Outline:

1. Review:
   - Transforms: CTFT, DTFT, DFT, FFT
   - Sampling and reconstruction (. . . really want to be comfortable with the mathematics now)

   Note: will not review
   - Discrete LTI systems (e.g., z-transforms, convolution)
   - FIR Filter Design

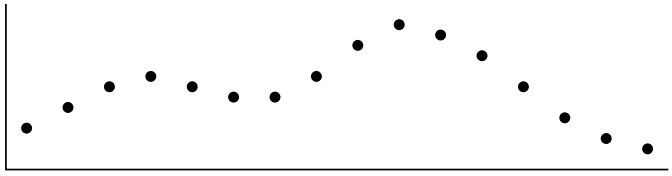
   At that point, I'll expect you to know (most of) chapters 1–7 from

   S. Mitra, *Digital Signal Processing*, 3rd Ed., ©2006

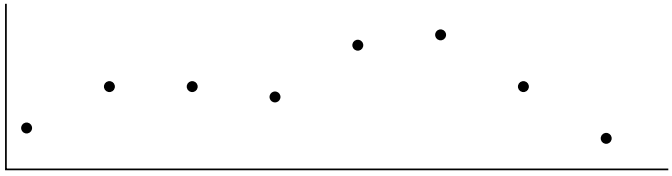
   except for material on, e.g., IIR filters and system stability.

2. Multirate DSP:

- Say that we're given sampled signal

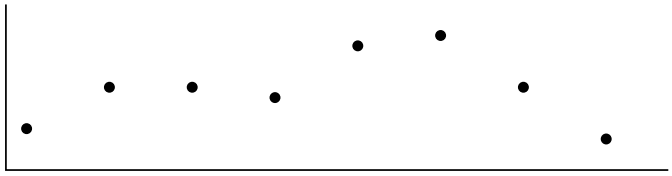


Do we need all these samples? (Are they redundant?)
What if we threw away every other sample?



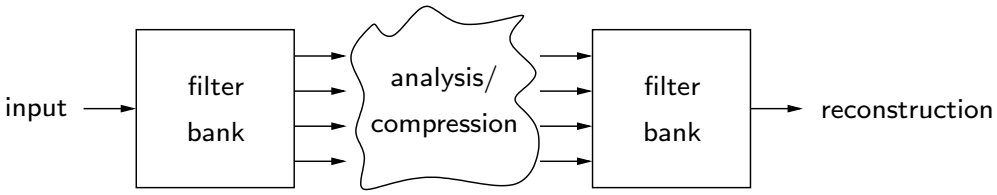
Did we really lose information?
What distortions resulted?

- Say instead that we start with

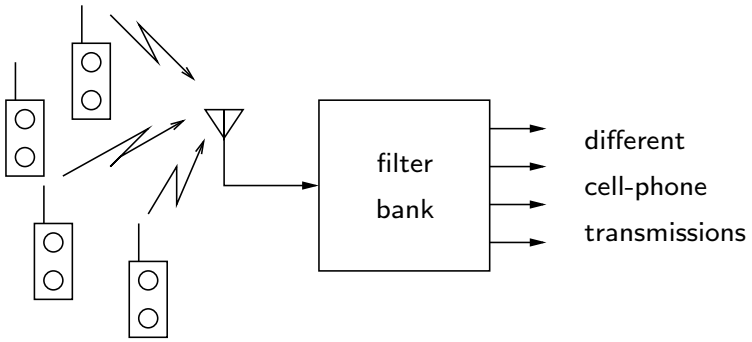


and we want to know "what happened" between the given samples.
When can we know *exactly* what happened?
How can we *efficiently* interpolate these values?

3. Filterbanks:
   - Can we break a signal into "important" / "not-important" components and throw away not-important ones for efficient storage? (Key idea behind MP3, JPEG, MPEG,...)



   - Can we separate a signal into it's "unique components"?

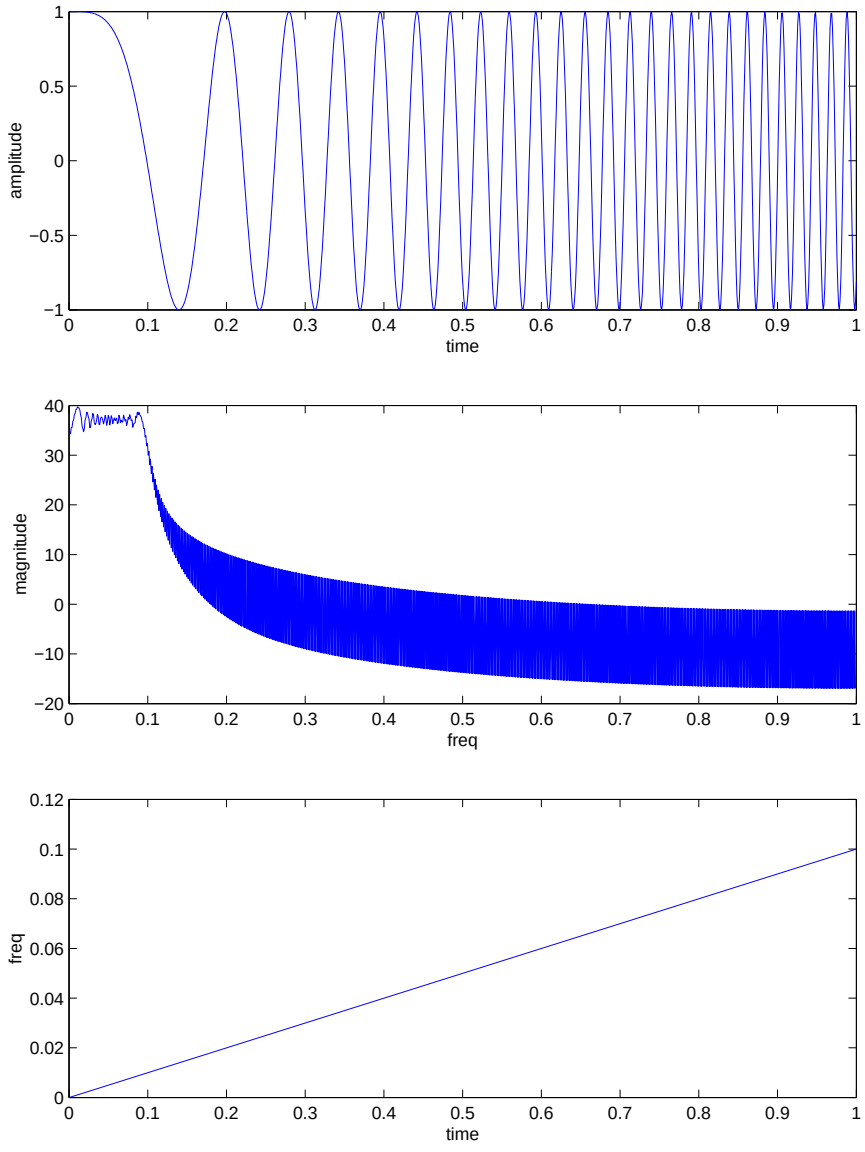


   - How can we build filterbanks to do this efficiently?

4. Joint Time-Frequency Analysis:

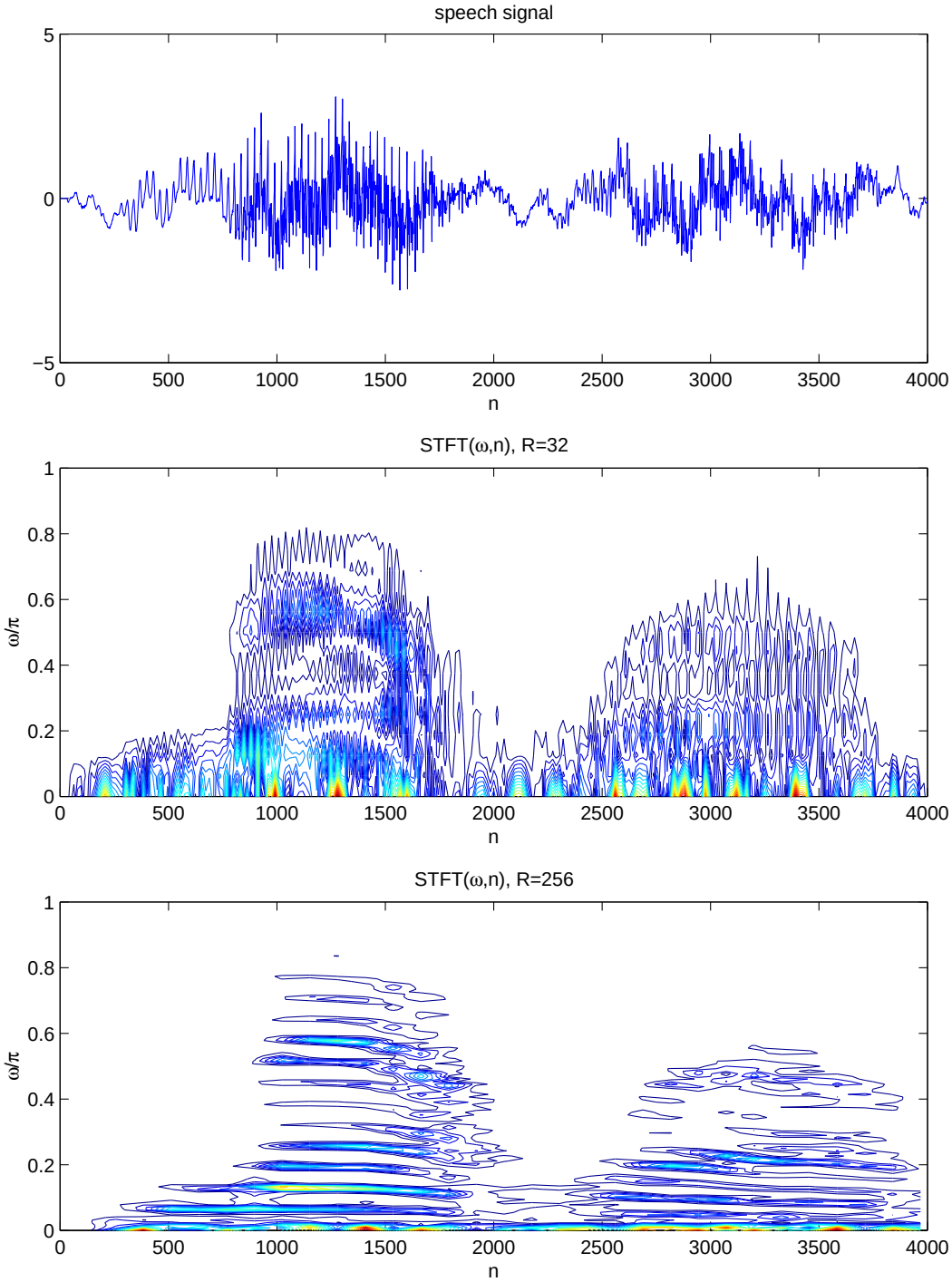
Some signals are difficult to analyze in both the time and frequency domains.

- Linear Chirp:



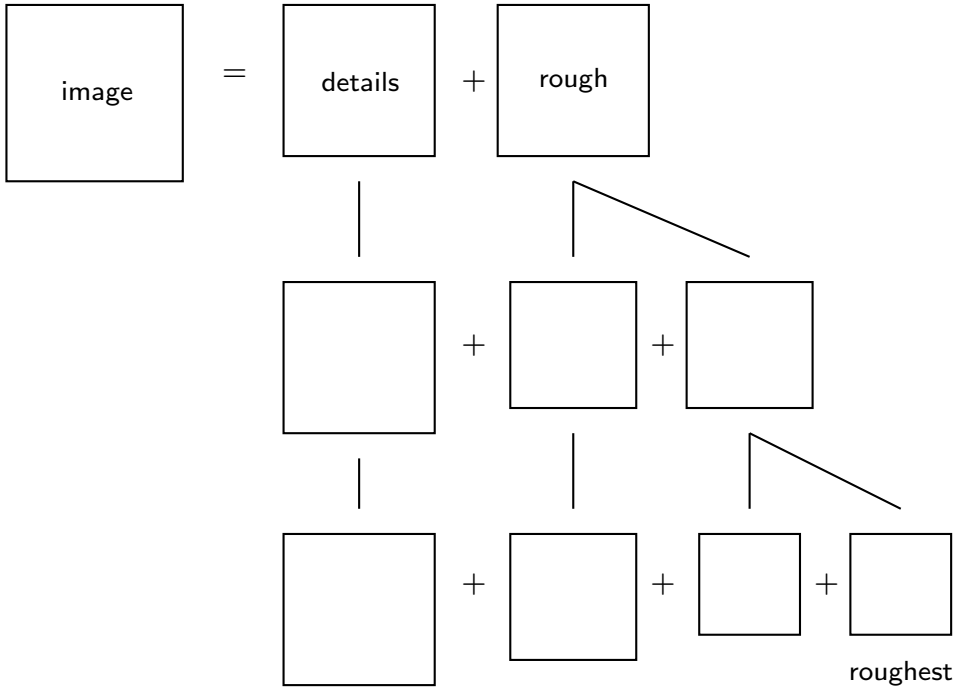
Joint time-frequency analysis clearly explains what is going on!

- Speech Signal:



5. Wavelets / Multiresolution DSP:

- Can we separate a signal into a "rough" component and layers of increasing "detail", so that it can be easily stored at various levels of resolution/quality?



- Each representation contains an equal number of coefficients, so there is no loss in storage efficiency. However, storing $only$ rough portion saves space!
- Convenient for, e.g., progressively downloading images over the web, or searching images in a large database.