

# CS440/ECE448 Lecture 25: Perception

4/23/2018

Mark Hasegawa-Johnson

# Perception

- Web Texts
  - Sentiment Analysis; Information Retrieval; Information Extraction
- Speech and Natural Language Processing
  - Parsing; Machine Translation; Speech Recognition
- Computer Vision
  - Motion Vectors; Object Recognition; Object Localization; image2speech

# Sentiment Analysis

(Textbook section 22.2: Text Classification)

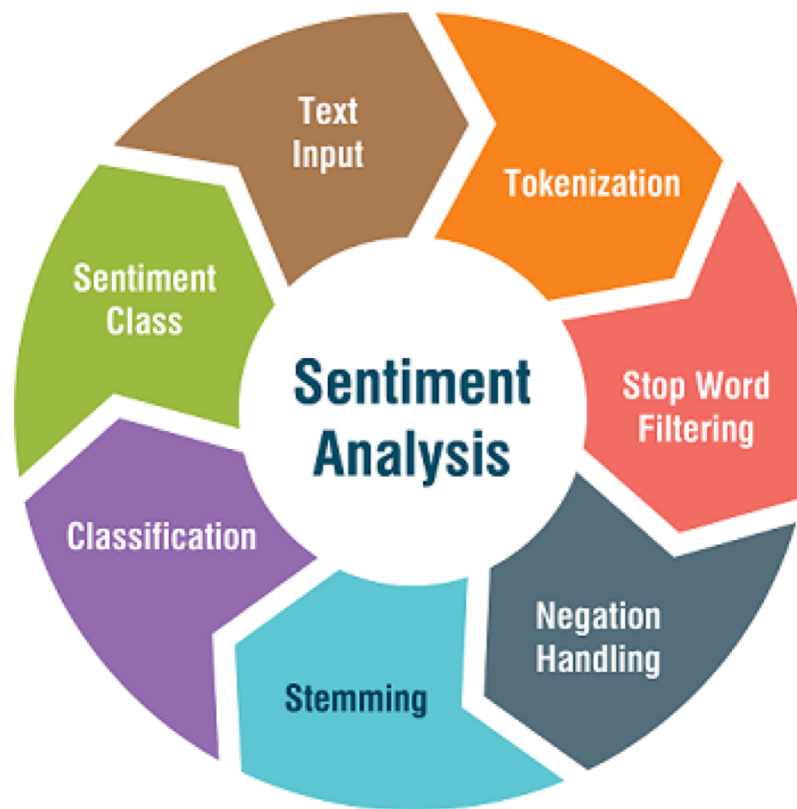


Image source:

John Cawley, Data Analysts, 5/20/2017

<https://www.quora.com/Who-are-the-leading-providers-of-sentiment-analysis-for-social-media-data-and-which-companies-use-them-versus-developing-their-own-technology>

# Sentiment Analysis

- Objective: automatically troll the internet to find out if people like or dislike your product.
- Methods:
  - Recognize keywords
  - Stemming --- convert different morphological forms to the same root
  - Tokenization --- merge phrases like “White House” into single words
  - Partial parsing, to handle negation
- Examples: from Wikipedia
  - Easy: “Pastel-colored 1980s day cruisers from Florida are ugly.”
  - Hard: “I love my mobile but would not recommend it to my colleagues.”



# Sentiment Analysis: an online demo, with partial source code

text-processing.com/demo/sentiment/

[Home](#) [NLTK Demos](#) [NLP APIs](#) [NLTK Models](#) [Contact](#) [StreamHacker Blog](#) [Follow Jacob on twitter](#)

## Sentiment Analysis with Python NLTK Text Classification

This is a demonstration of **sentiment analysis** using a **NLTK 2.0.4** powered **text classification** process. It can tell you whether it thinks the text you enter below expresses **positive sentiment**, **negative sentiment**, or if it's neutral. Using **hierarchical classification**, *neutrality* is determined first, and *sentiment polarity* is determined second, but only if the text is not neutral.

### Analyze Sentiment

Language

english

Enter text

great movie

Enter up to 50000 characters

Analyze

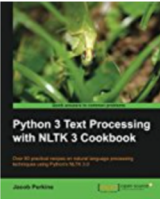
## How Sentiment Analysis with Text Classification Works

The english sentiment uses classifiers trained on both **twitter sentiment** as well as **movie reviews** from the data sets created by [Bo Pang and Lillian Lee](#) using [nlk-trainer](#) (also on [bitbucket](#)). The dutch sentiment is based on book reviews.

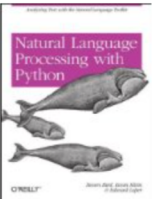
The results will be **more accurate on text that is similar to original training data**. If you get an odd result, it could be the words you've used are unrecognized. Try entering more words to improve accuracy.

## Sentiment Analysis Articles


To read more about how it works, please read the following articles I've written about the process:



Python 3 Text Processing with NLTK 3 Cookbook



Natural Language Processing with Python



Bad Data Handbook

# Perception

- Web Texts
  - Sentiment Analysis; Information Retrieval; Information Extraction
- Speech and Natural Language Processing
  - Parsing; Machine Translation; Speech Recognition
- Computer Vision
  - Optical Flow; Object Recognition; Object Detection; image2speech

# Information Retrieval

Textbook section 22.3; CS 410, 510

- Given:
  - A corpus of documents
  - A query posed in some query language
- Generate:
  - A list of results, usually rank-ordered
- In order to maximize:
  - Some measure of utility

# IR Measures of Utility

- Is there any relevant document on the first page?
  - Precision at N = (# correct documents in the first N)/N
- How many of the target documents did I get?
  - Recall at N = (# correct documents in first N)/(# correct in the database)
- How far do I have to search in order to find the correct document?
  - Expected reciprocal rank =  $E [ 1/n ]$ , where n = rank of the correct document

# TF-IDF (term freq -inverse document freq)

Karen Sparck Jones, 1972

- Given a query,  $q$ , containing terms,  $t$
- Find a document  $d$  that maximizes

$$tf \cdot idf(q, d) = \sum_{t \in q} tf(t, d)idf(t)$$

- TF (term frequency): how often does term “ $t$ ” occur in document “ $q$ ”?
- IDF (inverse document frequency): reduce the importance of the term “ $t$ ” if it occurs frequently across all documents, example,

$$idf(t) = \log \frac{(\# \text{ documents in database})}{(\# \text{ documents containing term } t)}$$

# PageRank

(Brin and Page, 1998)

$$PR(p) = \frac{1-d}{N} + d \sum_{i:p \in C(i)} \frac{PR(i)}{|C(i)|}$$

- $d$  (damping) = probability that the surfer continues browsing links, versus restarting with a new search
- $C(i)$  = set of outgoing links from page  $i$
- $PR(i)$  = page-rank = probability that the surfer is on page  $i$
- Significance: google's first search algorithm (1997)
- Significance: must be estimated using expectation-maximization

# Perception

- Web Texts
  - Sentiment Analysis; Information Retrieval; Information Extraction
- Speech and Natural Language Processing
  - Parsing; Machine Translation; Speech Recognition
- Computer Vision
  - Motion Vectors; Object Recognition; Object Detection; image2speech

# Information Extraction

Text Section 22.4; CS 412, 512

- Pattern recognition
- Ontology extraction
- Attribute extraction



# Pattern Matching in text: Regular Expressions

- Invented by Stephen Cole Kleene in the 1950s
- Just three basic operators:
  - + --- union of two sub-languages
  - x --- concatenation of two sub-languages
  - \* --- zero or more repetitions of a sub-language
- This expression contains (two|several) (very)\* useful examples.
  - This expression contains several very useful examples.
  - This expression contains several very very useful examples.
  - This expression contains several very very very useful examples.
  - This expression contains two useful examples.

# Ontology Extraction: Entity Discovery and Linking

*Slide credit: Heng Ji*

13岁以前的**杨丽萍**，是云南一个山村小镇里光着脚丫到处拾麦穗的乡下**小姑娘**，在洱海之源过着艰苦而又不无乐趣的童年生活。

Now, Ms. **Yang**, **one** of **China's** best-known dancers, is the **director**, **choreographer** and **star** of ...

Aunque nacida en **Dali**, a la edad de nueve años **Yang** se mudó con su familia a **Xishuangbanna**. Debido a su extraordinario talento, la eligieron para integrar la Agrupación Artística de Canto ...

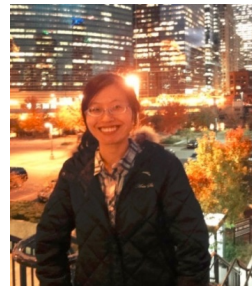
...

**KB**

Yang Liping	
Traditional Chinese	楊麗萍
Simplified Chinese	杨丽萍
Transcriptions	[show]



Liping Yang



Liping Yang



- Cross-lingual knowledge fusion: For certain entities and events, new and detailed information is only available in low-resource foreign incident languages
- Cross-lingual Knowledge transfer: Build cross-lingual links to transfer resources (e.g., annotated data, gazetteers and rich knowledge representations) from English to foreign language EDL

# Attribute Extraction: Combine with Tri-lingual Slot Filling

- Slide Credit: Heng Ji



Each query = an entity cluster of multi-lingual mentions, with type, KB and each mention's Document ID, offsets

Yang Liping	
Traditional Chinese	楊麗萍
Simplified Chinese	杨丽萍
Transcriptions	<a href="#">[show]</a>



## Source Collection

State/Province-of-Residence: 云南

13岁以前的杨丽萍，是云南一个山村小镇里光着脚丫到处拾麦穗的乡下小姑娘，在洱海之源过着艰苦而又不无乐趣的童年生活。十几年后，她摇身一变，成为舞台上最绚丽的“孔雀”...而关于杨丽萍的感情问题，曾经有个爆料人称，杨丽萍的前夫是中央民族歌舞团里的才子，一直帮着杨丽萍策划舞蹈，但后来，一个叫做托尼的美籍台湾人(刘淳晴)出现后，把杨丽萍给撬走了。

Spouse: 刘淳晴

Title: dancer, director, choreographer

Now, Ms. Yang, one of China's best-known dancers, is the director, choreographer and star of a new show that is drawing sellout crowds all over the country.

# Perception

- Web Texts
  - Sentiment Analysis; Information Retrieval; Information Extraction
- Speech and Natural Language Processing
  - Parsing; Machine Translation; Speech Recognition
- Computer Vision
  - Motion Vectors; Object Recognition; Object Detection; image2speech

# Morphology: What is a word?

- Most morphological rules can be written as Regular Expressions (e.g., English pluralization), but some are less regular than others:

unhappy \*unsad

unhealthy \*unill

unclean \*undirty

(example: Harald Trost)

- The rules, and the exceptions, are usually learned by expectation maximization from dictionaries, e.g.,

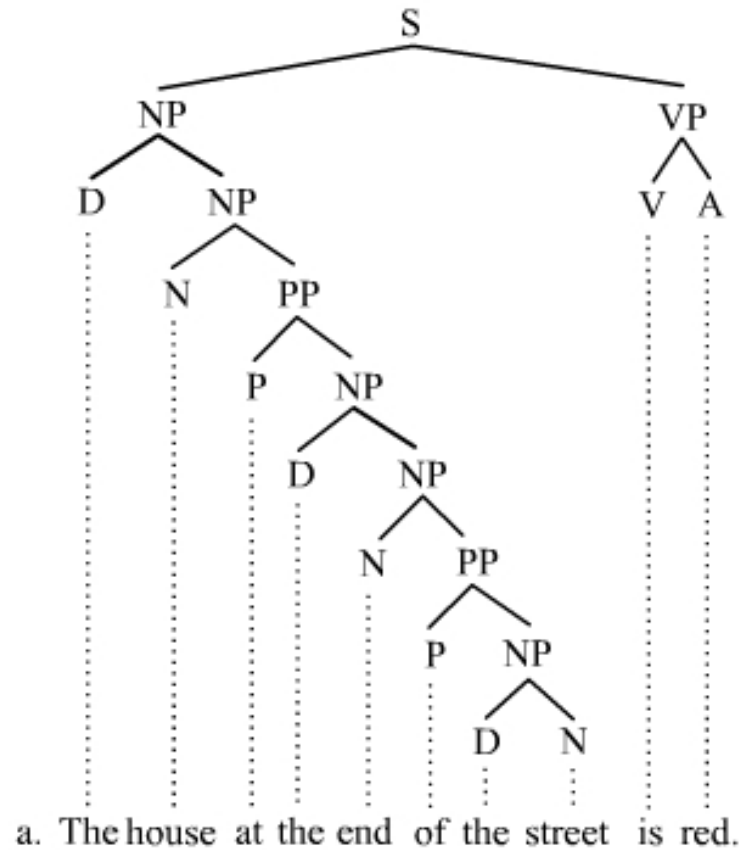
<https://github.com/AdolfVonKleist/Phonetisaurus>

- ...but in some interesting & influential cases, large parsers are constructed by hand, e.g., <https://catalog.ldc.upenn.edu/ldc2004l02>

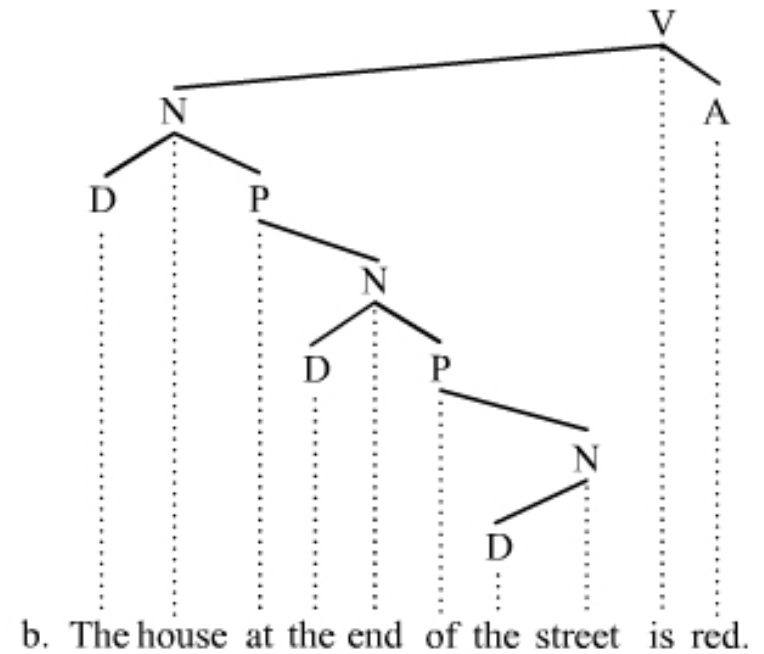
# Syntax: What is a sentence?

Textbook section 23.1-3

By Tjo3ya - Own work, CC BY-SA 3.0,  
<https://commons.wikimedia.org/w/index.php?curid=18436919>



**Constituency structure**



**Dependency structure**

# Syntax example: the Stanford Parser

## Stanford Parser

Please enter a sentence to be parsed:

My dog also likes eating sausage.

Language: English

[Sample Sentence](#)

Parse

### Your query

*My dog also likes eating sausage.*

### Tagging

My/PRP\$ dog/NN also/RB likes/VBZ eating/VBG sausage/NN ./.

### Parse

```
(ROOT
  (S
    (NP (PRP$ My) (NN dog))
    (ADVP (RB also))
    (VP (VBZ likes)
      (S
        (VP (VBG eating)
          (NP (NN sausage)))))
    (. .)))
```

## Parsing: UIUC courses

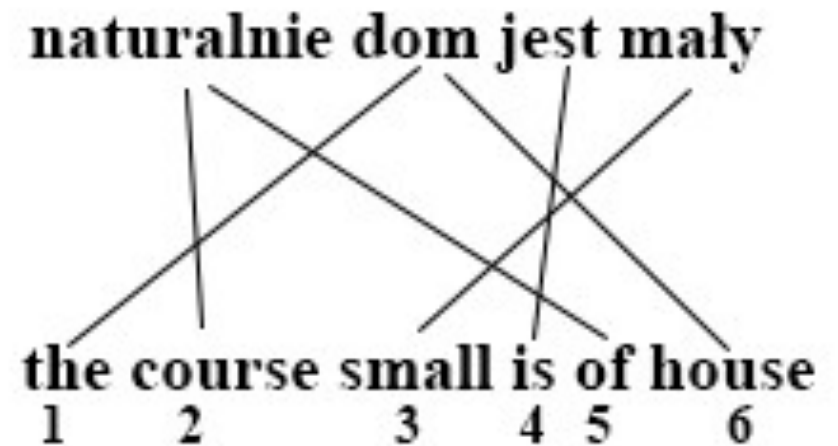
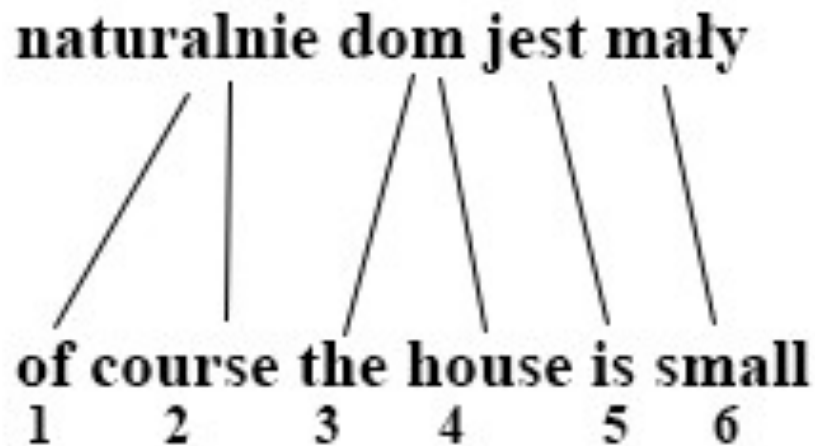
- CS 447, Natural Language Processing
- CS 546, Machine Learning in Natural Language Processing
- LING 406, Computational Linguistics
- LING 506, Computational Linguistics



# Machine Translation

Textbook section 23.4; UIUC course LING 415, Machine Translation

- Includes two processes: word reordering + word translation



By Krz.wolk - Own work, CC BY-SA 4.0,  
<https://commons.wikimedia.org/w/index.php?curid=44522757>

# Perception

- Web Texts
  - Sentiment Analysis; Information Retrieval; Information Extraction
- Speech and Natural Language Processing
  - Parsing; Machine Translation; Speech Recognition
- Computer Vision
  - Motion Vectors; Object Recognition; Object Detection; image2speech

# 3 steps to produce sounds

Slide credit: Odette Scharenborg

step 3: *articulation* =

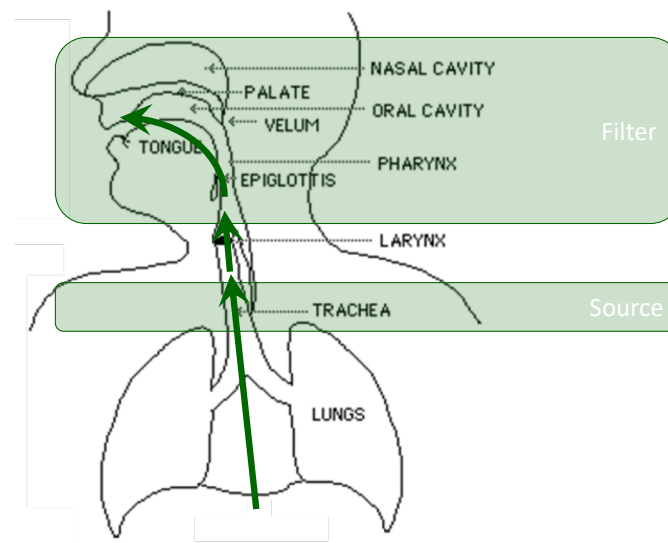
distortion of air

→ time-varying formant-frequency  
pattern

= speech

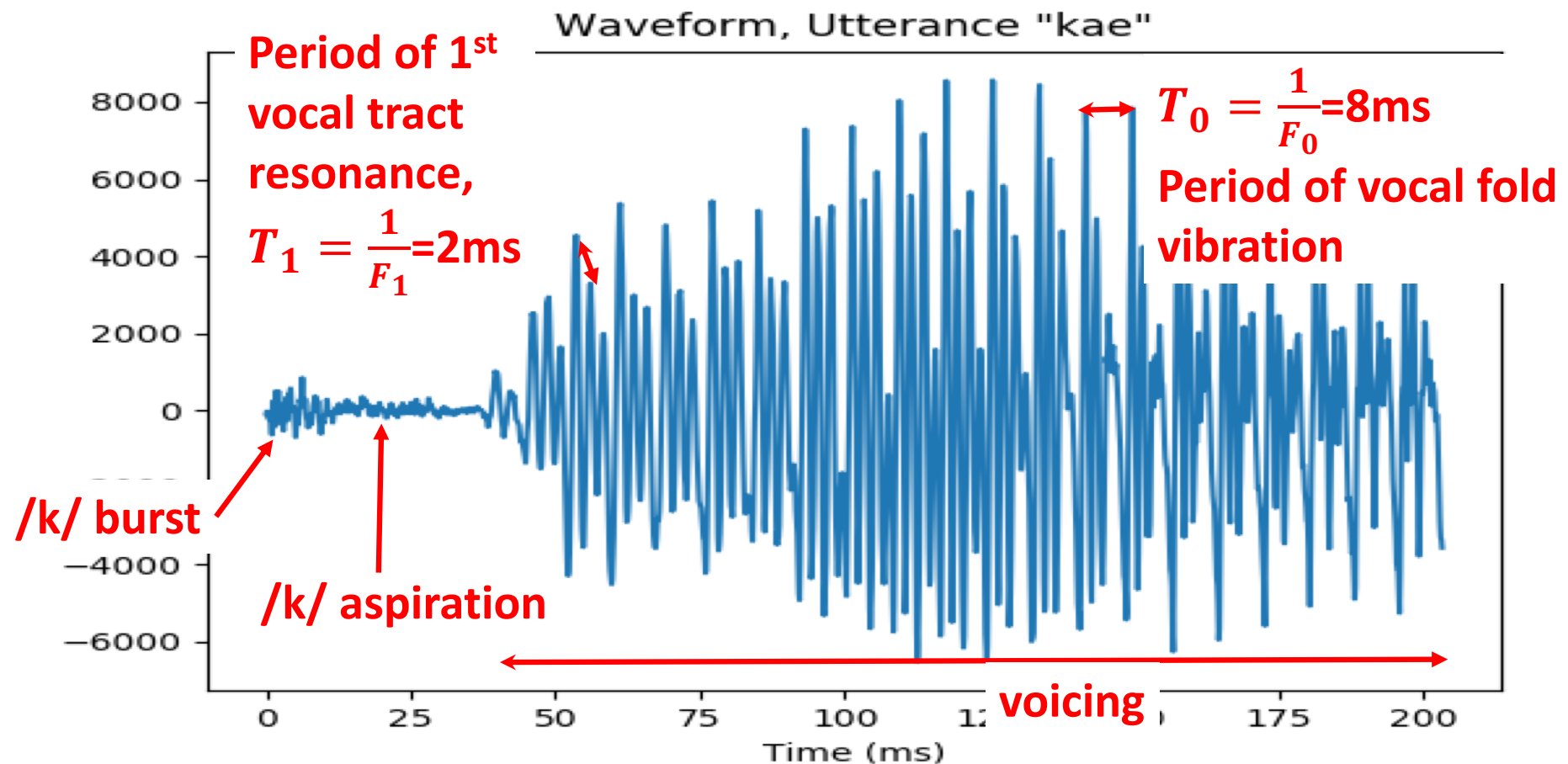
step 2: *phonation*

step 1: *initiation*



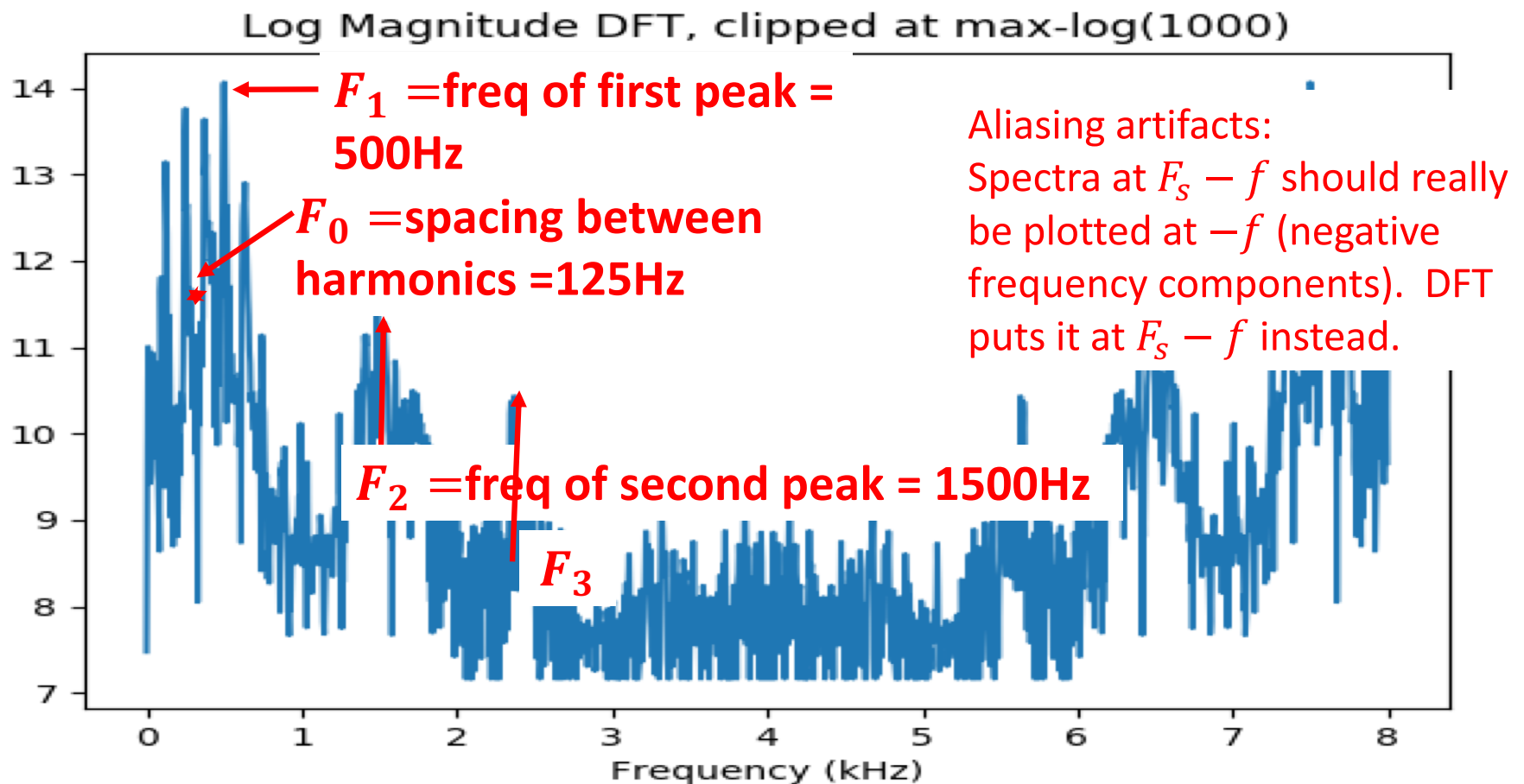
# Speech signal: Time domain

Slide credit: ECE 417



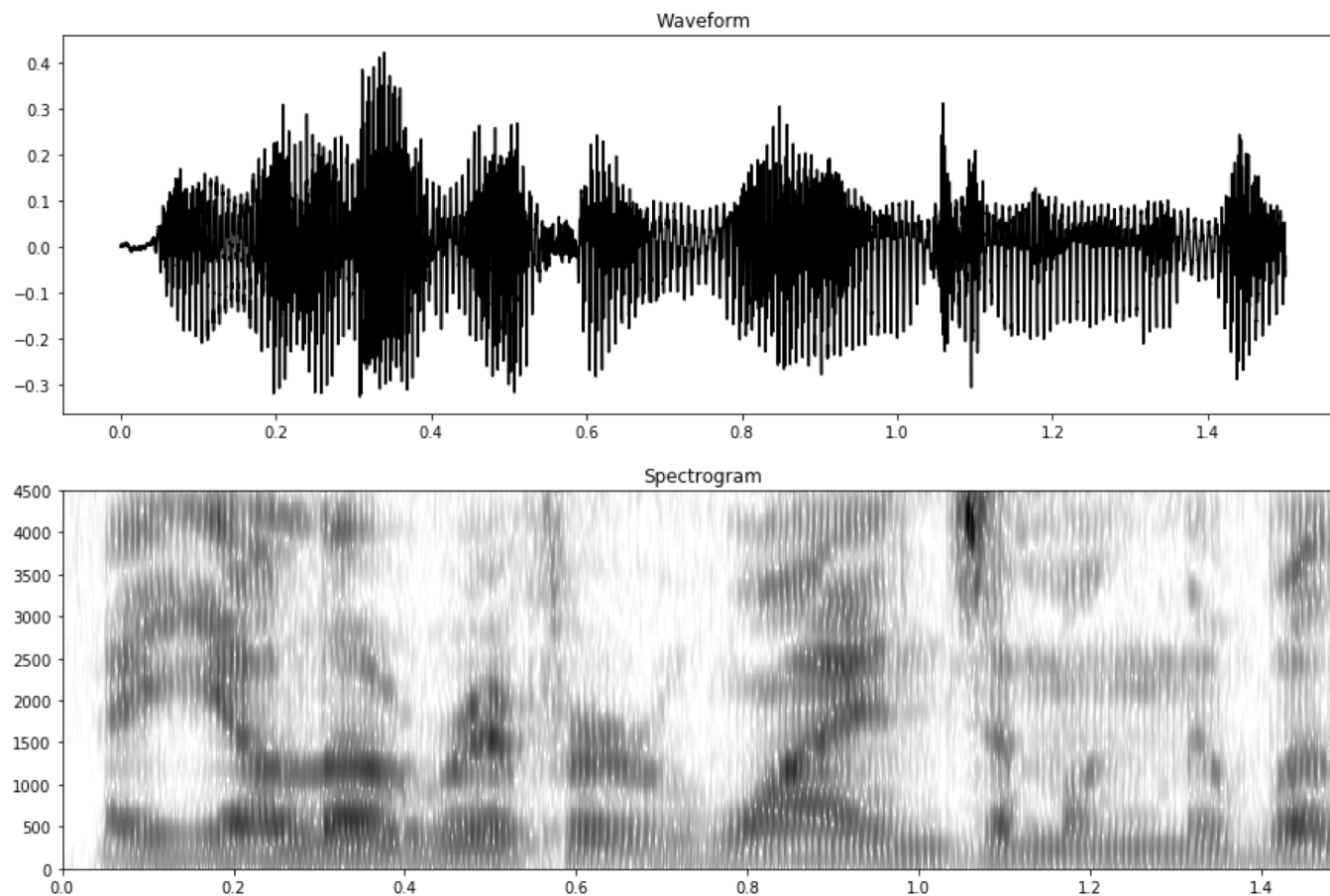
# Speech signal: Log Magnitude Transform

Slide credit: ECE 417

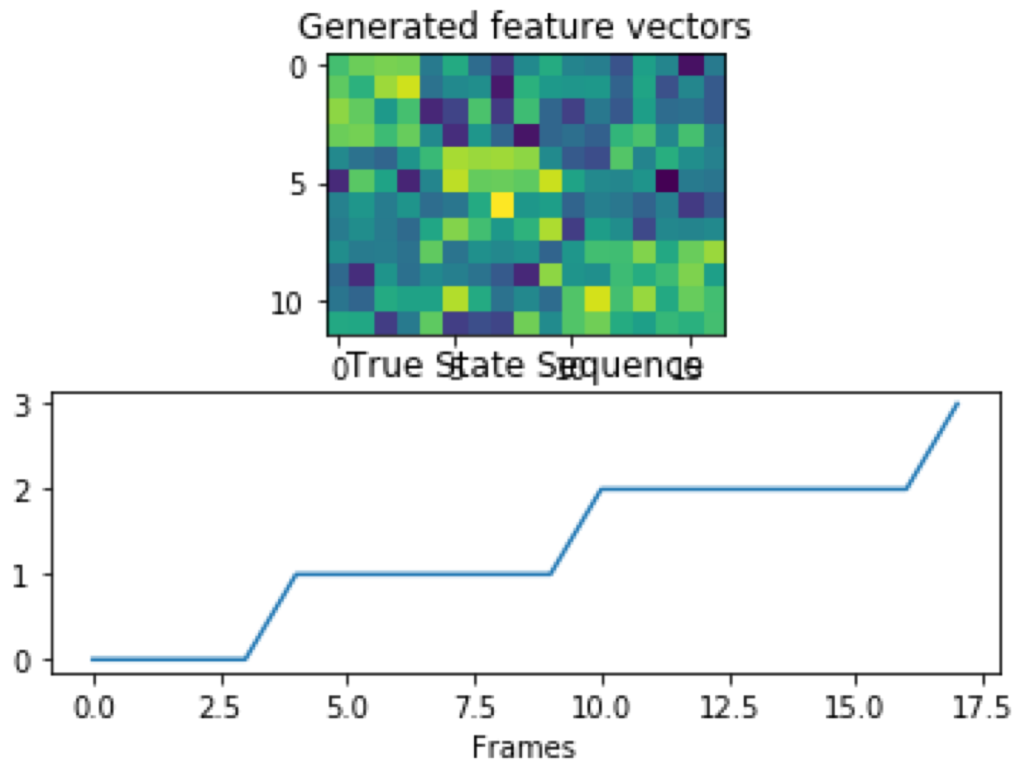


# Spectrogram: One spectral vector every 10ms

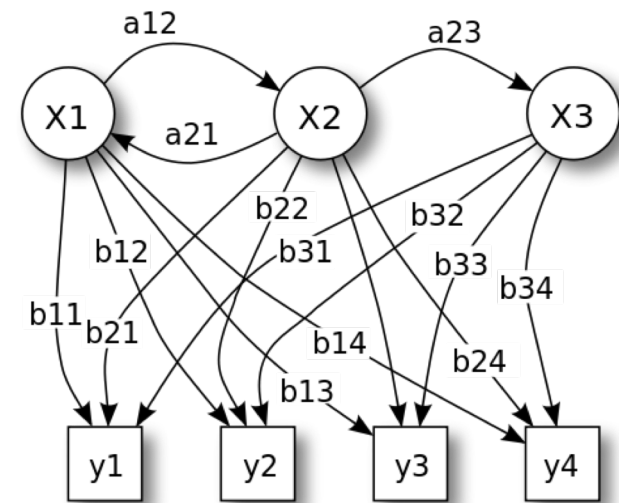
Slide credit: ECE 590SIP



# Recognition: Hidden Markov Model



Source: Mark Hasegawa-Johnson, ECE 417



Source: By Tdunningvectorization: Own work -  
Own work, CC BY 3.0,  
<https://commons.wikimedia.org/w/index.php?curid=18125206>

# Speech Processing: UIUC courses

Textbook section 23.5

- ECE 417: Multimedia Signal Processing (Speech & Video)
- ECE 537: Speech Processing
- ECE 594: Mathematical Models of Language
- CS 598PS: Machine Learning for Signal Processing



# Perception

- Web Texts
  - Sentiment Analysis; Information Retrieval; Information Extraction
- Speech and Natural Language Processing
  - Parsing; Machine Translation; Speech Recognition
- Computer Vision
  - Motion Vectors; Object Recognition; Object Detection; image2speech

# Motion Vectors

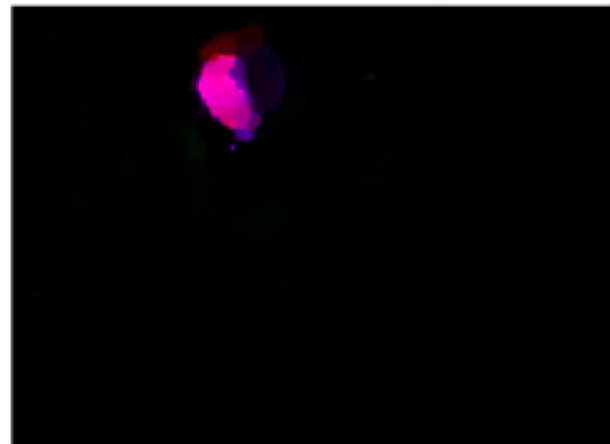
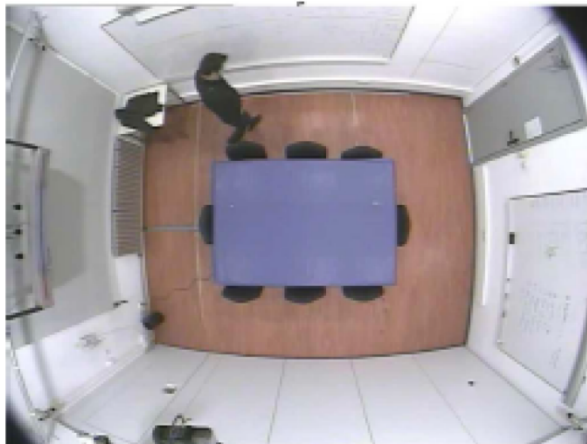
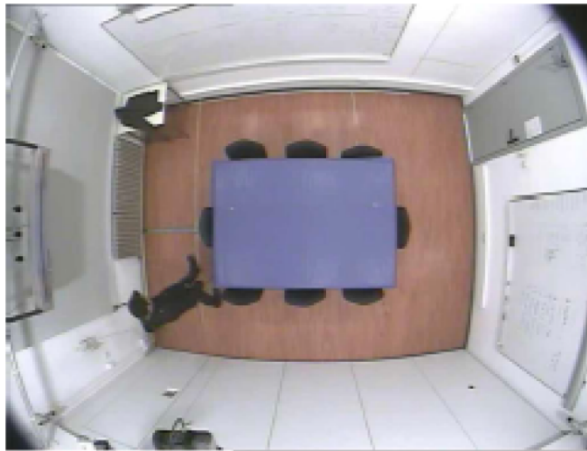
Textbook section 24.2.3; ECE417

- Motion vectors were originally invented for video coding; e.g., they were first standardized as part of MPEG-1
- For example, suppose that the pixels in a video were coded with 8 bits normally,  $0 \leq I(x, y, t) \leq 255$
- Suppose you can find  $\vec{V} = [V_x, V_y]$  so that  $-8 \leq \Delta I(x, y, t) \leq 7$ , where

$$\Delta I(x, y, t) = I(x, y, t) - I(x - V_x, y - V_y, t - 1)$$

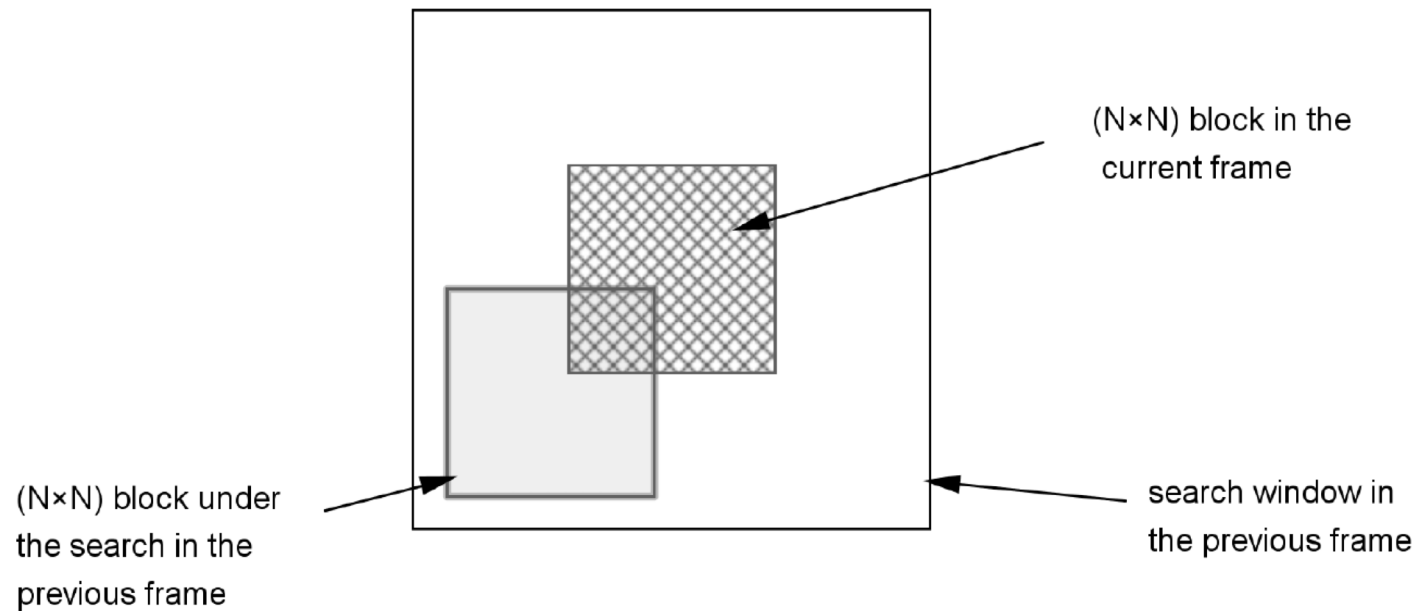
- Then you can code at just 4 bits/pixel, instead of 8 bits/pixel

## Applications of Motion Vectors: Object Tracking



Huang, Zhuang & Hasegawa-Johnson, 2011, Fig. 1

# The Block-Match Algorithm



By German iris – Own work, CC BY-SA 4.0, <https://commons.Wikimedia.org/w/index.php?curid=472>

## The Block-Match Algorithm

For each position,  $\vec{r}$ , in the frame at time  $t$ , we find a position  $\vec{r} - \vec{v}$  at time  $t - 1$  that best matches it. Here “best” is defined as minimum distance, e.g.,

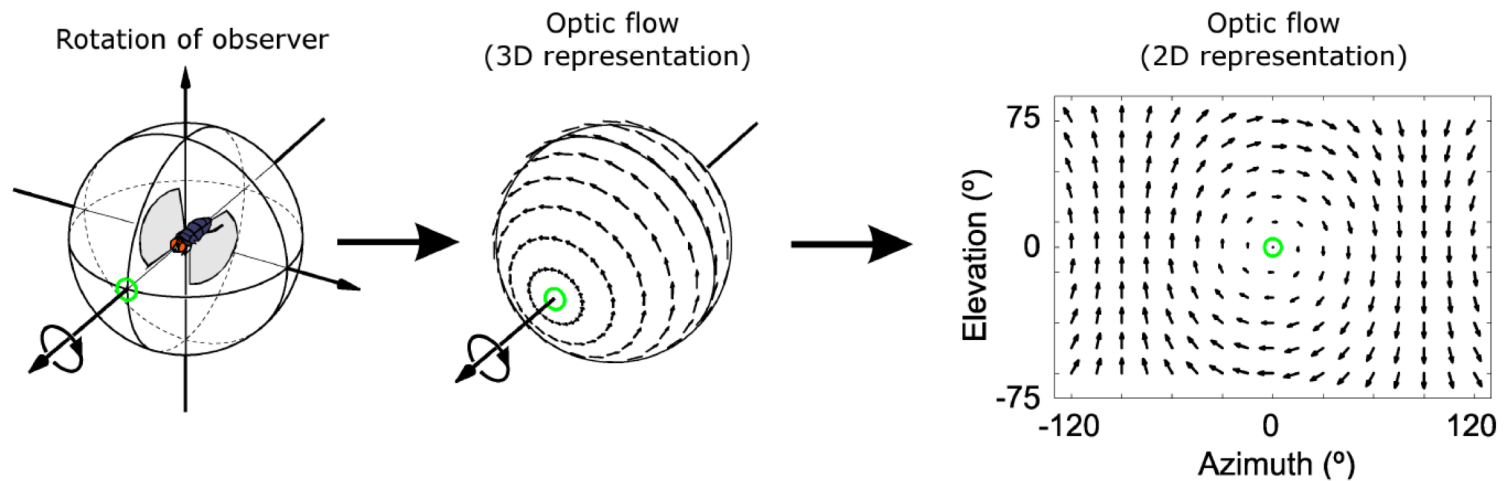
$$\vec{v}(\vec{r}) = \arg \min \frac{1}{N^2} \sum_{\vec{m}=(1,1)}^{(N,N)} |I(\vec{r} + \vec{m}, t) - I(\vec{r} + \vec{m} - \vec{v}(\vec{r}), t - 1)|$$

or mean-squared error (MSE):

$$\vec{v}(\vec{r}) = \arg \min \frac{1}{N^2} \sum_{\vec{m}=(1,1)}^{(N,N)} |I(\vec{r} + \vec{m}, t) - I(\vec{r} + \vec{m} - \vec{v}(\vec{r}), t - 1)|^2$$

where  $\vec{r} = [r_1, r_2]$  is the location of a size  $N \times N$  block, and  $\vec{v}(\vec{r})$  is the motion vector.

# Optical Flow



Adapted from PLoS Biology (CC licensed) article: Huston SJ, Krapp HG, 2008 Visuomotor Transformation in the Fly Gaze Stabilization System. PLoS Biol 6(7): e173.  
doi:10.1371/journal.pbio.0060173.

## Optical Flow

$$\begin{aligned} 0 &= \frac{\partial I}{\partial x} \frac{\Delta x}{\Delta t} + \frac{\partial I}{\partial y} \frac{\Delta y}{\Delta t} + \frac{\partial I}{\partial t} \frac{\Delta t}{\Delta t} \\ &= \frac{\partial I}{\partial x} V_x + \frac{\partial I}{\partial y} V_y + \frac{\partial I}{\partial t} \end{aligned}$$

Re-arranging, we get **the optical flow equation**:

$$\nabla I^T \vec{v} = -b$$

where we define  $b = \frac{\partial I}{\partial t}$ , and

$$\nabla I = \left[ \frac{\partial I}{\partial x}, \frac{\partial I}{\partial y} \right]^T$$

# Perception

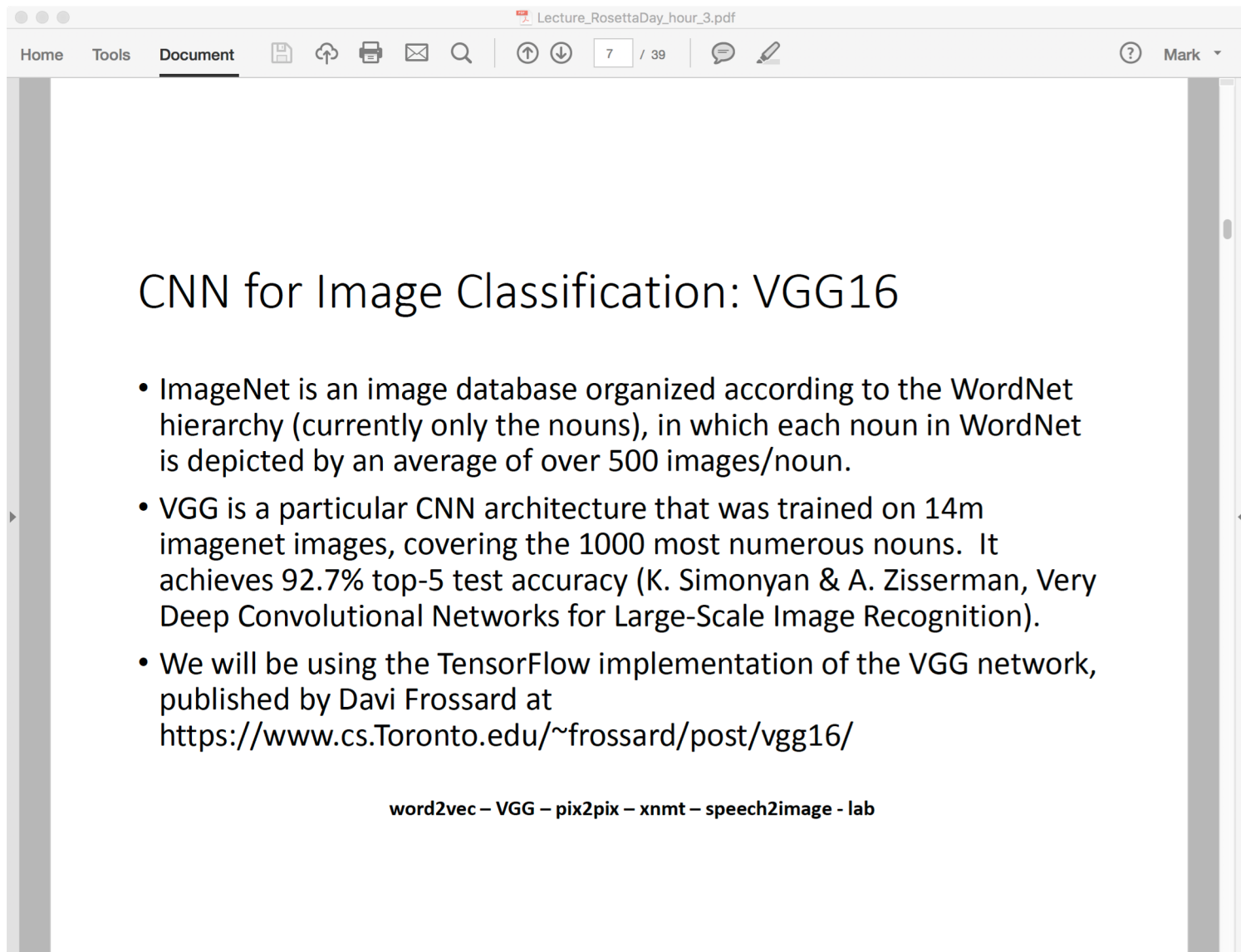
- Web Texts
  - Sentiment Analysis; Information Retrieval; Information Extraction
- Speech and Natural Language Processing
  - Parsing; Machine Translation; Speech Recognition
- Computer Vision
  - Motion Vectors; Object Recognition; Object Localization; image2speech



# Object Recognition

Textbook chapter 24; CS 446, 546, 549

- Now a classic problem in machine learning, thanks to big databases like [imagenet](#)
- Basically: given an input image, try to say what type of object is most visible in the input image

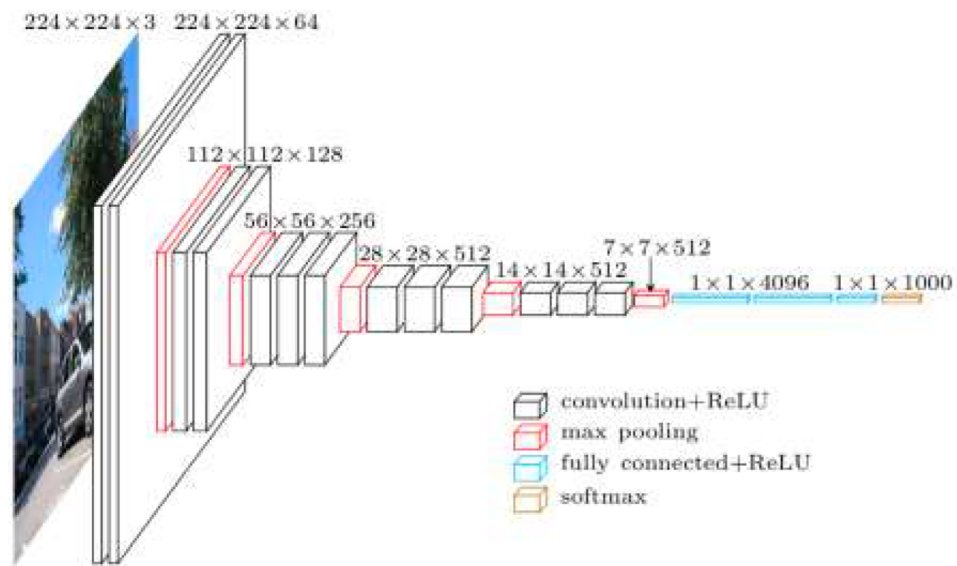


## CNN for Image Classification: VGG16

- ImageNet is an image database organized according to the WordNet hierarchy (currently only the nouns), in which each noun in WordNet is depicted by an average of over 500 images/noun.
- VGG is a particular CNN architecture that was trained on 14m imagenet images, covering the 1000 most numerous nouns. It achieves 92.7% top-5 test accuracy (K. Simonyan & A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition).
- We will be using the TensorFlow implementation of the VGG network, published by Davi Frossard at <https://www.cs.Toronto.edu/~frossard/post/vgg16/>

word2vec – VGG – pix2pix – xnmt – speech2image - lab

## CNN for Image Classification: VGG16



Lecture\_RosettaDay\_hour\_3.pdf

Home Tools Document 15 / 39 Mark

## CNN for Image Classification: vgg16.py

**# This method creates all the fully connected layers.**  
**# It doesn't have to be a separate method! But Davi wrote it that way.**

```
def fc_layers(self):  
    with tf.name_scope('fc1') as scope:  
        # Input to the first fully connected layer is a reshaped version of the last convolutional layer  
        shape = int(np.prod(self.pool5.get_shape()[1:]))  
        pool5_flat = tf.reshape(self.pool5, [-1, shape])  
        ....(stuff omitted)...
```

**# This variable is called the "penultimate" or "feature" layer, because it's last before the softmax**

```
self.fc2 = tf.nn.relu(fc2l)  
... (stuff omitted) ...
```

**# This layer is the "ultimate" or "softmax" layer.**  
**# You can't tell that from the following line: you only know this fact if you remember**  
**# that \_\_init\_\_ included the line "self.probs=tf.nn.softmax(self.fc3l)"**

```
self.fc3l = tf.nn.bias_add(tf.matmul(self.fc2, fc3w), fc3b)
```

word2vec – VGG – pix2pix – xnmt – speech2image - lab

# Perception

- Web Texts
  - Sentiment Analysis; Information Retrieval; Information Extraction
- Speech and Natural Language Processing
  - Parsing; Machine Translation; Speech Recognition
- Computer Vision
  - Motion Vectors; Object Recognition; **Object Localization; image2speech**

# Face Detection

Slide credit: ECE 417



rects.txt:

12 rectangles per line: lips, face,  
other

4 ints/rectangle:

[xmin,ymin,width,height]

showrects.m plots

Yellow: lips (first 4/line)

Cyan: face (next 4/line)

Red: other (next 4/line)

MP:

Discriminate face vs. other

## Example features: order 2, horizontal

Feature  $f(x; fr, q=2, v=0)$

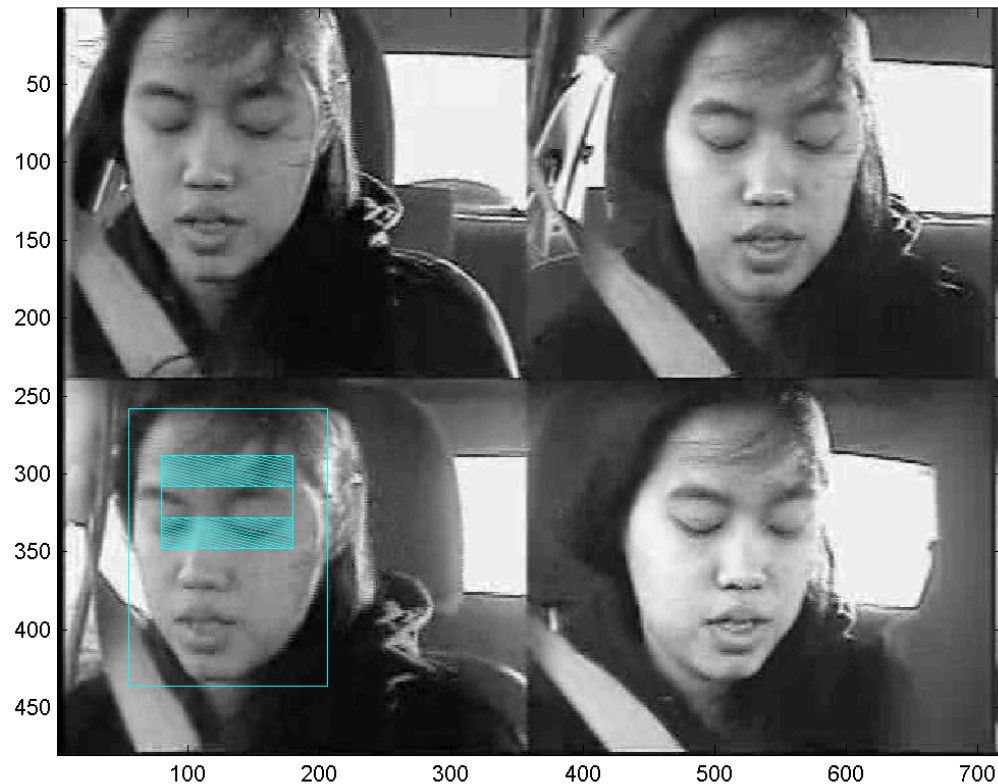
An order-2 horizontal feature is the sum of the right half, minus the sum of the left half.



## Other useful features: order 3, vertical

Feature  $f(x; fr, q=3, v=1)$

An order-3 vertical feature is the sum of the outer thirds, minus the sum of the middle third.





# Adaboost

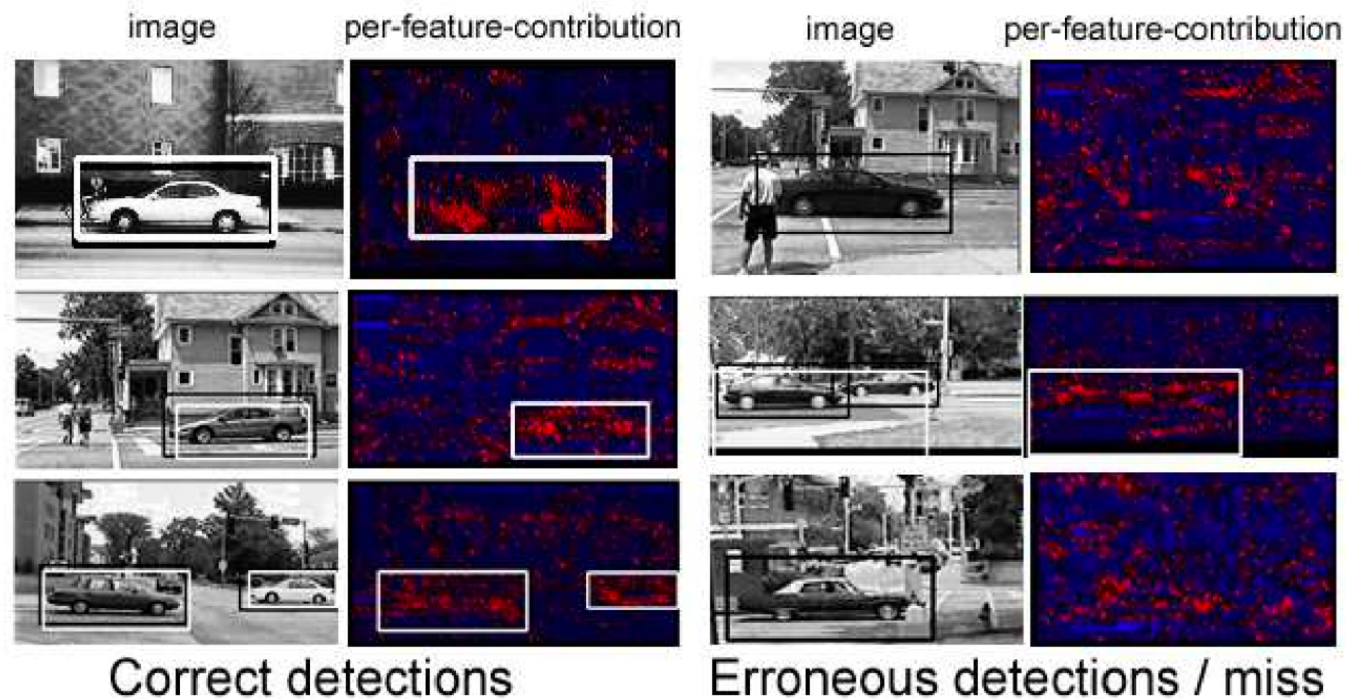
Suppose somebody told you: I'm going to take a whole bunch of scalar classifiers. Let's use  $h_t(x)$  to mean the classifier computed in the  $t$ 'th training iteration; remember that  $h_t(x)$  is either 0 or 1. Then I'm going to add them all together, and the final classifier will be

$$h(x) = \begin{cases} 1 & \text{if } \sum_t \alpha_t (2h_t(x) - 1) > 0 \\ 0 & \text{if } \sum_t \alpha_t (2h_t(x) - 1) < 0 \end{cases}$$

How would you choose the classifiers? How would you choose  $\alpha_t$ ?

The answer: Viola & Jones, 2001, Adaboost face detectors

# I-Vector Object Detectors



Zhuang, Zhou, Hasegawa-Johnson & Huang, "Efficient Object Localization with Gaussianized Vector Representation," IMCE 2009

# Perception

- Web Texts
  - Sentiment Analysis; Information Retrieval; Information Extraction
- Speech and Natural Language Processing
  - Parsing; Machine Translation; Speech Recognition
- Computer Vision
  - Motion Vectors; Object Recognition; Object Localization; **image2speech**

# How many languages are there in the world?

- According to ethnologue, there are 6900 languages. Methods of writing have been invented for about 4000 of them.
- There are 206 countries in the world, of which 193 have an official language; 101 have more than one. There are no figures on the number of different “official” languages, but maybe it is 300 languages.
- So the number of distinct languages in which children are taught to READ and WRITE (as opposed to just speaking) is about 300.
- All of the other  $6900 - 300 = 6600$  languages (dialects; codes) are purely spoken: a writing system may exist, but is rarely used.

$$\text{Image2speech} = \text{img2txt} + \text{TTS} - \text{text}$$

- If there is no writing system, then speech is the only communication tool, but:

Must one speak in Modern Standard Arabic to be understood by one's cell phone?

- Task Definition: Can we develop speech technology in a dialect that is almost never written down in any standardized text format?

Definition: An image2speech algorithm is an algorithm that observes an image, and generates a spoken description of the image, without requiring that the language of the description has any standardized text format.

# Datasets

- AMT recordings obtained from **Flickr8k** - 40k spoken captions available online (Julia Hockenmaier et al., 2009)
  - <https://groups.csail.mit.edu/sls/downloads/>
  - D. Harwath and J. Glass, *“Deep multimodal semantic embeddings for speech and images”* in IEEE ASRU, Scottsdale, Arizona, USA, December 2015



- A brown and white dog is running through the snow
- A dog is running in the snow
- A dog running through snow
- A white and brown dog is running through a snow covered field
- The white and brown dog is running over the surface of the snow

# Image representation: CNNFEAT $\vec{s}_{mn}$

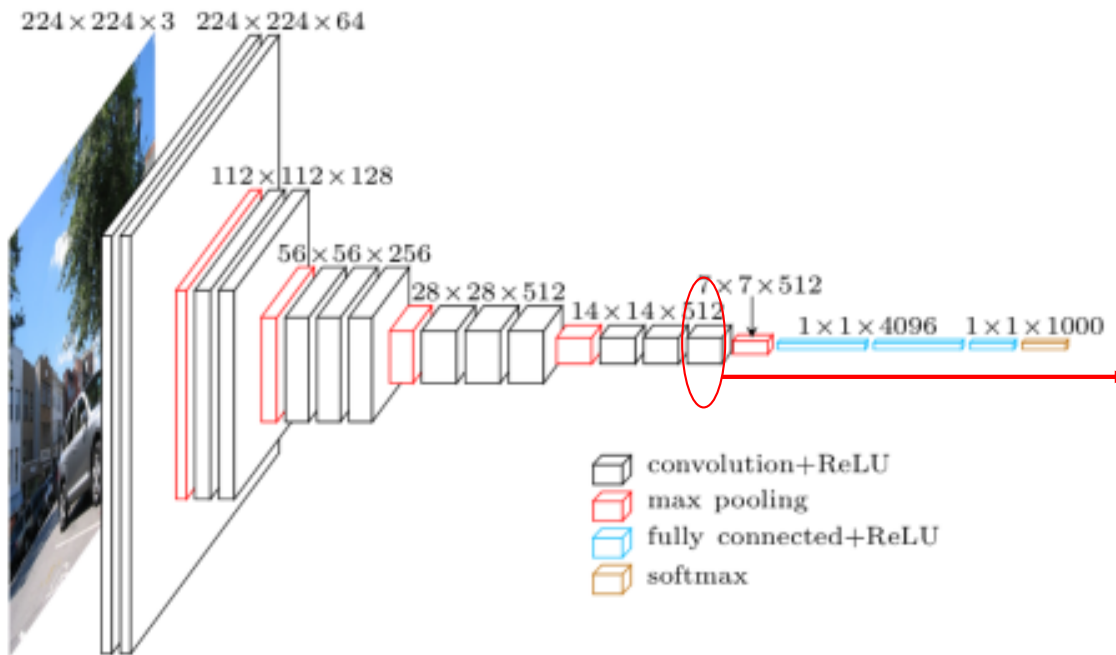
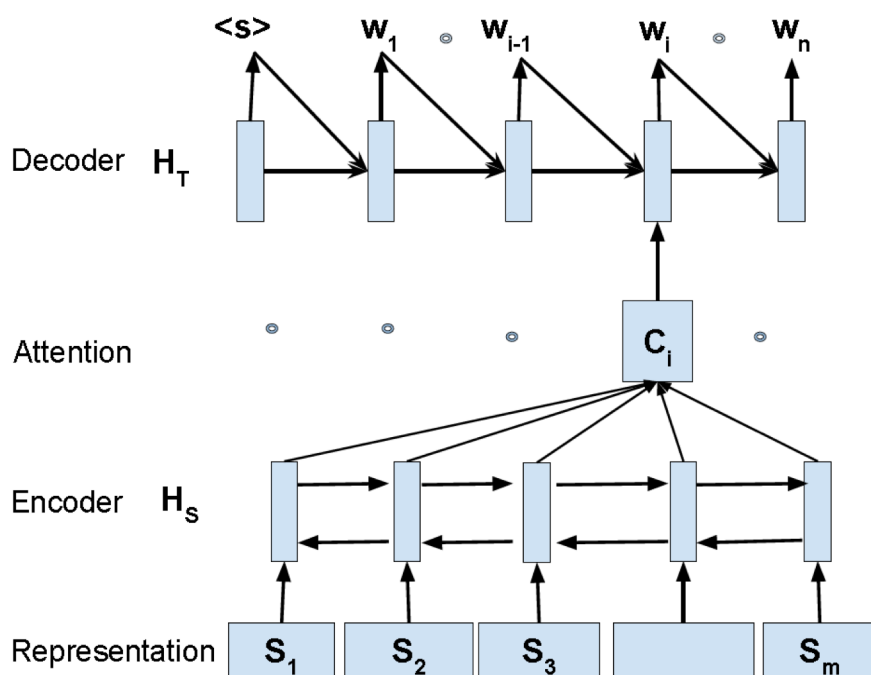


Figure copied from Simonyan & Zisserman, 2014.

- [ImageNet](#) = >500 images/noun of each of the nouns in WordNet.
- [VGG](#) = 13-layer CNN + 2-layer FCN, trained on 14m images, covering the 1000 most numerous nouns, 92.7% top-5 test accuracy.
- **CNNFEAT: 196 feature vectors/image, 512d/vector, from the last CNN layer. Each receptive field covers about 40x40 pixels in the original 224x224 image.**
- VGGFEAT (used later in today's talk, not right now): 1 vector/image, 4096d/vector, from penultimate FCN layer

# im2ph: phones from images

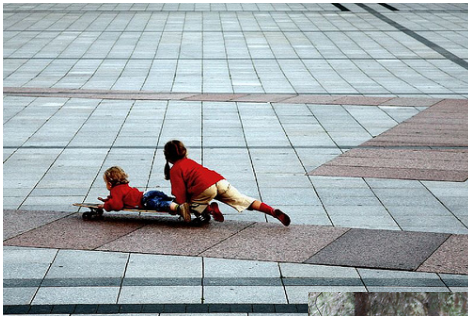


- “Representation:” 196 vectors/image
- “Encoder:” PyramidalLSTM with one 128d state vector. Sequence is row-wise raster scan of the image.
- “Attention:” StandardAttender, 128d input, 128d state vector, N hidden nodes
- “Decoder:” MlpSoftmaxDecoder, 3 layers, 1024d hidden vectors
- Output vocabulary: synthetic phones (MSCOCO), force-aligned phones (flickr8k), or acoustic unit discoveries (both)

Figure copied without permission from Duong, Anastasopoulos, Chiang, Bird & Cohn, NAACL-HLT 2016.



## flickr8K: American phones



- Reference 1: “The boy +um+ laying face down on a skateboard is being pushed along the ground by +laugh+ another boy.”



- Reference 2: “Two girls +um+ play on a skateboard +breath+ in a court +laugh+ yard.”



- Hypothesis (128d attender): SIL +BREATH+ SIL T UW M EH N AA R R AY D IX NG AX R EH D AE N W AY T SIL R EY S SIL



- Hypothesis (64d attender): SIL +BREATH+ SIL T UW W IH M AX N W AO K IX NG AA N AX S T R IY T SIL



- Reference 1: “A boy +laugh+ in a blue top +laugh+ is jumping off some rocks in the woods.”

- Reference 2: “A boy +um+ jumps off a tan rock.”



- Hypothesis (128d attender): SIL +BREATH+ SIL EY M AE N IH Z JH AH M P IX NG IH N DH AX F AO R EH S T SIL



- Hypothesis (64d attender): SIL +BREATH+ SIL EY Y AH NG B OY W EY R IX NG AX B L UW SH ER T SIL IH Z R AY D IX NG AX HH IH L SIL



Images and Reference Texts: Hodosh, Young & Hockenmaier, 2013. Waveforms: Harwath and Glass, 2015

# Perception

- Web Texts
  - Sentiment Analysis; Information Retrieval; Information Extraction
- Speech and Natural Language Processing
  - Parsing; Machine Translation; Speech Recognition
- Computer Vision
  - Motion Vectors; Object Recognition; Object Localization; image2speech