High Quality Real-Time DXT Compression of Texture Images for Computer Graphics

Fei Yue
Background

• Textures are essential for today’s computer graphics
• Textures can take up a lot of space
• A good compression algorithm can improve storage and performance
• A good real-time algorithm is important for dynamic texture generation
DXT

• A common format across most major GPUs
• DXT1
  – 16 input pixels, 64 bits output
  – Compression ratio: 6:1 for 24 bit source images
• DXT5
  – 16 input pixels, 128 bits output: 64 bits alpha, 64 bits color
  – Compression ratio: 4:1
Real-time YCoCg-DXT5 [1]

1. RGB Input
2. YCoCg Conversion
3. Find Max/Min
4. Find Chroma Quantization Line
5. Quantization
6. Encode
7. Decode
8. RGB Output

Our Method

1. RGB Input
2. YCoCg Conversion
3. Chroma Remap
4. Micro Sharpening
5. Noise Dithering
6. Find Max/Min
7. Find Chroma Quantization Line
8. Quantization
9. Encode
10. Decode
11. RGB Output
Non-linear Remap

• Chroma’ = \sqrt{\text{abs}(\text{Chroma})} \times \text{sign}(\text{Chroma})
• Expand near zero region
• Compress near saturation region
• The eye is more sensitive to chroma differences in less saturated colors
Micro-Sharpening

- Detail preservation
- Adding a scaled amount of the difference between the pixel and a low pass filter applied to the pixel
- Only done to the luma (Y) channel, limited to one Y quantization level (-/+ 1.0/512.0)
Noise Dithering

• Add noise to Y, Co and Cg
• Reduce banding and blocking
Results
High Quality
High Quality
Original
Modified Original
DXT5
High Quality