



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2023-0004124
(43) 공개일자 2023년01월06일

(51) 국제특허분류(Int. Cl.)
G06F 18/00 (2023.01) G06N 3/08 (2023.01)
H04N 7/18 (2023.01)
(52) CPC특허분류
G06V 20/52 (2023.01)
G06F 18/22 (2023.01)
(21) 출원번호 10-2021-0085994
(22) 출원일자 2021년06월30일
심사청구일자 2021년06월30일

(71) 출원인
세종대학교산학협력단
서울특별시 광진구 능동로 209 (군자동, 세종대학교)
(72) 발명자
김영갑
서울특별시 광진구 능동로 209, 세종대학교
대양AI 센터 701호
김유준
서울특별시 광진구 능동로 209, 세종대학교
대양AI 센터 732호
(74) 대리인
특허법인이상

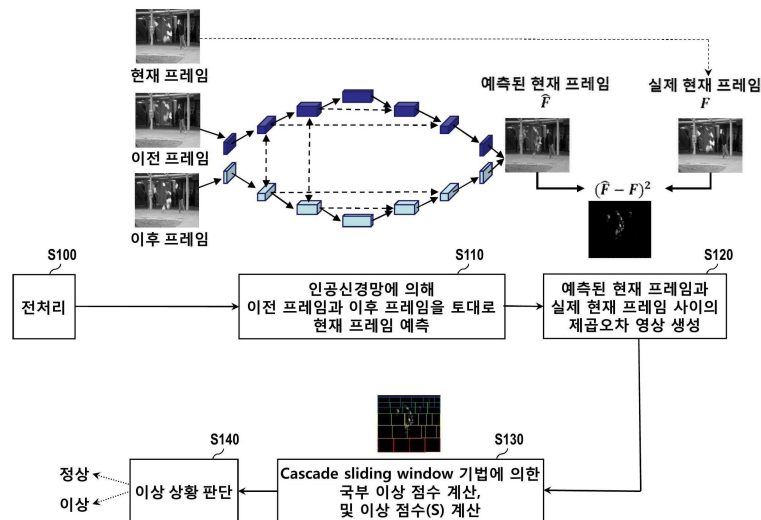
전체 청구항 수 : 총 20 항

(54) 발명의 명칭 영상 감시 시스템에서의 실시간 이상 상황 탐지 방법 및 장치

(57) 요약

영상 감시 시스템에서 감시 영상을 토대로 이상 상황을 정확하게 탐지할 수 있을 뿐만 아니라 신속하게 탐지할 수 있는 방법 및 장치를 제공한다. 본 발명의 이상 상황 탐지 방법은 시간적으로 연이어진 일련의 영상을 받아들이고 상기 영상 내에 이상 상황이 포함되어 있는지 탐지하기 위한 것으로서, 현재 프레임보다 시간적으로 앞선 이전 프레임과 상기 현재 프레임보다 시간적으로 뒤진 이후 프레임을 토대로 현재 프레임을 예측하여, 예측된 현재 프레임을 생성하는 단계; 상기 예측된 현재 프레임과 실제 현재 프레임에 대한 차이를 나타내는 이상 점수를 산출하는 단계; 및 상기 이상 점수가 소정의 조건을 충족하는 경우 상기 현재 프레임에 이상 상황이 포함되어 있는 것으로 결정하는 단계;를 포함한다.

대표도



(52) CPC특허분류

G06N 3/08 (2023.01)

H04N 7/18 (2023.01)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711103210
과제번호	2019-0-00231
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	정보보호핵심원천기술개발(R&D)
연구과제명	공공 인프라 안전을 위한 인공지능 기반 영상보안 기술 및 시스템 개발
기 여 율	1/1
과제수행기관명	세종대학교 산학협력단
연구기간	2021.01.01 ~ 2021.12.31

명세서

청구범위

청구항 1

시간적으로 연이어진 일련의 영상을 받아들이고 상기 영상 내에 이상 상황이 포함되어 있는지 탐지하는 방법으로서,

현재 프레임보다 시간적으로 앞선 이전 프레임과 상기 현재 프레임보다 시간적으로 뒤진 이후 프레임을 토대로 현재 프레임을 예측하여, 예측된 현재 프레임을 생성하는 단계;

상기 예측된 현재 프레임과 실제 현재 프레임에 대한 차이를 나타내는 이상 점수를 산출하는 단계; 및

상기 이상 점수가 소정의 조건을 충족하는 경우 상기 현재 프레임에 이상 상황이 포함되어 있는 것으로 결정하는 단계;를 포함하는 이상 상황 탐지 방법.

청구항 2

청구항 1에 있어서, 상기 이전 프레임은 상기 현재 프레임보다 복수의 프레임만큼 시간적으로 앞선 프레임이고, 상기 이후 프레임은 상기 현재 프레임보다 복수의 프레임만큼 시간적으로 뒤진 프레임인 이상 상황 탐지 방법.

청구항 3

청구항 2에 있어서, 상기 예측된 현재 프레임을 생성하는 단계는 상기 이전 프레임을 입력으로 받아들이는 제1 서버 네트워크와 상기 이후 프레임을 입력으로 받아들이는 제2 서버 네트워크를 포함하는 인공지능망에 의해 상기 예측된 현재 프레임을 도출하며,

상기 인공지능망에서 상기 제1 및 상기 제2 서버 네트워크 각각이 복수의 컨볼루션 레이어와 적어도 하나의 맥스풀링 레이어를 갖는 경로를 구비하는 이상 상황 탐지 방법.

청구항 4

청구항 3에 있어서, 상기 제1 및 상기 제2 서버 네트워크가 상대측 서버 네트워크의 피쳐 맵과 적어도 한번 결합되도록 구성되는 이상 상황 탐지 방법.

청구항 5

청구항 3에 있어서, 상기 인공지능망은 정상적인 상황에서 상기 이전 프레임과 상기 이후 프레임을 토대로 상기 예측된 현재 프레임을 예측할 수 있게 사전에

훈련된 후 사용되는 이상 상황 탐지 방법.

청구항 6

청구항 1에 있어서, 상기 이상 점수를 산출하는 단계는

상기 실제 현재 프레임에 대하여 소정의 윈도우를 사전에 정해진 이동간격 단위로 수평 및 수직 방향으로 이동시키면서, 상기 윈도우와 중첩되는 프레임 부분에서 상기 예측된 현재 프레임과 상기 실제 현재 프레임의 대응하는 픽셀들 간의 화소 값 차이에 대하여 소정의 연산을 행함으로써 국부 이상 점수를 계산하는 단계; 및

상기 윈도우의 이동 위치들에 상응하여 계산되는 복수의 국부 이상 점수들을 평균하거나 합산하여 상기 이상 점수를 결정하는 단계;를 포함하는 이상 상황 탐지 방법.

청구항 7

청구항 6에 있어서, 상기 국부 이상 점수를 계산하는 단계에서,

상기 윈도우를 이동시키면서, 상기 윈도우와 중첩되는 프레임 부분에서 상기 예측된 현재 프레임과 상기 실제 현재 프레임의 대응하는 픽셀들 간의 상기 화소 값 차이의 제곱을 평균하거나 합산하여 상기 국부 이상 점수를

계산하는 이상 상황 탐지 방법.

청구항 8

청구항 6에 있어서, 상기 이상 점수를 결정하는 단계는

상기 윈도우의 이동 위치들에 상응한 상기 국부 이상 점수들 중에서 크기순으로 정해지는 소정 개수의 점수들만을 평균하여 상기 이상 점수를 결정하는 단계;를 포함하는 이상 상황 탐지 방법.

청구항 9

청구항 6에 있어서, 상기 윈도우는 상기 실제 현재 프레임에 대하여 위로 이동할수록 상기 윈도우의 크기가 작아지도록 설정되는 이상 상황 탐지 방법.

청구항 10

청구항 1에 있어서,

상기 예측된 현재 프레임을 생성하는 단계를 수행하기 이전에, 상기 일련의 영상을 전처리하여 흑백 영상으로의 변환 및 영상 해상도 조정 중 적어도 하나를 포함하는 전처리를 수행하는 단계;를 더 포함하는 이상 상황 탐지 방법.

청구항 11

시간적으로 연이어진 일련의 영상을 받아들이고 상기 영상 내에 이상 상황이 포함되어 있는지 탐지하는 장치로서,

프로그램 명령들을 저장하는 메모리와; 상기 메모리에 통신가능하게 접속되고 상기 메모리에 저장된 상기 프로그램 명령들을 실행하는 프로세서;를 구비하며,

상기 프로그램 명령들은 상기 프로세서에 의해 실행될 때 상기 프로세서로 하여금:

현재 프레임보다 시간적으로 앞선 이전 프레임과 상기 현재 프레임보다 시간적으로 뒤진 이후 프레임을 토대로 현재 프레임을 예측하여, 예측된 현재 프레임을 생성하는 동작;

상기 예측된 현재 프레임과 실제 현재 프레임에 대한 차이를 나타내는 이상 점수를 산출하는 동작; 및

상기 이상 점수가 소정의 조건을 충족하는 경우 상기 현재 프레임에 이상 상황이 포함되어 있는 것으로 결정하는 동작;을 수행하도록 하는 이상 상황 탐지 장치.

청구항 12

청구항 11에 있어서, 상기 이전 프레임은 상기 현재 프레임보다 복수의 프레임만큼 시간적으로 앞선 프레임이고, 상기 이후 프레임은 상기 현재 프레임보다 복수의 프레임만큼 시간적으로 뒤진 프레임인 이상 상황 탐지 장치.

청구항 13

청구항 12에 있어서, 상기 프로세서로 하여금 상기 예측된 현재 프레임을 생성하는 동작을 수행하도록 하는 명령들은, 상기 프로세서로 하여금, 상기 이전 프레임을 입력으로 받아들이는 제1 서버 네트워크와 상기 이후 프레임을 입력으로 받아들이는 제2 서버 네트워크를 포함하는 인공지능망에 의해 상기 예측된 현재 프레임을 도출하도록 하며,

상기 인공지능망에서 상기 제1 및 상기 제2 서버 네트워크 각각은 복수의 컨볼루션 레이어와 적어도 하나의 맥스풀링 레이어를 갖는 경로를 구비하는 이상 상황 탐지 장치.

청구항 14

청구항 13에 있어서, 상기 제1 및 상기 제2 서버 네트워크가 상대측 서버 네트워크의 피쳐 맵과 적어도 한번 결합되도록 구성되는 이상 상황 탐지 장치.

청구항 15

청구항 13에 있어서, 상기 인공신경망은 정상적인 상황에서 상기 이전 프레임과 상기 이후 프레임을 토대로 상기 예측된 현재 프레임을 예측할 수 있게 사전에 훈련되어 있는 이상 상황 탐지 장치.

청구항 16

청구항 11에 있어서, 상기 프로세서로 하여금 상기 이상 점수를 산출하는 동작을 수행하도록 하는 명령들은, 상기 프로세서로 하여금:

상기 실제 현재 프레임에 대하여 소정의 윈도우를 사전에 정해진 이동간격 단위로 수평 및 수직 방향으로 이동시키면서, 상기 윈도우와 중첩되는 프레임 부분에서 상기 예측된 현재 프레임과 상기 실제 현재 프레임의 대응하는 픽셀들 간의 화소 값 차이에 대하여 소정의 연산을 행함으로써 국부 이상 점수를 계산하는 동작; 및

상기 윈도우의 이동 위치들에 상응하여 계산되는 복수의 국부 이상 점수들을 평균하거나 합산하여 상기 이상 점수를 결정하는 동작;을 수행하도록 하는 이상 상황 탐지 장치.

청구항 17

청구항 16에 있어서, 상기 프로세서로 하여금 상기 국부 이상 점수를 계산하는 동작을 수행하도록 하는 명령들은, 상기 프로세서로 하여금:

상기 윈도우를 이동시키면서, 상기 윈도우와 중첩되는 프레임 부분에서 상기 예측된 현재 프레임과 상기 실제 현재 프레임의 대응하는 픽셀들 간의 상기 화소 값 차이의 제곱을 평균하거나 합산하여 상기 국부 이상 점수를 계산하는 동작;을 수행하도록 하는 이상 상황 탐지 장치.

청구항 18

청구항 16에 있어서, 상기 프로세서로 하여금 상기 이상 점수를 결정하는 동작을 수행하도록 하는 명령들은, 상기 프로세서로 하여금:

상기 윈도우의 이동 위치들에 상응한 상기 국부 이상 점수들 중에서 크기순으로 정해지는 소정 개수의 점수들만을 평균하여 상기 이상 점수를 결정하는 동작;을 수행하도록 하는 이상 상황 탐지 장치.

청구항 19

청구항 16에 있어서, 상기 윈도우는 상기 실제 현재 프레임에 대하여 위로 이동할수록 상기 윈도우의 크기가 작아지도록 설정되는 이상 상황 탐지 장치.

청구항 20

청구항 11에 있어서, 상기 프로그램 명령들은 상기 프로세서에 의해 실행될 때 상기 프로세서로 하여금:

상기 예측된 현재 프레임을 생성하는 동작을 수행하기 이전에, 상기 일련의 영상을 전처리하여 흑백 영상으로의 변환 및 영상 해상도 조정 중 적어도 하나를 포함하는 전처리를 수행하는 동작;을 수행하도록 하는 이상 상황 탐지 장치.

발명의 설명

기술 분야

[0001] 본 발명은 영상 감시 시스템에 관한 것으로서, 특히, 획득한 영상을 토대로 이상 상황을 실시간으로 탐지하는 방법 및 장치에 관한 것이다.

배경 기술

[0002] 영상 감시 시스템의 숫자와 규모가 증가함에 따라, 감시 카메라에 의해 획득한 감시 영상을 토대로 감시 대상 지역 내의 이상 상황을 탐지하기 위해 투입되는 시간과 비용도 증가한다. 감시 영상 내의 움직임 객체를 자동으로 검출하여 이상 상황을 탐지하는 프로그램도 널리 활용되고 있지만, 오작동되는 경우가 많아 탐지 결과를 검증하기 위한 인력이 필요할 수 있다.

[0003] 탐지 정확도를 높이기 위하여, 최근에는 탐지 프로그램에 딥러닝을 채택하는 경우도 증가하고 있다. 그렇지만,

딥러닝을 채택하는 영상 감시 시스템의 경우 탐지 속도가 낮은 경우가 많아서, 감시 대상 지역 내에 강도, 폭력, 살인 등 위험한 상황이 발생했을 때 해당 상황에 신속하게 대응하는데 지장을 줄 수 있다.

발명의 내용

해결하려는 과제

[0004] 본 발명은 이와 같은 문제를 해결하기 위한 것으로서, 영상 감시 시스템에서 감시 영상을 토대로 이상 상황을 정확하게 탐지할 수 있을 뿐만 아니라 신속하게 탐지할 수 있는 방법 및 장치를 제공하는 것을 기술적 과제로 한다.

과제의 해결 수단

[0005] 상기 기술적 과제를 달성하기 위한 본 발명의 이상 상황 탐지 방법은 시간적으로 연이어진 일련의 영상을 받아들이고 상기 영상 내에 이상 상황이 포함되어 있는지 탐지하기 위한 것으로서, 현재 프레임보다 시간적으로 앞선 이전 프레임과 상기 현재 프레임보다 시간적으로 뒤진 이후 프레임을 토대로 현재 프레임을 예측하여, 예측된 현재 프레임을 생성하는 단계; 상기 예측된 현재 프레임과 실제 현재 프레임에 대한 차이를 나타내는 이상 점수를 산출하는 단계; 및 상기 이상 점수가 소정의 조건을 충족하는 경우 상기 현재 프레임에 이상 상황이 포함되어 있는 것으로 결정하는 단계;를 포함한다.

[0006] 상기 이전 프레임은 상기 현재 프레임보다 복수의 프레임만큼 시간적으로 앞선 프레임이고, 상기 이후 프레임은 상기 현재 프레임보다 복수의 프레임만큼 시간적으로 뒤진 프레임일 수 있다. 특히, 상기 이전 프레임은 두 번째 이전 프레임이고, 상기 이후 프레임은 두 번째 이후 프레임일 수 있다.

[0007] 상기 예측된 현재 프레임을 생성하는 단계는 상기 이전 프레임을 입력으로 받아들이는 제1 서버 네트워크와 상기 이후 프레임을 입력으로 받아들이는 제2 서버 네트워크를 포함하는 인공지능망에 의해 상기 예측된 현재 프레임을 도출할 수 있다. 상기 인공지능망에서 상기 제1 및 상기 제2 서버 네트워크 각각은 복수의 컨볼루션 레이어와 적어도 하나의 맥스풀링 레이어를 갖는 경로를 구비할 수 있다. 또한, 상기 제1 및 상기 제2 서버 네트워크는 상대측 서버 네트워크의 피쳐 맵과 적어도 한번 결합되도록 구성될 수 있다.

[0008] 상기 인공지능망은 정상적인 상황에서 상기 이전 프레임과 상기 이후 프레임을 토대로 상기 예측된 현재 프레임을 예측할 수 있게 사전에 훈련된 후 사용될 수 있다.

[0009] 상기 이상 점수를 산출하는 단계는, 상기 실제 현재 프레임에 대하여 소정의 윈도우를 사전에 정해진 이동간격 단위로 수평 및 수직 방향으로 이동시키면서, 상기 윈도우와 중첩되는 프레임 부분에서 상기 예측된 현재 프레임과 상기 실제 현재 프레임의 대응하는 픽셀들 간의 화소 값 차이에 대하여 소정의 연산을 행함으로써 국부 이상 점수를 계산하는 단계; 및 상기 윈도우의 이동 위치들에 상응하여 계산되는 복수의 국부 이상 점수들을 평균하거나 합산하여 상기 이상 점수를 결정하는 단계;를 포함할 수 있다.

[0010] 상기 국부 이상 점수를 계산함에 있어서는, 상기 윈도우를 이동시키면서, 상기 윈도우와 중첩되는 프레임 부분에서 상기 예측된 현재 프레임과 상기 실제 현재 프레임의 대응하는 픽셀들 간의 상기 화소 값 차이의 제곱을 평균하거나 합산하여 상기 국부 이상 점수를 계산할 수 있다.

[0011] 상기 이상 점수를 결정함에 있어서는, 상기 윈도우의 이동 위치들에 상응한 상기 국부 이상 점수들 중에서 크기 순으로 정해지는 소정 개수의 점수들만을 평균하여 상기 이상 점수를 결정할 수 있다.

[0012] 상기 윈도우는, 상기 실제 현재 프레임에 대하여 위로 이동할수록 상기 윈도우의 크기가 작아지도록, 설정될 수 있다.

[0013] 상기 예측된 현재 프레임을 생성하는 단계를 수행하기 이전에, 상기 일련의 영상을 전처리하여 흑백 영상으로의 변환 및 영상 해상도 조정 중 적어도 하나를 포함하는 전처리가 수행될 수 있다.

[0014] 본 발명에 의한 이상 상황 탐지 장치는 시간적으로 연이어진 일련의 영상을 받아들이고 상기 영상 내에 이상 상황이 포함되어 있는지 탐지하는 장치로서, 프로그램 명령들을 저장하는 메모리와; 상기 메모리에 통신가능하게 접속되고 상기 메모리에 저장된 상기 프로그램 명령들을 실행하는 프로세서;를 구비한다. 상기 프로그램 명령들은 상기 프로세서에 의해 실행될 때 상기 프로세서로 하여금: 현재 프레임보다 시간적으로 앞선 이전 프레임과 상기 현재 프레임보다 시간적으로 뒤진 이후 프레임을 토대로 현재 프레임을 예측하여, 예측된 현재 프레임을 생성하는 동작; 상기 예측된 현재 프레임과 실제 현재 프레임에 대한 차이를 나타내는 이상 점수를 산출하는

동작; 및 상기 이상 점수가 소정의 조건을 충족하는 경우 상기 현재 프레임에 이상 상황이 포함되어 있는 것으로 결정하는 동작;을 수행하도록 할 수 있다.

발명의 효과

- [0015] 본 발명의 일 실시예에 따르면, 영상 감시 시스템에서 감시 영상으로부터 이상 상황을 정확하게 탐지할 수 있을 뿐만 아니라 신속하게 탐지할 수 있다. 많은 연구에서 사용되는 데이터 세트들을 토대로 탐지 성능을 실제로 테스트한 결과, 본 발명에 의한 탐지 방법은 높은 정확도와 실시간 탐지 능력을 달성하는 것을 확인하였다.
- [0016] 특히, 이전 프레임과 이후 프레임을 서로 결합된 서브 네트워크들을 구비하는 인공신경망을 통해 처리하게 됨에 따라, 현재 프레임을 매우 잘 예측하게 되며, 예측된 현재 프레임과 실제 현재 프레임의 대비를 통해 이상 상황의 정확한 탐지가 가능하게 된다.
- [0017] 윈도우를 이동시키면서 이상 점수를 산출함에 있어, 윈도우의 크기가 위로 갈수록 작아지도록 하는 Cascade sliding window 기법은 연산량을 증가시키지 않으면서 각 객체의 원근감을 고려한 프레임 대비가 가능하게 해주기 때문에, 이상 상황을 오류가 없이 신속하고 정확하게 탐지할 수 있게 해준다.

도면의 간단한 설명

- [0018] 도 1은 본 발명의 일 실시예에 따른 영상 감시 시스템의 전체적인 구성을 보여주는 개략적인 블록도이다.
- 도 2는 본 발명의 일 실시예에 따른 이상 상황 탐지 과정을 보여주는 흐름도이다.
- 도 3은 도 2에 도시된 인공신경망의 아키텍처를 상세하게 보여주는 도면이다.
- 도 4는 제곱오차 영상을 토대로 현재 프레임에 대한 이상 점수를 추론하는 과정을 보여주는 흐름도이다.
- 도 5는 도 4의 이상 점수 추론을 구현하기 위한 의사코드(pseudo-code)를 보여준다.
- 도 6은 윈도우의 슬라이딩 및 크기 변형 방식을 보여주는 도면이다.
- 도 7은 본 발명의 일 실시예에 따른 이상 상황 탐지 장치의 블록도이다.
- 도 8은 윈도우 크기 감소단위(d)에 따른 각 데이터 세트의 AUC 값을 정리한 표이다.
- 도 9는 각 데이터 세트에 대한 본 발명에 의한 이상 상황 탐지 방법과 다른 최신 탐지 방법의 프레임 레벨 AUC를 비교한 표이다.
- 도 10은 본 발명에 의한 이상 상황 탐지 방법의 각 데이터 세트에 대한 단계별 프레임당 처리 속도를 정리한 표이다.
- 도 11은 본 발명에 의한 이상 상황 탐지 방법과 다른 기법의 프레임당 처리 속도를 비교한 표이다.
- 도 12a 내지 도 12c는 이상 상황을 포함한 영상에서 실제 프레임, 예측된 프레임, 두 프레임의 제곱오차 영상 프레임의 예들을 보여주는 도면들이다.
- 도 13은 에블레이션 연구(ablation study)에 관한 결과를 보여주는 표이다.

발명을 실시하기 위한 구체적인 내용

- [0019] 본 발명은 다양한 변형을 가할 수 있고 여러 가지 실시예를 가질 수 있는 바, 특정 실시예들을 도면에 예시하고 상세한 설명에 상세하게 설명하고자 한다. 그러나, 이는 본 발명을 특정한 실시 형태에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변형, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다. 각 도면을 설명하면서 유사한 참조부호를 유사한 구성요소에 대해 사용하였다.
- [0020] 제1, 제2, A, B 등의 용어는 다양한 구성요소들을 설명하는 데 사용될 수 있지만, 상기 구성요소들은 상기 용어들에 의해 한정되어서는 안 된다. 상기 용어들은 하나의 구성요소를 다른 구성요소로부터 구별하는 목적으로만 사용된다. 예를 들어, 본 발명의 권리 범위를 벗어나지 않으면서 제1 구성요소는 제2 구성요소로 명명될 수 있고, 유사하게 제2 구성요소도 제1 구성요소로 명명될 수 있다. "및/또는"이라는 용어는 복수의 관련된 기재된 항목들의 조합 또는 복수의 관련된 기재된 항목들 중의 어느 항목을 포함한다.
- [0021] 어떤 구성요소가 다른 구성요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성요소에

직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성요소가 다른 구성요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는, 중간에 다른 구성요소가 존재하지 않는 것으로 이해되어야 할 것이다.

[0022] 본 출원에서 사용한 용어는 단지 특정한 실시예를 설명하기 위해 사용된 것으로, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 출원에서, "포함하다" 또는 "가지다" 등의 용어는 명세서상에 기재된 특징, 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.

[0023] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가지고 있다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥 상 가지는 의미와 일치하는 의미를 가지는 것으로 해석되어야 하며, 본 출원에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.

[0024] 이하, 본 발명에 따른 바람직한 실시예를 첨부된 도면을 참조하여 상세하게 설명한다. 본 발명을 설명함에 있어 전체적인 이해를 용이하게 하기 위하여 도면상의 동일한 구성요소에 대해서는 동일한 참조부호를 사용하고 동일한 구성요소에 대해서 중복된 설명은 생략한다.

[0025] 도 1은 본 발명의 일 실시예에 따른 영상 감시 시스템의 전체적인 구성을 보여주는 개략적인 블록도이다. 도시된 영상 감시 시스템은 감시대상 지역에 분산설치되어 각각이 감시대상 지역 중 일부에 대한 감시 영상을 획득하는 복수의 감시 카메라(10a~10n)와, 상기 복수의 감시 카메라(10a~10n)가 접속할 수 있는 감시제어 서버(20)를 구비한다. 일 실시예에 있어서, 감시 카메라들(10a~10n)은 유선망 및/또는 무선망을 기반으로 한 IP 네트워크를 통해서 감시제어 서버(20)에 접속된다.

[0026] 각 감시 카메라(10a~10n)는 감시제어 서버(20)의 제어신호에 응답하여 수평회전(panning) 및 수직회전(tilting)하면서 적절한 줌 배율로 주변 지역을 촬영한다. 감시제어 서버(20)는 각 감시 카메라(10a~10n)로부터의 영상에서 움직임객체를 검출하거나 이상 상황을 탐지하고, 필요에 따라 감시 카메라(10a~10n)의 팬, 틸트, 줌을 제어할 수 있다. 또한, 감시제어 서버(20)는 감시 카메라(10a~10n)로부터 획득한 영상에서 이상 상황이 탐지된 경우 경보를 발령할 수 있다. 청구범위를 포함하여 본 명세서에서, "이상 상황"이란 예컨대 강도, 폭력, 살인 등 위험한 상황을 일컫는데, 이상 상황의 종류가 이에 한정되는 것은 아니다.

[0027] 영상데이터 처리, 움직임객체의 검출, 이상 상황 탐지, 영상 디스플레이, 경보 발령 등의 기능을 수행하는 감시제어 서버(20)는 후술하는 바와 같이 적어도 하나의 프로세서, 특히 그래픽 프로세서(GPU)를 구비하며, 상기 적어도 하나의 프로세서에 의한 프로그램 코드의 실행을 통해 이상 상황 탐지 동작을 수행하는 이상 상황 탐지 장치로 작용한다. 그렇지만, 변형된 실시예에 있어서는, 감시제어 서버(20)가 아닌 개별 감시 카메라(10a~10n)가 영상데이터 처리, 움직임객체의 검출, 이상 상황 탐지 등의 동작을 수행하는 탐지 장치로 작용할 수 있고, 각 감시 카메라(10a~10n)가 검출 내지 탐지 결과를 감시제어 서버(20)에 전송하게 될 수도 있다.

[0028] 도 2는 본 발명의 일 실시예에 따른 이상 상황 탐지 과정을 보여주는 흐름도이다. 본 실시예에 따른 이상 상황 탐지 과정은 영상에 대한 전처리 단계(S100), 인공지능망에 의한 현재 프레임 예측 단계(S110), 예측된 현재 프레임과 실제 현재 프레임 간의 차 영상을 계산하는 단계(S120), 상기 차 영상의 화소 값 분포를 토대로 이상 점수를 계산하는 단계(S130), 및 상기 이상 점수를 토대로 해당 프레임 내에 이상 상황이 발생하였는지 결정하는 단계(S140)를 포함한다.

[0029] 먼저 상기 전처리 단계(S100)에서, 영상 프레임은 칼라 영상에서 흑백 영상으로 변환된다. 아울러, 영상 프레임의 크기가 일정한 크기, 예컨대 $256 \times 256 \times 1$ 로 변환될 수 있는데, 여기서 256×256 은 가로 및 세로 방향의 해상도를 나타내고, 1은 채널 수를 나타낸다.

[0030] 전처리가 완료된 영상 프레임은 프레임 메모리(미도시)에 저장될 수 있다. 일 실시예에 있어서, 프레임 메모리는 현재 프레임(F_t)을 기준으로 적어도 두 번째 이전 프레임(F_{t-2})부터 두 번째 이후 프레임(F_{t+2})까지 최소한 5개의 프레임을 저장할 수 있다.

[0031] 인공지능망에 의한 현재 프레임 예측 단계(S110)에서는, 현재 프레임보다 이전 프레임과 이후 프레임의 영상을 입력으로 받아들이고, 본 발명에 의한 크로스 유-넷(Cross U-Net) 인공지능망에 의해 현재 프레임을 예측한다.

일 실시예에서, 인공신경망에서 입력으로 사용하는 영상은 현재 프레임(F_t)의 두 번째 이전 프레임(F_{t-2})과 두 번째 이후 프레임(F_{t+2})이다. 인공신경망은 이들 영상 프레임들을 토대로 현재 프레임을 예측하고 예측된 현재 프레임($\hat{F}_t$)을 출력한다.

[0032] 실제 이상 상황 탐지에 적용하기 이전에, 인공신경망은 이상 상황이 아닌 정상 상황에서 상기 이전 프레임과 이후 프레임을 사용하여 현재 프레임을 정확히 예측할 수 있도록 훈련된다. 따라서 훈련 이후의 인공신경망은 비정상적인 상황에서보다 정상적인 상황에서 현재 프레임을 잘 예측하게 되어 예측된 현재 프레임과 실제 현재 프레임 사이에 별 차이가 없게 되며, 본 발명은 이 차이를 이용해서 이상 상황을 탐지한다. 즉, 본 발명의 신경망에 의해 정해지는 예측된 현재 프레임과 실제 현재 프레임 사이에 일정 정도 이상의 차이가 존재하는 경우, 이상 상황 탐지 장치는 현재 프레임에 이상 상황이 발생한 것으로 판단하게 된다.

[0033] 제120단계에서는, 예측된 현재 프레임($\hat{F}_t$)과 실제 현재 프레임(F_t) 간의 차 영상을 계산한다. 청구범위를 포함하여 본 명세서에 있어서, "차 영상"이란 두 영상 프레임에 대하여 픽셀 단위로 화소 값을 차감하여 생성되는 숫자들의 배열을 말한다. 이와 같은 차 영상은 도 2에 예시한 바와 같이 시각적으로 인식될 수 있게 화면에 표시될 수도 있지만, 반드시 이에 한정되는 것은 아니며, 이후의 계산에 활용될 수 있게 프레임 메모리 또는 RAM과 같은 통상적인 저장장치에 저장될 수 있는 데이터면 족하다. 한편, "차 영상"의 각 화소 값은 두 영상 프레임의 대응 픽셀들의 화소 값의 단순한 차이가 아니라, 화소 값 차이의 절대값 또는 제곱한 값일 수도 있다.

아래에서 상술하는 예시적인 실시예에 있어서, "차 영상"은 각 픽셀의 화소 값이 두 영상 프레임($\hat{F}_t, F_t$)의 대응 픽셀들의 화소 값 차이의 제곱 즉, 제곱오차(squared error)인 제곱오차 영상이다. 이하, 상기 제곱오차 영상은 $(\hat{F}_t - F_t)^2$ 의 기호로 나타낼 수 있다.

[0034] 제130단계에서는, 상기 차 영상의 화소 값 분포를 토대로 현재 프레임(F_t) 내에 이상 상황이 포함되어 있을 가능성을 나타내는 지표인 이상 점수를 계산한다. 일 실시예에 있어서는, 이상 점수를 구하기 위해서, 소정의 크기를 가지는 윈도우를 영상 프레임 즉, 현재 프레임 또는 제곱오차 영상 프레임에 대하여 수평 및 수직 방향으로 일정한 이동간격(stride) 단위로 슬라이딩시켜 상대적으로 이동시키면서 국부 이상 점수(local anomaly score)를 계산할 수 있다. 국부 이상 점수는 각 슬라이딩 스텝마다 윈도우와 중첩되는 영상 프레임 부분의 화소들에 대한 제곱오차의 평균치로서 계산될 수 있다. 이어서, 영상 프레임 전 영역에 걸쳐 계산한 국부 이상 점수들 중에서 적어도 일부의 평균치를 영상 프레임 전체에 대한 이상 점수(S)로 결정할 수 있다. 국부 이상 점수 중 일부만을 사용하여 이상 점수(S)로 결정하는 경우, 크기가 큰 국부 이상 점수 값들을 일정한 개수만큼 선택하고 이 값들을 평균하여 이상 점수(S)를 추론할 수 있다.

[0035] 이상 점수(S)를 이러한 방식으로 결정하는 경우, 이상 점수(S)의 크기가 클수록 현재 프레임에서 이상 상황을 포함할 확률이 높다는 것을 의미할 수 있다. 따라서, 제140단계에서는 상기 이상 점수(S)가 소정의 임계치보다 큰 경우 현재 프레임에 이상 상황이 발생한 것으로 판단할 수 있다.

[0036] 도 3은 도 2에 도시된 인공신경망의 아키텍처를 상세하게 보여준다.

[0037] 도시된 크로스 유-넷(Cross U-Net) 인공신경망은 본 발명이 새로이 제시하는 인공신경망 모델로서, U-Net을 기반으로 하는 두 개의 서브 네트워크를 구비한다. 상기 두 개의 서브 네트워크 중에서 도면에서 상측에 있는 제1 서브 네트워크는 현재 프레임에서 두 번째 이전 프레임(F_{t-2})을 입력으로 받아들이고, 도면에서 하측에 있는 제2 서브 네트워크는 현재 프레임에서 두 번째 이후 프레임(F_{t+2})을 입력으로 받아들이는다. 인공신경망은 상기 두 번째 이전 프레임(F_{t-2})과 상기 두 번째 이후 프레임(F_{t+2})으로부터 현재 프레임을 예측한다. 위에서 언급한 바와 같이, 상기 두 번째 이전 프레임(F_{t-2})과 상기 두 번째 이후 프레임(F_{t+2})은 전처리 과정을 통해 흑백으로 변환되고 크기가 $256 \times 256 \times 1$ 로 변환된 영상일 수 있는데, 본 발명이 이에 한정되는 것은 아니다.

[0038] 각 서브 네트워크는 수축 경로(contracting path)와 확장 경로(expansive path)를 구비한다.

[0039] 수축 경로는 두 번의 3×3 컨볼루션 레이어(3×3 conv), 한 번의 2×2 맥스풀링(max-pooling) 레이어, 상응하는

다른 서브 네트워크의 피쳐 맵(feature map)과의 결합(concatenation)을 반복한다. 이와 같은 서브 네트워크 간의 결합은 본 발명에 의한 신경망이 현재 프레임을 더 잘 예측하게 해주며, 본 발명에 의한 신경망의 고유의 특징이라 할 수 있다. 수축 경로에서 드롭아웃(dropout)은 마지막 2×2 맥스풀링 레이어 전에 사용된다.

[0040] 확장 경로는 2×2 컨볼루션 레이어와 2×2 업샘플링(upsampling) 레이어(2×2 up-conv), 상응하는 수축 경로의 피쳐 맵과의 결합, 두 번의 3×3 컨볼루션 레이어가 반복된다. 확장 경로에서 드롭아웃은 처음 2×2 업샘플링 레이어 전에 사용된다.

[0041] 각 서브 네트워크의 마지막 레이어는 3×3 컨볼루션 레이어를 사용해 두 개의 채널을 가진 피쳐 맵을 만든다. 이 피쳐 맵은 다른 서브 네트워크에서 상응하는 피쳐 맵과 결합한 후 1×1 컨볼루션 레이어(1×1 conv)로 한 개의 채널을 가진 피쳐 맵을 만든다. 이 피쳐 맵이 예측된 현재 프레임을 나타낸다.

[0042] 이와 같은 신경망 모델에서 모든 컨볼루션 레이어는 활성화 함수로 ReLU(rectified linear unit)를 사용할 수 있다. 손실함수는 다음 수학적 식 1로 표시되는 픽셀 단위 평균제곱오차(pixel-wise mean squared error)가 사용될 수 있다.

수학적 식 1

$$L(\hat{F}, F) = \frac{1}{h \cdot w} \sum_{i=1}^h \sum_{j=1}^w (\hat{F}_{ij} - F_{ij})^2$$

[0043] 여기서, h는 프레임의 가로방향 크기 즉, 가로방향 해상도를 나타내고, w는 프레임의 세로방향 크기 즉, 세로방향 해상도를 나타낸다. F_{ij} 는 실제 현재 프레임에서의 (i,j) 위치에 있는 픽셀의 화소 값을 나타내고, $\hat{F}_{ij}$ 는

예측된 현재 프레임에서의 (i,j) 위치에 있는 픽셀의 화소 값을 나타낸다. $(\hat{F}_{ij} - F_{ij})^2$ 는 제곱오차 영상에서 해당 픽셀에 대한 제곱오차 값을 사용하면 되고, 중복하여 계산할 필요가 없다.

[0045] 위에서 언급한 바와 같이, 신경망은 이상 상황 탐지에 실제로 적용되기 이전에, 이상 상황이 아닌 정상 상황에서 두 번째 이전 프레임(F_{t-2})과 두 번째 이후 프레임(F_{t+2})으로부터 현재 프레임($\hat{F}_t$)을 정확히 예측할 수 있도록 훈련되며, 훈련 결과는 각 컨볼루션 연산에 사용되는 필터들 즉, 커널들에 반영된다. 이에 따라 훈련 이후의 신경망은 현재 프레임을 잘 예측하게 되어 예측된 현재 프레임($\hat{F}_t$)과 실제 현재 프레임(F_t) 사이에 큰 차이가 없게 되며, 본 발명은 이러한 특성을 이용해서 이상 상황을 탐지한다. 즉, 본 발명의 신경망에 의해 정해지는 예측된 현재 프레임($\hat{F}_t$)과 실제 현재 프레임(F_t) 사이에 일정 정도 이상의 차이가 존재하는 경우, 이상 상황 탐지 장치는 현재 프레임(F_t)에 이상 상황이 발생한 것으로 판단하게 된다.

[0046] 도 4는 제곱오차 영상을 토대로 현재 프레임에 대한 이상 점수를 추론하는 과정을 보여주는 흐름도이고, 도 5는 도 4의 이상 점수 추론을 구현하기 위한 의사코드(pseudo-code)를 보여준다. 도 6은 윈도우의 슬라이딩 및 크기 변형 방식을 보여주는 도면이다.

[0047] 위에서 언급한 바와 같이, 일 실시예에 따르면, 윈도우를 영상 프레임에 대하여 수평 및 수직 방향으로 일정한 이동간격(stride) 단위로 이동시키면서 국부 이상 점수(local anomaly score)를 계산할 수 있다. 국부 이상 점수는 윈도우와 중첩되는 영상 프레임 부분의 화소들에 대한 제곱오차의 평균치로서 계산될 수 있다. 그리고, 영상 프레임 전 영역에 걸쳐 계산한 국부 이상 점수들 중에서 적어도 일부의 평균치를 영상 프레임 전체에 대한 이상 점수로 결정할 수 있다.

[0048] 먼저, 이상 상황 탐지 장치는 제곱오차 영상이 준비되어 있는지 즉, 영상 프레임의 모든 픽셀에 대해서 예측된 현재 프레임($\hat{F}_t$)의 화소 값과 실제 현재 프레임(F_t)의 화소 값의 차이인 제곱오차(F_{ij})가 계산되어 있는지 확인할 수 있다(제300단계). 한편, 변형된 실시예에서는 제300단계의 확인 작업이 제300단계 이후에 수행될 수도 있다. 또 다른 실시예에 있어서는, 도 2에 도시된 제곱오차 영상 생성 단계(S120)가 생략되고, 도 4의 이상 점수 추론 과정에서 제302단계 이후에 필요할 때마다 제곱오차의 연산이 수행될 수도 있다.

[0049] 이어서, 윈도우의 좌하측 모서리 좌표인 (x, y) 를 $(0, 0)$ 으로 초기화한다. 이에 따라 윈도우는 제곱오차 영상에서 좌측 하단에 위치하게 된다(제302단계).

[0050] 제304단계에서, 이상 상황 탐지 장치는, 다음 수학식 2에 의해 표현되는 바와 같이, 윈도우와 중첩되는 영역 내에 있는 픽셀들에 대한 제곱오차의 평균을 계산함으로써 국부 이상 점수(p_k)를 구한다. 즉, 국부 이상 점수(p_k)는 윈도우에 상응한 제곱오차 영상 패치 내에 있는 화소 값들의 평균이라고 할 수 있다.

수학식 2

$$p_k = \frac{1}{s^2} \sum_{i=x}^{x+s} \sum_{j=y}^{y+s} I_{ij}$$

[0051]

[0052] 수학식 2와 도 5에서, s 는 프레임의 넓이와 높이를 나타내고 $\hat{s}$ 는 윈도우의 넓이와 높이를 나타낸다. 프레임의 넓이와 높이가 서로 동일하고 윈도우의 넓이와 높이가 서로 동일하다고 가정하였지만, 본 발명이 이에 한정되는 것은 아니며, 프레임의 넓이와 높이는 서로 다를 수 있고 윈도우의 넓이와 높이도 서로 다를 수 있다.

[0053] 이어서, 윈도우를 사전에 정해진 수평 이동간격만큼 오른쪽으로 이동시킨다(제306단계). 상기 수평 이동간격은 윈도우의 넓이와 동일한 크기를 가질 수 있다. 윈도우의 이동 후에는, 이동한 윈도우에 대한 국부 이상 점수(p_k)를 계산한다. 윈도우의 이동과 국부 이상 점수의 계산은 윈도우가 프레임의 오른쪽 모서리에 도달할 때까지 계속될 수 있다.

[0054] 윈도우가 프레임의 오른쪽 모서리에 도달하면(제308단계), 윈도우를 다시 왼쪽 끝으로 이동시키고, 사전에 정해진 수직 이동간격만큼 위로 이동시킨다(제310단계). 상기 수직 이동간격은 윈도우의 높이와 동일한 크기를 가질 수 있다. 이와 같이, 윈도우를 위로 수직 이동시킨 후에는, 윈도우의 크기를 일정한 감소단위(d)만큼 감소시킬 수 있다(제312단계). 윈도우 크기의 감소는 높이에 대해서만 행해질 수도 있고, 높이와 아울러 넓이에 대해서 행해질 수도 있다. 윈도우 크기의 감소는 영상 내에서 객체들의 원근 차이로 인하여 객체가 카메라로부터 멀어질수록 즉, 프레임 내에서 위로 갈수록 크기가 작아지는 현상을 고려한 것이다. 이에 따라, 본 발명에 따르면 윈도우가 프레임 내에서 위로 갈수록 점점 작아지게 된다. 이처럼 윈도우의 크기가 위로 갈수록 작아진다는 점에서, 도 4 및 도 5에 도시된 바와 같이 윈도우를 이동시키면서 이상 점수를 산출하는 기법은 Cascade sliding window 기법이라고 칭해질 수 있다.

[0055] 제310단계에서 윈도우를 이동시킨 후에는, 이동한 윈도우에 대한 국부 이상 점수(p_k)의 계산과 윈도우의 수평 이동이 계속된다(제304단계, 제306단계).

[0056] 그리고, 윈도우를 오른쪽으로 그리고 위쪽으로 이동시키면서 국부 이상 점수(p_k)를 계산하는 과정(제304단계 내지 제312단계)은 윈도우가 프레임의 상단 모서리에 도달할 때까지 계속될 수 있다.

[0057] 윈도우가 프레임의 상단 모서리에 도달하면(제314단계), 각 윈도우 위치에 대하여 계산된 국부 이상 점수(p_k) 중에서 적어도 일부의 국부 이상 점수(p_k)에 대하여 수학식 3에 의해 평균치를 계산하여 해당 프레임에 대한 이상 점수(S)로 결정할 수 있다(제316단계). 이때, 모든 국부 이상 점수(p_k)에 대한 평균을 프레임에 대한 이상 점수(S)로 결정할 수도 있지만, 일부의 국부 이상 점수(p_k)에 대한 평균을 프레임에 대한 이상 점수(S)로 결정할 수도 있다. 예컨대, 모든 국부 이상 점수(p_k)를 오름차순 또는 내림차순으로 정렬한 후, 일정한 개수(n 개)의 국부 이상 점수(p_k)에 대해서만 평균을 계산하여 이상 점수(S)로 결정할 수 있다. 이상 점수가 높을수록 현재 프레임에 이상 상황이 포함되어 있을 확률이 높다는 것을 의미할 수 있다.

수학식 3

$$S = \frac{1}{n} \sum_{i=1}^n p_i$$

[0058]

[0059]

한편, 제306단계에서 윈도우를 수평 이동간격만큼 오른쪽으로 이동시킬 때, 윈도우의 오른쪽에 이동할 공간이 부족하여 윈도우(410m)의 맨오른쪽 열의 좌표가 프레임(400)의 맨오른쪽 열의 좌표보다 크게 된다면, 윈도우의 x좌표를 $s - \hat{s}$ 만큼 왼쪽으로 이동하여, 윈도우의 맨오른쪽 열이 프레임의 맨오른쪽 열과 일치되도록 윈도우(410m')를 이동시킨 후 국부 이상 점수(p_k)를 계산한다(도 5의 스텝 8~10과, 도 6 참조).

[0060]

마찬가지로, 제310단계에서 윈도우를 수직 이동간격만큼 위로 이동시킬 때, 윈도우의 위쪽으로 이동할 공간이 부족하여 윈도우(410p)의 최상단 행의 좌표가 프레임(400)의 최상단 행의 좌표보다 크게 된다면, 윈도우의 y좌표를 $s - \hat{s}$ 만큼 아래쪽으로 이동하여, 윈도우의 최상단 행이 프레임의 최상단 행과 일치되도록 윈도우(410p')를 이동시킨 후 국부 이상 점수(p_k)를 계산한다(도 5의 스텝 17~19과, 도 6 참조).

[0061]

도 5에 예시된 실시예에서는 프레임의 넓이와 높이가 's'로 동일한 것으로 가정하였지만, 프레임의 넓이(w)와 높이(h)는 서로 다를 수 있다. 또한, 윈도우의 넓이와 높이가 ' $\hat{s}$ '로 동일한 것으로 가정하였지만, 프레임의 넓이($\hat{w}$)와 높이($\hat{h}$)는 서로 다를 수 있다. 윈도우의 크기 감소단위(d)도 좌우방향 및 상하 방향에 대하여 다르게 설정될 수 있다.

[0062]

도 7은 본 발명의 일 실시예에 따른 이상 상황 탐지 장치의 블록도이다. 본 발명의 일 실시예에 따른 이상 상황 탐지 장치는 적어도 하나의 프로세서(520), 메모리(540), 및 저장 장치(560)를 포함할 수 있다. 앞에서 언급한 바와 같이, 이상 상황 탐지 장치는 도 1에 도시된 영상 감시 시스템에서 감시제어 서버(20)에 의해 구현될 수 있다. 이러한 경우, 감시제어 서버(20)는 각 감시 카메라(10a~10n)가 획득한 감시 영상을 해당 카메라로부터 받아들이고, 수신된 감시 영상에서 이상 상황을 탐지한다. 그렇지만, 본 발명의 다른 실시예에서는, 감시제어 서버(20)가 아닌 개별 감시 카메라(10a~10n)가 이상 상황 탐지 장치로 작용할 수 있고, 탐지 결과를 감시제어 서버(20)에 전송할 수 있다.

[0063]

프로세서(520)는 메모리(540) 및/또는 저장 장치(560)에 저장된 프로그램 명령을 실행할 수 있다. 프로세서(520)는 적어도 하나의 중앙 처리 장치(central processing unit, CPU)나 그래픽 처리 장치(graphics processing unit, GPU)에 의해 구현될 수 있으며, 그밖에 본 발명에 따른 방법을 수행할 수 있는 여타의 프로세서일 수 있다.

[0064]

메모리(540)는 예컨대 RAM(Random Access Memory)와 같은 휘발성 메모리와, ROM(Read Only Memory)과 같은 비휘발성 메모리를 포함할 수 있다. 메모리(540)는 저장 장치(560)에 저장된 프로그램 명령을 로드하여, 프로세서(520)에 제공함으로써 프로세서(520)가 이를 실행할 수 있도록 할 수 있다.

[0065]

저장 장치(560)는 프로그램 명령과 데이터를 저장하기에 적합한 기록매체로서, 예컨대 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(Magnetic Media), CD-ROM(Compact Disk Read Only Memory), DVD(Digital Video Disk)와 같은 광 기록 매체(Optical Media), 플롭티컬 디스크(Floptical Disk)와 같은 자기-광 매체(Magneto-Optical Media), 플래시 메모리나 EPROM(Erasable Programmable ROM) 또는 이들을 기반으로 제작되는 SSD와 같은 반도체 메모리를 포함할 수 있다.

[0066]

저장 장치(560)는 상기 프로그램 명령을 저장한다. 특히, 상기 프로그램 명령은 본 발명에 따른 이상 상황 탐지 방법을 구현하기 위한 이상 상황 탐지 프로그램을 포함할 수 있다. 상기 이상 상황 탐지 프로그램은 시간적으로 연이어진 일련의 영상을 받아들이고 상기 영상 내에 이상 상황이 포함되어 있는지 탐지하는 방법으로서, 현재 프레임보다 시간적으로 앞선 이전 프레임과 상기 현재 프레임보다 시간적으로 뒤진 이후 프레임을 토대로 현재 프레임을 예측하여, 예측된 현재 프레임을 생성하는 단계; 상기 예측된 현재 프레임과 실제 현재 프레임에 대한 차이를 나타내는 이상 점수를 산출하는 단계; 및 상기 이상 점수가 소정의 조건을 충족하는 경우 상기 현재 프레임에 이상 상황이 포함되어 있는 것으로 결정하는 단계;를 포함하는 이상 상황 탐지 방법을 구현하는데

필요한 프로그램 명령을 포함한다. 이와 같은 프로그램 명령은 프로세서(520)의 제어에 의해 메모리(540)에 로드된 상태에서, 프로세서(520)에 의해 실행되어 본 발명에 의한 방법을 구현할 수 있다.

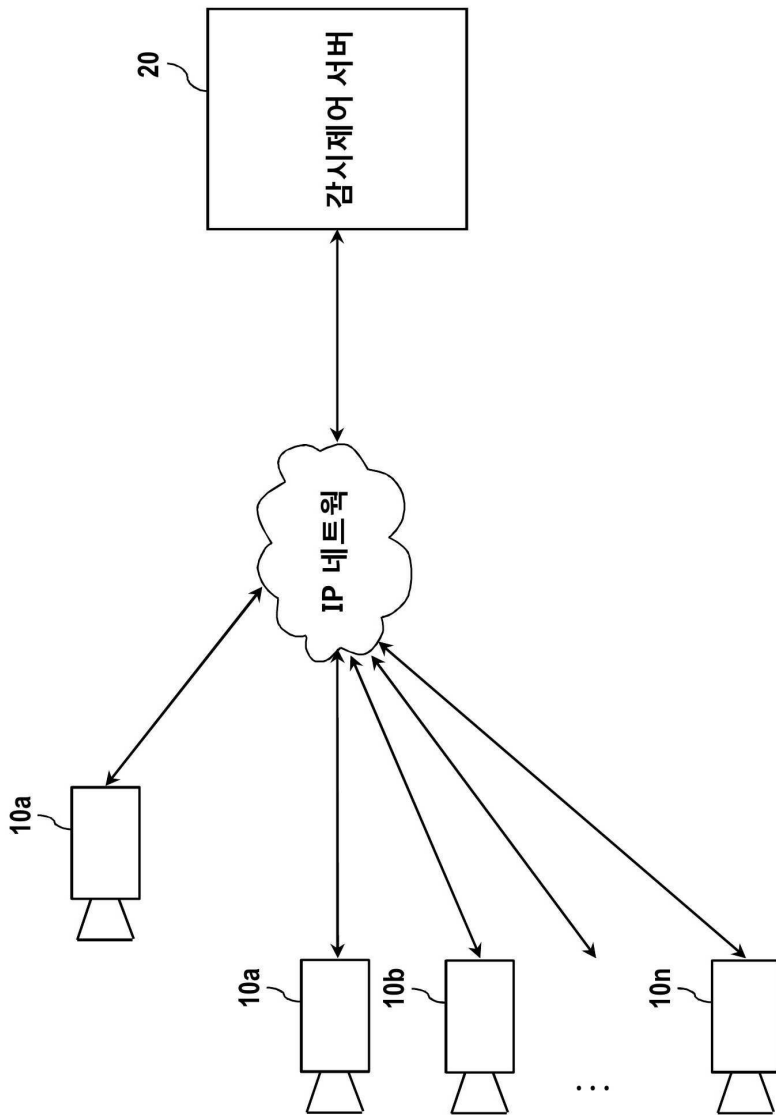
- [0067] 영상 분석 분야의 연구에서 일반적으로 사용되는 데이터 세트인 'CUHK Avenue', 'UCSD Ped2', 'ShanghaiTech Campus' 데이터 세트를 사용하여 본 발명에 의한 이상 상황 탐지의 정확도 및 속도를 테스트하였다. 각 데이터 세트의 특징은 다음과 같다.
- [0068] 'CUHK Avenue' 데이터 세트는 640×360 의 해상도를 가진 16개의 훈련 영상과 21개의 테스트 영상을 제공하며 프레임 레이트는 25 fps이다. 이 데이터 세트는 카메라로부터 객체가 멀어질수록 객체가 작아지는 특징이 있다.
- [0069] 'UCSD Ped2' 데이터 세트는 360×240 의 해상도를 가진 16개의 훈련 영상과 12개의 테스트 영상을 제공하며 프레임 레이트는 10 fps이다. 이 데이터 세트는 카메라로부터 객체가 멀어지더라도 객체의 크기가 비슷하다는 특징이 있다.
- [0070] 'ShanghaiTech Campus' 데이터 세트는 856×480 의 해상도를 가진 330개의 훈련 영상과 107개의 테스트 영상을 제공하며 프레임 레이트는 24 fps이다. 이 데이터 세트는 Avenue, Ped2 데이터 세트와 다르게 다양한 각도와 빛 조건을 가진 13개의 장소에서 촬영되었다는 특징이 있다.
- [0071] 본 발명자들은 각 데이터 세트에서 이상 상황 탐지 정확도와 속도를 측정하기 위해 각 데이터 세트의 훈련 영상으로 본 발명의 인공신경망(Cross U-Net) 모델을 학습시켰고 학습된 Cross U-Net 모델과 cascade sliding window 기법을 사용하여 테스트 영상의 프레임마다 이상 점수를 획득하였다. 그리고 프레임별 이상 점수와 실제 이상 상황 포함 여부를 토대로 수신자 조작 특성(ROC: Receiver operation characteristic)를 획득한 다음, ROC를 사용하여 frame-level AUC(Area Under the ROC Curve)를 획득하였다.
- [0073] 도 8은 윈도우 크기 감소단위(d)에 따른 각 데이터 세트의 AUC 값을 보여주는 표이다. Avenue 데이터 세트에서 90.77%, Ped2 데이터 세트에서 96.99%, ShanghaiTech에서 72.48%의 AUC 값을 획득하였다. Avenue 데이터 세트의 경우, 다른 두 데이터 세트와 달리 윈도우 크기 감소단위가 4일 때 가장 높은 AUC를 획득하였는데, 이는 카메라로부터 객체가 멀어질수록 객체가 작아지는 특징이 있는 Avenue 데이터 세트에서 cascade sliding window 기법이 효과적임을 보여준다.
- [0074] 도 9는 각 데이터 세트에 대한 본 발명에 의한 이상 상황 탐지 방법과 다른 최신 탐지 방법의 프레임 레벨 AUC를 비교한 표이다. Avenue 데이터 세트에서 본 발명에 의한 이상 상황 탐지 방법이 가장 높은 AUC를 획득한 것을 확인할 수 있으며, 다른 데이터 세트에서도 본 발명에 의한 이상 상황 탐지 방법이 비교적 높은 AUC를 획득한 것을 확인할 수 있다.
- [0075] 영상 감시 시스템에서 중요한 요소인 이상 상황 탐지 속도를 측정하기 위하여, 프레임당 처리 속도를 전처리 시간(preprocessing time), 현재 프레임 예측 시간(prediction time), 이상 상황 추론 시간(inference time)으로 구분하여 구하였다. 여기서, "전처리 시간(preprocessing time)"은 프레임이 Cross U-Net의 입력으로 사용되기 전에 영상을 이미지로 변환하는 시간, 컬러 이미지를 흑백 이미지로 변환하는 시간, 이미지의 크기를 $256 \times 256 \times 1$ 로 변환하는 시간을 포함하는 전처리 시간을 의미한다. "현재 프레임 예측 시간(prediction time)"은 Cross U-Net 모델이 이전 프레임과 이후 프레임을 사용해 현재 프레임을 예측하는 시간을 의미한다. "이상 상황 추론 시간(inference time)"은 cascade sliding window 기법을 사용하여 현재 프레임의 이상 점수를 추론하는 시간을 의미한다.
- [0076] 도 10은 데이터 세트마다 본 발명의 프레임당 처리 속도를 나타낸 것으로 NVIDIA TITAN RTX GDDR6 24GB with Intel(R) Core(TM) i9-10940X CPU @ 3.30GHz 환경에서 측정되었다. 도 10은 본 발명의 이상 상황 탐지 방법이 프레임당 Avenue 데이터 세트에 대해 31ms(약 32 fps), Ped2에 대해 33ms(약 33 fps), ShanghaiTech 데이터 세트에 대해 41ms(약 24fps)의 처리 속도를 가졌다는 것을 보여준다. 이는 본 발명의 이상 상황 탐지 방법이 frame rate가 25 fps인 Avenue 데이터 세트, 10 fps인 Ped2 데이터 세트, 24 fps인 ShanghaiTech 데이터 세트에서 실시간 탐지가 가능하다는 것을 나타낸다(NVIDIA TITAN RTX GDDR6 24GB with Intel(R) Core(TM) i9-10940X CPU @ 3.30GHz 환경 기준).
- [0077] Avenue 데이터 세트, Ped2 데이터 세트, 및 ShanghaiTech 데이터 세트에서 frame-level AUC가 본 발명의 방법보다 높은 Ionescu et al.의 기법과 프레임당 처리 속도를 비교했을 때 도 11에서 보는 바와 같이 본 발명의 방법이 Avenue 데이터 세트의 경우 4배, Ped2 데이터 세트의 경우 7배, ShanghaiTech 데이터 세트의 경우 3배 빠

른 것을 확인할 수 있다.

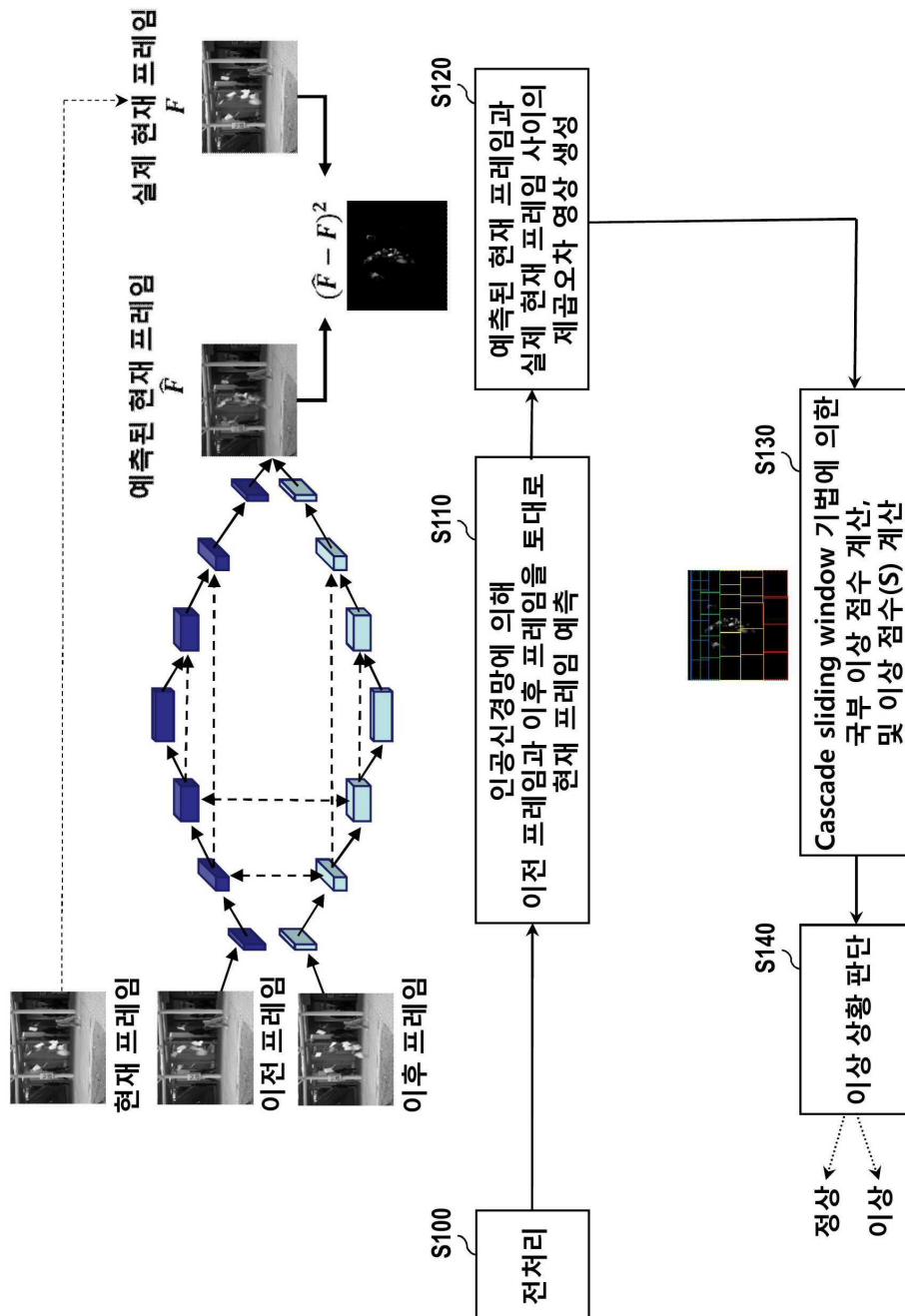
- [0078] 본 발명이 어떤 종류의 이상 상황을 잘 추론하고 어떤 이상 상황에 대하여 추론을 못하는지 분석하기 위하여, 이상 상황을 포함한 프레임들을 높은 이상 점수를 갖는 프레임과 낮은 이상 점수를 갖는 프레임으로 분류하였다. 그리고 각 프레임의 실제 프레임, 예측된 프레임, 실제 프레임과 예측된 프레임의 차이를 시각화하였다. 도 12a 내지 도 12c는 이상 상황을 포함한 영상에서 실제 프레임, 예측된 프레임, 두 프레임의 제곱오차의 시각화 영상 프레임의 예들을 보여준다. 구체적으로, 도 12a는 Avenue 데이터 세트에 대한 실제 프레임, 예측된 프레임, 제곱오차의 시각화 영상 프레임의 예들을 보여주고, 도 12b는 Ped2 데이터 세트에 대한 실제 프레임, 예측된 프레임, 제곱오차의 시각화 영상 프레임의 예들을 보여주고, 도 12c는 ShanghaiTech 데이터 세트에 대한 실제 프레임, 예측된 프레임, 제곱오차의 시각화 영상 프레임의 예들을 보여준다.
- [0079] 도 12a 내지 도 12c에서 보는 바와 같이, 본 발명에 의한 이상 상황 탐지 방법은 자전거, 자동차, 스케이트보드와 같은 탈 것; 춤을 추는 사람, 객체를 던지는 사람, 달리는 사람, 잘못된 방향으로 걷는 사람과 같은 이상 행동을 하는 사람들에 대해 높은 이상 점수를 나타내었다. 다만, 멈춰있거나 어둡거나 일부분만 보이거나 겹쳐있는 이상 객체에 대해서는 낮은 이상 점수를 나타내었다.
- [0080] 본 발명에 의한 인공지능망인 Cross U-Net 모델을 사용한 본 발명의 이상 상황 탐지 방법이 Cross U-Net 모델에서 결합 일부분을 제외한 기법보다 더 나은 성능을 보여주는지 확인하기 위해 애블레이션 연구(ablation study)를 진행하였다. Cross U-Net 모델에서 결합 부분이 없는 'No Concatenation', 하나의 서브 네트워크에 있는 수축 경로의 레이어와 다른 서브 네트워크에 있는 수축 경로에서 대응하는 레이어 결합 부분을 포함하는 'CC', 각 서브 네트워크마다 수축 경로의 레이어와 확장 경로에서 대응하는 레이어 결합 부분을 포함하는 'CE'의 frame-level AUC를 구하였다. 도 13은 ablation study에 관한 결과를 보여준다. 모델로부터 제외하는 부분이 없이 Cross U-Net 모델을 온전히 사용하는 경우에 가장 좋은 성능을 시현하게 됨을 보여준다.
- [0081] 위에서 언급한 바와 같이 본 발명의 실시예에 따른 장치와 방법은 이용자 스마트폰 또는 판정엔진 학습 서버에서 컴퓨터로 읽을 수 있는 기록매체에 컴퓨터가 읽을 수 있는 프로그램 또는 코드로서 구현하는 것이 가능하다. 컴퓨터가 읽을 수 있는 기록매체는 컴퓨터 시스템에 의해 읽혀질 수 있는 데이터가 저장되는 모든 종류의 기록 장치를 포함한다. 또한 컴퓨터가 읽을 수 있는 기록매체는 네트워크로 연결된 컴퓨터 시스템에 분산되어 분산 방식으로 컴퓨터로 읽을 수 있는 프로그램 또는 코드가 저장되고 실행될 수 있다.
- [0082] 상기 컴퓨터가 읽을 수 있는 기록매체는 롬(rom), 램(ram), 플래시 메모리(flash memory) 등과 같이 프로그램 명령을 저장하고 수행하도록 특별히 구성된 하드웨어 장치를 포함할 수 있다. 프로그램 명령은 컴파일러(compiler)에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터(interpreter) 등을 사용해서 컴퓨터에 의해 실행될 수 있는 고급 언어 코드를 포함할 수 있다.
- [0083] 본 발명의 일부 측면들은 장치의 문맥에서 설명되었으나, 그것은 상응하는 방법에 따른 설명 또한 나타낼 수 있고, 여기서 블록 또는 장치는 방법 단계 또는 방법 단계의 특징에 상응한다. 유사하게, 방법의 문맥에서 설명된 측면들은 또한 상응하는 블록 또는 아이템 또는 상응하는 장치의 특징으로 나타낼 수 있다. 방법 단계들의 몇몇 또는 전부는 예를 들어, 마이크로프로세서, 프로그램 가능한 컴퓨터 또는 전자 회로와 같은 하드웨어 장치에 의해서 또는 이를 이용하여 수행될 수 있다. 몇몇의 실시예에서, 가장 중요한 방법 단계들의 하나 이상은 이와 같은 장치에 의해 수행될 수 있다.
- [0084] 실시예들에서, 프로그램 가능한 로직 장치(예를 들어, 필드 프로그래머블 게이트 어레이)가 여기서 설명된 방법들의 기능의 일부 또는 전부를 수행하기 위해 사용될 수 있다. 실시예들에서, 필드 프로그래머블 게이트 어레이는 여기서 설명된 방법들 중 하나를 수행하기 위한 마이크로프로세서와 함께 작동할 수 있다. 일반적으로, 방법들은 어떤 하드웨어 장치에 의해 수행되는 것이 바람직하다.
- [0085] 위에서는 본 발명의 바람직한 실시예를 참조하여 설명하였지만, 해당 기술 분야의 숙련된 당업자는 하기의 특허 청구의 범위에 기재된 본 발명의 사상 및 영역으로부터 벗어나지 않는 범위 내에서 본 발명을 다양하게 수정 및 변경시킬 수 있음을 이해할 수 있을 것이다.

도면

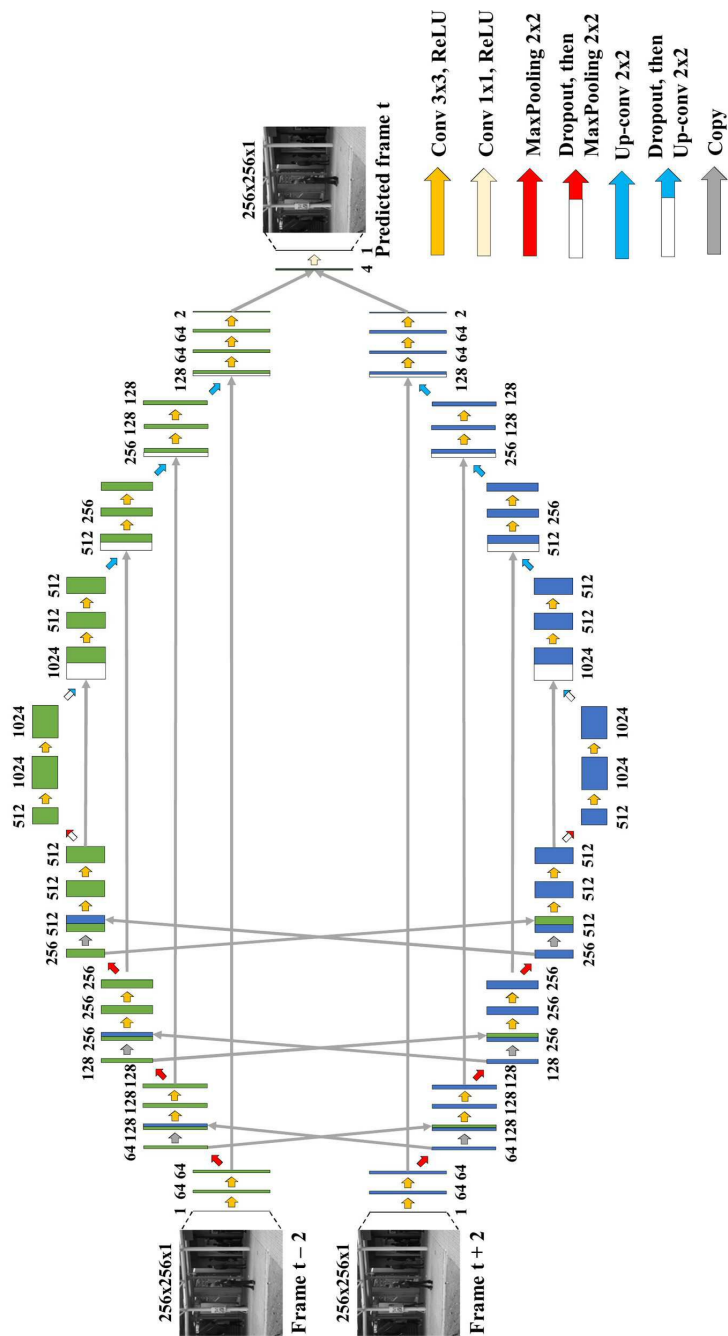
도면1



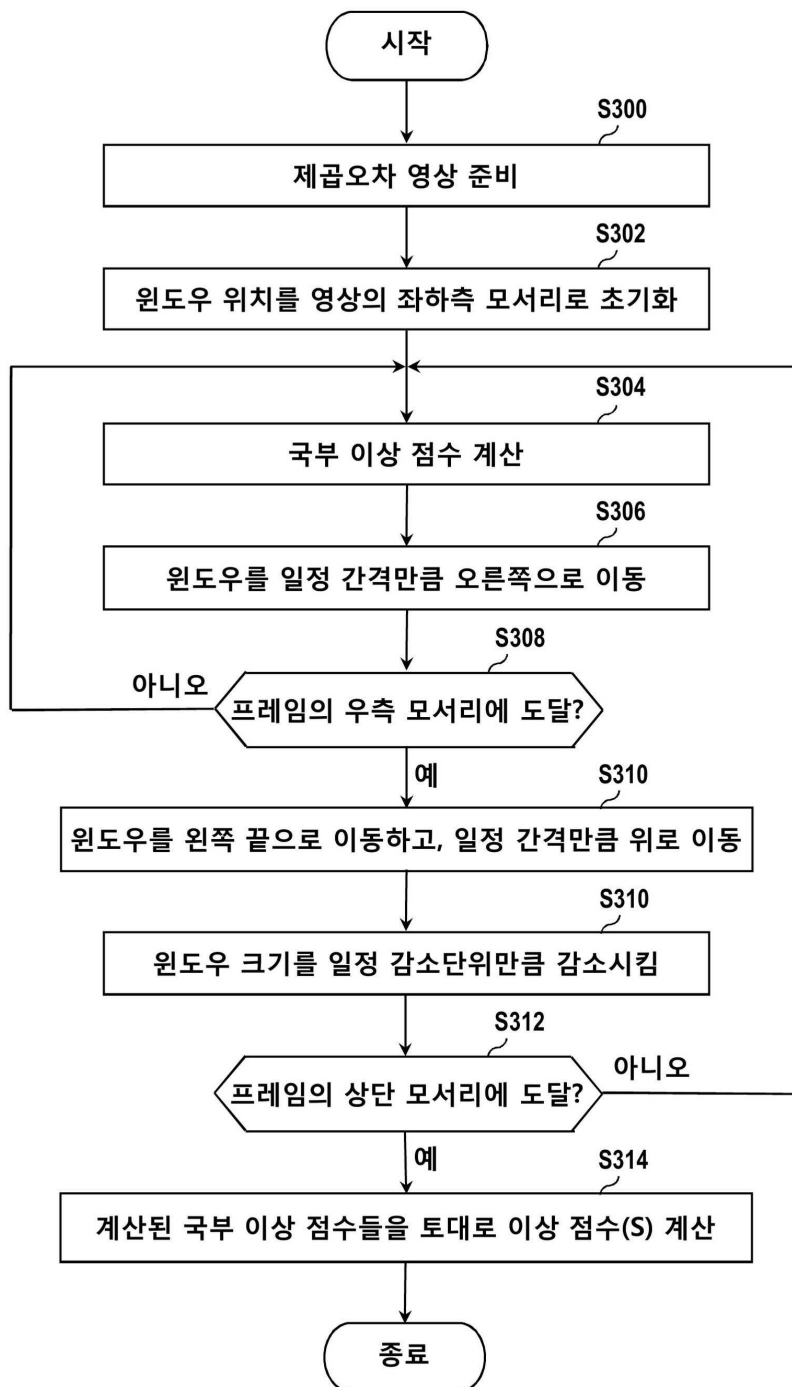
도면2



도면3



도면4



도면5

Algorithm 1. Cascade sliding window algorithm

Input: real current frame $F_{i,j}$, predicted current frame $\hat{F}_{i,j}$, frame size s , window size $\hat{s}$, window decrease size d

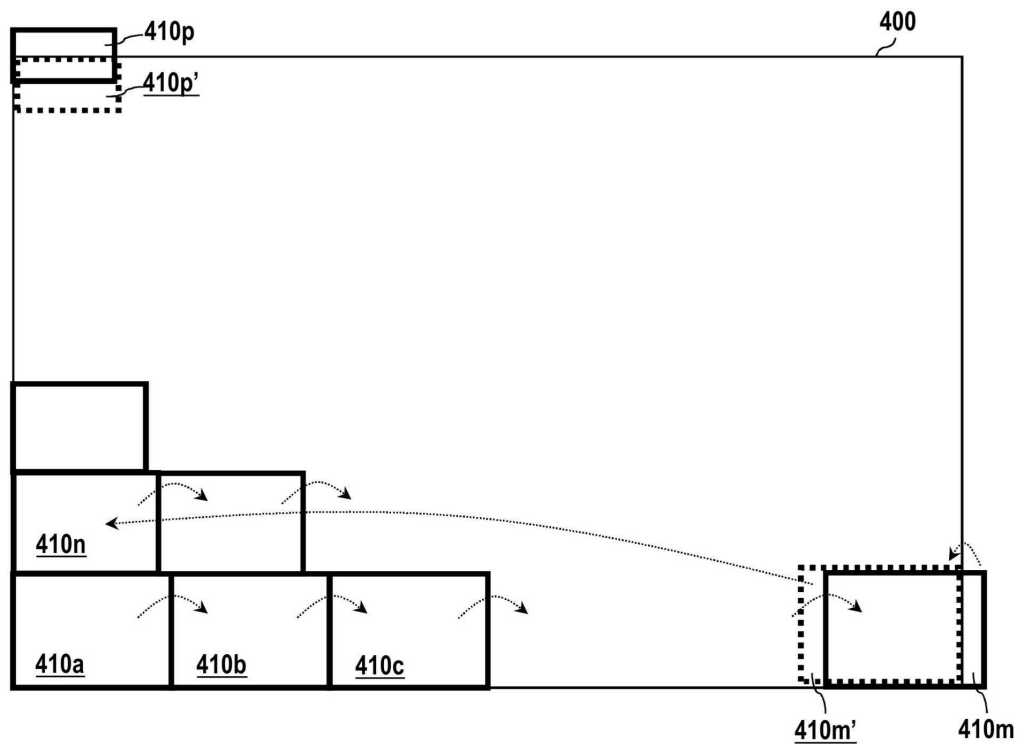
Output: anomaly score S

```

1   $I_{i,j} = (F_{i,j} - \hat{F}_{i,j})^2$  /* image on the square of the difference between  $F_{i,j}$  and  $\hat{F}_{i,j}$  */
2  Initialize coordinates  $x = 0$  and  $y = 0$ , mean squared error of each patch  $p_k$ 
3  while  $y < s$  do
4      if  $y + \hat{s} \leq s$  then
5          while  $x < s$  do
6              if  $x + \hat{s} \leq s$  then
7                   $p_k = \frac{1}{\hat{s}^2} \sum_{i=x}^{x+\hat{s}} \sum_{j=y}^{y+\hat{s}} I_{i,j}$ 
8              else
9                   $p_k = \frac{1}{\hat{s}^2} \sum_{i=s-\hat{s}}^s \sum_{j=y}^{y+\hat{s}} I_{i,j}$ 
10             end if
11              $x = x + \hat{s}$ 
12         end while
13     else
14         while  $x < s$  do
15             if  $x + \hat{s} \leq s$  then
16                  $p_k = \frac{1}{\hat{s}^2} \sum_{i=x}^{x+\hat{s}} \sum_{j=s-\hat{s}}^s I_{i,j}$ 
17             else
18                  $p_k = \frac{1}{\hat{s}^2} \sum_{i=s-\hat{s}}^s \sum_{j=s-\hat{s}}^s I_{i,j}$ 
19             end if
20              $x = x + \hat{s}$ 
21         end while
22      $x = 0$ 
23      $y = y + \hat{s}$ 
24      $\hat{s} = \hat{s} - d$ 
25     sort( $p_k$ ) in ascending order
26      $S = \frac{1}{n} \sum_{i=1}^n p_i$ 
27     Return  $S$ 

```

도면6



도면7



도면8

Dataset	Window decrease size	AUC (%) / Initial window size (pixels) / Patch number
Avenue	0	90.59 / 36 ² / 6
	1	90.65 / 35 ² / 7
	2	90.61 / 52 ² / 4
	3	90.74 / 54 ² / 3
	4	90.77 / 57² / 3
	5	90.68 / 59 ² / 3
Ped2	0	96.99 / 29² / 1
	1	96.62 / 40 ² / 1
	2	96.13 / 50 ² / 1
	3	95.51 / 54 ² / 1
	4	94.78 / 57 ² / 1
	5	94.09 / 59 ² / 1
ShanghaiTech	0	72.48 / 20² / 2
	1	71.82 / 36 ² / 1
	2	71.45 / 52 ² / 1
	3	71.26 / 58 ² / 1
	4	71.14 / 73 ² / 1
	5	71.07 / 74 ² / 1

도면9

Method	Avenue	Ped2	ShanghaiTech
AMDN [25]	N/A	90.8	N/A
Zaheer et al. [22]	N/A	98.1	N/A
Conv-AE [13]	70.2	90.0	N/A
ConvLSTM-AE [17]	77.0	88.1	N/A
TSC [14]	81.7	92.2	68.0
Liu et al. [18]	84.9	95.4	72.8
sRNN-AE [15]	83.5	92.2	69.6
Tang et al. [21]	83.7	96.2	71.5
Anomalynet [16]	86.1	94.9	N/A
MPED-RNN [11]	86.3	N/A	73.4
Nguyen et al. [23]	86.9	96.2	N/A
STAN [20]	87.2	96.5	N/A
Chen et al. [24]	87.8	96.6	N/A
Ionescu et al. [10]	90.4	97.8	84.9
Cross U-Net	90.8	97.0	72.5

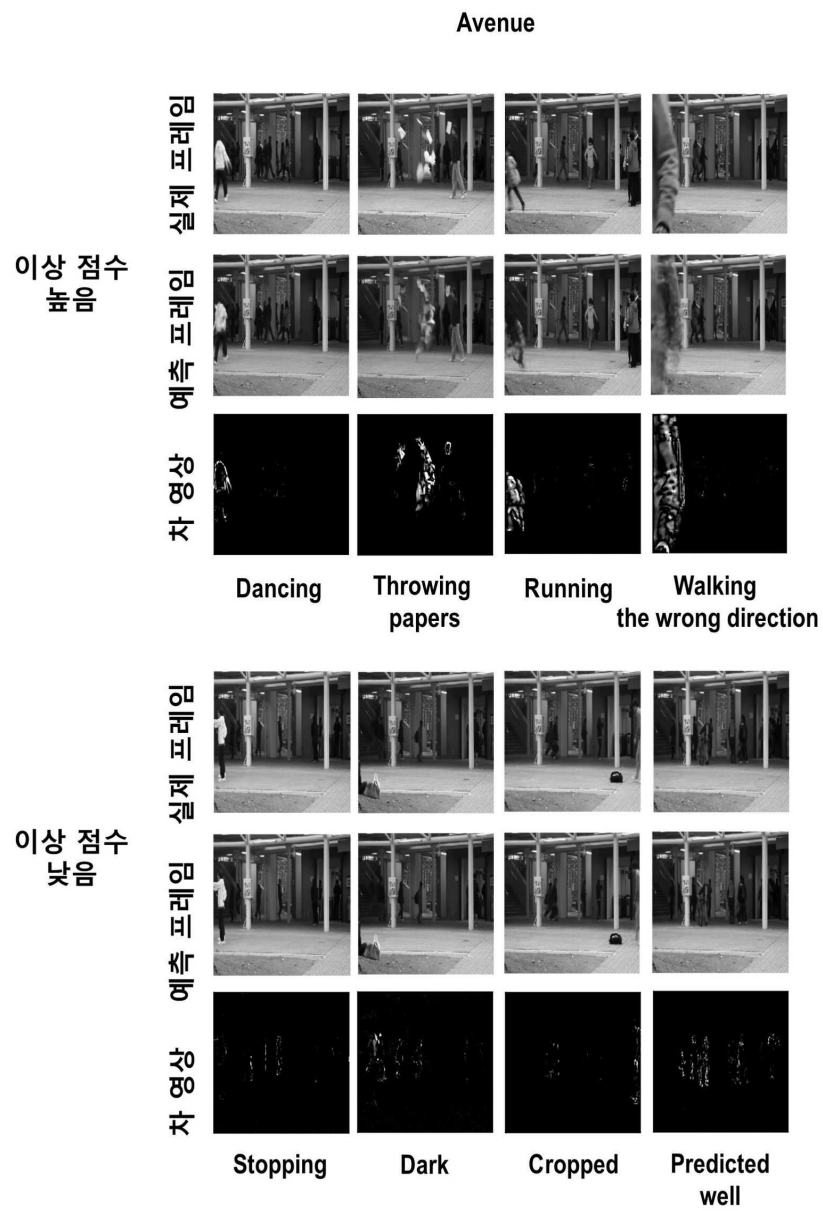
도면10

	Dataset		
	Avenue	Ped2	ShanghaiTech
Preprocessing time	1	1	6
Prediction time		28	
Inference time	2	4	7
Running time	31	33	41

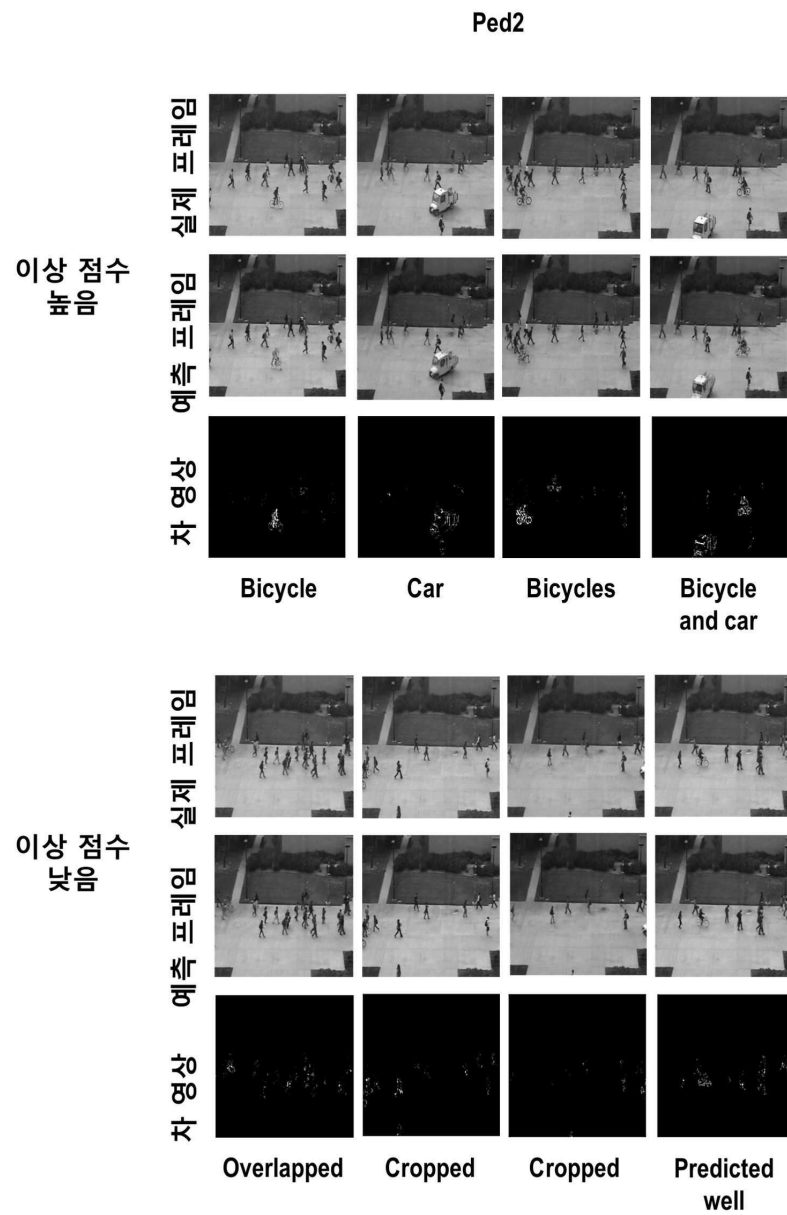
도면11

Method	Dataset	Running Time
Ionescu et al. [10]	Avenue	110
	Ped2	224
	ShanghaiTech	127
Cross U-Net	Avenue	31
	Ped2	33
	ShanghaiTech	41

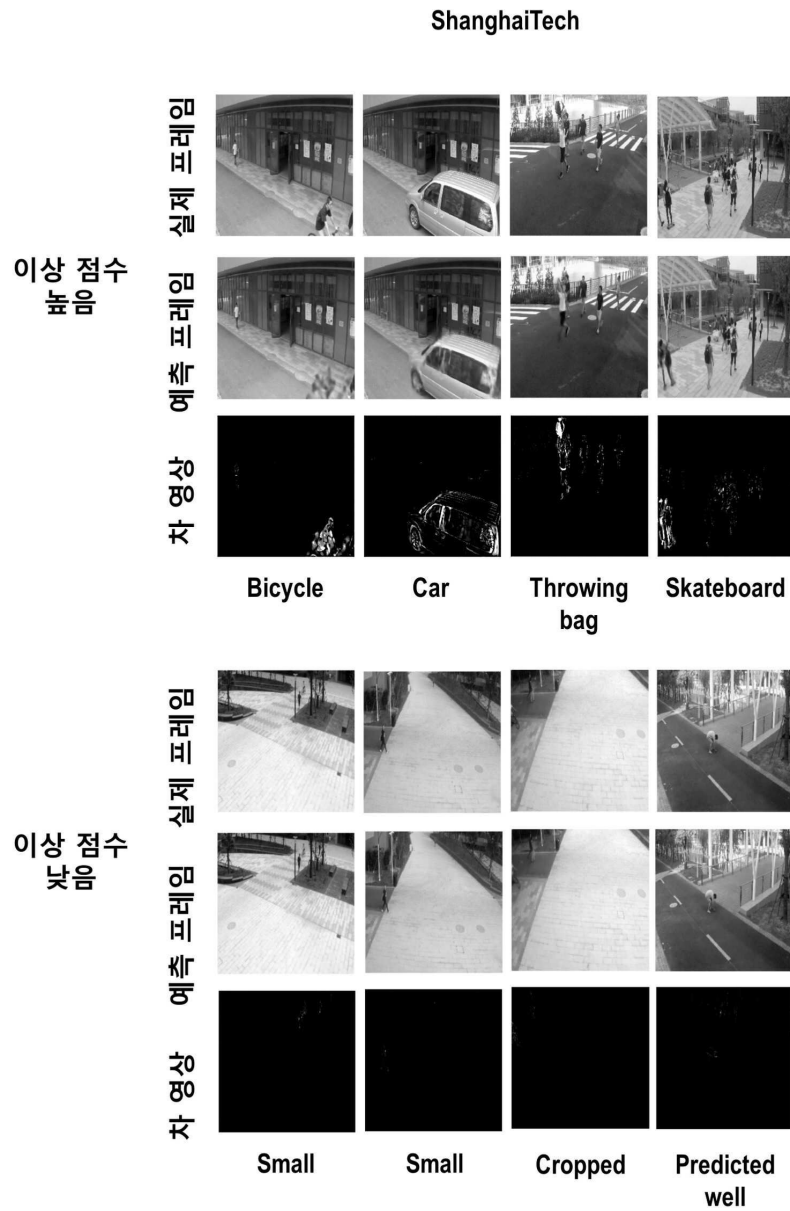
도면 12a



도면 12b



도면12c



도면13

Model	AUC (%)	
	Avenue	Ped2
No concatenation	87.88	80.05
CC	87.34	81.23
CE	90.23	93.55
Cross U-Net (CC + CE)	90.77	96.99