

MPEG-4 스튜디오 프로파일의 성능 향상 방법

Performance Improvement in the MPEG-4 Studio Profile

이정우¹, 김대희¹, 신중환¹, 김태완¹, 호요성¹, 전준근², 이명호²

¹광주과학기술원 정보통신공학과

²한국전자통신연구원 방송기술연구부

요약

MPEG-4 표준은 초기에는 낮은 비트율의 영상 통신을 지원하였으나, 표준화 작업이 진행되면서 스튜디오 프로파일(Studio Profile)과 같이 높은 비트율의 고품질 영상의 압축 전송을 포함하고 있다. 본 논문에서는 현재까지 진행된 MPEG-4 스튜디오 프로파일의 표준화 진행 상황을 살펴보고, 그 성능을 개선할 수 있는 방향을 제안한다.

1. 서론

현재 디지털 방송의 핵심은 고품질 영상 전송과 더불어 방송의 정보화를 실현하기 위한 데이터 방송이다. 프로그램 연동형 데이터 방송에서는 음성이나 영상 데이터에 동기된 부가정보가 전송되어 수신기에서 합성된다. 이것을 실현하기 위해서는 MPEG-4 표준에서 채택하고 있는 객체 부호화의 개념이 필요하다. 즉, 음성 객체, 영상 객체와 부가정보 객체가 병렬로 전송되며, 수신기에서는 이들 객체를 서로 복원하여 합성한다.

다양한 멀티미디어 서비스 응용을 목적으로 하는 MPEG-4 표준에서는 기존의 사각형 블록 단위의 화소 처리가 아니라, 객체(Object) 중심의 부호화 방식을 채택하고 있다. MPEG-4 표준의 임의 형상 부호화에서는, MPEG-1이나 MPEG-2 표준에서 개발된 프레임 단위 부호화의 하드웨어와 소프트웨어의 자원을 재이용하기 위해, 프레임 단위 부호화 기술로부터 최소한의 확장만을 수용하여 효율적인 객체 부호화를 수행한다.

MPEG-4 표준은 1998년 12월에 완성된 MPEG-4 1차 기본 규격과 2000년 말에 완성될

2차 확장 규격으로 구성되는데, 확장 규격은 기본 규격에서 논란이 되었던 문제들을 정리하여 새로운 프로파일로 추가하였다. MPEG-4 표준은 크게 다음 4가지의 특징을 가진다.

- (a) MPEG-4 표준은 부호화하여 전송해야 할 물체를 객체로 분류하여 부호화함으로써, 부호화 효율을 높이고 동시에 각각의 객체를 가공 편집할 수 있다. 이는 객체 단위로 영상 편집이 가능하여 기존의 영상들을 가지고 새로운 콘텐츠를 제작할 수 있으며, 객체 단위로 접근해서 얻어지는 많은 멀티미디어 응용 서비스를 창출할 수 있다.
- (b) MPEG-4 표준은 슬라이스 재동기 마커, 데이터 분할과 같은 새로운 기술을 이용하여 전송 오류에 대한 내성을 가진다. 이는 이동통신과 같은 전송 잡음이 많은 환경이나 네트워크의 과도한 부하로 패킷의 손실이 일어날 수 있는 환경에서도 효율적으로 정보를 전송할 수 있게 한다.
- (c) MPEG-4 표준은 처음에는 저속의 영상 부호화에 초점을 두었으나, 점차 높은 부호화 효율을 이용하여 다양한 영상 형식과 넓은 범위의 비트율을 지원하도록 설계되었다. 따라서 낮은 비트율의 이동통신 뿐만 아니라, 높은 비트율의 HDTV 서비스도 지원할 수 있다.
- (d) MPEG-4 표준에서는 구체적인 응용의 실현을 위한 도구 모음을 프로파일이라고 지정한다. 이러한 프로파일은 가능한 많은 응용에 효율적으로 적용되도록 구분한 것이다. MPEG-4 비주얼(Visual) 부분에는 9개의 프로파일이 정의되는데, 특히 메인

(Main) 프로파일은 영상의 크기 및 비트율을 주로 다룬다. 이것은 HDTV의 영상 크기까지 취급하도록 4레벨로 구성되며, 비트율도 최대 38.4 Mbps까지 허용한다. 각 프로파일은 보다 구체적인 응용에 맞도록 레벨에 의해 세부적으로 나뉜다.

본 논문에서는 고화질의 영상 전송을 위해 고려되고 있는 MPEG-4 스튜디오 프로파일의 부호화 방식에 대해 살펴보고, 이것의 성능을 개선할 수 있는 방안을 제시한다.

2. MPEG-4 스튜디오 프로파일

MPEG-4 스튜디오 프로파일은 주로 방송용이며, 주로 비디오 녹음, 편집 시스템, 가상 스튜디오 시스템 등에 응용될 것이다.

MPEG-4 스튜디오 프로파일은, 그림 1에 나타낸 것과 같이, 크게 스튜디오 심플(Simple) 프로파일, 스튜디오 코어(Core) 프로파일, 스튜디오 메인(Main) 프로파일의 3 부분으로 나뉜다.

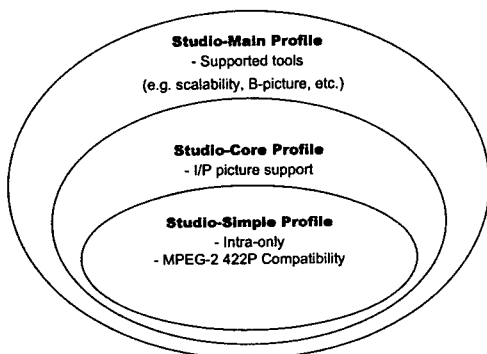


그림 1. 스튜디오 프로파일의 구조

심플 프로파일은 스튜디오 프로파일의 최소한의 집합으로 단순한 도구(Tool)를 포함한다. 10비트 도메인에서의 화면내 부호화와 크로마킹(Chromarking)을 위한 형태(Shape) 부호화를 지원한다. 알파플레인(Alpha Plane)의 동작을 위한 특별한 변수도 여기에 포함된다. 심플 프로파일은 MPEG-2 4:2:2 프로파일과 메인 프로파일에 적용된 비트율을 가지므로 MPEG-2 4:2:2 프로파일과의 호환성을 제공해야 한다. 따라서, 심플 스튜디오 프로파일이 MPEG-2 표준과 호환을 가지는 VLD(Variable Length Decoding) 기능과 P화면이나 B화면에 대한 움직임

보상 기능을 가져야 한다. 스튜디오 프로파일의 부호기와 복호기는 매우 높은 비트율을 가지고 큰 화면을 다루므로 하드웨어로 구현되어야 하며, 이러한 하드웨어의 구현을 쉽게 하기 위해 MPEG-4 표준의 기본 도구만을 지원한다.

코어 스튜디오 프로파일은 시간적인 예측 도구를 이용하여 좀 더 효율적인 비트열 전송을 지원하며, 메인 스튜디오 프로파일은 스튜디오의 응용 분야에 필요한 모든 도구를 지원한다.

MPEG-4 메인 프로파일과 비교해서 스튜디오 프로파일에 추가된 주요 기능은 4:2:2 형식과 4:4:4 형식, N 비트 화소, 고해상도 영상, 높은 비트율, MPEG-2 4:2:2 프로파일과의 호환성 등이다.

MPEG-4 비디오 복호화기를 주요 기능별로 분류하면, 형태 복호화기, 움직임 복호화기, 텍스처 복호화기의 3부분으로 나뉜다. 그림 2는 MPEG-4 복호화기를 도시하고 있다.

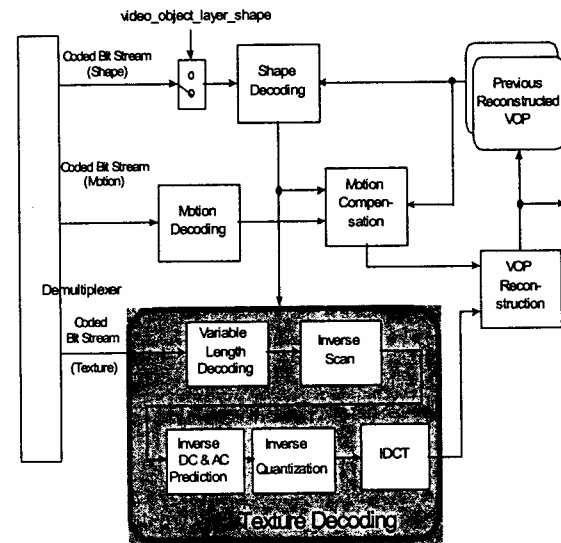


그림 2. MPEG-4 비디오 복호화기

복호화된 VOP는 이러한 복호화된 형태, 텍스처, 그리고 움직임에 대한 정보가 합성되어 만들어진다. 그러나 스튜디오 프로파일에서는 AC 예측이 사용되지 않는다. 그리고 스튜디오 심플 프로파일에서는 움직임 벡터 복호와 움직임 보상이 수행되지 않는데, 그 이유는 인터프레임(Interframe) 방식의 코딩이 화질의 열화를 초래할 수 있어 고화질의 영상 전송에

오히려 역효과가 있을 수 있기 때문이다.

지금까지 MPEG-4 스튜디오 프로파일에 대한 연구는 주로 일본의 SONY 회사와 NHK 방송사가 주도적으로 수행하였다. 초기에는 스튜디오 프로파일을 위해 MPEG-4 메인 프로파일에 추가된 기능에 대한 비트열을 교환하여 검증하는 방식으로 표준화 활동이 진행되었다. 즉, 4:2:2 형식과 4:4:4 형식을 수용하고, 10 비트의 화소를 처리하는 문제를 고려하였다. 그리고 화질 개선을 위해 화소가 n 비트라면 DCT의 정밀도를 $n+4$, $n+5$, $n+6$ 으로 확장하여 적용하는 방안을 검토하였다.

코어 스튜디오 프로파일에 해당하는 인터프레임 부호화와 스프라이트(Sprite) 코딩에 관한 연구가 주로 이루어지고 있고, 기존에 지원되지 않았던 스프라이트 코딩에 관한 4:2:2/4:4:4의 색차 형식에 대한 연구도 활발히 진행되고 있다. 2000년에 들어서는 입력 화소의 정밀도에 대해 가변적인 VLC 테이블을 이용하는 실험을 통하여 효율성을 증진시켰다.

또한 스튜디오 프로파일에서는 MPEG-2 비트열과의 호환성을 지원하기 위해 MPEG-2 비트열을 MPEG-4 비트열로 전환하는 방법에 대한 연구도 진행하였다. 그러나 현재는 인터프레임에 대한 전환은 고려되지 않고, 단지 인트라프레임에 대한 전환만이 고려되고 있다. 그리고 최근에는 스프라이트 코딩에 있어서의 자연스러운 합성 영상을 만들기 위해 디포커싱(Defocussing) 방법도 연구하고 있다.

MPEG-4 스튜디오 프로파일이 고화질의 영상을 전송하는 표준이지만, 근본적으로 데이터를 압축해야 한다. 그러나 높은 화질의 영상 전송을 추구하다 보면, DCT를 수행한 후 생성된 비트수가 원래의 화소를 그대로 전송하는 경우보다 늘어나는 경우가 발생할 수 있다. 이를 개선하기 위해 화소의 크기를 8 비트로 표현했을 때 한 블록의 데이터가 512 비트를 넘는 경우와, 화소의 크기를 10 비트로 표현했을 때 블록의 데이터가 640 비트를 넘는 경우에는 비압축(Uncompressed) 모드를 이용하여 원래의 화소값을 압축하지 않고 그대로 전송하는 방안을 제시하였다. 그러나, 비압축 모드는 무손실(Lossless) 전송이지만, 압축이 전혀 수행되지 않으므로 효율성이 떨어진

다. 이러한 문제를 해결하기 위하여 무손실 전송을 보장하면서 압축 효율을 향상시키기 위해 DPCM (Differential Pulse Coding Modulation)과 가변길이 부호화 방법으로 그 성능을 1.7에서 2배까지 향상시켰다.

고화질 영상 전송과 조작성을 위한 MPEG-4 스튜디오 프로파일에 대한 연구가 지금까지는 단순한 실험으로 이루어지고 있지만, 앞으로 디지털 TV 프로그램의 전송과 데이터 방송에서 활발히 적용될 것으로 보인다.

3. 스튜디오 프로파일의 성능 향상 방안

본 논문에서는 MPEG-4 스튜디오 프로파일의 성능을 향상하기 위한 두 가지 방안을 제시한다. 이들은 모두 MPEG-4 스튜디오 프로파일의 비압축 블록 모드와 관련된다. 첫번째 방법은 현재 화소를 예측하기 위해 간단한 예측 방법만을 이용하는 것 대신에 JPEG 표준과 같이 다양한 예측 모드를 지원하는 것이다. 두번째는 한 가지 스캔 모드만을 사용하는 기존의 방법 대신에 다양한 스캔 모드를 사용할 수 있도록 하는 것이다.

3.1 다양한 예측 모드를 이용한 방법

비압축 부호화 모드를 보완하기 위해 제안된 간단한 DPCM 방법을 이용한 알고리즘은 정확한 예측기를 이용하지 않아 그 출력 데이터의 분산이 영상에 따라 많이 다르기 때문에, 이 방법은 효과적으로 잉여 정보를 제거할 수 없다. 이상적인 예측 오류 필터(Prediction Error Filter)는 압축할 수 없는 백색 잡음을 만들어 내야 하지만, 이상적인 예측 오류 필터는 구현하기 어려우므로 가변적으로 수렴하는 적응적인 예측 오류 필터를 사용하여 통계적으로 백색 잡음에 근접한 신호를 만들 수 있다. 본 논문에서는 적응적인 예측 필터의 일종으로 다양한 예측 모드를 사용하여 현재 블록에 대해 최선의 예측 모드를 제공한다.

MPEG-4 스튜디오 프로파일의 비압축 블록 모드에서 사용되는 기존의 DPCM 예측 방법은 현재 화소를 예측하기 위해 주위의 3화소값을 이용한 한 가지 예측 방법만을 이용한다. 그림 3은 현재 화소값과 예측을 위해 사용되는 화소값들 사이의 관계를 나타낸다.

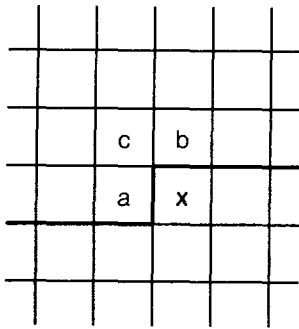


그림 3. 현재값과 예측값들 사이의 관계

그림 3에서 x 는 예측되는 화소이고, a, b, c 는 2-D에서 각각 x 에 바로 인접하면서 이전에 부호화되는 위, 왼쪽, 대각선에 위치하는 화소들이다. 부호화기와 복호화기 모두 같은 예측 방법을 이용하여 원 화소를 복원하므로 예측을 위해 사용되는 화소들은 반드시 현재 화소 이전에 부호화된 화소들만을 이용해야 한다. 그림 3에서 a, b, c 모두의 위치가 화면의 경계 밖에 있는 경우에 a, b, c 의 값은 모두 128로 설정하였고, a, b, c 중 두 개의 값이 화면의 경계 밖에 위치하는 경우에는 다른 하나의 값으로 두 개의 값을 결정하였다.

표 1은 본 논문에서 제안된 다양한 예측 모드를 제시하고 있으며, 이것은 DPCM 기반 무손실 JPEG 표준[1]과 유사한 것을 알 수 있다.

표 2는 다양한 예측 모드를 지원하기 위해 본 논문에서 제안한 구문 구조를 나타낸다. 표 2에서 `pred_block_mode`는, 표 1에 보이는 바와 같이, 현재 블록에 있는 화소값을 예측하기 위해 사용된 예측 모드를 가리키는 값으로 3비트 고정 비트수로 구성된다.

표 1. 예측 모드

pred_block_mode	Prediction Value
1	a
2	b
3	c
4	$a + b - c$
5	$a + (b - c)/2$
6	$b + (a - c)/2$
7	$(a + b)/2$

`rice_parameter`는 `rice_suffix_code` 필드의 길이를 가리키는 값이다. `rice_prefix_code`는, 표 3에 보이는 바와 같이, DPCM 차분값의 MSB 비트들을 가리키는 VLC 값이다. `rice_suffix_code`는 DPCM 차분값의 LSB 비트들을 가리키는 값이다.

부호화 과정

각각의 예측 모드에 대해, *residual* 값을 계산한다. 구해진 *residual* 값은 양수 뿐만 아니라 음수 값을 가질 수 있기 때문에, 이를 모두 양수로 처리하기 위해 식 (1)과 같이 변환된다.

$$\begin{aligned} residual &\leftarrow residual \ll 1 && \text{if } residual > 0 \\ residual &\leftarrow ((-residual) \ll 1) - 1 && \text{if } residual < 0 \end{aligned} \quad (1)$$

한 블록에 있는 64개의 *residual* 값에 대해 최대값을 계산하여, `rice_prefix_code` 값을 계산한다. `rice_prefix_code`는 0-7까지의 수를 표현할 수 있으므로, 이 코드의 길이는 최대값을 3비트 오른쪽으로 이동한 값의 길이가 된다.

표 2. 예측 모드 지원을 위한 구문 구조

StudioDPCMBlock() {	비트수	Mnemonic
<code>pred_block_mode</code>	3	<code>uimsbf</code>
<code>rice_parameter</code>	4	<code>uimsbf</code>
<code>for(i=0 ;i<8 ; i++) {</code>		
<code>for(j=0 ;j<8 ;j++) {</code>		
<code>rice_prefix_code</code>	1-12	<code>vlclbf</code>
<code>rice_suffix_code</code>	0-12	<code>uimsbf</code>
<code>}</code>		
<code>}</code>		
<code>}</code>		

표 3. rice_prefix_code의 VLC 값

Variable Length Code	Rice prefix code
1	0
01	1
001	2
0001	3
0000 1	4
0000 01	5
0000 001	6
0000 0001	7

즉, 현재 블록의 최대값이 200이라면, 이 값을 표현하기 위해서는 실제 8비트가 필요하므로, rice_prefix_code값으로 표현 가능한 MSB의 3비트를 제외한 5비트가 rice_suffix_code의 길이가 된다. 이 경우에 예측된 값의 대부분은 작은 값에 분포하므로 rice_prefix_code의 길이는 대부분의 화소들에서 작은 수의 비트를 차지하게 된다.

복호화 과정

복호화 과정은 부호화 과정의 반대이다. 즉 현재 블록에 대해, 먼저 블록에서 사용된 예측 모드를 결정하고, rice_parameter값을 이용하여 rice_prefix_code값과 rice_suffix_code값을 얻는다. 이 값을 residual이라 하면 최종적으로 이용되는 residual 값은 식 (2)와 같다.

$$\begin{aligned}
 residual &\rightarrow residual >> 1 && \text{if } residual \text{ is even} \\
 residual &\rightarrow -residual >> 1 && \text{if } residual \text{ is odd}
 \end{aligned}
 \tag{2}$$

3.2 다양한 스캔 방법을 이용한 방법

비압축 부호화 모드 블록을 부호화시 이용되는 DPCM에서 예측을 위해 사용되는 스캔 순서는 예측 이득에 상당한 영향을 미친다. 현재 사용되는 스캔 순서는 고정된 한가지 방법만을 이용하여 수행되고 있는데, 부호화되는 화면이 순행주사 방식의 화면이 아닌 격행주사 방식의 화면인 경우에 위 아래 화소간에 연속성을 상실하게 되어 성능의 저하를 가져올 수 있다. 이를 위해 다양한 스캔 순서 방식을 적용하는 것이 필요하다. 그림 4, 그림 5, 그림 6은 8x8 블록내에 있는 화소 값들이 각각의 스캔 순서를 이용하여 1-D 데이터 형식

으로 변환되는 것을 보여준다.

화소값들이 그림과 같은 방식으로 스캔되는 경우에, 예측 값들은 기존의 Raster 스캔 방식에 비해 더 작은 값들을 가지는 분포를 가진다. 그림 6에서 가장 왼쪽에 위치하는 화소들은 예측시 사용되는 값을 바로 위에 인접하는 화소값으로 설정한다.

복호화 과정에 사용된 구문 구조는 표 2에 기술하였다.

0	1	5	6	14	15	27	28
2	4	7	13	16	26	29	42
3	8	12	17	25	30	41	43
9	11	18	24	31	40	44	53
10	19	23	32	39	45	52	54
20	22	33	38	46	51	55	60
21	34	37	47	50	56	59	61
35	36	48	49	57	58	62	63

그림 4. 스캔 순서 #1

0	1	2	3	4	5	6	7
15	14	13	12	11	10	9	8
16	17	18	19	20	21	22	23
31	30	29	28	27	26	25	24
32	33	34	35	36	37	38	39
47	46	45	44	43	42	41	40
48	49	50	51	52	53	54	55
63	62	61	60	59	58	57	56

그림 5. 스캔 순서 #2

0	1	2	3	4	5	6	7
8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39
40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55
56	57	58	59	60	61	62	63

그림 6. 스캔 순서 #3

4. 실험 및 결과

실험은 다양한 예측 모드의 지원과 스캔 순서를 이용한 방법에 대해, 8 Bit 격행주사 방식의 시퀀스와 8 Bit 순차주사 방식의 시퀀스에 적용하였다. 표 4는 여러 가지 스캔 순서를 이용하여 나온 평균 비트율을 나타낸다.

표 4. 평균 비트율 (스캔모드)

시퀀스		8 Bit Interlaced	8 Bit Progressive
비압축 모드		16 bpp	16 bpp
DPCM 스캔 모드	Order #1	13.20 bpp	12.69 bpp
	Order #2	11.27 bpp	11.94 bpp
	Order #3	11.36 bpp	12.16 bpp

표 5는 다양한 예측 모드를 이용하여 화면을 부호화한 경우에 평균 비트율을 나타낸다.

표 5. 평균 비트율 (예측 선택 모드)

시퀀스		8 Bit Interlaced	8 Bit Progressive
비압축 모드		16 bpp	16 bpp
예측 선택 모드	1	9.73 bpp	10.50 bpp
	2	12.00 bpp	11.18 bpp
	3	12.08 bpp	11.70 bpp
	4	19.37 bpp	18.30 bpp
	5	19.59 bpp	18.42 bpp
	6	19.77 bpp	18.45 bpp
	7	20.05 bpp	18.58 bpp
	혼합	8.57 bpp	8.85 bpp

표 5에서 보는 바와 같이, 혼합 예측 모드 선택 방법이 다른 선택 모드에 비해 향상된 성능을 보이는 것을 알 수 있다. 선택 모드 4, 5, 6, 7의 경우에 부호화한 데이터의 평균 비트량이 원 영상의 비트량보다 커지는 이유는 한 블록 내에서 가장 큰 값이 256 보다 크게 되어 `rice_suffix_code`의 길이가 6으로 결정되기 때문이다. 하지만 이 경우에도 많은 화소들의 `rice_prefix_code`가 0으로 설정된다. 이 경우에는 `rice_suffix_code`의 길이를 6보다 적게 하고, 대신에 `escape_code`를 두어 `residual` 값 자체를 그대로 전송하는 방법이 필요하다.

5. 결론

본 논문에서는 현재 MPEG-4 표준화 작업의 일부인 MPEG-4 스튜디오 프로파일에 대한 개

요와 현재 논의되고 있는 사항들에 대해 기술하였다. MPEG-4 스튜디오 프로파일의 성능을 향상시키기 위한 방법으로 다양한 스캔 순서를 이용하는 방법과 다양한 예측 모드를 지원하는 방법을 제안하였다. 스캔 순서의 경우에 스튜디오 프로파일에서 사용되는 시퀀스의 일부가 격행주사된 것을 사용하므로 지그재그 스캔 순서와 같은 경우에는 좋지 못한 결과를 보인다는 것을 알았다. 따라서, 다양한 스캔 순서를 제공하는 것이 필요하다. 여러 개의 예측 모드를 이용하여 블록 단위로 예측 모드를 변화시켜 부호화 하는 경우에 다른 예측 모두 보다 낮은 비트율을 생성하였다. 또한 이를 지원하기 위한 구문 구조를 제공하였다.

감사의 글

본 연구는 한국전자통신연구원(ETRI)의 객체기반 A/V 편집/제작 도구 개발 과제, 광주과학기술원(K-JIST) 초고속광네트워크연구센터(UFON)를 통한 한국과학재단 우수연구센터(ERC)와 교육부 두뇌한국21(BK21) 정보기술사업단의 지원에 의한 것입니다.

참고문헌

- [1] ISO/IEC 10918-1 | ITU-T T.81, Information technology-Digital compression and coding of continuous-tone still images: Requirements and guidelines, 1994.
- [2] O. Sunohahra and Y. Yagasaki, "Study of normative DCT operation for MPEG-4 studio profile," ISO/IEC JTC1/SC29/WG11 MPEG99/M5549, Dec. 1999.
- [3] O. Sunohahra and Y. Yagasaki, "The proposal of on uncompressed block coding mode for MPEG-4 Studio Profile," ISO/IEC JTC1/SC29/ WG11 MPEG99/M5139, Oct. 1999.
- [4] F. Bossen, "Studio profile: compressing the uncompressed mode," ISO/IEC JTC1/SC29/WG11 MPEG2000/M6274, July 2000.
- [5] ISO/IEC JTC1/SC29/WG11, "Text of ISP/IEC 14496-2 Studio Profile Amendment WD 3.0," MPEG99, Dec. 1999