Ricoh GR IV – Authentic Capture Firmware Proposal

Camera-Level Cryptographic Provenance for Real Photographs

Author: Dante Sisofo
Status: Open Proposal / Open Source Concept
Target Platform: Ricoh GR IV (newest models)

Abstract

This proposal outlines a firmware-level feature for the Ricoh GR IV that enables **cryptographic authenticity and provenance** for photographs at the moment of capture.

At shutter press, the camera computes a **SHA-256 hash** of the captured image, **digitally signs it using a device-held private key**, and embeds a **tamper-evident authenticity record** directly into the file metadata.

The result is a photograph that can be independently verified as:

- Captured by a real Ricoh camera sensor
- Generated at a specific moment in time
- Unmodified since capture

This system does **not** claim to prove "human intent," but it *does* establish a strong, honest, cryptographically verifiable chain of authenticity starting at the camera itself.

Problem Statement

Generative AI has made it trivial to produce convincing synthetic images that are indistinguishable from real photographs at the file level.

Current solutions suffer from one or more of the following issues:

- Rely on centralized platforms or vendors
- Are applied *after* capture rather than at the source
- Can be trivially forged or stripped
- Are opaque, proprietary, or ecosystem-locked

Photographers, journalists, archivists, and historians need a **source-of-truth** mechanism that begins at the camera sensor.

Design Goals

This proposal prioritizes:

- **Cryptographic correctness**
- **Minimal trust assumptions**
- **Open verification**
- **Backward compatibility with existing workflows**
- **User choice and privacy**
- **No dependence on blockchains or third-party platforms**

Non-Goals (Important)

This system does **not**:

- Prove that a human intentionally pressed the shutter
- Prevent photographing screens, prints, or projections
- Prevent editing (only makes edits detectable)
- Replace artistic judgment or context

Its claim is precise and honest: **“This file originated from a real Ricoh camera and has not been altered.”**

High-Level Overview

When **Authentic Capture** is enabled in firmware:

1. The camera captures an image normally.
2. After encoding (JPEG / RAW), the camera:
3. Computes a **SHA-256 hash** of the canonical image data.
4. Builds a **capture manifest** (metadata summary).
5. Digitally signs the manifest hash using a **device private key**.
6. The authenticity block is embedded into the image metadata.
7. Anyone can later verify the file using a public verification tool.

Cryptographic Architecture

Hashing

- Algorithm: **SHA-256**
- Purpose: Detect any post-capture modification.
- Scope:
- JPEG: entire file excluding the authenticity block.
- RAW/DNG: sensor data blocks + critical capture tags.

Signing

- Algorithm: **Ed25519** (preferred) or ECDSA P-256.
- Private key:
- Generated per-device.
- Stored in secure hardware (secure element / TEE).
- Never exportable.
- Public key:
- Certified by Ricoh via a manufacturer certificate chain.

Capture Manifest (Minimum Fields)

```
{
  "camera_model": "RICOH GR IV",
  "firmware_version": "vX.Y.Z",
  "timestamp_utc": "2026-01-22T12:34:56Z",
  "device_id": "pseudonymous-or-serial",
  "image_format": "JPEG | DNG | RAW",
  "exposure": {
    "shutter": "1/500",
    "aperture": "f/8",
    "iso": 400
  },
  "hash_algorithm": "SHA-256",
  "image_hash": "hex-encoded-hash",
  "signature_algorithm": "Ed25519"
}
```

The manifest itself is hashed and signed.

Metadata Storage Options

Option A: MakerNotes (Fastest)

- Pros: Minimal engineering effort.
- Cons: Some software strips MakerNotes.

Option B: Embedded XMP (Recommended Baseline)

- Pros: Widely supported, editable-aware.
- Cons: Can be stripped by aggressive platforms.

Option C: C2PA-Compatible Container (Optional)

- Pros: Industry interoperability.
- Cons: Higher complexity and conformance burden.

Recommendation:
Ship Option B first. Add Option C later as an optional mode.

Firmware Integration Point

The authenticity pipeline runs **after encoding but before file finalization**:

1. Sensor capture
2. Image processing pipeline
3. Encode JPEG / RAW
4. Compute canonical SHA-256
5. Build capture manifest
6. Sign manifest hash
7. Embed authenticity block
8. Finalize file

Performance Considerations

- Single-shot latency target: **< 50 ms**
- Burst mode:
- User-selectable (always sign / first-frame only / disabled)
- Optimizations:
- Hardware SHA acceleration if available
- Ed25519 for speed
- Non-blocking UI thread

Firmware Pseudocode (Conceptual)

```
int authentic_capture(const char* filepath, CaptureInfo* info) {
    uint8_t image_hash[32];
    sha256_canonical(filepath, image_hash);

    Manifest m = build_manifest(info, image_hash);
    uint8_t manifest_hash[32];
    sha256(m.bytes, m.len, manifest_hash);

    uint8_t signature[64];
    secure_sign(manifest_hash, signature);

    embed_auth_block(filepath, m, signature);
    return OK;
}
```

Verification Tool (Required)

Ricoh must ship:

- Desktop verifier (macOS / Windows / Linux)
- CLI tool for journalists and archivists
- Public documentation + test vectors

Verification steps:

1. Parse authenticity block.
2. Recompute canonical hash.
3. Compare stored hash.
4. Verify signature using Ricoh certificate chain.
5. Display result:
6. ✅ Authentic Ricoh capture
7. ✕ Modified after capture
8. ⚠️ Authenticity data missing/stripped

Open-source reference implementation strongly recommended.

Key Provisioning & Trust

Factory Process

- Generate keypair inside secure hardware.
- Issue per-device certificate signed by Ricoh CA.
- Store certificate chain on device.

Revocation

- Publish certificate revocation list (CRL).
- Verifiers optionally check revocation when online.

User Controls

Menu options:

- Authentic Capture: Off / On
- Proof Format: Basic / C2PA
- Burst Behavior: Always / First Frame / Off
- Privacy:
- Include Serial Number
- Use Pseudonymous Device ID

Default: **Off** (opt-in).

Backward Compatibility

- Files remain standard JPEG / RAW.
- Software ignoring authenticity data continues to function normally.
- Authenticity block is additive, not disruptive.

Why This Fits Ricoh

- Aligns with Ricoh’s minimalist, photographer-first ethos.
- Avoids locked ecosystems.
- Supports independent creators and journalists.
- Technically honest and verifiable.
- Future-proof without hype.

Conclusion

Authenticity must begin **at the source**.

This proposal offers Ricoh a way to lead not through marketing, but through **cryptographic integrity, openness, and trust** — empowering photographers to prove what matters without surrendering control.

The camera becomes not just a tool for seeing, but a **witness**.

Contact / Attribution

Proposal by Dante Sisofo
Open for discussion, critique, and implementation.