



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Dpto. de Comunicaciones

Realidad Aumentada en tiempo real para Conducción
Remota a través de 5G

Trabajo Fin de Máster

Máster Universitario en Tecnologías, Sistemas y Redes de
Comunicaciones

AUTOR/A: Casanova Rivera, Hugo Fernando

Tutor/a: Gómez Barquero, David

Director/a Experimental: Lozano Teruel, Raúl

CURSO ACADÉMICO: 2024/2025

Objetivos – El objetivo general o principal es desarrollar una aplicación de realidad aumentada (AR) que permita la interacción en tiempo real entre robots físicos y objetos virtuales mediante la conectividad de redes 5G, proporcionando al usuario una experiencia inmersiva y precisa.

En cuanto a los objetivos específicos, serían los siguientes:

- Integrar una aplicación de realidad aumentada en un sistema de conducción remota de robots ya existente, permitiendo la interacción en tiempo real entre el robot físico y los objetos virtuales en el entorno.
- Evaluar el rendimiento de la aplicación de AR en redes 5G privadas y públicas, mediante la medida de la latencia en distintos puntos del sistema.
- Analizar la calidad de la experiencia del usuario en términos de precisión, inmersión y facilidad de uso de la aplicación de AR, para evaluar su efectividad en el contexto de la conducción remota.
- Identificar limitaciones y oportunidades de mejora en la sincronización y estabilidad de los elementos de realidad aumentada, optimizando así la interacción entre el entorno físico y los objetos virtuales.

Metodología – Para el desarrollo del proyecto, se ha seguido una metodología estructurada en fases que permiten el análisis y evaluación progresiva del sistema. Se han establecido cinco fases principales: (1) la definición de requerimientos y especificaciones técnicas para la aplicación AR en un entorno de conducción remota a través de redes 5G privadas y públicas; (2) el desarrollo inicial de la AR mediante marcadores (códigos QR) para evaluar su viabilidad en el entorno de conducción remota; (3) la implementación de AR basada en localización para mejorar la precisión y adaptabilidad del sistema; (4) la demostración del sistema en el evento internacional IEEE PIMRC 2024, recogiendo datos de experiencia de usuario; y (5) las mediciones de latencia E2E para evaluar el rendimiento en redes con diferentes condiciones de carga, donde se definieron y analizaron los indicadores clave de desempeño (KPIs) del sistema. Estas mediciones incluyeron la latencia extremo a extremo (E2E), que mide el tiempo desde el impacto con un objeto AR hasta que el robot físico recibe la señal; la latencia de control, que incluye el tiempo necesario para que el robot ejecute el cambio de velocidad tras la recepción de la señal; el jitter, como medida de la estabilidad en la entrega de datos; y la velocidad de transmisión, que refleja el bitrate alcanzado en diferentes escenarios.

Resultados – El principal resultado de este proyecto es el desarrollo de un prototipo funcional que combina tecnologías de Realidad Aumentada (AR), redes 5G y telecontrol en tiempo real para la conducción remota de robots. Este prototipo, diseñado para optimizar la interacción y la toma de decisiones en entornos dinámicos y complejos, fue evaluado a través de métricas técnicas de Calidad de Servicio (QoS) y aspectos subjetivos de Calidad de Experiencia (QoE).

En términos de QoS, se analizaron parámetros clave como la latencia extremo a extremo (E2E), la latencia de control, el jitter y la velocidad de transmisión. Estas métricas se midieron en diferentes escenarios para evaluar la estabilidad y el desempeño de las redes utilizadas. Por otro lado, la QoE se basó en encuestas para evaluar la percepción del usuario sobre aspectos como la inmersión, la precisión en el control y la satisfacción general con el sistema.

Future Directions – Basándose en los resultados obtenidos, se identifican áreas clave para potenciar el sistema y expandir su aplicabilidad:

- AR y dispositivos de visualización: Explorar el uso de gafas VR/AR en lugar de pantallas para mejorar la inmersión y la experiencia del usuario en aplicaciones de conducción remota.
- Optimización de la red 5G: Implementar network slicing en redes 5G para garantizar la calidad del servicio (QoS) y explorar tecnologías como WebRTC para reducir la latencia en transmisión de video en tiempo real.
- Sincronización sensorial: Mejorar la integración entre el feedback visual y háptico, incluyendo sensores adicionales, como acelerómetros avanzados, para incrementar el realismo.
- Integración Sensorial: Mejorar la sincronización entre el feedback visual y háptico, e integrar sensores de fuerzas G para incrementar la sensación de realismo.
- Adaptabilidad a la Red: Desarrollar algoritmos inteligentes que ajusten dinámicamente la calidad de transmisión según las condiciones de la red, mejorando la estabilidad en aplicaciones críticas.
- Ampliación del ámbito de pruebas: Realizar evaluaciones en entornos variados con un mayor número de usuarios y escenarios, incluyendo redes con diferentes niveles de carga, para validar la escalabilidad y robustez del sistema.

Resumen – Las redes de comunicación 5G han revolucionado el control remoto en tiempo real, abriendo nuevas oportunidades en aplicaciones de realidad aumentada (AR) y robótica avanzada. Este trabajo de fin de máster presenta el diseño, desarrollo e implementación de un sistema de realidad aumentada (AR) que integra objetos virtuales interactivos en el contexto de la conducción remota de robots a través de redes 5G. Este sistema permite a los usuarios interactuar de forma inmersiva con objetos AR, posicionados a partir de la señal GNSS recibida por los robots, de forma superpuesta al flujo de video transmitido por una cámara 360.

Las interacciones con estos objetos virtuales influyen directamente en el comportamiento de los robots en forma de bonificaciones o penalizaciones, afectando parámetros como la velocidad, dirección y acciones específicas, lo que añade una capa de gamificación y control dinámico a la experiencia de conducción. Además, la interacción con estos objetos AR es sentida por el usuario en forma de estímulos hápticos que aumentan su inmersión en la conducción.

El trabajo describe la arquitectura completa de la solución, dividida en tres nodos principales: red 5G, robot y usuario. La conducción remota de los robots se realiza a través de una infraestructura 5G, aprovechando la baja latencia y alta velocidad de transmisión de datos para garantizar que las interacciones con los objetos AR se reflejen en tiempo real. El control se realiza desde cockpits inmersivos cada uno compuesto por un volante con pedales, un visor de realidad virtual, y un chaleco háptico.

La evaluación de la implementación se realiza mediante medidas subjetivas (experiencia de usuario en el evento IEEE PIMRC) y objetivas (principalmente latencia a varios niveles de red y aplicación). De esta forma, se analiza el impacto de la AR en la conducción remota, y se analizan posibles optimizaciones y trabajo futuro.

Autor: Hugo Casanova Rivera, [email: hfcasriv@upv.edu.es](mailto:hfcasriv@upv.edu.es)

Director: David Gómez Barquero, [email: dagobar@iteam.upv.es](mailto:dagobar@iteam.upv.es)

Director experimental: Raúl Lozano Teruel, [email: raulote@iteam.upv.es](mailto:raulote@iteam.upv.es)

Palabras clave

Conducción remota, realidad aumentada (AR), redes 5G, robótica móvil, ROS, LiDAR, GNSS-RTK, cámara 360°, telemetría, gamificación, háptica, inmersión visual, interacción en tiempo real.

Lista de Figuras

1	Cronograma del Proyecto	8
2	Evolución 5G del 3GPP de 5G a 5G Avanzado [1]	10
3	Integración de los dominios físico, humano y virtual, basada en la visión Hexa-X 6G de mundos conectados y valores clave [2].	13
4	Arquitectura General del Sistema (Escenario en Red Privada UPV)	21
5	Robots móviles, RB SUMMIT, en el velódromo de la UPV	23
6	Componentes hardware de cada Cockpit para conducción remota	25
7	Componentes software del nodo usuario y flujo de comunicación usuario-robot	26
8	Diagrama de flujo de la lógica de movimiento	30
9	Integración de Objetos AR sobre velódromo digital	31
10	Diagrama de flujo de la gamificación	32
11	Interacción en tiempo real con el sistema de AR en el cockpit de conducción remota durante el IEEE PIMRC 2024.	37
12	Histograma y CDF de la latencia extremo a extremo (E2E) para ambas redes.	38
13	Histograma y CDF de la latencia de Control para ambas redes.	39
14	Histograma y CDF del jitter para ambas redes.	39
15	Izquierda: Comparación de latencia de Red; Derecha: Comparación de latencia de video.	40
16	Izquierda: Comparación de velocidad de transmisión en el velódromo.; Derecha: Histograma de velocidad de transmisión en el velódromo.	41

Contenido

1	Introducción	7
1.1	Motivación	7
1.2	Objetivos	7
1.3	Metodología	8
1.4	Estructura de la memoria	9
2	Estado del Arte	10
2.1	Tecnologías habilitantes	10
2.1.1	Redes Móviles	10
2.1.1.1	Release 15: Fundamentos del 5G	10
2.1.1.2	Release 16: Evolución del 5G	11
2.1.1.3	Release 17: Avances y Nuevos Casos de Uso	11
2.1.1.4	Release 18: 5G Avanzado	12
2.1.1.5	6G: Next Generation	13
2.1.2	Robótica Móvil	14
2.1.2.1	Evolución de la Robótica Móvil	14
2.1.2.2	Sensores y Sistemas de Percepción	14
2.1.2.3	Comunicación y Control	15
2.1.2.4	Conducción Remota Inmersiva	15
2.1.2.5	Gemelos Digitales en Robótica	16
2.1.2.6	Aplicaciones de la Robótica Móvil	16
2.1.3	Realidad Aumentada (AR)	17
2.1.3.1	AR basada en marcadores	18
2.1.3.2	AR basada en proyección	18
2.1.3.3	AR basada en superposición	18
2.1.3.4	AR basada en localización	19
2.1.3.5	Impacto de la AR en la Conducción Remota	19
2.2	Requerimientos técnicos	19
2.2.1	Calidad de Servicio	19
2.2.2	Calidad de experiencia	21
3	Arquitectura de la solución	21
3.1	Infraestructura de Red	22
3.1.1	Componentes Principales de la Red	22
3.1.2	Comunicación y Flujo de Datos	22
3.2	Nodo robot	23
3.2.1	Componentes de Hardware	23
3.2.2	Paquetes de Software en ROS	24
3.3	Nodo de usuario	24
3.3.1	Cockpits inmersivos	24
3.3.2	Componentes software	25

3.3.2.1	Subsistema Lógico	26
3.3.2.2	Subsistemas Periféricos	26
3.3.2.3	Flujo de Comunicación	27
4	Desarrollo e implementación del sistema	27
4.1	Lógica de movimiento	27
4.1.1	Ingreso de Datos	28
4.1.2	Procesamiento	28
4.1.3	Actualización del Escenario	29
4.2	Implementación de Objetos AR	30
4.3	Gamificación de la experiencia de conducción	31
4.3.1	Aparición y desaparición de objetos	32
4.3.2	Lógica Explosión	33
4.3.3	Lógica bonificación/penalización	33
4.3.4	Lógica semáforo	34
4.4	Puesta en marcha del cockpit	34
4.4.1	Configuración de equipos	34
4.4.2	Ejecución de la aplicación	35
5	Integración y demostración	36
5.1	Pruebas iniciales: AR basada en marcadores	36
5.2	Demostración: IEEE PIMRC 2024	36
6	Resultados	37
6.1	Medidas de QoS	38
6.1.1	Latencia Extremo a Extremo (E2E)	38
6.1.2	Latencia de Control	38
6.1.3	Jitter	39
6.1.4	Latencia de Red 5G y de Video	39
6.1.5	Velocidad de Transmisión	40
6.2	Medidas de QoE	41
7	Conclusiones	42
7.1	Limitaciones del trabajo	42
7.2	Futuras líneas de investigación	43

1 Introducción

1.1 Motivación

La conducción remota de robots en tiempo real enfrenta desafíos significativos para su implantación en entornos industriales. En aplicaciones que requieren alta precisión, es crucial que los operadores mantengan una percepción espacial clara y una sincronización precisa con el entorno físico. Las limitaciones de latencia en las redes de comunicación convencionales, y las dificultades para integrar entornos virtuales en tiempo real representan importantes obstáculos. Este proyecto surge de la necesidad de superar estos desafíos mediante la combinación de tecnologías de Realidad Aumentada (AR) y redes 5G, explorando un enfoque que permita un sistema de conducción remota más robusto y confiable.

Este esfuerzo también se alinea con la visión futura hacia redes 6G, en las que se anticipa una mayor convergencia entre el mundo físico y el virtual. Ejemplos concretos de esta convergencia se han implementado en plataformas de prueba como el testbed 6G desarrollado en la UPV, que combina tecnologías habilitadoras como comunicación inmersiva y gemelos ciberfísicos para evaluar aplicaciones de conducción remota en tiempo real a través de una red 5G privada, sirviendo como preludio hacia la visión de las redes 6G [3]. Además, estudios recientes han señalado que las capacidades de baja latencia, alta densidad de conexión y comunicación ultraconfiable serán determinantes para habilitar experiencias inmersivas y sistemas robustos en estas redes [4]. Aprovechar la Realidad Aumentada para integrar visual y sensorialmente ambos entornos no solo ofrece beneficios inmediatos en la conducción remota, sino que abre la puerta a aplicaciones de Gemelos Digitales, donde los robots puedan interactuar con representaciones virtuales en tiempo real. Con ello, se aspira a construir un sistema inmersivo que permita una comunicación fluida y una interacción segura en entornos complejos y dinámicos.

1.2 Objetivos

El objetivo general es desarrollar una aplicación de realidad aumentada (AR) que permita la interacción en tiempo real entre robots físicos y objetos virtuales mediante la conectividad de redes 5G, proporcionando al usuario una experiencia inmersiva y precisa.

En cuanto a los objetivos específicos, son los siguientes:

- Integrar una aplicación de realidad aumentada en un sistema de conducción remota de robots ya existente, permitiendo la interacción en tiempo real entre el robot físico y los objetos virtuales en el entorno.
- Evaluar el rendimiento de la aplicación de AR en redes 5G privadas y públicas, mediante la medida de la latencia en distintos puntos del sistema.
- Analizar la calidad de la experiencia del usuario en términos de precisión, inmersión y facilidad de uso de la aplicación de AR, para evaluar su efectividad en el contexto de la conducción remota.

- Identificar limitaciones y oportunidades de mejora en la sincronización y estabilidad de los elementos de realidad aumentada, optimizando así la interacción entre el entorno físico y los objetos virtuales.

1.3 Metodología

Para el desarrollo del proyecto, se ha seguido una metodología estructurada en fases que permiten el análisis y evaluación progresiva del sistema. Se han establecido cinco fases principales: (1) la definición de requerimientos y especificaciones técnicas para la aplicación AR en un entorno de conducción remota a través de redes 5G privadas y públicas; (2) el desarrollo inicial de la AR mediante marcadores (códigos QR) para evaluar su viabilidad en el entorno de conducción remota; (3) la implementación de AR basada en localización para mejorar la precisión y adaptabilidad del sistema; (4) la demostración del sistema en el evento internacional IEEE PIMRC 2024, recogiendo datos de experiencia de usuario; y (5) las mediciones de latencia E2E para evaluar el rendimiento en redes con diferentes condiciones de carga, donde se definieron y analizaron los indicadores clave de desempeño (KPIs) del sistema. Estas mediciones incluyeron la latencia extremo a extremo (E2E), que mide el tiempo desde el impacto con un objeto AR hasta que el robot físico recibe la señal; la latencia de control, que incluye el tiempo necesario para que el robot ejecute el cambio de velocidad tras la recepción de la señal; el jitter, como medida de la estabilidad en la entrega de datos; y la velocidad de transmisión, que refleja el bitrate alcanzado en diferentes escenarios.

La Figura 1 muestra el cronograma del proyecto indicando la distribución temporal de las fases mencionadas.



Figure 1: Cronograma del Proyecto

1.4 Estructura de la memoria

- Capítulo 1. Presenta la motivación del proyecto, sus objetivos generales y específicos, y la metodología empleada.
- Capítulo 2. Describe las tecnologías habilitantes, enfocándose en redes móviles, robótica móvil y realidad aumentada (AR), además de los requerimientos técnicos para implementar una aplicación AR en un sistema de conducción remota a través de 5G.
- Capítulo 3. Expone la estructura general de la solución, detallando los componentes de cada nodo: nodo de red 5G, nodo del robot, y nodo de usuario, y sus interacciones para lograr una experiencia de conducción remota en tiempo real.
- Capítulo 4. Detalla el proceso de desarrollo (especialmente en el nodo de usuario), desde la lógica de movimiento y visualización en Unity hasta la implementación de objetos AR y la gamificación de la experiencia, describiendo cada paso con la integración de hardware y software en el sistema.
- Capítulo 5. Describe las pruebas iniciales con AR basada en marcadores y las pruebas finales con AR basada en localización, junto con la demostración realizada en el IEEE PIMRC 2024, donde los participantes pudieron experimentar el sistema en un entorno controlado.
- Capítulo 6. Evalúa el rendimiento del sistema mediante medidas objetivas (latencia en diferentes redes) y subjetivas (experiencia del usuario), analizando el impacto de la AR en la conducción remota y las mejoras que pueden optimizar el sistema.
- Capítulo 7. Resume los hallazgos principales del proyecto, sus limitaciones y las posibles líneas de investigación futura.

2 Estado del Arte

2.1 Tecnologías habilitantes

2.1.1 Redes Móviles

La tecnología 5G ha transformado las comunicaciones móviles al proporcionar un marco robusto y flexible que satisface la creciente demanda de conectividad digital. Esta tecnología se caracteriza por su capacidad para soportar aplicaciones diversas, ofreciendo altas velocidades de transferencia, baja latencia y una gran eficiencia en la gestión simultánea de numerosos dispositivos conectados. El 3rd Generation Partnership Project (3GPP) ha liderado el desarrollo de 5G a través de especificaciones implementadas en distintas etapas, cada una con capacidades innovadoras.

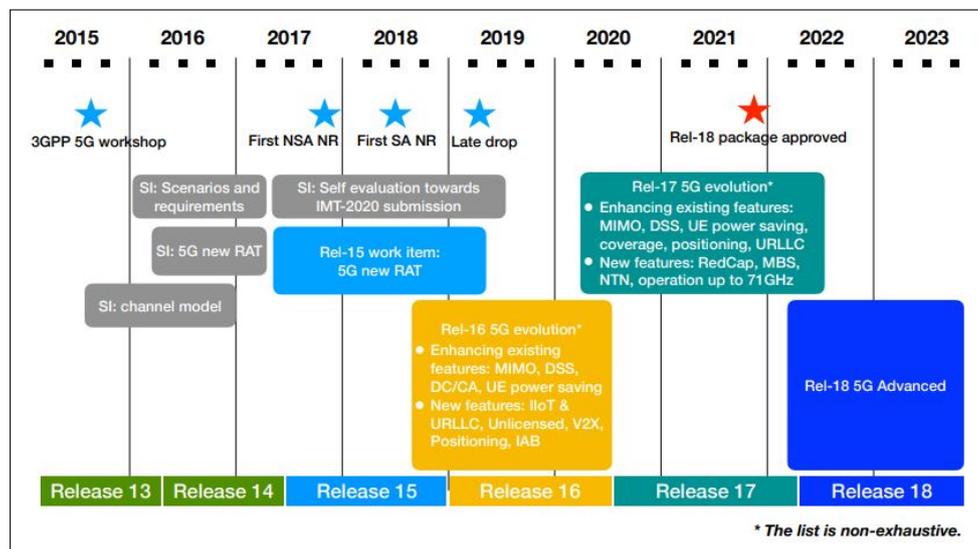


Figure 2: Evolución 5G del 3GPP de 5G a 5G Avanzado [1]

2.1.1.1 Release 15: Fundamentos del 5G

La Release 15 del 3GPP introdujo el 5G System (5GS), compuesto por la red de acceso (RAN) y la red troncal 5G Core, para soportar diversos escenarios de uso:

- Enhanced Mobile Broadband (eMBB): Apoya aplicaciones de alto rendimiento, como video en 4K y realidad virtual, con velocidades de hasta 10 Gbps..
- Ultra-Reliable Low Latency Communications (URLLC): Ofrece latencias de hasta 1 ms para aplicaciones críticas como la telemedicina y la conducción remota.
- Massive Machine Type Communications (mMTC): Facilita la conexión masiva de dispositivos IoT con bajo consumo energético.

La arquitectura 5G soporta modos autónomo (SA) y no autónomo (NSA), permitiendo la transición desde LTE. Entre las innovaciones de NR destacan la flexibilidad del espectro, con operación en bandas sub-6 GHz y ondas milimétricas, el diseño ultraligero para eficiencia espectral, y el uso de Multiple Input Multiple Output (MIMO) para aumentar la diversidad [1].

2.1.1.2 Release 16: Evolución del 5G

La Release 16 del 3GPP representa un avance importante en la evolución de 5G, mejorando las capacidades existentes y habilitando nuevos casos de uso clave para aplicaciones avanzadas [5]:

- **Dynamic Spectrum Sharing (DSS):** Permite la coexistencia de espectros 4G y 5G, optimizando el uso de recursos y aumentando la eficiencia en áreas urbanas densamente pobladas.
- **Beamforming Avanzado:** Mejora la dirección de la señal hacia usuarios específicos, incrementando la capacidad de la red y reduciendo interferencias, especialmente útil para aplicaciones críticas como la conducción remota.
- **Mejoras en URLLC:** Introduce técnicas avanzadas para reducir aún más la latencia, ofreciendo comunicaciones más rápidas y seguras en aplicaciones críticas de baja latencia.
- **Service-Based Architecture (SBA):** Refuerza la arquitectura de 5GC, facilitando una interacción más flexible y eficiente entre componentes de la red.

2.1.1.3 Release 17: Avances y Nuevos Casos de Uso

La Release 17, lanzada en 2022 por 3GPP, expande el ecosistema 5G hacia aplicaciones emergentes y el Internet Industrial de las Cosas (IIoT), optimizando el rendimiento y la conectividad para una amplia variedad de dispositivos. Las principales mejoras incluyen [6]:

- **Soporte para IIoT:** Mejora la capacidad de la red para manejar dispositivos industriales en entornos de producción automatizada y fábricas inteligentes, facilitando soluciones de alta eficiencia.
- **Dispositivos RedCap:** Introduce dispositivos de capacidad reducida (RedCap), ideales para sensores y dispositivos de baja complejidad, reduciendo costos de implementación para aplicaciones que no requieren altas tasas de datos.
- **Integración de Múltiples Antenas:** Se han realizado mejoras en la tecnología Massive MIMO y la conformación de haces (beamforming), lo que mejora la capacidad de la red para manejar múltiples conexiones simultáneas y aumentar la eficiencia del espectro.

- Soporte para V2X: Facilita la comunicación entre vehículos (V2X), permitiendo que los vehículos autónomos interactúen eficientemente con su entorno, un avance clave para la seguridad y la eficiencia en movilidad inteligente.
- Edge Computing: Integra procesamiento en el borde para disminuir la latencia y mejorar el rendimiento en aplicaciones críticas, permitiendo respuestas en tiempo real al procesar datos cerca del usuario.

2.1.1.4 Release 18: 5G Avanzado

La Release 18, o 5G Avanzado, representa una evolución importante en la arquitectura de 5G, integrando inteligencia artificial (IA) y aprendizaje automático (AM) para mejorar la eficiencia y adaptabilidad de la red. Esta release aporta optimizaciones clave de versiones previas y nuevas capacidades, esenciales para aplicaciones avanzadas como las comunicaciones inmersivas en AR y XR [1]. Sus innovaciones incluyen:

- Integración de Redes Satelitales y Cobertura IoT: Mejora la cobertura en áreas remotas mediante la integración de redes no terrestres (NTN) y el soporte eficiente para IoT y Comunicaciones de Tipo Máquina (MTC) a través de enlaces satelitales, permitiendo una conectividad estable en ubicaciones de difícil acceso.
- IA y AM para Gestión de la Red: Introduce IA y AM para optimizar la asignación de recursos y funciones como la retroalimentación del canal y el beamforming, proporcionando adaptabilidad y mejor eficiencia en tiempo real, crucial para las aplicaciones de AR.
- Soporte de XR, AR, y VR: Facilita comunicaciones inmersivas con mejoras en la latencia y el ancho de banda, permitiendo experiencias de realidad extendida (XR) y realidad aumentada (AR) con menor latencia y mayor interactividad.
- Sidelink, Posicionamiento y Comunicación de Proximidad: Mejora las capacidades de comunicación directa entre dispositivos y de posicionamiento preciso, aspectos importantes para la interacción en tiempo real entre usuario y robot en aplicaciones de teleconducción.
- Soporte para Nuevos Dispositivos y Eficiencia Energética: Amplía el soporte a dispositivos de menor complejidad, como drones y sensores industriales, mejorando la eficiencia energética para aplicaciones de largo alcance y bajo consumo.
- Servicios de Multicast/Broadcast y Network Slicing: Permite el envío simultáneo de contenido a múltiples usuarios y facilita la creación de redes segmentadas para aplicaciones específicas, lo que asegura calidad de servicio (QoS) en aplicaciones críticas.

Estas mejoras convierten a Release 18 en una plataforma flexible y potente para aplicaciones de alto rendimiento, como las experiencias inmersivas de teleconducción y realidad aumentada que se implementan en este proyecto.

2.1.1.5 6G: Next Generation

A pesar de que 6G no se espera hasta 2030, el desarrollo de tecnologías Beyond 5G (B5G) ya está en marcha como preámbulo de lo que será 6G. Las redes 6G buscarán la convergencia total entre los mundos físico, humano y digital, creando un continuo ciberfísico o Metaverso donde las barreras entre la realidad física y virtual se diluyen [7].

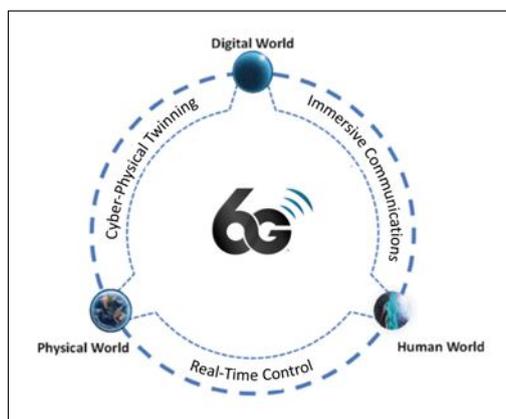


Figure 3: Integración de los dominios físico, humano y virtual, basada en la visión Hexa-X 6G de mundos conectados y valores clave [2].

- Velocidades y latencia: Las redes 6G prometen velocidades exponencialmente mayores y latencias casi imperceptibles, habilitando nuevas aplicaciones inmersivas, como comunicaciones holográficas, interfaces de realidad mixta y feedback háptico en tiempo real.
- Digital Twins: Los gemelos ciberfísicos o digitales serán un elemento central, permitiendo la simulación y optimización en tiempo real de sistemas físicos a través de sus contrapartes digitales, algo que será clave para la conducción remota avanzada y la gestión de infraestructuras robóticas.
- Tactile Internet: Esta evolución también permitirá el Tactile Internet, donde la información multisensorial (como el tacto) se transmitirá en tiempo real, mejorando la interacción hombre-máquina en entornos de Realidad Aumentada (AR) y Realidad Virtual (VR) [8].

Además, la computación en el borde será esencial en 6G, permitiendo el procesamiento en tiempo real de grandes cantidades de datos cerca del usuario final, reduciendo la latencia y mejorando la eficiencia del sistema. La convergencia entre las comunicaciones inmersivas y los gemelos digitales hará posible la interacción fluida en aplicaciones críticas como la conducción remota inmersiva [4].

2.1.2 Robótica Móvil

La robótica móvil ha progresado de manera significativa en las últimas décadas, impulsada por avances en procesadores, sensores y capacidades de procesamiento en tiempo real [9]. Estos desarrollos han ampliado su aplicación a sectores industriales y exploratorios complejos [10].

Originalmente, estos sistemas se enfocaban en tareas autónomas básicas; sin embargo, la incorporación de tecnologías de comunicación avanzadas, como las redes 5G han potenciado sus capacidades. Actualmente, los robots móviles pueden operar en entornos desafiantes mediante telepresencia y control remoto, permitiendo una interacción en tiempo real con el entorno físico gracias a la baja latencia y alta velocidad de transmisión del 5G [11] [12].

La AR proporciona una visualización inmersiva que mejora la percepción espacial del operador, facilitando decisiones precisas en tiempo real. En aplicaciones de alta precisión, donde la integración de objetos virtuales en el entorno físico permite una interacción más intuitiva y efectiva, optimizando así el control en entornos complejos y mejorando la eficiencia y seguridad en distintas operaciones [1].

2.1.2.1 Evolución de la Robótica Móvil

Desde los años 70, con robots pioneros como Shakey y Lunokhod 1, la robótica móvil ha avanzado con un enfoque en mejorar la autonomía y precisión de movimiento. Shakey, desarrollado en el Instituto de Investigación de Stanford, fue uno de los primeros robots en utilizar sensores básicos para ejecutar tareas autónomas simples, mientras que el Lunokhod 1, operado de forma remota desde la Tierra, exploró la Luna, demostrando el potencial del control remoto en entornos extremos [10].

Otro hito relevante fue el rover Sojourner de la NASA en 1997, que demostró cómo los robots podían adaptarse a terrenos desafiantes y operar de manera semi-autónoma en Marte, integrando sistemas avanzados de sensores y comunicación [13].

Un ejemplo representativo de la robótica móvil moderna es Spot de Boston Dynamics, lanzado en 2015 y mejorado continuamente. Este robot está diseñado para moverse de manera autónoma en entornos complejos, como fábricas y espacios de rescate. Equipado con tecnologías de percepción avanzada, como LiDAR y cámaras RGB-D, Spot también puede conectarse a redes 5G, lo que le permite operar con baja latencia en tiempo real y facilita su uso en aplicaciones de teleoperación [14].

2.1.2.2 Sensores y Sistemas de Percepción

En la conducción remota, los sensores desempeñan un papel crucial para proporcionar al robot una percepción precisa de su entorno. La integración de cámaras RGB-D, que capturan tanto imágenes como datos de profundidad, y de sensores LiDAR, que permiten la creación de mapas tridimensionales en tiempo real, han transformado la capacidad de los robots móviles para detectar y evitar obstáculos. Esta información detallada del entorno permite al robot tomar decisiones de navegación de forma autónoma o semiautónoma, mejorando significativamente la seguridad y eficiencia de la conducción remota [9].

Además, la incorporación del sistema de navegación global por satélite (GNSS) ha sido fundamental en aplicaciones de robótica móvil que requieren una localización precisa y en exteriores. El GNSS permite a los robots orientarse en entornos abiertos y calcular su posición exacta, lo cual es esencial para tareas de exploración o logística en espacios amplios. Este sistema se combina frecuentemente con otros sensores, como las unidades de medición inercial (IMU) y sensores ultrasonidos, para mejorar la precisión de la localización y permitir un seguimiento robusto en condiciones donde la señal GNSS puede ser inestable, como en áreas urbanas densas o interiores parcialmente expuestos [15].

En conjunto, estos sensores crean un sistema de percepción avanzado que permite la fusión de datos en tiempo real, mejorando la capacidad de los robots móviles para navegar de forma segura en entornos complejos. La combinación de GNSS, LiDAR, cámaras RGB-D, IMUs y otros sensores optimiza la detección y localización de obstáculos, facilitando una respuesta inmediata ante posibles riesgos durante la conducción remota y en aplicaciones de teleoperación en tiempo real.

2.1.2.3 Comunicación y Control

Uno de los retos más grandes en la robótica móvil para la conducción remota es asegurar que la comunicación entre el robot y el operador sea rápida y confiable. Los sistemas modernos de robótica móvil utilizan protocolos de comunicación como ROS (Robot Operating System), que permite la integración de diferentes módulos de hardware y software, facilitando la interacción entre los sensores del robot y el sistema de control. ROS proporciona una estructura modular que facilita el desarrollo de sistemas robóticos complejos y permite la rápida adaptación de los robots a entornos dinámicos.

Además, la computación en el borde (edge computing) se ha convertido en una tecnología clave para reducir la latencia en la conducción remota. En lugar de enviar todos los datos a un servidor en la nube para su procesamiento, los sistemas de computación en el borde permiten que los datos se procesen lo más cerca posible del robot, lo que reduce los tiempos de respuesta y mejora la eficiencia del sistema [9, 13].

El uso de redes 5G ha potenciado el rendimiento de la computación en el borde en aplicaciones de robótica móvil, ya que la baja latencia y alta capacidad de transmisión de 5G permiten una comunicación en tiempo real entre el robot y el sistema de control. La integración de edge computing y 5G es ideal para sistemas basados en ROS, que requieren transferencias de datos en tiempo real entre múltiples dispositivos y sensores. Esta integración permite la ejecución de aplicaciones complejas que requieren una sincronización precisa y confiable, como la conducción remota y la telepresencia en entornos de colaboración humano-robot, donde se necesitan respuestas rápidas y precisas para garantizar la seguridad y efectividad de las operaciones [16].

2.1.2.4 Conducción Remota Inmersiva

La conducción remota inmersiva es una evolución de los sistemas de teleoperación que incorpora tecnologías avanzadas de realidad virtual (VR) y aumentada (AR) para mejorar la interacción y el control de robots o vehículos a distancia. Este enfoque busca propor-

cionar una experiencia más intuitiva y natural para el operador, aumentando la precisión y seguridad en la operación al permitir que el usuario tenga una percepción en tiempo real del entorno remoto [17].

Gracias a la baja latencia y alta capacidad de transmisión de datos de las redes 5G, los operadores pueden visualizar y controlar robots de manera precisa mediante interfaces inmersivas y cockpits virtuales. Estos sistemas permiten que el operador tome decisiones informadas, mejorando su capacidad de navegación y operación en entornos complejos o peligrosos.

2.1.2.5 Gemelos Digitales en Robótica

Los gemelos digitales son representaciones virtuales dinámicas de sistemas físicos que permiten simular, monitorear y optimizar su funcionamiento en tiempo real. En la conducción remota de robots, los gemelos digitales permiten al operador visualizar el comportamiento del robot y su entorno de manera precisa en un espacio virtual, lo que facilita la identificación temprana de problemas y la optimización operativa.

La conectividad 5G permite a los gemelos digitales actualizarse en tiempo real, reflejando cualquier cambio en el entorno del robot sin retrasos perceptibles. Esto es crucial en entornos donde se necesita una respuesta rápida para evitar incidentes [9]. Además, los gemelos digitales permiten realizar simulaciones avanzadas y predicciones de fallos, mejorando la eficiencia operativa y reduciendo el tiempo de inactividad [18].

2.1.2.6 Aplicaciones de la Robótica Móvil

En la Industria 4.0, los robots móviles desempeñan un papel crucial en fábricas y centros logísticos, donde operan de forma autónoma o semiautónoma para transportar materiales y productos entre estaciones de trabajo. Estos robots, integrados en sistemas de control centralizados, permiten a los operadores monitorear su desempeño en tiempo real y realizar ajustes para optimizar las rutas y tiempos de entrega.

Un ejemplo representativo de este tipo de aplicaciones es el robot RB-SUMMIT, desarrollado por la empresa Robotnik, que cuenta con una estructura robusta y versátil, diseñada para entornos tanto interiores como exteriores. Este robot, equipado con ruedas y sensores avanzados como LiDAR, GNSS y cámara 360°, es adecuado para tareas de inspección, mantenimiento y misiones de rescate, además de tener capacidad para adaptarse a pendientes de hasta el 80%.

En el ámbito de la movilidad y exploración autónoma, los robots con ruedas han demostrado ser efectivos en tareas de exploración y navegación en terrenos complejos, como demuestra el robot Karo. Este tipo de robot ha sido optimizado para moverse de manera eficiente en terrenos irregulares, integrando un sistema de control que considera las propiedades dinámicas del robot para adaptarse a las condiciones del terreno [13].

Por último, en aplicaciones al aire libre, como la vigilancia de infraestructuras y la logística en áreas amplias, la combinación de robots móviles con capacidades de navegación avanzada y conectividad 5G permite una comunicación fluida y en tiempo real entre el robot y los sistemas de control remoto. Este tipo de integración permite mejorar

la seguridad y eficiencia de las operaciones, facilitando un control preciso en entornos dinámicos y complejos.

2.1.3 Realidad Aumentada (AR)

La Realidad Aumentada (AR) es una tecnología que ha cambiado radicalmente la forma en que interactuamos con nuestro entorno físico, añadiendo capas de información digital que enriquecen nuestra percepción y comprensión del mundo.

Esta tecnología ha cobrado popularidad en aplicaciones de entretenimiento, como los juegos Pokémon Go y Mario Kart Live: Home Circuit. Pokémon Go utiliza principalmente AR basada en localización, colocando criaturas virtuales en lugares específicos del mundo real mediante la posición GNSS del usuario. Además, cuando el jugador intenta capturar un Pokémon, el juego activa una función de superposición, mostrando al personaje virtual en el entorno visible a través de la cámara del dispositivo [19].

Por su parte, Mario Kart Live: Home Circuit combina AR basada en marcadores y en superposición. El juego utiliza marcadores físicos que el usuario coloca para definir la pista de carreras, permitiendo que el vehículo físico reconozca y siga el recorrido. Al mismo tiempo, elementos y personajes virtuales de Mario Kart se superponen sobre el entorno real captado por la cámara del vehículo, creando una experiencia de carrera inmersiva dentro de un espacio físico como el hogar [19].

La AR puede implementarse mediante diferentes tipos de técnicas, las cuales se explorarán en las siguientes secciones:

- AR basada en marcadores, que emplea patrones visuales como referencia para superponer información digital;
- AR basada en proyección, que proyecta información directamente sobre las superficies físicas;
- AR basada en superposición, que sustituye parcial o totalmente la vista de un objeto físico por una representación virtual;
- AR basada en localización, que posiciona objetos virtuales utilizando la ubicación geográfica del usuario.

Es igualmente importante considerar los medios a través de los cuales los usuarios interactúan con esta tecnología. Generalmente, la visualización de entornos AR se realiza mediante dispositivos como gafas de realidad mixta o pantallas convencionales. Las gafas de AR/VR permiten una experiencia inmersiva mediante el uso de sistemas ópticos avanzados que proyectan elementos virtuales directamente sobre el campo visual del usuario. Dispositivos como las Meta Quest 3 y las Microsoft HoloLens son ejemplos destacados en esta categoría. Por otro lado, el uso de pantallas, como monitores o dispositivos móviles, ofrece una alternativa más accesible para experimentar la AR sin necesidad de hardware especializado.

2.1.3.1 AR basada en marcadores

La AR basada en marcadores es una de las técnicas más utilizadas y consiste en el uso de patrones visuales (como códigos QR o símbolos específicos) que actúan como puntos de referencia para superponer elementos digitales en el entorno físico. Los sistemas de visión por computadora analizan las imágenes capturadas por las cámaras para identificar estos marcadores y generar modelos 3D o información relevante sobre el espacio físico, creando así una representación virtual precisa y dinámica en tiempo real [20].

Esta tecnología se aplica en entornos industriales, donde facilita tareas de ensamblaje y mantenimiento al proyectar instrucciones directamente sobre los componentes o equipos que los operadores manipulan. Por ejemplo, el Instituto de Tecnologías Industriales y Automatización (ITIA) en Italia emplea AR basada en marcadores para guiar a los operadores en el prototipado virtual, permitiéndoles ajustar y evaluar modelos antes de la producción, lo que reduce costos y tiempos de desarrollo [21]. En sistemas de conducción remota, esta tecnología permite que el robot detecte marcadores visuales, como códigos QR, que desencadenan la aparición de información adicional o de elementos virtuales. Esto proporciona al operador referencias visuales en tiempo real que facilitan la navegación y el control con mayor precisión.

2.1.3.2 AR basada en proyección

La AR basada en proyección elimina la necesidad de pantallas físicas y dispositivos portátiles, proyectando imágenes digitales directamente sobre las superficies físicas del entorno [22]. Este tipo de tecnología se emplea especialmente en entornos industriales, donde es fundamental que los operadores puedan acceder a instrucciones o diagramas directamente sobre los componentes o equipos que están manipulando. Un ejemplo destacado es el sistema de asistencia en ensamblaje que guía a los trabajadores mostrando información directamente sobre las piezas en el área de trabajo. Este sistema proyecta instrucciones detalladas para el montaje, ajustándose al punto de vista del operador y ofreciendo retroalimentación inmediata sobre errores o pasos completados [23].

2.1.3.3 AR basada en superposición

La AR basada en superposición se utiliza cuando se necesita reemplazar parcialmente la vista de un objeto físico por una representación virtual. Esta tecnología funciona mediante la detección y reconocimiento del objeto en el entorno físico, sobre el cual se superpone la imagen digital. Utilizando algoritmos de visión artificial, el sistema identifica el objeto y ajusta la imagen virtual para alinearse con su forma y posición, creando una experiencia visual cohesiva que mezcla el mundo real y el digital sobre modelos físicos [24]. En el diseño de productos, la AR basada en superposición permite visualizar prototipos virtuales sobre modelos físicos, facilitando la evaluación de características sin necesidad de fabricar modelos reales. En conducción remota, esta técnica permite a los operadores ver versiones aumentadas de los robots, destacando componentes clave y rutas de navegación, lo que mejora la eficiencia y seguridad. También es útil en diagnóstico preventivo, al representar

datos sensoriales directamente sobre el modelo del robot para detectar posibles problemas [25].

2.1.3.4 AR basada en localización

La AR basada en localización utiliza la posición geográfica del usuario, extraída a partir de sensores como el GNSS, giroscopios y acelerómetros, para posicionar los objetos virtuales. Este tipo de AR es ampliamente utilizado en aplicaciones de navegación y turismo, donde el contenido digital se adapta dinámicamente a la posición del usuario en el mundo real. Un ejemplo popular es Pokémon Go, en el cual los elementos virtuales, como personajes y objetos, se integran geográficamente en ubicaciones específicas del entorno físico del jugador, logrando una experiencia de juego inmersiva basada en la localización.

En el contexto de la conducción remota inmersiva, esta modalidad de AR permite que los operadores visualicen en tiempo real la posición exacta del robot en entornos de trabajo difíciles o peligrosos, como plantas industriales o zonas de desastre, facilitando el control preciso a través de la interacción con mapas 3D del entorno [26].

2.1.3.5 Impacto de la AR en la Conducción Remota

En el contexto de la conducción remota y el control de robots a través de redes 5G, la AR proporciona una interfaz inmersiva que enriquece la experiencia del usuario a través de la gamificación y la representación gráfica de información captada por los sensores del robot en tiempo real. Además, la AR añade una capa de seguridad a la conducción remota, puesto que notifica al usuario de eventos que suceden en el entorno del robot. Estas interfaces no solo proporcionan datos visuales, sino que también permiten que el operador manipule el entorno en tiempo real mediante gestos y comandos que son procesados instantáneamente gracias a la alta capacidad de transmisión de las redes 5G. [27].

La combinación de AR con los gemelos digitales es otro avance significativo que ha potenciado la eficiencia en la conducción remota. Los gemelos digitales permiten crear una réplica virtual del robot y su entorno, que puede interactuar con los objetos AR y generar retroalimentación visual o háptica. Esto no solo mejora la capacidad de respuesta del operador ante problemas técnicos, sino que también permite realizar simulaciones en tiempo real antes de ejecutar acciones críticas, reduciendo riesgos y optimizando el tiempo de operación [28].

2.2 Requerimientos técnicos

2.2.1 Calidad de Servicio

La Calidad de Servicio (QoS) en redes 5G juega un papel fundamental para asegurar el desempeño óptimo en aplicaciones de conducción remota y sistemas inmersivos de realidad aumentada (AR). Las redes 5G tienen la capacidad de soportar altas tasas de transferencia de datos, lo que permite un streaming de video de alta resolución y la transmisión de datos complejos en tiempo real [29].

Uno de los requerimientos más críticos para la transmisión de video 360° de dos robots es el ancho de banda en el enlace ascendente. Cada robot necesita un ancho de banda significativo de hasta 15 Mbps para transmitir video a máxima calidad, junto con 1 Mbps adicional reservado a la telemetría. Para soportar el uso de 2 robots, es necesario disponer del doble de ancho de banda. En la Tabla 1, se relacionan las resoluciones con el ancho de banda requerido por cada robot.

Resolución	Bitrate Generado (Mbps)	Ancho de Banda Necesario (Mbps)
4K (1920p)	15	32
HD (1280p)	8	18
SD (960p)	5	12
LD (640p)	3	8

Tabla 1: Ancho de banda requerido en función del formato de video.

Para asegurar una experiencia de usuario fluida, la latencia de red debe ser lo más baja posible para minimizar el impacto de las latencias de aplicación, que incluyen procesos como codificación de video, procesamiento en Unity y ejecución en los motores del robot. Estas latencias se dividen en:

- **Latencia de Video:** Depende de los procesos de codificación y decodificación, así como del tipo de compresión y la calidad de transmisión. En aplicaciones de realidad aumentada en tiempo real, la latencia recomendada para garantizar una experiencia fluida suele estar entre 84 ms y 100 ms, con 84 ms como valor ideal en condiciones óptimas de red y hardware [30]
- **Latencia de Control:** Mide el tiempo desde que el usuario interactúa con un objeto AR hasta que el robot físico ejecuta la acción. Depende de los motores y el sistema de respuesta del robot. En este proyecto, se ha identificado que esta latencia representa el cuello de botella principal debido a los retardos de los motores en ejecutar los comandos. Para garantizar una respuesta ágil en sistemas de teleoperación, estudios recomiendan una latencia de telecontrol de entre 700 a 800 ms, aunque puede extenderse hasta los 1500 ms en entornos menos críticos [31].

La latencia extremo a extremo (E2E) combina estas dos métricas, representando el tiempo total desde que el usuario interactúa con un objeto AR hasta que el robot ejecuta la acción y el cambio es visualizado por el usuario a través del video. En aplicaciones de teleoperación crítica, esta latencia debería mantenerse por debajo de 200 ms para garantizar una interacción precisa y rápida entre el operador y el robot [32]. Estudios recientes en redes URLLC aplicadas a sistemas robóticos respaldan este valor, dado que permite una interacción suficientemente rápida entre el operador y el robot, asegurando así una conducción precisa y segura [30, 32].

Por otro lado, de forma paralela a la latencia E2E, es necesario tener en cuenta la retroalimentación háptica que se activa cuando el usuario interactúa con un objeto AR. Los sistemas que incorporan dispositivos hápticos deben mantener una latencia de

entre 10 y 20 ms para proporcionar una experiencia táctil precisa y en tiempo real [33]. Esto es fundamental en aplicaciones de teleoperación, donde el operador debe sentir las interacciones con el entorno remoto para tomar decisiones rápidas y precisas [34].

2.2.2 Calidad de experiencia

La Calidad de Experiencia (QoE) es igualmente importante para aplicaciones inmersivas en redes 5G, ya que mide la percepción del usuario sobre el rendimiento y la efectividad del sistema. Se mide mediante métricas subjetivas de tipo Mean Opinion Score (MOS) sobre un gran número de usuarios, de forma que no solo se ve afectada por la QoS ofrecida por el sistema sino del contexto personal de cada usuario.

3 Arquitectura de la solución

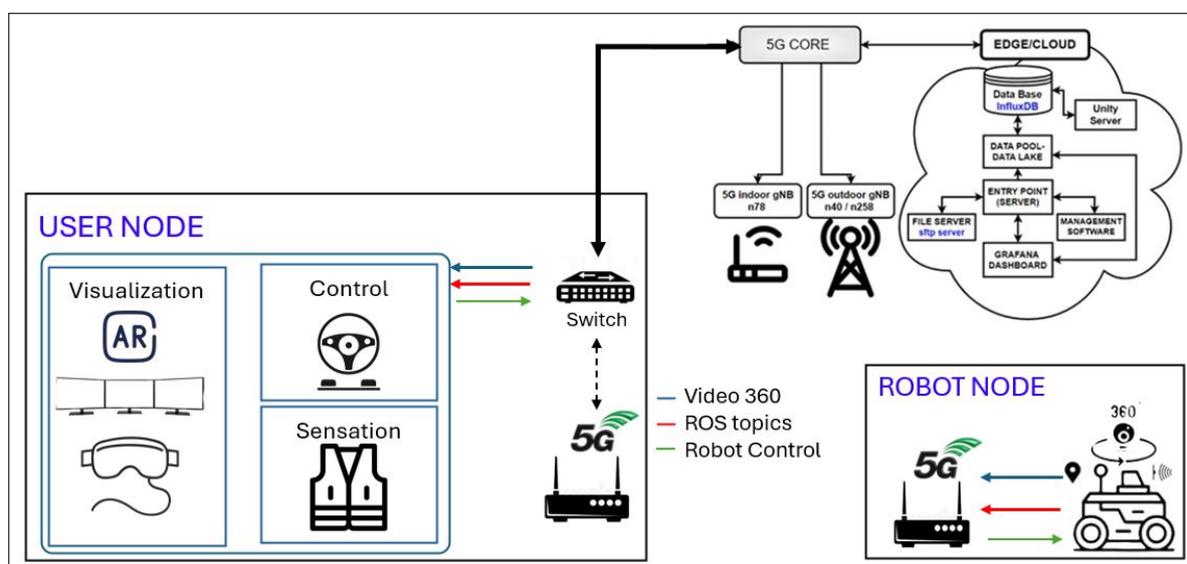


Figure 4: Arquitectura General del Sistema (Escenario en Red Privada UPV)

La arquitectura del sistema de conducción remota está compuesta por tres elementos principales: la infraestructura de red, el nodo del robot y el nodo del usuario. En esta sección, se describirá cómo estos componentes interactúan para garantizar una experiencia fluida y en tiempo real. Cada componente desempeña un rol fundamental dentro del sistema: la infraestructura de red garantiza la conectividad y la comunicación bidireccional entre el usuario y el robot; el nodo robot integra los sensores necesarios para la navegación, detección de obstáculos y posicionamiento preciso; mientras que el nodo usuario maneja el control remoto del robot, la visualización inmersiva en tiempo real y la interacción con el sistema mediante dispositivos hápticos.

La relación entre estos elementos, así como sus componentes principales, se puede observar en la Figura 4. Esta figura ilustra la arquitectura general del sistema en el

contexto de una red privada 5G, destacando las conexiones físicas, los flujos de datos y la infraestructura utilizada. Para la red pública 5G, aunque el núcleo de red no es personalizable, se mantiene una funcionalidad equivalente para cumplir con los objetivos del sistema.

3.1 Infraestructura de Red

La infraestructura de red es un componente central en la arquitectura del sistema, habilitando la comunicación entre el nodo usuario y el nodo robot. Este sistema ha sido diseñado para operar en dos escenarios distintos: una red privada 5G (UPV) y una red pública 5G (Orange). Aunque ambos escenarios utilizan tecnología 5G, su estructura y funcionalidades presentan diferencias importantes que impactan en el comportamiento y rendimiento del sistema.

En la red privada 5G, el nodo usuario se conecta directamente al núcleo de red (*5G Core*) y al servidor de aplicaciones alojados en el *edge*. Esta configuración permite un control más detallado de la red, incluyendo funcionalidades avanzadas como slicing para priorizar el tráfico crítico. Por otro lado, la red pública 5G se basa en la infraestructura proporcionada por el operador, donde el nodo usuario y el nodo robot se comunican a través del núcleo público. Aunque esta opción no permite el mismo nivel de control sobre la red, ofrece un entorno accesible para evaluar el desempeño del sistema en un entorno comercial.

Ambas configuraciones son fundamentales para explorar diferentes escenarios de implementación y la adaptabilidad del sistema.

3.1.1 Componentes Principales de la Red

- Red de Acceso (5G RAN): La Radio Access Network (RAN) es responsable de la conectividad de radio, ofreciendo cobertura tanto en interiores como exteriores y permitiendo la transmisión y recepción de datos mediante el uso de la banda n78, con un ancho de banda de 100 MHz; o la banda n40, con un ancho de banda de 20 MHz. Este enlace se establece entre el nodo de usuario y el nodo robot, garantizando un throughput adecuado para transmisión de video y telemetría.
- Núcleo de Red (5G Core): Basado en la solución de código abierto Open5GS, el núcleo de la red soporta la función de slicing, asegurando que cada segmento de red pueda priorizar el tráfico específico y asignar recursos adecuados, lo que es crítico en una aplicación de teleconducción remota. Esto permite gestionar el tráfico entre ambos nodos, optimizando la latencia y el ancho de banda para aplicaciones en tiempo real.

3.1.2 Comunicación y Flujo de Datos

La arquitectura de red soporta una transmisión bidireccional entre el nodo usuario y el nodo robot. El flujo de datos se estructura mediante sockets TCP y UDP:

- TCP para Telemetría y Control: Este protocolo asegura que los datos de telemetría del robot, incluyendo su posición, orientación, velocidad y estado de batería, se reciban de forma confiable en el nodo usuario.
- UDP para Video en Tiempo Real: La transmisión de video de la cámara 360° del robot al nodo usuario utiliza UDP con el protocolo RTSP, garantizando una mínima latencia en la entrega de video.

3.2 Nodo robot

El nodo robot está compuesto por el propio robot SUMMIT-XL de Robotnik, diseñado para operar en interiores y exteriores con una carga útil de hasta 65 kg y una velocidad máxima de 3 m/s. Este robot cuenta con sensores avanzados y sistemas de navegación que los hacen ideales para aplicaciones de conducción remota en entornos complejos.



Figure 5: Robots móviles, RB SUMMIT, en el velódromo de la UPV

3.2.1 Componentes de Hardware

Para operar de manera autónoma y eficiente en teleconducción, cada robot está equipado con los siguientes componentes de hardware:

- 4 Ruedas motorizadas: Diseñadas para tracción en diversos terrenos.
- Batería LiFePO4 de 15Ah @ 48VDC: Proporciona energía para hasta 5 horas de operación continua.
- IMU VectorNav VN-100: Una unidad de medición inercial avanzada que combina acelerómetros, giroscopios, magnetómetros y un sensor barométrico, ofreciendo datos de orientación en 3D con una precisión RMS de 0.5°.
- LiDAR Robosense RS-LiDAR-16: Sensor de escaneo láser de 360° que facilita la detección de obstáculos en un rango de hasta 150 metros, asegurando una navegación segura.

- Cámara 360° Labpano Pilot One: Cámara panorámica que permite capturar video de alta resolución (ver resoluciones disponibles en Tabla 1), útil para la transmisión de video en tiempo real en aplicaciones de realidad aumentada.
- Receptor GNSS u-blox C099: Un receptor de posicionamiento multi-constelación que, mediante correcciones RTK, proporciona una precisión submétrica para navegación y mapeo precisos.
- Módem 5G Teltonika RUTX50: Conecta el nodo robot a la red móvil 5G, permitiendo la transmisión de datos como velocidad, nivel de batería, posición, orientación, tiempo de última vuelta, distancia a obstáculos cercanos y video en tiempo real. También recibe comandos de control desde el nodo usuario para la conducción remota del robot, tales como avanzar, retroceder, girar y detenerse.

3.2.2 Paquetes de Software en ROS

Cada robot utiliza el sistema operativo ROS para la gestión y comunicación entre sus componentes, mediante los siguientes paquetes de software clave:

- `robotnik_base_hw`: Controla los motores y proporciona datos de odometría.
- `robot_local_control`: Supervisa las funciones de navegación y localización, asegurando una ejecución segura mediante la monitorización de sensores y actuadores.
- `sensors`: Define los sensores en formato URDF y transforma la nube de puntos 3D a un escaneo láser 2D.
- `localization`: Realiza estimaciones de estado mediante fusión de sensores, transformando coordenadas GPS a UTM.
- `navigation`: Controla el movimiento y la navegación autónoma del robot, limitando su velocidad y monitoreando su seguridad mediante el controlador `robotnik_safety_controller`.

Estos componentes y paquetes de software permiten que el nodo robot mantenga una operación robusta y precisa en aplicaciones de conducción remota y navegación en entornos complejos.

3.3 Nodo de usuario

3.3.1 Cockpits inmersivos

El control del robot se realiza mediante el cockpit F-GT Next Level Racing, que incluye componentes dedicados a la visualización, el control, y la sensación háptica. Estos dispositivos permiten a los usuarios manipular los controles con precisión, dirigiendo el robot de manera efectiva. La Figura 6 muestra los principales componentes hardware del Cockpit para conducción remota. Estos componentes se encargan de proporcionar al operador una experiencia de control inmersiva y responsiva:

1. Visualización: Utiliza las gafas Meta Quest 3 y monitores gaming ASUS VG27AQ para mostrar una vista en 360° del entorno del robot, mejorada con objetos de realidad aumentada. Los usuarios tienen la opción de sumergirse en una vista en primera persona mediante las gafas de realidad aumentada, o bien, pueden optar por controlar el robot mirando únicamente la pantalla del monitor. Esta visualización permite al operador tener una percepción completa del entorno del robot y facilita la toma de decisiones en tiempo real.
2. Control: El volante y los pedales Thrustmaster T150 RS permiten al operador manejar el robot de manera intuitiva, con comandos de dirección y velocidad. La integración con el servidor Python asegura que cada acción del operador se traduzca en movimientos precisos y efectivos del robot.
3. Sensación (Retroalimentación Háptica): Utiliza el chaleco háptico BHaptics Tact-Suit X40, que proporciona retroalimentación táctil basada en la interacción del robot con objetos virtuales y obstáculos.

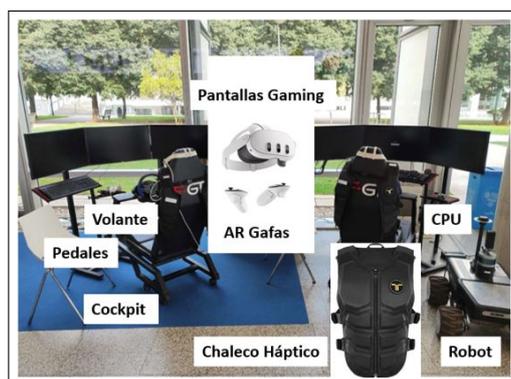


Figure 6: Componentes hardware de cada Cockpit para conducción remota

3.3.2 Componentes software

El nodo usuario es responsable de la interacción y control del robot en el sistema de conducción remota, integrando funciones de realidad aumentada (AR) y proporcionando una experiencia inmersiva al operador. Este nodo se organiza en dos partes principales: el subsistema lógico, compuesto por la plataforma Unity y el servidor Python, y los subsistemas periféricos, que incluyen los componentes de visualización, control y sensación háptica.

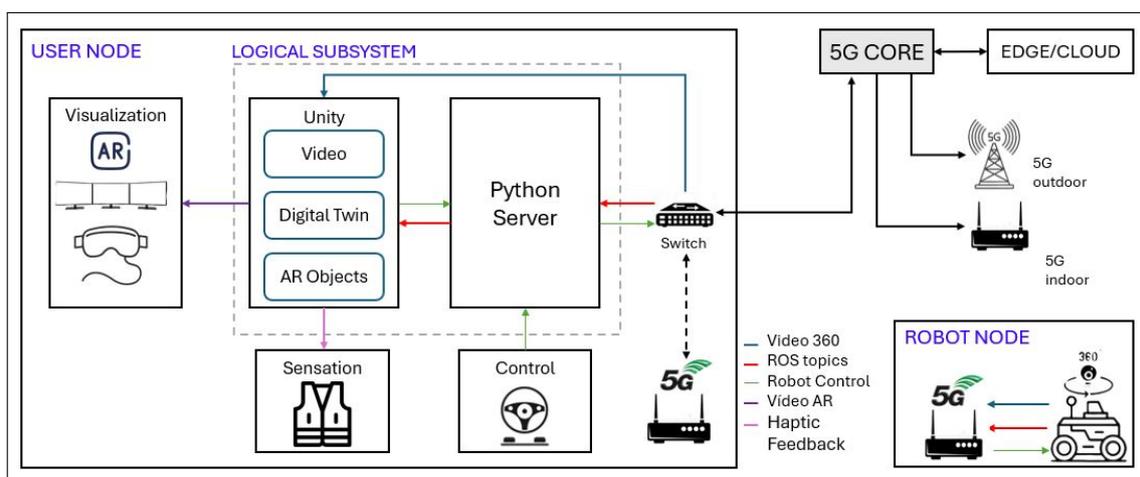


Figure 7: Componentes software del nodo usuario y flujo de comunicación usuario-robot

3.3.2.1 Subsistema Lógico

El subsistema lógico es el núcleo del nodo usuario y gestiona la comunicación, visualización e integración de objetos de realidad aumentada en la interfaz de control del robot. Consta de los siguientes componentes:

1. Servidor Python: Este servidor actúa como intermediario entre el nodo usuario y el nodo robot, estableciendo una comunicación bidireccional a través de la red 5G. A través de sockets TCP y el uso de la librería ROSlibpy, el servidor Python se suscribe a los topics de ROS del robot para recibir datos de telemetría en tiempo real, como posición, orientación, nivel de batería y velocidad. Además, transmite los comandos de control remoto enviados por el operador mediante el uso de la librería Pygame, que incluyen avanzar, retroceder, girar y detener el robot.
2. Unity: Esta aplicación gestiona la visualización inmersiva y la superposición de objetos AR en la transmisión de video en 360° que proviene del robot. Los datos de video en tiempo real son procesados mediante GStreamer y sincronizados con los objetos AR, permitiendo al operador ver una representación enriquecida del entorno físico del robot. Además, Unity envía señales de retroalimentación háptica y visual al sistema de control y sistema de sensación háptica, brindando una experiencia inmersiva y responsiva al usuario cuando el robot interactúa con elementos del entorno virtual.

3.3.2.2 Subsistemas Periféricos

Estos componentes se encargan de proporcionar al operador una experiencia de control inmersiva y responsiva:

1. Visualización: Utiliza el plugin Open XR y la aplicación MetaQuest Link para conectar con las gafas Meta Quest 3.

2. Control: Además de los drivers para comunicar con el volante y los pedales Thrustmaster T150 RS, se usa la librería Pygame de Python para introducir comandos de dirección y velocidad.
3. Sensación (Retroalimentación Háptica): Utiliza el SDK de BHaptics para enviar sensaciones hápticas al chaleco BHaptics TactSuit X40. Unity envía señales de vibración al chaleco cada vez que el robot impacta con un objeto en el entorno AR, lo que permite al operador “sentir” estos impactos en tiempo real, mejorando la inmersión y el control.

3.3.2.3 Flujo de Comunicación

El flujo de comunicación entre el nodo usuario y el nodo robot sigue la estructura mostrada en la Figura 7. Los principales canales de datos incluyen:

- Video en 360°: Transmitido desde el nodo robot al nodo usuario para visualización en tiempo real.
- Topics de ROS: Información de telemetría enviada desde el nodo robot al servidor Python.
- Control del Robot: Comandos enviados desde el nodo usuario al nodo robot para dirección y movimiento.
- Feedback Háptico: Señales de retroalimentación que activan vibraciones en el chaleco háptico cuando el robot impacta con objetos virtuales, permitiendo al operador sentir estos impactos.
- Visualización AR: Superposición de objetos de realidad aumentada en el entorno del robot, sincronizada con la posición del robot en tiempo real.

4 Desarrollo e implementación del sistema

4.1 Lógica de movimiento

Esta sección describe la implementación del sistema de conducción remota basado en Realidad Aumentada (AR), cuyo objetivo principal es lograr una interacción precisa y en tiempo real entre el usuario y el entorno físico del robot. Para alcanzar este objetivo, se desarrollaron diferentes etapas que incluyen la lógica de movimiento, la integración de objetos AR, la gamificación de la experiencia de conducción y la configuración de los equipos necesarios. Cada una de estas etapas contribuye a mejorar la precisión, sincronización y coherencia visual, garantizando así una experiencia de conducción inmersiva y optimizada para el usuario.

4.1.1 Ingreso de Datos

Para implementar la Realidad Aumentada en el proyecto, es crucial garantizar que los datos de los tópicos ROS se reciban correctamente en Unity. Este proceso involucra la interacción de varios componentes clave:

- **Sensores del Robot:** El robot, equipado con sensores GNSS RTK e IMU, proporciona datos precisos de posición y orientación. Los topics de ROS que se leen para obtener estos datos son:
 - Latitud y longitud (GNSS RTK): `/robot/gps/filtered`.
 - Orientación (IMU): `/robot/odometry/filtered_world`.

Estos datos son fundamentales para el funcionamiento de la Realidad Aumentada, ya que permiten la integración precisa de elementos virtuales en el flujo de vídeo del entorno físico.

- **Servidor Proxy y Red 5G:** El servidor proxy actúa como intermediario en la comunicación, y la red 5G proporciona el canal de transmisión de datos. En este servidor, los datos de latitud, longitud y orientación son enviados a Unity para su procesamiento.

4.1.2 Procesamiento

Una vez garantizada la recepción correcta de los datos de posición y orientación en Unity, se implementa un script en lenguaje C# que procesa estos datos para crear el movimiento del escenario en respuesta al movimiento del robot real. El script está diseñado para ajustar la posición del velódromo en la escena de Unity en función de las coordenadas del robot real.

A medida que el robot real se mueve, las coordenadas de latitud y longitud se utilizan para calcular el cambio en la posición del robot. Se ajustan estos cambios a metros y se aplican transformaciones de rotación para alinear el movimiento del escenario.

1. **Conversión de Coordenadas:** La conversión de coordenadas se realiza en dos pasos: el cálculo del offset y el escalado. El objetivo es traducir la posición geográfica (latitud y longitud) del robot a una posición en metros en el entorno virtual de Unity.
 - **Cambio del centro de coordenadas:** Se determina la diferencia entre las coordenadas actuales del robot y un punto de referencia fijo en la simulación (`latitude_ref` y `longitude_ref`, que corresponde al centro del velódromo).

$$\Delta\text{lat} = \text{latitude} - \text{latitude_ref}$$

$$\Delta\text{lon} = \text{longitude} - \text{longitude_ref}$$

- **Escalado:** Convierte estos cambios en coordenadas a metros, ajustando la escala para el entorno de Unity. Los factores de escalado corresponden a los metros por grado de latitud y longitud, respectivamente, en una determinada posición geográfica. Aquí, 110850 y 85380 son aproximaciones de los metros por grado de latitud y longitud, respectivamente, que corresponde a la posición geográfica del velódromo, situado en Valencia (España).

$$\text{latitude_scale} = 110850 \times \Delta\text{lat}$$

$$\text{longitude_scale} = 85380 \times \Delta\text{lon}$$

- **Rotación del eje de coordenadas:** Aplica un ángulo de offset (`anguloOffset`) fijo de 21.1° para ajustar la orientación del escenario de Unity en relación con la dirección Norte medida por el robot real. De esta forma, aplica una rotación que ajusta las coordenadas escaladas (`latitude_scale` y `longitude_scale`) al sistema de referencia de Unity.

$$\text{posX} = \text{latitude_scale} \cdot \cos(\text{anguloOffset}) - \text{longitude_scale} \cdot \sin(\text{anguloOffset})$$

$$\text{posY} = \text{latitude_scale} \cdot \sin(\text{anguloOffset}) + \text{longitude_scale} \cdot \cos(\text{anguloOffset})$$

2. **Transformación:** Las posiciones calculadas `posX` y `posY` reflejan el cambio en la ubicación del robot real en Unity, ajustando la posición del velódromo en el entorno virtual a partir de los datos del sensor GNSS RTK. En cuanto a la orientación, no es necesario realizar ninguna conversión de coordenadas, ya que los datos en grados proporcionados por la IMU son válidos para Unity.

Finalmente, estos pasos de conversión y transformación permiten que el robot digital en Unity se posicione y oriente de forma congruente con el robot físico. Esto asegura que cada movimiento en el escenario virtual sea una representación fiel de la posición y orientación del robot en el entorno real, manteniendo la coherencia y precisión necesarias para la conducción remota.

4.1.3 Actualización del Escenario

En lugar de mover el gemelo digital del robot, el script hace girar la pista del velódromo digital alrededor de él. Esto crea la ilusión de que el robot digital está avanzando mientras que, en realidad, la pista del velódromo digital se ajusta para reflejar el movimiento del robot real.

1. Posicionamiento del Velódromo.

- **Cálculo de Posición objetivo:** La nueva posición del velódromo se determina en función de las coordenadas calculadas para el robot. Para establecer la posición objetivo, se aplican `posX` y `posY` sobre una estructura de tipo `Vector3`.

- Interpolación: Para suavizar la transición del velódromo hacia la nueva posición, se utiliza la función `Vector3.Lerp`, que interpola linealmente entre la posición actual y la posición objetivo.

2. Sincronización del Escenario en Tiempo Real:

- Creación y Función del Objeto Padre: En Unity, el “Objeto Padre” es un contenedor para el velódromo digital y los elementos de Realidad Aumentada (AR) usados en la conducción remota. Este objeto organiza el entorno virtual y facilita la rotación y orientación en Unity (mediante el script `ProxyConnection`), esencial para sincronizar en tiempo real la orientación del escenario con la del robot físico.
- Creación y Función del Objeto Obj_AR: Para organizar los elementos de AR, estos se agrupan en el objeto `Obj_AR`, lo que permite un control centralizado de los objetos virtuales. Además, el script `Position`, integrado en `Obj_AR`, es responsable de gestionar el movimiento de estos objetos en Unity. Al hacerlo, asegura que tanto el velódromo digital como los objetos de AR se muevan y se posicionen correctamente en sincronía con el entorno físico del robot.

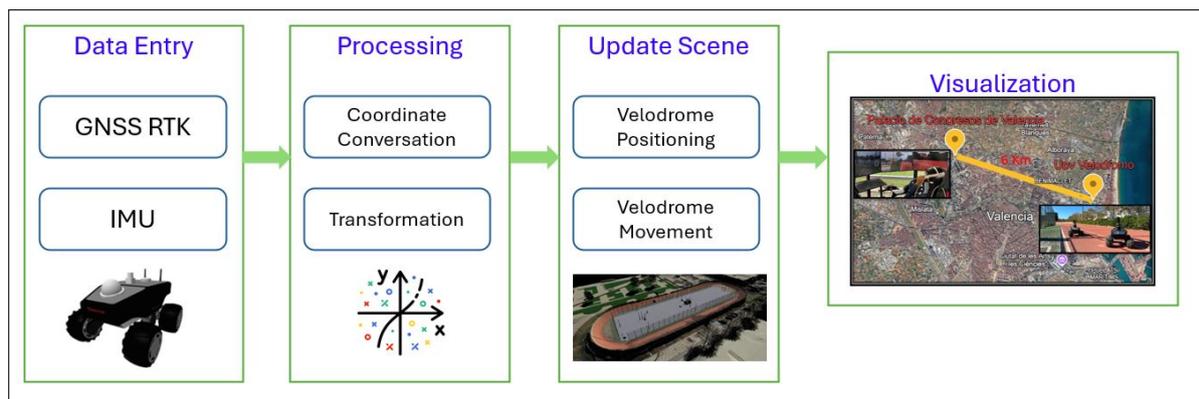


Figure 8: Diagrama de flujo de la lógica de movimiento

4.2 Implementación de Objetos AR

El proceso de implementación de objetos de Realidad Aumentada (AR) en el sistema de conducción remota consiste en integrar elementos virtuales dentro del entorno real del velódromo. Esta integración se logra mediante la superposición de la visualización del gemelo digital del velódromo con el flujo de video en tiempo real, capturado desde la perspectiva del robot en movimiento. De esta manera, se logra una interacción fluida y en tiempo real entre los objetos virtuales y el entorno físico, lo que enriquece la experiencia de conducción remota y aumenta la inmersión del operador.

En esta etapa, los objetos virtuales se ubican dentro del velódromo digital en coordenadas predefinidas, asegurando que su ubicación sea precisa y esté sincronizada con el

entorno físico. Entre estos objetos se incluyen elementos como señalizaciones, cajas de bonificación/penalización y la línea de meta, los cuales cumplen funciones específicas en la experiencia de conducción.

La escala del gemelo digital está ajustada para reflejar con exactitud las dimensiones del velódromo real, lo que minimiza los posibles errores de posición en la simulación. Esto garantiza que los objetos virtuales estén alineados con sus ubicaciones correspondientes en el entorno físico. Por ejemplo, las flechas de señalización que guían la dirección en las curvas del circuito se colocaron con precisión en los puntos clave de cada curva del velódromo, como se muestra en la parte izquierda de la Figura 9). Esta colocación exacta es fundamental para mejorar la orientación del usuario durante la conducción y asegurar una experiencia visualmente coherente.



Figure 9: Integración de Objetos AR sobre velódromo digital

La implementación de los objetos AR sigue un enfoque modular, independiente del resto de objetos de la aplicación y diferente para cada robot. Esto facilita el ajuste y la personalización de los elementos en el entorno, y permite realizar actualizaciones en las posiciones o características de los objetos sin afectar la estructura central del gemelo digital del velódromo. Además, este diseño facilita el mantenimiento de la precisión visual y espacial, especialmente cuando se realizan mejoras en el sistema de navegación o en el propio entorno del velódromo.

La ubicación precisa de los elementos AR garantiza una experiencia intuitiva y funcional, proporcionando al usuario una referencia visual clara y directa de los elementos relevantes durante la interacción remota.

4.3 Gamificación de la experiencia de conducción

Los elementos virtuales en el velódromo digital fueron seleccionados para gamificar la experiencia de conducción remota, incorporando características como flechas de dirección y una línea de meta, elementos comunes en videojuegos. Para fomentar la interacción lúdica, se implementó un sistema de bonificación y penalización que se activa al atravesar cajas en la pista (véase Figura 10).

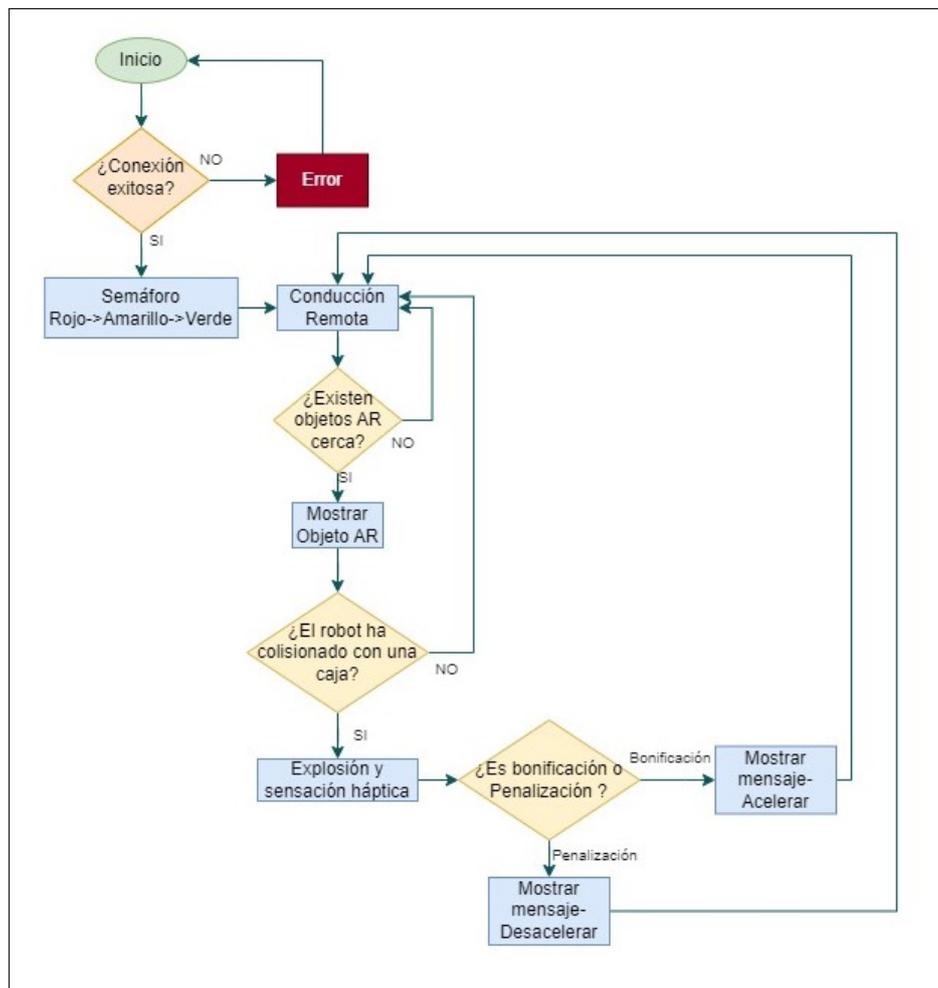


Figure 10: Diagrama de flujo de la gamificación

Cuando el robot real interactúa con estas cajas, el usuario experimenta efectos visuales y hápticos, como explosiones en la pantalla y vibraciones en el chaleco, además de recibir mensajes que indican bonificaciones o penalizaciones en la velocidad. Esto no solo añade un elemento de sorpresa, sino que también mantiene una experiencia visual clara al gestionar la aparición y desaparición de estos objetos según la proximidad del robot.

Adicionalmente, el sistema incluye un semáforo para el inicio de la carrera, que guía al usuario a través de una cuenta regresiva visual. Este mecanismo, junto con alertas sobre posibles colisiones y notificaciones de bonificación, enriquece la inmersión y la interactividad, estableciendo un marco atractivo para la conducción remota.

4.3.1 Aparición y desaparición de objetos

Este apartado explica cómo los objetos AR, incluidos las cajas, flechas de dirección y la línea de meta, aparecen y desaparecen en función de la distancia al robot. Este enfoque optimiza la experiencia del usuario al evitar la saturación visual, mostrando solo los

elementos relevantes en el momento adecuado. La lógica detrás de esta dinámica ayuda a mantener la atención del usuario en los aspectos importantes del recorrido, contribuyendo a una experiencia de conducción más fluida.

El script `Visibility Controller` utiliza un componente `MeshRenderer`, que permite modificar las propiedades visuales del objeto, incluyendo su transparencia. Al inicio, el objeto se encuentra oculto, y su visibilidad se ajusta en tiempo real calculando la distancia entre el jugador y el objeto mediante el método `Vector3.Distance`. Si esta distancia es menor o igual a un umbral definido, denominado `startFade`, el objeto comienza a hacerse visible gradualmente manipulando el componente alfa del color del material del objeto. Este proceso permite que, a medida que el robot se acerca, el objeto se vuelva visible de forma suave. Si la distancia supera el umbral, el objeto se oculta completamente.

4.3.2 Lógica Explosión

En este apartado se detalla el código que activa el sistema de partículas al interactuar el robot con las cajas. Cuando el robot se aproxima a una caja, se desencadena una explosión visual acompañada de una vibración en el chaleco háptico, creando una experiencia inmersiva. Esta explosión visual no solo añade un elemento dinámico a la escena, sino que también está vinculada al sistema de bonificación/penalización, proporcionando al usuario una retroalimentación inmediata sobre su desempeño.

El script `Visibility Controller` utiliza un componente `ParticleSystem` para controlar los efectos visuales de la explosión. Inicialmente, estos sistemas de partículas están desactivados, y su activación depende de la distancia entre el robot y el objeto. Si la distancia es menor o igual al umbral definido por `startFade`, los sistemas de partículas se activan, desencadenando el efecto visual de explosión a medida que el robot se acerca. Cuando el robot entra en contacto con el objeto, identificado a través del método `OnTriggerEnter()`, se activa la colisión. Además de las partículas, se genera una respuesta háptica utilizando la librería `BhapticsLibrary`, proporcionando al usuario no solo una retroalimentación visual, sino también física.

4.3.3 Lógica bonificación/penalización

Este apartado describe el sistema de bonificación y penalización activado cuando el robot entra en contacto con las cajas durante la conducción. Este sistema modifica la velocidad del robot, añadiendo un elemento de imprevisibilidad al recorrido. Dependiendo de un valor aleatorio, el robot puede ser penalizado con una reducción de velocidad o recibir una bonificación que aumenta su velocidad, lo que ofrece una experiencia de conducción más dinámica y desafiante.

El script `Visibility Controller` emplea el método `OnTriggerEnter()` para detectar la colisión del robot con una caja. Tras el contacto, el `BoxCollider` del objeto se desactiva temporalmente para evitar interacciones repetidas mientras dura el efecto (`buffTime`). Por defecto la velocidad del robot es del 80% sobre el máximo de 3 m/s, en función del resultado aleatorio, la velocidad del robot se ajusta al 60% (penalización) o se eleva al 100% (bonificación).

Para gestionar este cambio de velocidad, `Proxy Connection` se encarga de actualizar el tópic de velocidad en el robot real, asegurando que las indicaciones enviadas desde Unity se reflejen en el comportamiento del robot. Esto permite que los cambios se apliquen de manera sincronizada en tiempo real, garantizando la consistencia entre el entorno virtual y el sistema físico. Durante el tiempo que dure la bonificación o penalización, el usuario recibe retroalimentación visual a través de la interfaz, que muestra un mensaje indicando si la velocidad ha disminuido (`Slower`) o se ha incrementado (`Faster`).

4.3.4 Lógica semáforo

Este apartado describe el sistema que controla el inicio de la carrera mediante un semáforo virtual que asegura que el usuario solo pueda comenzar a acelerar el robot una vez que la señal sea verde. Esta mecánica sincroniza el momento de arranque entre el entorno virtual y el robot real, brindando una experiencia de conducción más controlada y realista.

El script emplea el método `Update()` para monitorear la entrada del usuario. Cuando se presiona la tecla “P”, se inicia un temporizador que controla el cambio de colores del semáforo. Este temporizador está vinculado a una animación que primero muestra el color rojo, luego amarillo, y finalmente verde. Solo cuando el semáforo llega al estado verde, se activa un booleano que permite el control de aceleración del robot.

Para garantizar que esta lógica se aplique correctamente en el robot real, Unity se comunica con la clase `ProxyConnection`, que se encarga de enviar un mensaje al tópic de velocidad del robot, indicando que está permitido el inicio del movimiento. Mientras el semáforo está en rojo o amarillo, el robot permanece en estado de reposo, sin poder recibir comandos de velocidad, lo que asegura que el usuario no pueda iniciar la carrera antes de tiempo.

Durante la fase de espera, el semáforo ofrece retroalimentación visual al usuario, mostrando el cambio de colores hasta el momento de la salida. Este sistema asegura que el inicio de la carrera sea sincronizado entre el entorno virtual y el robot físico, ofreciendo un punto de partida consistente y controlado para la competencia.

4.4 Puesta en marcha del cockpit

4.4.1 Configuración de equipos

A continuación, se describe la configuración de los equipos necesarios para implementar la conducción remota inmersiva.

En el nodo usuario, se deben instalar y ajustar las aplicaciones, entornos de desarrollo y frameworks esenciales para asegurar una interacción fluida y en tiempo real entre el usuario y el robot:

- Unity 2021.3.4f1: Plataforma para el desarrollo de la interfaz de usuario .
- Python 3.9.13: Scripts para la comunicación con el robot (librería `ROSLibpy`) y para mapear los inputs de los volantes y pedales (librería `Pygame`).

- GStreamer 1.22.6: Framework de multimedia para la recepción y decodificación del flujo de video de la cámara 360 grados.
- Meta Quest Link y OpenXR: Recursos software para reproducir la aplicación de Unity en las gafas Meta Quest 3.
- BHaptics SDK y BHaptics Player: Software necesario para enviar sensaciones hápticas al chaleco BHaptics Tactsuit X40.

El modem 5G que dota de conectividad al nodo usuario, se conecta físicamente mediante un cable Gigabit Ethernet y se configuran los siguientes parámetros:

- DNN de la red 5G SA para el uso de una slice determinada.
- Dirección IP fija para el PC o CPU que ejecuta el nodo de usuario.
- Configuración de puertos: el puerto 9090 se utiliza para la comunicación bidireccional (Websocket) entre el servidor Python y el robot; el puerto 5200 para recibir el flujo de video RTSP de la cámara de 360 grados; el puerto 22 se emplea para la conexión SSH al PC usuario; y el puerto 5900 se destina a la conexión de escritorio remoto mediante una interfaz gráfica.

En el nodo robot, es necesario instalar y configurar las siguientes aplicaciones, entornos de desarrollo y frameworks:

- MediaMTX: Framework multimedia que se utilizará para optimizar la transmisión del flujo de video proveniente de la cámara de 360 grados.

4.4.2 Ejecución de la aplicación

Para utilizar el cockpit se deberán de seguir los siguientes pasos:

1. Conectar el volante y pedales Thrustmaster T150 para iniciar la auto-calibración.
2. Conectar las gafas de realidad mixta Meta Quest 3, y comprobar que se detectan a través de la aplicación Meta Quest Link. En las gafas, realizar la calibración (crear nuevo límite y establecer nivel del suelo), e iniciar el servicio Quest Link.
3. Conectar el chaleco utilizando la aplicación bHapticsPlayer.
4. Encender el robot y esperar aproximadamente 1 minuto a que arranquen todos los servicios de ROS.
5. Abrir el archivo IPs.txt y configurar la IP del robot.
6. Lanzar el servidor de Python desde línea de comandos.
7. Ejecutar la aplicación de Unity, bien desde el editor o bien desde la build.

5 Integración y demostración

5.1 Pruebas iniciales: AR basada en marcadores

Las pruebas iniciales de AR se llevaron a cabo mediante la tecnología de Vuforia, buscando que la cámara 360° detectara códigos QR en el velódromo para activar la aparición de objetos de realidad aumentada en tiempo real. En una fase posterior, se exploró la posibilidad de utilizar características específicas del entorno, como mallas o arbustos, para desencadenar los objetos virtuales sin necesidad de marcadores visibles. No obstante, esta aproximación no fue viable, ya que la cámara 360° con efecto de ojo de pez solo lograba detectar estos elementos a distancias muy cortas, lo cual limitaba la efectividad en el contexto de la conducción remota.

Teóricamente, una solución que podría haber mitigado estas limitaciones consistía en “extender” la visión esférica de la cámara 360° a una proyección plana, dividiendo la imagen en varias secciones para abarcar los 360° del entorno. Este enfoque permitiría mejorar la detección en cada una de las cuatro vistas (frontal, trasera, izquierda y derecha) mediante un proceso de transformación de coordenadas esféricas a cartesianas. Sin embargo, aunque esta solución resolvería parcialmente la distorsión y mejoraría la precisión de detección, resultaría computacionalmente costosa y compleja para la aplicación en tiempo real.

Dado el alto costo de procesamiento y las limitaciones para la detección a mayores distancias, se optó por implementar una solución de AR basada en localización (descrita en la Sección 4), que ofreció una experiencia de usuario más coherente y precisa en el contexto de la conducción remota. Este sistema utiliza datos de GNSS RTK e IMU en lugar de la cámara 360 para realizar el posicionamiento AR. Este cambio permitió la colocación precisa de objetos virtuales en ubicaciones específicas del velódromo, ofreciendo a los usuarios una percepción espacial mejorada y una experiencia inmersiva alineada con los requisitos de la conducción remota.

5.2 Demostración: IEEE PIMRC 2024

La demostración del sistema de conducción remota se llevó a cabo en el Simposio Internacional Anual IEEE 2024 sobre Comunicaciones de Radio Personales, en Interiores y Móviles (IEEE PIMRC 2024), uno de los eventos más destacados organizados por la IEEE Communications Society (ComSoc) en el ámbito de las comunicaciones inalámbricas y redes. Celebrado en Valencia (España) del 2 al 5 de septiembre, este evento reunió a expertos de todo el mundo para discutir avances en tecnologías inalámbricas y aplicaciones emergentes, ofreciendo una plataforma única para presentar innovaciones de alto impacto.

Durante el evento, los asistentes tuvieron la oportunidad de experimentar de primera mano el sistema de conducción remota basado en realidad aumentada (AR) y 5G. La demostración se llevó a cabo en un entorno controlado, donde se mostró la capacidad del sistema para sincronizarse en tiempo real y ofrecer una experiencia inmersiva a través de un cockpit equipado con gafas AR y dispositivos de retroalimentación háptica. Este

enfoque permitía a los usuarios visualizar objetos virtuales superpuestos en el entorno físico y controlar el robot de forma precisa, mostrando la viabilidad de la tecnología para futuras aplicaciones en entornos industriales, médicos y educativos.

El IEEE PIMRC 2024 no solo brindó la oportunidad de exponer los resultados del proyecto ante una audiencia técnica, sino que también permitió recoger feedback valioso de expertos en redes 5G y AR, lo que ayudará a orientar futuros desarrollos. La presencia de patrocinadores como Huawei, Rohde & Schwarz, Orange, y VIAVI Solutions realzó la importancia del evento y subrayó el interés industrial en las tecnologías de conducción remota y realidad aumentada.

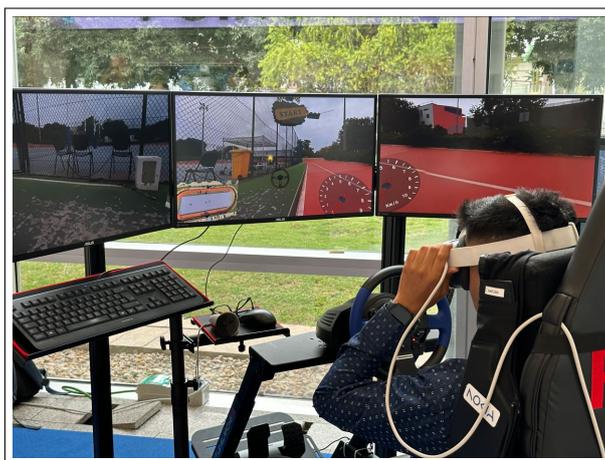


Figure 11: Interacción en tiempo real con el sistema de AR en el cockpit de conducción remota durante el IEEE PIMRC 2024.

6 Resultados

El principal resultado de este proyecto es el desarrollo de un prototipo funcional que combina tecnologías de Realidad Aumentada (AR), redes 5G y telecontrol en tiempo real para la conducción remota de robots. Este prototipo, diseñado para optimizar la interacción y la toma de decisiones en entornos dinámicos y complejos, fue evaluado a través de métricas técnicas de Calidad de Servicio (QoS) y aspectos subjetivos de Calidad de Experiencia (QoE).

En términos de QoS, se analizaron parámetros clave como la latencia extremo a extremo (E2E), la latencia de control, el jitter y la velocidad de transmisión. Estas métricas se midieron en diferentes escenarios para evaluar la estabilidad y el desempeño de las redes utilizadas. Por otro lado, la QoE se basó en encuestas para evaluar la percepción del usuario sobre aspectos como la inmersión, la precisión en el control y la satisfacción general con el sistema.

6.1 Medidas de QoS

6.1.1 Latencia Extremo a Extremo (E2E)

La latencia extremo a extremo (E2E) mide el tiempo total desde que un evento ocurre (impacto con un objeto AR) hasta que el robot físico recibe la señal de modificación. Esta métrica refleja la eficiencia del sistema en la transmisión de datos y la comunicación entre los diferentes elementos.

- En la red Orange 5G, la latencia E2E presentó una media de 305.27 ms, con un percentil 90 de 433.8 ms y un máximo de 472 ms.
- En la red UPV 5G, se obtuvo una media de 281.93 ms, un percentil 90 de 374.4 ms y un máximo de 391 ms.

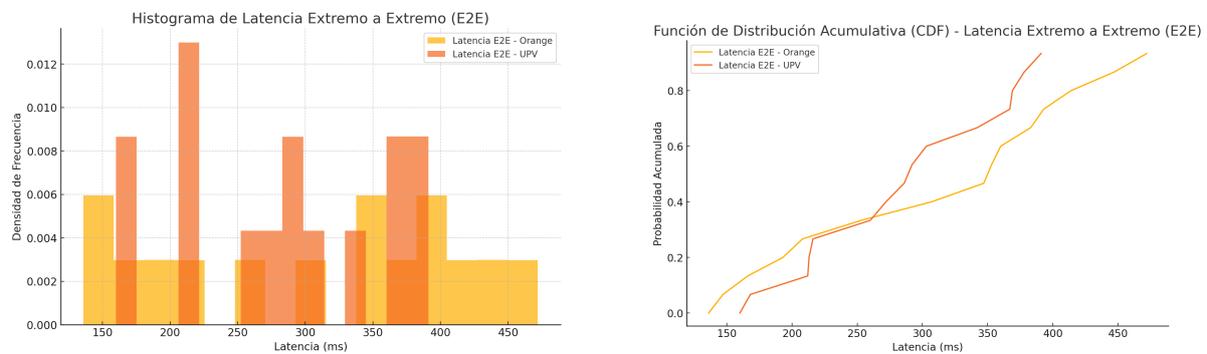


Figure 12: Histograma y CDF de la latencia extremo a extremo (E2E) para ambas redes.

La Figura 12 muestra el histograma comparativo y la función de distribución acumulativa (CDF) para esta métrica, destacando la menor dispersión y estabilidad de la red UPV en comparación con Orange.

6.1.2 Latencia de Control

La latencia de control mide el tiempo total desde que ocurre un evento (impacto con un objeto AR) hasta que Unity confirma que el robot físico ha cambiado su velocidad. Esta métrica incluye el tiempo adicional necesario para que el robot ejecute la acción, siendo un indicador clave del tiempo de respuesta operativo.

- En la red Orange 5G, la latencia de control presentó una media de 1340 ms, con un percentil 90 de 1530 ms y un máximo de 1700 ms.
- En la red UPV 5G, los valores fueron más bajos, con una media de 1042 ms, un percentil 90 de 1200 ms y un máximo de 1300 ms.

La Figura 13 presenta un histograma y una función de distribución acumulativa (CDF) para esta métrica, comparando ambas redes. Los resultados subrayan la menor variabilidad y mayor eficiencia de la red UPV.

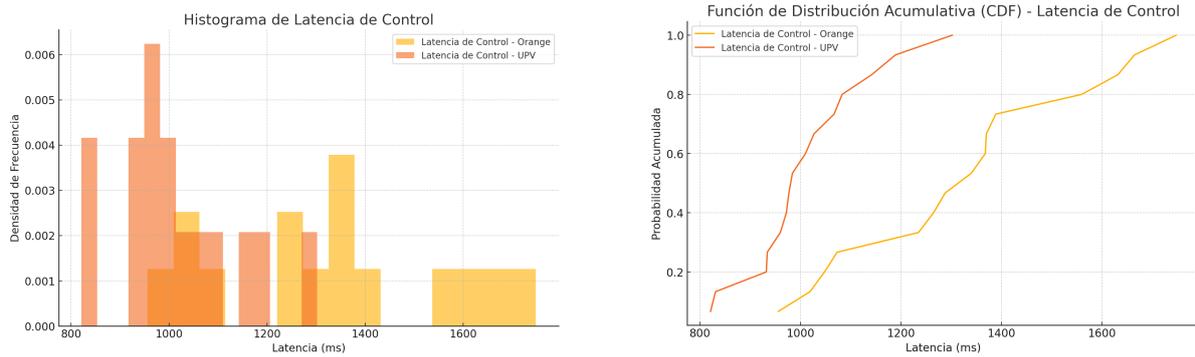


Figure 13: Histograma y CDF de la latencia de Control para ambas redes.

6.1.3 Jitter

El jitter representa la variabilidad en los tiempos de llegada de paquetes consecutivos, siendo un indicador clave de la estabilidad de la red. En la red Orange 5G, se obtuvo un jitter promedio de 30.06 ms, con un percentil 90 de 88.86 ms y un valor máximo de 459.9 ms. En contraste, la red UPV 5G presentó un jitter promedio de 7.89 ms, con un percentil 90 de 14 ms y un valor máximo de 157.5 ms.

La Figura 14 incluye un histograma y una función de distribución acumulativa (CDF) para esta métrica. Los resultados muestran que la red UPV presenta un jitter más concentrado y consistente en comparación con Orange, lo que sugiere mayor estabilidad en condiciones de operación.

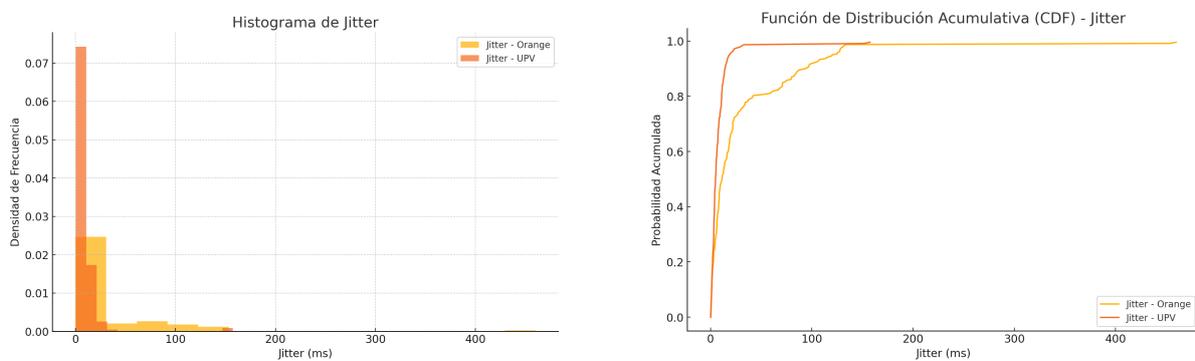


Figure 14: Histograma y CDF del jitter para ambas redes.

6.1.4 Latencia de Red 5G y de Video

La latencia de video es un factor crucial en aplicaciones de conducción remota y realidad aumentada, ya que influye directamente en la sincronización entre los datos visuales y las acciones realizadas por el usuario. Valores elevados de latencia pueden generar desincronizaciones entre los objetos virtuales y el entorno físico, comprometiendo la precisión en la toma de decisiones y afectando la percepción de control. Durante las evaluaciones,

se analizaron dos métricas clave: la latencia de red (RTT) y la latencia de video en tiempo real.

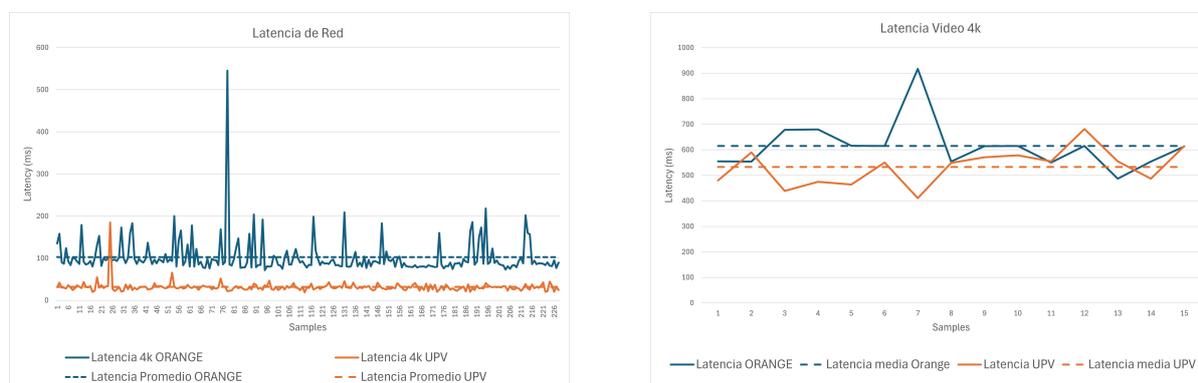


Figure 15: Izquierda: Comparación de latencia de Red; Derecha: Comparación de latencia de video.

- **Latencia de Red:** La red de Orange presentó un valor promedio igual a 103 ms, un valor típico para una red comercial 5G bajo carga moderada. Aunque suficiente para la transmisión de datos básica, esta latencia puede afectar ligeramente la sincronización en aplicaciones críticas. En cambio, la red UPV mostró un promedio significativamente menor, de 32 ms, destacándose como una opción más adecuada para escenarios sensibles al tiempo debido a su baja latencia y alta consistencia. Los valores están representados en la Figura 15.
- **Latencia de Video 4K:** Los resultados mostraron un promedio de 615 ms en la red Orange, un valor que se encuentra en el límite superior de lo aceptable para garantizar una experiencia inmersiva. Este nivel de latencia puede ocasionar desincronizaciones perceptibles en condiciones de alta carga de red. Por otro lado, la red UPV obtuvo un promedio de 533 ms, ofreciendo una transmisión más fluida y sincronizada, adecuada para aplicaciones en tiempo real. Estos resultados destacan la importancia de contar con redes de baja latencia y alta estabilidad para aplicaciones inmersivas como la conducción remota de robots. Los valores están representados en la Figura 15.

6.1.5 Velocidad de Transmisión

En el velódromo, se evaluaron las velocidades de transmisión utilizando dos redes: la red 5G comercial de Orange y la red 5G privada de UPV, esta última operando con un nodo outdoor de 20 MHz. Estas pruebas reflejan el desempeño de ambas redes en un entorno real de operación.

Los resultados mostraron que la red Orange alcanzó un promedio de 55 Mbps, con picos de hasta 98 Mbps y mínimos de 17 Mbps, destacando su capacidad para manejar altos picos de transmisión, aunque con mayor variabilidad. En contraste, la red UPV presentó

un promedio de 17 Mbps, mostrando una transmisión significativamente más estable y consistente, lo que subraya la fiabilidad del nodo outdoor en aplicaciones críticas.

El histograma de la Figura 16 ilustra estas diferencias, mostrando una concentración más estrecha en los valores de la red UPV frente a la mayor dispersión de la red Orange. Adicionalmente, la evolución temporal del bitrate, representada en la Figura 16, refuerza estas observaciones al mostrar las fluctuaciones características de cada red durante la prueba.

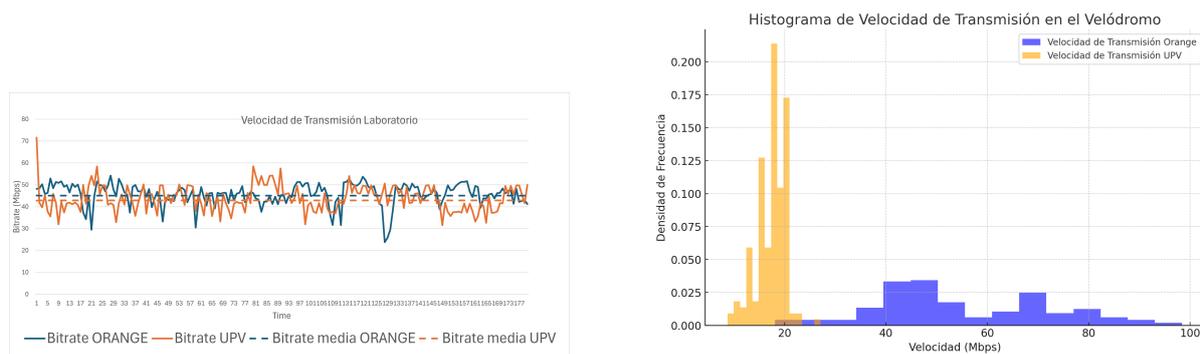


Figure 16: Izquierda: Comparación de velocidad de transmisión en el velódromo.; Derecha: Histograma de velocidad de transmisión en el velódromo.

6.2 Medidas de QoE

Se aplicó una encuesta a 53 participantes durante el evento IEEE PIMRC 2024 para evaluar los siguientes parámetros subjetivos.

- Satisfacción General: El 96,2% de los usuarios calificó con 4 o 5, indicando una alta aceptación del sistema.
- Sensación de Presencia: El 97,9% de los usuarios informó sentirse inmerso en el entorno.
- Compromiso: El 95,8% de los participantes se sintió altamente comprometido durante la teleconducción.
- Percepción de Control: El 75% de los usuarios percibió un buen nivel de control, aunque algunos informaron ligeras pérdidas de precisión.
- Integración Sensorial: Un 79,2% valoró positivamente la combinación de feedback visual y háptico.
- Carga Cognitiva: La tarea fue percibida como demandante para el 33,3% de los usuarios, pero aceptable para la mayoría.
- Futuras Aplicaciones: El 80,8% de los participantes consideró viable la expansión de sistemas ciberfísicos similares a corto plazo.

La integración de la realidad aumentada (AR) mejoró la experiencia del usuario al proporcionar información visual adicional que facilitó la percepción del entorno y la toma de decisiones. Los usuarios informaron un aumento en la sensación de inmersión debido a la superposición de objetos virtuales sobre el entorno físico, lo que permitió una conducción más precisa y rápida.

Sin embargo, algunos participantes experimentaron desincronizaciones menores entre los objetos AR y el entorno físico bajo condiciones de alta carga de red, lo que subraya la importancia de una baja latencia para mantener la coherencia entre los entornos físico y virtual.

7 Conclusiones

Los resultados del sistema de conducción remota de robots, basado en tecnologías de realidad aumentada (AR) y redes 5G, confirmaron la viabilidad del enfoque planteado y cumplieron con los objetivos técnicos. Se logró desarrollar e integrar una aplicación de AR que permite la interacción en tiempo real entre robots físicos y objetos virtuales, evaluar su rendimiento en redes 5G privadas y comerciales, y analizar la experiencia de usuario en términos de precisión, inmersión y facilidad de uso.

Las mediciones de latencia extremo a extremo (E2E) destacaron la superioridad de la red privada 5G de la UPV, con un promedio de 281.93 ms frente a los 305.27 ms de la red comercial de Orange. La latencia de control también fue menor en la red UPV, con 1042 ms en comparación con los 1340 ms de Orange, evidenciando la estabilidad de la red UPV en escenarios críticos. La latencia de red (RTT) promedió 32 ms en la red UPV y 103 ms en Orange, favoreciendo la transmisión en tiempo real.

En términos de velocidad de transmisión, Orange alcanzó picos de 98 Mbps, mientras que la red UPV ofreció un promedio más estable de 17 Mbps en el velódromo y 42 Mbps en el laboratorio. Estos resultados subrayan que, aunque Orange ofrece mayor capacidad de pico, la estabilidad de la red UPV es clave para aplicaciones críticas.

La experiencia de usuario también fue positiva, destacando la capacidad del sistema para ofrecer una interacción intuitiva e inmersiva. Estos hallazgos confirman el potencial de las tecnologías de AR y redes 5G para aplicaciones de conducción remota en tiempo real.

7.1 Limitaciones del trabajo

A pesar de los logros alcanzados, el proyecto presentó ciertas limitaciones:

- Dependencia de la infraestructura de red: El rendimiento del sistema es óptimo en redes de baja congestión; sin embargo, en redes saturadas, la latencia afecta la precisión de la conducción remota, limitando su aplicación en entornos con calidad de red variable.
- Sincronización sensorial: La integración del feedback visual y háptico, aunque satisfactoria en general, mostró desajustes en escenarios de alta latencia, afectando la

respuesta en tiempo real.

- Alcance de las pruebas: Al realizarse en un entorno controlado y con un número reducido de usuarios y escenarios, se recomienda ampliar las pruebas en entornos variados y con mayor participación para obtener una evaluación más robusta del sistema.

7.2 Futuras líneas de investigación

Basándose en los resultados obtenidos, se identifican áreas clave para potenciar el sistema y expandir su aplicabilidad:

- AR y dispositivos de visualización: Explorar el uso de gafas VR/AR en lugar de pantallas para mejorar la inmersión y la experiencia del usuario en aplicaciones de conducción remota.
- Optimización de la red 5G: Implementar network slicing en redes 5G para garantizar la calidad del servicio (QoS) y explorar tecnologías como WebRTC para reducir la latencia en transmisión de video en tiempo real.
- Sincronización sensorial: Mejorar la integración entre el feedback visual y háptico, incluyendo sensores adicionales, como acelerómetros avanzados, para incrementar el realismo.
- Integración Sensorial: Mejorar la sincronización entre el feedback visual y háptico, e integrar sensores de fuerzas G para incrementar la sensación de realismo.
- Adaptabilidad a la Red: Desarrollar algoritmos inteligentes que ajusten dinámicamente la calidad de transmisión según las condiciones de la red, mejorando la estabilidad en aplicaciones críticas.
- Ampliación del ámbito de pruebas: Realizar evaluaciones en entornos variados con un mayor número de usuarios y escenarios, incluyendo redes con diferentes niveles de carga, para validar la escalabilidad y robustez del sistema.

Referencias

- [1] Xingqin Lin. An overview of 5g advanced evolution in 3gpp release 18. *IEEE Communications Standards Magazine*, 6(3):77–83, 2022.
- [2] Mikko A. Uusitalo et al. 6g vision, value, use cases and technologies from european 6g flagship project hexa-x. *IEEE Access*, 9:160004–160020, 2021.
- [3] Maria Crespo-Aguado, Raul Lozano, Fernando Hernandez-Gobertti, Nuria Molner, and David Gomez-Barquero. Flexible hyper-distributed iot–edge–cloud platform for real-time digital twin applications on 6g-intended testbeds for logistics and industry. *Future Internet*, 16(11):431, 2024.
- [4] Cheng-Xiang Wang, Xiaohu You, Xiqi Gao, Xiuming Zhu, Zixin Li, Chuan Zhang, Haiming Wang, Yongming Huang, Yunfei Chen, Harald Haas, et al. On the road to 6g: Visions, requirements, key technologies, and testbeds. *IEEE Communications Surveys & Tutorials*, 25(2):905–974, 2023.
- [5] Dazhi He, Wanting Wang, Yin Xu, Xiuxuan Huang, Hao Cheng, Xiaohan Duan, Yihang Huang, Hanjiang Hong, Yiwei Zhang, and Wenjun Zhang. Overview of physical layer enhancement for 5g broadcast in release 16. *IEEE Transactions on Broadcasting*, 66(2):471–480, 2020.
- [6] Sandeep Narayanan Kadan Veedu, Mohammad Mozaffari, Andreas Höglund, Emre A Yavuz, Tuomas Tirronen, Johan Bergman, and Y-P Eric Wang. Toward smaller and lower-cost 5g devices with longer battery life: An overview of 3gpp release 17 redcap. *IEEE Communications Standards Magazine*, 6(3):84–90, 2022.
- [7] Mikko A Uusitalo et al. 6g vision, value, use cases and technologies from european 6g flagship project hexa-x. *IEEE access*, 9:160004–160020, 2021.
- [8] Gerhard P Fettweis. The tactile internet: Applications and challenges. *IEEE vehicular technology magazine*, 9(1):64–70, 2014.
- [9] Miguel Rodríguez Jodar. *Robótica conectada mediante 5G para la gestión de flotas de vehículos*. PhD thesis, Universitat Politècnica de València, 2020.
- [10] Jose Guillermo Guarnizo Marin, Daniela Bautista Díaz, and Juan Sebastián Sierra Torres. Una revisión sobre la evolución de la robótica móvil. 2021.
- [11] Yuanwei Liu, Xiao Liu, Xinyu Gao, Xidong Mu, Xiangwei Zhou, Octavia A Dobre, and H Vincent Poor. Robotic communications for 5g and beyond: Challenges and research opportunities. *IEEE Communications Magazine*, 59(10):92–98, 2021.
- [12] Markus Aleksy, Fan Dai, Nima Enayati, Peter Rost, and Guillermo Pocovi. Utilizing 5g in industrial robotic applications. In *2019 7th International Conference on Future Internet of Things and Cloud (FiCloud)*, pages 278–284. IEEE, 2019.

- [13] Ravi Raj and Andrzej Kos. A comprehensive study of mobile robot: history, developments, applications, and future research perspectives. *Applied Sciences*, 12(14):6951, 2022.
- [14] Adarsh Jagan Sathyamoorthy, Kasun Weerakoon, Mohamed Elnoor, and Dinesh Manocha. Using lidar intensity for robot navigation. *arXiv preprint arXiv:2309.07014*, 2023.
- [15] Guilherme N DeSouza and Avinash C Kak. Vision for mobile robot navigation: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 24(2):237–267, 2002.
- [16] Ignacio Rodriguez, Rasmus Suhr Mogensen, Andreas Fink, Taus Raunholt, Søren Markussen, Per Hartmann Christensen, Gilberto Berardinelli, Preben Mogensen, Casper Schou, and Ole Madsen. An experimental framework for 5g wireless system integration into industry 4.0 applications. *Energies*, 14(15):4444, 2021.
- [17] Alejandro Castilla García. *Telepresencia 5G. Sistema de control remoto y navegación inmersiva de vehículos*. PhD thesis, Universitat Politècnica de València, 2022.
- [18] Roberto Minerva, Gyu Myoung Lee, and Noel Crespi. Digital twin in the iot context: A survey on technical features, scenarios, and architectural models. *Proceedings of the IEEE*, 108(10):1785–1824, 2020.
- [19] Dylan D Schmorrow and Cali M Fidopiastis. *Augmented Cognition: 15th International Conference, AC 2021, Held as Part of the 23rd HCI International Conference, HCII 2021, Virtual Event, July 24–29, 2021, Proceedings*, volume 12776. Springer Nature, 2021.
- [20] Stefan Stumpp, Tobias Knopf, and Daniel Michelis. User experience design with augmented reality (ar). In *Proceedings of the ECIE 2019 14th European conference on innovation and entrepreneurship*, pages 1032–1040, 2019.
- [21] Julie Carmigniani, Borko Furht, Marco Anisetti, Paolo Ceravolo, Ernesto Damiani, and Misa Ivkovic. Augmented reality technologies, systems and applications. *Multi-media tools and applications*, 51:341–377, 2011.
- [22] Anna Syberfeldt, Oscar Danielsson, and Patrik Gustavsson. Augmented reality smart glasses in the smart factory: Product evaluation guidelines and review of available products. *Ieee Access*, 5:9118–9130, 2017.
- [23] Chengjun Chen, Zhongke Tian, Dongnian Li, Lieyong Pang, Tiannuo Wang, and Jun Hong. Projection-based augmented reality system for assembly guidance and monitoring. *Assembly Automation*, 41(1):10–23, 2021.
- [24] Ana Javornik. Augmented reality: Research agenda for studying the impact of its media characteristics on consumer behaviour. *Journal of Retailing and Consumer Services*, 30:252–261, 2016.

- [25] Aidan Fuller, Zhong Fan, Charles Day, and Chris Barlow. Digital twin: Enabling technologies, challenges and open research. *IEEE access*, 8:108952–108971, 2020.
- [26] Mark Billinghurst, Adrian Clark, Gun Lee, et al. A survey of augmented reality. *Foundations and Trends® in Human–Computer Interaction*, 8(2-3):73–272, 2015.
- [27] Raúl Lozano Teruel. *Aplicaciones de guantes hápticos para el control remoto de robots autónomos conectados por 5G*. PhD thesis, Universitat Politècnica de València, 2021.
- [28] Johanna Pirker, Enrica Loria, Saeed Safikhani, Andreas Künz, and Sabrina Rosmann. Immersive virtual reality for virtual and digital twins: A literature review to identify state of the art and perspectives. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pages 114–115. IEEE, 2022.
- [29] Marco Giordani, Michele Polese, Marco Mezzavilla, Sundeep Rangan, and Michele Zorzi. Toward 6g networks: Use cases and technologies. *IEEE communications magazine*, 58(3):55–61, 2020.
- [30] Ilja T Feldstein and Stephen R Ellis. A simple video-based technique for measuring latency in virtual reality or teleoperation. *IEEE Transactions on Visualization and Computer Graphics*, 27(9):3611–3625, 2020.
- [31] David G Black, Dragan Andjelic, and Septimiu E Salcudean. Evaluation of communication and human response latency for (human) teleoperation. *IEEE Transactions on Medical Robotics and Bionics*, 2024.
- [32] Harsh Tataria, Mansoor Shafi, Andreas F Molisch, Mischa Dohler, Henrik Sjöland, and Fredrik Tufvesson. 6g wireless systems: Vision, requirements, challenges, insights, and opportunities. *Proceedings of the IEEE*, 109(7):1166–1199, 2021.
- [33] Joachim Sachs, Lars AA Andersson, José Araújo, Calin Curescu, Johan Lundsjö, Göran Rune, Eckehard Steinbach, and Gustav Wikström. Adaptive 5g low-latency communication for tactile internet services. *Proceedings of the IEEE*, 107(2):325–349, 2018.
- [34] Aaron Raymond See, Jose Antonio G Choco, and Kohila Chandramohan. Touch, texture and haptic feedback: a review on how we feel the world around us. *Applied Sciences*, 12(9):4686, 2022.