# From Fourier Series to Analysis of Non-stationary Signals – II

Jan Přikryl, Miroslav Vlcek

October 7, 2019

# Contents

# Contents

# Contents

# Contents

# Contents

# Contents

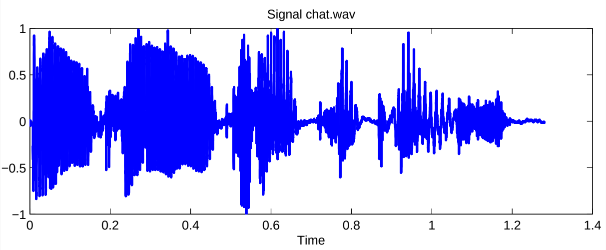
# Signals

We recognize fundamentally 1-dimensional, 2-dimensional, and multidimensional signals.

**1D**

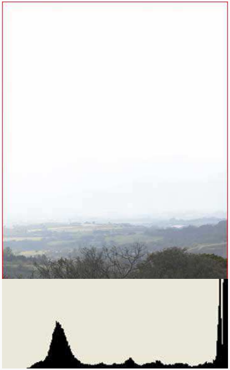
1. a real piano tone A 
2. a speech 

# 1-dimensional signals



# 2-dimensional signals

Histogram is a graph showing the number of pixels in an image at each different intensity value.



# Common Image Processing Problems

# Common Image Processing Problems

- Image restoration and denoising
- Edge detection and denoising
- Image compression

# Image denoising by filter application

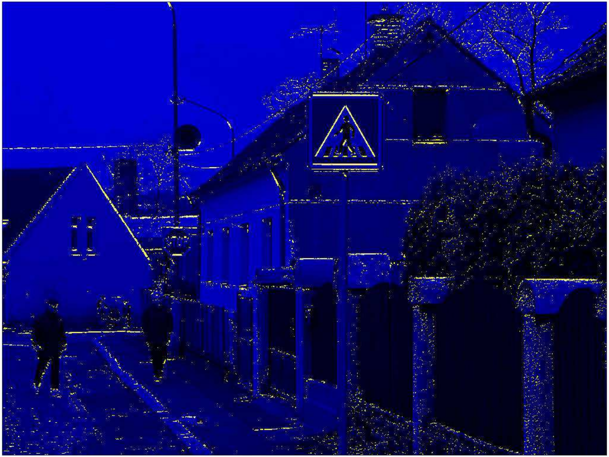
# Image denoising by filter application

# Image restoration and denoising

AKM 24
PZJ 49-49

# Edge detection and denoising

# Edge detection and denoising



# Edge detection and denoising

# Restoration, image denoising

Images can be of poor quality for variety reasons:

- low-quality image capture (security video cameras)
- blurring when the picture is taken
- physical damage to an actual photo
- noise contamination during the image capture process

Restoration seeks to return the image to its original quality.

# Edge detection

The features of interest in an image are the edges, areas of transition that indicate the end of one object and beginning of another.

Applicable in image processing — see Lena[1], or in automated vision and robotics.

[1] Lena Soderberg (*Sjööblom*) 1972

# Compression

Memory requirement for a typical photograph:

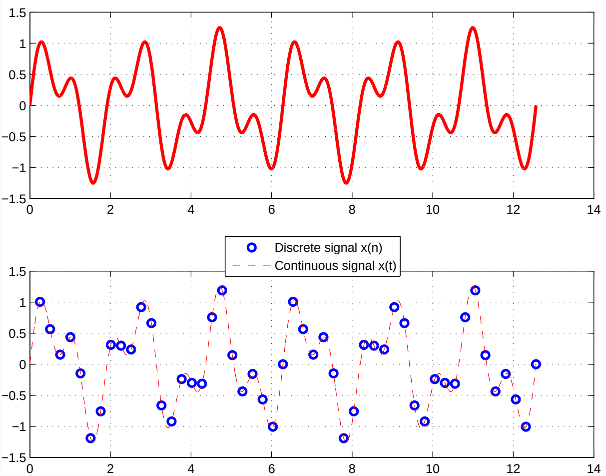
- 24-bit colour $\equiv$ 1 byte for each of the red, green, and blue components
- for $2048 \times 1526$ pixel image we need $2048 \times 1526 \times 3 = 9431040$ bytes
- 9 MB a picture, what can be stored in a 2 GB memory stick ?

Compression algorithms !!! and their drawbacks

# Signals and A to D conversion

- An analog or continuous signal $x(t)$ is a real-valued function of an independent variable $t$ in the definition domain $a \leq t \leq b$; variable $t$ is usually time.
- The function $x(t)$ can represent the intensity of a sound (audio signal), the speed of an object, ...
- For $N \geq 1$ we define the sampling period $T_s = \dfrac{b-a}{N}$, the quantity $f_s = \dfrac{1}{T_s}$ is proportional to number of samples taken during each time period and it is called sampling frequency.
- Finally we obtain digital or sampled signal $x(n) = x(a + n \times T_s)$ for $0 \leq n \leq N$.

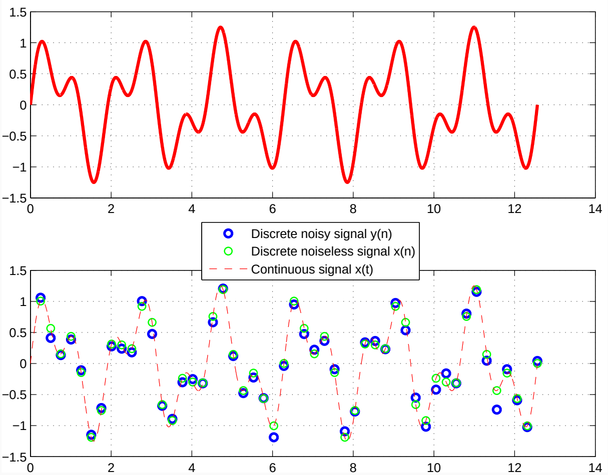
# Analog and Digital Signals



- Sampling is not the only source of error in A/D conversion.
- Consider an analog voltage signal that ranges from 0 to 1 volt.
- An A-to-D converter divides up this 1 volt range into $2^8 = 256$ equally sized intervals.
- the kth interval given by $k\Delta x \leq x < (k+1)\Delta x$ where $\Delta x = 1/256$ and $0 \leq k \leq 255$.
- If a measurement of the analog signal at an instant in time falls within the kth interval, then the A-to-D converter records the voltage at this time as $k\Delta x$.

- This is the quantization step, in which a continuously varying quantity is truncated or rounded to the nearest of a finite set of values $\Rightarrow$ quantization error.
- An A-to-D converter as above would be said to be 8-bit, because each analog measurement is converted into an 8-bit quantity.
- The combination of sampling and quantization allows us to digitize a signal or image, and thereby convert it into a form suitable for computer storage and processing.

# Quantization Error $y(n) = x(n) + \epsilon(n)$



# Vector Space for Signals and Images

# Vector space - revision

A vector space over the real numbers $\mathbb{R}$ is a set $V$ with two operations, vector addition and scalar multiplication, with the properties that

1. for all vectors $\mathbf{u}, \mathbf{v} \in V$ the vector sum $\mathbf{u} + \mathbf{v}$ is defined and lies in $V$(closure under addition);
2. for all $\mathbf{u} \in V$ and scalars $a \in \mathbb{R}$ the scalar multiple $a\mathbf{u}$ is defined and lies in $V$ (closure under scalar multiplication);
3. the *familiar* rules of arithmetic apply, specifically, for all scalars $a, b$ and $\mathbf{u}, \mathbf{v}, \mathbf{w} \in V$:
   a) $\mathbf{u} + \mathbf{v} = \mathbf{v} + \mathbf{u}$, e.g. addition is commutative,
   b) $(\mathbf{u} + \mathbf{v}) + \mathbf{w} = \mathbf{u} + (\mathbf{v} + \mathbf{w})$ e.g. addition is associative,

c) there is a *zero vector* $\mathbf{0}$ such that $\mathbf{u}+\mathbf{0}=\mathbf{0}+\mathbf{u}\equiv\mathbf{u}$ (additive identity),

d) for each $\mathbf{u}\in V$ there is an additive inverse vector $\mathbf{w}$ such that $\mathbf{u}+\mathbf{w}=\mathbf{0}$, we conventionally write $-\mathbf{u}$ for the additive inverse of $\mathbf{u}$,

e) $(ab)\mathbf{u}=a(b\mathbf{u})$,

f) $(a+b)\mathbf{u}=a\mathbf{u}+b\mathbf{u}$,

g) $a(\mathbf{u}+\mathbf{v})=a\mathbf{u}+a\mathbf{v}$.

If we replace $\mathbb{R}$ above by the field of complex numbers $\mathbb{C}$ , then we obtain the definition of a vector space over the complex numbers.

**Example**
The vector space $\mathbb{R}^N$ consists of vectors $\mathbf{x}$ of the form $\mathbf{x} = (x_1, x_2, \ldots, x_N)$ where the $x_k$ are all real numbers. Prove all the properties of the vector space, e.g. multiplication, addition ...

Warning: In later work we will almost always find it convenient to index the components of vectors in $\mathbb{R}^N$ or $\mathbb{C}^N$ starting with index 0, that is, as $\mathbf{x} = (x_0, x_1, ..., x_{N-1})$, rather than the more traditional range 1 to $N$.

**Example**
The sets $\mathbb{M}_{m,n}(\mathbb{R})$ or $\mathbb{M}_{m,n}(\mathbb{C})$ , $m \times n$ matrices with real or complex entries respectively, form vector spaces.

Note:

Any multiplicative properties of matrices are irrelevant in this context!!

The vector space $\mathbb{M}_{m,n}(\mathbb{R})$ is an appropriate model for the discretization of images on a rectangle. Analysis of images is often facilitated by viewing them as members of space $\mathbb{M}_{m,n}(\mathbb{C})$ .

# Discrete Signal Models

| notation | vector space description |
| --- | --- |
| $\mathbb{R}^N$ | $\mathbf{x} = (x_1, \dots, x_N) : x_k \in \mathbb{R}$, finite sampled signals |
| $\mathbb{C}^N$ | $\mathbf{x} = (z_1, \dots, z_N) : z_k \in \mathbb{C}$,<br>analysis of sampled signals |
| $L^\infty(\mathbb{N})$ or $\ell^\infty$ | $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_N) : \mathbf{x}_k = (x_i)_{i \in \mathbb{N}}, x_i \in \mathbb{R}, \forall k : \lvert\mathbf{x}_k\rvert \leq M$<br>bounded, sampled signals, infinite time |
| $L^2(\mathbb{N})$ or $\ell^2$ | $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_N) : \mathbf{x}_k$ dtto, $\sum_k \lvert\mathbf{x}_k\rvert^2 < \infty$<br>sampled signals, finite energy, infinite time |
| $\mathbb{M}_{m,n}(\mathbb{R})$ | Real $m \times n$ matrices, sampled rectangular image |
| $\mathbb{M}_{m,n}(\mathbb{C})$ | Complex $m \times n$ matrices, analysis of images |

# Matlab project

1. Start Matlab and open a new m-file.
2. Consider sampling the function

$$f(t) = \sin(2\pi(440)t)$$

on the interval $0 \leq t < 1$, at 8192 points
**Hint:** Choose sampling interval $\Delta t = 1/8192$ to obtain samples $f(k) = f(k\Delta t) = \sin(2\pi(440)k/8192)$ for $0 \leq k \leq 8191$ . The samples can be arranged in a vector `f`; you can do this in Matlab with

```
f = sin(2*pi * 440 * (0:8191)/8192);
```

**Note:** The sample vector `f` is stored in double precision floating point, about 15 significant digits. However, we'll consider `f` as not yet quantized. That is, the individual components $f(k)$ of `f` can be thought of as real numbers that vary continuously, since 15 digits is pretty close to continuous for our purposes.

Qestions and tasks:

a) What is the frequency in Hertz of the harmonic function $f(t)$?
b) Plot the sampled signal with the command `plot(f)`. It probably doesn't look too good, as it goes up and down 440 times in the plot range. Plot a smaller range, say the first 100 samples.
c) At the sampling rate 8192 Hz, what is the Nyquist frequency? Is the frequency of $f(t)$ above or below the Nyquist frequency?
d) Type `sound(f)` to play the sound out of the computer speaker.
   **Note:** By default, Matlab plays all sound files at 8192 samples per second, and assumes the sampled audio signal is in the range -1 to 1.

ČVUT ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

Consider a second signal

$$g(t) = \sin(2\pi(8632)t) = \sin(2\pi(440 + 8192)t).$$

Repeat parts (a) through (d) from previous part with sampled signal

```
g = sin(2*pi * (440+8192) * (0:8191)/8192);
```

The analog signal $g(t)$ oscillates much faster than $f(t)$, and we could expect it to yield a higher pitch. However, when sampled with $f_s = 8192$ Hz, $f(t)$ and $g(t)$ are aliased and yield precisely the same sampled vectors `f` and `g`. They should sound the same too.

Qestions and tasks:

a) To illustrate the effect of quantization error, construct a 2-bit (4 quantization levels) version of the audio signal $f(t)$. The command

```
qf2 = ceil(2 * (f+1))-1;
```

applied to `f` produces the quantized signal `qf2`.
Sample values of $f(t)$ in the ranges $(-1, -0.5]$, $(-0.5, 0]$, $(0, 0.5]$, and $(0.5, 1]$ are mapped to the integers $0, 1, 2, 3$, respectively.

b) To approximately reconstruct the quantized signal, apply the dequantization formula to reconstruct `f` as `fr2` using

```
fr2 = -1 + 0.5 * (qf2+0.5);
```

This maps the integers $0, 1, 2$ and $3$ to values $-0.75, -0.25, 0.25$, and $0.75$, respectively.

c) Plot the first hundred values of `fr2` with `plot(fr2(1:100))`.

d) Play the quantized signal with `sound(fr2)`.

e) Compute the distortion (as a percentage) between the quantized signal vector $\mathbf{f}$ and the dequantized signal vector $\mathbf{f}_r$ using

$$\epsilon = \frac{\|\mathbf{f} - \mathbf{f}_r\|^2}{\|\mathbf{f}\|^2}$$

The `norm(f)` command in MATLAB computes the standard Euclidean norm of the vector $\|\mathbf{f}\|$.

# Home work

# Assignment 2 – with MATLAB

- Repeat the quantization process from Project 2 using 4-,5-,6-, and 8-bit quantization.

  **Example:** 5-bit quantization is accomplished with

```
qf5 = ceil(16*(f+1))-1}
```

  and dequantization with

```
fr5 = -1 + (qf+0.5)/16
```

  **Hint** for other quantization levels: Note that $16 = 2^{5-1}$.
- Make sure to play the sound in each case.
- Make up a table showing the number of bits in the quantization scheme, the corresponding distortion $\epsilon$, and your subjective rating of the sound quality.
- At what point can your ear no longer distinguish the original audio signal from the quantized version?