REGULARIZED ADABOOST FOR RGBD VIDEO CONTENT IDENTIFICATION

BY

HONGHAI YU

THESIS

Submitted in partial fulfillment of the requirements for the degree of Master of Science in Electrical and Computer Engineering in the Graduate College of the University of Illinois at Urbana-Champaign, 2012

Urbana, Illinois

Adviser:

Professor Pierre Moulin
This thesis presents three contributions. First, we provide an information theoretic analysis to a recently developed learning-based content identification (ID) algorithm, symmetric pairwise boosting (SPB). Second, we propose a regularized Adaboost algorithm, which tackles SPB’s implicit assumption that video segments are statistically independent. Finally, we develop the first hybrid content ID system for synchronized RGB and depth (RGBD) videos. Experimental results show the regularized Adaboost algorithm vastly outperforms SPB for all considered distortions, while the hybrid system further improves the content ID performance of regularized Adaboost relative to RGB-alone or depth-alone content ID systems.
ACKNOWLEDGMENTS

I would like to express my gratitude to all those who gave me the possibility to complete this thesis. The deepest gratitude goes to my adviser Professor Pierre Moulin for his guidance, encouragement, and support in all the time of research for and writing of this thesis. I especially want to thank him for being my role model as a scholar who has integrity and passion for research.

I am also grateful to Dr. Sujoy Roy and Rohit Naini for their advice and insightful discussions. In addition, I would like to thank Patrick Johnstone, Benjamin Chidester and Xinqi Chu for their assistance and friendship during the course of my study.

Next, I would also like to thank A*STAR Graduate Academy for the AUIP Scholarship.

Lastly, I would like to extend my appreciation to my family for their love, support and unwavering faith in me.
TABLE OF CONTENTS

LIST OF SYMBOLS ................................................................. v

CHAPTER 1 INTRODUCTION ..................................................... 1
  1.1 Outline of the Thesis .................................................. 2

CHAPTER 2 BACKGROUND ....................................................... 4
  2.1 Statement of the Content ID Problem ................................. 4
  2.2 Structured Content ID Codes ......................................... 6
  2.3 Symmetric Pairwise Boosting (SPB) .................................. 7
  2.4 Kinect Depth Video ..................................................... 11

CHAPTER 3 REGULARIZED ADABOOST FOR CONTENT ID ........... 15
  3.1 Statistical Motivation .................................................. 15
  3.2 Proposed Regularizer .................................................. 19
  3.3 Learning Theoretic Analysis of the Regularized Adaboost
      Algorithm ............................................................... 20
  3.4 Performance Evaluation .............................................. 22

CHAPTER 4 A HYBRID CONTENT ID SYSTEM FOR RGB+DEPTH
      VIDEOS ........................................................................ 28
  4.1 Statistical Motivation for a Hybrid System ......................... 28
  4.2 Proposed Hybrid System .............................................. 31
  4.3 Performance Evaluation .............................................. 32

CHAPTER 5 CONCLUSION .......................................................... 34

REFERENCES ........................................................................... 35
LIST OF SYMBOLS

\[ M \] Number of database video signals.

\[ N \] Number of segments in a database video.

\[ L \] Number of segments in a query video, i.e. granularity.

\[ T \] Number of frames in a segment.

\[ u \] A segment of the database video \( u \).

\[ x \] Intermediate feature of a segment in the database video \( u \).

\[ v \] A segment of the query video \( v \).

\[ y \] Intermediate feature of a segment in the query video \( v \).

\[ f \] Fingerprint of a segment of a database video.

\[ g \] Fingerprint of a segment of a query video.

\[ \mathcal{U} \] Alphabet in the original video domain

\[ \mathcal{X} \] Alphabet in the intermediate feature domain.

\[ \mathcal{F} \] Alphabet in the fingerprint domain.

\[ \phi \] Fingerprint extraction function.

\[ \psi \] Decoding function.

\[ \mathcal{A} \] Alphabet denoting four-level quantization, \( \mathcal{A} = \{a, b, c, d\} \).

\[ J \] Predefined number of classifiers in the Adaboost algorithm, \( J = 16 \) in this thesis.

\[ \mathcal{H} \] The class of feasible classifiers.

\[ h_j \] A classifier indexed by \( j \).

\[ \mathcal{F} \] A filter.
\( Q \) A quantizer.
\( g \) Ensemble classifier.
\( \mathcal{T} \) A training dataset.
\( \mathcal{S} \) Half of the training dataset consisting of matching pairs.
\( \mathcal{D} \) Half of the training dataset consisting of nonmatching pairs.
\( w_t \) Weight of training example \( t \).
\( \alpha_j \) Confidence associated with classifier \( h_j \).
\( \mu(X) \) Statistical expectation of the random variable \( X \).
\( \sigma(X) \) Standard deviation of the random variable \( X \).
\( H(X) \) Entropy of the random variable \( X \).
\( I(X;Y) \) Mutual information between the random variables \( X \) and \( Y \).
\( \rho \) Correlation coefficient between a given filter’s outputs of two different frames.
\( R \) Correlation coefficient between two filters’ outputs of a single frame.
\( \tau \) Decoding threshold.
CHAPTER 1

INTRODUCTION

If you wonder why you cannot access Megaupload.com from the United States, one of the world’s biggest file sharing websites, it has been shut down, together with many other file sharing websites, since the beginning of 2012. This is considered a victory for Hollywood, which claims that online piracy costs the movie industry billions of dollars and tens of thousands of jobs annually [1]. However, file sharing websites are not just about movie piracy, they also host copyright free and public domain media sources which benefit users and spur innovation. Thus, simply shutting down file sharing websites might not be the right way to fight online movie piracy.

As people continuously search for better technologies to protect, manage and retrieve video content, content identification (ID) has received considerable attention from both academia and industry over the past decade [2]. Different from watermarking, which inserts an identifier into the video content and thus changes the content, content ID extracts a signature (fingerprint) from the video content without changing it. A video fingerprint is a short summary of the video content that is robust to content-preserving distortions. The goal is then to match any query video to a database video by measuring the distance between the query fingerprint and the fingerprints in the database. Content ID can be used to filter pirated movies for file sharing websites. In particular, YouTube uses content ID to detect copyrighted video uploads. Once detected, copyrighted uploads are then either deleted or permitted based on the copyright holders’ agreement with YouTube. If the video upload is permitted, YouTube splits the advertisement revenue with the copyright holder [3]. Besides fighting online movie piracy, content ID can also be used for advertisement tracking, broadcast monitoring and law enforcement investigations [4, 5].

In the literature, many video fingerprinting algorithms have been proposed based on heuristic signal features [5]. There have also been attempts to for-
mulate a theoretical framework for fingerprint-based content ID systems. For instance, the papers [6, 7] derive the fundamental relation between database size and query length under some statistical assumptions. Model-based decoding which exploits the underline statistical model in the fingerprint space is studied in [8, 9]. On the fingerprint code design side, a family of fingerprinting algorithms that employs a variation of Adaboost to select filters and quantizers, such as Asymmetric Pairwise Boosting (APB) [10] and Symmetric Pairwise Boosting (SPB) [11, 4], has demonstrated excellent content ID performance.

Despite promising results, video content ID systems still face the limitation that video frames are 2D projections of the 3D world and depth information is lost. Fortunately, the advance of sensing technology makes it possible to equip videos with depth information. In particular, Xbox Kinect cameras can output both RGB and depth videos at a very low cost [12]. To the best of our knowledge, all current video content ID systems are based on RGB videos only. We expect that RGB+depth (RGBD) videos will become widespread in the future, and that databases such as [13, 14, 15] will be commonplace. Hence a goal of this thesis is to investigate how depth information can help identify query videos.

The first contribution of this thesis is to provide a theoretical explanation to the SPB algorithm of [11, 4], showing each iteration of SPB maximizes a lower bound on the mutual information between matching fingerprint pairs. The second contribution is to develop a regularized Adaboost algorithm, which tackles SPB’s implicit assumption that video segments are statistically independent which is never the case in practice because of segment overlapping. The proposed algorithm is tested on both RGB and depth videos, demonstrating significantly better performance than SPB. The third contribution is to develop the first hybrid content ID system of synchronized RGB and depth videos based on our regularized Adaboost algorithm. The hybrid system shows large performance gain over regular RGB and depth systems.

1.1 Outline of the Thesis

The rest of the thesis is organized as follows:
• **Chapter 2** provides background material. In particular, it gives a mathematical overview of a video content ID system and the system parameters, summarizes the baseline algorithm, symmetric pairwise boost (SPB), and introduces the properties of depth video captured by Xbox Kinect and the RGB+depth dataset used in this thesis.

• **Chapter 3** first performs an information theoretic analysis of the SPB algorithm and then proposes a regularized Adaboost algorithm aiming at increasing the mutual information between original and degraded fingerprints. A learning theoretic analysis of the proposed algorithm is also provided, followed by experimental evaluations.

• **Chapter 4** develops the first hybrid content ID system for RGB+depth videos from the same statistical motivation as regularized Adaboost in Chapter 3. Performance evaluation is provided to justify our reasoning.

• **Chapter 5** concludes the thesis with a summary of the contributions. A brief discussion of possible future work is also included.

We follow the convention that uppercase letters represent random variables while lowercase letters represent particular realizations of these random variables (RVs). A vector is denoted by an underscore (e.g., $f$) and a temporal sequence by a boldface letter (e.g., $\mathbf{f}$).
CHAPTER 2

BACKGROUND

In the literature, video content ID is also called content-based video copy detection, video fingerprinting, robust video hashing, or perceptual hashing. They all refer to a system that takes a snippet of a video as a query and seeks a match in a fingerprint database. As shown in Fig. 2.1, a video content ID system can be broken down into an offline part and an online part. The fingerprint database is built offline by extracting fingerprints from all reference videos. When a query video comes in, its fingerprint will be extracted and used as a query in the fingerprint database.

In this section, we provide a mathematical overview of the video content ID problem with an emphasis on the design parameters. We focus on structured content ID codes, which encompass many current content ID algorithms. We investigate a learning-based approach, symmetric pairwise boosting (SPB) [4], which has recently demonstrated excellent content ID performance. We also present special properties of depth video pertinent to content ID.

![Figure 2.1: Overview of a video content ID system.](image)

2.1 Statement of the Content ID Problem

A video content database is defined as a collection of $M$ elements, $u(m) \in U^N$, $m = 1, 2, \ldots, M$, each of which is a sequence of $N$ segments $\{u_1(m), u_2(m)\}$.
A segment in video content ID systems is a short video snippet of $T$ consecutive frames. Segments are often chosen to overlap temporally to prevent misalignment during matching [4, 5]. For instance, the video fingerprinting paper [4] uses overlapping time windows that are 1 sec long and start every 100 ms; the temporal overlap is 9/10. A 3-minute video is represented by $N = 1800$ segments. It is desired that the video be identifiable from a short clip, say 5 sec long, corresponding to $L = 41$ segments. This is called the granularity of the video ID system [4]. Typically $L \ll N$.

The problem is to determine whether a given query consisting of $L$ segments, $v \in U^L$, is related to some element of the database, and if so, identify which one. To this end, an algorithm $\psi(\cdot)$ must be designed, returning the decision

$$\psi(v) \in \{0, 1, 2, \ldots, M\},$$

where $\psi(v) = 0$ indicates that $v$ is unrelated to any of the database elements.

Algorithm performance is evaluated using several metrics [16], some of which are listed here.

- **Robustness**: Ideally, a fingerprint should stay largely unaffected under various content-preserving signal degradations. Given two perceptually similar video signals, $u$ and $v$, a fingerprint function $\phi$ should satisfy $d(\phi(u), \phi(v)) \leq \tau$ with high probability, for some distance function $d$ in fingerprint space and threshold $\tau$.

- **Discriminative power**: If $u$ and $v$ are not perceptually similar, we should have $d(\phi(u), \phi(v)) > \tau$ with high probability.

- **Compactness**: The fingerprint size in bits per second should be much smaller than the bit rate of the original video.

- **Granularity**: This is the query length expressed in number of segments.

- **Search speed**: A video content ID system should operate in real time in commercial applications with large video databases.

- **Storage**: The physical storage space required to store the database signals is linear in $MN$. 
Besides the aforementioned parameters, false negative rate and false positive rate are often used as quantitative measures of the effectiveness of content ID systems. A false negative occurs when the algorithm fails to detect the correct match. This happens when the query is perceptually similar to one of the items in the database but their fingerprint distance exceeds the detection threshold. Conversely, a false positive occurs when the algorithm detects an incorrect match. This happens when the query is not related to any item in the database but its fingerprint is close to the fingerprint of some item in the database.

Note some fundamental tradeoffs between system parameters. For instance, larger granularity often leads to slower search speed, higher robustness reduces discriminative power, and false positive rate can be decreased at the expense of high false negative rate. Therefore, the choice of parameters is application-dependent.

2.2 Structured Content ID Codes

In this thesis, we restrict our attention to the following fairly general class of fingerprint-based content ID codes. The codes of [16, 11, 4] among others, fall in this category.

**Definition 1** A \((M, N, L)\) content ID code for a size-\(M\) database populated with \(U^N\)-valued content items, and granularity \(L\), is a pair consisting of a mapping \(\phi : U \rightarrow F\) generating an encoding function \(\Phi : U^N \rightarrow F^N\) that returns a fingerprint \(f = \Phi(u)\) with components \(f_i = \phi(u_i)\) for each \(1 \leq i \leq N\), and a decoding function \(\psi : F^L \rightarrow \{0, 1, \ldots, M\}\) returning \(\hat{m} = \psi(\Phi(v))\).

Hence the mapping \(\phi\) is applied independently to each segment. It might be convenient to impose additional structure on the code. For instance, the mapping \(\phi : U \rightarrow F\) in [11, 4] is obtained by applying a set of \(J\) optimized filters to each segment and quantizing each of the \(J\) real-valued filter outputs to four levels. Hence \(F\) takes the form \(A^J\) with \(A = \{a, b, c, d\}\). In this case we view the fingerprint as an array \(f = \{f_{ij}, 1 \leq i \leq N, 1 \leq j \leq J\}\) and the query fingerprint as an array \(g = \{g_{ij}, 1 \leq i \leq L, 1 \leq j \leq J\}\) where \(i\) denotes time and \(j\) filter index. We also use the notation \(\_ = \{f_j, 1 \leq j \leq J\}\) for
the subfingerprint associated with a given video segment. We also write \( \phi \) in vector form as \( \phi = \{ \phi_j, 1 \leq j \leq J \} \).

The decoding function \( \psi \), in most video content ID systems, measures distance between fingerprints \([6, 8, 4]\). If the distance is less than a predefined decision threshold, the fingerprint is declared as a match for the query. This is a variable-size list decoder: the number of matches could be 0, 1, 2 or more. Alternatively, a single-output decoder might be used, returning only the index of the closest match. In this thesis, the Hamming distance metric with a list decoder is used to make a fair comparison with the SPB algorithm of \([4]\).

Given a query fingerprint, \( g = \{ g_{ij}, 1 \leq i \leq L, 1 \leq j \leq J \} \), the Hamming distance between \( g \) and some database fingerprint \( f = \{ f_{N_0+i,j}, 1 \leq i \leq L, 1 \leq j \leq J \} \) at time offset \( N_0 \in \{0,1,2,\ldots,N-L\} \), is given by

\[
d_H(f,g|N_0) \triangleq \sum_{j=1}^{J} \sum_{i=1}^{L} \mathbb{1}\{g_{ij} \neq f_{N_0+i,j}\}.
\]  

(2.1)

The decoder outputs a list of all \( m \) such that the minimum distance (over all \( N_0 \)) between the database fingerprint subsequence \( f(m) \) starting at time \( N_0 \) and the query fingerprint \( g \) is below the threshold:

\[
\psi(g) \triangleq \left\{ m \in \{1,2,\ldots,M\} : \min_{0 \leq N_0 \leq N-L} d_H(f(m), g|N_0) < \tau \right\}.
\]  

(2.2)

### 2.3 Symmetric Pairwise Boosting (SPB)

The core of a video content ID system is the fingerprint extraction algorithm. Figure 2.2 (corresponds to the fingerprint extraction module in Fig. 2.1) shows a high-level description of the SPB fingerprint extraction algorithm. The code is a structured ID code in the sense of Def. 1.

#### 2.3.1 Intermediate Feature Extraction

It can be difficult to extract good fingerprints directly from raw video clips because of the difficulties of working with high-dimensional data. In most content ID systems, a dimensionality reduction module is implemented be-
fore fingerprint extraction, and is termed feature extraction. We call the output of this module intermediate features, to avoid confusion with the final fingerprint (which sometimes is called the binary feature).

Ideally, intermediate features should be sufficient statistics for the identification problem. It is not clear whether any nontrivial such feature exists, and moreover the probability distribution of video data is not accurately known. Therefore, many heuristic features have been proposed and evaluated on some large datasets. This includes spatial features based on intensity change of a single image frame, temporal features based on a sequence of consecutive frames, color features computed in some color space, transformed domain features such as wavelet transform coefficients, or a combination of different types of features. The paper [5] provides an excellent review of video features for content ID.

The intermediate features we use are block mean luminance (BML) for RGB video and block mean depth (BMD) for depth video. These features have been shown to work well with SPB for RGB video content ID [4]. First, each RGB (converted into grayscale) and depth frame is divided into $N_r \times N_c$ blocks ($N_r$ rows and $N_c$ columns). The intermediate feature $x \in X$ at block $B_{r,c,t}$ in the $r$-th row, $c$-th column and $t$-th frame ($1 \leq t \leq T$) of the video segment $u \in U$ is calculated as

$$x^{RGB}(r, c, t) = \frac{1}{|B_{r,c,t}|} \sum_{(i,j) \in B_{r,c,t}} u^{RGB}(i, j, t),$$  \hspace{1cm} (2.3)

and

$$x^{D}(r, c, t) = \frac{1}{|B_{r,c,t}|} \sum_{(i,j) \in B_{r,c,t}} u^{D}(i, j, t),$$  \hspace{1cm} (2.4)

Figure 2.2: Fingerprint extraction algorithm.
where $|\cdot|$ denotes set cardinality, and $u^{RGB}(i,j,t)$ and $u^{D}(i,j,t)$ are the luminance and depth values at coordinates $(i,j)$ in the $t$-th frame, respectively. Hence, $\mathcal{X} = \mathbb{R}^{N_r \times N_c \times T}$.

2.3.2 Adaboost for Filter and Quantizer Selection

Filters and quantizers had been heuristically chosen until learning-based methods, such as Adaboost, were proposed. The performance of Adaboost-chosen filters and quantizers has been shown to outperform the heuristically chosen ones [10, 11, 4]. The following is a summary of the learning algorithm, symmetric pairwise boosting (SPB), in [11, 4].

The training set $\mathcal{T} = \{(x_k, y_k, z_k) \in \mathcal{X}^2 \times \{\pm 1\}, k \in \mathcal{T}\}$ is comprised of a subset $\mathcal{S}$ of $|\mathcal{T}|/2$ matching pairs and a subset $\mathcal{D}$ of $|\mathcal{T}|/2$ nonmatching pairs. A pair $(x_k, y_k) \in \mathcal{X}^2$ is said to be matching if the second feature is a distorted version of the first, and nonmatching if the two features are independent. The binary variable (label) $z_k$ is equal to 1 (resp. -1) if $(x_k, y_k)$ is matching (resp. nonmatching). Define the classifier $h_j : \mathcal{X}^2 \rightarrow \{\pm 1\}$ as

$$h_j(x,y) = \begin{cases} +1 & \text{if } \phi_j(x) = \phi_j(y) \\ -1 & \text{otherwise}, \end{cases} \quad (2.5)$$

where $\phi_j$ is parameterized by a filter $\mathcal{F}_j : \mathcal{X} \rightarrow \mathbb{R}$ and a quantizer $Q_j : \mathbb{R} \rightarrow \mathcal{A}$,

$$\phi_j(x) = Q_j(f_j(x)), \quad (2.6)$$

and $1 \leq j \leq J$. Denoted by $\mathcal{H}$ is the class of feasible classifiers (indexed by the choice of filter and quantizer).

A popular family of filters is the Haar-like filters used in [10, 11, 4]. They are efficient to compute from integral images [17] and rich enough to describe perceptually significant visual features. The SPB algorithm uses 3D Haar-like filters. The filter function defined on the intermediate feature $x$ of a video segment can be calculated as

$$f(x) = \frac{1}{|A \cup B|} \left\{ \sum_{(r,c,t) \in A} x(r,c,t) - \sum_{(r,c,t) \in B} x(r,c,t) \right\}, \quad (2.7)$$

where $A$ and $B$ are the sets of spatio-temporal positions $(r,c,t)$ in the light
and dark regions shown in Fig. 2.3, respectively. For \( N_r = 4, N_c = 9, \) and \( T = 10, \) the number of candidate filters is 12,450 [4]. To reduce the computational complexity of the training, a limited number of candidate quantizers are evaluated. In [4], filter output is quantized into four levels (\( \mathcal{A} = \{a, b, c, d\} \)), and 680 = \( \binom{17}{3} \) candidate quantizers from 17 logarithmically spaced threshold values are considered.

![Figure 2.3: Haar-like filters: (a) spatio-temporal average, (b) temporal difference, (c,d) spatial difference, and (e,f) spatio-temporal difference.](image)

The SPB algorithm is an adaptation of the well-known Adaboost classification algorithm given in Table 2.1. Each \( h_j \in \mathcal{H} \) is a weak classifier parameterized by a filter \( \mathbb{f}_j \) and quantizer \( Q_j. \)

Table 2.1: Adaboost for filter and quantizer selection.

<table>
<thead>
<tr>
<th>Input:</th>
<th>training set ( T \triangleq {(x_k, y_k, z_k) \in \mathcal{X}^2 \times {\pm 1}, k \in T} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initialization:</td>
<td>define equal weights ( w^{(1)}_k = 1/</td>
</tr>
<tr>
<td>Do for ( j = 1, \ldots, J )</td>
<td>1. Choose the classifier ( h_j \in \mathcal{H} ) that minimizes the weighted error ( e_j = \sum_{k \in T} w^{(j)}_k \mathbb{1}{h_j(x_k, y_k) \neq z_k}. ) \hfill (2.8) \n2. Compute ( \alpha_j = \frac{1}{2} \log \frac{1 - e_j}{e_j}. ) \n3. Update the weights ( w^{(j+1)}_k = w^{(j)}<em>k \exp{-\alpha_j z_k h_j(x_k, y_k)}. ) \n4. Normalize the weights so that ( \sum</em>{k \in T} w^{(j+1)}_k = 1. )</td>
</tr>
<tr>
<td>Output:</td>
<td>( J ) pairs of filter and quantizer ( {(\mathbb{f}<em>j, Q_j)}</em>{j=1}^J ) which parameterize the chosen ( J ) classifiers ( {h_j}_{j=1}^J. )</td>
</tr>
</tbody>
</table>

Upon completion of the algorithm, Adaboost would output the boosted
A Statistical View of Adaboost

The discrete Adaboost algorithm in Table 2.1 (with predictor variable \((X,Y)\) and binary response variable \(Z \in \{\pm 1\}\)) admits a known interpretation as an iterative procedure for fitting an additive logistic regression model [18, 19].

\[
g(x, y) = \sum_{j \in \mathcal{J}} \alpha_j h_j(x, y),
\]

under the exponential loss function

\[
L(z, g(x, y)) = \exp\{-z g(x, y)\}.
\]

At the \(j\)-th iteration, one solves

\[
(\alpha_j, h_j) = \arg \min_{\alpha \in \mathbb{R}, h \in \mathcal{H}} \sum_{k \in T} w_k^{(j)} \exp\{-\alpha z_k h(x_k, y_k)\},
\]

where \(w_k^{(j)} = \exp\{-z_k g_{j-1}(x_k, y_k)\}\) and

\[
g_i(x, y) \triangleq \sum_{1 \leq j \leq i} \alpha_j h_j(x, y).
\]

The solution to (2.12) yields the same \((\alpha_j, h_j)\) as given in steps 1 and 2 of Table 2.1.

2.4 Kinect Depth Video

Depth information has traditionally been either estimated from RGB images using stereo matching, which is computationally expensive, or measured by
expensive laser scanners. However, recent development in sensing technology makes depth acquisition computationally and financially more affordable. In particular, Xbox Kinect (see Fig. 2.4) outputs real-time, high-quality, synchronized videos of RGB and depth at a cost of about one hundred dollars. Kinect was designed for gaming, but has soon found applications to various research problems in signal processing, computer vision, robotic navigation, and computer graphics. The application of Kinect to real-time human pose recognition won the best paper at the top computer vision conference CVPR [20] in 2011.

The Kinect sensor consists of an infrared laser emitter, an infrared camera, and an RGB camera, as shown in Fig. 2.4. The emitter emits fixed patterns to the environment, and the infrared camera receives the reflected signal. Then depth information is measured by a triangulation process [21]. The current Kinect works only in indoor environments and has a working range of 0.5 m to 5.0 m according to the specifications [21]. Still it has the potential to make consumer-grade video cameras produce RGBD videos and project them in a 3D TV [22]. Besides rendering for 3D TV, depth could also help for video content ID system to solve the problems stated in Chapter 1. In Chapter 4 of this thesis, we examine how depth helps RGB improve video content ID performance.

Following the introduction of Kinect, many datasets have been created and made publicly available including the NYU depth dataset V2 [13], ADSC human daily activity dataset [23], LIRIS human activities dataset [14], and University of Washington RGB-D object dataset [15]. Among them, the NYU depth dataset V2 captures the most comprehensive indoor environments and
is used in all of our experiments. It is comprised of 464 scenes taken from three cities, and each scene is recorded as a short RGBD video that represents a database item in the content ID system.

Fig. 2.5 shows some random RGBD images from the NYU depth dataset. Depth images are not compressed, and each pixel is presented by the 11 bits outputted by Kinect. It is clear that depth images are quite noisy. Moreover there are missing pixel values which are caused by shadows from the disparity between the infrared emitter and camera or random missing or spurious values from specular or low albedo surfaces [13]. These missing values could be filled by a colorization scheme [13], but it takes a few seconds to fill one image, which makes it impractical for a content ID system. Moreover, the intermediate feature uses only the block average information which alleviates the missing value problem. Another difficulty with the dataset is that the 464 scenes are all indoor environments coming from only 26 scene types. This means interclass (each scene is a class) similarity is high, which makes identification difficult. For such a difficult dataset, we will examine how much depth can help improve the content ID performance in Chapter 4.
Figure 2.5: NYU depth dataset.
A major shortcoming of Adaboost for filter selection is the implicit assumption that clips are drawn independently from some unknown distribution. In practice, segment overlapping is necessary to overcome misalignment during matching. For instance, the papers [11] and [4] used overlapping factors of 15/16 and 9/10, respectively. So segments are often correlated, which leads to correlated fingerprints. In this section, we propose to use a regularizer in the Adaboost algorithm to explicitly control fingerprint correlation, and thus improve the content ID performance.

3.1 Statistical Motivation

3.1.1 Content ID Capacity

As developed in the paper [6], a content ID system, like any other communication system, is subject to a fundamental capacity limit that upper bounds the rate at which information can be decoded with arbitrarily low probability of error. But unlike a communication system where the degradation channel is fixed, the content ID capacity is shown to be given by the constrained maximization problem

\[ C = \max_{P_{FG}} I(F; G), \]  

(3.1)

where the maximization is over the joint distribution \( P_{FG} = P_F P_{G|F} \), in the case subfingerprints are mutually independent and the subfingerprint degradation channel is memoryless. Both the marginal distribution \( P_F \) and the conditional distribution \( P_{G|F} \) are determined by the fingerprint extraction function \( \phi = \{ \phi_j, 1 \leq j \leq J \} \). The mutual information \( I(F; G) \) is a non-decreasing function of the number of filters \( J \), which is often constrained by
design parameters such as storage and search speed introduced in Section 2.1. As a result, the maximization of (3.1) is constrained by a fixed number of filters.

In case the subfingerprints form a stationary ergodic process and the sub-fingerprint degradation channel is stationary ergodic, one may conjecture that $C = \max_{P_{FG}} I(F; G)$, where $I(F; G)$ is the mutual information rate between the random processes $F$ and $G$.

### 3.1.2 Information Theoretic Analysis of SPB

In this section, we show that, at each iteration, SPB maximizes a lower bound on the single-segment mutual information $I(F, G) = H(F) - H(F|G)$ associated with the joint probability distribution $P_{FG}$. At the $j$-th iteration, SPB selects the classifier that minimizes the weighted error:

$$h_j = \arg\min_{h \in H} \left[ \sum_{k \in S} w_k^{(j)} \mathbb{1}\{h(x_k, y_k) = -1\} + \sum_{k \in D} w_k^{(j)} \mathbb{1}\{h(x_k, y_k) = 1\} \right],$$

(3.2)

where $S$ and $D$ are defined in Section 2.3.2, and (3.2) is equivalent to step 1 in Table 2.1. The two error terms in (3.2) are the empirical weighted false-negative and false-positive error probabilities, respectively. For a given classifier $h$, the empirical version of the false-negative error probability for matching fingerprints, $P_e = P_{FG}(F \neq G)$, is given by

$$\widehat{P}_e = \Pr(F \neq G|S, h) = \sum_{k \in S} w_k^{(j)} \mathbb{1}\{h(x_k, y_k) = -1\},$$

(3.3)

and the empirical version of the false-positive error probability, $P_F P_G(F = G)$, is

$$\widehat{P}(F = G|D, h) = \sum_{k \in D} w_k^{(j)} \mathbb{1}\{h(x_k, y_k) = 1\}.$$

(3.4)

First, we derive a link between $\Pr(F \neq G|S, h)$ and $H(F|G)$. From Fano’s inequality, we know

$$H(F|G) \leq H(P_e) + P_e \log(|A| - 1),$$

(3.5)

where $P_e = P_{FG}(F \neq G)$ and $A$ is the alphabet for scalar fingerprint in-
troduced in Section 2.2. We can easily verify the tightness of (3.5) for our video content ID system. In Fig. 3.1, we show the empirical equivocation \( \hat{H}(F|G) \) and Fano’s upper bound \( H(\hat{P}_e) + \hat{P}_e \log(|A| - 1) \) evaluated from 16,000 matching pairs and 16 classifiers. The upper bound is fairly tight. Thus, minimizing \( \Pr(F \neq G|S, h) \) is equivalent to minimizing a tight upper bound on \( H(F|G) \).

Figure 3.1: \( \hat{H}(F|G) \) and \( H(\hat{P}_e) + \hat{P}_e \log(|A| - 1) \). The x-coordinate is the classifier index.

Figure 3.2: \( \hat{H}(F) \) and \( -\log \hat{Pr}(F = G) \). The x-coordinate is the classifier index.
Next, we derive a link between \( \hat{\Pr}(F = G|\mathcal{D}, h) \) and \( H(F) \). When \( F \) and \( G \) are generated from nonmatching pairs, we model them by a product distribution with identical marginals. From Lemma 2.10.1 in [24], we have

\[
P_F P_G(F = G) \geq 2^{-H(F)}, \tag{3.6}
\]

for two independent identically distributed (iid) random variables \( F \) and \( G \). Then, \( H(F) \) is lower bounded by

\[
H(F) \geq -\log P_F P_G(F = G). \tag{3.7}
\]

Again, we can verify the tightness of (3.7) from nonmatching pairs of the training data. As shown in Fig. 3.2, the lower bound is tight. Thus, minimizing \( \hat{\Pr}(F = G|\mathcal{D}, h) \) is equivalent to maximizing a tight lower bound on \( H(F) \).

From the preceding argument, we conclude that each iteration of SPB simultaneously minimizes an upper bound on \( H(F|G) \) and maximizes a lower bound on \( H(F) \), thus maximizes a lower bound on \( I(F;G) = H(F) - H(F|G) \). As nicely as it fits the information theoretic framework, SPB trains on a single segment, and does not take into consideration that a query fingerprint consists of \( L \) consecutive segments.

Segments are temporally overlapped to overcome misalignment during matching, which leads to temporally correlated fingerprints \( \mathbf{F}_j = \{F_{1j}, F_{2j}, \ldots, F_{Lj}\} \) for each chosen classifier \( h_j \). In a memoryless channel where the output fingerprint only depends on the input fingerprint at that time and is conditionally independent of previous channel inputs or outputs, we know that \( I(F_j; G_j) \leq \sum_{i=1}^L I(F_{ij}; G_{ij}) \) in general, \( I(F_j; G_j) \ll \sum_{i=1}^L I(F_{ij}; G_{ij}) \) when \( \{F_{1j}, F_{2j}, \ldots, F_{Lj}\} \) are highly correlated, and \( I(F_j; G_j) = \sum_{i=1}^L I(F_{ij}; G_{ij}) \) when \( \{F_{1j}, F_{2j}, \ldots, F_{Lj}\} \) are iid. Thus we can increase the mutual information by decorrelating temporal fingerprints. For video content ID, most common distortions are frame-wise operations, such as resizing, cropping and rotation, leading to memoryless channels. If the channel is not memoryless, the effect of decorrelating \( \{F_{1j}, F_{2j}, \ldots, F_{Lj}\} \) on \( I(F_j; G_j) \) is not clear.

In Section 3.2, we show that the classifiers’ ability to decorrelate segments differs dramatically across different types of filters. In order to increase mutual information by decorrelating temporal fingerprints, we propose to use a
regularizer to effectively eliminate those filters that generate highly correlated fingerprints from the candidate pool $\mathcal{H}$. We use experimental evaluation to demonstrate the effectiveness of this regularizer.

3.2 Proposed Regularizer

By (2.5) and (2.6), a classifier $h$ is characterized by a filter $f$ and a quantizer $Q$. In this section, we focus on the ability to temporally decorrelate filter-outputs. Uncorrelated filter responses tend to produce uncorrelated quantized outputs, hence temporally uncorrelated fingerprints.

![Figure 3.3: Average correlation coefficient of filter responses, $\bar{\rho}(f)$, for the family of Haar-like filters on video frames.](image)

For a given filter $f$, the response $f(X) = \{f(X_i), 1 \leq i \leq L\}$ is an $L$-dimensional random vector. The correlation coefficient between segments $f(X_s)$ and $f(X_t)$ is defined by

$$\rho(s, t) = \frac{\mathbb{E}[(f(X_s) - \mu)(f(X_t) - \mu)]}{\sigma^2},$$

where $\mu$ and $\sigma$ are the mean and standard deviation of the common distribution underlying the filter response of each segment $f(X)$. The coefficient $\rho(s, t) \in [-1, +1]$ is a measure of the linear dependence between two random variables $f(X_s)$ and $f(X_t)$, where $\rho(s, t) = \pm 1$ indicates (negatively) perfect correlation and $\rho(s, t) = 0$ indicates $f(X_s)$ and $f(X_t)$ are uncorrelated. Define
the average correlation coefficient of \( \mathbf{f}(\mathbf{X}) \) as

\[
\overline{\rho}(f) = \frac{1}{L^2 - L} \sum_{s \neq t} |\rho(s, t)|. \tag{3.9}
\]

The functional \( \overline{\rho}(f) \) captures the filter’s ability to decorrelate overlapping segments and can be easily estimated from the training dataset. In Fig. 3.3, we show the estimated \( \overline{\rho}(f) \) for the family of Haar-like filters (see Fig. 2.3) used in [4] on RGB video segments. Within the family, type (b), (e) and (f) filters can decorrelate overlapping segments extremely well, type (a) and (c) filters produce almost perfectly correlated responses, and type (d) filters produce moderately correlated responses.

As \( \overline{\rho}(f) \) captures a filter’s ability to decorrelate overlapping segments, we use this criterion to regularize the Adaboost algorithm. In our new regularized Adaboost algorithm shown in Table 3.1, filters with large average correlation coefficient are penalized with the new objective function

\[
\text{err}_j = e_j + \gamma \overline{\rho}_j, \tag{3.10}
\]

where \( \overline{\rho}_j = \overline{\rho}(f) \) for \( h_j \)’s that are parameterized by \( f \), \( e_j \) is the weighted error given by (2.8), and \( \gamma \) is the tuning parameter which can be chosen by cross-validation. We also write \( \overline{\rho}(h) \) to express the dependence on \( h \), and \( \overline{\rho}(h) = \overline{\rho}(f) \) if classifier \( h \) is parameterized by \( f \).

### 3.3 Learning Theoretic Analysis of the Regularized Adaboost Algorithm

Here, we perform the same analysis for the regularized Adaboost algorithm that was given for SPB in Section 2.3.3. For regularized Adaboost, the loss function is now the regularized exponential loss

\[
L(z, g(x, y)) \triangleq \exp\{-z g(x, y)\} + \sum_{j \in \mathcal{J}} 2 \sinh(\alpha_j) \gamma \overline{\rho}_j, \tag{3.11}
\]

where \( g \) is defined in (2.10), and the second term in the right-hand side is a regularizer that penalizes highly correlated filter responses.
Table 3.1: Regularized Adaboost for filter and quantizer selection.

**Input:** training set $T \triangleq \{(x_t, y_t, z_t) \in X \times \{-1, 1\}, t \in T \}$

**Initialization:** define equal weights $w^{(1)}_t = 1/|T|, \forall t \in T$

**Do for** $j = 1, \ldots, J$

1. Choose the classifier $h_j \in H$ that minimizes the regularized weighted error

$$err_j = \sum_{t \in T} w^{(j)}_t \mathbb{1}\{h(x_t, y_t) \neq z_t\} + \gamma \bar{p}_j$$

2. Compute $\alpha_j = \frac{1}{2} \log \frac{1 - err_j}{err_j}$.

3. Update the weights

$$w^{(j+1)}_t = w^{(j)}_t \exp\{-\alpha_j z_t h_j(x_t, y_t)\}.$$  

4. Normalize the weights so that $\sum_{t \in T} w^{(j+1)}_t = 1.$

**Output:** $J$ pairs of filter and quantizer $\{(f_j, Q_j)\}_{j=1}^J$ which parameterize the chosen $J$ classifiers $\{h_j\}_{j=1}^J$.

At iteration $j$, we solve

$$(\alpha_j, h_j) = \arg \min_{\alpha \in \mathbb{R}, h \in H} \sum_{k \in T} \left[ w^{(j)}_k \exp\{-\alpha z_k h(x_k, y_k)\} + 2 \sinh(\alpha) \gamma \bar{p}(h) \right].$$  

(3.12)

Using the fact that $h(x, y) \in \{-1, 1\}$ and $\sum_{k \in T} w^{(j)}_k = 1$, the objective function of (3.12) can be rewritten as

$$\left[ e^{-\alpha} \sum_{h(x_k, y_k) = z_k} w^{(j)}_k + e^\alpha \sum_{h(x_k, y_k) \neq z_k} w^{(j)}_k \right] + \left( e^\alpha - e^{-\alpha} \right) \gamma \bar{p}(h)$$

$$= \left[ (e^\alpha - e^{-\alpha}) \sum_{k \in T} w^{(j)}_k \mathbb{1}\{h(x_k, y_k) \neq z_k\} + e^{-\alpha} \sum_{k \in T} w^{(j)}_k \right] + \left( e^\alpha - e^{-\alpha} \right) \gamma \bar{p}(h)$$

$$= 2 \sinh(\alpha) \sum_{k \in T} w^{(j)}_k \mathbb{1}\{h(x_k, y_k) \neq z_k\} + \gamma \bar{p}(h) + e^{-\alpha}. \quad (3.13)$$

21
The minimum over \( h \in \mathcal{H} \) is given by

\[
h_j = \arg \min_{h \in \mathcal{H}} \sum_{k \in T} w^{(j)}_k 1\{h(x_k, y_k) \neq z_k\} + \gamma \bar{p}(h).
\] (3.14)

Plugging \( h_j \) into (3.13) and solving for \( \alpha \), we obtain

\[
\alpha_j = \frac{1}{2} \log \frac{1 - \text{err}_j}{\text{err}_j},
\] (3.15)

where \( \text{err}_j \) is the regularized weighted error rate

\[
\text{err}_j = \sum_{k \in T} w^{(j)}_k 1\{h_j(x_k, y_k) \neq z_k\} + \gamma \bar{p}_j.
\] (3.16)

Thus, (3.14) and (3.15) are equivalent to steps 1 and 2 in Table 3.1.

Notice that the preceding derivation does not depend on the specific form of \( \bar{p}(h) \). As long as the regularizer is a functional of \( h \), it can be plugged into the regularized Adaboost algorithm, which makes this approach fairly general.

### 3.4 Performance Evaluation

In this section, we evaluate the performance of the proposed regularized Adaboost algorithm on both RGB and depth video datasets. The content ID performance is compared with SPB [4] and shows significant improvement.

#### 3.4.1 Experimental Setup

We follow the same experimental setup as in the SPB paper [4]. Before extracting intermediate features, both RGB and depth videos are resampled at 10 frames per second, converted to grayscale (RGB only) and resized to QVGA (320x240). These preprocessing steps aim to make the fingerprinting algorithm robust to frame rate change, color variation, and frame resizing. After preprocessing, block mean luminance (BML) and block mean depth (BMD) are extracted from RGB and depth video clips on 36 \((N_r = 4, N_c = 9)\) blocks per frame. The temporal length of the intermediate features is 1 second \((T = 10)\), and the query length is 5 seconds \((L = 41)\). We train
\( J = 16 \) classifiers each for RGB and depth. For the weight of the regularizer \( \gamma \), values between 0.1 and 0.3 worked well in our experiments. The results shown are obtained using \( \gamma = 0.2 \) for RGB and \( \gamma = 0.1 \) for depth.

Figure 3.4: Sample distorted images. Top row: Original RGB and depth images. Bottom two rows: Distorted RGB and depth images. Distortions from left to right are: cropping of 50\%, vertical mirroring, rotation at 15 degree and shift downward and left by 100 pixels each.

From the NYU depth dataset, we use 115 videos for training and another 115 videos for testing. The only selection criterion is to make the training set and testing set contain a roughly equal number of scenes from each scene type. Other than that, the training set and testing set are randomly selected. The training data includes 16,000 matching and 16,000 nonmatching pairs (\(|\mathcal{T}| = 32,000\)) of sequences of intermediate features from 10 consecutive synchronized RGB and depth frames. The matching pairs are generated from the following video distortions (see Fig. 3.4):

1. 50\% cropping: 50\% of the central portion of the image is retained while the boundaries are removed.
2. Vertical mirror: reflect pixels around the center vertical axis.
3. Rotation at an angle of 15 degrees.
4. Shifting downward and left by 100 pixels each.
We consider geometric distortions only as they represent the most challenging video distortions to detect, and SPB works nearly perfectly for simple distortions, such as lossy compression, resizing, and frame rate change [4] (so does regularized Adaboost). The nonmatching pairs are generated from intermediate feature sequences extracted from different video signals. In all experiments, we use four quantization levels ($\mathcal{A} = (a, b, c, d)$) and use gray code to convert to binary fingerprint. As noted earlier, 17 threshold values (680 candidate quantizers) logarithmically spaced in $[-255, 255]$ (RGB) and $[-2047, 2047]$ (depth) are considered to reduce the computational complexity.

3.4.2 Selected Filters

Table 3.2 summarizes the filter types selected by SPB and regularized Adaboost for both RGB and depth. The striking difference between SPB and regularized Adaboost is that the latter selects almost exclusively type (b) filters (temporal-difference filters, see Fig. 2.3), while SPB selects filters of various types. As Adaboost reweights training examples after each iteration, to correctly classify those higher weighted examples (incorrectly classified in previous iterations) may require a different type of filters. Thus, SPB selects different types of filters to best fit the training examples. However, in regularized Adaboost, reducing weighted classification error is not the only objective at each iteration. The ability to decorrelate overlapping segments is also considered. The regularizer effectively eliminates filters of type (a), (c) and (d) which generate highly correlated responses on overlapping segments. With similar ability to decorrelate overlapping segments, filters of type (b) clearly fit the training examples better than types (e) and (f). The superiority of the type (b) filters is demonstrated next in a comparative test.

3.4.3 Comparative Test

To compare the content ID performance of SPB and regularized Adaboost, we generate 40,667 matching and 25,073,193 nonmatching pairs of 5-second intermediate feature sequences for each distortion in Fig. 3.4. To quantify ID performance, we plot the receiver operating characteristics (ROC) curves for different distortions. Each point on the curve represents a false negative
Table 3.2: Filters selected by SPB and regularized Adaboost for RGB and depth videos.

<table>
<thead>
<tr>
<th>RGB</th>
<th>Depth</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPB</td>
<td>Regularized</td>
</tr>
<tr>
<td>(c)</td>
<td>(b)</td>
</tr>
<tr>
<td>(a)</td>
<td>(b)</td>
</tr>
<tr>
<td>(b)</td>
<td>(b)</td>
</tr>
<tr>
<td>(a)</td>
<td>(b)</td>
</tr>
<tr>
<td>(c)</td>
<td>(b)</td>
</tr>
<tr>
<td>(b)</td>
<td>(b)</td>
</tr>
<tr>
<td>(a)</td>
<td>(b)</td>
</tr>
<tr>
<td>(d)</td>
<td>(b)</td>
</tr>
<tr>
<td>(b)</td>
<td>(b)</td>
</tr>
<tr>
<td>(b)</td>
<td>(b)</td>
</tr>
<tr>
<td>(d)</td>
<td>(b)</td>
</tr>
<tr>
<td>(b)</td>
<td>(b)</td>
</tr>
<tr>
<td>(c)</td>
<td>(a)</td>
</tr>
<tr>
<td>(b)</td>
<td>(b)</td>
</tr>
<tr>
<td>(b)</td>
<td>(b)</td>
</tr>
<tr>
<td>(a)</td>
<td>(b)</td>
</tr>
</tbody>
</table>

rate and false positive rate pair corresponding to a decoding threshold \( \tau \).

Fig. 3.5 and Fig. 3.6 show the ROC curves of RGB and depth videos, respectively. Irrespective of the modality used, regularized Adaboost outperforms SPB for all the considered distortions. For vertical mirroring and image rotation, the performance gain is rather significant. In particular, at a false positive rate of \( 10^{-4} \), regularized Adaboost reduces the false negative rate by more than an order of magnitude under the distortion of image rotation.
Figure 3.5: ROC curves for RGB video against various distortions: (a) cropping; (b) vertical mirroring; (c) rotation; (d) shift.
Figure 3.6: ROC curves for depth video against various distortions: (a) cropping; (b) vertical mirroring; (c) rotation; (d) shift.
CHAPTER 4

A HYBRID CONTENT ID SYSTEM FOR RGB+DEPTH VIDEOS

In Chapter 3, we showed that the regularized Adaboost algorithm decorrelates temporal fingerprints and improves the content ID performance for both RGB video and depth video. In this chapter, we combine RGB and depth information in a hybrid content ID system. We first state the statistical motivation for such a hybrid system, and then propose a practical system under the general regularized Adaboost framework. Experimental evaluation shows the hybrid system outperforms regular RGB and depth systems by a large margin.

4.1 Statistical Motivation for a Hybrid System

After each iteration of SPB, training examples are reweighed such that the classifiers “similar” to those in previous iterations are not selected. However, this reweighing does not eliminate classifier correlation (see Table 4.2 on page 31). We show in this section that combining filters from RGB and depth further decorrelates classifiers.

4.1.1 Statistical Difference between RGB and Depth Images

As shown in Fig. 4.1a and 4.1b, we can reasonably infer the depth information from its corresponding RGB image. Likewise an algorithm has recently been developed to estimate depth information from a single RGB image [25]. So we want to examine whether depth information is redundant in a content ID system when combining with RGB. If it is not, by how much can it help improve the content ID performance?

Intuitively, depth images contain more homogeneous patches and fewer localized features, such as lines, edges and corners. One way to quantify this
is to fit the fine-scale wavelet coefficients into a two-parameter generalized Gaussian distribution (GGD) model [26, 27, 28]

\[ P_X(x : s, p) = \frac{\exp\left(-|x/s|^p\right)}{Z(s, p)}, \]  

(4.1)

where the normalization constant is \( Z(s, p) = 2s^p \Gamma\left(\frac{1}{p}\right) \) with \( \Gamma \) denoting the gamma function. Here, \( s \) is the standard deviation and \( p \) is the shape parameter. The GGD model contains the Gaussian and Laplacian probability density functions (PDFs) as special cases, using \( p = 2 \) and \( p = 1 \), respectively. For decreasing values of \( p \), the tails of the distribution become increasingly flat.

![Figure 4.1](image_url)

(a) RGB image; (b) depth image; (c) GGD fit for RGB; (d) GGD fit for depth.

Figure 4.1: Log histogram of Haar wavelet coefficient in LH subband: (a) RGB image; (b) depth image; (c) GGD fit for RGB; (d) GGD fit for depth.

Fig. 4.1 shows the fit of the GGD model to the log histogram of Haar wavelet coefficients in the LH subband, which captures the image’s horizontal
edges. The GGD parameters are estimated by maximizing the likelihood of the data [29]. A larger shape parameter for the depth image suggests that the depth image contains fewer horizontal edges than the RGB image. This is confirmed by the absence of book edges in Fig. 4.1b.

To be statistically significant, we fit the GGD model to another 1400 RGB+depth image pairs and summarize the estimated shape parameters in Table 4.1. The large difference between mean values of the shape parameters is consistent with our intuition that depth images contain more homogeneous regions and fewer localized features.

Table 4.1: Statistics of the estimated shape parameters.

<table>
<thead>
<tr>
<th></th>
<th>Range</th>
<th>Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>RGB</td>
<td>[0.15, 0.76]</td>
<td>0.4045</td>
</tr>
<tr>
<td>Depth</td>
<td>[0.16, 1.27]</td>
<td>0.6008</td>
</tr>
</tbody>
</table>

4.1.2 Statistical Motivation

The GGD model shows a clear statistical difference between RGB and depth images. Next, we show that this difference can help decorrelate classifiers.

In Chapter 3, we considered temporal correlation of fingerprints for filter \( j \). Here, we examine the filter correlation \( F = \{ F_j, 1 \leq j \leq J \} \) for a given segment. To capture the correlation between any two filters, we first define the within-modality correlation of two filters as

\[
R^m(j, k) = \frac{\mathbb{E} \left[ (\mathbb{f}_j^m(X^m) - \mu_j^m)(\mathbb{f}_k^m(X^m) - \mu_k^m) \right]}{\sigma_j^m \sigma_k^m},
\]

(4.2)

where \( m = 1, 2 \) denotes RGB and depth (D) respectively, \( X^m \) is the intermediate feature of one segment from the corresponding modality, and \( \mu_j^m \) and \( \sigma_j^m \) are the mean and standard deviation of \( \mathbb{f}_j^m(X^m) \). We define the between-modality correlation

\[
R^{RGBD}(j, k) = \frac{\mathbb{E} \left[ (\mathbb{f}_j^{RGB}(X^{RGB}) - \mu_j^{RGB})(\mathbb{f}_k^D(X^D) - \mu_k^D) \right]}{\sigma_j^{RGB} \sigma_k^D},
\]

(4.3)
the average absolute within-modality correlation,

\[ \overline{R}^m = \frac{2}{J^2 - J} \sum_{j=1}^{J-1} \sum_{k=j+1}^{J} |R^m(j, k)|, \]  
\[ (4.4) \]

and the average absolute between-modality correlation,

\[ \overline{R}^{RGBD} = \frac{1}{J^2} \sum_{j=1}^{J} \sum_{k=1}^{J} |R^{RGBD}(j, k)|, \]  
\[ (4.5) \]

of the \( J \) filters generated by the regularized Adaboost algorithm. The filters generated by SPB exhibit a similar pattern. The average correlations \( \overline{R}^m \) and \( \overline{R}^{RGBD} \) can be estimated from the training dataset, and their values are shown in Table 4.2. The average between-modality correlation is an order of magnitude smaller than the average within-modality correlation. This motivates us to propose a hybrid content ID system.

<table>
<thead>
<tr>
<th>( \overline{R}^{RGB} )</th>
<th>( \overline{R}^{D} )</th>
<th>( \overline{R}^{RGBD} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2303</td>
<td>0.1705</td>
<td>0.0199</td>
</tr>
</tbody>
</table>

4.2 Proposed Hybrid System

When both RGB and depth are available in video signals, we build the hybrid system based on regularized Adaboost. As illustrated in Fig. 4.2, half of the filters and quantizers are trained from RGB intermediate features, and the other half from depth intermediate features. Then for each RGD video, half of the fingerprint is generated from RGB and half from depth. The combined final fingerprint is used to identify the video in the hybrid system. From the previous analysis, these hybrid classifiers generate less-correlated fingerprints. In the next section, we show the performance evaluation of our hybrid system.
4.3 Performance Evaluation

We follow the same experiment setup as in Section 3.4.1. As shown in Fig. 4.3, the hybrid system outperforms the regular system for all the considered distortions. For the distortion of image rotation, performance gain is in orders of magnitude. Even more stunning is the result for the distortion of vertical mirroring, where false negative rate is zero for all possible values of false positive rates based on a simulation of 25,073,193 nonmatching query pairs.

We show in Fig. 4.4 the distributions of Hamming distance for matching and nonmatching pairs under the distortion of vertical mirroring. The progressive improvement in the histogram separation explains the better ROC curves of Fig. 3.5b and Fig. 4.3b.
Figure 4.3: ROC curves for hybrid content ID system under various distortions: (a) cropping; (b) vertical mirroring; (c) rotation; (d) shift.

Figure 4.4: Distributions of Hamming distance for matching and nonmatching pairs for the vertical mirroring distortion. (a) SPB for RGB, (b) regularized Adaboost for RGB, and (c) hybrid system based on the regularized Adaboost.
This thesis has made several contributions. First we have shown that each iteration of SPB maximizes a lower bound on the mutual information \( I(F; G) \) between matching subfingerprint pairs. Second, we have proposed a regularized Adaboost algorithm which reduces the temporal correlation of the fingerprint sequence. Third, we have proposed a hybrid system for synchronized RGB and depth videos. Experimental results show the regularized Adaboost algorithm vastly outperforms SPB for all considered distortions, while the hybrid system further improves the content ID performance of regularized Adaboost relative to RGB-alone or depth-alone content ID systems.

Based on the results of this thesis, many future research directions are possible. First, as mentioned at the end of Section 3.3 about the learning theoretic analysis of regularized Adaboost, other regularizers other than the average correlation coefficient can be examined without altering the structure of the loss function. Second, more complicated RGB and depth fusion methods can be applied to develop new hybrid systems. Third, better decoding metrics could be used, analogously to [8].
REFERENCES


