Chapter 4

Security

4.1 INTRODUCTION

This chapter deals with the construction of the video fingerprint-IDS and a technique for embedding the video fingerprint-IDS. Initially, an introductory note is given on the video fingerprinting technique, and then some preliminary preprocessing factual is given. The procedure for the generation of video fingerprint-IDS is given elaborately. Further, the investigation on traitors is given with a suitable example. Finally, through the simulation model the efficiency, effectivity, and performance is evaluated.

One of the major concerns in the p2pVoD system for the content owners is the susceptibility of the movie to piracy. Piracy can be unauthorized duplicating, copying, recording and redistributing the contents of the videos without taking legitimate permissions (Bloom, 2003), (Zou et al, 2009). The contents of the video can be easily copied or redistributed without any restrictions. Therefore, the need for content management in peers is very much essential to restrict the wide spread of piracy. One such content management is a Digital Rights Management (DRM) system, which must be mandated for the practical deployment of the videos in peers of p2pVoD system (Iwata et al, 2003), (Sung et al, 2006), (Liu et al, 2007), (Lee, 2009). A DRM system (Camp, 2003) controls the use of video contents by preventing access, denying duplications, and preventing the conversion to other formats.

Video fingerprinting (Kundur and Karthik, 2004), (Nikolaidis and Pitas, 2006), (Zhao and Liu, 2006) is one of the emerging techniques of a DRM system, which protects the video from unauthorized dissemination in the p2pVoD system. In this technique, each user of p2pVoD system is assigned a unique ID called as fingerprint-IDS. This fingerprint-ID represents a user as a legitimate representative of the p2pVoD system. Each video that is transmitted to the peer will have an embedded fingerprint-ID in it. If multiple number of users request for the same video at the same time then the fingerprint-IDS of each user is embedded in the same copy of the video, before being transmitted to them. If any of the users indulges in copying or redistributing the content of the video then the embedded fingerprint-ID can be used to trace the source of violation. But, there are risks that adversary may try to purge their identities by obscuring the fingerprint-IDS from the video. Such an attack from
the adversary on the fingerprint-ID is known as tampering (Wu et al, 2004). Another critical type of attack is known as collusion. In collusion, a group of adversaries will combine their identical copies of the video to generate a new version of pirated copy of the video with fingerprint-ID attenuated or completely removed. Good design of the fingerprint-IDS should resist such types of attacks.

The design of the good fingerprint-ID must focus on two factors: (a) the design of the fingerprint codes and, (b) conjointly considering the coding method and the embedding technique (He and Wu, 2006). The designing of good fingerprint codes cogitates on coding theories and then emphasizes on criterions. The criterions are attack assumptions, code size and length, collusion size, ability to trace traitors, etc. Also note that, both coding method and the embedding technique emphasis on the computational complexity. Supplementary to the designing of the fingerprint code, there exist non-coding fingerprints (He and Wu, 2007), where mutually orthogonal spreading sequences is assigned to the users as fingerprint-IDS. The non-coded fingerprint-IDS are easy to implement. But, the computational complexity is increased linearly with increase in number of users.

In this chapter, we have proposed an optimal design for generating the fingerprint codes, and an efficacious embedding technique. We have also proved the optimality of our proposed fingerprint design in terms of efficiency, effectivity, and performance.

4.2 PRELIMINARIES
Initially, the videos in the streaming server are preprocessed before transmitting to the peers. The format of movies stored in the streaming server is a standard MPEG format. The video blocks of the movie are herein used to distinguish between the AC components and DC components. Further, these DC components are preferred to embed the fingerprint-ID. Generally, in any MPEG video format (Raju et al, 2008), the pictures are related to a sequence of groups known as Group of Pictures (GoPs). This picture is a series of coded frames ordered sequentially. Hence, MPEG videos uses a block based coding scheme. A basic unit of MPEG coding is $16 \times 16$ pixels known as macro block. Each macro block constitutes a frame. There are three possible types of frame; I-frame, P-frame and B-frame. A P-frame is called as inter coded frame, because it uses the correlation between the current frame and the past frame. The past frame can be a reference to either an I-frame or a P-frame, but not a
B-frame. The B-frame is called as *bidirectional coded frame*, because it can refer to a past frame or a future frame, or both past and future frame. Both P-frame and B-frame are coded using *motion vectors* and *motion compensation* technique respectively. I-frame is called as *intra coded frame*, because all the necessary information is encoded within itself. Each block is processed independently with a Discrete Cosine Transform (DCT) technique.

The DCT technique generates a representation of each block from a spatial domain to a frequency domain. Wherein, the pixels in each block are likely to be correlated. Hence, the resulting DCT coefficients consist of few larger values and many smaller values. These values of the DCT coefficients in each block are compressed by a standard quantization matrix with a quality level of 50. The resultant of this quality level of quantization obtains an excellent compression quantity without compromising the quality on decompression. After quantization, these transformed coefficients are arranged approximately in a zigzag fashion. The arrangement is based on increasing frequency of the coefficients and then it is stored in a vector. This kind of approximation is important because the low frequencies with the larger values are grouped at the front part of the vector and high frequencies with the smaller values are grouped at the rear part of the vector. This vector will contain two types of coefficients; (a) *DC coefficient* and, (b) *AC coefficients*. The DC coefficient determines an average luminous, and the AC coefficients describe the variations around the DC coefficient in a block. For each block, DC coefficient is differentially coded with the last block’s DC term, and AC coefficients are encoded by a run length encoding technique. In a run length encoding technique for AC coefficients, any sequence of identical values will be replaced by a counter that identifies the number of repetitions and the values that are repeated. Huffman coding technique is used to code the AC coefficients statistically according to the probability of occurrence. So that, the short code are assigned to higher probable coefficients, and long code are assigned to lesser probable coefficients. This results in a compact bit stream from which the quantized DCT coefficients can be reconstructed perfectly. The reconstruction of the blocks is done by applying Inverse Discrete Cosine Transformation (IDCT).

Different DCT coefficients have different persuade on the embedded fingerprint robustness (Zhu and Zhang, 2009). Therefore, the DCT embedding fingerprint must satisfy the following conditions for a good robustness of fingerprint.
1) The DCT coefficients must not change much by the signal processing and the noise interference.

2) If fingerprints are attacked then only DCT coefficients can change slightly, but strictly not the embedded fingerprint-IDS.

3) After embedding the fingerprint-IDS the visual quality of the original video must not change significantly.

As per the above conditions, the low and medium frequencies of AC coefficients are widely preferred locations for fingerprint placements, whereas DC coefficients are rarely preferred for embedding the fingerprint. However, DC coefficients are more suitable for embedding the fingerprint than any of the AC coefficients for at least two reasons. The first reason is that, the amplitude of the DC coefficient is much larger than any of the AC coefficients in a block. According to the visual system illumination coverage characteristics, the higher DC coefficient - the higher is its perceptual capacity. This means that having a higher perceptual capacity allows a larger fingerprint to be embedded without having much of perceptual distortion. Usually, the DC coefficient is tens of multiples, and even hundreds of multiples bigger than the biggest AC coefficient. The change of proportion in DC coefficient is less than AC coefficient, which means that DC coefficient has a higher perceptual capacity than the biggest AC coefficient. The second reason is that, during signal processing such as lossy compression, low-pass filtering, sub-sampling, Digital-to-Analog and Analog-to-Digital conversions; the DC coefficients have much less influence on the signal processing than the AC coefficients. Therefore, embedding the fingerprint-IDS in DC coefficients is more stable than embedding the fingerprint-IDS in AC coefficients. In the next section a detailed discussion is given on how to generate a fingerprint-ID and how to embed it in the DC coefficients.

4.3 FINGERPRINT GENERATION

A robust embedding technique (Trappe et al, 2002) must have the capacitance to defy the attacks of the adversary, who might tamper the embedded fingerprint-ID. Hence, a DRM is used to enforce the authorization policy on the video blocks which contains the fingerprint-IDS. Wherefore, a unique label is used to identify the authorization assigned to the intended recipient, which herein is called fingerprint-ID. This unique fingerprint-ID is embedded into the video blocks prior to the distribution, so that, it can facilitate tracing the traitors, who might have used their video contents for
illegitimate purposes. But, the limitation of embedding fingerprint-IDS is that there are many possibilities for a group of adversaries with different versions of the fingerprint-IDS to collude together. Thereby, collectively attack against the fingerprint-ID is possible. Therefore, the design of the fingerprint-ID must resist such collusion attacks, and also must identify the colluders. Eventually, the spread spectrum is a prevailing technique to construct an impregnable fingerprint-ID. This technique protects the content by combining the robustness and the capacitance. In our proposed fingerprint generating technique we have used the concept of spread spectrum technique.

In our proposed fingerprint technique there are four steps involved in designing and embedding a fingerprint-ID. In the first step, the DC coefficients of a quantized DCT transformation from the video are identified for embedding the fingerprint. In the second step, a binary valued codeword are constructed using a modified balanced incomplete block designs to ensure imperceptibility, which apparently uses the spread spectrum technique. In the third step, the fingerprint-ID is embedded in the selected quantized DC coefficients. In the fourth step, the fingerprinted version of the video blocks is transmitted to the peers in a chaining fashion.

4.3.1 IDENTIFICATION OF DC COEFFICIENTS

The identification of DC coefficients is carried out through the statistical analysis. For the purpose of the analysis, many video blocks were randomly chosen, and the DCT blocks were evaluated. In these DCT blocks most of the higher energies are concentrated on the DC coefficients, and very few on the AC coefficients. In order to embed the fingerprint-ID, the distribution of DCT is exploited from the video blocks. For this purpose, the statistical analysis is done on the DCT coefficients after quantization. Experiments were conducted on each video block that has a group of pictures; these pictures are a series of coded frames ordered sequentially. Now, the I-frame in the picture is divided into 8x8 disjoint blocks to which Discrete Cosine Transform technique is applied. Let us consider, an 8x8 block in this frame. Let \( p(x, y), 0 \leq x \leq 7, 0 \leq y \leq 7 \) be the pixels, and let \( D(x, y) \) be its corresponding DCT coefficients. Then, the quantization technique is applied on these DCT coefficients as follows

\[
D'(x, y) = \text{round} \left( \frac{D(x, y)}{\text{round}(Q(x, y) \times Q_p)} \right)
\]  

(4.1)
where $Q(x,y)$ is the $xy^{th}$ element in the quantization table, and $Q_p$ is the scaling factor that controls the bit rate. In an 8x8 block more than one bit can be embedded in it, but not all blocks are used for embedding the bits. Because a slight mutation in the quantized DC coefficient $D'(0,0)$, can cause major visual artifacts in the video playback sequences. However, in the texture regions, a smaller mutation in the quantized DC coefficient can be perceptually undetectable because of the masking effect of the texture. Therefore, only the texture-rich blocks are preferred for embedding the fingerprint-IDS. For identifying the texture-rich blocks there are several known techniques, such as Variance (Puri and Aravind, 1991), Texture Energy (Tan et al, 1996), Bandpass Energy (Wong et al, 2000), Total Variation (Choi and Au, 1993) etc. In our proposal, to identify a texture-rich block we have used an activity measurement of Texture Energy in a block. In this activity measurement, we have used DC energy of the quantized $D'(0,0)$ coefficient for selecting the texture-rich block. The DC energy $E_{DC}$ of a block is computed as follows

$$E_{DC} = (D'(0,0) \times \text{round}[Q(0,0) \times Q_p])^2$$

(4.2)

Herein, the blocks with $E_{DC}$ larger than a threshold $T$ are declared as texture-rich blocks, whereas, the value of the threshold $T$ is a function of the scaling factor $Q_p$, which are obtained through experiments based on the texture in the picture. Hence, the texture-rich DC coefficients with the values larger than threshold $T$ in the video blocks are selected for embedding the fingerprint-IDS. This Selected DC Coefficients (SDC) of the video blocks is ordered sequentially as $DC_0, DC_1, DC_2, DC_3, \cdots, DC_{last}$. Next, fingerprint-ID is constructed and embedded in these SDC, which is described in the next section.

4.3.2 CONSTRUCTION OF FINGERPRINT-ID

The design of the fingerprint should consider embedding issues, which must survive in case of users’ collusion. To identify the colluders, an Anti-Collusion Codes (ACC) with code modulation can be used to construct a set of fingerprint-IDS. An anticollusion code is a set of code vectors that shares the bits within the codeword, which will identify colluders in a group of colluding users. The most frequently used technique for constructing the ACC is the Balanced Incomplete Block Design (BIBD) (Dinitz and Stinson, 1992), (Colbourn and Dinitz, 1996).

A $(v,k,\lambda)$ BIBD is a design that contains a pair $(X,A)$, where for a given set $X$ of $v$ elements there exists $A$ such that it contains $(k-1)$ subset elements of $v$
element set $X$. The condition is that each pair of elements of $X$ occurs together in exactly $\lambda$ blocks (subsets). Hence, in this proposal we have modified the BIBD to construct a binary valued anti-collision code vectors for generating the fingerprint codes. In the modified BIBD, we include two additional parameters, i.e., cluster and movie, known as Movie in a Cluster based Balanced Incomplete Block Design (MC-BIBD).

A MC-BIBD is a set $\chi$ of $v$ basis elements that contains orthogonal symbols of unique characteristics, and each such set is unique to a cluster $c$. $b$ is a collection of subsets of $\chi$, called slab, such that, it satisfies the following conditions:

1) Each slab consists of symbols of $v$ basis elements that are unique to a cluster $c$.
2) Each slab consists of exactly $k$ basis elements, such that, $v > k > 0$.
3) Each basis element of $k$ appears exactly in $r$ slabs, such that $r > 0$.
4) Each pair of basis elements for a movie $m$ appear simultaneously in exactly $\lambda$ slabs, such that $\lambda > 0$.

MC-BIBD is referred as $(v, b, r, k, \lambda, m, c)$ design. The design is represented by $v \times b$ incidence matrix $M$.

$$M = [m_{h \ell}], \text{ where } m_{h \ell} = \begin{cases} 1 & \text{if } h^{th} \text{ element of the set belongs to } \ell^{th} \text{ slab;} \\ 0 & \text{otherwise;} \end{cases}$$

$(k + (m + c))$ resilient anti-collision codes can be generated if the code vectors are assigned as bits in the column vector of $M$. For a given $v$ dimensional basis vector in a cluster, $n$ number of users can be accommodated in that cluster, where $n$ is derived from

$$n = \frac{r(v^2 - (cmr)^2)}{(k^2 - k)}$$

The procedure for constructing the fingerprint-IDS is explained with an example. Let us consider a cluster and a movie for constructing the fingerprint codes. In a cluster, let the set $\chi$ consists of $v$ orthogonal basis elements as $(\alpha, \beta, \gamma, \delta, \varepsilon, \varepsilon, \zeta)$ and $b$ subset slabs as $\{\alpha, \beta, \gamma\}, \{\beta, \gamma, \varepsilon\}, \{\gamma, \delta, \varepsilon\}, \{\delta, \varepsilon, \zeta\}, \{\varepsilon, \varepsilon, \alpha\}, \{\varepsilon, \zeta, \beta\}, \{\zeta, \alpha, \gamma\}$. In this given set, the value of $c = 1, m = 1, v = 7, b = 7, k = 3$ and $r = 1$. Each pair of basis elements appears together in exactly in only one slab for a movie, hence $\lambda = 1$. Therefore, the MC-BIBD design is $(7, 7, 1, 3, 1, 1, 1)$. The incidence matrix $M$ is constructed using this MC-BIBD design is
Each column of the matrix $\mathbf{M}$ has a unique code vector. From this matrix, 7 users can be accommodated. The values of the code vectors are mapped to code words ($\pm 1$) from $\{0,1\}$. The code words are constructed using linear combinations as follows:

$$
\begin{align*}
    w_1 &= +1 -1 -1 -1 +1 -1 +1 \\
    w_2 &= +1 +1 -1 -1 +1 -1 -1 \\
    w_3 &= +1 +1 +1 -1 -1 +1 +1 \\
    w_4 &= -1 -1 +1 +1 -1 -1 -1 \\
    w_5 &= -1 +1 -1 +1 +1 -1 -1 \\
    w_6 &= -1 -1 +1 -1 +1 +1 -1 \\
    w_7 &= -1 -1 -1 +1 -1 +1 +1 \\
\end{align*}
$$

where $w_i$ is a unique code word assigned to the $i^{th}$ user.

The above procedure for constructing the fingerprint-IDS is carried out in the authentication server of our p2pVoD architecture. Let us assume that only three users (User1, User2, and User3) in a cluster form a video chain. Initially, the authentication server authorizes these three users by constructing a Group Fingerprint ID (GFID).

The authentication server generates a session-ID, and identifies this GFID based on the generated session-ID. The session-ID (SID) constitutes a tuple which includes movie-ID, cluster-ID, and these three user-IDS with a timestamp.

The fingerprint-ID (FPID) for each user is chosen using the above code words. Thereby, we assign the code word $w_1$ to User1, $w_2$ to User2, and $w_3$ to User3 respectively. Therefore, FPIIDs of these three users are as follows:

$$
\begin{align*}
    \text{FPID}_1 &\leftarrow (+1 -1 -1 -1 +1 -1 +1) \\
    \text{FPID}_2 &\leftarrow (+1 +1 -1 -1 +1 -1 -1) \\
    \text{FPID}_3 &\leftarrow (+1 +1 +1 -1 -1 -1 +1) \\
\end{align*}
$$

The GFID for the SID ($\text{GFID}_{\text{sid}}$) is constructed by combining the FPIIDs of all three users. Therefore, the GFID$_{\text{sid}}$ is

$$
(+1 -1 -1 +1 -1 +1 +1 +1 -1 -1 +1 -1 +1 +1 -1 +1)
$$

Later on, the streaming server will embed the scrambled version of $\text{GFID}_{\text{sid}}$ in the video blocks, before transmitting it to the peers of three users. After embedding the scrambled version of $\text{GFID}_{\text{sid}}$ in the video blocks, it is transmitted to the peers in the
chaining fashion. Upon receiving the video blocks in the peer, the decoding procedure decodes them by using a bit map position vector, which are discussed in the next section.

4.3.3 EMBEDDING FINGERPRINT-IDS

The procedure for embedding the fingerprint-IDS is as follows. For each of the video chaining session, a unique GFID<sub>sid</sub> is constructed in the authentication server, as discussed in the previous section. The bits of original GFID<sub>sid</sub> are now scrambled and stored in another identifier GFID<sup>s</sup>sid along with its Scrambled Bit Positions (SBP). A copy of the original GFID<sub>sid</sub> and their Original Bit Positions (OBP) is retained. Note that, GFID<sub>sid</sub> facilitates in tracing the traitors, who are involved in an illegal act. Thereafter, the GFID<sup>s</sup>sid will be casted in the video blocks before transmitting it to the peers. To cast GFID<sup>s</sup>sid, a set of SDC in the video block are identified. Now, GFID<sup>s</sup>sid are embedded in the SDC coefficients of the video blocks. Before that, a Bitmap Position Vector (BPV) is used for storing the pair of OBP and SBP. Note that, at the time of tracing the traitors, these vectors are used for the autopsy. This type of tracing is known as non-blind tracing technique.

Continuing with the example, snapshot of the values in the vectors and its positions are shown in Figure 4.1.

<table>
<thead>
<tr>
<th>Vector</th>
<th>Values and its positions</th>
</tr>
</thead>
<tbody>
<tr>
<td>OBP</td>
<td>1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21</td>
</tr>
<tr>
<td>GFID&lt;sub&gt;sid&lt;/sub&gt;</td>
<td>+1 −1 −1 −1 +1 +1 +1 −1 −1 −1 −1 −1 +1 +1 +1 −1 −1 +1 +1</td>
</tr>
<tr>
<td>SBP</td>
<td>7 11 13 5 1 9 14 6 4 17 19 2 20 15 21 16 18 3 8 10 12</td>
</tr>
<tr>
<td>GFID&lt;sup&gt;s&lt;/sup&gt;sid</td>
<td>+1 −1 +1 +1 +1 −1 −1 −1 +1 +1 +1 +1 +1 −1 −1 +1 −1 −1</td>
</tr>
<tr>
<td>SDC</td>
<td>DC&lt;sub&gt;0&lt;/sub&gt;DC&lt;sub&gt;1&lt;/sub&gt;DC&lt;sub&gt;2&lt;/sub&gt;DC&lt;sub&gt;3&lt;/sub&gt;DC&lt;sub&gt;4&lt;/sub&gt;DC&lt;sub&gt;5&lt;/sub&gt; ...</td>
</tr>
<tr>
<td>BPV</td>
<td>(1,7)(2,11)(3,13)(4,5) ...</td>
</tr>
</tbody>
</table>

**Figure 4.1:** Snapshot of the Values in Vectors and its Positions.

The procedure to embed one bit of GFID<sup>s</sup>sid into the quantized SDC coefficient of 8 × 8 block is as follows: The value of DC coefficient of the j<sup>th</sup> block is taken from the frame of a video block, and one bit θ is taken from the GFID<sup>s</sup>sid. A Pseudo Random Number (PRN) is generated using a uniform distribution of [R<sub>1</sub>, R<sub>2</sub>], where R<sub>1</sub> and R<sub>2</sub> are random numbers, and PRN is the key for decoding the fingerprint information. The value of DC<sub>j</sub> from SDC is modified according to the equation

\[
DC_j' = \begin{cases} 
  \text{rnd} \left[ \PrN \cdot \text{rnd} \left( \frac{DC_j}{\PrN} \right) \right], & \text{if } \left( \text{mod} \left( \text{rnd} \left( \frac{DC_j}{\PrN}, 2 \right), 2 \right) = \theta \right) \\
  \text{rnd} \left[ \PrN \cdot \left( \text{rnd} \left( \frac{DC_j}{\PrN} \right) + Z_i \right) \right], & \text{if } \left( \text{mod} \left( \text{rnd} \left( \frac{DC_j}{\PrN}, 2 \right), 2 \right) \neq \theta \right)
\end{cases}
\] (4.3)
where,

\[
Z_j = \begin{cases} 
+1, & \text{if } \left( DC_j \geq \text{PRN} \cdot \text{rnd} \left( \frac{DC_j}{\text{PRN}} \right) \right) \text{or } \left( \text{rnd} \left( \frac{DC_j}{\text{PRN}} \right) = 0 \right) \\
-1, & \text{if } DC_j < \text{PRN} \cdot \text{rnd} \left( \frac{DC_j}{\text{PRN}} \right)
\end{cases}
\]

The value of \(Z_j\) is either +1 or −1, because it can minimize the absolute difference between the original \(DC_j\) coefficient and the modified \(DC'_j\) coefficient. If the value of PRN is larger, then the distortion rate for the modified \(DC'_j\) coefficient is greater. Hence, the value of PRN is chosen from a uniform distribution [0,1]. The \(\text{rnd}\) is a round off function. This procedure is repeated for embedding the remaining bits of GFID\(_{sid}\) into \(DC_j\) coefficients of SDC. The embedded GFID\(_{sid}\) in the video blocks are transmitted from the streaming server to the peers.

4.3.4 TRANSMISSION OF BPV AND PRN

The procedure for transmitting the embedded fingerprint version of video blocks from streaming server to the peers is dealt in detail under Chapter 7. To start with, before the transmission of the video blocks to the peers, the authentication server securely transmits the concerned users’ BPV range and the decoding fingerprint key PRN. At the peer, based on BPV range, each bit of GFID\(_{sid}\) of the concerned user’s FPID in the video blocks is complimented. By this process a unique fingerprint identifier for that user will be casted on the video blocks. This operation is internal to the system; the user may not able to either view the BPV, or change its positions in the video blocks.

For instance, continuing with the example, assume that if User1 receives the BPV range and the PRN from the authentication server. After the batching period (dealt in Chapter 7), the streaming server transmits the embedded fingerprint version of video blocks to the User1’s peer. From BPV range the User1’s FPID\(_1\) bit positions are identified. Accordingly, the bits of User1’s FPID\(_1\) are located at 5\(^{th}\), 12\(^{th}\), 18\(^{th}\), 9\(^{th}\), 4\(^{th}\), 8\(^{th}\) and 1\(^{st}\) positions of the SDC in the video blocks. The bits in this location are now complimented. After this, the video blocks stored in the User1’s peer will have a unique group fingerprint identifier GFID\(_1\) as:

\[-1 - 1 + 1 - 1 + 1 + 1 + 1 - 1 + 1 - 1 + 1 - 1 + 1 + 1 - 1 - 1 \]

Analogously, for User2 and User3 the GFID\(_2\) and GFID\(_3\) will be
and
\[ +1 - 1 + 1 + 1 + 1 - 1 - 1 - 1 + 1 + 1 - 1 - 1 + 1 + 1 + 1 - 1 - 1 \]
respectively.

### 4.4 IDENTIFICATION OF TRAITOR

After confiscating the pirated version of the video, the autopsy of it is done at the authentication server. Note that, before transmission of this video, the original bitmap position vectors are preserved at the authentication server. Also note that, as mentioned before, the non-blind technique is used for the autopsy. To identify the traitor, the value of GFID is extracted from the pirated video blocks using a non-blind technique, it means that the original SDC are available in the authentication server. The bit 0 from GFID in the video blocks can be easily extracted by using SDC. In order to decode the fingerprint-ID, PRN is needed, assertively it is available in the authentication server. The GFID\(_{sid}\) is decoded for some \(j^{th}\) block according to the equation

\[
0 = \text{mod} \left( \text{rnd} \left( \frac{DC_j}{PRN} \right), 2 \right)
\]

The procedure to identity the traitors is illustrated with respect to previous example, which is as follows. Suppose, if User1 indulges in betraying the trust of the system and involves illegally in distributing the video. Then, it is possible to identify User1 as a traitor by using the original GFID\(_{sid}\) at the authentication server. Every pirated version of the video will have a unique GFID\(_i\) of some \(i^{th}\) user from which the traitor can be identified. The authentication server will extract this GFID\(_i\) from the pirated versioned video, and identifies the deceitful traitor. In this trivial case, User1 is a traitor; to prove this; first the GFID must be extracted from the pirated video, and it must be unscrambled using the OBP vector. So that, the Extracted Fingerprint Identifier (EFPI) will be

\[-1 + 1 + 1 + 1 - 1 + 1 + 1 - 1 - 1 - 1 + 1 + 1 - 1 - 1 - 1 + 1\]

Now, this EFPI is averaged with the GFID\(_{sid}\), and the result is XORed with GFID\(_{sid}\) to obtain the traitor FPID\(_x\) as shown in Figure 4.2.
Figure 4.2: Shaded Area indicating that User1 is Traitor.

Figure 4.3: Shaded Area indicating that User1 and User2 are Traitors.

It can be observed from the Figure 4.2 that the bit positions of shaded area of User1 contains FPI\textsubscript{1}. Therefore, it can be inferred on the basis of convincing evidence that the User1 is the traitor.

In another case, suppose, if User1 and User2 collude together to attack on the embedded fingerprint-IDS in the video, then the pixel positions of User1’s video and the pixel positions of User2’s video will be averaged in the process of collusion. This will result in a new version of amalgamated fingerprint-IDS for that video. In this case, the EFPI will not give the correct FPI\textsubscript{1} of the traitors. Therefore, the Amalgamated Fingerprint Identifier (AFPI) of that video is extracted for the identification of traitors. The extracted AFPI from the pirated video block is

\[ -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 +1. \]

Now, this AFPI will be averaged with GFID\textsubscript{sid}, and the result is XORED with GFID\textsubscript{sid} to obtain the traitors FPI\textsubscript{x} as shown in Figure 4.3. It can be observed from Figure 4.3 that the shaded area contains FPI\textsubscript{1} and FPI\textsubscript{2} of User1 and User2 respectively. Therefore, it can be substantively concluded that User1 and User2 are traitors.
Figure 4.4: Shaded area indicating that User1, User2 and User3 are traitors.

In the worst case, suppose, if all three users collude together to attack on the embedded fingerprint-IDS in the video block. Analogously, the extracted AFPI for that video is

$$-1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1.$$  

After applying the procedure for identifying the traitors it can be observed from Figure 4.4 that User1, User2 and User3 have colluded and they all are traitors.

4.5 AUTOPSY AT THE AUTHENTICATION SERVER

The construction of fingerprint-IDS is carried out in the authentication server. Initially, the authentication server authorizes users by constructing a Group Fingerprint ID (GFID) whose peers are chained for the transmission. The authentication server generates a session-ID, and identifies this GFID based on the generated session-ID. The session-ID (SID) constitutes a tuple which includes movie-ID, cluster-ID, and user-IDS with a timestamp. For each of the video chaining session, a unique GFID<sub>sid</sub> is constructed in the authentication server. The bits of original GFID<sub>sid</sub> are now scrambled and stored in another identifier GFID<sub>s</sub><sup>sid</sup> along with its Scrambled Bit Positions (SBP). A copy of the original GFID<sub>sid</sub> and their Original Bit Positions (OBP) are retained at the authentication server. Note that, GFID<sub>sid</sub> facilitates in tracing the traitors, who are involved in copying of videos. Thereafter, the GFID<sub>s</sub><sup>sid</sup> will be casted in the video blocks before transmitting it to the peers. To cast GFID<sub>s</sub><sup>sid</sup>, a set of SDC in the video block are identified. Now, GFID<sub>s</sub><sup>sid</sup> are embedded in the SDC coefficients of the video blocks. Before that, a Bitmap Position Vector (BPV) is used for storing the pair of OBP and SBP. Note that, at the
time of tracing the traitors, these vectors are used for the autopsy. This type of tracing is known as non-blind tracing technique.

Whenever the user of the p2pVoD copies the content of the video from their peer’s storage unit then they would be violating video copyright. Also, when many users collude their copy of the video together then there is a possibility of tampering of fingerprint-IDS. Such users needed to be identified and should be named as the traitor. To identify such traitors, we need the copy of the video, which the user or users had illegally copied. After confiscating the pirated version of the video, the autopsy of it is done at the authentication server. The reason for the autopsy at the authentication server is, since, before transmission of this video, the original bitmap position vectors and other required parameters were preserved at the authentication server. Also note that, as mentioned before, the non-blind technique is used for the autopsy. To identify the traitor, the value of GFID is extracted from the pirated video blocks using a non-blind technique and compared with the original SDC that are available in the authentication server. The bit $b$ from GFID in the video blocks are extracted using SDC. In order to decode the fingerprint-ID, PRN is needed, which is also available in the authentication server. Since, all the required information are available at the authentication server, we use this information from the authentication server for the identification of the traitors.

4.6 QUALITY TRADEOFF BETWEEN EMBEDDING AND SIZE
Initially, the videos in the streaming server are preprocessed before transmitting to the peers. The format of movies stored in the streaming server is a standard MPEG format. The video blocks of the movie are herein used to distinguish between the AC components and DC components. Further, these DC components are preferred to embed the fingerprint-ID.

The DCT technique generates a representation of each block from a spatial domain to a frequency domain. Wherein, the pixels in each block are likely to be correlated. Hence, the resulting DCT coefficients consist of few larger values and many smaller values. The arrangement is based on increasing frequency of the coefficients and then it is stored in a vector. This kind of approximation is important because the low frequencies with the larger values are grouped at the rear part of the vector and high frequencies with the smaller values are grouped at the front part of the
vector. This vector will contain two types of coefficient; (a) DC coefficient and, (b) AC coefficients. The DC coefficient determines an average luminous, and the AC coefficients describe the variations around the DC coefficient in a block.

Different DCT coefficients have different persuade on the embedded fingerprint robustness (Zhu and Zhang, 2009). After embedding the fingerprint-IDS the visual quality of the original video must not change significantly. Therefore, the low and medium frequencies of AC coefficients are widely preferred locations for fingerprint placements, whereas DC coefficients are rarely preferred for embedding the fingerprint. However, DC coefficients are more suitable for embedding the fingerprint than any of the AC coefficients for at least two reasons. The first reason is that, the amplitude of the DC coefficient is much larger than any of the AC coefficients in a block. According to the visual system illumination coverage characteristics, the higher DC coefficient - the higher is its perceptual capacity. This means that having a higher perceptual capacity allows a larger fingerprint to be embedded without having much of perceptual distortion. Usually, the DC coefficient is tens of multiples, and even hundreds of multiples bigger than the biggest AC coefficient. The change of proportion in DC coefficient is less than AC coefficient, which means that DC coefficient has a higher perceptual capacity than the biggest AC coefficient. The second reason is that, during signal processing such as lossy compression, low-pass filtering, sub-sampling, Digital-to-Analog and Analog-to-Digital conversions; the DC coefficients have much less influence on the signal processing than the AC coefficients. Therefore, embedding the fingerprint-IDS in DC coefficients is more stable than embedding the fingerprint-IDS in AC coefficients.

In DCT blocks most of the higher energies are concentrated on the DC coefficients, and very few on the AC coefficients. In order to embed the fingerprint-ID, the distribution of DCT is used from the video blocks. In an 8x8 DCT block more than one bit can be embedded in it, but not all blocks are used for embedding the bits. A slight mutation in the quantized DC coefficient can cause major visual artifacts in the video playback sequences. However, in the texture regions, a smaller mutation in the quantized DC coefficient can be perceptually undetectable because of the masking effect of the texture. Therefore, only the texture-rich blocks are preferred for embedding the fingerprint-IDS. In our proposal, to identify a texture-rich block we have used an activity measurement of Texture Energy in a block. In this activity measurement, we have used DC energy of the quantized coefficient for selecting the
texture-rich block. Hence, the texture-rich DC coefficients with the values larger than threshold $T$ in the video blocks are selected for embedding the fingerprint-IDS.

### 4.7 BIBD DESIGNS

In order to evaluate the efficiency, effectivity and the performance of our MC-BIBD design, we have considered some of the existing BIBD designs from the literature, which are used for the video fingerprinting. First, we will introduce some of the existing BIBD designs with respect to our p2pVoD architecture, and then we will compare with our MC-BIBD design for the optimality test. Later on, we define the efficiency, effectivity and performance of each of the BIBD designs for the measurement of appropriateness to our p2pVoD architecture.

#### 4.7.1 GD-BIBD

Kang et al (2006) has proposed a new code generation algorithm for generating the fingerprint code using *Group Divisible Partially BIBD* (GD-BIBD). This code uses two associative classes on a group-divisible association scheme. The class defines the occurrence of pairing of elements within the group, and between the groups. These associative classes help in generating large number of fingerprint IDS. The code is then modulated using noise like pattern and embedded in the video. This code uses “ACC-AND” operation to identify the colluders after averaging attack. From a business and economic perspective, digital fingerprint have to manage a large number of users, probably more than thousands of users. The GD-BIBD is easy to implement, and also it can generate a greater number of unique fingerprint codes. But, the design is implemented based on the assumption of *number of colluders*. When the number of colluders is greater than the assumed number of colluders in the design then the GD-BIBD fails to identify the colluders. In our MC-BIBD design, it can identify more than the assumed colluders.

#### 4.7.2 TC-BIBD

Yang and Xu (2006) have proposed a new robust anti-collusion encoding scheme that combines *BIBD code with Turbo code* (TC-BIBD) for generating unique fingerprint codes. The turbo code is known as a parallel concatenated convolutional code. It succeeds in achieving pseudo random codes by combining the convolutional code with interleave. Initially, the turbo code is encoded in the fingerprint and then it is embedded in the video. To detect a traitor, the fingerprint is extracted by turbo iterative decoding process. The combination of the turbo code and the BIBD code...
shows a good performance in error correcting and also in collusion resistance. But, the scheme shortens the length of the fingerprint to some extent under a certain collusion resistance. This kind of fingerprint code can be easily generated but cannot trace the colluders effectively under common collusion. In our scheme, with the shortened codeword; it can effectively trace most of the colluders.

4.7.3 MR-BIBD

Seol and Kim (2006) have proposed a scalable fingerprint by spreading BIBD codes to cover larger number of codeword (M) with the repetition factor (R) (MR-BIBD). In this design, spreading is of two levels; in the first level the codeword is a direct spread spectrum, and in the second level this codeword is spread over M x R selected region of a video, thereby it increases the number of fingerprint codes. This scheme has stronger advantage over other schemes because it can directly control the number of fingerprint codes. In this scheme, for each user, the fingerprint code is composed of codeword and a Gaussian distributed random signal. The dimension of code vectors is increased accordingly to fit the length of the fingerprint code of the user. This code vector is embedded over M x R selected regions in a video. Basically, the fingerprint code is constructed by repetition (enlarged) and permutation (shuffled), so that the permutation sequence is unique to all users, and unknown to the attackers. Though, this design increases the scalability of the fingerprint codes, but decreases the perceptual quality of the video. Decorously embedding of the same fingerprint is repeated many times over the block of a video. But, this design pervades on the visual perceptibility. Hence, in our design the codeword is equally scaled for all users without increasing the size of the fingerprint code.

4.7.4 GO-BIBD

Yu et al (2010) has proposed a new Group-Oriented Fingerprint on BIBD code known as GO-BIBD. This design is composed of an outer code and an inner code, which is based on BIBD code. The inner code constitutes a group codes to a group of users. This inner code is orthogonal in different groups and non-orthogonal within the same groups. The outer code is Reed Solomon code which is used for encoding and decoding the fingerprint codes in the video. To trace a group, a correlation between the group code and the collusion code is calculated. If the correlation is higher than a certain threshold then the group is involved in the collusion. For the higher correlation value the collusion code is split into blocks of inner codes, and decoded to trace the inner codes in a group. To find the illegal users, the Reed Solomon decoding
algorithm is used to decode the outer codes. In this scheme, the proposal is to reduce
the fingerprint code length subjected to the condition that the collusion size and error
probability must be the same. The assumption is that, the users who have close
correlation or other kind of personal relationship have greater chances of colluding
together. This is because a group of users who take part in the collusion will have
their codeword orthogonal to different groups. However, in practice, this is not the
case because the users from different group can definitely collude together. In our
design, even if the users from different group collude together to tamper the
fingerprint codes, it can definitely identify each individual users who have involved in
the collusion.

4.8 ANALYSIS OF BIBD DESIGNS
The fingerprint coding system has different structures, and different methods for
generating the codeword as well as different principles to trace the colluders. We have
considered some of the existing fingerprint coding system such as GD-BIBD, TC-
BIBD, GO-BIBD and MR-BIBD, which were discussed in the Section 4.5. In order to
compare the characteristics of these fingerprinting codes with our proposed MC-
BIBD design, we need to evaluate the metrics to our p2pVoD system. For that, we
define some metrics that can provide an optimal way of selecting a best fingerprint
coding technique for our p2pVoD system.

In this evaluation metrics, we provide some common parameters that can simplify the
comparison among the fingerprint coding system. Common parameters are \( n, z \) and \( L \),
where \( n \) is the total number of users, \( z \) is the number of collusion, and \( L \) is the length
of the fingerprint code. The definition of the evaluation metrics of efficiency and
effectivity are:

**Efficiency** (\( \Phi \)): The efficiency is defined as the ratio between the number of users \( n \)
and the length of the fingerprint code \( L \) that can be supported.

\[
\Phi = \frac{n}{L}
\]

For the given number of collusion \( z \), the value of \( \Phi \) must be higher so that it can
support \( n \) number of users within the length of the fingerprint code \( L \).

**Effectivity** (\( \Lambda \)): The effectivity is computed as the ratio between the number of
collusion \( z \) and the number of users \( n \) that can resist against collusion attacks.

\[
\Lambda = \frac{z}{n}
\]
For the same code length $L$ among the fingerprint code, the value of $\Lambda$ must be higher so that it can resist against a larger collusion size.

<table>
<thead>
<tr>
<th>BIBD Design</th>
<th>Fingerprint Length($L$)</th>
<th>Number of Users($n$)</th>
<th>Collusion Size($z$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GD-BIBD</td>
<td>$\frac{v}{\lambda r}$</td>
<td>$\frac{vr}{(k - c)}$</td>
<td>$(k - c)$</td>
</tr>
<tr>
<td>TC-BIBD</td>
<td>$\frac{v^2}{k^2}$</td>
<td>$\frac{vr}{(k - c)}$</td>
<td>$(k - c)$</td>
</tr>
<tr>
<td>GO-BIBD</td>
<td>$\frac{v^2 - v}{r(k^2 - k)}$</td>
<td>$\frac{vr}{\lambda(k - c)}$</td>
<td>$(k - cr)$</td>
</tr>
<tr>
<td>MR-BIBD</td>
<td>$\frac{v}{}\lambda$</td>
<td>$\frac{v^2 - v}{k^2 - k}$</td>
<td>$(k - c)$</td>
</tr>
<tr>
<td>MC-BIBD</td>
<td>$v - \left( b \left( \frac{mc}{m+c} \right) \right)$</td>
<td>$\frac{r(v^2 - (cmr)^2)}{k^2 - k}$</td>
<td>$(k + (m + c))$</td>
</tr>
</tbody>
</table>

Table 4.1: Formulation of Length, Users and Collusion for the different Block Designs.

For our p2pVoD system, a high collusion resistance and easily implementable method are preferred for generating the fingerprint codes. Therefore, we have derived the efficiency and effectivity for both existing BIBD designs and proposed MC-BIBD design as shown in the Table 4.1 and 4.2. A higher efficiency and a higher effectivity support greater number of users within the length of fingerprint code, and also resist against the average collusion attacks. The performance of the fingerprinting system against the collusion attacks is determined based on the resistance of the fingerprint against the average collusion attacks.

<table>
<thead>
<tr>
<th>BIBD Design</th>
<th>Efficiency ($\Phi$)</th>
<th>Effectivity ($\Lambda$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GD-BIBD</td>
<td>$\frac{\lambda r^2}{(k - c)}$</td>
<td>$\frac{(k - c)^2}{vr}$</td>
</tr>
<tr>
<td>TC-BIBD</td>
<td>$\frac{k^2 r \lambda}{v(k - c)}$</td>
<td>$\frac{(k - c)^2}{vr \lambda}$</td>
</tr>
<tr>
<td>GO-BIBD</td>
<td>$\frac{r^2(k^2 - k)}{\lambda(k - c)(v - 1)}$</td>
<td>$\frac{\lambda (k^2 - 2ck + c^2)}{v}$</td>
</tr>
<tr>
<td>MR-BIBD</td>
<td>$\frac{v - 1}{\lambda(k^2 - k)}$</td>
<td>$\frac{k^3 - (c + 1)k^2 + ck}{v^2 - v}$</td>
</tr>
<tr>
<td>MC-BIBD</td>
<td>$\frac{r(v^2 - (cmr)^2)(m + c)}{(k^2 - k)(r(m + c) - b(mc))r}$</td>
<td>$\frac{m^2 + k(m + c) + 2mc + c^2}{v(m + c) - b(mc)}$</td>
</tr>
</tbody>
</table>

Table 4.2: Formulae of Efficiency and Effectivity for the different Block Designs.
The colluders having different marked copies of the same video can collude together to generate a pirated video. The pirated video will contain a composite version of DC components. Thereby, the traces of original fingerprint in the pirated video is removed or attenuated. For some \( j \)th block, the composite version of \( DC_j'' \) components after average collusion attack is as follows:

\[
DC_j'' = \frac{1}{k} \sum_{i \in s_k} DC_j' + d = \frac{1}{k} \sum_{i \in s_k} w_i + DC_j + d
\]

(4.5)

where all vectors have dimension as \( N \), \( k \) is the number of colluders, \( w_i \) is the code word of the \( i \)th user, \( s_k \) is the subset of colluders of size \( k \), \( s_k \subseteq \{1, 2, 3, \ldots, n\} \), \( n \) is the total number of users, \( d \) is a distortion vector having an independent and identical Gaussian Distribution \( \mathcal{N}\left(\frac{||w||}{k}, \sigma_d^2\right) \), where \( \sigma_d^2 = \frac{(||DC_j'||^2 - ||DC_j||^2)}{N} \), where \( N \) is the number of DC components. The Watermark to Noise Ratio (WNR) is defined as \( \text{WNR} = 10\log_{10}\left(\frac{||w||^2}{||d||^2}\right) \). The detection scheme identifies the colluders based on the observation on \( DC_j'' \). Because of the non-blind detection technique, the original \( DC_j \) can be subtracted from the \( DC_j'' \). Since, the basis vector has the property of orthogonality, it suffices to consider the correlator vector \( T_N \) to perform the detection as

\[
T_N(i) = \frac{(DC_j'' - DC_j)^T w_i}{\sqrt{||w_i||^2}}
\]

(4.6)

The performance criterion of the fingerprint is based on three parameters a) false negative, b) false positive, and c) true positive. A false negative is defined as the failure of detector to detect any of the colluders. A false positive is defined as the success of detector to perceive an innocent user as a colluder. A true positive is defined as the success in detecting true colluders. To evaluate the performance of the fingerprint, the detector must minimize the probabilities of a false negatives \( P_{fn} \) and false positives \( P_{fp} \), and must maximize the probability of true positives \( P_{tp} \). In general, both \( P_{fn} \) and \( P_{fp} \) must be exceptionally low, and \( P_{tp} \) must be sufficiently high. Therefore, a detector must have the capability of capturing greater number of colluders by maximizing the detection and minimizing the concealment. Hence, the correlator \( T_N \) must be compared with a threshold \( h \). The value of threshold \( h \) is
determined by the length of DC component, total number of users $n$, number of colluders $k$, and the WNR. Therefore, $h$ is expressed as

$$h = \frac{\left(\frac{||w||^2}{k} - \log\left(\frac{1}{n-k}\right)\right)}{\sigma^2}$$

To find all colluding fingerprints that exceed the threshold $h$, a correlator $T_N$ is given by $i = \arg_{i=1}^{n}\{T_N(i) \geq h\}$, where $i$ is the set of indices of colluders. To evaluate the performance of the fingerprint, the captured set of colluders $\hat{i}$ must have a lower probability of $P_{fn}$ and $P_{fp}$, and a higher probability of $P_{tp}$. The value of $P_{fn}$, $P_{fp}$, $P_{tp}$ is determined as follows:

$$P_{fn}(T_N(i)|\vec{s}_k) = \begin{cases} \Pr(i \cap \vec{s}_k) & \text{if } \exists \ i \ in \ \vec{s}_k \\ \phi & \text{otherwise} \end{cases}$$

(4.7)

$$P_{fp}(T_N(i)|s_k) = \begin{cases} \phi & \text{if } \forall \ i \ in \ s_k \\ \Pr(i \cap s_k) & \text{otherwise} \end{cases}$$

(4.8)

$$P_{tp}(T_N(i)|s_k) = \begin{cases} 1 & \text{if } \forall \ i \ in \ s_k \ not \ in \ \vec{s}_k \\ \left(1 - P_{fn} * P_{fp}\right) & \text{otherwise} \end{cases}$$

(4.9)

where $\vec{s}_k$ is the subset of non-colluders of size $(n-k)$. The resistance of the fingerprinting is analyzed based on the above equations for the average collusion attack model. A sufficiently high $P_{tp}$, and exceptionally low $P_{fn}$ and $P_{fp}$ are required to make a fingerprinting resistant to collusion attacks.

4.9 SIMULATION

In this simulation, we evaluate the efficiency, effectiveness and performance of the fingerprint codes. In order to assess our proposed MC-BIBD design, we have also considered some of the existing schemes such as TC-BIBD, GO-BIBD, GD-BIBD and MR-BIBD. The purpose of considering these schemes in the simulation is to compare the optimality, and to commend the best fingerprint coding scheme for our p2pVoD system.

The topology used for the simulation is similar to our p2pVoD architecture. The frames in the video blocks of the video were coded using MPEG coding technique with different scaling factor $Q_p$ to achieve different visual quality for embedding the fingerprint-IDS. The value of the threshold $T$ is obtained as the function of the scaling factor $Q_p$ for the different video blocks.

The visual quality of the fingerprinted version of the video is measured as Peak Signal to Noise Ratio (PSNR). The PSNR is calculated by taking the logarithmic ratio
of maximum amplitude of energy in DC components of the frame and Mean Square Error (MSE), where MSE is taken as the square of the differences between the DC coefficients in the original frame and the DC' coefficients of the fingerprinted version of the frame. The PSNR are computed as follows

$$\text{PSNR}[dB] = 10 \log_{10} \left( \frac{(2\eta - 1)^2}{\text{MSE}} \right)$$

(4.10)

where $\eta$ is the number of bits per frame of the original frame and

$$\text{MSE} = \frac{1}{M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} [DC_{ij} - DC'_{ij}]^2$$

(4.11)

where $M \times N$ is the block size. A larger value of PSNR means that the fingerprinted version of the video blocks preserves the quality of the original video.

Figure 4.5: PSNR Comparison for different Block Design.

The parameters and the default values taken for the simulation are given in Table 1.1. The procedure followed for generating and embedding fingerprint-IDS of our proposed MC-BIBD block design is given in Section 4.3. However, for other types of block designs, a brief introduction is given in the Section 4.5. But, for the purpose of the simulation, we have considered the relevant procedure for generating the fingerprint-IDS and followed relevant technique for embedding the fingerprint-IDS in
the video blocks. The simulation was executed for several trails, and the result shown is an average of all trails carried out in all five clusters.

The results of the PSNR for the video sequences are shown in Figure 4.5. In order to determine the quality of visual perception after embedding the fingerprint-IDS in the video, we have compared the PSNR with different block designs. It can be observed from Figure 4.5 that the average PSNR values are large for GO-BIBD, MC-BIBD, TC-BIBD and GD-BIBD. The average PSNR value for the MR-BIBD is less when compared to other block designs. This is because of the larger codeword of MR-BIBD that are embedded in the videos.

| Common Parameters and their Default values : |
| $v = 120, b = 16, r = 2, k = 15, \lambda = 1, m = 10, c = 5$ |

<table>
<thead>
<tr>
<th>BIBD Design</th>
<th>Block Design</th>
<th>$n$</th>
<th>$L$</th>
<th>$z$</th>
<th>$(\Phi)$</th>
<th>$(A)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>GD-BIBD</td>
<td>$(v, b, r, k, \lambda_1, \lambda_2)$</td>
<td>17</td>
<td>60</td>
<td>11</td>
<td>0.3</td>
<td>0.6</td>
</tr>
<tr>
<td>TC-BIBD</td>
<td>$(v, b, r, k, \lambda, 1)$</td>
<td>17</td>
<td>64</td>
<td>10</td>
<td>0.2</td>
<td>0.5</td>
</tr>
<tr>
<td>GO-BIBD</td>
<td>$(v, r, k + 1, \lambda, 1)$</td>
<td>17</td>
<td>34</td>
<td>05</td>
<td>0.5</td>
<td>0.2</td>
</tr>
<tr>
<td>MR-BIBD</td>
<td>$(v, k, \lambda)$</td>
<td>68</td>
<td>120</td>
<td>10</td>
<td>0.5</td>
<td>0.1</td>
</tr>
<tr>
<td>MC-BIBD</td>
<td>$(v, b, r, k, \lambda, m, c)$</td>
<td>41</td>
<td>66</td>
<td>30</td>
<td>0.6</td>
<td>0.7</td>
</tr>
</tbody>
</table>

Table 4.3: Results of the Efficiency and Effectivity of different Blocks Designs.

Generally, whenever the video is played normally there will not be visual degradation that could be sensed, because most of the fingerprint-IDS were embedded in the text-rich block of the picture in the video. The evaluation of the efficiency and effectiveness of the BIBD designs with respect to the number of users, length of fingerprint code, and the number of colluders are given in the Table 4.3.

The evaluation formulae of the efficiency, effectiveness and performance of the fingerprint codes for TC-BIBD, GO-BIBD, GD-BIBD, MR-BIBD and MC-BIBD schemes were given in the Section 4.6. These evaluation formulae are applied in the simulation model for the analysis and exertion of the results.

The results of the efficiency are depicted in Figure 4.6 and Figure 4.7. Figure 4.6 can be observed from the perspective of number of users against the fingerprint code length only. TC-BIBD, GD-BIBD, and GO-BIBD supports up to 17 users with efficiency of 0.2, 0.3 and 0.5 respectively, where as MR-BIBD supports up to 68 users with an efficiency of 0.5. The MC-BIBD supports up to 41 users with an
efficiency of 0.6. Hypothetically, for any given fingerprint code length, if it supports greater number of users then it cannot resist against collusion of many colluders. Intuitively, for a good design, a fingerprint code length should be able to support
many users with capability of resisting against collusion attacks. Though, TC-BIBD, GD-BIBD, and GO-BIBD show the support for optimal number of users, the fingerprint code bits are not optimally utilized. The MR-BIBD’s efficiency is better and supports greater number of users than the required support. While other BIBD designs obviously cannot resist even a smaller number of colluders. Hence, MC-BIBD design is optimal in terms of number of users and the number of colluders.

Figure 4.7 can be viewed from the perspective of fingerprint code length only. The results of GO-BIBD and MR-BIBD show good efficiency of 0.5 each. The needed fingerprint code length size for MR-BIBD is 120 bits and for GO-BIBD is 34 bits is good. But, the drawback of GO-BIBD is that it cannot support large number of users, and MR-BIBD cannot resist large number of colluders. Although, TC-BIBD and GD-BIBD have a good code lengths of size 64 bits and 60 bits respectively, lacks in efficiency, and has only 0.2 and 0.3 respectively. However, MC-BIBD has a good fingerprint code length of size 66 bits with a higher efficiency of 0.6. Hence, MC-BIBD scheme is the best in utilizing the fingerprint code length by allowing greater number of users with the collusion resistance.

Figure 4.8: Probability of Effectivity versus Supported Number of Users.
The result of the effectiveness is depicted in Figure 4.8 and Figure 4.9. Figure 4.8 can be viewed from the perspective of number of colluders against the number of users only. The GD-BIBD and TC-BIBD have good effectivity of 0.6 and 0.5 respectively; they cannot support more than 17 users. The MR-BIBD and GO-BIBD have a poorer effectivity of 0.1 and 0.2 respectively and although MR-BIBD supports up to 68 users, this cannot be considered as optimal because it cannot resist against even a smaller number of collusion attacks. The MC-BIBD shows a very good effectivity of 0.7 with the support of 41 users that can resist up to 30 collusion attacks.

Figure 4.9 can be viewed from the perspective of number of colluders against the length of the fingerprint code only. The MR-BIBD and GO-BIBD shows very poor effectivity of 0.1 and 0.2 respectively, though GO-BIBD has a minimum of 34 bits of fingerprint code length size, but has a lesser resistance against the collusion attacks. The TC-BIBD and GD-BIBD have a good effectivity of 0.5 and 0.6 respectively with the fingerprint code length sizes of 64 bits and 60 bits respectively. However, MC-BIBD has much better effectivity of 0.7 with fingerprint code length size of 66 bits, even though the fingerprint code length size of MC-BIBD is slightly higher than those of GD-BIBD and TC-BIBD, MC-BIBD still has the best effectivity against the collusion attacks.
The results of the effectivity and efficiency from the perspective of maximum tolerated collusion size are depicted in Figure 4.10 and Figure 4.11. It can be observed from the Figure 4.10, that GO-BIBD and MR-BIBD have efficiency of 0.5 and 0.5.
respectively, but the maximum number of colluders that it can resist is only 5 and 10 respectively. The TC-BIBD and GD-BIBD have poor efficiency of 0.2 and 0.3 respectively, though each can tolerate maximum of 10 colluders. The MC-BIBD has a very good efficiency of 0.6 and it can tolerate up to 30 colluders. Therefore, MC-BIBD is the best design that supports resistance against maximum number of colluders than any other designs.

It can be observed from Figure 4.11 that TC-BIBD and GD-BIBD have good effectivity of 0.5 and 0.6 respectively, but can resist only 10 colluders each. The MR-BIBD and GO-BIBD have the least effectivity of 0.1 and 0.2 respectively, though MR-BIBD can resist up to 10 colluders. The MC-BIBD has a very good effectivity of 0.7 and can resist up to 30 colluders. Therefore, the MC-BIBD is the best design in terms of effectivity and efficiency that can tolerate maximum number of colluders than any other designs.

![Figure 4.12: Number of Users supported versus Maximum Tolerated Collusion Size.](image-url)

Figure 4.12: Number of Users supported versus Maximum Tolerated Collusion Size.
Figure 4.13: Probability of False Positive versus WNR.

Figure 4.12 depicts the maximum number of users that can be supported against the maximum number of colluders. It can be observed from Table 4.3 that MR-BIBD supports up to 68 users, but can resist only 10 colluders. The GO-BIBD, TC-BIBD and GD-BIBD support up to 17 users each but can resist only 5, 10 and 11 colluders respectively. Though, MC-BIBD supports only 41 users, which is less than MR-BIBD and greater than GO-BIBD, TC-BIBD and GD-BIBD, but has a greater resistance up to 30 colluders. Definitely, the MC-BIBD design is the best design in terms of efficiency, effectivity and resistance against maximum number of collusion attack.

The performance of the fingerprint schemes are evaluated against the resistance of number of colluders. The procedure of performance evaluation used in the simulation model is described in detail in Section 4.6. In general, the performance of the fingerprint scheme must minimize the probability of false negative \( P_{fn} \) and false positive \( P_{fp} \), and should maximize the probability of true positive \( P_{tp} \). Figure 4.13 and Figure 4.14 can be viewed from the perspective of performance of the design against average collusion attacks only. It can be observed from Figure 4.13 that MR-BIBD is high in false negative because the detector detects many innocent users as the colluders; this is due of the fingerprint length and the support given to greater number of users. As it can be observed from the Table 4.3, MR-BIBD supports 68 users from...
120 bits of fingerprint code length with maximum resistance of 10 colluders. Hence, the probability of accusing the remaining 58 innocent users as colluders is very high. Analogously, it can be observed from Table 4.3, the GO-BIBD, GD-BIBD and TC-BIBD act at an average performance rate during the collusion attacks. Because each of these designs supports maximum of 17 users each from the fingerprint code length of 34, 60 and 64 bits respectively, and can resist up to 5, 10 and 11 colluders respectively. Hence, the probability of accusing remaining innocent users as colluders is less. It can also be observed from Table 4.3 that the performance of MC-BIBD is good during the average collusion attacks. The fact of supporting maximum of 41 users from the fingerprint code length of 66 bits and resisting up to 30 colluders is an amazing result. In spite of supporting greater number of users the MC-BIBD shows the best performance in resisting maximum number of colluders. But, MC-BIBD block design is constructed taking into the consideration the current

![Figure 4.14: Probability of False Negative versus WNR.](image)

streaming videos and the prevailing cluster. There are many possibilities of colluders colluding together from different clusters which can underperform the performance of MC-BIBD. For this kind of case, earlier to the transmission of video blocks from the streaming server, the authentication server would stamp a timestamp with a user group fingerprint ID in the video blocks to trace the colluders.
It can be observed from the Figure 4.14 that the performance of MR-BIBD is very low. There are more chances of concealing colluders as genuine users because the possibility of identifying the colluders is limited to only 10. Similarly, the performance of GO-BIBD, TC-BIBD and GD-BIBD are moderate and concealing of colluders is reduced. The performance of MC-BIBD is good and does not camouflage much on the genuine users.

![Figure 4.15: Probability of True Positive versus WNR.](image)

Finally, Figure 4.15 shows the optimal performance of the BIBD designs. It can be observed from Figure 4.15 that MC-BIBD performs well against the average collusion attacks than other designs. The detection of culprit colluders is high in MC-BIBD and, in case of TC-BIBD and GD-BIBD is moderate. The detection of culprit colluders is deteriorated in case of GO-BIBD and MR-BIBD. Therefore, we conclude that the MC-BIBD is the optimal BIBD design in terms of efficiency, effectivity and performance for our p2pVoD system.

4.10 SUMMARY

In this chapter, a modified BIBD design is proposed for our p2pVoD system that strongly resists the average collusion attacks. We have discussed in detail the procedure to construct the fingerprint-IDS as well as the procedures for identification of DC components, embedding of fingerprint-IDS in the video blocks, and
transmission of vectors. These procedures are dealt with examples for better understanding. The proposed MC-BIBD design is compared with other existing designs such as MR-BIBD, GO-BIBD, TC-BIBD and GD-BIBD for the appropriateness to our p2pVoD architecture. We have also evaluated these schemes for the efficiency, effectivity, and performance, so that the best design can be chosen for our p2pVoD architecture. In the evaluation, our MC-BIBD design stands justifiable for our p2pVoD architecture than any other BIBD designs. In the following chapter, we shall discuss about the creation of video chaining.