(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2019-0102915
(43) 공개일자 2019년09월04일

(51) 국제특허분류(Int. Cl.)
G06F 3/01 (2006.01) *G06F 3/0346* (2013.01)
G06N 3/04 (2006.01) *G06N 3/08* (2006.01)
(52) CPC특허분류
G06F 3/017 (2013.01)
G06F 3/014 (2013.01)
(21) 출원번호 10-2018-0024006
(22) 출원일자 2018년02월27일
심사청구일자 2018년02월27일

(71) 출원인
세종대학교산학협력단
서울특별시 광진구 능동로 209 (군자동, 세종대학교)
(72) 발명자
김형석
서울특별시 광진구
오준혁
서울특별시 광진구 능동로 209 세종대학교
(74) 대리인
김현승

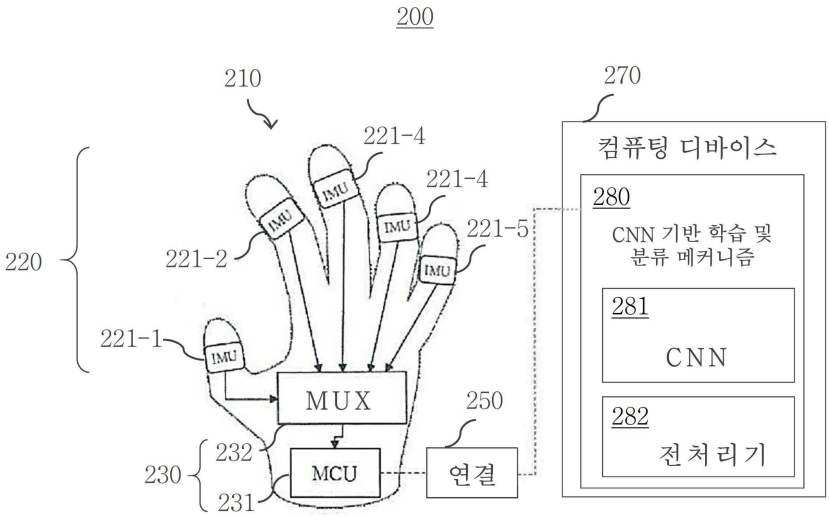
전체 청구항 수 : 총 10 항

(54) 발명의 명칭 **착용가능 디바이스를 사용하여 신경망 기반의 제스처 인식을 수행하는 기법**

(57) 요 약

제스처 인식 시스템이 제공된다. 개시된 제스처 인식 시스템은, 손 제스처에 관한 데이터를 감지하는 적어도 하나의 관성 측정 유닛(Inertial Measurement Unit: IMU) 및 감지된 데이터를 포함하는 제스처 데이터를 제공하는 제어기를 포함하는 글러브형 착용가능 디바이스와, 글러브형 착용가능 디바이스와 통신가능하게 커플링된 컴퓨팅 디바이스를 포함하되, 컴퓨팅 디바이스는 제스처 데이터를 사용하여 손 제스처를 인식하기 위한 콘볼루션 신경망(Convolutional Neural Network: CNN) 기반 학습 및 분류 메커니즘을 구현하도록 구성되고, 적어도 하나의 IMU 각각은 가속도계, 자이로스코프 및 지자기 센서를 포함한다.

대 표 도



(52) CPC특허분류
G06F 3/016 (2013.01)
G06F 3/0346 (2013.01)
G06F 3/038 (2013.01)
G06N 3/04 (2013.01)
G06N 3/08 (2013.01)

이 발명을 지원한 국가연구개발사업
과제고유번호 1711054220
부처명 과학기술정보통신부
연구관리전문기관 한국연구재단
연구사업명 개인기초연구(미래부)
연구과제명 압축센싱과 자원할당 기술 기반의 저전력 고효율 가시광통신시스템 연구개발
기 여 율 1/1
주관기관 세종대학교
연구기간 2017.06.01 ~ 2018.03.31

명 세 서

청구범위

청구항 1

제스처 인식 시스템으로서,

손 제스처에 관한 데이터를 감지하는 적어도 하나의 관성 측정 유닛(Inertial Measurement Unit: IMU) 및 상기 감지된 데이터를 포함하는 제스처 데이터를 제공하는 제어기를 포함하는 글러브형 착용가능 디바이스와,

상기 글러브형 착용가능 디바이스와 통신가능하게 커플링된 컴퓨팅 디바이스를 포함하되, 상기 컴퓨팅 디바이스는 콘볼루션 신경망(Convolutional Neural Network: CNN)에 기반하여 상기 제스처 데이터로부터 상기 손 제스처를 인식하는 CNN 기반 학습 및 분류 메커니즘을 구현하도록 구성되고, 상기 적어도 하나의 IMU 각각은 가속도계, 자이로스코프 및 지자기 센서를 포함하는,

제스처 인식 시스템.

청구항 2

제1항에 있어서,

상기 적어도 하나의 IMU는 상기 글러브형 착용가능 디바이스의 손가락 형상 부분에 구비되어 상기 손가락 형상 부분의 가속도, 각속도 및 지자기를 측정하는 IMU를 포함하는,

제스처 인식 시스템.

청구항 3

제1항에 있어서,

상기 제어기는 상기 IMU로부터 상기 감지된 데이터를 수신하고 통신 매체를 통해 상기 컴퓨팅 디바이스에 상기 제스처 데이터를 제공하는 마이크로제어기 유닛(Micro-Controller Unit: MCU)을 포함하는,

제스처 인식 시스템.

청구항 4

제3항에 있어서,

상기 글러브형 착용가능 디바이스에 구비된 상기 IMU는 상기 글러브형 착용가능 디바이스의 복수의 손가락 형상 부분에 각각 구비된 복수의 IMU이고, 상기 제어기는 상기 MCU와 상기 복수의 IMU 간의 통신을 가능하게 하는 멀티플렉서(multiplexer)를 더 포함하는,

제스처 인식 시스템.

청구항 5

제3항에 있어서,

상기 MCU는 또한 상기 가속도계, 상기 자이로스코프 및 상기 지자기 센서의 캘리브레이션을 수행하는,

제스처 인식 시스템.

청구항 6

제3항에 있어서,

상기 감지된 데이터는 상기 IMU 각각을 위한 3개의 직교 축 각각 상의 가속도 값, 각속도 값 및 지자기 값을 포함하고, 상기 MCU는 또한 상기 감지된 데이터에 기반하여 오일러 각(Euler angles), 쿼터니언(Quaternion), 또는 양자 모두를 포함하는 보조 데이터를 계산하며, 상기 제스처 데이터는 상기 보조 데이터를 더 포함하는,

제스처 인식 시스템.

청구항 7

제3항에 있어서,

상기 제어기는 촉각적 신호 생성기(haptic signal generator)를 더 포함하되, 상기 MCU는 상기 제스처 데이터로부터의 상기 손 제스처의 인식이 유보된다는 경보 신호를 상기 컴퓨팅 디바이스로부터 수신하는 것에 응답하여 상기 촉각적 신호 생성기로 하여금 촉각적 피드백(haptic feedback)을 제공하도록 명령하는,

제스처 인식 시스템.

청구항 8

제1항에 있어서,

상기 CNN 기반 학습 및 분류 메커니즘은 상기 CNN으로의 입력을 위해 상기 제스처 데이터에 대해 상기 CNN의 훈련 데이터와 동일한 전처리(preprocessing)를 수행하는,

제스처 인식 시스템.

청구항 9

제1항에 있어서,

상기 CNN은 적어도 하나의 콘볼루션 층(convolutional layer)과, 적어도 하나의 풀링 층(pooling layer)과, 상기 CNN의 최종 출력을 위한 피드포워드 신경망 층(feed-forward neural network layer)을 포함하되, 상기 적어도 하나의 콘볼루션 층의 활성화 함수는 시그모이드 함수 또는 정류화된 선형 유닛(Rectified Linear Unit: ReLU) 함수이고, 상기 적어도 하나의 풀링 층은 최대 풀링을 수행하며, 상기 피드포워드 신경망 층의 활성화 함수는 소프트맥스(softmax) 함수인,

제스처 인식 시스템.

청구항 10

제1항에 있어서,

상기 CNN 기반 학습 및 분류 메커니즘은 상기 CNN의 최종 출력에 따라 상기 손 제스처를 복수의 손 제스처 부류 중 하나로 분류하거나, 상기 최종 출력 또는 상기 최종 출력 전에 상기 CNN에서 계산된 특정 값에 따라 상기 손 제스처의 인식을 유보하기로 판정하는,

제스처 인식 시스템.

발명의 설명

기 술 분 야

[0001] 본 개시는 제스처 인식(gesture recognition)을 위한 기법에 관한 것이다.

배 경 기 술

[0002] 최근 정보통신 기술(Information and Communication Technology: ICT)이 고도화되면서, ICT 인프라스트럭처(infrastructure)를 통한 데이터 수집 및 인공 지능을 접목하는 지능 정보 기술이 확산되고 있다.

[0003] 이와 함께, 모바일 디바이스(mobile device) 기반의 구글 카드보드(Google Cardboard), 개인용 컴퓨터(Personal Computer: PC) 기반의 오큘러스 리프트(Oculus Rift) 등등과 같은 다양한 플랫폼 상에서 호스팅될 수 있는 실감형 콘텐츠에 대한 수요와 관심이 빠르게 늘어나면서, 사용자의 제스처(gesture)를 인식하기 위한 다양한 입력 인터페이스의 연구와 개발이 활발히 진행되고 있다.

[0004] 종래의 제스처 인식 시스템의 예를 마이크로소프트 사(Microsoft Corporation)에 의해 개발된 키넥트(Kinect)에서 볼 수 있는바, 이는 움직임 감지 입력 디바이스를 포함하는 움직임 제어기(motion controller)이다. Kinect는 카메라(예컨대, RGB 카메라 및 깊이 감지 적외선 카메라)를 구비한 시각 기반 제스처 인식(vision-based

gesture recognition)을 수행하는데, 사용자가 별도의 디바이스를 부착할 것을 요구하지는 않는다. 또한, Kinect 기술의 다양한 애플리케이션에서 사용자의 움직임 정보를 획득하는 데에 사용될 수 있는 소프트웨어 개발 키트(Software Development Kit: SDK)가 제공된다. 그러나, 카메라를 이용한 제스처 인식은 벽이나 장애물에 의해 방해를 받고, 사용자 행동의 반경이 카메라의 방향에 따라 제한적일 수 있다.

[0005] 다른 예로서, 3축 가속도계와 같은 센서를 사용하여 제스처를 인식하기 위한 시스템들이 제안되었다. 그러나, 그러한 시스템은 움직임의 측정이 충분히 정확하지 못하거나, 센서 의존성이 너무 높거나, 동적 제스처의 인식이 곤란하다.

[0006] 따라서, 제스처를 인식하기 위한 개선된 기법이 필요하다.

발명의 내용

해결하려는 과제

[0007] 착용가능 디바이스(wearable device)를 사용하여 신경망(neural network) 기반의 제스처 인식을 수행하기 위한 개선된 기법이 본 문서에 개시된다.

과제의 해결 수단

[0008] 적어도 하나의 실시예에 따르면, 제스처 인식 시스템은, 손 제스처에 관한 데이터를 감지하는 적어도 하나의 관성 측정 유닛(Inertial Measurement Unit: IMU) 및 상기 감지된 데이터를 포함하는 제스처 데이터를 제공하는 제어기를 포함하는 글러브형 착용가능 디바이스와, 상기 글러브형 착용가능 디바이스와 통신가능하게 커플링된 컴퓨팅 디바이스를 포함하되, 상기 컴퓨팅 디바이스는 콘볼루션 신경망(Convolutional Neural Network: CNN)에 기반하여 상기 제스처 데이터로부터 상기 손 제스처를 인식하는 CNN 기반 학습 및 분류 메커니즘을 구현하도록 구성되고, 상기 적어도 하나의 IMU 각각은 가속도계, 자이로스코프 및 지자기 센서를 포함한다.

[0009] 상기 적어도 하나의 IMU는 상기 글러브형 착용가능 디바이스의 손가락 형상 부분에 구비되어 상기 손가락 형상 부분의 가속도, 각속도 및 지자기를 측정하는 IMU를 포함할 수 있다.

[0010] 상기 제어기는 상기 IMU로부터 상기 감지된 데이터를 수신하고 통신 매체를 통해 상기 컴퓨팅 디바이스에 상기 제스처 데이터를 제공하는 마이크로제어기 유닛(Micro-Controller Unit: MCU)을 포함할 수 있다.

[0011] 상기 글러브형 착용가능 디바이스에 구비된 상기 IMU는 상기 글러브형 착용가능 디바이스의 복수의 손가락 형상 부분에 각각 구비된 복수의 IMU일 수 있고, 상기 제어기는 상기 MCU와 상기 복수의 IMU 간의 통신을 가능하게 하는 멀티플렉서(multiplexer)를 더 포함할 수 있다.

[0012] 상기 MCU는 또한 상기 가속도계, 상기 자이로스코프 및 상기 지자기 센서의 캘리브레이션을 수행할 수 있다.

[0013] 상기 감지된 데이터는 상기 IMU 각각을 위한 3개의 직교 축 각각 상의 가속도 값, 각속도 값 및 지자기 값을 포함할 수 있고, 상기 MCU는 또한 상기 감지된 데이터에 기반하여 오일러 각(Euler angles), 쿼터니언(Quaternion), 또는 양자 모두를 포함하는 보조 데이터를 계산할 수 있으며, 상기 제스처 데이터는 상기 보조 데이터를 더 포함할 수 있다.

[0014] 상기 제어기는 촉각적 신호 생성기(haptic signal generator)를 더 포함할 수 있되, 상기 MCU는 상기 제스처 데이터로부터의 상기 손 제스처의 인식이 유보된다는 경보 신호를 상기 컴퓨팅 디바이스로부터 수신하는 것에 응답하여 상기 촉각적 신호 생성기로 하여금 촉각적 피드백(haptic feedback)을 제공하도록 명령할 수 있다.

[0015] 상기 CNN 기반 학습 및 분류 메커니즘은 상기 CNN으로의 입력을 위해 상기 제스처 데이터에 대해 상기 CNN의 훈련 데이터와 동일한 전처리(preprocessing)를 수행할 수 있다.

[0016] 상기 CNN은 적어도 하나의 콘볼루션 층(convolutional layer)과, 적어도 하나의 풀링 층(pooling layer)과, 상기 CNN의 최종 출력을 위한 피드포워드 신경망 층(feed-forward neural network layer)을 포함할 수 있되, 상기 적어도 하나의 콘볼루션 층의 활성화 함수는 시그모이드 함수 또는 정류화된 선형 유닛(Rectified Linear Unit: ReLU) 함수일 수 있고, 상기 적어도 하나의 풀링 층은 최대 풀링을 수행할 수 있으며, 상기 피드포워드 신경망 층의 활성화 함수는 소프트맥스(softmax) 함수일 수 있다.

[0017] 상기 CNN 기반 학습 및 분류 메커니즘은 상기 CNN의 최종 출력에 따라 상기 손 제스처를 복수의 손 제스처 부류 중 하나로 분류하거나, 상기 최종 출력 또는 상기 최종 출력 전에 상기 CNN에서 계산된 특정 값에 따라 상기 손

제스처의 인식을 유보하기로 판정할 수 있다.

발명의 효과

[0018] 본 발명의 실시예에 따르면, 착용가능 디바이스를 사용하여 신경망 기반의 제스처 인식을 수행함으로써 인식 성공률을 개선하는 것이 가능하다.

[0019] 본 발명의 실시예에 따르면, 사용자의 제스처, 가령 손 제스처(특히, 손가락 움직임)이 더 정확히 인식될 수 있고, 따라서 교육이나 게이밍 등등에 관련된 가상 현실(Virtual Reality: VR) 환경에서 원활한 상호작용이 용이하게 될 수 있다.

[0020] 본 발명의 실시예는 센서 정확도에 덜 의존적이면서도 정지된 제스처뿐만 아니라 움직이는 제스처의 인식을 위한 개선된 기법을 제공한다.

[0021] 본 발명의 실시예는 실감형 콘텐츠의 저변 확대를 위한 효과적인 플랫폼 설계에 응용될 수 있다.

도면의 간단한 설명

[0022] 도 1은 예시적인 CNN의 개략적인 블록도이다.

도 2a 및 도 2b는 본 발명의 실시예에 따라 제스처를 인식하기 위한 예시적인 시스템을 개략적으로 도시한다.

도 3은 본 발명의 실시예에 따른 제스처 인식 시스템에 의해 수행될 수 있는 예시적인 프로세스의 흐름도이다.

도 4는 본 발명의 실시예에 따라 제스처 인식 시스템에 의한 사용을 위한 예시적인 제스처 데이터를 보여준다.

도 5는 본 발명의 실시예에 따른 제스처 인식 시스템에서 제스처에 관해 감지된 데이터의 그래프이다.

도 6 내지 도 11은 본 발명의 실시예에 따른 제스처 인식 시스템의 성능을 설명하기 위한 도면이다.

발명을 실시하기 위한 구체적인 내용

[0023] 이하에서는, 첨부된 도면을 참조하여 본 발명의 실시예를 상세히 설명한다. 본 발명은 여러 가지 실시예를 가질 수 있고, 몇몇 실시예가 본 명세서에 개시된다. 그러나, 이는 본 발명에 대한 한정이 아니라 예시로서 제공되며, 본 발명의 사상 및 범주에 속하는 모든 변환, 균등물 내지 대체물을 망라하는 것으로 이해되어야 한다. 개시된 실시예에 따른 방법, 장치 및/또는 시스템에 대한 포괄적인 이해를 돕기 위해 다음의 상세한 설명에서 특정한 세부사항이 제공되는데, 몇몇 실시예는 이들 세부사항 중 일부 또는 전부가 없더라도 실시될 수 있다. 또한, 본 발명의 다양한 양상을 불필요하게 모호하게 하지 않도록 공지 기술의 구체적인 설명은 생략될 수 있다.

[0024] 후술되는 용어는 단지 특정 실시예를 설명하기 위해 사용된 것으로, 한정적 의미로 고려되고자 의도된 것이 아니다. 단수 형태의 표현은 명확하게 달리 사용되지 않는 한, 복수 형태의 의미를 포함한다. 또한, 이 문서에서, "포함하다" 또는 "가지다"와 같은 용어는 어떤 특징, 숫자, 단계, 동작, 구성요소, 정보 또는 이들의 조합이 존재함을 나타내려는 것이며, 하나 또는 그 이상의 다른 특징, 숫자, 단계, 동작, 구성요소, 정보 또는 이들의 조합의 존재 또는 가능성을 배제하지 않는 것으로 이해되어야 한다.

[0025] 본 개시는 관성 측정 유닛(Inertial Measurement Unit: IMU)이 구비되어 사용자 제스처에 관해 감지된 정보를 제공하는 글러브형 착용가능 디바이스(glove-type wearable device)(이하에서 "데이터 글러브"(Data Glove)로 지칭될 수도 있음) 및 그러한 감지된 데이터로부터 제스처 특징을 추출하는 콘볼루션 신경망(Convolutional Neural Network: CNN)과 같은 인공 신경망이 조합된 효과적인 제스처 인식 기법을 예시한다. 다만, 본 개시의 범주는 이 점에 한정되지 않고, 다른 유형의 착용가능 디바이스(예컨대, 시계, 팔찌, 반지, 발찌, 머리띠 등등과 같은 다른 형상을 가짐) 및/또는 다른 유형의 신경망에 관해서도 아래의 논의가 적용될 수 있음에 유의하여야 한다. 아래에서, CNN에 대한 개괄적인 설명에 이어서, 본 개시의 제스처 인식 기법의 실시예가 상세히 논의된다.

[0026] CNN의 개요

[0027] 일반적으로, 다층 신경망은 입력 층(input layer)과, 하나 이상의 은닉 층(hidden layer)과 출력 층(output layer)을 포함한다. 신경망의 각각의 층은 하나 이상의 유닛(또는 노드)을 포함한다. 신경망 내의 유닛들, 특히 은닉 층과 출력 층의 유닛 각각은 복수의 입력을 취하여 하나의 출력을 계산한다. 예를 들어, 유닛의 출력은 바

로 이전의 층의 유닛으로부터의 출력의 가중화된 합에 (선택적으로 바이어스(bias)를 적용한 후) 활성화 함수(activation function)를 적용함으로써 산출될 수 있다. 한정이 아니라 예로서, 활성화 함수는 시그모이드(sigmoid) 함수, 램프(ramp) 함수, 항등 맵핑(identity mapping), 계단(step) 함수, 맥스아웃(maxout) 함수, 또는 소프트맥스(softmax) 함수일 수 있다. 램프 함수를 사용하는 유닛은 정류화된 선형 유닛(Recitified Linear Unit: ReLU)으로 칭해지기도 하며, 램프 함수는 정류화된 선형 함수(rectified linear function)로 또는 단순히 ReLU 함수로 칭해지기도 한다. 이러한 신경망은 구글 사(Google, Inc.)의 오픈 소스 라이브러리(open-source library)인 텐서플로우(TensorFlow), 또는 다른 소프트웨어 패키지를 통하여 구현될 수 있다.

[0028] CNN은 생물의 뇌 안의 시각 피질(visual cortex)을 모방한 것인데, 딥 러닝(deep learning)에서 지도 학습(supervised learning) 방식으로 특징 추출을 수행하는 데에 사용될 수 있다. 도 1은 예시적인 CNN의 개략적인 블록도이다. 도 1에 도시된 바와 같이, 예시적인 CNN(100)은 입력 층(110), 이어서 콘볼루션 층(convolutional layer)(120), 이어서 풀링 층(pooling layer)(130)을 포함한다. CNN(100)에서, 콘볼루션 층 및 후속하는 풀링 층의 추가적인 쌍(예컨대, 콘볼루션 층(140)과 풀링 층(150))이 더 포함될 수 있다. 또한, 비록 도시되지 않았으나, 그러한 콘볼루션-풀링 층 구조에서, 콘볼루션 층(가령, 콘볼루션 층(120))과 풀링 층(가령, 풀링 층(130)) 사이에 적어도 하나의 추가적인 콘볼루션 층이 배치될 수 있고/있거나, 풀링 층(가령, 풀링 층(130)) 뒤에 다른 층(가령, 국소 콘트라스트 정규화(Local Contrast Normalization: LCM) 층)이 추가로 배치될 수 있다. 콘볼루션 층과 풀링 층과 같은 중간 층의 출력 데이터는 특징 맵(feature map) 또는 맵이라 칭해지기도 한다. 예시적인 CNN(100)은 마지막 콘볼루션-풀링 층 구조 뒤에 배치된 적어도 하나의 피드포워드(feed-forward) 신경망 층, 예컨대 인접한 층의 유닛이 모두 서로 연결된 완전 연결 층(fully-connected layer)(가령, 완전 연결 층(160, 170))을 더 포함한다.

[0029] 콘볼루션 층(가령, 콘볼루션 층(120))의 유닛은 완전 연결 층과 달리, 인접한 층(가령, 입력 층(110))의 특정한 유닛과 연결되고(즉, 국소적 연결), 커널(kernel)로도 불리는 필터(filter)) 및 콘볼루션 층의 입력 데이터 간의 상관(correlation)을 도출한다. 이 필터의 계수는 그러한 국소적 연결의 가중치를 나타내는데, 이는 콘볼루션 층의 유닛에서 동일하다(즉, 가중치 공유). 콘볼루션 층에서 계산된 상관은 입력 데이터와 필터에 대한 콘볼루션(convolution; 합성곱)과 실질적으로 마찬가지이다.

[0030] 콘볼루션 층에서 필터가 입력 데이터에 대해 움직여지는 거리는 스트라이드(stride)로 칭해진다. 예를 들어, [0, 3, 2, 1, -1, 3, 3]이라는 입력 데이터에 대해 [-1, 0, 1]이라는 커널이 적용되는 위치의 간격이 1인 경우, 곧 스트라이드가 1인 콘볼루션 연산이 수행되는 경우, 그 연산의 결과는 [-1×0 + 0×3 + 1×2, -1×3 + 0×2 + 1×1, -1×2 + 0×1 + 1×(-1), -1×1 + 0×(-1) + 1×3, -1×(-1) + 0×3 + 1×3] = [2, -2, -3, 2, 4]이다. 다른 예에서, 동일한 커널과 동일한 입력 데이터에 대해 스트라이드가 2인 콘볼루션 연산이 수행되는 경우, 그 연산의 결과는 [-1×0 + 0×3 + 1×2, -1×2 + 0×1 + 1×(-1), -1×(-1) + 0×3 + 1×3] = [2, -3, 4]이다. 물론, 이들 예에서 보여지지는 않았으나, 콘볼루션 연산의 결과에 바이어스의 가산이 도입될 수도 있다.

[0031] 콘볼루션 층의 유닛은 출력을 위해 그러한 연산 결과에 활성화 함수(가령, 시그모이드 함수, ReLU 함수, 계단 함수 또는 유사한 것)를 적용한다. 예시된 바와 같이, 콘볼루션 층의 출력 데이터의 크기는 입력 데이터의 크기, 필터의 크기 및 스트라이드에 따라 정해질 수 있다. 예를 들어, N×N의 크기를 갖는 입력 이미지 데이터에 대해 F×F 커널이 스트라이드 S로 적용되는 경우, C=(N-F)/S+1인 C×C의 크기를 갖는 출력 데이터가 주어질 수 있다. 예시된 바와 같이, 스트라이드의 값이 증가할수록 출력 데이터의 크기는 감소함이 이해될 것이다. 따라서, 스트라이드의 값을 증가시킴으로써 계산량을 줄일 수 있다. 그러나, 과도한 스트라이드 값은 입력 데이터의 특징이 제대로 추출되지 못하게 할 수 있다.

[0032] 풀링 층(가령, 풀링 층(130))은 계산량을 줄이도록 풀링을 통해 감소된 크기의 특징 맵을 제공한다. 이러한 풀링은 콘볼루션 층에서 추출된 특징의 토폴로지(topology) 특성이나 위치가 변하더라도 풀링 층의 출력이 불변하게 할 수 있다. 풀링 층 역시 인접한 층(가령, 콘볼루션 층(120))과 국소적으로 연결된다. 예를 들어, 풀링 층은 콘볼루션 층으로부터의 특징 맵에 대해, 풀링 윈도우(window) 영역에 걸쳐 평균값을 출력하는 평균 풀링(average pooling)을 수행할 수 있거나(즉, 풀링 층은 평균 풀링 층일 수 있음) 또는 풀링 윈도우 영역 내의 최대값을 출력하는 최대 풀링(max pooling)을 수행할 수 있다(즉, 풀링 층은 최대 풀링 층일 수 있음). 또한, 풀링 층에서도 콘볼루션 층과 유사하게 스트라이드가 설정될 수 있다. 특히, 풀링 층에서는 그 취지에 부합하도록 2 이상의 스트라이드가 사용될 수 있다.

[0033] 피드포워드(feed-forward) 신경망 층(가령, 완전 연결 층(160, 170))은 어떤 층의 유닛이 오직 인접 층과 연결된 구조를 갖는 신경망이다. 도시된 바와 같이, CNN(100)의 최종적인 출력은 마지막 피드포워드 신경망

층(가령, 완전 연결 층(170))의 출력이다(즉, 마지막 피드포워드 신경망 층이 CNN(100)의 출력 층임). 예를 들어, CNN(100)이 다부류 분류(multi-class classification)에 정향된 경우, CNN(100)의 출력 층에서 소프트맥스 함수가 활성화 함수로서 사용될 수 있고, 그러한 출력 층은 소프트맥스 층으로 칭해질 수 있다.

[0034] CNN(100)에서 콘볼루션 층과 피드포워드 신경망 층의 파라미터(가령, 층간 연결의 가중치 및 바이어스)가 학습에 따라 변화할 수 있다는 점이 이해될 것이다. CNN 기반의 학습(가령, 딥 러닝(deep learning)) 알고리즘은 지도 학습(supervised learning)의 일종인데, 지도 학습은 입력 데이터로부터 예측된 값과 정답 레이블(label) 사이의 손실 오차를 오차 함수에 따라 구하고 그 오차를 갱신하는 방식으로 학습하는 일련의 과정을 갖는다. 특히, CNN 기반 학습 알고리즘은 분류 스테이지(stage)뿐만 아니라 특징 추출 스테이지를 수반한다. 그러한 특징 추출은 콘볼루션과 풀링을 통해 국소적인 특징으로부터 전역적인 특징을 얻는 것이다. 이에 따라, CNN 기반 학습 알고리즘은 위상(topology) 변화에 강인한 인식 능력을 가능하게 한다.

[0035] 예시적인 제스처 인식 시스템

[0036] 도 2a 및 도 2b는 본 발명의 실시예에 따라 제스처를 인식하기 위한 예시적인 시스템을 개략적으로 도시한다. 도 2a에 도시된 바와 같이, 예시적인 제스처 인식 시스템(200)은 사용자의 손에 착용될 수 있는 글러브형 착용가능 디바이스(210)와, 글러브형 착용가능 디바이스(210)와 통신가능하게 커플링된(communicatively coupled) 컴퓨팅 디바이스(computing device)(270)를 포함한다.

[0037] 도 2a를 참조하면, 글러브형 착용가능 디바이스(210)는 사용자의 손 제스처에 관한 데이터를 감지하기 위한 적어도 하나의 IMU(가령, 예시된 바와 같이, 5개의 손가락 형상 부분에 각각 구비된 5개의 IMU(221-1, 221-2, ..., 221-5)(이하에서 개별적으로 참조번호 221로 표기될 수 있음))를 포함하는 센서 모듈인 감지 메커니즘(sensing mechanism)(220)과, 감지 메커니즘(220)(특히, IMU(221))과의 통신(가령, I2C(Inter-Integrated Circuit) 버스 통신과 같은 버스 통신)이 되는 제어기(controller)(230)를 포함하는 것으로 예시된다.

[0038] 예시적인 제스처 인식 시스템(200)에서, IMU(221)는 몇몇 관성 센서, 예컨대 물체의 가속도를 측정하기 위한 적어도 하나의 가속도계(accelerometer) 및 물체의 각속도를 측정하기 위한 적어도 하나의 자이로스코프(gyroscope)를 포함할 수 있다. 또한, IMU(221)는 물체를 둘러싼 자기장을 측정하기 위한 적어도 하나의 자력계(magnetometer), 예컨대 지자기 센서(geomagnetic sensor)를 더 포함할 수 있다. 예를 들어, 손 제스처에 관해 IMU(221)에 의해 감지된 데이터는 IMU(221)에 포함된 가속도계, 자이로스코프 및 지자기 센서에 의해 각각 측정된 가속도 값, 각속도 값 및 지자기 값을 포함할 수 있다. 몇몇 실시예에서, IMU(221)의 가속도계, 자이로스코프 및 지자기 센서는 각각 3축 가속도계, 3축 자이로스코프 및 3축 지자기 센서일 수 있다.

[0039] 몇몇 실시예에서, IMU(221)는 앞서 언급된 바와 같이 글러브형 착용가능 디바이스(210)의 손가락 형상 부분에 구비될 수 있고, 이에 따라 손가락 형상 부분의 가속도, 각속도 및 지자기를 측정할 수 있다. 다만, 본 개시의 실시예는 이 점에 한정되지 않고, IMU(221)는 글러브형 착용가능 디바이스(210)의 다른 부분, 예컨대 손목 부분, 손등 부분, 손바닥 부분 등등에 구비될 수 있음이 이해될 것이다. 예시적인 제스처 인식 시스템(200)은 IMU(221)에 의해 감지된 데이터를 사용하여 글러브형 착용가능 디바이스(210)의 착용자의 손(가령, 손가락, 손목, 손등 및/또는 손바닥 부위)의 방향, 속도, 위치, 기울기 등등과 같은 물리적 양을 계산하는 일련의 프로세스를 수행할 수 있다.

[0040] 글러브형 착용가능 디바이스(210)가 착용된 초기 지점에서의 감지 메커니즘(220)의 초기화, 특히 IMU(221)에 의해 사용자 제스처에 관한 데이터를 적절히 감지하는 것을 위해, IMU(221) 주위의 환경에 IMU(221)의 센서들을 맞추는 캘리브레이션(calibration)이 요구될 수 있다. 예를 들어, 지자기 센서가 사용되는 경우 자기 왜곡(magnetic distortion)과 같이 측정 값의 왜곡을 야기하는 여러 가지 요인이 나타날 수 있으므로, 캘리브레이션을 통한 보정이 필요할 수 있다. 몇몇 실시예에서, 그러한 센서 캘리브레이션은 칼만 필터(Kalman filter), 예를 들어 확장 칼만 필터(extended Kalman filter)와 같은 필터를 적용하는 것을 포함할 수 있다. 칼만 필터는 잡음을 포함하는 측정 데이터로부터 동적 시스템(dynamic system)의 상태(state)를 추정하는 재귀적(recursive) 필터이다. 개략적으로 말하면, 칼만 필터의 동작은 다음과 같다. 시스템 상태 및 오차 공분산의 초기 추정치가 주어지면, 상태 전이 모델에 기반하여 상태가 예측되고 그것의 오차 공분산이 예측된다. 측정 모델 및 예측된 오차 공분산에 기반하여, 칼만 이득이 계산되는데, 상태의 추정치와 오차 공분산의 추정치를 갱신하는 데에 사용된다. 상태의 추정치의 갱신에서 측정치가 고려된다. 이러한 예측과 갱신이 반복되어 추정치가 재귀적으로 제공된다.

[0041] 제어기(230)는 감지 메커니즘(220)으로부터의 측정 데이터를 수신하고 처리하여 통신 매체(250)를 통해 컴퓨팅

디바이스(270)에 제스처 데이터를 송신하도록 구성될 수 있다. 통신 매체(250)는 직렬 케이블(serial cable)과 같은 유선 매체, 무선 통신을 위한 전자기파(electromagnetic wave)와 같은 무선 매체, 또는 이들의 조합일 수 있다.

[0042] 몇몇 실시예에서, 제어기(230)는 마이크로제어기 유닛(Micro-Controller Unit: MCU)(231)을 포함할 수 있는데, 이는 IMU(221)에 의해 감지된 데이터를 I2C 프로토콜에 따른 버스 상에서 IMU(221)로부터 수신하고, 그러한 감지된 데이터를 포함하는 제스처 데이터를 통신 매체(250)를 통해 컴퓨팅 디바이스(270)에 제공할 수 있다. 전술된 바와 같이, IMU(221)는 3축 가속도계, 3축 자이로스코프 및 3축 지자기 센서를 포함할 수 있는바, IMU(221)에 의해 감지된 데이터는 IMU(221)를 위한 3개의 직교 축 각각 상의 가속도 값, 각속도 값 및 지자기 값을 포함할 수 있다. 나아가, 제스처 데이터는 추가적인 데이터를 더 포함할 수 있다. 예를 들어, MCU(231)는 IMU(221)에 의해 감지된 데이터에 기반하여 오일러 각(Euler angles)(예컨대, 롤(roll), 피치(pitch), 요(yaw) 값) 및/또는 쿼터니언(Quaternion)을 포함하는 보조 데이터를 계산할 수 있고, 제스처 데이터는 앞서 언급된 감지된 데이터 외에도 그러한 보조 데이터를 포함할 수 있다.

[0043] MCU(231)는 이와 같이 IMU(221)에 의해 측정된 데이터를 IMU(221)로부터 수신하는 것, 그러한 데이터를 변환하는 것(가령, 오일러 각 및/또는 쿼터니언의 계산을 위한 좌표계 변환), 컴퓨팅 디바이스(270)로 데이터를 송신하는 것 등등에 더하여, 앞서 언급된 바와 같은 센서 캘리브레이션을 수행할 수 있다. 그러한 캘리브레이션은 후술되는 제스처 인식 모듈(280)에서 사용되는 CNN(281)을 위한 훈련 데이터 및 테스트 데이터의 수집뿐만 아니라, 실제 제스처 인식에서의 사용을 위한 데이터의 감지를 위해서도 수행될 수 있다.

[0044] 몇몇 실시예에서, 제어기(230)는 MCU(231)와 여러 IMU(221-1, 221-2, ..., 222-5) 간의 통신을 가능하게 하는 멀티플렉서(multiplexer)(232)를 더 포함할 수 있다. 앞서 언급된 바와 같이, 몇몇 실시예에서 IMU(221)는 I2C 통신을 사용하여 MCU(231)와 통신할 수 있으나, IMU(221-1, 221-2, ..., 221-5)로서 사용될 수 있는 통상적인 상용 IMU는 동일한 주소 값을 갖도록 구성될 수 있다. 이러한 상황에서, I2C 버스 상에서 MCU(231)와 여러 IMU(221-1, 221-2, ..., 221-5)와의 통신을 가능하게 하기 위해서 멀티플렉서(232)는 여러 IMU(221-1, 221-2, ..., 221-5)에 각자의 주소 값을 할당하여 주소 값의 충돌을 피할 수 있다.

[0045] 몇몇 실시예에서, 비록 도 2a에 도시되지 않았으나, 제어기(230)는 추가적인 컴포넌트를 더 포함할 수 있다. 예를 들어, 제어기(230)는 진동 모터(vibration motor), 압전 소자(piezoelectric element) 등등과 같은 촉각적 신호 생성기(haptic signal generator)를 더 포함할 수 있다. 촉각적 신호 생성기는 촉각적 피드백(haptic feedback)을 제공할 수 있다. 또한, 제어기(230)는 열 자극을 통해 피드백을 제공하는 히팅 패브릭(heating fabric)(예컨대, 펠티에(Peltier) 소자와 같은 열전소자를 포함함)을 더 포함할 수 있다. 촉각적 신호 생성기와 히팅 패브릭은, 예컨대 MCU(231)의 제어 하에, 글러브형 착용가능 디바이스(210)를 착용한 사용자를 위해 피드백을 제공할 수 있다. 예를 들어, MCU(231)는 촉각적 신호 생성기를 온(on)/오프(off)시키는 제1 온/오프 제어 신호, 히팅 패브릭을 온/오프시키는 제2 온/오프 제어 신호, 촉각적 신호 생성기에 의해 제공되는 진동의 세기를 조절하는 제1 전압 제어 신호, 히팅 패브릭에 발생하는 온도를 조절하는 제2 전압 제어 신호 등등을 출력할 수 있다. 추가적으로, MCU(231)는 소정의 신호(가령, 후술되는 바와 같이 제스처 데이터에 기반한 제스처 인식이 유보됨을 나타내는 경보 신호)를 컴퓨팅 디바이스(270)로부터 수신하는 것에 응답하여, 촉각적 신호 생성기로 하여금 적절한 진동을 수반하는 촉각적 피드백을 제공하도록 명령할 수 있다.

[0046] 예시적인 제스처 인식 시스템(200)에서, 컴퓨팅 디바이스(270)는 제어기(230)로부터의 제스처 데이터를 수신하고 이를 사용하여 글러브형 착용가능 디바이스(210)의 착용자의 손 제스처를 인식하기 위한 CNN 기반 제스처 인식 모듈인 CNN 기반 학습 및 분류 메커니즘(280)을 구현하도록 구성된다. 따라서, 예시적인 제스처 인식 시스템(200)의 성능은 단순히 IMU의 센서의 정확도에 의존하기보다는 인식 성공률의 측면에서 CNN 기반 학습 및 분류 메커니즘(280)으로부터 이로움을 얻을 수 있다.

[0047] CNN 기반 학습 및 분류 메커니즘(280)은 CNN(281)에 기반하여 제스처 데이터로부터 손 제스처를 인식할 수 있다. CNN(281)은 앞서 언급된 CNN(100)과 마찬가지 방식으로 구현될 수 있다. 예를 들어, 도 2b에 도시된 바와 같이, CNN(281)은 주어진 입력 데이터를 출력하는 입력 층(201), 활성화 함수(292)가 적용된 제1 콘볼루션 층(202), 제1 풀링 층(203), 활성화 함수(294)가 적용된 제2 콘볼루션 층(204), 제2 풀링 층(205), 활성화 함수(296)가 적용된 제1 피드포워드 신경망 층(206), 그리고 CNN(281)의 최종 출력(208)을 위한 활성화 함수(297)가 적용된 제2 피드포워드 신경망 층(207)을 포함할 수 있다. 다만, 본 개시의 범주는 이 점에 한정되지 않고, CNN(281)은 다른 실시예에서 상이한 구성(예를 들어, 콘볼루션 층, 풀링 층 및 피드포워드 신경망 층이 단 하나 있음)을 가질 수 있다는 점에 유의하여야 한다. 또한, 콘볼루션 층(202, 204)과 피드포워드 신경망 층(206,

207) 각각의 활성화 함수로서 임의의 적합한 함수가 사용될 수 있는데, 한정이 아니라 예시로서, 제1 콘볼루션 층(202), 제2 콘볼루션 층(204) 및 제1 피드포워드 신경망(206)(의 유닛들)의 활성화 함수로서 시그모이드 함수 또는 ReLU 함수가 사용될 수 있고, 제2 피드포워드 신경망 층(207)(의 유닛들)의 활성화 함수로서 소프트맥스 함수가 사용될 수 있다. 나아가, 풀링 층(203, 205)은 적절한 풀링(예컨대, 최대 풀링)을 수행할 수 있다.

[0048] 그러한 CNN(281)을 사용함으로써, CNN 기반 학습 및 분류 메커니즘(280)은 손 제스처에 관해 센서 모듈(220)에 의해 감지된 데이터를 포함하는 제스처 데이터로부터 특징을 추출하여 손 제스처를 어떤 손 제스처 부류(class)로 인식하는 방식으로 기계 학습과 이에 따른 제스처 분류를 수행할 수 있다. 예를 들어, CNN 기반 학습 및 분류 메커니즘(280)은 글러브형 착용가능 디바이스(210)를 착용한 사용자의 감지된 손 제스처를 CNN(281)으로부터의 출력(208)에 따라 복수의 사전정의된 손 제스처 부류 중 하나로 분류할 수 있다. 나아가, CNN 기반 학습 및 분류 메커니즘(280)은 CNN(281)의 출력(208)에 기반하여, 그리고/또는 출력(208)을 위해 CNN(281)에 의해 계산된 소정의 값에 기반하여, 글러브형 착용가능 디바이스(210)의 착용자의 감지된 손 제스처의 인식을 유보하기로 판정할 수 있다. 예로서, CNN(281)의 최종 출력(208)을 위한 피드포워드 신경망 층(207)은 n개의 손 제스처 부류를 각각 나타내는 n개의 유닛을 포함하고, 입력이 x_k인 각 유닛의 출력 y_k를 위해 다음 식에서 주어진 바와 같이 소프트맥스 함수가 피드포워드 신경망 층(207)의 활성화 함수(297)로 사용될 수 있다(즉, 출력 층(207)은 사용자의 손 제스처를 복수의 손 제스처 부류 중 하나로 분류하기 위한 출력을 CNN(281)의 최종 출력으로서 제공하는 소프트맥스 층일 수 있음).

수학식 1

[0049] $$y_k = \frac{e^{x_k}}{\sum_{i=1}^{n} e^{x_i}}$$

[0050] CNN 기반 학습 및 분류 메커니즘(280)은 n개의 유닛들의 출력들 중 최대 출력이 어떤 임계 값 이상인 경우, 또는 최대 출력 및 그 출력들 중 다른 출력(가령, 2번째로 큰 출력) 간의 차이가 어떤 임계 값 이상인 경우, 글러브형 착용가능 디바이스(210)의 착용자의 손 제스처를 최대 출력을 제공하는 유닛이 나타내는 손 제스처 부류로 분류할 수 있다. 그렇지 않은 경우, CNN 기반 학습 및 분류 메커니즘(280)은 글러브형 착용가능 디바이스(210)를 착용한 사용자의 손 제스처의 인식을 유보하기로 판정할 수 있다. 필요에 따라, CNN 기반 학습 및 분류 메커니즘(280)은 제스처 데이터로부터의 손 제스처의 인식이 유보됨을 나타내는 경보 신호가 컴퓨팅 디바이스(270)로부터 제어기(230)(특히, MCU(231))에 송신되도록 할 수 있다. 앞서 언급된 바와 같이 이러한 경보 신호에 응답하여 촉각적 피드백이 제공될 수 있으므로, 글러브형 착용가능 디바이스(210)를 착용한 사용자가 더욱 명확한 손 제스처를 행하도록 유도하는 것이 가능하다.

[0051] 추가적으로, CNN 기반 학습 및 분류 메커니즘(280)은 CNN(281)의 입력 데이터에 대해 전처리(preprocessing)를 수행하도록 구성된 전처리기(282)를 포함할 수 있다. 특히, 전처리기(282)는 CNN(281)을 위한 훈련 데이터와 테스트 데이터는 물론 실제 제스처 인식에서의 사용을 위해 감지된 데이터와 같은 입력 데이터에 대해 동일한 전처리를 수행할 수 있다. 몇몇 실시예에서, 전처리기(282)는 CNN 기반의 학습을 방해할 수 있는 어떤 경향을 제거하기 위한 데이터 정규화를 수행할 수 있다. 특히, IMU(211)는 구동될 때마다 초기에 감지되는 데이터가 급변할 가능성이 크므로, 이러한 정규화는 유익할 것이다. 예를 들어, 전처리기(282)는 다음과 같이 최소-최대 정규화(min-max normalization)를 수행할 수 있다.

수학식 2

[0052] $$V_i' = \frac{V_i - min_A}{max_A - min_A}(max_A' - min_A')$$

[0053] 여기서 V_i는 배치 학습(batch learning)을 위한 원래의 입력 데이터의 샘플의 i번째 성분이고, max_A 및 min_A는 각각 원래의 입력 데이터 전체에서의 해당 성분의 최대값 및 최소값이며, V_i'는 정규화된 입력 데이터 샘플의 i번째 성분이고, max_A' 및 min_A'는 각각 정규화된 입력 데이터 전체에서의 해당 성분의 사전정의된 최대값 및

사전정의된 최소값이다.

[0054] 성능 평가

[0055] 도 3은 본 발명의 실시예에 따른 제스처 인식 시스템에 의해 수행될 수 있는 예시적인 프로세스의 흐름도이다. 예시적인 프로세스(300)에 따르면, 우선 제스처 인식 시스템(200)의 글러브형 착용가능 디바이스(210)를 사용자가 착용한 후, 제스처 인식 시스템(200)의 글러브형 착용가능 디바이스(210)는 동작(310)에서 센서 캘리브레이션을 수행한다. 동작(320)에서, 사용자가 (가령, 평면이나 공중에 숫자나 문자를 쓰는) 제스처를 취하는 동안, 제스처 인식 시스템(200)의 글러브형 착용가능 디바이스(210)는 사용자의 제스처에 관한 감지를 통해 제스처 데이터를 획득한다. 동작(320)은 여러 번 반복될 수 있다. 동작(320)에서 획득된 제스처 데이터는 동작(330)에서 제스처 인식 시스템(200)의 컴퓨팅 디바이스(270)에 의해 CNN(281)을 위한 훈련 데이터로서 사용될 수 있다. 훈련(330) 후에, 제스처 인식 시스템(200)은 다음과 같이 제스처 인식을 수행할 수 있다. 제스처 인식 시스템(200)의 글러브형 착용가능 디바이스(210)는 CNN(281)의 훈련이 완료된 후에 초기화가 다시 필요할 수 있다. 이에 따라, 글러브형 착용가능 디바이스(210)는 동작(340)에서 센서 캘리브레이션을 다시 수행할 수 있다. 동작(350)에서, 제스처 인식 시스템(200)의 컴퓨팅 디바이스(270)는 제스처 데이터를 입력 받는다. 입력 데이터는 글러브형 착용가능 디바이스(210)에 의해 사용자의 제스처에 관해 감지된 데이터를 포함할 수 있다. 동작(360)에서, 제스처 인식 시스템(200)의 컴퓨팅 디바이스(270)는 동작(350)에서 제공된 제스처 데이터로부터 CNN 기반의 제스처 인식을 수행한다. 훈련 데이터에 대해 전처리가 수행된 경우, 입력 데이터도 마찬가지 방식으로 전처리될 수 있다.

[0056] 예시적인 제스처 인식 시스템(200)의 성능을 평가하기 위해, 도 3의 프로세스(300)가 수행될 수 있다. 이를 위해, 동작(360)에서 컴퓨팅 디바이스(270)에 테스트 데이터가 제공되고 그 테스트 데이터로부터의 CNN 기반 제스처 인식이 테스트될 수 있다.

[0057] 이제, 성능 평가를 위한 예시적인 시나리오(이하에서 "평가 시나리오"로 지칭될 수 있음)에서 제스처 인식 시스템(200)이 어떻게 구현되고 시험되는지 논의된다.

[0058] 평가 시나리오에서, 데이터 글러브(다시 말해, 글러브형 착용가능 디바이스)(210)는 5개의 손가락 형상 부분에 5개의 IMU(221-1, 221-2, ..., 221-5)가 각각 장착된다. 데이터 글러브(210)의 IMU(221)로는, 3축 자이로스코프 L3G4200D, 3축 가속도계 ADXL345 및 3축 지자기 센서 HMC5883L이 보드 상에 포함된, Arduino 호환가능한 센서 모듈인 GY-80이 사용된다. 표 1은 평가 시나리오에서의 사용을 위한 IMU(221)의 사양을 보여준다.

표 1

[0059]

센서	자력계	가속도계	자이로스코프
인터페이스	I2C	I2C	I2C
측정 범위	1G ~ 8G	$2\times g$ ~ $16\times g$	250 deg/s ~ 2000 deg/s
출력 레이트	12bit	13bit	16bit
샘플링 레이트	75Hz ~ 160Hz	6Hz ~ 3200Hz	100Hz ~ 800Hz

[0060] 평가 시나리오에서, 가속도는 가속도계에 의해 100 Hz의 샘플링 레이트(sampling rate)로, 각속도는 자이로스코프에 의해 200 Hz의 샘플링 레이트로, 그리고 지자기는 지자기 센서에 의해 75 Hz의 샘플링 레이트로 측정된다. IMU(221)의 센서들의 신뢰성은 정지된 자세의 제스처에 대한 데이터의 표준 편차로부터 확인될 수 있다.

[0061] 평가 시나리오에서, 아두이노 나노(Arduino Nano)가 부착되어 MCU(231)로 사용된다. 표 2는 평가 시나리오에서의 사용을 위한 MCU(231)의 사양을 도시한다.

표 2

마이크로제어기	ATmega328
아키텍처 유형	AVR 아키텍처
동작 전압	5V
플래시 메모리	32KB
SRAM	2KB
클록 속도	16MHz
아날로그 I/O 핀	8
Digital I/O 핀	22
PCB 크기	18×45mm
무게	7g

[0062]

[0063] 평가 시나리오에서, MCU(231)는 I2C 통신을 사용하여 IMU(221)와 통신하며, 각 IMU(221)로의 주소 할당을 위한 I2C 멀티플렉서가 구비된다.

[0064] 평가 시나리오에서, MCU(231)는 직렬 통신(serial communication) 방식으로 컴퓨팅 디바이스(270)와 통신한다. 특히, MCU(231)는 IMU(221)로부터의 감지된 데이터에 기반한 제스처 데이터를 직렬 케이블(250) 상에서 컴퓨팅 디바이스(270)에 송신한다. 평가 시나리오에서, 사용자 제스처를 검출하기 위한 각 시점에서, MCU(231)는 각각의 IMU(221)와 관련하여 아래의 표 3과 같이 주 처리 데이터와 보조 처리 데이터를 포함하여 제스처 데이터를 구성한다. 도 4는 평가 시나리오에서의 사용을 위한 포맷(format)으로 된 일련의 예시적인 제스처 데이터를 보여준다. 도 4에 도시된 제스처 데이터는 속성-값 쌍으로 된 JSON 데이터 포맷을 갖는다. 이러한 포맷은 프로그래밍 언어와 플랫폼에 독립적이어서 서로 다른 시스템 간의 객체 교환에 유용하고, 많은 웹 브라우저에서 지원되며, 가독성도 지니고 있다. 예를 들어, 도 4의 "Sensor:" 바로 뒤의 값은 5개의 IMU(221-1, 221-2, ..., 221-5) 중 하나의 식별자이고, 이어서 그 IMU와 연관된 제스처 데이터가 있음이 이해될 것이다.

표 3

구분	표시	설명
주 처리 데이터	Mag_x	자력계의 x축 데이터
	Mag_y	자력계의 y축 데이터
	Mag_z	자력계의 z축 데이터
	Acc_x	가속도계의 x축 데이터
	Acc_y	가속도계의 y축 데이터
	Acc_z	가속도계의 z축 데이터
	Gyro_x	자이로스코프의 x축 데이터
	Gyro_y	자이로스코프의 y축 데이터
	Gyro_z	자이로스코프의 z축 데이터
보조 처리 데이터	Qn_0	쿼터니언의 qx 데이터
	Qn_1	쿼터니언의 qy 데이터
	Qn_2	쿼터니언의 qz 데이터
	Qn_3	쿼터니언의 qw 데이터
	Yaw	요(yaw) 데이터
	pitch	피치(pitch) 데이터
	roll	롤(roll) 데이터

[0065]

[0066] 표 3 및 도 4에서 볼 수 있는 바와 같이, 평가 시나리오에서 제스처 데이터는 16개의 출력 변수에 대해 구성되는데, 주 처리 데이터로서 9개의 출력 변수를, 그리고 보조 처리 데이터로서 7개의 출력 변수를 포함한다. 구체적으로, 주 처리 데이터는 3개 축 각각에 대해 IMU(221)에 의해 감지된 지자기, 가속도 및 각속도 값을 나타낸다. 보조 처리 데이터는 주 처리 데이터에 기반하여 계산된 것인데, 3차원 공간에서의 회전을 표현하기 위한 4개의 원소 qx, qy, qz, qw로 이루어진 쿼터니언과, 3개의 오일러 각도(롤, 피치 및 요)를 포함한다.

[0067] 각각의 IMU(221)와 관련하여, 제스처 데이터는 몇 개의 시점 각각에 대한 출력 변수 값이 취합된 포맷으로 구성될 수 있다. 평가 시나리오에서, MCU(231)는 앞서 언급된 16개의 출력 변수의 값을 획득하는 것을 16개의 시점 각각에서 수행하고, 수신된 출력 변수 값들을 16×16 포맷의 데이터세트로 구성하여 컴퓨팅 디바이스(270)로 송신한다.

[0068] 평가 시나리오에서, 센서 캘리브레이션 후, 데이터 글러브(210)의 착용자가 자기 손가락을 사용하여 0부터 9까지의 숫자 중 하나를 수기하는(handwriting) 제스처를 취하는 방식으로 각 숫자에 대해 50개의 16×16 훈련 데이터세트와 20개의 16×16 테스트 데이터세트가 생성된다. 예를 들어, 도 5는 평가 시나리오에서 숫자 8을 손으로 쓰는 제스처의 4개의 데이터세트의 출력 변수 Mag_x의 그래프이다. 이들 데이터세트 1 내지 데이터세트 4는 대체로 비슷한 특성을 보여주되, 서로에 대해 시간에 따라 약간 이동되어 있을 뿐이다. 이런 점에서, 위치와 무관한 특징을 추출하도록 CNN이 사용될 수 있음이 이해될 것이다.

[0069] 평가 시나리오에서, CNN 기반 학습 및 분류 메커니즘(280)은 풀링 층(203, 205)이 최대 풀링 층(즉, 풀링 층(203, 205)은 최대 풀링을 수행함)이고 출력 층(207)이 소프트맥스 층(즉, 출력 층(207)의 활성화 함수(297)는 소프트맥스 함수임)인 CNN(281)을 사용한다. 또한, 평가 시나리오에서, CNN(281)의 피드포워드 신경망 층(206,

207)은 아핀 층(Affine layer)과 같은 완전 연결 층이다. 후술되는 바와 같이, 평가 시나리오에서 CNN(281)의 학습과 관련된 몇몇 사항은 비교를 위해 조정될 수 있다.

[0070] 평가 시나리오에서, 데이터세트에 대한 전처리를 수행하는 경우(이하에서 "전처리 있는 평가"로 지칭될 수 있음)와 그렇지 않은 경우(이하에서 "전처리 없는 평가"로 지칭될 수 있음)가 비교된다. 평가 시나리오를 위한 전처리에서, 데이터세트의 16개 성분, 곧 출력 변수 각각의 값은 최소-최대 정규화로 전처리되어 0부터 1000까지의 범위로 스케일링된다. CNN(281)의 학습을 위해, 전처리 있는 평가에 대해서든 전처리 없는 평가에 대해서든, 학습 횟수는 총 100,000회이고, 배치(batch) 크기는 30이며, 학습률(learning rate)은 0.0001로 설정되고, 활성화 함수(292, 294, 296)로 시그모이드 함수가 사용된다. 도 6은 전처리 있는 평가의 오차 갱신과 전처리 없는 평가의 오차 갱신을 보여준다. 도 6의 좌측 그래프는 전처리 있는 평가에서의 교차 엔트로피(cross entropy) 오차를 나타낸다. 도시된 바와 같이, 전처리 있는 평가에서, 학습 횟수가 증가함에 따라 오차는 계속해서 갱신되고, 학습 횟수가 100,000회에 접근함에 따라 오차는 0에 접근한다. 도 6의 우측 그래프는 전처리 없는 평가에서의 교차 엔트로피 오차를 나타낸다. 도시된 바와 같이, 전처리 없는 평가에서, 초기 학습 동안 오차가 갱신되나, 학습 횟수가 80,000회를 넘어감에 따라 오차는 거의 갱신되지 않는다. 도 7은 0부터 9까지의 숫자 각각을 수기하는 제스처에 대해 전처리 있는 평가가 내는 인식 성공률과 전처리 없는 평가가 내는 인식 성공률을 보여준다. 도시된 바와 같이, 전처리 없는 평가의 인식 성공률이 더 낮은데, 전처리되지 않은 데이터세트로는 CNN에 의해 전역적인 특징이 추출되기 어렵기 때문이라고 이해된다. 표 4는 도 7과 관련된 몇 가지 값을 보여준다. 아래의 표 4에서 볼 수 있듯이, 전처리를 통해 제스처 인식의 성공률이 15% 증가된다.

표 4

[0071]

구분	전처리 있는 평가	전처리 없는 평가
학습 횟수	100,000	100,000
성공률	97%	82%
학습 소요 시간	3070초	2873초
최저 성공률	85%	35%
인식이 가장 안 된 숫자	2	0

[0072] 평가 시나리오에서, 학습 횟수에 따른 제스처 인식의 성공률이 비교된다. 이를 위해, 다음의 세 가지 학습 횟수가 고려된다: 1,000회; 10,000회; 및 100,000회. 또한, 그러한 비교에서 전처리가 수반된다. CNN(281)의 학습을 위한 나머지 사항과 관련하여, 앞서 언급된 것과 동일하게, 배치 크기는 30이고, 학습률은 0.0001로 설정되며, 활성화 함수(292, 294, 296)로 시그모이드 함수가 사용된다. 도 8은 학습 횟수에 따른 오차 갱신을 보여준다. 도 8의 좌측 그래프에서 볼 수 있듯이, 학습 횟수가 1,000회에 이를 때까지 오차 갱신은 미미한바, 이 학습 횟수는 불충분하다. 도 8의 중앙 그래프는 학습 횟수가 10,000회에 다다르면서 오차가 갱신되어 많이 줄어들지만, 갈지자(zigzag)로 오차 갱신이 계속해서 일어나고 있음을 보여준다. 도 8의 우측 그래프(이는 도 6의 좌측 그래프와 같음)에 도시된 바와 같이, 학습 횟수가 100,000회까지 증가함에 따라 오차 갱신은 어느 정도 꾸준히 일어나긴 하지만, 상당량의 오차 갱신은 학습 횟수가 20,000회에 도달하기 전에 일어난다. 도 9는 총 학습 횟수에 따른 제스처 인식의 성공률을 보여준다. 도시된 바와 같이, 0부터 9까지의 숫자 각각을 수기하는 제스처의 대부분은 총 학습 횟수가 1000회일 때 인식 성공률이 낮다. 총 학습 횟수가 10,000회인 경우, 대부분의 제스처에 대해 인식 성공률이 높지만, 숫자 1과 숫자 7과 관련한 인식 성공률은 낮은데, 그 두 숫자 간의 특성 차이가 작기 때문이라고 여겨진다. 표 5는 도 9과 관련된 몇 가지 값을 보여준다.

표 5

구분	학습 횟수 = 1,000회	학습 횟수 = 10,000회	학습 횟수 = 100,000회
성공률	49%	86.5%	97%
학습 소요 시간	31초	314초	3070초
최저 성공률	25%	55%	85%
인식이 가장 안 된 숫자	9	1	2

[0073]

[0074] 평가 시나리오에서, 활성화 함수(292, 294, 296)로서 시그모이드 함수가 사용된 경우(이하에서 "시그모이드 함수 평가"로 지칭될 수 있음)와 ReLU 함수가 사용된 경우(이하에서 "ReLU 함수 평가"로 지칭될 수 있음)가 비교된다. CNN(281)의 학습을 위해, 학습 횟수는 10,000회이고, 배치 크기는 30이며, 학습률은 0.0001로 설정되고, 전처리가 수반된다. 도 10은 시그모이드 함수 평가의 오차 갱신과 ReLU 함수 평가의 오차 갱신을 보여준다. 도 10의 좌측 그래프에서 볼 수 있듯이, ReLU 함수 평가에서 오차 갱신은 학습 횟수가 2,000회에 도달하기 전에 0에 가깝게 된다. 도 10의 우측 그래프(이는 도 8의 중앙 그래프와 같음)는 시그모이드 함수 평가에서 학습 횟수가 10,000회에 이를 때까지 오차 갱신이 계속됨을 보여준다. 도 11은 시그모이드 함수 평가와 ReLU 함수 평가에서의 인식 성공률을 보여준다. 도시된 바와 같이, 0부터 9까지의 숫자 각각을 수기하는 제스처의 대부분에 대해서, ReLU 함수 평가가 내는 인식 성공률이 더 높지만, 숫자 4, 5, 6의 경우는 시그모이드 함수 평가가 내는 인식 성공률이 더 높다. 상대적으로 복잡한 특징을 갖는 제스처의 인식 성공률은 시그모이드 함수가 더 나은 인식 성공률을 보인다고 해석될 수 있다. 그러나, 전체적인 성공률의 관점에서 보면, ReLU 함수 평가의 성공률이 90%여서, 시그모이드 함수 평가의 성공률보다 3.5% 더 크다. 또한, ReLU 함수 평가가 시그모이드 함수 평가보다 학습에 더 적은 시간을 요구한다. 특히, ReLU 함수는 0 이하의 값을 0으로 다루므로, ReLU 함수 평가가 계산량 면에서 시그모이드 함수 평가에 비해 유리하다고 여겨진다. 표 6은 도 11과 관련된 몇 가지 값을 보여준다.

표 6

구분	시그모이드 함수 평가	ReLU 함수 평가
성공률	86.5%	90%
학습 소요 시간	314초	243초
최저 성공률	55%	70%
인식이 가장 안 된 숫자	1	2

[0075]

[0076] 예시적인 컴퓨팅 환경

[0077] 몇몇 실시예에서, 컴퓨팅 디바이스(270) 및/또는 제어기(230)와 같은 처리 장치는 임의의 적합한 유형의 컴퓨터일 수 있는데, 이는 하나 이상의 프로세서, 프로세서에 의해 판독가능한 컴퓨터 판독가능 저장 매체 및 다양한 주변기기를 포함할 수 있다. 예를 들어, 프로세서는 디지털 신호 프로세서(Digital Signal Processor: DSP), 중앙 처리 유닛(Central Processing Unit: CPU), 그래픽 처리 유닛(Graphics Processing Unit: GPU), 프로세서 코어, 마이크로프로세서, 마이크로제어기, 필드 프로그램가능 게이트 어레이(Field-Programmable Gate Array: FPGA), 애플리케이션 특정 집적 회로(Application Specific Integrated Circuit: ASIC), 무선 주파수 집적 회

로(Radio-Frequency Integrated Circuit: RFIC), 다른 하드웨어 및 로직 회로, 또는 이의 임의의 적합한 조합을 포함할 수 있다. 컴퓨터 판독가능 저장 매체에는, 프로세서에 의해 실행되는 경우 컴퓨터로 하여금 본 발명의 실시예에 따라 몇몇 동작을 수행하게 하는 컴퓨터 실행가능 명령어가 저장되어 있을 수 있다. 예를 들어, 컴퓨터 판독가능 저장 매체는 판독 전용 메모리(Read-Only Memory: ROM), 랜덤 액세스 메모리(Random-Access Memory: RAM), 휘발성(volatile) 메모리, 비휘발성(non-volatile) 메모리, 착탈가능(removable) 메모리, 비착탈가능(non-removable) 메모리, 하드 디스크, 플래시(flash) 메모리, 자기 디스크 저장 매체, 광 디스크 저장 매체, 다른 저장 디바이스 및 저장 매체, 또는 이의 임의의 적합한 조합을 포함할 수 있다. 예를 들어, 주변기기는 다양한 입출력(Input/Output: I/O) 디바이스, 가령 마우스와 같은 포인팅 디바이스, 키보드, 키패드, 마이크, CD/DVD 플레이어, 액정 디스플레이(Liquid Crystal Display: LCD), 터치 감지 디스플레이(touch-sensitive display), 스피커, 프린터, 통신 인터페이스 카드, 사운드 카드 등등을 포함할 수 있다.

[0078] 예시적인 실시예는 본 문서에 기술된 동작, 기법, 프로세스, 또는 이의 어떤 양상이나 부분이 체현된 컴퓨터 프로그램을 포함하는 컴퓨터 판독가능 저장 매체로서 구현될 수 있다. 이러한 컴퓨터 판독 가능 저장 매체에는 프로그램 명령어, 로컬 데이터 파일, 로컬 데이터 구조 등등이 단독으로 또는 조합되어 포함될 수 있다. 개시된 동작, 기법, 프로세스, 또는 이의 어떤 양상이나 부분을 구현하거나 이용할 수 있는 프로그램은 컴퓨터에 의해 실행될 수 있는 어떤 유형의 (가령, 컴파일형(compiled) 또는 해석형(interpreted)) 프로그래밍 언어, 예컨대, 어셈블리(assembly), 기계어(machine language), 프로시저형(procedural) 언어, 객체지향(object-oriented) 언어 등등으로 구현될 수 있고, 하드웨어 구현과 조합될 수 있다. 용어 "컴퓨터 판독가능 저장 매체"는, 컴퓨팅 장치에 의한 실행을 위한 명령어(실행 시에 컴퓨팅 장치로 하여금 개시된 기법을 수행하게 함)를 저장할 수 있고, 그러한 명령어에 의해 사용되거나 이와 연관된 데이터 구조를 저장할 수 있는 임의의 매체를 포함할 수 있다. 컴퓨터 판독가능 저장 매체의 예는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체, CD-ROM, DVD와 같은 광 기록 매체, 플롭티컬 디스크와 같은 자기-광 매체, 그리고 ROM, RAM, 플래시 메모리, 솔리드 스테이트(solid-state) 메모리와 같은 메모리 디바이스를 포함하되, 이에 한정되지 않는다.

[0079] 이상에서 본 발명의 몇몇 실시예가 상세하게 기술되었으나, 이는 제한적이 아니고 예시적인 것으로 간주되어야 한다. 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자는 개시된 실시예의 세부사항에 대해 본 발명의 범주에서 벗어나지 않고서 다양한 변경이 행해질 수 있음을 이해할 것이다. 그러므로 본 발명의 범주는 설명된 실시예에 국한되어 정해져서는 안 되며, 후술하는 특허청구범위 및 그 균등물에 의해 정해져야 한다.

부호의 설명

[0080] 100: 콘볼루션 신경망

200: 제스처 인식 시스템

210: 글러브형 착용가능 디바이스

220: 감지 메커니즘

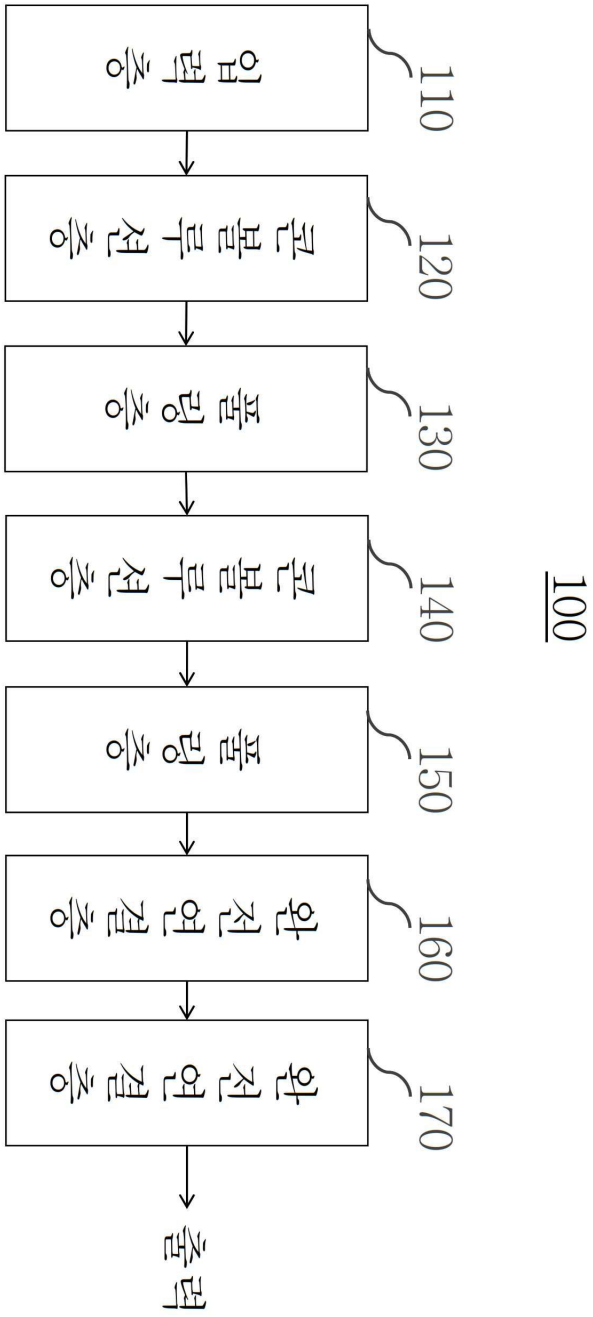
230: 제어기

270: 컴퓨팅 디바이스

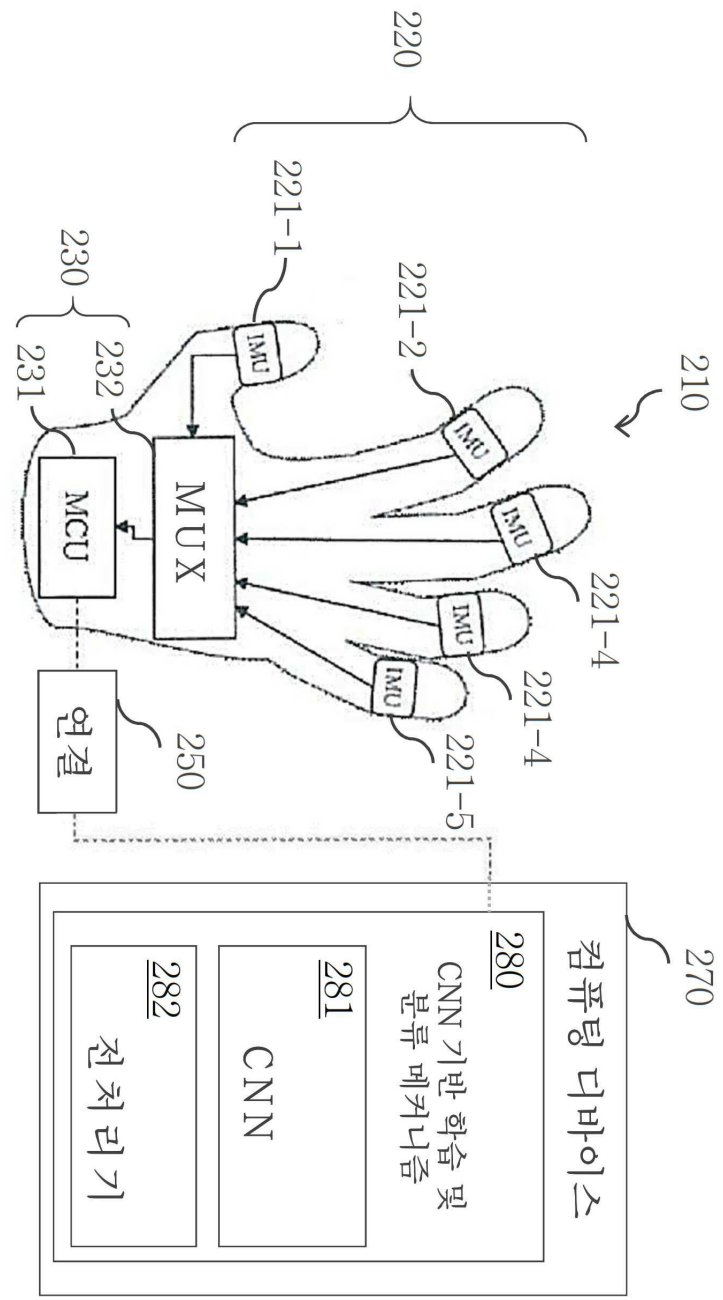
280: 콘볼루션 신경망 기반 학습 및 분류 메커니즘

도면

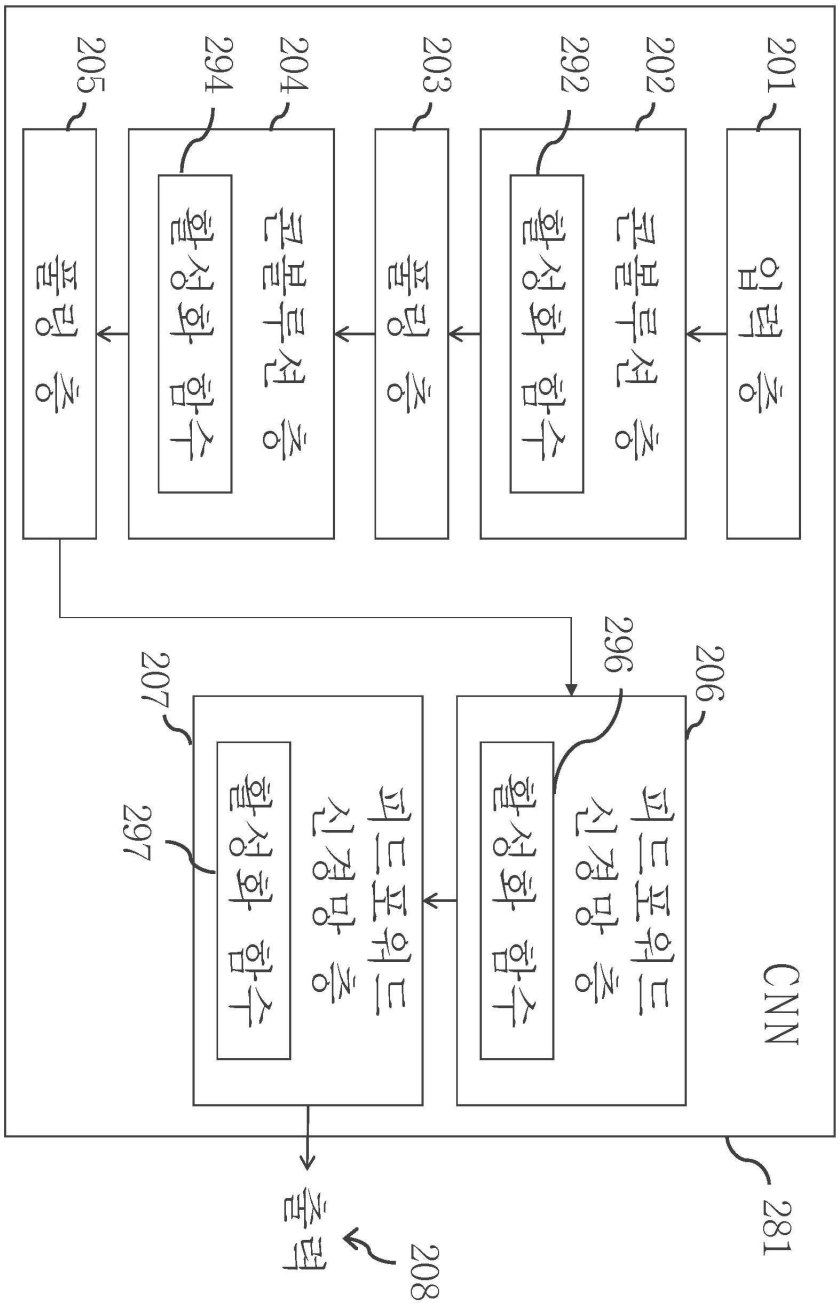
도면1


입력층
110
콘볼루션층
120
풀링층
130
콘볼루션층
140
풀링층
150
완전연결층
160
완전연결층
170
출력
100

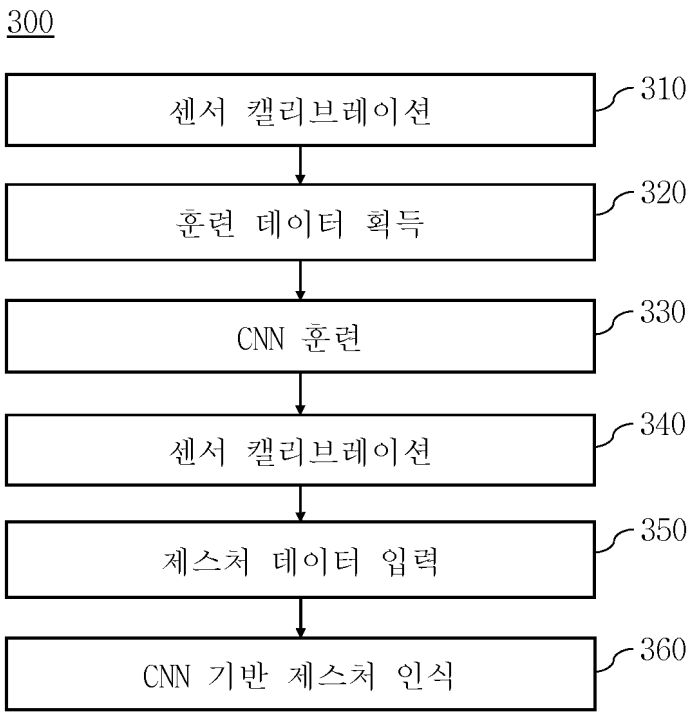
도면2a


200
210
220
221-1
221-2
221-4
221-4
221-5
230
231
232
250
270
IMU
MUX
MCU
연결
컴퓨팅 디바이스
280
CNN 기반 학습 및 분류 메커니즘
281
CNN
282
전처리기

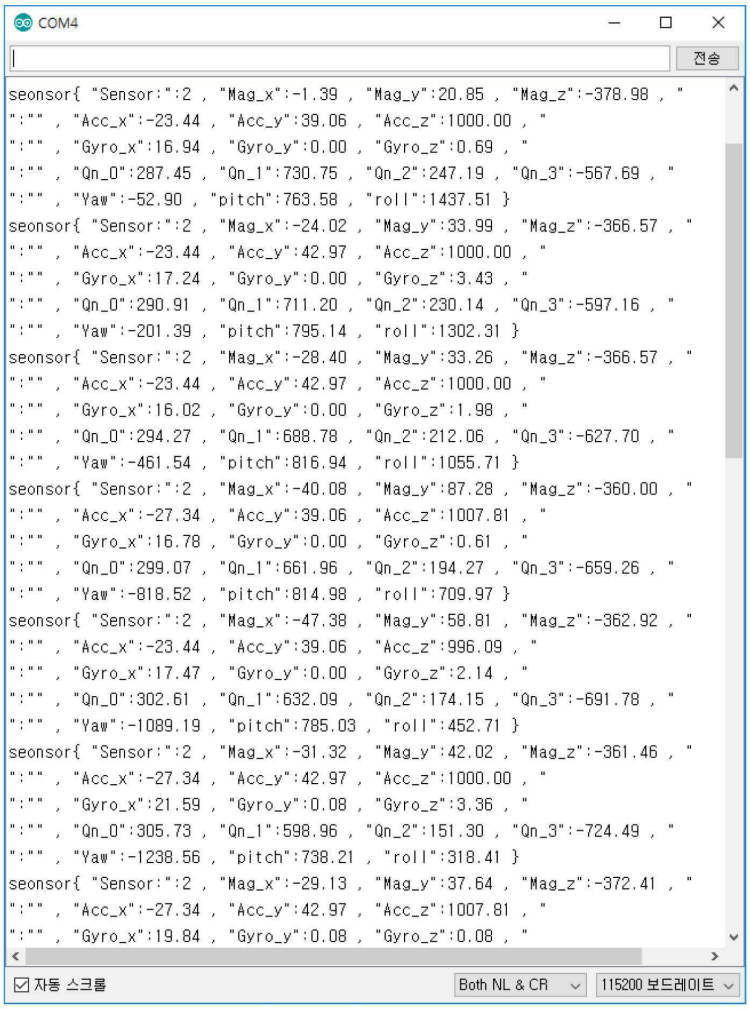
도면2b


CNN
281
201
입력 층
202
292
콘볼루션 층
활성화 함수
203
풀링 층
204
294
콘볼루션 층
활성화 함수
205
풀링 층
206
296
피드포워드 신경망 층
활성화 함수
207
297
피드포워드 신경망 층
활성화 함수
208
출력

도면3


300
센서 캘리브레이션
310
훈련 데이터 획득
320
CNN 훈련
330
센서 캘리브레이션
340
제스처 데이터 입력
350
CNN 기반 제스처 인식
360

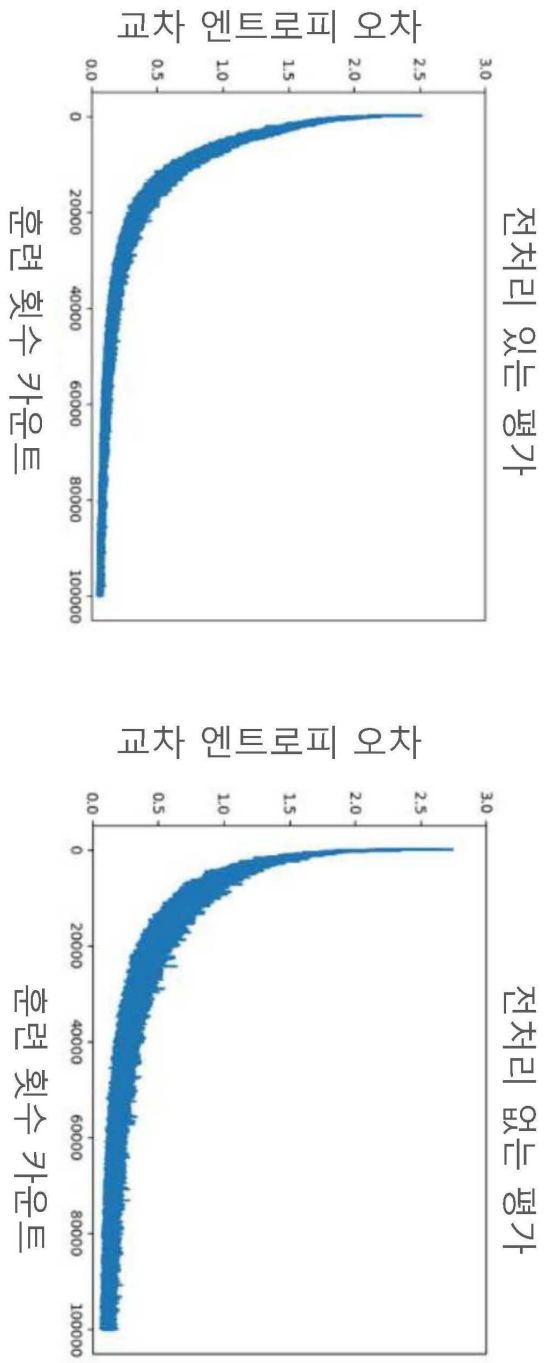
도면4


COM4
전송
seonsor{ "Sensor":2 , "Mag_x":-1.39 , "Mag_y":20.85 , "Mag_z":-378.98 , "
":"" , "Acc_x":-23.44 , "Acc_y":39.06 , "Acc_z":1000.00 , "
":"" , "Gyro_x":16.94 , "Gyro_y":0.00 , "Gyro_z":0.69 , "
":"" , "Qn_0":287.45 , "Qn_1":730.75 , "Qn_2":247.19 , "Qn_3":-567.69 , "
":"" , "Yaw":-52.90 , "pitch":763.58 , "roll":1437.51 }
seonsor{ "Sensor":2 , "Mag_x":-24.02 , "Mag_y":33.99 , "Mag_z":-366.57 , "
":"" , "Acc_x":-23.44 , "Acc_y":42.97 , "Acc_z":1000.00 , "
":"" , "Gyro_x":17.24 , "Gyro_y":0.00 , "Gyro_z":3.43 , "
":"" , "Qn_0":290.91 , "Qn_1":711.20 , "Qn_2":230.14 , "Qn_3":-597.16 , "
":"" , "Yaw":-201.39 , "pitch":795.14 , "roll":1302.31 }
seonsor{ "Sensor":2 , "Mag_x":-28.40 , "Mag_y":33.26 , "Mag_z":-366.57 , "
":"" , "Acc_x":-23.44 , "Acc_y":42.97 , "Acc_z":1000.00 , "
":"" , "Gyro_x":16.02 , "Gyro_y":0.00 , "Gyro_z":1.98 , "
":"" , "Qn_0":294.27 , "Qn_1":688.78 , "Qn_2":212.06 , "Qn_3":-627.70 , "
":"" , "Yaw":-461.54 , "pitch":816.94 , "roll":1055.71 }
seonsor{ "Sensor":2 , "Mag_x":-40.08 , "Mag_y":87.28 , "Mag_z":-360.00 , "
":"" , "Acc_x":-27.34 , "Acc_y":39.06 , "Acc_z":1007.81 , "
":"" , "Gyro_x":16.78 , "Gyro_y":0.00 , "Gyro_z":0.61 , "
":"" , "Qn_0":299.07 , "Qn_1":661.96 , "Qn_2":194.27 , "Qn_3":-659.26 , "
":"" , "Yaw":-818.52 , "pitch":814.98 , "roll":709.97 }
seonsor{ "Sensor":2 , "Mag_x":-47.38 , "Mag_y":58.81 , "Mag_z":-362.92 , "
":"" , "Acc_x":-23.44 , "Acc_y":39.06 , "Acc_z":996.09 , "
":"" , "Gyro_x":17.47 , "Gyro_y":0.00 , "Gyro_z":2.14 , "
":"" , "Qn_0":302.61 , "Qn_1":632.09 , "Qn_2":174.15 , "Qn_3":-691.78 , "
":"" , "Yaw":-1089.19 , "pitch":785.03 , "roll":452.71 }
seonsor{ "Sensor":2 , "Mag_x":-31.32 , "Mag_y":42.02 , "Mag_z":-361.46 , "
":"" , "Acc_x":-27.34 , "Acc_y":42.97 , "Acc_z":1000.00 , "
":"" , "Gyro_x":21.59 , "Gyro_y":0.08 , "Gyro_z":3.36 , "
":"" , "Qn_0":305.73 , "Qn_1":598.96 , "Qn_2":151.30 , "Qn_3":-724.49 , "
":"" , "Yaw":-1238.56 , "pitch":738.21 , "roll":318.41 }
seonsor{ "Sensor":2 , "Mag_x":-29.13 , "Mag_y":37.64 , "Mag_z":-372.41 , "
":"" , "Acc_x":-27.34 , "Acc_y":42.97 , "Acc_z":1007.81 , "
":"" , "Gyro_x":19.84 , "Gyro_y":0.08 , "Gyro_z":0.08 , "
자동 스크롤
Both NL & CR
115200 보드레이트

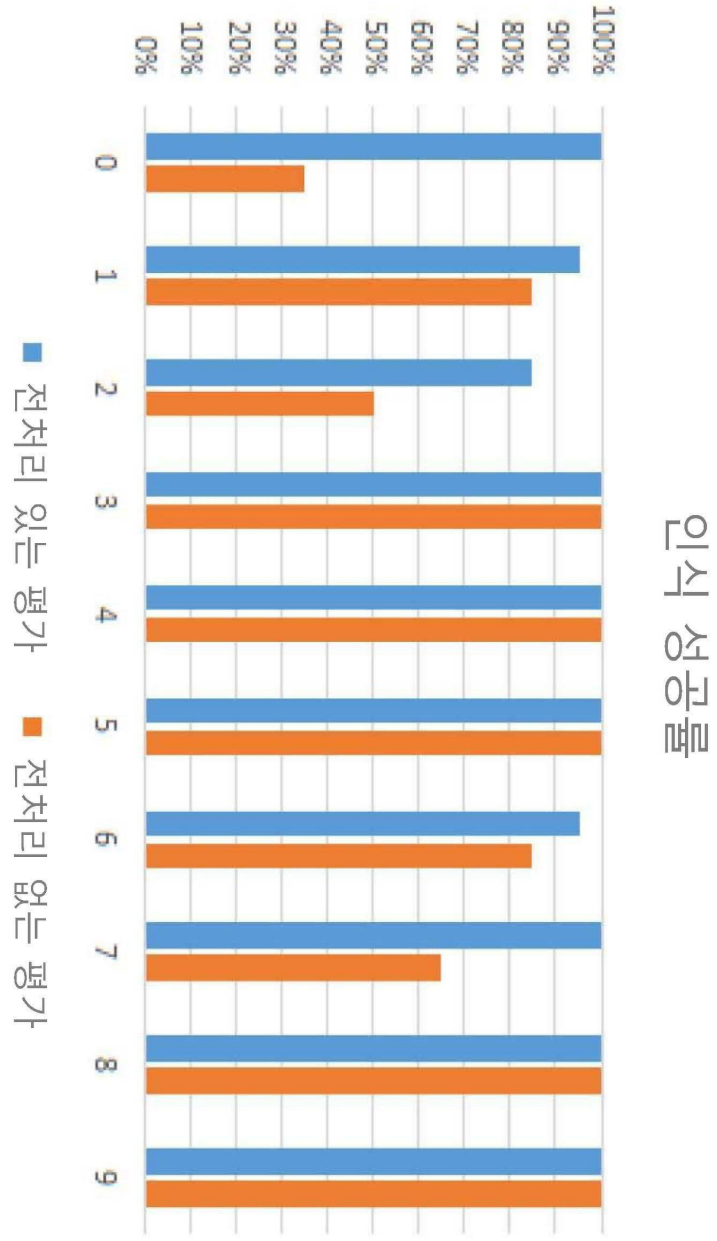
도면5


400
350
300
250
200
150
100
50
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
데이터 세트1
데이터 세트2
데이터 세트3
데이터 세트4

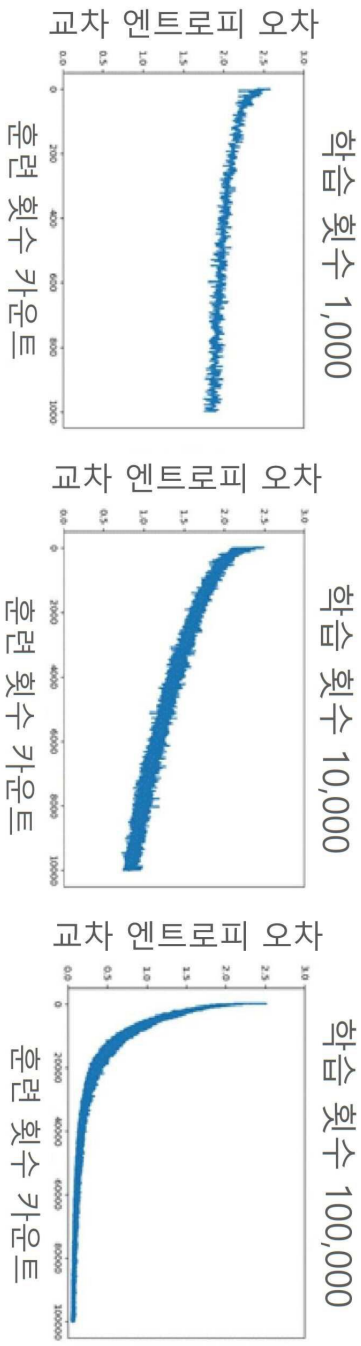
도면6



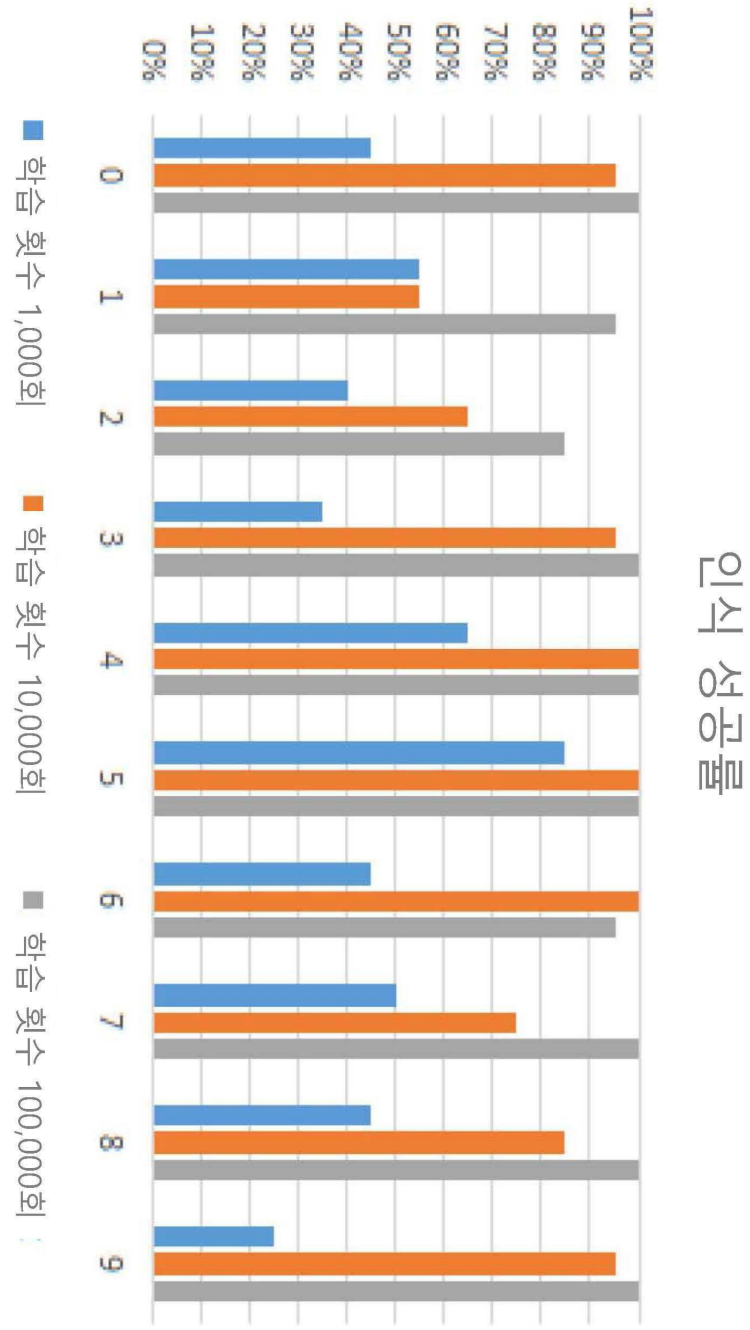
도면7


인식 성공률
100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
0
1
2
3
4
5
6
7
8
9
전처리 있는 평가
전처리 없는 평가

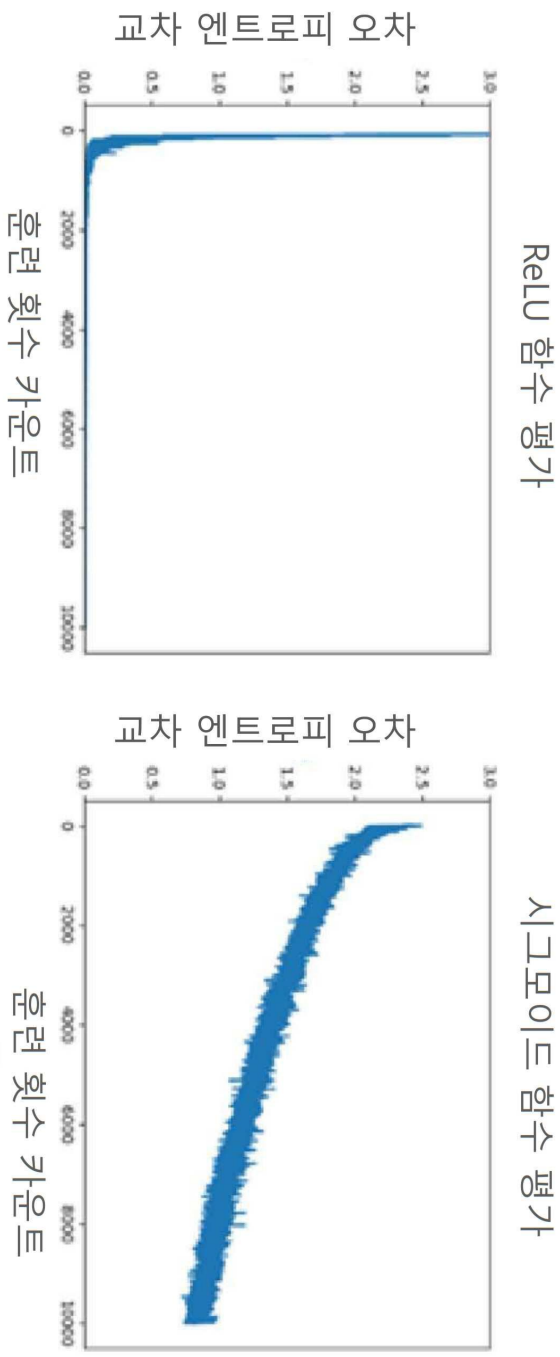
도면8


교차 엔트로피 오차
훈련 횟수 카운트
학습 횟수 1,000
교차 엔트로피 오차
훈련 횟수 카운트
학습 횟수 10,000
교차 엔트로피 오차
훈련 횟수 카운트
학습 횟수 100,000

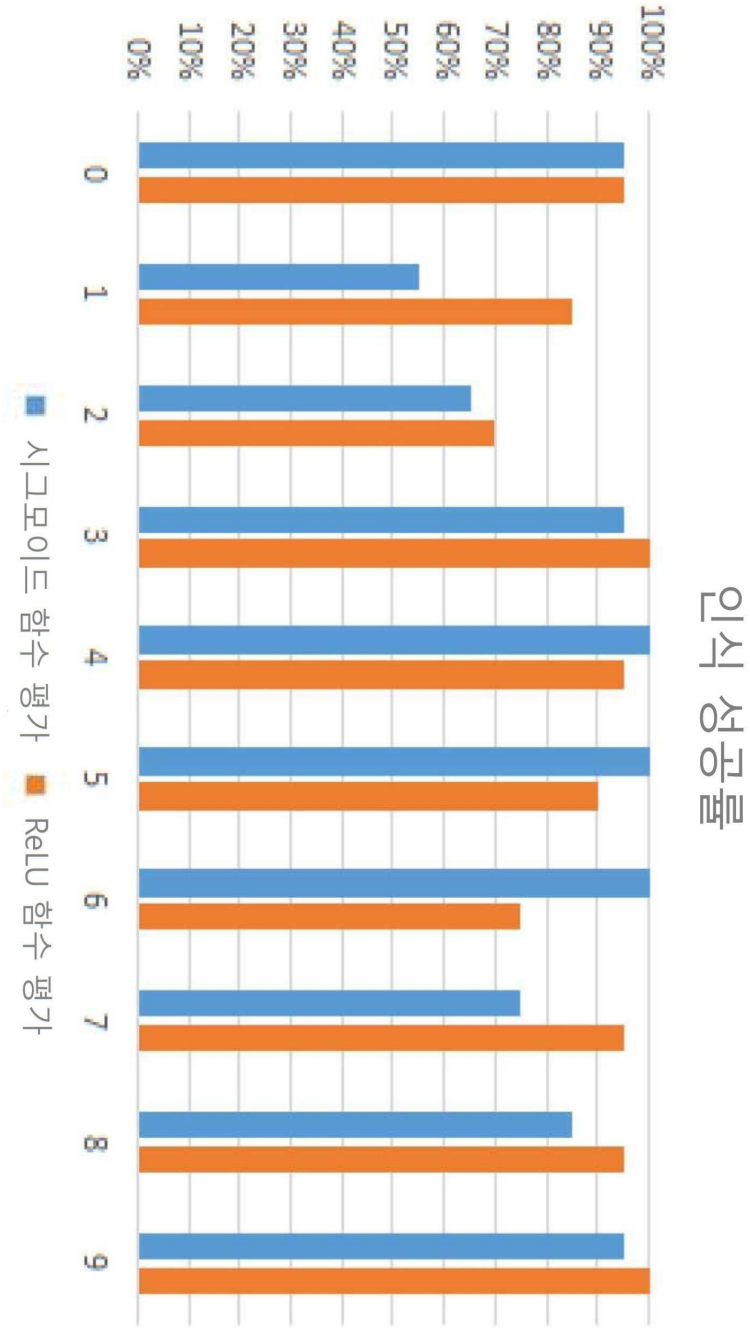
도면9


인식 성공률
100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
0
1
2
3
4
5
6
7
8
9
학습 횟수 1,000회
학습 횟수 10,000회
학습 횟수 100,000회

도면10



도면11


인식 성공률
0%
10%
20%
30%
40%
50%
60%
70%
80%
90%
100%
0
1
2
3
4
5
6
7
8
9
시그모이드 함수 평가
ReLU 함수 평가