Video Compression - Hw3

D11921B15 程煒倫

3. Write a program to convert among SDR, HDR PQ10 and HLG. Compare the color if no conversion is applied.

1. Format Conversion:

When converting among SDR, HDR PQ10, and HLG, their differences in dynamic range, color gamut, and brightness representation must be considered.

Commonalities

- **Color Space**: SDR, HDR PQ10, and HLG all use color spaces to represent colors. However, HDR formats (including PQ10 and HLG) support a broader color gamut and higher brightness range than SDR.
- **Brightness Range**: All formats are used to represent a range of brightness from darkest to brightest. SDR has a narrower range, whereas HDR (including PQ10 and HLG) provides a broader range of brightness and more detailed levels.

Conversion Principles

- **SDR to HDR (PQ10 or HLG):**
  - Involves expanding the limited dynamic range of SDR to the broader range of HDR.
  - Requires color mapping and brightness expansion to simulate the richer colors and brightness details in HDR.
- **HDR (PQ10 or HLG) to SDR:**
  - Needs tone mapping to reduce the dynamic range of HDR content to fit within the brightness range of SDR.
  - This process may result in the loss of some details in highlights and shadows.
- **Conversion Between HDR PQ10 and HLG:**
  - PQ10 and HLG use different methods to encode brightness information. PQ10 is based on a perceptual quantization curve, while HLG is designed to be compatible with both SDR and HDR.

- The conversion involves remapping brightness values to adapt to different encoding methods of each format.

2. Characteristics and Applicable Scenarios of Various Formats:

SDR (Standard Dynamic Range)

- **Characteristics**: Traditional dynamic range used in standard televisions and computer monitors. Limited in color and brightness representation.
- **Applicable Scenarios**: Regular TV programs, standard computers, and mobile displays.

HDR PQ10 (High Dynamic Range with Perceptual Quantization)

- **Characteristics**: Offers a higher range of brightness and richer colors, based on a perceptual quantization curve to more effectively utilize the brightness range.
- **Applicable Scenarios**: High-quality movies, TV shows, especially in 4K and 8K ultra-high-definition content.

HLG (Hybrid Log-Gamma)

- **Characteristics**: Designed for broadcasting, compatible with both SDR and HDR. Performs like SDR in low brightness ranges and offers the advantages of HDR in higher brightness ranges.
- **Applicable Scenarios**: Live broadcasting, suitable for various types of displays, including traditional SDR and modern HDR screens.

3. Key Steps and Principles

- **Tone Mapping (Tone Mapping):**
  - **Purpose**: To compress the wide dynamic range of HDR into the narrower range of SDR.
  - **Method**: Using nonlinear mapping functions to reduce the overall brightness of the image while maintaining important details and contrast. Common methods include Reinhard and Filmic tone mapping.
- **Local Contrast Adjustment:**
  - **Purpose**: To maintain local details and contrast while compressing the dynamic range.
  - **Method**: Analyzing the image locally and treating bright and dark areas differently to maintain local details. This may involve image segmentation or local histogram equalization.

- **Color Correction:**
    - **Purpose**: To adjust colors to fit within the color space of SDR.
    - **Method**: Adjusting saturation and hue based on the differences between the color spaces of SDR and HDR, such as converting BT.2020 to BT.709.
- **Gamma Correction:**
    - **Purpose**: To adjust the brightness curve of the image to suit the characteristics of SDR displays.
    - **Method**: Applying a gamma function to adjust the brightness levels, typically using a gamma value of 2.2 or 2.4.

4. HDR Image Creation and Tone Mapping Comparison

hdr.cpp / hdr.py: Creating and Tone Mapping HDR Images

- **Objective**: To create an HDR image from a series of images with different exposures and apply various tone mapping techniques to generate different styles of SDR (Standard Dynamic Range) images.
- **Steps**:
    - **Read Images and Exposure Times**: Retrieves a set of images and records the exposure time for each.
    - **Image Alignment**: Aligns these images to prevent blurring during HDR composition.
    - **Camera Response Function (CRF) Calculation**: Essential for correctly mapping tones in the HDR composition.
    - **HDR Image Composition**: Combines images with different exposures into a single HDR image.
    - **Apply Different Tone Mapping Methods**: Converts the HDR image into various styles of SDR images using different tone mapping algorithms.
- **Tone Mapping Methods**:
    - **Drago Method**: A logarithmic mapping-based tone mapping algorithm, aimed at maintaining details in the low-lit areas of the image.
    - **Durand Method**: Focuses on edge preservation, aiming to reduce halo effects in the image while maintaining important details.
    - **Reinhard Method**: Balances image contrast while maintaining natural colors and brightness.
    - **Mantiuk Method**: Emphasizes perceptual adjustment of image contrast to maintain consistency with human eye perception in the real world.

Hdr-detailed.py: Converting a Single HDR Image to SDR

- **Objective**: To convert a single HDR image into an SDR image, including tone mapping, local contrast adjustment, color space conversion, and gamma correction.
    - **Key Processes**:
    - **Tone Mapping (Tone Mapping)**: Compresses the high dynamic range of the HDR image to standard dynamic range using the Reinhard method.
    - **Local Contrast Enhancement**: Uses CLAHE (Contrast Limited Adaptive Histogram Equalization) to enhance local contrast in the image.
    - **Color Space Conversion**: Transforms the image from HDR to SDR color space.
    - **Gamma Correction**: Adjusts image brightness to improve overall visual perception.

5. Detailed Steps for HDR Image Creation

Step 1: Capturing Multiple Images at Different Exposures

- **Use of Auto Exposure Bracketing (AEB)**: Captures multiple photos at different exposures, typically using exposure times of 1/30 sec, 0.25 sec, 2.5 sec, and 15 sec.
    - **Types of Images**:
    - **Underexposed Images**: Capture very bright parts of the image.
    - **Correctly Exposed Images**: Regular images taken based on the camera's lighting estimate.
    - **Overexposed Images**: Capture very dark parts of the image.

Step 2: Aligning Images

- **Importance of Alignment**: Unaligned images can cause significant artifacts in HDR composition.
- **Method**: Uses OpenCV's `AlignMTB` for alignment, converting images into median threshold bitmaps (MTB), unaffected by exposure time changes.

Step 3: Extracting Camera Response Function

- **Ease of Extraction**: Only 2 lines of code with OpenCV's CalibrateDebevec or CalibrateRobertson are required to find the CRF.
- **Importance of CRF**: It's crucial to understand the non-linear relationship between the camera's response and real-world brightness. CRF tells us the relationship between pixel values in the photo and actual world brightness.

Without knowing the CRF, it's impossible to accurately merge differently exposed photos into a single HDR image.

- **Challenge and Solution**: Estimating CRF involves mathematical methods to transform pixel values into a linear relationship. This is often a challenge requiring optimization techniques to find the best estimate for CRF. The problem simplifies to a linear least squares problem and can be solved using tools like Singular Value Decomposition (SVD).