

大规模图像搜索中结合深度神经网络的 语义一致性哈希的监督学习方法

Supervised Learning of Semantics-Preserving Hashing via Deep Neural Networks for Large-Scale Image Search

Huei-Fang Yang, Kevin Lin, Chu-Song Chen

摘要：本文提出了一种监督型的深度哈希方法，该方法适用于大规模图像搜索领域，能从带标签的数据中构造出二进制哈希码。我们假设语义标签由一系列隐藏属性控制，每个属性有“开”和“关”两种状态，并且分类器根据这些属性进行分类。基于这个假设，我们的方法，称为“监督型语义一致性深度哈希（SSDH）”，将哈希函数作为深度神经网络的隐藏层。此时，哈希码可以从预定义的目标函数中习得。目标函数可由分类器错误和其它哈希属性共同定义。这种设计是 SSDH 具有良好的分类和检索性能，具体说来，就是习得的哈希码能够有效保存图像间的语义相似度，并高效应用于图像检索。SSDH 在现有的深度分类架构上进行轻微修改即可获得。相比其它的半监督和监督型的哈希方法来说，SSDH 在超过 1000000 幅图像的大规模数据集上表现更为优异。

索引词：图像检索，监督型哈希，二进制码，深度学习，卷积神经网络

一、简介

图像搜索领域的语义一致性哈希已经获得了视觉社区的关注。这些哈希方法的关键在于将相似的图像映射到相似的哈希码上，换句话说，就是将高维的视觉数据投射到低维的汉明空间中。经过这一步骤，人们就可以简单地计算汉明距离来进行相似最近邻搜索，汉明距离的计算是非常快速的。

近来，基于学习的哈希方法受到了愈来愈广泛的专注，因为该方法利用了训练数据来获得哈希码，该哈希码有效于不利用训练数据获得了哈希码。以 LSH（局部敏感哈希）为例，LSH 利用随机投影，试图将相似图像以高概率投影到同一个桶里，它并没有使用训练数据。如果 LSH 想获得较高的准确率是，则它需使用更长的编码。在不同的学习型哈希方法中，监督型哈希能更好地保存图像间的语义结构。虽然监督型哈希表现优异，但是它们需要被提供训练数据，带来的结果是计算时间增加，并要足够大的存储空间去存储训练数据。这使这些方法难以在大规模图片集上实际应用。

在本文中，结合深度学习，我们提出了从带标签的训练集中生成二进制码的监督型语义一致性深度哈希方法。SSDH 最基本的思想是图像标签可以由一系列潜在属性表示，并且分类可依赖于这些属性进行。基于这一思想，我们将哈希函数作为隐藏层，图片作为输入层，分类结果作为输出层，这样就有了卷积神经网络（CNN）的基本三层。二进制码可由最小化目标函数（如分类误差）获得。该设计利用深度网络，实现了分类和搜索，同时尽可能使语义相似的图像就有相似的哈希码。此外，为了使哈希码更加紧凑，我们为目标函数添加了额外的限制，使每一位哈希码尽可能承载更多的信息。Figure 1 描述了 SSDH 的基本框架。所进行的实验结果表明 SSDH 比其它哈希表现优异。我们也在超过 1000000 幅图像

的大数据集上进行了实验来检测它的可扩展性。

总结说来，本文主要的贡献是：

1.统一了检索和分类：我们提出了一种监督型哈希方法，SSDH。该方法利用了深度学习，在同一个模型中统一了分类和检索。

2.可扩展性：SSDH 逐点学习，不需要二元组或三元组图像输入。这个特性使之能应用于大规模图像检索中。

3.轻量级：SSDH 利用深度分类架构。对深度分类网络进行轻微修改就可以实现 SSDH。

4.大规模数据实验：我们在不同的基准和一个百万级数据集上进行了大量的实验以确保 SSDH 能获得预期结果。

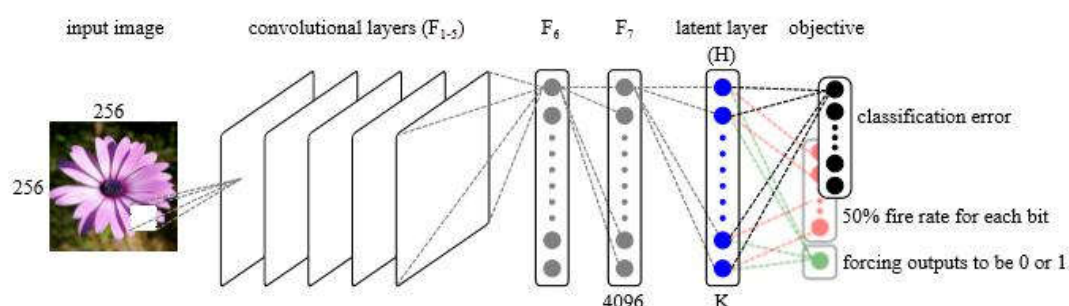


Figure 1 SSDH 基本框架。哈希函数作为隐藏层，含有 K 个神经元。输入层为图像表示，输出层为分类结果。该 CNN 中卷积层有 5 层，其中 2 层为全连接，1 层是输出层。SSDH 将 256*256 大小的图片作为输入。习得的哈希码保存了图片间的语义相似性。

该工作之前的结果在[8][9]中已有展示。该工作的扩展如下：

- (1) 提出带有额外约束的新目标函数来学习二进制编码；
- (2) 进行多组实验综合考察 SSDH 的性能及其在大数据几下的可扩展性；
- (3) 对架构设计及实验结果进行了详细的分析。

二、背景

学习型哈希算法利用训练数据获得哈希码，并能克服数据无关类方法（如 [5]-[7][10] 及 LSH 方法族）的不足。根据对监督信息（如标签数据）的利用程度，可以将学习型方法分为三类：非监督、半监督和监督方法。非监督算法（[1][3][11][12]）使用不带标签的数据构建哈希码并试图保留数据在原始空间（如欧式空间）上的领域关系。该类下具有代表性的方法是谱哈希（SH）[12]、核局部敏感哈希（KLSH）[3]、以及迭代量化（ITQ）[1]。

半监督算法[13]-[15]使用标签数据和无标签数据来学习哈希函数。比如，半监督哈希（SSH）[14]最小化标签数据对（相似数据对和不相似数据对）上的经验错误，同时最大化带标签和无标签数据的哈希码的方差。半监督标签哈希（SSTH）[15]以监督方式为哈希码与类标签的正确性建模，同时以无监督方式保存图片样本的相似度。

监督哈希[16]-[24]旨在充分利用标签数据来获得哈希码，这种方法能获得比半监督和非监督方法更高的准确度。利用样本对携带的信息，二元重构插入法（BRE）[16]最小化欧式空间和汉明空间上的平方误差。最小损失哈希（MLH）最小化编码生成的经验误差。基于排序的方法[21],[24]利用了三元组的排序信息。训练阶段依赖二元或三元图片对的方法需要较大的存储空间，因而并不适用于大

数据集。用逐点方式学习二进制编码可能是实现哈希方法可扩展性的选择。逐点的方式利用提供的标签信息指导哈希函数的学习。提振相似敏感编码 (BCCS) [25] 在提升算法 (AdaBoost) 中利用标签学习汉明加权值。我们受到启发, 为语义标签和散列函数学习的二进制码之间的关系建模, 即将语义标签转化为二进制码。在我们的发展的同时, 监督离散哈希 (SDH) [23] 制定, 提出了以优化分类为目标学习二进制散列码的方式。这和我们设计有异曲同工之妙, SDH 将散列学习分成子问题, 需要仔细选择损失函数进行分类以使整个优化是高效和可扩展的。我们的模型简化了优化过程, 并且可扩展至大规模数据集上。

在学习型哈希方法中, 基于深度网络的方法[26] - [32], 形成一个特殊的群体, 所以我们在这里单独讨论它们。其中一个最早应用深层网络的哈希方法是语义哈希 (SH) [30]。它通过堆叠受限玻尔兹曼机 (RBMs) 从原始图像构建了哈希码。习得的二进制码被视为内存地址, 从而查找相似物品可以通过简单地访问汉明距离内的内存地址来完成。深度网络也应用于深度哈希 (DH) 和监督型深度哈希 (SDH) [28]。最近, 还提出了基于卷积神经网络的哈希方法。CNNH 和 CNNH+ 的[31]采用两阶段的学习方法, 首先将成对相似性矩阵分解成近似哈希码, 然后训练 CNN 以用于学习哈希函数。该法[27]和深度语义排序哈希 (DSRH) [32]通过对给定的三元组确定排序损失。虽然这些基于深度学习的哈希方法效果不错, 但是一些在学习散列函数之前需要执行矩阵分解 (例如, CNNH 和 CNNH + [31]), 一些需要输入采取图像对 (如 SDH [28]) 或三元图像组的形式 (如[27]和 DSRH [32]), 这使得它们不适用于大数据量的情况下。

三、通过深度网络学习哈希码

设 $I = \{I_n\}_{n=1}^N$ 为 N 幅图像, $Y = \{y_n\}^{M \times N}$ 是它们的关联标签矩阵, 其中, M 代表总的类标签数。 $y_n = 1$ 表示该图像属于某个类, 否则, $y_n = 0$ 。我们的目标是学习一个映射 $F: I \rightarrow \{0,1\}^{K \times N}$, 即将图像映射到 K 位二进制码 $B = \{b_n\} \in \{0,1\}^{K \times N}$, 同时保持图片间的语义相似度。值得一提的是, 我们需要设计一个监督型的哈希算法利用语义标签自动地生成带有以下属性的哈希码:

- 1) 编码图片间的语义相似关系。具有同类标签的图片将会映射到同一个或相似的二进制码上。
- 2) 编码分布均匀且具有同等识别度。每个码位尽可能携带等量的信息。

A. 深度哈希函数

结合深度学习, 我们利用一个能够从图片中学习语义的 CNN 构造哈希函数。研究表明: 在图像分类、目标检测、语义分割和图片检索中, 深度特征比表面特征更具代表性。

基于 AlexNet 深度模型, 我们设计了一个新的网络。该网络具有 5 个最大化合并操作的正交层 (F_{1-5}), 正交层之后是 2 个全连接层 (F_{6-7}), 之后是一个输出层。

AlexNet 的隐藏层 (F_{1-7}) 由修正线性单元 (ReLU) 构成。ReLU 比非线性单元 (如

tanh 和 sigmoid 函数) 训练速度快。输出层使用 softmax 函数, 整个网络的训练目标是最大化多分类的多项式逻辑回归函数。为了将深度的概念引入哈希函数学习中, 我们在 F_7 的首部添加了一个含有 K 个单元的隐藏层 H , 正如 Figure1 中所展示。 H 和 F_7 之间是全连接的, 并且使用 sigmoid 单元, 这样激活值就被限定在 0 和 1 之间。

设 $W^H \in \mathbb{R}^{d \times K}$ 表示隐藏层权重。对于一个给定的图像 I_n , 假设该图像在 F_7 层的特征向量是 $a_n^7 \in \mathbb{R}^d$, 则 H 层的激活值计算如下: $a_n^H = \sigma(a_n^7 W^H + e^H)$, 其中 e^H 是偏置项, $\sigma(\cdot)$ 是 logistic-S 型函数, 定义为 $\sigma(z) = 1/(1 + \exp(-z))$, 其中 z 是一个实数。二进制编码函数定义如下:

$$b_n = \text{sign}(\sigma(a_n^7 W^H + e^H) - 0.5) = \text{sign}(a_n^H - 0.5) \quad (1)$$

其中, 当 $v > 0$ 时, $\text{sign}(v) = 1$, 否则 $\text{sign}(v) = 0$ 。对于向量和矩阵型数据, $\text{sign}(\cdot)$ 是逐项计算的。

B. 标签一致性二进制码

图片标签不仅可以在分类时提供信息, 在学习哈希函数时也能提供有用的监督信息。为了生成语义一致性二进制码, 我们为标签和二进制码之间的关系建模。假设语义标签可以由 K 个隐藏属性导出, 每个属性有“开”和“关”两种状态。当输入一幅图像后, 它会产生一个 K 位二进制输出, 这种分类取决于隐藏属性。这意味着通过对错分损失函数的优化, 我们可以保证语义相似的图片会拥有相似二进制编码。

设矩阵 $W^C \in \mathbb{R}^{K \times M}$ 表示二进制隐藏属性和类标签之间的线性映射关系。把这个矩阵融合进我们的网络中相当于在 H 层首部加一个分类层 (Figure1 中的黑虚线就代表 W^C)。设 \hat{y}_n 为网络对图片 I_n 的预测值 (在 Figure1 中表示为黑色的点)。考虑分类规则, 我们可以选择优化如下目标函数:

$$\arg \min_W \sum_{n=1}^N L(y_n, \hat{y}_n) + \lambda \|W\|^2 \quad (2)$$

其中, $L(\cdot)$ 是最小化分类误差的损失函数。 W 表示网络权重, λ 控制正则项的相对重要性。

C. 高效二进制码

除了保语义性, 二进制码应该足够紧凑并且具有识别度 [12]。这可以通过最小化 $\sum_{n=1}^N (\text{mean}(a_n^H) - 0.5)^2$, 这在某种程度上等价于最大化 0-1 向量的熵。但是,

只优化这一标准可能会使 H 层每一个节点的激活值在 0.5 附近。为了解决这一问题，我们加入另一标准使 H 层每一节点的激活值尽可能逼近 0 或 1。修改后的目标函数如下：

$$\arg \min_W - \sum_{n=1}^N \|a_n^H - 0.5\|^2 + \sum_{n=1}^N (\text{mean}(a_n^H) - 0.5)^2 \quad (3)$$

其中，第一项最大化激活值与最佳均值间的差异，使激活值逼近 0 或 1。第二项最小化 a_n^H 的激活值均值与最佳均值的差异。该公式使每个节点有 50% 的可能输出为 0 或 1。

在网络设计中，我们加了一个单元（Figure1 里的绿色节点）。该单元合并 H 层节点来获得公式(3) 第二项中的均激活值。该点的连接权值（Figure1 中绿色的线）需进行 1/K 修正。公式(3) 中的第一项直接使用 H 层的输出值。用 Figure1 中红色的节点指明这个输出常值。

D. 目标函数

完整的目标函数包括公式(2)(保语义性)和公式(3)（有效性）：

$$\arg \min_W \alpha \sum_{n=1}^N L(y_n, \hat{y}_n) + \lambda \|W\|^2 - \beta \sum_{n=1}^N \|a_n^H - 0.5\|^2 + \gamma \sum_{n=1}^N (\text{mean}(a_n^H) - 0.5)^2 \quad (4)$$

其中， α 、 β 、 γ 是每一项的权重，在本实验中设置为 1。

我们使用了开源的 CAFFE 包。因为我们的网络改自于 AlexNet。AlexNet 在 1.2 百万数据集 ILSVRC 上已经训练完毕。所以我们的网络中 F_{1-7} 层的初始权值使用以训练好的值，其它层的权值随机生成。训练时使用了随机梯度下降和误差反向传播算法。

四、实验

我们在三个基准和一个超过 1 百万图像的数据集上进行实验。这三个基准数据集是 CIFAR-10、MNIST 和 SUN397。大数据集 Yahoo-1M 由多种类型的产品图片组成。它用来验证 SSDH 的可扩展性。

接下来，我们将先介绍评价基准和数据集，然后对实验结果进行说明和评价。

A. 评价指标

我们使用学术界通用的评价准则。它们从不同的角度对哈希算法的表现进行了评价。

1) 平均准确率 (mAP)：根据图片与查询样例之间的汉明距离从小到大对图片进行排序，选择最靠前的 k 个作为查询结果，并计算这 k 个结果的 mAP。实验中，k 设置为 1000。mAP 表现为 P-R 曲线下的面积，是对哈希函数总体性能评价的一个指标。

2) 精确度：前 k 个检索结果图片中，正确的图片所占的比例。

3) 汉明半径 r 内的精确度：将与查询样例相距 2 个汉明距离的所有图片作为

查询结果，在这个结果上计算精确度。一个失败的查询表现为没有图片返回，精确度为 0。

B.数据集

CIFAR-10: 该数据集由 60000 幅 32*32 的彩色图像组成，一共有 10 个类别。类标签彼此独立，所以每个类别含有 6000 幅图像。将数据集拆分成两部分：训练集和测试集，其中训练集有 50000 幅图片，测试集有 10000 幅图片。

MNIST: 该数据集由 70000 幅 28*28 的灰色图像组成。每幅图都是 0-9 之间的一个手写体数字。在其中选择 60000 幅作为训练集，剩下 10000 作为测试集。

SUN397: 该数据集由 397 个类别上 108754 幅图片构成，每个类别包含至少 100 幅图片。我们随机选择 8000 幅图片作为测试集，剩下 100754 幅作为训练样本。

Yahoo-1M 商品图: 从 Yahoo 购物网站上收集了 1124086 幅有背景或无背景的杂乱的商品图片。Figure2 展示了部分图片样例。每个图片与一个类标签相关联，数据集中共有 116 个类。类图片数量存在较大差异，从 1007 到 150211 不等。从每个类中选择 90%组成训练集（1011723），剩下 10%作为测试集（112363）。

网络输入采用统一的 256*256 的图片，所以需对样例大小进行预处理。



Figure 2 Yahoo-1M 数据集中部分样例

C.CIFAR-10 上的实验结果

我们将我们的方法和其它几种无监督（LSH[5]、ITQ[1]、SH[12]）和有监督（BRE[16]、MLH[20]、CCA-ITQ[1]、CNNH+[31]、CNNH[31]、Lai et al.[27]）哈希算法比较。其中，CNNH+，CNNH 和 Lai et al.也利用了深度学习的思想。

Figure 3a 展示了不同方法在不同编码长度下头 1000 幅返回结果的 mAP。如图所示，SSDH 在不同长度的编码上表现最佳，比 Lai et al.提高了至少 28%。这些结果表明利用语义标签学习二进制码更能保存图片间的语义结构，也更能取得更好的效果。此外，可以发现 1) 监督方法比无监督方法效果好；2) 在监督方法中，基于深度学习的方法给出的结果相对较好。

Figure 3b 展示了 k 样例上的精确度，k 取值范围是 100 至 1000，哈希码长度固定为 48 位。这图隐藏的信息和 Figure 3a 差不多。

Figure 3c 展示了汉明半径为 2 时的精确度。虽然随着哈希码长度的增加我们的方法效果逐渐变差，但我们的方法比其他方法效果好。这是因为我们的方法需要更多的潜在层节点去学习更长的码数。但是，CIFAR-10 的训练数据相对较少，深度网络训练可能会出现过拟合。

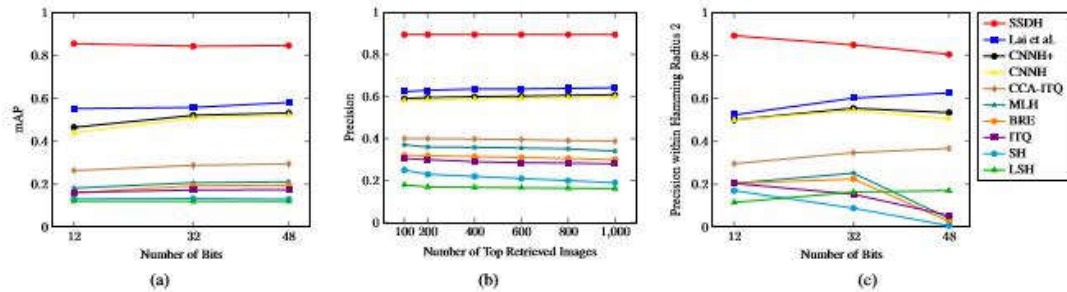


Figure 3 不同方法在 CIFAR-10 数据集上的实验结果

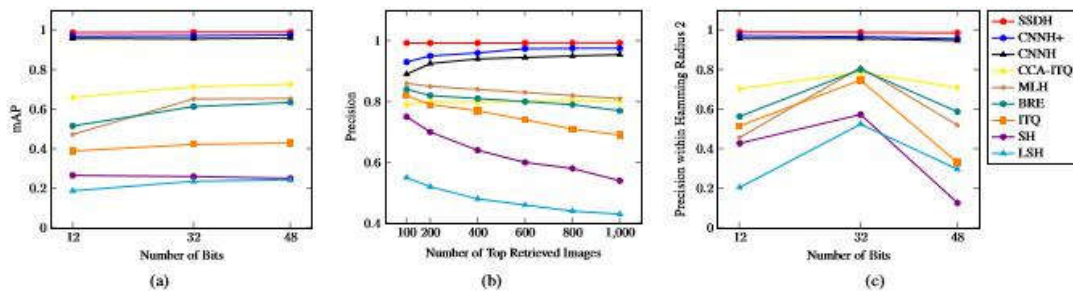


Figure 4 不同方法在 MNIST 数据集上的实验结果

D.MNIST 上的实验结果

Figure 4 展示了不同哈希算法在 MNIST 上的实验结果。我们可以发现，在这个数据集上，SSDH 的效果比其他监督和无监督方法好得多，基于深度学习的方法给出的结果相对较好。

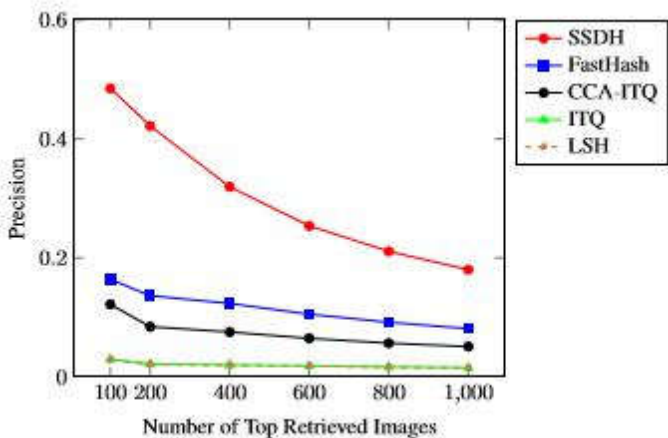


Figure 5 1024 位码长下不同取值 k 时的精确度

E.SUN397 上的实验结果

SUN397 数据集共有 397 个类别超过 100000 幅图片，所以比 CIFAR-10 和 MNIST 更具挑战性，它可能需要更长的编码来训练哈希函数以取得较好的效果。

Figure 5 是在哈希码长固定为 1024，k 取值不同时，SSDH、FastHash[18]、CCA-ITQ、ITQ 和 LSH 各自的精确度。从中可以看出，不管 k 取值多少，SSDH 总是比其他方法表现优异。当 k 取值为 200 时，SSDH 比 FastHash 高出 30%。因此，即便在长哈希码下，SSDH 也比现有的最好的哈希算法表现好。

Method	mAP
l_2 + AlexNet-ft	0.4895
LSH + AlexNet-ft	0.4639
ITQ + AlexNet-ft	0.5386
CCA-ITQ + AlexNet-ft	0.6169
SSDH	0.6470

Table 1 1000 幅返回结果上的 mAP

F.Yahoo-1M 数据集上的实验结果

我们已经证明，使用深度特征能取得更好的检索效果。我们在 Yahoo-1M 上对 AlexNet 进行调整。为了探求哈希方法取得良好效果的原因，我们也用欧式距离进行实验，并将实验结果与调整后的 AlexNet 进行比较。在这个实验中，设置哈希码长度为 128。

表 1 展示了 1000 幅返回结果上的 mAP。Figure 6 展示了不同数目返回结果上的精确度。可以发现，所有的哈希算法（LSH 除外）都比直接使用欧氏距离得到的效果好。

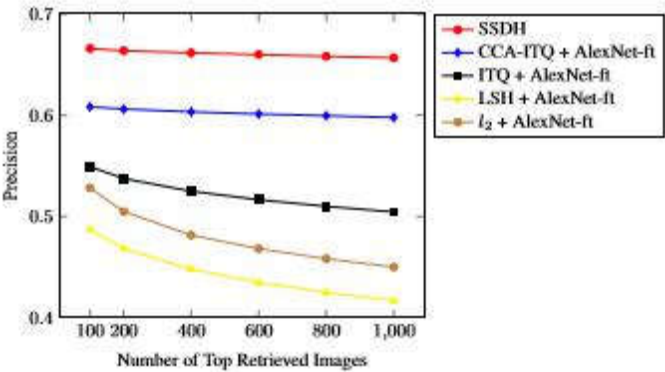


Figure 6 不同数目返回结果上的精确度

Method	Test Error (%)
Stochastic Pooling [42]	15.13
CNN + Spearmint [43]	14.98
AlexNet + Fine-tuning	11.69
NIN + Dropout [44]	10.41
NIN + Dropout + Augmentation [44]	8.81
SSDH w/ 12-bit codes	10.69
SSDH w/ 32-bit codes	10.66
SSDH w/ 48-bit codes	10.47
SSDH w/ 64-bit codes	10.25

Table 2 不同方法在 CIFAR-10 数据集上的精确度

Method	Test Error (%)
Stochastic Pooling [42]	0.47
AlexNet + Fine-tuning	0.84
NIN + Dropout [44]	0.47
SSDH w/ 12-bit codes	0.75
SSDH w/ 32-bit codes	0.72
SSDH w/ 48-bit codes	0.78
SSDH w/ 64-bit codes	0.62

Table 3 不同方法在 MNIST 数据集上的精确度

G. 图片分类

因为 SSDH 将检索和分类统一于一个模型中，所以它也可以用于图像分类问题。我们在 CIFAR-10 和 MNIST 上比较 SSDH 和其他先进的哈希算法的效果。表 2 和表 3 展示了不同方法的准确度。可以看出，SSDH 比其他方法的准确度更高。这些结果表明，将检索和分类统一到一个模型中能取得不错的检索和分类结果。

五、总结

我们提出了一个监督型哈希模型，SSDH，它能保存标签之间的语义关系。SSDH 在特征层和分类层之间加入了一个代表哈希函数的潜在层。通过优化分类误差目标函数，SSDH 能够学习二进制码、特征值和分类结果。这样的网络具有如下的优点：1) SSDH 在一个模型中统一了检索和分类；2) 通过简单修改现有的深度网络模型就能得到 SSDH 模型；3) SSDH 能扩展到大数据集上。我们进行了若干组实验，在三个基准和一个百万级数据集上比较了 SSDH 和现有的先进的哈希算法。结果显示 SSDH 能取得更好的检索效果和分类结果。

参考文献

- [1] Y. Gong, S. Lazebnik, A. Gordo, and F. Perronnin, "Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval," IEEE Trans. Pattern Anal. Mach. Intell., vol. 35, no. 12, pp. 2916–2929, 2013.
- [2] J. He, W. Liu, and S. Chang, "Scalable similarity search with optimized kernel hashing," in Proc. ACM SIGKDD, 2010, pp. 1129–1138.
- [3] B. Kulis and K. Grauman, "Kernelized locality-sensitive hashing," IEEE Trans. Pattern Anal. Mach. Intell., vol. 34, no. 6, pp. 1092–1104, 2012.
- [4] X. Liu, J. He, B. Lang, and S. Chang, "Hash bit selection: A unified solution for selection problems in hashing," in Proc. CVPR, 2013, pp. 1570–1577.
- [5] A. Andoni and P. Indyk, "Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions," in Proc. FOCS, 2006, pp. 459–468.
- [6] M. Charikar, "Similarity estimation techniques from rounding algorithms," in Proc. ACM STOC, 2002, pp. 380–388.
- [7] A. Gionis, P. Indyk, and R. Motwani, "Similarity search in high dimensions via

hashing,” in Proc. VLDB, 1999, pp. 518–529.

[8] K. Lin, H.-F. Yang, J.-H. Hsiao, and C.-S. Chen, “Deep learning of binary hash codes for fast image retrieval,” in Proc. CVPRW on DeepVision: Deep Learning in Computer Vision, 2015.

[9] K. Lin, H.-F. Yang, K.-H. Liu, J.-H. Hsiao, and C.-S. Chen, “Rapid clothing retrieval via deep learning of binary codes and hierarchical search,” in Proc. ICMR, 2015.

[10] M. Raginsky and S. Lazebnik, “Locality-sensitive binary codes from shift-invariant kernels,” in Proc. NIPS, 2009, pp. 1509–1517.

[11] W. Liu, J. Wang, S. Kumar, and S. Chang, “Hashing with graphs,” in Proc. ICML, 2011, pp. 1–8.

[12] Y. Weiss, A. Torralba, and R. Fergus, “Spectral hashing,” in Proc. NIPS, 2008, pp. 1753–1760.

[13] Y. Mu, J. Shen, and S. Yan, “Weakly-supervised hashing in kernel space,” in Proc. CVPR, 2010, pp. 3344–3351.

[14] J. Wang, S. Kumar, and S. Chang, “Semi-supervised hashing for largescale search,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 34, no. 12, pp. 2393–2406, 2012.

[15] Q. Wang, L. Si, and D. Zhang, “Learning to hash with partial tags: Exploring correlation between tags and hashing bits for large scale image retrieval,” in Proc. ECCV, 2014, pp. 378–392.

[16] B. Kulis and T. Darrell, “Learning to hash with binary reconstructive embeddings,” in Proc. NIPS, 2009, pp. 1042–1050.

[17] G. Lin, C. Shen, D. Suter, and A. van den Hengel, “A general two-step approach to learning-based hashing,” in Proc. ICCV, 2013, pp. 2552–2559.

[18] G. Lin, C. Shen, Q. Shi, A. van den Hengel, and D. Suter, “Fast supervised hashing with decision trees for high-dimensional data,” in Proc. CVPR, 2014, pp. 1971–1978.

[19] W. Liu, J. Wang, R. Ji, Y. Jiang, and S. Chang, “Supervised hashing with kernels,” in Proc. CVPR, 2012, pp. 2074–2081.

[20] M. Norouzi and D. J. Fleet, “Minimal loss hashing for compact binary codes,” in Proc. ICML, 2011, pp. 353–360.

[21] M. Norouzi, D. J. Fleet, and R. Salakhutdinov, “Hamming distance metric learning,” in Proc. NIPS, 2012, pp. 1070–1078.

[22] G. Shakhnarovich, P. A. Viola, and T. Darrell, “Fast pose estimation with parameter-sensitive hashing,” in Proc. ICCV, 2003, pp. 750–759.

[23] F. Shen, C. Shen, W. Liu, and H. T. Shen, “Supervised discrete hashing,” in Proc. CVPR, 2015.

[24] J. Wang, W. Liu, A. X. Sun, and Y. Jiang, “Learning hash codes with listwise supervision,” in Proc. ICCV, 2013, pp. 3032–3039.

[25] G. Shakhnarovich, “Learning task-specific similarity,” Ph.D. dissertation, Massachusetts Institute of Technology, 2005.

[26] A. Krizhevsky and G. E. Hinton, “Using very deep autoencoders for content-based image retrieval,” in Proc. ESANN, 2011.

[27] H. Lai, Y. Pan, Y. Liu, and S. Yan, “Simultaneous feature learning and hash

- coding with deep neural networks,” in Proc. CVPR, 2015.
- [28] V. E. Liong, J. Lu, G. Wang, P. Moulin, and J. Zhou, “Deep hashing for compact binary codes learning,” in Proc. CVPR, 2015.
- [29] A. Torralba, R. Fergus, and Y. Weiss, “Small codes and large image databases for recognition,” in Proc. CVPR, 2008.
- [30] R. Salakhutdinov and G. E. Hinton, “Semantic hashing,” *Int. J. Approx. Reasoning*, vol. 50, no. 7, pp. 969–978, 2009.
- [31] R. Xia, Y. Pan, H. Lai, C. Liu, and S. Yan, “Supervised hashing for image retrieval via image representation learning,” in Proc. AAAI, 2014.
- [32] F. Zhao, Y. Huang, L. Wang, and T. Tan, “Deep semantic ranking based hashash for multi-label image retrieval,” in Proc. CVPR, 2015.
- [33] D. C. Ciresan, U. Meier, and J. Schmidhuber, “Multi-column deep neural networks for image classification,” in Proc. CVPR, 2012.
- [34] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in Proc. CVPR, 2014.
- [35] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, “Overfeat: Integrated recognition, localization and detection using convolutional networks,” in Proc. ICLR, 2014.
- [36] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, “Learning hierarchical features for scene labeling,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1915–1929, 2013. [Online]. Available: <http://doi.ieeecomputersociety.org/10.1109/TPAMI.2012.231>
- [37] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” in Proc. NIPS, 2012.
- [38] X. Glorot, A. Bordes, and Y. Bengio, “Deep sparse rectifier neural networks,” in Proc. AISTATS, vol. 15, 2011, pp. 315–323.
- [39] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. B. Girshick, S. Guadarrama, and T. Darrell, “Caffe: Convolutional architecture for fast feature embedding,” in Proc. ACM MM, 2014, pp. 675–678.
- [40] A. Krizhevsky, “Learning multiple layers of features from tiny images,” Computer Science Department, University of Toronto, Tech. Report, 2009.
- [41] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba, “SUN database: Large-scale scene recognition from abbey to zoo,” in Proc. CVPR, 2010, pp. 3485–3492.
- [42] M. D. Zeiler and R. Fergus, “Stochastic pooling for regularization of deep convolutional neural networks,” in Proc. ICLR, 2013.
- [43] J. Snoek, H. Larochelle, and R. P. Adams, “Practical bayesian optimization of machine learning algorithms,” in Proc. NIPS, 2012, pp. 2960–2968.
- [44] M. Lin, Q. Chen, and S. Yan, “Network in network,” in Proc. ICLR, 2014.