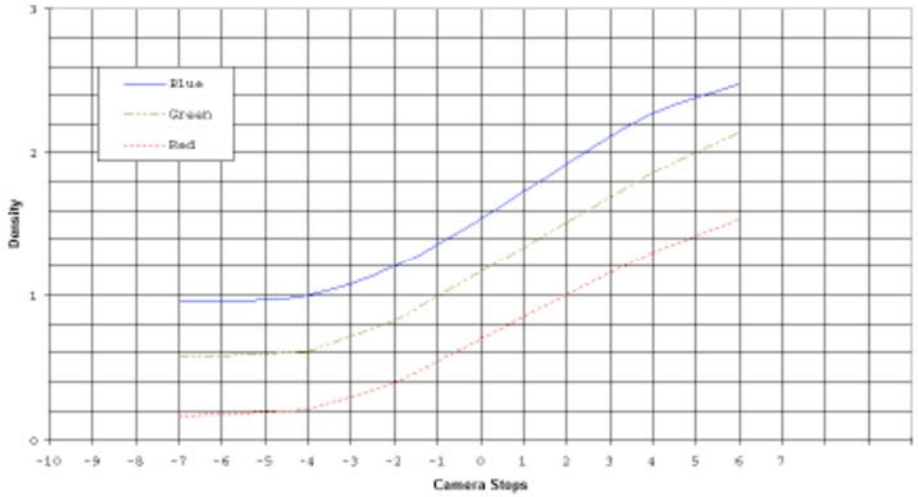
Lossless Image Compression

- In film days the negative held the information
- Generally negative had at least 20,000:1 dynamic range
- About 14-15 bit accuracy (if scan the film)
- Color film had same accuracy for each color
- Typical film resolution = 1.25 um pixels
- For 36x24 mm need 28,880 x 19,200 pixel scan
- At 16 bit
- 552 MB to 1.044 TB file



Notice: While the data presented are typical of production coatings, they do not represent standards which must be met by Eastman Kodak Company. Varying storage, exposure, and processing conditions will affect results. The company reserves the right to change and improve the product characteristics at any time.

Lossless Image Compression

- With modern sensors the number of pixels large but not film level
- Typically 15Mpixel to 50 Mpixel
- Each pixel typically is 10-14 bit information
- Simplest RAW recording would be bit map (BMP) but very large
- For 10 Mpixel image would be 10 MB (8 bit) or 20 MB (if 16bit)
- RAW images
- Generally RAW files instead use some level of compression
- Lossless compression: always can get the original info back
- Eg TIFF: generally gets a 2:1 improvement over BMP
- Lossy compression or irreversible compression
- Makes inexact copy of the information
- Reason get much higher compression

RAW Image Files

- Contains raw sensor data with minimal processed data
- (also scanners use same concept)
- Also called Digital Negatives
- Generally only some noise suppression applied
- Ideally replicates the information level on film negatives
- Captures radiometric date: ie light landing on sensor
- Each camera company has its own RAW
- File contents
- Short header – byte ordering of the file, file identifier
  Offset to file data
- Camera sensor metadata: size of sensor
  Color Filter Array arrangement
  Color profile
- Image Metadata:
- exposure settings, camera/scanner/lens model,
- date (and, optionally, place) of shoot/scan,
- authoring information
- other. Some contain a standardized metadata section in Exif format
- Most RAW files use TIFF lossless compression
- Reduces file size by 50%
- TIFF has many options so companies implement variations
- Adobe created open source standard DNG in 2004
- Digital NeGative (DNG)
- Adopted by Leica, Samsung, Ricoh, Pentax, Hasselblad

EXIF

- EXchangeable Image Format
- JEIDA/JEITA/CIPA Standard format
- From Japanese Electronic Industries Development Association
- Created in 1998
- Supported by all camera manufactures
- Date and time information.
- Camera setting: static: camera model and make, a
- Variable with each image
- orientation (rotation), aperture, shutter speed, focal length,
- metering mode, and ISO speed information.
- A thumbnail for previewing the picture on the camera's LCD screen
- Descriptions
- Copyright information.

EXIF Data

- Sample EXIF data

Tag	Value
Manufacturer	CASIO
Model	QV-4000
Orientation (rotation)	top-left [8 possible values[22]]
Software	Ver1.01
Date and time	2003:08:11 16:45:32
YCbCr positioning	centered
Compression	JPEG compression
X resolution	72.00
Y resolution	72.00
Resolution unit	Inch
Exposure time	1/659 s
F-number	f/4.0
Exposure program	Normal program
Exif version	Exif version 2.1
Date and time (original)	2003:08:11 16:45:32
Date and time (digitized)	2003:08:11 16:45:32
Components configuration	Y Cb Cr –
Compressed bits per pixel	4.01
Exposure bias	0.0
Max. aperture value	2.00
Metering mode	Pattern
Flash	Flash did not fire
Focal length	20.1 mm
MakerNote	432 bytes unknown data
FlashPix version	FlashPix version 1.0
Color space	sRGB
Pixel X dimension	2240
Pixel Y dimension	1680
File source	DSC
Interoperability index	R98
Interoperability version	(null)

TIFF

- Tagged Image File Format (TIFF)
- Latest version is 6.0 from 1992
- Flexible, adaptable format for images & data
- Allows Header tags
- Generally uses LZW compression
- This created problem until patent expired in 2004
- All contain 3 options
- Baseline TIFF – all programs can read this
- TIFF extensions – sets image types and compression
- Created Thousands of Incompatible File Formats (TIFF)
- Most images use LZW compression

Lempel–Ziv–Welch (LZW) Compression

- LZW is the standard lossless compression used in mages
- Abraham Lempel, Jacob Ziv, and Terry Welch in 1984
- Improved version of their 1978 algorithm
  - Works by reading a sequence of symbols,
  - grouping the symbols into strings
  - converting the strings into codes
- As codes take less space get compression
- Uses a code table typical 4096
- Codes 0-255 single bits
- Initially after 256 entries table is blank
- Identify repeated sequences in data
- adds them to the code table
- Decoding takes each code from the compressed file and translates
- Example 1:
- LZW algorithm to compress the string: BABAABAAA

BABAABAAA P=A C = empty

Encoder Output		String Table	
Output Code	representing	codeword	string
66	B	256	BA

LZW compression step 1

BABAABAAA P=B C = empty

Encoder Output		String Table	
Output Code	representing	codeword	string
66	B	256	BA
65	A	257	AB

LZW compression step 2

BABAABAAA P=A C = empty

Encoder Output		String Table	
Output Code	representing	codeword	string
66	B	256	BA
65	A	257	AB
256	BA	258	BAA

LZW compression step 3

BABAABAAA P=A C = empty

Encoder Output		String Table	
Output Code	representing	codeword	string
66	B	256	BA
65	A	257	AB
256	BA	258	BAA
257	AB	259	ABA

LZW compression step 4

BABAABAAA P=A C = A

Encoder Output		String Table	
Output Code	representing	codeword	string
66	B	256	BA
65	A	257	AB
256	BA	258	BAA
257	AB	259	ABA
65	A	260	AA

LZW compression step 5

BABAABAAA P=AA C = empty

Encoder Output		String Table	
Output Code	representing	codeword	string
66	B	256	BA
65	A	257	AB
256	BA	258	BAA
257	AB	259	ABA
65	A	260	AA
260	AA		

LZW compression step 6

LZW Decompression

- Example 2: LZW Decompression:
- Decompress the output sequence of
- <66><65><256><257><65><260>

<66><65><256><257><65><260> Old = 65 S=A
New = 66 C=A

Encoder Output	String	Table
string	codeword	string
B		
A	256	BA

LZW compression step 1

<66><65><256><257><65><260> Old = 256 S=BA
New = 256 C=B

Encoder Output	String	Table
string	codeword	string
B		
A	256	BA
BA	257	AB

LZW compression step 2

<66><65><256><257><65><260> Old = 257 S=AB
New = 257 C=A

Encoder Output	String	Table
string	codeword	string
B		
A	256	BA
BA	257	AB
AB	258	BAA

<66><65><256><257><65><260> Old = 65 S=A
New = 66 C=A

Encoder Output	String	Table
string	codeword	string
B		
A	256	BA
BA	257	AB
AB	258	BAA
A	259	ABA

LZW compression step 4

<66><65><256><257><65><260> Old = 260 S=AA
New = 260 C=A

Encoder Output	String	Table
string	codeword	string
B		
A	256	BA
BA	257	AB
AB	258	BAA
A	259	ABA
AA	260	AA

LZW compression step 5

LZW Summary

- LZW compresses repetitive sequences of data very well.
- Since the codewords are 12 bits,
- any single encoded character will expand the data size
- rather than reduce it
- Advantages of LZW over Huffman:
- LZW requires no prior information about the input data stream.
- LZW can compress the input stream in one single pass.
- Another advantage of LZW its simplicity, allowing fast execution
- Used in TIFF means the original data is always there
- JPEG (lossy) ends of meaning the picture loses data
- But gets about 7x data reduction