Boosting Dilated Convolutional Networks with Mixed Tensor Decompositions

Nadav Cohen Ronen Tamari Amnon Shashua

Institute for Advanced Study Hebrew University of Jerusalem

International Conference on Learning Representations (ICLR) 2018

Collaborators

Ronen Tamari

Amnon Shashua

האוניברסיטה העברית בירושלים
THE HEBREW UNIVERSITY OF JERUSALEM

Expressive Efficiency

Key to success of deep networks is their expressiveness

Expressive Efficiency

Key to success of deep networks is their expressiveness

How can this be formally analyzed?

Expressive Efficiency

Key to success of deep networks is their expressiveness

How can this be formally analyzed?

Definition (**Expressive Efficiency**)

Expressive Efficiency

Key to success of deep networks is their expressiveness

How can this be formally analyzed?

Definition (**Expressive Efficiency**)

A, B – network archs w/size params r_A, r_B

Expressive Efficiency

Key to success of deep networks is their expressiveness

How can this be formally analyzed?

Definition (**Expressive Efficiency**)

A, B – network archs w/size params r_A, r_B

We say that A is expressively efficient w.r.t. B if:

Expressive Efficiency

Key to success of deep networks is their expressiveness

How can this be formally analyzed?

<u>**Definition**</u> (**Expressive Efficiency**)

A, B – network archs w/size params r_A, r_B

We say that A is expressively efficient w.r.t. B if:

- Any func realized by B can be realized by A w/at most linear growth

$$\boxed{r_A \in \mathcal{O}(r_B)}$$

Expressive Efficiency

Key to success of deep networks is their expressiveness

How can this be formally analyzed?

<u>**Definition**</u> (**Expressive Efficiency**)

A, B – network archs w/size params r_A, r_B

We say that A is expressively efficient w.r.t. B if:

- Any func realized by B can be realized by A w/at most linear growth

$$\boxed{r_A \in \mathcal{O}(r_B)}$$

- There exist func realized by A requiring B to grow super-linearly

$$\boxed{r_B \in \Omega(f(r_A)) \text{ w/super-linear } f(\cdot)}$$

Connectivity

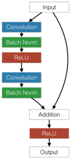
Existing expressive efficiency analyses focus on the effect of depth

Connectivity

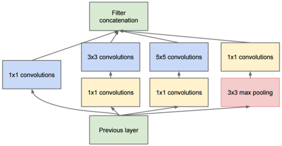
Existing expressive efficiency analyses focus on the effect of depth

A central aspect of state of the art networks has not been analyzed:

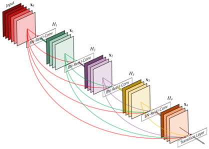
Connectivity



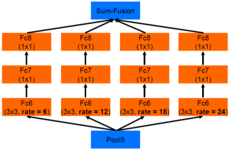
ResNet



Inception (GoogLeNet)



DenseNet



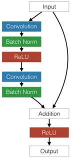
DeepLab

Connectivity

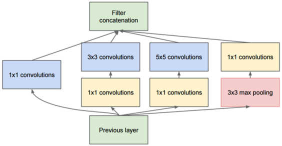
Existing expressive efficiency analyses focus on the effect of depth

A central aspect of state of the art networks has not been analyzed:

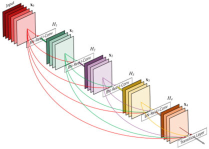
Connectivity



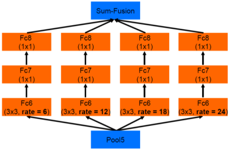
ResNet



Inception (GoogLeNet)



DenseNet



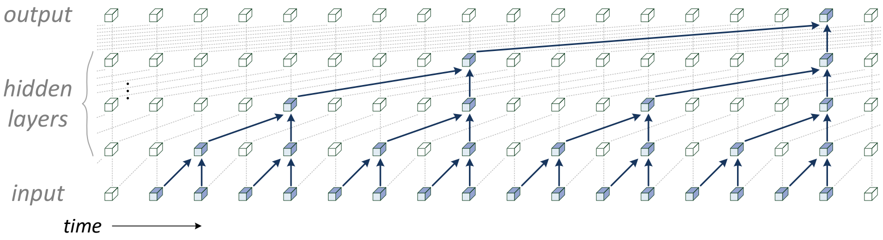
DeepLab

<u>Question of interest</u>

Can modern connectivity schemes lead to expressive efficiency?

Dilated Convolutional Networks

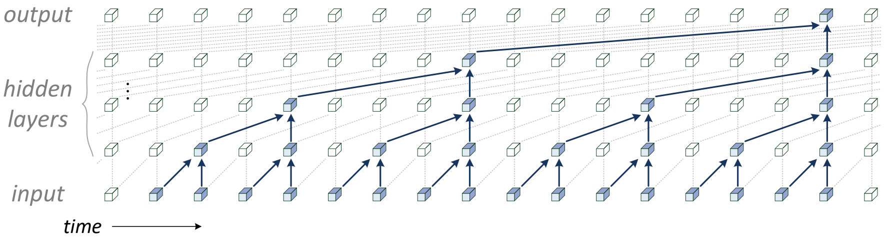
We focus on **dilated ConvNets** for sequence data:



- 1D ConvNets; no pooling; dilated (gapped) conv windows
- Underlie state of the art models for audio & text (e.g. WaveNet)!

Dilated Convolutional Networks

We focus on **dilated ConvNets** for sequence data:



- 1D ConvNets; no pooling; dilated (gapped) conv windows
- Underlie state of the art models for audio & text (e.g. WaveNet)!

Our main result:

Interconnecting hidden layers of networks with different dilations can lead to expressive efficiency

Grid Tensor

Grid Tensor

T – receptive field of a dilated ConvNet

Grid Tensor

T – receptive field of a dilated ConvNet

Network realizes func over T sequence elements:

$$h(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T)$$

Grid Tensor

T – receptive field of a dilated ConvNet

Network realizes func over T sequence elements:

$$h(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T)$$

Discretize each input $\mathbf{x}_i$ to vary between finite # of possibilities

Grid Tensor

T – receptive field of a dilated ConvNet

Network realizes func over T sequence elements:

$$h(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T)$$

Discretize each input $\mathbf{x}_i$ to vary between finite # of possibilities

$\Longrightarrow$ func $h(\cdot)$ boils down to T-dim lookup table, a.k.a. **grid tensor**

Grid Tensor

T – receptive field of a dilated ConvNet

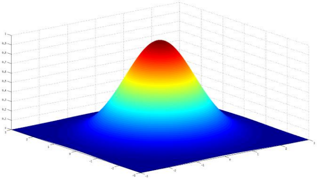
Network realizes func over T sequence elements:

$$h(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T)$$

Discretize each input $\mathbf{x}_i$ to vary between finite # of possibilities

$\Longrightarrow$ func $h(\cdot)$ boils down to T-dim lookup table, a.k.a. **grid tensor**

Illustration for T=2:



$h(\boldsymbol{x}_1, \boldsymbol{x}_2)$

10^{-4}	10^{-3}	10^{-2}	10^{-2}	10^{-2}	10^{-3}	10^{-4}
10^{-3}	10^{-2}	0.1	0.2	0.1	10^{-2}	10^{-3}
10^{-2}	0.1	0.3	0.6	0.3	0.1	10^{-2}
10^{-2}	0.2	0.6	1.0	0.6	0.2	10^{-2}
10^{-2}	0.1	0.3	0.6	0.3	0.1	10^{-2}
10^{-3}	10^{-2}	0.1	0.2	0.1	10^{-2}	10^{-3}
10^{-4}	10^{-3}	10^{-2}	10^{-2}	10^{-2}	10^{-3}	10^{-4}

2D grid tensor

Hierarchical Tensor Decompositions

High-dim tensors (arrays) are exponentially large – cannot be used directly

Hierarchical Tensor Decompositions

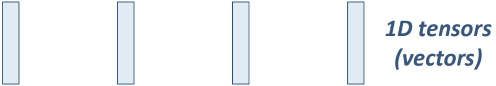
High-dim tensors (arrays) are exponentially large – cannot be used directly

May be represented via **hierarchical tensor decompositions**:

Hierarchical Tensor Decompositions

High-dim tensors (arrays) are exponentially large – cannot be used directly

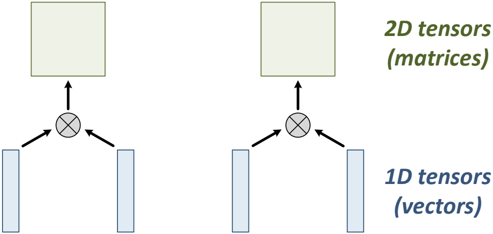
May be represented via **hierarchical tensor decompositions**:



Hierarchical Tensor Decompositions

High-dim tensors (arrays) are exponentially large – cannot be used directly

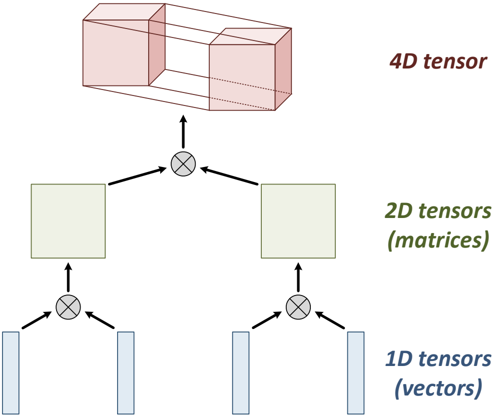
May be represented via **hierarchical tensor decompositions**:



Hierarchical Tensor Decompositions

High-dim tensors (arrays) are exponentially large – cannot be used directly

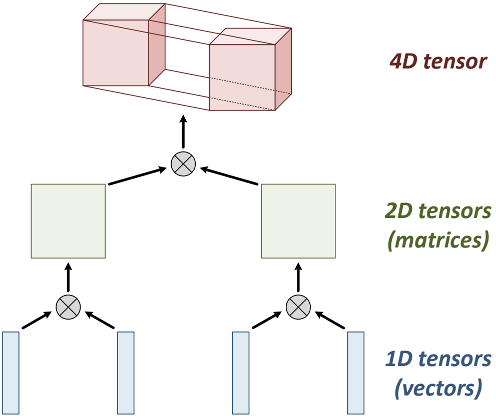
May be represented via **hierarchical tensor decompositions**:



Hierarchical Tensor Decompositions

High-dim tensors (arrays) are exponentially large – cannot be used directly

May be represented via **hierarchical tensor decompositions**:



Hier decomp is characterized by **tree over tensor modes** (axes)

Dilated Convolutional Networks $\longleftrightarrow$ Hierarchical Tensor Decompositions

Observation

Grid tensors of func realized by dilated ConvNet adhere to hier decomp

Dilated Convolutional Networks $\longleftrightarrow$ Hierarchical Tensor Decompositions

Observation

Grid tensors of func realized by dilated ConvNet adhere to hier decomp

Moreover, there is a correspondence:

dilations across network $\longleftrightarrow$ mode tree of decomp

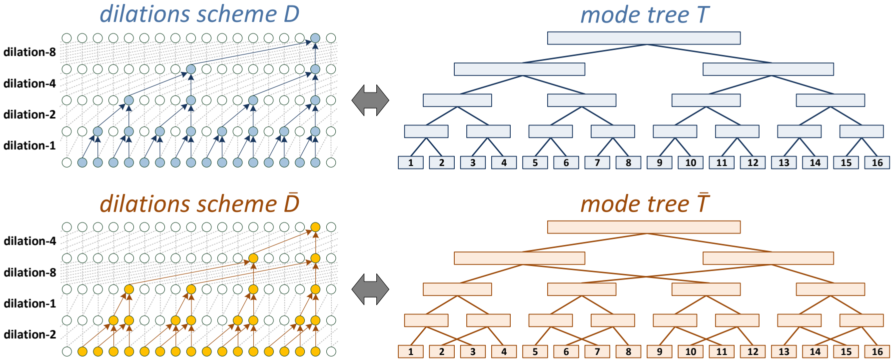
Dilated Convolutional Networks $\longleftrightarrow$ Hierarchical Tensor Decompositions

Observation

Grid tensors of func realized by dilated ConvNet adhere to hier decomp

Moreover, there is a correspondence:

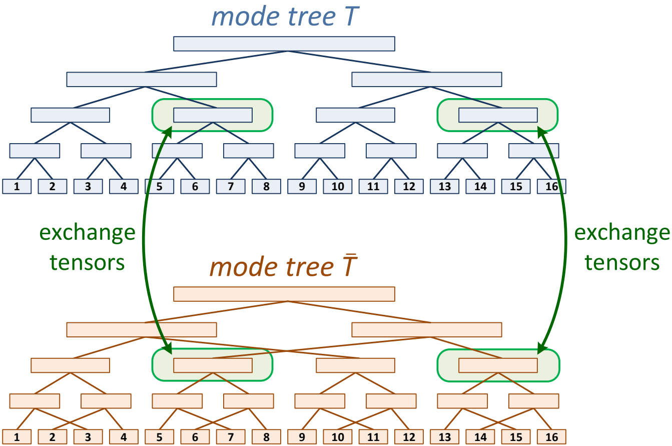
dilations across network $\longleftrightarrow$ mode tree of decomp



Mixed Tensor Decompositions

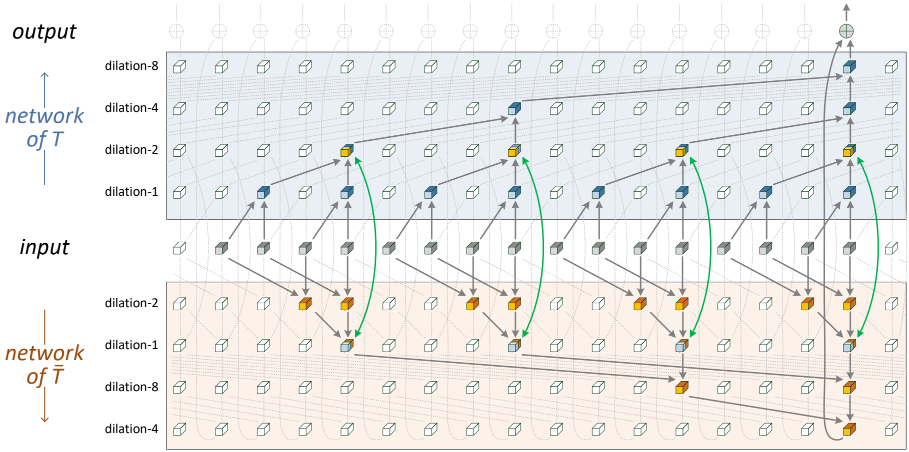
Definition

A **mixed tensor decomposition** blends mode trees T and $\bar{T}$ by running their decomp in parallel, exchanging tensors along the way



Mixed Dilated Convolutional Networks

Mixed decomp captures grid tensors of **mixed dilated ConvNet**, formed by interconnecting networks of T and $\bar{T}$



Expressive Efficiency Analysis

Expressive Efficiency Analysis

Theorem (*result on tensor decomp*)

Expressive Efficiency Analysis

Theorem (*result on tensor decomp*)

- *Any tensor realized by decomp of T or $\bar{T}$ can be realized by their mixed decomp w/at most linear growth*

Expressive Efficiency Analysis

Theorem (*result on tensor decomp*)

- *Any tensor realized by decomp of T or $\bar{T}$ can be realized by their mixed decomp w/at most linear growth*
- *There exist tensors realized by mixed decomp requiring individual decomp of T and $\bar{T}$ to grow quadratically*

Expressive Efficiency Analysis

Theorem (*result on tensor decomp*)

- *Any tensor realized by decomp of T or $\bar{T}$ can be realized by their mixed decomp w/at most linear growth*
- *There exist tensors realized by mixed decomp requiring individual decomp of T and $\bar{T}$ to grow quadratically*

Corollary (*implication for dilated ConvNets*)

Expressive Efficiency Analysis

Theorem (*result on tensor decomp*)

- *Any tensor realized by decomp of T or $\bar{T}$ can be realized by their mixed decomp w/at most linear growth*
- *There exist tensors realized by mixed decomp requiring individual decomp of T and $\bar{T}$ to grow quadratically*

Corollary (*implication for dilated ConvNets*)

- *Any func realized by individual network of T or $\bar{T}$ can be realized by mixed network w/at most linear growth*

Expressive Efficiency Analysis

Theorem (*result on tensor decomp*)

- *Any tensor realized by decomp of T or $\bar{T}$ can be realized by their mixed decomp w/at most linear growth*
- *There exist tensors realized by mixed decomp requiring individual decomp of T and $\bar{T}$ to grow quadratically*

Corollary (*implication for dilated ConvNets*)

- *Any func realized by individual network of T or $\bar{T}$ can be realized by mixed network w/at most linear growth*
- *There exist func realized by mixed network requiring individual networks to grow quadratically*

Expressive Efficiency Analysis

Theorem (*result on tensor decomp*)

- *Any tensor realized by decomp of T or $\bar{T}$ can be realized by their mixed decomp w/at most linear growth*
- *There exist tensors realized by mixed decomp requiring individual decomp of T and $\bar{T}$ to grow quadratically*

Corollary (*implication for dilated ConvNets*)

- *Any func realized by individual network of T or $\bar{T}$ can be realized by mixed network w/at most linear growth*
- *There exist func realized by mixed network requiring individual networks to grow quadratically*

Mixed network is expressively efficient w.r.t. individual ones

Experiment

Experiment

DeepLab model (Chen et al. 2016) showed that together w/other techniques, mixing dilated ConvNets can lead to state of the art

Experiment

DeepLab model (Chen et al. 2016) showed that together w/other techniques, mixing dilated ConvNets can lead to state of the art

Objective: isolate the effect of mixing

Experiment

DeepLab model (Chen et al. 2016) showed that together w/other techniques, mixing dilated ConvNets can lead to state of the art

Objective: isolate the effect of mixing

Model: 2 networks w/diff dilations; interconnections incrementally added

Experiment

DeepLab model (Chen et al. 2016) showed that together w/other techniques, mixing dilated ConvNets can lead to state of the art

Objective: isolate the effect of mixing

Model: 2 networks w/diff dilations; interconnections incrementally added

Task: phoneme recognition on TIMIT (no pre/post-processing)

Experiment

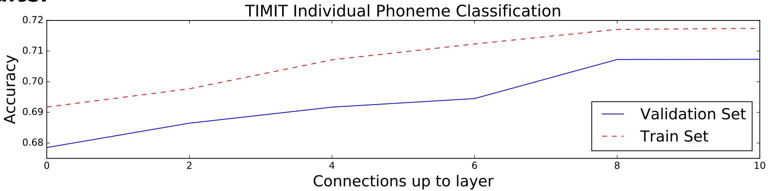
DeepLab model (Chen et al. 2016) showed that together w/other techniques, mixing dilated ConvNets can lead to state of the art

Objective: isolate the effect of mixing

Model: 2 networks w/diff dilations; interconnections incrementally added

Task: phoneme recognition on TIMIT (no pre/post-processing)

Results:



Experiment

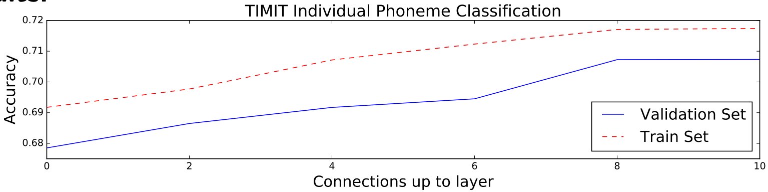
DeepLab model (Chen et al. 2016) showed that together w/other techniques, mixing dilated ConvNets can lead to state of the art

Objective: isolate the effect of mixing

Model: 2 networks w/diff dilations; interconnections incrementally added

Task: phoneme recognition on TIMIT (no pre/post-processing)

Results:



Interconnections improve accuracy, with no overhead in computation or model capacity

Conclusion

Conclusion

- **Expressive efficiency**: concept formalizing representational superiority

Conclusion

- **Expressive efficiency**: concept formalizing representational superiority
- We studied expressive efficiency of **connectivity** for dilated ConvNets

Conclusion

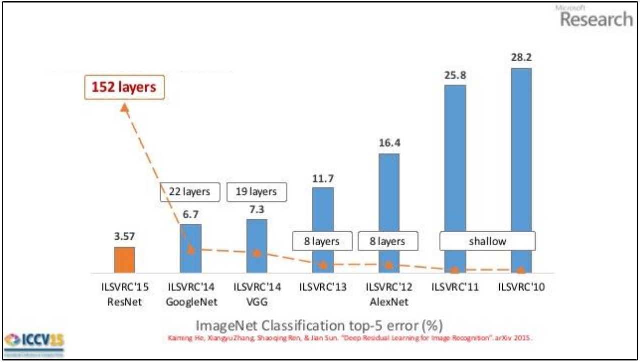
- **Expressive efficiency**: concept formalizing representational superiority
- We studied expressive efficiency of **connectivity** for dilated ConvNets
- Analysis shows **interconnections can lead to expressive efficiency**

Conclusion

- **Expressive efficiency**: concept formalizing representational superiority
- We studied expressive efficiency of **connectivity** for dilated ConvNets
- Analysis shows **interconnections can lead to expressive efficiency**
- Experiment demonstrates **gains in accuracy** (w/o overheads)

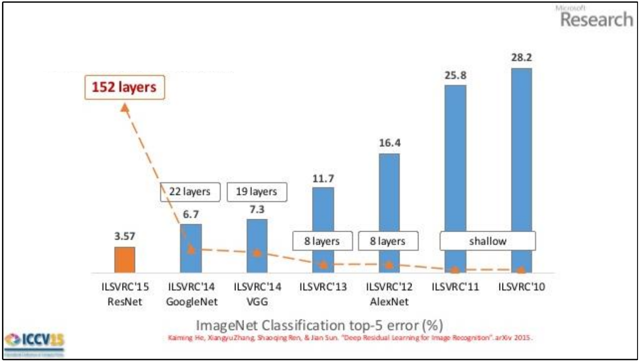
Final Thought

Expressive efficiency coincides w/improved accuracies in the case of depth



Final Thought

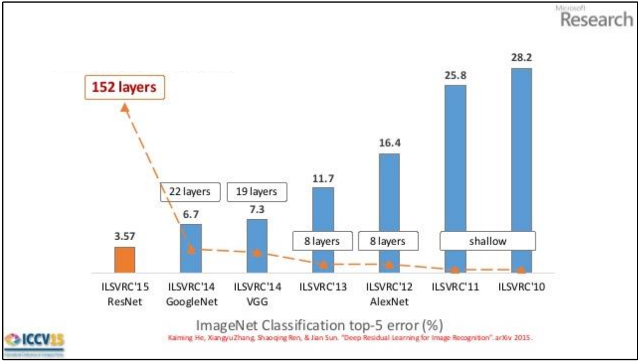
Expressive efficiency coincides w/improved accuracies in the case of depth



Same holds for connectivity!

Final Thought

Expressive efficiency coincides w/improved accuracies in the case of depth



Same holds for connectivity!

Expressive efficiency may be key in developing new theoretical tools for deep network design

Thank You