Diminished Reality Designed with Interactive Cutaway Opening for Spatial Perception Improvement

Chaoyi Li

A Dissertation

Presented to the University of Dublin, Trinity College

in partial fulfilment of the requirements for the degree of

Master of Science in Computer Science

Supervisor: John Dingliana

August 2021

Declaration

I, the undersigned, declare that this work has not previously been submitted as an exercise for a degree at this, or any other University, and that unless otherwise stated, is my own work.

Chaoyi Li

August 31, 2021

Permission to Lend and/or Copy

I, the undersigned, agree that Trinity College Library may lend or copy this thesis upon request.

Chaoyi Li

August 31, 2021

Diminished Reality Designed with Interactive Cutaway Opening for Spatial Perception Improvement

Chaoyi Li, Master of Science in Computer Science
University of Dublin, Trinity College, 2021

Supervisor: John Dingliana

Diminished Reality refers to methodologies that change the appearance of real-world objects. This thesis reviews literature and clarifies concepts in Augmented Reality, DR, and spatial perception issues for these systems. It proposes an innovative interaction of cutaway for DR that allows for user-defined cutaway opening and adaptive cavity generation. Evaluation on the demo system has proven its usability and states that it improves spatial perception compared to conventional methods in DR.

Acknowledgments

The author hereby shows his sincere appreciation towards his supervisor, Prof. John Dingliana. He has shown great understanding and support to the author's predicament. It is his concrete and enlightening suggestions step by step along the way that have improved this research significantly.

CHAOYI LI

University of Dublin, Trinity College
August 2021

Contents

List of Figures

Chapter 1

Introduction

This section contains the motivation of a newly proposed cutaway technique for Diminished Reality (DR); the objective of the development and evaluation; contribution to current state-of-the-art DR applications; and finally an outline of the dissertation.

1.1 Motivation

Augmented Reality or AR has been one of the most popular research topics over recent years. It enhances human's vision further than traditional screens do by combining the real-world and digital information. Its potential application in traditional industries such as manufacturing, repairing, education, and medicine can create revolutionary waves by offering efficient human-computer interaction and remote work opportunities.

However, when we look at the top charts of mobile apps, many are video-based, such as YouTube and Tik-tok; none is AR-based. The lack of a killer app reflects some deep challenges faced by AR. We have not tapped the full potential of AR due to technological difficulties on both software and hardware fronts. Among all the issues the author will review in the next chapter, Background and State of the Art, spatial perception in AR stands out as a typical barrier to a better and more usable AR application.

Perceptual Issues can affect the task performance of all Mixed Reality systemsDrascic and Milgram (1996); spatial perception is among the most important issues. The cutaway technique offers an effective solution by taking advantage of the occlusion cue. In effect, it renders not only the object of interest but also its surrounding. However, the state-of-the-art cutaway is limited in its customizability. Hence, the author proposes a more customizable and adaptive cutaway technique for better spatial perception.

This thesis will examine its effects on monitor-based AR applications. The author chooses monitor-based AR due to the following reasons. First, the possible improvement on spatial perception would be independent of hardware since AR developing platform such as ARCore and ARKit ensure consistent performance; second, these monitor-based AR platforms and the applications based on those are more accessible and popular than see-through devices.

1.2 Objective

Primarily, the objective is to develop an interactive and adaptive cutaway technique to improve the spatial perception in Diminished Reality. For the best visual results, this technique, however, will be accompanied by other techniques, some are established and proven; others rather more theoretical and untested. Therefore, secondarily, the objective includes the evaluation of all these techniques regarding its visual results and/or computing performance.

1.3 Contribution

Primarily, the author has developed an interactive and adaptive solution to present rendered objects inside real-world objects; the main techniques used include: A.Interactively user-defined cutaway opening shape and depth; a major contribution/novelty is an algorithm that infers the hole opening shape from closed lines drawn by the user in 3D for AR purposes. B.Optimization of wall/cutting direction. This direction is automatically optimized for the best viewing angle of the inner object from the hole opening. C.Color transfer. Rendering the walls of the hole with colors sampled from user-defined vertices; also they are tinted for better depth perception. Though color has long been listed as an important depth cue, its theoretical effects are rarely tested. This thesis has applied and evaluated relevant coloring theories for AR or DR purposes. D.Ambient lighting for better visibility of the inside of the wall; combined with directional lighting and shadows for space perception. E.Tinted the inner object with contrast color for better focus and visibility.

Secondarily, it is evaluating the performance of the algorithm mentioned in A and B; and evaluating the impact of all the techniques above on spatial perception.

1.4 Outline

The thesis consists of the following six chapters:

- Chapter 2 is literature review on Mixed Reality systems; Augmented Reality and its state of the art; and the emerging Diminished Reality. It will introduce relevant DR applications and techniques. This chapter also reviews perceptual issues of these systems and visual cues that pertain to their improvement.

- Chapter 3 is the design of a new cutaway interaction where the author elaborates on the systems components: user-drawn cutaway opening, user-defined cutaway depth, adaptive cutaway direction, adaptive hole color with color transfer, and other rendering technique applied for visual results. Specifically, algorithm design will be the main topic in this chapter.
- Chapter 4 is the implementation of the experiment system, which includes hardware and software specifications; preprocessing required; and data structure for algorithms.
- Chapter 5 is the evaluation on the visual and/or computing performance of major system components.
- Chapter 6 is conclusion on the research objective and future work to further improve the adaptibility and performance of the proposed set of techniques.

Chapter 2

Background and State of the Art

This chapter reviews the background needed for the thesis. It begins with an introduction to the Augmented-Virtuality Continuum. Relevant concepts, such as Augmented Reality, Virtual Reality, and Diminished Reality will be discussed. This will also include the state-of-the-art platform solutions and applications. The chapter, then, continues to elaborate on perceptual issues in AR, especially those that pertain to spatial perception. This includes important visual cues and their influence.

2.1 Reality-Virtuality Continuum

2.1.1 Augmented and Virtual Reality

MilgramMilgram et al. (1995) proposed the concept of Reality-Virtuality continuum; in his taxonomy, different names are given based on the ratio of reality and computer-generated graphics in the final presenting. On the very left of the spectrum is total reality without computer-generated graphics; and on the very right is total virtuality without any real world scene being seen. Mixed Reality is then defined as anywhere between the two extremes. It includes the commonly used term Augmented Reality (AR) where the majority of what the user sees is through direct view or stereo video and part of the user sees is computer-generated graphics; and correspondingly, Augmented Virtuality (AV) where the majority is computer-generated and minority is reality. See Figure 2.1

It is worth noting that while here we focus on the graphical aspect of Mixed Reality systems, they have advanced to include more than what we see, but also sound, such as an early system Lyons et al. (2000) and the latest application Nagele et al. (2021); hapticsHayward et al. (2004) and other modalities.

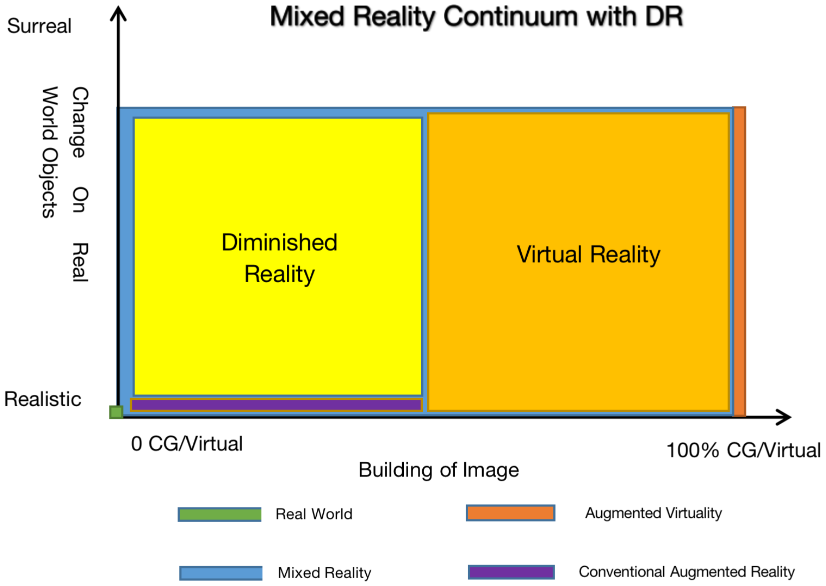


Figure 2.1: Mixed Reality Continnum with DR

2.1.2 Devices

Regarding on the video device used, AR systems can be based on monitors; head-mounted devices; and large immersive AV systems. Regarding monitor-based AR systems (Figure 2.4Azuma (1997)), PC monitors are much less frequently used than mobile devices, e.g. mobile phones since the latter have better mobility and built-in cameras. Head-mounted devices (HMDs) for AR are mostly immersive, incorporating optical or video see-through, see figure2.3Azuma (1997). The former can be classified as direct AR with computer graphics (CG) overlaid upon optical see-through vision; the latter is indirect AR with CG overlaid upon stereo-monitors. Direct see-through systems allow for perfectly accurate vision of the real-world because they do not change the light coming into eyes from the environment. For the same reason, they have difficulties in the composition stage as explained in the section Difficulties in DR. Indirect see-through systems have more flexibility with presenting the real environment. However, since they have to capture and display more content (real-world environment) than the indirect, this type of systems suffer more of difficulties that originate from capture and display as discussed in Spatial Perception/Sources of Problems. Projector-camera systems are newer to this family. Like direct see-through systems, the user can see real-world objects directly; but the CG is projected/overlaid onto real-world objects to realize the mix of real and virtual content. A camera is used to track real-world interactions or ”canvas” movement. This type of systems aim at a different working scenario: allowing multiple users to enjoy the MR

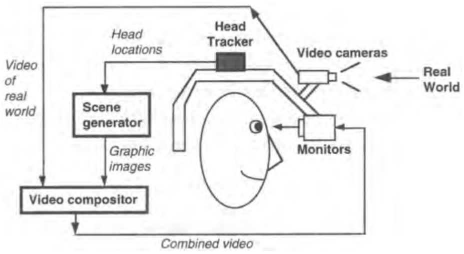


Figure 2.2: Video see-through AR configuration

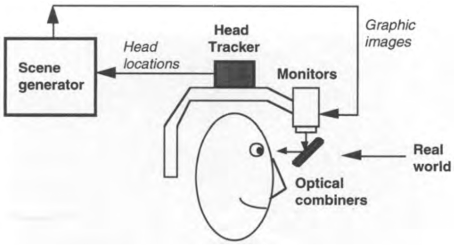


Figure 2.3: Optical see-through AR configuration

experience. See figure2.5Lee et al. (2015).

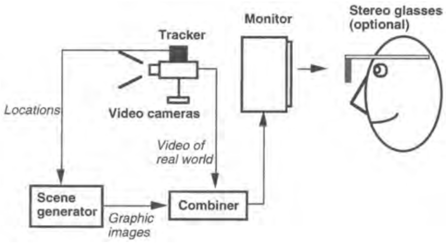


Figure 2.4: Monitor-based AR configuration

2.1.3 Principles and Platform Solutions

The three basic components in all AR applications are projection, calibration, and reconstruction. It is summarized in the equation: p = C * P where p is projection; C is camera calibration; and P is model pose. A simple AR application would be projecting a known 3D model into the real world, namely solving for p with C and P given. In effect, C is often known from system knowledge of hardware and software specifications while pose requires some detection and estimation. The pose is essentially the position of the model to be rendered, its orientation, and its size, corresponding to the translation, rotation, and scaling of the model. One way to estimate the pose is using predefined markers, which can be a picture, namely a matrix of color values. With algorithms such as SIFTLowe

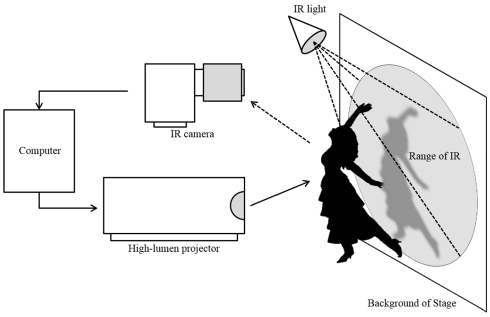


Figure 2.5: Projection-based AR configuration

(1999) and SURFBay et al. (2008) , the system can extract features from the matrix and compare them with new images based on feature vectors. Pose estimation, then, can be done based on the difference between the pre-defined marker and the one detected.

It is worth noting that the marker can also be a 3D object. An AR system is able to detect and track the features of a real-world object, instead of artificially printed image, in order to do pose estimation. The features can be corners and edges that are easily detected through image processing. For systems like Vuforia, sharp corners are recognizable features compared with rounded edges; and tracking improves with the number of these features.Siti Sendari (2020) In order to detect and track the marker, the AR system has to assume the spatial co-relation of the features is unchanged throughout the process. This, thus, means the 3D marker should be geometrically rigid (having few moving parts, not deformable) and have stable surface (not too shinny to obscure contrast features)Vuf (a). These features should be fed to the AR system beforehand through camera scanning or the analysis of its 3D mesh.

2D marker can be intrusive to the scene. While 3D markers overcome this drawback, they have to be known by the system beforehand: it limits the adapatibility of the system. Markerless AR systems overcome this entirely by extracting features from images and tracking them through the sequence of images. With proper hardware support (e.g. mobile phones), feature-based tracking can also be fused with inertial measurementHol et al. (2007). Markerless AR allows for AR's application in a broader area, including live broadcasting, production, maintenance, architecture, design, and tourismMichael Felsberg (2007).

Since 1968 when the first AR equipment was developed at Harvard, AR has seen remarkable advancement in visual results and ease of use. Commercial AR platforms have contributed to it in recent years by packaging the implementation of the above basic components: camera calibration, pose estimation through feature extraction and tracking, along with rendering. Besides the basics, ARCore from Google offers motion tracking, environmental understanding, depth understanding, and light estimationARC.

It is a across-the-board solution supporting Android, iOS, Unity, and Unreal. Its counterpart, ARKit from Apple provides even more sophisticated depth detection with iPhones' LiDAR. This new hardware also allows for faster plane findingARK. ARKit features its face tracking, with which pose estimation can be done based on features on human faces. Vuforia is another cross-platform AR engine. It supports the recognition and tracking of cylindrical images, such as coke cans; it allows for the offline scanning of real-world objects for feature extraction before online tracking. The scanning can also be replaced by the input of 3D mesh of objects. With the latest development, the size of the object to be scanned is largely expanded to include room-size environmentVuf (b).

2.1.4 Mediated Reality and Diminished Reality

Looking at the ratio of real-world objects and computer-generated graphics, one may be satisfied with the one-dimension of reality-virtuality spectrum. However, if one considers the function of all the computer graphics, a new dimension emerges. Computer graphics can conceal, highlight, diminish, augment, and alter what human eyes naturally see. Consider an apple on the table. Concealing means entirely wiping out the apple from vision; highlighting could be increasing its brightness or lowering the brightness of the table; diminishing could be rendering the apple translucent or half-bitten; and augmenting may be showing its nutrition information on the side. This function-wise approach leads to the concept of Mediated RealityMann (1994). The term AR, as we defined above in the reality-virtuality continuum, is with respect to the ratio of virtual objects. However, despite some confusion, the author points out here it is also associated with the functional meaning as in Mediated Reality.

Diminished Reality (DR), on the other hand, is a less used and more clearly defined term. Originating from the function-wise approach that gives birth to Mediated Reality, DR refers to the set of techniques that diminish real-world objects from vision through disappearance, translucency, cutaway, etc. Although etymology does not support it, the word meaning of "diminish" in DR has expanded over the years to include wiping out, distorting, and partially or entirely replacing. Different from AR that primarily adds to the real-world scene, DR focuses more on the interaction between computer graphics and reality. The next subsection will elaborate on these various interactions. See DR's integretion into the Mixed Reality Conitinuum in figure 2.1.

2.1.5 Basic Functions of DR

A comprehensive review on DR in 2017Shohei Mori (2017) summarizes the basic functions and usage of DR as diminish (narrowly referring to degrading visual function), see-through, replace, and inpaint. However, we believe this taxonomy is confusing because it mixes the techniques and functions in DR and fails to put everything in the same dimension, thereby causing unnecessary overlapping. For example, Replace is a purely functional description of DR, including any technique that visually hide the real object and show the virtual one. The hiding step can be realized with either see-through or inpaint techniques.

Hence, we propose a new taxonomy purely based on the visual functions of DR, or in other words, its interaction with the real environment. The categories are clearly defined here without confusing overlapping to call for rigorous use of terminology in literature.

Weaken

Weaken is the application of DR to weaken the visual importance of real-world objects. Its aim is to divert the user's attention from the object or to highlight others. Its visual effects are surreal (unusual for naked eyes) but functional. A simple visual effect can be rendering the object of interest normally but reduces the color saturation and/or brightness of everything else in the real-world environment. This clearly helps to focus the user's attention on the object of interest. We can image this type of DR to be applied in driving assistant systems to highlight signposts or traffic lights. Mann also proposed that it can be used to help the visually impairedMann (1994). See figure 2.6

Figure 2.6: Weaken the surrounding, illustration

Conceal

Conceal is employing DR to partially or entirely conceal the appearance of real objects. It aims to create illusions that the occluding object is non-existent or semi-transparent. The visual results can be realistic or surreal depending on task objectives. Both Weaken and Conceal can mean reduce the visual importance. However, note the former recognizes the existence of the object to be "diminished"; while the latter tries to deprives all its existence or leaves a transparent ghost for spatial perception only. Experiments have shown that concealing real-world objects may improve human perception through reducing distraction and increasing focus on the AR-rendered object of interestKim et al. (2021). Concealing can also be used to remove people from videos for privacy protectionYagi et al. (2017). If the object is concealed in the screen space by generating plausible pixels surrounding it, this process is also known as inpaintHerling and Broll (2010). Besides, this function reveals the object occluded. Mori's survey in 2017 mentioned some typical applications: for a driver to see through car interiorYoshida et al. (2008); and to see through wallsAvery et al. (2009) Barnum et al. (2009). This type of applications are sometimes called see-through. Nevertheless, many new and more sophisticated applications have emerged: to conceal landscape in frontKido et al. (2020); to reveal an entirely occluded object in the back in real-timeKunert et al. (2019); to remove the occlusion caused by the robot in telemanipulationTaylor et al. (2020).

DR applications concentrate on this function. Among them, medical area is one of the most tempted. Bichlmeier et al. (2007) manipulates the transparency of video images recorded by indirect HMDs and overlay the virtual medical imaging onto it. As a step to anatomy education efforts, Ienaga et al. (2016) tries to conceal the human body parts by projecting images recovered by an RGB-D camera.

Cutaway

Cutaway is a class of DR that renders the images of cutting into the object and revealing its inside. Compared with the above two approaches, Cutaway tries to attract attention to the object to be "diminished" itself as opposed to distracting attention from it. It focuses on the interaction between the real and the virtual objects. Therefore, Cutaway requires knowledge of the object to be "diminished". For example, what the dissection surface and the interior look like.

Cutaway is a visual function more commonly used in traditional computer graphics. Cutaway illustration has been an important educational tool long before computers were invented. Literature in computer graphics has summarized some of its cutting conventions as object-aligned box cuts, tube cuts, and window cutsLi et al. (2007). This system allows for a process to adjust cutaway parameters, such as occlusions, before the automatic presentation. The system has successfully reached a balance between ease of use and clarity of presentation in presenting 3D models with complex interior structures.

Regarding cutting techniques that allows for more flexible user control, several sketch-based 3D modelling systems have been proposed in earlier literatureOwada et al. (2006)IGARASHI et al. (1999). These systems allows for the presentation and sketching of 2D profiles. An inflation algorithm is responsible for inferring 3D models from 2D polygons. Because of this, cutting as part of the systems, accounts for planar cutting surface only. An inclusive survey of general mesh-cutting was presented in 2002Bruyns et al. (2002) where mesh-cutting techniques are discussed in the following dimensions: definition of cut path, primitive removal and re-meshing; number of new primitives or primitives created; when re-meshing is performed; and representation of the tool. When sophisticated 3D meshes with volume, these generalized techniques can be particularly useful. However, when the model is surface meshes without volume, there will not be any dissection surface created. A extrusion-based method was propose to solve thisCoffin and Hollerer (2006) though its cutting surface is planar or has to be defined by a rotational sweep.

Compared to literature in 3D rendering, there are fewer applications of cutaway in DR. MendezMendez and Schmalstieg (2009) proposed to use an importance mask to expose the occluded object inside a box for improved spatial perception. The system cuts out holes of different shapes and sizes to compare spatial perception. Besides work based on meshes, there are also efforts to present volumetric data with DRKutter et al. (2008). However, without a plausible contact surface between the real and the virtual, e.g. a dissection surface, the rendering result looks more like x-ray vision in Conceal DR. A medical example of this is Wimmer et al. (2008) where rectangular and oval windows are created to reveal the virtual anatomy model inside.

2.1.6 Difficulties in DR

Currently, major difficulties in DR can be classified into: background observation and generation; and region of interest detectionShohei Mori (2017).

Background observation and generation

When performing a Conceal task, a DR system needs to render the scene occluded by an object in front. This occluded scene can be based on observation, if possible. or when it is not, conjecture, i.e. inpaint techniques that fill the occluded pixels with surrounding information. Observation can be done before the online DR process. As in Mori et al. (2015), DR Conceal is regarded as a process of regaining the previous scene with the current viewpoint. Before the online DR process, a camera (can be RGB-D cameras as in Meerits and Saito (2015)Sugimoto et al. (2014))is used to capture the scene with and without the object to be concealed for 3D reconstruction. Or observation can go simultaneously with reconstruction and render the occluded object on the fly as in Kunert et al. (2019). Alternatively, the occluded can be observed from a second camera in-real time. Though it is faced with the calibration of a new camera and complex composition, this approach presents the most authentic occluded scene. It is often used in X-ray vision applications, such asAvery et al. (2009)Barnum et al. (2009).

Besides 3D reconstruction, the hidden view can also be generated through the homography images from other camerasJarusirisawad and Saito (2007a)Barnum et al. (2009). Another approach is image-based as in early effortsDebevec et al. (1996); more examples are listed here Shohei Mori (2017).

Region of Interest Detection

The determination of which pixels to diminish is called region of interest detection (ROI detection). This process can be skipped if the replacement image can be entirely overlaidJarusirisawad and Saito (2007b). Otherwise, ROI detection can be done manually through labelling; or it can be processed through computer vision algorithms. There are traditional featured-based Haar-like solutionsViola and Jones (2001) and emerging convolutional neural network-based onesRedmon et al. (2016)Liu et al. (2016). These new advancement have supported latest applications with semantics-based ROI detectionNakajima et al. (2017).

In some scenarios, the geometry of object to be diminished is known beforehand (input as 3D meshes or contructed through Structure from Motion algorithmsWestoby et al. (2012). Then ROI can be determined by projecting the model to the camera perspective.

2.2 Spatial Perception in AR/DR

Spatial perception in Augmented Reality refers to the perceived position, size, and shape of an object in AR. A summaryKruijff et al. (2010) of all problems in AR/DR has found out three categories of perceptual issues: scene distortion and abstraction, depth distortions and object ordering, and visibility; and that depth perception is the most common issue. According to GregoryGregory (1973) , perception of human eyes is not solely determined by the physics of light; but rather an interpretation of all the visual cues. Drascic and colleaguesDrascic and Milgram (1996) further point out that if there is an agreement among the visual cues, they will add to the accuracy of depth perception; otherwise, to confusion. Therefore, a discussion on all depth cues in AR/DR system is appropriate.

2.2.1 Depth Cues

As classified by Drascic et al.Drascic and Milgram (1996), perceptual cues can present in static pictures(interposition, linear perspective, texture perspective, aerial perspective, relative brightness, shadows); in motion(relative motion parallax, motion perspective, and the kinetic depth effect); in eyeball movement(convergence, accommodation); and in vision difference between two eyes. However, another category is often neglected: psychological cues. Adults have learned the size of common objects in life, e.g. an apple is usually 10cm across while a car can be 500cm long. An effort to render a car-size apple beside a car would naturally make the apple look much closer to the camera/observer than the car. Nevertheless, these are fringe cases that will have the most impact on the depth perception for AR/DR. Among a variety of depth cues, it is wise to focus on those that can be generalized and technically feasible. Furthermore, according to CuttingCutting (2003) in figure 2.7, some depth cues are more important than others. Hence, we discuss in particular the following cues.

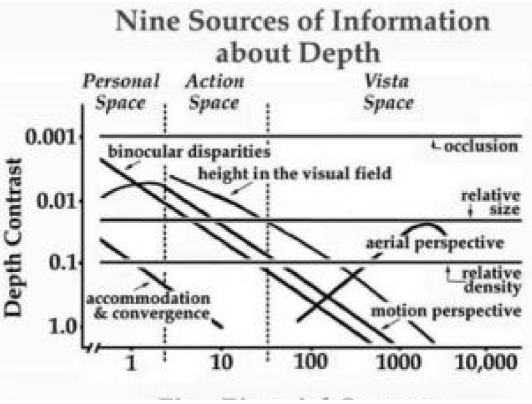


Figure 2.7: Importance of depth cues

Occlusion

Occlusion is the most impactful depth cue. It offers dominant information about depth across all distances, covering personal, action, and visa space as the graph suggestedCutting (2003). If you see object B where you should be seeing object A, it is an unmistakable cue that B is in front of A. However, it only offers ordinal information instead of the amount. Difficulties about occlusion in AR/DR systems will be covered in the next section.

Perspective

Human eyes search for straight lines or repetitive patterns in order to infer the vanishing point in an image. The brain can then infer the distance of objects from the position of the imaginary point. This content-based depth cue has been known to us since early civilization. Many paintings and photographs would add/capture converging lines to give the picture depth.

Relative Size and Density

The importance of relative size and density remain constant across the distances. This is where psychology comes in. When judging the depth of a familiar object, people draw from their experience and assume the object they see is the same size as the one in their memory; they infer the distance based on the size and density of objects relative to others. These cues can offer the amount in depth since sizes can be intuitively compared: twice the size in view suggests twice the distance given real sizes are about the same.

Color and Luminance

Studies around color stereopsis (see figure 2.8)Dengler and Nitschke (1993) and more general literature Sundet (1978) have shown that colors have effects on depth perception. A cool color, such as green, tends to appear farther than a warm color, e.g. red. However, experiments show that this warmth-depth relationship can be affected by environment brightness; furthermore color may not be dominant depth cue for complex models to be seenBailey et al. (2006). Luminance is more influential than hue to depth perceptionPayne (1964). Stronger luminance indicates closer to the observer. Besides, luminance contrast is also an important depth cueO'Shea et al. (1994): more contrast means closer. Beyond these early literature with simple shapes or lines as objects, this contrast-depth relationship has been further proved true in experiments with 3D scenesNan-Ching and Inanici (2012). In AR/DR systems, however, depth perception may rely on the complex interaction between color, luminance, and others. One studyDo et al. (2020) with handheld AR devices has shown that the depth perception of a simple high-fidelity shape can be influenced by a very warm color on the condition that the luminance is strong; complex shapes are not affected by color hue. More interactive influence of color and luminance remain unclear.

2.2.2 Sources of Problems

Kruijff et al.Kruijff et al. (2010) have summarized all origins of perceptual issues into: environment; capturing; augmentation; display devices; and user. Among them, environmental and capturing causes can largely

Figure 2.8: Chromostereopsis of red and blue

be avoided or mitigated through non-AR solutions, e.g. through creating environment with less clutter and pattern; better object visibility; even lighting; and high-quality capturing devices. Display devices make advancements continuously offering better brightness and contrast; less reflection and latency. User-dependent issues are evasive and constrained by sensor capabilities, e.g. accurate kinetic depth requires accurate movement and pose detection; and to improve physiological depth, sensors must be in place to track eyeball changes, such as vergence, accommodation, and pupil dilation. Therefore, issues caused by augmentation, in other words, the registration of virtual content over the real, is worth most attention of current AR researchers. They are registration errors, occlusions, layer interference and layout, and rendering and resolution mismatchKruijff et al. (2010).

Chapter 3

Design of a New Cutaway Interaction

After a study into literature, we have found out that a great amount of efforts have been put into presenting objects hidden behind an object of interest as in Conceal. However, presenting the inside of an object has received much less attention. Cutaway, as a conventional illustration technique for spatial understanding of models with complex inside structures , is worth more efforts in DR. Therefore, we design a new cutaway interaction in this chapter. Its cutaway shape is flexible and adaptive; it is equipped with rendering techniques that further facilitate spatial perception. This chapter elaborates on all system components: user-drawn cutaway opening, user-defined cutaway depth, adaptive cutaway direction, adaptive cavity color, and other rendering techniques and choices. motivations of certain designs. See figure 3.1 for the system workflow.

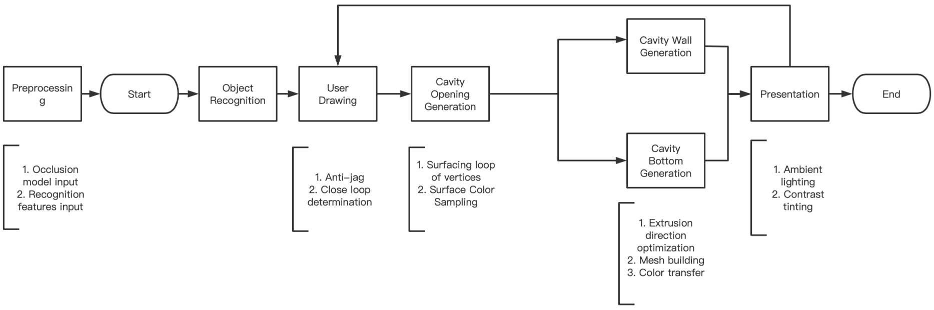


Figure 3.1: System Workflow

3.1 User-drawn Cutaway Opening

An innovative feature of this system is that it allows the user to define the cavity opening on the object of interest. When the system launches, it starts to detect the object of interest. When the object is detected, the cursor will turn into a red dot to prompt user input. The user may click on the model to define vertices on the cavity opening to be cut out. The user may turn or move the target object to find appropriate cutting path. The cavity opening should be a closed loop of segments. When a closed loop is formed, the system will generate a fully triangulated mesh based on the loop of vertices user defined. The mesh will then be used as a mask of stencil buffer. When rendering the inner object of the target object, the stencil buffer will be read to determine whether the pixel is rendered or not, thus creating a portal effect.

This innovative user-defined cutaway opening will give the user more freedom to expose the inner object: the user can decide what part of the inner object to be occluded by the outer object based on importance.

3.1.1 How to Form the Loop

The first point the user clicks on the target object will be used as the start of the loop; every point the user defines since will be measured against it regarding the Euclidean distance. If the distance is smaller than a threshold value, the new point will be considered as the end point. The end point will be connected to the start point to complete loop. Also note that when the user tries to draw a new point, the system measures its Euclidean distance from the last point the user has created. The new point is successfully created only when the distance is beyond a certain threshold. This is a design to prevent overly dense points in a small area, or jagged lines.

The system is demonstrated on a laptop computer with a touch pad for positional user input. This design can be easily transplanted to a mobile hand-held AR system, where the user can move the finger on the screen while moving the viewpoint. With this loop formation algorithm would be able to sample points based on the track of the finger and produce smooth 3D loops of segments.

3.1.2 Mask Mesh Generation and Triangulation

After the sequenced vertices are defined, the system generates a mask mesh. A key problem is to calculate the geometry of the cavity opening. The desired cavity geometry should be outlined/bounded by the sequenced vertices drawn by the user; it should look natural and compact without vertices added to create unnecessary surface area, which could like artifacts. Therefore, the algorithm needed can be a triangulation of sequenced vertices optimized for the least total area of all triangles. Barequet and SharirBarequet and Sharir (1995) have given us an algorithm that optimized for the total length of all triangle edges. The same thinking of dynamic programming can be used to design an algorithm optimized for the total triangle area. A general formula for this is presented by Zou et al.Zou et al. (2013).

$$W(D) = \min_{t \in T_D}(w(t) + W(D_1) + W(D_2)) \tag{3.1}$$

where $W(\cdot)$ is defined as the weight function (total area) of a 3D polygon; D is a 3D polygon; t is a triangle with all 3 vertices on D; $w(\cdot)$ is the weight (area) of triangle t; e_D is the incident edge shared by D and another polygon. The optimization target for the triangulation is the lowest total area of D, the 3D polygon that includes all the user-defined vertices. The algorithm we have designed is a top-down divide and conquer. In order to find the optimal W(D), when the algorithm starts, it finds all potential D1, t, and D2 combinations. It then recursively finds the optimal $W(D_1)$ and $W(D_2)$ by going through all combinations of triangle and sub-polygons. Once a 3D-polygon or triangle is evaluated for optimization, its weight (in our case, area) is memorized to avoid repetitive calculation. The algorithm is explained in steps here:

1. Start with all vertices as D.
2. If D is a triangle, calculate its area and record its indices and go to step 4; otherwise go to step 3.
3. Arbitrarily choose an edge as the incident edge; iterate through all of its incident triangles and incident 3D polygons D_1 and D_2 and record the minimum sum of $w(t), W(D_1), and W(D_2)$ and its triangulation; go to step 1 if D_1 or D_2 is not a triangle.
4. Return the minimum sum and its triangulation

This triangulation promises a manifold shape because D_1 and D_2 shares only one common vertex thus no common edges.

3.2 Cavity Generation

This section will explain how the wall and bottom of the cavity is generated.

3.2.1 User-defined Cutaway Depth

After an cavity opening is formed, the wall and bottom of the cavity is generated. The bottom of the cavity is the same geometry as the cavity opening but translated to a certain distance that is defined by the user through a slide bar. The wall of the cavity is thus an extrusion from the opening to the bottom. It is rendered with quads: every two vertices from the bottom and every two from the opening form a primitive.

Again, this designs gives the user more freedom regarding how deep he/she wants to expose the inner object.

3.2.2 Adaptive Cutaway Direction

The direction of the extrusion of the wall is adaptive to fit the best viewing angle at the inner object. When the user cuts out a disk-like shape from a object, he/she intuitively expects to look through the cavity opening from the top of the cavity in order to have the most view of the inside. Therefore, the extrusion to form the wall should be vertical to the "cap" cut out. Although the opening cap is not not a plane, we can approximate a plane to fit the most vertices on its geometry. Therefore, an algorithm is needed to find the best fitting plane for the set of vertices. The system employs a least square solution. Assume the plane equation $ax + by + c = z$; the n vertices $V = (x_0, y_0, z_0), (x_1, y_1, z_1), (x_2, y_2, z_2), ...(x_n, y_n, z_n)$; we have the following equation:

$$\begin{bmatrix} x_0 & y_0 & 1 \\ x_1 & y_1 & 1 \\ ... & & \\ x_n & y_n & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} z_0 \\ z_1 \\ ... \\ z_n \end{bmatrix} \tag{3.2}$$

or

$$Ax = B \tag{3.3}$$

thus:

$$x = A^{-1}B \tag{3.4}$$

when A is not n by n, we use generalized inverse:

$$x = (A^TA)^{-1}A^TB \tag{3.5}$$

After we have the equation of the best fitting plane, its normal vector can be $N_1 = (a, b, -1)$ or $N_2 = (-a, -b, 1)$. Sight direction from the camera to the last vertex should be less than 90 degrees from N. Therefore, given sight vector S, if $N_1 \cdot S$ ¿ 0, we take N_1

as the extrusion direction, otherwise N_2.

3.2.3 Color Transfer for Cavity Wall

In order to render the cavity as a organic part of the object target, this system employs a color transfer technique. The technique has three steps: sampling the pixel color around the user-defined vertices; make the color cooler; render the corresponding opening and bottom vertex in the cavity with the new color. As we discusses in 2.2.1 , a cooler color may indicate a farther distance. Therefore, the system transfer a cooler color to the cavity as an experiment. The color sample at the opening pixel is firstly converted from RGB to HSV space; and then the Hue value is increased by a certain margin (20% in the experiment).

3.2.4 Other Rendering Techniques

Besides the directional lighting, the system employs ambient lighting to give the inner object a more evenly lit environment. As we assume the outer object is in a evenly lit, this lighting complement can render the inner more like part of the outer. This will also gives more clarity to the inner object. The color of the inner object is also tinted as a contrast to the cavity wall and the surface of the outer object. This is an effort to improve the user's focus on the inner object.

The red drawing line is essential for the user to see the cutting. After the cavity is generated, it is also necessary for the user to differentiate the real surface of the outer and the virtual wall of the inner cavity because the wall is rendered in a similar color as the surface. In order to make the red lines look more natural, we use alpha-blending to avoid any solid color.

Chapter 4

Implementation

This chapter presents the hardware and software required by the system; its workflow including pre-processing and real-time functioning; and the last but not the least, data structures and choices employed in algorithm implementation.

4.1 Hardware and Software Specifications

4.1.1 Unity

Unity is a 3D rendering platform that is able to work with the AR platform solutions we discussed in chapter 2. It allows for custom shaders for individual objects and and meshes; simple creation of graphical user interface ; and most importantly, high compatibility. The rendered project can export to mobile platforms for handheld AR experience, such as iOS and Android; it can work on desktop platforms as it is, with a single camera and no inertial movement units or other sensors; or it can directly render to head-mounted devices with binocular vision and sophisticated sensors, such as vive and OculusUni. Its employment showcases that the system is a general-purpose and platform-independent DR interaction.

4.1.2 Vuforia

As we introduced in 2.1.3, Vuforia is AR platform solution that packages object recognition, tracking, pose estimation, and projection. It is able to work with the rendering platform, Unity, to form an entire AR workflow. The system employs Vuforia instead of others because of its excellent compatibility. It is capable of deploying to multiple platforms as Unity, such as iOS, Android, PC, and head-mounted devices. Together with Unity, the software specifications guarantee a generalized system of compatibility.

4.1.3 Preprocessing

The preprocessing of the system consists of three steps, scanning for tracking, input of the outer object model, and manual registration of the inner object. The system employs Vuforia's object recognition and tracking capability.

Step1 is the input the mesh model of the outer object as the occlusion model in Unity. When the user clicks on the object of interest, Unity runs a collision detection algorithm that is based on the geometry of the object. The collision positions are the vertices that will be used to generate the cavity opening in the later step. The mesh model can be obtained through photogrametry softwares, such as Meshroom, or through a third-party laser scanning service as we did for this project.

Step 2 is recognition and tracking preparation. With the model obtained from step 1, we could feed Vuforia Model Target Vuf (a) with it and complete the preparation with little effort. Or we make use of Vuforia's Object Scanner to extract features from the object of interest. The Object Scanner is an android-based app that should be used with a printable background image. The image serves two purpose: provide pose information for the object to be scanned and decide the culling areaVuf (c).

Step 3 is the manual registration of the inner object to the tracking target (outer object). In this step, the user decide the spatial relationship between the inner object and the real-world object of interest.

4.2 Algorithms Implementation

The algorithms are implemented with C# in Unity environment.

4.2.1 Cavity Opening

For the mesh generation algorithm, we define a recursive function MinAreaTrg that takes in an array of vertex indices as the parameter. Each array of vertex indices corresponds to a 3D polygon. We identify the 3D polygons by a hash function that maps the array of indices to a string. The hash function sees the array as a stream of bytes and converts it to an string of base64 chars. With the increasing order of indices, the hash function ensures that each 3D polygon is mapped to a different string without collision. This one-to-one mapping allows us to use the string as the key to a hashtable that memorizes the smallest area for each 3D polygon to avoid repetitive calculation. The value stored in the hashtable is of a custom data structure: a struct consisting of a float as the total area and an int array as the indices of the 3D polygon.

4.2.2 Cavity Wall and Bottom

The least squared method employed for the adaptive cutting direction involves matrix calculation, such as transpose and inverse operations on matrices of an arbitrary size. This is realized through CSML libary, a lightweight C# library for linear algebraCSM.

Chapter 5

Evaluation

This chapter firstly evaluates the overall usability of the system through a cognitive walk-through; then it assesses the major components separately regarding visual results and computational performance.

5.1 Usability

The system claims to improve the experience of presenting the inside of an object through an innovative DR interaction: more freedom of creating cutaway opening and better spatial perception. This section evaluates the usability of this interaction through moderated cognitive walk-through.

5.1.1 Motivation

There are many usability evaluation models, such as claim analysis and heuristic evaluation. Although studies John and Marks (1997) have shown these models are not highly effective regarding resulting change in designs, most are able to predict major problems with system interaction. Therefore, we have decided to employ the cognitive walk-through model to assess our system's usability because of its task-oriented approach. Due to time constraint and COVID epidemic, we have employed a simplified version. The test assesses how easily the user carries out certain tasks with the system.

5.1.2 Experiment Design

We try to follow the steps and principles by Wharton et al.Wharton et al..

- User: medical student (substituted by the developer due to constraints)

- Scenario: learning the position of brain partitions relative to the skull with a desktop monitor-based AR/DR setting
- Task: expose and watch the position of brain stem
- Actions:
    1. Rotate the skull model to find an appropriate opening position
    2. Slide the depth bar on top to define cutaway depth
    3. Click on the skull to define cutting lines (or simultaneously rotating the skull)
    4. Finish cutting by clicking near the first point defined
    5. Rotate and move the model and watch brain stem's position
    6. If not happy with the cutaway, cut out a new hole by going to step 1

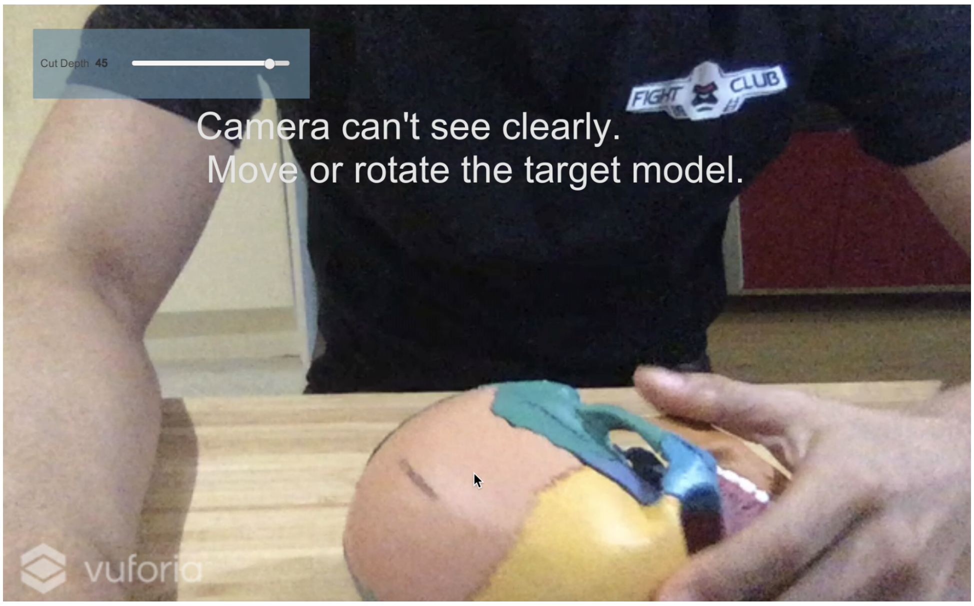
For each step in the actions, we ask four key questions:

1. Will the user try to achieve the correct effect?
2. Whether the user can notice the best way to achieve the goal?
3. Whether the user can identify the correct option?
4. Will the user understand the feedback received from the system?

5.1.3 Results

Step 1 figure 5.1, the user is instructed to rotate the skull model in front of the camera and look at the monitor for best cutaway angles. When the system starts, it scans the camera images and tries to recognize the skull model. A line of text on the screen prompts the user to move or rotate the target model to ensure the recognition. When the recognition is successful, the text prompt changes and a red dot appears on the skull to suggest cutaway point. Therefore, the user can understand where to start the cutaway even though the viewing angle from the camera is different from his own eyes. It is intuitive to rotate the object with hands and there is no other explicit operations required. The screen image and the text prompt that change with the user's rotation is self-explanatory and serve as clear feedback for the user to understand.

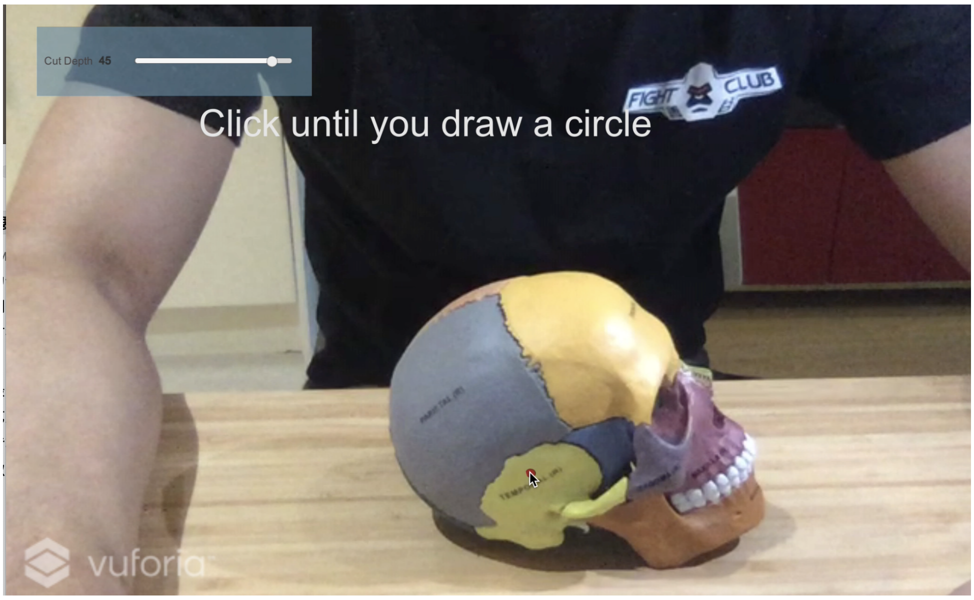
Step 2 figure 5.2, the user has been instructed that the cavity depth can be adjusted through the slide bar. The user may estimate the desirable depth and adjust later. The slide bar has a clear text prompt and scale unit for the user to identify the correct usage. When the user slides the bar, the potential cavity depth changes simultaneously as a result.



Camera can't see clearly.
Move or rotate the target model.

Figure 5.1: Action step 1: Rotate the skull model to find an appropriate opening position

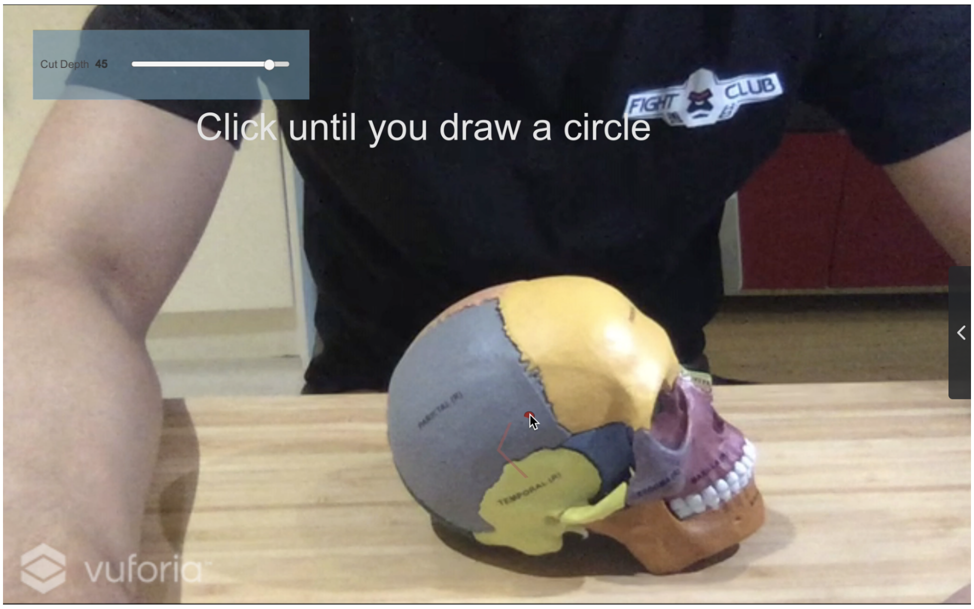
It is easy to associate the sliding action and text change as feedback. Furthermore, even if the user forgets to slide the bar, the system's default value will suffice for a reasonable result.



Click until you draw a circle

Figure 5.2: Action step 2: Slide the depth bar on top to define cutaway depth

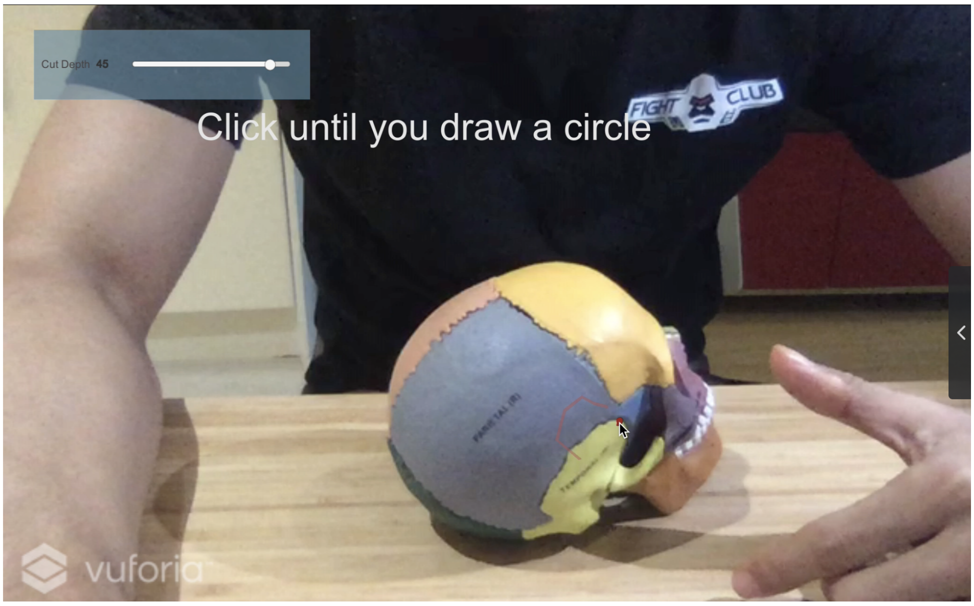
Step 3 figure 5.4, the user has been instructed to draw a circle-like shape by clicking on the skull in this step. Although the user does not know what to expect when clicking, he/she will understand points created are connected one by one in the order of creation once he/she clicks the second time. The lines grow organically with user's clicks. It is intuitive for the user to associate the clicking operation with new dots created and connected. In fact, the user has found out that rotating is essential to drawing accuracy.



Cut Depth 45

Click until you draw a circle

Figure 5.3: Action step 3: Click on the skull to define cutting lines (or simultaneously rotating the skull)



Cut Depth 45

Click until you draw a circle

Figure 5.4: Action step 3: Turn and draw

This is because it can be very hard to trace a line extending into the image (z-axis) since the distance on the z-axis is increasingly compressed as the line extends farther. Figure 5.5 is what happened when the user tries to trace the coronal suture and then the sagittal suture.

However, a potential failure could be that the tracking component fails during user's operation. To address this, the system will prompts the text "Camera can't see clearly. Move or rotate the target model." This serves as the feedback for the user to associate with the loss of response when clicking. The correct action is suggested to the user. After the user moves the skull to recover tracking, the text changes back to the previous drawing prompt suggesting drawing can resume now.

Also, note that clicking is reasonable for desktop monitor-based AR/DR applications while it can be adapted to tapping or dragging when transplanted to other hardware settings.

Step 4 figure 5.6, the prompt suggests the user to draw a circle, which is reasonably easy to understand. When the user fails to draw the circle by connecting to the first dot, the system waits for the correct operation. When the first dot is connected, the user can immediately see a cavity bounded by the circle he/she has just drawn, inside of which, there is the brain stem. The feedback is instant and can be easily associated with the last click the user has performed. Step 5 figure 5.7, because the user has seen how the monitor images change to the rotation and movement of the real-world model, he/she naturally turns to the same operations to scrutinize the new graphical change: the cavity. Again, the potential failure could be the loss of tracking. This is addressed by the prompt text in the same way as in step 3.

Step 6 figure 5.9, when the user is not happy with the previous cut, he/she may try to redraw by repeating the previous steps. This action is also prompted by the text on the screen. The how in this step has been resolved in previous steps. The feedback is instant when the user clicks two more times to see a new line connected. Considering previous experience, it is straightforward for the user to see it as the start of drawing a new circle and a cavity cut out after.

Through these steps, the user can freely choose a viewpoint to look at the brain stem without worrying that it would be occluded; at the same time, if thoughtfully done, a majority of the skull model can be preserved in the view as a reference of spatial relationship.

5.1.4 Advanced Task

In the experiment, we have conducted a separate cognitive walk-through with a more advanced task goal. This is a simulation of making exploratory blurr holes in craniotomy. In clinics, trauma patients may experience transtentorial herniation/brainstem compression. Surgeons may choose to make holes in the skull for examination or pressure releasebra. We have asked the user to expose certain areas of the cerebral cortex according to a pre-drawn sketch. The user is asked to define the cavities on the brain model as the shape and location indicated. Refer to figure 5.10. The user has successfully completed the task in about a minute: the cavities are reasonably similar to the sketch in shape and location marking the usability of the system.

5.2 Cavity Opening Generation

5.2.1 Visual Result

The system allows the user to define the cavity opening according to their own needs. This subsection will firstly elaborate on its advantages.

Compare in the figure 5.11, user-defined opening produced by our system v.s. system-defined opening commonly seen as in Wimmer et al. (2008).

- Spatial perception: user-defined cutting lines are more naturally shaped giving the impression that the cutting is *growing on* the object surface; while the rectangular cutting window is evidently *added to* the object, flat with no depth, looking like an artefact. Note that the still frames only have pictorial depth cues; in video MR systems, its spatial perception will be enhanced by kinetic depth cues as we introduced in 2.2.1, such as relative motion parallax and motion perspective.
- Information preservation: User-defined opening can reveal most of the inner object while preserving more detailed information on the outer object than conventional cutting: note labels on the right are unnecessarily occluded resulting in loss of information.

For inner objects that are not as deep buried under the surface, this depth perception improvement is less evident but still observable. See figure 5.12 In medical imaging and other use cases, the user has better understanding and expertise regarding what to show and what not. This figure5.13 shows the view of the brain stem when the entire temporal bone cut out. This could be an useful view for anatomy education, archaeology, or even for clinics but cannot be easily done with traditional method.

- Cavity opening as information: a user-defined shape of the cavity opening may contain information while a primitive opening cannot. Here the cavity opening suggests the anatomical structure of the temporal bone.

5.2.2 Time and Space Complexity

According to the equation 3.1, each 3D polygon is divided into a triangle with the incident edge and two polygons D_1 and D_2; then the function is recursively called for D_1 and D_2;

this process goes on until D becomes a triangle. If we look at the process bottom-up, each triangle for the original D is calculated; its total number is C_n^3. Then each quad is optimized based on the result of the last step, total number C_{n-1}^4. This process builds up until $C_n^n$when the entire 3D polygon is optimized. Since optimized results for all the polygons are done once and memorized, we can easily have the upper bound of complexity of this equation:

$$C_n^3 + C_n^4 + C_n^5 + ... + C_n^n = 2^n - C_n^2 - C_n^1 \tag{5.1}$$

Hence the time and space complexity for a naive implementation is $O(2^n)$. A simple optimization is done in this project. When dividing 3D polygon D, we stop recursion for current division if any term in equation 3.1 is bigger than the previously calculated value. This trimming technique has proven to be effective in practice. However, further optimization can still be done. The best result can be as low as $O(n^3)$ and $O(n^2)$ for time and space with minimal sets Zou et al. (2013).

5.3 Cavity Wall and Bottom

5.3.1 Visual Result

The wall is critical to the building of depth. It offers essential depth cues of occlusion and perspective as we discussed in 2.2.1. Compare the rendering with cavity wall on the left and without on the right in figure 5.14. The system optimizes the cutaway direction to obtain an intuitive and useful viewpoint at the inner object. Compare the figures 5.16. We want to cut out a cavity from the top of the skull. The top figure shows that the wall generation without direction optimization: instead of going down to the brain tissue, the wall extrudes to the left, which is not an intuitive cutaway. The bottom figure, however, shows that with the optimization, the wall extends down intuitively. This auto-adaptation is especially effectively when trying to draw a narrow but deep cavity. The bottom of the cavity has also contributed to the depth perception. Its role can be best observed when the inner object is smaller than the hole opening, in which case the bottom is exposed and serves as a background to the inner object. This background impresses the user that this area is inside, different from the surface. Figure 5.17 shows that the rendering without the bottom confuses the surface with the inside of cavity interfering user's perception of depth.

5.3.2 Time and Space Complexity

We assess the performance of wall direction optimization. Consider formula 3.5, in which A^T size 3 x n and matrix A of size n x 3 are firstly multiplied as step 1. This multiplication takes $O(n^2)$ regarding time complexity. In step 2, the inversion of the result in step 1, an 3 x 3 size matrix takes $O(1)$. In step 3, the multiplication between the result of step 2, an 3 x 3 matrix and A^T , takes $O(n^2)$; step 4, the result of step 3, matrix of 3 x n is multiplied by B of size n x 1 takes $O(n)$. Therefore, the overall time complexity is $O(n^2) + O(1) + O(n^2) + O(n)$, which equals $O(n^2)$ where n is the number of user-defined points. The space complexity of the least square method is straightforwardly O(n) as the maximum size of matrix it stores is n x 3.

5.4 Color Transfer and Lighting

The color transfer from the surface of the real object to the virtual inner wall has significantly improved the credibility of the latter. When human eyes see structures of the same color close together, the intuition is that they are structurally connected, see figure 5.18. Also, the slightly cooler color on the wall has indicated a difference of depth than the model surface. Moreover, ambient lighting has proven to be effective in improving the visibility of the inner object. The contrast color used by the inner object has made it stand out from the background with improved focus, see figure 5.19

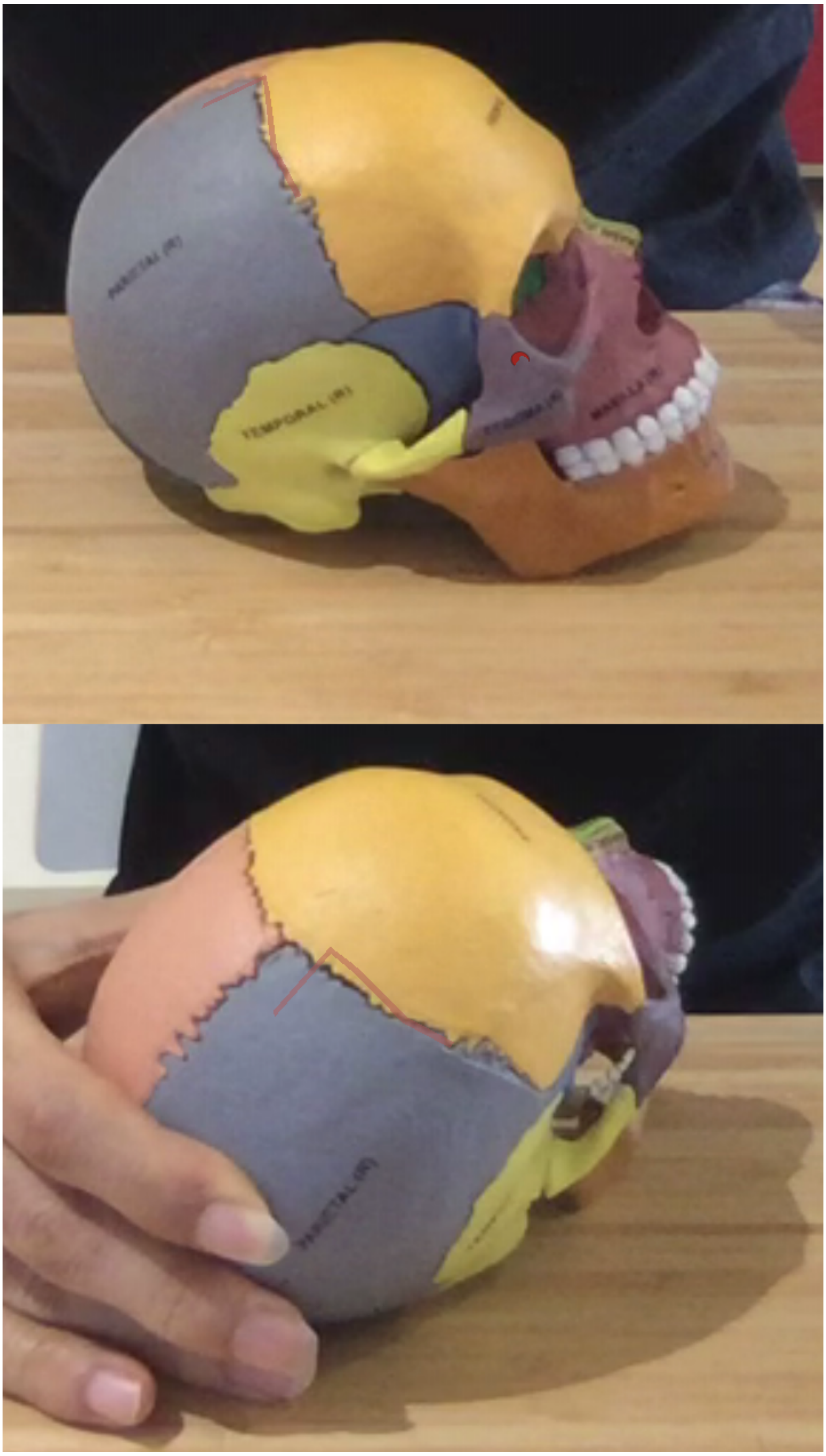
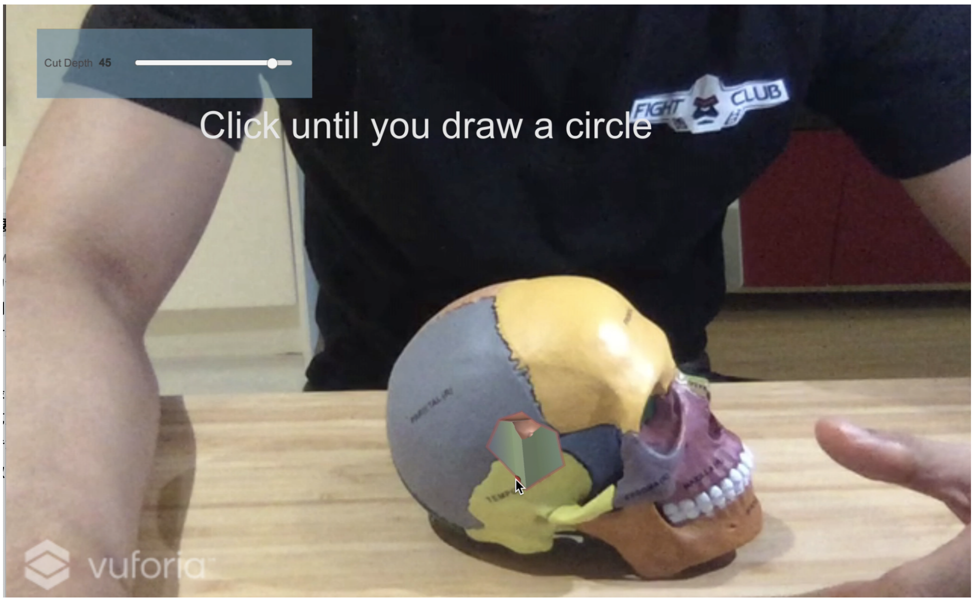
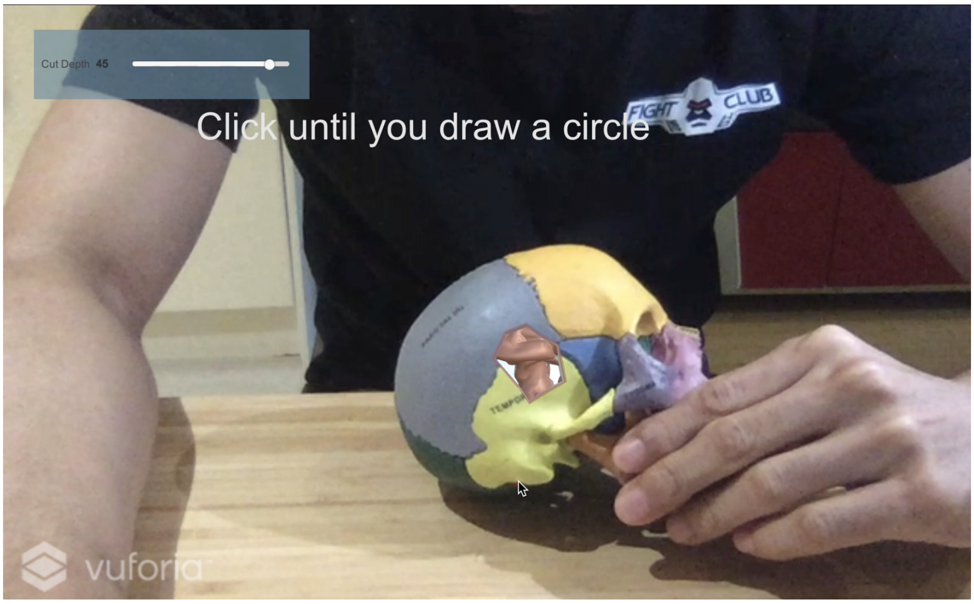


Figure 5.5: Failure to trace the sutures on the skull due to lack of viewpoint adjustment


Cut Depth 45

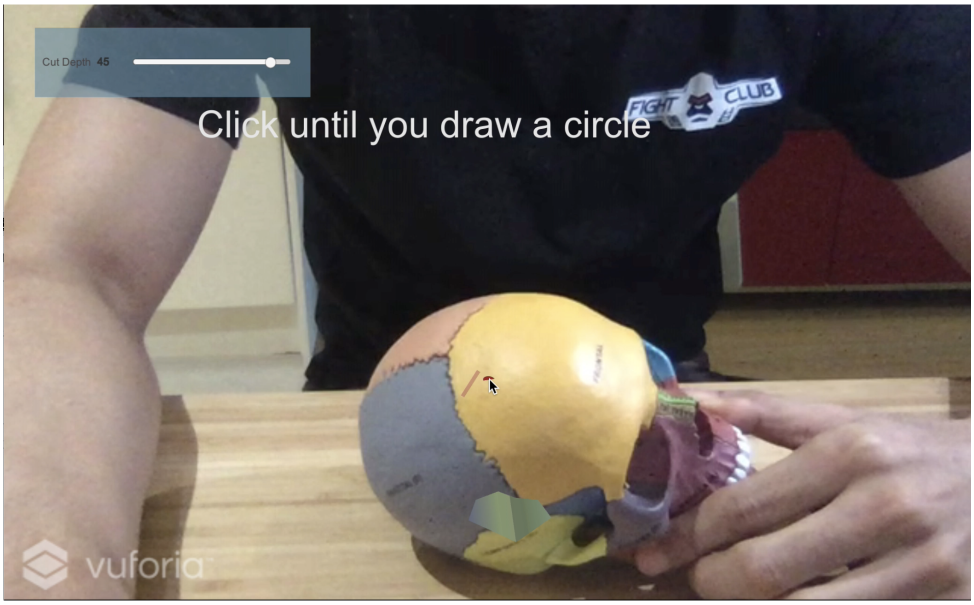
Click until you draw a circle

Figure 5.6: Action step 4: Finish cutting by clicking near the first point defined


Cut Depth 45

Click until you draw a circle

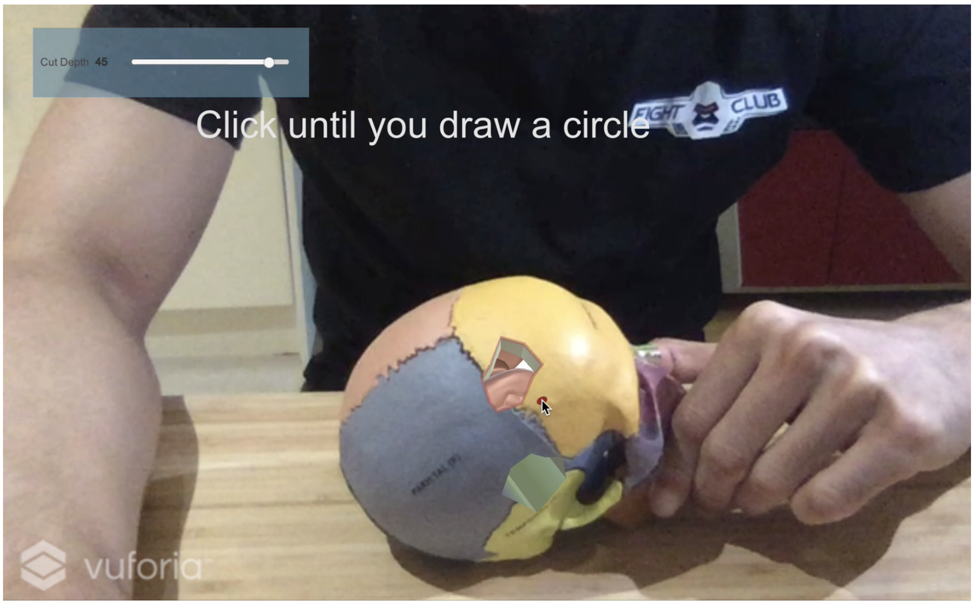
Figure 5.7: Action step 5: Rotate and move the model and watch brain stem's position



Cut Depth 45

Click until you draw a circle

Figure 5.8: Action step 6: If not happy with the cutaway, cut out a new hole by going to step 1



Cut Depth 45

Click until you draw a circle

Figure 5.9: Action step 6: Finished two drawings

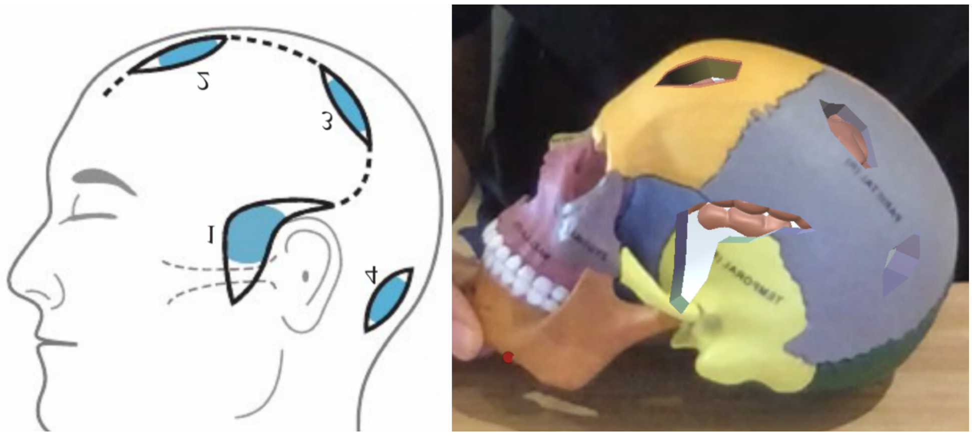


Figure 5.10: Left: sketched holes Right: user-defined holes

Figure 5.11: Contrast between user-defined opening and default primitive

Figure 5.12: Shallow inner object: contrast between user-defined opening and default primitive

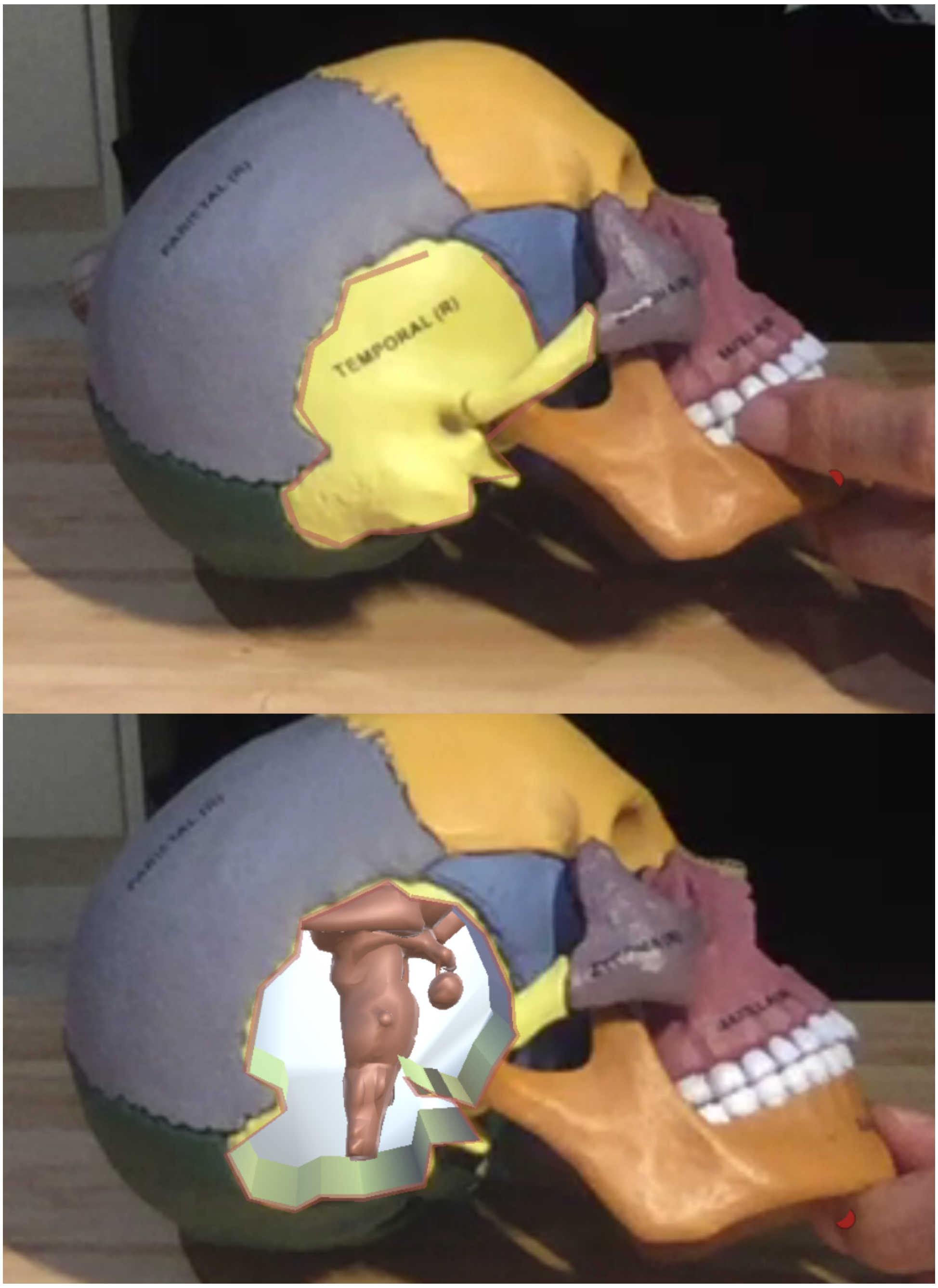


Figure 5.13: User-defined cutaway: temporal bone removed

Figure 5.14: Left: rendering with wall; Right: without wall

Figure 5.15: Without wall direction optimization: not intuitively extruded wall

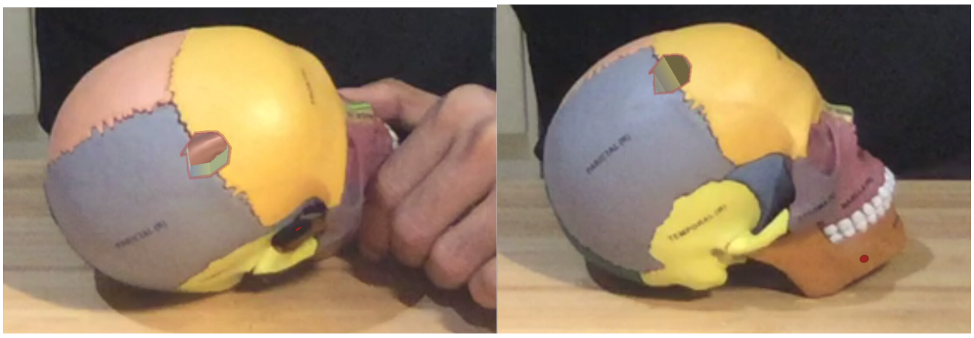


Figure 5.16: With wall direction optimization: intuitive extruded wall

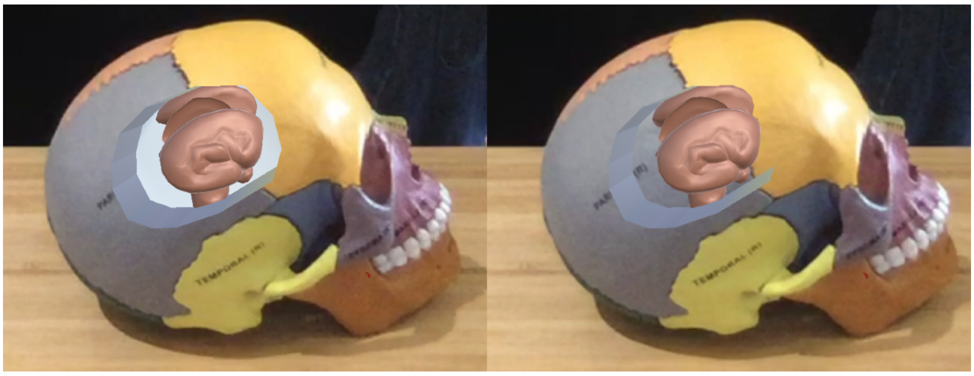


Figure 5.17: Left: without bottom; Right: without bottom

Figure 5.18: Left: with color transfer and ambient lighting; Right: without

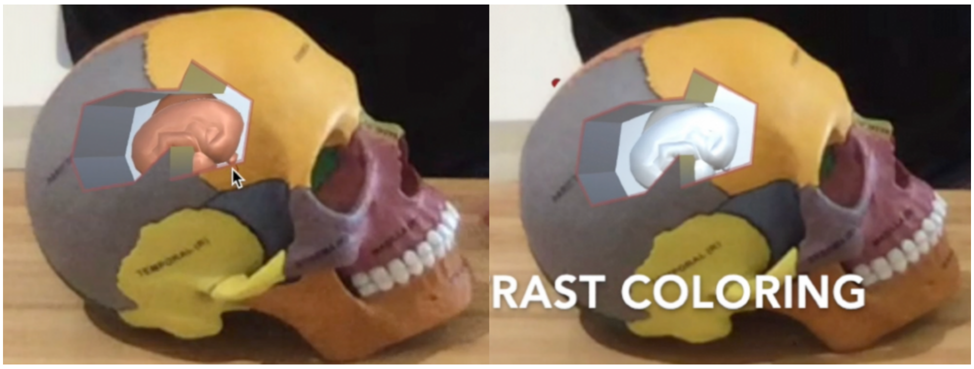


Figure 5.19: Left: without contrast; Right: with contrast

Chapter 6

Conclusion

This dissertation firstly introduces the definition of AR and related concepts; their principles and platform solutions; it then moves its focus to DR by classifying its functions and applications. Specifically, it discusses the obstacles in DR. This dissertation then shifts its weight to spatial perception for AR/DR systems explaining important depth cues, such as occlusion and color and luminance.

We then introduces an innovative design of cutaway interaction that allows the user to freely cut out a real-world object's surface. This design is based on cutaway, a basic function of DR, as defined in the background chapter. It answers the typical challenges in DR: background generation is performed through user-defined extrusion-based mesh generation; the region of interest is detected through a feature-based system. It has taken advantage of the depth cues mentioned above to improve spatial perception and visual credibility.

Evaluation has proven that the system is highly usable to the scenario of medical education. Time and space performance of the key algorithms are reasonable and improvable.

6.1 Future work

The opening generation algorithm can be further improved regarding is computational performance. New interactions and principles can be designed to allow for the arbitrary change of existing opening and cavity.

In order to further improve the visual credibility and presentability of the system, more efforts can be done in rendering techniques, such as adaptable resolution, environment light sensing, automatic tinting algorithms based on contrasting colors;

The current system is based on polygonal meshes. Volumetric data, as a important data format in medical applications, can be included in the future.

With regards to ROI detection, point cloud can be used to replace the model scanning in the pre-processing stage; the occlusion model can also be substituted with it as part of the effort to reduce or eliminate the pre-processing stage. Registration of the virtual object can be performed through an dedicated human-computer interface.

Bibliography

David Drascic and Paul Milgram. Perceptual issues in augmented reality. *Proc. SPIE*, Vol. 2653, 04 1996. doi: 10.1117/12.237425.

Paul Milgram, Haruo Takemura, Akira Utsumi, and Fumio Kishino. Augmented reality: a class of displays on the reality-virtuality continuum. In Hari Das, editor, *Telemanipulator and Telepresence Technologies*, volume 2351, pages 282 – 292. International Society for Optics and Photonics, SPIE, 1995. URL `https://doi.org/10.1117/12.197321`.

K. Lyons, M. Gandy, and T. Starner. Guided by voices : An audio augmented reality system. *Proceedings of the International Conference on Auditory Display*, pages 57–62, 2000.

Anna N. Nagele, Valentin Bauer, Patrick G. T. Healey, Joshua D. Reiss, Henry Cooke, Tim Cowlishaw, Chris Baume, and Chris Pike. Interactive audio augmented reality in participatory performance. *Frontiers in Virtual Reality*, 2021.

V. Hayward, O. R. Astley, M. Cruz-Hernandez, D. Grant, and G. Robles-De-La-Torre. Haptic interfaces and devices. *Sensor Review*, pages 16–29, 2004.

Ronald T. Azuma. A Survey of Augmented Reality. *Presence: Teleoperators and Virtual Environments*, 6(4):355–385, 08 1997. doi: 10.1162/pres.1997.6.4.355. URL `https://doi.org/10.1162/pres.1997.6.4.355`.

Jaewoon Lee, Yeonjin Kim, Myeong-Hyeon Heo, Dongho Kim, and Byeong-Seok Shin. Real-time projection-based augmented reality system for dynamic objects in the performing arts. *Symmetry*, 7(1):182–192, 2015. ISSN 2073-8994. doi: 10.3390/sym7010182. URL `https://www.mdpi.com/2073-8994/7/1/182`.

David G. Lowe. Object recognition from local scale-invariant features. *Proceedings of the International Conference on Computer Vision*, 2:1150–1157, 1999.

Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features. *Computer Vision and Image Understanding*, 110(3):346–359, 2008.

Aripriharta Siti Sendari, Adim Firmansah. Performance analysis of augmented reality based on vuforia using 3d marker detection. pages 294–298, 2020.

Model targetsg, a. URL `https://library.vuforia.com/features/objects/model-targets.html`.

Jeroen D. Hol, Thomas B. Schön, Henk Luinge, Per J. Slycke, and Fredrik Gustafsson. Robust real-time tracking by fusing measurements from inertial and vision sensors. *Journal of Real-Time Image Processing*, 2:149–160, 2007.

Reinhard Koch Michael Felsberg. Editorial for the special issue on markerless real-time tracking for augmented reality image synthesis. *Journal of Real-Time Image Processing*, 2:67–68, 2007.

Fundamental concepts. URL `https://developers.google.com/ar/develop/fundamentals?hl=en`.

More to explore with arkit 5. URL `https://developer.apple.com/augmented-reality/arkit/`.

Getting started, b. URL `https://library.vuforia.com`.

Steve Mann. Mediatedreality, 1994.

Hideo Saito Shohei Mori, Sei Ikeda. A survey of diminished reality: Techniques for visually concealing, eliminating, and seeing through real objects. *IPSJ Transactions on Computer Vision and Applications*, 9:1–14, 2017.

Hanseob Kim, Taehyung Kim, Myungho Lee, Gerard Jounghyun Kim, and Jae-In Hwang. Ciro: The effects of visually diminished real objects on human perception in handheld augmented reality. *Electronics*, 10(8), 2021. ISSN 2079-9292. doi: 10.3390/electronics10080900. URL `https://www.mdpi.com/2079-9292/10/8/900`.

Kentaro Yagi, Kunihiro Hasegawa, and Hideo Saito. Diminished reality for privacy protection by hiding pedestrians in motion image sequences using structure from motion. *2017 IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct), Mixed and Augmented Reality (ISMAR-Adjunct), 2017 IEEE International Symposium on, ISMAR-ADJUNCT*, pages 334 – 337, 2017. ISSN 978-0-7695-6327-5. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edseee&AN=edseee.8088516`.

J. Herling and W. Broll. Advanced self-contained object removal for realizing real-time diminished reality in unconstrained environments. *2010 IEEE International Symposium on Mixed and Augmented Reality, Mixed and Augmented Reality (ISMAR), 2010 9th IEEE International Symposium on*, pages 207 – 212, 2010. ISSN 978-1-4244-9345-6. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edseee&AN=edseee.5643572`.

Takumi Yoshida, Kensei Jo, Kouta Minamizawa, Hideaki Nii, Naoki Kawakami, and Susumu Tachi. Transparent cockpit: Visual assistance system for vehicle using retro-reflective projection technology. *2008 IEEE Virtual Reality Conference, Virtual Reality Conference, 2008. VR '08. IEEE*, pages 185 – 188, 2008. ISSN 978-1-4244-1971-5. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edseee&AN=edseee.4480771`.

Benjamin Avery, Christian Sandor, and Bruce H. Thomas. Improving spatial perception for augmented reality x-ray vision. *2009 IEEE Virtual Reality Conference, Virtual Reality Conference, 2009. VR 2009. IEEE*, pages 79 – 82, 2009. ISSN 978-1-4244-3943-0. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edseee&AN=edseee.4811002`.

P. Barnum, Y. Sheikh, A. Datta, and T. Kanade. Dynamic seethroughs: Synthesizing hidden views of moving objects. *2009 8th IEEE International Symposium on Mixed and Augmented Reality, Mixed and Augmented Reality, 2009. ISMAR 2009. 8th IEEE International Symposium on*, pages 111 – 114, 2009. ISSN 978-1-4244-5390-0. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edseee&AN=edseee.5336483`.

Daiki Kido, Tomohiro Fukuda, and Nobuyoshi Yabuki. Diminished reality system with real-time object detection using deep learning for onsite landscape simulation during redevelopment. *Environmental Modelling and Software*, 131, 2020. ISSN 1364-8152. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edselp&AN=S1364815220303753`.

Christian Kunert, Tobias Schwandt, and Wolfgang Broll. An efficient diminished reality approach using real-time surface reconstruction. *2019 International Conference on Cyberworlds (CW), Cyberworlds (CW), 2019 International Conference on*, pages 9 – 16, 2019. ISSN 978-1-7281-2297-7. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edseee&AN=edseee.8919241`.

Ada V. Taylor, Ayaka Matsumoto, Elizabeth J. Carter, Alexander Plopski, and Henny Admoni. Diminished reality for close quarters robotic telemanipulation. *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Intelligent Robots and Systems (IROS), 2020 IEEE/RSJ International Conference on*, pages 11531 – 11538, 2020. ISSN 978-1-7281-6212-6. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edseee&AN=edseee.9341536`.

Christoph Bichlmeier, Felix Wimmer, Sandro Michael Heining, and Nassir Navab. Contextual anatomic mimesis hybrid in-situ visualization method for improving multi-sensory depth perception in medical augmented reality. *2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on*, pages 129 – 138, 2007. ISSN 978-1-4244-1749-0. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edseee&AN=edseee.4538837`.

Naoto Ienaga, Felix Bork, Siim Meerits, Shohei Mori, Pascal Fallavollita, Nassir Navab, and Hideo Saito. First deployment of diminished reality for anatomy education. *2016 IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct), Mixed and Augmented Reality (ISMAR-Adjunct), 2016 IEEE International Symposium on, ISMAR-ADJUNCT*, pages 294 – 296, 2016. ISSN 978-1-5090-3740-7. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edseee&AN=edseee.7836520`.

Wilmot Li, Lincoln Ritter, Mancesh Agrawala, Brian Curless, and David Salesin. Interactive cutaway illustrations of complex 3d models. *ACM TRANSACTIONS ON GRAPHICS*, 26(3):31, 2007. ISSN 07300301. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edswsc&AN=000248914000034`.

Shigeru Owada, Frank Nielsen, Kazuo Nakazawa, and Takeo Igarashi. A sketching interface for modeling the internal structures of 3d shapes. *ACM SIGGRAPH 2006 Courses*, pages 12 – es, 2006. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edb&AN=108312583`.

T. IGARASHI, S. MATSUOKA, and H. TANAKA. Teddy: A sketching interface for 3d freeform design. *Proceedings of ACM SIGGRAPH 99*, pages 409–416, 1999.

CD Bruyns, S Senger, A Menon, K Montgomery, S Wildermuth, and R Boyle. A survey of interactive mesh-cutting techniques and a new method for implementing generalized interactive mesh cutting using virtual tools. *JOURNAL OF VISUALIZATION AND COMPUTER ANIMATION*, 13(1):21 – 42, 2002. ISSN 10498907. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edswsc&AN=000177407800003`.

C. Coffin and T. Hollerer. Interactive perspective cut-away views for general 3d scenes. *3D User Interfaces (3DUI'06), 3D User Interfaces, 2006. 3DUI 2006. IEEE Symposium on*, pages 25 – 28, 2006. ISSN 1-4244-0225-5. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edseee&AN=edseee.1647502`.

Erick Mendez and Dieter Schmalstieg. Importance masks for revealing occluded objects in augmented reality. *Proceedings of the 16th ACM Symposium: Virtual Reality Software and Technology*, pages 247 – 248, 2009. ISSN 9781605588698. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edb&AN=83626212`.

Oliver Kutter, A. Aichert, Christoph Bichlmeier, J. Traub, S. Heining, B. Ockert, E. Euler, and Nassir Navab. Real-time volume rendering for high quality visualization in augmented reality. 2008.

Felix Wimmer, Christoph Bichlmeier, Sandro-Michael Heining, and Nassir Navab. Creating a vision channel for observing deep-seated anatomy in medical augmented reality a cutaway technique for in-situ visualization. *Informatik aktuell*, 01 2008.

Shohei Mori, Fumihisa Shibata, Asako Kimura, and Hideyuki Tamura. Efficient use of textured 3d model for pre-observation-based diminished reality. *2015 IEEE International Symposium on Mixed and Augmented Reality Workshops*, pages 32 – 39, 2015. ISSN 9781467384711. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edb&AN=112661119`.

Siim Meerits and Hideo Saito. Real-time diminished reality for dynamic scenes. In *2015 IEEE International Symposium on Mixed and Augmented Reality Workshops*, pages 53–59, 2015. doi: 10.1109/ISMARW.2015.19.

K. Sugimoto, H. Fujii, A. Yamashita, and H. Asama. Half-diminished reality image using three rgb-d sensors for remote control robots. *2014 IEEE International Symposium on*

Safety, Security, and Rescue Robotics (2014), Safety, Security, and Rescue Robotics (SSRR), 2014 IEEE International Symposium on, pages 1 – 6, 2014. ISSN 978-1-4799-4199-5. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edseee&AN=edseee.7017676`.

S. Jarusirisawad and H. Saito. Diminished reality via multiple hand-held cameras. *2007 First ACM/IEEE International Conference on Distributed Smart Cameras, Distributed Smart Cameras, 2007. ICDSC '07. First ACM/IEEE International Conference on*, pages 251 – 258, 2007a. ISSN 978-1-4244-1353-9. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edseee&AN=edseee.4357531`.

Paul E. Debevec, Camillo J. Taylor, and Jitendra Malik. Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, page 11–20, New York, NY, USA, 1996. Association for Computing Machinery. ISBN 0897917464. doi: 10.1145/237170.237191. URL `https://doi.org/10.1145/237170.237191`.

S. Jarusirisawad and H. Saito. Diminished reality via multiple hand-held cameras. *2007 First ACM/IEEE International Conference on Distributed Smart Cameras, Distributed Smart Cameras, 2007. ICDSC '07. First ACM/IEEE International Conference on*, pages 251 – 258, 2007b. ISSN 978-1-4244-1353-9. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edseee&AN=edseee.4357531`.

P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on, Computer vision and pattern recognition*, 1, 2001. ISSN 0-7695-1272-0. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edseee&AN=edseee.990517`.

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on*, pages 779 – 788, 2016. ISSN 978-1-4673-8851-1. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edseee&AN=edseee.7780460`.

Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. Ssd: Single shot multibox detector. 2016. URL `http://arxiv.org/abs/1512.02325`. To appear.

Yoshikatsu Nakajima, Shohei Mori, and Hideo Saito. Semantic object selection and detection for diminished reality based on slam with viewpoint class. *2017 IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct), Mixed and Augmented Reality (ISMAR-Adjunct), 2017 IEEE International Symposium on, ISMAR-ADJUNCT*, pages 338 – 343, 2017. ISSN 978-0-7695-6327-5. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edseee&AN=edseee.8088517`.

Matthew J Westoby, James Brasington, Niel F Glasser, Michael J Hambrey, and Jennifer M Reynolds. 'structure-from-motion'photogrammetry: A low-cost, effective tool for geoscience applications. *Geomorphology*, 179:300–314, 2012.

E. Kruijff, J.E. Swan, and S. Feiner. Perceptual issues in augmented reality revisited. *2010 IEEE International Symposium on Mixed and Augmented Reality, Mixed and Augmented Reality (ISMAR), 2010 9th IEEE International Symposium on*, pages 3 – 12, 2010. ISSN 978-1-4244-9345-6. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edseee&AN=edseee.5643530`.

R.L. Gregory. Eye and brain: The psychology of seeing. *World University Library*, pages 11–12, 1973.

James Cutting. *Reconceiving perceptual space*, pages 215–238. 01 2003.

M. Dengler and W. Nitschke. Color stereopsis: A model for depth reversals based on border contrast. *Perception and Psychophysics*, 53(2):150 – 156, 1993. ISSN 0031-5117. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=psyh&AN=1993-32074-001`.

Jon M. Sundet. Effects of colour on perceived depth: Review of experiments and evaluation of theories. *Scandinavian Journal of Psychology*, 19(2):133 – 143, 1978. ISSN 0036-5564. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=psyh&AN=1980-00190-001`.

Authors Bailey, Reynold Grimm, Cindy Davoli, Reynold Bailey, Cindy Grimm, and Christopher Davoli. The real effect of warm-cool colors. 05 2006.

M. Carr Jr. Payne. Color as an independent variable in perceptual research. *Psychological Bulletin*, 61(3):199 – 208, 1964. ISSN 0033-2909. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=pdh&AN=1964-07115-001&site=ehost-live`.

Robert P. O'Shea, Shane G. Blackburn, and Hiroshi Ono. Contrast as a depth cue. *Vision Research*, 34(12):1595 – 1604, 1994. ISSN 0042-6989. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=psyh&AN=1995-08185-001`.

Tai Nan-Ching and Mehlika Inanici. Luminance contrast as depth cue: Investigation and design applications. *Computer-Aided Design and Applications*, 9(5):691 – 705, 2012. ISSN 16864360. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=bth&AN=88857652`.

Tiffany D. Do, Joseph J. LaViola, and Ryan P. McMahan. The effects of object shape, fidelity, color, and luminance on depth perception in handheld mobile augmented reality. *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR), Mixed and Augmented Reality (ISMAR), 2020 IEEE International Symposium on, ISMAR*, pages 64 – 72, 2020. ISSN 978-1-7281-8508-8. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=edseee&AN=edseee.9284799`.

Gill Barequet and Micha Sharir. Filling gaps in the boundary of a polyhedron. *Computer Aided Geometric Design*, 12(2):207–229, 1995. ISSN 0167-8396. doi: https://doi.org/10.1016/0167-8396(94)00011-G. URL `https://www.sciencedirect.com/science/article/pii/016783969400011G`.

Ming Zou, Tao Ju, and Nathan Carr. An algorithm for triangulating multiple 3d polygons. *Computer Graphics Forum*, 32(5):157 – 166, 2013. ISSN 01677055. URL `https://elib.tcd.ie/login?url=https://search-ebscohost-com.elib.tcd.ie/login.aspx?direct=true&db=bth&AN=89806362`.

Unity vr overview. URL `https://docs.unity3d.com/2018.2/Documentation/Manual/VROverview.html`.

Vuforia object scanner, c. URL `https://library.vuforia.com/articles/Training/Vuforia-Object-Scanner-Users-Guide.html`.

C# matrix library. URL `https://www.codeproject.com/Articles/19032/C-Matrix-Library`.

Bonnie E. John and Steven J. Marks. Tracking the effectiveness of usability evaluation methods. *Behaviour & Information Technology*, 16(4-5):188–202, 1997. doi: 10.1080/014492997119789. URL `https://doi.org/10.1080/014492997119789`.

C. Wharton, C. Lewis J. Rieman, and P. Polson. The cognitive walk-through method: A practioner's guide. URL `https://www.colorado.edu/ics/sites/default/files/attached-files/93-07.pdf`.

Exploratory burr holes. URL `https://operativeneurosurgery.com/doku.php?id=exploratory_burr_holes`.