Chapter 5

Efficient Image on Demand System

Digital Rights Management is a rapidly growing area of research that provides solutions towards several aspects of secure data communication [93, 94, 95, 96]. It includes system-level key exchange protocols, signal processing and encryption algorithms that make contents unusable for unauthorized parties. Some major applications designed with the help of DRM techniques are copyright protection, authentication and CAS.

In CAS, multimedia content can be shared following certain policies; a thumbnail or a low-resolution version of the content can be made available for free and the user has to pay in order to see the high quality content. One major CAS based system is Image On Demand, where a user browses a database of multimedia files in order to retrieve the content of interest. In a typical IOD scenario, a low-resolution image can be quickly downloaded from image databases in order to select the desired content, which can be purchased in a higher resolution version later. Figure 5.1 is a block dia-

89
gram representation of the general methodology adopted for any IOD based systems.

In this chapter, we present a novel scheme that can efficiently be adapted in such a scenario.

5.1 Introduction

Most of the existing conditional access based systems follow a standard methodology. The service provider shares two copies for a single information (specifically image for this work). One copy is a low-resolution version which is shared in the public domain for preview purpose. The other one is a high-resolution version to be provided to the customers through a secure channel on demand (after payment). We analyze the images in the DCT domain and note that polynomials of suitable degree, representing the sorted DCT coefficients together with original index locations, can uniquely represent an image. The DCT coefficients are sorted by its magnitude, which results into two set of data: (a) Sorted DCT coefficients and (b) An array which contains their original index locations. We show that the distribution of values (DCT
Index locations) in the generated array is significantly different for various images, and we exploit this to design an efficient CAS based scheme. The amount of private data, which a service provider needs to transmit through a secure channel to the customers on demand, is reduced significantly by our technique. This reduction in transmitted data makes the system apt for real-time secure applications.

Many websites with huge image databases are available today [97, 98, 99]. Most of these existing websites target a specific segment of customers who require very high-resolution images. They usually share a visible watermarked image of low quality for the purpose of preview and send the high-resolution version through a secure channel when paid for the same. The preview versions are generally freely downloadable and are acceptable for the purpose of recognizing the content.

One major technique being used to realize CAS based systems is selective encryption. Suitable portion and size of the actual data (image in this case) is scrambled in the process of selective encryption. It is performed in such a way that the image rendered from the rest of the ‘in-the-clear portion’ (i.e., the segment of the data that is not scrambled) is imperceptible to a significant fraction of the consumers [14]. In CAS like systems, one does not need to obscure the data completely to support rights management but make it such that the rendered data is indiscernible.

The majority of existing encryption standards such as DES and AES have been developed for independent and identically distributed i.i.d data sources. However, multimedia data are typically non i.i.d, so applying encryption on the pixel domain is not at all feasible due to computational complexity, and its in-suitability for the real-time applications. But, transform domain is preffered [100, 101, 102] for applying selective encryption approach since we can exploit the fact that the data are represented as a sequence of approximately i.i.d samples. This samples contribute unequally to the quality of
the reconstructed signal. So it is sufficient to target only a specific portion of the coefficients as well as provide advanced functionalities for conditional access [103, 104].

Motivation: Existing IOD schemes keeps at least two copies of a particular image: one for preview purpose and the other is the high-resolution version which is sent separately through secure channel when customer pays for the same. This is one of the major drawbacks, since to follow this protocol, the amount of storage required is quite high. Another drawback is that, the huge amount of data in the form of high-resolution image needs to be sent through the secure channel. This increases the computational and communication overhead at real-time. In this chapter, we resolve the drawbacks mentioned above.

Real-time data transmission and computations should be minimized to the best possible extent to make the above mentioned system simple and efficient. At the same time, quality and security aspects should also be ascertained. Security aspects include data integrity, authentication etc. [105, 53, 106, 107]. Efficient and secure key management protocols are also necessary to exchange keys between service providers and users. Some of the risk aspects related to piracy are explained and solutions have been proposed through a two-period model in [94]. A novel privacy preserving content distribution mechanism for digital rights management without relying on the trusted third party assumption has been proposed in [96]. Authors have used some simple primitives such as blind decryption and one way hash chain to avoid the assumption of trusted third party. Delgado et al. in [93] proposes a modular architecture which provides digital rights and privacy policies management features. It can be integrated by invoking some web service calls, depending on which services are needed. Several similar protocols and scheme exist, such as those described in [95, 108, 109].

Result: In this chapter we propose a scheme for image on demand in which
we store only single copy of the data to represent an image in the database. It leads to significant reduction in the storage requirement. Figure 5.2 gives a broad overview of the proposed IOD based system. An image is transformed into the DCT domain. After a certain preprocessing methodology (explained in Section 5.3) we divide the transformed data of each image into two subsets namely private significant data and free publicly shared data. We construct a low quality version from one subset (free publicly shared data subset) and keep the other subset (private significant data subset) private. The low quality version is sufficient to preview but is not usable in any sense to any group of customers. We provide the private significant data subset to construct a high-resolution version only when the customer demands and pays for the same. The private significant data subset of an image which is kept secret, constitutes of an optimal fraction of the total data required to represent a single image and the remaining data related to the image is publicly downloadable for preview purpose. In general the size of private significant data subset is significantly less than the free publicly shared data subset for any specific image (explained in Section 5.5.1). So the amount of data required to be sent through the secure channel in real-time is significantly less than most of the existing schemes. One time installation of free image viewer
application is required at the customer end which is inline with the already existing similar kind of service providers. We focus here on gray level images while the scheme can be generalized for colour images and video streams.

An image $I$, in the DCT domain can be efficiently represented by a set of polynomials of suitable degree together with the corresponding index location matrix $\Pi_{I_d}$ [12, 13]. In this work we show that $\Pi_{I_d}$ is the most significant attribute and can be used as a unique descriptor for any image while on the other hand the polynomials that approximate the actual DCT coefficients, as such have less significance. We utilize $\Pi_{I_d}$ to develop a new IOD system. We would also like to claim that to the best of our knowledge that there is no existing work which utilizes $\Pi_{I_d}$ for designing any application.

Organization of the chapter: A brief survey of existing selective encryption techniques which could possibly be used for various commercial applications is given in Section 5.2. We present the methodology in Section 5.3. Detailed description of the proposed IOD scheme is given in Section 5.4. Experimental results, interpretations, analysis of efficiency and robustness of our scheme are given in Section 5.5 followed by conclusion in Section 5.7.

5.2 Existing Selective Encryption Schemes

Selective encryption is a technique to save computational complexity or enable interesting new system functionality by only encrypting a portion of a compressed bitstream while still achieving adequate security. Although suggested in a number of specific cases, selective encryption could be much more widely used in consumer electronic applications ranging from mobile multimedia terminals through digital cameras. Some of the schemes based on the concept are presented next.

Two efficient approaches to conceal Regions Of Interest (ROIs) based
on transform-domain or code-stream-domain scrambling have been proposed in [110]. In the first technique, the sign of selected transform coefficients is pseudo-randomly flipped during encoding. In the second method, some bits of the code-stream are pseudo-randomly inverted. An index-based selective audio encryption scheme for Wireless Multimedia Sensor Networks (WM-SNs) is presented in [111]. It protects data transmissions by incorporating both resource allocation and selective encryption based on Modified DCT (MDCT). In this proposed scheme, the audio data importance is leveraged using the MDCT audio index, and wireless audio data transmission proceeds with energy efficient selective encryption.

A partial encryption scheme in wavelet domain based on secure encryption principles with respect to the existing attacks (cryptographic attack, replacement attack and statistical model based attack) is proposed in [112]. Schemes have been proposed in [113] to optimize the energy, distortion, and encryption performance of video streaming in WSNs. Two significant contributions have been claimed. First, a channel-aware selective encryption approach is proposed to minimize the extra encryption dependency overhead at the application layer. Second, an Unequal Error Protection (UEP)-based network resource allocation scheme is proposed to improve the communication efficiency at the lower layers. An integrated approach of fingerprinting and encryption is proposed in [114] where keys of different receivers helps in the fingerprinting aspects. A joint encryption and compression framework based on selective bit scrambling, block shuffling and block rotation of the transform coefficients and motion vectors of video is proposed in [112].

A fine-grained access control mechanism based on selective encryption is proposed in [115]. Using this approach, the owner of a file specifies access control policies over various byte ranges in the file. The separate byte ranges are then encrypted and signed with different keys. Users of the file only receive the encryption keys for the ranges they are authorized to read and signing
keys for the ranges they are authorized to write. An approach to reduce the computational cost of multimedia encryption while also preserving the properties of compressed video is presented in [116]. A hardware-amenable design of the proposed algorithms makes them suitable for real-time embedded multimedia systems. This approach alleviates the need for additional hardware for encryption in resource-constrained scenarios and can be otherwise used to augment existing encryption methods used for content delivery on the internet or in other applications.

A novel method allowing the protection of the newly emerging video codec HEVC (High Efficiency Video Coding) is presented in [117]. Visual protection is achieved through the selective encryption of HEVC-CABAC binstrings in a format compliant manner. To protect the privacy in the CCTV video, an encryption scheme for region of interest of H.264 video based on flexible macroblock ordering and chaos is proposed in [118]. The proposed scheme can effectively protect the private information of H.264 video and, therefore, can strike a good balance among the security, encryption efficiency, and coding performance.

Cryptanalysis of various discrete orthogonal transforms have been carried out in [100]. They have also proposed a DCT-based scheme which significantly improves the security of the scrambler. A design which selectively encrypts the fixed-length codewords of MPEG-video bit streams is proposed in [119]. Strict size-preservation, on-the-fly encryption and multiple perceptibility are the features supported in this design.

Before proceeding further, let us now describe our methodology in detail.
5.3 Our Methodology

5.3.1 Polynomial representation of an image

Any digital image $I$ can be uniquely represented by a set of polynomial equations $P_i$’s and the corresponding permutation of actual index locations $\Pi_{I_d}$ of the DCT transformed image $I_d$. Details of the representation scheme are given next.

Step 1: Image Representation

- A given digital image $I$ can be interpreted as a 2-dimensional matrix in spatial domain.
- Let $N$ and $M$ be the width and height of $I$.
- We perform $N \times M$ DCT transform on $I$ to form $I_d$ also of size $N \times M$. This is given as

$$I_d(k_1, k_2) = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} x_{n,m} \cos \left( \frac{\pi}{M} (m + \frac{1}{2}) k_2 \right) \cdot \cos \left( \frac{\pi}{N} (n + \frac{1}{2}) k_1 \right)$$

$$I_d := DCT_{N \times M}[I]$$

- Scan $I_d$ in zig-zag manner (similar to the one used in baseline JPEG compression technique [7]) and store the resultant array as $Z_{I_d}$ also of length $N \times M$.

$$Z_{I_d} := SCAN_{zig-zag}[I_d]$$
CHAPTER 5. EFFICIENT IMAGE ON DEMAND SYSTEM

- Now sort the DCT coefficients of $Z_{ld}$ in descending order and store in an array $A_{ld}$.

- Simultaneously store the initial index locations corresponding to each DCT coefficient of $A_{ld}$ in an array $\Pi_{ld}$.

$$A_{ld} := \text{SORT}_{\text{Desc.}}[Z_{ld}[i]; 1 \leq i \leq N \times M]$$

$$\Pi_{ld} := [\Pi_{ld}(i) : A_{ld}[\Pi_{ld}(i)] = Z_{ld}[i]]$$

$\Pi_{ld}$ will be required during recovery of the image.

**Step 2: Polynomial Approximation**

- Now, partition the values of $A_{ld}$ contiguously in $q$ different parts $A_{ld_1}, A_{ld_2}, \ldots, A_{ld_q}$ leaving aside the DC value. Note that each of $A_{ld_1}, A_{ld_2}, \ldots, A_{ld_q}$ is also a sorted array.

$$A_{ld_1} = [A_{ld}[j] : 2 \leq j \leq \frac{NM}{q}]$$

$j$ starts from 2 for $A_{ld_1}$ since DC value i.e. $A_{ld}[1]$ is not included for polynomial approximation process.

- For $2 \leq i \leq q$ :

$$A_{ld_i} = [A_{ld}[j] : (i - 1)\frac{NM}{q} + 1 \leq j \leq i\frac{NM}{q}]$$

$A_{ld_1}, A_{ld_2}, \ldots, A_{ld_q}$ are also sorted arrays.

- Now corresponding to each $A_{ld_i}$, we fit a polynomial of degree $\theta_i$, such that the Mean Square Error (MSE) is minimized. It is clear that as we increase the degree $\theta_i$ of the polynomial, the mean square error is less. However, since the data in $A_{ld_i}$’s are monotonically decreasing,
very good approximation can be obtained by using polynomials with moderate degrees. Thus, we get a series of polynomials $P_1, P_2, \ldots, P_q$ approximating the DCT coefficient matrix $I_d$. Apart from the polynomials, we also have the DC value $\eta = I_d[1,1]$ which is kept intact.

$$P_i = Poly_{\theta_i}(A_{I_{di}}), \quad \forall 1 \leq i \leq q; \quad \text{deg}(P_i) = \theta_i$$

$Poly_{\theta_i}$ is used to denote approximation by a polynomial of degree $\theta_i$. We keep the values of $\theta_i$'s constant for all $i$ and call it $\theta$. The MSE $\phi(I^P, I)$ of any image $I^P$ (approximation of $I$ in this case) with respect to another image $I$ is calculated by (5.1).

$$\phi(I^P, I) = \frac{1}{NM} \sum_{u=1}^{N} \sum_{v=1}^{M} [I^P(u,v) - I(u,v)]^2$$ \hspace{1cm} (5.1)

**Step 3: Image Reconstruction through Polynomials**

- To reconstruct an approximated image $I'$ of size $N \times M$, we first extract data from the corresponding $P_i$'s (DCT polynomials).

$$A'_{I_{di}} = [P_i(k); \quad 1 \leq k \leq \frac{NM}{q} - 1],$$

$$A'_{I_{di}} = [P_i(k); \quad 1 \leq k \leq \frac{NM}{q}], \quad \text{for} \quad 2 \leq i \leq q.$$  

The size of first segment is $\frac{NM}{q} - 1$ instead of $\frac{NM}{q}$ since DC coefficient is not included.

- Then we place them back into their proper locations in the DCT matrix
using the index matrix $\Pi_{I_d}$ to get $Z'_{I_d}[i]$.

$$
A'_{I_d} = \{\eta\} \bigcup_{i=1}^{q} A'_{I_d_i}
$$

$$
Z'_{I_d} := [A'_{I_d}[\Pi_{I_d}(k)]: 1 \leq k \leq NM]
$$

Above step actually places the DCT coefficient back to their original locations where it was present before sorting.

- Now we perform inverse zig-zag scan on $Z'_{I_d}$ to get $I'_d$.

$$
I'_d := \text{SCAN}_{\text{inv-zig-zag}}[Z'_{I_d}]
$$

- Finally $N \times M$ Inverse DCT (IDCT) is applied to get an image $I'$ of size $N \times M$.

$$
I' := \text{IDCT}(I'_d)
$$

For example, if we take $q = 16$ for a $256 \times 256$ gray level image of Lena and approximate each $A_{I_d_i}$ by a polynomial $P_{i}$ of degree $\theta = 15$ then Figure 5.3(b) is the resultant image of Lena which is visually indistinguishable and has PSNR value as high as 55 dB w.r.t. the original Lena image.

### 5.3.2 Significance of $\Pi_{I_d}$

$A_{I_d}$ follows a generic pattern and may be considered almost similar for most images. Even if $A_{I_d}$ is completely known then also it is computationally infeasible to recover the corresponding $I'$ without the knowledge of $\Pi_{I_d}$.

The above statement is justified by certain experimental analysis whose
details are given below.

- Let $I, J$ be two distinct images. Let $A'_{Id}$ be the set of DCT coefficients of $I$ which have been approximated by polynomials of suitable degree. We know that along with $A'_{Id}$, we require the set of DCT index locations $\Pi_{Id}$ to reconstruct the image $I'$ which is visually close to $I$. Suppose we attempt to perform the reconstruction using $A'_{Id}$ and the DCT index locations of the other image $J$ i.e. $\Pi_{Jd}$, and obtain the reconstructed image $I''$. Then it has been observed that $I''$ is visually almost same as $J$ instead of $I$.

- Figure 5.3(a) shows the curve corresponding to $A_{Id}$ of Lena (DC value has been discarded for better visual). Figure 5.3(c) is the resultant $I''$ by using the $A_{Id}$ of Lena ($I$) and $\Pi_{Jd}$ of Peppers ($J$).

- Similarly Figure 5.3(d) and Figure 5.3(e) are recovered by using $A_{Id}$ of Lena and $\Pi_{Jd}$ of mandrill and jetplane image respectively.

The above analysis proves that $\Pi_{Id}$ is the most important information required for proper recovery of $I'$.

On the basis of the methodology described above we utilize the uniqueness property of $\Pi_{Id}$ and propose a new IOD scheme described in detail in the next section.

### 5.4 Proposed Scheme

The methodology developed in Section 5.3 can be suitably adapted for any CAS based application, but we will explain the overall scheme in context of a new IOD system. In the proposed IOD system given an image $I$,

- A specific subset of $\Pi_{Id}$ known as $\Pi^S_{Id}$ is kept secret by the service
Figure 5.3: Sorted DCT coefficient ($A_{Id}$) dataset of Lena (q number of partitions of $A_{Id}$); Recovered image $I''$ using $A_{Id}$ corresponding to Lena and DCT index location dataset ($\Pi_{Id}$) of Lena; $\Pi_{Jd}$ of Peppers image as $J$; $\Pi_{Jd}$ of Mandrill image as $J$; $\Pi_{Jd}$ of Jetplane image as $J$ respectively.
CHAPTER 5. EFFICIENT IMAGE ON DEMAND SYSTEM

provider.

- The service provider constructs a low quality image $I^\$ \backslash \Pi^I_d$ and set of coefficients $A_{I_d}^1$, $A_{I_d}^2$, \ldots, $A_{I_d}^{dq}$.

- $I^\$ is shared in the public domain to lure customers. The finer details $\Pi^I_d$ needed to construct the corresponding high quality image $I^*$ is sent only when the customer demands and pays for it.

A general overview of the scheme is shown in Figure 5.4. Major steps involved in the design of the proposed IOD system is described next.

![Figure 5.4: Overview of proposed Image on Demand System](image)

5.4.1 Major steps in design of proposed IOD system

1. Calculate a set of minimum $K$ number of segments $S_i$ from $\Pi_I_d$ and accumulate and store it in an array $\Pi^*_I_d$, which makes $\Pi^*_I_d$ a subset of $\Pi_I_d$. $S_i$ is defined as an array which consists of $\frac{B}{2}$ number of contiguous index locations picked from the beginning and $\frac{B}{2}$ number of contiguous index locations picked from the end side of $\Pi_I_d$ and stored from the beginning towards end in sequential manner as shown in Figure 5.5 (details given in Algorithm 3). $K$ and $B$ are numerical values related to actual experiments. We call this process as Calculation of $\Pi^*_I_d$. 

103
2. From the $K$ number of segments already defined in $\Pi_{Id}^*$ we identify a specific segment $S_i$ and store it in an array $\Pi_{Id}^\#$. We call this process as a **Broad Search** operation.

3. Again find a minimum possible subset of index locations from $\Pi_{Id}^\#$ and store it in an array $\Pi_{Id}^\$$. We call this process as a **Narrow Search** operation.

4. The DC value $\eta$ and $A_{Id_1}, A_{Id_2}, \ldots, A_{Id_q}$ information of $I$ in the form of $P_1, P_2, \ldots, P_q$ polynomials of degree $\theta$ and $\Pi_{Id}^* \setminus \Pi_{Id}^\$$ set of index locations are declared in public so that $I^\$ could be constructed for preview purpose.

5. $\Pi_{Id}^\$ constitutes the secret data which is supplied to the customer on demand to construct the corresponding high quality image $I^*$.

PSNR $Q(I^P, I)$ of any image $I^P$ with respect to another image $I$ is calculated by

$$Q(I^P, I) = 10 \cdot \log_{10} \frac{MAX^2}{MSE}$$  \hspace{1cm} (5.2)
CHAPTER 5. EFFICIENT IMAGE ON DEMAND SYSTEM

where $\text{MAX}_I$ represents the maximum possible pixel value of the image. When the pixels are represented using 8 bits per pixel, this is 255.

Let us assume that $I$ to be an i.i.d information source which generates a random sequence of symbols from a finite set of possible gray level values $g_1, \ldots, g_n$ and $n$ represents the span of gray level $[120]$. For example, in case of an 8-bit gray level image $n = 256$ and $g_i = 0, 1, \ldots, 255$. The probability of the event that the source will produce symbol $g_i$ is $p_I(g_i)$. $p_I(g_i)$ is calculated as

$$p_I(g_i) = \frac{\text{# of pixels having value } g_i \text{ in } I}{N \times M}$$

and $\sum_{i=0}^{255} p_I(g_i) = 1$ then the entropy of an image $I$ tells the amount of information contained in that and is given by

$$H(I) = -\sum_{i=0}^{n} p_I(g_i) \log_2 p_I(g_i) \quad (5.3)$$

Details of Step 1, 2 and 3 are given next.

**Step 1: Calculation of $\Pi_{I_d}^*$** Let us first calculate and identify $\Pi_{I_d}^*$ required for constructing a high quality image $I^*$ (for example, PSNR around 40 dB with respect to $I$).

To calculate $\Pi_{I_d}$ we chose a threshold $Q_T$ for $Q(I^*, I)$, where $Q(I^*, I)$ is the PSNR of $I^*$ w.r.t. $I$. Then we find out the minimum $K$ number of $S_i$'s from $\Pi_{I_d}$ which are required to construct $I^*$ such that $Q(I^*, I) \geq Q_T$. Accumulate and store $S_1, S_2, \ldots, S_K$ in $\Pi_{I_d}^*$. The detailed algorithm to calculate $\Pi_{I_d}^*$ is described in Algorithm 3.

Details of the notations and abbreviations used in the subsequent algorithms.
Algorithm 3 Calculation of $\Pi^*_I$

1: Perform step 1, 2 and first operation of step 3.
2: Choose a sufficient size $B$ of a specific $S_i$ of $\Pi_I : (N \times M) \mod B = 0$.
   $\#(B\text{-sized } S_i) = \frac{NM}{B}$.
3: $D = 0$.
4: while $Q(I^*, I) < Q_T$ do
5:     $D = D + 1$.
6:     $\Pi^*_I \leftarrow \text{Zeros}(BD, 1)$
7:     for $i = 1$ to $\frac{BD}{2}$ do
8:         $\Pi^*_I[i] = \Pi_I[i]$.
9:     end for
10:    $k = \frac{BD}{2}$.
11:    $r = (NM) - k$.
12:   for $i = 1$ to $\frac{BD}{2}$ do
13:       $\Pi^*_I[k + i] = \Pi_I[r + i]$.
14:   end for
15:   $V \leftarrow \text{Zeros}(NM, 1)$
16:   for $j = 1$ to $\frac{BD}{2}$ do
17:       $V[\Pi^*_I[j]] = A'_I[j]$.
18:   end for
19:   $p = 1$.
20: for $j = \frac{BD}{2} + 1$ to $BD$ do
23: end for
24: $I^* := \text{IDCT}_{N \times M}(\text{SCAN}_{\text{inv-sig-zag}}(V))$
25: Calculate PSNR $Q(I^*, I)$ of $I^*$ w.r.t $I$.
26: end while
27: $\Pi^*_I$ is the set of required minimum number of index locations corresponding to $I^*$. $D$ is the required $K$. 
CHAPTER 5. EFFICIENT IMAGE ON DEMAND SYSTEM

- **Zeros**(x, y): Initialization of an array of size x × y with all zero values.
- **SCAN**\(_{\text{inv-zig-zag}}()\): Scan the array in inverse zig-zag manner.
- **SORT**\(_{\text{Asc.}}()\): Sort the array in ascending order.
- **SORT**\(_{\text{Desc.}}()\): Sort the array in descending order.
- **CORR**(X, Y): Correlation between random variables X and Y.
- **mean()**: Average or mean of the input array.

\(\Pi^*_{I_d}\) is the subset of minimum size identified from \(\Pi_{I_d}\) i.e. required to construct a high quality image \(I^*\) whose PSNR \(Q(I^*, I)\) is greater than a particular threshold value \(Q_T\). The reason behind taking half of the coefficients from beginning and half from the end is that \(A'_{I_d}\) contains the high magnitude coefficients but with negative sign towards the end of the array.

**Step 2: Broad Search** In this we identify a specific segment \(S_i\) and store it in an array \(\Pi^#_{I_d}\). This \(\Pi^#_{I_d}\) should be such that if the image \(I^#\) constructed from \(\Pi^*_{I_d}\ \backslash \ \Pi^#_{I_d}\) number of index locations should have the lowest entropy \(H(I^#)\) using (5.3) and maximum MSE \(\phi(I^#, I)\). \(I^#\) is constructed from \(\Pi^*_{I_d}\ \backslash \ \Pi^#_{I_d}\) number of index locations assuming that \(V\) is available, where \(V\) is the specific subset of \(A'_{I_d}\) corresponding to \(\Pi^*_{I_d}\) index locations. Detailed description of broad search is given in Algorithm 4.

Figures 5.6a and 5.6b shows the variation of \(H(I^#)\) and \(\phi(I^#, I)\) of \(I^#\) with respect to different index location segments of size \(B\) (except for the first segment where size is \(B - 1\)). Mostly the images are of low frequency nature and since the DCT coefficients are sorted in descending order, so the resultant curves will be of similar pattern for most of the images. It has also been verified through experiments. Curves shown in Figures 5.6a and 5.6b correspond specifically for Lena image. First segment will always be the best choice for \(\Pi^#_{I_d}\) since \(A'_{I_d}\) is in sorted form. In almost all of the cases, the segment corresponding to first entry of sorted \(E\) and \(F\) will be same. If not
same then first segment of \( E \) will be selected for \( \Pi^E \). \( E \) and \( F \) contains the 
\[ H(I^E) \] and \( \phi(I^E, I) \) of the \( I^E \) constructed from step 11 of Algorithm 4 in 
ascending and descending order respectively.

**Algorithm 4 Broad Search**

1: for \( j = 0 \) to \( K - 1 \) do
2:   for \( r = 1 \) to \( \frac{B}{2} \) do
3:     if not \( (j = 0 \) and \( r = 1) \) then
4:       \( V[\Pi^E_j|\frac{B}{2} + r]] = 0 \).
5:     end if
6:   end for
7:   \( p = (KB) - \frac{B(j+1)}{2} \).
8:   for \( r = 1 \) to \( \frac{B}{2} \) do
9:     \( V[\Pi^E[p + r]] = 0 \).
10: end for
11: \( I^E := IDCT_{(N×M)}(SCAN_{inv−zig−zag}(V)) \)
12: Calculate entropy \( H(I^E) \) using (5.3) and store it in an array \( E \).
13: Calculate \( \phi(I^E, I) \) using (5.1) and store it in an array \( F \)
14: end for
15: \( E \leftarrow \text{SORT}(\text{Asc.})(E) \)  \( F \leftarrow \text{SORT}(\text{Desc.})(F) \)
16: Select the segment of \( \Pi^E \) corresponding to the first entry of the sorted 
  \( E \) and \( F \) as \( \Pi^E \) if the corresponding segment is same. Else pick the one 
  with the first entry of \( E \).

**Step 3: Narrow Search** Once \( \Pi^E \) is identified for an \( I \), we go one more step 
further to find and narrow it down to an optimal subset of index locations 
\( \Pi^S \) such that the following conditions could be satisfied.

- Correlation \( \nu \) between \( \Pi^S \)'s of the various test images should be sig-
  nificantly low. \( \nu \) between any two dataset \( \Pi^S \) and \( \Pi^S \) is calculated 
as

\[
\nu = \frac{\sum_{i=1}^{\tau} (\Pi^S_{l_d}(i) - \overline{\Pi^S_{l_d}})(\Pi^S_{l_d}(i) - \overline{\Pi^S_{l_d}})}{\sqrt{\sum_{i=1}^{\tau} (\Pi^S_{l_d}(i) - \overline{\Pi^S_{l_d}})^2 \sum_{i=1}^{\tau} (\Pi^S_{l_d}(i) - \overline{\Pi^S_{l_d}})^2}}
\]

(5.4)
CHAPTER 5. EFFICIENT IMAGE ON DEMAND SYSTEM

(a) Entropy Variation

(b) Mean Square Error Variation

Figure 5.6: Graphs: Variation of $H(I^\#))$ and $\phi(I^\#, I)$ with respect to different index location segments of size $B$

Here $\tau$ represents the size of $\Pi_{I_d}^8$. $\overline{\Pi}_{I_d}^8$, $\overline{\Pi}_{I_d}^8$ are the means of $\Pi_{I_d}^8$ and $\Pi_{I_d}^8$ respectively.

- Corresponding PSNR $Q(I^8, I)$ of $I^8$ w.r.t. $I$ suffices for preview purpose, where $I^8$ is constructed from $\Pi_{I_d}^8 \setminus \Pi_{I_d}^8$ number of index locations.

- The size $\tau$ of $\Pi_{I_d}^8$ is sufficient enough to resist cryptanalytic attacks by exhaustive search.

- $H(I^8)$ is sufficiently low.
CHAPTER 5. EFFICIENT IMAGE ON DEMAND SYSTEM

Algorithm 5 describes the pseudo-code to find out a set of mean correlation data \( \Omega(y) \). \( \Omega(y) \) represents the mean of \( \nu \)'s being calculated over all possible combination of test images for a specific data size corresponding to the value of \( y \), where \( 1 \leq y \leq 128 \). Size of the resultant \( \Pi_{I_d}^S \) is given by \( \tau = 2y - 1 \).

Figure 5.7a shows the variation of \( \Omega \) with respect to \( y \) by performing Algorithm 5 on various test images. Similarly Figure 5.7b and Figure 5.7c show the variation of \( H(I^S) \) and \( Q(I^S, I) \) respectively.

On the basis of Figures 5.7a, 5.7b, 5.7c and considering the conditions to be satisfied by \( \Pi_{I_d}^S \) the minimum possible value of \( \tau \) and corresponding \( \Pi_{I_d}^S \) is finalized.

Now if a customer is interested in the high quality version of \( I^S \) and pays for the same then the service provider may provide \( \Pi_{I_d}^S \) to the customer so that \( I^* \) which is a high quality version of \( I^S \) can be constructed at the customer end.

Next we describe the various results which have been achieved through the experiments performed and give the analysis of the efficiency and robustness of the system against possible attack scenario.
Algorithm 5 Calculation of $\Omega(y)$

1: Let $T$ number of test images be used.
2: $y = 1$.
3: for $j = 2 : 2 : B$ do
4:     for $p = 1$ to $T$ do
5:         if $j \neq 2$ then
6:             for $i = 1$ to $\frac{j}{2} - 1$ do
7:                 $\Pi^S_{\text{dp}}[i] = \Pi^\#_{\text{dp}}[i]$.
8:         end if
9:     end for
10:     $temp = \frac{j}{2} - 1$
11:     $h = 1$.
12:     for $i = \frac{j}{2} : -1 : 1$ do
13:         $\Pi^S_{\text{dp}}[temp + h] = \Pi^\#_{\text{dp}}[B - i]$.
14:         $h = h + 1$.
15:     end for
16: end for
17: for $s = 1$ to $\left(\frac{T}{2}\right)$ do
18:     Calculate $\nu[s] = \text{CORR}(\Pi^S_{\text{de}}, \Pi^S_{\text{df}})$: $1 \leq e \neq f \leq T$.
19: end for
20: Calculate $\Omega(y) = \text{mean}(\nu)$.
21: $y = y + 1$.
22: end for

5.5 Experimental Results, System’s Efficiency and Robustness Analysis

We conducted experiments on a number of standard gray level test images of size $256 \times 256$ available in uncompressed TIFF at [88] to prove the suitability of proposed scheme for IOD system on the basis of the methodology developed. Experimental Results and Observations: As stated earlier,
the low frequency images in general depict similar pattern of curves as shown in Figure 5.6a and 5.6b. Figures also show that if the segment indexed 1 is removed from $A_{ld}$ then the corresponding entropy $H(I^*)$ of $I^*$ is the lowest as well as the corresponding MSE $\phi(I^*, I)$ will be maximum. Therefore this segment is chosen during Broad Search for removal. Size of $\Pi^*_{ld}$ will be 255 since we don’t consider DC whose location will always be $\Pi_{ld}[1]$.

![Mean Correlation](image1.png)

(a) $\Omega$ Variation

![Entropy Variation](image2.png)

(b) Entropy Variation

Figure 5.7
By analyzing Figures 5.7a, 5.7b, 5.7c and considering the conditions to be satisfied by $\Pi_{I_d}^y$ as previously mentioned in **Step 3: Narrow Search** in Section 5.4, the optimal value of $\tau$ comes out to be 161 (this corresponds to $y = 81$). For $y = 81$ the value of $\Omega$ is 0.27 which is significantly low. Size of $\Pi_{I_d}^y$ i.e. $\tau$ should also be not low enough which could be approximated by an adversary. So with respect to this condition also $y = 81$ is an apt value since trying out 161! number of trials is significantly complex enough from an adversary’s point of view. Table 5.1 shows the correlation between $\Pi_{I_d}^y$ segment of various test images for $y = 81$.

<table>
<thead>
<tr>
<th></th>
<th>Lena</th>
<th>Jetplane</th>
<th>Mandrill</th>
<th>Peppers</th>
<th>House</th>
<th>Tree</th>
<th>Elaine</th>
<th>Clock</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lena</td>
<td>1</td>
<td>0.29</td>
<td>0.18</td>
<td>0.16</td>
<td>0.15</td>
<td>0.24</td>
<td>0.30</td>
<td>0.28</td>
</tr>
<tr>
<td>Jetplane</td>
<td>0.29</td>
<td>1</td>
<td>0.26</td>
<td>0.40</td>
<td>0.29</td>
<td>0.31</td>
<td>0.30</td>
<td>0.36</td>
</tr>
<tr>
<td>Mandrill</td>
<td>0.18</td>
<td>0.26</td>
<td>1</td>
<td>0.31</td>
<td>0.13</td>
<td>0.31</td>
<td>0.17</td>
<td>0.24</td>
</tr>
<tr>
<td>Peppers</td>
<td>0.16</td>
<td>0.40</td>
<td>0.31</td>
<td>1</td>
<td>0.28</td>
<td>0.22</td>
<td>0.31</td>
<td>0.25</td>
</tr>
<tr>
<td>House</td>
<td>0.15</td>
<td>0.29</td>
<td>0.13</td>
<td>0.28</td>
<td>1</td>
<td>0.23</td>
<td>0.34</td>
<td>0.16</td>
</tr>
<tr>
<td>Tree</td>
<td>0.24</td>
<td>0.31</td>
<td>0.31</td>
<td>0.22</td>
<td>0.23</td>
<td>1</td>
<td>0.31</td>
<td>0.21</td>
</tr>
<tr>
<td>Elaine</td>
<td>0.30</td>
<td>0.30</td>
<td>0.17</td>
<td>0.31</td>
<td>0.34</td>
<td>0.31</td>
<td>1</td>
<td>0.40</td>
</tr>
<tr>
<td>Clock</td>
<td>0.28</td>
<td>0.36</td>
<td>0.24</td>
<td>0.25</td>
<td>0.16</td>
<td>0.21</td>
<td>0.40</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 5.1: Correlation between $\Pi_{I_d}^y$ segment of various test images for $y = 81$

Figure 5.7(b) is the preview version of $I$ with PSNR value of around 15.
dB with respect to $I$. Figure 5.7(c) is the resultant attacked image $\tilde{I}$ when an adversary tries to recover $I^*$ by adding $\Pi^S_{J_d}$ of another image $J$. In this case $I$ corresponds to lena and $J$ corresponds to peppers image.

Interpretation:

- $\Pi^S_{I_d}$ represents the optimal segment of index locations which has very low $\nu$ with respect to the same attribute of another images.
- So $I$ can not be approximated even by $\Pi^S_{J_d}$ of any $J$ or by any random $\Pi^S_{I_d}$.
- It also signifies that $A_{I_d}[\Pi^S_{I_d}]$ represents the minimal segment of coefficients which if kept hidden leads to a maximum degradation in visual quality of $I^S$.
- $\Pi^S_{I_d}$ also represents the minimal segment of index locations which if received by a customer in image on demand system will lead to a construction of high quality image similar to $I^*$. 

Figure 5.7: Images: Desired image ($I^*$); Shared image ($I^S$); Attacked image $\tilde{I}$
• $A_{I_d}$ corresponding to any image has negligible significance.

• Figure 5.7(c) resembles more closely to $J$ instead of $I$ with fair enough perception of $I$ as well. So we can conclude that an adversary can not recover a good $I^*$ corresponding to $I$ by adding the $\Pi^S_{I_d}$ of any $J$ on the publicly available $I^S$.

5.5.1 System’s efficiency

The scheme proposed manages that the amount of data to be transmitted through secure channel is kept as low as possible. Exact details of the amount of data which is publicly available in the form of image database and the amount of data which is kept secret by considering a $256 \times 256$ gray level images is shown below.

1. Since each coefficient of the polynomials $P_i$ have been represented by using 16 bytes, the size of Polynomial Representation Set ($\rho$) shared in public domain (OFF-LINE) is calculated as $\rho = q \times (\theta_i + 1) \times 16 + 2$ bytes, where the 2 extra bytes is for the DC value.

2. The size of Initial index locations ($\chi$) also shared in public domain (OFF-LINE) is calculated as $\chi = B \times K \times 2 - \tau \times 2$ bytes.

3. The size of on demand data ($\psi$) which is transmitted only when demanded (REAL-TIME) consists of the finer details in the form of $\Pi^S_{I_d}$ and is given by $\psi = \tau \times 2$ bytes. This is the data that is communicated through a secure channel on demand.

For example, Figure 5.7(a) represents $I^*$ where $Q_T = 40$ dB, $q = 16$, $\theta = 15$, $B = 256$ and the corresponding value of $K$ from Algorithm 3 is 121. So the overall data transmission required is given by

• $\rho$ (Constant for same set of $q$ and $\theta$) is 4098 bytes.
• \( \chi \) is 61630 bytes.

• \( \psi \) is 322 bytes.

• Total transmission (\( \eta \)), where \( \eta = \rho + \chi + \psi \) comes out to be 66050 bytes.

From the above analysis we may conclude the following.

1. Publicly available data comprises of \( \rho + \chi \) which in the given case comes out to be 65728 bytes and the secret data \( \psi \) merely consists of 322 bytes. So real-time transmission of secret data through secure channel hardly consists of 0.0048 (less than 0.5 \%) of \( \eta \). This feature makes the proposed system efficient in real-time.

2. Since \( \psi \) is negligible, the amount of secure communication is very low, which is advantageous due to less computational and communication overheads.

3. Only \( \frac{66050 - 65536}{65536} = 0.78\% \) of the actual image size (for the given case image size is of 65536 bytes) is the only extra data required for storing the complete data for a single image. This is significantly much less when compared to the existing schemes where one preview copy and one actual high-resolution copy of the image is simultaneously stored in the image database.

### 5.5.2 Robustness Analysis

For an adversary the main challenge to attack the proposed scheme are:

1. To find out the exact reverse permutation of \( \Pi^8_{l_d} \) whose size is \( \tau \) in a possible space of \( Z = N \times M - \frac{\chi}{2} - 1 \).

2. To find out the sign’s of the DCT coefficients for the corresponding
reverse permutation.

With respect to the given values an adversary needs to try at-least \( (\frac{Z}{\tau}) \times \tau! \times 2^\tau \) number of trials to get one instance of \( \tilde{I} \) equivalent to \( I^* \). So if all other values remains same then \( Z = 34720 \) and \( \tau = 161 \). Accordingly the number of trials comes out to be \( \frac{34720}{34559} \times 2^{161} \), which is a huge number and will require a very strong computational resource for the same.

5.6 Possible application in Image/Video Encryption

Another very useful and efficient application that can be designed by the similar approach as discussed in this chapter could be to apply it for image/video encryption. More specifically for JPEG/MPEG standards since they work on the DCT domain. But as of now in the standard encoder structure of JPEG/MPEG, they compulsorily work on \( 8 \times 8 \) DCT blocks. Applying the similar kind of scheme in \( 8 \times 8 \) DCT block is although feasible, but it will result into a complex system with lot of computations as well as there will be significant increase in the storage requirement as well. If the DCT based image/video standard could be defined for a global DCT block size as being adopted in the proposed scheme for IOD or if the compulsion of using \( 8 \times 8 \) DCT block size may be removed from the existing JPEG/MPEG standard, then there is a very good possibility to design an efficient image/video encryption in general or specifically for JPEG/MPEG.
5.7 Conclusion

In this chapter we have proposed a scheme for CAS. The actual DCT index locations are the most important attributes of any image in the DCT domain. Thus, it can be utilized to uniquely represent an image and for designing simple and efficient system for CAS based applications like image on demand. We use this attribute of the DCT indices, in which the real-time transmission of the secure and significant data is negligible. Our strategy suffices as the good quality image can never be discovered without this data and thus it helps in achieving the cryptanalytic security. In the proposed scheme, the real-time transmission of most significant and secret data is only around 0.5% of the total data required to represent an image. This leads to significant reduction in the overhead of secure communication. Further, data storage requirement is also less when compared to the existing schemes as we do not need to maintain two separate copies (one low quality and another high quality). All these salient features make the proposed scheme highly suitable for designing an efficient and simple real-time IOD application with less communication in the secure channel and less requirement of power, which is a significant advantage in wireless applications.

In the next chapter we present an efficient encryption scheme for JPEG/MPEG which results into complete obscure results by using a selective encryption approach in the DCT domain.