LSTM and its variants for visual recognition

Xiaodan Liang
xdliang328@gmail.com
Sun Yat-sen University

Outline

- Context Modelling with CNN
- LSTM and its Variants
- LSTM Architecture Variants
- Application in Semantic Segmentation

Outline

- Context Modelling with CNN
- LSTM and its Variants
- LSTM Architecture Variants
- Application in Semantic Segmentation

Visual Recognition

中山大學
SUN YAT-SEN UNIVERSITY

➢ Semantic Segmentation

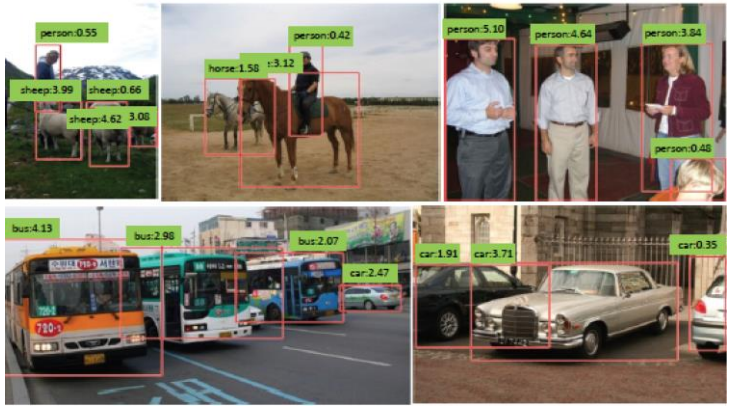
Object Segmentation

Object Parsing

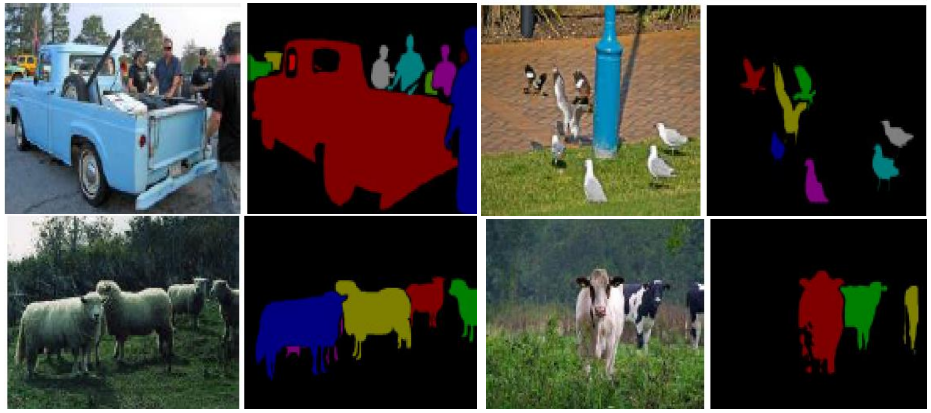
RGB/RGB-D Scene Labelling

➢ Object Recognition

Object Detection

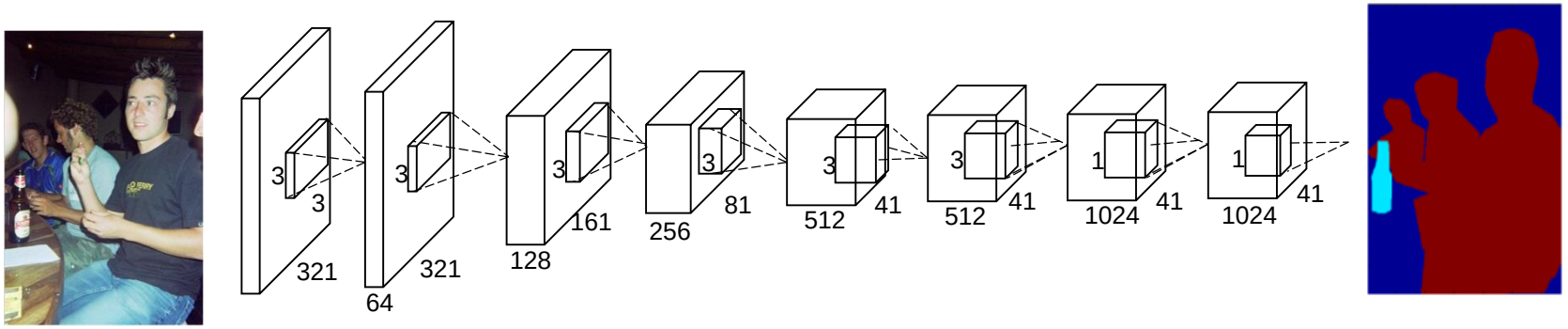


Instance-level Object Segmentation

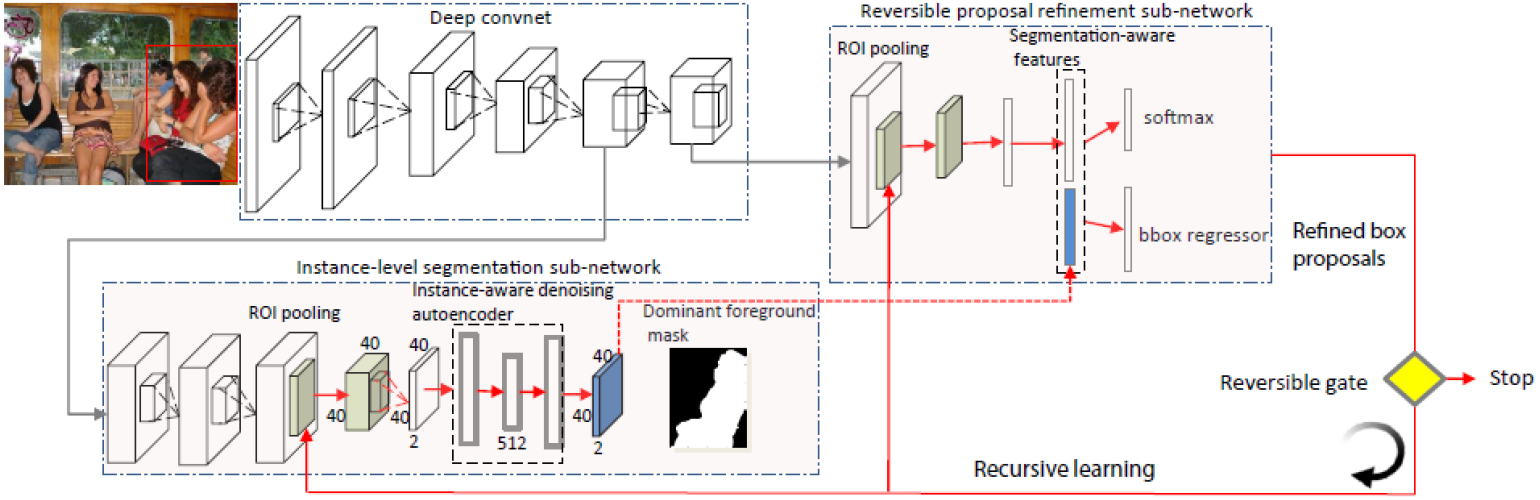


Traditional CNN architectures

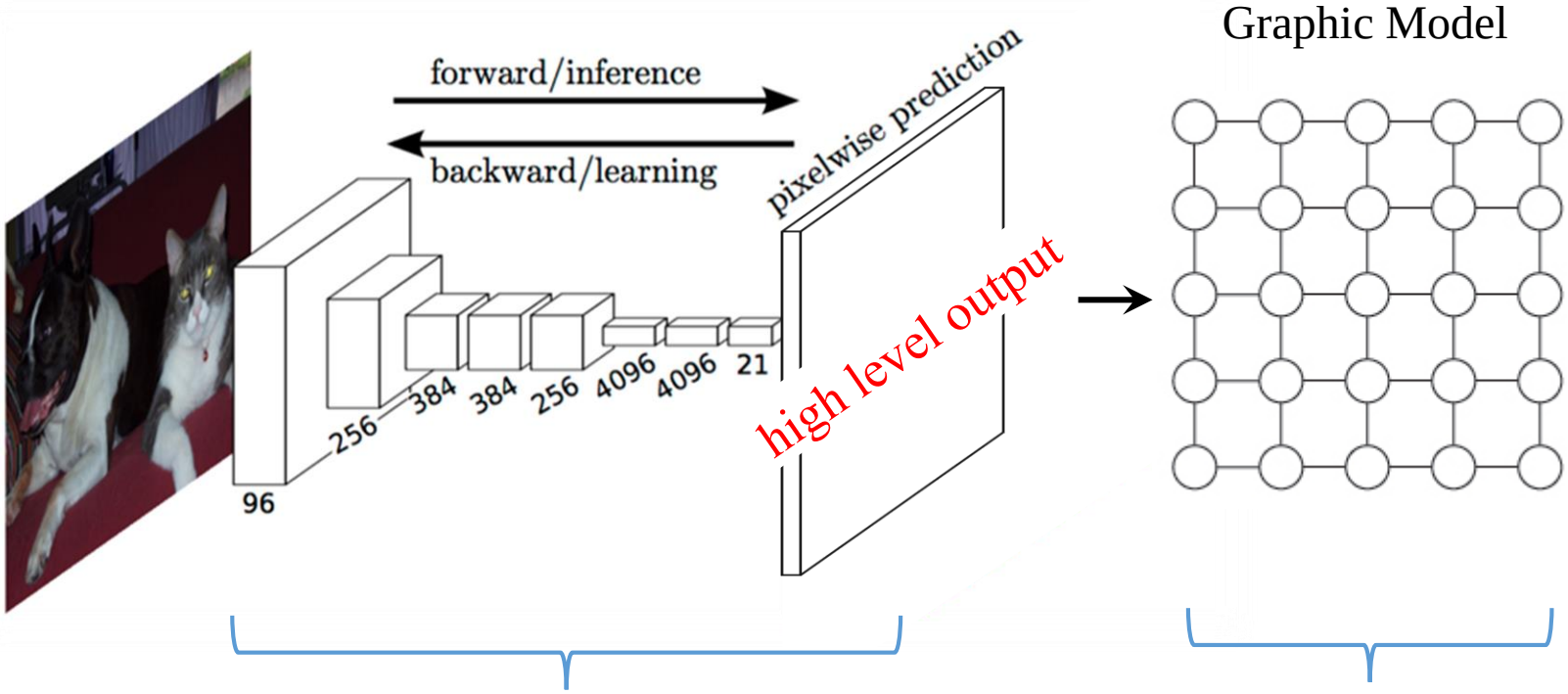
- Fully convolutional networks for semantic segmentation



- Proposal-based object recognition (Xiaodan Liang et al. CVPR 2016)



Context Modeling with Traditional CNN architectures



Local context modeling with limited reception field

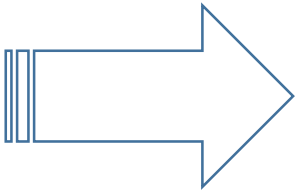
Inference on confidence maps
instead of explicitly improving features

Traditional basic CNN architectures

中山大學
SUN YAT-SEN UNIVERSITY

Limitations:

- Fixed-sized inputs
- Fixed amount of computational steps
- Fixed-sized outputs
- Local convolutional filters



Limited local context modelling via conv filters !!!

Global perspective?

Ignore correlations among instances !!!

Capturing correlation?

Outline

- Context Modelling with CNN
- LSTM and its Variants
- LSTM Architecture Variants
- Application in Semantic Segmentation

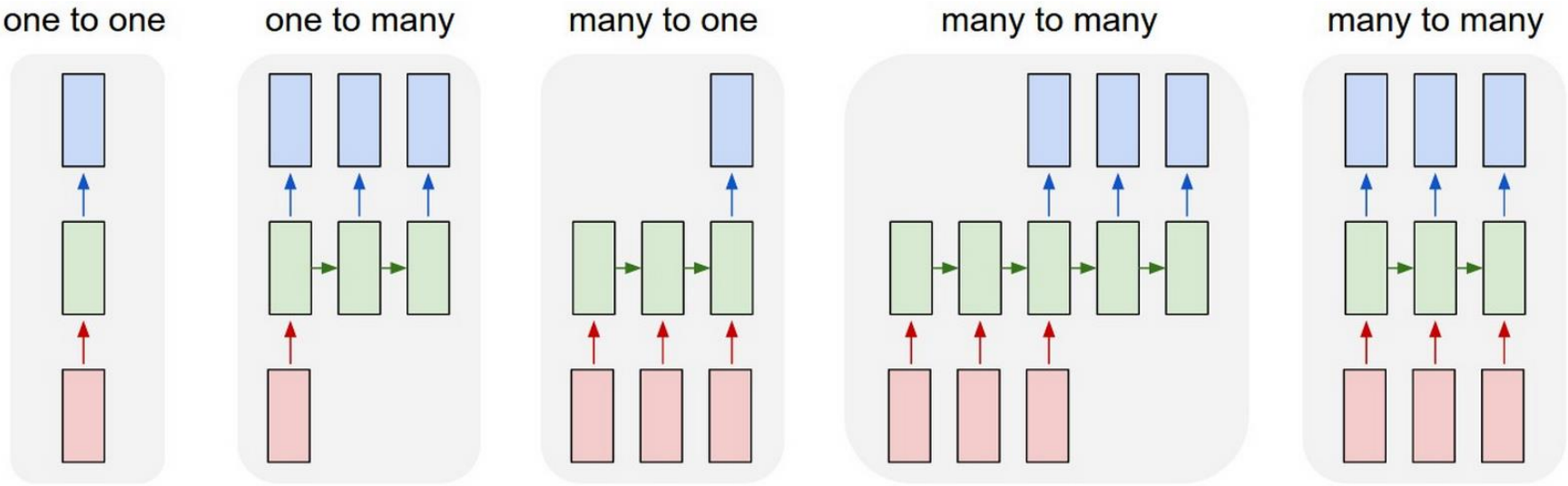
Recurrent neural networks for sequential prediction

SUN YAT-SEN UNIVERSITY

- ✓ allow operate over *sequences* of inputs with *variable* steps

$$h_t = \sigma(W^{hh} h_{t-1} + W^{hx} x_t)$$

$$y_t = softmax(W^S h_t)$$



The Unreasonable Effectiveness of Recurrent Neural Networks,Andrej Karpathy's blog.

Recurrent neural networks for sequential prediction

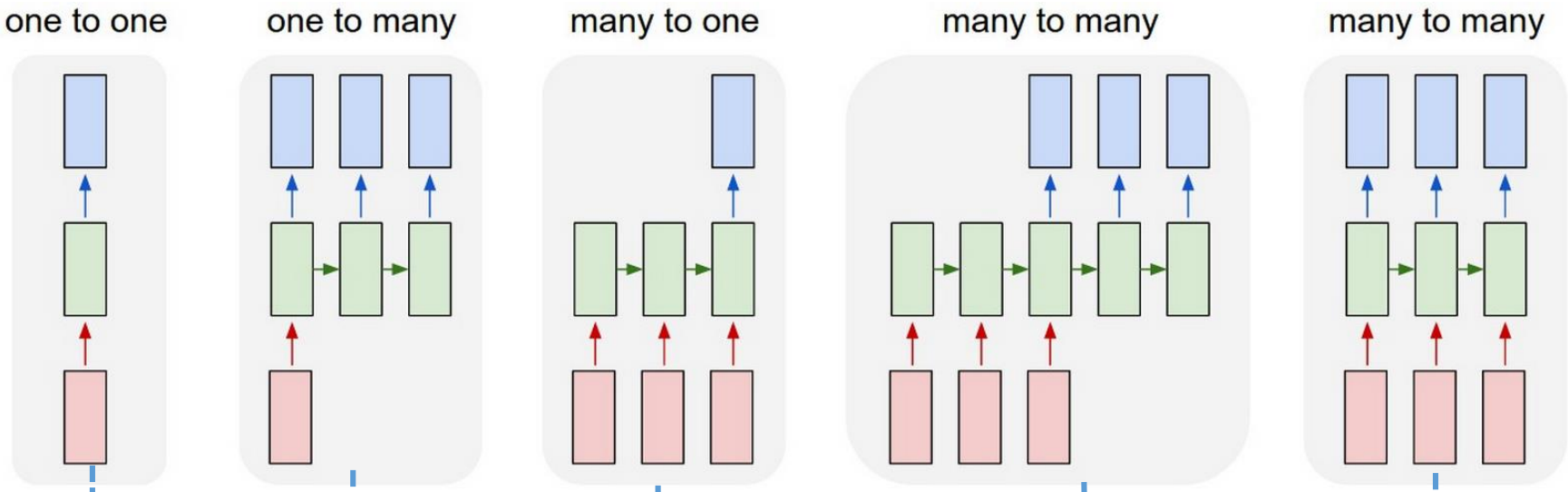


Image classification

image captioning

sentiment analysis
or video recognition

Machine Translation

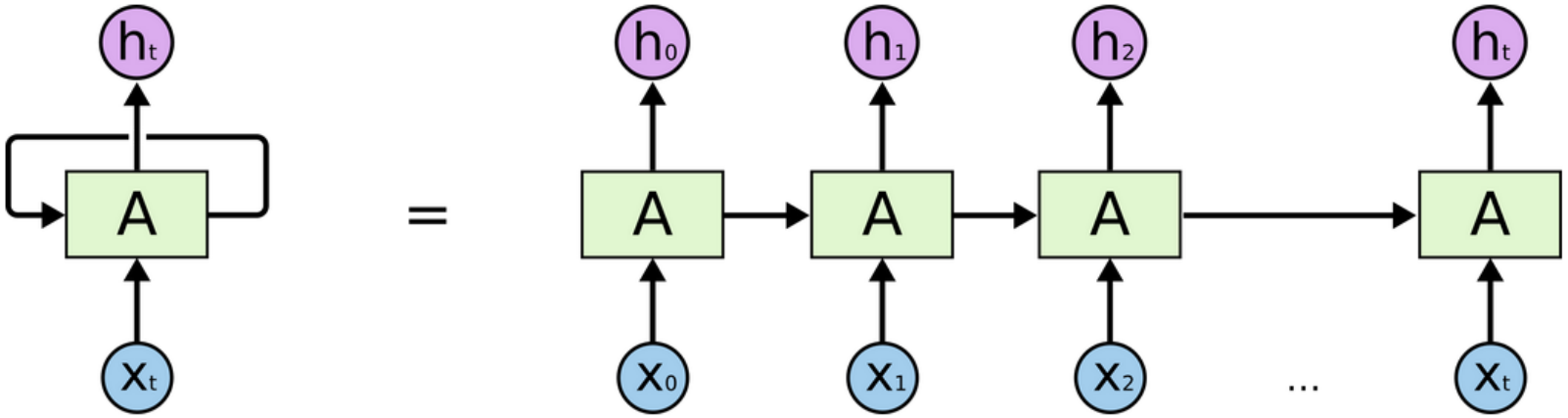
Semantic segmentation
Object recognition

The Unreasonable Effectiveness of Recurrent Neural Networks,Andrej Karpathy's blog.

Recurrent neural networks for sequential prediction

中山大學 SUN YAT-SEN UNIVERSITY

Unrolling recurrent neural networks for back-propagation



RNNs are hard to train with back-propagation

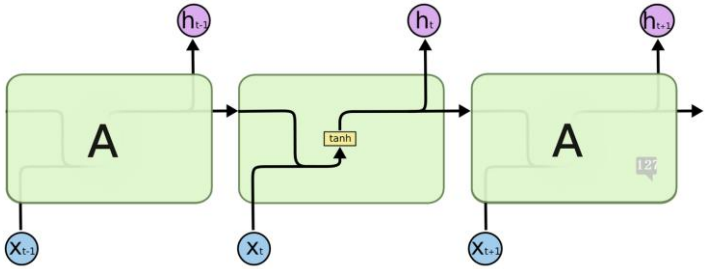
- Vanishing gradient problems (Hochreiter 1991; Bengio et al., 1994)
- Has trouble learning “long-term dependencies as the depth grows

Understanding LSTM Networks, colah’s blog

Long Short-Term Memory (LSTM) Horchreiter & Schmidhuber (1997)

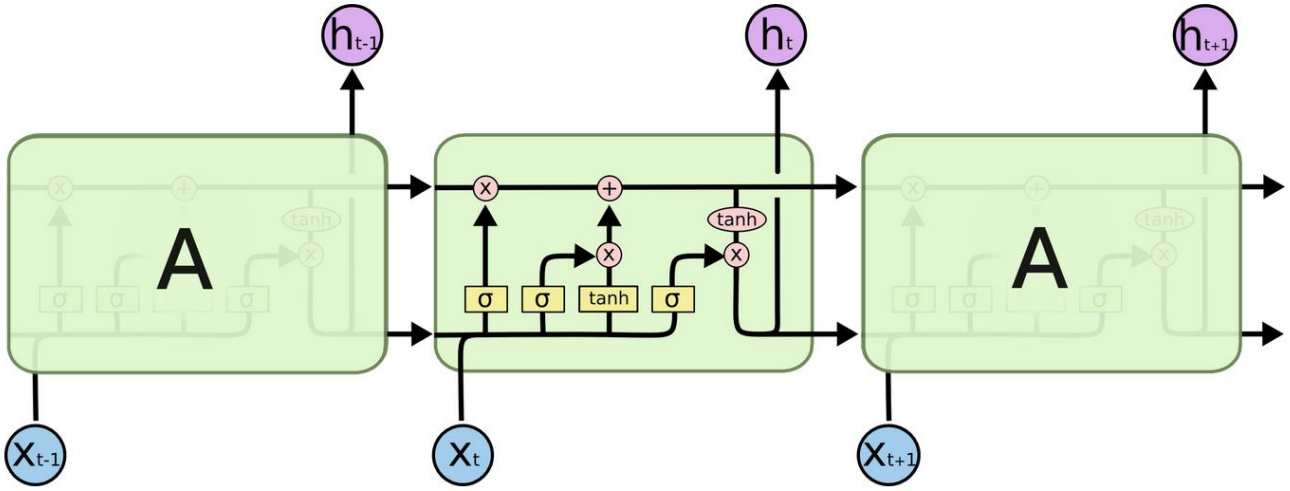
- ✓ LSTMs are explicitly designed to avoid the long-term dependency problem.
- ✓ It uses linear memory cells surrounded by multiplicative gate units to store read, write and reset information

Standard RNN



Gating functions to select information for passing and remembering

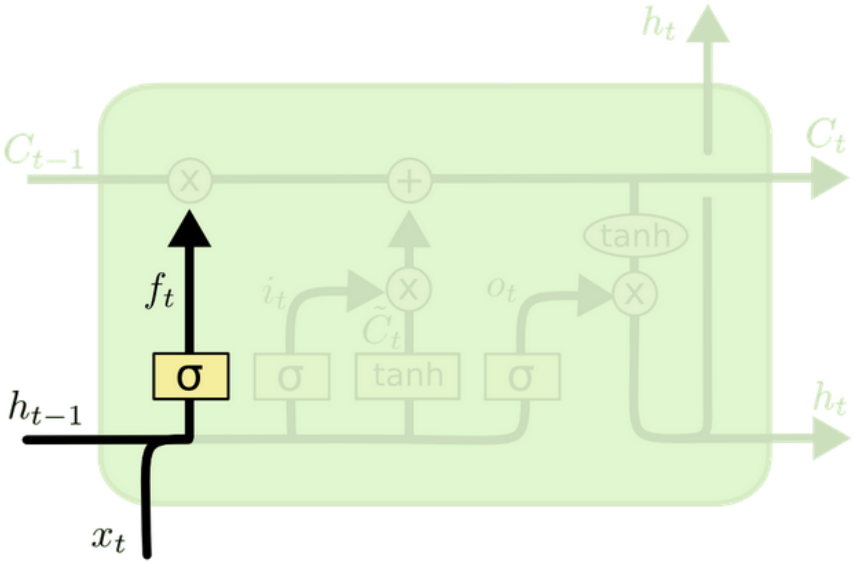
LSTM



Understanding LSTM Networks, colah's blog

➢ Forget gate layer

to decide what information we're going to throw away from the memory cells

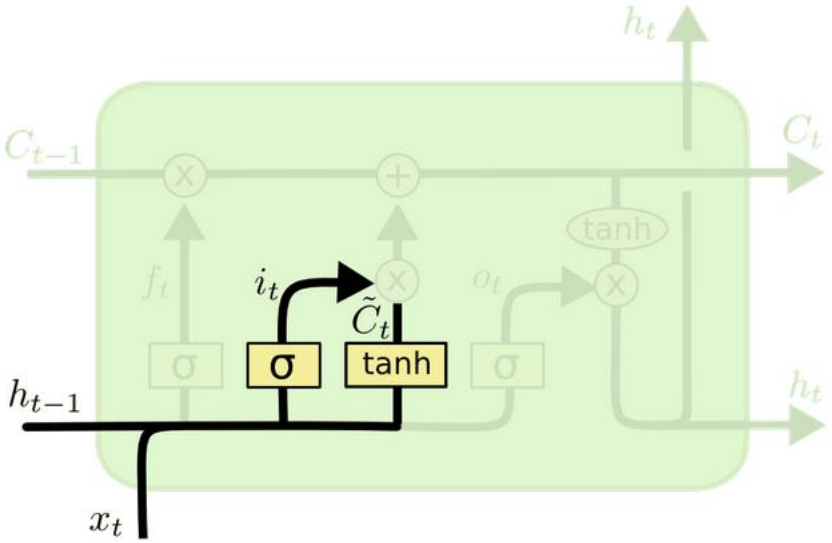


$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

Understanding LSTM Networks, colah's blog

➢input gate layer and a tanh layer

to decide what new information we're going to store in the memory cells



$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$

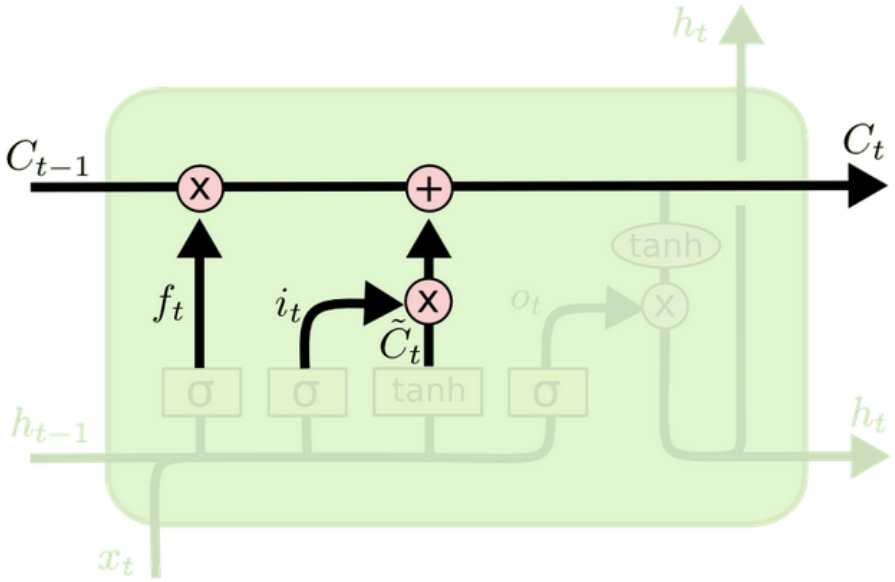
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$

Input gate layer to decide which values we'll update

a tanh layer creates a vector of new candidate memory states

Understanding LSTM Networks, colah's blog

➢Update memory cells



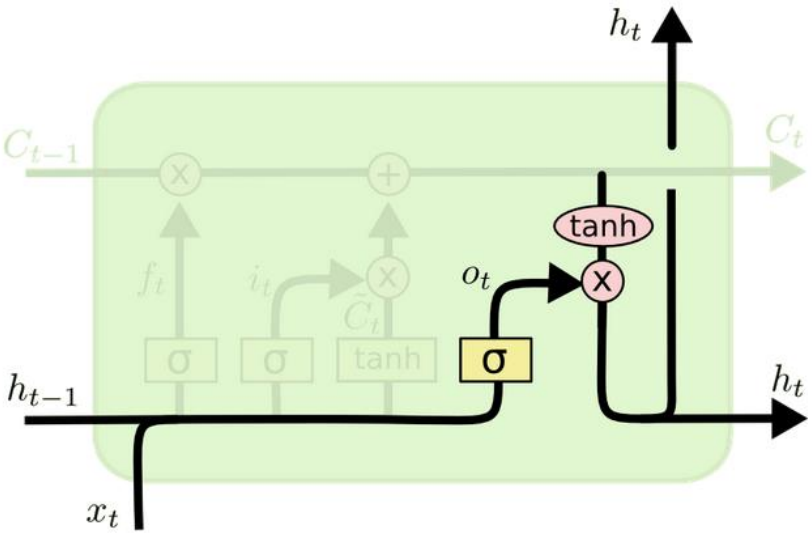
$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

forgetting the things we decided to forget earlier

Scaling the new candidate values by how much we decided to update each state value.

Understanding LSTM Networks, colah's blog

➢Update hidden cells



$$o_t = \sigma\left(W_o\left[h_{t-1}, x_t\right] + b_o\right)$$

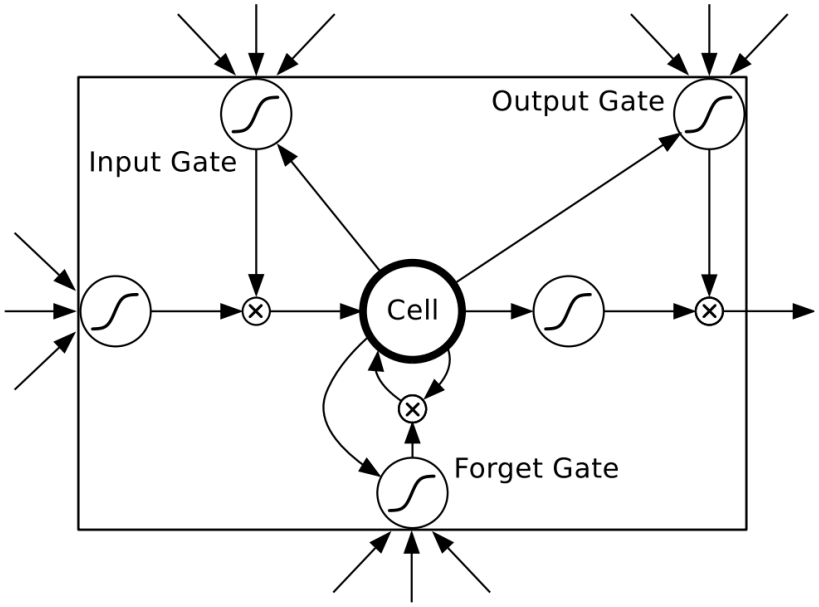
$$h_t = o_t * \tanh\left(C_t\right)$$

a sigmoid layer which decides what parts of the cell state we're going to output.

only output the parts we decided to.

Understanding LSTM Networks, colah's blog

Long Short-Term Memory (LSTM) Horchreiter & Schmidhuber (1997)



Input gate: scales input to cell (write)

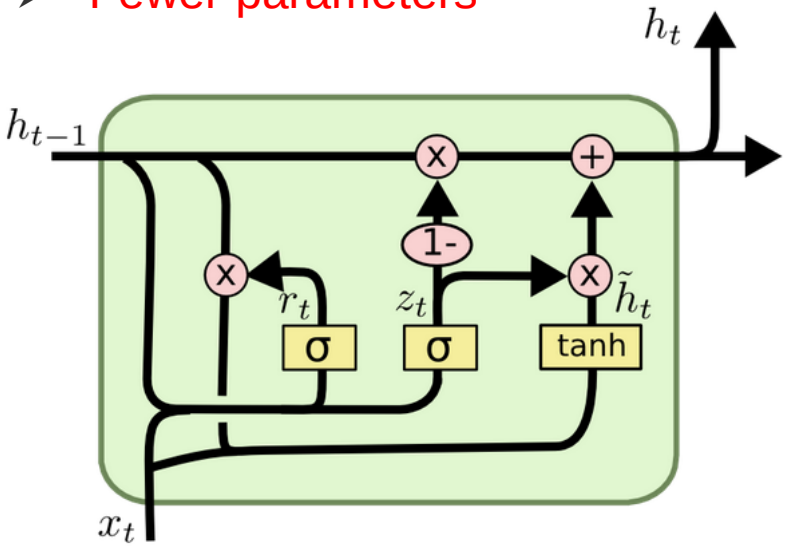
Output gate: scales output from cell (read)

Forget gate: scales old cell value (reset)

[Alex Graves]

Gated Recurrent Unit (GRU), a popular variant

- Combines the forget and input gates into a single update gate.
- Merges the cell state and hidden states ….
- Fewer parameters



$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$

$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$

$$\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

Greff, et al. (2015) experimented on popular function variants of LSTM, finding that they're all about the same.

Understanding LSTM Networks, colah's blog

Outline

- Context Modelling with CNN
- LSTM and its Variants
- LSTM Architecture Variants
- Application in Semantic Segmentation

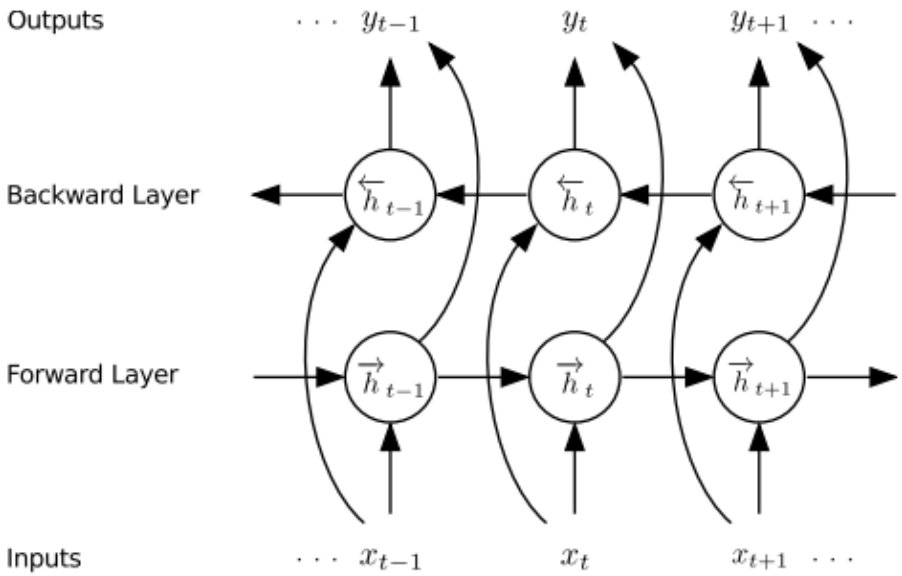
LSTM architecture variants: in temporal domain

Bidirectional LSTM

To capture both past and future information.

The hidden state of the Bi-directional LSTM is the concatenation of the forward and backward hidden states.

$$\overrightarrow{h}_t = \mathcal{H}\left(W_{x\overrightarrow{h}}x_t + W_{\overrightarrow{h}\overrightarrow{h}}\overrightarrow{h}_{t-1} + b_{\overrightarrow{h}}\right)$$
$$\overleftarrow{h}_t = \mathcal{H}\left(W_{x\overleftarrow{h}}x_t + W_{\overleftarrow{h}\overleftarrow{h}}\overleftarrow{h}_{t+1} + b_{\overleftarrow{h}}\right)$$
$$y_t = W_{\overrightarrow{h}y}\overrightarrow{h}_t + W_{\overleftarrow{h}y}\overleftarrow{h}_t + b_o$$



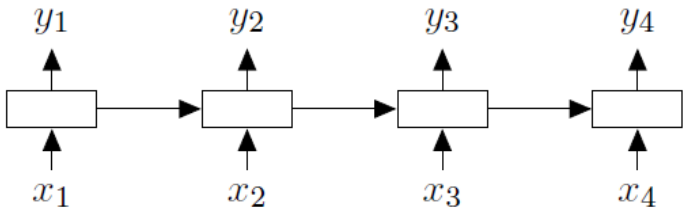
[Speech Recognition with Deep Recurrent Neural Networks, Alex Graves]

LSTM architecture variants: domain-specific structure

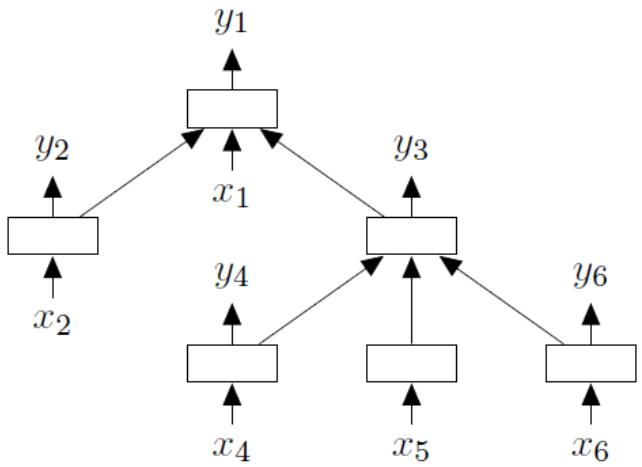
Tree-structured LSTM

Tree-LSTM, composes its state from an input vector and the hidden states of arbitrary many child units. The standard LSTM can then be considered a special case of the Tree-LSTM where each internal node has exactly one child.

Chain-structured LSTM



Tree-structured LSTM



[Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks, Stanford University]

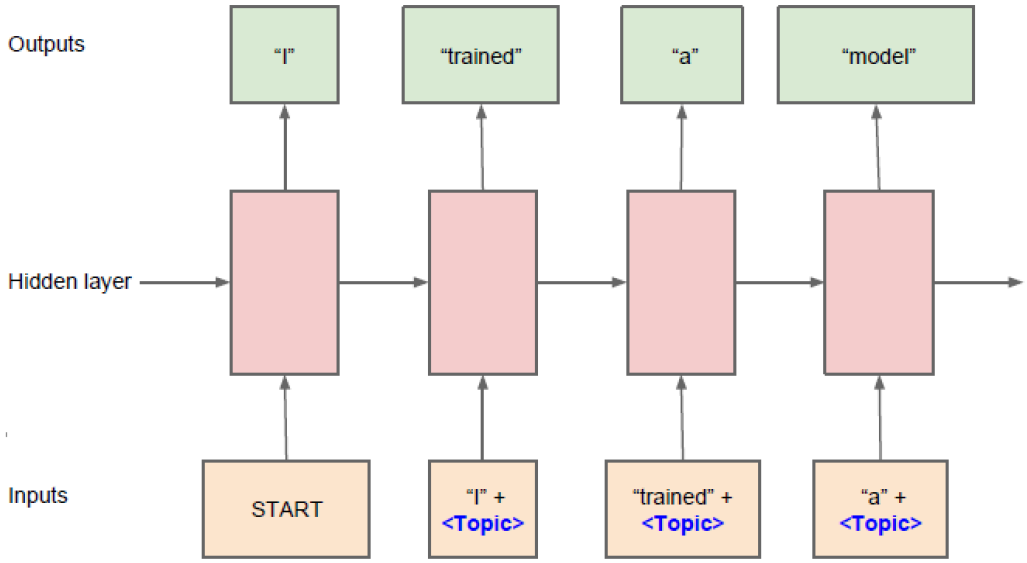
LSTM architecture variants: input feature augmentation

Contextual LSTM

Incorporate contextual features (e.g., topics) into the LSTM.

Topic

$$
\begin{aligned}
i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i + \mathbf{W_{Ti}T}) \\
f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f + \mathbf{W_{Ti}T}) \\
c_t &= f_t c_{t-1} + i_t \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c + \mathbf{W_{Ti}T}) \\
o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o + \mathbf{W_{Ti}T}) \\
h_t &= o_t \tanh(c_t)
\end{aligned}
$$

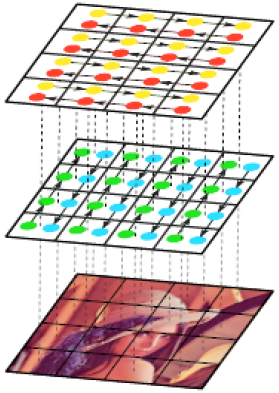


[Contextual LSTM (CLSTM) models for Large scale NLP tasks, Google]

LSTM architecture variants: 2-D image processing

Renet

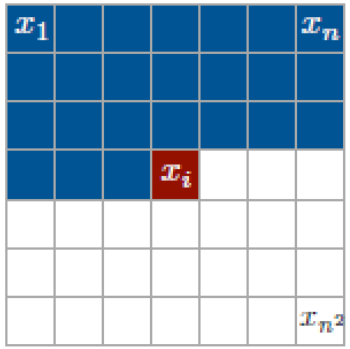
Alternative to CNN



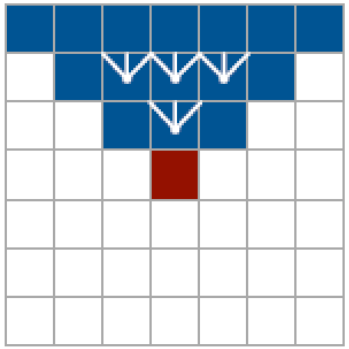
[ReNet: A Recurrent Neural Network Based Alternative to Convolutional Networks]

Pixel RNN

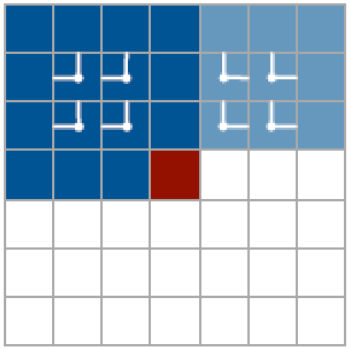
Multi-dimensional LSTM



Row LSTM



Diagonal BiLSTM



[Pixel Recurrent Neural Networks, Google DeepMind]

Grid LSTM: a unified way for both deep and sequential computation

- N sides with **incoming** hidden and memory vectors
- N sides with **outgoing** hidden and memory vectors

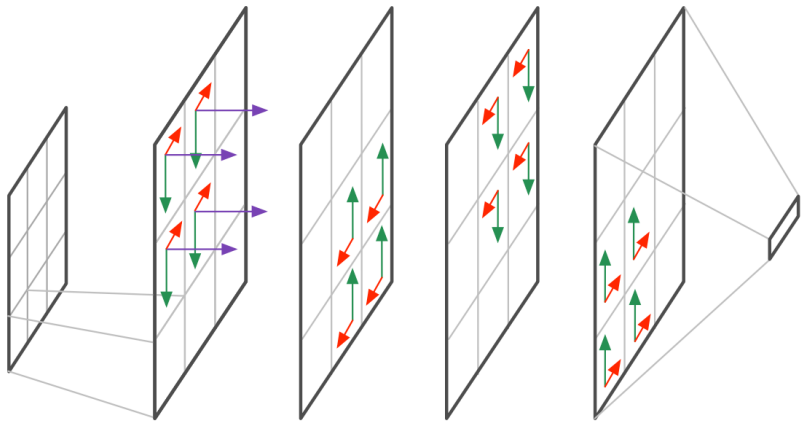
$$\mathbf{H} = \begin{bmatrix} \mathbf{h}_1 \\ \vdots \\ \mathbf{h}_N \end{bmatrix}$$

$$(\mathbf{h}'_1, \mathbf{m}'_1) = \mathsf{LSTM}(\mathbf{H}, \mathbf{m}_1, \mathbf{W}_1)$$

$$\vdots$$

$$(\mathbf{h}'_N, \mathbf{m}'_N) = \mathsf{LSTM}(\mathbf{H}, \mathbf{m}_N, \mathbf{W}_N)$$

Example: 3D-LSTM network



[Grid Long Short-Term Memory, Google DeepMind]

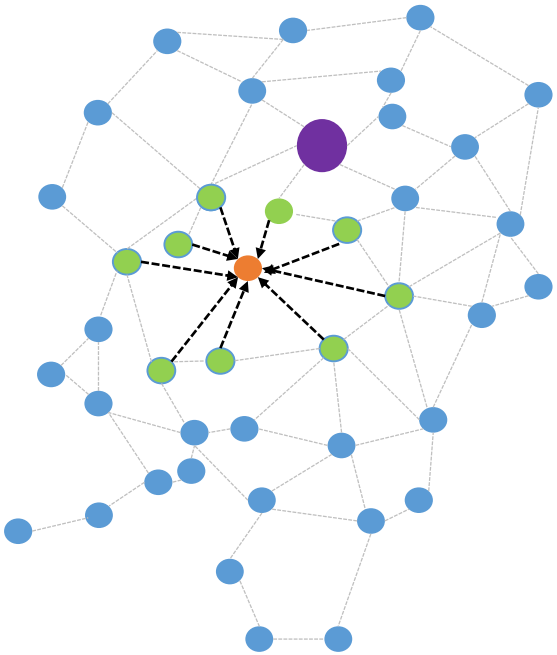
LSTM architecture variants: Graph LSTM

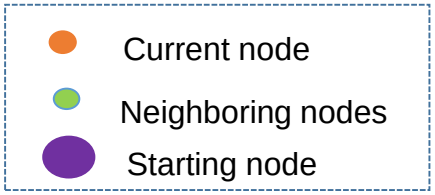
SUN YAT-SEN UNIVERSITY

- Generalize the LSTM for sequential data or multi-dimensional data to general graph-structured data

✓ adaptive graph topology with different numbers of neighbors

✓ adaptive starting node





Averaged Hidden States for Neighboring Nodes

$$\bar{\mathbf{h}}_{i,t} = \frac{\sum_{j \in \mathcal{N}_{\mathcal{G}}(i)} (\mathbb{1}(q_j = 1)\mathbf{h}_{j,t+1} + \mathbb{1}(q_j = 0)\mathbf{h}_{j,t})}{|\mathcal{N}_{\mathcal{G}}(i)|}.$$

Graph LSTM unit

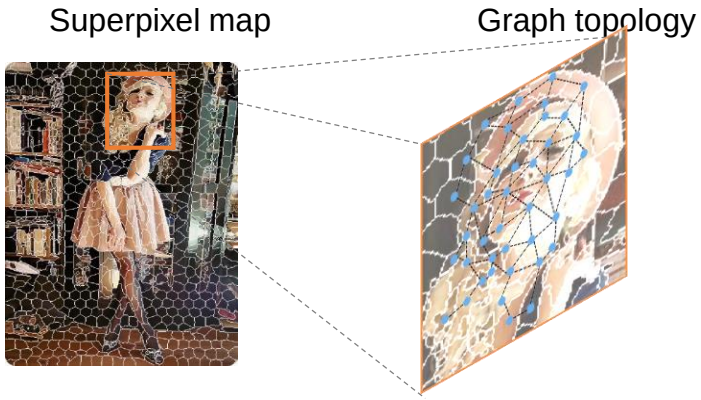
Adaptive forget gate

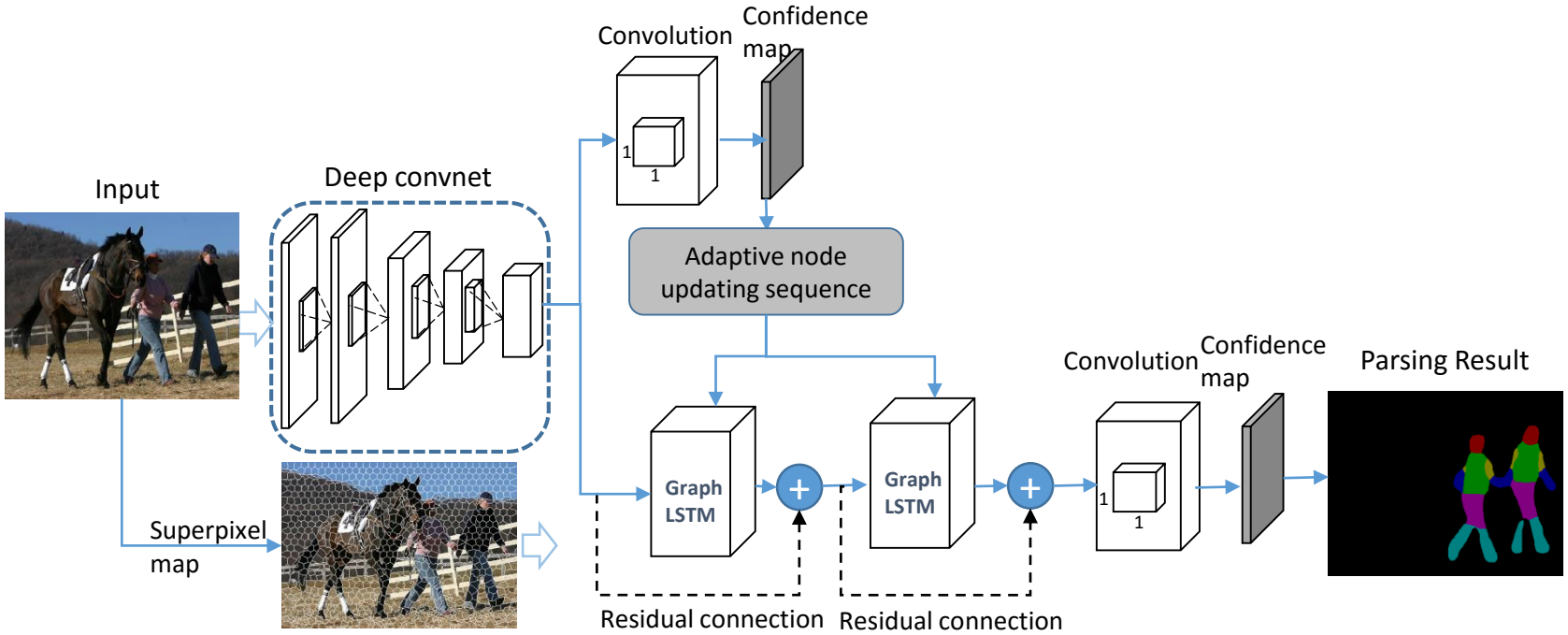
$$\begin{aligned}
g_i^u &= \delta(W^u \mathbf{f}_{i,t+1} + U^u \mathbf{h}_{i,t} + U^{un} \bar{\mathbf{h}}_{i,t} + b^u), \\
\bar{g}_{ij}^f &= \delta(W^f \mathbf{f}_{i,t+1} + U^{fn} \mathbf{h}_{j,t} + b^f), \\
g_i^f &= \delta(W^f \mathbf{f}_{i,t+1} + U^f \mathbf{h}_{i,t} + b^f), \\
g_i^o &= \delta(W^o \mathbf{f}_{i,t+1} + U^o \mathbf{h}_{i,t} + U^{on} \bar{\mathbf{h}}_{i,t} + b^o), \\
g_i^c &= \tanh(W^c \mathbf{f}_{i,t+1} + U^c \mathbf{h}_{i,t} + U^{cn} \bar{\mathbf{h}}_{i,t} + b^c), \\
\mathbf{m}_{i,t+1} &= \frac{\sum_{j \in \mathcal{N}_{\mathcal{G}}(i)} (\mathbb{1}(q_j = 1)\bar{g}_{ij}^f \odot \mathbf{m}_{j,t+1} + \mathbb{1}(q_j = 0)\bar{g}_{ij}^f \odot \mathbf{m}_{j,t})}{|\mathcal{N}_{\mathcal{G}}(i)|} \\
&\quad + g_i^f \odot \mathbf{m}_{i,t} + g_i^u \odot g_i^c, \\
\mathbf{h}_{i,t+1} &= \tanh(g_i^o \odot \mathbf{m}_{i,t+1}).
\end{aligned}$$

[Xiaodan Liang et al. Semantic Object Parsing with Graph LSTM]

LSTM architecture variants: Graph LSTM

Example:



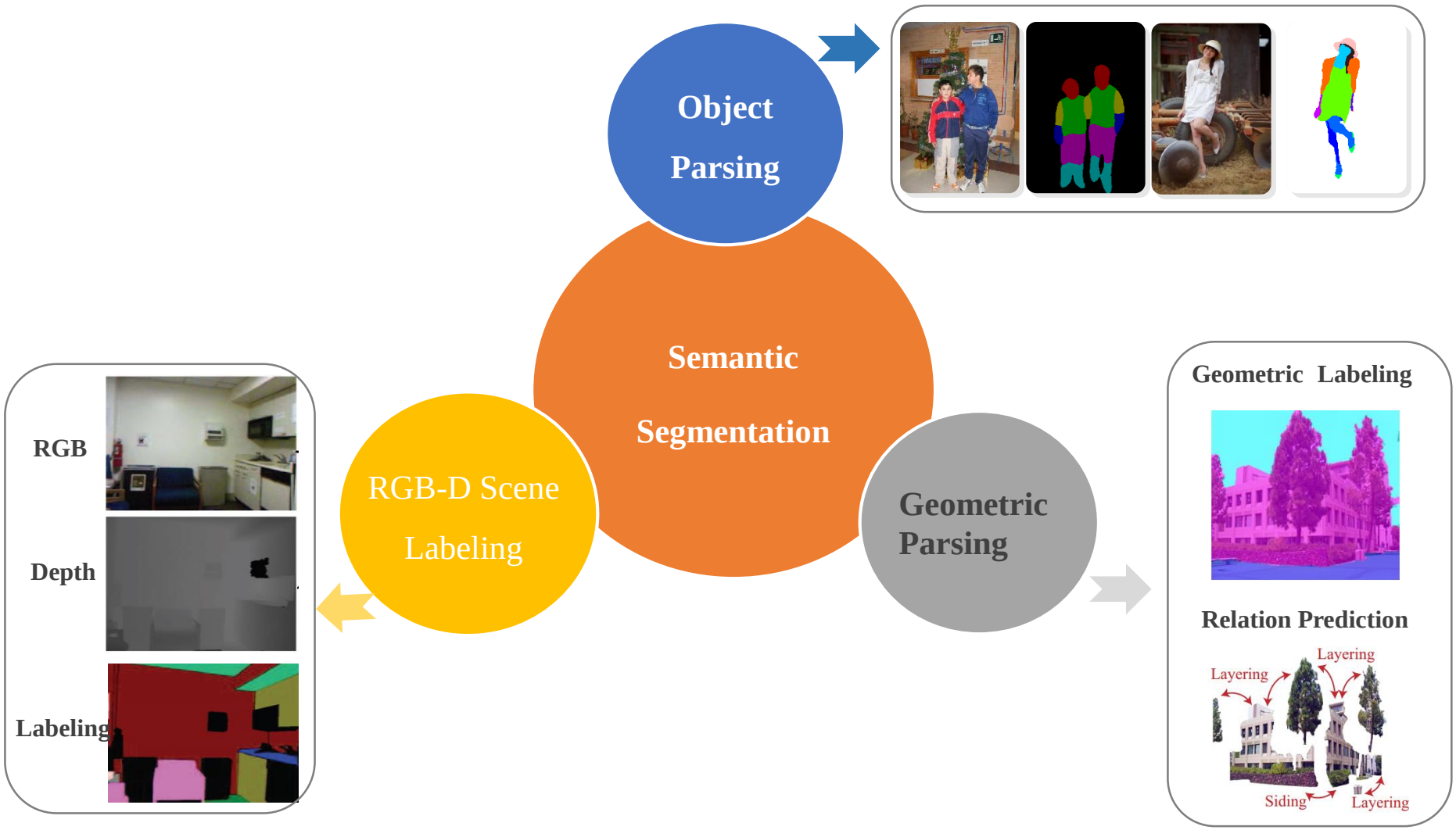


[Liang et al. Semantic Object Parsing with Graph LSTM.]

Outline

- Context Modelling with CNN
- LSTM and its Variants
- LSTM Architecture Variants
- Application in Semantic Segmentation

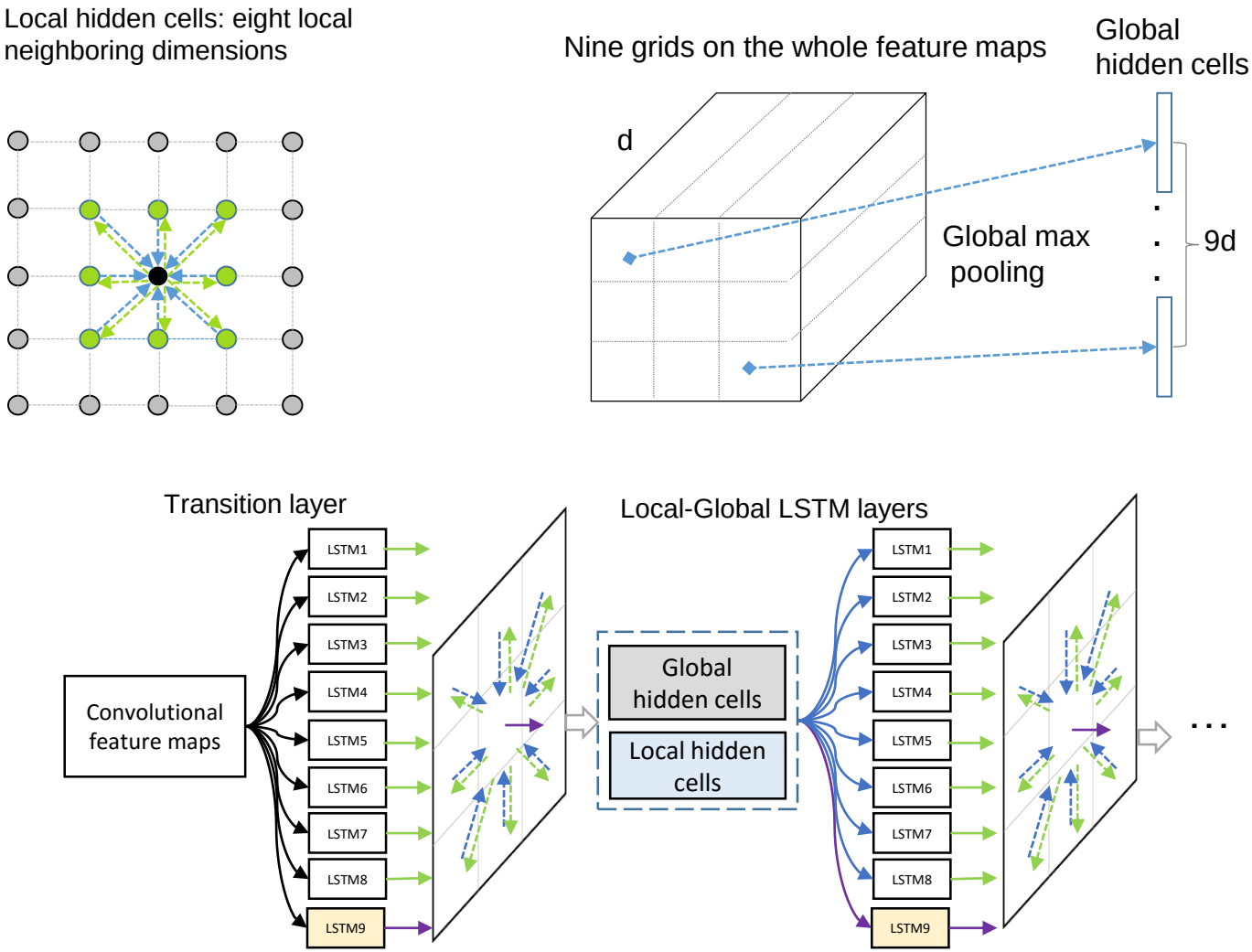
Application in Semantic Segmentation



Jointly work with Prof. Liang Lin, Prof. Shuicheng Yan and other lab members

Application in Semantic Segmentation: Object Parsing

Local-Global LSTM layers: jointly model local and global contexts

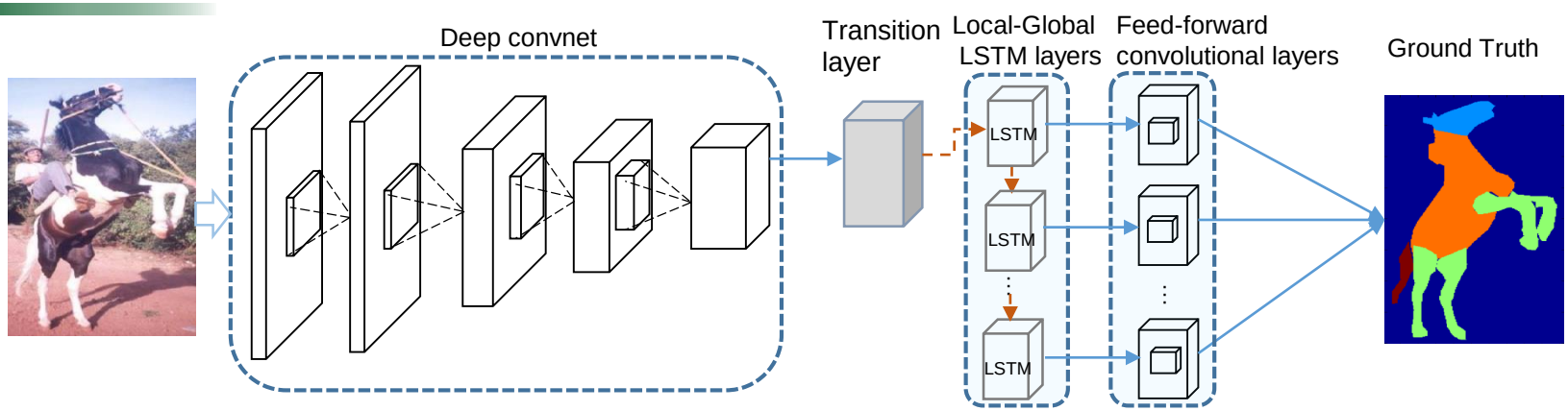


[Xiaodan Liang et al. Semantic Object Parsing with Local-Global Long Short-Term Memory]

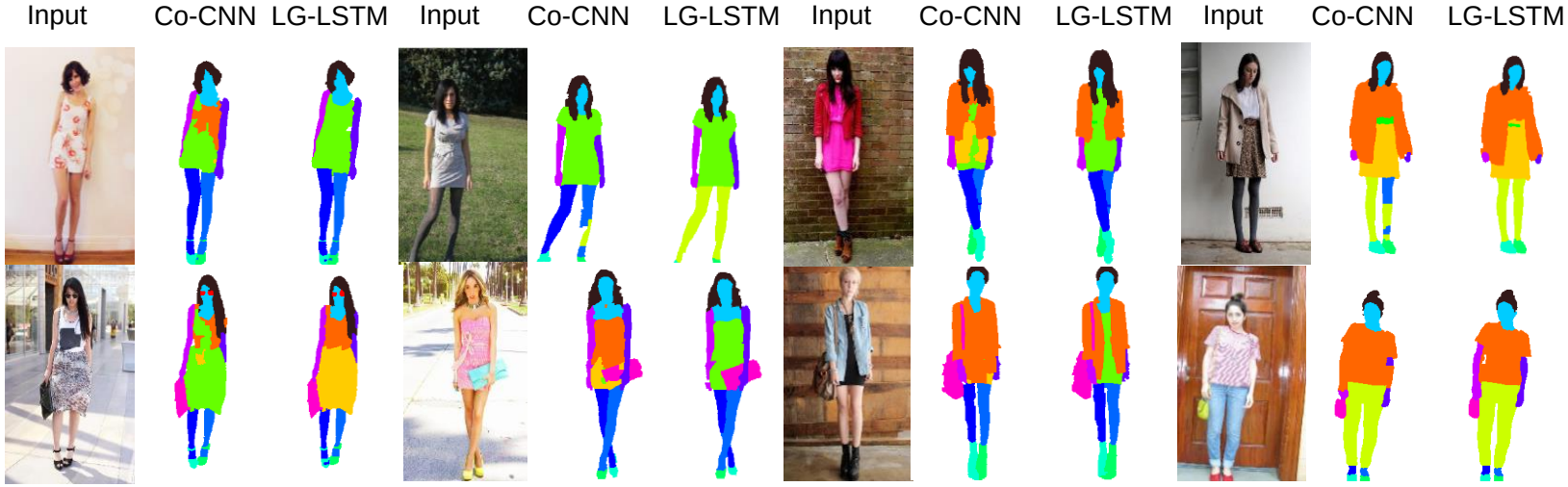
Application in Semantic Segmentation: Object Parsing

中山大學 SUN YAT-SEN UNIVERSITY

LG-LSTM architecture for object parsing:

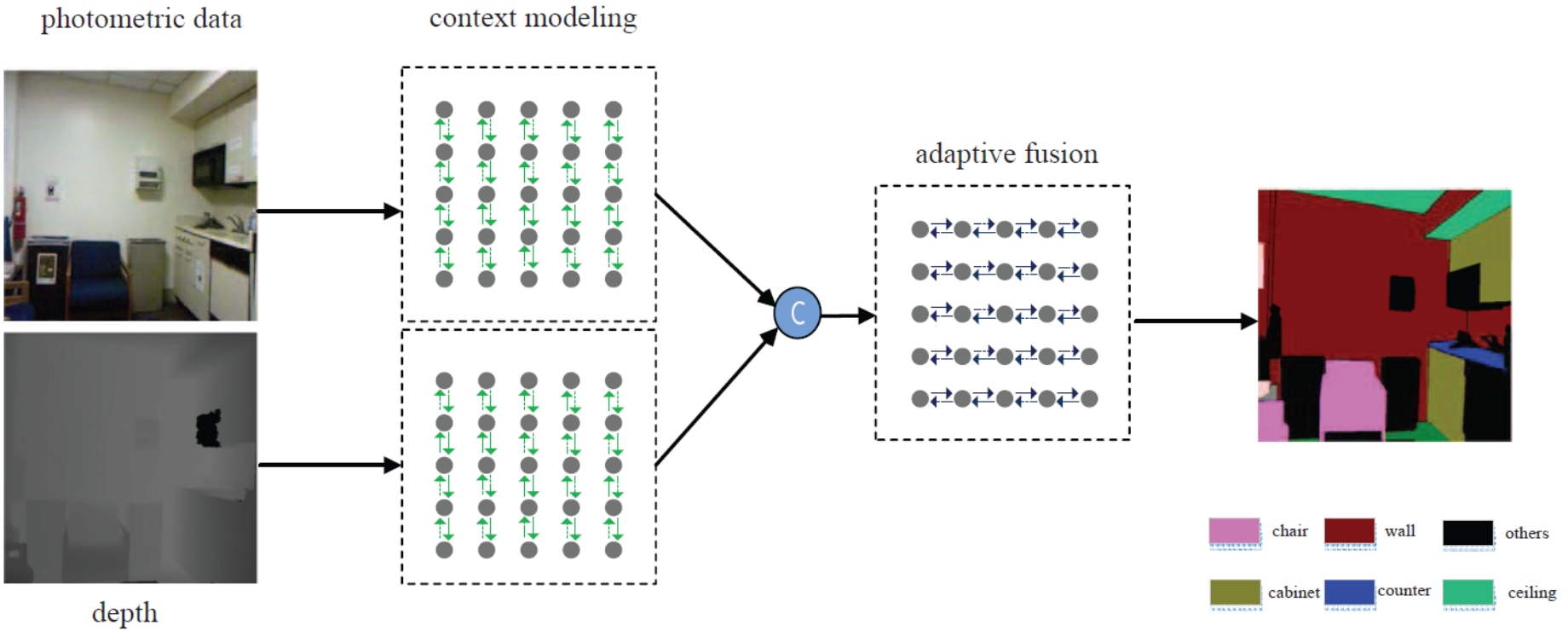


Results:



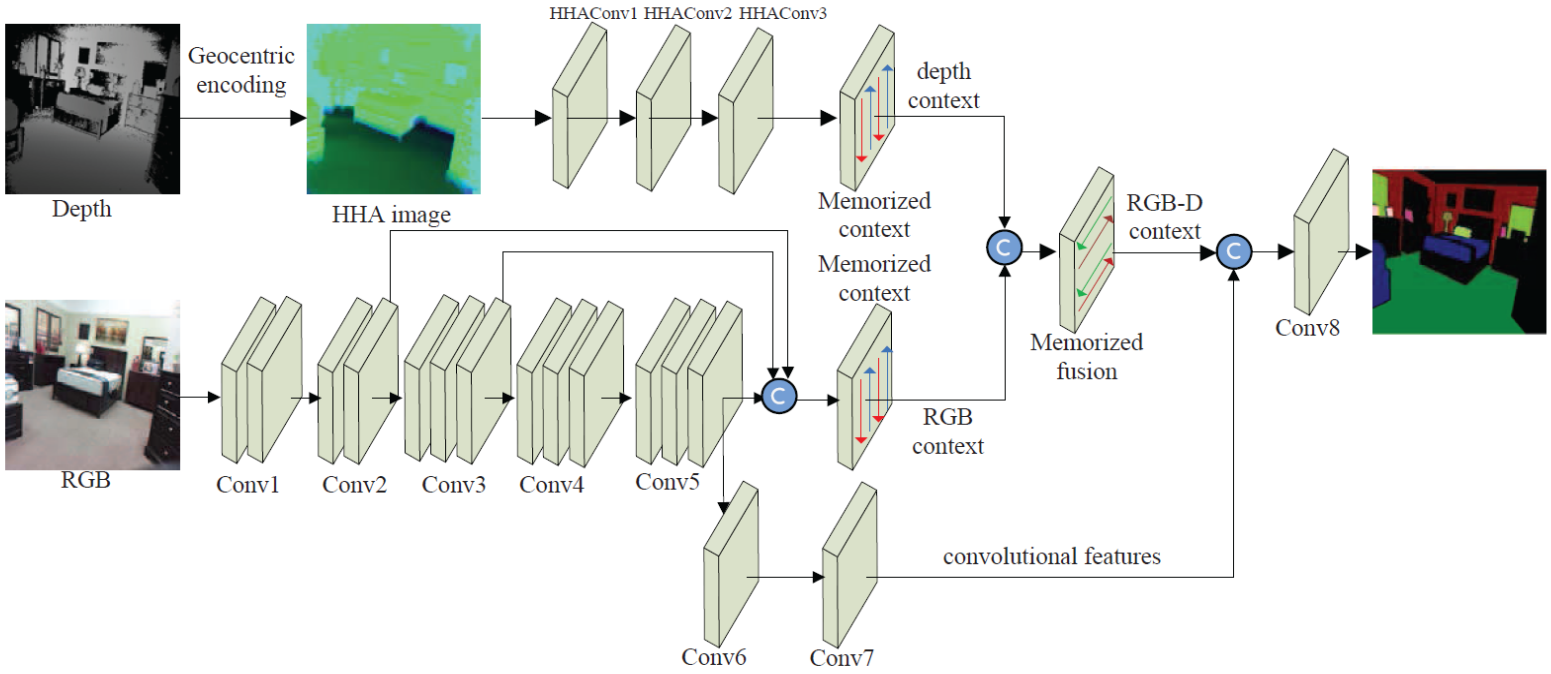
[Xiaodan Liang et al. Semantic Object Parsing with Local-Global Long Short-Term Memory]

LSTM Fusion for scene Labeling



[Zhen Li et al. RGB-D Scene Labeling with Long Short-Term Memorized Fusion Model

LSTM Fusion for scene Labeling



- ✓ Two LSTM layers in parallel for vertical context modelling
- ✓ LSTM fusion layers for context fusion in RGB and depth image

[Zhen Li et al. RGB-D Scene Labeling with Long Short-Term Memorized Fusion Model]

Application in Semantic Segmentation: RGB-D scene Labeling

中山大學 SUN YAT-SEN UNIVERSITY

➢ Results on Sun-RGBD dataset

Average and individual accuracy of 37 classes：

	Wall	floor	cabinet	bed	chair	sofa	table	door	window	bookshelf	picture	counter	blinds	desk	shelves	curtain	dresser	pillow	mirror
[28]	37.8	45.0	17.4	21.8	16.9	12.8	18.5	6.1	9.6	9.4	4.6	2.2	2.4	7.3	1.0	4.3	2.2	2.3	6.9
[28]	32.1	42.6	2.9	6.4	21.5	4.1	12.5	3.4	5.0	0.8	3.3	1.7	14.8	2.0	15.3	2.0	1.4	1.2	0.9
[28]	36.4	45.8	15.4	23.3	19.9	11.6	19.3	6.0	7.9	12.8	3.6	5.2	2.2	7.0	1.7	4.4	5.4	3.1	5.6
[36]	38.9	47.2	18.8	21.5	17.2	13.4	20.4	6.8	11.0	9.6	6.1	2.6	3.6	7.3	1.2	6.9	2.4	2.6	6.2
[36]	33.3	43.8	3.0	6.3	22.3	3.9	12.9	3.8	5.6	0.9	3.8	2.2	32.6	2.0	10.1	3.6	1.8	1.1	1.0
[36]	37.8	48.3	17.2	23.6	20.8	12.1	20.9	6.8	9.0	13.1	4.4	6.2	2.4	6.8	1.0	7.8	4.8	3.2	6.4
[22]	43.2	78.6	26.2	42.5	33.2	40.6	34.3	33.2	43.6	23.1	**57.2**	31.8	42.3	12.1	**18.4**	59.1	31.4	**49.5**	24.8
Ours	**74.9**	**82.3**	**47.3**	**62.1**	**67.7**	**55.5**	**57.8**	**45.6**	**52.8**	**43.1**	56.7	**39.4**	**48.6**	**37.3**	9.6	**63.4**	**35.0**	45.8	**44.5**
	floormat	clothes	ceiling	books	fridge	tv	paper	towel	shower	box	board	person	nightstand	toilet	sink	lamp	bathhub	bag	mean
[28]	0.0	1.2	27.9	4.1	7.0	1.6	1.5	1.9	0.0	0.6	7.4	0.0	1.1	8.9	14.0	0.9	0.6	0.9	8.3
[28]	0.0	0.3	9.7	0.6	0.0	0.9	0.0	0.1	0.0	1.0	2.7	0.3	2.6	2.3	1.1	0.7	0.0	0.4	5.3
[28]	0.0	1.4	35.8	6.1	9.5	0.7	1.4	0.2	0.0	0.6	7.6	0.7	1.7	12.0	15.2	0.9	1.1	0.6	9.0
[36]	0.0	1.3	39.1	5.9	7.1	1.4	1.5	2.2	0.0	0.7	10.4	0.0	1.5	12.3	14.8	1.3	0.9	1.1	9.3
[36]	0.0	0.6	13.9	0.5	0.0	0.9	0.4	0.3	0.0	0.7	3.5	0.3	1.5	2.6	1.2	0.8	0.0	0.5	6.0
[36]	0.0	1.6	49.2	8.7	10.1	0.6	1.4	0.2	0.0	0.8	8.6	0.8	1.8	14.9	16.8	1.2	1.1	1.3	10.1
[22]	**5.6**	27.0	**84.5**	35.7	24.2	36.5	26.8	19.2	**9.0**	11.7	**51.4**	35.7	25.0	64.1	53.0	44.2	47.0	18.6	36.3
Ours	0.0	**28.4**	68.0	**47.9**	**61.5**	**52.1**	**36.4**	**36.7**	0	**38.1**	48.1	**72.6**	**36.4**	**68.8**	**67.9**	**58.0**	**65.6**	**23.6**	**48.1**

Over **11.8%** improvement on average accuracy

State-of-the-art on **30 categories**

[28] Song et al.: A RGB-D scene understanding benchmark suite. CVPR, 2015.

[36] Liu et al.: Sift fow: Dense correspondence across scenes and its applications. TPAMI, 2011

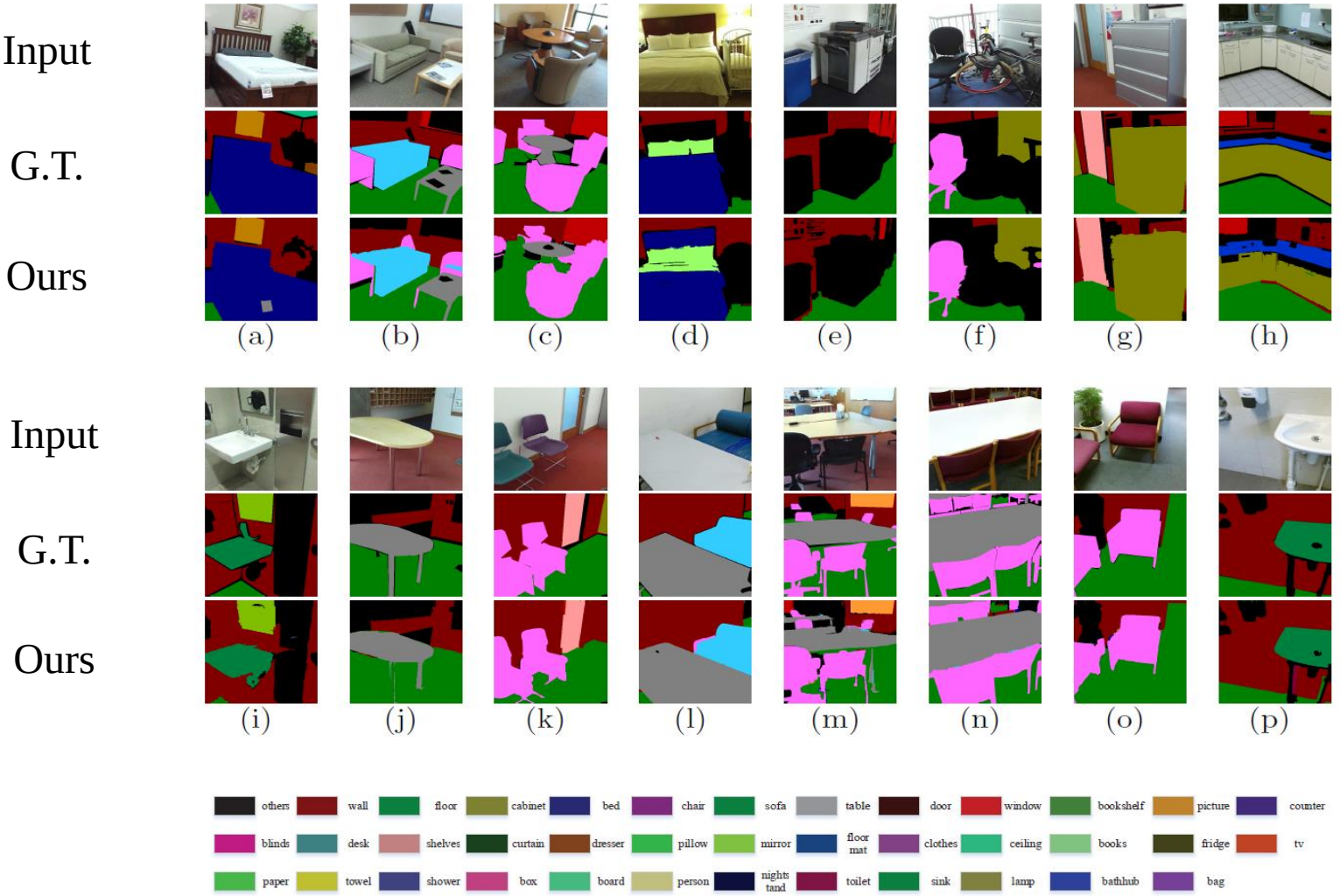
[22] Ren et al.: RGB-D scene labeling: Features and algorithms. CVPR，2012

[Zhen Li et al. RGB-D Scene Labeling with Long Short-Term Memorized Fusion Model]

Application in Semantic Segmentation: RGB-D scene Labeling

SUN YAT-SEN UNIVERSITY

- Results on **Sun-RGBD dataset**



[Zhen Li et al. RGB-D Scene Labeling with Long Short-Term Memorized Fusion Model]

Application in Semantic Segmentation: RGB-D scene Labeling

中山大學 SUN YAT-SEN UNIVERSITY

➢ Results on **NYU-Depth v2 Dataset**

Average and individual accuracy of 37 classes：

	Wall	floor	cabinet	bed	chair	sofa	table	door	window	bookshelf	picture	counter	blinds	desk	shelves	curtain	dresser	pillow	mirror
Freq	21.4	9.1	6.2	3.8	3.3	2.7	2.1	2.2	2.1	1.9	2.1	1.4	1.7	1.1	1.0	1.1	0.9	0.8	1.0
[25]	60.7	77.8	33.0	40.3	32.4	25.3	21.0	5.9	29.7	22.7	35.7	33.1	40.6	4.7	3.3	27.4	13.3	18.9	4.4
[22]	60.0	74.4	37.1	42.3	32.5	28.2	16.6	12.9	27.7	17.3	32.4	38.6	26.5	10.1	6.1	27.6	7.0	19.7	17.9
[23]	67.4	80.5	41.4	56.4	40.4	44.8	30.0	12.1	34.1	20.5	38.7	50.7	44.7	10.1	1.6	26.3	21.6	31.3	14.6
[24]	61.4	66.4	38.2	43.9	34.4	33.8	22.6	8.3	27.6	17.6	27.7	30.2	33.6	5.1	2.7	18.9	16.8	12.5	10.7
[14]	65.7	62.5	40.1	32.1	44.5	50.8	43.5	**51.6**	**49.2**	36.3	41.4	39.2	55.8	**48.0**	**45.2**	53.1	55.3	**50.5**	46.1
Ours	**79.6**	**83.5**	**69.3**	**77.0**	**58.3**	**64.9**	**42.6**	47.0	43.6	**59.5**	**74.5**	**68.2**	**74.6**	33.6	13.1	**53.2**	**56.5**	48.0	**47.7**
	floormat	clothes	ceiling	books	fridge	tv	paper	towel	shower	box	board	person	nightstand	toilet	sink	lamp	bathhub	bag	mean
Freq	0.7	0.7	1.4	0.6	0.6	0.5	0.4	0.4	0.4	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.2	
[25]	7.1	6.5	73.2	5.5	1.4	5.7	12.7	0.1	3.6	0.1	0.0	6.6	6.3	26.7	25.1	15.9	0.0	0.0	17.5
[22]	20.1	9.5	53.9	14.8	1.9	18.6	11.7	12.6	5.4	3.3	0.2	13.6	9.2	35.2	28.9	14.2	7.8	1.2	20.2
[23]	28.2	8.0	61.8	5.8	14.5	14.4	14.1	19.8	6.0	1.1	12.9	1.5	15.7	52.5	47.9	31.2	29.4	0.2	30.0
[24]	13.8	2.7	46.1	3.6	2.9	3.2	2.6	6.2	6.1	0.8	28.2	5	6.9	32	20.9	5.4	16.2	0.2	29.2
[14]	**54.1**	**35.4**	50.6	39.1	**53.6**	**50.1**	35.4	39.9	**41.8**	**36.3**	60.6	35.6	**32.5**	31.8	22.5	26.3	38.5	**37.3**	43.9
Ours	0.0	22.7	**70.2**	**49.7**	0.0	0.0	**52.1**	**60.6**	0	17.6	**93.9**	**77.0**	0	**81.8**	**58.4**	**67.6**	**72.6**	7.5	**49.4**

Over **5.5%** improvement on average accuracy

State-of-the-art on **24 categories**

[25] Silberman et al.: Indoor segmentation and support inference from rgbd images. ECCV, 2012.

[23] Gupta, et al.: Indoor scene understanding with rgb-d images: Bottom-up segmentation, object detection and semantic segmentation. IJCV, 2015.

[24] Wang, et al.: Unsupervised joint feature learning and encoding for rgb-d scene labeling. TIP, 2015.

[14] Khan, et al.: Integrating geometrical context for semantic labeling of indoor scenes using rgbd images. IJCV, 2015.

[Zhen Li et al. RGB-D Scene Labeling with Long Short-Term Memorized Fusion Model]

Application in Semantic Segmentation: RGB-D scene Labeling

中山大學
SUN YAT-SEN UNIVERSITY

- Results on Sun-RGBD dataset

[23] Gupta, et al.: Indoor scene understanding with rgb-d images: Bottom-up segmentation, object detection and semantic segmentation. IJCV, 2015.

[Zhen Li et al. RGB-D Scene Labeling with Long Short-Term Memorized Fusion Model]

Application in Semantic Segmentation: Geometric Parsing

Input Image

Geometric Labeling

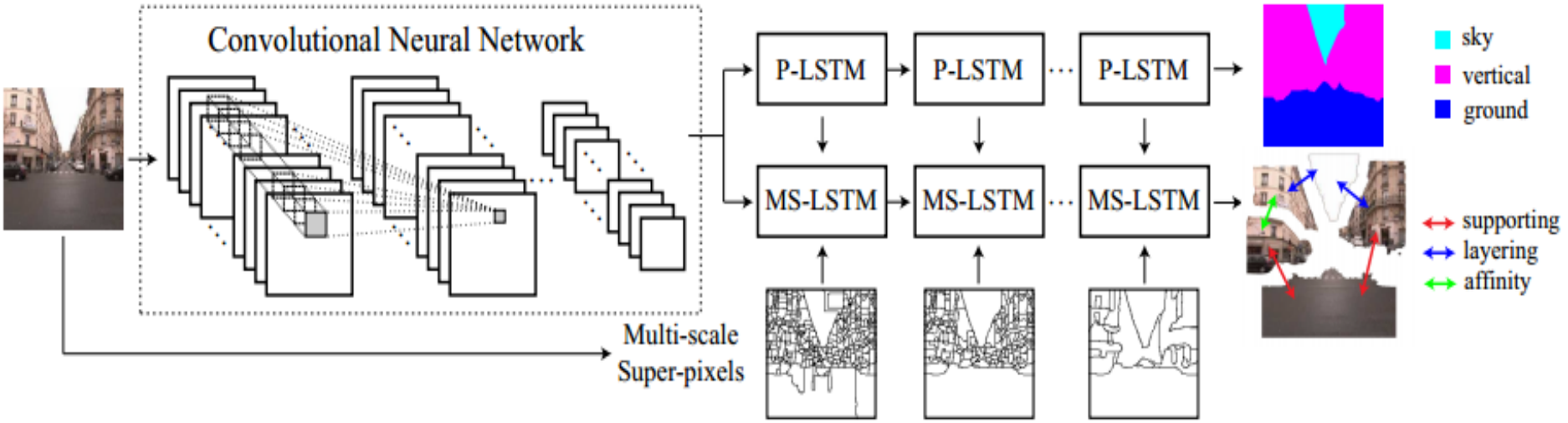
Relation Prediction

[Zhanglin Peng et al. Geometric Scene Parsing with Hierarchical LSTM]

Application in Semantic Segmentation: Geometric Parsing

SUN YAT-SEN UNIVERSITY

➢ Hierarchical LSTM archictecture



✓ Stacked Pixel LSTM (P-LSTM) layers for geometric surface labeling

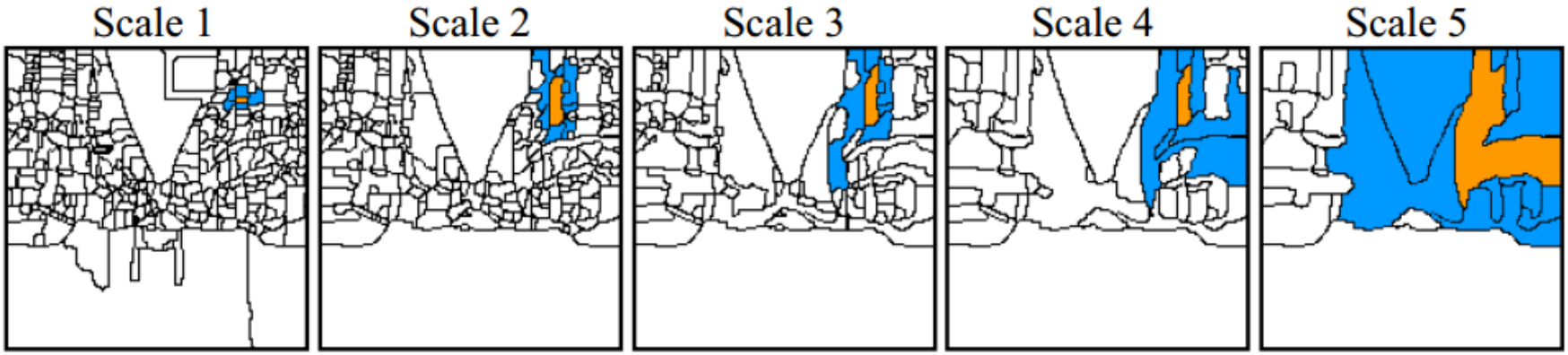
✓ Stacked Multi-scale Super-pixel LSTM(MS-LSTM) for region relation prediction

[Zhanglin Peng et al. Geometric Scene Parsing with Hierarchical LSTM]

Application in Semantic Segmentation: Geometric Parsing

SUN YAT-SEN UNIVERSITY

- Multi-scale Super-pixel LSTM



Hierarchical context modeling with multi-scale super-pixels and their neighboring super-pixels

[Zhanglin Peng et al. Geometric Scene Parsing with Hierarchical LSTM]

Application in Semantic Segmentation: RGB-D scene Labeling

SUN YAT-SEN UNIVERSITY

- Geometric surface labeling performance on SIFT-Flow dataset

Method	Sky	Ground	Vertical	Mean Acc.
Superparsing	-	-	-	89.2
FCN	96.4	93.1	91.8	93.8
DeepLab	96.1	93.8	93.4	94.4
Ours	**96.4**	**95.1**	**93.1**	**94.9**

- Geometric surface labeling performance on LM+SUN dataset

Method	Sky	Ground	Vertical	Mean Acc.
Superparsing	-	-	-	86.8
FCN	81.8	83.5	94.1	86.4
DeepLab	76.2	72.8	94.6	81.2
Ours	**83.9**	**83.6**	**94.1**	**87.2**

[Zhanglin Peng et al. Geometric Scene Parsing with Hierarchical LSTM]

Application in Semantic Segmentation: RGB-D scene Labeling

中山大學 SUN YAT-SEN UNIVERSITY

- Geometric Labeling

Model settings	SIFT-Flow	LM+SUN
Convolution	94.66	89.92
P-LSTM	94.68	90.13
P-LSTM + S-LSTM	95.24	91.06
H-LSTM (ours)	**95.41**	**91.34**

- Geometric Relation Prediction

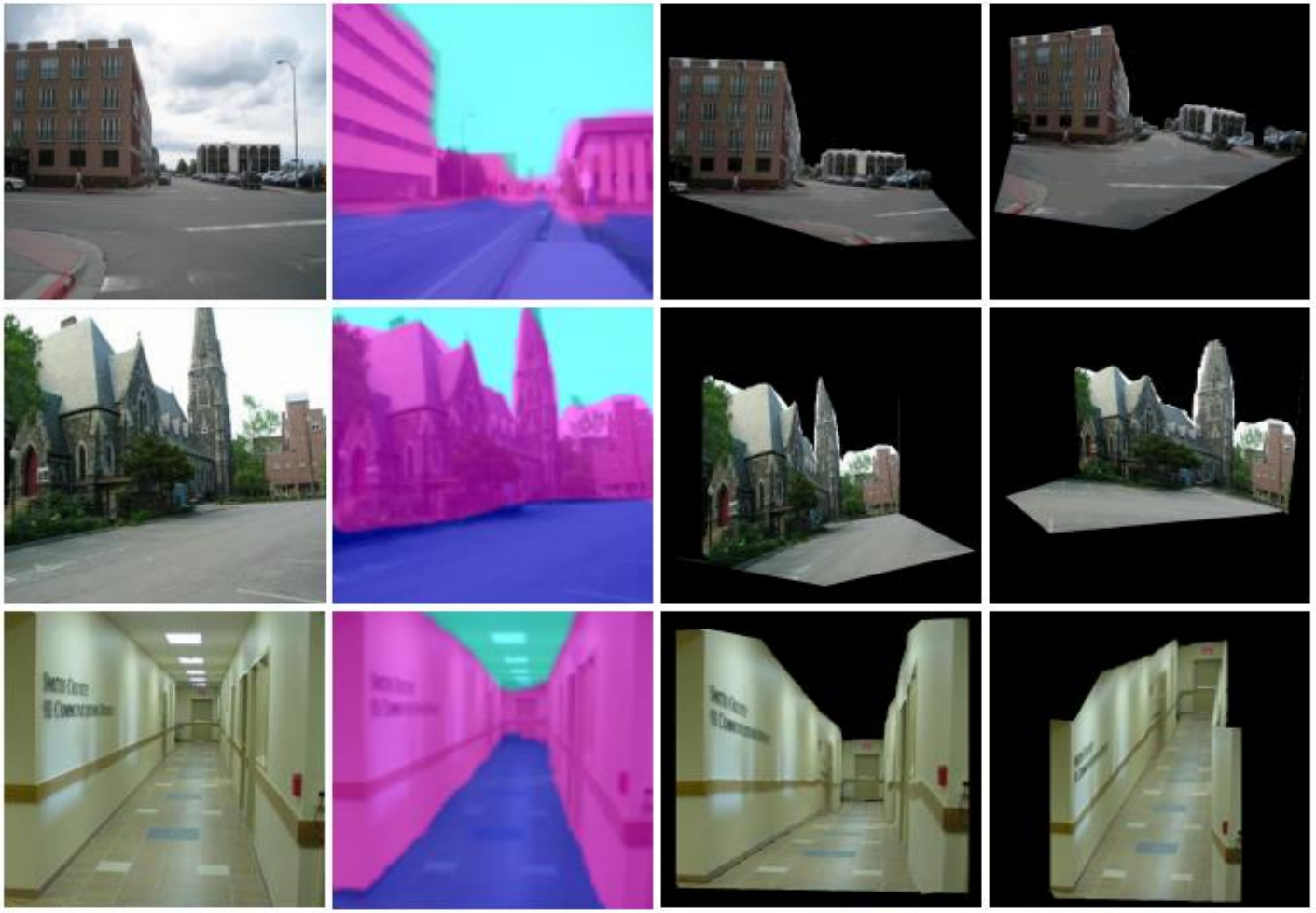
The number of MS-LSTM layers	SIFT-Flow	LM+SUN	G-Context
H-LSTM_1	85.8	89.1	87.8
H-LSTM_2	89.8	94.7	90.6
H-LSTM_3	90.3	95.6	89.8
H-LSTM_4	90.4	**96.7**	90.7
H-LSTM	**91.2**	95.8	**90.8**

[Zhanglin Peng et al. Geometric Scene Parsing with Hierarchical LSTM]

Application in Semantic Segmentation: RGB-D scene Labeling

SUN YAT-SEN UNIVERSITY

- 3D Geometric Reconstruction



Input Image Geometric Labeling 3D Reconstruction

[Zhanglin Peng et al. Geometric Scene Parsing with Hierarchical LSTM]

Codes and Platforms

SUN YAT-SEN UNIVERSITY

- Caffe platform-C++:
  - LRCN by Jeff Donahue: LSTMLayer and LSTMUnitLayer (*http://jeffdonahue.com/lrcn/*)
    -----Example: image captioning, scene labeling
  - Apollocaffe by Russell Stewart: LstmUnitLayer *(http://apollocaffe.com/)*
    ----- *dynamic network structure with variable LSTM layers and lengths of inputs*
    -----Example: person detection, object parsing and Geometric parsing
- Torch-Lua
  - DenseCap by Andrej Karpathy: (*https://github.com/jcjohnson/densecap*)
    ----- Dense Captioning task that jointly generates the object detection and descriptions
  - RNN by Nicholas Leonard : (*https://github.com/jcjohnson/densecap*)
    ----- General library for implementing RNN, LSTM, BRNN and BLSTM
- Theano-python
  - Visual attention by Shikhar Sharma: (*https://github.com/kracwarlock/action-recognition-visual-attention*)
    ----- Visual attention implementations
- Etc. Tensorflow, RNNLib, Neuraltalk2…

Questions?

http://vision.sysu.edu.cn/

http://www.lv-nus.org/