Office of Graduate Studies

Dissertation / Thesis Approval Form

This form is for use by all doctoral and master's students with a dissertation/thesis requirement. Please print clearly as the library will bind a copy of this form with each copy of the dissertation/thesis. All doctoral dissertations must conform to university format requirements, which is the responsibility of the student and supervising professor. Students should obtain a copy of the Thesis Manual located on the library website.

Dissertation/Thesis Title: New Methods for Detecting Frame Deletion in Modern Video

Author: Hunter Kippen

This dissertation/thesis is hereby accepted and approved.

Signatures:

Examining Committee

Chair ____________________

Members ____________________

____

____

____

____

Academic Advisor ____________________

Department Head ____________________

New Methods for Detecting Frame Deletion in Modern Video

A Thesis

Submitted to the Faculty

of

Drexel University

by

Hunter Kippen

in partial fulfillment of the

requirements for the degree

of

Master of Science

May 2019

Dedications

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

This is the second paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

And after the second paragraph follows the third paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Acknowledgments

After this fourth paragraph, we start a new paragraph sequence. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

This is the second paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Table of Contents

List of Tables

List of Figures

Abstract

New Methods for Detecting Frame Deletion in Modern Video
Hunter Kippen
Dr. Matthew Stamm, Ph.D.

And after the second paragraph follows the third paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

After this fourth paragraph, we start a new paragraph sequence. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Chapter 1: Background

1.1 Video Encoding

Computer storage and processing power is often obtained at a premium. While this is less true today, as reliable high-capacity Hard Disk and Solid-state Drives have become widely available, high definition raw video footage still places a large constraint on these high-capacity drives. They can quickly become filled with only a few minutes of content. For example, a 1080p video has a frame resolution of 1920x1080, meaning a single frame consists of 2,073,600 pixels. Each pixel is represented by three values, one for red, one for blue, and one for green. For standard 8-bit color, a single pixel requires 24 bits of storage. Thus, a single frame requires 49,766,400 bits or about 6 Megabytes of storage. Videos are often played at 24, 30, or even 60 frames per second (fps). A raw 1080p video at 24 fps requires 142.4 Megabytes of storage per second of content. A 1 Terabyte hard drive could store only 122 minutes of content, or a little over 2 hours. If the video was 30 or 60 fps, then the maximum capacity of the hard drive would be even less.

Thus there is a need to be able to compress video files such that they retain the same or similar visual quality while being tens or hundreds of times smaller, depending on the use case. The information stored in video files has a large amount redundancy. This redundancy is primarily along two axes, spatial and temporal. Video compression standards (or codecs) smartly exploit these redundancies to produce highly compressed video files. Two popular video codecs are MPEG-2 [1], used in DVDs and H.264 [2], used for encoding high definition video.

Spatial redundancy in video is similar to that of still images. Most of the information in the visual domain in an image is encoded in low frequency components. One can reduce the amount of information in the high frequency regions of an image and incur very little perceptual loss while reducing the overall size of the image. Joint Photography Experts Group (JPEG) encoding is a widely used compression standard for images that makes use of this particular phenomenon. A simple way to encode video is to treat each video frame as a still image, and encode it using a

JPEG-like process. One of the first video encoding standards, Motion JPEG, did this. However, this method of video compression does not reach the levels of compression when exploiting both spacial and temporal redundancy.

Temporal redundancy is exploiting the fact that in most cases, one video frame does not change too much from the previous frame or frames. Often portions of a video will remain completely static, while certain sections move (think a traffic camera). As such, for a sufficiently small time interval, the current frame can be predicted from the previous frame. Obviously, the current frame is not an exact match for the previous frame. Thus, the difference between the two frames, referred to as the prediction error residual, is stored alongside the previous frame. As shown in figure 1.1, an error residual for a small amount of motion has an extremely low dynamic range. Most values are black, or close to it. It is possible then, to also highly compress the residual frame using a JPEG-like process to get even more space savings without losing much in the way of prediction accuracy.

Figure 1.1: Visualization of a Video Prediction Error Residual

MPEG-2 takes both of these concepts a step further. Both Temporal and Spatial redundancy are exploited in the frame prediction process. Successive video frames are first grouped together to minimize prediction error. The first frame in the group is encoded entirely using a JPEG-like process. This frame is called an intra-coded frame or I-frame. Following this I-frame is a sequence of predicted frames. Predicted frames come in two types. Regular predicted frames or P-frames derive their predictions from past frames. Bi-directional frames, or B-frames can mix predictions from both past and future frames. These groupings of frames are called groups of pictures (GOPs).

An example GOP sequence can be seen in Fig. 1.2. In most modern codecs, including MPEG-2,

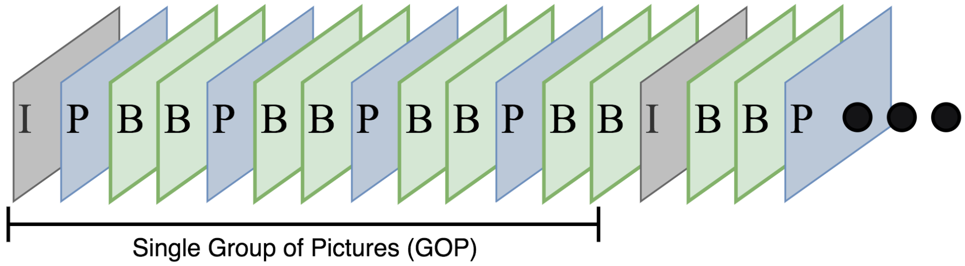


Figure 1.2: Example of a GOP Sequence

these GOP structures do not have to be fixed in size. The GOP length is determined by the distance between subsequent I-frames. Often, it is more efficient to make the GOP smaller during segments of high motion in a video. This is known as variable GOP encoding, versus fixed GOP encoding where the GOP sequence is always the same length.

Predictions can only be derived from I or P-frames, sometimes referred to as anchor frames. In MPEG-2, P-frames can only derive their predictions from a single previous anchor frame, and B-frames can only derive their predictions from a single previous and a single future anchor frame [1]. To decrease the prediction error, predicted frames are partitioned into 16x16 pixel regions called macroblocks. These macroblocks are compared with blocks in the previous anchor frame and/or the next anchor frame in the case of B-frames. A fast search algorithm is performed to find a macroblock sized region with the smallest error when subtracted from the macroblock of interest.

The pixel displacement in both the x and y directions from the centroids of the macroblocks is stored in what is known as a motion vector. Then the error residual between the two macroblocks mapped by the motion vector is also stored. In this way, predicted frames are transmitted as a set of motion vectors and macroblock sized prediction error residuals. Decoding a predicted frame requires using the motion vectors to obtain the pixel values of the source macroblock for each macroblock in the frame. Then the error residual for each macroblock is added back, thus reproducing the original picture. The entire prediction error and motion vector generation process is known as motion compensation and estimation.

The H.264 specification contains many advancements in the motion compensation and estimation process [2] [3]. Macroblocks no longer have to be 16x16 pixel blocks, and instead can be a variety of shapes including 8x16, 16x8, and 8x8. Single macroblocks can be encoded like an I-frame, instead of being predicted. These I-blocks are useful in scenes with high motion. Also, macroblocks can derive predictions from other portions of their frame. Intra-frame predictions are useful for large, mostly single colored regions like the sky. In this way, only one inter-frame motion vector has to be produced for an entire section of the image.

It is also possible for the encoder to entirely skip a prediction for a given macroblock. Instead, the macroblock is directly copied from a previously decoded frame. These are known as skip macroblocks, or S-blocks.

The most notable improvement from H.264 over MPEG-2 is the ability for the encoder to have a substantially frame buffer. This allows for macroblocks in predicted frames to derive the motion vectors from across multiple different anchor frames. Effectively, a B-frame could be composed of macroblocks sourced from all anchor frames in the encoder's frame buffer. This substantially reduces the prediction error of any given macroblock. Fig. 1.3 shows an example of this. The first B-frame can source macroblocks from any I and P-frame in the encoder's frame buffer.

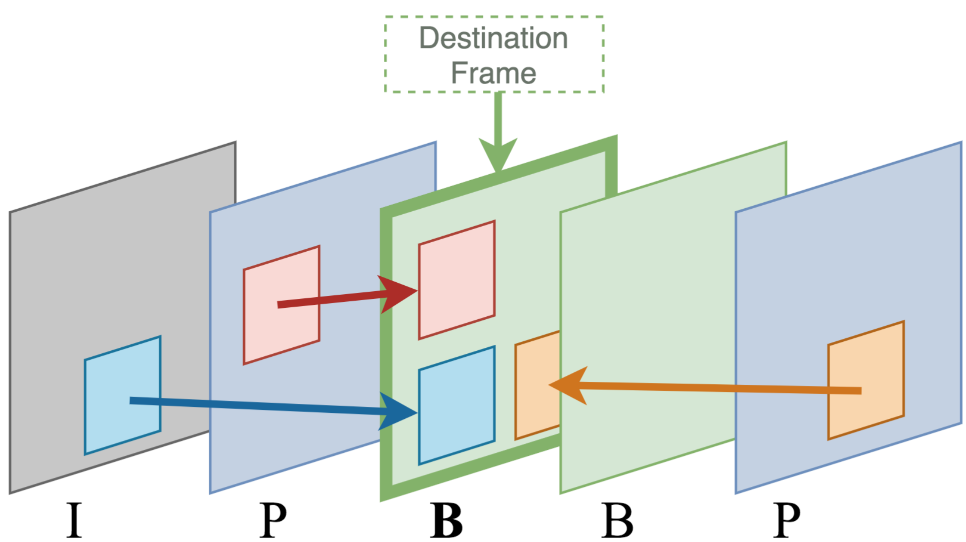


Figure 1.3: Visualization of Using Multiple Anchor Frames for Prediction in H.264

1.2 Frame Deletion Detection

1.3 Autoregressive Models

1.4 Support Vector Machines

Chapter 2: Problem Formulation

Detecting frame deletion in a video requires detecting the structural changes in a video due to the deletion process. In particular, Wang and Farid's work on temporal traces for detecting frame deletion shows that for MPEG-2 video, the P-frame prediction error can be formulated into a sequence. This sequence can then be monitored to detect frame deletion. Both Wang and Farid, and Stamm et al. use a system like in Fig. 2.1 to detect frame deletion. The prediction error sequence $e(n)$ is extracted from the decoded video file and processed to produce detection features. Wang and Farid's work did not propose features for automatic detection, and instead relied on visual inspection of the DFT of the prediction error sequence [4] [5].

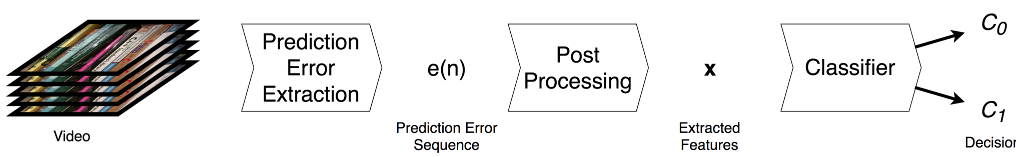


Figure 2.1: Generalized Approach to Frame Deletion Detection

This broad approach can also be applied to work with H.264 encoded video as well. While video encoding has advanced significantly, the fundamental structures of a compressed digital video have remained unchanged. Regardless of codec, the hallmark of video compression is the motion compensation and estimation process. Video frames are organized into GOPs that begin with an I-frame, and have varying structures of P and B-frames. In H.264, GOP structures are more dynamic due to the ability to derive motion-vector predictions from across multiple anchor frames. To remain robust to these advances in video compression, the prediction error extraction and post processing steps must be altered or augmented.

This work is concerned particularly with the detection of frame deletion in H.264 and similar modern video codecs. Frame addition has also been observed to introduce similar traces in the P-frame prediction error sequence as frame deletion. Our proposed system can be applied to detecting frame addition and for simplicity we will not discuss the detection of frame addition for the remainder of this thesis. We have made the following assumptions regarding our proposed system. First, we assume that all altered video has undergone re-compression. In fact, since most consumer video recording devices do not have the storage capability or processing power to record high-definition raw video, it is assumed that all video sources have been compressed by either MPEG-4 or H.264, and that all frame deleted video will be re-compressed using H.264 or a similar codec, where the reencoding is set to match the GOP structure of the source video.

In addition, it is assumed that all videos that are passed to the detector are of sufficient length to make a classification. Without multiple full GOPs, the presence of a deletion fingerprint is negligible. Lastly, we make the assumption that if indeed frames have been removed from a video, they have not been removed from the end of the video. The detection features are dependent on differences between the structure of the prediction error sequences in natural videos versus videos with frame deletion. When frames are removed from the end of the video sequence, this difference is not observable.

A user of our proposed system will not need physical access to a specific device to analyze a video captured by the device. The system should accept videos of an arbitrary length, and will not require metadata unrelated to video playback to be intact. It will work with videos of any resolution, frame rate, or GOP structure. Also, as our approach will be data driven, it is imperative that a user have access to a sufficient database of videos with known labels.

2.1 Video Frame Deletion Detection

Detecting frame deletion is a binary classification problem. Given a Video V, there are two possible classes:

$$\begin{aligned} &C_0 : \text{The video is genuine, and has not had frames removed from it.} \\ &C_1 : \text{The video is altered, and has had frames removed from it.} \end{aligned} \tag{2.1}$$

Note that in this case, *genuine* refers to the fact that the video has not undergone any frame deletion. A video may have underwent other post processing operations such as color correction and re-sizing but not have had any frames removed. In this case, the video would be said to be genuine. From this point forward, any mention of a genuine video simply refers to a video that has not had frames removed from it.

In general, it is difficult to classify whether or not a video has had frames removed based on the entirety of a video directly. Thus, the problem must be reworked. As shown above, a feature extraction system will be used to produce the P-frame prediction error sequence $e(n)$, and a feature vector $\boldsymbol{x}$. The feature vector ideally contains information about the prediction error sequence that can perfectly separate the two classes. As such, the classification problem is as follows. Given a feature vector $\boldsymbol{x'}$, it belongs to one of two classes:

$$\begin{aligned} &C_0 : \boldsymbol{x'}\text{resulted from a genuine video that has not had frames removed from it.} \\ &C_1 : \boldsymbol{x'}\text{resulted from an altered video which has had frames removed from it.} \end{aligned} \tag{2.2}$$

In the following chapter, we will propose both a new method for extracting $e(n)$, and additional augmentations to $\boldsymbol{x}$ that allow for improved separation of data and increased robustness of the overall system.

Chapter 3: Proposed Approach

3.1 Prediction Error Sequence Extraction

In previous work on frame deletion detection in MPEG-2, the prediction error sequence was extracted directly from the video decoder using the DCT coefficients of the prediction error residuals located in the compressed video file. The prediction error was averaged over all macroblocks in a frame. This prediction error was then stored as a sequence. Due to the nature of the correlation between P-frame prediction errors across a single GOP, any prediction made across GOP boundaries would result in increased prediction error [4]. Wang and Farid showed that for fixed GOP video, the increase in average prediction error is periodic with respect to the number of frames deleted from the video. Stamm's work expands the idea of the prediction error trace by introducing the formulation of the problem as detecting the presence of a fingerprint signal $s(n)$. As H.264 uses variable GOP structures we will only be concerned with the model defined for variable GOP video. Stamm et al. defines the model of $s(n)$ as

$$s(n) = \beta \mathbb{1} \left(\Theta(n) = 0 \right). \tag{3.1}$$

where $\beta > 0$ is a constant and $\Theta(n)$ is a random variable distributed over the set $\{0, 1\}$ [5]. This model corresponds to modeling the fingerprint signal as randomly occurring sequence of discrete impulses with a magnitude of β. From this model they pose the detection of frame deletion as distinguishing between two hypotheses:

$$\begin{aligned} &H_0 : e(n) = e_1(n). \\ &H_1 : e(n) = e_2(n) = e_1(n) + s(n)e_1(n). \end{aligned} \tag{3.2}$$

Thus, detection of frame deletion is detection of the presence of the modulated fingerprint signal $s(n)e_1(n)$. Given an unknown video, Stamm et al. first makes an approximation of the unaltered

P-frame prediction error sequence. To do this they use a median filter with a filter width of 3.

$$\hat{e}(n) = \text{median}\{e(n-1), e(n), e(n+1)\}. \tag{3.3}$$

Thus, the relationship between the estimate and $e_1(n)$ is

$$e_1(n) = \hat{e}(n) + \epsilon(n). \tag{3.4}$$

where $\epsilon(n)$ is a zero mean random variable representing estimation error.

Using the estimate of the unaltered P-frame prediction error sequence, Stamm et al. calculates $\hat{s}(n)$, which is an estimate of the fingerprint signal modulated by the prediction error sequence as defined by

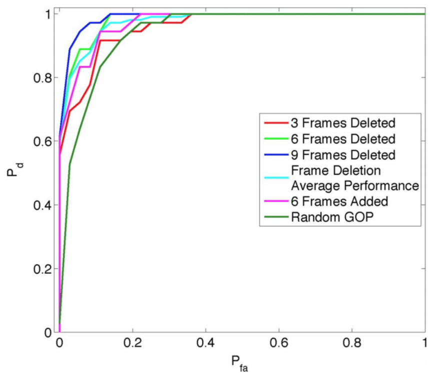
$$\hat{s}(n) = \max(e(n) - \hat{e}(n), 0). \tag{3.5}$$

The estimate of the fingerprint signal is floored at 0, as the model of the $s(n)$ dictates that it must be greater than or equal to 0. This estimate of the fingerprint signal can be used to build a detector. The decision function found by Stamm et al. for variable GOP video is

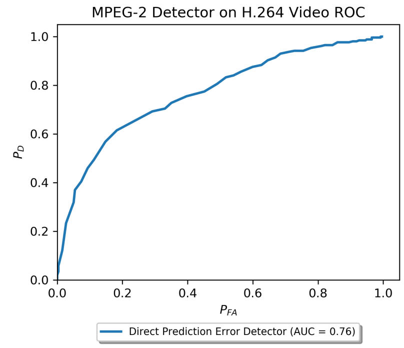
$$\delta_{var} = \begin{cases} H_0 & \text{if } \frac{1}{N}\sum_{n=1}^{N} |\hat{s}(n)| < \tau_{var} \\ H_1 & \text{if } \frac{1}{N}\sum_{n=1}^{N} |\hat{s}(n)| \geq \tau_{var} \end{cases} \tag{3.6}$$

where the decision is made on the basis of the energy in $\hat{s}$.

In MPEG-2, a P-frame is encoded by searching the previous anchor frame for the macroblock which incurs the least error [1]. This means that the average prediction error for a single P-frame is only asssociated with the previous I or P-frame. H.264 expands the capabilities of its motion compensation and estimation system by allowing prediction from multiple previous frames (and subsequent frames in the case of B-frames) [2]. If the prediction error trace is extracted via the codec for H.264, the average prediction error associated with one frame is comprised of a linear



(a) MPEG-2 Detector on MPEG-2



(b) MPEG-2 Detector on H.264

Figure 3.1: Comparison Between MPEG-2 Detection Methods used on (a) MPEG-2[a] and (b) H.264

[a]Figure reprinted with permission from Stamm et al.

combination of the average prediction error associated with motion vectors that map to the different anchor frames used in the motion estimation and compensation process. Thus, cross GOP predictions are smoothed out in such a way that it makes the fingerprint energy detector in Stamm et al.'s paper perform inadequately.

To test this, we collected 230 videos from a cell phone camera (the ASUS ZenFone 3 Laser), and generated an altered video with 15 frames removed from the beginning corresponding to each collected video. The encoding parameters were kept constant, and we fixed the GOP of the altered videos to that of the unaltered videos. In this particular case, the GOP structure was 30 frames in length, with 1 I-frame followed by 29 P-frames. We extracted the prediction error sequence directly from the codec, and measured the estimated fingerprint energy as described above.

As shown in Fig. 3.1 the perfomance of the detector using the methodology derived for MPEG-2 videos suffers a significant decrease when used on H.264 video, particularly at low false alarm rates. This is due to a limitation in the model used by Stamm et al. above in Equation 3.1. As it is possible to predict across multiple previous anchor frames in H.264, the contribution of the fingerprint signal is variable over time. This variation is also not regular, as scene content and motion determine how many cross GOP predictions are present in each P-frame. Thus we propose the updated model for the fingerprint signal as

$$s(n) = \beta(n)\mathbb{1}\left(\Theta(n) = 0\right). \tag{3.7}$$

where $\beta(n)$ is now a random variable that takes values in $\mathbb{R}_{\geq 0}$. Thus, we propose the following methodology for extracting the prediction error sequence in H.264.

3.1.1 Proposed Prediction Error Sequence Extractor for H.264

The goal of the proposed extraction algorithm is to maximize the probability that should frame deletion exist, a given measurement of the prediction error comes from a cross-GOP prediction. To this end, instead of directly measuring the prediction error from the DCT coefficients from the decoder, we decode the frame of interest and store the motion vectors associated with said frame. For each motion vector in the current P-frame, we find the x and y coordinates defining the source macroblock which provides the least error mapping from a particular previous anchor frame. Then for that previous anchor frame, we subtract the pixels in the source macroblock from the destination macroblock in the current frame. This leaves us with a prediction error residual associated with the motion vector. We then calculate the average absolute value of this residual.

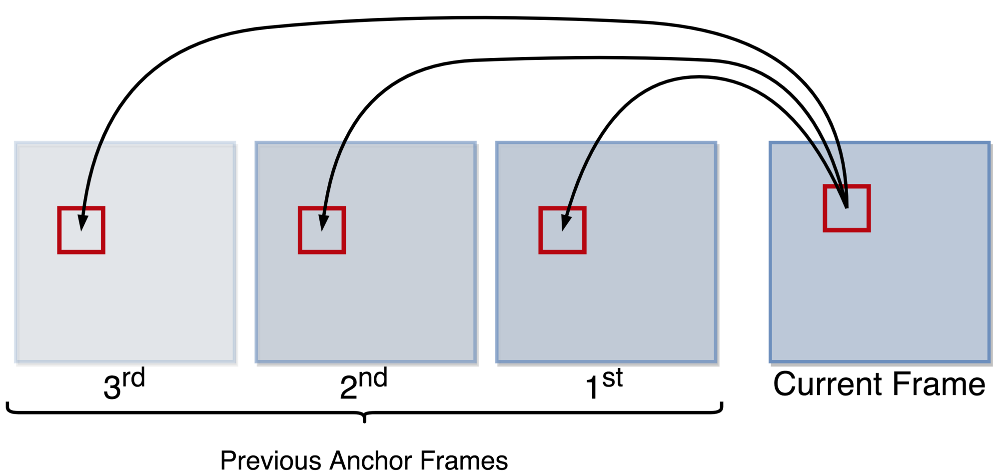


Figure 3.2: Identification of the source macroblock in 3 previous anchor frames

We repeat this process for each of D previous anchor frames. Then we store these prediction error values in a matrix $M[n]$, where n denotes the P-frame index of the current frame. Each row of the matrix corresponds to the errors associated with a single motion vector, and the columns are the errors associated with each of the previous anchor frames.

After obtaining the $M[n]$ matrix for the current frame, we create the matrix $\tilde{M}[n]$ defined like so:

$$\tilde{M}_{i,j}[n] = \mathbb{1}\left(j = \operatorname*{argmin}_{l}\left(M_{i,l}[n]\right)\right) * M_{i,j}[n] \tag{3.8}$$

Thus $\tilde{M}[n]$ is a copy of $M[n]$ where the non-zero entries in each row correspond to the minimum average error associated with the macroblock and all other elements are zero. Effectively, the column index of the non-zero entry is an estimation of which previous frame the motion vector associated with the macroblock maps to in the decoding process.

Further processing is done on $\tilde{M}[n]$ to output only a single prediction error value. First, $\tilde{M}[n]$ is reduced into a vector $P[n]$, such that:

$$P_j[n] = \frac{1}{N_j}\sum_{i}\tilde{M}_{i,j}[n] \tag{3.9}$$

Where N_j is the number of non-zero elements in the j^{th} column of $\tilde{M}[n]$. Then, the reported prediction error for the current frame $e*[n]$ is calculated as

$$e^*[n] = \max_{j} P_j[n] \tag{3.10}$$

This entire process is repeated for every P frame in the video. This method of error extraction estimates which previous anchor frame contributes the maximum error per macroblock to the overall prediction error residual obtained by the codec for a given P frame. Since prediction across GOP boundaries results in spikes in the prediction error, the anchor frame that contributes the most error is most likely to be from a different original GOP. In this manner, we obtain a trace that is resilient to advances made in the motion compensation and estimation process in modern codecs as well as robust to variable frame rates and dynamic GOP structures.

$$M[n] = \begin{bmatrix} 30.4 & 20.5 & 40.2 \\ 16.3 & 22.1 & 30.4 \\ 25.5 & 23.4 & 19.8 \\ \vdots & \vdots & \vdots \end{bmatrix}$$

$$\tilde{M}[n] = \begin{bmatrix} 0.00 & \mathbf{20.5} & 0.00 \\ \mathbf{16.3} & 0.00 & 0.00 \\ 0.00 & 0.00 & \mathbf{19.8} \\ \vdots & \vdots & \vdots \end{bmatrix}$$

$$P[n] = \begin{bmatrix} P_1[n] & P_2[n] & P_3[n] \end{bmatrix}$$

Figure 3.3: Methodology for forming $P[n]$

3.2 Proposed Detection Algorithm

In addition to the new methods for prediction error extraction, we propose an expanded detection algorithm to better capture the statistical differences between videos. In fact, depending on scene content, video capture settings, and the amount of motion captured in a single recording, the prediction error sequence and fingerprint signal exhibit different structural behavior. This is true even for videos captured from a single camera model. Figure 3.4 shows this clearly. The two videos were captured from an LG Nexus 5X using the high quality 1080p capture mode. Both videos were shot using similar scene content, but the amount of motion in each video is different. The first video was shot with hight motion, while the second video was comparatively low motion. The top row shows the different prediction error sequences, while the bottom row shows the different estimated fingerprint signals.

Notice the large discrepancy in magnitude between both types of signals depending on the amount of motion in the video. The original detection criteria defined by Stamm et al. in Equation 3.6 is based on the energy of the estimated fingerprint signal. This will lead to undesirable misclassifica-

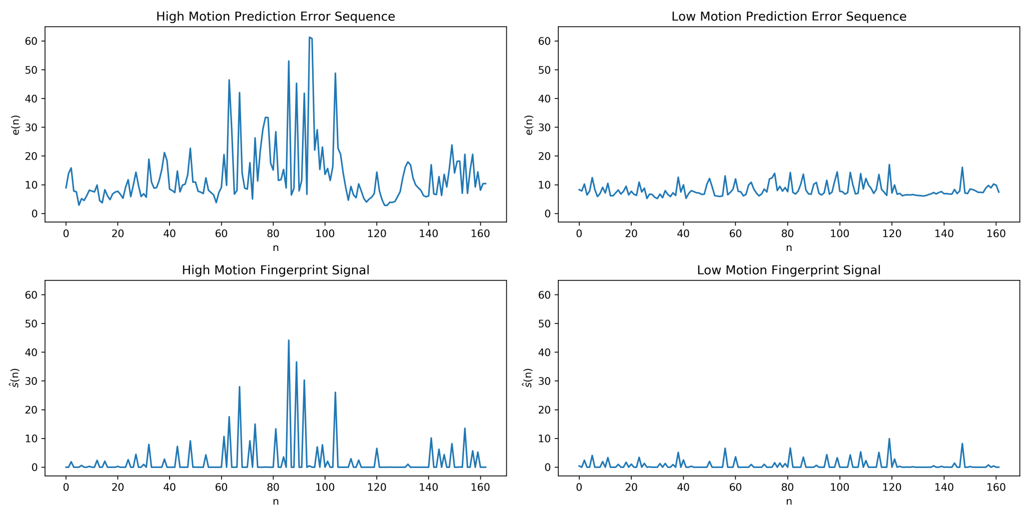


Figure 3.4: *Top Left* - The extracted prediction error sequence for a high motion video. *Top Right* - The extracted prediction error sequence for a low motion video. *Bottom Left* - Estimated fingerprint signal for a high motion video. *Bottom Right* - Estimated fingerprint signal for a low motion video.

tions when only using signal energy as the detection feature. Thus, we need a set of new features that can help account for this difference in fingerprint signal energy between videos.

Under the old model for the fingerprint signal, β was a constant, meaning the fingerprint signal was thought of as a randomly occurring sequence of discrete impulses with a magnitude of β. With the new model defined in Equation 3.7, $\beta(n)$ is a random variable that takes nonnegative real values. The effect of $\beta(n)$ can be seen in Fig. 3.5. The sample low motion video was reencoded with the first 15 frames removed. This accounts for half of a GOP. As the GOP structure of the video is fixed, it is expected that the fingerprint signal will be periodic [5]. The estimated fingerprint signal of the sample video with frame deletion shows some amount of periodicity but the amplitude of the signal varies with time. Modeling this variation of $\beta(n)$ for a given video can help aid in the detection of frame deletion.

Over several seconds of video both the prediction error sequence and the fingerprint signal are wide sense stationary. Due to this, we propose modeling both the fingerprint signal and prediction error sequence as autoregressive (AR) processes. The model parameters capture some of the statistical information about $\beta(n)$, and thus are added to a feature vector along with the fingerprint

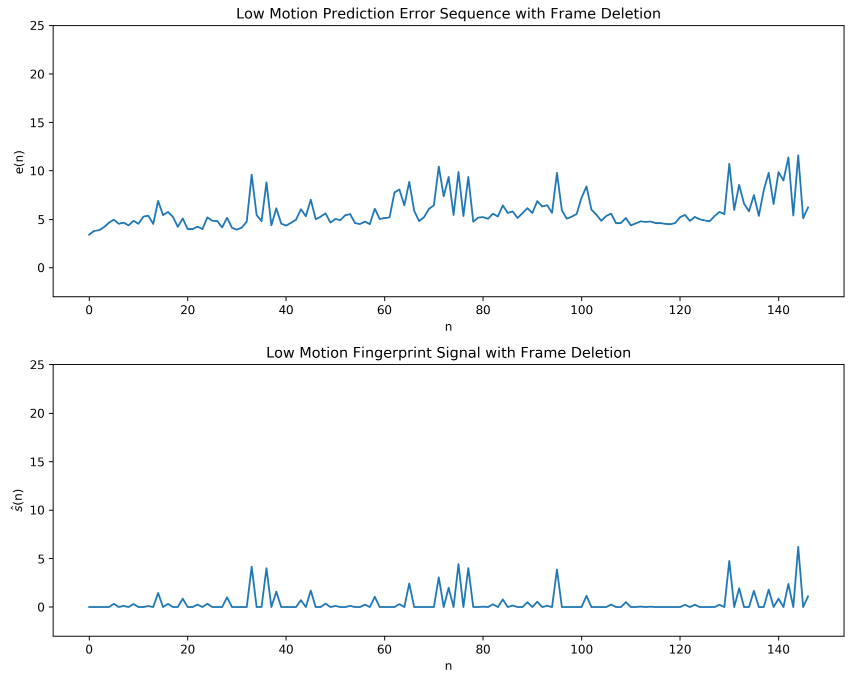


Figure 3.5: *Top* - The extracted prediction error sequence for a low motion video with frame deletion. *Bottom* - The Estimated fingerprint signal for a low motion video with frame deletion.

energy. In order to capture the degree to which the model fits a given sequence, the error variance of each AR model is also included. In addition, some basic statistical features are included to scale the overall decision surface. We propose including the mean and variance of both the prediction error sequence and fingerprint signal to the feature vector as well.

For a given video V, the feature vector used for classification $\boldsymbol{x}_V$ is structured as

$$\boldsymbol{x}_V = \begin{bmatrix} \frac{1}{N}\sum_{n=1}^{N} |\hat{s}[n]| \\ \mu_{\hat{s}[n]} \\ \sigma^2_{\hat{s}[n]} \\ \mu_{e^*[n]} \\ \sigma^2_{e^*[n]} \\ \boldsymbol{a}_1 \\ \boldsymbol{a}_2 \end{bmatrix} \tag{3.11}$$

Where $\hat{s}[n]$ is the estimated fingerprint sequence of $e^*[n]$ obtained by using Equation 3.5 above, but substituting $e^*[n]$ for $e[n]$, μ is the sample mean, σ^2 is the sample variance, $\boldsymbol{a}_1$ are the Q^{th} order AR model parameters for $e^*[n]$, and $\boldsymbol{a}_2$ are the Q^{th} order AR model parameters for $\hat{s}[n]$.

Note that after creating a feature vector for each video, the feature vector is quite large. It is inadvisable to create a probabilistic model of the feature vector for classification. Instead, we propose using a discriminative function to map an incoming feature vector directly to the set of natural videos or the set of videos altered by frame deletion. As such, we propose using a Support Vector Machine (SVM) classifier with a Radial Basis Kernel function for classification [6].

Bibliography

[1] ISO Central Secretary, "Information technology - Generic coding of moving video pictures and associated audio information - part 2: Video", en, International Organization for Standardization, Geneva, Switzerland, Standard ISO/IEC IS 13818-2, 2008.

[2] ——, "Information technology - coding of audio-visual objects - part 10: Advanced video coding (avc)", en, International Organization for Standardization, Geneva, Switzerland, Standard ISO/IEC IS 14496-10, 2010.

[3] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the h.264/avc video coding standard", *IEEE Trans. Cir. and Sys. for Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003, issn: 1051-8215. doi: 10.1109/TCSVT.2003.815165. [Online]. Available: https://doi.org/10.1109/TCSVT.2003.815165.

[4] W. Wang and H. Farid, "Exposing digital forgeries in video by detecting double mpeg compression", in *Proceedings of the 8th Workshop on Multimedia and Security*, ser. MM&Sec '06, Geneva, Switzerland: ACM, 2006, pp. 37–47, isbn: 1-59593-493-6. doi: 10.1145/1161366.1161375. [Online]. Available: http://doi.acm.org/10.1145/1161366.1161375.

[5] M. C. Stamm, W. S. Lin, and K. J. R. Liu, "Temporal forensics and anti-forensics for motion compensated video", *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 4, pp. 1315–1329, 2012, issn: 1556-6013. doi: 10.1109/TIFS.2012.2205568.

[6] C. Cortes and V. Vapnik, "Support-vector networks", *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995, issn: 1573-0565. doi: 10.1007/BF00994018. [Online]. Available: https://doi.org/10.1007/BF00994018.

Appendix A: Some Appendix Heading

This is the second paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

And after the second paragraph follows the third paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

After this fourth paragraph, we start a new paragraph sequence. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

This is the second paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Appendix B: Another Appendix Heading

And after the second paragraph follows the third paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

After this fourth paragraph, we start a new paragraph sequence. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

This is the second paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like

“Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

And after the second paragraph follows the third paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Vita

After this fourth paragraph, we start a new paragraph sequence. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

This is the second paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.