ISO/IEC JTC 1/SC 29/WG 1
(& ITU-T SG16)

Coding of Still Pictures

JBIG	**JPEG**
Joint Bi-level Image Experts Group	Joint Photographic Experts Group

TITLE: JPEG XL Use Cases and Requirements

SOURCE: WG1

PROJECT: JPEG XL

EDITOR: Jan Wassenberg and Jon Sneyers

STATUS: Approved

REQUESTED ACTION: For dissemination

DISTRIBUTION: Public

Contact:
ISO/IEC JTC 1/SC 29/WG 1 Convener – Prof. Touradj Ebrahimi
EPFL/STI/IEL/GR-EB, Station 11, CH-1015 Lausanne, Switzerland
Tel: +41 21 693 2606, Fax: +41 21 693 7600, E-mail: Touradj.Ebrahimi@epfl.ch

JPEG XL Use Cases and Requirements

1 Background

The JPEG Committee has launched the Next-Generation Image Coding activity, also referred to as JPEG XL. This activity aims to develop a standard for image coding that offers substantially better compression efficiency than existing image formats (e.g. >60% over JPEG), along with features desirable for web distribution and efficient compression of high-quality images.

The JPEG XL Final Call for Proposals (CfP) for a Next-Generation Image Coding Standard has been issued at the 79th JPEG meeting, La Jolla, USA, 9-15 April 2018 and the proposals evaluation results have been made available to the 81st JPEG meeting, Vancouver, Canada, 15-19 October 2018.

2 Introduction

The need for efficient image compression is self-evident, when taking into account that billions of images are captured, created, uploaded, and shared daily. Applications are becoming increasingly image-rich, and websites and user interfaces (UIs) rely on images for sharing experiences and stories, visual information and appealing design.

On the low end of the spectrum, UIs can target devices with stringent constraints on network connection and/or power consumption. Even though network download speeds are improving globally, in many situations bandwidth is constrained to speeds that inhibit responsiveness in applications. On the high end, UIs utilize images that have larger resolutions, higher dynamic range and wider color gamut, as well as higher bit depths, which leads to a further explosion of image data.

For most applications, JPEG, PNG and WebP are still used as the primary coding formats. More efficient compression will benefit the described applications, and will lead to reduced network transmission times and more interactive applications.

When compared to video data, images can be stored with relatively few bits. Still, websites and UIs can contain hundreds of images, or several high-resolution images, leading to several megabytes worth of data – which could be equivalent to more than a minute of video. While video streams can be buffered before playback, image-based UIs have to be responsive and interactive, without several seconds of loading and stalling when downloading or scrolling.

Newer image formats with more efficient compression performance than JPEG have been developed over the last decades, but these formats have various shortcomings with respect to the use cases detailed below.

Recently, evidence has been presented of compression technologies that outperform image coding standards in common use. For example, in the conclusions of the Grand Challenge comparisons held at the Picture Coding Symposium (PCS 2015) [1] and the IEEE Conference on Image Processing (ICIP 2016) [2], it was reported that "there is evidence that significant improvements in compression efficiency can be obtained using latest state of the art in lossy and lossless cases". Several metrics showed the HEVC HM encoder with SCC extensions [3] to be superior according to most metrics, and for most test images. Subjectively, Daala [4] was competitive, with a limited difference in MOS scores between HEVC and Daala. Despite these technical advances, no widespread standard is available that has state-of-the-art compression performance, and is widely supported in consumer devices and browsers.

3 Scope

This new JPEG activity aims to develop a new image coding standard that provides state-of-the-art image compression performance. The JPEG XL format will allow current and future applications to realize several benefits compared with existing codecs:

- Significant compression efficiency improvement over coding standards in common use at equivalent subjective quality, e.g. >60% over JPEG.
- Features for web applications, such as support for alpha channel coding and animated image sequences.
- Support of high-quality image compression, including higher resolution, higher bit depth, higher dynamic range, very high quality coding and wider color gamut coding.

To encourage widespread adoption, an important goal for this standard is to support a royalty-free baseline.

4 Use Cases

This section presents a list of use cases that motivate the need for a new image coding standard.

4.1 Image-rich UIs and web pages on bandwidth-constrained connections

Web sites and user interfaces become more and more image-driven. Images play a major role in the interaction between users, the selection of topics, stories, movies, articles and so on. In these UIs, formats are preferred that are widely supported in browsers and/or CE devices, such as JPEG, PNG and WebP.

4.1.1 Social media applications

Billions of user-generated images are captured and uploaded daily. After uploading, the images are typically converted into multiple quality versions and formats and stored on content delivery network (CDN) servers. More efficient image compression will aid to distribute social media images to users worldwide, including to locations with limited connectivity or low-bandwidth mobile connections. Image formats need to be supported that are widely supported on consumer devices, such as smartphones and tablets, and on browsers. Compression efficiency is key in delivering the images to devices over low-bandwidth connections, and in making the UIs and web sites as responsive as possible. Additionally, it is desirable to have a bitstream that allows the option of decoding progressively, which allows useful previews to be shown while the images are still loading. It is also important to allow high-resolution photos to be decoded to lower-resolution versions sufficient for displaying at common screen resolutions, without sending or allocating memory for the entire high-resolution version.

Some images become “viral” in the sense that they are widely shared across different social media. They are also often downloaded, modified (e.g. adding a text overlay) and shared again, for example when “internet memes” are being created. Since in these cases the source images are usually not lossless originals, and social media applications typically apply relatively low bitrate recompression to the images uploaded to their platform, the issue of generation loss (accumulation of compression artifacts) is particularly relevant.

4.1.2 Screen readers/clicking on image text

Examples of text in images include event flyers, company brochures and documents in general. For these, it would be desirable to store the actual text into optional metadata, such that users can interact with it (click on links, copy text).

4.1.3 Media distribution applications

In many media distribution applications, UIs and web sites contain a wide array of artwork images that guide users through the catalog. Images are typically derived from high-quality studio shots, artwork, or movie/show masters. Derived images can include natural and synthetic images, transparent overlays, multilingual text, animation, gradients etc. Multiple quality/resolution versions of the same image are finally encoded, and stored in the CDN. The UIs can contain hundreds of images, ranging from small thumbnail-like images to screen-spanning billboard images. In order to reduce the number of versions that are needed of the same image (for responsive web design), it would be useful if the bitstream can be decoded to different-resolution versions of the same image.

Some e-commerce applications require very high quality. The quality of catalog images has been observed to correlate with sales, and in some cases there are legal requirements for 'what you see is what you get'. Users expect the image format to be color-aware, e.g. by honoring ICC metadata or a similar equivalent representation of color space and transfer functions, because ICCv2 has difficulty expressing HDR transfer functions. The format should also enable coding of images with some quality guarantees, such that webmasters do not need to manually verify the quality of each encoded image (infeasible for large collections).

The large number of images invites parallel decoding. It is desirable for concurrent decoders to be efficient on multicore systems.

4.1.4 Cloud storage applications

Cloud storage applications amass a huge amount of images captured by users. After uploading, these images are stored on servers either as a copy, or after a lossless [5] or lossy transcoding operation. For browsing and timeline-style thumbnail generation, lossy transcoding can be performed to more efficient formats, lower resolutions, and preview images. Both for storage and browsing, more efficient formats are desirable. For storage (in particular for archival), efficient coding at very high quality (visually lossless, as defined below) is desirable. Because of the very high storage volume and ongoing storage cost in such applications, efficient means "economically viable" (mathematically lossless is too expensive). Further to "high quality", archiving entails spatial resolution, bit depth, gamut and compression artifacts that will still be acceptable after 5 years, otherwise users would choose mathematically lossless representations.

4.1.5 Media web sites

Images are captured by news agencies, journalists and users, and are selected for publication on media web sites. Images can range from high resolution to thumbnail-size, resulting in web pages that contain dozens of megabytes worth of images.

To facilitate the adoption of a new format, it is important to provide a benefit for webmasters with large existing collections of JPEG images. Clients will take some time to adopt JPEG XL, so servers will need to provide backwards-compatible JPEG options for quite some time. If implemented by transcoding from JPEG XL to JPEG, this risks generational loss. If storing both JPEG and JPEG XL, the webmaster's storage requirements actually increase. Instead, it is highly desirable to support a lossless transcoding of existing JPEG bitstreams (without going back to the pixel domain). To avoid disrupting existing caching, version control, and hashing/checksums, the decoded bitstream should offer the option of bit-exact reconstructions of the original JPEG.

Because existing images are often stored as medium-quality JPEGs and the originals are no longer available, it is desirable to provide an enhancement step to improve the visual quality of the decoded image. Encoders will want to rely on the fact that decoders will (or will not) carry out this enhancement.

During a transition period where some servers and clients already support JPEG XL, it is also desirable for new encoders to produce XL bitstreams that can be reversibly transcoded to JPEG bitstreams. Furthermore, it is desirable for the bitstream to contain additional information that allows improved visual quality of decoded images.

The preceding two use cases provide an incremental growth path to JPEG XL, with several intermediate benefits:

- Users can update to XL software for decoding JPEG images, which improves quality;
- Updating both the encoder and decoder can further improve quality via enhancements informed by additional information

4.1.6 Animated image applications

Animated image sequences have become very popular, e.g. for increasing interactivity and expressing emotions,. The wide majority of animated image sequences currently rely on the GIF image format, which suffers from inefficient compression and a limited color palette. These animations are typically short and played in a loop, and often fit into memory. Examples of existing animations can be found in the usage of existing aPNG, aWebP and GIF files. Animations should also support transparency for use as overlays.

Seeking can be useful in browsers when users switch away from a tab. Upon returning to the tab, users expect the animation to resume where it was.

4.1.6 Mobile applications and games

The download and install size of mobile applications and games is an important factor when users decide whether or not to download or remove the app or game. Images typically constitute a large proportion of the total size. Efficient compression helps to reduce the size. Decoding has to be fast enough to allow short loading times. In terms of image content, these images are often non-photographic in nature, and alpha transparency is often required. Some assets such as UI overlays may also require lossless coding in order to avoid unacceptable compression artifacts.

4.2 High-quality imaging applications

On the high end, UIs utilize images that have larger resolutions and higher bit depths, and the availability of higher dynamic range and wider color gamut is a benefit for vivid color imagery. 4K TVs are becoming mainstream, and HDR/WCG technology is picking up, leading to a shift to high-quality UIs.

Although these higher-end applications typically target more stable network connections, transmission of multiple high-quality images still takes a significant time on most current network connections. The storage cost of large image collections (such as those of hobby photographers) is also an important consideration. A new standard should provide efficient compression and high visual quality for these applications.

Images in these applications can contain a mixture of natural images and synthetic elements (overlays, multilingual text, gradients etc.). A new standard should include coding tools that can efficiently compress synthetic content while avoiding visible quality artifacts (e.g. aliasing, banding).

4.2.1 Rapid photo viewing

Users often collect albums of hundreds of images, e.g. vacation snapshots. These photos are often large, e.g. 20-30 Megapixels. Users expect fast response times and faithful reproduction when browsing through them on tablets at full screen resolution (currently around 2 Megapixels).

4.2.2 HDR/WCG user interfaces

In many applications, such as on-demand video services and gaming, HDR/WCG images are necessary to support HDR/WCG video or to increase user experience. For example, users may want to store stills (single frames) from movies. Current popular image formats do not allow for representation of HDR/WCG content. A new HDR/WCG image coding standard is needed to efficiently cope with such applications. The format should allow content creators to specify the rendering intent (as defined by ICC).

4.2.3 Augmented/virtual reality

Applications such as augmented reality, virtual reality, and 360-degree images require high-resolution images that need to be efficiently compressed. For these high-resolution images, region-of-interest coding is a desirable feature to support interactive applications.

These applications typically require additional metadata. Users want to transfer such images to various storage media and it is desirable to avoid separate sidecar files. Thus, the bitstream should support such well-defined and non-opaque metadata.

4.2.4 Fast training of machine learning models

Training models such as image classifiers requires large numbers of training images. These need to be stored at high quality (or even lossless) to avoid compression artifacts that can interfere with training. As input pipelines may be a bottleneck, they are often parallelized, so concurrent (multithreaded) decoding of many images must be fast despite sharing limited memory bandwidth (a common bottleneck on multi-CPU systems).

4.2.5 Image bursts

Cameras increasingly store bursts of images in order to decrease the noise power, increase the apparent resolution and/or capture more light. Users expect these to be stored in a single container to simplify copying them and associate them with metadata. The size of these images may exceed available RAM on mobile devices, so it must be possible to stream them to nonvolatile storage and conversely process them without loading all parts into memory. Users also wish to record longer bursts, such that the write speed is important. In particular, a hardware implementation of the encoder should reach throughputs corresponding to expected flash write speeds, otherwise users may be tempted to write raw pixels.

4.2.6 High-end photography

Existing (medium-format) cameras generate 100-400 MPixel images. Users wish to open these for viewing or processing (at full resolution) without unreasonable delays.

Users sometimes also wish to store raw (color-filter array output) images in a lossless or near-lossless encoding. This allows the images to be 'cooked' later with possibly more advanced processing algorithms.

4.2.7 Image mosaics

Advanced users generate gigapixel-scale images - for example, panoramas or image mosaics of static scenes such as fine art or landscapes. For these to be useful, users will need to pan and zoom within such images at interactive speeds.

4.2.8 Depth images

Recent phone cameras have multiple image sensors and often compute depth maps from the images. These are useful for viewpoint synthesis and other computational photography applications. Users expect that depth maps and related metadata should be stored in the same bitstream as the main photo.

4.2.9 Decoding of untrusted sources

Malicious codestreams have been known to have harmful effects. Such malicious codestreams often originate from untrusted sources. The bitstream should be designed to avoid the potential for harm from such sources.

4.2.10 Printing

In the printing industry, the CMYK color model is widely used. Additionally, extra color channels can be used for spot colors. For distribution of images intended for printing, high-resolution and lossless or very high quality is desired, as well as the ability to represent CMYK and possibly additional channels (for spot colors/OGV). CMYK will typically be stored as lossless or near-lossless because it is used for interchange instead of storage and delivery.

5 Requirements

This section presents the requirements that should be met by the standard so as to be suitable for the above described use cases. Requirements are split between “core requirements” which are essential and “desirable requirements” which are nice to have and will be decided depending on their cost.

5.1 Uncompressed image attributes

This standard targets image coding technology that can at least support images with the following attributes:

- Image resolution: from thumbnail-size images up to very large images.

- Bit depth: at least 8-bit, 10-bit, 12-bit and 16-bit.
- Channels: any number from 1 to 4096
- Metadata required for reconstructing input images:
    - color primaries
    - white points
    - transfer functions, including those listed in BT. 1886 [10], sRGB and BT. 2100 [11]
- Chrominance subsampling (where applicable): 4:0:0, 4:2:0/4:2:2 (for lossless transcoding of existing JPEG), and 4:4:4.
- Different types of content, including:
    - natural (photographs, aerial/satellite, document scans)
    - synthetic (rendered/screen content)
    - illustrations/logos/UI elements/comics.
    - game graphics/textures (for 3D models)

5.2 Compressed bitstream requirements

The standard shall cover at least the core requirements, and is encouraged to cover desirable requirements as well.

Core requirements
Significant compression efficiency improvement over commonly used coding standards and solutions at equivalent subjective quality, and superior subjective quality at equivalent bitrates, across a wide range of perceptual qualities in common use.
Hardware and software implementation-friendly encoding and decoding (including memory and power consumption)
Fast encoding configuration with input throughput comparable to the peak write throughput of commonly used flash media.
Fast full-resolution decoding configuration: as a target, at most twice the typical JPEG decoding time (single thread).
Scalability on multicore systems: high parallel efficiency when decoding single and multiple images in parallel.
Decoding full-resolution without requiring the entire image to be held in memory (for interactive viewing of large images)
Support for alpha channel / transparency coding.
Support for lossless coding: of all channels, and of alpha channel even if other channels are stored with lossy coding.

Support for animation image sequences and photo bursts.
Support for 16-bit integer and 16-bit float inputs, and outputs with corresponding precision.
Support for high dynamic range coding.
Support for wide color gamut coding.
Support for efficient coding of images with different types of content.
Support for "visually lossless" coding, without requiring mathematically lossless coding. NOTE: "visually lossless" is as defined in AIC-2 (ISO/IEC 29170-2).
Support for progressive coding, in terms of quality, spatial resolution, and scanning order.
Support for lossless transcoding between JPEG bitstreams and a subset of JPEG XL bitstreams, with exact reconstruction or with optional removal of some metadata.
Support enhanced decoded image quality from lossless-transcoded JPEG.
Support for signaling whether the enhanced decoding for lossless-transcoded JPEG bitstreams shall be carried out.
Support for higher-quality decoding based on extra information (generated from the original pixels) that can be included in lossless-transcoded JPEG bitstreams.
Support for all accompanying side information useful for interpreting an image, such as color encoding, HDR intensity target, stereo-related metadata.
Desirable requirements
Support for embedded preview images
Support for very low file size image coding (e.g. <200 bytes for 64×64 pixel images)
Support for coding designed for consumption by machine learning or non-human systems.
Support for a low-complexity profile - reasonable encode/decode time even on limited mobile hardware
Support for interactive panning/zooming in high-resolution images that do not fit in working memory.
Support for efficient coding of non-photographic content, including up to 12-bit palette indices.
Support for additional channels (e.g. depth image, spot colors), for which the decoder understands their interpretation (e.g. whether it can be ignored).
Minimal generation loss when lossy compression is applied multiple times.
Avoid potentially insecure features such as remote code execution and unbounded resource consumption.
Support for text embedded in the format for interacting with links and copy/pasting.

6 Royalty-free goal

The royalty-free patent licensing commitments made by contributors to previous standards, e.g. JPEG 2000 Part 1, have arguably been instrumental to their success. JPEG expects that similar commitments would be helpful for the adoption of a next-generation image coding standard.