(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2025-0032575
(43) 공개일자 2025년03월07일

(51) 국제특허분류(Int. Cl.)
A61B 5/00 (2021.01) *A61B 5/055* (2006.01)
G16H 30/20 (2018.01) *G16H 50/20* (2018.01)
G16H 50/30 (2018.01) *G16H 50/50* (2018.01)
(52) CPC특허분류
A61B 5/4088 (2013.01)
A61B 5/0042 (2013.01)
(21) 출원번호 10-2023-0115569
(22) 출원일자 2023년08월31일
심사청구일자 2024년10월18일

(71) 출원인
조선대학교산학협력단
광주광역시 동구 조선대길 126(서석동, 산학협력단)
(72) 발명자
권구락
광주광역시 서구 화운로 24 힐스테이트 303동 303호
(74) 대리인
특허법인주원

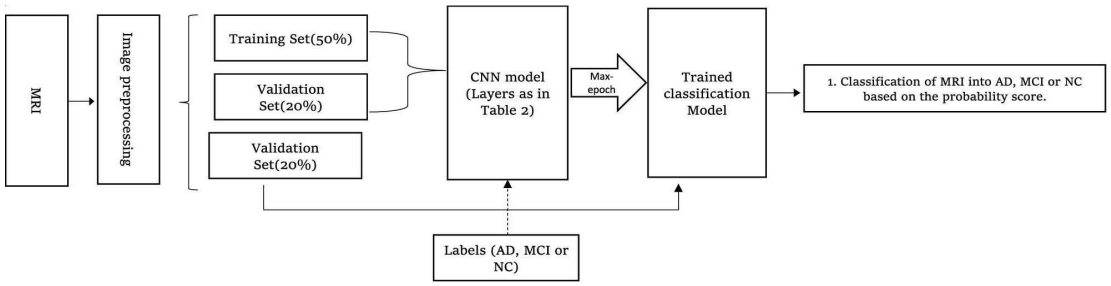
전체 청구항 수 : 총 4 항

(54) 발명의 명칭 **3차원 컨볼루션 신경망의 스케일링된 감마 탄 활성화 함수를 이용한 알츠하이머병 진행단계 분류를 위한 진단정보 제공방법**

(57) 요 약

본 발명은 3차원 컨볼루션 신경망의 스케일링된 감마 탄 활성화 함수를 이용한 알츠하이머병 진행단계 분류를 위한 진단정보 제공방법에 관한 것이다.

대 표 도



(52) CPC특허분류
A61B 5/055 (2022.01)
A61B 5/4842 (2013.01)
G16H 30/20 (2018.01)
G16H 50/20 (2018.01)
G16H 50/30 (2018.01)
G16H 50/50 (2018.01)

이 발명을 지원한 국가연구개발사업
과제고유번호 1345365105
과제번호 2021R1I1A3050703
부처명 교육부
과제관리(전문)기관명 한국연구재단
연구사업명 이공학학술연구기반구축
연구과제명 치매케어를 위한 ATN(V), MRI-PET, 커넥톰 기반 진단 지원 및 진행 예측 기술
기 여 율 1/2
과제수행기관명 조선대학교
연구기간 2023.03.01 ~ 2024.02.29
이 발명을 지원한 국가연구개발사업
과제고유번호 1425170613
과제번호 S3312710
부처명 중소벤처기업부
과제관리(전문)기관명 중소기업기술정보진흥원
연구사업명 산학협력거점형플랫폼(R&D)
연구과제명 인지 기능 저하에 따른 문진과 MRI를 이용한 자가진단 서비스
기 여 율 1/2
과제수행기관명 조선대학교산학협력단
연구기간 2022.09.01 ~ 2022.12.31

명 세 서

청구범위

청구항 1

1) 정상 대조군, 경도인지장애 및 알츠하이머병 환자의 뇌 MRI 이미지를 획득하는 단계;

2) 상기 뇌 MRI 이미지를 전처리하는 단계;

3) 상기 뇌 MRI 이미지를 훈련(train), 검증(validation) 및 테스트(test) 세트로 나누는 단계;

4) 3차원 컨볼루션 신경망을 이용하여 상기 뇌 MRI 이미지를 학습하는 단계;

5) 훈련된 분류 모델을 획득하는 단계;

6) 상기 훈련된 분류 모델에서 획득한 확률 점수(probability score)에 따라 알츠하이머병의 진행단계를 분류하는 단계;를 포함하는 것이고,

상기 3차원 컨볼루션 신경망은 스케일링된 감마 탄(scaled gamma tanh, SGT) 활성화 함수를 포함하는 것이며, 상기 스케일링된 감마 탄 활성화 함수는 하기 [수학식 1] 및 [수학식 5]를 사용하는 것인, 알츠하이머병 진행단계 분류를 위한 머신러닝 시스템에 의한 진단정보 제공방법

[수학식 1]

$y = f(x) =$ $x < 0$인 경우 ax^{α}이고, $x > 0$인 경우 bx^{β}

[수학식 5]

$$Z_l = real\begin{bmatrix} \tanh(Y_1^1) & \cdots & \tanh(Y_1^b) \\ \vdots & \ddots & \vdots \\ \tanh(Y_n^1) & \cdots & \tanh(Y_n^b) \end{bmatrix} = \begin{bmatrix} Z_1^1 & \cdots & Z_1^b \\ \vdots & \ddots & \vdots \\ Z_n^1 & \cdots & Z_n^b \end{bmatrix}$$

여기에서, x는 4차원 행렬/텐서에 의해 X_l로 정의되는 입력이고, x는 레이어 'l'의 'b번째' 배치 및 'n번째' 필터에 대한 각 픽셀/피처 값 X_n^b을 갖는 X_l로 4D 행렬/텐서에 의해 정의되는 입력이며, a와 b는 스케일링 상수인자이고, α, β는 학습 가능한 매개 변수임.

청구항 2

제1항에 있어서, 상기 4)는

4-1) 뇌 MRI 이미지 입력 레이어;

4-2) 컨볼루션 레이어, 정규화 레이어, 활성화 레이어 및 풀링 레이어를 순차적으로 반복하는 단계; 및

4-3) 적어도 하나의 완전연결 레이어(Fully connected layer, FCL), 드롭아웃 레이어 및 분류출력 레이어;를 포함하는 것인, 알츠하이머병 진행단계 분류를 위한 머신러닝 시스템에 의한 진단정보 제공방법

청구항 3

제2항에 있어서, 상기 정규화 레이어는 배치 정규화(batch normalization)를 사용하는 것인, 알츠하이머병 진행단계 분류를 위한 머신러닝 시스템에 의한 진단정보 제공방법

청구항 4

제1항에 있어서, 하기 [수학식 11]을 사용하여 분류 손실을 최적화하는 단계를 추가로 포함하는 것인, 알츠하이머병 진행단계 분류를 위한 머신러닝 시스템에 의한 진단정보 제공방법

[수학식 11]

$$loss(E) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{K} t_{ni}\ln\left(y_{ni}\right)$$

여기서 N은 K개의 상호 배타적인 라벨을 가진 훈련 샘플의 총 수이고 t_{ni}는 목표 출력이며 y_{ni}는 i번째 클래스에 속하는 n번째 샘플에 대해 계산된 자연 로그(ln)로 예측된 값임.

발명의 설명

기 술 분 야

[0001] 본 발명은 3차원 컨볼루션 신경망의 스케일링된 감마 탄 활성화 함수를 이용한 알츠하이머병 진행단계 분류를 위한 진단정보 제공방법에 관한 것이다.

배 경 기 술

[0003] 활성화 함수는 DNN에서 두 가지 목적으로 주로 사용되는데, 첫 번째는 복잡한 패턴을 학습하기 위해 전체 시스템에서 비선형성을 추가하는 것이고 두 번째는 계산 부담을 줄이기 위해 각 레이어의 출력을 정규화하거나 임계값을 지정하는 것이다. 여기서 CNN의 경우 선형 활성화 f(x) = wx + b만 사용되는 경우, 컨볼루션 레이어 자체도 선형 연산 레이어라는 것을 주목하여 f(x)의 다중 함수를 적층하면 단일 정도의 출력만 생성된다. 출력 값을 제외하고 최대 또는 최소 수준으로 단조롭게 폭발하여 수렴에 도달하는 훈련에 어려움을 초래할 수 있다. 따라서 학습된 다항식 식은 다중 패턴 특성으로 인해 복잡한 패턴을 학습하려면 1보다 커야 한다. 즉, 결정 경계가 비선형적이어야 한다. 이를 위해서는 훈련 역학과 필요한 작업 성능에 중요한 영향을 미치기 때문에 심층 네트워크에서 활성화 함수를 적절하게 선택해야 한다.

[0004] 전통적으로 MLP(Multilayer Perceptron)를 구현하는 신경망은 뉴런 또는 노드에서 Sigmoid 함수 또는 tanh를 비선형 연산자로 사용했다. 이후 DNN의 복잡성이 증가함에 따라 많은 다른 활성화 함수가 제안되었다. 그러나 대부분은 매우 복잡하고 ImageNet과 같은 자연 이미지 데이터 세트에서 높은 수준의 추상적 표현을 위해 매우 심층적인 네트워크를 위해 설계되었다. 그것은 네트워크의 작동 메커니즘과 특징 추출 과정을 이해하기 위해 네트워크를 더 복잡하게 만들었다. 따라서 ReLU와 그 변형인 Leaky-ReLU와 같은 더 간단한 비선형 정류기는 CNN과 같은 DNN에서 때때로 사용되는 다른 파라메트릭 ReLU(P-ReLU), GELU, ELU, SELU와 함께 가장 일반적인 것이다. CNN 개선을 위한 연구에서 Zhang 등은 컨볼루션 레이어와 FCL에 대한 비선형 변환의 사용을 지원했다. 비대칭 커널 근사의 추가는 분류 작업과 일반화 능력을 향상시켰다. 마찬가지로, Hayou 등은 활성화 함수가 DNN에 미치는 영향을 연구했고 부적절한 선택은 순방향 변환 중 입력의 정보 손실과 역방향 변환 중 그레이디언트의 기하급수적 소멸/감소로 이어질 수 있다고 결론 내렸다. 최근, Dubeey 등은 활성화 함수의 비선형 변환 동작을 이해하기 위해 딥 러닝의 성능 분석에 대한 포괄적인 조사를 수행했다. ReLU는 양의 기울기 흐름에 대한 음의 입력을 완전히 차단하는 반면, 다른 변형은 작은 음의 기울기 손실에 대해 계산된 음의 입력 흐름을 허용한다. ReLU에서 양의 기울기 손실로 해결되었지만, 나중에 이러한 음의 입력을 우세하게 되면 이러한 음의 문제를 계속해서 해결하게 되는 '재귀도'라는 또 다른 문제가 발생하게 된다. 사라지는 그레이디언트 문제는 ReLU에서 양의 그레이디언트 손실로 해결되었지만, 희소성을 희생하여 더 높은 음의 입력이 계속 우세할 경우 발생하는 'dying ReLU'라는 또 다른 유사한 문제를 발생시켰다. 이후 이러한 문제는 f(x) = αx와 같이 음의 입력에 대한 0이 아닌 활성화로 Leaky-ReLU 및 P-ReLU를 사용하여 해결되었으며, 여기서 α는 일정한 스칼라 또는 학습 가능한 매개 변수이다. 그러나 MRI 및 PET(Positron Emission Tomography)와 같은 의료 영상 분류의 경우, ReLU와 Leaky-ReLU는 단순성과 훈련 영상이 회색조 형식이기 때문에 여전히 우세하다. DNN을 사용한 MRI 분류의 최근 연구에는 강력하고 우수한 아키텍처 설계, 임상 특징과 함께 앙상블 모델 및 새로운 학습 및 최적화 알고리듬(algorithm)을 적용하기 위한 실험이 포함된다. 대부분의 연구자들이 기존 활성화 방법을 사용하기 때문에 MRI

에 특화된 새로운 활성화 기능을 설계하는 작업은 거의 이루어지지 않았지만, Hosseini-Asl 등은 Sigmoid 및 ReLU 기능을 사용하여 알츠하이머병(AD) 대 경도 인지 손상(MCI) 대 대조 정상(CN) 작업의 예측을 위해 구조 MRI(sMRI) 이미지에 대해 심층 감독되고 적응 가능한 3D CNN(DSA-3D-CNN)을 설계했다. Payan 등은 MRI 스캔을 분류하기 위해 시그모이드 활성화 함수를 사용하여 희소 자동 인코더(SAE, sparse auto-encoder) 패치 기반 3D CNN을 제안했다. 마찬가지로, Oh 등은 sMCI 대 pMCI 분류에 대한 감독된 전달 학습과 함께 AD 대 NC 분류의 활성화 함수로 ReLU와 컨볼루션 자동 인코더(CAE, convolutional auto-encoder) 기반 체적 CNN을 사용하여 5배 교차 검증(CV)을 수행했다. Gupta 등은 시그모이드 활성화 기능이 있는 CNN을 사용하여 MRI를 자동 인코더를 사용하여 자연 영상에서 학습된 전달 특징을 가진 3개 클래스로 분류했다. E. Goceri는 3D-CNN에 대한 Sobolev 그레이디언트 기반 최적화를 제안했으며, MRI 분류 정확도에 대한 결과는 시그모이드 및 ReLU에 비해 Leaky-ReLU로 더 높게 보고되었다. 최근 Huang 등은 뇌종양 영상 분류를 위한 DNN 모델에 GELU와 ReLU의 조합을 구현했으며 95.49%의 성공률을 달성했다.

[0005] 일반적으로 감마 보정($f(x) = x^{\gamma}$)은 입력 x에 대한 지수 γ에 따라 대조도 향상 및 새 값에 대한 비결정적 강도 매핑에 관한 것이다. 딥러닝 시나리오에서 감마 보정은 훈련 자료를 증가시키기 위한 증강 이미지(γ = 0.5, 1.5, 2 등과 같이 정의된 γ 값 사용)를 생성하는 데 대부분 사용된다. 이 아이디어는 $f(x) = x^{\gamma}$에서 서로 다른 γ 값을 사용하여 여러 버전의 감마 이미지를 생성함으로써 훈련 결과를 높이는 데 도움이 될 것으로 보인다. 그러나 일치하지 않는 감마 버전으로 인해 여러 이미지의 품질이 저하될 수 있다는 점도 유의해야 한다. γ 값이 높으면 이미지를 깨끗하게 할 수 있는 반면 γ 값이 낮으면 중요한 픽셀 정보를 잃을 수 있다. 따라서 'γ'는 '고정' 상수가 아닌 채널에 따라 '만능' 상수 또는 기술적으로 학습 가능한 매개 변수여야 한다.

선행기술문헌

특허문헌

[0007] (특허문헌 0001) 한국등록특허 제2152957호

(특허문헌 0002) 한국등록특허 제2436035호

(특허문헌 0003) 한국공개특허 제2023-0116315호

(특허문헌 0004) 한국공개특허 제2023-0108213호

(특허문헌 0005) 한국공개특허 제2023-0079140호

발명의 내용

해결하려는 과제

[0008] 본 발명의 목적은 3차원 컨볼루션 신경망의 스케일링된 감마 탄 활성화 함수를 이용한 알츠하이머병 진행단계 분류를 위한 진단정보 제공방법을 제공하는 것이다.

과제의 해결 수단

[0010] 본 발명은 1) 정상 대조군, 경도인지장애 및 알츠하이머병 환자의 뇌 MRI 이미지를 획득하는 단계; 2) 상기 뇌 MRI 이미지를 전처리하는 단계; 3) 상기 뇌 MRI 이미지를 훈련(train), 검증(validation) 및 테스트(test) 세트로 나누는 단계; 4) 3차원 컨볼루션 신경망을 이용하여 상기 뇌 MRI 이미지를 학습하는 단계; 5) 훈련된 분류 모델을 획득하는 단계; 6) 상기 훈련된 분류 모델에서 획득한 확률 점수(probability score)에 따라 알츠하이머병의 진행단계를 분류하는 단계;를 포함하는 것이고, 상기 3차원 컨볼루션 신경망은 스케일링된 감마 탄(scaled gamma tanh, SGT) 활성화 함수를 포함하는 것이며, 상기 스케일링된 감마 탄 활성화 함수는 하기 [수학식 1] 및 [수학식 5]를 사용하는 것인, 알츠하이머병 진행단계 분류를 위한 머신러닝 시스템에 의한 진단정보 제공방법을 제공한다.

[0011] [수학식 1]

[0012] $y = f(x) = x < 0$인 경우 ax^{α}이고, $x > 0$인 경우 bx^{β}

[0013] [수학식 5]

[0014] $$Z_l = real\begin{bmatrix} \tanh(Y_1^1) & \cdots & \tanh(Y_1^b) \\ \vdots & \ddots & \vdots \\ \tanh(Y_n^1) & \cdots & \tanh(Y_n^b) \end{bmatrix} = \begin{bmatrix} Z_1^1 & \cdots & Z_1^b \\ \vdots & \ddots & \vdots \\ Z_n^1 & \cdots & Z_n^b \end{bmatrix}$$

[0015] 여기에서, x는 4차원 행렬/텐서에 의해 X_l로 정의되는 입력이고, x는 레이어 'l'의 'b번째' 배치 및 'n번째' 필터에 대한 각 픽셀/피처 값 X_n^b을 갖는 X_l로 4D 행렬/텐서에 의해 정의되는 입력이며, a와 b는 스케일링 상수인자이고, α, β는 학습 가능한 매개 변수임.

[0016] '활성화 함수(activation function)'라는 용어는 여기서 인공 뉴런 네트워크에서 출력 값을 생성하기 위해 선형 또는 비선형 변환을 일부 입력 값에 적용하기 위한 임의의 방법 또는 알고리즘을 나타낸다. 활성화 함수의 예들은 식별(identity), 정류 선형(rectified linear), 누수된(leaky) 정류 선형, 임계값(thresholded) 정류 선형, 파라메트릭(parametric) 정류 선형, 시그모이드(sigmoid), 탄(tanh), 소프트맥스(softmax), 로그(log) 소프트맥스, 맥스 풀(max pool), 다항식(polynomial), 사인(sine), 감마(gamma), 소프트 사인(soft sign), 헤비사이드(heavyside), 스위시(swish), 지수 선형(exponential linear), 스케일된(scaled) 지수 선형, 및 가우스 오류(gaussian error) 선형 함수를 포함한다.

[0017] 본 발명의 '스케일링된 감마 탄(scaled gamma tanh, SGT) 활성화 함수'는 각 이미지에 대해 적절한 감마 값을 선택하거나 배치 정규화(BN) 후 모든 채널에서 출력된 모든 이미지(또는 그 특징)에 대해 보다 구체적으로 선택하는 것이다. 따라서, 본 발명의 방법은 활성화 함수로 작동하는 훈련 샘플의 수를 증가시키지 않고 각 필터 출력에 대해 적절한 감마 값을 찾고 모델에 비선형성을 동시에 가져오는 것이 아니라 증강 이미지의 수를 증가시키는 것이다(도 1).

[0018] 본 발명은 감마 보정 기술과 쌍곡 탄젠트(tanh) 함수의 단계적 조합으로 새로운 활성화 함수를 제안한다. 그러나 Sigmoid와 tanh와 같은 0 중심 대칭 함수는 비뚤어지지 않은 그래디언트에 대한 활성화 함수에 바람직하지만, 이러한 함수는 사라지는 그래디언트 문제로 인해 가치가 없는 것으로 입증되었다. 가장 잘 입증된 최근의 활성화 함수는 대부분 0 주변에서 비대칭이므로, 본 발명에서는 또한 비대칭 함수를 개발하고 있다. 본 발명의 제안된 아이디어를 적용하기 위해, 본 발명에서는 완전연결 레이어(fully connected layer, FCL)의 수가 감소된 이전에 사용된 아키텍처를 사용하여 MRI 분류를 위해 제안된 SGT 활성화 기술을 구현했다.

[0019] 또한, 다양한 실험 분석이 본 발명의 결과를 뒷받침하기 위해 도입된다. 각 활성화 레이어는 BN 레이어에 의해 선행되기 때문에, 원래 단위 분산을 갖는 0에 중심을 두고 있던 입력 데이터의 낮은 강도와 높은 강도에서 포화를 갖는 히스토그램을 분배하는 것이 아이디어이다. 다시 말해, 강도 프로파일은 중심 영역에서 가장자리로 분산된다. 이것은 분류를 지원하기 위해 특징에서 상당한 구별을 갖는 가중치 분포에서 더 높은 분산을 가져온다(도 7).

[0020] 본 발명의 일 구현예에 따르면, 상기 4)는 4-1) 뇌 MRI 이미지 입력 레이어; 4-2) 컨볼루션 레이어, 정규화 레이어, 활성화 레이어 및 풀링 레이어를 순차적으로 반복하는 단계; 및 4-3) 적어도 하나의 완전연결 레이어(Fully connected layer, FCL), 드롭아웃 레이어 및 분류출력 레이어;를 포함하는 것이다.

[0021] 본 발명의 다른 구현예에 따르면, 상기 정규화 레이어는 배치 정규화(batch normalization)를 사용하는 것이다.

[0022] 본 발명의 또 다른 구현예에 따르면, 하기 [수학식 11]을 사용하여 분류 손실을 최적화하는 단계를 추가로 포함하는 것이다.

[0023] [수학식 11]

[0024] $$loss(E) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{K} t_{ni}\ln(y_{ni})$$

[0025] 여기서 N은 K개의 상호 배타적인 라벨을 가진 훈련 샘플의 총 수이고 t_{ni}는 목표 출력이며 y_{ni}는 i번째 클래스에 속하는 n번째 샘플에 대해 계산된 자연 로그(ln)로 예측된 값임.

발명의 효과

[0027] 본 발명의 정상인으로부터 알츠하이머병 또는 경도인지장애 분류를 위한 진단정보 제공방법으로 3차원 컨볼루션 신경망에서 표준 ReLU 및 tanh 함수보다 우수한 성능을 나타내는 스케일링된 감마 탄(scaled gamma tanh, SGT)을 활성화 함수로 사용하여 알츠하이머병 진행 단계를 분류할 수 있는 최적의 진단 수단을 제공하는 효과를 가질 수 있다.

도면의 간단한 설명

[0029] 도 1은 본 발명의 일 실시예에 따른 SGT를 사용한 입력(a), 출력(b), ReLU를 사용한 출력(c) 및 Leaky-ReLU를 사용한 출력(d)으로부터 몽타주로서 63 × 63 × 63 이미지에 해당하는 2차 활성화 레이어(64개 채널 중 22번째)에서 관찰된 샘플 MRI에 대해 다른 기능을 사용한 활성화 비교 결과를 나타낸 것이다. ReLU(c) 및 Leaky ReLU(d)에 비해 SGT로부터의 출력은 처음 세 개 및 마지막 몇 개의 슬라이스에 존재하는 특징 속성을 잘 보존하고 있음을 알 수 있고, 여기서 (a)는 입력 특징 행렬이다.

도 2는 본 발명의 일 실시예에 따른 (a) 입력 x와 f(x)에 대한 활성화 함수 플롯과 x = 0 근처의 다른 일반적인 활성화 함수를 함께 표시; (b) 임계 및 스콰싱 함수의 필요성을 설명하기 위해 감마 보정('전용-감마(only-gamma)')과 쌍곡 탄젠트('tanh')의 조합을 사용하여 두 지수가 모두 1인 활성화(제안-SGT) 및 1차 도함수(d(제안-SGT) 플롯; (c) 레이어 4의 18번째 필터(64개 중)에서 훈련된 네트워크에 대한 실제 활성화 플롯; (d) 레이어 4의 31번째 필터(64개 중)에서 훈련된 네트워크에 대한 실제 활성화 플롯을 나타낸 것이다. 여기서 파란색 곡선은 SGT 활성화 함수를 나타내는 반면 빨간색 곡선은 1차 도함수를 나타낸다.

도 3은 본 발명의 일 실시예에 따른 SGT 활성화를 사용한 MRI 분류를 위하여 적용된 CNN 아키텍처를 나타낸 것이다.

도 4는 본 발명의 일 실시예에 따른 제안된 방법에 대한 블록 다이아그램을 나타낸 것이다.

도 5는 본 발명의 일 실시예에 따른 제안된 SGT 레이어가 점진적으로 Leaky-ReLU 레이어를 대체하여 제어된 실험 결과를 나타낸 것이다. 여기서 감마_X는 SGT 층을 나타내고, 여기서 X는 SGT 층의 수이다. 본 발명은 최종 실험에서 총 4개의 활성화 레이어를 교체했다. 결과는 SGT 레이어의 수가 증가함에 따라 계속해서 개선되고 있다.

도 6은 본 발명의 일 실시예에 따른 (a) 활성화 기능이 다른 기준선 CNN 모델을 사용한 MRI 분류에 대한 훈련 정확도 플롯; (b) 활성화 기능이 다른 기준선 CNN 모델을 사용한 MRI 분류에 대한 검증 정확도 플롯을 나타낸 것이다.

도 7은 본 발명의 일 실시예에 따른 상이한 레이어들, 즉 4, 8, 12, 및 16에 대해 플롯된 단일 MRI 입력에 대한 다양한 활성화 함수들을 사용하는 출력에 대한 입력 특징들의 히스토그램(레이어들에 대해서는 표 2 참조)을 나타낸 것이다. 여기서, 히스토그램들은 64개의 필터들/채널들로부터의 모든 데이터 값들을 사용하여 조합적으로 생성된다. 일반적으로, 모든 채널들의 조합된 히스토그램은 각 채널들의 단일 히스토그램과 유사하다.

도 8은 본 발명의 일 실시예에 따른 활성화 함수와 기존의 활성화 함수의 작동방식을 나타낸 것이다. 기존의 활성화 함수는 모든 입력에 대해 일정한 방식으로 작동하는 반면, 제안된 SGT 함수는 [수학식 2]와 관련하여 레이어 채널 내에서 매개 변수 α_n, β_n의 값을 변경하기 때문에 서로 다른 채널에 대해 다르게 작동한다.

도 9는 본 발명의 일 실시예에 따른 아담 최적화를 사용하여 감마4_아담 네트워크에 대한 상이한 레이어에서 훈련된 모델에 대한 α 및 β 값의 대표적인 그림을 나타낸 것이다. 여기서 α 및 β는 각각 64개 채널에 해당하는 SGT 레이어의 채널별 학습 가능 매개 변수이다.

도 10은 본 발명의 일 실시예에 따른 최종 FCL 1728은 각 클래스 레이블에 해당하는 훈련된 감마4_adam 네트워크의 가중치 플롯을 나타낸다. 여기서 FC_AD_row는 AD 클래스에 속하는 완전 훈련된 감마4_adam 네트워크에서

레이어의 최종 가중치를 나타내며, 유사하게, FC_CN_row와 FC_MCI_row는 각각 CN 및 MCI 범주에 대해 나타낸다. act1_AD의 플롯은 테스트 단계 동안 훈련된 모델을 사용하여 얻은 일반적인 AD 분류 MRI에 대해 계산된 가중치이다. 따라서 CN과 MCI에 대해 각각 act1_CN 및 act_MCI로 계산된 가중치는 테스트 중에 MRI를 분류한 것인가. 이 플롯은 테스트 샘플 (act1_xx)이 부모 클래스 특성 (FC_xx_row)을 얼마나 가깝게 따르는지 보여주기 위한 것이다. 또한 이러한 특성을 평가하기 위해 표 5와 같이 상관 테이블을 계산했는데, 여기서 xx는 샘플과 부모 모두에 대해 동일한 클래스를 나타내는 것이 매우 명확하다. FC_xx_row와 act_xx 사이의 동일한 클래스 높은 상관관계는 네트워크가 클래스별 특성을 정확하게 학습하고 있음을 보여준다.

도 11은 본 발명의 일 실시예에 따른 다른 활성화 함수를 사용하여 기준 CNN 모델에서 최종 FCL 레이어의 편향 값 플롯을 나타낸 것이다.

도 12는 본 발명의 일 실시예에 따른 t-SNE 알고리즘을 사용하여 1728 차원에서 3으로 축소된 테스트 세트 특징에 대해 동일한 각도에서 본 3D 프로젝션을 나타낸 것이다. 여기서 각 색상 점은 MRI 스캔을 나타내므로 296 테스트 MRI에 대해 총 296개의 점이 있다. 비선형 특징 분포는 분류를 위한 복잡한 경계의 요구 사항을 보여준다. 여기서 왼쪽에서 오른쪽으로 방향으로 상기 도는 동일한 기준선 3D CNN 모델에서 별도로 ReLU, Leaky-ReLU 및 SGT 활성화를 사용한 t-SNE 분포의 결과로 얻어진다.

발명을 실시하기 위한 구체적인 내용

[0030] 이하, 실시예를 통하여 본 발명을 더욱 상세히 설명하기로 한다. 이들 실시예는 단지 본 발명을 예시하기 위한 것이므로, 본 발명의 범위가 이들 실시예에 제한되는 것으로 해석되지는 않는다.

[0032] **<실시예 1> Proposed SGT activation and training process**

[0033] 제안된 SGT 활성화는 두 단계로 수행된다:

[0035] [수학식 1]

[0036] 단계 1: $y = f(x) = x < 0$인 경우 ax^{α}이고, $x > 0$인 경우 bx^{β}

[0037] 여기서 첫 번째 단계는 [수학식 1]과 같이 입력 x의 감마 보정 버전을 찾는 것이다. x는 4차원 행렬/텐서에 의해 X_l로 정의되는 입력이다. x는 레이어 'l'의 'b번째' 배치 및 'n번째' 필터에 대한 각 픽셀/피처 값 X^{b}_{n}을 갖는 X_l로 4D 행렬/텐서에 의해 정의되는 입력이다. a와 b는 수동으로 설정된 상수 스케일링 인자이다. n개의 필터의 경우, 학습 가능한 매개 변수 n개(즉, α 또는 β)의 값이 있으며, 이는 동일한 미니 배치에 속하는 모든 다른(또는 동일한) 클래스 이미지에 대해 지수의 값은 동일하게 유지되는 반면, 지수의 값은 다른 채널의 동일한 클래스 이미지에 대해 다르므로 각 채널에서 서로 다르게 활성화된다. [수학식 2]의 행렬 표현에서 볼 수 있듯이, ^는 열에서 열 요소의 현명한 지수 연산에서 수행되는 연산을 의미한다.

[0039] [수학식 2]

$$Y_l = a \cdot \begin{bmatrix} X_1^1 & \cdots & X_1^b \\ \vdots & \ddots & \vdots \\ X_n^1 & \cdots & X_n^b \end{bmatrix} \wedge \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix} = a \cdot \begin{bmatrix} X_1^{1\alpha_1} & \cdots & X_1^{b\alpha_1} \\ \vdots & \ddots & \vdots \\ X_n^{1\alpha_n} & \cdots & X_n^{b\alpha_n} \end{bmatrix} = \begin{bmatrix} Y_1^1 & \cdots & Y_1^b \\ \vdots & \ddots & \vdots \\ Y_n^1 & \cdots & Y_n^b \end{bmatrix} \quad (\text{for } X_n^b < 0)$$

$$= b \cdot \begin{bmatrix} X_1^1 & \cdots & X_1^b \\ \vdots & \ddots & \vdots \\ X_n^1 & \cdots & X_n^b \end{bmatrix} \wedge \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_n \end{bmatrix} = b \cdot \begin{bmatrix} X_1^{1\beta_1} & \cdots & X_1^{b\beta_1} \\ \vdots & \ddots & \vdots \\ X_n^{1\beta_n} & \cdots & X_n^{b\beta_n} \end{bmatrix} = \begin{bmatrix} Y_1^1 & \cdots & Y_1^b \\ \vdots & \ddots & \vdots \\ Y_n^1 & \cdots & Y_n^b \end{bmatrix} \quad (\text{for } X_n^b > 0)$$

[0040]

$$X_l = \begin{bmatrix} X_1^1 & \cdots & X_1^b \\ \vdots & \ddots & \vdots \\ X_n^1 & \cdots & X_n^b \end{bmatrix}$$

[0041] 여기에서 는 레이어 l에 대한 입력이다.

[0042] 여기서 a와 b는 수동으로 선택한 스케일링 상수로, 본 발명의 경우에 대해 각각 0.1과 1.1로 선택했다. 지수가 1과 같고 첫 번째 단계(도 2a)에서 Leaky-ReLU 함수와 유사할 때 단조 함수로 약간 동작한다. 두 번째 단계의 후반에서, 쌍곡 탄젠트(두 지수 모두 1로서) 함수를 통과할 때, 양의 부분에 대한 출력은 탄과 유사할 것이고 음의 부분에 대한 출력은 부분적으로 Leaky-ReLU 함수와 유사할 것이다(도 2b). 그러나 지수 값과 부호를 변경하면 도 2c 및 2d와 같이 다른 활성화 플롯이 생성될 수 있다. 여기서 활성화를 위해 단계 1을 사용하는 것만이 양의 영역에서 활성화된 값을 폭발시킬 수 있고 음의 영역에서 기울기가 사라지는 것(도 2b의 '전용-감마(only-gamma)')으로 이어질 수 있어 훈련 중 수렴에 계산상 어려움을 초래할 수 있다는 점에 유의해야 한다. 따라서 양과 음의 축에 비선형 및 대칭 특성을 갖는 임계 함수가 필요하며, 이에 대해 tanh 함수를 선택했다. 학습 가능한 매개 변수 α 및 β 값은 양의 감마 보정기로 작동하므로 값 α 및 β의 가중치 업데이트는 [수학식 3] 및 [수학식 4]에서와 같이 후방 전파 중 [수학식 1]의 편미분으로부터 계산된다:

[0044] [수학식 3]

$$\frac{\mathrm{d}l}{\mathrm{d}\alpha} = \sum_b \sum_n 0.1 \times real\left(\log_{10} X_b^n\right) \cdot real\left(X_b^{n\alpha}\right) \cdot \frac{\mathrm{d}l}{\mathrm{d}z} for\ X_n^b < 0$$

[0045]

[0047] [수학식 4]

$$\frac{\mathrm{d}l}{\mathrm{d}\beta} = \sum_b \sum_n 1.1 \times real\left(\log_{10} X_b^n\right) \cdot real\left(X_b^{n\beta}\right) \cdot \frac{\mathrm{d}l}{\mathrm{d}z} for\ X_n^b > 0$$

[0048]

[0049] $X_b^n = X$가 음수이고 α가 유리 십진수(rational decimal number)일 때 결과 X^{α}는 복소수가 되는데, 이 경우 복소수의 실수 부분만 사용하게 된다. $\log_{10}X$와 X^{β}의 경우도 마찬가지이다. 또한, α 또는 β의 절대값은 양의 지수를 얻기 위해 [수학식 2], [수학식 3] 및 [수학식 4]에서 사용된다.

[0051] 단계 2: z = tanh(y) 또는 행렬 형태는 다음과 같다:

[0053] [수학식 5]

$$Z_l = real\begin{bmatrix} \tanh\left(Y_1^1\right) & \cdots & \tanh\left(Y_1^b\right) \\ \vdots & \ddots & \vdots \\ \tanh\left(Y_n^1\right) & \cdots & \tanh\left(Y_n^b\right) \end{bmatrix} = \begin{bmatrix} Z_1^1 & \cdots & Z_1^b \\ \vdots & \ddots & \vdots \\ Z_n^1 & \cdots & Z_n^b \end{bmatrix}$$

[0054]

[0055] 여기서 모든 연산은 요소 단위 행렬 연산이므로 [수학식 5]와 같이 [수학식 2]를 이용하여 계산된 행렬을 행렬 계산에 전달한 다음 레이어 l의 출력 행렬 Z_l을 풀링 레이어에 전달한다. 레이어 손실 $\frac{dl}{dX}$에 대해서는 먼저 (w.r.t) X_l에 대한 Y_l의 도함수를 [수학식 6]을 이용하여 계산하여 출력 Y' 차원이 레이어 입력, 즉 X_l의 차원과 정확히 일치하도록 한다.

[0057] [수학식 6]

$$Y' = \frac{dY_l}{dX_l} = 0.1 \times \alpha \cdot real\left(X^{\alpha-1}\right) for\ X < 0$$

[0058] $$= 1.1 \times \beta \cdot real\left(X^{\beta-1}\right) for\ X \geq 0$$

[0059] 그 다음, 전체 구배 손실 $\frac{dl}{dX}$ 은 [수학식 7]을 사용하여 이전 레이어들로 역방향 전파되는 Z_l w.r.t Y'의 도함수로서 이 레이어의 출력을 통해 계산된다.

[0061] [수학식 7]

[0062] $$\frac{dl}{dX} = \frac{dZ_l}{dY'} \cdot \frac{dl}{dZ} = \frac{d\tanh\left(Y'\right)}{dY'} \cdot \frac{dl}{dZ} = \operatorname{sech}^2\left(Y'\right) \cdot \frac{dl}{dZ}$$

[0063] 여기서, $\frac{dl}{dZ}$ 는 심층 레이어에서 역산된 손실이다. z = tanh(y)가 스콰싱 함수로 사용되기 때문에, 레이어의 최종 출력 값은 다음 레이어로 전달되기 전에 불균일하게 스케일링되어 z가 0을 중심으로 하는 비대칭 함수가 된다. 이는 도 2c와 2d에 나타나 있는데, 여기서 d(proposed-SGT)는 제안된 함수의 1차 도함수의 최종 출력에 대한 플롯을 보여준다. 지수 α와 β가 모두 1인 조건의 경우, 활성화 층은 양의 부분에서는 tanh처럼 행동하고 음의 부분에서는 leaky ReLU처럼 행동하는 반면, 도함수의 경우 1차 도함수는 상수이므로 양과 음의 부분에서 각각 출력 상수 0.3592와 0.99006으로 정확히 leaky ReLU처럼 행동한다. 이러한 행동은 18번째 필터에서와 같이 β(양) > α(음)를 갖는 몇개의 필터에서 관찰되었는데, 이는 0으로 비선형적으로 출력되는 두 개의 다른 필터에서와 같이 일정한 출력으로 보인다. 그러나 α와 β 모두 채널별로 학습 가능한 매개 변수이기 때문에 모든 채널에서 값이 동일하지 않다(도 8). α와 β의 최종 값은 -0.2와 1.3 사이로 조사되었으며 거의 동일한 값이 아닌 것으로 나타났다. 실험과 관련하여, 대부분의 필터에서 α와 β의 값은 소수점이 있는 양의 유리수였으며, 대다수의 경우 β가 α보다 크다. β(양) > α(양) 모두의 경우, 31번째 필터에서와 같은 그래프를 따르며 양의 x에 대한 그래디언트 값은 x의 값으로 점진적으로 감소하지만, 감소 속도는 tanh 도함수보다 낮다. 이는 그래디언트 값이 무한히 작아지는 것을 방지하는 데 도움이 되는 반면 음의 도함수 부분에서는 값이 거의 일정하고 상당히 동일하여 모든 경우에 대해 1이 된다. 따라서 네트워크는 사라지는 그래디언트 또는 폭발하는 그래디언트가 덜 쉬워진다. 입력 X, α, β가 0이 될 때 이 경우에는 Sech (0) = ∞ 또한 log (0) = ∞와 같이 불확정적인 형태를 일으켜 훈련을 계속한다. 수렴 후에도 여전히 정의되지 않은 값이 기록되어 있지만(도 9), 무시할 수 있다.

[0064] 네트워크 훈련 및 파라미터 최적화를 위해 Adam 최적화 기법을 사용했다. 수렴에 도달할 때까지 파라미터를 업데이트하는 1차 기울기 기반 최적화 알고리즘이다. t번째 반복 동안 학습 가능한 파라미터(wt)(가중치/편견/α 및 β와 같은 정의된 항)는 다음과 같이 Adam 최적화를 사용하여 업데이트된다:

[0066] [수학식 8]

[0067] $$w_{t+1} = w_t - \frac{am_t}{\sqrt{v_t} + \varepsilon}$$

[0068] 여기서 α는 본 발명의 경우에 0.001로 유지되는 학습률 상수 값이고, ε는 0이 아닌 분모를 유지하기 위한 오프셋으로 사용되는 매우 작은 정규화 상수 값 (10^{-8})이다. 파라미터 그래디언트 (mt)와 그 제곱 값 (vt)의 요소별 이동 평균은 [수학식 9]와 [수학식 10]에서와 같이 계속 업데이트된다. 여기서 b1과 b2는 각각 0.9와 0.990으로 유지되는 mt와 vt의 붕괴율이다.

[0070] [수학식 9]

[0071] $$m_t = b_1 m_{t-1} + (1 - b_1)\nabla E(w_t)$$

[0073] [수학식 10]

[0074] $$v_t = b_2 v_{t-1} + (1 - b_2)[\nabla E(w_t)]^2$$

[0075] 여기서 $\nabla E(w_t)$는 매개변수 w_t에 대한 손실(E)의 1차 도함수, 즉 교차 엔트로피 손실을 나타낸다.

[0077] [수학식 11]

[0078] $$loss(E) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{K} t_{ni} \ln (y_{ni})$$

[0079] 여기서 N은 K개의 상호 배타적인 라벨을 가진 훈련 샘플의 총 수이고 t_{ni}는 목표 출력이며 y_{ni}는 i번째 클래스에 속하는 n번째 샘플에 대해 계산된 자연 로그(ln)로 예측된 값이다.

[0081] **<실시예 2> CNN model and methodology**

[0082] 제안된 함수의 성능 평가는 ADNI 웹사이트 40에서 얻은 AD, CN, MCI로 임상적으로 분류된 3개의 MRI 코호트를 분류하여 수행되었다. 사용된 MRI의 인구 통계학적 세부 사항은 표 1에 나와 있다. 이질성을 추가하고 실험 샘플 수를 증가시키기 위해 상이한 기울기 랩핑 및 스케일 보정 기술을 가진 동일한 환자의 다중 스캔이 사용되었다. 분석에 사용된 세부 아키텍처는 표 2에 나와 있다. 전체 데이터 세트를 5:2:3의 비율로 세 부분 즉, 훈련(train), 검증(validation) 및 테스트(test) 세트로 나누어 495개의 MRI가 훈련에 사용되었고, 검증을 위해 197개의 MRI가 사용되었으며, 훈련된 모델을 테스트하기 위해 296개의 MRI가 분리되었다.

표 1

[0084]

Dataset properties	AD participants	CN participants	MCI participants
Male/female	29/36	22/38	54/33
Mean age	73.55/75.43	75.57/74.43	77.06/72.41
Total number of Participant	65	60	87
Number of MRI scans	209	305	474

표 2

Layer number	Layer name	Layer description	Output size	Number of learnable parameters
1	Image input	64×64×64×1 images with 'zero-center' normalization	64×64×64×1	0
2	Convolution	64 3×3×3×1 convolutions with stride [1 1 1] and padding 'same'	64×64×64×64	Weights = 1728 Bias = 64
3	Batch Normalization	Batch normalization with 64 channels	64×64×64×64	Offset = 64, Scale = 64
4	layer_gamma3d or ReLU/Leaky-ReLU/tanh	Proposed SGT function with 2 learnable parameters for 64 channels	64×64×64×64	α = 64, β = 64 or 0
5	3-D Max Pooling	2×2×2 max pooling with stride [1 1 1] and padding [0 0 0; 0 0 0]	63×63×63×64	0
6	Convolution	64 5×5×5×64 convolutions with stride [1 1 1] and padding 'same'	63×63×63×64	Weights = 512 K Bias = 64
7	Batch Normalization	Batch normalization with 64 channels	63×63×63×64	Offset = 64, Scale = 64
8	layer_gamma3d or ReLU/Leaky-ReLU/tanh	Proposed SGT function with 2 learnable parameters for 64 channels	63×63×63×64	α = 64, β = 64 or 0
9	3-D Max Pooling	2×2×2 max pooling with stride [2 2 2] and padding [0 0 0; 0 0 0]	31×31×31×64	0
10	Convolution	64 7×7×7×64 convolutions with stride [1 1 1] and padding 'same'	31×31×31×64	Weights = 1.404 M Bias = 64
11	Batch Normalization	Batch normalization with 64 channels	31×31×31×64	Offset = 64, Scale = 64
12	layer_gamma3d or ReLU/Leaky-ReLU/tanh	Proposed SGT function with 2 learnable parameters for 64 channels	31×31×31×64	α = 64, β = 64 or 0
13	3-D Max Pooling	2×2×2 max pooling with stride [3 3 3] and padding [0 0 0; 0 0 0]	10×10×10×64	0
14	Convolution	64 9×9×9×64 convolutions with stride [1 1 1] and padding 'same'	10×10×10×64	Weights = 2.985 M Bias = 64
15	Batch Normalization	Batch normalization with 64 channels	10×10×10×64	Offset = 64, Scale = 64
16	layer_gamma3d or ReLU/Leaky-ReLU/tanh	Proposed SGT function with 2 learnable parameters for 64 channels	10×10×10×64	α = 64, β = 64 or 0
17	3-D Max Pooling	2×2×2 max pooling with stride [4 4 4] and padding [0 0 0; 0 0 0]	3×3×3×64	0
18	Fully Connected	1728 fully connected layer	1×1×1×1728	Weights = 2.98 M Bias = 1728
19	Dropout	50% dropout	1×1×1×1728	0
20	Fully Connected	3 fully connected layer	1×1×1×3	Weights = 5.18 K Bias = 3
21	SoftMax	SoftMax function	1×1×1×3	0
22	Classification Output	Cross-entropy with 'AD', 'CN' and 'MCI' labels	1×1×1×3	0

[0086]

[0087] [표 2]에서 layer_gamma3d 레이어에 대한 Matlab 코드 구현

```
classdef layer_gamma3d < nnet.layer.Layer
    properties(Learnable)
        Alpha
        Beta
    end

    methods
        function layer = layer_gamma3d(numchannel,name)
            layer.Name = name;
            layer.Description = " Proposed SGT layer with" +numchannel + "
channels";
            layer.Alpha = rand([1 1 1 numchannel]);
            layer.Beta = rand([1 1 1 numchannel]);
        end

        function Z = predict(layer, X)
              X(isnan(X)) = 0.1;          %for NaN case
              layer.Alpha(isnan(layer.Alpha))= 0.1;
              layer.Beta(isnan(layer.Beta))= 0.1;
              layer.Beta=abs(layer.Beta);
              layer.Alpha=abs(layer.Alpha);
              X_F= 0.1.*(power(complex(X),complex(layer.Alpha,0)));
              check_X = 1.1*(power(complex(X),complex(layer.Beta,0)));
              X_F(X>0) = check_X(X>0);
              Z = tanh(real(X_F));
        end
```

[0088]

```
        function [dLdX,dLdAlpha,dLdBeta] = backward(layer, X, ~, dLdZ, ~)
```

```
        X(isnan(X))= 0.001;                    %for NaN case
        dLdZ(isnan(dLdZ))= 0.001;
        layer.Alpha(isnan(layer.Alpha))= 0.001;
        layer.Beta(isnan(layer.Beta))= 0.001;
        layer.Beta=abs(layer.Beta);
        layer.Alpha=abs(layer.Alpha);
        X_loss = 0.1.*layer.Alpha.*real(power(complex(X),(layer.Alpha-1)));
        dLdX = power(sech(X_loss),2).*dLdZ;
        X_loss2 = 1.1*layer.Beta.*real(power(complex(X),(layer.Beta-1)));
        check = power(sech(X_loss2),2).*dLdZ;
        dLdX(X>0) = check(X>0);
        dLdAlpha =
0.1.*real((log10(complex(X))).*(real(power(complex(X),complex(layer.Alpha,0))
).*(X<0))).*dLdZ;
        dLdAlpha = sum(dLdAlpha,[1 2 3]);
        dLdAlpha = sum(dLdAlpha,5);
        dLdBeta =
1.1*real((log10(complex(X))).*(real(power(complex(X),complex(layer.Beta,0))).
*(X>0))).*dLdZ;
        dLdBeta = sum(dLdBeta,[1 2 3]);
        dLdBeta = sum(dLdBeta,5);
        end
    end
end
```

[0089]

[0090] 사용된 CNN 아키텍처는 도 3과 같이 대표 그림으로 표시된 반면 모든 레이어와 매개 변수 수에 대한 자세한 내용은 표 2에 나와 있다. 사용된 CNN 모델은 과적합을 극복하기 위해 완전연결 레이어가 감소한 이전 연구를 기반으로 한다. 도 3과 같이 컨볼루션 커널의 크기가 3 × 3 × 3에서 최종 활성화 크기인 3 × 3 × 3에 도달할 때까지 9 × 9 × 9 × 9로 계속 증가하는 것을 볼 수 있다. 따라서 이를 발산 네트워크 'divNet'이라고 한다. 제안된 SGT 레이어(녹색 큐브)는 CNN 모델에 원래 사용된 ReLU 활성화를 대체한다. 유사하게 도 4는 전체 분류 프로세스를 나타낸다. NIFTI(Neuroimaging Informationatics Technology Initiative) 형식의 MRI 스캔은 최소 이미지 전처리 단계를 거쳐 64 × 64 × 64로 크기를 조정한 후 CNN에 입력된다. 그런 다음 컨볼루션, 정규화, 활성화 및 풀링 프로세스를 순차적으로 따라 다운샘플링된 특징 벡터를 얻는다. 이 인코딩 프로세스는 FCL까지 반복되고 드롭아웃 레이어가 뒤따른다. 마지막으로, 손실 계산에 사용되는 확률 점수를 출력하는 SoftMax 레이어이다. 사용된 손실 함수는 다중 클래스 분류를 위해 교차 엔트로피이다. 여기서 활성화 레이어에서 수행되는 주요 초점, 입력 및 출력 분석(히스토그램을 통한) 및 학습 가능한 매개 변수의 다른 학습 값을 가진 활성화 함수의 거동이다.

[0091] 모든 실험은 윈도우 10 OS에서 MATLAB R2022a 학술 소프트웨어를 사용하여 수행되었다. 네트워크 모델은 24GB 전용 메모리를 갖춘 NVIDIA GeForce RTX 3090 GPU에서 훈련되었고 32GB 메모리를 가진 인텔®Core™i9-10900 K CPU @ 3.70GHz에서 테스트되었다.

[0093] **<실시예 3> Classification performance**

[0094] 표 3과 도 5에서 SGT 활성화를 사용하는 네트워크의 모든 버전(즉, gamma2, gamma2_alt, gamma4_adam, gamma4_sgdm)이 다른 활성화 체계보다 높은 유효성 정확도를 갖는 것으로 관찰된다. 이러한 분류 성능 매개 변수는 작업의 신뢰성과 정확성을 측정한다. 예를 들어, 정확도는 실제 예측에 대한 정확도에 대하여 정확한 예측의 수를 측정하는 반면, 정밀도는 측정된 값이 실제 값에 얼마나 가까운지를 측정한다. 마찬가지로, 코헨의 카파 점수는 더 견고하고 모델이 다중 클래스 불균형 데이터 세트에 가장 적합한 모델의 성능에 비해 얼마나 더 나은지를 측정한다는 점을 제외하고는 정확도와 같다. 또한 SGT 함수의 효과를 정확하게 조사하기 위해 표준 활성화 레이어를 제안된 활성화 레이어로 점진적으로 대체하여 거의 통제된 실험을 수행하지 않았다. 이 조절된 실험의 결과는 도 5에 나와 있다. 여기서 gamma2는 처음 두 활성화가 SGT와 다른 ReLU임을 의미하고, gamma2_alt는 첫 번째 및 세번째가 SGT와 다른 ReLU를 의미하며, gamma4_adam은 아담 최적화인자가 있는 SGT로 4개의 활성화 레이어를 모두 사용하는 반면 gamma4_sgdm도 4개의 SGT 활성화 레이어를 사용하지만 최적화인자는 SGDM(Stocastic Gradient Descent with Momentum)이다. 검증 세트는 서로 다른 에포크(epoch)에서 예측 정확도를 계산하기 위해 훈련 중에 사용되는 세트이므로 네트워크가 얼마나 잘 학습하고 있는지 아는 데 도움이 된다. 도 6b는 도 6a의 훈련 정확도와 함께 서로 다른 에포크에서 계산된 유효성 검사 정확도를 보여준다. SGT 활성화된 네트워크 (gamma4_sgdm, gamma4_adam)가 훈련의 마지막 단계에서 다른 활성화 체계보다 높은 검증 정확도에 도달한다는 것을 분명히 알 수 있다. 표 3에 보고된 최종 검증 정확도는 80번째 에포크 또는 최종 에포크에 설정된 검증 세트에 대한 정확도이다. 유사하게, 테스트 세트는 훈련된 모델에 대해 완전히 보이지 않는 세트이며

테스트 세트에서 더 높은 성능은 네트워크가 잘 일반화되고 보이지 않는 데이터에 대해 좋은 성능을 가지고 있다는 것을 의미한다. 편향되지 않은 결과를 얻기 위해, 실험 환경과 모든 하이퍼 파라미터 및 참여 MRI는 활성화 함수의 선택에 관계없이 항상 모든 네트워크에 대해 동일하게 유지되었다. 테스트 세트 분류 동안, Leaky-ReLU는 gamm4_sgdm보다 약 0.5% 높은 테스트 정확도로 가장 우수한 성능을 보였다. 여전히 모든 SGT 활성화된 네트워크의 테스트 정확도는 ReLU와 tanh보다 각각 2%와 1% 높았는데, 이는 제안된 SGT 활성화 계획이 전통적인 ReLU 활성화를 명확한 차이로 능가한다는 것을 나타낸다. 표 4는 본 발명의 결과(proposed method)를 일부 최근 연구와 비교한 것을 보여준다.

표 3

Type	Name	Final validation accuracy (%)	Test accuracy (%)	Final validation Loss	Cohen's kappa	Precision (class-wise [AD CN MCI])	Predicted confusion matrix	True confusion matrix
Standard Activation Functions	tanh	90.86	92.57	0.5338	0.897	[0.8889 0.9120 0.9507]	56 1 6 0 83 8 1 6 135	63 0 0 0 91 0 0 0 142
	ReLU	87.81	91.22	1.0425	0.860	[0.9048 0.9011 0.9225]	57 3 3 1 82 8 1 10 131	
	Leaky-ReLU	90.35	**93.92**	0.8201	**0.902**	[0.9206 0.9121 0.9648]	58 2 3 1 83 7 1 4 137	
	Swish	90.35	92.57	0.6022	0.881	[0.9048 0.8901 0.9577]	57 2 4 2 81 8 0 6 136	
SGT Function	gamma4_adam	92.89	92.57	0.5638	0.881	[0.8730 0.9451 0.9366]	55 6 2 0 86 5 1 8 133	
	gamma4_sgdm	**92.89**	93.24	**0.3086**	0.892	[0.9048 0.9121 0.9577]	57 2 4 1 83 7 1 5 136	

[0095]

표 4

[0096]

Authors	Methods	Number of MRI scans	Activation function	Testing accuracy (%)
Gupta et al	Sparse Auto encoder (SAE) based CNN	CN:1278 AD:755 MCI:2282	Sigmoid	94.74
Payan et al	SAE patch-based 3D CNN	CN:755 AD:755 MCI:755	Sigmoid	89.47
Hosseini-Asl et al	DSA-3D-CNN	CN:70 AD:70	Sigmoid and ReLU	97.60
Oh et al	Inception auto encoder-based 3D CNN	NC:230 AD:198 sMCI:101	ReLU	84.50
Proposed method	Diverging CNN	CN:305 AD:209 MCI:474	SGT	93.24

[0097] Gupta, A., Ayhan, M. & Maida, A. Natural image bases to represent neuroimaging data. in International Conference on Machine Learning, pp. 987?994 (2013).

[0098] Payan, A. & Montana, G. Predicting Alzheimer's disease: A neuroimaging study with 3D convolutional neural networks. arXiv Prepr. arXiv:1502.02506 (2015).

[0099] Hosseini-Asl, E., Gimel'farb, G. & El-Baz, A. Alzheimer's disease diagnostics by a deeply supervised adaptable 3D convolutional network. arXiv Prepr. arXiv:1607.00556 (2016).

[0100] Oh, K., Chung, Y.-C., Kim, K. W., Kim, W.-S. & Oh, I.-S. Classification and visualization of Alzheimer's disease using volumetric convolutional neural network and transfer learning. Sci. Rep. 9(1), 1?16 (2019).

[0103] **<실험예 1> Histogram analysis and asymmetric distribution**

[0104] [수학식 2] 또는 [수학식 5]에서와 같이 각 층의 입력(X_l) 또는 출력(Z_l)의 가중치는 히스토그램에서 빈도에 대해 표시된다. BN의 정규화된 출력 값은 거의 정규 분포를 가진 0 평균이므로 ReLU 또는 시그모이드와 같은 활성화 함수를 사용하여 모든 음의 값을 가진 매개 변수/가중치를 버리는 것은 좋은 생각이 아니다. ReLU에서 기울기의 흐름이 양수이지만 가중치의 무더기가 음수이면 음의 가중치에 대해 '0' 도함수를 가진 죽은 ReLU를 초래하므로 매번 ReLU가 현명한 선택은 아니다. MRI와 같은 경우에는 대부분 검은 배경(낮은 픽셀 값)으로 기울기 손실의 흐름을 보장하는 음의 가중치에 대해 0이 아닌 그래디언트를 제공하는 Leaky-ReLU, GELU, SELU와 같은 대체 활성화 기능을 사용하는 것이 좋다.

[0105] 도 7은 ReLU 버전과 비교하여 SGT 레이어를 통한 입력 및 출력 히스토그램 플롯을 보여준다. 여기서 활성화 레이어에 대한 입력은 배치 정규화로부터의 출력이고 활성화 레이어의 출력은 풀링 레이어에 대한 입력이라는 점에 유의해야 한다. 도 7a에서 모든 활성화 레이어의 입력 히스토그램은 거의 대칭적인 분포를 가지고 있는데, 이는 이미지 픽셀의 대부분이 BN 다음의 회색 영역에 있다는 것을 의미한다. 감마 보정의 목표는 이 회색 영역을 줄이고 흰색(밝은)과 검은색(어두운) 영역을 구분하는 것이다. 도 7b를 보면 제안된 SGT 레이어의 출력의 경우 중간 회색 영역이 매우 적은 반면 ReLU 도 7c의 출력은 여전히 매우 높은 0과 leaky-ReLU 도 7d 출력이 여전히 0에 집중되어 있으므로 명확한 왜도가 긍정적인 부분으로 보인다. SGT 레이어의 출력 데이터는 BN과 달리 반대 에지 영역에서 분산되어 있으며 tanh과 leaky-ReLU 히스토그램의 출력의 조합처럼 보인다. 또한 SGT 레이어의 이러한 비대칭 특성 분포는 에지 영역 간의 더 높은 분산으로 인해 분류 작업을 지지한다(도 8).

[0107] **<실험예 2> Channel wise activation**

[0108] 본 발명은 특히 SGT (표 2의 layer_gamma3d) 매개 변수가 서로 다른 활성화 레이어에서 어떤 값을 학습할지에

대해 관심이 있었다. 도 9와 같은 모든 활성화 레이어의 α 및 β 값에 대한 줄기 플롯은 첫 번째 SGT 활성화 레이어(즉, 레이어 4)의 경우 α 및 β 값이 대부분 양수이고 소수만 음수로 유지되었으며, 또한 α보다 1 이상의 값을 가진 β가 더 많다는 것을 보여준다. α 및 β 값의 범위는 -0.4와 1.4 사이에 있었다. 흥미롭게도 중간 활성화 레이어(즉, 레이어 8, 레이어 12)와 최종 활성화 레이어(즉, 레이어 16)에서는 β에 대한 값 중 어느 것도 음수로 유지되지 않았고 대부분의 채널에서 α 값은 음수로 유지되었다. 이는 특징 값 x > 0의 경우 양의 감마 보정이 필요했고, 음의 특징 값 x < 0의 경우 중간 레이어에서 음의 감마 보정이 필요했다는 것을 의미할 수 있다. 보다 일반적인 진술에서, 감마 활성화는 더 밝은 픽셀을 더 밝게, 더 어두운 픽셀을 더 어둡게 보이게 했고, 이는 더 뚜렷한 강도 프로파일을 초래했다.

[0110] **<실험예 3> Analyzing weights and bias in the final fully connected layer**

[0111] FCL은 학습 가능한 가중치와 편향을 가지고 있지만 CNN에서 사용될 때 대부분 활성화 기능이 없는 MLP Feedforward 네트워크를 나타낸다. FCL에서는 패치 기반 특징 추출기로 사용되는 컨볼루션 레이어와 달리 모든 입력이 출력에 매핑되므로 FCL의 가중치와 편향은 결과를 예측하는 데 크게 책임이 있으며 가중치 자체는 어떤 입력이 출력에 더 많은 영향(또는 이득)을 미치는지를 시사한다. 따라서 FCL의 가중치 분포 패턴은 테스트 단계 동안 네트워크가 어떻게 동작하는지를 나타낼 수 있다. 이를 해석하기 위해 도 10과 같이 3개의 모든 클래스에 대해 최종 FCL의 모든 훈련된 가중치(입력 노드 = 1728, 출력 노드 = 3, 연결 = 5184)를 표시했다. 나중에 상관 행렬은 표 5와 같이 계산되며, 이는 FC 레이어에서 계산된 샘플 MRI의 특징(또는 가중치)이 부모 클래스와 밀접한 상관 관계가 있음을 보여준다. 예를 들어, 테스트 샘플 CN MRI의 FC 가중치는 훈련된 네트워크 해당 레이어 가중치 [FC_AD_row FC_CN_row FC_MCI_row]와 상관 관계 값이 있다. 따라서 가장 높은 상관 값은 FC_CN_row에 대해 0.24265146이며, MRI 테스트 샘플은 분류 중 'CN' 클래스 가중치에 대해 더 높은 친화도를 가지고 있으며, 네트워크가 테스트 샘플 레이블을 'CN'으로 예측하는 논리를 뒷받침한다.

표 5

[0113]

	FC_AD_row	act1_AD	FC_CN_row	act1_CN	FC_MCI_row	act1_MCI
FC_AD_row	1	0.215182	0.39671942	−0.14342	0.341976354	−0.19525
act1_AD	0.215182223	1	0.04948163	0.786964	−0.187160957	0.630355
FC_CN_row	0.396719425	0.049482	1	0.242651	0.309621325	−0.00694
act1_CN	−0.143417277	0.786964	0.24265146	1	−0.009627914	0.85748
FC_MCI_row	0.341976354	−0.18716	0.30962132	−0.00963	1	0.275327
act1_MCI	−0.195245895	0.630355	−0.00693744	0.85748	0.275326654	1

[0115] 가중치 분석 후, 본 발명에서는 또한 편향(bias) 값을 분석하는 데 관심이 있었다. 따라서, 이 아이디어는 최종 FCL 레이어에서 계산된 편향을 통해 각 클래스에 얼마나 많은 네트워크가 편향되어 있는지 확인하는 것이다. 획득된 편향 값은 확률 계산을 위해 SoftMax에 들어가는 마지막 FCL에서 나온 것이다. 본 발명에서는 네트워크의 가중치가 입력을 위한 출력 값에 직접적으로 영향을 미치는 반면, 편향은 입력/가중치가 0이고 출력에 연속적인 레이어별 영향을 미치지 않을 때 0이 아닌 출력을 만들기 위해 정규화 상수로 작용한다는 것을 알고 있다. 편향 값을 이론적으로 정확하게 해석하기는 어렵지만, 서로 가까운 편향 값이 서로 수치적으로 어떻게 관련되어 있는지를 상관시킬 수 있다고 가정한다. 예를 들어, tanh 훈련된 CNN의 경우 획득된 편향 값은 [AD CN MCI] = [-0.006021075 0.0003164 0.004943716]이며, 이는 AD (음의 값을 가진)가 CN (작은 양)과 밀접하게 관련되어 있으며, AD와 CN 사이의 값 차이가 AD와 MCI보다 크며, 이는 AD가 모두 치매 질환이라는 일반적인 가정에 반대되는 것이다. 이것은 또한 tanh 네트워크가 AD와 CN보다 쉽게 구별할 수 있음을 나타낼 수 있는데, 이는 원래 되어야 할 것이 아닌 AD와 CN 사이를 구별할 수 있으며, Leaky ReLU의 경우에도 마찬가지이다. 놀랍게도, 이것은 편향의 차이가 더 높으면 네트워크가 클래스별 확률 점수를 계산하기가 더 쉬워지기 때문에 분류 작업에 도움이 될 수 있다. 반대로, 제안된 SGT 네트워크(gamma4_adam 및 gamma4_sgdm)는 AD와 CN 편향 값 사이의 차이가 더

크며, 하나는 양수이고 다른 하나는 음수이다. MCI는 거의 0에 가까운 반면 AD와 CN 사이의 중간 상태를 나타낸다. gamma4_adam 네트워크에서 MCI와 CN 편향 값의 차이가 더 낮다는 것은 실제 시나리오를 지원하는 CN과 MCI 사이의 분류 및 일반화에 더 높은 어려움을 시사할 수 있다.

[0116] 또한 모델의 성능을 분석하기 위해 t-SNE 3D 프로젝션을 사용하여 모델 분류 성능을 평가했다. 도 12는 최종 FCL에서 축소된 특징의 시각화를 위한 3D t-SNE 프로젝션을 나타낸다. FCL로 들어가는 특징은 원래 여러 채널에서 나중에 단일 채널로 축소되었기 때문에 평평한 특징으로 간주된다. 그러나 적절한 시각화를 위해 각 MRI의 평평한 특징을 2D 또는 3D 차원으로 축소해야 한다. 프로젝션에서 독특한 군집 분포는 네트워크가 적합도가 좋은 클래스 판별 특성을 학습하고 있다는 것을 의미한다. 이것은 도 12c에서 볼 수 있는데, SGT는 다른 ReLU 및 Leaky-ReLU보다 더 나은 밀도 높은 클래스 클러스터링(컬러를 통해 볼 수 있음)을 생성한다.

[0117] DNN 설계 및 하이퍼 파라미터 선택은 모두에게 보편적으로 작동할 수 있는 단일 모델이나 기능이 없는 작업별이지만 모든 실험과 분석 후 다음과 같은 결론을 내릴 수 있다:

[0118] - MRI 분류를 위해 3D CNN에서 표준 ReLU 및 tanh 함수보다 우수한 성능으로 새로운 채널별 동적 활성화 함수가 도입되었다.

[0119] - 본 발명에서는 제안된 활성화 함수가 기울기 문제가 사라지거나 폭발할 가능성이 적고 기울기 문제가 없는 음의 가중치로 발생하는 음의 기울기 손실을 줄일 수 있음을 보여주었다 (도 2c 및 2d).

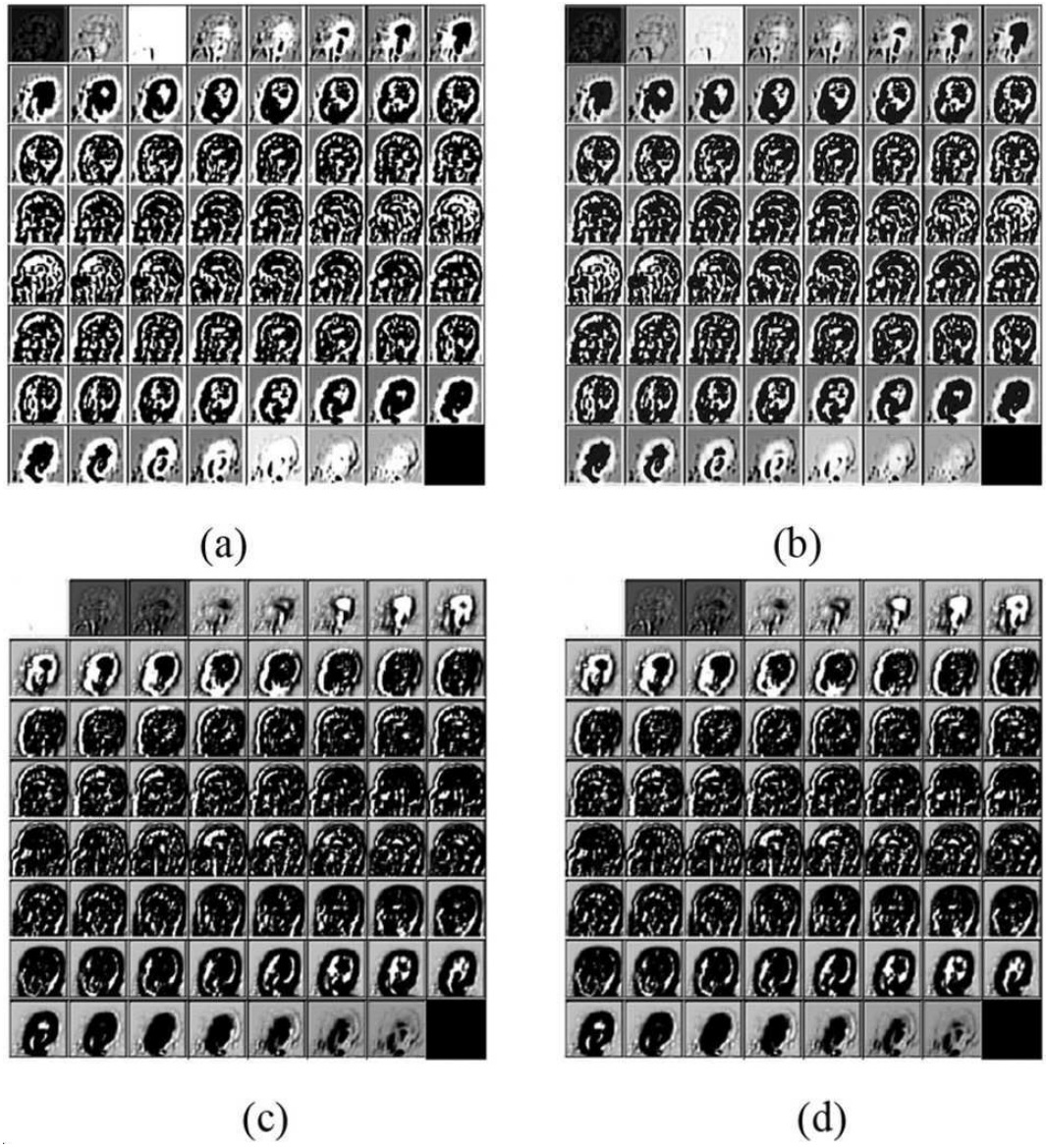
[0120] - 히스토그램에서 수행된 분석(도 7)은 컨볼루션 및 배치 정규화 작업 동안 음의 가중치가 상당히 큰 척도로 생성되므로, 음의 가중치를 활용하여 기울기 손실에 상대적으로 기여하는 아이디어는 제안된 활성화 함수로 의미가 있음이 입증되었다.

[0121] - 본 발명에서는 최종 FCL에서 가중치와 편향의 패턴과 그들이 분류 작업과 관련하여 수치적으로 얼마나 관련되어 있는지(도 11, 12, 표 5)를 탐구하려고 노력했다. 가중치는 다양한 접근법에서 최적화될 수 있지만 분석하기가 어렵기 때문에 이 분야에서 몇 안 되는 시도 중 하나일 수 있다.

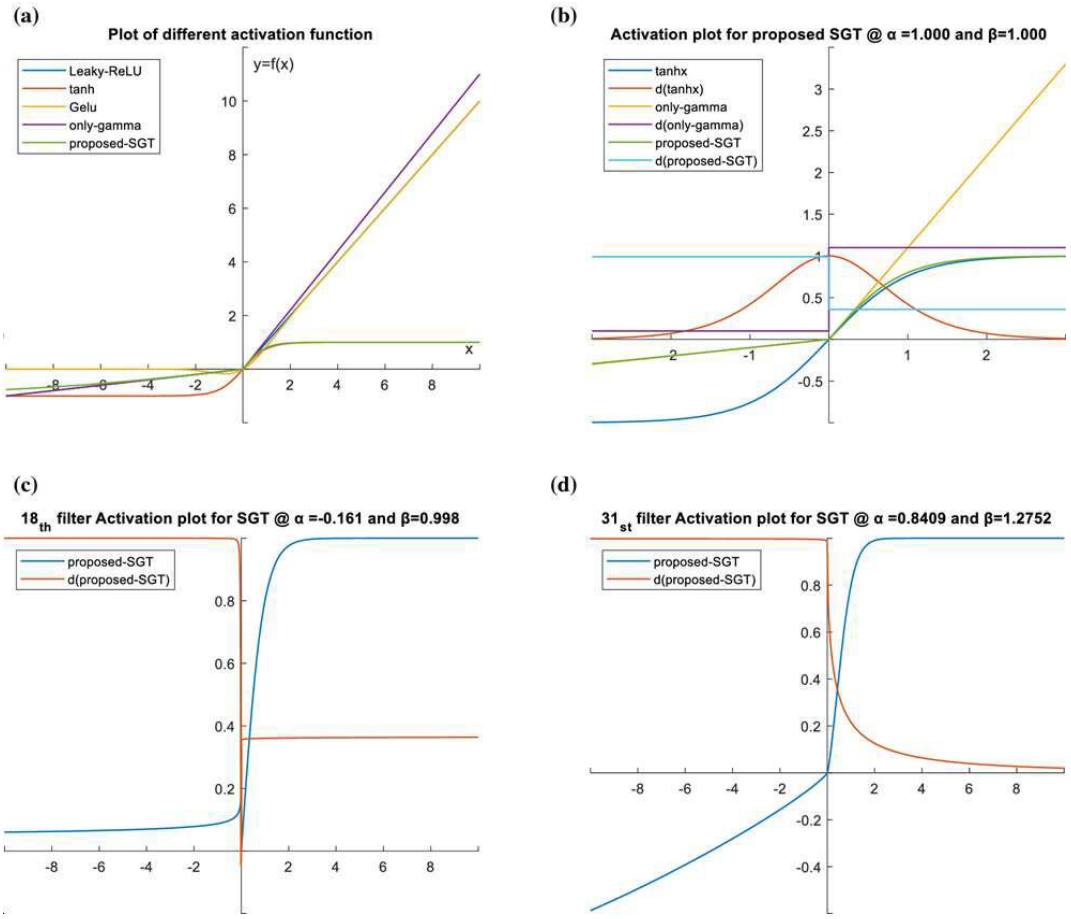
[0122] - DNN은 과적합이 매우 발생하기 쉬우므로 이에 대해 매우 확신할 수 없다. 그러나 동일한 훈련 환경(즉, 하이퍼 파라미터, 모델 및 훈련 자료)에서 과적합이 있는 경우 표준 활성화를 사용하는 경우와 SGT를 사용하는 경우 모두에 영향을 미칠 것이다. 따라서 두 경우 모두 결과가 편향되어 다른 활성화 함수와 SGT 특성을 비교하는 최종 목표가 (있는 경우) 과적합에 의해 동일하게 영향을 받는다. 그리고 매개 변수 레이어의 사용이 일부 과적합을 가져온다면 모든 경우에 대한 검증 및 테스트 정확도를 보고했으므로 결과는 여전히 SGT에 대해 설득력이 있다.

도면

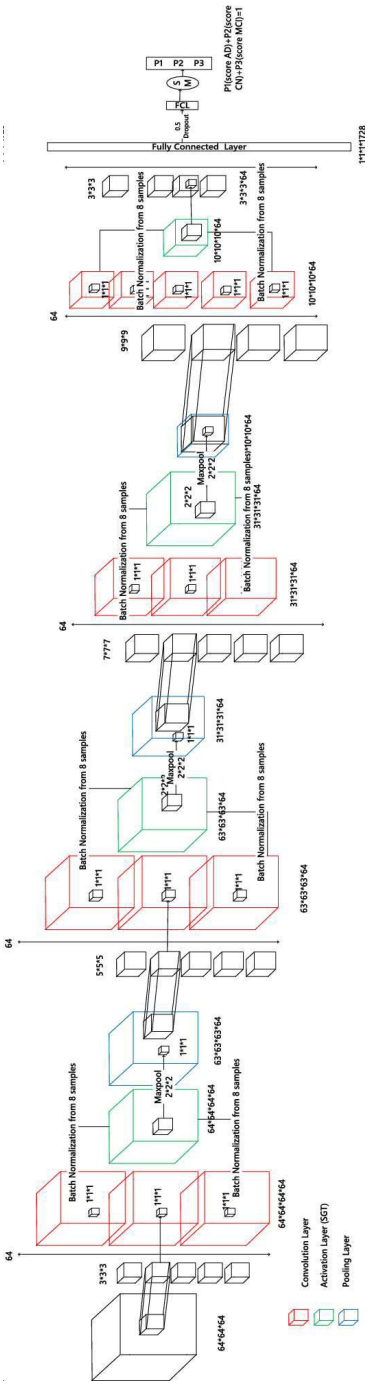
도면1



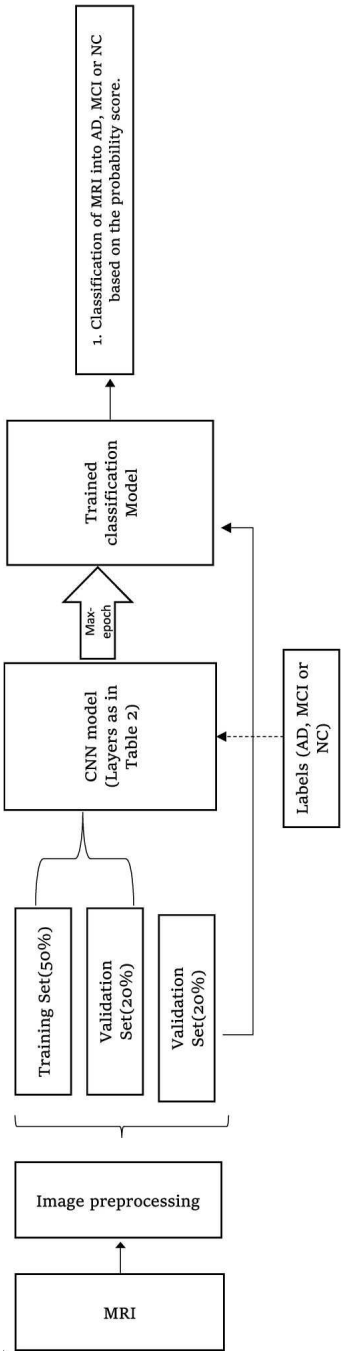
도면2


(a)
Plot of different activation function
Leaky-ReLU
tanh
Gelu
only-gamma
proposed-SGT
y=f(x)
x
(b)
Activation plot for proposed SGT @ α =1.000 and β=1.000
tanhx
d(tanhx)
only-gamma
d(only-gamma)
proposed-SGT
d(proposed-SGT)
(c)
18th filter Activation plot for SGT @ α =-0.161 and β=0.998
proposed-SGT
d(proposed-SGT)
(d)
31st filter Activation plot for SGT @ α =0.8409 and β=1.2752
proposed-SGT
d(proposed-SGT)

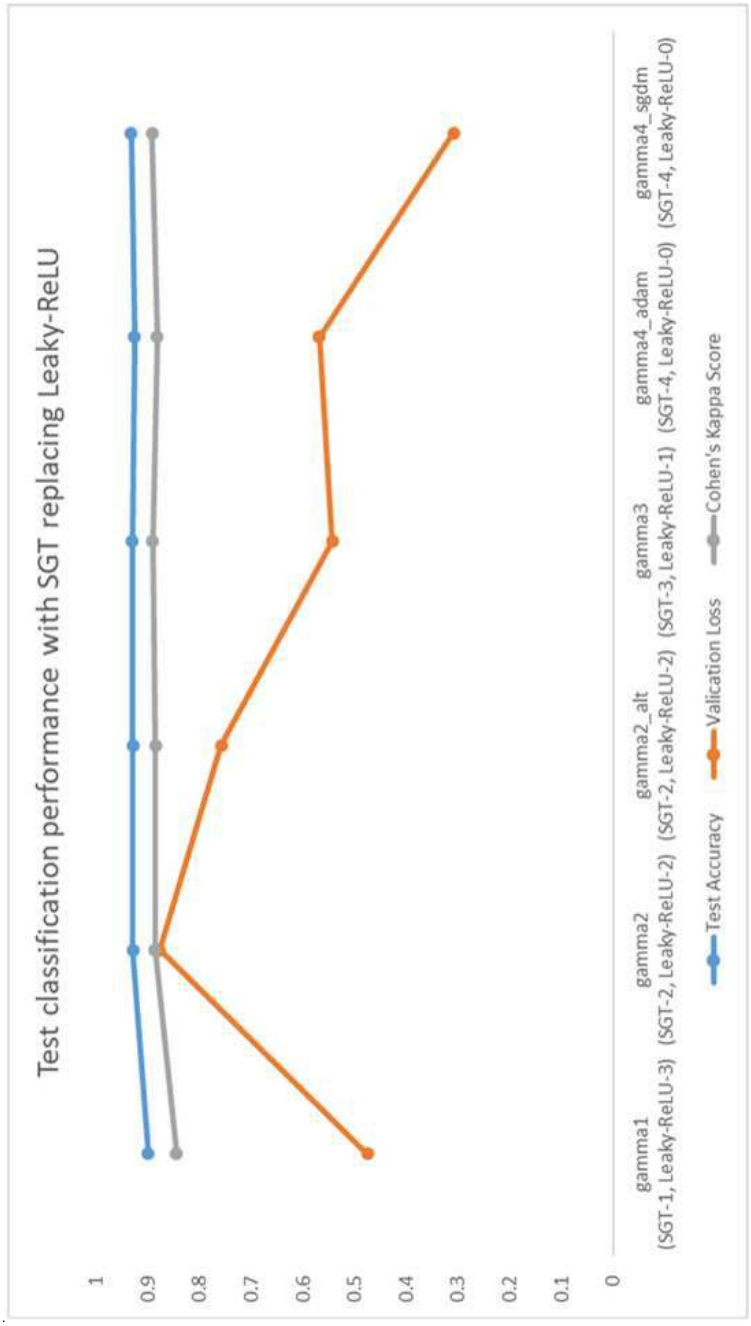
도면3


P1 P2 P3
P1(score AD)+P2(score CN)+P3(score MCI)=1
S
M
0.5
Dropout
Fully Connected Layer
1*1*1728
3*3*3
3*3*3*64
Batch Normalization from 8 samples
10*10*10*64
1*1*1
64
9*9*9
Maxpool
2*2*2
2*2*2
31*31*31*64
7*7*7
63*63*63*64
5*5*5
64*64*64*64
Convolution Layer
Activation Layer (SGT)
Pooling Layer

도면4

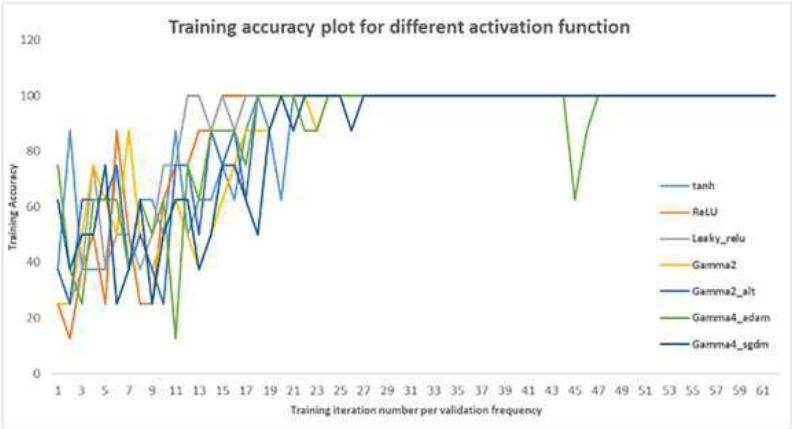

1. Classification of MRI into AD, MCI or NC based on the probability score.
Trained classification Model
Max-epoch
CNN model (Layers as in Table 2)
Labels (AD, MCI or NC)
Training Set(50%)
Validation Set(20%)
Validation Set(20%)
Image preprocessing
MRI

도면5

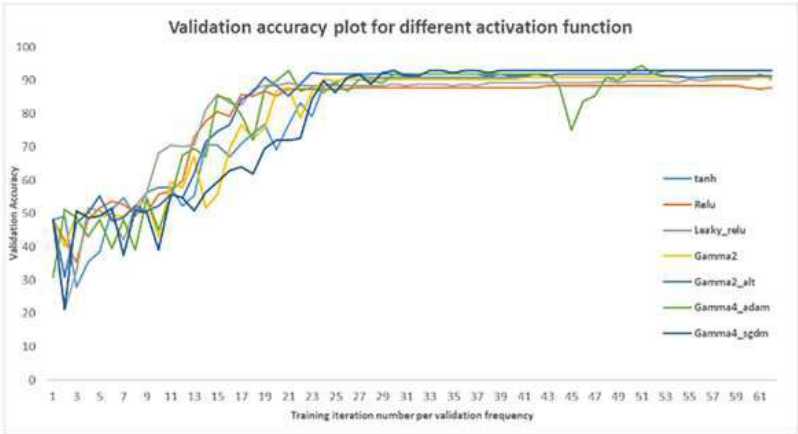

Test classification performance with SGT replacing Leaky-ReLU
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
gamma1 (SGT-1, Leaky-ReLU-3)
gamma2 (SGT-2, Leaky-ReLU-2)
gamma2_alt (SGT-2, Leaky-ReLU-2)
gamma3 (SGT-3, Leaky-ReLU-1)
gamma4_adam (SGT-4, Leaky-ReLU-0)
gamma4_sgdm (SGT-4, Leaky-ReLU-0)
Test Accuracy
Valication Loss
Cohen's Kappa Score

도면6

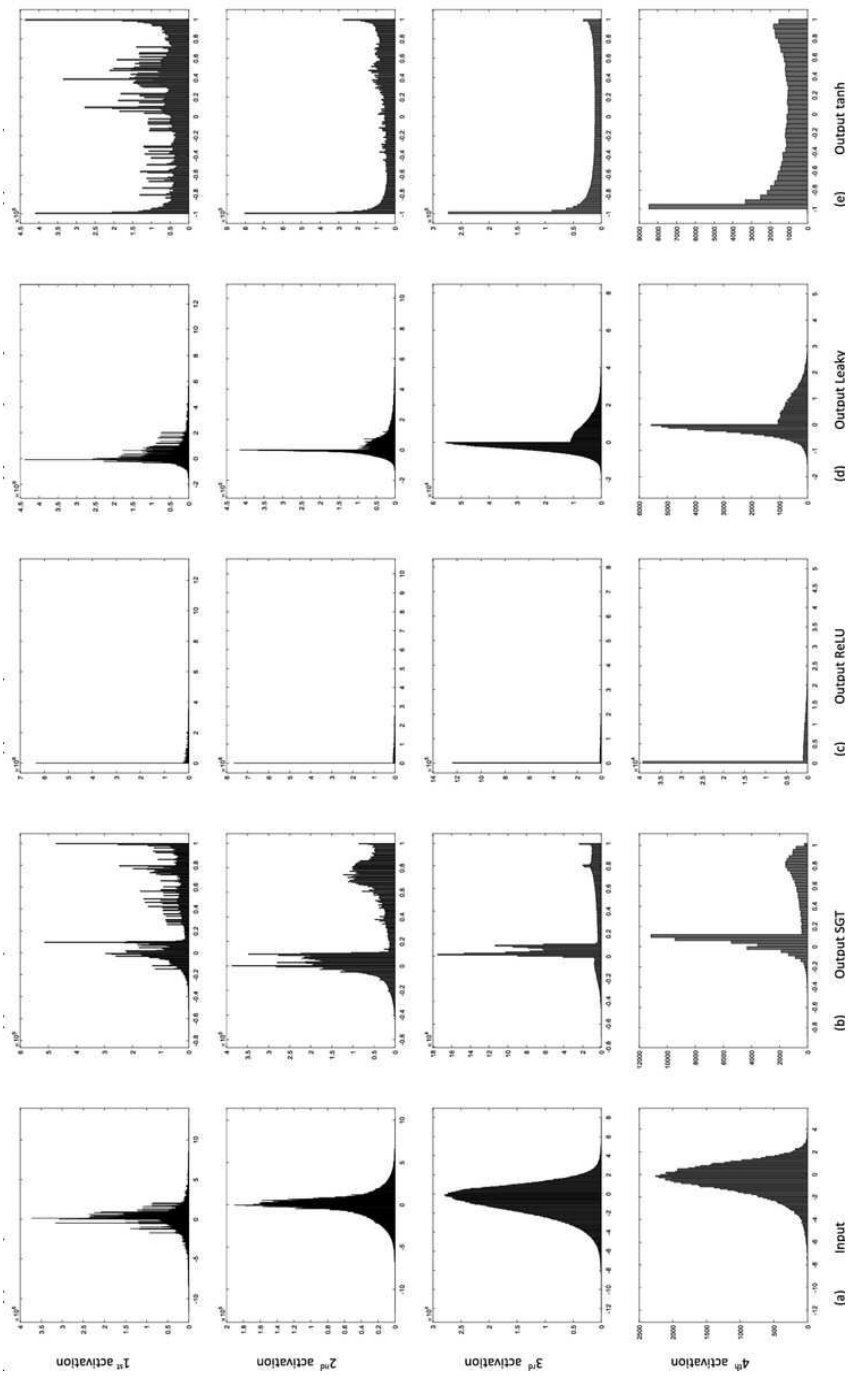
(a)


Training accuracy plot for different activation function
Training Accuracy
Training iteration number per validation frequency
tanh
ReLU
Leaky_relu
Gamma2
Gamma2_alt
Gamma4_adam
Gamma4_sgdm

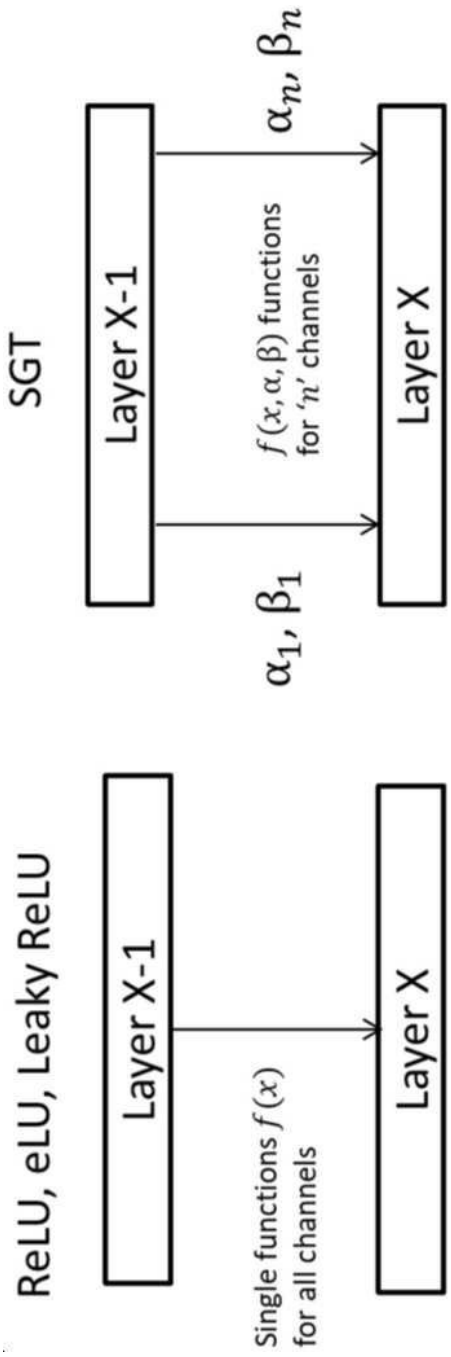
(b)


Validation accuracy plot for different activation function
Validation Accuracy
Training iteration number per validation frequency
tanh
Relu
Leaky_relu
Gamma2
Gamma2_alt
Gamma4_adam
Gamma4_sgdm

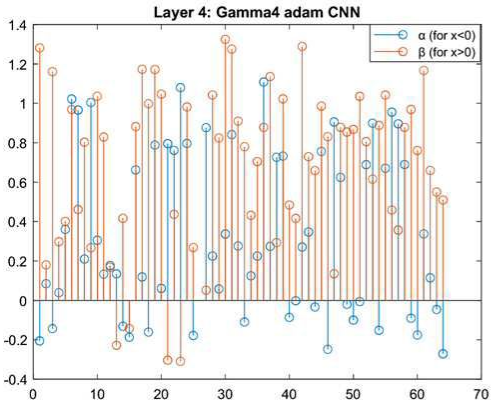
도면7

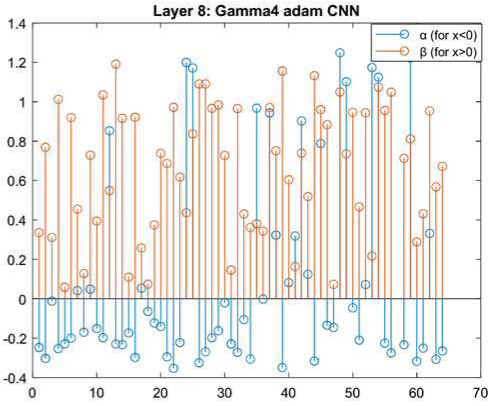

(a) Input
(b) Output SGT
(c) Output ReLU
(d) Output Leakv
(e) Output tanh
1st activation
2nd activation
3rd activation
4th activation

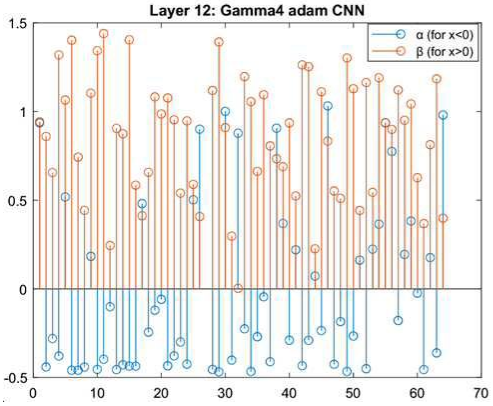
도면8

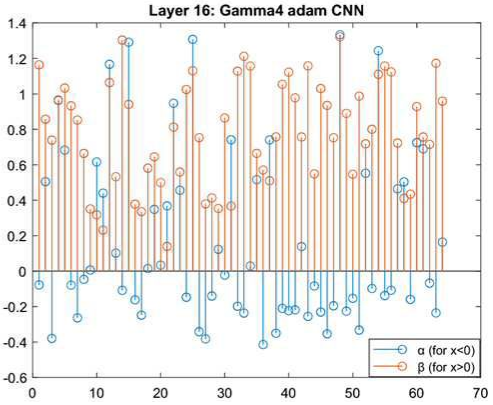

SGT
Layer X-1
Layer X
α_n, β_n
$f(x, \alpha, \beta)$ functions for 'n' channels
α_1, β_1
ReLU, eLU, Leaky ReLU
Layer X-1
Layer X
Single functions $f(x)$ for all channels

도면9

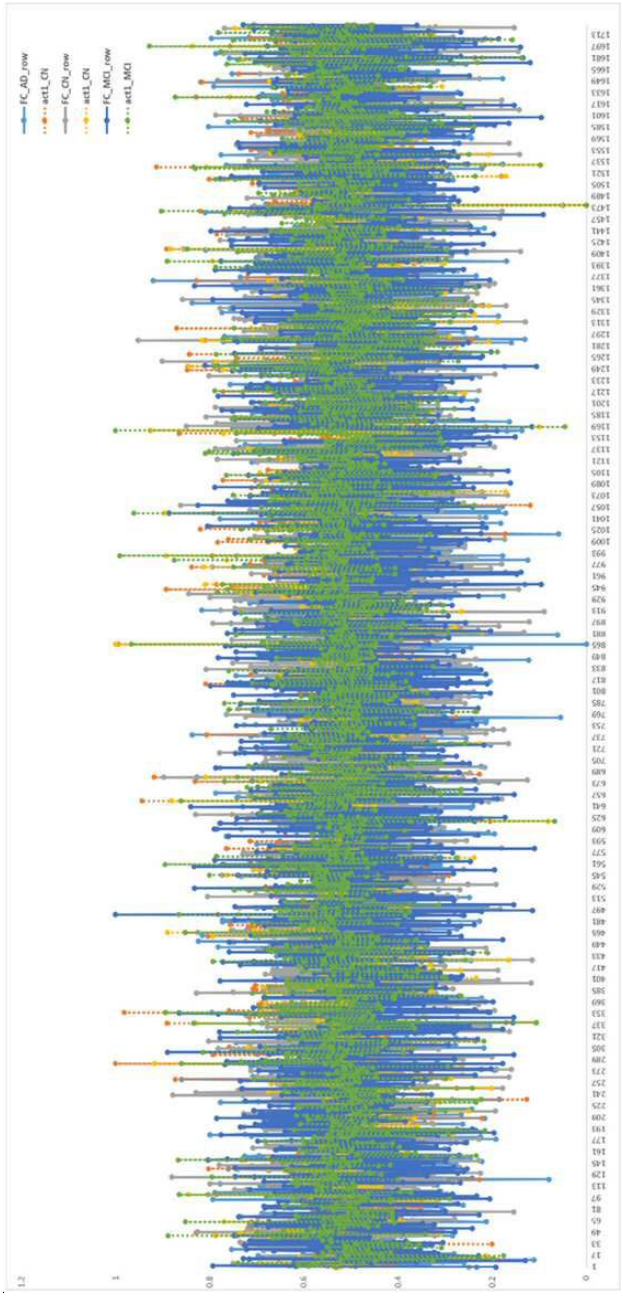

Layer 4: Gamma4 adam CNN
α (for x<0)
β (for x>0)


Layer 8: Gamma4 adam CNN
α (for x<0)
β (for x>0)

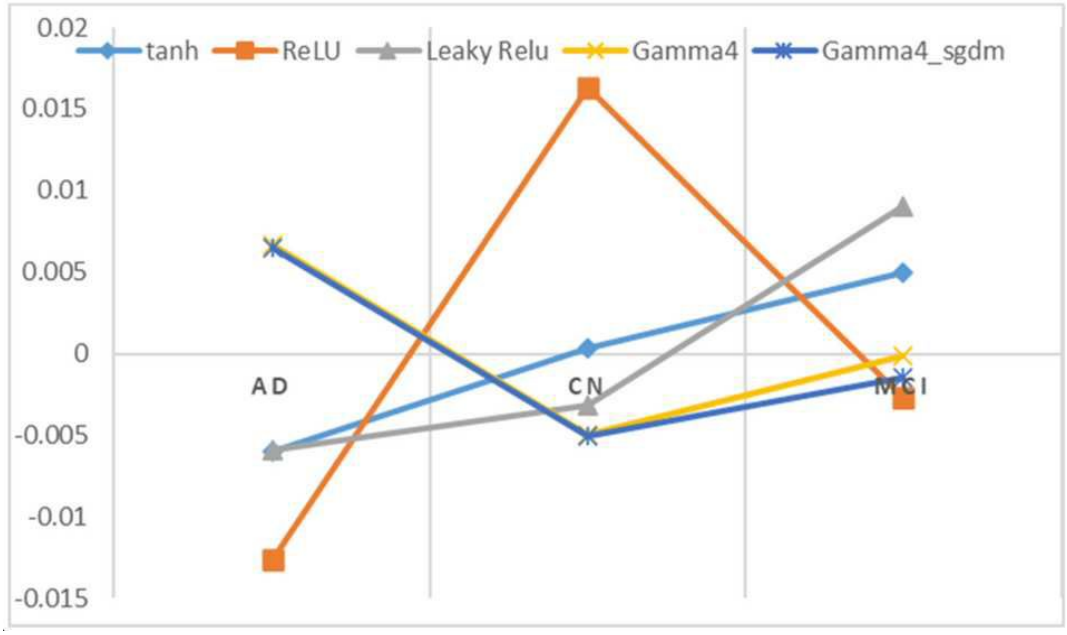

Layer 12: Gamma4 adam CNN
α (for x<0)
β (for x>0)


Layer 16: Gamma4 adam CNN
α (for x<0)
β (for x>0)

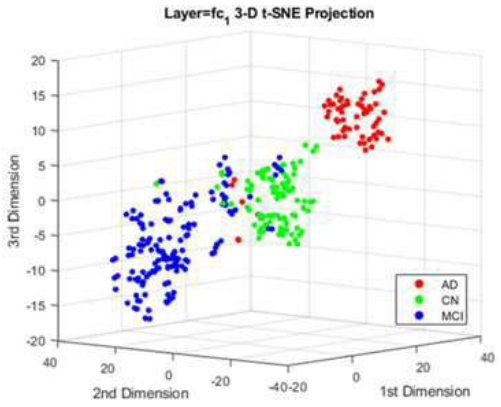
도면10



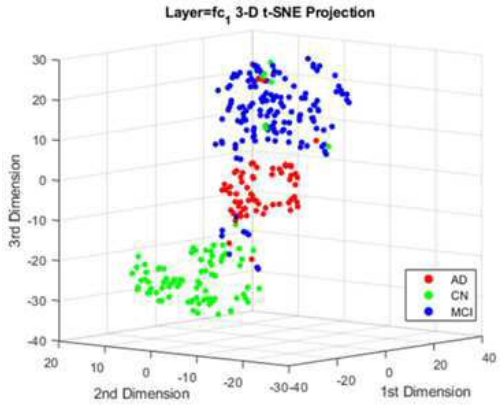
도면11


0.02
0.015
0.01
0.005
0
-0.005
-0.01
-0.015
tanh
ReLU
Leaky Relu
Gamma4
Gamma4_sgdm
AD
CN
MCI

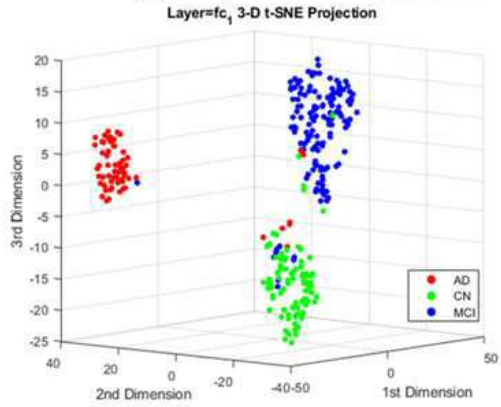
도면12



(a) ReLU activated CNN



(b) Leaky-ReLU activated CNN



(c) SGT activated CNN