基于非对称监督深度离散哈希的图像检索

顾广华* 霍文华 苏明月 付 灏
(燕山大学信息科学与工程学院 秦皇岛 066000)
(河北省信息传输与信号处理重点实验室 秦皇岛 066000)

摘 要：哈希广泛应用于图像检索任务。针对现有深度监督哈希方法的局限性，该文提出了一种新的非对称监督深度离散哈希(ASDDH)方法来保持不同类别之间的语义结构，同时生成二进制码。首先利用深度网络提取图像特征，根据图像的语义标签来揭示每对图像之间的相似性。为了增强二进制码之间的相似性，并保证多标签语义保持，该文设计了一种非对称哈希方法，并利用多标签二进制码映射，使哈希码具有多标签语义信息。此外，引入二进制码的位平衡性对每个位进行平衡，鼓励所有训练样本中的–1和+1的数目近似。在两个常用数据集上的实验结果表明，该方法在图像检索方面的性能优于其他方法。

关键词：图像检索；监督哈希；语义保持；深度学习
中图分类号：TN911.73; TP391.4 **文献标识码**：A **文章编号**：1009-5896(2021)12-3530-08
DOI: 10.11999/JEIT200988

Asymmetric Supervised Deep Discrete Hashing Based Image Retrieval

GU Guanghua HUO Wenhua SU Mingyue FU Hao
(*School of Information Science and Engineering, Yanshan Engineering University, Qinhuangdao* 066000, *China*)
(*Hebei Key Laboratory of Information Transmission and Signal Processing, Qinhuangdao* 066000, *China*)

Abstract: Hashing is widely used for image retrieval tasks. In view of the limitations of existing deep supervised hashing methods, a new Asymmetric Supervised Deep Discrete Hashing (ASDDH) method is proposed to maintain the semantic structure between different categories and generate binary codes. Firstly, a deep network is used to extract image features and reveal the similarity between each pair of images according to their semantic labels. To enhance the similarity between binary codes and ensure the retention of multi-label semantics, this paper designs an asymmetric hashing method that utilizes a multi-label binary code mapping to make the hash codes have multi-label semantic information. In addition, the bit balance of the binary code is introduced to balance each bit, which encourages the number of -1 and +1 to be approximately similar among all training samples. Experimental results on two benchmark datasets show that the proposed method is superior to other methods in image retrieval.

Key words: Image retrieval; Supervised hashing; Semantic preservation; Deep learning

1 引言

随着实际应用中数据的爆炸式增长，最近邻搜索在信息检索、计算机视觉等领域有着广泛的应用。然而，在大数据应用中，对于给定的查询，最近邻搜索通常是很耗时的。因此，近年来，近似最近邻(Artificial Neural Network，ANN)搜索[1]变得越来越流行。在现有的ANN技术中，哈希以其快速的查询速度和较低的内存成本成为最受欢迎和有效的技术之一。哈希方法[2,3]的目标是将多媒体数据从原来的高维空间转换为紧凑的汉明空间，同时保持数据的相似性。这些二进制哈希码不仅可以显著降低存储成本，在信息搜索中实现恒定或次线性的时间复杂度，而且可以保持原有空间中存在的语义结构。

现有的哈希方法大致可分为两类：独立于数据的哈希方法和依赖于数据的哈希方法。局部敏感哈希(Locality Sensitive Hashing, LSH)[4]及其扩展作

收稿日期：2020-11-23；改回日期：2021-10-25；网络出版：2021-11-09
*通信作者： 顾广华 guguanghua@ysu.edu.cn
基金项目：国家自然科学基金(62072394)，河北省自然科学基金(F2021203019)
Foundation Items: The National Natural Science Foundation of China (62072394), The Natural Science Foundation of Hebei Province (F2021203019)

为最典型的独立于数据的哈希方法，利用随机投影得到哈希函数。但是，它们需要较长的二进制代码才能达到很高的精度。由于数据独立哈希方法的局限性，近年来的哈希方法尝试利用各种机器学习技术，在给定数据集的基础上学习更有效的哈希函数。

依赖于数据的哈希方法从可用的训练数据中学习二进制代码，也就是学习哈希。现有的数据依赖哈希方法根据是否使用监督信息进行学习，可以进一步分为无监督哈希方法和监督哈希方法。代表性的无监督哈希方法包括迭代量化(IteraTive Quantization, ITQ)[5]，离散图哈希(Discrete Graph Hashing, DGH)[6]、潜在语义最小哈希(Latent Semantic Minimal Hashing, LSMH)[7]和随机生成哈希(Stochastic Generative Hashing, SGH)[8]。无监督哈希只是试图利用数据结构学习紧凑的二进制代码来提高性能，而监督哈希则是利用监督信息来学习哈希函数。典型的监督哈希方法包括核监督哈希(Supervised Hashing with Kernels, KSH)[9]，监督离散哈希(Supervised Discrete Hashing, SDH)[10]和非对称离散图哈希(Asymmetric Discrete Graph Hashing, ADGH)[11]。近年来，基于深度学习的哈希方法[12]被提出来同时学习图像表示和哈希编码，表现出优于传统哈希方法的性能。典型的深度监督哈希方法包括深度成对监督哈希(Deep Supervised Hashing with Pairwise Labels, DPSH)[13]，深度监督离散哈希(Deep Supervised Discrete Hashing, DSDH)[14]，和深度离散监督哈希(Deep Discrete Supervised Hashing, DDSH)[15]。通过将特性学习和哈希码学习集成到相同的端到端体系结构中，深度监督哈希[16,17]可以显著优于非深度监督哈希。然而，现有的深度监督哈希方法主要利用成对监督进行哈希学习，语义信息没有得到充分利用，这些信息有助于提高哈希码的语义识别能力。更困难的是，对于大多数数据集，每个项都由多标签信息进行注释。因此，不仅需要保证多个不同的项对之间具有较高的相关性，还需要在一个框架中保持多标签语义，以生成高质量的哈希码。

为了解决上述问题，本文提出了一种非对称监督深度离散哈希(Asymmetric Supervised Deep Discrete Hashing, ASDDH)方法。具体来说，为了生成能够完全保留所有项的多标签语义的哈希码，提出了一种非对称哈希方法，利用多标签二进制码映射，使哈希码具有多标签语义信息。此外，本文还引入了二进制代码的位平衡性，进一步提高哈希函数的质量。在优化过程中，为了减小量化误差，利用离散循环坐标下降法对目标函数进行优化，以保持哈希码的离散性。

2　非对称监督深度离散哈希

2.1　符号和问题定义

给定一组N张图像的训练集：$\boldsymbol{X}=\{x_i\}_{i=1}^{N}\in\boldsymbol{R}^{d\times N}$，哈希的目的是学习一组$K$位二进制码$\boldsymbol{B}\in\{-1,1\}^{K\times N}$，其中第$i$列$b_i\in\{-1,1\}^{K}$代表第$i$个样本$x_i$的$K$位二进制编码。一般来说，可以把二进制代码写成$b_i=h(x_i)=[h_1(x_i),h_2(x_i),\cdots,h_c(x_i)]$，其中$h(\cdot)$表示要学习的哈希函数。

对于监督哈希方法，监督信息可以是单标签、成对的标签或三重标签。本文只关注基于成对标签的监督哈希，这是一个常见的应用场景。在监督哈希中，标签信息表示为$\boldsymbol{Y}=\{y_i\}_{i=1}^{N}\in\boldsymbol{R}^{c\times N}$，其中$y_i\in\{0,1\}^{c}$对应于样本$x_i$，$c$是数据集类别的数量。注意，在多标签数据集中，一个样本可能属于多个类别。此外，成对的图像与相似性标签s_{ij}相关联。这里，使用$\boldsymbol{S}=\{s_{ij}\}$，$s_{ij}\in\{0,1\}$来表示两个图像的相似性，$s_{ij}=1$表示$x_i$和$x_j$相似，$s_{ij}=0$表示$x_i$和$x_j$不相似。监督哈希的目的是学习一个哈希函数，将数据点从原始空间映射到二进制代码空间，在二进制代码空间中保留$\boldsymbol{S}$中的语义相似性。对于两个二进制代码b_i和b_j，他们之间的汉明距离定义为：$\mathrm{dist}_{\mathrm{H}}(b_i,b_j)=\frac{1}{2}(K-\langle b_i,b_j\rangle)$。因此，可以利用内积测量哈希码的相似性。为了保持数据点之间的相似性，如果数据点x_i和x_j相似(即$s_{ij}=1$)，二进制编码b_i和b_j之间的汉明距离应该相对较小。反之，如果数据点x_i和x_j不相似，则二进制码b_i和b_j之间的汉明距离应该较大，即$s_{ij}=0$。

2.2　模型表述

该方法的主要框架如图1所示，其中包含两个重要的组件：特征学习部分和损失函数部分。由于深度神经网络在数据表示方面的强大功能，卷积神经网络被广泛应用于图像检索任务进行特征学习。为了与几种基线进行公平比较，本文采用AlexNet网络作为特征学习部分的主干网络进行学习，并且在3.4节本文讨论了几种不同的卷积神经网络。AlexNet网络包含5个卷积层和3个全连接层，前7层的激活函数均为ReLU。为了得到最终的二进制代码，AlexNet模型的最后一层被1个完全连通的哈希层（激活函数为tanh）所取代，该层可以将前7层的输出投影到$\boldsymbol{R}^{K}$空间中。本文将顶层输出定义为：$\boldsymbol{H}=\{h_i\}_{i=1}^{N}\in\boldsymbol{R}^{K\times N}$，将二进制代码定义为：$b_i=\mathrm{sign}(h_i)$。损失函数部分的目的是学习最优二进制代码以保持成对语义相似性。ASDDH模型将这两个组件集成到同一个端到端框架中。在训练过程中，每个部分都可以给另一个部分反馈。

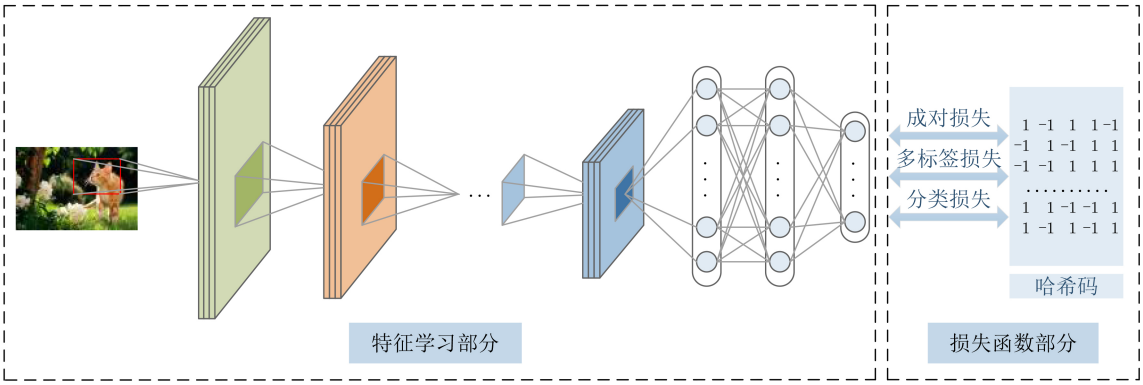


图 1 ASDDH模型体系结构

2.3 损失函数

成对监督哈希的主要目的是使相似(不相似)对的二进制码之间的汉明距离小(大)。利用上述原理，文献[18]提出了一种学习紧凑二进制代码的铰链式损失函数，Liu等人[9]采用了哈希码间监督信息和内积之间的l_2范数损失。在这项工作中，内积被用来作为汉明距离的一个很好的替代品来量化成对的相似性。给定所有图像$\boldsymbol{X}$的二进制码$\boldsymbol{B}=\{\boldsymbol{b}_i\}_{i=1}^{N}\in\{-1,1\}^{K\times N}$，定义成对相似性损失函数为

$$L_1=\sum_{s_{ij}\in S}\left(\lg\left(1+e^{\alpha\Theta_{ij}}\right)-\alpha s_{ij}\Theta_{ij}\right) \tag{1}$$

其中$\Theta_{ij}=\frac{1}{2}\langle\boldsymbol{b}_i,\boldsymbol{b}_j\rangle=\frac{1}{2}\boldsymbol{b}_i^{\mathrm{T}}\boldsymbol{b}_j$，式(2)为成对相似度的负对数似然。成对逻辑函数定义为

$$p(s_{ij}|\boldsymbol{b}_i,\boldsymbol{b}_j)=\begin{cases}\sigma(\Theta_{ij}),s_{ij}=1\\1-\sigma(\Theta_{ij}),s_{ij}=0\end{cases} \tag{2}$$

其中$\sigma(x)=\dfrac{1}{1+\mathrm{e}^{-\alpha x}}$是用超参数$\alpha$控制它的带宽的自适应sigmoid函数。$\alpha$较大的sigmoid函数将有更大的饱和区域梯度为零。为了执行更有效的反向传播，需要$\alpha\le 1$。

为了提高所学哈希码的准确性，本文还使学习的二进制码具有以下特性：(1)语义分类最优。直接利用标签信息，使所学习的二进制码对于联合学习的线性分类器是最优的。(2)多标签语义保存。引入一种非对称哈希方法，该方法利用多标签二进制码映射，使哈希码保留多标签语义信息。(3)位平衡。使学习的哈希码的每个位有50%的概率为1或–1。

本文使用一个简单的线性分类器来建模学习的二进制代码和标签信息之间的关系：

$$\boldsymbol{Y}=\boldsymbol{W}^{\mathrm{T}}\boldsymbol{B} \tag{3}$$

其中$\boldsymbol{W}\in\boldsymbol{R}^{K\times c}$是分类器权重。则分类损失可以表示为

$$\begin{aligned}L_2&=\sum_{i=1}^{N}L(\boldsymbol{y}_i,\boldsymbol{W}^{\mathrm{T}}\boldsymbol{b}_i)+\lambda\|\boldsymbol{W}\|_{\mathrm{F}}^2\\&=\sum_{i=1}^{N}\left\|\boldsymbol{y}_i-\boldsymbol{W}^{\mathrm{T}}\boldsymbol{b}_i\right\|_F^2+\lambda\|\boldsymbol{W}\|_{\mathrm{F}}^2\end{aligned} \tag{4}$$

这里$L(\cdot)$是损失函数，本文选择的是线性分类器的l_2损失。$\|\cdot\|_{\mathrm{F}}$是矩阵的Frobenius范数，λ是正则化参数虽然式(1)实现了成对监督信息的保存，式(4)实现了最优的线性分类，但忽略了多标签语义的保存。

为了进一步增强所获得的二进制代码之间的相似性，并保证多标签语义保存，本文提出了一个多标签二进制代码映射，并以非对称的方式对其进行优化，以获得非对称哈希的潜在能力。该映射表示语义标签的二进制表示形式，记为$\boldsymbol{Q}\in\{-1,1\}^{K\times c}$。生成的哈希代码应该与它所附加的语义标签相似。因此，非对称离散损失可以表示为

$$L_3=\sum_{i=1}^{N}\left(\boldsymbol{b}_i{}^{\mathrm{T}}\boldsymbol{Q}-K\widehat{\boldsymbol{y}}_i{}^{\mathrm{T}}\right)^2 \tag{5}$$

其中$\widehat{\boldsymbol{y}}_i=\boldsymbol{y}_i\times 2-1\in\{-1,1\}^c$。

为了使哈希码的每一位在所有训练集上保持平衡。本文增加了位平衡损失项来最大化每一位所提供的数据点信息。更具体地，在所有训练点上，对每个位进行了平衡，鼓励所有训练样本中的–1和+1的数目近似。此时编码达到平衡，信息量最大，哈希编码最优。该损失项表示为

$$L_4=\|\boldsymbol{B}\boldsymbol{I}_{N\times 1}\|_{\mathrm{F}}^2 \tag{6}$$

其中$\boldsymbol{I}_{N\times 1}$表示所有元素都等于1的矩阵。综合考虑式(1)、式(4)、式(5)和式(6)，得到整体目标函数为

$$\begin{aligned}\min_{B,W,Q} L &= L_1+\beta L_2+\gamma L_3+\tau L_4\\ &=\sum_{s_{ij}\in S}\left(\lg\left(1+e^{\alpha\Theta_{ij}}\right)-\alpha s_{ij}\Theta_{ij}\right)\\ &\quad+\beta\sum_{i=1}^{N}\left\|\boldsymbol{y}_i-\boldsymbol{W}^{\mathrm{T}}\boldsymbol{b}_i\right\|_{\mathrm{F}}^2+\mu\|\boldsymbol{W}\|_{\mathrm{F}}^2\\ &\quad+\gamma\sum_{i=1}^{N}\left(\boldsymbol{b}_i^{\mathrm{T}}\boldsymbol{Q}-K\widehat{y}_i^{\mathrm{T}}\right)^2+\tau\left\|\boldsymbol{B}\boldsymbol{I}_{N\times 1}\right\|_{\mathrm{F}}^2\\ \text{s.t. }&\boldsymbol{B}=\{\boldsymbol{b}_i\}_{i=1}^{N}\in\{-1,1\}^{K\times N},\boldsymbol{Y}=\{\boldsymbol{y}_i\}_{i=1}^{N}\in\boldsymbol{R}^{c\times N}\end{aligned}\tag{7}$$

其中，β，γ，τ分别表示各个项之间的权衡参数，且$\mu=\beta\lambda$。

由于具有式(7)中的二进制约束离散优化求解非常具有挑战性，现有的方法大多采用对二进制约束进行连续松弛的方法。在测试阶段，对连续输出应用阈值函数得到哈希码。然而，这种连续松弛会通过对哈希码的连续嵌入进行二值化而产生不可控制的量化误差。为了克服这种局限性，本文采用了一种新的离散求解策略，将$\operatorname{sign}(h_i)$设置为接近它对应的哈希码b_i。然而，由于$\operatorname{sign}(h_i)$中的h_i梯度处处为零，很难进行反向传播。本文将$\tanh(\cdot)$应用于$\operatorname{sign}(\cdot)$函数的软逼近。为了控制量化误差，缩小期望二进制码与松弛之间的距离，在h_i上加了额外的惩罚项来逼近期望的离散二进制码b_i。将式(7)重新表述为

$$\begin{aligned}\min_{B,W,Q} L &=\sum_{s_{ij}\in S}\left(\lg\left(1+\mathrm{e}^{\alpha\Phi_{ij}}\right)-\alpha s_{ij}\Phi_{ij}\right)\\ &\quad+\beta\sum_{i=1}^{N}\left\|\boldsymbol{y}_i-\boldsymbol{W}^{\mathrm{T}}\boldsymbol{b}_i\right\|_{F}^2+\mu\|\boldsymbol{W}\|_{\mathrm{F}}^2\\ &\quad+\gamma\sum_{i=1}^{N}\left(\tanh^{\mathrm{T}}(h_i)\boldsymbol{Q}-K\widehat{\boldsymbol{y}}_i^{\mathrm{T}}\right)^2\\ &\quad+\tau\left\|\tanh(\boldsymbol{H})\,I_{N\times 1}\right\|_{\mathrm{F}}^2\\ &\quad+\eta\sum_{i=1}^{N}\left\|\boldsymbol{b}_i-\tanh(h_i)\right\|_{\mathrm{F}}^2\\ \text{s.t. }&\boldsymbol{b}_i\in\{-1,1\}^{K},\boldsymbol{H}=\{h_i\}_{i=1}^{N}\in\boldsymbol{R}^{K\times N},\\ &\boldsymbol{Q}\in\{-1,1\}^{K\times c}\end{aligned}\tag{8}$$

其中$\Phi_{ij}=\dfrac{1}{2}\tanh^{\mathrm{T}}(h_i)\tanh(h_j)$，$h_i\in R^{K\times 1}(i=1,2,\cdots,N)$。$\beta$，$\mu$，$\eta$，$\tau$，$\gamma$是在各种项之间进行平衡的超参数。

2.4 优化算法

由目标函数式(8)可知，该损失函数的优化问题是非凸非光滑的，很难直接得到最优解。为了找到一个可行的解，本文使用的是交替优化的方法，这种方法在哈希文献[13,14]中得到了广泛的应用：固定其他变量更新一个变量。更具体地说，本文依次对深度神经网络参数、分类器权重$\boldsymbol{W}$、二进制码矩阵$\boldsymbol{B}$和多标签二进制映射$\boldsymbol{Q}$的参数进行迭代更新，步骤如下:

(1)更新$\boldsymbol{H}$，固定$\boldsymbol{W}$, $\boldsymbol{B}$和$\boldsymbol{Q}$。当固定$\boldsymbol{B}$和$\boldsymbol{Q}$时，利用随机梯度下降法(SGD)学习深度神经网络的参数。特别地，在每次迭代中，从整个训练数据集中抽取一小批图像样本，并使用反向传播算法对整个网络进行更新。这里表示$\boldsymbol{U}=\tanh(\boldsymbol{H})$。损失函数的导数为

$$\begin{aligned}\frac{\partial L}{\partial \boldsymbol{U}}=&\frac{1}{2}\sum_{s_{ij}\in S}\left(\sigma\left(\Phi_{ij}\right)-\alpha s_{ij}\right)\boldsymbol{U}+2\gamma\left(\boldsymbol{U}^{\mathrm{T}}\boldsymbol{Q}-K\widehat{\boldsymbol{Y}}^{\mathrm{T}}\right)\\ &+2\tau\boldsymbol{U}\boldsymbol{I}_{N\times 1}-2\eta\left(\boldsymbol{B}-\boldsymbol{U}\right)\end{aligned}\tag{9}$$

然后利用链式法则来计算$\dfrac{\partial L}{\partial \boldsymbol{H}}$，更新神经网络的参数。

(2)更新$\boldsymbol{W}$，固定$\boldsymbol{H}$, $\boldsymbol{B}$和$\boldsymbol{Q}$。将目标函数式(8)以矩阵形式转化为

$$\min_{W} L=\beta\left\|\boldsymbol{Y}-\boldsymbol{W}^{\mathrm{T}}\boldsymbol{B}\right\|_{F}^2+\mu\|\boldsymbol{W}\|_{F}^2\tag{10}$$

式(10)为最小二乘问题，其解为闭形式

$$\boldsymbol{W}=\left(\boldsymbol{B}\boldsymbol{B}^{\mathrm{T}}+\frac{\mu}{\beta}\boldsymbol{I}\right)^{-1}\boldsymbol{B}Y^{\mathrm{T}}\tag{11}$$

(3)更新$\boldsymbol{B}$，固定$\boldsymbol{H}$, $\boldsymbol{W}$和$\boldsymbol{Q}$。将问题式(8)重写为矩阵形式：

$$\begin{aligned}\min_{B} L&=\beta\left\|\boldsymbol{Y}-\boldsymbol{W}^{\mathrm{T}}\boldsymbol{B}\right\|_{F}^2+\eta\|\boldsymbol{B}-\boldsymbol{U}\|_{\mathrm{F}}^2\\ \text{s.t. }&\boldsymbol{B}\in\{-1,1\}^{K\times N}\end{aligned}\tag{12}$$

这里$\boldsymbol{U}=\tanh(\boldsymbol{H})$。式(12)的优化问题是离散优化问题。本文采用离散循环坐标下降法逐行迭代求解$\boldsymbol{B}$。式(12)可以改写为

$$\begin{aligned}\min_{B}&\|\boldsymbol{Y}\|^2-2\mathrm{tr}\left(\boldsymbol{Y}^{\mathrm{T}}\boldsymbol{W}^{\mathrm{T}}\boldsymbol{B}\right)+\left\|\boldsymbol{W}^{\mathrm{T}}\boldsymbol{B}\right\|_{\mathrm{F}}^2\\ &+\frac{\eta}{\beta}\left(\|\boldsymbol{B}\|_{\mathrm{F}}^2-2\mathrm{tr}\left(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{U}\right)+\|\boldsymbol{U}\|_{\mathrm{F}}^2\right)\\ \text{s.t. }&\boldsymbol{B}\in\{-1,1\}^{K\times N}\end{aligned}\tag{13}$$

式(13)可简化为

$$\begin{aligned}\min_{B}&\left\|\boldsymbol{W}^{\mathrm{T}}\boldsymbol{B}\right\|_{\mathrm{F}}^2-2tr\left(\boldsymbol{B}\boldsymbol{S}^{\mathrm{T}}\right)+\mathrm{const}\\ \text{s.t. }&\boldsymbol{B}\in\{-1,1\}^{K\times N}\end{aligned}\tag{14}$$

其中$\boldsymbol{S}=\boldsymbol{W}\boldsymbol{Y}+\dfrac{\eta}{\beta}\boldsymbol{U}$且$\mathrm{tr}(\cdot)$是迹范数，“const”是一个与$\boldsymbol{B}$无关的常数。

然后，$\boldsymbol{B}$可以逐位更新。即，固定$\boldsymbol{B}$的其余的行来更新$\boldsymbol{B}$中的一行。让$\boldsymbol{b}^{\mathrm{T}}$是$\boldsymbol{B}$的第$k$行，$k=1,2,\cdots,K$。$\widetilde{\boldsymbol{B}}$是$\boldsymbol{B}$不含$\boldsymbol{b}$的矩阵。那么$\boldsymbol{b}$是所有

N个样本的一个位。同时，令$\boldsymbol{s}^{\mathrm{T}}$是$\boldsymbol{S}$的第$k$行，$\widetilde{\boldsymbol{S}}$是$\boldsymbol{S}$不含$\boldsymbol{s}$的矩阵，$\boldsymbol{w}^{\mathrm{T}}$是$\boldsymbol{W}$的第$k$行，$\widetilde{\boldsymbol{W}}$是$\boldsymbol{W}$不含$w$的矩阵。然后有

$$\begin{aligned}&\min_{b}\operatorname{tr}\left(\boldsymbol{b}\left(2\boldsymbol{w}^{\mathrm{T}}\widetilde{\boldsymbol{W}}^{\mathrm{T}}\widetilde{\boldsymbol{B}}-2\boldsymbol{s}^{\mathrm{T}}\right)\right)+\text{const}\\&\text{s.t. }\boldsymbol{b}\in\{-1,1\}^{N}\end{aligned}\tag{15}$$

显然，这个问题有最优解

$$\boldsymbol{b}=\operatorname{sign}(\boldsymbol{s}-\widetilde{\boldsymbol{B}}^{\mathrm{T}}\widetilde{\boldsymbol{W}}\boldsymbol{w})\tag{16}$$

(4)更新$\boldsymbol{Q}$，固定$\boldsymbol{H}$, $\boldsymbol{W}$和$\boldsymbol{B}$。优化$\boldsymbol{Q}$的方法和步骤和优化$\boldsymbol{B}$类似，都是采用离散循环坐标下降法进行优化。当固定$\boldsymbol{H}$, $\boldsymbol{W}$和$\boldsymbol{B}$时，可以把(8)写成

$$\begin{aligned}&\min_{Q}L=\left\|\widehat{\boldsymbol{U}}^{\mathrm{T}}\boldsymbol{Q}-K\widehat{\boldsymbol{Y}}^{\mathrm{T}}\right\|_{\mathrm{F}}^{2}\\&=\left\|\widehat{\boldsymbol{U}}^{\mathrm{T}}\boldsymbol{Q}\right\|_{\mathrm{F}}^{2}-2Ktr\left(\widehat{\boldsymbol{U}}^{\mathrm{T}}\boldsymbol{Q}\widehat{\boldsymbol{Y}}\right)+\left\|\boldsymbol{K}\widehat{\boldsymbol{Y}}^{\mathrm{T}}\right\|_{\mathrm{F}}^{2}\\&=\left\|\widehat{\boldsymbol{U}}^{\mathrm{T}}\boldsymbol{Q}\right\|_{\mathrm{F}}^{2}-\operatorname{tr}\left(\boldsymbol{Q}\boldsymbol{Z}^{\mathrm{T}}\right)+\text{const}\\&\text{s.t. }\boldsymbol{Q}\in\{-1,1\}^{K\times c}\end{aligned}\tag{17}$$

其中$\boldsymbol{Z}=2k\widehat{\boldsymbol{Y}}^{\mathrm{T}}\boldsymbol{U}$，$\boldsymbol{U}=\tanh(\boldsymbol{H})$，$\widehat{\boldsymbol{U}}=\operatorname{sign}(\boldsymbol{U})$。按照优化$\boldsymbol{B}$的优化过程，逐位更新$\boldsymbol{Q}$。令$\boldsymbol{q}^{\mathrm{T}}$表示$\boldsymbol{Q}$的第$k$行。$\widetilde{\boldsymbol{Q}}$是$\boldsymbol{Q}$不含$\boldsymbol{q}$的矩阵。同时，令$\boldsymbol{z}^{\mathrm{T}}$表示$\boldsymbol{Z}$的第$k$行，$\widetilde{\boldsymbol{Z}}$是$\boldsymbol{Z}$不含$\boldsymbol{z}$的矩阵，$\boldsymbol{u}^{\mathrm{T}}$是$\widehat{\boldsymbol{U}}$的第$k$行，$\widetilde{\boldsymbol{U}}$是$\widehat{\boldsymbol{U}}$不含$\boldsymbol{u}$的矩阵。

因此，待优化问题可以表示为

$$\begin{aligned}&\min_{q}\operatorname{tr}\left(\boldsymbol{q}\left(2\boldsymbol{u}^{\mathrm{T}}\widetilde{\boldsymbol{U}}^{\mathrm{T}}\widetilde{\boldsymbol{Q}}-z^{\mathrm{T}}\right)\right)+\text{const}\\&\text{s.t. }\boldsymbol{q}\in\{-1,1\}^{c}\end{aligned}\tag{18}$$

问题式(18)的最优解为

$$\boldsymbol{q}=\operatorname{sign}(z-2\widetilde{\boldsymbol{Q}}^{\mathrm{T}}\widetilde{\boldsymbol{U}}u)\tag{19}$$

从式(19)中可以看出，每个位$\boldsymbol{q}$都是基于预先学习的$K-1$位$\widetilde{\boldsymbol{Q}}$进行计算的。然后迭代更新每个位，直到算法收敛为止。

3 实验

3.1 实验设置

本文在两个广泛使用的基准数据集上进行了实验：CIFAR-10和NUS-WIDE。每个数据集被分为查询集和检索集，从检索集中随机选取训练集。CIFAR-10数据集是一个单标签数据集，包含60000张像素为32×32的彩色图像和10类图像标签。对于CIFAR-10数据集，如果两幅图像共享一个相同的标签，则它们将被认为是相似的。NUS-WIDE数据集由与标签相关的269,648幅Web图像组成。它是一个多标签数据集，其中每个图像都使用来自5018个标签的一个或多个类标签进行注释。与文献[13,14]类似，只使用属于21个最常见的标签的195,834张图像。每个类在这个数据集中至少包含5000张彩色图像。对于NUS-WIDE数据集，如果两个图像至少共享一个公共标签，则它们将被认为是相似的。

遵循文献[14]的实验设置。对于CIFAR-10数据集，每个类随机选取100张图像作为查询集，其余的图像作为检索集。从检索集中随机抽取每类500幅图像作为训练集。对于NUS-WIDE数据集，随机抽取2100幅图像(每个类100幅图像)作为查询集，其余的图像构成检索集。并且使用检索集中每类500幅图像作为训练集。对于ASDDH方法，算法的参数都是基于标准的交叉验证过程设定的。实验中设置$\alpha=1$，$\beta=1$，$\mu=0.1$，$\eta=55$，$\tau=0.001$，$\gamma=1$。为了避免正相似和负相似信息的类不平衡问题所带来的影响，将S中元素–1的权重设为元素1与元素–1在S中的数量之比。

3.2 基线和评价标准

本文选择了一些典型方法作为基线进行比较。对于基线，大致将其分为两组：传统哈希方法和深度哈希方法。传统哈希方法包括无监督哈希方法和监督哈希方法。无监督哈希方法包括SH(Spectral Hashing)[19], ITQ[5]。监督哈希方法包括SPLH(Sequential Projection Learning for Hashing)[20], KSH[9], FastH(Fast Supervised Hashing)[21], LFH(Latent Factor Hashing)[22]和SDH[10]。传统的哈希方法采用手工特征作为输入。本文将传统哈希方法的手工特征替换为由卷积神经网络提取的深度特征作为基线进行比较，比如表1中的“FastH+CNN”表示FastH方法利用卷积神经网络提取特征取代手工特征，其它方法同理。深度哈希方法包括DQN(Deep Quantization Network)[23]，DHN(Deep Hashing Network)[24]，CNNH(Convolutional Neural Network Hashing)[25]，NINH[26]，DPSH[13]，DTSH(Deep Supervised Hashing with Triplet Labels)[27]和DSDH[14]。对于深度哈希方法，首先将所有图像的大小调整为224像素×224像素，然后直接使用原始图像像素作为输入。本文采用在ImageNet数据集上预训练的AlexNet网络初始化ASDDH框架的前7层，其它深度哈希方法也采用了类似的初始化策略。

为了定量地评估本文方法和基线方法，本文采用了一种常用的度量方法：平均准确率均值(Mean Average Precision, MAP)。NUS-WIDE数据集上的MAP值是根据返回的前5000个最近邻来计算的。CIFAR-10数据集的MAP是基于整个检索集计算的。所有实验运行5次，并报告平均值。

3.3 精确度

表1报告了两个数据集上所有基线方法和提出的ASDDH方法的MAP结果。其中DSDH*表示本文重新运行DSDH原作者提供代码的实验结果。从表1可以看出：(1)本文ASDDH方法显著优于所有基线；(2)在大多数情况下，监督方法优于无监督方法。这表明深度监督哈希是一种更兼容的哈希学习体系结构。

在CIFAR-10数据集上，随着编码长度从12增加到48，ASDDH计算得到的MAP分数从0.763增加到0.785，远远优于传统的基于深度学习特征的哈希方法。基于深度哈希的基线方法中，DPSH, DTSH和DSDH的学习效果显著优于DQN, DHN, NINH和CNNH。将所提出的ASDDH方法与DPSH、DTSH和DSDH进行比较可以看出，在不同的编码长度下，ASDDH方法获得的性能都有不同程度的提高。与DSDH相比，ASDDH方法进一步提高了3%到5%的性能。

在NUS-WIDE数据集上，ASDDH在MAP性能方面取得了显著的提高。在所有编码长度情况下，ASDDH得到的MAP分数始终高于0.834，尤其是当编码长度为48时达到0.874；而DPSH, DTSH和DSDH得到的最佳结果仅为0.824，远远低于本文方法。与DPSH和DSDH相比，当编码长度在12到48之间时，所提出的方法获得了大约6%~8%的增强。与DTSH相比，ASDDH的性能也有5%左右的提高。

本文方法在NUS-WIDE数据集上的效果比CIFAR-10数据集提高得更多，主要原因是NUS-WIDE数据集内包含的图像类别比CIFAR-10数据集更多，而且每个图像都包含多个标签。损失函数中的L_3利用多标签二进制码映射进行非对称训练，使哈希码具有多标签语义信息，进一步提高了实际应用中的检索性能。因此，本文提出的ASDDH方法在NUS-WIDE多标签数据集上更有效。

3.4 特征学习网络

为了分析不同深度卷积神经网络对检索结果的影响，本文将原来ASDDH模型中的AlexNet网络改为预训练的VGG-16, ResNet50和ResNeXt50网络进行训练。VGG-16网络由13个卷积层和3个全连接层组成，比AlexNet网络更为复杂且参数更多。残差网络ResNet50包含49个卷积层和1个全连接层，该网络主要通过跳跃连接的方式，在加深网络的情况下又解决梯度爆炸和梯度消失的问题。ResNeXt是ResNet和Inception的结合体，ResNeXt结构可以在不增加参数复杂度的前提下提高准确率，同时还减少了超参数的数量。ResNeXt50和ResNet50类似，包含49个卷积层和1个全连接层。

为了得到最终的二进制代码，本文将VGG-16, ResNet50和ResNeXt50网络的最后一层用1个完全连通的哈希层(激活函数为tanh)所取代，并将这3种模型分别表示为ASDDH-V16, ASDDH-RN50和ASDDH-RNX50。同时对基于深度哈希的DSDH进行了同样的实验并做对比，以保证结论的

表 1　两个数据集上不同方法的MAP

方法	CIFAR-10 (bit)				NUS-WIDE (bit)			
	12	24	32	48	12	24	32	48
ASDDH	**0.763**	**0.771**	**0.781**	**0.785**	**0.834**	**0.851**	**0.868**	**0.874**
DSDH*[14]	0.723	0.734	0.749	0.751	0.763	0.780	0.784	0.801
DPSH[13]	0.713	0.727	0.744	0.757	0.752	0.790	0.794	0.812
DTSH[27]	0.710	0.750	0.765	0.774	0.773	0.808	0.812	0.824
DQN[23]	0.554	0.558	0.564	0.580	0.768	0.776	0.783	0.792
DHN[24]	0.555	0.594	0.603	0.621	0.708	0.735	0.748	0.758
NINH[26]	0.552	0.566	0.558	0.581	0.674	0.697	0.713	0.715
CNNH[25]	0.439	0.511	0.509	0.522	0.611	0.618	0.625	0.608
FastH+CNN[21]	0.553	0.607	0.619	0.636	0.779	0.807	0.816	0.825
SDH+CNN[10]	0.478	0.557	0.584	0.592	0.780	0.804	0.815	0.824
KSH+CNN[9]	0.488	0.539	0.548	0.563	0.768	0.786	0.790	0.799
LFH+CNN[22]	0.208	0.242	0.266	0.339	0.695	0.734	0.739	0.759
SPLH+CNN[20]	0.299	0.330	0.335	0.330	0.753	0.775	0.783	0.786
ITQ+CNN[5]	0.237	0.246	0.255	0.261	0.719	0.739	0.747	0.756
SH+CNN[19]	0.183	0.164	0.161	0.161	0.621	0.616	0.615	0.612

可靠性。DSDH对应的3种模型分别表示为DSDH-V16, DSDH--RN50和DSDH-RNX50。表2显示了CIFAR-10数据集上每个模型的MAP值。

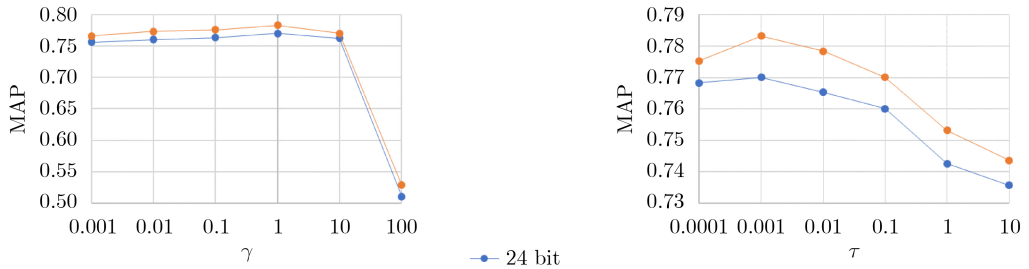
如表2所示，使用VGG-16, ResNet50和ResNeXt50网络代替AlexNet网络使得最终的检索准确率有所上升，这种趋势在基线DSDH和提出的ASDDH中都有体现。这说明不同的深度网络对模型的性能有一定的影响。并且随着网络复杂度的增加，提取的特征也更加准确，使得模型训练更加可靠。

3.5 参数敏感性分析

图2给出了针对ASDDH的超参数γ和τ在CIFAR-10数据集上的影响，二进制代码长度分别为24 bit和48 bit。值得注意的是，当对一个参数进行调优时，其他参数是固定的。例如在[0.001, 0.01, 0.1, 1, 10, 100]范围内调优γ时，分别固定其他超参数。由图2 (a)可以看出，在$0.001 < \gamma < 10$的范围内，ASDDH对γ并不敏感。同样，由图2 (b)给出的CIFAR-10数据集上不同τ的MAP结果可以知道，当τ在范围[0.0001，0.1]的时候，该方法总是取得令人满意的效果。除此之外，本文提出的ASDDH方法不管是在24 bit还是48 bit哈希码上，在$\gamma = 1, \tau = 0.001$时取得最好的结果。

表 2 不同网络的MAP

方法	CIFAR-10 (bit)			
	12	24	36	48
ASDDH	0.763	0.771	0.781	0.785
ASDDH-V16	0.783	0.792	0.798	0.810
ASDDH-RN50	0.794	0.803	0.810	0.822
ASDDH-RNX50	0.810	0.827	0.839	0.841
DSDH	0.723	0.734	0.749	0.751
DSDH-V16	0.741	0.752	0.763	0.774
DSDH--RN50	0.755	0.767	0.770	0.786
DSDH-RNX50	0.781	0.792	0.794	0.798



(a) 超参数γ的影响　(b) 超参数τ的影响

图 2 CIFAR-10数据集上的超参数影响

4 结论

本文提出了一种新的非对称深度监督离散哈希方法，即ASDDH，用于大规模的最近邻搜索。首先利用深度网络提取图像特征，将特征表示和哈希函数学习集成到端到端框架中。然后引入成对损失和分类损失来保存每对输出之间的语义结构。在此基础上，提出了一种非对称哈希方法，既能捕获离散二进制码与多标签语义之间的相似性，又能在训练阶段快速收敛。值得注意的是，非对称哈希项尤其针对多标签数据库更有效。在实际数据集上的实验表明，ASDDH在实际应用中可以达到最先进的性能。

参 考 文 献

[1] SHEN Fumin, ZHOU Xiang, YANG Yang, *et al*. A fast optimization method for general binary code learning[J]. *IEEE Transactions on Image Processing*, 2016, 25(12): 5610–5621. doi: 10.1109/TIP.2016.2612883.

[2] QIANG Haopeng, WAN Yuan, XIANG Lun, *et al*. Deep semantic similarity adversarial hashing for cross-modal retrieval[J]. *Neurocomputing*, 2020, 400: 24–33. doi: 10.1016/j.neucom.2020.03.032.

[3] 彭天强, 栗芳. 基于深度卷积神经网络和二进制哈希学习的图像检索方法[J]. 电子与信息学报, 2016, 38(8): 2068–2075. doi: 10.11999/JEIT151346.
PENG Tianqiang and LI Fang. Image retrieval based on deep convolutional neural networks and binary hashing learning[J]. *Journal of Electronics & Information Technology*, 2016, 38(8): 2068–2075. doi: 10.11999/JEIT151346.

[4] DATAR M, IMMORLICA N, INDYK P, *et al* Locality-sensitive hashing scheme based on p-stable distributions[C]. Proceedings of the 20th Annual Symposium on Computational Geometry, New York, USA, 2004: 253–262. doi: 10.1145/997817.997857.

[5] GONG Yunchao and LAZEBNIK S. Iterative quantization: A procrustean approach to learning binary codes[C]. Proceedings of CVPR 2011, Colorado Springs, USA, 2011: 817–824. doi: 10.1109/CVPR.2011.5995432.

[6] LIU Wei, MU Cun, SANJIV K, *et al*. Discrete graph hashing[C]. Proceedings of the 27th International Conference on Neural Information Processing Systems,

Montreal, Canada, 2014: 3419–3427.

[7] LU Xiaoqiang, ZHENG Xiangtao, and LI Xuelong. Latent semantic minimal hashing for image retrieval[J]. *IEEE Transactions on Image Processing*, 2017, 26(1): 355–368. doi: 10.1109/TIP.2016.2627801.

[8] DAI Bo, GUO Ruiqi, KUMAR S, *et al*. Stochastic generative hashing[C]. Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 2017: 913–922.

[9] LIU Wei, WANG Jun, JI Rongrong, *et al*. Supervised hashing with kernels[C]. Proceedings of 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, USA, 2012: 2074–2081. doi: 10.1109/CVPR.2012.6247912.

[10] SHEN Fumin, SHEN Chunhua, LIU Wei, *et al*. Supervised discrete hashing[C]. Proceedings of 2015 IEEE Conference on Computer Vision and Pattern Recognition, Boston, USA, 2015: 37–45. doi: 10.1109/CVPR.2015.7298598.

[11] SHI Xiaoshuang, XING Fuyong, XU Kaidi, *et al*. Asymmetric discrete graph hashing[C]. Proceedings of the 31st AAAI Conference on Artificial Intelligence, San Francisco, USA, 2017: 2541–2547.

[12] 陈昌红, 彭腾飞, 干宗良. 基于深度哈希算法的极光图像分类与检索方法[J]. 电子与信息学报, 2020, 42(12): 3029–3036. doi: 10.11999/JEIT190984.
CHEN Changhong, PENG Tengfei, and GAN Zongliang. Aurora image classification and retrieval method based on deep hashing algorithm[J]. *Journal of Electronics & Information Technology*, 2020, 42(12): 3029–3036. doi: 10.11999/JEIT190984.

[13] LI Wujun, WANG Sheng, and KANG Wangcheng. Feature learning based deep supervised hashing with pairwise labels[C]. Proceedings of the 25th International Joint Conference on Artificial Intelligence, New York, USA, 2016: 1711–1717.

[14] GUI Jie, LIU Tongliang, SUN Zhenan, *et al*. Fast supervised discrete hashing[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018, 40(2): 490–496. doi: 10.1109/TPAMI.2017.2678475.

[15] JIANG Qingyuan, CUI Xue, and LI Wujun. Deep discrete supervised hashing[J]. *IEEE Transactions on Image Processing*, 2018, 27(12): 5996–6009. doi: 10.1109/TIP.2018.2864894.

[16] MA Lei, LI Hongliang, WU Qingbo, *et al*. Multi-task learning for deep semantic hashing[C]. Proceedings of 2018 IEEE Visual Communications and Image Processing, Taichung, China, 2018: 1–4. doi: 10.1109/VCIP.2018.8698627.

[17] YANG Zhan, RAYMOND O I, SUN Wuqing, *et al*. Asymmetric deep semantic quantization for image retrieval[J]. *IEEE Access*, 2019, 7: 72684–72695. doi: 10.1109/ACCESS.2019.2920712.

[18] MOHAMMAD N and FLEET D J. Minimal loss hashing for compact binary codes[C]. Proceedings of the 28th International Conference on Machine Learning, Bellevue, USA, 2011: 353–360.

[19] WEISS Y, TORRALBA A, and FERGUS R. Spectral hashing[C]. Proceedings of the 21st International Conference on Neural Information Processing Systems, Vancouver, Canada, 2008: 1753–1760. doi: 10.5555/2981780.2981999.

[20] WANG Jun, KUMAR S, and CHANG S F. Sequential projection learning for hashing with compact codes[C]. Proceedings of the 27th International Conference on International Conference on Machine Learning, Haifa, Israel, 2010: 1127–1134.

[21] LIN Guosheng, SHEN Chunhua, SHI Qinfeng, *et al*. Fast supervised hashing with decision trees for high-dimensional data[C]. Proceedings of 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, USA, 2014: 1971–1978. doi: 10.1109/CVPR.2014.253.

[22] ZHANG Peichao, ZHANG Wei, LI Wujun, *et al*. Supervised hashing with latent factor models[C]. Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval, Gold Coast, Australia, 2014: 173–182. doi: 10.1145/2600428.2609600.

[23] CAO Yue, LONG Mingsheng, WANG Jianmin, *et al*. Deep quantization network for efficient image retrieval[C]. Proceedings of the 30th AAAI Conference on Artificial Intelligence, Phoenix, USA, 2016: 3457–3463.

[24] ZHU Han, LONG Mingsheng, WANG Jianmin, *et al*. Deep hashing network for efficient similarity retrieval[C]. Proceedings of the 30th AAAI Conference on Artificial Intelligence, Phoenix, USA, 2016: 2415–2421.

[25] XIA Rongkai, PAN Yan, LAI Hanjiang, *et al*. Supervised hashing for image retrieval via image representation learning[C]. Proceedings of the 28th AAAI Conference on Artificial Intelligence, Québec City, Canada, 2014: 2156–2162.

[26] LAI Hanjiang, PAN Yan, LIU Ye, *et al*. Simultaneous feature learning and hash coding with deep neural networks[C]. Proceedings of 2015 IEEE Conference on Computer Vision and Pattern Recognition, Boston, USA, 2015: 3270–3278. doi: 10.1109/CVPR.2015.7298947.

[27] WANG Xiaofang, SHI Yi, and KITANI K M. Deep supervised hashing with triplet labels[C]. Proceedings of the 13th Asian Conference on Computer Vision, Taipei, China, 2016: 70–84. doi: 10.1007/978-3-319-54181-5_5.

顾广华：男，1979年生，博士，教授，研究方向为图像检索、图像分类和图像识别.

霍文华：女，1995年生，硕士生，研究方向为图像检索和深度哈希.

苏明月：女，1996年生，硕士生，研究方向为图像检索和深度哈希.

付　灏：男，1996年生，硕士生，研究方向为跨模态检索.

责任编辑：陈　倩