CS 1674: Intro to Computer Vision

Sequential Data: Language and Vision; Video and Motion

Prof. Adriana Kovashka
University of Pittsburgh

April 14, 2022

Plan for this lecture

- Language and vision
  - Image captioning
  - Tool: Recurrent neural networks
  - Video captioning
  - Visual question answering
- Motion and video
  - Modeling and replicating motion
  - Tracking how an object moves

Motivation: Descriptive Text for Images

"It was an arresting face, pointed of chin, square of jaw. Her eyes were pale green without a touch of hazel, starred with bristly black lashes and slightly tilted at the ends. Above them, her thick black brows slanted upward, cutting a startling oblique line in her magnolia-white skin–that skin so prized by Southern women and so carefully guarded with bonnets, veils and mittens against hot Georgia suns"

Scarlett O'Hara described in Gone with the Wind

Some pre-RNN good results

This is a picture of one sky, one road and one sheep. The gray sky is over the gray road. The gray sheep is by the gray road.

This is a picture of two dogs. The first dog is near the second furry dog.

Here we see one road, one sky and one bicycle. The road is near the blue sky, and near the colorful bicycle. The colorful bicycle is within the blue sky.

Kulkarni et al., CVPR 2011

Some pre-RNN bad results

Missed detections:

Here we see one potted plant.

This is a picture of one dog.

False detections:

There are one road and one cat. The furry road is in the furry cat.

This is a picture of one tree, one road and one person. The rusty tree is under the red road. The colorful person is near the rusty tree, and under the red road.

Incorrect attributes:

This is a photograph of two sheeps and one grass. The first black sheep is by the green grass, and by the second black sheep. The second black sheep is by the green grass.

This is a photograph of two horses and one grass. The first feathered horse is within the green grass, and by the second feathered horse. The second feathered horse is within the green grass.

Kulkarni et al., CVPR 2011

Results with Recurrent Neural Networks

"man in black shirt is playing guitar."

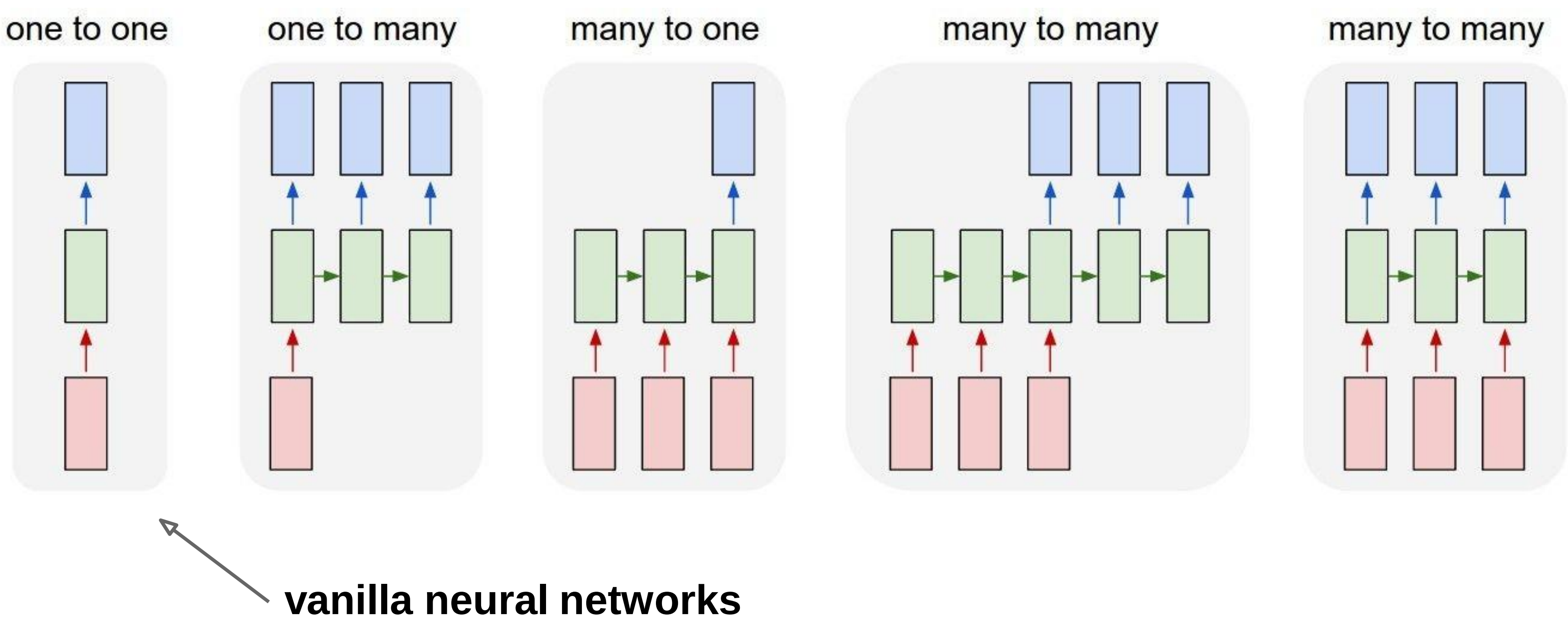
"construction worker in orange safety vest is working on road."

"two young girls are playing with lego toy."

"boy is doing backflip on wakeboard."

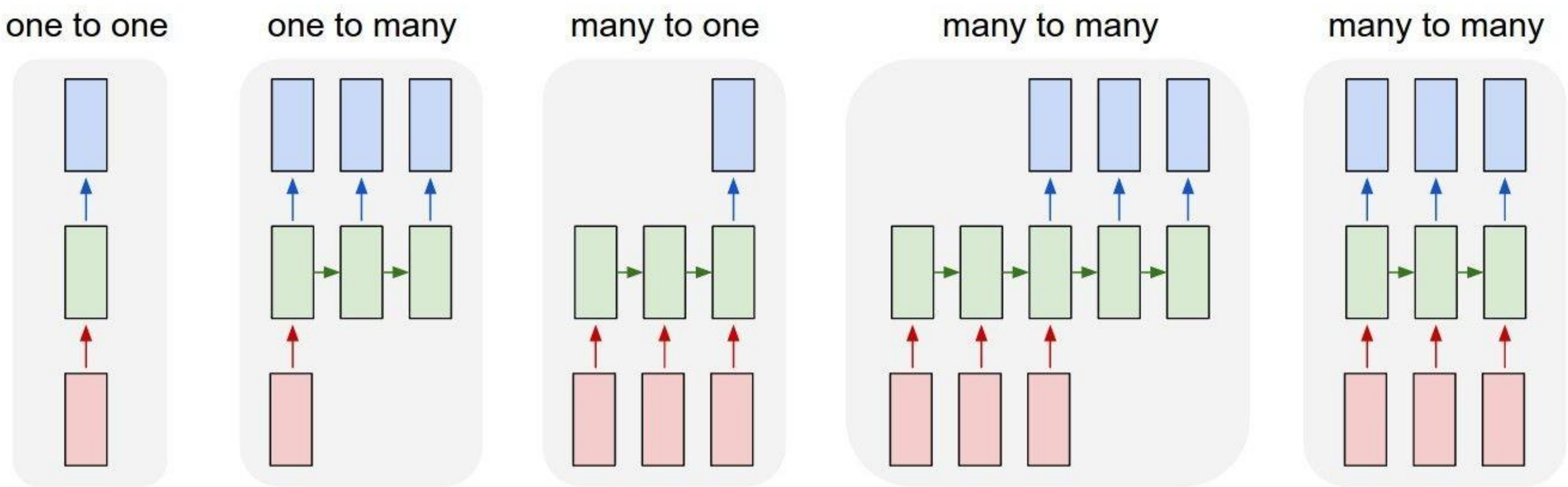
Karpathy and Fei-Fei, CVPR 2015

Recurrent Networks offer a lot of flexibility:



Andrej Karpathy

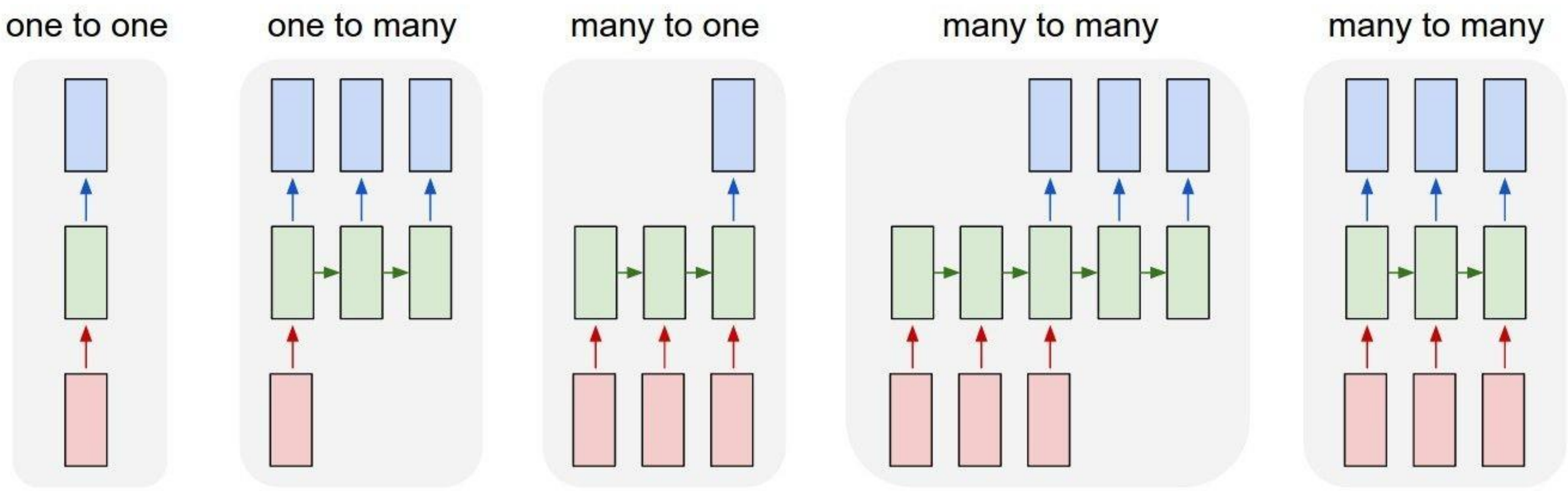
Recurrent Networks offer a lot of flexibility:



e.g. **image captioning**
image -> sequence of words

Andrej Karpathy

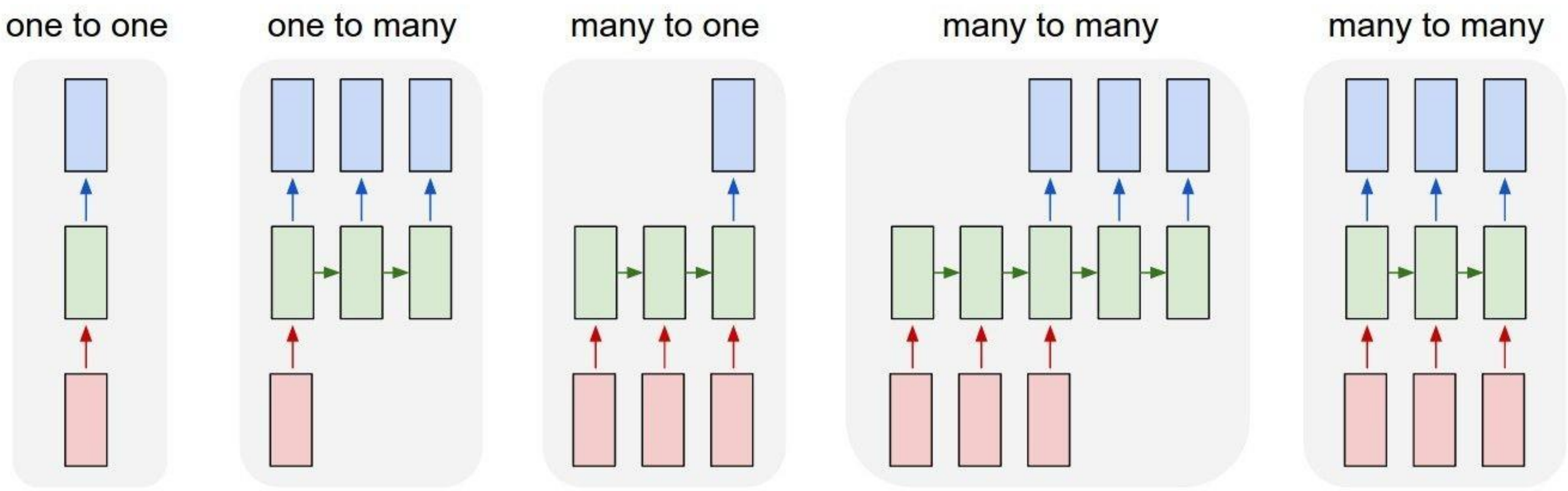
Recurrent Networks offer a lot of flexibility:



e.g. **sentiment classification**
sequence of words -> sentiment

Andrej Karpathy

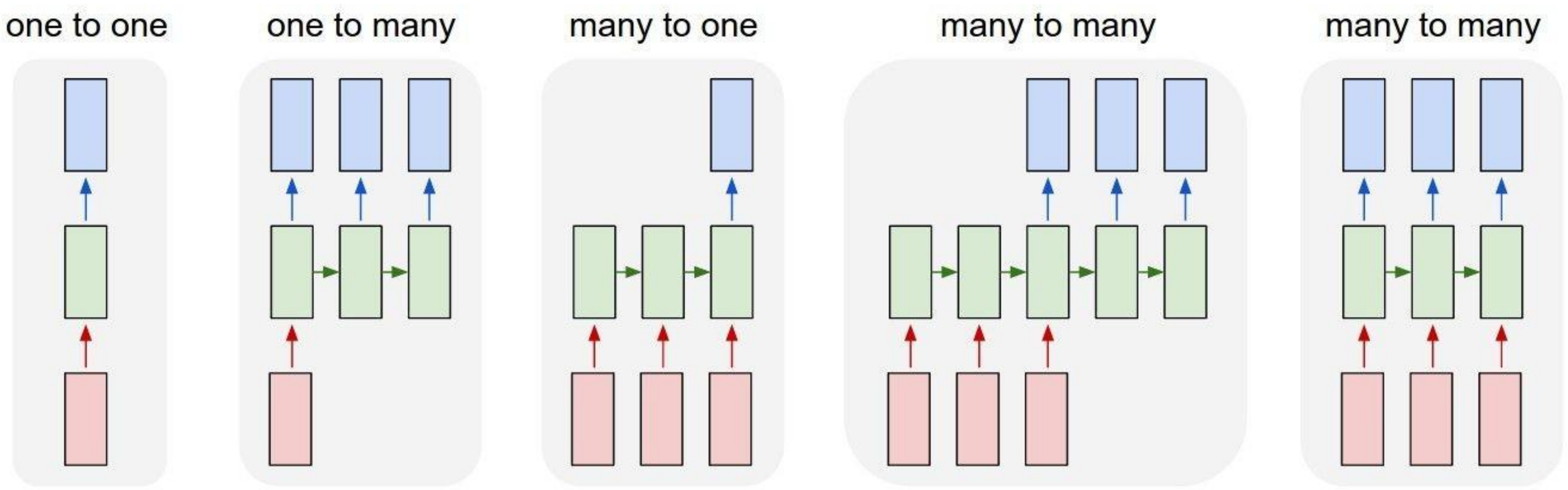
Recurrent Networks offer a lot of flexibility:



e.g. **machine translation**
seq of words -> seq of words

Andrej Karpathy

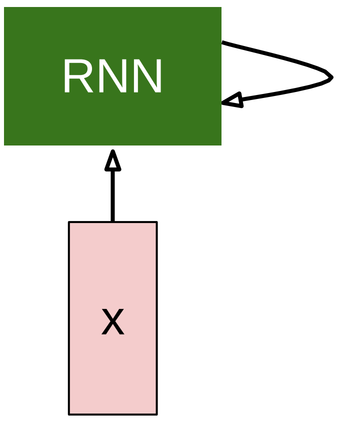
Recurrent Networks offer a lot of flexibility:



e.g. **video classification on frame level**

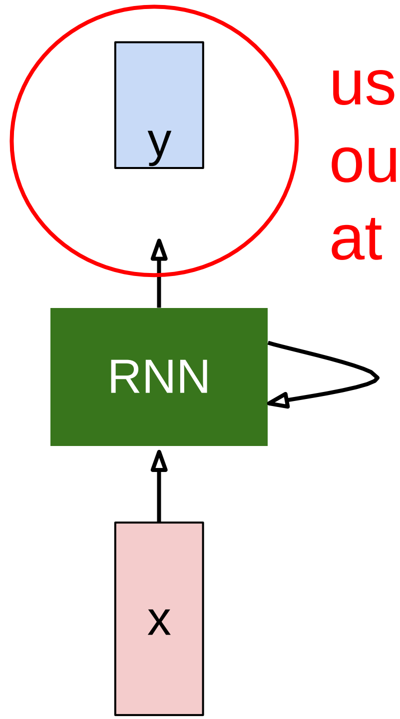
Andrej Karpathy

Recurrent Neural Network



Andrej Karpathy

Recurrent Neural Network



usually want to output a prediction at some time steps

Adapted from Andrej Karpathy

Recurrent Neural Network

We can process a sequence of vectors **x** by applying a recurrence formula at every time step:

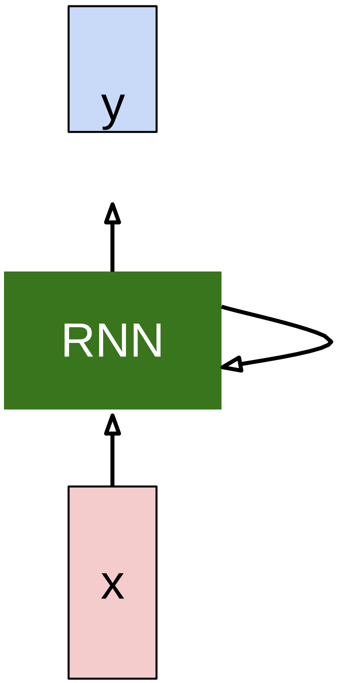
$$\boxed{h_t} = \boxed{f_W}(\boxed{h_{t-1}}, \boxed{x_t})$$

new state

old state

input vector at some time step

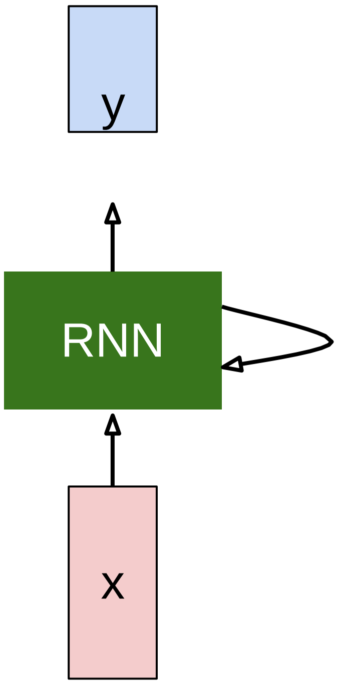
some function with parameters W



Recurrent Neural Network

We can process a sequence of vectors **x** by applying a recurrence formula at every time step:

$$h_t = f_W(h_{t-1}, x_t)$$

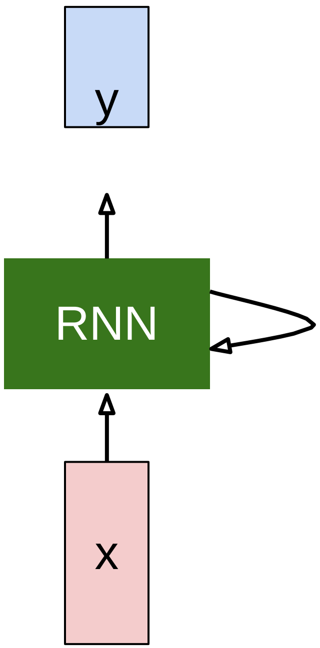


Notice: the same function and the same set of parameters are used at every time step.

Andrej Karpathy

(Vanilla) Recurrent Neural Network

The state consists of a single *"hidden"* vector **h**:



$$h_t = f_W(h_{t-1}, x_t)$$

$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t)$$

$$y_t = W_{hy}h_t$$

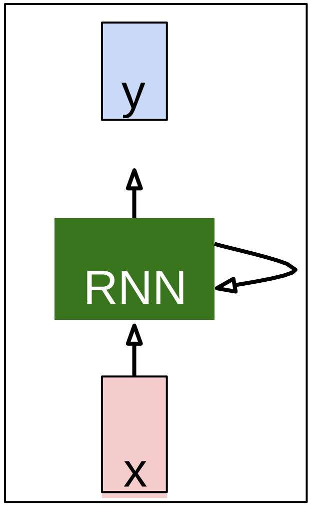
Andrej Karpathy

Example

Character-level language model example

Vocabulary:
[h,e,l,o]

Example training sequence:
“hello”



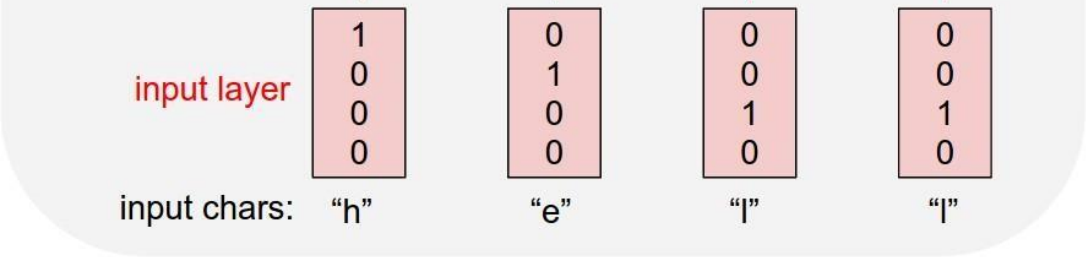
Andrej Karpathy

Example

Character-level language model example

Vocabulary: [h,e,l,o]

Example training sequence: **“hello”**



Andrej Karpathy

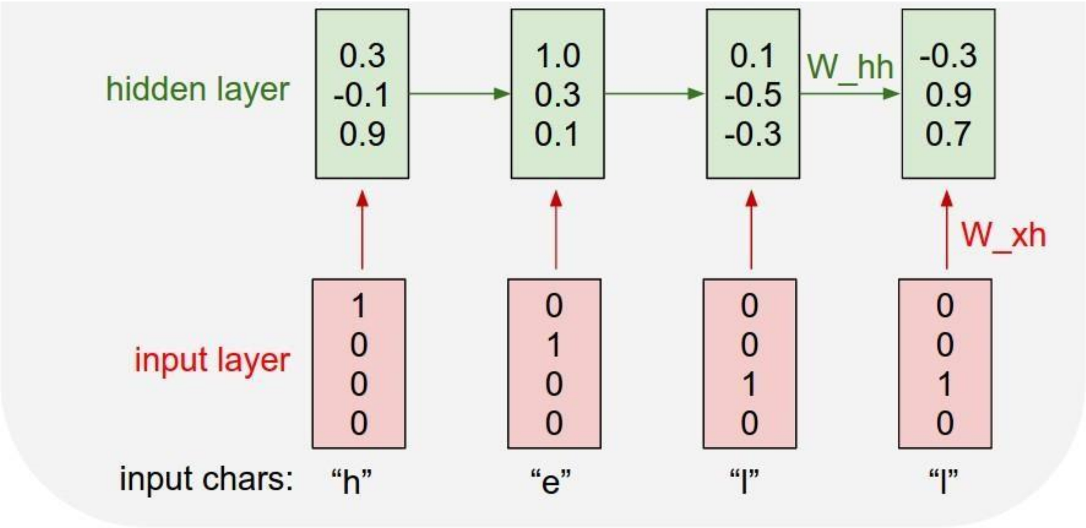
Example

Character-level language model example

Vocabulary:
[h,e,l,o]

Example training sequence:
“hello”

$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t)$$



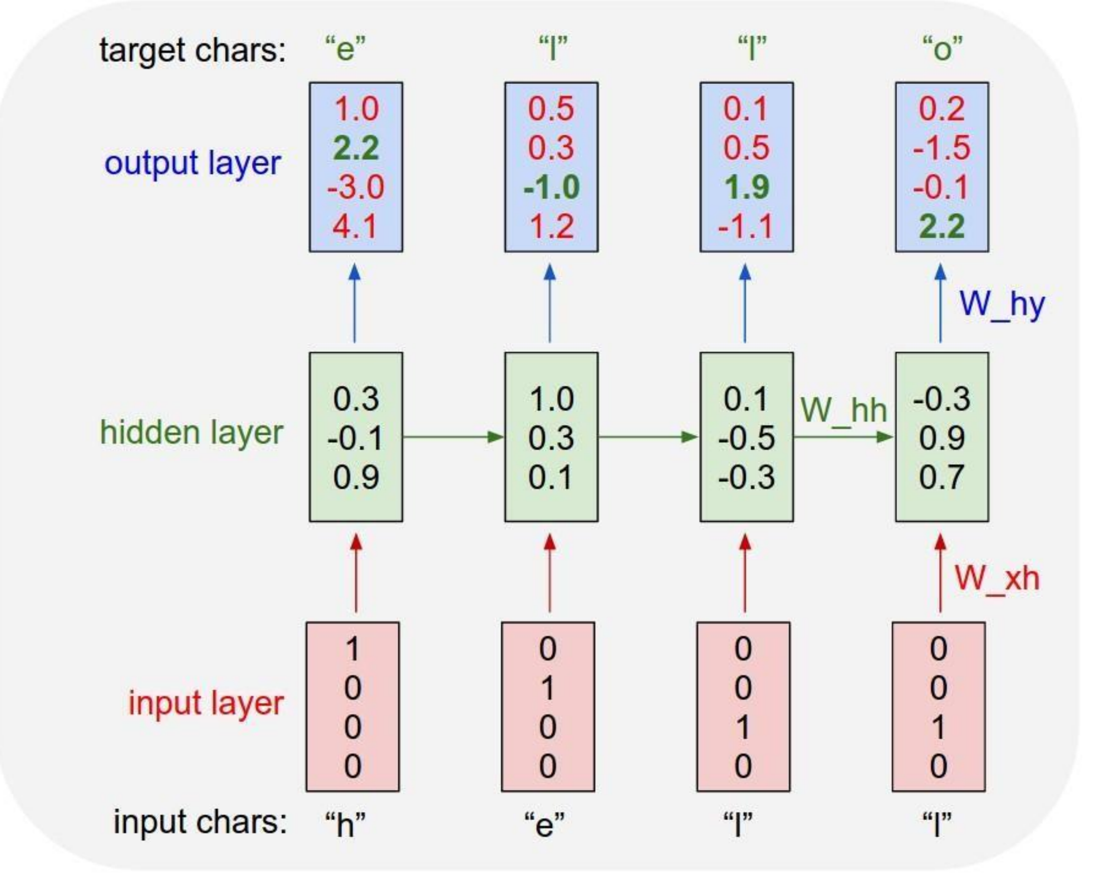
Andrej Karpathy

Example

Character-level language model example

Vocabulary:
[h,e,l,o]

Example training sequence:
"hello"



Andrej Karpathy

Extensions

- Vanishing gradient problem makes it hard to model long sequences
  - Multiplying together many values between 0 and 1 (range of gradient of sigmoid, tanh)
- One solution: Use RELU
- Another solution: Use RNNs with gates
  - Adaptively decide how much of memory to keep
  - Gated Recurrent Units (GRUs), Long Short Term Memories (LSTMs)

Generating poetry with RNNs

Sonnet 116 – Let me not ...

by William Shakespeare

Let me not to the marriage of true minds
        Admit impediments. Love is not love
Which alters when it alteration finds,
        Or bends with the remover to remove:
O no! it is an ever-fixed mark
        That looks on tempests and is never shaken;
It is the star to every wandering bark,
        Whose worth's unknown, although his height be taken.
Love's not Time's fool, though rosy lips and cheeks
        Within his bending sickle's compass come:
Love alters not with his brief hours and weeks,
        But bears it out even to the edge of doom.
If this be error and upon me proved,
        I never writ, nor no man ever loved.

Generating poetry with RNNs

at first:

```
tyntd-iafhatawiaoihrdemot  lytdws  e ,tfti, astai f ogoh eoase rrranbyne 'nhthnee e
plia tklrgd t o idoe ns,smtt   h ne etie h,hregtrs nigtike,aoaenns lng
```

train more

```
"Tmont thithey" fomesscerliund
Keushey. Thom here
sheulke, anmerenith ol sivh I lalterthend Bleipile shuwy fil on aseterlome
coaniogennc Phe lism thond hon at. MeiDimorotion in ther thize."
```

train more

```
Aftair fall unsuch that the hall for Prince Velzonski's that me of
her hearly, and behs to so arwage fiving were to it beloge, pavu say falling misfort
how, and Gogition is so overelical and ofter.
```

train more

```
"Why do what that day," replied Natasha, and wishing to himself the fact the
princess, Princess Mary was easier, fed in had oftened him.
Pierre aking his soul came to the packs and drove up his father-in-law women.
```

More info: http://karpathy.github.io/2015/05/21/rnn-effectiveness/

Andrej Karpathy

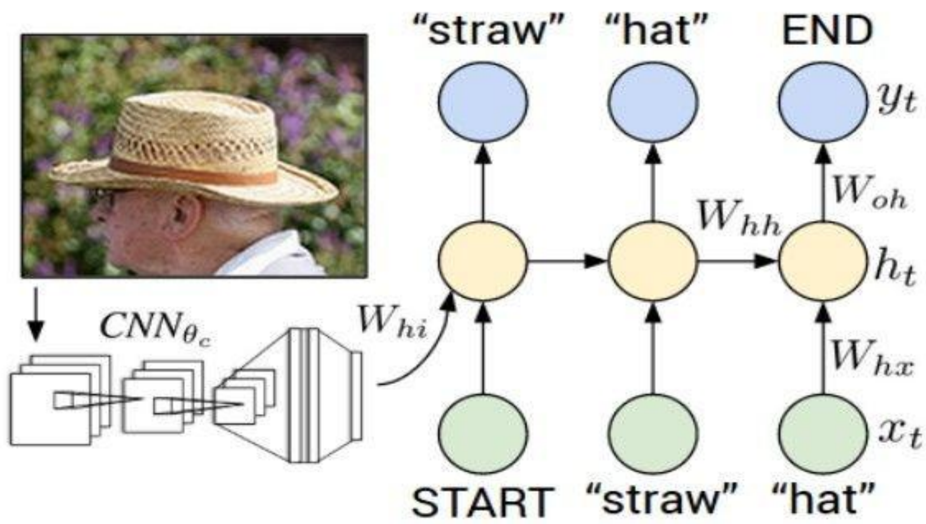
Generating poetry with RNNs

```
PANDARUS:
Alas, I think he shall be come approached and the day
When little srain would be attain'd into being never fed,
And who is but a chain and subjects of his death,
I should not sleep.

Second Senator:
They are away this miseries, produced upon my soul,
Breaking and strongly should be buried, when I perish
The earth and thoughts of many states.

DUKE VINCENTIO:
Well, your wit is in the care of side and that.

Second Lord:
They would be ruled after this chamber, and
my fair nues begun out of the fact, to be conveyed,
Whose noble souls I'll have the heart of the wars.

Clown:
Come, sir, I will make did behold your worship.

VIOLA:
I'll drink it.
```

```
VIOLA:
Why, Salisbury must find his flesh and thought
That which I am not aps, not a man and in fire,
To show the reining of the raven and the wars
To grace my hand reproach within, and not a fair are hand,
That Caesar and my goodly father's world;
When I was heaven of presence and our fleets,
We spare with hours, but cut thy council I am great,
Murdered and by thy master's ready there
My power to give thee but so much as hell:
Some service in the noble bondman here,
Would show him to her wine.

KING LEAR:
O, if you were a feeble sight, the courtesy of your law,
Your sight and several breath, will wear the gods
With his heads, and my hands are wonder'd at the deeds,
So drop upon your lordship's head, and your opinion
Shall be against your honour.
```


Andrej Karpathy

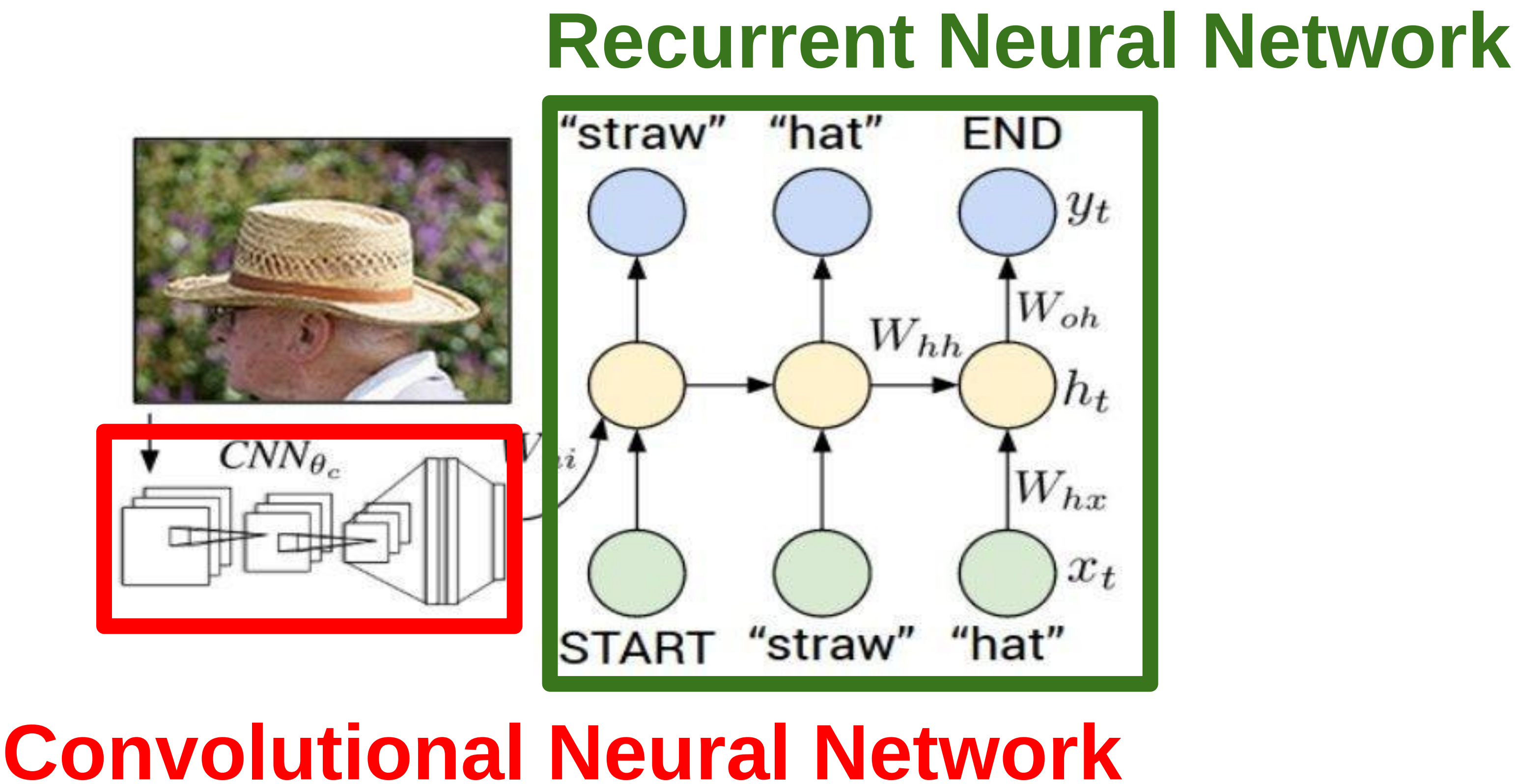
Image Captioning



CVPR 2015:
Deep Visual-Semantic Alignments for Generating Image Descriptions, Karpathy and Fei-Fei
Show and Tell: A Neural Image Caption Generator, Vinyals et al.
Long-term Recurrent Convolutional Networks for Visual Recognition and Description, Donahue et al.
Learning a Recurrent Visual Representation for Image Caption Generation, Chen and Zitnick

Adapted from Andrej Karpathy

Image Captioning

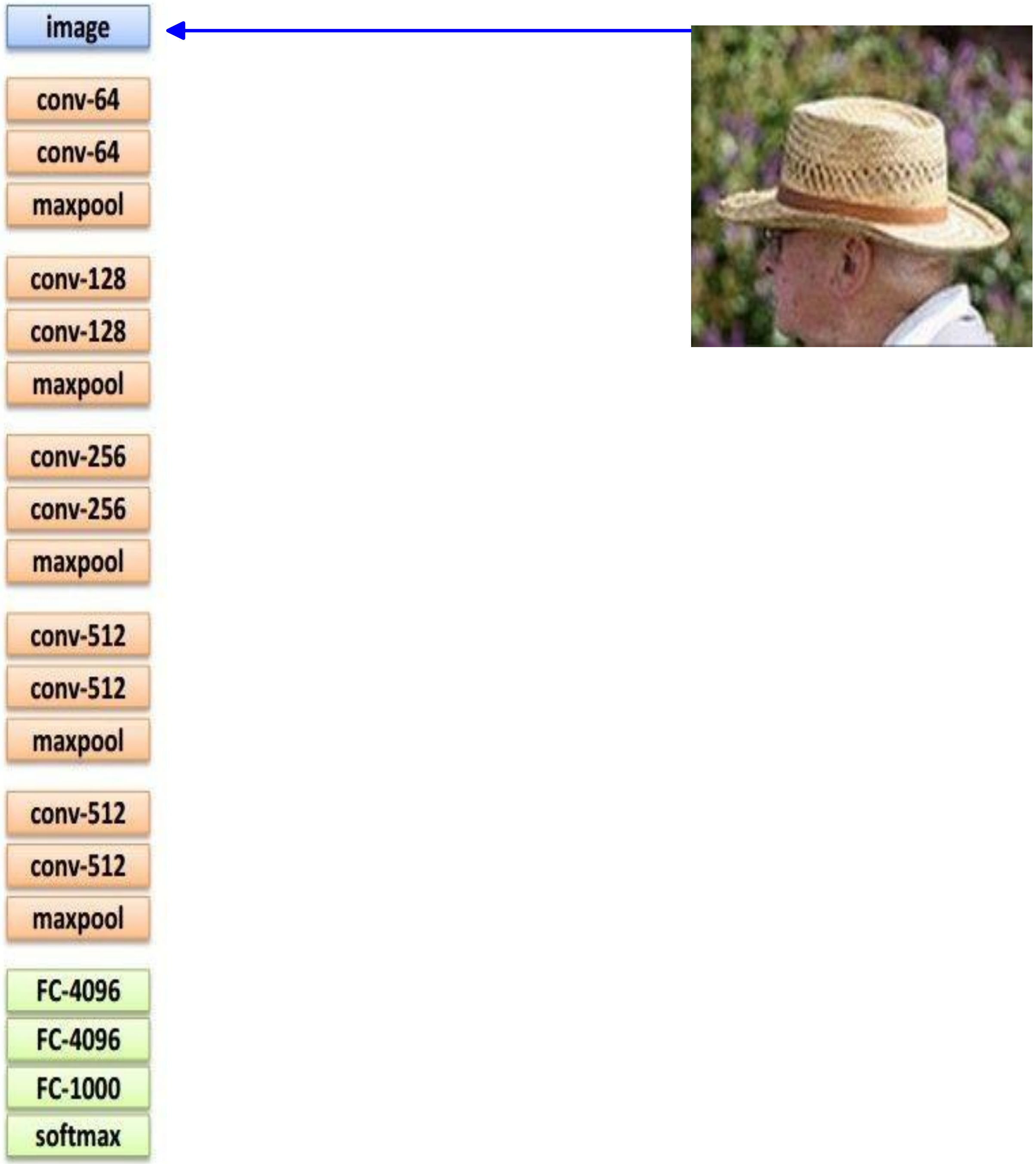


Andrej Karpathy

Image Captioning

test image

Andrej Karpathy



test image

Andrej Karpathy

test image

Andrej Karpathy

Image Captioning

image

conv-64
conv-64
maxpool

conv-128
conv-128
maxpool

conv-256
conv-256
maxpool

conv-512
conv-512
maxpool

conv-512
conv-512
maxpool

FC-4096
FC-4096

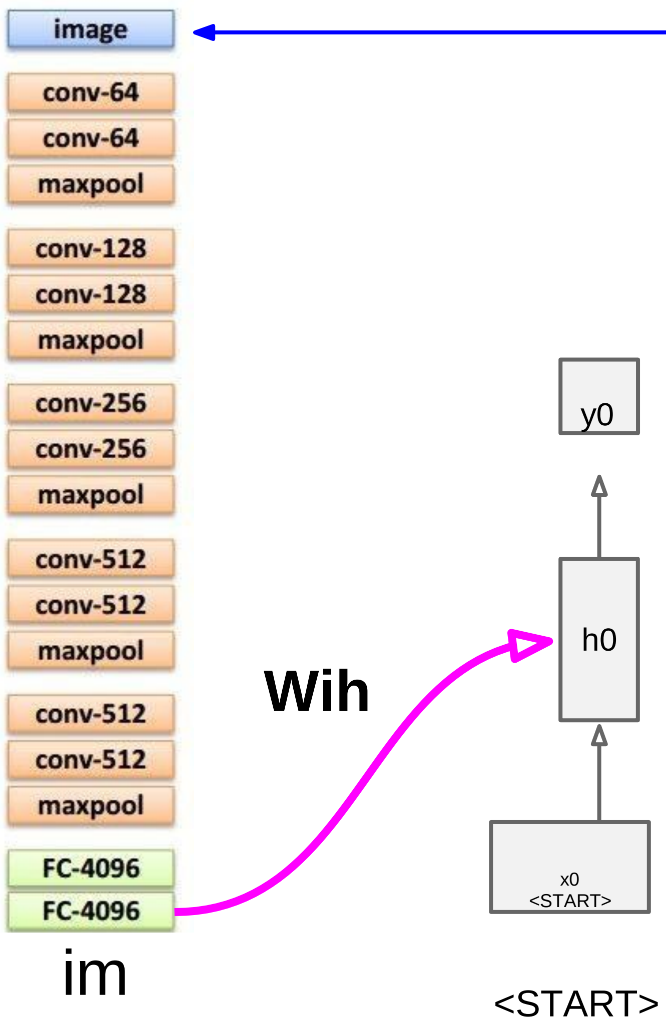
test image

x0
<START>

<START>

Andrej Karpathy

Image Captioning



test image

before:

$$h = \tanh(W_{xh} * x + W_{hh} * h)$$

now:

$$h = \tanh(W_{xh} * x + W_{hh} * h + \mathbf{W_{ih} * im})$$

Image Captioning

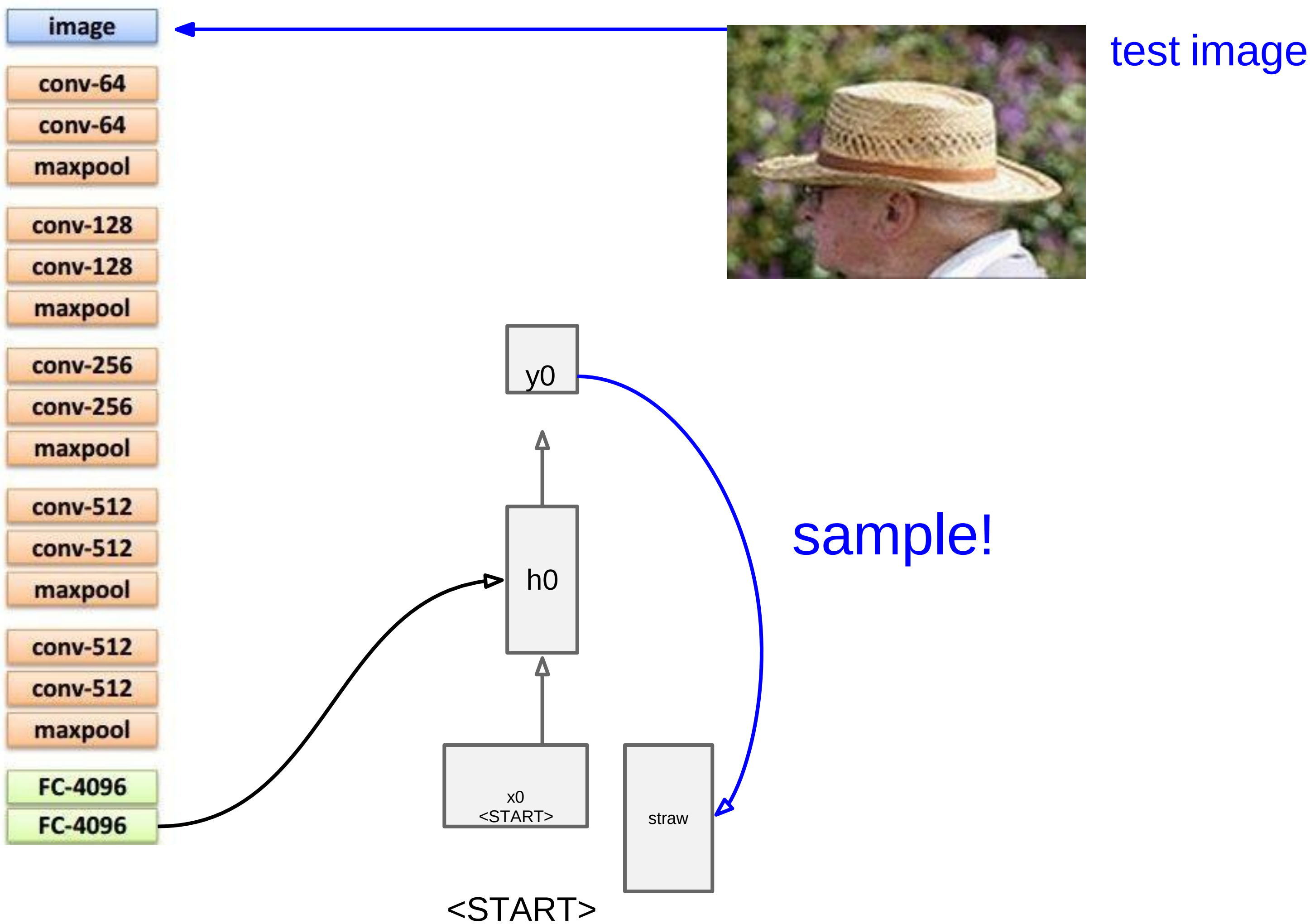


Image Captioning

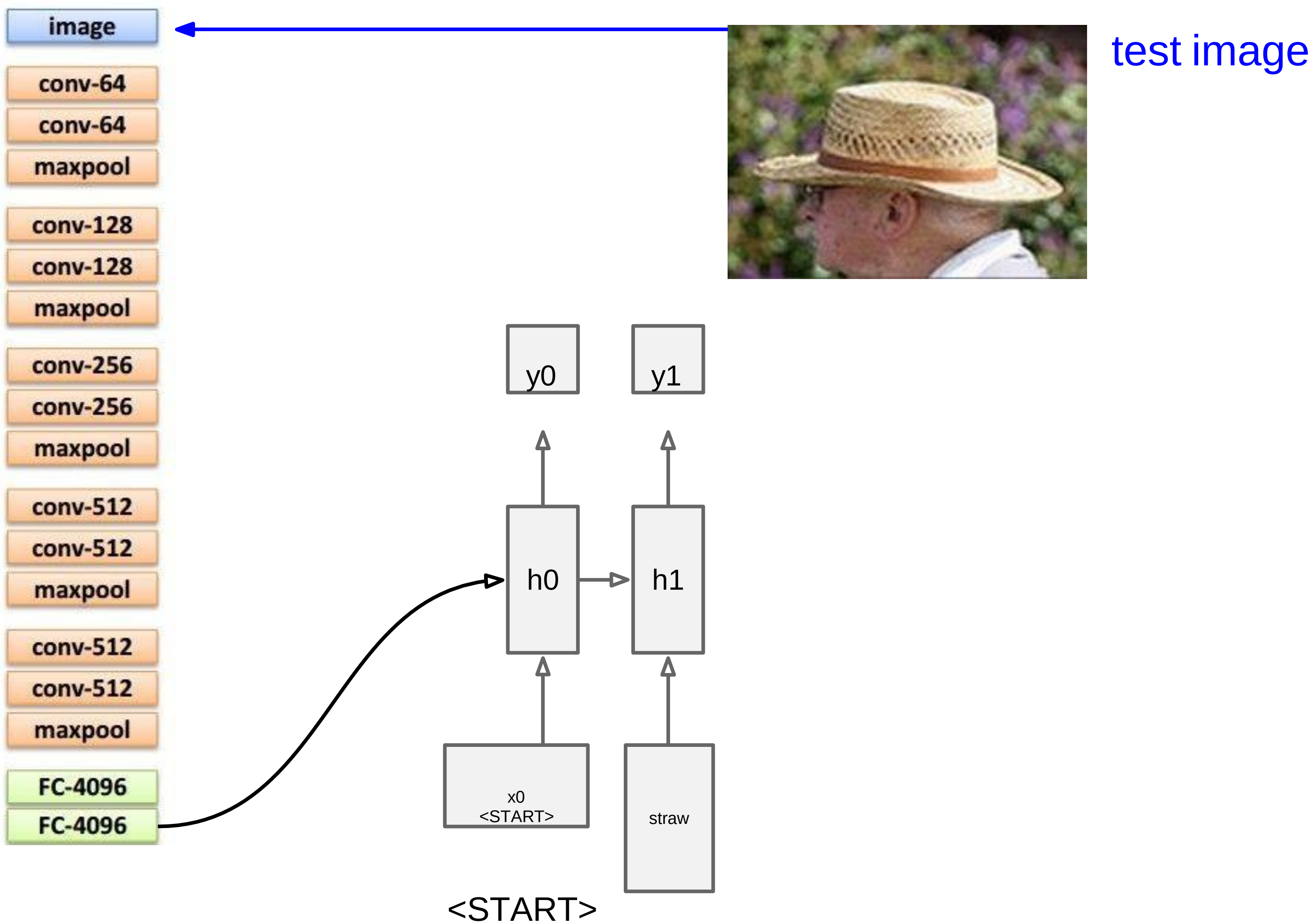


Image Captioning

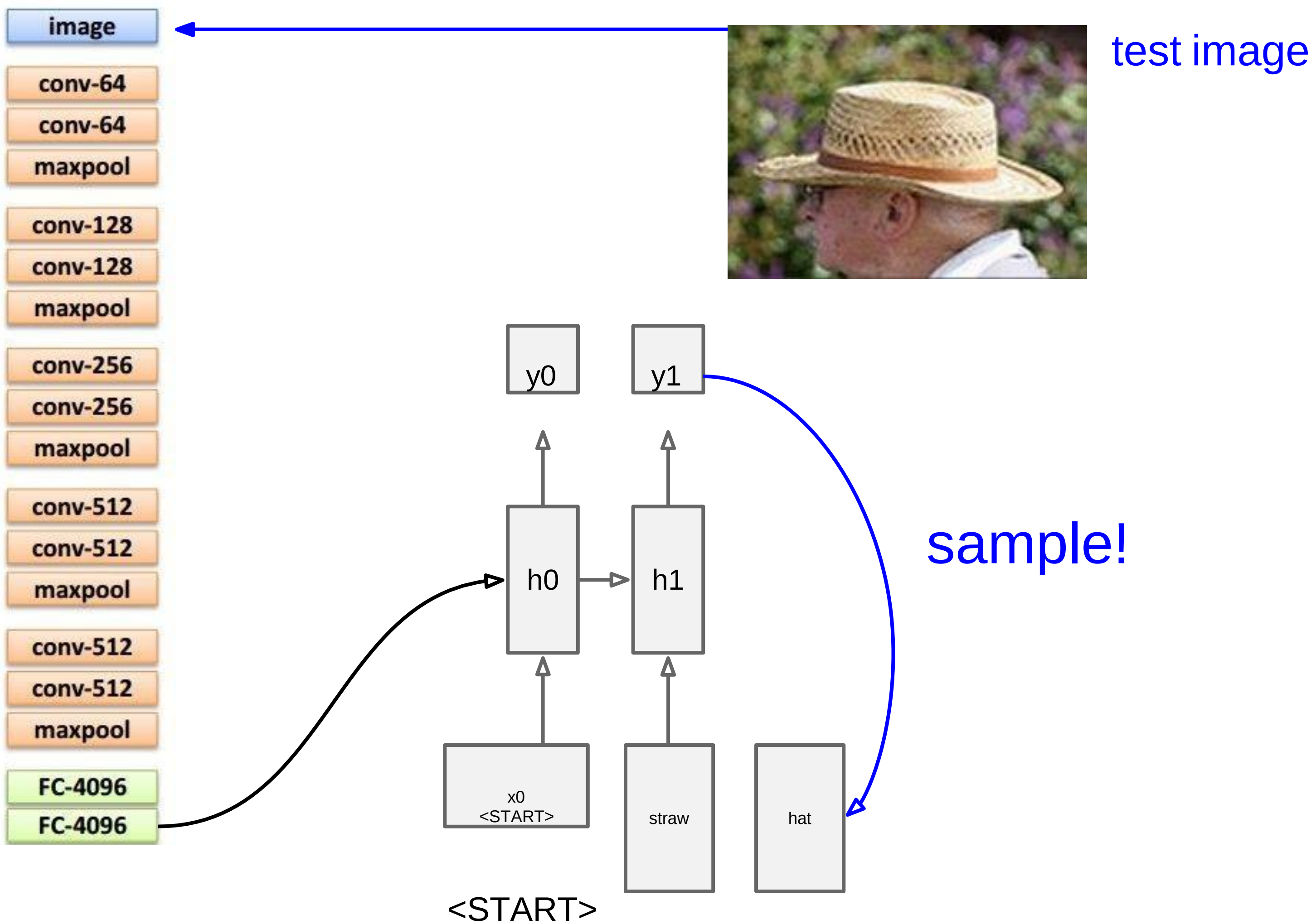
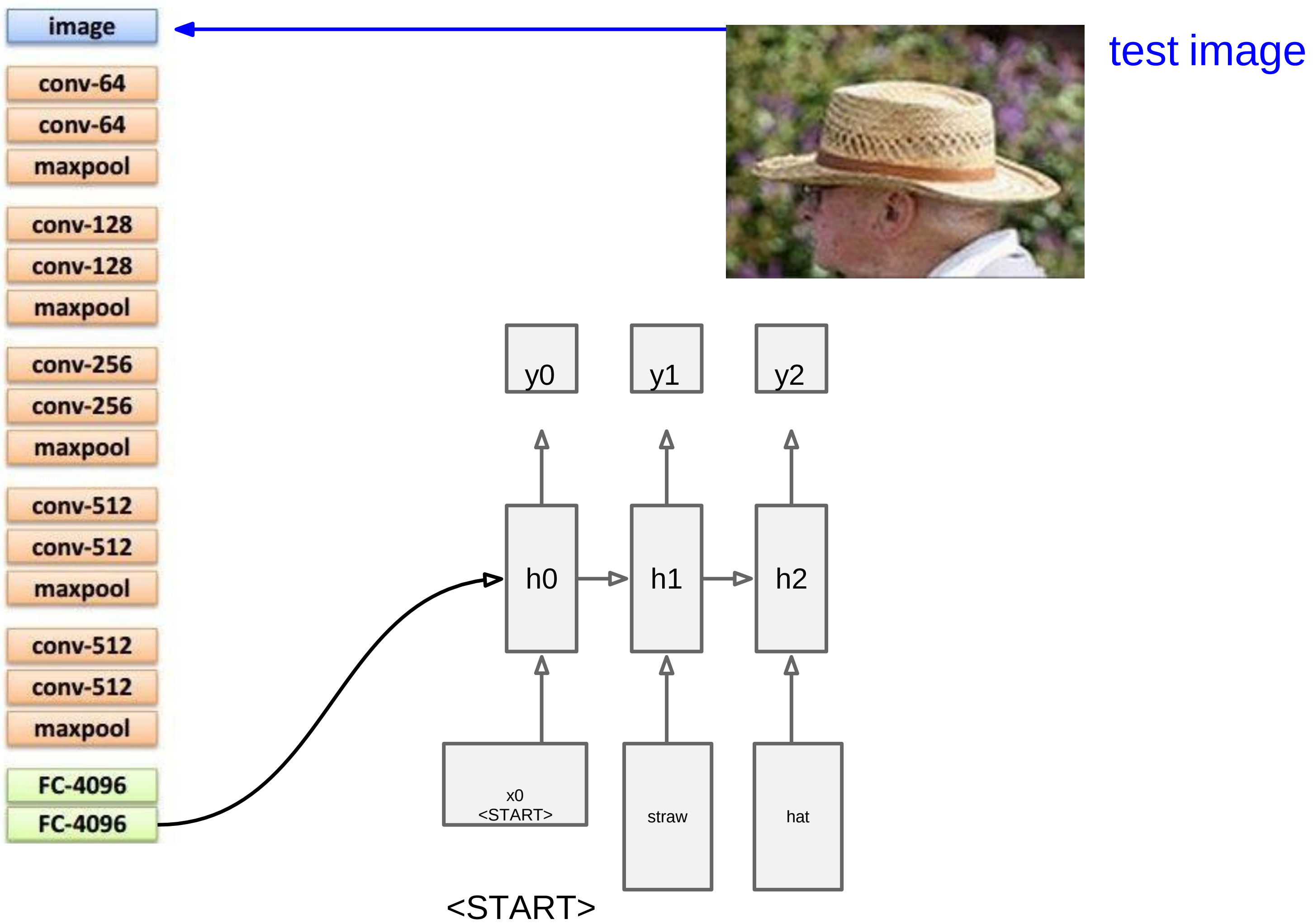
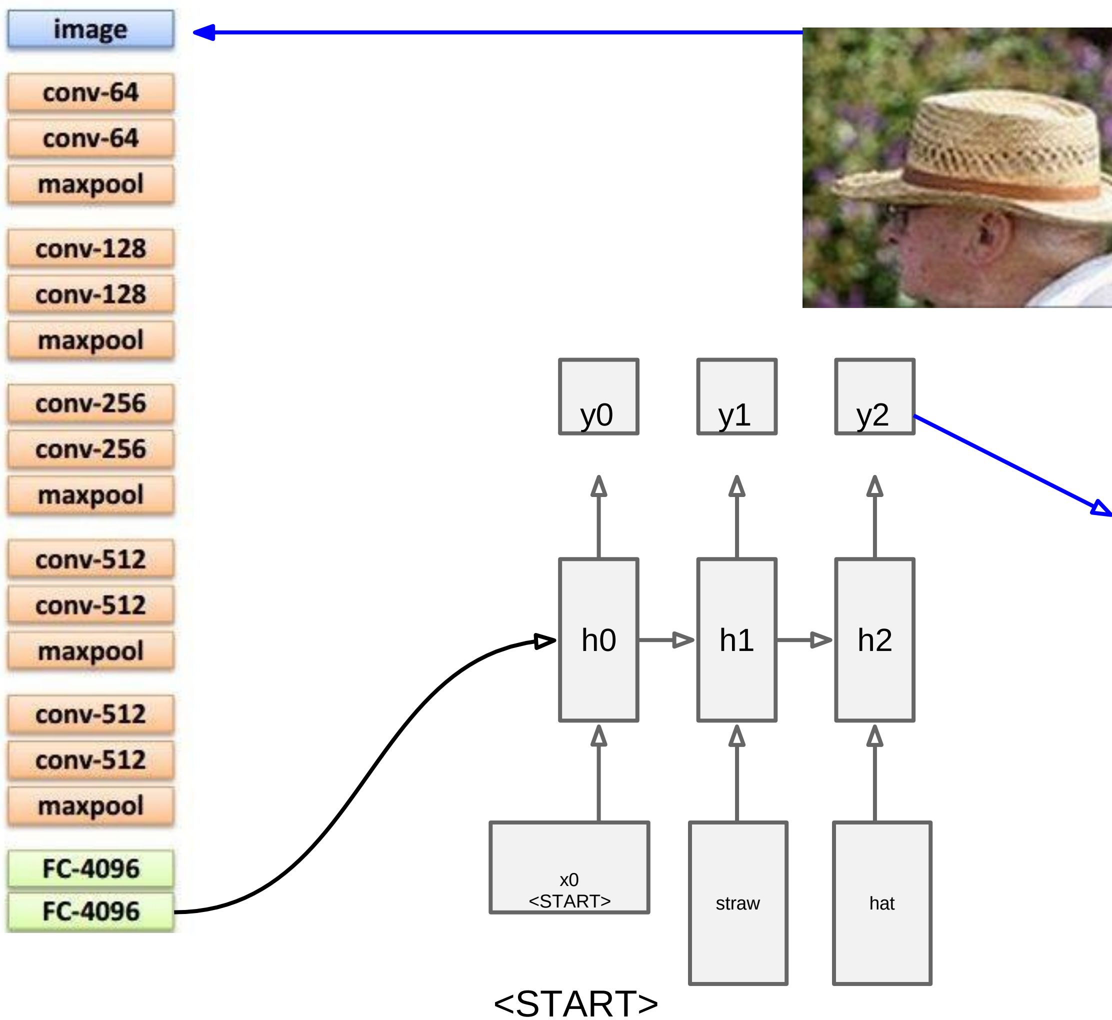


Image Captioning



Andrej Karpathy

Image Captioning



Caption generated:
“straw hat”

sample
<END> token
=> finish.

Adapted from Andrej Karpathy

Image Captioning

"man in black shirt is playing guitar."

"construction worker in orange safety vest is working on road."

"two young girls are playing with lego toy."

"boy is doing backflip on wakeboard."

"a young boy is holding a baseball bat."

"a cat is sitting on a couch with a remote control."

"a woman holding a teddy bear in front of a mirror."

"a horse is standing in the middle of a road."

Andrej Karpathy

Plan for this lecture

- Language and vision
  - Image captioning
  - Tool: Recurrent neural networks
  - Video captioning
  - Visual question answering
- Motion and video
  - Modeling and replicating motion
  - Tracking how an object moves

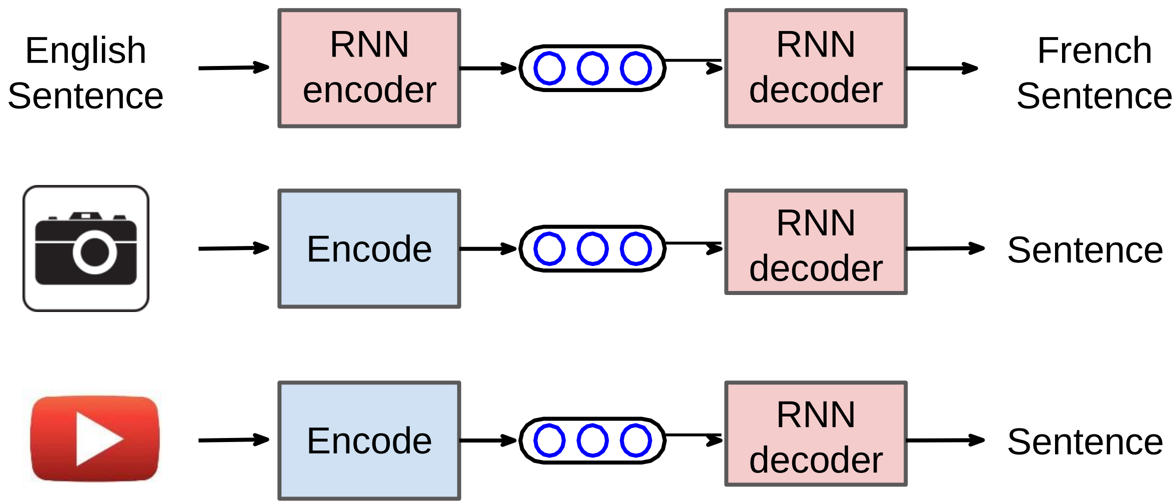
Video Captioning

Generate descriptions for events depicted in video clips

A monkey pulls a dog’s tail and is chased by the dog.

Venugopalan et al., “Translating Videos to Natural Language using Deep Recurrent Neural Networks”, NAACL-HTL 2015

Video Captioning



[Sutskever et al. NIPS'14]

[Donahue et al. CVPR'15]
[Vinyals et al. CVPR'15]

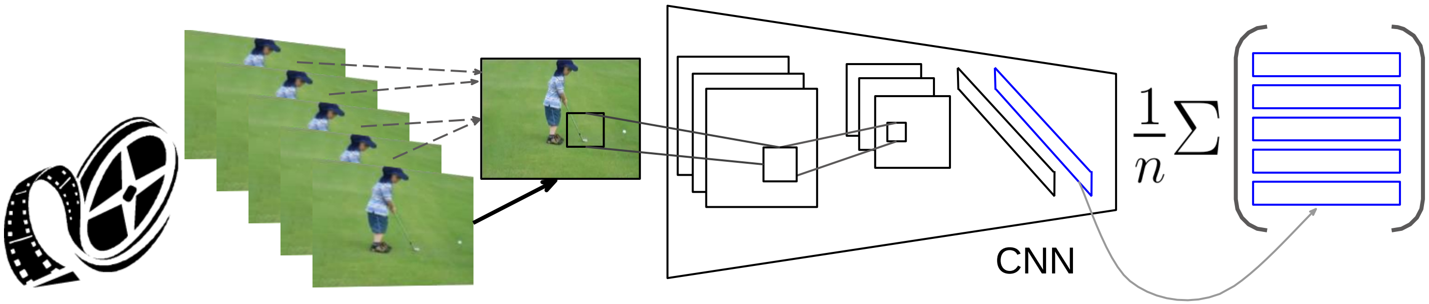
[Venugopalan et. al. NAACL'15] (this work)

Key Insight:

Generate feature representation of the video and "decode" it to a sentence

Venugopalan et al., "Translating Videos to Natural Language using Deep Recurrent Neural Networks", NAACL-HTL 2015

Video Captioning



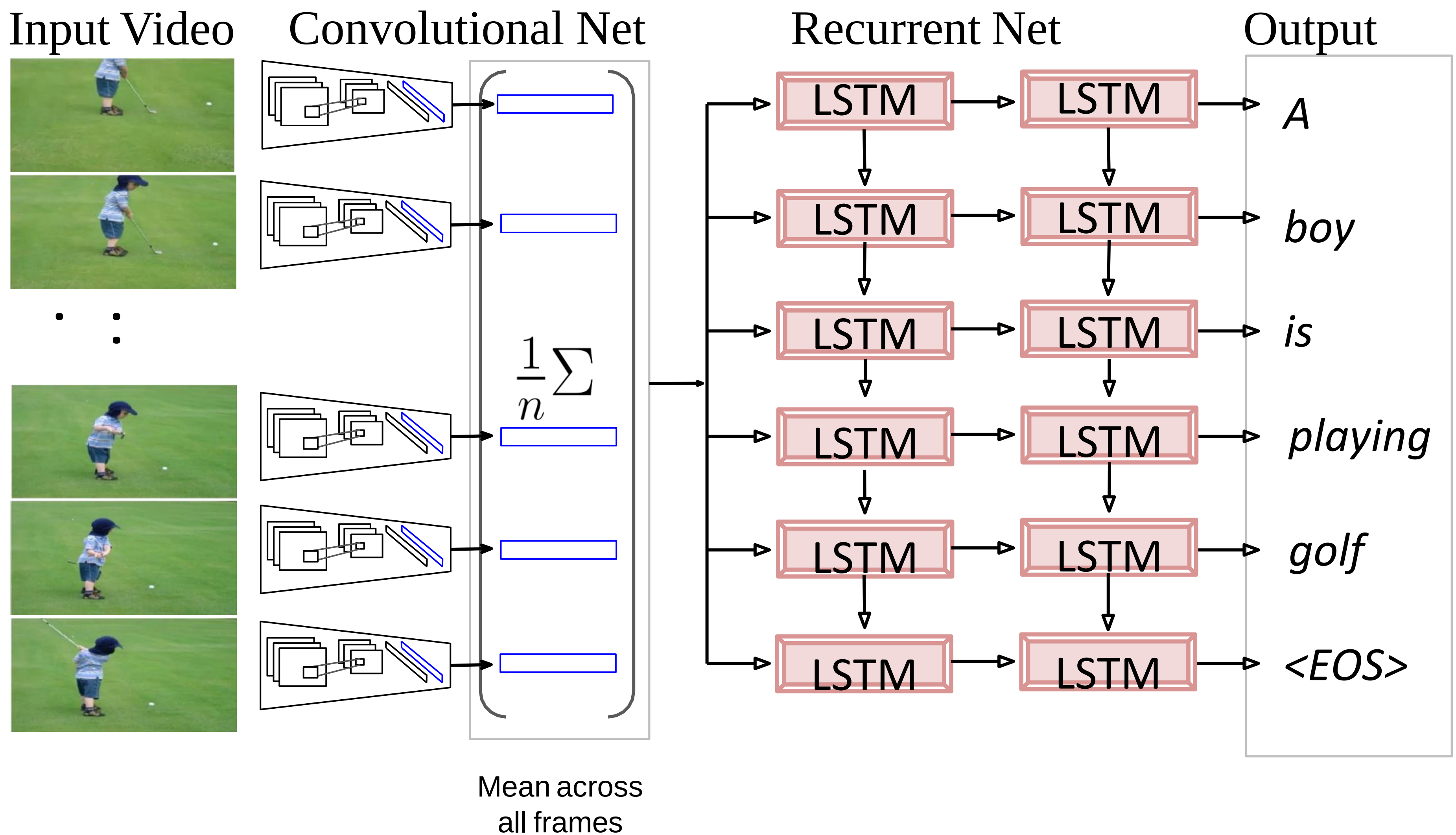
Input Video → Sample frames @1/10

Forward propagate
Output: “fc7” features
(activations before classification layer)

fc7: 4096 dimension
“feature vector”

Venugopalan et al., “Translating Videos to Natural Language using Deep Recurrent Neural Networks”, NAACL-HTL 2015

Video Captioning



Venugopalan et al., “Translating Videos to Natural Language using Deep Recurrent Neural Networks”, NAACL-HTL 2015

Video Captioning

FGM: A person is dancing with the person on the stage.
YT: A group of men are riding the forest.
I+V: **A group of people are dancing.**
GT: Many men and women are dancing in the street.

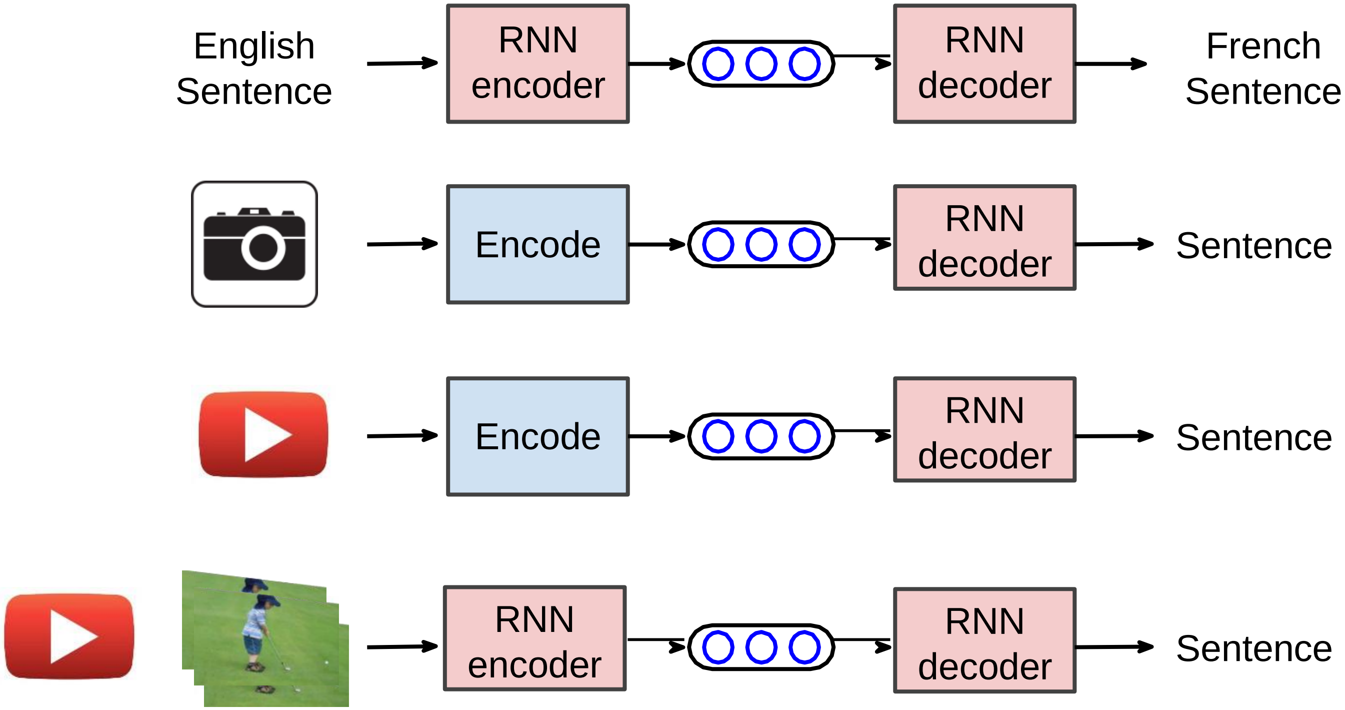
FGM: A person is cutting a potato in the kitchen.
YT: A man is slicing a tomato.
I+V: **A man is slicing a carrot.**
GT: A man is slicing carrots.

FGM: A person is walking with a person in the forest.
YT: A monkey is walking.
I+V: **A bear is eating a tree.**
GT: Two bear cubs are digging into dirt and plant matter at the base of a tree.

FGM: A person is riding a horse on the stage.
YT: A group of playing are playing in the ball.
I+V: **A basketball player is playing.**
GT: Dwayne wade does a fancy layup in an allstar game.

Venugopalan et al., “Translating Videos to Natural Language using Deep Recurrent Neural Networks”, NAACL-HTL 2015

Video Captioning



[Sutskever et al. NIPS'14]

[Donahue et al. CVPR'15]
[Vinyals et al. CVPR'15]

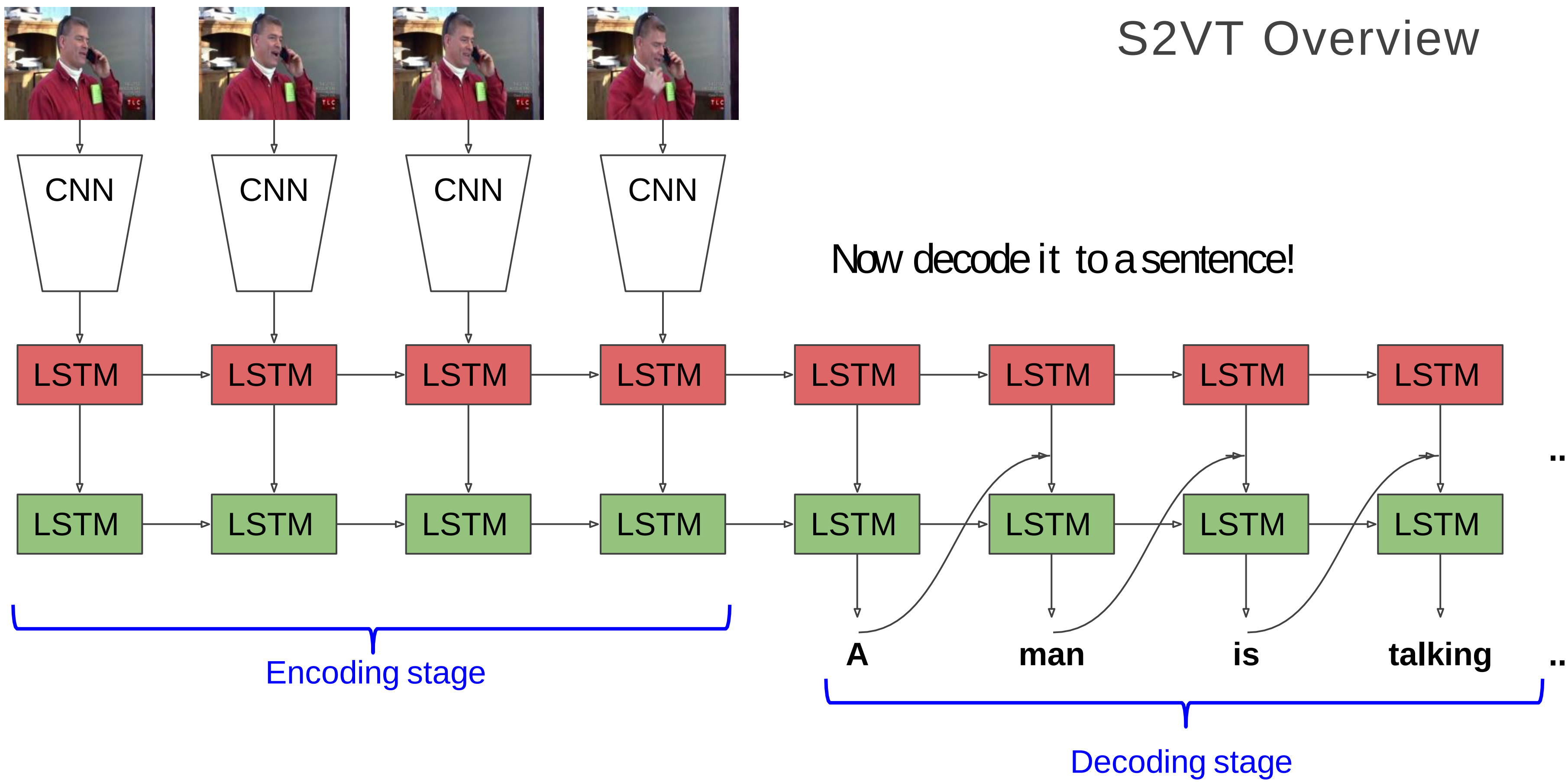
[Venugopalan et. al. NAACL'15]

[Venugopalan et. al. ICCV' 15] (this work)

3

Venugopalan et al., "Sequence to Sequence - Video to Text", ICCV 2015

Video Captioning



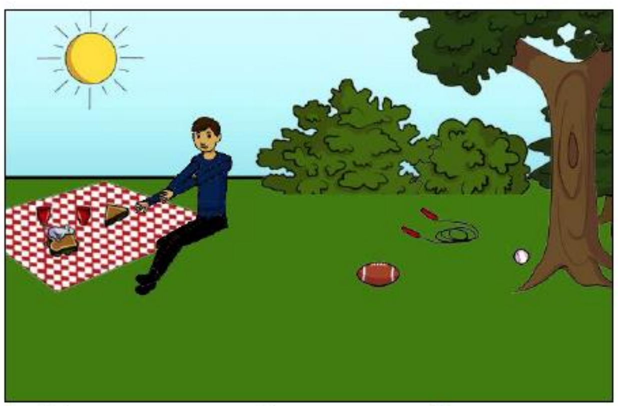
Venugopalan et al., “Sequence to Sequence - Video to Text”, ICCV 2015

Visual Question Answering (VQA)

Task: Given an image and a natural language open-ended question, generate a natural language answer.

What color are her eyes?
What is the mustache made of?

How many slices of pizza are there?
Is this a vegetarian pizza?



Is this person expecting company?
What is just under the tree?

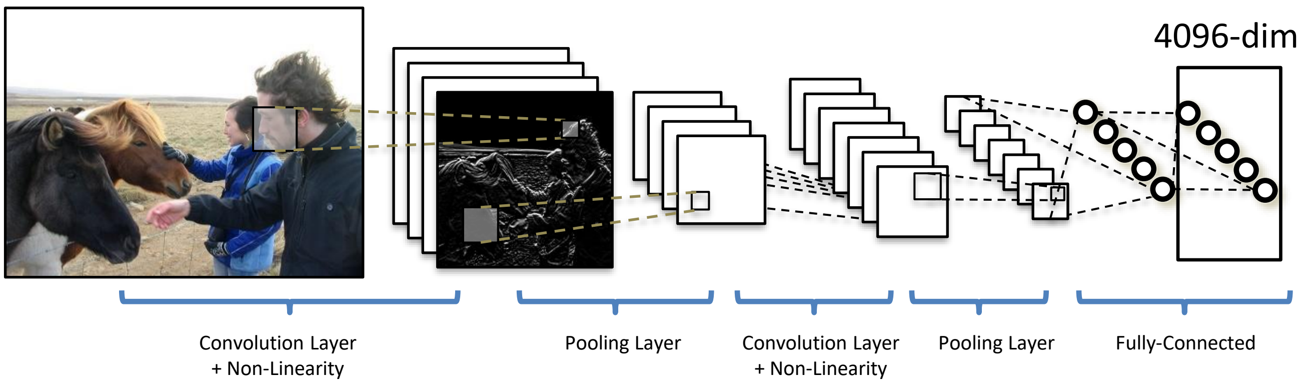
Does it appear to be rainy?
Does this person have 20/20 vision?

Agrawal et al., "[VQA: Visual Question Answering]", ICCV 2015

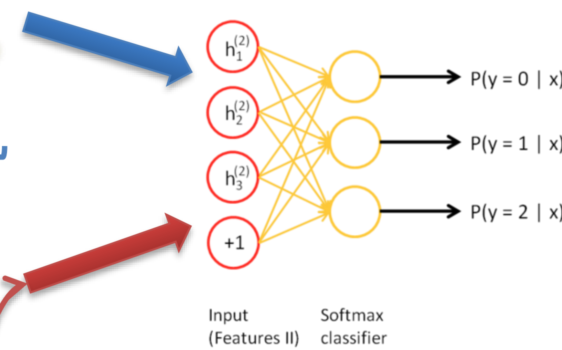
Visual Question Answering (VQA)

Image Embedding

4096-dim



Neural Network
Softmax
over top K answers

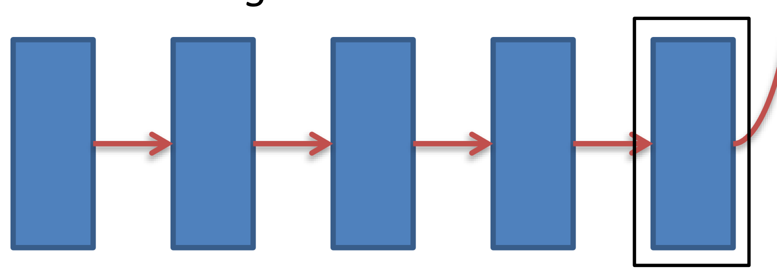


Question Embedding

"How many horses are in this image?"

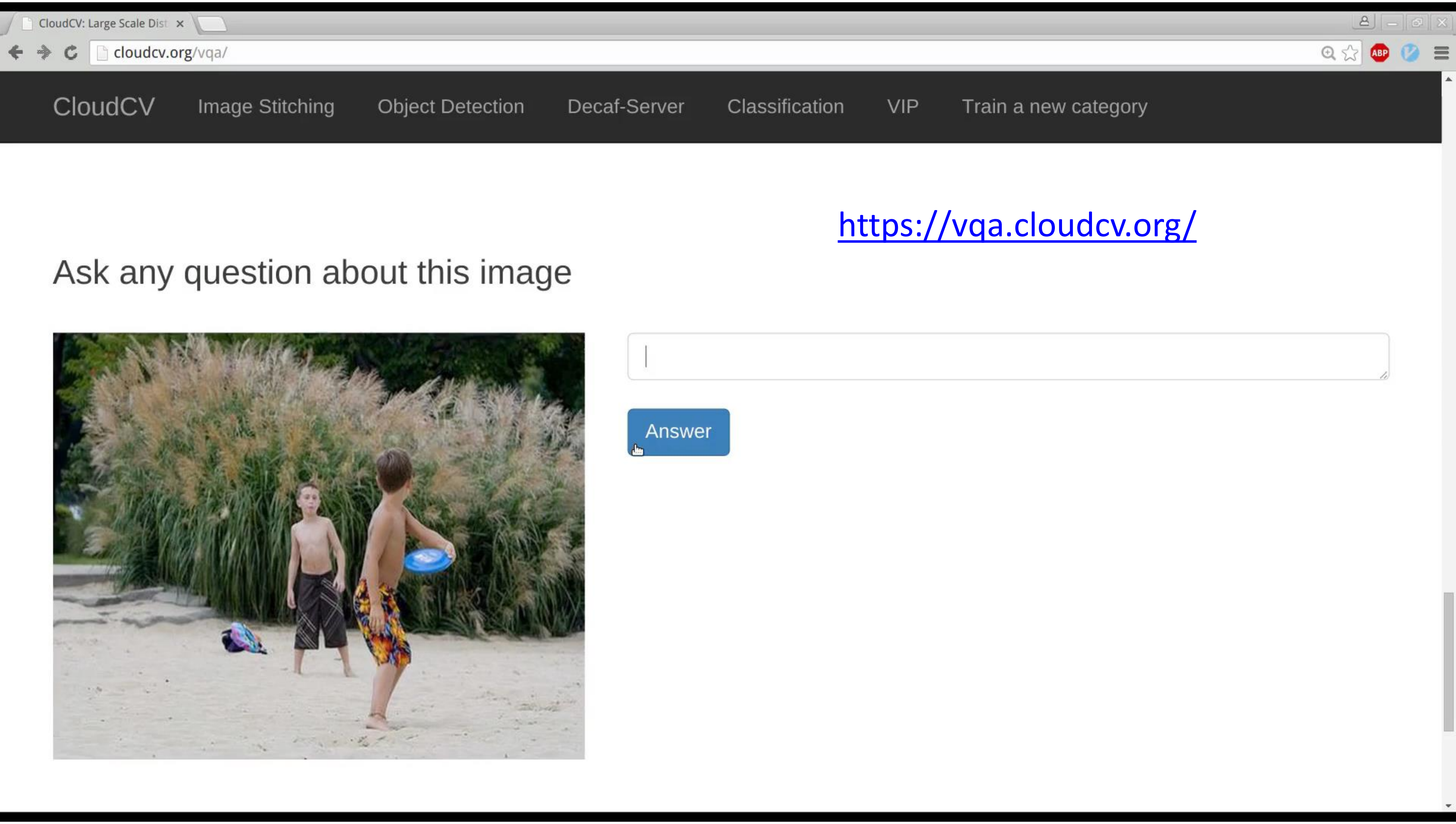
1024-dim

LSTM



Agrawal et al., "[VQA: Visual Question Answering]", ICCV 2015

Visual Question Answering (VQA)

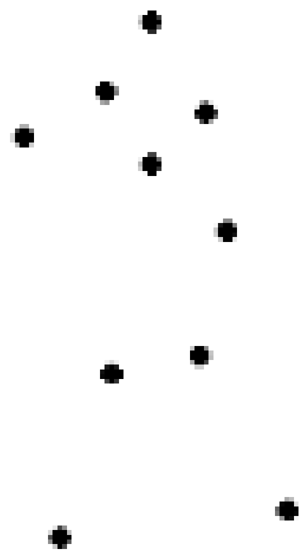


Agrawal et al., "VQA: Visual Question Answering", ICCV 2015

Plan for this lecture

- Language and vision
  - Image captioning
  - Tool: Recurrent neural networks
  - Video captioning
  - Visual question answering
- Motion and video
  - Modeling and replicating motion
  - Tracking how an object moves

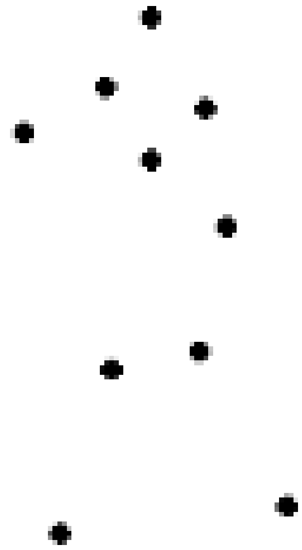
Motion: Why is it useful?



Derek Hoiem

Motion: Why is it useful?

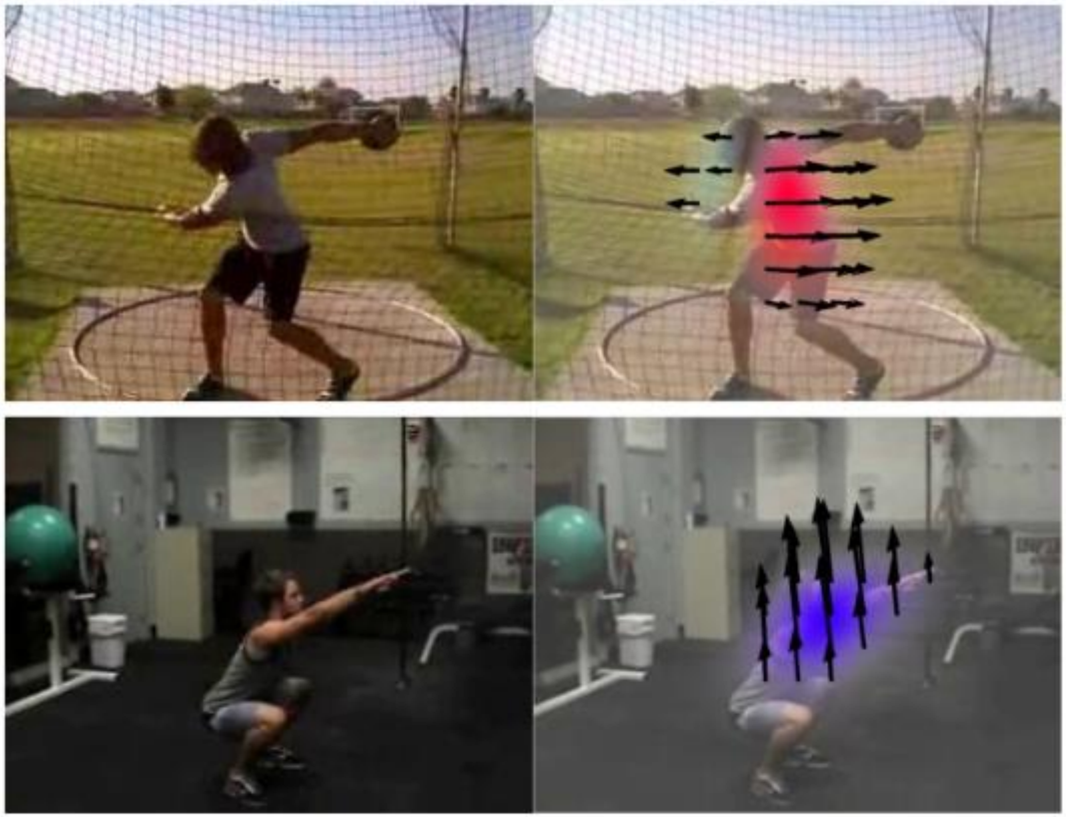
- Even “impoverished” motion data can evoke a strong percept



G. Johansson, “Visual Perception of Biological Motion and a Model For Its Analysis", *Perception and Psychophysics 14, 201-211, 1973.*

Derek Hoiem

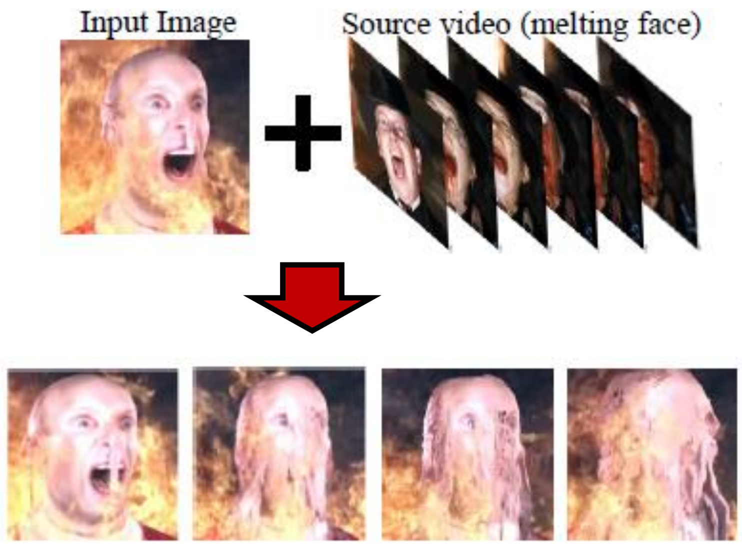
Modeling Motion: Optical Flow



(a) Input Image (b) Prediction

Walker et al., "Dense Optical Flow Prediction from a Static Scene", ICCV 2015

Transferring Motion

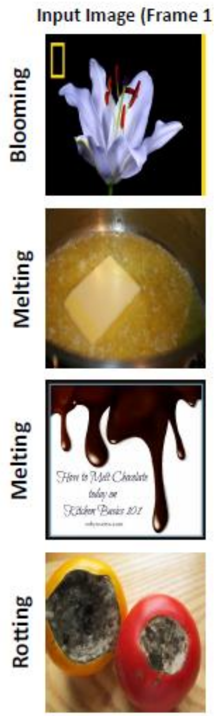


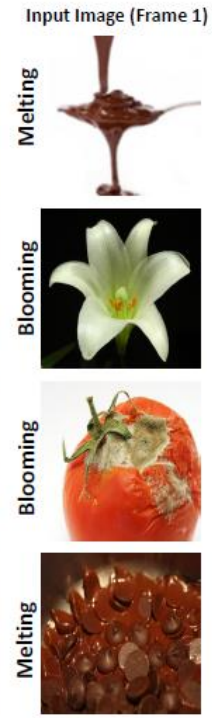
$$\mathcal{L}_{\text{flow}}(\mathbf{y}_{i-1}, \mathbf{y}_i; \mathbf{s}_{i-1}, \mathbf{s}_i) = \sum_l \frac{1}{C_l H_l W_l} \| \underbrace{\Xi(\mathbf{y}_{i-1}, \mathbf{y}_i)_l}_{\text{Optical flow in generated video}} - \underbrace{\Xi(\mathbf{s}_{i-1}, \mathbf{s}_i)_l}_{\text{Optical flow in source video}} \|_2^2$$

Key idea: Generate videos with similar flow patterns as source videos (+ many details).

Thomas, Song and Kovashka

Transferring Motion

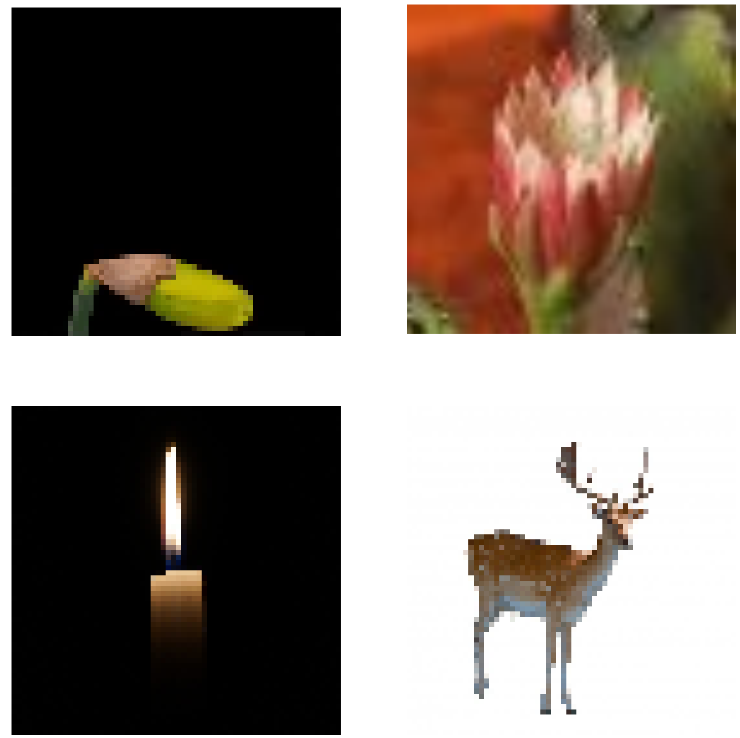




Thomas, Song and Kovashka

Transferring Motion

Baking



Blooming

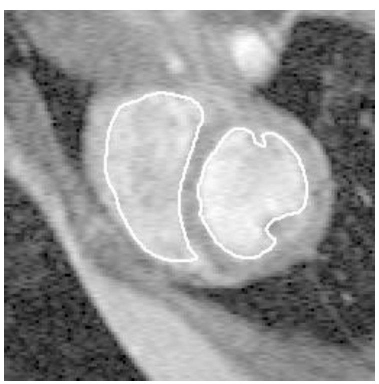
Thomas, Song and Kovashka

Tracking: some applications

Body pose tracking, activity recognition

Censusing a bat population

Video-based interfaces



Medical apps

Surveillance

Tracking examples

Traffic: https://www.youtube.com/watch?v=DiZHQ4peqjg

Soccer: http://www.youtube.com/watch?v=ZqQIItFAnxg

Face: http://www.youtube.com/watch?v=i_bZNVmhJ2o

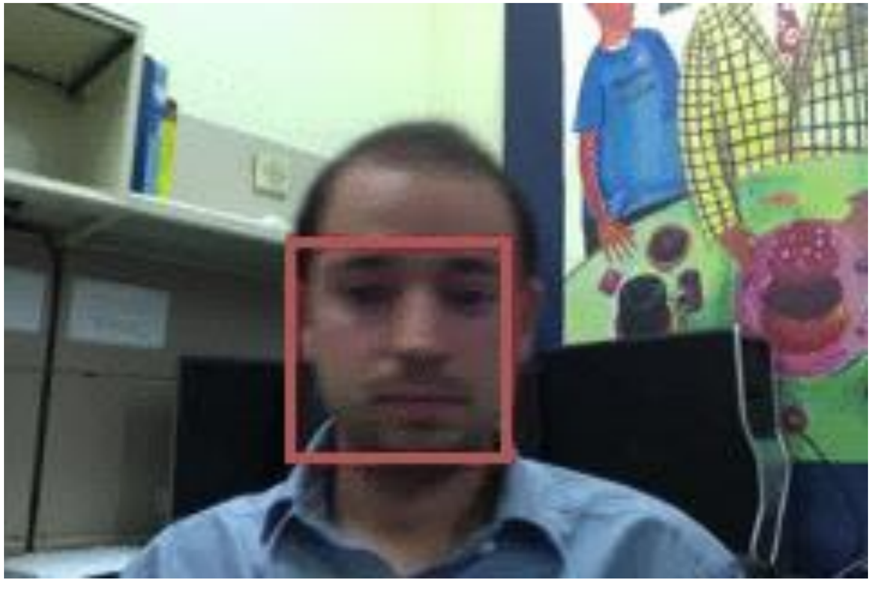
Body: https://www.youtube.com/watch?v=_Ahy0Gh69-M

Eye: http://www.youtube.com/watch?v=NCtYdUEMotg

Gaze: http://www.youtube.com/watch?v=-G6Rw5cU-1c

Things that make visual tracking difficult

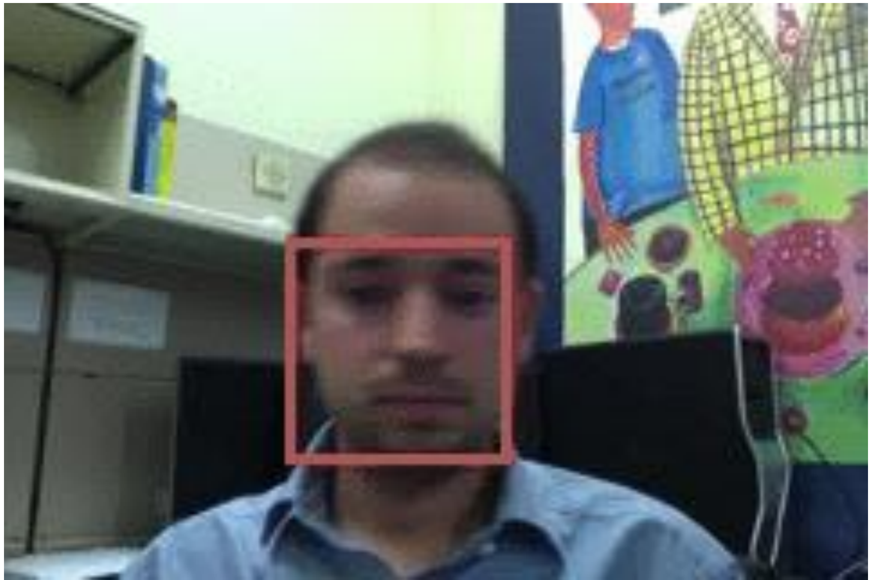
- Erratic movements, moving very quickly
- Occlusions, leaving and coming back
- Surrounding similar-looking objects



Adapted from Amin Sadeghi

Strategies for tracking

- Tracking by repeated detection
  - Works well if object is easily detectable (e.g., face or colored glove) and there is only one
  - Need some way to link up detections



Amin Sadeghi

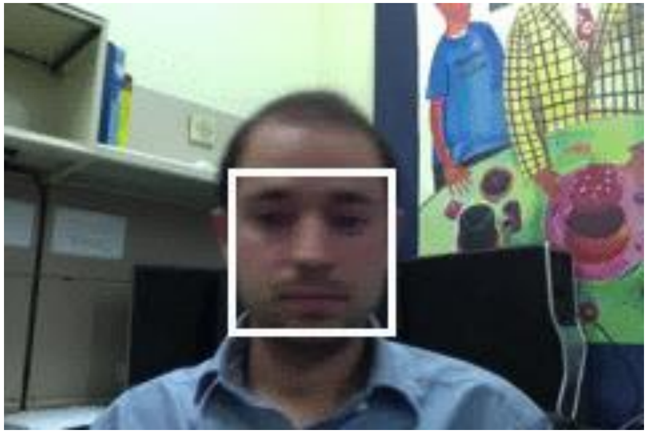
Strategies for tracking

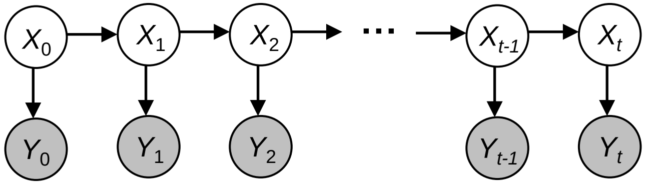
- Tracking w/ dynamics: Using model of expected motion, *predict* object location in next frame
  - Restrict search for the object
  - Measurement noise is reduced by trajectory smoothness
  - Robustness to missing or weak observations
  - Assumptions: Camera is not moving instantly to new viewpoint, objects do not disappear/reappear in different places in the scene

Amin Sadeghi

General model for tracking

- *State* X: The actual state of the moving object that we want to estimate but cannot observe
  - E.g. position, velocity
- *Observations* Y: Our actual measurement or observation of state X, which can be very noisy
- At each time t, the state changes to X_t and we get a new observation Y_t
- Our goal is to recover the most likely state X_t given:
  - All observations so far, i.e. $y_1, y_2, \ldots, y_t$
  - Knowledge about dynamics of state transitions





Adapted from Amin Sadeghi and Kristen Grauman

Steps of tracking

- **Prediction:** What is the next state of the object given *past* measurements?

$$P\left(X_t \middle| Y_0 = y_0, \ldots, Y_{t-1} = y_{t-1}\right)$$

Kristen Grauman

Steps of tracking

- **Prediction:** What is the next state of the object given *past* measurements?

$$P\left(X_t \middle| Y_0 = y_0, \ldots, Y_{t-1} = y_{t-1}\right)$$

- **Correction:** Compute an updated estimate of the state from prediction and measurements

$$P\left(X_t \middle| Y_0 = y_0, \ldots, Y_{t-1} = y_{t-1}, Y_t = y_t\right)$$

Kristen Grauman

Problem statement

- We have models for

  Likelihood of next state given current state (dynamics model):

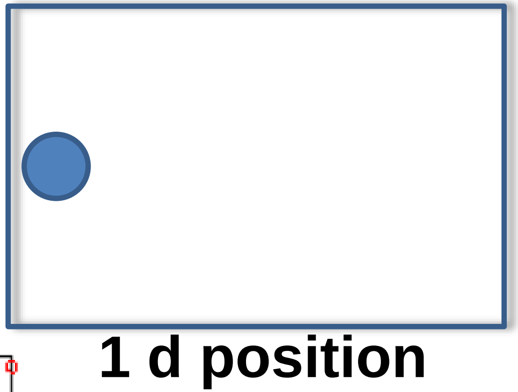
  $$P\left(X_t \middle| X_{t-1}\right)$$

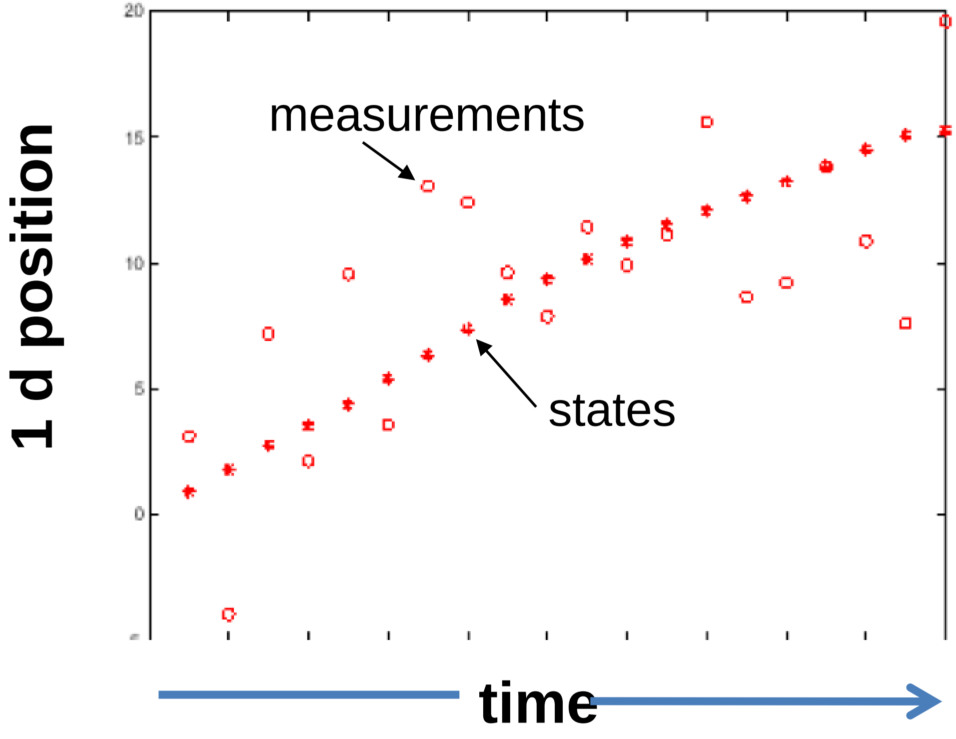
  Likelihood of observation given the state (observation or measurement model):

  $$P\left(Y_t \middle| X_t\right)$$

- We want to recover, for each t: $P\left(X_t \middle| y_0, \ldots, y_t\right)$

Example: Constant velocity (1D points)





Kristen Grauman

Example: Constant velocity (1D points)

- State vector: position *p* and velocity *v*

$$x_t = \begin{bmatrix} p_t \\ v_t \end{bmatrix} \qquad \begin{aligned} p_t &= p_{t-1} + (\Delta t) v_{t-1} + \varepsilon \\ v_t &= v_{t-1} + \xi \end{aligned}$$

$$x_t = \boxed{D_t x_{t-1} + noise} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} p_{t-1} \\ v_{t-1} \end{bmatrix} + noise$$

- Measurement is position only

$$y_t = \boxed{M x_t + noise} = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} p_t \\ v_t \end{bmatrix} + noise$$

Kristen Grauman

Prediction and correction

Prediction:

$$P(X_t \mid y_0, \dots, y_{t-1}) = \int \underbrace{P(X_t \mid X_{t-1})}_{\text{dynamics model}} \underbrace{P(X_{t-1} \mid y_0, \dots, y_{t-1})}_{\text{corrected estimate from previous step}} dX_{t-1}$$

Correction:

t++

$$P(X_t \mid y_0, \dots, y_t) = \frac{\overbrace{P(y_t \mid X_t)}^{\text{observation model}} \overbrace{P(X_t \mid y_0, \dots, y_{t-1})}^{\text{predicted estimate}}}{\int P(y_t \mid X_t) P(X_t \mid y_0, \dots, y_{t-1}) dX_t}$$

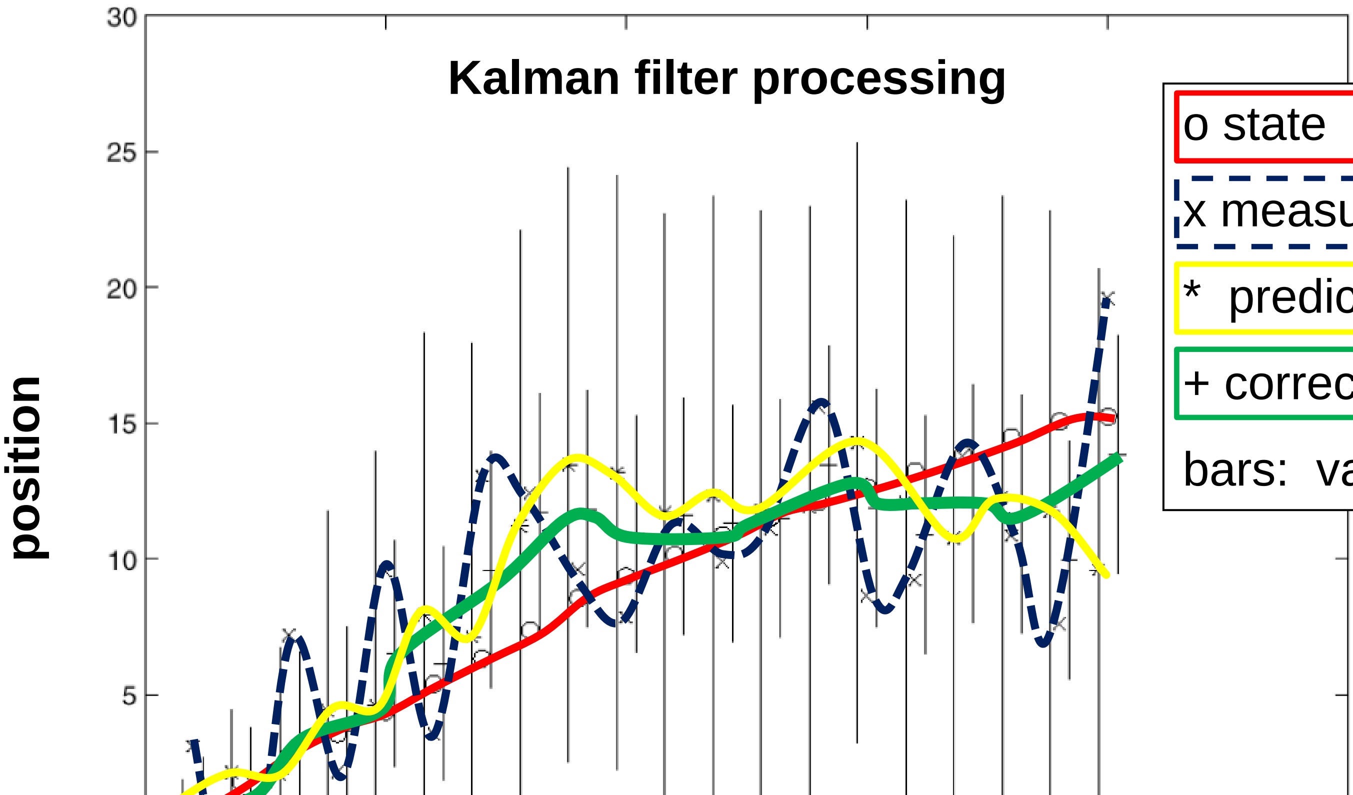
Adapted from Amin Sadeghi

The Kalman filter

- Linear dynamics model: state undergoes linear transformation plus Gaussian noise
- Observation model: measurement is linearly transformed state plus Gaussian noise
- Predicted/corrected state distributions are Gaussian
  - You only need to maintain the mean and covariance
  - The calculations are easy

Amin Sadeghi

Example w/ constant velocity



o state

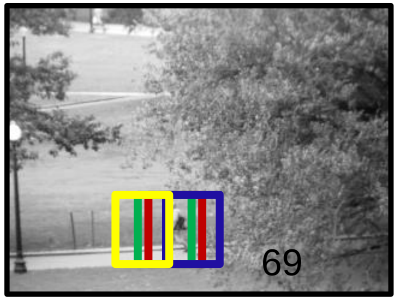
x measurement

* predicted mean estimate

+ corrected mean estimate

bars: variance estimates

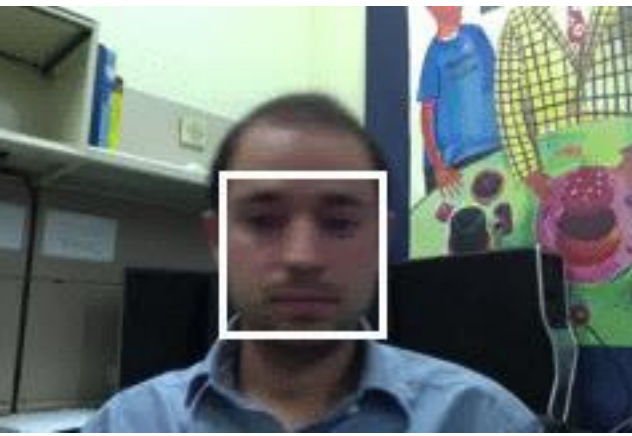
Time t



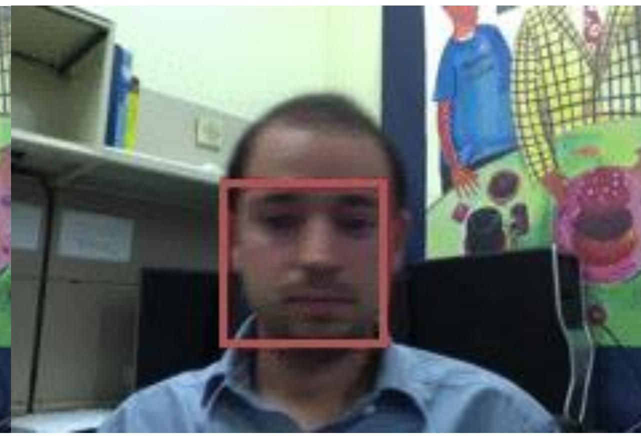
Time t+1

Adapted from Kristen Grauman

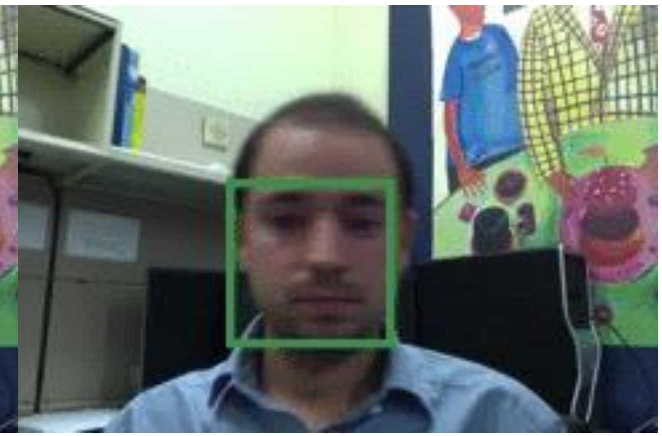
Example w/ constant velocity



Ground Truth



Observation



Correction