

基于超像素滤波的非对称失真隐写方法

朱利妍^{1),3)} 罗向阳^{1),2)} 张 祎^{1),2)} 王金伟⁴⁾ 卢 伟⁵⁾ 刘粉林^{1),2)}

¹⁾(中国人民解放军战略支援部队信息工程大学网络空间安全学院 郑州 450001)

²⁾(河南省网络空间态势感知重点实验室 郑州 450001)

³⁾(河南警察学院 郑州 450001)

⁴⁾(南京信息工程大学计算机学院、网络空间安全学院 南京 210044)

⁵⁾(中山大学计算机学院 广州 510006)

摘 要 数字图像隐写将秘密信息嵌入到图像的冗余部分,并通过公开信道(如 E-mail、社交平台等)传输以实现隐蔽通信。近年来,主流的数字图像隐写技术往往通过设计多样化的对称/非对称失真函数,将信息自适应嵌入到图像纹理复杂区域,以提升方法的抗统计检测性能。相比对称失真函数,最新的基于图像恢复或滤波原理的非对称失真函数能更好地指示信息嵌入时的修改方向。然而,现有基于图像恢复的非对称失真隐写往往耗时较长,基于滤波的非对称失真隐写抗检测能力还有待提升。为此,本文首先分析指出现有基于滤波的非对称失真隐写会弱化图像中物体的边缘与轮廓,降低边信息图像的质量;然后,提出超像素滤波的概念及算法,将描绘物体边缘与轮廓的超像素加入到滤波过程中,以提高构造边信息图像的质量;接着,给出一种超像素去噪算法,避免过小的超像素成为图像的“干扰噪声”,以更好地描绘物体的边缘与轮廓;最后,利用生成的边信息图像,结合现有非对称失真函数设计策略,得到能够更准确指示信息嵌入修改方向的非对称失真函数,进而给出基于超像素滤波的非对称失真隐写方法。基于常用的 Bossbase-1.01、UCIB、BOWSE、iStego100K 等多个图像库,在 0.1~0.5 bpnzAC 的嵌入率、65~95 的 JPEG 压缩质量因子下进行了大量实验,结果表明:与现有基于滤波构造边信息图像的典型隐写方法相比,本文方法所得边信息图像的峰值信噪比平均提高 1.05 dB,结构相似度平均提高 0.04,验证了超像素滤波的优势;与 J-UNIWARD、UED、UERD 等对称失真隐写典型方法相比,所提方法的抗统计检测性能分别平均提升 7.6%、4.2%、6.0%;与现有基于图像恢复的非对称失真隐写典型方法相比,所提方法生成边信息图像的耗时降低约 87%;与现有最新的基于均值滤波的非对称失真隐写方法相比,抗统计检测性能提升了 0.3%~12%。

关键词 隐写;非对称失真;超像素滤波;边信息;抗统计检测

中图法分类号 TP391 **DOI号** 10.11897/SP.J.1016.2023.01473

Asymmetric Distortion Steganography Method Based on Superpixel Filtering

ZHU Li-Yan^{1),3)} LUO Xiang-Yang^{1),2)} ZHANG Yi^{1),2)} WANG Jin-Wei⁴⁾ LU Wei⁵⁾ LIU Fen-Lin^{1),2)}

¹⁾(School of Cyberspace Security, PLA Strategic Support Force Information Engineering University, Zhengzhou 450001)

²⁾(Key Laboratory of Cyberspace Situation Awareness, Zhengzhou 450001)

³⁾(Henan Police College, Zhengzhou 450001)

⁴⁾(School of Computer Science, Nanjing University of Information Science and Technology, Nanjing 210044)

⁵⁾(School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510006)

Abstract Digital image steganography embeds secret messages into the redundant part of an image and transmits the image through open channels (such as E-mail, social platforms, etc.) to realize secure covert communication. In recent years, the dominant research direction in digital image steganography has been adaptive steganography. It can adaptively embed messages into the

收稿日期:2022-04-05;在线发布日期:2023-01-06。本课题得到国家自然科学基金(62172435,62202495,62261160653,62102101)、中原科技创新领军人才项目(214200510019)和阿里巴巴创新研究计划资助。朱利妍,博士研究生,主要研究方向为图像隐写技术。E-mail: zhuliyan1943@163.com。罗向阳(通信作者),博士,教授,博士生导师,主要研究领域为网络与多媒体安全。E-mail: luoxxy_ieu@sina.com。张 祎,博士,讲师,主要研究方向为图像隐写与分析技术。王金伟,博士,教授,博士生导师,主要研究领域为信息安全与多媒体取证。卢 伟,博士,教授,博士生导师,主要研究领域为多媒体安全和数字取证。刘粉林,博士,教授,博士生导师,主要研究领域为网络信息安全。

complex texture regions of an image by designing diversified distortion functions to continuously improve the anti-statistical detection performance of a method. Distortion functions are divided into the symmetric distortion function and the asymmetric distortion function according to whether the addition or subtraction distortion of a position is the same. Compared with the symmetric distortion function, the asymmetric distortion function can better guide the modification direction of the message embedding with the help of side-information, and thus has more advantages in anti-statistical detection performance. In the existing asymmetric distortion steganographic methods, there are two main design directions: the method based on image restoration and the method based on filtering. However, image restoration based methods usually take a long time, while the anti-detection ability of filtering based methods still needs to be improved. To address these issues, this manuscript firstly analyzes and points out the problems of the existing filtering based asymmetric distortion steganography: such methods weaken the contours and edges of objects in an image to a certain extent, and thus reduce the quality of a side-information image. Then, to remedy this defect, the concept and algorithm of superpixel filtering are proposed, and the superpixel depicting the edges and contours of objects are added into a filtering process. In this way, the negative influence of filtering on contour and edge can be reduced, and the quality of a side information image can be improved. Next, to avoid tiny superpixels becoming “interference noises” of an image, the superpixel denoising algorithm is given. It can better depict the edges and contours of objects. Last, the asymmetric distortion function, which can better guide the modification direction of the message embedding, is obtained by exploiting the generated side information images in combination with the existing distortion design strategy, and further, the asymmetric distortion steganographic method based on superpixel filtering is presented. Extensive experiments were performed on image databases commonly used in steganography, such as Bossbase-1.01, UCID, BOWS2, and IStego100K, with payloads between 0.1 and 0.5 bpnzAC and JPEG compression quality factors between 65 and 95. The results verify that compared with the existing side-information image construction typical methods based on filtering, the *PSNR* and *SSIM* of side-information images constructed by superpixel filtering are increased by 1.05 dB and 0.04, respectively, which verify the validity of superpixel filtering. Compared with symmetric distortion steganographic typical methods such as J-UNIWARD, UED, and UERD, the anti-statistical detection performance of the proposed method is improved by about 7.6%, 4.2%, 6.0%, respectively. Compared with the existing asymmetric distortion steganographic typical method based on image restoration, the time of generating a side-information image is reduced by about 87%. Compared with the latest average filtering-based asymmetric distortion steganographic typical method, the anti-statistical detection performance of the proposed method is improved by 0.3%~12%.

Keywords steganography; asymmetric distortion; superpixel filtering; side information; anti-statistical detection

1 引 言

数字图像隐写是将秘密信息嵌入到正常图像中,通过公开信道传输实现隐蔽通信的技术^[1-3].近年来,基于“失真函数设计+STC(Syndrome-Trellis Codes)编码”框架^[4],自适应隐写根据图像内容自适

应地将信息嵌入到图像纹理复杂区域,取得了良好的抗统计检测性能,成为近年来图像隐写领域的研究热点.其中,失真函数根据图像特征度量每个位置因信息嵌入引起的图像失真,是最小化嵌入失真编码(如 STC 编码、SPC 编码^[5]等)的前提,也是自适应隐写方法的研究重点.寻找满足边缘、噪声等纹理复杂区域像素失真小,光滑区域像素失真大的失真

函数是研究人员追求的目标^[6-8]。

早期的失真函数设计多集中在对称失真函数方面,其特点是图像中任意像素或系数的失真在两个修改方向(加减)上是相等的,信息嵌入时对像素或系数的修改方式为随机加减。基于对称失真函数提出的一大批自适应隐写方法,使得载密图像的抗统计检测性能不断提升。J-UNIWARD(JPEG UNI-versal Wavelet Relative Distortion)是经典的 JPEG 图像隐写方法^[9],该方法通过综合考虑小波的方向性定义失真,将信息嵌入区域集中在纹理与噪声区域,大幅提高了隐写图像的安全性。UED(Uniform Embedding Distortion)首次提出统一嵌入策略,为 JPEG 隐写失真函数设计开辟了一个新的思路^[10]。基于该思路的隐写方法后续被陆续提出,如 UERD(Uniform Embedding Revisited Distortion)^[11]、GUED(Generalized Uniform Embedding Distortion)^[12]等。

后来,研究者们提出了非对称失真,在两个修改方向赋予像素或系数不同的失真,使生成的载密图像更接近自然图像,在对称隐写方法的基础上进一步提升方法的安全性^[13-14]。根据是否已知原始图像相关信息,可以将非对称失真隐写分为两类。一类是原始图像相关信息已知的非对称失真隐写,原始图像相关信息一般指载体图像的更高质量版本,如空域图像^[9]、量化前系数^[15]、连拍图像^[16]等,这些信息往往需要额外提供和获取。另一类是原始图像信息未知的非对称失真隐写,此类方法一般利用载体图像构造边信息(图像)。高质量的边信息图像可以帮助失真函数构造得更准确,失真函数越准确,信息嵌入到载体图像后生成的载密图像越接近载体图像,从而更不容易被检测出隐藏了信息,故而方法的抗统计检测性能更好。目前构造边信息(图像)的主流方法有两种,一种是利用图像恢复相关算法构造边信息^[17-18],如 SSRQC(Structural Sparse Representation and Quantization Constraint)算法^[19],该算法同时增强自然图像的固有局部稀疏性和非局部自相似性,恢复出的边信息接近自然图像^[18]。另一种是使用滤波相关算法构造边信息图像,如维纳滤波^[20]、均值滤波^[21-22]等,滤波算法计算简单,在生成边信息图像的耗时方面具有明显优势。

目前非对称失真隐写还存在一些不足:对于原始图像相关信息已知的非对称失真隐写,信息的额外获取往往较为困难。原始图像相关信息未知的非对称失真隐写中,利用图像恢复相关算法构造边信息的非对称失真隐写方法(简称:基于图像恢复的非

对称失真隐写)计算过程往往很复杂,需要大量的块操作与反复迭代,耗时较长;利用滤波相关算法构造边信息图像的非对称失真隐写方法(简称:基于滤波的非对称失真隐写)的边信息图像质量有待提高,方法抗统计检测性能有待提升。为此,本文尝试给出一种高效的边信息构造方法,在保证耗时短的前提下,提升基于滤波的非对称隐写方法的抗统计检测性能。本文主要工作如下:

(1) 结合突出物体边缘和轮廓的超像素,提出超像素滤波的概念及算法,该算法弥补滤波操作造成的图像轮廓、纹理等位置处因周围像素值差别较大而弱化当前位置特征的缺陷。

(2) 为避免太大的超像素成为图像的“干扰噪声”,影响生成图像的质量,给出了一种超像素去噪算法,以更好地描绘物体边缘与轮廓,生成的图像质量更高。

(3) 将超像素滤波值作为边信息应用于隐写,结合现有非对称失真函数设计策略,得到能够更准确指示信息嵌入修改方向的非对称失真函数,进而给出基于超像素滤波的非对称失真隐写方法,该方法生成载密图像的抗统计检测性能更高。

本文第 2 节简要介绍与分析基于滤波的非对称失真隐写;第 3 节提出超像素滤波的概念和算法,并对其进行原理分析与统计对比,验证超像素滤波的有效性;在此基础上,第 4 节提出基于超像素滤波的非对称失真隐写,详细介绍方法框架与细节;第 5 节给出实验结果与分析,首先对超像素滤波及失真函数参数进行测试,接着用大量实验验证本文提出的非对称失真隐写方法的优越性;第 6 节对本文进行总结与讨论。

2 基于滤波的非对称失真隐写简介与分析

借助边信息的非对称失真隐写^[15]自提出后,得到了进一步发展^[16-22]。其中,基于滤波的非对称失真隐写方法^[20-22]不需要借助载体以外的图像,且耗时较短,相较其他类型的非对称失真隐写方法,更为实用。非对称失真隐写方法的重点是失真函数的设计和边信息图像的构造,因此,本节首先对基于滤波的非对称失真函数原理进行简单介绍,再对基于滤波的边信息图像构造方法原理进行分析。

2.1 基于滤波的非对称失真函数^[20-22]原理简介

基于滤波的非对称失真函数的设计思路一般如

下:首先利用现有对称失真隐写方法计算 JPEG 载体图像的对称失真;然后对 JPEG 载体图像进行滤波,构造边信息图像;最后利用非对称失真设计策略,结合边信息图像与 JPEG 载体图像的对比结果,将对称失真调整为非对称失真.具体步骤如下:

(1) 对称失真计算. 利用现有隐写方法(如 J-UNIWARD^[9]、UERD^[11]等)计算 JPEG 载体图像 $\mathbf{X}=[x_{i,j}]_{w \times h}$ 中每个 DCT 系数 $x_{i,j}$ 的失真 $\rho_{i,j}$, 此时系数“+1”和“-1”引起的失真被定义为相等的,即得到的是对称失真. 其中 (i,j) 是 $\mathbf{X}$ 的像素位置, w 和 h 分别是载体图像的宽和高.

(2) 边信息图像构造. 对载体图像执行滤波操作,利用载体图像与滤波后的图像构造边信息图像,或者直接使用滤波后的图像作为边信息图像,边信息图像记为 $\mathbf{X}'=[x'_{i,j}]_{w \times h}$.

(3) 非对称失真生成. 根据式(1)所示的非对称失真设计策略对对称失真进行调整,得到非对称失真. 其中 $\rho_{i,j}^+$ 和 $\rho_{i,j}^-$ 分别表示系数“+1”和“-1”引起的失真; $q_{i,j}$ 是 $\mathbf{X}$ 中位置 (i,j) 处的量化步长; $0 \leq \alpha \leq 1$, 是非对称失真调整参数.

$$\rho_{i,j}^+ = \begin{cases} \alpha \times \rho_{i,j}, & x_{i,j} < x'_{i,j}/q_{i,j} \\ \rho_{i,j}, & x_{i,j} \geq x'_{i,j}/q_{i,j} \end{cases}, \quad \rho_{i,j}^- = \begin{cases} \alpha \times \rho_{i,j}, & x_{i,j} > x'_{i,j}/q_{i,j} \\ \rho_{i,j}, & x_{i,j} \leq x'_{i,j}/q_{i,j} \end{cases} \quad (1)$$

2.2 基于滤波的边信息图像构造方法原理分析

基于滤波的边信息图像构造方法利用滤波操作在抑制图像噪声的同时尽可能保留图像细节特征的特点,生成一幅边信息图像. 方法一般包含两步: (1) 对载体图像执行滤波操作获得滤波后的图像,如对载体图像进行均值滤波,得到均值滤波后的图像; (2) 利用滤波后图像构造边信息图像. 构造方法主要包含两种:一种是结合载体图像与滤波后图像构造边信息图像,如将载体图像分成 8×8 的块,块边界的像素值用滤波图像中相同位置的值替换,替换后的图像作为边信息图像^[21]. 一种是直接使用滤波后图像作为边信息图像,如直接将均值滤波后的图像作为边信息图像^[22].

尽管滤波操作尽可能保留图像细节特征,但滤波后图像的边缘、轮廓等可能被模糊,影响图像的质量. 以一个 10×10 的矩阵 $\mathbf{M}$ 为例,对 $\mathbf{M}$ 做均值滤波,滤波窗口为 3×3 ,滤波后结果记为 $\mathbf{M}'$. 对比 $\mathbf{M}$ 与 $\mathbf{M}'$ 可以发现,两者在部分区域差别较大. 如式(2)与式(3)所示,根据数值可将 $\mathbf{M}$ 分为左右两个区域,左侧区域值的范围为 $11 \sim 16$,右侧区域值的范围为 $59 \sim 62$. 同样地,可将 $\mathbf{M}'$ 分为左中右三个区域,左侧

区域值的范围为 $11 \sim 15$,中间区域值的范围为 $19 \sim 45$,右侧区域值的范围为 $50 \sim 62$. 从值的对比与分布可知, $\mathbf{M}$ 中左右两个区域相邻处, $\mathbf{M}'$ 与 $\mathbf{M}$ 的差别较大.

图 1 是 $\mathbf{M}$ 与 $\mathbf{M}'$ 的灰度表示, $\mathbf{M}$ 对应的图(a)中,左右区域分别以明显的黑、白色显示, $\mathbf{M}'$ 对应的图(b)中,除黑、白色外,还存在不同程度的灰色,且灰色区域对应图(a)中黑白交界处. 若将 $\mathbf{M}$ 视为一幅图像,则黑白交界处(相邻像素相差较大)可视为图像的边缘与轮廓. 从 $\mathbf{M}$ 与 $\mathbf{M}'$ 的对比可知,当相邻像素相差较大时, $\mathbf{M}$ 与 $\mathbf{M}'$ 的差别明显.

$$\mathbf{M} = \begin{bmatrix} 12 & 13 & 14 & 14 & 15 & 15 & 60 & 61 & 62 & 62 \\ 13 & 13 & 14 & 14 & 15 & 15 & 16 & 60 & 62 & 62 \\ 13 & 14 & 14 & 14 & 15 & 15 & 59 & 60 & 61 & 61 \\ 13 & 12 & 12 & 14 & 15 & 60 & 59 & 60 & 61 & 61 \\ 13 & 14 & 15 & 15 & 16 & 59 & 59 & 60 & 61 & 61 \\ 12 & 14 & 14 & 15 & 15 & 16 & 60 & 60 & 61 & 62 \\ 13 & 13 & 13 & 14 & 15 & 60 & 60 & 60 & 61 & 61 \\ 11 & 11 & 12 & 13 & 14 & 15 & 61 & 61 & 62 & 61 \\ 12 & 12 & 13 & 13 & 14 & 16 & 60 & 61 & 61 & 61 \\ 11 & 12 & 13 & 14 & 14 & 59 & 59 & 60 & 62 & 62 \end{bmatrix} \quad (2)$$

$$\mathbf{M}' = \begin{bmatrix} 13 & 13 & 14 & 14 & 15 & 25 & 40 & 56 & 62 & 62 \\ 13 & 13 & 14 & 14 & 15 & 25 & 40 & 56 & 61 & 62 \\ 13 & 13 & 13 & 14 & 20 & 30 & 45 & 55 & 61 & 61 \\ 13 & 13 & 14 & 14 & 25 & 40 & 55 & 60 & 61 & 61 \\ 13 & 13 & 14 & 15 & 25 & 40 & 55 & 60 & 61 & 61 \\ 13 & 13 & 14 & 15 & 25 & 40 & 55 & 60 & 61 & 61 \\ 12 & 13 & 13 & 14 & 20 & 35 & 50 & 61 & 61 & 61 \\ 12 & 12 & 13 & 13 & 19 & 35 & 50 & 61 & 61 & 61 \\ 11 & 12 & 13 & 13 & 19 & 35 & 50 & 61 & 61 & 61 \\ 12 & 12 & 13 & 14 & 24 & 39 & 55 & 60 & 61 & 62 \end{bmatrix} \quad (3)$$

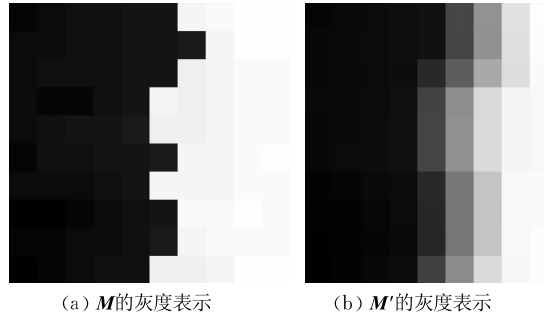


图 1 矩阵的灰度表示

之所以会有这样的差距,是因为滤波会考虑周围像素对当前位置的影响. 若当前位置位于图像轮廓、边缘等区域,由于周围像素值差别较大,对当前位置的参考价值有限,故会弱化当前位置的特征,进而影响生成图像的质量. 在隐写中,该图像被用于构

造边信息图像(2.1节中的第(2)步),并直接影响非对称失真函数的设计(2.1节中的第(3)步).边信息图像质量越高,非对称失真函数设计的越合理,反之亦然.简而言之,在图像物体边缘与轮廓处直接使用滤波会影响生成图像的质量,进而影响非对称失真函数设计的合理性和基于该失真函数提出的隐写方法的抗统计检测性能.

3 超像素滤波 SUFI

目前,利用滤波生成边信息图像的算法大多直接使用规则的数据块^[21-22](如 3×3 的块)计算.块是滤波计算的单位,并未考虑图像中事物的特征.为提高滤波生成图像的质量,考虑在对规则块计算之前,结合图像中事物的含义对块中的值进行适当调整.为此,本节尝试将超像素这一简单有效的描绘图像中事物的方式加入到滤波中,提出超像素滤波的概念,并给出相关算法、原理分析和统计验证.

3.1 概念

从2.2节的分析可知,在图像轮廓、边缘等区域,周围像素值差别较大,此时综合考虑周围像素对当前位置的影响反而会影响生成图像的质量.一个自然的想法是在边界处分别滤波.依然以 M 为例,在 M 中相差较大的值(图1(a)中黑白交界处)的两侧分别进行滤波,滤波结果记为 M'' .图2是对应的灰度表示.可以看出,相比图1(b),图2中黑白交界处两侧的滤波效果明显,且未影响边缘.

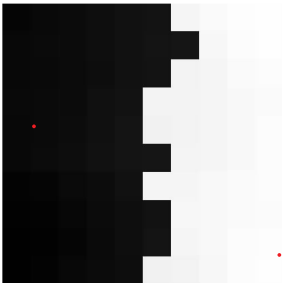


图2 M'' 的灰度表示

受该现象启发,本文考虑在图像中物体边缘或轮廓(相邻像素相差较大)处不直接使用滤波,而是先对这些位置的像素值进行合理地调整,使窗口内像素值差别缩小,再进行滤波.为准确描述边缘与轮廓,本文引入超像素^[23].超像素是具有共同或类似特征(如颜色、纹理等)的像素集合.如图3所示,黑色不规则线条表示超像素边缘,黑色线条组成的连通区域为超像素.每个超像素中包含的像素个数称

为超像素尺寸.在对图像执行滤波操作前,先利用超像素对滤波窗口内的像素值进行合理地调整,调整后窗口内像素值彼此差距更小、范围更集中,然后再对调整后的像素值执行滤波,本文称为超像素滤波 SUFI(SUPERpixel Filtering).超像素滤波的目的是优化普通滤波生成的图像,在耗时与普通滤波相当的情况下快速生成更高质量的图像.

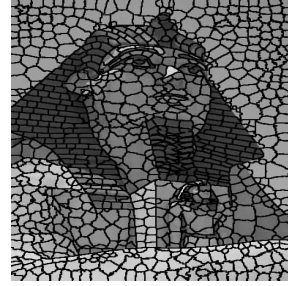


图3 超像素举例

3.2 算法描述

超像素滤波的核心思想为:如果一个滤波窗口内的像素均属于同一个超像素,则直接计算滤波,反之,按调整规则更新该窗口内与中心像素属于不同超像素的像素值,再计算滤波.此时,该窗口称为待调整窗口;若窗口内某像素对应的超像素索引与中心像素索引不同,则说明两者属于不同的超像素,该像素需要调整,称为待调整像素.

基于上述思想,本文提出超像素滤波算法,如算法1所示.

算法1 超像素滤波算法.

输入:空域图像 $C=[c_{i,j}]_{w\times h}$, $1\leq i\leq w, 1\leq j\leq h, c_{i,j}$ 是

图像的像素值.滤波窗口大小为 $N\times N$

输出: C 的超像素滤波图像 Z

1. 对图像 C 进行超像素划分,得到对应的超像素索引 $P=[p_{i,j}]_{w\times h}$, $0\leq p_{i,j}\leq m$, m 表示该图像被划分的超像素个数.
2. 计算图像 C 的滤波 $Z=[z_{i,j}]_{w\times h}=\text{Filter}(C)$.
3. For $i=1:w$
4. For $j=1:h$
5. 以 (i,j) 为中心分别在图像 C 和超像素索引 P 中取滤波窗口 $B_C=[c'_{u,v}]_{N\times N}$ 和对应超像素索引窗口 $B_P=[p'_{u,v}]_{N\times N}$.
6. If B_P 中超像素索引种类 >2
7. 根据式(4)对窗口内待调整像素进行调整.
8. 对调整过的窗口求均值,并利用其更新滤波值 $z_{i,j}=\text{Mean}(B_C)$.
9. End
10. End
11. End
12. Return $Z=\langle Z \rangle$.

算法主要步骤如下:

(1) 超像素索引计算. 如算法 1 第 1 行所示, 对一幅图像进行超像素划分, 得到对应的超像素索引. 每一个像素对应一个超像素索引, 同一个超像素内的像素对应相同的超像素索引.

(2) 普通滤波计算. 如算法 1 第 2 行所示, 由于超像素滤波是在普通滤波的基础上进行的, 故先对图像进行普通滤波, 得到滤波后的值.

(3) 待调整窗口查找. 如算法 1 第 3~6 行所示, 依次遍历滤波窗口, 结合第一步得到的超像素索引, 查找包含多个超像素索引的待调整窗口, 窗口内部分像素值需要调整.

(4) 像素值调整. 按序依次调整窗口内的待调整像素. 其中调整顺序不是顺序遍历窗口内元素而是按由内向外的顺序, 调整策略如式(4)所示. 其中 $\langle \cdot \rangle$ 指四舍五入的取整操作.

$$c'_{u,v} = \begin{cases} c'_{\langle N/2 \rangle, v}, v = \langle N/2 \rangle \\ c'_{u, \langle N/2 \rangle}, u = \langle N/2 \rangle \\ c'_{u,v+1}, u < \langle N/2 \rangle \& v < \langle N/2 \rangle \& |c'_{u,v+1} - c'_{u,v}| < |c'_{u+1,v} - c'_{u,v}| \text{ or } \\ u > \langle N/2 \rangle \& v < \langle N/2 \rangle \& |c'_{u,v+1} - c'_{u,v}| < |c'_{u-1,v} - c'_{u,v}| \\ c'_{u+1,v}, u < \langle N/2 \rangle \& v < \langle N/2 \rangle \& |c'_{u,v+1} - c'_{u,v}| \geq |c'_{u+1,v} - c'_{u,v}| \text{ or } \\ u < \langle N/2 \rangle \& v > \langle N/2 \rangle \& |c'_{u,v-1} - c'_{u,v}| \geq |c'_{u+1,v} - c'_{u,v}| \\ c'_{u,v-1}, u < \langle N/2 \rangle \& v > \langle N/2 \rangle \& |c'_{u,v-1} - c'_{u,v}| < |c'_{u+1,v} - c'_{u,v}| \text{ or } \\ u > \langle N/2 \rangle \& v > \langle N/2 \rangle \& |c'_{u,v-1} - c'_{u,v}| < |c'_{u-1,v} - c'_{u,v}| \\ c'_{u-1,v}, u > \langle N/2 \rangle \& v < \langle N/2 \rangle \& |c'_{u,v+1} - c'_{u,v}| \geq |c'_{u-1,v} - c'_{u,v}| \text{ or } \\ u > \langle N/2 \rangle \& v > \langle N/2 \rangle \& |c'_{u,v-1} - c'_{u,v}| \geq |c'_{u-1,v} - c'_{u,v}| \end{cases} \quad (4)$$

(5) 调整后窗口计算. 如算法 1 第 8 行所示, 对调整后的窗口执行均值计算. 当一幅图像的所有滤波窗口都调整并计算完成后, 得到该图像对应的超像素滤波图像.

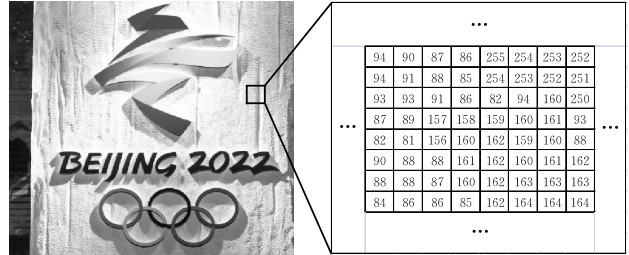
在算法 1 中, 第 2 行、第 8 行的 $\text{Filter}(\cdot)$ 、 $\text{Mean}(\cdot)$ 分别表示滤波和均值计算函数. 算法 1 描述的是将滤波窗口内与中心像素的超像素索引不一致的所有像素进行调整, 适用于对物体边缘扩散严重的滤波, 如均值滤波; 当物体边缘扩散有限, 甚至有保留时,

可减少调整范围, 只对物体边缘像素进行调整, 如维纳滤波.

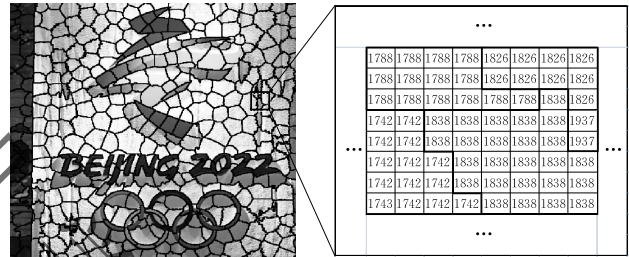
例 1. 超像素滤波过程举例.

为了便于理解, 以 $N=3$ 为例介绍超像素滤波过程, 滤波类型为均值滤波. 假设一幅图像 C 如图 4(a) 所示. 图中一个像素块的像素值如图中右侧所示. 超像素滤波过程如下:

(1) 对 C 进行超像素划分, 得到对应的超像素索引 P . 如图 4(b) 所示, 超像素索引值如图中右侧所示.



(a) 图像像素块



(b) 图像超像素索引

图 4 超像素滤波举例示意图

(2) 对 C 进行普通滤波计算, 则部分滤波值 (第 2~4 行、4~6 列) 如式(5) 所示:

$$Z = \begin{bmatrix} \dots & 123.78 & 161.00 & 206.33 & \dots \\ \dots & 128.89 & 147.89 & 175.00 & \dots \\ \dots & 134.56 & 135.56 & 144.11 & \dots \\ \dots & & & & \dots \end{bmatrix} \quad (5)$$

(3) 以 (i, j) 为中心取 3×3 的滤波窗口 B_c , 其中 (i, j) 为图像 C 中任意像素位置. 以图 4(a) 中像素块第 3 行第 5 列的值为例, $c_{i,j} = 82$, $p_{i,j} = 1788$, B_c 和 B_p 分别为

$$B_c = [c'_{u,v}]_{3 \times 3} = \begin{bmatrix} 85 & 254 & 253 \\ 86 & 82 & 94 \\ 158 & 159 & 160 \end{bmatrix} \quad (6)$$

和

$$B_p = [p'_{u,v}]_{3 \times 3} = \begin{bmatrix} 1788 & 1826 & 1826 \\ 1788 & 1788 & 1788 \\ 1838 & 1838 & 1838 \end{bmatrix} \quad (7)$$

B_p 中存在 4 种超像素索引, 故 B_c 中有待调整像素。

(4) 按序遍历 B_p 中 $p'_{u,v} \neq 1788$ 的位置并对 B_c 中对应位置处的待调整像素进行调整. 以第 1 行第 2 列的 $p'_{u,v} = 1826$ 为例, 在 C 中对应的值为 254, 根据式(4)将其更新为 82.

(5) 继续按序遍历 $p'_{u,v} \neq 1788$ 的位置, 对 B_c 中对应位置处的待调整像素进行调整. 最终, B_c 被更新为

$$B_c = \begin{bmatrix} 85 & 82 & 94 \\ 86 & 82 & 94 \\ 86 & 82 & 94 \end{bmatrix} \quad (8)$$

(6) 对 B_c 执行滤波操作, $v_{i,j} = 87.22$.

至此, 以 (i, j) 为中心、滤波窗口为 B_c 的超像素滤波完成, 滤波后该位置处的值为 87.22. 对 C 中所有位置执行超像素滤波后, 得到 C 对应的超像素滤波值, 取整后部分滤波值(第 2~4 行、4~6 列)如式(9)所示:

$$Z = \begin{bmatrix} \dots & & & & \\ \dots & 86 & 254 & 253 & \dots \\ \dots & 87 & 87 & 175 & \dots \\ \dots & 135 & 136 & 144 & \dots \\ \dots & & & & \end{bmatrix} \quad (9)$$

3.3 算法原理分析

本节从原理上分析超像素滤波比普通滤波(以均值滤波为例)生成的图像质量更高.

一个图像 $C = [c_{i,j}]_{w \times h}$, 对应的超像素索引 $P = [p_{i,j}]_{w \times h}$. 对于图像内任意位置 (i, j) 处的像素 $c_{i,j}$, 其对应的滤波窗口为 $B_c = [c'_{u,v}]_{N \times N}$, 该窗口执行均值滤波和超像素滤波后的值分别记为 $s_{i,j}$ 和 $z_{i,j}$. B_c 有两种可能的情况:

(1) B_c 中所有像素对应的超像素索引都相同. 这意味着 B_c 内的所有像素都只属于一个超像素. 此时, 对 B_c 执行超像素滤波与普通滤波无差别.

(2) B_c 中像素对应的超像素索引至少包含两个不同的值. 这意味着 B_c 中的像素属于不同的超像素. 则由超像素划分原则^[23]可知, 在 B_c 中存在部分 $|c_{i,j} - c'_{u,v}|$ 较小, 且 $p'_{u,v} = p_{i,j}$; 同时存在部分 $|c_{i,j} - c'_{u,v}|$ 较大, 且 $p'_{u,v} \neq p_{i,j}$.

根据均值滤波的规则可知, 对于滤波窗口 B_c , 其滤波后的值可根据式(10)计算得到. 记调整后的 B_c 为 $[\zeta_{u,v}]_{N \times N}$, 则超像素滤波后的值可根据式(11)计算得到.

$$s_{i,j} = \sum_{u=1}^N \sum_{v=1}^N c'_{u,v} / (N \times N) \\ = c_{i,j} + \sum_{u=1}^N \sum_{v=1}^N (c'_{u,v} - c_{i,j}) / (N \times N) \quad (10)$$

$$z_{i,j} = c_{i,j} + \sum_{u=1}^N \sum_{v=1}^N (\zeta_{u,v} - c_{i,j}) / (N \times N) \quad (11)$$

则均值滤波和超像素滤波后的值与窗口中心位置的值 $c_{i,j}$ 之间的差分别为

$$|s_{i,j} - c_{i,j}| = \left| \sum_{u=1}^N \sum_{v=1}^N (c'_{u,v} - c_{i,j}) / (N \times N) \right| \quad (12)$$

$$|z_{i,j} - c_{i,j}| = \left| \sum_{u=1}^N \sum_{v=1}^N (\zeta_{u,v} - c_{i,j}) / (N \times N) \right| \quad (13)$$

由此可知, 当

$$\left| \sum_{u=1}^N \sum_{v=1}^N (c'_{u,v} - c_{i,j}) \right| > \left| \sum_{u=1}^N \sum_{v=1}^N (\zeta_{u,v} - c_{i,j}) \right| \quad (14)$$

时, 超像素滤波的值更接近窗口中心位置的值. 由于超像素滤波调整的是边缘、轮廓等区域, 这些区域的像素在未调整前往往差别较大. 由超像素滤波的定义可知, 超像素滤波会先对滤波窗口内的像素值进行调整, 调整后窗口内像素值彼此差距更小, 范围更集中. 调整后值的差距小于调整前值的差距, 该条件满足. 因此, 相较均值滤波, 基于本文提出的超像素滤波算法得到的像素值更接近窗口中心位置的像素值, 从而避免因周围像素差别太大导致边缘、轮廓被滤波模糊掉, 进而保证生成图像质量更高.

3.4 统计验证

为了验证与均值滤波、维纳滤波相比, 超像素滤波生成的图像质量更高, 本节分别从数据块和图像两个视角, 在 Bossbase-1.01^[24]、UCID^[25]、BOWS2^① 三个图像库进行详细的统计验证.

3.4.1 块的对比

本小节以数据块为单位进行统计验证, 包括两个方面的对比实验: 一是对图像中的像素块进行差值分布验证; 二是对图像中的像素块进行像素值的统计验证.

(1) 差值分布验证

取 Bossbase-1.01 中的图像 9.pgm, 如图 5 所示, 其中图(a)是该图像的空域图像, 图(b)是该空域图像被 JPEG 压缩(质量因子为 75)后的图像. 图(a)和图(b)中的白色矩形框对应的像素块分别如图(c)与图(d)所示, 大小为 16×16 . 对图(b)中图像分别使用均值滤波和超像素滤波, 滤波生成图像中对应的块如图(e)和图(f)所示.

① Bas P, Furon T. Break our watermarking system. <http://bowski2.ec-lille.fr/>, 2008

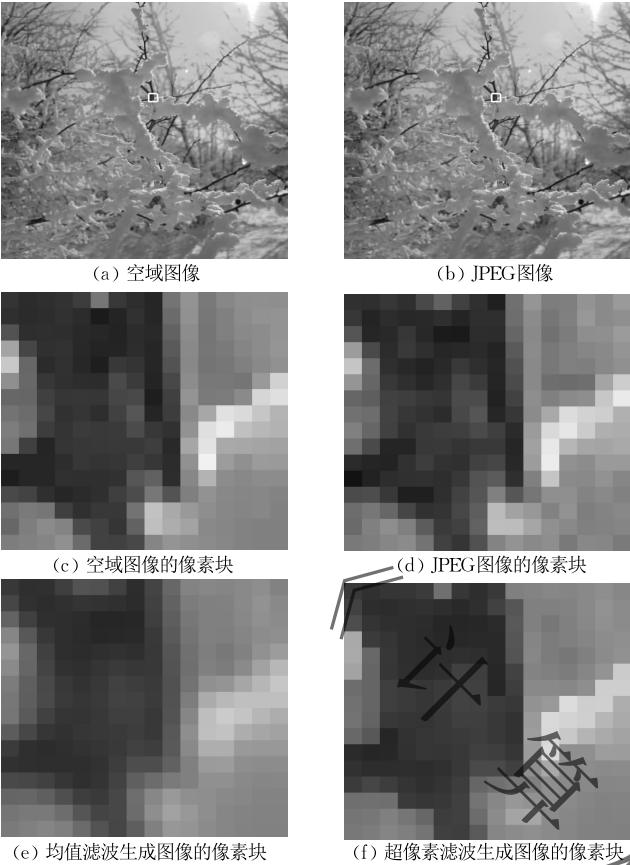


图 5 图像的局部像素块对比

均值滤波和超像素滤波生成图像的像素块与空域图像的像素块差距越小,表示生成的图像越接近空域图像,即图像的质量越高.如图 6 所示,横轴是均值滤波和超像素滤波生成图像中的像素块与原始像素块的差,左边的纵轴是某差值的数量在所有差值数量中的占比,右边的纵轴是差值.其中柱状图和火柴图对应左边的纵轴,面积图对应右边的纵轴.柱状图是均值滤波生成图像的像素块与空域图像像素块之间的差,火柴图是超像素滤波生成图像的像素块与空域图像像素块之间的差,面积图表示火柴图与柱状图的差.

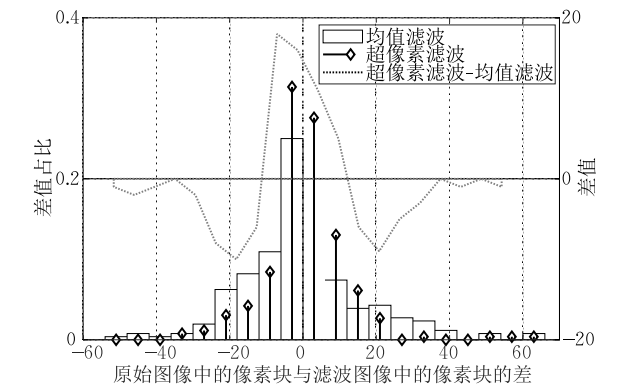


图 6 像素块的差对比

由图 6 可知,在像素差值为 0 的位置附近,火柴图中的柱高明显高于柱状图中的柱高;在像素差远离 0 的地方则恰好相反,柱状图的柱高高于火柴图中的柱高.从面积图可以发现,在像素差等于 0 的位置附近,火柴图与柱状图的差为正值,且相对较大.这说明超像素滤波生成的图像有更多的像素接近空域图像.在像素差远离 0 的地方,火柴图与柱状图的差为负值.这说明超像素滤波生成的图像与空域图像之间差别大的像素更少.所以,与均值滤波相比,超像素滤波生成的图像与空域图像的差别更小,换句话说,超像素滤波生成的图像与空域图像更接近,质量更高.

(2) 像素值统计验证

本小节以数据块为单位进行差值统计.在三个图像库中任取 3 幅图像:Bossbase-1.01 库中的 12.pgm, UCID 库中的 ucid00123.tif (以下简称 123.tif), BOWS2 库中的 1234.pgm.并对三幅图像做质量因子为 70 到 95、步长为 5 的 JPEG 压缩.分别对图像执行维纳滤波及其对应的超像素滤波操作、均值滤波及其对应的超像素滤波.

统计过程如下:将图像中的像素划分为不相交的 8×8 的块,块编号为 (l, g) ,其中 $1 \leq l \leq \lfloor w/8 \rfloor$, $1 \leq g \leq \lfloor h/8 \rfloor$,其中 $\lfloor \cdot \rfloor$ 表示向下取整.分别记录普通滤波及超像素滤波生成图像中每个像素块和原空域图像中对应的像素块的差距(计算方式见式(15)和(16)).

$$\eta = \sum_{i=1}^8 \sum_{j=1}^8 |c_{i,j} - s_{i,j}| \quad (15)$$

$$\xi = \sum_{i=1}^8 \sum_{j=1}^8 |c_{i,j} - z_{i,j}| \quad (16)$$

其中, (i, j) 是块中像素的坐标,统计结果如表 1~表 6 所示.

表 1 12.pgm 及对应 JPEG 图像中部分像素块的维纳滤波及其对应超像素滤波的统计结果

图像类型	滤波生成图像中的像素块和原空域图像中对应像素块的差距							
	(8,8)	(16,16)	(24,24)	(32,32)	(40,40)	(48,48)	(56,56)	(64,64)
空域	285	79	273	233	220	73	245	75
	70	30	119	135	69	34	91	27
频域 (95)	286	82	270	234	215	71	249	73
	101	53	167	189	107	66	91	47
频域 (90)	286	86	288	245	220	73	244	78
	174	66	175	182	137	74	164	74
频域 (85)	308	84	290	266	230	79	260	84
	177	77	226	227	154	74	147	79
频域 (80)	342	90	273	269	227	81	257	83
	179	87	209	240	165	93	163	75
频域 (75)	322	94	297	288	229	87	241	85
	191	88	260	309	177	90	170	83
频域 (70)	293	93	280	301	225	92	260	91
	218	92	226	275	187	98	180	93

表 2 12.pgm 及对应 JPEG 图像中部分像素块的均值滤波及其对应超像素滤波的统计结果

图像类型	滤波生成图像中的像素块和原空域图像中对应像素块的差距 (8,8) (16,16) (24,24) (32,32) (40,40) (48,48) (56,56) (64,64)							
空域	453	79	297	389	308	73	263	75
	163	49	219	314	147	62	122	61
频域 (95)	456	82	297	391	307	71	266	68
	233	61	206	195	155	71	192	70
频域 (90)	456	86	299	394	308	73	268	77
	204	72	268	303	177	88	151	76
频域 (85)	454	84	301	387	306	79	268	78
	306	82	246	240	212	76	216	79
频域 (80)	455	90	293	383	317	82	268	73
	279	95	224	322	209	80	205	93
频域 (75)	457	94	301	388	318	87	262	78
	254	82	267	330	218	97	204	80
频域 (70)	461	93	281	408	310	94	284	87
	252	100	234	319	188	107	226	92

表 3 123.tif 及对应 JPEG 图像中部分像素块的维纳滤波及其对应超像素滤波的统计结果

图像类型	滤波生成图像中的像素块和原空域图像中对应像素块的差距 (8,8) (16,16) (24,24) (32,32) (40,40) (48,48) (56,56) (64,64)							
空域	201	293	500	580	216	270	301	339
	60	166	232	207	70	125	135	129
频域 (95)	344	343	530	662	229	319	321	342
	248	315	354	367	195	262	288	277
频域 (90)	436	323	535	661	222	322	312	375
	300	333	416	329	206	285	311	319
频域 (85)	433	377	529	664	225	315	321	364
	301	372	376	382	219	273	312	345
频域 (80)	532	432	557	682	216	349	325	351
	388	411	436	395	215	310	318	345
频域 (75)	536	494	550	713	219	399	330	349
	418	463	491	406	207	365	333	338
频域 (70)	565	420	534	720	229	364	325	349
	452	434	451	411	223	312	348	336

表 4 123.tif 及对应 JPEG 图像中部分像素块的均值滤波及其对应超像素滤波的统计结果

图像类型	滤波生成图像中的像素块和原空域图像中对应像素块的差距 (8,8) (16,16) (24,24) (32,32) (40,40) (48,48) (56,56) (64,64)							
空域	390	703	508	758	216	280	311	319
	94	378	313	313	113	190	187	211
频域 (95)	453	710	530	779	229	325	323	345
	352	332	452	466	218	286	290	303
频域 (90)	525	700	537	774	222	323	327	365
	402	365	498	613	220	310	320	326
频域 (85)	510	728	530	767	225	304	323	375
	382	546	449	507	228	292	323	366
频域 (80)	598	756	558	784	216	334	333	358
	399	802	490	511	228	334	326	358
频域 (75)	606	808	550	811	219	387	340	363
	506	459	446	532	214	386	341	357
频域 (70)	661	759	534	799	229	356	333	374
	593	541	468	482	223	353	332	373

表 5 1234.pgm 及对应 JPEG 图像中部分像素块的维纳滤波及其对应超像素滤波的统计结果

图像类型	滤波生成图像中的像素块和原空域图像中对应像素块的差距 (8,8) (16,16) (24,24) (32,32) (40,40) (48,48) (56,56) (64,64)							
空域	23	26	21	273	245	247	320	278
	7	6	8	469	421	107	117	135
频域 (95)	23	23	21	273	254	246	326	284
	22	22	24	381	356	144	144	155
频域 (90)	28	25	27	326	276	247	318	290
	28	24	26	393	387	173	184	169
频域 (85)	30	32	26	366	315	242	323	287
	36	31	32	410	477	179	252	222
频域 (80)	27	30	33	347	397	246	322	277
	32	31	32	420	511	176	231	201
频域 (75)	32	43	38	433	416	248	324	288
	33	49	33	457	504	211	269	210
频域 (70)	33	42	37	414	434	256	322	283
	33	43	33	577	512	247	306	229

表 6 1234.pgm 及对应 JPEG 图像中部分像素块的均值滤波及其对应超像素滤波的统计结果

图像类型	滤波生成图像中的像素块和原空域图像中对应像素块的差距 (8,8) (16,16) (24,24) (32,32) (40,40) (48,48) (56,56) (64,64)							
空域	23	26	21	1008	740	253	324	236
	15	10	14	667	549	187	291	169
频域 (95)	23	23	21	1014	737	253	329	241
	19	22	25	651	541	181	238	188
频域 (90)	28	25	27	1015	733	247	321	238
	28	24	25	671	532	245	239	194
频域 (85)	30	32	26	1023	735	244	324	234
	32	31	29	616	515	236	255	201
频域 (80)	27	30	33	1009	733	246	324	230
	33	30	31	593	565	257	269	219
频域 (75)	32	43	38	997	738	248	326	232
	33	48	33	771	465	248	310	244
频域 (70)	33	42	37	1000	729	256	326	237
	34	43	32	697	557	252	312	248

表 1、表 3、表 5 是维纳滤波及其对应超像素滤波的统计结果,表 2、表 4、表 6 是均值滤波及其对应超像素滤波的统计结果.标题第 2 行是块编号.第 1 列图像类型中,空域与频域指执行滤波操作的图像类型,频域后的数字指将空域图像 JPEG 压缩成频域图像时使用的质量因子,每两行为一组,每组数据中,上下两行分别是普通滤波对应的 η 与其对应的超像素滤波的 ξ .加粗的数字表示该数字在该组中更小,即对应滤波生成图像中的像素块和原空域图像中对应像素块的差距更小.

由表 1~表 6 可知,当原图像为空域图像时,与维纳滤波生成图像中的块相比,92%的超像素滤波生成图像中的块与空域图像中的块的差距更小.与均值滤波生成图像中的块相比,100%的超像素滤波生成图像中的块与空域图像中的块的差距更小.对

于 JPEG 图像,当质量因子为 95、90、85 时,80%以上的超像素滤波生成图像中的块与空域图像中的块的差距更小.其他 20%的块中,部分块的差距相等.当质量因子为 80、75、70 时,62.5%以上的超像素滤波生成图像中的块与空域图像中的块的差距更小.所以,不论对于空域图像还是 JPEG 图像,超像素滤波生成的图像都更接近原始空域图像,质量更高.

3.4.2 图的对比

本小节对三个数据库中所有图像进行统计验证.首先对图像做质量因子为 70 到 95、步长为 5 的 JPEG 压缩;然后对图像分别执行均值/维纳滤波及其对应的超像素滤波;接着分别对生成图像与未压缩图像的像素差、生成图像的质量进行统计验证.

(1) 图像像素差的均值统计验证

分别统计维纳/均值/超像素滤波生成图像与未压缩图像像素差的平均值,计算方式见式(17)和式(18).

$$\eta' = \frac{\sum_{k=1}^{\kappa} \sum_{i=1}^w \sum_{j=1}^h |c_{i,j}^k - s_{i,j}^k|}{\kappa} \tag{17}$$

$$\xi' = \frac{\sum_{k=1}^{\kappa} \sum_{i=1}^w \sum_{j=1}^h |c_{i,j}^k - z_{i,j}^k|}{\kappa} \tag{18}$$

其中,(i,j)是块中像素的坐标,k 为图像索引,κ 为图像库中的图像总数.维纳滤波与均值滤波的统计结果分别如表 7、表 8 所示.标题第 2 行是数据库,每个数据库对应的列是在该数据库上统计的未压缩图像与滤波生成图像像素差的平均差.第 1 列图像类型中,频域后的数字是压缩质量因子,每两行为一组,每组数据中,上下两行分别是普通滤波对应的 η' 与其对应的超像素滤波的 ξ'.加粗的数字表示在该

表 7 维纳滤波及对应超像素滤波生成图像平均差的统计结果

图像类型	不同数据库原始图像与滤波生成图像像素差的平均值		
	Bossbase-1.01	UCID	BOWS2
空域	632 196	942 925	675 815
	334 078	448 888	356 545
频域(95)	637 980	1 079 462	680 458
	422 001	795 205	442 385
频域(90)	654 384	1 114 517	697 619
	490 748	876 300	513 822
频域(85)	675 112	1 144 480	719 031
	549 551	941 106	575 638
频域(80)	695 905	1 171 874	740 250
	598 647	995 630	627 325
频域(75)	716 829	1 198 335	761 590
	641 966	1 044 816	673 000
频域(70)	738 444	1 222 598	783 205
	680 374	1 086 051	712 973

表 8 均值滤波及对应超像素滤波生成图像平均差的统计结果

图像类型	不同数据库原始图像与滤波生成图像像素差的平均值		
	Bossbase-1.01	UCID	BOWS2
空域	937 070	1 371 921	985 760
	588 170	834 781	623 010
频域(95)	937 311	1 441 766	985 961
	626 527	1 032 183	660 339
频域(90)	942 104	1 465 092	991 238
	661 766	1 085 582	696 925
频域(85)	948 950	1 483 299	998 179
	695 334	1 129 574	732 031
频域(80)	955 658	1 499 268	1 004 804
	724 969	1 167 829	763 148
频域(75)	962 691	1 514 211	1 011 763
	752 348	1 202 892	791 957
频域(70)	972 458	1 528 144	1 021 116
	778 951	1 233 402	819 383

组中该数字更小,即平均差更小.由表 7、表 8 可知,在三个图像库中,不论是空域图像还是 JPEG 图像,超像素滤波生成图像与未压缩图像像素差的平均值都小于维纳/均值滤波生成图像与原始图像像素差的平均值.超像素滤波在空域图像时效果最好,对于 JPEG 图像,质量因子越大,超像素滤波的效果就越明显.

(2) 生成图像的质量统计验证

本节利用峰值信噪比 PSNR(Peak Signal-to-Noise Ratio)和结构相似度 SSIM(Structural Similarity)衡量滤波生成图像的质量.

分别统计超像素滤波生成图像及维纳/均值滤波生成图像的 PSNR 和 SSIM 值,在计算 PSNR 与 SSIM 时用到的对比图像是未压缩空域图像,结果如表 9~表 12 所示.第 1 列图像类型中频域后的数字是 JPEG 压缩质量因子,每两行为一组,每组数据中,第 1 行是维纳/均值滤波对应的 PSNR/SSIM,

表 9 维纳滤波及对应超像素滤波生成图像的 PSNR 对比

图像类型	不同数据库中滤波生成图像的 PSNR/dB		
	Bossbase-1.01	UCID	BOWS2
空域	38.79	32.71	37.39
	40.90	35.75	39.99
频域(95)	38.67	31.34	37.32
	39.93	32.85	39.20
频域(90)	38.35	31.01	37.07
	39.10	32.16	38.38
频域(85)	37.97	30.71	36.76
	38.38	31.62	37.65
频域(80)	37.61	30.45	36.45
	37.79	31.17	37.04
频域(75)	37.26	30.20	36.14
	37.28	30.78	36.52
频域(70)	36.95	29.99	35.87
	36.86	30.47	36.10

表 10 均值滤波及对应超像素滤波生成图像的 PSNR 对比

图像类型	不同数据库中滤波生成图像的 PSNR/dB		
	Bossbase-1.01	UCID	BOWS2
空域	33.58	28.01	32.58
	36.81	31.17	35.87
频域(95)	33.58	27.65	32.59
	36.54	30.13	35.67
频域(90)	33.56	27.54	32.58
	36.26	29.83	35.40
频域(85)	33.53	27.45	32.56
	35.98	29.56	35.13
频域(80)	33.49	27.37	32.54
	35.73	29.34	34.88
频域(75)	33.45	27.30	32.52
	35.49	29.13	34.64
频域(70)	33.39	27.24	32.48
	35.26	28.95	34.43

表 11 维纳滤波及对应超像素滤波生成图像的 SSIM 对比

图像类型	不同数据库中滤波生成图像的 SSIM		
	Bossbase-1.01	UCID	BOWS2
空域	0.9393	0.8813	0.9403
	0.9852	0.9744	0.9855
频域(95)	0.9393	0.8628	0.9403
	0.9786	0.9307	0.9796
频域(90)	0.9383	0.8574	0.9393
	0.9717	0.9174	0.9732
频域(85)	0.9365	0.8535	0.9377
	0.9652	0.9070	0.9670
频域(80)	0.9345	0.8500	0.9358
	0.9594	0.8984	0.9614
频域(75)	0.9323	0.8465	0.9336
	0.9539	0.8904	0.9560
频域(70)	0.9299	0.8433	0.9315
	0.9490	0.8839	0.9514

表 12 均值滤波及对应超像素滤波生成图像的 SSIM 对比

图像类型	不同数据库中滤波生成图像的 SSIM		
	Bossbase-1.01	UCID	BOWS2
空域	0.9052	0.8289	0.9082
	0.9595	0.9277	0.9601
频域(95)	0.9054	0.8143	0.9084
	0.9559	0.8944	0.9569
频域(90)	0.9053	0.8096	0.9082
	0.9517	0.8844	0.9530
频域(85)	0.9049	0.8067	0.9078
	0.9474	0.8766	0.9488
频域(80)	0.9043	0.8043	0.9072
	0.9433	0.8701	0.9449
频域(75)	0.9035	0.8020	0.9064
	0.9394	0.8640	0.9411
频域(70)	0.9024	0.8001	0.9055
	0.9358	0.8589	0.9376

第 2 行是其对应的超像素滤波的 PSNR/SSIM. 其中表 9、表 11 的滤波类型为维纳滤波,表 10、表 12 的滤波类型为均值滤波. 加粗的数字表示图像质量更高.

由表 9 可知,维纳滤波对应的超像素滤波除质量因子为 70 时的 Bossbase-1.01 外,其他情况下比维纳滤波本身生成图像的 PSNR 值提高 0.02 dB~3.03 dB,平均提高 1.05 dB;由表 10 可知,均值滤波对应的超像素滤波比均值滤波本身生成图像的 PSNR 值提高 1.71 dB~3.28 dB,平均提高 2.44 dB;由表 11、表 12 可知,当滤波类型为维纳滤波和均值滤波时,SSIM 值分别提高 0.02~0.09 与 0.03~0.1,平均分别提高 0.04 与 0.05,验证了超像素滤波生成图像的质量更高.

由以上原理分析与统计验证可知,超像素滤波既保留了滤波操作综合考虑周围像素的优点,又避免了边缘被弱化的缺点. 不论是空域图像还是 JPEG 压缩图像,不论从块的角度还是图像的角度,与未加入超像素的滤波相比,超像素滤波生成图像与原始未压缩图像的差距更小,质量更高.

4 基于 SUFI 的非对称失真隐写

高质量的边信息图像可提升隐写图像的安全性^[18],本节将超像素滤波估计的图像作为边信息图像,结合非对称失真调整策略与自适应隐写框架,提出基于 SUFI 的非对称失真隐写方法. 接下来依次介绍方法框架及主要步骤、超像素去噪等细节.

4.1 方法框架及主要步骤

图 7 是提出方法的框架及主要步骤示意图,主要包含 6 个步骤:超像素索引生成、超像素去噪、超像素滤波、失真函数生成、信息嵌入及信息提取,如图中虚线框所示,其中前 5 个步骤属于发送端.

(1) 超像素索引生成. 对一幅 JPEG 载体图像 $\mathbf{X}=[x_{i,j}]_{w \times h}$,首先将其转成空域图像 $\mathbf{C}$. 如图中步骤 1 所示,使用超像素分割算法对 $\mathbf{C}$ 进行超像素划分,并得到图像的超像素索引.

(2) 超像素去噪. 太小的超像素就像图像的噪音,太多噪音会弱化超像素的作用,可以在滤波前先对小的超像素进行合并,这些小的超像素称为噪音超像素,该步骤称为超像素去噪. 如图中步骤 2 所示,首先过滤待合并超像素,接着通过计算超像素与周围像素的最小差值对其进行合并,完成超像素去噪,具体去噪细节请参考 4.2 节.

(3) 超像素滤波. 如图中步骤 3 所示,对空域图像 $\mathbf{C}$ 中的像素依次取滤波窗口,结合去噪后超像素索引,调整窗口像素值,利用 3.2 节的方法执行超像素滤波,滤波后的值作为边信息,记为 $\mathbf{X}'=[x'_{i,j}]_{w \times h}$.

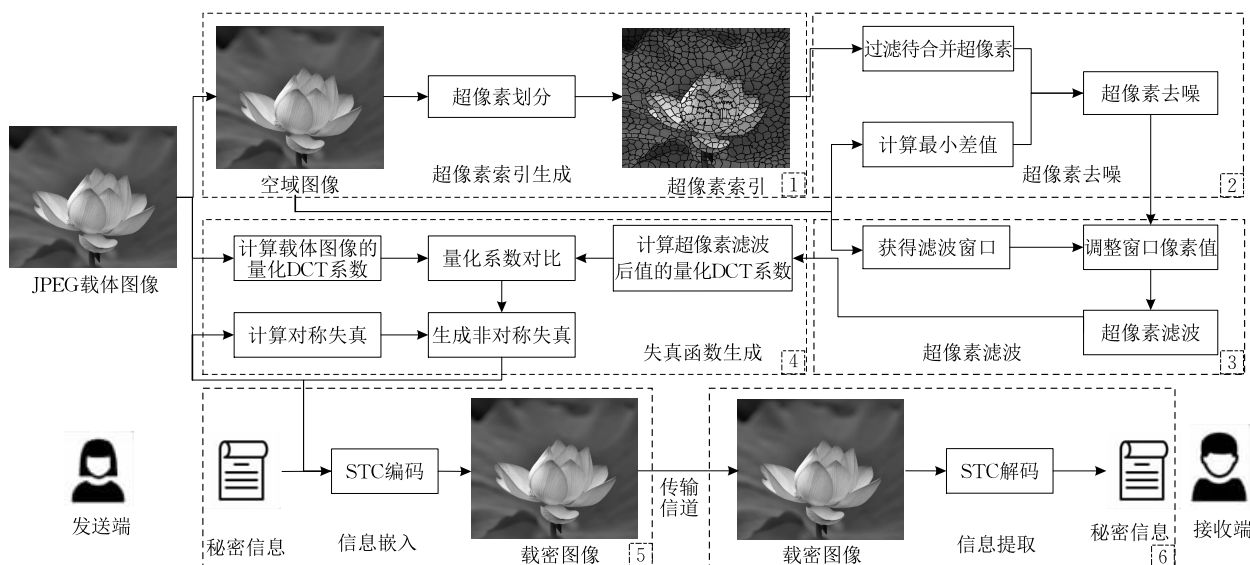


图7 方法的框架与主要步骤示意图

值得说明的是,在具体实现时,将滤波后的值直接保存而不再取整,以减少取整误差,进而设计更准确的失真函数,提高隐写算法的抗统计检测性能。

(4) 失真函数生成. 如图中步骤4所示,首先对 X' 进行 DCT 变换、量化(同样地,这里量化后的值不取整以减少取整误差),得到滤波后值的量化 DCT 系数,并将其与 X 中对应系数进行对比. 接着,根据 2.1 节中的非对称失真设计步骤获得系数“+1”和“-1”引起的失真 $\rho_{i,j}^+$ 和 $\rho_{i,j}^-$.

(5) 信息嵌入. 如图中步骤5所示,基于自适应隐写框架,利用得到的失真,结合 STC 编码,将秘密信息嵌入到载体图像 X 中,生成载密图像. 其中 STC 编码参数及信息个数等参数是发送端和接收端共享的密钥。

(6) 信息提取. 如图中步骤6所示,载密图像经过信道传输后被接收. 接收端利用双方共享的密钥,结合 STC 解码,从接收到的载密图像中提取出秘密信息。

4.2 超像素去噪

在一幅图像中,每个超像素包含的像素个数不完全相同. 为避免噪音超像素影响生成图像的质量,需要在滤波前将其合并到其他超像素中. 记噪音超像素的尺寸上限为 φ , 即像素个数 $< \varphi$ 的超像素为噪音超像素,其中 φ 简称为超像素去噪尺寸。

基于上述思想,本文给出超像素去噪算法,如算法2所示. 其中第2行与第4行的 $\text{Count}(\cdot, *)$ 、 $\text{Find}(\cdot, *)$ 分别表示统计“ $\cdot$ ”中“ $*$ ”的个数和位置,第7行的 $\min(\cdot)$ 表示“ $\cdot$ ”的最小值. 算法主要步骤如下:

(1) 待合并超像素查找. 如算法2第2~3行所示,在超像素索引中依次遍历,查找像素个数 $< \varphi$ 的噪音超像素,其内部像素称为待合并像素。

(2) 合并位置计算. 如算法2第4~6行所示,在待合并像素的周围像素中查找与其像素值差别最小且与其超像素索引不同的像素,其对应坐标称为最优合并位置。

(3) 超像素索引替换. 如算法2第7行所示,将最优合并位置处的超像素索引替换待合并像素对应的超像素索引。

为便于理解,附录1给出算法2的一个例子。

算法2. 超像素去噪算法.

输入: 一幅空域图像 $C = [c_{i,j}]_{w \times h}$, 该图像对应的超像素索引 $P = [p_{i,j}]_{w \times h}$, $0 \leq p_{i,j} \leq m$, m 表示该图像被划分的超像素数量

输出: 去噪后的超像素索引 P'

1. For $r = 1:m$
2. 计算 P 中索引等于 r 的个数 $n = \text{Count}(P, r)$.
3. If $n < \varphi$
4. 找出索引 r 在 P 中的位置 $[i, j]_{n \times 2} = \text{Find}(P, r)$.
5. For $r' = 1:n$
6. 以第 r' 个位置 (i, j) 为中心在 C 和 P 中取九宫格窗口 $J_C = [c'_{u,v}]_{3 \times 3}$ 和 $J_P = [p'_{u,v}]_{3 \times 3}$.
7. 计算 J_C 中 $c'_{u,v} = \min(|c'_{u,v} - c_{i,j}|)$ 且 $p'_{u,v} \neq p_{i,j}$ 的位置 (u, v) 对应的超像素索引值 $p'_{u,v}$, 并用该值更新 $p_{i,j}$.
8. End
9. End
10. End
11. Return $P' = P$.

5 实验结果与分析

为验证方法的有效性,本节从参数测试与对比测试两方面展开实验.接下来,本节首先介绍实验的参数设置,然后对超像素滤波参数和失真函数参数进行测试,最后与现有隐写方法进行对比.

5.1 实验设置

任取 Bossbase-1.01 库^[24]中的 2000 幅图像进行质量因子为 65、75、85、95 的 JPEG 压缩,将得到的 8000 幅压缩图像作为载体图像;使用经典 SLIC (Simple Linear Iterative Clustering)超像素分割算法^[23]进行超像素划分;除非特别说明,否则信息嵌入使用最优模拟嵌入;使用两种隐写分析^[26]方法对提出方法进行性能分析:一种提取面向 JPEG 图像的主流特征 DCTR (Discrete Cosine Transform Residual)^[27],并使用 FLD(Fisher Linear Discriminant)集成分类器^[28]进行训练分类.其中一半数据用于训练,一半数据用于分类;另一种使用基于深度学习的隐写分析方法 SRNet^[29]进行性能评估.表 13 是实验用到的其他部分参数设置.

表 13 实验设置

图像库	Bossbase-1.01 ^[24] 、IStego100K ^[30]
载体图像数量	8000
超像素数量	4096、5462、8192、10923、16384
超像素去噪尺寸	5、7、8、9、12、16、17、20、22、24、27、31、32、40、42、53、64
质量因子	65、75、85、95
非对称失真调整参数	0.5、0.55、0.6、0.65、0.7
滤波类型	均值滤波、中值滤波、高斯滤波、维纳滤波
信息嵌入率(bpnzAC)	0.1、0.2、⋯、0.5
对比的对称失真隐写方法	J-UNIWARD ^[9] 、UED ^[10] 、UERD ^[11]
对比的非对称失真隐写方法	文献[18、20-22]中的隐写方法
隐写分析方法	基于 DCTR 特征的隐写分析 ^[27] 、基于深度学习的隐写分析 SRNet ^[29]
载密图像数量	370 000

- 本文实验主要包括
- (1) 超像素滤波参数测试实验;
 - (2) 失真函数参数验证实验;
 - (3) 与对称失真隐写方法的对比实验;
 - (4) 与基于图像恢复的非对称失真隐写方法的对比实验;
 - (5) 与基于滤波的非对称失真隐写方法的对比实验;

- (6) SRNet 对本文方法的性能评估;
- (7) 载密图像质量验证实验.

5.2 超像素滤波参数测试实验

本节对超像素滤波中与隐写相关的参数进行测试,以寻找更合适的超像素和滤波参数,帮助超像素滤波与隐写更好的结合.实验包括不同滤波对比、超像素数量选择、超像素去噪尺寸选择等.

5.2.1 不同滤波的对比实验

本节分别验证均值滤波、中值滤波、高斯滤波和维纳滤波对提出隐写方法的影响.滤波窗口设置为 3×3,载体图像质量因子为 75,超像素个数设为 10 000,φ=9,在嵌入率为 0.1~0.5 bpnzAC 下进行信息模拟嵌入.抗统计检测结果如表 14 所示,值越大,表示抗统计检测性能越好.

表 14 不同滤波对应的检测错误率

滤波类型	不同嵌入率(bpnzAC)时的检测错误率				
	0.1	0.2	0.3	0.4	0.5
均值滤波	0.4729	0.4342	0.3743	0.3094	0.2360
中值滤波	0.4768	0.4261	0.3584	0.2805	0.2013
高斯滤波	0.4739	0.4331	0.3761	0.3059	0.2322
维纳滤波	0.4712	0.4322	0.3753	0.3042	0.2325

表 14 中,加粗和斜体数字分别是每一个嵌入率下最高与次高的检测错误率.从表中可以观察到两个现象:(1)没有一个滤波对应的检测错误率对所有的嵌入率都是最高的.比如均值滤波,当嵌入率为 0.2、0.4、0.5 时,检测错误率均为最高;当嵌入率为 0.1、0.3 时,检测错误率一般.再如高斯滤波,虽然只有在嵌入率为 0.3 时,检测错误率是最高的,但在嵌入率为 0.1、0.2、0.4 时,检测错误率都是次高的;(2)虽然相同嵌入率下各种滤波的检测错误率并不相同,但是彼此之间差距并不大.除了中值滤波,其他三个滤波的检测错误率差异甚至可以忽略不计(数量级为 10⁻³).由表 14 可知,当超像素个数为 10 000 时,相同嵌入率下不同滤波类型对本方法的抗统计检测性能影响不大.

5.2.2 超像素数量的选择实验

为验证超像素数量对提出方法的影响,分别统计质量因子为 65、75、85、95 的载体图像在 0.1~0.5 bpnzAC 嵌入率下进行模拟信息嵌入时方法的抗统计检测性能,滤波类型为均值滤波,φ=9.

结果如表 15 所示,第 1 列是载体图像的压缩质量因子,第 2 列是预设的超像素数量. Bossbase-1.01 库中的图像尺寸为 512×512,则根据预设超像素数量可计算出超像素大小分别约为 64、48、32、24、16. 每个嵌入率下,加粗和斜体数字分别表示最好和次

好的抗统计检测性能. 如对于压缩质量因子为 75 的载体图像, 当嵌入率为 0.3 时, 超像素数量 5462 与 4096 对应的抗统计检测性能较高.

表 15 不同超像素数量对应的检测错误率						
质量因子	超像素数量	不同嵌入率(bpnzAC)时的检测错误率				
		0.1	0.2	0.3	0.4	0.5
65	4096	0.4649	0.4281	0.3670	0.3003	0.2341
	5462	0.4713	0.4258	0.3666	0.3020	0.2320
	8192	0.4713	0.4241	0.3645	0.2996	0.2201
	10923	0.4699	0.4234	0.3629	0.2842	0.2117
	16384	0.4711	0.4183	0.3474	0.2737	0.1944
75	4096	0.4705	0.4307	0.3798	0.3156	0.2458
	5462	0.4711	0.4330	0.3833	0.3150	0.2429
	8192	0.4712	0.4362	0.3753	0.3095	0.2346
	10923	0.4743	0.4309	0.3742	0.3020	0.2258
	16384	0.4739	0.4324	0.3618	0.2886	0.2120
85	4096	0.4734	0.4427	0.3995	0.3419	0.2690
	5462	0.4760	0.4426	0.4037	0.3385	0.2670
	8192	0.4768	0.4470	0.4002	0.3360	0.2652
	10923	0.4797	0.4482	0.3956	0.3373	0.2602
	16384	0.4812	0.4435	0.3966	0.3221	0.2491
95	4096	0.4842	0.4567	0.4309	0.3899	0.3441
	5462	0.4869	0.4586	0.4298	0.3948	0.3431
	8192	0.4857	0.4638	0.4326	0.3941	0.3403
	10923	0.4916	0.4688	0.4370	0.3929	0.3428
	16384	0.4928	0.4678	0.4393	0.3946	0.3345

由表 15 可知: (1) 载体图像的质量因子不同, 嵌入率不同, 对应的拥有最高检测错误率的超像素数量也不同. 随着嵌入率的增大, 最高检测错误率对应的超像素数量在一定程度上与嵌入率成反比. 如当质量因子为 75, 嵌入率为 0.1~0.4 时, 最高检测错误率分别对应超像素数量 10923、8192、5462、4096; (2) 虽然超像素数量对检测错误率有一定影响, 但只要超像素数量在一定范围内, 检测错误率变化有限. 这意味着一幅图像对超像素数量的要求并不精确, 预设的超像素数量在一个大概范围即可. 如当质量因子为 75 时, 最高和次高检测错误率之间差的数量级为 10^{-3} . 这意味着给定嵌入率和载体图像, 合适的超像素数量是容易确定的.

由以上分析可知, 对于给定的嵌入率和载体图像, 可总结出适合该载体图像的超像素数量计算方式. 当滤波类型为均值滤波, 嵌入率为 0.1~0.5, 质量因子为 65 时, 合适的超像素大小约为 32~64; 质量因子为 75 时, 合适的超像素大小约为 24~64; 质量因子为 85、95 时, 合适的超像素大小约为 16~64. 根据合适的超像素大小, 可计算出与对应载体图像匹配的超像素数量, 从而获得更好的抗统计检测性能.

5.2.3 超像素去噪尺寸的选择实验

为验证超像素去噪时, 超像素去噪尺寸对隐写

方法抗统计检测性能的影响, 本节对超像素去噪尺寸 φ 进行讨论. 根据上一节中的超像素数量与嵌入率之间的关系, 我们使用质量因子为 75 的图像设置如下实验参数: 当嵌入率为 0.1 时, 超像素数量为 10923, 此时每个超像素大小约为 24, 参数 φ 设置为 5、9、12、16、20、24; 当嵌入率为 0.2 时, 超像素数量为 8192, 此时每个超像素大小约为 32, 参数 φ 设置为 7、12、17、22、27、32; 当嵌入率为 0.3 时, 超像素数量为 5462, 此时每个超像素大小约为 48, 参数 φ 设置为 8、16、24、32、40; 当嵌入率为 0.4 与 0.5 时, 超像素数量为 4096, 此时每个超像素大小约为 64, 参数 φ 设置为 9、20、31、42、53、64.

超像素去噪尺寸与检测错误率的对应关系如图 8 所示. 横轴是参数 φ , 纵轴是检测错误率, 不同标记分别对应不同的嵌入率. 由图 8 可知, 当嵌入率为 0.1 时, 超像素去噪尺寸对检测错误率的影响极小. 进一步观察可发现, 当 $\varphi=12$ 时达到峰值. 当嵌入率为 0.2~0.5 bpnzAC 时, 超像素去噪尺寸对检测错误率有一定影响, 尺寸与检测错误率基本成反比. 如嵌入率为 0.2 时, $\varphi=7$ 的检测错误率最高. 之后随着 φ 的增大, 检测错误率呈现缓慢的下降趋势. 为兼顾不同的信息嵌入率, 接下来基于均值滤波的超像素滤波相关实验将 φ 设置为 7 到 12 之间的一个中间值 9.

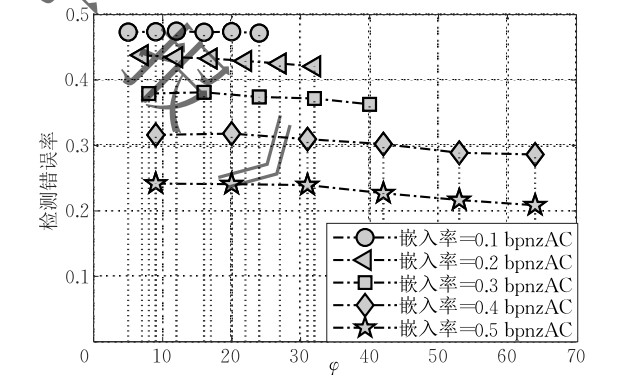


图 8 参数 φ 与检测错误率的对应关系

5.3 失真函数参数验证实验

本节对失真函数实现过程中的部分参数进行讨论, 包括边信息图像构造过程中的取整验证、非对称失真调整参数的验证等, 本节实验用到的滤波类型为均值滤波.

5.3.1 取整验证实验

由于基于对称失真生成非对称失真的理论前提是量化过程的取整误差^[20-21], 所以本文在设计与非对称失真函数的过程中, 对取整误差进行了额外

的关注. 在整个过程中, 有两个地方需要考虑取整误差: 一是当利用 JPEG 压缩图像生成边信息图像时, 对 JPEG 压缩图像进行超像素滤波后, 如果保存成图像, 就需要对其取整, 该步取整操作记为 $Prec_R$; 此时尚未完成非对称失真函数的设计, 该数据可以不保存成图像, 即不取整, 该步不取整操作记为 $Prec_NR$; 二是在设计非对称失真函数时, 由于要与 JPEG 图像中的 DCT 系数进行对比, 所以需要将超像素滤波值转成 DCT 系数并量化, 量化后的数据可以取整保存成图像, 该步取整操作记为 $Quan_R$; 也可以直接使用量化后的非取整值, 该步不取整操作记为 $Quan_NR$. 这些数值一般小数位较多, 本节还考虑了此时保留有限位小数(比如 4 位)的情形, 记为 $Quan_R4$.

表 16 给出不同取整类型对应方法的检测错误率, 第 1 列的取整类型“ $[A, B]$ ”为两个取整类型的组合. 如 $[Prec_NR, Quan_R]$ 表示对 JPEG 图像做超像素滤波后不取整, 对量化后的数据取整. 粗体和斜体分别表示不同的嵌入率下最高与次高的检测错误率. 由表 16 可知: 最高和次高的检测错误率主要集中在前两行: $[Prec_NR, Quan_NR]$ 和 $[Prec_NR, Quan_R4]$, 这说明任何一个阶段的取整都会导致检测错误率的下降, 特别是图像超像素滤波后的取整, 会导致检测错误率的大幅下降. 这是因为每次取整都会带来一定的信息损失, 使非对称失真函数设计出现偏差, 进而导致检测错误率的下降.

取整类型	不同嵌入率(bpnzAC)时的检测错误率				
	0.1	0.2	0.3	0.4	0.5
$[Prec_NR, Quan_NR]$	0.4743	0.4343	0.3807	0.3098	0.2430
$[Prec_NR, Quan_R4]$	0.4746	0.4339	0.3823	0.3165	0.2435
$[Prec_NR, Quan_R]$	0.4660	0.4191	0.3611	0.2858	0.2143
$[Prec_R, Quan_NR]$	0.4473	0.3690	0.2846	0.2040	0.1349
$[Prec_R, Quan_R4]$	0.4442	0.3675	0.2834	0.2064	0.1347
$[Prec_R, Quan_R]$	0.4458	0.3663	0.2809	0.2068	0.1389

5.3.2 调整参数验证实验

根据 2.1 节非对称失真隐写策略中非对称失真生成步骤可知(式(1)), 非对称失真被赋值为 α 与对称失真的乘积. 本小节讨论调整参数 α 对隐写方法的影响.

隐写方法常用的 α 值有 0.6、0.65、0.7, 图 9 给出 $\alpha=0.6$ 、0.65、0.7、0.75、0.8 时隐写方法的检测错误率, 不同标记代表不同 α 对应的检测错误率. 点线是辅助高斯线, 其目的是更清楚地显示检测错误率的变化趋势. 由图 9 可知, 每个 α 对应的检测错误

率均呈现先扬后抑的类高斯走向. 当 $\alpha<0.65$ 时, 检测错误率呈上升趋势, $\alpha=0.65$ 时达到顶峰, 接着随着 α 的增大, 检测错误率逐渐降低. 这是因为当 α 较大时, 会越来越接近对称失真隐写方法的抗统计检测性能. 极限情况下, 当 $\alpha=1$ 时, 退化成对称失真隐写方法. 随着 α 的减小, 非对称失真函数的作用逐渐明显. 但由于根据 JPEG 压缩图像估计的边信息图像并不是真正的原始未压缩图像, 其与原始未压缩图像之间可能存在一定差异, 需要考虑容错性. 在估计边信息图像时带来的错误会直接影响到抗统计检测能力, 故 α 取一个中间值时效果最好.

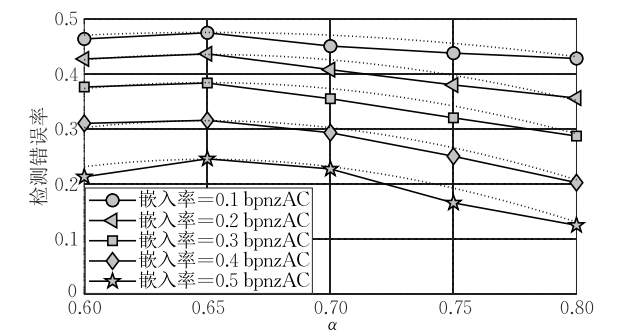


图 9 参数 α 与检测错误率的对应关系

5.4 与对称失真隐写方法的对比实验

非对称失真是在对称失真的基础上构造而得的. 本节与三种对称失真隐写典型方法: J-UNIWARD、UED、UERD 进行对比, 其中在对称失真基础上加入超像素滤波后的隐写方法分别记为 SP-J、SP-UED、SP-UERD. 不同嵌入率下使用表 15 中合适的超像素数量, 滤波类型为均值滤波, $\varphi=9$.

表 17 给出方法的抗统计检测结果, 首列的质量因子是载体图像的 JPEG 压缩质量因子. 每个质量因子对应的行为一组, 每组的奇数行是对称失真隐写方法, 偶数行是对应的非对称失真隐写方法. 粗体表示在对应相邻两行中, 该行对应的方法抗统计检测性能更高. 由表中数据可知: (1) 粗体都出现在偶数行, 验证了这三种对称失真隐写方法在加入超像素滤波后, 得到的非对称失真隐写方法的检测错误率都有提升. 提升范围分别为 0.2%~15.0%、0.6%~7.0%、0.5%~10.8%, 平均分别提升 7.6%、4.2%、6.0%. 这是因为非对称失真在对称失真的基础上考虑了 JPEG 压缩的取整误差; (2) 检测错误率的提升幅度与 JPEG 压缩质量因子呈反比. 三种方法在质量因子为 65、75、85、95 时的平均提升幅度分别为 10.8%、9.1%、7.3%、3.4%, 4.5%、4.5%、4.0%、3.9%、8.2%、6.7%、5.9%、3.2%. 这是因为当质量

表 17 与对称失真隐写方法的抗统计检测性能对比

质量因子	隐写方法	不同嵌入率(bpnzAC)时的检测错误率				
		0.1	0.2	0.3	0.4	0.5
65	J-UNIWARD ^[9]	0.4387	0.3406	0.2460	0.1595	0.1005
	SP-J	0.4713	0.4281	0.3670	0.3020	0.2341
	UED ^[10]	0.3785	0.2380	0.1287	0.0613	0.0204
	SP-UED	0.4030	0.3059	0.1960	0.1028	0.0448
	UERD ^[11]	0.4299	0.3325	0.2396	0.1602	0.1042
	SP-UERD	0.4618	0.4116	0.3438	0.2684	0.1921
75	J-UNIWARD ^[9]	0.4512	0.3680	0.2723	0.1899	0.1209
	SP-J	0.4743	0.4362	0.3833	0.3156	0.2458
	UED ^[10]	0.3888	0.2594	0.1549	0.0816	0.0298
	SP-UED	0.4115	0.3228	0.2252	0.1252	0.0557
	UERD ^[11]	0.4402	0.3565	0.2641	0.1870	0.1157
	SP-UERD	0.4643	0.4127	0.3480	0.2721	0.1997
85	J-UNIWARD ^[9]	0.4683	0.3960	0.3173	0.2347	0.1618
	SP-J	0.4812	0.4482	0.4037	0.3419	0.2690
	UED ^[10]	0.4155	0.3011	0.2003	0.1153	0.0559
	SP-UED	0.4271	0.3423	0.2641	0.1694	0.0877
	UERD ^[11]	0.4583	0.3858	0.2932	0.2170	0.1513
	SP-UERD	0.4734	0.4239	0.3651	0.3015	0.2356
95	J-UNIWARD ^[9]	0.4908	0.4577	0.4092	0.3427	0.2695
	SP-J	0.4928	0.4688	0.4393	0.3948	0.3441
	UED ^[10]	0.4709	0.4180	0.3405	0.2462	0.1510
	SP-UED	0.4766	0.4400	0.3811	0.3129	0.2094
	UERD ^[11]	0.4827	0.4398	0.3857	0.3236	0.2473
	SP-UERD	0.4872	0.4640	0.4170	0.3673	0.3022

因子很高时,载体图像的细节很多,能供信息嵌入的像素位置也很多,对称失真隐写方法的抗统计检测性能已经很高,提升空间有限.当质量因子较低时,载体图像细节较少,供信息嵌入的像素区域也较少,

现有对称失真隐写方法的抗统计检测性能相对不是很高,提升空间较大,故本文方法的抗统计检测性能提升效果逐渐明显.

5.5 与基于图像恢复的非对称失真隐写对比实验

本节分别在抗统计检测性能和耗时两方面与基于图像恢复的非对称失真隐写典型方法^[18](本节简记为Li的方法)进行比较.任取 Bossbase-1.01 和 IStego100K 库中 2000 幅图像,其中 Bossbase-1.01 中为 512×512 的空域图像,做质量因子为 65~95 的 JPEG 压缩生成载体图像;IStego100K 中为 1024×1024 的频域图像,每幅图像的质量因子为 65~95 中的随机值,直接将库中图像作为载体图像.

使用两种方法在不同嵌入率下进行信息模拟嵌入,其中本文方法使用的滤波类型为维纳滤波.图 10 是两种方法生成边信息图像的耗时对比,其中横轴是图像索引,纵轴是耗时,单位是秒(s).为显示得更清楚,在 2000 个数值(对应 2000 幅图像)中以步长 20 依次抽取其中的 100 个值用于显示.从图中可以观察到,当图像大小为 512×512 时,Li^[18]的方法生成边信息图像耗时约为 12 s,本文方法生成边信息图像耗时约为 1.5 s,降低了约 87%.当图像大小为 1024×1024 时,Li^[18]的方法生成边信息图像耗时约为 57 s,本文方法生成边信息图像耗时约为 5.7 s,降低了约 90%.从耗时角度看,当图像较大时,Li^[18]的方法实用性一般.

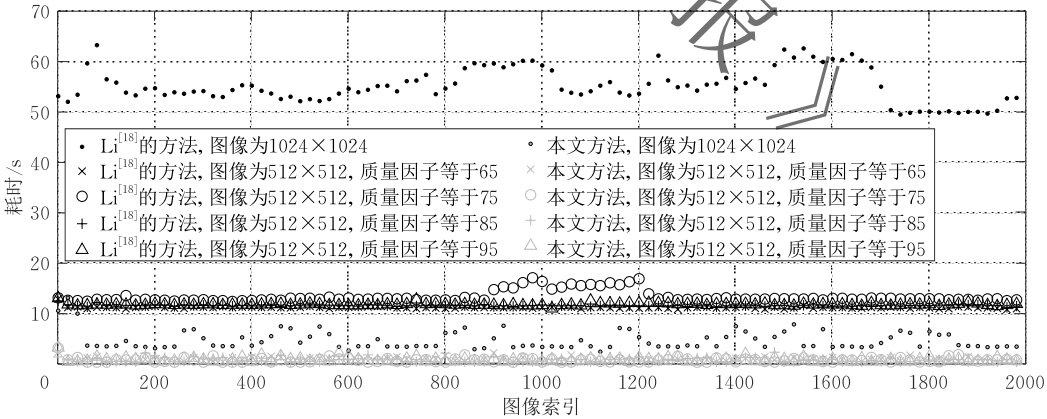


图 10 与基于图像恢复的非对称失真隐写方法^[18]的耗时对比

表 18 是抗统计检测性能对比.首列是载体图像的 JPEG 压缩质量因子;第 2 列是隐写方法;第 3 到 7 列是检测错误率,数据越大,表示方法抗统计检测性能越好;第 8 列是方法的平均耗时,单位是秒(s),数据越小,表示方法耗时越少,实用性越强.在每种质量因子下,前两行均为 Li^[18]的方法,a、b 分别表示 SSRQC 算法使用 1 次和 2 次^[18]迭代.从表中可

以看出,当迭代次数为 1 时,方法的检测错误率远远低于本文方法;当迭代次数为 2、嵌入率为 0.1~0.2 时,两种方法的检测错误率基本相当.当嵌入率为 0.3~0.5 时,本文方法逊于 Li^[18]的方法,平均差距为 0.029.这说明,与基于图像恢复的非对称失真隐写典型方法^[18]相比,本文方法在较低嵌入率下保持基本相当的检测错误率,且耗时较少.

表 18 与基于图像恢复的非对称失真隐写方法^[18]的抗统计检测性能对比

质量因子	隐写方法	不同嵌入率(bpnzAC)时的检测错误率					耗时/s
		0.1	0.2	0.3	0.4	0.5	
65	Li ^[18] 的方法 ^a	0.4334	0.3381	0.2390	0.1529	0.0919	6.5394
	Li ^[18] 的方法 ^b	0.4684	0.4413	0.4088	0.3524	0.2877	11.1724
	本文方法	0.4766	0.4346	0.3793	0.3210	0.2518	0.8017
	Li ^[18] 的方法 ^a	0.4485	0.3590	0.2640	0.1837	0.1111	6.5356
75	Li ^[18] 的方法 ^b	0.4734	0.4484	0.4114	0.3652	0.2935	13.3962
	本文方法	0.4790	0.4427	0.3885	0.3245	0.2555	0.7263
	Li ^[18] 的方法 ^a	0.4625	0.3945	0.3114	0.2231	0.1493	6.5439
	Li ^[18] 的方法 ^b	0.4763	0.4529	0.4278	0.3891	0.3238	11.5776
	本文方法	0.4840	0.4504	0.4063	0.3515	0.2754	0.8141
95	Li ^[18] 的方法 ^a	0.4858	0.4514	0.3926	0.3307	0.2539	6.5604
	Li ^[18] 的方法 ^b	0.4851	0.4637	0.4427	0.4143	0.3753	11.8056
	本文方法	0.4919	0.4708	0.4420	0.4036	0.3490	0.8241
	本文方法	0.4919	0.4708	0.4420	0.4036	0.3490	0.8241

5.6 与基于滤波的非对称失真隐写的对比实验

本节与基于滤波的非对称失真隐写方法进行对比,使用 JPEG 压缩质量因子为 65~95 的图像进行模拟信息嵌入. 现有基于滤波的非对称失真隐写方法有维纳滤波与均值滤波两种. Wang^[20]使用维纳滤波生成边信息图像,并通过反复迭代计算 α 为兼顾该方法思想与耗时,本节对其使用常见的三种 α 值进行实验,结果如表 19 所示,该表中本文方法的滤波类型为维纳滤波. Wang^[21]使用均值滤波对 JPEG 图像的 DCT 块边缘进行补偿生成边信息图像;Yu^[22]直接用均值滤波图像作为边信息图像. 本节对两种方法使用与原文相同的 α 值进行实验,结果如表 20 所示,该表中本文方法的滤波类型为均值滤波.

表 19 与基于维纳滤波的非对称失真隐写方法的抗统计检测性能对比

质量因子	隐写方法	不同嵌入率(bpnzAC)时的检测错误率				
		0.1	0.2	0.3	0.4	0.5
65	Wang ^[21] 的方法 ^a	0.4608	0.4250	0.3814	0.3262	0.2679
	Wang ^[21] 的方法 ^b	0.4714	0.4343	0.3839	0.3213	0.2494
	Wang ^[21] 的方法 ^c	0.4741	0.4354	0.3788	0.3069	0.2396
	本文方法	0.4766	0.4346	0.3793	0.3210	0.2518
75	Wang ^[21] 的方法 ^a	0.4585	0.4252	0.3830	0.3337	0.2666
	Wang ^[21] 的方法 ^b	0.4705	0.4348	0.3919	0.3320	0.2660
	Wang ^[21] 的方法 ^c	0.4746	0.4403	0.3886	0.3226	0.2446
	本文方法	0.4790	0.4427	0.3885	0.3245	0.2555
85	Wang ^[21] 的方法 ^a	0.4617	0.4267	0.3869	0.3382	0.2769
	Wang ^[21] 的方法 ^b	0.4712	0.4384	0.4029	0.3483	0.2827
	Wang ^[21] 的方法 ^c	0.4768	0.4493	0.4061	0.3513	0.2766
	本文方法	0.4840	0.4504	0.4063	0.3515	0.2754
95	Wang ^[21] 的方法 ^a	0.4768	0.4390	0.4055	0.3705	0.3227
	Wang ^[21] 的方法 ^b	0.4833	0.4531	0.4140	0.3804	0.3336
	Wang ^[21] 的方法 ^c	0.4848	0.4609	0.4322	0.3929	0.3444
	本文方法	0.4919	0.4708	0.4420	0.4036	0.3490

表 20 与基于均值滤波的非对称失真隐写方法的抗统计检测性能对比

质量因子	隐写方法	不同嵌入率(bpnzAC)时的检测错误率				
		0.1	0.2	0.3	0.4	0.5
65	Wang ^[21] 的方法	0.4462	0.3745	0.2947	0.2074	0.1323
	Yu ^[22] 的方法	0.4621	0.4209	0.3666	0.2996	0.2270
	本文方法	0.4713	0.4281	0.3754	0.3099	0.2395
	Wang ^[21] 的方法	0.4466	0.3846	0.3019	0.2076	0.1303
75	Yu ^[22] 的方法	0.4689	0.4326	0.3782	0.3064	0.2363
	本文方法	0.4743	0.4362	0.3833	0.3156	0.2458
	Wang ^[21] 的方法	0.4639	0.4171	0.3313	0.2328	0.1456
	Yu ^[22] 的方法	0.4747	0.4438	0.3999	0.3392	0.2600
	本文方法	0.4812	0.4482	0.4037	0.3419	0.2690
95	Wang ^[21] 的方法	0.4827	0.4517	0.4089	0.3414	0.2400
	Yu ^[22] 的方法	0.4832	0.4599	0.4297	0.3908	0.3348
	本文方法	0.4928	0.4688	0.4393	0.3948	0.3441
	本文方法	0.4928	0.4688	0.4393	0.3948	0.3441

在表 19 中,前 3 行均为基于维纳滤波的非对称失真隐写典型方法, a 、 b 、 c 分别是 α 为 0.6、0.65、0.7 时的检测错误率. 可以看出,当质量因子为 85、95 时,本文方法的抗统计检测性能基本都是最好;当质量因子为 65、75 时,在较低嵌入率下性能较好. 在表 20 中,首列是载体图像的 JPEG 压缩质量因子,第 2 列是隐写方法;加粗数据表示该数据在对应嵌入率下检测错误率最高. 可以看出,本文提出的方法都拥有最高的检测错误率. 相较现有最新的基于均值滤波的非对称失真隐写典型方法,本文方法对应的检测错误率提升约 0.3%~12%. 综合表 19、表 20 中的数据可知,本文提出的方法在大多数情况下都拥有较高的检测错误率,原因是本文是在滤波的过程中增加了对图像中物体边缘、轮廓等的考虑. 这样生成的边信息图像既保留了基于滤波构造边信息图像算法的优点,又避免了生成图像中物体边缘被弱化的缺点. 最极端的情况是超像素数量为 1,即整幅图为 1 个超像素,退化为基于滤波的非对称失真隐写方法. 故本文提出的方法优于基于滤波的非对称失真隐写方法.

5.7 SRNet 对本文方法的性能评估

虽然目前非对称失真隐写方法(如本文的对比方法)普遍通过提取高维特征如 DCTR 进行训练分类来评估方法性能,鉴于基于深度学习的隐写分析方法是目前较为先进的性能评估方法之一,本节我们用基于深度学习的经典隐写分析方法 SRNet^[29]对本文方法进行性能评估.

SRNet 在 TensorFlow 1.3.0 版本上实现,在 NVIDIA RTX 3090 GPU 上运行,超参数均与 SRNet 原文中保持一致. 取 Bossbase-1.01 库 10000 幅 512×512 的图像进行信息嵌入,质量因子为 65~95,嵌入

率为 0.1~0.5,训练集、验证集、测试集分别为 5000、1000、4000 张图像. 结果如表 21 所示.

表 21 SRNet 对本文方法的性能评估

质量因子	隐写方法	不同嵌入率(bpnzAC)时的方法检测准确率				
		0.1	0.2	0.3	0.4	0.5
65	J-UNIWARD ^[9]	69.2496	78.7624	85.2000	90.9752	95.4001
	本文方法	68.0621	77.7248	84.9001	89.0502	93.6127
75	J-UNIWARD ^[9]	68.1121	77.8498	83.8125	89.6751	93.2002
	本文方法	67.6996	77.1873	82.7250	87.5502	91.9878
85	J-UNIWARD ^[9]	66.0245	76.7998	83.4250	89.4251	92.1252
	本文方法	64.4370	76.5747	82.9124	87.3376	91.5002
95	J-UNIWARD ^[9]	50.1125	64.7745	73.3497	77.9874	83.7250
	本文方法	50.0250	63.7870	72.2622	77.3748	83.2125

SRNet 利用检测准确率评估隐写方法,值越大,表示检测准确率越高,代表评估的方法越不安全. 由表 21 可知,相较经典隐写方法 J-UNIWARD,本文方法的检测准确率较低. 如当质量因子为 75 和 85、嵌入率为 0.4 时,方法检测准确率分别降低了 2.12 与 2.09,验证了本文方法的安全性.

5.8 载密图像质量验证实验

本节统计经典隐写方法 J-UNIWARD 和本文方法隐写前后图像的 PSNR 和 SSIM 值,以验证载密图像的质量,结果如表 22、表 23 所示.

表 22 载密图像的 PSNR 对比

质量因子	隐写方法	不同嵌入率下载密图像的 PSNR/dB				
		0.1	0.2	0.3	0.4	0.5
65	J-UNIWARD ^[9]	56.02	52.42	50.22	48.59	47.28
	本文方法	56.04	52.45	50.25	48.62	47.30
75	J-UNIWARD ^[9]	57.48	53.86	51.59	49.92	48.55
	本文方法	57.47	53.84	51.61	49.94	48.58
85	J-UNIWARD ^[9]	59.29	55.79	53.61	51.95	50.56
	本文方法	59.29	55.81	53.64	51.98	50.59
95	J-UNIWARD ^[9]	62.64	59.48	57.52	55.98	54.63
	本文方法	62.68	59.50	57.55	56.01	54.66

表 23 载密图像的 SSIM 对比

质量因子	隐写方法	不同嵌入率下载密图像的 SSIM				
		0.1	0.2	0.3	0.4	0.5
65	J-UNIWARD ^[9]	0.9992	0.9988	0.9982	0.9974	0.9966
	本文方法	0.9996	0.9992	0.9986	0.9978	0.9968
75	J-UNIWARD ^[9]	0.9993	0.9989	0.9984	0.9979	0.9972
	本文方法	0.9998	0.9994	0.9989	0.9982	0.9975
85	J-UNIWARD ^[9]	0.9993	0.9990	0.9987	0.9983	0.9978
	本文方法	0.9998	0.9995	0.9991	0.9987	0.9982
95	J-UNIWARD ^[9]	0.9994	0.9992	0.9990	0.9988	0.9985
	本文方法	0.9998	0.9997	0.9995	0.9992	0.9990

由表 22 可知,本文方法的 PSNR 值与经典的自适应隐写 J-UNIWARD 基本持平,均保持了很高的 PSNR 值. 这意味着本文方法与现有自适应隐写方法

一样,对图像的影响均微乎其微,原因是两个方法均在图像像素最低有效位上进行嵌入,且修改位置与修改幅度类似. 由表 23 可知,本文方法与 J-UNIWARD 生成图像的 SSIM 值都很高,隐写前后的图像都保持了很高的结构相似度. 相较 J-UNIWARD,本文方法的 SSIM 值略有提高,这是因为本文方法在边缘与轮廓处更接近载体图像.

6 结束语

隐写失真函数设计是隐写领域的研究热点之一. 近年来,有实验与理论证实相较对称失真函数,非对称失真函数能更好地预测信息嵌入时的修改方向. 目前,基于图像恢复的非对称失真隐写方法耗时较长;基于滤波的非对称失真隐写方法抗统计检测性能有待提高. 为此,本文首先分析指出现有基于滤波的非对称失真隐写会弱化图像中物体的轮廓与边缘,进而降低生成边信息图像的质量,并进一步降低方法的抗统计检测性能;基于此分析,本文在滤波操作时加入增强图像中物体边缘与轮廓的超像素,提出超像素滤波的概念与算法;接着,为避免过小的超像素成为图像的“干扰噪声”影响生成图像的质量,给出一种超像素去噪算法,以更好地描绘物体边缘与轮廓;最后,利用生成的边信息图像,结合现有非对称失真函数设计策略,得到能够更准确指示信息嵌入修改方向的非对称失真函数,进而给出基于超像素滤波的非对称失真隐写方法,提高载密图像的抗统计检测性能. 经原理分析与统计验证,超像素滤波构造的边信息图像质量更高;在五种嵌入率与四种质量因子下的实验表明,与对称失真隐写典型方法 J-UNIWARD、UED、UERD 相比,本文提出方法的抗统计检测性能分别平均提升 7.6%、4.2%、6.0%;与基于图像恢复的非对称失真隐写典型方法相比,本文提出的方法耗时降低约 87%;与现有最新的基于均值滤波的非对称失真隐写方法相比,抗统计检测性能提升了 0.3%~12%.

下一步我们将从三个方面继续开展研究:第一,用更好的超像素算法替换本文使用的 SLIC 分割算法;第二,寻找替代超像素的、能够更好地定义物体物理意义的方法. 通过这些方法快速、高效地构造质量更高的边信息图像,从而进一步提升隐写图像的安全性;第三,在耗时较短的前提下,构造出与基于图像恢复的隐写方法的边信息图像质量相当甚至超过的图像,进而达到甚至超过其抗统计检测性能依然是非对称失真隐写的目标之一.

参 考 文 献

- [1] Fridrich J. *Steganography in Digital Media: Principles, Algorithms, and Applications*. New York, USA: Cambridge University Press, 2010
- [2] Fu Zhang-Jie, Wang Fan, Sun Xing-Ming, Wang Yan. Research on steganography of digital images based on deep learning. *Chinese Journal of Computers*, 2020, 43(9): 1656-1672(in Chinese)
(付章杰, 王帆, 孙星明, 王彦. 基于深度学习的图像隐写方法研究. *计算机学报*, 2020, 43(9): 1656-1672)
- [3] Zhang Yi, Luo Xiang-Yang, Wang Jin-Wei, et al. Research progress on digital image robust steganography. *Journal of Image and Graphics*, 2022, 27(1): 3-26(in Chinese)
(张祎, 罗向阳, 王金伟等. 数字图像鲁棒隐写综述. *中国图象图形学报*, 2022, 27(1): 3-26)
- [4] Filler T, Judas J, Fridrich J. Minimizing additive distortion in steganography using syndrome-trellis codes. *IEEE Transactions on Circuits and Systems for Video Technology*, 2011, 6(3): 920-935
- [5] Li W, Zhang W, Li L, et al. Designing near-optimal steganographic codes in practice based on polar codes. *IEEE Transactions on Communications*, 2020, 68(7): 3948-3962
- [6] Sedighi V, Cogan R, Fridrich J. Content-adaptive steganography by minimizing statistical detectability. *IEEE Transactions on Information Forensics and Security*, 2016, 11(2): 221-234
- [7] Su W, Ni J, Hu X, Huang J. New design paradigm of distortion cost function for efficient JPEG steganography. *Signal Processing*, 2022, 190: 108319
- [8] Tang W, Li B, Barni M, Li J, Huang J. Improving cost learning for JPEG steganography by exploiting JPEG domain knowledge. *IEEE Transactions on Circuits and Systems for Video Technology*, 2022, 32(6): 4081-4095
- [9] Holub V, Fridrich J, Denemark T. Universal distortion function for steganography in an arbitrary domain. *EURASIP Journal on Information Security*, 2014, 1(2014): 1-13
- [10] Luo L, Ni J, Shi Y. Uniform embedding for efficient JPEG steganography. *IEEE Transactions on Information Forensics and Security*, 2014, 9(5): 814-825
- [11] Guo L, Ni J, Shi Y. An efficient JPEG steganographic scheme using uniform embedding//*Proceedings of the IEEE International Workshop on Information Forensics and Security*. Tenerife, Spain, 2012: 169-174
- [12] Su W, Ni J, Li X, Shi Y. A new distortion function design for JPEG steganography using the generalized uniform embedding strategy. *IEEE Transactions on Circuits and Systems for Video Technology*, 2018, 28(12): 3545-3549
- [13] Denemark T, Fridrich J. Model based steganography with precover. *Electronic Imaging*, 2017, 7: 56-66
- [14] Butora J and Fridrich J. Revisiting perturbed quantization//*Proceedings of the 9th International Conference on Information Hiding and Multimedia Security*. Brussels, Belgium, 2021: 125-136
- [15] Denemark T, Fridrich J. Side-informed steganography with additive distortion//*Proceedings of the IEEE International Workshop on Information Forensics and Security*. Roma, Italy, 2015: 1-6
- [16] Denemark T, Fridrich J. Steganography with multiple JPEG images of the same scene. *IEEE Transactions on Information Forensics and Security*, 2017, 12(10): 2308-2319
- [17] Zhao Xin, Wang Yao-Fei, Chen Ke-Jiang, et al. Spatial image steganography based on side information estimated by super resolution. *Journal of Image and Graphics*, 2022, 27(1): 226-237(in Chinese)
(赵鑫, 王尧飞, 陈可江等. 引入超分辨率下采样误差的图像边信息估计隐写. *中国图象图形学报*, 2022, 27(1): 226-237)
- [18] Li W, Chen K, Zhang W, et al. JPEG steganography with estimated side-information. *IEEE Transactions on Circuits and Systems for Video Technology*, 2020, 30(7): 2288-2294
- [19] Zhao C, Zhang J, Ma S, et al. Reducing image compression artifacts by structural sparse representation and quantization constraint prior. *IEEE Transactions on Circuits and Systems for Video Technology*, 2017, 27(10): 2057-2071
- [20] Wang Z, Qian Z, Zhang X, et al. On improving distortion functions for JPEG steganography. *IEEE Access*, 2018, 6: 74917-74930
- [21] Wang Z, Yin Z, Zhang X. Asymmetric distortion function for JPEG steganography using block artifact compensation. *International Journal of Digital Crime and Forensics*, 2019, 11(1): 90-96
- [22] Yu X, Chen K, Wang Y, et al. Robust adaptive steganography based on generalized domain and expanded embedding domain. *Signal Processing*, 2020, 168: 107374
- [23] Achanta R, Shaji A, Smith K, et al. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012, 34(11): 2274-2282
- [24] Bas P, Filter T, Pevny T. Break our steganographic system: The ins and outs of organizing BOSS//*Proceedings of the Information Hiding Conference*. Prague, Czech Republic, 2011, 96(454): 488-499
- [25] Schaefer G, Stich M. UCID: An uncompressed color image database//*Proceedings of the International Society for Optical Engineering*. San Jose, CA, USA, 2004, 5307: 472-480
- [26] Wang Shuo-Zhong, Zhang Xin-Peng, Zhang Wei-Ming. Recent advances in image-based steganalysis research. *Chinese Journal of Computers*, 2009, 32(7): 1247-1263(in Chinese)
(王朔中, 张新鹏, 张卫明. 以数字图像为载体的隐写分析研究进展. *计算机学报*, 2009, 32(7): 1247-1263)
- [27] Holub V, Fridrich J. Low-complexity features for JPEG

steganalysis using undecimated DCT. IEEE Transactions on Information Forensics and Security, 2015, 10(2): 219-228

[28] Kodovsky J, Fridrich J, Holub V. Ensemble classifiers for steganalysis of digital media. IEEE Transactions on Information Forensics and Security, 2012, 7(2): 432-444

[29] Boroumand M, Chen M, Fridrich J. Deep residual network for steganalysis of digital images. IEEE Transactions on Information Forensics and Security, 2019, 14(5): 1181-1193

[30] Yang Z, Wang K, Ma S, et al. IStego100K: Large-scale image steganalysis dataset//Proceedings of the 18th International Workshop on Digital Forensics and Watermarking. Chengdu, China, 2020, 12022: 352-364

附录 1.

例 2. 超像素去噪过程举例.

为了便于理解,以 $\varphi=9$ 为例描述超像素去噪的具体过程.假设一个图像 C 中的局部像素块为

131 134 133 128 128 131 131

125 126 126 125 125 129 128

120 122 125 126 125 124 124

96 97 102 105 105 107 107

80 81 83 83 84 85 85

81 82 83 84 84 85 86

(19)

该像素块对应的超像素图 P 中的超像素索引块为

59 59 59 59 59 59 59

60 60 60 60 60 59 59

60 60 60 60 60 60 60

61 61 61 61 61 61 61

62 62 62 62 62 62 62

62 62 62 62 62 62 62

(20)

其中的数字是超像素索引,相同的索引表示它们属于同一个超像素.超像素去噪过程如下:

- (1)在超像素索引块中,索引为 61 的超像素尺寸为 $7 < \varphi$,故该超像素为噪音超像素,将其合并到其他超像素.
- (2)分别计算每个噪音超像素中像素的合并位置.以第

4 行第 3 列的像素 102 为例, $c_{i,j}=102, p_{i,j}=61$. 取以 $c_{i,j}$ 为中心的九宫格

$J_c=[c'_{u,v}]_{3\times3}=\begin{bmatrix}122 & 125 & 126 \\ 97 & 102 & 105 \\ 81 & 83 & 83\end{bmatrix}$

(21)

对应的像素索引为

$J_p=[p'_{u,v}]_{3\times3}=\begin{bmatrix}60 & 60 & 60 \\ 61 & 61 & 61 \\ 62 & 62 & 62\end{bmatrix}$

(22)

J_c 中满足超像素索引不等于 61,且与 102 最接近的值为 83,位于第 3 行第 2 列.

(3)将第 3 行第 2 列对应的超像素索引 62 替换 $p_{i,j}$ 原来的超像素索引 61.

接下来,继续合并噪音超像素中的其他像素,该超像素最终被合并到其他超像素,超像素索引 P 被调整为

59 59 59 59 59 59 59

60 60 60 60 60 59 59

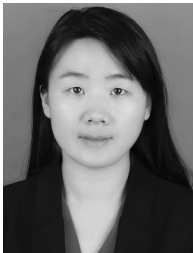
60 60 60 60 60 60 60

62 62 62 60 60 60 60

62 62 62 62 62 62 62

62 62 62 62 62 62 62

(23)



ZHU Li-Yan, Ph.D. candidate. Her research interests focus on image steganography.

LUO Xiang-Yang, Ph.D. , professor, Ph.D. supervisor. His research interests focus on network and multimedia security.

ZHANG Yi, Ph.D. , lecturer. Her research interests focus on image steganography and steganalysis technique.

WANG Jin-Wei, Ph.D. , professor, Ph.D. supervisor. His research interests focus on information security and multimedia forensics.

LU Wei, Ph.D. , professor, Ph.D. supervisor. His research interests focus on multimedia security and digital forensics.

LIU Fen-Lin, Ph.D. , professor, Ph.D. supervisor. His research interests focus on network information security.

Background

Asymmetric distortion steganography is an important research direction in information security(digital steganography), which helps secret messages to hide in images more safely.

Steganography technology not only hides the existence of secret messages but also hides the behavior of information transmission. Thus provides a good way for data security

transmission, which is of great significance to military and national defense.

Image steganography has been focused on symmetric adaptive steganography for a long time. The focus of symmetric adaptive steganographies is by designing reasonable symmetric distortion functions to keep messages embedding adaptively in the regions that are not easy to detect, so as to ensure the security of the generated stego images. However, in recent years, through theoretical analysis and experimental verification, researchers found that for JPEG images, the asymmetric distortion is more reasonable than the symmetric distortion. Stego images generated by embedding messages utilizing asymmetric distortion steganography are more secure than those by symmetric distortion steganography. Asymmetric distortion steganography has become an important path to improve the security of stego images.

Recently, researchers have proposed some asymmetric distortion steganography methods. The main idea is to utilize high-quality side-information images. However, there are still some problems existing in obtaining high-quality side-information images. For example, obtaining an additional precover image is very difficult; the image restoration algorithm to generate a side-information image takes a long time; the quality of a side-information image generated by filtering still

needs to be improved. To this end, this manuscript proposes an asymmetric distortion steganography method based on superpixel filtering, which would obtain a high-quality side-information image in a short time and would take a short time and enhance the anti-statistical detection ability.

This research group mainly studies image steganography and steganalysis, and publishing a large number of papers in related fields, such as: “Inverse Interpolation and Its Application in Robust Image Steganography” (see IEEE TCSVT), “Invariances of JPEG-quantized DCT Coefficients and Their Application to Robust Image Steganography” (see Signal Processing), “Image Robust Adaptive Steganography Adapted to Lossy Channels in Open Social Networks” (see Information Sciences), “Selection of Rich Model Steganalysis Features Based on Decision Rough Set α -Positive Region Reduction” (see TCSVT, high cited paper), “On the Fault-Tolerant Performance for a Class of Robust Image Steganography” (see Signal Processing, high cited paper), etc.

This work was supported by the National Natural Science Foundation of China (Nos. 62172435, 62202495, 62261160653, and 62102101), the Zhongyuan Science and Technology Innovation Leading Talent Project of China (No. 214200510019) and the Alibaba Group through Alibaba Innovative Research (AIR) Program.