Deep Hashing with Hash Center Update for Efficient Image Retrieval

Abin Jose, Daniel Filbert, Christian Rohlfing, Jens-Rainer Ohm

Institut für Nachrichtentechnik, RWTH Aachen University, Paper number : 4514

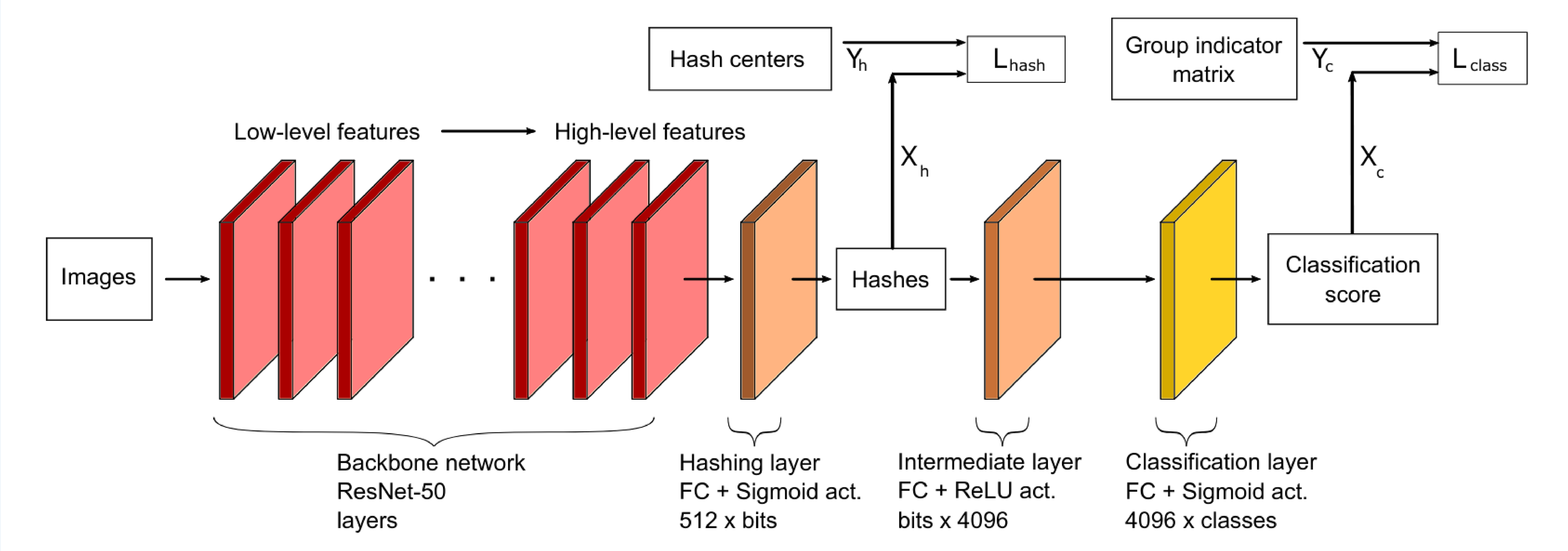
Abstract

- An approach for learning binary hash codes for image retrieval.
- Canonical Correlation Analysis (CCA) is used to design two loss functions for training the neural network.
- The correlation between the two views to CCA is maximized by the network.
- The main motivation of using CCA - Dimensionality reduction.
- First loss maximizes the correlation between the hash centers and hash codes.
- Second loss maximizes the correlation between class labels and classification scores.
- A novel weighted mean and thresholding-based hash center update scheme for adapting hash centers is proposed.
- Training loss reaches the theoretcial lower bound of the proposed loss function.
- The measured mean average precision outperforms other state-of-the-art methods.

Experimental Setup

- Two multi-labeled datasets, were used for evaluation of the performance - NUS-WIDE and MS-COCO.
- Training and test curves were plotted for each dataset.
- Retrieval performance was measured by calculating the mean average precision (MAP).
- The neural network architecture used is ResNet-50.
- Batch size used was 200, learning rate 0.0008 and learning rate decay 0.1 every 10th epoch.
- Optimizer used was Stochastic Gradient Descent (SGD).

Network Architecture



Proposed Deep Central Similarity Hashing network architecture. ResNet layers are used as the backbone network. Hashing and classification layer consist of a fully connected layer with subsequent sigmoid activation. Intermediate layer consists of fully connected layer with ReLU activation.

Algorithm

Algorithm DCSH hash center update

Require: Hash centers $\mathcal{H}^{(i)} = \{\vec{h}_c^{(i)}\}, c = 1, ..., C$ with C classes, and output hashes $\{\vec{x_h}^{(i)}\} \in [0;1]^n$ with $n = 1, ..., N$ from N training images at epoch i.

1: For each class c, group all hash outputs associated with c in their label l_n:
$\mathcal{G}_c^{(i)} = \{\vec{x_h}^{(i)} : c \text{ in } l_\mathrm{n}\}$, for $c = 1, ..., C$.

2: Calculate weights for each output hash based on number of classes in its label $|l_\mathrm{n}|$:
$w_\mathrm{n} = \frac{1}{|l_\mathrm{n}|}$.

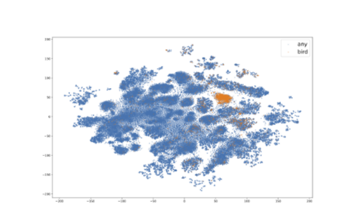
3: Calculate weighted mean of grouped hashing values:
$\vec{\tilde{h}}_c^{(i+1)} = \frac{1}{|\mathcal{G}_c^{(i)}|} \sum_{\vec{x_h}^{(i)} \in \mathcal{G}_c^{(i)}} w_\mathrm{n} \, \vec{x_h}^{(i)}$, for $c = 1, ..., C$.

4: Create updated binary hash centers by thresholding:
$\vec{h}_c^{(i+1)} = \mathrm{sign}(\vec{\tilde{h}}_c^{(i+1)})$.

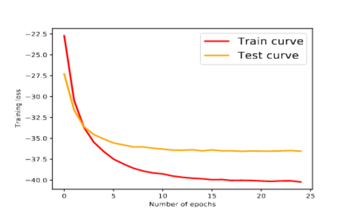
5: **return** Updated hash centers $\mathcal{H}^{(i+1)}$ for epoch $i + 1$.

Proposed DCSH hash center update algorithm.

Results - MSCOCO

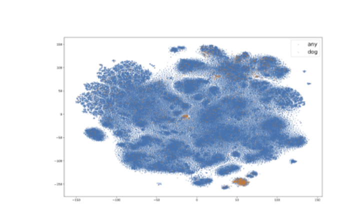


(a) MS-COCO bird class in orange

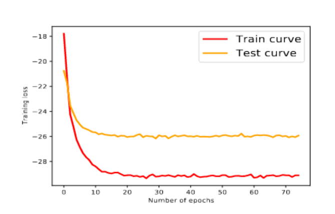


(c) MS-COCO loss curve

Results - NUSWIDE



(b) NUS-WIDE dog class in orange



(d) NUS-WIDE loss curve

MAP values

- MAP value is highest for our approach.
- As the bit length increases MAP values increase as well.

MS-COCO					NUS-WIDE				
Method	16 bits	32 bits	48 bits	64 bits	Method	12 bits	24 bits	32 bits	48 bits
DCSH (ours)	**0.805**	**0.847**	**0.859**	**0.861**	**DCSH (Ours)**	**0.823**	**0.833**	**0.841**	**0.857**
CSQ [21]	0.796	0.838	-	**0.861**	DPSH [15]	0.794	0.822	0.838	0.851
DCCH [19]	0.659	0.729	0.731	0.739	DCCH [19]	0.782	0.814	0.825	0.834
HashNet [3]	0.687	0.718	0.730	0.736	CSQ [21]	-	-	0.825	-
DHN [28]	0.677	0.701	0.695	0.694	DSDH [26]	0.776	0.808	0.820	0.829
DNNH [29]	0.593	0.603	0.604	0.610	DDSH [17]	0.791	0.815	0.821	0.827
CNNH [30]	0.564	0.574	0.571	0.567	DTSH [27]	0.773	0.808	0.812	0.814

Conclusions

- An efficient approach for learning hash codes for image retrieval is proposed.
- The correlation between hash codes and hash centers is maximized.
- Experiments substantiate formation of an optimized feature space with minimum intra-class scatter and maximum inter-class scatter.
- This is infact possible due to the equivalence of CCA with Linear Discriminant Analysis (LDA).
- As a future work more effective representations of individual classes could be explored.

jose@ient.rwth-aachen.de

www.ient.rwth-aachen.de

Institut für Nachrichtentechnik, Melatener Str. 23, 52074 Aachen

ICASSP 2022