Computer Graphics

Master in Computer Science
Master in Electrical Engineering

...
Computer Graphics

The service...

Eric Béchet (it's me !)

- Engineering Studies in Nancy (Fr.)
- Ph.D. in Montréal (Can.)
- Academic career in Nantes and Metz (Fr.) then Liège...

Christophe Leblanc

- Assistant at ULg

Web site

http://www.cgeo.ulg.ac.be/infographie

Emails: { eric.bechet, christophe.leblanc }@uliege.be
Computer Graphics

Course schedule

- 6-7 theory lessons (≈ 4 hours)
  - may be split in 2x2 hours and mixed with labs
  - This room in building B52 (+2/441) (or my office)
- 7 practice lessons on computer (≈ 4 hours)
  - may be split in 2x2 hours and mixed with lessons
  - Room +0/413 (B52, floor 0) or +2/441 (own PC)
- Practical evaluation (labs / on computer)
- Written exam about theory
- Project
- Availability time: Monday PM (or on appointment)
Computer Graphics

Course schedule

- **Project**
  - Implementation of realistic rendering techniques in a small in-house ray-tracing software
  - Your own topics
Computer Graphics

Course schedule

- The lectures begin on February 6 and:
  - Feb 27, March 13, etc.
    (up-to-date agenda on the website)
- The labs begin on February 20th.
  - Alternating with the theoretical courses.
Computer graphics: The study of the creation, manipulation and the use of images in computers
Some bibliography:

- Computer graphics: principles and practice in C
  James Foley et al.
  Old but there exists a french translation

- Computer graphics: Theory into practice
  Jeffrey McConnell

- Fundamentals of computer graphics
  Peter Shirley et al.

- Algorithmes pour la synthèse d'images (in French)
  R. Malgouyres
Uses of Computer Graphics

- Leisure
  - Games, Special effects
  - Animation film
  - Computer game
- Sciences and technology
  - Computer Aided Design
  - Scientific Visualization
- Simulators (flight, etc...) / virtual reality
- Graphics (Photoshop, Illustrator...)
- Arts
Computer Graphics
Applications

Toy Story - Pixar effects (Renderman)
Ratatouille - Pixar effects (Renderman)
King Kong - WETA Digital effects
Computer Graphics
Applications

Lord of the Rings - WETA Digital
Avatar - WETA Digital / ILM
Quake – 3D « real time » rendering
Uses of Computer Graphics

- Leisure
  - Games, Special effects
  - Animation film
  - Computer game

- Sciences and technology
  - Computer Aided Design
  - Scientific visualization

- Simulators (flight, etc...) / virtual reality
- Graphics (photoshop, illustrator...)
- Arts
Expanded view of a printer head
Schematic view of a protein (myoglobin)
Crack in a thick plate
Uses of Computer Graphics

- Leisure
  - Games, Special effects
  - Animation film
  - Computer game

- Sciences and technology
  - Computer Aided Design
  - Scientific visualization

- Simulators (flight, etc...) / virtual reality

- Graphics (photoshop, illustrator...)

- Arts
Boeing 747 flight simulator (Wash. DC aeronautics and space museum)
Uses of Computer Graphics

- Leisure
  - Games, Special effects
  - Animation film
  - Computer game

- Sciences and technology
  - Computer Aided Design
  - Scientific visualization

- Simulators (flight, etc...) / virtual reality

- Graphics (photoshop, illustrator...)

- Arts
Adobe Illustrator
Uses of Computer Graphics

- **Leisure**
  - Games, Special effects
  - Animation film
  - Computer game

- **Sciences and technology**
  - Computer Aided Design
  - Scientific visualisations

- **Simulators (flight, etc...) / virtual reality**

- **Graphics (photoshop, illustrator...)**

- **Arts**
Computer sculptures and jewelry
(T Sauermann / Steinbach & Condes)
“Computer Graphics” links to a very broad domain, where many issues have to be addressed...
Computer Graphics

Issues in computer graphics

- 2D pictures
  - Composite picture
  - Digital filtering
  - Colorimetry
  - Conversion

- 2D drawing
  - Illustration, sketches
  - Fonts, graphical user interfaces
Computer Graphics

Issues in computer graphics

- 3D Imaging
  - 3D Scanners
  - Segmentation
  - Compression
Computer Graphics

Issues in computer graphics

- 3D Modeling
  - Representation of 3D shape
  - Polygons, curves and curved surfaces
  - Procedural modeling
Issues in computer graphics

- **3D Rendering**
  - 2D representation of a 3D geometry
  - Projection and perspective
  - Hidden faces
  - Illumination simulations
Computer Graphics

Issues in computer graphics

- Interaction with the user
  - 2D graphic interfaces
  - 3D modeling interfaces
  - Virtual Reality
Computer Graphics

Issues in computer graphics

- Animations
  - Animations via “keyframing”
  - Use of laws of physics

Resolution of a PDE

Interpolation techniques « keyframing »
Make mathematics visible !
Summary of the course
Computer Graphics

Summary

- Introduction
- Images and display techniques
  - Bases
  - Gamma correction
  - Aliasing and techniques to remedy
  - Storage
Computer Graphics

Summary

- 3D Perspective & 2D / 3D transformations
  - Go from a 3D space to a 2D display device

- Representation of curves and surfaces
  - Splines & co.
  - Meshes

- Realistic rendering by ray tracing
  - Concepts and theoretical bases

- Lighting
  - Law of reflexion, etc
Computer Graphics

Summary

- Textures
- Colorimetry
  - Color space
  - Metamerism
- Graphic pipeline and OpenGL
  - Primitives
  - Discretization (*Rasterization*)
  - Hidden faces
- Animations ?
Images and display techniques
What is an image?

- A photo print?
- A negative photo?
- This screen?
- Numbers in RAM?
An image is:

- A 2D intensity and/or color distribution
- A function defined on a 2D plane

\[ I : \mathbb{R}^2 \rightarrow \ldots \]

- We are not talking about pixels for the moment

To do computer imaging, we need:

- Represent images ie digitally encode
- View images – make digital data corresponding to variations in light intensity visible
Display technologies

- « Evanescent » images
  - Computer screen (television ... etc...)
    - Cathode ray tube (CRT)
    - Flat screen (LCD, LED)
  - Projectors
- « Permanent » images
  - Printers
    - Laser
    - Inkjet
  - Photographic process
  - Print media (offset)
- A combination of both
  - Cinema - chemical film… recently replaced by digital projectors
Cathode ray tube
Cathode ray type display

- Vector (oscilloscope-type)
  - Variable refresh rate
  - Limited complexity of drawings
  - Resolution limited only by the size of the electron brush

- Scanning type (e.g. in TV sets)
  - Fixed refresh rate
  - Analog signal
  - Resolution limited by the step of the mask for color screens (on one or two dimensions)
LCD technology

- The LC forces a rotation of the polarization plane when an electric field is present
- Resolution is imposed
- Decomposition of the image in pixels "by construction"
Computer Graphics
Images and display techniques

- Cinema

[Diagram showing the process of film projection with labels for Light shutter, Loop, Film supply, Film gate, Optical sound reader, Shock absorber, Film take-up, First projection of image, Block light without frame advance, Second projection of image, Block light and advance to next image, and a Maltese Cross illustration.]

Maltese Cross
Computer video system

- The image is a 2D array in the video memory
- Usually, this is a dedicated memory (but not always)
- Memory that directly corresponds with the displayed image is the frame memory or frame buffer.

Graphic subset

- Video memory
- «frame buffer»
- Video controller
- Screen
- Display processor
- Main memory
- Bus system
- Other I/O peripherals
- μ-processor
Color
Color?

- The human eye is trichromatic
  - Any color in the visible spectrum is decomposed by the eye into three components: the primary colors
  - By combining these primary colors, one can reconstruct the appearance of any color of the visible spectrum. But it is an optical illusion!

Spectrum of white light

Spectrum of light emitted from a CRT computer screen
Some animals have a much more complex eye

- Mantis shrimp:
  - 12 color filters (i.e. 12 primary colors !)
  - 4 filters for detecting polarized light
  - Trinocular view for each eye ...

- Birds:
  - «only» 4 color filters ...
Color synthesis

- **Additive**
  - Used for screens (light sources)
  - The candles are red, green and blue (RGB system)
  - No signal -> no color -> black

- **Subtractive**
  - Use for printing, photo etc..
  - Use for complementary color: cyan, magenta, yellow (CMY)
  - No pigments -> white (color support)
  - A black pigment is often added (CMY mixing pigment gives only a very dark brown)
  - This is called 4 color printing (CMYK = CMY + blacK)
Synthetic colors on CRT screen ...

... and LCD

- Each subpixel is controllable in intensity ...
- The eye blurs the subpixels and thinks is sees a solid color.
Printing techniques

- **Xerography (laser printing)**
  - Binary (black or white)
  - High resolution and speed
  - Very small isolated dots impossible!
  - Check for the color dithering
  - Color: the operation is repeated 4 times with CMYK toner
Printing technics

- Inkjet
  - Liquid ink projected in very small quantities (a few picoliters)
  - Isolated points are possible
  - Binary image
  - Control of the shade
    - By the volume of the drops
    - By the dot density
  - Ability to print a large number of pigments (sometimes 7!) for a higher fidelity color reproduction.
Printing and screening

- Each dot is a “pure” color
- The colors are controlled by the density and/or the surface of the dots (the ratio between dot surface and total surface)
Numerical camera

- Matrix input device
- The image sensor is an array of millions of photosites arranged regularly

Components of the camera:
- Optical element / Lens
- Pentaprism
- Screen focus
- Eyepiece
- Sensor (in red)
- Shutter
- Movable Mirror
Bayer filter-based digital sensor

- 11 Mp 24x36
- 4 Mp 18x28
- 6 Mp 15x22
Bayer filter-based digital sensor

- The human eye's sensitivity/selectivity is strong for green, so green is favored in the pattern
- Need to extrapolate color info at neighboring pixels
- Many algorithm do exist, most simple one is bilinear interpolation, but yields a « soft » image
- Actual resolution is lower than one can expect from the size of the photosites
There are other systems:

- Some cameras have more complex sensors
  - RGBW (4 type of photosites)
  - Foveon (RGB photosites are stacked, thanks to the transparency of the layers!)
    - 3 color known at each position – however, a lot of numerical treatment is needed to get back the right RGB color info.
- Professional video cameras (often 3 separate B&W CCD sensors)
- Flatbed scanners
  - The sensor consists of a single row of RGB photosites
  - The final image is made of numerous such lines placed side by side.
  - Each line is "shot" made at a different time.
All these systems suggest a 2D array of numbers representing the image in memory

- Advantage: we can represent any image
  - Better approximation as the resolution increases
  - This works because memory is cheap (« brute force » approach)
  - It is possible to take advantage of the image structure to reduce its size in memory
Meaning of a matrix image

- Function on the 2D plane
- Result of an input device???
  - But: there are several types of input devices
  - But: sometimes leads to images can not be displayed (e.g., too large)
- The real problem is the reconstruction
  - An image is a discrete representation.
  - The value of a pixel means "the intensity is such in this place"
    - This is a sampled value
      - LCD: the intensity is constant over a square region
      - CRT intensity varies continuously (looks like a Gaussian)
  - Problems of reconstruction will be discussed later
Types of images and associated data

- **B&W**: 1 bit per pixel
  - Interpretation: fax image
  
  \[ I : \mathbb{R}^2 \rightarrow \{0,1\} \]

- **Grayscale**: 1 value per pixel
  - Black & white image or photograph
  - Accuracy: typically 8 bits (but sometimes 10, 12 ou 16 bpp)
  
  \[ I : \mathbb{R}^2 \rightarrow [0,1] \]

- **Color**: 3 values per pixel
  - Color photography
  - Accuracy: typically 3*8 bits (24 bits/pixel)
  - Sometimes 16 (5+6+5) ou 30,36,48 bits
  - Indexed color: sometimes useful (line-art)

\[ I : \mathbb{R}^2 \rightarrow \left\{0, \frac{1}{2^n-1}, \cdots, \frac{2^n-1}{2^n-1}\right\} \]

\[ I : \mathbb{R}^2 \rightarrow [0,1]^3 \]
Types of images and associated data

- Sometimes we use floating point numbers instead of integers
  \[ I : \mathbb{IR}^2 \to \mathbb{IR}_+ \quad \text{ou} \quad I : \mathbb{IR}^2 \to \mathbb{IR}^3 \]
  - More abstract, as no output device does have an infinite scale
  - Used to represent high contrast image (High Dynamic Range = HDR)
  - Represents scenes regardless of the output device
  - Becomes a standard in professional image processing
Types of images and associated data

- « Clipping » and « white point »
  - It is customary to calculate floating point and then converted into n-bit integers (usually n = 8) for displaying
  - The total scale may not fit within the range of the output device (monitor or printer)
  - Simple solution: choose a maximum value (white point), it becomes the maximum intensity ($2^n-1$ in an n-bit representation)
    - Anything that exceeds the maximum value is white (clipping = loss of detail)
Exposure +0 f stops
  - f/8, 1 s
Exposure: -8 f stops
- f/8, 1/250 s
- Exposure +5 f stops
  - f/8, 30 s
Types of images and associated data

- For color images and grayscale, sometimes we add an “alpha” channel
  - Alpha is the transparency
  - Between 0 and 1
  - Usually encoded with the same precision as the RGB color
Storage constraints for images

- 1024x1024 pixels Image (1 megapixel)
  - B&W = 128 KB
  - Grayscale 8 bpp : 1 MB
  - Grayscale 16 bpp : 2 MB
  - Color 8 bpp : 3 MB
  - Color 8 bpp +alpha : 4 MB
  - Color 12 bpp : 4.5 MB
- Color HDR Floating Point (32x3 bpp): 12 MB
- Current SLR cameras (2015) take pictures between 16 megapixels (24x36) and 75 megapixels (medium format)
Conversion between types of images

- Color to grayscale of the same "precision" (eg 24 bits to 8 bits)
  - Take one of the channels (R, G ou B)
    - Sometimes strange appearance
  - Combining channels is better
    - Basic RGB colors contribute differently to the luminance
    - What is more bright: 100% blue or 100% green?
      - A good choice: gray (luminance) = 0.2 R + 0.7 G + 0.1 B
      - We'll talk about it later ...
Change in number of bits / plane (precision)

- Up is easy
  - No loss of information
- Down: beware!

Many levels   16 levels   8 levels   4 levels   2 levels
Reducing the number of bits per pixel (bpp) is called quantization.

- If the quantization is consistent, global “Mach” bands are often visible.
  - “Consistent” means the final value for one pixel does not depend on the other pixels, only on the original pixel at the same place.
- It may not be consistent – we call that “dithering”.
  - It only lights up some pixels in the gray areas.
  - It is a compromise between spatial resolution and tonal resolution.
  - You can choose the type of dithering depending on the output device.
    - Laser, offset printing: packs of points (halftone)
    - LCD, inkjet: can display / print isolated dots at the resolution limit.
Examples of dithering algorithms $8 \rightarrow 4 \text{ bpp}$

- Consistent (Threshold)
  - Mach bands are very visible. If the choice of the threshold and the image are appropriate, there is not too much loss of detail.
Examples of dithering algorithms $8 \rightarrow 1 \text{ bpp}$

- Consistent (Threshold)
  - Mach bands are very visible. If the choice of the threshold and the image are appropriate, there is not too much loss of detail
Halftone

Based on optical solutions

- This is suitable for laser printing, and offset, but the effective resolution is a fraction of that of the printer (1200 dpi → 75 dpi, typically)
Computer Graphics
Images and display techniques

- Bayer dithering
- Advantageous for devices capable of reproducing isolated points
  - Technique is quite old but still used on LCD screens...
Error diffusion dithering (Floyd-Steinberg algorithm)

Advantageous for devices capable of reproducing isolated points

- Replaces halftoning for inkjet printers
Resolution may be artificially increased before dithering ... But one cannot exceed the resolution of the device!
Digital dithering algorithms

- Consistent dithering

**Principle**

- Scans the image in any order (the result for each pixel does not depend on its neighbors)
- For each pixel value, the nearest value is sought in a palette and this value is displayed.

```
For y from 0 to nblines-1
    For x from 0 to nbcol-1
        oldpixel = pixel[x,y]
        newpixel = round(oldpixel)
        pixel[x,y] = newpixel
    EndFor
EndFor
```

Round to nearest
Ordered dithering

Principle (B&W)

- Scan the image in any order (the result for each pixel does not depend on neighbors)
- For each point, check that the value is greater or less than a test value found in a matrix (the Bayer matrix)
- If lower, draw black, if higher, draw white
- This dithering can be used to convert to 4, 9 or 16 distinct intensities, see next algorithm.
General algorithm for an ordered dithering

- The intensities of the pixels are scaled to take a real value between 0 and 1
- Return value is a boolean (0 or 1)

For \( y \) from 0 to \( \text{nblines}-1 \)
  For \( x \) from 0 to \( \text{nbcol}-1 \)
    \( \text{olddpixel} = \text{pixel}[x,y] + \text{bayer}[x \mod n, y \mod m] \)
    \( \text{newpixel} = \text{floor}(\text{olddpixel}) \)
    \( \text{pixel}[x][y] = \text{newpixel} \)
  EndFor
EndFor
Generation of Bayer matrices for the ordered dithering

- Block-like dithering (simulates "halftoning")

\[
\begin{align*}
\text{Intensity} & \quad \frac{1}{10} \begin{pmatrix} 8 & 3 & 4 \\ 6 & 1 & 2 \\ 7 & 5 & 9 \end{pmatrix} \\
\end{align*}
\]

- Dispersed dithering (classical)

\[
\begin{align*}
\text{Intensity} & \quad \frac{1}{10} \begin{pmatrix} 1 & 7 & 4 \\ 5 & 8 & 3 \\ 6 & 2 & 9 \end{pmatrix} \\
\end{align*}
\]

Error diffusion dithering (Floyd Steinberg)

Principle:
- Pixels scanned from left to right and top to bottom
- We compute the closest allowed value (by rounding)
- The error is then calculated
- This error is transferred to the neighboring pixels
- Thus, the global error is kept minimal

Floyd & Steinberg's matrix

\[
\begin{pmatrix}
1/16 & 0 & 0 & 0 \\
0 & 3 & 5 & 1 \\
7 & 0 & 0 & 0 \\
\end{pmatrix}
\]

Actual pixel \(k=1, l=0\)

Error diffusion dithering algorithm

For y from 0 to nblines-1
    For x from 0 to nbcol-1
        oldpixel = pixel[x,y]
        newpixel = round(oldpixel)
        pixel[x,y] = newpixel
        error = oldpixel-newpixel
        For j from 0 to m-1
            For i from 0 to n-1
                If matrix[i,j]<>0
                    pixel[x+i-k,y+j-l] = pixel[x+i-k,y+j-l] +
                    error*matrix[i,j]
                EndIf
            EndFor
        EndFor
    EndFor
EndFor
## Variants of the Floyd-Steinberg algorithm

<table>
<thead>
<tr>
<th>Variant</th>
<th>Filter Coefficients</th>
<th>Actual Pixel</th>
<th>k, l</th>
</tr>
</thead>
</table>
| Jarvis et al.      | \[
\begin{pmatrix}
\frac{1}{48} & 0 & 0 & (0) & 7 & 5 \\
3 & 5 & 7 & 5 & 3 \\
1 & 3 & 5 & 3 & 1 \\
\end{pmatrix}
\] | k=2, l=θ       |
| Stucki             | \[
\begin{pmatrix}
\frac{1}{42} & 0 & 0 & (0) & 8 & 4 \\
2 & 4 & 8 & 4 & 2 \\
1 & 2 & 4 & 2 & 1 \\
\end{pmatrix}
\] | k=2, l=θ       |
| Sierra             | \[
\begin{pmatrix}
\frac{1}{32} & 0 & 0 & (0) & 5 & 3 \\
2 & 4 & 5 & 4 & 2 \\
0 & 2 & 3 & 2 & 0 \\
\end{pmatrix}
\] | k=2, l=θ       |
| Sierra (« Lite »)  | \[
\begin{pmatrix}
\frac{1}{4} & 0 & (0) & 2 \\
1 & 1 & 0 \\
\end{pmatrix}
\] | k=1, l=θ       |
Ordered dithering
Floyd – Steinberg dithering
Computer Graphics
Images and display techniques

Ordered dithering
Floyd – Steinberg dithering
Encoding light intensity within images

- What is the exact meaning of the value stored in pixels?
  - They determine the brightness
  - The higher the number, the more bright it is (usually)

- Transfer function: A function that associates the value stored in a pixel with the luminance of the displayed pixel
  \[ I = f(p) \quad f : [0, N] \rightarrow [I_{\text{min}}, I_{\text{max}}] \]

What determines this function?

- Physical constraint in the medium or device
- The human eye has a non-linear transfer function!
- Desired visual characteristics
A small experiment ...

- A pattern using only white and black is used there (Moiré effect!)
  \[ 0.5(I_{\text{max}} + I_{\text{min}}) \]

- Uniform brightness is used there (no pattern)
  \[ p \in [0..2^n - 1] \]
We then check when the inner and outer area have roughly the same brightness, seen from afar.
Computer Graphics
Images and display techniques

p=127
Computer Graphics
Images and display techniques

p=191
Computer Graphics
Images and display techniques

\[ p = 169 \]
In fact, the projector/screen does something like this:

\[ I(p) = 0.5(I_{\text{max}} + I_{\text{min}}) \]

Linear response

\[ N = 255 \]
Parameters of the transfer function

- **Maximum intensity** \( (I_{\text{max}}) \)
  - What light power can be transmitted by a pixel?
    - LCD: transmission efficiency: less than 10%! / Projectors are better

- **Minimum intensity** \( (I_{\text{min}}) \)
  - It is the emitted intensity when the pixel is off.
    - Depends on the quality of e.g. LCD polarizers / OLED screens much better in this aspect

- **Reflection of the ambient light on the device** \( (r) \)
  - Very important factor determining the apparent contrast
    - 5% \( I_{\text{max}} \) typically, 1% \( I_{\text{max}} \) for a dedicated environment
    - Explains why video screens tend to be black (if possible) because the environment is uncontrolled, and why light is dimmed in a cinema (the screen is white to reflect most of the incoming light!)
Contrast ratio

- \[ C_d = \frac{I_{\text{max}}}{I_{\text{min}}} \text{ ou } \frac{(I_{\text{max}} + r)}{(I_{\text{min}} + r)} \]
  - Important factor with respect to the quality of a displayed image

- "Usual" values
  - Screen in a normal office environment: 20:1 (sRGB)
  - Paper photograph 30:1
  - Screen in controlled lighting conditions: 100:1 (sRGB)
  - Slide / film (viewed in good conditions) 1000:1
  - HDR screen 10000:1 (lab measurements without factor r)
Shape of the transfer function

- Desired property: the intensity gap from one value to the next should not be visible
  - Eliminates banding ("Mach" bands) on smooth images.
- What minimum contrast the eye is able to distinguish?
  - In good lightning conditions, 1-2% in relative intensity
    - 2% relative, not absolute
    - So we should have intensity values closer in the "dark gray" than in "light gray"
    - Exponential transfer function is optimal.
How many levels do we need?

- It depends on the maximum degree of contrast
  - (unequal) intervals of 2%:
    
    \[
    0 \rightarrow I_{\text{min}} ; 1 \rightarrow 1.02 I_{\text{min}} ; 2 \rightarrow (1.02)^2 I_{\text{min}} ; \cdots
    \]

  \[
  \log_{10} 1.02 \approx 0.086 \approx 1/120
  \]

- So it takes about 120 distinct levels per decade of contrast ratio
  - 240 for a display in controlled lighting
  - 360 for slides/movie
  - 480 for a high quality screen (HDR)

- If one wants equal intervals: each interval should be < 2 % I_{\text{min}}
  - It must go from ~0 to I_{\text{min}}*Cd therefore close to 50*Cd intervals.
  - 1500 for a paper print, 5000 for a screen printing in controlled lighting, 500 000 for HDR display. A huge difference!
Morality

- The 8 bit quantization that is so widespread is barely enough for "low-end" applications
  - And this, only if the transfer function is adequate!
- In this case, quantization is not sufficient to perform image processing involving colors
  - E.g. contrast adjustment etc. ...
- This is OK as final image format for display on a screen, in an office environment.
Quantization in practice:

- **Linear quantization**
  \[ I(n) = \left( \frac{n}{N} \right) I_{\text{max}} \]
  - Simple, practical, integer arithmetic
  - Large number of intervals required
  - 12-bit or 16-bit floating point numbers minimum for HDR

- **Power law quantization**
  \[ I(n) = \left( \frac{n}{N} \right)^{\gamma} I_{\text{max}} \]
  - Still simple; approximation of an exponential quantization
  - Need to linearize for intensities close to zero
  - 8 bits are OK, 12 bits for critical applications

- **Exponential quantization**
  \[ I(n) = I_{\text{min}} C_d \left( \frac{n}{N} \right) \]
  - Ideal quantization
  - Expensive
  - Requires choosing a non-zero minimum intensity ... (ambiance)
In practice, the power law quantization is used (gamma quantification)

- Bad reason: CRT tubes work this way!
  \[ I_{\text{screen}} \propto V^{2.2} \]
  - Real reason: the human eye is a non-linear sensor. The CRTs were also designed so.

- Another reason: inertia and low memory requirements
  - Inertia: gamma correction is close to the exponential correction no reason to change for little improvement.
  - Memory: With gamma correction, it is possible to encode an image with 8 bits / channel for an acceptable result (the case 99% of the images on PC)
  - Suitable for transmission of images on the web.
  - Lossy compression algorithms usually work with gamma-encoded 8-bit images (e.g. jpg)
Gamma quantification

Close enough to the ideal quantification...
The display and perception are non-linear

- Displays can be generally approximated by a gamma coefficient of 2.2
  - This has become a “de facto” standard
  - All that is displayed assumes this quantification principle.
- More specifically, non-calibrated equipment is expected to respond according to the IEC sRGB profile
  - IEC = International Electrotechnical Commission
  - Standard IEC 61966-2-1:1999
  - The image files are supposed to be encoded with this profile if nothing is specified.
  - We will see later in the course what it means.
Why gamma 2.2?

- It comes precisely from the physiology of the eye.
  - Basically, a light intensity equal to 18% of a reference intensity appears half as bright.
  - The sensitivity of the eye can be approximated by the formula *

\[
L^* = 116 \left( \frac{Y}{Y_0} \right)^{1/3} - 16 \quad \text{; } \quad \frac{Y}{Y_0} > \left( \frac{6}{29} \right)^3
\]

\[
L^* = \frac{116}{3} \left( \frac{29}{6} \right)^2 \left( \frac{Y}{Y_0} \right) \quad \text{; } \quad \frac{Y}{Y_0} \leq \left( \frac{6}{29} \right)^3
\]

Y is the brightness (luminance), L* is the apparent brightness

*This formula is used to move from a linear color space (ie light intensities measured physically) to a perceptual color space (i.e. relative brightness impression for the eye).

- It corresponds roughly to a gamma of 2.4 . Why 2.2 then?
  - We often watch TV in a room with uncontrolled environment .... and in these conditions, a little more contrast does make sense.
  - Gamma = 1 at the bottom of the curve (to avoid infinite slope of the reciprocal function). That must be compensated with the rest ...
- sRGB

\[ \gamma = 2.2 \]

\( \Gamma \) exponent (slope in a log-log scale)

\( \gamma = 2.2 \) (dashed)

sRGB (continuous)
Gamma correction

- One sometimes wants to display unencoded images whose values represent real light intensities « $I_r$ ».
  - Either they were computed like this (ray tracing, etc.)
  - Or they were obtained with linear physical devices (e.g. numerical photography)
- One should take the implicit gamma quantization of the display device!

\[ I(n) = I_r = \left( \frac{n}{N} \right)^\gamma \]

- Computer screen with a zero value for black

Solve to get:
\[ n = N \left( I_r \right)^{\frac{1}{\gamma}} \]

- This is called "Gamma correction", and it should be applied to linear brightness data before conversion to 8bits for display purposes.
  - If this is forgotten, images are dark and overcontrasted.
Gamma Correction

Uncorrected  Corrected for the screen's gamma (2.2)  Corrected for a higher gamma (3.5)
Gamma correction and dithering

- If a 8-bit grayscale image is dithered (e.g. to be printed), one must use the gamma used in the encoding to correct the algorithm.

- Reason: only min and max levels of brightness, which are independent of the gamma, are used. However, the intentional rendering depends on the gamma ...

\[
I(n) = (n/N)^y
\]

\[
n(I) = N I^y
\]

For \( y \) from 0 to \( n_{\text{blines}} - 1 \)
For \( x \) from 0 to \( n_{\text{bcol}} - 1 \)

\[
\text{oldpixel} = \text{pixel}[x,y] + \text{bayer}[x \mod n, y \mod m]
\]

\[
\text{newpixel} = \text{floor}(\text{oldpixel})
\]

\[
\text{pixel}[x][y] = \text{newpixel}
\]


Here, use physical (linear) intensity scale!
How to roughly approximate the gamma correction of an image processing pipeline ... (this projector for instance).