

Deep Transfer Learning for Multiple Class Novelty Detection

2019 Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)

Yunseon Byun
July 06, 2020

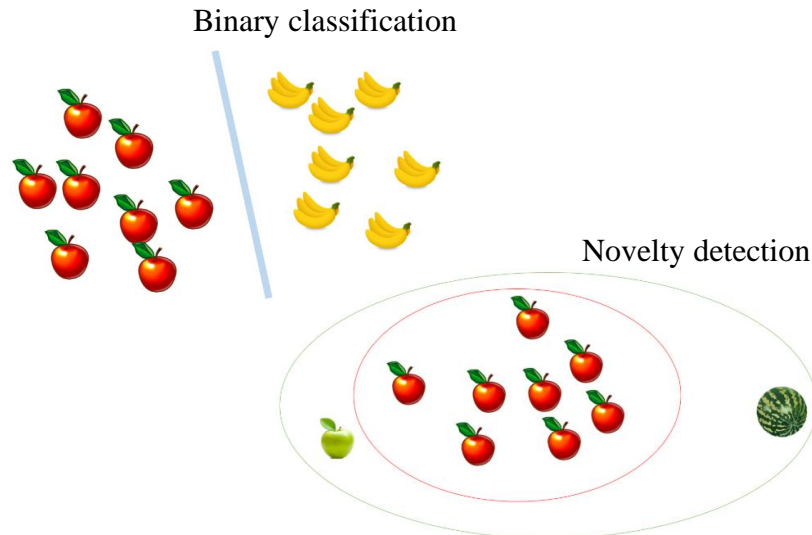
Contents

1. Introduction
 - What is novelty detection?
 - Limitation of Cross Entropy Loss
 - Transfer Learning
2. Background
3. Methodology
 - Contributions
 - Membership Loss
 - Globally Negative Filters
 - Training & Testing Procedure
4. Experimental Setup & Results
5. Conclusion










1. Introduction

What is novelty detection?

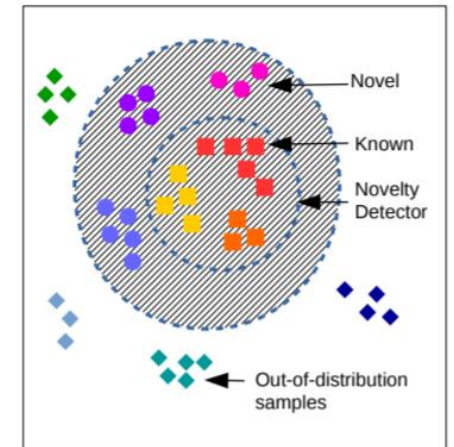
- Novel data(outliers): 다른 관측치와 비교해 많이 벗어나 있는 관측치
- Positive/Negative class 모두 학습하는 binary classification과 달리, majority class(target)만을 학습함
- Novel data는 찾고자 하는 벗어난 지점이고, noise data는 random error로서 outlier 탐지 전에 데이터 전처리 과정에서 제거해야 하는 부분임



The difference of binary classification and novelty detection

Known Classes (Dogs)			
Novel classes (Dogs)			
Out-of-distribution classes			

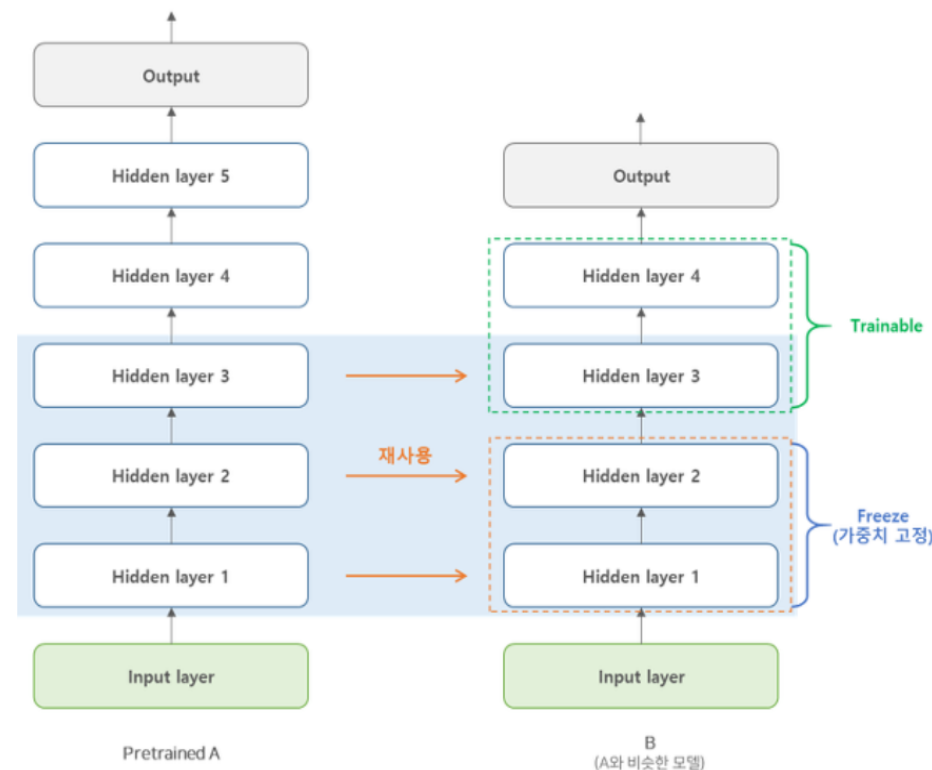
The difference of novelty and out-of-distribution samples



1. Introduction

Transfer Learning

- 이미 학습된 모델 A에서의 일부분을 재사용하여 모델 B를 학습하는 것
- 새로운 문제를 해결할 때 데이터의 분포가 바뀌면 기존의 통계적 모델을 새로운 데이터로 다시 만들어야 함
 - ✓ Layers 수, activation, hyper parameters 등 고려 사항이 매우 많음
 - ✓ 복잡한 모델일수록 학습에 많은 시간이 소요됨
- 전이학습(transfer learning)을 통해 높은 정확도를 비교적 짧은 시간 내에 달성할 수 있게 됨
- 이미 잘 훈련된 모델이 있고, 해당 모델과 유사한 문제를 해결할 때에 효과적임



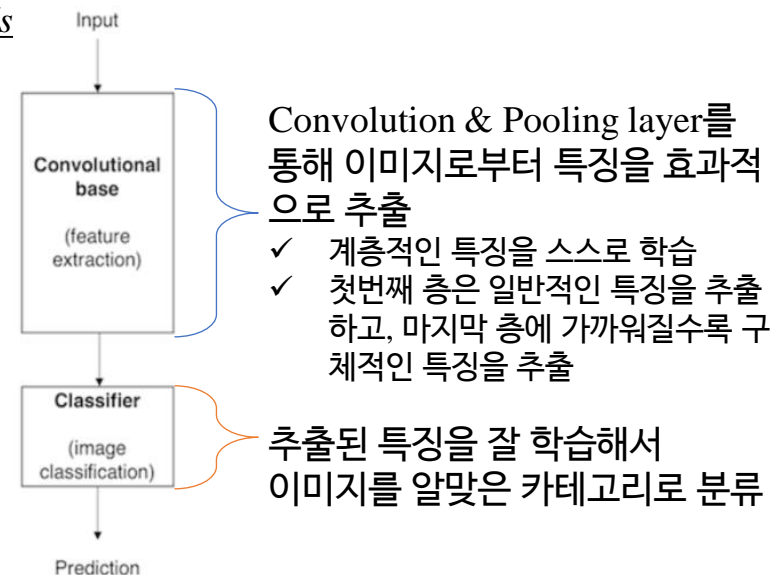
The example of transfer learning

1. Introduction

Transfer Learning

- 새로운 문제의 목적에 맞는 classifier 추가
- 1. 사전학습 모델의 구조만 사용하고 dataset에 맞게 모두 새로 학습
 - ✓ 큰 사이즈의 dataset
- 2. Convolutional base의 일부분은 고정시키고 나머지 계층과 classifier 새로 학습
 - ✓ Dataset 적거나 parameter 수가 많을 때
 - ✓ Overfitting의 위험이 클 때
- 3. Convolutional base는 고정시키고 classifier만 새로 학습
 - ✓ 컴퓨팅 연산 능력이 부족하거나 dataset이 너무 적을 때
 - ✓ 해결하고자 하는 문제가 모델이 이미 학습한 dataset과 매우 비슷할 때 (특징 추출 메커니즘으로 사용, classifier만 재 학습)

Conventional CNNs



Transfer Learning

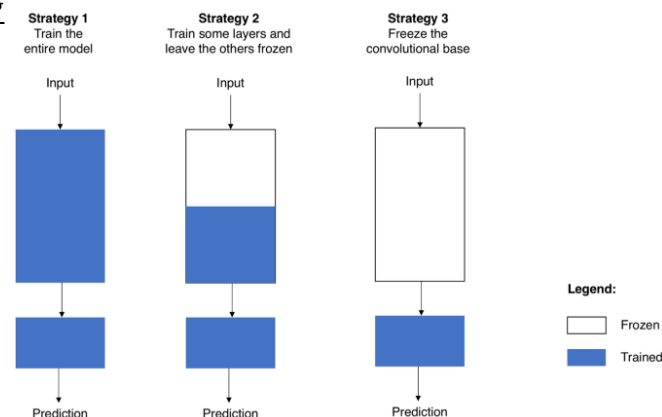


Figure of CNNs and transfer learning

2. Background

Conventional classification method

- Top most convolutional filter activation g (ex. Conv5-3 layer in VGG16, conv5c in Resnet50)
 - ⇒ Non-linear Transformation
 - ⇒ Final activation vector f (ex. fc8 layer in VGG16, fc1000 in Resnet50)
- $\operatorname{argmax} f = y_i$ by optimizing the network based on the **cross-entropy loss**
- 특정 이미지에 대한 활성화(activation)는 i 번째 class에 속할 것이라고 지지하는(support) 것을 의미함
- i 번째 filter에 대해 활성화되었다면 나머지 filter에서는 비활성화됨

“Positive filter”

(Evidence for particular class)

Fully connected layer에서
얻어지는 weight matrix W 중
positive weight와 연결된 filter

“Negative filter”

(Evidence against each class)

Fully connected layer에서
얻어지는 weight matrix W 중
negative weight와 연결된 filter

3. Methodology

Contributions

- 딥러닝 기반의 end-to-end novelty detection framework 제안
- Multiple classes에 대해 novelty detection 하기 위하여 out-of-distribution data를 사용한 transfer learning 방식 사용
- 제안방법
 1. Membership loss라는 새로운 loss function 제안 후 cross entropy와 함께 사용
 - ✓ Known class에 대해 일관적으로 높은 활성화 점수(activation score)를 부여함
 - ✓ False negative error를 줄이기 위함
 2. Globally negative filter 학습
 - ✓ Novel data에서 높은 활성화 점수를 부여함
 - ✓ False positive error를 줄이기 위함
 3. Maximal activation thresholding을 통해 novelty detection 함

3. Methodology

Limitation of Cross-Entropy Loss

- Cross-Entropy 식: $-\sum_x P(y|x) \log P(y|x; \theta)$
- Multi-class classification에서는 softmax 함수를 활용

$$P(y_i|x_i; \theta) = \frac{\exp\{f(x_i)\}}{\sum_j \exp\{f(x_j)\}}$$

- ✓ 정답 index에 해당하는 f 값을 높임 = 입력 X 에 대해 Y 의 score를 높임
- ✓ 입력 X 와 parameter θ 가 주어졌을 때, 정답 Y 가 나타날 확률(likelihood)을 최대화하는 θ 를 찾고자 함
- ✓ Cross-Entropy를 최소화하는 θ 를 찾고자 함
- Cross-Entropy는 상대적인 지표(relative measure)임
 - ✓ Non-matching class의 활성화 점수(activation score)가 낮으면, 학습 과정에서 penalty가 적게 부여되고 추론 과정에서 known class에 대한 점수가 낮게 부여됨
 - ✓ Threshold-based novelty detection 방식에서 false negative error를 발생시킴
- 관련 없는 class에 대해 0 이하의 값을 부여하지 않으므로 부적절한 cross-class relationship을 학습하게 됨

3. Methodology

Membership Loss

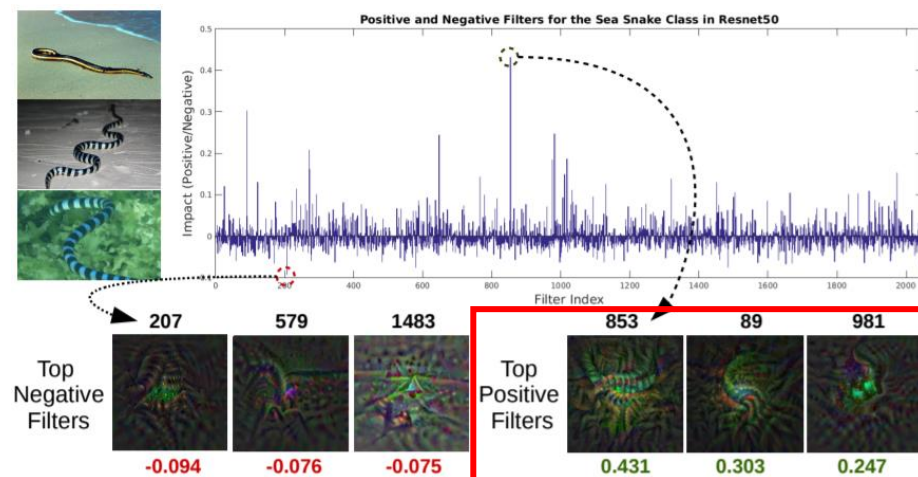
- 활성화 점수(activation score)를 각 class에 속할 확률로 변환
 - ✓ f_i 값에 대하여 sigmoid 함수를 통해 0~1 사이의 값으로 변환: $P(y = i) = \sigma(f(x)_i)$
- 잘못된 class에 대해 높은 점수를 부여하는 risk (R_{W1}), 정답 class에 대해 낮은 점수를 부여하는 risk (R_{C0}) 고려해 membership loss 제안
- 적게 활성화되는 known samples에는 penalty를 부여함

$$R_{W1}(x, y) = [1 - P(y = 1)]^2 = [1 - \sigma(f(x)_y)]^2$$
$$R_{C0}(x, y) = \frac{1}{c-1} \sum_{i=1, i \neq y}^c [P(y = 1)]^2 = \frac{1}{c-1} \sum_{i=1, i \neq y}^c [\sigma(f(x)_y)]^2$$
$$L_M(x, y) = \lambda [1 - \sigma(f(x)_y)]^2 + \frac{1}{c-1} \sum_{i=1, i \neq y}^c [\sigma(f(x)_y)]^2$$

3. Methodology

Necessity of Globally Negative Filters

- Top positive filter는 모델이 비슷한 모양을 찾았을 때 활성화됨
- 모델이 잘 학습되었다면 positive filter 값이 negative filter 값을 넘어서 높은 활성화 점수를 얻음으로써 detection 가능해짐
- Negative filter에 충분히 자극이 가해지지 않아서 positive activation이 커지는 경우가 있음
- 모든 known classes를 지지하지 않는(not supporting) negative filters를 globally negative filter라고 함

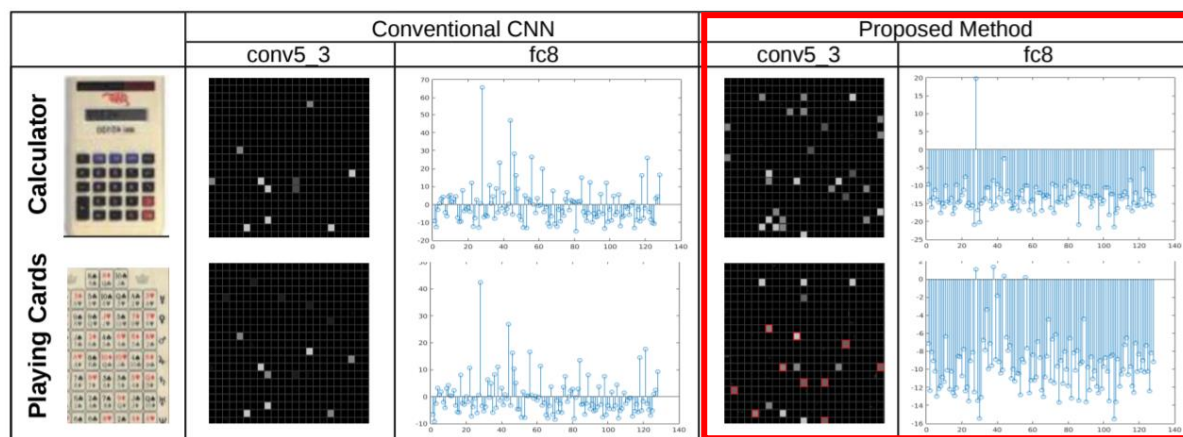


The example of positive and negative filters

3. Methodology

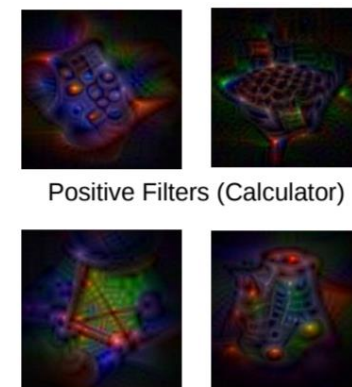
Globally Negative Filters

- Globally negative filter에 자극이 주어지면 novel data로 판정할 수 있고, known class 외의 이미지에서 자극이 주어질 때에만 novelty detection의 의미가 있음
- Filter 학습을 위해 joint learning network 구조를 제안함
 - ✓ Joint learning: 모든 task를 한 번에 학습하기 위해, 여러 loss를 더하여 하나의 최종 loss로서 사용하는 방식
- Out-of-distribution(reference) data를 학습한 filter는 모든 known classes에 대해 negative filter의 역할을 함



(a)

Activations of the conventional and proposed method



Positive Filters (Calculator)

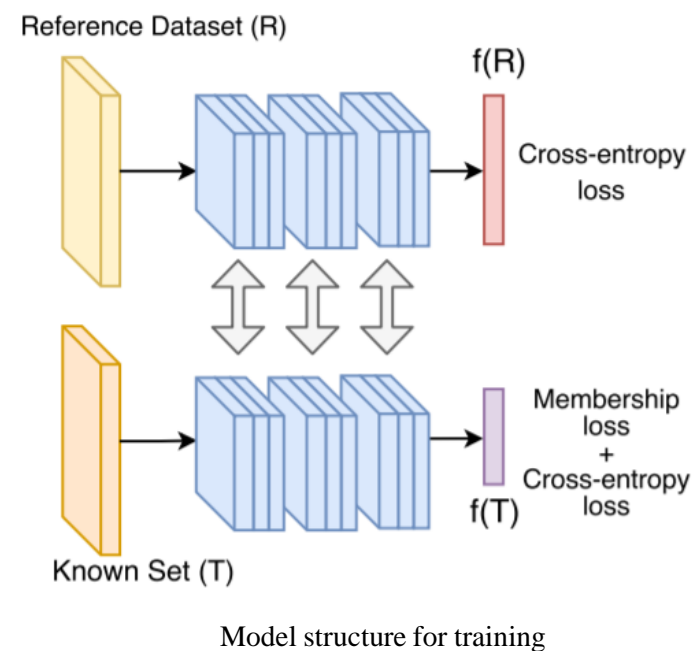
Globally Negative Filters

(b)

3. Methodology

Training Procedure

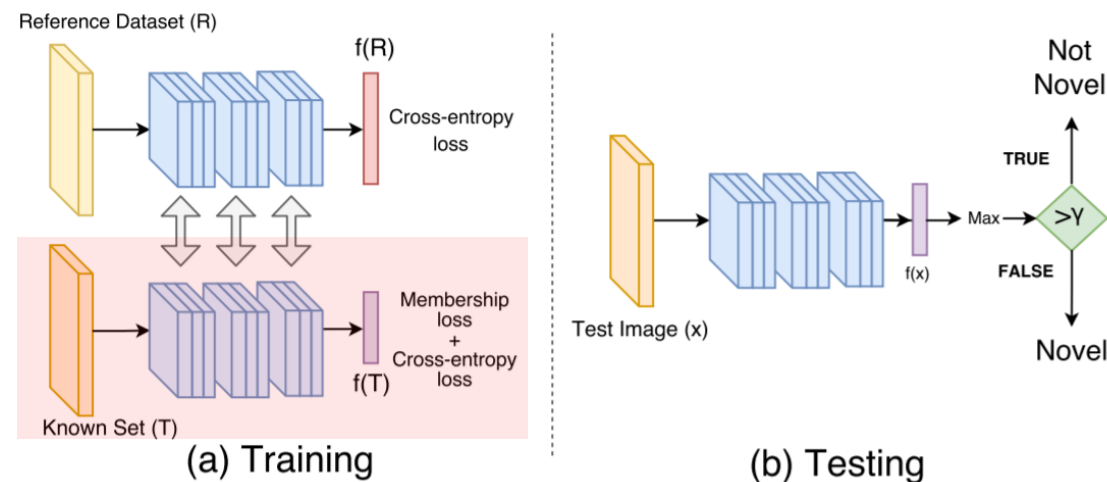
- 2개의 parallel CNN backbone 선택 (ex. AlexNet, DenseNet)
 - ✓ 동일한 structure 및 initial weights 설정
 - ✓ Final fully-connected layer의 output 수만 다르며 네트워크 간에 weights를 공유하지 않음
- 모델은 cross-entropy loss (L_{ce})를 사용해 reference dataset을 학습하고, membership loss (L_m)와 cross-entropy loss를 사용해 known classes를 학습함
 - ✓ Membership loss를 사용하여 적게 활성화되는 known samples에 penalty를 부여함
 - ✓ Membership loss와 cross-entropy loss를 함께 사용하여 correct class에 대해 상대적으로 더 높은 활성화 점수를 부여하도록 함
- $Cumulative Loss = L_{ce}(R) + \alpha_1 L_{ce}(T) + \alpha_2 L_m(T)$



3. Methodology

Testing Procedure

- Known set을 학습시킨 모델을 기반으로 진행
- Forward pass 후 얻어지는 final feature $f(x)$ 를 기반으로 가장 큰 $\sigma(f(x))$ 값을 $\text{threshold}(\gamma)$ 로 설정
 - ✓ Ex. Matched score 분포의 95th percentile 값을 threshold로 설정
- 활성화 점수(activation score)가 threshold보다 작으면 novel data로 판정함



The architecture of proposed model for novelty detection

4. Experimental Setup & Results

Datasets



The samples of dataset

Setting

- λ (relative weight to risk) = 5
- α_1, α_2 (linear combination weight of cumulative loss) = 1

Caltech256 dataset

- ✓ 128 classes as the known classes
- ✓ 128 classes as the novel images
- ✓ Reference: Places365 dataset

CUB-200 dataset

- ✓ First 100 classes as the known (bird categories)
- ✓ Remaining classes as the novel
- ✓ Reference: ILSVRC12 dataset

Stanford dogs dataset

- ✓ First 60 classes as the known
- ✓ Remaining classes as the novel
- ✓ Reference: ILSVRC12 dataset

FounderType-200 dataset

- ✓ First 100 classes as the known (Chinese character)
- ✓ 100 classes as the novel
- ✓ Reference: ILSVRC12 dataset

4. Experimental Setup & Results

Results

Novelty detection results on the evaluation datasets

Method	Caltech-256		CUB-200		Dogs		FounderType	
	VGG16	AlexNet	VGG16	AlexNet	VGG16	AlexNet	VGG16	AlexNet
Finetune[26], [12]	0.827	0.785	0.931	0.909	0.766	0.702	0.841	0.650
One-class SVM[23]	0.576	0.561	0.554	0.532	0.542	0.520	0.627	0.612
KNFST pre[4]	0.727	0.672	0.842	0.710	0.649	0.619	0.590	0.655
KNFST[4], [13]	0.743	0.688	0.891	0.748	0.633	0.602	0.870	0.678
Local KNFST pre[3]	0.657	0.600	0.780	0.717	0.652	0.589	0.549	0.523
Local KNFST[3]	0.712	0.628	0.820	0.690	0.626	0.600	0.673	0.633
K-extremes[24]	0.546	0.521	0.520	0.514	0.610	0.592	0.557	0.512
OpenMax[2]	0.831	0.787	0.935	0.915	0.776	0.711	0.852	0.667
Finetune($c + \mathcal{C}$)	0.848	0.788	0.921	0.899	0.780	0.692	0.754	0.723
Deep Novelty (ours)	0.869	0.807	0.958	0.947	0.825	0.748	0.893	0.741

Classification accuracy for conventional fine-tuning and the proposed method

	Caltech-256	CUB-200	Dogs	FounderType
VGG16	0.908	0.988	0.730	0.945
Proposed Method	0.939	0.990	0.801	0.950

- Single CNN with the cross-entropy loss
⇒ AUC 0.854
- Single CNN with the cross-entropy loss
+ membership loss
⇒ AUC 0.865
- Two Parallel CNNs with cross-entropy
loss
⇒ AUC 0.864
- Proposed method (membership loss and
a parallel network structure)
⇒ AUC 0.906

5. Conclusion

Summary

- End-to-end deep learning-based solution for image novelty detection
- Output vector의 가장 높은 활성화 점수(activation score)를 thresholding 하는 방식
- Contributions:
 - ✓ Membership loss
 - ✓ Training procedure that produces globally negative filters

Thank you for your listening 😊

Yunseon Byun
July 06, 2020