

ASTC Does It

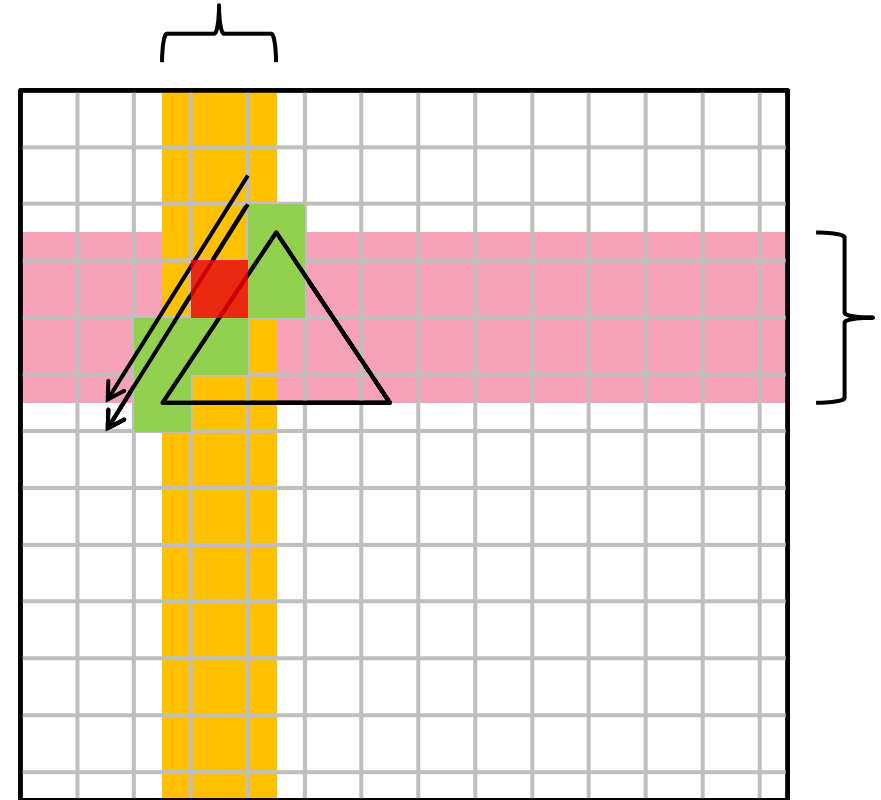
Stacy Smith
Senior Software Engineer, ARM

Complete Guide to Texture Compression (Abridged)

- Texture data is big
- $32\text{bpp} * 1024 * 1024 = 4\text{MB}$ per texture
- This is the reason for jpg and png
- Hardware decompression saves memory and bandwidth

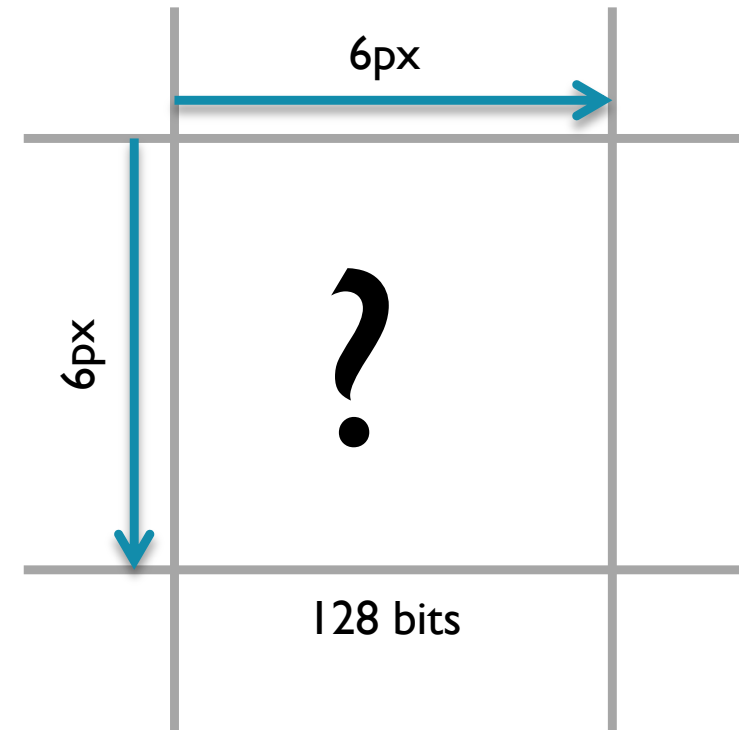
Complete Guide to Texture Compression (Abridged)

- Hardware needs random access
- Texture compression is block based
- Look up a block from the texel coords
- Decompress into local cache
- Sample cached block



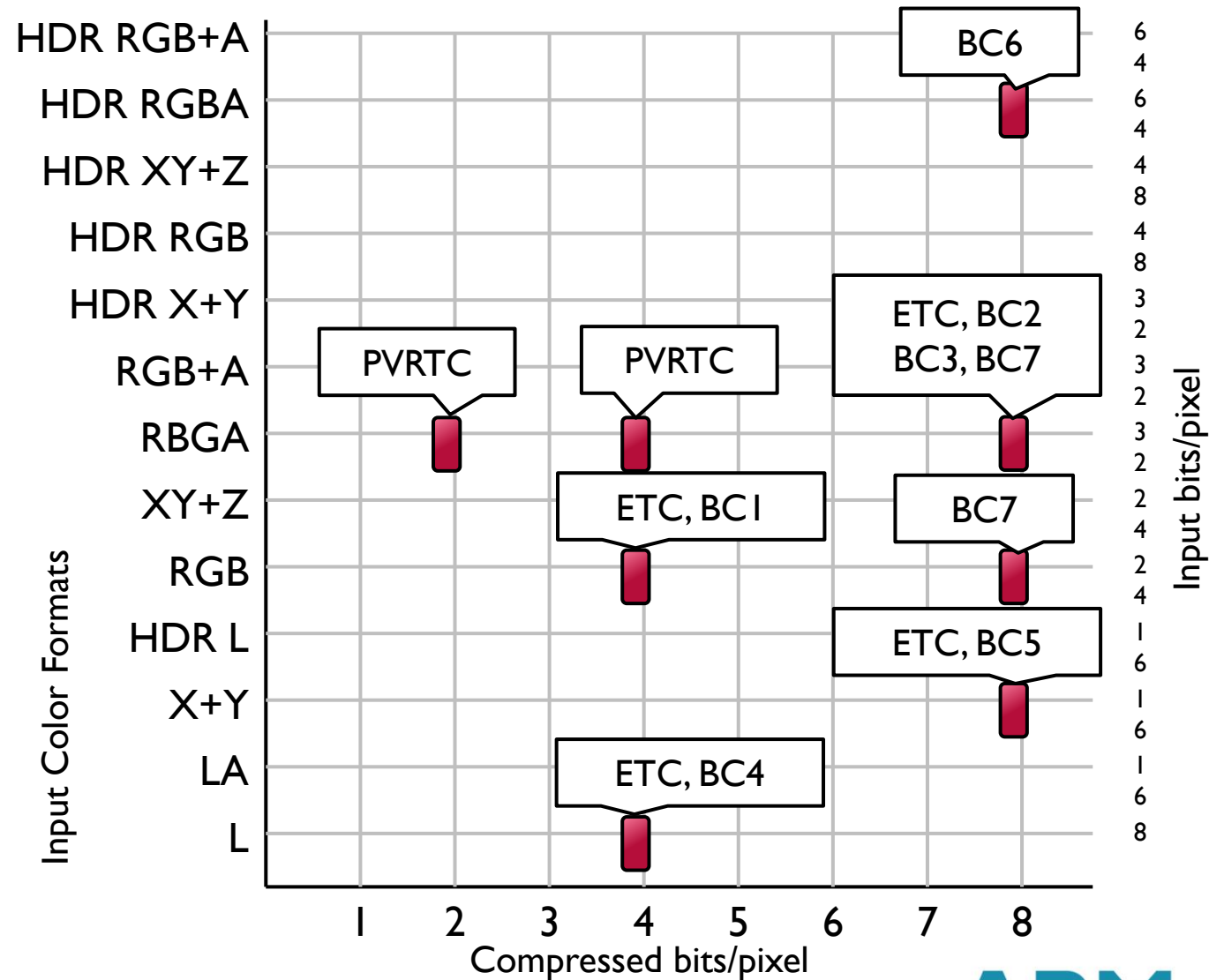
Complete Guide to Texture Compression (Abridged)

- Details about block must be known
- Block footprint dimensions
- Block size in memory
- How the block is encoded
- There are many, *many* encodings



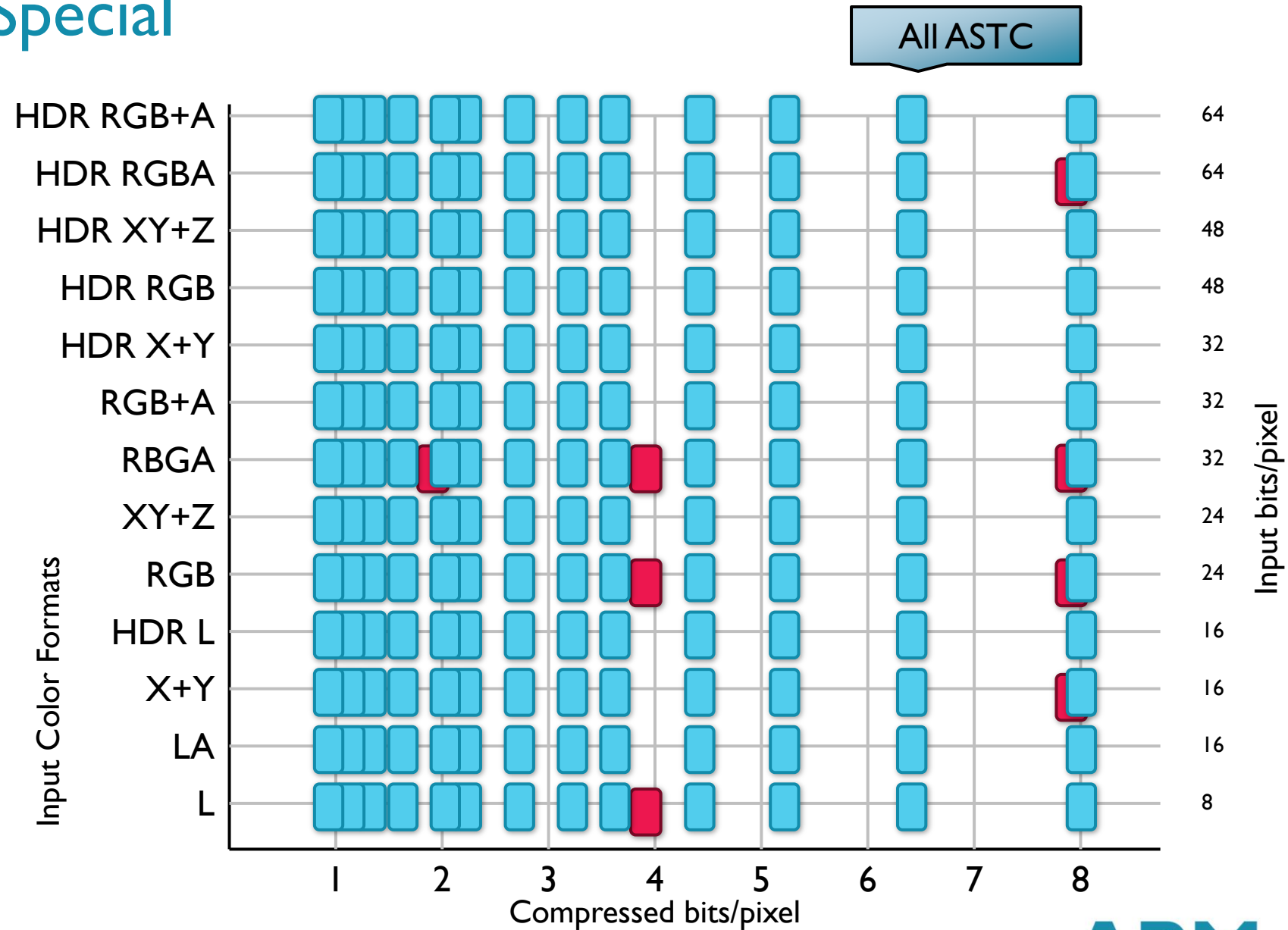
Complete Guide to Texture Compression (Abridged)

- Different codecs support different footprints and bitrates
- They also support different color encodings (HDR / sRGB / alpha)
- Hardware limits access to proprietary codecs



What Makes ASTC Special

- Wide range of bitrates
- Wide range of formats
- Handles sRGB
- Handles HDR
- 3D Textures
- Non Proprietary



Why Not Try ASTC Right Now?

- Command line compressor
 - ASTC Evaluation Codec
- GUI compressor
 - ARM® Mali™ Texture Compression Tool
- Lacking compatible hardware?
 - ARM Mali OpenGL® ES 3.0 Emulator

Mali Developer Centre:

MaliDeveloper.arm.com

What Defines Quality?

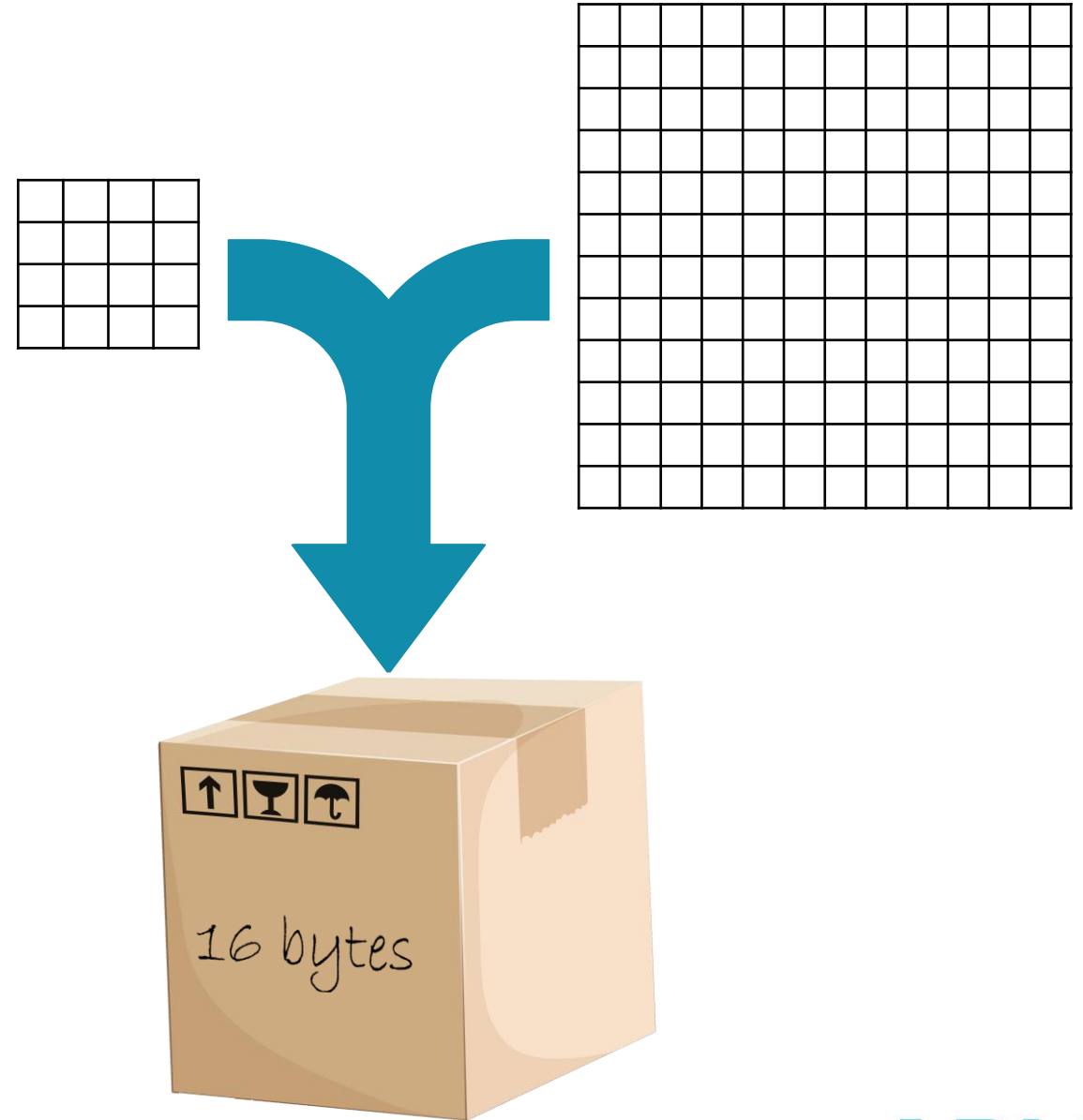
- Compression process is asymmetric
 - Tile extracted from data points
 - Finding right points is mostly trial and error
- Quality decided by 3 factors
 - Precision of data points (bit rate)
 - Number of attempts per tile (limits)
 - Types of error to reject/ignore (priority)



What Defines Quality?

Bit Rates and Block Size

- Every block in ASTC has the same data size
 - 128 bits
- Bit rate is decided by block footprint
 - From 4x4 (8 bpp) to 12x12 (0.89 bpp)
- More data per pixel, closer to original
- Saves less bandwidth



What Defines Quality?

Limits and Leeway

- Trial and error process
 - More tries, more opportunities to find a good match
 - Takes longer to compress
- Error bounds and quality limits
 - Move on if the early tries look bad
 - Stick with an early try if it is 'good enough'
 - Stop looking if a block takes too long
- Mostly managed through presets



What Defines Quality?

Priority and Perception

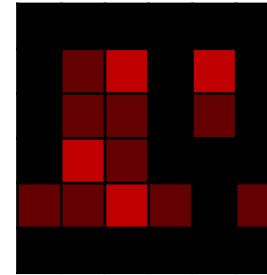
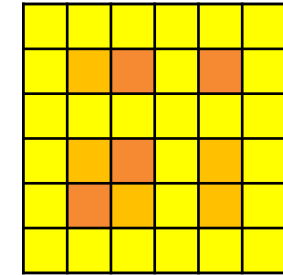
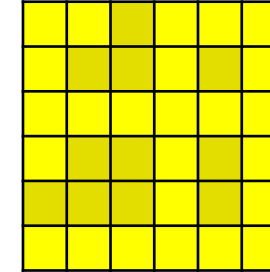
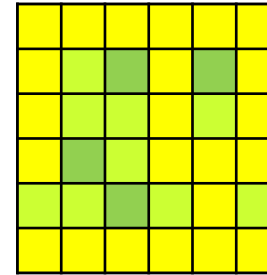
- Weightings on specific error types



What Defines Quality?

Priority and Perception

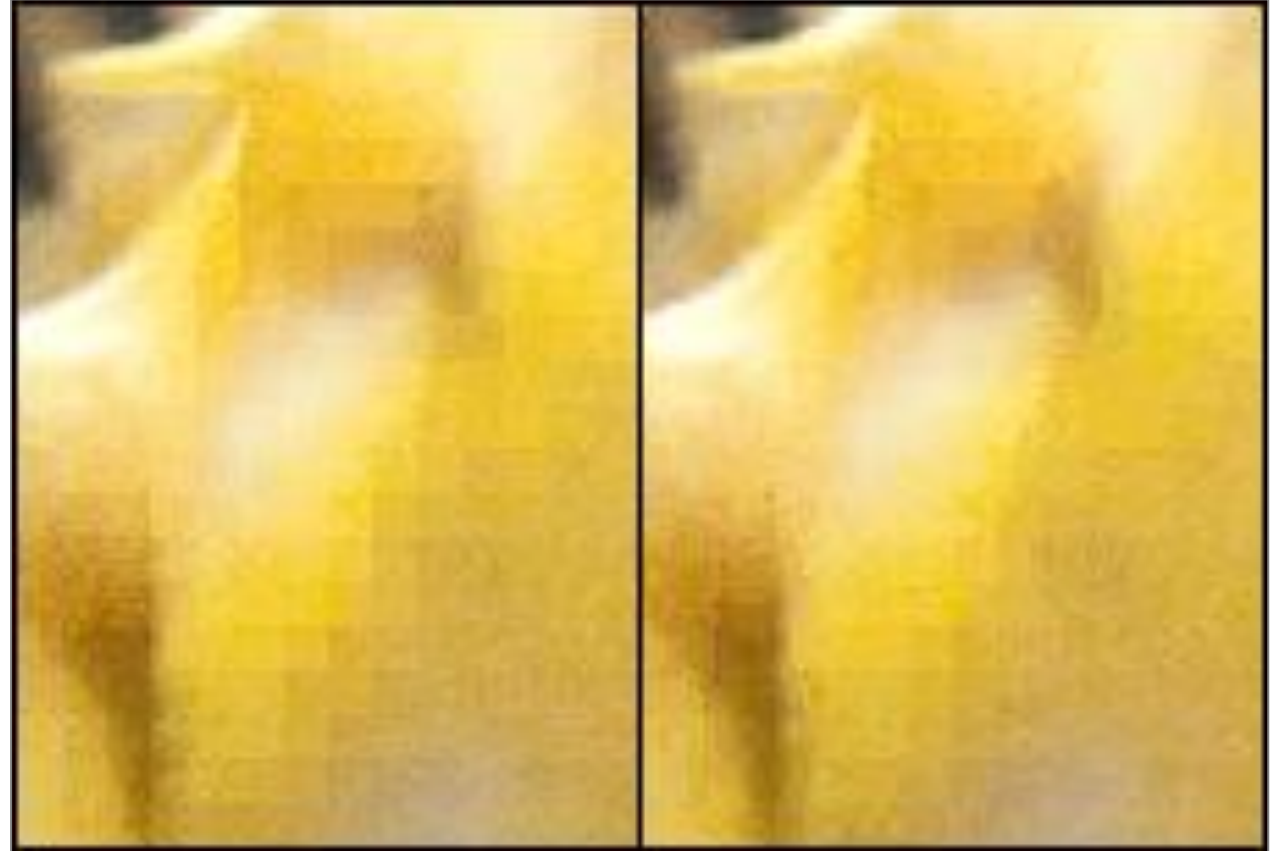
- Weightings on specific error types
- Per channel weighting



What Defines Quality?

Priority and Perception

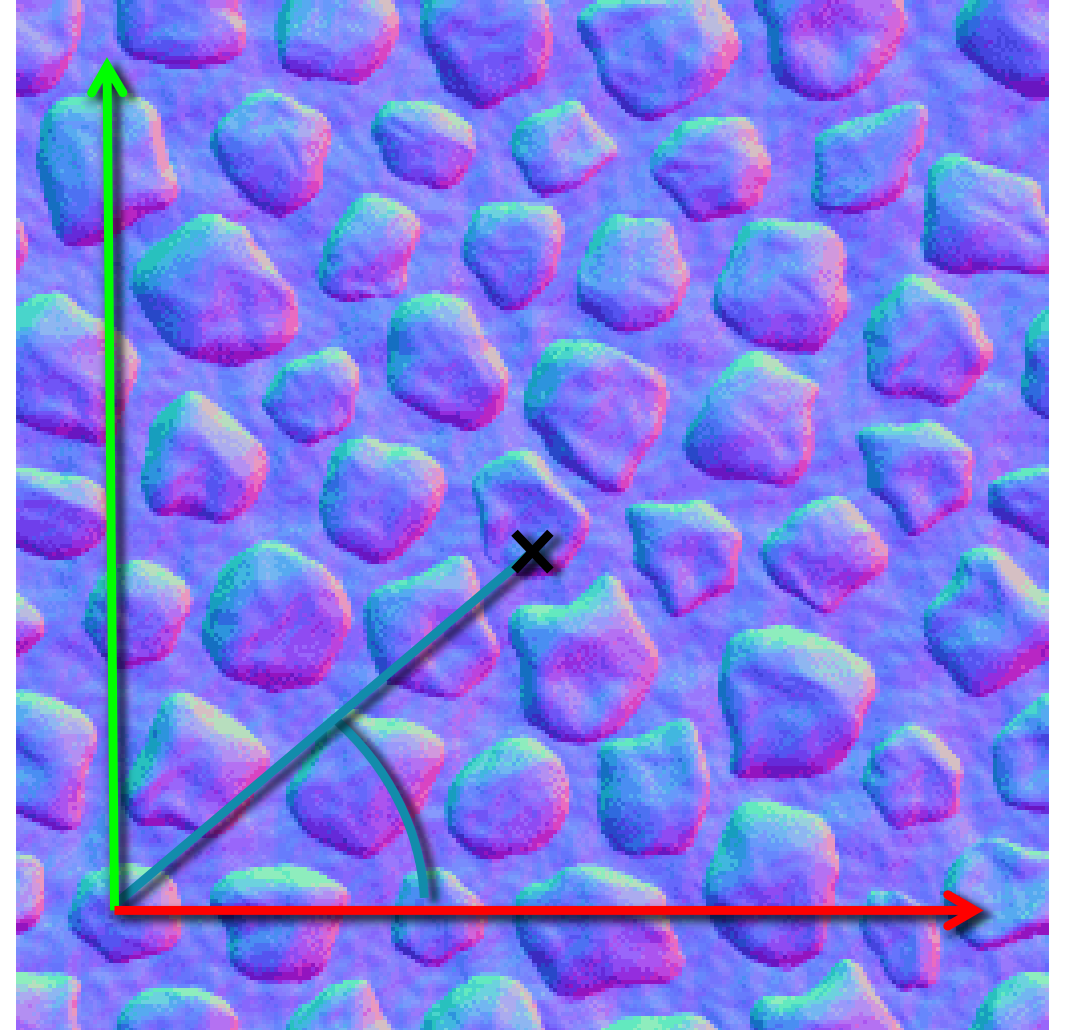
- Weightings on specific error types
- Per channel weighting
- Block weighting



What Defines Quality?

Priority and Perception

- Weightings on specific error types
- Per channel weighting
- Block weighting
- Angular weighting



What Defines Quality?

Priority and Perception

- Weightings on specific error types
- Per channel weighting
- Block weighting
- Angular weighting
- Absolute value weighting

$$error = \frac{1}{baseweight + avg\,scale^2 * average + stdev\,scale * stdev^2}$$

Per-texel relative error weighting for the RGB color channels

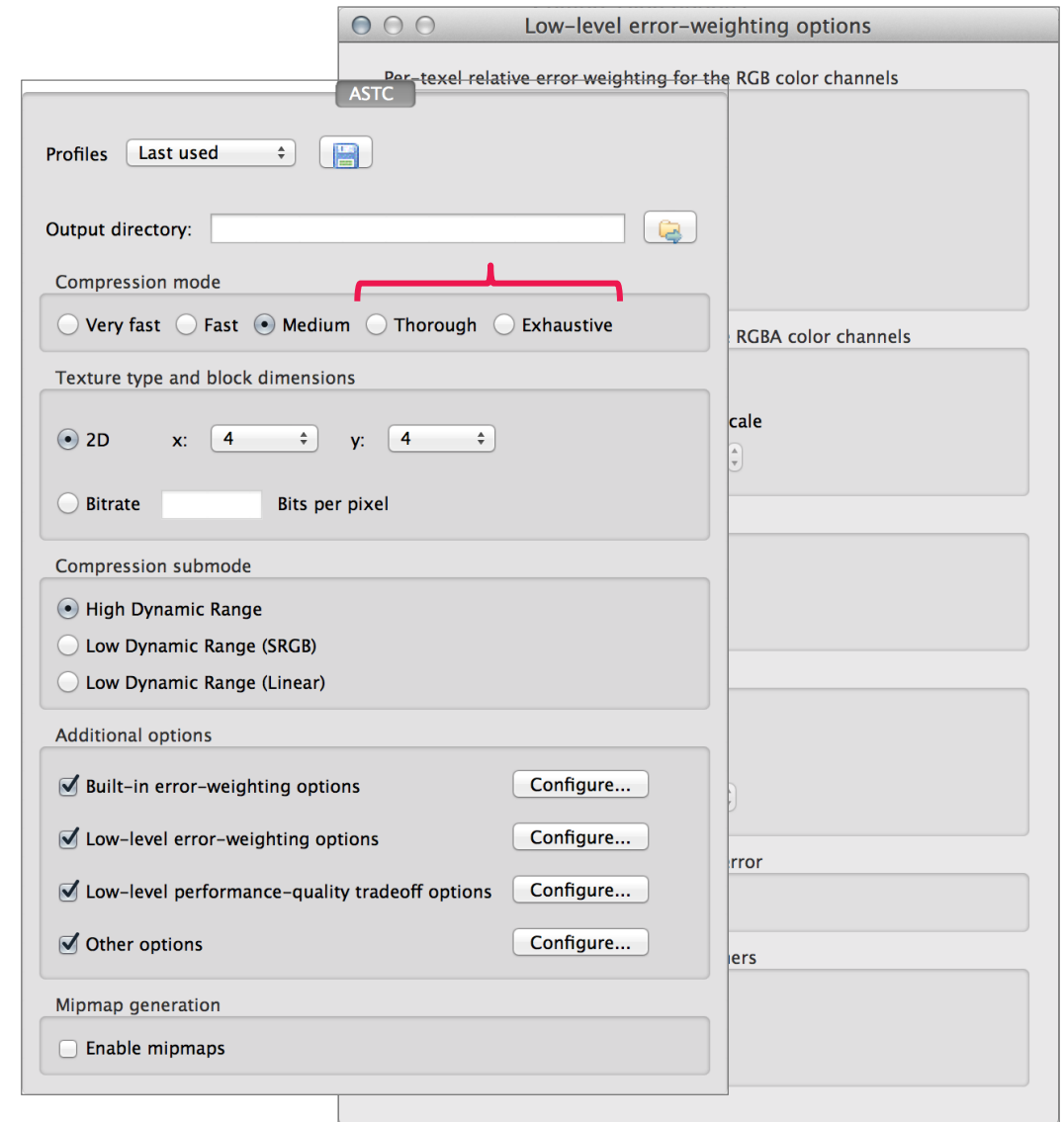
☐ Apply

Radius	Power	Base weight
<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
Avg scale	Stdev scale	Mixing factor
<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>

Use Case

Standard Diffuse Color Texture

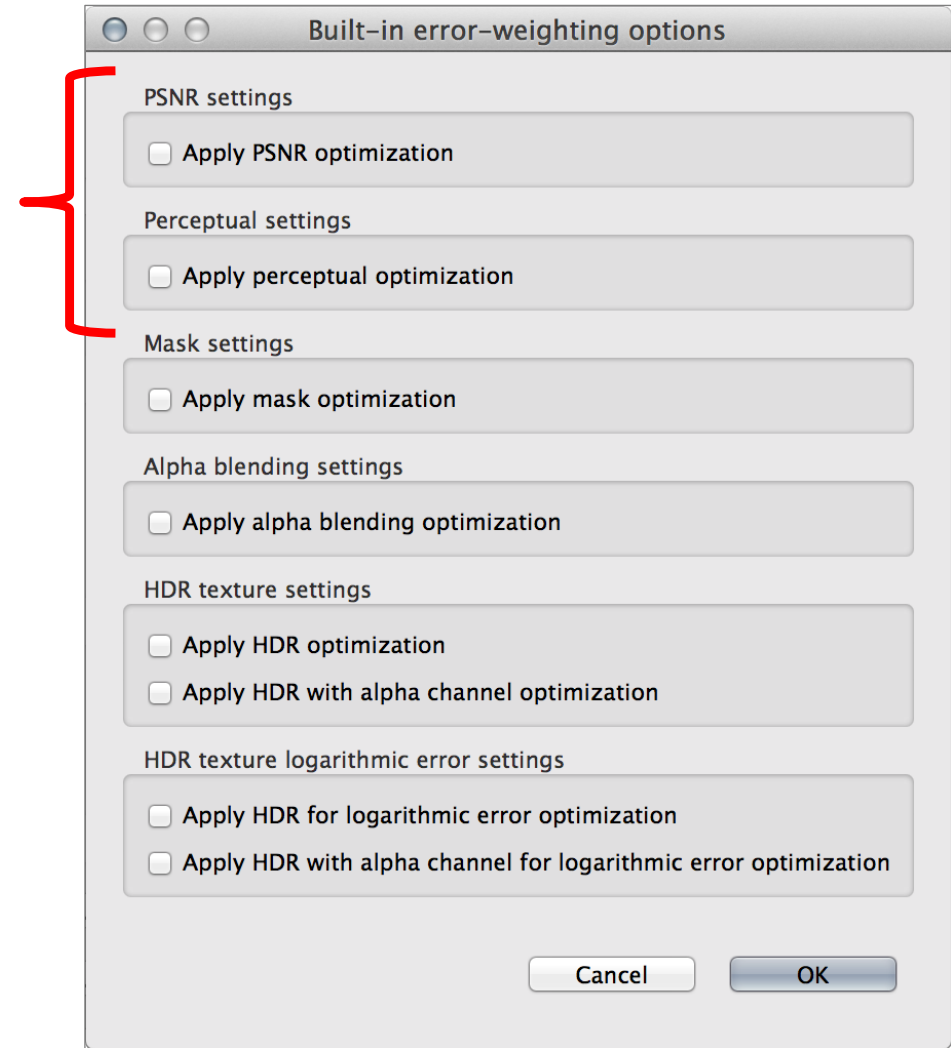
- Standard settings normally okay
- Add block weighting for large gradients
- Channel weighting for photos
 - Green is most noticeable
- Always use `–thorough` or `–exhaustive` for final assets



Use Case

Tangent Space Normal Map

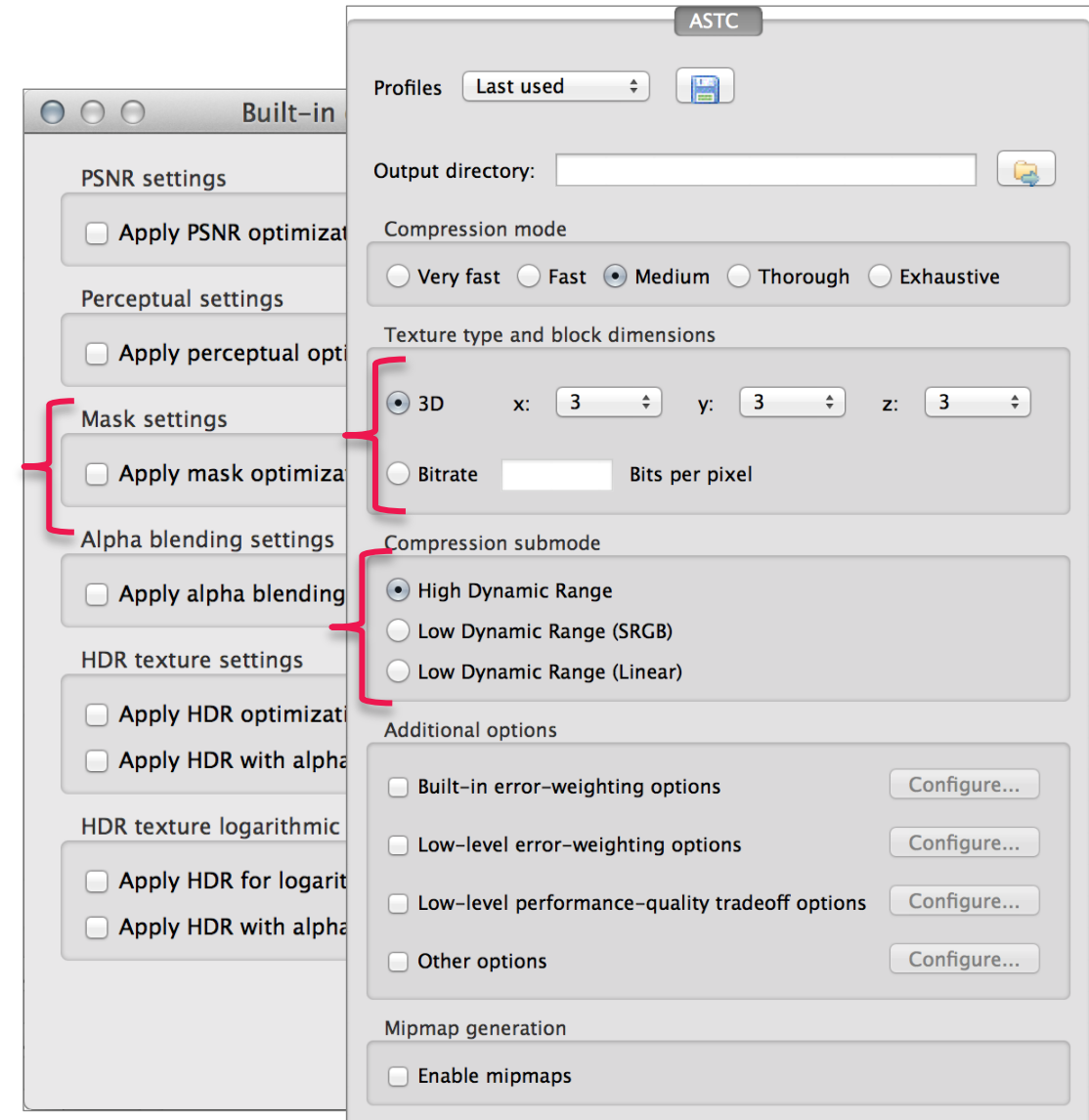
- Use -normal-psnr or -normal-percep
- This encodes the X and Y in Lum/Alpha
- $z = \sqrt{1 - (x^2 + y^2)}$
- Automatic weightings:
 - Angular component
 - Block edges



Use Case

Something a Bit Different

- Use -mask to dissociate channel errors
- Use -array <size> to load slices of a 3D texture
- Use -srgb to linearize rgb before encode



Why Not Try ASTC Right Now?

- Command line compressor
 - ASTC Evaluation Codec
- GUI compressor
 - ARM® Mali™ Texture Compression Tool
- Lacking compatible hardware?
 - ARM Mali OpenGL® ES 3.0 Emulator

Mali Developer Centre:

MaliDeveloper.arm.com

Thank you
Any questions?

MaliDeveloper.arm.com

community.arm.com

ARM

Thank You

The trademarks featured in this presentation are registered and/or unregistered trademarks of ARM Limited (or its subsidiaries) in the EU and/or elsewhere. All rights reserved. Any other marks featured may be trademarks of their respective owners