MgNet: Algorithms, Theories, and Applications

Juncai He

Applied Mathematics and Computational Sciences, KAUST
juncai.he@kaust.edu.sa

Morgan State University
CBMS: Deep Learning and Numerical PDEs
June 20th, 2023

Outline

Feature extraction: a constrained linear model

Linearly separable feature

A linear model: given an image g, find its feature u satisfying

$$A * u = g \tag{1}$$

with a constraint

$$u \geq 0. \tag{2}$$

Namely

$$u = \phi(g).$$

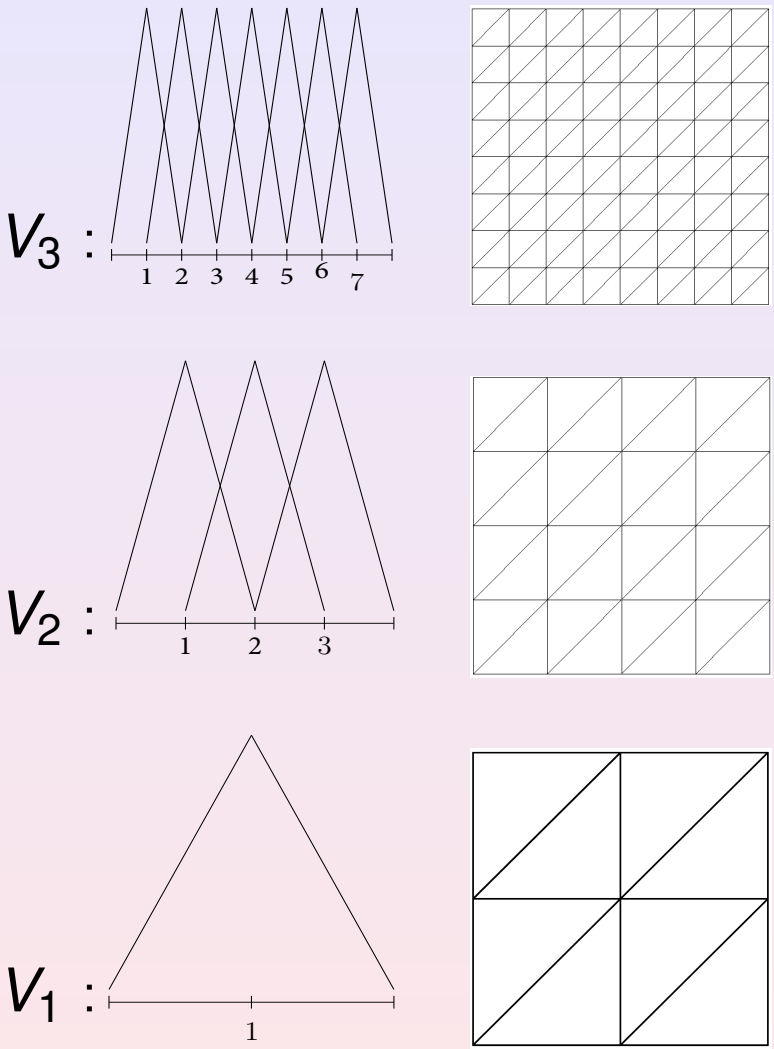
Questions:

1. What is A? (to be trained ...)
2. How to solve (1)?

Iterative method $\Rightarrow$ Multigrid method

Ref: J. He and J. Xu 2019; J. He, L. Zhang, J. Xu and J. Zhu 2021

Multigrid methods for $A * u = g$ (convolution language)



$$\phi_{3,1} = \frac{1}{2}\phi_{2,1} + \phi_{2,2} + \frac{1}{2}\phi_{2,3}$$

Smoothing and restriction

- For $\ell = 1 : J$
  - For $i = 1 : \nu_\ell$

$$u^\ell \leftarrow u^\ell + B^\ell * (g^\ell - A^\ell * u^\ell).$$

  - Form restricted residual and set initial guess:

$$u^{\ell+1,0} \leftarrow \Pi_\ell^{\ell+1} *_2 u^\ell,$$
$$g^{\ell+1} \leftarrow R_\ell^{\ell+1} *_2 (g^\ell - A^\ell * u^\ell) + A^{\ell+1} * u^{\ell+1,0}.$$

Prolongation with post-smoothing

- For $\ell = J - 1 : 1$

$$u_\ell \leftarrow u_\ell + R_\ell^{\ell+1} *_2^\top (u^{\ell+1} - u^{\ell+1,0}).$$

  - For $i = 1 : \nu'_\ell$

$$u^\ell \leftarrow u^\ell + [B^\ell]' * (g^\ell - A^\ell * u^\ell)$$

Ref: J. He and J. Xu 2019

From multigrid to MgNet

MgNet (CNN): a "trained" multigrid method:

1. Initialization of inputs:
   $g^1 \leftarrow \theta * g, \quad u^1 \leftarrow 0$
2. For $\ell = 1 : J$
   - For $i = 1 : \nu_\ell$

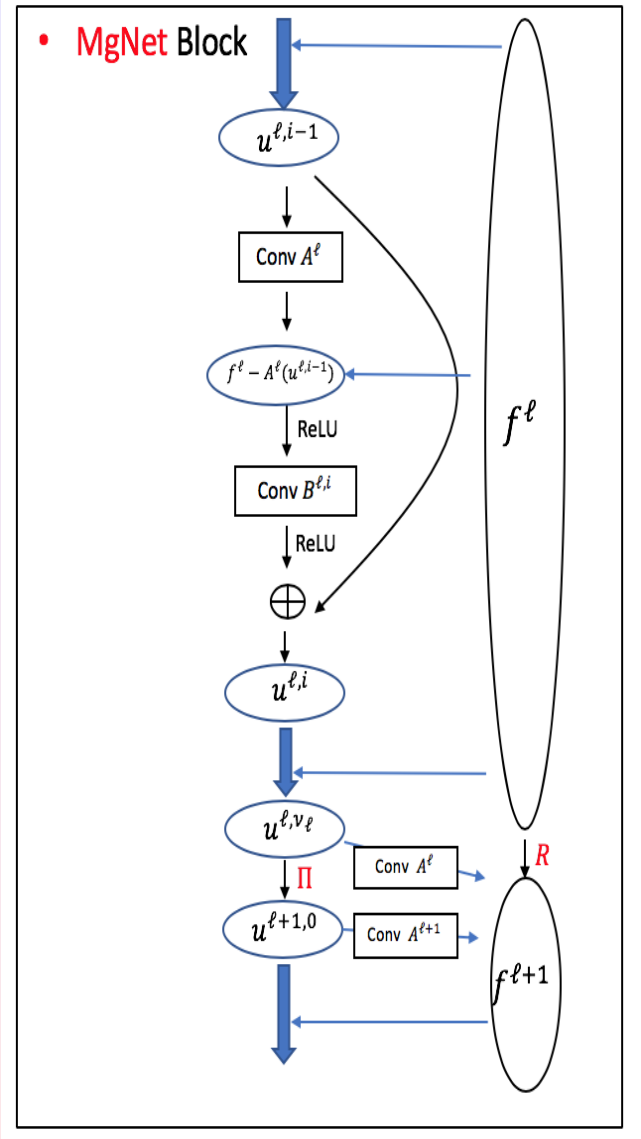
$$u^\ell \leftarrow u^\ell + \sigma \circ B^{\ell,i} * \sigma(g^\ell - A^\ell * u^\ell).$$

   - Restriction-Pooling

$$u^{\ell+1,0} \leftarrow \Pi_\ell^{\ell+1} *_2 u^\ell$$
$$g^{\ell+1} \leftarrow R_\ell *_2 (g^\ell - A^\ell * u^\ell) + A^{\ell+1} * u^{\ell+1,0}.$$

3. Output = feature map :

$$\phi(g) = u^J.$$



MgNet: From multigrid to CNN

Multigrid:

- $A^\ell, B^\ell, \Pi_\ell^{\ell+1}, R_\ell^{\ell+1}$ are all given a priori.

CNN:

- Almost identically same structure as multigrid!
- $A^{\ell,i}, B^{\ell,i}, R_\ell^{\ell+1}$ are all trained!
- Activation, ReLU, is introduced (to drop-off negative pixel values).
- Extra channels are introduced.

- J. Xu, Deep Neural Networks and Multigrid Methods, (Lecture Notes), 2017-2023.
- J. He, J. Xu. MgNet: A Unified Framework of Multigrid and Convolutional Neural Network. Sci China Math, 2019, 62: 1331-1354.
- J. He, L. Li, J. Xu. Approximation Properties of Deep ReLU CNNs. Research in the Mathematical Sciences, 2022, 9(3).
- J. He, J. Xu, L. Zhang, J. Zhu. An Interpretive Constrained Linear Model for ResNet and MgNet. Neural Networks, 2023, 162: 384-392.
- J. Zhu, J. He, J. Xu, L. Zhang. FV-MgNet: Fully Connected V-cycle MgNet for Interpretable Time Series Forecasting. Journal of Computational Science, 2023 69: 102005.
- J. Zhu, J. He, Q. Huang, L. Zhang. An Enhanced V-cycle MgNet Model for Operator Learning in Numerical Partial Differential Equations. Accepted by Computational Geosciences 2023.

Comparisons between MgNet and classic CNNs on CIFAR-10 and CIFAR-100

Model	Accuracy	# Parameters
VGG19	93.56	20.0M
ResNet18	95.28	11.2M
pre-act ResNet18	95.08	10.2M
MgNet[2,2,2,2],256	96.00	8.2M

Table: Comparisons between MgNet and classical CNNs on CIFAR-10

Model	Accuracy	# Parameters
VGG19	70.48	20.04
ResNet18	77.54	11.2M
pre-act ResNet18	77.29	11.2M
MgNet[2,2,2,2],256	79.94	8.3M
MgNet[2,2,2,2],512	81.35	33.1M
MgNet[2,2,2,2],1024	82.46	132.2M

Table: Comparisons between MgNet and classical CNNs on CIFRA-100

Comparisons between MgNet and classic CNNs on ImageNet

Model	Accuracy	# Parameters
VGG19	71.30	20.0M
ResNet18	72.12	11.2M
pre-act ResNet18	72.34	11.2M
MgNet[3,4,6,3][64,128,256,512]	73.78	9.9M
ResNet152	77.65	61.0M
DenseNet264	77.85	34.2M
MgNet[2,2,4,2][128,256,512,1024]	77.58	38.5M
MgNet[3,4,8,4][128,256,512,1024], $B^{\ell,i}$	78.73	85.4M

Table: Comparisons between MgNet and classical CNNs on ImageNet

MgNet on CIFRA-10 and CIFRA-100

Dataset	Model	Accuracy	# Parameters
CIFRA-10	MgNet[2,2,2,2],256	96.00	8.2M
CIFRA-10	MgNet[3,4,6,3],256	95.98	8.3M
CIFRA-10	MgNet[3,4,6,3],512	96.48	33.1M
CIFRA-100	MgNet[2,2,2,2],256	79.94	8.3M
CIFRA-100	MgNet[8,2,2,2],256, $B^{\ell,i}$	81.42	14.3M
CIFRA-100	MgNet[2,2,2,2],512	81.35	33.1M
CIFRA-100	MgNet[2,2,2,2],1024	82.46	132.2M

Table: The stability and scalability of MgNet on CIFRA-10 and CIFRA-100

MgNet on ImageNet

Model	Accuracy	# Parameters
MgNet[3,4,6,3][64,128,256,512]	73.78	9.9M
MgNet[2,2,8,2][64,128,256,512], $B^{\ell,i}$	75.18	16.6M
MgNet[2,2,4,2][128,256,512,1024]	77.58	38.5M
MgNet[3,4,6,3][128,256,512,1024], $B^{\ell,i}$	78.59	71.3M
MgNet[3,4,8,4][128,256,512,1024], $B^{\ell,i}$	78.73	85.4M

Table: The stability and scalability of MgNet on ImageNet

MgNet vs Transformer on ImageNet

Model	Type	Accuracy	Parameters
DeiT-Small	Transformer	79.8	22.1M
PVT-Small	Transformer	79.8	24.5M
ConvMixer	Transformer	80.2	21.1M
CrossViT-Small	Transformer	81.0	26.7M
Swin-Tiny	Transformer	81.2	28.3M
CvT-13	Transformer	81.6	20.0M
CoAtNet-0	Transformer	81.6	25.0M
CaiT-XS-24	Transformer	81.8	26.6M
ResNet-50	CNN	80.4	25.0M
MgNet-small	CNN	**81.0**	26.1M
MgNet	CNN	**82.0**	39.3M
CMT-XS	CNN+Transformer	81.8	15.2M
MgNet-CMT-XS	CNN +Transformer	**82.6**	17.9M
MgNet-CMT	CNN +Transformer	**83.4**	30.1M

Table: ImageNet results of transformers and CNNs

An encouraging observation:

MgNet has competitive performance with transformer models.

CNN: ResNet and Pre-act ResNet

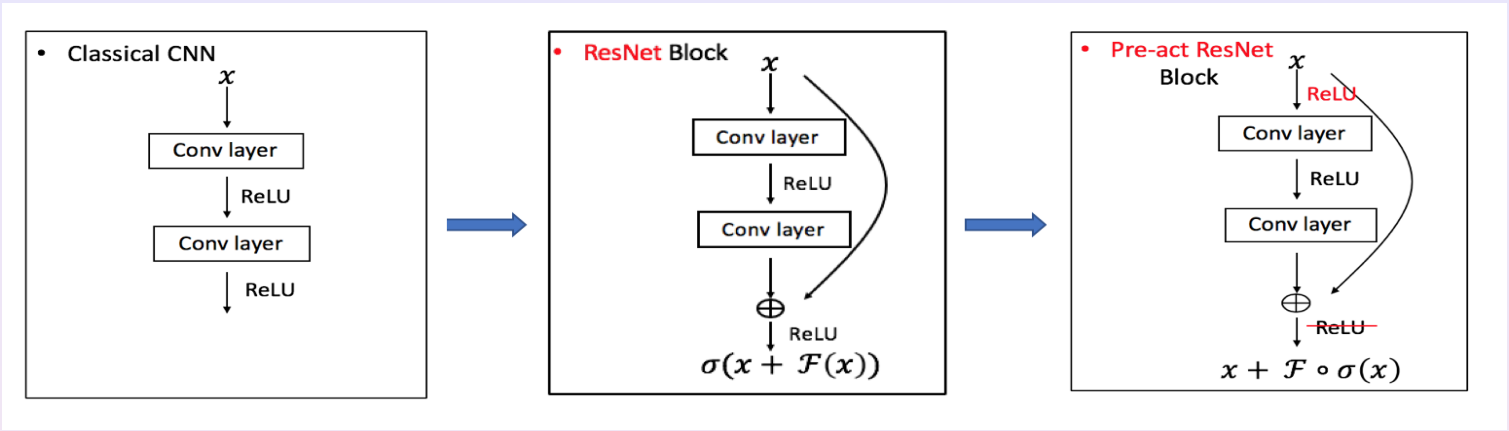


Figure: From classical CNNs to ResNet and Pre-act ResNet (K. He, X. Zhang, S. Ren and J. Sun, 2015 and 2016)

Mathematical formula

ResNet

$$r^{\ell,i} = \sigma\left(r^{\ell,i-1} + \alpha^{\ell,i} * \sigma \circ \beta^{\ell,i} * r^{\ell,i-1}\right). \tag{3}$$

Pre-act ResNet

$$r^{\ell,i} = r^{\ell,i-1} + \alpha^{\ell,i} * \sigma \circ \beta^{\ell,i} * \sigma(r^{\ell,i-1}). \tag{4}$$

Pre-act ResNet: a "residual" version of MgNet

Theorem (MgNet and pre-act ResNet, He and Xu 2019)

The MgNet model recovers the pre-act ResNet (K. He et al 2016) as follows

$$r^{\ell,i} = r^{\ell,i-1} + A^{\ell,i} * \sigma \circ B^{\ell,i} * \sigma(r^{\ell,i-1}), \quad i = 1 : \nu_\ell, \tag{5}$$

where

$$r^{\ell,i} = g^\ell - A^\ell * u^{\ell,i}.$$

Proof.

$$\begin{aligned}
u^{\ell,i} &= u^{\ell,i-1} + \sigma \circ B^{\ell,i} * \sigma(g^\ell - A^\ell * u^{\ell,i-1}),\\
\Rightarrow A^\ell * u^{\ell,i} &= A^\ell * u^{\ell,i-1} + A^\ell * \sigma \circ B^{\ell,i} * \sigma(g^\ell - A^\ell * u^{\ell,i-1}),\\
\Rightarrow g^\ell - A^\ell * u^{\ell,i} &= g^\ell - A^\ell * u^{\ell,i} - A^\ell * \sigma \circ B^{\ell,i} * \sigma(g^\ell - A^\ell * u^{\ell,i-1}),\\
\Rightarrow r_\ell^i &= r^{\ell,i} = r^{\ell,i-1} + A^{\ell,i} * \sigma \circ B^{\ell,i} * \sigma(r^{\ell,i-1}).
\end{aligned} \tag{6}$$

□

Approximation power of MgNet and CNNs

Theorem (H, Li and Xu, 2022)

Let $\Omega \subset \mathbb{R}^{d\times d}$ be bounded and $f : \Omega \mapsto \mathbb{R}$, then there exist a MgNet function $\tilde{f} : R^{d\times d} \mapsto \mathbb{R}$ with multi-channel 3×3 kernels with padding, where

$$\begin{aligned} &\textit{depth (number of convolutional layers):} \quad L = \lfloor d/4 \rfloor, \\ &\textit{width (number of channels at each layer):} \quad c_{f,\ell} = (4\ell+1)^2, c_{u,\ell} = 2(4\ell-1)^2 \quad \ell = 1 : L-1, \end{aligned} \tag{7}$$

and $c_{u,L} = N + 2$, such that

$$\|f - \tilde{f}\|_{L^2(\Omega)} \lesssim N^{-\frac{1}{2}-\frac{3}{2d^2}}. \tag{8}$$

Main properties:

- For 2D images with or without multi-channel.
- Convolution (the most commonly used type): 3×3 kernel with multichannel for both periodic and zero padding.
- Achieve the same asymptotic approximation rate (for example, J. Xu, 2020, J. Siegel and J. Xu. 2021.) to ReLU DNN with one-hidden layer.
- Similar results hold for Classical CNNs, ResNet and pre-act ResNet networks.

Some remarks

Sketch of proof

1. Connection between DNNs with one hidden layer and CNNs with large kernel and multi-channel.
2. A decomposition theorem for convolution with large kernel

Theorem (H, Li and Xu, 2021)

Let $K \in \mathbb{R}^{1\times M\times(2k+1)\times(2k+1)}$, *then there exists a series of kernels* $S^\ell \in \mathbb{R}^{c_{\ell-1}\times c_\ell\times 3\times 3}$ *and* $P \in \mathbb{R}^{c_{\ell-1}\times M\times 3\times 3}$ *such that*

$$K * X = P * S^{k-1} * S^{k-2} * \cdots * S^1 * X, \quad \forall X \in \mathbb{R}^{d\times d}, \tag{9}$$

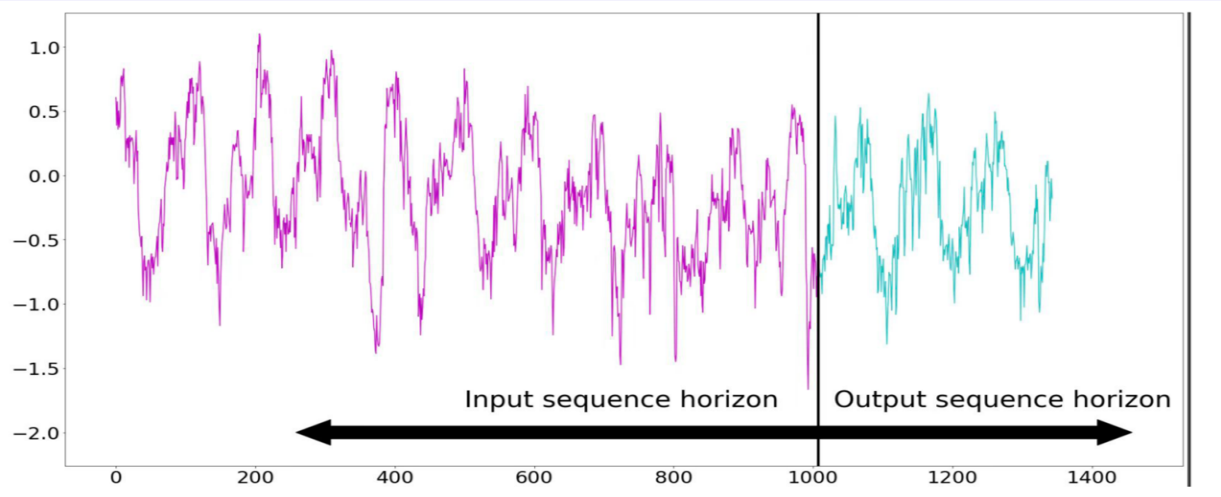
where $c_\ell = (2\ell+1)^2$ *for* $\ell = 1 : k-1$.

3. Rewrite CNNs with one hidden layer and large kernel as deep CNNs with 3×3 kernels (D.-X. Zhou, 2020).

Related works

- C. Bao, Q. Li, S. Zuo, C. Tai, L. Wu and X. Xiang, 2018 & D.-X. Zhou, 2018, 2020(a,b)
  - one-dimensional input, convolution $\leftrightarrow$ polynomial multiplication.
- K. Oono and T. Suzuki, 2019 & P. Petersen and F. Voigtlaender, 2020 & W. Kumagai and A. Sannai, 2020
  - special function classes, only for periodic padding.

Applications of MgNet in forecasting problems



Data sets:

- ETT: Two electricity transformers at two stations.
- Electricity: Electricity consumption of clients.
- Exchange: Current exchange of 8 countries.
- Traffic: Occupation rate of freeway system across the State of California.
- Weather: 21 meteorological indicators for a range of 1 year in Germany.
- ILIness: Influenza-like illness patients in the United States.

Math problem: Find (construct) a function

$$\mathcal{F} : \mathbb{R}^{d_i \times c} \mapsto \mathbb{R}^{d_o \times c}, \qquad \text{typically} \quad d_i = d_o.$$

One challenge: The memory and inference costs may increase fast as $d_i (d_o)$ increases.

Methodology: Fully connected V-cycle MgNet

Feature extraction and interpolation scheme (Zhu, He, Zhang and Xu 2022):

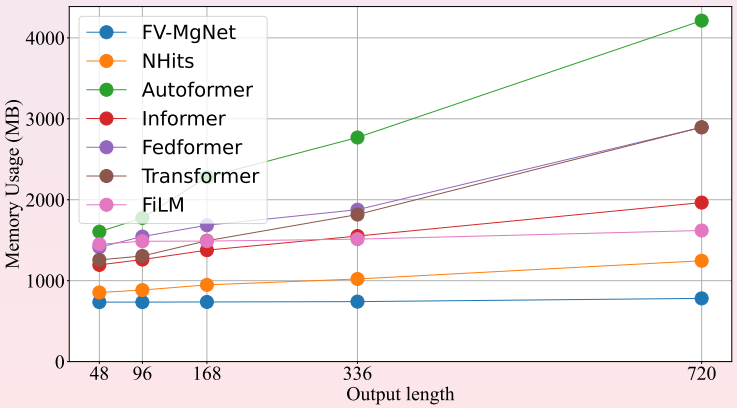
$$\text{Input: } f \xrightarrow[\text{constrained linear model: } Au=f]{\text{feature extraction}} \text{Feature: } u \xrightarrow[\text{neural network approximation}]{\text{feature interpolation: } \mathcal{I}} \text{Output: } y. \tag{10}$$

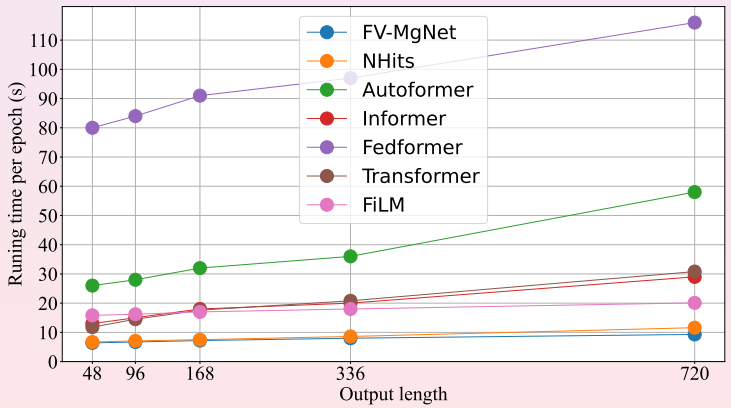
Feature extraction: A fully connected V-cycle type MgNet

$$u = \text{FV-MgNet}(f).$$

Feature interpolation: A one-hidden layer ReLU NN

$$y = \mathcal{I}(u) \approx W^2 \sigma \left(W^1 u \right).$$

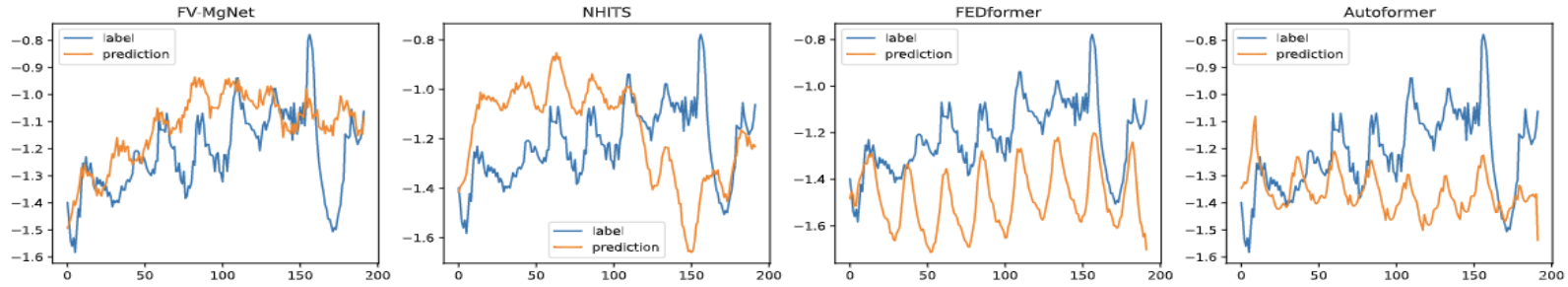




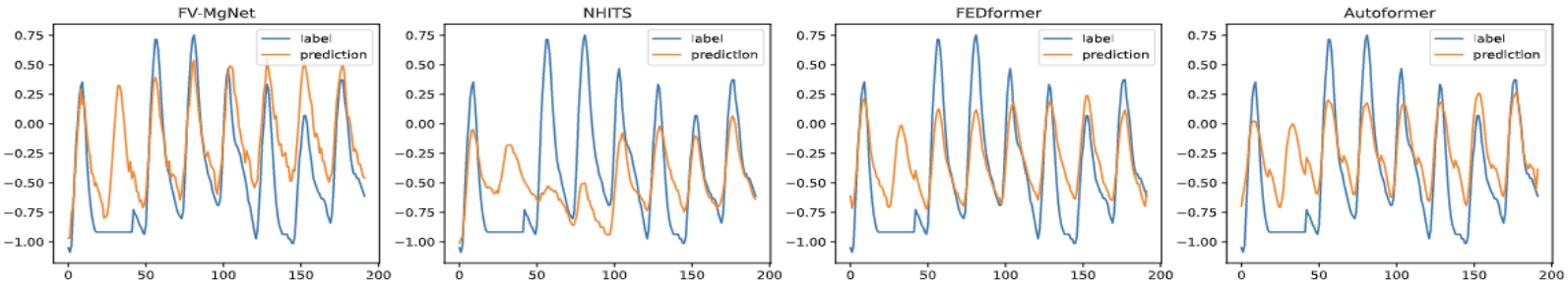
Accuracy of FV-MgNet

Methods		FV-MgNet		FiLM		N-HiTS		ETSformer		FEDformer		Autoformer	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
$ETTm2$	96	0.173	**0.253**	**0.165**	0.256	0.176	0.255	0.189	0.280	0.203	0.287	0.255	0.339
	192	0.230	**0.296**	**0.222**	**0.296**	0.245	0.305	0.253	0.319	0.269	0.328	0.281	0.340
	336	0.279	**0.329**	**0.277**	0.333	0.295	0.346	0.314	0.357	0.325	0.366	0.339	0.372
	720	**0.367**	**0.385**	0.371	0.389	0.401	0.426	0.414	0.413	0.421	0.415	0.422	0.419
$Electricity$	96	**0.144**	**0.250**	0.154	0.267	0.147	0.249	0.187	0.304	0.183	0.297	0.201	0.317
	192	**0.163**	0.262	0.164	**0.258**	0.167	0.269	0.199	0.315	0.195	0.308	0.222	0.334
	336	**0.176**	**0.276**	0.188	0.283	0.186	0.290	0.212	0.329	0.212	0.313	0.231	0.338
	720	**0.212**	**0.308**	0.249	0.338	0.243	0.340	0.233	0.345	0.231	0.343	0.254	0.361
$Exchange$	96	0.082	0.206	**0.079**	**0.204**	0.092	0.211	0.085	0.204	0.139	0.276	0.197	0.323
	192	0.184	0.314	**0.159**	**0.292**	0.208	0.322	0.182	0.303	0.256	0.369	0.300	0.369
	336	0.307	0.416	**0.270**	**0.398**	0.371	0.443	0.348	0.428	0.426	0.464	0.509	0.524
	720	**0.554**	**0.582**	0.830	0.721	0.888	0.723	1.025	0.774	1.090	0.800	1.447	0.941
$Traffic$	96	**0.396**	**0.285**	0.416	0.294	0.402	0.282	0.607	0.392	0.562	0.349	0.613	0.388
	192	0.417	0.295	**0.408**	**0.288**	0.420	0.297	0.621	0.399	0.562	0.346	0.616	0.382
	336	0.436	0.302	**0.425**	**0.298**	0.448	0.313	0.622	0.396	0.570	0.323	0.622	0.337
	720	**0.468**	**0.315**	0.520	0.353	0.539	0.353	0.632	0.396	0.596	0.368	0.660	0.408
$Weather$	96	**0.155**	**0.196**	0.199	0.262	0.158	0.195	0.197	0.281	0.217	0.296	0.266	0.336
	192	**0.201**	**0.239**	0.228	0.288	0.211	0.247	0.237	0.312	0.276	0.336	0.307	0.367
	336	**0.244**	**0.279**	0.267	0.323	0.274	0.300	0.298	0.353	0.339	0.380	0.359	0.395
	720	**0.313**	**0.329**	0.319	0.361	0.351	0.353	0.352	0.388	0.403	0.428	0.419	0.428
ILI	24	**1.647**	**0.764**	1.970	0.875	1.862	0.869	2.527	1.020	2.203	0.963	3.483	1.287
	36	**1.841**	**0.839**	1.982	0.859	2.071	0.969	2.615	1.007	2.272	0.976	3.103	1.148
	48	**1.831**	**0.853**	1.868	0.896	2.346	1.042	2.359	0.972	2.209	0.981	2.669	1.085
	60	**1.765**	**0.814**	2.057	0.929	2.560	1.073	2.487	1.016	2.545	1.061	2.770	1.125

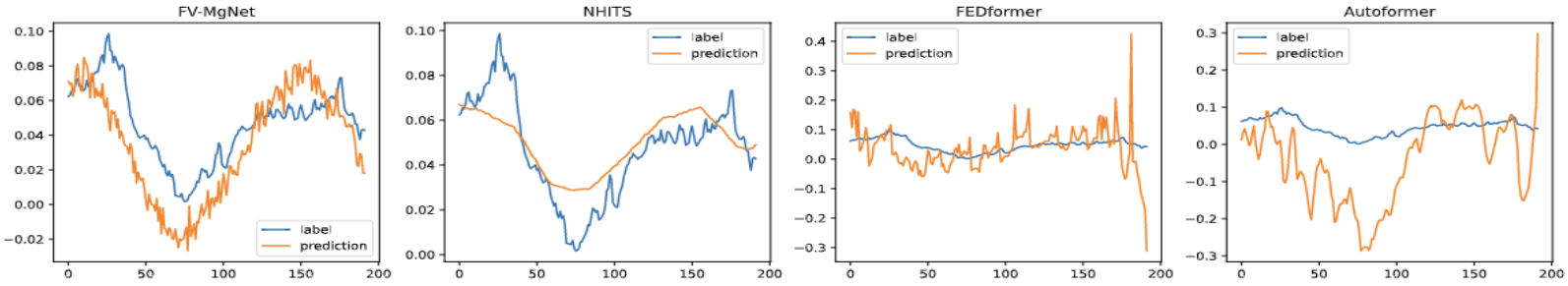
More observations: Capturing high frequencies ?



(a) $ETTh1$



(b) $ETTh2$



(c) $Weather$

Applications of MgNet in numerical PDEs

Problem:

$$\mathcal{L}(u; a) = f$$

Motivation:

- For linear operator: given f find u when $\mathcal{L}$ is linear and elliptic

$$\text{Multigrid} \approx (\mathcal{L})^{-1}.$$

- For nonlinear operator: given a find u or $\mathcal{L}$ is nonlinear

$$\text{multilevel representation: } u = \sum_{\ell=1}^{\infty} (I_\ell - I_{\ell-1})\, u$$

V-cycle MgNet for operator learning

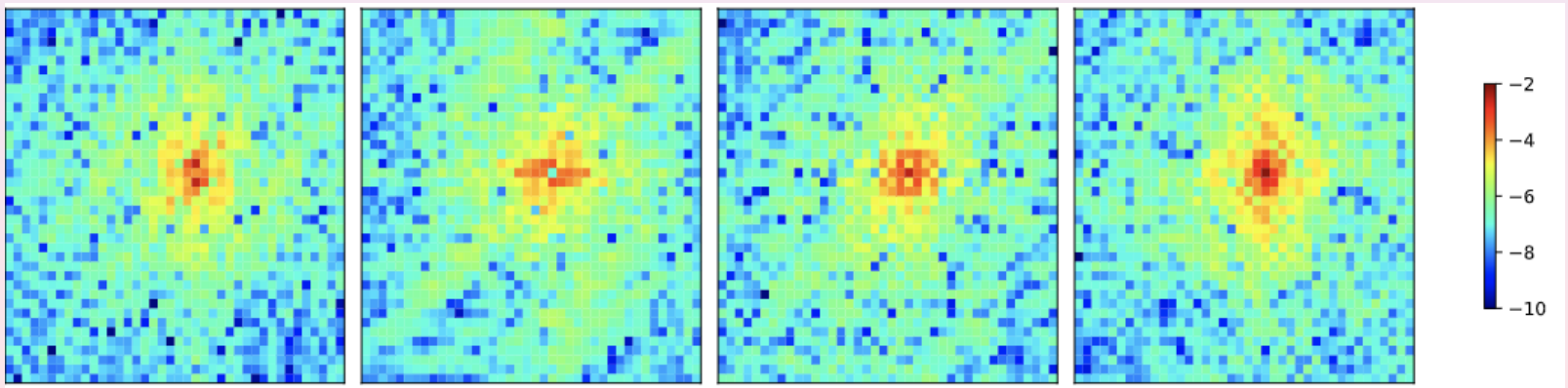
Example: $a(x) \mapsto u(x)$ with fixed $f(x) = 1$ for Darcy flow:

$$\begin{cases} -\nabla \cdot (a(x)\nabla u(x)) = f(x) & x \in (0,1)^2 \\ u(x) = 0 & x \in \partial(0,1)^2 \end{cases}$$

Naive idea:

$$\text{Convolutional V-cycle MgNet (V-MgNet)} \Rightarrow u \approx \text{V-MgNet}(a)$$

Error in Fourier domain (with four random examples):



An enhanced V-MgNet for operator learning

Smoother in original V-MgNet:

$$u^{\ell,i} = u^{\ell,i-1} + \sigma \circ B^{\ell,i} * \sigma \left(f^{\ell} - A^{\ell} * u^{\ell,i-1} \right)$$

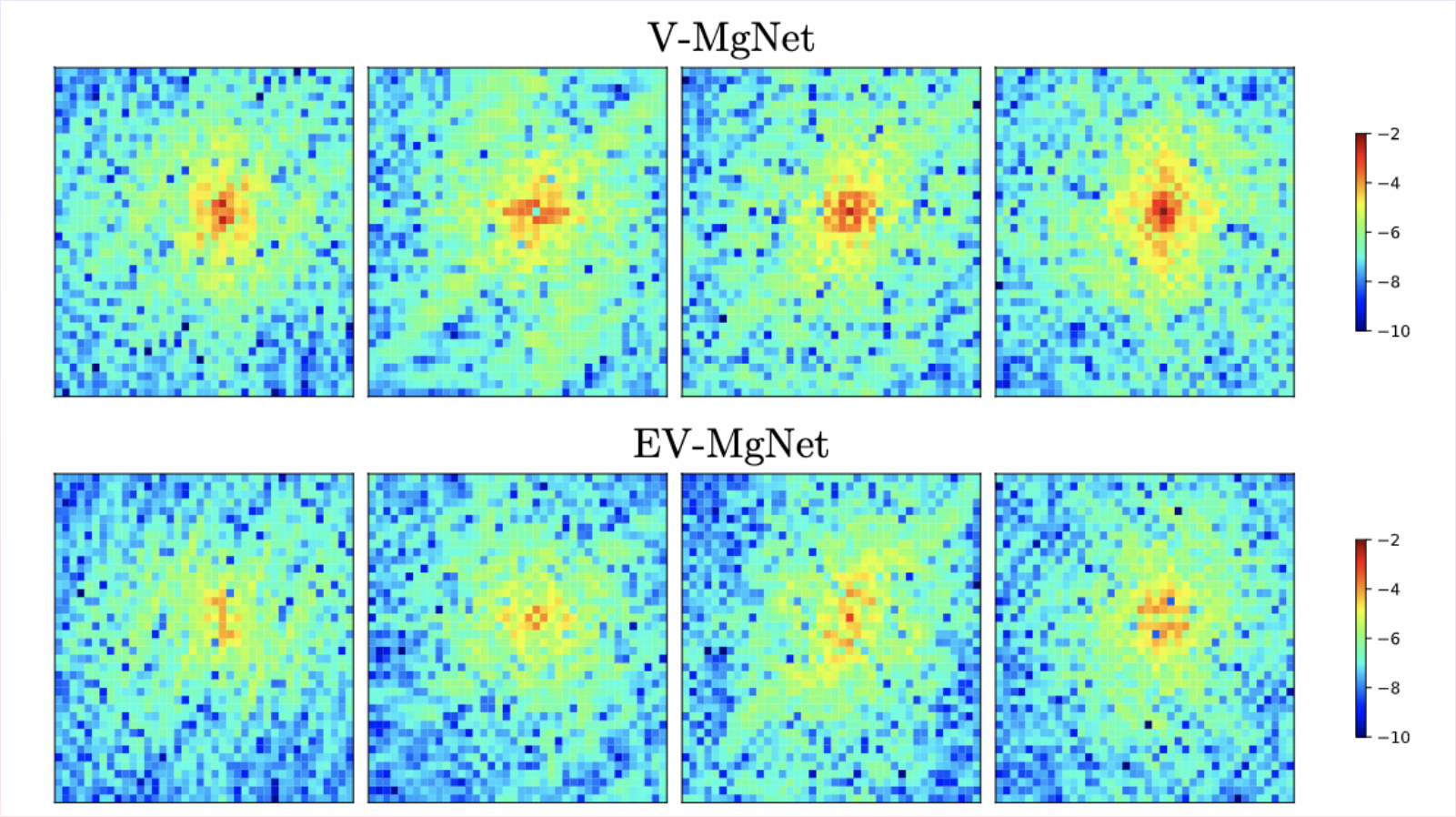
Enhanced V-MgNet (EV-MgNet) by correcting low frequencies explicitly:

$$u^{\ell,i} = u^{\ell,i-1} + \sigma \circ B^{\ell,i} * \sigma \left(f^{\ell} - A^{\ell} * u^{\ell,i-1} \right) + \sigma \circ \mathcal{F}^{-1} \mathcal{W}^{\ell,i} \mathcal{F} \left(f^{\ell} - A^{\ell} * u^{\ell,i-1} \right)$$

- $\mathcal{F}$: Fourier transform
- $W^{\ell,i}$: a linear operator defined only on coefficients of low frequencies.

Ref: Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, A. Anandkumar 2021; J. Zhu, J. He and Q. Huang. 2023.

Error after the low-frequency correction



Numerical example: 2D Darcy flow

Problem: Approximate the operator $a \mapsto u$ with $f = 1$.

$$\begin{cases} -\nabla \cdot (a(x)\nabla u(x)) = f(x), & x \in (0,1)^2, \\ u(x) = 0, & x \in \partial(0,1)^2. \end{cases}$$

Model	85×85	141×141	211×211	421×421
GNO	3.46×10^{-2}	3.32×10^{-2}	3.42×10^{-2}	3.69×10^{-2}
LNO	5.20×10^{-2}	4.61×10^{-2}	4.45×10^{-2}	-
MGNO	4.16×10^{-2}	4.28×10^{-2}	4.28×10^{-2}	4.20×10^{-2}
FNO	1.08×10^{-2}	1.09×10^{-2}	1.09×10^{-2}	0.98×10^{-2}
GT	8.51×10^{-3}	8.40×10^{-3}	8.50×10^{-3}	8.93×10^{-3}
MWT Leg	8.54×10^{-3}	7.32×10^{-3}	7.27×10^{-3}	6.86×10^{-3}
MWT Cheb	9.43×10^{-3}	8.28×10^{-3}	8.83×10^{-3}	8.74×10^{-3}
V-MgNet	5.07×10^{-3}	1.07×10^{-2}	2.63×10^{-2}	5.16×10^{-2}
EV-MgNet	4.57×10^{-3}	3.79×10^{-3}	3.79×10^{-3}	3.83×10^{-3}

More results: 1D Burgers' equation, 1D KDV equation, and 2D Navier-Stokes equations.

Ref: J. Zhu, J. He and Q. Huang. 2023.

Operator learning with different resolutions

Motivation: multigrid with convolutional form is independent from mesh size of input.

Practical issue: training with lower cost.

Example of 1D Burgers' equation: Approximate the operator $a \mapsto u$.

$$\begin{cases} \partial_t u(x,t) + \partial_x \left(u^2(x,t)/2\right) = \nu \partial_{xx} u(x,t), & x \in (0,1), t \in (0,1], \\ u(x,0) = a(x), \quad x \in (0,1), \\ u(0,t) = u(1,t), \quad t > 0. \end{cases}$$

Table: EV-MgNet model trained at low resolutions and predicted at high resolutions

Train \ Test	2048	4096	8192
128	3.99×10^{-3}	4.11×10^{-3}	4.19×10^{-3}
256	8.93×10^{-4}	9.26×10^{-4}	9.39×10^{-4}
512	6.46×10^{-4}	6.56×10^{-4}	6.62×10^{-4}

Conclusions and future works

- Multigrid and convolutional neural networks
  - MgNet;
  - data-feature mapping;
  - approximation;
  - applications:
    - ★ forecasting problems;
    - ★ operator learning.
- Future works:
  1. involve more properties of MG;
  2. apply CNN to MG;
  3. graphic CNNs;
  4. ...

THANK YOU!