



# Deep Super-Resolution Hashing Network for Low-Resolution Image Retrieval

Feng Dai<sup>1</sup>, Zhuangzi Li<sup>1(✉)</sup>, Naiguang Zhang<sup>2(✉)</sup>, Qian Wang<sup>3</sup>, Xiaobin Zhu<sup>4</sup>,  
and Peng Li<sup>5</sup>

<sup>1</sup> School of Computer and Information Engineering,  
Beijing Technology and Business University, Beijing, China  
daifeng1994@163.com, lizhuangzii@163.com

<sup>2</sup> Information Technology Institute, Academy of Broadcasting Science,  
NRTA, Beijing, China  
zhangnaiguang@abs.ac.cn

<sup>3</sup> Beijing Goldwind Science & Creation Windpower Equipment Co., Ltd.,  
Beijing, China  
wangqian36084@goldwind.com.cn

<sup>4</sup> School of Computer and Communication Engineering,  
University of Science and Technology Beijing, Beijing, China  
zhuxiaobin@ustb.edu.cn

<sup>5</sup> College of Information and Control Engineering,  
China University of Petroleum, Beijing, China  
lipeng@upc.edu.cn

**Abstract.** In image retrieval, deep learning based hashing approaches have achieved promising performance in recent years. However, they are usually trained on a specific resolution, so it will cause unpleasant retrieval results with low-resolution input. In this paper, we propose a novel end-to-end deep super-resolution hashing network (DSRHN) for low-resolution image retrieval. It aims to adopt super-resolution techniques to promote semantic information of low-resolution images, so that benefits hashing code generation in a more representative fashion. The proposed network consists of two major components, which are trained alternatively, named super-resolution network and hashing network. The super-resolution network is not only optimized by MSE loss for pixel-wise image reconstruction, but also optimized by perceptual loss extracted by the hashing network for semantic learning. As for the hashing network, we adopt hashing semantic loss to optimize it for accurate hash code generation, and utilize a discriminative loss to improve the discriminative ability for the super-resolved images and high-resolution images. Extensive experiments show that our method achieve state-of-the-art performance on low-resolution images retrieval.

**Keywords:** Image retrieval · Deep hashing · Image super-resolution

---

F. Dai and Z. Li—These authors contributed equally to this paper and share the first authorship.

# 1 Introduction

Image similarity search in the era of big data has attracted wide attention in different applications such as information retrieval, data mining and pattern recognition. In order to efficiently store and real-time match millions of images, images discriminative representation in huge datasets has become an important research direction. Many existing methods represent images as binary hash codes by hashing function, so that the similarity search of high-dimensional images is replaced by calculating Hamming distance [18]. In addition, hashing functions are robust to various image transformations such as rotation, translation, scale and lightning, since they are carefully designed to extract distinctive patterns from images.

Many methods of learning to hash [8, 22] have been proposed to achieve efficient image retrieval. But the traditional learning-based approach can not effectively represent images feature [16, 17, 30, 32]. Recently, deep learning based hashing methods [13, 27, 34] have shown that deep neural networks can enable end-to-end representation learning and hash coding with nonlinear hash functions. And it achieve advanced performance. However, in the real world, the retrieval images is not necessarily the same quality as the image of the training model, and the images resolution often decreases due to some factors. Existing deep hashing approaches cannot effectively represent LR images with binary hash codes, which may result in poor hash retrieval. In order to solve the abovementioned problems, this paper presents an end-to-end deep hash model (DSRHN) to generate efficient binary hash codes directly from LR images. To the best of our knowledge, our approach is the first attempt to use an end-to-end multitask framework [19] for LR images hashing tasks. We not only learn the intensity similarity mapping between HR images and LR images, but also explore the mapping of images in Hamming space. DSRHN consists of two parts: a super-resolution network (SRNet) and a hash encoding network (HashNet). SRNet are trained to produce SR images from LR images. And the HashNet is trained to generate binary hash code from images, which is also used to constrain the SR images generated by SRNet and the corresponding HR images to be consistent on hash semantics. Because of this limitation, the retrieval result of LR images are close to that of HR images.

In short, we make contributions of our work. (1) We propose a novel end-to-end learning [28, 29, 38] framework for LR images retrieval. This method allows LR images to get more semantic information via high-efficiency repair ability of super-resolution network. Thus achieving efficient LR images retrieval. (2) We conduct extensive experiments on two benchmark datasets and achieve state-of-the-art performance. The rest of the paper is organized as follows: Sect. 2 describes the related work. Sections 3 and 4 introduce the proposed method and experimental results respectively. Section 5 concludes the paper.

## 2 Related Work

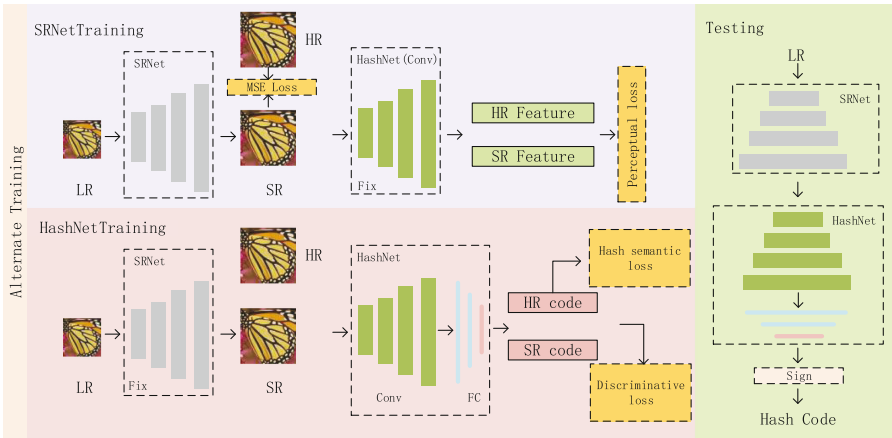
### 2.1 Image Hashing Method

In general, hashing methods can be divided into data-independent and data-dependent methods. Locality Sensitive Hash (LSH) [7] was one of the data-independent method of early research. LSH uses random linear projections to map the original data to a low-dimensional feature space, and then obtains binary hash codes. LSH and its several variants (e.g., kernel LSH [12] and p-norm LSH [4]) are widely used for large-scale image retrieval. However, data-independent methods are affected by some factors such as low efficiency and the need for longer hash codes, so it cannot be effectively used in practical applications. Due to the limitations of data-independent methods, the current research on hash function mainly uses various machine learning techniques for given datasets. Data-independent methods can be further divided into supervised, semi-supervised, and unsupervised methods. The unsupervised hash method directly learns the hash function from unlabeled data points and represents the data points as binary codes. Typical learning criteria methods include reconstruction error minimization [1, 33, 37] and graph structure learning [23, 26]. Iterative Quantization (ITQ) [8] is an unsupervised method of generating binary codes by achieving better quantization rather than random projection. The semi-supervised hashing method improves the quality of the hash code by leveraging the supervisory information into learning process. Semi-Supervised Hashing (SSH) [25] is one of the semi-supervised methods that uses pairwise information on labeled samples to preserve semantic similarity. Compared to unsupervised and semi-supervised methods, supervised methods use semantic labels to improve performance. One of the representatives is Kernel-based Supervised Hashing (KSH) [22] to generate high quality hash codes by using pairwise relationships between data points.

Recently, the deep learning hashing methods [23, 31, 37] have achieved breakthrough results in images retrieval datasets due to the powerful learning ability of deep networks. The first deep hash model (CNNH) [27] decomposes hash learning into approximate hash code learning and subsequent hash functions and image feature learning. DNNH [13] improved CNNH to learn feature representation [15] and hash code simultaneously via triple loss optimization model. DHN [34] preserving pairwise semantic similarity and controlling the quantization error by simultaneously optimizing the pairwise cross-entropy loss and quantization loss improved DNNH. DPSH [20] learns feature representation and hash codes simultaneously with pairwise data points via an end-to-end approach. Deep cauchy hashing (DCH) [2] utilize a pairwise cross-entropy loss based on Cauchy distribution to learn binary hash code. However, existing learning-based hashing methods are only for a given dataset of the same resolution. This is an ideal situation in the real world and many images are low-resolution. When reduced-resolution images is processed with a trained hash model, the search results will be poor. This paper proposes a high quality binary hash code generation method for LR images. We apply images super-resolution technology to images hashing, and use multi-task framework to deal with hashing problem.

## 2.2 Image Super Resolution Method

Image Super Resolution technology (ISR) is to estimate HR images from LR images. With the rapid development of deep learning, the ISR method based on Convolutional Neural Network (CNN) [6,35,36] shows excellent performance. Dong et al. [5] first proposed the image super-resolution framework SRCNN based on deep learning, and achieved excellent performance compared with the previous method. A faster network framework FSRCNN [6] was proposed to accelerate the training and testing of SRCNN. Ledig et al. [14] introduced the deep network ResNet [10] into the super-resolution task and used the perceptual loss and generative adversarial network (GAN) [9] for photo-realistic SR. Since the image super-resolution technology can make the LR images get more semantic information, we combine the images super-resolution task to deal with the hash retrieval problem of the LR images.



**Fig. 1.** The framework of the proposed deep super-resolution hashing network (DSRHN).

## 3 Our Method

### 3.1 Overview

In order to solve the LR images hashing problem in an end-to-end learning framework, this paper proposes an advanced end-to-end multi-task deep learning framework (DSRHN), as shown in Fig. 1. DSRHN mainly consists of two parts: a super-resolution network (SRNet) and a hash encoding network (HashNet). Our framework is an alternating learning process. First, we fix HashNet to train super-resolution networks via two loss functions: the last convolutional layer features of HashNet for  $x_i$  and  $x_i^{SR}$  produce perceptual loss, the output of SRNet and HR images produce pixel-wise mean square error (MSE) loss. Second, we fix SRNet to train hash network utilizes two losses: a hash semantic loss that

preserves the hash semantic similarity and a discriminant loss that maintains the robustness of the HashNet. For test process, we directly input the LR images into SRNet, then input the SR images of SRNets' output into HashNet, and finally represent the output of HashNet as a hash code through the sign function. The sign function is defined as:

$$\text{sign}(x) = \begin{cases} 1, & \text{if } x \geq 0 \\ -1. & \text{otherwise} \end{cases} \quad (1)$$

### 3.2 SRNet

The top half of Fig. 1 shows our super-resolution network SRNet. The main purpose of SRNet is to learn a mapping function of LR images to HR images. The input and output of SRNet are LR images  $\{x_i^{LR}, x_j^{LR}\}$  and SR images  $\{x_i^{SR}, x_j^{SR}\}$ . This process can be defined as:

$$X^{SR} = F_{SR}(X^{LR}) \quad (2)$$

where  $F_{SR}(\cdot)$  denotes the super-resolution function. The main part of SRNet is 16 residual blocks. For the residual blocks we use two convolutional layers with small  $3 \times 3$  kernels and 64 feature maps combined batch-normalization layers and ParametricReLU activation function. In order to improve the resolution of the input images, we adopt two trained sub-pixel convolution layers proposed by Shi et al. [24]. In particular, unlike traditional image super-resolution methods, we super-resolution LR images on hash semantics. We make content loss for the features of SR and HR images in the final convolution layer of HashNet. This method can ensure that the SR images have the similar binary hash codes to the HR images.

Considering the feature preservation of SR and HR images, the following loss functions are used to learn HR images:

$$L_{mse} = \|X^{HR} - X^{SR}\|_2^2 \quad (3)$$

where the  $X_{LR}$  denotes the input LR images, and  $X_{SR}$  is ground truth HR images.  $\|\cdot\|_2$  denotes the L2 norm. Considering the SR images and corresponding HR images should keep similarity on hash semantics, we adopt perceptual loss based on the feature of the last convolutional layer of the HashNet:

$$\begin{aligned} L_{per} &= \|F_{cov}(X^{HR}) - F_{cov}(X^{SR})\|_2^2 \\ &= \|F_{cov}(X^{HR}) - F_{cov}(F_{SR}(X^{LR}))\|_2^2 \end{aligned} \quad (4)$$

where  $F_{cov}(\cdot)$  denotes the HashNet convolutional layer. Overall, combining Eqs. (3) and (4), the loss of the SRNet can be written as:

$$L_{SR} = L_{per} + \lambda L_{mse} \quad (5)$$

where  $\lambda$  is the hyper-parameter.

### 3.3 HashNet

The bottom half of Fig. 1 is our hash encoding network denoted as HashNet. The purpose of HashNet is to learn nonlinear hash function  $f : x \mapsto h \in \{-1, 1\}^K$  from an input space  $R^D$  to Hamming space  $\{1, 1\}^K$  via deep neural networks. Alexnet [11] is adopted as the basic network of HashNet. HashNet consists of five convolution layers (c1-c5) and two fully connected layers (fc6,fc7), which are pre-trained on the ImageNet dataset. In order to obtain hash codes, we add a K-node hash layer called fch, each node of fch layer corresponds to a target hash code. Using fch layer, the previous representation is converted to k-dimensional representation. HashNet can encode each data point into a K-bit binary hash code with similar information S preserved. For HashNet, the input is SR images pairs, HR images pairs, and pairwise similarity relation  $\{x_i, x_i^{SR}, x_j, x_j^{SR}, S_{ij}\}$ .  $S = \{s_{ij}\}$  is defined as:

$$S_{ij} = \begin{cases} 0, & \text{if images } x_i \text{ and } x_j \text{ share same class label} \\ 1. & \text{otherwise} \end{cases} \quad (6)$$

After the HashNet is trained, the binary hash code  $B$  is calculated through the trained hashing network:

$$b_i = \text{sign}(F_{hash}(x_i|\theta)) \quad (7)$$

where  $b_i$  is hash code,  $F_{hash}(\cdot)$  denotes the hash function,  $\theta$  denotes the parameters of the HashNet,  $x_i$  represents the input images. In addition, the hash function we learned has the ability to distinguish SR from HR images. It is contrary to SRNet's expectation that SR and HR images remain hash semantically identical. This learning method enables SRNet and HashNet to learn better.

Usually calculate the distance between binary hash codes using Hamming distance, it can be calculated as:

$$\text{dist}_H = \frac{1}{2}(K - \langle b_i, b_j \rangle) \quad (8)$$

where K is the length of the hash code,  $\langle \cdot \rangle$  denotes the inner product between hash codes. From Eq. (8) we know that Hamming distance is related to inner product. The larger the inner product of the two hash codes, the smaller the Hamming distance and vice versa. Therefore, we can learn the discriminative hash code by replacing the similarity of semantic labels with the inner product between hash codes. Based on this fact, we use the following loss function as DPSH [20] does. Firstly, it is hoped that pairwise label can be fitted by similarity (inner product) between sample binary codes.

$$p(s_{ij}|b_i, b_j) = \begin{cases} \sigma(\langle b_i, b_j \rangle), & s_{ij} = 1 \\ 1 - \sigma(\langle b_i, b_j \rangle), & s_{ij} = 0 \end{cases} \quad (9)$$

where  $\sigma(x) = 1/(1 + e^{-x})$  is the sigmoid activation function,  $\langle b_i, b_j \rangle = \frac{1}{2}b_i^T b_j$ .  $p(\cdot)$  is the conditional probability of  $s_{ij}$  given the pair of corresponding hash

codes  $[b_i, b_j]$ . Based on the above, the following loss function is used to preserve the similarity of hash semantics:

$$\begin{aligned} L_{hs} &= -\log p(S|B) = -\sum_{s_{ij} \in S} \log p(s_{ij} | \langle b_i, b_j \rangle) \\ &= -\sum_{s_{ij} \in S} (s_{ij} \langle b_i, b_j \rangle - \log(1 + e^{\langle b_i, b_j \rangle})) \end{aligned} \quad (10)$$

Equation (10) is negative log likelihood loss function, which represents the Hamming distance of two similar images as small as possible and the Hamming distance of two different images as large as possible. However, the hash code  $b_i \in \{-1, +1\}$  is discrete. This is hard to optimize. To solve this problem, equality constraints can be optimized by moving them to regularization terms. So use the following loss function to optimize:

$$L_{hs} = -\sum_{s_{ij} \in S} (s_{ij} \langle v_i, v_j \rangle - \log(1 + e^{\langle v_i, v_j \rangle})) + \gamma \sum_1^n \|b_i - v_i\| \quad (11)$$

where  $v$  is the binary code after relaxation. The last part ( $\gamma \sum_1^n \|b_i - v_i\|$ ) of the equation is the regularization term,  $\gamma$  is the hyper-parameter.

In order to ensure that the hash codes learned by the hash function can distinguish SR and HR images and make the HashNet robust, we design a discriminative loss for optimizing the hashing network:

$$L_{dis} = \max(m - \|F_{hash}(x_i^{HR}) - F_{hash}(x_i^{SR})\|_2^2, 0) \quad (12)$$

where  $F_{hash}(\cdot)$  denotes hash network (HashNet), and  $m > 0$  is a margin threshold parameter. Overall, combining Eqs. (11) and (12), the loss of the hash network can be written as:

$$L_{hash} = L_{hs} + \alpha L_{dis} \quad (13)$$

where  $\alpha$  is the hyper-parameter.

## 4 Experiments

To validate the performance of our proposed method, we conducted extensive experiments on two widely used benchmark datasets (i.e. NUS-WIDE and MS-COCO) to verify the effectiveness of our approach. We start with introducing the datasets and then present our experimental results.

### 4.1 Datasets and Evaluation Metrics

In the experiments, we conduct hashing on two experiments widely-used datasets: NUS-WIDE [3] and MS-COCO [21].

- **NUS-WIDE** dataset consists of 269,648 web images associated with the tag. It is a multi-label dataset where each image can be annotated with multiple tags. Following DPSH [20], we selected only 195,834 images belonging to the 21 most common concepts. For NUS-WIDE datasets, if two images share at

least one public tag, they will be defined as similar. We randomly sampled 100 images per class (i.e., a total of 2,100 images) as a test set, with 500 images per class (i.e., a total of 10,500 images) as a training set. The rest of the image is considered a gallery during the testing phase.

- **MS-COCO** datasets we used consists 82,783 training images and 40,504 validation images, each of which is labeled by some of the 80 semantic concepts. Following DCH [2], we randomly extracted 5,000 images as query points, the rest as databases, and 10,000 images are randomly sampled from the database for training. For MS-COCO dataset, two images will be defined as a ground-truth neighbor (similar pair) if they share at least one common label.

**Evaluation Metrics:** We use mean average precision (mAP), precision, and recall to evaluate the performance of DSRHN compared to the aforementioned unsupervised hashing functions for low-resolution image retrieval. In particular, our low-resolution images are obtained from experimental datasets by interpolation, that is, the original dataset images are high-quality images and interpolated to obtain low-resolution images. We report the results of image retrieval with  $\text{map}@5000$ ,  $\text{mAP}@5000$  is mAP calculated over the top 5000 ranked images from the gallery set.

## 4.2 Ablation Study

We performe an ablation experiment to examine the true validity of our framework for low-resolution images hashing. We evaluated this experiment with DSRHN, HashNet-LR, HashNet-HR, and HashNet-SRGAN. The HashNet-LR approach is to directly use the low resolution images as input to our hash network without using a super resolution network; The HashNet-HR method inputs high resolution images into our hash network; The HashNet-SRGAN approach uses SRGAN work instead of our super-resolution network for a non-end-to-end LR images hashing.

As shown in Table 2. First of all, we can see from HashNet-SRGAN and HashNet-LR that super-resolution network does have effect on LR images hash retrieval. In particular, our DSRHN works best. Secondly, we can see that the result of HashNet-HR directly retrieving HR images and DSRHN retrieving LR images is very close, which shows that our framework is very good at learning the mapping of LR images to HR images on hash semantics. A comparison of DSRHN and HashNet-SRGAN shows that our end-to-end framework can better enhance the semantics of LR images on hash retrieval tasks.

## 4.3 Comparisons

For image retrieval, we compare our method with the previous deep supervised hash functions including deep pairwise-supervised hashing (DPSH) [20], deep hashing network (DHN) [34], and deep cauchy hashing (DCH) [2].



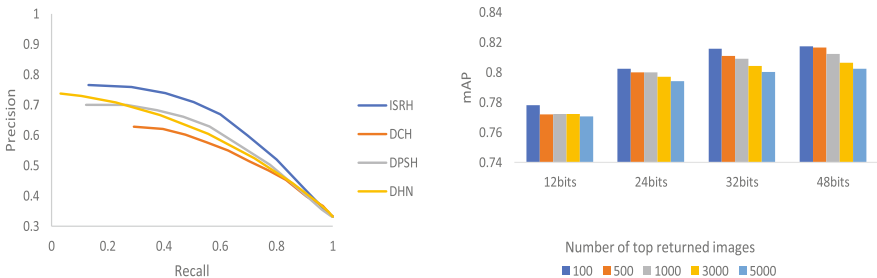
**Table 1.** LR images retrieval results (mAP@5000) of DSRHN and Its Variants, HashNet-LR, HashNet-HR, HashNet-SRGAN on two Datasets.

Method	NUS-WIDE				MS-COCO			
	12bits	24bits	32bits	48bits	12bits	24bits	32bits	48bits
HashNet-LR	71.32	72.29	72.97	73.01	55.73	56.61	57.86	56.61
HashNet-SRGAN	71.31	72.84	72.88	73.21	57.67	61.27	61.10	61.27
<b>DSRHN</b>	<b>77.07</b>	<b>79.40</b>	<b>80.02</b>	<b>80.23</b>	<b>63.94</b>	<b>68.34</b>	<b>69.21</b>	<b>69.63</b>
HashNet-HR	77.42	79.87	80.56	80.78	64.43	69.40	69.92	70.38

**Table 2.** LR images retrieval results (mAP@5000) of DSRHN on NUS-WIDE and MS-COCO datasets, when the number of hash bits are 12, 24, 32 and 48.

Method	NUS-WIDE				MS-COCO			
	12bits	24bits	32bits	48bits	12bits	24bits	32bits	48bits
DCH	63.92	66.22	66.98	67.21	47.82	48.21	46.18	48.95
DHN	67.91	69.75	69.39	71.30	49.89	51.66	52.44	53.47
DPSH	69.54	72.10	71.61	72.56	55.09	55.73	57.86	56.61
<b>DSRHN(Ours)</b>	<b>77.07</b>	<b>79.40</b>	<b>80.02</b>	<b>80.23</b>	<b>63.94</b>	<b>68.34</b>	<b>69.21</b>	<b>69.63</b>

Table 1 shows the DSRHN and other optional models' mAP@5000 results on different hash bits. The results show that our model is always superior to other models with significant effects in terms of different bit numbers, datasets and metrics. The main reason is our framework directly enhances the hash semantics of low resolution images. Figure 2 show the results of precision-recall curves with 12 bits, and mAP@5000 with 48 bits w.r.t. different numbers of top returned images. It can be seen that our method not only works best, but also has excellent retrieval stability.



(a) Precision-Recall curve @ 12 bits.

(b) mAP curve w.r.t. top-n @ 48 bits.

**Fig. 2.** The results of DSRHN and comparison methods on the NUS-WIDE dataset.

## 4.4 Implementation Details

The proposed network consists of two parts: SRNet and HashNet. The SRNet is trained specially for  $4\times$  scale factor super-resolution. We randomly crop the  $224\times 224$  patch in each images as the ground truth, and downsample it to  $56\times 56$  as the input LR patch for training. We use 16 residual blocks with two convolutional layers with small  $3\times 3$  kernels and 64 feature maps combined batch-normalization layers and ParametricReLU activation function. SRNet is optimized by Adam with the learning rate 0.001. For the HashNet, we use AlexNet network directly. And it is optimized by SGD with the learning rate 0.01. The  $\lambda$  of  $L_{SR}$  is set as 0.1, and the  $\alpha$  of  $L_{hash}$  is set as 0.01. The training process is stopped when the training reaches 150 epochs and we select the best model for comparison. Experiments are performed on a NVIDIA TitanXp GPU.

## 5 Conclusion

This paper proposes a novel images hash retrieval framework (DSRHN) based on deep super-resolution. The framework consists of two main components: a super-resolution network and a hash encoding network. Super-resolution networks trained on large-scale image retrieval datasets can recover low-resolution images to high-resolution images while providing more abundant semantic information. The hash encoding network can represent the recovered images as a binary code. We use binary code for image retrieval experiments. Experimental results show that the proposed DSRHN algorithm has achieved the state-of-the-art performance.

**Acknowledgements.** This work was supported by National Key R&D Program of China (2018YFB0803700) and National Natural Science Foundation of China (61602517).

## References

1. Cao, Y., et al.: Binary hashing for approximate nearest neighbor search on big data: a survey. *IEEE Access* **6**, 2039–2054 (2018)
2. Cao, Y., Long, M., Liu, B., Wang, J.: Deep cauchy hashing for hamming space retrieval. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1229–1237 (2018)
3. 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*, p. 48. ACM (2009)
4. Datar, M., Immorlica, N., Indyk, P., Mirrokni, V.S.: Locality-sensitive hashing scheme based on p-stable distributions. In: *Proceedings of the Twentieth Annual Symposium on Computational Geometry*, pp. 253–262. ACM (2004)
5. Dong, C., Loy, C.C., He, K., Tang, X.: Learning a deep convolutional network for image super-resolution. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) *ECCV 2014*. LNCS, vol. 8692, pp. 184–199. Springer, Cham (2014). [https://doi.org/10.1007/978-3-319-10593-2\\_13](https://doi.org/10.1007/978-3-319-10593-2_13)

6. Dong, C., Loy, C.C., Tang, X.: Accelerating the super-resolution convolutional neural network. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9906, pp. 391–407. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-46475-6\\_25](https://doi.org/10.1007/978-3-319-46475-6_25)
7. Gionis, A., Indyk, P., Motwani, R., et al.: Similarity search in high dimensions via hashing. In: VLDB, vol. 99, pp. 518–529 (1999)
8. Gong, Y., Lazebnik, S., Gordo, A., Perronnin, F.: Iterative quantization: a procrustean approach to learning binary codes for large-scale image retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(12), 2916–2929 (2013)
9. Goodfellow, I., et al.: Generative adversarial nets. In: *Advances in Neural Information Processing Systems*, pp. 2672–2680 (2014)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778 (2016)
11. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: *Advances in Neural Information Processing Systems*, pp. 1097–1105 (2012)
12. Kulis, B., Grauman, K.: Kernelized locality-sensitive hashing. *IEEE Trans. Pattern Anal. Mach. Intell.* **34**(6), 1092–1104 (2012)
13. Lai, H., Pan, Y., Liu, Y., Yan, S.: Simultaneous feature learning and hash coding with deep neural networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3270–3278 (2015)
14. Ledig, C., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4681–4690 (2017)
15. Li, C., Liu, Q., Dong, W., Zhu, X., Liu, J., Hanqing, L.: Human age estimation based on locality and ordinal information. *IEEE Trans. Cybern.* **45**(11), 2522–2534 (2014)
16. Li, C., Liu, Q., Liu, J., Lu, H.: Ordinal distance metric learning for image ranking. *IEEE Trans. Neural Netw. Learn. Syst.* **26**(7), 1551–1559 (2014)
17. Li, C., Wang, X., Dong, W., Yan, J., Liu, Q., Zha, H.: Joint active learning with feature selection via cur matrix decomposition. *IEEE Trans. Pattern Anal. Mach. Intell.* **41**(6), 1382–1396 (2018)
18. Li, C., Wei, F., Dong, W., Wang, X., Liu, Q., Zhang, X.: Dynamic structure embedded online multiple-output regression for streaming data. *IEEE Trans. Pattern Anal. Mach. Intell.* **41**(2), 323–336 (2018)
19. Li, C., Yan, J., Wei, F., Dong, W., Liu, Q., Zha, H.: Self-paced multi-task learning. In: *Thirty-First AAAI Conference on Artificial Intelligence* (2017)
20. Li, W.-J., Wang, S., Kang, W.-C.: Feature learning based deep supervised hashing with pairwise labels. *arXiv preprint arXiv:1511.03855* (2015)
21. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). [https://doi.org/10.1007/978-3-319-10602-1\\_48](https://doi.org/10.1007/978-3-319-10602-1_48)
22. Liu, W., Wang, J., Ji, R., Jiang, Y.-G., Chang, S.-F.: Supervised hashing with kernels. In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2074–2081. IEEE (2012)
23. Liu, Y., Zhu, X., Zhao, X., Cao, Y.: Adversarial learning for constrained image splicing detection and localization based on atrous convolution. *IEEE Trans. Inf. Forensics Secur.* **14**, 2551–2566 (2019)

24. Shi, W., et al.: Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1874–1883 (2016)
25. Wang, J., Kumar, S., Chang, S.-F.: Semi-supervised hashing for large-scale search. *IEEE Trans. Pattern Anal. Mach. Intell.* **34**(12), 2393–2406 (2012)
26. Weiss, Y., Torralba, A., Fergus, R.: Spectral hashing. In: Advances in Neural Information Processing Systems, pp. 1753–1760 (2009)
27. 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 (2014)
28. Zhang, X.-Y.: Simultaneous optimization for robust correlation estimation in partially observed social network. *Neurocomputing* **205**, 455–462 (2016)
29. Zhang, X.-Y., Shi, H., Li, C., Zheng, K., Zhu, X., Duan, L.: Learning transferable self-attentive representations for action recognition in untrimmed videos with weak supervision. arXiv preprint [arXiv:1902.07370](https://arxiv.org/abs/1902.07370) (2019)
30. Zhang, X.-Y., Shi, H., Zhu, X., Li, P.: Active semi-supervised learning based on self-expressive correlation with generative adversarial networks. *Neurocomputing* **345**, 103–113 (2019)
31. Zhang, X.-Y., Wang, S., Yun, X.: Bidirectional active learning: a two-way exploration into unlabeled and labeled data set. *IEEE Trans. Neural Netw. Learn. Syst.* **26**(12), 3034–3044 (2015)
32. Zhang, X.-Y., Wang, S., Zhu, X., Yun, X., Wu, G., Wang, Y.: Update vs. upgrade: modeling with indeterminate multi-class active learning. *Neurocomputing* **162**, 163–170 (2015)
33. Zhang, X.: Interactive patent classification based on multi-classifier fusion and active learning. *Neurocomputing* **127**, 200–205 (2014)
34. Zhu, H., Long, M., Wang, J., Cao, Y.: Deep hashing network for efficient similarity retrieval. In: Thirtieth AAAI Conference on Artificial Intelligence (2016)
35. Zhu, X., Li, Z., Zhang, X., Li, C., Liu, Y., Xue, Z.: Residual invertible spatio-temporal network for video super-resolution. In: Thirtieth AAAI Conference on Artificial Intelligence (2019)
36. Zhu, X., Li, Z., Zhang, X., Li, H., Xue, Z., Wang, L.: Generative adversarial image super-resolution through deep dense skip connections. In: *Computer Graphics Forum*, vol. 37, pp. 289–300. Wiley Online Library (2018)
37. Zhu, X., Liu, J., Wang, J., Li, C., Hanqing, L.: Sparse representation for robust abnormality detection in crowded scenes. *Pattern Recogn.* **47**(5), 1791–1799 (2014)
38. Zhu, X., Zhang, X., Zhang, X.-Y., Xue, Z., Wang, L.: A novel framework for semantic segmentation with generative adversarial network. *J. Vis. Commun. Image Represent.* **58**, 532–543 (2019)