Software Engineering and Applications 软件工程与应用, 2022, 11(1), 130-138
Published Online February 2022 in Hans. http://www.hanspub.org/journal/sea
https://doi.org/10.12677/sea.2022.111015

Hans汉斯

基于帧间特征矩阵的同源视频检测

张雪梅[1]，康宇哲[1]，沈学文[2]

[1]浙江理工大学，浙江 杭州
[2]浙江传媒学院，浙江 杭州

收稿日期：2022年1月10日；录用日期：2022年2月9日；发布日期：2022年2月16日

摘　要

近年来，随着数字媒体软件的火热，网络中的近重复相似视频呈爆炸似增长，因此快速准确地筛选出海量视频中的同源视频是当下研究的重点课题。针对该课题，本文采取了基于帧间特征矩阵的同源视频检测方案，首先在视频帧间时空关系矩阵的基础上确定视频相应类别，然后进一步通过视频帧间特征序列对比来确认所检测视频是否与该类下的其他视频存在重复片段，并定位重复片段在视频中的位置。当重复片段占比超过一定阈值，即可判定被检测视频为同源视频。实验表明该方法在CC_WEB数据集上平均准确率可达93.2%，由此证明了该方法在保护视频知识产权领域的可用性。

关键词

帧间时空特征，三帧差分法，帧间特征序列

Homologous Video Detection Based on Inter-Frame Feature Matrix

Xuemei Zhang[1], Yuzhe Kang[1], Xuewen Shen[2]

[1]Zhejiang Sci-Tech University, Hangzhou Zhejiang
[2]Communication University of Zhejiang, Hangzhou Zhejiang

Received: Jan. 10th, 2022; accepted: Feb. 9th, 2022; published: Feb. 16th, 2022

Abstract

In recent years, with the popularity of digital media software, the nearly repeated similar videos in the network are exploding. Therefore, it is a key topic of current research to quickly and accurately screen out the homologous videos in the massive videos. For the subject, this paper adopted homologous video detection scheme based on characteristic matrix between frames. First, the corresponding categories of the video are determined on the basis of the space-time relationship

文章引用: 张雪梅, 康宇哲, 沈学文. 基于帧间特征矩阵的同源视频检测[J]. 软件工程与应用, 2022, 11(1): 130-138.
DOI: 10.12677/sea.2022.111015

matrix between the video frames, and then through the characteristics of sequence comparison between video frames to confirm whether the detected video has duplicate segments with other videos under this category, and locate the position of duplicate segments in the video. When the proportion of repeated clips exceeds a certain threshold, the detected video can be judged as homologous video. Experimental results show that the MAP (Mean Average Precision) of this method can reach 93.2% on CC_WEB data set, which proves the applicability of this method in the field of video intellectual property protection.

Keywords

Inter-Frame Feature Matrix, Three-Frame Difference Method, Inter-Frame Feature Sequence

1. 引言

随着互联网视频共享服务的普及，近年来网络上的视频内容量达到了前所未有的规模[1]。据中国网络版权监测报告显示，2019 年新增监测影视等领域精品版权作品超过 628 万件，同比增长超 200%，主要增长集中在短视频、游戏及图片等领域[2]。随之而来的就是原创视频的盗版问题，近几年来，由于视频技术的发展，对原创视频的搬运，剪辑行为愈演愈烈，主要体现为：原视频的内容被打码，增加画中画，以及未经授权被用于二次创作剪辑等[3]。因此，打击盗版视频，保护知识产权已经成为亟待解决的问题，而如何快速准确地鉴别同源视频更是解决该问题的首要任务[4]。

本文就此研究课题提出了基于帧间特征矩阵的重复视频检测方法，该方法首先通过帧间时空特征提取网络判断被检测视频所属类别，并计算其与视频库中视频的相似度，然后通过帧间特征序列对比方案来定位被检测视频中的重复片段，从而实现对视频的同源性检测。

2. 相关研究

近几年来，众多学者致力于同源视频检测技术的研究。该研究的关键是找到鲁棒性强的检测方案以准确的检索并定位被检测视频的同源片段。目前的同源视频检测方法主要分为三大类：基于全局特征检测、基于局部关键点检测以及基于视频时空特征检测[5]。张乃光等人使用卷积神经网络通过比较切帧提取的特征来对视频进行检测分类，该方法只能做到基本的分类且准确率只有 79.4% [6]；Hsu [7]和黄添强[8]等人采用提取视频帧中的噪声特征进而比较其相似度来确定非同源帧的插入及修改，该方法专注于粒度较小的视频帧，时间复杂度较高；范清宇[9]等人提出了基于 SimRank 算法的音视频数据同源性分析方法，但仅局限在静态数据且检索效率有待优化；Jiang 等人证实在 VCDB 数据集上，使用时序网络方法有更好的召回率[10]。

结合上述方法的优势，本文提出一种基于帧间特征矩阵的重复视频检测方法，该方法既利用了帧间时空特征矩阵检测速度快的特性，又结合了帧间特征序列对比精度高的特点。该方法首先使用帧间时空特征矩阵对被检测视频进行类别划分并给出该视频与视频库中视频的相似度，然后通过将该视频与视频库中相似度最高的视频进行帧间特征序列对比，判断视频是否有重复片段并定位重复片段出现的时间轴，最终实现对视频同源性的检测以及对同源视频中重复片段的定位。

3. 帧间时空特征提取网络设计

本节将对所提出的帧间时空网络检测方案做详细的介绍，本方法具体流程可以分为三个阶段：帧间时空特征提取模型结构设计，帧间时空特征提取模型损失函数设计和模型训练。

3.1. 帧间时空特征提取模型结构设计

视频片段是一个连续的前后关联的图片集合，帧间具有前后语意特征。因此在设计视频片段特征提取网络时，既要提取单个视频帧的图像特征信息，也要着重提取视频帧间的前后语意信息。

帧间时空特征提取模型的结构如图 1 所示，其中图 1(a)描述视频片段特征提取网络 LeNet3D 的结构，图 1(b)描述 Residue 模块。本文在图像特征提取网络常用的二维度卷积[11]基础上引入三维度卷积[12]，设计了一个视频片段特征提取网络 LeNet3D，其结构如图 1(a)所示，该结构使用 2 个带有最大池化的三维度卷积对视频片段进行语意信息与图像信息特征提取，将得到的特征图输入 3 个 Residue 模块后分别抽取 3 组特征图输入到 concat 结构[13]里面进行特征融合。该网络以 LeNet [14]作为骨干网络可以同时提取视频帧特征信息和视频帧间语意信息。同时引入 Residue 模块[15]如图 1(b)所示，该模块将输入特征图与卷积后的输出特征图进行元素相加操作并向后传播[16]，以此缓解由于网络过深造成的反向传播时的梯度消失问题。该特征提取结构与 VISIL 模型[17]相比添加了时间轴维度卷积，可以使用更简单的模型结构提取帧间的语意特征信息。

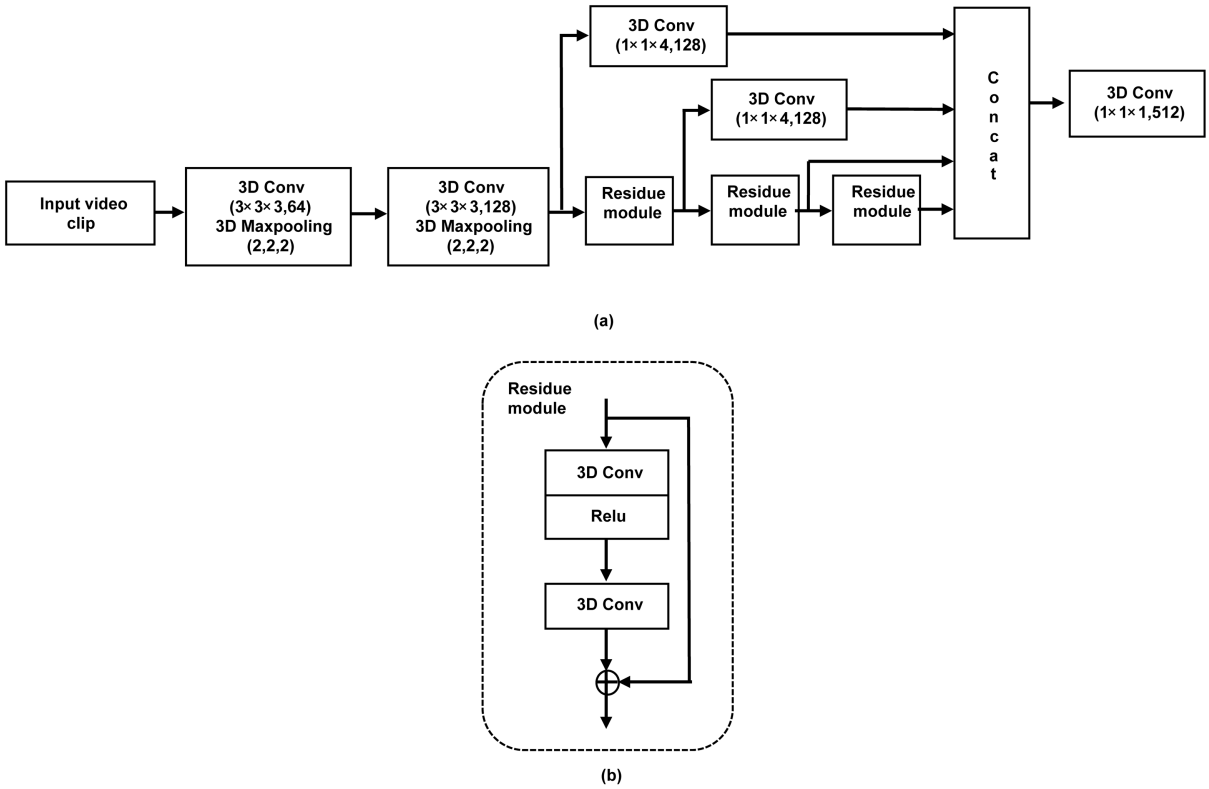

Figure 1. LeNet3D network structure
图 1. LeNet3D 网络结构

首先将数据集中的视频分割为 150 帧的视频片段并将视频片段的长和高变换为 112 像素。将两个片段分别通过 LeNet3D 结构提取特征后通过自适应池化得到尺寸为 7 × 512，512 × 7 的特征序列并使用矩

阵点乘获得 512 阶相似度方阵，最后使用倒角相似度算法[18]计算两个视频片段的相似度，详见公式(1)，该公式主要描述两视频相似度的计算方式，如下所示：

$$CS_f\left(M_b,M_d\right)=\frac{1}{N}\sum_{i,j=1}^{N}\max_{i,j\in[1,N]} f_1 * f_2 \tag{1}$$

在公式(1)中，将输入网络的两个视频片段定义为 M_b,M_d，N 定义为特征方阵的阶数，本文取 512，f_1,f_2 为特征序列。视频片段相似度计算流程如图 2 所示。该过程利用区域向量捕获几何信息，并提供一定程度的空间不变性。

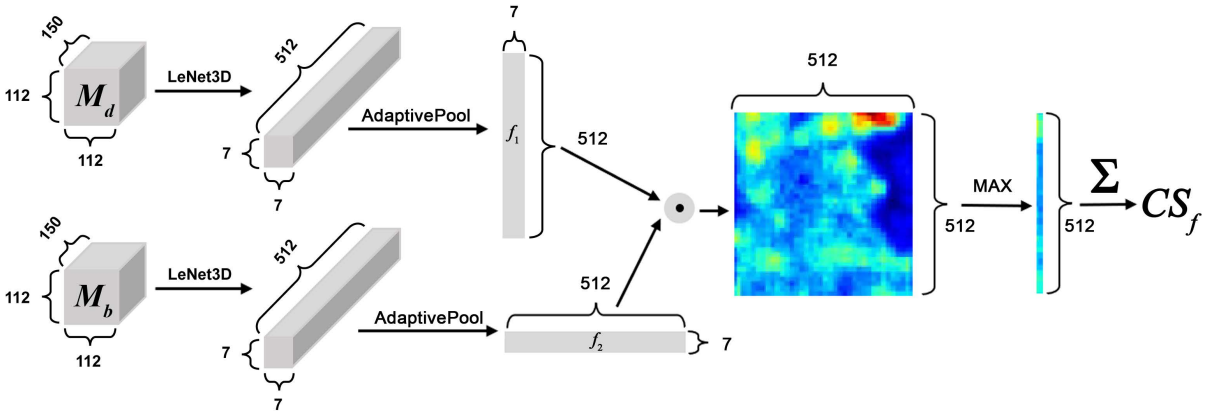

Figure 2. Video clip similarity calculation process
图 2. 视频片段相似度计算流程

3.2. 帧间时空特征提取模型损失函数设计

由于相关视频的目标视频相似度得分 CS_f 应较高，不相关视频的 CS_f 应较低。为了训练设计好的网络，本文将视频数据集分为两个部分输入网络。其中一路随机输入一个视频片段，另一路输入与之相似或者完全不相似的视频片段并携带一个视频相似度标签，该标签是一个带有两位有效数字的维度为 0~1 之间的小数，该数字代表与第一路视频的相似度。为了使网络将更高的相似性分数分配给正确的视频对，将更低的相似性分数分配给负样本视频对，本文设计的损失函数如公式(2)

$$L_{tr}=\max\left\{0,CS_f\left(v,v^-\right)-CS_f\left(v,v^+\right)+\gamma\right\} \tag{2}$$

其中 γ 是一个阈值参数，为视频样本，为视频负样本，为视频正样本。此外，本文定义了一个相似性正则化函数，该函数会抑制 tanh 输入中可能导致饱和输出的高值。可以使网络在范围[−1, 1]内输出相似度矩阵。为了计算正则化损失，只需将剪切范围之外的输出相似性矩阵中的所有值相加如公式(3)。

$$L_{reg}=\sum_{i=1}^{X'}\sum_{j=1}^{Y'}\left|\max\left\{0,S_v^{qp}\left(i,j\right)-1\right\}\right|+\left|\min\left\{0,S_v^{qp}\left(i,j\right)+1\right\}\right| \tag{3}$$

最后，公式(4)给出了总损耗函数。

$$L=L_{tr}+r*L_{reg} \tag{4}$$

其中，r 是一个正则化超参数，用于调整相似性正则化对总损失的贡献。

3.3. 模型训练

本文通过在线视频平台获取的视频片段进行变换，人工生成正样本视频。通过三类变换：i) 颜色，包

括转换为灰度、亮度、对比度、色调和饱和度调整，ii) 几何，包括水平或垂直翻转、裁剪、旋转、调整大小和重新缩放，以及 iii) 时间，包括慢动作、快进、帧插入、视频暂停或反转。在训练期间，从每个类别中随机选择一个变换并应用于所选视频。我们构造了两个由正样本视频对组成的视频池。首先，对于每一个正样本视频对，将相似度较高的两个视频的标签值设置在 0.80~0.99 范围，然后在两个视频池内随机选择两个视频组成负样本对，将几乎没有相关性的两个视频的标签值设置在 0.10~0.40 范围内。最后随机将正负视频对输入网络进行训练。为尽可能提高模型的鲁棒性，后续训练时将 batch_size 设置为 16。

4. 视频特征序列对比

经过第三节帧间时空特征矩阵检测方案确定待检测视频所属视频类别后，本节采用视频特征序列对比方案来判断视频是否有重复片段并定位重复片段出现的时间轴。

4.1. 对视频进行初始化

取视频帧的长宽值，遍历视频的每一帧。

4.2. 三帧差分法获取视频帧的运动轮廓

三帧差分法是将连续的三帧图像进行两两差分，然后将其进行二值化处理，最后将所得到的两幅差分图像进行逻辑“与”运算，以此检测出物体在中间一帧的位置[19]，其运算原理如下所述：

$f_{k-1}(x,y)$、$f_k(x,y)$、和 $f_{k+1}(x,y)$ 为相邻的三帧图像，将 $f_{k-1}(x,y)$ 和 $f_k(x,y)$，$f_k(x,y)$ 和 $f_{k+1}(x,y)$ 分别进行差分，差分后得到差分图像分别为 $A_{(k-1,k)}(x,y)$ 和 $A_{(k,k+1)}(x,y)$，如公式(5)所示：

$$\begin{cases} A_{(k-1,k)}(x,y) = f_k(x,y) - f_{k-1}(x,y) \\ A_{(k,k+1)}(x,y) = f_{k+1}(x,y) - f_k(x,y) \end{cases} \tag{5}$$

对差分图像进行二值化，如公式(6)所示其中 T 代表预设阈值[20]，若将该值设定过大，则易造成个别帧的漏检；若设定过小，易产生噪声[21]，根据多次实验发现：设定阈值 $T=25$ 时可取得最佳效果。

$$\begin{aligned} B_{(k,k-1)}(x,y) &= \begin{cases} 1 & A_{(k,k-1)}(x,y) \geq T \\ 0 & A_{(k,k-1)}(x,y) < T \end{cases} \\ B_{(k,k+1)}(x,y) &= \begin{cases} 1 & A_{(k,k+1)}(x,y) \geq T \\ 0 & A_{(k,k+1)}(x,y) < T \end{cases} \end{aligned} \tag{6}$$

对所得的二值化差分图像进行逻辑“与”运算得到三帧差分图像 $D_k(x,y)$：

$$D_k(x,y) = \begin{cases} 1 & B_{(k,k-1)}(x,y) \cap B_{(k-1,k)}(x,y) = 1 \\ 0 & B_{(k,k-1)}(x,y) \cap B_{(k-1,k)}(x,y) = 0 \end{cases} \tag{7}$$

4.3. 计算运动点与静止点的差值

差分图像经过上述操作之后，此时图像中只存在像素值为 0 的白色像素点和像素值为 255 的黑色像素点，将图像中白色像素点记为运动点，像素点数目记为 m，黑色像素点数目记为静止点，像素点数目为 q，取移动像素点与静止像素点的差值，记作 V_{point}：

$$V_{point} = |m - q| \tag{8}$$

4.4. 计算动静点差值的比重

记该差值占图像总像素点的比值为 Vp

$$Vp = \frac{V_{point}}{thresh.shape[0] \times thresh.shape[1]} \tag{9}$$

其中V_{point}是运动点与静止点的差值，$thresh.shape[0]$是每帧图像的长度，$thresh.shape[1]$是每帧图像的宽度。

4.5. 获取视频特征序列

当前视频帧中动静点差值的比重 VP 大于阈值时，将该帧标记为 1，小于阈值时，标记为 0。在本方案中，通过比较多次实验所得的结果来选取该阈值的数值，实验表明将 VP 阈值设为 0.7 时实验效果最佳，因此本实验选取 0.7 作为 VP 的阈值。经过上述步骤可以得到一个由 0，1 构成的一维视频特征序列，其中，序列长度等于该视频的总帧数。

4.6. 比较视频的特征序列

对两视频的特征序列进行相与操作，若两视频序列长度相同，计算相与结果为 1 的片段长度占视频总帧长的比值，当该值超过设定阈值时，即判断两视频同源。若两视频特征序列长度不同，则将长度较小的特征序列作为滑动窗口在较长的特征序列上进行步长值为 1 的滑动相与操作，并记录每次滑动操作的结果。计算相与结果为 1 的片段长度占视频总帧长的比值，同样当该值超过设定阈值时，判断两视频间存在重复片段，并可根据帧间间隔可定位重复片段在视频中的位置。

5. 实验结果与分析

本论文中对比实验 GPU 服务器硬件环境如下：

1) CPU：Intel(R)SilverXeon4114*2，主频 2.2 GHz 10 核 20 线程；

2) 内存：128 GB DDR4REGECC3600 MHz；

3) GPU：NVIDIATESLAV100*216 GB GBBR5；

4) 操作系统：Ubuntu18.04 STL；

5) 硬盘：1TBSSDPCIE*2

本论文中对比实验软件环境如下：

1) CUDA10cuDNN7.6.5；

2) Python 3.6；

3) Tensor Flow-gpu 1.15.0；

4) Opencv-Python 3.4.1；

5) Keras 2.1.3

实验数据：测试视频数据集使用 CC_WEB_VIDEO，它是由香港城市大学视频检索小组 VIDEO 提供的一个用于视频同源性检测的开放的专用数据集，包含了 24 个查询视频和 12,790 个目标视频以及人工标注的近似重复视频基准结果。

评价指标：

1) 平均准确率(mean average precision, MAP)：反映的是系统在检索相关视频文档的性能指标[22]。

$$MAP_{(q)} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{m_i}\sum_{j=1}^{m_i} Precision\left(R_{ij}\right) \tag{10}$$

其中，q 表示查询集，N 表示查询集的个数，m_i 表示相关文档的个数，$Precision\ (R_{ij})$表示返回的结果中第 j 个相关文档在返回结果的位置与该文档在返回结果中的位置。

2) 耗时：在本实验配置环境下识别完成所有数据集视频所需的时间，单位为秒。

6. 帧间时空特征提取模型在 CC_WEB_VIDEO 数据集的表现

本身使用了 SGD [23]优化器对帧间模型进行训练，经过对学习率和 γ 阈值参数的调整，最终确定学习率为 0.0004，$\gamma = 0.5$，$r = 0.1$，在训练 100 epoch，200,000 次迭代后得到了最佳的模型训练结果。

实验结果及分析：为了评估该同源视频检测方案的好坏，本实验选取了 24 个查询视频，用准确率和耗时两项指标进行实验效果的评估。从图 3，图 4 可以看出，本方案与 ViSiL 方案相比较，平均准确率更高且检索耗时更少，虽然有个别类别标签视频相似度结果预测较差，这是因为帧间时空特征模型在某些需要精确定位视频帧相似的地方性能较差，总体而言本案性能优于 ViSiL 方案。同样，将本方案与另外两种经典的视频检测方案：GF [24]，SPHL [10]作比较，如表 1 所示，本文方法的平均准确率均高于另外两种方案，由此可见本方案在 MAP 和耗时两项指标上均实现了一定的提升。

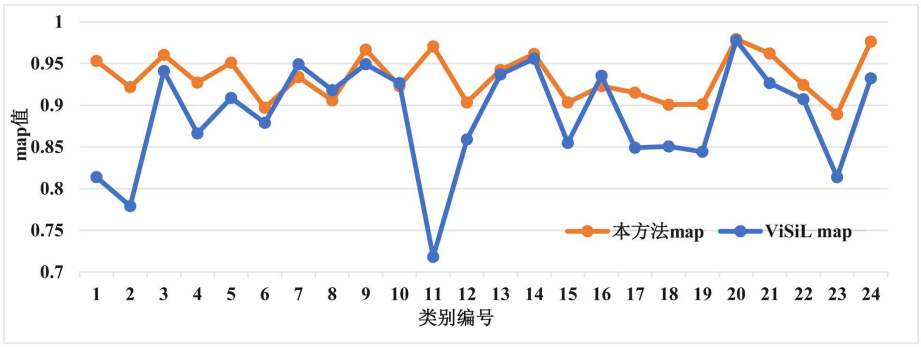

Figure 3. MAP curve of homologous video detection scheme and VISIL scheme
图 3. 同源视频检测方案与 ViSiL 方案 MAP 曲线

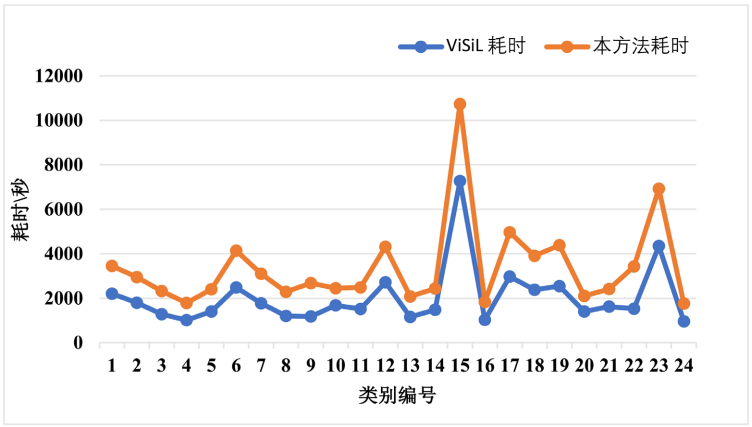

Figure 4. Time-consuming curve of homologous video detection scheme and VISIL scheme
图 4. 同源视频检测方案与 VISIL 方案的耗时曲线

Table 1. The comparison of MAP values
表 1. MAP 值的比较

检测方案	MAP
GF	0.892
SPH	0.826
ViSiL	0.887
本方法	0.932

7. 结束语

本文主要针对视频检测领域的同源性判定问题提出了基于帧间特征矩阵的同源视频检测方案，该方案首先采用帧间时空特征矩阵检测方案确定被检测视频所属类别，然后通过帧间特征序列对比方案将被检测视频与视频库中相似度最高的视频进行帧间特征序列对比，从而确定被检测视频是否与视频库中视频的同源性并定位同源片段所出现的时间轴，最终实现对视频同源性的判断与同源判断的定位。实验表明，该方法与传统方法相比，在保证检索效率的基础上，有效提高了检索的准确率，在今后的研究中有一定的参考价值。

参考文献

[1] 顾佳伟, 赵瑞玮, 姜育刚. 视频拷贝检测方法综述[J]. 计算机研究与发展, 2017, 54(6): 1238-1250.

[2] 12426 版权监测中心. 2019 年中国网络版权监测报告(摘要) [N]. 中国新闻出版广电报, 2020-05-14(007).

[3] 王娜. 我国网络视频产业的版权困局与破解[J]. 当代电影, 2017(10): 193-195.

[4] 李小琛. 数字视频同源帧内复制－粘贴篡改取证研究[D]: [硕士学位论文]. 福州: 福建师范大学, 2018.

[5] 胡瑞娟. 网络舆情中的同源视频检测[D]: [硕士学位论文]. 天津: 中国民航大学, 2014.

[6] 张乃光, 李珊珊, 薛子育. 基于深度学习的盗版视频分类[J]. 广播电视信息, 2019(S1): 84-87.

[7] Hsu, C.C., Hung, T.Y., Lin, C.W., *et al*. (2008) Video Forgery Detection Using Correlation of Noise Residue. *Proceedings of the* 2008 *IEEE* 10*th Workshop on Multimedia Signal Processing*, Cairns, 8-10 October 2008, 170-174.

[8] 黄添强, 吴铁浩, 袁秀娟, 陈智文. 利用模式噪声聚类分析的视频非同源篡改检测[J]. 计算机科学与探索, 2011, 5(10): 914-920.

[9] 范清宇. 音视频数据获取与同源性分析关键技术研究[D]: [硕士学位论文]. 成都: 电子科技大学, 2018.

[10] Jiang, Y.G. and Wang, J. (2016) Partial Copy Detection in Videos: A Benchmark and an Evaluation of Popular Methods. *IEEE Transactions on Big Data*, **2**, 32-42. https://doi.org/10.1109/TBDATA.2016.2530714

[11] 栗志磊, 李俊, 施智平, 姜那, 张永康. 用于视频行为识别的高效二维时序建模网络[J/OL]. 计算机工程与应用, 2021.

[12] 詹克羽, 孙岳, 李颖. 一种多尺度三维卷积的视频超分辨率方法[J]. 西安电子科技大学学报, 2021, 48(5): 8-14.

[13] Lin, T.Y., Dollár, P., Girshick, R., *et al*. (2017) Feature Pyramid Networks for Object Detection. *Proceedings of the* 2017 *IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, 21-26 July 2017, 2117-2125. https://doi.org/10.1109/CVPR.2017.106

[14] Lecun, Y., Bottou, L., Bengio, Y., *et al*. (1998) Gradient-Based Learning Applied to Document Recognition. *Proceedings of the IEEE*, **86**, 2278-2324. https://doi.org/10.1109/5.726791

[15] He, K.M., Zhang, X.Y., Ren, S.Q., *et al*. (2016) Deep Residual Learning for Image Recognition. *Proceedings of the* 2016 *IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, 27-30 June 2016, 770-778. https://doi.org/10.1109/CVPR.2016.90

[16] Hara, K., Kataoka, H. and Satoh, Y. (2018) Can Spatiotemporal 3D CNNS Retrace the History of 2D CNNS and ImageNet? *Proceedings of the* 2018 *IEEE*/*CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, 18-23 June 2018, 6546-6555. https://doi.org/10.1109/CVPR.2018.00685

[17] Kordopatis-Zilos, G., Papadopoulos, S., Patras, I., *et al*. (2019) ViSiL: Fine-Grained Spatio-Temporal Video Similarity Learning. *Proceedings of the* 2019 *IEEE*/*CVF International Conference on Computer Vision*, Seoul, 27 October-2 November 2019, 6351-6360.

[18] 樊丽萍. Chamfer 距离特征提取的数据匹配算法研究[D]: [硕士学位论文]. 南昌: 南昌大学, 2012.

[19] 孙挺, 齐迎春, 耿国华. 基于帧间差分和背景差分的运动目标检测算法[J]. 吉林大学学报(工学版), 2016, 46(4): 1325-1329.

[20] 王霏. 基于视频的运动目标检测算法研究[D]: [硕士学位论文]. 长春: 吉林大学, 2014.

[21] 亢洁, 李晓静. 基于均值背景与三帧差分的运动目标检测[J]. 陕西科技大学学报, 2018, 36(1): 148-153

[22] 王飞飞. 基于稀疏自动编码器的近重复视频检索[J]. 电子技术与软件工程, 2017(3): 194-196.

[23] Chen, H., Zheng, L., Al Kontar, R., *et al*. (2020) Gaussian Process Inference Using Mini-Batch Stochastic Gradient

Descent: Convergence Guarantees and Empirical Benefits.

[24] Wu, X., Hauptmann, A.G. and Ngo, C. (2007) Practical Elimination of Near-Duplicates from Web Video Search. *Proceedings of the* 15*th ACM international conference on Multimedia*, Augsburg, 25-29 September 2007, 218-227. https://doi.org/10.1145/1291233.1291280