https://doi.org/10.2352/EI.2026.38.10.HVEI-212

The preservation and repackaging of color and tonal metadata

Coleman Earlywine and Henry Dietz
Department of Electrical and Computer Engineering, University of Kentucky; Lexington, Kentucky

Abstract

In raw imaging workflows, metadata is as critical as pixel data. Raw files do not merely store sensor measurements; they also encode the information required to interpret those measurements as color, brightness, and dynamic range. When computational methods modify raw-domain image data, this interpretive metadata is often lost, invalidated, or omitted, producing substantial rendering inconsistencies in downstream software. `PARSEK` *(the Probabilistic Alignment Raw Stitcher Experiment from Kentucky) exposes this problem clearly: although its super-resolution output preserves useful raw-domain image content, results saved without appropriate metadata can exhibit severe color and tonal shifts when opened in conventional raw processors. This paper presents* **`KYDNG`**, *a DNG repackaging pipeline designed to preserve perceptual consistency by embedding reconstructed raw image data together with metadata derived from the source capture. The implementation writes raw-specific structural tags, including CFA pattern and repeat dimensions, and propagates key camera-dependent rendering metadata, including ColorMatrix1, AsShotNeutral, and black and white levels, while also embedding a JPEG preview. The resulting files are intended to be interpreted by standard raw development tools using the same camera-consistent rendering assumptions as the original capture.*

Introduction

The present work grew out of a limit observed during the development of `PARSEK`, the *Probabilistic Alignment Raw Stitcher Experiment from Kentucky* [1]. `PARSEK` accepts multiple input images, *preferably raw captures,* aligns them, and computes a super-resolution output using a statistical reconstruction method. When the inputs are raw, the data are maintained in a linear raw color space throughout processing to preserve numerical fidelity. Thus, even though the output contains values for all three-color channels at each pixel location, rather than the single-channel-per-site sampling used by a conventional CFA image, the result remains fundamentally a raw-domain image representation.

The difficulty is not in computing the output image, but in packaging it so that other software interprets it correctly. Common raw formats do not provide a simple or well-supported mechanism for replacing image data while preserving the metadata needed for proper rendering, especially metadata governing color and tonality. Earlier `PARSEK` workflows therefore produced outputs such as 16-bit PNG files. Those files preserved image values, but not the camera-dependent metadata required by raw processors. As a result, images opened in downstream software often exhibited strong deviations in white balance, contrast, tonal mapping, and overall color appearance relative to the original raw captures. This can be observed in figure 1.

This is not a `PARSEK`-specific issue. More generally, any computational imaging workflow that modifies raw-domain image data must confront the fact that raw rendering is metadata-driven. Denoising, enhancement, demosaicing, and super-resolution pipelines may preserve or improve the underlying sensor-derived signal while still producing visually unsatisfactory results if the associated metadata is absent or inconsistent. A practical output format is therefore needed that can accommodate modified raw-domain image data while also preserving, reconstructing, or re-expressing the metadata required for standard raw rendering pipelines.

Figure 1: *Rendering of .ARW file and PARSEK output*

The Digital Negative (DNG) format [2] is a natural candidate for this role. DNG provides a more standardized container for raw image data and associated metadata than most manufacturer-specific formats. However, DNG packaging alone does not solve the problem. If the metadata embedded in the DNG is incomplete, mismatched, or only partially expressible, standard raw processors may still render the image incorrectly. The central question addressed in this work is therefore not merely whether transformed raw-domain data can be written into a DNG, but whether it can be written together with enough camera-consistent metadata to preserve expected rendering behavior.

To address that question, this paper presents **`KYDNG`**, a DNG repackaging pipeline that embeds reconstructed raw image data together with metadata derived directly from a source raw capture. Rather than attempting to synthesize a new camera model for the transformed image, KYDNG reuses metadata that remains valid under PARSEK's controlled capture assumptions. The following sections describe the role of metadata in raw image interpretation, the design and implementation of the KYDNG pipeline, and the qualitative and quantitative basis for evaluating whether the resulting files preserve rendering consistency.

Metadata and the Interpretation of Raw Images

Raw image formats encode substantially more than sensor values. They also carry the metadata required to convert those values into a perceptually meaningful image. This metadata includes, among other elements, color calibration matrices, white-balance information, black and white levels, and the spatial organization of the sensor's color filter array. Raw processors rely on these parameters to determine how pixel values should be mapped into display-referred color and tone. Consequently, the visible appearance of a raw image depends not only on the recorded measurements, but also on the metadata used to interpret them.

In workflows such as PARSEK, the status of that metadata is more favorable than might initially be assumed. PARSEK operates on multiple captures of the same scene acquired under controlled sub-pixel shifts and nominally identical camera settings. Because the source images are produced by the same sensor under the same exposure and camera configuration, much of the metadata relevant to rendering remains stable across the input set. Metadata governing camera color calibration, white balance, and tonal scaling does not become irrelevant merely because the images are aligned and combined into a super-resolution reconstruction. On the contrary, those parameters continue to describe how the underlying sensor-domain measurements should be interpreted.

This observation motivates a metadata-reuse strategy rather than a metadata-reconstruction strategy. For PARSEK outputs, the objective is not to infer a new rendering model from scratch, but to preserve and correctly reapply the rendering model already associated with the source capture. The difficulty lies in expressing that model in a form that downstream software can understand, particularly when the reconstructed image differs from the original files in resolution, organization, or storage structure.

Metadata Categories

A useful way to understand the KYDNG approach is to separate the reused metadata into functional categories. Not all metadata serves the same purpose in a raw workflow. Some fields are required to make the file structurally interpretable as a raw image, while others primarily determine how that image will be rendered in terms of color and tone. Treating all metadata as a single undifferentiated set obscures the fact that different fields contribute to different stages of downstream interpretation. In the present work, the most important categories are structural metadata, color interpretation metadata, tonal metadata, and preview or contextual metadata. These categories align with the practical problem identified in the current PARSEK workflow: the reconstructed image data may remain numerically valid, yet the rendered result can still appear incorrect if any of these interpretive layers is missing or inconsistent.

Structural metadata defines how the raw processor should parse the image as a raw capture. In KYDNG, this includes fields such as the CFA pattern, CFA repeat dimensions, image geometry, sample organization, and other tags required to represent the output as a single-channel CFA-style image within a DNG container. These fields are essential because the reconstructed PARSEK output must be packaged in a form that existing raw processors recognize as belonging to their normal raw pipeline rather than as an already-rendered RGB image. In the current implementation, this structural role is reflected directly in the use of `PHOTOMETRIC_CFA`, `SAMPLESPERPIXEL = 1`, and explicit tag definitions for `CFARepeatPatternDim` and `CFAPattern`. Without these fields, downstream software may fail to interpret the file as raw at all, regardless of whether the numerical image data is otherwise usable.

Color interpretation metadata governs how camera-domain values are mapped into a standardized color space. In the present work, this category includes fields such as `ColorMatrix1`, the calibration illuminant, and white-balance-related values represented through `AsShotNeutral`. These fields are especially important because the principal visual failures observed in metadata-free PARSEK outputs are not necessarily caused by corruption of reconstructed raw-domain values, but by loss of the camera-specific interpretation needed to render those values correctly. Since PARSEK combines multiple captures taken under controlled conditions using the same camera and nominally identical settings, the color interpretation metadata from a reference input image remains a reasonable description of how the reconstructed sensor-domain data should be rendered. KYDNG therefore reuses these fields rather than attempting to synthesize a new camera profile for the transformed image.

Tonal metadata determines how raw values are scaled and normalized before rendering. In this work, the most significant fields in that category are black level and white level. The black level defines the baseline sensor response that should be treated as zero-referenced signal, while the white level defines the effective saturation point. These parameters play a direct role in brightness scaling, highlight handling, and dynamic-range interpretation.

The final category consists of preview and contextual metadata. This includes descriptive EXIF fields such as camera make, model, lens information, exposure settings, and embedded preview imagery. Although these fields do not define the raw sample array itself, they remain important for two reasons. First, they improve compatibility with raw-processing software and preserve the contextual identity of the source capture. Second, the embedded preview can serve as a practical rendering reference. Earlywine and Dietz [3] showed that preview imagery is relevant because it may embody a visually meaningful rendering of the original raw capture, and that replacing or improving the preview could materially improve downstream interpretation, especially when a raw developer uses that preview as a cue for tonal or color matching. In that sense, the preview image is not merely cosmetic; it is part of the broader metadata ecosystem that can influence how corrected output is presented or approximated.

Viewed together, these metadata categories clarify the design logic of KYDNG. Structural metadata allows the output to be recognized as raw, color metadata preserves camera-dependent chromatic interpretation, tonal metadata preserves sensor-dependent scaling behavior, and preview or contextual metadata helps maintain practical compatibility and rendered consistency. This categorization also makes clear why metadata reuse is more appropriate than metadata regeneration for the present application.

Metadata-Guided DNG Construction

KYDNG addresses this problem by constructing a DNG representation of the PARSEK output while propagating metadata from a reference input image. The implementation uses metadata extraction from the source capture together with explicit writing of DNG structural and rendering tags. In the current implementation, metadata is obtained from the first input image using **LibRaw** [4] and **Exiv2** [5], while image data and DNG-specific tags are written through **libTIFF** [6]. The output image itself is stored as a single-channel CFA-style raw image, allowing the reconstructed file to remain compatible with conventional raw-processing expectations.

The software design separates metadata access from container construction because no single library used in the present workflow provides equally strong support for raw-domain calibration data, descriptive EXIF metadata, and low-level DNG writing. **LibRaw** is an active super-set of **dcraw** [7] and is used primarily for access to raw-specific camera information that is difficult to recover reliably from general-purpose metadata interfaces alone. In particular, it provides access to sensor-interpretation parameters and camera color information needed to populate rendering-related DNG fields such as the color matrix. **Exiv2** is used for complementary metadata extraction. Whereas LibRaw is well suited to raw parsing and sensor-level metadata access, Exiv2 provides convenient access to higher-level EXIF fields such as camera make and model and other descriptive information associated with the source image. These fields help preserve the contextual identity of the original capture and improve compatibility with downstream raw-processing software. The DNG itself is constructed using libTIFF, which was selected because DNG is fundamentally TIFF-based and the implementation requires direct control over tag creation, image directories, and raw-specific storage structure. The present method creates a new DNG representation of the reconstructed PARSEK output rather than modifying an existing file in place. As a result, the code must explicitly define image geometry, storage format, photometric interpretation, CFA structure, and related fields. libTIFF provides the level of control needed to write the main image as a single-channel CFA-style raw image, register DNG-specific tags such as CFARepeatPatternDim and CFAPattern, and create an additional image directory for an embedded preview image.

The pipeline begins by selecting a reference input image and extracting camera-relevant metadata from it. This includes the camera make and model, tonal limits, white-balance-related parameters, and camera color calibration information. In addition, the implementation records the camera make, unique camera model, DNG version identifiers, and white level, and it assigns a calibration illuminant for the color transformation. A JPEG preview is embedded as a secondary image directory so that downstream software can use an immediately renderable reference image when appropriate.

A central design choice in KYDNG is the reuse of source-derived rendering metadata rather than the synthesis of new metadata from the reconstructed image. The camera color transform is carried through the DNG using fields such as ColorMatrix1, while white balance is represented through AsShotNeutral. Tonal interpretation is constrained using black and white level metadata, which establish the baseline sensor response and the saturation point. These values are not arbitrary annotations; they directly affect how raw developers scale, normalize, and render the data. Reusing source-derived values therefore provides a principled way to keep the reconstructed image in the same interpretive regime as the original capture. The structural metadata also matters. Even though the PARSEK reconstruction yields a higher-resolution image, KYDNG writes the output in a CFA-consistent raw representation rather than as a conventional demosaiced RGB image. Encoding the CFA pattern and repeat dimensions allows downstream software to treat the file as a raw capture within its expected processing model. Additional EXIF metadata can then be copied into the packaged file using post-processing tools such as **ExifTool** [8] to improve completeness and preserve contextual descriptors.

The embedded preview image is also important in light of the work done by Earlywine and Dietz [3]. That earlier study showed that preview imagery can serve as a practical rendering reference because it already reflects a corrected visual interpretation of the original capture, even if it may also include compression, resizing, or manufacturer-specific corrections. KYDNG adopts that insight by embedding a JPEG preview during DNG construction so that the output contains not only reconstructed raw-domain data, but also a readily renderable representation that can assist with compatibility and practical interpretation in downstream software.

The result is a DNG that is not merely syntactically valid, but semantically closer to a native raw capture as seen in figure 2. In effect, KYDNG repackages the output of computational raw processing in a form that standard raw tools can interpret without requiring custom rendering logic.

Figure 2: *Comparison of Original .ARW and PARSEK + KYDNG Output*

Evaluation of Rendering Consistency

The effectiveness of KYDNG is best evaluated in the rendered domain. The objective of the method is not to alter the reconstructed image content, but to ensure that downstream software interprets that content correctly. Accordingly, evaluation focuses on whether a metadata-guided DNG package produces rendered output that is perceptually consistent with the original raw capture. A useful comparison involves three versions of the same scene: (1) the original raw input image, (2) the PARSEK output saved without metadata, for example as a 16-bit PNG, and (3) the KYDNG-generated DNG containing reconstructed metadata.

When the metadata-free output is opened in standard image viewers or raw-processing environments, noticeable shifts in white balance, color rendering, and tonal response are typically observed. These deviations do not primarily arise from errors in the reconstructed pixel values; rather, they arise because the output is no longer accompanied by the metadata that standard tools require for correct interpretation.

When the KYDNG output is opened in raw processing software such as RawTherapee, the behavior differs significantly. The file is recognized as a valid raw image, and the software applies its existing camera-specific processing pipeline. As a result, the rendered image more closely matches the appearance of the original raw input, with improved color fidelity and consistent tonal mapping. Qualitative comparisons shown in figure 2 illustrate that the KYDNG output avoids the color shifts and contrast inconsistencies observed in the metadata-free representation.

Table 1: KYDNG Metrics

Comparison	SSIM	PSNR (dB)	Mean Delta E	Median Delta E
Original vs. PARSEK	0.3353	7.04	48.05	58.50
Original vs. PARSEK + KYDNG	0.9177	29.32	3.23	2.26

In addition to qualitative comparison, quantitative evaluation was performed in table 1 on rendered outputs of the original ARW image, the metadata-free PARSEK output, and the PARSEK + KYDNG output .

To quantify rendering consistency, three complementary metrics are used. Structural Similarity Index Measure (SSIM) evaluates perceptual similarity between two images on a scale from 0 to 1, where 1 indicates a perfect match and values below 0.5 represent substantial structural deviation. Peak Signal-to-Noise Ratio (PSNR), expressed in decibels, measures pixel-level fidelity; values above 20 dB are generally considered acceptable, while values above 30 dB indicate high fidelity. CIELAB Delta E quantifies perceptual color difference in a way that correlates with human vision; a Delta E below 1 is imperceptible to most observers, values between 1 and 5 are considered acceptable in many imaging applications, and values above 10 represent clearly visible color errors. Applied together, these metrics capture both structural integrity and color accuracy in the rendered output.

Using the original rendering as the reference, the metadata-free PARSEK result produced an SSIM of 0.3353, a PSNR of 7.04 dB, and a mean CIELAB Delta E of 48.05, indicating substantial structural and perceptual deviation. In contrast, the PARSEK + KYDNG result achieved an SSIM of 0.9177, a PSNR of 29.32 dB, and a mean Delta E of 3.23. These results indicate that metadata-guided DNG packaging substantially improves rendering consistency relative to the original source image and supports the claim that the major visual errors in the metadata-free PARSEK output arise from loss of rendering metadata rather than corruption of the reconstructed image content.

Together, these qualitative and quantitative evaluations demonstrate that the primary source of rendering discrepancies in PARSEK outputs is not the transformation of pixel data, but the loss or misinterpretation of metadata. By reapplying camera-consistent metadata within a DNG container, KYDNG enables downstream tools to correctly interpret the reconstructed image without requiring custom adjustments or calibration.

Limitations

The current method depends on a specific set of assumptions. Most importantly, it assumes that the source images are captured under identical or near-identical conditions, so that metadata extracted from a reference input remains valid for the reconstructed output. This assumption is well matched to PARSEK's intended acquisition model, but it may not hold for workflows involving changing illumination, mixed exposures, multiple sensors, or more heterogeneous capture conditions.

A second limitation is representational rather than algorithmic. Not all manufacturers expose their complete rendering model through metadata that is standardized, documented, or expressible within the DNG specification. Camera vendors such as Nikon and Canon often rely on proprietary metadata fields, private MakerNote content, or undocumented processing conventions that influence how raw files are rendered by manufacturer-specific software. Although standard fields such as color matrices, white balance, and black and white levels can often be transferred successfully, that transfer does not guarantee that the full manufacturer rendering intent can be reproduced in DNG form. In such cases, some of the original camera-specific behavior may remain unavailable to generic raw processors even when the core DNG metadata is correct.

Thus, KYDNG should not be interpreted as a universal reconstruction of every aspect of every proprietary raw pipeline. It is better understood as a practical metadata-guided packaging strategy that preserves the subset of rendering information that is both available from the source capture and representable within a standard raw container. Its success therefore depends in part on how much of the original camera's rendering behavior is captured by open metadata fields rather than proprietary logic.

Conclusion

The results of this work demonstrate that the primary challenge in producing visually consistent outputs from PARSEK is not the transformation of pixel data, but the preservation and correct interpretation of metadata. PARSEK demonstrates that raw-domain image content can survive substantial computational transformation while still requiring camera-consistent metadata to be rendered correctly by downstream software. Without that metadata, even high-quality reconstructed outputs may appear visually incorrect when opened in standard tools.

KYDNG addresses this problem by repackaging a reconstructed PARSEK output into a DNG container while propagating source-derived metadata that remains valid under the capture assumptions of the workflow. By explicitly encoding CFA structure, color calibration, white balance, tonal limits, and a preview rendering, the method enables standard raw processors to interpret the reconstructed image using a camera-consistent model. The resulting behavior suggests that metadata reuse, when capture conditions justify it, is a practical and effective alternative to attempting full metadata re-derivation.

More broadly, the method suggests that metadata-preserving repackaging may be valuable for other computational imaging systems that operate in the raw domain. The present implementation is not universal and remains bound by the limits of standardized metadata expressibility, especially in the presence of proprietary vendor-specific processing. Even so, `KYDNG` provides a concrete demonstration that raw-domain reconstruction and metadata-aware packaging can be integrated into a workflow that preserves perceptual consistency while remaining compatible with existing raw-development software.

References

[1] H. G. Dietz, "Leveraging Pixel Value Certainty in Pixel-Shift and Other Multi-Shot Super-Resolution Processing," *Electronic Imaging,* vol. 36, no. 15, 2024.

[2] Adobe, "The public archival format for digital camera raw data," Adobe, 10 March 2026. [Online]. Available: https://helpx.adobe.com/camera-raw/digital-negative.html. [Accessed 18 March 2026].

[3] C. P. Earlywine and H. G. Dietz, "Preservation of Color Metadata Across Transformations of Pixel Data," *Electronic Imaging,* vol. 37, 2025.

[4] I. Borg and A. Tutubalin, "About LibRaw," LibRaw LLC, [Online]. Available: https://www.libraw.org/about.

[5] R. Mills, "C++ metadata library and tools," [Online]. Available: https://exiv2.org/index.html. [Accessed 18 March 2026].

[6] LibTIFF Project, "TIFF Library and Utilities," LibTIFF Documentation, 2026. [Online]. Available: https://libtiff.gitlab.io/libtiff/. [Accessed 18 March 2026].

[7] D. Coffin, "Decoding raw digital photos in Linux," [Online]. Available: https://www.dechifro.org/dcraw/. [Accessed 18 March 2026].

[8] P. Harvey, "ExifTool," [Online]. Available: https://exiftool.org/. [Accessed 18 March 2026].