一种端到端的对抗生成式视频数字水印算法

崔凯元[1]，申 静[2]，李叶凡[1]，王 晗[1]，王忠芝[1]
（1. 北京林业大学信息学院，北京 100083；2. 北京市延庆区园林绿化局，北京 102100）

摘 要：提出了一种端到端的对抗生成式视频盲水印嵌入提取算法。该算法主要由编码器和解码器组成，编码器用于生成包含水印信息的视频，解码器用于提取视频中所包含的水印信息。不同于传统的基于频域或空域的视频水印方法，用端到端的训练方式的同时优化编码器和解码器网络。在编码器训练过程中模拟不同的信号和几何攻击类型，生成对抗样本，优化整个网络，以保证所生成水印样本的不可感知性和鲁棒性。实验结果表明：该算法对缩放、平移、裁剪等几何类攻击和压缩、噪声等信号类攻击都具有较强的鲁棒性；同时，该算法独立地训练每一个关键帧，因此还可以抵抗视频中的时间同步攻击。

关键词：视频数字水印；编码器；解码器；几何攻击；时间同步攻击
中图分类号：TP309 **文献标志码**：A
文章编号：2095－2783(2021)07－0687－08 **开放科学(资源服务)标识码(OSID)**：

An end-to-end generative adversarial video digital watermarking algorithm

CUI Kaiyuan[1], SHEN Jing[2], LI Yefan[1], WANG Han[1], WANG Zhongzhi[1]
(1. *School of Information, Beijing Forestry University, Beijing* 100083, *China*;
2. *Yanqing Gardening and Greening Bureau, Beijing* 102100, *China*)

Abstract: An end-to-end generative adversarial video blind watermarking embedding and extraction framework in this paper was introduced. Under this framework, an encoder and a decoder were applied to generate video with watermark information and extract the video's watermark information, respectively. Unlike traditional video watermarking methods based on a frequency domain or spatial domain, an end-to-end training method was used to optimize both encoder and decoder networks. In the training process, different signal and geometric attack types were simulated to generate counter samples to optimize the whole network to ensure the generated watermark samples' imperceptibility and robustness. The experimental results show that the method is robust to geometric attacks such as scaling, translation, and clipping, as well as signal attacks such as compression and noise. Simultaneously, since the key-frame is trained independently, the embedding watermarking can also resist the time synchronization attack in the video.

Keywords: video digital watermarking; encoder; decoder; geometric attack; synchronization attack

计算机技术和多媒体系统的飞速发展使以数字形式出现的视频内容迅速增长，数字视频成为了网络多媒体、监控系统、远程会议、视频网站等应用中不可或缺的组成部分。数字视频的传播，导致视频数据容易被非法复制、篡改，严重侵犯了视频原作者的版权。据悉，某电影上映之后在短短 6 个月的时间就被非法拷贝超 700 万份。如何减少视频的非法传播，以及如何对泄露视频进行及时追踪，已成为视频版权保护领域一个亟待解决的问题。

为了更好地追踪泄露视频，且不影响视频的视觉感知效果，有研究者[1]通过在视频中加入不可见的盲水印，对视频的版权提供保护。盲水印即数字水印，是将一些特定的信息嵌入到数字图像或数字视频中[2-4]，允许人们建立所有权、识别购买者或提供有关数字内容的一些附加信息，这些嵌入的信息可以从视频中提取或检测。目前，盲水印相关工作主要关注不可感知性和鲁棒性 2 个方面的问题。其中，不可感知性是指水印嵌入后视频的主观视觉质量不会受到影响，人眼不能区分原视频和含水印视频；鲁棒性是指视频在经受攻击后其中的水印仍然能保持其原有的功能。

传统的数字水印方法有以最低有效位算法[5]与统计[6]为代表的空间域算法及以离散余弦变换(discrete cosine transform, DCT)[7]为代表的频率域算法。不论是图像水印还是视频水印，这些算法大都可以抵抗噪声、变色和压缩等常见攻击，而对于几何形变的攻击抵抗能力较弱。这是由于即使是微小的变换，也会导致全部像素值的改变，最终无法有效提取出水印。因此，本文提出一种端到端的对抗生成式视频数字水印算法，为关键帧添加扰动生成对抗性样本，并模拟对抗性样本的各种变换，以抵抗几何攻击。

收稿日期：2020-08-30
基金项目：国家自然科学基金资助项目(61703046)；中央高校基本科研业务费专项资金资助项目(2015ZCQ-XX)
第一作者：崔凯元(1996—)，男，硕士研究生，主要研究方向为计算机视觉、图像处理
通信作者：王晗，副教授，主要研究方向为多媒体信息的内容分析与检索、模式识别与机器学习，wanghan@bjfu.edu.cn

区别于图像水印，视频水印还会受到时间同步攻击，例如帧插入、帧删除、帧平均和帧重组。现有的算法大部分基于固定的一组视频帧进行嵌入水印操作，因此提取水印时需要检测出准确的帧位置。然而在受到时间同步攻击后，很难从视频中准确定位出这些包含水印的和视频帧，传统的图像水印算法在迁移到视频上后遇到了极大的挑战。本文算法训练时在关键帧上模拟扰动攻击，由于关键帧（B帧、P帧、I帧）在内容上相似，因此I帧相邻的帧也可以用于提取水印，解决时间同步攻击问题。

本文提出的端到端对抗生成式视频数字水印算法由编码器和解码器相结合，用编码器来模拟视频关键帧受到的各种攻击扰动，并训练解码器将受到任何扰动的关键帧都解码为与水印相同的序列，实现抵抗几何攻击和时间同步攻击的目的。

1 相关工作

视频数字水印的提取分为盲检测[8]和明检测2种。在提取水印的过程中，不需要原始视频数据作参考比对的叫作盲检测，对应的水印算法为盲水印算法；需要原始载体作参考比对的叫作明检测，对应的水印算法为明水印算法。实际情况是，有时并不能获得原始视频，因此盲水印算法的应用更广，本文算法即为盲水印算法。

针对几何攻击，Tsai等[9]提出了基于特征点匹配的方法，利用图像的几何不变性特征来嵌入和提取水印，但特征点检测有一些局限性，由于几何变换可能改变特征提取的结果，产生假特征点，导致水印提取失败。此外，由于在大量的视频帧中很难保留相同的显著特征点，基于特征点的算法迄今主要用于图像水印。Lin等[10]使用穷举搜索、图像配准或模板插入来计算几何参数，然后利用这些参数对原始格式进行补偿，最后再从校正后的版本中提取水印，这种基于同步的方法是非盲水印算法，在实际的使用中并不方便。Dong等[11]利用具有缩放和旋转不变性性质的傅里叶-梅林变换校正图像，用以抵抗缩放、旋转和平移攻击，虽然基于傅里叶-梅林变换的方法在理论上是有效的，但该变换方法的计算量大，且对于裁剪攻击的鲁棒性较低。

传统的基于几何不变量的算法在解决几何攻击时，效果并不理想。Zhu等[12]、Tancik等[13]研究了一种基于深度神经网络的水印算法。Zhu等[12]提出通过神经网络将信息写入图像之中；而Tancik等[13]提出在训练解码器时，模拟真实包括仿射变换的图像质量退化模型，对嵌入信息的图像做变换，但该方法在水印位数大于100时对视觉感知造成一定影响。在训练编解码器过程中，如模拟攻击是固定的，则训练出来的编解码器可能会对其他类型的攻击具有较差的鲁棒性。据此Luo等[14]提出了一种专用于产生不同图像攻击的辅助网络，但该方法训练出的模型对裁剪攻击抵抗性较弱。Liu等[15]提出针对各种噪声攻击较为鲁棒的方法，对于拉伸变形攻击则效果一般。本文算法能抵抗一定的裁剪攻击，且在水印位数为256时，不可见性仍然很高，很好地平衡了鲁棒性和不可见性。

2 基于解码器的视频数字盲水印算法

算法主要包含编码器和解码器2部分。编码器根据Goodfellow等[16]提出的快速递度符号法（fast gradient sign method，FGSM）为视频关键帧X添加扰动p，生成对抗性样本X'，表示为

$$X' = X + \varepsilon \times p。 \tag{1}$$

式中，ε为扰动强度，取值范围为$[0,1]$，ε越大，扰动越明显，在降低模型精度方面越有效，但同时原始图像的改变也越容易被人眼察觉。

计算X'在变换函数$t(X')$的分布集合T，表示为

$$T = \{t_1(X'), t_2(X'), \cdots, t_{n-1}(X'), t_n(X')\}。 \tag{2}$$

式中，$t_1, t_2, \cdots, t_{n-1}, t_n$为添加噪声、裁剪、缩放等不同的变换函数。

式(2)可简写为

$$T = t(X, p)。 \tag{3}$$

式中，t为变换函数。

解码器D由1个深度神经网络构成，将对抗性样本解码为与水印W长度相同的序列W'，表示为

$$W' = D(T, \theta)。 \tag{4}$$

式中，θ为模型参数。

编码器与解码器采用对抗的方式进行优化。编码器通过加入扰动p，生成让解码器解码错误的对抗性样本；解码器将对抗性样本解码成功，即$W' = W$。

利用编码器生成包含噪声样本进行对抗训练时主要包括2个过程[17-18]：

1）生成使预测损失最大化的扰动p，有

$$\text{argmax BCELoss}(W, D(t(X, p), \theta))； \tag{5}$$

2）更新使预测损失最小化的模型参数θ，有

$$\text{argmin BCELoss}(W, D(t(X, p), \theta))。 \tag{6}$$

其中交叉熵损失函数BCELoss为

$$\text{BCELoss}(X_i, y_i) = -\omega_i[y_i \log x_i + (1 - y_i)\log(1 - x_i)]。 \tag{7}$$

通过不断迭代对抗训练的过程，学习可得可抵抗几何攻击的嵌入水印。端到端的对抗生成式视频数字水印算法流程如图1所示。

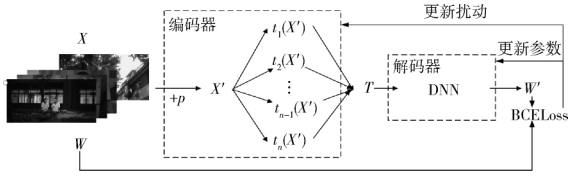


图1 端到端的对抗生成式视频数字水印算法流程

Fig. 1 Flow chart of end-to-end generative adversarial video digital watermarking

2.1　编码器

如何提高水印鲁棒性，抵抗各种攻击仍然是一个不容忽视的问题[19-20]。当一个通过神经网络正确分类的图像受到扰动时，神经网络会以较高的概率将其分类错误，这就是抗性示例[21-24]。Goodfellow 等[16]认为，由于深度神经网络维度高，且大部分结构是线性的，因此对图像进行细微的扰动后，会导致神经网络分类错误。根据该理论，视频关键帧经过噪声、压缩或几何变换等攻击后，解码器将无法解码出正常的、有意义的水印。因此，本文通过在视频关键帧中加入噪声来将对抗性样本注入训练集中。

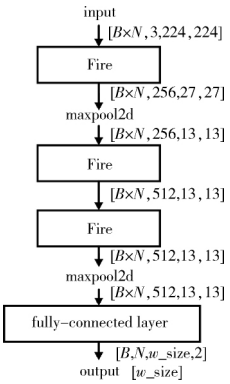
FGSM 用于简单计算生成对抗性样本，如式(1)所示。该方法可以总结为在原始图像上添加微小的扰动，扰动强度 ε 越大，扰动越明显，在降低模型精度方面越有效，但同时原始图像的改变越容易被人眼察觉。因此对抗样本需要在干扰强度和攻击的成功率中取一个平衡值。

虽然神经网络容易受对抗样本影响，但在现实世界中，根据 FGSM 生成的对抗性样本被不同视角捕捉或者受到光线和摄像机噪声等自然现象影响[21]后就失去了对抗性，神经网络依然能正确分类。针对对抗性样本缺乏可迁移性这一弊端，本文算法在 FGSM 生成的对抗性样本上采用 EOT(expectation over transformation)[24]框架来产生扰动。采用 EOT 框架的关键是在优化模型过程中模拟扰动，但不是对一个单一样本的对数似然优化，而是选择经过变换后的对抗性样本的集合 T 做优化[24]，即将对抗样本 X' 变换为 $t(X')$ 作为解码器网络的输入。简单来说，EOT 框架会模拟对抗性样本 X' 的各种变换结果，这些变换可以包含缩放、旋转、平移等几何类攻击以及噪声、压缩等信号类攻击，然后将所有这些变换的结果进行融合训练。

2.2　解码器

解码器网络使用经过预训练的轻量化网络 SqeezeNet[25]，该网络具有参数数量少、准确率高的特点。SqeezeNet 网络结构共有 13 层，本文使用其中的第 6、8、10、12 层，并在 SqeezeNet 后面连接 1 个最大池化层。需对经过最大池化后的数据进行标准化处理，以数据本身的均值和方差作为标准化处理时的均值和方差。解码器网络的最后一层是全连接层，经过全连接层，对抗性样本被解码输出为与水印相同长度的一维数组。解码器结构如图 2 所示。SqeezeNet 中，Fire 模块示意图如图 3 所示。

本文使用期望最大化 EM (expectation-maximum)算法对解码器进行优化，EM 算法可以间接求解由数据不完整或数据丢失带来隐含变量情况下的模型最大似然估计[26]。EM 算法的思想是利用迭代进行优化，在每次迭代时分为 E 步(expectation-step)和 M 步(maximization-step)这 2 步。不断迭代更新隐含的数据和模型的参数，直到结果收敛，即得到所需要的模型参数。EM 框架下训练解码器的步骤如下。



B 为 batch_size；N 为集合 T 中元素的个数；w_size 为水印的长度

图 2　解码器结构

Fig. 2　Decoder structure

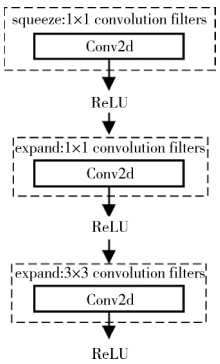


图 3　Fire 模块示意图

Fig. 3　Diagram of Fire module

步骤 1　随机初始化解码器的参数以及产生对抗性样本的扰动 p。

步骤 2　E 步时，根据解码器的参数初始值和扰动的初始值，或者上一次迭代更新的参数和扰动 p，产生对抗性样本 X'。

步骤 3　M 步时，根据对抗性样本 X' 求出似然函数的下界，并利用 Jensen 不等式[27]使该下界最大化，得到新的网络参数值和扰动 p。

步骤 4　将 M 步得到的网络参数和扰动 p 用于接下来的 E 步中，以计算新的对抗性样本，重复步骤 2和步骤 3，直到结果收敛，得到最后的网络参数以及扰动 p。

2.3　水印嵌入与提取过程

水印为二值的一维数组，水印的前 32 位设置为固定的标志位，提取水印时可以根据标志位匹配提取的序列是否为水印。首先读取视频，抽取视频的关键帧，记录关键帧在原视频中的位置，对于每个关键帧，都进行图像归一化操作，随机初始化解码器的参数和扰动 p 的大小，开始训练网络，根据原始的关

键帧和扰动生成对抗性样本 X'，将 X' 作为网络的输入。

利用 EOT 框架计算 X' 在多个变换函数 t 的分布集合 T，并用 SqeezeNet 特定层提取出分布集合 T 的特征。经过最大池化层和全连接层，X' 被解码为与水印相同长度的一维数组。根据 X' 在解码器的输出和对应的水印计算损失函数，更新网络的参数值和扰动 p 的大小。重复 E 步和 M 步的迭代，直到训练的模型达到收敛状态。根据步骤 1 中记录的关键帧的原位置，把模型最后收敛时的对抗性样本 X' 组合到原视频中。

提取水印时，首先读取视频，抽取视频的关键帧，从第一个关键帧开始，每一帧重复以下的操作，即将关键帧作为解码器的输入，解码得到与水印长度相等的一维数组，直到提取出来的序列前 32 位与嵌入的水印标志位相匹配，提取水印过程结束。由于不需要原始视频作为提取水印的过程的参考，所以由上述编码器、解码器组成的端到端的数字水印算法为盲水印算法。

3 实验结果

客观评价视频数字水印性能主要有峰值信噪比(peak signal to noise ratio，PSNR)和归一化相关系数(normalized cross-correlation，NC)这 2 种方法。

3.1 峰值信噪比

图像在嵌入水印后，像素级别会与原始图像有差异，PSNR 值越大，含水印图像与原图像的差异越小，意味着水印的不可见性越高，表示为

$$\mathrm{PSNR}=10\times\lg\frac{\max I_{m,n}^2}{\mathrm{MSE}}(\mathrm{dB})。\tag{8}$$

式中，MSE 为均方误差，且有

$$\mathrm{MSE}=\frac{1}{M\times N}\sum_{m,n}(I_{m,n}-I'_{m,n})^2。\tag{9}$$

式中：I 为原始图像；I' 为嵌入水印的图像；$m\times n$ 和 $M\times N$ 为图像的大小。

3.2 归一化相关系数

原始水印序列和提取出来的水印序列的相似程度可以用归一化相关系数来衡量，表示为

$$\mathrm{NC}=\frac{W\times W'}{\sqrt{W\times W'}}=\frac{\sum_{i=1}^{n}w_i\times w'_i}{\sqrt{\sum_{i=1}^{n}w_i^2\times w'^2_i}}。\tag{10}$$

式中：W 为原始水印序列；W' 为提取出来的水印序列。

本文实验采用深度学习框架 Pytorch，GPU 型号为 GeForce RTX 2080 Ti。选取分辨率大小为 1 920×1 080、帧率为 30 帧/s、时长为 10 min 的视频，共抽取 137 帧关键帧进行实验。为了平衡干扰强度和攻击的成功率，扰动强度 ε 的大小设置为 0.009，训练过程的学习率为 0.002 5，嵌入的水印序列为 256 位随机二值数组。由于对抗性样本经过了变换函数 t 的处理，导致样本的尺度不一致，会影响收敛的结果，因此对对抗性样本进行标准化处理，以加快收敛的速度及提高收敛的精度。图像标准化公式为

$$X''=\frac{t(X')-\mu}{\sigma}。\tag{11}$$

式中：X'' 为经过标准化处理的样本；$t(X')$ 为经过 EOT 框架处理的对抗性样本；μ 为均值；σ 为方差。本文算法使用 Imagenet 数据集的均值和方差，RGB 三通道对应的均值分别为 0.485、0.456、0.406，对应的方差分别为 0.229、0.224、0.225。

通过实验发现，由于视频分辨率高，因此对每一个关键帧解码的计算量大，耗费的时间较长，在上述实验参数情况下，需要 140 min 才能完成全部的水印嵌入过程。为提高算法的效率，对要嵌入水印的关键帧做不同的变换设置用于训练。

实验均采用上述参数，关键帧不同变换设置情况下的实验结果见表 1。表中，在关键帧变换设置分别为 128×128、500×500、960×540 时，在关键帧的图像中心选取一部分作为训练样本，训练结束后再组合到原帧的方法，虽然提高了算法的效率，但是降低了提取水印的 NC 值。在关键帧变换设置为原尺寸×1/3 作为训练样本的实验结果表明，将关键帧缩放为原尺寸的 1/3 作为训练样本不仅耗时少，且 NC 值为 1，水印的提取效果非常好。从视频中选取原始关键帧的图像如图 4 所示。为原始关键帧添加水印的图像如图 5 所示。

表 1 关键帧不同变换设置情况下的实验结果

Table 1 Experimental results of different keyframe transformation settings

变换设置/像素	训练时间/min	NC
128×128	20	0.664 0
500×500	28	0.703 1
960×540	35	0.867 1
原尺寸×1/3	16	1

图 4 原始关键帧

Fig. 4 Original keyframe

将图 4 和图 5 进行对比可知，嵌入水印后，视频帧没有肉眼可见的质量下降，含水印图像的 PSNR 值为 39.39。为了检验本文算法水印的鲁棒性，在实

图 5　含水印关键帧

Fig. 5　Keyframe with watermarking

验中对视频进行包括模糊、噪声、JPEG 压缩、旋转、缩放、平移、裁剪和时间同步攻击等各种攻击。

3.3　运动模糊攻击

模糊可能是由于相机运动和不精确的自动对焦造成的。为了模拟运动模糊，本文算法采用随机角度，并产生具有 3～11 个像素宽度的直线模糊核。运动模糊攻击实验结果见表 2，经过运动模糊攻击后，提取水印的 NC 值仍然很高。模糊核为 19 时的含水印关键帧如图 6 所示，可见，虽然图像已经被破坏得很严重，但是仍然能有效地提取出水印，说明本文算法可以有效地抵抗运动模糊攻击。

表 2　运动模糊攻击实验结果

Table 2　Experimental results of motion blur attack

模糊核/个	NC
3	1
7	1
11	0.988 2
15	0.988 2
19	0.984 3

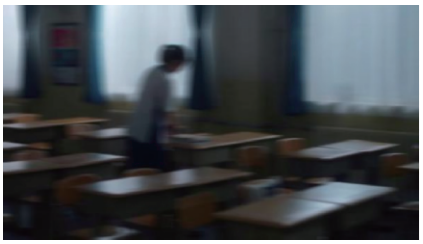


图 6　模糊核为 19 时的含水印关键帧

Fig. 6　Watermarked keyframe with blur kernel of 19

3.4　椒盐噪声攻击

脉冲噪声可能是视频影像信号突然受到干扰而产生的。本算法采用密度为[0.01 0.05]的椒盐噪声模拟变换。椒盐噪声攻击实验结果见表 3，随着椒盐噪声密度的增大，提取出水印的 NC 值下降。

椒盐噪声密度为 0.020 时的含水印关键帧如图 7所示。可见，加了密度为 0.020 的噪声后，能辨认出图像原来的内容，提取出的水印基本有效，说明本文算法可以抗椒盐噪声攻击。

表 3　椒盐噪声攻击实验结果

Table 3　Experimental results of salt and pepper noise attack

密度	NC
0.005	0.964 8
0.010	0.949 2
0.015	0.878 9
0.020	0.867 1

图 7　椒盐噪声密度为 0.020 时的含水印关键帧

Fig. 7　Watermarked keyframe with salt and pepper noise density of 0.020

3.5　JPEG 压缩攻击

由于视频数据中会有大量冗余，因此要对视频进行压缩处理，JPEG 是现有的压缩标准中常用的算法。本文算法采用[10，100]范围内的质量因子控制 JPEG 压缩质量。质量因子是控制压缩质量的参数，取值范围为 0～100，与压缩比率成反比。JPEG 压缩攻击实验结果见表 4。

表 4　JPEG 压缩攻击实验结果

Table 4　Experimental results of JPEG compression attack

质量因子/%	NC
90	1
70	1
50	0.996 0
30	0.984 3

质量因子为 30%时的含水印关键帧如图 8 所示。由图 8 与表 4 可以看出，当压缩的质量因子为 30%时，虽然图像的模糊程度已经能被人眼观察到，但是对提取出水印的 NC 值影响并不大，因此本文算法抵抗 JPEG 压缩的鲁棒性很高。

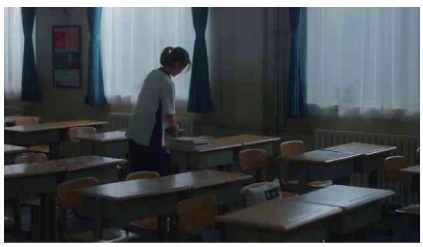


图 8　质量因子为 30%时的含水印关键帧

Fig. 8　Watermarked keyframes with a quality factor of 30%

3.6 旋转攻击

采用沿着图像中心随机地进行[−120°,120°]范围内角度的旋转来模拟视频可能会遇到的旋转攻击。对含水印的关键帧做旋转攻击实验,如图9所示。旋转攻击实验结果见表5,可见:旋转5°、10°时,提取水印的NC值均在0.9以上;旋转20°、30°时,提取水印的NC值下降到0.9以下。实验结果表明,本文算法对于旋转攻击的鲁棒性有待提高。

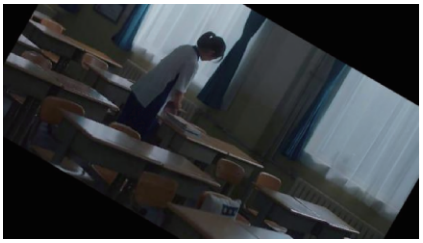


图9 旋转角度为30°时的含水印关键帧

Fig. 9 Watermarked keyframes with rotation angle of 30°

表5 旋转攻击实验结果

Table 5 Experimental results of rotation attack

旋转角度/(°)	NC
5	0.992 1
10	0.945 3
20	0.890 6
30	0.873 4

3.7 缩放攻击

按照图像中心随机进行[0.20,1.79]倍数的缩放进行来模拟视频可能受到的缩放攻击。缩放倍数大于1时,生成的图像大于原图像尺寸,因此将生成的图像裁剪为与原图尺寸相同;缩放倍数小于1时,生成的图像小于原图像尺寸,可进行边缘填充操作,以达到原尺寸。缩放攻击实验结果见表6,可见含水印关键帧的长宽都扩大到原尺寸的150%时,提取水印的NC值仍然为1。

表6 缩放攻击实验结果

Table 6 Scaling attack experiment results

缩放比例/%	NC
120	1
150	1
40	0.996 0
50	0.976 5

缩小比例为50%时的含水印关键帧如图10所示。可见,提取水印的NC值仍然较高。缩放攻击的实验结果表明,正常比例缩放时,本文算法有很高的鲁棒性。

3.8 平移攻击

沿x方向或者y方向进行原尺寸的[0,0.55]倍数的平移来模拟视频可能会遭受的平移攻击。发生

图10 缩小比例为50%时的含水印关键帧

Fig. 10 Watermark keyframes reduced by 50%

平移时,对空缺的部分进行边缘填充。平移比例为30%时,对含水印的关键帧做平移攻击实验,如图11所示。平移攻击实验结果见表7,可见本文算法有较好的抗平移攻击的能力。

图11 平移比例为30%的含水印关键帧

Fig. 11 Watermarked keyframes with a translation ratio of 30%

表7 平移攻击实验结果

Table 7 Experimental results of translation attack

平移比例/%	NC
5	1
10	0.996 0
20	0.996 0
30	0.992 1

3.9 裁剪攻击

采用随机抽取裁剪比例的方式模拟视频可能会遭受的裁剪攻击。裁剪比例的范围为[0.2,1.0],裁剪后图像小于原图像,对空缺的部分用像素值1代替,以达到原尺寸。裁剪攻击实验结果见表8。裁剪比例分别为5%、10%、50%的时含水印关键帧,如图12所示。其中,图12(b)对图像的重要区域进行了裁剪,NC值为0.906 3。实验结果说明,本文算法不能抵抗裁剪重要区域的攻击,但是对于其他类型的裁剪攻击,本文算法仍有较高的鲁棒性。

表8 裁剪攻击实验结果

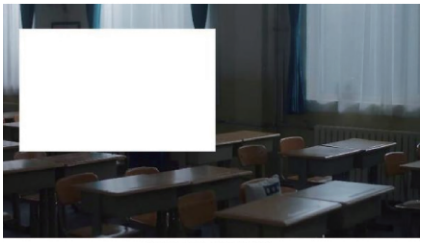
Table 8 Experimental results of tailoring attack

裁剪比例/%	NC
5	0.988 2
10	0.972 6
25	0.906 3
50	0.882 8

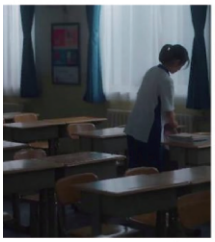
3.10 时间同步攻击

由于本文算法采用的是独立解码每一个关键帧

(a) 裁剪比例为5%



(b) 裁剪比例为10%



(c) 裁剪比例为50%

图 12　不同裁剪比例的含水印关键帧

Fig. 12　Watermarked keyframes with different cropping ratios

的方法，能够抵抗帧插入攻击。由于视频中的 B 帧、P 帧都是根据相邻的 I 帧的信息解码得到的，因此帧的内容与相邻的 I 帧有相似性。而且本文算法在关键帧训练时模拟扰动，与关键帧相邻的帧也可用于提取水印。时间同步攻击实验结果见表 9。

表 9　时间同步攻击实验结果

Table 9　Experimental results of synchronization attack

相邻帧数/帧	NC
2	1
5	1
10	1
15	0.734 3

由表 9 可以看出，与关键帧相邻的第 10 帧提取水印的 NC 值为 1，相邻的第 15 帧与关键帧内容差别较大，已经无法有效提取出水印。实验结果表明，即使有人恶意去除关键帧或与关键帧相邻的某几帧，本文算法仍然可以提取出水印，能够抵抗帧时间同步攻击。

4　结　论

本文针对现有算法难以解决时序同步攻击与几何攻击这一问题，提出了一种端到端的对抗生成式视频数字水印算法。该算法将对抗性样本注入到训练集，并采用了在期望最大化框架下训练解码器的方法。实验结果表明，使用端到端对抗生成方法进行水印嵌入和提取，在对抗 JPEG 压缩、运动模糊、椒盐噪声、缩放、平移、裁剪和时间同步攻击方面具有较好的鲁棒性。未来工作将在此基础上着重考虑针对旋转等攻击的鲁棒性。

（由于印刷关系，查阅本文电子版请登录：http://www.paper.edu.cn/journal/zgkjlw.shtml）

[参考文献]（References）

[1] TIRKEL A Z, RANKIN G A, van SCHYNDEL R M, et al. Electronic watermark [C]// Digital Image Computing, Technology and Applications (DICTA'93). Sydney: [s. n.], 1993: 666-673.

[2] ZHENG D, WANG S, ZHAO J Y. RST invariant image watermarking algorithm with mathematical modeling and analysis of the watermarking processes [J]. IEEE Transactions on Image Processing, 2009, 18(5): 1055-1068.

[3] GU Q L, GAO T G. A novel reversible robust watermarking algorithm based on chaotic system [J]. Digital Signal Processing, 2013, 23(1): 213-217.

[4] MEMON N, WONG P W. Protecting digital media content [J]. Communications of ACM, 1998, 41(7): 34-43.

[5] PODILCHUK C I, DELP E J. Digital watermarking: algorithms and applications [J]. IEEE Signal Processing Magazine, 2001, 4(18): 33-46.

[6] BENDER W, GRUHL D, MORIMOTO N, et al. Techniques for data hiding [J]. IBM Systems Journal, 1996, 35(3/4): 313-336.

[7] SWANSON M D, ZHU B, TEWFIK A H. Multiresolution scene-based video watermarking using perceptual models [J]. IEEE Journal on Selected Areas in Communications, 1998, 16(4): 540-550.

[8] CRAVER S, MEMOM N, YEO B L, et al. Resolving rightful ownerships with invisible at ermarking techniques: limitations, attacks, and implications [J]. IEEE Journal on Selected Areas in Communications, 1998, 16(4): 573-586.

[9] TSAI J S, HUANG W B, KUO Y H. On the selection of optimal feature region set for robust digital image watermarking [J]. IEEE Transactions on Image Processing, 2011, 20(3): 735-743.

[10] LIN Y T, HUANG C Y, LEE G C. Rotation, scaling, and translation resilient watermarking for images [J]. IET Image Processing, 2011, 5(4): 328-340.

[11] DONG P, BRANKOV J G, GALATSANOS N P, et al. Digital watermarking robust to geometric distortions [J]. IEEE Transactions on Image Processing, 2005, 14(12): 2140-2150.

[12] ZHU J R, KAPLAN R, JOHNSON J, et al. Hidden: Hiding data with deep networks [J]. arXiv, 2018: 1807.09937v1.

[13] TANCIK M, MILDENHALL B, NG R. StegaStamp:

invisible hyperlinks in physical photographs [J]. arXiv, 2019: 1904.05343.

[14] LUO X Y, ZHANG R H, CHANG H W, et al. Distortion agnostic deep watermarking [C]// 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). New York: IEEE, 2020: 19876586.

[15] LIU Y, GUO M X, ZHANG J, et al. A novel two-stage separable deep learning framework for practical blind watermarking [C]// Proceedings of the 27th ACM International Conference on Multimedia. [S. l.: s. n.], 2019: 1509-1517.

[16] GOODFELLOW I J, SHLENS J, SZEGEDY C. Explaining and harnessing adversarial examples [J]. arXiv, 2015: 1412.6572.

[17] DHILLON G S, AZIZZADENESHELI K, LIPTON Z C, et al. Stochastic activation pruning for robust adversarial defense [J]. arXiv, 2018: 1803.01442.

[18] CHEN P-Y, ZHANG H, SHARMA Y, et al. ZOO: zeroth order optimization based black-box attacks to deep neural networks without training substitute models [C]// Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security. New York: ACW, 2017: 15-26.

[19] BUCKMAN J, ROY A, RAFFEL C, et al. Thermometer encoding: One hot way to resist adversarial examples [C]// ICLR 2018 Conference Blind Submission. [S. l.: s. n.], 2018: 1-22.

[20] BIGGIO B, CORONA I, MAIORCA D, et al. Evasion attacks against machine learning at test time [C]// Joint European Conference on Machine Learning and Knowledge Discovery in Databases. [S. l.: s. n.], 2013: 387-402.

[21] ALZANTOT M, SHARMA Y, ELGOHARY A, et al. Generating natural language adversarial examples [J]. arXiv, 2018: 1804.07998.

[22] FAWZI A, FAWZI O, FROSSARD P. Analysis of classifiers' robustness to adversarial perturbations [J]. Machine Learning, 2018, 107(3): 481-508.

[23] ARNAB A, ZHENG S, JAYASUMANA S, et al. Conditional random fields meet deep neural networks for semantic segmentation: combining probabilistic graphical models with deep learning for structured prediction [J]. IEEE Signal Processing Magazine, 2018, 35(1): 37-52.

[24] ATHALYE A, ENGSTROM L, ILYAS A, et al. Synthesizing robust adversarial examples [J]. arXiv, 2017: 1707.07397.

[25] IANDOLA F N, HAN S, MOSKEWICZ M W, et al. SqueezeNet: alexNet-level accuracy with 50x fewer parameters and <0.5 MB model size [J]. arXiv, 2016: 1602.07360.

[26] DEMPSTER A P, LAIRD N M, RUBIN D B. Maximum likelihood from incomplete data via the EM algorithm [J]. Journal of the Royal Statistical Society: Series B (Methodological), 1977: 39(1): 1-22.

[27] **吴善和. 几何凸函数与琴生型不等式[J]. 数学的实践与认识**, 2004, 34(2): 155-163.
WU S H. Geometric convex function and jensen type inequality [J]. Mathematics in Practice and Theory, 2004, 34(2): 155-163. (in Chinese)