(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2023-0125733
(43) 공개일자 2023년08월29일

(51) 국제특허분류(Int. Cl.)
H04N 19/82 (2014.01) *G06N 3/045* (2023.01)
G06N 3/0455 (2023.01) *G06N 3/0464* (2023.01)
G06T 5/00 (2019.01) *G06T 9/00* (2019.01)
H04N 19/105 (2014.01) *H04N 19/117* (2014.01)
H04N 19/176 (2014.01) *H04N 19/513* (2014.01)
(52) CPC특허분류
H04N 19/82 (2015.01)
G06N 3/045 (2023.01)
(21) 출원번호 10-2023-0007352
(22) 출원일자 2023년01월18일
심사청구일자 없음
(30) 우선권주장
1020220022404 2022년02월21일 대한민국(KR)

(71) 출원인
현대자동차주식회사
서울특별시 서초구 헌릉로 12 (양재동)
기아 주식회사
서울특별시 서초구 헌릉로 12 (양재동)
이화여자대학교 산학협력단
서울특별시 서대문구 이화여대길 52 (대현동, 이화여자대학교)
(72) 발명자
강제원
서울특별시 마포구 서강로 53 서강쌍용예가아파트 104-904호
허진
경기도 용인시 수지구 풍덕천로 91 신정마을1단지 101동 1201호
박승욱
경기도 용인시 수지구 성복2로 10 108동 304호
(74) 대리인
이철희

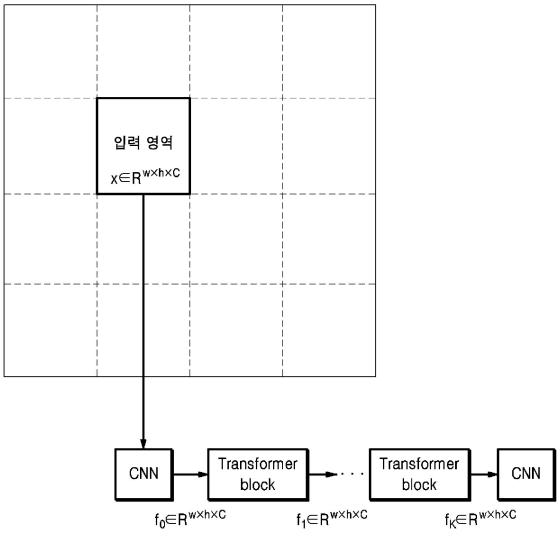
전체 청구항 수 : 총 16 항

(54) 발명의 명칭 **트랜스포머 기반 인루프 필터를 이용하는 비디오 코딩방법 및 장치**

(57) 요 약

트랜스포머 기반 인루프 필터를 이용하는 비디오 코딩방법 및 장치에 관한 개시로서, 본 실시예는, 현재 비디오 블록을 딥러닝 모델인 트랜스포머(transformer)의 어텐션 모듈(attention module)에 적용하고, 이에 따른 트랜스포머 기반 인루프 필터를 이용하는 비디오 코딩방법 및 장치를 제공한다.

대 표 도 - 도18



(52) CPC특허분류
G06N 3/0455 (2023.01)
G06N 3/0464 (2023.01)
G06T 5/002 (2023.01)
G06T 9/002 (2013.01)
H04N 19/105 (2015.01)
H04N 19/117 (2015.01)
H04N 19/176 (2015.01)
H04N 19/513 (2015.01)
G06T 2207/20084 (2013.01)

명 세 서

청구범위

청구항 1

영상 복호화 장치가 수행하는, 복원(reconstructed) 프레임의 화질을 개선하는 방법에 있어서,

상기 복원 프레임으로부터 기설정된 크기의 입력 영역을 획득하는 단계, 여기서, 상기 복원 프레임은, 원본(original) 프레임을 복원한 프레임으로서, 상기 영상 복호화 장치에 의해 사전에 복원됨; 및

상기 입력 영역을 딥러닝 기반 인루프 필터(inloop filter)에 입력하여 상기 원본 프레임을 근사하는 개선 비디오 영역을 생성하는 단계

를 포함하되,

상기 인루프 필터는 K(여기서, K는 자연수) 개의 연속된 트랜스포머(transformer) 블록들을 포함하고,

상기 개선 비디오 영역을 생성하는 단계는,

상기 트랜스포머 블록들을 이용하여, 어텐션 연산(attention operation)을 기반으로 상기 입력 영상을 최종 출력 특성으로 변환하는 것을 특징으로 하는, 방법.

청구항 2

제1항에 있어서,

상기 인루프 필터는 제1 CNN(Convolutional Neural Network)을 더 포함하고,

상기 개선 비디오 영역을 생성하는 단계는,

상기 입력 영역을 상기 제1 CNN에 입력하여 입력 특성(feature)을 생성하는 단계; 및

상기 트랜스포머 블록들을 이용하여, 상기 어텐션 연산을 기반으로 상기 입력 특성을 상기 최종 출력 특성으로 변환하는 단계

를 더 포함하는 것을 특징으로 하는, 방법.

청구항 3

제2항에 있어서,

상기 인루프 필터는 제2 CNN을 더 포함하고,

상기 개선 비디오 영역을 생성하는 단계는,

상기 최종 출력 특성을 상기 제2 CNN에 입력하여 상기 개선 비디오 영역을 생성하는 것을 특징으로 하는, 방법.

청구항 4

제2항에 있어서,

상기 변환하는 단계는,

각 트랜스포머 블록의 입력 특성을 패치들로 분할하고, 각 패치에 상기 어텐션 연산을 적용하여, 상기 각 트랜스포머 블록의 입력 특성에 대해 출력 특성을 생성하는 것을 특징으로 하는, 방법.

청구항 5

제4항에 있어서,

상기 변환하는 단계는,

상기 각 패치를 큐어리로 설정하는 단계;

상기 어텐션 연산에 이용되는 두 패치들 간의 유사도에 기초하여, 상기 각 패치에 대한 셀프 어텐션 스코어(attention score), 및 상기 각 패치와 다른 패치들 간의 어텐션 스코어들을 산정하는 단계; 및

상기 어텐션 스코어들을 가중합하여 상기 각 패치의 어텐션 값(attention value)을 생성하는 단계

를 포함하는 것을 특징을 하는 것을 특징으로 하는, 방법.

청구항 6

제4항에 있어서

상기 변환하는 단계는,

상기 어텐션 연산에 이용되는 두 패치들의 크기가 동일하지 않은 경우, 상기 두 패치들의 크기가 동일하여지도록 패딩을 적용하는 것을 특징으로 하는, 방법.

청구항 7

제5항에 있어서,

상기 변환하는 단계는,

상기 어텐션 연산에 이용되는 두 패치들이 상기 입력 특성에 함께 존재하지 않는 경우, 상기 어텐션 스코어를 연산하지 않는 것을 특징으로 하는, 방법.

청구항 8

제5항에 있어서,

상기 변환하는 단계는,

상기 어텐션 연산에 이용되는 두 패치들이 상이한 데이터 처리 단위들에 존재하는 경우, 상기 어텐션 스코어를 연산하지 않되, 상기 데이터 처리 단위는, CTU(Coding Tree Unit) 또는 VPDU(Virtual Pipeline Data Unit)인 것을 특징으로 하는, 방법.

청구항 9

제4항에 있어서,

상기 변환하는 단계는,

상기 각 패치가 기설정된 데이터 처리 단위의 경계를 벗어나는 경우, 상기 각 패치에 상기 어테션 연산을 적용하지 않고, 상기 각 패치의 일부가 상기 경계를 벗어나는 경우, 상기 경계를 벗어난 일부 영역을 표시하는 마스크(mask)를 이용하여, 상기 마스크로 표시된 영역에 대해 상기 어텐션 연산을 적용하지 않는 것을 특징으로 하는, 방법.

청구항 10

제4항에 있어서,

상기 변환하는 단계는,

상기 각 패치가 현재블록의 복호화 순서보다 후순위에 있는 경우, 상기 각 패치에 상기 어테션 연산을 적용하지 않고, 상기 각 패치의 일부가 상기 후순위에 존재하는 경우, 상기 후순위에 존재하는 일부 영역을 표시하는 마스크를 이용하여, 상기 마스크로 표시된 영역에 대해 상기 어텐션 연산을 적용하지 않는 것을 특징으로 하는, 방법.

청구항 11

제4항에 있어서,

상기 각 트랜스포머 블록은,

연속하는 트랜스포머 계층(transformer layer)들을 포함하고, 하나의 콘볼루션 레이어를 선택적으로 포함하며,

각 트랜스포머 계층은 하나의 인코더 층과 하나의 디코더 층을 포함하는 것을 특징으로 하는, 방법.

청구항 12

제11항에 있어서,

상기 각 트랜스포머 블록은,

스킵 연결(skip connection)을 이용하여 상기 트랜스포머 계층들의 잔차 블록(residual block) 형태로 구성되는 것을 특징으로 하는, 방법.

청구항 13

영상 부호화 장치가 수행하는, 복원(reconstructed) 프레임의 화질을 개선하는 방법에 있어서,

상기 복원 프레임으로부터 기설정된 크기의 입력 영역을 획득하는 단계, 여기서, 상기 복원 프레임은, 원본(original) 프레임을 복원한 프레임으로서, 상기 영상 부호화 장치에 의해 사전에 복원됨; 및

상기 입력 영역을 딥러닝 기반 인루프 필터(inloop filter)에 입력하여 상기 원본 프레임을 근사하는 개선 비디오 영역을 생성하는 단계

를 포함하되,

상기 인루프 필터는 K(여기서, K는 자연수) 개의 연속된 트랜스포머(transformer) 블록들을 포함하고,

상기 개선 비디오 영역을 생성하는 단계는,

상기 트랜스포머 블록들을 이용하여, 어텐션 연산(attention operation)을 기반으로 상기 입력 영상을 최종 출력 특성으로 변환하는 것을 특징으로 하는, 방법.

청구항 14

제13항에 있어서,

상기 인루프 필터는 제1 CNN(Convolutional Neural Network)을 더 포함하고,

상기 개선 비디오 영역을 생성하는 단계는,

상기 입력 영역을 상기 제1 CNN에 입력하여 입력 특성을 생성하는 단계; 및

상기 트랜스포머 블록들을 이용하여, 상기 어텐션 연산을 기반으로 상기 입력 특성을 상기 최종 출력 특성으로 변환하는 단계

를 더 포함하는 것을 특징으로 하는, 방법.

청구항 15

제14항에 있어서,

상기 인루프 필터는 제2 CNN을 더 포함하고,

상기 개선 비디오 영역을 생성하는 단계는,

상기 최종 출력 특성을 상기 제2 CNN에 입력하여 상기 개선 비디오 영역을 생성하는 것을 특징으로 하는, 방법.

청구항 16

영상 부호화 방법에 의해 생성된 비트스트림을 저장하는 컴퓨터 판독 가능한 기록매체로서, 상기 영상 부호화 방법은,

복원 프레임으로부터 기설정된 크기의 입력 영역을 획득하는 단계, 여기서, 상기 복원 프레임은, 원본(original) 프레임을 복원한 프레임으로서, 상기 영상 부호화 장치에 의해 사전에 복원됨; 및

상기 입력 영역을 딥러닝 기반 인루프 필터(inloop filter)에 입력하여 상기 원본 프레임을 근사하는 개선 비디오 영역을 생성하는 단계

를 포함하되,

상기 인루프 필터는 K(여기서, K는 자연수) 개의 연속된 트랜스포머(transformer) 블록들을 포함하고,

상기 개선 비디오 영역을 생성하는 단계는,

상기 트랜스포머 블록들을 이용하여, 어텐션 연산(attention operation)을 기반으로 상기 입력 영상을 최종 출력 특성으로 변환하는 것을 특징으로 하는, 기록매체.

발명의 설명

기 술 분 야

[0001] 본 개시는 트랜스포머 기반 인루프 필터를 이용하는 비디오 코딩방법 및 장치에 관한 것이다.

배 경 기 술

[0002] 이하에 기술되는 내용은 단순히 본 실시예와 관련되는 배경 정보만을 제공할 뿐 종래기술을 구성하는 것이 아니다.

[0003] 비디오 데이터는 음성 데이터나 정지 영상 데이터 등에 비하여 많은 데이터량을 가지기 때문에, 압축을 위한 처리 없이 그 자체를 저장하거나 전송하기 위해서는 메모리를 포함하여 많은 하드웨어 자원을 필요로 한다.

[0004] 따라서, 통상적으로 비디오 데이터를 저장하거나 전송할 때에는 부호화기를 사용하여 비디오 데이터를 압축하여 저장하거나 전송하며, 복호화기에서는 압축된 비디오 데이터를 수신하여 압축을 해제하고 재생한다. 이러한 비디오 압축 기술로는 H.264/AVC, HEVC(High Efficiency Video Coding) 등을 비롯하여, HEVC에 비해 약 30% 이상의 부호화 효율을 향상시킨 VVC(Versatile Video Coding)가 존재한다.

[0005] 그러나, 영상의 크기 및 해상도, 프레임률이 점차 증가하고 있고, 이에 따라 부호화해야 하는 데이터량도 증가하고 있으므로 기존의 압축 기술보다 더 부호화 효율이 좋고 화질 개선 효과도 높은 새로운 압축 기술이 요구된다. 최근 비디오 프레임의 화질의 개선하기 위해 딥러닝 기술인 CNN(Convolutional Neural Network) 외에도 트랜스포머(transformer)가 활용되고 있다. 특히, 인루프 필터링(loop filtering)에 있어서, 비디오 부호화 효율을 향상시키고 화질을 개선하기 위해, 딥러닝 기술의 효율적인 활용이 고려될 필요가 있다.

발명의 내용

해결하려는 과제

[0006] 본 개시는, 비디오 부호화 효율을 향상시키고 비디오 화질을 개선하기 위해, 현재 비디오 블록을 딥러닝 모델인 트랜스포머(transformer)의 어텐션 모듈(attention module)에 적용하고, 이에 따른 트랜스포머 기반 인루프 필터를 이용하는 비디오 코딩방법 및 장치를 제공하는 데 목적이 있다.

과제의 해결 수단

[0007] 본 개시의 실시예에 따르면, 영상 복호화 장치가 수행하는, 복원(reconstructed) 프레임의 화질을 개선하는 방법에 있어서, 상기 복원 프레임으로부터 기설정된 크기의 입력 영역을 획득하는 단계, 여기서, 상기 복원 프레임은, 원본(original) 프레임을 복원한 프레임으로서, 상기 영상 복호화 장치에 의해 사전에 복원됨; 및 상기 입력 영역을 딥러닝 기반 인루프 필터(inloop filter)에 입력하여 상기 원본 프레임을 근사하는 개선 비디오 영역을 생성하는 단계를 포함하되, 상기 인루프 필터는 K(여기서, K는 자연수) 개의 연속된 트랜스포머(transformer) 블록들을 포함하고, 상기 개선 비디오 영역을 생성하는 단계는, 상기 트랜스포머 블록들을 이용하여, 어텐션 연산(attention operation)을 기반으로 상기 입력 영상을 최종 출력 특성으로 변환하는 것을 특징으로 하는, 방법을 제공한다.

[0008] 본 개시의 다른 실시예에 따르면, 영상 부호화 장치가 수행하는, 복원(reconstructed) 프레임의 화질을 개선하는 방법에 있어서, 상기 복원 프레임으로부터 기설정된 크기의 입력 영역을 획득하는 단계, 여기서, 상기 복원 프레임은, 원본(original) 프레임을 복원한 프레임으로서, 상기 영상 부호화 장치에 의해 사전에 복원됨; 및 상기 입력 영역을 딥러닝 기반 인루프 필터(inloop filter)에 입력하여 상기 원본 프레임을 근사하는 개선 비디오

영역을 생성하는 단계를 포함하되, 상기 인루프 필터는 K(여기서, K는 자연수) 개의 연속된 트랜스포머(transformer) 블록들을 포함하고, 상기 개선 비디오 영역을 생성하는 단계는, 상기 트랜스포머 블록들을 이용하여, 어텐션 연산(attention operation)을 기반으로 상기 입력 영상을 최종 출력 특성으로 변환하는 것을 특징으로 하는, 방법을 제공한다.

[0009] 본 개시의 다른 실시예에 따르면, 영상 부호화 방법에 의해 생성된 비트스트림을 저장하는 컴퓨터 판독 가능한 기록매체로서, 상기 영상 부호화 방법은, 복원 프레임으로부터 기설정된 크기의 입력 영역을 획득하는 단계, 여기서, 상기 복원 프레임은, 원본(original) 프레임을 복원한 프레임으로서, 상기 영상 부호화 장치에 의해 사전에 복원됨; 및 상기 입력 영역을 딥러닝 기반 인루프 필터(inloop filter)에 입력하여 상기 원본 프레임을 근사하는 개선 비디오 영역을 생성하는 단계를 포함하되, 상기 인루프 필터는 K(여기서, K는 자연수) 개의 연속된 트랜스포머(transformer) 블록들을 포함하고, 상기 개선 비디오 영역을 생성하는 단계는, 상기 트랜스포머 블록들을 이용하여, 어텐션 연산(attention operation)을 기반으로 상기 입력 영상을 최종 출력 특성으로 변환하는 것을 특징으로 하는, 기록매체를 제공한다.

발명의 효과

[0010] 이상에서 설명한 바와 같이 본 실시예에 따르면, 현재 비디오 블록을 딥러닝 모델인 트랜스포머의 어텐션 모듈에 적용하고, 이에 따른 트랜스포머 기반 인루프 필터를 이용하는 비디오 코딩방법 및 장치를 제공함으로써, 비디오 부호화 효율을 향상시키고 비디오 화질을 개선하는 것이 가능해지는 효과가 있다.

도면의 간단한 설명

[0011] 도 1은 본 개시의 기술들을 구현할 수 있는 영상 부호화 장치에 대한 예시적인 블록도이다.

도 2는 QTBTTT 구조를 이용하여 블록을 분할하는 방법을 설명하기 위한 도면이다.

도 3a 및 도 3b는 광각 인트라 예측모드들을 포함한 복수의 인트라 예측모드들을 나타낸 도면이다.

도 4는 현재블록의 주변블록에 대한 예시도이다.

도 5는 본 개시의 기술들을 구현할 수 있는 영상 복호화 장치의 예시적인 블록도이다.

도 6은 콘볼루션 레이어의 연산을 나타내는 예시도이다.

도 7은 SISR(Single Image Super Resolution) 네트워크를 나타내는 예시도이다.

도 8은 SISR에 이용되는 잔차 블록을 나타내는 예시도이다.

도 9는 CNN(Convolutional Neural Network) 기반 고정 계수 인루프 필터를 나타내는 예시도이다.

도 10a 및 도 10b는 벡터의 셀프 어텐션을 나타내는 예시도이다.

도 11a 및 도 11b는 행렬의 셀프 어텐션을 나타내는 예시도이다.

도 12는 멀티헤드 어텐션을 나타내는 예시도이다.

도 13은 트랜스포머의 구조를 나타내는 예시도이다

도 14는 비전 트랜스포머의 입력 전처리를 나타내는 예시도이다.

도 15는 본 개시의 일 실시예에 따른, 현재 비디오 블록에 대해 어텐션 과정의 적용을 나타내는 예시도이다.

도 16은 본 개시의 일 실시예에 따른, 통합된 영역을 나타내는 예시도이다.

도 17은 본 개시의 일 실시예에 따른, 어테션 스코어가 연사되지 않는 경우를 나타내는 예시도이다.

도 18은 본 개시의 일 실시예에 따른, 트랜스포머 기반 인루프 필터를 나타내는 예시도이다.

도 19a 및 도 19b는 본 개시의 일 실시예에 따른, 트랜스포머 블록을 나타내는 예시도이다.

도 20은 본 개시의 일 실시예에 따른, 화질을 개선하는 방법을 나타내는 순서도이다.

도 21은 본 개시의 다른 실시예에 따른, 화질을 개선하는 방법을 나타내는 순서도이다.

발명을 실시하기 위한 구체적인 내용

[0012] 이하, 본 개시의 일부 실시예들을 예시적인 도면을 참조하여 상세하게 설명한다. 각 도면의 구성요소들에 참조 부호를 부가함에 있어서, 동일한 구성요소들에 대해서는 비록 다른 도면상에 표시되더라도 가능한 한 동일한 부호를 가지도록 하고 있음에 유의해야 한다. 또한, 본 실시예들을 설명함에 있어, 관련된 공지 구성 또는 기능에 대한 구체적인 설명이 본 실시예들의 요지를 흐릴 수 있다고 판단되는 경우에는 그 상세한 설명은 생략한다.

[0013] 도 1은 본 개시의 기술들을 구현할 수 있는 영상 부호화 장치에 대한 예시적인 블록도이다. 이하에서는 도 1의 도시를 참조하여 영상 부호화 장치와 이 장치의 하위 구성들에 대하여 설명하도록 한다.

[0014] 영상 부호화 장치는 픽처 분할부(110), 예측부(120), 감산기(130), 변환부(140), 양자화부(145), 재정렬부(150), 엔트로피 부호화부(155), 역양자화부(160), 역변환부(165), 가산기(170), 루프 필터부(180) 및 메모리(190)를 포함하여 구성될 수 있다.

[0015] 영상 부호화 장치의 각 구성요소는 하드웨어 또는 소프트웨어로 구현되거나, 하드웨어 및 소프트웨어의 결합으로 구현될 수 있다. 또한, 각 구성요소의 기능이 소프트웨어로 구현되고 마이크로프로세서가 각 구성요소에 대응하는 소프트웨어의 기능을 실행하도록 구현될 수도 있다.

[0016] 하나의 영상(비디오)은 복수의 픽처들을 포함하는 하나 이상의 시퀀스로 구성된다. 각 픽처들은 복수의 영역으로 분할되고 각 영역마다 부호화가 수행된다. 예를 들어, 하나의 픽처는 하나 이상의 타일(Tile) 및/또는 슬라이스(Slice)로 분할된다. 여기서, 하나 이상의 타일을 타일 그룹(Tile Group)으로 정의할 수 있다. 각 타일 또는/슬라이스는 하나 이상의 CTU(Coding Tree Unit)로 분할된다. 그리고 각 CTU는 트리 구조에 의해 하나 이상의 CU(Coding Unit)들로 분할된다. 각 CU에 적용되는 정보들은 CU의 신택스로서 부호화되고, 하나의 CTU에 포함된 CU들에 공통적으로 적용되는 정보는 CTU의 신택스로서 부호화된다. 또한, 하나의 슬라이스 내의 모든 블록들에 공통적으로 적용되는 정보는 슬라이스 헤더의 신택스로서 부호화되며, 하나 이상의 픽처들을 구성하는 모든 블록들에 적용되는 정보는 픽처 파라미터 셋(PPS, Picture Parameter Set) 혹은 픽처 헤더에 부호화된다. 나아가, 복수의 픽처가 공통으로 참조하는 정보들은 시퀀스 파라미터 셋(SPS, Sequence Parameter Set)에 부호화된다. 그리고, 하나 이상의 SPS가 공통으로 참조하는 정보들은 비디오 파라미터 셋(VPS, Video Parameter Set)에 부호화된다. 또한, 하나의 타일 또는 타일 그룹에 공통으로 적용되는 정보는 타일 또는 타일 그룹 헤더의 신택스로서 부호화될 수도 있다. SPS, PPS, 슬라이스 헤더, 타일 또는 타일 그룹 헤더에 포함되는 신택스들은 상위수준(high level) 신택스로 칭할 수 있다.

[0017] 픽처 분할부(110)는 CTU(Coding Tree Unit)의 크기를 결정한다. CTU의 크기에 대한 정보(CTU size)는 SPS 또는 PPS의 신택스로서 부호화되어 영상 복호화 장치로 전달된다.

[0018] 픽처 분할부(110)는 영상을 구성하는 각 픽처(picture)를 미리 결정된 크기를 가지는 복수의 CTU(Coding Tree Unit)들로 분할한 이후에, 트리 구조(tree structure)를 이용하여 CTU를 반복적으로(recursively) 분할한다. 트리 구조에서의 리프 노드(leaf node)가 부호화의 기본 단위인 CU(coding unit)가 된다.

[0019] 트리 구조로는 상위 노드(혹은 부모 노드)가 동일한 크기의 네 개의 하위 노드(혹은 자식 노드)로 분할되는 쿼드트리(QuadTree, QT), 또는 상위 노드가 두 개의 하위 노드로 분할되는 바이너리트리(BinaryTree, BT), 또는 상위 노드가 1:2:1 비율로 세 개의 하위 노드로 분할되는 터너리트리(TernaryTree, TT), 또는 이러한 QT 구조, BT 구조 및 TT 구조 중 둘 이상을 혼용한 구조일 수 있다. 예컨대, QTBT(QuadTree plus BinaryTree) 구조가 사용될 수 있고, 또는 QTBTTT(QuadTree plus BinaryTree TernaryTree) 구조가 사용될 수 있다. 여기서, BTTT를 합쳐서 MTT(Multiple-Type Tree)라 지칭될 수 있다.

[0020] 도 2는 QTBTTT 구조를 이용하여 블록을 분할하는 방법을 설명하기 위한 도면이다.

[0021] 도 2에 도시된 바와 같이, CTU는 먼저 QT 구조로 분할될 수 있다. 쿼드트리 분할은 분할 블록(splitting block)의 크기가 QT에서 허용되는 리프 노드의 최소 블록 크기(MinQTSize)에 도달할 때까지 반복될 수 있다. QT 구조의 각 노드가 하위 레이어의 4개의 노드들로 분할되는지 여부를 지시하는 제1 플래그(QT_split_flag)는 엔트로피 부호화부(155)에 의해 부호화되어 영상 복호화 장치로 시그널링된다. QT의 리프 노드가 BT에서 허용되는 루트 노드의 최대 블록 크기(MaxBTSize)보다 크지 않은 경우, BT 구조 또는 TT 구조 중 어느 하나 이상으로 더 분할될 수 있다. BT 구조 및/또는 TT 구조에서는 복수의 분할 방향이 존재할 수 있다. 예컨대, 해당 노드의 블록이 가로로 분할되는 방향과 세로로 분할되는 방향 두 가지가 존재할 수 있다. 도 2의 도시와 같이, MTT 분할이 시작되면, 노드들이 분할되었는지 여부를 지시하는 제2 플래그(mtt_split_flag)와, 분할이 되었다면 추가적

으로 분할 방향(vertical 혹은 horizontal)을 나타내는 플래그 및/또는 분할 타입(Binary 혹은 Ternary)을 나타내는 플래그가 엔트로피 부호화부(155)에 의해 부호화되어 영상 복호화 장치로 시그널링된다.

[0022] 대안적으로, 각 노드가 하위 레이어의 4개의 노드들로 분할되는지 여부를 지시하는 제1 플래그(QT_split_flag)를 부호화하기에 앞서, 그 노드가 분할되는지 여부를 지시하는 CU 분할 플래그(split_cu_flag)가 부호화될 수도 있다. CU 분할 플래그(split_cu_flag) 값이 분할되지 않았음을 지시하는 경우, 해당 노드의 블록이 분할 트리 구조에서의 리프 노드(leaf node)가 되어 부호화의 기본 단위인 CU(coding unit)가 된다. CU 분할 플래그(split_cu_flag) 값이 분할됨을 지시하는 경우, 영상 부호화 장치는 전술한 방식으로 제1 플래그부터 부호화를 시작한다.

[0023] 트리 구조의 다른 예시로서 QTBT가 사용되는 경우, 해당 노드의 블록을 동일 크기의 두 개 블록으로 가로로 분할하는 타입(즉, symmetric horizontal splitting)과 세로로 분할하는 타입(즉, symmetric vertical splitting) 두 가지가 존재할 수 있다. BT 구조의 각 노드가 하위 레이어의 블록으로 분할되는지 여부를 지시하는 분할 플래그(split_flag) 및 분할되는 타입을 지시하는 분할 타입 정보가 엔트로피 부호화부(155)에 의해 부호화되어 영상 복호화 장치로 전달된다. 한편, 해당 노드의 블록을 서로 비대칭 형태의 두 개의 블록으로 분할하는 타입이 추가로 더 존재할 수도 있다. 비대칭 형태에는 해당 노드의 블록을 1:3의 크기 비율을 가지는 두 개의 직사각형 블록으로 분할하는 형태가 포함될 수 있고, 혹은 해당 노드의 블록을 대각선 방향으로 분할하는 형태가 포함될 수도 있다.

[0024] CU는 CTU로부터의 QTBT 또는 QTBTTT 분할에 따라 다양한 크기를 가질 수 있다. 이하에서는, 부호화 또는 복호화하고자 하는 CU(즉, QTBTTT의 리프 노드)에 해당하는 블록을 '현재블록'이라 칭한다. QTBTTT 분할의 채용에 따라, 현재블록의 모양은 정사각형뿐만 아니라 직사각형일 수도 있다.

[0025] 예측부(120)는 현재블록을 예측하여 예측블록을 생성한다. 예측부(120)는 인트라 예측부(122)와 인터 예측부(124)를 포함한다.

[0026] 일반적으로, 픽처 내 현재블록들은 각각 예측적으로 코딩될 수 있다. 일반적으로 현재블록의 예측은 (현재블록을 포함하는 픽처로부터의 데이터를 사용하는) 인트라 예측 기술 또는 (현재블록을 포함하는 픽처 이전에 코딩된 픽처로부터의 데이터를 사용하는) 인터 예측 기술을 사용하여 수행될 수 있다. 인터 예측은 단방향 예측과 양방향 예측 모두를 포함한다.

[0027] 인트라 예측부(122)는 현재블록이 포함된 현재 픽처 내에서 현재블록의 주변에 위치한 픽셀(참조 픽셀)들을 이용하여 현재블록 내의 픽셀들을 예측한다. 예측 방향에 따라 복수의 인트라 예측모드가 존재한다. 예컨대, 도 3a에서 보는 바와 같이, 복수의 인트라 예측모드는 planar 모드와 DC 모드를 포함하는 2개의 비방향성 모드와 65개의 방향성 모드를 포함할 수 있다. 각 예측모드에 따라 사용할 주변 픽셀과 연산식이 다르게 정의된다.

[0028] 직사각형 모양의 현재블록에 대한 효율적인 방향성 예측을 위해, 도 3b에 점선 화살표로 도시된 방향성 모드들(67 ~ 80번, -1 ~ -14 번 인트라 예측모드들)이 추가로 사용될 수 있다. 이들은 "광각 인트라 예측모드들(wide angle intra-prediction modes)"로 지칭될 수 있다. 도 3b에서 화살표들은 예측에 사용되는 대응하는 참조샘플들을 가리키는 것이며, 예측 방향을 나타내는 것이 아니다. 예측 방향은 화살표가 가리키는 방향과 반대이다. 광각 인트라 예측모드들은 현재블록이 직사각형일 때 추가적인 비트 전송 없이 특정 방향성 모드를 반대방향으로 예측을 수행하는 모드이다. 이때 광각 인트라 예측모드들 중에서, 직사각형의 현재블록의 너비와 높이의 비율에 의해, 현재블록에 이용 가능한 일부 광각 인트라 예측모드들이 결정될 수 있다. 예컨대, 45도보다 작은 각도를 갖는 광각 인트라 예측모드들(67 ~ 80번 인트라 예측모드들)은 현재블록이 높이가 너비보다 작은 직사각형 형태일 때 이용 가능하고, -135도보다 큰 각도를 갖는 광각 인트라 예측모드들(-1 ~ -14 번 인트라 예측모드들)은 현재블록이 너비가 높이보다 큰 직사각형 형태일 때 이용 가능하다.

[0029] 인트라 예측부(122)는 현재블록을 부호화하는데 사용할 인트라 예측모드를 결정할 수 있다. 일부 예들에서, 인트라 예측부(122)는 여러 인트라 예측모드들을 사용하여 현재블록을 인코딩하고, 테스트된 모드들로부터 사용할 적절한 인트라 예측모드를 선택할 수도 있다. 예를 들어, 인트라 예측부(122)는 여러 테스트된 인트라 예측모드들에 대한 비트율 왜곡(rate-distortion) 분석을 사용하여 비트율 왜곡 값들을 계산하고, 테스트된 모드들 중 최선의 비트율 왜곡 특징들을 갖는 인트라 예측모드를 선택할 수도 있다.

[0030] 인트라 예측부(122)는 복수의 인트라 예측모드 중에서 하나의 인트라 예측모드를 선택하고, 선택된 인트라 예측모드에 따라 결정되는 주변 픽셀(참조 픽셀)과 연산식을 사용하여 현재블록을 예측한다. 선택된 인트라 예측모드에 대한 정보는 엔트로피 부호화부(155)에 의해 부호화되어 영상 복호화 장치로 전달된다.

[0031] 인터 예측부(124)는 움직임 보상 과정을 이용하여 현재블록에 대한 예측블록을 생성한다. 인터 예측부(124)는 현재 픽처보다 먼저 부호화 및 복호화된 참조픽처 내에서 현재블록과 가장 유사한 블록을 탐색하고, 그 탐색된 블록을 이용하여 현재블록에 대한 예측블록을 생성한다. 그리고, 현재 픽처 내의 현재블록과 참조픽처 내의 예측블록 간의 변위(displacement)에 해당하는 움직임벡터(Motion Vector: MV)를 생성한다. 일반적으로, 움직임 추정은 루마(luma) 성분에 대해 수행되고, 루마 성분에 기초하여 계산된 움직임벡터는 루마 성분 및 크로마 성분 모두에 대해 사용된다. 현재블록을 예측하기 위해 사용된 참조픽처에 대한 정보 및 움직임벡터에 대한 정보를 포함하는 움직임 정보는 엔트로피 부호화부(155)에 의해 부호화되어 영상 복호화 장치로 전달된다.

[0032] 인터 예측부(124)는, 예측의 정확성을 높이기 위해, 참조픽처 또는 참조 블록에 대한 보간을 수행할 수도 있다. 즉, 연속한 두 정수 샘플 사이의 서브 샘플들은 그 두 정수 샘플을 포함한 연속된 복수의 정수 샘플들에 필터 계수들을 적용하여 보간된다. 보간된 참조픽처에 대해서 현재블록과 가장 유사한 블록을 탐색하는 과정을 수행하면, 움직임벡터는 정수 샘플 단위의 정밀도(precision)가 아닌 소수 단위의 정밀도까지 표현될 수 있다. 움직임벡터의 정밀도 또는 해상도(resolution)는 부호화하고자 하는 대상 영역, 예컨대, 슬라이스, 타일, CTU, CU 등의 단위마다 다르게 설정될 수 있다. 이와 같은 적응적 움직임벡터 해상도(Adaptive Motion Vector Resolution: AMVR)가 적용되는 경우 각 대상 영역에 적용할 움직임벡터 해상도에 대한 정보는 대상 영역마다 시그널링되어야 한다. 예컨대, 대상 영역이 CU인 경우, 각 CU마다 적용된 움직임벡터 해상도에 대한 정보가 시그널링된다. 움직임벡터 해상도에 대한 정보는 후술할 차분 움직임벡터의 정밀도를 나타내는 정보일 수 있다.

[0033] 한편, 인터 예측부(124)는 양방향 예측(bi-prediction)을 이용하여 인터 예측을 수행할 수 있다. 양방향 예측의 경우, 두 개의 참조픽처와 각 참조픽처 내에서 현재블록과 가장 유사한 블록 위치를 나타내는 두 개의 움직임벡터가 이용된다. 인터 예측부(124)는 참조픽처 리스트 0(RefPicList0) 및 참조픽처 리스트 1(RefPicList1)로부터 각각 제1 참조픽처 및 제2 참조픽처를 선택하고, 각 참조픽처 내에서 현재블록과 유사한 블록을 탐색하여 제1 참조블록과 제2 참조블록을 생성한다. 그리고, 제1 참조블록과 제2 참조블록을 평균 또는 가중 평균하여 현재블록에 대한 예측블록을 생성한다. 그리고 현재블록을 예측하기 위해 사용한 두 개의 참조픽처에 대한 정보 및 두 개의 움직임벡터에 대한 정보를 포함하는 움직임 정보를 부호화부(150)로 전달한다. 여기서, 참조픽처 리스트 0은 기복원된 픽처들 중 디스플레이 순서에서 현재 픽처 이전의 픽처들로 구성되고, 참조픽처 리스트 1은 기복원된 픽처들 중 디스플레이 순서에서 현재 픽처 이후의 픽처들로 구성될 수 있다. 그러나 반드시 이에 한정되는 것은 아니며, 디스플레이 순서 상으로 현재 픽처 이후의 기복원 픽처들이 참조픽처 리스트 0에 추가로 더 포함될 수 있고, 역으로 현재 픽처 이전의 기복원 픽처들이 참조픽처 리스트 1에 추가로 더 포함될 수도 있다.

[0034] 움직임 정보를 부호화하는 데에 소요되는 비트량을 최소화하기 위해 다양한 방법이 사용될 수 있다.

[0035] 예컨대, 현재블록의 참조픽처와 움직임벡터가 주변블록의 참조픽처 및 움직임벡터와 동일한 경우에는 그 주변블록을 식별할 수 있는 정보를 부호화함으로써, 현재블록의 움직임 정보를 영상 복호화 장치로 전달할 수 있다. 이러한 방법을 '머지 모드(merge mode)'라 한다.

[0036] 머지 모드에서, 인터 예측부(124)는 현재블록의 주변블록들로부터 기 결정된 개수의 머지 후보블록(이하, '머지 후보'라 함)들을 선택한다.

[0037] 머지 후보를 유도하기 위한 주변블록으로는, 도 4에 도시된 바와 같이, 현재 픽처 내에서 현재블록에 인접한 좌측블록(A0), 좌하단블록(A1), 상단블록(B0), 우상단블록(B1), 및 좌상단블록(A2) 중에서 전부 또는 일부가 사용될 수 있다. 또한, 현재블록이 위치한 현재 픽처가 아닌 참조픽처(현재블록을 예측하기 위해 사용된 참조픽처와 동일할 수도 있고 다를 수도 있음) 내에 위치한 블록이 머지 후보로서 사용될 수도 있다. 예컨대, 참조픽처 내에서 현재블록과 동일 위치에 있는 블록(co-located block) 또는 그 동일 위치의 블록에 인접한 블록들이 머지 후보로서 추가로 더 사용될 수 있다. 이상에서 기술된 방법에 의해 선정된 머지 후보의 개수가 기설정된 개수보다 작으면, 0 벡터를 머지 후보에 추가한다.

[0038] 인터 예측부(124)는 이러한 주변블록들을 이용하여 기 결정된 개수의 머지 후보를 포함하는 머지 리스트를 구성한다. 머지 리스트에 포함된 머지 후보들 중에서 현재블록의 움직임정보로서 사용할 머지 후보를 선택하고 선택된 후보를 식별하기 위한 머지 인덱스 정보를 생성한다. 생성된 머지 인덱스 정보는 부호화부(150)에 의해 부호화되어 영상 복호화 장치로 전달된다.

[0039] 머지 스킵(merge skip) 모드는 머지 모드의 특별한 경우로서, 양자화를 수행한 후, 엔트로피 부호화를 위한 변환 계수가 모두 영(zero)에 가까울 때, 잔차신호들의 전송 없이 주변블록 선택 정보만을 전송한다. 머지 스킵 모드를 이용함으로써, 움직임이 적은 영상, 정지 영상, 스크린 콘텐츠 영상 등에서 상대적으로 높은 부호화 효

율을 달성할 수 있다.

[0040] 이하, 머지 모드와 머지 스킵 모드를 통칭하여, 머지/스킵 모드로 나타낸다.

[0041] 움직임 정보를 부호화하기 위한 또 다른 방법은 AMVP(Advanced Motion Vector Prediction) 모드이다.

[0042] AMVP 모드에서, 인터 예측부(124)는 현재블록의 주변블록들을 이용하여 현재블록의 움직임벡터에 대한 예측 움직임벡터 후보들을 유도한다. 예측 움직임벡터 후보들을 유도하기 위해 사용되는 주변블록으로는, 도 4에 도시된 현재 픽처 내에서 현재블록에 인접한 좌측블록(A0), 좌하단블록(A1), 상단블록(B0), 우상단블록(B1), 및 좌상단블록(A2) 중에서 전부 또는 일부가 사용될 수 있다. 또한, 현재블록이 위치한 현재 픽처가 아닌 참조픽처(현재블록을 예측하기 위해 사용된 참조픽처와 동일할 수도 있고 다를 수도 있음) 내에 위치한 블록이 예측 움직임벡터 후보들을 유도하기 위해 사용되는 주변블록으로서 사용될 수도 있다. 예컨대, 참조픽처 내에서 현재블록과 동일 위치에 있는 블록(collocated block) 또는 그 동일 위치의 블록에 인접한 블록들이 사용될 수 있다. 이상에서 기술된 방법에 의해 움직임벡터 후보의 개수가 기설정된 개수보다 작으면, 0 벡터를 움직임벡터 후보에 추가한다.

[0043] 인터 예측부(124)는 이 주변블록들의 움직임벡터를 이용하여 예측 움직임벡터 후보들을 유도하고, 예측 움직임벡터 후보들을 이용하여 현재블록의 움직임벡터에 대한 예측 움직임벡터를 결정한다. 그리고, 현재블록의 움직임벡터로부터 예측 움직임벡터를 감산하여 차분 움직임벡터를 산출한다.

[0044] 예측 움직임벡터는 예측 움직임벡터 후보들에 기 정의된 함수(예컨대, 중앙값, 평균값 연산 등)를 적용하여 구할 수 있다. 이 경우, 영상 복호화 장치도 기 정의된 함수를 알고 있다. 또한, 예측 움직임벡터 후보를 유도하기 위해 사용하는 주변블록은 이미 부호화 및 복호화가 완료된 블록이므로 영상 복호화 장치도 그 주변블록의 움직임벡터도 이미 알고 있다. 그러므로 영상 부호화 장치는 예측 움직임벡터 후보를 식별하기 위한 정보를 부호화할 필요가 없다. 따라서, 이 경우에는 차분 움직임벡터에 대한 정보와 현재블록을 예측하기 위해 사용한 참조픽처에 대한 정보가 부호화된다.

[0045] 한편, 예측 움직임벡터는 예측 움직임벡터 후보들 중 어느 하나를 선택하는 방식으로 결정될 수도 있다. 이 경우에는 차분 움직임벡터에 대한 정보 및 현재블록을 예측하기 위해 사용한 참조픽처에 대한 정보와 함께, 선택된 예측 움직임벡터 후보를 식별하기 위한 정보가 추가로 부호화된다.

[0046] 감산기(130)는 현재블록으로부터 인트라 예측부(122) 또는 인터 예측부(124)에 의해 생성된 예측블록을 감산하여 잔차블록을 생성한다.

[0047] 변환부(140)는 공간 영역의 픽셀 값들을 가지는 잔차블록 내의 잔차신호를 주파수 도메인의 변환 계수로 변환한다. 변환부(140)는 잔차블록의 전체 크기를 변환 단위로 사용하여 잔차블록 내의 잔차신호들을 변환할 수 있으며, 또는 잔차블록을 복수 개의 서브블록으로 분할하고 그 서브블록을 변환 단위로 사용하여 변환을 할 수도 있다. 또는, 변환 영역 및 비변환 영역인 두 개의 서브블록으로 구분하여, 변환 영역 서브블록만 변환 단위로 사용하여 잔차신호들을 변환할 수 있다. 여기서, 변환 영역 서브블록은 가로축 (혹은 세로축) 기준 1:1의 크기 비율을 가지는 두 개의 직사각형 블록 중 하나일 수 있다. 이런 경우, 서브블록 만을 변환하였음을 지시하는 플래그(cu_sbt_flag), 방향성(vertical/horizontal) 정보(cu_sbt_horizontal_flag) 및/또는 위치 정보(cu_sbt_pos_flag)가 엔트로피 부호화부(155)에 의해 부호화되어 영상 복호화 장치로 시그널링된다. 또한, 변환 영역 서브블록의 크기는 가로축 (혹은 세로축) 기준 1:3의 크기 비율을 가질 수 있으며, 이런 경우 해당 분할을 구분하는 플래그(cu_sbt_quad_flag)가 추가적으로 엔트로피 부호화부(155)에 의해 부호화되어 영상 복호화 장치로 시그널링된다.

[0048] 한편, 변환부(140)는 잔차블록에 대해 가로 방향과 세로 방향으로 개별적으로 변환을 수행할 수 있다. 변환을 위해, 다양한 타입의 변환 함수 또는 변환 행렬이 사용될 수 있다. 예컨대, 가로 방향 변환과 세로 방향 변환을 위한 변환 함수의 쌍을 MTS(Multiple Transform Set)로 정의할 수 있다. 변환부(140)는 MTS 중 변환 효율이 가장 좋은 하나의 변환 함수 쌍을 선택하고 가로 및 세로 방향으로 각각 잔차블록을 변환할 수 있다. MTS 중에서 선택된 변환 함수 쌍에 대한 정보(mts_idx)는 엔트로피 부호화부(155)에 의해 부호화되어 영상 복호화 장치로 시그널링된다.

[0049] 양자화부(145)는 변환부(140)로부터 출력되는 변환 계수들을 양자화 파라미터를 이용하여 양자화하고, 양자화된 변환 계수들을 엔트로피 부호화부(155)로 출력한다. 양자화부(145)는, 어떤 블록 혹은 프레임에 대해, 변환 없이, 관련된 잔차 블록을 곧바로 양자화할 수도 있다. 양자화부(145)는 변환블록 내의 변환 계수들의 위치에 따라 서로 다른 양자화 계수(스케일링 값)을 적용할 수도 있다. 2차원으로 배열된 양자화된 변환 계수들에 적용되

는 양자화 행렬은 부호화되어 영상 복호화 장치로 시그널링될 수 있다.

[0050] 재정렬부(150)는 양자화된 잔차값에 대해 계수값의 재정렬을 수행할 수 있다.

[0051] 재정렬부(150)는 계수 스캐닝(coefficient scanning)을 이용하여 2차원의 계수 어레이를 1차원의 계수 시퀀스로 변경할 수 있다. 예를 들어, 재정렬부(150)에서는 지그-재그 스캔(zig-zag scan) 또는 대각선 스캔(diagonal scan)을 이용하여 DC 계수부터 고주파수 영역의 계수까지 스캔하여 1차원의 계수 시퀀스를 출력할 수 있다. 변환 단위의 크기 및 인트라 예측모드에 따라 지그-재그 스캔 대신 2차원의 계수 어레이를 열 방향으로 스캔하는 수직 스캔, 2차원의 블록 형태 계수를 행 방향으로 스캔하는 수평 스캔이 사용될 수도 있다. 즉, 변환 단위의 크기 및 인트라 예측모드에 따라 지그-재그 스캔, 대각선 스캔, 수직 방향 스캔 및 수평 방향 스캔 중에서 사용될 스캔 방법이 결정될 수도 있다.

[0052] 엔트로피 부호화부(155)는, CABAC(Context-based Adaptive Binary Arithmetic Code), 지수 골롬(Exponential Golomb) 등의 다양한 부호화 방식을 사용하여, 재정렬부(150)로부터 출력된 1차원의 양자화된 변환 계수들의 시퀀스를 부호화함으로써 비트스트림을 생성한다.

[0053] 또한, 엔트로피 부호화부(155)는 블록 분할과 관련된 CTU size, CU 분할 플래그, QT 분할 플래그, MTT 분할 타입, MTT 분할 방향 등의 정보를 부호화하여, 영상 복호화 장치가 영상 부호화 장치와 동일하게 블록을 분할할 수 있도록 한다. 또한, 엔트로피 부호화부(155)는 현재블록이 인트라 예측에 의해 부호화되었는지 아니면 인터 예측에 의해 부호화되었는지 여부를 지시하는 예측 타입에 대한 정보를 부호화하고, 예측 타입에 따라 인트라 예측정보(즉, 인트라 예측모드에 대한 정보) 또는 인터 예측정보(움직임 정보의 부호화 모드(머지 모드 또는 AMVP 모드), 머지 모드의 경우 머지 인덱스, AMVP 모드의 경우 참조픽처 인덱스 및 차분 움직임벡터에 대한 정보)를 부호화한다. 또한, 엔트로피 부호화부(155)는 양자화와 관련된 정보, 즉, 양자화 파라미터에 대한 정보 및 양자화 행렬에 대한 정보를 부호화한다.

[0054] 역양자화부(160)는 양자화부(145)로부터 출력되는 양자화된 변환 계수들을 역양자화하여 변환 계수들을 생성한다. 역변환부(165)는 역양자화부(160)로부터 출력되는 변환 계수들을 주파수 도메인으로부터 공간 도메인으로 변환하여 잔차블록을 복원한다.

[0055] 가산부(170)는 복원된 잔차블록과 예측부(120)에 의해 생성된 예측블록을 가산하여 현재블록을 복원한다. 복원된 현재블록 내의 픽셀들은 다음 순서의 블록을 인트라 예측할 때 참조 픽셀로서 사용된다.

[0056] 루프(loop) 필터부(180)는 블록 기반의 예측 및 변환/양자화로 인해 발생하는 블록킹 아티팩트(blocking artifacts), 링잉 아티팩트(ringing artifacts), 블러링 아티팩트(blurring artifacts) 등을 줄이기 위해 복원된 픽셀들에 대한 필터링을 수행한다. 필터부(180)는 인루프(in-loop) 필터로서 디블록킹 필터(182), SAO(Sample Adaptive Offset) 필터(184) 및 ALF(Adaptive Loop Filter, 186)의 전부 또는 일부를 포함할 수 있다.

[0057] 디블록킹 필터(182)는 블록 단위의 부호화/복호화로 인해 발생하는 블록킹 현상(blocking artifact)을 제거하기 위해 복원된 블록 간의 경계를 필터링하고, SAO 필터(184) 및 alf(186)는 디블록킹 필터링된 영상에 대해 추가적인 필터링을 수행한다. SAO 필터(184) 및 alf(186)는 손실 부호화(lossy coding)로 인해 발생하는 복원된 픽셀과 원본 픽셀 간의 차이를 보상하기 위해 사용되는 필터이다. SAO 필터(184)는 CTU 단위로 오프셋을 적용함으로써 주관적 화질뿐만 아니라 부호화 효율도 향상시킨다. 이에 비하여 ALF(186)는 블록 단위의 필터링을 수행하는데, 해당 블록의 에지 및 변화량의 정도를 구분하여 상이한 필터를 적용하여 왜곡을 보상한다. ALF에 사용될 필터 계수들에 대한 정보는 부호화되어 영상 복호화 장치로 시그널링될 수 있다.

[0058] 디블록킹 필터(182), SAO 필터(184) 및 ALF(186)를 통해 필터링된 복원블록은 메모리(190)에 저장된다. 한 픽처 내의 모든 블록들이 복원되면, 복원된 픽처는 이후에 부호화하고자 하는 픽처 내의 블록을 인터 예측하기 위한 참조픽처로 사용될 수 있다.

[0059] 도 5는 본 개시의 기술들을 구현할 수 있는 영상 복호화 장치의 예시적인 블록도이다. 이하에서는 도 5를 참조하여 영상 복호화 장치와 이 장치의 하위 구성들에 대하여 설명하도록 한다.

[0060] 영상 복호화 장치는 엔트로피 복호화부(510), 재정렬부(515), 역양자화부(520), 역변환부(530), 예측부(540), 가산기(550), 루프 필터부(560) 및 메모리(570)를 포함하여 구성될 수 있다.

[0061] 도 1의 영상 부호화 장치와 마찬가지로, 영상 복호화 장치의 각 구성요소는 하드웨어 또는 소프트웨어로 구현되거나, 하드웨어 및 소프트웨어의 결합으로 구현될 수 있다. 또한, 각 구성요소의 기능이 소프트웨어로 구현되고

마이크로프로세서가 각 구성요소에 대응하는 소프트웨어의 기능을 실행하도록 구현될 수도 있다.

[0062] 엔트로피 복호화부(510)는 영상 부호화 장치에 의해 생성된 비트스트림을 복호화하여 블록 분할과 관련된 정보를 추출함으로써 복호화하고자 하는 현재블록을 결정하고, 현재블록을 복원하기 위해 필요한 예측정보와 잔차신호에 대한 정보 등을 추출한다.

[0063] 엔트로피 복호화부(510)는 SPS(Sequence Parameter Set) 또는 PPS(Picture Parameter Set)로부터 CTU size에 대한 정보를 추출하여 CTU의 크기를 결정하고, 픽처를 결정된 크기의 CTU로 분할한다. 그리고, CTU를 트리 구조의 최상위 레이어, 즉, 루트 노드로 결정하고, CTU에 대한 분할정보를 추출함으로써 트리 구조를 이용하여 CTU를 분할한다.

[0064] 예컨대, QTBTTT 구조를 사용하여 CTU를 분할하는 경우, 먼저 QT의 분할과 관련된 제1 플래그(QT_split_flag)를 추출하여 각 노드를 하위 레이어의 네 개의 노드로 분할한다. 그리고, QT의 리프 노드에 해당하는 노드에 대해서는 MTT의 분할과 관련된 제2 플래그(MTT_split_flag) 및 분할 방향(vertical / horizontal) 및/또는 분할 타입(binary / ternary) 정보를 추출하여 해당 리프 노드를 MTT 구조로 분할한다. 이에 따라 QT의 리프 노드 이하의 각 노드들을 BT 또는 TT 구조로 반복적으로(recursively) 분할한다.

[0065] 또 다른 예로서, QTBTTT 구조를 사용하여 CTU를 분할하는 경우, 먼저 CU의 분할 여부를 지시하는 CU 분할 플래그(split_cu_flag)를 추출하고, 해당 블록이 분할된 경우, 제1 플래그(QT_split_flag)를 추출할 수도 있다. 분할 과정에서 각 노드는 0번 이상의 반복적인 QT 분할 후에 0번 이상의 반복적인 MTT 분할이 발생할 수 있다. 예컨대, CTU는 바로 MTT 분할이 발생하거나, 반대로 다수 번의 QT 분할만 발생할 수도 있다.

[0066] 다른 예로서, QTBT 구조를 사용하여 CTU를 분할하는 경우, QT의 분할과 관련된 제1 플래그(QT_split_flag)를 추출하여 각 노드를 하위 레이어의 네 개의 노드로 분할한다. 그리고, QT의 리프 노드에 해당하는 노드에 대해서는 BT로 더 분할되는지 여부를 지시하는 분할 플래그(split_flag) 및 분할 방향 정보를 추출한다.

[0067] 한편, 엔트로피 복호화부(510)는 트리 구조의 분할을 이용하여 복호화하고자 하는 현재블록을 결정하게 되면, 현재블록이 인트라 예측되었는지 아니면 인터 예측되었는지를 지시하는 예측 타입에 대한 정보를 추출한다. 예측 타입 정보가 인트라 예측을 지시하는 경우, 엔트로피 복호화부(510)는 현재블록의 인트라 예측정보(인트라 예측모드)에 대한 신택스 요소를 추출한다. 예측 타입 정보가 인터 예측을 지시하는 경우, 엔트로피 복호화부(510)는 인터 예측정보에 대한 신택스 요소, 즉, 움직임벡터 및 그 움직임벡터가 참조하는 참조픽처를 나타내는 정보를 추출한다.

[0068] 또한, 엔트로피 복호화부(510)는 양자화 관련된 정보, 및 잔차신호에 대한 정보로서 현재블록의 양자화된 변환계수들에 대한 정보를 추출한다.

[0069] 재정렬부(515)는, 영상 부호화 장치에 의해 수행된 계수 스캐닝 순서의 역순으로, 엔트로피 복호화부(510)에서 엔트로피 복호화된 1차원의 양자화된 변환계수들의 시퀀스를 다시 2차원의 계수 어레이(즉, 블록)로 변경할 수 있다.

[0070] 역양자화부(520)는 양자화된 변환계수들을 역양자화하고, 양자화 파라미터를 이용하여 양자화된 변환계수들을 역양자화한다. 역양자화부(520)는 2차원으로 배열된 양자화된 변환계수들에 대해 서로 다른 양자화 계수(스케일링 값)을 적용할 수도 있다. 역양자화부(520)는 영상 부호화 장치로부터 양자화 계수(스케일링 값)들의 행렬을 양자화된 변환계수들의 2차원 어레이에 적용하여 역양자화를 수행할 수 있다.

[0071] 역변환부(530)는 역양자화된 변환계수들을 주파수 도메인으로부터 공간 도메인으로 역변환하여 잔차신호들을 복원함으로써 현재블록에 대한 잔차블록을 생성한다.

[0072] 또한, 역변환부(530)는 변환블록의 일부 영역(서브블록)만 역변환하는 경우, 변환블록의 서브블록만을 변환하였음을 지시하는 플래그(cu_sbt_flag), 서브블록의 방향성(vertical/horizontal) 정보(cu_sbt_horizontal_flag) 및/또는 서브블록의 위치 정보(cu_sbt_pos_flag)를 추출하여, 해당 서브블록의 변환계수들을 주파수 도메인으로부터 공간 도메인으로 역변환함으로써 잔차신호들을 복원하고, 역변환되지 않은 영역에 대해서는 잔차신호로 "0"값을 채움으로써 현재블록에 대한 최종 잔차블록을 생성한다.

[0073] 또한, MTS가 적용된 경우, 역변환부(530)는 영상 부호화 장치로부터 시그널링된 MTS 정보(mts_idx)를 이용하여 가로 및 세로 방향으로 각각 적용할 변환 함수 또는 변환 행렬을 결정하고, 결정된 변환 함수를 이용하여 가로 및 세로 방향으로 변환블록 내의 변환계수들에 대해 역변환을 수행한다.

[0074] 예측부(540)는 인트라 예측부(542) 및 인터 예측부(544)를 포함할 수 있다. 인트라 예측부(542)는 현재블록의 예측 타입이 인트라 예측일 때 활성화되고, 인터 예측부(544)는 현재블록의 예측 타입이 인터 예측일 때 활성화된다.

[0075] 인트라 예측부(542)는 엔트로피 복호화부(510)로부터 추출된 인트라 예측모드에 대한 신택스 요소로부터 복수의 인트라 예측모드 중 현재블록의 인트라 예측모드를 결정하고, 인트라 예측모드에 따라 현재블록 주변의 참조 픽셀들을 이용하여 현재블록을 예측한다.

[0076] 인터 예측부(544)는 엔트로피 복호화부(510)로부터 추출된 인터 예측모드에 대한 신택스 요소를 이용하여 현재블록의 움직임벡터와 그 움직임벡터가 참조하는 참조픽처를 결정하고, 움직임벡터와 참조픽처를 이용하여 현재블록을 예측한다.

[0077] 가산기(550)는 역변환부로부터 출력되는 잔차블록과 인터 예측부 또는 인트라 예측부로부터 출력되는 예측블록을 가산하여 현재블록을 복원한다. 복원된 현재블록 내의 픽셀들은 이후에 복호화할 블록을 인트라 예측할 때의 참조픽셀로서 활용된다.

[0078] 루프 필터부(560)는 인루프 필터로서 디블록킹 필터(562), SAO 필터(564) 및 ALF(566)를 포함할 수 있다. 디블록킹 필터(562)는 블록 단위의 복호화로 인해 발생하는 블록킹 현상(blocking artifact)을 제거하기 위해, 복원된 블록 간의 경계를 디블록킹 필터링한다. SAO 필터(564) 및 ALF(566)는 손실 부호화(lossy coding)으로 인해 발생하는 복원된 픽셀과 원본 픽셀 간의 차이를 보상하기 위해, 디블록킹 필터링 이후의 복원된 블록에 대해 추가적인 필터링을 수행한다. ALF의 필터 계수는 비스트림으로부터 복호한 필터 계수에 대한 정보를 이용하여 결정된다.

[0079] 디블록킹 필터(562), SAO 필터(564) 및 ALF(566)를 통해 필터링된 복원블록은 메모리(570)에 저장된다. 한 픽처 내의 모든 블록들이 복원되면, 복원된 픽처는 이후에 부호화하고자 하는 픽처 내의 블록을 인터 예측하기 위한 참조픽처로 사용된다.

[0080] 본 실시예는 이상에서 설명한 바와 같은 영상(비디오)의 부호화 및 복호화에 관한 것이다. 보다 자세하게는, 입력 비디오 블록을 딥러닝 모델인 트랜스포머(transformer)의 어텐션 모듈(attention module)에 적용하고, 이에 따른 트랜스포머 기반 인루프 필터를 이용하는 비디오 코딩방법 및 장치를 제공한다.

[0081] 이하의 실시예들은 영상 부호화 장치(video encoding device) 내 루프 필터부(180)에 의해 수행될 수 있다. 또한, 영상 복호화 장치(video decoding device) 내 루프 필터부(542)에 의해 수행될 수 있다.

[0082] 영상 부호화 장치는, 복원 프레임의 인루프 필터링에 있어서, 비트율 왜곡 최적화 측면에서 본 실시예와 관련된 시그널링 정보를 생성할 수 있다. 영상 부호화 장치는 엔트로피 부호화부(155)를 이용하여 이를 부호화한 후, 영상 복호화 장치로 전송할 수 있다. 영상 복호화 장치는 엔트로피 복호화부(510)를 이용하여 비트스트림으로부터 복원 프레임의 인루프 필터링과 관련된 시그널링 정보를 복호화할 수 있다.

[0083] 이하의 설명에서, '대상 블록'이라는 용어는 현재블록 또는 코딩유닛(CU, Coding Unit)과 동일한 의미로 사용될 수 있고, 또는 코딩유닛의 일부 영역을 의미할 수도 있다.

[0084] 또한, 하나의 플래그의 값이 참이라는 것은 플래그가 1로 설정되는 경우를 나타낸다. 또한, 하나의 플래그의 값이 거짓이라는 것은 플래그가 0으로 설정되는 경우를 나타낸다.

[0085] I. CNN(Convolutional Neural Network)

[0086] CNN은 복수의 콘볼루션 레이어(convolution layer)와 풀링 레이어(pooling layer)로 구성된 신경망을 지칭하고, 영상 처리에 가장 적합한 것으로 알려진 딥러닝 기술이다. 콘볼루션 레이어는 다수의 커널(kernel) 또는 필터(filter)를 이용하여 특징맵(feature map, 또는 '특성'과 호환적으로 이용)을 추출한다. 이때 필터를 구성하는 커널 계수(kernel coefficient)가 학습 과정에서 결정되는 파라미터이다.

[0087] CNN의 콘볼루션 레이어 중에서 입력에 가까운 전단 레이어는 선, 점, 또는 면과 같은 단순하고 낮은(lower) 수준의 영상 특징에 반응하는 특징맵을 추출하고, 출력에 가까운 후단 계층은 텍스처(texture), 사물 일부(object parts) 등 높은(higher) 수준에 반응하는 특징맵을 추출한다.

[0088] 도 6은 본 개시의 일 실시예에 따른 콘볼루션 레이어의 연산을 나타내는 예시도이다.

[0089] 콘볼루션 레이어는 콘볼루션 연산을 이용하여 입력 영상으로부터 특징맵을 생성한다. 도 6의 예시에는, 커널 크

기가 3×3인 커널(또는 필터)이 표현되어 있다. 커널 크기는 커널 사이즈(kernel size) 또는 필터 사이즈(filter size)로도 지칭된다. 커널은 가중치(weight)로도 불리우는 커널 파라미터(kernel parameter 또는 filter parameter)를 갖는다. 도 6에 예시된 커널은 총 9 개의 커널 파라미터를 갖는다. 커널 파라미터는 초기에 임의의 값으로 설정되고, 트레이닝(learning)을 기반으로 그 값이 업데이트될 수 있다.

[0090] 콘볼루션 레이어는 입력 영상에서 커널 사이즈만큼의 블록을 이용하여 콘볼루션 연산을 수행한다. 이때, 입력 영상 내의 커널 사이즈만큼의 블록을 윈도우(window)로 지칭한다.

[0091] 입력 영상에 대한 필터링을 래스터 스캔 순서(raster-scan order)로 수행할 때, 윈도우의 이동 크기를 스트라이드(stride)라고 한다. 도 6의 예시에서, 스트라이드는 1이다. 만약 스트라이드를 2로 설정하면, 윈도우를 2 샘플씩 띄어서 콘볼루션 연산을 수행하고, 결과적으로 특징맵의 가로와 세로의 크기는 입력 영상의 가로와 세로의 크기의 반이 된다.

[0092] 전술한 바와 같이, 하나의 콘볼루션 레이어는 다수의 필터를 포함할 수 있다. 필터의 개수(number of filter/filters) 또는 커널의 개수(number of kernel/kernels)를 채널(channel)이라고 한다. 즉, 채널의 개수는 필터의 개수와 동일하다. 또한 필터의 개수는 특징맵의 차원(dimension)의 크기를 결정한다.

[0093] 패딩(padding)은 콘볼루션 연산을 수행하기 전, 입력 데이터 주변을 특정값으로 채워서 입력 데이터를 확장하는 방법을 나타낸다. 패딩은 주로 출력 데이터의 공간적(spatial) 크기를 조절하기 위해 사용된다. 패딩 시 이용되는 값은 하이퍼파라미터에 의해 결정될 수 있으나, 주로 제로패딩(zero-padding)이 이용된다. 패딩을 사용하지 않을 경우, 콘볼루션 레이어를 거칠 때마다 출력 데이터의 공간적 크기가 감소하여, 경계의 정보들이 사라지는 문제가 발생할 수 있다. 따라서, 이러한 문제를 방지하기 위해, 패딩이 사용된다. 즉, 콘볼루션 레이어의 출력 데이터와 입력 데이터의 공간적 크기를 동일하게 맞추기 위해 패딩이 사용될 수 있다.

[0094] 디콘볼루션 레이어는 콘볼루션 레이어와 반대되는 연산을 한다. 디콘볼루션 레이어는 입력인 특징맵으로부터 원하는 데이터 영상을 출력으로 생성한다.

[0095] 풀링 레이어는 콘볼루션 레이어에 의해 생성된 특징맵을 서브샘플링(sub sampling)하는 과정인 풀링을 수행한다. 풀링 레이어는 2×2 크기의 윈도우를 이용하여 출력 결과가 입력의 가로와 세로 각각의 절반이 되도록 샘플을 선택한다. 즉, 풀링 계층은 2×2 영역을 샘플 하나로 집약하여 입력 영상 또는 입력 특징맵의 크기를 축소하기 위해 이용된다.

[0096] 풀링 레이어의 반대 개념은 언풀링(unpooling) 레이어로 정의한다. 언풀링 레이어는 풀링 레이어와 반대로 차원을 확대하는 역할을 하고, 주로 디콘볼루션 레이어 이후에 사용된다.

[0097] 콘볼루션 인코더(convolutional encoder)-디코더(decoder) 구조는 콘볼루션 레이어들와 디콘볼루션 레이어들의 짝(pair)으로 구성되는 네트워크 구조이다. 콘볼루션 인코더는 콘볼루션 레이어와 풀링 레이어로 구성되어, 입력 영상으로부터 특징맵(또는 특징 벡터)을 출력한다. 콘볼루션 인코더의 최종 출력 벡터를 잠재 벡터(latent vector)로도 지칭한다. 콘볼루션 디코더(convolutional decoder)는 디콘볼루션 레이어와 언풀링 레이어로 구성되어, 특징맵 또는 잠재 벡터로부터 출력 영상을 생성한다.

[0098] 콘볼루션 인코더-디코더의 입력과 출력은 애플리케이션과 네트워크의 목적에 따라 다양하게 설정될 수 있다. 예컨대, 입력과 출력은 옵티컬 플로우 맵(optical flow map), 돌출 맵(saliency map), 영상 프레임(image frame) 등일 수 있다.

[0099] 도 7은 SISR 네트워크를 나타내는 예시도이다.

[0100] CNN이 적용되는 일 예로서 SISR(Single Image Super Resolution)이 있다. SISR 네트워크는 저해상도 입력 영상으로부터 고해상도 이미지를 출력으로 생성한다. SISR 네트워크는 도 7에 예시된 바와 같이, 다수의 콘볼루션 레이어들을 포함할 수 있다. 각 콘볼루션 레이어는 ReLU(Rectified Linear Unit)과 같은 활성화 함수(activation function)를 포함한다. SISR 네트워크의 파라미터들은 생성되는 SR(Super Resolution) 영상이 GT(Ground Truth)에 근접하도록 트레이닝될 수 있다.

[0101] CNN을 이용한 SR 방식들은 깊이를 증가시켜(예컨대, 콘볼루션 레이어의 개수를 증가시키는 것을 의미함) SR 성능을 향상시킬 수 있다. 깊이의 증가에 따라 발생할 수 있는 학습에서의 과적합(overfitting) 문제를 극복하기 위해, 스킵 연결(skip connection) 및 잔차 학습(residual learning)을 수행할 수 있는 잔차 블록(residual block)이 SISR 네트워크에 이용될 수 있다. 잔차 블록은, 도 8에 예시된 바와 같이, 입력 특징 x_l에 콘볼루션

연산을 적용하는 경로 외에 스킵 경로를 포함한다. 또한, 잔차 블록은 출력 x_{l+1}을 생성 시, 학습 효율에 기초하여 콘볼루션 연산을 적용하는 경로 또는 스킵 경로를 선택할 수도 있다. 도 8의 예시에서, 잔차 블록은 BN(Batch Normalization) 레이어를 포함한다.

[0102] 일 예로서, EDSR(Enhanced Deep residual networks for SISR)은 잔차 블록들을 연속적으로 연결하여 깊이를 증가시킴으로써, 네트워크의 성능을 증가시킨다. 다른 예로서, VDSR(Accurate Image Super-Resolution Using Very Deep Convolutional Networks)은 VGG(Visual Geometry Group) 네트워크 기반의 CNN 모델로서, 잔차 프레임을 최종 출력에 더해 주는 방식인 잔차신호 기반 학습(residual learning)을 사용한다. VDSR은 네트워크의 맨 마지막에 잔차 신호을 추가하여, 입력 신호에 잔차 신호를 더한다.

[0103] 다른 예로서, CNN은 영상 부호화 장치 또는 영상 복호화 장치에서 인루프 필터로 이용될 수 있다. 이때, 딥러닝 기반 인루프 필터는, 디블로킹 필터, SAO 필터, 및 ALF로 구성된 기존 루프 필터부(180, 560) 내 어느 위치에든 적용이 될 수 있다.

[0104] 딥러닝 기반 인루프 필터로서 고정 계수 인루프 필터가 있다. 고정 계수 딥러닝 인루프 필터는, 영상 부호화 장치 및 영상 복호화 장치 측에 저장된 동일한 CNN 커널 파라미터들을 사용한다.

[0105] 도 9는 CNN 기반 고정 계수 인루프 필터를 나타내는 예시도이다.

[0106] 입력 블록(또는 입력 프레임)은 정규화 QP 맵(normalized QP map)을 거친 후, 이후의 단계에 전달된다. 정규화 QP 맵은 학습과 추론 과정에서 서로 다른 강도의 양자화 잡음이 섞인 경우, 추론 오차를 감소시키기 위하여 사용된다. DRU(Dense Residual Unit) 및 콘볼루션 레이어를 구성하는 커널 파리미터들은, 영상 부호화 장치 및 영상 복호화 장치에 동일하게 저장된 채로 사용될 수 있다. 도 9의 예시에서, 각 DRU는 콘볼루션 레이어, ReLU 레이어, 및 DSC(depth-wise separable convolutional) 레이어들의 전부 또는 일부를 포함할 수 있다.

[0107] 고정 계수 딥러닝 인루프 필터는 다양한 비디오 프레임에 대해 일반적인 성능을 제공해야 함으로, CNN의 레이어의 개수가 깊어지고 그에 따라 연산 시간이 길어진다는 단점을 갖는다.

[0108] **II. 트랜스포머**

[0109] 트랜스포머(transformer)는 자연어 처리에서 기계 번역 등의 시퀀스-투-시퀀스(sequence-to-sequence) 과제를 수행하기 위해 제안된 모델이다.

[0110] 기존 시퀀스-투-시퀀스 모델은 RNN(Recurrent Neural Network) 또는 LSTM(Long Short-Term Memory)을 이용하여 입력 단어가 입력되면 출력 단어를 순서대로 출력하는 인코더-디코더 구조을 포함한다. 인코더는 입력 시퀀스를 벡터 표현으로 표현하고 디코더는 이 벡터 표현에 기초하여 출력 시퀀스를 생성한다.

[0111] 하지만, RNN 기반의 모델은 입력 벡터의 길이가 제한된다는 한계를 보인다. 또한, 인코딩 시 입력 시퀀스 정보가 일부 손실되기 때문에, RNN 모델은 복잡한 문장 처리에 한계를 보인다. 이를 보완하기 위해, 디코더의 특정 시간에서의 출력과 인코더의 출력과의 연관성을 파악하는 어텐션(Attention) 메커니즘에 기초하여 맥락 벡터(context vector)를 생성함에도, RNN 모델은 전술한 한계를 극복하지는 못한다.

[0112] 트랜스포머는 어텐션만으로 기존의 인코더-디코더 구조를 유지하고, RNN 또는 LSTM를 사용하지 않음에도 번역 성능에서 기존 모델보다 우수한 성능을 보인다.

[0113] 먼저, 트랜스포머의 전체적인 구조를 기술하기 전에, 트랜스포머에서 이용되는 셀프 어텐션을 기술한다.

[0114] 도 10a 및 도 10b는 벡터의 셀프 어텐션을 나타내는 예시도이다.

[0115] 셀프 어텐션(sef attention)은 트랜스포머 내 어텐션 모듈(또는 '어텐션 층')에 의해 수행된다. 이하, 트랜스포머가 "I am a student"라는 문장을 순차적으로 처리하되, 어텐션 모듈이 현재 'student'라는 입력 단어를 처리하는 것을 가정한다.

[0116] 먼저, 어텐션 모듈은 도 10a에 예시된 바와 같이, 트랜스포머에 초기 입력된 각 단어에 해당하는 단어 벡터(word vector)에 가중치 행렬을 곱하여, Q(Query) 벡터, K(Key) 벡터, 및 V(Value) 벡터를 생성한다. 도 10a의 예시에서, 가중치 행렬 W^Q, W^K 및 W^V는 학습된 파라미터들이다. 또한, 입력 단어 'student'에 대해 $Q_{student}$, $K_{student}$ 및 $V_{student}$는 각각 Q 벡터, K 벡터 및 V 벡터를 나타낸다.

[0117] 이후, 어텐션 모듈은 주어진 Q 벡터에 대해 모든 K 벡터들과의 어텐션 스코어(attention score)들을 산정한다.

이후, 어텐션 모듈은 해당 유사도, 즉 어텐션 스코어를 가중치로 이용하여, 각 K 벡터와 매핑된 V 벡터에 반영한다. 마지막으로, 어텐션 모듈은 유사도가 반영된 V 벡터들을 모두 가중합하여, 입력 단어에 대해 어텐션 값(attention value)을 산정할 수 있다.

[0118] 한편, 어텐션 스코어는 도 10b의 예시와 같이, 두 벡터의 내적을 이용하는 scaled dot-product attention에 의해 산정될 수 있다. scaled dot-product attention은 수학식 1과 같이 표현될 수 있다.

수학식 1

[0119]
$$score\ function(q,k) = q \cdot k / \sqrt{d_k}$$

[0120] 여기서, d_k는 사전에 정의되는 상수이다.

[0121] 어텐션 스코어 함수는 q 벡터와 k 벡터 간의 연관도를 표현한다. 수학식 1에서 스코어 함수는 내적을 이용하나, 코사인 유사도(cosine similarity), 상관도(correlation) 등과 같은 두 벡터들 간의 유사도 또한 이용될 수 있다.

[0122] 위와 같은 연산은, 도 11a 및 도 11b의 예시와 같이, 'student' 이외에도 'I', 'am', 및 'a' 모두에 병렬적으로 적용될 수 있다. 도 11a 및 도 11b의 예시에서, 입력 문장 "I am a student"에 대해 Q, K 및 V는 각각 Q 행렬, K 행렬 및 V 행렬을 나타낸다. 최종적인 어텐션 값 행렬은 수학식 2와 같이 산정될 수 있다.

수학식 2

[0123]
$$\mathrm{Attention}(\mathrm{Q},\mathrm{K},\mathrm{V}) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}} + B\right)V$$

[0124] 여기서, B는 위치 인코딩 값(positional encoding value)의 행렬로서, 사전에 정의된다. 도 11b의 예시에는 B가 생략되어 있다.

[0125] 도 12는 멀티헤드 어텐션을 나타내는 예시도이다.

[0126] 한편, 한 번의 어텐션보다 병렬 어텐션을 수행하는 것이, 학습의 일반화에 더 효과적이다. 이는 상이한 가중치들을 이용하여 각 단어 벡터에 대해 다른 시각에서 정보를 수집할 수 있기 때문이다. 트랜스포머에서는 병렬 어텐션이 멀티헤드 어텐션(multi-head attention)으로 표현된다.

[0127] 멀티헤드 어텐션을 적용하기 위해, 가중치 행렬 W^Q, W^K 및 W^V이 어텐션 헤드의 개수(num_heads)만큼 학습될 수 있다. 도 12에 예시된 바와 같이 가중치 행렬 $W^Q_{0...num_head-1}$, $W^K_{0...num_head-1}$ 및 $W^V_{0...num_head-1}$를 이용하여 병렬 어텐션이 수행될 수 있다. 이후, 각 헤드에서의 어텐션 출력을 연결(concatenate)하고, 가중치 행렬을 곱하여 최종 어텐션이 생성될 수 있다.

[0128] 도 13은 트랜스포머의 구조를 나타내는 예시도이다.

[0129] 트랜스포머는 N 개의 인코더와 N 개의 디코더를 포함한다.

[0130] 인코더는 N 개의 층(layer)들을 포함한다. 하나의 층은 두 개의 서브층(sulayer)들, 즉 셀프 어텐션(self-attention) 및 피드포워드(feed forward) 신경망을 포함한다. 여기서, 셀프 어텐션은 Q, K 및 V가 동일한 경우를 나타낸다. 또한, 인코더의 셀프 어텐션 층은 전술한 바와 같은 멀티헤드 어텐션을 수행한다.

[0131] 한편, 인코더의 입력 문장에는 위치 인코딩(positional encoding)이 적용된다. 트랜스포머는 단어를 순차적으로 입력받지 않을 수 있다. 따라서, 단어의 위치 정보를 표현하기 위해 사인(sinusoid) 함수를 이용하여 토큰(token) 순서대로 위치 정보가 추가된다. 도 13의 예시에서, 'input embedding'은 임베딩 형태로 표현된 입력 문장을 나타낸다.

[0132] 디코더는 인코더와 동일하게 N 개의 층들을 포함한다. 하나의 층은 세 개의 서브층들을 포함한다. 인코더와 마찬가지로, 포지셔널 인코딩 후의 문장 행렬이 디코더에 입력된다. 도 13의 예시에서, 'output embedding'은 임베딩 형태로 표현된 디코더의 입력 문장을 나타낸다.

[0133] 시퀀스-투-시퀀스와 마찬가지로 디코더의 첫 번째 서브층은 GT를 디코더의 다음 입력으로 하는 티처 포싱(teacher forcing)에 기초하여 트레이닝되므로, 디코더는 타깃 문장 행렬을 입력받아 각 시점의 단어를 예측하도록 트레이닝된다. 입력 문장에 기초하는 행렬에서 미래 시점의 단어를 참조하지 않도록 마스킹을 적용하여, 자기 자신과 그 이전 단어들이 참조된다. 근본적으로, 셀프 어텐션이라는 점과 멀티헤드 어텐션을 수행한다는 점에서 디코더의 첫 번째 서브층은 인코더의 첫 번째 서브층과 구조적으로 동일하다. 디코더의 첫 번째 서브층의 동작을 룩 어헤드 어텐션(look-ahead attention)이라 한다.

[0134] 디코더의 두 번째 서브층은, 첫 번째 셀프 어텐션 층의 출력을 Q 행렬로 이용하고, 인코더의 마지막 층에서 온 K 행렬과 V 행렬을 사용하여 어텐션 스코어 행렬을 계산한다. 디코더의 두 번째 서브층은 인코더-디코더 어텐션(encoder-decoder attention)이라 한다. 한편, 세 번째 서브 층은 피드포워드(feed forward) 신경망이다.

[0135] 비전 분야에서는 CNN 구조가 많이 사용된다. 자연언어처리(Natural Language Processing)에서 트랜스포머의 활용 성과에 기초하여, 스탠다드 트랜스포머를 최소 한도로 수정한 후, 수정된 트랜스포머가 직접 이미지에 적용될 수 있다. 이를 위해, 이미지를 패치들로 분할하고, 패치들은 NLP에서의 토큰과 같은 방식으로 처리될 수 있다. 즉, 이러한 패치들의 선형 임베딩 시퀀스(linear embedding sequence)가 트랜스포머에 입력될 수 있다. 비전 분야 활용되는 트랜스포머를 비전 트랜스포머(vision transformer)로 명칭한다.

[0136] 도 14는 비전 트랜스포머의 입력 전처리를 나타내는 예시도이다.

[0137] 스탠다드 트랜스포머는 토큰 임베딩들의 1차원 시퀀스를 입력으로 획득한다. 비전 트랜스포머에 1차원 시퀀스를 입력하기 위해 2차원 이미지가 전처리되어야 한다. 이를 위해, 도 14에 예시된 바와 같이, 2차원 입력 이미지 $x \in R^{W \times H \times C}$가 평탄화된(flattened) 패치들의 시퀀스인 $x_p \in R^{N \times P \times P \times C}$로 변경된다. 이때, 2차원 입력 이미지가 분할된 각 패치의 크기는 P×P이다. N은 패치들의 개수이고, C는 색상 정보의 차원을 나타낸다.

[0138] 한편, 이미지 패치들을 사용하는 대신, 입력 시퀀스로서 CNN의 특성맵(feature map)들이 사용될 수 있다.

[0139] 이하의 실시예들은 영상 부호화 장치를 중심으로 기술되나, 영상 복호화 장치에도 동일하게 적용될 수 있다.

[0140] **III. 트랜스포머 기반 인루프 필터**

[0141] 도 15는 본 개시의 일 실시예에 따른, 현재 비디오 블록에 대해 어텐션 과정의 적용을 나타내는 예시도이다.

[0142] 도 15의 예시에서, W×H 크기의 비디오 프레임은 비디오 블록들로 분할되어 있다. 현재 비디오 블록(이하, '현재블록')은 수학식 1과 수학식 2에 나타낸 어텐션 함수를 이용하여 자신 및 프레임 내 다른 블록들과의 어텐션 스코어를 출력한다. 수학식 1과 수학식 2에서 어텐션 함수는 내적을 이용하나, 코사인 유사도, 상관도, MSE(Mean Squared Error), SAD(Sum of Absolute Difference) 등과 같은 두 블록들 간의 유사도 또한 이용될 수 있다.

[0143] 어텐션 함수에 입력으로 들어가는 입력 블록들은 N×M의 그리드(grid)에 위치하여 상호 중첩되지 않는다. 또는 그리드에서 벗어난 채로 입력 블록들이 중첩될 수 있다. 도 15에 예시된 바와 같이, 현재블록을 큐어리(Query)로 설정하여, 현재블록 및 주위 블록들 간의 어텐션 스코어를 계산함으로써, 최종적으로 현재블록의 어텐션 값 벡터(attention value vector)가 출력될 수 있다. 이때, 수학식 1과 수학식 2를 이용하기 위해, 2차원의 현재블록을 1차원 벡터로 변경하는 단계가 필요할 수 있다. 도 15에 예시된 어텐션 스코어들 중에서, 상단으로부터 두 번째 벡터는 현재블록 자신과의 셀프 어텐션 스코어를 나타내고, 나머지는 주위 블록들과의 어텐션 스코어들을 나타낸다.

[0144] 이때, 어텐션 함수를 연산하는 어텐션 모듈에 적용되는 비디오 블록은 다음과 같이 정의될 수 있다.

[0145] 비디오 블록은 CTU, CU, PU(Prediction Unit) 등 부호화의 기본 단위로 사용될 수 있다. 비디오 블록의 너비 및 높이는 동일하다. 또는, 비디오 블록의 너비 및 높이는 상이하게 설정될 수 있다.

[0146] 비디오 블록은 서브블록들로 분할된 채로 사용될 수 있다. 서브블록들은 4×4의 크기와 같이 너비 및 높이가 고정된 채로 사용될 수 있다. 또는, 스케일 인자(scale factor) S를 이용하여, A/S로 크기가 설정될 수 있다. 여기서, A는 분할전 비디오 블록의 너비 또는 높이를 나타낸다.

[0147] 도 16의 예시와 같이, 다수의 비디오 블록들을 통합하여 하나의 영역이 생성될 수 있다. 이때, 해당 영역(도 16의 예시에서 window)은 임의의 크기를 갖는 블록들로 분할될 수 있다.

[0148] 수학식 1에 나타낸 어텐션 연산에 이용되는 두 비디오 블록들은 동일한 크기를 갖는다. 어텐션 계산에 이용하는 두 비디오 블록들의 크기가 다른 경우, 두 비디오 블록들의 크기가 동일하여지도록 패딩이 이용된다.

[0149] 기존의 비디오 블록은 다운샘플링된 후, 어텐션 함수의 입력으로 사용될 수 있다. 또는, 기존의 비디오 블록은 업샘플링된 후, 어텐션 함수의 입력으로 사용될 수 있다. 또는, 기존의 비디오 블록은 다운샘플링되고 업샘플링된 후, 어텐션 함수의 입력으로 사용될 수 있다.

[0150] 한편, 다음과 같은 경우에, 두 비디오 블록들 간의 어텐션 스코어가 연산되지 않는다.

[0151] 두 비디오 블록들이 도 16에서 정의한 영역들(즉, windows)에 대해 상이한 윈도우에 위치하는 경우, 어텐션 스코어가 연산되지 않는다.

[0152] 두 비디오 블록들이 상이한 CTU들에 존재하는 경우, 어텐션 스코어가 연산되지 않는다.

[0153] 도 17의 예시와 같이, 두 비디오 블록들이 상이한 VPDU(Virtual Pipeline Data Unit)들에 존재하는 경우, 어텐션 스코어가 연산되지 않는다. 여기서, VPDU는 가상 파이프라인에 의해 처리될 수 있는 데이터 단위이다. VPDU는 한번에 부호화 및 복호화를 수행할 수 있는 최대 단위로서, CTU의 크기 증가에 따른 하드웨어 구현의 비용을 감소시키기 위해 활용될 수 있다. 또한, VPDU는 부호화 및 복호화를 위한 데이터 처리 단위를 지칭하나, 반드시 VPDU라는 용어에 한정되지 않는다.

[0154] 현재블록 이후에 디코딩되는 블록에 대해, 해당 블록과 현재블록 간의 어텐션 스코어가 연산되지 않을 수 있다.

[0155] 영상 부호화 장치는 복원된 프레임의 화질을 개선하기 위해, 비전 트랜스포머를 이용하는 이미지 복원 및 노이즈 제거 기술을 인루프 필터링에 적용할 수 있다.

[0156] 도 18은 본 개시의 일 실시예에 따른, 트랜스포머 기반 인루프 필터를 나타내는 예시도이다.

[0157] 트랜스포머 기반 인루프 필터(이하, '인루프 필터')는 디블로킹 필터(182), SAO 필터(184) 또는 ALF(186)의 입력 또는 출력 중 하나를 입력 영상으로 이용한다. 영상 부호화 장치는 입력 영상에서 입력 영역 $x \in R^{w \times h \times C}$를 선택한다. 여기서, w 및 h는 입력 영역의 너비 및 높이를 나타낸다. C는 색상 정보의 차원을 나타낸다.

[0158] 인루프 필터는 해당 블록을 직접 트랜스포머의 입력으로 이용할 수 있다. 또는, 도 18의 예시와 같이, 인루프 필터는 CNN을 이용하여 비디오 특성(feature) $f_0 \in R^{w \times h \times C}$를 추출한 후, 추출된 특성을 입력으로 사용할 수 있다. 이후, 인루프 필터는 연속된 K 개의 트랜스포머 블록들을 이용하여 특성 f_1 내지 특성 f_K를 생성한다. 이때, 특성 f_1 내지 특성 f_K의 차원은 f_0의 차원과 동일하다.

[0159] 영상 부호화 장치는 특성 f_K를 임의의 영상 복원을 수행하는 신경망에 입력하여, 입력 영역 $x \in R^{w \times h \times C}$에 대응하는 개선된 비디오 영역을 생성한다. 이때, 영상 복원에 사용하는 신경망의 구조로서, 전술한 바와 같은 EDSR, VDSR 등이 이용될 수 있다.

[0160] 도 19a 및 도 19b는 본 개시의 일 실시예에 따른, 트랜스포머 블록을 나타내는 예시도이다.

[0161] 도 19a의 예시와 같이, 트랜스포머 블록은 연속하는 트랜스포머 계층(transformer layer)들을 포함할 수 있다. 이때, 마지막의 콘볼루션 레이어(convolutional layer)는 선택적으로 배치될 수 있다.

[0162] 도 19b의 예시와 같이, 트랜스포머 블록은 스킵 연결을 이용하여 연속하는 트랜스포머 계층들의 잔차 블록(residual block) 형태로 구성될 수 있다. 이때, 마지막의 콘볼루션 레이어(convolutional layer)는 선택적으로 배치될 수 있다. 이러한 방식은 도 8 및 도 9의 예시된 바와 같은 콘볼루션 레이어의 잔차 블록 형태를 트랜스포머 계층들에 응용하는 것이다.

[0163] 한편, 도 19a에 예시된 하나의 트랜스포머 계층은 도 13에 예시된 바와 같은 트랜스포머 구조에 따라 구현될 수 있다. 이때, N=1로 설정되어, 트랜스포머 계층은 하나의 인코더 층과 하나의 디코더 층을 갖는다.

[0164] 트랜스포머 블록의 입력은, 도 18에 예시된 바와 같은 $f_i \in R^{w \times h \times C}$(i=0,1, ..., K-1)이다. 첫 번째 트랜스포머 블

록 앞에 CNN이 존재하지 않는 경우, 첫 번째 트랜스포머 블록의 입력으로 f_0 대신 입력 영역 $x \in R^{w \times h \times C}$가 이용될 수 있다. 트랜스포머 블록은 해당 입력을 서로 겹치거나 겹치지 않는 패치들로 분할하고, 각 분할된 패치를 특성으로 변환한다. 즉, 각 트랜스포머 블록의 입력은, 도 16에 예시된 window에 해당하고, 패치들은 window가 분할된 비디오 블록들일 수 있다. 이때 각 패치는 도 15의 예시와 같이, P×P 크기를 가질 수 있다. 또는, 패치들은 상이한 임의의 크기를 가질 수 있다.

[0165] 트랜스포머 블록은 도 10a, 도 10b 및 수학식 1에 따른 셀프 어텐션 연산을 이용하여 각 패치의 특성을 연산할 수 있다.

[0166] 이때, 각 패치가 CTU 경계를 벗어나거나 VPDU 경계를 벗어나는 경우, 해당 패치의 셀프 어텐션이 연산되지 않는다. 분할된 패치의 일부가 경계를 벗어난 경우, 경계를 벗어난 일부 영역을 표시하는 마스크(mask)를 이용하여, 해당 영역에 대해 셀프 어텐션이 연산되지 않을 수 있다.

[0167] 또는, 각 패치가 현재블록의 복호화 순서보다 후순위에 존재하는 경우, 해당 패치의 셀프 어텐션이 연산되지 않는다. 분할된 패치의 일부가 후순위에 존재하는 경우, 후순위에 존재하는 일부 영역을 표시하는 마스크를 이용하여, 해당 영역에 대해 셀프 어텐션이 연산되지 않을 수 있다.

[0168] 한편, 트랜스포머 기반 인루프 필터는 고정 계수 인루프 필터일 수 있다. 따라서, 인루프 필터에 포함된 CNN 및 각 트랜스포머 블록을 구성하는 파라미터들은 사전에 트레이닝된 후, 영상 부호화 장치 및 영상 복호화 장치에 동일하게 저장된 채로 사용될 수 있다.

[0169] 이하, 영상 부호화 장치 또는 영상 복호화 장치가 트랜스포머 기반 인루프 필터를 이용하여 복원 프레임의 화질을 개선하는 방법을 기술한다. 이하 컴퓨팅 장치는 영상 부호화 장치 또는 영상 복호화 장치를 나타낸다.

[0170] 도 20은 본 개시의 일 실시예에 따른, 화질을 개선하는 방법을 나타내는 순서도이다.

[0171] 컴퓨팅 장치는 복원 프레임으로부터 기설정된 크기의 입력 영역을 획득한다(S2000). 여기서, 복원 프레임은 원본(original) 프레임을 복원한 프레임으로서, 컴퓨팅 장치에 의해 사전에 복원된다.

[0172] 컴퓨팅 장치는 입력 영역을 딥러닝 기반 인루프 필터에 입력하여 원본 프레임을 근사하는 개선 비디오 영역을 생성한다(S2002).

[0173] 여기서, 인루프 필터는 K(여기서, K는 자연수) 개의 연속된 트랜스포머 블록들 및 제2 CNN을 포함한다.

[0174] 컴퓨팅 장치는 개선 비디오 영역을 생성하는 단계(S2002 단계)를 다음과 같이 수행할 수 있다.

[0175] 트랜스포머 블록들을 이용하여, 어텐션 연산을 기반으로 입력 영상을 최종 출력 특성으로 변환한다(S2010).

[0176] 컴퓨팅 장치는 각 트랜스포머 블록의 입력 특성을 패치들로 분할하고, 각 패치에 어텐션 연산을 적용하여, 각 트랜스포머 블록의 입력 특성에 대해 출력 특성을 생성한다. 컴퓨팅 장치는 각 패치를 큐어리로 설정하고, 어텐션 연산에 이용되는 두 패치들 간의 유사도에 기초하여, 각 패치에 대한 셀프 어텐션 스코어, 및 각 패치와 다른 패치들 간의 어텐션 스코어들을 산정한다. 이후, 컴퓨팅 장치는 어텐션 스코어들을 가중합하여 각 패치의 어텐션 값을 생성할 수 있다.

[0177] 컴퓨팅 장치는 최종 출력 특성을 제2 CNN에 입력하여 개선 비디오 영역을 생성한다(S2012).

[0178] 도 21은 본 개시의 다른 실시예에 따른, 화질을 개선하는 방법을 나타내는 순서도이다.

[0179] 컴퓨팅 장치는 복원 프레임으로부터 기설정된 크기의 입력 영역을 획득한다(S2100). 여기서, 복원 프레임은 원본 프레임을 복원한 프레임으로서, 컴퓨팅 장치에 의해 사전에 복원된다.

[0180] 컴퓨팅 장치는 입력 영역을 딥러닝 기반 인루프 필터에 입력하여 원본 프레임을 근사하는 개선 비디오 영역을 생성한다(S2102).

[0181] 여기서, 인루프 필터는 제1 CNN, K(여기서, K는 자연수) 개의 연속된 트랜스포머 블록들 및 제2 CNN을 포함한다.

[0182] 컴퓨팅 장치는 개선 비디오 영역을 생성하는 단계, 즉 S2102 단계를 다음과 같이 수행할 수 있다.

[0183] 컴퓨팅 장치는 입력 영역을 제1 CNN에 입력하여 입력 특성을 생성한다(S2110).

[0184] 컴퓨팅 장치는 트랜스포머 블록들을 이용하여, 어텐션 연산을 기반으로 입력 특성을 최종 출력 특성으로 변환한

다(S2112).

[0185] 컴퓨팅 장치는 최종 출력 특성을 제2 CNN에 입력하여 개선 비디오 영역을 생성한다(S2114).

[0186] 본 명세서의 흐름도/타이밍도에서는 각 과정들을 순차적으로 실행하는 것으로 기재하고 있으나, 이는 본 개시의 일 실시예의 기술 사상을 예시적으로 설명한 것에 불과한 것이다. 다시 말해, 본 개시의 일 실시예가 속하는 기술 분야에서 통상의 지식을 가진 자라면 본 개시의 일 실시예의 본질적인 특성에서 벗어나지 않는 범위에서 흐름도/타이밍도에 기재된 순서를 변경하여 실행하거나 각 과정들 중 하나 이상의 과정을 병렬적으로 실행하는 것으로 다양하게 수정 및 변형하여 적용 가능할 것이므로, 흐름도/타이밍도는 시계열적인 순서로 한정되는 것은 아니다.

[0187] 이상의 설명에서 예시적인 실시예들은 많은 다른 방식으로 구현될 수 있다는 것을 이해해야 한다. 하나 이상의 예시들에서 설명된 기능들 혹은 방법들은 하드웨어, 소프트웨어, 펌웨어 또는 이들의 임의의 조합으로 구현될 수 있다. 본 명세서에서 설명된 기능적 컴포넌트들은 그들의 구현 독립성을 특히 더 강조하기 위해 "...부(unit)" 로 라벨링되었음을 이해해야 한다.

[0188] 한편, 본 실시예에서 설명된 다양한 기능들 혹은 방법들은 하나 이상의 프로세서에 의해 판독되고 실행될 수 있는 비일시적 기록매체에 저장된 명령어들로 구현될 수도 있다. 비일시적 기록매체는, 예를 들어, 컴퓨터 시스템에 의하여 판독가능한 형태로 데이터가 저장되는 모든 종류의 기록장치를 포함한다. 예를 들어, 비일시적 기록매체는 EPROM(erasable programmable read only memory), 플래시 드라이브, 광학 드라이브, 자기 하드 드라이브, 솔리드 스테이트 드라이브(SSD)와 같은 저장매체를 포함한다.

[0189] 이상의 설명은 본 실시예의 기술 사상을 예시적으로 설명한 것에 불과한 것으로서, 본 실시예가 속하는 기술 분야에서 통상의 지식을 가진 자라면 본 실시예의 본질적인 특성에서 벗어나지 않는 범위에서 다양한 수정 및 변형이 가능할 것이다. 따라서, 본 실시예들은 본 실시예의 기술 사상을 한정하기 위한 것이 아니라 설명하기 위한 것이고, 이러한 실시예에 의하여 본 실시예의 기술 사상의 범위가 한정되는 것은 아니다. 본 실시예의 보호범위는 아래의 청구범위에 의하여 해석되어야 하며, 그와 동등한 범위 내에 있는 모든 기술 사상은 본 실시예의 권리범위에 포함되는 것으로 해석되어야 할 것이다.

부호의 설명

[0190] 180: 루프 필터부

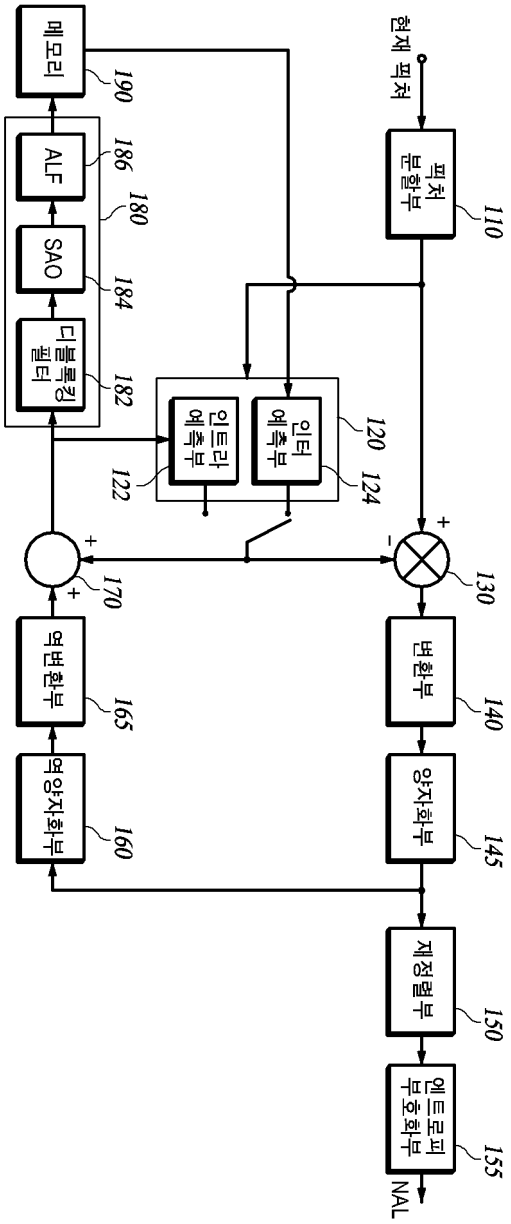
155: 엔트로피 부호화부

510: 엔트로피 복호화부

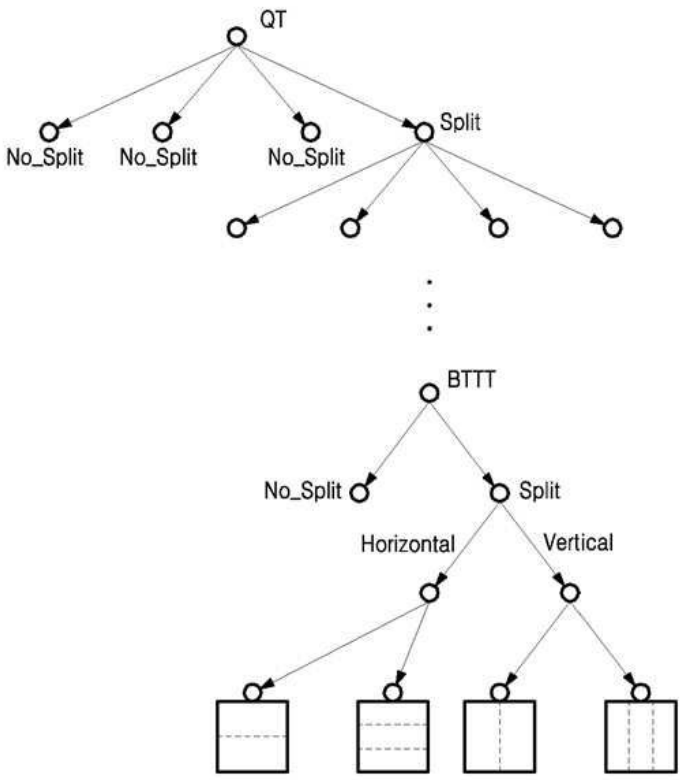
560: 루프 필터부

도면

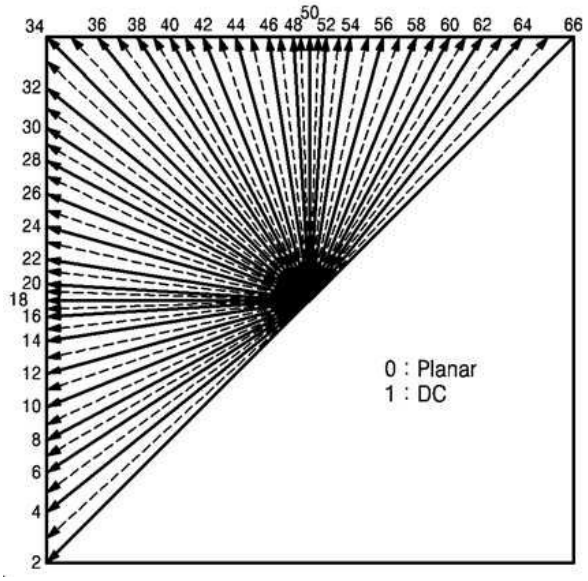
도면1


현재 픽처
픽처 분할부
110
인터 예측부
120
124
인트라 예측부
122
130
변환부
140
양자화부
145
재정렬부
150
엔트로피 부호화부
155
NAL
역양자화부
160
역변환부
165
170
디블록킹 필터
182
SAO
184
ALF
186
180
메모리
190

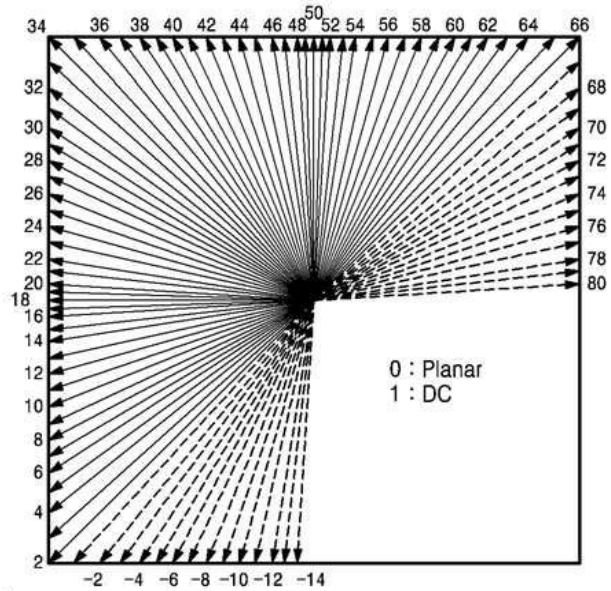
도면2


QT
No_Split
No_Split
No_Split
Split
BTTT
No_Split
Split
Horizontal
Vertical

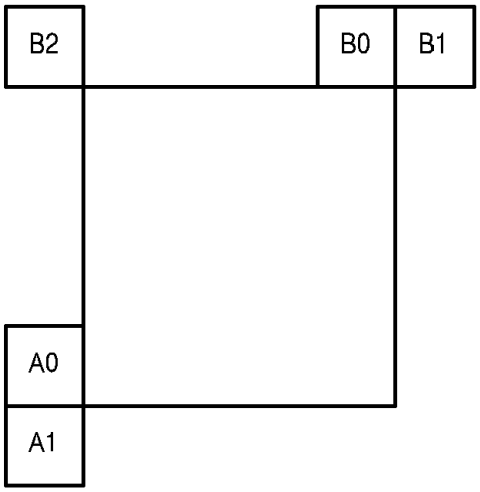
도면3a


34 36 38 40 42 44 46 48 50 52 54 56 58 60 62 64 66
32 30 28 26 24 22 20 18 16 14 12 10 8 6 4 2
0 : Planar
1 : DC

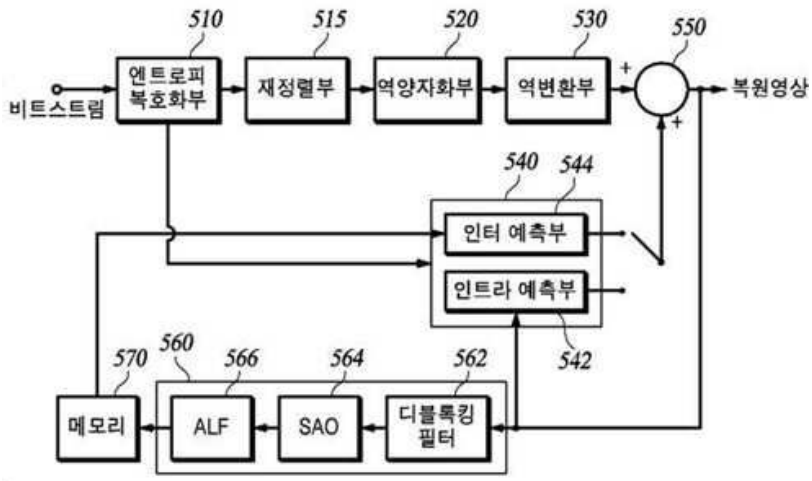
도면3b


34 36 38 40 42 44 46 48 50 52 54 56 58 60 62 64 66
32 30 28 26 24 22 20 18 16 14 12 10 8 6 4 2
68 70 72 74 76 78 80
-2 -4 -6 -8 -10 -12 -14
0 : Planar
1 : DC

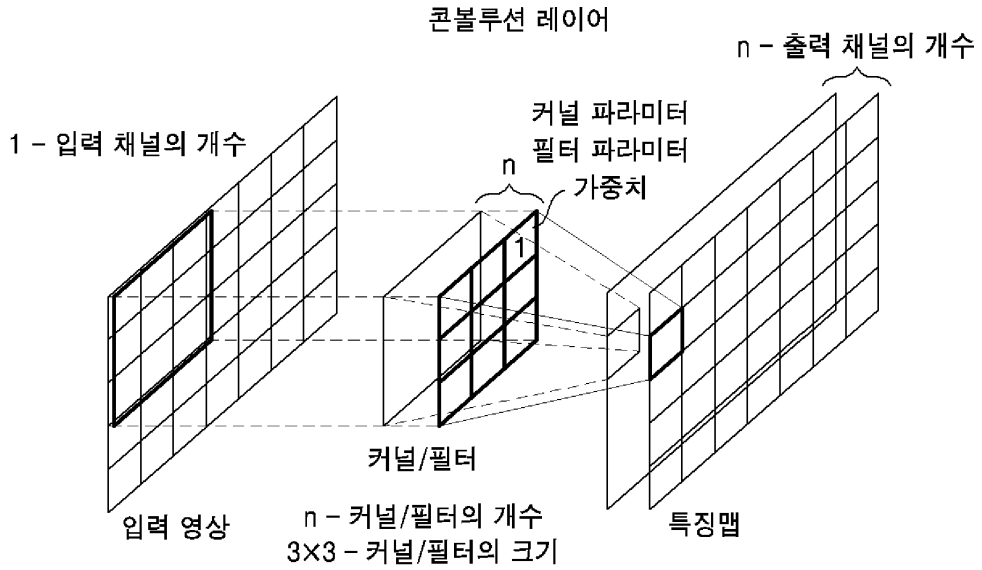
도면4


B2
B0
B1
A0
A1

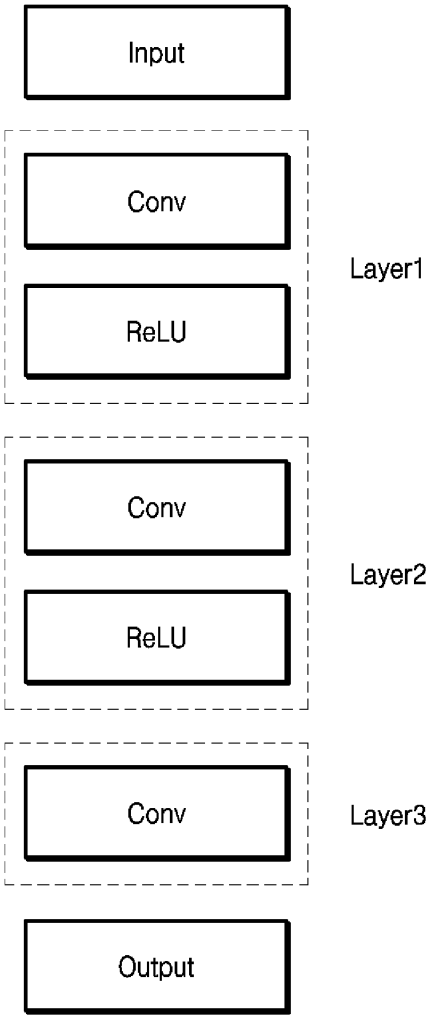
도면5


510
515
520
530
550
비트스트림
엔트로피 복호화부
재정렬부
역양자화부
역변환부
복원영상
540
544
인터 예측부
인트라 예측부
542
570
560
566
564
562
메모리
ALF
SAO
디블록킹 필터

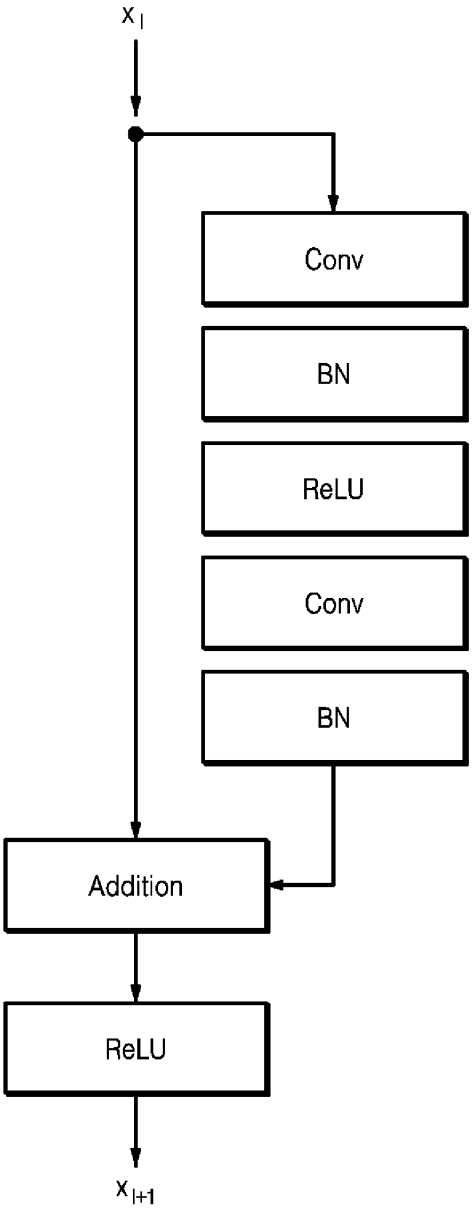
도면6


콘볼루션 레이어
n - 출력 채널의 개수
1 - 입력 채널의 개수
커널 파라미터
필터 파라미터
가중치
n
1
커널/필터
입력 영상
n - 커널/필터의 개수
3×3 - 커널/필터의 크기
특징맵

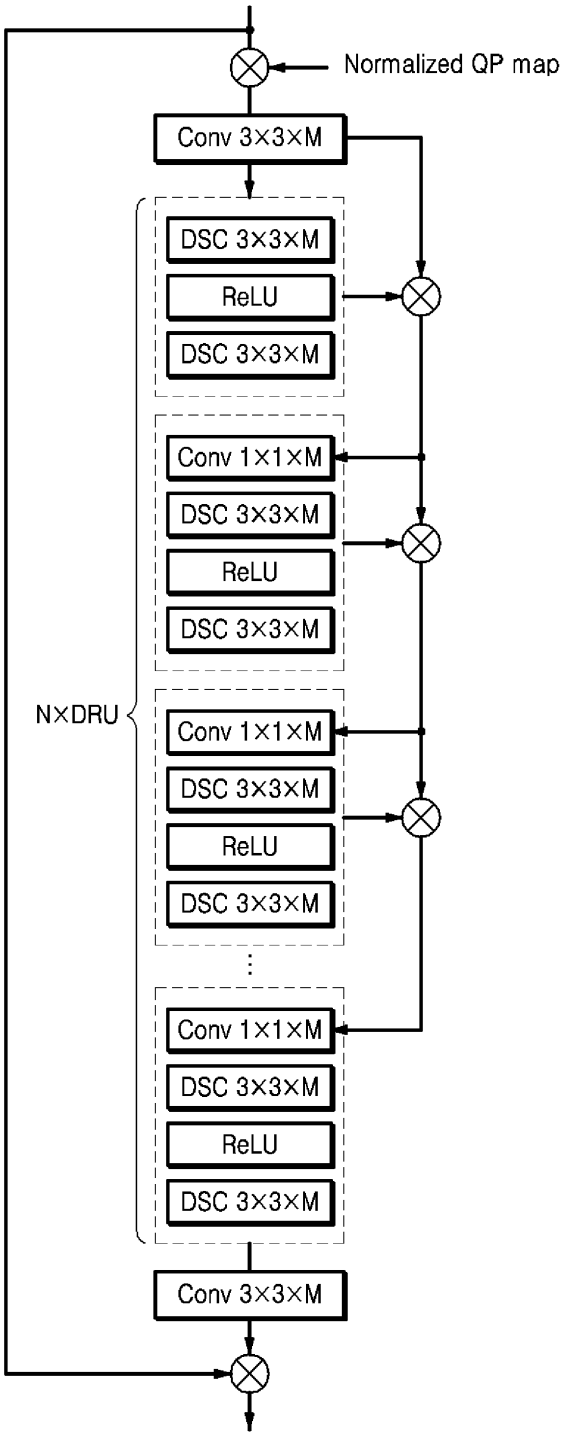
도면7


Input
Conv
ReLU
Layer1
Conv
ReLU
Layer2
Conv
Layer3
Output

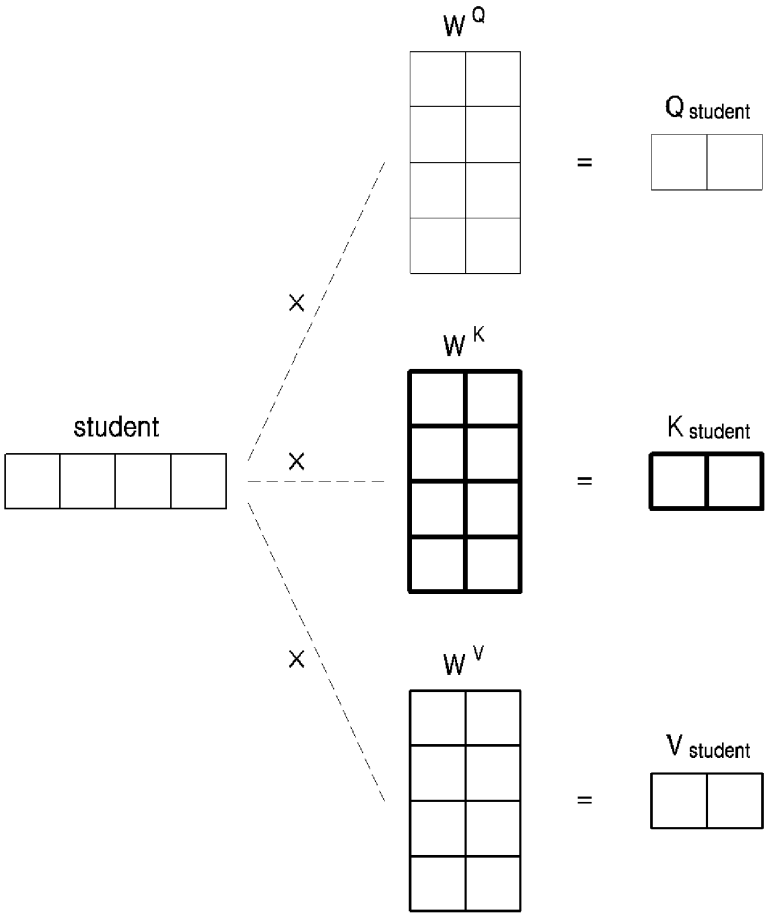
도면8



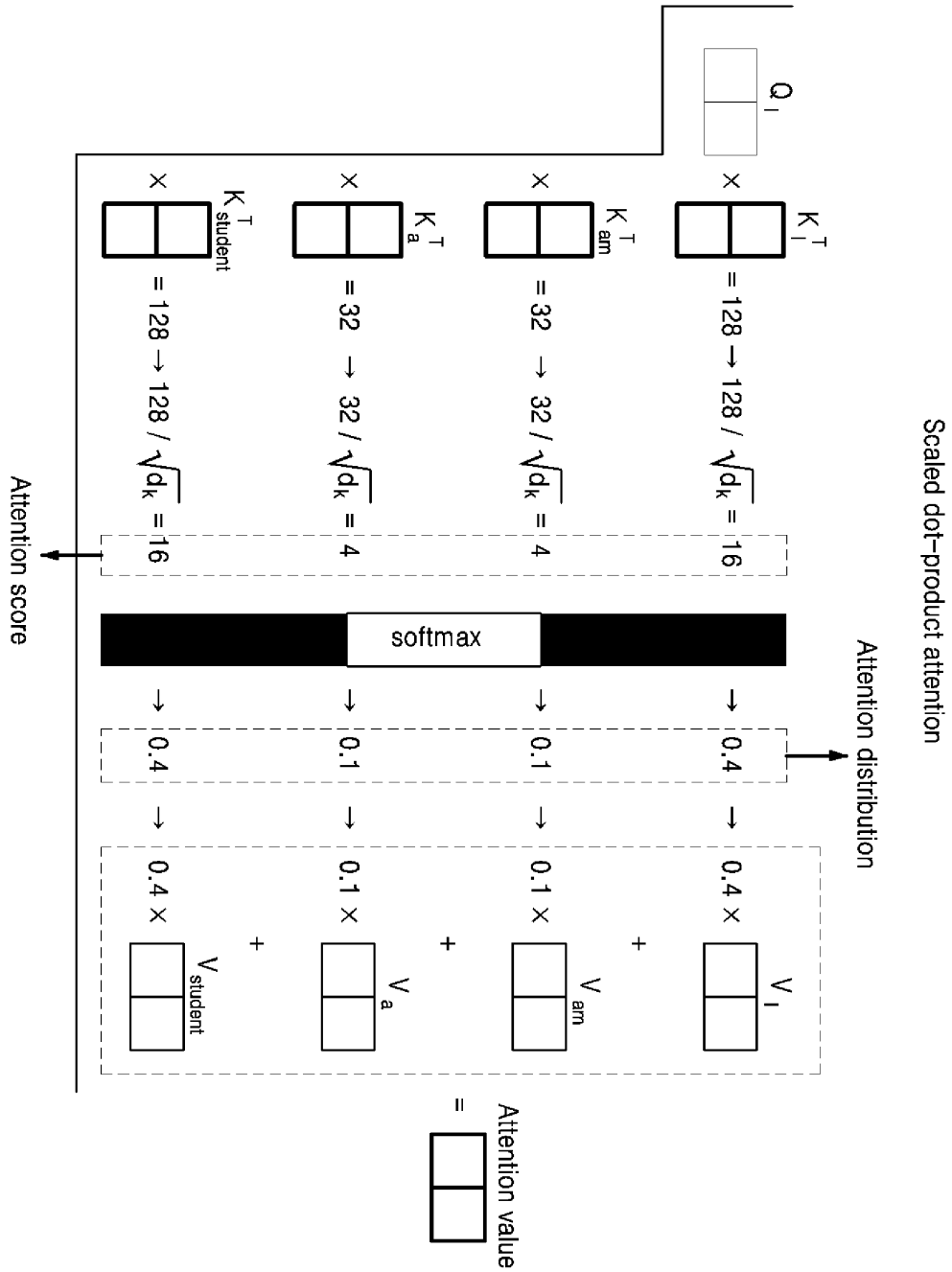
도면9


Normalized QP map
Conv 3×3×M
DSC 3×3×M
ReLU
DSC 3×3×M
Conv 1×1×M
DSC 3×3×M
ReLU
DSC 3×3×M
N×DRU
Conv 1×1×M
DSC 3×3×M
ReLU
DSC 3×3×M
Conv 1×1×M
DSC 3×3×M
ReLU
DSC 3×3×M
Conv 3×3×M

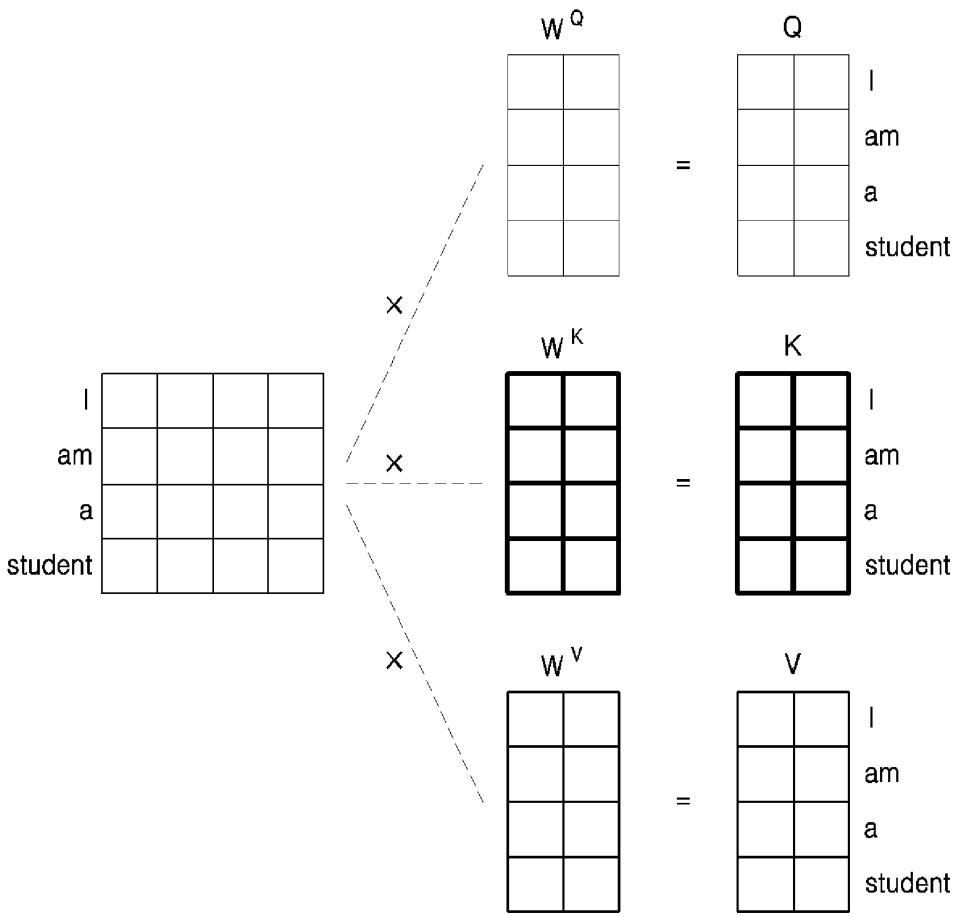
도면10a



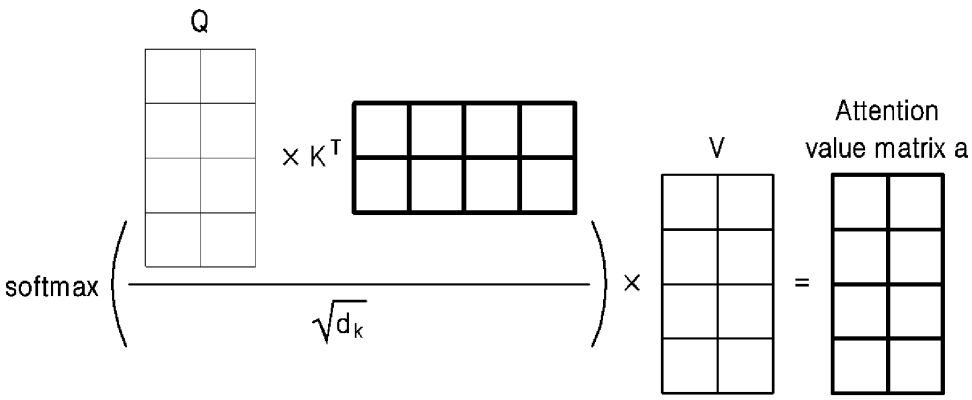
도면10b


Scaled dot-product attention
Attention distribution
Attention value
Attention score
Q_I
K_I^T
K_am^T
K_a^T
K_student^T
×
= 128 → 128 / √d_k = 16
= 32 → 32 / √d_k = 4
= 32 → 32 / √d_k = 4
= 128 → 128 / √d_k = 16
softmax
0.4
0.1
0.1
0.4
0.4 ×
0.1 ×
0.1 ×
0.4 ×
V_I
V_am
V_a
V_student
+
+
+
=

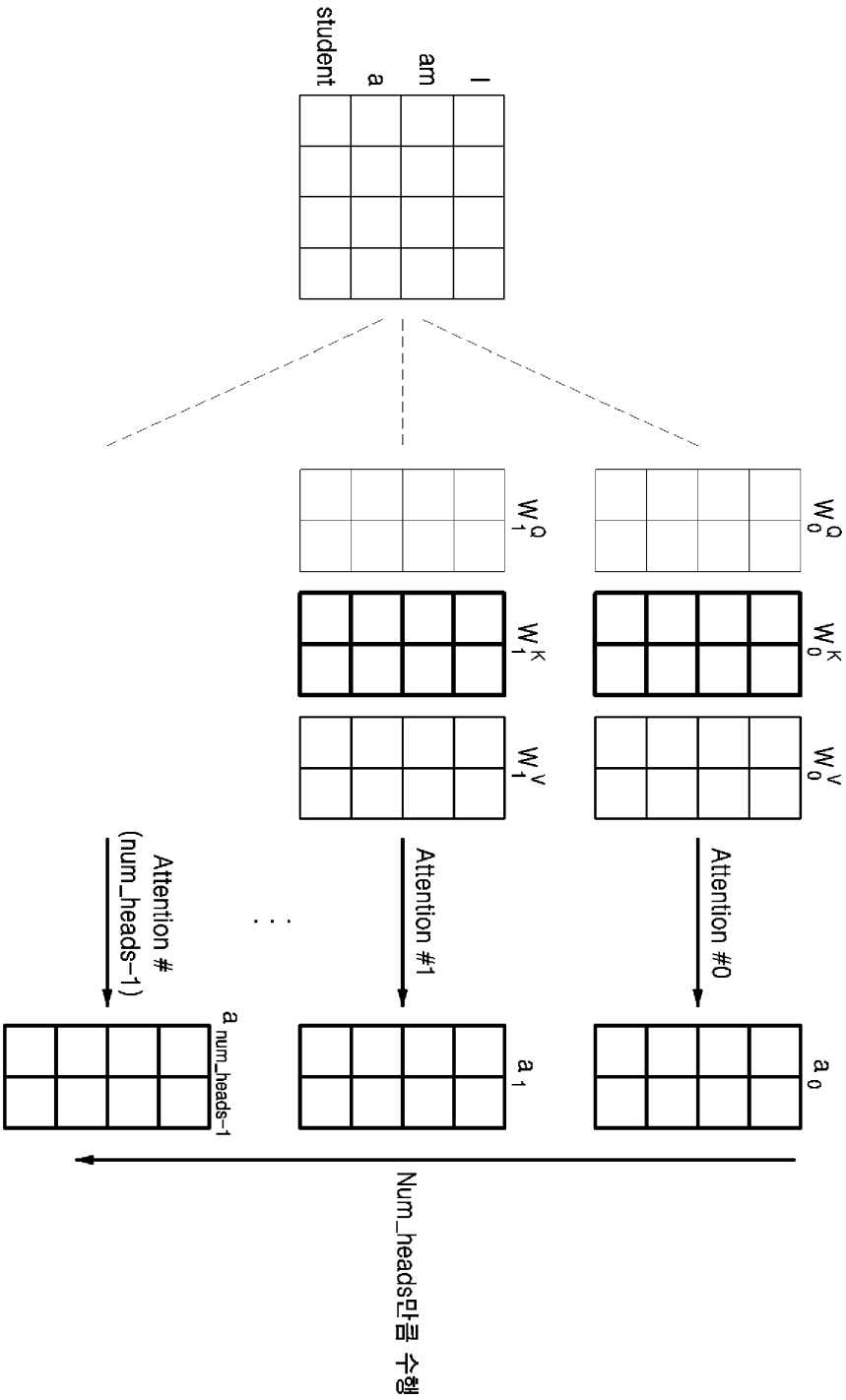
도면11a


W Q
Q
I
am
a
student
=
×
W K
K
I
am
a
student
=
×
W V
V
I
am
a
student
=
×
I
am
a
student

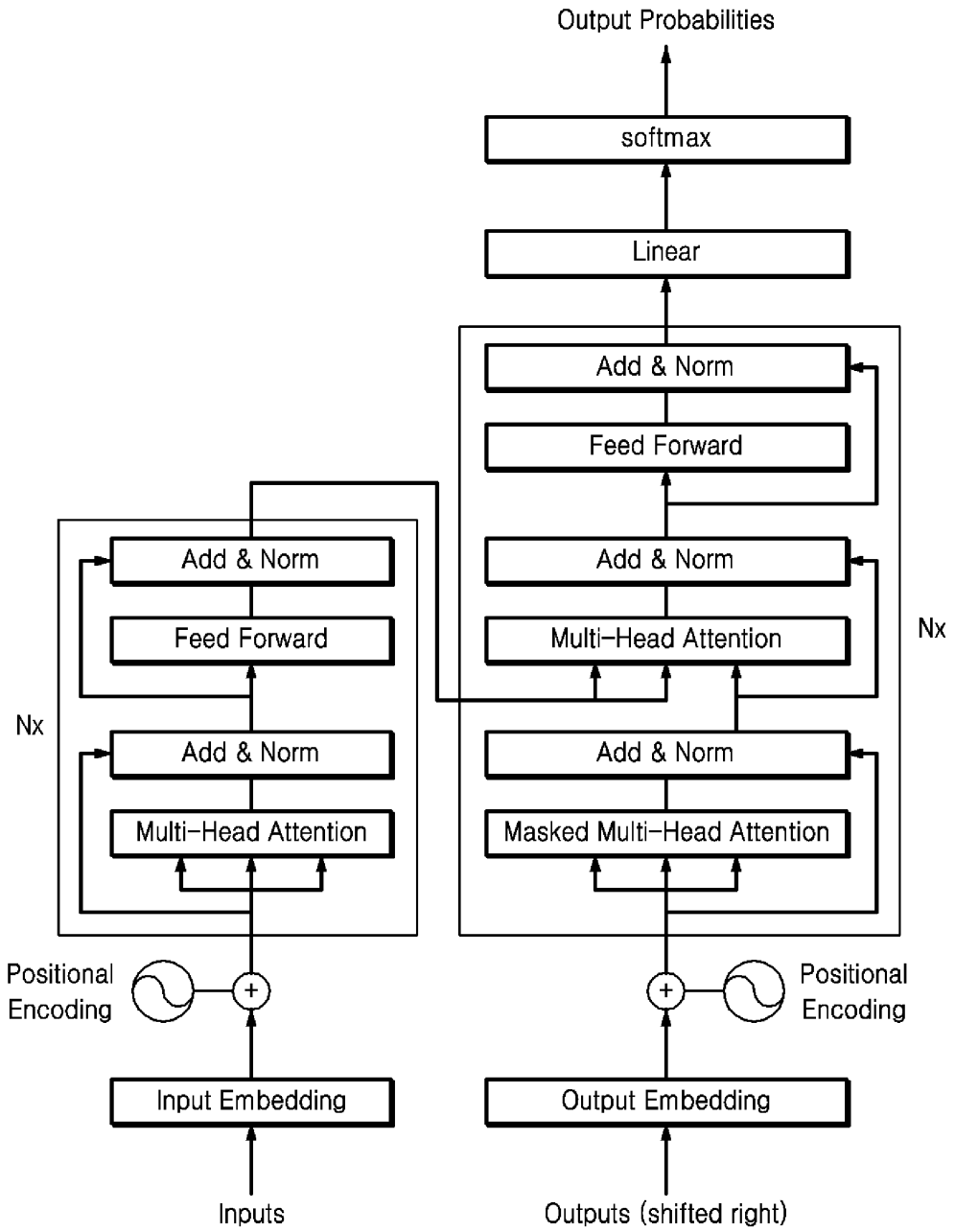
도면11b


Q
× K T
softmax
√d k
×
V
=
Attention
value matrix a

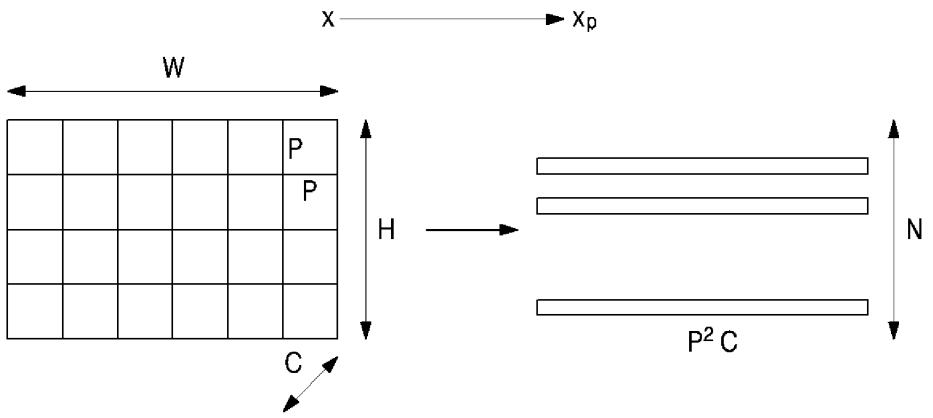
도면12


I
am
a
student
W_0^Q
W_0^K
W_0^V
W_1^Q
W_1^K
W_1^V
Attention #0
Attention #1
Attention #
(num_heads-1)
a_0
a_1
$a_{num_heads-1}$
Num_heads만큼 수행

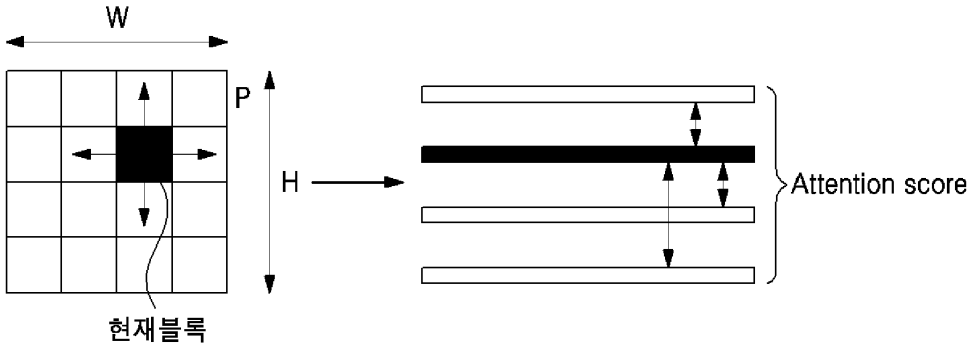
도면13


Output Probabilities
softmax
Linear
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Add & Norm
Masked Multi-Head Attention
Nx
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Nx
Positional Encoding
Positional Encoding
Input Embedding
Output Embedding
Inputs
Outputs (shifted right)

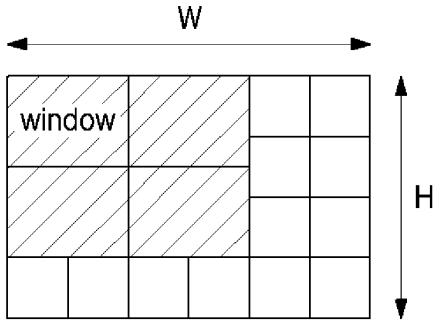
도면14



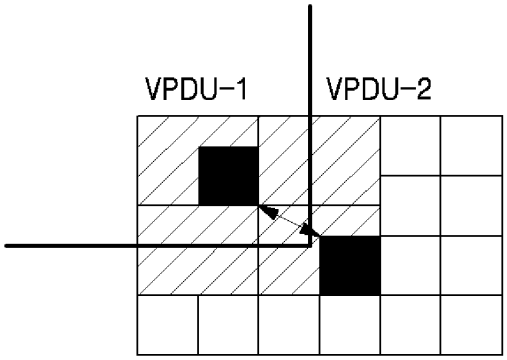
도면15



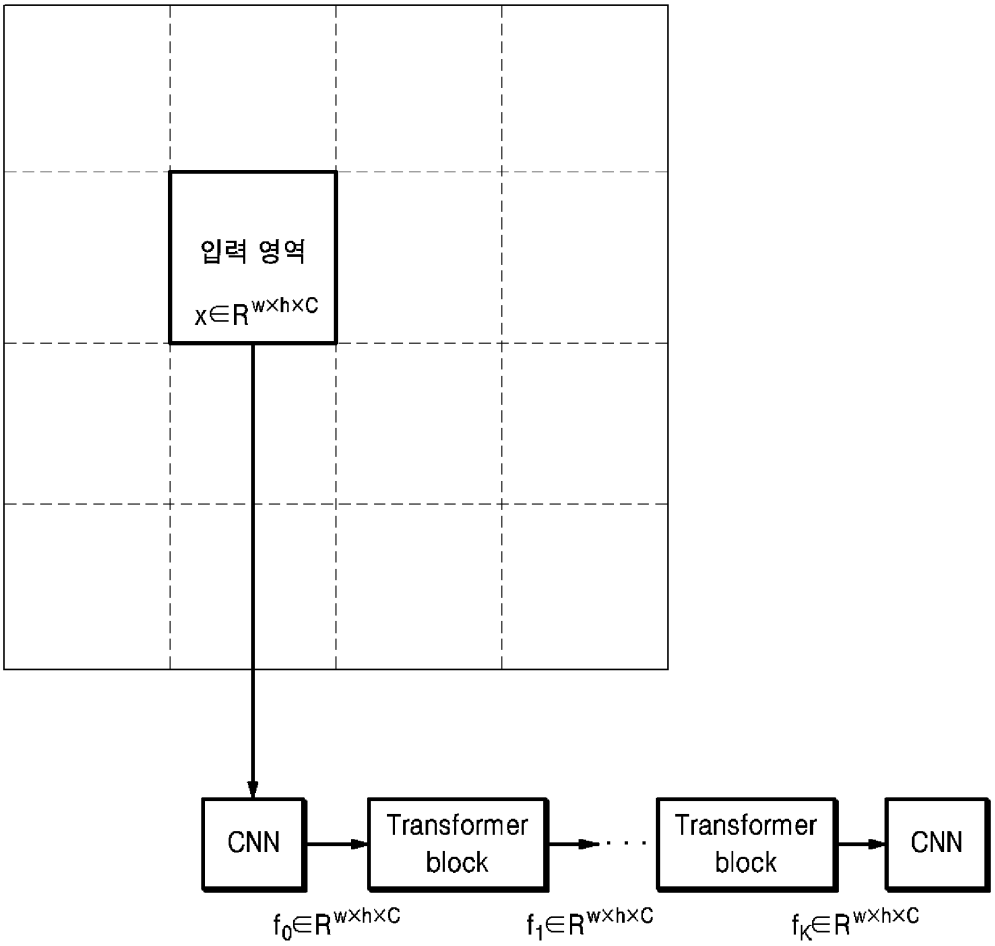
도면16



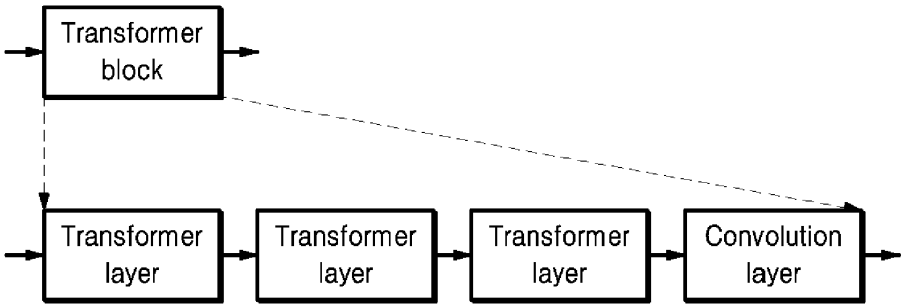
도면17


VPDU-1
VPDU-2

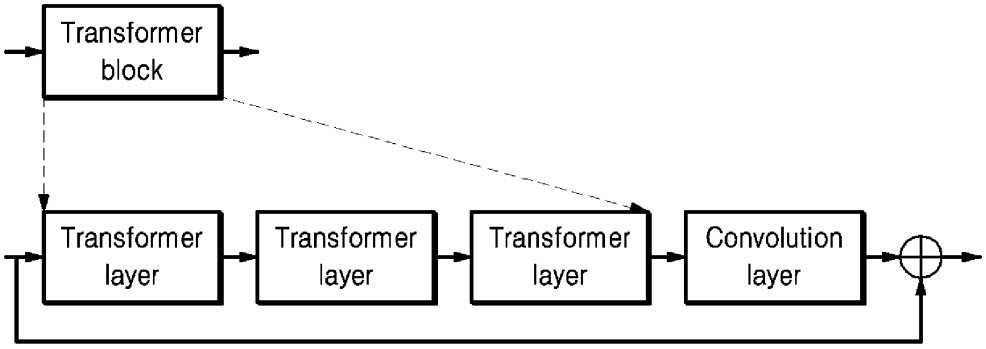
도면18


입력 영역
x∈R^w×h×C
CNN
Transformer block
Transformer block
CNN
f0∈R^w×h×C
f1∈R^w×h×C
fK∈R^w×h×C

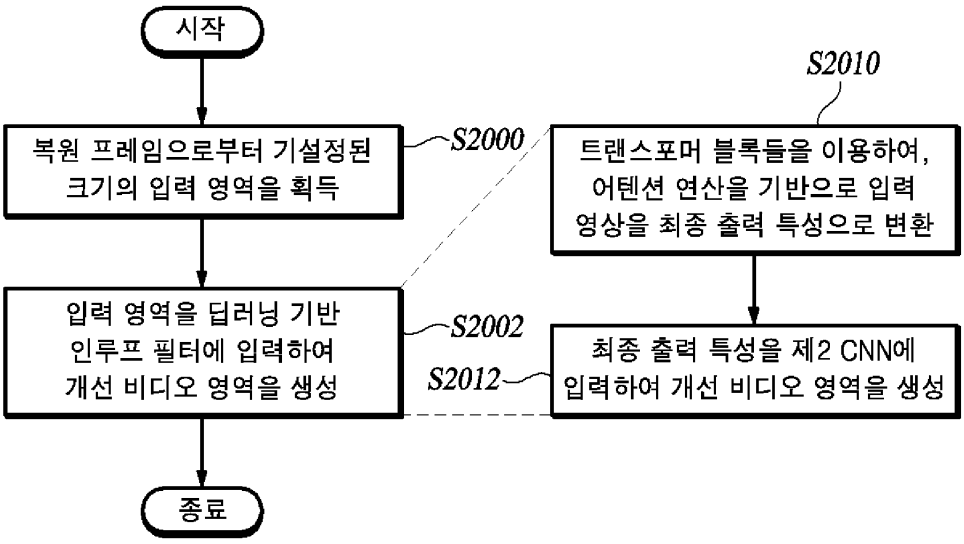
도면19a



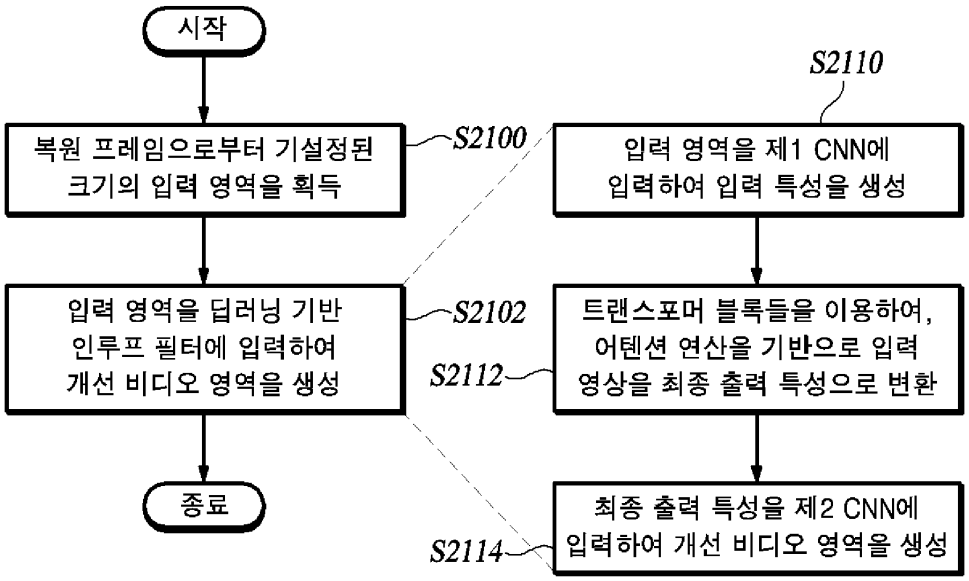
도면19b



도면20



도면21


시작
복원 프레임으로부터 기설정된 크기의 입력 영역을 획득
S2100
입력 영역을 딥러닝 기반 인루프 필터에 입력하여 개선 비디오 영역을 생성
S2102
종료
S2110
입력 영역을 제1 CNN에 입력하여 입력 특성을 생성
트랜스포머 블록들을 이용하여, 어텐션 연산을 기반으로 입력 영상을 최종 출력 특성으로 변환
S2112
최종 출력 특성을 제2 CNN에 입력하여 개선 비디오 영역을 생성
S2114