JUNE 18-22, 2023
CVPR
VANCOUVER, CANADA

Poster ID# **TUE-PM-201**

Learning Partial Correlation based Deep Visual Representation for Image Classification


Saimunur Rahman
CSIRO Data61, UOW


Piotr Koniusz
CSIRO Data61, ANU

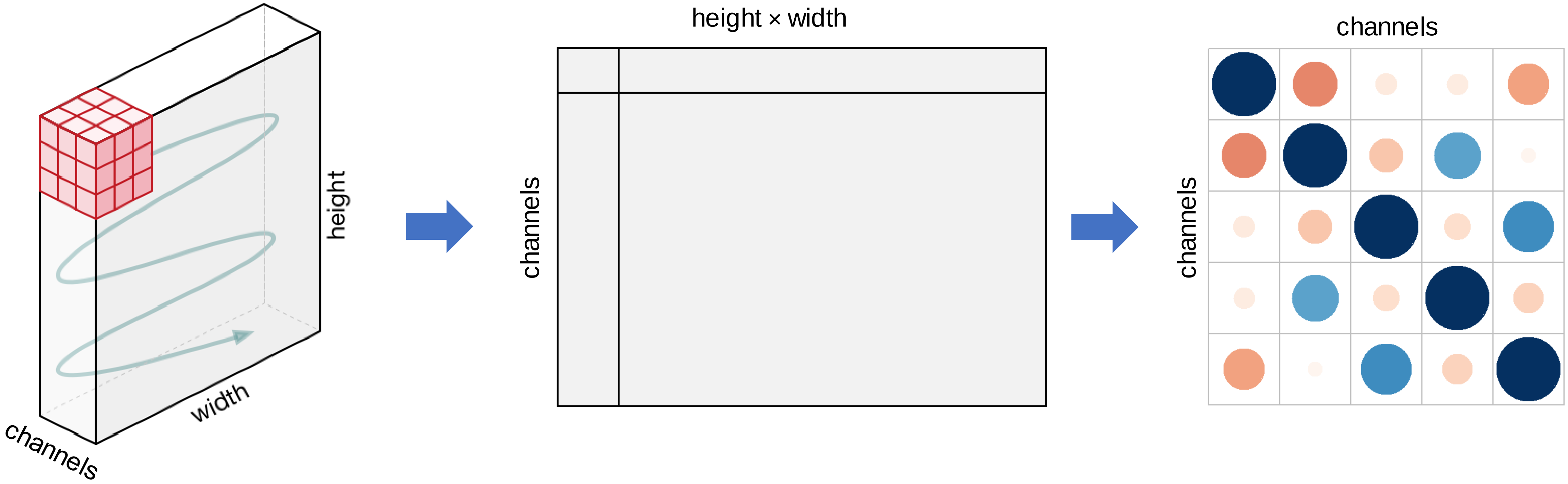

Lei Wang
UOW


Luping Zhou
USyD


Peyman Moghadam
CSIRO Data61

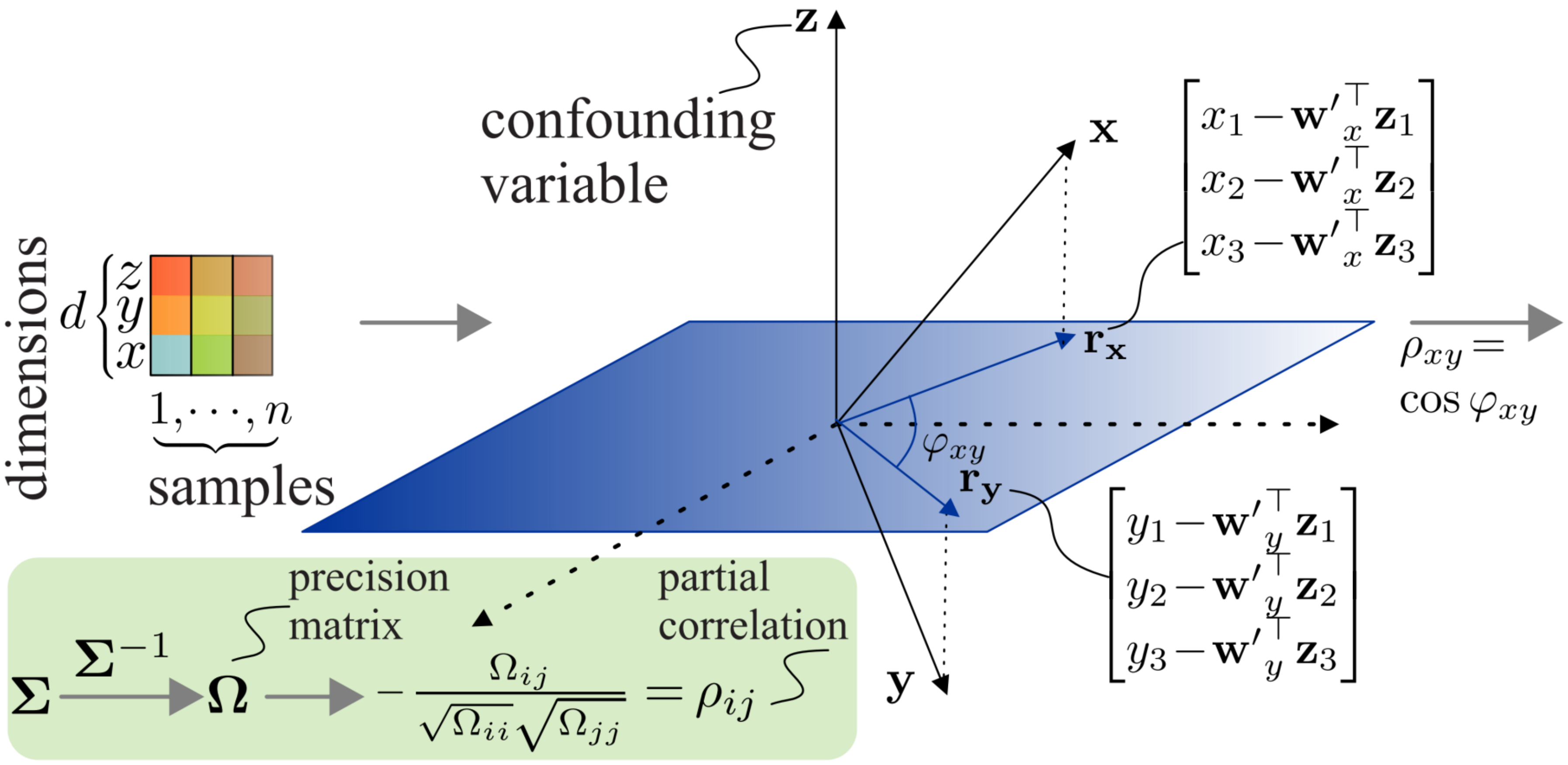

Changming Sun
CSIRO Data61

Pairwise Correlation Based Representation

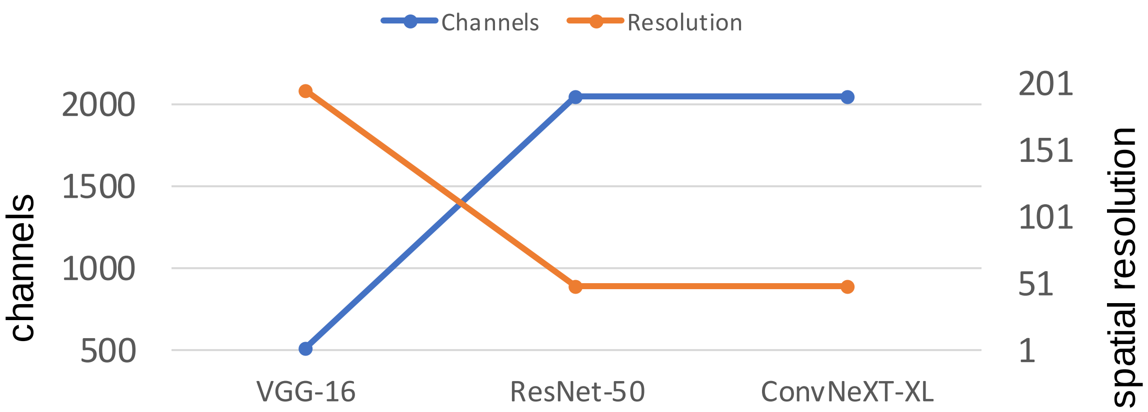


Example: covariance matrix based visual representation.

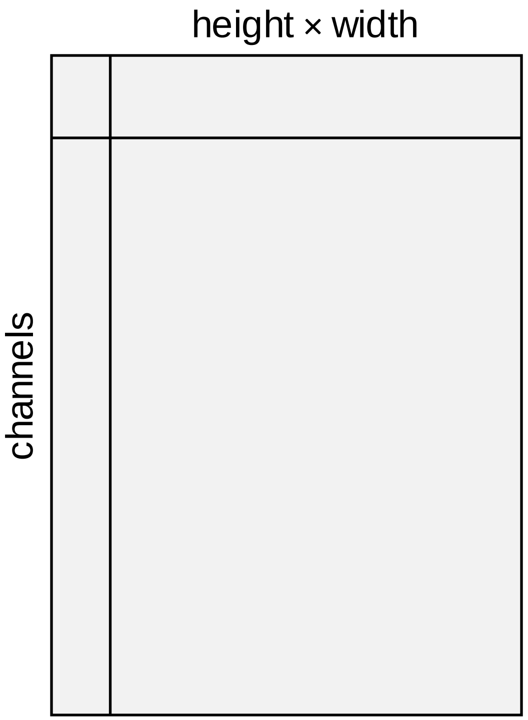
Confounding Effect in Pairwise Correlation



Partial Correlation Estimation from CNN



Number of channels *vs.* their resolutions in recent CNNs.



$$\rightarrow \Sigma \rightarrow \Sigma^{-1}$$

matrix inverse

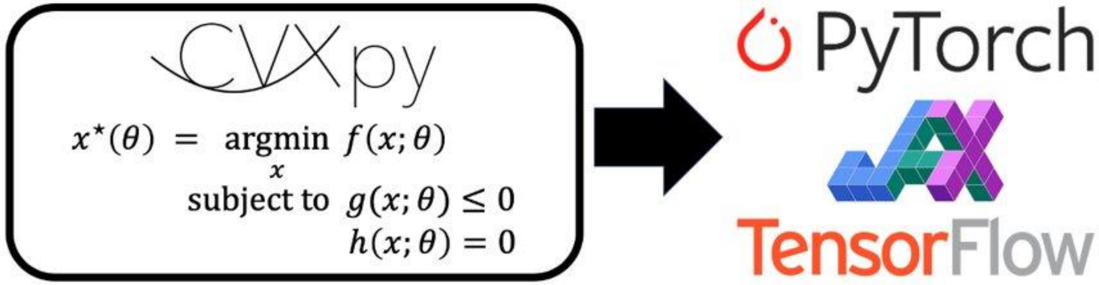
channels **>>** height × width

Sparse Inverse Covariance Estimate (SICE)

SICE is defined as follows:

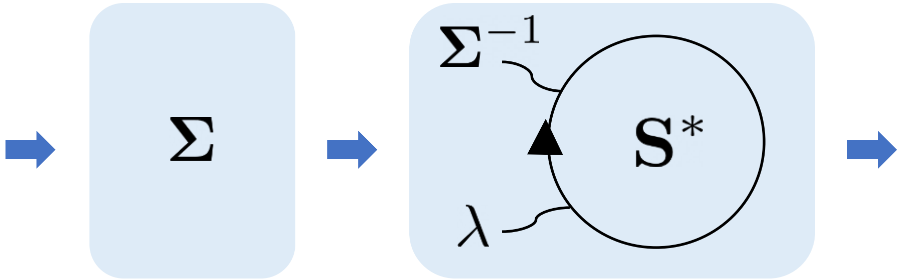
$$\mathbf{S}^* = \underset{S \succ 0}{\arg\max}\ \operatorname{logdet}(\mathbf{S}) - \operatorname{trace}(\mathbf{\Sigma S}) - \lambda \|\mathbf{S}\|_1$$

where $\mathbf{\Sigma}$ is a sample-based covariance matrix, and $\det(\cdot)$, $\operatorname{trace}(\cdot)$ and $\|\bullet\|_1$ denote the determinant, trace and $\ell 1$-norm of a vectorization of matrix, respectively.

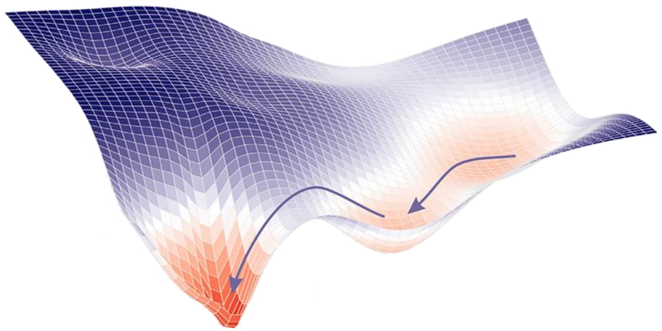


Proposed Iterative SICE (iSICE)

$$\frac{\partial J}{\partial \mathbf{S}} = \frac{\partial}{\partial \mathbf{S}} \log\ \det(\mathbf{S}) - \frac{\partial}{\partial \mathbf{S}} \mathrm{trace}(\mathbf{\Sigma}\mathbf{S}) - \lambda \frac{\partial}{\partial \mathbf{S}} \|\mathbf{S}\|_1$$
$$= \mathbf{S}^{-1} - \mathbf{\Sigma} - \lambda\Big(\frac{\partial}{\partial \mathbf{S}}\mathbf{S}^{+} - \frac{\partial}{\partial \mathbf{S}}\mathbf{S}^{-}\Big)$$
$$= \mathbf{S}^{-1} - \mathbf{\Sigma} - \lambda\big(\mathrm{sign}(\mathbf{S}^{+}) - \mathrm{sign}(\mathbf{S}^{-})\big),$$



iSICE module.

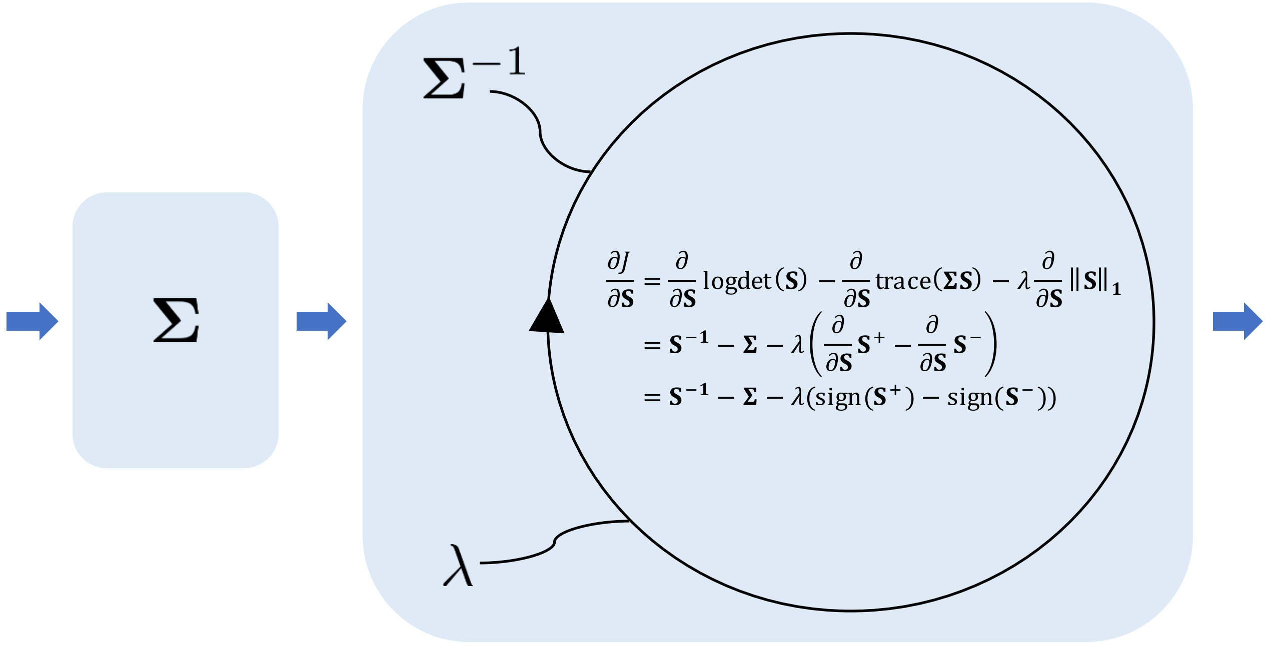


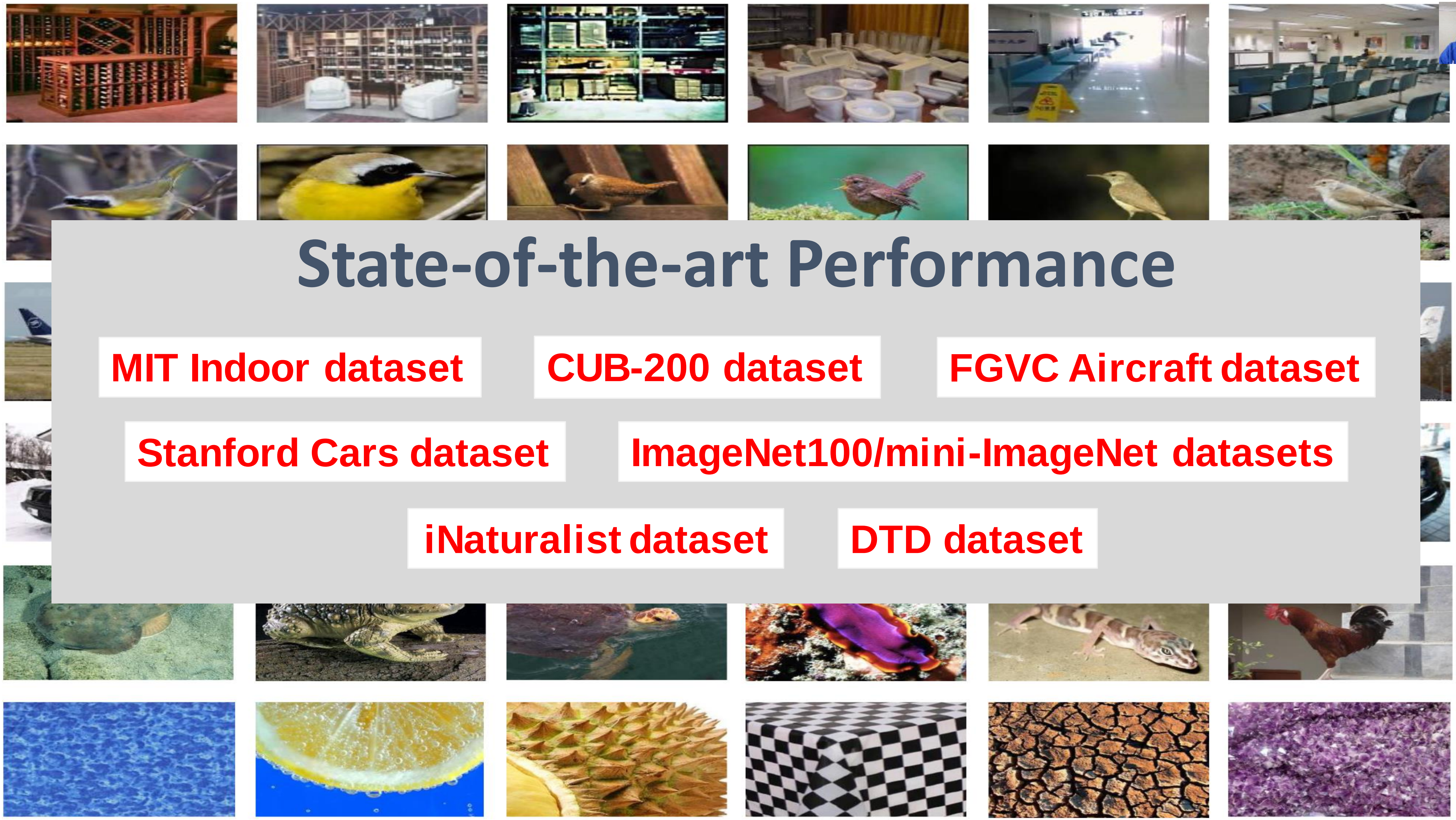
Stochastic gradient descend (SGD) based optimisation.

PyTorch
JAX
TensorFlow

JUNE 18-22, 2023
CVPR
VANCOUVER, CANADA

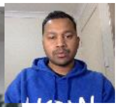
Partial Correlation Representaion



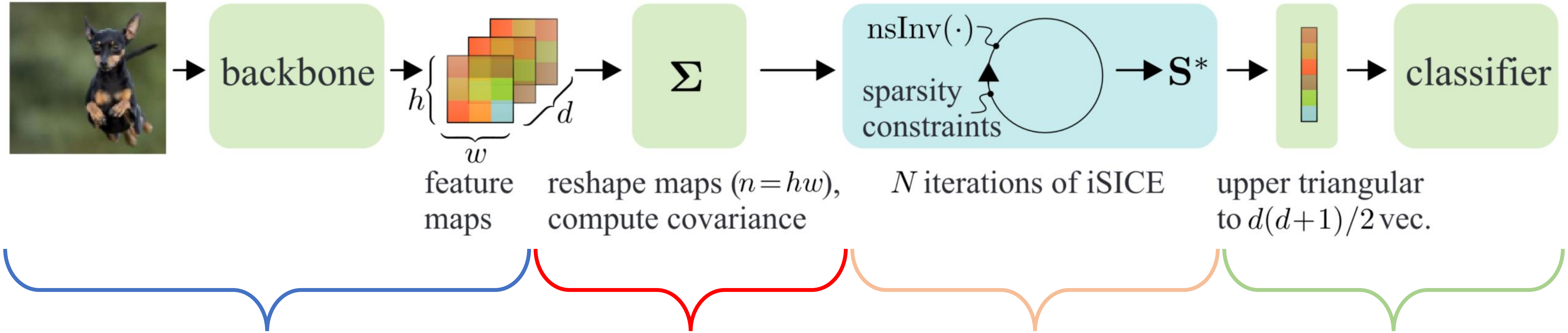


State-of-the-art Performance

- MIT Indoor dataset
- CUB-200 dataset
- FGVC Aircraft dataset
- Stanford Cars dataset
- ImageNet100/mini-ImageNet datasets
- iNaturalist dataset
- DTD dataset



iSICE Integration With CNN



Feature extraction layers

Sample based covariance

Proposed iSICE

Routine layers

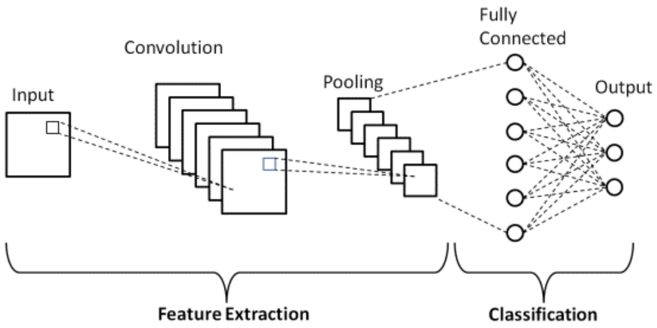
Three important iSICE paraments 1) **Sparsity** 2) **Number of iterations** and 3) **Learning rate.**

Implementation Details

Implementation Library

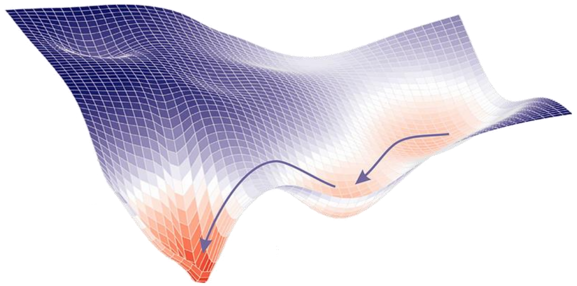
PyTorch

Network Architectures



- VGG-16/19
- ResNet-50/101
- ResNeXt-50/101
- ConvNeXt-T/B
- Swin-T/B

Optimiser



AdamW

Evaluation Datasets

Dataset	Total classes	Total images	Predefined protocol		Major difficulty
			Training images	Testing images	
MIT Indoor	67	6,700	5,360	1,340	difficult environment
Birds	200	11,788	5,994	5,794	subtle class difference
Aircraft	100	10,000	6,600	3,400	subtle class difference
Cars	196	16,185	8,144	8,041	cluttered background
DTD	47	5,640	4512	1128	complex structure
iNaturalist	5,089	675,170	579,184	95,986	class imbalance
mini-ImageNet	100	60,000	54,000	6,000	difficult environment
ImageNet100	100	1,35,000	130,000	5,000	difficult environment

Experimental Results

iSICE Hyper-parameter Selection, *e.g.*, sparsity.

Dataset	Backbone	iSQRT-COV [23]	Precision **Ω**	iSICE							
				Sparsity constant λ							Mean ± Std. (iSICE)
				1.0	0.5	0.1	0.01	0.001	0.0001	0.00001	
MIT	VGG-16	76.12	**80.15**	77.46	78.13	78.13	78.66	78.58	78.96	78.96	78.41±0.54
	ResNet-50	78.81	80.75	78.43	80.75	80.45	80.52	80.90	80.37	**81.34**	80.39±0.93
Airplane	VGG-16	90.01	89.44	92.26	92.71	92.77	92.23	**92.83**	92.74	92.44	92.56 ±0.25
	ResNet-50	90.88	91.15	**92.89**	92.65	**92.89**	92.74	92.83	92.56	92.56	92.73±0.14
Birds	VGG-16	84.47	83.36	86.04	86.47	86.35	**86.52**	85.59	86.31	86.28	86.22±0.32
	ResNet-50	84.26	84.67	84.62	85.16	85.30	85.90	**86.05**	85.90	85.59	85.50±0.51
Cars	VGG-16	91.21	92.04	93.60	93.98	**94.06**	94.03	93.88	93.91	93.50	93.85±0.22
	ResNet-50	92.13	91.99	93.01	93.36	93.69	93.51	93.22	**93.72**	93.40	93.41±0.25

JUNE 18-22, 2023 CVPR VANCOUVER, CANADA

Experimental Results

iSICE *vs.* its covariance counterparts on scene, fine-grained and generic image dataset.

Method	Matrix Dim.	MIT		Airplane		Birds		Cars		Average	
		VGG	ResNet	VGG	ResNet	VGG	ResNet	VGG	ResNet	VGG	ResNet
iSQRT-COV	256×256	76.1	78.8	90.0	90.9	84.5	84.3	91.2	92.1	85.5	86.5
	512×512	76.9	**82.8**	91.5	91.1	85.0	84.5	92.2	92.1	86.4	87.6
Precision $\boldsymbol{\Omega}$	256×256	**80.2**	**80.8**	89.4	91.2	83.4	84.7	92.0	92.0	86.3	87.1
	512×512	80.7	82.7	90.1	91.5	84.9	84.0	92.5	92.6	87.0	87.7
SICE	128×128	71.0	73.1	85.5	86.9	77.3	78.0	87.0	87.9	80.2	81.5
	256×256	73.7	75.4	87.9	89.2	79.7	80.3	89.5	89.3	82.7	83.6
iSICE	256×256	78.7	80.5	**92.2**	**92.7**	**86.5**	**85.9**	**94.0**	**93.5**	**87.9**	**88.2**
	512×512	**81.1**	81.7	**92.9**	**92.6**	**86.8**	**86.0**	**94.6**	**93.8**	**88.9**	**88.5**

ImageNet-100 dataset

Method	Backbone	Top-1	Top-5
GAP [13]	ResNet-50/ VGG-16	71.0/69.5	90.9/88.9
iSQRT-COV [23]		71.5/70.2	90.5/89.7
Precision $\boldsymbol{\Omega}$		71.1/71.0	90.1/90.1
iSICE		**74.8/73.4**	**92.0/91.8**

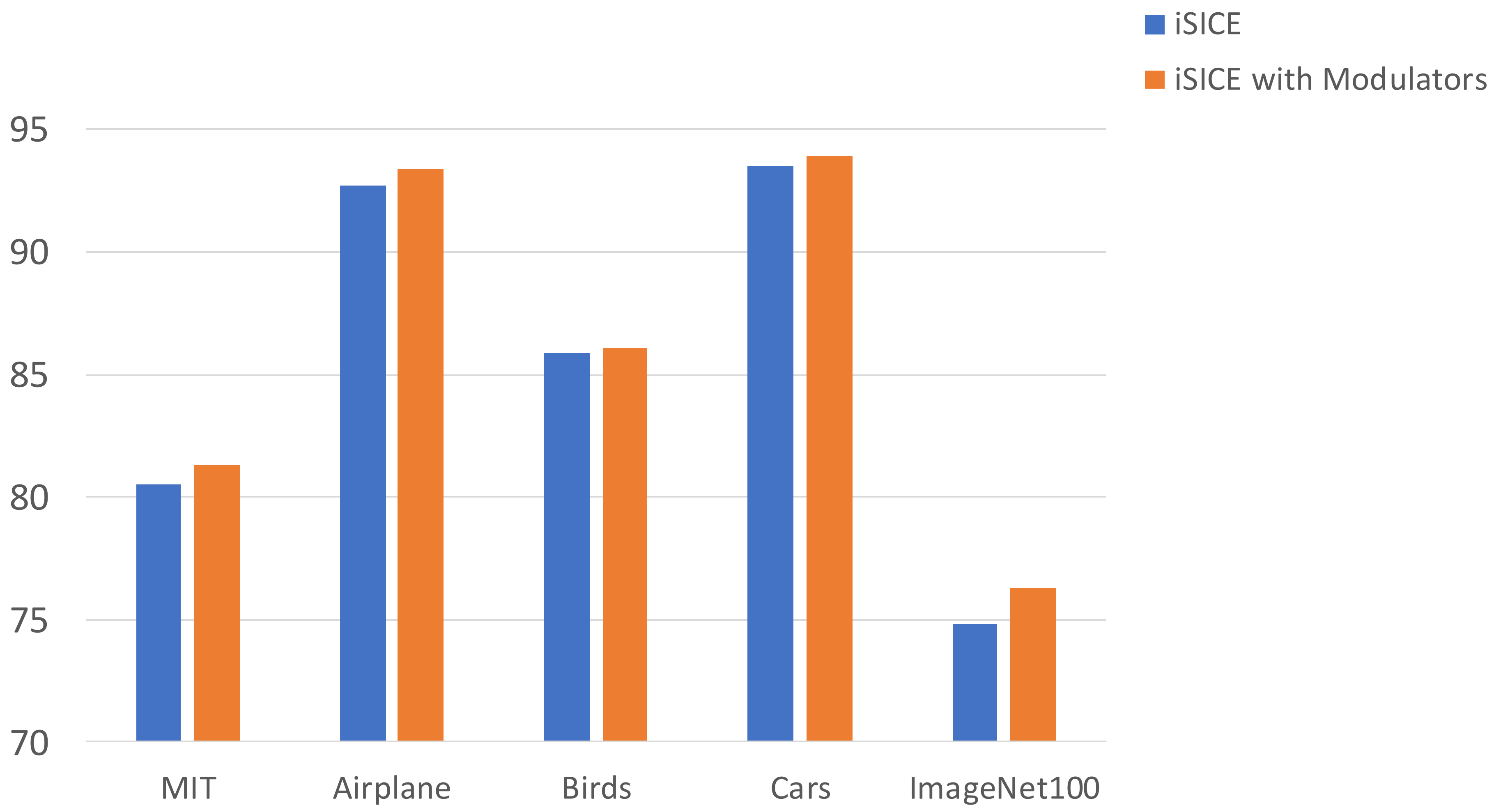
Experimental Results

iSICE *vs.* its covariance counterparts on various datasets and backbones.

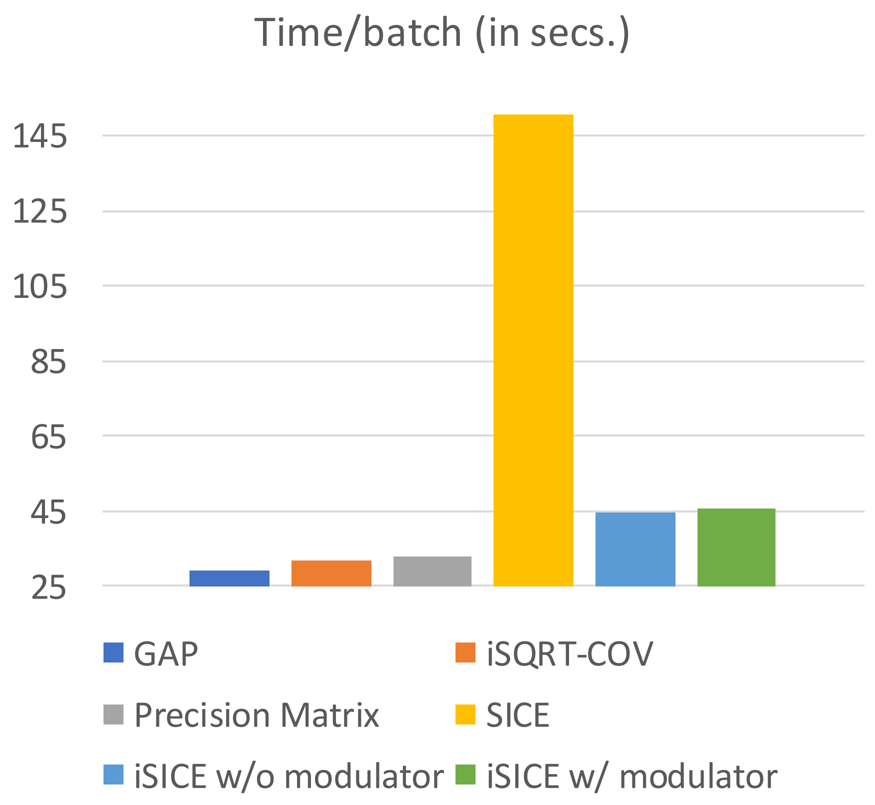
Method	Backbone	MIT	Airplane	Birds	Cars	DTD	iNatuarlist	mini-ImageNet
GAP [37]		–	76.6	70.4	79.8	–	–	–
NetVLAD [2]		–	81.8	81.6	88.6	–	–	–
NetFV [28]		–	79.0	79.9	86.2	–	–	–
BCNN [27]		77.6	83.9	84.0	90.6	70.6	–	–
CBP [11]		76.2	84.1	84.3	91.2	67.7	–	–
LRBP [17]		–	87.3	84.2	90.9	–	–	–
KP [6]		–	86.9	86.2	92.4	–	–	–
HIHCA [4]		–	88.3	85.3	91.7	–	–	–
Improved BCNN [25]		–	88.5	85.8	92.0	–	–	–
SMSO [46]	VGG-16	79.5	–	85.0	–	–	–	–
MPN-COV [43] (reproduced)		–	86.1	82.9	89.8	–	–	–
iSQRT-COV [23] (reproduced)		76.1	90.0	84.5	91.2	71.3	56.2	76.2
DeepCOV [9]		79.2	88.7	85.4	91.7	–	–	–
DeepKSPD [9]		**81.0**	90.0	84.8	91.6	–	–	–
RUN [47]		80.5	91.0	85.7	–	–	–	–
FCBN [48]		80.3	90.5	85.5	–	–	–	–
TKPF [49]		80.5	91.4	86.0	–	–	–	–
Precision Ω		80.2	89.4	83.4	92.0	74.0	57.9	81.0
iSICE (ours)		78.7	**92.2**	**86.5**	**94.0**	**74.7**	**59.8**	**78.7**
CBP [11]		–	81.6	81.6	88.6	–	–	–
KP [6]		–	85.7	84.7	91.1	–	–	–
SMSO [46]		79.7	–	85.8	–	–	–	–
iSQRT-COV [23] (reproduced)		78.8	90.9	84.3	92.1	73.0	57.7	70.7
DeepCOV-ResNet [34]	ResNet-50	83.4	83.9	**86.0**	85.0	–	–	–
TKPF [49]		**84.1**	92.1	85.7	–	–	–	–
Precision Ω		80.8	91.2	84.7	92.0	73.7	59.6	65.6
iSICE (ours)		80.5	**92.7**	85.9	**93.5**	**75.7**	**60.7**	**72.0**
iSQRT-COV [23]		76.3	90.3	84.1	91.4	71.8	56.9	75.4
Precision Ω	VGG-19	79.6	91.1	83.2	92.2	74.2	57.3	73.8
iSICE (ours)		**80.6**	**92.5**	**86.6**	**93.9**	**74.9**	**59.6**	77.1
iSQRT-COV [23]		79.3	91.0	84.4	92.3	73.0	70.6	73.9
Precision Ω	ResNet-101	77.9	90.1	83.3	91.4	71.2	69.8	73.0
iSICE (ours)		**81.0**	**92.9**	**86.6**	**93.6**	**75.4**	**72.0**	**78.0**
iSQRT-COV [23]		81.6	91.3	86.2	92.4	75.7	72.2	76.1
Precision Ω	ResNeXt-101	85.7	90.2	84.6	89.9	76.9	72.3	77.6
iSICE (ours)		**86.3**	**94.6**	**87.2**	**94.5**	**78.7**	**73.8**	**81.0**
iSQRT-COV [23]		77.8	88.1	83.5	89.4	84.7	61.5	82.0
Precision Ω	ConvNext-T	78.5	81.2	83.7	92.2	83.9	59.3	83.6
iSICE (ours)		**85.4**	**90.4**	**86.7**	**93.1**	**88.9**	**65.0**	**85.1**
iSQRT-COV [23]		82.1	87.6	85.1	89.7	86.1	58.1	67.7
Precision Ω	Swin-T	82.5	88.2	84.9	90.5	86.5	59.1	65.6
iSICE (ours)		**85.9**	**89.6**	**86.5**	**91.3**	**88.3**	**61.9**	**69.1**
iSQRT-COV [23]		86.6	91.3	88.0	92.0	79.4	69.7	64.9
Precision Ω	Swin-B	87.0	90.7	87.7	93.1	**80.1**	67.3	66.4
iSICE (ours)		**87.6**	**92.9**	**88.3**	**93.3**	79.8	**72.4**	**68.4**

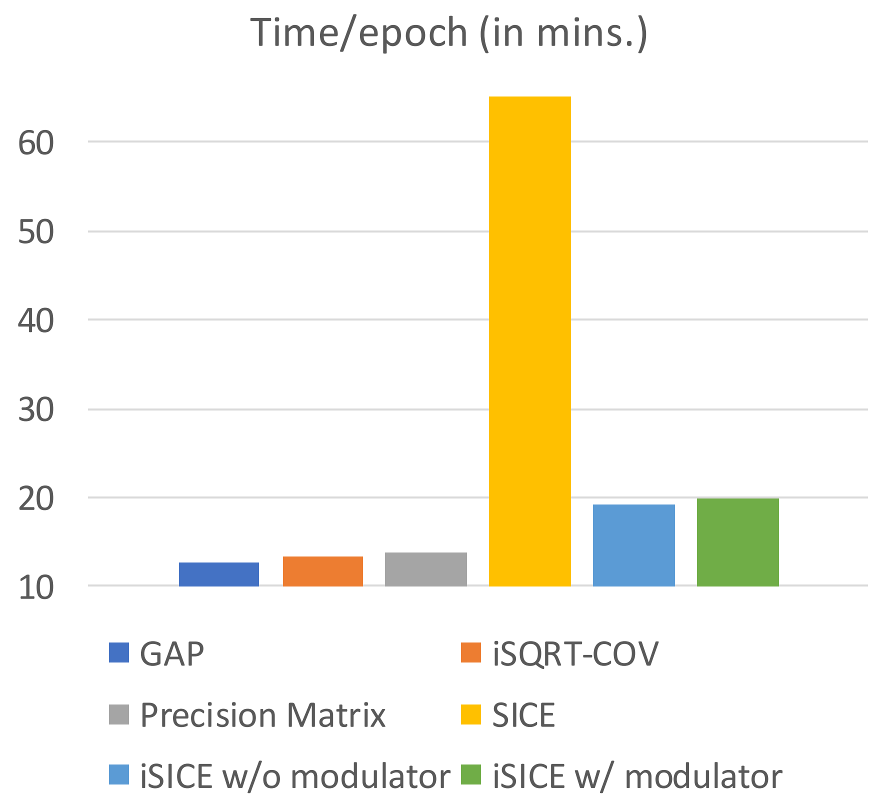
Experimental Results

iSICE with learning rate and sparsity modulators.

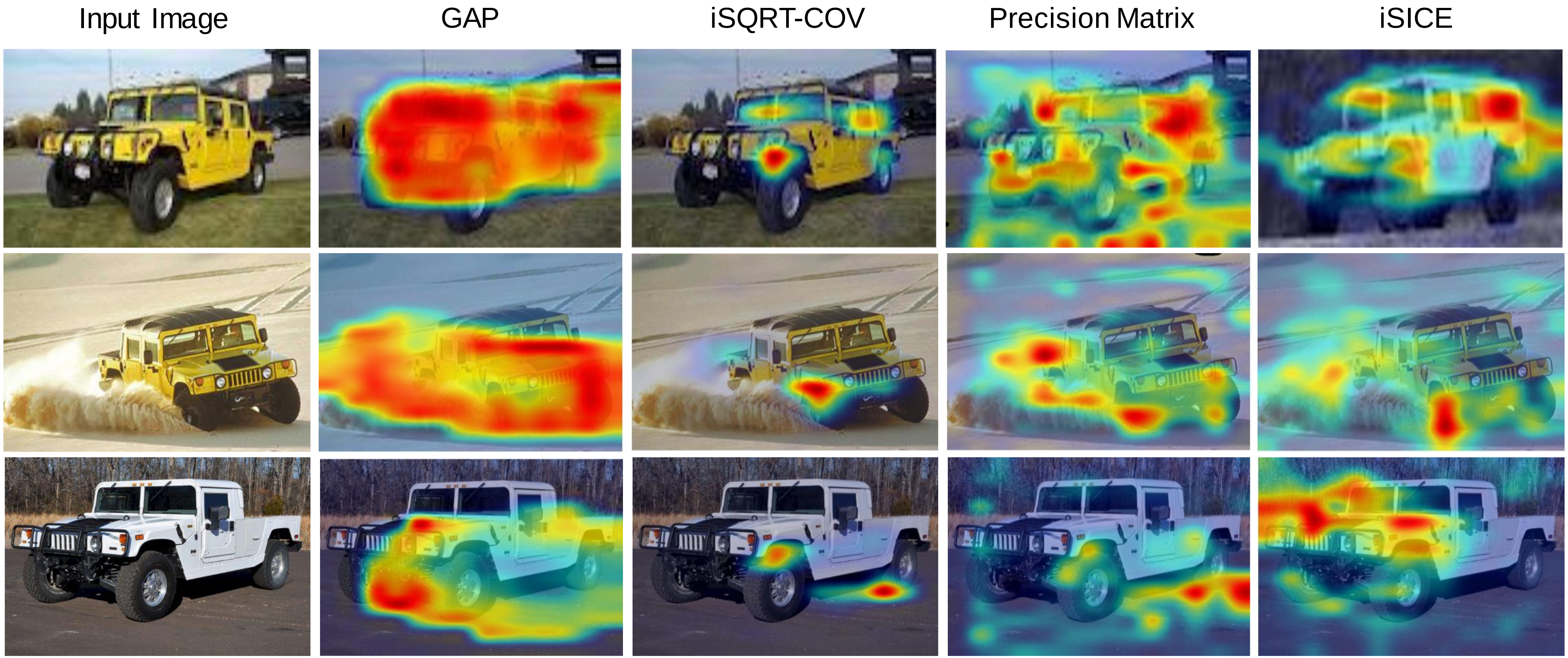


Computation Cost





Visualisation of Learned Features



Visit our project website for code and more details

https://csiro-robotics.github.io/iSICE

https://github.com/csiro-robotics/iSICE

Scan with QR Reader

Thank You