doi:10.6041/j.issn.1000-1298.2019.07.022

基于 Multi-probe LSH 的菊花花型相似性计算

袁培森[1]　翟肇裕[2]　钱淑韵[1]　徐焕良[1,3]

(1.南京农业大学信息科学技术学院,南京 210095;2.马德里理工大学技术工程和电信系统高级学院,马德里 28040;
3.国家信息农业工程技术中心,南京 210095)

摘要:针对海量高维菊花图像相似性计算带来的挑战,研究了基于多探测局部位置敏感哈希技术的菊花表型相似性计算方法。针对菊花图像,采用 SIFT 技术提取菊花图像特征,并采用 BoVW 模型进行建模。由于图像特征的高维性质,海量的菊花表型相似性计算效率不高,为了提高计算效率,提出采用近似相似性技术中的多探测局部位置敏感哈希技术,用此方法构建菊花图像数据的哈希数据结构,在菊花相似性查询方面提高了计算效率,并确保了计算结果的质量。在菊花数据集上进行了计算效率和查询质量两方面的测试,并与典型的方法进行了试验对比和分析。结果表明,相比线性式扫描,平均查询成功概率达到 0.90 以上,平均加速比为 3.3 ~ 19.8。本文方法能够在查询质量和计算效率两方面通过参数设置提供灵活的优化选择,并对参数的选择提供了参考范围,可为海量菊花花型相似性计算提供参考。

关键词:菊花;花型;花型相似性;多探测;局部位置敏感哈希

中图分类号:TP391　　**文献标识码**:A　　**文章编号**:1000-1298(2019)07-0208-08

Chrysanthemum Petal Similarity Evaluation Based on Multi-probe Locality Sensitive Hashing

YUAN Peisen[1]　ZHAI Zhaoyu[2]　QIAN Shuyun[1]　XU Huanliang[1,3]

(1. *College of Information Science and Technology*, *Nanjing Agricultural University*, *Nanjing* 210095, *China*
2. *Superior School of Technical Engineering and Telecommunication Systems*, *Technical University of Madrid*, *Madrid* 28040, *Spain*
3. *National Engineering and Technology Center for Agriculture*, *Nanjing* 210095, *China*)

Abstract: Plant phenotyping is an important research topic in the field of botany. The similarity of plant phenotypes is widely used in plant taxonomy, ecology and digital agriculture etc. It is one of the important contents of plant phenotype research. Chrysanthemum is an important plant in China as well as in the world, and the phenotype similarity evaluation of chrysanthemum plays an important role in chrysanthemum classification and phenotypic research. The feature of high-dimension of massive chrysanthemum data brings great challenge for chrysanthemum phenotype analysis, from this point of view, the chrysanthemum phenotypic similarity query and evaluation were studied based on multi-probe locality sensitive hashing technique. For evaluating the similarity of chrysanthemum image, the SIFT features of the chrysanthemum images were extracted and clustered based on the K-means method. Hereafter, the bag of visual words (BoVW) model was built. Due to the high-dimensional nature of the image features, especially for the massive chrysanthemum images, the computing efficiency of the query was a big challenge for the high dimensional problem. The multi-probe locality sensitive hashing (LSH) was applied for chrysanthemum phenotype similarity computing. The multi-probe locality sensitive hashing technique was an optimization technique for high-dimensional data similarity query. By means of the technique, a hash data structure of chrysanthemum image data was constructed, which improved query efficiency in chrysanthemum similarity query and ensured the query result quality. The theory of the multi-probe locality sensitive hashing was analyzed, in addition to this, extensive experiments were conducted and important results were gained as well. Experiments showed that compared with linear scanning, the average success probability of the query can reach above 0.90, and the average

收稿日期:2019-01-06　修回日期:2019-03-13

基金项目:国家自然科学基金项目(61502236)和中央高校基本科研业务费专项资金项目(KYZ201752、KJQN201651)

作者简介:袁培森(1980—),男,讲师,博士,主要从事智能信息处理、海量数据处理与分析研究,E-mail:peiseny@njau.edu.cn

通信作者:徐焕良(1963—),男,教授,博士,主要从事农业信息化与大数据技术研究,E-mail:huanliangxu@njau.edu.cn

acceleration ratio was 3.3 ~ 19.8, furthermore, it was also compared with the typical method in the aspects of query quality and query efficiency, and the results demonstrated that the method was better than the entropy based LSH in quality and performance. The experimental results revealed that the query quality and query efficiency could be tuned flexibly through the parameter settings of hash function number and the hash tables, which provided an elastic way for the choice for tuning the quality and efficiency. In addition, it can provide technical reference for massive chrysanthemum phenotypic similarity calculation.

Key words: chrysanthemum; phenotyping; petal similarity; multi-probe; locality sensitive hashing

0　引言

表型(Phenotype)是生物某一特定物理外观或组成,是由基因和环境共同作用的结果[1-2]。植物表型代表了植物形态,是植物学领域研究的重要课题,在生态学、植物育种等领域应用广泛[3-4]。其中,表型相似性计算在植物表型[4-6]、基因组学[7]、疾病诊断[8-10]等研究中具有重要应用。

当前,数据挖掘技术为解决植物表型研究提供了重要支撑手段[11]。相似性计算作为数据挖掘和机器学习的基础问题,又称为最近邻查询。对于海量、高维数据的相似性计算,由于“维灾难”(Curse of dimensionality)问题,线性计算效率不高。为此,研究者提出了近似最近邻(Approximate nearest neighbor, ANN)查找技术[12],这是一种对查询结果质量和效率折中的技术。局部位置敏感哈希技术(Locality sensitive hashing, LSH)是近似最近邻查询计算的高效方法[13]。该方法在查询质量和查询效率方面均具有理论保证。基于 LSH 有多种优化查询质量和效率的技术,例如,基于熵的 LSH[14],多探测的 LSH(Multi-probe LSH, MPLSH)[15]等,其中 MPLSH 在高维数据相似性计算效率和结果质量方面具有更好的效果[16-17]。

在表型相似性计算研究方面,PENG 等[6]指出,测量表型相似性在疾病诊断中发挥重要作用,提出了一种基于网络的表型相似性测量方法,用于计算表型之间的相似性。在菊花表型及分类方面,ZHANG 等[18]通过简单序列重复标记建立的独特 DNA 指纹和分子身份鉴定中国传统菊花品种,为基于微卫星标记多态性的菊花鉴定和分类的起点。ROEIN 等[19]研究利用扩增片段长度多态性和表型特征,评估菊花的遗传多样性和种群结构。KHODAKOVSKAYA 等[20]通过基因片段的控制来增强菊花中的开花和分枝表型。但是,专门针对菊花表型的计算研究较少。YAN 等[21]定义了菊花的不同花色表型,完成了菊花品种表型颜色的分布分析。

菊花是全球第二重要的观赏植物,具有品种数量庞大、花型变异丰富的特点[22]。菊花花型是菊花的重要表型特征之一,其相似性计算对菊花形态分类和表型研究具有重要作用,同时菊花花型种类多、瓣形繁多,这些特点给菊花的品种分类和表型研究带来较大困难[18]。表型相似性计算可以为菊花分类提供重要的参考,对于海量的菊花相似性计算,效率和质量至关重要。

本文针对海量菊花图像数据进行菊花花型相似性比较,提出采用 SIFT 特征和视觉词袋模型[23-24]提取菊花图像重要特征,并使用 K-means 对重要特征进行聚类和优化。针对菊花图像数据的高维特性,采用多探测 LSH 解决海量菊花花型相似性计算的质量和效率问题。在查询质量和查询效率方面寻求一种优化计算方案。

1　材料与方法

1.1　菊花图像

本文数据集共包括 4 100 幅菊花图像,共 5 种花型:翻卷型、雏菊型、飞舞型、球型和莲座型[25]。菊花图像示例如图 1 所示。每一种花型有 800 幅,共 4 000 幅用于训练,100 幅图像用于测试。

(a) 翻卷型　(b) 雏菊型　(c) 飞舞型　(d) 球型　(e) 莲座型

图 1　5 种菊花类型示例

Fig. 1　Illustration of five types of chrysanthemum images

1.2　菊花图像 SIFT 特征

常用的图像特征包括形状、纹理等。在诸多的图像特征提取技术中,尺度不变特征转换(Scale invariant feature transform, SIFT)[26]应用最广。SIFT 通过在图像空间中搜索关键点,并提取出其位置、尺度、旋转不变量。因此 SIFT 对尺度变化、旋转以及一定视角和光照变化等具有不变性、稳定性及很强的可区分性和扩展性[26]。鉴于 SIFT 良好的性质,本文采用 SIFT 方法提取菊花的特征。

SIFT 算法有 4 个计算步骤[26]:高斯差(Difference of Gaussians, DoG)空间极值检测、关键点定位、方向确定和关键点描述。

图 2 是一幅菊花图像经过灰化后提取 SIFT 特

征示意图,每个点代表一个定位关键点。

(a) 原图 (b) SIFT特征提取图

图 2 菊花图像 SIFT 特征提取示意图

Fig. 2 Illustration of SIFT features extraction of chrysanthemum image

1.3 BoVW - SIFT 模型

由于 SIFT 特征提取之后的高维性,本文采用 K-means 聚类算法对这些向量数据进行初步聚类,用聚类中的簇作为词典的词,进而将同一幅图像的 SIFT 向量映射到视觉词序列生成码本,这样菊花图像可以使用一个码本向量来进行描述。本文采用了基于 SIFT 的视觉词袋(Bag of visual word, BoVW)[23-24]特征来对菊花图像建模,菊花图像预处理过程如图 3 所示。

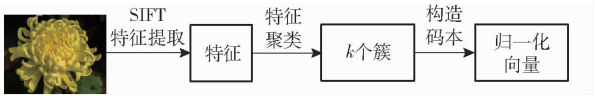

图 3 菊花图像预处理过程

Fig. 3 Illustration of chrysanthemum image preprocessing

针对 SIFT 特征,构建 BoVW 码本步骤如下:①对每一幅菊花图像提取 SIFT 特征,SIFT 特征用一个 128 维描述子向量表示,假设菊花图像数据集共提取出 M 个 SIFT 特征。②用 K-means 对提取的 M 个 SIFT 特征进行聚类,此算法把 M 个 SIFT 特征分为 k 个簇,表示为 $\boldsymbol{C}_i$, $i=1,2,\cdots,k$,其中 $\boldsymbol{C}_i$ 为聚类中心。

$$E=\sum_{i=1}^{k}\sum_{\boldsymbol{x}\in\boldsymbol{C}_i}\|\boldsymbol{x}-\boldsymbol{\mu}_i\|^2 \tag{1}$$

其中 $$\boldsymbol{\mu}_i=\frac{1}{|\boldsymbol{C}_i|}\sum_{\boldsymbol{x}\in\boldsymbol{C}_i}\boldsymbol{x}$$

式中 $\boldsymbol{\mu}_i$——簇 $\boldsymbol{C}_i$ 的均值向量

E——平方误差

$\boldsymbol{x}$——由 SIFT 特征构成的向量

最小化式(1)的平方误差,以使簇内具有较高的相似度,而簇间相似度较低[27]。

本文使用 K-means 聚类返回的 k 个簇,计算每一幅图像的每一个 SIFT 特征到这 k 个视觉词的距离,并将其映射到距离最近的视觉词中。研究表明,k 太小会导致图像表达能力不足,太大导致量化过拟合[28],本文设置 k 为 512。

菊花图像数据经过提取 SIFT 特征后,将这些特征映射到码本向量。利用 TF - IDF 模型构造向量,并对向量进行归一化。菊花向量的构造包含两项因子:词频(Term frequency, TF)和逆向文件频率(Inverse document frequency, IDF)[29]。TF 是给定的关键点在该菊花图像中出现的频率,IDF 是一个关键点普遍重要性的度量。

$$T_{F_{ij}}=\frac{n_{ij}}{\sum_k n_{kj}} \tag{2}$$

式中 $T_{F_{ij}}$——关键点 j 在图像 i 中的频率归一化值

n_{ij}——关键点 j 在图像 i 中出现的次数

n_{kj}——图像 k 中关键点 j 出现的次数

$$I_{DF_i}=\lg\frac{|D|}{|T_i|} \tag{3}$$

其中 $$T_i=\{j:t_i\in d_j\}$$

式中 I_{DF_i}——关键点 i 在图像中的逆向文件频率

D——菊花图像数量

T_i——包含关键点 j 的图像数量

图像向量采用 $T_{F_{ij}}I_{DF_i}$ 进行计算,最后对图像向量进行数据归一化

$$\bar{\boldsymbol{x}}=\frac{\boldsymbol{x}-\boldsymbol{\mu}}{\sigma} \tag{4}$$

式中 $\boldsymbol{\mu}$——均值 σ——标准差

$\bar{\boldsymbol{x}}$——向量归一化结果

采用欧氏距离作为图像距离度量。距离最小的菊花图像作为相似性查询结果。

$$d=\sum_{i=1}^{N}\sqrt{(\boldsymbol{v}_i-\boldsymbol{u}_i)^2} \tag{5}$$

式中 $\boldsymbol{u}_i$、$\boldsymbol{v}_i$——归一化后的向量

N——向量维度 d——距离

1.4 局部位置敏感哈希技术

1.4.1 LSH 基本原理

相似性查询又称为最近邻查询(Nearest neighbor search, NNS),给定 N 维向量 $\boldsymbol{O}=\{\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_n\}\in\mathbf{R}^N$,查询向量 $\boldsymbol{q}\notin\boldsymbol{O}$,返回元素 $\boldsymbol{x}^*\in\boldsymbol{O}$,$\boldsymbol{x}^*$ 与查询向量 $\boldsymbol{q}$ 在某种距离度量上最近。为解决高维数据最近邻查询计算效率问题,研究者提出了近似最近邻查询技术来折衷查询的效率和质量[12]。近似最近邻查询返回满足 $d(\boldsymbol{x},\boldsymbol{q})\leqslant(1+\varepsilon)d(\boldsymbol{x}^*,\boldsymbol{q})$ 的结果,其中 $\varepsilon>0$,$\boldsymbol{x}^*$ 是查询向量 $\boldsymbol{q}$ 的精确最近邻,该查询称为 $(1+\varepsilon)$-近似最近邻[30]。图 4 是 $(1+\varepsilon)$-近似最近邻查询示意图,查询向量 $\boldsymbol{q}$ 真实的最近邻是 o_1,距离为 r,该图中的 o_2 和 o_4 都可作为该查询的近似结果。

LSH[12]是一种基于过滤-验证框架的计算高维数据近似最近邻的高效查询技术,过滤-验证框架计算过程如图 5 所示。LSH 能够理论上在次线性

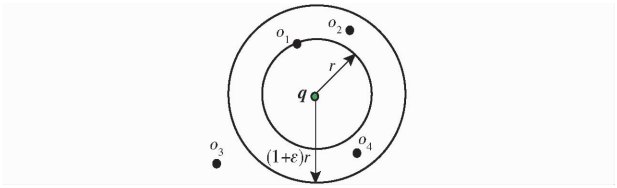

图 4　$(1+\varepsilon)$-近似最近邻查询示意图

Fig. 4　Illustration of $(1+\varepsilon)$-approximate nearest neighbor query

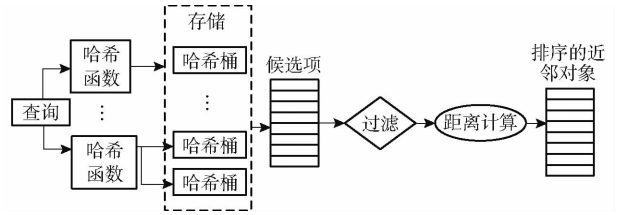

图 5　LSH 过滤-验证计算过程

Fig. 5　LSH filtering-verification evaluation procedure

(Sub-linear)时间内近似求解高维数据的最近邻问题。

给定距离 r、近似因子 $\varepsilon(\varepsilon>0)$和概率 p_1、p_2，其中 $p_1>p_2$，哈希函数 h 把 N 维空间的向量映射为整数集合 $\mathbf{Z}$，记为 $h:\mathbf{R}^N\to\mathbf{Z}$，$\boldsymbol{x}_1$、$\boldsymbol{x}_2$ 同时满足：若 $d(\boldsymbol{x}_1,\boldsymbol{x}_2)\leqslant r$，则 $P[h(\boldsymbol{x}_1)=h(\boldsymbol{x}_2)]\geqslant p_1$；若 $d(\boldsymbol{x}_1,\boldsymbol{x}_2)\geqslant(1+\varepsilon)r$，则 $P[h(\boldsymbol{x}_1)=h(\boldsymbol{x}_2)]\leqslant p_2$，则称为 $(r,(1+\varepsilon)r,p_1,p_2)$-sensitive 的 LSH。

LSH 的基本原理为

$$P[h(\boldsymbol{x}_i)=h(\boldsymbol{x}_j)]=\mathrm{sim}(\boldsymbol{x}_i,\boldsymbol{x}_j)\tag{6}$$

式中　sim()——相似性度量函数

$\boldsymbol{x}_i$、$\boldsymbol{x}_j$——数据向量

P——概率

即把两个高维向量的相似性计算转变为计算两个哈希值相等的概率。

1.4.2　LSH 结果放大

为了提升查询成功概率，通常使用多个独立的哈希表[31]。设局部敏感哈希函数族 H，每个哈希表由 k 个哈希函数 $h\in H$ 创建。

通过以下 3 种操作来提高成功概率：AND 操作；OR 操作；级联 AND 和 OR 操作。

AND 操作从函数族 H 中随机选择 t 个哈希函数，构造 $g_i(\boldsymbol{x})=(h_1(\boldsymbol{x}),h_2(\boldsymbol{x}),\cdots,h_t(\boldsymbol{x}))$，因此当且仅当 $h_i(\boldsymbol{x})=h_i(\boldsymbol{y})$，$i=1,2,\cdots,t$ 时，$g(\boldsymbol{x})=g(\boldsymbol{y})$。AND 操作使得两个数据的 t 个哈希值都相同时，才会被投影到相同的桶内。该操作构造一个新的哈希函数族 H'，H'为(d_1,d_2,p_1^t,p_2^t)-sensitive。

OR 操作构造 L 个哈希表，使得落在任何哈希表的结果作为候选结果。OR 操作构造哈希函数族 H''，H'' 为 $(d_1,d_2,(1-p_1)^L,(1-p_2)^L)$-sensitive。AND 和 OR 级联操作是先通过 AND 操作再进行 OR 操作，得到一个新的哈希函数族 $\overline{H}$，$\overline{H}$ 为 $(d_1,d_2,(1-p_1^t)^L,(1-p_2^t)^L)$-sensitive。经过 AND 和 OR 级联操作，相似度为 s 的菊花，可以被检索出来的概率为

$$P=1-(1-s^t)^L\tag{7}$$

式(7)的可视化如图 6 所示，图 6 是 L 为 10 时，被检出的概率与 t、s 的关系，AND 操作时，增大 t 能提高过滤掉相似度小的对象的概率。通过连接 t 个哈希函数构造哈希关键字，降低了相似对象之间的冲突概率。为了提高查询的召回率，LSH 一般使用 L 个函数 $g_1,g_2,\cdots,g_L$，共 L 个哈希表。

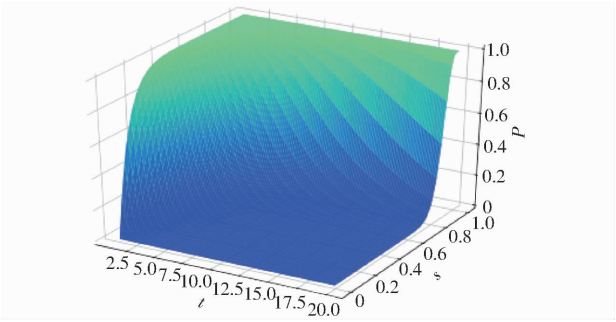

图 6　公式(7)的成功概率与相似度及参数($L=10$)

Fig. 6　Success ratio with similarity s and parameter t of formula (7) ($L=10$)

1.5　查询计算过程

基于 LSH 的查询计算处理框架如图 7 所示。给定一个菊花查询 $\boldsymbol{x}$，计算 $g_i(\boldsymbol{x})(1<i<L)$，把相同的桶中数据作为查询的候选对象。给定哈希函数族 H，LSH 独立地选取函数族中 t 个哈希函数，通过连接运算形成一个组合的哈希关键字。

$$g(\boldsymbol{x})=(h_1(\boldsymbol{x}),h_2(\boldsymbol{x}),\cdots,h_t(\boldsymbol{x}))\tag{8}$$

函数 $g(\boldsymbol{x})$ 通过连接 t 个哈希函数构造哈希关键字，该关键字作为索引桶识别号，降低了相似对象之间的冲突概率。为了提高查询的召回率，LSH 一般使用 L 个函数 $g_1,g_2,\cdots,g_L$，形成 L 个哈希表。

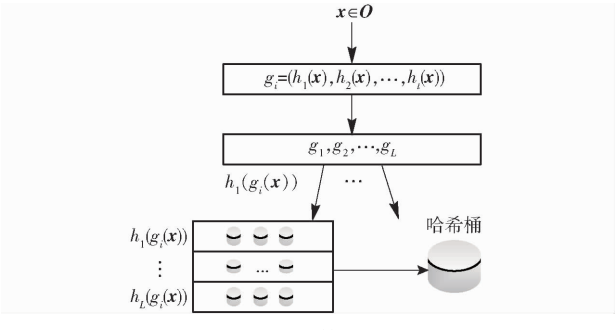

图 7　LSH 计算处理框架

Fig. 7　Basic processing framework of locality sensitive hashing

根据 LSH 的性质，与查询向量 $\boldsymbol{q}$ 相近的菊花没有被哈希到同一个桶中，它很有可能以很高的概率哈希到周围的桶中，查询向量 $\boldsymbol{q}$ 的近邻点落在其相

邻区域的概率分布如图 8 所示[15]。图 8 是某一查询向量 $\boldsymbol{q}$ 在菊花图像集上的相邻区域映射概率分布,该图以 0 为中心,落在 $x=0$ 两侧的概率基本呈现正态分布。由图 8 可以看出通过检查查询向量 $\boldsymbol{q}$ 附近的桶,可以增加查找近邻数据的概率。本文方法通过仔细推导出的探测序列,探测可能包含查询结果的哈希桶来提高查询的效率,极大地降低了哈希表所需的数量,提高菊花花型相似性的查询质量和查询效率。

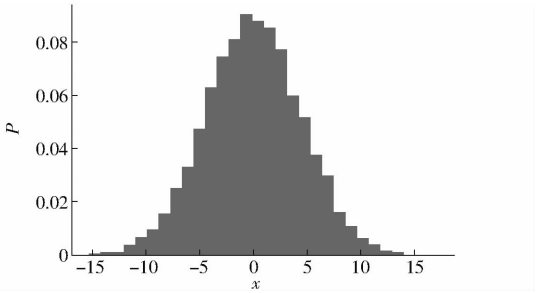

图 8 查询向量 $\boldsymbol{q}$ 的近邻点落在其相邻区域的概率分布图

Fig. 8 Illustration of probability distribution of query points $\boldsymbol{q}$ in adjacent buckets

采用文献[15]的方法,定义哈希扰动向量 $\boldsymbol{\Delta}=(\delta_1,\delta_2,\cdots,\delta_T)$,其中 $\delta_i\in\{-1,0,1\}$。给定菊花查询向量 $\boldsymbol{q}$,基本的 LSH 检索 $g(\boldsymbol{q})=(h_1(\boldsymbol{q}),h_2(\boldsymbol{q}),\cdots,h_L(\boldsymbol{q}))$ 的桶中元素作为候选对象,顺序地探测哈希桶 $g(\boldsymbol{q})+\boldsymbol{\Delta}$,探测过程如图 9 所示。给定哈希桶探寻序列 $\boldsymbol{\Delta}_1,\boldsymbol{\Delta}_2,\cdots,g_i(\boldsymbol{q})+\boldsymbol{\Delta}_1$ 是应用扰动向量 $\boldsymbol{\Delta}_1$ 后产生的新哈希值,它指向表中一个新的哈希桶,通过使用扰动向量,可以获得多个与查询向量 $\boldsymbol{q}$ 指向的哈希桶邻近的桶,这些桶中含有与 $\boldsymbol{q}$ 邻近的元素的概率较高。

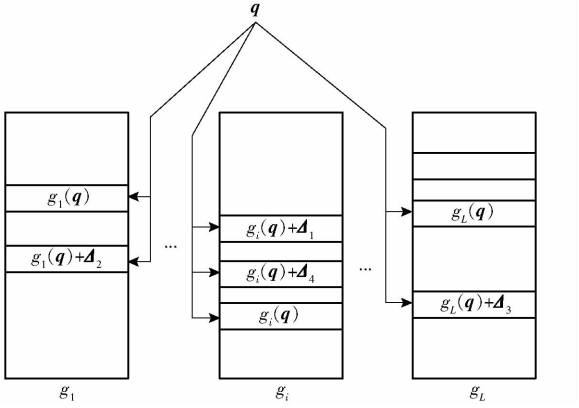

图 9 哈希探测过程

Fig. 9 Probing procedure of hash

图 9 中点 $\boldsymbol{p}$ 落在 $h_i(\boldsymbol{q})+\delta$ 的概率[15]估计为

$$P[h_i(\boldsymbol{p})=h_i(\boldsymbol{q})+\delta]\approx\exp(-\eta x_i(\delta)^2) \tag{9}$$

式中 η——取决于 $\|\boldsymbol{p}-\boldsymbol{q}\|^2$ 的常量

$x_i(\delta)$——查询向量 $\boldsymbol{q}$ 与横边界 $h_i(\boldsymbol{q})+\delta$ 的距离

假设扰动向量 $\boldsymbol{\Delta}=(\delta_1,\delta_2,\cdots,\delta_T)$,则经过 t 个哈希函数,得到 $\boldsymbol{p}$ 与 $\boldsymbol{q}$ 相邻的概率为

$$P[g(\boldsymbol{p})=g(\boldsymbol{q})+\Delta]\approx\prod_{i=1}^{t}\exp(-\eta x_i(\delta_i)^2)=\exp\left(-\eta\sum_i x_i(\delta_i)^2\right) \tag{10}$$

2 结果

2.1 试验环境

试验运行环境为 Intel Core i5 1.6 GHz,8 GB 内存,128 GB 固态硬盘;操作系统为 MacOS 10.13.3。开发环境为 OpenCV 3.4 和 C++。

图像码本长度 k 为512,ε 设置为 0.02。查询次数设置为 100,哈希探测次数默认设置为 2 000。试验中参数 t 为每个哈希表的哈希函数个数,L 为哈希表个数。试验中参数 t 的取值范围为 14 ~ 18,L 的范围为 3 ~ 18。探测数量 T 分别为 500、1 000、2 000和 3 000。

2.2 查询计算效率

为了测试计算效率,对数据集采用了增加随机高斯噪声进行扩充,使得数据集的总量为 100 000,查询质量的测试仍然采用原始数据集。查询计算效率使用平均查询时间和加速比两个指标进行评价。平均查询时间结果如表 1 所示。其中线性式扫描的平均查询时间为 12.31 ms。

表 1 平均查询时间

Tab. 1 Average query time ms

L	t				
	14	15	16	17	18
3	3.640	2.102	1.266	0.835	0.596
6	3.654	2.164	1.294	0.831	0.622
9	3.690	2.146	1.329	0.887	0.629
12	3.742	2.254	1.310	0.867	0.590
15	3.823	2.156	1.354	0.889	0.665
18	3.735	2.249	1.365	0.905	0.627

表 1 中,随着参数 t 和 L 的增加,平均查询时间在减小,从 3.823 ms 降低到 0.590 ms。每个哈希表的哈希函数越多,其查询时间越短,表明 AND 操作过滤后的结果越少,表 1 结果表明哈希表个数对查询时间影响不大。

根据加速比评价查询效率,加速比是相对于线性扫描式查找的比例因子,计算公式为

$$S=\frac{T_L}{T_A} \tag{11}$$

式中 T_A——近似查询时间

T_L——线性扫描时间

加速比越大,说明本文方法相对于线性扫描式

计算越快。

图 10 显示随着参数 t 和 L 的增加，平均加速比从 3.3 左右增加到 19.8。哈希函数数量越多，加速比越大，哈希表个数对查询时间影响不明显。

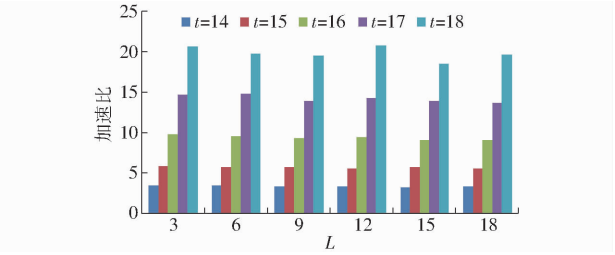

图 10　相对线性扫描加速比

Fig. 10　Speedup ratio compared with linear scanning

2.3　查询结果及质量

查询结果示例如图 11 所示，输入一幅飞舞型菊花图像，右侧输出查询到的相似图像。

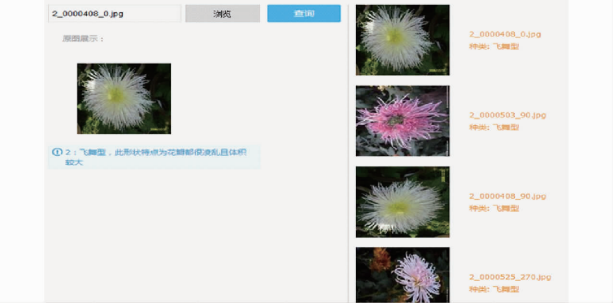

图 11　查询结果示例

Fig. 11　Illustration of query results

为了度量查询结果质量，采用平均成功概率来计算。

对于一个查询集合 Q 和查询点 $\boldsymbol{q}_i \in Q$，$\boldsymbol{x}^* = \{\boldsymbol{x} \mid \min_{\boldsymbol{x} \in \boldsymbol{O}} d(\boldsymbol{q}_i, \boldsymbol{x})\}$。设 $\boldsymbol{O}'$ 为查询在 $\min_{\boldsymbol{x} \in \boldsymbol{O}} d(\boldsymbol{q}_i, \boldsymbol{x})(1+\varepsilon)$ 内的所有数据对象，即 $\boldsymbol{O}' = \{\boldsymbol{x} \mid \boldsymbol{x} \in \boldsymbol{O}, d(\boldsymbol{q}_i, \boldsymbol{x}) \leqslant (1+\varepsilon) \min_{\boldsymbol{x}^* \in \boldsymbol{O}} d(\boldsymbol{q}_i, \boldsymbol{x}^*)\}$。查询点 $\boldsymbol{q}_i$ 的查询成功率 s_i 为

$$s_i = \begin{cases} 1 & (\boldsymbol{x}^* \in \boldsymbol{O}') \\ 0 & (\text{其他}) \end{cases} \tag{12}$$

对查询集合 Q，平均成功概率（Average success ratio，ASR）为

$$A_{\mathrm{SR}} = \frac{\sum_{i=1}^{|Q|} s_i}{|Q|} \tag{13}$$

式中　$|Q|$——查询数量

A_{SR} 值越大，表明查询结果越好。

图 12 为平均成功概率试验结果，由图 12 可知，t 为 14、15 时，平均成功概率在 0.91 以上，L 大于 6、t 为 14 时平均成功概率大于 0.95，查询结果的质量较好；t 从 15 增加到 18 时，平均成功概率从 0.95 降到 0.63 左右；t 一定时，L 从 3 增加到 18，平均成功概率呈上升趋势，增加了 11.9% ~19%。

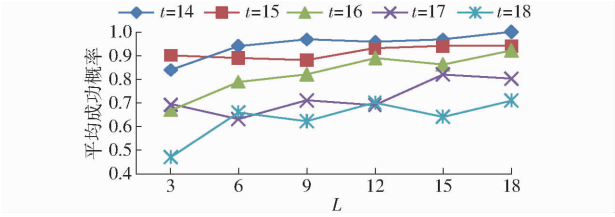

图 12　查询平均成功概率结果

Fig. 12　Average success rate of query

从图 10、12 可以得出，t 为 14 ~16、$L>6$ 时，平均加速比从 3.3 上升到 9.3，此时平均查询时间为 1.31 ms，平均成功概率在 0.8 以上。

2.4　探测次数的影响

在 $t=14$、$L=6$ 时，测试了探测次数对查询质量的影响，如图 13 和图 14 所示。探测次数设置为 500、1 000、2 000 和 3 000。

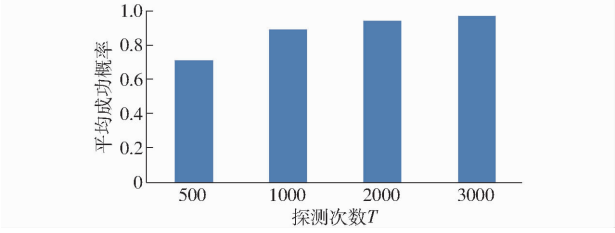

图 13　查询平均成功概率与探测次数关系

Fig. 13　Relationship of average success rate with probe number

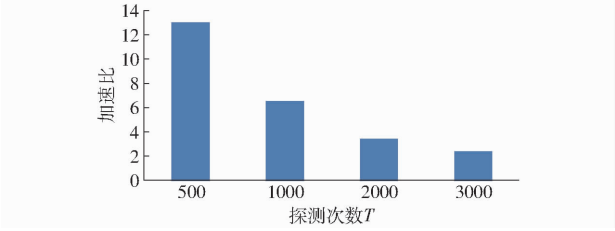

图 14　加速比与探测次数关系

Fig. 14　Relationship of speedup ratio with probe number

图 13 中，查询平均成功概率随着探测次数的增加呈现上升趋势，从 0.71 提高到 0.97，提高了 36.6%。

图 14 中，加速比随着探测次数的增加呈下降趋势，从 13 降到 2.4。

图 13、14 的结果表明，探测次数对查询性能有较大的影响，探测次数的增加可以提高查询结果的平均成功概率。探测次数在 1 000 ~2 000 范围可以获得较好的查询质量，但是如果要提高查询的效率，可以适当地减少探测次数。

2.5　试验对比

在查询质量和性能方面，设置哈希函数个数 t 为 14，将本文方法与基于熵的 ELSH[14] 进行了对比试验。试验结果如图 15、16 所示。图 15 结果表明，随着参数 L 的增加，两种方法的平均成功概率都在增加，而基于熵的 ELSH 从 0.82 缓慢增加到 0.92

左右,本文方法从 0.88 增加到 0.94 左右,表明本文方法优于基于熵的 ELSH。

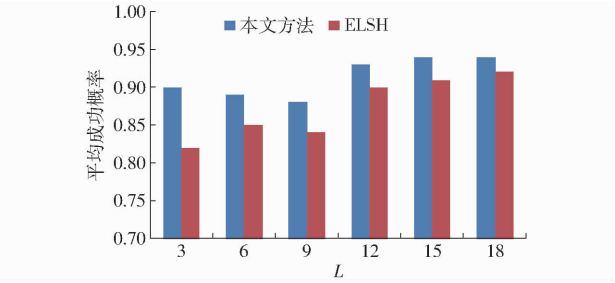

图 15 查询平均成功概率对比

Fig. 15 Average success rate comparison

图 16 表明,在查询性能方面,随着参数增加,本文方法的查询时间在 2.1 ms 左右,比较稳定,而 ELSH 的平均查询时间是本文方法的两倍左右。以上试验结果表明,本文方法具有较好的平均查询成功概率和查询性能。

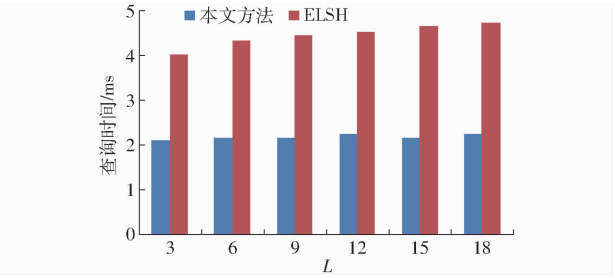

图 16 查询时间对比

Fig. 16 Query time comparison

3 讨论

本文使用多探测局部位置敏感哈希技术,对菊花表型相似性进行了初步研究。植物的表型重要表现是外观,其相似性研究对菊花形态的变化、不同菊花品种之间的演变等具有重要作用。通过提取菊花的 SIFT 特征并对特征进行 K-means 聚类,构建了菊花图像的 BoVW - SIFT 模型。鉴于高维性质对菊花图像计算性能的影响,提出采用近似查询方案,其中基于多探测的局部位置敏感哈希针对 LSH 特点进一步查询优化,在提升查询效率和查询质量方面提供理论保证。

系统中多探测 LSH 通过哈希表个数 L、哈希函数个数 t 和探测数量 T 这 3 个参数来灵活地调整质量和查询效率。为了寻找最佳参数设置,首先根据内存容量选择哈希表个数 L,然后在哈希函数个数 t 和探测数量 T 之间进行权衡。结果表明,t 为 14 ~ 16、L 为 6 ~ 12、探测次数在 1 000 ~ 2 000 范围内可以取得较好的结果。

4 结束语

为了提升菊花表型相似性计算的质量和效率,采用多探测局部位置敏感哈希技术,提取菊花的 SIFT 特征,并对特征进行 K-means 聚类,构建了菊花图像的 BoVW - SIFT 模型。在菊花数据集上进行了系统性试验测试,结果表明,查询平均成功概率可以达到 0.90 以上,系统的平均加速比在 3.3 ~ 19.8 之间。通过试验测试得出了相关参数优化选择范围。与基于熵的 ELSH 近似查询方法相比,本文方法在查询质量和效率方面具有明显的优势。

参 考 文 献

[1] 杨有新, 杨泽茂, 吴才君, 等. 植物表型组学研究进展[J]. 江西农业大学学报, 2015, 37(6): 1105 - 1112.
YANG Youxin, YANG Zemao, WU Caijun, et al. Advances in plant phenomics research[J]. Acta Agriculturae Universitatis Jiangxiensis, 2015, 37(6): 1105 - 1112. (in Chinese)

[2] 周济, FRANCOIS T, TONY P, 等. 植物表型组学:发展、现状与挑战[J]. 南京农业大学学报, 2018, 41(4): 580 - 588.
ZHOU Ji, FRANCOIS T, TONY P, et al. Plant phenomics: history, present status and challenges[J]. Journal of Nanjing Agricultural University, 2018, 41(4): 580 - 588. (in Chinese)

[3] FIORANI F, SCHURR U. Future scenarios for plant phenotyping[J]. Annual Review of Plant Biology, 2013, 64(1): 267 - 291.

[4] KRAJEWSKI P, CHEN D, WIEK H, et al. Towards recommendations for metadata and data handling in plant phenotyping[J]. Journal of Experimental Botany, 2015, 66(18): 5417 - 5427.

[5] BERGAMO P J, WOLOWSKI M, MARUYAMA P K, et al. The potential indirect effects among plants via shared hummingbird pollinators are structured by phenotypic similarity[J]. Ecology, 2017, 98(7): 1849 - 1858.

[6] PENG J, HUI W, SHANG X. Measuring phenotype-phenotype similarity through the interactome[J]. BMC Bioinformatics, 2018, 19(Supp. 5): 114.

[7] GREENE D, RICHARDSON S, TURRO E, et al. Phenotype similarity regression for identifying the genetic determinants of rare diseases[J]. The American Journal of Human Genetics, 2016, 98(3): 490 - 499.

[8] BRZOZOWSKI M, KWIATKOWSKI P, KOSIK-BOGACKA D, et al. The application of genotyping and phenotyping techniques for epidemiological analysis of microorganisms[J]. Postepy Mikrobiologii, 2017, 56(3): 353 - 366.

[9] WANG J, ZHOU X, ZHU J, et al. Bias of phenotype similarity scores between diseases [C]// International Conference on Bioinformatics and Biomedical Engineering. IEEE, 2010: 1 - 4.

[10] SAKLATVALA J R, DAND N, SIMPSON M A. Text-mined phenotype annotation and vector-based similarity to improve identification of similar phenotypes and causative genes in monogenic disease patients[J]. Human Mutation, 2018, 39(11): 643 – 652.

[11] TSAFTARIS S A, MINERVINI M, SCHARR H. Machine learning for plant phenotyping needs image processing[J]. Trends in Plant Science, 2016, 21(12):989 – 991.

[12] HAR-PELED S, INDYK P, MOTWANI R. Approximate nearest neighbor: towards removing the curse of dimensionality[J]. Theory of Computing, 2012,8(1):321 – 350.

[13] SLANEY M, CASEY M. Locality-sensitive hashing for finding nearest neighbors[J]. IEEE Signal Processing Magazine, 2008, 25(2): 128 – 131.

[14] WANG Q, GUO Z, LIU G, et al. Entropy based locality sensitive hashing[C]// IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 2012:1045 – 1048.

[15] AUMÜLLER M, BERNHARDSSON E, FAITHFULL A. ANN-benchmarks: a benchmarking tool for approximate nearest neighbor algorithms[C]// International Conference on Similarity Search and Applications. Springer, 2017: 34 – 49.

[16] LV Q, JOSEPHSON W, WANG Z, et al. Multi-probe LSH: efficient indexing for high-dimensional similarity search[C]// International Conference on Very Large Data Bases. VLDB Endowment, 2007:950 – 961.

[17] ANDONI A, INDYK P, LAARHOVEN T, et al. Practical and optimal LSH for angular distance[C]// Proceedings of the 28th International Conference on Neural Information Processing Systems. MIT Press, 2015: 1225 – 1233.

[18] ZHANG Y, DAI S, HONG Y, et al. Application of genomic SSR locus polymorphisms on the identification and classification of chrysanthemum cultivars in China[J]. Plos One, 2014, 9(8):e104856.

[19] ROEIN Z, HASSANPOUR ASIL M, SABOURI A, et al. Genetic structure of chrysanthemum genotypes from Iran assessed by AFLP markers and phenotypic traits[J]. Plant Systematics & Evolution, 2014, 300(3):493 – 503.

[20] KHODAKOVSKAYA M, VANKOVá R, MALBECK J, et al. Enhancement of flowering and branching phenotype in chrysanthemum by expression of ipt under the control of a 0.821 kb fragment of the LEACO1 gene promoter[J]. Plant Cell Reports, 2009, 28(9):1351 – 1362.

[21] YAN H, BAI X X, WEI S, et al. The numerical classification of chrysanthemum flower color phenotype[J]. Acta Horticulturae Sinica, 2012,39(7): 1330 – 1340.

[22] DAI S L, HONG Y, DAI S L, et al. Chrysanthemum: rich diversity of flower color and full possibilities for flower color modification[J]. Acta Horticulturae, 2017,2017(1171):193 – 208.

[23] ZHANG J, SUI L, ZHUO L, et al. An approach of bag-of-words based on visual attention model for pornographic images recognition in compressed domain[J]. Neurocomputing, 2013, 110(8):145 – 152.

[24] NGUYEN Q K, LE T L, PHAM N H. Leaf based plant identification system for Android using SURF features in combination with Bag of Words model and supervised learning[C]// International Conference on Advanced Technologies for Communications. IEEE, 2014:404 – 407.

[25] LUO C, CHEN D, CHENG X, et al. SSR analysis of genetic relationship and classification in *Chrysanthemum germplasm* collection[J]. Horticultural Plant Journal, 2018, 4(2):73 – 82.

[26] LOWE D G. Object recognition from local scale-invariant features[C]// IEEE International Conference on Computer Vision. IEEE, 1999: 1150 – 1157.

[27] HAN J. Data mining: concepts and techniques[M]. Morgan Kaufmann Publishers Inc., 2011.

[28] NISTER D, STEWENIUS H. Scalable recognition with a vocabulary tree[C]// IEEE Computer Society Conference on Computer Vision & Pattern Recognition. IEEE Computer Society, 2006:2161 – 2168.

[29] CHEN K, ZHANG Z, LONG J, et al. Turning from TF – IDF to TF – IGM for term weighting in text classification[J]. Expert Systems with Applications, 2016, 66: 245 – 260.

[30] 袁培森, 沙朝锋, 王晓玲,等. 一种基于学习的高维数据 c-近似最近邻查询算法[J]. 软件学报, 2012, 23(8):2018 – 2031.
YUAN Peisen, SHA Chaofeng, WANG Xiaoling, et al. C-approximate nearest neighbor query algorithm based on learning for high-dimensional data[J]. Journal of Software, 2012, 23(8):2018 – 2031. (in Chinese)

[31] LESKOVEC J, RAJARAMAN A, ULLMAN J D. Mining of massive datasets[M]. Cambridge: Cambridge University Press, 2014.