Advanced Graphics

Additional Notes Lecture 3

Image Acquisition Pipeline

- **Camera response curve** - a function that defines the relation between the amount of incoming radiance and image pixel intensities produced by a camera.

- Response function is likely to be non-linear.

Image Acquisition Pipeline

Process:

- Light rays travel towards camera lens and the incoming radiance is collected onto the sensor transforming into irradiance.
- Radiance is collected for time period Δt (exposure time) according to shutter speed.
- For *film camera* – irradiance is stored into film format, which has non-linear response to exposure time and needs to be further processed.
- *Electronic camera* – irradiance goes directly to storage in the sensor (e.g. CCD) and produces more linear response.
- CCD-collected data is converted from analogue to digital signal by ADC and mapped to corresponding pixel intensities.

Response Curve Recovery

- From HDR exposure stack choose several pixels to track in each image.
- For each tracked pixel in the image we define $Exposure = Radiance \times \Delta t$.
- Can consider logarithmic scale, which is faster mathematically and has better numerical precision.
- The objective is to record pixel intensities for each exposure and try to fit one curve to explain all changes for selected pixels.

Image Formation Model

- $Z_{ij} = f\left(E_i \Delta t_j\right)$
  - Z_{ij} is the intensity value of i^{th} pixel for j^{th} exposure stack image.
  - E_i is irradiance at pixel i.
  - Δt_j is exposure time of j^{th} image in the stack.
  - f is camera response function, which is likely to be non-linear.

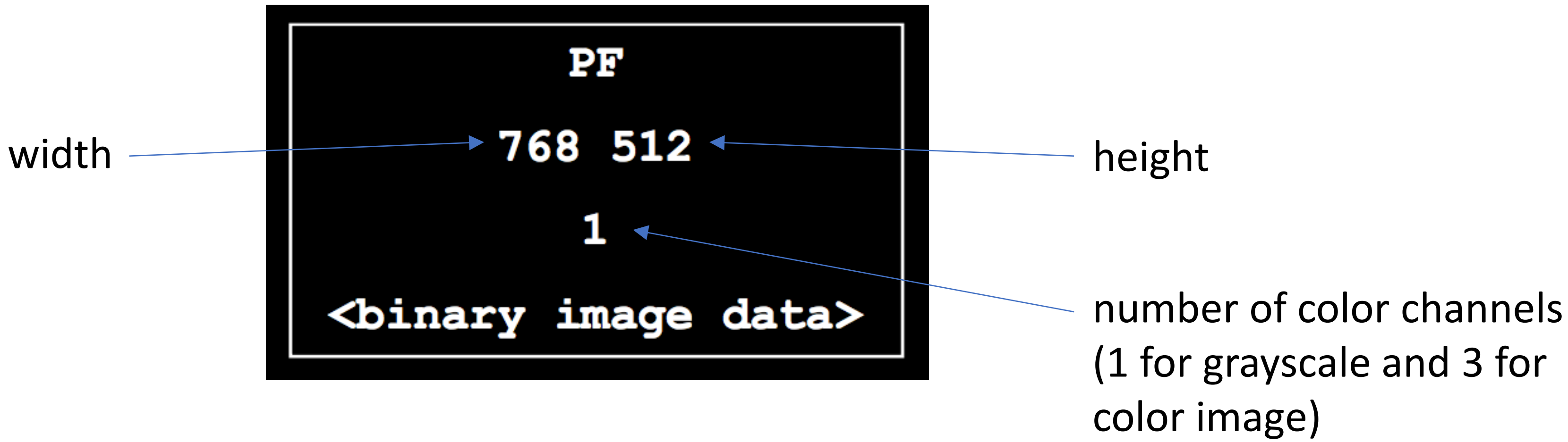
- For HDR images we want to recover E_i for each pixel.
- Hence, invert camera response function to linearize values.
- Finally, merge calculated E_i values for image pixels into one HDR value using weighted averaging (i.e. give higher weights to intensities in the middle rather than to very bright or very dark ones).

HDR Imaging – Motion Blur

- HDR imaging helps to maintain original brightness of the scene for photographs with motion blur.
- Otherwise intensities are smeared over neighboring pixels resulting in overall brightness reduction (e.g. white color in the scene might become greyish in LDR image with motion blur).

PFM Image Format

- Simple to read, *uncompressed* format
- However, *expensive* in terms of storage
- 32 bit (4 bytes) per channel
- For color images we have RGB values, so $4 \times 3 = 12$ bytes per pixel



Radiance Format (RGBE)

- 1 byte for each color channel plus 1 byte *shared exponent*, so 4 bytes per pixel.
- *No sign* bit, so only positive floating point precision storage.
- *Shared exponent*: considering the order of magnitude of image brightness is normally not very different across RGB channels.
- Less expensive than PFM in terms of storage, at the same time preserving HDR.

EXR format

- Highly used in industry for producing visual effects.
- 16 bit (2 bytes) per channel, 6 bytes per pixel.
- Negative values allowed.
- More info: http://www.openexr.com/

Tone Mapping

- Technique that allows to display approximation of HDR images by compressing actual stored HDR data and mapping it to the medium with a more limited dynamic range.
- *Global tone mapping* can be done using *histogram equalization*.
- *Global* means applying same operation over the entire image (automatic operation).

Histogram Equalization

- Many images have clustered pixel histogram, where some of the bins are either not occupied (no pixels in the image with those values) or significantly less occupied compared to other bins.
- Histogram equalization technique artificially stretches the histogram to adjust image contrast.

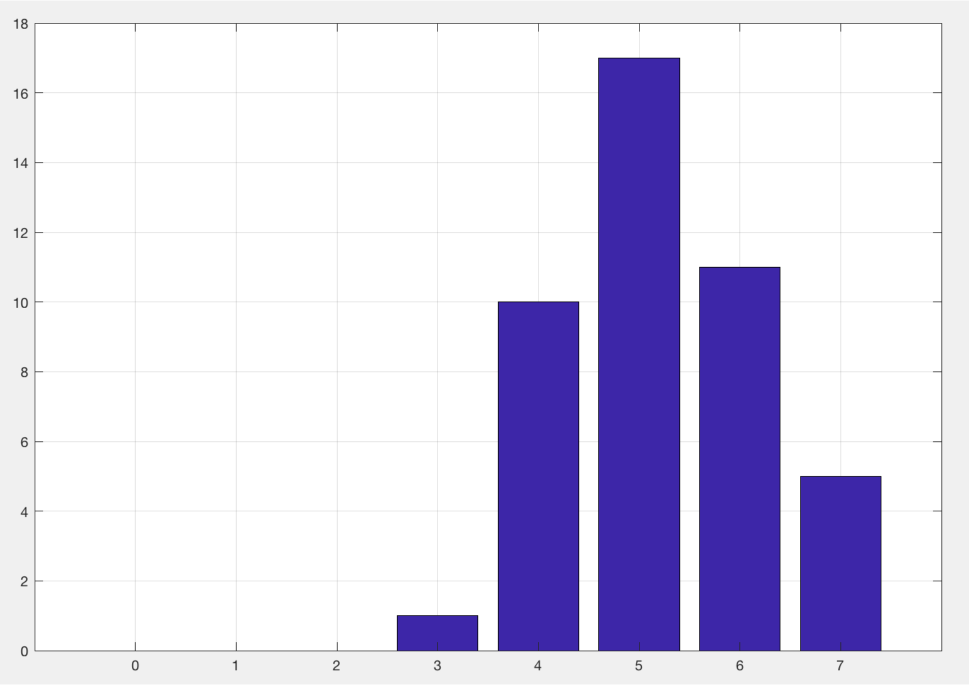
Histogram Equalization

Algorithm:

1) Determine *pixel histogram* of an image.
2) *Normalize* the histogram by dividing frequency values for each intensity by total number of pixels in the image.
3) Calculate *CDF* (Cumulative Distribution Function) of the histogram from step 2.
4) *Multiply* CDF from step 3 by the *maximum intensity* of the image for every bin.
5) *Round* the resulting cumulative sums from step 4 to the nearest integer.
6) *Remap* old pixel intensities to the new ones computed in step 5.

Histogram Equalization

- Step 1:



Intensity	0	1	2	3	4	5	6	7
Frequency	0	0	0	1	10	17	11	5

Histogram Equalization

- Steps 2-5

Max intensity = 7

Intensity	0	1	2	3	4	5	6	7
Frequency	0	0	0	1	10	17	11	5
Normalized freq.	0	0	0	0.023	0.227	0.386	0.25	0.114
CDF	0	0	0	0.023	0.25	0.636	0.886	1
Multiplied CDF	0	0	0	0.161	1.75	4.452	6.202	7
Rounded	0	0	0	0	2	4	6	7

Histogram Equalization

- Step 6

Intensity	0	1	2	3	4	5	6	7
Frequency	0	0	0	1	10	17	11	5
Rounded	0	0	0	0	2	4	6	7

Intensities 0, 1, 2 and 3 are now all remapped to bin with intensity 0

Frequency of new bin 0 is $0 + 0 + 0 + 1 = 1$

Histogram Equalization

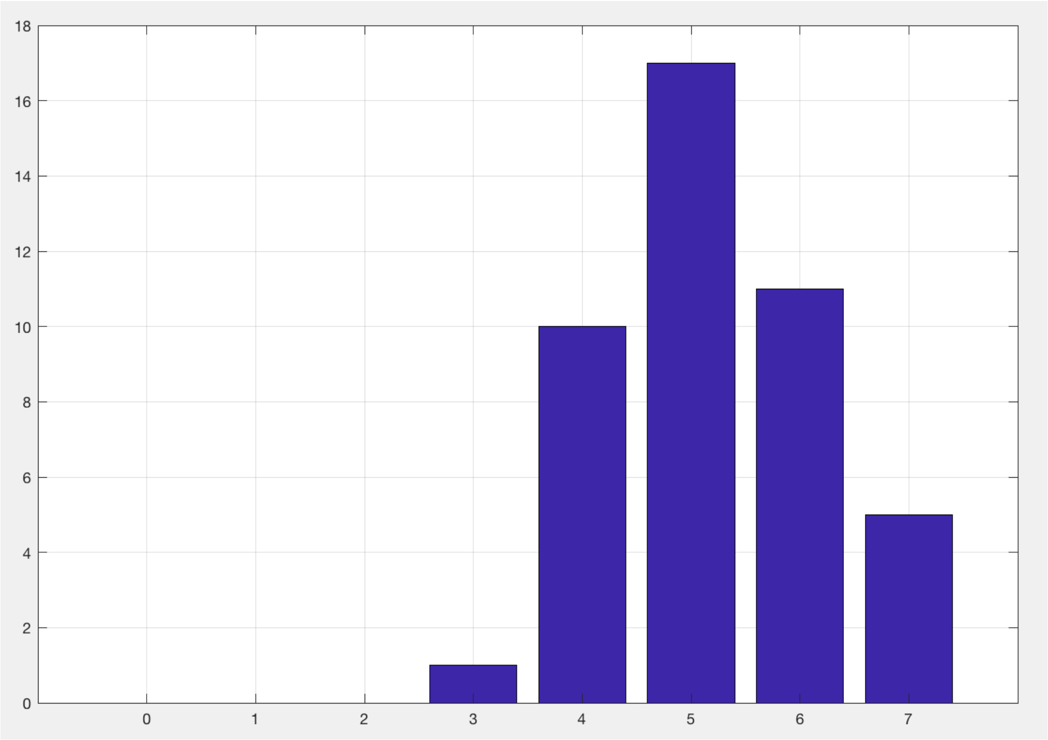
Intensity	0	1	2	3	4	5	6	7
Frequency	0	0	0	1	10	17	11	5
Rounded	0	0	0	0	2	4	6	7

Remapping

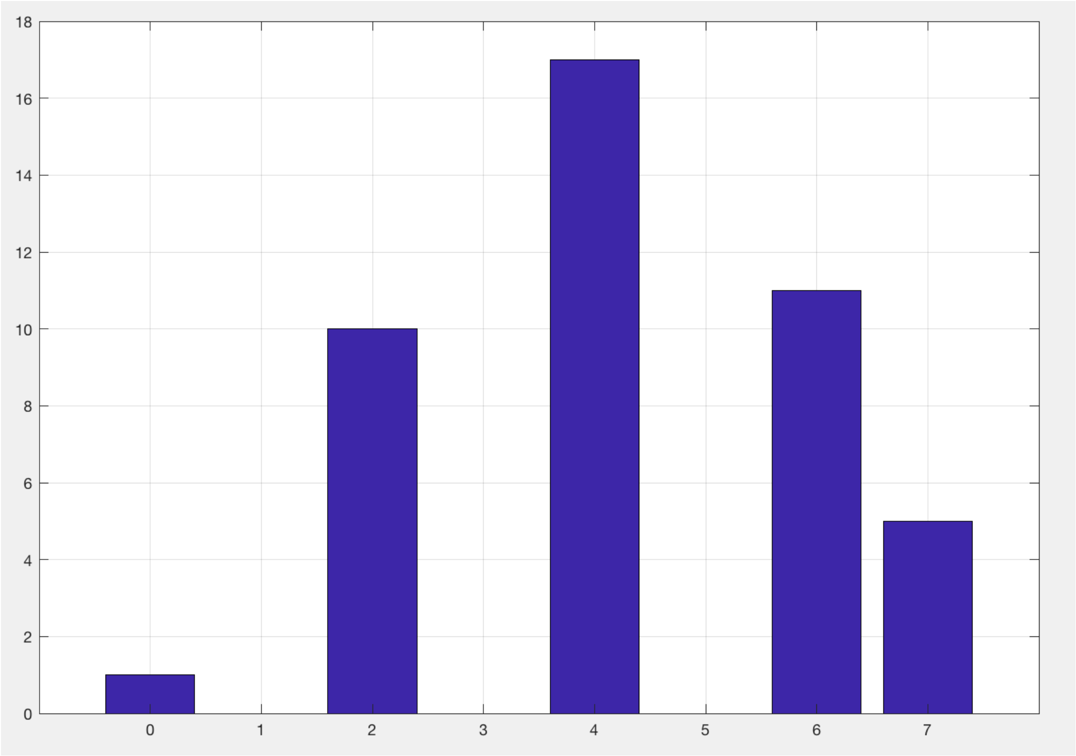
Intensity	0	1	2	3	4	5	6	7
Frequency	1	0	10	0	17	0	11	5

Histogram Equalization

Original histogram



New histogram



Histogram Equalization in HDR Imaging

- For HDR images <u>should not</u> apply separate histogram equalization for red, green and blue channels since color distributions of individual pixels can be significantly different.
- Instead, separate intensity and color first and apply histogram equalization on intensity channel <u>only</u>.
- Finally add back the original color to produce tone mapped image.
- Can do this in HSV space (Hue-Saturation-Value), where H and S hold color information and V holds intensity data.
- Can also use LUV space, where U and V store color data and L is intensity information.

Tone Mapping

- Often tone mapping is a subjective process, which is performed manually by a human.
- Typical manual way to do it is exposure bracketing, i.e. given HDR image subjectively keep scaling values by a factor of 2 until getting desired result.
- Can also change gamma parameter to adjust the contrast.

Local Tone Mapping

- Preserves local contrast and edge details.

Steps:

1. Pre-process, i.e. remove color information to apply operations only to intensity channel.
2. Apply edge-preserving filter to get *base layer*.
3. Subtract filtered image from image derived in step 2 to get *detail layer*.
4. Compress base layer.
5. Combine compressed base layer with detail layer.
6. Add back color information to get final tone-mapped LDR image.