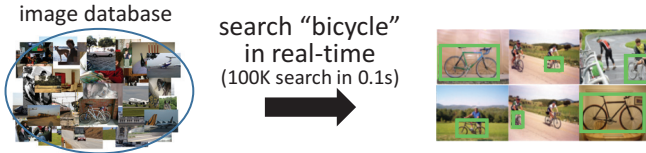


Real-time, Large-scale Object Detection

Our goal: Retrieve objects for a certain category from large image collections immediately and accurately



Problem: object detection methods requires huge costs, which makes it hard to extend them for large-scale retrieval

Prior work:

- BoVW (video google): not effective for category detection
- Aytar+ (CVPR2014): extend DPM using BoVW-like approach → we achieved significant improvement over both methods

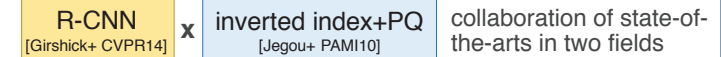
Our Idea

Summary: extend R-CNN for large scale

→ apply SVM to features extracted from all object proposal

Challenge: classify millions/billions of features in real-time

→ use the techniques of nearest neighbor search



Task definition

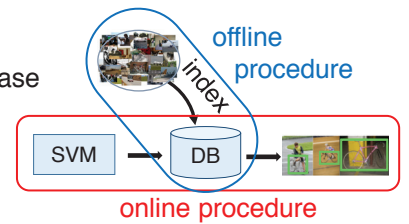
Pre-given: image database

→ indexed offline

Input: object detector

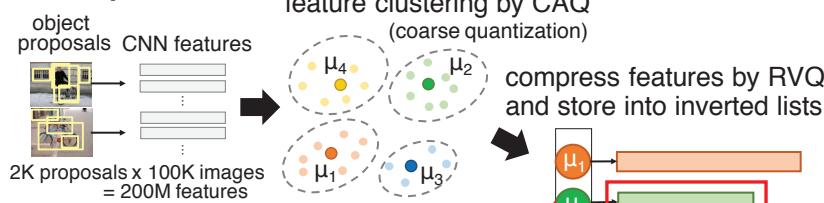
Output: ranked list

→ real-time retrieval

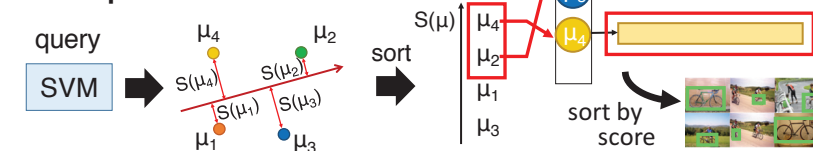


Framework of Large-scale R-CNN

Offline procedure:



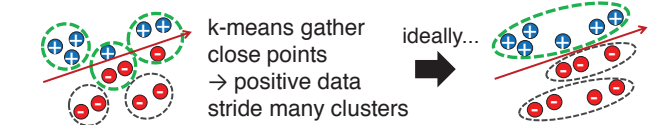
Online procedure:



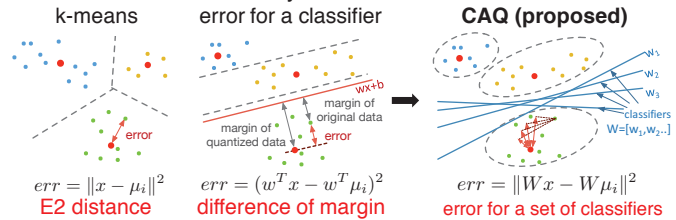
- Avoid exhaustive search by an inverted index → we use CAQ to learn codebook instead of k-means
- Efficient linear classification by PQ-based quantization (we used residual vector quantization (RVQ))

Classifier Adaptive Quantization (CAQ)

Intuition: k-means is not optimal clustering for our task



Key idea: define novel quantization error that captures common attribute of object classifier



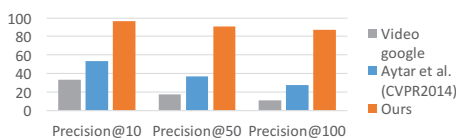
Formulation of CAQ is derived from quantization error

- encoding: $i = \arg \min \|Wx - W\mu_i\|^2$
 - codebook learning: minimize $E = \sum_i \sum_{x \in N_i} \|Wx - W\mu_i\|^2$
- similar to k-means with different metric, easy to use

Experiments

Object category retrieval: evaluated by top-rank precision

1. PASCAL VOC dataset (~5K images)



significant improvement over the state-of-the-art:
Aytar+: 28%, 37%, 54%
→ Ours: 87%, 91%, 97%

2. Large-scale dataset PASCAL+Imagenet (~105K images)

	5K	105K
bicycle	PR@10 94.0%	91.0%
chair	PR@50 87.6%	88.0%
bird	PR@100 81.1%	81.2%
time	29ms	130ms

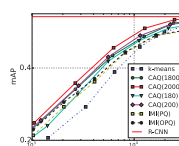
105K search in 130ms (precision is not affected by distractor)

Efficiency over R-CNN: evaluated on PASCAL dataset (around 5K images, 10M features)

	mAP	time	memory	R-CNN...
CAQ only	52.4%	518.0ms	163GB	apply SVM to 10M features
RVQ only	52.4%	69.5ms	0.64GB	
R-CNN	54.2%	6258ms	163GB	
Ours	50.7%	24.5ms	0.91GB	inverted index by CAQ quantize features by RVQ

250x faster, 106x memory reduction with comparable accuracy

CAQ vs k-means: compare methods of inverted index



Test 4 sets of classifiers in CAQ:
CAQ(200): 200 categories in ILSVRC detection task
CAQ(180): 200 in ILSVRC - 20 in PASCAL
CAQ(2000), CAQ(1800): CAQ(200), CAQ(180) x 10 (trained on randomly sampled part of dataset)

CAQ improves by ~10% mAP over k-means