Deep Hashing Regularization Towards Hamming Space Retrieval

Christos Nasioutzikis
Aristotle University of Thessaloniki
Department of Informatics
nasioutz@csd.auth.gr

Maria Tzelepi
Aristotle University of Thessaloniki
Department of Informatics
mtzelepi@csd.auth.gr

Anastasios Tefas
Aristotle University of Thessaloniki
Department of Informatics
tefas@csd.auth.gr

ABSTRACT

Hashing methods have been extensively used so as to resolve image and video retrieval problems due to their computation and storage efficiency. During the recent years, following the impressive performance of deep learning methods in various computer vision tasks, deep hashing methods have powerfully emerged in the field of large scale image and video retrieval, accomplishing superior performance over the previous approaches. In this paper, a novel regularization framework is proposed, on the top of Deep Cauchy Hashing method, for improving the performance of the produced hash codes towards Hamming space retrieval. The proposed framework includes two regularization approaches, namely *Class-Agnostic* regularizer and *Class-Aware* regularizer. The experimental evaluation on two retrieval datasets validates the efficiency of both the proposed approaches in improving the retrieval performance, outperforming previous state-of-the-art approaches.

KEYWORDS

Hashing, Deep Cauchy Hashing, Image Retrieval, Hamming Space Retrieval, Regularization, Class-Agnostic Regularizer, Class-Aware Regularizer.

ACM Reference Format:

Christos Nasioutzikis, Maria Tzelepi, and Anastasios Tefas. 2021. Deep Hashing Regularization Towards Hamming Space Retrieval. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Image Retrieval (IR) refers to the task of obtaining relevant images from a large collection given a query image. IR is a vivid research field since 1990s. Earlier works focus on primitive features describing the image content such as texture, color, etc., while more recent works have striven towards finding semantically richer representations, such as [18]. Motivated by the successful performance of Deep learning algorithms [5] and especially deep Convolutional Neural Networks (CNN) in a wide range of computer vision tasks [4, 24], deep CNNs introduced to image retrieval field [22]. However, despite their outstanding performance, deep learning based techniques, still suffer from major limitations that restrict their efficiency. That is, they are computationally and memory demanding. During the recent years, due to the explosive growth of available images and videos on the web, hashing techniques have been established as an effective solution for image and video retrieval tasks [26]. Hashing methods project the high dimensional feature representations of images into low dimensional binary codes, circumventing the storage and memory limitations, while they are also computationally efficient since computing Hamming distance is of low complexity. Earlier hashing works focus on data independent methods, e.g. Locality Sensitive Hashing (LSH) [12]. However, these methods generally require long hash codes to achieve satisfactory results, while the semantic information of data is also ignored. To this end, recent works focus on data dependent methods (also known as *learning to hash*), where hash functions are learned from the data, accomplishing superior performance over the data independent ones [11] utilizing shorter binary codes.

Recently, following the success of Deep Learning algorithms [5] in a wide spectrum of computer vision tasks, deep learning based hashing methods were introduced in the field of image retrieval, achieving superior performance over the previous approaches, by proposing to learn simultaneously the feature representations and the hash codes [14]. Subsequently, since previous deep hashing methods focus on maximizing the retrieval performance based on linear hash codes which remains costly despite of the hash codes, Deep Cauchy Hashing (DCH), [1], turns to hamming space retrieval [17], where instead of linear scan as in the existing hashing methods, relevant images to a query image are retrieved within a given Hamming radius using hash table lookups.

In this paper, we propose a regularization framework on the top of DCH method for improving the performance of binary hash codes towards Hamming space retrieval. Generally, regularization techniques have been used widely in order to improve the generalization ability of deep learning models. Common regularization techniques include L1, L2 regularization which penalize large weights during the network's optimization, and Dropout where for each training sample a randomly selected subset of the activations is zeroed in each epoch. Besides, multitask-learning [3] has been proposed as a way to improve the generalization ability of a model.

The proposed regularization framework includes two approaches namely *Class-Agnostic* regularizer and *Class-Aware* regularizer. As the main learning objective in DCH penalizes on similar image pairs with Hamming distance larger than the given radius threshold, both the proposed regularizers induce the hash code representations to shrink, while the Class-Aware regularizer also preserves discriminant power.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

The rest of the paper is organized as follows. Section 2 discusses relevant hashing techniques towards image retrieval. The proposed regularization techniques are described in Section 3. Subsequently, the experiments conducted in order to validate the proposed method are presented in Section 4. Finally, conclusions are drawn in Section 5.

2 PREVIOUS WORK

In this Section we briefly survey recent hashing techniques for the problem of image retrieval. As previously mentioned, hashing methods can be divided into two broad categories, that is data independent and data depended methods. In this Section we focus on data-depended hashing methods, since they are more accurate and require smaller hash codes than the data-independent.

Several unsupervised hashing methods have been proposed in the recent literature. Common strategies include graph embedding [27], and reconstruction error minimization [9]. Semi-supervised methods have also been recently proposed. For example, a semi-supervised hashing framework, which minimizes the empirical error over the labeled pairs and prevents overfitting utilizing an information-theoretic regularizer over both labeled and unlabeled data is proposed in [25].

Several supervised hashing methods have also recently been proposed. A method that balances the discrimination and learnability of hash codes is presented in [19]. That is, the method learns codes that maximize separability of classes unless there is strong visual evidence against this. Subsequently, a method that uses Linear Discriminant Analysis to reduce the covariance of similar feature representations while increasing the covariance between dissimilar ones is presented in [21]. Finally, other supervised hashing methods propose to generate nonlinear or discrete hash codes by minimizing the Hamming distances across similar pairs of data points and maximizing the Hamming distances across dissimilar pairs, [20].

Recently, deep learning based hashing methods introduced in the field of image retrieval accomplishing superior performance over previous approaches. Convolutional Neural Network Hashing (CNNH) [28] learns approximate hash codes and then uses them to learn a hash function in a two step approach, while Deep Neural Network Hashing (DNNH) [14] improves on CNNH by learning features and hash functions simultaneously effectively improving each other in the process. Deep Hashing Network (DHN) [30] is the first method that both preserves semantic similarities and manages the quantization error by using cross-entropy loss, whilst Deep Supervised Hashing [15] improves on that by using a loss function that maximizes discriminability of the output to approximate discrete values. Hashnet [2] is another architecture for deep learning to hash by continuation method from imbalanced similarity data. Subsequently, a method that proposes to learn hash functions by optimizing tie-aware ranking metrics, is presented in [10].
Subsequently, DHA [29] proposes to scale and shift the loss function avoiding in this way the saturation of gradients during training, while simultaneously to adjust the loss so as to adapt to different levels of similarities of data. Finally, Deep Cauchy Hashing (DCH) [1] produces compact and concentrated hash codes for efficient Hamming space retrieval using a pairwise cross-entropy loss based on Cauchy distribution. In this paper, we propose a regularization framework on the top of the DCH method for improving the performance of the method towards Hamming space retrieval.

3 PROPOSED METHOD

In this work, we propose a novel regularization technique on the top of Deep Cauchy Hashing method. A brief description of DCH method follows below.

3.1 Deep Cauchy Hashing Overview

The utilized deep architecture is based on AlexNet model [13] which originally contains five convolutional layers and three fully connected layers. The last fully connected layer of AlexNet is replaced by a new hash layer with K hidden units, transforming the representation of the penultimate fully connected layer into K-dimensional continuous output $\boldsymbol{z}_i \in \mathbb{R}^K$ for each input image $\boldsymbol{x}_i$. Hash code is obtained through sign thresholding $h_i = \operatorname{sgn}(z_i)$. The hyperbolic tangent (tanh) function is utilized to squash z_i into $[-1, 1]$, in order to overcome the shortcoming of sign function pertaining to ill-posed gradient. Furthermore two loss functions are introduced, one to preserve pairwise similarity and another to correct the quantization error of the above approximation. Both of them are based on the Cauchy Distribution and derived in the Maximum a Posteriori estimation framework. Two loss functions are used based on the long-tailed Cauchy distribution. That is, a pairwise Cauchy cross-entropy loss and a pointwise Cauchy quantization loss. Both the utilized losses derive in the Maximum a Posteriori estimation framework.
Given a training set of N images, each image is represented as a D-dimensional vector $\boldsymbol{x}_i \in \mathbb{R}^D$, and for each pair of images $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, a similarity label $s_{ij} \in \mathbb{S}$ is produced, where $s_{ij} = 1$ indicates similar pairs, while $s_{ij} = 0$ indicated dissimilar ones. The logarithm Maximimum a Posteriori estimation of the hash codes $\boldsymbol{H} = [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_N]$ is defined as:

$$\log P(\boldsymbol{H}|S) \propto \log P(S|\boldsymbol{H})P(\boldsymbol{H}) = \sum_{s_{ij} \in S} w_{ij} \log P(s_{ij}|\boldsymbol{h}_i, \boldsymbol{h}_j) + \sum_{i=1}^{N} \log P(\boldsymbol{h}_i) \tag{1}$$

where $P(S|\boldsymbol{H}) = \Pi_{s_{ij} \in S}[P(s_{ij}|\boldsymbol{h}_i, \boldsymbol{h}_j)]^{w_{ij}}$ is the weighted likelihood function, and w_{ij} is the weight for each training pair $(\boldsymbol{x}_i, \boldsymbol{x}_j, s_{ij})$. In order to weight the image pairs according to their miss-classification importance, w_{ij} is defined as

$$w_{ij} = \begin{cases} |S|/|S_1|, & s_{ij} = 1 \\ |S|/|S_0|, & s_{ij} = 0 \end{cases} \tag{2}$$

where $S_1 = \{s_{ij} \in S : s_{ij} = 1\}$ is the set of similar pairs and $S_0 = \{s_{ij} \in S : s_{ij} = 0\}$ is the set of dissimilar pairs and $P(s_{ij}|\boldsymbol{h}_i, \boldsymbol{h}_j)$ is the conditional probability of similarity label s_{ij} given a pair of hash codes $\boldsymbol{h}_i$ and $\boldsymbol{h}_j$, which can be naturally defined by the Bernoulli distribution,

$$\begin{aligned} P(s_{ij} \mid \boldsymbol{h}_i, \boldsymbol{h}_j) &= \begin{cases} \sigma\left(\mathrm{d}\left(\boldsymbol{h}_i, \boldsymbol{h}_j\right)\right), & s_{ij} = 1 \\ 1 - \sigma\left(\mathrm{d}\left(\boldsymbol{h}_i, \boldsymbol{h}_j\right)\right), & s_{ij} = 0 \end{cases} \\ &= \sigma(\mathrm{d}(\boldsymbol{h}_i, \boldsymbol{h}_j))^{s_{ij}}(1 - \sigma(\mathrm{d}(\boldsymbol{h}_i, \boldsymbol{h}_j)))^{1-s_{ij}}, \end{aligned} \tag{3}$$

where $\mathrm{d}(\boldsymbol{h}_i,\ \boldsymbol{h}_j)$ is used to denote the Hamming distance between hash codes $\boldsymbol{h}_i$ and $\boldsymbol{h}_j$, and σ a well-defined probability function based on the Cauchy distribution. That is,

$$\sigma(\mathrm{d}(h_i,\ h_j)) = \frac{\gamma}{\gamma + \mathrm{d}(\boldsymbol{h}_i,\ \boldsymbol{h}_j)}, \tag{4}$$

where γ is the scale parameter of the Cauchy distribution. In addition, in order to control the quantization error a prior for each individual hash code $\boldsymbol{h}_i$ is proposed:

$$P(\boldsymbol{h}_i) = \frac{\gamma}{\gamma + \mathrm{d}(|\boldsymbol{h}_i|,\ 1)}, \tag{5}$$

where $1 \in \mathbb{R}^K$ is the vector of ones. Since continuous relaxation is used, Hamming distance is approximated as: $\mathrm{d}(\boldsymbol{h}_i,\ \boldsymbol{h}_j) = \frac{K}{2}(1 - \cos(\boldsymbol{h}_i, \boldsymbol{h}_j)$.

The final optimization problem is formulated as: $\min_{\Theta} L + \lambda Q$ (6), where λ is a hyper-parameter to trade-off the Cauchy cross-entropy loss L and the Cauchy quantization loss Q, and Θ denotes the set of network parameters to be optimized. Specifically, the Cauchy cross-entropy loss L is formulated as:

$$L = \sum_{s_{ij} \in S} w_{ij}(s_{ij} \log \frac{\mathrm{d}(\boldsymbol{h}_i, \boldsymbol{h}_j)}{\gamma} + \log(1 + \frac{\gamma}{\mathrm{d}(\boldsymbol{h}_i, \boldsymbol{h}_j)})), \tag{7}$$

and similarly, the Cauchy quantization loss is derived as:

$$Q = \sum_{i=1}^{N} \log(1 + \frac{\mathrm{d}(|\boldsymbol{h}_i|, 1)}{\gamma}), \tag{8}$$

where $d(.,.)$ corresponds either to the Hamming distance between the hash codes or to the normalized Euclidean distance between the continuous codes.

3.2 Regularization Framework

In this paper, two regularizers, that is the Class-Agnostic regularizer and the Class-Aware regularizer is proposed to be attached to the last layer of the network, that is the hash layer. The output vectors $\boldsymbol{H} = [\boldsymbol{h}_1, \cdots, \boldsymbol{h}_N]$ of the last layer represent the continuous hash code representations bounded by the tanh function for each of the total number of N image samples. Each of the proposed regularizers, as it is shown in the subsequent subsections, defines a distinct target, based on the utilization of the class label information. Then the new optimization problem is formulated as: $\min_{\Theta} L + \lambda Q + \lambda_r R$ (9), where R defines the additional regularization loss, and λ_r is a hyper-parameter to trade-off the Regularization loss R with both Cauchy cross-entropy loss L and the Cauchy quantization loss Q.

3.2.1 Class-Agnostic Regularizer. In the Class-Agnostic regularization approach, we apply an additional objective which aims at minimizing the variance among the hash codes. That is, considering the mini-batch training procedure, we propose to regularize the main learning objective which significantly penalizes on similar image pairs with Hamming distance larger than the given radius threshold, by forcing the hash codes come closer to their batch center. The Class-Agnostic regularizer is rooted in the radius-margin based SVM. That is, according to [6, 7] the performance of the max margin methods in classification, depends not only to the margin between positive and negative samples, but also to the radius of the enclosing ball of all samples. Thus, it has been shown [23], that as the main objective aims at distinguishing the training samples, it is useful to attach a regularizer which aims at at shrinking the radius of the minimum enclosing ball of all the training samples' representations.

Thus, considering N images and their continuous hash representations $H = [\boldsymbol{h}_1, \cdots, \boldsymbol{h}_N]$ the regularization objective is formulated as: $R = \frac{1}{N}\sum_{i=1}^{N} ||\mathbf{h}_i - \boldsymbol{\mu}_b||_2^2$, where $\boldsymbol{\mu}_b = \frac{1}{N}\sum_{i=1}^{N} \mathbf{h}_i$.

3.2.2 Class-Aware Regularizer. In the Class-Aware regularization approach, we apply an additional objective which aims at best separating the hash codes belonging to different classes producing more discriminative hash codes. We draw inspiration from the Linear Discriminant Analysis (LDA) algorithm [8], that aims at best separating training samples belonging to different classes, by projecting them into a new low-dimensional space, so as the between-class separability is maximized, while the within-class variability is minimized. Thus, considering that the main learning objective that penalizes on similar image pairs with Hamming distance larger than the given radius threshold preserves the between-class separability, the proposed objective encourages hash codes belonging to the same class to become more compact by coming closer to their class centroid.

Thus, considering N images and their continuous hash representations $H = [\boldsymbol{h}_1, \cdots, \boldsymbol{h}_N]$, and denoting as C^i the set of hash codes of images belonging to the same class to the image with hash code $\boldsymbol{h}_i$, the regularization objective is formulated as: $R = \frac{1}{N}\sum_{i=1}^{N} ||\mathbf{h}_i - \boldsymbol{\mu}_c^i||_2^2$, where $\boldsymbol{\mu}_c^i = \frac{1}{|C^i|}\sum_j \mathbf{h}_j$.

4 EXPERIMENTS

In order to evaluate the proposed regularization framework we conduct experiments on two image retrieval datasets, that is CIFAR-10, and MS-COCO. CIFAR-10 which contains 60000, 32×32 color images divided in 10 classes. The training set is formed using 500 images per class, while the test set is formed using 100 per class. MS-COCO which contains 82,783 training and 40.504 validation images, each labeled with multiple of 80 semantic concepts.

4.1 Experimental Setup

The proposed method is implemented in Tensorflow framework and trained on an NVIDIA GeForce GTX 1080 Ti with 12 GB of memory, as well as on an NVIDIA GeForce RTX 2080 with 8 GB of memory. The parameter λ_r in eq. (9) is set to 0.025, since we have seen that in most cases provides best performance. Throughout this work we use Mean Average Precision (MAP) within Hamming Radius 2 (MAP@H$\leq$ 2) to evaluate the proposed method. We note that we have used the same configuration as in [1] for fair comparisons.

4.2 Experimental Results

Experimental results for both the proposed regularizers utilizing various code lengths, as well as comparisons against the baseline DCH method and other state-of-the art hashing methods are presented in Table 1. The proposed Class-Agnostic regularizer is abbreviated as CAg, while the Class-Aware as CAw. Best results are printed in bold.

Table 1: Comparison against other deep hashing methods

Method	MS-COCO				CIFAR-10			
	16 bits	32 bits	48 bits	64 bits	16 bits	32 bits	48 bits	64 bits
KSH [16]	0.5797	0.5532	0.2338	0.0216	0.4368	0.4585	0.4012	0.3819
SDH [20]	0.6449	0.6766	0.5226	0.5108	0.5620	0.6428	0.6069	0.5012
CNNH [28]	0.5602	0.5685	0.5376	0.5058	0.5512	0.5468	0.5454	0.5364
DNNH [14]	0.5771	0.6023	0.5235	0.5013	0.5703	0.5985	0.6421	0.6118
DNH [30]	0.6901	0.7021	0.6685	0.5664	0.6929	0.6445	0.5835	0.5883
HashNet [2]	0.6851	0.6900	0.5589	0.5344	0.7446	0.7776	0.6399	0.6259
DCH [1]	0.7010	**0.7576**	0.7251	0.7013	0.7901	0.7979	0.8071	0.7936
DCH & CAg	**0.7129**	0.7493	0.730	**0.7381**	0.8004	**0.8148**	**0.8136**	**0.8209**
DCH & CAw	0.7106	0.7402	**0.7435**	0.7351	**0.8036**	0.8123	0.8134	0.8106

From the demonstrated results several remarks can be drawn. First, we can see that both the Class-Agnostic and Class-Aware regularizers improve the DCH method in all the considered cases except for one case, that is hash code of length 32 in the MS-COCO dataset. Furthermore, we can see that the Class-Agnostic regularizer performs better than the Class-Aware in the most of the considered cases. We can also see that we can achieve better improvements with longer code lengths. Finally, we can observe that the proposed regularizers are superior over the compared state-of-the-art deep hashing methods, in all the considered cases except for one case.

5 CONCLUSIONS

In this paper, we proposed a regularization framework on the top of Deep Cauchy Hashing method, for improving the performance of the produced hash codes towards Hamming space retrieval. Two different regularizers are proposed, based on the class label information utilization. That is, Class-Agnostic regularizer and Class-Aware regularizer. Experiments conducted on two image retrieval datasets, validating the effectiveness of both the proposed regularizers in improving the performance of the hash codes in the retrieval task, also outperforming state-of-the-art deep hashing methods.

ACKNOWLEDGMENTS

This research has been financially supported by General Secretariat for Research and Technology (GSRT) and the Hellenic Foundation for Research and Innovation (HFRI) (Scholarship Code: 2826.)

REFERENCES

[1] Yue Cao, Mingsheng Long, Bin Liu, and Jianmin Wang. 2018. Deep cauchy hashing for hamming space retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1229–1237.

[2] Zhangjie Cao, Mingsheng Long, Jianmin Wang, and Philip S Yu. 2017. Hashnet: Deep learning to hash by continuation. In *Proceedings of the IEEE international conference on computer vision*. 5608–5617.

[3] Rich Caruana. 1997. Multitask learning. *Machine learning* 28, 1 (1997), 41–75.

[4] Eleftherios Daskalakis, Maria Tzelepi, and Anastasios Tefas. 2018. Learning deep spatiotemporal features for video captioning. *Pattern Recognition Letters* 116 (2018), 143–149.

[5] Li Deng. 2014. A tutorial survey of architectures, algorithms, and applications for deep learning. *APSIPA Transactions on Signal and Information Processing* 3 (2014).

[6] Huyen Do, Alexandros Kalousis, and Melanie Hilario. 2009. Feature weighting using margin and radius based error bound optimization in svms. *Machine Learning and Knowledge Discovery in Databases* (2009), 315–329.

[7] Huyen Do, Alexandros Kalousis, Adam Woznica, and Melanie Hilario. 2009. Margin and radius based multiple kernel learning. *Machine Learning and Knowledge Discovery in Databases* (2009), 330–343.

[8] Ronald A Fisher. 1936. The use of multiple measurements in taxonomic problems. *Annals of eugenics* 7, 2 (1936), 179–188.

[9] Yunchao Gong, Svetlana Lazebnik, Albert Gordo, and Florent Perronnin. 2012. Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35, 12 (2012), 2916–2929.

[10] Kun He, Fatih Cakir, Sarah Adel Bargal, and Stan Sclaroff. 2018. Hashing as tie-aware learning to rank. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4023–4032.

[11] Kaiming He, Fang Wen, and Jian Sun. 2013. K-means hashing: An affinity-preserving quantization method for learning binary compact codes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2938–2945.

[12] Piotr Indyk. 2006. Near optimal hashing algorithms for approximate near (est) neighbor problem. In *MMDS 2006. Workshop on Algorithms for Modern Massive Data Sets, Stanford, USA*.

[13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.

[14] Hanjiang Lai, Yan Pan, Ye Liu, and Shuicheng Yan. 2015. Simultaneous feature learning and hash coding with deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3270–3278.

[15] Haomiao Liu, Ruiping Wang, Shiguang Shan, and Xilin Chen. 2016. Deep supervised hashing for fast image retrieval. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2064–2072.

[16] Wei Liu, Jun Wang, Rongrong Ji, Yu-Gang Jiang, and Shih-Fu Chang. 2012. Supervised hashing with kernels. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2074–2081.

[17] Mohammad Norouzi, Ali Punjani, and David J Fleet. 2012. Fast search in hamming space with multi-index hashing. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 3108–3115.

[18] Florent Perronnin, Yan Liu, Jorge Sánchez, and Hervé Poirier. 2010. Large-scale image retrieval with compressed fisher vectors. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 3384–3391.

[19] Mohammad Rastegari, Ali Farhadi, and David Forsyth. 2012. Attribute discovery via predictable discriminative binary codes. In *European Conference on Computer Vision*. Springer, 876–889.

[20] Fumin Shen, Chunhua Shen, Wei Liu, and Heng Tao Shen. 2015. Supervised discrete hashing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 37–45.

[21] Christoph Strecha, Alex Bronstein, Michael Bronstein, and Pascal Fua. 2011. LDAHash: Improved matching with smaller descriptors. *IEEE transactions on pattern analysis and machine intelligence* 34, 1 (2011), 66–78.

[22] Maria Tzelepi and Anastasios Tefas. 2016. Exploiting supervised learning for finetuning deep CNNs in content based image retrieval. In *2016 23rd International Conference on Pattern Recognition (ICPR)*. IEEE, 2918–2923.

[23] Maria Tzelepi and Anastasios Tefas. 2019. Graph Embedded Convolutional Neural Networks in Human Crowd Detection for Drone Flight Safety. *IEEE Transactions on Emerging Topics in Computational Intelligence* (2019).

[24] Maria Tzelepi and Anastasios Tefas. 2020. Improving the performance of lightweight CNNs for binary classification using Quadratic Mutual Information regularization. *Pattern Recognition* (2020), 107407.

[25] Jun Wang, Sanjiv Kumar, and Shih-Fu Chang. 2012. Semi-supervised hashing for large-scale search. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34, 12 (2012), 2393–2406.

[26] Jingdong Wang, Ting Zhang, Nicu Sebe, Heng Tao Shen, et al. 2017. A survey on learning to hash. *IEEE transactions on pattern analysis and machine intelligence* 40, 4 (2017), 769–790.

[27] Yair Weiss, Antonio Torralba, and Rob Fergus. 2009. Spectral hashing. In *Advances in neural information processing systems*. 1753–1760.

[28] Rongkai Xia, Yan Pan, Hanjiang Lai, Cong Liu, and Shuicheng Yan. 2014. Supervised hashing for image retrieval via image representation learning. In *Twenty-eighth AAAI conference on artificial intelligence*.

[29] Jiehao Xu, Chengyu Guo, Qingjie Liu, Jie Qin, Yunhong Wang, and Li Liu. 2019. DHA: Supervised Deep Learning to Hash with an Adaptive Loss Function. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*. 0–0.

[30] Han Zhu, Mingsheng Long, Jianmin Wang, and Yue Cao. 2016. Deep hashing network for efficient similarity retrieval. In *Thirtieth AAAI Conference on Artificial Intelligence*.