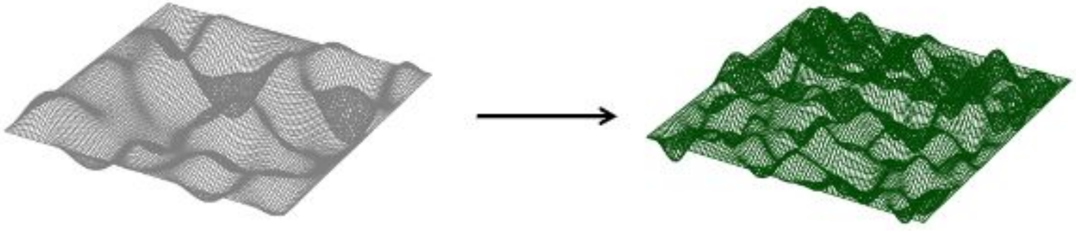
S4ND: Modeling Images and Videos as Multidimensional Signals Using State Spaces

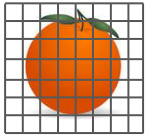
Eric Nguyen*, Karan Goel*, Albert Gu*, Gordon W. Downs, Tri Dao,
Preey Shah, Stephen A. Baccus, Christopher Ré



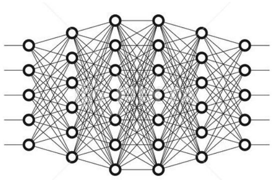
* equal contribution

Current vision approaches model pixels, not signals

SotA vision models



Discrete pixels

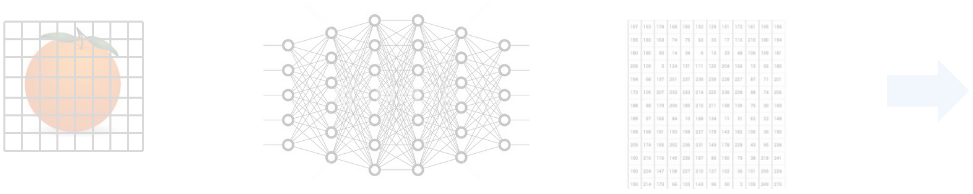


Discrete representation

fixed resolutions

Current vision approaches model pixels, not signals

SotA vision models



Discrete pixels

Discrete representation

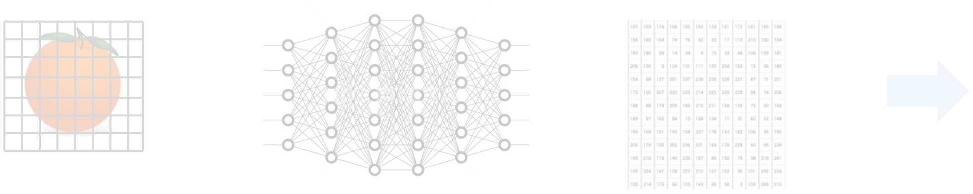
fixed resolutions

Continuous signals

Discrete pixels

Current vision approaches model pixels, not signals

SotA vision models

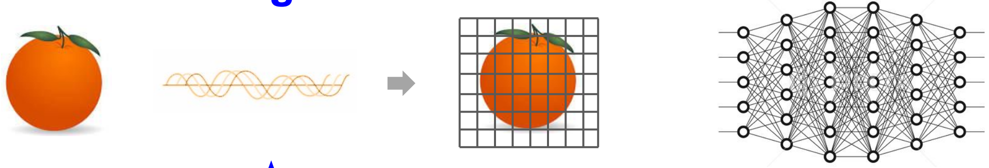


Discrete pixels

Discrete representation

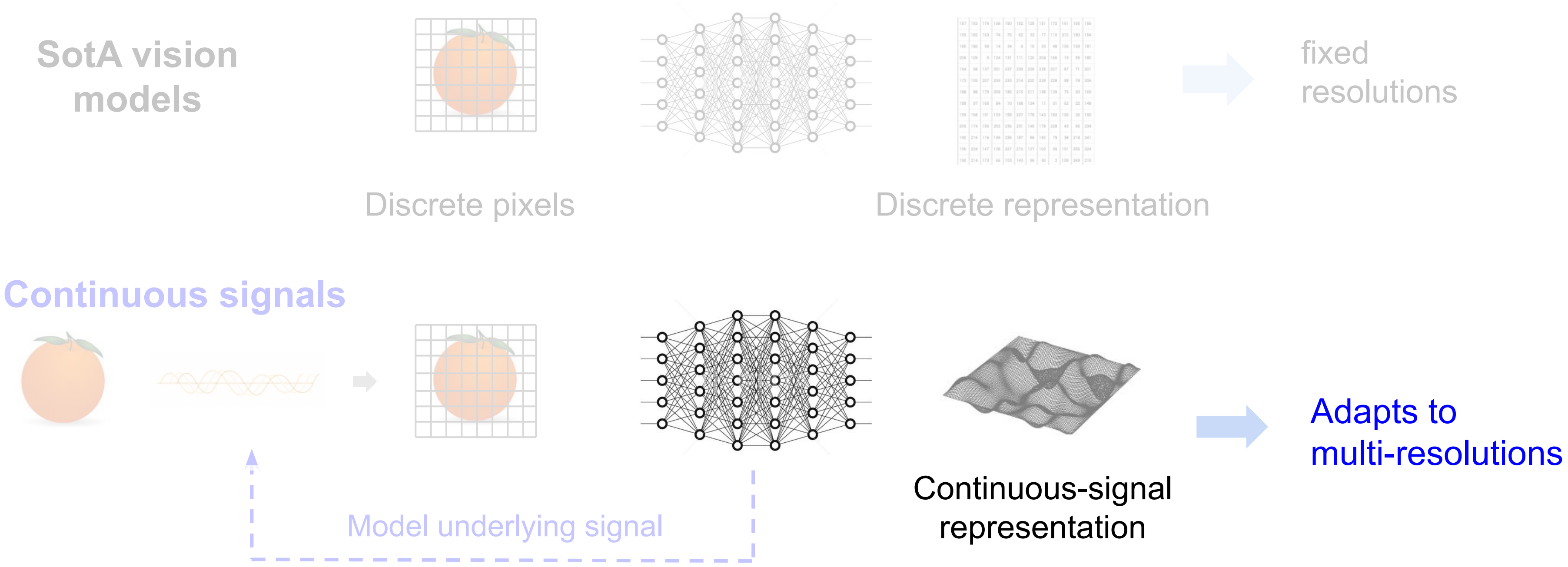
fixed resolutions

Continuous signals



Model underlying signal

Current vision approaches model pixels, not signals



Continuous convolutions w/ S4: SotA on long range tasks

Efficiently Modeling Long Sequences with Structured State Spaces

Albert Gu, Karan Goel, and Christopher Ré

Department of Computer Science, Stanford University

{albertgu,krng}@stanford.edu, chrismre@cs.stanford.edu

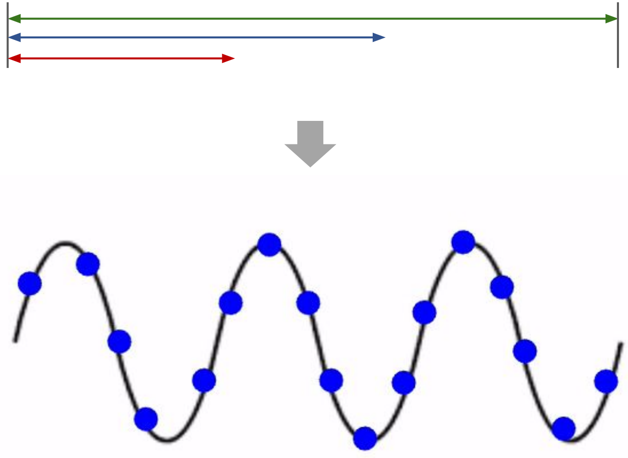
CKCONV: CONTINUOUS KERNEL CONVOLUTION FOR SEQUENTIAL DATA

David W. Romero[1], **Anna Kuzina**[1], **Erik J. Bekkers**[2], **Jakub M. Tomczak**[1], **Mark Hoogendoorn**[1]
[1] Vrije Universiteit Amsterdam [2] University of Amsterdam
The Netherlands
{d.w.romeroguzman, a.kuzina}@vu.nl

FLEXCONV: CONTINUOUS KERNEL CONVOLUTIONS WITH DIFFERENTIABLE KERNEL SIZES

David W. Romero[*,1], **Robert-Jan Bruintjes**[*,2],
Erik J. Bekkers[3], **Jakub M. Tomczak**[1], **Mark Hoogendoorn**[1], **Jan C. van Gemert**[2]
[1] Vrije Universiteit Amsterdam [2] Delft University of Technology [3] University of Amsterdam
The Netherlands
d.w.romeroguzman@vu.nl, r.bruintjes@tudelft.nl

Flexible kernel sizes



1D signal, sampled at different rates

Continuous convolutions w/ S4: SotA on long range tasks

Efficiently Modeling Long Sequences with Structured State Spaces

Albert Gu, Karan Goel, and Christopher Ré

Department of Computer Science, Stanford University

{albertgu,krng}@stanford.edu, chrismre@cs.stanford.edu

CKCONV: CONTINUOUS KERNEL CONVOLUTION FOR SEQUENTIAL DATA

David W. Romero[1], Anna Kuzina[1], Erik J. Bekkers[2], Jakub M. Tomczak[1], Mark Hoogendoorn[1]
[1] Vrije Universiteit Amsterdam [2] University of Amsterdam
The Netherlands
{d.w.romeroguzman, a.kuzina}@vu.nl

FLEXCONV: CONTINUOUS KERNEL CONVOLUTIONS WITH DIFFERENTIABLE KERNEL SIZES

David W. Romero[*,1], Robert-Jan Bruintjes[*,2],
Erik J. Bekkers[3], Jakub M. Tomczak[1], Mark Hoogendoorn[1], Jan C. van Gemert[2]
[1] Vrije Universiteit Amsterdam [2] Delft University of Technology [3] University of Amsterdam
The Netherlands
d.w.romeroguzman@vu.nl, r.bruintjes@tudelft.nl

Long Range Arena Benchmark

Benchmark spanning text, images, symbolic reasoning (length 1K-16K)

Model	ListOps	Text	Retrieval	Image	Pathfinder	Path-X	Avg
Random	10.00	50.00	50.00	10.00	50.00	50.00	36.67
Transformer	36.37	64.27	57.46	42.44	71.40	✗	53.66
Local Attention	15.82	52.98	53.39	41.46	66.63	✗	46.71
Sparse Trans.	17.07	63.58	59.59	44.24	71.71	✗	51.03
Longformer	35.63	62.85	56.89	42.22	69.71	✗	52.88
Linformer	35.70	53.94	52.27	38.56	76.34	✗	51.14
Reformer	37.27	56.10	53.40	38.07	68.50	✗	50.56
Sinkhorn Trans.	33.67	61.20	53.83	41.23	67.45	✗	51.23
Synthesizer	36.99	61.68	54.67	41.61	69.45	✗	52.40
BigBird	36.05	64.02	59.29	40.83	74.87	✗	54.17
Linear Trans.	16.13	65.90	53.09	42.34	75.30	✗	50.46
Performer	18.01	65.40	53.82	42.77	77.05	✗	51.18
FNet	35.33	65.11	59.61	38.67	77.80	✗	54.42
Nyströmformer	37.15	65.52	79.56	41.58	70.94	✗	57.46
Luna-256	37.25	64.57	79.29	47.38	77.72	✗	59.37
S4	**58.35**	**76.02**	**87.09**	**87.26**	**86.05**	**88.10**	**80.48**

S4 outperforms by **+20-30 pts**

S4ND: extending S4 to multidimensional signals

1D input signals

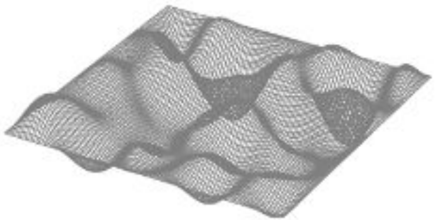
1D output signal
representation

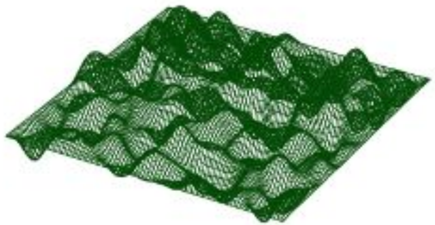
S4ND: extending S4 to multidimensional signals

1D input signals

1D output signal
representation

S4ND:

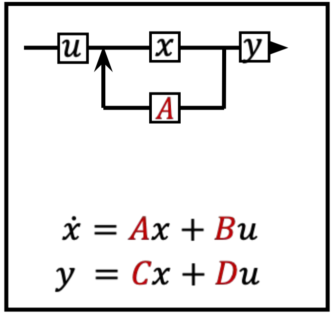




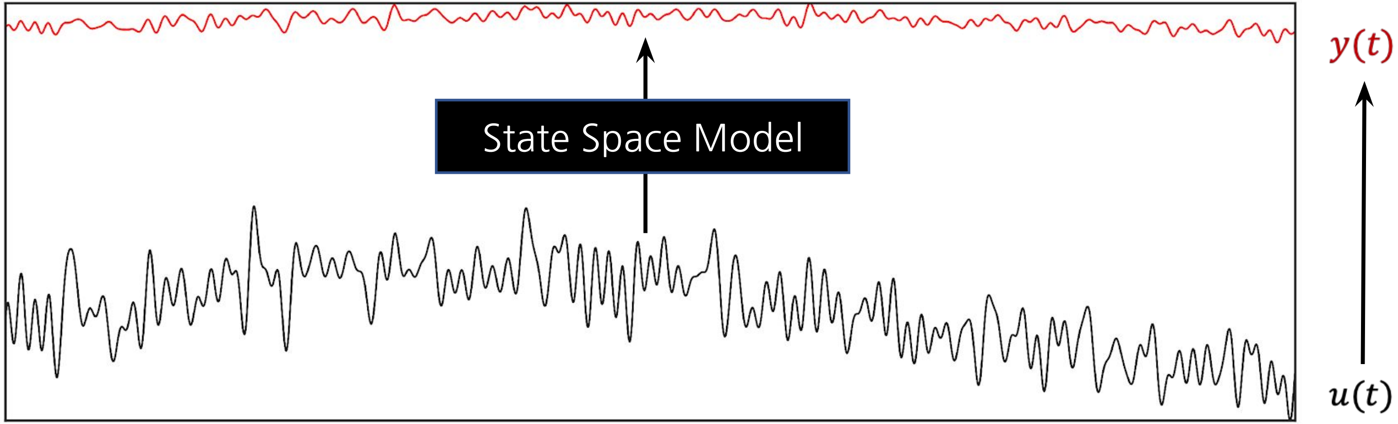
N-D input signals

N-D output signal representations

S4 and State Space Models (SSMs)

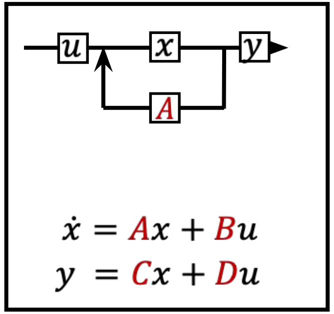


Continuous-signal SSM

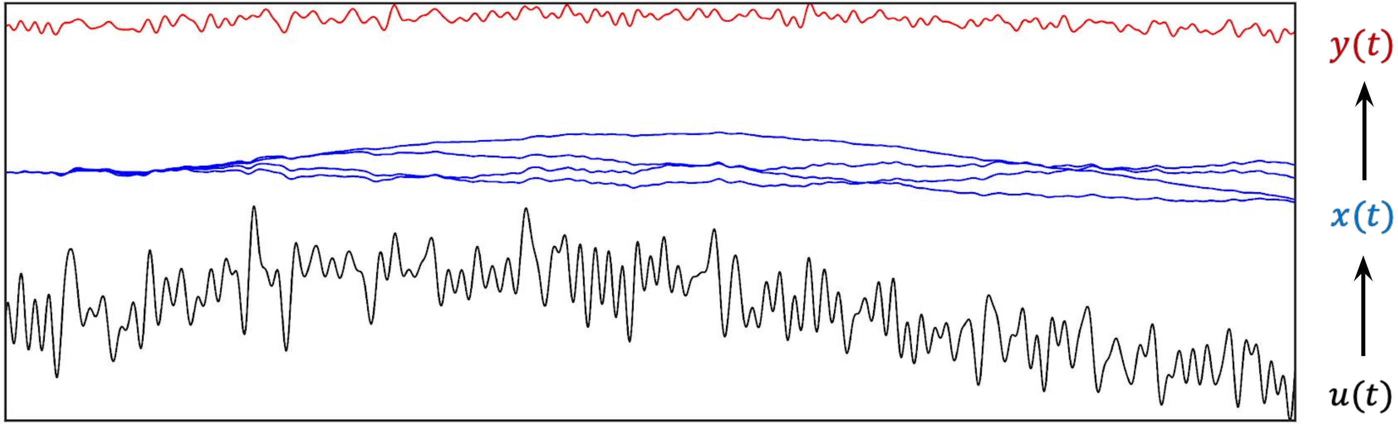


SSMs are just a **sequence modeling layer**

S4 and State Space Models (SSMs)

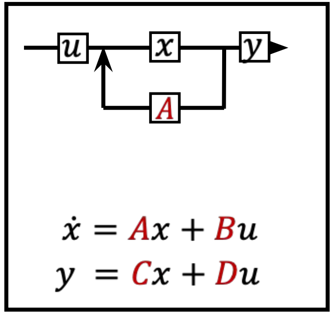


Continuous-signal SSM



SSM maps **input** to **output** through a higher-dimensional **state**

We can create a global kernel from the SSM



Continuous-signal SSM

Discretize

$$\xrightarrow{\Delta t}$$

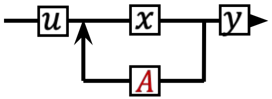
$$y_k = \overline{CA}^k\overline{B}u_0 + \overline{CA}^{k-1}\overline{B}u_1 + \cdots + \overline{CAB}u_{k-1} + \overline{CB}u_k$$

$$\overline{K} \in \mathbb{R}^L := (\overline{CB}, \overline{CAB}, \ldots, \overline{CA}^{L-1}\overline{B})$$

$$\boxed{y = \overline{K} * u}$$

Produces a global convolutional kernel

We can create a global kernel from the SSM



$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

Continuous-signal SSM

Discretize

$$\xrightarrow{\Delta t}$$

$$y_k = \overline{C}\overline{A}^k\overline{B}u_0 + \overline{C}\overline{A}^{k-1}\overline{B}u_1 + \cdots + \overline{C}\overline{A}\overline{B}u_{k-1} + \overline{C}\overline{B}u_k$$

$$\overline{K} \in \mathbb{R}^L := (\overline{C}\overline{B}, \overline{C}\overline{A}\overline{B}, \ldots, \overline{C}\overline{A}^{L-1}\overline{B})$$

$$\boxed{y = \overline{K} * u}$$

Produces a global convolutional kernel

Key idea: turn the standard 1D SSM into S4ND

S4 ➡ **S4ND**

S4	S4ND
1 SSM	SSM per dimension
1-D Ordinary Diff Eq	N-D Partial Diff Eq
1D continuous conv	N-D continuous conv

- S4ND: governed by an independent SSM per dimension
- Equivalent to continuous convolutions in N-dimensions
- Fast and easy to implement

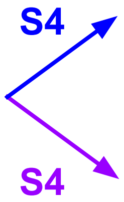
Example: S4ND flow chart for 2D kernel

$$x'(t) = \boldsymbol{A}x(t) + \boldsymbol{B}u(t)$$
$$y(t) = \boldsymbol{C}x(t)$$

Initialize SSM
parameters for
each S4 kernel

Example: S4ND flow chart for 2D kernel

$$x'(t) = \boldsymbol{A}x(t) + \boldsymbol{B}u(t)$$
$$y(t) = \boldsymbol{C}x(t)$$

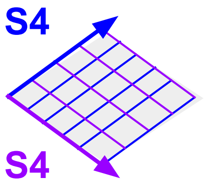


Initialize SSM parameters for each S4 kernel

For 2D input, create N=2 independent S4 kernels, spanning full length (e.g. 224 each)

Example: S4ND flow chart for 2D kernel

$$x'(t) = \boldsymbol{A}x(t) + \boldsymbol{B}u(t)$$
$$y(t) = \boldsymbol{C}x(t)$$



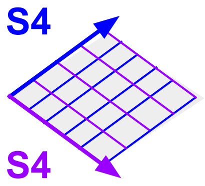
Initialize SSM parameters for each S4 kernel

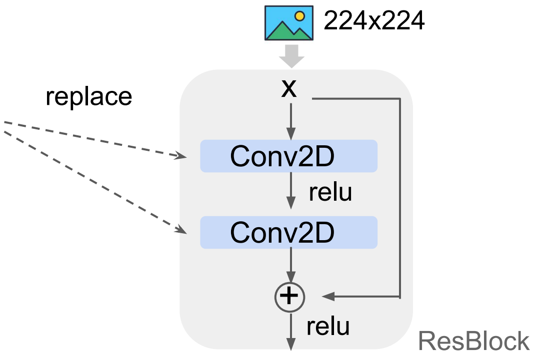
For 2D input, create N=2 independent S4 kernels, spanning full length (e.g. 224 each)

Outer product creates a global 2D convolutional kernel, (e.g. size 224x224)

Example: S4ND flow chart for 2D kernel

$$x'(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t)$$





Initialize SSM parameters for each S4 kernel

For 2D input, create N=2 independent S4 kernels, spanning full length (e.g. 224 each)

Outer product creates a global 2D convolutional kernel, (e.g. size 224x224)

Replace local Conv2D (e.g. 3x3) with global kernel

S4ND is the **1st continuous-signal model** to be competitive w/SotA models on large-scale image & video data

Vision experiments applying S4ND in 1D, 2D & 3D

ViT

Self-attn

ConvNeXt

Conv2D

Vision experiments applying S4ND in 1D, 2D & 3D

ViT	ConvNeXt		ConvNeXt-3D
~~Self-attn~~	~~Conv2D~~	Temporal inflation → 2D to 3D	~~Conv3D~~
S4ND	S4ND		**S4ND**

MODEL	DATASET	PARAMS	ACC	
ViT-B	ImageNet	88.0M	78.9	
S4ND-ViT-B	ImageNet	88.8M	**80.4**	+1.5%
ConvNeXt-T	ImageNet	28.4M	82.1	
S4ND-ConvNeXt-T	ImageNet	30.0M	**82.2**	+0.1%
Conv2D-ISO	CIFAR-10	2.2M	93.7	
S4ND-ISO	CIFAR-10	5.3M	**94.1**	+0.4%
ConvNeXt-M	Celeb-A	9.2M	91.0	
S4ND-ConvNeXt-M	Celeb-A	9.6M	**91.3**	+0.3%

	PARAMS	FLOW	RGB	
Inception-I3D	25.0M	61.9	49.8	
ConvNeXt-I3D	28.5M	-	58.1	
ConvNeXt-S3D	27.9M	-	58.6	
S4ND-ConvNeXt-3D	31.4M	-	**62.1**	+3.5%

HMDB-51 video dataset

S4ND resolution capabilities

Zero-shot resolution:

- Train at lower res, test on ***unseen*** higher res

Train

Test

S4ND resolution capabilities in 2 settings

Zero-shot resolution:

- Train at lower res, test on ***unseen*** higher res

Train

Test

Progressive resizing:

- Gradually upsample, and train and test on final resolution

Train

Train

Train/Test

Fast → Slow

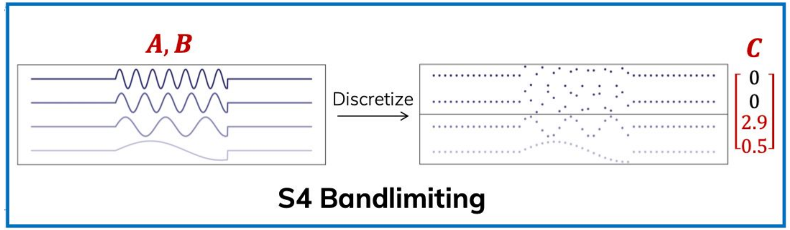
New bandlimiting regularizer helps both resolution settings

Zero-shot resolution:

- Train at lower res, test on ***unseen*** higher res

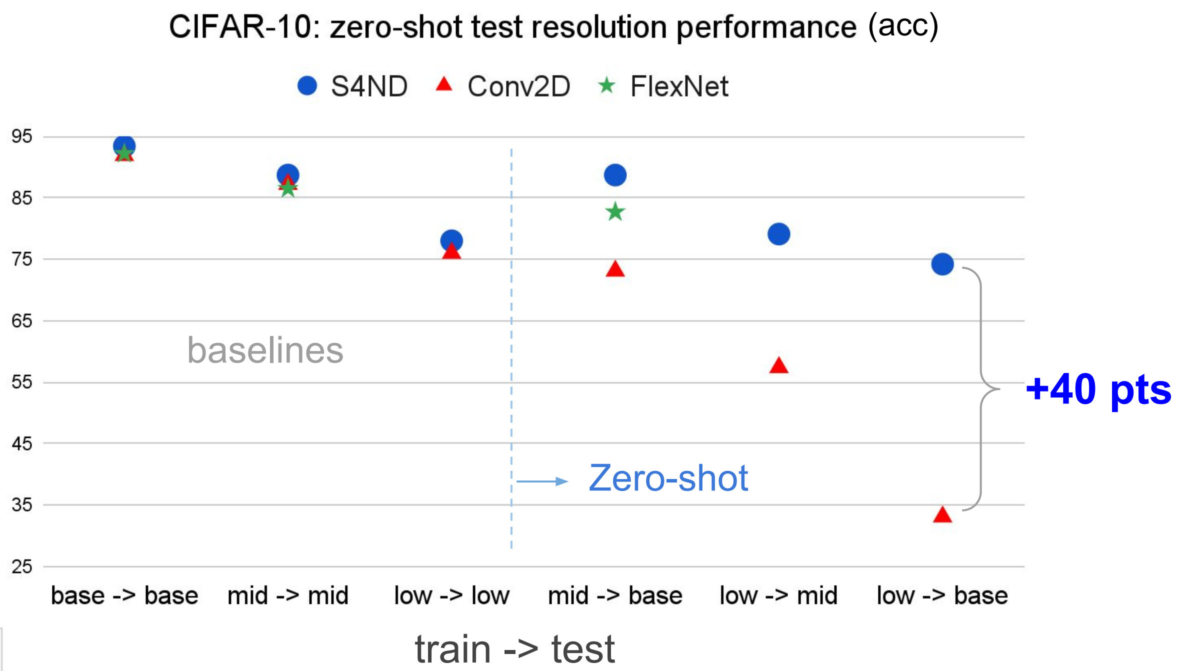
Progressive resizing:

- Gradually upsample, and train and test on final resolution



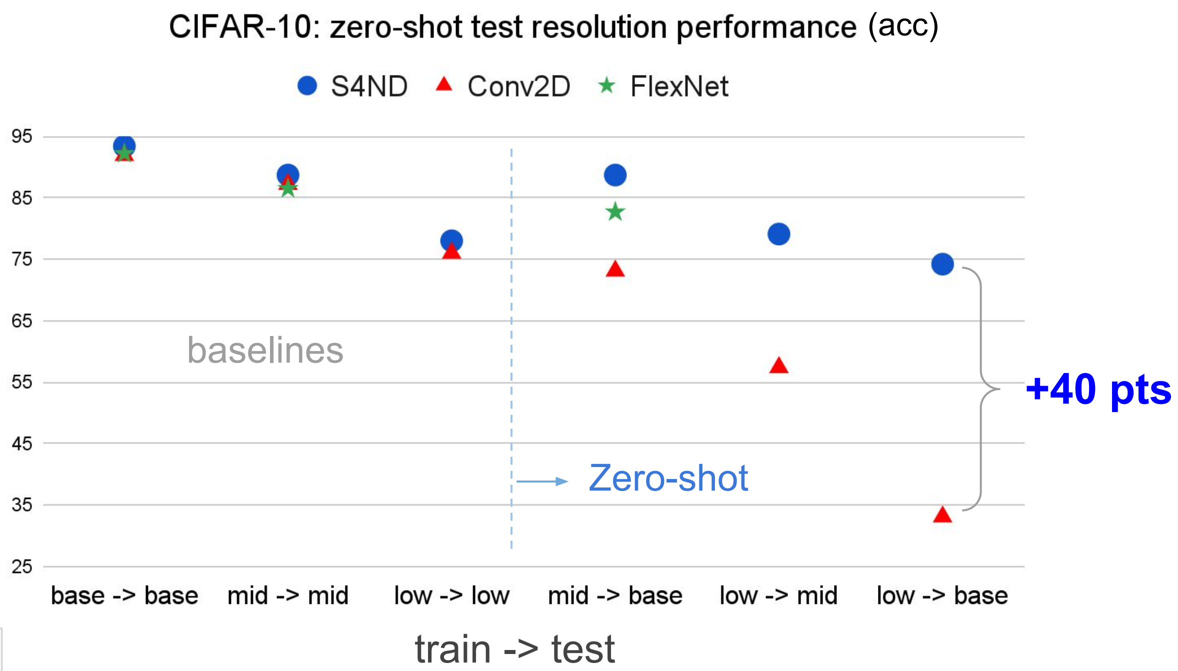
- Bandlimiting regularizer as a low pass filter
- Removes high frequencies, addresses aliasing
- Controlled by SSM parameters
  - (details in paper)

S4ND outperforms baselines in all zero-shot settings



CIFAR-10 Resolutions	low: 8x8 mid: 16x16 base: 32x32

S4ND outperforms baselines in all zero-shot settings



CIFAR-10 Resolutions	low: 8x8 mid: 16x16 base: 32x32

Summary

- S4ND -> S4 extends to N dimensions
- Strong candidate for general vision backbones
  - Boosts or matches performance in images and videos
  - Ability to train and test at different resolutions
- Excited for what other capabilities S4ND can unlock!
  - In both vision & other fields that seek to model continuous-signals

Contact us

{etnguyen,albertgu,gwdowns,preey,trid,baccus}@stanford.edu,
{kgoel,chrismre}@cs.stanford.edu

Thanks!