Journal of Computer Applications
计算机应用,2019,39(9):2789 - 2794
ISSN 1001-9081
CODEN JYIIDU
2019-09-10
http://www.joca.cn

文章编号:1001-9081(2019)09-2789-06
DOI:10.11772/j.issn.1001-9081.2019020792

基于 pHash 分块局部探测的海量图像查重算法

唐林川,邓思宇,吴彦学,温柳英*
(西南石油大学 计算机科学学院,成都 610500)
(* 通信作者电子邮箱 wenliuying1983@163.com)

摘　要: 数据库中大量重复图片的存在不仅影响学习器性能,而且耗费大量存储空间。针对海量图片去重,提出一种基于 pHash 分块局部探测的海量图像查重算法。首先,生成所有图片的 pHash 值;其次,将 pHash 值划分成若干等长的部分,若两张图片的某一个 pHash 部分的值一致,则这两张图片可能是重复的;最后,探讨了图片重复的传递性问题,针对传递和非传递两种情况分别进行了算法实现。实验结果表明,所提算法在处理海量图片时具有非常高的效率,在设定相似度阈值为 13 的条件下,传递性算法对近 30 万张图片的查重仅需 2 min,准确率达到了 53%。

关键词: 重复图片检测;海量数据;感知 Hash;局部探测;传递性
中图分类号: TP391　　**文献标志码:** A

Duplicate detection algorithm for massive images based on pHash block detection

TANG Linchuan, DENG Siyu, WU Yanxue, WEN Liuying*
(*School of Computer Science, Southwest Petroleum University, Chengdu* 610500, *China*)

Abstract: The large number of duplicate images in the database not only affects the performance of the learner, but also consumes a lot of storage space. For massive image deduplication, a duplicate detection algorithm for massive images was proposed based on pHash (perception Hashing). Firstly, the pHash values of all images were generated. Secondly, the pHash values were divided into several parts with the same length. If the values of one of the pHash parts of the two images were equal to each other, the two images might be duplicate. Finally, the transitivity of image duplicate was discussed, and corresponding algorithms for transitivity case and non-transitivity case were proposed. Experimental results show that the proposed algorithms are effective in processing massive images. When the similarity threshold is 13, detecting the duplicate of nearly 300 000 images by the proposed transitive algorithm only takes about two minutes with the accuracy around 53%.

Key words: duplicate image detection; massive data; perception Hashing (pHash); block detection; transitivity

0　引言

随着计算机多媒体技术的快速发展,数字图像已经普遍出现在人们的日常生活中。同时,数字信息呈几何级数增长,对现有存储系统的容量、吞吐性能、可扩展性、可维护性和能耗管理等各个方面带来全新的挑战,且存储效率低和存储成本高等问题凸显,仅增加存储空间无法解决根本问题。在此情况下,消除冗余数据成为优化存储性能的重要手段,海量图像去重也是热门的研究分支之一,其目标是删除海量图像中重复的图像。

图像检索技术是图像去重的基本步骤,流行的图像检索技术是基于内容的(Content Based Image Retrieval, CBIR)[1-2]。CBIR 提取图像的颜色、形状、纹理等可视特征,对其特征进行量化表达,然后选择合适的度量方式进行匹配。图像的特征往往需要用高维向量来表达,因此大规模图像检索具有明显的特征维度高的特性。在此情况下,基于 Hash 的检索方法[3-4]被提出并得到快速发展,已经被广泛地应用在电子商务[5-7]、医学研究[8]、刑侦勘察[9]、版权保护[10]等领域。

目前,常用的 Hash 算法有 MD5[11]、SHA[12]和感知 Hash(perception Hashing, pHash)[13]等。基于 MD5 的图像去重算法存在严重的局限性,对于图像数据,任何微小的改变都会导致 MD5 的剧变,比如添加水印等,因此,针对图像去重问题,一般采用 pHash 检索算法。

图像 Hash 是将图像映射成较短的编码序列,叫作哈希指纹,用来表示其内容特征。通过计算图片间的哈希指纹的海明距离来判断两张图片间的相似度。传统感知 Hash 算法是一个 pair-wised 的算法,它对每一对图片都进行匹配,存在复杂度高、效率低等问题,不适合运用在大规模数据量的情况下。本文提出一种基于 pHash 的分块局部探测的海量图片查重算法,能够提高检索速度,同时避免误删除。

1　相关工作

1.1　图像感知 Hash 的特征表示

图像是一个定义在二维平面上的信号,它包括低频信号和高频信号。图像信号的幅值对应像素的灰度,因此图像频

收稿日期:2019-03-22;修回日期:2019-05-07;录用日期:2019-05-29。
基金项目:浙江省海洋大数据挖掘与应用重点实验室开放课题项目(OBDMA201601)。
作者简介:唐林川(1993—),男,四川成都人,硕士研究生,主要研究方向:主动学习、推荐系统;　邓思宇(1993—),女,四川遂宁人,硕士研究生,主要研究方向:主动学习;　吴彦学(1995—),男,四川巴中人,硕士研究生,主要研究方向:深度学习、特征学习;　温柳英(1983—),女,广西柳州人,讲师,博士,CCF 会员,主要研究方向:粗糙集、属性提取、粒计算。

率则反映了图像的像素灰度在空间中的变化情况。高频信号是信幅值变换强烈的地方，如图像的轮廓(边缘)，它描述了图像的细节；低频信号则是对图像亮度的综合度量。

图像 Hash 技术已经被广泛地运用在图像检索[14]、认证协议研究[15]、网络安全等领域[16]。其定义如下：

$$pHash = PH(I) \tag{1}$$

其中：PH 是感知哈希函数，I 是目标图像。感知哈希值有诸多形式，如实数向量和复杂的数据结构等，但通常情况下，感知哈希值一个二值化序列。

在信息论中，字符串 a、b 的海明距离定义如下：

$$dis(a,b) = \sum_{i=0}^{n-1} notNqual(a(i),b(i)) \tag{2}$$

它表示两个字符串对应位置的不同字符的个数，哈希指纹可以用字符串表示。相应地，可以用任何两个长度相同的哈希指纹间的距离来度量其相似度。

1.2 几种哈希算法

1.2.1 均值哈希算法

近年来，利用二值序列描述图像特征是图像检索中的一个热门方法，如随机 Ferns[17] 分类算法、基于局部二值模式(Local Binary Pattern，LBP)的人脸识别算法[18]等。均值 Hash 特征描述方法利用图像的低频信息，生成一个二值化序列，相对于 LBP 等特征描述方式，均值 Hash 带有更加丰富的结构信息。在图像检索中，图像均值 Hash 因计算量小，速度快，占有举足轻重的地位。算法根据其二值化序列计算图像间的相似度。均值 Hash 图像匹配算法步骤如下：

1) 图像预处理。确定目标尺寸，过滤高频信息，通常将图像缩小到 8×8 的尺寸，即保留 64 个像素。

2) 图像简化。将图像进行灰度变换，同时计算 64 个像素点的灰度平均值。

3) 灰度二值化。将像素的灰度与平均值进行比较，如灰度大于平均值，则取值为 1；反之为 0。

4) 计算 Hash 值。将上述结果拼接在一起就构成一个 64 位的二值化序列。

局部均值 Hash 与图像的像素点的灰度平均值有关，其特征提取公式如下：

$$hash(m,n) = \begin{cases} 1, & I(m,n) > V \\ 0, & I(m,n) < V \end{cases}; \quad (m,n) \in Rect \tag{3}$$

其中：$Rect$ 表示目标图像，V 表示目标图像像素平均值。

1.2.2 增强型哈希算法

基于均值的 Hash 算法主要问题是提取的特征值易受到均值影响，如对图像进行伽马矫正或直方图均衡。pHash 利用离散余弦变换(Discrete Cosine Transform，DCT)来获取图像的低频信息，能够有效避免这一问题。

DCT 是与傅里叶变换类似的运算，常用作信号和图处理，被视为一种数据压缩技术。DCT 将原始图像信息块转化成代表不同频率分量的系数集，大部分能量常常集中在频率域的低频范围内，因此，DCT 对图像进行压缩的原理是减少图像的高频分量。

增强型 Hash 算法分为以下几个步骤：缩小图像尺寸、简化色彩、图像 DCT、缩小 DCT 矩阵、计算 DCT 的均值、计算 Hash 值。如图 1 所示。

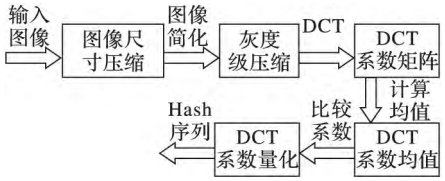

图 1 增强型 Hash 算法流程

Fig. 1 Flowchart of enhanced Hash algorithm

1) 图像预处理。确定目标尺寸，将图像缩小到 32×32 的尺寸。

2) 计算 DCT 系数。首先将图像进行灰度变换，再计算 DCT 系数。

3) 缩小 DCT 矩阵。将 DCT 系数矩阵缩小成 8×8，仅保留图像的低频区域。

4) 计算 DCT 系数矩阵的平均值。

5) 二值化。将矩阵中 DCT 系数与平均值进行比较：如 DCT 系数大于平均值，则取值为 1；反之为 0。

6) 计算 Hash 值。将上述结果拼接在一起就构成一个 64 位的 Hash 序列。

1.2.3 局部敏感哈希算法

Datar 等[19]提出的局部敏感哈希(Locality Sensitive Hashing，LSH)是较为流行的一种近似最近邻搜索算法，专门用于处理高维海量数据，可以具体运用到文本相似度检测、网页搜索等领域。

LSH 的基本思想是：高维空间的两个相似数据通过 Hash 函数映射到低维空间仍然具有很高的相似度。通过这种方式，避免在高维空间查找数据，能够有效降低时间、空间复杂度。

根据相似度度量方法和哈希函数族适用的数据空间不同，LSH 可以分为基于比特采样、基于随机投影、基于 p 稳定分布和基于最小独立排列等几种方式。

2 pHash 分块局部探测算法

传统的 pHash 算法进行图片去重往往是穷举法两两比较，在具有海量图片，比如图片量大于 10^5 级的情况下，穷举法的查重效率是非常低的，在单台 PC 上的运行时间是相当长的。本文提出的算法极大地提高了穷举法的运行效率，它可以被细分为两个子算法，分别是图像 pHash 索引构建算法和 pHash 分块探测算法。在此基础上，本文讨论了重复图像的传递性问题。

2.1 图像 pHash 索引构建算法

在初始阶段，算法将依次对训练集大小为 n 的图像集合进行 3 步处理，从而生成图像的 pHash 块：

1) 生成所有图像的 pHash(64 bit)指纹特征，或图像的二值化特征向量。该步骤的时间复杂度为 $O(n)$。

2) 将每个图像的二值化特征等分成 m 份，如指纹特征长度为 16，分为 4 等份，每个等份的长度为 4；该步骤的时间复杂度为 $O(mn)$。

3) 根据图像的 pHash 分块值生成索引，索引长度为 m。同一组分块特征值下，具有相同分块特征值的图像，位于同一索引下。该步骤的是时间复杂度为 $O(mn)$。

算法 1 给出了图像索引的构建过程。

算法 1 图像索引构建算法。

输入: 所有图像的 pHash 值集合 P; pHash 值的长度 p; 图像路径集合 M; pHash 值分裂块数 s;

输出: 图像的映射关系 f。

```
1)  f ← [∅]_{i=1}^{s}; l ← p/s
2)  for i ← 1 to s {
3)    B ← ∅
4)    for j ← 1 to | P | {
5)      b ← P_j [i* l,(i + 1) * l]
6)      if b ∉ B {
7)        f_i(b) ← ∅
8)        B ← B ∪ {b}
9)      }                                   //end if
10)     f_i(b) ← f_i(b) ∪ {M_j}
11)   }                                     //end for
12) }                                       //end for
13) return f
```

2.2　pHash 分块探测算法

该阶段利用算法 1 获得的图像索引 f, 计算索引块 f_i 中每一个局部特征值下的图像相互间的海明距离, 进而判断图片是否重复。通常情况下, 海明距离越大说明图片间的相似程度越低。

算法 2 给出了传递性重复图像的检测过程。

算法 2　传递性重复图像检测算法。

输入: 所有图像的 pHash 值集合 P; 图像路径集合 M; pHash 图像索引 f; 相似度阈值 t;

输出: 集合 D, D_i 是一组重复图像集合。

```
1)  D ← ∅
2)  for i ← 1 to | f | {
3)    for b ∈ f_i{
4)      for (j1, j2) ∈ C [f_i(b)] {
5)        if dist(P_j1, P_j2) ≤ t {
6)          if ∀D_k1, D_k2 ∈ D, M_j1 ∉ D_k1 ∧ M_j2 ∉ D_k2{
7)            D ← D ∪ {M_j1, M_j2}
8)          } else if( ∃D_k1 ∈ D, M_j1 ∈ D_k1) ∧ ( ∀D_k2 ∈ D,
              M_j2 ∉ D_k2) {
9)            D ← D - {D_k1}
10)           D_k1 ← D_k1 ∪ { M_j2}
11)           D ← D ∪ { D_k1}
12)         } else if ( ∃D_k2 ∈ D, M_j2 ∈ D_k2) ∧ ( ∀D_k1 ∈ D,
              M_j1 ∉ D_k1) {
13)           D ← D - {D_k2}
14)           D_k2 ← D_k2 ∪ {M_j1}
15)           D ← D ∪ {D_k2}
16)         } else if( ∃D_k1 ∈ D, M_j1 ∈ D_k1) ∧ ( ∀D_k2 ∈ D,
              M_j2 ∉ D_k2) {
17)           D ← D - {D_k1, D_k2}
18)           D_k1 ← D_k1 ∪ D_k2 ∪ { M_j1, M_j2}
19)           D ← D ∪ {D_k1}
20)         }                                 //end if
21)       }                                   //end if
22)     }                                     //end for
23)   }                                       //end for
24) }                                         //end for
25) return D
```

在算法 2 中,1) 进行初始化操作; 2) 定位到第 i 个索引集; 3) ~23) 对第 i 个索引集中每一个映射, 判断其中的图片对的相似度是否小于或等于阈值, 根据条件调整最终的重复图片集合, 其中,17) ~19) 将两个重复图片集合融合到一起, 这是由重复的传递性决定的。

针对 pHash 分块探测算法, 需要遵循的原则是图片间重复性的传递思想; 即图片 a 和图片 b 互为重复图片, 图片 a 和图片 c 互为重复图片, 那么图片 b 和图片 c 互为重复图片。文中的相似度阈值由专家设定, 具有一定的随机性和差异性。

算法 3 给出了非传递性重复图像的检测过程。

算法 3　非传递性重复图像检测算法。

输入: 所有图像的 pHash 值集合 P; 图像路径集合 M; pHash 图像索引 f; 相似度阈值 t;

输出: 集合 D, D_i 是一组重复图像集合。

```
1)  Define g: Z+ → 2^{Z+}
2)  D, S ← ∅, ∅
3)  l ← | p_l | / | f |
4)  for i ← 1 to | M | {
5)    if i ∉ S{
6)      for j ← 1 to | f | {
7)        b ← P_i [j* l,(j + l) * l]
8)        for k ∈ (f_j(b) - {i}) {
9)          if dist(P_i, P_k) ≤ t {
10)           if i ∉ g {
11)             g(i) ← ∅
12)             g(i) ← g(i) ∪ {k}
13)           }                             //end if
14)         }                               //end if
15)       }                                 //end for
16)     }                                   //end for
17)     if i ∈ g {
18)       g(i) ← g(i) ∪ {i}
19)       S ← S ∪ g(i)
20)     }                                   //end if
21)     for j ∈ g(i) {
22)       for k ← 1 to | f | {
23)         b ← P_j [k* l,(k + 1) * l]
24)         f_i(b) ← f_i(b) - {j}
25)       }                                 //end for
26)     }                                   //end for
27)   }                                     //end if
28) }                                       //end for
29) for i ∈ g {
30)   D ← D ∪ {g(i)}
31) }                                       //end for
32) return D
```

在算法 3 中,1) ~3) 进行初始化操作,1) 中定义了一个映射, 键为图片 id, 值为图片 id 对应的重复图片集; 4) ~5) 定位第 i 张图片并判断第 i 张图片是否已经被判断为重复; 6) ~16) 将第 i 张图片的 pHash 分块并在不同的 pHash 索引集中探测重复图片; 17) ~26) 实现非传递性, 已经判断为重复的图片集合将从 pHash 索引中去除。

2.3　样例分析

本节提供了一个样例来说明 pHash 分块局部探测算法如何进行 pHash 分块, 并将全局探测方法转化为局部探测方法。

第一步, 生成每张图像的哈希值, 并等分成四组, 如表 1 所示。

第二步, 根据哈希块中的哈希值完成匹配, 可建立映射关系, 如表 2 所示。x_3, x_4 的 pHash1 中的值都是 caad, 即 x_3, x_4 可

能是重复图像，则将 x_3，x_4 两张图像放在同一索引下。同理可计算 pHash2、pHash3 和 pHash4 块。

第三步，计算得 x_3，x_4 完整哈希值的海明距离为 9，则判断 x_3，x_4 是不重复图片。

表 1 图像的 pHash 分块值

Tab. 1 pHash blocks of images

Image/Block	pHash1	pHash2	pHash3	pHash4
x_1	c5d8	f419	06e9	bc53
x_2	c5e8	f419	06f9	bc53
x_3	caad	95c6	6c49	9ba4
x_4	caad	b55a	684b	95a9
x_5	e4b7	4b38	06e9	c370
x_6	ec90	f419	470d	8562

表 2 pHash1 中的索引

Tab. 2 Index of pHash1

pHash	mapping
c5d8	[x_1]
c5e8	[x_2]
caad	[x_3, x_4]
e4b7	[x_5]
ec90	[x_6]

2.4 算法优势

本文提出的算法相比于传统 pHash 穷举查重的算法而言，优势在于：在保证精度与传统方法相当的情况下，极大提高了算法的运行效率。传统 pHash 算法通过穷举来进行查重，穷举的手段是进行两两比较。很显然，任意两张图片都需要比较其 pHash 序列的汉明距离。假设 pHash 分块数为 s，那么存在以下 3 种情况：

1）相似度阈值 $t < s$。此时必然存在某个 pHash 块 $pHash_i$，使得相似（即相似度小于或等于 t）的图片对被存放到 $pHash_i$ 中的同一个索引，也就是说，所有相似的图片对必然能够被算法检测到。

2）相似度阈值 $t = s$。此时存在这样一种特殊情况无法被算法检测到：图片 A 和图片 B 在每个 pHash 块中均存在且只存在 1 个比特位不同，此时相似度为 t。又假设图片 A 和图片 B 在相似度为 t 的条件下，不同的比特位之间相互独立，且在不同的 pHash 块中出现的概率是相等的。此时，无法被算法检测到的情况发生的概率仅为 $s!/s^s$。这相当于将 s 个不同的小球等概率地放进 s 个不同的盒子，每个盒子均不为空的概率。特别地，若 $t = s = 4$，那么这个概率为 3/32，当 s 越大，这个值越小。

3）相似度阈值 $t > s$。此时采用与 2）种情形相同的假设，那么无法被检测到的情况则有 $t - s + 1$ 种，分别是当相似度为 s、$s+1$、…、t 时，每个 pHash 块均有至少一个比特位相同的情况。当 t 比 s 大得多的时候，无法被检测到的情况发生的概率较大。

所以，本文算法在第一种情况下，能够达到与传统算法一样的精度。但后两种情况则表明，本文提出的算法与传统方法依然存在一定的精度差距。尽管如此，传统算法也只是局限于理论精度，它在处理海量图片时显得非常乏力，甚至不可计算。

本文算法运行效率的提高是通过将传统 pHash 穷举查重转化为局部查重来实现的。此时，图片全集将被划分成若干较小的不相交子集，在子集上进行两两比较远高于在全集上进行两两比较的运行效率。例如，有 100 张图片需要查重，本文算法将其划分成了 20 个子集，每个子集有 5 张图片，那么本文算法仅需做 $20 \times 10 = 200$ 次两两比较，而传统算法将要做 4 950 次比较，这极大地降低了比较次数。

3 实验及分析

在本章中，首先说明数据集的来源和生成方式；其次，自定义一个查重精度的评价指标；最后，分别探讨重复的传递性和非传递性对实验结果的影响，并给出算法的运行时间。

3.1 数据集

本文采用的数据集为淘宝的商品图片集合，图片总量为 81 293 张，因为无法确定该数据集是否重复，所以先需要对这些图片去一次重复。设定汉明距离阈值为 3，在此情况下，有 9 546 张图片重复，将重复图片删除，最终得到的图片总量为 71 747。

接着，利用一系列图像处理方法生成重复的图片数据，具体方法如下：

1）亮度调整。随机地选择一个亮度调整率 $\alpha \in [-0.25, 0.25]$，负值表示图片变暗；反之表示图片变亮。

2）对比度调整。随机地选择一个对比度调整率 $\beta \in [0, 3]$。

3）饱和度调整。随机地选择一个饱和度调整率 $\gamma \in [0, 3]$。

4）剪裁。随机剪裁图片，剪裁后的图片大小为原始图片大小的 0.8 ~ 1 倍。

5）噪声。随机给图像添加高斯白噪声、泊松噪声或是椒盐噪声。

6）高斯模糊。随机设定正态分布的标准差 $\sigma \in (0, 3]$，模糊半径 $r \in \{1, 2, 3, 4, 5\}$。

通过以上方式，能够知道哪些图片是重复的，并可利用先验信息对本文算法作出较为合理的评价。按照上述方法，针对数据集中的每一张图片，重复生成三张图片，最后得到的图像总量为 286 988 张。

2.4 节从理论上分析了本文算法和传统算法所得到的精度是相当的，所以，这里生成的数据集是为了验证算法可用并且具有较高的执行效率。在接下来的实验中，将会看到这一点。

3.2 评价指标

由于本文研究图片去重问题，但此处所说的重复并不是指图片一定完全一致。从人的感知上讲，图片的重复应该具有局部不敏感的特点，即图片中少量像素点的不同不影响人的感知。由于图片查重相当于将重复图片聚到相同的簇，这可被看成是一个聚类问题，因此，本文使用如下评价指标来衡量算法的查重效果：

$$acc(A) = \left(\sum_{a \in A} max\_dup(a) \right) / N \tag{4}$$

其中：A 是图片重复检测的结果集合，其中的元素为检测到的重复图片。max_dup 函数为 a 中最大的真实重复图片个数。例如，假设 $a = [1, 1, 2, 2, 2, 3, 3]$，那么 $max\_dup(a) = 3$，即 a 中

出现次数最多的元素个数,也就是 2 的个数。实际上,acc 就是聚类纯度。

3.3　实验结果

本文从多个角度来衡量算法的有效性,分别是查重精度 acc、检测到的重复图片数量分布和查重时间。实验设置主要是针对相似度阈值的设置,这里相似度阈值范围被设定为 {0,1/64,…,17/64}(为了更好地显示结果,在图中省略了分母)。实验在单台 MacOS Intel i5 环境上进行,查重精度随相似度阈值的变化曲线如图 2 所示。

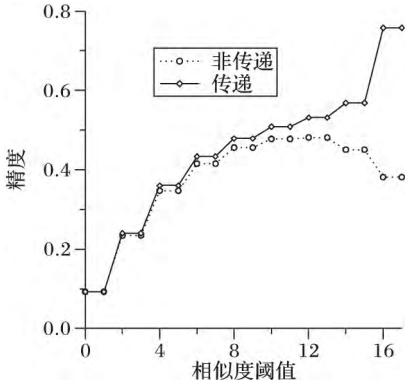

图 2　查重精度随相似度阈值的变化曲线

Fig. 2　Duplicate detection accuracy w. r. t. similarity threshold

图 2 显示,具有传递性的图片重复检测效果明显优于非传递性,且非传递性的结果并不是单调的,这是因为随着相似度阈值的增加,有较多的不同图片被错误地检测为重复。但是具有传递性的结果恰恰相反,尽管有较多的不同图片被错误地检测为重复,但是由于传递的性质,较多的相同图片也会被放到同一个集合。

图 3(a) 和(b) 所示为相似度阈值为 10,非传递性和传递性检测算法分别得到的重复图片集合 size 的分布直方图。

可以看到,对非传递性检测算法而言,大部分情况下,该算法检测到的重复图片为两张。少部分情况下可以探测到更多的重复图片。传递性检测算法和非传递性检测算法得到的结果类似,不同地是,该算法所检测的图片集合的 size 分布具有更广的范围,这是因为传递性本身就会使得集合的 size 增加。

图 4 展示了两种重复图片检测算法随相似度阈值变化的时间花费曲线。可以看到,非传递的检测算法运行时间随着相似度阈值呈线性增长关系,且时间花费增长不明显。相反地,传递性检测算法在相似度阈值为 14 的时候,时间花费陡增,由于相似度阈值为 16,17 的时候,时间花费过大,图中没有给出。此时传递性算法几乎没有实用性。但这也符合预期,因为在相似度阈值达到一定值的时候,由于传递性的存在,许多小的重复图片集合将会被融合到一起,此时针对该集合的检测将花费大量时间。综合图 2 和图 4 来看,在相似度阈值为 13 的时候,对近 30 万张图片的传递性重复检测算法的时间花费仅需 2 min。此时传递性算法的精度达到了 53%,非传递性算法的精度尽管达到了最高点,但仍不及传递性算法。如果进一步提升相似度阈值,传递性算法的运行时间将急剧上升。这是因为传递性本身会使得检测到的重复图片集合增大。如果在相似度设定也较大的情况,传递性算法检测到的重复图片集合会非常庞大,针对该重复图片集合元素两两比较所花费的时间也会非常多。

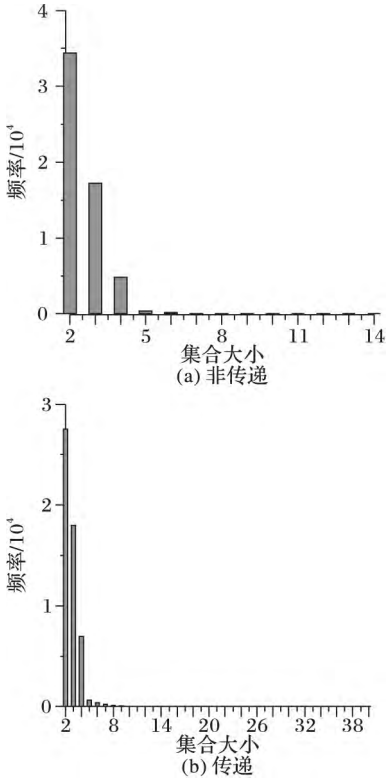

图 3　重复图片集合 size 分布直方图

Fig. 3　Distribution histogram of the size of duplicate images

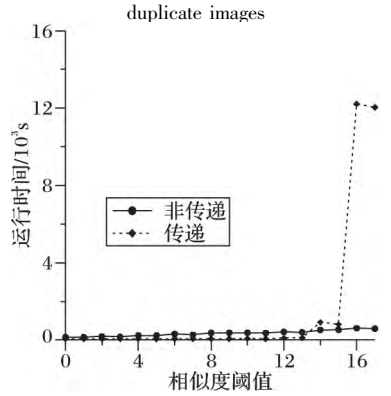

图 4　运行时间随相似度阈值的变化曲线

Fig. 4　Runtime w. r. t. similarity threshold

4　结语

本文提出了一种新型的图片查重算法,首先计算出所有图片的 pHash 指纹,接着对 pHash 指纹进行分块,目的是将全局查重转变为局部查重。这极大地提高了重复图片检测的效率。设置相似度阈值为 13 的条件下,采用传递性查重算法处理近 30 万张图片仅需大约 2 min,精度达到了 53%。未来将会采用更加优质的图片指纹算法,以期获得更好的结果。

参考文献(References)

[1] SMEULDERS A W M, WORRING M, SANTINI S, et al. Content-based image retrieval at the end of the early years [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, 22(12): 1349 - 1380.

[2] LIU Y, ZHANG D, LU G, et al. A survey of content-based image retrieval with high-level semantics [J]. Pattern Recognition, 2007,

40(1): 262 - 282.

[3] KUO Y H, CHEN K T, CHIANG C H, et al. Query expansion for hash-based image object retrieval [C]// MM '09: Proceedings of the 17th ACM International Conference on Multimedia. New York: ACM, 2009: 65 - 74.

[4] ZHEN Y, YEUNG D Y. Active hashing and its application to image and text retrieval [J]. Data Mining and Knowledge Discovery, 2013, 26(2): 255 - 274.

[5] LI Q-N, LI Y-Q, JIANG S-J. Application of a new Hash function in e-commerce [C]// ICEE '12: Proceedings of the 2012 3rd International Conference on E-Business and E-Government. Washington, DC: IEEE Computer Society, 2012, 4: 223 - 225.

[6] WRIGHT A. Controlling risks of e-commerce content [J]. Computers and Security, 2001, 20(2): 147 - 154.

[7] **张文丽,钟晓燕,冯前进,等. 基于** Hash **函数敏感性的医学图像精确认证**[J]. **中国图象图形学报**, 2008, 13(2): 204 - 208. (ZHANG W L, ZHONG X Y, FENG Q J, et al. Hard authentication for medical image based on sensitivity of Hash function [J]. Journal of Image and Graphics, 2008, 13(2): 204 - 208.)

[8] **赵峰**. Hash **签名在电子商务中的应用**[J]. **计算机与数字工程**, 2014, 42(3): 531 - 534. (ZHAO F. Hash signature application in the electronic commerce [J]. Computer and Digital Engineering, 2014, 42(3): 531 - 534.)

[9] ZHAN S, ZHAO J, TANG Y, et al. Face image retrieval: super-resolution based on sketch-photo transformation[J]. Soft Computing, 2018, 22(4): 1351 - 1360.

[10] AL-MANSOORI S, KUNHU A. Hybrid DWT-DCT-Hash function based digital image watermarking for copyright protection and content authentication of DubaiSat-2 images [C]// Proceedings of the High-Performance Computing in Remote Sensing IV. Bellingham, WA: SPIE, 2014, 9247: 924707.

[11] DEEPAKUMARA J, HEYS H M, VENKATESAN R. FPGA implementation of MD5 hash algorithm [C]// Proceedings of the 2001 Canadian Conference on Electrical and Computer Engineering. Piscataway, NJ: IEEE, 2001, 2: 919 - 924.

[12] GREMBOWSKI T, LIEN R, GAJ K, et al. Comparative analysis of the hardware implementations of hash functions SHA-1 and SHA-512 [C]// Proceedings of the 2002 International Conference on Information Security, LNCS 2433. Berlin: Springer, 2002: 75 - 89.

[13] SHIM H. PHash: a memory-efficient, high-performance key-value store for large-scale data-intensive applications [J]. Journal of Systems and Software, 2017, 123: 33 - 44.

[14] LIN K, YANG H-F, HSIAO J-H, et al. Deep learning of binary hash codes for fast image retrieval [C]// Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops. Piscataway, NJ: IEEE, 2015: 27 - 35.

[15] **刘明生,王艳,赵新生. 基于** Hash **函数的** RFID **安全认证协议的研究**[J]. **传感技术学报**, 2011, 24(9): 1317 - 1321. (LIU M S, WANG Y, ZHAO X S. Research on RFID security authentication protocol based on Hash function [J]. Chinese Journal of Sensors and Actuators, 2011, 24(9): 1317 - 1321.)

[16] **周国强,田先桃,张卫丰,等. 基于图像感知哈希技术的钓鱼网页检测**[J]. **南京邮电大学学报(自然科学版)**, 2012, 32(4): 59 - 63. (ZHOU G Q, TIAN X T, ZHANG W F, et al. Detecting phishing Web pages based on image perceptual hashing technology [J]. Journal of Nanjing University of Posts and Telecommunications (Natural Science), 2012, 32(4): 59 - 63.)

[17] KURSA M B, JANKOWSKI A, RUDNICKI W R. Boruta—a system for feature selection [J]. Fundamenta Informaticae, 2010, 101(4): 271 - 285.

[18] **宁星,蒋年德. 基于** LBP **人脸识别算法的预处理研究**[J]. **电子质量**, 2012(4): 76 - 77. (NING X, JIANG N D. Pretreatment research for face recognition based on LBP [J]. Electronic Quality, 2012(4): 76 - 77.)

[19] DATAR M, IMMORLICAL 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: ACM, 2004: 253 - 262.

This work is partially supported by the Open Project of Key Laboratory of Data Mining and Application of Zhejiang Ocean University (OBDMA201601).

TANG Linchuan, born in 1993, M. S. candidate. His research interests include active learning, recommender systems.

DENG Siyu, born in 1993, M. S. candidate. Her research interests include active learning.

WU Yanxue, born in 1995, M. S. candidate. His research interests include deep learning, feature learning.

WEN Liuying, born in 1983, Ph. D., lecturer. Her research interests include rough set, attribute extraction, granular computing.