Discrete Semantic Matrix Factorization Hashing for Cross-Modal Retrieval

Jianyang Qin, Lunke Fei, Shaohua Teng, Wei Zhang, Genping Zhao, Haoliang Yuan

School of Computer Science and Technology, Guangdong University of Technology, Guangzhou, China

2021.1.13

Introduction

- **Cross-modal hashing** aims to map the heterogeneous data into a common binary code with semantic relations for cross similarity searching
- Two categories
  - ➢ **Unsupervised Hashing**, including Inter-Media Hashing (IMH), Collective Matrix Factorization Hashing (CMFH) and Latent Semantic Sparse Hashing (LSSH)
  - ➢ **Supervised hashing**, including Discrete Cross-Modal Hashing (DCH), Label Consistent Matrix Factorization Hashing (LCMFH) and Generalized Semantic Preserving Hashing (GSPH)
- One useful technique
  - ➢ **Collective Matrix Factorization**

Main Challenge

- *How to* **efficiently and effectively utilize the available label information** *for more accurate and reliable cross-modality retrieval needs to be further investigated.*

- *How to* **reduce the accumulated errors** *during hashing learning remains a central and challenging problem.*

The Proposed Method

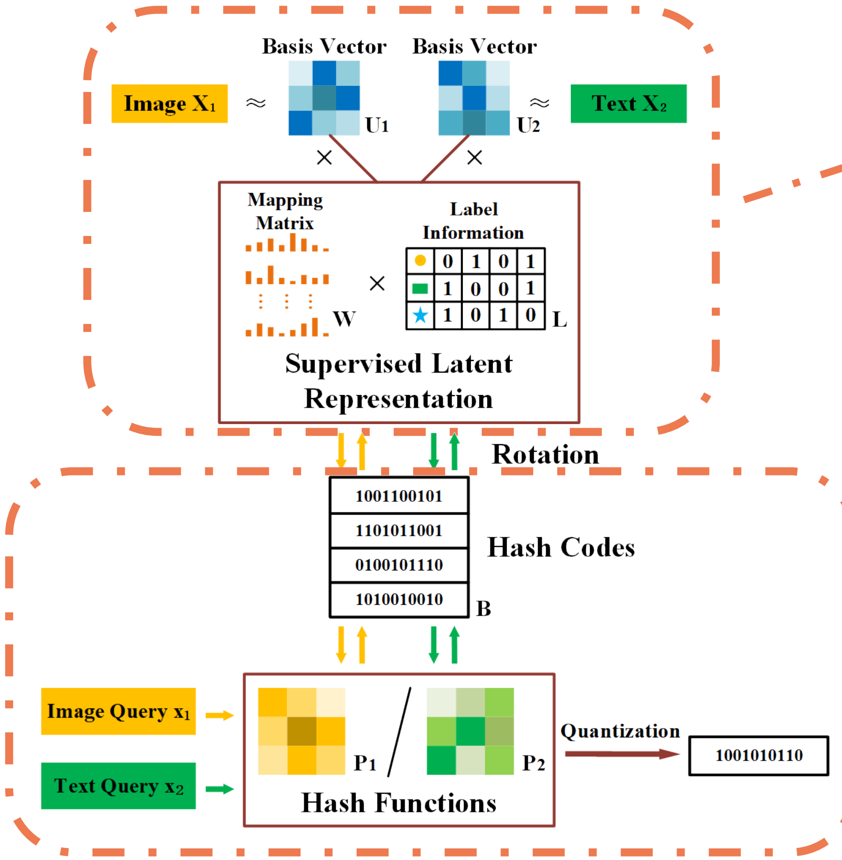

Fig. 1. The basic idea of the proposed Discrete Semantic Matrix Factorization Hashing.

First, we conduct the matrix factorization via directly utilizing the available label information to obtain a latent representation,

Then, we jointly learn the discriminative hash codes and corresponding hash functions by deriving the matrix factorization into a discrete optimization.

Finally, we employ an alternatively iterative procedure to efficiently optimize the matrix factorization and discrete learning.

Overall Objective Function

$$\underbrace{\sum_{t=1}^{M}\lambda_t\left\|X^{(t)}-U^{(t)}WL\right\|_F^2}_{\text{Learning Supervised Latent Representation}}+\underbrace{\left\|Q^TB-WL\right\|_F^2+\left\|Q\right\|_F^2}_{\text{Learning Hash Codes}}+\underbrace{\sum_{t=1}^{M}\left\|B-P^{(t)}X^{(t)}\right\|_F^2}_{\text{Learning Hash Functions}},\quad s.t.\ B\in\{-1,1\}^{r\times n}$$

By embedding the labels into the collective matrix factorization, we can learn a discriminative latent representation which preserve the semantic consistency between different modalities and corresponding label information

We directly learn the discrete hash codes via the quantization of rotation technique, and the binary codes can be optimized bit by bit by the Discrete Cyclic Coordinate Descent method.

Having obtaind different hash functions, the binary code of a new query data $x'^{(t)}$ can be easily encoded by $b'^{(t)} = sign(P^{(t)}x'^{(t)})$, which is used to retrieve similar data.

Experiments

We conduct the extensive experiments on three benchmark databases: Wiki, MIRFlickr, NUS-WIDE

- ***mAP evaluation***
- ***Precision-Recall curve evaluation***
- ***Visualized distribution***
- ***Computational complexity analysis***

Databases	Wiki	MIRFlickr25k	NUS-WIDE
Image Example			
Ground-Truth	sport	streat, clouds, sign, sky, tree	flowers, trees, grass, garden

Fig. 2. Some typical samples selected from the Wiki, MIRFlickr25k and NUS-WIDE databases.

mAP evaluation

TABLE I. The mAP results of seven methods with various code lengths on the Wiki, MIRFlickr and NUS-WIDE databases.

Task	Method	Wiki				Mirflick				NUS-WIDE			
		24bits	48bits	96bits	128bits	24bits	48bits	96bits	128bits	24bits	48bits	96bits	128bits
Image query Tag	CMFH	0.2093	0.2320	0.2357	0.2407	0.5580	0.5869	0.5848	0.5841	0.3909	0.3922	0.3931	0.3939
	SCM	0.1496	0.1511	0.1501	0.1509	0.5993	0.6019	0.6060	0.6073	0.4706	0.4732	0.4747	0.4713
	FSH	0.1539	0.1586	0.1636	01645	0.6114	0.6111	0.6093	0.6094	0.4803	0.4861	0.4874	0.4876
	SDMFH	0.3019	0.3307	0.3491	0.3495	0.6476	0.6628	0.6692	0.6687	0.5821	0.5812	0.6421	0.6242
	GSPH	0.2499	0.2809	0.2940	0.2823	0.6919	0.7022	0.7105	0.7118	0.5902	0.5967	0.5986	0.6067
	LCMFH	0.3512	0.3603	0.3752	0.3752	0.6681	0.6728	0.6776	0.6785	0.6225	0.6379	0.6395	0.6458
	DSMFH	**0.3553**	**0.3688**	**0.3921**	**0.3841**	**0.6923**	**0.7200**	**0.7261**	**0.7310**	**0.6236**	**0.6596**	**0.6695**	**0.6680**
Tag query Image	CMFH	0.4936	0.5246	0.5317	0.5416	0.5977	0.5959	0.5949	0.5928	0.4049	0.4109	0.4123	0.4169
	SCM	0.5354	0.5443	0.5423	0.5427	0.6775	0.6904	0.7084	0.7142	0.6126	0.6334	0.6495	0.6390
	FSH	0.4869	0.5129	0.5245	0.5256	0.6479	0.6491	0.6478	0.6472	0.5546	0.5676	0.5611	0.5667
	SDMFH	0.6605	0.7061	0.7115	0.7075	0.7448	0.7744	0.7765	0.8020	0.7100	0.7676	0.7665	0.7824
	GSPH	0.6041	0.6027	0.6300	0.6331	0.7459	0.7563	0.7695	0.7705	0.6827	0.6911	0.6898	0.6962
	LCMFH	0.7073	0.7369	0.7400	0.7315	0.7301	0.7468	0.7557	0.7615	0.7279	0.7393	0.7437	0.7470
	DSMFH	**0.7257**	**0.7526**	**0.7554**	**0.7564**	**0.7544**	**0.7919**	**0.8029**	**0.8100**	**0.7576**	**0.7871**	**0.7994**	**0.7956**

Precision-Recall curve evaluation

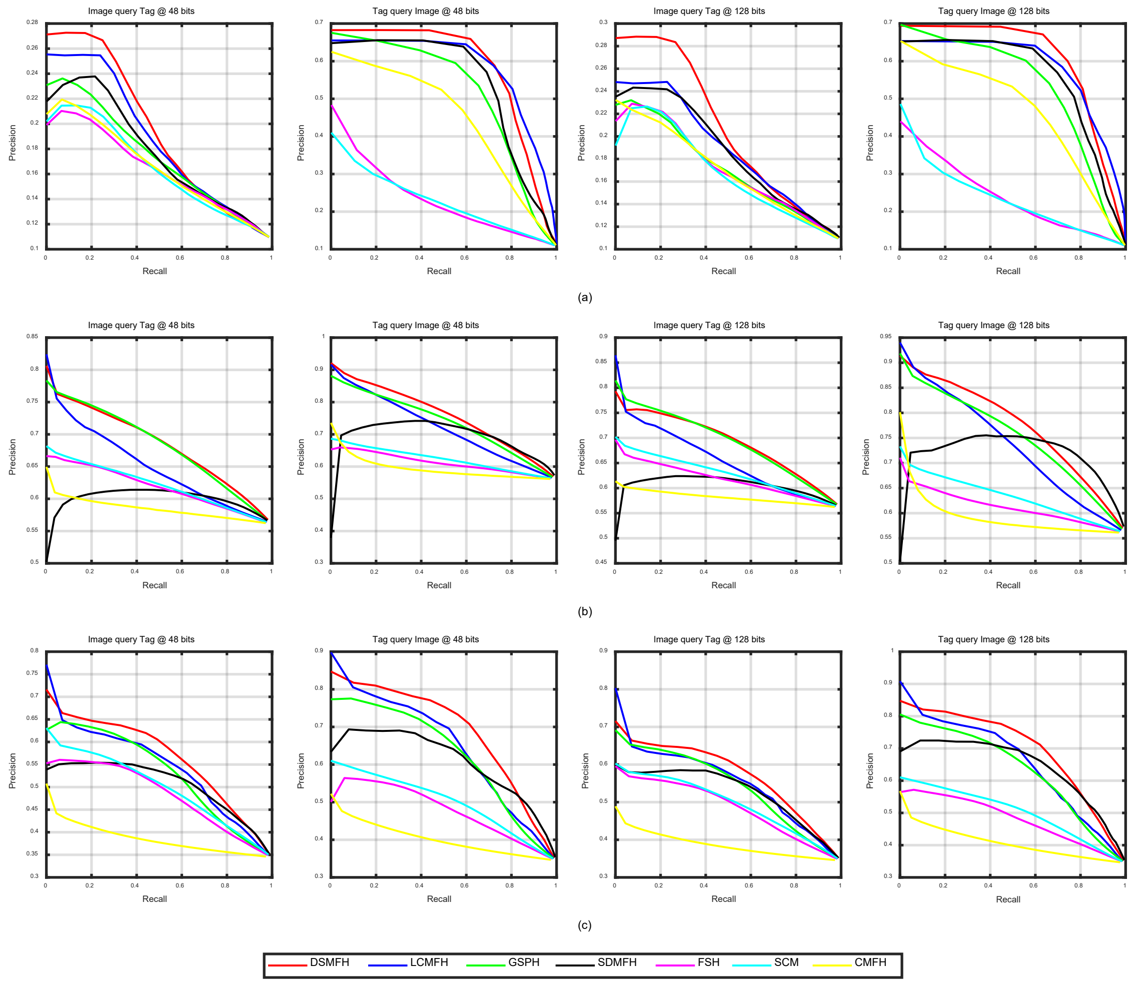

Fig. 3. The PR curves of different methods on the (a) Wiki, (b) MIRFlickr25k, (c) NUS-WIDE databases, respectively

Visualized distribution

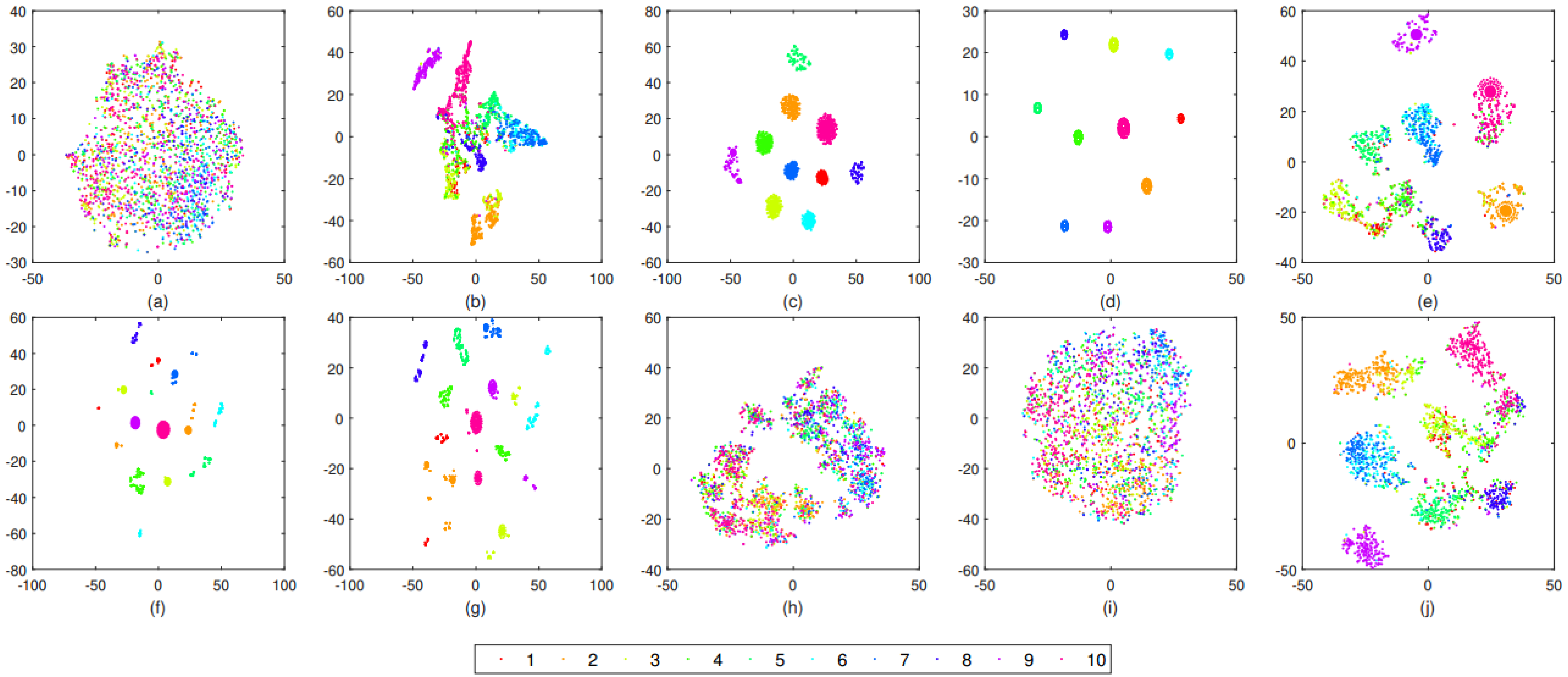

Fig. 4. The visualized distributions of (a) Image, (b) Textual Tag, (c) DSMFH, (d) LCMFH, (e) GSPH, (f) SDMFH-Image (g) SDMFH-Textual-Tag (h) FSH, (i) SCM and (j) CMFH features, respectively, on the Wiki database in the 2D space via t-SNE.

Computational complexity analysis

TABLE II. Training time (in second) of six methods on the NUS-WIDE database

Methods\Size of training set	1500	3000	5000	7000	10000
CMFH	7.0	11.6	17.4	23.4	31.9
SCM	17.7	17.8	16.9	18.1	18.5
FSH	10.1	17.9	26.5	35.9	47.9
SDMFH	0.6	3.3	6.8	12.2	18.6
GSPH	89.4	140.3	218.5	306.6	453.6
LCMFH	1.5	2.3	3.4	4.3	5.9
DSMFH	0.8	2.8	3.9	6.9	10.6

Thank You!