인터넷에 비디오 데이터 전송시 강건한 오류 내성 기법

김진옥[0], 황대준
성균관대학교 정보통신공학부
jinny[0]@ece.skku.ac.kr, djhwang@skku.ac.kr

A robust error resilient approach for video data transmission over Internet

Jinok Kim[0], DaeJoon Hwang
Sungkyunkwan University
School of Information and communication engineering

요 약

압축한 비디오데이타를 네트워크으로 전송 시 채널이 불안정한 경우 패킷이 분실될 우려가 있다. 패킷 분실은 대부분 버스트오류로 나타난다. 본 연구에서는 디코더에서 버스트 오류를 효과적으로 은닉,처리하는 방법으로 오류 내성 비디오 인코딩 방법을 제안한다. 이를 위해 공간적 오류은닉법으로 오류 패킷 분실을 야기시키는 손실 블록을 분리하는데 효과적인 블록 인터리빙을 적용한다. 시간적 오류 은닉에 대해서는 연속적인 내부프레임 또는 프레임간에 움직임벡터의 프레임간 패리티 비트를 삽입하는 구조를 적용한다. 비디오 인코딩 단계를 거쳐 디코더에서 수신한 블록들에 대해서는 쌍선형 보간법을 적용하여 전송시 발생한 국지적 오류를 적절하게 은닉 처리한다. 본 논문에서 제안한 인코딩 방법을 전송 블록에 부가 데이터로 포함하는 것은 표준 엔코더의 복잡도에 거의 영향을 미치지 않는다.

1. 서 론

인터넷의 발달로 디지털 멀티미디어 데이터를 네트워크에 실어 보내는 것은 일반적인 일이 되었다. 그러나 압축된 미디어 데이터는 사소한 에러비트라도 디코더에서 영상 왜곡을 일으킬 수 있기 때문에 불완전한 네트워크 채널에 민감한 특성이 있다. 인터넷상에서 비디오데이타를 전송할 때 데이터가 밀집되는 성질 때문에 패킷이 분실되고 분실된 데이터로 인해 버스트 오류가 발생한다[1]. 이 때문에 데이터 수신 단계에서 오류 은닉 기술의 적용이 필요하다. 오류 은닉은 전송된 비트열에 에러가 발생하여 정상적으로 복호화 되지 않고 복원된 영상의 화질을 떨어뜨리는 블록을 수신부에서 은닉하는 방법으로 손상된 데이터 블록과 주변 블록간의 공간적 시간적 연관 관계를 이용하여 사람이 인지할 수 있을 정도로 원신호에 가장 가까운 신호로 복원하는 것을 목적으로 하며 오류 제어 가장 많이 적용되는 기술이다[2].

비디오데이타에 대한 오류 은닉 방법은 시간적 영역의 오류 은닉 방법과 공간적 영역에서의 오류 은닉 방법 두 가지로 나뉜다. 시간적 영역에서의 오류 은닉 방법은 복원된 주변 블록의 정보를 이용해서 손실 블록의 움직임 벡터를 추정한 후 이전 프레임으로부터 손실된 블록을 보상하는 방법이다. 공간 방법은 움직임이 많은 영역에서 프레임내 부호화 된 블록에 효과적인 방법으로 주변 블록을 보간함으로써 가능하며 높은 복잡도를 보이지만 시각적인 품질은 더 낫다. 그래서 공간적 영역에서의 방법은 I 프레임에만 적용되고 시간 은닉방법은 P와 B 프레임에 적용한다. 본 연구에서는 인터넷과 같이 불안정한 채널에서 비디오 스트림 전송에 대해 디코더에서의 오류 은닉 효과를 개선하는 인코더단의 오류 내성법을 제안한다.

공간적 오류 은닉을 처리하기 위해 인코딩시 새로운 블록 인터리빙 구조를 적용해서 오류 블록들을 분리시켜 이 분리된 블록에서 패킷 분실로 인한 오류가 발생하도록 만든다.

시간적 오류 은닉을 처리하기 위해서는 움직임이 많은 비디오에서 움직임벡터의 복원은 쉽지 않다는 점을 고려하여 데이터 은닉을 이용한 연속적인 내부프레임 또는 프레임에 움직임벡터의 프레임간 패리티 비트를 삽입하는 구조를 제안한다. 디코더에서는 실제 오류 은닉 처리를 위해 주변 블록으로부터 내용 블록을 예측하는 대신 내용블록 그 자체에서 엣지 방향을 추출하여 추출한 정보를 쌍선형 보간법[3]을 이용한다.

인코딩시 제안된 에러 내성 기법을 데이터 블록에 적용했을 경우 오버헤드는 거의 없다. 제안 방법의 실험은 각 8 X 8의 DCT 블록과 16 X 16의 매크로블럭으로 이루어진 MPEG으로 코딩한 비디오 데이터를 이용한다.

본 논문에서는 인터넷의 IP 계층이 랜덤 비트 오류가 발생한 패킷은 전송하지 않기 때문에 패킷 손실만을 다루었다. 각 패킷은 재동기 비트를 가진 매크로 블록 한 라인을 포함하고 있다고 전제한다.

2장에서는 공간적 오류 은닉에 적용한 블록 인터리빙 구조를 설명한다. 움직임 벡터를 보호하는 데이터 은닉 방법은 3장에서 설명한다. 오류 은닉에 대한 전체 구조는 4장에서 다루고 실험 및 고찰은 5장에서 기술하며 6장에서 결론을 맺는다.

2. 블록 인터리빙

디코더에서 오류 은닉을 적용할 때 손상된 블록은 정확히 수신한 블록으로 싸여 있어야 복원 결과가 좋다. 손상된 블록을 손상되지 않은 블록으로 싸기 위해 엔트로피 코딩 전에 블록을 인터리빙한 다음 엔트로피 코딩후 다시 블록을 역 인터리빙하는 방법을 제안한다.

2.1 제안 구조

인터넷에서 전송된 패킷은 분실될 경우 그 분실 위치가 노출되는 버스트 오류이다.

버스트 오류는 코딩 블록을 심하게 손상시켜 오류 은닉이 불가능하게 만들기도 한다. 따라서 버스트 오류를 처리하는 동

해 규칙적 또는 랜덤하게 이루어 지는 치환 과정인 블록인터리빙 방법을 적용한다. 오류가 분리 되지 않으면 영상에서 불필요한 부산물이 두드러지지 않기 때문에 디코더 단에서 오류 은닉을 처리하기 어렵다. 본 연구에서 제안하는 인터리빙 패턴은 오류를 이미지의 전체 영역으로 분산시켜 중요한 연결 블록들이 동시에 분실될 수 있는 가능성을 낮춘다. 이 인터리빙 패턴을 '오류 분산 인터리빙' 이라 부른다. 오류 분산 인터리빙은 오류를 효과적으로 분산시키고 압축할 때는 오버헤드를 줄인다. 오류 분산 인터리빙은 송신된 패킷에서 블록의 모든 열과 행은 분실될 수 있는 가능성이 동일하다는 점을 고려했다.

18352746
61835274
46183527
74618352

그림 1. 인터리빙 패턴

그림 1은 제안 인터리빙 패턴을 설명하고 있다. $S = N \times M$ 크기의 이미지가 8 x 8크기의 블록으로 DCT 변환되었다고 할 때 이 패턴은 다음과 같은 방법으로 생성한다. N개의 열과 M개의 행으로 구성된 블록/ 매크로블럭의 이미지에서 M > N 이라고 가정한다. 최상 좌측 모서리부터 시작하여 (i,i) 블록을 "1"로 레이블을 붙인다. N블록을 "1"로 레이블링 한 다음 $(1, N+1)$ 블록부터 시작하여 모든 $(i, i+N)$ 블록을 "2"로 레이블링한다. 레이블링이 이미지의 우측 끝에 도달하면 "$k+1$"로 $(1, j+1)$ 블록부터 레이블링한다.만약 $(1, j+1)$ 이 이미 래이블링되어 있으면 "$k+1$"을 $(1, j')$ 블록에 할당한다. j' 은 $j+1$ 과 가장 큰 값으로 레이블된 인접한 행 인덱스의 중간값이다. 만약 이 역시 레이블되어 있다면 "$k+1$"을 $(1, j'')$ 블록에 할당한다. j'' 은 $j+2$ 와 가장 큰 값으로 레이블된 인접한 행 인덱스의 중간값이다. 모든 블록이 레이블 될 때까지 이 방법을 취한다. 이 인터리빙 방법은 버스트 오류가 발생한 손실 패킷상의 오류 블록을 격리시킨다.
고정된 방향의 인터리빙 패턴은 긴 비디오 스트림에서 오류를 발생시키는 쓸모없는 부산물 데이터를 만들 수 있다. 그래서 제안 방법과 같이 방향이 다른 인터리빙 패턴을 혼합하여 적용하면 부산물 데이터를 줄인다.

2.2 소스코딩과 채널 코딩

MPEG 표준을 기반으로 한 많은 연구는 압축한 비디오 비트스트림에 재동기워드와 같은 오류 내성 기술을 임베딩하면 소스 디코더가 랜덤 오류보다는 버스트 오류를 처리하는데 더 효과적이라고 보고했다[4].
소스 디코더는 랜덤 블록 오류의 복원을 위해 남아있는 자연잉여 데이터를 사용하는 반면 채널 디코더는 FEC(Forward Error Correction)코드를 이용하여 랜덤 비트 오류를 복원한다. MPEG 비디오의 구조는 보통 비트 오류를 블록 오류로 만드는데 코딩된 비디오에는 언제나 약간의 잉여 데이터가 있기 때문이다. 인터리빙 방식의 채널 디코더는 버스트오류 보다는 랜덤 오류 비트를 처리하는데 더 낫고 버스트 오류를 처리하는데는 소스 블록 인터리빙이 채널 인터리빙보다 더 효과적이다. 모든 표준 비디오 코딩 구조는 모두 블록 기반이기 때문에 같은 평균 비트 오류일 경우 소스 디코더가 랜덤오류 보다는 버스트 오류를 더 잘 처리한다.

시에 프레임내에서 손상된 블록들을 가능한 멀리 분리하기 위

3. 데이터 은닉을 이용한 움직임 벡터 보호

움직임 벡터는 시간적 영역에서의 오류 은닉을 위해 필수적이다 그래서 많은 오류은닉 방법이 손상된 움직임 벡터를 복원하는데 제안되었다[2] 그러나 손상된 영역이 그 주변과 연관관계가 거의 없을 경우 움직임 벡터를 복원하는 것은 매우 어려운 일이다. 움직임 벡터 복원을 위해 가장 많이 적용하는 FEC는 움직임을 잘 복원하지만 인터넷과 같은 네트워크를 통한 데이터 전송시 발생하는 버스트 오류를 처리하기 위해서는 많은 잉여 교정 코드가 필요하다[5]. 이 때문에 패킷은 더 많이 분실될 수 있다. 본 연구에서는 움직임 벡터를 복원하는데 명확한 오류 수정 비트를 삽입하는 데이터 은닉 방법을 적용하며 이 때 높은 데이터량을 가짐에도 감지 불가능한 데이터 은닉 구조를 이용한다. 이 구조는 비디오 압축에서도 유지 가능하다.
제안 방법에서 오류 수정은 패리티 비트를 통해 이루어진다. Song은 픽쳐에서 매크로블럭 그룹 (GOB) 전체에 패리티 비트를 삽입하는 방법을 제안했다[6]. 이 방법은 손실된 움직임 벡터가 프레임에서 한 개 GOB에만 한정되기 때문에 연속적인 GOB들이 한 개 프레임에서 손실될 수 있는 인터넷 전송에는 적절하지 않다. 그래서 본 연구에서는 미리 정해진 프레임간 그룹의 움직임 벡터에 대해 프레임 지향 패리티 비트를 생성하는 방법과 코딩구조와 지연요구에 따라 생성한 패리티 비트를 연속적인 프레임 내 또는 프레임 간에 삽입하는 방법을 제안한다. 제안 방법은 데이터 은닉방법을 이용함으로써 데이터 코딩 비율을 변하게 않게 유지한다. 이 구조는 버스트 오류 환경에서 손실된 움직임 벡터를 잘 복원한다. 프레임간에서는 P 프레임만 앵커프레임으로 사용하는데 이 프레임은 오류 확산에 민감하기 때문에 이 프레임들에만 움직임 벡터 보호방법을 적용한다.
제안 구조의 패리티 비트의 생성은 그림 2와 같다.

호프만 코딩된 움직임 벡터

P 프레임셋 P_1 01001....
P_2 00111....
P_3 01100...
P_4 10011... ⊕

패리티 코드(합을 모듈러 2계산) ──▶ 패리티비트

그림 2. 패리티 비트를 이용한 움직임 벡터 보호

그림 2와 같이 코딩된 움직임 벡터의 각 라인의 합을 모듈러-2 하여 패리티 코드를 생성한다. 각 라인들의 길이가 동일하지 않기 때문에 긴 라인과 짧은 라인을 맞추기 위해 짧은 라인의 빈 부분에는 '0'비트를 삽입한다. 이 구조는 각 비트 평면에 따라 발생하는 비트 오류 하나를 수정하기 위해 1개 패리티 비트를 사용하는 것이다. 생성된 패리티 비트를 그 다음 연속적인 프레임의 양자화된 DCT계수에 삽입된다. 두드러져 보이는 부산물 데이터를 덜 만들기 위해 DC와 낮은 AC계수는 바꾸지 않고 데이터 임베딩에 사용하는 상관계수를 선택하는데는 시각인지 모델을 이용한다. 프레임간에 패리티 비트를 적용하면 최소의 패리티 비트를 이용하여 같은 프레임에서 연속적으로 발생하는 오류를 보호하는 장점이 있다. 균일하지 않은 블록의 임베딩 용량을 균일하게 만들기 위해 인터리빙은 임베딩 전에 적용한다. 만약 비디오의 임베딩 용량이 작으면 움직임 벡터를 선택적으로 임베딩하거나 한 개 슬라이드에 움직임 벡터의 평균이나 합만을 임베딩한다. 한 개 패리티 비트 집합을 임베딩하기 위해 한 개 이상의 프레임을 이용할 수 있

4. 오류 은닉

엔코더에서 MPEG 스트림은 블록으로 분리되어 블록의 특징이 추출된 후 인터리빙된다. DCT와 양자화가 이루어진 다음 주파수방식의 임베딩이 블록에 행해진다. 만약 프레임이 I 픽쳐(미리 세팅된 픽쳐그룹의 첫번째 프레임)라면 프레임방식의 패리티 비트들이 I 프레임에 임베딩된다. 그 다음 엔트로피 코딩 되는데 재귀동기화 방식의 코드워드가 블록 각 스캔 라인의 처음에 붙여져 한 블록의 전송 에러는 한 개 라인만을 손상시킨다. 엔트로피 코딩 후 새롭게 생성된 MPEG 스트림이 패킷으로 만들어져 전송된다. 디코더에서는 패킷 번호를 이용해 손상된 블록을 찾는 에러감지가 먼저 수행되고 블록의 특징정보가 연관 블록에서 추출된 후 역 인터리빙을 한 다음에 쌍선형 보간이 손실 블록을 복원하는데 적용된다.

P 프레임들에 대해서는 GOP의 각 P 프레임에서 같은 위치인 모든 슬라이스에서 만약 한 개 슬라이스만이 손실된다면 이 슬라이스의 움직임 벡터들은 정확하게 수신된 다음 I 프레임 (또는 P프라임)에서 해당 슬라이스임을 확인하여 복원한다.

만약 오류가 발생한 블록의 특징을 임베딩한 연관 블록이 손실되었다면 미디안필터를 이용하여 인접한 움직임 벡터로부터 손실된 움직임 벡터를 보간하거나 H.263에서 제안된 것과 같은 중첩된 움직임 보상 방법을 이용하여 시간적 오류 은닉을 할 수 있다.

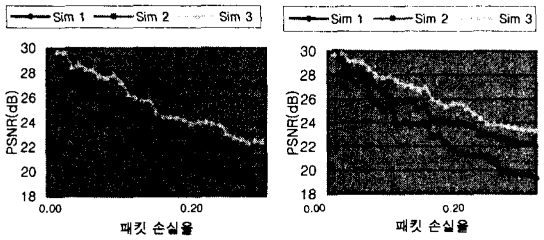
5. 실험

높은 움직임을 보이는 'football' 비디오데이타로 실험을 했다. 연속움직임은 초당 10프레임으로 QSIF(176 X 112) 의 크기의 MPEG-1으로 인코딩 되었다. GOP 프레임은 IPPP이고 B 프레임은 에러 확산에 영향을 미치지 않으므로 2B 프레임은 건너 뛰었다. 코딩 효율을 보여주는 데 고정된 양자화기를 적용했다. I 프레임에 대한 양자화 크기 팩터는 10이고 프레임간 그룹의 수는 4로 설정했다. 움직임벡터는 ±15픽셀의 크기 내 이고 정확도는 반픽셀이다. 인트라 블록은 오버헤드 때문에 P 프레임으로 한정한다. 비트 전송율은 256Kbps이다. 세가지 시뮬레이션을 수행하고 그 결과를 비교했다. 각 시뮬레이션은 인터리빙 구조, 공간 오류 은닉 방법, 시간 오류 은닉 방법등 세부분으로 행했다.표 1에 각 시뮬레이션 방법을 열거했다. 시뮬레이션 1의 블록 복사는 같은 공간위치를 점유한 이전 프레임의 블록으로 손상프레임을 대치하는 방법이다. 시뮬레이션 3의 공간적 오류 은닉 방법에서는 인코더 단에서 내용 블록의 엣지 정보를 추출해서 디코더로 보내 디코더에서는 이 정보를 이용하여 쌍선형 보간을 수행했다.다. 매크로 블록이 프레임 내 코딩인지 프레임 간 코딩인지를 판별하게 위해 1비트를오버헤드한다.

	없음	블럭 복사	블럭 복사
	2BI	쌍선형 보간법	Median Filter
	제안 인터리빙	쌍선형 보간법	제안 MV 패리티 임베딩 +MFr

표 1. 시뮬레이션 방법

먼저 비트율은 시뮬레이션 1에 대해 시뮬레이션 2와 시뮬레이션 3이 0.35%, 1.02%로 증가되었다. 이것은 제안 알고리즘이 무시할 정도의 낮은 압축율을 보임을 보여준다. 디코더에서는 패킷 번호로 오류 비트의 위치를 감지한다. 그림 3에서는 5개의 다른 채널 조건을 이용하여 시뮬레이션 했을 때 P_i 함수로써 원본 이미지와 변화된 이미지의 색상 변화를 나타낸 수치인 평균 PSNR(Peak Signal to Noise Ratio)을 보여준다. 인터넷 송신에서 전형적인 버스트오류 길이가 2일때 랜덤 오류 손실의 경우와 버스트 오류 경우를 시뮬레이션 했다. 실험결과 랜덤 손실의 경우 시뮬레이션 3이 시뮬레이션 1에 대해 3.81Db의 PSNR를, 시뮬레이션 2에 대해서는 1.98dB의 PSNR을 보였다. 버스트 오류에 대해서는 각각 3.37dB, 1.63dB의 PSNR을 보였다.



(a) 랜덤 손실 (b) L_b = 2 버스트 오류 손실

그림 3. 시뮬레이션 방법에 따른 평균 PSNR

6. 결론

본 논문에서는 인터넷상에서 비디오를 효과적으로 전송하기 위해 디코더에서 오류 은닉의 효과를 개선하는 방법으로 강건한 오류 내성 방법을 제안했다. 본 논문의 제안 방법들은 부가적인 복잡도는 그다지 높지 않으면서 코딩 효율은 유지하는 장점이 있다. 실험 결과는 제안 방법을 적용하면 패킷 손실율이 25-30%까지 높아져도 높은 비디오 품질로 복원됨을 보여준다. 제안 방법은 소스코딩에 잉여 데이타를 더하는 방법을 택했는데 이 방법의 대체안으로는 채널코딩에서 내용블럭에 잉여 데이타를 더하는 방법이 있다. 제안방법은 비디오의 구조와 내용을 이용하였으며 패킷 손실을 조합하는데 유용하다. 향후 연구에서는 제안 방법을 소스 코딩과 채널 코딩에 각각 적용하여 그 성능을 비교하고 조합하는 방안에 대한 검토를 계속해야 할 것이다.

참고문헌

[1] Jill M. Boyce and Robert D. Gaglianello, "Packet loss effects on MPEG video sent over the public Internet," ACM Multimedia'98, 1998.

[2] Yao Wang and Qin Fan Zhu, " Error control and concealment for video Communication: an overview," Proceedings of IEEE, vol. 86, no. 5, pp. 974-997, May 1998.

[3] H. Sun and W. Kwok, " Concealment of damaged block transform coded images using projections onto convex sets," IEEE Transactions on Image Processing, vol 4, pp. 470-477, April 1995.

[4] S. Gringeri, R. Egorov, K. Shuaib, A. Lewis and B. Basch, " Robust compression and transmisstion of MPEG-4 video," in ACM Multimedia'00 Electronic Proceedings, June 2000.

[5] U. Horn, K. Stuhlmuller, M. Link and B. Girod, "Robust Internet video transmission based on scalable coding and unequal error protection," Image Communication, Sept. 1999.

[6] Jie Song and K. J. R. Liu, " A data embedding scheme for H.263 compatible video coding," ISCAS'99, vol. 4, 1999.