CENG 477
Introduction to Computer Graphics

Images, Cameras, Displays
Images and Displays

• Computer graphics is all about creating and displaying images
• A video is nothing but an image sequence
• A computer game is nothing but many images rendered at rapid succession
• As such, we should first understand what an image is and how it is displayed
Image

• An image, or a digital image, is an intensity distribution over a bounded two dimensional region
• More formally, $I: \mathbb{R}^2 \to ...$
Image

• In practice, we place two restrictions in order to handle images on a computer:
  – Discretization
  – Data type
Discretization

- Discretization is the partitioning of the image region to a non-overlapping grid
- Each cell is called a **pixel** (short for picture element)
Discretization

• The total number of pixels in an image called image resolution
• The same term is used for display devices as well
Data Type

- Data type represents what can be stored in a pixel
  - Bitmaps: $I: \mathbb{R}^2 \rightarrow \{0,1\}$
  - Grayscale: $I: \mathbb{R}^2 \rightarrow \{0,1, \ldots, 255\}$
  - Color: $I: \mathbb{R}^2 \rightarrow \{0,1, \ldots, 255\}^3$
  - Color (generalized): $I: \mathbb{R}^2 \rightarrow \mathbb{R}^3$
Bitmap

- A sample bitmap image (1 bpp):
Grayscale

- A sample grayscale image (8 bpp):
Color

• A sample color image (24 bpp):
Color

- In color images, each pixel has three components: red, green, blue
- Their relative contribution determines the actual color
Color

- Typically, each color value is an 8-bit integer in [0, 255] range
- This corresponding space is called the RGB color cube
Storage

- These colors can be stored as *interleaved* or as *planar* in an image file
Image Formats

- Images generated by a CG program can be sent to a display device or written to an image file
Image Formats

• There are hundreds of image formats perhaps the best well-known being the JPEG format
• A very simple image format is the PPM format:

```
P3
# simple.ppm
4 4
255
128 128 128 128 128 128 128 255 0 255
128 128 128 0 255 255 128 128 128 128 128
255 0 0 128 128 128 128 128 128 128 128
0 0 255 128 128 128 128 128 128 128 128 128
```
Image Formats

• Despite being simple, the PPM format is very inefficient due to lack of compression
• How many MBs would an 18 megapixel (MP) plain PPM image would occupy?

\[
\frac{18 \times 1024 \times 1024 \times 3 \times 3.57}{1024 \times 1024} \approx 193 \text{ MBs}
\]
Image Formats

• 3.57 in this formula comes from the expected number of bytes that each component will occupy:

\[
\frac{10 \times 1 + 90 \times 2 + 156 \times 3}{256} + 1 \approx 3.57
\]

• The last +1 is for the whitespace between the components
Image Formats

• If we used **binary PPM** (each component is 1 byte and there is no white space) this produce a file size of 54 MBS for an 18 MP image (still too much)

• For this reason, compressed image formats are available
  – Lossless compression
  – Lossy compression
Lossless Compression

• With lossless compression, a decoder will read the exact information that an encoder wrote to a file
• Various lossless compression techniques are used such as Huffman encoding, run-length encoding, etc.
• A well-known lossless image format is the PNG format
• Efficient for computer generated images but not for natural (photographic) images due to presence of noise
Lossy Compression

- In **lossy compression**, numerical match between the input and output is not required.
- Some information is lost to improve compression efficiency.
- However, lossy formats can still be **visually lossless**.
- The most well-known lossy format is the **JPEG** format.
Lossy vs Lossless

- Assume we encode and decode this data as PPM and JPEG:

```
P3
# simple.ppm
4 4
255
128 128 128 128 128 128 128 255 0 255
128 128 128 0 255 255 128 128 128 128 128 128
255 0 0 128 128 128 128 128 128 128 128 128
0 0 255 128 128 128 128 128 128 128 128 128
```

PPM (or PNG)  JPEG
JPEG Quality

- A JPEG image with different quality settings:
Image Capture

• We defined a color image as $I: \mathbb{R}^2 \rightarrow \{0,1, ..., 255\}^3$
• The light intensity (more precisely *luminance*) in the world is not restricted to such a set
• Larger and smaller values are clamped (saturated) and all values are quantized
• Which values get saturated depends on the exposure setting of the camera
Image Capture

• A low exposure image:
Image Capture

• A high exposure image:
Dynamic Range

- Dynamic range (DR) is defined as the ratio of the highest luminance to the lowest luminance in a given scene.

\[ DR = \frac{L_{\text{max}}}{L_{\text{min}}} \]
Exposure Control

- Cameras control their exposure to decide on a proper range:
Quantization

• With quantization, world luminance to pixel value mapping looks like this:
Rendering

• Why this matters for CG?
• In CG, we generate images just like a camera captures images
• Our artificial scene may have luminance values not limited to [0, 255] range
• As such, we have to mimic what the camera does
What if we want to capture (or render) the world as it is without mapping to $[0, 255]$ range?

We can capture multiple exposures and merge them to create an HDR image.
HDR

• HDR image capture process:

1/1000 s.  1/500 s.  1/250 s.

1/125 s.  1/60 s.  1/30 s.

Merge  HDR Image
HDR

- Unfortunately, HDR images cannot be directly displayed on standard display devices
- They must be tonemapped first (see Chapter 23 in our textbook):
HDR

- HDR images can also be saved in HDR file formats such as .hdr and .exr
- Novel HDR displays allow direct display of HDR imagery:
HDR

- HDR images can also be saved in HDR file formats such as .hdr and .exr.
- Novel HDR displays allow direct display of HDR imagery:

See [http://hdr.sim2.it/](http://hdr.sim2.it/) for more info on HDR displays.
Display Devices

- Generated images are sent to a display device via the video card
- Therefore, understanding the basic properties of display devices is important
Display Devices

- CRT
- Plasma
- LCD
- LED
- OLED
- E-ink
- ...

- There are numerous types of display devices
- But they all share some basic properties
Display Devices

- The **digital** RGB signals sent from the video card are translated into **analog** voltages in the display device.
- These voltages determine the luminance emitted from each pixel.
- However, voltage to luminance relationship is **nonlinear**.
Gamma

• This nonlinearity is called the display gamma
  – E.g. twice the pixel value (twice the voltage) does not result in twice the luminance
• Also, zero RGB value does not mean zero luminance due to leaking light and reflections off the screen
• A simplified display model is:

\[ L_{out} = cV^\gamma + b \]

- Output luminance
- Contrast
- Voltage
- Gamma
- Brightness
Gamma

- Measured gamma of a NEC Spectraview 241 monitor calibrated to the sRGB profile:
  - Measurements are taken at intervals of 10
  - Ideal is a gamma value of 2.2
Gamma Correction

• Most display devices have a gamma value around 2.2
• **Gamma correction** is performed to account for this non-linearity:
  – The input signal is raised to the power $1/\gamma$ before being stored
  – Typically cameras do this for us unless capturing in RAW format
• This makes the luminance emitted by a display device linear
Gamma Correction

Gamma characteristics of monitors

Color information adjusted to match gamma characteristics

Color handling approaching the “y = x” idealcs
Gamma Correction

- In CG, we must also apply gamma correction as the last step of the rendering process.
- Without gamma correction, we may obtain unnaturally dark images.
Gamma Correction

- In CG, we must also apply gamma correction as the last step of the rendering process.
- Without gamma correction, we may obtain unnaturally dark images.