计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2020,29(12):117−125 [doi: 10.15888/j.cnki.csa.007721]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于自适应提点鲁棒定位的图像复制粘贴篡改检测①

于　亮, 杨红颖

(辽宁师范大学 计算机与信息技术学院, 大连 116029)
通讯作者: 于　亮, E-mail: yl20200220@foxmail.com

摘　要: 复制-粘贴篡改检测 (Copy-Move Forgery Detection, CMFD) 是数字图像篡改的一种常见方式, 近年来已成为多媒体取证领域一个重要的研究方向. 本文提出一种鲁棒的复制-粘贴篡改检测算法, 基于构造波动函数自适应获取阈值的方法均匀提取图像特征点, 可在篡改区域小或平滑的情况下进行鲁棒检测. 引入 DBQ-LSH 匹配算法进行特征匹配, 降低了时间复杂度. 提出基于不变矩 LBP 图像的定位方法, 在图像受到噪声攻击和 JPEG 压缩攻击下能精准定位篡改位置. 实验结果表明, 该算法具有优良的检测正确率 (图像级) 和检测精度 (像素级).

关键词: 数字图像取证; 复制-粘贴篡改; 局部敏感哈希; 自适应阈值; 局部二值模式

引用格式: 于亮,杨红颖.基于自适应提点鲁棒定位的图像复制粘贴篡改检测.计算机系统应用,2020,29(12):117–125. http://www.c-s-a.org.cn/1003-3254/7721.html

Copy-Move Forgery Detection Based on Adaptive Keypoints Extraction and Robust Localization

YU Liang, YANG Hong-Ying

(School of Computer and Information Technology, Liaoning Normal University, Dalian 116029, China)

Abstract: Copy move forgery is a common way of digital image tampering, which has become an important research direction in the field of multimedia forensics in recent years. We propose a robust copy-move forgery detection algorithm. We construct wave function to extract keypoint evenly in the image, which can extract enough keypoint even in small or smooth areas. We introduce DBQ-LSH for feature matching, which greatly reduces time consumption. We propose a novel approach which use invariant moment LBP image to locate forged areas, even if the image is seriously attacked, the algorithm detection accuracy is still excellent. A large number of experimental results verify the reliability of proposed algorithm, in terms of validity and accuracy.

Key words: digital image forensic; copy-move forgery; local sensitive hash; adaptive threshold; local binary pattern

数字图像是人们日常生活中获取信息的重要渠道, 在新闻、学术、法庭等领域有着重要的应用. 然而随着智能手机等电子设备的日益普及, 和图像处理软件的普遍应用, 越来越多的图像被篡改成混淆视觉的伪造图像. 图像篡改事件层出不穷, 严重影响着人们的生活. 图像复制-粘贴篡改是最普遍的篡改方式. 复制-粘贴篡改 (CMF) 是将一幅图像的一部分复制并粘贴到同一幅图像的另一个位置, 覆盖图像的内容以达到混淆视觉并隐藏信息的作用. 复制-粘贴篡改检测 (CMFD) 过程可分为 4 步: (1) 预处理: 将 RGB 图像转换成灰度图像, 将图像分割成块等; (2) 特征提取: 提取图像的局部或全局特征; (3) 特征匹配: 对已有的图像特征进行匹配; (4) 后处理: 对匹配结果进一步处理, 剔除异常值、定位篡改位置等. 主流的 CMFD 算法分为两方向: (1) 基

① 收稿时间: 2020-05-13; 修改时间: 2020-06-10; 采用时间: 2020-06-19; csa 在线出版时间: 2020-11-30

于图像块的算法; (2) 基于图像高熵位置的特征点算法. 基于图像块的算法是将图像分为重叠块或非重叠块, 利用图像变换、图像不变矩、颜色等方法提取图像块特征, 匹配特征并得到匹配对, 使用一系列后处理操作定位篡改区域. 基于图像块的算法时间复杂度较高, 在图像遭受缩放等攻击下算法鲁棒性较差. 基于特征点的算法首先提取图像的高熵区域作为特征点, 描述特征点或其对应区域的特征并匹配, 得到疑似匹配点对. 通过聚类或分割等算法, 筛除误匹配点对, 最后定位篡改区域. 基于特征点的算法在篡改区域为平滑区域或小区域时检测效果较差. 此外, 已有算法对受到攻击的图像 (如白噪声攻击和 JPEG 压缩攻击) 的检测效果较差.

为解决上述问题, 本文提出一种鲁棒的复制-粘贴篡改检测算法. 在特征点提取阶段, 使用尺度不变的 SURF[1] 特征点和特征, 引入波动函数的思想, 提出一种自适应的特征点提取方法, 该方法能够在图像的平滑区域或小区域提取足够多的特征点. 在特征匹配阶段, 引入了双比特迭代量化局部敏感哈希 (DBQ-LSH)[2,3] 算法. 在后处理阶段, 提出了一种不变矩 LBP[4] 定位算法, 即使图像受到严重攻击也能对其检测并精准定位篡改区域.

本文贡献如下:

(1) 提出一种自适应提点算法, 在图像小区域或平滑区域能提取足够的特征点, 使特征点分布更均匀.

(2) 引入 DBQ-LSH 算法, 降低了匹配成本.

(3) 提出一种鲁棒的后处理算法, 对受到攻击的图像, 检测结果更精准.

本文在第 1 节介绍了图像复制-粘贴篡改检测. 在第 2 节详细描述所提出的复制-粘贴篡改检测算法. 在第 3 节通过一系列的仿真实验和对比实验, 验证本方案的有效性和鲁棒性.

1 图像复制-粘贴篡改检测

主流图像复制-粘贴篡改检测算法可分为两方向: 基于图像块的方法和基于特征点的方法. 它们都试图提取局部图像块或关键点的图像特征, 并评估不同图像区域之间特征的相似性.

1.1 基于图像块的方法

基于块的复制粘贴篡改检测算法将宿主图像分割为重叠或非重叠图像块, 计算并匹配每个图像块特征, 得到疑似匹配对. Fridrich 等[5] 提出了一种基于块的图像复制粘贴篡改检测算法, 将图像分割成重叠子块, 引入离散余弦变换 (DCT) 特征对每个子块进行特征描述, 匹配所有特征得到可疑匹配对. 但该方法时间消耗大, 且无法检测旋转攻击伪造图像. 一些研究者利用旋转不变特征来解决这个问题. Ryu 等[6] 利用旋转不变的不变矩特征进行检测, 且对白噪声等攻击具有较强的鲁棒性. 为了提高匹配效率, Cozzolino 等[7,8] 提出了一种基于重叠图像块的 CMFD 方法, 使用 Zernike 矩 (ZM)、Fourier-Mellin 变换 (FMT) 等特征, 改进 Patch Match (PM) 近邻搜索算法并将其应用与篡改检测中, 大大降低了时间复杂度. Bi 等[9] 提出了一致敏感哈希 (CSH) 算法, 该算法集 PM 算法和局部敏感哈希 (LSH) 算法于一身, 比 PM 算法具有更快的处理速度和更高的精度. PM 算法和 CSH 算法需要多次迭代来传播和搜索匹配, 这也导致了时间消耗的极大增加, 基于此, Zhong 等[10] 使用极性余弦变换 (PCT) 来提取图像块特征, 将块特征转换为哈希特征, 提出了双程哈希特征表示和搜索算法, 无需迭代, 提高了检测性能和运算效率. 然而由于计算的复杂性, 基于变换或矩特征的方法的时间消耗大, 此外它们使用固定的图像块来计算特征, 其特征仅在在一定范围内只具有尺度不变性, 这使得基于块的篡改检测算法的鲁棒性较差.

1.2 基于特征点的方法

为了解决基于块的方法计算复杂度较高, 尺度不变性较差等问题, 部分学者提出了基于特征点的算法. 基于特征点的篡改检测算法识别宿主图像中的高熵区域 (特征点), 与基于块的篡改检测算法相比, 可以减少特征数目和计算时间. Amerini 等[11] 引入尺度不变特征变换 (SIFT) 特征, 提出了 g2NN 匹配算法, 利用层次聚类进行多目标篡改检测, 不仅能确定图像是否被篡改, 且能得到被篡改区域的几何变换, 但其在篡改区域较小或篡改区域发生在图像平滑区域时的检测效果较差. Zandi 等[12] 提出了一种适合于 CMFD 的特征点检测算法, 该算法可以将关键点均匀地分布在整个图像上, 利用 SLIC 对图像进行分割, 并根据对应子块的关键点数目过滤去除错误的匹配点对, 根据先验信息调整子块关键点的密度, 迭代定位篡改区域. Li 等[13] 提出了一种分层的特征点匹配策略, 利用颜色信息来

定位被篡改的区域. 基于特征点的算法除使用 SIFT 特征点外, 部分学者使用了加速鲁棒特征 (SURF) 算法[14,15]、Harris 算法[16]、KAZE 算法[17]、A-KAZE 算法[18] 等.

基于关键点的方法时间复杂度较低, 对缩放等攻击的鲁棒性强, 但对小区域复制粘贴或发生在图像平滑区域的复制粘贴篡改, 检测效果并不理想, 且抗噪声攻击、JPEG 压缩攻击能力有待提高.

2 本文算法

为了改善图像小区域和平滑区域提取特征点数量不足, 受攻击图像的检测精度低等问题, 本文提出一种自适应提取特征点、快速匹配和鲁棒篡改区域定位的复制移动篡改检测算法. 图 1 表明本算法结构框架. 第一步将图像分割成不重叠的块, 构造波动函数, 根据波动函数在图像内均匀提取 SURF 特征点. 第二步使用 SURF 特征进行特征表示. 第三步引入双比特量化局部敏感哈希 (DBQ-LSH) 快速匹配特征. 第四步去除孤立匹配, 利用 k-多均值聚类方法对匹配点对进行聚类, 将灰度图像转化为不变矩 LBP 图像, 定位篡改区域.

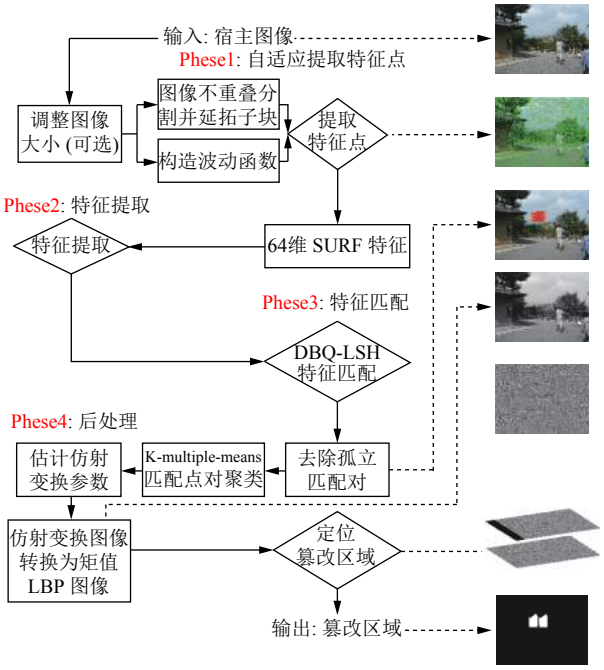


图 1 算法流程

2.1 基于波动函数的自适应提取图像特征点

基于特征点的篡改检测算法主要问题为在复制-粘贴区域较小或其发生在平滑区域时的检测效果较差. 余江[19] 提出了一种构造波动函数并对图像块分类, 选择复杂纹理块进行图像隐写的新方法. 本文据此提出了一种基于波动函数确定提点阈值的自适应特征点提取方法. 本文提点算法如算法 1 所示.

算法 1. 自适应提取特征点

输入: 宿主图像
输出: 图像均匀分布的特征点

1. 根据条件对图像放大处理
2. 图像不重叠分块并延拓子块
3. 构造波动函数
4. 根据波动函数确定阈值并提取 SURF 特征点及特征
5. 将子块特征点集合并筛除重复的特征点

2.1.1 SURF

SURF[1] 基于尺度空间, 对平移、缩放、旋转等具有很强的不变性, 并在一定范围内对仿射变换和亮度变化也具有鲁棒性. 使用盒型过滤器, 通过增加窗口大小来构建不同比例的图像金字塔. 在 Hessian 矩阵中使用积分图像来获得两步特征, 在保持较高描述能力的同时提高了计算效率. 例如, 在图像 I 中, 一个点是 $X=(x,y)$, X 在σ尺度上的 Hessian 矩阵是以下定义:

$$H(x,\sigma)=\begin{bmatrix} L_{xx}(x,\sigma) & L_{xy}(x,\sigma) \\ L_{xy}(x,\sigma) & L_{yy}(x,\sigma) \end{bmatrix} \tag{1}$$

其中, $L_{xx}(x,\sigma)$, $L_{xy}(x,\sigma)$, $L_{yy}(x,\sigma)$是高斯滤波器在 xx、xy、yy 方向的二阶偏导数与图像 I 在点 X 处的卷积, 为简化运算, 使用$D_{xx}(x,\sigma)$, $D_{xy}(x,\sigma)$, $D_{yy}(x,\sigma)$分别代替$L_{xx}(x,\sigma)$, $L_{xy}(x,\sigma)$, $L_{yy}(x,\sigma)$, 为便于计算, Hessian 矩阵可近似表示为式 (2):

$$\det(H)=D_{xx}D_{yy}-(wD_{xy})^2 \tag{2}$$

一般设置 4 组 6 层尺度空间. 与该尺度空间的上下层的尺度空间中的 26 个点相比, 如果该像素是极值, 则该像素被确定为特征点. 计算了特征点圆邻域的 Haar 小波响应. 找出特征点的主要方向. 沿着特征点主方向的邻域选取 4×4×4 矩形小区域, 计算每个小区域的 Haar 小波响应, 得到每个区域的 4 维特征向量. 一个特征点有 64 维特征向量作为 SURF 特征的描述符.

2.1.2 图像不重叠分块并延拓

我们将图像分为不重叠的块, 设图像像素值为$M\times N$, 将图像分为$m_0\times n_0$(设置 $m_0=n_0=30$) 个子块, 子块大小为 $(M/m_0)\times(N/n_0)$. 由于子块之间的连接处包含大量信息, 为将子块连接处覆盖, 将每个子块的长度和宽度分别扩

展 L_1(设置 L_1=20) 个像素, 此时, 子块大小为$(M/m_0+L_1)\times(N/n_0+L_1)$.

2.1.3 构造波动函数

构造波动函数, 计算各子块的波动函数值. 根据波函数的值确定特征点提取的阈值, 从而得到一致的特征点提取结果. 设子块的长度和宽度分别为 r、s, 定义如下:

$$r=M/m_0+L_1,\ s=N/n_0+L_1 \tag{3}$$

$x_{i,j}$ 是子块中的任一像素, x_{mea} 为该子块像素值的中值, P 是该子块像素点的数量. 则波动函数F_w定义如下:

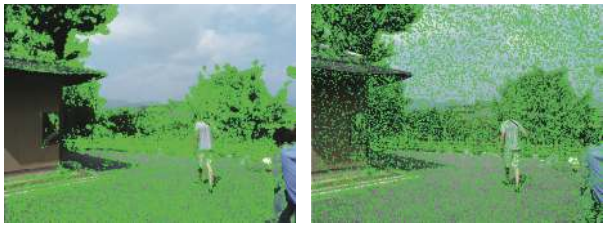
$$F_w=\exp\left(\sum_{i=1}^{r}\sum_{j=1}^{s}\left|\frac{x_{\text{mea}}-x_{i,j}}{100\times n}\right|\right)-1 \tag{4}$$

2.1.4 提取特征点及特征

设初始 SURF 特征点阈值为 $T_0(T_0=10)$, 每个子块的阈值为T_d. T_d与T_0之间的关系如下:

$$T_d=T_0\times F_w \tag{5}$$

根据每个子块的T_d阈值, 均匀地提取每个子块的特征点, 将结果汇总到原始图像中, 去除同一位置的冗余特征点, 即可得到在图像内均匀分布的特征点, 在我们得到特征点的同时得到其 64 维特征. 图 2 显示了我们的提点算法与原始特征点提取方法之间的对比结果. 如图 2 所示, 传统的方法无法在平滑区域提取足够的特征点, 而我们的方法可在平滑区域均匀提取特征点.



(a) 原始 SURF 算法的结果 (b) 本文方法的结果

图 2 特征点提取方法的比较

2.2 特征匹配

局部敏感哈希 (LSH) 是解决海量高维数据匹配问题的常用方法. 我们引入了双比特量化局部敏感哈希 (DBQ-LSH) 算法[2,3], 它比传统的 LSH 算法匹配结果更精准, 匹配效率更高. 特征匹配过程如算法 2 所示.

算法 2. 特征匹配

输入: 特征点集合 X 与特征向量集合 F

输出: 匹配点对集合 P

1. 对特征向量集$F=\{f_1,f_2,\cdots,f_u\}$迭代量化分桶

2. 对任意特征向量 f_i, 找到其对应的桶, 并从对应桶内选取其 K 个最近邻特征向量, 组成 f_i 的最近邻集合 D_i

3. 对任意$f_i\in D_i$, 按相似度阈值T_{sml}和空间距离阈值T_{dist}进行筛选. 若$s_i=\{x_i,f_i\}$和$s_j=\{x_j,f_j\}$构成匹配, 则需满足条件$(x_i-x_j)^2\geq T_{\text{dist}}$且$(f_i-f_j)^2\leq T_{\text{sml}}$

2.2.1 双比特量化局部敏感哈希

主流的哈希方法通常采用两阶段策略. 在第一步中, 生成一些投影尺寸的实际值. 在第二步中, 将多个阈值量化为二进制码. 双比特量化 (DBQ) 从数据中自适应地获得最佳阈值, 并分别将 a 和 b 设置为左阈值和右阈值, $a<b$, S 表示投影上整个点集的实际值. S_1、S_2 和 S_3 是是由阈值 a, b 分割的子集, 如式 (6) 所示:

$$\begin{cases} S_1=\{x|-\infty<x\leq a,\ x\in S\} \\ S_2=\{x|a<x\leq b,\ x\in S\} \\ S_3=\{x|b<x<+\infty,\ x\in S\} \end{cases} \tag{6}$$

将 u_i 设为 S_i 中的平均值, 找到适当的 a 和 b 值以使 E 值最小化, 如式 (7) 所示:

$$E=\sum_{x\in S_1}(x-\mu_1)^2+\sum_{x\in S_2}(x-\mu_2)^2+\sum_{x\in S_3}(x-\mu_3)^2 \tag{7}$$

在得到 a 和 b 之后, 我们可以用它们将整个集合划分为 S_1、S_2 和 S_3, 然后量化子集中的点. 双比特量化有两个优点: 一是精度更好, 二是时间消耗更低. 采用双比特迭代量化, 大大提高了二进制码在哈希过程中的性能. 在复制-粘贴篡改检测中, 该方法匹配更精准, 时间复杂度更低.

2.2.2 快速特征匹配

在得到匹配点对集合和特征向量集合后, 利用 DBQ-LSH 进行特征匹配. 初步匹配对集为$X=\{x_i=(x,y)_i\}_{i=1}^{u}$, 特征向量集为$F=\{f_1,f_2,\cdots,f_u\}$, f_i与x_i一一对应. T_{sml}为相似阈值, T_{dist}为空间距离阈值. 首先, 我们将特征向量集 F 迭代量化到桶中, 相似的特征被散列到同一桶中. 对于一个特征向量f_i, 找到其对应的桶, 并从对应的桶中选择其 K 个最近邻特征向量. 若$(x_i-x_j)\geq T_{\text{dist}}$且$(f_i-f_j)\leq T_{\text{sml}}$, 则$x_i$与$x_j$匹配. 我们得到初步匹配对集合 P:

$$P=\{(x,x')_1,(x,x')_2,\cdots,(x,x')_n\} \tag{8}$$

本文使用 DBQ-LSH 与 g2NN 和 KD-Tree 匹配算法在匹配效果上进行对比, 实验主观结果见图 3. 表 1 为客观匹配结果, 本方法可得到更多匹配点对, 且时间复杂度更低.

(a) 原始图像 (b) 篡改图像 (c) 图像复制粘贴篡改区域

(d) KD-Tree 算法结果 (e) g2NN 算法结果 (f) 本算法结果

图 3 不同匹配算法的主观对比结果

表 1 不同匹配方法的对比结果

方法	特征提取		匹配		CPU总值
	#特征点	CPU	#matches	CPU	
KD-Tree			78	1044.0	1049.3
g2NN	25177	5.3	93	39.1	44.4
DBQ-LSH			122	22.7	28.0

2.3 后处理

有效的后处理算法不仅可以减少误检测率, 还可使篡改区域定位结果更精准. 后处理算法步骤见算法 3.

算法 3. 后处理

输入: 匹配点对集合 P 与宿主图像
输出: 标记篡改位置的二值图像

1. 筛除误匹配
2. 匹配点对聚类
3. RANSAC 估计仿射变换参数
4. 不变矩 LBP 图像定位篡改区域

2.3.1 筛除误匹配

特征匹配过程常常会得到部分误匹配点对. 由于篡改区域的连续性, 正确匹配点对具有连续性, 而误匹配点对通常是孤立存在的. 我们在提取特征点阶段将图像分割为不重叠的块, 在筛除误匹配对时利用各子块内特征点的数量性质, 如果子块中匹配点对数小于 K (K=6), 则该子块中的匹配对为误匹配.

2.3.2 匹配点对聚类

本方案使用聚类算法解决多目标篡改问题. 图像的原始区域和篡改区域或是相互接近的, 使用传统的聚类方法, 原始区域和篡改区域的特征点可能会聚为一类. 引入 K-multiple-means[20] 聚类算法, 该算法具有良好的聚类性能. K-multiple-means 定义了显示的目标函数. 在给定总聚类数 k 和总样本数 m 的情况下, 将 m 个样本和 n 个原始数据点自适应地划分为 k 类. K-multiple-means 解决了原始区域与篡改区域之间空间距离过近的问题.

2.3.3 RANSAC 估计仿射变换参数

原始区域与篡改区域存在一定的几何关系[11], 在得到聚类结果后, 利用 RANSAC 估计每一聚类结果的仿射变换参数. 变换矩阵如式 (9)、式 (10):

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = H \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{9}$$

$$H = \begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix} \tag{10}$$

其中, a_{11}, a_{12}, a_{21}, a_{22}表示旋转和各向异性缩放信息, t_x和t_y是平移因子.

2.3.4 定位篡改区域

LBP[4] 具有亮度和旋转不变性等优点. 本方案提出使用 LBP 图像计算图像相似度. 本文引入中值鲁棒扩

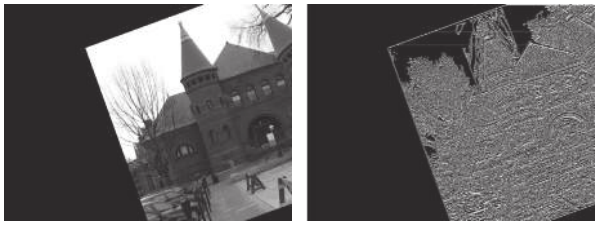
展局部二值模式 (MRELBP)[21], 其具有很强的噪声鲁棒性. 利用 Zernike 矩[22] 将像素值 MRELBP 扩展到矩值 MRELBP, 用 Zernike 矩代替图像像素值从而生成矩值 MRELBP 图像, 将空域 LBP 拓展到变换域 LBP, 以提高算法的鲁棒性.

篡改区域 (R_D) 像素与原始区域 (R_O) 像素之间存在一致的相关性[23], 我们称它为同一仿射变换参数 T, H 是它的矩阵形式.

$$R_D = HR_O, \quad R_O = H^{-1}R_D \tag{11}$$

对宿主图像进行仿射变换, 得到仿射变换图像 I_D, 计算得到 I_D 的 MRELBP 图像 (见图 4), 计算两幅 MRELBP 图像的相似度并得到篡改区域 (见图 5).

图 6 为本方案与基于图像灰度的相似度计算算法的对比结果. 所用宿主图像对篡改区域添加了白噪声, 如图示本算法对噪声攻击更具鲁棒性. 由于宿主图像的自相似性, 得到篡改区域定位结果的二值图像包含了部分未篡改区域, 这些区域很少且大部分为孤立存在, 本方案删除像素值低于整幅图像 0.05% 的区域, 并使用形态学方法填充孔洞.

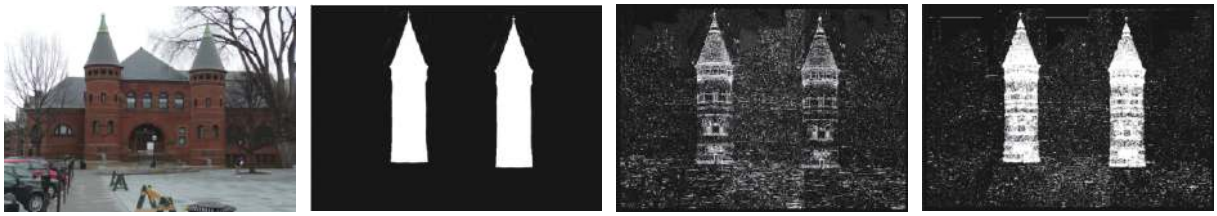


(a) 经仿射变换的灰度图像 (b) MRELBP 图像

图 4 仿射变换图像转换为 LBP 图像

(a) 灰度图像相似度计算 (b) 本文MRELBP图像相似度计算 (c) 计算相似度得到二值图像

图 5 图像相似度计算



(a) 宿主图像 (b) 复制粘贴篡改区域 (c) 使用常规灰度图像计算相似度的结果 (d) 本文使用矩值 LBP 图像计算相似度的结果

图 6 噪声攻击图像对比实例

3 实验结果

通过对该方案进行图像级和像素级实验并与其他算法进行比较, 验证了本算法在多个图像数据集的有效性. 进行图像级实验, 检测图像是否被篡改, 确定算法的有效性; 进行像素级实验, 对检测结果进行细化, 精准定位篡改区域. 实验在 Windows 7 64 位操作系统下进行, 处理器为 Inter(R) Corei7-4790 @3.6 GHz, 内存 16.0 GB, 仿真软件 Matlab R2016a 64 Bit.

3.1 评价标准与图像库

使用真阳性率 (*TPR*) 和假阳性率 (*FPR*) 评价算法的有效性. 高 *TPR* 与低 *FPR* 代表理想的结果. 分别定义:

$$TPR = \frac{TP}{TP+FN}, \quad FPR = \frac{FP}{TN+FP} \tag{12}$$

TPR 和 *FPR* 公式参数在不同情况下有不同含义[13]. 图像级实验, *TP* (True Positive), *TN* (True Negative), *FN* (False Negative), *FP* (False Positive) 分别表示正确检测到的篡改图像数, 正确检测到的真实图像数, 未检测到的篡改图像数和错误检测为篡改图像的真实图像数; 像素级实验, *TP*, *TN*, *FN*, *FP* 分别表示正确检测到的篡改像素数, 正确检测到的真实像素数, 未检测到的篡改像素数和错误检测为篡改像素的真实像素数. 本文在 *TPR* 和 *FPR* 的基础上用 F_1 值来表明综合实验结果. F_1 定义见式 (13), 我们使用 F-pixel 和 F-image 分别表示像素级和图像级的 F_1 分数.

$$F_1 = \frac{2TP}{2TP+FP+FN} \tag{13}$$

本方案选择 FAU[24]、GRIP[8] 和 MICC-F600[23] 3 个图像库验证算法的有效性.

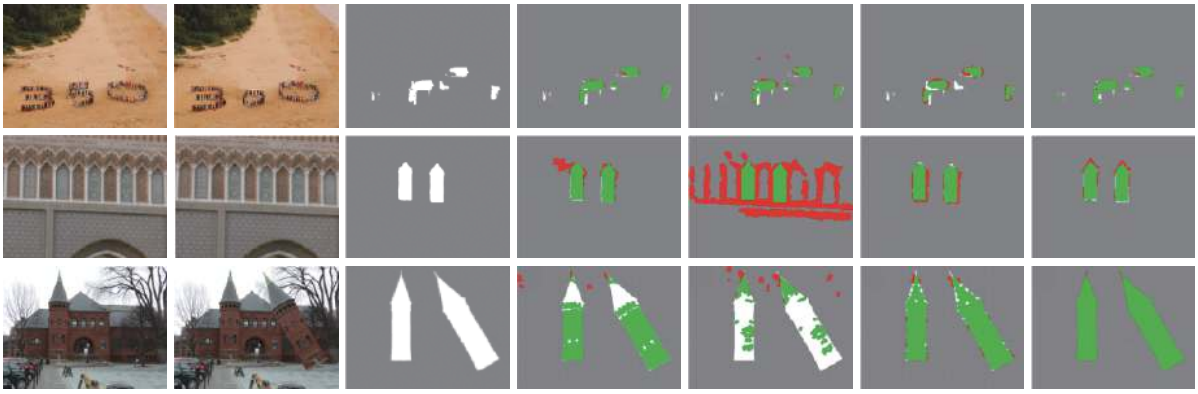
3.2 普通图像实验

我们对 3 个图像库[8,23,24] 进行了实验, 并与 3 种算法[8,12,13] 进行了对比, 3 种算法的代码在对应文章中获得, 实验结果见表 5. 我们得到满意的结果, 有着较高的 *TPR*、*FPR*、F_1 和 *FP* 值, 证明本算法不仅能检测出图像是否被篡改, 而且能更准确地定位篡改区域. 值得注意的是, 文献 [13] 算法的结果在某些情况下高于我们的算法, 然而这只是对普通篡改图像的结果. 本算法的优势是精准定位受攻击的图像, 在下一小节我们将证明本算法的优越性.

表 5 图像库对比实验结果

方法	FAU					GRIP					MICC-F600				
	TPR	*FPR*	F_1	*FP*	CPU	*TPR*	*FPR*	F_1	*FP*	CPU	*TPR*	*FPR*	F_1	*FP*	CPU
文献[8]	97.92	8.33	94.95	93.79	165.2	98.75	8.75	95.48	92.99	14.8	96.25	5.91	90.59	91.40	72.3
文献[12]	100	52.08	79.34	86.07	468.2	100	33.75	85.56	66.44	25.7	94.38	50.91	56.45	75.29	193.7
文献[13]	100	2.08	98.97	94.28	86.6	100	0	100	94.66	13.9	97.50	5.68	91.50	91.80	45.3
本文	100	4.17	97.95	96.38	81.5	100	2.50	98.76	96.60	27.36	98.75	5.23	92.67	94.11	88.7

图 7 中 3 个示例分别选自 FAU、GRIP、MICC-F600 图像库. 图 7(a) 第一幅图选自 FAU 图像库, 不仅有多目标克隆, 还包含小区域篡改. 第二幅图选自 GRIP 图像库, 其具有高度自相似性. 第三幅图选自 MICC-F600 图像库, 篡改区域包含旋转和缩放攻击. 图像中的绿色区域表示正确检测到的像素, 红色区域表示错误检测到的像素. 如图 7 所示, 本算法能够准确定位篡改区域. 在图 7(g) 中有部分红色区域, 因宿主图像是具有高度自相似性, 本算法在放大图像时, 增加了图像的原始相似区域, 使得误检测像素数量增多, 这也是未来改进方向之一.



(a) 原始图像 (b) 宿主图像 (c) 复制粘贴区域 (d) 文献[8]结果 (e) 文献[12]结果 (f) 文献[13]结果 (g) 本文结果

图 7 篡改检测主观结果对比实例

3.3 攻击图像实验

部分攻击图像库选自 FAU[24], 余下图像使用 FAU 生成攻击图像的方法对 FAU 基础数据集进行处理, 得到一系列不同类型的攻击图像. 以下为本实验所选攻击图像库:

(1) 缩放: 将原始区域缩放得到篡改区域, 比值从 90% 到 110%, 步长为 4%, 同时补充使用 50%、80%、120% 和 200% 的缩放图像库, 共 480 张图像.

(2) 旋转: 将原始区域旋转 0°、10°、30°、50°和 180°得到篡改区域, 共 240 张图像.

(3) JPEG 压缩: 篡改区域的 JPEG 比例因子从 20 到 100, 步长 10. 共 392 张图像.

(4) 噪声: 篡改区域添加 0.02, 0.04, 0.06, 0.08, 0.1 std 高斯白噪声. 共 240 张图像.

我们在图像级和像素级分别进行了对比实验, 对比文献 [8,12,13] 的算法. 图 8、图 9 分别表示图像级与像素级实验结果.

如图 8、图 9 所示, 该算法在缩放、旋转、噪声和 JPEG 压缩 4 种攻击下的检测效果良好. 不仅在图像级进行正确的检测, 且在像素级得到精准的结果. 在极端缩放攻击 (50% 和 200%) 下, 文献 [8] 和文献 [12] 表现不佳, 因其特征只有一定程度的尺度不变性, 无法

抵抗大比例缩放攻击, 相反本方案选用的 SURF 特征具有良好的尺度不变性. 与其他 3 种算法相比, 本算法在噪声攻击和 JPEG 压缩攻击下均具有更好的性能, 这归功于后处理算法的鲁棒性.

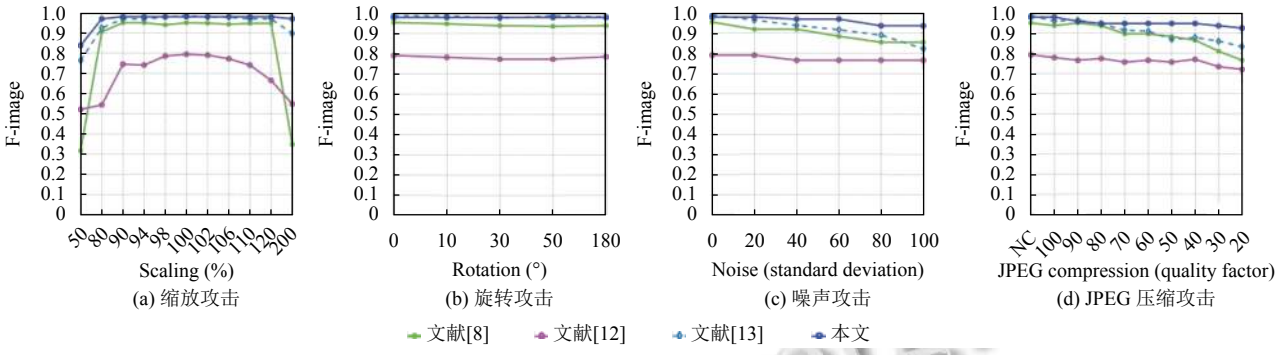


图 8 图像级对比结果

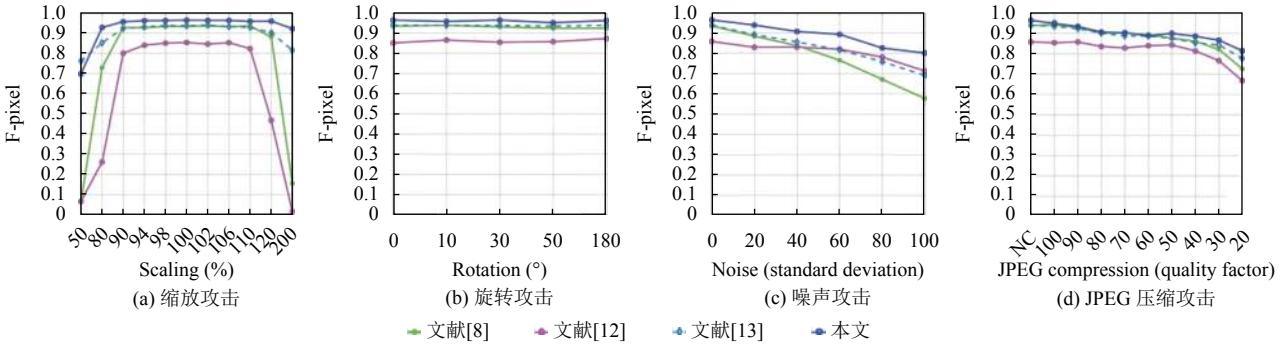


图 9 像素级对比结果

4 结论

图像复制粘贴篡改是一种常见的数字图像篡改方式. 我们将图像分割成不重叠的块并延拓, 构造波动函数, 根据波动函数值确定阈值均匀地提取图像特征点. 使用尺度不变和强描述力的 SURF 特征. 引入 DBQ-LSH 匹配算法. 提出了一种新的定位方法, 不变矩 LBP 定位方法. 通过使用不变矩矩值代替像素值将 LBP 算法从空间域拓展到变换域, 比较两幅不变矩 MRELBP 图像的相似度而不是两幅灰度图像的相似度, 从而提高对攻击图像检测的鲁棒性. 实验结果表明, 该方案具有较好的检测性能, 在检测效果得到提高的同时, 定位精度也明显提升. 在未来的工作中, 我们将寻找更具鲁棒性的特征, 并尝试引入软计算.

参考文献

1 Bay H, Ess A, Tuytelaars T, *et al*. Speeded-up robust features (SURF). Computer Vision and Image Understanding, 2008, 110(3): 346–359. [doi: 10.1016/j.cviu.2007.09.014]

2 Kong WH, Li WJ. Double-bit quantization for hashing. Proceedings of the 26th AAAI Conference on Artificial Intelligence. Toronto, ON, Canada. 2012. 634–640.

3 Gong YC, Lazebnik S, Gordo A, *et al*. Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(12): 2916–2929. [doi: 10.1109/TPAMI.2012.193]

4 Ojala T, Pietikainen M, Maenpaa T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002, 24(7): 971–987. [doi: 10.1109/TPAMI.2002.1017623]

5 Fridrich AJ, Soukal BD, Lukáš AJ. Detection of copy-move forgery in digital images. Proceedings of Digital Forensic Research Workshop. Cleveland, OH, USA. 2003. 55–61.

6 Ryu SJ, Lee MJ, Lee HK. Detection of copy-rotate-move forgery using Zernike moments. Proceedings of the 12th International Workshop on Information Hiding. Calgary, AB, Canada. 2010. 51–65.

7 Cozzolino D, Poggi G, Verdoliva L. Copy-move forgery detection based on PatchMatch. Proceedings of 2014 IEEE International Conference on Image Processing (ICIP). Paris, France. 2014. 5312–5316.

8 Cozzolino D, Poggi G, Verdoliva L. Efficient dense-field copy-move forgery detection. IEEE Transactions on Information Forensics and Security, 2015, 10(11): 2284–2297. [doi: 10.1109/TIFS.2015.2455334]

9 Bi XL, Pun CM. Fast copy-move forgery detection using local bidirectional coherency error refinement. Pattern Recognition, 2018, 81: 161–175. [doi: 10.1016/j.patcog.2018.03.028]

10 Zhong JL, Pun CM. Two-pass hashing feature representation and searching method for copy-move forgery detection. Information Sciences, 2020, 512: 675–692. [doi: 10.1016/j.ins.2019.09.085]

11 Amerini I, Ballan L, Caldelli R, *et al*. A sift-based forensic method for copy-move attack detection and transformation recovery. IEEE Transactions on Information Forensics and Security, 2011, 6(3): 1099–1110. [doi: 10.1109/TIFS.2011.2129512]

12 Zandi M, Mahmoudi-Aznaveh A, Talebpour A. Iterative copy-move forgery detection based on a new interest point detector. IEEE Transactions on Information Forensics and Security, 2016, 11(11): 2499–2512. [doi: 10.1109/TIFS.2016.2585118]

13 Li YM, Zhou JT. Fast and effective image copy-move forgery detection via hierarchical feature point matching. IEEE Transactions on Information Forensics and Security, 2019, 14(5): 1307–1322. [doi: 10.1109/TIFS.2018.2876837]

14 Shivakumar BL, Baboo SS. Detection of region duplication forgery in digital images using SURF. IJCSI International Journal of Computer Science Issues, 2011, 8(4): 199–205.

15 Sachdev K, Kaur M, Gupta S. A robust and fast technique to detect copy move forgery in digital images using SLIC segmentation and SURF keypoints. In: Singh R, Choudhury S, eds. Proceeding of International Conference on Intelligent Communication, Control and Devices. Singapore: Springer, 2017. 787–793.

16 Liu Y, Wang HX, Wu HZ, *et al*. An efficient copy-move detection algorithm based on Superpixel segmentation and Harris key-points. Proceedings of the 3rd International Conference on Cloud Computing and Security. Nanjing, China. 2017. 61–73.

17 Yang F, Li JW, Lu W, *et al*. Copy-move forgery detection based on hybrid features. Engineering Applications of Artificial Intelligence, 2017, 59: 73–83. [doi: 10.1016/j.engappai.2016.12.022]

18 Wang CY, Zhang Z, Zhou X. An image copy-move forgery detection scheme based on A-KAZE and SURF features. Symmetry, 2018, 10(12): 706. [doi: 10.3390/sym10120706]

19 余江. 数字图像隐写分析研究 [博士学位论文]. 上海: 上海大学, 2016. 71–74.

20 Nie FP, Wang CL, Li XL. K-multiple-means: A multiple-means clustering method with specified K clusters. Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. Anchorage, AK, USA. 2019. 959–967.

21 Liu L, Lao SY, Fieguth PW, *et al*. Median robust extended local binary pattern for texture classification. IEEE Transactions on Image Processing, 2016, 25(3): 1368–1381. [doi: 10.1109/TIP.2016.2522378]

22 Khotanzad A, Hong YH. Invariant image recognition by Zernike moments. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1990, 12(5): 489–497. [doi: 10.1109/34.55109]

23 Amerini I, Ballan L, Caldelli R, *et al*. Copy-move forgery detection and localization by means of robust clustering with J-linkage. Signal Processing: Image Communication, 2013, 28(6): 659–669. [doi: 10.1016/j.image.2013.03.006]

24 Christlein V, Riess C, Jordan J, *et al*. An evaluation of popular copy-move forgery detection approaches. IEEE Transactions on Information Forensics and Security, 2012, 7(6): 1841–1854. [doi: 10.1109/TIFS.2012.2218597]