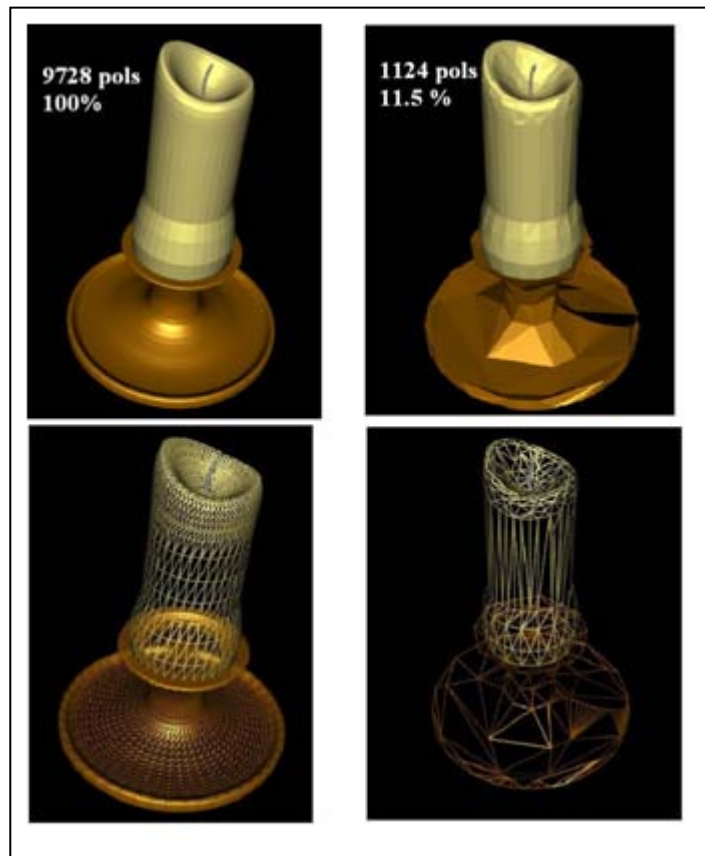
Figure 66 Left up is the original candle model and right up is the diminished version. Wireframe editions of the objects appear at the bottom where the polygon quality difference of the models can be noticed better.

To maintain consistency of appearance the polygons of each object were reduced on an individual basis until it was felt that the object contained the minimum number of polygons required to retain the object's recognizable shape.

Thus, scenes were created in which all objects communicated the same perceived visual quality per condition, regardless of polygon count per object, as determined by detailed pilot studies. Polygon count reduction analysis and statistics are presented below.

Kitchen Objects		
Name	Polygons	Percentage
Saltcellar_v1	32245	100%
Saltcellar_v2	16808	50%
Fruits_v1	4922	100%
Fruits_v2	678	14%
Glasses_v1	4612	100%
Glasses_v2	1932	40%
Hamburger_v1	24282	100%
Hamburger_v2	1541	6,30%
Mixer_v1	32188	100%
Mixer_v2	3757	11,60%
Plates_v1	130368	100%
Plates_v2	4836	3,70%
Saucepan_v1	7960	100%
Saucepan_v2	683	8,50%
Spoons_v1	7680	100%
Spoons_v2	376	5%
Teapot_v1	2256	100%
Teapot_v2	964	42%
Toaster_v1	1042	100%
Toaster_v2	354	33%
KitchenTable_v1	2512	100%
KitchenTable_v2	1100	40%
KitchenFurniture_v1	14413	100%
KitchenFurniture_v2	3362	25%

Figure 67: Polygon count reduction statistics of the kitchen objects. Version 1 (_v1) is the original high polygon detail version of objects and version 2 (_v2) is the diminished polygon count version.

For example (Figure 67), spoon_v1 is the original version of the model (high quality) and consists of 7680 polygons (100%) while spoon_v2 is the low quality version which consists of 376 polygons (5% of the initial quality or 95% reduction).
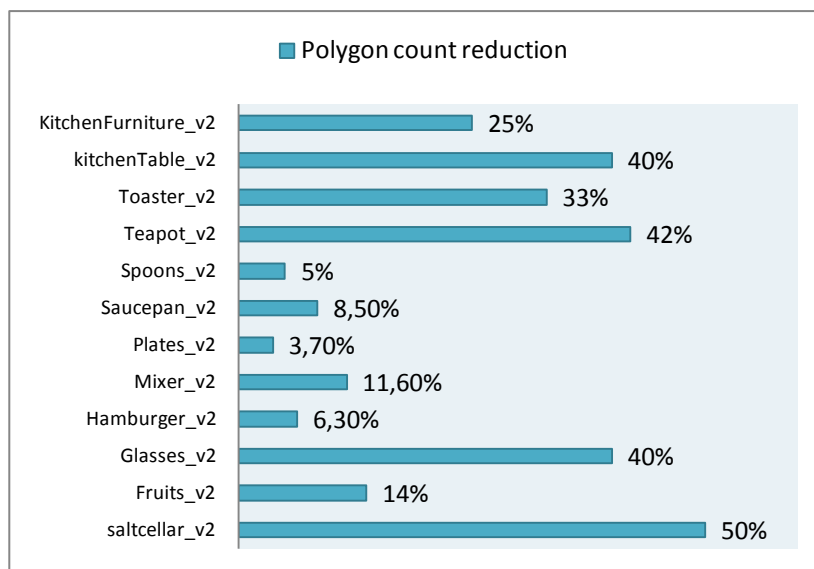
Figure 68: Polygon count reduction statistics of the kitchen objects. Listed bars represent the percentage of initial polygon quality. For example, spoon_v2 has 5% of the initial quality or 95% reduction.

Lounge - Sofa Objects		
Name	Polygons	Percentage
ashTray_v1	104	100%
ashTray_v2	50	50%
Candle_v1	9728	100%
Candle_v2	1124	11,50%
cigaretteSet_v1	896	100%
cigaretteSet_v2	432	50%
glass_v1	720	100%
glass_v2	124	17,20%
Keys_v1	1560	100%
Keys_v2	638	40%
loungeBooks_v1	5032	100%
loungeBooks_v2	170	3,50%
magazine_v1	60088	100%
Magazine_v2	1530	2,50%
Mobile_v1	15503	100%
Mobile_v2	2191	14%
remoteControl_v1	43564	100%
remoteControl_v2	1046	2,30%
Sofa1Pillow_v1	4608	100%
Sofa1Pillow_v2	123	2,60%
Sofa2Pillow_v1	4608	100%
Sofa2Pillow_v2	123	2,60%
Speakers_v1	138	100%
Speakers_v2	70	50%
Tv_v1	4044	100%
Tv_v2	1905	45%
Vase_v1	1225	100%
Vase_v2	611	50%
Wine_v1	14990	100%
Wine_v2	888	6%
Biblioteca_v1	2774	100%
Biblioteca_v2	1660	60%
loungeTable_v1	25174	100%
loungeTable_v2	7814	30%
Sofa1_v1	7440	100%
Sofa1_v2	654	8,70%
Sofa2_v1	7440	100%
Sofa2_v2	654	8,70%

Figure 69: Polygon count reduction statistics of the lounge objects. Version 1 (_v1) is the original high polygon detail version of objects and version 2 (_v2) is the diminished polygon count version.
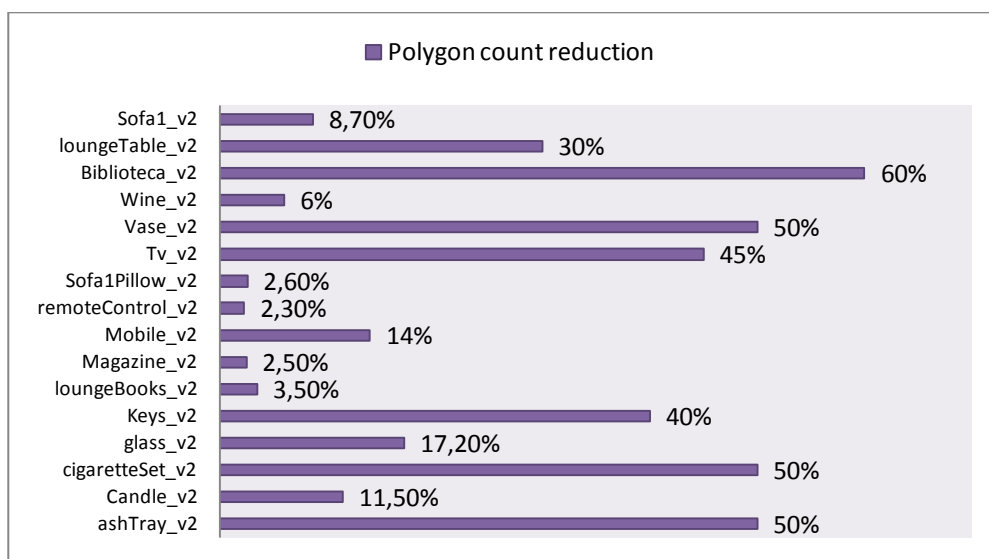
Figure 70: Polygon count reduction statistics of lounge objects. Listed bars represent the percentage of initial polygon quality. For example, wine_v2 has 6% of the initial quality or 94% reduction.

Office – Desk - Bookcase Objects		
Name	Polygons	Percentage
Printer_v1	21621	100%
Printer_v2	12577	60%
TrashCan_v1	1064	100%
TrashCan_v2	93	10%
Clock_v1	1556	100%
Clock_v2	660	40%
Selotape_v1	43940	100%
Selotape_v2	3662	4%
pencilHolderDesk_v1	880	100%
pencilHolderDesk_v2	432	50%
computerSet_v1	31942	100%
computerSet_v2	14666	40%
Cd_v1	3406	100%
Cd_v2	1501	50%
fireExtinguisher_v1	5374	100%
fireExtinguisher_v2	1208	22%
Plant_v1	10608	100%
Plant_v2	7996	75%
deskLamp_v1	22459	100%
deskLamp_v2	11250	50%
Mug_v1	4668	100%
Mug_v2	319	6,80%
Laptop_v1	49904	100%
Laptop_v2	1885	4%
libraryBooks_v1	55352	100%
libraryBooks_v2	1882	3,50%
libraryMouse_v1	11324	100%
libraryMouse_v2	328	3%
telephoneLibrary_v1	18343	100%
telephoneLibrary_v2	1343	7%
libraryPencilHolder_v1	1492	100%
libraryPencilHolder_v2	940	60%
Ruler_v1	896	100%
Ruler_v2	426	50%
Desk_v1	490	100%
Desk_v2	301	60%
bookcase_v1	986	100%
Bookcase_v2	350	30%

Figure 71: Polygon count reduction statistics of the office objects. Version 1 (_v1) is the original high polygon detail version of objects and version 2 (_v2) is the diminished polygon count version.
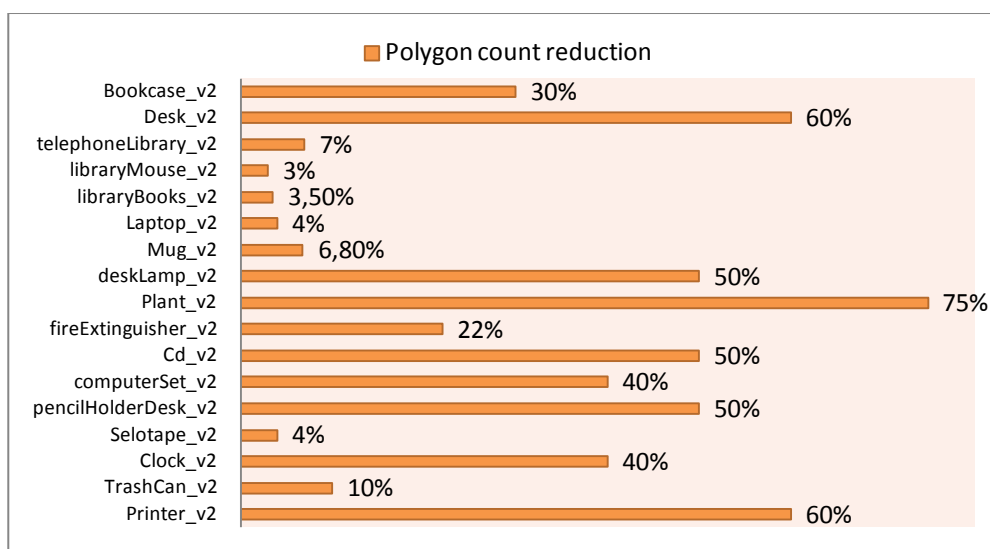
Figure 72: Polygon count reduction statistics of the office objects. Listed bars represent the percentage of initial polygon quality. For example, Laptop_v2 has 4% of the initial quality or 96% reduction.

5.4 Assembling the Selectively Rendered Scene

After creating the final radiosity scene the next step is to produce the real time selectively rendered scene by enriching the scene graph with appropriate metadata information as analyzed in Chapter 4. Based on the selective rendering framework, the scene s will dynamically upload different polygon versions of objects in relation to which memory schema is activated.

5.4.1 Switch Nodes

In order to dynamically upload the appropriate polygon count version of an object, x3d *switch* nodes are defined as described in Chapter 2 which contain alteration directions. For example, the polygon count of a candle model will change in real-time depending on its level of scene context consistency driven by an x3d *switch* node defined as follows:

```
<Transform DEF='Transform_CandleDefinition' containerField="children" scale='0.8447 0.8447 0.8447'
  <Switch DEF="Switch_CandleDefinition"  containerField="children" whichChoice="1">
      <Inline DEF="Inline_Candle_v1_Definition"  url="inlineNodes/candle_v1.x3d"/>
      <Inline DEF="Inline_Candle_v2_Definition"  url="inlineNodes/candle_v2.x3d"/>
  </Switch>

</Transform>
```

Figure 73: *Switch* node example related to the candle model. Two different versions of the candle are loaded but only one is rendered each time. The WhichChoice field points to the version that should be rendered.

Two different versions of the same model (candle_v1.x3d, candle_v2.x3d) are available, but only one is rendered at each time. The *WhichChoice* x3d field points to the version that should be rendered. This procedure is applied to all objects that will be dynamically uploaded to the system.

5.4.2 Proximity Nodes

The *Switch* nodes that have been created point to different polygon count versions of objects. These versions will change in real time in relation to the activated schema. In order to "sense" which schema is activated, *proximity* sensor nodes should be defined for all areas and sub areas representing a unique memory schema. It is crucial to know when the navigation avatar enters or exits a specific area (or context of the scene) in order to guide system decisions concerning which polygon version of each objects should be rendered.

Three *proximity* sensors have been defined (Figure 74, Figure 75), One for the sofa sub area (A), one for the desk sub area (B) and one for the bookcase sub area (C).
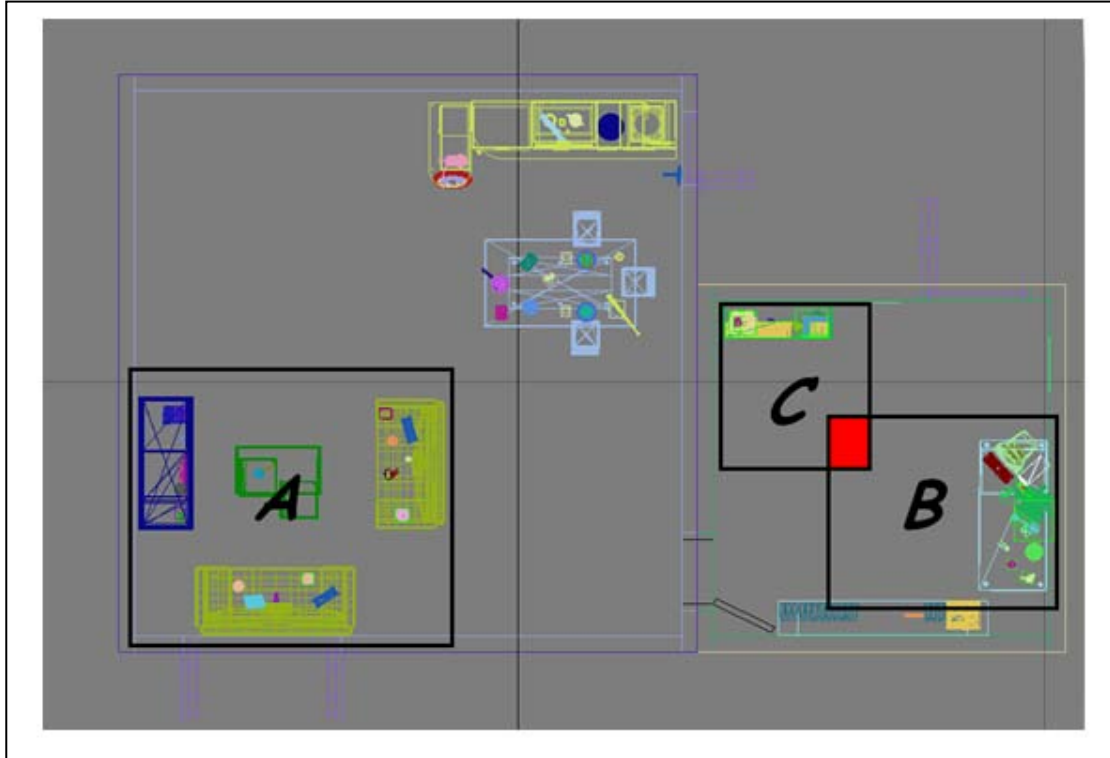
Figure 74: 2D top view screen shot of the scene which shows three areas related to the three *proximity* sensors. Area A is the sofa sub area, Area B is the desk sub area, Area C is the bookcase sub area.
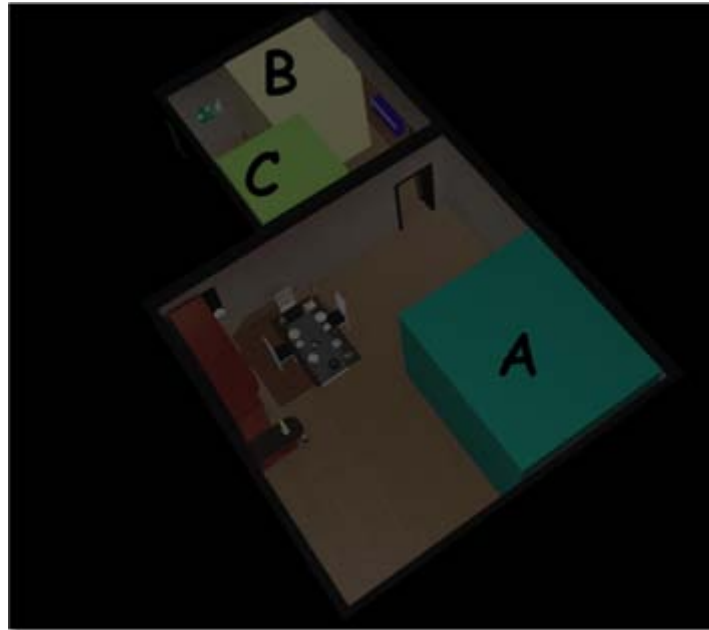
Figure 75: 3D top view screen shot of the scene which shows three areas related to the three *proximity* sensors. Area A is the sofa sub area, Area B is the desk sub area, Area C is the bookcase sub area.

The X3D code that defines these *proximity* nodes is the following:

```
<Transform DEF='boxLibrary' scale='1.078 2.368 1.338' translation='3.744 -1.93 -1.107'>
    <Transform translation='0.0 0.8219 0.0'>
        <ProximitySensor DEF='BookcaseProx ' size='2.643 1.644 3.159'> </ProximitySensor>
    </Transform>
</Transform>
<Transform DEF='boxDesk' scale='1.809 2.182 1.0' translation='5.779 -1.946 0.7717'>
    <Transform translation='0.0 0.8541 0.0'>
        <ProximitySensor DEF='DeskProx' size='3.255 1.708 3.449'> </ProximitySensor>
    </Transform>
</Transform>
<Transform DEF='boxLounge' scale='1.14 1.058 1.136' translation='-5.461 -1.963 0.4947'>
    <Transform translation='0.0 1.76 0.0'>
        <ProximitySensor DEF=' SofaProx ' size='5.727 3.52 4.799'> </ProximitySensor>
    </Transform>
</Transform>
```

Figure 76 X3D code that defines 3 *proximity* nodes. Bookcase *Proximity* (in office area), Desk *Proximity* (in office area) and Sofa *Proximity* (in lounge area).

As we can see above, the X3D code defines 3 *proximity* nodes as follows: The *Bookcase Proximity* representing the bookcase in the office area, the *Desk Proximity* representing the desk in the office area and the *Sofa Proximity* representing the bookcase in the lounge area. For each *proximity* node various parameters have been defined such as the size and the translation of the *proximity* area.

5.4.3 Metadata Enrichment

As analyzed in Chapter 4, metadata information should be defined for every *proximity* node of the scene.

It is crucial to attach metadata information to every *proximity* node which describes whether objects are consistent or inconsistent in relation to the context of each sub-area (Figure 77). When the navigation avatar approaches a scene's region, this action is acknowledged and by exploiting relevant metadata information it is feasible to know which knowledge schema should be activated. 'Activation' will occur when the user navigates closely to each area.

```xml
<ProximitySensor DEF=' SofaProx ' size='5.727 3.52 4.799'>
<MetadataSet  containerField='metadata' DEF="MetadataSet_LoungeProxDefinition'>
    <MetadataString DEF="enterTimeSchemaDefinitionLoungeProx" name="enterTimeSchema' value='SofaSchema'
    <MetadataString DEF="exitTimeSchemaDefinitionLoungeProx" name="exitTimeSchema' value="LoungeSchema'
 </MetadataSet>
 </ProximitySensor>
```

```xml
<ProximitySensor DEF='DeskProx' size='3.255 1.708 3.449'>
<MetadataSet  containerField='metadata' DEF="MetadataSet_OfficeProxDefinition">
    <MetadataString DEF="enterTimeSchemaDefinitionOfficeProx" name="enterTimeSchema' value="DeskSchema'
    <MetadataString DEF="exitTimeSchemaDefinitionOfficeProx" name="exitTimeSchema' value='OfficeSchema'
 </MetadataSet>
</ProximitySensor>
```

```xml
<ProximitySensor DEF=BookcaseProx' size='2.643 1.644 3.159'>
<MetadataSet  containerField='metadata' DEF="MetadataSet_LibraryProxDefinition'>
    <MetadataString DEF="enterTimeSchemaDefinitionLibraryProx" name='enterTimeSchema' value=BookcaseSchema
    <MetadataString DEF="exitTimeSchemaDefinitionLibraryProx" name='exitTimeSchema' value="OfficeSchema' c
 </MetadataSet>
 </ProximitySensor>
```

Figure 77: Metadata information of the *proximity* nodes. *EnterTimeSchema* will be activated when the avatar enters the specific area and *exitTimeSchema* will be activated when the avatar exits the specific area which is defined by a *proximity* node.

Therefore, metadata information defines which schema should be activated when the avatar approaches a region. In detail:

- When the avatar enters the Sofa Area, the *SofaSchema* (Sub-Schema) is activated.

- When the avatar exits the Sofa Area, the *LoungeSchema* (Main Schema) is activated.

- When the avatar enters the DeskArea, the *DeskSchema* (Sub-Schema) is activated.

- When the avatar exits the DeskArea, the *OfficeSchema* (Main Schema) is activated.

- When the avatar enters the Bookcase Area, the *BookcaseSchema* (sub-Schema) is activated.

- When the avatar exits the Bookcase Area, the *OfficeSchema* (main Schema) is activated.

The Kitchen area doesn't require any *proximity* nodes and metadata enrichment since there is only one active general schema and there is no need to alter objects in real time.

The next step of the metadata enrichment procedure is to devise a method that can relate the activated 'memory schema' with the appropriate polygon quality version of each object. As analyzed in Chapter 4, the *Switch* nodes indicate which version of the object should be rendered depending on user navigation. Thus, metadata information should be attached to every *switch* node which associates each version of the object with the appropriate level of consistency in relation to each context (Figure 78).

```
<Switch DEF="Switch_TrashCanDefinition"  containerField="children" whichChoice="1">
   <Inline DEF="Inline_TrashCan_v1_Definition"  url="inlineNodes/trashCan_v1.x3d"/>
   <Inline DEF="Inline_TrashCan_v2_Definition"  url="inlineNodes/trashCan_v2.x3d"/>


<MetadataSet  containerField='metadata' DEF="MetadataSet_TrashCanDefinition">
   <MetadataSet  containerField='value' DEF="MetadataSetTrashCan_Part1Definition" >
      <MetadataString DEF="consistencyTypeDefinitionTrashCan_Part1" name="consistencyType" value="consistent" conta:
      <MetadataString DEF="degreeOfConsistencyDefinitionTrashCan_Part1" name="degreeOfConsistency" value="high" cont
      <MetadataString DEF="schemaBelonginessDefinitionTrashCan_Part1" name="schemaBelonginess" value="OfficeSchema"
   </MetadataSet>
   <MetadataSet  containerField='value' DEF="MetadataSetTrashCan_Part2Definition" >
      <MetadataString DEF="consistencyTypeDefinitionTrashCan_Part2" name="consistencyType" value="consistent" conta:
      <MetadataString DEF="degreeOfConsistencyDefinitionTrashCan_Part2" name="degreeOfConsistency" value="low" conte
      <MetadataString DEF="schemaBelonginessDefinitionTrashCan_Part2" name="schemaBelonginess" value="DeskSchema" cc
   </MetadataSet>
</MetadataSet>
</Switch>
```

Figure 78: *Switch* **node metadata information of a trashcan model. The trashcan is consistent (high degree of consistency) for the office schema and inconsistent (low degree of consistency) for the desk schema.**

The X3D code defines a *switch* node which contains two versions of a trashcan model. Two *metadataSet* nodes have also been defined which determine the degree of consistency for each version. The low polygon count version is linked to the office schema. Similary, the high polygon count version is linked to the desk schema (*schemaBelonginess*).

The metadata information example above describes the rendering behavior of a trashcan that is placed over the desk of the office (Figure 79).
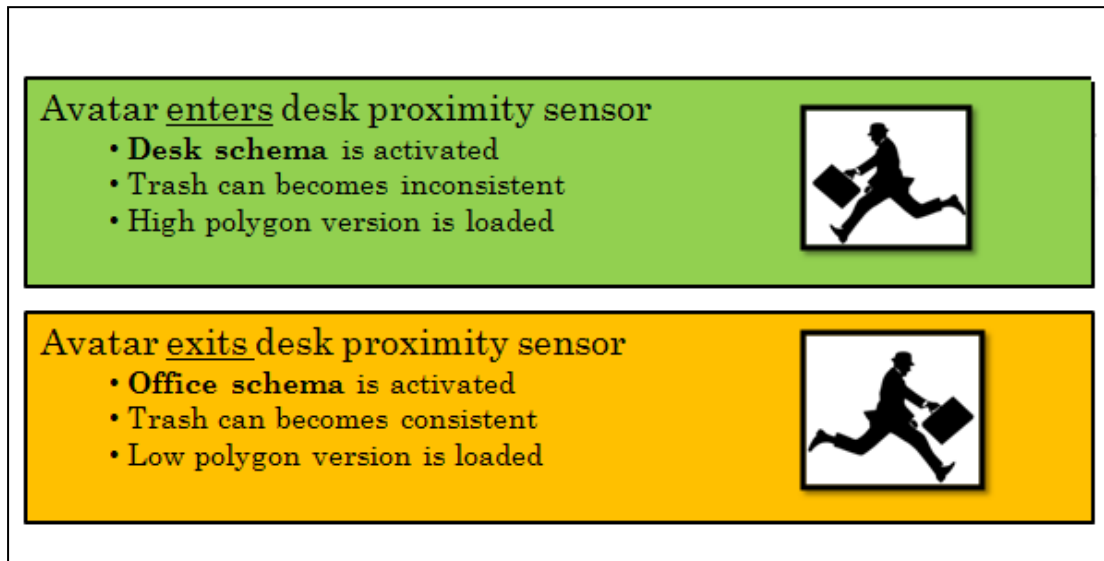
Figure 79: Real time rendering behavior of the trashcan.

When the avatar enters the desk *proximity* sensor, the desk schema is activated and the trashcan becomes inconsistent in relation to the desk schema. As a result, the high polygon version of the model is loaded. Similarly, when the avatar exits the desk *proximity* sensor, the office schema is activated and the trash can becomes consistent in relation to the office schema. As a result, the low polygon version is loaded.

The trashcan is consistent in relation to the general main office schema (high degree of consistency) and should be rendered in low quality (Figure 80). Furthermore, the same trashcan is inconsistent in relation to the desk sub-schema and should be rendered in high quality (Figure 81).

Figure 80: Low polygon quality version of the trashcan model. The trashcan is consistent in relation to the office schema.

As shown in Figure 80, the avatar has not entered the desk sub area yet. Therefore, the desk sub-schema has not been activated. The trashcan is consistent in relation to the main office schema and it is subsequently rendered in low polygon quality.

Figure 81: High polygon quality version of the trashcan model. The trashcan is inconsistent in relation to the desk schema.

As shown in Figure 81, the avatar has entered the desk sub area and the desk sub-schema has been activated. As a result, the trashcan is inconsistent in relation to the desk sub-schema. Therefore, a high polygon count version of the model is loaded.

5.5 Chapter Summary

This chapter provided detailed information concerning the implementation of the real time selective rendering application which is based on the *Selective Rendering Framework*. The next chapter presents an evaluation study of the real-time selectively rendered scene by comparing it to a high quality rendering solution. The task utilized for this evaluation is memory recognition based on schemata, drawn from formalized methodologies of memory psychology as well as formalized statistical manipulations of data..

"Dubito ergo cogito; cogito ergo sum"   .
(I doubt, therefore I think; I think therefore I am)
Rene Descartes quotes (French Mathematician,
Philosopher and Scientist, 1596-1650)
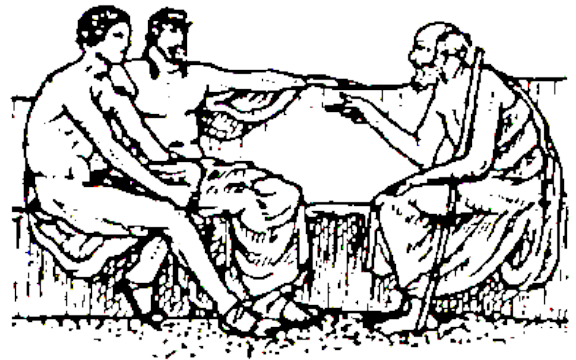
Chapter 6

Evaluation

The next chapter presents an evaluation study of the real-time selectively rendered scene by comparing it to a high quality rendering solution. The task utilized for this evaluation is memory recognition based on schemata, drawn from formalized methodologies of memory psychology. We predict that the polygon count savings of the selectively rendered system will not diminish spatial awareness while exposed to the immersive simulation implemented. The selectively rendered scene is displayed on a stereo-capable, head tracked Head Mounted Display.

The Main Experiment

In order to evaluate spatial awareness while exposed to a selectively rendered scene based on memory schema in relation to a fully-fledged rendering solution, a scene was designed which included several areas and sub-areas and a memory recognition task was completed (Figure 82). The scene included a kitchen, a lounge and an office area and comprised of consistent and inconsistent objects relevant to each context. The office area included two sub-regions, the desk area and the bookcase area. Similarly, the lounge area included a sofa area. The scene was designed so that it contains objects with differing levels of scene consistency based on the area and sub-area they were placed. The polygon quality of such objects could, therefore, could change dynamically. For example, according to the pilot studies which were presented in Chapter 5, a fire extinguisher is consistent in relation to the office schema but it is inconsistent in relation to the desk schema. Thus, this object was considered to be consistent and rendered with low detail when the office area was navigated, however, it was considered as inconsistent and rendered with high detail when the desk area was approached.
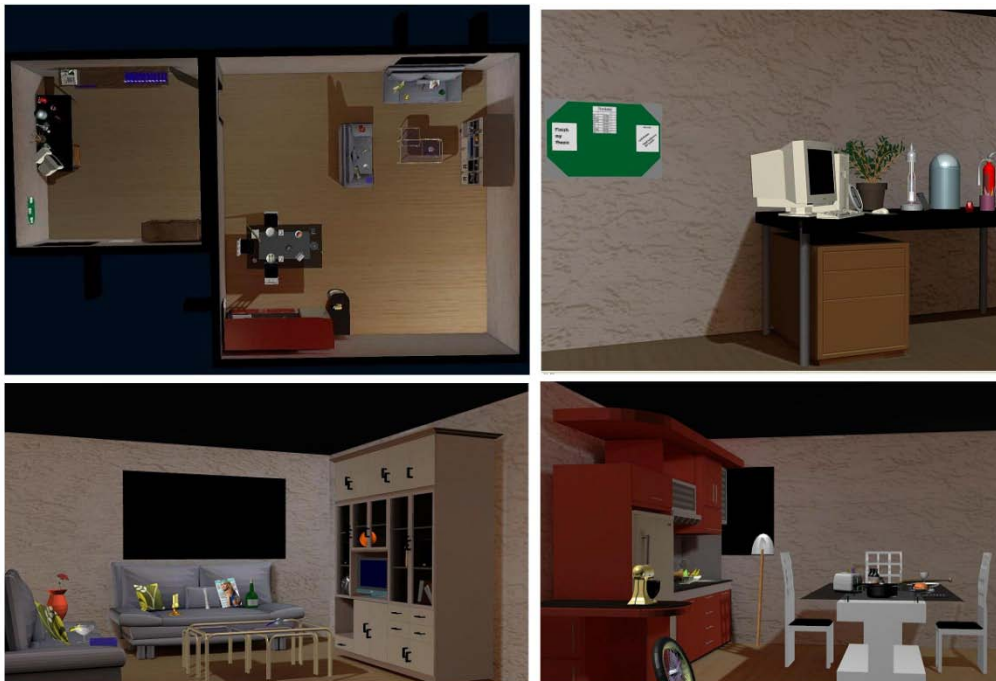
Figure 82: Main experimental rooms (high polygon quality version).

6.1 Materials and Methods

6.1.1 Apparatus

The VEs were presented in stereo at VGA resolution on a Kaiser Electro-optics Pro-View 50 Head Mounted Display with a Field-of-View comprising 50 degrees diagonal. An Intersense Intertrax2, three degree of freedom tracker was utilized for rotation. The viewpoint was set in the middle of the virtual room and navigation was restricted to a 360 degrees circle around that viewpoint and 180 degrees vertically (rotation). Participants were sitting on a swivel chair during exposure. The application ran on a standard PC with an average cost graphics card. Participants could navigate through the scene and move forward/backward by using a wireless mouse (right click – forward, left click - backward).

6.1.2 Visual Content

According to the group they were assigned to, participants were exposed to an interactive pre computed radiosity simulation of a synthetic scene in one of the 'polygon count' conditions:

- All objects of the scene were rendered in high quality, referred to as the 'high poly' condition.

- The consistent objects were rendered in low quality, the inconsistent objects were rendered in high quality and the objects of varying level of consistency, which was based on sub-area activated were rendered in high quality. The condition described above is referred to as the 'high-low poly' condition.

- The consistent objects were rendered in low quality, the inconsistent objects were rendered in high quality and the objects of varying level of consistency based on sub-area activated were rendered either in low or high quality based on their degree of consistency in relation to the sub-area entered. The condition described above is referred to as the 'selective poly' condition.

In detail, the scene viewed contained:

- A kitchen area including 23 objects from which 17 were consistent and 6 inconsistent.

- A lounge area comprising of a sofa sub-area including 21 objects from which 12 were consistent, 4 inconsistent and 5 of a different level of consistency in relation to the sub-area entered. For example, a bottle of wine is consistent in relation to the lounge schema but inconsistent in relation to the sofa schema.

- An academic office area comprising of a desk sub-area and a book case sub-area including 26 objects from which 9 were consistent, 1 inconsistent and 16 of a different level of consistency in relation to the sub-area entered. For example, a fire extinguisher is consistent in relation to the office schema but inconsistent in relation to the desk schema.

6.1.3 Participants

54 participants were recruited belonging to the research population of the Technical University of Crete, their age ranging from 18-25. The 54 participants formed 3 balanced groups of 18, corresponding to the three viewing conditions. Participants in all conditions were naïve as to the purpose of the experiment. All participants had normal or corrected to normal vision and no reported neuromotor impairment.

Figure 83: Participant under experimental conditions.

6.1.4 Procedure

Participants were led to believe that the main experiment was just a practice phase before the actual main experiment took place, aiming to familiarise them with the HMD (see Appendix C for a detailed description). The reason for this was to prevent the participants from being aware of the experimental task prior to exposure and avoid the development of mnemonics. Participants were given identical instructions across conditions. The room where the experiment was taking place was kept dark during exposure.

All participants had time to feel comfortable with the apparatus and 3D environments during the practice phase. A practice scene was created which included 3D primitive shapes such as boxes and cylinders (Figure 84, Figure 85). After exposure, the main scene of the experiment was uploaded while participants were still led to believe that this was just a practice phase.
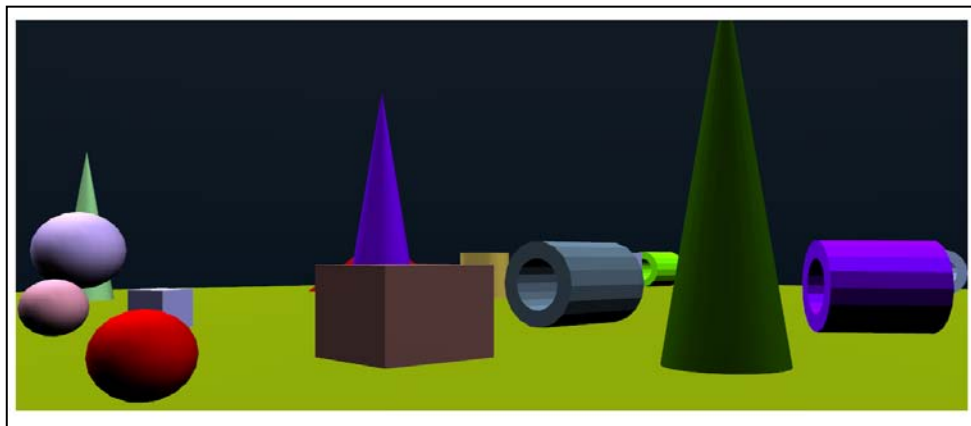
Figure 84: Participants were instructed to navigate through shapes in order to feel comfortable with the apparatus.
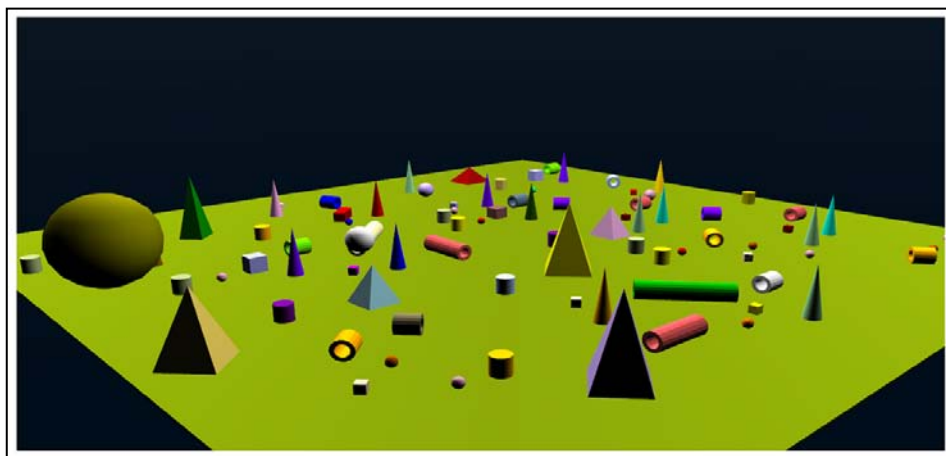
Figure 85: Practice scene top view screen shot.

The experiments were conducted over two weeks during April 2009. Participants completed a memory recognition questionnaire after 135 seconds (45 seconds in each area) of exposure to either the 'high' or 'high–low' or 'selective poly' viewing condition. The time of exposure was defined after detailed pilot studies which ensured that there were no apparent floor (the task being too hard) or ceiling (the task being too easy) effects.

The memory questionnaire (Figure 86, see Appendix D for detailed description) consisted of 6 sections, e.g. 3 sections containing objects in relation to the kitchen, lounge and academic office areas and 3 sections in relation to the sofa, desk and book case sub-areas. Each section contained of a list of objects which fell under 4 distinct categories:

1. Schema-consistent objects that were present in the scene (e.g. saucepan, toaster).

2. Schema-inconsistent objects that were present in the scene (e.g. shovel).

3. Schema-consistent objects that were absent (e.g. forks).

4. Schema-inconsistent objects that were absent (e.g. kite).

Please read below:

Reflecting on your brief exposure to the academics office, consider the objects listed below. Please circle Yes to indicate if you believe the object was present in the room, or No to indicate if you believe the object was not present in the room. Then also circle a number to indicate your confidence about your answer, where 5 indicates absolute certainty and 1 indicates no confidence.

Please circle as you feel appropriate

	Was the object present?			How confident are you about your answer?				
				No conf	Low conf	Moderate	Confident	Certain
Donkey	Yes	No		1	2	3	4	5
Basketball	Yes	No		1	2	3	4	5
Hammer	Yes	No		1	2	3	4	5
Pillow	Yes	No		1	2	3	4	5
Golf/Tennis table/Billiard ball	Yes	No		1	2	3	4	5
Scissors	Yes	No		1	2	3	4	5
Plant	Yes	No		1	2	3	4	5
Coffee mug	Yes	No		1	2	3	4	5
Picture frame	Yes	No		1	2	3	4	5
Viking helmet	Yes	No		1	2	3	4	5
Desk	Yes	No		1	2	3	4	5

Figure 86: Sample of final memory recognition questionnaire.

Participants were instructed to answer 'yes' if the object was present in the scene or 'no' if the object was absent. Participants also indicated on a scale of 1 to 5 their confidence concerning the presence or absence of each object, where 1 indicated no confidence and 5 absolute certainty.

The list of objects was assembled based on an initial pilot study which explored which objects were expected to be found in each area and which were not (section 5.1). 25 participants ranked the objects on the list. The consistency of each item was rated on a scale from 1 to 6 according to whether they expected to find each object in each area, or not with 6 being the most expected and 1 being the least. Based on these ratings, consistent objects were selected from the high end of the scale and the inconsistent ones from the low end.

Participants were then led to another room, where they were asked to indicate whether they felt that each object mentioned in the memory recognition list given was present in the environment to which they had been exposed or not. They were asked to rate their confidence in each response using a scale of 1 to 5, with 1 being not confident and 5 being certain. These ratings are referred to as confidence ratings. The amount of time between exposure and memory testing was the same across conditions.

As suggested by Brewer & Treyens 1981, the amount of time spent looking at an individual object may affect memory encoding. Whilst neither the Brewer & Treyens experiment nor this research included systems necessary to track eye movement, a record of each test participant's head movement was monitored through the use of a script. The script used the orientation of the tracker to ascertain which objects were being viewed and at what point in time and whether the HMD was moving at the time, or had not moved since the previous frame (idle). Although this provided useful approximate information, it cannot be said with certainty whether or not a person was actually looking at the object that the tracker was directed at. This is because of the variation in position between the tracker and the eyes and because of the assumption that a person was at all times looking straight through the HMD without rolling eyes from side to side or blinking. The tracking data was sufficiently detailed however, to justify excluding one participant's self-report data based on the fact that the room had not been fully viewed.

6.2 Polygon analysis

Three different polygon versions of the experimental scene were created:

- All objects of the scene were rendered in high quality, referred to as the 'high poly' condition.

- The consistent objects were rendered in low quality, the inconsistent objects were rendered in high quality and the objects of varying level of consistency, which was based on sub-area activated were rendered in high quality. The condition described above is referred to as the 'high-low poly' condition.

- The consistent objects were rendered in low quality, the inconsistent objects were rendered in high quality and the objects of varying level of consistency based on sub-area activated were rendered either in low or high quality based on their degree of consistency in relation to the sub-area entered. The condition described above is referred to as the 'selective poly' condition.

Each polygon count version of the the experimental scene was comprised of a specific number of polygons. This section presents the polygon count statistics of each condition. The Kitchen area is represented by one main schema referred to as the kitchen schema including no sub-schemata. Thus, two different polygon versions of the kitchen have been created: A high quality version (277224 polygons in total, 100% quality) which includes all objects rendered in high quality and the Kitchen Schema version (49135 polygons in total, 17.7% of the initial quality) which includes consistent objects rendered in low quality and inconsistent objects rendered in high quality. Object polygon quality variations dependent on user interaction do not exist since there are no sub-schemata.

Kitchen	Number of Polygons	Percentage
High Version	277224	17,70% of initial quality or 82.30% reduction
Kitchen Schema	49135	

Table 3: Kitchen area. Polygon statistics per activated schema.
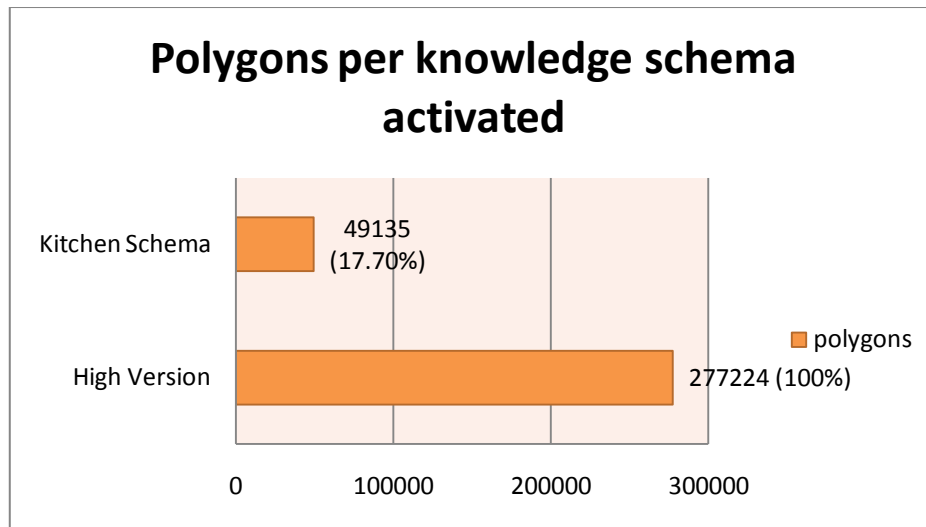
Figure 87: The high quality version of the kitchen consists of 277224 polygons while the selectively rendered kitchen consists of 49135 polygons.

The lounge area is represented by one main schema referred to as the Lounge Schema and the sofa area is represented by one sub-schema referred to as the Sofa Schema. Thus, 3 different versions of lounge area have been created: A high quality version (215558 polygons in total, 100% quality) which includes all objects rendered in high quality, Lounge Schema version (27729 polygons in total, 17.8% of the initial quality) which includes consistent objects rendered in low quality and inconsistent objects rendered in high quality and the Sofa Schema version (51699 polygons in total, 23.9% of the initial quality) which includes consistent objects rendered in low quality and inconsistent objects rendered in high quality.

Lounge	Number of Polygons	Percentage
High Version	215558	12,80% of initial quality or 87,20% reduction
Lounge Schema	27729	
High Version	215558	23,90% of initial quality or 76,10% reduction
Sofa Schema	51699	

Table 4: Lounge/Sofa area. Polygon statistics per activated schema.

The Sofa Schema Version consists of more polygons than the Lounge Schema Version. In the Lounge Schema Version there are objects such as a candle or a bottle of wine which are consistent in relation to the lounge schema and are rendered in low quality. In contrast, the same objects are inconsistent in relation to the sofa schema (a bottle of wine is inconsistent if it is placed on a couch according to the pilot questionnaires) and are rendered in high quality (more polygons).
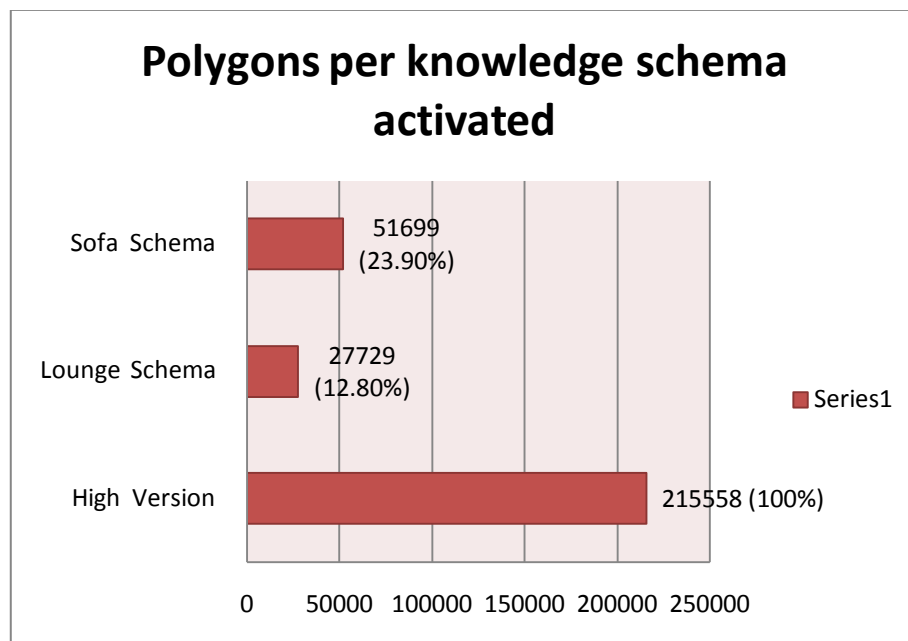
Figure 88: The lounge area and the sofa sub area. Polygons per activated schema.

The office area is more complex than the lounge and the kitchen area because it consists of one main schema, e.g the office schema and two sub-schemata, e.g. the desk schema and the bookcase schema.
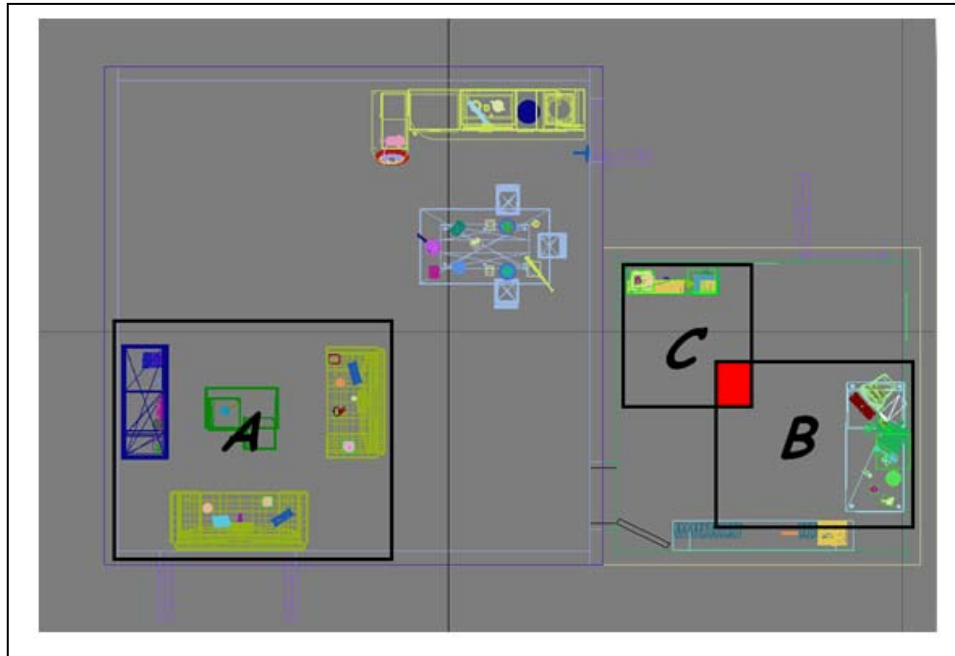
Figure 89: 2D top view screen shot of the scene. The office Area includes sub area B and sub area C.

As shown in Figure 89, the office area consists of two sub areas, e.g. what is shown as sub area B representing the desk sub area and sub area C representing the bookcase sub area. Therefore, there are four different routes that the user could navigate through as explained below:

- The user may navigate in areas that include neither sub area B nor sub area C. Therefore, only the main office schema could be activated.
- The user may navigate in areas that include sub area B but not sub area C. Therefore, the desk schema (area B) is activated, while for the rest of the room, the office main schema is activated.
- The user may navigate in areas that include sub area C but not sub area B. Therefore, the bookcase schema (area C) is activated, while for the rest of the room, the office main schema is activated.
- The user may navigate in areas that include both sub area B and sub area C shown as the red area in Figure 89. Therefore, the desk schema (area B) and the bookcase schema (area C) are activated.

Thus, different versions (Table 5) of the office area have been created.

- The high quality version consists of 291489 polygons including all objects rendered in high quality.

- The Office Schema version consists of 67003 polygons assuming that the office schema is activated. In this case, the consistent objects in relation to the office schema were rendered in low quality and the inconsistent objects in relation to the office schema were rendered in high quality.

- The Desk Schema & Bookcase Schema version consists of 163608 polygons assuming that both sub-schemata are activated (Figure 89). In this case, the consistent objects in relation to the sub-schemata were rendered in low quality and the inconsistent objects in relation to the sub-schemata were rendered in high quality. This version consists of the maximum number of polygons (excluding the high quality version) since both sub-schemata are activated.

- The Desk Schema & Office Schema version consists of 76968 polygons assuming that both the desk sub-schema and office schema are activated. In this case, the objects which are consistent in relation to the office schema and inconsistent in relation to the desk schema were rendered in high quality.

- The Bookcase Schema & Office Schema Version consists of 156963 polygons assuming that both the bookcase sub-schema and office schema are activated. In this case, the objects which are consistent in relation to the office schema and inconsistent in relation to the bookcase schema were rendered in high quality.

Office	Number of Polygons	Percentage
High Version	291489	23% of the initial quality or 77% reduction
Office Schema	67003	
High Version	291489	56,80% of the initial quality or 43,2% reduction
Desk Schema & Bookcase Schema	163608	
High Full Version	291489	26,00% of the initial quality or 74% reduction
Desk Schema & Office Schema	76968	
High Full Version	291489	54% of the initial quality or 46% reduction
Bookcase Schema & Office Schema	156963	

Table 5 : Office/Desk/Bookcase area. Polygon Statistics per activated schema.
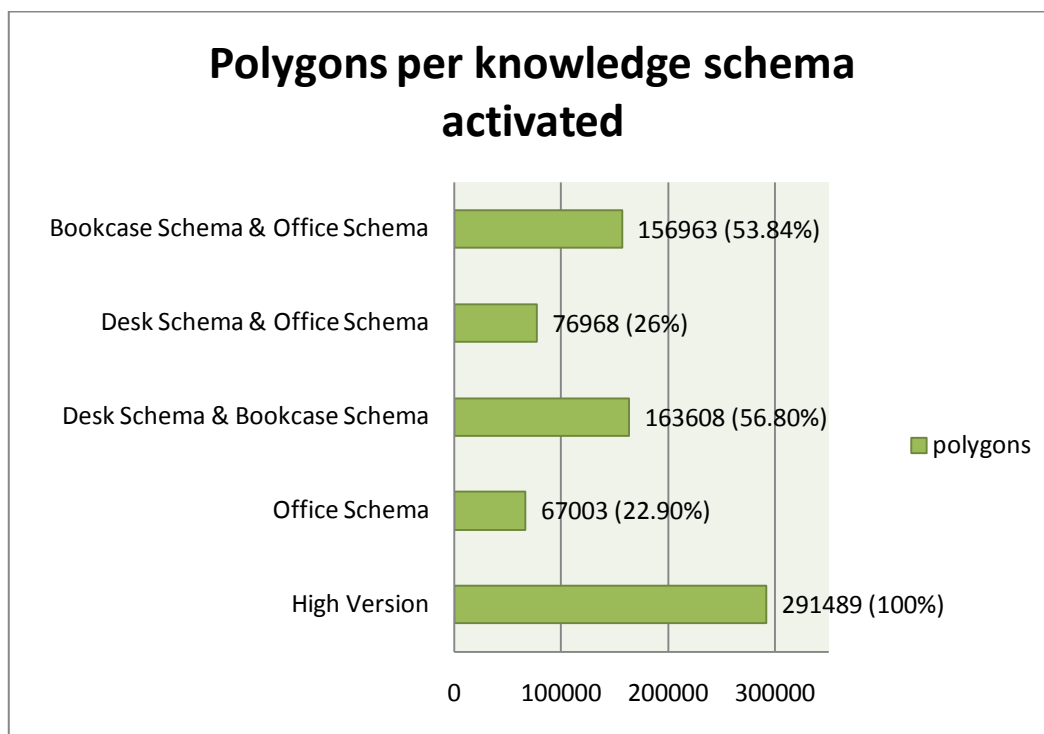
Figure 90: office area and the desk & bookcase sub areas. Polygons per activated schema..

6.3 Statistical Analysis

This section presents the basic statistical principles employed in order to analyze the acquired memory recognition self-report.

6.3.1 Signal detection Theory

The Signal detection theory (SDT) is used when psychologists want to measure the way we make decisions under conditions of uncertainty, such as how we would perceive distances in foggy conditions. SDT assumes that the decision maker is not a passive receiver of information, but an active decision-maker who makes difficult perceptual judgments under conditions of uncertainty. In foggy circumstances, we are forced to decide how far away from us an object is based solely upon visual stimulus which is impaired by the fog. Since the brightness of the object, such as a traffic light, is used by the brain to discriminate the distance of an object and the fog reduces the brightness of objects, we perceive the object to be much further away than it actually is.

To apply signal detection theory to a data set where stimuli were either present or absent and the observer categorized each trial as having the stimulus present or absent, the trials are sorted into one of four categories (Figure 91): "Miss" where the stimulus is present and the respond is 'absent' , "Hit" where the stimulus is present and the respond is 'present', "Correct Rejection" where the stimulus is absent and the respond is 'absent' and "False Alarm" where the stimulus is absent and the respond is 'present' .

	Respond "Absent"	Respond "Present"
Stimulus Present	Miss	Hit
Stimulus Absent	Correct Rejection	False Alarm

Figure 91: Signal detection theory data set.

Based on the proportions of these types of trials, numerical estimates of sensitivity can be obtained through statistics such as the sensitivity index d' and response bias β.

Sensitivity or discriminability

Conceptually, sensitivity refers to how hard or easy it is to detect that a target stimulus is present from background events. For example, in a recognition memory paradigm, having longer to study to-be-remembered words makes it easier to recognize previously seen or heard words. In contrast, having to remember 30 words rather than 5 makes the discrimination harder. One of the most commonly used statistics for computing sensitivity is the so-called sensitivity index, or d'.

Bias

Bias is the extent to which one response is more probable than another. That is, a receiver may be more likely to respond that a stimulus is present or more likely to respond that a stimulus is not present. Bias is independent of sensitivity. For example, if there is a penalty for either false alarms or misses, this may influence bias. If the stimulus is a bomber, then a miss (failing to detect the plane) may increase deaths, so a liberal bias is likely. In contrast, a crying wolf (a false alarm) too often may make people less likely to respond, grounds for a conservative bias.

6.3.2 Analysis of Variance

ANalysis Of VAriance: ANOVA procedures are powerful parametric methods for testing the significance of the differences between sample means where more than two conditions are used, or even when several independent variables are involved (Coolican 1999). ANOVA is used to compare the variance between the two groups with the variability within each of the groups. This comparison is in the form of a ratio known as the F test. A high value for F indicates a strong effect, i.e. the variance between groups is higher than the variance within the groups. The strength of the effect is given by the p value. The p value represents the probability that there is no between groups variance. This is called the **null hypothesis** and is disproved **if a value of p below 0.05 is returned**.

6.4 Data analysis overview

The experiment had two independent variables. The first between-subjects variable was 'polygon count' comprised of three levels (as described in Section 6.2):

- 'high poly'

- 'high-low poly'

- 'selective poly'

The second within-subjects independent variable was 'object type' comprised of two levels: 'consistent' and 'inconsistent'. The dependant variables were 'hit rate', i.e. rate of successful recognition of present objects and 'false alarm rate', i.e. rate of successful recognition of absent objects.

The hit and false alarm rates obtained were used to calculate the signal detection constituents d' (sensitivity) and β (bias), as outlined in by T. D. Wickens (2001), with d' being the distance between signal and noise distributions in participant responses and β being the bias towards replying 'yes' or 'no'. The d' is calculated by transforming hit and false alarm rates into z-scores, which give the position of a score in relation to the number of standard deviations it is above or below the mean. The d' score equals the hit ratio z-score minus the false alarm ratio z-score: d' = z(HIT)– z(FA). The β value is the ratio of the height of the Hit and False Alarm rate distribution curves: β = h(HIT)/h(FA). These values were calculated for each condition in the experiment and scores were compared using Analysis of Variance (ANOVA), using log β, as the β distribution is not statistically normal.

When participants were asked to recognize which items were present in the scene, they were also asked to rate how confident they were of their responses on a scale of 1 to 5. These confidence ratings were subsequently used to correct the recognition responses for possible guessing. This provided three sets of d' and β results for each area/sub area: one of uncorrected confidence; one of moderate confidence (i.e. corrected so that only recognition scores with corresponding confidence ratings of 3, 4 or 5 were used); and one set of high confidence (i.e. only confidence scores of 5 were used).

Statistical analysis using ANOVA for the d' and β scores were calculated. Note that the greater the d' score, the easier it was for participants to distinguish between signal and noise (i.e. recognize correctly), whilst the closer a β score is to 0, the less bias was found in

participant responses, with positive scores indicating a bias towards responding 'no' and negative scores a bias towards responding 'yes'.

6.5 Results and Discussion

Because of the large amount of data, in this section we will present the data analysis in relation to memory recognition while exposed to the kitchen area. Memory recognition for consistent and inconsistent items included in the kitchen area will be presented together in the category of d' and β unconfident (recognition scores with corresponding confidence ratings of 1, 2, 3, 4 or 5) rating results. Results related to remaining scene areas will be presented in summary.

Descriptive Statistics

	Group	Mean	Std. Deviation	N
d_cons_unconfidence	,00	,9707	,52560	18
	1,00	1,3333	,62797	18
	2,00	1,1318	,44838	18
	Total	1,1453	,54955	54
d_incons_unconfidence	,00	,9653	,64920	18
	1,00	1,0962	,50336	18
	2,00	1,1147	,56709	18
	Total	1,0587	,56931	54

Figure 92: Descriptive Statistics for d' (sensitivity) unconfidence case– kitchen area.

Descriptive statistics is a branch of statistics dealing with summarization and description of collections of data—data sets, including the concepts of arithmetic mean. For example, from descriptive statistics above, we can deduce that there are 18 participants for each condition (total 54). Also, we can be informed about the means and the standard deviation of the d' for the consistent and the inconsistent objects in the uncofidence rating category for each group.

Multivariate Tests[b]

Effect		Value	F	Hypothesis df	Error df	Sig.
d	Pillai's Trace	,010	,531[a]	1,000	51,000	,470
	Wilks' Lambda	,990	,531[a]	1,000	51,000	,470
	Hotelling's Trace	,010	,531[a]	1,000	51,000	,470
	Roy's Largest Root	,010	,531[a]	1,000	51,000	,470
d * Group	Pillai's Trace	,016	,403[a]	2,000	51,000	,671
	Wilks' Lambda	,984	,403[a]	2,000	51,000	,671
	Hotelling's Trace	,016	,403[a]	2,000	51,000	,671
	Roy's Largest Root	,016	,403[a]	2,000	51,000	,671

a. Exact statistic

b.

Design: Intercept+Group
Within Subjects Design: d

Figure 93: Multivariate Tests Table for d' (sensitivity) unconfidence case– kitchen area.

Between-subject variables (in this example, the Group variable) are independent variables or factors in which a different group of subjects is used for each level of the variable. Furthermore, a within-subjects variable (in this example, the sensitivity d') is an independent variable that is manipulated by testing each subject at each level of the variable. From multivariate tests results we can conclude whether there is a significant effect of viewing condition on a variable such as d' at a level of $p<0.05$ which signifies that d' consistent induces significantly different behavior from d' inconsistent. We also explore whether there is any significant interaction between the within subjects variable (d') and the between subject variable (Group). An interaction effect is a change in the simple main effect of one variable over levels of the second.

Here, there are no significant main effects of object type ($F(1,51)=0.531$, p=0.470>0.05) and no interaction ($F(2,51)=0.403$, p=0.671>0.05).

Tests of Between-Subjects Effects

Measure: MEASURE_1

Transformed Variable: Average

Source	Type III Sum of Squares	df	Mean Square	F	Sig.
Intercept	131,156	1	131,156	542,166	,000
Group	1,121	2	,560	2,317	,109
Error	12,337	51	,242		

Figure 94: Test of Between Subjects effects for d' (sensitivity) unconfidence case – kitchen area.

Between subjects effects predict whether there is a significant effect of viewing condition on memory recognition. Here, there is no main significant effect since $F(2,51)=0.560$, $p=0.109>0.05$. Thus regardless of viewing condition d' is similar, therefore, memory recognition performance is similar across viewing conditions.
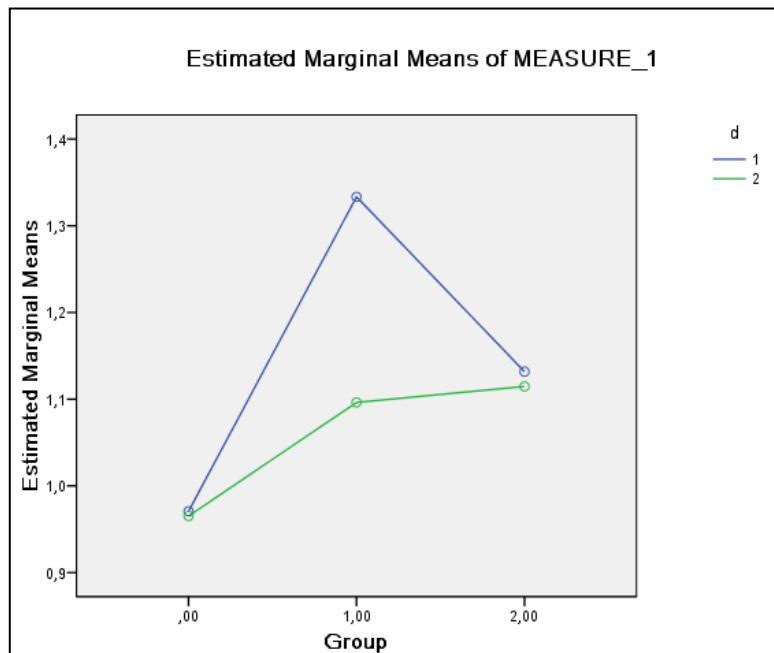
Figure 95: Estimated Marginal Plot for d' (sensitivity) unconfidence case – kitchen area.

Finally, in this plot we can notice that there are two lines. The blue line represents d' consistent and the green line represents d' inconsistent. This plot visualizes the results that have been described above. There is no interaction between group and object type and there are no significant effects of viewing condition and object type since the two lines don't have any significant distances or overlapping. Memory performance is similar across conditions. Moreover, participants' memory recognition performance was also similar when comparing memory performance related to the inconsistent as opposed to the consistent items.

Similarly, for β (bias):

Descriptive Statistics

	Group	Mean	Std. Deviation	N
b_cons_unconfidence	,00	-,2777	,58752	18
	1,00	,1014	,59260	18
	2,00	,1221	,63805	18
	Total	-,0181	,62317	54
b_incons_unconfidence	,00	,4795	,36068	18
	1,00	,5952	,34626	18
	2,00	,4513	,50206	18
	Total	,5087	,40618	54

Figure 96: Descriptive Statistics for β (bias) unconfidence case– kitchen area.

Multivariate Tests^b

Effect		Value	F	Hypothesis df	Error df	Sig.
b	Pillai's Trace	,360	28,743^a	1,000	51,000	,000
	Wilks' Lambda	,640	28,743^a	1,000	51,000	,000
	Hotelling's Trace	,564	28,743^a	1,000	51,000	,000
	Roy's Largest Root	,564	28,743^a	1,000	51,000	,000
b * Group	Pillai's Trace	,059	1,609^a	2,000	51,000	,210
	Wilks' Lambda	,941	1,609^a	2,000	51,000	,210
	Hotelling's Trace	,063	1,609^a	2,000	51,000	,210
	Roy's Largest Root	,063	1,609^a	2,000	51,000	,210

a. Exact statistic

b.

Design: Intercept+Group
Within Subjects Design: b

Figure 97: Multivariate Tests Table for β (bias) unconfidence case– kitchen area.

In this case, results reveal a significant effect for β. In detail, results reveal significant main effects of object type (F(1,51)=28.743, p<0.05) and no interaction (F(2,51)=1.609, p=0.210>0.05). In detail, the significant main effect of object type reveals that β inconsistent is significantly larger than β consistent (Figure 99). This effect indicates that the participants had larger bias for inconsistent than for consistent objects which indicates a bias towards responding 'no', therefore not recognizing inconsistent objects.

Tests of Between-Subjects Effects

Measure: MEASURE_1
Transformed Variable: Average

Source	Type III Sum of Squares	df	Mean Square	F	Sig.
Intercept	6,498	1	6,498	23,670	,000
Group	1,194	2	,597	2,174	,124
Error	14,001	51	,275		

Figure 98: Test of Between Subjects effects for β (bias) unconfidence case– kitchen area.

Moreover, there is no significant effect of viewing condition (F(2,51)=2.174, p=0.124>0.05).
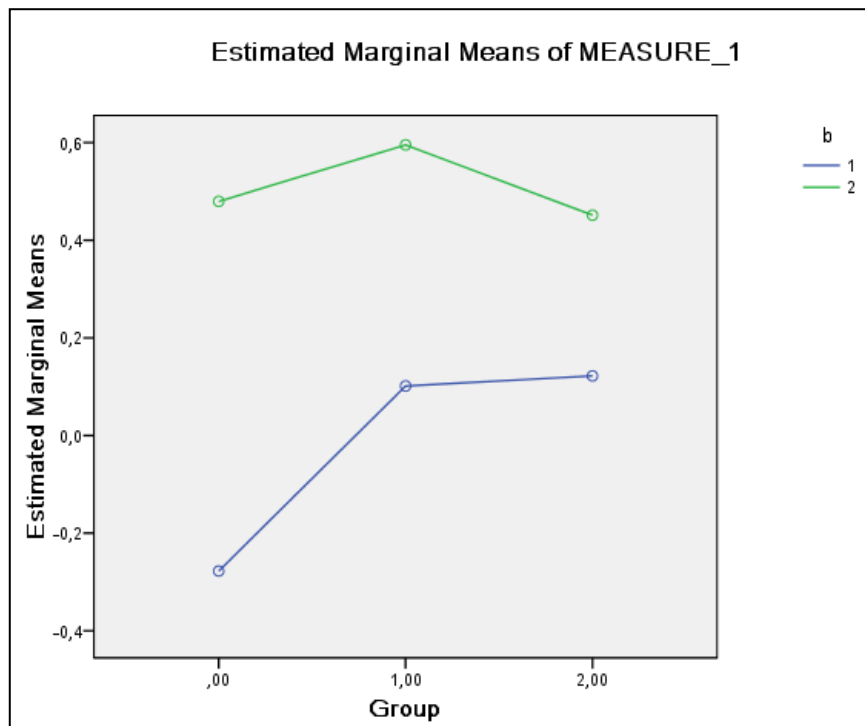
Figure 99: Estimated Marginal Plot for β (bias) unconfidence case– kitchen area.

Figure 99 shows that regardless viewing condition, although there is no significant effect of viewing condition and there is also no interaction, a significant effect of type of object is revealed in relation to the within subject variable (β consistent / β inconsistent), F(1,51)=28.743, **p<0.05**. In detail, β inconsistent (b2 – green line) is significantly larger than β consistent (b1–blue line). Therefore, in this case, regardless viewing condition, all participants had larger bias for inconsistent than for consistent objects which indicates a bias towards responding 'no', therefore not recognizing inconsistent objects.

Analysis for all areas is presented. The descriptive statistics for the d' and β scores were calculated. Six tables (one for each area) contain three sets of d' and β, e.g., uncorrected, moderate and high confidence (Tables 6-11). Note that the greater the d' score, the easier it was for participants to distinguish between signal and noise, i.e. recognize correctly, whilst the closer a β score is to 0, the less bias was found in participant responses, with positive scores indicating a bias towards responding 'no' and negative scores a bias towards responding 'yes'. In some areas when objects' quality changes in real-time such as for the office and lounge area, it is not possible to categorize objects as firmly consistent or inconsistent. The bookcase contained mainly inconsistent objects apart from books.

Each table shows d' and β scores and standard deviations (in brackets) as a functional of viewing condition (N=Total Number of Participants, O=Total number of objects, H=High Poly, HL=High Low Poly, S=Selective Poly, T=Total).

Table 6 shows the descriptive statistics for the d' and β scores of the kitchen area. Table 7, Table 8 and Table 9 shows the descriptive statistics for the d' and β scores of the office area, the desk sub area and the bookcase sub area. Table 10 and Table 11 show the descriptive statistics for the d' and β scores of the lounge area and the sofa sub area.

Kitchen Area

Consistent Objects

N	d'un	b'un	d'mod	b'mod	d'high	b'high
54	(O=12)	(O=12)	(O=12)	(O=12)	(O=12)	(O=12)
H	0.97(.52)	-.27(.58)	0.90(.42)	0.11(.54)	0.88(.50)	0.34(.52)
HL	1.33(.62)	0.10(.59)	1.03(.44)	0.23(.62)	0.95(.47)	0.42(.39)
S	1.13(.44)	0.12(.63)	0.97(.34)	0.26(.55)	0.82(.47)	0.40(.39)
T	1.14(.54)	-.01(.62)	0.97(.40)	0.20(.56)	0.89(.48)	0.40(.43)

Inconsistent Objects

N	d'un	b'un	d'mod	b'mod	d'high	b'high
54	(O=6)	(O=6)	(O=6)	(O=6)	(O=6)	(O=6)
H	0.96(.64)	0.47(.36)	0.96(.63)	0.52(.37)	0.97(.70)	0.59(.30)
HL	1.09(.50)	0.59(.34)	1.01(.51)	0.67(.27)	0.96(.53)	0.73(.27)
S	1.11(.56)	0.45(.50)	1.06(.63)	0.45(.46)	1.03(.65)	0.59(.25)
T	1.05(.56)	0.50(.40)	1.01(.58)	0.54(.38)	0.99(.62)	0.64(.28)

d' and β scores and standard deviations (in brackets) as a functional of viewing condition (N= Total Number of Participants, O=total number of objects, H=High Poly, HL=High Low Poly, S=Selective Poly, T=Total, un=uncorrected, mod=moderate)

Table 6: Kitchen area confidence results.

Office Area

N	d'un	b'un	d'mod	b'mod	d'high	b'high
54	(O=20)	(O=20)	(O=20)	(O=20)	(O=20)	(O=20)
H	0.86(.50)	0.20(.34)	0.91(.49)	0.35(.31)	0.97(.64)	0.70(.61)
HL	0,82(.33)	0.10(.27)	0.83(.28)	0.24(.22)	0.91(.37)	0.65(.37)
S	1.08(.46)	0.28(.57)	1.07(.50)	0.43(.51)	1.02(.54)	0.75(.57)
T	0.92(.44)	0.19(.41)	0,94(.44)	0.34(.37)	0.97(.52)	0.70(.52)

d' and β scores and standard deviations (in brackets) as a functional of viewing condition (N= Total Number of Participants, O=total number of objects, H=High Poly, HL=High Low Poly, S=Selective Poly, T=Total)

Table 7: Office area confidence results.

Desk Sub Area

Consistent Objects

N	d'un	b'un	d'mod	b'mod	d'high	b'high
54	(O=5)	(O=5)	(O=5)	(O=5)	(O=5)	(O=5)
H	0.70(.70)	0.08(.33)	0.61(.62)	0.20(.39)	0.48(.56)	0.27(.52)
HL	0.55(.75)	-.06(.34)	0.55(.50)	0.10(.29)	0.63(.52)	0.36(.39)
S	0.65(.76)	0.13(.43)	0.77(.56)	0.29(.43)	0.82(.52)	0.42(.39)
T	0.63(.72)	0.02(.41)	0.69(.56)	0.20(.38)	0.64(.54)	0.34(.43)

Inconsistent Objects

N	d'un	b'un	d'mod	b'mod	d'high	b'high
54	(O=4)	(O=4)	(O=4)	(O=4)	(O=4)	(O=4)
H	0.65(.69)	0.32(.30)	0.52(.69)	0.27(.30)	0.46(.69)	0.26(.36)
HL	0.43(.56)	0.26(.34)	0.34(.54)	0.20(.34)	0.29(.46)	0.20(.31)
S	0.65(.65)	0.28(.27)	0.59(.63)	0.31(.27)	0.36(.52)	0.21(.26)
T	0.58(.63)	0.30(.30)	0.48(.62)	0.26(.30)	0.37(.56)	0.23(.31)

d′ and β scores and standard deviations (in brackets) as a functional of viewing condition (N= Total Number of Participants, O=total number of objects, H=High Poly, HL=High Low Poly, S=Selective Poly, T=Total)

Table 8: Desk sub area confidence results.

Bookcase Area

N 54	d'un (O=7)	b'un (O=7)	d'mod (O=7)	b'mod (O=7)	d'high (O=7)	b'high (O=7)
H	0.27(.72)	0.10(.41)	0.39(.70)	0.17(.49)	0.33(.65)	0.15(.44)
HL	0.61(.61)	0.23(.27)	0.59(.56)	0.34(.34)	0.52(.53)	0.36(.33)
S	0.50(.54)	0.17(.45)	0.65(.58)	0.28(.52)	0.58(.63)	0.26(.49)
T	0.46(.63)	0.16(.38)	0.54(.62)	0.26(.45)	0.48(.60)	0.26(.43)

d′ and β scores and standard deviations (in brackets) as a functional of viewing condition (N= Total Number of Participants, O=total number of objects, H=High Poly, HL=High Low Poly, S=Selective Poly, T=Total)

Table 9: Bookcase sub area confidence results.

Lounge Area

N 54	d' un (O=17)	b' un (O=17)	d' mod (O=17)	b' mod (O=17)	d' high (O=17)	b' high (O=17)
H	0.57(.62)	0.18(.32)	0.58(.69)	0.30(.54)	0.67(.58)	0.49(.47)
HL	0.40(.43)	0.03(.21)	0.46(.54)	0.12(.32)	0.65(.46)	0.51(.48)
S	0.75(.73)	0.26(.49)	0.70(.72)	0.31(.58)	0.82(.75)	0.62(.59)
T	0.57(.57)	0.15(.37)	0.58(.65)	0.25(.49)	0.71(.60)	0.54(.51)

d' and β scores and standard deviations (in brackets) as a functional of viewing condition (N= Total Number of Participants, O=total number of objects, H=High Poly, HL=High Low Poly, S=Selective Poly, T=Total)

Table 10: Lounge area confidence results.

Sofa Sub Area

Consistent Objects

N 54	d' un (O=4)	b' un (O=4)	d' mod (O=4)	b' mod (O=4)	d' high (O=4)	b' high (O=4)
H	0.84(.95)	0.01(.35)	0.72(.93)	0.06(.32)	0.69(.86)	0.08(.19)
HL	1.24(.83)	-.01(.31)	0.98(.79)	0.14(.28)	0.69(.62)	0.26(.29)
S	1.30(.82)	0.13(.36)	1.07(.82)	0.15(.35)	1.07(.76)	0.25(.32)
T	1.13(.88)	0.04(.34)	0.93(.85)	0.12(.31)	0.81(.76)	0.20(.28)

Inconsistent Objects

N 54	d' un (O=4)	b' un (O=4)	d' mod (O=4)	b' mod (O=4)	d' high (O=4)	b' high (O=4)
H	0.37(.82)	0.07(.31)	0.44(.75)	0.10(.31)	0.25(.40)	0.12(.25)
HL	0.43(.36)	0.07(.39)	0.42(.31)	0.12(.32)	0.39(.36)	0.03(.28)
S	0.73(.56)	0.16(.35)	0.67(.58)	0.22(.20)	0.37(.49)	0.20(.25)
T	0.51(.62)	0.10(.35)	0.51(.58)	0.15(.28)	0.33(.42)	0.12(.27)

d' and β scores and standard deviations (in brackets) as a functional of viewing condition (N= Total Number of Participants, O=total number of objects, H=High Poly, HL=High Low Poly, S=Selective Poly, T=Total)

Table 11: Sofa sub area confidence results.

ANOVA analysis of the d' and β results revealed no main effect of viewing condition in <u>all cases</u> (e.g. office d' score moderate confidence analysis: $F_{(2,51)}=1.370$, p=0,263>0.05). This indicates that memory performance was similar across conditions.

Nevertheless, significant main effects of object type (consistent vs inconsistent) were revealed. When participants were asked to recognize which items were present in the scene, they were also asked to rate how confident they were of their responses on a scale of 1 to 5. These confidence ratings were subsequently used to correct the recognition responses for possible guessing. This provided three sets of d' and β results for each area/sub area: one of uncorrected confidence; one of moderate confidence (i.e. corrected so that only recognition scores with corresponding confidence ratings of 3, 4 or 5 were used); and one set of high confidence (i.e. only confidence scores of 5 were used).

In relation to β scores, the following significant main effects of object type appeared. Results revealed significant main effects of object type for the kitchen area (Table 6) concerning uncorrected confidence ($F_{(1,51)}=28.743$, p<0.05), moderate confidence ($F_{(1,51)}=14.158$, p<0.05) and high confidence ($F_{(1,51)}=12.278$, p<0.05). Moreover, results revealed a significant main effect of object type for the desk area (Table 8), concerning moderate confidence ($F_{(1,51)}=16.812$, p<0.05).

- Kitchen area (Table 6), uncorrected confidence, $F_{(1,51)}=28.743$, p<0.05

- Kitchen area (Table 6), moderate confidence, $F_{(1,51)}=14.158$, p<0.05 (Figure 100)

- Kitchen area (Table 6), high confidence, $F_{(1,51)}=12.278$, p<0.05

- Desk area (Table 8), moderate confidence, $F_{(1,51)}=16.812$, p<0.05

These results revealed a significant main effect of object type (β inconsistent > β consistent) but no main effect of viewing condition, nor a significant interaction. This indicates that participants, regardless of viewing condition were significantly more biased to responding that inconsistent objects were not present in the scene compared to recognition of consistent objects for which there was close to no bias at all. This is consistent with memory psychology literature (Rojahn & Pettigrew 1992) and indicates that such effects could be simulated when immersed in synthetic scenes.

Concerning d' scores, the following significant main effects of object type appeared. Results revealed significant main effects of object type for sofa area (Table 11) about uncorrected

confidence (F(1,51)=16.225, p<0.05), moderate confidence (F(1,51)=7.526, p<0.05) and high confidence (F(1,51)=15.481, p<0.05). Moreover, results revealed a significant main effect of object type for the desk area (Table 8) about high confidence (F(1,51)=7.893, p<0.05).

- Sofa area (Table 11), uncorrected confidence, F(1,51)=16.225, p<0.05

- Sofa area (Table 11), moderate confidence, F(1,51)=7.526, p<0.05

- Sofa area (Table 11), high confidence, F(1,51)=15.481, p<0.05

- Desk sub-area (Table 8), high confidence, F(1,51)=7.893, p<0.05

These results revealed a significant main effect of object type (d' consistent > d' inconsistent) but no main effect of viewing condition, nor a significant interaction. The consistent objects regardless of viewing condition (high poly, high – low poly, selective poly), have a significantly greater distance between signal and noise distribution (thus better recognition) than the inconsistent objects.
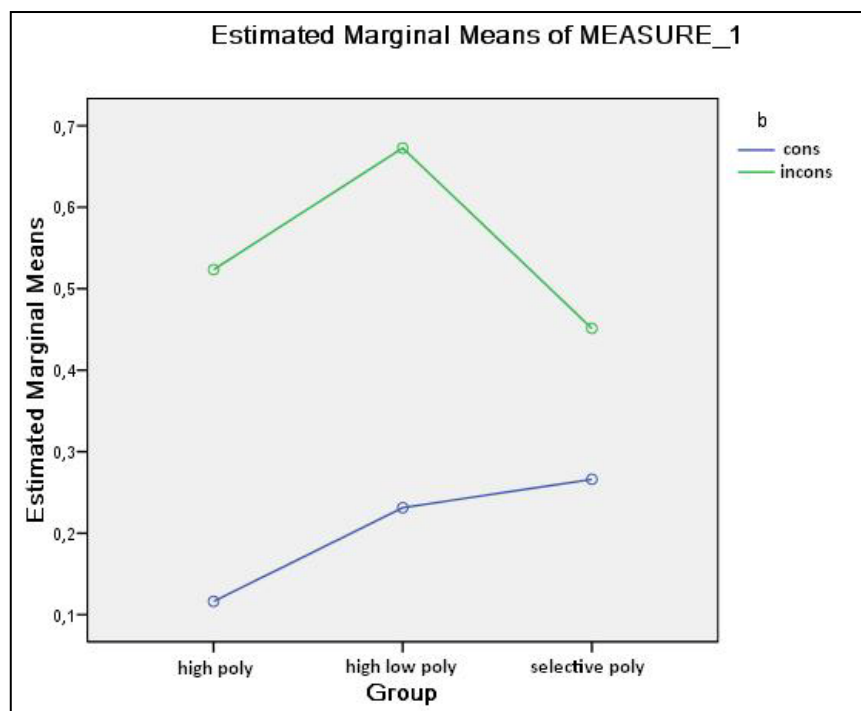
Figure 100: ANOVA Estimated Marginal Means Plot. Kitchen area - moderate confidence case. b inconsistent score is significantly greater than b consistent score regardless of viewing condition (high poly, high low poly, selective poly).

Figure 100 reveals a significant main effect of object type (β inconsistent > β consistent) but no main effect of viewing condition, nor a significant interaction since β inconsistent is

significantly greater than β consistent in all groups and there is no overlapping between two lines. This indicates that participants, regardless of viewing condition were significantly more biased to responding that inconsistent objects were not present in the scene compared to recognition of consistent objects for which there was close to no bias at all.

6.6 Chapter Summary

This chapter presented an evaluation study of the real-time selectively rendered scene by comparing it to a high quality rendering solution. The task utilized for this evaluation is memory recognition based on schemata, drawn from formalized methodologies of memory psychology.

In order to evaluate spatial awareness while exposed to a selectively rendered scene based on memory schema in relation to a fully-fledged rendering solution, a scene was designed which included several areas and sub-areas and a memory recognition task was completed. The scene was designed so that it contains objects with differing levels of scene consistency based on the area and sub-area they were placed. The polygon quality of such objects could, therefore, change dynamically.

Furthermore, the results of the research project have been presented and discussed in this chapter. We predicted that the polygon count savings of the selectively rendered system will not diminish spatial awareness while exposed to the immersive simulation implemented.

The following chapter draws conclusions from the findings, discusses limitations of the current research and makes recommendations for the future.

Chapter 7

Conclusions and Future Work

In this thesis, we proposed a novel selective rendering approach which is *task* and *gaze* **independent**, simulating cognitive creation of spatial hypotheses. Scene objects are rendered in varying polygon quality according to how they are associated with the context (schema) of the scene. Experimental studies in synthetic scenes have revealed that consistent objects which are expected to be found in a scene can be rendered in lower quality without affecting information uptake. Exploiting such expectations, inconsistent items which are salient require a high level of rendering detail in order for them to be perceptually acknowledged.

In order to evaluate spatial awareness while exposed to a selectively rendered scene based on memory schema in relation to a fully-fledged rendering solution, a scene was designed which included several areas and sub-areas and a memory recognition task was completed. The scene was designed so that it contains objects with differing levels of scene consistency based on the area and sub-area they were placed. The polygon quality of such objects could, therefore, change dynamically.

Participants' memory performance was assessed after exposure to scenes of varying rendering quality through recognition of present objects in a scene as well as inferred objects. It has been shown that a scene which incorporates objects of high and low quality provokes similar memory recognition performance and associated confidence levels when compared to a fully-fledged rendering solution. Thus, in relation to the recognition task utilized, the selective rendering system presented is of similar functional fidelity but of significant less computational complexity than a scene including objects of a high polygon count.

7.1 Main Contributions

Cognitive psychology is a discipline within psychology that investigates the internal mental processes of thought such as visual processing, memory, problem solving, and language. Computer science or computing science (CS) is the study of the theoretical foundations of information and computation, and of practical techniques for their implementation and application in computer systems. The core focus of cognitive psychology is on how people acquire, process and store information. There are numerous practical applications for cognitive research, such as ways to improve memory or how to increase decision-making accuracy. The term cognitive in cognitive science is used for any kind of mental operation or structure that can be studied in precise terms" (Lakoff and Johnson, 1999). In this thesis, we showcased that we could optimize fundamental processes in computer graphics through cognitive psychology principles.

A central tenet of cognitive science is that a complete understanding of the mind/brain cannot be attained by studying only a single level. For example, consider the problem of remembering a phone number and recalling it later. How does this process occur? One approach would be to study behavior through direct observation. A person could be presented with a phone number, asked to recall it after some delay. Then the accuracy of the response could be measured. Another approach would be to study the firings of individual neurons while a person is trying to remember the phone number. Neither of these experiments on their own would fully explain *how* the process of remembering a phone number works. Even if the technology to map out every neuron in the brain in real-time were available, and it were known when each neuron was firing, it would still be impossible to know how a particular firing of neurons translates into the observed behavior. Thus an understanding of how these two levels relate to each other is needed. This can be provided by a functional level account of the process. Studying a particular phenomenon from multiple levels creates a better understanding of the processes that occur in the brain to give rise to a particular behavior.

One of the major applications areas for VEs is simulation and training for implementations that are meant for skill transfer in the real world. Since spatial perception is closely linked with memory, there is a body of research that has explored issues of transfer of training, employing spatial memory tasks under distinct technological conditions. Spatial memory tasks are considered quite significant in that sense, since in order to accomplish any task using a VE implementation, one should realize the structure of a particular space first. Thus,

the utility of VEs for any applications for which they are being proposed is predicated upon the accuracy of the spatial representation formed in the VE. In that sense, spatial memory tasks are considered essential and they are often employed in usability studies and benchmarking platforms that assess VE interface task efficiency.

The ultimate goal of VE technologies for simulation and training is to induce a spatial sense similar to the sense of three dimensional space humans get in the real world. This is essential for applications that often target successful transfer of training in the real world and consequently require a high level of simulation fidelity. Thus, spatial perception issues are clearly involved in interactive computer graphics rendering and should be considered at various levels while assembling a VE implementation.

This thesis introduced a novel selective rendering approach simulating cognitive creation of spatial hypotheses. Using classic findings from memory research in which schemata are used to explain memory processes, this thesis examined whether computationally more expensive scenes containing higher polygon counts and thus greater visual fidelity, add to the functional realism of an immersive virtual environment displayed on a stereo capable, head tracked Head Mounted Display, or whether they are they merely more aesthetically pleasing than their lower polygon count counterparts.

An object-based rendering engine was developed which renders objects in a scene according to their association with the schema in context, incorporating experimental results. Objects which are consistent with a given schema can be recalled better because of this association with the schema in context, therefore, they don't require a high amount of rendering computation.

The contribution of this thesis is an innovative x3D-based selective rendering framework based on memory schemata and implemented through metadata enrichment. We exploit the X3D's capability of defining metadata information in order to describe the level of consistency that every object entails in relation to the specific context of a scene's region.

7.2 Implications for Future Work

The experiments of Chapter 6 were formally designed. However, certain improvements could be accomplished by the following actions:

- It would be useful and innovative if we could automatically identify objects' level of consistency by analyzing gaze direction and fixation. For cognitive psychology, it is challenging to produce reliable predictive models of attention based on meaning rather than image properties such as edges or contrast.

- Quality variations, besides polygon count quality, could include global illumination calculations, shadow accuracy and texture resolution. Therefore, rendering quality could vary depending on the integration of different rendering computation parameters or a combination, and therefore, selective rendering may be more efficient.

- Levels of object consistency could be more than the two utilized. Certainly, an object in a scene of a specific context may be associated with a more complex 'consistency' rating, even in probabilistic terms, rather than the simplified assumption of being either consistent or inconsistent. The relationship between the meaning of scenes and attention levels are open research issues in cognitive psychology. We could define more levels of consistency and, therefore, additional levels of rendering quality resulting in a more complex schema model.

- Finally, we could focus on whether the proposed metric is sensitive to object quality assignment variations. It would be very interesting if we could add a viewing condition which doesn't take schema theory into account, e.g. half of the objects are randomly rendered as high-polygon and half as low-polygon objects. By this way, it is possible to prove the functional equivalence without considering only the null effects since we could use another metric which doesn't take into account the schema theory.

Human perception now determines to a large extent what we do in computer graphics and indeed, why we do it. Exploiting schema models of visual perception is one way to improve the viewing experience within a limited budget of computational resources.

References/Bibliography

Akeine-Moller and Haines 2002.   Akeine-Moller, T., and Haines, E. 2002. Real-time Rendering. Second Edition. A.K. Peters Ltd.

ANST 2003.   American National Standard for Telecommunications [www], http://www.atis.org/tg2k/, (Accessed September 2003)

Ashdown 2004.   Ian Ashdown, *Radiosity Bibliography* [www], http://tralvex.com/ pub/rover/abs-ian0.htm (Accessed March 2004)

Birn 2000   Birn, J. 2000. *[digital] Lighting and Rendering.* New Riders.

Bolin et all 1998   Bolin, M.R., Meyer, G.W. A (1998). Perceptually based Adaptive Sampling Algorithm. *Proc. of ACM SIGGRAPH 1998*, 299-310.

Bos 2001   Bos, Bert and The W3C Communications Team. "XML in 10 points," World Wide Web Consortium (W3C), Cambridge, MA, revised June 2003. Available at www.w3.org/XML/1999/XML-in-10-points

Bouknight 1970   Bouknight J. 1970. "A procedure for generation of three dimensional half-toned computer graphics presentations." *Communications of the ACM*, Vol. 13 (9) 527—536

Brewer & Treyens 1981.   Brewer W.F. and Treyens J.C. (1981). *Role of Schemata in Memory for places*

Bruce et al. 1996.   Bruce, V., Green, P. R., Georgeson, M.A. (1996). *Visual Perception: Physiology, Psychology and Ecology*. Psychology Press

Cater et all 2003.   K. Cater, A. Chalmers, A., and G. Ward, "Detail to attention: Exploiting visual tasks for selective rendering," in *Eurographics Workshop on Rendering,* 2003, pp. 270–280.

Challis et al. 1996,   Challis, B.H. Velichkovsky, B.M. and Craik, F.I.M. 1996. "Levels-of-processing effects on a variety of memory tasks: New findings and theoretical implications." *Consciousness and Cognition*, 5 (1).

Chalmers and Cater 2002.   Chalmers A.G, and Cater K. 2002 "Realistic Rendering in Real-Time." In proceedings of the 8th International Euro- Par Conference on Parallel Processing, Springer-Verlag 21-2

Chalmers et al. 2000.   Chalmers A.G., McNamara A., Troscianko T., Daly S. and Myszkowski K. 2000. "Image Quality Metrics", *ACM SIGGRAPH 2000*, course notes

Chalmers, McNamara et all 2000   Chalmers, A., McNamara, A., Troscianko, T., Daly, S., Myszkowski, K. (2000). *Image Quality Metrics*, ACM SIGGRAPH 2000, course notes

Cohen and Greenberg 1985   Cohen M.F., and Greenberg D.P. 1985. "The Hemi-Cube: A radiosity solution for complex environments." In B. A. Barsky, editor, *Computer Graphics (SIGGRAPH '85 Proceedings)*, volume 19, pages 31-40.

Cohen et al. 1988	Cohen M.F., Chen S.E., Wallace J.R., and Greenberg D.P. 1988. "A progressive refinement approach to fast radiosity image generation." In John Dill, editor, *Computer Graphics (SIGGRAPH 1988 Proceedings)*, volume 22, pages 75—84.
Coolican 1999	Coolican, H. 1999. *Research Methods and Statistics in Psychology*, Hodder & Stoughton Educational, U.K
Daly,S. 1993.	Daly, S. (1993). The Visible Differences Predictor: An algorithm for the assessment of image fidelity. In A. Watson, editor, *Digital Image and Human Vision*, 179-206. Cambridge, MA, MIT Press.
Dingliana 2004.	Dingliana, J., Image Synthesis Group, Trinity College, Dublin *(Presentation Slides)* [www], http://isg.cs.tcd.ie/dingliaj/ 3d4ii/Light1.ppt (Accessed March 2004)
Ferwerda 2001	FERWERDA, J. 2001. Hi-Fi Rendering. Proc. of the ACM Siggraph campfire in Perceptually adaptive graphics [Online]. Available from: http://isg.cs.tcd.ie/campfire/jimferwerda2.html [Accessed 15 Aug 2005]
Ferwerda 2003	Ferwerda, J.A. 2003. "Three varieties of realism in computer graphics." In IS&T/SPIE Conference on Human Vision and Electronic Imaging VIII, SPIE Proceedings Vol. 5007, 290-297.
Flannery and Walles 2003.	FLANNERY, K.A., WALLES, R. 2003. How does schema theory apply to real versus virtual memories? Cyberspychology and Behavior, 6(2), 151-159.
Flavios 2004.	Flavios, *Light and optic theory and principles* [www], http://homepages.tig.com.au/~flavios/diffrac.htm (Accessed March 2004)
Gibson and Hubbold 1997.	Gibson, S., and Hubbold R.J. 1997. "Perceptually Driven Radiosity." *Computer Graphics Forum*, 16 (2): 129-140.
Glassner 1984	Glassner, A.S. 1984. "Space Subdivision for Fast Ray Tracing", *IEEE Computer Graphics & Applications*, Vol.4, No. 10, pp 15-22.
Glassner 1989.	Glassner, A.S. 1989. *An Introduction to Ray tracing.* Morgan Kaufmann.
Goral et al. 1984.	Goral C.M., Torrance K.E., Greenberg D.P., and Battaile B. 1984. "Modeling the interaction of light between diffuse surfaces." In proceedings of the *11th Annual Conference on Computer Graphics and Interactive Techniques*. Vol. 18 (3) 212-222
Gouraud 1971	Gouraud H., "Continuous shading of curved surfaces", *IEEE Transactions on Computers,* 20(6): 623-628.
Hall and Greenberg 1983	Hall R.A., and Greenberg D.P. 1983. "A Testbed for Realistic Image Synthesis." *IEEE Computer Graphics and Applications*, Vol. 3, No. 8. pp. 10-19.
Heckbert 1989	Heckbert P.S. 1989. *Fundamentals of Texture Mapping and Image Warping.* Master's thesis, University of California, Berkeley
Itti and Koch 2001.	Itti, L., and Koch, C. 2001 "Computational modeling of visual attention", In *Nature Reviews Neuroscience*, Vol. 2(3), 194-203.

Itti et al. 1998	Itti, L., Koch, C., and Niebur, E. 1998 "A Model of Saliency-Based Visual Attention for Rapid Scene Analysis", *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) 20*, 11, 1254-1259.
Kajiya 1986:	Kajiya, J.T. 1986. "The Rendering Equation." *ACM SIGGRAPH 1986 Conference Proceedings,* volume 20,143-150.
Kajiya 1990:	Kajiya, J.T. (1990). Radiometry and Photometry for Computer Graphics.*Proc. of ACM SIGGRAPH 1990 Advanced Topics in Ray Tracing course notes.*
Katedros 2004.	*Illumination: simulation and perception* (course), http://www.maf.vu.lt/katedros/cs2/lietuva/courses/spalvos/ illumination5.pdf (Accessed February 2004)
Kirk 2003	Kirk, D. 2003. "Graphics Architectures: The Dawn of Cinematic Computing." In proceedings of GRAPHITE 2003, ACM, 9.
Koch and Ullman 1985	Koch C., and Ullman S. 1985. "Shifts in selective visual attention: towards the underlying neural circuitry." *Human Neurobiology*, 219--227.
Krivánek et al. 2003	Krivánek J., Zara J., and Bouatouch K. 2003. "Fast Depth of Field Rendering with Surface Splatting." *Computer Graphics International 2003*: 196-201.
Kuipers 1975	KUIPERS, B.J. 1975. A frame for frames: Representing knowledge for recognition. In D.G. Bobrow & A. Collins (Eds.), Representation and Understanding: Studies in Cognitive Science. New York: Academic Press, 1975.
Lampinien et al 2001	LAMPINEN, J., COPELAND, S., NEUSCHATZ, J. 2001. Recollections of things schematic: rooms schemas revisited. Cognition, 27, 1211-1222.
Langbein 2004.	Langbein, F.C., *Advanced Rendering - Radiosity*, Cardiff University, (course notes) http://cyl.cs.cf.ac.uk/teaching/graphics/G-20-V_2.pdf (Accessed February 2004)
Languénou et al. 1992.	Languénou E., Bouatouch K., and Tellier P. 1992. "An adaptive Discretization Method for Radiosity." *Computer Graphics Forum* 11(3): 205-216.
Lischinski 2003.	Lischinski, D., Radiosity, (Lecture notes) http://www.cs.huji.ac.il/~danix/advanced/notes3.pdf (Accessed December 2003)
Lischinski et al 1994.	Lischinski, D., Smits, B., Greenberg, D.P. (1994). Bounds and Error Estimates for Radiosity. In A. Glassner, editor, *Proc. of ACM SIGGRAPH 1994*, 67- 74.
Lubin, J. 1995.	Lubin, J. (1995). A Visual Discrimination Model for Imaging System Design and Development. In Peli E. (ed.), *Vision Models for target detection and recognition*, 245-283. World Scientific
Luebke 2001	LUEBKE, D., P., 2001 A Developer's Survey of Polygonal Simplification Algorithms IEEE Computer Graphics and Applications May/June 2001 (Vol. 21. No 3) pp. 24-35

Luebke and Hallen 2001	Luebke D. and Hallen B. 2001 "Perceptually driven simplification for interactive rendering", *12th Eurographics Workshop on Rendering*, 221-223.
Luebke et al. 2000	Luebke D., Reddy M., Watson B., Cohen J. and Varshney A. 2001. "Advanced Issues in Level of Detail." *Course #41 at ACM SIGGRAPH 2000*. Los Angeles, CA. August 12-17.
Maciel and Shirley 1995.	Maciel P.W.C. and Shirley P. 1995 "Visual Navigation of Large Environments Using Textured Clusters", *Symposium on Interactive 3D Graphics*, 95-102.
Mack and Rock 1998.	Mack, A. and Rock, I. 1998. *Inattentional Blindness*. Massachusetts Institute of Technology Press.
Mania 2001	Mania, A. (2001). *Fidelity metrics for virtual environment simulations based on human judgements of spatial awareness states*
Mania K. & Robinson, A., Brandt, K. 2005	The Effect of Memory Schemas on Object Recognition in Virtual Environments. Presence Teleoperators and Virtual Environments, MIT Press
Mania, K., & Chalmers, A., 2001	The Effects of Levels of Immersion on Memory and Presence in Virtual Environments: A Reality Centered Approach. Cyberpsychology & Behavior 4(2), 247-264
Mourkoussis, **Mania** et all (accepted 2009)	Quantifying Fidelity for Virtual Environment simulations employing memory schema assumptions. *ACM Transactions on Applied Perception.*
Mania, et all. 2003	Fidelity Metrics for Virtual Environment Simulations based on Human Judgments of Spatial Memory Awareness States. Presence, Teleoperators and Virtual Environments, 12(3), 296-310. MIT Press
Mania, Mourkoussis, Zotos 2008	Selective Rendering based on Perceptual Importance of Scene Regions IEEE conference on Systems, Man and Cybernetics, Singapore 2008
Marmitt and Duchowski 2002	Marmitt G., and Duchowski A.T. 2002. "Modeling Visual Attention in VR: Measuring the Accuracy of Predicted Scanpaths." *Eurographics 2002.* Short Presentations, 217-226.
Marvie et al. 2003	Marvie J-E., Perret J., and Bouatouch K. 2003. "Remote Interactive Walkthrough of City Models." *Pacific Conference on Computer Graphics and Applications*, 389- 393.
McConkie and Loschky 1997;	McConkie, G.W. and Loschky, L.C. 1997 "Human Performance with a Gaze-Linked Multi-Resolutional Display". *ARL Federated Laboratory Advanced Displays and Interactive Displays Consortium*, Advanced Displays and Interactive Displays First Annual Symposium, Adelphi, MD. January 28-29, (Pt. 1)25-34
McConkie and Loschky 2000;	McConkie, G.W. and Loschky, L.C. 2000. "Attending to Objects in a complex Display". Proceedings of *ARL Federated Laboratory Advanced Displays and Interactive Displays Consortium Fourth Annual Symposium*, 21-25.
McNamara 2000	McNamara, Ann, *Comparing Real and Synthetic Scenes using Human Judgements of Lightness*, PhD Thesis, Bristol, October 2000.

McNamara 2000.   McNamara, A. *Comparing Real and Synthetic Scenes using Human Judgements of Lightness*. Ph.D thesis, Dept. of Computer Science, University of Bristol, UK.

ME 2004.   Molecular Expressions, *Physics of light And colour* [www],http://micro.magnet.fsu.edu/primer/java/reflection/specular/ (Accessed February 2004)

Meyer and Liu 1992   Meyer, G.W. & Liu, A. (1992). Color Spatial Acuity Control of a Screen Subdivision Image Synthesis Algorithm. *Human Vision, Visual Processing and Digital Display,* 1666(3), 387-399.

Minsky 1975   M. MINSKY 1975, A Framework for Representing Knowledge. The Psychology of Computer Vision, P. H. Winston (ed.), McGraw-Hill 1975

Myszkowski 1998   Myszkowski, M. The Visible Differences Predictor: Applications to Global Illumination Problems. (1998). In G. Drettakis and N. Max, editors, *Proc. of the Eurographics Rendering Workshop 1998*, 233-236. NY, Springer Wien.

Nusselt 1928   Nusselt, W. 1928. "Grapische Bestimmung des Winkelverhaltnisses bei der Warmestrahlung," *Zeitschrift des Vereines Deutscher Ingenieure* 72(20):673.

Okino 2001.   OKINO COMPUTER GRAPHICS 2001 Overview of the Okino Polygon Reduction Algorithm [online]. Available from: http://www.okino.com/conv/polygon_reduction/geoman2/polygon_reduction2.htm [Accessed 12 Aug 2005

Osterberg 1935.   Osterberg, G. 1935. "Topography of the layer of rods and cones in the human retina." *Acta Ophthalmologica,* 6, 1-102.

Palmer 1999.   Palmer, S.E. 1999. Vision Science – Photons to Phenomenology. Massachusetts Institute of Technology Press

Phong 1975   Phong, B.T. 1975. "Illumination for Computer-Generated Images." *Communications of the ACM*, Vol 18 (6) 449— 455.

Privitera and Stark 2000;   Privitera, C.M. and Stark, L.W. 2000. "Algorithms for defining visual regions-of-interest: Comparison with eye fixations." *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 22, 9, 970-982

Rojahn and Pettigrew 1992   ROJAHN, K. PETTIGREW, T. 1992. Memory for schema relevant information: a meta-analytic resolution. British Journal of Social Psychology 31, 81-109.

Rushmeier et al 1995.   Rushmeier, H., Ward, G., Piatco, C., Sanders, P., Rust, B. (1995). Comparing Real and Synthetic Images: Some ideas about metrics. *Proc. of EGRW 1995*.

Siegel and Howell 1992   Siegel R., and Howell J.R. 1992. *Thermal Radiation Heat Transfer*, 3rd Edition. Hemisphere Publishing Corporation, New York, NY.

Slater 1999   Slater, M. (1999). Measuring Presence: A Response to the Witmer and Singer Questionnaire. *Presence: Teleoperators and Virtual Environments,* 8(5), 560-565. MIT Press.

Stanley et all 1998.	Stanney, K.M et al (1998). Report sponsored by the life sciences division at NASA headquarters. *International Journal of Human-Computer Interaction.*
T. D. Wickens, 2001,	T. D. Wickens, *Elementary Signal Detection Theory.* New York: Oxford University Press, 2001
Tumblin & Rushmeier 1993	Tumblin, J. & Rushemeier, H.E. (1993). Tone Reproduction for Realistic Images*. IEEE Computer Graphics and Applications*, 13(6), 42-48.
Waller et al 1998.	Waller, D., Hunt, E., Knapp, D. (1998). The Transfer of Spatial Knowledge in Virtual Environment Training. *Presence: Teleoperators and Virtual Environments,* 7(2), MIT Press.
Ward Larson and Shakespeare 1997.	Ward Larson, G and Shakespeare, R. 1998. "Rendering with RADIANCE: The art and science of lighting simulation", San Francisco: Morgan Kauffman
Watson et al. 1997a;	Watson, B., Friedman, A. and McGaffey, A. 1997 "An evaluation of Level of Detail Degradation in Head- Mounted Display Peripheries". *Prescence*, 6, 6, 630-637.
Watson et al. 1997b;	Watson, B., Walker, N., Hodges, L.F., and Worden, A. 1997 "Managing Level of Detail through Peripheral Degradation: Effects on Search Performance with a Head- Mounted Display". *ACM Transactions on Computer- Human Interaction 4*, 4 (December 1997), 323-346.
Watson et al. 2000; 2001.	Watson, B., Friedman, A. and McGaffey, A. 2000. "Using naming time to evaluate quality predictors for model simplification." Proceedings of the *SIGCHI conference on Human factors in computing systems*, 113-120.
Watson, Friedman, and McGaffey 2001	WATSON, B., FRIEDMAN, A., MCGAFFEY, A., 2001 Measuring and Predicting Visual Fidelity International Conference on Computer Graphics and Interactive Techniques Proceedings of the 28th annual conference on Computer graphics and interactive techniques 2001 pp: 213 - 220
Whitted 1980	Whitted T. 1980. "An Improved Illumination Model for Shaded Display," Communications of the ACM, vol. 23, no. 6, 343-349.
Wooding 2002	Wooding, D.S. 2002 "Eye movements of large populations: II. Deriving regions of interest, coverage, and similarity using fixation maps." *Behaviour Research Methods, Instruments & Computers*. 34(4), 518-528.
Yee et al. 2001 and Haber et al. 2001	Yee, H., Pattanaik, S., and Greenberg, D.P. 2001 "Spatiotemporal sensitivity and Visual Attention for efficient rendering of dynamic Environments", In *ACM Transactions on Computer Graphics*, Vol. 20, 39-65.
Zotos A, Mania K, Mourkoussis N 2009a	A Schema-based Selective Rendering Framework, APGV 2009

Zotos A, Mania K, Mourkoussis N 2009b   A Selective Rendering Algorithm based on Memory Schemas, Siggraph 2009

3d model repositories   http://archive3d.net/
http://www.3dmodelfree.com/
http://artist-3d.com/

APPENDIX A

First Pilot Study

Ενότητα A: Φόρμα συγκατάθεσης

Project Title: *Real time selective rendering algorithm based on spatial cognition*

Παρακαλώ διαβάστε τα παρακάτω

Η έρευνα και το πειραματικό στάδιο της παρούσας μεταπτυχιακής εργασίας πραγματοποιείται στο Πολυτεχνείο Κρήτης, στο τμήμα Ηλεκτρονικών Μηχανικών και Μηχανικών Υπολογιστών. Θα σας ζητηθεί να συμπληρώσετε ένα ερωτηματολόγιο.

Υπεύθυνοι αυτού του πειράματος είναι ο μεταπτυχιακός φοιτητής Ζώτος Αλέξανδρος και η επίκουρη καθηγήτρια Μανιά Κατερίνα. Η έρευνα πραγματοποιείται για τον καθορισμό της πιστότητας σε εικονικά περιβάλλοντα. Αναμένεται να απασχοληθείτε το πολύ 15 λεπτά σε κάθε ενότητα. Θα χρησιμοποιήσουμε τα δεδομένα σας ανώνυμα μαζί με πολλά άλλα άλλων συμμετεχόντων.

Να θυμάστε ότι η συμμετοχή σας είναι **εθελοντική**. Μπορείτε να επιλέξετε να **μη συμμετάσχετε** σε κανένα ή σε οποιοδήποτε από τα επιμέρους στάδια του ερωτηματολογίου .

Παρακαλώ κυκλώστε τις απαντήσεις σας για τα παρακάτω θέματα:

Κατανοείτε τη φόρμα συγκατάθεσης;	Ναι	Όχι
Δίνετε τα δεδομένα σας για περαιτέρω ανάλυση στα πλαίσια της παρούσας εργασίας;	Ναι	Όχι
Επιβεβαιώνετε ότι είστε τουλάχιστον 18 χρονών;	Ναι	Όχι

Υπογραφή_____   Ημερομηνία_____

Όνομα _____

Ενότητα Β: Προσωπικά στοιχεία

Παρακαλώ συμπληρώστε τα ακόλουθα κενά:

Όνομα: _____

Ηλικιακή ομάδα: 18-22   23-27   28-32   33-37   38-42   πάνω από 43

Φύλο: Αρσενικό / Θηλυκό

Ημερομηνία και ώρα: _____   _____

(Αν σπουδάζετε τώρα

παρακαλώ επιλέξτε ποια

είναι η βαθμίδα σας):   Προπτυχιακός   Μεταπτυχιακός   Διδακτορικός

Τρέχουσα κατάσταση: Φοιτητής   Ερευνητής

                              Ακαδημαϊκός   Άλλο (αναφέρατε) _____

Ενότητα Γ: Schema expectancy

Παρακαλώ διαβάστε τα παρακάτω:

Στις ακόλουθες σελίδες θα βρείτε μία λίστα από αντικείμενα. Δίπλα από κάθε αντικείμενο θα συναντήσετε μία κλίμακα από 1 έως 6. Θα θέλαμε να χρησιμοποιήσετε αυτή την κλίμακα με σκοπό να αξιολογήσετε πόσο πιθανό είναι να βρεθεί το εκάστοτε αντικείμενο στη **κουζίνα** ενός σπιτιού. Ως παράδειγμα θεωρήστε πως αν έπρεπε να απαντήσετε το ίδιο ερωτηματολόγιο για ένα εστιατόριο θα δίνατε *μεγάλο βαθμό πιθανότητας* σε ένα τραπέζι ενώ σε ένα κρεβάτι θα δίνατε ένα *μικρό βαθμό πιθανότητας*. Όταν απαντάτε στο ερωτηματολόγιο παρακαλώ αγνοήστε αν τα συγκεκριμένα αντικείμενα είναι παρόντα ή όχι στο δωμάτιο που βρίσκεστε, απλά βασιστείτε στη μέχρι τώρα εμπειρία σας για το συγκεκριμένο χώρο.

Μία απάντηση της τάξης 1 θεωρείται ως "εξαιρετικά **απίθανο** αυτό το αντικείμενο να **βρίσκεται** στη κουζίνα ενός σπιτιού", και μια απάντηση της τάξης του 6 ορίζεται ως "εξαιρετικά **πιθανό** αυτό το αντικείμενο να **βρίσκεται** στη κουζίνα ενός σπιτιού."

Παρακαλώ κυκλώστε την απάντηση σας

Αλατιέρα	μη πιθανό	1	2	3	4	5	6	πιθανό
Αναψυκτικό	μη πιθανό	1	2	3	4	5	6	πιθανό
Ανδροειδές (κούκλα πραγματικών διαστάσεων)	μη πιθανό	1	2	3	4	5	6	πιθανό
Απορροφητήρας	μη πιθανό	1	2	3	4	5	6	πιθανό
Βιολί	μη πιθανό	1	2	3	4	5	6	πιθανό
Γάλα	μη πιθανό	1	2	3	4	5	6	πιθανό
Εκτυπωτής	μη πιθανό	1	2	3	4	5	6	πιθανό
Ζακέτα	μη πιθανό	1	2	3	4	5	6	πιθανό
Ζάχαρη	μη πιθανό	1	2	3	4	5	6	πιθανό
Ηχεία	μη πιθανό	1	2	3	4	5	6	πιθανό

Καρέκλες	μη πιθανό	1	2	3	4	5	6	πιθανό
Κατσαρόλα	μη πιθανό	1	2	3	4	5	6	πιθανό
Καφές	μη πιθανό	1	2	3	4	5	6	πιθανό
Κονσέρβα	μη πιθανό	1	2	3	4	5	6	πιθανό
Κοτόπουλο μπούτι	μη πιθανό	1	2	3	4	5	6	πιθανό
Κουζίνα	μη πιθανό	1	2	3	4	5	6	πιθανό
Κουρδιστό σκιουράκι	μη πιθανό	1	2	3	4	5	6	πιθανό
Κουτί υπολογιστή	μη πιθανό	1	2	3	4	5	6	πιθανό
Κοχύλι	μη πιθανό	1	2	3	4	5	6	πιθανό
Κρασί	μη πιθανό	1	2	3	4	5	6	πιθανό
Κρέας	μη πιθανό	1	2	3	4	5	6	πιθανό
Κρουασάν	μη πιθανό	1	2	3	4	5	6	πιθανό
Κυάλια	μη πιθανό	1	2	3	4	5	6	πιθανό
Μακαρόνια	μη πιθανό	1	2	3	4	5	6	πιθανό
Μαρμελάδα	μη πιθανό	1	2	3	4	5	6	πιθανό
Μαχαιροπίρουνα	μη πιθανό	1	2	3	4	5	6	πιθανό
Μέλι	μη πιθανό	1	2	3	4	5	6	πιθανό
Μίξερ	μη πιθανό	1	2	3	4	5	6	πιθανό
Μολύβια	μη πιθανό	1	2	3	4	5	6	πιθανό
Μολυβοθήκη	μη πιθανό	1	2	3	4	5	6	πιθανό
Μπάλα Golf	μη πιθανό	1	2	3	4	5	6	πιθανό
Μπαστούνι Baseball	μη πιθανό	1	2	3	4	5	6	πιθανό
Μπρίκι	μη πιθανό	1	2	3	4	5	6	πιθανό
Μπύρα	μη πιθανό	1	2	3	4	5	6	πιθανό

Νεροπίστολο	μη πιθανό	1	2	3	4	5	6	πιθανό
Ντουλάπια κουζίνας	μη πιθανό	1	2	3	4	5	6	πιθανό
Ξηρούς καρπούς	μη πιθανό	1	2	3	4	5	6	πιθανό
Ο Μπομπ Σφουγγαράκης	μη πιθανό	1	2	3	4	5	6	πιθανό
Οδοντόβουρτσα	μη πιθανό	1	2	3	4	5	6	πιθανό
Οδοντόκρεμα	μη πιθανό	1	2	3	4	5	6	πιθανό
Πάγκος κουζίνας	μη πιθανό	1	2	3	4	5	6	πιθανό
Πιάτο	μη πιθανό	1	2	3	4	5	6	πιθανό
Πιστολάκι μαλλιών	μη πιθανό	1	2	3	4	5	6	πιθανό
Πολεμικό κράνος	μη πιθανό	1	2	3	4	5	6	πιθανό
Ποτήρι	μη πιθανό	1	2	3	4	5	6	πιθανό
Ρόδα ποδηλάτου	μη πιθανό	1	2	3	4	5	6	πιθανό
Ρόπαλο (ρωμαϊκό όπλο με καρφιά και σιδερένια μπάλα)	μη πιθανό	1	2	3	4	5	6	πιθανό
Ρύζι	μη πιθανό	1	2	3	4	5	6	πιθανό
Σαμπουάν	μη πιθανό	1	2	3	4	5	6	πιθανό
Σαπούνι	μη πιθανό	1	2	3	4	5	6	πιθανό
Σίδερο (συσκευή σιδερώματος)	μη πιθανό	1	2	3	4	5	6	πιθανό
Σούπα	μη πιθανό	1	2	3	4	5	6	πιθανό
Σουρωτήρι	μη πιθανό	1	2	3	4	5	6	πιθανό
Σύριγγα	μη πιθανό	1	2	3	4	5	6	πιθανό
Ταψιά	μη πιθανό	1	2	3	4	5	6	πιθανό
Τηλέφωνο	μη πιθανό	1	2	3	4	5	6	πιθανό
Τοστιέρα	μη πιθανό	1	2	3	4	5	6	πιθανό
Τραπέζι	μη πιθανό	1	2	3	4	5	6	πιθανό

Τρυπάνι	μη πιθανό	1	2	3	4	5	6	πιθανό
Τσαγιέρα	μη πιθανό	1	2	3	4	5	6	πιθανό
Υπολογιστής Τσέπης	μη πιθανό	1	2	3	4	5	6	πιθανό
Φούρνος Μικρ.	μη πιθανό	1	2	3	4	5	6	πιθανό
Φρούτα	μη πιθανό	1	2	3	4	5	6	πιθανό
Φτυάρι	μη πιθανό	1	2	3	4	5	6	πιθανό
Φυτό	μη πιθανό	1	2	3	4	5	6	πιθανό
Φωτογραφική μηχανή	μη πιθανό	1	2	3	4	5	6	πιθανό
Χαρταετός	μη πιθανό	1	2	3	4	5	6	πιθανό
Χαρτοπετσέτες	μη πιθανό	1	2	3	4	5	6	πιθανό
Ψαλίδι	μη πιθανό	1	2	3	4	5	6	πιθανό
Ψυγείο	μη πιθανό	1	2	3	4	5	6	πιθανό
Ψωμί	μη πιθανό	1	2	3	4	5	6	πιθανό

Ενότητα Δ: Ανασκόπηση

Παρακαλώ κυκλώστε την απάντηση σας

Ήταν το ερωτηματολόγιο δύσκολο για να το συμπληρώσετε?   Ναι   Όχι

Σας πήρε πολύ ώρα (πάνω από 20 λεπτά)?   Ναι   Όχι

Βρήκατε το ερωτηματολόγιο κουραστικό?   Ναι   Όχι

Αν έχετε απαντήσει (Ναι) σε οποιαδήποτε από τις παραπάνω ερωτήσεις ίσως θεωρήσετε σκόπιμο να γράψετε μια πιο αναλυτική άποψη.

_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____

First Pilot Study

Ενότητα Α: Φόρμα συγκατάθεσης

Project Title: *Real time selective rendering algorithm based on spatial cognition*

Παρακαλώ διαβάστε τα παρακάτω

Η έρευνα και το πειραματικό στάδιο της παρούσας μεταπτυχιακής εργασίας πραγματοποιείται στο Πολυτεχνείο Κρήτης, στο τμήμα Ηλεκτρονικών Μηχανικών και Μηχανικών Υπολογιστών. Θα σας ζητηθεί να συμπληρώσετε ένα ερωτηματολόγιο.

Υπεύθυνοι αυτού του πειράματος είναι ο μεταπτυχιακός φοιτητής Ζώτος Αλέξανδρος και η επίκουρη καθηγήτρια Μανιά Κατερίνα. Η έρευνα πραγματοποιείται για τον καθορισμό της πιστότητας σε εικονικά περιβάλλοντα. Αναμένεται να απασχοληθείτε το πολύ 15 λεπτά σε κάθε ενότητα. Θα χρησιμοποιήσουμε τα δεδομένα σας ανώνυμα μαζί με πολλά άλλα άλλων συμμετεχόντων.

Να θυμάστε ότι η συμμετοχή σας είναι **εθελοντική**. Μπορείτε να επιλέξετε να **μη συμμετάσχετε** σε κανένα ή σε οποιοδήποτε από τα επιμέρους στάδια του ερωτηματολογίου .

Παρακαλώ κυκλώστε τις απαντήσεις σας για τα παρακάτω θέματα:

Κατανοείτε τη φόρμα συγκατάθεσης;	Ναι	Όχι
Δίνετε τα δεδομένα σας για περαιτέρω ανάλυση στα πλαίσια της παρούσας εργασίας;	Ναι	Όχι
Επιβεβαιώνετε ότι είστε τουλάχιστον 18 χρονών;	Ναι	Όχι

Υπογραφή_____ Ημερομηνία_____

Όνομα _____

Ενότητα Β: Προσωπικά στοιχεία

Παρακαλώ συμπληρώστε τα ακόλουθα κενά:

Όνομα: _____

Ηλικιακή ομάδα: 18-22   23-27   28-32   33-37   38-42   πάνω από 43

Φύλο: Αρσενικό / Θηλυκό

Ημερομηνία και ώρα: _____   _____

(Αν σπουδάζετε τώρα

παρακαλώ επιλέξτε ποια

είναι η βαθμίδα σας): Προπτυχιακός   Μεταπτυχιακός   Διδακτορικός

Τρέχουσα κατάσταση: Φοιτητής   Ερευνητής

Ακαδημαϊκός   Άλλο (αναφέρατε) _____

Ενότητα Γ: Schema expectancy

Παρακαλώ διαβάστε τα παρακάτω:

Στις ακόλουθες σελίδες θα βρείτε μία λίστα από αντικείμενα. Δίπλα από κάθε αντικείμενο θα συναντήσετε μία κλίμακα από 1 έως 6. Θα θέλαμε να χρησιμοποιήσετε αυτή την κλίμακα με σκοπό να αξιολογήσετε πόσο πιθανό είναι να βρεθεί το εκάστοτε αντικείμενο στο **_σαλόνι_** ενός σπιτιού. Ως παράδειγμα θεωρήστε πως αν έπρεπε να απαντήσετε το ίδιο ερωτηματολόγιο για ένα εστιατόριο θα δίνατε _μεγάλο βαθμό πιθανότητας_ σε ένα τραπέζι ενώ σε ένα κρεβάτι θα δίνατε ένα _μικρό βαθμό πιθανότητας._ Όταν απαντάτε στο ερωτηματολόγιο παρακαλώ αγνοήστε αν τα συγκεκριμένα αντικείμενα είναι παρόντα ή όχι στο δωμάτιο που βρίσκεστε, απλά βασιστείτε στη μέχρι τώρα εμπειρία σας για το συγκεκριμένο χώρο.

Μία απάντηση της τάξης 1 θεωρείται ως "εξαιρετικά **απίθανο** αυτό το αντικείμενο να **βρίσκεται** στο σαλόνι ενός σπιτιού", και μια απάντηση της τάξης του 6 ορίζεται ως "εξαιρετικά **πιθανό** αυτό το αντικείμενο να **βρίσκεται** στο σαλόνι ενός σπιτιού."

Παρακαλώ κυκλώστε την απάντηση σας

CDs	μη πιθανό	1	2	3	4	5	6	πιθανό
Αναπτήρας	μη πιθανό	1	2	3	4	5	6	πιθανό
Αναψυκτικά	μη πιθανό	1	2	3	4	5	6	πιθανό
Ανθοδοχείο	μη πιθανό	1	2	3	4	5	6	πιθανό
Βαρέλι	μη πιθανό	1	2	3	4	5	6	πιθανό
Βιβλία	μη πιθανό	1	2	3	4	5	6	πιθανό
Γλάστρα	μη πιθανό	1	2	3	4	5	6	πιθανό
Γυαλιά ηλίου	μη πιθανό	1	2	3	4	5	6	πιθανό
Ενυδρείο	μη πιθανό	1	2	3	4	5	6	πιθανό
Έπιπλο βιβλιοθήκης – συρταριέρας	μη πιθανό	1	2	3	4	5	6	πιθανό

Έπιπλο τηλεόρασης	μη πιθανό	1	2	3	4	5	6	πιθανό
Κάδρο	μη πιθανό	1	2	3	4	5	6	πιθανό
Καλόγερος ρούχων	μη πιθανό	1	2	3	4	5	6	πιθανό
Καλοριφέρ	μη πιθανό	1	2	3	4	5	6	πιθανό
Καναπές	μη πιθανό	1	2	3	4	5	6	πιθανό
Καπέλο (αθλητικό)	μη πιθανό	1	2	3	4	5	6	πιθανό
Καπέλο (cowboy)	μη πιθανό	1	2	3	4	5	6	πιθανό
Κεφάλι κατσίκας (διακοσμητικό τοίχου)	μη πιθανό	1	2	3	4	5	6	πιθανό
Κηροπήγιο	μη πιθανό	1	2	3	4	5	6	πιθανό
Κιθάρα	μη πιθανό	1	2	3	4	5	6	πιθανό
Κινητό τηλέφωνο	μη πιθανό	1	2	3	4	5	6	πιθανό
Κλειδιά	μη πιθανό	1	2	3	4	5	6	πιθανό
Κολοκύθα Halloween	μη πιθανό	1	2	3	4	5	6	πιθανό
Κορνίζα	μη πιθανό	1	2	3	4	5	6	πιθανό
Κούπα	μη πιθανό	1	2	3	4	5	6	πιθανό
Κράνος	μη πιθανό	1	2	3	4	5	6	πιθανό
Λεφτά	μη πιθανό	1	2	3	4	5	6	πιθανό
Λουλούδια	μη πιθανό	1	2	3	4	5	6	πιθανό
Μαξιλάρια καναπέ	μη πιθανό	1	2	3	4	5	6	πιθανό
Μάσκα αερίου	μη πιθανό	1	2	3	4	5	6	πιθανό
Μίξερ	μη πιθανό	1	2	3	4	5	6	πιθανό
Μπάλα μπάσκετ	μη πιθανό	1	2	3	4	5	6	πιθανό
Μπιμπελό	μη πιθανό	1	2	3	4	5	6	πιθανό
Μπισκότα	μη πιθανό	1	2	3	4	5	6	πιθανό

Μπουκάλι (αλκοόλ)	μη πιθανό	1	2	3	4	5	6	πιθανό
Μπρίκι	μη πιθανό	1	2	3	4	5	6	πιθανό
Ντοσιέ	μη πιθανό	1	2	3	4	5	6	πιθανό
Ξυράφι	μη πιθανό	1	2	3	4	5	6	πιθανό
Οδοντόβουρτσα	μη πιθανό	1	2	3	4	5	6	πιθανό
Παιδικά παιχνίδια - Αυτοκινητάκι	μη πιθανό	1	2	3	4	5	6	πιθανό
Παιδικά παιχνίδια – Κούκλα	μη πιθανό	1	2	3	4	5	6	πιθανό
Παιχνιδομηχανή Play station	μη πιθανό	1	2	3	4	5	6	πιθανό
Παντόφλες	μη πιθανό	1	2	3	4	5	6	πιθανό
Παπούτσια	μη πιθανό	1	2	3	4	5	6	πιθανό
Περιοδικό	μη πιθανό	1	2	3	4	5	6	πιθανό
Πίνακας	μη πιθανό	1	2	3	4	5	6	πιθανό
Πίτσα	μη πιθανό	1	2	3	4	5	6	πιθανό
Πολυθρόνα	μη πιθανό	1	2	3	4	5	6	πιθανό
Πορτοφόλι	μη πιθανό	1	2	3	4	5	6	πιθανό
Ποτήρι	μη πιθανό	1	2	3	4	5	6	πιθανό
Πούρα	μη πιθανό	1	2	3	4	5	6	πιθανό
Ραδιοκασετόφωνο	μη πιθανό	1	2	3	4	5	6	πιθανό
Ρακέτα τέννις	μη πιθανό	1	2	3	4	5	6	πιθανό
Ρόδα ποδηλάτου	μη πιθανό	1	2	3	4	5	6	πιθανό
Ρούχα	μη πιθανό	1	2	3	4	5	6	πιθανό
Σαπούνι	μη πιθανό	1	2	3	4	5	6	πιθανό
Σοκολατάκια	μη πιθανό	1	2	3	4	5	6	πιθανό
Σπαθί	μη πιθανό	1	2	3	4	5	6	πιθανό

Σπιρτόκουτο	μη πιθανό	1	2	3	4	5	6	πιθανό
Συραπτικό	μη πιθανό	1	2	3	4	5	6	πιθανό
Σύριγγα	μη πιθανό	1	2	3	4	5	6	πιθανό
Σφυρί	μη πιθανό	1	2	3	4	5	6	πιθανό
Τασάκι	μη πιθανό	1	2	3	4	5	6	πιθανό
Τηλεκοντρόλ	μη πιθανό	1	2	3	4	5	6	πιθανό
Τηλεόραση	μη πιθανό	1	2	3	4	5	6	πιθανό
Την κούκλα Mr. Potato	μη πιθανό	1	2	3	4	5	6	πιθανό
Τούρτα	μη πιθανό	1	2	3	4	5	6	πιθανό
Τραπεζάκι	μη πιθανό	1	2	3	4	5	6	πιθανό
Τρομπέτα	μη πιθανό	1	2	3	4	5	6	πιθανό
Τσαγιέρα	μη πιθανό	1	2	3	4	5	6	πιθανό
Τσιγάρα	μη πιθανό	1	2	3	4	5	6	πιθανό
Φυτό	μη πιθανό	1	2	3	4	5	6	πιθανό
Φωτιστικό δαπέδου	μη πιθανό	1	2	3	4	5	6	πιθανό
Χαλί – Μοκέτα	μη πιθανό	1	2	3	4	5	6	πιθανό
Χειροβομβίδα	μη πιθανό	1	2	3	4	5	6	πιθανό
Χειροπέδες	μη πιθανό	1	2	3	4	5	6	πιθανό
Ψαλίδι	μη πιθανό	1	2	3	4	5	6	πιθανό

Ενότητα Δ: Schema expectancy

Παρακαλώ διαβάστε τα παρακάτω:

Στις ακόλουθες σελίδες θα βρείτε μία λίστα από αντικείμενα. Δίπλα από κάθε αντικείμενο θα συναντήσετε μία κλίμακα από 1 έως 6. Θα θέλαμε να χρησιμοποιήσετε αυτή την κλίμακα με σκοπό να αξιολογήσετε πόσο πιθανό είναι να βρεθεί το εκάστοτε αντικείμενο **πάνω στον καναπέ** ενός σαλονιού. Ως παράδειγμα θεωρήστε πως αν έπρεπε να απαντήσετε το ίδιο ερωτηματολόγιο για ένα εστιατόριο θα δίνατε *μεγάλο βαθμό πιθανότητας* σε ένα τραπέζι ενώ σε ένα κρεβάτι θα δίνατε *ένα μικρό βαθμό πιθανότητας.* Όταν απαντάτε στο ερωτηματολόγιο παρακαλώ αγνοήστε αν τα συγκεκριμένα αντικείμενα είναι παρόντα ή όχι στο δωμάτιο που βρίσκεστε, απλά βασιστείτε στη μέχρι τώρα εμπειρία σας για το συγκεκριμένο χώρο.

Μία απάντηση της τάξης 1 θεωρείται ως "εξαιρετικά **απίθανο** αυτό το αντικείμενο να **βρίσκεται** πάνω στον καναπέ ενός σαλονιού", και μια απάντηση της τάξης του 6 ορίζεται ως "εξαιρετικά **πιθανό** αυτό το αντικείμενο να **βρίσκεται** πάνω στον καναπέ ενός σαλονιού."

Παρακαλώ κυκλώστε την απάντηση σας

Ανθοδοχείο	μη πιθανό	1	2	3	4	5	6	πιθανό
Κηροπήγειο	μη πιθανό	1	2	3	4	5	6	πιθανό
Κινητό τηλέφωνο	μη πιθανό	1	2	3	4	5	6	Πιθανό
Κλειδιά	μη πιθανό	1	2	3	4	5	6	Πιθανό
Μαξιλάρια καναπέ	μη πιθανό	1	2	3	4	5	6	Πιθανό
Μπουκάλι (αλκοόλ)	μη πιθανό	1	2	3	4	5	6	πιθανό
Περιοδικό	μη πιθανό	1	2	3	4	5	6	πιθανό
Ποτήρι	μη πιθανό	1	2	3	4	5	6	πιθανό
Τασάκι	μη πιθανό	1	2	3	4	5	6	πιθανό
Τηλεκοντρόλ	μη πιθανό	1	2	3	4	5	6	πιθανό

Ενότητα Ε: Ανασκόπηση

Παρακαλώ κυκλώστε την απάντηση σας

Ήταν το ερωτηματολόγιο δύσκολο για να το συμπληρώσετε?   Ναι   Όχι

Σας πήρε πολύ ώρα (πάνω από 20 λεπτά)?   Ναι   Όχι

Βρήκατε το ερωτηματολόγιο κουραστικό?   Ναι   Όχι

Αν έχετε απαντήσει (Ναι) σε οποιαδήποτε από τις παραπάνω ερωτήσεις ίσως θεωρήσετε σκόπιμο να γράψετε μια πιο αναλυτική άποψη.

_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____

First Pilot Study

Ενότητα Α: Φόρμα συγκατάθεσης

Project Title: *Real time selective rendering algorithm based on spatial cognition*

Παρακαλώ διαβάστε τα παρακάτω

Η έρευνα και το πειραματικό στάδιο της παρούσας μεταπτυχιακής εργασίας πραγματοποιείται στο Πολυτεχνείο Κρήτης, στο τμήμα Ηλεκτρονικών Μηχανικών και Μηχανικών Υπολογιστών. Θα σας ζητηθεί να συμπληρώσετε ένα ερωτηματολόγιο.

Υπεύθυνοι αυτού του πειράματος είναι ο μεταπτυχιακός φοιτητής Ζώτος Αλέξανδρος και η επίκουρη καθηγήτρια Μανιά Κατερίνα. Η έρευνα πραγματοποιείται για τον καθορισμό της πιστότητας σε εικονικά περιβάλλοντα. Αναμένεται να απασχοληθείτε το πολύ 15 λεπτά σε κάθε ενότητα. Θα χρησιμοποιήσουμε τα δεδομένα σας ανώνυμα μαζί με πολλά άλλα άλλων συμμετεχόντων.

Να θυμάστε ότι η συμμετοχή σας είναι **εθελοντική**. Μπορείτε να επιλέξετε να **μη συμμετάσχετε** σε κανένα ή σε οποιοδήποτε από τα επιμέρους στάδια του ερωτηματολογίου .

Παρακαλώ κυκλώστε τις απαντήσεις σας για τα παρακάτω θέματα:

Κατανοείτε τη φόρμα συγκατάθεσης;	Ναι	Όχι
Δίνετε τα δεδομένα σας για περαιτέρω ανάλυση στα πλαίσια της παρούσας εργασίας;	Ναι	Όχι
Επιβεβαιώνετε ότι είστε τουλάχιστον 18 χρονών;	Ναι	Όχι

Υπογραφή_____   Ημερομηνία_____

Όνομα _____

Ενότητα Β: Προσωπικά στοιχεία

Παρακαλώ συμπληρώστε τα ακόλουθα κενά:

Όνομα: _____

Ηλικιακή ομάδα:　　　　18-22　　23-27　　28-32　　33-37　　38-42　　πάνω από 43

Φύλο:　　　　　　　Αρσενικό / Θηλυκό

Ημερομηνία και ώρα:　　　_____　　　　_____

(Αν σπουδάζετε τώρα

παρακαλώ επιλέξτε ποια

είναι η βαθμίδα σας):　　Προπτυχιακός　　Μεταπτυχιακός　　　Διδακτορικός

Τρέχουσα κατάσταση:　　Φοιτητής　　Ερευνητής

　　　　　　　　　Ακαδημαϊκός　Άλλο (αναφέρατε) _____

Ενότητα Γ: Schema expectancy

Παρακαλώ διαβάστε τα παρακάτω:

Στις ακόλουθες σελίδες θα βρείτε μία λίστα από αντικείμενα. Δίπλα από κάθε αντικείμενο θα συναντήσετε μία κλίμακα από 1 έως 6. Θα θέλαμε να χρησιμοποιήσετε αυτή την κλίμακα με σκοπό να αξιολογήσετε πόσο πιθανό είναι να βρεθεί το εκάστοτε αντικείμενο σε έναν **_ακαδημαϊκό χώρο εργασίας_**. Ως παράδειγμα θεωρήστε πως αν έπρεπε να απαντήσετε το ίδιο ερωτηματολόγιο για ένα εστιατόριο θα δίνατε _μεγάλο βαθμό πιθανότητας_ σε ένα τραπέζι ενώ σε ένα κρεβάτι θα δίνατε _ένα μικρό βαθμό πιθανότητας_. Όταν απαντάτε στο ερωτηματολόγιο παρακαλώ αγνοήστε αν τα συγκεκριμένα αντικείμενα είναι παρόντα ή όχι στο δωμάτιο που βρίσκεστε, απλά βασιστείτε στη μέχρι τώρα εμπειρία σας για το συγκεκριμένο χώρο.

Μία απάντηση της τάξης 1 θεωρείται ως "εξαιρετικά **απίθανο** αυτό το αντικείμενο να **βρίσκεται** στον ακαδημαϊκό χώρο εργασίας", και μια απάντηση της τάξης του 6 ορίζεται ως "εξαιρετικά **πιθανό** αυτό το αντικείμενο να **βρίσκεται** στον ακαδημαϊκό χώρο εργασίας."

Παρακαλώ κυκλώστε την απάντηση σας

Ακουστικά	μη πιθανό	1	2	3	4	5	6	πιθανό
Αναπτήρας	μη πιθανό	1	2	3	4	5	6	πιθανό
Βιβλία	μη πιθανό	1	2	3	4	5	6	πιθανό
Βιβλιοθήκη	μη πιθανό	1	2	3	4	5	6	πιθανό
Γλάστρα	μη πιθανό	1	2	3	4	5	6	πιθανό
Γραφείο	μη πιθανό	1	2	3	4	5	6	πιθανό
Εκτυπωτής	μη πιθανό	1	2	3	4	5	6	πιθανό
Ηλεκτρική κιθάρα	μη πιθανό	1	2	3	4	5	6	πιθανό
Κάδος απορριμμάτων	μη πιθανό	1	2	3	4	5	6	πιθανό
Κάδρο	μη πιθανό	1	2	3	4	5	6	πιθανό

Κλειδί	μη πιθανό	1	2	3	4	5	6	πιθανό
Κούπα καφέ	μη πιθανό	1	2	3	4	5	6	πιθανό
Κουτί με cds	μη πιθανό	1	2	3	4	5	6	πιθανό
Κουτί υπολογιστή	μη πιθανό	1	2	3	4	5	6	πιθανό
Κράνος αμερικάνικου ποδοσφαίρου	μη πιθανό	1	2	3	4	5	6	πιθανό
Λαπτοπ	μη πιθανό	1	2	3	4	5	6	πιθανό
Μολύβια	μη πιθανό	1	2	3	4	5	6	πιθανό
Μολυβοθήκη	μη πιθανό	1	2	3	4	5	6	πιθανό
Μπαλάκι Τέννις	μη πιθανό	1	2	3	4	5	6	πιθανό
Ντοσιέ	μη πιθανό	1	2	3	4	5	6	πιθανό
Ξυράφι	μη πιθανό	1	2	3	4	5	6	πιθανό
Οθόνη Υπολογιστή	μη πιθανό	1	2	3	4	5	6	πιθανό
Περιστρεφόμενη καρέκλα γραφείου	μη πιθανό	1	2	3	4	5	6	πιθανό
Πίνακας ημερολογίου / σημειώσεων	μη πιθανό	1	2	3	4	5	6	πιθανό
Πινακίδα με κλιπ χαρτιών	μη πιθανό	1	2	3	4	5	6	πιθανό
Πιστολάκι μαλλιών	μη πιθανό	1	2	3	4	5	6	πιθανό
Πιστωτική κάρτα	μη πιθανό	1	2	3	4	5	6	πιθανό
Ποντίκι Υπολογιστή	μη πιθανό	1	2	3	4	5	6	πιθανό
Πυροσβεστήρας	μη πιθανό	1	2	3	4	5	6	πιθανό
Ρακέτα τέννις	μη πιθανό	1	2	3	4	5	6	πιθανό
Ράφια Τοίχου	μη πιθανό	1	2	3	4	5	6	πιθανό
Ρολόι τοίχου	μη πιθανό	1	2	3	4	5	6	πιθανό
Σελοτέηπ	μη πιθανό	1	2	3	4	5	6	πιθανό
Σπαθί	μη πιθανό	1	2	3	4	5	6	πιθανό

Στυλό	μη πιθανό	1	2	3	4	5	6	πιθανό
Συραπτικό	μη πιθανό	1	2	3	4	5	6	πιθανό
Ταμιακή μηχανή	μη πιθανό	1	2	3	4	5	6	πιθανό
Τηγάνι	μη πιθανό	1	2	3	4	5	6	πιθανό
Τηλέφωνο	μη πιθανό	1	2	3	4	5	6	πιθανό
Τρομπέτα	μη πιθανό	1	2	3	4	5	6	πιθανό
Υπολογιστής Τσέπης	μη πιθανό	1	2	3	4	5	6	πιθανό
Φρυγανιέρα	μη πιθανό	1	2	3	4	5	6	πιθανό
Φωτιστικό	μη πιθανό	1	2	3	4	5	6	πιθανό
Χάρακας	μη πιθανό	1	2	3	4	5	6	πιθανό
Ψαλίδι	μη πιθανό	1	2	3	4	5	6	πιθανό

Ενότητα Δ: Schema expectancy

Παρακαλώ διαβάστε τα παρακάτω:

Στις ακόλουθες σελίδες θα βρείτε μία λίστα από αντικείμενα. Δίπλα από κάθε αντικείμενο θα συναντήσετε μία κλίμακα από 1 έως 6. Θα θέλαμε να χρησιμοποιήσετε αυτή την κλίμακα με σκοπό να αξιολογήσετε πόσο πιθανό είναι να βρεθεί το εκάστοτε αντικείμενο **πάνω στο γραφείο** ενός ακαδημαϊκού χώρου. Ως παράδειγμα θεωρήστε πως αν έπρεπε να απαντήσετε το ίδιο ερωτηματολόγιο για ένα εστιατόριο θα δίνατε *μεγάλο βαθμό πιθανότητας* σε ένα τραπέζι ενώ σε ένα κρεβάτι θα δίνατε ένα *μικρό βαθμό πιθανότητας*. Όταν απαντάτε στο ερωτηματολόγιο παρακαλώ αγνοήστε αν τα συγκεκριμένα αντικείμενα είναι παρόντα ή όχι στο δωμάτιο που βρίσκεστε, απλά βασιστείτε στη μέχρι τώρα εμπειρία σας για το συγκεκριμένο χώρο.

Μία απάντηση της τάξης 1 θεωρείται ως "εξαιρετικά **απίθανο** αυτό το αντικείμενο να **βρίσκεται** πάνω στο γραφείο", και μια απάντηση της τάξης του 6 ορίζεται ως "εξαιρετικά **πιθανό** αυτό το αντικείμενο να **βρίσκεται** πάνω στο γραφείο ενός ακαδημαϊκού χώρου."

Παρακαλώ κυκλώστε την απάντηση σας

Γλάστρα	μη πιθανό	1	2	3	4	5	6	πιθανό
Ηλεκτρική κιθάρα	μη πιθανό	1	2	3	4	5	6	πιθανό
Κάδος απορριμμάτων	μη πιθανό	1	2	3	4	5	6	πιθανό
Κλειδί	μη πιθανό	1	2	3	4	5	6	πιθανό
Κουτί με cds	μη πιθανό	1	2	3	4	5	6	πιθανό
Κουτί υπολογιστή	μη πιθανό	1	2	3	4	5	6	πιθανό
Μολύβια	μη πιθανό	1	2	3	4	5	6	πιθανό
Μολυβοθήκη	μη πιθανό	1	2	3	4	5	6	πιθανό
Οθόνη Υπολογιστή	μη πιθανό	1	2	3	4	5	6	πιθανό
Πιστωτική κάρτα	μη πιθανό	1	2	3	4	5	6	πιθανό

Ποντίκι Υπολογιστή	μη πιθανό	1	2	3	4	5	6	πιθανό
Πυροσβεστήρας	μη πιθανό	1	2	3	4	5	6	πιθανό
Ρολόι τοίχου	μη πιθανό	1	2	3	4	5	6	πιθανό
Σελοτέηπ	μη πιθανό	1	2	3	4	5	6	πιθανό
Φωτιστικό	μη πιθανό	1	2	3	4	5	6	πιθανό

Ενότητα Ε: Schema expectancy

Παρακαλώ διαβάστε τα παρακάτω:

Στις ακόλουθες σελίδες θα βρείτε μία λίστα από αντικείμενα. Δίπλα από κάθε αντικείμενο θα συναντήσετε μία κλίμακα από 1 έως 6. Θα θέλαμε να χρησιμοποιήσετε αυτή την κλίμακα με σκοπό να αξιολογήσετε πόσο πιθανό είναι να βρεθεί το εκάστοτε αντικείμενο **στα ράφια της βιβλιοθήκης** ενός ακαδημαϊκού χώρου. Ως παράδειγμα θεωρήστε πως αν έπρεπε να απαντήσετε το ίδιο ερωτηματολόγιο για ένα εστιατόριο θα δίνατε *μεγάλο βαθμό πιθανότητας* σε ένα τραπέζι ενώ σε ένα κρεβάτι θα δίνατε ένα *μικρό βαθμό πιθανότητας*. Όταν απαντάτε στο ερωτηματολόγιο παρακαλώ αγνοήστε αν τα συγκεκριμένα αντικείμενα είναι παρόντα ή όχι στο δωμάτιο που βρίσκεστε, απλά βασιστείτε στη μέχρι τώρα εμπειρία σας για το συγκεκριμένο χώρο.

Μία απάντηση της τάξης 1 θεωρείται ως "εξαιρετικά **απίθανο** αυτό το αντικείμενο να **βρίσκεται** στη βιβλιοθήκη", και μια απάντηση της τάξης του 6 ορίζεται ως "εξαιρετικά **πιθανό** αυτό το αντικείμενο να **βρίσκεται** στη βιβλιοθήκη ενός ακαδημαϊκού χώρου."


Παρακαλώ κυκλώστε την απάντηση σας

Ακουστικά	μη πιθανό	1	2	3	4	5	6	πιθανό
Αναπτήρας	μη πιθανό	1	2	3	4	5	6	πιθανό
Βιβλία	μη πιθανό	1	2	3	4	5	6	πιθανό
Εκτυπωτής	μη πιθανό	1	2	3	4	5	6	πιθανό
Κούπα καφέ	μη πιθανό	1	2	3	4	5	6	πιθανό
Λαπτοπ	μη πιθανό	1	2	3	4	5	6	πιθανό
Μπαλάκι τέννις	μη πιθανό	1	2	3	4	5	6	πιθανό
Μολυβοθήκη	μη πιθανό	1	2	3	4	5	6	πιθανό
Ποντίκι Υπολογιστή	μη πιθανό	1	2	3	4	5	6	πιθανό
Στυλό	μη πιθανό	1	2	3	4	5	6	πιθανό

Συραπτικό	μη πιθανό	1	2	3	4	5	6	πιθανό
Τηλέφωνο	μη πιθανό	1	2	3	4	5	6	Πιθανό
Τρομπέτα	μη πιθανό	1	2	3	4	5	6	Πιθανό
Υπολογιστής Τσέπης	μη πιθανό	1	2	3	4	5	6	πιθανό
Χάρακας	μη πιθανό	1	2	3	4	5	6	πιθανό

Ενότητα Ζ: Ανασκόπηση

Παρακαλώ κυκλώστε την απάντηση σας

Ήταν το ερωτηματολόγιο δύσκολο για να το συμπληρώσετε?   Ναι   Όχι

Σας πήρε πολύ ώρα (πάνω από 20 λεπτά)?   Ναι   Όχι

Βρήκατε το ερωτηματολόγιο κουραστικό?   Ναι   Όχι

Αν έχετε απαντήσει (Ναι)σε οποιαδήποτε από τις παραπάνω ερωτήσεις ίσως θεωρήσετε σκόπιμο να γράψετε μια πιο αναλυτική άποψη.

_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____

APPENDIX B

Kitchen					
4 =< Expectant =< 6		1 =< Non-Expectant =< 3		3 < Neutral < 4	
Ξηρούς καρπούς	4,043478	Ρόπαλο (ρωμαϊκό όπλο με καρφιά και σιδερένια μπάλα)	1	Φυτό	3,347826
Κοτόπουλο μπούτι	4,086957	Μπαστούνι Baseball	1,086957	Ψαλίδι	3,521739
Σούπα	4,086957	Πολεμικό κράνος	1,086957	Σαπούνι	3,695652
Κρουασάν	4,26087	Ρόδα ποδηλάτου	1,086957	Τηλέφωνο	3,73913
Μπύρα	4,521739	Φτυάρι	1,086957		
Κονσέρβα	4,652174	Χαρταετός	1,086957		
Τσαγιέρα	4,652174	Μπάλα Golf	1,130435		
Μίξερ	4,695652	Οδοντόβουρτσα	1,130435		
Μαρμελάδα	4,782609	Οδοντόκρεμα	1,173913		
Μέλι	4,782609	Ανδροειδές (κούκλα πραγματικών διαστάσεων)	1,26087		
Κρασί	4,826087	Πιστολάκι μαλλιών	1,304348		
Κρέας	4,826087	Σαμπουάν	1,304348		
Αναψυκτικό	4,869565	Βιολί	1,347826		
Φούρνος Μικρ.	4,869565	Εκτυπωτής	1,347826		
Γάλα	4,956522	Κουτί υπολογιστή	1,434783		
Ρύζι	5	Κυάλια	1,434783		
Τοστιέρα	5,043478	Σύριγγα	1,434783		
Σουρωτήρι	5,130435	Τρυπάνι	1,434783		
Μακαρόνια	5,304348	Νεροπίστολο	1,478261		
Ταψιά	5,434783	Ο Μπομπ Σφουγγαράκης	1,478261		
Φρούτα	5,434783	Κουρδιστό σκιουράκι	1,521739		
Ψωμί	5,434783	Μολυβοθήκη	1,695652		
Καφές	5,521739	Φωτογραφική μηχανή	1,695652		
Καρέκλες	5,695652	Σίδερο (συσκευή σιδερώματος)	1,73913		
Πάγκος κουζίνας	5,695652	Ηχεία	2,217391		
Απορροφητήρας	5,73913	Κοχύλι	2,26087		
Μπρίκι	5,73913	Υπολογιστής Τσέπης	2,391304		
Ζάχαρη	5,782609	Ζακέτα	2,521739		
Χαρτοπετσέτες	5,782609	Μολύβια	2,681818		
Αλατιέρα	5,826087				

Πιάτο	5,826087				
Κουζίνα	5,869565				
Μαχαιροπίρουνα	5,869565				
Ποτήρι	5,869565				
Τραπέζι	5,869565				
Κατσαρόλα	5,956522				
Ντουλάπια κουζίνας	6				
Ψυγείο	6				

Lounge					
4 =< Expectant =< 6		1 =< Non-Expectant =< 3		3 < Neutral < 4	
Γλάστρα	4,217391	Μάσκα αερίου	1	Τούρτα	2,217391
Καλόγερος ρούχων	4,217391	Χειροβομβίδα	1	Τσαγιέρα	2,26087
Μπουκάλι (αλκοόλ)	4,217391	Ρόδα ποδηλάτου	1,043478	Καπέλο (αθλητικό)	2,304348
Ποτήρι	4,26087	Οδοντόβουρτσα	1,086957	Ντοσιέ	2,434783
Κλειδιά	4,304348	Ξυράφι	1,130435	Μπισκότα	2,695652
Βιβλία	4,347826	Σαπούνι	1,130435	Παπούτσια	2,73913
Σοκολατάκια	4,347826	Βαρέλι	1,173913	Σπιρτόκουτο	2,782609
Τσιγάρα	4,391304	Σφυρί	1,173913	Κιθάρα	2,826087
Αναπτήρας	4,521739	Μίξερ	1,217391	Παιδικά παιχνίδια - Αυτοκινητάκι	2,956522
Κηροπήγιο	4,565217	Χειροπέδες	1,217391		
Κινητό τηλέφωνο	4,652174	Σύριγγα	1,26087		
Φωτιστικό δαπέδου	4,652174	Καπέλο (cowboy)	1,347826		
Περιοδικό	4,695652	Κολοκύθα Halloween	1,434783		
Φυτό	4,782609	Μπρίκι	1,434783		
Λουλούδια	4,826087	Ρακέτα τέννις	1,434783		
Ραδιοκασετόφωνο	4,826087	Μπάλα μπάσκετ	1,478261		
Πίνακας	4,913043	Τρομπέτα	1,478261		
Ανθοδοχείο	4,956522	Κεφάλι κατσίκας (διακοσμητικό τοίχου)	1,565217		
CDs	5	Κράνος	1,73913		
Έπιπλο βιβλιοθήκης – συρταριέρας	5,130435	Σπαθί	1,73913		
Κορνίζα	5,217391	Συραπτικό	1,782609		
Τασάκι	5,26087	Την κούκλα Mr. Potato	1,782609		
Τηλεκοντρόλ	5,478261	Ψαλίδι	1,956522		
Χαλί – Μοκέτα	5,478261	Τούρτα	2,217391		
Τραπεζάκι	5,521739	Τσαγιέρα	2,26087		
Κάδρο	5,608696	Καπέλο (αθλητικό)	2,304348		
Τηλεόραση	5,608696	Ντοσιέ	2,434783		
Έπιπλο τηλεόρασης	5,652174	Μπισκότα	2,695652		
Μαξιλάρια καναπέ	5,652174	Παπούτσια	2,73913		

Καλοριφέρ	5,695652	Σπιρτόκουτο	2,782609		
Πολυθρόνα	5,695652	Κιθάρα	2,826087		
Καναπές	6	Παιδικά παιχνίδια - Αυτοκινητάκι	2,956522		
		Παιδικά παιχνίδια – Κούκλα	3		
		Πίτσα	3		

Schema Lounge		Sub Schema Couch	
Μπουκάλι (αλκοόλ)	4,217391	Μπουκάλι (αλκοόλ)	2
Ποτήρι	4,26087	Ποτήρι	1,869565
Κλειδιά	4,304348	Κλειδιά	3,826087
Κηροπήγιο	4,565217	Κηροπήγειο	1,173913
Κινητό τηλέφωνο	4,652174	Κινητό τηλέφωνο	4,26087
Περιοδικό	4,695652	Περιοδικό	4,26087
Ανθοδοχείο	4,956522	Ανθοδοχείο	1,173913
Τασάκι	5,26087	Τασάκι	2,695652
Τηλεκοντρόλ	5,478261	Τηλεκοντρόλ	4,869565
Μαξιλάρια καναπέ	5,652174	Μαξιλάρια καναπέ	5,956522

Office					
4 =< Expectant =< 6		1 =< Non-Expectant =< 3		3 < Neutral < 4	
Ακουστικά	4,304348	Σπαθί	1	Γλάστρα	3,434783
Αναπτήρας	4,347826	Τηγάνι	1,086957	Κάδρο	3,565217
Σελοτέηπ	4,347826	Τρομπέτα	1,086957		
Ψαλίδι	4,391304	Κράνος αμερικάνικου ποδοσφαίρου	1,130435		
Ρολόι τοίχου	4,434783	Πιστολάκι μαλλιών	1,130435		
Ράφια Τοίχου	4,478261	Ρακέτα τέννις	1,26087		
Πυροσβεστήρας	4,521739	Φρυγανιέρα	1,347826		
Υπολογιστής Τσέπης	4,608696	Ταμιακή μηχανή	1,391304		
Χάρακας	4,695652	Μπαλάκι Τέννις	1,478261		
Κλειδί	4,73913	Ηλεκτρική κιθάρα	1,608696		
Πινακίδα με κλιπ χαρτιών	4,73913	Ξυράφι	1,608696		
Φωτιστικό	4,782609	Πιστωτική κάρτα	2,391304		
Κουτί με cds	4,913043				
Κούπα καφέ	5				
Πίνακας ημερολογίου / σημειώσεων	5				
Κάδος απορριμμάτων	5,043478				
Μολυβοθήκη	5,304348				
Συραπτικό	5,304348				
Λαπτοπ	5,347826				
Μολύβια	5,347826				
Ντοσιέ	5,434783				
Τηλέφωνο	5,565217				
Εκτυπωτής	5,695652				
Κουτί υπολογιστή	5,782609				
Βιβλία	5,826087				
Βιβλιοθήκη	5,826087				
Οθόνη Υπολογιστή	5,826087				
Περιστρεφόμενη καρέκλα γραφείου	5,826087				
Ποντίκι Υπολογιστή	5,826087				

| Στυλό | 5,913043 | | | | |
| Γραφείο | 6 | | | | |

Schema Office		Sub Schema Desk	
Ηλεκτρική κιθάρα	**1,608696**	**Ηλεκτρική κιθάρα**	**1**
Πιστωτική κάρτα	**2,391304**	**Πιστωτική κάρτα**	**2,913043**
Γλάστρα	**3,434783**	**Γλάστρα**	**1,695652**
Σελοτέηπ	**4,347826**	**Σελοτέηπ**	**4,130435**
Ρολόι τοίχου	**4,434783**	**Ρολόι τοίχου**	**1,086957**
Πυροσβεστήρας	**4,521739**	**Πυροσβεστήρας**	**1,130435**
Κλειδί	**4,73913**	**Κλειδί**	**4,26087**
Φωτιστικό	**4,782609**	**Φωτιστικό**	**4,304348**
Κουτί με cds	**4,913043**	**Κουτί με cds**	**4,434783**
Κάδος απορριμμάτων	**5,043478**	**Κάδος απορριμμάτων**	**1,391304**
Μολυβοθήκη	**5,304348**	**Μολυβοθήκη**	**5,565217**
Μολύβια	**5,347826**	**Μολύβια**	**5,347826**
Κουτί υπολογιστή	**5,782609**	**Κουτί υπολογιστή**	**4,434783**
Οθόνη Υπολογιστή	**5,826087**	**Οθόνη Υπολογιστή**	**5,913043**
Ποντίκι Υπολογιστή	**5,826087**	**Ποντίκι Υπολογιστή**	**5,869565**

Schema Office		Sub Schema Library	
Τρομπέτα	**1,086957**	**Τρομπέτα**	**1,173913**
Μπαλάκι Τέννις	**1,478261**	**Μπαλάκι τέννις**	**1,304348**
Ακουστικά	**4,304348**	**Ακουστικά**	**2,434783**
Αναπτήρας	**4,347826**	**Αναπτήρας**	**1,73913**
Υπολογιστής Τσέπης	**4,608696**	**Υπολογιστής Τσέπης**	**2,434783**
Χάρακας	**4,695652**	**Χάρακας**	**2,434783**
Κούπα καφέ	**5**	**Κούπα καφέ**	**2,086957**
Μολυβοθήκη	**5,304348**	**Μολυβοθήκη**	**2,521739**
Συραπτικό	**5,304348**	**Συραπτικό**	**2,391304**
Λαπτοπ	**5,347826**	**Λαπτοπ**	**1,826087**
Τηλέφωνο	**5,565217**	**Τηλέφωνο**	**2,26087**
Εκτυπωτής	**5,695652**	**Εκτυπωτής**	**2,521739**
Βιβλία	**5,826087**	**Βιβλία**	**6**
Ποντίκι Υπολογιστή	**5,826087**	**Ποντίκι Υπολογιστή**	**1,73913**
Στυλό	**5,913043**	**Στυλό**	**3,043478**

APPENDIX C

<div align="center">**Σενάριο πειράματος**</div>

Σας ευχαριστούμε που συμμετέχετε στη διαδικασία των πειραμάτων μας. Το πείραμα αποτελείται από **3 στάδια**. Ακολουθούν οι λεπτομέρειες για κάθε στάδιο ξεχωριστά. Αφού διαβάσετε προσεκτικά τις λεπτομέρειες για κάθε στάδιο, είστε έτοιμοι να ξεκινήσετε.

Συνοπτικά, το πρώτο στάδιο αποσκοπεί στην εξοικείωση σας με τον εξοπλισμό και τα τρισδιάστατα γραφικά. Το δεύτερο στάδιο είναι και το κύριο πείραμα. Το τελευταίο στάδιο αποτελείται από μία ενότητα ανασκόπησης.

Στάδιο 1. *Εξοικείωση με τον εξοπλισμό και τα τρισδιάστατα γραφικά*

Εξάσκηση σε ένα απλό περιβάλλον

Είστε έτοιμοι να συμμετάσχετε στο πρώτο στάδιο. Σκοπός είναι να εξοικειωθείτε με τον εξοπλισμό (Head Mounted Display ή HMD) και τα τρισδιάστατα γραφικά. Έχετε όσο χρόνο επιθυμείτε για να περιηγηθείτε σε έναν ανοιχτό χώρο που περιέχει διάφορα τρισδιάστατα γεωμετρικά σχήματα όπως κύβοι, σφαίρες και πυραμίδες. Χρησιμοποιήστε τον εξοπλισμό για να μετακινηθείτε προς όλες τις κατευθύνσεις και να κοιτάξετε προς όλες τις γωνίες. Ρωτήστε ελεύθερα για οποιαδήποτε απορία ή διακόψτε την διαδικασία εάν αισθανθείτε αδιαθεσία.

Εξάσκηση σε ένα πιο ρεαλιστικό περιβάλλον

Αφού αισθανθείτε άνετα με τον εξοπλισμό και την περιήγηση στο πρώτο τρισδιάστατο χώρο ζητήστε από τον υπεύθυνο του πειράματος να "φορτώσει" μια καινούργια πιο ρεαλιστική σκηνή. Σκοπός είναι να εγκλιματιστείτε σε πιο ρεαλιστικές συνθήκες από τις προηγούμενες ώστε να είστε έτοιμοι για το επόμενο στάδιο που είναι και το βασικό. Η καινούρια σκηνή περιγράφει ένα **δίχωρο** διαμέρισμα με αντικείμενα και το απαιτούμενο είναι να κινηθείτε σε όλους τους χώρους, ώστε να συνηθίσετε την πλοήγηση ανάμεσα σε αντικείμενα.

Κατά τη διάρκεια της περιήγησης σας στον πρώτο από τους δύο χώρους του διαμερίσματος θα ακουστεί ένας προειδοποιητικός ήχος. Όταν τον ακούσετε, και μόνο τότε, προχωρήστε προς τη πόρτα και μεταφερθείτε στο δεύτερο χώρο του διαμερίσματος. Εκεί περιηγηθείτε αντίστοιχα σε όλες τις περιοχές που αυτός περιέχει.

Στάδιο 2. *Το κύριο πείραμα. Περιήγηση σε μια τρισδιάστατη σκηνή.*

Αφού έχετε εξοικειωθεί πλήρως με τον εξοπλισμό, τη περιήγηση σε τρισδιάστατους ρεαλιστικούς χώρους αλλά και εξωτερικά ερεθίσματα όπως ήχοι, καλείστε να περιηγηθείτε σε μια πολυπλοκότερη σκηνή. Λεπτομέρειες για την πλοήγηση και το σενάριο του πειράματος θα δοθούν αμέσως πριν την πραγματοποίηση του.

Στάδιο 3. *Ανασκόπηση*

Σύμφωνα με την εμπειρία που αποκομίσατε από τη τελευταία σκηνή που είδατε (στάδιο 2), απαντήστε στη φόρμα ανασκόπησης που ακολουθεί. Έχετε όσο χρόνο επιθυμείτε για να διαβάσετε τις οδηγίες.

Παρακαλώ μη συζητήσετε με τους υπόλοιπους συμμετέχοντες οτιδήποτε έχει σχέση με τις εντυπώσεις σας από την πλοήγηση ή το στάδιο ανασκόπησης. Παρόλα αυτά, μπορείτε να ρωτήσετε τους υπεύθυνους του πειράματος για οποιαδήποτε απορία έχετε.

Αυτό είναι και το τελικό στάδιο. Ευχαριστούμε για τη συμμετοχή σας και ελπίζουμε να διασκεδάσατε τη διαδρομή!! ☺

APPENDIX D

Ενότητα Α: Φόρμα Συγκατάθεσης

Project Title: *Real time selective rendering algorithm based on spatial cognition*

Παρακαλώ διαβάστε τα παρακάτω

Η έρευνα και το πειραματικό στάδιο της παρούσας μεταπτυχιακής εργασίας πραγματοποιείται στο Πολυτεχνείο Κρήτης, στο τμήμα Ηλεκτρονικών Μηχανικών και Μηχανικών Υπολογιστών. Θα σας ζητηθεί να συμπληρώσετε ένα ερωτηματολόγιο.

Υπεύθυνοι αυτού του πειράματος είναι ο μεταπτυχιακός φοιτητής Ζώτος Αλέξανδρος και η επίκουρη καθηγήτρια Μανιά Κατερίνα. Αναμένεται να απασχοληθείτε το πολύ 15 λεπτά σε κάθε ενότητα. Θα χρησιμοποιήσουμε τα δεδομένα σας ανώνυμα μαζί με πολλά άλλα άλλων συμμετεχόντων.

Να θυμάστε ότι η συμμετοχή σας είναι **εθελοντική**. Μπορείτε να επιλέξετε να **μη συμμετάσχετε** σε κανένα ή σε οποιοδήποτε από τα επιμέρους στάδια του ερωτηματολογίου .

Παρακαλώ κυκλώστε τις απαντήσεις σας για τα παρακάτω θέματα:

Κατανοείτε τη φόρμα συγκατάθεσης;	Ναι	Όχι
Δίνετε τα δεδομένα σας για περαιτέρω ανάλυση στα πλαίσια της παρούσας εργασίας;	Ναι	Όχι
Επιβεβαιώνετε ότι είστε τουλάχιστον 18 χρονών;	Ναι	Όχι
Επιβεβαιώνετε ότι πότε στο παρελθόν δεν είχατε ενδείξεις ή συμπτώματα που να σχετίζονται με την νόσο της επιληψίας;	Ναι	Όχι

Υπογραφή_____   Ημερομηνία_____

Όνομα _____

Ενότητα Β: Προσωπικά στοιχεία

Παρακαλώ συμπληρώστε τα ακόλουθα κενά:

Όνομα: _____

Ηλικιακή ομάδα: 18-22   23-27   28-32   33-37   38-42   πάνω από 43

Φύλο: Αρσενικό / Θηλυκό

Ημερομηνία και ώρα: _____   _____

(Αν σπουδάζετε τώρα

παρακαλώ επιλέξτε ποια

είναι η βαθμίδα σας):   Προπτυχιακός   Μεταπτυχιακός   Διδακτορικός

Τρέχουσα κατάσταση: Φοιτητής   Ερευνητής

                        Ακαδημαϊκός   Άλλο (αναφέρατε) _____

Ενότητα Γ: Ερωτηματολόγιο μνήμης – Ακαδημαϊκός χώρος εργασίας

Παρακαλώ διαβάστε τα παρακάτω:

Σύμφωνα με την προηγούμενη σύντομη περιήγηση σας στον **ακαδημαϊκό χώρο εργασίας** του διαμερίσματος, θεωρείστε την παρακάτω λίστα με αντικείμενα. Κυκλώστε Ναι εάν πιστεύετε ότι το αντικείμενο βρισκόταν στο συγκεκριμένο χώρο και Όχι εάν πιστεύετε ότι το αντικείμενο απουσίαζε από τον συγκεκριμένο χώρο.

Επίσης, κυκλώστε έναν από τους αριθμούς που βρίσκονται δίπλα από κάθε απάντηση και αντιπροσωπεύει το βαθμό βεβαιότητας για την επιλογή σας, όπου 5 αντιπροσωπεύει ότι ήσασταν απόλυτα βέβαιος και 1 ότι δεν ήσασταν καθόλου βέβαιος για τη συγκεκριμένη επιλογή.

Απαντήστε τις ερωτήσεις με τη σειρά που σας παρουσιάζονται και σε καμία περίπτωση μην ανατρέξετε στις προηγούμενες απαντήσεις σας για να τις αλλάξετε.

Παρακαλώ κυκλώστε την απάντηση σας

	Ήταν το αντικείμενο παρών?		Πόσο **βέβαιοι** είστε για την απάντηση σας?				
			Καθόλου	Λίγο	Μέτρια	Αρκετά	Απόλυτα
Flash disk	Ναι Όχι		1	2	3	4	5
Web camera	Ναι Όχι		1	2	3	4	5
Ακουστικά	Ναι Όχι		1	2	3	4	5
Αφίσα/ πόστερ	Ναι Όχι		1	2	3	4	5
Βιβλία	Ναι Όχι		1	2	3	4	5
Γλάστρα	Ναι Όχι		1	2	3	4	5
Εκτυπωτής	Ναι Όχι		1	2	3	4	5
Ημερολόγιο	Ναι Όχι		1	2	3	4	5
Ηχεία	Ναι Όχι		1	2	3	4	5
Θήκη με CD	Ναι Όχι		1	2	3	4	5

Κάδος απορριμμάτων	Ναι	Όχι		1	2	3	4	5
Καλόγερος ρούχων	Ναι	Όχι		1	2	3	4	5
Καρέκλα γραφείου	Ναι	Όχι		1	2	3	4	5
Κλειδιά	Ναι	Όχι		1	2	3	4	5
Κούπα καφέ	Ναι	Όχι		1	2	3	4	5
Κουτί υπολογιστή	Ναι	Όχι		1	2	3	4	5
Κρεμάστρα	Ναι	Όχι		1	2	3	4	5
Λάπτοπ	Ναι	Όχι		1	2	3	4	5
Μολυβοθήκη	Ναι	Όχι		1	2	3	4	5
Οθόνη	Ναι	Όχι		1	2	3	4	5
Πίνακας (ζωγραφικής)	Ναι	Όχι		1	2	3	4	5
Πιστολάκι μαλλιών	Ναι	Όχι		1	2	3	4	5
Πληκτρολόγιο	Ναι	Όχι		1	2	3	4	5
Ποντίκι υπολογιστή	Ναι	Όχι		1	2	3	4	5
Πυροσβεστήρας	Ναι	Όχι		1	2	3	4	5
Ρολόι τοίχου	Ναι	Όχι		1	2	3	4	5
Σελοτέηπ	Ναι	Όχι		1	2	3	4	5
Σόμπα αλογόνου	Ναι	Όχι		1	2	3	4	5
Στυλό	Ναι	Όχι		1	2	3	4	5
Συρραπτικό	Ναι	Όχι		1	2	3	4	5
Ταμιακή μηχανή	Ναι	Όχι		1	2	3	4	5
Τηλέφωνο	Ναι	Όχι		1	2	3	4	5
Υπολογιστής τσέπης	Ναι	Όχι		1	2	3	4	5
Φάκελοι - Ντοσιέ	Ναι	Όχι		1	2	3	4	5
Φωτιστικό	Ναι	Όχι		1	2	3	4	5
Χάρακας	Ναι	Όχι		1	2	3	4	5
Ψαλίδι	Ναι	Όχι		1	2	3	4	5
Ψύκτης νερού	Ναι	Όχι		1	2	3	4	5

Ενότητα Δ: Ερωτηματολόγιο μνήμης - Σαλόνι

Παρακαλώ διαβάστε τα παρακάτω:

Σύμφωνα με την προηγούμενη σύντομη περιήγηση σας στο **σαλόνι** του διαμερίσματος, θεωρείστε την παρακάτω λίστα με αντικείμενα. Κυκλώστε Ναι εάν πιστεύετε ότι το αντικείμενο βρισκόταν στο συγκεκριμένο χώρο και Όχι εάν πιστεύετε ότι το αντικείμενο απουσίαζε από τον συγκεκριμένο χώρο.

Επίσης, κυκλώστε έναν από τους αριθμούς που βρίσκονται δίπλα από κάθε απάντηση και αντιπροσωπεύει το βαθμό βεβαιότητας για την επιλογή σας, όπου 5 αντιπροσωπεύει ότι ήσαστ−αν απόλυτα βέβαιος και 1 ότι δεν ήσασταν καθόλου βέβαιος για τη συγκεκριμένη επιλογή.

Απαντήστε τις ερωτήσεις με τη σειρά που σας παρουσιάζονται και σε καμία περίπτωση μην ανατρέξετε στις προηγούμενες απαντήσεις σας για να τις αλλάξετε.

Παρακαλώ κυκλώστε την απάντηση σας

	Ήταν το αντικείμενο παρών?		Πόσο **βέβαιοι** είστε για την απάντηση σας?				
			Καθόλου	Λίγο	Μέτρια	Αρκετά	Απόλυτα
Cds	Ναι Όχι		1	2	3	4	5
Ανθοδοχείο	Ναι Όχι		1	2	3	4	5
Βαρέλι	Ναι Όχι		1	2	3	4	5
Βιβλία	Ναι Όχι		1	2	3	4	5
Γλάστρα	Ναι Όχι		1	2	3	4	5
Γυάλινο δοχείο με ποτ πουρί	Ναι Όχι		1	2	3	4	5
Ηχεία	Ναι Όχι		1	2	3	4	5
Καλόγερος ρούχων	Ναι Όχι		1	2	3	4	5
Καλοριφέρ	Ναι Όχι		1	2	3	4	5
Κηροπήγιο	Ναι Όχι		1	2	3	4	5

Κινητό τηλέφωνο	Ναι	Όχι		1	2	3	4	5
Κλειδιά	Ναι	Όχι		1	2	3	4	5
Κορνίζα	Ναι	Όχι		1	2	3	4	5
Κράνος (Πολεμικό)	Ναι	Όχι		1	2	3	4	5
Λαμπατέρ	Ναι	Όχι		1	2	3	4	5
Μαξιλάρια	Ναι	Όχι		1	2	3	4	5
Μπάλα μπάσκετ	Ναι	Όχι		1	2	3	4	5
Μπουκάλι αλκοόλ	Ναι	Όχι		1	2	3	4	5
Ξυραφάκι	Ναι	Όχι		1	2	3	4	5
Οδοντόβουρτσα	Ναι	Όχι		1	2	3	4	5
Περιοδικό	Ναι	Όχι		1	2	3	4	5
Πίνακας	Ναι	Όχι		1	2	3	4	5
Πολυθρόνα	Ναι	Όχι		1	2	3	4	5
Ποτήρι	Ναι	Όχι		1	2	3	4	5
Ραδιοκασετόφωνο	Ναι	Όχι		1	2	3	4	5
Ρακέτα τένις	Ναι	Όχι		1	2	3	4	5
Σοκολατάκια	Ναι	Όχι		1	2	3	4	5
Σφυρί	Ναι	Όχι		1	2	3	4	5
Τηλεκοντρόλ	Ναι	Όχι		1	2	3	4	5
Τηλεόραση	Ναι	Όχι		1	2	3	4	5
Τσιγάρο	Ναι	Όχι		1	2	3	4	5
Φωτιστικό δαπέδου	Ναι	Όχι		1	2	3	4	5
Χαλί μοκέτα	Ναι	Όχι		1	2	3	4	5
Χειροβομβίδα	Ναι	Όχι		1	2	3	4	5

Ενότητα Ε: Ερωτηματολόγιο μνήμης - Κουζίνα

Παρακαλώ διαβάστε τα παρακάτω:

Σύμφωνα με την προηγούμενη σύντομη περιήγηση σας στη **κουζίνα** του διαμερίσματος, θεωρείστε την παρακάτω λίστα με αντικείμενα. Κυκλώστε Ναι εάν πιστεύετε ότι το αντικείμενο βρισκόταν στο συγκεκριμένο χώρο και Όχι εάν πιστεύετε ότι το αντικείμενο απουσίαζε από τον συγκεκριμένο χώρο.

Επίσης, κυκλώστε έναν από τους αριθμούς που βρίσκονται δίπλα από κάθε απάντηση και αντιπροσωπεύει το βαθμό βεβαιότητας για την επιλογή σας, όπου 5 αντιπροσωπεύει ότι ήσασταν απόλυτα βέβαιος και 1 ότι δεν ήσασταν καθόλου βέβαιος για τη συγκεκριμένη επιλογή.

Απαντήστε τις ερωτήσεις με τη σειρά που σας παρουσιάζονται και σε καμία περίπτωση μην ανατρέξετε στις προηγούμενες απαντήσεις σας για να τις αλλάξετε.

Παρακαλώ κυκλώστε την απάντηση σας

	Ήταν το αντικείμενο παρών?			Πόσο **βέβαιοι** είστε για την απάντηση σας?				
				Καθόλου	Λίγο	Μέτρια	Αρκετά	Απόλυτα
Αλατιέρα	Ναι	Όχι		1	2	3	4	5
Απορροφητήρας	Ναι	Όχι		1	2	3	4	5
Αποχυμωτής	Ναι	Όχι		1	2	3	4	5
Βιολί	Ναι	Όχι		1	2	3	4	5
Καρέκλες	Ναι	Όχι		1	2	3	4	5
Κατσαρόλα	Ναι	Όχι		1	2	3	4	5
Κονσέρβα	Ναι	Όχι		1	2	3	4	5
Κούπα	Ναι	Όχι		1	2	3	4	5
Κουτάλι	Ναι	Όχι		1	2	3	4	5

Κρουασάν	Ναι	Όχι		1	2	3	4	5
Μακαρόνια	Ναι	Όχι		1	2	3	4	5
Μίξερ	Ναι	Όχι		1	2	3	4	5
Μολυβοθήκη	Ναι	Όχι		1	2	3	4	5
Μπαστούνι Baseball	Ναι	Όχι		1	2	3	4	5
Μπρίκι	Ναι	Όχι		1	2	3	4	5
Νεροπίστολο	Ναι	Όχι		1	2	3	4	5
Οδοντόκρεμα	Ναι	Όχι		1	2	3	4	5
Πιάτο	Ναι	Όχι		1	2	3	4	5
Πιρούνι	Ναι	Όχι		1	2	3	4	5
Πιστολάκι μαλλιών	Ναι	Όχι		1	2	3	4	5
Ποτήρι	Ναι	Όχι		1	2	3	4	5
Ρόδα (ποδηλάτου/μηχανής)	Ναι	Όχι		1	2	3	4	5
Σούπα	Ναι	Όχι		1	2	3	4	5
Σουρωτήρι	Ναι	Όχι		1	2	3	4	5
Σύριγγα	Ναι	Όχι		1	2	3	4	5
Τηγάνι	Ναι	Όχι		1	2	3	4	5
Τρομπέτα	Ναι	Όχι		1	2	3	4	5
Τσαγιέρα	Ναι	Όχι		1	2	3	4	5
Υπολογιστής Τσέπης	Ναι	Όχι		1	2	3	4	5
Φούρνος Μικροκυμάτων	Ναι	Όχι		1	2	3	4	5
Φρούτα	Ναι	Όχι		1	2	3	4	5
Φρυγανιέρα/ Τοστιέρα	Ναι	Όχι		1	2	3	4	5
Φτυάρι	Ναι	Όχι		1	2	3	4	5
Χαρταετός	Ναι	Όχι		1	2	3	4	5
Ψυγείο	Ναι	Όχι		1	2	3	4	5
Ψωμί	Ναι	Όχι		1	2	3	4	5

Υποενότητα : Ερωτηματολόγιο μνήμης - Γραφείο

Παρακαλώ διαβάστε τα παρακάτω:

Σύμφωνα με την προηγούμενη σύντομη περιήγηση σας στον **ακαδημαϊκό χώρο εργασίας** του διαμερίσματος, θεωρείστε την παρακάτω λίστα με αντικείμενα. Κυκλώστε <u>Ναι</u> εάν πιστεύετε ότι το αντικείμενο βρισκόταν πάνω στο **ΓΡΑΦΕΙΟ** του συγκεκριμένο χώρου και <u>Όχι</u> εάν πιστεύετε ότι το αντικείμενο απουσίαζε από τη συγκεκριμένη περιοχή.

Επίσης, κυκλώστε <u>έναν</u> από τους αριθμούς που βρίσκονται δίπλα από κάθε απάντηση και αντιπροσωπεύει το βαθμό βεβαιότητας για την επιλογή σας, όπου 5 αντιπροσωπεύει ότι ήσασταν <u>απόλυτα βέβαιος</u> και 1 ότι δεν ήσασταν <u>καθόλου βέβαιος</u> για τη συγκεκριμένη επιλογή.

Απαντήστε τις ερωτήσεις με τη σειρά που σας παρουσιάζονται και σε καμία περίπτωση μην ανατρέξετε στις απαντήσεις σας για να τις αλλάξετε.

Παρακαλώ κυκλώστε την απάντηση σας

	Ήταν το αντικείμενο παρών?		Πόσο **βέβαιοι** είστε για την απάντηση σας?				
			Καθόλου	Λίγο	Μέτρια	Αρκετά	Απόλυτα
Flash disk	Ναι	Όχι	1	2	3	4	5
Web camera	Ναι	Όχι	1	2	3	4	5
Ακουστικά	Ναι	Όχι	1	2	3	4	5
Γλάστρα	Ναι	Όχι	1	2	3	4	5
Θήκη με CD	Ναι	Όχι	1	2	3	4	5
Κάδος απορριμμάτων	Ναι	Όχι	1	2	3	4	5
Κλειδιά	Ναι	Όχι	1	2	3	4	5
Κουτί υπολογιστή	Ναι	Όχι	1	2	3	4	5
Κρεμάστρα	Ναι	Όχι	1	2	3	4	5
Οθόνη	Ναι	Όχι	1	2	3	4	5

Πίνακας (ζωγραφικής)	Ναι	Όχι		1	2	3	4	5
Πληκτρολόγιο	Ναι	Όχι		1	2	3	4	5
Πυροσβεστήρας	Ναι	Όχι		1	2	3	4	5
Ρολόι τοίχου	Ναι	Όχι		1	2	3	4	5
Σελοτέηπ	Ναι	Όχι		1	2	3	4	5
Σόμπα αλογόνου	Ναι	Όχι		1	2	3	4	5
Φωτιστικό	Ναι	Όχι		1	2	3	4	5
Ψύκτης νερού	Ναι	Όχι		1	2	3	4	5

Υποενότητα : Ερωτηματολόγιο μνήμης - Βιβλιοθήκη

Παρακαλώ διαβάστε τα παρακάτω:

Σύμφωνα με την προηγούμενη σύντομη περιήγηση σας στον **ακαδημαϊκό χώρο εργασίας** του διαμερίσματος, θεωρείστε την παρακάτω λίστα με αντικείμενα. Κυκλώστε <u>Ναι</u> εάν πιστεύετε ότι το αντικείμενο βρισκόταν **στη ΒΙΒΛΙΟΘΗΚΗ** του συγκεκριμένο χώρου και <u>Όχι</u> εάν πιστεύετε ότι το αντικείμενο απουσίαζε από τη συγκεκριμένη περιοχή.

Επίσης, κυκλώστε <u>έναν</u> από τους αριθμούς που βρίσκονται δίπλα από κάθε απάντηση και αντιπροσωπεύει το βαθμό βεβαιότητας για την επιλογή σας, όπου 5 αντιπροσωπεύει ότι ήσασταν <u>απόλυτα βέβαιος</u> και 1 ότι δεν ήσασταν <u>καθόλου βέβαιος</u> για τη συγκεκριμένη επιλογή.

Απαντήστε τις ερωτήσεις με τη σειρά που σας παρουσιάζονται και σε καμία περίπτωση μην ανατρέξετε στις απαντήσεις σας για να τις αλλάξετε.

Παρακαλώ κυκλώστε την απάντηση σας

	Ήταν το αντικείμενο παρών?		Πόσο **βέβαιοι** είστε για την απάντηση σας?				
			Καθόλου	Λίγο	Μέτρια	Αρκετά	Απόλυτα
Web camera	Ναι Όχι		1	2	3	4	5
Ακουστικά	Ναι Όχι		1	2	3	4	5
Βιβλία	Ναι Όχι		1	2	3	4	5
Εκτυπωτής	Ναι Όχι		1	2	3	4	5
Κούπα καφέ	Ναι Όχι		1	2	3	4	5
Λάπτοπ	Ναι Όχι		1	2	3	4	5
Μολυβοθήκη	Ναι Όχι		1	2	3	4	5
Πληκτρολόγιο	Ναι Όχι		1	2	3	4	5
Ποντίκι υπολογιστή	Ναι Όχι		1	2	3	4	5
Συρραπτικό	Ναι Όχι		1	2	3	4	5

Τηλέφωνο	Ναι	Όχι		1	2	3	4	5
Υπολογιστής τσέπης	Ναι	Όχι		1	2	3	4	5
Φάκελοι - Ντοσιέ	Ναι	Όχι		1	2	3	4	5
Ψαλίδι	Ναι	Όχι		1	2	3	4	5

Υποενότητα : Ερωτηματολόγιο μνήμης - Καναπέδες

Παρακαλώ διαβάστε τα παρακάτω:

Σύμφωνα με την προηγούμενη σύντομη περιήγηση σας στο **σαλόνι** του διαμερίσματος, θεωρείστε την παρακάτω λίστα με αντικείμενα. Κυκλώστε <u>Ναι</u> εάν πιστεύετε ότι το αντικείμενο βρισκόταν πάνω στους **ΚΑΝΑΠΕΔΕΣ** του συγκεκριμένου χώρου και <u>Όχι</u> εάν πιστεύετε ότι το αντικείμενο απουσίαζε από την συγκεκριμένη περιοχή.

Επίσης, κυκλώστε <u>έναν</u> από τους αριθμούς που βρίσκονται δίπλα από κάθε απάντηση και αντιπροσωπεύει το βαθμό βεβαιότητας για την επιλογή σας, όπου 5 αντιπροσωπεύει ότι ήσασταν <u>απόλυτα βέβαιος</u> και 1 ότι δεν ήσασταν <u>καθόλου βέβαιος</u> για τη συγκεκριμένη επιλογή.

Απαντήστε τις ερωτήσεις με τη σειρά που σας παρουσιάζονται και σε καμία περίπτωση μην ανατρέξετε στις απαντήσεις σας για να τις αλλάξετε.

Παρακαλώ κυκλώστε την απάντηση σας

	Ήταν το αντικείμενο παρών?		Πόσο **βέβαιοι** είστε για την απάντηση σας?				
			Καθόλου	Λίγο	Μέτρια	Αρκετά	Απόλυτα
Cds	Ναι Όχι		1	2	3	4	5
Αναπτήρας	Ναι Όχι		1	2	3	4	5
Ανθοδοχείο	Ναι Όχι		1	2	3	4	5
Βιβλία	Ναι Όχι		1	2	3	4	5
Γυάλινο δοχείο με ποτ πουρί	Ναι Όχι		1	2	3	4	5
Κηροπήγιο	Ναι Όχι		1	2	3	4	5
Κινητό τηλέφωνο	Ναι Όχι		1	2	3	4	5
Κλειδιά	Ναι Όχι		1	2	3	4	5
Κορνίζα	Ναι Όχι		1	2	3	4	5

Μπουφάν - Ζακέτα	Ναι	Όχι		1	2	3	4	5
Λαμπατέρ	Ναι	Όχι		1	2	3	4	5
Μπουκάλι αλκοόλ	Ναι	Όχι		1	2	3	4	5
Περιοδικό	Ναι	Όχι		1	2	3	4	5
Ποτήρι	Ναι	Όχι		1	2	3	4	5
Σοκολατάκια	Ναι	Όχι		1	2	3	4	5
Τηλεκοντρόλ	Ναι	Όχι		1	2	3	4	5

Ενότητα ΣΤ:

Οι παρακάτω ερωτήσεις αφορούν την εμπειρία σας από την περιήγηση σας στην τρισδιάστατη σκηνή. Παρακαλώ θυμηθείτε τι ή πως νιώσατε και απαντήστε τα παρακάτω. Βαθμολογήστε από 1 έως 5 τη κάθε ερώτηση

Παράδειγμα:

Πως αισθάνεσαι τώρα?

Ζαλισμένος   *1*   *2*   *3*   *4*   *5*   *Μια χαρά*

Αν κυκλώσετε απαντήσεις που είναι κοντά στο Ζαλισμένος σημαίνει ότι δε νιώθετε και πολύ καλά.

1. Παρακαλώ βαθμολογήστε την αίσθηση του να είστε στο τρισδιάστατο χώρο όπου το 5 αντιπροσωπεύει τη φυσιολογική σας εμπειρία στο να βρίσκετε σε έναν ανάλογο πραγματικό χώρο

   Καθόλου   1   2   3   4   5   Πάρα πολύ

2. Σε τι βαθμό υπήρχαν στιγμές που νιώσατε ότι το δωμάτιο ήταν πραγματικό για σας?

   Καθόλου   1   2   3   4   5   Τις περισσότερες

3. Όταν σκέφτεστε το δωμάτιο που πλοηγηθήκατε το θυμάστε ως εικόνες ή ως κάποιο χώρο που επισκεφτήκατε?

   Εικόνες   1   2   3   4   5   Ένα μέρος που επισκέφτηκα

4. Έχετε προηγούμενη εμπειρία από τρισδιάστατες εικονικές εφαρμογές/ παιχνίδια?

   Πότε   1   2   3   4   5   Πολύ συχνά

5. Κατά τη διάρκεια της περιήγησης, ποιο ήταν το δυνατότερο συναίσθημα, ότι βρίσκεστε στο

τρισδιάστατο δωμάτιο ή ότι βρίσκετε κάπου αλλού;

Αλλού	1	2	3	4	5	Στο δωμάτιο

6. Σε τι βαθμό χρησιμοποιείτε υπολογιστή καθημερινά;

Καθόλου	1	2	3	4	5	Πάρα πολύ

7. Κατά τη διάρκεια της πλοήγησης, σε τι βαθμό πιστεύατε ότι βρισκόσασταν στο τρισδιάστατο δωμάτιο;

Όχι πολύ συχνά	1	2	3	4	5	Πολύ συχνά

Ενότητα Ζ: Εντυπώσεις από την περιήγηση σας

Οι ακόλουθες ερωτήσεις αφορούν τις εντυπώσεις σας από τη περιήγηση σας στο τρισδιάστατο διαμέρισμα.
Παρακαλώ κυκλώστε την καταλληλότερη βαθμολογία από 1 έως 7 για κάθε μία ερώτηση.

στενάχωρο	1	2	3	4	5	6	7	**ευρύχωρο**
σκοτεινό	1	2	3	4	5	6	7	**φωτεινό**
αδιάφορο	1	2	3	4	5	6	7	**ενδιαφέρον**
μελαγχολικό	1	2	3	4	5	6	7	**ζωηρό**
πολύπλοκο	1	2	3	4	5	6	7	**απλό**
κρύο	1	2	3	4	5	6	7	**ζεστό**
δυσάρεστο	1	2	3	4	5	6	7	**ευχάριστο**
άβολο	1	2	3	4	5	6	7	**άνετο**

μικρό	1	2	3	4	5	6	7	μεγάλο
αντιπαθητικό	1	2	3	4	5	6	7	συμπαθητικό
ακατάστατο	1	2	3	4	5	6	7	τακτοποιημένο

Ενότητα Η:

Παρακαλώ κυκλώστε από το 1 έως το 5 την πιο κατάλληλη απάντηση

Περιμένατε ότι θα σας ζητηθεί να απαντήσετε σε ερωτηματολόγιο μνήμης για τη <u>συγκεκριμένη φάση</u> εξάσκησης του πειράματος ?

    Το περίμενα   1   2   3   4   5   Δεν το περίμενα

Παρατηρήσατε κατά της διάρκεια της πλοήγησης να αλλάζει οποιοδήποτε αντικείμενο της σκηνής?

  Ναι παρατήρησα   Όχι δεν παρατήρησα

Αν στη παραπάνω ερώτηση απαντήσατε ναι, παρακαλώ στο διαθέσιμο χώρο που βρίσκεται στο τέλος του ερωτηματολογίου περιγράψτε τις αλλαγές που παρατηρήσατε.

Παρακαλώ κυκλώστε την απάντηση

Ήταν δύσκολο να συμπληρώσετε το ερωτηματολόγιο?   Ναι   Όχι

Σας πήρε πολύ ώρα?   Ναι   Όχι

Το βρήκατε κουραστικό?   Ναι   Όχι

Το βρήκατε προσβλητικό?   Ναι   Όχι

Αν έχετε απαντήσει <u>Ναι</u> σε κάποια από τις παραπάνω ερωτήσεις ίσως θέλετε να χρησιμοποιήσετε το παρακάτω χώρο για διάφορα σχόλια που έχετε να κάνετε.

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____