**2. Data Transmission Method Evaluation and Design Decision**

*For Camera Motion Tracking and Real-Time Scene Synchronization*

**2.1 Introduction**

This section evaluates three communication interfaces—USB, Ethernet, and Wi-Fi—for transmitting fused IMU and IR tracking data from the microcontroller to external systems. The system’s objective is to track camera motion in real time and send this information to rendering engines or media servers to dynamically adjust the scene displayed on LED walls or chroma key backgrounds in virtual production environments.

**2.2 Communication Options Overview**

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature** | **USB** | **Ethernet** | **Wi-Fi (ESP32)** |
| Interface Type | USB CDC (Virtual COM) | UDP over Ethernet | UDP over Wi-Fi (802.11n) |
| Topology | Point-to-point | Broadcast / Multicast capable | Broadcast / Multicast capable |
| Typical Data Rate | ~1 Mbps | 10/100 Mbps | ~2–5 Mbps practical |
| Latency | Low | Very low (≤1 ms typical) | Moderate (variable, 10–30 ms) |
| Power Consumption | Low | Moderate | Higher (due to radio activity) |
| Driver Requirements | Built-in OS drivers | No drivers; uses sockets | No drivers; uses sockets |
| Integration Target | Debugging, PC logging | Game engines, studio networks | Mobile or wearable systems |
| Hardware Complexity | Low | Medium (PHY + RJ45) | Low (ESP32 SoC) |
| EMI Susceptibility | Medium | Low (shielded, wired) | High (wireless interference) |
| Reliability | Good | Excellent | Variable |

**2.3 System Requirements**

The communication interface must:

* Stream fused motion data at 20–100 Hz
* Maintain low latency and jitter to ensure scene stability
* Integrate into standard studio or stage production networks
* Avoid reliance on proprietary or platform-specific drivers
* Be robust against environmental noise and suitable for live use

**2.4 Design Decision**

After evaluating the options, **Ethernet** is selected as the primary communication interface for the following reasons:

* Offers low-latency and high-bandwidth data transmission
* Supports open protocols such as UDP and OSC used in game engines (e.g., Unreal) and real-time media systems
* Provides reliable and deterministic performance, critical for accurate scene rendering
* Integrates well into professional audio-visual infrastructure

**USB** remains useful for debugging during development. **Wi-Fi** may be revisited in future mobile or battery-operated configurations, but is not suited for high-reliability studio environments at this stage.

**2.5 Next Steps**

* STM32 microcontroller with built-in Ethernet MAC (e.g., STM32F407, F429, H743)
* Elect an Ethernet PHY (e.g., LAN8720 or DP83848) via RMII
* Adding RJ45 Ethernet connector with integrated magnetics
* Considering UDP or OSC protocol for sending fused motion data
* Consider time synchronization if multiple trackers or cameras are involved