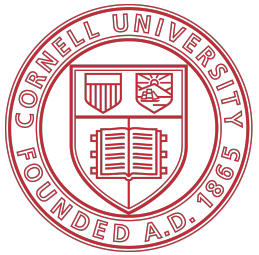


Cornell University

Optimizing JPEG Quantization for Classification Networks

Zhijing Li, Christopher De Sa, Adrian Sampson

Resource-Constrained Machine Learning (ReCoML)
at Machine Learning and Systems (MLSys)
Mar 4th, 2020

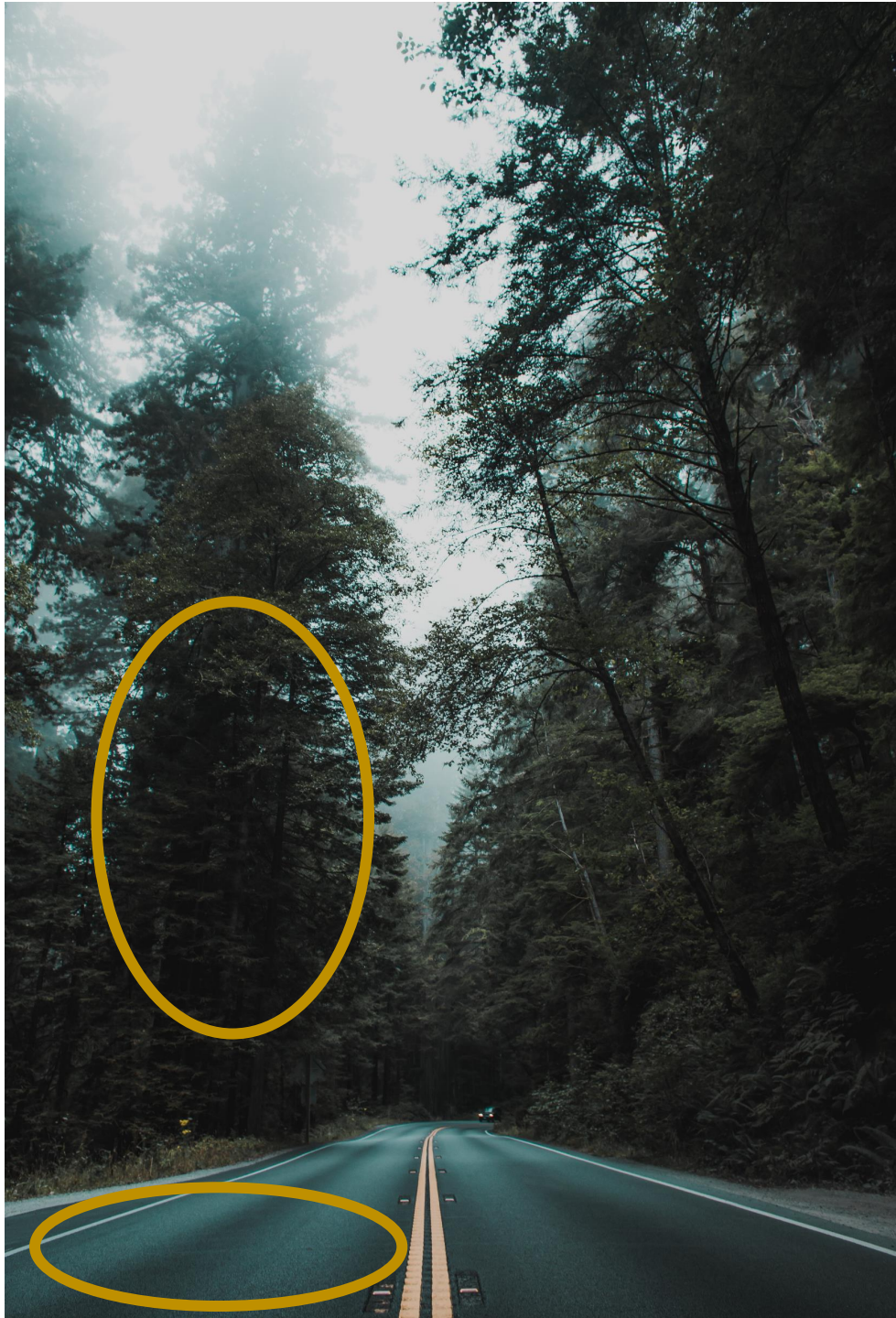
Deep Neural Networks Datasets

- DNN Datasets are
  - **large**
  - compressed in **JPEG**!

COCO	~25GB
ImageNet	~150GB
Open Image Dataset	~500GB

Why do we need to redesign JPEG?

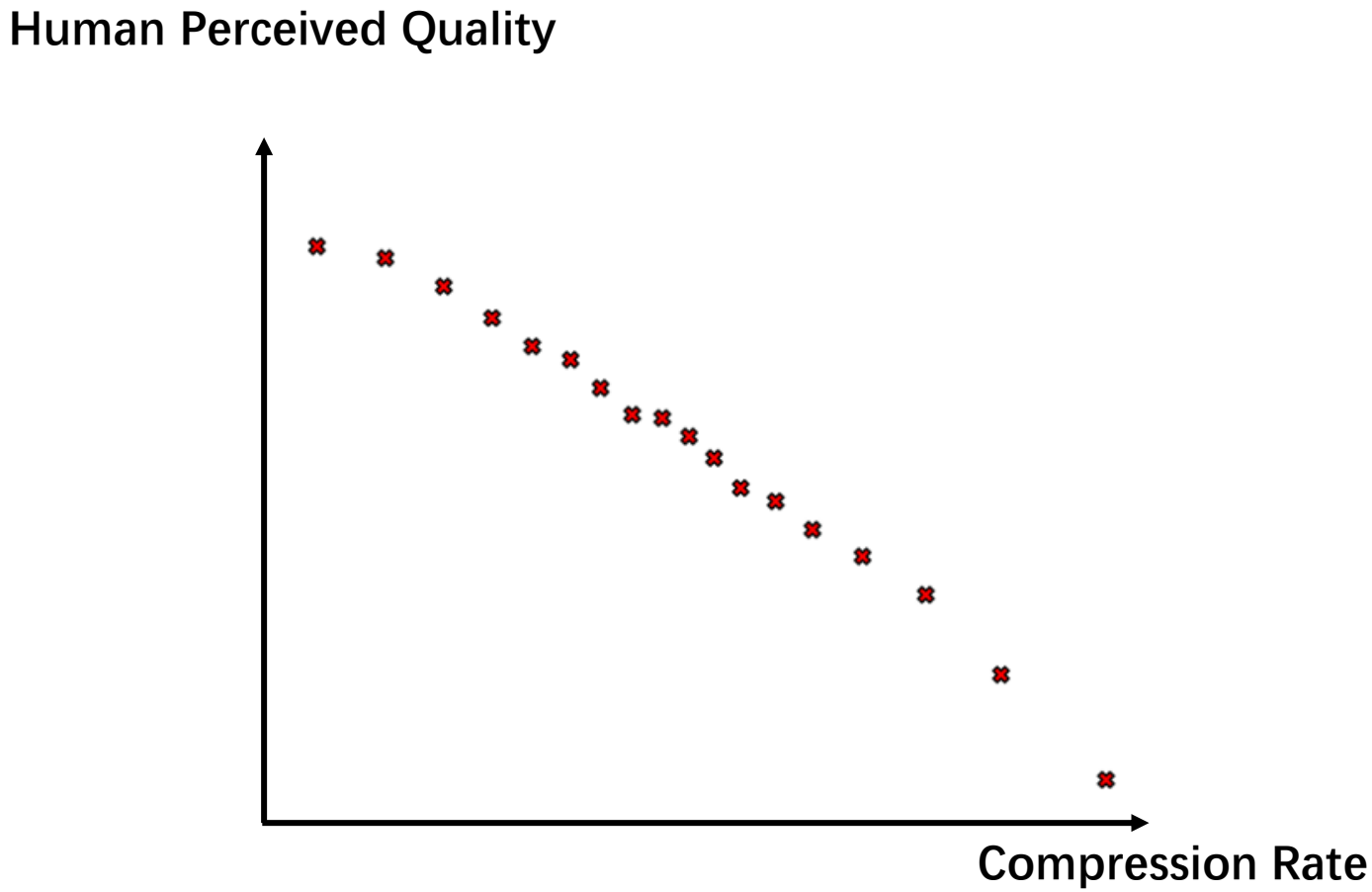
- JPEG are so designed to optimize for:
  - minimal distortion - PSNR
  - human visual system (HVS)

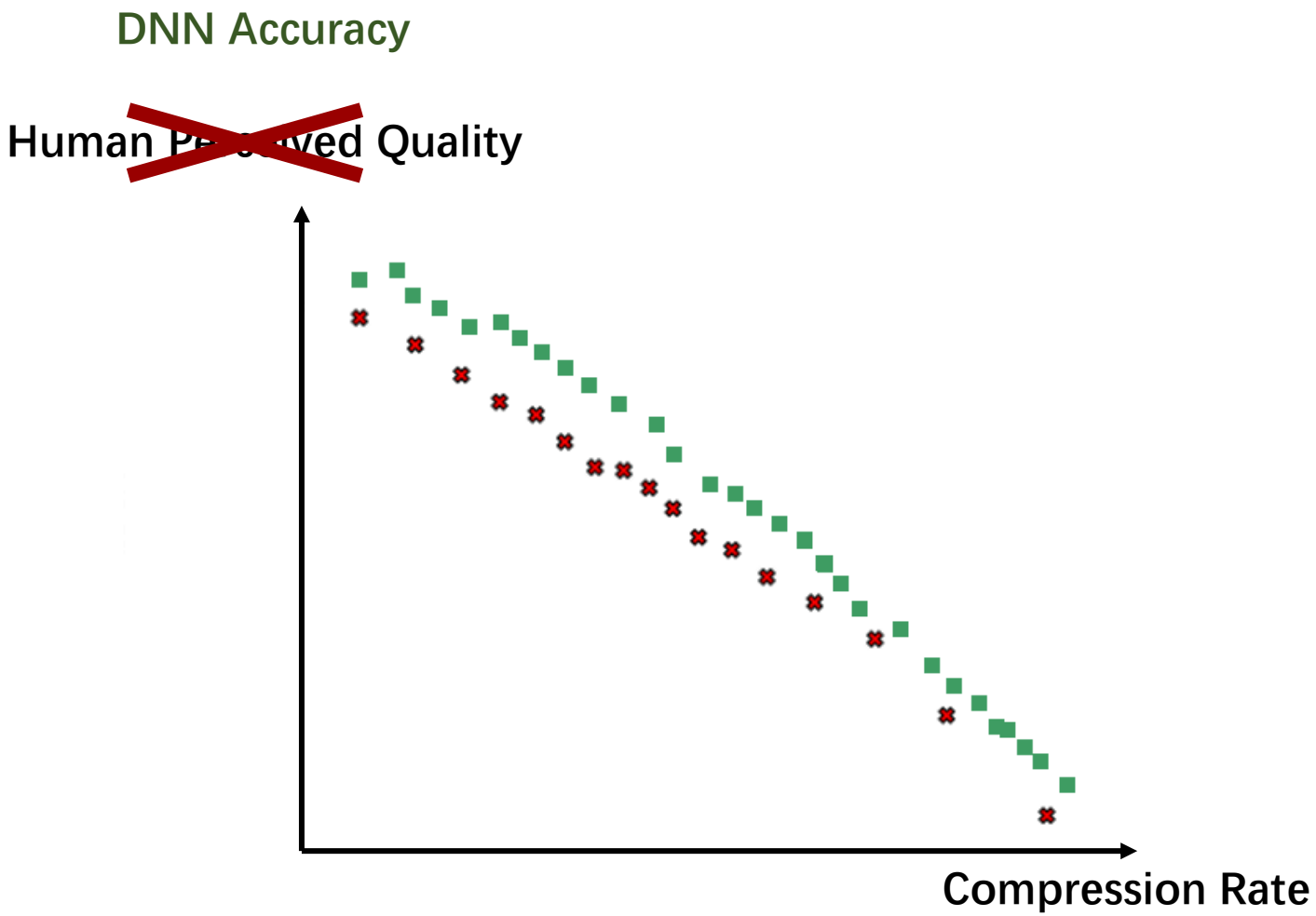


Why do we need to redesign JPEG?

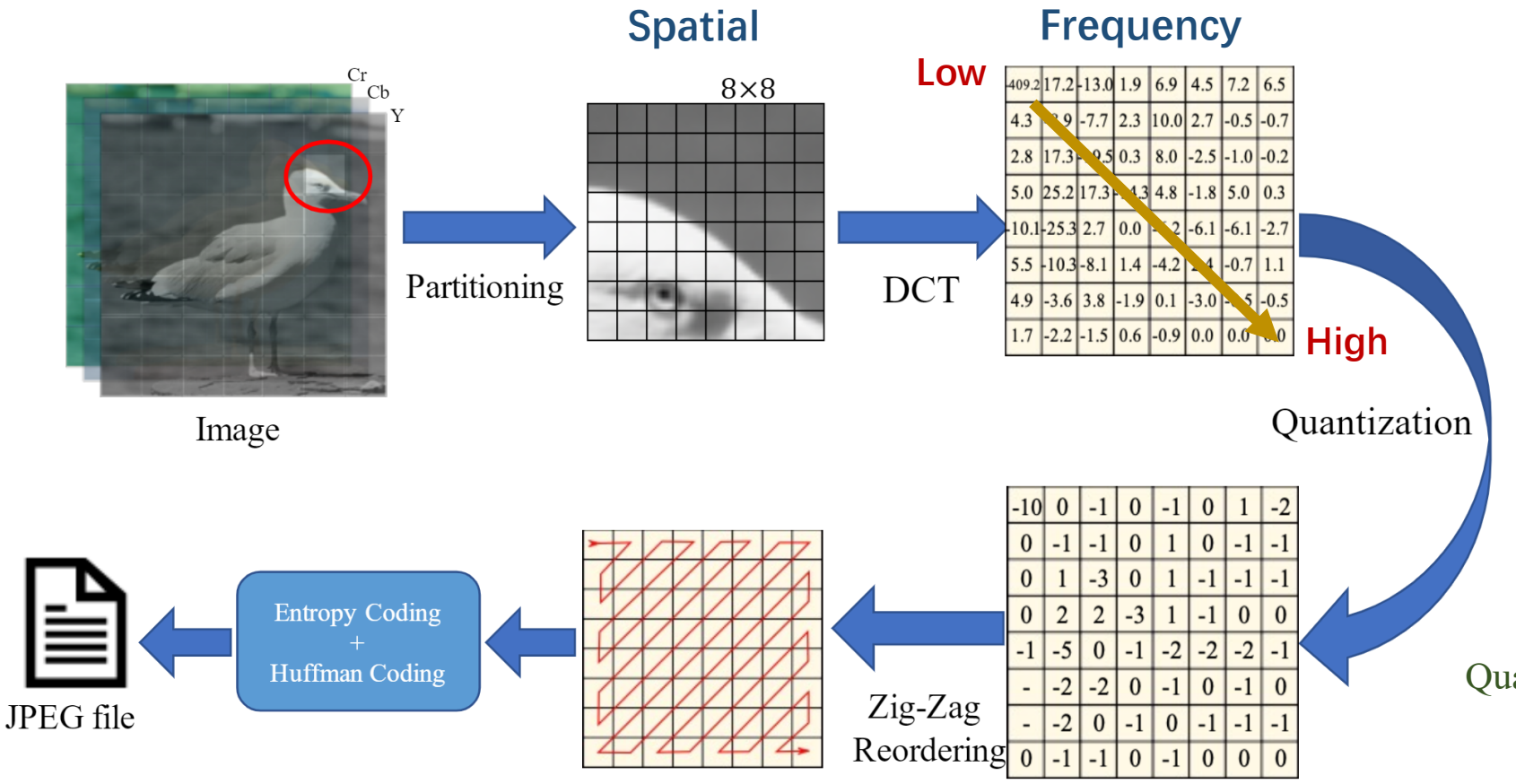
Original Image **76.2MB**

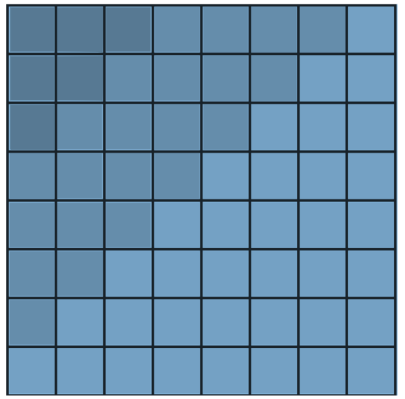
Compressed Image **1.4 MB**


Human Perceived Quality
Compression Rate


DNN Accuracy
Human Perceived Quality
Compression Rate

How does JPEG work?





Quantization Table (Q-table)

- Compression
  - round(409.2/40) = 10
- Decompression
  - 40 x 10 = 400
- 9.2 never recovered
- JPEG is lossy

Redesign Q-table for Classification DNN

Redesign Q-table for Classification DNN

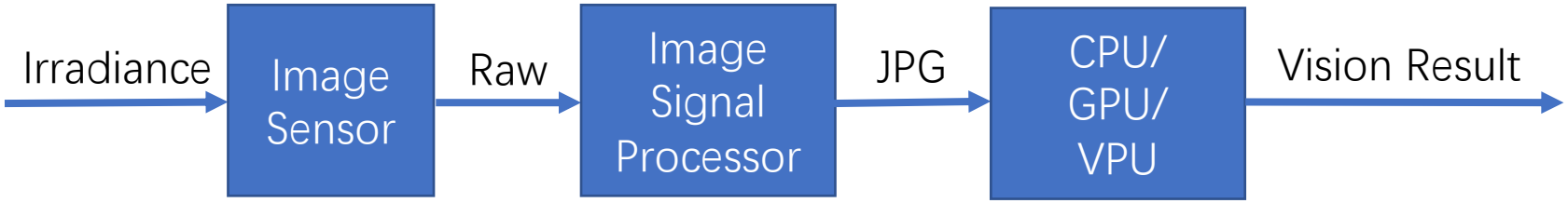
Existing Work

- Q-table optimization targets are different from DNN.
- DeepN-JPEG tunes and tests their Q-table on ImageNet.
  - No cross-validation
  - ImageNet is already compressed!

Redesign Q-table for Classification DNN

Construct Datasets

- Why not ImageNet?
  - Already downsized and after lossy compression



- Reconstruct high-resolution dataset
  - ImageNetV2 - 3 testing datasets with 1000 each
  - Id and url for images on Flickr
  - Simulate the effect of compressing raw pixels

ImageNet 2013 Val	482 x 415 pixels
ImageNetV2	1933 x 1592 pixels

Tuning Methodology

- Inferencing on pretrained ResNet
- Aiming best compression rate and accuracy
- Part of ImageNetV2

Matched-Frequncy

500 classes

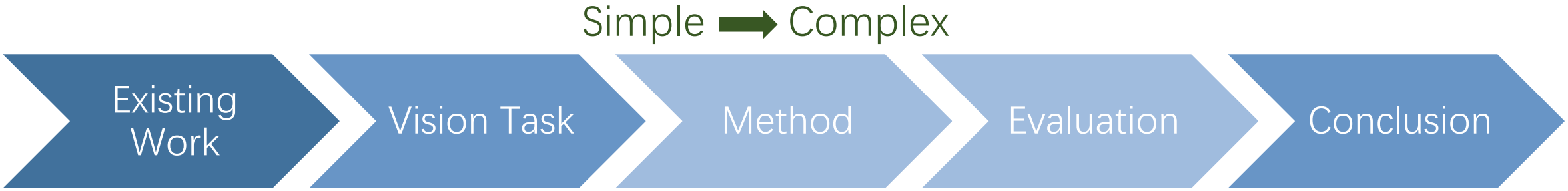
5 images

Threhold0.7

TopImages

- Speedup training
- Reserve for cross validation

Redesign Q-table for Classification DNN



Sorted Random Search

- How large is the search space for uniform random search?
  - $255^{64} = 1.04 \times 10^{154}$!
- Borrow the idea from standard JPEG!

Low Frequency
Large Value

-409.2	17.2	-13.0	1.9	6.9	4.5	7.2	6.5
4.3	-8.9	-7.7	2.3	10.0	2.7	-0.5	-0.7
2.8	17.3	-12.5	0.3	8.0	-2.5	-1.0	-0.2
5.0	25.2	17.3	-1.3	4.8	-1.8	5.0	0.3
-10.1	-25.3	2.7	0.0	-0.2	-6.1	-6.1	-2.7
5.5	-10.3	-8.1	1.4	-4.2	2.4	-0.7	1.1
4.9	-3.6	3.8	-1.9	0.1	-3.0	-0.5	-0.5
1.7	-2.2	-1.5	0.6	-0.9	0.0	0.0	0.0

High Frequency
Small Value

More Important

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

Less Important

Sorted Random Search

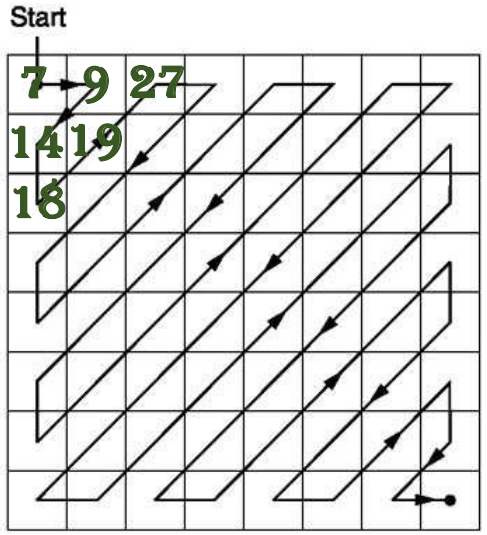
- How large is the search space for uniform random search?
  - $255^{64} = 1.04 \times 10^{154}$!
- Borrow the idea from standard JPEG!

7, 128, 75, 64, 9, 27, 189, ⋯

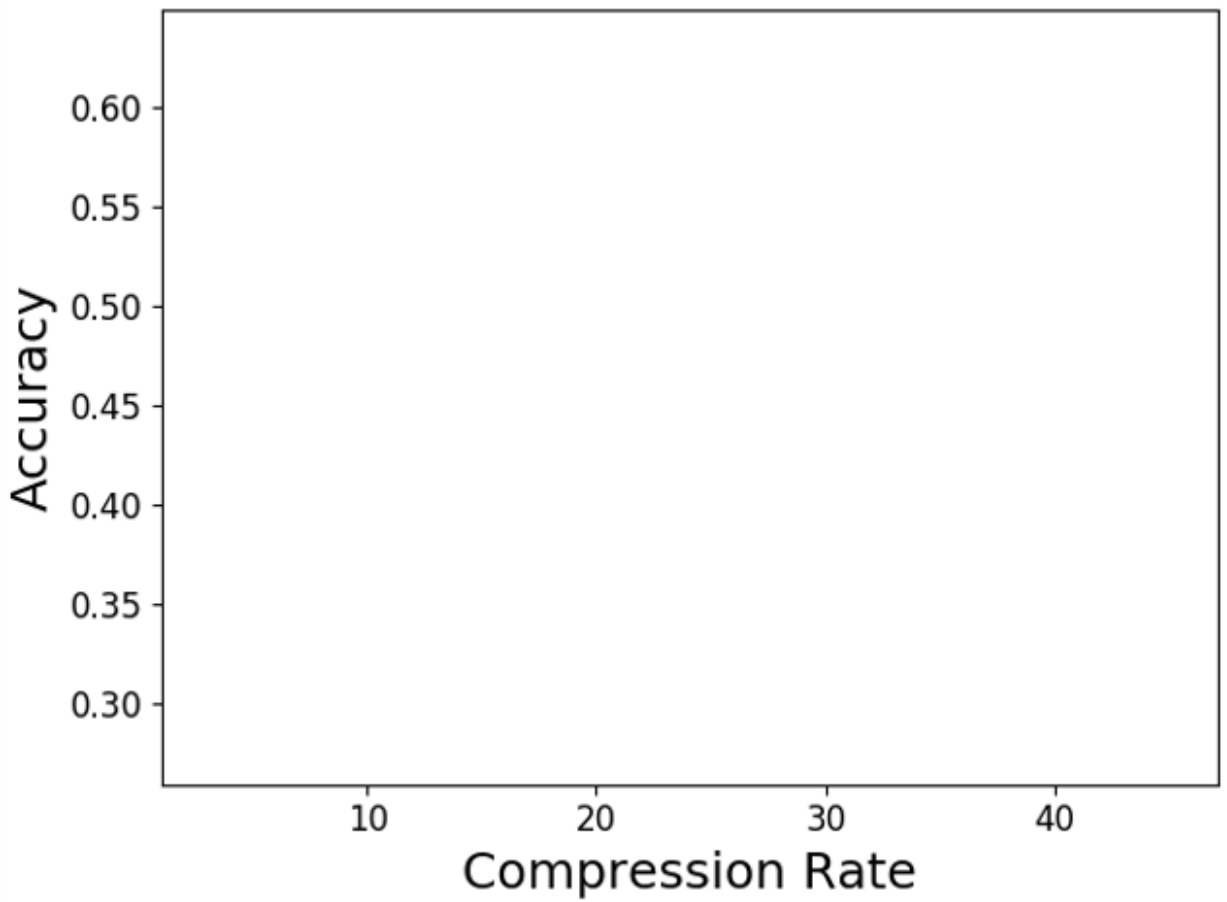
64 uniformly random numbers in [L, U]

7, 9, 14, 18, 19, 27, ⋯

64 sorted numbers

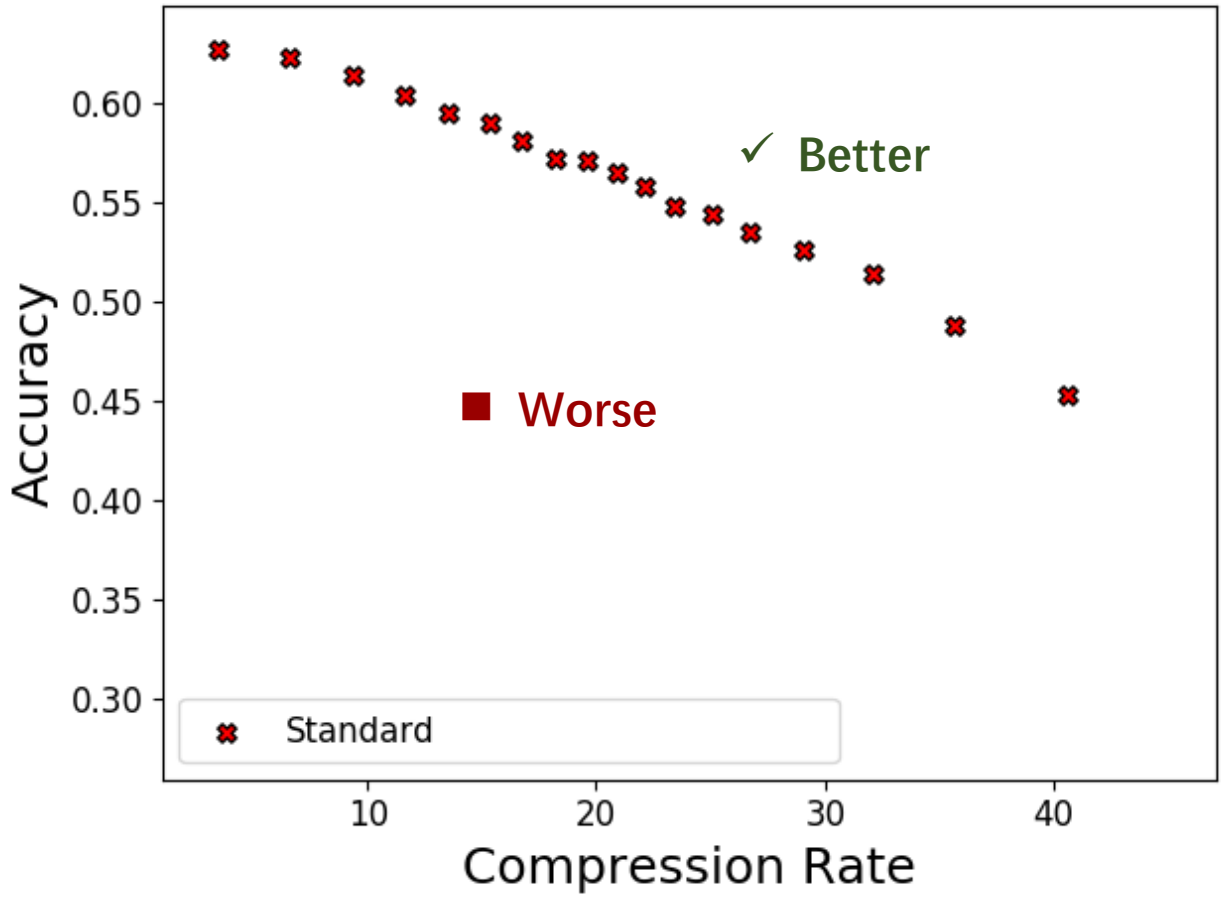


Sampling Result



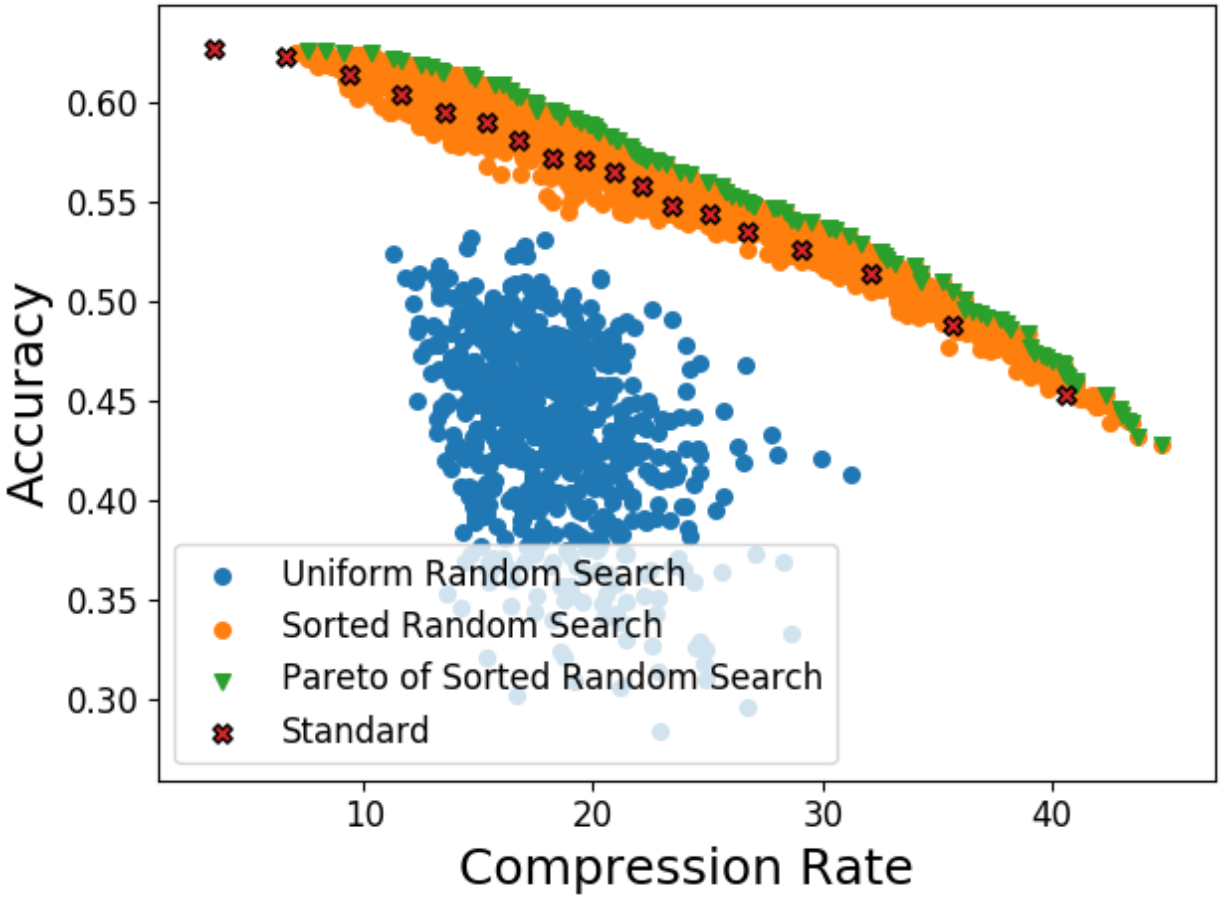
Sampling Result

- Take standard JPEG quality
  - 10, 15, …, 95



Sampling Result

- Take standard JPEG quality
  - 10, 15, …, 95



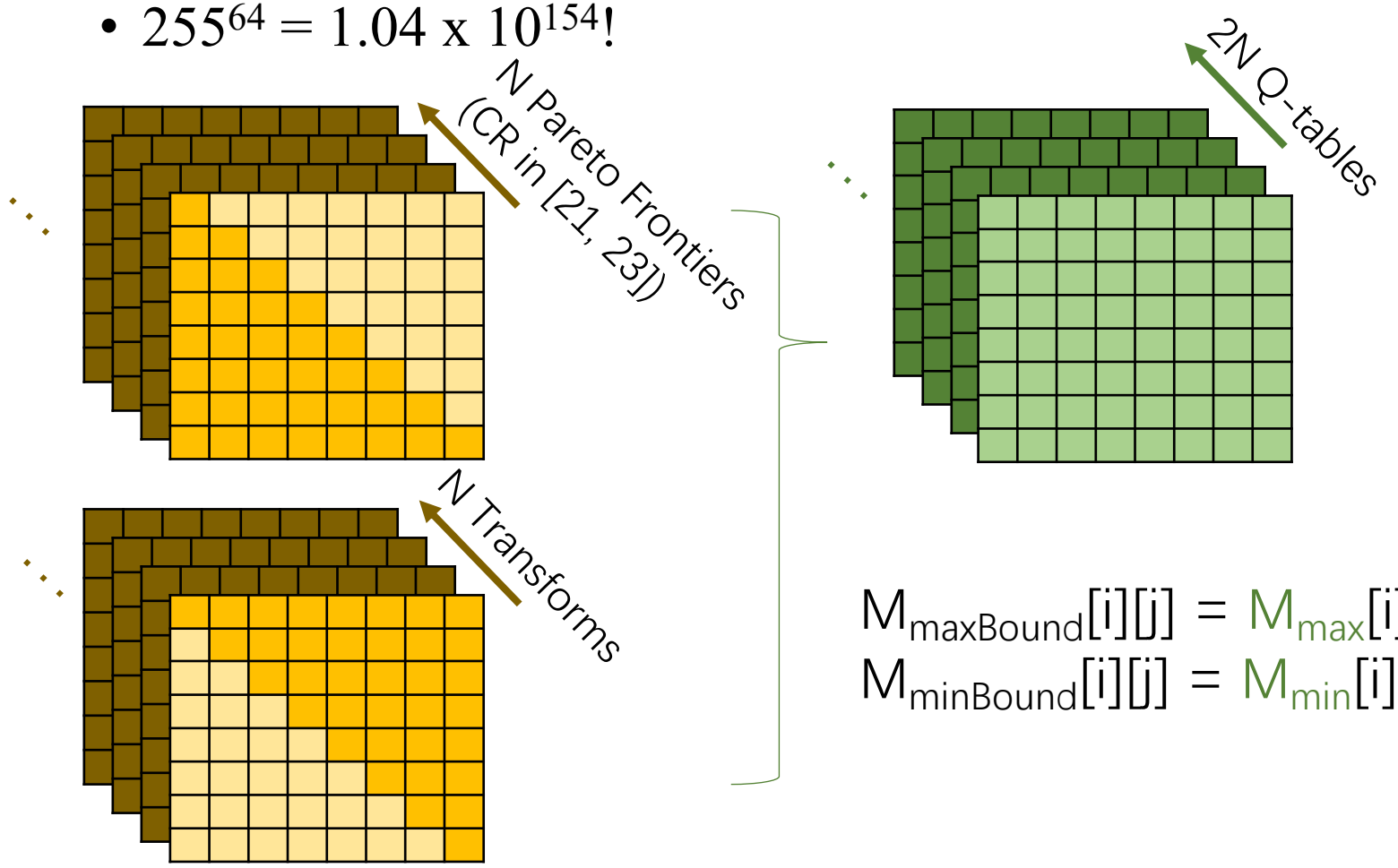
- Compression rate 10% - 200% better
- Accuracy improvement up to 2%

Sorted random search ✔

Can we do better?

Bounded Search

- How large is the search space for uniform random search?
  - $255^{64} = 1.04 \times 10^{154}$!



$$M_{maxBound}[i][j] = M_{max}[i][j] + 0.5\ M_{std}[i][j]$$
$$M_{minBound}[i][j] = M_{min}[i][j] - 0.5\ M_{std}[i][j]$$

Bounded Search

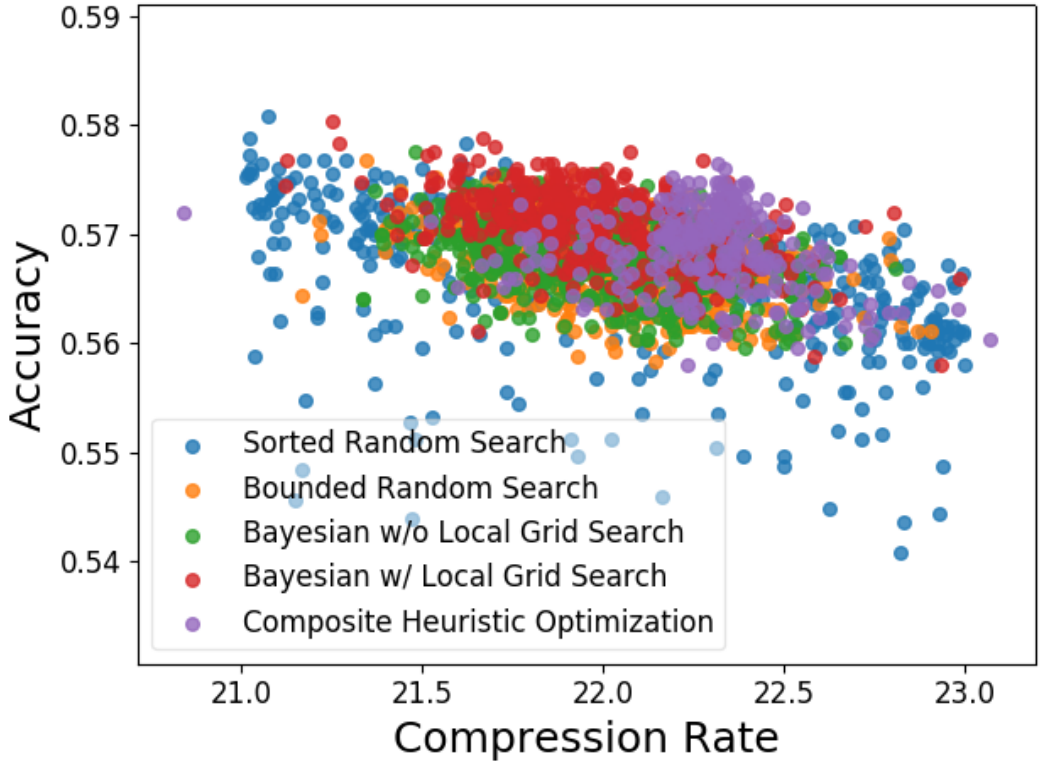
- Bounded Random Search
  - Uniformly sample in the bound
- Bayesian Optimization w/ local grid search
  - One objective
  - 5 indexes in the area of interest
  - Exhaustively apply the cheap acquisition function
- Composite Heuristic Optimization
  - One objective
  - OpenTuner using multi-armed bandit(MAB) approach
  - Swarm optimization, simulated annealing, differential evolution, greedy mutation and Nelder Mead as bandit arm

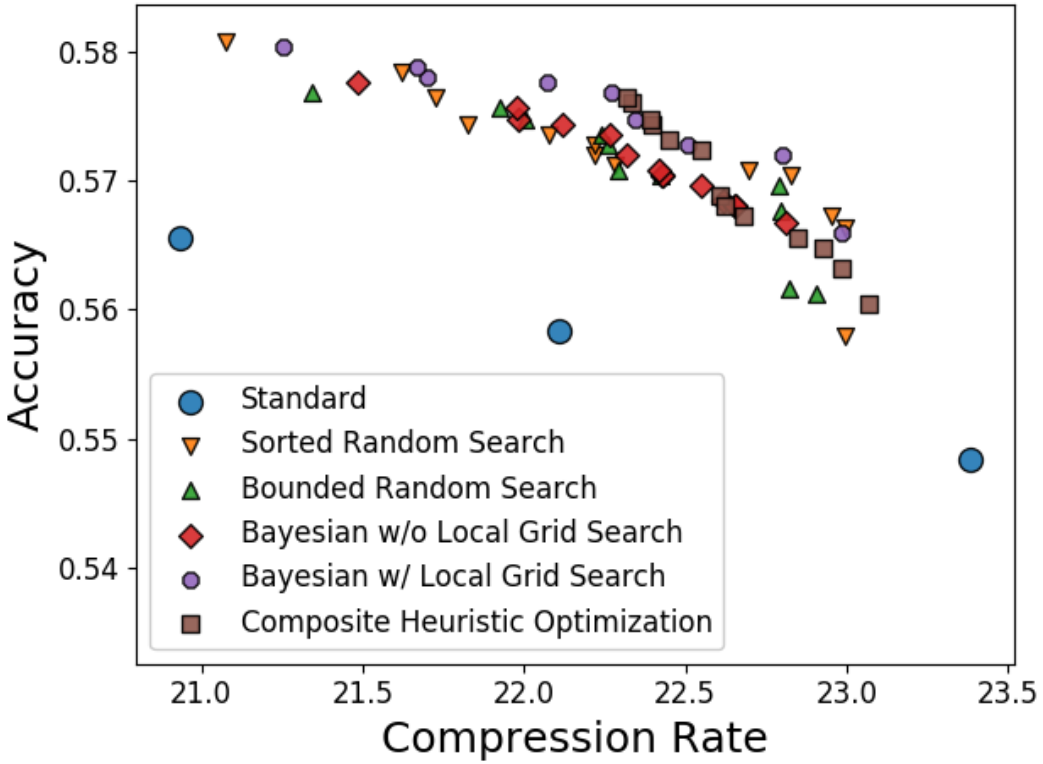
Fitness parabola:

$\text{fitness}(CR) = aCR^2 + bCR + c$

$\text{Objective} = Acc - \text{fitness}(CR)$

Sampling Results



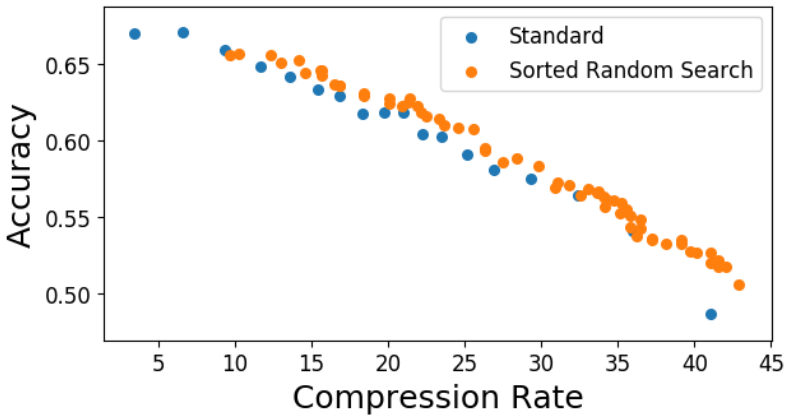


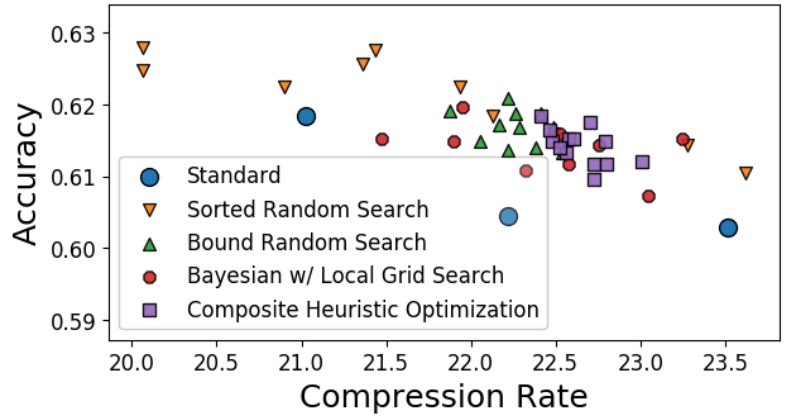
- Accuracy improvement up to 2.5%
- Composite heuristic optimization and Bayesian optimization w/ local grid search outperform others

Redesign Q-table for Classification DNN

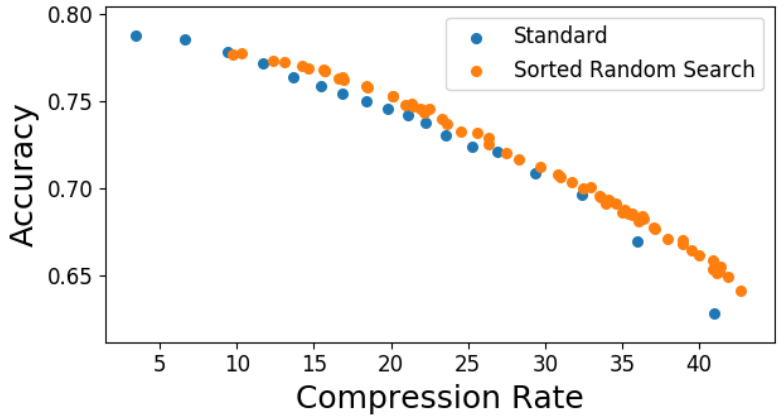
Cross Validation

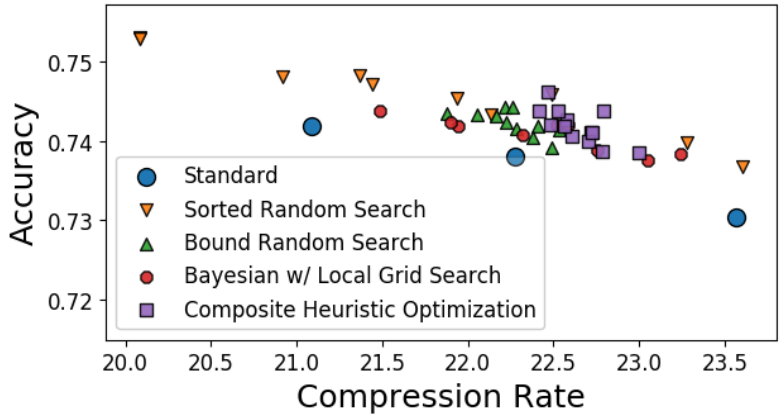
MatchedFreqency (other 500 classes)



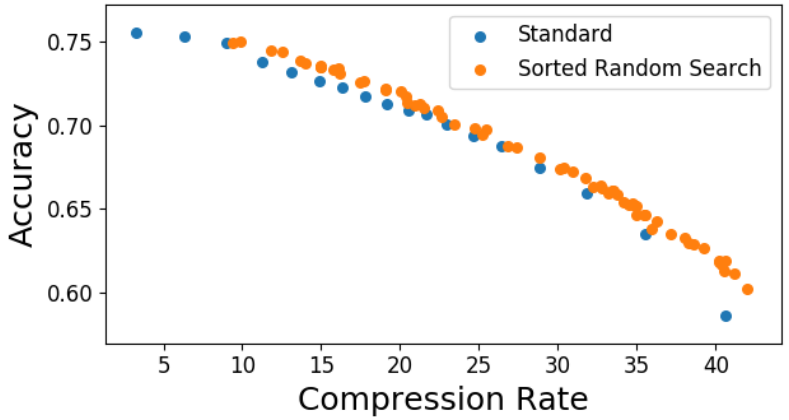


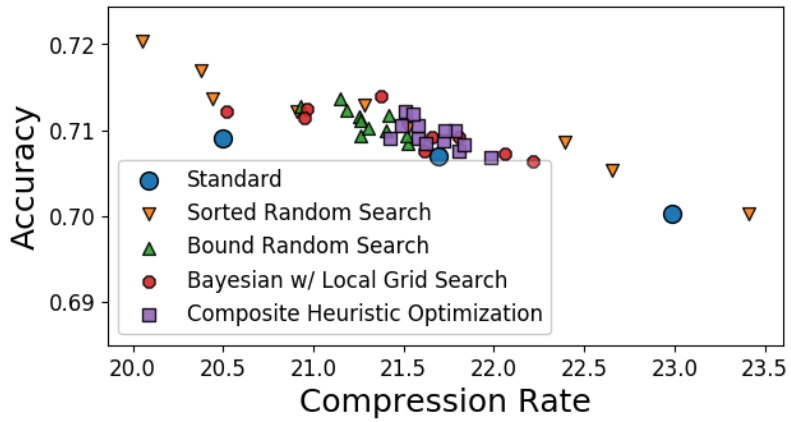
TopImages





ImageNet





- Improvement exists but decreases.
- The complex Bayesian and composite heuristic optimization no longer take the lead.

Significance of Improvement

- Is the improvement significant?

Method	MatchedFrequency	ImageNet
Sorted Random Search	0.91%**	1.16%**
Bounded Random Search	0.72%*	0.66%*
Bayesian Optimization	0.55%*	0.77%*
Composite Heuristic Optimization	0.73%*	1.17%**

* denotes $p < 10^{-5}$, ** denotes $p < 10^{-11}$

- Improvement sometimes is as small as 0.5%, but it is statistically significant.

Redesign Q-table for Classification DNN

To Tune JPEG for DNN Vision:

- Use sorted random search.
- It improves accuracy ~1% for the same storage size.
- The improvement remains under cross validation
- Not a fluke - the difference is statistically significant.

More to explore:

- DNN applications - detection, segmentation, etc.
- Retraining and finetuning - preliminary experiments give positive results!
- Quantization bits - 8 bits to 3 bits.