DEEP MULRI-REGION HASHING

Quan Zhou, Xiushan Nie*, Yang Shi,
Xingbo Liu, Yilong Yin*

Email: woodschou@outlook.com

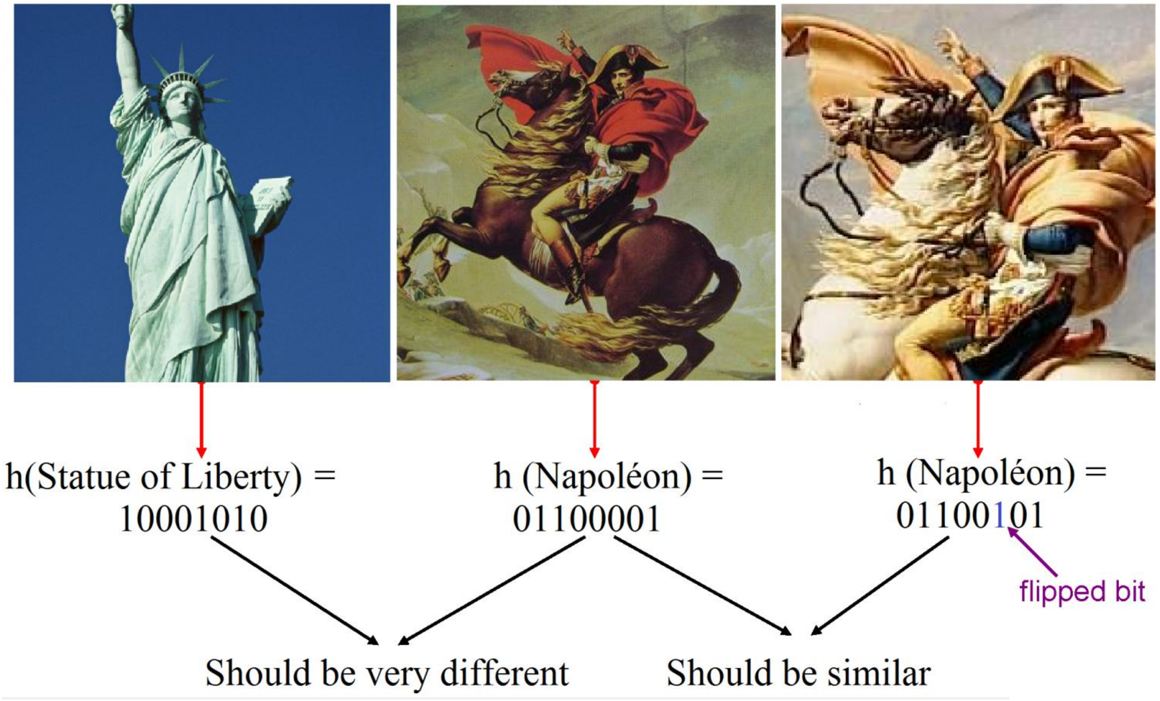
Outline

- Hashing
- DMRH
- Experiments
- Summary

Outline

- Hashing
- DMRH
- Experiments
- Summary

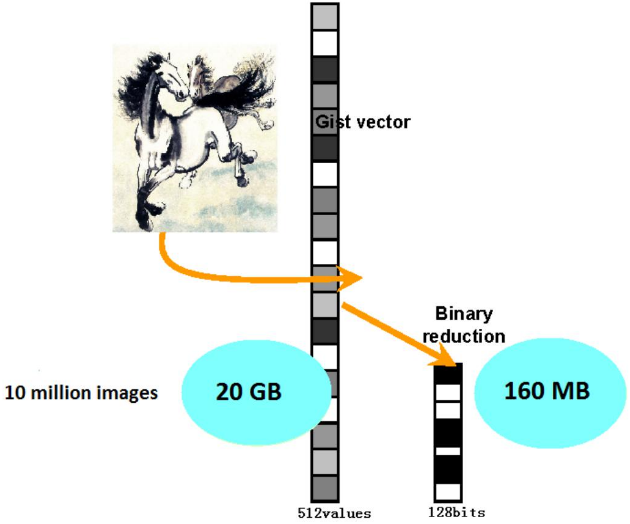
Similarity Preserving Hashing



[Wu-Jun Li , 2015]

Advantages

- Reduce Dimensionality and Storage Cost
- Fast Query Speed



[Wu-Jun Li , 2015]

Outline

- Hashing
- DMRH
- Experiments
- Summary

Deep Multi-Region Hashing

- Motivation
- Network
- Formulation
- Out-of-sample extension

Motivation

- In the existing hashing methods, **deep supervised hashing** methods have achieved the best performance by utilizing the semantic labels on data with deep learning.

- Most of deep methods only consider the semantics of the **whole image** but **ignore the local information** which contains much more semantic details. Evidently, the sematic details are beneficial for hash learning.

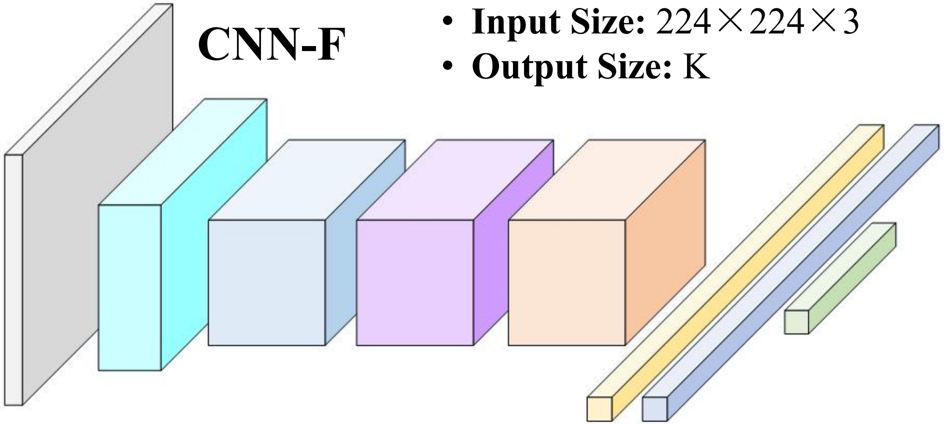
- Fusing local information includes two ways:
  a. Fusing feature of regions
  b. Fusing discriminants (i.e. hash codes) of regions

- There are several deep hashing methods which utilize the local information by the former way, but no one adopts the latter way.

Network

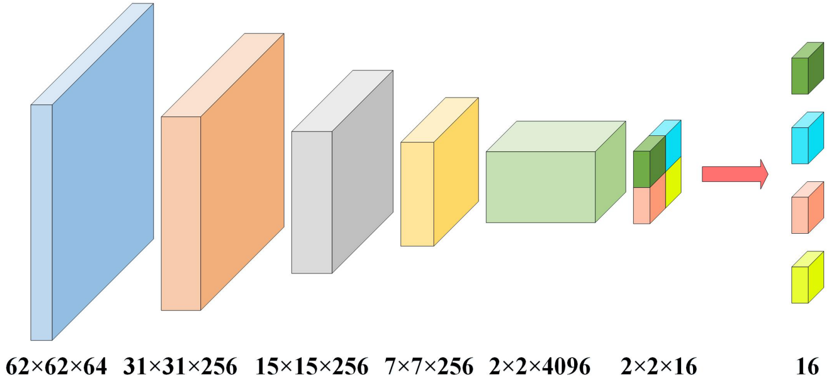
CNN-F

- **Input Size:** 224×224×3
- **Output Size:** K



[Chatfield et al. , 2014]

Network



N=2

Layer	CNN-F	Modified CNN-F
Layer6	FC-4096	Conv6-4096
Layer7	FC-4096	Conv1-4096
Layer8	FC-K	Conv1-K

- **Input Size:** $(192+32*N)\times(192+32*N)\times 3$
- **Output Size:** $N^2\times K$

[Isola et al. , 2017]

Formulation

- Category Label information:
  - $Y = \{\boldsymbol{y}_i\}_{i=1}^{N} \in \mathbb{R}^{c\times N}, \boldsymbol{y}_i \in \{0,1\}^c$
- Pairwise label information:
  - $\boldsymbol{S} \in \{0,1\}^{N\times N}$
- Hamming Distance ***$dist_H(\cdot,\cdot)$*** with Inner Product **$<\cdot,\cdot>$**:
  - $dist_H(\boldsymbol{b}_i,\boldsymbol{b}_j) = \frac{1}{2}(K- <\boldsymbol{b}_i,\boldsymbol{b}_j>)$
- The formulation of DPSH:
  - $p(\boldsymbol{B}|\boldsymbol{S}) \propto p(\boldsymbol{S}|\boldsymbol{B})p(\boldsymbol{B}) = \prod_{s_{ij}\in\boldsymbol{S}} p(s_{ij}|\boldsymbol{B})p(\boldsymbol{B})$
  - $p(s_{ij}|\boldsymbol{B}) = \begin{cases} \sigma(\Phi_{ij}), & s_{ij}=1 \\ 1-\sigma(\Phi_{ij}), & s_{ij}=0 \end{cases}, \Phi_{ij} =<\boldsymbol{b}_i,\boldsymbol{b}_j>= \frac{1}{2}\boldsymbol{b}_i^T\boldsymbol{b}_j$

[Liu et al. , 2012][Li et al. , 2016]

Formulation

- The formulation of DPSH:
  - $J = -\log p(\boldsymbol{S}|\boldsymbol{B}) = -\sum_{s_{ij}\in\boldsymbol{S}} (s_{ij}\Phi_{ij} - log(1+e^{\Phi_{ij}}))$
- Our formulation:
  - $J = -\frac{1}{n^2}\sum_{s_{ij}\in\boldsymbol{S}} (s_{ij}\Theta_{ij} - log(1+e^{\Theta_{ij}})) + \frac{\eta}{n\times K}\sum_{i=1}^{n}\|\boldsymbol{b}_i - \boldsymbol{h}_i\|_2^2$
  - $\Theta_{ij} =< \boldsymbol{h}_i, \boldsymbol{h}_j >= \frac{1}{2}\boldsymbol{h}_i^T\boldsymbol{h}_j$

Notation	Description
$\boldsymbol{x}_i$	The i_{th} image
n	The number of images
K	The length of the hash code
s_{ij}	Similarity between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$
$\boldsymbol{b}_i$	The hash code of image $\boldsymbol{x}_i$
$\boldsymbol{h}_i$	The relaxed hash code of image $\boldsymbol{x}_i$
$\boldsymbol{\eta}$	Hyper-parameter

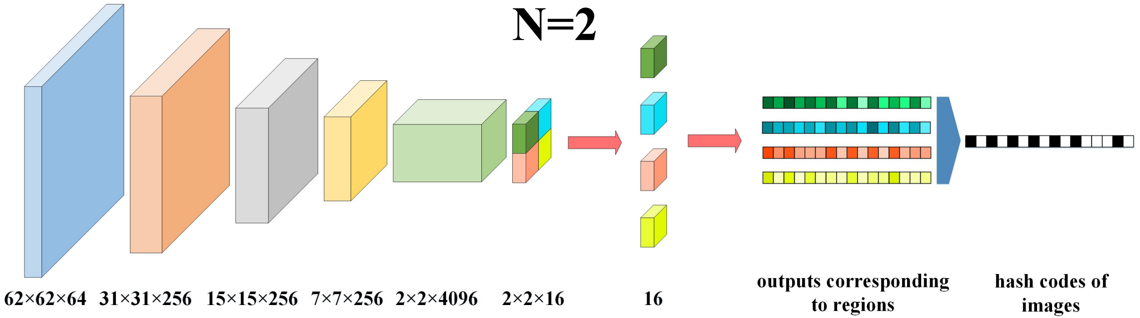
[Li et al. , 2016]

Formulation

Notation	Description
$\boldsymbol{x}_i$	The i_{th} image
$\boldsymbol{x}_{ik}$	The k_{th} region of image $\boldsymbol{x}_i$
N^2	The number of regions from an image
$\boldsymbol{h}_i$	The relaxed hash code of image $\boldsymbol{x}_i$
$\boldsymbol{h}_{ik}$	The network output when input region $\boldsymbol{x}_{ik}$
$\boldsymbol{b}_{ik}$	The hash code of region $\boldsymbol{x}_{ik}$
$\boldsymbol{w}_{ik}$	The weight of hash code $\boldsymbol{b}_{ik}$ when we obtain hash code $\boldsymbol{h}_i$
θ	Parameter set of the network
$\otimes$	element-wise multiplication

- How to get $\boldsymbol{h}_i$:
  - $f(\boldsymbol{x}_i, \theta) = \{\boldsymbol{h}_{ik}\}_{k=1}^{N^2} \xrightarrow{sign(\cdot)} \{\boldsymbol{b}_{ik}\}_{k=1}^{N^2}$
  - $\boldsymbol{h}_{ik} = \boldsymbol{w}_{ik} \otimes \boldsymbol{b}_{ik}$
  - $\boldsymbol{h}_i = \frac{1}{N^2}\sum_{i=1}^{N^2} \boldsymbol{h}_{ik} = \frac{1}{N^2}\sum_{i=1}^{N^2}(\boldsymbol{w}_{ik} \otimes \boldsymbol{b}_{ik})$

Out-of-sample Extension



$$b_q = sign(mean(f(x_q;\theta))) = sign(\frac{1}{N^2}\sum_{k=1}^{N^2} h_{qk})$$

Outline

- Hashing
- DMRH
- Experiments
- Summary

Experiment Setting

Dataset	Label	size	Training/Testing
CIFAR-10	Single(10)	60K	50K/10K
MS-COCO	Multiple(81)	122K	82K/40K
NUS-WIDE	Multiple(21)	195K	185K/10K

Method Parameters	Value	Method Parameters	Value
Implementation Framework	Pytorch	Optimizer	(SGD)Stochastic Gradient Descent
Initial Network Parameters	Pre-trained CNN-F on ImageNet	Initial Learning Rate	0.1
Batch Size	32	Hyper-Parameter η	0.02
Epoch	150		

Baseline

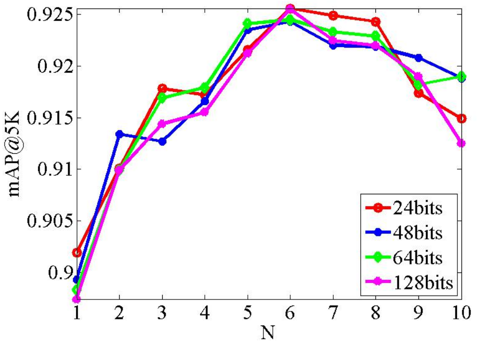
Baseline	Property	Baseline	Property
LSH(2004)	non-deep	DSH(2016)	deep
SH(2009)	non-deep	DPSH(2016)	deep
SKLSH(2009)	non-deep	DSDH(2017)	deep
PCAH(2010)	non-deep	DCH(2018)	deep
ITQ(2012)	non-deep	DDSH(2018)	deep
FSSH(2018)	non-deep	ADSH(2018)	deep

MAP Score

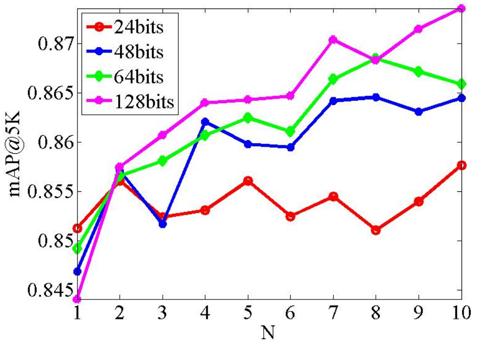
mAP@5K of DMRH and baselines on three dataset

Method	CIFAR-10				NUS-WIDE				MS-COCO			
	24 bits	48 bits	64 bits	128 bits	24 bits	48 bits	64 bits	128 bits	24 bits	48 bits	64 bits	128 bits
LSH	0.2722	0.3586	0.4490	0.4887	0.0654	0.1882	0.2993	0.3900	0.0868	0.1462	0.1774	0.3007
SH	0.2346	0.2959	0.3187	0.5168	0.1238	0.1729	0.2358	0.3448	0.0837	0.1048	0.1289	0.2373
SKLSH	0.2378	0.2983	0.3872	0.5517	0.0922	0.1387	0.2596	0.4354	0.0551	0.1369	0.1893	0.3966
PCAH	0.1430	0.1720	0.1863	0.2018	0.0924	0.0809	0.0890	0.1131	0.0662	0.0633	0.0702	0.0918
ITQ	0.3648	0.4245	0.4283	0.4502	0.3109	0.3884	0.4139	0.4571	0.2289	0.2862	0.3085	0.3515
FSSH	0.6853	0.7124	0.6919	0.7204	0.3959	0.3716	0.4462	0.5411	0.3105	0.3415	0.4063	0.4316
DSH	0.7864	0.7830	0.7834	0.7835	0.6598	0.6653	0.6587	0.6598	0.5135	0.5069	0.5147	0.5072
DPSH	0.8821	0.8853	0.8858	0.8876	0.8390	0.8429	0.8423	0.8468	0.6623	0.6871	0.6965	0.7073
DSDH	0.8985	0.9004	0.9002	0.8970	0.8225	0.8328	0.8347	0.8415	0.6988	0.7191	0.7220	0.7227
DCH	0.8753	0.8752	0.8749	0.8273	0.7552	0.7632	0.7647	0.7602	0.5858	0.5954	0.5948	0.5953
DDSH	0.8681	0.8875	0.8922	0.8995	0.7672	0.8171	0.8161	0.8008	0.5807	0.6004	0.6127	0.6292
ADSH	0.9043	0.9073	0.9073	0.9072	**0.8962**	**0.9030**	**0.9035**	**0.8927**	0.6605	0.6596	0.6648	0.6762
DMRH	**0.9256**	**0.9243**	**0.9245**	**0.9255**	0.8577	0.8645	0.8659	0.8736	**0.7510**	**0.7831**	**0.7890**	**0.8061**

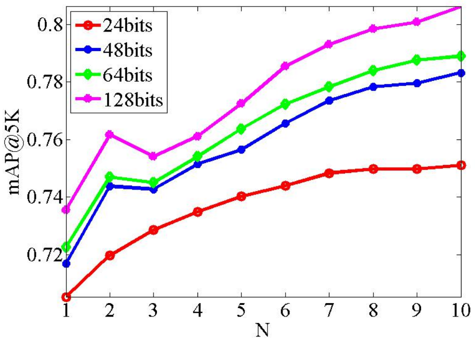
Performance with different N



(a) mAP@5K of different N on CIFAR-10

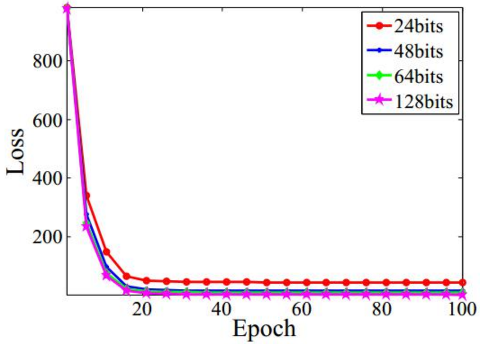


(b) mAP@5K of different N on NUS-WIDE

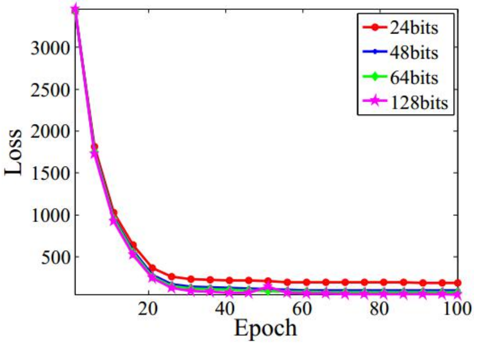


(c) mAP@5K of different N on MS-COCO

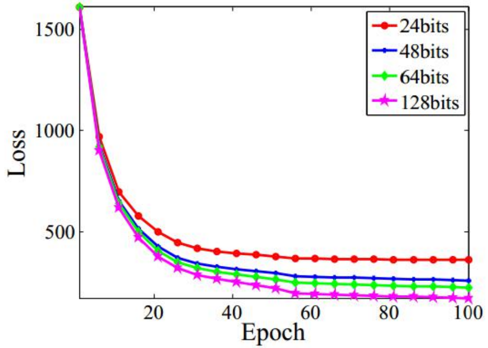
Convergence Analysis



(a) Convergence on CIFAR-10



(b) Convergence on NUS-WIDE



(c) Convergence on MS-COCO

Outline

- Hashing
- DMRH
- Experiments
- Summary

Summary

- We propose a deep supervised hashing method based on multiple regions of image. Different from the existing deep supervised hashing methods, the proposed DMRH can simply capture more semantic details from local regions by directly fusing them to hash codes. Experimental results show that DMRH could achieve state-of-the-art performance.

- However, the proposed method only uses local hash codes to obtain the global hash code. In the future work, we will consider utilizing the relationship among the local hash codes for better performance.

Thanks !

Questions ?