一种稳健的紧凑图像哈希算法

张智丰, 裴志利*

内蒙古民族大学计算机科学与技术学院, 内蒙古 通辽 028000

摘要 当前图像哈希认证技术对旋转操作较为敏感,稳健性与伪造检测率不高。为克服此类问题,设计了基于改进的局部二值模式(LBP)算子与动态更新变换的紧凑图像哈希算法。引入线性插值技术,对输入图像实现预处理,改善哈希序列对尺度缩放的稳健性。利用 Ring 分割,将插值图像转化成二次图像。考虑中心像素特性与周围像素的差异,将 Heaviside 函数嵌入传统的 LBP 中,形成了描述能力较强的 H-LBP 算子,提取图像的抗旋转稳健特征。引入压缩感知,对高维特征矢量完成降维,输出紧凑的过渡哈希数组。利用混沌变换思想,设计动态更新变换机制,对过渡哈希数组进行加密,得到图像哈希序列。最后,利用汉明距离测算输入图像与待检测图像的哈希相似度,通过优化认证阈值,完成图像内容的真伪识别。实验结果显示,与当前哈希算法相比,提出的算法生成的哈希序列尺寸更小,对旋转、噪声等操作具有更好的感知稳健性。

关键词 图像处理; 哈希算法; Ring 分割; Heaviside 函数; H-LBP 算子; 动态更新变换; 压缩感知; 汉明距离

中图分类号 TP391 **文献标识码** A

doi: 10.3788/LOP54.101002

A Robust Compact Image Hash Algorithm

Zhang Zhifeng, Pei Zhili

College of Computer Science and Technology, Inner Mongolia University for Nationalities, Tongliao, Inner Mongolia 028000, *China*

Abstract In order to overcome such defects as low robustness and low detection rate induced by the sensitivity of current hash image authentication algorithm to the rotation operation, the compact image hash algorithm based on an improved local binary pattern (LBP) operator and the dynamic update transform is proposed. The linear interpolation technique is introduced to preprocess the input image and improve the scaling robustness of the hash sequence. The Ring division is used to transform the interpolation image into the secondary image. A H-LBP operator with strong descriptive ability was designed when the Heaviside function is embedded into traditional LBP operator. Considering the difference between the center pixel characteristics and the adjacent pixels, the anti-rotation robustness features are extracted. The compression sensing is introduced to reduce the dimensions of high-dimensional feature vector, and a compact transition hash array is output. A dynamic update transform mechanism is designed with the chaotic transform idea to encrypt the transition hash array, and the image hash sequence is obtained. Finally, the hash similarity between the input image and the detected image is calculated based on Hamming distance to achieve the authenticity of the image through the optimized decision threshold. The experimental results show that the proposed algorithm generates a smaller hash sequence and has better perceptual robustness to rotation, noise and other operations than the current hash algorithms.

Key words image processing; hash algorithm; Ring segmentation; Heaviside function; H-LBP operator; dynamic update transform; compression sensing; Hamming distance

OCIS codes 100.2000; 100.3008; 110.2970; 110.3055

收稿日期: 2017-04-07; **收到修改稿日期**: 2017-05-10

基金项目: 国家自然科学基金(61163034,61373067)、内蒙古自然科学基金(2013MS0911)、内蒙古自治区草原英才工程(2013)、内蒙古自治区青年科技领军人才计划(NJYT-14-A09)、内蒙古自治区 321 人才工程二层次人选(2010)、内蒙古人才开发基金(2011)、内蒙古自治区高等学校教学改革科学研究项目(2015NMJG036)、内蒙古民族大学科学研究项目(NMDYB1453)、内蒙古自治区高等学校科学研究项目(NJZY17192)、内蒙古自治区科技创新引导项目(2017)

作者简介: 张智丰(1972—),男,硕士,副教授,主要从事计算机图形图像技术、信息安全、数据挖掘等方面的研究。E-mail: nmGzhangzf1972@163.com

* **通信联系人**。E-mail: PeiZhili1968@163.com

1 引 言

伴随着计算机科学的发展与完善,市场上出现了各种各样强大的图像编辑工具。由于人眼难以识别数字图像真伪,攻击者可任意篡改数字图像,给图像信息安全带来巨大隐患[1-2]。为了使用户能够决策图像真伪,研究人员提出了图像哈希技术[3-4]。当前图像哈希生成机制分为 3 个阶段:预处理、稳健特征提取以及哈希序列的生成。特征提取是整个哈希算法的核心[3]。蒋翠玲等[4]设计了基于遗传算法(GA)和反向传播(BP)网络的稳健图像哈希方法,利用提升小波变换与傅里叶变换处理图像的低频分量,提取图像幅度与相位信息,再通过构建 GA-BP 模型,获取最终的图像哈希序列。但是,该技术没有充分利用图像中的稳健特征,对大角度旋转攻击的稳健性较弱,且哈希序列的空间维数较高,使该技术效率较低。Tang 等[5]提出基于颜色矢量与边缘检测的图像哈希认证算法,通过提取归一化图像的颜色矢量与边缘,并计算图像颜色矢量角度与边缘的统计特征,生成哈希序列。实验结果验证了该算法具有良好的认证精度,表现出理想的受试者工作特征(ROC)曲线。但是该技术仅利用颜色矢量与边缘等特质生成哈希序列,忽略了图像结构与纹理信息,使其对旋转攻击的检测能力不佳。Vadlamudi 等[6]设计了基于图像内容直方图的图像哈希算法,实验数据验证了该算法的合理性与优异性。虽然该技术对图像内容篡改攻击具有很好的检测精度,但是子块直方图对旋转攻击的敏感性较低。

为此,本文提出基于改进的局部二值模式(LBP)算子与动态更新变换的紧凑图像哈希算法。利用插值技术与 Ring 变换机制,将初始图像变为二次图像,改善其对缩放与旋转的敏感性;考虑中心像素特性,将 Heaviside 函数嵌入传统的 LBP 中,形成了一种描述能力较强的算子,即 H-LBP 算子,提取图像的抗旋转稳健特征;利用压缩感知技术对特征矢量进行降维,获取紧凑哈希数组;设计了动态更新变换机制,对哈希数组进行加密,输出哈希序列;优化决策阈值 W,结合汉明距离 D,利用哈希序列的唯一性识别图像内容;最后,验证了所提哈希技术的稳健性与安全性。

2 紧凑图像哈希算法

基于改进的 LBP 算子与动态更新变换的紧凑图像哈希算法的认证过程如图 1 所示。

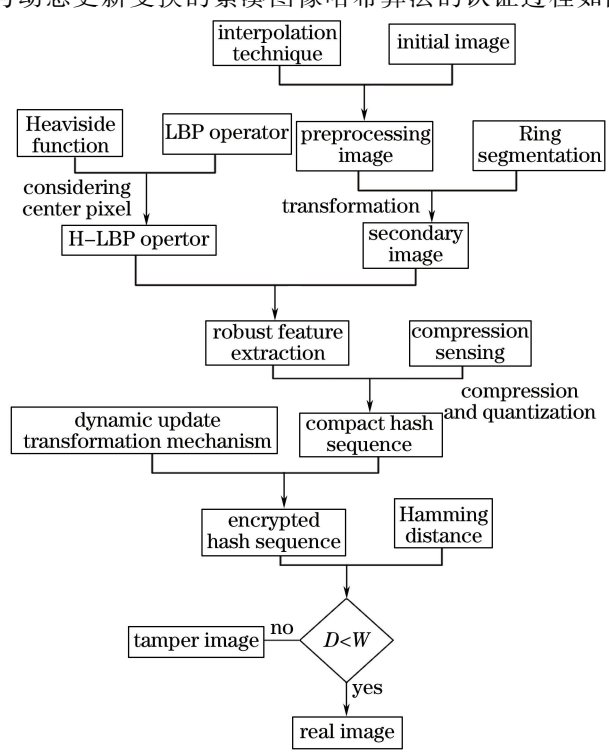

图 1 图像哈希算法过程

Fig. 1 Process of the proposed image hash algorithm

2.1 图像预处理

引入插值技术[7]来规范图像尺寸,增强哈希序列对缩放的稳健性。随后,利用高斯低通滤波对插值图像进行预处理。高斯低通滤波模型为[7]

$$\begin{cases} T_{\mathrm{G}}(i,j)=\dfrac{T(i,j)}{\sum\limits_{i}\sum\limits_{j}T(i,j)} \\ T(i,j)=\exp\left(-\dfrac{i^2+j^2}{2\sigma^2}\right) \end{cases}, \tag{1}$$

式中 $T_{\mathrm{G}}(i,j)$是卷积掩模在(i,j)处的元素,σ 代表卷积掩模的标准差。利用插值技术与(1)式处理尺寸为 $M\times N$ 的初始明文 $f(x,y)$后,输出预处理图像 $f_0(x,y)$。

2.2 基于 Ring 分割的二次图像生成

借助 Ring 分割[8],对预处理图像 $f_0(x,y)$进行分环处理,根据其他位置像素与中心像素间的距离 d 形成一个像素集合,通过对像素集合进行排序与插值处理,输出新的图像,本文简称为二次图像。该处理技术有效地增强了二次图像对旋转变换的稳健性。若 n 为环形数量,R_k 代表像素值集合,r_k 为 $f_0(x,y)$内第 k $(k=1,2,\cdots,n)$个环形的半径,则 $f_0(x,y)$中 r_n 为

$$r_n=\mathrm{floor}\left(\frac{M}{2}\right), \tag{2}$$

式中 floor()是向下取整操作。

对于其他环形半径,需利用内接圆面积 A 及其均值 u_A 来计算:

$$A=\pi r_n^2,\quad u_A=\mathrm{floor}\left(\frac{A}{n}\right)。 \tag{3}$$

因此,对于第一个环形,其半径为

$$r_1=\mathrm{floor}\left(\frac{u_A}{\pi}\right)。 \tag{4}$$

根据(4)式,其他环形区域的半径 $r_k(k=2,3,\cdots,n-1)$的计算公式为

$$r_k=\sqrt{\frac{u_A+\pi^2}{\pi}}。 \tag{5}$$

再根据(2)~(5) 式,获取其他位置与中心像素的距离 d。

令$(x_{\mathrm{c}},y_{\mathrm{c}})$为图像中心,则距离 d 为

$$d=\sqrt{(x-x_{\mathrm{c}})^2+(y-y_{\mathrm{c}})^2}。 \tag{6}$$

基于(6)式的 d 值,将剩余像素值归类为 n 个集合:

$$R_1=\{p(x,y)\,|\,d\leqslant r_1\}, \tag{7}$$

$$R_k=\{p(x,y)\,|\,r_{k-1}<d\leqslant r_k\}\,(k=2,3,\cdots,n), \tag{8}$$

式中 $p(x,y)$为(x,y)处的像素值,r_k 为第 k 个环形半径。

由(7)式和(8)式,可获取 $f_0(x,y)$对应的 $R_k(k=1,2,\cdots,n)$,对其元素进行升序排列,获取新的矢量 $\boldsymbol{u}_k$,即可确保 $\boldsymbol{u}_k$ 与旋转操作无关。再次利用插值技术[7],将 $\boldsymbol{u}_k$ 变为 $u_A\times 1$ 维矢量 $\boldsymbol{v}_k$。对 $\boldsymbol{v}_k$ 进行升序重排,输出二次图像:

$$\boldsymbol{V}=[\boldsymbol{v}_1,\boldsymbol{v}_2,\cdots,\boldsymbol{v}_n]。 \tag{9}$$

2.3 基于 H-LBP 算子的稳健特征提取

LBP 算子[9-10]以矩形中心点的灰度值为阈值,对矩形内其他像素作二值化处理,然后根据像素的不同位置进行加权求和得到该窗口的 LBP 值,计算公式为

$$Q_{\mathrm{LBP}}^{P,R}=\sum_{i=1}^{P}S(g_i-g_{\mathrm{c}})\,2^{i-1},\ i=1,2,\cdots,P, \tag{10}$$

$$S(x)=\begin{cases}1, & x\geqslant 0\\ 0, & x<0\end{cases}, \tag{11}$$

式中 g_c 为中心像素值，g_i 为圆环上第 i 个像素的像素值，R 是圆形半径，P 为像素数量，2^{i-1} 代表像素权重，用 W^{i-1} 来表示。3×3 窗口的 LBP 算子计算过程如图 2 所示。

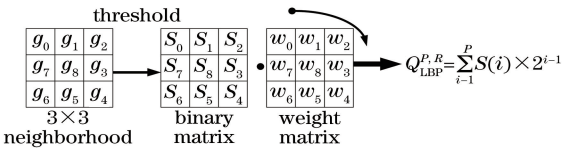

图 2 LBP 算子的计算过程

Fig. 2 Computation of LBP operator

虽然 LBP 算子对图像纹理具有较强的描述能力，但也存在以下缺陷：1) 抗噪声干扰能力较差；2) 虽然 LBP 算子能够描述局部区域的差异，但是无法显示这个差异值。如图 3 所示，位于不同区域的两个像素点的 LBP 值却是相同的，说明传统的 LBP 算子赋予相邻像素点的权重是相等的。虽然诸多学者对 LBP 算子进行了拓展，但是改进后的 LBP 算子主要是利用硬阈值来描述特征，因此稳健性不佳[10]。

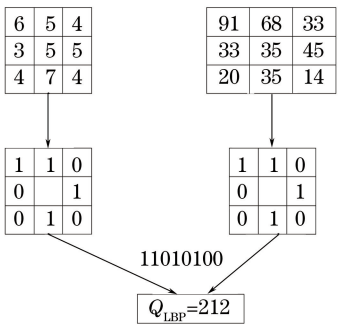

图 3 不同区域产生相同的 LBP 值

Fig. 3 Same LBP value in different regions

为了改善哈希算法对噪声的稳健性，充分提取二次图像 $\boldsymbol{V}$ 的稳健特征，将二次图像 $\boldsymbol{V}$ 中灰度值高于当前位置像素灰度值的邻域像素记录下来：

$$T=\sum_{i=1}^{8}S(f_i-f_c),S(x)=\begin{cases}1, & x\geqslant 0\\ 0, & x<0\end{cases},\tag{12}$$

式中 f_i 为当前像素的像素值，f_c 为邻域像素的像素值，$S(x)$为比较函数，T 为像素记录数量，在本文中作为一个阈值，用来识别细小灰度变化，删除平滑区域内的像素点。

LBP 算子的比较函数过于简单，无法描述相邻像素间的差异程度，故本文引入 Heaviside 函数，简称 H 函数[11]，其表达式为

$$H(f_i-f_c)=\begin{cases}0, & f_i-f_c<T\\ \dfrac{1}{2}\left[1+\dfrac{2}{\pi}\arctan\left(-\dfrac{f_i-f_c}{d}\right)\right], & \text{otherwise}\end{cases},\tag{13}$$

$$d=\frac{(f_i-f_c)^2}{f_i+f_c},\tag{14}$$

式中 d 为二次图像 $\boldsymbol{V}$ 的中心像素与其邻域像素间的相似度距离，f_i-f_c 为像素差异度。

根据(13)式和(14)式可知，Heaviside 函数较好地描述了二次图像 $\boldsymbol{V}$ 的中心像素与其邻域像素间的差异度。中心像素与其邻域像素间的相似度越大，则(14)式的 d 值越大。

为了使(13)式中较大的 H 值获得更大的权重，以单调递增的方法对 Heaviside 函数值进行排序，将此操作简称为 Compose 过程，其函数为

$$H_C(f_i-f_c)=\mathrm{Compose}[H(f_i-f_c)]。\tag{15}$$

将 Heaviside 函数嵌入(10)式中，形成 H-LBP 算子：

$$Q_{\text{H-LBP}}^{P,R}=\begin{cases}0, & T=0\\ \sum_{i=1}^{P}H_{\text{C}}(f_i-f_{\text{c}})\,2^{i-1}, & R\leqslant T\leqslant P-1,\\ 0, & T=P\end{cases}\tag{16}$$

式中 R 为圆形半径，P 为像素数量，2^{i-1} 代表像素权重。

取 $R=1$，$P=8$，则(16)式变为

$$Q_{\text{H-LBP}}^{8,1}=\begin{cases}0, & T=0\\ \sum_{i=1}^{8}H_{\text{C}}(f_i-f_{\text{c}})\,2^{i-1}, & 1\leqslant T\leqslant 7。\\ 0, & T=8\end{cases}\tag{17}$$

当 $T=0$ 时，表示该像素点为孤立点；当 $T=8$ 时，表示该像素点为孤立点或者是位于平滑区域内的像素点。

将 Heaviside 函数引入 LBP 算子后，不同区域的两个像素点的 H-LBP 值也是不同的，有效提高了特征提取的准确度，如图 4 所示。

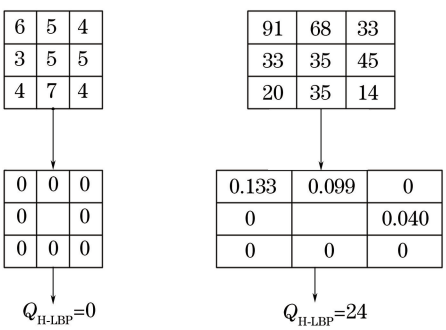

图 4 不同区域产生不同的 H-LBP 值

Fig. 4 Different H-LBP values in different regions

对 H-LBP 算子进行归一化处理：

$$H(b)=\frac{1}{M\times N}\sum_{i=1}^{M}\sum_{j=1}^{N}f\left[Q_{\text{H-LBP}}(i,j),b\right],\ b\in[0,S],\tag{18}$$

$$f(x,y)=\begin{cases}1, & x=y\\ 0, & \text{otherwise}\end{cases},\tag{19}$$

式中 S 是最大的 H-LBP 值。

将由(18)式得到的 $H(b)$ 视为二次图像 $\boldsymbol{V}$ 的特征矢量 $\boldsymbol{F}=[\boldsymbol{v}_1,\boldsymbol{v}_2,\cdots,\boldsymbol{v}_N]$。可见，H-LBP 算子是将那些灰度值高于中心像素的总数量视为识别阈值，有效消除噪声。借助相似度距离与 Heaviside 函数充分描述不同位置像素点的差异，显著地体现了不同位置的像素点之间的空间关系，提高 H-LBP 算子对图像空间分布的描述能力。对那些与中心像素差异更大的局部像素，赋予其更大的权重，有效突出此类像素的结构纹理信息，并根据(12)式的阈值 T，将差异度 f_i-f_{c} 超过 T 值的像素点丢弃，从而使提取特征的稳健性更高。

2.4 基于压缩感知的哈希序列压缩与量化

为了压缩特征矢量 $\boldsymbol{F}=[\boldsymbol{v}_1,\boldsymbol{v}_2,\cdots,\boldsymbol{v}_N]$ 的空间维数，引入压缩感知(CS)[12]，对 $\boldsymbol{F}$ 进行降维处理：

$$\boldsymbol{v}_i'=\varphi\boldsymbol{v}_i,\tag{20}$$

式中 $\boldsymbol{v}_i'$ 为压缩后的特征矢量，φ 为压缩感知的稀疏基。

将(20)式得到的 $\boldsymbol{v}_i'$ 进行组合，输出二次图像 $\boldsymbol{V}$ 的紧凑哈希序列 $F'=\{\boldsymbol{v}_1',\boldsymbol{v}_2',\cdots,\boldsymbol{v}_N'\}$。设计相应的量化规则，对 F' 完成量化处理。利用 $F'=\{\boldsymbol{v}_1',\boldsymbol{v}_2',\cdots,\boldsymbol{v}_N'\}$ 中的元素 $\boldsymbol{v}_i'$ 来计算其差 δ_i，再计算所有 δ_i 的均值 t，

$$t=(\delta_1+\delta_2+\cdots+\delta_i)/N,\tag{21}$$

式中 N 为哈希序列长度。

利用均值 t 来设计量化机制：当 $\boldsymbol{v}_i'\geqslant t$，则 $B_i=1$；反之，$B_i=0$。利用该量化机制处理特征矢量 $F'=\{\boldsymbol{v}_1',\boldsymbol{v}_2',\cdots,\boldsymbol{v}_N'\}$，输出比特数组 $B=\{B_1,B_2,\cdots,B_N\}$。

2.5 基于动态更新变换与汉明距离的哈希认证

未经加密处理的哈希序列的安全性较低，为此采用混沌变换理论[13]设计动态更新变换机制，对比特哈希进行加密。Logistic 映射[13]具有良好的混沌性能，其模型为

$$x_{k+1}=\lambda x_k(1-x_k),\ \lambda\in[0,4], \tag{22}$$

式中 λ 为混沌控制参数，x_k 为系统变量。

但是，Logistic 映射的混沌区域较窄，在区间 [0,4] 上存在非混沌区域，如图 5(a)所示。为此，基于分数阶理论[14]，通过设置 2 个分数阶参数 v 和 a，将(22)式演变为

$$\begin{cases}\Delta_a^v x_{k+1}=\lambda x_{k+u-1}(1-x_{k+u-1})\\ x_a=x_0\end{cases}, \tag{23}$$

式中 Δ 为分阶符号，x_a 是系统变量，v 和 a 均为整数。

利用 2 个分数阶参数 v 和 a，有效地扩大了 Logistic 映射的混淆混沌区域，如图 5(b)所示，增强了混沌序列的随机性。对比图 5(a)与图 5(b)可知，(23)式的混沌区域更大，在区间(0.243,4]内，都呈现出混沌行为。

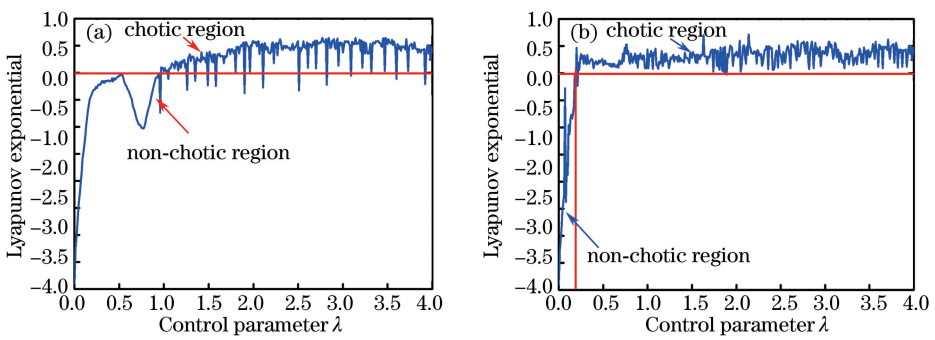

图 5 不同 Logistic 映射的混沌性能。(a) Logistic 映射；(b)分数阶 Logistic 映射

Fig. 5 Chaotic performance of different Logistic maps. (a) Logistic map; (b) fractional order Logistic map

单纯利用(23)式的迭代随机数组存在周期性，削弱了哈希序列的安全性，故构建了动态更新函数

$$\lambda_{i+1}=4-\lambda_i, \tag{24}$$

式中 λ_i 是第 i 次加密对应的初值。

例如，对于(23)式的第 1 次迭代，利用 λ_1 与 x_0 来完成，得到输出值 x_1。对于第 2 次迭代，利用(24)式更新 λ 值，利用 λ_2 与 x_0 完成计算，输出 x_2。以此类推，由于哈希序列长度为 N，故需对(23)式迭代 N 次，利用 λ_N 与 x_0 来完成计算，获取 x_N。依据迭代顺序，将(23)式的输出值 x_i 进行组合，获取随机序列 $X=\{x_1,x_2,\cdots,x_N\}$。再对 $X=\{x_1,x_2,\cdots,x_N\}$ 进行升序排列，输出新数组 $X'=\{x_1',x_2',\cdots,x_N'\}$。随后，在 $\{x_i\}$ 中找出 $\{x_i'\}$ 对应元素的位置，形成加密集合 $\{Y_i\}$：

$$x_i=x'_{Y_i}。 \tag{25}$$

利用加密集合 $\{Y_i\}$ 对比特数组 $B=\{B_1,B_2,\cdots,B_N\}$ 进行位置混淆，输出图像哈希序列 $H=\{H_1,H_2,\cdots,H_N\}$。由于使用了加密技术来置乱哈希序列的位置，进一步提高了哈希序列的敏感性，从而改善了哈希序列的安全性。攻击者若不知道密钥，哪怕对图像实施极其微小的篡改，所输出的哈希序列与初始图像的哈希序列是截然不同的。

若输入图像为 I_0，接收图像为 I_1，获取 I_0 和 I_1 对应的哈希序列分别为 $H_0=\{H_1^0,H_2^0,\cdots,H_N^0\}$ 和 $H_1=\{H_1^1,H_2^1,\cdots,H_N^1\}$，再利用汉明距离 D 来计算 H_0 与 H_1 的相似度[14]

$$D=d(H_0,H_1)=\frac{1}{L-1}\sum_{i=1}^{L-1}(H_0\oplus H_1), \tag{26}$$

式中⊕为异或运算。

对比汉明距离 D 与用户阈值 W，若 $D\leqslant W$，则把图像决策为真实图像；反之，把图像判别为篡改图像。

3 实验结果与分析

借助 Matlab 平台，在 UCID 图像库[15]中验证所提图像哈希算法的性能。UCID 库中含有 1338 幅彩图，尺寸主要分为两类：512 pixel×384 pixel 和 384 pixel×512 pixel。同时，为了体现所提算法的优异性，将文献[5]和文献[16]中两种哈希技术视为对照组。另外，利用 ROC 曲线来量化三种哈希技术的稳健性[17]，其计算函数为

$$\begin{cases} P_{\mathrm{TPR}}(\lambda)=\dfrac{n_1(D_1<\lambda)}{M_1} \\ P_{\mathrm{FPR}}(\lambda)=\dfrac{n_2(D_1<\lambda)}{M_2} \end{cases}, \tag{27}$$

式中 P_{TPR} 为正确识别率，P_{FPR} 为虚警率，n_1 为准确识别图像数量，n_2 为错误识别图像数量，M_1 和 M_2 分别为视觉相同和差异图像数量。

利用所提哈希技术识别图像时，用户阈值 W 对其稳健性及检测性能的影响较大。为了提高本文哈希算法的敏感性与稳健性，需对 W 进行优化。其余参数设置为：$\lambda_0=1.56$，$\sigma=0.5$，环形数量 $n=6$，LBP 半径 $R=1$，$P=8$；分数阶 $u=1$，$l=2$。

3.1 认证阈值 W 的优化

在 UCID 库[15]中任意择取 100 幅图像作为初始图像，根据表 1 中的攻击内容对每幅图像完成操作。

表 1 不同强度下的图像攻击形式

Table 1 Image attack patterns under different intensities

Operation type	Parameter
Salt and pepper noise	0.01, 0.04, 0.20, 0.80
Brightness adjustment	0.4, 0.8, 1.8, 2.0
Gamma correction	0.2, 0.4, 0.6, 1.0
JPEG compression	10, 50, 70, 100
Rotation	10°, 30°, 90°, 120°
Scaling	0.5, 0.7, 1.4, 1.5

图 6 所示为不同图像的汉明距离 D 对应的频数分布状况。由图 6 可知，对于存在视觉差异的两幅图像，当 $D\geqslant0.56$ 时，图像的频数分布变化剧烈；对于视觉相同的两幅图像，当 $D\leqslant0.56$ 时，图像的频数分布变化较大。因此，设置阈值 $W=0.56$。

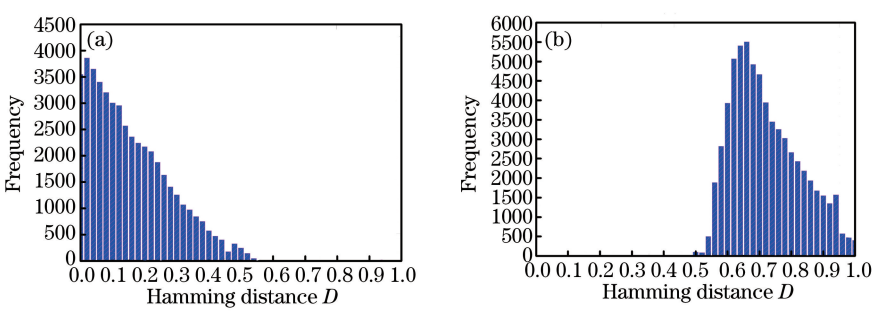

图 6 用户阈值 W 的优化测试。(a)视觉相同图像；(b)视觉差异图像

Fig. 6 Optimization of user threshold W. (a) Visually identical images; (b) visually different images

3.2 哈希算法的稳健性测试

理想的哈希技术通常需满足较强的感知稳健性、敏感性与安全性[16]。利用优化阈值 $W=0.56$ 进行后续实验。

3.2.1 感知稳健性测试

在 UCID 库中任意挑选 4 幅彩图作为测试样本，如图 7 所示，并对测试样本施加表 1 中的攻击内容。同时，根据(26)式计算汉明距离，测试数据如图 8 所示。根据图 8 中数据可知，面对 6 种攻击手段，所提哈希技

术的汉明距离 D 均小于 0.56。这表明该技术具备良好的感知稳健性，这是因为其考虑了中心像素与其邻域像素的差异，将 Heaviside 函数嵌入传统的局部二值模式中，设计了描述能力较强的 H-LBP 算子，准确提取稳健特征，并设计动态更新变换机制，对过渡哈希数组进行加密，使其对旋转、噪声、缩放等攻击手段具有较高的检测能力。

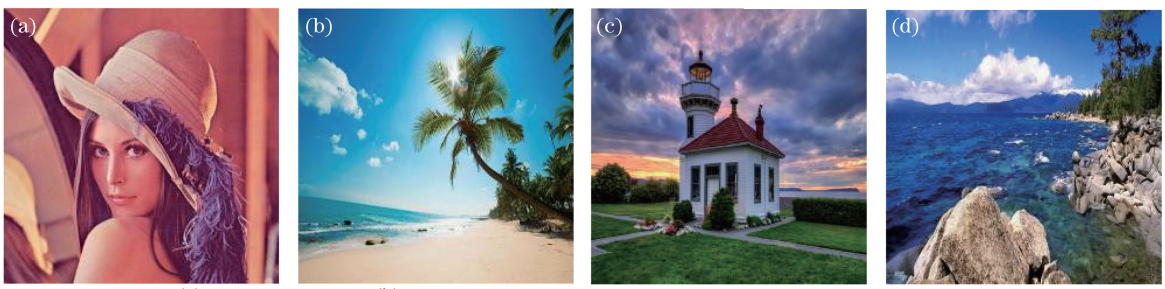

图 7 测试图像。(a) Lena；(b) sea；(c) house；(d) landscape

Fig. 7 Images for test. (a) Lena; (b) sea; (c) house; (d) landscape

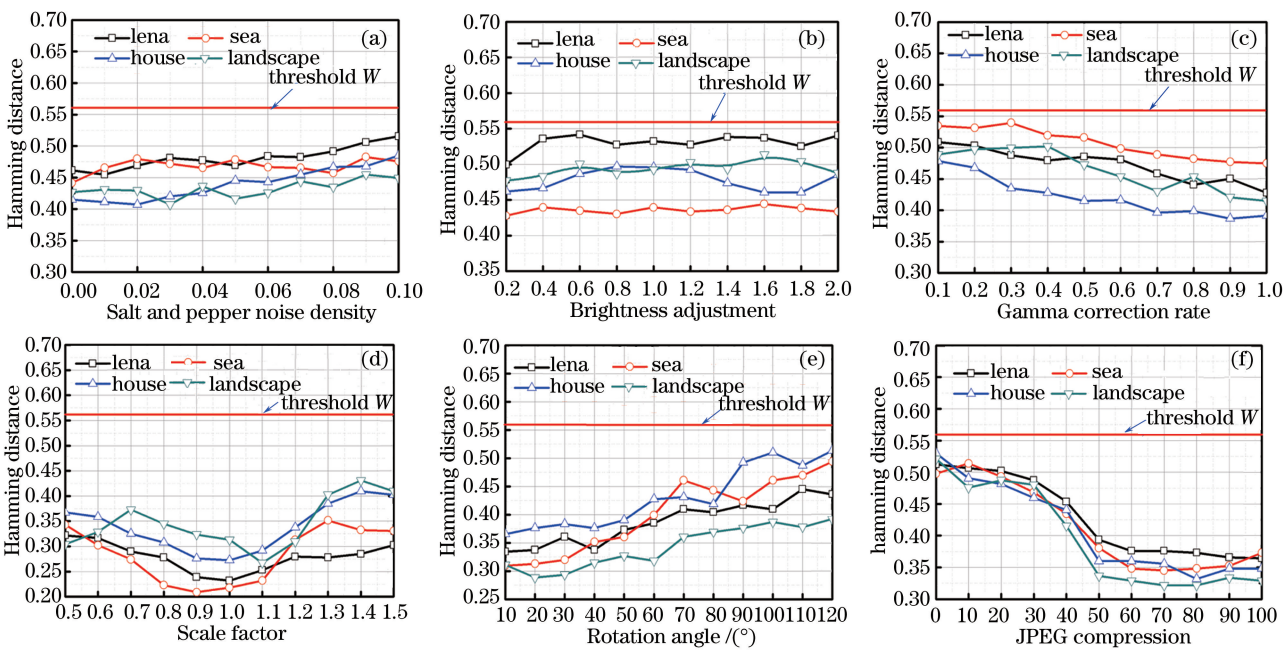

图 8 本文算法的感知稳健性测试结果。(a)椒盐噪声；(b)亮度调整；(c)伽马校正；(d)缩放尺度；(e)旋转角度攻击；(f) JPEG 压缩攻击

Fig. 8 Robustness test results obtained by the proposed algorithm. (a) Salt and pepper noise; (b) brightness adjustment; (c) Gamma correction; (d) scaling; (e) rotation angle attack; (f) JPEG compression attack

3.2.2 敏感性测试

当图像内容遭遇攻击时，其相应的哈希序列会出现剧烈变化，与初始图像的哈希序列存在巨大差异[3]。为了测试所提哈希技术的敏感性，对初始图像进行移动-复制、尺度缩放等攻击，使其形成视觉差异图像，测试结果如图 9 所示。此时，若汉明距离大于 0.56，则表明所提哈希技术对这些篡改检测成功，具备较强的敏感性。根据所提哈希序列生成过程，利用(26)式计算攻击前后图像的汉明距离，结果如表 2 所示。根据表 2 中数据可知，经过伪造后的图像的汉明距离都大于阈值 0.56，这表明所提哈希技术具备理想的敏感性。

表 2 篡改图像的归一化汉明距离

Table 2 Normalized Hamming distance of tampered image

Image No.	Fig. 8(b)	Fig. 8(c)	Fig. 8(d)	Fig. 8(e)
Hamming distance	0.5981	0.6125	0.6175	0.8034

图 9 不同攻击手段形成的伪造图像。(a)初始图像;(b)移动复制攻击;(c)伪造目标尺寸放大 25%;(d)伪造目标尺寸放大 60%;(e)缩放、旋转和移动-复制组合攻击

Fig. 9 Forged images from different attacks. (a) Original image; (b) moving-copying attack; (c) forged target size magnification by 25%; (d) forged target size magnification by 60%; (e) combination attack of scaling, rotation, and moving-copying

3.2.3 安全性测试

优异的哈希算法除了具备理想的感知稳健性外,还必须具备足够高的安全性[6]。为此,测试 2400 对错误密钥所对应的汉明距离,结果如图 10 所示。由图可知,对于这些错误密钥,汉明距离均高于 0.56。所提技术生成的哈希序列长度为 400,如果篡改者无法获知加密密钥和整个哈希过程,则篡改者获取相同哈希序列的概率为 $(1/2)^{400}$。因此,想要篡改该哈希序列的难度是非常大的。这表明该技术具备较高的安全性。

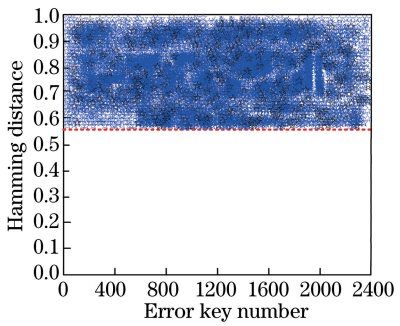

图 10 哈希算法的安全性测试

Fig. 10 Security test of hash algorithm

3.2.4 不同哈希算法的稳健性对比测试

在 UCID 库[15]中抽取 500 幅图像来生成相似图像对与不同图像对,以测试本文算法、文献[16]算法和文献[5]算法的 ROC 特性曲线,结果如图 11 所示。根据图中的 ROC 特性曲线可知,所提哈希技术的 ROC 特性更为理想,面对旋转攻击时,当 $P_{\mathrm{FPR}}=0$ 时 $P_{\mathrm{TPR}}=0.93$,当 $P_{\mathrm{FPR}}=0.2$ 时 $P_{\mathrm{TPR}}=0.99$。而文献[16]与文献[5]算法的 ROC 特性不佳,均低于本文算法,其中,文献[5]算法对旋转篡改的稳健性最差,当 $P_{\mathrm{FPR}}=0$ 时 $P_{\mathrm{TPR}}=0.71$,当 $P_{\mathrm{FPR}}=0.2$ 时 $P_{\mathrm{TPR}}=0.915$,而文献[16]算法的稳健性略低于本文算法,但其对亮度攻击的稳健性最为理想。原因是本文所提哈希算法通过插值机制来规范图像,并利用 Ring 分割对预处理结果进行分环,增强哈希序列对缩放、旋转的稳健性,且考虑了中心像素与其周围像素的差异,将 Heaviside 函数嵌入 LBP 中,设计了描述能力较强的 H-LBP 算子,提取图像的抗旋转稳健特征,提高哈希算法的抗噪与 JPEG 压缩能力。文献[16]算法的稳健性也较高,对旋转、缩放等攻击同样具备与本文技术相接近的水平,但是文献[16]算法采用了扩展的 LBP 算子,忽略了中心像素与其周围像素的差异,对特征的描述能力较弱,导致整体感知稳健性略低于本文哈希算法。但是文献[16]算法使用了 Gabor 滤波,且将图像的 RGB 转换为 YCbCr 空间,有效改善其对亮度调整的稳健性。文献[5]算法仅利用颜色矢量与边缘等特质来生成哈希序列,忽略了图像结构与纹理信息,降低了算法的稳健性,无法准确描述其抗旋转特征。

3.2.5 不同哈希算法效率对比测试

为了反映本文算法、文献[16]算法与文献[5]算法的哈希生成效率,借助 Matlab 进行测试,仿真条件为:睿酷 3.5 Hz 双核 CPU,4 GB 内存,测试结果见表 3。根据表中数据可知,因本文算法与文献[16]算法均对高维特征矢量进行了数据压缩,二者的效率较高。但是文献[16]算法利用了数据投影降维机制,需要用拉格朗日乘数法求解特征值,且需要不断迭代来形成加权邻接图。本文算法仅利用压缩感知,复杂度小于文献[16]算法。文献[5]算法忽略了哈希生成效率,缺乏特征降维机制,因此哈希生成耗时最高,约为 0.92 s。

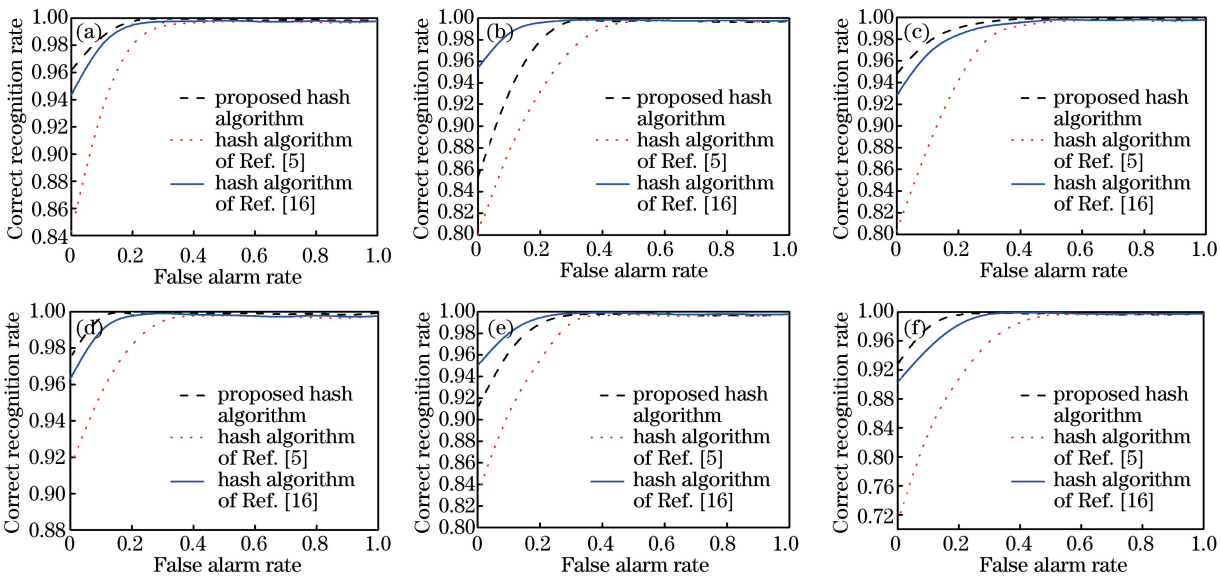

图 11 三种算法的稳健性测试结果。(a)椒盐噪声;(b)亮度调整;(c)伽马校正;(d)缩放尺度;(e) JPEG 压缩;(f)旋转攻击

Fig. 11 Robustness test results of the three algorithms. (a) Salt and pepper noise; (b) brightness adjustment; (c) Gamma correction; (d) scaling; (e) JPEG compression; (f) rotation attack

表 3 三种算法的哈希性能与效率测试

Table 3 Hash performance and efficiency test of three algorithms

Algorithm	Proposed	Ref. [16]	Ref. [5]
Hash length /bit	400	500	700
Hash generation time /s	0.53	0.64	0.92

4 结 论

为了兼顾哈希序列的稳健性与生成效率,提出了基于 H-LBP 算子与动态更新变换的紧凑图像哈希算法。利用 Ring 分割与插值机制,提高该算法对缩放与旋转的识别能力。基于 LBP 算子,引入 Heaviside 函数,构建 H-LBP 算子,增强算法对抗旋转特征与噪声的稳健性。利用压缩感知对高维特征矢量进行数据压缩,获取紧凑的哈希序列,并设计了动态更新变换机制,对过渡哈希数组进行加密,有效提高了哈希序列的安全性。优化用户阈值,利用汉明距离,完成图像内容识别。实验数据验证了所提哈希算法的有效性与优异性。

参 考 文 献

[1] Qin C, Chen X Q, Ye D P, *et al*. A novel image hashing scheme with perceptual robustness using block truncation coding[J]. Information Sciences, 2016, 361: 84-89.

[2] Davarzani R, Mozaffari S, Yaghmaie K. Perceptual image hashing using center-symmetric local binary patterns[J]. Multimedia Tools and Applications, 2016, 75(8): 4639-4667.

[3] Feng He, Chang Guoquan, Guo Xiaobo. Hash algorithm for color image based on super-complex Fourier transform coupled with positon permutation[J/OL]. Journal of Frontiers of Computer Science and Technology: 1-11. (2016-12-16) [2017-09-21]. http://kns.cnki.net/kcms/detail/11.5602.TP.20161216.1049.002.html.
冯贺, 常国权, 郭晓波. 超复数 Fourier 变换耦合位置扰乱的彩色图像哈希算法[J/OL]. 计算机科学与探索: 1-11. (2016-12-16) [2017-09-21]. http://kns.cnki.net/kcms/detail/11.5602.TP.20161216.1049.002.html.

[4] Jiang Cuiling, Lin Jiajun. Robust image hashing based on genetic algorithm and BP network[J]. Journal of Applied Science, 2016, 34(5): 537-546.
蒋翠玲, 林家骏. 一种基于遗传算法和 BP 网络的鲁棒图像哈希方法[J]. 应用科学学报, 2016, 34(5): 537-546.

[5] Tang Z J, Huang L Y, Zhang X Q, *et al*. Robust image hashing based on color vector angle and canny operator[J].

International Journal of Electronics and Communications, 2016, 70(6): 833-841.

[6] Vadlamudi L N, Vaddella R P V, Devara V. Robust hash generation technique for content-based image authentication using histogram[J]. Multimedia Tools and Applications, 2016, 75(11): 6585-6604.

[7] Ruan Linlin. Image hashing algorithms based on locally linear embedding and locality preserving projection[D]. Guilin: Guangxi Normal University, 2015: 24-26.
阮林林. 基于局部线性嵌入和局部保持投影的图像哈希算法[D]. 桂林: 广西师范大学, 2015: 24-26.

[8] Tang Z J, Zhang X Q, Zhang S C. Robust perceptual image hashing based on ring partition and NMF[J]. IEEE Transactions on Knowledge & Data Engineering, 2014, 26(3): 711-724.

[9] Ylioinas J, Poh N, Holappa J, *et al*. Data-driven techniques for smoothing histograms of local binary patterns[J]. Pattern Recognition, 2016, 60: 734-747.

[10] Zhang Zhifeng, Pei Zhili. Image forgery detection algorithm based on fuzzy local binary pattern operator[J]. Computer Engineering and Design, 2015, 36 (12): 3284-3290, 3296.
张智丰, 裴志利. 基于模糊局部二值模式算子的图像伪造检测[J]. 计算机工程与设计, 2015, 36(12): 3284-3290, 3296.

[11] Wang L, He L, Mishra A, *et al*. Active contours driven by local Gaussian distribution fitting energy[J]. Signal Processing, 2009, 89(12): 2435-2447.

[12] Qureshi M A, Deriche M. A new wavelet based efficient image compression algorithm using compressive sensing[J]. Multimedia Tools and Applications, 2016, 75(12): 6737-6754.

[13] Sun Qian, Hu Su. Improved cat map and chaotic system based fast color image encryption algorithm[J]. Application Research of Computer, 2017, 34(1): 233-237, 255.
孙倩, 胡苏. 基于改进 cat 映射与混沌系统的彩色图像快速加密算法[J]. 计算机应用研究, 2017, 34(1): 233-237, 255.

[14] Jiang Y Y, Wang Y R, Luo H. Image denoising method for unknown noise based on 2-D FWT with optimal fractional order[J]. Journal of Computers, 2014, 9(2): 412-419.

[15] Schaefer G, Stich M. UCID: an uncompressed colour image database[C]. SPIE, 2003, 5307: 472-480.

[16] Wang Yanchao, Guo Jingbo, Zhou Liyan. Image hash algorithm based on data dimension reduction and symmetric binary pattern[J]. Laser & Optoelectronics Progress, 2017, 54(2): 021004.
王彦超, 郭静博, 周丽宴. 基于数据降维与对称二值模式的图像 Hash 算法[J]. 激光与光电子学进展, 2017, 54(2): 021004.

[17] Li Xinwei, Li Leida. Robust image hashing based on polar harmonic transform[J]. Computer Simulation, 2014, 31 (5): 293-296.
李新伟, 李雷达. 基于极谐变换的鲁棒图像哈希算法[J]. 计算机仿真, 2014, 31(5): 293-296.