Computer Vision 2
WS 2018/19

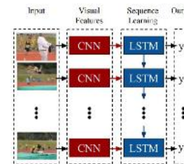
Part 17 – CNNs for Video Analysis I
15.01.2019

Prof. Dr. Bastian Leibe

RWTH Aachen University, Computer Vision Group
http://www.vision.rwth-aachen.de

Course Outline

- Single-Object Tracking
- Bayesian Filtering
- Multi-Object Tracking
- Visual Odometry
- Visual SLAM & 3D Reconstruction
  - Online SLAM methods
  - Full SLAM methods
- Deep Learning for Video Analysis
  - CNNs for video analysis
  - Optical flow
  - Video object segmentation

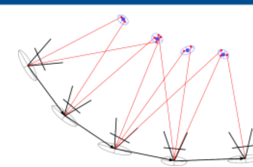


Topics of This Lecture

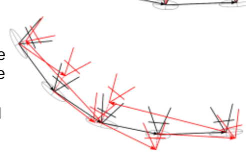
- Recap: Full SLAM methods
- CNNs for Video Analysis
  - Motivation
  - Example: Video classification
- CNN + RNN
  - RNN, LSTM
  - Example: Video captioning
- Matching and correspondence estimation
  - Metric learning
  - Correspondence networks

Recap: Full SLAM Approaches

- SLAM graph optimization:
  - Joint optimization for poses and map elements from image observations of map elements and control inputs

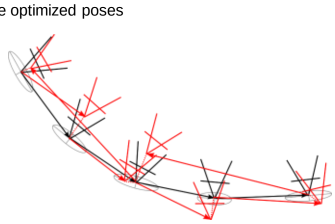


- Pose graph optimization:
  - Optimization of poses from relative pose constraints deduced from the image observations
  - Map recovered from the optimized poses

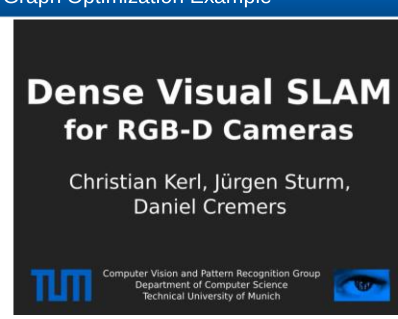


Pose Graph Optimization

- Optimization of poses
  - From relative pose constraints deduced from the image observations
  - Map recovered from the optimized poses
- Deduce relative constraints between poses from image observations, e.g.,
  - 8-point algorithm
  - Direct image alignment



Pose Graph Optimization Example



Kerl et al., Dense Visual SLAM for RGB-D Cameras, IROS 2013

Topics of This Lecture

- Recap: Full SLAM methods
- CNNs for Video Analysis
  - Motivation
  - Example: Video classification
- CNN + RNN
  - RNN, LSTM
  - Example: Video captioning
- Matching and correspondence estimation
  - Metric learning
  - Correspondence networks

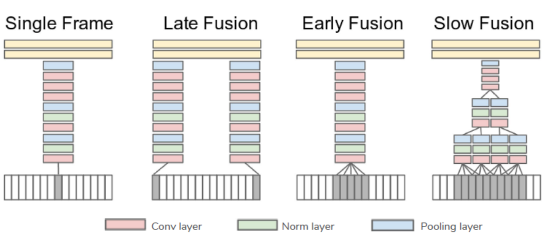
Video Analysis with CNNs

- Modeling perspective
  - What architecture to use to best capture temporal patterns?
- Computational perspective
  - Video processing is expensive!
  - How to reduce computation cost without sacrificing accuracy

Slide credit: Fei-Fei Li

Large-Scale Video Classification with CNNs

- Architecture
  - Different ways to fuse features from multiple frames



Slide credit: Fei-Fei Li
Image source: Andrej Karpathy

Large-Scale Video Classification with CNNs

- Computational cost
  - Reduce spatial dimension to reduce model complexity
  - Multi-resolution: low-res context + high-res foveate

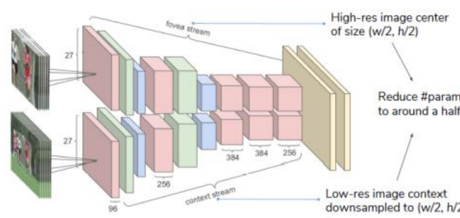
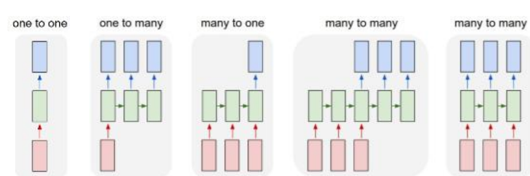


Image source: Andrej Karpathy

Topics of This Lecture

- Recap: Full SLAM methods
- CNNs for Video Analysis
  - Motivation
  - Example: Video classification
- CNN + RNN
  - RNN, LSTM
  - Example: Video captioning
- Matching and correspondence estimation
  - Metric learning
  - Correspondence networks

Recap: Recurrent Networks



- Feed-forward networks
  - Simple neural network structure: 1-to-1 mapping of inputs to outputs
- Recurrent Neural Networks
  - Generalize this to arbitrary mappings

Image source: Andrej Karpathy

Recap: RNNs

- RNNs are regular NNs whose hidden units have additional forward connections over time.
  - You can unroll them to create a network that extends over time.
  - When you do this, keep in mind that the weights for the hidden units are shared between temporal layers.

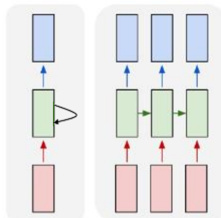
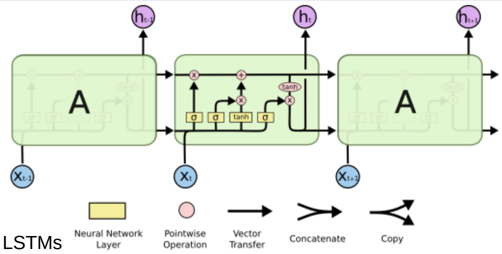


Image source: Andrej Karpathy

Extension: Long Short-Term Memory (LSTM)

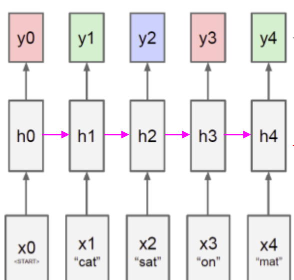


- LSTMs
  - Inspired by the design of memory cells
  - Each module has 4 layers, interacting in a special way.
  - Effect: LSTMs can learn longer dependencies (~100 steps) than RNNs

Image source: Christopher Olah, http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Recap: RNNs for Text Generation

- RNN for text generation



10,001D class scores (Softmax over 10k words and a special <END> token)

$$\mathbf{y}_4 = \mathbf{W}_{hy}\mathbf{h}_4$$

Hidden layer (e.g., 500D vectors)

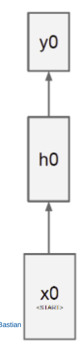
$$\mathbf{h}_4 = \max\{0, \mathbf{W}_{xh}\mathbf{x}_4 + \mathbf{W}_{hh}\mathbf{h}_3\}$$

Slide credit: Andrej Karpathy, Fei-Fei Li
Image source: Andrej Karpathy

Recap: RNNs for Text Generation

- Training this on a lot of sentences would give us a language model.
- I.e., a way to predict

$p(\textit{next word} \mid \textit{previous words})$

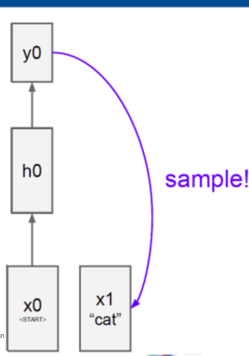


Slide credit: Andrej Karpathy, Fei-Fei Li

Recap: RNNs for Text Generation

- Training this on a lot of sentences would give us a language model.
- I.e., a way to predict

$p(\textit{next word} \mid \textit{previous words})$

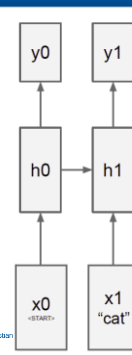


Slide credit: Andrej Karpathy, Fei-Fei Li

Recap: RNNs for Text Generation

- Training this on a lot of sentences would give us a language model.
- I.e., a way to predict

$p(\textit{next word} \mid \textit{previous words})$

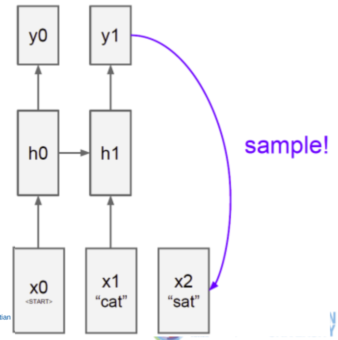


Slide credit: Andrej Karpathy, Fei-Fei Li

Recap: RNNs for Text Generation

- Training this on a lot of sentences would give us a language model.
- I.e., a way to predict

  $p(\textit{next word} \mid \textit{previous words})$

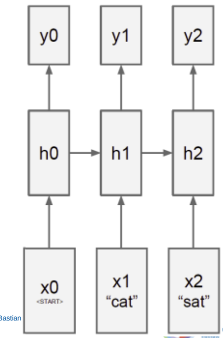


Slide credit: Andrei Karpathy, Fei-Fei Li

Recap: RNNs for Text Generation

- Training this on a lot of sentences would give us a language model.
- I.e., a way to predict

  $p(\textit{next word} \mid \textit{previous words})$

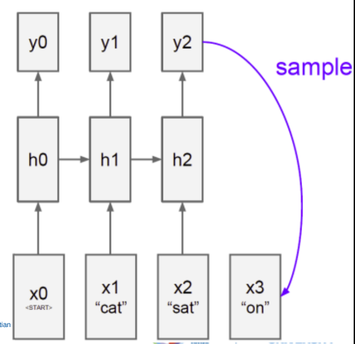


Slide credit: Andrei Karpathy, Fei-Fei Li

Recap: RNNs for Text Generation

- Training this on a lot of sentences would give us a language model.
- I.e., a way to predict

  $p(\textit{next word} \mid \textit{previous words})$

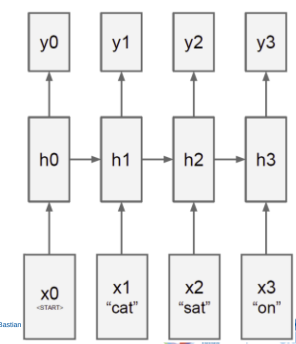


Slide credit: Andrei Karpathy, Fei-Fei Li

Recap: RNNs for Text Generation

- Training this on a lot of sentences would give us a language model.
- I.e., a way to predict

  $p(\textit{next word} \mid \textit{previous words})$

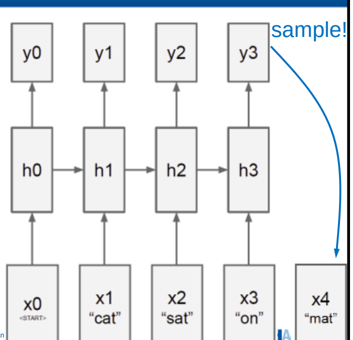


Slide credit: Andrei Karpathy, Fei-Fei Li

Recap: RNNs for Text Generation

- Training this on a lot of sentences would give us a language model.
- I.e., a way to predict

  $p(\textit{next word} \mid \textit{previous words})$

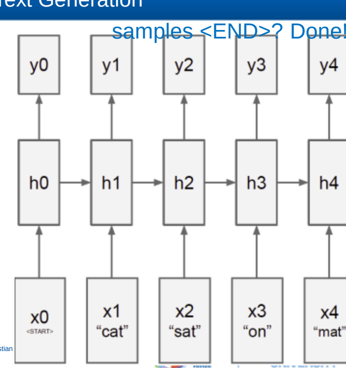


Slide credit: Andrei Karpathy, Fei-Fei Li

Recap: RNNs for Text Generation

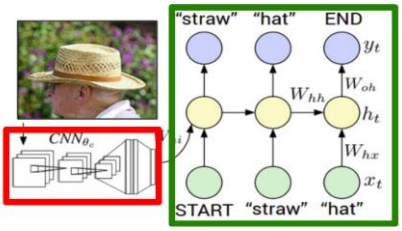
- Training this on a lot of sentences would give us a language model.
- I.e., a way to predict

  $p(\textit{next word} \mid \textit{previous words})$



Slide credit: Andrei Karpathy, Fei-Fei Li

Applications: Image Tagging



- Simple combination of CNN and RNN
  - Use CNN to define initial state $\mathbf{h}_0$ of an RNN.
  - Use RNN to produce text description of the image.

Slide adapted from Andrei Karpathy

Applications: Image Tagging

- Setup
  - Train on corpus of images with textual descriptions
  - E.g. Microsoft CoCo
    - 120k images
    - 5 sentences each

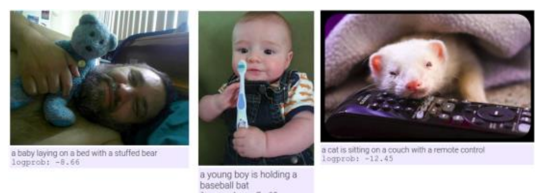
Slide adapted from Andrei Karpathy

Results: Image Tagging

Spectacular results!

Slide adapted from Andrei Karpathy

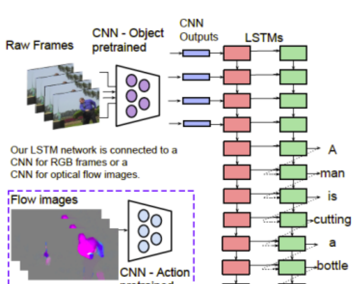
Results: Image Tagging



- Wrong, but one can still see why those results were selected...

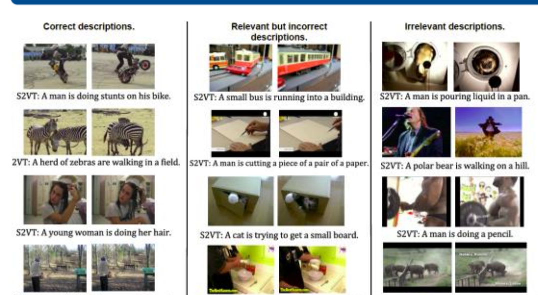
Slide adapted from Andrei Karpathy

Application: Video to Text Description



Source: Subhashini Venugopalan, ICCV'15

Video-to-Text Results



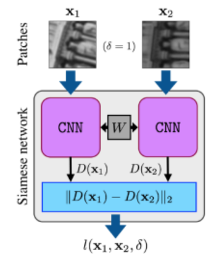
Source: Subhashini Venugopalan, ICCV'15

Topics of This Lecture

- Recap: Full SLAM methods
- CNNs for Video Analysis
  - Motivation
  - Example: Video classification
- CNN + RNN
  - RNN, LSTM
  - Example: Video captioning
- Matching and correspondence estimation
  - Metric learning
  - Correspondence networks

Learning Similarity Functions

- Siamese Network
  - Present the two stimuli to two identical copies of a network (with shared parameters)
  - Train them to output similar values if the inputs are (semantically) similar.
- Used for many matching tasks
  - Face identification
  - Stereo estimation
  - Optical flow
  - …

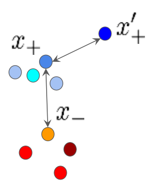


Metric Learning: Contrastive Loss

- Mapping an image to a metric embedding space
  - Metric space: distance relationship = class membership

$$\|f(x) - f(x_+)\| \to 0$$

$$\|f(x) - f(x_-)\| \geq m$$

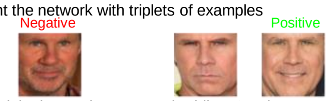


Yi et al., LIFT: Learned Invariant Feature Transform, ECCV 16

Slide credit: Christopher Choy

Metric Learning: Triplet Loss

- Learning a discriminative embedding
  - Present the network with triplets of examples



  - Apply triplet loss to learn an embedding $f(\cdot)$ that groups the positive example closer to the anchor than the negative one.

$$\|f(x_i^a) - f(x_i^p)\|_2^2 < \|f(x_i^a) - f(x_i^n)\|_2^2$$

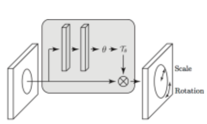
⇒ Used ntification

Patch Normalization with Spatial Transformer Nets

- Patch Normalization
  - Key component of local feature matching
  - Finding the scale and rotation
  - Invariant to perspective transformation

[SIFT patch normalization]

- Spatial Transformer Network
  - Adaptively apply transfomation

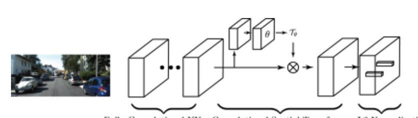


[Spatial Transformer Network]

Slide credit: Christopher Choy
Jaderberg et al., Spatial Transformer Network, NIPS 2015

Universal Correspondence Network

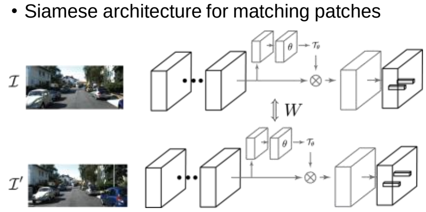
- Computing a patch descriptor



Slide credit: Christopher Choy

Universal Correspondence Network

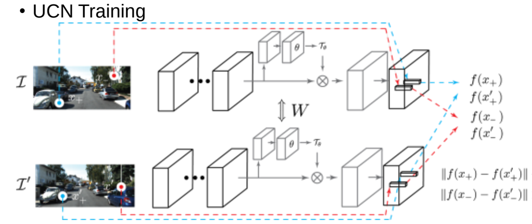
- Siamese architecture for matching patches



Slide credit: Christopher Choy

Universal Correspondence Network

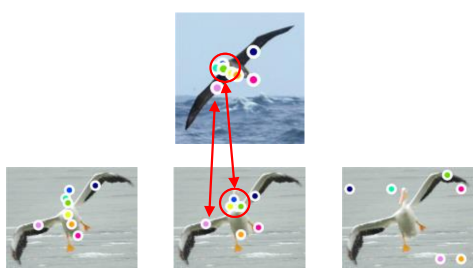
- UCN Training



- Contrastive loss $\|f(x_+) - f(x'_+)\| \rightarrow 0$
  $\|f(x_-) - f(x'_-)\| > m$

Slide credit: Christopher Choy

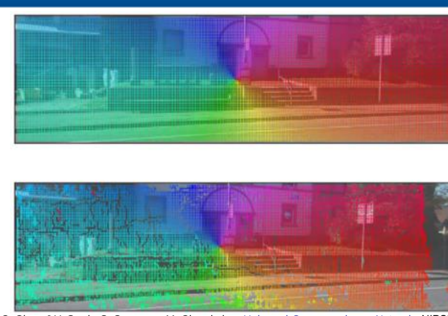
Semantic Correspondences with UCN



Ground truth UCN VGG Conv4

Slide credit: Christopher Choy

Exact Correspondences with UCN (Disparity Estimation)



C. Choy, J.Y. Gwak, S. Savarese, M. Chandraker, Universal Correspondence Network, NIPS'16

Slide credit: Christopher Choy

References and Further Reading

- RNNs
  - R. Pascanu, T. Mikolov, Y. Bengio, On the difficulty of training recurrent neural networks, JMLR, Vol. 28, 2013.
  - A. Karpathy, The Unreasonable Effectiveness of Recurrent Neural Networks, blog post, May 2015.
- LSTM
  - S. Hochreiter , J. Schmidhuber, Long short-term memory, Neural Computation, Vol. 9(8): 1735–1780, 1997.
  - A. Graves, Generating Sequences With Recurrent Neural Networks, ArXiV 1308.0850v5, 2014.
  - C. Olah, Understanding LSTM Networks, blog post, August 2015.