doi:10.3772/j.issn.1002-0470.2020.12.005

# 视频可见水印检测与去除关键技术研究①

董 慧② 冯 杰③ 马汉杰 杨小利 王 健

(浙江理工大学信息学院 杭州 310018)

**摘 要** 针对可见水印多样性的特点和视频可见水印检测与去除过程中需要大量人工干预的问题,提出了一种基于图像修复的视频可见水印去除方法。该方法主要分为水印检测和水印去除 2 部分,水印检测采用目标检测和图像分割结合的方法,使检测的水印图案更准确;水印去除采用基于深度图像先验的图像修复实现,并在损失函数中引入结构相似度,修复网络通过学习水印以外的区域获得图像先验修复图像。为了验证该方法的有效性,用公开网站上的真实视频进行了综合的实验评估,实验结果表明该方法与其他修复方法相比,修复后的视频帧结构相似度和峰值信噪比更高。

**关键词** 视频可见水印;图像分割;图像先验;图像修复;目标检测

## 0 引 言

近年来,计算机视觉技术不断发展,很多学者研究视频标注[1-2]算法用于训练模型、研究水印嵌入算法用于所有权声明。但在一些特殊情况下,需要一定的技术将水印去除。比如水印版权已过期,而水印的设计公司不再提供技术支持[3]。在去除视频水印之前,需要检测出视频水印的图案和位置信息。在现有的视频内容检测、识别[4-5]和修复算法中,用于视频水印检测和去除的方法可以分为基于单图像和基于图像集两类。基于单图像的方法有:Pinjarkar 和 Tuptewar[6]把基于 Graph 区域分割和样本块填充修复图像的方法应用到视频修复中,单帧修复较好但分割不够精确,需要一定的人工交互。Cheng 等人[7]针对图片水印提出了一种基于 RetinaNet 和 Unet 架构的水印检测和去除方法,此方法对有一定透明度且结构单一的水印修复效果较好。基于图像集的方法有,Xu 等人[8]提出了累加灰度图和基于样本的源区域块稀疏线性组合填充的方法检测和去除图像水印。这种已知区域填充的方法对于小区域修复效果较好,区域较大时往往过于平滑。Dekel 等人[9]提出了一种广义的多图像 Matting 算法,该算法通过计算图像集的梯度自动估计“前景”(水印)、$\alpha$ 值和“背景”。该算法对透明水印的去除有一定效果,对彩色水印修复效果较差。Dashti 等人[10]连续使用下一帧的最佳匹配块填充当前帧的水印部分进行水印去除,帧间过渡较自然但单帧修复痕迹较明显。以上方法在某些情况可以去除水印,但由于水印技术的发展,水印位置和形态都有可能改变,甚至一个视频帧有多个水印,传统的水印检测和去除方法已经不再适用。因此,针对上述问题,本文提出了一种基于目标检测、图像分割和图像修复融合的水印检测和去除技术。

本文的主要贡献如下。

(1) 提出了一种基于目标检测与图像分割算法的可见水印检测方法。能够有效节省人力、改善动态水印提取困难的问题。

(2) 提出了一种基于深度图像先验的可见水印去除方法。在损失函数中引入结构相似度,能够有效去除水印、改善图像先验网络生成图像结构不够

① 国家自然科学基金青年基金(61501402)资助项目。

② 女,1994 年生,硕士生;研究方向:计算机视觉,机器学习;E-mail: 461373069@qq.com

③ 通信作者,E-mail: arlose@zstu.edu.cn

(收稿日期:2019-12-20)

清晰的问题。

（3）通过实验分析对比不同方法对图片可见水印去除的效果，实验结果表明所提方法优于其他方法。

## 1 本文方法概述

本文提出的视频可见水印检测和去除技术主要分为水印检测和水印去除。在水印检测前首先根据视频长短选择间隔帧数保存视频帧。水印检测部分首先使用目标检测算法进行水印区域检测。根据检测得到的水印区域坐标，生成二进制水印框图，与视频帧一起作为图像分割的输入。图像分割算法对水印框内的图像进行前景分割，提取出准确的水印图案。水印去除部分首先对水印图案进行掩膜预处理，根据水印在原图的坐标，生成与原图大小一样的二进制掩膜。把视频帧和掩膜输入图像修复网络进行水印去除，深度网络从视频帧水印外的区域中学习图片特征以生成没有水印的新图片。最后，把修复后的视频帧转为视频。本文方法整体流程图如图 1所示。

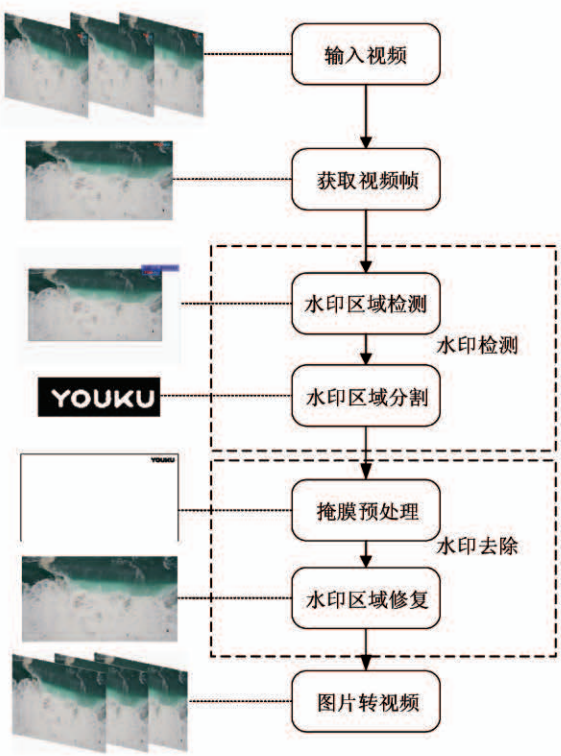


图 1 本文算法流程

## 2 水印检测

水印检测是为了准确地自动获取动态或者静态水印的位置信息和图案信息，节省大量的人力资源。常见的方法有基于视频水印一致性的方法，如计算中值梯度和累加灰度图[8]等。基于水印结构的方法，如根据结构分解图像的方法[11]等。

在现有的网站中，许多视频的可见水印是动态的。动态水印主要分为两种情况：水印位置不变、形态变化的情况，如西瓜视频和人人视频；水印形态不变、位置变化的情况，如韩剧 TV。如图 2 所示，图 2 从上到下依次为西瓜视频、人人视频和韩剧 TV 同一个视频不同时刻的视频帧。针对上述 2 种情况，利用中值梯度和累加灰度图的方法对视频进行水印提取会出现水印不完整和提取不到的问题。且对于十几秒的短视频，视频内容变化较小，往往有把视频帧内容当作水印提取出来的问题。基于结构的方法往往出现把字幕错认为水印的现象。本文使用目标检测算法先检测出水印区域减小水印切割范围，再使用图像分割算法切割出准确的水印图案。

图 2 视频动态水印示例图

### 2.1 水印区域检测

针对现有水印检测算法在水印动态变化和视频内容变化较小的情况下检测不到水印的问题，本文把每类水印当作一种目标，使用目标检测算法进行

检测。目前,常见的目标检测算法有 SSD(single shot multibox detector)[12],Faster R-CNN(faster regions with convolutional neural networks features)[13],YOLO(you only look once)[14]算法等。Faster R-CNN 检测效果较好,但存在着检测速度慢的缺点。传统的 YOLO 算法检测速度较快,但小目标检测效果不佳、定位不准。SSD 在一定程度上取得了速度与准确率之间的平衡,而 YOLOv3 与 SSD 的准确率相当,但是速度快 3 倍,因此本文使用 YOLOv3 目标检测算法检测水印区域。

在水印区域检测部分,本文将数据集中 5164 张视频帧作为训练集,其余的 1290 张视频帧作为测试集。YOLOv3 在测试集中检测的准确率为 98%。检测结果如图 3 所示。

**图 3　水印区域检测结果示意图**

## 2.2　水印区域分割

水印区域检测到的是个矩形区域,但整个矩形区域除了实际的水印图案,还包含一些视频内容,这种方法修复得到的视频帧语义效果较差,边缘处修复痕迹较明显。因此本文在检测出水印区域之后再进行水印区域分割,以提取出准确的水印图案。图像分割主要有基于像素的方法、基于边界的方法和基于区域的方法。前 2 种方法需要大量的人工交互,本文选择简单快速的基于区域的方法。基于区域的方法常用的有 GraphCut 和 GrabCut[15],Cheng 等人[16]在 GrabCut 的基础上提出的 DenseCut 算法,用紧密连接的条件随机场(conditional random field,CRF)代替传统 GrabCut 公式中耗时的全局颜色模型迭代细化,能获得较好的分割效果。水印通常由图案和文字构成,文字往往较细小且与视频内容差距较大,对分割的精确度要求比较高,因此本文采用 DenseCut 进行水印区域分割。

CRF 定义在随机变量 $X=\{X_1, X_2, \cdots, X_n\}$ 上,每个随机变量对应于一个像素,其中 $X_i$ 表示像素 $i$ 的二进制标记,值为 0 或 1。值为 0 表示像素 $i$ 为背景,否则表示为前景,$i \in N=\{1,2,\cdots, n\}$。$\boldsymbol{x}$ 表示这些随机变量的联合结构,$\boldsymbol{I}$ 表示视频帧的图像数据。全连接的二进制 CRF 定义如式(1)所示。

$$E(\boldsymbol{x}) = \sum_{i \in N} \boldsymbol{\psi}_i(x_i) + \sum_{i<j} \boldsymbol{\psi}_{ij}(x_i, x_j) \tag{1}$$

其中,$i$ 和 $j$ 表示图像中像素的位置。一元项 $\boldsymbol{\psi}_i(x_i)$ 代表像素 $i$ 标记为 $x_i$ 的损失,定义如式(2)所示。可以通过对标签 $x_i$ 产生分布的分类器对每个像素单独计算。

$$\boldsymbol{\psi}_i(x_i) = -\log P(x_i) \tag{2}$$

式(2)中的前景项或背景项 $P(x_i)$ 为

$$P(x_i) = \frac{P(\boldsymbol{\Theta}_{x_i}, \boldsymbol{I}_i)}{P(\boldsymbol{\Theta}_{x_0}, \boldsymbol{I}_i) + P(\boldsymbol{\Theta}_{x_1}, \boldsymbol{I}_i)} \tag{3}$$

式(3)中 $P(\boldsymbol{\Theta}_0, \boldsymbol{I}_i)$ 和 $P(\boldsymbol{\Theta}_1, \boldsymbol{I}_i)$ 分别代表像素颜色 $\boldsymbol{I}_i$ 属于背景模型 $\boldsymbol{\Theta}_0$ 和前景模型 $\boldsymbol{\Theta}_1$ 的概率密度。根据水印框图使用混合高斯模型(Gaussian mixture model,GMM)来估计概率密度值 $P(x_i)$。矩

形框对应位置以外的部分作为背景，框内图像为可能的前景。考虑到自然图像中的颜色通常只覆盖整个颜色空间的一小部分，本文选择最频繁的颜色，确保这些颜色覆盖超过 95% 的图像像素的颜色，其余像素的颜色（占图像像素 5% 以下）被直方图中最接近的颜色替换。本文选择 5 个高斯前景模型和 5 个高斯背景模型。

全连接二元项 $\psi_{ij}(x_i, x_j)$ 鼓励相似像素和相邻像素分为同样的标签。使用对比比较明显的三核势能，如式(4)、(5)所示。

$$\psi_{ij} = g(i,j)[x_i \neq x_j] \tag{4}$$

$$g(i,j) = w_1 g_1(i,j) + w_2 g_2(i,j) + w_3 g_3(i,j) \tag{5}$$

其中，[ · ]为 Iverson 括号，如果条件为真值，其值为 1，否则为 0；$w_1$、$w_2$、$w_3$ 为比例系数；相似度函数 $g(i,j)$ 根据颜色向量 $\boldsymbol{I}_i$、$\boldsymbol{I}_j$ 和位置 $\boldsymbol{p}_i$、$\boldsymbol{p}_j$ 来定义，具体如式(6)～(8)所示。

$$g_1(i,j) = \exp\left(-\frac{|\boldsymbol{p}_i - \boldsymbol{p}_j|^2}{\theta_\alpha^2} - \frac{|\boldsymbol{I}_i - \boldsymbol{I}_j|^2}{\theta_\beta^2}\right) \tag{6}$$

$$g_2(i,j) = \exp\left(-\frac{|\boldsymbol{p}_i - \boldsymbol{p}_j|^2}{\theta_\gamma^2}\right) \tag{7}$$

$$g_3(i,j) = \exp\left(-\frac{|\boldsymbol{I}_i - \boldsymbol{I}_j|^2}{\theta_\mu^2}\right) \tag{8}$$

式(6)代表外观相似度，并鼓励具有相似颜色的附近像素具有相同的二进制标签。式(7)代表局部平滑度，辅助去除小的孤立区域。邻近度、相似度和平滑度由 $\theta_\alpha$、$\theta_\beta$、$\theta_\gamma$ 和 $\theta_\mu$ 控制，具体值与文献[12]相同，分别为 $w_1 = 6$、$w_2 = 10$、$w_3 = 2$、$\theta_\alpha = 20$、$\theta_\beta = 33$、$\theta_\gamma = 3$、$\theta_\mu = 43$。

数据集中水印提取结果如图 4 所示。其中，图 4(a)为视频帧原图，图 4(b)为根据坐标位置生成的水印位置框图，图 4(c)为 DenseCut 输出的水印图案。

(a) 原图　(b) 水印位置框图　(c) 水印图案

图 4　水印提取示例图

## 3　水印去除

### 3.1　掩膜预处理

在图像修复之前，需要对检测到的水印图案做一些预处理以获得修复网络需要的掩膜。因为掩膜大小应该与原视频帧大小一致，所以需要根据水印在原图的坐标，把水印图案对应位置的像素值赋给与原图大小一样的二进制掩膜，二进制掩膜水印坐标以外的区域全为 0。再对二进制掩膜进行膨胀反转处理，反转后的图像为掩膜 $\boldsymbol{m}$，如图 5 所示。其中图 5(a)为水印检测得到的水印图案，图 5(b)为由图 5(a)经过赋值操作得到的掩膜，图 5(c)为图 5(b)膨胀反转后的掩膜，即网络里用到的 $\boldsymbol{m}$。

(a) 水印图案　(b) 掩膜　(c) 膨胀反转后的掩膜

图 5　掩膜示意图

### 3.2　水印区域修复

水印去除最常用的方法就是图像修复。图像修复算法主要包括两个方向：基于结构和纹理的图像修复和基于深度学习的图像修复。基于结构和纹理的图像修复算法可以修复细小区域的缺失，随着缺失区域增大，修复效果逐渐恶化，修复后的图片存在语义信息不完整、图像模糊等问题。而深度学习相对于传统的基于结构和纹理的修复算法能够捕捉更多图像的高级特征[17]。

因此本文水印去除部分基于 Ulyanov 等人[18]提出的图像修复算法，该方法利用深度卷积网络获得

图像先验修复水印区域,不需要训练大量的数据集来获得图像的特征,而是通过深度学习捕获当前输入的图片信息,能够得到更合理的图像结构。本文在文献[18]的基础上改进了损失函数,引入了结构相似度。

图像生成器通过学习生成器(编码器)网络把随机编码向量 $z$ 映射为图像 $\boldsymbol{x}$,如式(9)所示。其中,$\theta$ 为参数,其初始化值使用标准正态分布,$\boldsymbol{x} \in R^3 \times H \times W$ 为网络的输出,$z$ 与 $\boldsymbol{x}$ 空间大小一样,为随机初始化的张量。

$$\boldsymbol{x} = f_\theta(z) \tag{9}$$

网络修复任务的目标函数表示为能量最小化问题,如式(10)所示。

$$\boldsymbol{x}^* = \min_{\boldsymbol{x}} E(f_\theta(z); \boldsymbol{x}_0) + R(\boldsymbol{x}) \tag{10}$$

其中,$\boldsymbol{x}^*$ 为解空间里的最优解,$\boldsymbol{x}_0$ 为要修复的图片即原视频帧,$E(f_\theta(z); \boldsymbol{x}_0)$ 为数据项,具体如式(11)所示。$R(\boldsymbol{x})$ 为神经网络捕获到的隐式先验,是一种指示函数。

$$E(\boldsymbol{x}; \boldsymbol{x}_0) = \| (f_\theta(z) - \boldsymbol{x}_0) \odot \boldsymbol{m} \|^2 \tag{11}$$

$$\begin{cases} \theta^* = \arg\min_\theta E(f_\theta(z); \boldsymbol{x}_0) \\ \boldsymbol{x}^* = f_{\theta^*}(z) \end{cases} \tag{12}$$

式(11)中 $\boldsymbol{m}$ 为预处理得到的二进制掩膜,$\odot$ 为 Hadamard 乘积,表示对应像素相乘,最小化的 $\theta^*$ 使用优化器从初始化参数训练得到。

当图片先验信息是由 $z$ 通过深度卷积网络的某种架构获得时,$R(\boldsymbol{x}) = 0$,其他情况 $R(\boldsymbol{x}) = \infty$。该算法在输入网络之前并没有任何图片先验知识,而是通过网络学习获得,因此本文中 $R(\boldsymbol{x}) = 0$,根据式(9),可以将式(10)表示为式(12)。

该算法计算有水印的视频帧与修复后的视频帧之间的损失。本文在文献[15]损失函数的基础上添加了 $L2$ 损失项,如式(13)~(16)所示。式(14)代表网络找到一个最优的 $\theta$ 使得网络输出的图片 $f_\theta(z)$ 与原视频帧 $\boldsymbol{x}_0$ 在掩膜 $\boldsymbol{m}$ 以外的地方像素值差距最小,具体如式(14)所示。式(15)是计算网络输出图片与原视频帧在掩膜以外区域的结构相似度(structural similarity index,SSIM),因为 SSIM 越高代表生成图片与原图片结构越接近,所以使用式(15)的形式。式(16)中的 $\mu_x$ 和 $\mu_y$ 分别代表 $\boldsymbol{x}$, $\boldsymbol{y}$ 的平均值,$\sigma_x$ 和 $\sigma_y$ 分别代表 $\boldsymbol{x}$, $\boldsymbol{y}$ 的标准差,$\sigma_{xy}$ 代表 $\boldsymbol{x}$ 和 $\boldsymbol{y}$ 的协方差。$c_1$、$c_2$ 为经验值,具体设置为 $c_1 = 0.0001$, $c_2 = 0.0009$。

$$Loss = L1 + L2 \tag{13}$$

$$L1 = \| \boldsymbol{x} - \boldsymbol{y} \|_2 \tag{14}$$

$$L2 = 1 - SSIM(\boldsymbol{x}, \boldsymbol{y}) \tag{15}$$

$$\begin{cases} SSIM(\boldsymbol{x}, \boldsymbol{y}) = \dfrac{(2\mu_x \mu_y + c_1)(\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \\ \boldsymbol{x} = f_\theta(z) \odot \boldsymbol{m} \\ \boldsymbol{y} = \boldsymbol{x}_0 \odot \boldsymbol{m} \end{cases} \tag{16}$$

由于视频中同一个镜头的视频帧内容变化较小,因此可以使用同一镜头内前一帧修复时保存的网络参数作为下一帧图像修复的初始参数,来减少迭代次数。

网络结构由卷积层(Conv)、批量归一化(batch normalization,BN)、带泄露修正线性单元(rectified linear unit,LeakyReLU)、上采样(Upsample)和下采样(Downsample)组成,如图 6 所示。其中,$d_i$、$u_i$ 分别为深度为 $i$ 时的下采样和上采样操作,具体如图 6 右半部分所示。

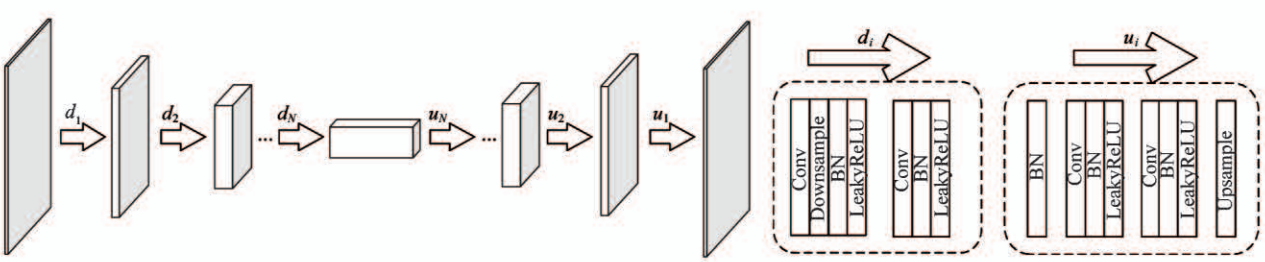


**图 6　修复网络结构图**

本文采用 pytorch 作为主要框架,网络超参数设置如表 1 所示。其中,$n$ 代表上采样操作和下采样

操作在深度 $i$ 处的滤波器数量，$k$ 代表上采样操作和下采样操作在深度 $i$ 处的核的大小。num iter 为迭代次数、LR 为学习率（learning rate）。其他参数设置具体为 upsampling = nearest，padding = reflection，optimizer = Adam。

**表 1　网络超参数设置**

| 超参数 | $n$ | $k$ | num iter/次 | LR |
|---|---|---|---|---|
| 值 | 128 | 3 | 20 000 | 0.03 |

## 4　实验分析

为了评估算法的性能，本文从 15 个常见视频网站上收集了 450 个视频进行实验验证。其中包括时长为十几秒左右的短视频（如抖音短视频），和其他时长为几十分钟以内的视频。这 450 个视频分辨率各不相同，同一个网站的水印也会有所不同。本文在视频帧中选择场景不同或水印形态、位置不同的 6 454 帧作为本文的数据集，并且把属于同一个网站的水印分为同一类，共 15 类。图 7 为数据集中 15 类水印的视频帧示例图。

本节分别从主观评价和客观评价两个方面对本文方法与另外 4 种算法进行修复对比。这 4 种算法为深度图像先验算法[18]、经典的 Criminisi 修复算法[19]、全局局部一致的修复算法[20]以及 EdgeConnect 算法[21]。其中客观评价指标为峰值信噪比（peak signal to noise ratio，PSNR）和结构相似度。PSNR 基于原视频帧与修复后视频帧的对应位置的像素差，值越大代表图像质量越好。SSIM 是一种衡量两幅图像相似度的指标，分别从亮度、对比度、结构 3 方面度量图像相似性。其值可以较好地反映人眼主观感受，一般取值范围是 0 ~ 1。


**图 7　数据集示例图**

为了使修复效果对比更准确，本文对所有算法使用同样的水印检测方法，即本文提出的 YOLOv3 和 DenseCut 结合的水印检测方法。检测到水印以后，把比水印略大的二进制图片转化为与原视频帧同样大小的二进制掩膜。对同样的视频帧和二进制掩膜分别使用文献[18-21]和本文的方法进行修复。

### 4.1　主观评价

本文在数据集中挑选几种比较典型的视频帧作

为视觉对比的示例,如图 8 所示。西瓜视频为多水印且水印背景与水印差距较大的示例,爱奇艺为水印区域细小结构单一、背景较复杂的示例。酷 6 为水印结构与背景都较复杂的示例。其中,每一张图片中实线框代表水印区域,虚线框为对应区域的局部放大效果图。

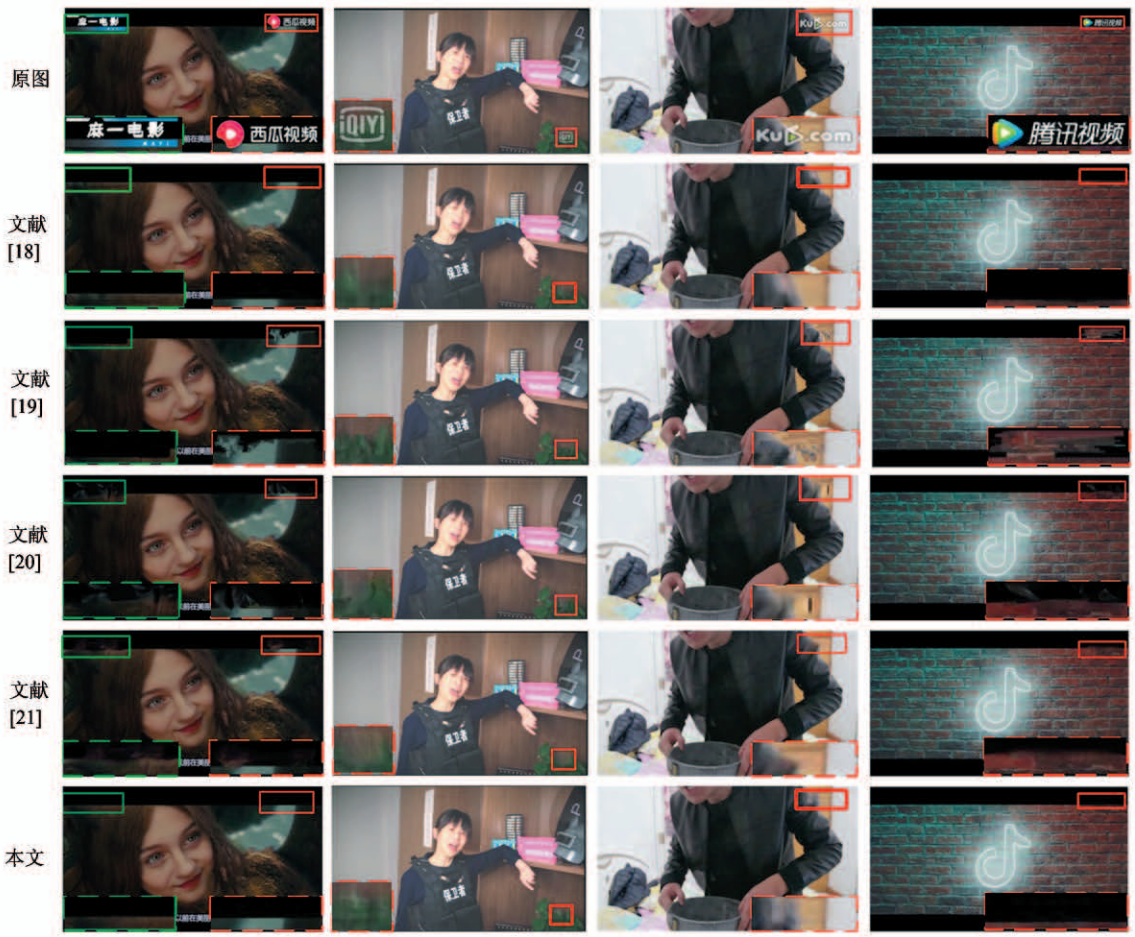


**图 8　修复效果主观对比图**

从图 8 可以看出,文献[19]修复后的水印区域结构和纹理较清晰,但有把补丁填充到错误区域的现象且有明显的块效应。如西瓜视频中左边矩形框内的水印修复把黑色区域填充到了本该是头发纹理的区域,右边矩形框内的水印修复把水和翅膀的结构填充到了本该是黑色像素的区域。而对于爱奇艺水印这种结构纹理单一、较细小的水印修复后的结构和纹理较合理。文献[20]修复后的结果较为细腻,但依然有填充错误且边缘效果不理想的问题。如酷 6 视频帧修复后的衣服褶皱边缘有不自然的突起,门上有不合理的黑色条形结构,把错误的像素扩散到了水印区域。对于西瓜视频和腾讯视频,文献[21]修复结果边缘不够合理。对于爱奇艺视频,文献[21]的修复结果丢失了纹理和结构信息,较为模糊。对于西瓜视频、腾讯视频和酷 6 视频,文献[18]与本文方法修复后的水印部分边缘过渡较自然。对于边缘处理,本文修复效果比文献[18]视觉效果更好,人眼几乎看不出修复痕迹。而对于爱奇艺视频,本文修复结果与文献[19,20]相比结构和纹理不够清晰。

### 4.2　客观评价

由于资源的限制,本文只在公开视频网站上获得了 2 类未添加水印的视频,即爱奇艺和优酷,如图 9所示。优酷水印大小为 140 × 50 像素,视频帧大小为 1 280 × 720 像素。爱奇艺水印大小为 140 × 51 像素,视频帧大小为 896 × 504 像素。


图 9 真实水印图

本文获取这两类水印添加到视频中作为待修复视频，未添加水印的视频作为真实视频，客观评价使用这两类视频帧。本文对每一类视频选择50个视频镜头，每个镜头选取 10 帧，总共 100 个镜头 1000 张视频帧。计算每张修复后的视频帧与未添加水印的视频帧之间的 SSIM 和 PSNR，对同类的 500 张视频帧取平均值，结果如表 2 ~ 3 所示。

由于文献[19-21]仅仅改变了掩膜位置区域，而深度图像先验是重新生成整张图片，因此本文在使用改进损失函数的基础上还保留原视频帧掩膜以外的像素值。

表 2 SSIM 对比表

| 算法 | 文献[18] | 文献[19] | 文献[20] | 文献[21] | 本文 |
|---|---|---|---|---|---|
| 爱奇艺 | 0.9362 | 0.9660 | 0.9617 | 0.9668 | 0.9837 |
| 优酷 | 0.9268 | 0.9667 | 0.9633 | 0.9661 | 0.9690 |

表 3 PSNR 对比表

| 算法 | 文献[18] | 文献[19] | 文献[20] | 文献[21] | 本文 |
|---|---|---|---|---|---|
| 爱奇艺 | 34.1108 | 32.5230 | 34.7003 | 36.5060 | 39.0502 |
| 优酷 | 31.7192 | 33.6557 | 35.7302 | 35.4461 | 35.7873 |

对这两种水印进行修复后的 SSIM 值如表 2 所示，从中可以看出，利用本文方法所得结果比另外 4 种方法所得结果高，这在一定程度上表示本文修复的视频帧在结构纹理各方面与原视频帧更接近。从表 3 可以看出，对于爱奇艺视频和优酷视频，本文修复结果的 PSNR 均高于另外 4 种方法。对于优酷视频，与文献[20]相比没有明显的优势，这是因为优酷水印与视频帧的占比较小，且本文使用的修复方法是通过网络学习重新生成的图片，清晰度有一定程度的下降。

综上所述，从主观和客观两方面来看，本文方法在水印区域较大或水印结构相对复杂的情况下，修复效果明显优于另外 4 种算法。在水印较为细小时，本文修复结果没有明显优势。这是由于本文方法使用隐式先验的方法训练卷积网络生成图像，并且在损失函数中引入结构相似度，使得卷积网络在图像空间中选择自然且与输入图像结构更接近的图片作为输出，因此本文修复结果优于另外 4 种方法。总体来说，本文修复结果视觉上更符合现实场景，且 SSIM 值和 PSNR 值都有一定程度的提高。

## 5 结 论

本文提出了一种目标检测、图像分割融合的水印检测与深度网络图像修复的水印去除结合的视频可见水印检测和去除方法。无需人力干预就能够有效检测和去除视频中可见动态水印和较复杂水印，且不限制视频的分辨率大小，是一种通用的视频可见水印自动检测和去除方法。实验对比表明，本文方法能有效去除可见水印，边缘修复效果较好，且对于大区域水印经本文修复方法修复后的视频帧更接近自然图像。但修复视频所需时间较长，因此，提升视频修复速度是下一步的研究方向。

**参考文献**

[1] Zhang X Y, Wang S P, Yun X C. Bidirectional active learning: a two-way exploration into unlabeled and labeled data set[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2015, 26(12): 3034-3044

[2] Zhang X Y, Shi H, Zhu X, et al. Active semi-supervised learning based on self-expressive correlation with genera-

tive adversarial networks[J]. *Neurocomputing*, 2019, 345: 103-113

[3] 李瑞民. 数字水印擦除技术的研究[J]. 电视技术, 2015, 39(1): 12-14

[4] Zhang X Y, Li C, Shi H, et al. AdapNet: adaptability decomposing encoder-decoder network for weakly supervised action recognition and localization[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2020 (99): 1-12

[5] Zhang X Y, Shi H, Li C, et al. Learning transferable self-attentive representations for action recognition in untrimmed videos with weak supervision[C]// Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, USA, 2019: 1-8

[6] Pinjarkar A, Tuptewar D J. Robust exemplar based image and video inpainting for object removal and region filling [C]// Proceedings of 2017 International Conference on Intelligent Computing and Control, Coimbatore, India, 2017: 1-4

[7] Cheng D, Li X, Li W H, et al. Large-scale visible watermark detection and removal with deep convolutional networks[C]// Proceedings of the 1st Chinese Conference on Pattern Recognition and Computer Vision, Guangzhou, China, 2018: 27-40

[8] Xu C R, Lu Y, Zhou Y. An automatic visible watermark removal technique using image inpainting algorithms[C] // Proceedings of the 4th International Conference on Systems and Informatics, Hangzhou, China, 2017: 1152-1157

[9] Dekel T, Rubinstein M, Liu C, et al. On the effectiveness of visible watermarks[C]// Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, USA, 2017: 6864-6872

[10] Dashti M, Safabakhsh R, Pourfard M, et al. Video logo removal using iterative subsequent matching[C]// Proceedings of the International Symposium on Artificial Intelligence and Signal Processing, Mashhad, Iran, 2015: 84-88

[11] Santoyo-Garcia H, Fragoso-Navarro E, Reyes-Reyes R, et al. An automatic visible watermark detection method using total variation[C]// Proceedings of the 5th International Workshop on Biometrics and Forensics, Coventry, UK, 2017:1-5

[12] Liu W, Anguelov D, Erhan D, et al. SSD: Single shot multibox detector[C]// Proceedings of the 14th European Conference on Computer Vision, Amsterdam, Netherlands, 2016: 21-37

[13] Ren S, He K, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017, 39(6): 1137-1149

[14] Redmon J, Divvala S, Girshick R, et al. You only look once: unified, real-time object detection[C]// 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, USA, 2016: 779-788

[15] Rother C, Kolmogorov V, Blake A. Grab cut-interactive foreground extraction using iterated graph cuts[J]. *ACM Transactions on Graphics*, 2004, 23(3): 309-314

[16] Cheng M M, Prisacariu V A, Zheng S, et al. DenseCut: densely connected CRFs for realtime GrabCut[J]. *Computer Graphics Forum*, 2015, 34 (7): 193-201

[17] 赵立怡. 基于生成式对抗网络的图像修复算法研究[D]. 西安: 西安理工大学印刷包装与数字媒体学院, 2018: 4-5

[18] Ulyanov D, Vedaldi A, Lempitsky V. Deep image prior [C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, USA, 2018: 9446-9454

[19] Criminisi A, Pérez P, Toyama K. Object removal by exemplar-based inpainting[C]// Proceedings of 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Madison, USA, 2003: 721-728

[20] Iizuka S, Simo-Serra E, Ishikawa H. Globally and locally consistent image completion[J]. *ACM Transactions on Graphics*, 2017, 36(4):1-14

[21] Nazeri K, Ng E, Joseph T, et al. Edgeconnect: generative image inpainting with adversarial edge learning[J]. *arXiv*:1901.00212, 2019

# Research on key technologies of video visible watermark detection and removal

Dong Hui, Feng Jie, Ma Hanjie, Yang Xiaoli, Wang Jian

(School of Informatics Science and Technology, Zhejiang Sci-Tech University, Hangzhou 310018)

**Abstract**

Considering the diversity of visible watermark, a video visible watermark removal method based on image inpainting is proposed to solve the problem of large amount of manual intervention in the process of video visible watermark detection and removal. It is mainly divided into two parts: watermark detection and watermark removal. Target detection and image segmentation are combined to detect watermark patterns more accurately. Watermark removal is realized by image inpainting based on deep image prior, introducing structural similarity into loss function, the inpainting network acquires inpainting image by learning regions other than watermark. The experimental results show that compared with other image inpainting methods, the video frame repaired by the proposed method has higher structure similarity and peak signal to noise ratio.

**Key words**: video visible watermark, image segmentation, image prior, image inpainting, object detection