Condensing CNNs with Partial Differential Equations

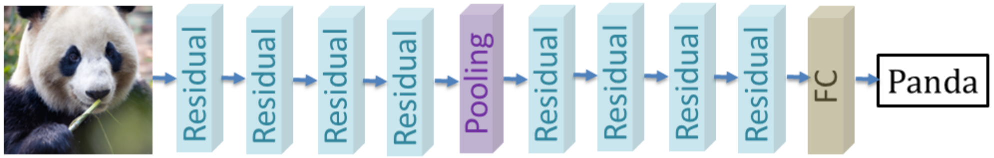
Anil Kag, Venkatesh Saligrama
Department of Electrical and Computer Engineering, Boston University
{anilkag, srv}@bu.edu

(2022)

Main Message

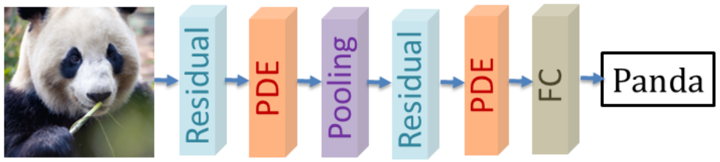
Discrete CNNs (ResNet, EfficientNet)

➢ *Repetitions are expensive : compute & storage*



Our Proposal : Global Layer based on PDEs

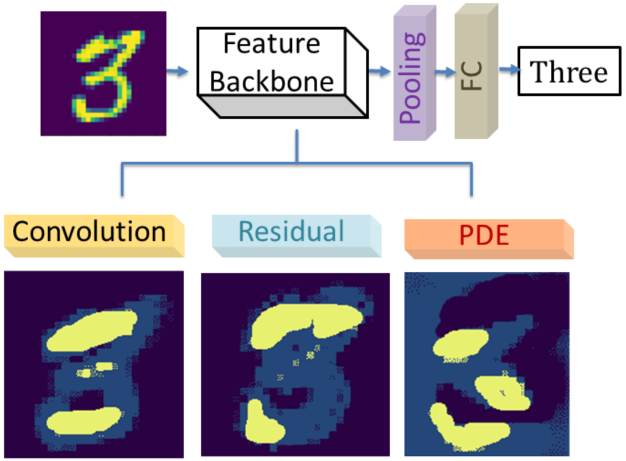
➢ *Apply PDE instead of repetitions*



- ❑ *Shallow Networks*
- ❑ *Low Storage & Compute*
- ❑ *Integrable in any network*

Comparison on learned representation

- *Visualization : MNIST-10*



Feature Backbone	Accuracy (%)	Confusion b/w 3 & 5
Convolution	92.01	27
Residual	92.53	30
Global	**95.03**	**9**

- *Baseline Comparison : CIFAR-10*

Architecture	Accuracy (%)	#Params	#MACs	Inference Time (s)	Depth
ResNet-32	92.49	460K	70M	4.5	15
MDEQ	92.28	1.1M	1.5B	23.3	-
ResNet-Global	91.93	162K	15M	1.9	6

Bold Numbers

Empirical Evaluation Highlights

- *Low Model Footprint @ Similar Accuracy*

Architecture	CIFAR-10 Acc. (%)	CIFAR-100 Acc. (%)	#Params	#MACs
NeuPDE MDEQ	95.4 93.8	76.4 71.1	9M 10M	4.1B 8.3B
ResNet-32 ResNet-Global	92.4 91.9	68.6 68.1	473K **168K**	70M **15M**
DenseNet DenseNet-Global	95.3 95.0	77.2 75.7	800K **481K**	297M **136M**
DARTS DARTS-Global	97.1 96.9	82.5 81.9	3.4M **835K**	539M **213M**

- *High Accuracy @ Similar Model Footprint*

Architecture	CIFAR-100 Acc. (%)	#Params	#MACs	Tiny Models: CIFAR-100 Acc. (%)	Tiny Models: #Params	Tiny Models: #MACs
ResNet-56 ResNet-Global	70.4 **74.3**	861K 1.3M	127M 119M	35.2 **43.6**	14K 16K	3.4M 3.6M
DenseNet DenseNet-Global	77.2 **78.9**	800K 922K	297M 247M	- -	- -	- -
Wide-ResNet Wide-ResNet-Global	79.1 **80.5**	9M 9M	1.3B 1.3B	39 **50.2**	23K 24K	9.8M 8.7M
DARTS DARTS-Global	82.5 **84.2**	3.4M 2.4M	539M 519M	54.6 **60.7**	43K 41K	7.7M 8.2M

- *Scales easily to Imagenet-1K*

Architecture	Top-1 Acc. (%)	#Params	#MACs
MDEQ-Small	75.5	18M	21B
MBV2 MBV2-Global	71.9 71.6	3.4M **1.6M**	300M **193M**
MBV3 MBV3-Global	75.2 74.1	5.4M **3M**	219M **156M**
EfficientNet-B0 EfficientNet-B0-Global	77.1 76.1	5.3M **2.4M**	390M **244M**

PDE: advection-diffusion

(a) PDE: At the heart of the Global feature layer is the following generic advection-diffusion PDE [1]

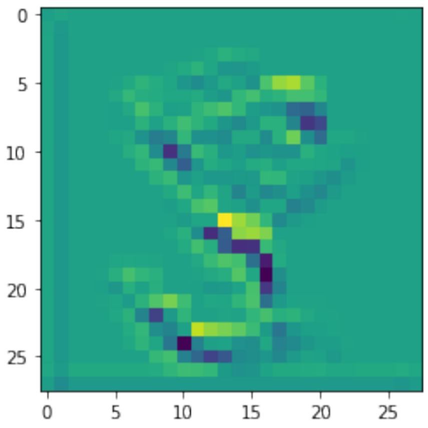
$$\frac{\partial}{\partial t}H(x,y,t) + \frac{\partial}{\partial x}\big(u(x,y,t)H(x,y,t)\big) + \frac{\partial}{\partial y}\big(v(x,y,t)H(x,y,t)\big)$$

$$= \frac{\partial}{\partial x}\left(D_x \frac{\partial}{\partial x}H(x,y,t)\right) + \frac{\partial}{\partial y}\left(D_y \frac{\partial}{\partial y}H(x,y,t)\right) + f(I(x,y))$$

t, x, y: time and spatial coordinates
H(x,y, t): image evolving in time
H(x,y, 0): initial guess for learned representation to provide for PDE solver, (can be input image or a function of it)

Free parameters in PDE:
(a) function f (b) particle velocity (u, v), and (c) diffusion coefficient (D_x , D_y). (d) boundary condition

Physical meaning: consider image as a concentration distribution of particles inside a fluid with a velocity field. Movements in fluid change the image and result in new representations. Aim is to learn these movements inside the fluid.



Learned velocity field

Why it works better

- No clear answer in the paper
- My understanding: instead of learning multiple convolution kernels that work locally the method learns two matrices with the same size of the image (velocity field).
- Novelty: introduces a new way to concentrate information and exploits dynamics of PDEs for constraining the feature maps.

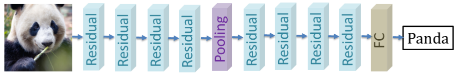
More Details:
https://www.youtube.com/watch?v=6mNl2XFAjcQ&ab_channel=AnilKag
https://openaccess.thecvf.com/content/CVPR2022/html/Kag_Condensing_CNNs_With_Partial_Differential_Equations_CVPR_2022_paper.html

Condensing CNNs with Partial Differential Equations

Anil Kag and Venkatesh Saligrama (ECE Department, Boston University)

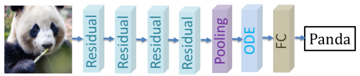
Discrete CNNs (ResNet, EfficientNet)

- *Repetitions are expensive : compute & storage*



ODE CNNs (Neural-ODE, NeuPDE)

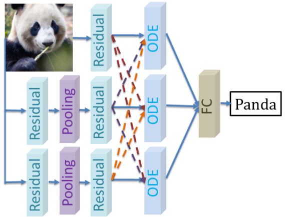
- *Does not scale to large scale datasets*



MNIST	Acc. (%)	#Params	#MACs
Neural-ODE	99.49	220K	100M
NeuPDE	99.49	180K	50M
ResNet	99.51	600K	30M

ODE CNNs (MDEQ)

- *Scales up to Imagenet, **But** computationally expensive*



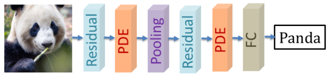
Imagenet-1K	Acc. (%)	#Params	#MACs
ResNet	77.8	60M	11B
DenseNet	76.3	14M	3.5B
MDEQ-Small	75.5	18M	21B

Feature Comparison with Prior Work

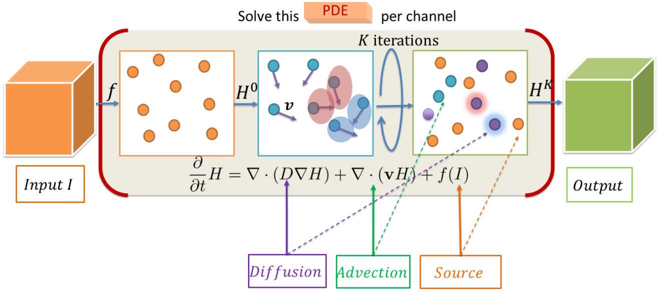
Architecture Type	Architecture	Storage Reduction	Compute Reduction	Large Scale Datasets
Discrete-CNNs	ResNet	1x	1x	✓
ODE-CNNs	Neural-ODE	✓	✗	✗
	NeuPDE	✓	✗	✗
	MDEQ	✓	✗	✓
Global-CNNs (ours)	ResNet-Global	✓	✓	✓

Our Proposal : Global Layer based on PDEs

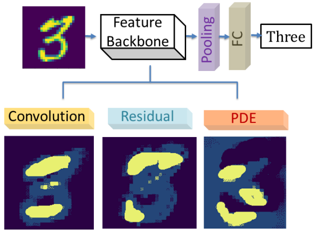
- *Apply PDE instead of repetitions*



- *Shallow Networks*
- *Low Storage & Compute*
- *Integrable in any network*



- *Visualization : MNIST-10*



Feature Backbone	Accuracy (%)	Confusion b/w 3 & 5
Convolution	92.01	27
Residual	92.53	30
Global	**95.03**	**9**

- *Baseline Comparison : CIFAR-10*

Architecture	Accuracy (%)	#Params	#MACs	Inference Time (s)	Depth
ResNet-32	92.49	460K	70M	4.5	15
MDEQ	92.28	1.1M	1.5B	23.3	-
ResNet-Global	91.93	162K	15M	1.9	6

Empirical Evaluation Highlights

- *Low Model Footprint @ Similar Accuracy*

Architecture	CIFAR-10 Acc. (%)	CIFAR-100 Acc. (%)	#Params	#MACs
NeuPDE	95.4	76.4	9M	4.1B
MDEQ	93.8	71.1	10M	8.3B
ResNet-32	92.4	68.6	473K	70M
ResNet-Global	91.9	68.1	**168K**	**15M**
DenseNet	95.3	77.2	800K	297M
DenseNet-Global	95.0	75.7	**481K**	**136M**
DARTS	97.1	82.5	3.4M	539M
DARTS-Global	96.9	81.9	**835K**	**213M**

- *High Accuracy @ Similar Model Footprint*

Architecture	CIFAR-100 Acc. (%)	#Params	#MACs	Tiny Models		
				CIFAR-100 Acc. (%)	#Params	#MACs
ResNet-56	70.4	861K	127M	35.2	14K	3.4M
ResNet-Global	**74.3**	1.3M	119M	**43.6**	16K	3.6M
DenseNet	77.2	800K	297M	-	-	-
DenseNet-Global	**78.9**	922K	247M	-	-	-
Wide-ResNet	79.1	9M	1.3B	39	23K	9.8M
Wide-ResNet-Global	**80.5**	9M	1.3B	**50.2**	24K	8.7M
DARTS	82.5	3.4M	539M	54.6	43K	7.7M
DARTS-Global	**84.2**	2.4M	519M	**60.7**	41K	8.2M

- *Scales easily to Imagenet-1K*

Architecture	Top-1 Acc. (%)	#Params	#MACs
MDEQ-Small	75.5	18M	21B
MBV2	71.9	3.4M	300M
MBV2-Global	71.6	**1.6M**	**193M**
MBV3	75.2	5.4M	219M
MBV3-Global	74.1	**3M**	**156M**
EfficientNet-B0	77.1	5.3M	390M
EfficientNet-B0-Global	76.1	**2.4M**	**244M**

- https://github.com/anilkagak2/PDE_GlobalLayer