

The FFV1 Video Codec in Practice: Opportunities and Challenges

Reto Kromer • AV Preservation by reto.ch

**Digital Preservation:
Digital Formats, Network and Data Curation**

Università degli Studi della Tuscia

Università degli Studi di Udine

19 July 2024

1

Summary

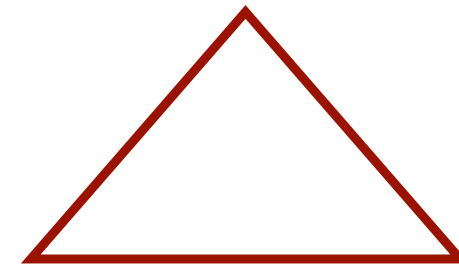
- The Dilemma
- Today's Solutions
- Future Directions

2

The Dilemma

3

image quality



encoding time

file size

4

Compression

- uncompressed
- lossless compression
- lossy compression
- chroma subsampling
- born compressed

5

Uncompressed

- + data processing is simpler
 - + less computing power is needed
 - more storage is need
 - slower writing, transmission and reading
- examples: TIFF, DPX, DNG, OpenEXR

6

Lossless Compression

- + less storage is need
 - + faster writing, transmission and reading
 - data processing is more complex
 - more computing power is needed
- examples: JPEG 2000, FFV1

7

Lossy Compression

- optimised for postproduction
examples: ProRes 422, ProRes 4444;
DNxHD, DNxHR
- optimised for access
examples: H.264 (AVC), H.265 (HEVC),
H.266 (VVC); AV1

8

Chroma Subsampling

from analogue television to digital television

4:4:4 sampling

- 4:2:2 subsampling for postproduction
- 4:2:0 and 4:1:1 subsamplings for access

9

Born Compressed

- optimised for both image acquisition and postproduction
- examples: CineForm RAW, ProRes RAW, Blackmagic RAW

10

Today's Solutions

11

JPEG 2000

- ...

12

FFV1

- lossless video compression
- frames divided in slices, with checksums
- open source and patent free
- adopted by several archives
 - Claudio Santancini

13

Archive Master and Mezzanine

single images ("film")

- Matroska, FFV1, 2K, RGB, 16 bit
- Matroska, FFV1, 2K, R'G'B', 12 bit

streams ("video")

- Matroska, FFV1, "HD", Y'CbCr 4:2:2, 10 bit
- Matroska, FFV1, "HD", Y'CbCr 4:4:4, 12 bit

audio

- Matroska, FLAC, 96 kHz, 24 bit
- Matroska, FLAC, 192 kHz, 24 bit

14

	avantages	disavantages
TIFF DPX OpenEXR	data easier to process	bigger files
JPEG 2000 FFV1	smaller files	data complexer to process

15

A Bridge

RAWcooked (command-line interface)

→ mediaarea.net/RAWcooked

to encode:

```
rawcooked input_path_of_folder
```

to decode:

```
rawcooked rawcooked_input_file.mkv
```

to get help:

```
rawcooked -h
```

```
man rawcooked
```

16

RAWcooked

- encoding into Matroska (.mkv) using FFV1 video codec and FLAC audio codec
- all metadata preserved
- decoding with bit-by-bit reversibility
- possibility to embed sidecar files such as MD5, LUT, XML and PDF
- compatibility with media players

17

Future Directions

18

Sustainability

from 2004 to 2012

- all power consumption from the grid

from 2013 to 2019

- no electricity from the grid is used for the IT

from 2020 to 2024

- no electricity at all comes from the grid

19

No Time to Wait

- 2016 in Berlin
- 2017 in Vienna
- 2018 in London
- 2019 in Budapest
- 2021 online edition
- 2022 in The Hague
- 2023 in Prague
- 2024 in Karlsruhe

20

Reading (1)

Reto Kromer: **Matroska and FFV1: One File Format for Film and Video Archiving?**, in «Journal of Film Preservation», n. 96 (April 2017), FIAF, Brussels, Belgium, p. 41–45

→ retokromer.ch/publications/JFP_96.html

21

Ways to store Bayer-type data

- pixel values generated by one de-mosaicking algorithm (digital blow-up)
- pixel values generated by mixing two green sensel values into one (digital reduction)
- raw sensel values

23

Reading (2)

Reto Kromer: **On the Bright Side of Data Migrations**, in «IASA Journal», n. 49 (December 2018), IASA, p. 18–22

→ retokromer.ch/publications/IASA_49.html

22

Reading (3)

Reto Kromer: **From the Sensor to the File: Different Ways to Work with Bayer-Type Data**, coming soonish

24

Improvements (1)

support of any type of channel, including:

- Bayer-type data
- multispectral imaging
- $Y'CoCg$ colour model

25

Improvements (2)

- support of HDR
- support of 1D and 3D LUTs
- revision of the bit stream
- tuning of the compression algorithm (speed and rate)

26

The screenshot shows a GitHub pull request interface for the repository 'MediaArea / RAWcooked'. The pull request is titled '[WIP] Error Correction Code #235' and is being merged by JeromeMartinez. The interface includes navigation tabs for Code, Issues (45), Pull requests (3), Actions, Projects, and Security. Below the title, there are statistics for Conversation (1), Commits (1), Checks (0), and Files changed (26). A comment from JeromeMartinez, dated Jun 20, 2019, describes the 'Error Correction Code' feature, explaining that it allows for file correction if slightly damaged, avoiding complete retransmission. The comment lists the principle of the feature with the following details:

- a (file) shard is 1 MiB long
- a shard is considered damaged if 1 to 1 MiB bytes are wrong
- every 248 data shards (so 248 MiB), 8 parity shards are encoded, as well as their

27

AV Preservation by
reto.ch

Sandrainstrasse 3
3007 Bern
Switzerland

reto.ch
info@reto.ch



28