

# Visual Computing Project: Augmented Reality for Anatomical Education

Javier Santiago Giraldo Jiménez, *Student Member, IEEE,*  
 Jesús Ernesto Quiñones, *Student Member, IEEE,*  
 Julián Andrés Córdoba Cely, *Student Member, IEEE,*  
 Miguel Angel Martinez Fernandez, *Student Member, IEEE*  
 and Brayan Manuel Rubiano Paramo, *Student Member, IEEE*

**Abstract**—Este proyecto detalla el desarrollo e implementación de una aplicación móvil de Realidad Aumentada (RA) diseñada para facilitar el estudio de la anatomía humana. La enseñanza tradicional basada en atlas 2D impone una alta carga cognitiva debido a la necesidad de abstracción espacial. Para mitigar esto, se desarrolló un sistema que utiliza visión por computadora para detectar marcadores planos (imágenes de órganos) y superponer modelos anatómicos 3D de alta fidelidad. La solución fue construida utilizando el motor Unity y el SDK de Vuforia, tras descartar soluciones nativas debido a inconsistencias técnicas. El sistema despliega cinco modelos anatómicos en formato FBX, permitiendo a los estudiantes explorar la volumetría y disposición espacial de las estructuras, transformando el aprendizaje pasivo en una experiencia visualmente inmersiva. Se discuten también los desafíos encontrados en la implementación de la interactividad táctil usuario-objeto.

**Index Terms**—Realidad Aumentada, Anatomía, Unity, Vuforia, Computación Visual, Image Targets, Educación Médica.

## I. INTRODUCCIÓN

La enseñanza de la anatomía humana ha sido históricamente fundamental en las ciencias de la salud. Desde las disecciones de Vesalio hasta los atlas modernos, el objetivo es comprender la arquitectura tridimensional del cuerpo [1]. Sin embargo, la pedagogía tradicional enfrenta limitaciones: la disección de cadáveres es costosa y destructiva, y los recursos 2D (libros, diapositivas) obligan al estudiante a realizar una “reconstrucción mental” compleja para entender la profundidad y orientación de los órganos [2].

Este fenómeno, conocido como abstracción espacial, a menudo resulta en una comprensión fragmentada. Este proyecto de computación visual propone el uso de Realidad Aumentada (RA) móvil para superar estas barreras. Al utilizar la cámara de un dispositivo móvil para reconocer imágenes de órganos y proyectar modelos 3D, se busca reducir la carga cognitiva mediante el efecto de “Espejo Mágico” [3].

## II. MOTIVACIÓN Y FUNDAMENTACIÓN TEÓRICA

### A. Crisis de Habilidad Espacial

La literatura identifica la habilidad espacial como un predictor crítico de éxito en medicina. Los estudiantes con menor capacidad de rotación mental rinden peor con atlas tradicionales porque agotan sus recursos cognitivos imaginando

Manuscrito preparado para la presentación del curso de Computación Visual. Los autores pertenecen a la Universidad Nacional de Colombia.

vistas en lugar de estudiar funciones [4]. La RA externaliza este procesamiento: el sistema asume la carga de la rotación mental, permitiendo al alumno centrarse en el aprendizaje declarativo.

### B. Incentivación y Engagement

El proyecto busca explícitamente incentivar el estudio. A diferencia de la memorización pasiva, la RA introduce elementos de gamificación visual. Estudios indican que, aunque la retención teórica puede ser similar, la satisfacción y percepción de utilidad son superiores con RA [5].

## III. HERRAMIENTAS Y ARQUITECTURA TÉCNICA

### A. Selección del Motor de Desarrollo: Unity

El núcleo del desarrollo se realizó en **Unity**, debido a su versatilidad para gestionar entornos 3D y su capacidad de compilación multiplataforma para Android.

### B. Selección del SDK de RA: Vuforia vs. AR Foundation

Inicialmente, se intentó implementar la solución utilizando **AR Foundation** y **ARCore**. Sin embargo, durante las pruebas de desarrollo, se encontraron problemas críticos de compatibilidad en dispositivos de gama media, manifestándose en pantallas negras que impedían el acceso al feed de la cámara.

Por consiguiente, se optó por **Vuforia Engine**. Vuforia ofrece una solución robusta para *Image Targets*, utilizando algoritmos de visión por computadora que extraen características naturales (puntos de alto contraste y esquinas) para calcular la pose de la cámara en tiempo real [6].

### C. Activos Digitales

Los modelos anatómicos tridimensionales fueron obtenidos de la plataforma **Sketchfab** en formato **.fbx**. Se seleccionaron cinco modelos principales (corazón, cerebro, pulmones, hígado e intestino), optimizados para su renderizado móvil.

## IV. METODOLOGÍA DE IMPLEMENTACIÓN

### A. Procesamiento de Imágenes y Detección

El sistema se basa en el reconocimiento de marcadores fiduciales, ver figura 1. Para que una imagen funcione como marcador, el motor convierte la entrada de video a escala de

grises y extrae puntos de interés (*keypoints*). Se seleccionaron imágenes con alta distribución de contraste y bordes definidos (“esquinas duras”) para asegurar la estabilidad del rastreo [7].

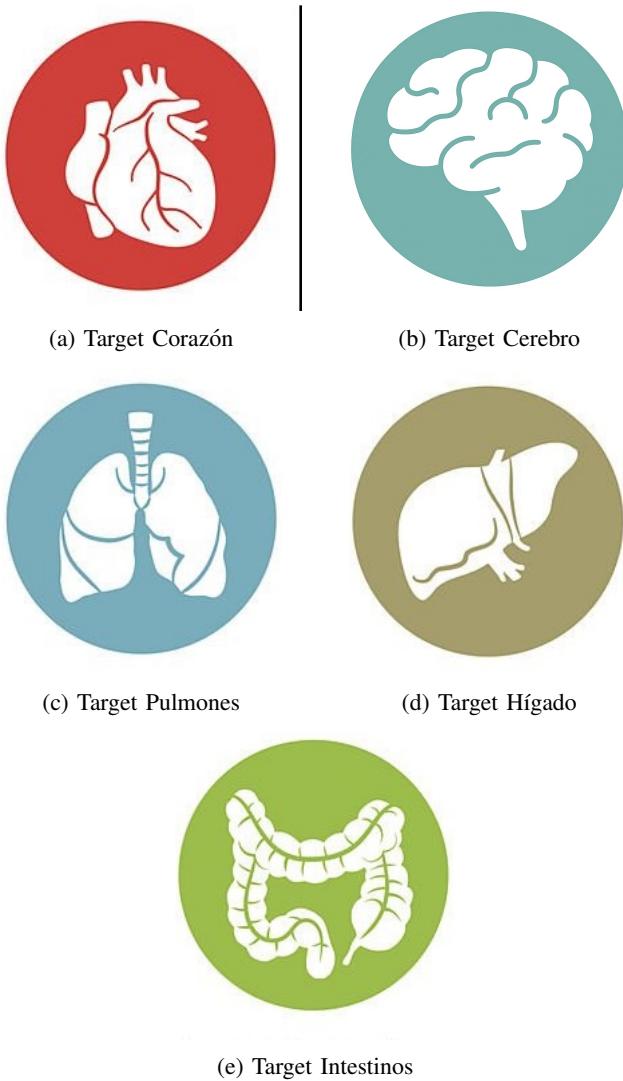


Fig. 1: Patrones visuales utilizados como marcadores. Se seleccionaron ilustraciones anatómicas con alta densidad de características (bordes y contraste) para asegurar un rastreo estable en Vuforia.

### B. Configuración de la Escena

La arquitectura de la escena se organizó jerárquicamente:

- 1) **ARCamera:** Gestiona el ciclo de vida de la RA y renderiza el video de fondo.
- 2) **Image Targets:** Actúan como padres de los objetos virtuales en la jerarquía de Unity.
- 3) **Modelos 3D:** Instanciados como hijos (*children*). Al detectar el marcador, el motor actualiza la matriz de transformación del parent, moviendo el modelo solidariamente.

### C. Compilación para Android

El despliegue presentó retos resueltos mediante la configuración del *Player Settings*:

- **Scripting Backend:** IL2CPP para soportar arquitecturas ARM64.
- **API Gráfica:** Se forzó el uso de **OpenGL ES 3**, ya que Vulkan generaba conflictos de renderizado en dispositivos con chipsets Samsung Exynos [8].
- **Permisos:** Gestión de acceso a cámara en el *AndroidManifest.xml*.

### D. Flujo de Operación del Sistema

El funcionamiento de la aplicación desde la perspectiva del usuario final sigue un flujo lineal simplificado (“Point-and-View”):

- 1) **Inicialización:** Al ejecutar la aplicación en el dispositivo Android, se solicita acceso a la cámara trasera y se inicializa el motor de Vuforia.
- 2) **Exploración:** El usuario apunta con el dispositivo móvil hacia las tarjetas impresas con las imágenes de los órganos (marcadores).
- 3) **Reconocimiento y Aumento:** El sistema detecta los patrones de contraste en la imagen física. Una vez reconocido el marcador, instancia el modelo 3D correspondiente (ej. modelo del corazón sobre la tarjeta del corazón), anclándolo a la posición y rotación de la tarjeta real.
- 4) **Navegación Espacial:** El usuario se desplaza físicamente alrededor del marcador para observar las estructuras anatómicas desde diferentes ángulos, aprovechando los 6 grados de libertad (6DoF) que ofrece el rastreo de la cámara.

## V. RESULTADOS Y DISCUSIÓN

El desarrollo del prototipo arrojó resultados mixtos, con un éxito notable en el componente de visión computacional y desafíos significativos en la capa de interacción de usuario.

### A. Visualización y Rastreo (Tracking)

Como se evidencia en las figuras de la 2 a la 6, la aplicación logra reconocer exitosamente las láminas impresas. El tiempo de latencia entre la detección del marcador y la aparición del modelo es imperceptible para el usuario. El sistema mantiene el anclaje del modelo (corazón, hígado, pulmones, cerebro e intestinos) incluso cuando el usuario rota el dispositivo o mueve la hoja de papel, validando la robustez de los algoritmos de detección de características de Vuforia. Esto permite al estudiante inspeccionar la volumetría del órgano desde múltiples ángulos, cumpliendo el objetivo de reducir la abstracción espacial.

### B. Limitaciones de Interacción (UI)

Originalmente, el diseño contemplaba que al tocar (*tap*) sobre un órgano, se desplegará un panel (*Canvas*) con información anatómica detallada. Sin embargo, esta funcionalidad

**no pudo ser integrada exitosamente** en la versión final presentada.

Se identificaron problemas técnicos relacionados con el sistema de *Raycasting* (lanzamiento de rayos) de Unity en el entorno de RA:

- **Conflictos de Colliders:** La malla compleja de los modelos anatómicos dificultó la precisión de los *Mesh Colliders*, haciendo que los toques en pantalla no fueran registrados correctamente sobre el objeto 3D.
- **Prioridad de Renderizado:** En varios intentos, la interfaz de usuario (UI) quedaba oculta por el video de fondo o no seguía correctamente la posición del modelo en el espacio mundial (*World Space*).

Por lo tanto, la aplicación actual funciona estrictamente como un visualizador anatómico pasivo. El trabajo futuro se centrará en refactorizar el sistema de entrada (*Input System*) para permitir dicha interacción.

## VI. CONCLUSIONES

Se desarrolló un prototipo funcional de realidad aumentada capaz de mejorar la visualización de estructuras anatómicas complejas. El uso de Vuforia demostró ser superior a soluciones nativas en términos de estabilidad de rastreo para este caso de uso específico. Si bien la interacción para obtener metadatos no fue completada debido a conflictos de *raycasting*, el componente visual principal satisface la necesidad de proveer una referencia espacial 3D que los libros de texto no pueden ofrecer.

## REFERENCES

- [1] Bond University Research Repository, “The effectiveness of virtual and augmented reality in health sciences and medical anatomy,” 2025. [Online].
- [2] NIH, “The role of augmented reality in anatomical education: an overview,” *Semantic Scholar*, 2025.
- [3] M. Chytas and P. Johnson, “Augmented reality in medical education: students’ experiences and learning outcomes,” *NIH*, 2021.
- [4] R. E. Mayer, *Multimedia Learning*. Cambridge University Press, 2009.
- [5] PubMed, “Immersive virtual reality and augmented reality in anatomy education: A systematic review,” 2024.
- [6] Vuforia Engine Library, “Image Optimization & Best Practices for Designing Image-Based Targets,” Developer Portal, 2025.
- [7] Sliced Bread Animation, “How to create Image Targets on Vuforia Developer,” 2025.
- [8] P. Scheper, “How to fix the black screen issue on Android when using Vuforia Engine in Unity,” *Medium*, 2025.
- [9] Unity Technologies, “Android permissions in Unity Manual,” 2025.

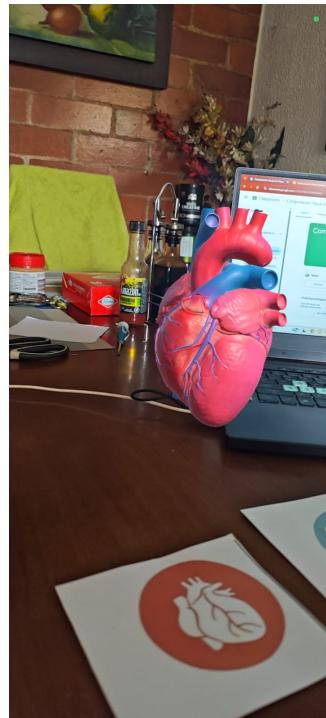


Fig. 2: Vista modelo corazón



Fig. 3: Vista modelo hígado

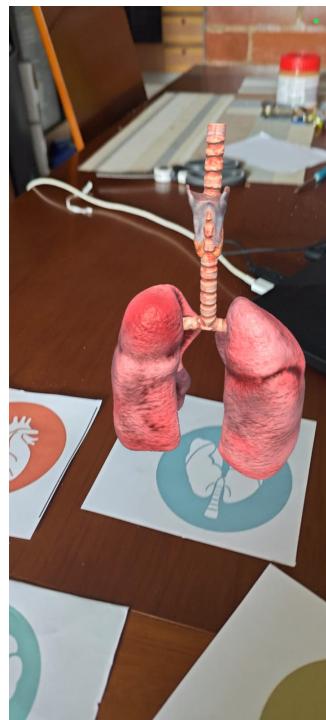


Fig. 4: Vista modelo pulmones

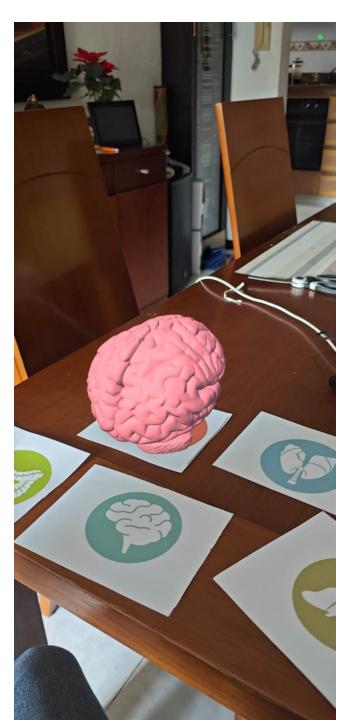


Fig. 5: Vista modelo cerebro



Fig. 6: Vista modelo intestino grueso