What is an Image?

- Today’s lecture provides a mathematical characterization
  — to understand how to represent images digitally
  — to understand how to compute with images

First, we consider the continuous case. This both corresponds to our intuition and simplifies the development of a mathematical theory of image processing.

Continuous Case

- The term “image” suggests a 2D surface whose “appearance” varies from point–to–point
  — the surface typically is a plane (but might be curved)
- Appearance can be B&W or colour
- In B&W, variation in appearance can be described by a single parameter corresponding to the amount of light reaching the image at a given point in a given time
  — this is what, for example, B&W film records

Continuous Case (cont’d)

Denote the image as a function, \( i(x, y) \), where \( x \) and \( y \) are spatial variables

Aside: The convention for this section of the notes is to use lower case letters for the continuous case and upper case letters for the discrete case

Continuous Case (cont’d)

Observations:

- \( i(x, y) \) is a real-valued function of real spatial variables, \( x \) and \( y \)
• \(i(x, y)\) is bounded above and below. That is,

\[
0 \leq i(x, y) \leq M
\]

for some maximum brightness, \(M\)

• \(i(x, y)\) is bounded in spatial extent. That is, \(i(x, y)\) is non-zero (i.e., strictly positive) over, at most, a bounded region

Our intuition is that an image function, \(i(x, y)\), need only be defined over a finite domain. The notion of “bounded spatial extent” captures this intuition. Mathematically, we need to allow both spatial variables, \(x\) and \(y\), to be unbounded. That is, \(-\infty \leq x \leq \infty\) and \(-\infty \leq y \leq \infty\). To handle this, we embed the finite domain of interest into an infinite background, with \(i(x, y) = 0\) in the background. Conceptually, as was done to create the slide, we draw a (rectangular) box such that \(i(x, y) = 0\) outside the box. Note: Of course, there still can be locations within the box at which \(i(x, y) = 0\). The constraint within the box is that \(0 \leq i(x, y) \leq M\).

**Continuous Case (cont’d)**

Some additional considerations:

- Images also can be considered a function of time. Then, we write \(i(x, y, t)\) where \(x\) and \(y\) are spatial variables and \(t\) is a temporal variable
- To make the dependence of brightness on wavelength explicit, we write \(i(x, y, t, \lambda)\) where \(x\), \(y\) and \(t\) are as above and where \(\lambda\) is a spectral variable
- More commonly, we think of “colour” already as discrete and write

\[
\begin{align*}
i_R(x, y) \\
i_G(x, y) \\
i_B(x, y)
\end{align*}
\]

for specified colour channels, R, G and B

Aside: There are times when we also want to consider \(i(x, y)\) to be a function of the state of polarization of the light or, in the case of coherent (laser) optics, a function of the phase of the light. This is beyond the scope of CPSC 505.

**Discrete Case**

* Idea: Superimpose (regular) grid on continuous image
Sample the underlying continuous image according to the *tessellation* imposed by the grid.

**Discrete Case (cont’d)**

Each grid cell is called a picture element (or *pixel*).

There are a finite number of grid lines in each direction.

Without loss of generality, let the grid lines correspond to integer values of $x$ and $y$.

We can refer to the $i,j^{th}$ pixel in the discrete image.

**Discrete Case (cont’d)**

Denote the discrete image as $I(X,Y)$

$I(i,j)$

$I_{i,j}$

*Important:* We can store the pixels in a matrix or array.

Recall: The convention for this section of the notes is to use lower case letters for the continuous case and upper case letters for the discrete case.
Discrete Case (cont’d)

Question: How to sample?

- Sample brightness at point?
- “Average” brightness over entire pixel?

Answer:

- Point sampling is useful for theoretical development
- Area-based sampling occurs in practice

Aside: Typically, one can model area-based sampling as the point sampling of a “blurred” version of the original image

Discrete Case (cont’d)

Question: What about the brightness samples themselves?

Answer: We make the values of \( I(X, Y) \) discrete, as well

Recall: \( 0 \leq i(x, y) \leq M \)

We divide the range \([0, M]\) into a finite number of equivalence classes. This is called quantisation (American: quantization)

The values are called grey-levels (American: gray-levels)

Discrete Case (cont’d)

Quantisation is a topic in its own right

For now, a simple linear scheme is sufficient

Suppose \( n \) bits–per-pixel are available. One can divide the range \([0, M]\) into evenly spaced intervals as follows:

\[
i(x, y) \rightarrow \left\lfloor \frac{i(x, y)}{M} (2^n - 1) + 0.5 \right\rfloor
\]

where \( \lfloor \rfloor \) is floor (i.e., greatest integer less than or equal to)

Typically \( n = 8 \) resulting in grey-levels in the range \([0, 255]\)

Sampling

Sampling Theory (Informal)

Question: When is \( I(X, Y) \) an exact characterization of \( i(x, y) \)?

Question (modified): When can we reconstruct \( i(x, y) \) exactly from \( I(X, Y) \)?
**Intuition:** Reconstruction involves some kind of interpolation

**Heuristic:** When in doubt, consider simple cases

For our “simple cases,” we consider image functions, \( i(x, y) \), where brightness (possibly) varies in the \( x \) direction but is constant in the \( y \) direction.

**Sampling Theory (cont’d)**

*Case 0:* Suppose \( i(x, y) = k \) (with \( k \) one of our grey-levels)

This case is easy! \( I(X, Y) = k \). Any standard interpolation function would give \( i(x, y) = k \) for non-integer \( x \) and \( y \).

**Sampling Theory (cont’d)**

*Case 1:* Suppose \( i(x, y) \) has a discontinuity not falling precisely at integer \( x, y \)

This case is impossible! We can not reconstruct \( i(x, y) \) exactly because we can never know exactly where the discontinuity lies.
Sampling Theory (cont’d)

*Question:* How do we close the gap between “easy” and “impossible?”

*Answer:* Next, we build intuition based on informal argument

*Tell me more!*

(Brief Summary of) Sampling Theory

- Exact reconstruction requires constraint on the rate at which \( i(x, y) \) can change between samples
  - “rate of change” means derivative
  - the formal concept is *bandlimited signal*
  - “bandlimit” and “constraint on derivative” are linked

- Think of music
  - bandlimited if it has some maximum *temporal frequency*
  - the upper limit of human hearing is about 20 kHz

- Think of imaging systems. Resolving power is measured in
  - “line pairs per mm” (for a bar test pattern)
  - “cycles per mm” (for a sine wave test pattern)

- Think of an image
  - bandlimited if it has some maximum *spatial frequency*

(Brief Summary of) Sampling Theory (cont’d)

- The *challenge to intuition* is the fact that music (in the 1D case) and images (in the 2D case) can be represented as linear combinations of individual sine waves of differing frequencies and phases

- A fundamental result (*Sampling Theorem*) is:
  
  For bandlimited signals, if you sample regularly at or above twice the maximum frequency (called the *Nyquist rate*), then you can reconstruct the original signal exactly

(Brief Summary of) Sampling Theory (cont’d)

*Question:* For a bandlimited signal, what if you *oversample* (i.e., sample at greater than the Nyquist rate)

*Answer:* Nothing bad happens!

Samples are redundant and there are wasted bits

*Question:* For a bandlimited signal, what if you *undersample* (i.e., sample at less than the Nyquist rate)
Answer: Two bad things happen!

- Things are missing (i.e., things that should be there aren’t)
- There are artifacts (i.e., things that shouldn’t be there are)

Aside: If a signal is not bandlimited, then it is undersampled at any sampling rate.

**Sampling Theory: Our Take Away Message?**

- We asked, “When is \( I(X, Y) \) an exact characterization of \( i(x, y) \)’’?
- We considered a two stage process:

\[
i(x, y) \quad \text{ sampling } \quad I(X, Y) \quad \text{ interpolation } \quad i(x, y)
\]

and asked, “When is \( \bar{i}(x, y) \equiv i(x, y) \)?

- We use associated formal analysis tools to assess the adequacy of a digital image
- **Bottom line:** We understand how to obtain a digital image as a 2D matrix (or array) of numbers with which we can compute!

In digital image processing, we sometimes use oversampling in order to minimize artifacts produced. Typically, however, one deals with situations in which the underlying continuous image is undersampled. When this is the case, there is a trade-off between “things missing” and “artifacts.” How filter designers make this trade-off depends very much on the application. In medical imaging, for example, one usually wants to maximize information content. Artifacts are tolerated because they are easily recognized as such and thus dismissed by the expert viewer as not medically significant. On the other hand, computer graphics works very hard to minimize artifacts in order to minimize visual distractions. To achieve this, computer graphics necessarily tolerates more information missing.

**A Model of Digital Image Processing**

In digital image processing, the input is a digital image and the output is a digital image. We analyze the computation by asking two questions:

1. What (continuous) image is interpolated?
2. How is that (continuous) image (re)sampled?

**A Model for Digital Image Processing**
A Model for Digital Image Processing (cont’d)

analyze computation on digital image

sampling \( i(x,y) \) \( \rightarrow \) interpolation \( I(X,Y) \) \( \rightarrow \) (re)sampling \( \bar{i}(x,y) \) \( \rightarrow \) \( I(X,Y) \)

adequacy of digital image