國立臺灣師範大學
資訊工程研究所碩士論文

指導教授：張鈞法　博士

於擴增實境中虛擬物件的光照與陰影之擬真

Lighting and Shadowing for Virtual Objects in Augmented Reality

研究生：　陳凱恩　撰

中華民國　108　年　8　月

摘要

於擴增實境中虛擬物件的光照與陰影之擬真

陳凱恩

隨著硬體設備的型態變得輕巧、支援更多偵測功能以及系統分析的提升，近幾年擴增實境普遍的在行動裝置上應用，人們藉此體驗真實生活裡不存在的情境、和非現實的生物互動，亦或是作為工具測量真實物品、模擬室內擺設，因此越貼近真實生活的應用，虛擬物體與真實環境的不協調問題越加明顯，虛擬物體實質上仍存在於虛擬世界，獲取真實環境數據的多寡直接影響虛實的融合程度，近代攝影技術搭配機器學習試圖即時的從環境中獲取更多環境資訊，但分析一部分的環境畫面與真實世界裡無數光線所累積而成的光照相比，仍舊可以從虛擬物體的陰影與表面發現破綻；而非即時的環境分析雖然會受變化的程度降低靜態數據的效果，但是以較全面的環境圖像補充缺漏的資訊，得以在繪製過程中考慮到整個真實環境的影響。

本研究探討如何讓虛擬物體以合理之姿出現在現實畫面中，透過 ARCore 即時分析環境以及重構虛擬世界，並以 OpenGL ES 銜接空間資訊繪製模型，採用環景圖補充缺漏的環境數據，在即時繪製下即使拍攝範圍不足，仍可以計算整個環境對虛擬物體的影響，並進一步提升真實感。

關鍵字：擴增實境、ARCore、環景圖

ABSTRACT

Lighting and Shadowing for Virtual Objects in Augmented Reality

by

Kai-En Chen

With the development of mobile devices, the augmented reality becomes more common in our daily life. While augmented reality is emerging as one of key technology in life, the problem of combination of virtual objects and real world becomes obvious. The augmented reality system must detect the real objects within the environment. Then, people can interact with this enhanced version of reality.

In recent years, developers have tried to analyze environment in real-time. However, based on the camera of mobile devices, it is hard to consider the whole environment lighting from the partial view. Therefore, we use panorama images to get the complete light estimate and improve the reflection effects.

In this paper, we analyze equirectangular panorama images and store them as offline data about environment lighting. Then, we detect the environment in real-time and reconstruct a virtual world by ARCore. Finally, we use OpenGL ES to draw the models from those data and the perception of virtual objects can be improved.

Keyword: Augmented Reality, ARCore, Equirectangular Panorama

目錄

附表目錄

附圖目錄

第一章　緒論

1.1 研究背景

AR(Augmented Reality)又稱為擴增實境，在1901年作家 L. Frank Baum 提出以電子展示覆蓋真實生活的相似概念，直到1992年『Augmented Reality』正式成為專業術語。早期 AR 的相關研究就不斷進行著，直至近幾年才有較高的討論度與廣泛的應用，其中硬體設備和系統是最主要的兩大因素。

AR 發展之初是為追求在現實世界中體驗虛擬世界，當時研發走向更趨近 VR（Virtual Reality，虛擬實境），以頭戴顯示器呈現虛擬影像，投影系統與照相機結合將陰影投射在螢幕上，藉此讓人產生虛實空間交互作用的感覺，而這個時期大多用在軍事或各項研究中，並未廣泛應用；到了下一個發展階段 AR、VR 產生明顯的發展差異，1994年 AR 第一次應用在藝術領域上，舞台劇《Dancing in Cyberspace》中觀眾可以看到現實的表演者在虛擬的環境裡穿梭，藉虛擬物件強化了整體戲劇效果，這是 AR 繼正名後完整演繹其核心概念的展示，之後 AR 在各領域的發展更加活躍，其趨勢轉往追求互動性與視覺質感，尤其在遊戲中各方面都明顯的提升。

隨著偵測現實世界的技術演進，先是誕生 AR 卡、特殊條碼、特定形狀等具有標誌性的真實物體，藉由標的物觸發系統並顯示相應的虛擬物件。另一款積木遊戲《 FUSION 系列積木》則是玩家可以在現實世界任意組裝積木，經系統辨識後載入虛擬世界中；一直到近代辨識技術上有明顯突破，不需要標的物即可辨別目前拍攝的場景，遊戲《Pokemon GO》不僅加入虛擬草植美化、凸顯遊戲主題，遊戲過程中可發現精靈與裝置有了相對位置，距離的遠近也會影響捕獲的機率。

在電腦圖學的繪製概念中除了虛擬物件自身的材質特性外，若能掌握更多光源、位置等有關環境的資訊，會使虛擬物件更加逼真，當 AR 結合電腦圖學的繪製方式，裝置不單單只有顯示一個圖像，而是產生虛擬物件受到環境影響的錯覺，更是加快了現實世界與虛擬世界融合的腳步。

現今行動裝置的效能不只足以展示 AR，也支援電腦圖學部分繪製方法的計算；電腦圖學的發展中大概可分為兩種繪製方式，第一為光線追蹤(Ray Tracing)，以接近物理方式模擬光線在空間中和物體碰撞的效果，模擬光行進的路線越多，繪製結果會越加逼真，相對的計算量也更加龐大；第二為光柵化繪圖(Rasterization)為傳統的繪製方式，其犧牲物體的全域性、環境反射等效果簡化計算過程，加速繪製時間，雖然受限於設備條件與繪製方法，但我們可以藉傳統光柵化繪圖跟預先處理資訊，將環境資訊事先存成另一種形式，並在繪製過程中直接導入數據，盡可能的縮短繪製時間的同時也能展示更多環境的影響。

1.2 研究目的

我們在遊戲《Pokemon GO》中可觀察到背景為現實環境，前景為精靈，透過預設光照增加了精靈的立體感，但真實性卻不高，因為精靈沒有受到現實空間的影響；近期遊戲再發佈了新功能--GO Snapshot，就是針對現實世界做更深入的分析，尤其偵測出實際平面最為關鍵，前期遊戲中精靈的位置隨意擺放或漂浮在畫面中，因此產生虛實世界分離的突兀感；取得真實平面後，精靈依照偵測到的平面高度擺放、還原精靈的體態並加入陰影，確實有比較真實，但格格不入的衝擊感並未消除，虛擬物件與實際物件如同強塞在同一個畫面的兩個

平行空間，在電腦繪圖的概念中不只物體的特性，還要掌握環境中的光照條件，才能大幅提升真實感，而目前大多應用中以單光源或單一方向照明，使虛擬物件和真實環境的光照調性差太多；另一方面陰影也未能展現更多特性。

本研究將結合 AR 與 OpenGL ES，藉系統的辨識分析功能以及環境資訊的預先處理，彙整更多現實環境對虛擬物件的影響因素，以行動裝置展示虛擬物件的真實度與虛實的混和效果。

1.3 論文架構

本論文共有五章，第一章為緒論，主要介紹擴增實境的崛起以及近幾年在行動裝置上較主流的繪製方法；第二章為文獻探討，講述近期世界對於擴增實境的支援功能，探討目前虛實融合的程度與技術層面；第三章為研究方法與實作，介紹研究設備、環境資訊的分析與型態轉換以及實作方法；第四章為實驗結果，比較其他支援功能的繪製差異，第五章為結論與未來方向。

第二章　文獻探討

2.1 ARCore

由古至今虛擬物件從直接畫在螢幕上到根據立體空間、環境資訊才畫到現實畫面上，有此演變的最大推力是對環境的理解越加深入，Google 發佈的 ARCore 以三個要點掌握現實環境並整合虛實空間[1]，分別為運動追蹤、環境解析及光照評估。

（一）運動追蹤

指的是紀錄手機在現實空間的位置與運動方向，以 VIO(Visual Inertial Odometry)技術取得相機的平移量與旋轉角度等重要數據，定義裝置在虛擬空間裡的位置與旋轉角度；追蹤的最大用意在於固定虛擬物件的位置，現實世界中，物體不會隨人眼變換位置，同理虛擬物件對應到現實世界中不應該隨意的變動位置，即使移動觀察為置使得虛擬物件超出螢幕範圍，當視角返回時虛擬物體仍舊在原處。

（二）環境解析

偵測現實世界的物體、平面進而重構空間。Google 在較早之前開發的 Tango 是以紅外線探測環境，但實際上在現實世界操作時，反而因障礙物的阻擋與探測距離的限制無法完整感知空間資訊；改用 SLAM(simultaneous localization and mapping)就可以打破這個困難，此為 Smith, R. C.和 Cheeseman, P.在1986年提出定位同步化與地圖建構的概念[2]，通過重複偵測物品特徵確立環境的樣貌，達到定位和建構虛擬空間的目的；自攝影畫面中辨識物品的不同之處，並為其標記特徵點組成雲集，最後整合所有的特徵點雲與相機的位置，

使用三角定位法取得特徵點的深度值，掌握的數據足以撐起一個立體空間，重構出與現實世界相應的虛擬世界。

（三）光照評估

分析畫面中光的顏色與明暗對比的連續性找到光的顏色與強度，其概念與光學相關，提高偏移量的信任度，調和虛擬物件的光照調性，使虛擬物件可以更接近環境的照明程度。

相較於以往的 AR 系統，現代結合了許多領域的技術支援，以平面影像的分析就能建立立體空間，藉由硬體設備獲取位置與運動軌跡，甚至開發者需要佈署的架構、導入其他支援功能等繁瑣過程大多已被內化、包裝，降低了開發的困難度，且 ARCore 在 Android、Unity、Unreal 及 iOS 等不同系統開發，也使硬體設備的選擇更加多樣。

2.2 OpenGL ES 傳統光柵化繪圖 Rasterization

OpenGL ES 自 OpenGL 簡化而來，主要因素是 GLES 是針對嵌入式系統，像是手機等行動裝置。由於設備條件不同造成效能的差異，無法支援 OpenGL 所有的功能，因此在這類設備上運行的支援系統進展速度相對較慢，即使如此 GLES 仍有保留主要的繪製功能，在有限的資源下還是可以高效完成繪製。

電腦圖學中基礎的繪圖方法是傳統光柵化繪圖，經由 OpenGL 提供3D 圖形應用編程介面以及管線化繪製過程，不只降低了開發困難度也以硬體加速提升了效能。OpenGL 中包含了在 CPU 執行的主體程式以及在 GPU 執行的 GLSL 語言，再透過兩個階段完成光柵化繪圖，第一階段將三維座標轉成二維座標，從模型檔中可得到一系列頂點座標及其他配套的頂點資訊，傳入頂點著色器(Vertex Shader)計算頂點在不同空間系的座標、法向量、貼圖座標等，而這些都

會成為下一階段的重要數據，爾後經過圖形建構使三維座標轉換為二維平面圖；第二階段將二維平面圖上色並對應到螢幕，自上階段取得頂點的著色數據傳入片段著色器(Fragment Shader)結合視角與光源內插出目前像素的顏色。

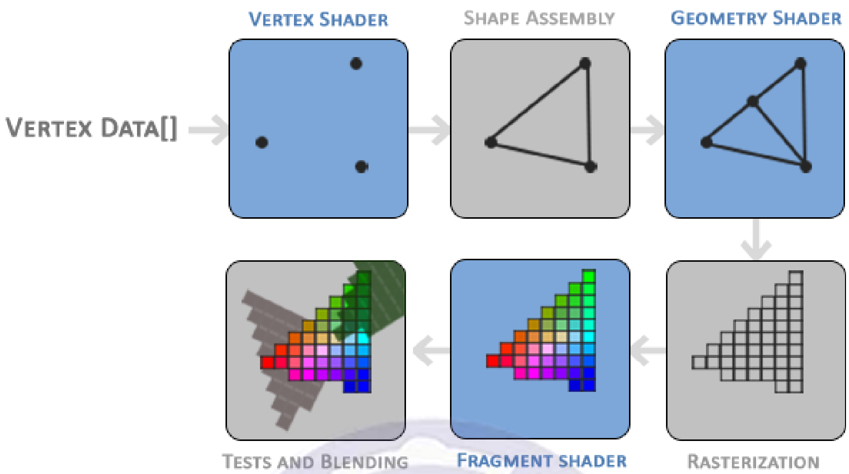


圖 1：GLSL pipeline 運作流程[3]

2.3 Filtering Environment Illumination for Interactive Physically-Based Rendering in Mixed Reality

虛擬物件的逼真程度往往在物體的光照程度、反射與陰影備受考驗，由 Soham Uday Mehta 和 Kihwan Kim 等人發表的研究中[4]以混何實境(Mixed Reality，簡稱 MR)展示相當高的虛實融合度。

MR 中包含 AR、VR 的概念與技術，先用 SLAM 整合虛實空間，同時自攝影畫面中分析現實環境的光照資訊，接著以蒙地卡羅光線追蹤採樣模擬光線行進，通常會遇到的困難是即使空間對應，實際物體的顏色或環境資訊也無法完全納入虛擬世界中計算，採用光線追蹤相當於光在虛擬與現實中穿梭，攜帶了虛實物體的貢獻度，展現兩物體或多個物體之間的能量可以相互傳遞、影響，使光照程度與陰影更接近現實狀況，雖然光線追蹤的繪製速度慢，但在虛實空

間的整併下起了很大效用，此特性也成為提升融合度的最大優勢，不只在虛擬物體上可見直接照明與間接照明的效果，現實世界中的物體也接收到虛擬物體的反射貢獻。

圖 2：真實世界的物品[4]

圖 3：於真實世界加入虛擬物品[4]

光線追蹤將環境資訊發揮得更加徹底，使環境因素能充斥每個追蹤的計算過程，而我們知道光的路線複雜且模擬數量並非無上限，因此在研究中使用蒙地卡羅以機率的方式決定呈現的顏色，之後利用濾波器將顏色撫平，同時也展現陰影或光照的柔和，於此才算是完成了整個繪製過程，如圖3展示的結果。

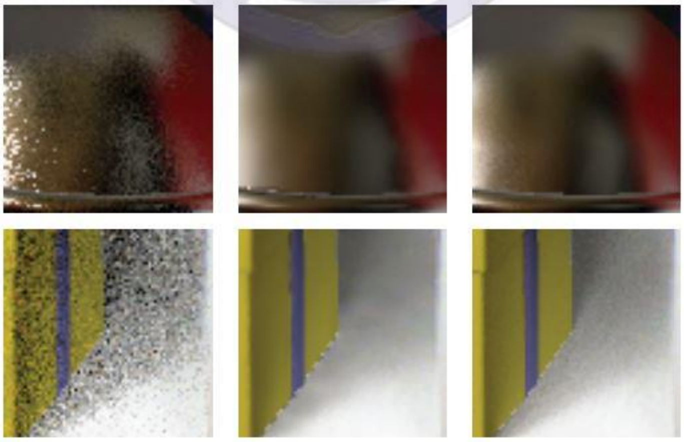


圖 4：(左)輸入蒙地卡羅的結果；(中)此研究的過濾效果；(右)參照結果[4]

2.4 基於圖像照明 Image Based Lighting

基於圖像的照明一是種簡化的照明方法，將環境圖視為一個大光源，圖上的每一點皆貢獻不同強度的照明，使得虛擬物體在不同環境也能呈現相應的光照，如圖5，我們期望可以得到比較完整的光照效果，將圖像中的每個像素都視為光源貢獻顏色與強度，但全都納入考量無疑會造成龐大的計算量，因此在許多研究中提出不同的方法預先處理環境貼圖以加速計算過程，而常見的其中一種方法 Spherical Harmonic Lighting(球諧函數照明)[5]，將圖像轉為訊號以數學的方式將大量的資訊轉為函數簡化繪製的過程。

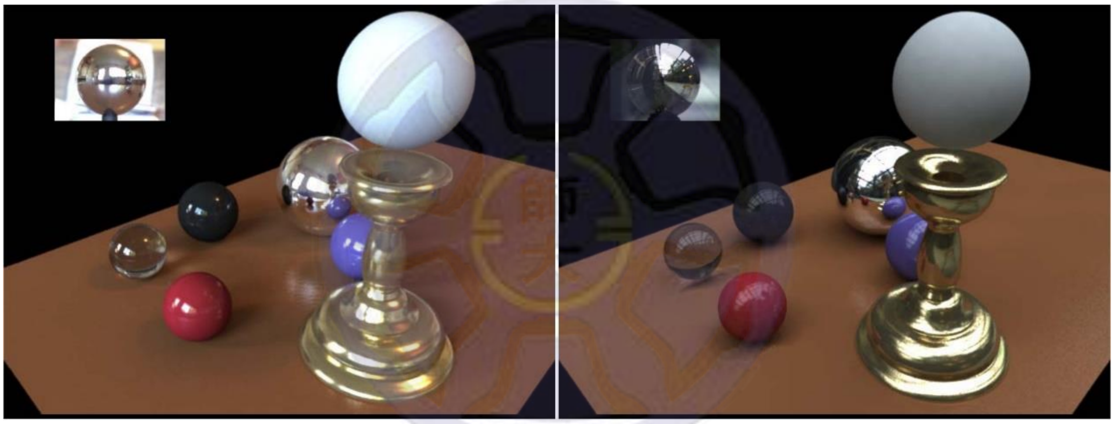


圖 5：IBL 展示不同環境貼圖的照明[6]

近期 Google 發佈了 ARCore 新的光照評估模式 Environmental_HDR[7]，在之前的評估模式中 AMBIENT_INTENSITY，只能知曉光的顏色以及強度，而新模式提供了光的強度、方向、球諧係數(Harmonics Coefficients)以及立方體貼圖(CubeMap)，從圖6中可以了解這些資訊帶來的效果。

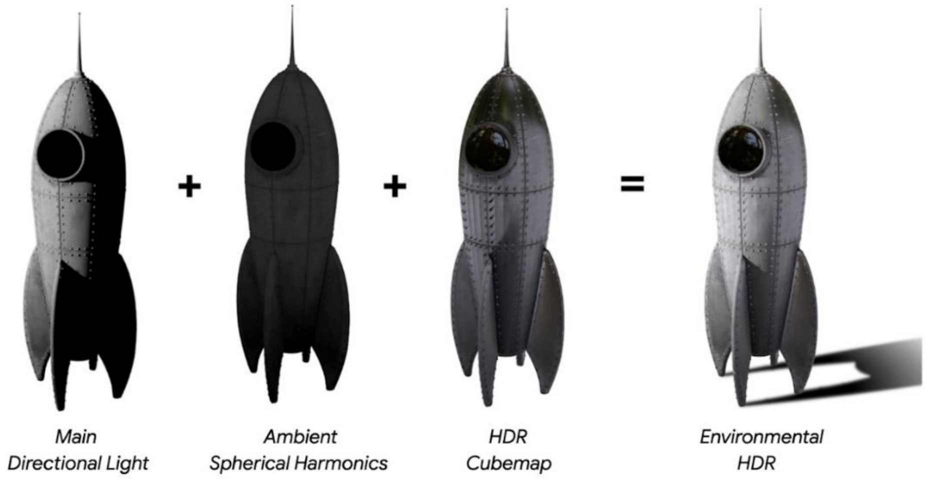


圖 6：ARCore-Environmental_HDR 模式可提供的效果[7]

光的強度和方向使物體表面產生明暗變化，且漸變方向逼近實際狀況；球諧係數是經由機器學習分析目前拍攝的畫面，即時獲得的光照係數；HDR 立方體貼圖則是取自真實世界的環景圖像，使光滑表面的物件可以呈現反射效果；將這些效果整合後虛擬物體就能更貼近現實環境，下列圖7展示新功能在不同空間中測試光照效果。

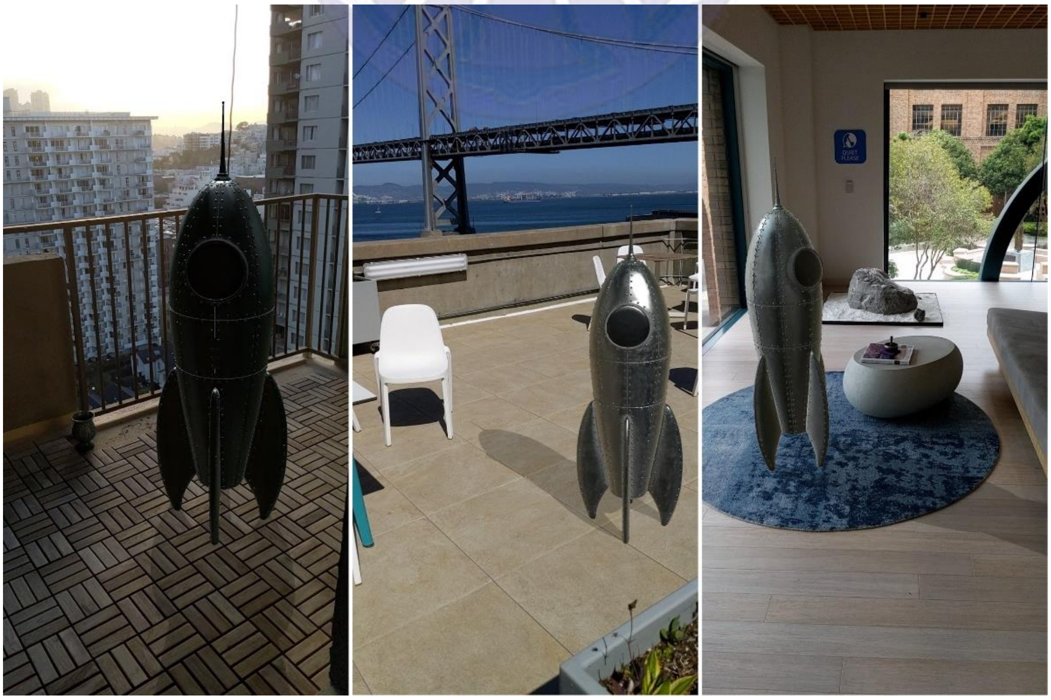


圖 7：Environmental_HDR 模式在不同環境的展示[7]

2.5 360環景圖

環景圖(Equirectangular Panorama)如同包裹球體的表皮經過投影後形成的二維平面圖，圖像的頂端與底端會匯集到球體的上下兩極，因此展開後就會如圖11標示的上下兩端有有拉伸現象，繪圖過程中常用的兩種貼圖為球體映射(Sphere Mapping)和立方體映射(Cube Mapping)，在現實中物體不僅反映自身的色彩，也會反射周圍的景物，而以一個球體或天空盒(SkyBox)包覆模型模擬反射效果，以圖8可以了解環景圖在不同的應用中轉換為不同的表示方法計算。

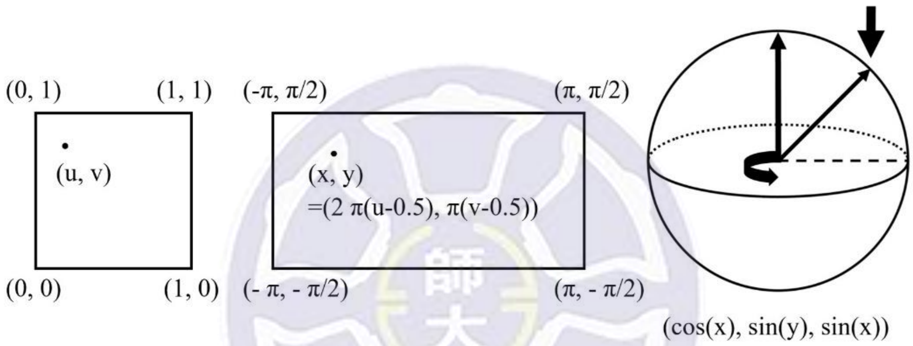


圖 8：環景圖轉換關係[8]

以貼圖環繞模型呈現環境反射早在1976年的研究中[9]就被提出，將矩形環境貼圖反射在茶壺的表面，如圖9所示，當時攝影技術尚未成熟，雖然僅能使用一般的平面圖作為環境資訊，但已然成為驅動更全面的環境反射之動力。

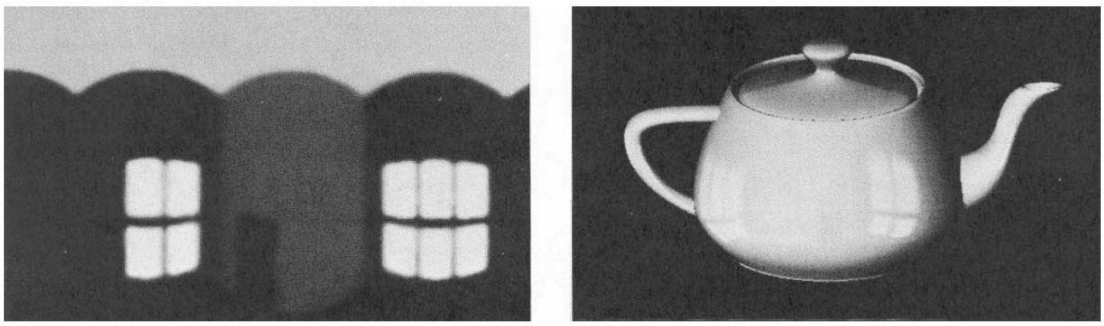


圖 9：電腦繪製平面貼圖反射在茶壺上[9]

早期的環景攝影是在真實環境中放置鏡面球體繪製球體映射圖，球體的外觀即為所需的紋理資訊，但是受到觀察視角的限制，實際上無法捕捉到完整環境。立方體貼圖的生成相對簡單，六個方向分別對應六個面以90度錐狀體向外延展進而捕捉某一面的環境畫面，之後將六個畫面拚在一起呈現環境圍繞效果。隨著對環境資訊的需求增加，拍攝角度也逐漸增加；Paul Haeberli[10]利用魚眼鏡頭可拍攝達180度的範圍，朝多個方向拍攝再加以拼接產生完整的環景圖，如圖10。

圖 10：(左圖)180拍攝結果；(右圖)拼接兩張180圖像的360圖像[10]

到了360相機的誕生，使用一組雙鏡頭拍攝角度擴展到360度，透過儀器校對並拼接兩個方向的圖像，並以圖11般變形的平面圖保存，本質上這張圖仍是球體貼圖，也可以轉換為立方體貼圖的型態，不論何種形式皆保留了我們需要的資訊。

圖 11：360環景圖

2.6 陰影柔和 Soft Shadow

通常透過影子我們可以判斷物體的位置、高度或物體特性，而近代發展於虛擬世界中加入陰影，使虛擬物體看起來更加真實，因此陰影成為提升視覺質感的重要角色。陰影本身又分為全影(Umbra)跟半影(Penumbra)，產生一個完全的全影表示光源要是一個無限小的點光源，但現實中這樣的光源並不存在，我們比較常見的是全影跟半影混和的陰影，於全影的邊緣處會與半影區域漸進式轉換，這是由於光具有面積、體積等因素影響半影區域的大小。而常見的繪製陰影方法有陰影映射(Shadow Mapping)與陰影錐(Shadow Volume)；前者自光源打出光線，記錄光通過物體之後照在平面上的深度值，當物體互相遮蔽時就會有深度差值，就可以用來判斷目前位置是不是在陰影區，概念上直接而快速，但會有邊緣鋸齒狀的問題；後者以錐體範圍跟眼睛位置判別物體是否在陰影中，雖然邊緣平滑且支援多光源，但速度相對慢。

而在 Eric Chan 和 Fredo´ Durand 的研究中[11]，以陰影映射繪製陰影結合陰影錐的概念繪製柔和的陰影。首先取深度圖但不是用來做深度比較，而是辨識深度圖中每個物體的外輪廓，並朝外擴展一圈作為半影的範圍，如圖12(b)；之後在這個區域以相似於陰影錐的方法判別光、遮擋物和接收位置的距離比，得到的比率即為調整透明度的 Alpha 值，在圖12(c)中可以看到輪廓外的部分具有漸層效果向外擴散，最後比較深度值與 Alpha 值決定陰影的模糊程度，由此產生柔和邊緣的效果。

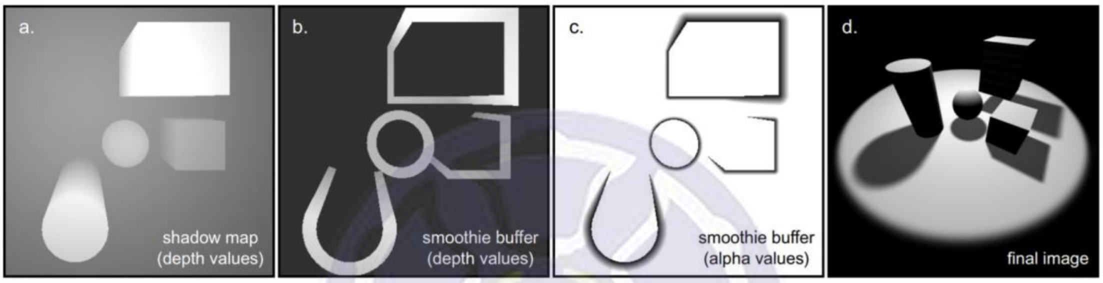


圖 12：(a)陰影映射繪製的深度圖；(b)影像辨識(a)圖後定義的模糊範圍；(c)以距離比計算 Alpha 值；(d)整合深度圖與模糊程度繪製陰影[11]

陰影映射繪製半影的品質相對粗糙許多，所以融合陰影錐的概念與其研究方法改善結果，但我們還需考慮在多光源的場景中能否即時繪製，且不論輪廓辨識與陰影錐判別需要花費的計算時間，在陰影映射的實作條件下，基本上一張深度圖僅能由一個光源繪製，意即若在多光源的場景下就會有多張深度圖，每當更新深度圖時後續的所有辨識與判別都要重新計算，而2006年由 P. Supan 等人的研究中[12]則是疊加多張陰影映射模擬出柔和邊緣，在虛擬擴增實境中利用環景圖作為多光源的依據，如圖13中將環景圖切割多個區域，採集多達64個不同強度、方向的光源。

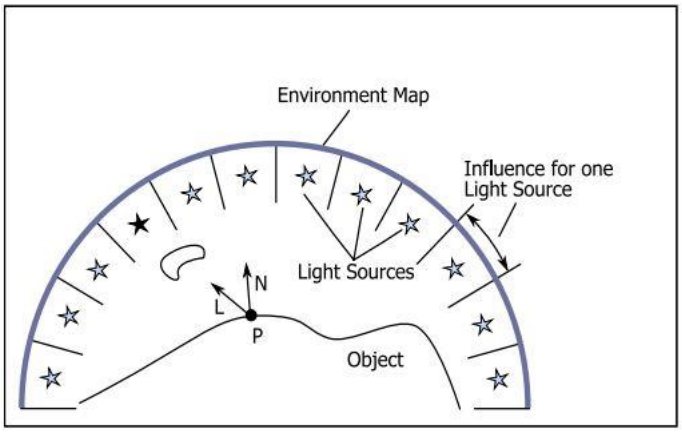


圖 13：將環景圖做為球體貼圖，切割不同區域作為光源[12]

為了提升真實度進一步更新深度圖，考量更新的必要性以及即時繪製時可容許的更新數量，避免更新過多的深度圖降低效率，因此系統僅可在每一幀重繪一張深度圖，圖15中可了解每組映射貼圖經過多次疊加後做模糊處理，最後再與其他效果結合。

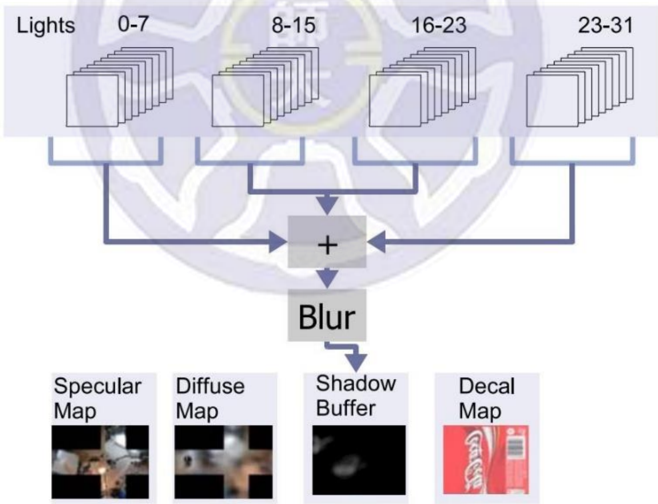


圖 14：每組陰影映射的處理架構圖[12]

成果中展示取多個來自不同方向的陰影，堆疊後較飽和的部分是本影區，本影邊緣以外漸漸淡出的部分則為半影，隨著光源採樣數量的增加，可以改善陰影數量不足致使可見度低的問題，圖16展示隨著疊加數量提高後茶壺彎嘴的陰影輪廓變得更加明顯，但並非越柔和就越逼真，在現實環境中若僅有一個光

源且亮度夠強，則實際影子的半影區會比較小，此時虛擬物件的陰影若過度柔和就會和實際狀況產生衝突。

圖 15：(左)8層陰影；(中)48層陰影；(右)64層陰影[12]

第三章　研究方法

本章於第一小節介紹實作環境及設備，其他小節將依序說明研究方法。

3.1 實驗環境

3.1.1 AUSU ZenFone Ares

本實驗使用 AUSU ZenFone Ares 為測試機，其原因為此機型通過 Google 的驗證，確保裝置具有強大的 CPU、夠高的相機畫素、手機本身的運動傳感器以及系統相容性，而這些都是促使 ARCore 可以順利運行的因素。

作業系統	Android™ 7.0
處理器	CPU: 64 位元 高通® 四核心處理器 Snapdragon™ 821 @2.35Ghz GPU: Adreno™ 530
記憶體	RAM 8GB
相機	後鏡頭：23 百萬畫素，f/2.0 大光圈，六鏡式鏡片

表 1：實驗室測試設備

3.1.2 ASUS 360度相機

作業系統	Android 5.0 以上
相機	雙 210°魚眼鏡頭：500 萬像素
USB 連接埠	Micro-USB OTG，Type-C

表 2：實驗室攝影設備

3.2 環景圖前處理

360環景圖中雖然沒有具體的三維資訊，但保有大部分的光照資訊，可以再轉換後得到三維資訊，此階段我們將環景圖視為球體，在球體上找到光的分佈作為光的照射方向。

首先圖片會以 RGB 的格式儲存顏色，若直接用 RGB 判別會得到不精確的結果，如圖16，所以採用 HSL(Hue, Saturation, Lightness)，分別為色相、飽和度及亮度，透過亮度值可以比較準確的篩選出強度高的部分，圖17則展示加入強度判別的結果。

圖 16：使用 RGB 偵測

圖 17：使用 HSL 偵測

接下來考慮合適的規格與模糊程度得到較佳的偵測結果，以原圖的大小1920×3840，可以取得很多較精確的光源細節，但也容易將小範圍的反射誤判

為光源，且納入所有座標會使計算時間冗長，因此以等比率縮小圖像到適當大小降低過濾時的資料量，本實驗採192×384的規格，透過高斯模糊過濾掉細節同時保留夠強的部分，藉此提高偵測光源分佈的精確度，使用不同大小的遮罩或標準差可以適度過濾掉較破碎或小範圍的高光反射，下列是在不同場景中的偵測結果。

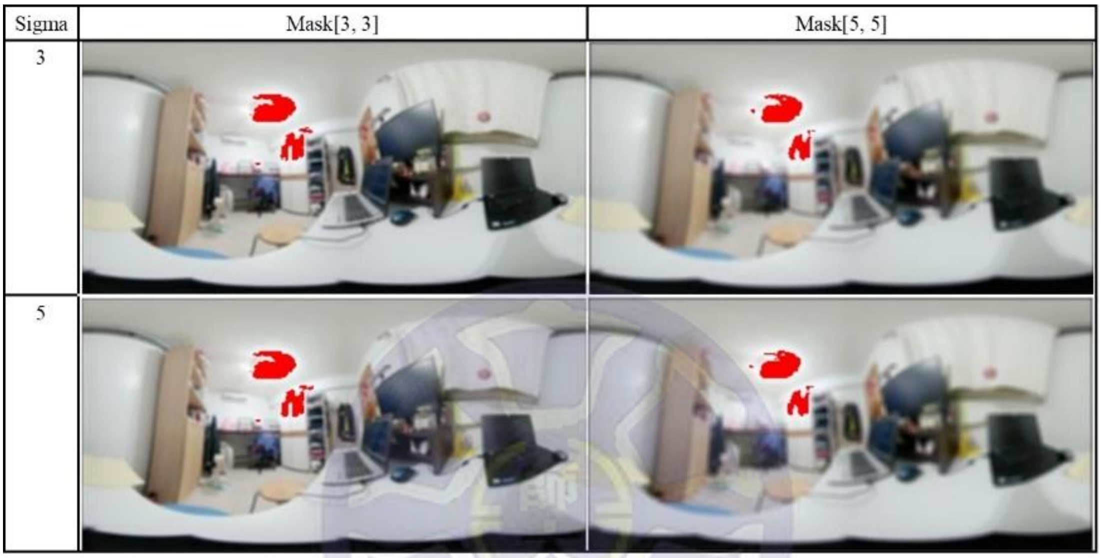


圖 18：場景光源集中

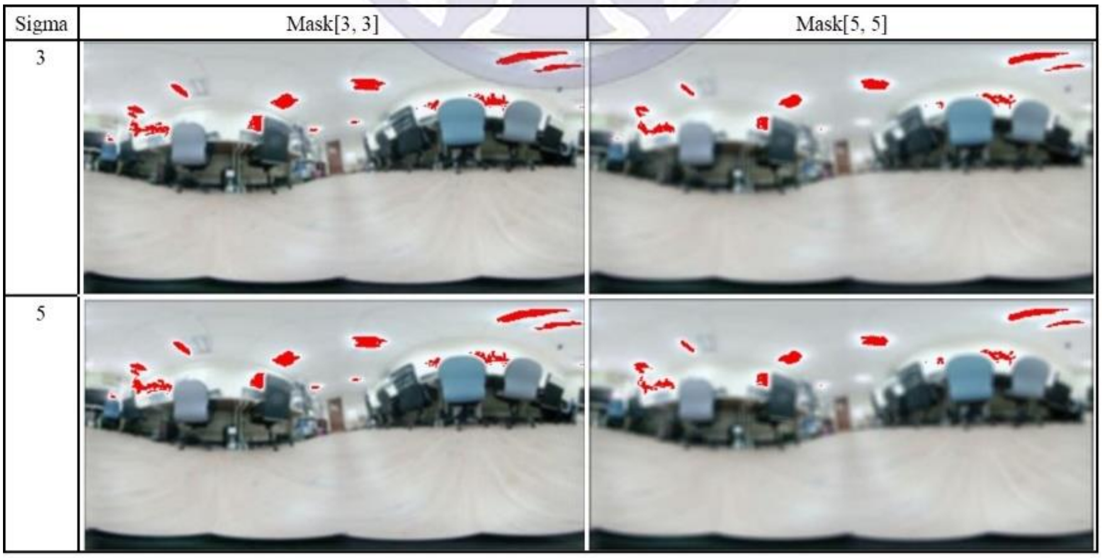
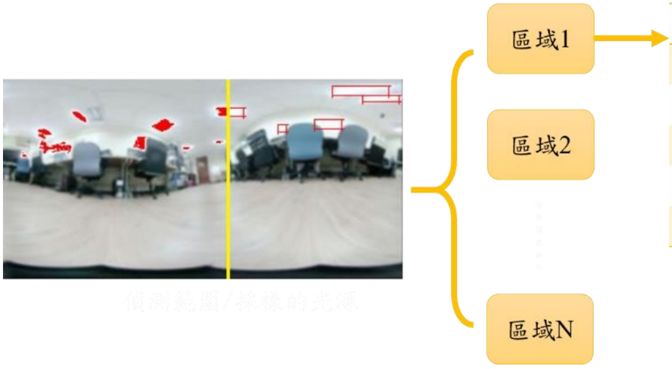


圖 19：場景光源分散

經過第一階段過濾得到多個座標，大部分座標分區集中，所以在第二階段以邊緣追蹤(Border Tracing)劃出每個區間的範圍，將各個範圍內光的座標及強度分別儲存。



i	j	Lightness
0.34375	0.6770833	0.9908981
0.4010417	0.765625	0.9905794
0.8958333	0.9375	0.9904697
0.5677083	0.8333333	0.9903342
0.1927083,	0.8072917	0.9901962

圖 20：光源劃分區域儲存亮度示意圖

最後一個階段，自每個區域定義一個光源方向，根據場景的條件採不同的定義方法，若是光源區域數量少於五的場景，將如圖21所示將每個區域評分四塊再互相比較一次亮度後取其中心點；反之在光源較多且分散的場景中，則不分割區域集合，比較亮度後直接取中心位置，如圖22的示意圖，從這些偵測結果中取前五個平均亮度最高的作為光源方向。

圖 21：場景光源區域偏少，定義光源方向示意圖

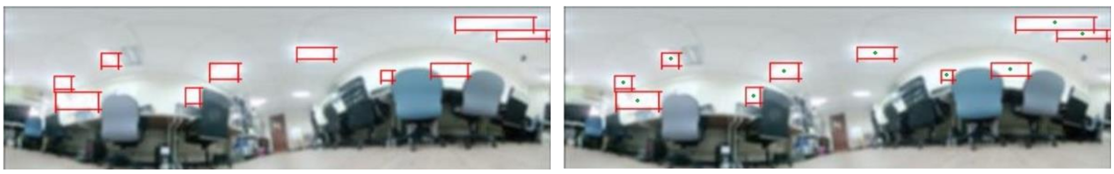


圖 22：場景光源區域偏多，定義光源方向示意圖

3.3 環景圖的轉換

我們所見的環景圖是以二維平面圖的形式呈現，但通常繪製過程中每個參數皆具立體空間的含意，面對不同的計算需求我們將圖像轉為適當的形式取得我們需要的表示法。

3.3.1 光源位置

自上一個階段我們得到光在平面上的座標(i, j)，此型態無法代入虛擬空間中運算。在 OpenGL 繪製過程中使用的圖片素材不論原始大小，進入系統後都會壓縮成範圍0～1的 UV 貼圖，統一表示範為後除去了圖像實際寬高的影響，因此我們將篩選出來的平面座標(i, j)與圖像的寬高相除，使表示範圍轉至0～1之間，意即我們現在已知 UV 座標經轉換後可以求得單位球體上的三維座標。

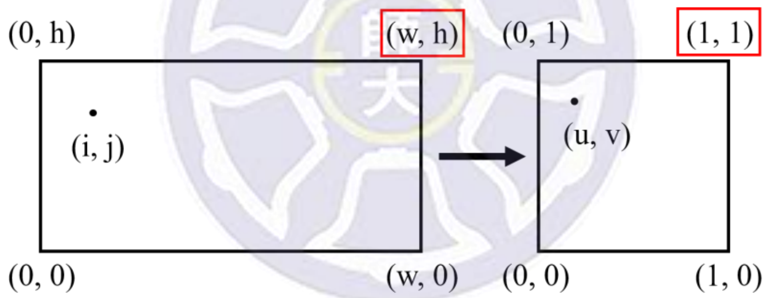


圖 23：環景圖轉 UV 貼圖

球體座標不僅有象限性質還有角度涵義，將 UV 座標各減去貼圖寬高的一半使0～1的恆正範圍改到正負表示的值域，並延伸貼圖的寬高2π 倍與 π 倍，即可符合球體展開後的平面，最後得到的(x, y)座標經三角函數換算後會對應到一個單位球體上的 P 座標，即為我們需要的光源座標。

$$P = (\cos(2\pi(u-0.5)), \sin(\pi(v-0.5)), \sin(2\pi(u-0.5))) \quad (1)[8]$$

3.3.2 環境反射

此小節與上一節的流程相反，我們期望利用環景圖補充裝置無法攝影的區

域，依據反射的概念光打到物體表面後入射角等於反射角，光沿著反射角度繼續行進直到碰撞下一個物體或進入人的眼睛，因此我們可以看到物體本身的顏色，若物體是光滑的表面，同時也能看到表面上呈現其他物件的影像，在光柵化繪製方法我們可以從眼睛與物體表面的發向量推出反射方向，此向量可視為對應到一個單位球體上的 P 座標，而在上一節的正推過程中了解 UV 貼圖與球體座標的轉換關係，所以逆推(1)後就可以得 (u, v)座標。

$$(u, v) = \left((\tan^{-1}\frac{P_z}{Px})\times\frac{1}{2\pi} + 0.5,\ (\sin^{-1} P_y)\times\frac{1}{\pi} + 0.5 \right) [8]$$

3.4 陰影柔和

陰影映射(Shadow Mapping)搭配光柵化繪製可以快速畫出基本的陰影，但每一張深度圖僅能由一個光源生成且經映射後會有鋸齒狀邊緣的問題，先以PCF(Percentage-Closer Filtering)將邊緣模糊，再多次繪製不同方向的深度圖，每層陰影都單獨完成，最後再彙整所有陰影；而觀察現實中的影子，即使在本影的範圍，仍舊透出下方物體的顏色，因此將陰影性質設為半透明，在多層陰影的疊加下中心的飽和度較高、邊緣較低。

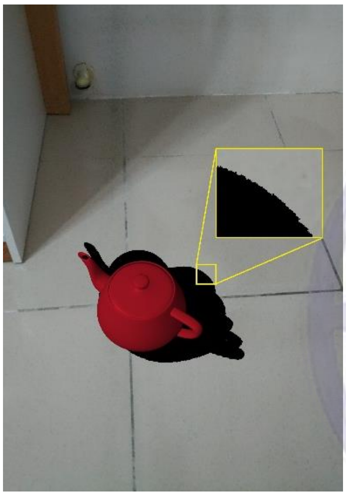


圖 24：陰影不透明

圖 25：陰影透明度為0.6，且邊緣模糊

圖 26：調整透明度為0.2

接下來我們要將陰影畫在實際偵測到的平面上，ARCore 的繪製流程中有幾個關鍵點掌握虛擬物件的定位，流程圖中當特徵點雲的量足夠時就可以確定平面的所在位置，系統依使用者點擊平面的位置標記錨點(Anchor Node)，而錨點就是虛擬物件平移的根據，左邊的原始流程中先更新模型矩陣(Model Matrix)再畫虛擬物件；右邊則是加入陰影的流程，理想的作法是將偵測到的平面直接繪製深度圖與映射計算，但實際上平面會一直更新或是一直增加，若有多個平面則深度圖就會畫很多次，於是我先預設虛擬物件下有一個夠大的平面，由此深

度圖的繪製次數就會減少到跟光源的數量一樣；到了陰影映射的階段，原本是以同一組模型繼續判斷陰影位置，而現在直接改用新的平面，雖然可藉錨點確定虛擬物件在平面上的位置，但是無法確定錨點標在哪一層平面，所以此部分還是以遍歷所有平面來確認。

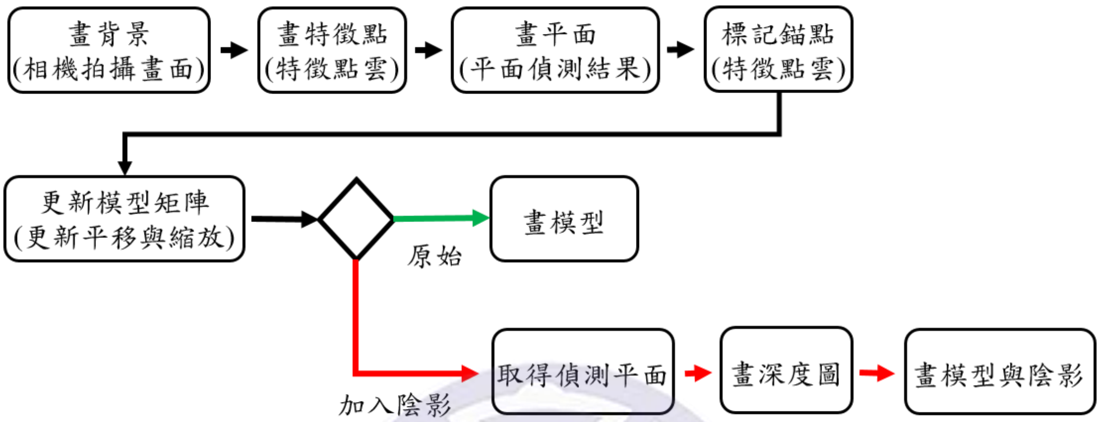


圖 27：陰影繪製架構圖，綠線為原始流程；紅線為更新流程

第四章　實驗結果

4.1 陰影效果

比較兩種光源的取樣方法，一為取光源集合的中位數；二為 ARCore 的 Environmental_HDR 模式提供主光源方向此部分將與光源過濾的資訊做延伸應用， 首先取主光源方向自過濾光源的資訊中選擇位置相近的其他光源，由於不同場景使偵測結果出現偏差致使無法找到附近的光源，則將轉為找較近且亮度較強的光源，圖28示意處理主光源的流程。

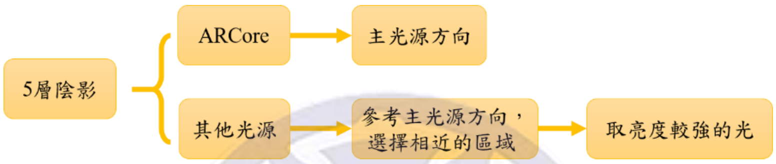


圖 28：ARCore 的 Environmental_HDR 模式延伸應用流程

在不同條件的場景中測試兩種方法的陰影效果，第一種如圖29場景光源較集中。由圖30的左圖展示結果中可發現光源相對分散，則是繪製陰影過程中採用定向光，不考慮光的面積而是光的方向，將區域中心視為光源方向的同時也打散了陰影的集中程度；圖30的右圖則是因為有主要光源主導陰影方向和其他篩選出來的光源整合後整體陰影的方向較一致。

圖 29：光源較集中的場景

圖 30：(左)我的方法；(右) Environmental_HDR 光照評估模式與其他光源

第二種場景如圖31，在圖32的左圖中可以發現我的方法使陰影方向比較分散，與真實寶特瓶的陰影方向相近，而目前 Environmental_HDR 模式僅能提供一個方向，因此與實際環境的光源數量差距太大或偵測方向偏差太多時陰影擬真的效果有限，也因光的方向來源較集中使模型的明暗對比強烈，如圖32的右圖。

圖 31：光源較分散的場景

圖 32：(左)我的方法；(右) Environmental_HDR 光照評估模式與其他光源

4.2 金屬反射效果

擴增實境執行的過程中，僅依靠前鏡作為解析環境的來源，模擬反射材質就會面臨缺失大部分環境資訊的問題，因此以環景圖補充資訊，藉金屬反射材質展示環景圖的輔助效果。

4.2.1 黃金茶壺

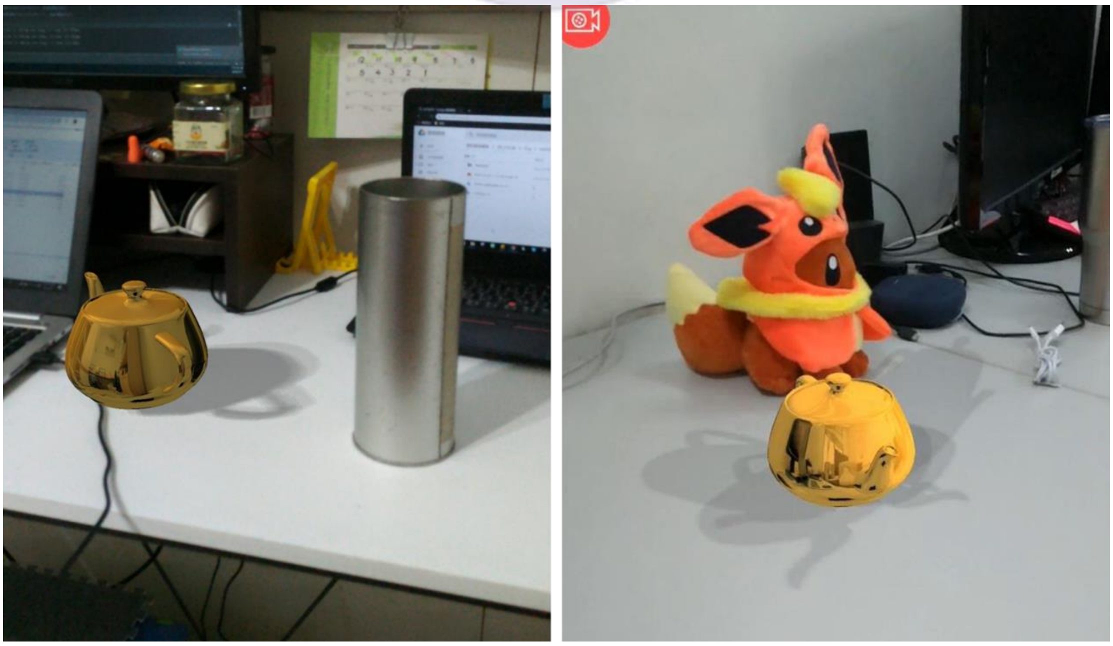


圖 33：(左)場景光源集中；(右)場景光源分散

4.2.2 鋁金屬機器頭模型

圖 34：(左)原始場景；(右)展示結果

4.2.3 鏡面反射球

環景圖在鏡面反射的輔助成效，圖35的左右兩圖分別在不同高度拍攝，而模型的放置高度也盡可能的相近於拍攝高度，若位置或高度差太多展示結果就會偏差很大，於此凸顯靜態環景圖的限制。

圖 35：(左)場景光源集中；(右)場景光源分散

4.3 多光源照明

4.3.1 文具

比較兩種光照效果，圖36左圖中僅以多光源照明，沒有考慮光的強弱；圖36右圖採 ARCore 的 Environmental_HDR 光照評估模式以機器學習即時分析的球諧係數調和光照程度，因此在細節上可以發現 Environmental_HDR 受到主光源方向影響與真實的筆相比似乎暗了一點，而我的方法中光源方向相對精準，明暗變化的方向與真實的筆相近。

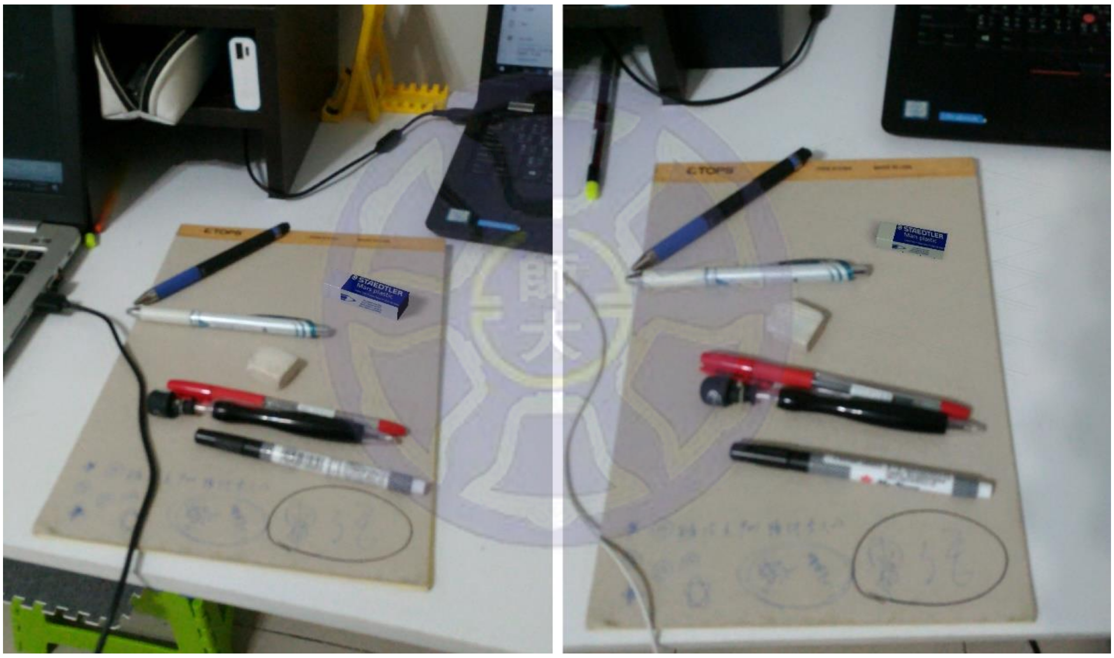


圖 36：(左)採多光源照明；(右)採多光源照明且加入 SH 係數調整。

第五章　結論與未來展望

本實驗模擬虛擬物體接收實際的環境資訊後所能展現的真實程度，使用 ARCore 即時偵測的環境數據，再加上靜態環景圖輔助相機當下攝影不到的範圍，在非即時繪製的階段分析靜態資訊，從環景圖中找到光源的分布；在即時繪製階段，將環景圖做為物體計算反射效果的依據，若能透過數學的方法更完善的處理整個環境的光照訊息，虛擬物體就可以呈現更細緻的光照變化，而相較於此，ARCore 的 Environmental_HDR 環境偵測模式，即使不使用環景圖也能偵測環境照明，但裝置與光源的遠近就會劇烈影響照明係數。

雖然盡可能的分析、運用靜態環境資訊，但如果是在光源劇烈變動的場景中，大部分的環境資訊則失去效用，這也是 Google 當前積極推出新功能的目的，即時分析動態場景得到更精確的資訊。

擴增實境發展至今其理念的實踐越趨完整，這當中整合了不同領域的技術，有硬體設備的運動偵測、影像辨識、空間中的三角定位或機器學習的即時分析等支援，集結了各項技術才獲得關鍵資訊，提升了對環境的掌握度，但仍存有一些侷限，未來若能整併360相機同步攝影整個環境，就能更全面的展現光照、位置以及虛實的交互影響。

文獻參考

[1] Google Developers, “ARCore Overview”, [Online] Available: https://developers.google.com/ar/discover/ [Accessed 1 2019]

[2] Smith, R. C., & Cheeseman, P. (1986). “On the representation and estimation of spatial uncertainty.” *The international journal of Robotics Research,* 5(4), 56-68.

[3] Joey de Vries, “Hello Triangle”, [Online] Available: https://learnopengl.com/Getting-started/Hello-Triangle. [Accessed 6 2019]

[4] Mehta, S. U., Kim, K., Pajak, D., Pulli, K., Kautz, J., & Ramamoorthi, R. (2015, June). “Filtering Environment Illumination for Interactive Physically-Based Rendering in Mixed Reality.” In *EGSR (EI&I)* (pp. 107-118).

[5] Green, R. (2003, March). “Spherical harmonic lighting: The gritty details.” In *Archives of the Game Developers Conference* (Vol. 56, p. 4).

[6] Debevec, P. (2008, August). “Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography.” In *ACM SIGGRAPH* 2008 classes (p. 32). ACM.

[7] Google Developers, “Updates to ARCore Help You Build More Interactive & Realistic AR Experiences”, [Online] Available: https://developers.googleblog.com/2019/05/ARCore-IO19.html [Accessed 5 2019]

[8] 陳佑欣. (2017). 虛擬實境中基於物理之擬真材質的繪製 (Masters dissertation).

[9] Blinn, J. F., & Newell, M. E. (1976). “Texture and reflection in computer generated images.” *Communications of the ACM*, 19(10), 542-547.

[10] Paul Debevec, "The Story of Reflection Mapping", [Online] Available: https://www.pauldebevec.com/ReflectionMapping/ [Accessed 3 2019]

[11] Chan, E., & Durand, F. (2003, June). "Rendering Fake Soft Shadows with Smoothies." In *Rendering Techniques* (pp. 208-218)

[12] Supan, P., Stuppacher, I., & Haller, M. (2006). "Image Based Shadowing in Real-Time Augmented Reality." *IJVR*, 5(3), 1-7.