



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2019-0077428
(43) 공개일자 2019년07월03일

- (51) 국제특허분류(Int. Cl.)
H04N 21/2662 (2011.01) *H04N 19/46* (2014.01)
H04N 19/513 (2014.01) *H04N 19/587* (2014.01)
H04N 19/87 (2014.01) *H04N 21/218* (2011.01)
H04N 21/234 (2014.01) *H04N 21/4402* (2011.01)
H04N 21/61 (2011.01) *H04N 21/84* (2011.01)
- (52) CPC특허분류
H04N 21/2662 (2013.01)
H04N 19/46 (2015.01)
- (21) 출원번호 10-2019-7014759
- (22) 출원일자(국제) 2017년11월08일
 심사청구일자 없음
- (85) 번역문제출일자 2019년05월22일
- (86) 국제출원번호 PCT/IB2017/056989
- (87) 국제공개번호 WO 2018/087675
 국제공개일자 2018년05월17일
- (30) 우선권주장
 15/346,392 2016년11월08일 미국(US)
- (71) 출원인
 에이티아이 테크놀로지스 유엘씨
 캐나다 온타리오 엘3티 7엑스6 마크햄 커머스 밸리 드라이브 이스트 1
- (72) 발명자
 이바노비치 보리스
 캐나다 엘3티 7엑스6 마크햄 원 커머스 밸리 드라이브 이스트
- (74) 대리인
 박장원

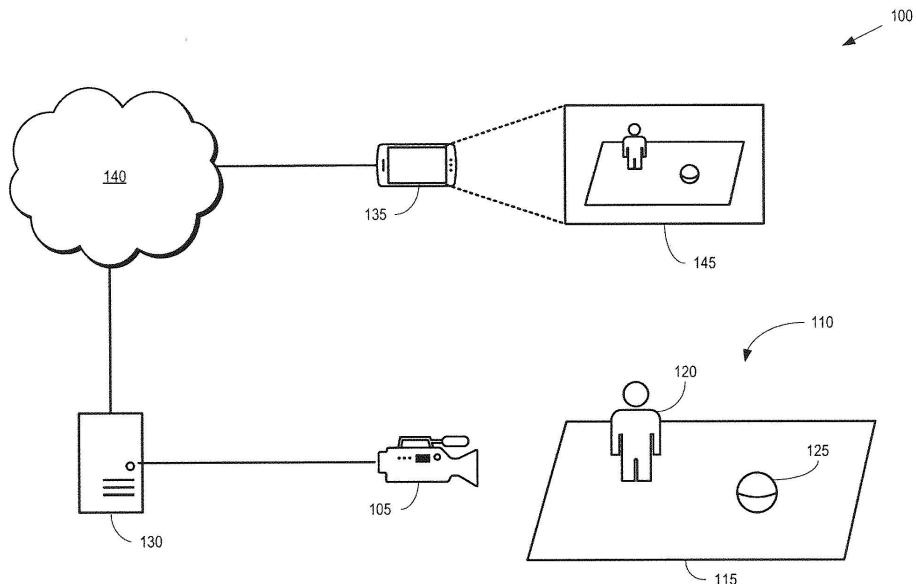
전체 청구항 수 : 총 20 항

(54) 발명의 명칭 **스트리밍된 메타데이터를 사용한 비디오 프레임 변환**

(57) 요 약

비디오 서버(130, 305, 405, 505, 605)는 제 1 프레임을 포함하는 프레임들(315, 335, 415)의 스트림에서 씬(110)을 나타내는 제 1 프레임(200)의 부분들에 대한 보간 파라미터들(240 내지 244)을 나타내는 메타데이터(530, 535)를 생성한다. 보간 파라미터들은 프레임들의 스트림에서 제 1 프레임 다음에 그리고 제 2 프레임에 앞 (뒷면에 계속)

대 표 도



서 씬을 나타내는 적어도 하나의 보간된 프레임(205)을 생성하기 위해 사용된다. 비디오 서버는 스트림으로 메타데이터를 통합하며 다중화된 메타데이터를 포함하는 스트림을 송신한다. 비디오 클라이언트(135, 310, 410, 510, 610)는 메타데이터를 포함하는 프레임들의 스트림을 나타내는 제 1 프레임을 수신한다. 비디오 클라이언트는 제 1 프레임 및 메타데이터에 기초하여 프레임들의 스트림에서 제 1 프레임 다음에 그리고 제 2 프레임에 앞서 씬을 나타내는 하나 이상의 보간된 프레임들(205)을 생성한다. 비디오 클라이언트는 제 1 프레임, 하나 이상의 보간된 프레임들, 및 제 2 프레임을 디스플레이한다.

(52) CPC특허분류

HO4N 19/521 (2015.01)
HO4N 19/587 (2015.01)
HO4N 19/87 (2015.01)
HO4N 21/218 (2013.01)
HO4N 21/23418 (2013.01)
HO4N 21/440281 (2013.01)
HO4N 21/6125 (2013.01)
HO4N 21/84 (2013.01)

명세서

청구범위

청구항 1

방법에 있어서,

비디오 서버(130, 305, 405, 505, 605)에서, 제 1 프레임을 포함하는 프레임들의 스트림(315, 335, 415)에서 씬(scene)(110)을 나타내는 상기 제 1 프레임(200)의 부분들에 대한 보간 파라미터들(240 내지 244)을 나타내는 메타데이터(525, 530)를 생성하는 단계로서, 상기 보간 파라미터들은 상기 프레임들의 스트림에서 상기 제 1 프레임 다음에 그리고 제 2 프레임에 앞서 상기 씬을 나타내는 적어도 하나의 보간된 프레임(205)을 생성하기 위해 사용되는, 상기 메타데이터 생성 단계;

상기 비디오 서버에서, 상기 메타데이터를 상기 스트림으로 통합하는 단계; 및

상기 비디오 서버로부터, 상기 메타데이터를 포함한 상기 스트림을 송신하는 단계를 포함하는, 방법.

청구항 2

청구항 1에 있어서,

상기 메타데이터를 생성하는 단계는 상기 제 1 프레임의 부분들과 연관된 모션 벡터들(240 내지 244) 및 상기 모션 벡터들의 신뢰 척도(confidence measure)들을 나타내는 메타데이터를 생성하는 단계를 포함하는, 방법.

청구항 3

청구항 1에 있어서,

상기 메타데이터를 생성하는 단계는 상기 제 1 프레임의 이웃하는 부분들에 대한 이웃하는 보간 파라미터들과 통계적으로 상이한 상기 제 1 프레임의 일 부분에 대한 이상치(outlier) 보간 파라미터를 식별하는 단계를 포함하며, 상기 이상치 보간 파라미터가 보간에 대해 무시됨을 나타내거나 또는 상기 이웃하는 모션 벡터들에 기초하여 상기 이상치 보간 파라미터를 수정한 메타데이터를 생성하는, 방법.

청구항 4

청구항 1에 있어서,

상기 제 1 프레임의 오클루전(occlusion)된 부분에 대한 보간 파라미터를 식별하기 위해 오클루전 검출을 수행하는 단계를 더 포함하는, 방법.

청구항 5

청구항 4에 있어서,

오클루전 검출을 수행하는 단계는 상기 비디오 서버에서 오클루전 검출을 수행하는 단계를 포함하며, 상기 메타데이터를 생성하는 단계는 상기 제 1 프레임의 오클루전된 부분에 대한 상기 보간 파라미터가 보간에 대해 무시됨을 나타내는 메타데이터를 생성하는 단계를 포함하는, 방법.

청구항 6

방법에 있어서,

비디오 클라이언트(135, 310, 410, 510, 610)에서, 제 1 프레임의 부분들에 대한 보간 파라미터들(240 내지 244)을 나타내는 메타데이터(525, 530)를 포함하는 프레임들의 스트림(315, 335, 415)에서 씬(110)을 나타내는 상기 제 1 프레임(200)을 수신하는 단계;

상기 비디오 클라이언트에서, 상기 제 1 프레임 및 상기 메타데이터에 기초하여 상기 프레임들의 스트림에서 상기 제 1 프레임의 다음에 그리고 제 2 프레임에 앞서 상기 씬을 나타내는 적어도 하나의 보간된 프레임(205)을 생성하는 단계; 및

상기 비디오 클라이언트에서, 상기 제 1 프레임, 상기 적어도 하나의 보간된 프레임, 및 상기 제 2 프레임을 디스플레이하는 단계를 포함하는, 방법.

청구항 7

청구항 6에 있어서,

상기 보간 파라미터들을 나타내는 상기 메타데이터를 수신하는 단계는 상기 프레임의 부분들과 연관된 모션 벡터들(240 내지 244) 및 상기 모션 벡터들의 신뢰 척도들을 나타내는 메타데이터를 수신하는 단계를 포함하는, 방법.

청구항 8

청구항 6에 있어서,

상기 메타데이터를 수신하는 단계는 상기 프레임의 이웃하는 부분들에 대한 이웃하는 보간 파라미터들과 통계적으로 상이한 상기 프레임의 일 부분에 대한 이상치 보간 파라미터를 식별하는 메타데이터를 수신하는 단계를 포함하며, 상기 적어도 하나의 보간된 프레임을 생성하는 단계는 상기 적어도 하나의 보간된 프레임을 생성할 때 상기 이상치 보간 파라미터를 무시하는 단계를 포함하는, 방법.

청구항 9

청구항 6에 있어서,

상기 프레임의 오클루전된 부분에 대한 보간 파라미터를 식별하기 위해 상기 비디오 클라이언트에서 오클루전 검출을 수행하는 단계; 및

상기 적어도 하나의 보간된 프레임을 생성할 때 상기 프레임의 오클루전된 부분에 대한 상기 보간 파라미터를 무시하는 단계를 더 포함하는, 방법.

청구항 10

청구항 6에 있어서,

상기 메타데이터를 생성한 것을 수신하는 단계는 상기 프레임의 오클루전된 부분에 대한 보간 파라미터를 나타내는 메타데이터를 수신하는 단계를 포함하며, 상기 적어도 하나의 보간된 프레임을 생성하는 단계는 상기 적어도 하나의 보간된 프레임을 생성할 때 상기 프레임의 오클루전된 부분에 대한 상기 보간 파라미터를 무시하는 단계를 포함하는, 방법.

청구항 11

비디오 서버(130, 305, 405, 505, 605)에 있어서,

제 1 프레임을 포함하는 프레임들의 스트림(315, 335, 415)에서 씬(110)을 나타내는 상기 제 1 프레임(200)의 부분들에 대한 보간 파라미터들(240 내지 244)을 나타내는 메타데이터(525, 530)를 생성하기 위한 프로세서(625)로서, 상기 보간 파라미터들은 상기 프레임들의 스트림에서 상기 제 1 프레임의 다음에 그리고 제 2 프레임에 앞서 상기 씬을 나타내는 적어도 하나의 보간된 프레임(205)을 생성하기 위해 사용되며, 상기 프로세서는 또한 상기 메타데이터를 상기 스트림으로 통합하도록 구성되는, 상기 프로세서; 및

다중화된 메타데이터를 포함하는 상기 스트림을 송신하기 위한 네트워크 인터페이스(615)를 포함하는, 비디오 서버.

청구항 12

청구항 11에 있어서,

상기 프로세서는 상기 제 1 프레임의 부분들과 연관된 모션 벡터들(240 내지 244) 및 상기 모션 벡터들의 신뢰 척도들을 나타내는 메타데이터를 생성하도록 구성되는, 비디오 서버.

청구항 13

청구항 11에 있어서,

상기 프로세서는 상기 제 1 프레임의 이웃하는 부분들에 대한 이웃하는 보간 파라미터들과 통계적으로 상이한 상기 제 1 프레임의 일 부분에 대한 이상치 보간 파라미터를 식별하도록 구성되는, 비디오 서버.

청구항 14

청구항 13에 있어서,

상기 프로세서는 상기 이상치 보간 파라미터가 보간에 대해 무시될 것임을 나타내거나 또는 상기 이웃하는 모션 벡터들에 기초하여 상기 이상치 보간 파라미터를 수정한 메타데이터를 생성하도록 구성되는, 비디오 서버.

청구항 15

청구항 11에 있어서,

상기 프로세서는 또한 상기 제 1 프레임의 오클루전된 부분에 대한 보간 파라미터를 식별하기 위해 오클루전 검출을 수행하며, 상기 제 1 프레임의 오클루전된 부분에 대한 상기 보간 파라미터가 보간에 대해 무시됨을 나타내는 메타데이터를 생성하도록 구성되는, 비디오 서버.

청구항 16

비디오 클라이언트(135, 310, 410, 510, 610)에 있어서,

제 1 프레임의 부분들에 대한 보간 파라미터들(240 내지 242)을 나타내는 메타데이터(530, 535)를 포함하는 프레임들의 스트림(315, 335, 415)에서의 씬(110)을 나타내는 상기 제 1 프레임(200)을 수신하기 위한 네트워크 인터페이스(635);

상기 제 1 프레임 및 상기 메타데이터에 기초하여 상기 프레임들의 스트림에서 상기 제 1 프레임 다음에 그리고 제 2 프레임에 앞서 상기 씬을 나타내는 적어도 하나의 보간된 프레임(205)을 생성하기 위한 프로세서(645); 및 상기 제 1 프레임, 상기 적어도 하나의 보간된 프레임, 및 상기 제 2 프레임을 디스플레이하기 위한 스크린(145, 640)을 포함하는, 비디오 클라이언트.

청구항 17

청구항 16에 있어서,

상기 네트워크 인터페이스는 상기 프레임의 부분들과 연관된 모션 벡터들(240 내지 244) 및 상기 모션 벡터들의 신뢰 척도들을 나타내는 메타데이터를 수신하도록 구성되는, 비디오 클라이언트.

청구항 18

청구항 16에 있어서,

상기 네트워크 인터페이스는 상기 프레임의 이웃하는 부분들에 대한 이웃하는 보간 파라미터들과 통계적으로 상이한 상기 프레임의 일 부분에 대한 이상치 보간 파라미터를 식별하는 메타데이터를 수신하도록 구성되며, 상기 프로세서는 상기 적어도 하나의 보간된 프레임을 생성할 때 상기 이상치 보간 파라미터를 무시하도록 구성되는, 비디오 클라이언트.

청구항 19

청구항 16에 있어서,

상기 프로세서는 상기 프레임의 오클루전된 부분에 대한 보간 파라미터를 식별하기 위해 오클루전 검출을 수행하며, 상기 적어도 하나의 보간된 프레임을 생성할 때 상기 프레임의 오클루전된 부분에 대한 상기 보간 파라미터를 무시하도록 구성되는, 비디오 클라이언트.

청구항 20

청구항 16에 있어서,

상기 네트워크 인터페이스는 상기 프레임의 오클루전된 부분에 대한 보간 파라미터를 나타내는 메타데이터를 수

신하도록 구성되며, 상기 프로세서는 상기 적어도 하나의 보간된 프레임을 생성할 때 상기 프레임의 오클루전된 부분에 대한 상기 보간 파라미터를 무시하도록 구성되는, 비디오 클라이언트.

발명의 설명

기술 분야

배경기술

[0001] 프레임 레이트는 프레임들(또한 흔히 이미지를 또는 필드들로서 불리우는)이 카메라들에 의해 캡처되거나 또는 영사기들, 텔레비전들, 디지털 디스플레이들 등과 같은 디바이스들에 의해 디스플레이되는 레이트를 나타낸다. 예를 들면, 종래의 영화 카메라들은 초당 24 프레임들(FPS)의 레이트로 프레임들을 캡처하며 종래의 영사기들은 24 FPS의 동일한 레이트로 프레임들을 투사한다. 몇몇 디지털 이미징 디바이스들은 30 FPS, 48 FPS, 60 FPS 이상과 같은 더 높은 프레임 레이트들로 프레임들을 캡처할 수 있다. 고화질 텔레비전들(HDTV들)과 같은, 디지털 디스플레이들은 60 FPS 이상과 같은 보다 높은 프레임 레이트들로 프레임들을 디스플레이할 수 있다. 이미지 캡처 디바이스의 프레임 레이트들이 종종 디스플레이 디바이스의 프레임 레이트와 상이하므로, 디스플레이 디바이스들은 디스플레이 디바이스의 프레임 레이트에 매칭시키도록 캡처된 프레임들의 프레임 레이트를 수정하기 위해 프레임 레이트 변환을 사용한다. 예를 들면, 24 FPS의 레이트로 캡처된 프레임들은 5개의 디스플레이된 프레임들에 대응하는 지속 기간 동안 두 개의 캡처된 프레임들을 디스플레이함으로써 60 FPS에서 디스플레이될 수 있다. 이것은 일련의 5개의 디스플레이된 프레임들: AAABB을 형성하기 위해, 두 개의 연속적인 캡처된 프레임들(A 및 B)이 각각 3회 및 2회 반복되기 때문에 3:2 변환으로 불리운다. 캡처된 프레임드를 반복함으로써 프레임 레이트 상향-변환을 수행하는 것은 상대적 단순성의 이점을 갖지만, 저더(judder) 및 블러와 같은 원치 않는 시각적 효과들을 도입하는 것으로 알려져 있다.

도면의 간단한 설명

[0002] 본 개시는 수반되는 도면들을 참조함으로써 보다 양호하게 이해될 수 있으며, 그것의 다수의 특징들 및 이점들이 이 기술분야의 숙련자들에게 명백해진다. 상이한 도면들에서 동일한 참조 심볼들의 사용은 유사한 또는 동일한 아이템들을 나타낸다.

도 1은 몇몇 실시예들에 따른 비디오 획득 및 디스플레이 시스템의 다이어그램이다.

도 2는 몇몇 실시예들에 따라 비디오 프레임 및 모션 벡터들에 기초하여 생성된 보간된 프레임을 예시하는 다이어그램이다.

도 3은 몇몇 실시예들에 따라 비디오 서버 및 비디오 클라이언트를 포함하는 비디오 프로세싱 시스템의 제 1 예를 예시한 블록도이다.

도 4는 몇몇 실시예들에 따라 비디오 서버 및 비디오 클라이언트를 포함하는 비디오 프로세싱 시스템의 제 2 예를 예시한 블록도이다.

도 5는 몇몇 실시예들에 따른 비디오 프레임들, 메타데이터, 및 보간된 프레임들을 예시한 비디오 프로세싱 시스템의 블록도이다.

도 6은 몇몇 실시예들에 따라 비디오 프레임들로부터 메타데이터를 생성하기 위한 비디오 서버 및 메타데이터 및 비디오 프레임들에 기초하여 보간된 프레임들을 생성하기 위한 비디오 클라이언트를 포함하는 비디오 프로세싱 시스템의 블록도이다.

도 7은 몇몇 실시예들에 따른 이미지에서 오브젝트들과 연관된 모션 벡터들을 결정하기 위해 탐색될 수 있는 이미지를 디스플레이하는 스크린을 포함한 다이어그램이다.

발명을 실시하기 위한 구체적인 내용

[0003] 60 FPS와 같은, 높은 프레임 레이트들을 지원하는 비디오 디스플레이 디바이스들은 종종, 수신된 프레임들의 부분들의 모션 벡터들에 기초하여, 수신된 프레임들 사이에서 보간함으로써 비디오 서버들로부터 수신된 보다 낮은 프레임 레이트 스트림들에 대한 비디오 레이트 상향 변환을 수행한다. 모션 벡터들을 결정하기 위해, 24 FPS

의 프레임 레이트로 캡처되는 프레임들은 하나 이상의 픽셀들을 포함하는 부분들로 세분된다. 제 1 프레임에서 각각의 부분은 후보 모션 벡터에 의해 표시된 거리만큼 제 1 프레임에서의 부분의 위치로부터 오프셋되는 뒤이은 (제 2) 프레임에서의 대응 부분들에 비교된다. 유사한 비교들이 제 1 프레임의 부분의 가능한 모션들을 나타내는 후보 모션 벡터들의 세트에 대해 수행된다. 제 1 프레임에서의 부분 및 제 2 프레임에서의 오프셋 부분 사이에서의 최상의 매치를 생성하는 모션 벡터가 제 1 프레임에서의 부분의 모션을 나타내는 모션 벡터로서 선택된다. 모션 벡터 산출은 그 후 제 1 프레임에 대한 모션 벡터 필드를 결정하기 위해 제 1 프레임의 모든 부분에 대해 반복된다. 비디오 디스플레이 디바이스는 종래의 프레임 레이트 변환에서 사용된 반복된 프레임들을 대체할 추정 프레임들을 생성하기 위해 모션 벡터 필드를 사용한다. 예를 들면, 24 FPS로부터 60 FPS로의 프레임 레이트 상향 변환은 AA'A"BB'로서 표현될 수 있으며, 여기에서 A'는 프레임 A로부터 보간함으로써 생성된 제 1 추정 프레임이고, A"는 프레임 A로부터 보간함으로써 생성된 제 2 추정 프레임이며, B'는 프레임 B로부터 보간함으로써 생성된 추정 프레임이다. 그러나, 비디오 프레임 레이트 상향 변환은 계산 집약적이며, 이것은 비디오 디스플레이 디바이스에 의한 전력 사용을 상당히 증가시키고 모션 벡터 필드의 무차별 대입 산출들을 수행하기에 충분한 계산 전력을 가진 비디오 디스플레이 디바이스들에 대한 프레임 레이트 상향 변환의 가능성을 제한한다.

[0004] 비디오 디스플레이 디바이스들에 의한 전력 소비는 또한 계산적으로 덜 강력한 비디오 디스플레이 디바이스들이 비디오 서버에서 스트림으로 프레임에 대한 모션 추정을 수행하며 그 후 프레임에 대한 모션 벡터 필드를 나타내는 메타데이터를 가진 비디오 디스플레이 디바이스로 프레임을 제공함으로써 비디오 프레임 레이트 상향 변환으로부터 이익을 얻도록 허용하면서 감소될 수 있다. 몇몇 실시예들에서, 메타데이터는 또한 (0) 보간이 모션 벡터에 기초하여 수행되지 않고, (1) 보간이 단지 시간적으로 앞으로 수행되고, (2) 보간이 단지 시간적으로 뒤로 수행되거나, 또는 (3) 보간이 시간적으로 양방향으로 수행됨을 나타내는 모션 벡터 필드 또는 플래그들에서 모션 벡터들에 대한 신뢰 척도들을 포함한다. 비디오 서버는 제 1 프레임 레이트로 스트림에서 프레임을 제공하며 메타데이터를 다중화하거나 또는 그 외 그것을 스트림으로 통합한다. 비디오 서버의 몇몇 실시예들은 또한 프레임 상에서 씬(scene) 변화 검출을 수행하며 씬 변화가 프레임에서 검출되었는지를 나타내는 부가적인 메타데이터를 제공하도록 구성된다. 모션 벡터 프로세싱은 이웃하는 모션 벡터들과 예상 외로 상이한 이상치 모션 벡터들을 식별하기 위해 사용되며, 예로서 그것들은 반대 방향을 가리키거나 또는 이웃하는 모션 벡터들에 대한 평균과 훨씬 상이한 크기를 가진다. 이상치 모션 벡터들은 이웃하는 모션 벡터들의 값들에 기초하여 무시되거나 또는 수정될 수 있다. 오클루전(occlusion) 검출은 오클루전된 모션 벡터들에 기초하여 보간이 수행되지 않고, 단지 시간적으로 앞으로 수행되고, 단지 시간적으로 뒤로 수행되거나, 또는 시간적으로 양방향으로 수행되도록 오클루전에 의해 영향을 받은 프레임의 부분들에 대한 모션 벡터들을 식별하기 위해 사용될 수 있다. 몇몇 실시예들에서, 모션 벡터 프로세싱 및 오클루전 검출은 비디오 서버에 의해 수행되며, 이것은 프레임에서 이상치 모션 벡터들 또는 오클루전된 모션 벡터들을 나타내는 메타데이터를 생성하고 프레임을 메타데이터에 제공한다. 비디오 디스플레이 디바이스는 대응하는 메타데이터와 함께 스트림에서 프레임을 수신하며 메타데이터에 기초하여 프레임으로부터 보간함으로써 추정된 프레임들을 생성하기 위해 메타데이터를 사용한다. 추정 프레임들은 제 1 프레임 레이트에서 제 2(더 높은) 프레임 레이트로 스트림에서 프레임들의 프레임 레이트 상향 변환을 위해 사용된다.

[0005] 도 1은 몇몇 실시예들에 따른 비디오 획득 및 디스플레이 시스템(100)의 다이어그램이다. 시스템(100)은 비디오 카메라와 같은 비디오 획득 디바이스(105)를 포함한다. 비디오 획득 디바이스(105)는 독립형 디바이스일 수 있거나 또는 비디오 획득 디바이스(105)는 데스크탑 컴퓨터, 랩탑 컴퓨터, 태블릿 컴퓨터, 스마트폰 등과 같은 또 다른 컴퓨팅 디바이스로 통합될 수 있다. 비디오 획득 디바이스(105)는 씬(110)의 일련의 이미지들을 획득한다. 예시된 실시예에서, 씬(110)은 필드(115), 사람(120), 및 공(125)을 포함한다. 그러나, 씬(110)은 비디오 획득 디바이스(105)에 의해 모니터링될 수 있는 임의의 씬일 수 있다. 비디오 획득 디바이스(105)에 의해 캡처된 이미지들은 프레임에서 픽셀들의 값들로 표현된다. 비디오 획득 디바이스(105)는 초당 24 프레임들(FPS) 또는 30 FPS와 같은, 프레임 레이트로 캡처된 이미지에 기초하여 프레임들을 생성한다.

[0006] 비디오 획득 디바이스(105)에 의해 생성된 프레임들은 프레임들을 저장하며(적어도 일시적으로) 예로서, 매개 네트워크(140)를 통해, 하나 이상의 비디오 클라이언트들(135)로 프레임들을 제공하도록 구성되는 비디오 서버(130)로 제공된다. 예를 들면, 씬(110)은 사용자가 비디오 클라이언트(135)의 스크린(145) 상에서 보고 있는 축구 또는 풋볼 시합의 일 부분을 포함한다. 이 예에서, 비디오 서버(130)는 비디오 획득 디바이스(105)에 의해 생성된 프레임들의 스트림을 수신하며 비디오 획득 디바이스(105)의 프레임 레이트로 프레임들의 스트림을 비디오 클라이언트(135)로 송신한다. 여기에서 논의된 바와 같이, 비디오 획득 디바이스(105)의 프레임 레이트는 비디오 클라이언트(135)에서 프레임들의 스트림에 의해 표현된 비디오를 디스플레이하기 위해 사용될 수 있는 프

레이트에 반드시 일치하는 것은 아니다. 예를 들면, 비디오 획득 디바이스(105)는 24 FPS의 프레임 레이트로 이미지들을 획득할 수 있는 반면, 비디오 클라이언트(135)는 30 FPS, 48 FPS, 60 FPS 이상과 같은 더 높은 프레임 레이트들로 프레임들을 디스플레이할 수 있다.

[0007] 비디오 클라이언트(135)는 더 낮은 프레임 레이트(24 FPS와 같은)로 수신된 프레임들을 더 높은 프레임 레이트들(60 FPS와 같은)로 디스플레이될 수 있는 보다 많은 수의 프레임들로 변환하기 위해 비디오 프레임 레이트 상향 변환을 수행할 수 있다. 예를 들면, 비디오 클라이언트(135)는 비디오 서버(130)로부터 수신된 프레임들 사이에서 보간함으로써 부가적인 프레임들을 생성할 수 있다. 비디오 클라이언트(135)는 기준 프레임과 수신된 프레임들의 블록-기반 비교들, 수신된 프레임들의 광학적 흐름 분석, 또는 수신된 프레임들의 부분들의 상관, 예로서 자동상관들, 컨볼루션들, 교차-상관들, 또는 위상 상관들을 사용하여 생성되는 수신된 프레임들의 부분들의 모션 벡터들과 같은, 수신된 프레임들로부터 도출된 보간 파라미터들에 기초하여 보간을 수행할 수 있다. 그러나, 여기에서 논의되는 바와 같이, 보간 파라미터들을 생성하는 것은 계산 집약적이며, 이것은 비디오 클라이언트(135)에서 전력 소비를 증가시키고 비디오 레이트 상향 변환을 수행할 수 있는 비디오 클라이언트들(135)의 유형들을 제한할 수 있다.

[0008] 비디오 클라이언트(135)에 대한 계산 부담을 감소시키기 위해, 비디오 서버(130)의 몇몇 실시예들은 비디오 획득 디바이스(105)로부터 수신된 프레임들을 사용하여 보간 파라미터들을 생성한다. 예를 들면, 비디오 서버(130)는 비디오 획득 디바이스(105)에 의해 사용된 프레임 레이트(예로서, 24 FPS)로부터 비디오 클라이언트(135)에서 프레임들을 디스플레이하기 위해 사용된 프레임 레이트(예로서, 60 FPS)로 비디오 레이트 상향 변환을 수행하기 위해 사용될 수 있는 보간 파라미터들의 하나 이상의 세트들을 생성할 수 있다. 비디오 획득 디바이스(105)에 의해 생성된 프레임들의 스트림에서 제 1 프레임에 대한 보간 파라미터들은 비디오 획득 디바이스(105)에 의해 생성된 스트림에서 제 1 프레임 다음에 그리고 제 2 프레임에 앞서 씬(110)을 나타내는 하나 이상의 보간 프레임들을 생성하기 위해 사용된다. 비디오 서버(130)는 그 후 보간 파라미터들을 나타내는 메타데이터를 생성하며 메타데이터를 다중화하거나 또는 그 외 비디오 클라이언트(135)로 송신되는 프레임들의 스트림으로 통합한다. 따라서, 비디오 클라이언트(135)는 보간 파라미터들을 생성하는 계산 집약적 태스크를 수행하도록 요구되지 않는다.

[0009] 비디오 클라이언트(135)는 비디오 서버(130)로부터 다중화된 메타데이터를 포함하는 프레임들의 스트림을 수신한다. 예를 들면, 비디오 클라이언트(135)는 제 1 프레임의 부분들에 대한 보간 파라미터들을 나타내는 다중화된 메타데이터를 포함하는 프레임들의 스트림에서 씬(110)을 나타내는 제 1 프레임을 수신할 수 있다. 비디오 클라이언트(135)는 그 후 프레임들의 스트림에서 제 1 프레임 다음에 그리고 제 2 프레임에 앞서 시간 간격들에서 씬을 나타내는 하나 이상의 보간 파라미터들을 생성할 수 있다. 예를 들면, 비디오 클라이언트(135)는 보간 프레임들에서 픽셀들의 추정된 값들을 생성하도록 제 1 프레임에서 픽셀들의 값들을 보간하기 위해 제 1 프레임의 부분들(픽셀들 또는 픽셀들의 그룹들과 같은)에 대한 모션 벡터들을 사용할 수 있다. 보간된 프레임들의 수는 비디오 획득 디바이스(105)에 의해 사용된 프레임 레이트 및 비디오 클라이언트(135)에 의해 사용된 프레임 레이트의 비에 기초하여 결정된다. 예를 들면, 비디오 클라이언트(135)는 24 FPS에서 60 FPS로 3:2 프레임 레이트 상향 변환을 수행하기 위해 제 1 프레임에 대해 두 개의 보간된 프레임들을 및 제 2 프레임에 대해 하나의 보간된 프레임을 반복적으로 생성할 수 있다. 비디오 클라이언트(135)는 그 후 제 1 프레임, 제 1 프레임으로부터 보간된 두 개의 프레임들, 제 2 프레임, 제 2 프레임으로부터 보간된 하나의 프레임 등을 디스플레이한다. 보간은 여기에서 논의되는 바와 같이, 몇몇 실시예들에서 신뢰 척도들 또는 플래그들에 기초하여 선택적으로 수행된다. 예를 들면, 보간은 메타데이터에서 플래그들의 값들에 기초하여 바이패스되고, 시간적으로 앞으로 수행되고, 시간적으로 뒤로 수행되거나, 또는 시간적으로 양방향으로 수행될 수 있다.

[0010] 도 2는 몇몇 실시예들에 따라 비디오 프레임(200) 및 모션 벡터들에 기초하여 생성된 보간된 프레임(205)을 예시하는 다이어그램이다. 비디오 프레임(200)은 도 1에 도시된 비디오 획득 디바이스(105)의 몇몇 실시예들에 의해 생성된 프레임들을 나타낸다. 보간된 프레임(205)은 도 1에 도시된 비디오 클라이언트(135)의 몇몇 실시예들에 의해 생성된 보간된 프레임들을 나타낸다.

[0011] 비디오 프레임(200)은 비디오 획득 디바이스에 의해 모니터링되는 씬을 나타내는 값들을 가진 픽셀들의 어레이로 구성된다. 예를 들면, 픽셀들(210, 211, 212)(여기에서 총괄하여 "픽셀들(210 내지 212)"로 불리우는)은 비디오 프레임(200)에서 사람(215)의 대응 부분들을 나타내는 값들을 가진다. 또 다른 예를 들면, 픽셀들(220, 221)은 비디오 프레임(200)에서 공(225)의 대응 부분들을 나타내는 값들을 가진다. 또 다른 예를 들면, 픽셀(230)은 비디오 프레임(200)에서 펜드(235)의 대응 부분을 나타내는 값을 가진다.

[0012]

픽셀들은 대응하는 모션 벡터들과 연관된다. 예를 들면, 픽셀들(210 내지 212)은 픽셀들(210 내지 212)에 대해 추정된 모션의 진폭들 및 방향들을 나타내는 대응하는 모션 벡터들(240, 241, 242)(여기에서 총괄하여 "모션 벡터들(240 내지 242)"로 불리우는)을 가진다. 또 다른 예를 들면, 픽셀들(220, 221)은 픽셀들(220, 221)에 대해 추정된 모션의 진폭들 및 방향들을 나타내는 대응하는 모션 벡터들(243, 244)을 가진다. 픽셀(230)은 필드(235)의 정지 부분을 나타내는 값을 가지며 따라서 픽셀(230)과 연관된 모션 벡터는 없다. 대안적으로, 0의 진폭을 가지며 방향이 없는(또는 임의의 방향) 모션 벡터는 픽셀(230)과 연관될 수 있다. 여기에서 논의된 바와 같이, 모션 벡터들(240 내지 244)(뿐만 아니라 다른 보간 파라미터들)은 도 1에 도시된 비디오 서버(130)와 같은 비디오 서버에 의해 픽셀들(210 내지 212, 220, 221, 230)에 대해 결정된다. 개개의 픽셀들(210 내지 212, 220, 221, 230)이 도 2에 묘사되지만, 픽셀들(210 내지 212, 220, 221, 230)은 또한 몇몇 실시예들에서 픽셀들의 16 × 16 블록들과 같은 픽셀들의 블록들을 나타낸다. 비디오 서버는 비디오 프레임(200)을 나타내는 정보와 모션 벡터들(240 내지 244)(또는 다른 보간 파라미터들)을 나타내는 메타데이터를 다중화하며 그 후 다중화된 프레임/메타데이터 스트림을 도 1에 도시된 비디오 클라이언트(135)와 같은 비디오 클라이언트로 송신한다.

[0013]

비디오 클라이언트는, 수신된 프레임들 및 메타데이터를 사용하여, 예로서 수신된 프레임들 및 메타데이터에 기초하여 보간된 프레임들을 생성함으로써, 비디오 레이트 상향 변환을 수행한다. 예시된 실시예에서, 보간된 프레임(205)은 모션 벡터들(240 내지 244)에 기초하여 보간된 프레임(205)에서 픽셀들의 값들을 생성하기 위해 비디오 프레임(200)에서 픽셀들의 값들을 보간함으로써 생성된다. 예를 들면, 픽셀들(250, 251, 252)(여기에서 총괄하여 "픽셀들(250 내지 252)"로 불리우는)의 값들은 모션 벡터들(240 내지 242)을 사용하여 픽셀들(210 내지 212)의 값들을 보간함으로써 생성된다. 또 다른 예를 들면, 픽셀들(253, 254)의 값들은 모션 벡터들(243, 244)을 사용하여 픽셀들(220, 221)의 값들을 보간함으로써 생성된다. 픽셀들(210 내지 212, 220, 221) 및 픽셀들(250 내지 254) 사이에서의 오프셋은 모션 벡터들(240 내지 244)의 진폭들 및 방향들과 비디오 프레임(200) 및 보간된 프레임(205) 사이에서의 시간 간격에 의해 결정된다. 픽셀(230)은 보간된 프레임(205)에서 비디오 프레임(200)으로부터의 값을 유지한다.

[0014]

도 3은 몇몇 실시예들에 따라 비디오 서버(305) 및 비디오 클라이언트(310)를 포함하는 비디오 프로세싱 시스템(300)의 제 1 예를 예시한 블록도이다. 비디오 프로세싱 시스템(300)은 도 1에 도시된 비디오 획득 및 디스플레이 시스템(100)의 몇몇 실시예들을 구현하기 위해 사용된다. 예를 들면, 비디오 서버(305)는 비디오 서버(130)의 몇몇 실시예들을 구현하기 위해 사용되며 비디오 클라이언트(310)는 도 1에 도시된 비디오 클라이언트(135)의 몇몇 실시예들을 구현하기 위해 사용된다. 비디오 서버(305)는 24 FPS와 같은 제 1 프레임 레이트로 비디오 획득 디바이스(도 1에 도시된 비디오 획득 디바이스(105)와 같은)에 의해 제공되는 프레임들을 포함하는 스트림(315)을 수신한다.

[0015]

비디오 서버(305)는 수신된 프레임들에서 픽셀들 또는 픽셀들의 그룹들에 대한 모션 벡터들을 추정하기 위해 사용되는 모션 추정 모듈(320)을 포함한다. 예를 들면, 모션 추정 모듈(320)은 스트림에서 이전에 수신된 프레임과 같은, 기준 프레임에서의 픽셀들의 값들에 현재 프레임에서의 픽셀들의 값들을 비교할 수 있다. 비교는 후보 모션 벡터에 의해 결정된 오프셋만큼 현재 프레임에서의 픽셀들을 시프트하고 그 후 기준 프레임에서의 픽셀들의 값들에 오프셋 픽셀들의 값들을 비교함으로써 수행된다. 그러나, 여기에서 논의된 바와 같이, 비교는 또한 상관 분석들, 광학적 흐름 분석 등에 기초하여 수행될 수 있다. 픽셀 값들의 유사도가 그 후 계산된다. 이러한 프로세스는 후보 모션 벡터들의 세트에 대해 반복되며 최고 유사도를 가진 후보 모션 벡터가 픽셀(또는 픽셀들의 그룹)에 대한 모션 벡터로서 선택된다. 몇몇 실시예들에서, 모션 추정 모듈(320)은 상이한 후보 모션 벡터들 사이에서의 유사도들의 구배를 측정하며, 또한 후보 모션 벡터들 사이에서의 "거리에서의 비용(cost in distance)"으로서 비교한다. 구배 및 비용은 픽셀 또는 픽셀들의 그룹에 대한 모션 벡터로서 후보 모션 벡터들 중 하나를 선택하기 위해 가중되고 조합된다. "거리에서의 비용"은 L-1 놈(norm), 예로서 후보 모션 벡터들 사이에서의 거리의 그리드 측정치에 대한 택시 거리, 피타고라스의 정리에 따라 후보 모션 벡터들 사이에서의 유클리드 거리를 결정하는 L-2 놈, 또는 상이한 후보 모션 벡터들 사이에서의 거리를 특성화하는 다른 측정치들을 사용하여 결정될 수 있다. 몇몇 실시예들에서, "러너-업(runner-up)" 모션 벡터가 또한 1차 모션 벡터 외에 선택될 수 있다.

[0016]

모션 추정 모듈(320)의 몇몇 실시예들은 선택된 모션 벡터 및 임의의 "러너-업" 모션 벡터들에 대한 신뢰 척도들을 생성한다. 신뢰 척도들은 선택된 모션 벡터가 대응하는 픽셀 또는 픽셀들의 그룹에서 표현된 이미지의 부분의 모션을 정확하게 나타내는 가능성을 표시한다. 예를 들면, 벡터에 대한 신뢰 척도는 범위 0..n 내에서의 수로 표현될 수 있으며, 보다 작은 수들은 더 낮은 신뢰 레벨들을 나타내며 보다 큰 수들은 더 높은 신뢰 레벨들을 나타낸다. 신뢰 척도들을 나타내는 수들은 부동 소수점 수들, 3-비트 수들, 또는 다른 표현들일 수 있다.

[0017]

모션 추정 모듈(320)은 스트림(315)의 프레임들에서 픽셀들(또는 픽셀들의 그룹들)에 대한 모션 벡터들을 나타내는 메타데이터를 생성한다. 예를 들면, 픽셀들(또는 픽셀들의 그룹들)의 각각에 대한 모션 벡터들은 스크린의 평면에서 X 및 Y 방향들로 차동 거리들(dx, dy)로서 표현될 수 있다. 또 다른 예를 들면, 픽셀들(또는 픽셀들의 그룹들)의 각각에 대한 모션 벡터들은 모션 벡터의 진폭을 나타내는 정보 및 프레임에서 모션 벡터의 방향을 나타내는 정보로 표현될 수 있다. 모션 벡터들의 각각에 대한 메타데이터는 또한 대응하는 픽셀들(또는 픽셀들의 그룹들)을 식별하는 정보를 포함한다. 모션 추정 모듈(320)의 몇몇 실시예들은 또한 메타데이터에서 모션 벡터들의 각각에 대한 신뢰 척도들을 포함한다. 도 3에 도시된 모션 추정 모듈(320)은 모션 벡터들을 계산하지만, 모션 추정 모듈(320)의 몇몇 실시예들은 광학적 흐름 결과들, 상관 분석 결과들 등과 같은 다른 보간 파라미터들을 생성한다. 모션 추정 모듈(320)은 그러므로 이들 다른 보간 파라미터들을 나타내는 메타데이터를 생성할 수 있다.

[0018]

비디오 서버(305)의 몇몇 실시예들은 씬 변화 검출 모듈(325)을 포함한다. 씬 변화는 현재 프레임에 의해 표현된 씬이 스트림(315)에서 이전 프레임에 의해 표현된 씬과 상이할 때 발생한다. 씬 변화 검출 모듈(325)은 이전 프레임에서의 픽셀들의 값들에 현재 프레임에서의 픽셀들의 값들을 비교함으로써 씬 변화들을 검출할 수 있다. 예를 들면, 씬 변화가 현재 프레임 및 이전 프레임 사이에서 발생하면, 현재 프레임 및 이전 프레임에서의 픽셀들 중 일부 또는 모두의 값들은 비연속적으로 변한다. 씬 변화 검출 모듈(325)은 그러므로 현재 프레임 및 이전 프레임에서 픽셀들의 값들 사이에서의 차이들의 측정치들을 결정할 수 있다. 거리 측정치가 임계치보다 크면, 씬 변화 검출 모듈(325)은 씬 변화를 검출한다. 씬 변화 검출 모듈(325)은 씬 변화가 없는 경우 "0"의 값을 및 씬 변화가 검출되는 경우 "1"의 값을 제공받은 비트와 같은, 씬 변화를 나타내기 위해 메타데이터를 생성할 수 있다. 메타데이터의 값은 스트림(315)에서 프레임들 사이에서 보간을 시도할지를 결정하기 위해 사용된다.

[0019]

스트림(315)에서의 프레임들, 모션 추정 모듈(320)에 의해 생성된 메타데이터, 씬 변화 검출 모듈(325)에 의해 생성된 메타데이터, 및 비디오 서버(305)에서 임의의 다른 비디오 프로세싱 모듈들에 의해 생성된 메타데이터는 다중화기(330)로 제공된다. 다중화기(330)는 메타데이터를 다중화하거나 또는 그 외 스트림(315)으로 통합한다. 예를 들면, 다중화기(330)는 프레임들의 각각과 연관된 메타데이터에 의해 분리된 스트림(315)에서 프레임들을 포함하는 출력 스트림(335)을 생성할 수 있다. 출력 스트림(335)은 비디오 클라이언트(310)로 송신된다. 몇몇 실시예들에서, 프레임들 및 메타데이터는 비디오 서버(305)에 저장된다. 다중화된 출력 스트림(335)은 그 후 비디오 클라이언트(310)로부터의 요청에 응답하여 비디오 클라이언트(310)로 제공된다. 결과적으로, 메타데이터는 실시간으로 생성될 필요가 없다.

[0020]

비디오 클라이언트(310)의 몇몇 실시예들은 오클루전 및 모션 벡터 프로세싱 모듈(340)을 포함한다. 오클루전은 씬에서의 하나의 오브젝트가 또 다른 오브젝트 앞에 또는 뒤에서 지나갈 때 발생한다. 예를 들면, 공이 나무 뒤에서 이동할 때, 공의 부분들은 나무에 의해 가려진다. 현재 프레임에서 가려진 이전 프레임에서 오브젝트의 부분들의 모션 벡터들은 그것이 가려진 오브젝트의 부분들에 대응하는 값들을 할당받은 오클루전 오브젝트의 부분들을 나타내는 픽셀들을 값들을 야기할 수 있기 때문에 보간을 위해 사용되지 않아야 한다. 예를 들면, 프레임에서의 모션 벡터들에 기초하여 나무 뒤에서 이동하는 공을 포함한 씬을 나타내는 프레임을 보간하는 것은 공의 부분이 보간된 프레임에서 나무 앞에서 이동하는 것처럼 보이는 것을 야기할 수 있다. 오클루전 및 모션 벡터 프로세싱 모듈(340)은 씬의 부분들에서 오클루전을 검출하고 대응하는 메타데이터를 생성할 수 있다. 오클루전 및 모션 벡터 프로세싱 모듈(340)의 몇몇 실시예들은 시간적으로 앞으로 결정된 모션 벡터들(예로서, 이전 프레임에 대하여 현재 프레임에서 모션 벡터들을 결정함으로써) 및 시간적으로 뒤로 결정되는 모션 벡터들(예로서, 현재 프레임에 대하여 이전 프레임에서 모션 벡터들을 결정함으로써)을 비교함으로써 오클루전을 검출한다. 모션 벡터들이 일치하면, 오클루전은 가능성이 없다. 그러나, 전방향 및 역방향 모션 벡터들은 오클루전이 존재하는 경우 다를 것이다. 오클루전 및 모션 벡터 프로세싱 모듈(340)은 픽셀들(또는 픽셀들의 그룹들)에 대한 모션 벡터들은 모션 벡터에서 낮은 신뢰도를 나타내기 위해 0 또는 다른 낮은 값의 신뢰 척도를 제공받을 수 있다. 또 다른 예를 들면, 오클루전된 픽셀들(또는 픽셀들의 그룹들)은 모션 벡터와 연관된 어떤 오클루전도 없는 경우 "0"의 값을 및 모션 벡터에 대해 오클루전이 검출되는 경우 "1"의 값을 제공받는 비트와 연관될 수 있다. 메타데이터의 값은 스트림(335)에서 프레임들 사이에서의 보간을 위해 모션 벡터를 사용할지를 결정하기 위해 사용된다.

[0021]

오클루전 및 모션 벡터 프로세싱 모듈(340)은 또한 예러들 또는 아티팩트들일 수 있는 이상치 모션 벡터들을 검출하기 위해 사용될 수 있다. 예를 들면, 오클루전 및 모션 벡터 프로세싱 모듈(340)은 제 1 픽셀의 모션 벡터가 하나 이상의 이웃하는 픽셀들의 모션 벡터들과 통계적으로 상이하다고 결정한다면, 오클루전 및 모션 벡터

프로세싱 모듈(340)은 이상치로서 제 1 픽셀의 모션 벡터를 식별한다. 통계적 차이들의 예들은 이웃하는 모션 벡터들의 진폭들의 평균 값으로부터 벗어난 미리 결정된 수의 표준 편차들보다 큰 진폭, 이웃하는 모션 벡터들의 평균 방향으로부터 벗어난 미리 결정된 수의 표준 편차들보다 큰 방향 등을 가진 모션 벡터들을 포함한다. 오클루전 및 모션 벡터 프로세싱 모듈(340)의 몇몇 실시예들은 이웃하는 모션 벡터들의 값들에 기초하여, 예로서 이웃하는 모션 벡터들의 진폭들 또는 방향들의 평균으로 이상치 모션 벡터의 진폭 또는 방향을 대체함으로써 이상치 모션 벡터를 수정한다. 모션 벡터들은 또한, 예로서 국소 평균치들로 또는 가장-유사한 이웃하는 모션 벡터로 이상치들을 대체하는 공간-시간적 메리디안 필터들을 사용하여, 이상치들을 제거하기 위해 필터링될 수 있다. 이상치 모션 벡터들(또는 이상치 모션 벡터들의 대체된 값들)과 연관된 신뢰 척도들은 모션 벡터들의 정확도에서의 낮은 신뢰도를 나타내기 위해 낮은 값으로 설정될 수 있다. 오클루전 및 모션 벡터 프로세싱 모듈(340)은 이상치 모션 벡터에 대한 수정들을 나타내거나 또는 이상치 모션 벡터가 보간을 위해 사용되어야 하는지를 나타낼 수 있는 신뢰 척도들과 같은 메타데이터를 생성할 수 있다.

[0022] 비디오 클라이언트(310)에서 보간 모듈(345)은 스트림(315)의 프레임들을 포함한 출력 스트림(335) 및 비디오 서버(305)에 의해 생성된 메타데이터, 뿐만 아니라 오클루전 및 모션 벡터 프로세싱 모듈(340)에 의해 생성된 임의의 메타데이터를 수신한다. 보간 모듈(345)은 여기에서 논의된 바와 같이, 하나 이상의 보간된 프레임들을 생성하기 위해 수신된 비디오 프레임들 및 메타데이터를 사용한다. 보간 모듈(345)은 그 후 스트림(315)에서의 프레임들 및 프레임들 및 메타데이터에 기초하여 생성된 보간된 프레임들을 포함하는 보간된 비디오 스트림(350)을 제공한다.

[0023] 도 4는 몇몇 실시예들에 따라 비디오 서버(405) 및 비디오 클라이언트(410)를 포함하는 비디오 프로세싱 시스템(400)의 제 2 예를 예시한 블록도이다. 비디오 프로세싱 시스템(400)은 도 1에 도시된 비디오 획득 및 디스플레이 시스템(100)의 몇몇 실시예들을 구현하기 위해 사용된다. 예를 들면, 비디오 서버(405)는 비디오 서버(130)의 몇몇 실시예들을 구현하기 위해 사용되며 비디오 클라이언트(410)는 도 1에 도시된 비디오 클라이언트(135)의 몇몇 실시예들을 구현하기 위해 사용된다. 비디오 서버(405)는 24 FPS와 같은 제 1 프레임 레이트로 비디오 획득 디바이스(도 1에 도시된 비디오 획득 디바이스(105)와 같은)에 의해 제공되는 프레임들을 포함하는 스트림(415)을 수신한다.

[0024] 비디오 서버(405)는 모션 벡터들을 결정하며 스트림(415)의 프레임들에서 픽셀들(또는 픽셀들의 그룹들)에 대한 모션 벡터들을 나타내는 메타데이터를 생성하는 모션 추정 모듈(420)을 포함한다. 비디오 서버(405)는 또한 스트림(415)의 프레임들에서 씬 변화들을 검출하고 씬 변화를 나타내기 위해 메타데이터를 생성하는 씬 변화 검출 모듈(425)을 포함한다. 모션 추정 모듈(420) 및 씬 변화 검출 모듈(425)은 도 3에 도시된 모션 추정 모듈(320) 및 씬 변화 검출 모듈(325)의 몇몇 실시예들과 동일한 방식으로 동작하도록 구성된다.

[0025] 도 4에 묘사된 비디오 프로세싱 시스템(400)의 제 2 예는 비디오 서버(405)가 오클루전 및 모션 벡터 프로세싱 모듈(430)을 구현하기 때문에 도 3에 도시된 비디오 프로세싱 시스템(300)의 제 1 예와 상이하다. 오클루전 및 모션 벡터 프로세싱 모듈(430)의 적당히 계산 집약적 동작들을 이동시키는 것은 비디오 클라이언트(410)에 대한 계산 부담을 감소시킨다. 오클루전 및 모션 벡터 프로세싱 모듈(430)은 스트림(415)의 프레임들에서 오클루전을 검출하며 프레임들에서 픽셀들(또는 픽셀들의 그룹들)이 오클루전을 경험하는지를 나타내는 메타데이터를 생성하도록 구성된다. 오클루전 및 모션 벡터 프로세싱 모듈(430)은 또한 스트림(415)의 프레임들에서 이상치 모션 벡터들을 검출하도록 구성된다. 오클루전 및 모션 벡터 프로세싱 모듈(430)의 몇몇 실시예들은 여기에서 논의된 바와 같이, 이상치 모션 벡터들의 값들을 수정하며, 이상치 모션 벡터에 대한 수정들을 나타내거나 또는 이상치 모션 벡터가 보간을 위해 사용되어야 하는지를 나타낼 수 있는 메타데이터를 생성한다.

[0026] 스트림(415)에서의 프레임들, 모션 추정 모듈(420)에 의해 생성된 메타데이터, 씬 변화 검출 모듈(425)에 의해 생성된 메타데이터, 오클루전 및 모션 벡터 프로세싱 모듈(430)에 의해 생성된 메타데이터, 및 비디오 서버(405)에서 임의의 다른 비디오 프로세싱 모듈들에 의해 생성된 메타데이터는 다중화기(435)로 제공된다. 다중화기(435)는 메타데이터를 다중화하거나 또는 그 외 스트림(415)으로 통합한다. 예를 들면, 다중화기(435)는 프레임들의 각각과 연관된 메타데이터에 의해 분리된 스트림(415)에서의 프레임들을 포함하는 출력 스트림(440)을 생성할 수 있다. 출력 스트림(440)은 비디오 클라이언트(410)로 송신된다. 몇몇 실시예들에서, 프레임들 및 메타데이터는 비디오 서버(405)에 저장된다. 다중화된 출력 스트림(440)은 그 후 비디오 클라이언트(410)로부터의 요청에 응답하여 비디오 클라이언트(410)로 제공된다. 결과적으로, 메타데이터는 실시간으로 생성될 필요가 없다.

[0027] 비디오 클라이언트(410)에서 보간 모듈(445)은 스트림(415)의 프레임들 및 비디오 서버(405)에 의해 생성된 메

타데이터를 포함한 출력 스트림(440)을 수신한다. 보간 모듈(445)은 여기에서 논의된 바와 같이, 하나 이상의 보간된 프레임들을 생성하기 위해 수신된 비디오 프레임들 및 메타데이터를 사용한다. 보간 모듈(445)은 그 후 스트림(415)에서의 프레임들 및 프레임들과 메타데이터에 기초하여 생성된 보간된 프레임들을 포함하는 보간된 비디오 스트림(450)을 제공한다.

[0028] 도 5는 몇몇 실시예들에 따라 비디오 프레임들, 메타데이터, 및 보간된 프레임들을 예시하는 비디오 프로세싱 시스템(500)의 블록도이다. 비디오 프로세싱 시스템(500)은 비디오 서버(505) 및 비디오 클라이언트(510)를 포함하며, 이것은 도 1, 도 3, 및 도 4에 도시된 비디오 서버들(130, 305, 405) 및 비디오 클라이언트들(135, 310, 410)의 몇몇 실시예들을 사용하여 구현된다.

[0029] 비디오 서버(505)는 비디오 프레임들(515, 520)을 포함하는 스트림을 수신한다(또는 생성한다). 비디오 서버(505)는 또한 대응하는 비디오 프레임들(515, 520)에 대한 메타데이터(525, 530)를 생성한다. 메타데이터는 모션 추정 모듈, 씬 변화 검출 모듈, 오클루전 및 모션 벡터 프로세싱 모듈(비디오 서버(505)에서 구현된다면), 또는 비디오 서버(505)에 구현된 다른 비디오 프로세싱 모듈들에 의해 생성될 수 있다. 비디오 프레임들(515, 520) 및 메타데이터(525, 530)는 다중화기(535)로 제공되며, 이것은 비디오 프레임들(515, 520) 및 메타데이터(525, 530)를 다중화하거나 또는 그 외 출력 스트림(540)으로 통합한다.

[0030] 비디오 서버(505)의 몇몇 실시예들은 출력 스트림(540)을 형성하기 위해 비디오 프레임들(515, 520) 및 메타데이터(525, 530)를 압축한다. 출력 스트림(540)을 형성하는 비트들을 압축하는 것은 출력 스트림(540)을 송신하도록 요구된 대역폭에서의 작은 증가만을 갖고 비디오 품질을 상당히 개선할 수 있다. 예를 들면, 넷플릭스(Netflix)와 같은 서비스들은 24 FPS의 프레임 레이트를 갖는 영화에 대해 압축된 데이터의 화면당 대략 208,000비트들에 대응하는, 대략 5Mbit/초의 레이트로 데이터를 스트리밍한다. 압축된 데이터는 대략 2500만 비트들의 압축되지 않은 데이터를 나타낸다. 메타데이터는 모션 벡터 정보 및 신뢰 척도들을 포함하는 16-비트 벡터, 예로서 수평 모션을 나타내기 위한 6비트들, 수직 모션을 나타내기 위한 6비트들, 신뢰 척도를 나타내기 위한 2비트들, 및 모션 벡터가 오클루전과 연관되며 전방향, 역방향, 또는 양방향에서 보간을 위해 사용되어야 하는지를 나타내기 위한 2비트들로서 표현될 수 있다. 여기에서 논의되는 바와 같이, 1920×1080 픽셀 스크린 상에 디스플레이된 프레임당 8100 벡터들이 있으며, 이것은 프레임당 $8100 \times 16 = 129,600$ 비트들의 압축되지 않은 데이터를 야기한다. 메타데이터에 대한 압축비의 보수적 추정치는 이미지들에서 모션 간에 통상적으로 많은 양의 상관으로 인해 10:1이다. 따라서, 압축된 메타데이터는 압축된 비디오 프레임들에 비교하여 매우 작은 대역폭을 소비한다. 대역폭은 그러므로 아껴 쓰여질 수 있으며, 예로서 60 Hz 비디오를 수송하기 위해 요구된 대역폭은 수송되지 않은 프레임들을 어떻게 복원하거나 또는 보간할지를 나타내는 메타데이터를 포함한 30Hz 비디오로서 60Hz 비디오를 수송함으로써 거의 50%만큼 감소될 수 있다.

[0031] 비디오 서버(505)의 몇몇 실시예들은 또한 출력 스트림(540)으로 축소되거나 또는 챔네일 버전들을 다중화할 수 있다. 이것은 비디오 서버(505)가 더 높은 프레임 레이트를 가진 스트림에 있으며 더 낮은 프레임 레이트로 남아있는 프레임들을 수송하는 몇몇 프레임들을 빼도록 허용한다. 비디오 서버(505)는 그 후 비디오 클라이언트(510)가 더 높은 프레임 레이트로 수신된 프레임들을 갖고 디스플레이를 위한 프레임들을 재구성하거나 또는 보간하기 위해 축소되거나 또는 챔네일 버전들을 사용할 수 있도록 뺀 프레임들의 축소된 또는 챔네일 버전들로 출력 스트림(540)에서의 정보를 보완할 수 있다. 축소된 또는 챔네일 버전들은 또한 오클루전 면적들의 형태들을 식별하거나 또는 이미지의 오클루전 면적들 또는 애매한 면적들에서 보간을 수행하기 위해 사용될 수 있다.

[0032] 비디오 클라이언트(510)는 비디오 서버(505)로부터 출력 스트림(540)을 수신한다. 비디오 클라이언트(510)는 여기에서 논의되는 바와 같이, 보간된 프레임들(545, 550, 555)을 생성하기 위해 비디오 프레임들(515, 520) 및 메타데이터(525, 530)를 사용한다. 예를 들면, 메타데이터(525)는 보간된 프레임들(545, 550)의 픽셀 값들을 생성하도록 비디오 프레임(515)에서 픽셀 값들을 보간하기 위해 사용된다. 또 다른 예를 들면, 메타데이터(530)는 보간된 프레임(555)의 픽셀 값들을 생성하도록 비디오 프레임(520)에서 픽셀 값들을 보간하기 위해 사용된다. 비디오 클라이언트(510)는 비디오 프레임들(515, 520) 및 보간된 프레임들(545, 550, 555)을 포함하는 디스플레이 스트림(560)을 생성한다. 디스플레이 스트림(560)은 비디오 클라이언트(510)의 스크린상에 비디오를 디스플레이하기 위해 사용된다.

[0033] 도 6은 몇몇 실시예들에 따라 비디오 프레임들로부터 메타데이터를 생성하기 위한 비디오 서버(605) 및 메타데이터 및 비디오 프레임들에 기초하여 보간된 프레임들을 생성하기 위한 비디오 클라이언트(610)를 포함하는 비디오 프로세싱 시스템(600)의 블록도이다. 비디오 서버(605) 및 비디오 클라이언트(610)는 도 1 및 도 3 내지 도 5에 도시된 비디오 서버들(130, 305, 405, 505) 및 비디오 클라이언트들(135, 310, 410, 510)의 몇몇 실시

예들을 구현하기 위해 사용된다.

[0034] 비디오 서버(605)는 신호들을 송신하고 수신하기 위한 네트워크 인터페이스(615)(예로서, 네트워크 인터페이스)를 포함한다. 예를 들면, 네트워크 인터페이스(615)는 비디오 획득 디바이스(620)에 의해 생성된 스트림에서 프레임들을 나타내는 신호들을 수신할 수 있다. 네트워크 인터페이스(615)는 또한 여기에서 논의된 바와 같이, 비디오 프레임들 및 연관된 메타데이터를 나타내는 신호들을 송신할 수 있다. 네트워크 인터페이스(615)는 단일 집적 회로로서(예로서, 단일 ASIC 또는 FPGA를 사용하여) 또는 네트워크 인터페이스(615)의 기능을 구현하기 위한 상이한 모듈들을 포함하는 시스템-온-칩(SOC)으로서 구현될 수 있다. 비디오 서버(605)는 또한 프로세서(625) 및 메모리(630)를 포함한다. 프로세서(625)는 메모리(630)에 저장된 지시들을 실행하기 위해 및 비디오 프레임들 또는 연관된 메타데이터를 포함할 수 있는, 실행된 지시들의 결과들과 같은 정보를 메모리(630)에 저장하기 위해 사용될 수 있다.

[0035] 비디오 클라이언트(610)는 신호들을 송신하고 수신하기 위한 네트워크 인터페이스(635)를 포함한다. 예를 들면, 네트워크 인터페이스(635)는 비디오 프레임들 및 비디오 서버(605)에 의해 생성된 메타데이터를 나타내는 신호들을 수신할 수 있다. 또 다른 예를 들면, 네트워크 인터페이스(635)는 비디오 프레임들 및 수신된 메타데이터에 기초하여 생성된 보간된 프레임들을 디스플레이를 위해 스크린(640)으로 송신할 수 있다. 네트워크 인터페이스(635)는 단일 집적 회로로서(예로서, 단일 ASIC 또는 FPGA를 사용하여) 또는 네트워크 인터페이스(635)의 기능을 구현하기 위한 상이한 모듈들을 포함하는 시스템-온-칩(SOC)으로서 구현될 수 있다. 비디오 클라이언트(610)는 또한 프로세서(645) 및 메모리(650)를 포함한다. 프로세서(645)는 메모리(650)에 저장된 지시들을 실행하며 실행된 지시들의 결과들과 같은 정보를 메모리(650)에 저장하기 위해 사용될 수 있다. 예를 들면, 프로세서(645)는 비디오 프레임들 및 비디오 서버(605)로부터 수신된 메타데이터에 기초하여 보간된 프레임들을 생성하기 위해 사용될 수 있다. 보간된 프레임들은 그 후 스크린(640) 상에 이미지들을 생성하기 위해 네트워크 인터페이스(635)로 제공된다.

[0036] 도 7은 몇몇 실시예들에 따른 이미지에서의 오브젝트들과 연관된 모션 벡터들을 결정하기 위해 탐색될 수 있는 이미지를 디스플레이하는 스크린(700)을 포함한 다이어그램이다. 스크린(700)은 픽셀들의 1920×1080 어레이이지만, 스크린(700)의 다른 실시예들이 상이한 수들의 로우들 또는 컬럼들로 배열된 상이한 수들의 픽셀들을 포함한다. 예시된 실시예에서, 현재 프레임에 도시된 이미지는 스크린(700)에 의해 구현된 어레이에서 픽셀들의 상이한 값들에 의해 표현되는 사람(705), 공(710), 및 필드(715)를 포함한다.

[0037] 이미지에서 모션을 검출하기 위해, 현재 프레임에서 이미지를 나타내는 픽셀들의 값들의 서브세트들은 이전 프레임에서 이미지를 나타내는 픽셀들의 값들의 기준 서브세트들에 비교된다. 예를 들면, 스크린(700)의 픽셀들은 탐색 윈도우(720)와 같은 64×64 탐색 윈도우들로 분할될 수 있으며 그 후 탐색 윈도우(720) 내에서 16×16 탐색 블록들이 공(710)의 이전 위치를 나타내는 픽셀들의 값들을 포함하는 16×16 기준 블록(725)과 같은 기준 블록들에 비교된다. 탐색 프로세스는 그러므로 탐색 윈도우(720) 내에서 후보 모션 벡터들을 평가하기 위해 기준 블록(725)에 대하여 상이한 오프셋들에서 탐색 블록들의 $64 \times 64 = 4096$ 개별 비교들을 요구한다. 각각의 비교의 결과들은 스코어(S)에 의해 표현될 수 있다:

$$S = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} |C_{ij} - R_{ij}|$$

[0038] 여기에서 C_{ij} 는 탐색 윈도우(720)에서 비교 블록 내에서의 위치(i, j)에서 픽셀들의 값들을 나타내며 R_{ij} 는 기준 블록(725) 내에서의 위치(i, j)에서 픽셀들의 값들을 나타낸다. 따라서, 현재 예에서, 각각의 스코어는 256개 감산들 및 256개 절대 값 연산들을 요구한다.

[0040] 모션 벡터 탐색의 계산 부담은 스코어들이 탐색 윈도우(720)와 같은 각각의 후보 면적에 대한 탐색을 수행하기 위해 프로세싱 코어 당 대략 30개 지시들을 요구하는 단일-지시-다중-데이터(SIMD) 그래픽스 프로세싱 유닛(GPU)을 사용하여 결정된다고 가정함으로써 추정될 수 있다. 스크린(700) 상에서의 픽셀들에 의해 표현된 이미지의 하나의 완전한 탐색은 그러므로 GPU 상에서 $4096 \times 30 = 122,880$ 사이클들을 요구한다. 상이한 기준 블록들의 수는 스크린(700) 상에서의 픽셀들에 의해 표현된 이미지에 대해 $1920/16 \times 1080/16 = 8100$ 기준 블록들이다.

각각의 이미지를 탐색하기 위해 요구된 사이클들의 총 수는 그러므로 $8100 \times 122,880 \approx 10^9$ 사이클들이다. 여기에서 논의된 바와 같이, 오클루전 검출 및 다른 기능들은 순방향 탐색(예로서, 이전 프레임에 대하여 현재 프레임을 비교하는) 및 역방향 탐색(예로서, 현재 프레임에 대하여 이전 프레임을 비교하는)을 수행하는 것을 요구

하며, 이것은 이미지당 사이클들의 수의 두 배이다. 통상적인 입력 프레임 레이트는 24 FPS이며, 이것은 초당 480억 사이클들의 총 프로세서 요건을 야기한다. 이러한 양의 프로세싱 전력은 모든 디바이스들 상에서 이용 가능하지 않으며, 그것이 이용 가능할 때, 상당한 양의 전력을 소비한다. 더욱이, 이러한 추정치는 부가적인 산출들이 통상적으로 후-처리를 위해, 예로서 이상치들, 오클루전들 등을 찾고 헨들링하기 위해 요구되기 때문에 보다 낮은 추정치이다. 게다가, 부가적인 산출들은 상이한 스케일들로 표현된 이미지 상에서 수행될 수 있다.

[0041] 본 출원에서 설명된 비디오 획득 및 디스플레이 시스템들의 몇몇 실시예들은 관례에 비해 다수의 이점들을 가진다. 예를 들면, 비디오 서버에서 모션 추정(및 몇몇 경우들에서 다른 비디오 프로세싱)을 수행하는 것, 및 비디오 클라이언트들로 보간 파라미터들을 나타내는 메타데이터를 가진 비디오 프레임들을 제공하는 것은 비디오 프레임 레이트 상향 변환을 지원하는 비디오 클라이언트들에 대한 최소 요건들을 감소시킬 뿐만 아니라, 비디오 클라이언트들에서 전력 소비를 감소시킨다. 비디오 클라이언트로부터 비디오 서버로 모션 추정을 시프트하는 것(및 몇몇 경우들에서 다른 비디오 프로세싱)은 또한, 비디오 서버의 계산 리소스들을 사용하여 또는 보다 넓은 범위의 가능한 선택들을 조사하고 어떤 선택들이 최상의 비디오 품질을 야기할지를 결정하기 위해 보다 정교한 분석을 사용하여 보다 정교한 모션 추정을 구현함으로써, 비디오 클라이언트에서 비디오 품질을 증가시킬 뿐만 아니라, 아티팩트들의 발생률 및 심각도를 감소시킨다. 더욱이, 몇몇 실시예들에서, 모션 추정(및 몇몇 경우들에서 다른 비디오 프로세싱)은 비디오 서버에서 실시간으로 수행되도록 요구되지 않는다. 예를 들면, 비디오 스트림을 위한 메타데이터는 비디오 스트림이 비디오 클라이언트에 의해 요청되며 그 후 요청 시 제공되기 전에 생성될 수 있다.

[0042] 몇몇 실시예들에서, 상기 설명된 장치 및 기술들은 도 1 내지 도 6을 참조하여 상기 설명된 비디오 획득 및 디스플레이 시스템들과 같은, 하나 이상의 집적 회로(IC) 디바이스들(또한 집적 회로 패키지들 또는 마이크로칩들로 불리우는)을 포함한 시스템에서 구현된다. 전자 설계 자동화(EDA) 및 컴퓨터 보조 설계(CAD) 소프트웨어 툴들은 이들 IC 디바이스들의 설계 및 제작에 사용된다. 이들 설계 툴들은 통상적으로 하나 이상의 소프트웨어 프로그램들로서 표현된다. 하나 이상의 소프트웨어 프로그램들은 회로를 제작하도록 제조 시스템을 설계하거나 또는 적응시키기 위해 프로세스의 적어도 일 부분을 수행하도록 하나 이상의 IC 디바이스들의 회로를 나타내는 코드 상에서 동작하도록 컴퓨터 시스템을 조작하기 위해 컴퓨터 시스템에 의해 실행 가능한 코드를 포함한다. 이러한 코드는 지시들, 데이터, 또는 지시들 및 데이터의 조합을 포함할 수 있다. 설계 툴 또는 제작 툴을 나타내는 소프트웨어 지시들은 통상적으로 컴퓨팅 시스템으로 액세스 가능한 컴퓨터 판독 가능한 저장 매체에 저장된다. 마찬가지로, IC 디바이스의 설계 또는 제작의 하나 이상의 단계들을 나타내는 코드는 동일한 컴퓨터 판독 가능한 저장 매체 또는 상이한 컴퓨터 판독 가능한 저장 매체에 저장되며 그로부터 액세스된다.

[0043] 컴퓨터 판독 가능한 저장 매체는 컴퓨터 시스템으로 지시들 및/또는 데이터를 제공하기 위해 사용 동안 컴퓨터 시스템에 의해 액세스 가능한 임의의 비-일시적 저장 매체, 또는 비-일시적 저장 미디어의 조합을 포함한다. 이러한 저장 미디어는, 이에 제한되지 않지만, 광학 미디어(예로서, 컴팩트 디스크(CD), 디지털 다목적 디스크(DVD), 블루-레이 디스크), 자기 미디어(예로서, 플로피 디스크, 자기 테이프, 또는 자기 하드 드라이브), 휘발성 메모리(예로서, 랜덤 액세스 메모리(RAM) 또는 캐시), 비-휘발성 메모리(예로서, 판독-전용 메모리(ROM) 또는 플래시 메모리), 또는 미세전자기계 시스템들(MEMS)-기반 저장 미디어를 포함할 수 있다. 컴퓨터 판독 가능한 저장 매체는, 몇몇 구현들에서, 컴퓨팅 시스템에 내장되고(예로서, 시스템 RAM 또는 ROM), 컴퓨팅 시스템에 단단히 부착되고(예로서, 자기 하드 드라이브), 컴퓨팅 시스템에 착탈 가능하게 부착되거나(예로서, 광학 디스크 또는 범용 직렬 버스(USB)-기반 플래시 메모리), 또는 유선 또는 무선 네트워크를 통해 컴퓨터 시스템에 결합될 수 있다(예로서, 네트워크 액세스 가능한 저장 장치(NAS)).

[0044] 몇몇 실시예들에서, 상기 설명된 기술들의 특정한 양상들은 소프트웨어를 실행하는 프로세싱 시스템의 하나 이상의 프로세서들에 의해 구현될 수 있다. 소프트웨어는 비-일시적 컴퓨터 판독 가능한 저장 매체 상에 저장되거나 또는 그 외 유형으로 내장된 실행 가능한 지시들의 하나 이상의 세트들을 포함한다. 소프트웨어는, 하나 이상의 프로세서들에 의해 실행될 때, 상기 설명된 기술들 중 하나 이상의 양상들을 수행하도록 하나 이상의 프로세서들을 조작하는 지시들 및 특정한 데이터를 포함할 수 있다. 비-일시적 컴퓨터 판독 가능한 저장 매체는, 예를 들면, 자기 또는 광학 디스크 저장 디바이스, 플래시 메모리와 같은 고체 상태 저장 디바이스들, 캐시, 랜덤 액세스 메모리(RAM) 또는 다른 비-휘발성 메모리 디바이스 또는 디바이스들 등을 포함할 수 있다. 비-일시적 컴퓨터 판독 가능한 저장 매체에 저장된 실행 가능한 지시들은 소스 코드, 어셈블리어 코드, 오브젝트 코드, 또는 하나 이상의 프로세서들에 의해 해석되거나 또는 그 외 실행 가능한 다른 지시 포맷에 있을 수 있다.

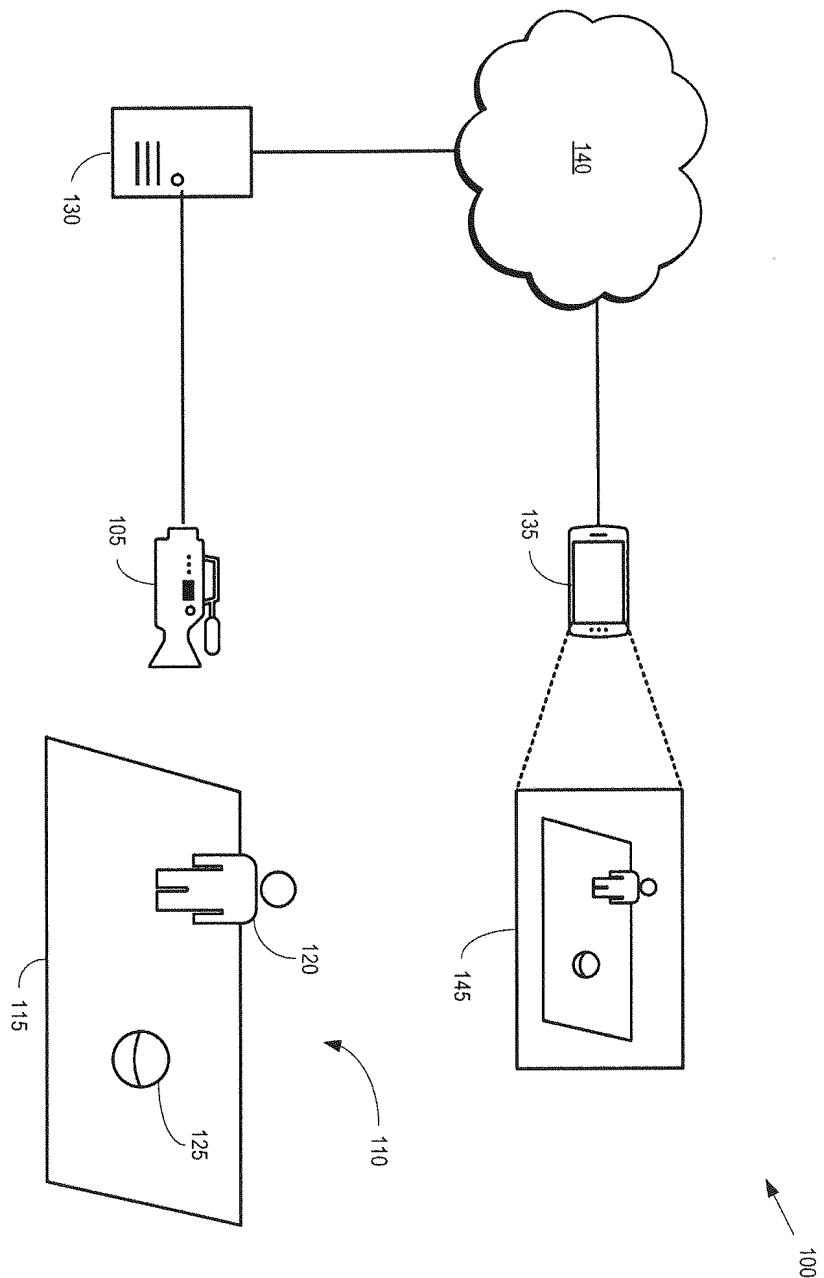
[0045] 전반적인 설명에서 상기 설명된 활동들 또는 요소들 모두가 요구되는 것은 아니고, 특정 활동 또는 디바이스의 일 부분이 요구되지 않을 수 있으며, 설명된 것들 외에, 하나 이상의 추가 활동들, 또는 포함된 요소들이 수행

될 수 있다는 것을 주의하자. 더 나아가, 활동들이 나열되는 순서는 반드시 그것들이 수행되는 순서는 아니다. 또한, 개념들은 특정 실시예들을 참조하여 설명되었다. 그러나, 이 기술분야의 통상의 기술자는 다양한 수정들 및 변화들이 이하에서 청구항들에서 제시된 바와 같이 본 개시의 범위로부터 벗어나지 않고 이루어질 수 있다는 것을 이해할 것이다. 따라서, 명세서 및 도면들은 제한적인 의미보다는 예시적인 것으로 간주될 것이며, 모든 이러한 수정들은 본 개시의 범위 내에 포함되도록 의도된다.

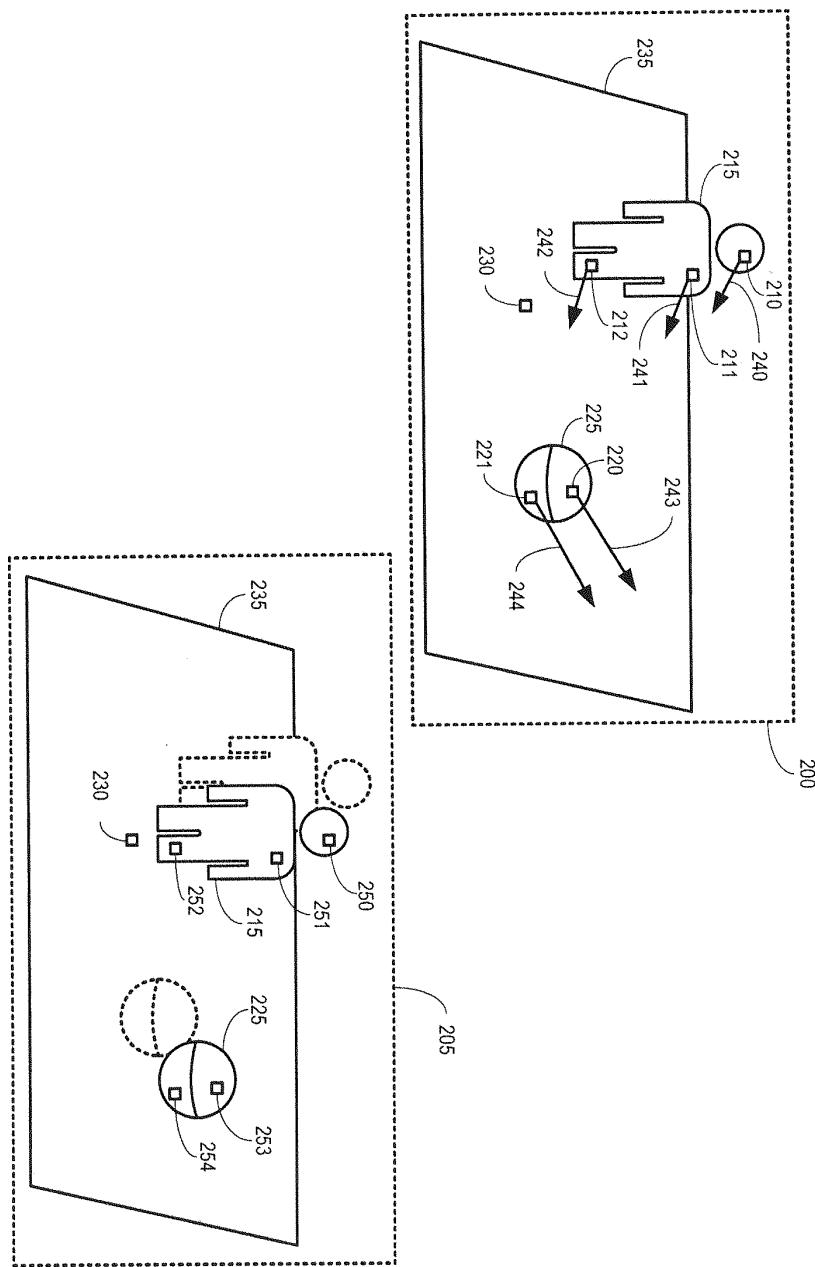
[0046] 이익들, 다른 이점들, 및 문제들에 대한 해법들은 특정 실시예들에 대하여 상기 설명되었다. 그러나, 이익들, 이점들, 문제들에 대한 해법들, 및 임의의 이익, 이점, 또는 해법이 발생하거나 또는 보다 확연해지게 할 수 있는 임의의 특징(들)은 임의의 또는 모든 청구항들의 중대한, 필수적인, 또는 극히 중요한 특징으로서 해석되지 않을 것이다. 게다가, 상기 개시된 특정한 실시예들은, 개시된 주제가 여기에서의 교시들의 이익을 가진 기술분야에서의 숙련자들에게 분명한 상이하지만 동등한 방식들로 수정되고 실시될 수 있으므로, 단지 예시적이다. 어떤 제한들도, 이항서의 청구항들에서 설명된 것 외에, 여기에서 도시된 구성 또는 설계의 세부사항들에 대해 의도되지 않는다. 그러므로, 상기 개시된 특정한 실시예들은 변경되거나 또는 수정될 수 있으며 모든 이러한 변화들은 개시된 주제의 범위 내에서 고려된다는 것이 분명하다. 따라서, 여기에서 추구되는 보호는 이하에서의 청구항들에서 제시된 바와 같다.

도면

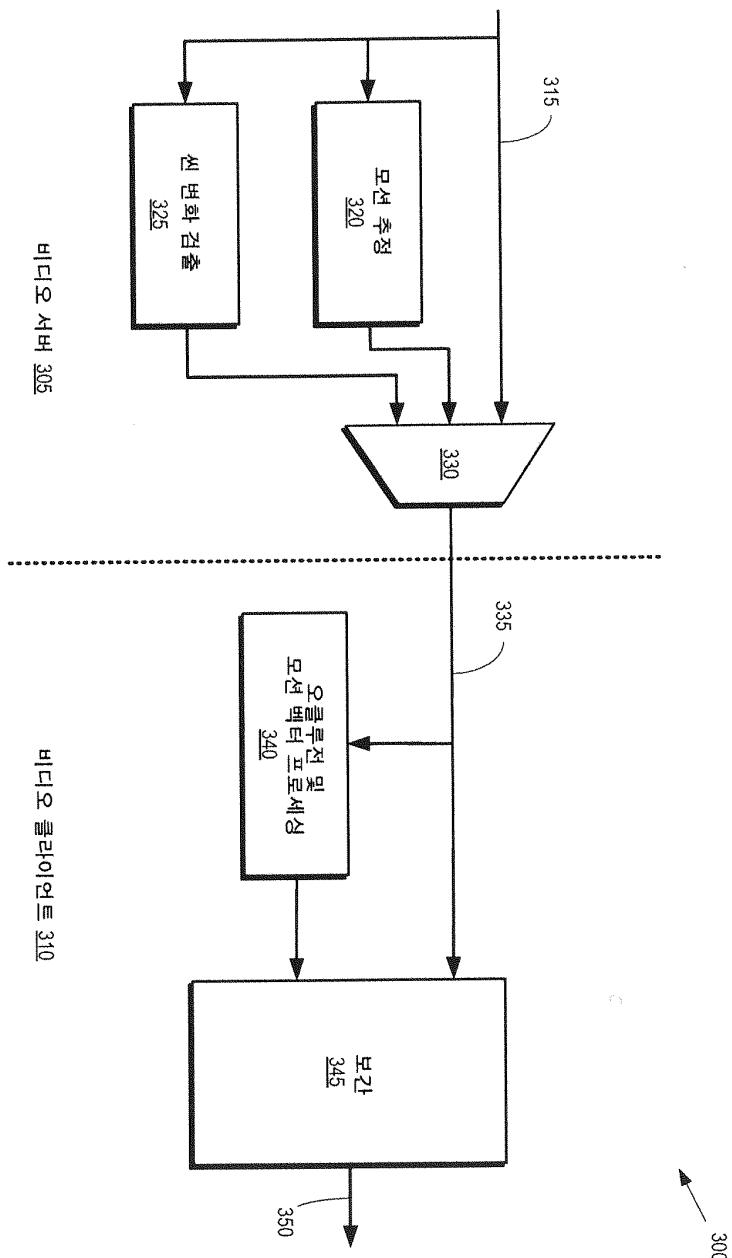
도면1



도면2



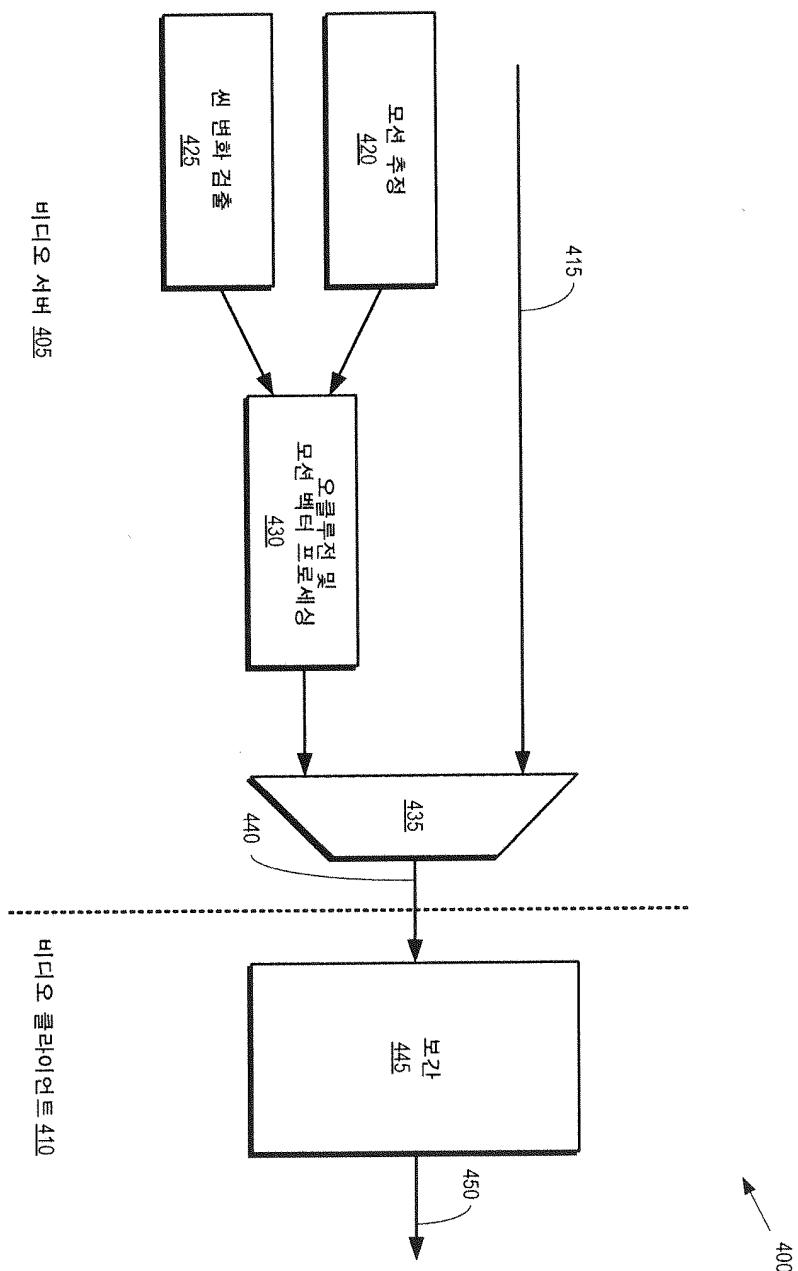
도면3



비디오 서버 305

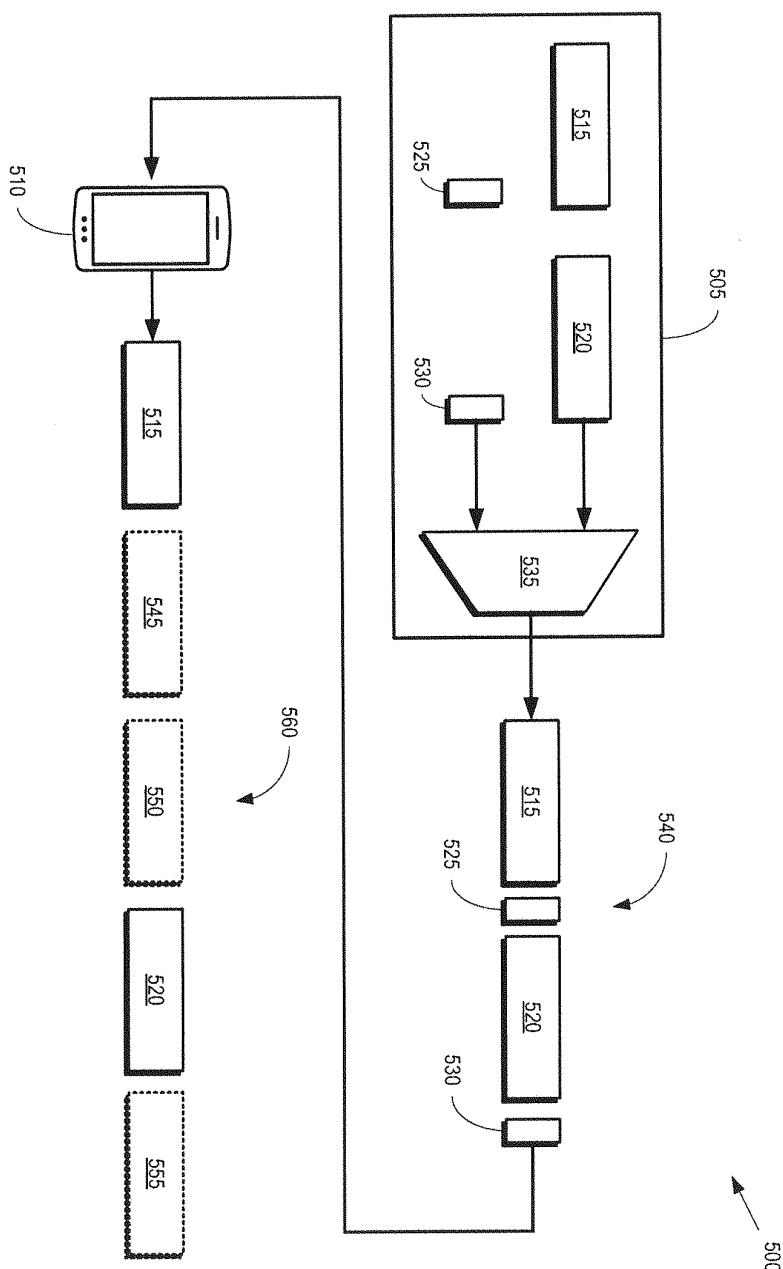
비디오 클라이언트 310

도면4

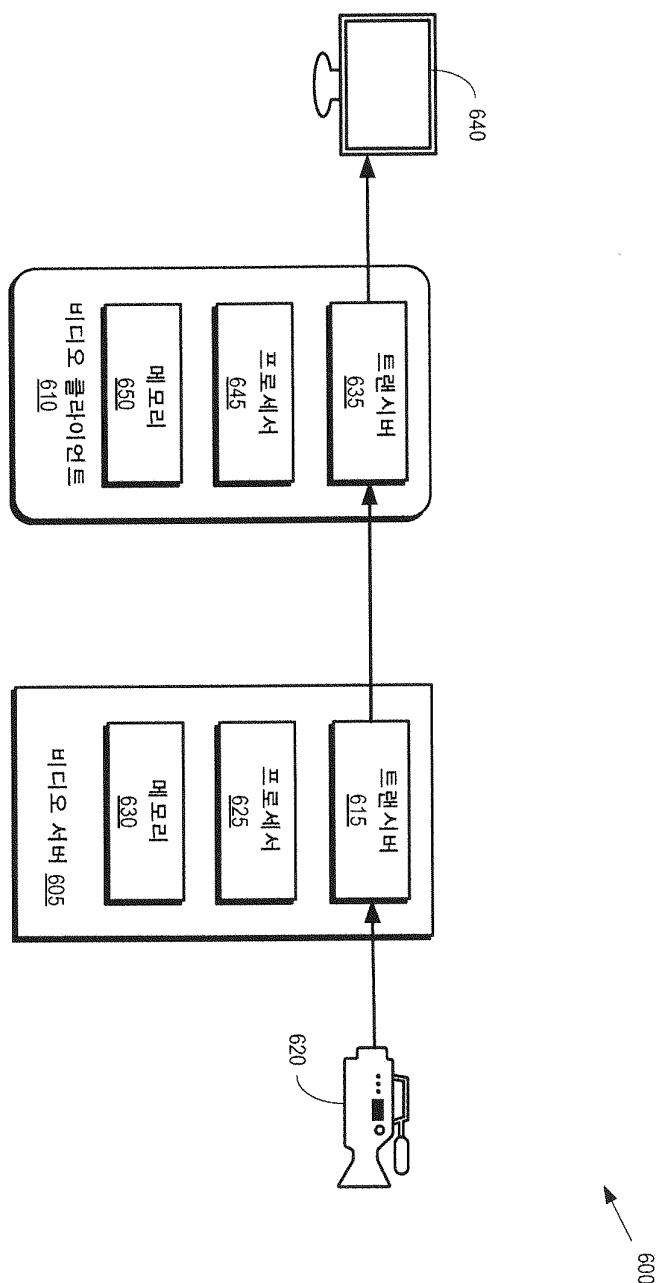


비디오 서버 405

도면5



도면6



도면7

