



Lossy PNG

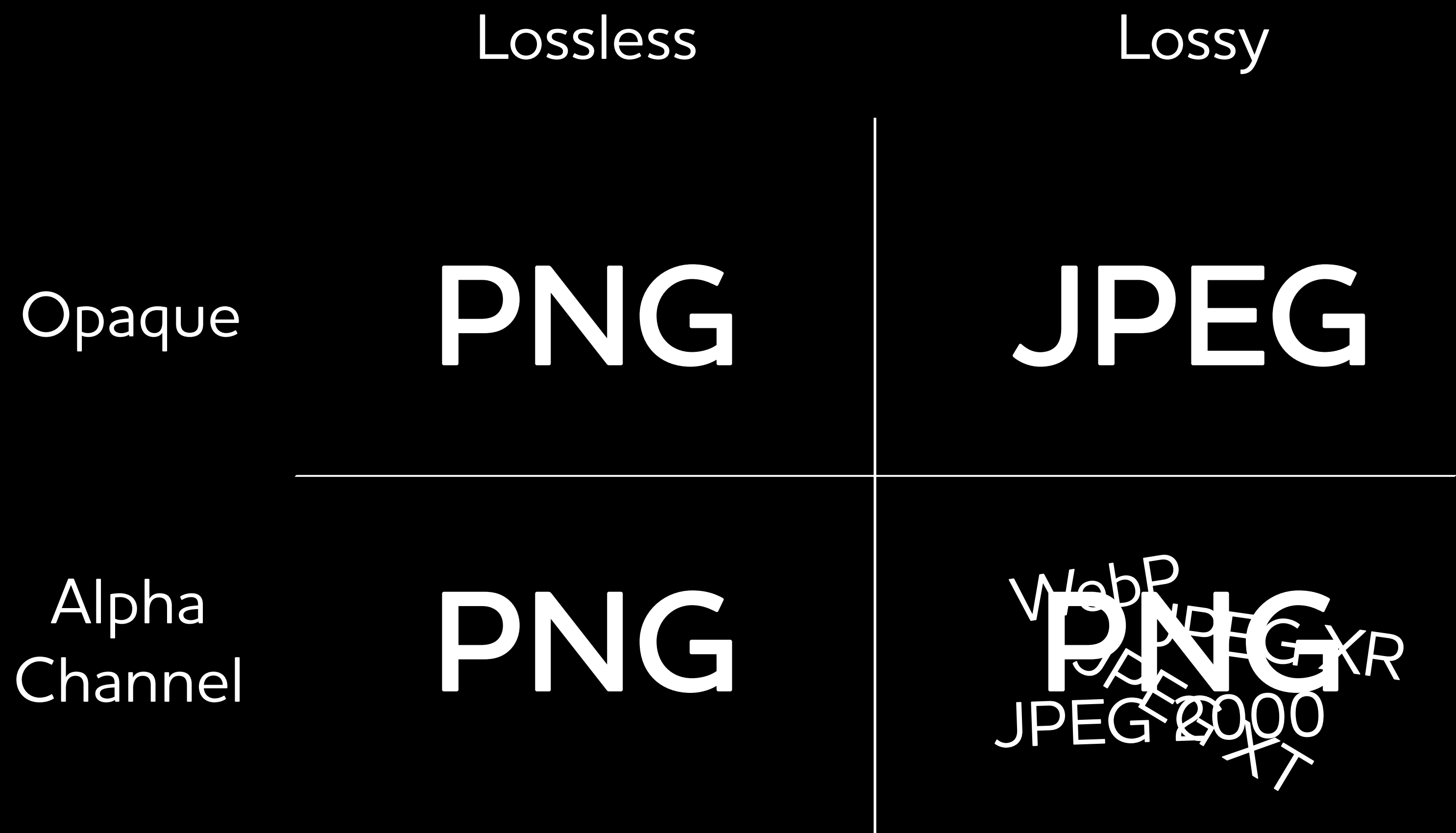
Kornel Lesiński

Lossless

PNG

Lossy

JPEG



if you need lossy *and* alpha, it becomes complicated. You may need to serve 4 different files for every image!

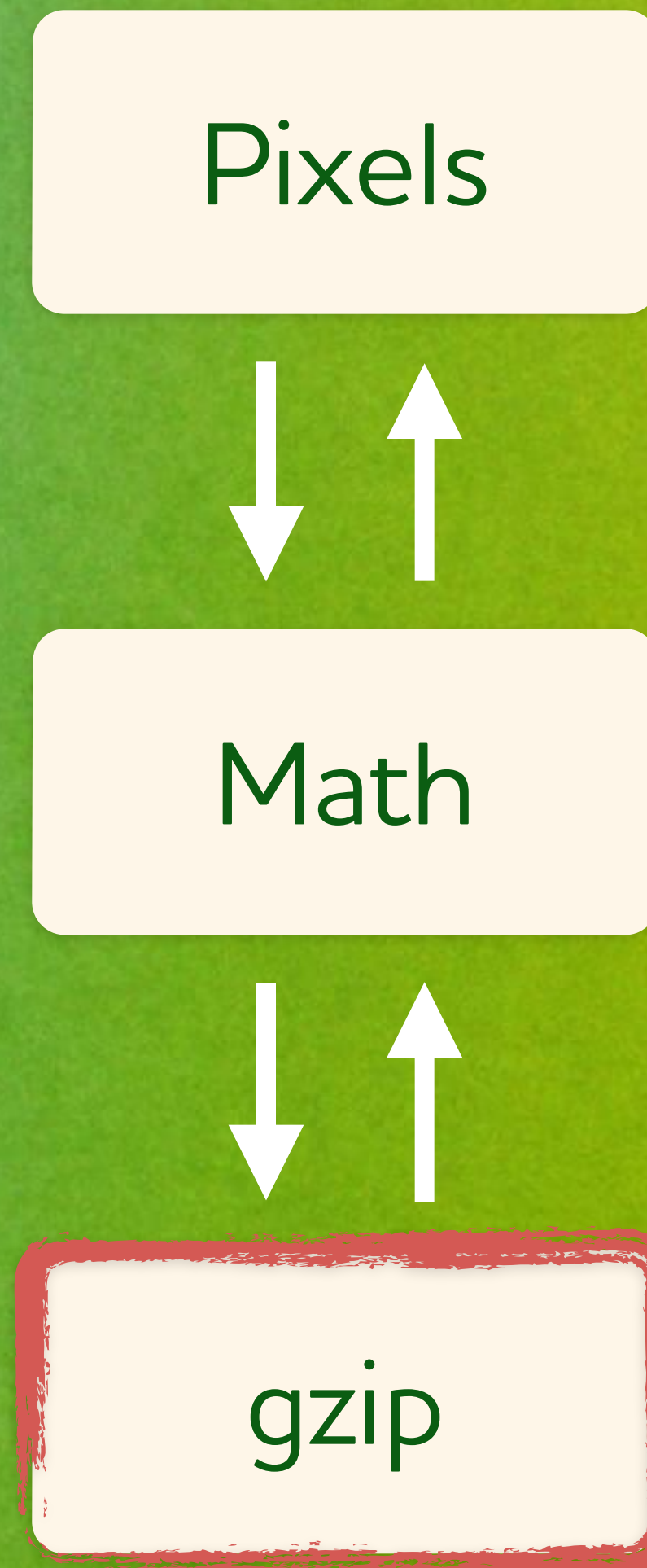


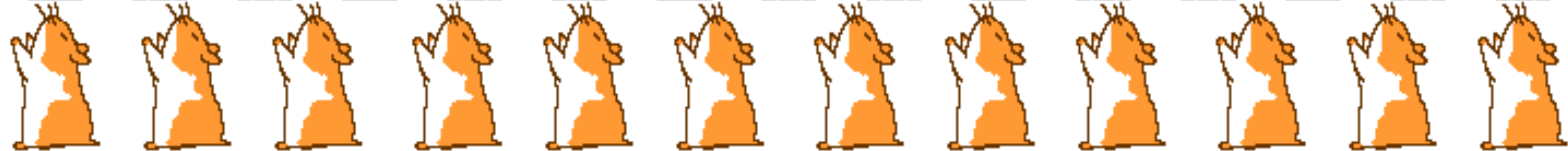
[flickr.com/photos/ama711/8760361969](https://www.flickr.com/photos/ama711/8760361969)



Lossy PNG is a hack, but it works like a truly lossy format

Standard PNG





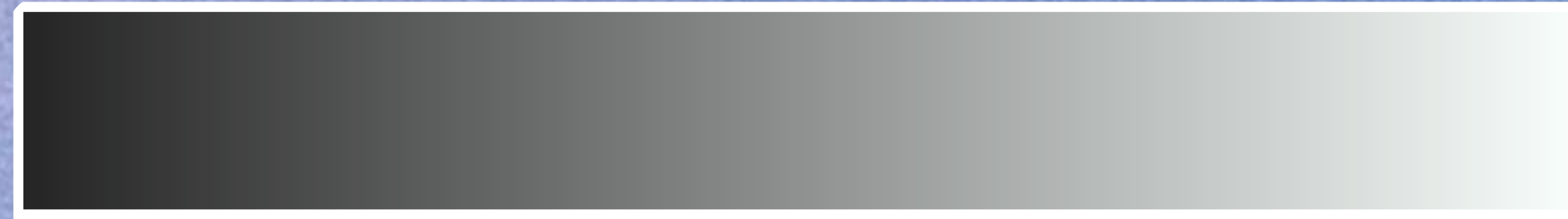


gzip finds and removes repeated sequences of identical pixels

00000000	→	0
00000001	→	10
00000010	→	110
00000011	→	1110
00000100	→	11110
00000101	→	111110
00000110	→	1111110
00000111	→	11111110
00001000	→	111111110
00001001	→	1111111110
00001010	→	11111111110

Instead of storing 8 bits per byte, gzip uses variable-length sequences. Most popular bytes are given shortest sequences.

Some images don't have repeated pixels
and few very popular colors



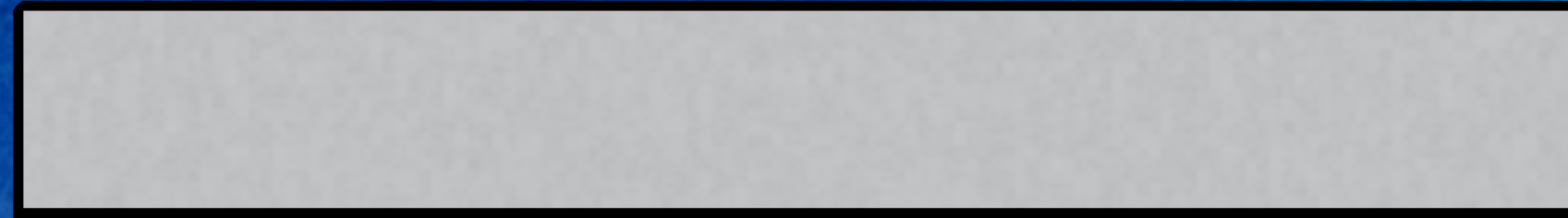
=

0 1 2 3 4 5 6 7 8 9

=

0 +1 +1 +1 +1 +1 +1 +1 +1 +1

Instead of storing uncompressible absolute values
PNG can store difference between neighboring pixels



=

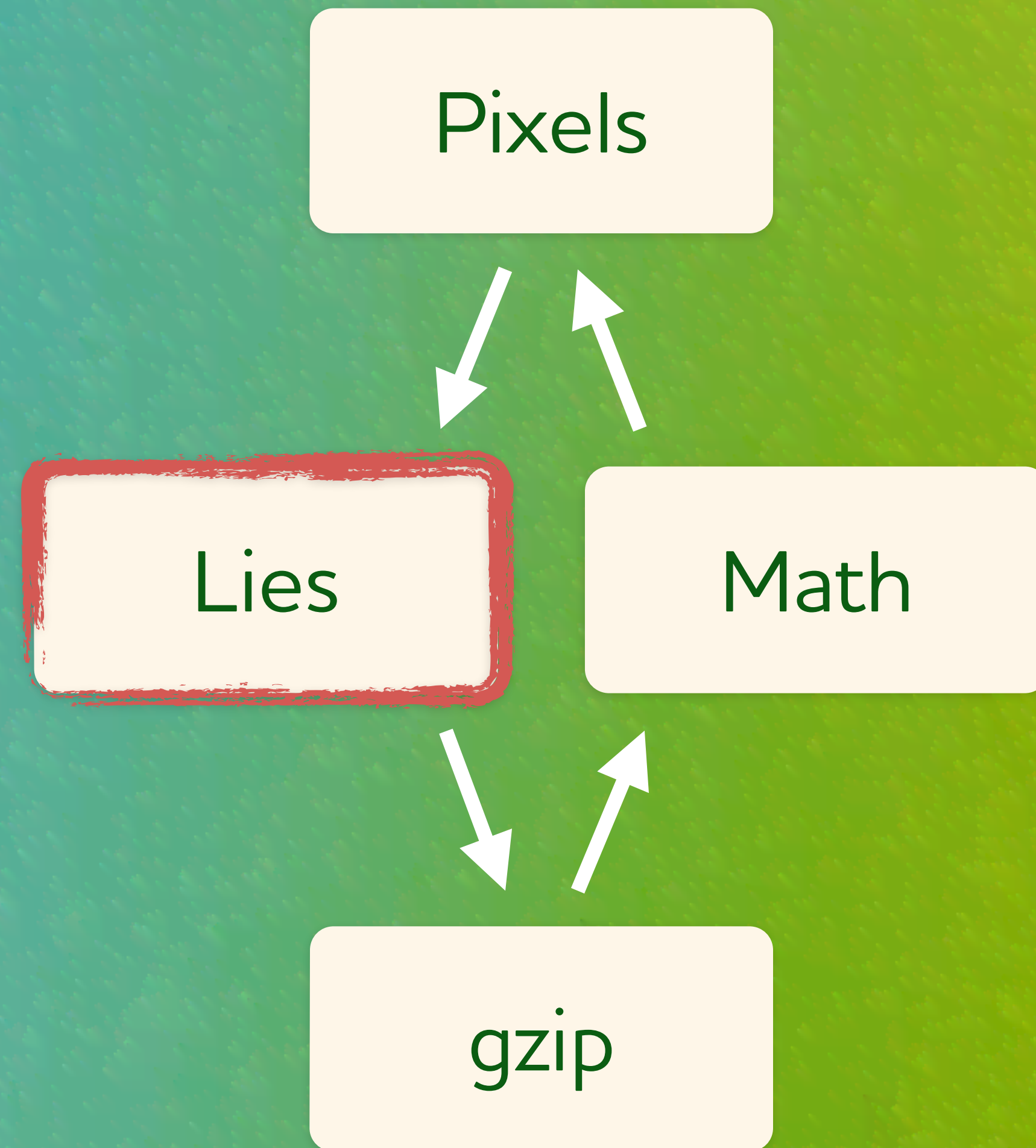
5 6 4 6 5 5 6 4 5 4

=

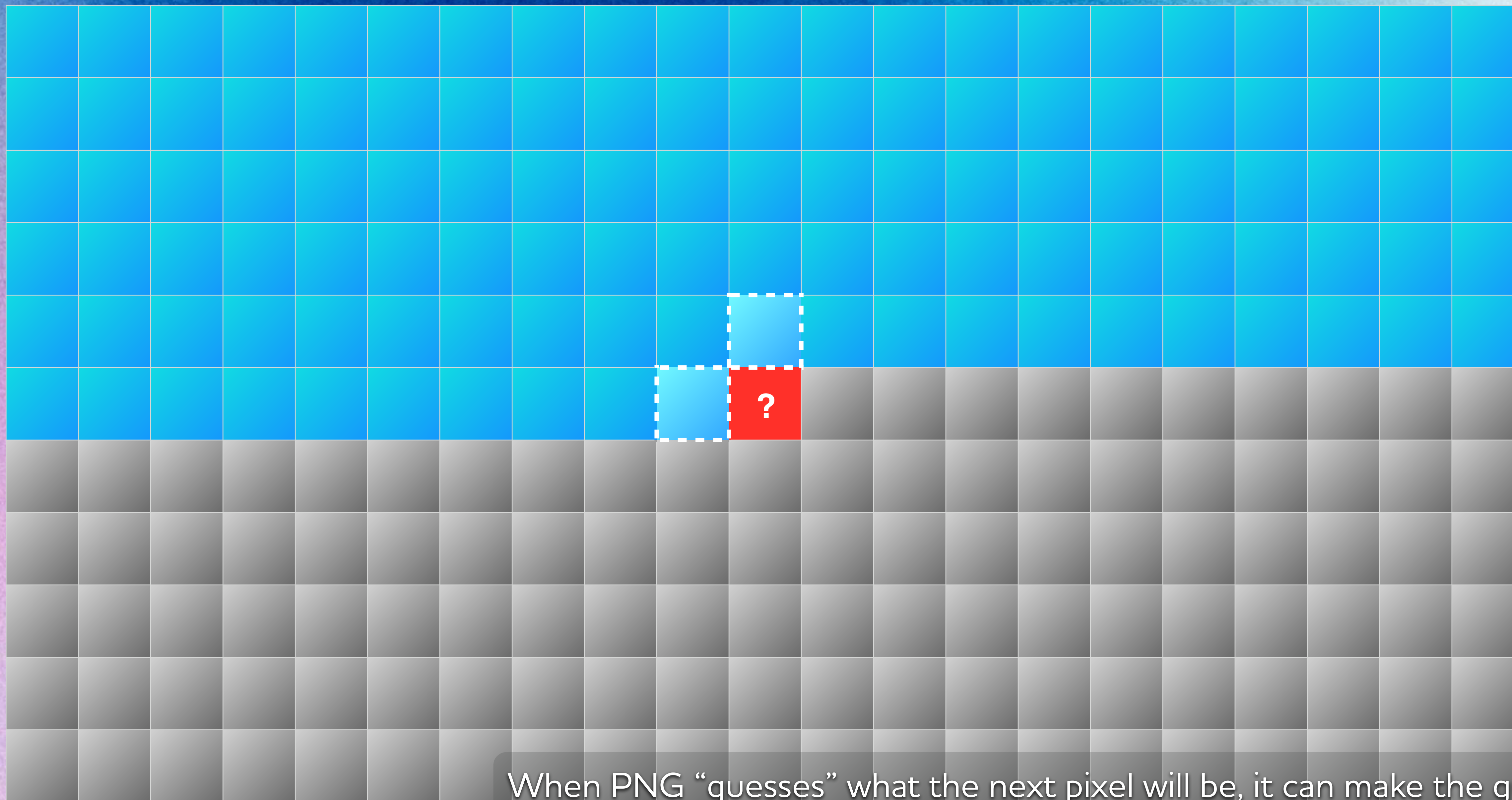
~~5 +1 -2 +2 -1 0 +1 -2 +1 -1~~

5 0 0 0 0 0 0 0 0 0

Noisy images won't compress well...
unless you fake the data and remove the noise



Lossy PNG trick is about replacing math with lies. Only the encoder changes, so it's 100% compatible.



When PNG “guesses” what the next pixel will be, it can make the guess based on average of 2 neighbors, which in lossy PNG causes blur

Original: 2MB



0.6MB



0.3MB



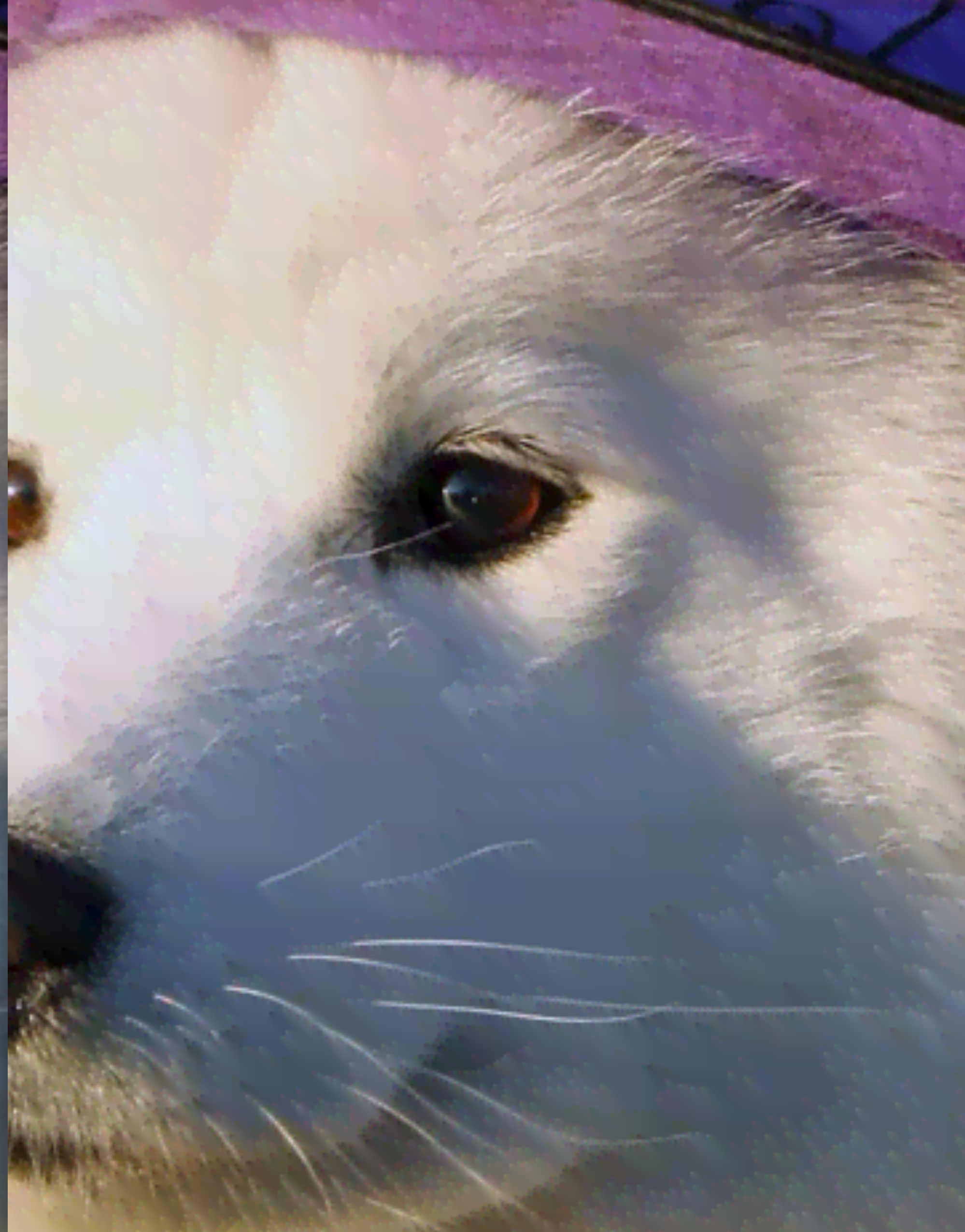
Original: 2MB



0.6MB

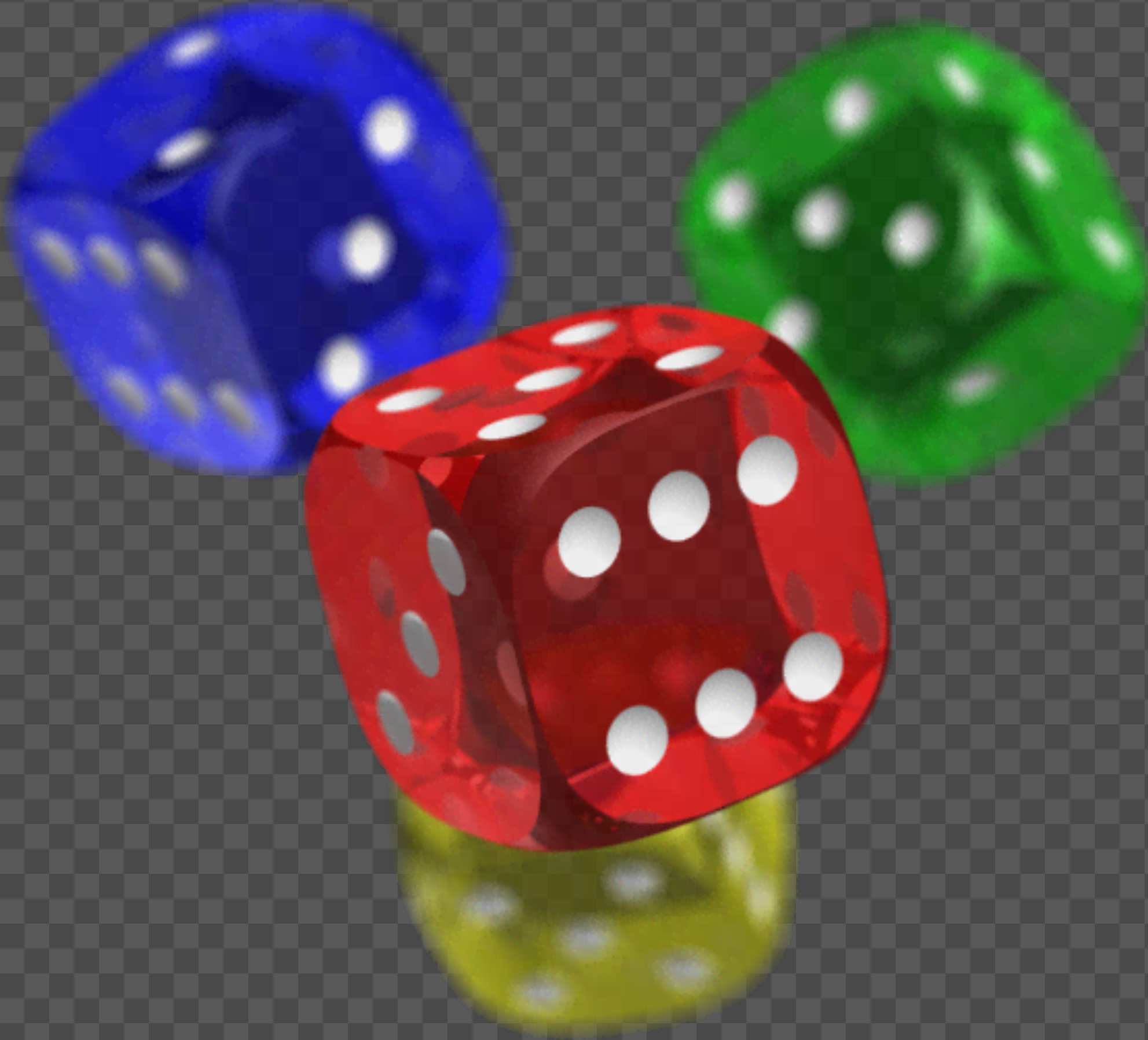


0.3MB



Original: 221KB

Lossy: 67KB



182KB

compression

WOW

27KB

compression

WOW

The algorithm can decide quality of every pixel, so it tends to preserve sharp edges, especially text

28KB

compression

WOW

Top of the file has 0% quality, bottom 80% quality

- **Works everywhere**—fully backwards-compatible
- **No color limit**—supports 24 and 32-bit PNG
- **Preserves edges**—per-pixel quality control
- **Very fast**—small tweak in the encoder
- **Streamable**—only needs 2 lines of the image

```
pngquant -Q 75-100 --skip-if-larger \  
-o out.png in.png ||  
posterize -b -Q 75 in.jpg out.jpg
```

pngquant is often more efficient, but can't guarantee quality. You can combine the two for best results.

@pornelski

pngmini.com/lossypng.html

GUI pngmini.com — ImageAlpha

GUI logicnet.dk/Analyzer — Image Analyzer

CLI github.com/pornel/mediancut-posterizer