1 Statistical Signal Modelling

1.2: Modelling of memoryless sources

1.2.a: Sample-wise operators

Statistical Signal Modelling

1.2

1. Introduction to IPA and Random variable

2. **Modelling of memoryless processes**
   - Sample-wise operators
   - Uniform and non-uniform quantization
   - Examples: Sample-wise video processing

3. **Discrete Stochastic Processes**
   - Definition
   - Autocorrelation: Deterministic signals and processes
   - Stationarity and Ergodicity
   - Power Spectral Density (PSD)
   - Stochastic processes filtering
   - Examples

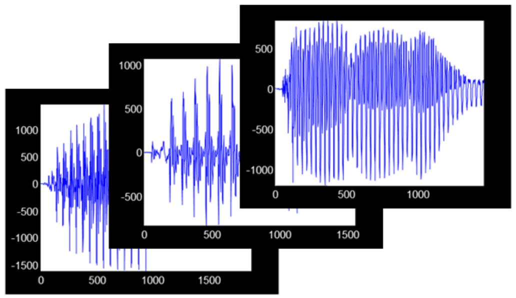
Unit Structure

1. **Introduction**
   - Memoryless processes
   - Image model definition
2. **Generic operators**
   - Range transform operation
   - Implementation
3. **Histogram based operators**
   - Histogram definition
   - Histogram equalization
4. **Example: Biomedical application**
5. **Summary and Conclusions**

Introduction: Stochastic Processes

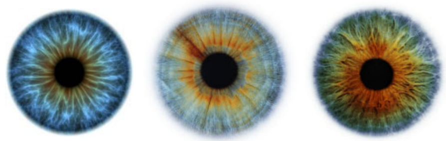
- When we have a set of signals that can be analyzed as being the result of the same given experiment, we model these signals as a **stochastic process** to jointly study them:
  - The formal definition of a stochastic process will be given in Section 1.3

Cork texture



Word "el"

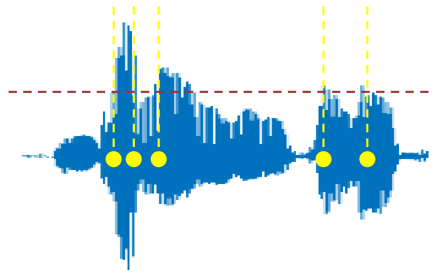
Human face (M2VTS Data base)

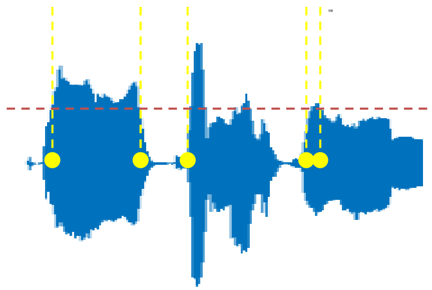


Human iris

Introduction: Memoryless Processes

- The analysis of a stochastic process relies on the **statistic** of the samples and on the **dependencies** among samples
- **Memoryless processes** assume that every sample of the process is **independent** of its neighbor samples.
- **Processing of memoryless processes**: Only take into account the **sample values** (**sample-wise operators**), but not their index (time instant or position) or their neighbor sample values:
  - Non-linear operations
  - Operations defined by a mapping function
  - They process in the same way all samples with the same value
  - Commonly used for **perceptual purposes**





Introduction: Image Model

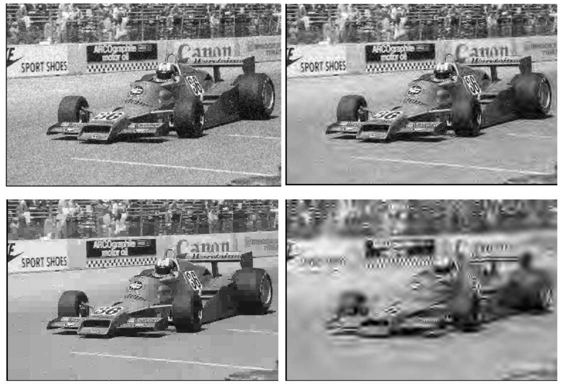
- Sample-wise operators are largely used in **image processing**:
  - In the **pixel-based image model**, the image is understood as a collection of independent picture elements (pixels).
  - Other examples (speech, audio, video) will be presented
- Operations only take into account the pixel values:
  - They process in the same manner all pixels with the same value.
- Pixel-based image operators are defined:
  - In a generic way, **without taking into account the specificity of the images**: Range transforms operators.
  - In a specific way, **adapting the operator to the image pixel statistics**: Histogram-based operators.

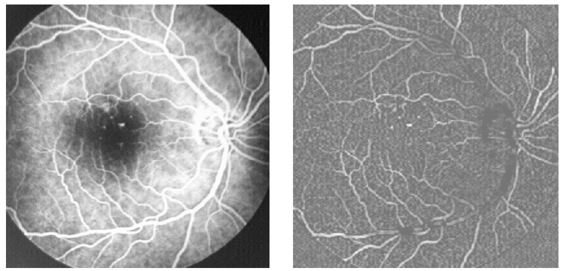
Pixels with values equal to 42 (Yellow) and 240 (Green) pixels

Introduction: Memoryless Operators

1.2

- They are **very fast operators** since only require accessing at the pixel value of the pixel being processed
  - They are **memory-less operations** since do not require storing any neighbor pixel values.
  - Other image models require analyzing a neighborhood of the pixel being processed:
    - Space/Frequency,
    - Geometrical,
    - Region-based models
- We analyze **two main types** of memoryless operators:
  - **Range transform operators**.
  - **Histogram-based operators**.





Unit Structure

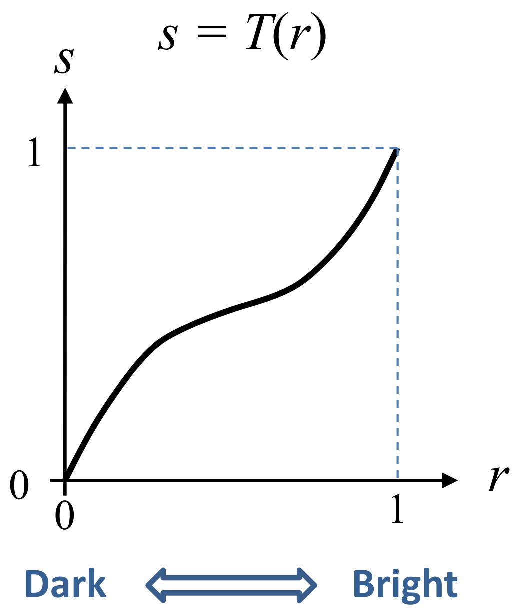
1.2

1. **Introduction**
   - Memoryless processes
   - Image model definition
2. **Generic operators**
   - Range transform operation
   - Implementation
3. **Histogram based operators**
   - Histogram definition
   - Histogram equalization
4. **Example: Biomedical application**
5. **Summary and Conclusions**

Range Transform Operators

1.2

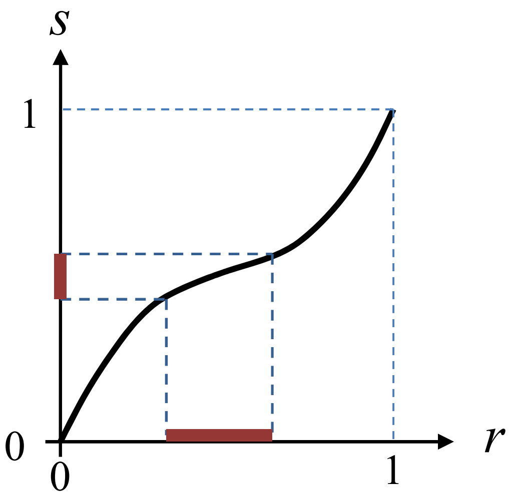
- Define a transformation or **mapping** ($T(.)$) on the range of values of the input image (r) onto the range of values of the output image (s)
  - In the examples, **ranges are normalized [0, 1]** but they may represent different real ranges.



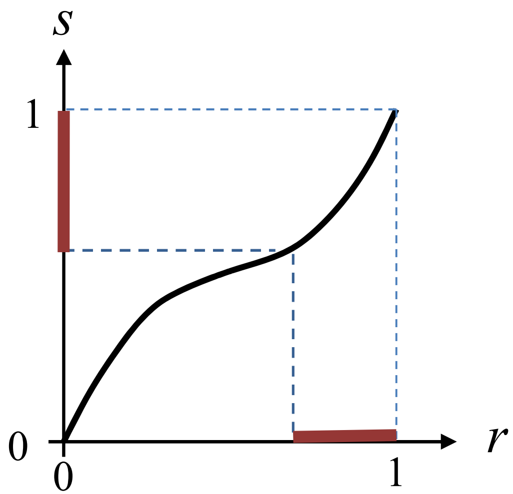
Range Transform Operators

1.2

Grey level mapping: Different segments of the input range are **expanded or compressed** depending on the transform features



Compression

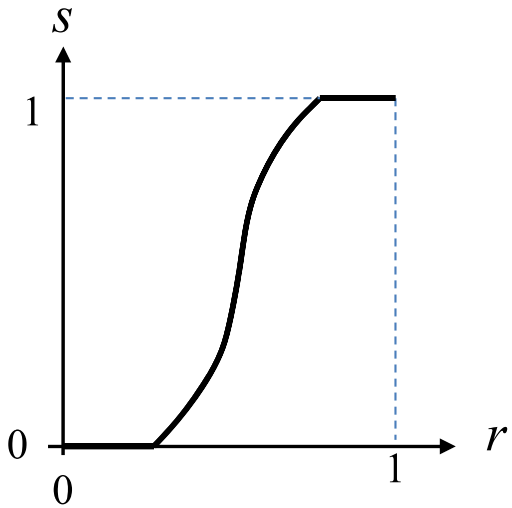


Expansion

- The segments of r where the (magnitude of the) **derivative** of $T(r)$ is greater than 1 are expanded and vice versa.

Range Transform Operators

Contrast mapping: It **expands** (stretches) a range of the input image, mapping it into the whole output image range.



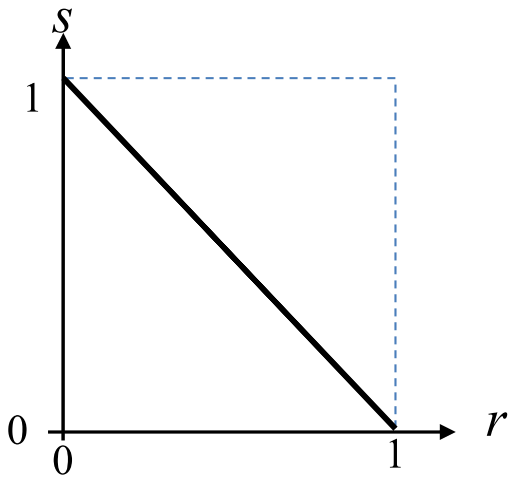
Clipping: A set of values of r are mapped into a single value of s.

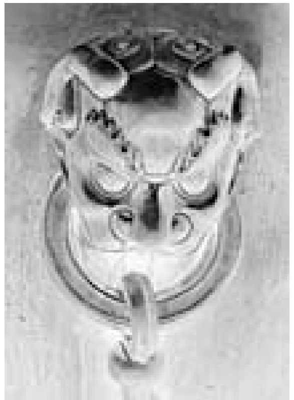
- Typically, lowest and highest values of r are mapped to the minimum and maximum values of s, respectively.
- This is a **non-reversible transform**: It is not bijective.

Range Transform Operators

1.2

Negative mapping: It **inverts** the range of values of the input image creating a negative version of it.

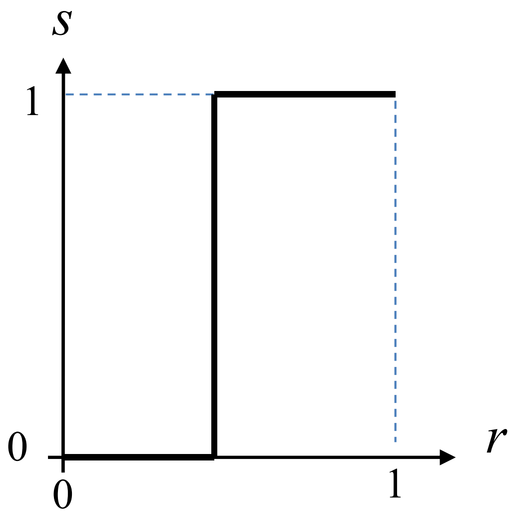




- Negative mappings **do not change the contrast** of the image:
  - The difference between two neighbor pixels remain the same.
  - The magnitude of the **derivative of $T(r)$ is equal 1** in the whole range of the input image.

Range Transform Operators

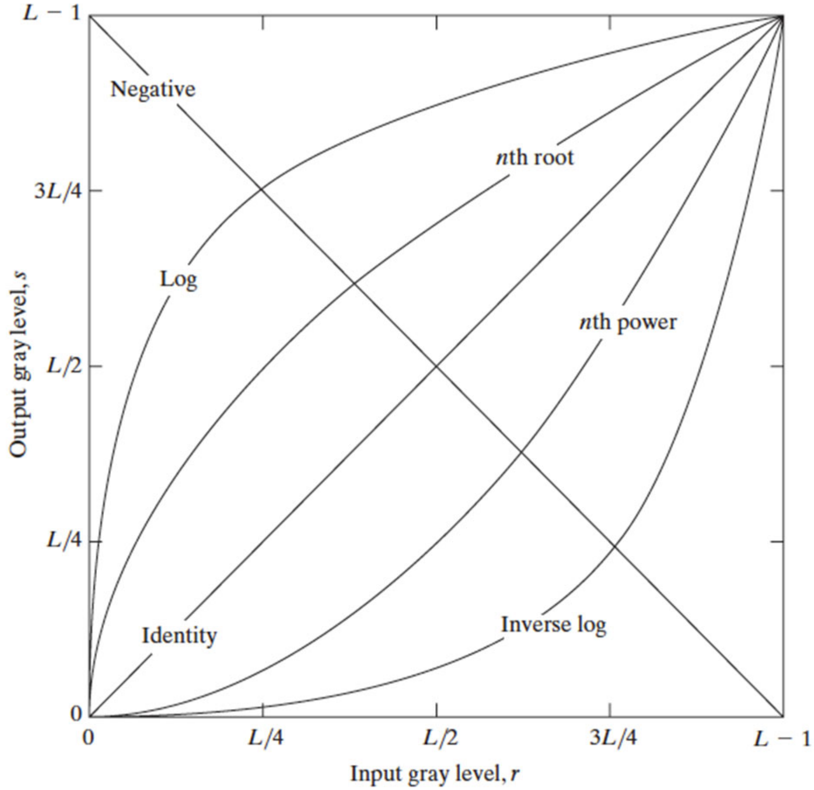
Binarization mapping: It **binarizes** the image by clipping all values below a given threshold to 0 and all values above this threshold to 1.

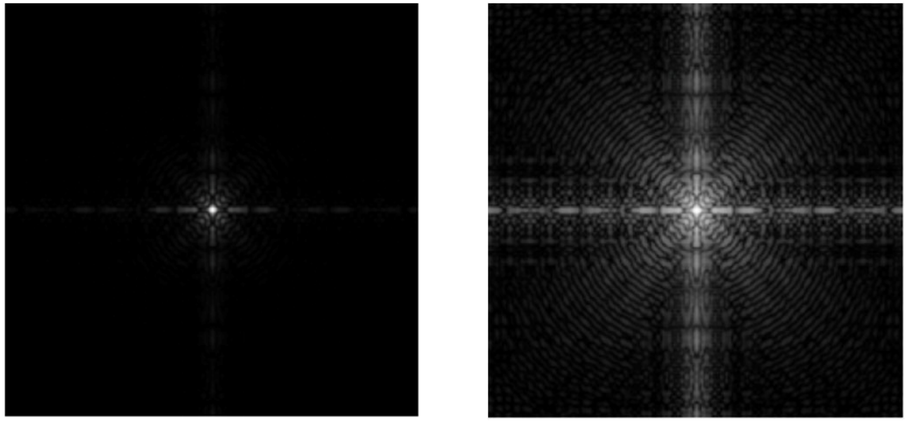


- It is commonly known as **thresholding**.
- It is a **non-reversible operation**, since it is based on two clippings.

Range Transform Operators

Log transformation: mainly used to **compress** the dynamic range





Log transform of
2D Fourier transform

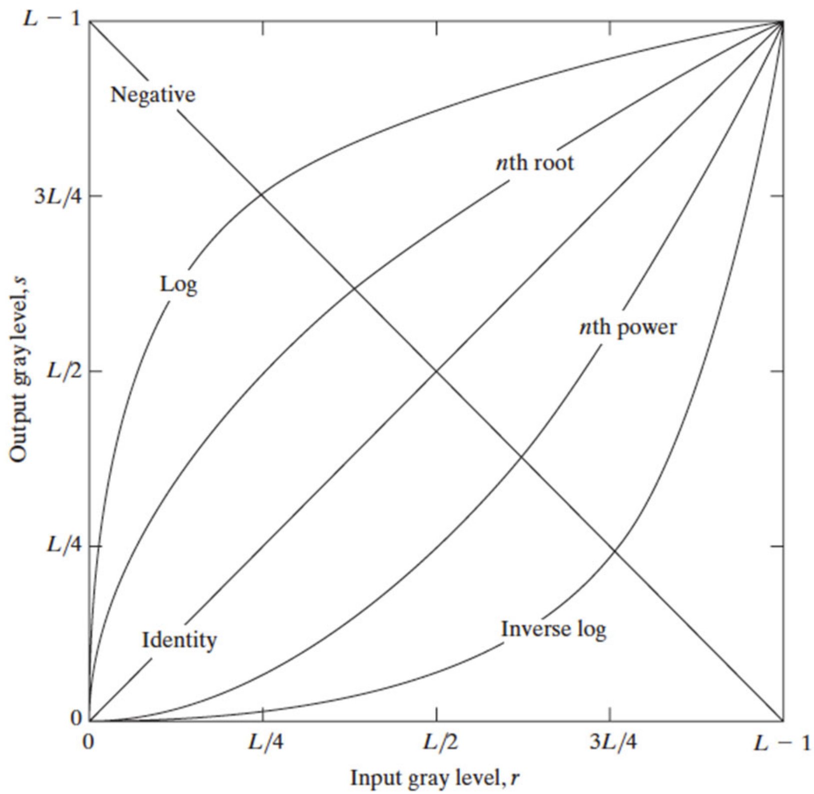
R. Gonzalez & R. Woods, Digital Image Processing, Prentice Hall.

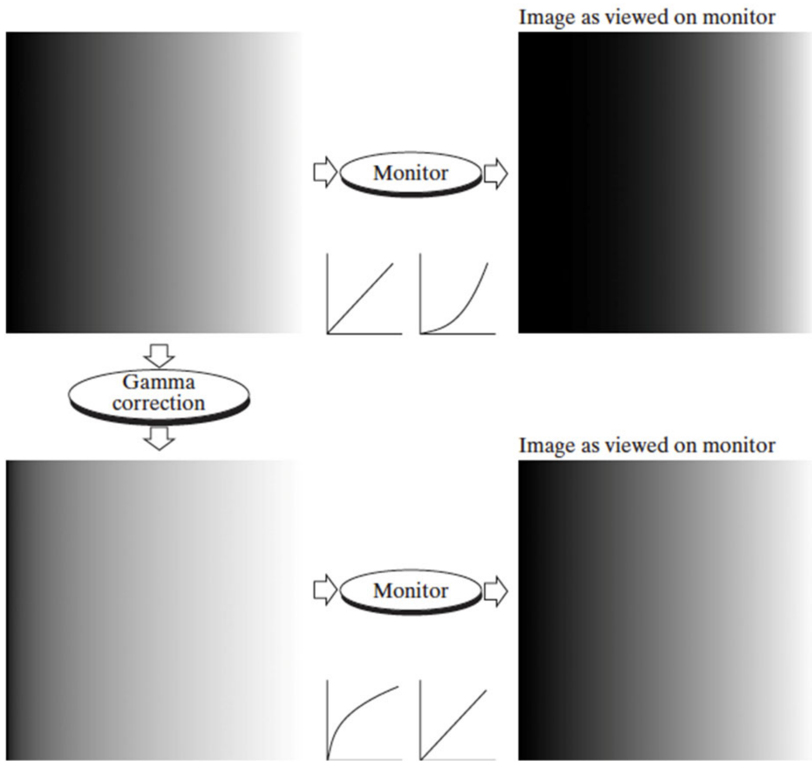
Range Transform Operators

1.2

Power-law: (aka) Gamma correction was originally developed for **correcting Cathode Ray Tubes (CRT) distortion**.

- Useful to implement the **Stevens power law** for brightness perception





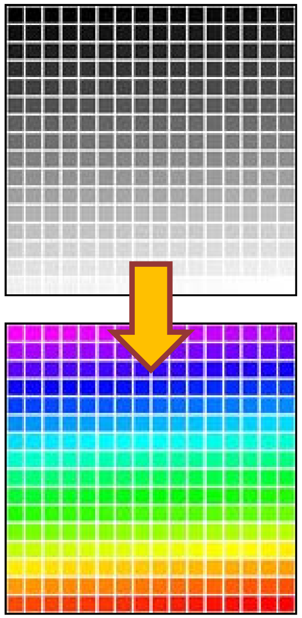
R. Gonzalez & R. Woods, Digital Image Processing, Prentice Hall.

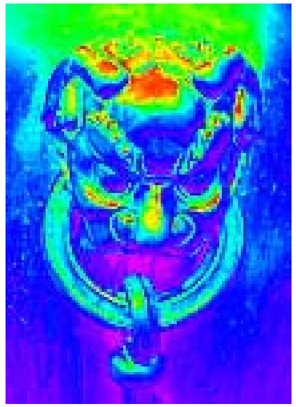
Range Transform Operators

Pseudo color: The input range (r) is mapped into a higher dimensional space; for example, a **3D space representing a color-space**.

- A color (s) is assigned to every grey level value (r)

- The transform can be represented by means of a **Look-Up-Table** (LUT) that puts in correspondence the input and output values.

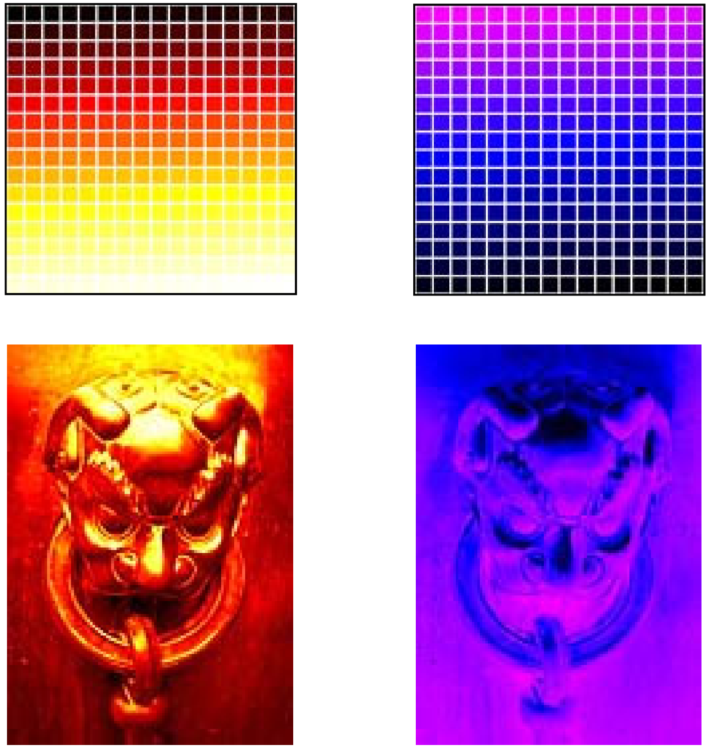




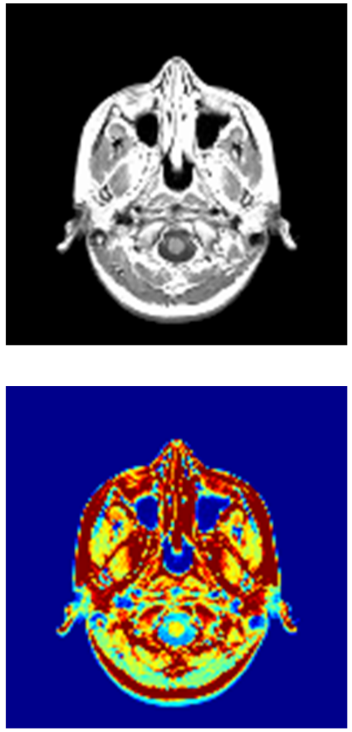
Range Transform Operators

1.2

Pseudo color: Commonly used for visualization purpose in:



Artistic applications

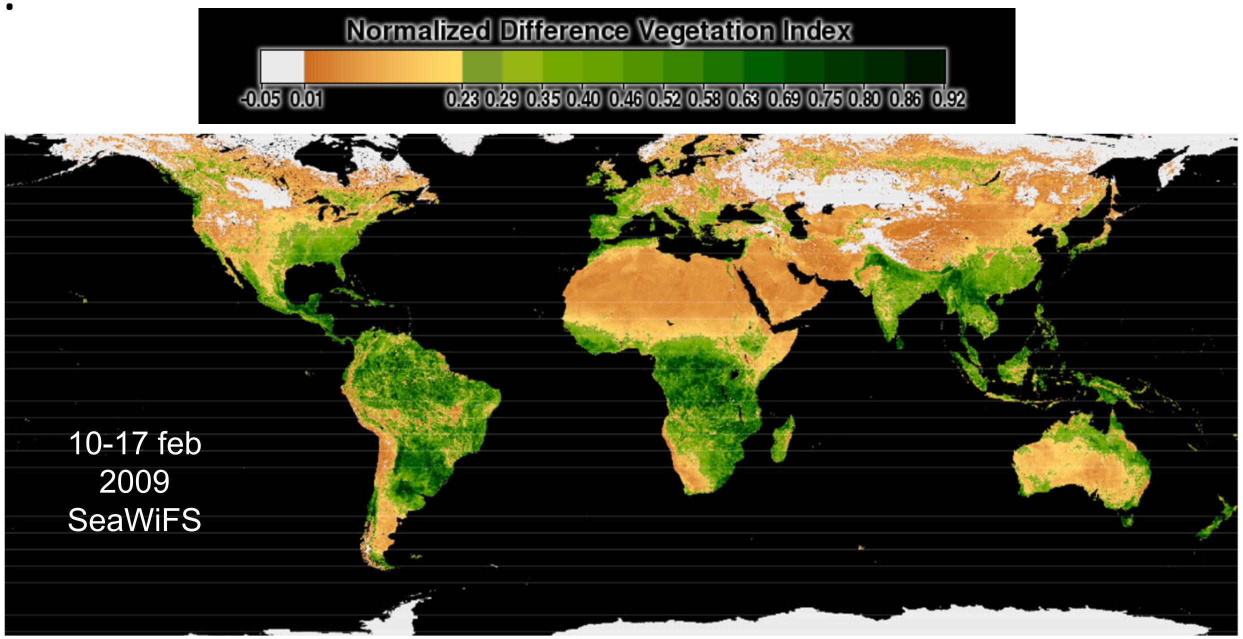


Biomedical applications

Range Transform Operators

1.2

Pseudo color: In satellite imagery, **spectral indexes** are combinations between bands to obtain a parameter of interest (vegetation, water, minerals..):



- Based of ratios between NIR and R bands.
- **Normalized Difference Vegetation Index** (NDVI) is very used

Unit Structure

1.2

1. **Introduction**
   - Memoryless processes, Image model definition
2. **Generic operators**
   - Range transform operation
   - Implementation
3. **Histogram based operators**
   - Histogram definition
   - Histogram equalization
4. **Example: Biomedical application**
5. **Summary and Conclusions**

Direct implementation

1.2

Let us assume that we want to implement $s = T(r) = \log(r+1)$.

A possible approach is to **compute the transform sample by sample**:

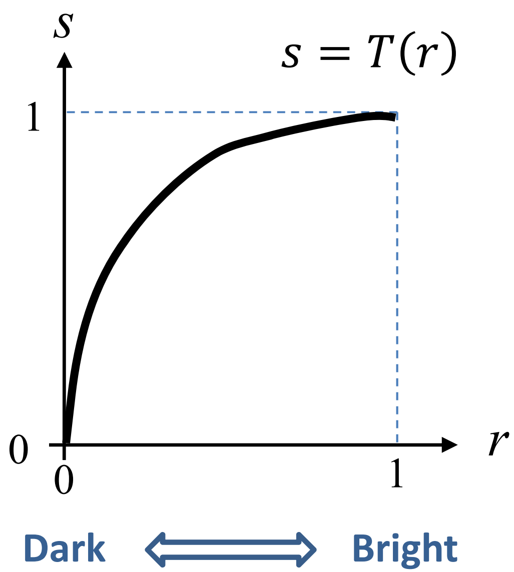
```
% Transform the image
for (i=0; i<M; i++) {
        for (j=0; j<N; j++) {
                y(i,j) = log(x(i,j) + 1);
        }
}
```

- **Example**: For a **grey level, 4K image** (8 bits: 256 levels and 3840x2160 pixels), this implementation performs 8.294.400 logarithm computations
- … But the resulting image has only **256 possible different values**!:
  - Those resulting from computing $\log(k+1) \quad k \in \{0, 1, \cdots, 255\}$

Look Up Table

1.2

Look-Up-Table (LUT): The transform can be implemented by means of a LUT. A LUT is an array that puts into correspondence the input and output values. It replaces **runtime computation** with a simpler **array indexing operation**



- ❑ Assuming that your input (r) and output (s) are discrete variables, **discuss the implementation** of a Range Transform using a Look-Up-Table. Write a pseudo-code of the operation.

Implementation: Look Up Table

1.2

Look-Up-Table (LUT): The transform can be implemented by means of a LUT. A LUT is an array that puts into correspondence the input and output values. It replaces **runtime computation** with a simpler **array indexing operation**

```
% Build the LUT
for (k=0; k<Max_val; k++) {
        LUT(k) = log(k + 1);
        }

% Transform the image applying the LUT
for (i=0; i<M; i++) {
        for (j=0; j<N; j++) {
                y(i,j) = LUT(x(i,j));
        }
}
```

Unit Structure

1. **Introduction**
   - Memoryless processes, Image model definition
2. **Generic operators**
   - Range transform operation
   - Implementation
3. **Histogram based operators**
   - Histogram definition
   - Histogram equalization
4. **Example: Biomedical application**
5. **Summary and Conclusions**

Histogram definition

1.2

- The **histogram** $h(r_k)$ of a grey-level image with range [0 … L-1] is a discrete function that stores for each possible image value (r_k) **the number of occurrences** of that value in the image (n_k); that is, the number of pixels in the image with a given grey-level value.

$$h(r_k) = \#\, pixels\, with\, value\, r_k = n_k \quad \forall r_k \in [0 \ldots L-1]$$

- The histogram information is related to the probability of occurrence of a given value in the image. The **normalized histogram** $p(r_k)$ is an **estimation of the probability density function** (pdf) of a random variable associated to the grey-level values of the image pixels.

$$p(r_k) = \frac{h(r_k)}{\sum_{k=0}^{L-1} h(r_k)} = \frac{n_k}{\sum_{k=0}^{L-1} n_k} = \frac{n_k}{n} \quad \forall r_k \in [0 \ldots L-1]$$

Histogram computation

1.2

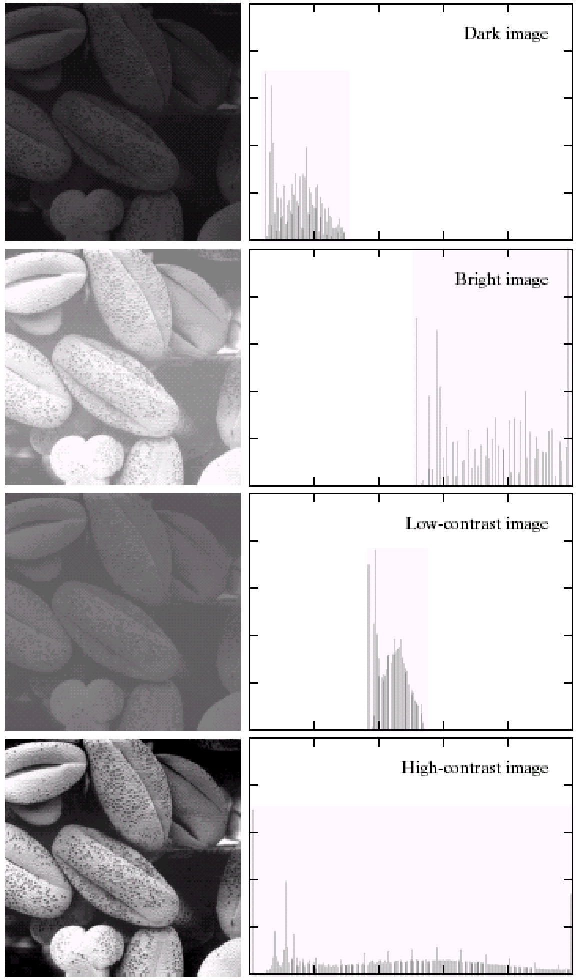
- The **histogram** $h(r_k)$ of a grey-level image with range [0 … L-1] is a discrete function that stores for each possible image value (r_k) **the number of occurrences** of that value in the image (n_k); that is, the number of pixels in the image with a given grey-level value.

$$h(r_k) = \# \, pixels \; with \; value \; r_k = n_k \;\; \forall r_k \in [0 \ldots L-1]$$

```
Compute the histogram of an image
for (k=0; k<Max_val; k++) h(k) = 0;

for (i=0; i<M; i++) {
        for (j=0; j<N; j++) {
                h(x(i,j))++;
        }
}
```

Histogram examples



Dark Image: Grey level values concentrated in the lowest range

Bright Image: Grey level values concentrated in the highest range

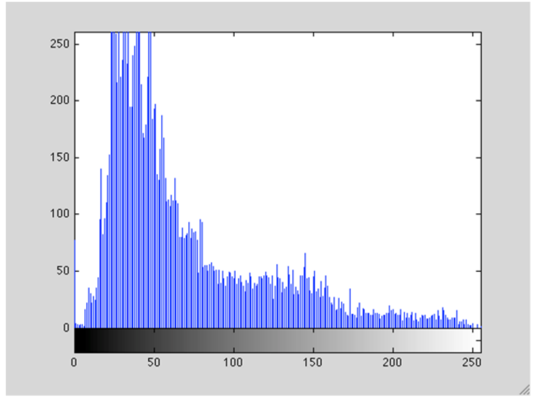
Low contrast Image: Grey level values concentrated in a small range

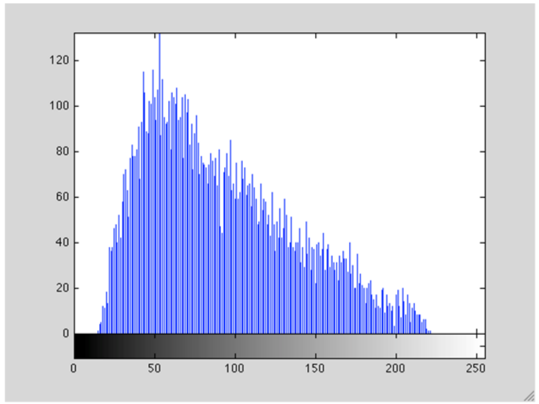
High contrast Image: Grey level values concentrated in a large range

R. Gonzalez & R. Woods, Digital Image Processing, Prentice Hall.

Histogram examples

1.2





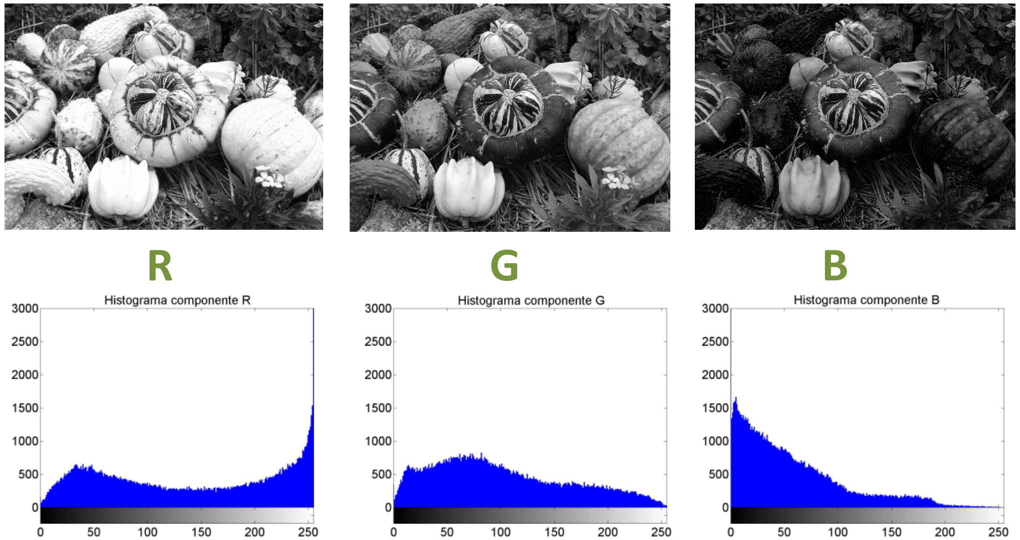
- **Analyze the previous Range Transform operations, knowing now the image histogram**

Color image histogram

The **histogram of a color image** can be defined in several ways:

1. A separate histogram for each component
2. A 3D histogram (joint histogram)
3. A luminance 1D histogram + a joint chrominance 2D histogram

1. A separate **1D histogram for each component**
   - It does not represent the joint probability



Color image histogram

1.2

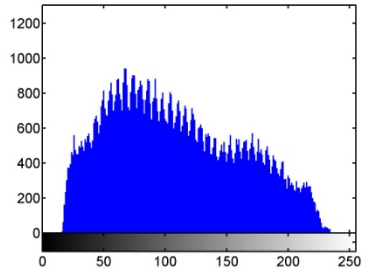
2. A 3D histogram (**joint histogram**):
    - To count all the occurrences of every possible color (c_1, c_2, c_3)
    - A matrix of L^3 elements is created; typically, 256x256x256.

3. A **luminance 1D histogram** + a joint **chrominance 2D histogram**

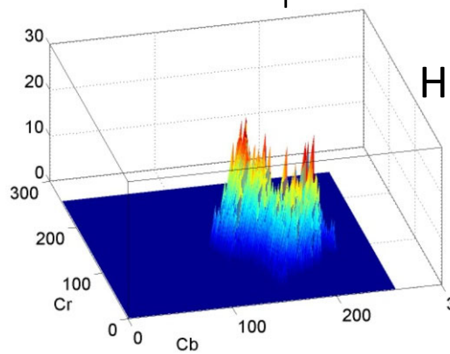
Y

C_b

C_r



Hist. Y



Hist. C_bC_r

Unit Structure

1.2

1. **Introduction**
   - Memoryless processes, Image model definition
2. **Generic operators**
   - Range transform operation
   - Implementation
3. **Histogram based operators**
   - Histogram definition
   - Histogram equalization
4. **Example: Biomedical application**
5. **Summary and Conclusions**

Continuous case

Histogram equalization implements a pixel-based transform aiming at producing a **flat histogram output image**.

- It increases the global contrast of the image
- The transform depends on the input image histogram

In the **continuous case**, we can use the following **change of variable** result:

If x is a continuous variable and $y = g(x)$ is a strictly monotonic function with inverse function $x = h(y)$, then the pdf of $y = g(x)$ is given by:

$$f_y(y) = \left|\frac{dy}{dx}\right|^{-1} f_x(x) = \left|\frac{dg(x)}{dx}\right|^{-1} f_x(x)$$

From continuous to discrete case

1.2

A mapping using the curve of the accumulated probability of x produces an output image **with a uniform pdf** (equalized in the $[0 \ldots 1]$ range).

$$f_y(y) = \left|\frac{dg(x)}{dx}\right|^{-1} f_x(x) \qquad f_y(y) = 1 \;\Rightarrow\; \left|\frac{dg(x)}{dx}\right| = f_x(x) \;\Rightarrow\; y = g(x) = \int_{-\infty}^{x} f_x(w)dw$$

The continuous case mapping has to be **adapted to the discrete case**:

- **Constrain**: Elements having originally the same value (being in the same bin) should receive the same value (be in the same bin) after transformation
- **Note**: Two different input bins can be merged into a single transformed one

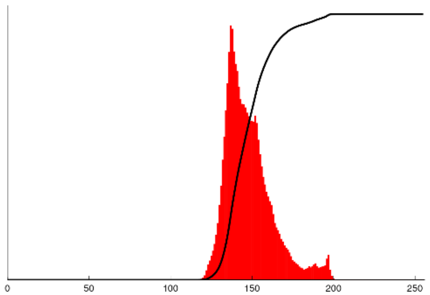
$$s = T(r) = \int_0^r f_r(w)\,dw \quad \Longleftrightarrow \quad s_k = T(r_k) = \sum_{j=0}^{k} p(r_j) = \sum_{j=0}^{k} \frac{n_j}{n}$$

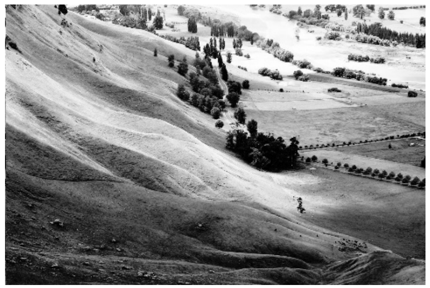
Discrete case

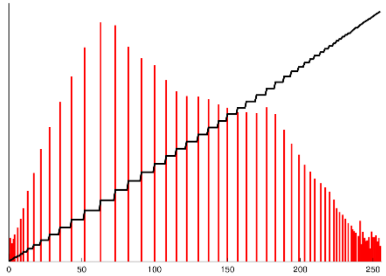
The resulting values (s_k) are defined on the range $[0 \ldots 1]$. In order to have the values in the range $[0 \ldots L\text{-}1]$, they should be scaled and rounded. One possible approach is

$$s_k = T(r_k) = \sum_{j=0}^{k} p(r_j) = \sum_{j=0}^{k} \frac{n_j}{n} \qquad\qquad t_k = round((L-1)\cdot s_k)$$

The final equalization maps all pixels with value r_k into the value t_k.







Red: Histogram
Black: Cumulative Histogram

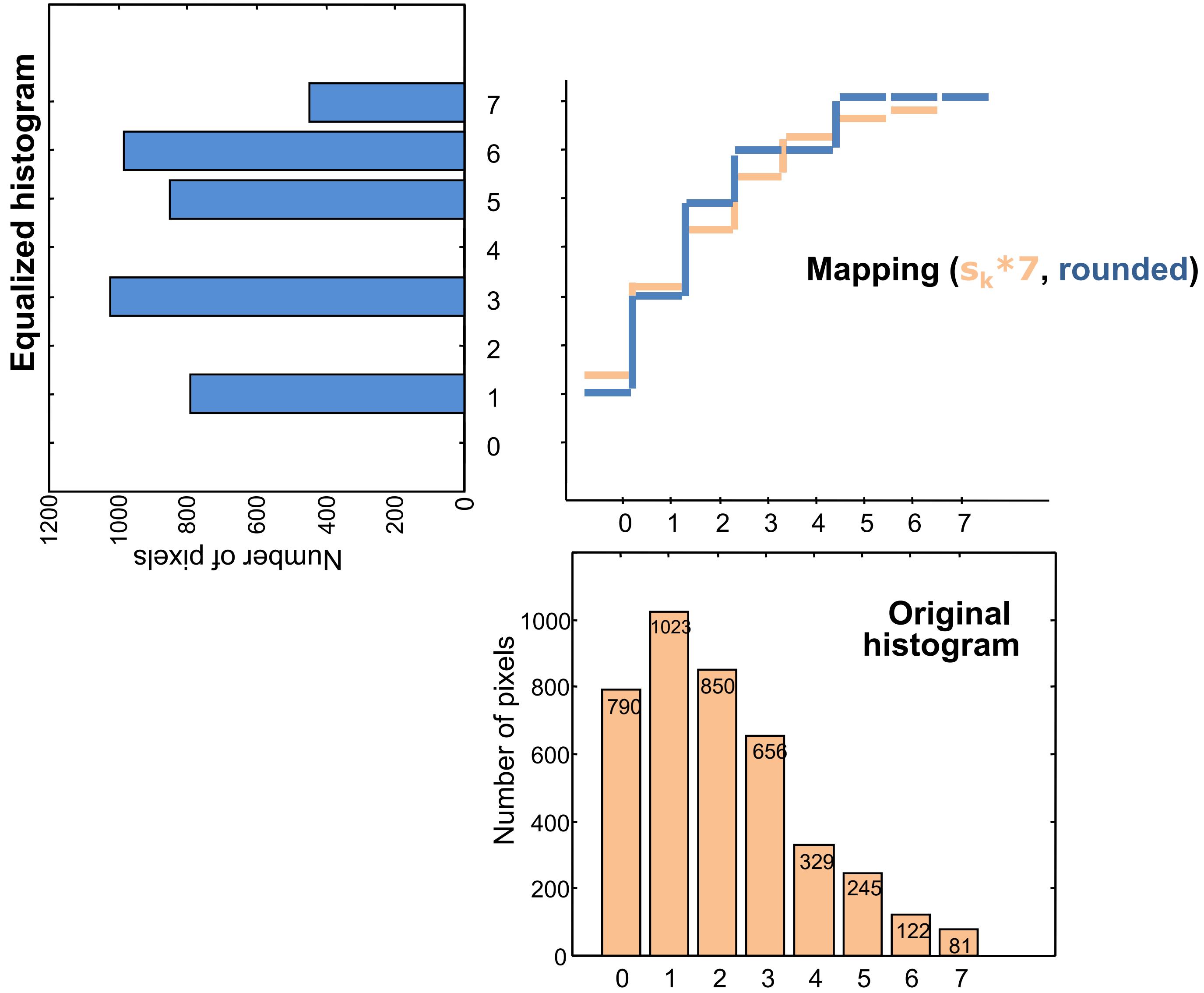
Discrete case: Example

1.2

r_k	Initial level	Number of pixels	$P(r_k)$	Accumulated probability	s_k	s_k*7	Final level	t_k
r_o	0	790	0.19	s_0 = 0.19=	0.19	1,33	1	t_0
r_1	1	1023	0.25	s_1 = 0.19+0.25=	0.44	3,08	3	t_1
r_2	2	850	0.21	s_2 =0.44+0.21=	0.65	4,55	5	t_2
r_3	3	656	0.16	s_3 =0.65+0.16=	0.81	5,67	6	t_3
r_4	4	329	0.08	s_4 =0.81+0.08=	0.89	6,23	6	t_4
r_5	5	245	0.06	s_5 =0.89+0.06=	0.95	6,65	7	t_5
r_6	6	122	0.03	s_6 =0.95+0.03=	0.98	6,86	7	t_6
r_7	7	81	0.02	s_7 =0.98+0.02=	1	7	7	t_7

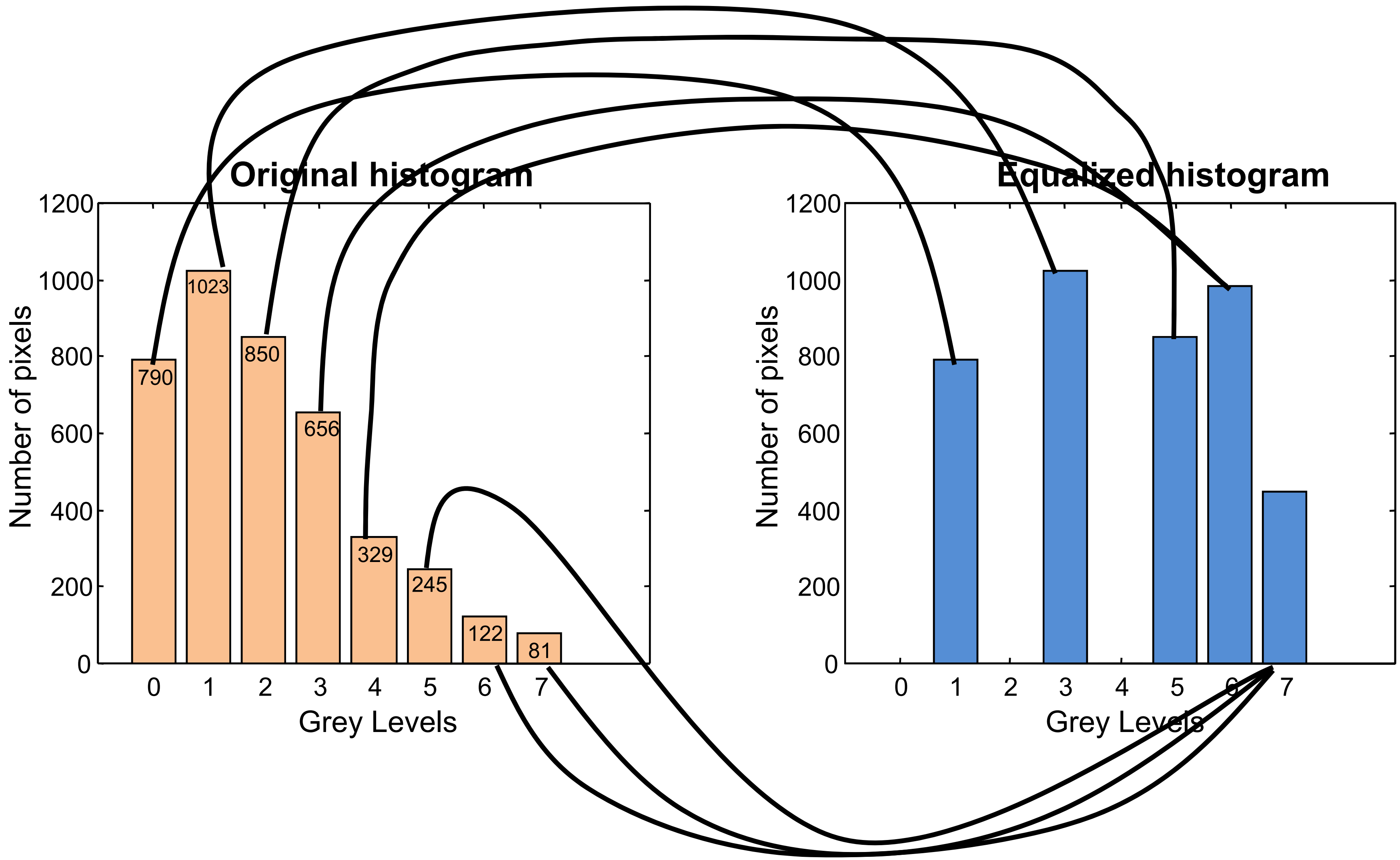
Discrete case: Example

1.2



Discrete case: Example

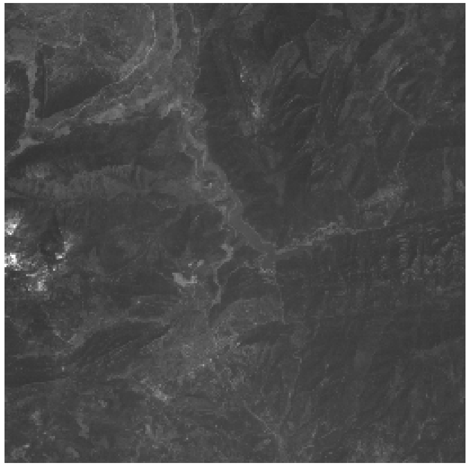
1.2

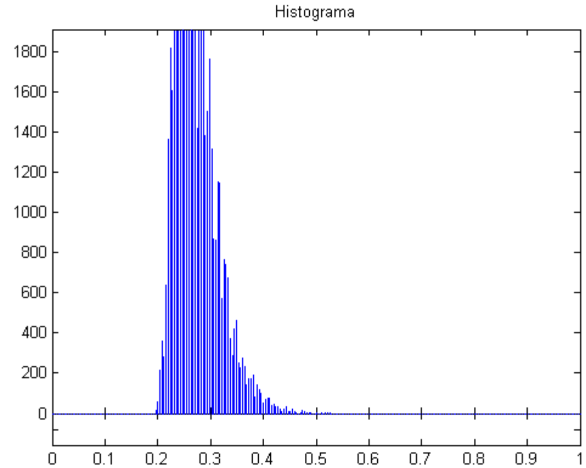


Discrete case: Example

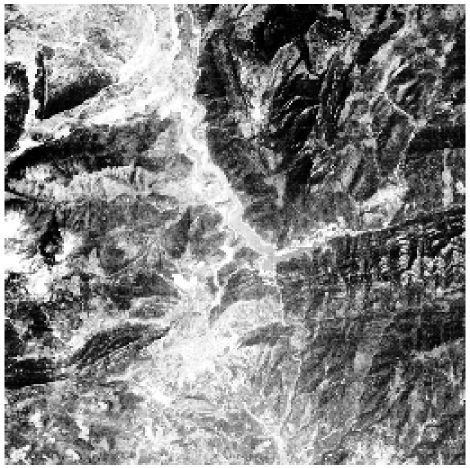
1.2

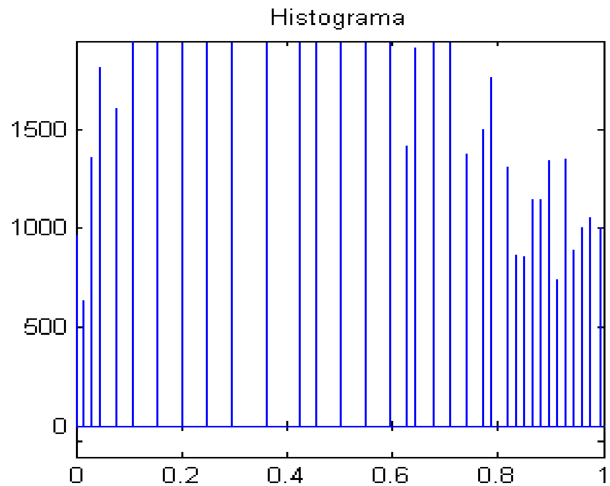
Landsat Image: Original band
Thermal Power Station Cercs
End of activity: 2011 ▶





Landsat Image: Equalized band
Thermal Power Station in Cercs
End of activity: 2011 ▶

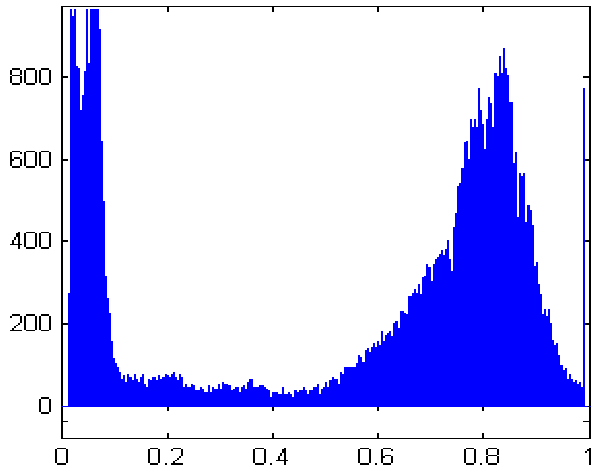


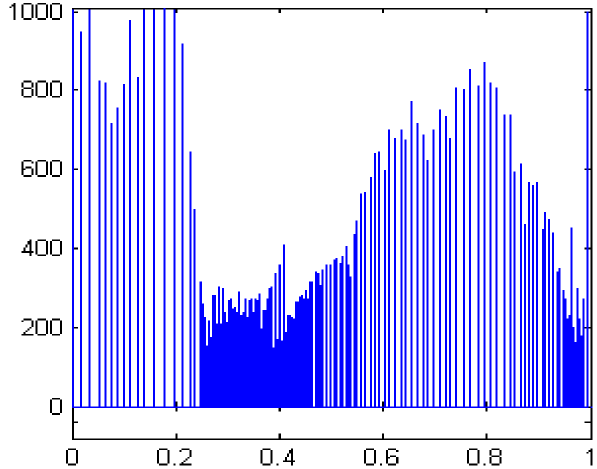


Discrete case: Example

1.2

Cameraman Original Image ▶



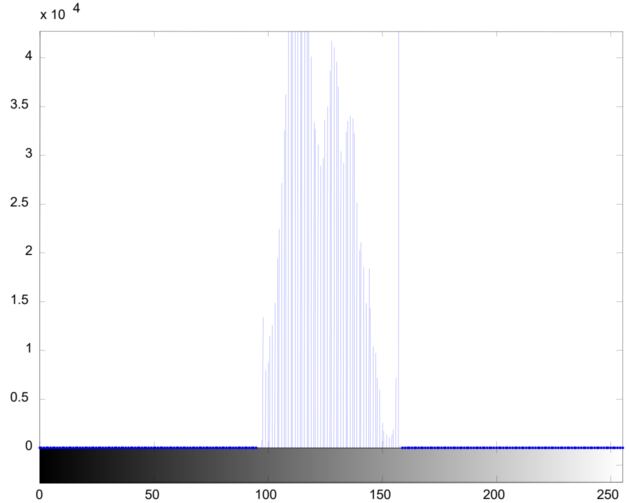


Cameraman Equalized Image ▶

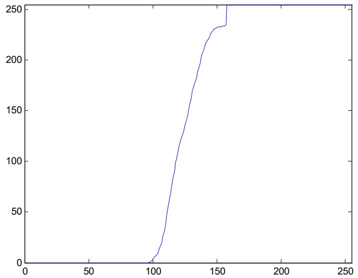
Discrete case: Example

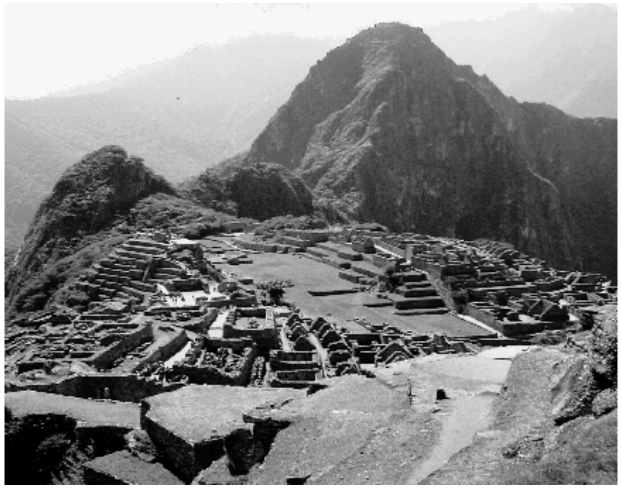
1.2

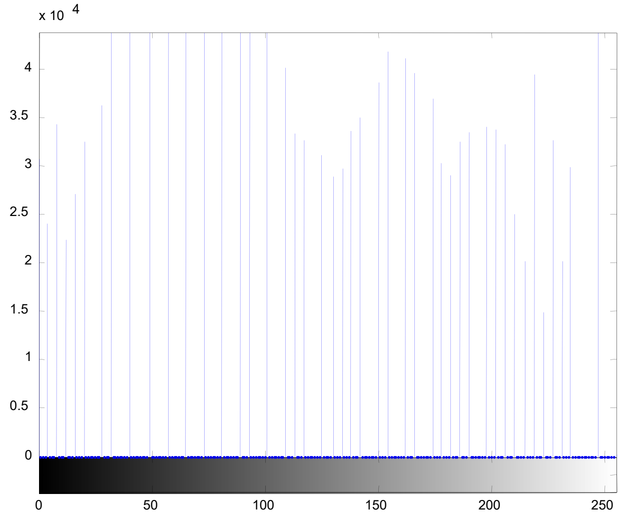
Machu Picchu
Original image ▶



Cumulative Histogram



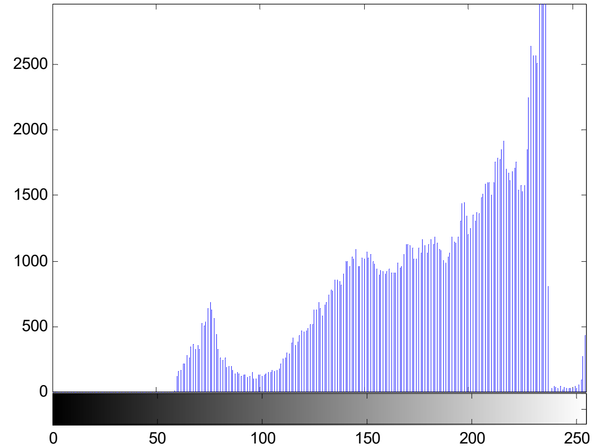




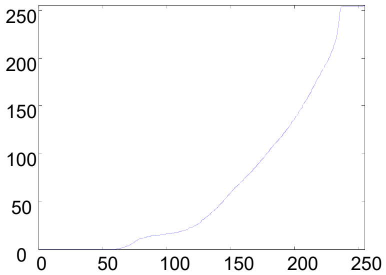
Machu Picchu
Equalized image ▶

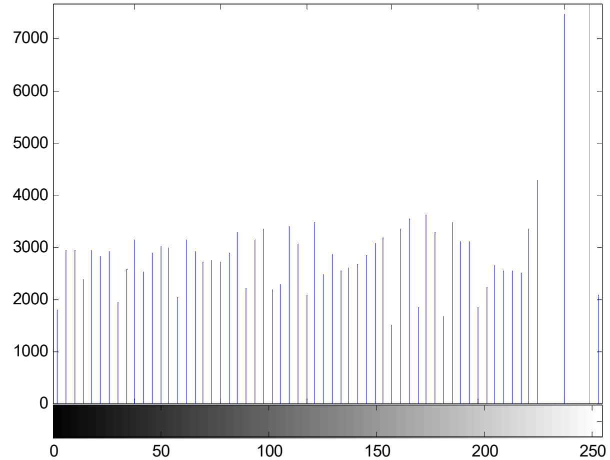
Discrete case: Example

X-Ray Modality
Original image



Cumulative Histogram





X-Ray Modality
Equalized image

Unit Structure

1.2

1. **Introduction**
   - Memoryless processes, Image model definition
2. **Generic operators**
   - Range transform operation
   - Implementation
3. **Histogram based operators**
   - Histogram definition
   - Histogram equalization
4. **Example: Biomedical application**
5. **Summary and Conclusions**

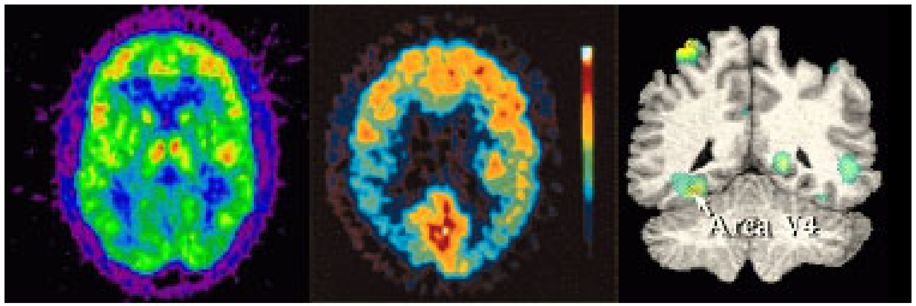
Biomedical Image Modalities

1.2

- Nowadays there exists a **wide range of 3D medical image modalities** that allow neuroscientists to see inside a living human brain.

MR CT Ultrasound

- **Magnetic Resonance**: Based on magnetic water properties. High resolution.
- **Computed Tomography**: X-ray imaging system.
- **Ultrasound:** Sound pressure. Between 2 and 18 MHz. Useful for soft tissues.



PET SPECT fMRI

- **Positron emission tomography:** positron-emitting tracer is injected.
- **Single-photon emission computed tomography:** a gamma-emitting radioisotope is injected
- **Functional MRI:** measure changes in blood flow

Biomedical Image Modalities

1.2

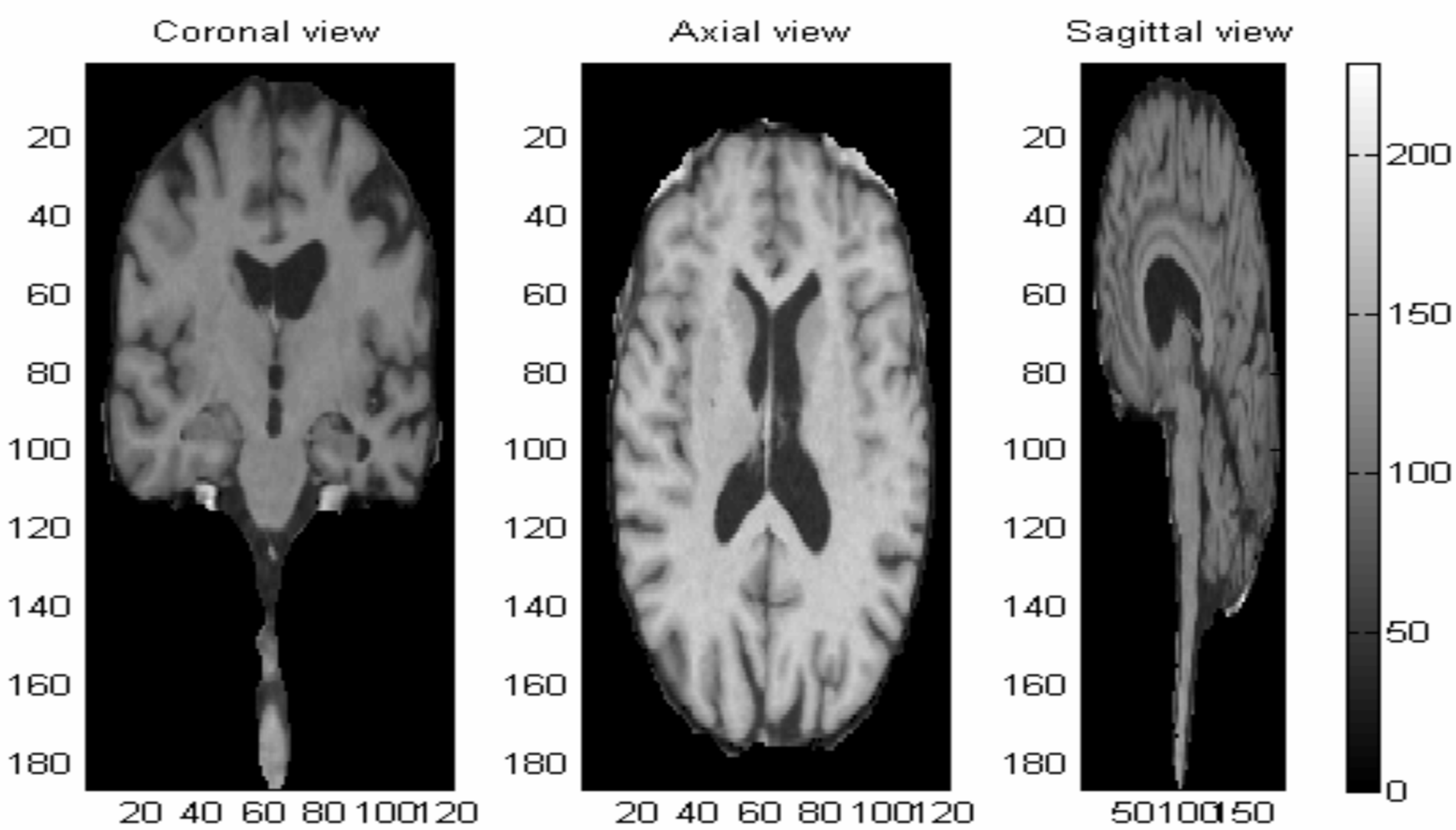
Magnetic Resonance Imaging (MRI) has been increasingly used since its invention in 1973 since it provides **an anatomical view** (no functional) of the tissues, **without the use of ionic radiation.**

Currently, MRI is used both for:

- **Visualization purpose**: to help physicians in their diagnosis or treatment planning.
- **Quantitative analysis**: to obtain automatic estimations of measures and regions of interests.

M. Bach-Cuadra, J-Ph Thiran, F. Marques, "How can physicians quantify brain degeneration?" In T. Dutoit, F. Marques, *Applied Signal Processing: A Matlab-based proof of concept*, Chapter 11, pp. 411-449, Springer, 2009.

MRI: An example

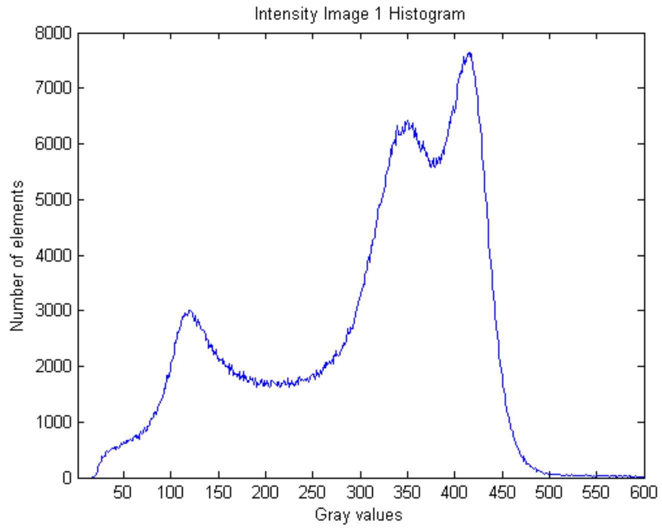


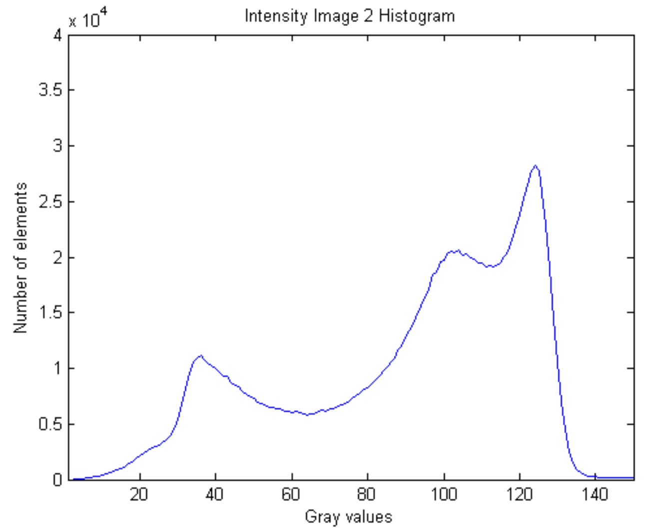
MRI: Problem statement

1.2

The **variability** of Magnetic Resonance Images (MRI) at various time instants can be very high, even in the case of the same person and of the same equipment because of changes in the person posture or the **equipment configuration**. Here we deal with the second problem.

The figures show two histograms of the brain of the same patient taken at two time instants. We assume that **these two histograms represent two extreme cases** of variability in terms of peaks shape and curve smoothness.





MRI: Problem statement

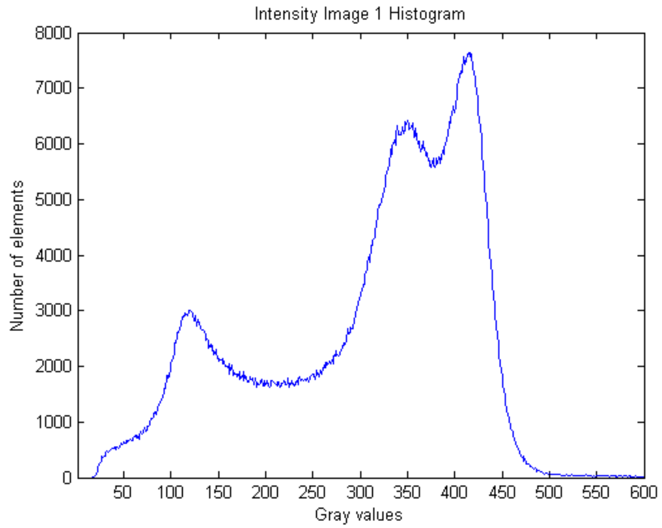
1.2

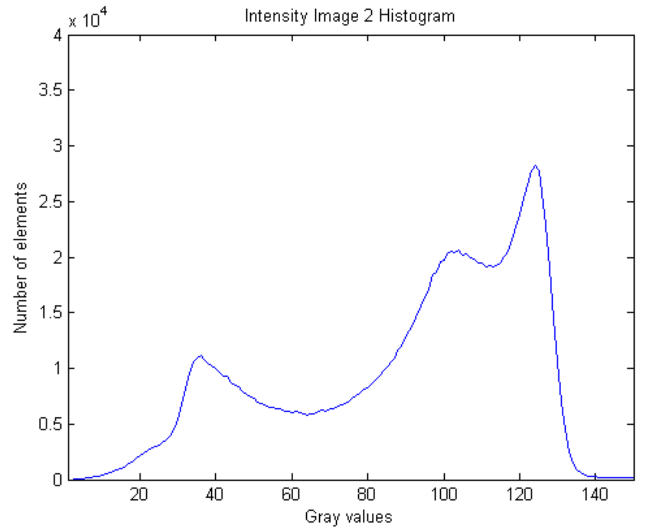
Brain tissues can be classified in **three classes**:

- Cerebrospinal fluid (**CF**),
- Gray matter (**GM**) and
- White matter (**WM**).

Each of these classes is represented by a **gray level distribution** whose height and mean gray level values **may change depending on the system configuration** at the time of acquisition.

The union of these 3 distributions results in **histograms similar to the ones shown**.





MRI: Visualization

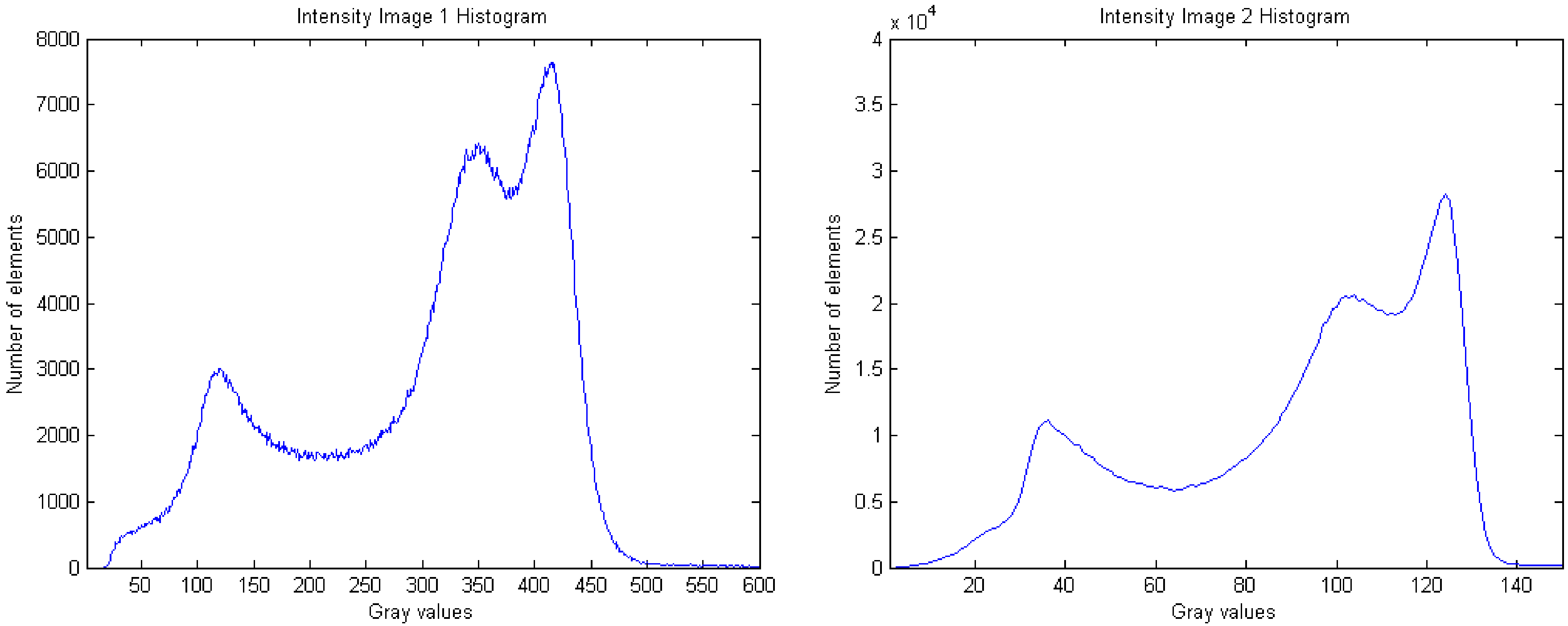
- **First step**: Design a histogram transformation algorithm improving the **visualization** of the 3 tissues. The algorithm should **maximize the contrast** of the transformed image **while preserving possible anomalies** that may be present in the tissues. This algorithm has to be unsupervised (it should not depend on any parameter defined by an operator).

  1. Define the **various steps of this algorithm** and **draw the function** defining the histogram transformation (indicating in particular the values characterizing this transformation).

  2. Apply this algorithm to the two previous histograms : Give the results obtained at each step of the algorithm and **draw the histograms of the resulting transformed images**.

MRI: Visualization

1.2



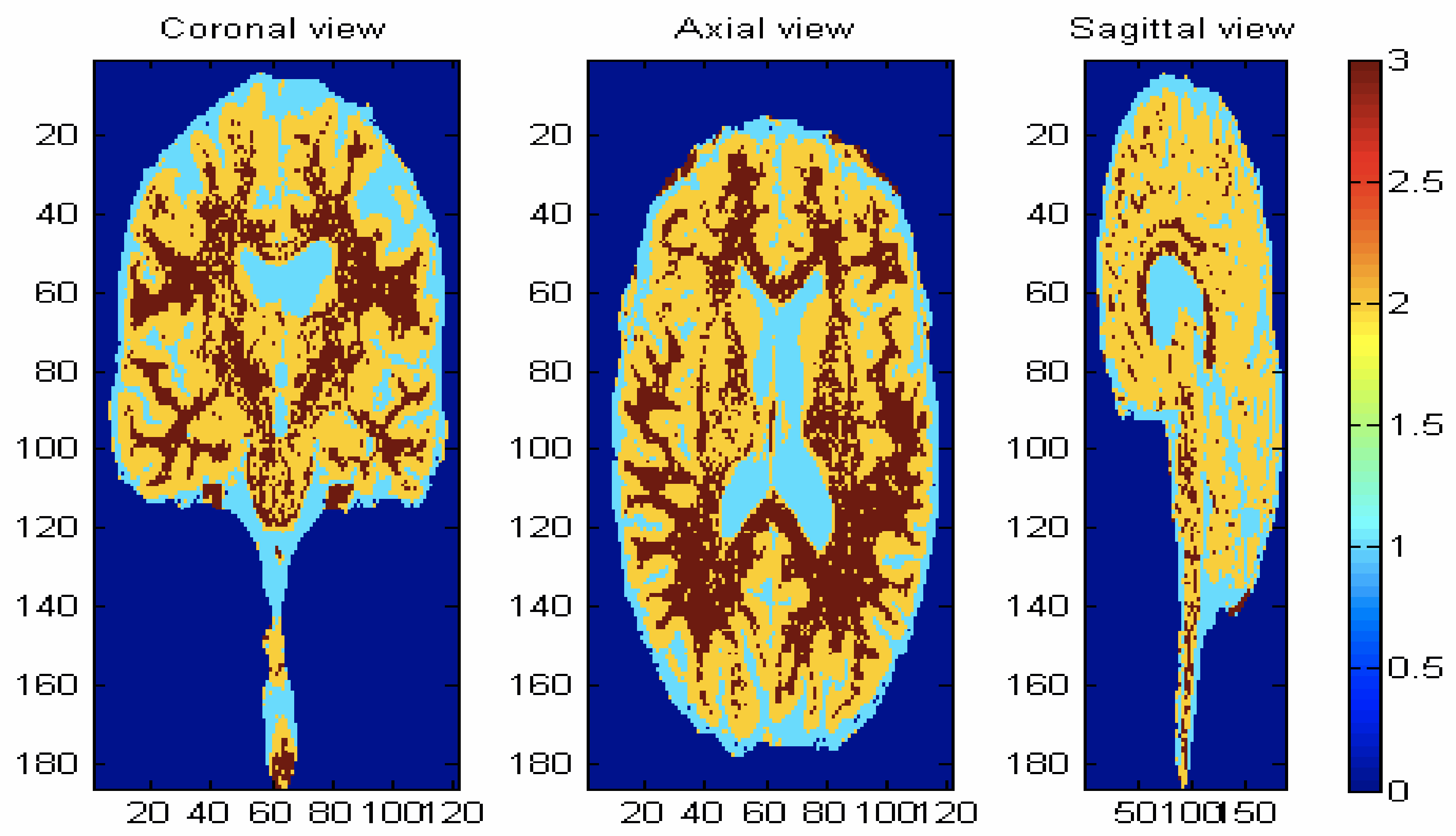
MRI: Quantitative analysis

- **Second step**: perform a **statistical analysis** of the images. To this end, we want to classify the pixels in one of the previously mentioned classes: CF (Peak of lower gray level value), GM (Peak of intermediate gray level value) and WM (Peak of higher gray level value). Here again, we want to define an unsupervised algorithm.

1. Define a **pixel-wise transform** that will allow classifying the input image into these three classes.

2. The result of this type of classification is shown (in pseudo-color) in the following image. Comment
   1. How the pseudo-color has been created
   2. What the **mean weaknesses** of the approach are
   3. Which approach could be used to improve the results.

MRI: Quantitative analysis



Unit Structure

1.2

1. **Introduction**
   - Memoryless processes, Image model definition
2. **Generic operators**
   - Range transform operation
3. **Histogram based operators**
   - Histogram definition
   - Histogram equalization
4. **Example: Biomedical application**
5. **Summary and Conclusions**

Conclusions

1.2

In the **pixel-based image model**, operations only take into account the values of the pixels (**point-wise operators**), but neither their position nor the values of their neighbor pixels.

In range transform operations, a **mapping** ($T(.)$) is defined on the range of values of the input image (r) onto the range of values of the output image (s)

- The mapping **expands/contracts** segments of the input range depending on the magnitude of the derivative of the transform.
- If the mapping is not bijective, it **cannot be inverted**.

The **histogram information** is related to the probability of occurrence of a given value in the image.

- If the histogram of an image is known, **specific transforms** (such as the equalization transform) **can be defined** for that image.