Deep learning to ternary hash codes by continuation

Mingrui Chen,¹ Weiyu Li,² and Weizhi Lu¹,✉

¹School of Control Science and Engineering, Shandong University, Jinan, P. R. China

²Zhongtai Securities Institute for Financial Studies, Shandong University, Jinan, P. R. China

✉Email: wzlu@sdu.edu.cn

Recently, it has been observed that $\{0, \pm 1\}$ -ternary codes, which are simply generated from deep features by hard thresholding, tend to outperform $\{-1, 1\}$ -binary codes in image retrieval. To obtain better ternary codes, the authors for the first time propose to jointly learn the features with the codes by appending a smoothed function to the networks. During training, the function could evolve into a non-smoothed ternary function by a continuation method, and then generate ternary codes. The method circumvents the difficulty of directly training discrete functions and reduces the quantization errors of ternary codes. Experiments show that the proposed joint learning indeed could produce better ternary codes. For the first time, the authors propose to generate ternary hash codes by jointly learning the codes with deep features via a continuation method. Experiments show that the proposed method outperforms existing methods.

Introduction: Existing hashing methods mainly exploit binary codes for image and text retrieval [1]. The codes are generated by binarizing the features learned by data-independent or data-dependent methods. Among the methods, the data-driven deep learning method tends to perform best [2–9], thanks to its powerful capability in generating discriminative features. Despite excellent performance, the binary codes generated by deep features are not perfect, tending to suffer from performance bottlenecks and even declines [10, 11], with the increasing of code dimension. The reason is as follows. With the increasing of feature dimension, deep features incline to become sparse, and then involve a large number of small/ambiguous elements close to zero [12, 13]. These elements would cause large quantization errors and worse code features, as being quantized to bipolar values $+1$ or -1 . To address the issue, it is natural to introduce a third state ‘zero’ to specially denote the ambiguous elements, thus yielding $\{0, \pm 1\}$ -ternary codes. Such kind of codes have recently proved to achieve higher retrieval accuracy than binary codes [13, 14].

Up to date, to the best of our knowledge, only two methods [13, 14] have been proposed to generate ternary codes for image retrieval. However, the two methods are suboptimal because they have deep features learned and ternarized in two separated steps. Specifically, in [14] the features are generated with AlexNet [15], and then ternarized by two thresholds selected empirically. To obtain better thresholds, a searching algorithm is proposed in [13], based on the principle of maximizing the expectation of pairwise ternary hamming distances between similar samples and decreasing the distances between dissimilar samples. Obviously, the performance is still limited by the quality of the features previously learned alone.

To overcome the limitation of independent learning, we propose to jointly learn the features with the codes. An intuitive idea is to append a ternary function to the feature extraction network and then take them as a whole to train. For this framework, the main challenge comes from the optimization of the ternary function, which has zero gradients and makes back-propagation infeasible. To avoid directly training the discrete function, inspired by [16], we propose a smoothed function to gradually evolve into the targeted ternary function in the iterative network training. This continuation method is known for being able to optimize discrete functions with guaranteed convergence [17]. Experiments show that the proposed joint learning method indeed outperforms existing independent learning methods.

Method: As illustrated in Figure 1, the joint learning forms a network pipeline consisting of four parts: (1) a convolutional neural network

Fig. 1 The proposed network architecture for jointly learning the features and ternary codes by a continuation method

(CNN) for learning deep features, (2) a fully connected hash layer for transforming the features into d dimensions, (3) a smoothed ternary function for converting each element of d -dimensional features to be a value close to 1, -1 or 0, and (4) a loss function. To obtain generic results, as in [13, 14], we will exploit the known AlexNet [15] for feature learning, and the cross-entropy loss for network training, although other more selective networks and losses may lead to better codes. The smoothed ternary function is proposed as follows:

$$f(x) = \tanh((x/\alpha)^k), \quad (1)$$

where α is a positive constant and k is required to be an odd number greater than one, namely $k = 3, 5, 7$, and so on. Note that the odd k renders $f(x)$ take both positive and negative values, while the even k can only lead to non-negative values. With the odd k tending to infinity, as shown in Figure 1, $f(x)$ will quickly converge to a ternary function

$$g(x) = \begin{cases} 1, & x \geq \alpha \\ -1, & x \leq -\alpha \\ 0, & \text{others,} \end{cases} \quad (2)$$

where the threshold parameter α is identical to the scale parameter α provided in (1). In our experiments, we will set $\alpha = 0.5$ in terms of the fact that the CNN feature elements x usually have values normalized to the range $[-1, +1]$. Considering the equivalence between (1) and (2) in the limit of k , we propose to gradually increase the value of k during training, such that the smoothed function (1) finally approaches the targeted ternary function (2). In the testing phase, we need to substitute (1) with (2) to generate ternary codes, and further, as in [13], the codes will be converted to binary bits via the mapping $\{-1, 0, 1\} \rightarrow \{01, 00, 10\}$, in order to conduct image retrieval with hamming distances on binary machines.

Experiments: We aim to prove that the proposed deep ternary codes (DTC) outperform existing other two kinds of ternary codes termed deep polarized networks (DPN) [14] and ternary hashing (TH) [13], by conducting image retrieval on three typical databases CIFAR10 [18], NUS-WIDE [19], and ImageNet100 [20]. Note that DPN and TH have reported better performance than binary codes, and so the latter are not compared in our experiments. Following the settings in DPN and TH, the databases are processed as follows: (1) CIFAR10 consists of 60,000 coloured images in 10 classes. We randomly select 1000 images (100 images per class) as a query set, and the remaining 59,000 images are taken as a retrieval set. From the retrieval set, 5000 images (500 images per class) are randomly selected for network training. (2) NUS-WIDE comprises 269,648 multi-labelled images in 81 classes. As usual, we select the most frequent 21 classes for experiments. Among them, we randomly select 100 images per class for query and the rest for retrieval. From the retrieval set, 500 images per class are randomly sampled for network training. (3) ImageNet contains 1000 categories of images, including over 1.2M images in the training set and 50k images in the validation set. ImageNet100 collects 100 categories from ImageNet, with the entire validation set for query and the entire training set for retrieval. From the retrieval set, we randomly select 100 images per category for network training.

For generality, as stated before, we adopt the AlexNet equipped with cross-entropy loss for feature learning. The network is trained with

Table 1. Hashing accuracies in terms of mean average precision (mAP) for three different methods on three typical datasets. The best results are highlighted in bold

CIFAR10@all					
Method	16dim	24dim	32dim	64dim	128dim
DTC	0.830	0.831	0.840	0.842	0.843
DPN [14]	0.825	-	0.838	0.830	0.829
TH [13]	0.820	0.823	0.836	0.836	0.825
NUS-WIDE@5K					
Method	16dim	24dim	32dim	64dim	128dim
DTC	0.850	0.855	0.862	0.865	0.864
DPN [14]	0.847	-	0.859	0.863	0.862
TH [13]	0.830	0.841	0.843	0.846	0.840
ImageNet100@1K					
Method	16dim	24dim	32dim	64dim	128dim
DTC	0.685	0.720	0.748	0.763	0.765
DPN [14]	0.684	-	0.740	0.756	0.756
TH [13]	0.668	0.710	0.732	0.754	0.762

stochastic gradient decent with 0.9 momentum. The learning rate is initialized as 10^{-3} , with cosine learning rate decay. We set the batch size as 64 and the weight decay parameter as 10^{-4} . The parameter k for $f(x)$ is gradually increased from 3 to 11 in a total of 150 epoches, with a stride of 2 every 30 epoches. The retrieval accuracy is evaluated with the mean average precision (mAP). As in DPN [14] and TH [13], the mAP is calculated with all retrieval images as returned images for CIFAR-10, with top 5000 returned images for NUS-WIDE, and with top 1000 returned images for ImageNet100. For fair comparisons, we directly compare with the best results reported by DPN and TH in their original literature [13, 14].

The retrieval performance is compared in Table 1. It is seen that our ternary codes (termed DTC) achieve consistent performance gains over other two kinds of ternary codes, on three databases with varying code dimensions. The gains range from 0.1% to 1.4%. It is noteworthy that the results for TH [13] are mostly obtained with a hinge-like loss, which tends to generate more discriminative features than the cross-entropy loss used in our method. This means that our performance gap with TH could be further enlarged, if using the same loss. Overall, compared to existing independent learning methods [13, 14], the proposed joint learning method indeed could generate better ternary codes, thanks to its advantage in reducing quantization errors.

Conclusion: We have proposed a continuation method to jointly learn the features and ternary codes in an end-to-end manner. The core of the method is to introduce a smoothed function to gradually approach a targeted ternary function during network training, and this avoids the difficulty of directly optimizing discrete ternary functions. As expected, the proposed joint learning method generates better ternary codes than existing independent learning methods. To obtain generic results, we simply tested the method on the AlexNet trained with cross-entropy loss. It is believed that better ternary codes should be obtained, if adopting more selective networks and losses.

Acknowledgements: This work has been supported by the National Natural Science Foundation of China (grant numbers 61801264 and 61991412) and the Fundamental Research Funds of Shandong University (grant number 2019HW018).

Data availability statement: The databases that support the findings of this study are openly available by their references.

© 2021 The Authors. *Electronics Letters* published by John Wiley & Sons Ltd on behalf of The Institution of Engineering and Technology

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

Received: 7 July 2021 Accepted: 5 September 2021

doi: 10.1049/ell2.12317

References

- Wang, J., Zhang, T., Sebe, N., Shen, H.T., et al.: A survey on learning to hash. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **40**(4), 769–790 (2017)
- Xia, R., Pan, Y., Lai, H., Liu, C. & Yan, S.: Supervised hashing for image retrieval via image representation learning. In Twenty-Eighth AAAI Conference on Artificial Intelligence, Québec City, Québec, Canada (2014)
- Lin, K., Yang, H.-F., Hsiao, J.-H. & Chen, C.-S.: Deep learning of binary hash codes for fast image retrieval. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 27–35, Boston, MA, USA (2015)
- Zhu, H., Long, M., Wang, J. & Cao, Y.: Deep hashing network for efficient similarity retrieval. In Proceedings of the AAAI Conference on Artificial Intelligence, Phoenix, Arizona USA (2016)
- Liu, H., Wang, R., Shan, S. & Chen, X.: Deep supervised hashing for fast image retrieval. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2064–2072, Las Vegas, Nevada, USA (2016)
- Bai, C., Huang, L., Pan, X., Zheng, J., Chen, S.: Optimization of deep convolutional neural network for large scale image retrieval. *Neurocomputing* **303**, 60–67 (2018)
- Jin, L., Li, Z., Tang, J.: Deep semantic multimodal hashing network for scalable image-text and video-text retrievals. *IEEE Transactions on Neural Networks and Learning Systems* (2020)
- Li, Z., Tang, J., Zhang, L., Yang, J.: Weakly-supervised semantic guided hashing for social image retrieval. *International Journal of Computer Vision*, **128**, 2265–2278 (2020)
- Bai, C., Zeng, C., Ma, Q., Zhang, J. & Chen, S.: Deep adversarial discrete hashing for cross-modal retrieval. In Proceedings of the 2020 International Conference on Multimedia Retrieval, pp. 525–531, Dublin, Ireland (2020)
- Lai, H., Pan, Y., Liu, Y. & Yan, S.: Simultaneous feature learning and hash coding with deep neural networks. 2015 IEEE Conference on Computer Vision and Pattern Recognition, pp. 3270–3278, Boston, MA, USA (2015)
- Cao, Y., Liu, B., Long, M. & Wang, J.: Hashgan: Deep learning to hash with pair conditional wasserstein gan. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1287–1296, Salt Lake City, Utah, USA (2018)
- Kong, W. & Li, W.-J.: Double-bit quantization for hashing. In Proceedings of the AAAI Conference on Artificial Intelligence (2012)
- Liu, C., Fan, L., Ng, K.W., Jin, Y., Ju, C., Zhang, T., Chan, C.S. & Yang, Q.: Ternary hashing. arXiv preprint arXiv:2103.09173 (2021)
- Fan, L., Ng, K., Ju, C., Zhang, T. & Chan, C.S.: Deep polarized network for supervised learning of accurate binary hashing codes. In Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, Yokohama, Japan (2020)
- Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In Advances in Neural Information Processing Systems (2012)
- Cao, Z., Long, M., Wang, J. & Yu, P.S.: Hashnet: Deep learning to hash by continuation. In Proceedings of the IEEE International Conference on Computer Vision, pp. 5608–5617, Santorini, Fira Greece (2017)
- Allgower, E.L., Georg, K.: *Numerical Continuation Methods: An Introduction*. vol. 13. Springer Science & Business Media (2012)
- Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images. Master's thesis, Department of Computer Science, University of Toronto (2009)
- Chua, T.-S., Tang, J., Hong, R., Li, H., Luo, Z. & Zheng, Y.: NUS-WIDE: A real-world web image database from National University of Singapore. In Proceedings of the ACM International Conference on Image and Video Retrieval, pp. 1–9, Santorini, Fira Greece (2009)
- Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K. & Fei-Fei, L.: ImageNet: A large-scale hierarchical image database. In IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255, Miami, FL, USA (2009)