

## 動画像符号化におけるフレーム並べ替えの一検討

木全 英明 北原 正樹 上倉 一人 八島 由幸

日本電信電話株式会社 NTT サイバースペース研究所 〒239-0847 横須賀市光の丘 1-1

E-mail: {kimata.hideaki, kitahara.masaki, kamikura.kazuto, yashima.yoshiyuki}@lab.ntt.co.jp

あらまし フレーム間予測符号化を使って動画像を符号化する場合にはフレームの参照関係に依存して符号化効率が異なる。複数フレーム分の参照画像メモリを持つ場合には、復号画像を一旦蓄積しておいて出力順序を変更することができるため、フレームの符号化順序を変更して符号化効率を向上することが期待される。符号化順序を決定するには全ての符号化順序で符号化して最も効率の良い符号化順序を選択しても良いが、この方法では符号化時間が膨大となる問題がある。そこで本稿では符号化順序を決定するために必要なコスト量を予め計算しておき、これをもとにフレームの符号化順序を決定する手法を提案する。

キーワード 符号化順序, フレーム並べ替え, 参照画像選択予測, H.264, MPEG-4

## A study on frame arrangement for video coding

Hideaki KIMATA Masaki KITAHARA Kazuto KAMIKURA and Yoshiyuki YASHIMA

NTT Cyber Space Laboratories, NTT Corporation 1-1 Hikari-no-oka, Yokosuka-shi, 239-0847 JAPAN

E-mail: {kimata.hideaki, kitahara.masaki, kamikura.kazuto, yashima.yoshiyuki}@lab.ntt.co.jp

**Abstract** Coding efficiency of video coding using inter-frame prediction depends on the relations of pictures in terms of references. With multiple-frame size of reference picture memory, coding efficiency is expected to be improved with changing the coding order of frames, while a decoded picture is stored in the memory and is delayed to be output. It is possible that the coding order is determined from the actual coding results of all coding orders, however the processing time is quite long in that method. Then we propose the quick way of determination of the coding order, based on the calculated costs in advance.

**Keyword** coding order, frame arrangement, reference picture selection prediction, H.264, MPEG-4

### 1. はじめに

映像配信サービスでは即時性についての要求条件は映像通信ほど厳しくない。映像通信の場合には送信側と受信側との映像の同期が重要であり遅延は許されないが、映像配信の場合には、映像データを受信して映像を再生するまでの間に、ある程度の遅延が許される。例えばデジタル放送の場合には、チャンネル切り替え時の映像切り替えに約 0.5 秒ほどの遅延が許される要求条件となっている[1]。

このような条件の下で映像を符号化する場合には、GOP の長さを許容される遅延時間に対応させ、GOP 毎のランダムアクセスを実現する。また GOP 内でのフレームの表示順序と復号順序(符号化順序)は異なってもよい。この時、符号化側と復号側には複数フレーム分の復号画像を蓄積できるだけの参照画像メモリを用意する必要がある。例えば MPEG-1 と MPEG-2 で B-picture を適用する場合には、2 フレーム分の復号画像を蓄積できる参照画像メモリを用意しておき、復号処理において、表示順序で直前と直後の P-picture を復

号した後に、これらを参照して B-picture を復号する。また H.264 ではこの概念を拡張して、復号側で複数フレーム分の参照画像メモリを DPB (Decoded Picture Buffer)として用意しておき、復号画像の出力順序を管理する[2]。例えば図 1 に示すように、I-picture として符号化した I1 フレームと、P-picture として符号化した P0 と P2 フレームを、P0, I1, P2 の表示順序で表示することも可能である。

このように GOP の長さ分だけ、フレームを受信した時刻から表示するまでの遅延が許容されるのであれば、参照画像メモリを複数フレーム分用意しておき、GOP 内のフレームの符号化順序を入れ替えることが可能である。これにより GOP 内フレームの符号化効率を向上

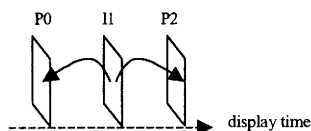


Figure 1. Example of coding order

することが期待される。例えば図1に示す順序によれば、表示順序で先頭フレームをI-pictureとして符号化して、残る2フレームをP-pictureとして符号化するよりも、I-pictureとP-pictureの時間間隔が短くなり符号化効率の向上が期待される。

フレームの並べ替えについて、[3]では、複数の静止画像をまとめて動画像とみなしてフレーム間予測符号化を用いて符号化する手法が提案されており、その静止画像の並べ順の決定方法が提案されている。ここでは、動画像符号化時の動きベクトルやフレーム間輝度差分の大きさを元にフレームの並べ替え順序を決定している。しかしこの方法では、動き補償における動き情報等のオーバーヘッド量を加味していないため、必ずしもレート歪特性が最適な符号化順序を決定できない。[4]では、フレームをMPEG-2におけるB-pictureで符号化するのかP-pictureで符号化するのかを、レート歪特性の最適条件下で適応的に決定する処理が提案されている。この枠組みにおいてB-pictureで符号化するには符号化順序が変更される。しかしながらここでの手法ではあくまでも1フレームの符号化モードを決定しているため、参照画像が複数ある場合には拡張できない。

また全フレームで想定される符号化順序の数だけ符号化を行い、レート歪特性の最も良い場合の符号化順序を求める方法も考えられる。しかしながら全フレームのパターン総数はフレーム数の順列分あり、全てのパターンについて符号化するには膨大な計算量が必要となる問題がある。そこで本稿では、マクロブロック毎のコストを予め計算しておき、これを元にフレームの符号化順序を決定する手法を提案する。本稿では復号画像を全て蓄積して参照画像とする場合について検討した結果を示す。

第2節では、H.264を参照して、本稿で適用する参照画像選択予測符号化の概要を示す。第3節では本稿で提案する符号化順序決定方法の概要を示し、第4節で更に決定方法を複数種類に分類して方法の詳細を示す。第5節で実験結果を用いて提案方法の有効性を議論し、第6節で今後の予定を含めて結論を示す。

## 2. 参照画像選択予測

### 2.1. 概要

複数の復号画像を蓄積できる参照画像メモリを用意しておき、その中から参照画像を選択して予測画像を作成する。H.264では予測モードに従ってマクロブロックを分割し、16x16, 16x8, 8x16, 8x8の大きさのブロック毎に参照画像を選択する。

デコーダは参照画像メモリに蓄積されている画像に0から始まる正の整数値をListとして割り当ててお

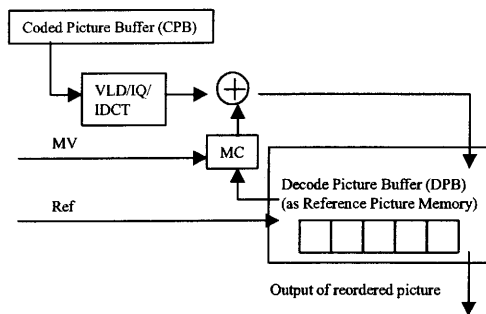


Figure 2. Illustration of decoded picture buffer

き、この番号を指定して参照画像を指定する。通常最も新しく蓄積した画像を0と指定するように番号を割り当てる。この割り当てはフレームを符号化または復号する度に更新する。

H.264の場合、P-pictureで複数の復号画像から参照画像を選択することができる。MPEG-2における後方向予測は、H.264では参照画像が切り替わっているに過ぎない。従ってP-pictureで実現できずB-pictureで実現できる機能は両方向予測のみである。H.264ではB-pictureで両方向予測を実現するために、上記の参照画像リスト情報Listを2系統管理する。

### 2.2. デコーダメモリ管理

デコーダへの入力は符号化データであり、出力は復号画像である。ここで、参照画像が複数ある場合には、画像の出力順序(表示順序)を任意に指定することができる。すなわち参照画像が複数ある場合のデコーダの動作は次のようになる。H.264で採用されているHRD(Hypothetical Reference Decoder)を参照して説明する。図2に示すブロック図のように、デコーダは入力符号化データ用バッファCPB(Coded Picture Buffer, MPEG-2ではVBVバッファに相当)を持っており、指定された時刻で符号化データをバッファから取り出してフレーム単位の復号処理を行う。そして復号画像を復号画像メモリDPB(Decoded Picture Buffer)に蓄積しておき、デコーダは指定された時刻に復号画像メモリから画像を出力する[2,5]。n番目のフレームを出力する時刻は次の式で与えられる。

$$t_o(n) = t_r(n) + t_c \cdot dpb\_output\_delay(n) \quad (1)$$

ここで $t_r(n)$ はn番目のフレームの符号化データをCPBから取り出す時刻であり、 $t_c$ はシステムの単位時間である。 $dpb\_output\_delay(n)$ で指定される時刻分だけ遅延して出力されることになり、この値は各フレームで指定することができる。

デコーダは復号画像メモリに蓄積した複数の画像

の一部を参照画像として使用する。画像の出力時刻は式(1)で与えられるため、画像の復号順序は出力順序である必要はない。

### 3. フレーム並び替え

参照画像選択予測符号化では、画像を復号する順序は出力順序と異なっても良いので、復号順序(符号化順序)を変更することにより、シーケンス全体の符号化効率を向上することが期待できる。実現するには、エンコーダで画像の符号化順序を決定する処理が必要である。単純に全ての符号化順序で符号化してみても、最も符号化効率の良い場合を最終的な符号化順序と決定する手法が最も良い。このときマクロブロックの符号化モードは式(2)で示される評価値  $J$  を最低にするものを選択する。

$$\begin{cases} J = SSD + \lambda \cdot R \\ SSD = \sum (r(i, j) - o(i, j))^2 \end{cases} \quad (2)$$

$R$  は符号量を示し、 $r(i, j)$  と  $o(i, j)$  はそれぞれ復号画像と原画像を示す。なお符号量  $R$  には動きベクトルと参照画像を指定する情報の符号量が加味されている。動きベクトルの符号化では周囲のブロックの動きベクトルからの差分を符号化対象とするが、周囲のブロックでの参照画像の選択状況によって符号量が異なる。

しかしながら符号化対象フレーム数が  $N$  フレームある場合の符号化順序パターン数  $Pn(N)$  は、

$$Pn(N) = N! \quad (3)$$

だけあり、符号化フレーム数  $N$  が大きい場合には膨大な処理時間が必要なため現実的では無い。式(2)において変数  $\lambda$  は例えば式(4)で計算される数値である[6]。

$$\lambda = C \cdot 2^{QP/m} \quad (4)$$

ここで  $C$  は実数精度の定数であり、 $m$  は整数精度の定数である。従って式(2)の評価値  $J$  は量子化スケール  $QP$  に依存して決定される。レートコントロールを行うつつ符号化順序を決定するためには、式(2)の評価値  $J$  を異なる  $QP$  について予め計算しておく必要があり処理時間が膨大となる。そこで予め計算しておく式(2)の評価値  $J$  を利用あたり、下記の仮定をする。

- 仮定1: GOP 内で必要とする映像の品質はシーケンス中の  $QP$  を第1フレームの値に固定した場合とほぼ同じである。

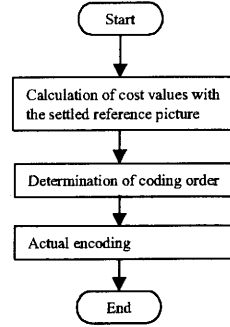


Figure 3. Flow of encoders

- 仮定2: GOP 内では  $QP$  を変更した場合の参照関係と  $QP$  を第1フレームの値に固定した場合とほぼ同じ参照関係を持つ。
- 仮定3: 評価値における動きベクトル符号量の寄与分は少ない。
- 仮定4: イントラ予測時の予測画像の画品質への、周囲のブロックの画品質からの影響は少ない。

以上の仮定を踏まえ、本稿では図3に示すようなフローに従って符号化順序を決定する方法を提案する。まず I-picture で符号化するフレームを選択し、それを参照画像に固定して参照画像が1フレームの場合の評価値(2)を計算する。全  $N$  フレームあり、フレーム内のマクロブロックが  $M$  個あるとすると、シーケンス全体での評価値の計算回数は、 $N \cdot (N-1) \cdot M$  回である。そしてこの評価値を使って符号化順序を決定する。このステップで設定する符号化順序の探索方法により符号化効率が異なる。そこで複数の探索方法を提案して性能を検証した。探索方法を次節で示す。

## 4. 探索方法

### 4.1. 全探索

式(3)で示される全符号化順序について、シーケンス全体の評価値  $J$  の合計を計算し、合計が最も小さい場合を最終的な符号化順序とする。この手法では各符号化順序パターンで評価値の合計を計算する必要がある。その計算に必要な評価値の計算回数  $Cn(N)$  は

$$Cn(N) = M \cdot \sum_{i=1}^{N-1} \min(i) \quad (5)$$

である。ここで  $\min(i)$  はフレーム数  $i$  の複数のフレームから最も評価値の小さいフレームを探索するために必要な計算回数を示す。従って全探索に必要な計算回数  $CnFS(N)$  は

$$CnFS(N) = Pn(N) \cdot Cn(N) = N! \cdot M \cdot \sum_{i=1}^{N-1} \min(i) \quad (6)$$

となる。

## 4.2. 優先探索

全探索は実際に符号化までは行わないが、やはりフレーム数が増えると処理時間が膨大となる問題がある。そこで全探索を行わない手法を検討する。

### 4.2.1. イントラ優先探索

通常 I-picture の方が、符号化効率が低い。最初に符号化するフレームの I-picture 符号化時の符号化効率がシーケンス全体の符号化効率に大きく影響を及ぼすと考えられる。そこで各フレームの I-picture 符号化時の評価値合計を元に、最初に符号化するフレームを決定する手法を考える。I-picture の符号量のみでは推定誤差が大きくなる場合があるため、第 1 フレームから第 3 フレームまでの評価値の合計を全探索で計算して最初の 3 フレームを決定し、第 4 フレーム以後の符号化順序を逐次決定する。

第 4 フレーム以後は次のように決定する。既に符号化順序が決定されている  $N$  フレームに 1 フレームを追加する時に、評価合計値が最も小さくなるように追加フレームを配置する(図 4)。この手法における評価値の計算回数の合計  $CnIP(N)$  は、

$$CnIP(N) = \begin{cases} CnFS(N) & \text{: for } N \leq 3 \\ CnFS(3) + \sum_{i=4}^N (i-1) \cdot Cn(i) & \text{: otherwise} \end{cases} \quad (7)$$

である。

### 4.2.2. 参照回数優先探索

予測符号化では効率の良い参照画像が参照画像メモリ中にあるかどうかで大きく符号化効率が異なる。そこで参照回数が多いフレームほど早く符号化するような決定方法を考える。

まず参照される可能性の高いフレームを探索する。フレームについてカウンタ  $C[j](j=1\dots N)$  を設ける。そ

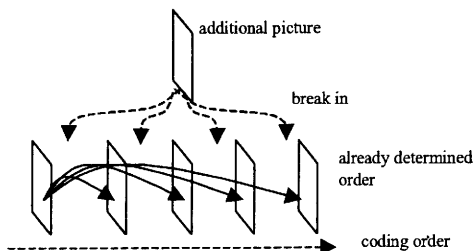


Figure 4. Illustration of breaking a picture in

Table 1. Coding conditions

# of reference pictures	All decoded pictures are stored
Base software	H.264 JM5.0
Quantization parameter	24,28,32,36 (fix)
Sequence	Coast, Mobile, Tempete (CIF)
Encoded frames	5, 7, 9, 11

Table 2. Simulation conditions

CPU	Pentium 4 1.5GHz
RAM	512 Mbyte
OS	Linux (Red Hat 8.0 J)

して第  $i$  フレームでマクロブロック毎に式(2)の評価値が最も小さいフレーム  $j$  を探索し、カウンタ  $C[j]$  の値を増加する。これを全マクロブロック、全フレームについて行う。このカウンタ  $C[j](j=1\dots N)$  の値が大きいほど、そのフレームは参照される回数が多いと仮定する。

上記のカウンタの数が大きい順序を符号化順序に決定する。この手法における評価値の計算回数の合計  $CnRP(N)$  は、

$$CnRP(N) = M \cdot \sum_{i=1}^N \min(i) \quad (8)$$

である。

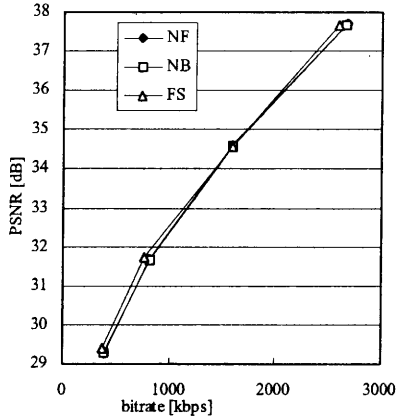
## 5. 実験と考察

提案する符号化順序決定方法の有効性を実験により評価した。符号化条件を表 1 に、実験環境を表 2 に示す。本稿ではフレーム間予測符号化するフレームは全て P-picture で符号化した。また復号画像は参照画像メモリに全て蓄積した。

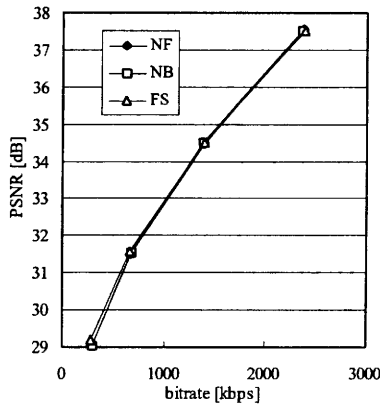
### 5.1. 全探索の符号化効率

全探索による符号化順序決定方法の符号化効率についての有効性を評価するために、2 種類の比較方式を実験した。比較方式 1(以後 NF)では時系列で並んだシーケンスを時刻順序で符号化する通常の符号化順序である。比較方式 2(以後 NB)では時系列で並んだシーケンスを時刻とは逆の順序で符号化する。

図 5 に画像 Coast におけるフレーム数 5 とフレーム数 11 の場合の、それぞれの PSNR の値を示す。図中の FS は提案手法の全探索による結果を表す。画像 Coast についてフレーム数が 5 フレームの時に全探索が最も符号化効率が良いことが分かる。また方法 NF と方法 NB はほぼ同じ符号化効率であることが分かる。フレーム数が 11 フレームの場合にはほとんど差がみられない。他方、画像 Mobile と画像 Tempete についてはフレーム数が 5 フレームの場合に全探索が、若干符号化効率が高く、11 フレームの場合には 3 方式でほぼ同じ



(a) the number of pictures is five



(b) the number of pictures is eleven

Figure 5. PSNR for Coast

符号化効率となった。

この結果より、画像内容に依存するものの、フレーム数が少ない場合には、フレーム並び替えは符号化効率を向上できること、また提案する図3のフローのように、予め評価値を計算しておき符号化順序を決定する手法は効果があることが示される。

### 5.2. 優先探索と全探索との比較

次に探索方法の違いによる効果の差について評価する。図6に画像 Coast について全探索方法(FS)とイントラ優先探索(IP)と参照回数優先探索(RP)によるPSNRの結果を示す。フレーム数が5フレームの時の結果である。この結果より、方式RPは全探索より符号化効率が高い(特にPSNRが35dB付近)ことが分かる。方式IPはPSNRが35dB付近では全探索よりも符号化効率が高いが、その他のPSNRでは符号化効率が低い

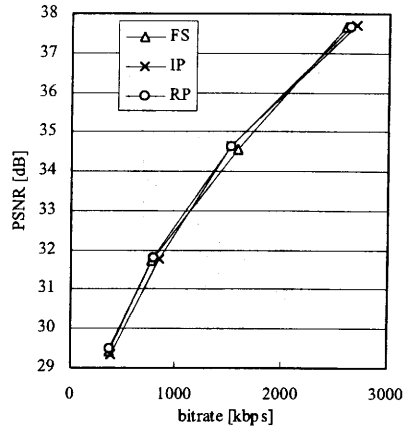


Figure 6. PSNR for successive search (Coast)

ことが分かる。

方法RPが方法FS(全探索)よりも符号化効率が高かったのは、全探索で評価値を使用する際の仮定3の妥当性に起因すると推測される。仮定3が妥当となるのは動きベクトル符号量の寄与が少ない場合であり、比較的符号化レートの高い場合と考えられる。他方、方式RPのように参照回数が多いフレーム符号化順序を早くすると、比較的同じフレームを参照するブロックが多くなる。従って符号化レートとは関係なく、評価値の計算で想定している動きベクトルの符号量は妥当なものとなる。すなわち方式RPの方が方式FS(全探索)の場合よりも、使用する評価値が妥当であると考えられる。

図7に方式RPで計算された参照回数の分布を示す。QPに依存して偏りがあることが分かる。このように偏りがあるため、参照回数の多いフレーム程早く符号化しておけば、より効率の良い予測が可能となる。

また図8に各QPにおけるP-pictureの平均符号量に対するI-pictureの符号量の比を示す。符号量比はQPが36の場合に最も大きく、3画像で平均して約10倍である。QPが24の場合に最も低く平均約3倍である。これらの結果より、I-pictureの符号量は確かに多いが、P-pictureのフレーム数が多くなるにつれてI-pictureの符号量を元に優先的に符号化順序を決定する効果が低くなる。

### 5.3. 処理時間

全探索と優先探索における符号化順序を決定するまでの所要時間を比較する。図9に処理時間を示す。このように全探索はフレーム数が増えると急激に探索時間が大きくなる。逐次探索であるイントラ優先探索と参照回数探索ではフレーム数が増えると探索時

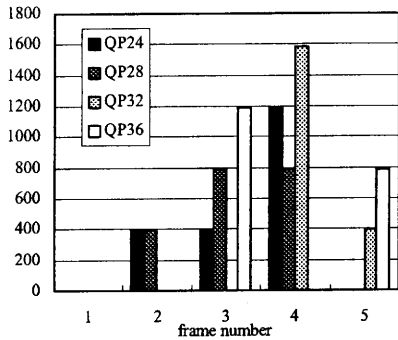


Figure 7. Distribution of reference pictures with the minimum cost per MB

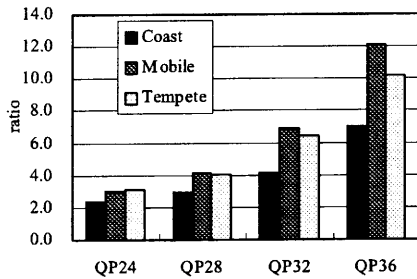


Figure 8. Ratio of bit number of I-picture to P-picture

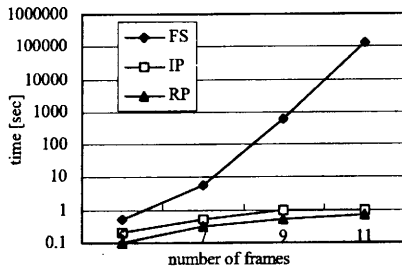


Figure 9. Measured processing time for determining coding orders

間は増えるものの、全探索よりは増加率は低い。

## 6. まとめ

本稿では、マクロブロック毎のコストを予め計算しておき、これを元に符号化順序を決定する手法を提案した。実験により、提案方法によれば、フレーム数が少ない場合に、通常的时间順序通りに符号化するよりも符号化効率の高い符号化順序を探索することができることを示した。

本稿の実験では、シーケンス全体のフレームを蓄積できる参照画像メモリを用いたが、実際にはメモリ量は有限であり、限定されたメモリ量での符号化順序を決定する手法が望まれる。本稿で示したフレーム数が少ない場合における符号化効率の向上は、メモリ量が限定された場合での、符号化効率の向上の可能性を示唆するものである。今後はメモリ量が限定された場合の符号化順序決定方法を検討する。

また本稿の実験では両方向予測符号化は使わなかった。今後両方向予測符号化を用いる場合も検討する。

## 文献

- [1] “デジタル方法における映像符号化、音声符号化及び多重化方式,” ARIB STD-B32, 1.3 版, 2002.
- [2] Hideaki Kimata, “A Proposal on B-picture Related Issue,” JVT-D075, Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG Meeting, 2002.
- [3] 村木隆浩, 石橋聡, 小寺博, “静止画像アーカイブのためのフレーム並び替えに関する検討,” 信学会ソサイエティ大会, D-11-19, 1997.
- [4] Jungwoo Lee and Bradley W. Dickinson, “Rate-Distortion Optimized Frame Type Selection for MPEG Encoding,” IEEE Trans. Circuits Syst. Video Technol., vol. 7, no. 3, pp. 501-510, 1997.
- [5] “Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC),” JVT-G050r1, Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG Meeting, 2003.
- [6] “Joint Model Reference Encoding Methods and Decoding Concealment Methods,” JVT-I049, Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG Meeting, 2003.