

H.261이 안고 있던 두 가지 문제를 모두 만족시킨 결과를 낳았다. 아래 그림은 각 영상압축 알고리즘의 구현을 나타내어 보았다. 각각의 영상은 QCIF(176 pels * 144 lines)의 크기에 따른다. 양자화 특성치는 1에서 가장 고화질의 영상을 얻을 수 있고, 31에서 가장 저품질의 영상을 얻을 수 있다. Bit-rate는 그 반대이며, 2.3절에서 구체적인 압축률에 대해 다루었다. 그림 2.에서 볼 수 있듯이 H.261에서는 각 블록당 Quantization 및 loop filter를 사용함으로써 blocking 현상이 나타남을 볼 수 있다.



그림 2. H.261(quantization = 31)



그림 3. H.263+(quantization = 31)

주목할 만한 특징으로 loop filter와 Full pixel precision을 사용하는 H.261과 비교해 Half Pixel precision을 수행하는 H.263이 좀더 나은 화질을 얻을 수 있음을 볼 수 있다. 또한 H.263+에서는 더욱 직접적인 방법으로 Blocking 현상을 제거하도록 한다. 그림 3.은 H.263+에서 복원된 그림이다. 그림과 같이 양자화특성치 31에서 H.261과 H.263+에서의 화질의 차이는 많은 변화가 있었다. 특히, H.263+에서의 Annex J(Deblocking Filter Mode)는 더욱 부드러운 화질을 보장해준다.

2.3 구현된 압축 알고리즘의 압축률 비교

다음 표는 각 각의 코덱과 해당 mode의 사용유무에 따른 압축 Bit-Rate와 Luminance요소에서의 각

Pixel당 평균 MSE(Mean Square Error, 이하 MSE라 칭한다.)를 보여주고 있다. 입력 Sequence로는 보통 많이 쓰이는 Carphone.QCIF을 사용하였다. Carphone은 1420형식의 382Frame으로 이루어져 있다. 각 코덱의 Motion Estimation Algorithm은 Full Search Algorithm으로 검색범위는 ±15로 하였다. 아래 표에서 Quant 1은 1부터 31까지의 양자화 특성치 중 1임을 뜻하며, 나머지 역시 같은 의미로 쓰였다.

표 1.는 구현된 각 Codec에서의 Compression Ratio와 MSE(Mean Square Error)를 비교해 보았다. H.263은 Annex D, Annex E, Annex F, Annex G를 사용하였고, H.263+에서는 Annex D, Annex F, Annex J, Annex M, Annex S를 사용한 결과이다. H.263 및 H.263+는 H.261에 비해 화질면에서나 압축률면에서나 동시에 상당한 진전이 있음을 본 표를 통해 알 수 있다.

		Quant 1	Quant 8	Quant 16	Quant 24	Quant 31
Default Mode(H.261)	bitrate(BYTE)	1:2.90	1:27.85	1:68.62	1:108.92	1:132.59
	Average MSE	1.13	19.75	54.24	89.51	119.15
Full Mode(H.263)	bitrate(BYTE)	1:4.25	1:72.10	1:183.61	1:269.58	1:311.15
	Average MSE	1.16	28.54	63.91	96.50	122.52
D. F. J. M. S. T(H.263+)	bitrate(BYTE)	1:6.21	1:78.85	1:189.76	1:246.86	1:253.02
	Average MSE	1.65	32.00	68.50	99.63	122.71

표 1. 양자화 특성치에 따른 각 Codec에서의 stream의 길이

3. Motion Estimation 알고리즘의 구현 및 비교

Motion Estimation Algorithm은 H.263+에 구현하였고, Default Mode에서 적용하였다. 입력 Sequence로는 앞서 실험한 Carphone을 사용하였다. 표 2.는 각 알고리즘별 발생한 Bit-Rate와 MSE, 그리고 각 MacroBlock당 평균 Searching Point의 수를 나타내었다. 아래 표에서 Quant 1은 1부터 31까지의 양자화 특성치중 1임을 뜻하며, 나머지 역시 같은 의미로 쓰였다. 본 논문에서는 Four Step 검색 알고리즘(이하 FSS), Central Biased Orthogonal 검색 알고리즘(이하 CBOSA), Hybrid Adaptive 검색 알고리즘(이하 HASA)를 구현해 보았다.

표 2.와 같이 FSS와 CBOSA, HASA는 Full Search Algorithm에 비해 Bit-Rate와 MSE가 증가했으나, Search Point의 수는 오히려 급격히 감소했음을 볼 수 있다. 특이할 만한 점은 Full Search방법은 오히려 정확한 Vector를 찾아내는 대신 양자화 특성치 31에서 다른 알고리즘에 비해 Bit-Rate가 늘어났음을 보이는데, 이는 표 3.과 같이 양자화 Step Size를 높일수록 Vector길이가 길어지며 이에 따라 가변장 부호화시 Motion Vector VLC table에 의해 Bit-Rate가 늘어나기 때문이다. 양자화 특성치를 높일수록 Motion Vector가 커지는 것은 Half Pixel precision의 특성으로 화질이 부드러워져 주변값들이 비슷해지

기 때문이다. 또한 이결과는 움직임이 많을수록 FSS가 CBOSA보다 MSE가 더 작다는 것을 보여준다.

		Quant 1	Quant 8	Quant 16	Quant 24	Quant 31
Full Search	bitrate(BYTE)	2,592,749	253,810	96,511	58,359	46,190
	Average Search Point	780.16	780.16	780.16	780.16	780.16
	Average MSE	1.93	26.57	62.91	97.24	126.35
FSS	bitrate(BYTE)	2,626,531	260,005	96,986	58,131	44,339
	Average Search Point	15.47	15.54	15.56	15.58	15.58
	Average MSE	1.98	26.65	63.54	99.15	130.40
CBOSA	bitrate(BYTE)	2,646,509	263,747	98,049	58,305	44,340
	Average Search Point	9.02	9.04	9.03	9.02	9.02
	Average MSE	1.97	26.68	63.85	99.73	131.46
HASA	bitrate(BYTE)	2,649,701	262,966	97,305	57,793	43,969
	Average Search Point	6.53	6.76	6.96	7.10	7.13
	Average MSE	1.98	26.93	64.33	100.48	133.35

표 2. 양자화 특성치별 Motion Estimation 방법에 따른 결과

		Quant 1	Quant 8	Quant 16	Quant 24	Quant 31
FS	MV (Horizontal)	0.59	0.64	0.66	0.68	0.71
	MV (Vertical)	0.49	0.53	0.62	0.71	0.76
FSS	MV (Horizontal)	0.37	0.39	0.39	0.39	0.38
	MV (Vertical)	0.35	0.37	0.37	0.38	0.38
CBOSA	MV (Horizontal)	0.35	0.35	0.35	0.35	0.34
	MV (Vertical)	0.30	0.30	0.31	0.31	0.32
HASA	MV (Horizontal)	0.31	0.30	0.29	0.29	0.28
	MV (Vertical)	0.29	0.29	0.29	0.29	0.28

표 3. 양자화 특성치별 Motion Estimation 방법에 따른 Vector 움직임량

표 4.은 Pentium III(550MHz)과 650MHz)를 이용해 382frame으로 이루어진 Carphone을 입력 Sequence로 하여 실험한 결과이다. Pentium III(550MHz)에서 Full Search Algorithm을 사용하여 Encoding을 했을 때 68.39초가 걸려 초당 5.6Frame을 Encoding함을 알았고, HASA를 사용했을 때 16.43초가 걸려 초당 23.3Frame을 Encoding 할 수 있었다. 특히 Pentium III(650MHz)에서 HASA를 사용했을 때 13.07초로 29.2 Frame을 Encoding 할 수 있어 거의 실시간 Encoding이 가능하다는 것을 알 수 있었다.

4. 결론

최근 Desktop상에서의 영상의 시스템 구축은

단위 : sec.(Frame per sec.)

	Full Search	FSS	CBOSA	HASA
P-III 550MHz with 64MB RAM	68.39(5.6)	18.85(20.3)	16.70(22.9)	16.43(23.3)
P-III 650MHz with 128MB RAM	55.09(6.9)	14.12(27.1)	13.34(28.6)	13.07(29.2)

표 4. Carphone을 Encoding하는데 걸린 시간

internet의 보급과 더불어 필수적인 요소로 자리 잡고 있으며, 그만큼 활용도 또한 방대해 지고 있다. 이는 PC의 성능향상과 고속통신망서비스의 확대에 힘입어 가능하게 되었으며, 더 이상 화상회의 시스템은 특정 단체의 전유물이 아니다. 그저 생활 속의 편리한 도구일 뿐이며, 앞으로의 이동통신에서의 지표가 되어 가고 있다. 현재까지 영상압축 알고리즘은 진일보하게 발전해왔다. ISDN 대응의 영상압축 알고리즘인 H.261의 등장 이후, H.263+와 MPEG-4까지 수많은 시행착오와 연구를 거쳐 압축과 화질이라는 두 마리 토끼를 잡기 위해 끊임없이 노력해왔다. 이에 본 논문에서는 H.261과 H.263, 그리고 H.263+라 일컬어지는 H.263 ver.2의 구현을 통하여 각 알고리즘의 장단점을 직접 비교해 보았으며, 그 변화과정을 직접 볼 수 있는 지표를 마련하여 앞으로의 개선점을 찾아 보고자 하였다. 또한 Motion Estimation의 보완을 통해 실시간 Encoding에 대한 가능성을 보여주었다.

최근 차세대 이동통신 IMT-2000에 H.263이 필수로 MPEG4가 선택사항으로 제시되었다. 이에 따라 개인 휴대통신이라는 개념이 새로이 정의되고 있으며, 영상압축 알고리즘의 중요성 역시 다시 떠오르고 있다. 1990년 이후, 10년 동안 통신업체를 중심으로 한 연구는 이제 영상의 질이라는 문제에 더욱 접근해야 할 것이다. 이에 H.26x와 MPEGx를 중심으로 진행중인 연구는 통합될 필요가 있다. 차후의 영상압축 알고리즘으로 WAVLET이 실제 적용될 것을 기대해 본다.

5. 참고문헌

- [1] ITU-T Recommendation H.261
- [2] ITU-T Recommendation H.263(27January,1998)
- [3] 최신 MPEG, 후지와라 히로시 작, 교보문고
- [4] 'A Novel Four-Step Search Algorithm for Fast Block Motion Estimation', Lai-Man Po & Wing-Chung Ma,
- [5] 'A New Center-Biased Orthogonal Search Algorithm For Fast Block Motion Estimation', Lai-Man Po & Chok-Kwan Cheung
- [6] 'A Hybrid Adaptive Search Algorithm For Fast Block Motion Estimation', Chok-Kwan Cheung & Lai-Man Po