

Estimation of Video Authenticity through Collaborative Use of Available Video Signals

Michael Penkov[†], Takahiro Ogawa (member)[†], Miki Haseyama (member)[†]

Abstract This paper proposes a method for identifying the video that retains the most information from its parent video. Since the parent video is often unavailable, the proposed method estimates its content through the collaborative use of the available video signals that are edited copies of the parent video. By reducing the difference between the video signals of the edited videos, the proposed method then enables the use of conventional no-reference video quality assessment algorithms. Since editing a video requires recompressing it, and since quality assessment algorithms can detect signs of recompression, the proposed method can identify the edited video that retains the most information from the parent video. The effectiveness of the proposed method is verified by subjective experiments over artificial and real-world data sets that include a total of over 400 videos.

Key words: video authenticity, video quality, near-duplicate video, video retrieval, video phylogeny.

1. Introduction

Videos uploaded to Web portals (hereafter, “Web videos”) have become an important medium for sharing information, with hundreds of millions of videos viewed daily. People increasingly view Web videos to satisfy their information need for news and current affairs. In general, there are two kinds of Web videos: (a) parent videos, which are created when a camera images a real-world event; and (b) edited copies of a parent video that is already available somewhere. Creating such copies may involve video editing, for example, by shot removal, resampling and recompression, to the extent that they misrepresent the event they are portraying. Such edited copies can be undesirable to people searching for videos to watch¹⁾, since the people are interested in viewing an accurate depiction of the event in order to satisfy their information need. In other words, they are interested in the video that retains the most information from the parent video – the most *authentic* video.

Since Web videos contain metadata that includes the upload timestamp and view count, a simple method for determining authenticity is to focus on the metadata. In general, videos that were uploaded earlier or receive much attention from other users are less likely to be edited copies of existing videos. However, since

the metadata is user-contributed and not directly related to the actual video signal, it is often incomplete or inaccurate. Therefore, using metadata by itself is not sufficient, and it is necessary to examine the actual video signal of the Web video.

Conventional methods for estimating the authenticity of a video can be divided into two categories: active and passive. Active approaches, also known as *digital watermarking*, embed additional information into the parent video, enabling its editing history to be tracked²⁾. However, their use is limited, as not all videos contain embedded information. Passive approaches, also known as *digital forensics*, make an assessment using only the video in question without assuming that it contains explicit information about its editing history³⁾. However, forensic algorithms are typically limited to examining videos individually, and cannot directly and objectively estimate the authenticity of videos.

Recently, forensic techniques have been extended to studying not only individual videos, but also the *relationships* between videos that are near-duplicates of each other. For example, *video phylogeny* examines causal relationships within a group of videos, and creates a tree to express such relationships⁴⁾. Other researchers have focused on reconstructing the parent sequence from a group of near-duplicate videos⁵⁾. The reconstructed sequence is useful for understanding the motivation behind creating the near-duplicates: what was edited and, potentially, why.

We approach the problem of authenticity estimation from a radically different direction: *no reference visual*

Received April 23, 2015; Accepted May 19, 2015

[†] Graduate School of Information Science and Technology, Hokkaido University
(N-14, W-9, Kita-ku, Sapporo 060-0814, Japan)

*quality assessment*⁶⁾ (hereafter, “NRQA”). NRQA algorithms evaluate the strength of common quality degradations such as blurring⁷⁾. Our motivation for introducing these algorithms is as follows: editing a video requires recompressing the video signal, which is usually a lossy operation that causes a decrease in visual quality⁸⁾. Since the edited videos were created by editing the parent video, the parent will have a higher visual quality than any of the edited videos created from it. Therefore, visual quality assessment is relevant to authenticity estimation. However, since the video signals of the edited videos can differ significantly due to the editing operations, directly applying the algorithms to compare the visual quality of the edited videos is difficult⁹⁾.

This paper, which refines and expands on our previous work¹⁰⁾, proposes a method that identifies the most authentic video by estimating the proportion of information remaining from the parent video, even if the parent is not available. We refer to this measure as the “authenticity degree”. The novelty of this proposed method consists of four parts: (a) in the absence of the parent video, we estimate its content by comparing edited copies of it; (b) we reduce the difference between the video signals of edited videos before performing a visual quality comparison, thereby enabling the application of conventional NRQA algorithms; (c) we collaboratively utilize the outputs of the NRQA algorithms to determine the visual quality of the parent video; and (d) we enable the comparison of NRQA algorithm outputs for visually different shots. Finally, since the proposed method is capable of detecting shot removal, scaling, and recompression, it is effective for identifying the most authentic edited video. To the best of our knowledge, we are the first to apply conventional NRQA algorithms to videos that have significantly different signals, and the first to utilize video quality to determine the authenticity of a video. The proposed method has applications in video retrieval: for example, it can be used to sort search results by their authenticity. The effectiveness of our proposed method is demonstrated by experiments on real and artificial data.

2. Preliminaries

This section provides initial definitions and introduces the notation used in the remainder of this paper. First, borrowing some notation and terminology from

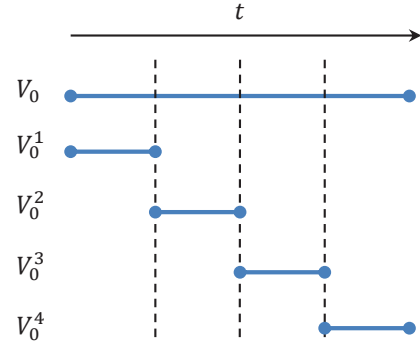


Fig. 1 An example of a parent video and its constituent shots.

existing literature⁵⁾, we define the *parent video* $\mathbf{V}_0$ as a video that contains some new information that is unavailable in other videos. The parent video consists of several shots $\mathbf{V}_0^1, \mathbf{V}_0^2, \dots, \mathbf{V}_0^{N_{V_0}}$, where a shot is defined as a continuous sequence of frames that were captured by the same camera. The top part of Fig. 1 shows a parent video that consists of $N_{V_0} = 4$ shots. The dashed vertical lines indicate shot boundaries. The horizontal axis shows time.

The parent video can be divided into its constituent shots as shown in the bottom part of Fig. 1. Shot boundaries can be detected automatically by comparing the color histograms of adjacent frames, and thresholding the difference¹¹⁾. More specifically, each frame is divided into 4×4 blocks, and a 64-bin color histogram is computed for each block. The difference between corresponding blocks is then computed as:

$$F(\mathbf{f}_1, \mathbf{f}_2, r) = \sum_{x=0}^{63} \frac{\{H(\mathbf{f}_1, r, x) - H(\mathbf{f}_2, r, x)\}^2}{H(\mathbf{f}_1, r, x)}, \quad (1)$$

where $\mathbf{f}_1$ and $\mathbf{f}_2$ are the two frames being compared; r is an integer corresponding to one of the 16 blocks; and $H(\mathbf{f}_1, r, x)$ and $H(\mathbf{f}_2, r, x)$ correspond to the x th bin of the color histogram of the r th block of $\mathbf{f}_1$ and $\mathbf{f}_2$, respectively. The difference between the two frames is calculated as:

$$E(\mathbf{f}_1, \mathbf{f}_2) = \underset{8 \text{ of } 16, r=1 \text{ to } 16}{\text{SumOfMin}} F(\mathbf{f}_1, \mathbf{f}_2, r), \quad (2)$$

where the SumOfMin operator computes Eq. (1) for 16 different values of r and sums the smallest 8 results, and is explained in further detail by its authors¹¹⁾. Given a threshold v , if $E(\mathbf{f}_1, \mathbf{f}_2) > v$ for any two adjacent frames, then those two frames lie on opposite sides of a shot boundary. Although this method often fails to detect certain types of shot boundaries¹²⁾, it is simple,

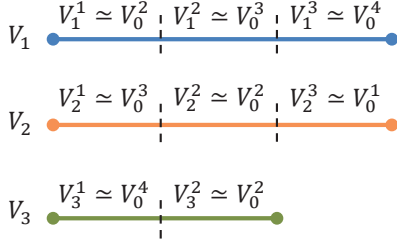


Fig. 2 An example of videos created by editing the parent video of Fig. 1.

effective and sufficient for the purposes of this paper.

After segmentation, the shots can be edited and then rearranged to create *edited videos* $\mathbf{V}_1, \mathbf{V}_2, \dots, \mathbf{V}_M$ as shown in Fig. 2. Each shot of the edited videos will be visually similar to some shot of the parent video. We use the notation $\mathbf{V}_{j_1}^{k_1} \simeq \mathbf{V}_{j_2}^{k_2}$ to indicate that $\mathbf{V}_{j_1}^{k_1}$ is visually similar to $\mathbf{V}_{j_2}^{k_2}$.

Next, we use the notation I_j^k to refer to the *shot identifier* of $\mathbf{V}_j^k$, and I_j to denote the set of all the shot identifiers contained in $\mathbf{V}_j$. These shot identifiers are calculated as follows. First, the shots from all the edited videos are compared to each other exhaustively using a conventional algorithm. Next, a graph is constructed, where each node corresponds to a shot, and edges connect two shots $\mathbf{V}_{j_1}^{k_1}$ and $\mathbf{V}_{j_2}^{k_2}$ where $\mathbf{V}_{j_1}^{k_1} \simeq \mathbf{V}_{j_2}^{k_2}$. Finally, connected components are detected, and an arbitrary integer is assigned to each connected component — these integers are the shot identifiers. Figure 3 illustrates the process of calculating the shot identifiers for the edited videos shown in Fig. 2. In this example, $I_1^1 = I_3^2 = I_2^2 = i_1$, $I_1^2 = I_2^1 = i_2$, $I_1^3 = I_3^1 = i_3$, $I_2^3 = i_4$ and i_1 through to i_4 are different from each other. The color of each node indicates its connected component. The actual value of the shot identifier assigned to each connected component and the order of assigning identifiers to the connected components are irrelevant.

There are many conventional algorithms that can be used to compare shots to each other⁽¹³⁾⁽¹⁴⁾⁽¹⁵⁾. Lu et al. provide a good survey of existing algorithms⁽¹⁶⁾. In this paper, we focus on a robust hashing algorithm that yields a 64-bit hash for each shot⁽¹⁵⁾, since this method has low computational and space complexity. Furthermore, once the hashes have been calculated, access to the original video is not required, enabling the application of the method to Web video sharing portals (e.g. as part of the processing performed when a video is uploaded to the portal). After the videos have been segmented into shots, as shown in the bottom of Fig. 1, the hashing algorithm can be directly applied to each

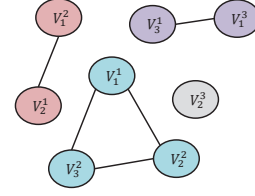


Fig. 3 An example of calculating the shot identifiers for the edited videos in Fig. 2

shot. The calculated hashes enable shots to be compared using the Hamming distance:

$$D(\mathbf{V}_{j_1}^{k_1}, \mathbf{V}_{j_2}^{k_2}) = \sum_{b=1}^{64} h_{j_1}^{k_1}[b] \oplus h_{j_2}^{k_2}[b], \quad (3)$$

where $\mathbf{V}_{j_1}^{k_1}$ and $\mathbf{V}_{j_2}^{k_2}$ are the two shots being compared; $h_{j_1}^{k_1}$ and $h_{j_2}^{k_2}$ are their respective hashes, represented as sequences of 64 bits each; and $\oplus$ is the exclusive OR operator. Finally, thresholding the Hamming distance enables the determination whether two shots are visually similar:

$$\mathbf{V}_{j_1}^{k_1} \simeq \mathbf{V}_{j_2}^{k_2} \iff D(\mathbf{V}_{j_1}^{k_1}, \mathbf{V}_{j_2}^{k_2}) < \tau, \quad (4)$$

where τ is an empirically determined threshold between 1 and 63.

3. The Authenticity Degree

The authenticity degree of an edited video $\mathbf{V}_j$ is defined as the proportion of remaining information from its parent video $\mathbf{V}_0$. It is a positive real number between 0 and 1, with higher values corresponding to high authenticity.

Figure 4 illustrates the strategy of calculating the authenticity degree. Each shot in the parent video $\mathbf{V}_0$ conveys an amount of information. Editing a video causes a loss of some of this information: for example, recompressing the video causes a loss of detail; removing a shot causes a loss of all the information contained in that shot. The edited video $\mathbf{V}_j$ thus contains less of the parent video's information than the parent video itself. The proposed method estimates the amount of information lost and penalizes each edited video accordingly. In order to do this, the proposed method attempts to detect the editing operations that were performed. Each detected editing operation contributes to a penalty, which is calculated individually for each shot. If $\mathbf{V}_j$ and $\mathbf{V}_0$ are identical, then $A(\mathbf{V}_j)$ is at its maximum value of 1, since the penalties become zero. Finally, the penalties are aggregated to calculate the authenticity degree for the entire video.

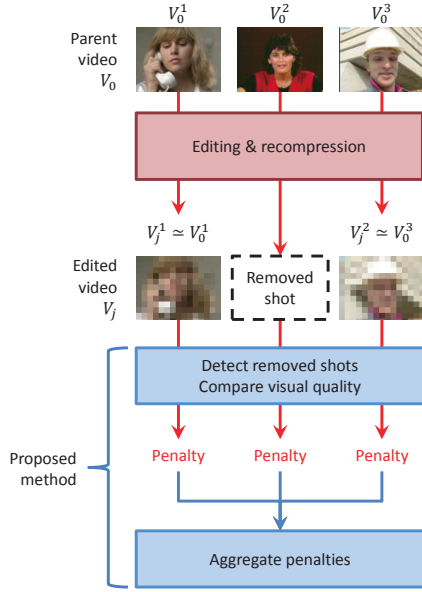


Fig. 4 The strategy of calculating the authenticity degree.

The input to the proposed method is a set of videos $\mathbb{V} = \{\mathbf{V}_1, \dots, \mathbf{V}_M\}$ that are all edited versions of the same parent $\mathbf{V}_0$, which is unknown. In practice, it is possible to obtain $\mathbb{V}$ by searching for a set of keywords describing $\mathbf{V}_0$ on a video sharing portal, for example “Usain Bolt false start” or “Czech president steals pen”. For each edited video $\mathbf{V}_j$ ($j = 1, \dots, M$), the proposed method detects editing operations, determines the corresponding penalties and calculates the authenticity degree as:

$$A(\mathbf{V}_j) = 1 - \frac{\sum_{i \in \hat{I}_0} w_i p_i(\mathbf{V}_j)}{\sum_{i \in \hat{I}_0} w_i}, \quad (5)$$

where $\hat{I}_0$ is an estimate of the set of shot identifiers present in the parent video, described in detail in Sec. 3.1; $p_i(\cdot)$ is an estimate of the penalty function for the shot whose identifier is equal to i ; and w_i is the weighting coefficient of the shot with shot identifier equal to i . The value of w_i should be proportional to the “importance” of the shot: since some shots are more important than others, the loss of their information should have a greater impact on the authenticity degree. The proposed method calculates w_i as the duration of the shot, in seconds. In other words, the proposed method assumes that longer shots contain more information. This assumption is consistent with existing work in video summarization¹⁷⁾.

Specifically, each $p_i(\cdot)$ is calculated as:

$$p_i(\mathbf{V}_j) = \begin{cases} 1 & \text{if } i \notin I_j \\ \gamma g_i(\mathbf{V}_j) & \text{otherwise} \end{cases}, \quad (6)$$

where the notation $i \notin I_j$ means “ $\mathbf{V}_j$ doesn’t contain a shot with an identifier equal to i ”; $g_i(\mathbf{V}_j)$ is the global frame modification penalty, described in detail in Sec. 3.2, for the shot of $\mathbf{V}_j$ whose identifier is equal to i . The parameter γ determines the trade-off between information loss through global frame modifications and information loss through shot removal. Since, in practice, global frame modifications can never remove all the information in a shot entirely, γ should be a positive value less than 1. Equation (6) thus assigns the maximum penalty for shot removal, and a fraction of the maximum penalty for global frame modifications.

There may be significant differences between the edited videos: for example some shots may be added/missing (e.g. Fig. 4). This prevents the application of conventional NRQA algorithms, since the output of such algorithms will be influenced by the added/missing shots. Before NRQA algorithms can be applied effectively, this influence must be reduced. The proposed method achieves this in two ways. First, when calculating the global frame modification penalty, the proposed method works on a shot ID basis, and only compares the raw visual quality of shots that have the same shot identifier (common content). Second, the proposed method prevents missing/added shots from affecting the outcome of NRQA algorithms. More specifically, if a shot is added (not part of the estimated parent $\hat{I}_0$), then it is ignored completely by Eq. (5). If a shot is missing (part of the estimated parent), then a shot removal penalty is assigned by Eq. (6).

3.1 Parent Video Estimation

In order to detect shot removal, the proposed method first defines B_i as the number of edited videos that contain a shot with an identifier equal to i . Then, $\hat{I}_0$, the estimate of the set of shot identifiers present in $\mathbf{V}_0$, is calculated as:

$$\hat{I}_0 = \bigcup_{j=1}^M \{i | i \in I_j, \frac{B_i}{M} > \delta\}, \quad (7)$$

where δ is an empirically determined constant between 0 and 1, and M is the total number of edited videos. Therefore, Eq. (7) selects only those shots that appear in a significant proportion of edited videos. The motivation for this equation is to avoid shots that were inserted into only $\mathbf{V}_j$ from another, less relevant, parent video.

This does not prevent the proposed method from effectively handling edited videos with multiple parents — since the proposed method estimates the *proportion* of remaining information, it does not need to distinguish between the different parent videos, and treats them as a single larger parent.

The obtained $\hat{I}_0$ can be thought of as a “reference”, but this reference is not an existing video — it is calculated by the proposed method through comparison of the available videos, without any access to the parent $\mathbf{V}_0$. Therefore, the proposed method can be thought of as “reduced reference” (as opposed to “full reference” or “no reference”).

The proposed method thus estimates $\hat{I}_0$.

3.2 Global Frame Modification Detection

This section focuses on the detection and penalization of global frame modifications, specifically, scaling and recompression. Scaling a video requires interpolation and recompression. Since both scaling and recompression cause a loss of information through loss of detail, the penalty for scaling and recompression should be proportional to this loss. The proposed method detects scaling and recompression by comparing the visual quality of shots that have the same shot identifier. Based on the visual quality of each shot, the proposed method calculates the global frame modification penalty as:

$$g_i(\mathbf{V}_j) = \begin{cases} 0 & \text{if } Z_j^{k_i} < -1 \\ 1 & \text{if } Z_j^{k_i} > 1 \\ \frac{Z_j^{k_i} + 1}{2} & \text{otherwise} \end{cases}, \quad (8)$$

where k_i is the ordinal number of the shot in $\mathbf{V}_j$ that has the shot identifier equal to i (it satisfies $I_j^{k_i} = i$); and $Z_j^{k_i}$ represents the normalized visual quality degradation strength of the k_i th shot in $\mathbf{V}_j$. Shots with visual quality degradation strength less than one standard deviation from the mean are not penalized, since such weak degradations are typically not noticed by human viewers. A significant loss in quality for a frame thus corresponds to a higher penalty, with the maximum being 1.0. This penalty is visualized in Fig. 5.

In order to calculate $g_i(\mathbf{V}_j^{k_i})$, the proposed method first calculates $Q_j^{k_i}$, the raw visual quality of $\mathbf{V}_j^{k_i}$ as:

$$Q_j^{k_i} = \text{mean}(Q_{NR}(\mathbf{V}_j^{k_i}[l_1]), \dots, Q_{NR}(\mathbf{V}_j^{k_i}[l_L])), \quad (9)$$

where Q_{NR} is an NRQA algorithm that measures edge width⁷⁾; and $l_1, \dots, l_L$ are the frame numbers of the

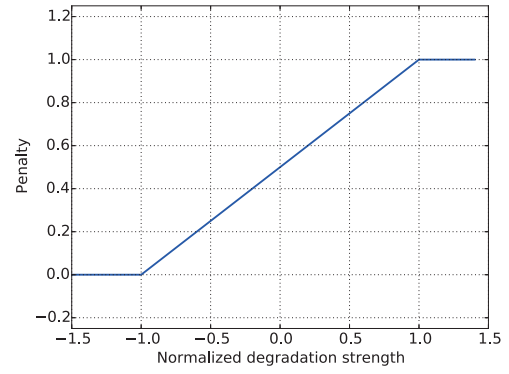


Fig. 5 Global frame modification penalty $p(\cdot)$.

intra-predicted frames (frames that are encoded independently of other frames¹⁸⁾, hereafter referred to as “keyframes”) of the shot. The proposed method focuses only on the keyframes to reduce computational complexity and speed up processing. The output of Q_{NR} corresponds to the strength of visual quality degradations in the frame: a high value indicates low quality, and a low value indicates high quality.

Next, since the strength of detected visual quality degradations depends on the actual content of the frame, the raw visual quality is normalized as:

$$Z_j^{k_i} = \frac{Q_j^{k_i} - \mu_i}{\sigma_i}, \quad (10)$$

where μ_i and σ_i are normalization parameters for the shot whose identifier is equal to i , representing the mean and standard deviation, respectively. The proposed method performs this normalization to fairly penalize shots with different visual content, since visual content is known to affect NRQA algorithms⁷⁾. To illustrate this effect, Fig. 6 shows images of similar visual quality, yet significantly different edge widths: 2.92, 2.90, 3.92 and 4.68 for Figs. 6 (a), (b), (c) and (d), respectively. Without normalization, the images of Figs. 6(c) and (d) would be penalized more heavily than the images of Figs. 6(a) and (b). However, since these images are all uncompressed, they are free of compression degradations, and there is no information loss with respect to the parent video. Therefore, the images should be penalized equally, that is, not at all. Normalization thus reduces the effect of visual content on the NRQA algorithm and achieves a fairer penalty function.

The normalization step above is a novel component of the proposed method. Typically, the output of an NRQA algorithm for a single image is not enough to qualitatively determine its visual quality. In other words, without a reference value, it is impossible to

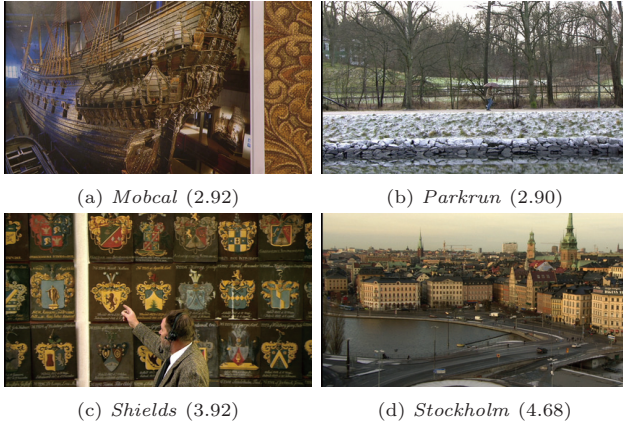


Fig. 6 Images with the same visual quality, but different edge widths (shown in parentheses).

judge whether the visual quality of the image is good or bad. The normalization step provides the proposed method with a reference visual quality level, thus enabling (a) the qualitative assessment of a shot's visual quality; and (b) the comparison of visual quality between shots with different content (see Fig. 6). The effect of normalization is shown empirically in Table 3.

Finally, the result of Eq. (8) is a real number that is lower for higher quality videos. Therefore, it is an effective penalty for recompressed and scaled videos. The proposed method thus detects and penalizes recompression and scaling. While there are other editing operations, such as color transformations, the addition of logos and subtitles, and spatial cropping, the proposed method does not explicitly detect or penalize them. However, applying these editing operations requires recompressing the video, which causes a decrease in visual quality. Since the proposed method explicitly focuses on visual quality degradation, it will penalize the edited videos in proportion to the extent of the visual quality degradation.

4. Experimental Results

This section describes three experiments that were conducted to verify the effectiveness of the proposed method. Section 4.1 evaluates the proposed method in a semi-controlled environment, using a small amount of artificial data that is uploaded to YouTube*, with a known parent. Section 4.2 evaluates the proposed method in a controlled environment, using a large amount of artificial data, with a known parent. Finally, Section 4.3 evaluates the method using real-world data, including a wide range of editing operations, without a known parent.

Table 1 The artificial dataset for the small-scale experiment.

Video	Comments	ER
V_0	Parent video	1
V_1	Reuploaded V_0 to YouTube	2
V_2	Removed 10 frames from each shot of V_1	2
V_3	Reversed order of shots of V_1	4
V_4	Added a shot to V_1	5
V_5	Added a logo to V_1	5
V_6	Downsampled V_0 to 720p	7
V_7	Removed one shot from V_0	8
V_8	Removed two shots from V_0	9
V_9	Removed 60 frames from each shot of V_0	10

4.1 Small-scale Experiment on Artificial Data

This experiment demonstrates the proposed method using a small artificial dataset, which consists of a single short example parent video and several edited copies of it.

(1) Creating the Artificial Data

The single parent video is created by joining the four test videos: *Mobcal*, *Parkrun*, *Shields* and *Stockholm*. The first frames of these four test videos are shown in Figs. 6(a), (b), (c) and (d), respectively. The shot identifiers for these four shots are 1, 2, 3, and 4, respectively. Each test video is approximately 10s long, has 1080p resolution, no audio signal and contains a single shot. The created parent video thus consists of four shots and is 40 seconds long. Next, edited videos were created by editing the parent video by adding logos, adding/removing shots entirely and partially, and reversing the number of shots. The editing operations performed on each video are shown in Table 1. After editing, each video was reuploaded to YouTube. Thus, with the exception of the parent video, each video in the dataset was recompressed more than once. Figure 7 shows screenshots from some selected videos. All the videos are available online**. Finally, each video was automatically preprocessed to detect shot boundaries and calculate shot identifiers as described in Sec. 2.

(2) Evaluation Method

To evaluate the proposed method, this section estimates the authenticity degree of the videos in the dataset. Although the parent video V_0 is part of the dataset, it is not used as any sort of reference, and treated the same as any other video. The main purpose of including it in the dataset was to show that the proposed method is capable of identifying it correctly.

After the authenticity degree for each video has been calculated, the videos are ranked in order of highest

* <http://www.youtube.com>

** <http://tinyurl.com/ok92dg2>

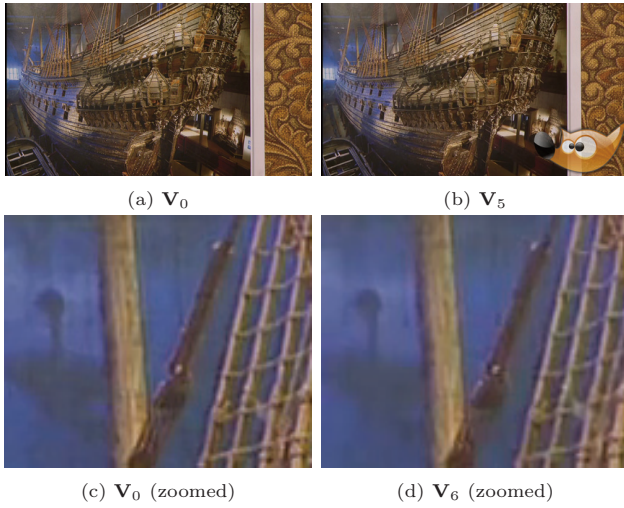


Fig. 7 Frames from the small-scale artificial data set.

Table 2 Results for the small-scale experiment on artificial data. The values shown are direct outputs of Eq. (5). The rank of each video is shown in parentheses. The r row shows the sample correlation coefficient between the ranks and the expected ranks in Table 1.

Video	$\gamma = 0.12$	$\gamma = 0.25$	$\gamma = 0.50$	$\gamma = 0.75$
$\mathbf{V}_0$	0.99 (1)	0.97 (1)	0.94 (1)	0.92 (1)
$\mathbf{V}_1$	0.97 (3)	0.93 (3)	0.87 (3)	0.80 (3)
$\mathbf{V}_2$	0.96 (4)	0.92 (4)	0.84 (4)	0.75 (4)
$\mathbf{V}_3$	0.94 (6)	0.88 (6)	0.77 (6)	0.65 (6)
$\mathbf{V}_4$	0.95 (5)	0.89 (5)	0.78 (5)	0.67 (5)
$\mathbf{V}_5$	0.97 (2)	0.94 (2)	0.88 (2)	0.81 (2)
$\mathbf{V}_6$	0.88 (7)	0.75 (7)	0.50 (8)	0.25 (9)
$\mathbf{V}_7$	0.72 (8)	0.68 (8)	0.61 (7)	0.55 (7)
$\mathbf{V}_8$	0.48 (9)	0.45 (9)	0.40 (9)	0.36 (8)
$\mathbf{V}_9$	0.00 (10)	0.00 (10)	0.00 (10)	0.00 (10)
r	0.88	0.88	0.87	0.85

authenticity degree. These ranks are then compared to the expected ranks, which are determined based on the definition from Sec. 3. More specifically, the authenticity degree of $\mathbf{V}_0$ is 1.0 by definition, since it is the parent video, so its rank must be 1. The remaining expected ranks are determined based on the amount of information removed by the respective editing operation for each video. For example, downsampling removes much more information than simply reuploading, so the rank of $\mathbf{V}_5$ is greater than the rank of $\mathbf{V}_1$. According to the present definition of the authenticity degree, operations such as adding a shot, adding a logo or reversing the order of the shots do not cause a loss of information, but require recompression. Furthermore, the proposed method does not explicitly handle partial shot removal: if a significant part of a shot is removed, then a different shot identifier will be assigned to the shot, and the edited video will be penalized for shot removal, as illustrated by the results for $\mathbf{V}_2$ and $\mathbf{V}_9$. However, since partial shot removal requires recompression of the

video, the proposed method detects this recompression and penalizes the edited video. The expected rank of each video is shown in the “ER” column of Table 1. Finally, the value of δ was empirically set to 0.5 for this experiment.

(3) Discussion of the Results

Table 2 shows the results for various values of γ . The numbers in parentheses indicate the rank of each particular video based on its authenticity degree. The effect of different values of γ can be seen by comparing the authenticity degrees of $\mathbf{V}_6$ and $\mathbf{V}_8$. The latter has higher video quality, but is missing two shots; the former has lower video quality due to downsampling, but is not missing any shots. For lower values of γ , $\mathbf{V}_2$ has the higher authenticity degree, since the penalty for missing shots outweighs the penalty for recompression. For higher values of γ , the situation is reversed. The r row shows the sample correlation coefficient¹⁹⁾ between the expected ranks of Table 1 and the ranks based on the authenticity degrees obtained using each particular value of γ .

Finally, Table 3 demonstrates the effect of normalization on part of the artificial dataset, using two different shots as an example: *Mobcal* and *Stockholm*, shown in Figs. 6(a) and (b), respectively. The “Raw” and “Normalized” columns show the output of Eqs. (9) and (10), respectively. The “Qual.” column shows a qualitative assessment of the visual quality of each video. The mean and standard deviation for the *Mobcal* shot for the entire artificial dataset were 2.98 and 0.22, respectively. Similarly, the mean and standard deviation for the *Stockholm* shot for the entire artificial dataset were 3.94 and 0.22, respectively. From the raw value alone, it is impossible to determine whether the shot quality is good or poor. Furthermore, a “excellent” raw value for one shot may be a “poor” raw value for a different shot (examine values in **bold**). Therefore, the raw score cannot be used to calculate a fair penalty for all shots. On the other hand, the normalized values better correspond to the qualitative assessments and enable a fairer penalty.

In conclusion, the results of this experiment show that: (A) the method is able to correctly detect and penalize shot removal; (B) the method is sensitive to multiple compression and downsampling; (C) the method does not explicitly penalize shot insertion, logo insertion, modification of shot order, or partial shot removal, but detects and penalizes the side-effect of recompression.

Table 3 The effect of normalization on the small artificial dataset, illustrated by example shots and videos. The “Raw” and “Normalized” columns show the output of Eqs. (9) and (10), respectively. The “Qual.” column shows a qualitative assessment of the visual quality of each video.

Video	Raw		Normalized		Qual.
	<i>Mobcal</i>	<i>Stockholm</i>	<i>Mobcal</i>	<i>Stockholm</i>	
V₀	2.82	3.75	-0.68	-0.89	Excellent
V₁	2.88	3.84	-0.42	-0.45	Good
V₆	3.62	4.54	2.80	2.71	Poor

Table 4 A summary of the parent videos used in the large-scale artificial data experiment.

Dataset	Resolution	Duration	Description
<i>Argo</i>	1920 × 800	2 min 32s	Movie trailer
<i>Bionic</i>	1920 × 1080	2 min 19s	Charity
<i>Budweiser</i>	1920 × 1080	1 min 0s	Product
<i>Daylight</i>	1280 × 720	3 min 16s	Comedy
<i>Drifting</i>	1920 × 1080	3 min 25s	Sport
<i>Ducks</i>	1920 × 1080	1 min 14s	Cartoon
<i>Echo</i>	1280 × 720	2 min 6s	Product
<i>Entourage</i>	1920 × 1080	2 min 27s	Movie trailer
<i>Frozen</i>	1920 × 852	0 min 39s	Movie trailer
<i>Hummingbird</i>	1920 × 1080	3 min 25s	Documentary
<i>Jimmy</i>	1920 × 1080	2 min 21s	Comedy
<i>Lebron</i>	1280 × 720	1 min 21s	Sport
<i>Nfl</i>	1280 × 720	4 min 8s	Sport
<i>Ridge</i>	1920 × 1080	7 min 38s	Nature
<i>Tie</i>	1280 × 720	7 min 27s	Cartoon
<i>Tricks</i>	1920 × 1080	5 min 40s	Sport

4.2 Large-scale Experiment on Artificial Data

This experiment demonstrates the proposed method using a wider range of data than the previous section. Since collecting a large volume of near-duplicate videos from the Web is difficult, this experiment utilizes artificial data.

(1) Creating the Artificial Data

The experiment utilizes 16 datasets. Each dataset consists of a single parent video and edited copies of the parent video. Table 4 shows a summary of the parent videos, which were all acquired from YouTube by browsing the *#PopularOnYouTube* video channel. A YouTube playlist with links to each of the parent videos is available online*.

For each dataset, the edited copies were created by editing the parent video with a range of editing operations: downsampling, shot removal, single compression, and double compression. More specifically, the downsampling operation reduced the resolution of the video to the common resolutions used by YouTube. The shot removal operation selected a percentage of shots, at random, and removed them from the video entirely. The compression operation utilized H.264, one of the codecs

Table 5 Editing operations used to generate artificial data.

Operation	Parameter type	Parameter value
Downsampling	Resolution	720p, 480p, 360p
H.264 compression	CRF	18, 26, 34, 40
Shot removal	Percentage	10%, 20%, ..., 90%

used by YouTube, to compress the video using a range of Constant Rate Factor (CRF) values. Finally, double compression was also performed with H.264, using a CRF of 18 on both iterations. Each dataset thus consisted of 17 videos (the parent was not included in the dataset). A summary of the editing operations and the parameters used are shown in Table 5.

(2) Evaluation Method

In contrast to the previous experiment, the amount of data and range of editing operations is much greater. Therefore, it is not possible to objectively rank each of the videos. Instead, we relied on subjective evaluations for each video. The subjects were asked to rate each video on 3 points: (A) relative visual quality; (B) number of deleted shots; and (C) estimated proportion of information remaining from the parent. The subjects were asked to rate (A) and (B) independently, and to rate (C) based on their answers to (A) and (B). All answers were given using a scale of 1 to 5, since this scale is commonly used for visual quality assessment**. For each video, the mean answer to (C) across all subjects was recorded as that particular video’s estimated proportion of information remaining from the parent. The parent video itself was not shown to the test subjects. A total of 12 subjects participated in the experiment. Next, in order to evaluate the proposed method, the method was used to estimate the authenticity degree of each video. Similar to the previous section, the sample correlation coefficient and rank-order correlation coefficient¹⁹⁾ between the obtained authenticity degrees and the subjective estimates were also calculated.

(3) Discussion of the Results

Table 6 shows the results of the experiment. For the comparative method, the experiment used the mean edge width for the entire video, with the output scaled such that positive values correspond to high quality. For the proposed method, the γ parameter was empirically set to 0.25. The sample correlation coefficient and the rank-order correlation coefficient are shown in the r and ρ columns, respectively, for each method. The higher value for each dataset is shown in bold. These results

* <http://tinyurl.com/o7u2mfz>

** See ITU-T Recommendation BT.500: “Methodology for the subjective assessment of the quality of television pictures”

Table 6 Results for the experiment on large-scale artificial data. r and ρ show the sample correlation coefficient and the rank order correlation coefficient, respectively.

Dataset	Comparative		Proposed	
	r	ρ	r	ρ
<i>Argo</i>	-0.08	0.17	0.82	0.84
<i>Bionic</i>	-0.30	-0.01	0.93	0.90
<i>Budweiser</i>	-0.37	-0.48	0.92	0.86
<i>Daylight</i>	-0.04	0.08	0.84	0.88
<i>Drifting</i>	-0.19	-0.06	0.78	0.69
<i>Ducks</i>	0.12	0.15	0.96	0.93
<i>Echo</i>	0.21	0.38	0.85	0.83
<i>Entourage</i>	-0.41	-0.41	0.90	0.88
<i>Frozen</i>	-0.16	-0.02	0.81	0.77
<i>Hummingbird</i>	-0.36	-0.60	0.86	0.88
<i>Jimmy</i>	-0.57	-0.73	0.76	0.68
<i>Lebron</i>	-0.01	-0.01	0.88	0.82
<i>Nfl</i>	-0.20	0.06	0.97	0.95
<i>Ridge</i>	-0.44	-0.54	0.92	0.91
<i>Tie</i>	0.11	0.42	0.95	0.88
<i>Tricks</i>	-0.38	-0.53	0.91	0.85

Table 7 Summary of real datasets.

Name	Videos	Total dur.	Shots	IDs	ASL	AKI
<i>Bolt</i>	68	4 h 42 min	1933	275	7.99	1.87
<i>Kerry</i>	5	0 h 47 min	103	24	25.23	1.88
<i>Klaus</i>	76	1 h 16 min	253	61	21.47	1.49
<i>Lagos</i>	8	0 h 6 min	79	17	4.58	1.97
<i>Russell</i>	18	2 h 50 min	1748	103	5.80	1.94
Total	175	9 h 41 min	4116	480	13.01	1.83

show that the proposed method significantly outperforms the comparative method in the majority of cases.

4.3 Experiment on Real Data

This experiment utilized real, near-duplicate videos obtained from YouTube. The parent videos for each dataset are unknown. In order to evaluate the proposed method, the same subjective evaluation method as in the previous section was used. A total of 20 subjects participated in the experiment.

(1) Description of the Real Data

Table 7 shows the datasets used in the experiment, where each dataset consists of several edited videos collected from YouTube. The “Shots” and “IDs” columns show the total number of shots and unique shot identifiers, respectively. The “ASL” and “AKI” columns show the average shot length and average keyframe interval (time between keyframes), respectively, in seconds. Since the average keyframe interval is significantly smaller than the average shot length for all datasets, the majority of shots are represented by at least one keyframe. This validates the optimization strategy employed by Eq. (9). Finally, each video was automatically preprocessed to detect shot boundaries and calculate shot identifiers as described in Sec. 2.

(2) Comparative Methods

For comparative methods, this experiment uses:

- (A) The number of views of each video,
- (B) The upload timestamp for each video,
- (C) The mean visual quality of each video, and
- (D) The proposed method with $\gamma = 0$ (consider shot removal only).

The number of views and upload timestamp are available from the metadata of each video. The mean visual quality was obtained by calculating the mean edge width⁷⁾ for each individual frame. Finally, the output of all methods was configured such that positive values correspond to higher authenticity. Therefore, high correlation coefficients correspond to the best methods.

(3) Automatic Shot Segmentation

The parameter v determines the shot boundary detection threshold. It is applied to the result of Eq. (2). For this experiment, its value was empirically set to 5.0. Since the actual shot boundaries are known, it is possible to evaluate the shot boundary detector in terms of precision, recall and F_1 :

$$P = \frac{TP}{TP + FP}, \quad (11)$$

$$R = \frac{TP}{TP + FN}, \quad (12)$$

and

$$F_1 = 2 \times \frac{P \times R}{P + R}, \quad (13)$$

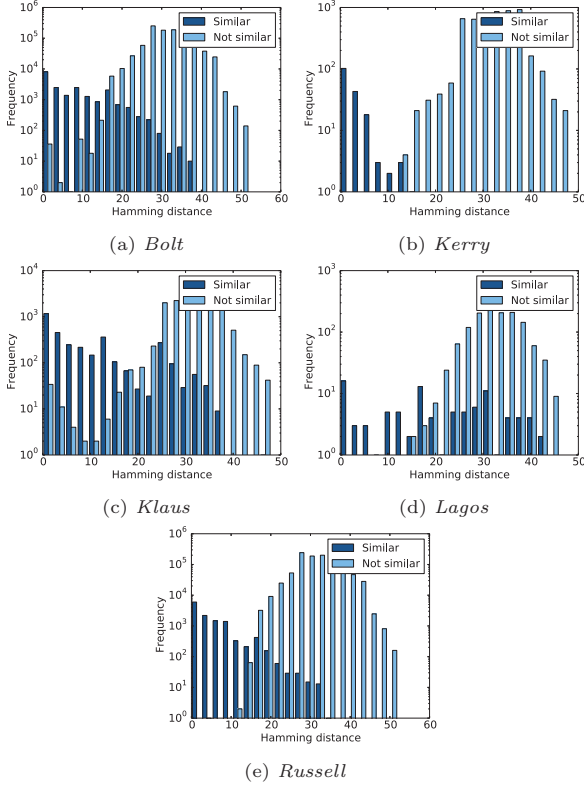
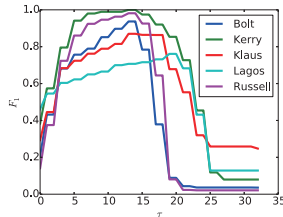
respectively, where TP is the number of true positives: actual boundaries that were detected by the automatic algorithm; FP is the number of false positives: boundaries that were detected by the automatic algorithm but that are not actual shot boundaries; and FN is the number of false negatives: actual boundaries that were not detected by the automatic algorithm. The mean precision, recall and F_1 score for each dataset is shown in Table 8. These results show that while the precision of the algorithm is relatively high for all datasets, the recall depends on the data. Through manual examination of the results, the main cause of false negatives (low recall) was identified as “fade” or “dissolve” shot transitions. The *Bolt* and *Lagos* datasets contain many such transitions, leading to the relatively low recall for those datasets.

(4) Automatic Shot Identification

The parameter τ of Eq. (4) determines the Hamming distance threshold for visual similarity. In order to investigate the significance of τ , we exhaustively compared the Hamming distances between all shots in each

Table 8 Accuracy of the automatic shot boundary detector ($v = 5.0$).

Dataset	Precision	Recall	F_1 score
<i>Bolt</i>	0.81	0.65	0.72
<i>Klaus</i>	0.88	0.77	0.81
<i>Kerry</i>	0.91	0.81	0.86
<i>Lagos</i>	0.92	0.69	0.79
<i>Russell</i>	0.98	0.89	0.93


Fig. 8 A histogram of Hamming distances for the datasets in Table 7.

Fig. 9 F_1 as a function of τ for the datasets in Table 7.

dataset and plotted their histograms. The histograms are shown in Fig. 8, with frequency on a logarithmic scale. Two histograms are shown for each dataset: (i) the “similar” histogram corresponds to Hamming distances between shots that are visually similar; and (ii) the “not similar” histogram corresponds to Hamming distances between shots that are not visually similar. Figure 8 shows that as the Hamming distance increases, the proportion of comparisons between visually similar shots decreases, and the proportion of comparisons between visually dissimilar shots increases. However, it is

obvious that picking a value for τ that allows all shots to be identified correctly is impossible, since the two distributions always overlap.

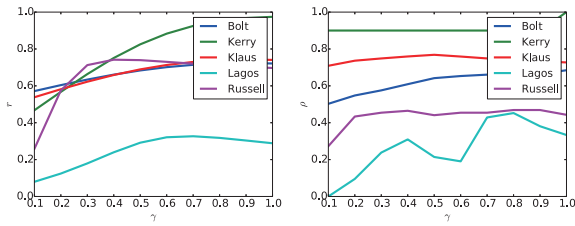
In order to quantitatively judge the accuracy of the automatic method for calculating shot identifiers, we conducted an experiment using the query-response paradigm, where the query is a shot, and the response is the set of all shots that have the same shot identifier, as calculated by the automatic method. Using the manually-assigned shot identifiers as ground truth, we then measured the precision, recall and F_1 score as Eqs. (11), (12), and (13). In this case, true positives are shots in the response that have the same manually-assigned shot identifier as the query shot; false positives are shots in the response that have different manually-assigned shot identifiers to the query shot; and false negatives are shots that were not part of the response, but that have the same manually-assigned shot identifiers as the query shot. The average values across all shots for each dataset were then calculated.

Finally, Fig. 9 shows the F_1 score for each dataset for different values of τ . From this figure, we set the value of τ to 14 in the remainder of the experiment, as that value achieves the highest average F_1 score across all the datasets. Table 9 shows the obtained results for that value of τ . From this table, it is obvious that some datasets are easier to calculate shot identifiers for than others: for example, *Kerry*; and some are more difficult: for example, *Lagos*. More specifically, in the *Lagos* case, the recall is particularly low. This is because the dataset consists of videos that have been edited significantly, for example, by inserting logos, captions and subtitles. In such cases, the hashing algorithm used is not robust enough, and yields significantly different hashes for shots that a human would judge to be visually similar. Nevertheless, Table 9 shows that automatic shot identifier calculation is possible.

(5) Parameter Selection

The parameter γ of Eq. (6) determines the trade-off between penalizing shot removal and penalizing global frame modifications. Figure 10 shows the effect of different values of γ on the results for each of the each of the datasets in Table 7. From the figure, it is obvious that the optimal value of γ is different for each dataset. We set $\gamma = 0.7$ in the remainder of the experiments.

The value for the parameter δ of Eq. (7), which determines what proportion of edited videos must contain a particular shot identifier in order for that identifier to be part of the parent video, was empirically set to 0.1.



(a) Sample correlation (b) Rank-order correlation
Fig. 10 Effect of different values of γ on the results for each of the datasets in Table 7.

Table 9 Accuracy of the automatic shot identifier function.

Dataset	Precision	Recall	F_1 score
<i>Bolt</i>	0.99	0.92	0.94
<i>Kerry</i>	1.00	1.00	1.00
<i>Klaus</i>	1.00	0.84	0.87
<i>Lagos</i>	1.00	0.62	0.71
<i>Russell</i>	1.00	0.98	0.98

(6) Discussion of the Results

Tables 10 and 11 show the sample correlation coefficient r and rank-order correlation coefficient ρ , respectively. Each line corresponds to one of the datasets in Table 7. The number of asterisks indicates the rank of the particular method within a specific dataset: $\star\star\star$, $\star\star$ and $\star$ indicate the best, second best and third best methods, respectively. From the tables, the following observations can be made:

- The correlation coefficients for the number of views and upload timestamps (Comp. A and B) vary significantly between the datasets. This demonstrates that the number of views and upload timestamps are not always reliable for estimating the authenticity of Web video.
- The correlation coefficients for the mean visual quality of each video (Comp. C) are moderate to high for all datasets. This demonstrates that visual quality is relevant and effective for estimating the authenticity of Web video.
- The correlation coefficients for Comp. D are high for the *Lagos* and *Russell* datasets, and moderate for the others. This demonstrates that shot removal is relevant to video authenticity. Furthermore, it shows that the optimal value of the constant γ differs between various datasets.
- The correlation coefficients for the proposed method are high for most of the datasets.

In summary, Tables 10 and 11 show that the results of the proposed method correlate with the subjective estimates better than any of the comparative methods. More specifically, the proposed method signifi-

Table 10 Sample correlation coefficients (r) for the experiment on real data.

	Comp. A	Comp. B	Comp. C	Comp. D	Prop.
<i>Bolt</i>	0.11	0.31	$\star\star$ 0.61	$\star$ 0.54	$\star\star\star$ 0.74
<i>Kerry</i>	$\star\star$ 0.79	0.19	$\star$ 0.54	0.37	$\star\star\star$ 0.97
<i>Klaus</i>	0.27	0.08	$\star\star$ 0.47	$\star$ 0.42	$\star\star\star$ 0.68
<i>Lagos</i>	0.05	-0.51	$\star\star$ 0.37	$\star$ 0.35	$\star\star\star$ 0.43
<i>Russell</i>	-0.13	$\star$ 0.14	$\star\star\star$ 0.80	-0.04	$\star\star$ 0.72

Table 11 Rank order correlation coefficients (ρ) for the experiment on real data.

	Comp. A	Comp. B	Comp. C	Comp. D	Prop.
<i>Bolt</i>	0.27	0.26	$\star\star$ 0.67	$\star$ 0.45	$\star\star\star$ 0.69
<i>Kerry</i>	$\star\star$ 0.70	0.20	$\star$ 0.40	0.35	$\star\star\star$ 1.00
<i>Klaus</i>	0.08	0.10	$\star$ 0.46	$\star\star$ 0.55	$\star\star\star$ 0.67
<i>Lagos</i>	-0.12	-0.38	$\star$ 0.26	$\star\star$ 0.27	$\star\star\star$ 0.43
<i>Russell</i>	0.16	$\star$ 0.33	$\star\star\star$ 0.55	-0.18	$\star\star$ 0.47

Table 12 Correlation coefficients of the average evaluation score to the individual evaluation score vs. the proposed method.

	Subjective		Proposed	
Dataset	$\bar{r}$	$\bar{\rho}$	r	ρ
<i>Bolt</i>	0.66	0.64	0.74	0.69
<i>Kerry</i>	0.77	0.77	0.97	1.00
<i>Klaus</i>	0.74	0.74	0.68	0.67
<i>Lagos</i>	0.71	0.72	0.43	0.43
<i>Russell</i>	0.74	0.54	0.72	0.47

cantly outperforms comparative method C: 28 vs. 18 stars. Therefore, the proposed method realizes more effective authenticity degree estimation than the comparative methods.

Finally, Table 12 shows the correlation coefficients of the average evaluation score to the individual evaluation score ($\bar{r}$ and $\bar{\rho}$), and compares them to the proposed method. More specifically, $\bar{r}$ for each dataset was calculated as follows: first, calculate the mean subjective visual quality for each video in the dataset; then, calculate the sample correlation coefficient of each individual subject's response with the mean subjective visual quality, yielding a total of 20 coefficients; and finally calculate the mean of the 20 coefficients. The value for $\bar{\rho}$ was calculated in a similar fashion, replacing the sample correlation coefficient with the rank order correlation coefficient. The coefficients for the proposed method are taken directly from Tables 10 and 11. This table shows that while the correlation coefficients obtained by the proposed method are not very high, they are comparable to the average performance of a human viewer.

5. Conclusion

This paper proposed a new method for estimating the amount of information remaining from the parent video: the authenticity degree. The novelty of this pro-

posed method consists of four parts: (a) in the absence of the parent video, we estimate its content by comparing edited copies of it; (b) we reduce the difference between the video signals of edited videos before performing a visual quality comparison, thereby enabling the application of conventional NRQA algorithms; (c) we collaboratively utilize the outputs of the NRQA algorithms to determine the visual quality of the parent video; and (d) we enable the comparison of NRQA algorithm outputs for visually different shots. To the best of our knowledge, we are the first to apply such algorithms to videos that have significantly different signals, and the first to utilize video quality to determine the authenticity of a Web video. Experimental results have shown that while conventional NRQA algorithms are closely related to a video's authenticity, they are insufficient where the video signals are significantly different due to editing. Furthermore, the results have shown that the proposed method outperforms conventional NRQA algorithms and other comparative methods. Finally, the results show that the effectiveness of the proposed method is similar to that of an average human viewer. The proposed method has applications in video retrieval: for example, it can be used to sort search results by their authenticity.

The proposed method has several limitations. First, partial shot removal is not penalized: as long as the shot identifiers match, the video containing that shot is not penalized. Finally, the order of the shots is not considered. Overcoming these limitations is the subject of future work.

References

- 1) R. D. Oliveira, M. Cherubini, and N. Oliver, "Looking at near-duplicate videos from a human-centric perspective," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 6, no. 3, 2010.
- 2) I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, "Digital Watermarking and Steganography," Nov. 2007.
- 3) S. Milani, M. Fontani, P. Bestagini, M. Barni, A. Piva, M. Tagliasacchi, and S. Tubaro, "An overview on video forensics," *APSIPA Transactions on Signal and Information Processing*, vol. 1, p. e2, Aug. 2012.
- 4) Z. Dias, A. Rocha, and S. Goldenstein, "Video Phylogeny: Recovering near-duplicate video relationships," in *2011 IEEE International Workshop on Information Forensics and Security*. IEEE, Nov. 2011, pp. 1–6.
- 5) S. Lameri, P. Bestagini, A. Melloni, S. Milani, A. Rocha, M. Tagliasacchi, and S. Tubaro, "Who is my parent? Reconstructing video sequences from partially matching shots," in *IEEE International Conference on Image Processing (ICIP)*, 2014.
- 6) A. K. Moorthy and A. C. Bovik, "Visual quality assessment algorithms: what does the future hold?" *Multimedia Tools Appl.*, vol. 51, no. 2, pp. 675–696, Jan. 2011.
- 7) P. Marziliano, F. Dufaux, S. Winkler, and T. Ebrahimi, "A no-reference perceptual blur metric," in *Proceedings of the International Conference on Image Processing. 2002. Proceedings. 2002 International Conference on*, vol. 3, Genimedia SA, Lausanne, Switzerland. Toronto, Ont., Canada: IEEE, 2002, Conference proceedings (article), pp. III–57–III–60 vol.3.

- 8) Z. Wang and A. C. Bovik, "Modern Image Quality Assessment," *Synthesis Lectures on Image, Video, and Multimedia Processing*, vol. 2, no. 1, pp. 1–156, Jan. 2006.
- 9) Q. Huynh-Thu and M. Ghanbari, "Scope of validity of PSNR in image/video quality assessment," *Electronics Letters*, vol. 44, no. 13, p. 800, 2008.
- 10) M. Penkov, T. Ogawa, and M. Haseyama, "A Method for Estimating the Authenticity Degree of Web Videos and Its Evaluation," in *International Workshop on Advanced Image Technology (IWAIT)*, 2013, pp. 534–538.
- 11) A. Nagasaka and Y. Tanaka, "Automatic Video Indexing and Full-Video Search for Object Appearances." North-Holland Publishing Co., 1992, Conference proceedings (article), pp. 113–127.
- 12) R. Lienhart, "Comparison of automatic shot boundary detection algorithms," pp. 290–301, 1999.
- 13) C. Kim and B. Vasudev, "Spatiotemporal sequence matching for efficient video copy detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, 2005.
- 14) S.-c. Cheung and A. Zakhor, "Fast similarity search and clustering of video sequences on the world-wide-web," *IEEE Transactions on Multimedia*, vol. 7, no. 3, pp. 524–537, Jun. 2005.
- 15) B. Coskun, B. Sankur, and N. Memon, "Spatio-Temporal Transform Based Video Hashing," *IEEE Transactions on Multimedia*, vol. 8, no. 6, pp. 1190–1208, Dec. 2006.
- 16) J. Lu, "Video fingerprinting for copy identification: from research to industry applications," in *IS&T/SPIE Electronic Imaging*, E. J. Delp III, J. Dittmann, N. D. Memon, and P. W. Wong, Eds., Feb. 2009, pp. 725 402–725 402–15.
- 17) S. Uchihashi and J. Foote, "Summarizing video using a shot importance measure and a frame-packing algorithm," in *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No.99CH36258)*, vol. 6. IEEE, 1999, pp. 3041–3044 vol.6.
- 18) I. Richardson, *The H.264 Advanced Video Compression Standard*. John Wiley & Sons, 2010.
- 19) Y. Dodge, *The Concise Encyclopedia of Statistics*. New York, NY: Springer New York, 2008.



Michael Penkov received his B.E. degree in Software Engineering from the University of New South Wales, Sydney, Australia, in 2004. He received his M.S. degree in Electronics and Information Engineering from Hokkaido University, Sapporo, Japan, in 2012 and is currently pursuing his Ph.D. at the same university.



Takahiro Ogawa received his B.S., M.S. and Ph.D. degrees in Electronics and Information Engineering from Hokkaido University, Japan in 2003, 2005 and 2007, respectively. He is currently an assistant professor in the Graduate School of Information Science and Technology, Hokkaido University. His research interests are digital image processing and its applications. He is a member of the IEEE, EURASIP, IEICE, and Institute of Image Information and Television Engineers (ITE).



Miki Haseyama received her B.S., M.S. and Ph.D. degrees in Electronics from Hokkaido University, Japan in 1986, 1988 and 1993, respectively. She joined the Graduate School of Information Science and Technology, Hokkaido University as an associate professor in 1994. She was a visiting associate professor of Washington University, USA from 2005 to 2006. She is currently a professor in the Graduate School of Information Science and Technology, Hokkaido University. Her research interests include image and video processing and its development into semantic analysis. She is a member of the IEEE, IEICE, Institute of Image Information and Television Engineers (ITE) and Acoustical Society of Japan (ASJ).