基于 PCA 的哈希图像检索算法*

苏毅娟[1]，余 浩[2†]，雷 聪[2]，郑 威[2]，李永钢[2]
(1. 广西师范学院 计算机与信息工程学院，南宁 530023；2. 广西师范大学 广西多源信息挖掘与安全重点实验室，广西 桂林 541004)

摘 要： 为了解决传统图像检索算法低效和耗时的缺点，提出一种基于 PCA 哈希的图像检索算法。通过结合 PCA 与流形学习将原始高维数据降维；然后通过最小方差旋转得到哈希函数和二值化阈值，进而将原始数据矩阵转换为哈希编码矩阵；最后通过计算样本间汉明距离得到样本相似性。在三个公开数据集上的实验结果表明，提出的哈希算法在多个评价指标下均优于现有算法。

关键词： 哈希；图像检索；主成分分析；流形学习
中图分类号： TP181；TP301.6 **文献标志码：** A **文章编号：** 1001-3695(2018)10-3147-04
doi:10.3969/j.issn.1001-3695.2018.10.062

PCA hashing for image data retrieval

Su Yijuan[1], Yu Hao[2†], Lei Cong[2], Zheng Wei[2], Li Yonggang[2]
(1. *College of Computer & Information Engineering, Guangxi Teachers Education University, Nanning* 530023, *China*; 2. *Guangxi Key Laboratory of Multi-source Information Mining & Security, Guangxi Normal University, Guilin Guangxi* 541004, *China*)

Abstract: In order to solve the inefficiency and time-consuming of traditional image retrieval algorithms, this paper proposed an image retrieval algorithm based on PCA hash. Specifically, by combining PCA and manifold learning, it reduced the dimensionality of the original high-dimensional data, and then obtained hash function and the binarization by minimum variance rotation. Then it converted the raw data matrix to a hash coded matrix. Finally, obtained the sample similarity by calculating the Hamming distance between samples. The experimental results on three public datasets show that the proposed hash algorithm outperforms the existing algorithms under multiple evaluation criteria.

Key words: hashing; image retrieval; principal component analysis; manifold learning

0 引言

常见的图像检索技术主要包括基于索引的图像检索[1,2]和基于哈希的图像检索[3]。基于索引的图像检索方法只对低维数据集有较好的效果，基于哈希的图像检索通过将图像转换为二进制哈希编码表示，要求哈希编码尽可能地保留数据初始空间中的相近关系。使用哈希编码表示图像数据所需要的存储空间会大幅减小，且在高维图像数据集进行相似近邻检索只需要很小的检索成本。因此，基于哈希的图像检索成为了大数据研究的一个重要方向。

常见的哈希算法有基于局部敏感的哈希方法，如局部敏感哈希(locality sensitive hashing，LSH)[4]、核局部敏感哈希(kernelized locality sensitive hashing，KLSH)[5])和基于学习的哈希方法，如多维谱聚类哈希(multidimensional spectral hashing，MDSH)[6]、迭代量化哈希(iterative quantization hashing，ITQ)[7]、锚点图哈希(anchor graph hashing，AGH)[8]等。基于局部敏感哈希算法的哈希函数构造方法的训练和测试时间与样本个数无关，但需要多个哈希表来取得合理的哈希成立，所以常被称为数据独立的哈希算法。基于学习的哈希算法通过考虑数据内在的信息构造哈希函数需要较长的训练时间和常数级的测试时间。

基于学习的哈希算法主要包括基于主成分分析哈希和基于流形学习哈希两类。基于主成分分析的哈希算法通过保留最大方差来构造哈希函数，而基于流形学习的哈希算法通过保留数据的局部相似性构造哈希函数。然而目前尚未有文献考虑同时结合这两种技术。

本文提出了一种同时结合主成分分析和流形学习的高效哈希算法，具体地说：a) 通过主成分分析考虑数据的全局结构，同时通过流形学习考虑数据的局部结构；b) 通过最小方差旋转模型同时学习哈希函数和二值化阈值。此方法称做基于 PCAR 哈希算法(PCA rotation hashing)。与之前只考虑其中一种相似结构的哈希算法相比，本文方法在构造哈希函数的同时考虑到了保留原始数据局部和全局的相似结构，所以本方法可取得更好的哈希效果。本文提出的 PCAR 算法具有以下优点：

a) 与传统数据独立哈希算法不同，本文提出的哈希算法通过考虑原始数据的结构分布来构造哈希函数，一定程度上增加了哈希函数与真实数据的关联性，确保能取得更好的哈希效果。

b) 与之前的依赖数据的哈希算法通常只考虑一种相似结构相比，本算法同时考虑到保留原始数据整体和局部的相似结构，因此学习到的哈希函数比仅考虑一种相似结构的依赖数据的哈希算法取得了更加显著的表现。实验结果证明在两种相

收稿日期：2017-06-19；修回日期：2017-08-04 基金项目：国家自然科学基金资助项目(61672177，61573270)；国家“973”计划资助项目(2013CB329404)；广西自然科学基金资助项目(2015GXNSFCB139011，2015GXNSFAA139306)；广西研究生教育创新计划资助项目(XYCSZ2017064，XYCSZ2017067，YCSW2017065)

作者简介：苏毅娟(1976-)，女，广西桂林人，副教授，主要研究方向为机器学习、数据挖掘；余浩(1994-)，男(通信作者)，江西上饶人，硕士，主要研究方向为数据挖掘、机器学习(yuhao.gxnu@qq.com)；雷聪(1991-)，男，湖北大冶人，硕士研究生，主要研究方向为数据挖掘、机器学习；郑威(1989-)，男，吉林延吉人，硕士研究生，主要研究方向为数据挖掘、机器学习；李永钢(1989-)，男，河北保定人，硕士，主要研究方向为数据挖掘、机器学习.

似结构之间得到了一些额外的信息，使本文的哈希算法具有更好的效果。

c) 针对原始高维图像数据难以直接处理的问题，传统图像哈希算法通常使用低维的二进制编码来表征高维图像数据。本文算法在已有哈希算法的基础上进一步缩短了单张图像二进制码的位数。例如，可拓展图哈希(scalable graph hshing，SGH)算法中表征高维图像数据的比特数大部分都超过了 80 bit[9]，但是本文算法最多只需要 64 bit。所以本算法在取得良好效果的同时提高了哈希的效率。

d) 与现有的哈希算法分别学习哈希函数和二值化阈值不同，本文提出的哈希算法通过最小方差旋转模型构造哈希函数。此模型中包含了一个最小方差正则项 $\rho\sum_{i=1}^{n}\| y_i-\mu\|_2^2$，通过此正则项可以同时学习到哈希函数和二值化阈值，从而大大减小了现有的一些哈希算法在分别学习哈希函数和二值化阈值过程中造成的误差。

1 相关理论背景

1.1 PCA 哈希简介

PCA 哈希是一种基于 PCA 的哈希方法。图像哈希算法一般情况下是指通过哈希函数将高维图像数据转换为二进制编码。在构造哈希函数时，期望能用较短的哈希编码尽可能地表征原始数据的大部分信息。根据香农信息理论，一组方差较大的数据能携带较大的信息量。所以基于 PCA 的哈希算法首先假设最大化每个比特之间的方差，得到如下公式:

$$\arg\max_{T} J(T)=\arg\max_{T}\sum_{k}\mathrm{var}(\mathrm{sgn}(X^{\mathrm{T}}t_k^{\mathrm{T}})) \tag{1}$$

其中: $X\in\mathrm{R}^{d\times n}$ 为样本矩阵; $T\in\mathrm{R}^{c\times d}$ 是降维矩阵; k 是哈希函数使用比特的位数($k=1,\cdots,c$)，其中 d 和 n 分别表示样本维度和样本数量。本文采用 $\mathrm{sgn}(v)$ 作为二值化函数，$\mathrm{var}(\cdot)$ 表示一个向量(矩阵)的方差。

因为相互正交的两个向量之间没有关联，能够消除数据中的冗余属性，所以在哈希函数中加入限制条件 $TT^{\mathrm{T}}=I_c$，而且由于式(1)的离散问题，导致公式难以直接求解，将式(1)转换为如下目标函数:

$$\widetilde{J}(T)=\frac{1}{n}\sum_{k}t_kXX^{\mathrm{T}}t_k^{\mathrm{T}}=\frac{1}{n}\mathrm{tr}(TXX^{\mathrm{T}}T^{\mathrm{T}})$$
$$\text{s.t.}\quad TT^{\mathrm{T}}=I_c \tag{2}$$

PCA(主成分分析)方法是一种常见的数据分析方法，通过线性变换将原始数据变为一组各维度线性无关的表示，通过选定协方差矩阵中特征值最大的 k 个特征向量重新组合成低维矩阵，达到降维的目的。本算法通过上述公式，将 PCA 方法引入到本文算法模型。当目标降维矩阵 T 是 XX^{T} 的特征向量矩阵时，得到的哈希编码方差最大[10]，于是本文将特征值最大的 k 个特征向量组合成目标降维矩阵 T，用于保留原始数据整体的相似结构。

1.2 流形哈希简介

流形哈希(manifold based hashing)主要是指采用了流形学习方法的哈希方法，本文采用的流形学习方法是指拉普拉斯特征映射(Laplacian eigenmaps)，主要思想是期望关系图中有关联的点在降维后仍然保留关联关系，用局部的角度去构建数据之间的关系，因此拉普拉斯特征映射可以反映出数据内在的流形结构。与 PCA 哈希保留原始数据整体的相似结构不同，流形哈希保留数据局部的相似结构，即为每个数据点采用 K 最近邻(KNN)算法[11]寻找其关联数据点。

具体地，首先建立一个相似结构矩阵 $S\in\mathrm{R}^{n\times n}$，步骤如下:

a) 构建图。通过 K 最近邻(KNN)算法将每个点最近的 k 个点相连。

b) 确定权重。本文采用简化的权重设定方法，即: 如果点 i 和 j 相连，权重设为 1; 否则，设为 0。

通过上述步骤得到的稀疏 KNN 图构造一个与原始数据有相似结构的矩阵 S，达到保留数据局部相似结构的目的，同时也减少了计算消耗。然后定义一个对角矩阵 $D\in\mathrm{R}^{n\times n}$，降维后的目标矩阵 $Z\in\mathrm{R}^{c\times n}$，$D$ 矩阵中的对角线上的元素等于矩阵 S 的每行元素之和，即 $D_{i,i}=\sum_{j=1}^{n}s_{ij}$，然后让 $L=D-S$，得到了如下的目标函数:

$$\min_{B}\mathrm{tr}(ZLZ^{\mathrm{T}})\qquad \text{s.t.}\quad ZZ^{\mathrm{T}}=I_c\quad Z^{\mathrm{T}}1=0 \tag{3}$$

其中: 矩阵 L 是一个图拉普拉斯矩阵，包含了原始数据的局部相似结构。限制条件 $ZZ^{\mathrm{T}}=I_c$ 保证优化问题有解，并且保证映射后的数据点不会被压缩到一个小于 c 维的子空间内。使得公式最小化的 Z 的列向量是广义特征值问题的 c 个最小非 0 特征值对应的特征向量。

基于流形学习的哈希方法[12]通过拉普拉斯特征映射保留数据局部的相似结构来构造哈希函数。已有研究证明，基于流形学习的哈希算法一定情况下优于基于 PCA 的哈希算法和基于 LSH 的哈希算法，但是需要较大的时间成本。

2 算法描述和优化

2.1 算法描述

本文算法主要包括两部分: a) 通过结合 PCA 哈希与流形哈希的图 PCA 哈希算法，得到最优降维矩阵 T 以及原始数据降维之后的新表征数据矩阵 Z; b) 通过最小方差旋转，得到哈希函数和二值化阈值。通过步骤 a) 可以把原始的高维真实数据转换成低维的真实数据，步骤 b) 把低维真实数据转换成二进制编码[13]。给定样本 $X\in\mathrm{R}^{d\times n}$，其中 d 和 n 分别代表维度和样本的数量。定义 $\mathrm{sgn}(v)$ 作为二值化函数形式如下:

$$\mathrm{sgn}(v)=\begin{cases}1 & \text{if } v\geqslant 0\\ -1 & \text{otherwise}\end{cases} \tag{4}$$

假定 X 是一个零中心矩阵，即 $\sum_{j=1}^{n}x_{j,i}=0(i=1,\cdots,d)$。本文的目标就是学习旋转矩阵 $G=\mathrm{R}^{c\times c}$，通过旋转矩阵 G 推导出哈希函数进行哈希处理。

1) 图 PCA 算法

a) 对式(2)进行简化得到如下目标函数:

$$\max\mathrm{tr}(TXX^{\mathrm{T}}T^{\mathrm{T}})\qquad \text{s.t.}\quad TT^{\mathrm{T}}=I_c \tag{5}$$

b) 根据降维公式 $Z=TX$ 将式(3)转换为

$$\min_{W}\mathrm{tr}(TXLX^{\mathrm{T}}T^{\mathrm{T}})\quad \text{s.t.}\quad TT^{\mathrm{T}}=I_c \tag{6}$$

c) 通过结合式(5)(6)，得到如下的目标函数:

$$\min\mathrm{tr}(TXLX^{\mathrm{T}}T^{\mathrm{T}})-\lambda\,\mathrm{tr}(TXX^{\mathrm{T}}T^{\mathrm{T}})\qquad \text{s.t.}\quad TT^{\mathrm{T}}=I_c \tag{7}$$

其中: λ 是一个可以调节的参数，得到最终的目标函数公式为

$$\min\mathrm{tr}(TX(L-\lambda I)X^{\mathrm{T}}T^{\mathrm{T}})\quad \text{s.t.}\quad TT^{\mathrm{T}}=I_c \tag{8}$$

其中: $I_n\in\mathrm{R}^{n\times n}$ 是一个单位矩阵。式(7)通过结合式(5)中最大化整体相似结构保留和式(6)中最小化局部相似结构保留同时取得两种相似结构保留。通过调节参数 λ 来平衡两种相似结构的权重[14]。最后，求出最优的降维矩阵 T，也就意味着得到的原始数据 X 的新表征数据矩阵: $Z=TX\in\mathrm{R}^{c\times n}$。

2) 最小方差旋转

在 a) 中得到了原始数据 X 的新表征数据矩阵 $Z\in\mathrm{R}^{c\times n}$，通过图 PCA 算法把原始的高维数据变成了低维数据。然后需要学习一个哈希函数将每个数据点转换为二进制编码[15]。为此，本文通过一个旋转矩阵将每个数据点编码成向量然后进行二值化来得到最终的哈希编码的方法来对数据进行预处理。通过最小化目标编码矩阵与本文求得的最终编码矩阵的差提出如下模型来学习目标旋转矩阵:

$$\min_{Y,G}\|Y-G^{\mathrm{T}}Z\|_F^2+\rho\sum_{i=1}^{n}\|y_i-\mu\|_2^2 \tag{9}$$

其中: $G \in \mathrm{R}^{c\times c}$为旋转矩阵; $Y = [y_1,\cdots,y_n] \in \mathrm{R}^{c\times n}$假设是量化之后的哈希编码矩阵; y_i 是每个样本的哈希编码; $\mu = \frac{1}{n}\sum_{i=1}^{n} y_i$ 是所有哈希编码向量的均值向量; 正则化项 $\sum_{i=1}^{n} \| y_i - \mu \|_2^2$ 的主要作用是通过减小每个哈希编码的值与均值(阈值)间的偏差,降低在学习哈希函数过程中由于离群点造成的影响; ρ 用来调节惩罚项的权重。

因为本文提出的最小方差旋转模型同时优化旋转矩阵 G 和二值化阈值向量 μ,所以在得到编码矩阵的同时得到了均值向量 μ。然后采用旋转矩阵推导得到最终的哈希函数,采用均值向量 μ 作为二值化阈值进行二值化,最终得到二值编码矩阵 $B = [b_1,\cdots,b_n]$。与 PCAR 相反,很多现有的哈希算法[16]单独地学习哈希函数和二值化阈值。本文提出的算法同时学习哈希函数和二值化阈值,从而大大减小了哈希过程造成的信息缺失[17]。因此本文的哈希算法能取得更好的哈希效果。

2.2　算法优化

本节对式(9)所示的目标函数进行优化。由于目标函数是凸函数且非光滑,难以直接求出最优解,所以本文采用两步交替优化法[18]进行求解。

a) 固定 Y,问题变成:

$$\min_{G} \| Y - G^{\mathrm{T}} Z \|_F^2 \tag{10}$$

很容易解得:

$$G^{\mathrm{T}} = YZ^{-1} \tag{11}$$

b) 固定 G,问题式(10)变成:

$$\min_{Y} \| Y - G^{\mathrm{T}} Z \|_F^2 + \rho \sum_{i=1}^{n} \| y_i - \mu \|_2^2 \tag{12}$$

通过上面的公式进行数学推导,得到了 y_i 的结果:

$$\hat{y}_i = \arg\min_{Y} \| Y - G^{\mathrm{T}} Z \|_F^2 + \rho \sum_{i=1}^{n} \langle y_i - \mu, y_i - \mu \rangle =$$

$$\arg\min_{Y} \| Y - G^{\mathrm{T}} Z \|_F^2 + \rho \sum_{i=1}^{n} (\langle y_i, y_i \rangle - 2\langle y_i, \mu \rangle + \langle \mu, \mu \rangle) =$$

$$\arg\min_{Y} \| Y - G^{\mathrm{T}} Z \|_F^2 + \rho \sum_{i=1}^{n} \| y_i \|_2^2 - \rho n \| \mu \|_2^2 \tag{13}$$

对上式进行求导,并令导数为 0:

$$(1+\lambda)\hat{y}_i - (G^{\mathrm{T}} Z)_i = \rho \frac{1}{n} \sum_{i=1}^{n} (G^{\mathrm{T}} Z)_i \tag{14}$$

最后得到 y_i 的最优解:

$$\hat{y}_i = \frac{1}{1+\rho}\left((G^{\mathrm{T}} Z)_i + \rho \frac{1}{n} \sum_{i=1}^{n} (G^{\mathrm{T}} Z)_i\right) \tag{15}$$

得到所有的 y_i 之后,同时得到了二值化均值向量 $\mu = \mathrm{mean}(y_i)$,进行二值化 $y_i (i = 1,\cdots,n)$,公式为

$$\begin{cases} b_i(j) = 1 & \text{if } a_i(j) \geqslant \mu(j) \\ b_i(j) = 0 & \text{if } a_i(j) < \mu(j) \end{cases} \tag{16}$$

其中: $j = 1,\cdots,n$,使用均值向量 μ 作为二值化阈值向量,最终得到原始数据集 X 的哈希编码矩阵 $B = [b_1,\cdots,b_n]$。

算法 1　PCAR 算法的伪代码

输入: 训练样本 $X \in \mathrm{R}^{d\times n}$,控制参数 ρ,比特数量 c,随机初始化 $Y = [y_1,\cdots,y_n] \in \mathrm{R}^{c\times n}$,随机初始化 $G \in \mathrm{R}^{c\times c}$。

输出: 矩阵 $B \in \mathrm{R}^{c\times n}$, $T \in \mathrm{R}^{c\times n}$, $\mu \in \mathrm{R}^{c}$。

1　通过式(8)计算转换矩阵 T;
2　通过 $Z = TX \in \mathrm{R}^{c\times n}$,得到原始数据 X 的低维表征 Z;
3　重复:
4　通过式(9)计算 G;
5　通过式(9)计算 Y;
6　直到收敛;
7　通过式(15)得到最终的哈希函数;
8　通过 $\mu = \mathrm{mean}(y_i)$ 计算 Y 所有样本的均向量得到二值化阈值向量;
9　通过 μ 得到最终哈希编码矩阵 B;
10　结束。

3　实验结果和分析

3.1　实验数据集和对比算法

本文所有实验均在 Windows 7 系统的 MATLAB 2014a 软件下运行测试,系统内存为 64 GB。实验使用的数据集介绍如下:

CIFAR-10 数据集包括 60 000 张彩色图像,并把这些图像分成 10 类,其中每类包含 6 000 个图像。每张图像由一个 512 维的特征向量表示,在本文实验中,选择 50 000 张图像(每类 5 000张图像)作为训练步骤中学习哈希函数使用的代表数据集,剩余的 10 000 张图像作为测试数据集。

MNIST 数据集包括 70 000 张图像,这些图像都是从 0 ~ 9 的手写数字,每张图像由一个 784 维的特征向量表示。本文把数据集分成两部分,即 60 000 张图像作为训练数据集,10 000 张图像作为测试数据集。

UQCIFAR 数据集包括 60 000 张图像,这些图像分为 10 类,每张图由一个 1 024 维的特征向量表示。本文把数据集分成两部分,50 000 张图像作为训练数据集,剩余 10 000 张图像当做测试数据集。

为了验证本文算法的性能,选择了以下七种目前较好的哈希算法作为对比算法。

AGH(anchor graph hashing): 是一个基于流形学习的哈希算法,首先为每个样本生成新的表征,然后使用新表征把测试样本编码成二进制码。

DSH(deep supervised hashing): 通过分析数据的几何结构,选择与原始数据分布最相似的哈希函数,避免单纯地使用随机目标函数。

KLSH(kernelized locality sensitive hashing): 与 LSH 方法基本相似,不同之处在于将目标函数映射到核空间内。

LSH(locality sensitive hashing): 局部敏感哈希算法通过生成随机的线性函数来作为哈希函数。在本文实验中,参照文献[19]生成一个随机的高斯矩阵作为哈希函数。

MDSH(multidimensional spectral hashing): 多维谱聚类哈希首先通过学习原始数据的相似矩阵,保留数据的整体相似结构,然后通过在相似矩阵上优化矩阵分解,得到特征值最大的 k 个特征向量,把这些特征向量作为哈希函数来对数据进行哈希处理。

SGH(scalable graph hashing): 可扩展图哈希通过属性转换在不需要计算相似性图矩阵的情况下,可以高效地近似整个图的结构。

PCAH(PCA hashing): PCA 哈希通过主成分分析方法将原始高维数据转换为简短但信息量大的哈希编码。通过保留原始数据协方差矩阵中前 k 个最大特征值对应的特征向量,尽可能地保留原始数据的主要信息。

为保证算法公平性,所有算法都未对原始数据作任何处理。

3.2　评价指标

本文实验中采用如下三个评价指标来评价算法的性能:

a) 评价指标 topkham2。本文把汉明半径设为 2,对于一个给定的检索样本,输出前 k 个汉明距离小于 2 的原始样本。然后通过计算得到前 k 个原始样本的平均精确率作为 topkham2。

b) 评价指标 MAP。指的是检索得到的所有训练样本的平均精确率。非常明显,topkham2 和 MAP 指标的值越大,哈希算法的性能越好。

c) 评价指标 precision-recall。Precision 是指精确率,即已被检索样本的准确率[20]; recall 是指召回率,即正确的样本被检索到的概率。一般情况下,人们希望精确率和召回率都是越高越好,实际上两个指标是负相关关系,但是 precision-recall 曲线的高低和算法的准确率是正相关关系,所以可以使用 precision-recall 曲线来评估算法的性能。

3.3　实验结果和分析

所有哈希算法在三个数据集上的准确率—召回率(preci-

sion-recall) 曲线、不同比特对应的平均准确率(MAP) 、不同比特对应的汉明半径为 2 的平均准确率(topkham2) 如图 1 ~ 3 所示。通过图 1 ~ 3 可以明显地看出,本文提出的 PCAR 算法在三个数据集上的哈希效果在多个评价指标下均比其他六个对比算法要好,具体地,本文依据实验结果发现依赖数据的哈希算法(PCA 哈希、流形学习哈希、AGH) 以及本文提出的 PCAR 要比数据独立的哈希算法(LSH) 效果好很多。例如,与数据独立的哈希算法 LSH 相比,实验中涉及到的数据依赖的哈希算法准确率平均提高了 35.6%。也就意味着在目标哈希函数学习的过程中,考虑原始数据的结构分布是较为合理的[21]。实际上,本文将原始数据的结构分布作为先验知识,在构建哈希函数中同时结合了先验知识,而数据独立的哈希算法 LSH 没有考虑这点。研究结果表明,PCAR 算法比基于流形学习的哈希算法 AGH 平均高 13.7%,比基于 PCA 的哈希算法 MDSH 平均高 28.1%。鉴于 PCAR 算法同时保留数据的整体与局部的相似结构[22],与传统只保留一种相似结构的哈希算法相比,本文提出的算法可以取得更好的哈希效果。

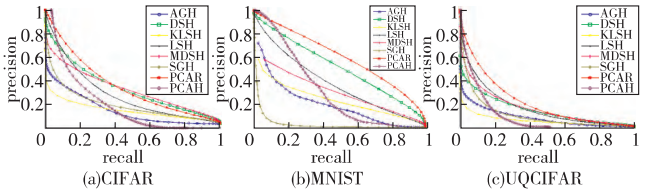

图 1 所有哈希算法在三个数据集上的准确率—召回率(precision-recall) 曲线

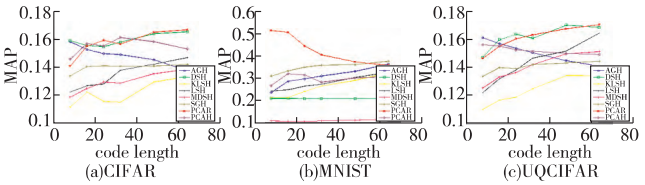

图 2 所有哈希算法在三个数据集上的不同比特对应的平均准确率(MAP)

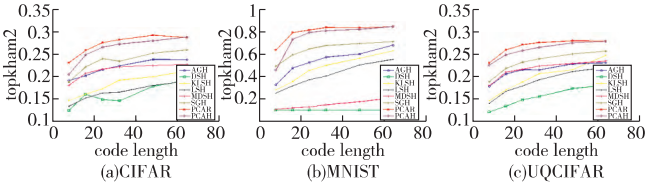

图 3 所有哈希算法在三个数据集上不同比特对应的汉明半径为 2 的平均准确率(topkham2)

本文提出的 PCAR 算法,由于首先依赖数据,其次同时保留了数据的整体与局部的相似结构,所以学习到的哈希函数性能更好。本文实验也明确地证明了这一点。

4 结束语

本文基于保留数据局部和整体的相似结构提出了一个高效的哈希算法—PCAR 算法: a) 同时结合 PCA 哈希和流形学习哈希求得最优的降维矩阵,从而得到原始数据的低维表征; b) 采用最小方差旋转模型同时学习到旋转矩阵和二值化阈值; c) 结合旋转矩阵和二值化阈值得到最终的哈希函数。本算法因为同时保留了原始数据的局部和整体的相似结构,在一定程度上改善了单一哈希算法在保留数据结构上的不足。经实验表明,本文提出的 PCAR 哈希算法比当前主流哈希算法在哈希效果上有较为明显的提高。在本文中,采用了基于 PCA 的哈希算法构造哈希函数,时间复杂度为 $O(n^2)$,导致哈希实验耗时较长。所以在今后的工作中,本文将进一步改进本文提出的框架去降低算法的时间复杂度。

参考文献:

[1] Zhu Xiaofeng, Huang Zi, Yang Yang, *et al*. Self-taught dimensionality reduction on the high-dimensional small-sized data [J]. Pattern Recognition, 2013, 46(1): 215-229.

[2] 贺玲, 吴玲达, 蔡益朝. 基于内容图像检索中的索引技术 [J]. 计算机应用研究, 2005, 22(11): 219-221,224.

[3] 张敏, 康志伟, 陈步真. 基于提升小波变换和 BP 神经网络的图像哈希算法 [J]. 计算机应用研究, 2010, 27(10): 3974-3976.

[4] Datar M, Immorlica N, Indyk P, *et al*. Locality-sensitive hashing scheme based on p-stable distributions [C] //Proc of the 20th Annual Symposium on Computational Geometry. New York: ACM Press, 2004: 253-262.

[5] Kulis B, Grauman K. Kernelized locality-sensitive hashing-for scalable image search [C] //Proc of the 12th IEEE International Conference on Computer Vision. Piscataway, NJ: IEEE Press, 2009: 2130-2137.

[6] Weiss Y, Fergus R, Torralba A. Multidimensional spectral hashing [C] //Proc of the 12th European Conference on Computer Vision. Berlin: Springer-Verlag, 2012: 340-353.

[7] Gong Yunchao, Lazebnik S, Gordo A, *et al*. Iterative quantization: a procrustean approach to learning binary codes for large-scale image retrieval [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2013, 35(12): 2916-2929.

[8] Liu Wei, Wang Jun, Kumar S, *et al*. Hashing with graphs [C] // Proc of the 28th International Conference on Machine Learning. 2011: 1-8.

[9] Zhu Xiaofeng, Huang Zi, Hong Cheng, *et al*. Sparse hashing for fast multimedia search [J]. ACM Trans on Information Systems, 2013, 31(2): 1-24.

[10] Weiss Y, Torralba A, Fergus R. Spectral hashing [C] //Advances in Conference on Neural Information Processing Systems. 2008: 1753-1760.

[11] Zhang Shichao, Li Xuelong, Zong Ming, *et al*. Efficient KNN classification with different numbers of nearest neighbors [J]. IEEE Trans on Neural Networks and Learning Systems, 2017, 25(9): 1774-1785.

[12] Zhu Xiaofeng, Zhang Shichao, Zhang Jilian, *et al*. Cost-sensitive imputing missing values with ordering [C] //Proc of the 22nd AAAI Conference on Artificial Intelligence. 2007: 1922-1923.

[13] Qin Yongsong, Zhang Shichao, Zhu Xiaofeng, *et al*. Semi-parametric optimization for missing data imputation [J]. Applied Intelligence, 2007, 27(1): 79-88.

[14] Zhu Xiaofeng, Zhang Shichao, Jin Zhi, *et al*. Missing value estimation for mixed-attribute data sets [J]. IEEE Trans on Knowledge and Data Engineering, 2011, 23(1): 110-121.

[15] Zhu Xiaofeng, Zhang Lei, Huang Zi. A sparse embedding and least variance encoding approach to hashing [J]. IEEE Trans on Image Processing, 2014, 23(9): 37-50.

[16] Zhang Shichao, Jin Zhi, Zhu Xiaofeng. Missing data imputation by utilizing information within incomplete instances [J]. Journal of Systems and Software, 2011, 84(3): 452-459.

[17] Wu Xindong, Zhang Chengqi, Zhang Shichao. Efficient mining of both positive and negative association rules [J]. ACM Trans on Information Systems, 2004, 22(3): 381-405.

[18] Zhu Xiaofeng, Xie Qing, Zhu Yonghua, *et al*. Multi-view multi-sparsity kernel reconstruction for multi-class image classification [J]. Neurocomputing, 2015, 169(10): 43-49.

[19] Zhang Chengqi, Qin Yongsong, Zhu Xiaofeng, *et al*. Clustering-based missing value imputation for data preprocessing [C] //Proc of the 4th IEEE International Conference on Industrial Informatics. Piscataway, NJ: IEEE Press, 2006: 1081-1086.

[20] Zhu Xiaofeng, Suk H I, Shen Dinggang. Matrix-similarity based loss function and feature selection for Alzheimer's disease diagnosis [C] // Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2014: 3089-3096.

[21] Zhu Xiaofeng, He Wei, Li Yonggang, *et al*. One-step spectral clustering via dynamically learning affinity matrix and subspace [C] //Proc of the 21st AAAI Conference on Artificial Intelligence. 2017: 2963-2969.

[22] Zhu Xiaofeng, Huang Zi, Shen Hengtao, *et al*. Dimensionality reduction by mixed kernel canonical correlation analysis [J]. Pattern Recognition, 2012, 45(8): 3003-3016.