Efficient Hashing Methods for Finding Objects in Large Volumes of Data

Lubov Ivanova, Sergei Ivanov

Abstract—**This article analyzes machine learning methods for efficient access and rapid retrieval of objects such as images, videos, and documents in large data sets. Various hashing methods are considered: deep lifelong cross-modal hashing, learned locality-sensitive hashing, graph-collaborated auto-encoder hashing for multi-view binary clustering, sparsity-induced generative adversarial hashing, contrastive language-image pre-training multimodal hashing, locality-sensitive hashing with query-based dynamic bucketing, deep supervised hashing, and cross-modal hashing methods. A computational experiment was conducted to comparatively analyze and evaluate the accuracy, loss function, and performance of the hashing algorithms: deep hashing, deep supervised hashing, and deep learning hashing. Python programs were developed for calculating the hashing algorithms and presenting graphical results. For multimodal tasks, where data from various sources must be integrated and supplemented with new data, deep lifelong cross-modal hashing is the most suitable solution. An analysis of deep hashing methods has demonstrated the superiority of deep supervised hashing when used with labeled data and distinct object classes.**

Keywords—**Hashing Methods, Deep Hashing, Deep Supervised Hashing, Deep Learning Hashing**

I. INTRODUCTION

Modern search engines widely use various hashing methods to handle large volumes of data. The use of hashing algorithms can significantly improve performance and reduce search time for large volumes of data.

Search engines widely use various hashing methods, each with its own advantages and disadvantages. Selecting the most effective hashing method can significantly improve search engine performance for large data sets. Determining the most effective hashing method for improving search engine performance is a pressing practical problem.

There are numerous scientific papers devoted to hashing methods. Here is a brief overview of current research.

In a research study [1], deep lifelong cross-modal hashing is proposed to solve the problem of continuous data arrival and high overtraining cost for updating hash functions in cross-modal retrieval. The study proposes a continuous learning method that updates hash functions by directly training on incremental data. As a result, the training time is reduced. To improve the performance, a lifetime hashing loss is introduced to ensure that the original hash codes remain invariant when participating in continuous training. The proposed method incorporates multi-label semantic similarity to control the distribution heterogeneity during continuous data arrival. Experiments on benchmark datasets showed more than 20% improvement in retrieval accuracy and more than 80% reduction in training time when new data arrives. The proposed deep lifelong cross-modal hashing method has significantly expanded the capabilities of hashing for cross-modal retrieval tasks. To improve performance on large volumes of data, the method uses deep learning and allows for fast execution of search queries due to the efficiency of processing nonlinear heterogeneous features.

The paper [2] presents a method called Learned Locality-Sensitive Hashing (LLSH), based on deep neural networks, that efficiently maps high-dimensional data to a lower-dimensional space. This method takes advantage of GPUs and deep neural networks to create an improved form of locality-sensitive hashing. LLSH replaces traditional Locality-Sensitive Hashing (LSH) families of functions with parallel multi-layer neural networks. This reduces both time and memory consumption while maintaining query accuracy. The paper presents experimental results on various datasets, which show that LLSH is highly efficient in terms of query accuracy, processing time, and memory usage.

In [3], a new hashing method called Graph-Collaborated Auto-Encoder Hashing for Multi-view Binary Clustering (GCAE) is proposed to solve the problems of using unsupervised hashing methods for large amounts of data. By training on compact binary data, this GCAE method reduces the storage and computation costs for binary clustering with a large number of views. The algorithm is dynamically trained on affinity graphs with low-rank constraints. The GCAE method uses joint training between autoencoders and affinity graphs to train a unified binary code. In this paper, a learning model is presented on affinity graphs with low-rank constraints, and the authors develop an encoder-decoder paradigm. The GCAE method uses an alternating iterative optimization scheme to obtain multi-view clustering results. The experimental results show the effectiveness of the method and high performance.

In the research paper [4], a new unsupervised hashing method called Sparsity-Induced Generative Adversarial Hashing (SiGAH) is proposed to encode large-scale multidimensional features into binary codes. The SiGAH

Manuscript received September 21, 2025.
Lubov Nikolaevna Ivanova, senior lecturer, Saint Petersburg State Marine Technical University (e-mail: 45is@mail.ru, https://orcid.org/0000-0001-6880-0897, Scopus Author ID: 57194244847)
Sergei Evgenevich Ivanov, candidate of physical and mathematical sciences, associate professor, ITMO National University (corresponding author e-mail: serg_ivanov@itmo.ru, https://orcid.org/0000-0002-2366-9458 , Scopus Author ID: 56457387600).

method uses a generative adversarial learning framework to ensure that the learned Hamming space matches the data distribution similar to the target metric space. A ReLU-based neural network is adopted as the generator and a mean-squared error-based autoencoder network is adopted as the discriminator. A compressed probing procedure is presented to generate synthetic features from hash codes. The experimental results show that the SiGAH method has high performance.

In [5], a new method, Contrastive Language-Image Pre-training multimodal hashing (CLIPMH), is proposed to solve the problem of low search accuracy in modern multimodal hashing methods. The CLIPMH method is used to extract and combine features of multimedia data and significantly improves the search performance using multimodal hashing. The experimental results show that the CLIPMH method outperforms modern unsupervised and supervised multimodal hashing methods.

The study [6] reviews various deep learning-based supervised hashing methods for large image datasets, focusing on the growth of big data. The generation of complex hash functions for nearest neighbor search is presented. The study classifies the methods based on the network architecture, training strategy, loss function type, and similarity measures used. The authors compare different datasets and discuss future directions such as incremental hashing and cross-modal hashing. The comparative analysis shows that generative adversarial network-based hashing models outperform other methods due to the use of larger data sets.

In [7], deep semantic hashing is considered to improve search performance by transforming source texts into compact hash codes. However, learning these codes is a challenging task due to significant information loss, uneven distribution of codes, and the presence of noise. In this paper, a general unsupervised semantic hashing (MASH) framework is proposed to learn balanced and compact hash codes. The encoder introduces a new relevance constraint among informative multidimensional representations to guide the learning of a compact hash code. External memory optimizes hash learning and improves search efficiency. An improved SMASH model is applied in this paper, which uses a noise-aware encoder-decoder structure. The experimental results demonstrate the high efficiency and performance of the MASH and SMASH methods.

The paper [8] proposes a new method DB-LSH (Locality-Sensitive Hashing with Query-based Dynamic Bucketing), which organizes projection spaces using multidimensional indices instead of fixed hash containers. It also proposes an incremental search technique DBI-LSH, which improves query performance by accessing the next best point. The paper presents a method DBA-LSH, which adaptively adjusts termination conditions without reducing the success probability. The paper conducts a theoretical analysis, which shows that DB-LSH provides lower query costs, and experiments on real data confirm its efficiency and accuracy.

The paper [9] considers the nearest neighbor problem, which plays a key role in such fields as computer vision and data mining. Here, hashing has become popular due to its computational efficiency and saving of disk space. The paper considers modern deep hashing algorithms, including supervised and unsupervised methods. Supervised methods are divided into pairwise ranking-based methods, point methods, and quantization. Unsupervised methods are divided into similarity reconstruction-based methods, pseudo-label methods, and self-supervised learning methods without prediction. The paper also considers semi-supervised deep hashing, domain-adapted deep hashing, and multimodal deep hashing.

In [10], a new approach to bipartite graph convolutional hashing, called the BGCH method, is presented to improve the efficiency of Top-N search using a convolutional graph network on bipartite graphs. The approach includes three modules: adaptive graph convolutional hashing, latent feature variance, and serialized Fourier gradient estimation. These modules preserve structural information despite the hashing loss, and the third module develops a frequency-domain Fourier series expansion for more accurate gradient estimation. The experimental results demonstrate the effectiveness of all the proposed components of the model.

In [11], hashing-based methods are successfully applied in cross-modal similarity search due to their high query speed and low storage cost. However, they face problems such as sensitivity to noise and outliers, quantization loss, and high computational requirements. A novel cross-modal search algorithm, RDMH, addresses these problems by using robust and discrete hashing with matrix factorization. The two-stage strategy improves robustness by directly learning hash codes and semantic labels without manipulating a large similarity matrix. The automatic encoding strategy preserves valuable information and improves efficiency. The developed RDMH algorithm has demonstrated high performance and efficiency in comprehensive experiments.

In [12], a weakly supervised deep multiple hashing (DMIH) method is proposed for multi-object image retrieval. The DMIH method uses a CNN model to establish an end-to-end relationship between a raw image and its binary hash codes corresponding to its multi-object data. It combines object detection with hashing learning by treating object detection as a binary multiple hashing problem. A two-level inverse index method is also proposed to speed up the retrieval of multi-object queries. The DMIH method shows high performance for multi-object queries.

The paper [13] addresses the problem of developing an efficient algorithm that balances construction time, search time, and representation space. In 1992, Fox, Chen, and Heath proposed an algorithm for fast search evaluation, but it was criticized for its long construction time and high memory consumption. The authors presented an improved algorithm that scales well to large data sets, reduces memory consumption without affecting search time, and provides 2-4 times better search time.

In [14], a visual pattern mining tool PSEUDo is presented, which is designed to address the problem of overfitting in deep learning methods for multivariate time series. It uses query-aware and locality-aware hashing to create a representation of multivariate time series. The performance of the tool is demonstrated using quantitative tests. The tool effectively detects patterns in multivariate time series, improving the results by using relevance feedback. This tool is important for using sensors and data warehouses for multivariate time series.

The paper [15] discusses multimodal hashing-based learning systems, which are popular due to their efficiency and low storage cost. The paper presents a new two-stage approach called the two-stage supervised discrete hashing (TSDH) method. The TSDH method generates hidden representations for each modality, maps them to a common Hamming space, and directly supplies hash codes with semantic labels. This approach avoids large quantization losses and improves the discrimination of hash functions. Experiments show the effectiveness of the TSDH method.

In [16], a novel architecture, modality-invariant asymmetric networks (MIAN), is proposed to overcome the semantic and heterogeneous gaps between different modalities in cross-media similarity retrieval applications. The MIAN architecture captures the intrinsic pairwise similarities for each modality using probabilistic asymmetric learning. MIAN also constructs a modality alignment network to extract visual features without redundancy and maximize the information about conditional bottlenecks between different modalities. This approach eliminates heterogeneity and domain bias, enabling the production of discriminant modality-invariant hash codes. The experimental results show the high performance of the MIAN architecture compared with cross-modal hashing methods.

The paper [17] considers vector databases (VDB) for multidimensional data management tasks. The paper presents an overview of various algorithms, hashing methods, data storage and retrieval. It also compares modern VDB solutions, their strengths and weaknesses, limitations and typical application scenarios. The paper also discusses new possibilities for interfacing VDB with large language models.

In [18], cross-modal hashing is considered for search engines and autopilot systems in similarity search problems. However, existing methods have limitations such as relaxation of discrete constraints, data loss, and conversion of real data points to binary codes. In this paper, a new scheme is proposed to project original data points into a low-dimensional latent space and find cluster centroid points using cluster unsupervised hashing (CUH). The proposed scheme jointly trains compact hash codes and linear hash functions. The experiments conducted show the effectiveness of the model in unsupervised cross-modal hashing problems.

The paper [19] considers the problem of cross-modal search when working with new, previously unencountered categories (zero-rank cross-modal search, ZSCMR). This problem is related to semantic inconsistency and semantic gap in heterogeneous data. In this paper, a new method is proposed to solve this problem - discrete bidirectional hashing with matrix factorization for ZSCMR (DMZCR). The DMZCR method uses bidirectional matrix factorization for discriminative representation, a multi-layer semantic transfer scheme, and discrete hashing to reduce quantization error. Experiments demonstrate the effectiveness of DMZCR for ZSCMR.

The paper [20] presents an indexing approach (Palm Hash Net) designed to speed up palmprint identification by transforming the search process into a constant-time operation. This is achieved by generating highly discriminant vector representations from palmprint images using a feature extraction network pre-trained with a modified Softmax loss algorithm. The index table is formed using locality-sensitive hashing (LSH), where hash values are used as indices. Query palmprints are matched with the most relevant index cells, thereby narrowing the search space to a small list of candidates. The presented method provides identification guarantees in order to speed up the search by tens of times.

The paper [21] presents DUCMH (Deep Uniform Cross-Modal Hashing), an end-to-end method designed to solve cross-modal hashing problems. DUCMH aims to simultaneously learn uniform hash codes and uniform hash functions by using interleaved learning and data alignment. For text data, hash codes are generated using uniform hash functions. For image data, DUCMH first performs auxiliary matching to align images with texts and then outputs hash codes from these matched texts. The method uses interleaved learning to update both uniform hash codes and functions. Experimental results show that DUCMH is an effective cross-modal hashing method.

In [22], cross-media hashing (CMH) methods are considered. Most CMH methods are single-stage, which makes optimization difficult. To solve this problem, a new two-stage discrete cross-media hashing method called WATCH is proposed. The WATCH method adaptively manages fields using a label relaxation strategy, which reduces quantization errors. The method is a two-stage model that uses a discrete smooth matrix factorization model to generate the hash code and an efficient hash function learning strategy to achieve even more efficient hashing. Experiments show that WATCH provides high performance.

In [23], an improved image hashing method called Bitwise Complementary Deep Attention Supervised Hashing (BADCSH) is proposed. This method trains a sequence of improved hash tables, each of which is trained by correcting previous errors. It uses features from different layers of the network to train different hash tables, revealing the image semantic content and structural information. The hash layer is used as an embedded layer to generate hash codes. In the method, a high-density attention layer is added to reduce redundancy and maintain overall similarity. Hash tables trained at different feature layers are combined using weights calculated based on their respective features. Experiments show that the BADCSH method has high performance.

In [24], a face identification system is proposed that uses a product quantization-based hash table to index and retrieve secure face templates. These templates are secured using fully homomorphic encryption schemes, which provides a high degree of security. The proposed method achieves a false negative identification rate of 0.0% to 0.2%. The paper presents a competitive workload reduction scheme with privacy preservation for pattern matching in an encrypted domain.

The paper [25] addresses cross-modal search for large-scale multimedia databases, but matrix factorization is often used to learn hash codes. In this paper, a new method called average approximate hashing (AAH) is proposed. The AAH method integrates locality and remainder preservation into the graph embedding structure by projecting data from different modalities into different semantic spaces and embedding a projection matrix. The AAH method obtains the final hash codes using the average approximation strategy.

In [26], a deep irrelevant discrete hashing (DUDH) method is proposed. It is a novel asymmetric deep hashing method designed for large-scale nearest neighbor search. It addresses the limitations of state-of-the-art asymmetric deep hashing methods that suffer from quantization errors and efficiency issues. The DUDH method decouples the query from the database by using a small set of images for similarity transfer to implicitly preserve the semantic structures from the database to the query. This strategy splits a large similarity matrix into two smaller ones. The method simultaneously trains both the database codes and the similarity transfer codes. Importantly, decoupling the query from the database means that the cost of training a CNN model for the query is independent of the database size. To further speed up the training, DUDH optimizes the similarity transfer codes, making their training cost negligible. Empirical evaluations show that the DUDH method significantly reduces the training cost by 30–50 times.

However, applying the many presented methods to specific practical search problems requires further research, identification of limitations, and comparative analysis.

Let's consider a class of modern deep hashing methods for search engines running on large data sets.

I. DEEP HASHING METHODS

Deep learning methods are based on training a deep neural network to generate binary codes—hashes. These methods allow for efficient storage and rapid retrieval of objects, such as images, among large volumes of data.

Deep learning methods map objects (images) into a space of binary vectors of minimal length, so that the distance between these vectors reflects the semantic similarity between the objects.

The deep learning algorithm employs data preparation. This preparation involves labeling the dataset with groups of images with a known degree of similarity. These groups help the neural network learn to correctly generate binary hashes.

Deep Hashing uses a network architecture based on Convolutional Neural Networks (CNN).

The basic network architecture consists of:

1. Feature extractor. A CNN is used to extract high-level features from images.
2. Hashing. A linear transformation layer is applied, followed by nonlinear activation to normalize the values before binarization.
3. Final binarization. The output real values are rounded to 0 or 1, creating the final binary vector.

The method employs a learning strategy to minimize the difference between distances in the feature space and semantic similarity between images. Various types of loss functions are used for this purpose. Contrastive loss minimizes the distance between pairs of similar images and maximizes the distance between dissimilar ones. Triplet loss considers three images simultaneously: a reference, a near, and a far image. It minimizes the distance to the near image and maximizes it to the far image. Cross-entropy loss is used in conjunction with classification methods, accounting for differences between classes.

The method utilizes stochastic gradient descent (SGD) or an Adam optimizer for optimization to update the neural network weights during training. To generate binary codes, each element of the normalized neural network output is binarized. This results in a short, fixed-size binary code.

Deep learning has the following advantages. Using binary codes ensures fast searches, reducing the cost of comparing objects. Binary codes also take up significantly less space, ensuring efficient storage. Furthermore, working with binary codes is faster and suitable for processing huge volumes of data, ensuring scalability. The method preserves some semantic information, allowing for efficient detection of similar images.

A drawback of the Deep Hashing method is the need for pre-labeling the data to effectively train the model. The method also faces difficulties with optimally choosing the length of the binary code. Furthermore, errors in preserving the precise distribution of features are common.

The method was further developed by adding an element of supervised learning, and the new method was named Deep Supervised Hashing (DSH). DSH uses class labeling to guide the generation of binary codes based on categories or labels.

The basic components of the DSH method are:

1. A convolutional neural network (CNN) for extracting features from images. Architectures such as AlexNet, VGG, or ResNet are commonly used.
2. A hash function transforms the extracted features into binary codes of a fixed length. A linear transformation and quantization are performed.
3. Special losses are applied to minimize the Hamming distance between similar objects and maximize the distance between dissimilar ones.

The DSH algorithm includes the following steps:

1. Data preparation: splitting the dataset into pairs or triplets.
2. CNN training: extracting deep features from images.
3. Binary code generation: applying a linear transformation and quantization to obtain binary representations.
4. Loss optimization: minimizing the distance between similar objects and maximizing the distance between dissimilar ones.
5. Quality evaluation: testing the effectiveness of the resulting binary codes on a test dataset.

The DSH method has the following advantages. Compact, small binary codes are easily indexed, speeding up search. Efficient binary comparison algorithms are used for fast search. The DSH method scales well and is applicable to large datasets. The method preserves semantic information about the proximity of objects.

A disadvantage of the DSH method is the need for a large amount of labeled data for effective model training. There is also the problem of choosing the optimal binary code length.

Another group of methods, Deep Learning Hashing (DLH), uses efficient hashing schemes to create short binary codes while preserving data semantics to speed up search.

The main stages of the DLH method include:

1. Data preprocessing, normalization, scaling, and standardization.

2. Feature extraction using a deep convolutional neural network (CNN). The CNN architecture used is AlexNet, VGG, GoogleNet, or ResNet.

3. Generation of binary codes. Additional linear layers are added, passing through nonlinear activation functions. Continuous features are then converted to binary values.

4. Loss minimization is performed. Special loss types are introduced: Contrastive loss and Triplet loss.

5. Generation of binary codes and storing them for fast lookup in a hash table.

DLH methods enable efficient search on large volumes of data using binary comparisons and the Hamming distance. Furthermore, DLH preserves small binary vectors, requiring negligible memory. The DLH method scales well and enables automatic feature extraction.

Another method, Deep Lifelong Cross-Modal Hashing (DLCMH), is designed to solve problems with data from different sources and supplemented with new data without complete retraining. DLCMH is designed to efficiently solve cross-modal image and text retrieval by constructing compact binary codes that preserve semantic similarity between different multimodal data. DLCMH incorporates deep neural network models to extract features from images and text separately. This allows features from different data types to be integrated into a single representation space.

The method jointly trains both data types to ensure representation consistency and minimize loss. It adapts to new data without losing previously acquired knowledge and uses optimized hashing functions to transform features into short binary codes.

Important advantages of DLCMH include high search speed, storage efficiency, compact binary codes, search accuracy across both modalities, adaptation to new data, and high search accuracy. The disadvantages of DLCMH are: the need for large amounts of labeled data, high computational power, the choice of the optimal binary code length, the cost of retraining, and difficulties in scalability.

II.EXPERIMENTAL COMPUTATIONS FOR HASHING METHODS

To compare Deep Hashing, Deep Supervised Hashing, and Deep Learning Hashing methods, the authors developed Python programs using TensorFlow, NumPy, OpenCV, Matplotlib, and PyTorch libraries.

The experiment used a generated dataset of 8,000 objects (320x320 pixel images), and computations were performed on an Intel Core i5-12400F, 32 GB RAM, 512 GB SSD, and NVIDIA GeForce RTX 3070 (8 GB). A batch size of 32 was used, which determines how many examples are processed in one training step. The number of epochs determines the number of complete passes through the entire dataset. The output binary codes are 32 bits long.

The main metrics of the methods are discussed: accuracy, loss, and training time.

Table 1 lists the main metrics of the methods.

Table 1 Main metrics of the methods.

METHODS	TEST LOSS	ACCURACY	TRAINING TIME (HOURS)
Deep Hashing	0.44	0.78	5
Deep Supervised Hashing	0.34	0.97	6
Deep Learning Hashing	0.39	0.93	5.5

For the Deep Hashing method, a test loss of 0.44 indicates that it performs poorly because it doesn't use class labels for training. This method has the lowest test accuracy of 0.78, but its training time of 5 hours is relatively short.

For the Deep Supervised Hashing method, a low test loss of 0.34 indicates that it adapted well to the data thanks to the use of class labels. This method has the highest test accuracy of 0.97, but its training time is slightly longer than 6 hours due to the model complexity and the use of labels.

For the Deep Learning Hashing method, a test loss of 0.39 is slightly higher than that of Deep Supervised Hashing, indicating insufficient optimization. This method has a good test accuracy of 0.93, and its training time of 5.5 hours is comparable to Deep Hashing, making it suitable for rapid testing.

Fig.1 shows a comparison of accuracy for Deep Hashing, Deep Supervised Hashing, and Deep Learning Hashing. The abscissa shows the initial epoch number, and the ordinate shows accuracy.

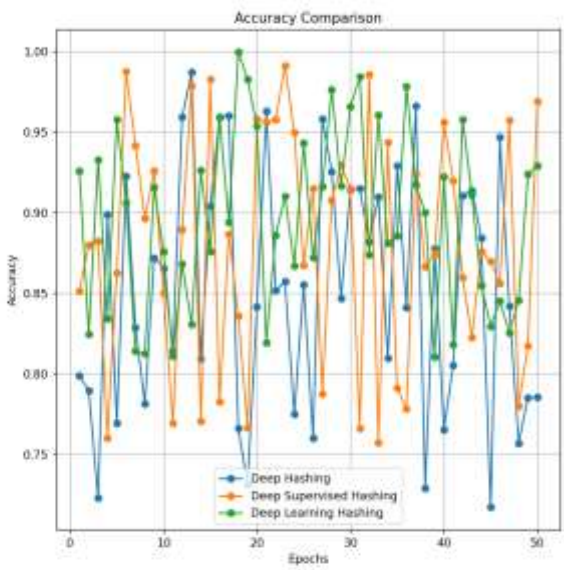

Figure 1. Comparison of accuracy for methods: Deep

Hashing, Deep Supervised Hashing and Deep Learning Hashing.

Among the methods considered, Deep Supervised Hashing shows the best results due to the use of class labels in the training process.

III. CONCLUSION

Various hashing methods for efficient access and rapid search of objects in large data sets are considered: deep lifelong cross-modal hashing, learned locality-sensitive hashing, graph-collaborated auto-encoder hashing for multi-view binary clustering, sparsity-induced generative adversarial hashing, contrastive language-image pre-training multimodal hashing, locality-sensitive hashing with query-based dynamic bucketing, deep supervised hashing, and cross-modal hashing methods. A computational experiment was conducted to comparatively analyze and evaluate the accuracy, loss function, and performance of hashing algorithms: deep hashing, deep supervised hashing, and deep learning hashing. Python programs were developed for calculating hashing algorithms and presenting graphical results. The choice of a deep hashing method depends on the characteristics of the problem being solved and the type and size of the dataset. An analysis of deep hashing methods has demonstrated the superiority of deep supervised hashing when used with labeled data and distinct object classes. For multimodality problems, where data from various sources must be considered and supplemented with new data, deep lifelong cross-modal hashing is the most suitable solution.

REFERENCES

[1] L. Xu, H. Li, B. Zheng, W. Li. and J. Lv, "Deep Lifelong Cross-modal Hashing," *IEEE Transactions on Circuits and Systems for Video Technology*, 2024.

[2] R. Liu, J. Zhao, X. Chu, Y. Liang, W. Zhou and J. He, "Can LSH (locality-sensitive hashing) be replaced by neural network?," *Soft Computing*, vol. 28, no. 2, pp. 1041-1053, 2024.

[3] H. Wang, M. Yao, G. Jiang, Z. Mi, and X. Fu, "Graph-collaborated auto-encoder hashing for multiview binary clustering," *IEEE Transactions on Neural Networks and Learning Systems*, vol.35, no.7, pp. 10121-10133, 2023.

[4] H. Liu, "Sparse-Inductive Generative Adversarial Hashing for Nearest Neighbor Search," *arXiv preprint*, arXiv:2306.06928, 2023.

[5] J. Zhu, M. Sheng, M. Ke, Z. Huang and J. Chang, "CLIP Multi-modal Hashing: A new baseline CLIPMH," *arXiv preprint*, arXiv:2308.11797, 2023.

[6] A. Singh and S. Gupta, "Learning to hash: a comprehensive survey of deep learning-based hashing methods," *Knowledge and Information Systems*, vol. 64., no. 10, pp. 2565-2597, 2022.

[7] L. He, Z. Huang, E. Chen, Q. Liu, S. Tong, H. Wang and S. Wang, "An efficient and robust semantic hashing framework for similar text search," *ACM Transactions on Information Systems*, vol. 41, no. 4, pp. 1-31, 2023.

[8] Y. Tian, X. Zhao and X. Zhou, "DB-LSH 2.0: Locality-sensitive hashing with query-based dynamic bucketing," *IEEE Transactions on Knowledge and Data Engineering*, vol. 36, no. 3, pp. 1000-1015, 2023.

[9] X. Luo, H. Wang, D. Wu, C. Chen, M. Deng, J. Huang and X. S. Hua, "A survey on deep hashing methods," *ACM Transactions on Knowledge Discovery from Data*, vol. 17, no. 1, pp. 1-50, 2023.

[10] Y. Chen, Y. Fang, Y. Zhang and I. King, "Bipartite graph convolutional hashing for effective and efficient top-n search in hamming space," In *Proceedings of the ACM Web Conference 2023*, pp. 3164-3172, 2023.

[11] D. Zhang and X. J. Wu, "Robust and discrete matrix factorization hashing for cross-modal retrieval," *Pattern Recognition*, vol. 122, pp.108343, 2022.

[12] W. Zhao, Z. Guan, H. Luo, J. Peng and J. Fan, "Deep multiple instances hashing for fast multi-object image search," *IEEE Transactions on Image Processing*, vol. 30, pp. 7995-8007, 2021.

[13] G. E. Pibiri and R. Trani, "PTHash: Revisiting FCH minimal perfect hashing," In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pp. 1339-1348, 2021.

[14] Y. Yu, D. Kruyff, J. Jiao, T. Becker and M. Behrisch, "PSEUDo: Interactive pattern search in multivariate time series with locality-sensitive hashing and relevance feedback," *IEEE Transactions on Visualization and Computer Graphics*, vol. 29, no. 1, pp. 33-42, 2022.

[15] D. Zhang, X. J. Wu, T. Xu and J. Kittler, "Two-stage supervised discrete hashing for cross-modal retrieval," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 52, no.11, pp. 7014-7026, 2022.

[16] Z. Zhang, H. Luo, L. Zhu, G. Lu and H. T. Shen, "Modality-invariant asymmetric networks for cross-modal hashing," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 5, pp. 5091-5104, 2022.

[17] Y. Han, C. Liu and P. Wang, "A comprehensive survey on vector database: Storage and retrieval technique, challenge," *arXiv preprint*, arXiv:2310.11703, 2023.

[18] L. Wang, J. Yang, M. Zareapoor and Z. Zheng, "Cluster-wise unsupervised hashing for cross-modal similarity search," *Pattern Recognition*, vol. 111, 107732, 2021.

[19] D. Zhang, X. J. Wu and J. Yu, "Discrete bidirectional matrix factorization hashing for zero-shot cross-media retrieval," In *Chinese conference on pattern recognition and computer vision (PRCV)*, Cham, Springer International Publishing, pp. 524-536, 2021.

[20] G. Arora, S. Kalra, A. Bhatia and K. Tiwari, "Palmhashnet: Palmprint hashing network for indexing large databases to boost identification," *IEEE Access*, no.9, pp. 145912-145928, 2021.

[21] Y. Wang, B. Xue, Q. Cheng, Y. Chen and L. Zhang, "Deep Unified Cross-Modality Hashing by Pairwise Data Alignment," *IJCAI*, pp. 1129-1135, 2021.

[22] D. Zhang, X. J. Wu, T. Xu and J. Kittler, "WATCH: Two-stage discrete cross-media hashing," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 6, pp. 6461-6474, 2022.

[23] W. W. Ng, J. Li, X. Tian and H. Wang, "Bit-wise attention deep complementary supervised hashing for image retrieval," *Multimedia Tools and Applications*, vol. 81, no. 1, pp. 927-951, 2022.

[24] D. Osorio-Roig, C. Rathgeb, P. Drozdowski and C. Busch, "Stable hash generation for efficient privacy-preserving face identification," *IEEE Transactions on Biometrics, Behavior, and Identity Science*, vol. 4, no. 3, pp. 333-348, 2021.

[25] X. Fang, K. Jiang, N. Han, S. Teng, G. Zhou and S. Xie, "Average approximate hashing-based double projections learning for cross-modal retrieval," *IEEE Transactions on Cybernetics*, vol. 52, no. 11, pp. 11780-11793, 2021.

[26] D. Wu, Q. Dai, B. Li and W. Wang, "Deep uncoupled discrete hashing via similarity matrix decomposition," *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 19, no. 1, pp. 1-22, 2023.

Lubov Nikolaevna Ivanova, senior lecturer, Saint Petersburg State Marine Technical University (e-mail: 45is@mail.ru, https://orcid.org/0000-0001-6880-0897, Scopus Author ID: 57194244847)

Sergei Evgenevich Ivanov, candidate of physical and mathematical sciences, associate professor, ITMO National University (corresponding author e-mail: serg_ivanov@itmo.ru, https://orcid.org/0000-0002-2366-9458 , Scopus Author ID: 56457387600).