

로 세션의 모든 참가자들에게 제어패킷을 주기적으로 전송하여 분실패킷율(lost packet rate), 지터(jitter), 지연(delay) 등과 같은 QoS 정보를 교환하는 기능과 세션의 크기에 따른 여러 가지 전송률 제어(rate control) 및 가입/탈퇴와 같은 세션제어(session control) 기능을 수행한다 [3].

2.2 동영상 압축

아래의 그림 1에서처럼 하이브리드 코딩 구조를 갖는 방법들은 세 가지의 기본 원리들을 사용하고 있다. 첫 번째, 화면 내(공간적) 상관관계에 의해 압축한다. 이를 INTRA 모드라 하고, I-프레임을 말한다. 두 번째, 화면간(시간적) 상관관계에 의해 압축한다. 이전 화면에 대한 정보를 사용해서 압축하는 경우이다. 이것은 INTER 모드라 하고 P-프레임을 말한다. 마지막으로 부호의 발생 확률에 따른 가변장부호화를 이용한다[4].

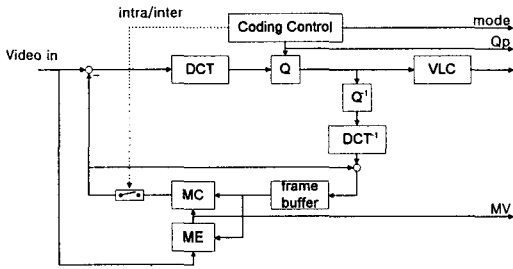


그림 1 The hybrid coding structure of H.261 and H.263

2.3 네트워크 상태 예상

네트워크 상태는 RTCP RR의 정보인 패킷 손실률을 사용하여 예상하는데, 우선 패킷 손실이 발생하는 경우는 2가지로 나눌 수 있다[2]. 첫 번째는 비트 에러가 발생한 경우이다. 하지만 이 경우는 현재의 네트워크 상황을 고려한다면 발생 확률이 아주 작다[1]. 두 번째는 중간 라우터에서 버퍼 오버플로우가 발생한 경우이다.

본 논문에서는 첫 번째의 경우를 무시하고 두 번째 경우만을 고려한다.

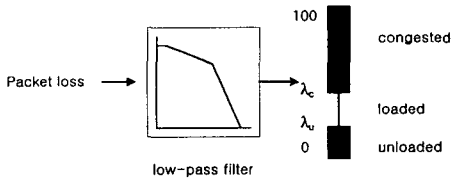


그림 2 Receiver Classification

[2]에서는 그림 2에서처럼 네트워크를 “congestion,” “load” 그리고 “unload”의 세 가지로 분류(RC : Receiver

Classification)한다. 이 분류를 본 논문에서는 “congestion”은 송신측과 수신측 네트워크가 “혼잡이 발생중인 상태” 그리고 “load”와 “unload”는 “혼잡이 해결된 상태” 또는 “충돌이 없는 상태”라고 결정한다. $\lambda_c = 2\%$ 와 $\lambda_u = 4\%$ 를 사용한다.

3. 제안 방법

그림 3에서처럼 하이브리드 코딩 구조에 의해 인코딩된 영상 패킷이 전송 중 손실이 발생하면 수신측은 손실된 패킷에 대해 은닉하는 여러 방법들을 사용한다[2]. 하지만 RTP기반의 실시간 전송에서는 손실된 패킷에 대해 재전송을 받을 수 없기 때문에 계속된 P-프레임 수신은 I-프레임을 받기 전까지 수신 품질을 떨어뜨린다. 본 논문에서 제안하는 방법은 저하된 수신 품질을 복구하기 위해 수신측으로부터의 피드백 정보를 이용하여 전송될 I-프레임의 시기를 결정하는 것이다. 가장 적절한 전송시기를 결정하기 위해 수신측은 RTCP의 RR (Receiver Report)에 부가 정보를 포함하고, 송신측은 이 부가정보를 가지고 예측된 수신품질과 네트워크 상황을 고려한다.

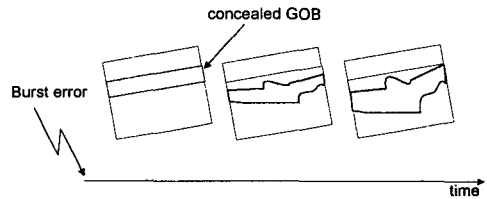


그림 3 공간/시간적 에러 전파

3.1 FMB의 결정

수신측은 패킷 손실이 발생한 후에 참조 벡터를 가지고 손실된 패킷에 대한 에러전파(propagation)와 에러정도(error energy: $E(i)$)를 계산하여 저장한다. 수신측은 자신의 RR 전송 주기 때 에러전파가 임계치 이상의 값이면, 에러정도가 최고인 블록을 FMB (Feedback Macro Block ; 16*16)로 결정하고 이 블록을 RTCP에 포함하여 전송한다.

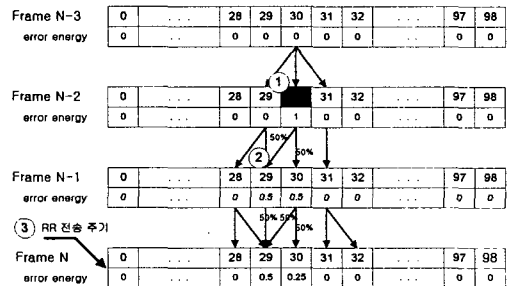


그림 4 에러전파와 에러정도

그림 4는 N-3에서 N까지의 4개의 프레임을 나타내며, 각 프레임은 0에서 98까지의 블록을 포함하고 각각의 블록은 에러정도를 기록하고 있다. I-프레임을 전송 받으면, 에러정도는 0의 값을 갖는다. 또 그림 4에서 화살표는 참조 벡터를 나타내고 있다.

①은 n-2 프레임에서 블록30의 손실을 나타낸다. 에러를 은닉하기 위해 에러 블록30은 N-3 프레임의 블록30으로 대체되고, 이때의 에러정도는 1이라 가정한다.

②는 에러 블록30을 N-1 프레임의 블록29와 블록30이 참조한 경우이다. 이때 각각의 블록은 50%씩 참조하므로 에러정도는 각각 0.5의 값을 갖는다.

③은 RR 전송 주기일때, 에러전과값이 임계치 이상이면 에러정도가 가장 큰 블록29가 FMB로 결정된다.

에러전과값은 전체블록 중에 에러정도가 0이 아닌 블록의 수로 구한다.

3.2 FMB를 이용한 수신품질 측정

송신측은 보관중인 영상과 FMB를 가지고 PSNR(Peak Signal-to-Noise Ratio)값을 구하고 이것을 이용하여 수신측 품질을 결정한다. 패킷 손실이 발생한 t시간의 PSNRt을 구하는 식은 다음과 같다.

$$PSNR_t = 10 \log_{10} \frac{1}{M} \sum_{i=1}^M \frac{255^2}{(o_i - c_i)^2} \text{ dB} \quad (1)$$

식 (1)에서 M은 전체 화소(pixel)의 수를 나타내고, o_i 와 c_i 는 각각 원영상과 수신받은 FMB의 i번째 화소값을 나타낸다.

수신측에서 FMB의 위치를 3.1의 방법을 이용하여 결정하므로 PSNRt는 급격히 변할 수 있다. 이런 진동현상을 방지하기 위해 low-pass filter를 사용한다. 수신품질 결정에 사용되는 PSNRt은 식 (2)와 같다. 여기서 α 는 $0 < \alpha < 1$ 의 범위를 갖는다.

$$PSNR_L \leftarrow (1 - \alpha) * PSNR_L + \alpha * PSNR_t \quad (2)$$

3.3 송신측 상태전이도

송신측은 3가지 상태를 갖는다. NORMAL 상태는 수신된 RR의 패킷 손실률을 이용하여 네트워크의 상태를 결정한다[5]. 또 FMB를 이용하여 수신측 품질을 구하고 I-프레임의 전송을 결정한다. 이 NORMAL 상태에서는 PSNRt의 값이 임계치보다 좋지 않을 때 네트워크의 상태에 따라 다른 상태로 전이할 수 있다. WAIT 상태는 I-프레임의 전송을 위해 네트워크 상태를 감시한다. e 경우는 PSNRt이 임계치보다 좋아지므로 I-프레임 전송 결정을 취소하며, 상태는 NORMAL로 전이한다. TRANSFER 상태에서는 타임아웃 전에 I-프레임의 전송 결정을 하지 않는다. 또 c 경우처럼 타임아웃 후 PSNRt 여전히 좋지 않다면 I-프레임 전송 결정을 하고 WAIT 상태로 전이한다

(그림 5 참조).

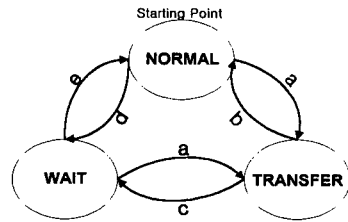


그림 5 송신측 상태 전이도

아래의 표 1은 각 상태에서의 동작과 전이 조건을 나타낸다.

표 1 동작(action)과 조건

Label	Action	Condition
a	PSNR _L 나쁨	I 프레임 전송
b	PSNR _L 좋음 && timer(RTO) == 0	PTO time out
c	PSNR _L 나쁨 && timer(RTO) == 0 && RC == congestion	PSNR _L 증가
d	PSNR _L 나쁨 && RC == unload, load	I 프레임 전송결정
e	PSNR _L 좋음	PSNR _L 감소

3.4 I 프레임 전송 후 PSNRt의 처리

I-프레임 전송 후 수신 품질은 패킷 전송 시간과 수신측 버퍼링 시간이 지난 후에야 좋아진다. 그러므로 만약 I-프레임을 전송한 시간부터 이 기간동안에 받은 RR의 FMB를 가지고 PSNRt을 계산한다면 연속된 I-프레임 전송을 결정할 수도 있다. 이 문제의 해결은 I-프레임 전송 후 PSNRt의 값을 감소시키고, 상응하는 RR을 수신받기전(PTO : Packet TimeOut)까지 I-프레임 전송 결정을 하지 않는다.

• 타임아웃계산

PTO는 다음과 같이 구할 수 있다. 식 (3)에서 M은 측정된 RTT값, R은 smoothed RTT estimator, 그리고 α 은 $0 < \alpha < 1$ 의 범위를 갖는 smooth factor이다. 또 γ 은 상수이고 범위는 $0 < \gamma < P$ 이다(P는 RR전송주기).

$$R \leftarrow \alpha R + (1 - \alpha) * (M + \gamma) \quad (3)$$

$$PTO = R \beta \quad (4)$$

식 (4)의 β 는 delay variance factor이다. TCP의 타임아웃인 경우에 권고 값은 2이다[6].

3.5 다수의 수신측인 경우

멀티캐스트 경우에는 다수의 수신측 상황을 고려하여야 한다. 만약 모든 수신측에서 RTCP_FMB를 전송한다

면 평균 RR 패킷 크기가 증가한다. 이것으로 인해 발생하는 문제는 RTCP 전송을 위해 RTP 표준은 전체 세션 대역폭의 5%를 할당하는데, 증가된 RTCP 패킷 사이즈는 전체 제어 대역폭의 사용을 증가시켜 제어 패킷 전송 주기를 늘린다. 결과적으로 송신측은 수신측으로부터 제어 패킷을 자주 받지 못하므로 적절한 제어를 할 수 없다.

해결책으로는 RTCP 패킷은 모든 수신측에서 수신하므로 RR 패킷을 받은 세션멤버는 수신 받은 패킷 손실률과 자신의 패킷 손실률을 비교하여 자신이 더 높다면 RR 전송시 RTCP_FMB 전송한다.

3.6 FMB정보를 포함한 RTCP 패킷 포맷

그림 6는 [7]의 FIR패킷과 [8]의 RTCP UM(Upstream Message) 패킷을 수정한 FMB 정보가 추가된 RTCP_FMB 패킷을 보여준다. 송신측은 length 필드와 UMT 필드를 이용하여 FMB 포함 여부를 알 수 있다.

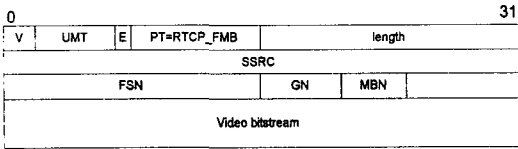


그림 6 RR : RR with FMB

• 헤더 블록

UMT : (Upstream Message Type) 5 비트 필드이다.

- 1 ~ 3 : 사용중
- 4 : FMB 사용을 지정한다.
- 5 ~ 31 : reserved

E : (Early Upstream Message) : 1 비트로 표준 RTCP RR 주기보다 빠른 UM(Upstream Message) 응답을 위해 사용된다[7][8].

• FMB 블록

FSN : (Frame Number) 16 비트 필드로, 패킷 손실이 처음으로 발생했을때, 시퀀스 넘버를 나타낸다.

GN : (Group Number) 5 비트 필드로, Video bitstream의 GOB(Group Of Block) 번호를 나타낸다.

MBN : (Macro Block Number) 4비트 필드로, GOB안의 MB위치를 나타낸다.

4. 결론

본 논문에서는 RTP기반의 실시간 멀티미디어 전송에서 패킷 손실 이후 여러 축적으로 저하된 수신측 품질을 복구하기 위해 전송할 I-프레임의 적절한 시기를 결정하는 방법을 제안하였다. 이 제안된 방법으로 네트워크 상황과 RTCP RR에 포함시킨 FMB를 이용하여 전송될 I-프레임 적절한 전송 시기를 구할 수 있다.

향후 연구 과제로는 다수의 수신측들이 발생하는 FMB 정보를 포함한 RTCP_FMB들과 다른 제어 패킷들의 집성(aggregation)과 이것을 수행하기 위한 수신측 대역폭 선출 방법 등이 있다.

참고 문헌

- [1] Echehard steinbach, Niko Faber, Bernd Girod, "Standard Compatible Extension of H.263 for Robust Video Transmission in Mobile Environments," vol. 7, no. 6, pp. 872-881, Dec. 1997.
- [2] W. M. Lam, A. R. Reibman, and B. Lin, "Recovery of lost or erroneously received motion vectors," in Proc. ICASSP, vol. 5, April 1993.
- [3] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP:A transport protocol for real time application," RFC-1889. Feb. 1996.
- [4] B. Girod, E. Steinbach, N. Farber, "Comparison of the H.263 and H.261 Video Compression Standards," Proceedings SPIE Conference on Standards and Common Interfaces for Video Information Systems, vol. CR60, Oct. 1995.
- [5] I. B. Busse, B. Deffner, H. Schulzrinne, "Dynamic QoS Control of Multimedia Applications based RTP," R2116 TOMQAT, May, 1995.
- [6] V. Jacobson, "Congestion avoidance and control," ACM Computer Communication Review, Vol. 18, pp.314-329, Aug. 1988.
- [7] Turletti, T., and C. Huitema, "RTP Payload Format for H.261 Video Streams," RFC-2032, Oct. 1996.
- [8] Stephan Wenger, Joerg Ott, "RTCP-based Feedback for Predictive Video Coding," Internet draft (work-in-progress) draft-wenger-avt-rtcp-feedback-00.txt, July 2000.