

# Hashing by Deep Learning

Wei Liu

IBM T. J. Watson Research Center  
Yorktown Heights, New York 10598, USA

weiliu@us.ibm.com

During the past decade (since around 2006), *Deep Learning* [7], also known as *Deep Neural Networks*, has drawn increasing attention and research efforts in a variety of artificial intelligence areas including speech recognition, computer vision, machine learning, text mining, *etc.* Since one main purpose of deep learning is to learn robust and powerful feature representations for complex data, it is very natural to leverage deep learning for exploring compact hash codes which can be regarded as binary representations of data. In this section, we briefly introduce several recently proposed hashing methods that employ deep learning. Table 1 compares eight hashing methods in terms of four key characteristics that can be used to differentiate these methods.

The earliest work in deep learning based hashing may be *Semantic Hashing* [12]. This method builds a deep generative model to discover hidden binary units (*i.e.*, latent topic features) which can model input text data (*i.e.*, word-count vectors). Such a deep model is made as a stack of *Restricted Boltzmann Machines* (RBMs) [4]. After learning a multi-layer RBM through pre-training and fine-tuning on a collection of documents, the hash code of any document is acquired by simply thresholding the output of the deepest layer. Such hash codes provided by the deep RBM were shown to preserve semantically similar relationships among input documents into the code space, in which each hash code (or hash key) is used as a memory address to locate corresponding documents. In this way, semantically similar documents are mapped to adjacent memory addresses, thereby enabling efficient search via hash table lookup. To enhance the performance of deep RBMs, a supervised version was proposed in [13], which borrows the idea of nonlinear Neighbourhood Component Analysis (NCA) embedding [11]. The supervised information stems from given neighbor/nonneighbor relationships between training examples. Then, the objective function of NCA is optimized on the top of a deep RBM, making the deep RBM yield discriminative hash codes. Note that supervised deep RBMs can be applied to broad data domains other than text data. In [13], supervised deep RBMs using a Gaussian distribution to model visible units in the first layer were successfully applied to handle massive image data.

A recent work named *Sparse Similarity-Preserving Hashing* [10] tried to address the low recall issue pertaining to relatively long hash codes, which affects most of previous hashing techniques. The idea is enforcing sparsity into the hash codes to be learned from training examples with pairwise supervised information, that is, similar and dissimilar pairs of examples (also known as side information in the machine learning literature). The relaxed hash functions, actually nonlinear embedding functions, are learned by training a *Tailored Feed-Forward Neural Network*. Within this architecture, two ISTA-type networks [3] that share the same set of parameters and conduct fast approximations of sparse coding are coupled in the training phase. Since each output of the neural network is continuous albeit sparse, a hyperbolic tangent function is applied to the output followed by a thresholding operation, leading to the final binary hash code. In [10], an extension to hashing multimodal data, *e.g.*, web images with textual tags, was also presented.

Another work named *Deep Hashing* [8] developed a deep neural network to learn a multiple hierarchical nonlinear transformation which maps original images to compact binary hash codes and hence supports large-scale image retrieval with the learned binary image representation. The deep hashing model is established under three constraints which are imposed on the top layer of the deep neural network: 1) the reconstruction error between an original real-valued image feature vector and the resulting binary code is minimized, 2) each bit of binary codes has a balance, and 3) all bits are independent of each other. Similar constraints have been adopted in prior unsupervised hashing or binary coding approaches such as Iterative Quantization (ITQ) [2]. A supervised version called *Supervised Deep Hashing*<sup>1</sup> was also presented in [8], where a discriminative term incorporating pairwise supervised information is added to the objective function of the deep hashing model. The authors of [8] showed the superiority of the supervised deep hashing model over its unsupervised counterpart. Both of them

---

<sup>1</sup>It is essentially semi-supervised as abundant unlabeled examples are used for training the deep neural network.

produce hash codes through thresholding the outputs of the top layer in the neural network, where all activation functions are hyperbolic tangent functions.

It is worthwhile to point out that the above methods, including Sparse Similarity-Preserving Hashing, Deep Hashing, and Supervised Deep Hashing, did not include a pre-training stage during the training of the used deep neural networks. The absence of pre-training may make the generated hash codes less effective. Specifically, the Sparse Similarity-Preserving Hashing method is found to be inferior to the state-of-the-art supervised hashing method Kernel-Based Supervised Hashing (KSH) [9] in image search accuracy on some datasets [10]; the Deep Hashing method and its supervised version are slightly better than ITQ and its supervised version CCA+ITQ [2] in image search accuracy, respectively [8]. Note that KSH, ITQ and CCA+ITQ exploit only shallow learning frameworks.

Almost all existing hashing techniques including the aforementioned ones relying on deep neural networks take a vector of hand-crafted visual features extracted from an image as input. Therefore, the quality of produced hash codes heavily depends on the quality of hand-crafted features. To remove this barrier, a recent method called *Convolutional Neural Network Hashing* [14] was developed to integrate image feature learning and hash value learning into a joint learning model. Given pairwise supervised information, this model consists of a stage of learning approximate hash codes and a stage of training a deep *Convolutional Neural Network* (CNN) [5] that outputs continuous hash values. Such hash values can be generated by activation functions like sigmoid, hyperbolic tangent or softmax, and then quantized into binary hash codes through appropriate thresholding. Thanks to the power of CNNs, the joint model is capable of simultaneously learning image features and hash values, directly working on raw image pixels. The deployed CNN is composed of three convolution-pooling layers that involve rectified linear activation, max pooling, and local contrast normalization, a standard fully-connected layer, and an output layer with softmax activation functions.

Also based on CNNs, a latest method called as *Deep Semantic Ranking Hashing* [15] was presented to learn hash values such that multilevel semantic similarities among multi-labeled images are preserved. Like the Convolutional Neural Network Hashing method, this method takes image pixels as input and trains a deep CNN, by which image feature representations and hash values are jointly learned. The deployed CNN consists of five convolution-pooling layers, two fully-connected layers, and one hash layer (*i.e.*, output layer). The key hash layer is connected to both fully-connected layers and in the function expression  $\mathbf{h} = 2\sigma(\mathbf{w}^\top [f_1(I); f_2(I)]) - 1$ , in which  $I$  represents an input image,  $\mathbf{h}$  represents the vector of hash values for image  $I$ ,  $f_1(I)$  and  $f_2(I)$  respectively denote the feature representations from the outputs of the first and second fully-connected layers,  $\mathbf{w}$  is the weight vector, and  $\sigma(\cdot)$  is the logistic function. The Deep Semantic Ranking Hashing method leverages listwise supervised information to train the CNN, which stems from a collection of image triplets that encode the multilevel similarities, *i.e.*, the first image in each triplet is more similar to the second one than the third one. The hash code of image  $I$  is finally obtained by thresholding the output  $\mathbf{h}$  of the hash layer at zero.

The above Convolutional Neural Network Hashing method [14] requires separately learning approximate hash codes to guide the subsequent learning of image representation and finer hash values. A latest method called *Deep Neural Network Hashing* [6] goes beyond, in which the image representation and hash values are learned in one stage so that representation learning and hash learning are tightly coupled to benefit each other. Similar to the Deep Semantic Ranking Hashing method [15], the Deep Neural Network Hashing method incorporates listwise supervised information to train a deep CNN, giving rise to a currently deepest architecture for supervised hashing. The pipeline of the deep hashing architecture includes three building blocks: 1) a triplet of images which are fed to the CNN, and upon which a triplet ranking loss is designed to characterize the listwise supervised information; 2) a shared sub-network with a stack of eight convolution layers to generate the intermediate image features; 3) a divide-and-encode module to divide the intermediate image features into multiple channels, each of which is encoded into a single hash bit. Within the divide-and-encode module, there are one fully-connected layer and one hash layer. The former uses sigmoid activation, while the latter uses a piecewise thresholding scheme to output nearly discrete hash values. Eventually, the hash code of any image is yielded by thresholding the output of the hash layer at 0.5. In [6], the Deep Neural Network Hashing method was shown to surpass the Convolutional Neural Network Hashing method as well as several shallow learning based supervised hashing methods in terms of image search accuracy.

Last, a few observations about the introduced deep learning based hashing methods are worth mentioning:

1. The majority of those methods did not report the time of hash code generation. In real-world search scenarios, the speed for generating hashes should be substantially fast. There might be concern about the hashing speed of those deep neural network driven approaches, especially the approaches involving image feature learning, which may take much longer time to hash an image compared to shallow learning driven approaches like ITQ and KSH.
2. Instead of employing deep neural networks to seek hash codes, another interesting problem is to design a proper hashing technique to accelerate deep neural network training or save memory space. The latest work [1] presented a hashing

**Table 1:** The characteristics of eight recently proposed deep learning based hashing methods.

Deep learning based hashing methods	Data domain	Unsupervised or supervised?	Learning features?	Hierarchy of deep neural networks
Semantic Hashing [12]	text	unsupervised	no	4
Restricted Boltzmann Machine [13]	text and image	supervised	no	4 and 5
Tailored Feed-Forward Neural Network [10]	text and image	supervised	no	6
Deep Hashing [8]	image	unsupervised	no	3
Supervised Deep Hashing [8]	image	supervised	no	3
Convolutional Neural Network Hashing [14]	image	supervised	yes	5
Deep Semantic Ranking Hashing [15]	image	supervised	yes	8
Deep Neural Network Hashing [6]	image	supervised	yes	10

trick named *HashedNets*, which shrinks the storage costs of neural networks significantly while mostly preserving the generalization performance in image classification.

## References

- [1] W. Chen, J. T. Wilson, S. Tyree, K. Q. Weinberger, and Y. Chen. Compressing neural networks with the hashing trick. In *Proc. International Conference on Machine Learning (ICML)*, pages 2285–2294, 2015. 2
- [2] Y. Gong, S. Lazebnik, A. Gordo, and F. Perronnin. Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12):2916–2929, 2013. 1, 2
- [3] K. Gregor and Y. LeCun. Learning fast approximations of sparse coding. In *Proc. International Conference on Machine Learning (ICML)*, pages 399–406, 2010. 1
- [4] G. Hinton and R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006. 1
- [5] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In *Proc. Advances in Neural Information Processing Systems (NIPS)*, volume 25, pages 1106–1114, 2012. 2
- [6] H. Lai, Y. Pan, Y. Liu, and S. Yan. Simultaneous feature learning and hash coding with deep neural networks. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2, 3
- [7] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521:436–444, 2015. 1
- [8] V. E. Liong, J. Lu, G. Wang, P. Moulin, and J. Zhou. Deep hashing for compact binary codes learning. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 1, 2, 3
- [9] W. Liu, J. Wang, R. Ji, Y.-G. Jiang, and S.-F. Chang. Supervised hashing with kernels. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2074–2081, 2012. 2
- [10] J. Masci, A. Bronstein, M. Bronstein, P. Sprechmann, and G. Sapiro. Sparse similarity-preserving hashing. In *Proc. International Conference on Learning Representations (ICLR)*, 2014. 1, 2, 3
- [11] R. Salakhutdinov and G. Hinton. Learning a nonlinear embedding by preserving class neighbourhood structure. In *Proc. International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 412–419, 2007. 1
- [12] R. Salakhutdinov and G. Hinton. Semantic hashing. *International Journal of Approximate Reasoning*, 50(7):969–978, 2009. 1, 3
- [13] A. Torralba, R. Fergus, and Y. Weiss. Small codes and large image databases for recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008. 1, 3
- [14] R. Xia, Y. Pan, H. Lai, C. Liu, and S. Yan. Supervised hashing for image retrieval via image representation learning. In *Proc. AAAI Conference on Artificial Intelligence (AAAI)*, pages 2156–2162, 2014. 2, 3
- [15] F. Zhao, Y. Huang, L. Wang, and T. Tan. Deep semantic ranking based hashing for multi-label image retrieval. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2, 3