RACV研讨会
2021年10月16日，恩施

开放环境自适应感知的研究问题和基本思路

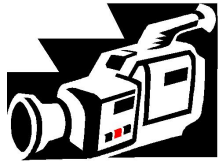
刘成林
中国科学院自动化研究所
模式识别国家重点实验室
liucl@nlpr.ia.ac.cn
http://www.nlpr.ia.ac.cn/liucl/

Outline

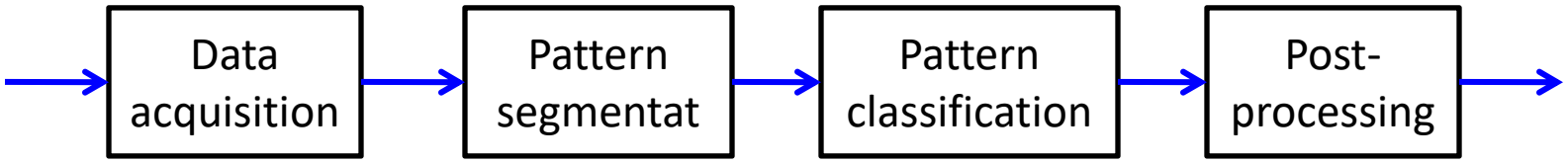
- Status of Pattern Recognition
- Problems in Open World Recognition
- Basic Idea for Open World Recognition
- Convolutional Prototype Network (CPN) for Open Set Recognition
- CPN in Incremental Learning
- Future Work

Status of Pattern Recognition

- Pattern Recognition: Simulating human perception ability, to enable machine to detect and recognize objects and events in sensory data

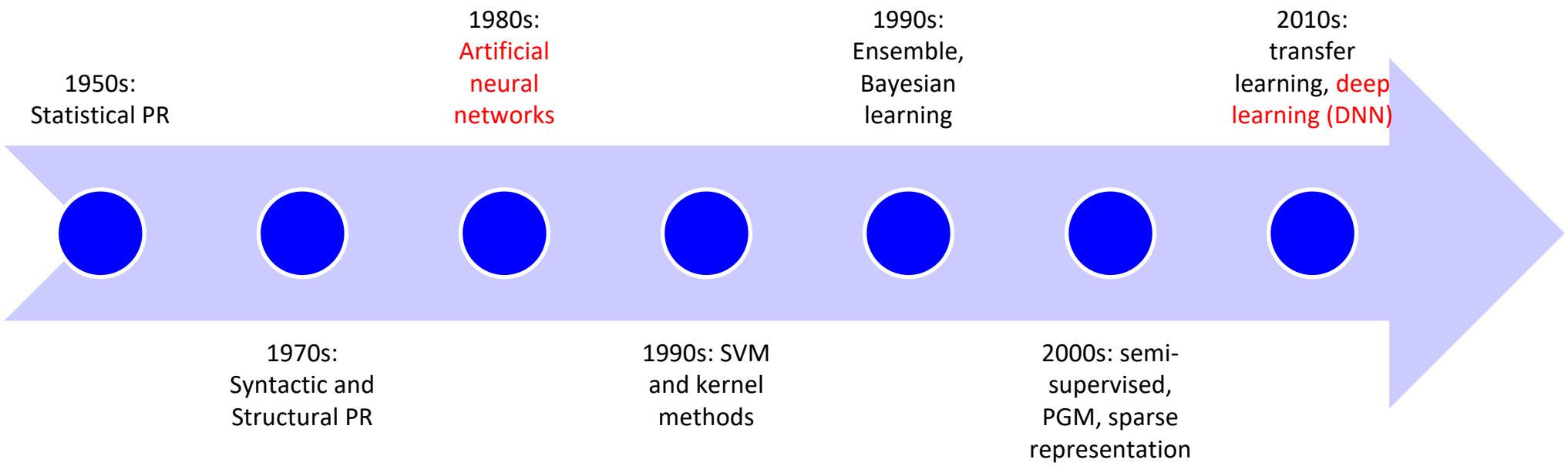


姚明/田亮	姚明
ROCKET8	ROCKETS
11	11



Evolution of PR Methods

- Core technique: pattern classification
  - Feature extraction and selection, classifier design (learning)



Categorization of PR Methods

Two broad categories

- Statistical PR
  - Parametric (Gaussian)
  - Non-parametric (Parzen, k-NN)
  - Semi-parametric (GM)
  - Neural
  - Decision tree
  - Kernel (SVM)
  - Ensemble (Boosting)

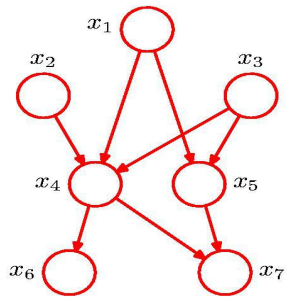
- Structural PR
  - Syntactic parsing
  - String matching, tree
  - Graph matching
  - Hidden Markov model (HMM)
  - Markov random field (MRF)
  - Structured prediction
  - Graph Neural Network (GNN)

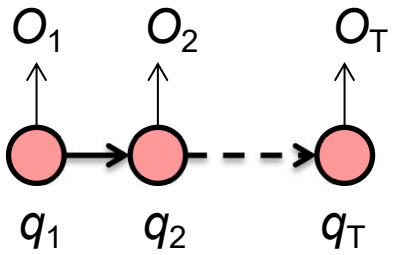
Hybrid Statistical-Structural: Statistical primitive/relationship

Attributed graphs, HMM and MRF/CRF are instances of hybrid models

Generative vs Discriminative

$$p(\mathbf{x} \mid c) = f(\mathbf{x}, \theta)$$





Generative (descriptive) model

Discriminative (boundary) model

Generative learning

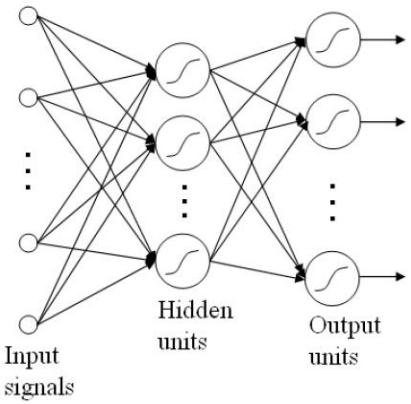
Discriminative learning

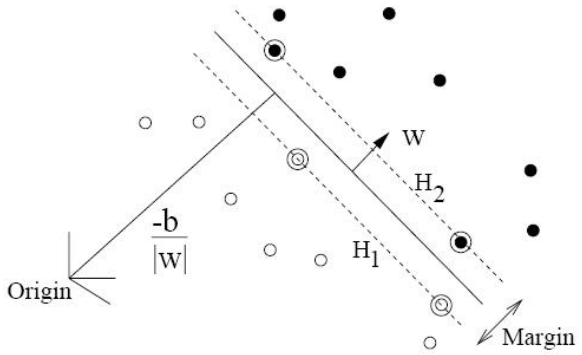
Density/description for each class

Separation between different classes

Hybrid Discriminative-Generative

$$y_i(\mathbf{x}) \sim P(\omega_i \mid \mathbf{x})$$



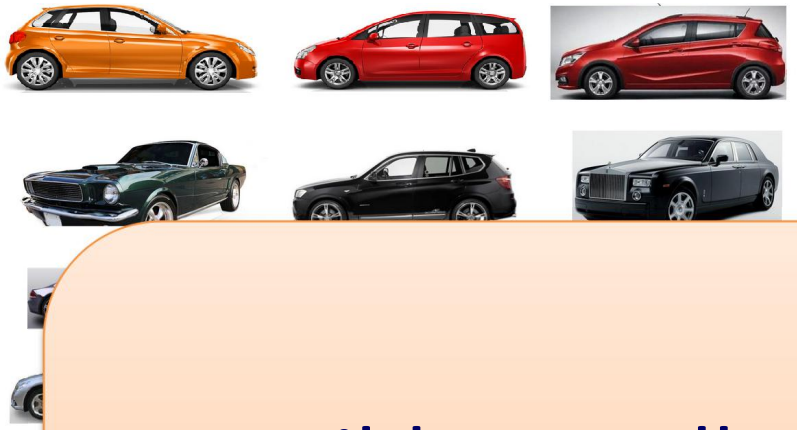


Restrictions of Current Methods

- Great Success of Deep Learning, but
- Restrictive Assumptions
  - Closed world: fixed number of classes $\sum_{i=1}^{C} P(\omega_i \mid \mathbf{x}) = 1$
    - Real world: New classes, noise, outlier
  - Same distribution (style) for training data and test data
    - Real world: data background and pattern distribution change constantly
  - Large/huge training set
- Other Issues
  - Explainability: model explainability, structural/semantic understanding
  - Flexibility of learning: mixed data, continuous learning

Pattern Recognition in Open World

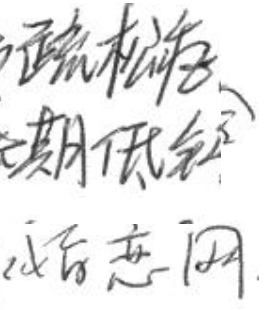
Object Recognition



distributions

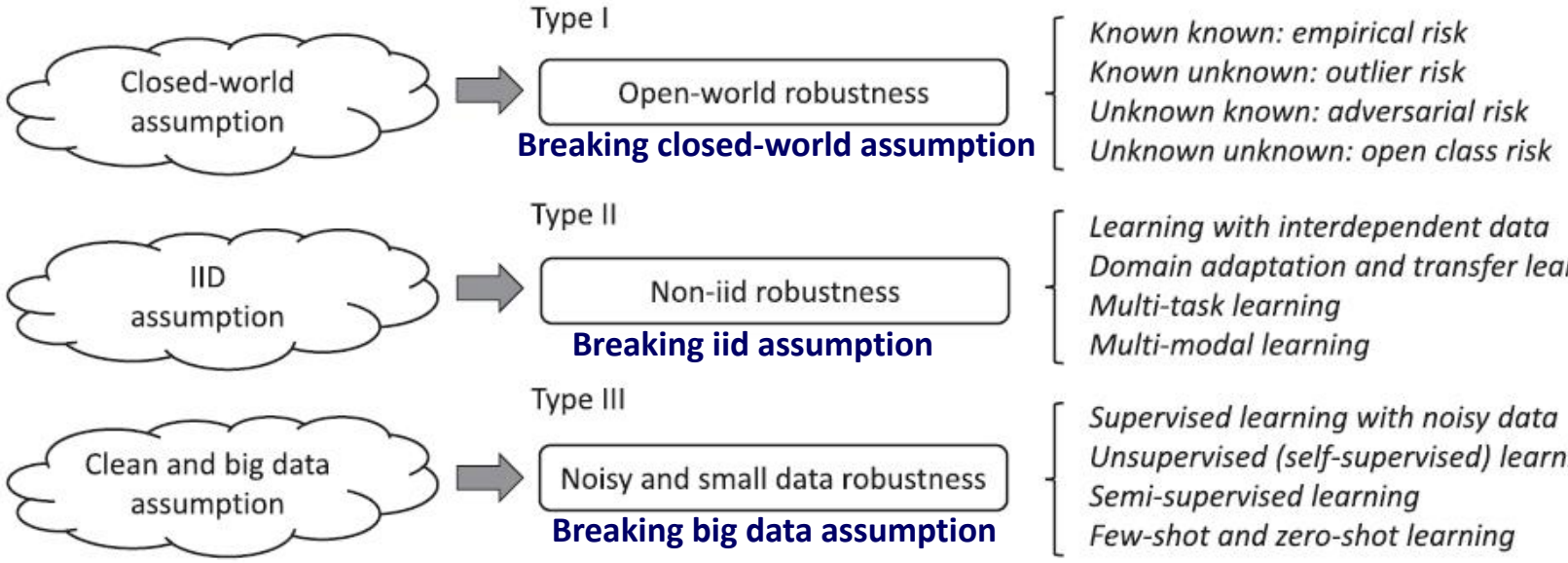
Character Recognition

啊
爱
盎
啊
爱隘鞍氨安俺按暗岸胺案肮昂
盎凹敖熬翱袄傲奥懊澳芭捌扒



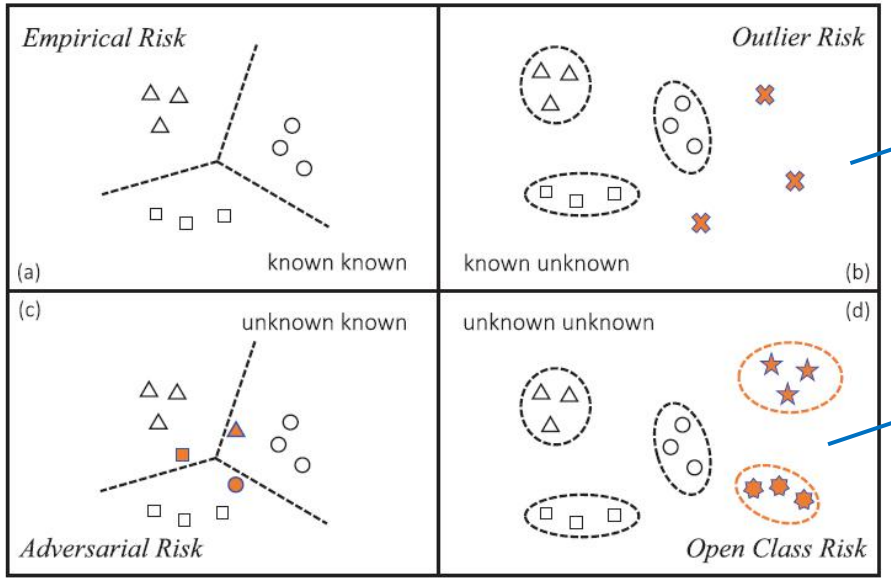
Feasible to collect enough training samples for all classes and all variations ?

Robust PR in Open World: Research Issues



Three types,
12 problems

Type I problems:
- Known known
- Known unknown
- Unknown known
- Unknown unknown



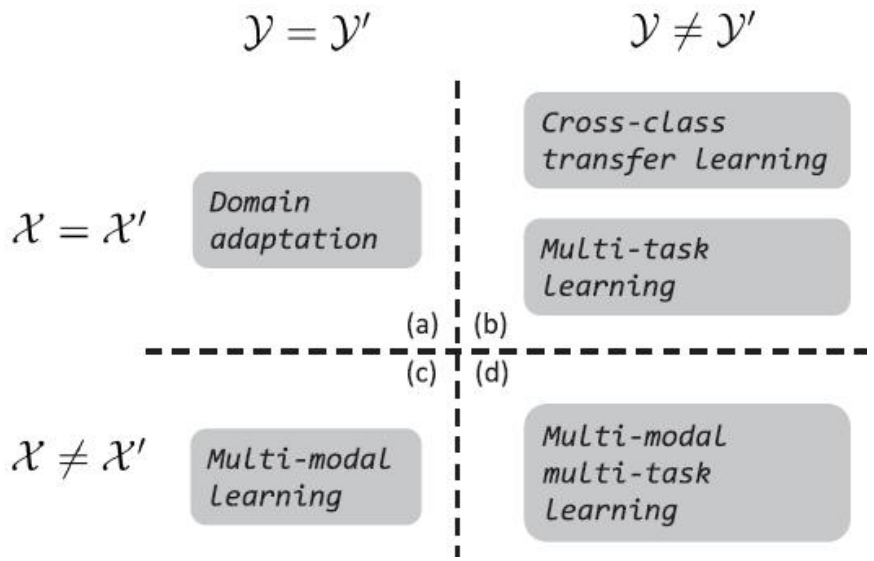
Open set recognition

Class-incremental recognition

X.-Y. Zhang, C.-L. Liu, C. Y. Suen, Towards Robust Pattern Recognition: A Review, *Proc. IEEE*, 2020.

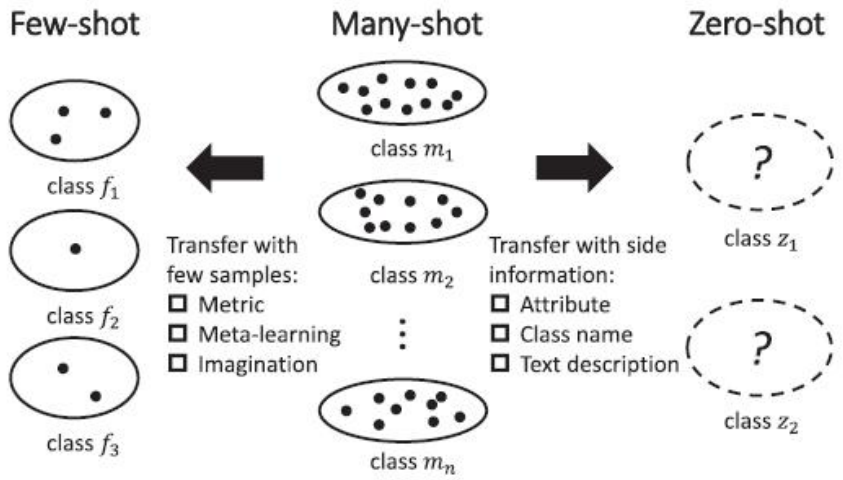
Type II: Breaking iid assumption

- Non-independent: contextual classification, structured prediction
- Non-identical: adaptation, transfer, multi-task



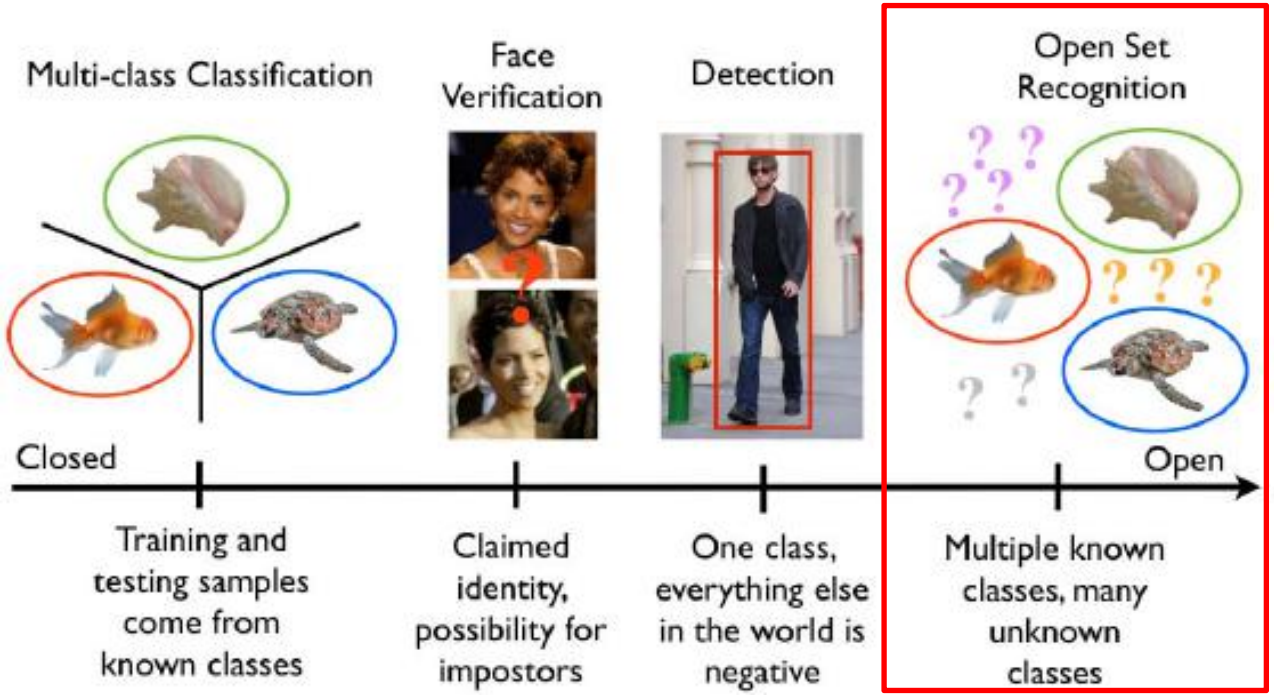
Type III: Breaking big data assumption

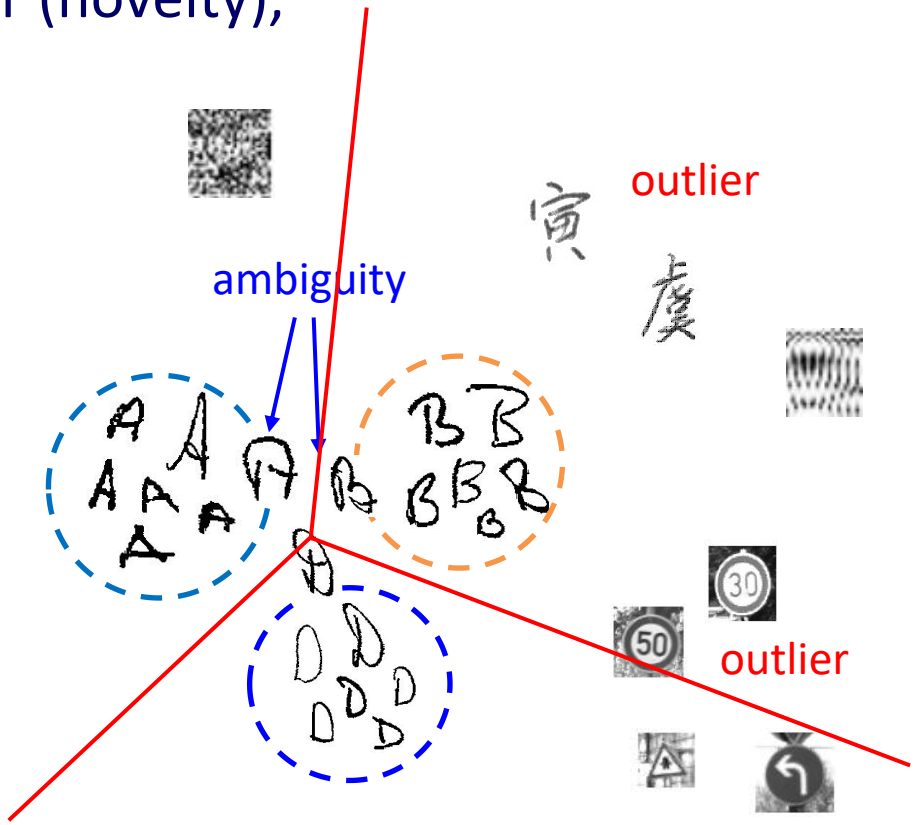
- Supervised learning with noisy data
- Unsupervised and self-supervised learning
- Semi-supervised learning
- Few-shot and zero-shot learning



Open Set Recognition

Outlier risk: many unknown negative classes or outlier (novelty), no samples available for training.

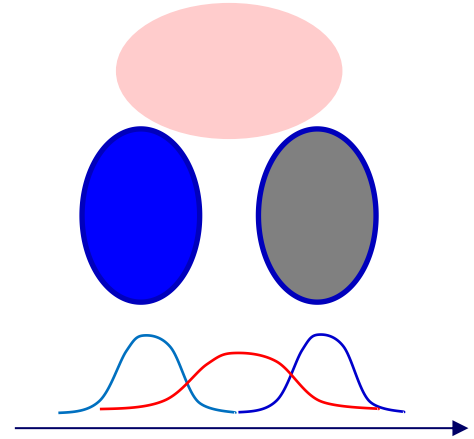




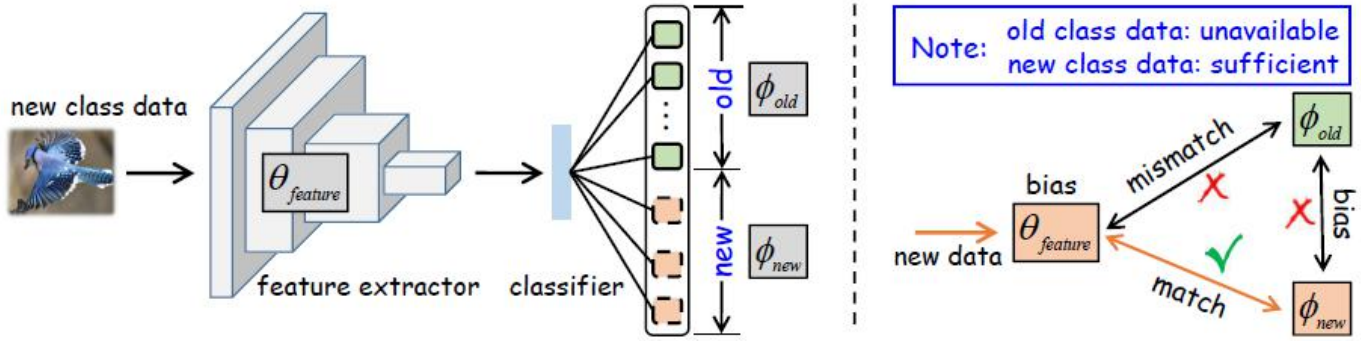
Objective: closed-set classification accuracy, outlier rejection (false negative-positive tradeoff)

W.J. Scheirer, A. de Resende Rocha, A. Sapkota, T.E. Boult, Toward open set recognition, *IEEE Trans. PAMI*, 35(7): 1757-1772, 2013.

Difficulties in Open World Recognition



- Classifier trained to fit the known classes
  - No sample for outlier, which has arbitrary distribution
  - In discriminative training, the representation tends to a subspace, where outliers are confused with know classes
- Class-incremental learning
  - Catastrophic forgetting: old classes will be mis-classified after learning for new classes
  - Representation tends to fit old classes, in the subspace new classes are confused with old classes



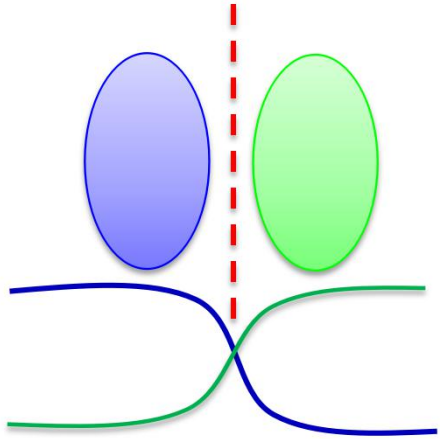
Basic Idea for Open World Recognition

- Anticipate more classes
  - Learning a representation with higher dimensionality than the number of known classes
    - Generative loss in learning, e.g., reconstruction loss
    - Generating imaginary classes in training
- Minimize feature space change in incremental learning
- Open world assumption for the model
  - Generative model, one-class model

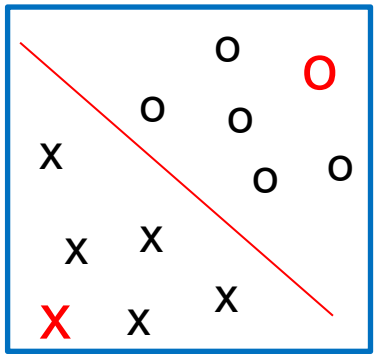
    $p(\mathbf{x}|c)$, reject if $P(c)p(\mathbf{x}|c)<T$ for all c
  - One-vs-all learning

Why Classifiers Are Poor in Open Set Recognition

- Closed World Assumption
  - *C* classes, $\sum_{i=1}^{C} P(\omega_i \mid \mathbf{x}) = 1$
- Improper Model
  - max $P(\omega_i|\mathbf{x})$ on outlier can be large
    - Discriminative models directly approximate posterior probabilities



- Improper Learning Algorithm
  - Trained model deviates from the pattern class distribution or structure
    - E.g., learning vector quantization (nearest prototype classifier)



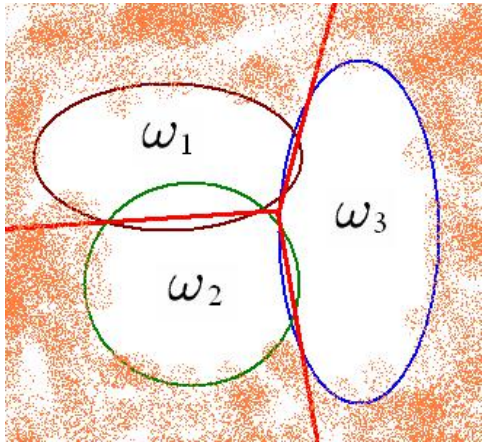
Prototype classifier

How to Improve Open Set Recognition

- Open World Assumption

$$\sum_{i=1}^{C} P(\omega_i \mid \mathbf{x}) \le 1 \qquad \sum_{i=1}^{C+1} P(\omega_i \mid \mathbf{x}) = 1$$

  - Hard to model ω_{C+1} explicitly (insufficient samples)
  - Hopefully, learning without outlier samples
- Generative Models
  - Density/template-based
  - Structure model



- Hybrid discriminative-Generative
  - Learn to represent classes in addition to discrimination

Models for Open Set Recognition

- Density-Based Classifier $\max_i P(\omega_i)p(\mathbf{x}|\omega_i)$
  - Ambiguity rejection $\max_i \frac{P(\omega_i)p(\mathbf{x}|\omega_i)}{\sum_{j=1}^{C} P(\omega_j)p(\mathbf{x}|\omega_j)} < \tau$
  - Outlier (distance) rejection $\max_i P(\omega_i)p(\mathbf{x}|\omega_i) < T$
- Template (prototype, distance)-Based Classifier
  - Ambiguity rejection $d(\mathbf{x},\omega_{r_2}) - d(\mathbf{x},\omega_{r_1}) < T_1$
  - Outlier rejection $\min_i d(\mathbf{x},\omega_i) < T_2$
- Take advantage of Deep Learning
  - Generative model in the learned feature space of DNN
  - Generative neural networks (RBM, DBN, autoencoder)

Learning Algorithms for Open Set Recognition

- Objective: High classification accuracy and robust to outlier
- Hybrid Discriminative Generative
  - Empirical risk minimization

$$\min_{\theta} R_{emp} = \frac{1}{N}\sum_{n=1}^{N} L(y_n, f(\mathbf{x}_n, \theta))$$

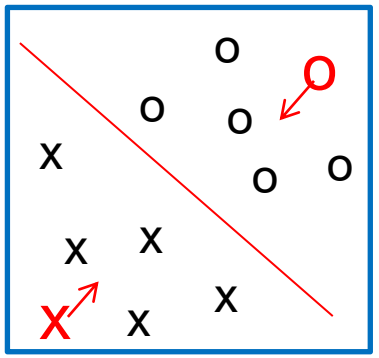
  - Maximum likelihood (ML), for generative models only

$$\max_{\theta_i} LL_i(\theta_i) = \log p(X_i \mid \theta_i) = \sum_{n=1}^{N_i} \log p(\mathbf{x}_n \mid \theta_i) \propto -\sum_{n=1}^{N_i} d(\mathbf{x}_n, \omega_i)$$

  - ML regularization (a.k.a. I-smoothing in speech recognition)

$$\min_{\theta} R_{emp} = \frac{1}{N}\sum_{n=1}^{N} [L(y_n, f(\mathbf{x}_n, \theta)) + \lambda d(\mathbf{x}_n, \omega_{y_n})]$$

  - E.g. prototype classifier



Prototype classifier

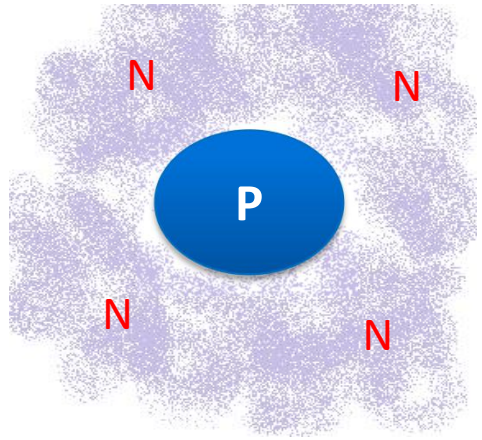
One-Versus-All (OVA) Training

- Popular in neural networks, SVM and Adaboost
  - E.g. Squared error, cross-entropy

$$E(\mathbf{w}) = \sum_{k=1}^{n}\sum_{j=1}^{c}\left(t_j^k - z_j^k\right)^2$$

- Benefit of OVA for open set classification
  - Two classes (positive, negative) cover all distributions in the world (including the outlier class)
    - Negative class: union of all the rest
    - (Binary) posterior probabilities

$$P^b(\omega_i \mid \mathbf{x}) + P^b(\overline{\omega}_i \mid \mathbf{x}) = 1$$

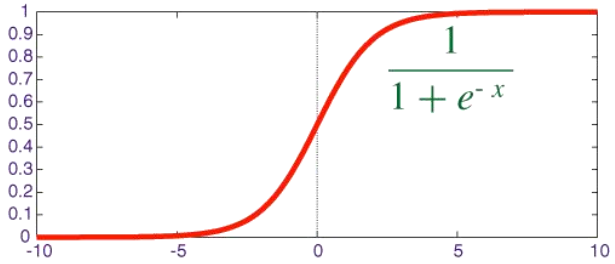


  - Multi-class: multiple OVA classifiers, each separating one class from the rest (including outlier)
    - Need to transform the multiple binary probabilities to multi-class probabilities such that

$$\sum_{i=1}^{C} P^m(\omega_i \mid \mathbf{x}) \le 1$$

- Multi-class probabilities by combining binary probabilities
  - Fusion according to Dempster-Shafer Theory of Evidence
    - Binary probability assumed to be sigmoidal

$$z_i^b = P^b(\omega_i \mid \mathbf{x}) = \sigma[\alpha f_i(\mathbf{x}) + \beta]$$



    - Evidence combination

$$P^m(\omega_i \mid \mathbf{x}) = z_i^m = m(\omega_i) = A \cdot m_i(\omega_i) \prod_{j=1, j\neq i}^{C} m_j(\bar{\omega}_j) = A \cdot z_i^b \prod_{j=1, j\neq i}^{C} (1 - z_j^b)$$

$$A^{-1} = \sum_{i=1}^{C} [z_i^b \prod_{j=1, j\neq i}^{C} (1 - z_j^b)] + \prod_{j=1}^{C} (1 - z_j^b)$$

Desired property $\sum_{i=1}^{C} P^m(\omega_i \mid \mathbf{x}) \leq 1$

C.-L. Liu, Classifier combination based on confidence transformation, *Pattern Recognition*, 38(1): 11-28, 2005.

- OVA Training: How to apply to generative models
  - Reformulate as multiple binary classifiers:
    Class distance transformed to binary discriminant

$$f_i(\mathbf{x}) = -(d(\mathbf{x}, \omega_i) - \tau_i) >< 0$$

    - Multi-prototype case

$$f_i(\mathbf{x}) = \max_j f_{ij}(\mathbf{x}) = -\min_j(\| \mathbf{x} - \mathbf{m}_{ij} \| - \tau_{ij})$$

  - Binary probability

$$p_i(\mathbf{x}) = \sigma[\xi f_i(\mathbf{x})] = \frac{1}{1 + e^{-\xi f_i(\mathbf{x})}}$$

  - Empirical loss (one-versus-all)

$$R_{emp} = \frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{C} L[\delta(y_n, i), f_i(\mathbf{x}_n)] = \frac{1}{N}\sum_{i=1}^{C}\sum_{n=1}^{N} L[\delta(y_n, i), f_i(\mathbf{x}_n)]$$

  - ML regularized loss

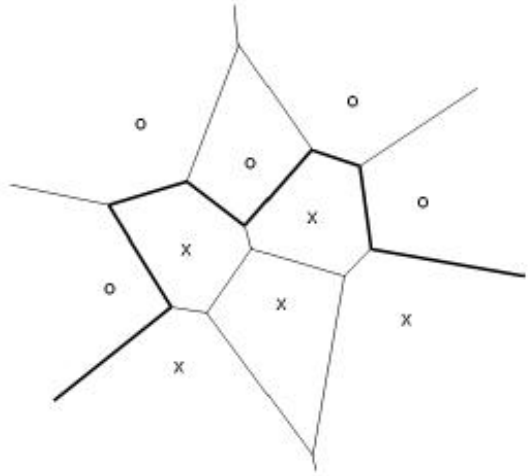
$$R_{emp} = \frac{1}{N}\sum_{n=1}^{N}\left\{\sum_{i=1}^{C} L[\delta(y_n, i), f_i(\mathbf{x}_n)] + \lambda d(\mathbf{x}_n, \omega_{y_n})\right\}$$

  - Stochastic gradient descent

Example: OVA Prototype Classifier

- Nearest Prototype Classifier

$$i = \arg\min_{i,j} \| \mathbf{x} - \mathbf{m}_{ij} \|$$



- Transform to one-vs-all
  - Each prototype as a thresholded unit

$$\| \mathbf{x} - \mathbf{m}_{ij} \| - \tau_{ij}$$

  - Class discriminant function

$$f_i(\mathbf{x}) = \max_j f_{ij}(\mathbf{x}) = -\min_j (\| \mathbf{x} - \mathbf{m}_{ij} \| - \tau_{ij})$$

  - Class probability (binary)

$$p_i(\mathbf{x}) = \sigma[\xi f_i(\mathbf{x})] = \frac{1}{1 + e^{-\xi f_i(\mathbf{x})}}$$

- Training objective: cross entropy with ML regularization

$$CE_1 = -\sum_{n=1}^{N} \left\{ \sum_{i=1}^{C} [y_i^n \log p_i + (1 - y_i^n) \log(1 - p_i)] - \lambda \| \mathbf{x}^n - \mathbf{m}_{cl} \| \right\}$$

C.-L. Liu, One-vs-all training of prototype classifiers for pattern classification and retrieval, *Proc. 20th ICPR*, 2010.

- Experiments: digit recognition error rates

n: number of prototypes per class

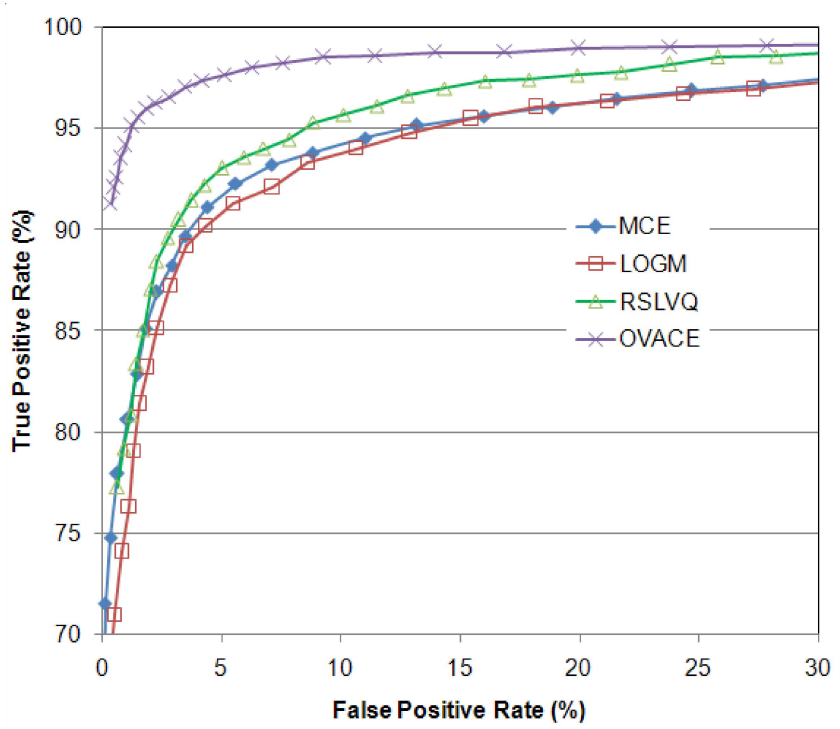
USPS

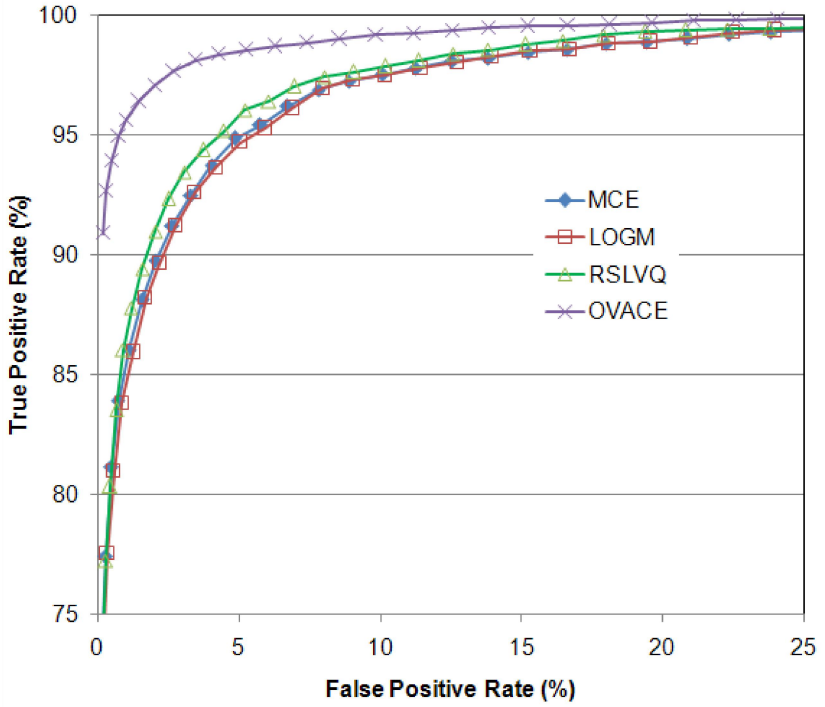
n	6	8	10
MCE	5.63	**5.43**	5.68
LOGM	5.58	**5.53**	6.03
OVACE	6.58	5.83	**5.83**

Letter

n	20	30	40
MCE	**5.08**	5.25	5.10
LOGM	4.92	5.03	**4.72**
OVACE	6.30	5.47	**4.67**

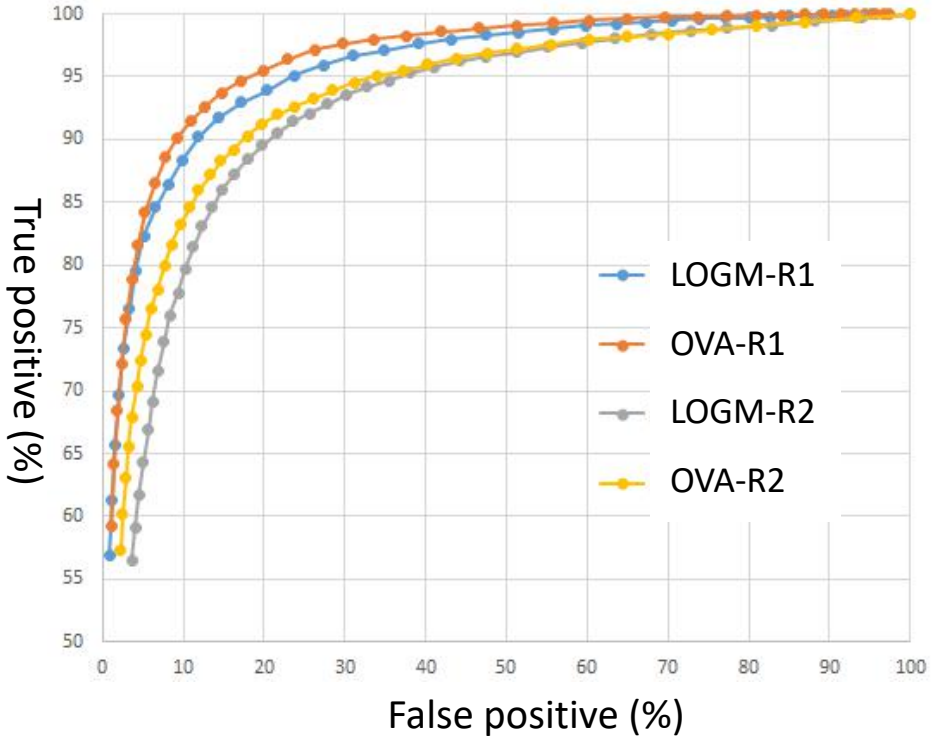
Retrieval (2-class) ROC





- OVA prototype classifier: Performance of outlier rejection

MNIST test data

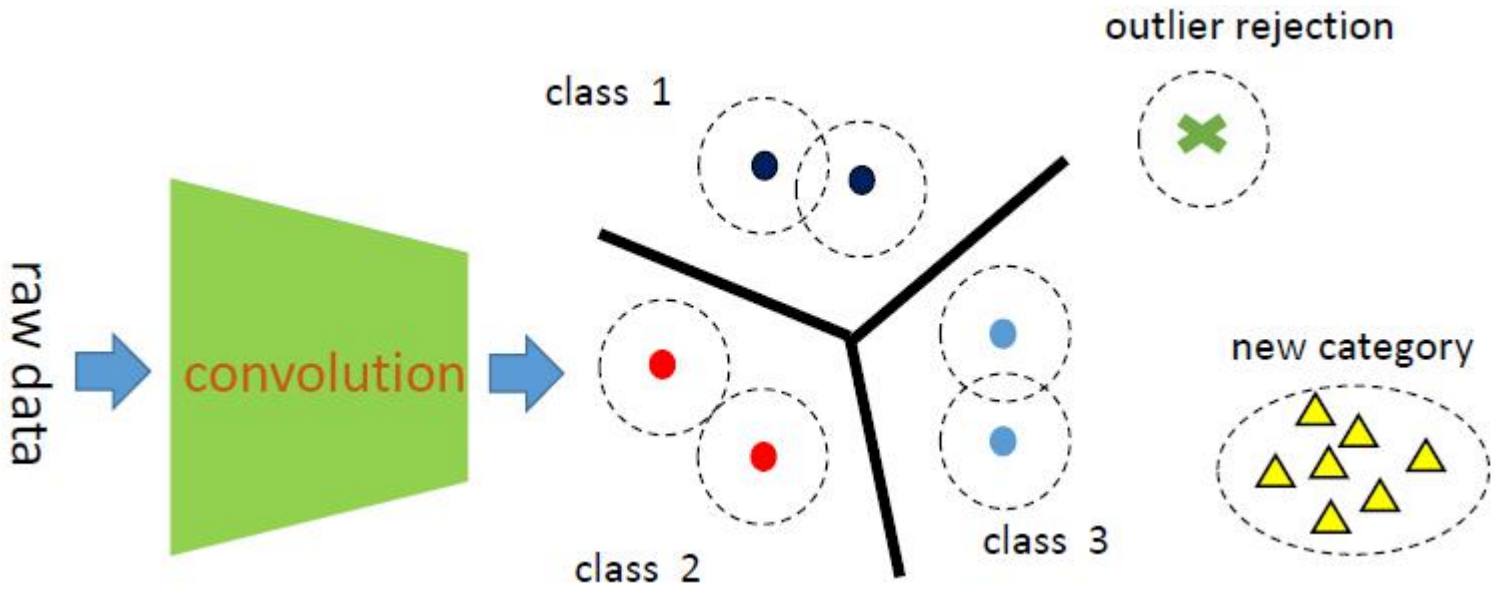


Test outlier data

R1: density-based rejection
R2: posterior-based rejection
Both models with ML regularization

Convolutional Prototype Network

- ✓ Convolutional NN: Learning discriminative feature space
- ✓ Prototype classifier: Distance based, robust to outlier
- ✓ Within-class compactness for separability and outlier rejection



H.-M. Yang, X.-Y. Zhang, C.-L. Liu, Convolutional prototype learning for robust pattern classification, CVPR 2018.

Convolutional Prototype Learning

- Learning Objective
  - ✓ Classification Loss (CL)： MCE, MCL (margin-based classification loss, GMCL, DCE (distance-based cross-entropy)
  - ✓ Feature space: desired to be compact within each class

- Regularization: Prototype Loss (PL)
  - ✓ Minimize within-class (sample-to-mean) distances
  - ✓ PL is actually a maximum likelihood (ML) regularization:

$$logp(f|y) = log\mathcal{N}(f|m_{yj}, kI) \propto \|f - m_{yj}\|_2^2$$

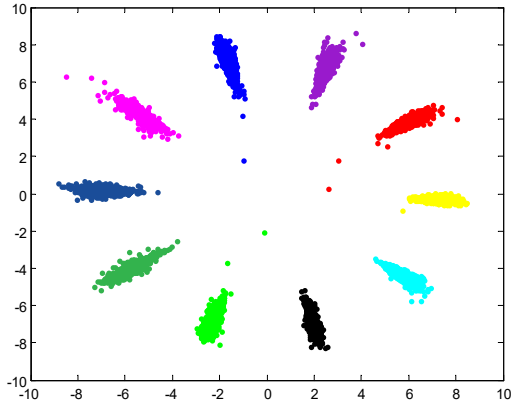
  - ✓ Based on the Gaussian assumption, can be viewed as a generative model

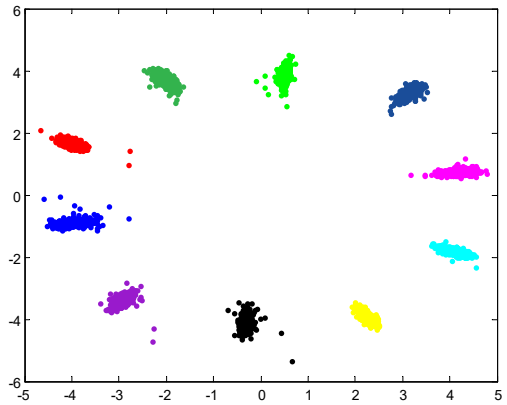
- Hybrid Discriminative-Generative Learning

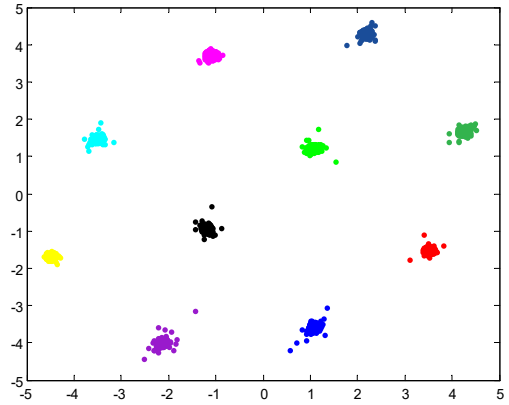
GCPL： $CL + \lambda \cdot PL$

CPL: Learned Feature Space

- Prototype Loss (PL) drives the samples of each class to be compact in feature space
- Discriminative classification loss makes different classes separate





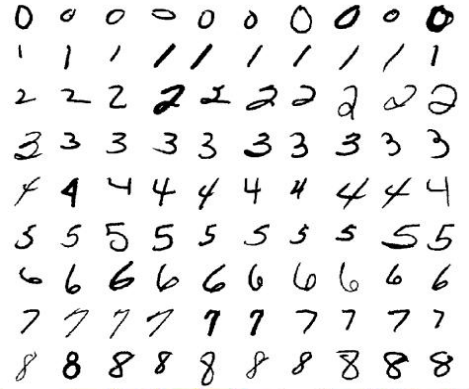


- Compactness of each class benefits robustness to outlier (which has larger distance than within-class samples)

✓ Comparable or higher accuracy than CNN with soft-max

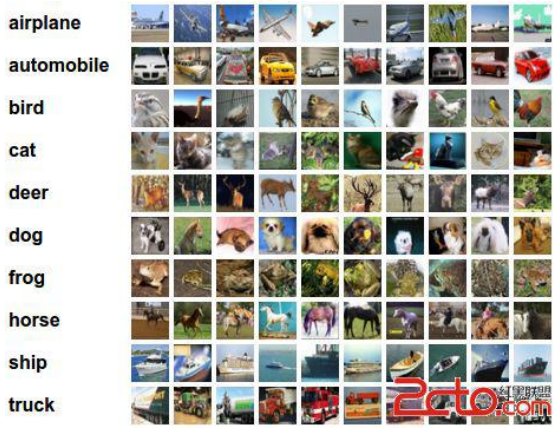
method		test accuracy (%)
soft-max		99.13
CPL (DCE)		99.28
GCPL(DCE+PL)	$\lambda = 0.0001$	99.45
	$\lambda = 0.001$	99.33
	$\lambda = 0.01$	99.29
	$\lambda = 0.1$	99.30

Table 1. Test accuracy of different methods on MNIST



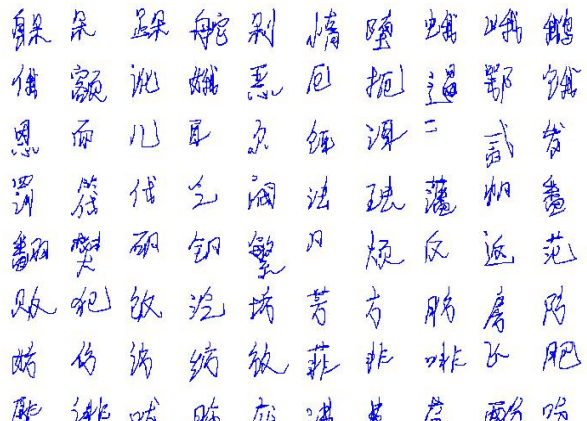
CNN structure	soft-max	CPL	GCPL
model C [34]	90.26 [34]	90.70	90.80
model C with BN	91.37	91.59	91.90
ResNet 20	91.32	91.46	91.63
ResNet 32	92.50	92.60	92.63

Table 2. The accuracy of different CNN structures and different models on CIFAR-10



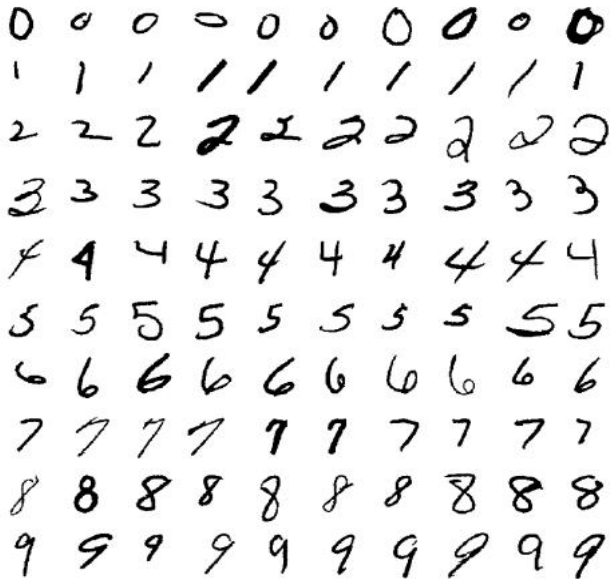
loss function	accuracy (%)
soft-max	97.55 [39]
MCE	97.35
MCL	97.61
GMCL	97.36
DCE	97.58

Table 3. The accuracy of GCPL on OLHWDB dataset



✓ Superior rejection to outlier samples

AR: Accept rate of in-class samples (MNIST digits)
RR: Reject rate of outlier samples (CIFAR-10)



softmax		GCPL Prob		GCPL Dist	
AR	RR	AR	RR	AR	RR
100.0	0.000	100.0	0.000	**94.20**	**100.0**
99.98	0.200	99.99	15.67	**97.39**	**100.0**
99.72	8.110	99.80	46.12	**98.07**	**100.0**
99.14	25.17	99.39	87.07	**98.43**	**100.0**
98.52	40.60	99.30	93.43	**98.57**	**99.99**
97.61	57.54	99.21	96.31	**98.73**	**99.99**
83.95	71.66	98.96	98.69	**98.89**	**99.99**
76.67	85.97	98.73	99.46	**99.09**	**99.99**
75.49	98.02	98.21	99.86	**99.20**	**99.99**

Table 4. The tradeoff between acceptance rate AR (%) and rejection rate RR (%) for different methods.

✓ Better generalization on small training sample

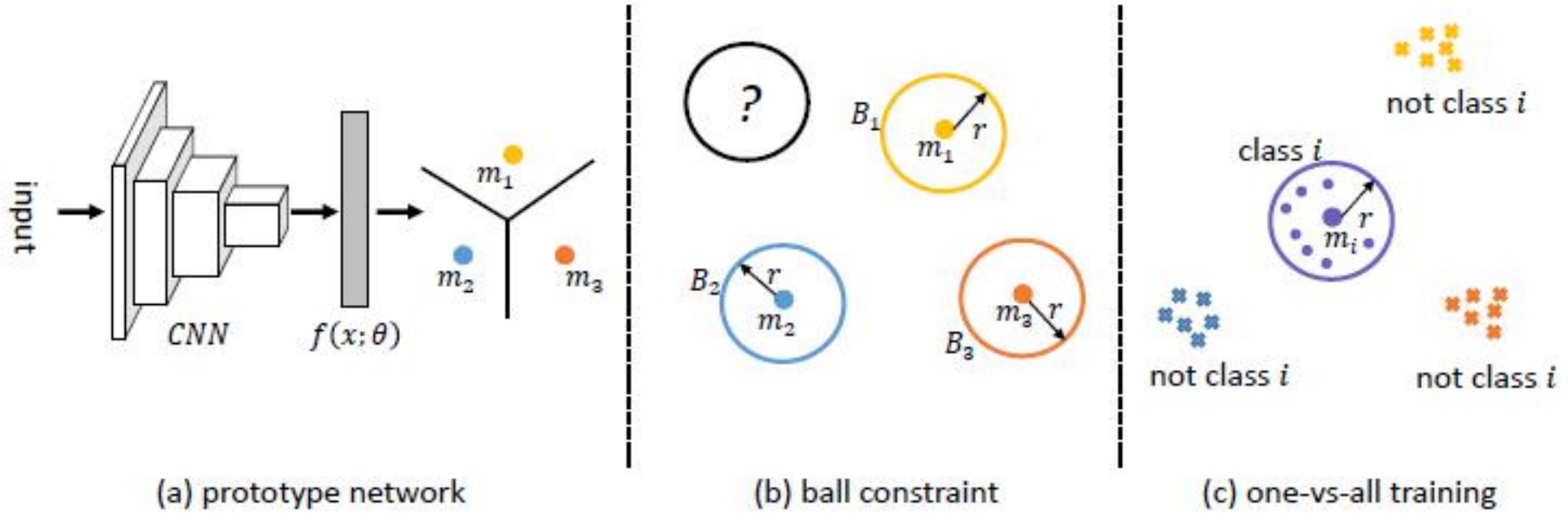
Generalized accuracies with variable number of training samples

sample size (%)	soft-max	GCPL
100	99.13 ± 0.10	99.33 ± 0.10
70	98.37 ± 0.10	99.29 ± 0.10
50	98.07 ± 0.39	99.12 ± 0.10
30	92.68 ± 4.52	98.89 ± 0.10
10	86.12 ± 6.00	97.80 ± 0.22
5	73.95 ± 6.10	96.44 ± 0.40
3	50.79 ± 17.44	94.90 ± 0.58

Table 1. Test accuracy (%) under different percentages of training samples. It is shown that GCPL is much more robust for small sample size.

CPN: Improved Learning

- ✓ Ball constrained regularization: better constrain the compactness, variable radius depending on class $f_i(\mathbf{x}) = \max_j f_{ij}(\mathbf{x}) = -\min_j(\| \mathbf{x} - \mathbf{m}_{ij} \| - \tau_{ij})$
- ✓ One-Versus-All training: without outlier sample, better separating one class from the rest



H.-M. Yang, et al., Convolutional prototype network for robust pattern recognition, IEEE T-PAMI accepted 2020.

Network structure for MNIST.

layer type	kernel size	stride	channel number	activation
Conv	5×5	1×1	32	ReLU
Pool	2×2	2×2	-	max pooling
Conv	5×5	1×1	64	ReLU
Pool	2×2	2×2	-	max pooling
Conv	5×5	1×1	128	ReLU
Pool	2×2	2×2	-	max pooling
FC	-	-	50	None

Network structure for SVHN and TinyImageNet.

layer type	kernel size	stride	channel number	activation
Dropout	-	-	-	drop rate = 0.2
Conv	5×5	1×1	32	ReLU
Conv	5×5	1×1	32	ReLU
Pool	2×2	2×2	-	max pooling
Dropout	-	-	-	drop rate = 0.2
Conv	5×5	1×1	64	ReLU
Conv	5×5	1×1	64	ReLU
Pool	2×2	2×2	-	max pooling
Dropout	-	-	-	drop rate = 0.2
Conv	5×5	1×1	128	ReLU
Conv	5×5	1×1	128	ReLU
Pool	2×2	2×2	-	max pooling
Dropout	-	-	-	drop rate = 0.2
FC	-	-	100	None

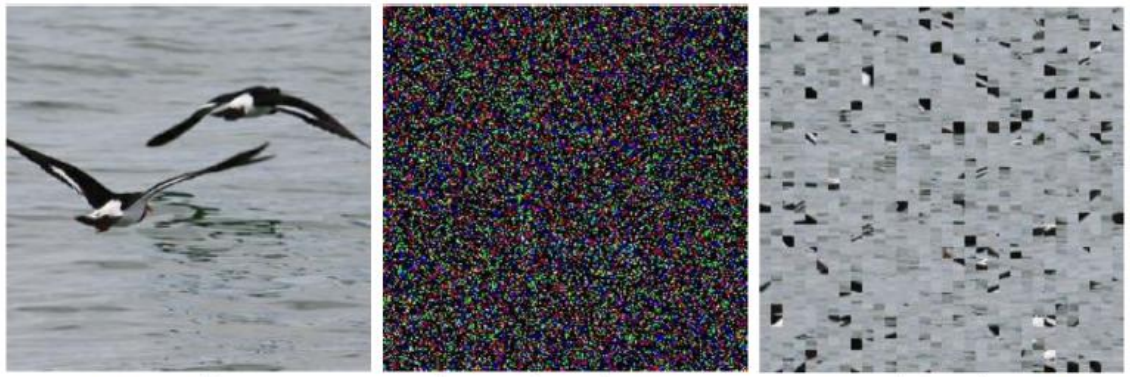
Open Set Recognition Performance (AUC).

method	CIFAR+10	CIFAR+50	TinyImageNet
softmax threshold [46]	.816	.805	.577
OpenMax [14]	.817	.796	.576
G-OpenMax [42]	.827	.819	.580
OSRCI [46]	.838	.827	.586
CPN	.864	.858	.617

Open Set Recognition Performance.

method	closed-set accuracy			open-set AUC score		
	MNIST	SVHN	CIFAR-10	MNIST	SVHN	CIFAR-10
softmax threshold [46]	.995±.002	.947±.006	.801±.032	.978±.006	.886±.014	.677±.038
OpenMax [14]	.995±.002	.947±.006	.801±.032	.981±.005	.894±.013	.695±.044
G-OpenMax [42]	.996±.001	.948±.008	.816±.035	.984±.005	.896±.017	.675±.044
OSRCI [46]	.996±.001	.951±.006	.821±.029	.988±.004	.910±.010	.699±.038
CPN	.997±.001	.967±.004	.928±.011	.990±.001	.917±.009	.772±.037

Performance of Rejecting Noise Images



(a) Normal image (b) Gaussian noise (GN) (c) Synthetic noise (SN)

Synthetic noise by randomly re-combining image blocks.

method	GNs		SNs	
	CIFAR-10	ImageNet-100	CIFAR-10	ImageNet-100
softmax	.889±.004	.998±.002	.931±.002	.863±.009
CPN (PR)	.905±.013	.998±.002	.892±.002	.825±.041
CPN (DR)	**.916±.021**	**.998±.002**	**.938±.004**	**.893±.012**

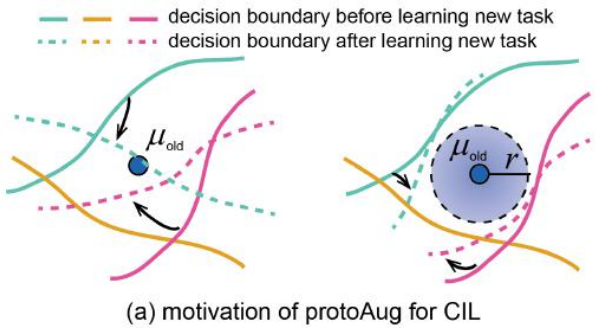
PR: posterior probability-based rejection

DR: distance-based rejection, relevant to density-based

Softmax still performs fairly well because outlier patterns can be rejected as ambiguous (comparably close to two known classes)

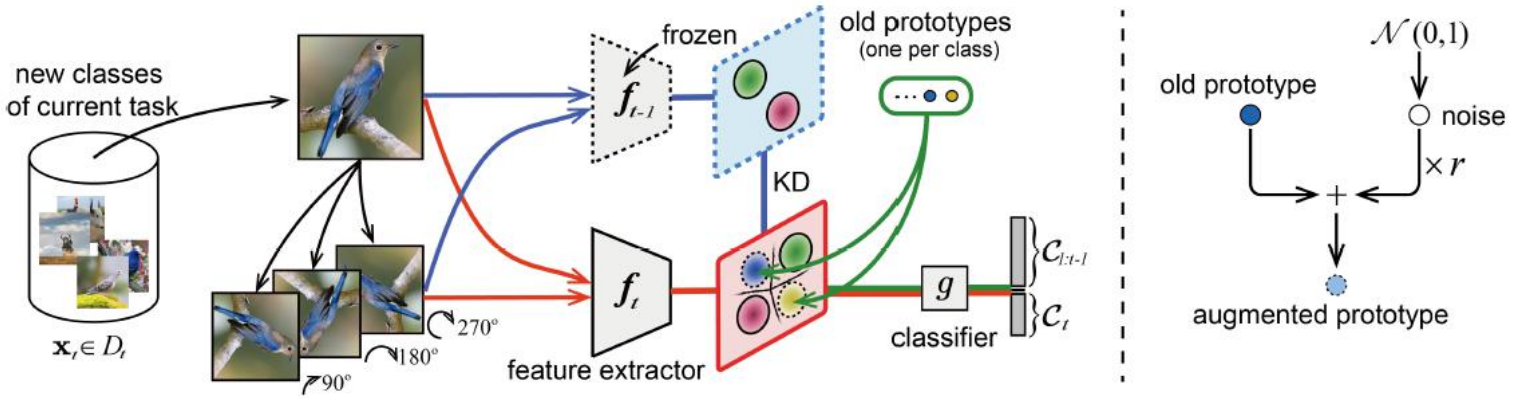
CPN in Class Incremental Learning

- Class incremental learning: model tuned on data of new classes, suffers from catastrophic forgetting (overfitting on new classes)
  - Deep models are more sensitive to forgetting because of feature representation tuning
- Convolutional prototype network (CPN) in incremental learning
  - Prototypes function as Gaussian distributions, generate samples of old classes
  - No need of storing samples of old classes

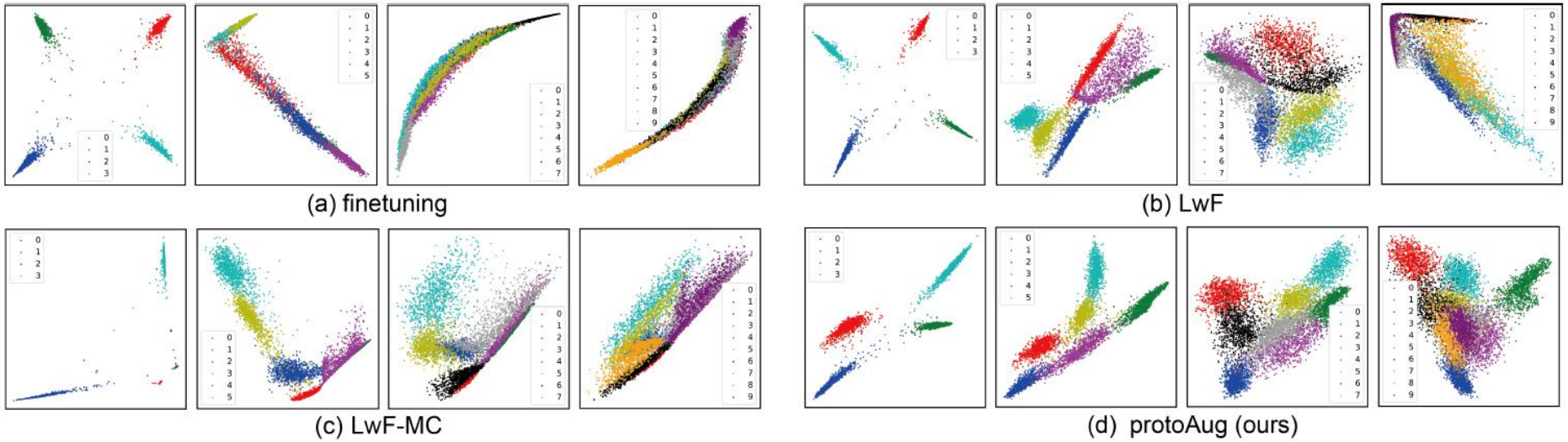


(a) motivation of protoAug for CIL

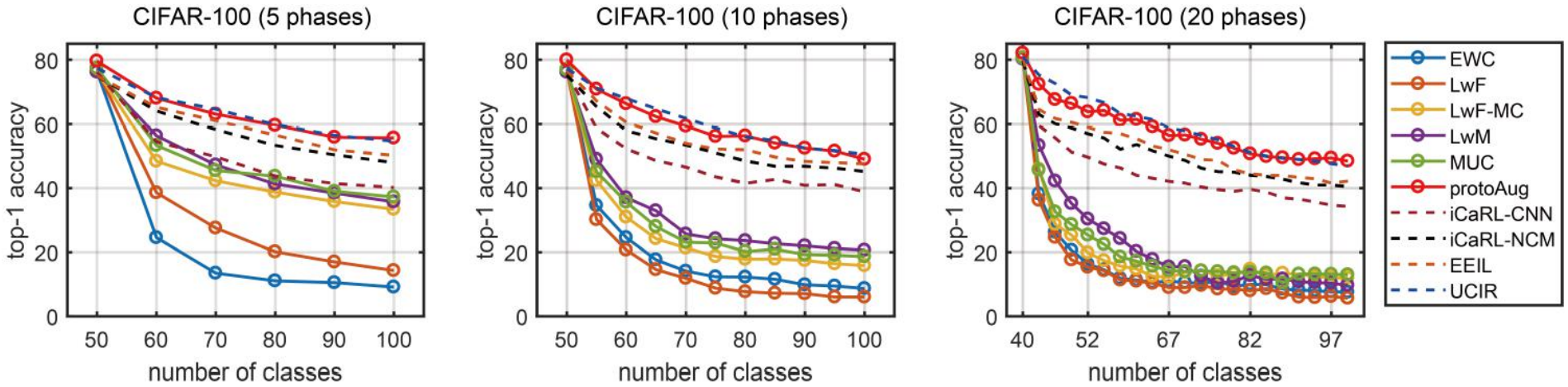
Prototype augmentation to restrain the decision boundary of old classes



Zhu, Zhang, Wang, Yin, Liu, Prototype Augmentation and Self-Supervision for Incremental Learning, CVPR 2021.



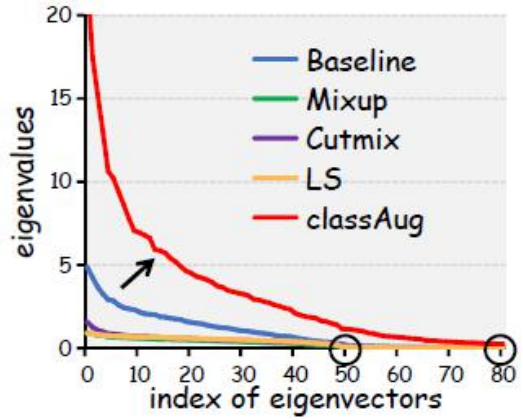
ProtoAug keeps old class distributions and separability.



ProtoAug outperforms previous non-exemplar based methods and performs comparably with exemplar-based methods.

CIL via Dual Augmentation

- Class augmentation (classAug)
  - Randomly interpolates two samples from two classes to general a new sample representing a new class
  - This can increase the intrinsic dimension of the learned representation
- Semantic augmentation (semanAug)
  - Generate samples for old classes from prototypes
  - Upper bound loss for old classes when generating infinite number of samples

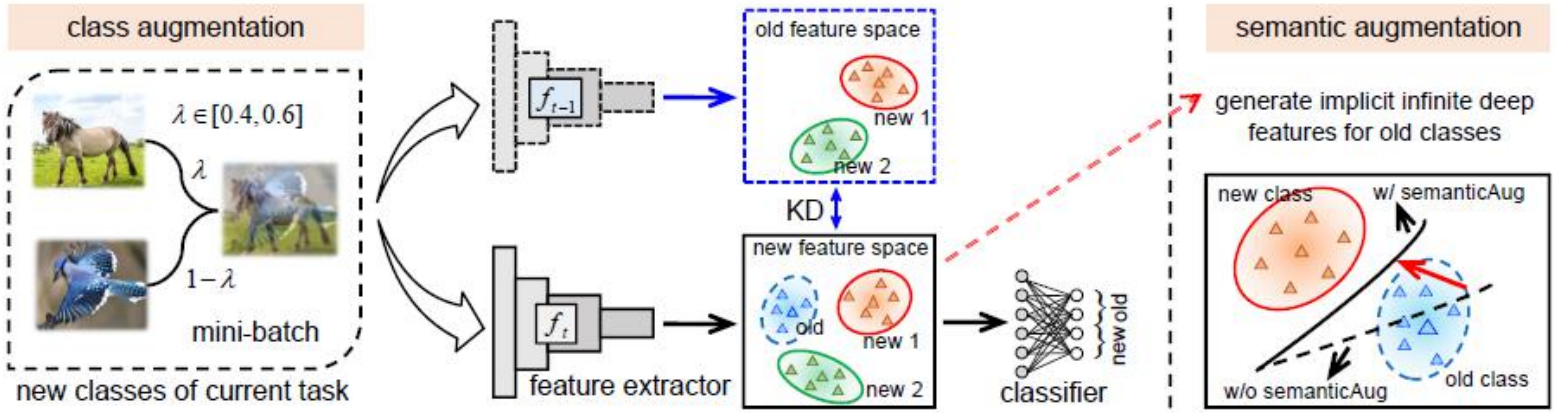


$$\mathcal{L}_{t,old} = \frac{1}{C_{old}} \sum_{i=1}^{C_{old}} \mathbb{E}_{f_i} \left[-\log \left(\frac{e^{w_{y_i}^{\mathrm{T}} f_i^m + b_{y_i}}}{\sum_{c=1}^{C_{all}} e^{w_c^{\mathrm{T}} f_i^m + b_c}} \right) \right] = \frac{1}{C_{old}} \sum_{i=1}^{C_{old}} \mathbb{E}_{f_i} \left[\log \left(\sum_{c=1}^{C_{all}} e^{(w_c^{\mathrm{T}} - w_{y_i}^{\mathrm{T}}) f_i + (b_c - b_{y_i})} \right) \right]$$

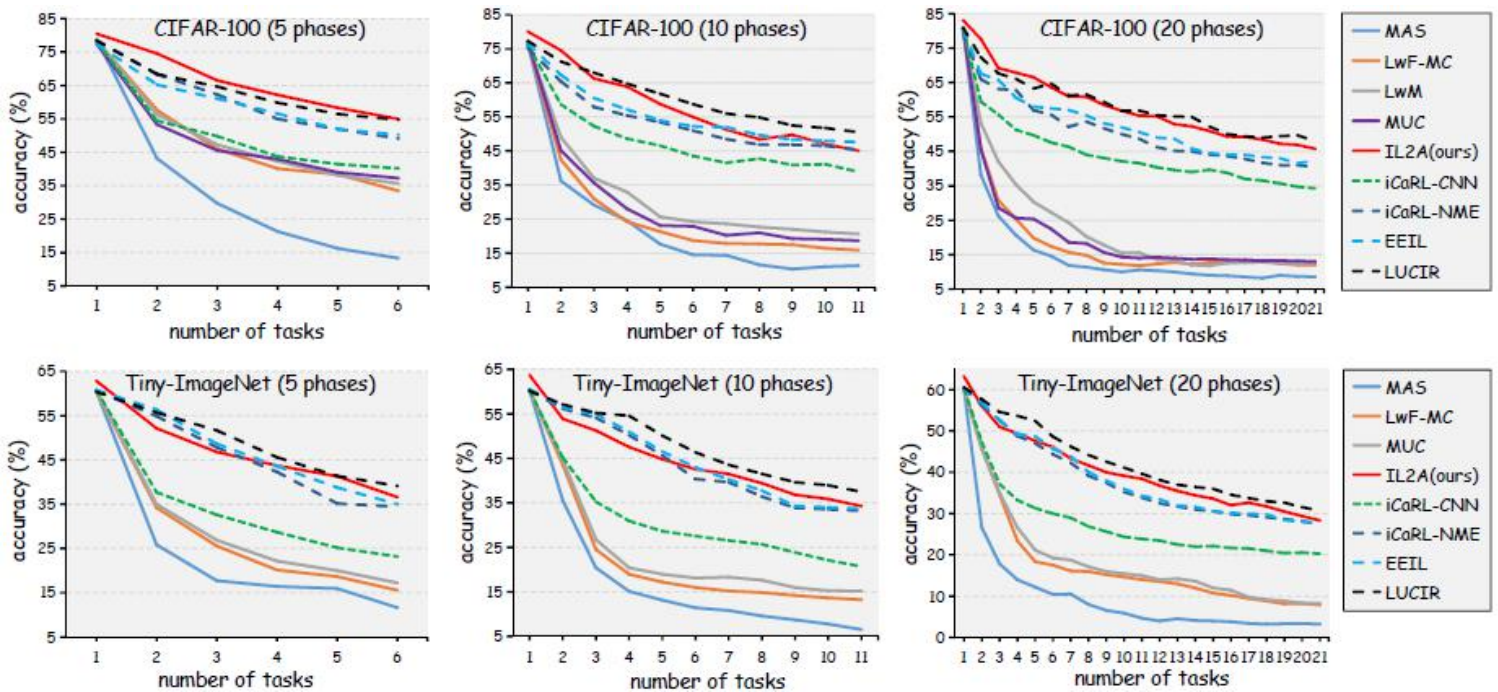
$$\leqslant \frac{1}{C_{old}} \sum_{i=1}^{C_{old}} \log \left(\mathbb{E}_{f_i} \left[\sum_{c=1}^{C_{all}} e^{(w_c^{\mathrm{T}} - w_{y_i}^{\mathrm{T}}) f_i + (b_c - b_{y_i})} \right] \right)$$

$$= \frac{1}{C_{old}} \sum_{i=1}^{C_{old}} \log \left(\sum_{c=1}^{C_{all}} e^{v_{c,y_i}^{\mathrm{T}} f_i + (b_c - b_{y_i}) + \frac{1}{2} v_{c,y_i}^{\mathrm{T}} \Sigma_{y_i} v_{c,y_i}} \right) \triangleq \mathcal{L}_{t,semanAug}.$$

Zhu, et al., Class-Incremental Learning via Dual Augmentation, NeurIPS 2021.



$$\mathcal{L}_t = \mathcal{L}_{t,new} + \alpha \mathcal{L}_{t,semanAug} + \beta \mathcal{L}_{t,kd}$$



Comparable performance with previous methods which stores samples for old classes.

Future Work

- More problems in open world recognition
  - Robustness to adversarial sample
  - Exploring new classes
  - Non-stationarity (changing distribution)
  - Small sample, long-tailed incremental learning
  - Multi-modal cooperation
- How to realize human-like learning
  - Objective: explainable, robust, small sample, incremental, adaptive
  - Generative model, structural
  - Low-level feature pre-trained?
  - Controlled forgetting and adaptation

Acknowledgements

- Sponsored by National Key Research and Development Program (创新2030—新一代人工智能重大项目), National Natural Science Foundation of China (NSFC)
- Collaborated with colleagues/students Xu-Yao Zhang, Hong-Ming Yang, Fei Zhu, Zhen Cheng

Thank You for Your Attention!

Feel free to contact:

liucl@nlpr.ia.ac.cn