DHTC: An Effective DXTC-based HDR Texture Compression Scheme

Wen Sun[1,2] Yan Lu[1] Feng Wu[1] Shipeng Li[1]

yanlu@microsoft.com

[1] Microsoft Research Asia

[2] University of Science and Technology of China

Microsoft Research

Outline

- Background and related work
  - High dynamic range (HDR) texture and its compression
- DHTC: DXTC-based HDR texture compression
  - Extension of LDR texture format (i.e. DXTC) to HDR texture compression
  - Unified 8-bpp format for LDR textures, HDR textures and alpha maps
- Results and summary

HDR Textures

Low exposure

Medium exposure

High exposure

- The real world is high dynamic range
- A dynamic range of 10000:1 is common
- HDR rendering is gaining popularity in practice

HDR Texture Compression

- HDR textures are huge in size
  - Currently used FP32/FP16 formats: 96/48 bits per pixel, 4x/2x size of raw LDR RGB textures
  - Consume too much memory and bandwidth
- Current status
  - No HDR texture compression standard in industry
  - No graphics card supports rendering from block-wise compressed HDR textures

Previous Work

- [Wang et al. 2007]
  - 16-bpp HDR texture format
  - Utilization of current generation GPUs
- [Roimela et al. 2006, 2008]
  - 8-bpp HDR texture format
  - Simple hardware decoder
- [Munkberg et al. 2006, 2008]
  - 8-bpp HDR texture format
  - Near lossless visual quality

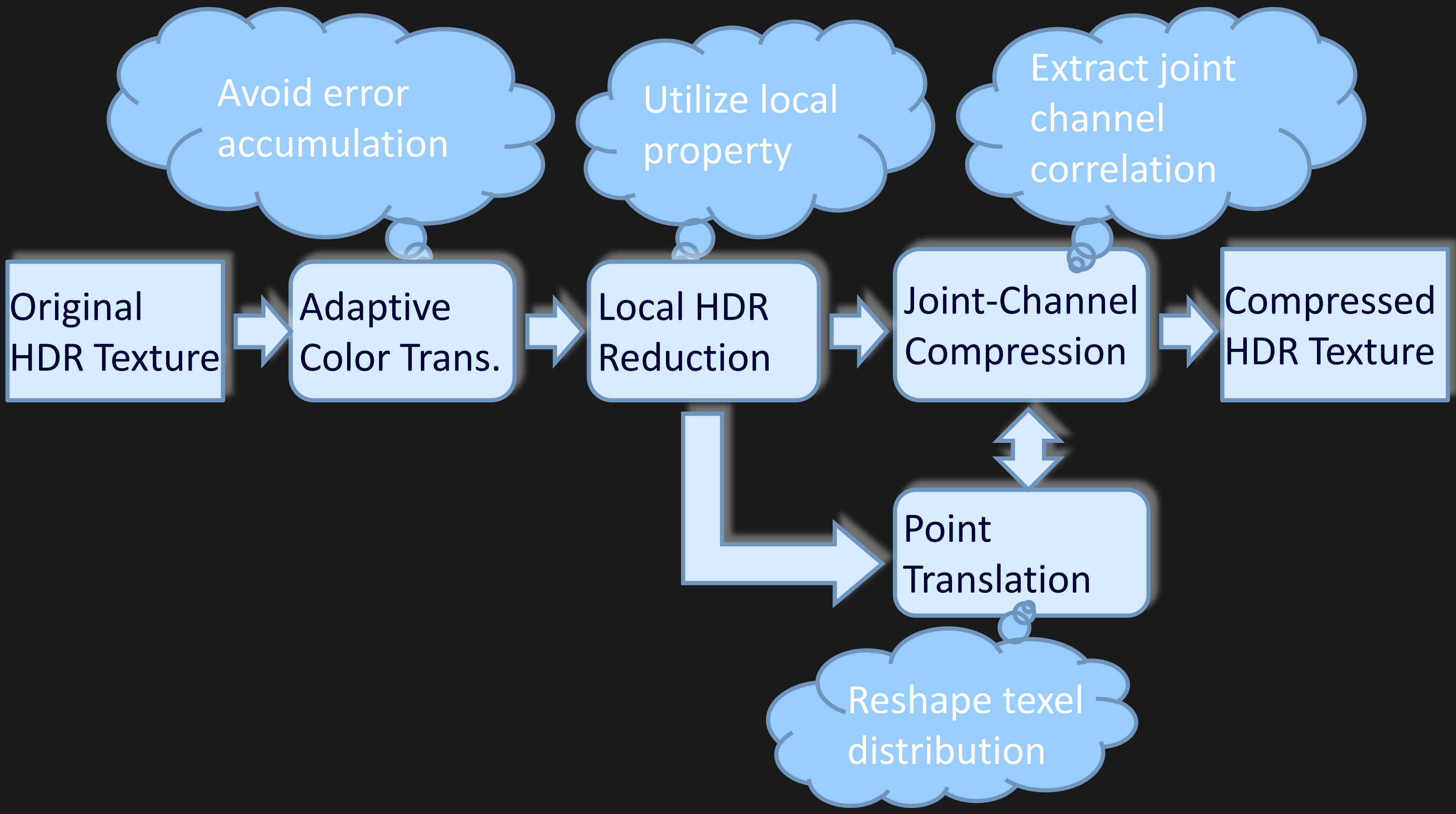
Our Insights

- HDR texture compression scheme can be built upon existing LDR texture compression scheme
  - Lead to a unified compression framework
  - Reuse existing hardware
- Joint color-channel compression can result in better visual quality
  - Joint-channel bit allocation
  - Utilization of cross-channel correlations
- It is a plus to support LDR textures and alpha maps in a single HDR texture format

Our Solution

- DXTC-based HDR texture compression framework
  - Utilize joint color-channel compression to provide advanced bit allocation
  - Utilize the existing DXTC hardware to reduce the adoption cost in industry
- 8 bpp compressed DHTC texture format
  - Near lossless visual quality for HDR textures
  - Support 1 bit alpha channel for HDR textures
  - Backward compatible to LDR RGBA textures

Framework



Avoid error accumulation
Utilize local property
Extract joint channel correlation
Original HDR Texture
Adaptive Color Trans.
Local HDR Reduction
Joint-Channel Compression
Compressed HDR Texture
Point Translation
Reshape texel distribution

Adaptive Color Transform

- Traditional color transform

Forward Color Transform

$$Y = w_r R + w_g G + w_b B$$

$$U = \frac{w_r R}{Y}$$

$$V = \frac{w_g G}{Y}$$

Explicit channels (Y, U, V)

$$W = \frac{w_b B}{Y}$$

Implicit channel (W)

Inverse Color Transform

$$R = U \times Y / w_r$$

$$G = V \times Y / w_g$$

Error controllable channels (R, G)

$$B = (Y - w_r R - w_g G) / w_b$$

Error accumulative channel (B)

Absolute errors

Adaptive Color Transform

- Our solution
  - Adaptively select the implicit channel to minimize the impact of error accumulation

Luminance and chrominance channels

$$Y = \sum_{t \in \{r,g,b\}} w_t C_t$$

$$S_t = \frac{w_t C_t}{Y}, t \in \{r,g,b\}$$

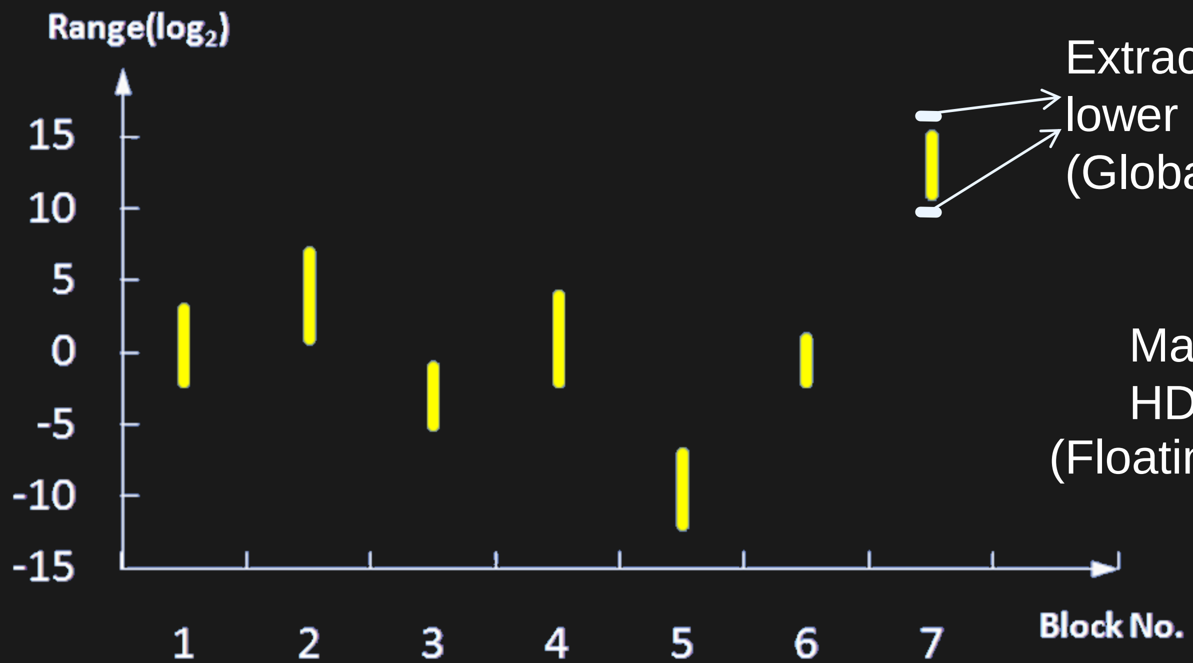
Keep the block dominant color channel from being explicitly encoded

Adaptive color transform mode

$$Ch\_mode = \arg\max_{t \in \{r,g,b\}} \{ \sum_{i \in \{all\ texels\ in\ a\ block\}} S_t^{(i)} \}$$

Luminance Local Dynamic Range Reduction

- Extract the block upper and lower bounds



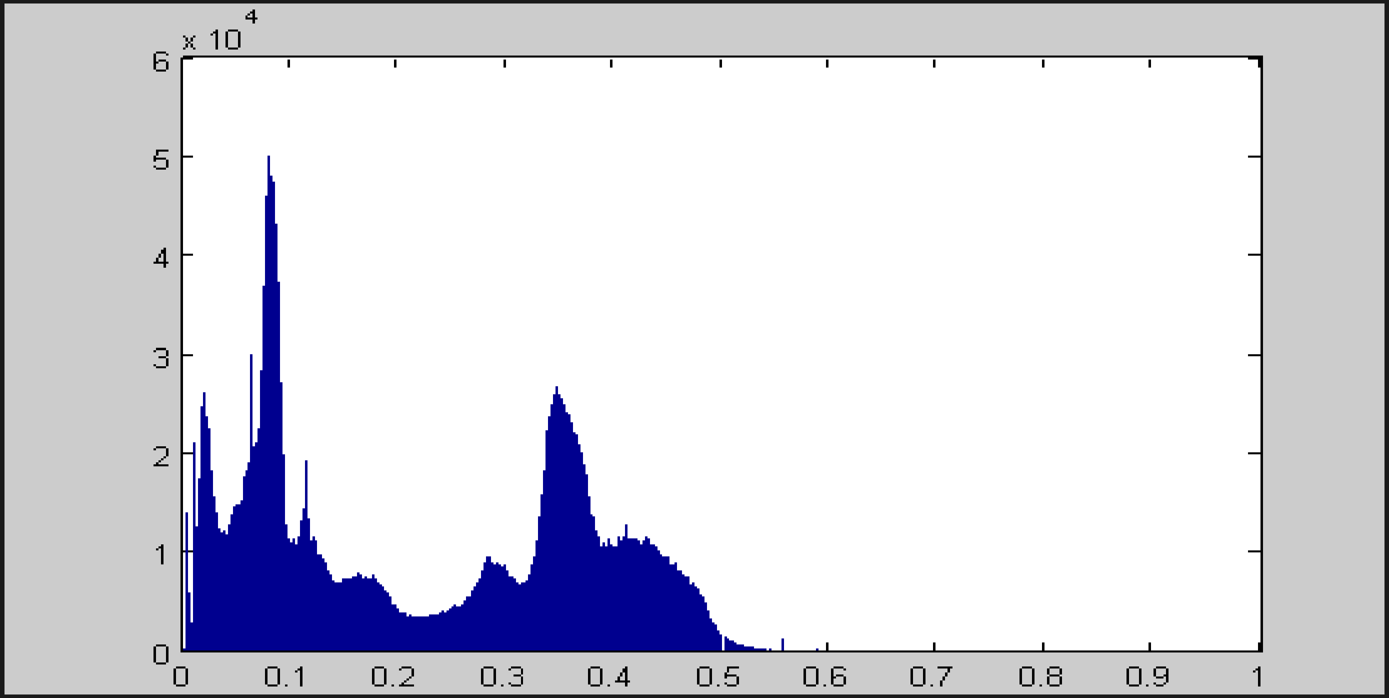
Extract the block upper & lower bounds
(Global Base Luma – L0, L1)

Mapping Global HDR to Local LDR
(Floating Point -> integer)

Chrominance Local Dynamic Range Reduction

- Adaptive color transform significantly reduces the chrominance dynamic range

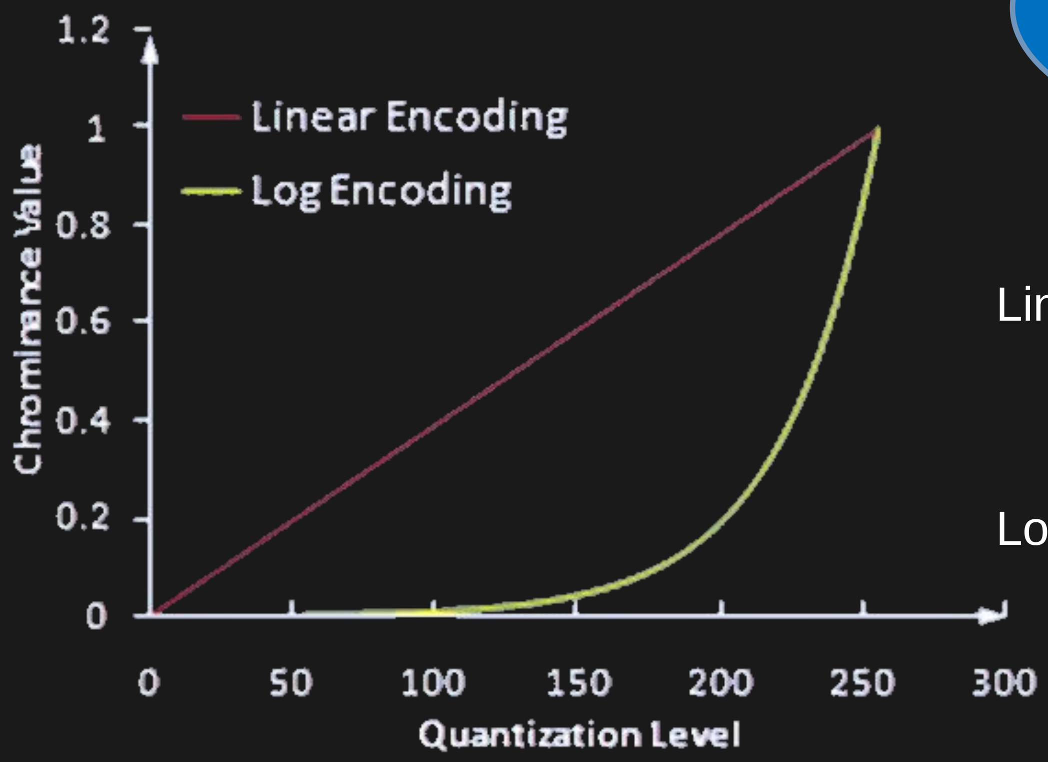
Chrominance value distribution in [0, 1]



Chrominance Local Dynamic Range Reduction

- Proposed adaptive log/linear encoding

Use a 1-bit flag in each texture block to adaptively select the encoding mode

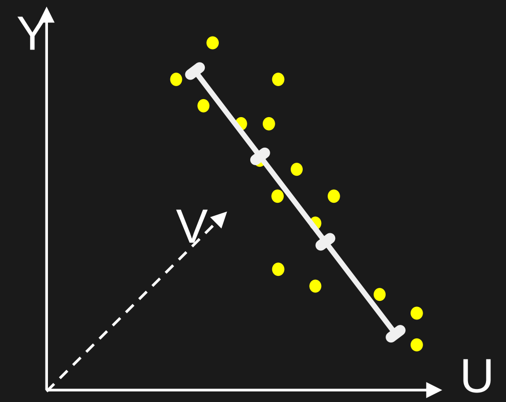


Linear encoding is better

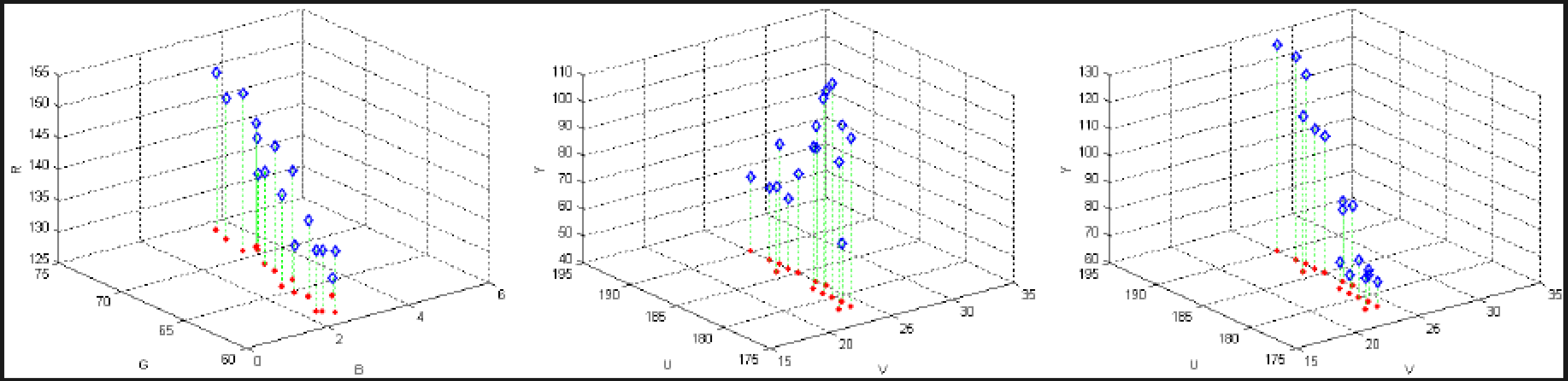
Log encoding is better

Joint Channel Compression & Point Translation

- Basic idea – Joint channel linear fitting
  - Simple and hardware friendly
  - But rely on texel distribution

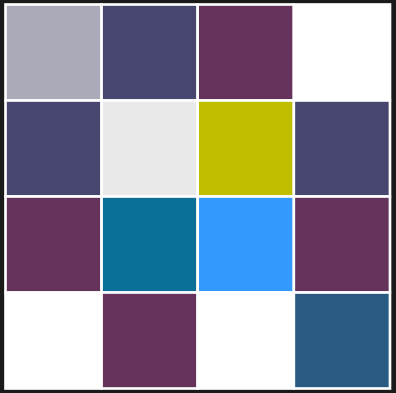


- Point translation
  - Translate texels along Y axis to reshape the distribution

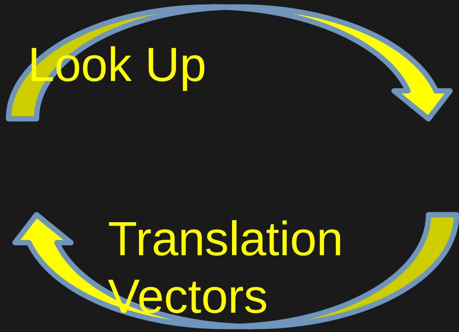


Point Translation

- We use a constant modifier table to provide translation vector for each texel



One HDR texture block

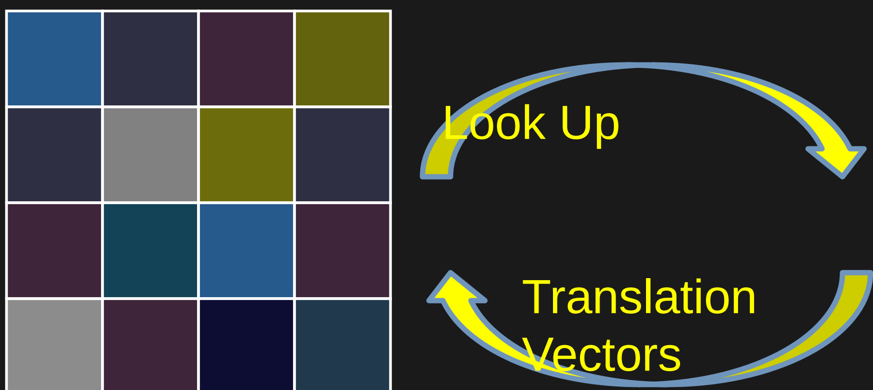


M_idx \ T_idx	0.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	13.	14.	15.
0.	1	1	1	1	2	2	2	2	4	4	4	4	8	8	8	8
1.	-1	-1	-1	-1	-2	-2	-2	-2	-4	-4	-4	-4	-8	-8	-8	-8
2.	2	3	4	5	4	6	8	10	8	12	16	20	16	24	32	40
3.	-2	-3	-4	-5	-4	-6	-8	-10	-8	-12	-16	-20	-16	-24	-32	-40
4.	3	5	7	9	6	10	14	18	12	20	28	36	24	40	56	72
5.	-3	-5	-7	-9	-6	-10	-14	-18	-12	-20	-28	-36	-24	-40	-56	-72
6.	4	7	10	13	8	14	20	26	16	28	40	52	32	56	80	104
7.	-4	-7	-10	-13	-8	-14	-20	-26	-16	-28	-40	-52	-32	-56	-80	-104

Modifier table

Point Translation

- We use a constant modifier table to provide translation vector for each texel



One HDR texture block

M_idx \ T_idx	0.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	13.	14.	15.
0.	1	1	1	1	2	2	2	2	4	4	4	4	8	8	8	8
1.	-1	-1	-1	-1	-2	-2	-2	-2	-4	-4	-4	-4	-8	-8	-8	-8
2.	2	3	4	5	4	6	8	10	8	12	16	20	16	24	32	40
3.	-2	-3	-4	-5	-4	-6	-8	-10	-8	-12	-16	-20	-16	-24	-32	-40
4.	3	5	7	9	6	10	14	18	12	20	28	36	24	40	56	72
5.	-3	-5	-7	-9	-6	-10	-14	-18	-12	-20	-28	-36	-24	-40	-56	-72
6.	4	7	10	13	8	14	20	26	16	28	40	52	32	56	80	104
7.	-4	-7	-10	-13	-8	-14	-20	-26	-16	-28	-40	-52	-32	-56	-80	-104

Modifier table

DHTC Format

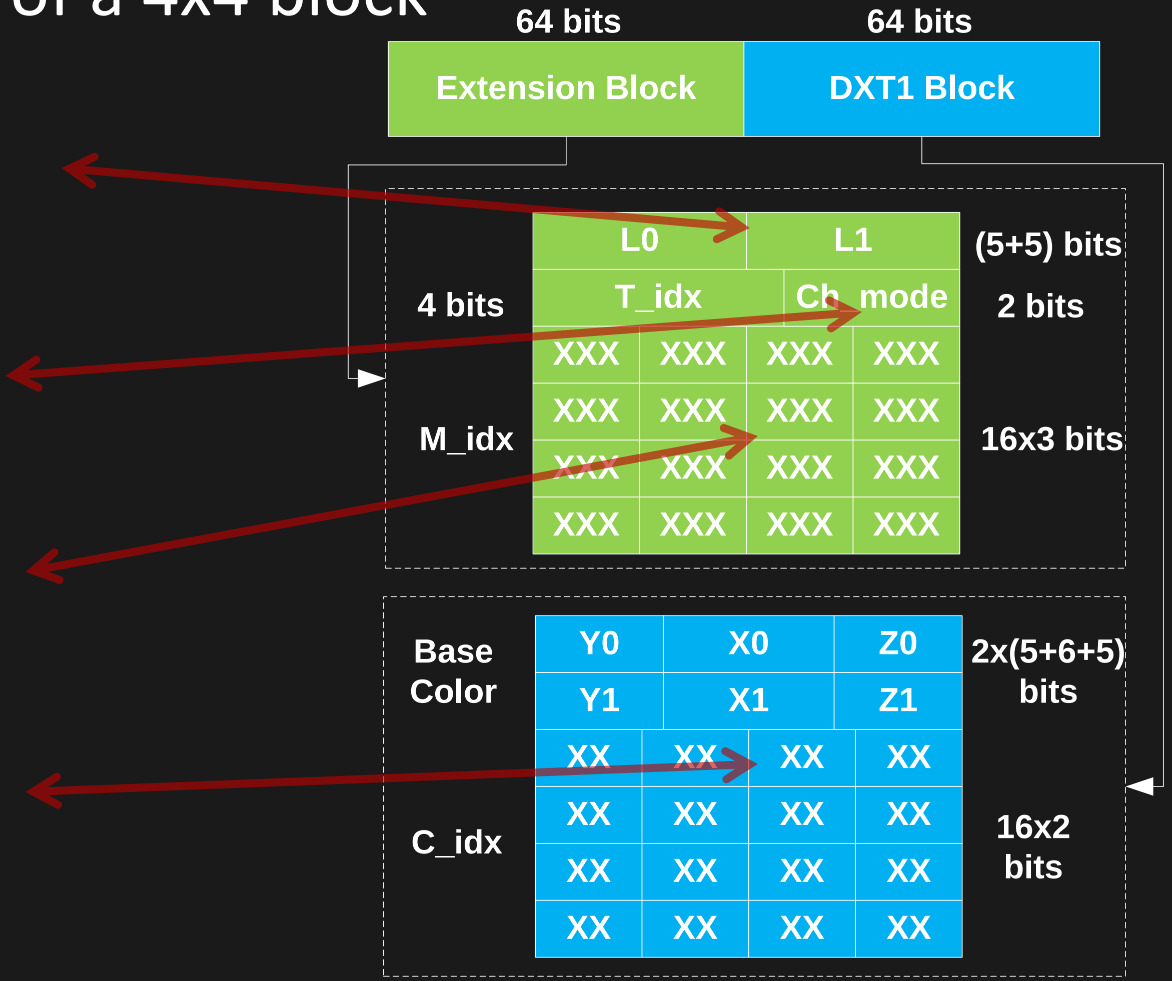
- Bits layout of a 4x4 block

Local dynamic range reduction

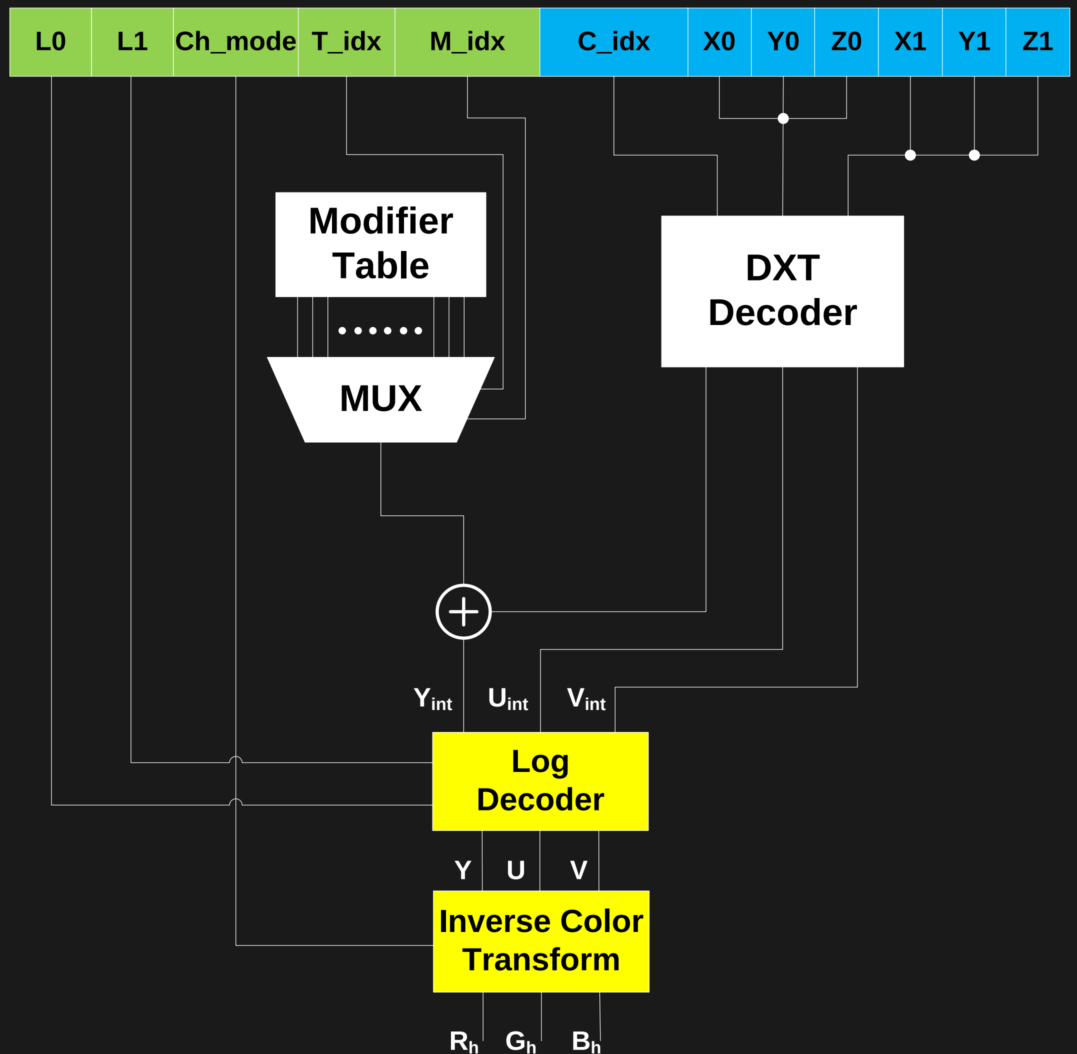
Adaptive color transform

Point translation

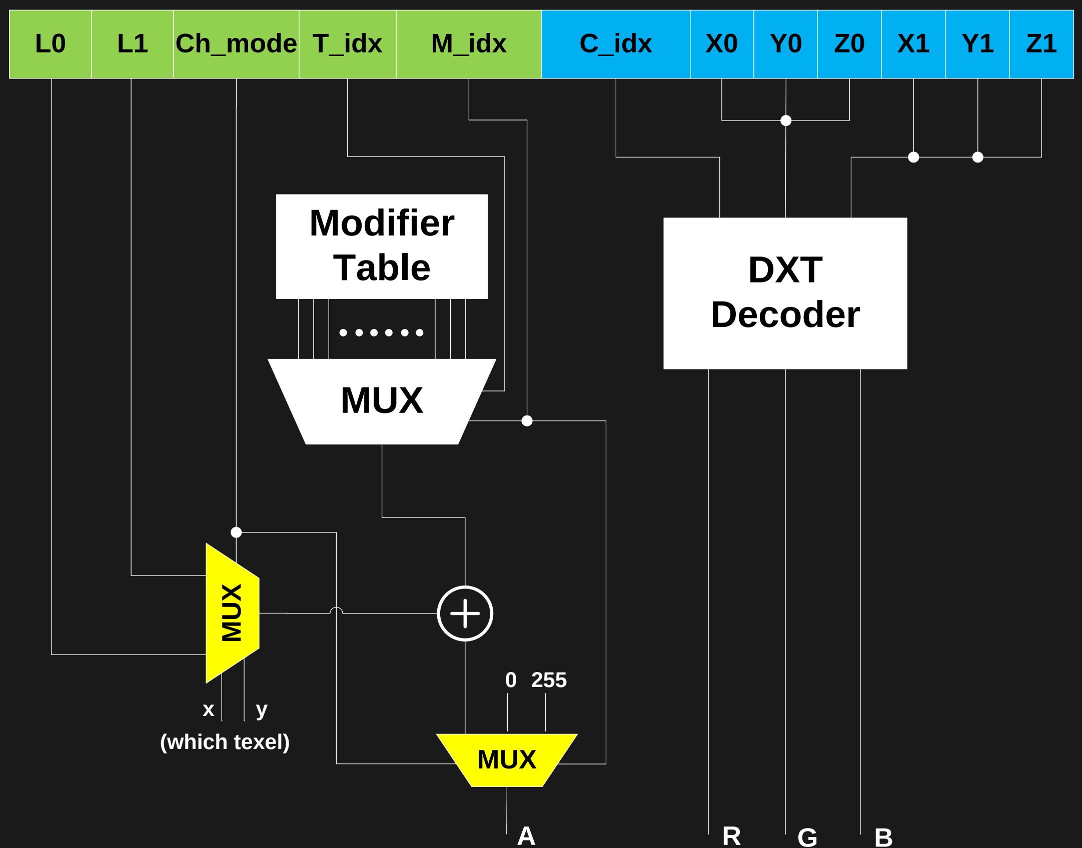
Joint channel compression



Decoding Logic for HDR Textures

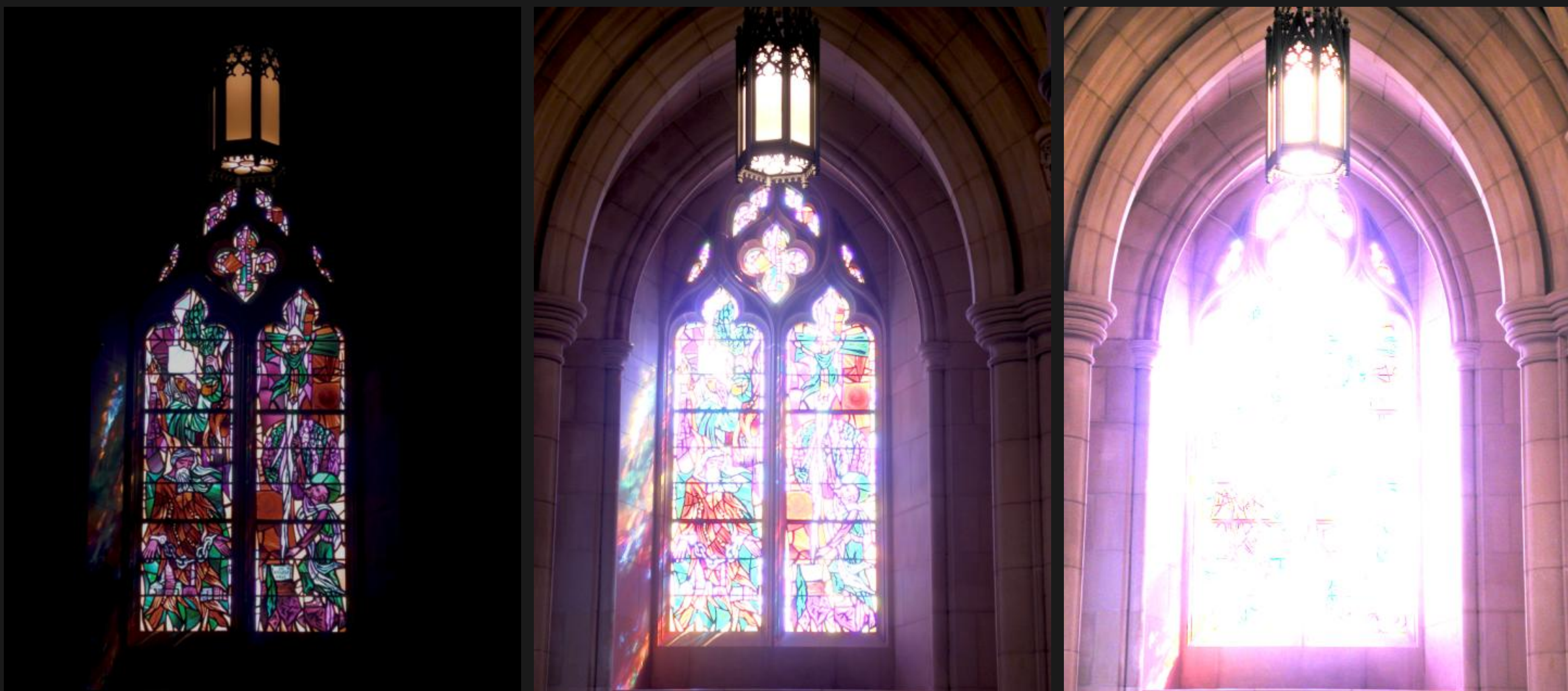


Decoding Logic for LDR Textures



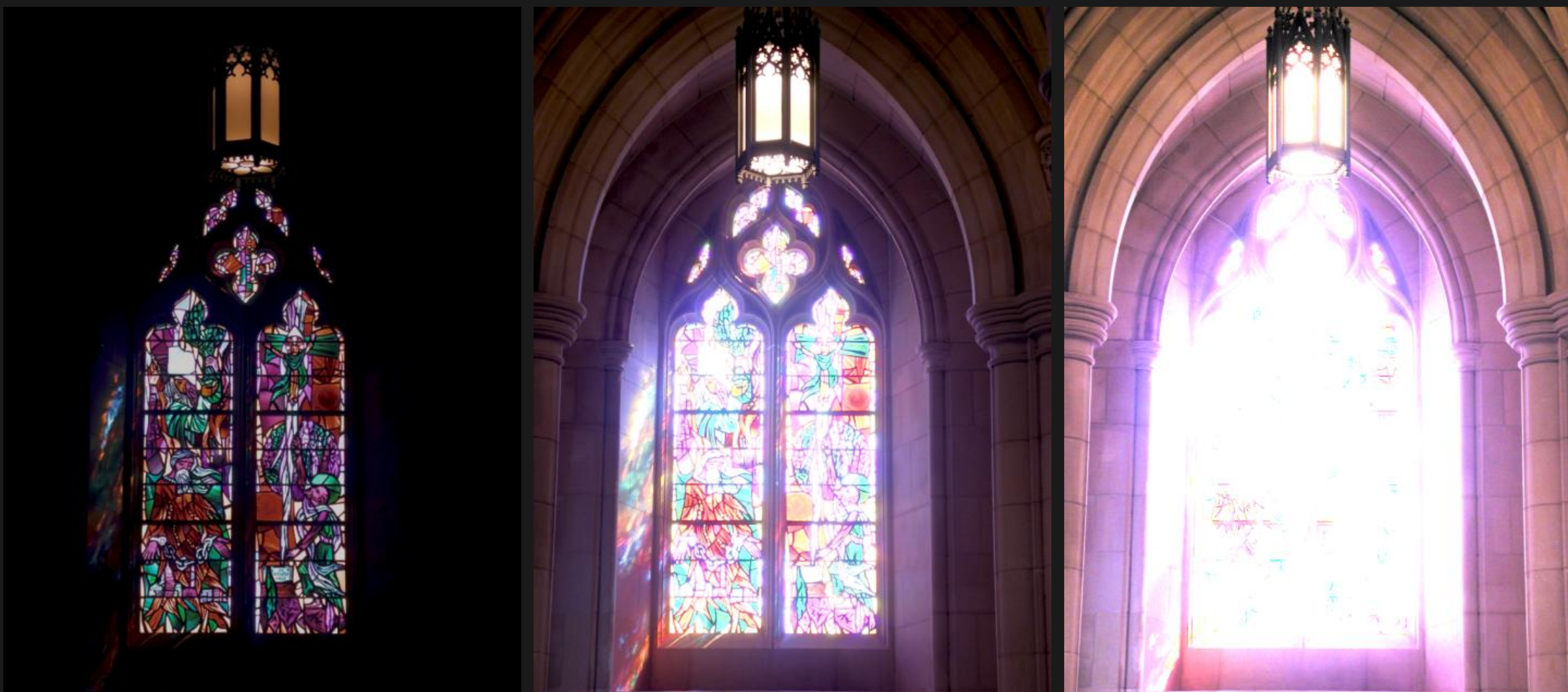
Results

- Original HDR texture at different exposures



Results

- DHTC compressed at 8 bpp



Results

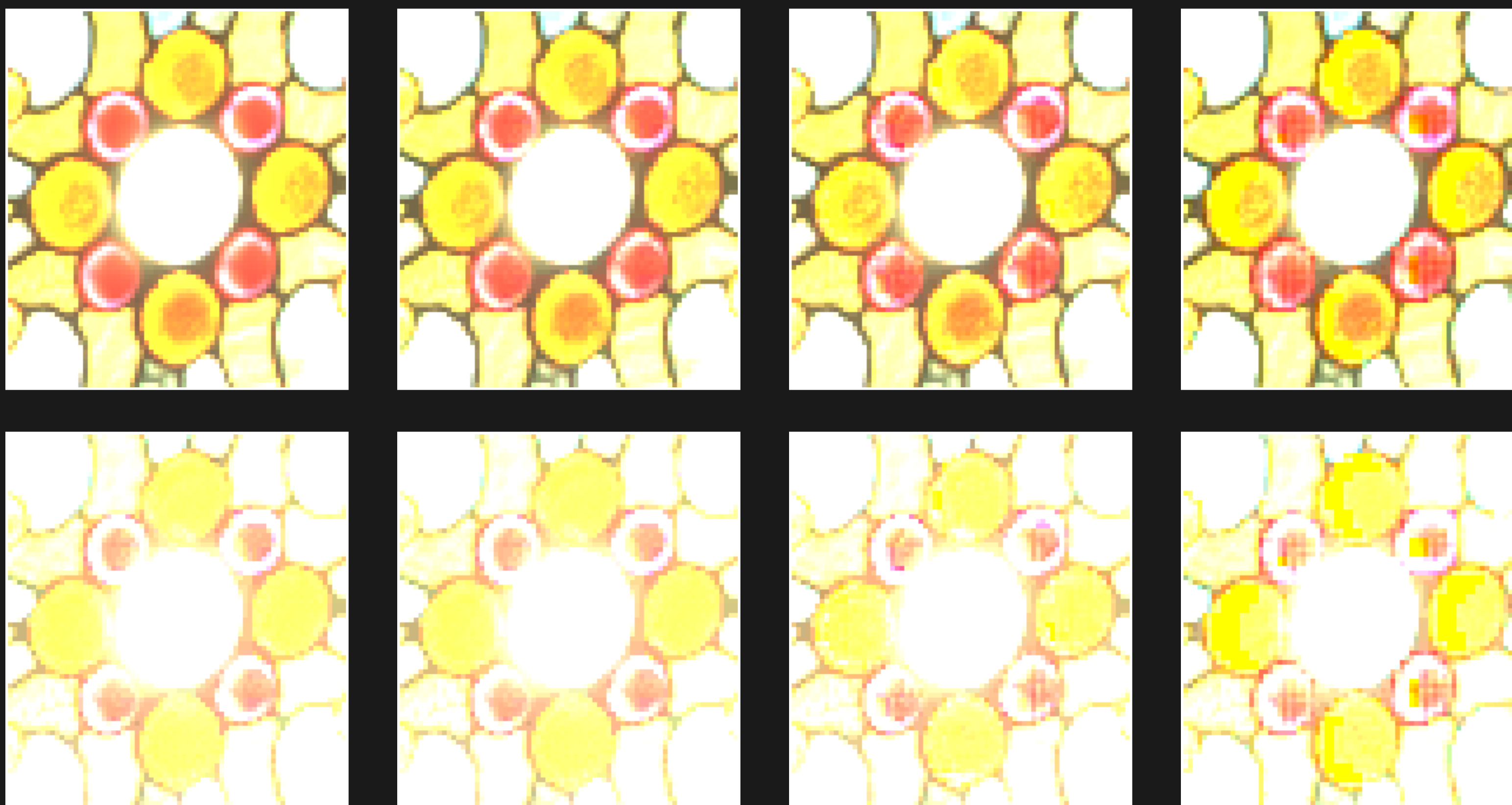
- Original HDR texture at different exposures

Results

- DHTC compressed at 8 bpp

Results

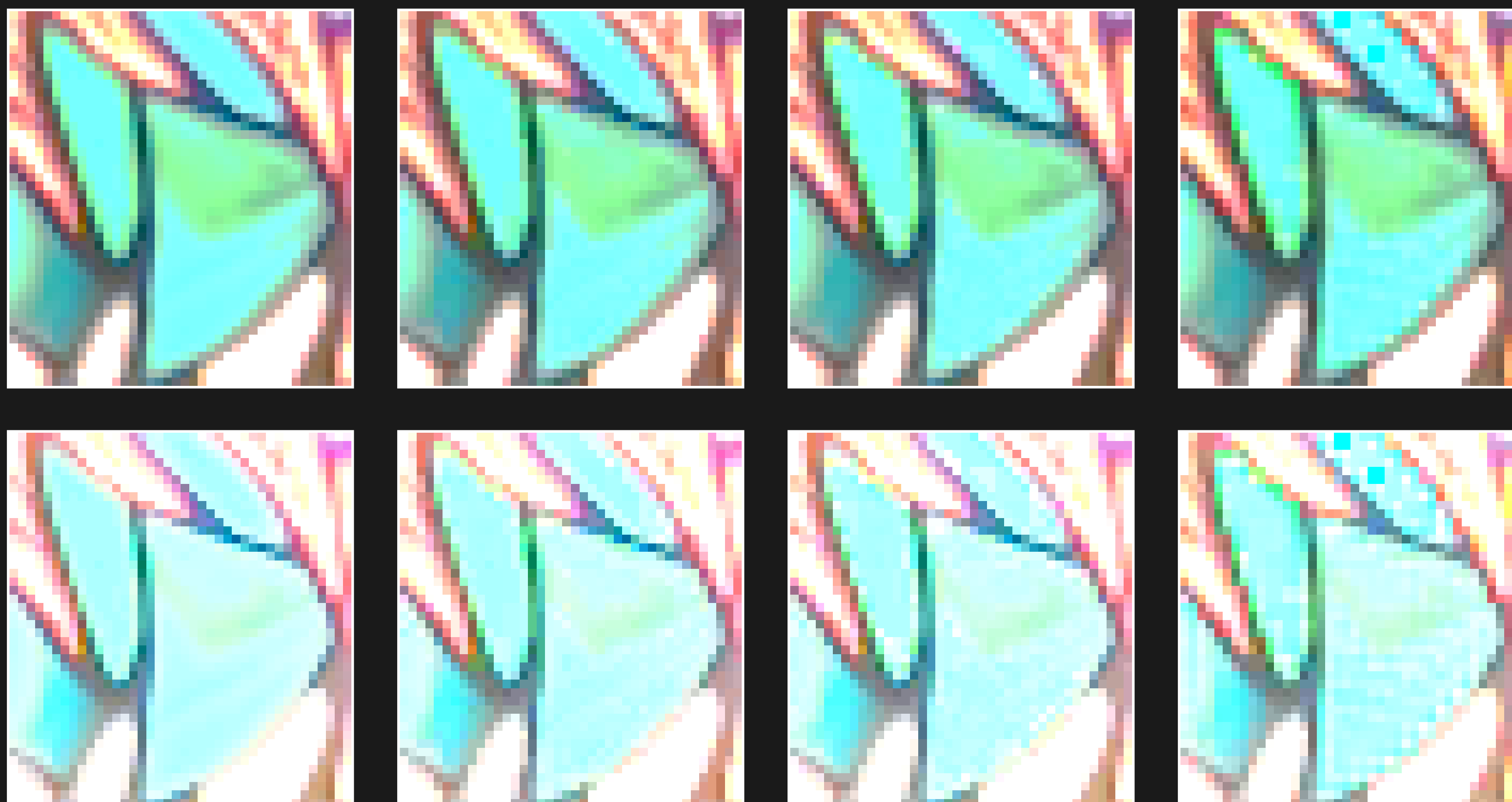
- Visual comparison with the state-of-the-art



Original | DHTC | Munkberg et al 2007 | Roimela et al 2008

Results

- Visual comparison with the state-of-the-art



Original | DHTC | Munkberg et al 2007 | Roimela et al 2008

Results

- mPSNR (dB)

Textures	DHTC	Munkberg 2007	Roimela 2008
BigFogMap	51.0	51.9	50.4
Cathedral	39.7	40.0	34.3
Memorial	46.8	46.5	41.7
Room	48.1	48.6	44.0
Desk	41.5	40.3	28.4
Tubes	35.7	35.7	27.0
Average	**43.8**	**43.8**	**37.6**

Results

- Log[RGB] RMSE

Textures	DHTC	Munkberg 2007	Roimela 2008
BigFogMap	0.06	0.06	0.07
Cathedral	0.17	0.17	0.35
Memorial	0.14	0.13	0.31
Room	0.09	0.08	0.15
Desk	0.17	0.22	1.26
Tubes	0.32	0.28	0.81
Average	**0.16**	**0.16**	**0.49**

Results

- HDR-VDP above 75% error (%)

Textures	DHTC	Munkberg 2007	Roimela 2008
BigFogMap	0.00	0.00	0.00
Cathedral	0.10	0.02	0.03
Memorial	0.01	0.00	0.00
Room	0.01	0.01	0.00
Desk	0.03	0.01	0.00
Tubes	0.87	1.25	1.20
Average	**0.17**	**0.22**	**0.21**

Results

- Rendered scenes

Original

DHTC, 50dB

Results

- Rendered scenes

Original

DHTC, 50dB

Results

- Rendered scenes

Original

DHTC, 53dB

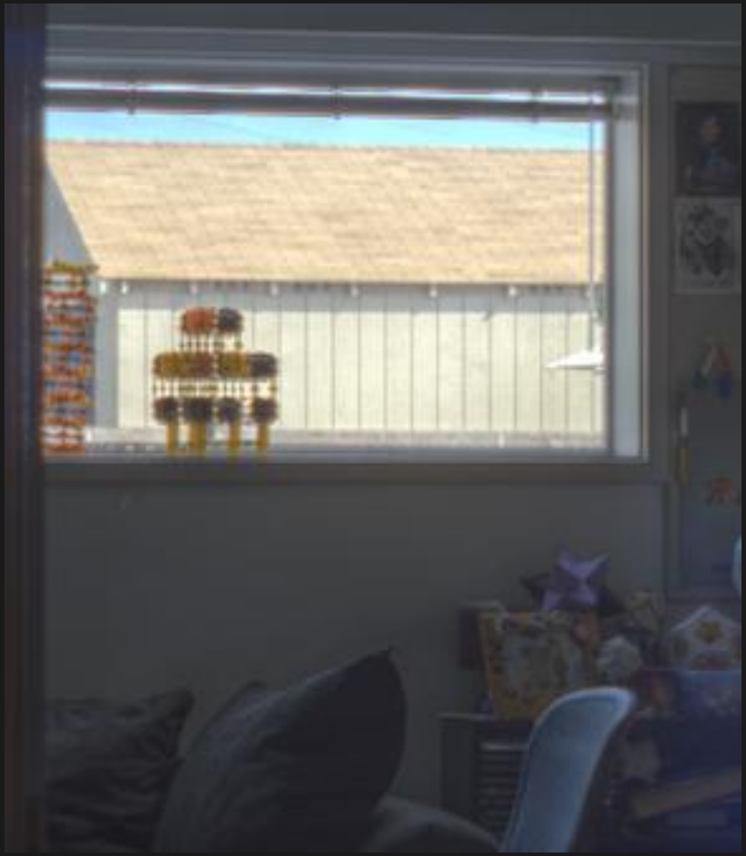
Results

- 1-bit HDR alpha blending

DHTC compressed HDR texture with 1-bit alpha channel

Results

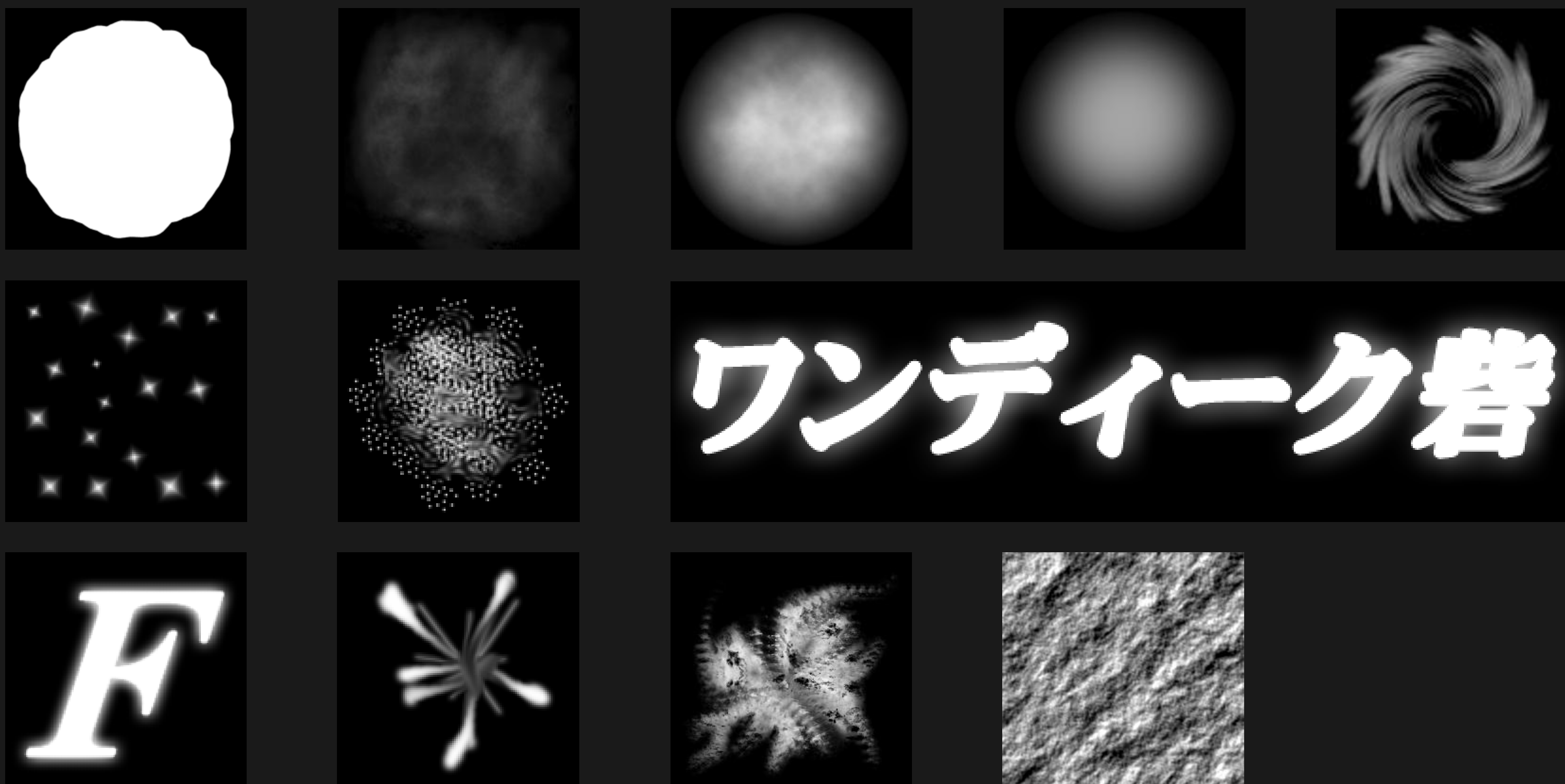
- 1-bit HDR alpha blending



Alpha blended scenes

Results

- Alpha coding for LDR textures
  - We test 12 alpha maps from textures used in an Xbox game



Results

- Alpha coding for LDR textures

Our method provides better results in most cases than DXT5 alpha coding

PSNR(dB)	DHTC	DXT5
AlphaMap1	49.2	48.6
AlphaMap2	53.6	54. 8
AlphaMap3	54.4	57.1
AlphaMap4	56.4	61.2
AlphaMap5	46.3	45.3
AlphaMap6	47.5	45.2
AlphaMap7	40.2	39.0
AlphaMap8	47.1	46.1
AlphaMap9	47.3	44.2
AlphaMap10	44.9	42.9
AlphaMap11	45.0	43.8
AlphaMap12	41.9	40.2
Average	**47.8**	**47.4**

Results

- Alpha coding for LDR textures
  - Less block artifact in our method

DHTC

DXT5

Summary

- DXTC-based HDR texture compression
  - Compress HDR textures into 8 bpp with the best quality so far
  - Utilize the existing DXTC decoding hardware to minimize the adoption cost
  - Provide a unified solution to compress HDR textures, LDR textures and alpha maps

Acknowledgements

- Kimmo Roimela of Nokia Research Center and Jacob Munkberg of Lund University for providing their testing results for comparison.
- Xin Tong, Liyi Wei, Baining Guo, John Tardif, Matt Bronder, Andrew Flavell of Microsoft for the valuable comments and suggestions.
- John Owens of Univ. of California, Davis, for the helps in improving the paper quality.

Thank You!