Do *et al.* (ECCV 2016), "Learning to Hash with Binary Deep Neural Network"

20183385 Huisu Yun

27 November 2018

CS688 Fall 2018 Student Presentation

Review: Song *et al.* (BMVC 2016)

- **Fine-grained** sketch-based image retrieval (SBIR)
  - Retrieval by fine-grained ranking (main task): **triplet ranking**
  - Attribute prediction (auxiliary task): predict **semantic attributes** that belong to sketches and images
  - Attribute-level ranking (another auxiliary task): **compare** attributes

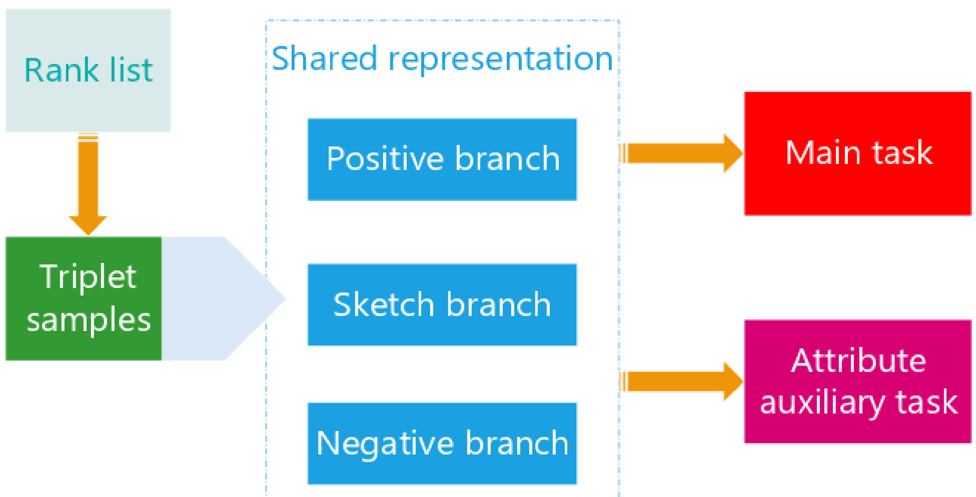


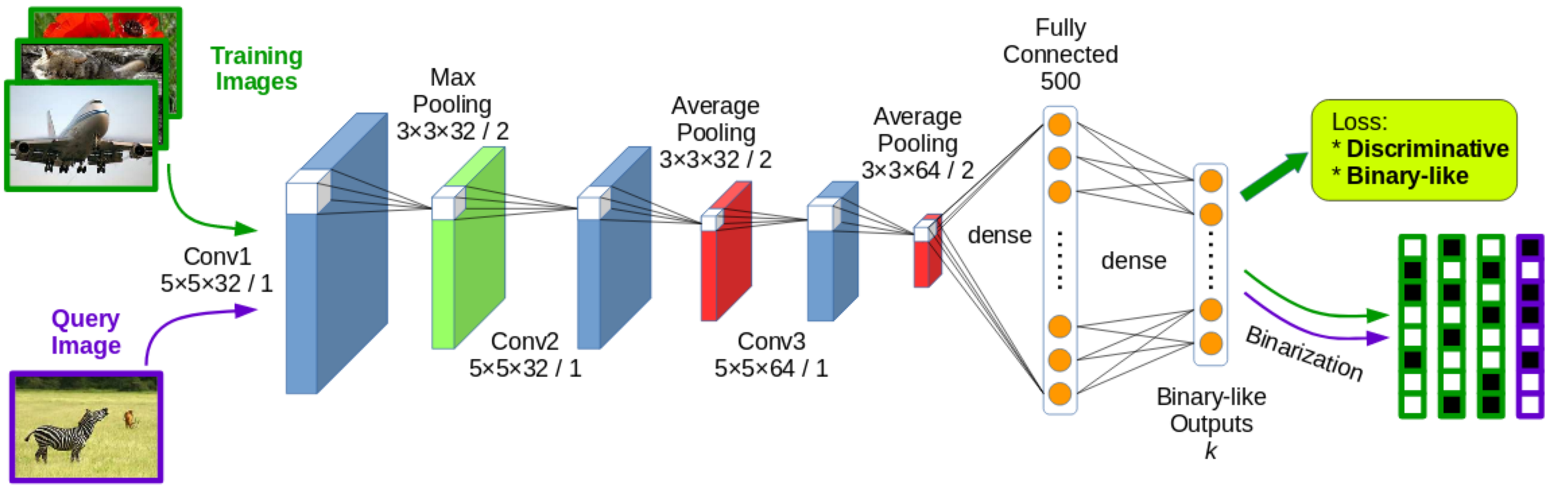
Image reproduced from Song et al. *2016. "Deep multi-task attribute-driven ranking for fine-grained sketch-based image retrieval"*

Motivation

- Raw feature vectors are **very long** (*cf.* PA2)
  - ...which is why we want to use specialized binary codes
- Binary codes for image search (*cf.* lecture slides)
  - ...should be of **reasonable length**
  - ...and provide **faithful representation**
- Important criteria
  - **Independence:** bits should be independent to each other
  - **Balance:** each bit should divide the dataset into equal halves

Background: Supervised codes (1/3)

- Liu *et al.* (CVPR 2016): **pairwise** supervision



Pairwise loss function

$$L_r(\mathbf{b}_1, \mathbf{b}_2, y) = \frac{1}{2}(1-y)||\mathbf{b}_1 - \mathbf{b}_2||_2^2$$

Similar images—similar codes

(Hamming distance approximated using Euclidean distance)

$$+\frac{1}{2}y\ \max(m - ||\mathbf{b}_1 - \mathbf{b}_2||_2^2,\ 0)$$

Dissimilar images—different codes

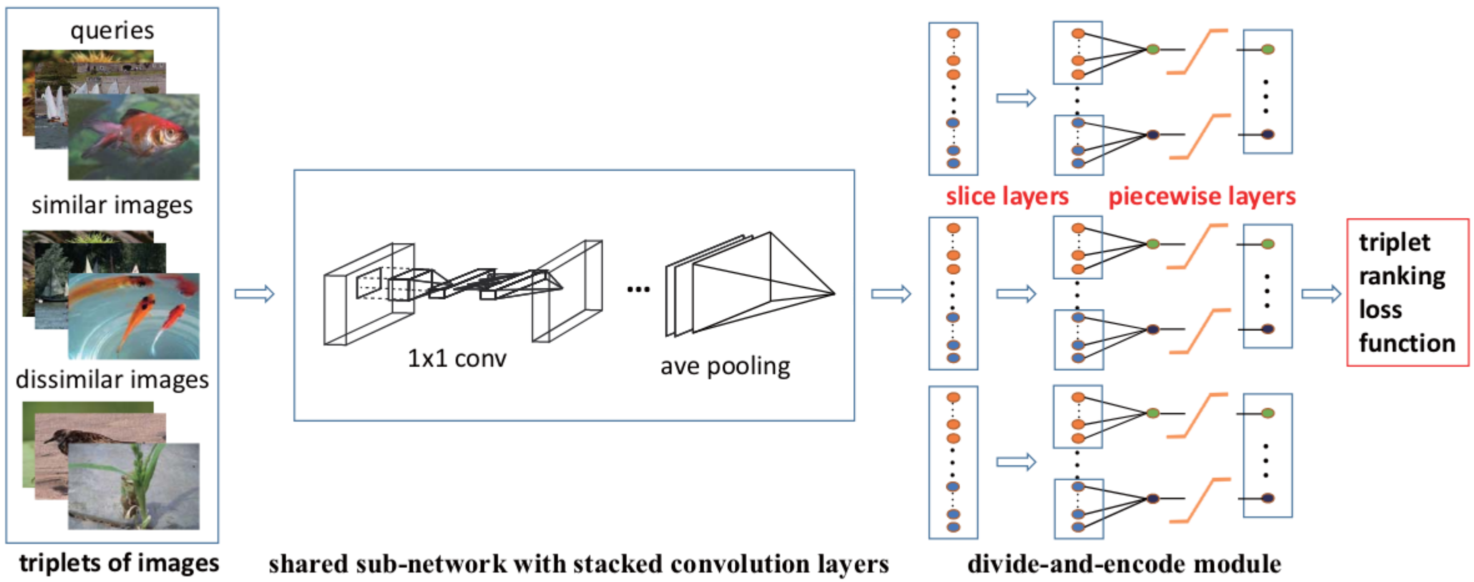
$$+\alpha(||\ |\mathbf{b}_1| - \mathbf{1}||_1 + ||\ |\mathbf{b}_2| - \mathbf{1}||_1)$$

Regularization (+1 or −1)

Image reproduced from Liu et al. *2016. "Deep supervised hashing for fast image retrieval"*

Background: Supervised codes (2/3)

- Lai *et al.* (CVPR 2015): **triplet** supervision



Triplet ranking loss

$$\ell_{triplet}(\mathcal{F}(I), \mathcal{F}(I^+), \mathcal{F}(I^-))$$
$$= \max(0, ||\mathcal{F}(I) - \mathcal{F}(I^+)||_2^2 - ||\mathcal{F}(I) - \mathcal{F}(I^-)||_2^2 + 1)$$
$$s.t.\ \mathcal{F}(I),\ \mathcal{F}(I^+),\ \mathcal{F}(I^-) \in [0,1]^q.$$

 Image reproduced from Lai et al. *2015. "Simultaneous feature learning and hash coding with deep neural networks"*

Background: Supervised codes (3/3)

- Jain *et al.* (ICCV 2017): point-wise supervision, **quantized output**

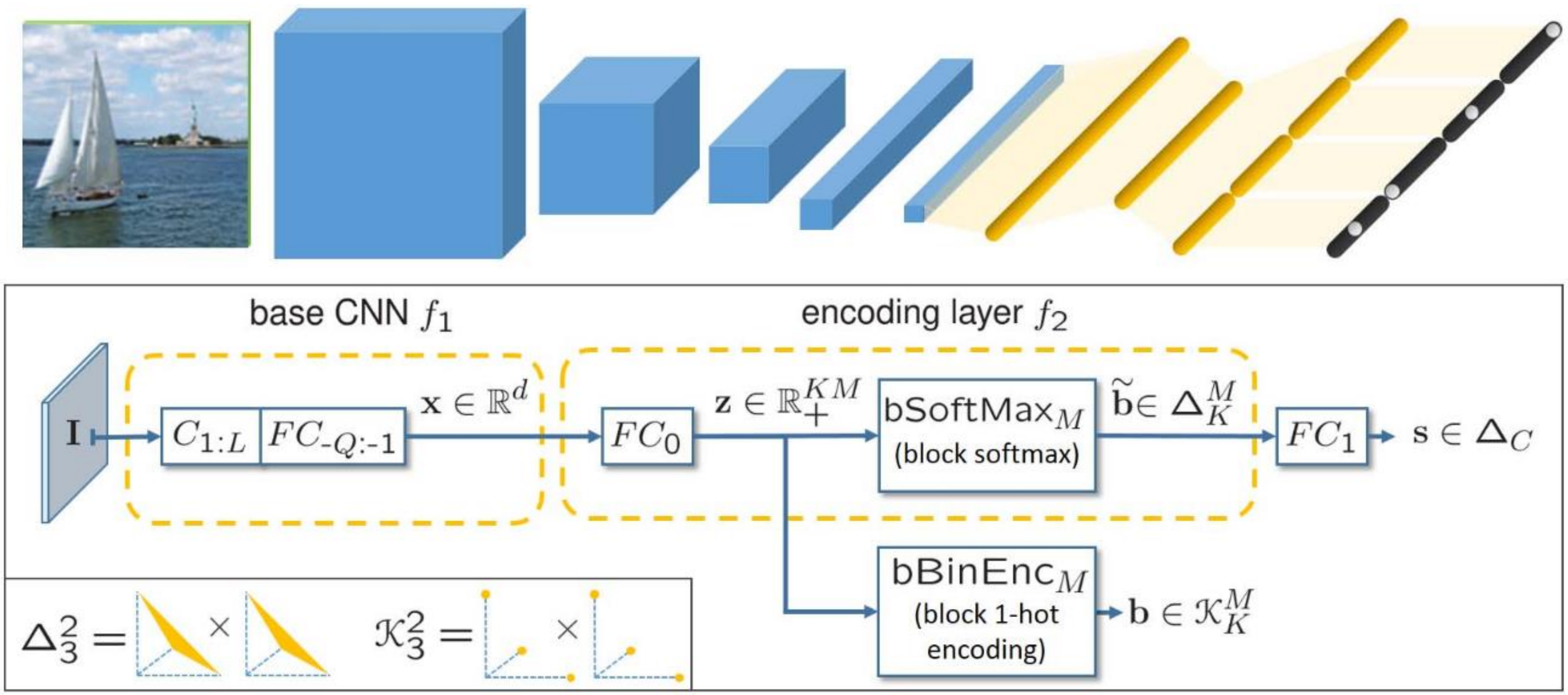


 Image reproduced from Jain et al. *2017. "SuBiC: A supervised, structured binary code for image search"*

Background: Deep Hashing

- Liong *et al.* (CVPR 2015)
  - Fully connected layers
  - Binary hash code **B** is constructed from the output value of the last layer, $\mathbf{H}^{(n)}$, as follows: $\mathbf{B} = \text{sgn}\, \mathbf{H}^{(n)}$
  - Note that "binary" means ± 1 here

Quantization loss *Balance loss*

$$\min_{\mathbf{W},\mathbf{c}} J = \frac{1}{2}\left\| sgn(\mathbf{H}^{(n)}) - \mathbf{H}^{(n)} \right\|^2 - \frac{\alpha_1}{2m} tr\left(\mathbf{H}^{(n)}(\mathbf{H}^{(n)})^T\right)$$

$$+ \frac{\alpha_2}{2}\sum_{l=1}^{n-1}\left\|\mathbf{W}^{(l)}(\mathbf{W}^{(l)})^T - \mathbf{I}\right\|^2 + \frac{\alpha_3}{2}\sum_{l=1}^{n-1}\left(\left\|\mathbf{W}^{(l)}\right\|^2 + \left\|\mathbf{c}^{(l)}\right\|^2\right)$$

Independence loss *Regularization loss*

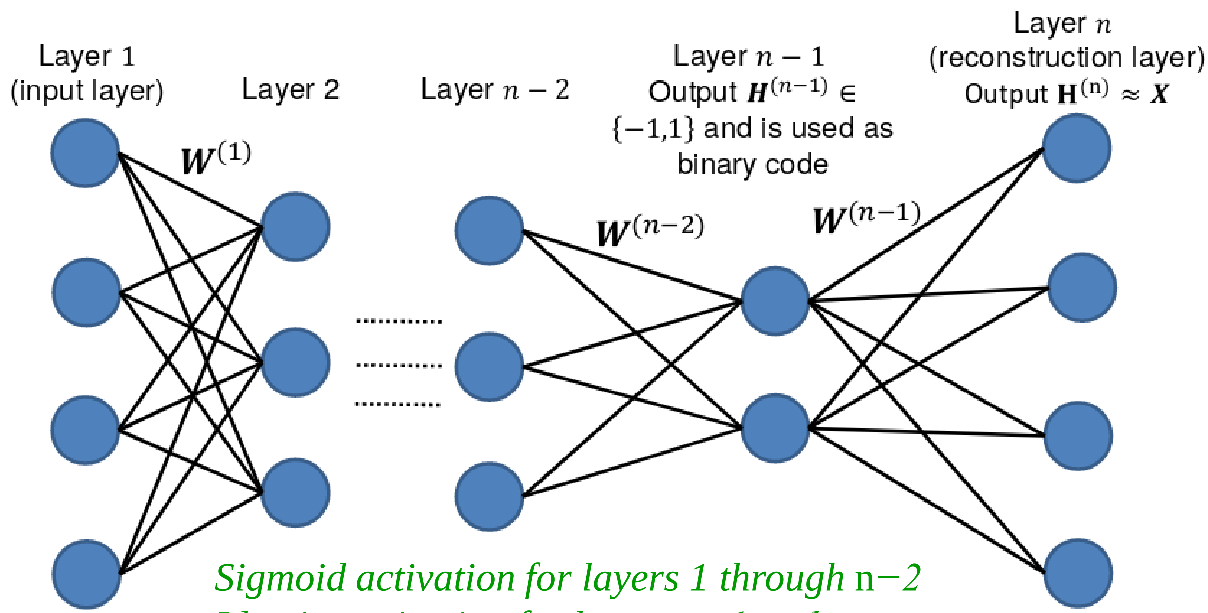
Introduction

- **Binary** Deep Neural Network (BDNN)
  - **Real** binary codes *(how?)*
  - **Real** independence loss *(not relaxed/approximated)*
  - **Real** balance loss *(again, not relaxed/approximated)*
  - **Reconstruction loss** *(like autoencoders!)*

- **Unsupervised** (UH-) and **supervised** (SH-) variants

Overview

- "Unsupervised Hashing with BDNN (UH-BDNN)"



Sigmoid activation for layers 1 through n−*2*
Identity activation for layers n−*1 and* n

$$\min_{\mathbf{W},\mathbf{c},\mathbf{B}} J = \frac{1}{2m}\left\|\mathbf{X} - \mathbf{W}^{(n-1)}\mathbf{B} - \mathbf{c}^{(n-1)}\mathbf{1}_{1\times m}\right\|^2 + \frac{\lambda_1}{2}\sum_{l=1}^{n-1}\left\|\mathbf{W}^{(l)}\right\|^2$$

$$+\frac{\lambda_2}{2m}\left\|\mathbf{H}^{(n-1)} - \mathbf{B}\right\|^2 + \frac{\lambda_3}{2}\left\|\frac{1}{m}\mathbf{H}^{(n-1)}(\mathbf{H}^{(n-1)})^T - \mathbf{I}\right\|^2 + \frac{\lambda_4}{2m}\left\|\mathbf{H}^{(n-1)}\mathbf{1}_{m\times 1}\right\|^2$$

 Image reproduced from Do et al. *2016. "Learning to Hash with Binary Deep Neural Network"*

Optimization

- Alternating optimization with respect to (**W**, **c**) and **B**
  - Network parameters (weight $\mathbf{W}^{(\cdot)}$, bias $\mathbf{c}^{(\cdot)}$) using **L-BFGS**
  - Binary code (**B**) using **discrete cyclic coordinate descent**
- Note that, ideally, $\mathbf{H}^{(n-1)}$ should be equal to **B**

Reconstruction loss *Regularization loss*

$$\min_{\mathbf{W},\mathbf{c},\mathbf{B}} J = \frac{1}{2m}\left\|\mathbf{X} - \mathbf{W}^{(n-1)}\mathbf{B} - \mathbf{c}^{(n-1)}\mathbf{1}_{1\times m}\right\|^2 + \frac{\lambda_1}{2}\sum_{l=1}^{n-1}\left\|\mathbf{W}^{(l)}\right\|^2$$

$$+\frac{\lambda_2}{2m}\left\|\mathbf{H}^{(n-1)} - \mathbf{B}\right\|^2 + \frac{\lambda_3}{2}\left\|\frac{1}{m}\mathbf{H}^{(n-1)}(\mathbf{H}^{(n-1)})^T - \mathbf{I}\right\|^2 + \frac{\lambda_4}{2m}\left\|\mathbf{H}^{(n-1)}\mathbf{1}_{m\times 1}\right\|^2$$

Equality loss *Independence loss* *Balance loss*

$$\text{s.t. } \mathbf{B} \in \{-1, 1\}^{L\times m}$$

Deep Hashing vs. UH-BDNN

Quantization loss *Balance loss*

$$\min_{\mathbf{W},\mathbf{c}} J = \frac{1}{2}\left\|sgn(\mathbf{H}^{(n)}) - \mathbf{H}^{(n)}\right\|^2 - \frac{\alpha_1}{2m} tr\left(\mathbf{H}^{(n)}(\mathbf{H}^{(n)})^T\right)$$

$$+\frac{\alpha_2}{2}\sum_{l=1}^{n-1}\left\|\mathbf{W}^{(l)}(\mathbf{W}^{(l)})^T - \mathbf{I}\right\|^2 + \frac{\alpha_3}{2}\sum_{l=1}^{n-1}\left(\left\|\mathbf{W}^{(l)}\right\|^2 + \left\|\mathbf{c}^{(l)}\right\|^2\right)$$

Independence loss *Regularization loss*

Reconstruction loss *Regularization loss*

$$\min_{\mathbf{W},\mathbf{c},\mathbf{B}} J = \frac{1}{2m}\left\|\mathbf{X} - \mathbf{W}^{(n-1)}\mathbf{B} - \mathbf{c}^{(n-1)}\mathbf{1}_{1\times m}\right\|^2 + \frac{\lambda_1}{2}\sum_{l=1}^{n-1}\left\|\mathbf{W}^{(l)}\right\|^2$$

$$+\frac{\lambda_2}{2m}\left\|\mathbf{H}^{(n-1)} - \mathbf{B}\right\|^2 + \frac{\lambda_3}{2}\left\|\frac{1}{m}\mathbf{H}^{(n-1)}(\mathbf{H}^{(n-1)})^T - \mathbf{I}\right\|^2 + \frac{\lambda_4}{2m}\left\|\mathbf{H}^{(n-1)}\mathbf{1}_{m\times 1}\right\|^2$$

Equality loss *Independence loss* *Balance loss*

$$\text{s.t. } \mathbf{B} \in \{-1, 1\}^{L\times m}$$

Using class labels

- "Supervised Hashing with BDNN (SH-BDNN)"
  - No reconstruction layer
  - Uses pairwise label matrix $\mathbf{S}_{ij} = \begin{cases} 1 & \text{if } \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ are same class} \\ -1 & \text{if } \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ are not same class} \end{cases}$
- Hamming distance between binary codes should correlate with the pairwise label matrix **S**

Classification loss *Regularization loss*

$$\min_{\mathbf{W},\mathbf{c},\mathbf{B}} J = \frac{1}{2m}\left\|\frac{1}{L}(\mathbf{H}^{(n)})^T\mathbf{H}^{(n)} - \mathbf{S}\right\|^2 + \frac{\lambda_1}{2}\sum_{l=1}^{n-1}\left\|\mathbf{W}^{(l)}\right\|^2 + \frac{\lambda_2}{2m}\left\|\mathbf{H}^{(n)} - \mathbf{B}\right\|^2$$

Equality loss

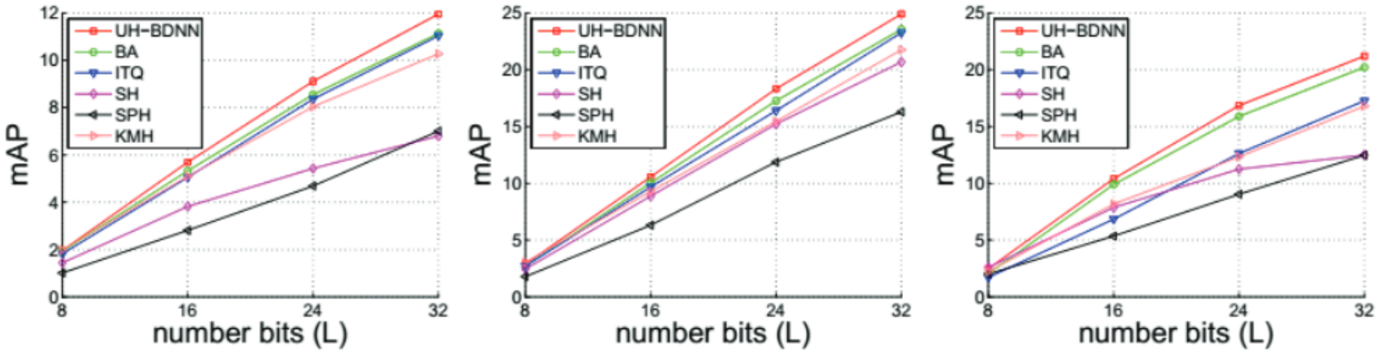
$$+\frac{\lambda_3}{2}\left\|\frac{1}{m}\mathbf{H}^{(n)}(\mathbf{H}^{(n)})^T - \mathbf{I}\right\|^2 + \frac{\lambda_4}{2m}\left\|\mathbf{H}^{(n)}\mathbf{1}_{m\times 1}\right\|^2$$

Independence loss *Balance loss*

$$\text{s.t. } \mathbf{B} \in \{-1, 1\}^{L\times m}$$

Results (1/2)

Evaluation of Unsupervised Hashing



(a) CIFAR10 (b) MNIST (c) SIFT1M

Figure 1: mAP comparison between UH-BDNN and the state of the art.

	CIFAR10				MNIST				SIFT1M			
L	8	16	24	32	8	16	24	32	8	16	24	32
UH-BDNN	0.55	5.79	22.14	18.35	0.53	6.80	29.38	38.50	4.80	25.20	62.20	80.55
BA	0.55	5.65	20.23	17.00	0.51	6.44	27.65	35.29	3.85	23.19	61.35	77.15
ITQ	0.54	5.05	18.82	17.76	0.51	5.87	23.92	36.35	3.19	14.07	35.80	58.69
SH	0.39	4.23	14.60	15.22	0.43	6.50	27.08	36.69	4.67	24.82	60.25	72.40
SPH	0.43	3.45	13.47	13.67	0.44	5.02	22.24	30.80	4.25	20.98	47.09	66.42
KMH	0.53	5.49	19.55	15.90	0.50	6.36	25.68	36.24	3.74	20.74	48.86	76.04

Table 1: Precision at Hamming distance $r = 2$ comparison between UH-BDNN and the state of the art.

Image reproduced from Do et al. *2016. "Learning to Hash with Binary Deep Neural Network"*

Results (2/2)

Evaluation of Supervised Hashing

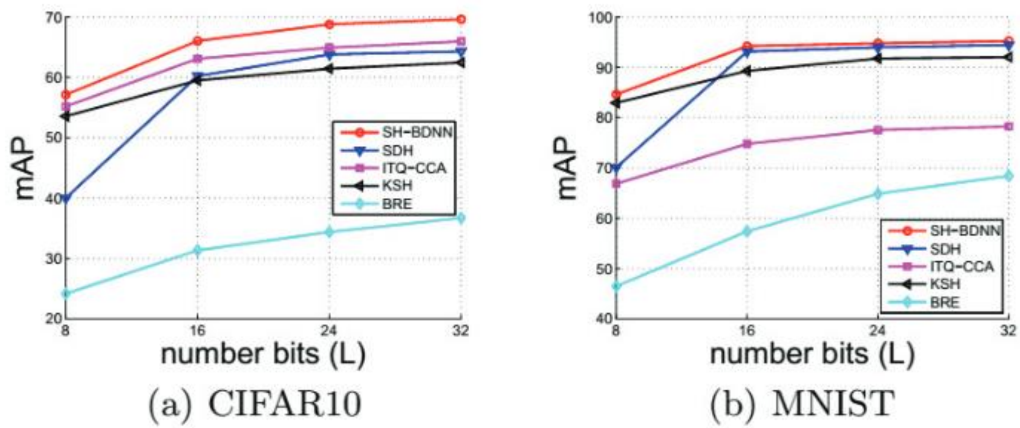


Figure 2: mAP comparison between SH-BDNN and the state of the art.

	CIFAR10				MNIST			
L	8	16	24	32	8	16	24	32
SH-BDNN	54.12	67.32	69.36	69.62	84.26	94.67	94.69	95.51
SDH	31.60	62.23	67.65	67.63	36.49	93.00	93.98	94.43
ITQ-CCA	49.14	65.68	67.47	67.19	54.35	79.99	84.12	84.57
KSH	44.81	64.08	67.01	65.76	68.07	90.79	92.86	92.41
BRE	23.84	41.11	47.98	44.89	37.67	69.80	83.24	84.61

Table 2: Precision at Hamming distance $r = 2$ comparison between SH-BDNN and the state of the art.

Image reproduced from Do et al. *2016. "Learning to Hash with Binary Deep Neural Network"*

Discussion

- The framework's capability of generating both unsupervised and supervised binary codes using nearly identical architectures would be useful for many applications

- The fact that the optimization algorithms used in BDNN (especially L-BFGS) do not fully benefit from the amount of parallelism available on modern machines might result in suboptimal utilization of computing resources