



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2025-0020953
(43) 공개일자 2025년02월11일

(51) 국제특허분류(Int. Cl.)
G06V 10/764 (2022.01) G06V 10/46 (2022.01)
G06V 10/774 (2022.01)
(52) CPC특허분류
G06V 10/764 (2023.08)
G06N 3/0985 (2023.01)
(21) 출원번호 10-2023-0102344
(22) 출원일자 2023년08월04일
심사청구일자 2023년08월04일

(71) 출원인
고려대학교 산학협력단
서울특별시 성북구 안암로 145, 고려대학교 (안암
동5가)
(72) 발명자
고종현
광주광역시 남구 노대실로 49, 605동 1404호
정원주
서울특별시 강남구 삼성로64길 5, 107동 302호
(74) 대리인
김동용

전체 청구항 수 : 총 5 항

(54) 발명의 명칭 아이젠이미지를 이용한 퓨샷러닝 이미지 분류 모델 생성 장치 및 방법

(57) 요약

퓨샷러닝(Few-shot learning)을 이용한 이미지 분류 모델 생성 장치 및 방법이 개시된다. 상기 이미지 분류 모델 생성 방법은 적어도 프로세서(processor)를 포함하는 컴퓨팅 장치에 의해 수행되는, 퓨샷러닝을 이용한 이미지 분류 모델 생성 방법으로써, 상기 이미지 분류 모델 생성 방법은 복수의 에피소드들로 구성된 학습을 수행하는 에피소드 학습 방법을 이용하고, 각 에피소드는, 학습 이미지를 포함하는 학습 데이터 세트와 쿼리 이미지를 포함하는 쿼리 데이터 세트를 획득하는 단계, 학습 이미지를 열벡터로 변환하는 단계, 아이젠이미지(eigenimage)를 생성하는 단계, 아이젠 이미지를 이용하여 상기 쿼리 이미지를 재구성하여 재구성된 쿼리 이미지를 생성하는 단계, 보조 손실 함수를 산출하는 단계, 및 미리 정해진 손실 함수와 상기 보조 손실 함수의 가중합을 손실 함수로 하여 이미지 분류 모델의 파라미터를 업데이트하는 단계를 포함한다.

대표도 - 도1



(52) CPC특허분류

G06V 10/469 (2023.08)

G06V 10/774 (2023.08)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711193622
과제번호	RS-2021-II212068
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	정보통신방송혁신인재양성
연구과제명	인공지능 혁신 허브 연구 개발
기 여 율	1/2
과제수행기관명	고려대학교산학협력단
연구기간	2023.01.01 ~ 2023.12.31

이 발명을 지원한 국가연구개발사업

과제고유번호	1711193973
과제번호	2019-0-00079-005
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	정보통신방송혁신인재양성
연구과제명	인공지능대학원지원(고려대학교)
기 여 율	1/2
과제수행기관명	고려대학교산학협력단
연구기간	2023.01.01 ~ 2023.12.31

명세서

청구범위

청구항 1

적어도 프로세서(processor)를 포함하는 컴퓨팅 장치에 의해 수행되는, 퓨샷러닝(Few-shot learning)을 이용한 이미지 분류 모델 생성 방법에 있어서,

상기 이미지 분류 모델 생성 방법은 복수의 에피소드들로 구성된 학습을 수행하는 에피소드 학습 방법을 이용하고, 각 에피소드는,

학습 이미지를 포함하는 학습 데이터 세트와 쿼리 이미지를 포함하는 쿼리 데이터 세트를 획득하는 단계;

학습 이미지를 열벡터로 변환하는 단계;

아이겐이미지(eigenimage)를 생성하는 단계;

아이겐 이미지를 이용하여 상기 쿼리 이미지를 재구성하여 재구성된 쿼리 이미지를 생성하는 단계;

보조 손실 함수를 산출하는 단계; 및

미리 정해진 손실 함수와 상기 보조 손실 함수의 가중합을 손실 함수로 하여 이미지 분류 모델의 파라미터를 업데이트하는 단계를 포함하는,

이미지 분류 모델 생성 방법.

청구항 2

제1항에 있어서,

상기 학습 데이터 세트는 C(C는 2 이상의 자연수) 개의 클래스(class) 각각에 대한 k(k는 임의의 자연수) 개의 학습 이미지를 포함하고,

상기 쿼리 데이터 세트는 C 개의 클래스 각각에 대한 q(q는 임의의 자연수) 개의 쿼리 이미지를 포함하는,

이미지 분류 모델 생성 방법.

청구항 3

제2항에 있어서,

상기 아이겐이미지를 생성하는 단계는, 각 클래스별로,

열벡터로부터 열벡터의 평균값을 차감하여 정규화 벡터를 생성하는 단계;

정규화 벡터 각각을 열(column)의 구성으로 포함하는 행렬의 공분산행렬을 생성하는 단계; 및

공분산행렬의 아이겐벡터를 상기 아이겐이미지로 도출하는 단계를 포함하는,

이미지 분류 모델의 생성 방법.

청구항 4

제3항에 있어서,

상기 재구성된 쿼리 이미지를 생성하는 단계는 수학식을 이용하여 클래스 a에 대한 상기 재구성된 쿼리 이미지($r(a)$)를 생성하고,

상기 수학식은

$$r(a) = \psi^a + \sum_{i=1}^k w_i^a v_i^a$$

이고,

상기 ψ^a 는 열벡터의 평균값을 나타내고,

상기 v_i^a 는 상기 아이겐이미지이고,

상기 w_i^a 는 가중치인,

이미지 분류 모델의 생성 방법.

청구항 5

제4항에 있어서,

상기 보조 손실 함수를 산출하는 단계는 수학적식 2를 상기 보조 손실 함수($J_{\text{eig}}(x_j)$)로 결정하고,

상기 수학적식 2는 $J_{\text{eig}}(x_j) = -\log p(y_j = a | x_j)$ 이고,

상기 x_j 는 j번째 쿼리 이미지이고,

상기 p는 수학적식 3에 의해 산출되고,

상기 수학적식 3은
$$p(y_j = a | x_j) = \frac{\exp(-d(x_j, r(a)))}{\sum_{a=1}^C \exp(-d(x_j, r(a)))}$$
 이고,

상기 $d(\cdot)$ 는 벡터화된 이미지들 사이의 거리를 나타내는,

이미지 분류 모델의 생성 방법.

발명의 설명

기술 분야

[0001] 본 발명은 퓨샷러닝(Few-shot learning) 이미지 분류 방법에 관한 것으로, 특히 아이겐이미지(eigenimages)를 이용하여 퓨샷러닝 모델 학습시 과적합(overfitting)을 줄이기 위해 사용할 수 있는 보조 손실 함수에 관한 것이다.

배경 기술

[0002] 최근 딥 러닝(Deep Learning, DL) 기술의 발전으로 머신 러닝의 여러 영역에서 많은 발전이 이루어 졌다. 이러한 발전은 많은 수의 데이터 세트로부터 엄청난 양의 지식을 보존할 수 있는 딥 러닝 모델을 훈련하는 능력에 크게 의존한다. 그러나 많은 이미지 분류 작업의 경우 각 클래스에 대해 많은 레이블이 지정된 데이터를 사용할 수 없다. 따라서 레이블이 지정된 데이터 몇 개만으로 만족스러운 분류 성능을 반환하는 딥 러닝 기술이 필요하다. 그 결과 더 작은 데이터셋을 입력 받아도 되는 분류기를 개발하려는 퓨샷러닝(Few-shot learning)에 대한 연구가 많은 관심을 끌고 있다.

[0003] 퓨샷러닝은 각 클래스에 대해 몇 개의 레이블이 지정된 데이터만이 표시된 상태에서 학습 중에 관찰되지 않은 보이지 않는 클래스를 분류하도록 모델 분류기를 학습해야 하는 개념이다.

[0004] 기존 대부분의 퓨샷러닝 알고리즘들은 메타러닝(meta-learning) 기법을 사용한다. 메타러닝에서 훈련 세트는 아직 모델이 본적이 없는 테스트 때의 상황의 시뮬레이션 역할을 하는 연속적인 새로운 작업(tasks, 에피소드(episodes)라고도 불림)으로 제공된다. 각 클래스에 대해 레이블이 있는 서포트(Support) 이미지 몇 개를 제공한 후 모델은 레이블이 없는 쿼리(Query) 이미지 세트에 대한 클래스를 올바르게 추측하여야 한다. 이러한 퓨샷러닝의 가장 큰 문제는 한 클래스에 대해서 레이블이 있는 데이터의 개수가 적기 때문에 모델 학습시 오버피팅(과적합)이 일어난다는 것이다. 오버피팅을 줄이기 위해 주로 (1)메트릭 학습(metric-learning) 기반 네트워크와 (2)모델 최적화(optimization) 기법이 사용되었다.

[0005] 퓨샷러닝에서 가장 많이 쓰이는 메트릭 학습 기반 네트워크는 데이터가 임베딩 공간에 투영되었을 때 가장 가까운 이웃 분류기 또는 선형 분류기의 도움으로 투영된 데이터의 특징들을 쉽게 식별할 수 있게 하는 것을 목표로 한다. 최적화 기반 방법은 잘 초기화된 옵티마이저를 훈련하는 것을 목표로 하여 훈련시 적은 수의 에폭(epoc

h)만으로도 보지 못했던 클래스에 빠르게 적응할 수 있도록 한다. 오버피팅을 줄이기 위해 최근 가장 많이 쓰이는 방법으로는 추가적인 손실 함수(보조 손실 함수로 불림)를 기존의 손실 함수에 더해 사용하는 것이다. 이러한 방법은 모델이 기존에 학습하지 못한 부분을 추가로 학습할 수 있게 해줘서 가장 높은 퓨샷 분류 성능을 낼 수 있게 해준다.

[0006] Cross Attention Network for Few-shot Classification은 기존에 사용한 최근접분류 손실 함수에 광역분류 손실 함수를 추가하여 사용한다. 광역분류 손실 함수는 임베딩 모듈이 정규화되는 데 도움을 준다. 광역분류 손실 함수의 효과는 ProtoNet을 최근접 분류 손실함수만 사용해서 학습시켰을 때와 광역분류 손실 함수를 같이 더해서 사용했을 때 모델의 분류성능을 비교함으로써 보였다.

[0007] 광역분류 손실 함수와 최근접분류 손실 함수를 선형결합한 손실 함수로 모델을 학습시켰을 때 미니이미지넷 데이터로 실험한 5-way 1-shot 문제에서 3.1%, 5-way 5-shot 문제에서 4.3% 성능이 향상되었다.

[0008] Transductive Information Maximization(TIM)은 기존에 있던 교차엔트로피 함수에 한계 엔트로피 손실 함수와 조건부 엔트로피 손실 함수 두 개의 새로운 손실 함수를 더하여 사용하였다. 한계 엔트로피 손실은 클래스 예측의 한계 분포를 균일하게 하여 결과적으로 쿼리 데이터가 특정 클래스로 과도하게 분류(예측)되는 것을 방지한다. 조건부 엔트로피 손실 함수는 레이블이 지정되지 않은 쿼리 샘플 예측의 사후 확률의 불확실성을 낮추는 것을 목표로 하며, 그 결과 모델이 보다 확실한 예측을 제공할 수 있다. 한계 엔트로피 손실 함수만 기존 교차 엔트로피 손실 함수와 선형적으로 결합했을 때 미니이미지넷 데이터를 사용한 5-way 1-shot 문제에서 8.1%, 5-way 5-shot 문제에서 2.5%의 성능향상을 보였다. 조건부 엔트로피 손실 함수까지 더해서 학습했을 때 5-way 1-shot 문제와 5-way 5-shot 문제에서 각각 6.6%, 3.0% 성능향상이 있었다.

[0009] 본 발명 또한 또한 서포트 데이터의 아이겐이미지를 사용하여 설계된 추가 손실 함수(아이겐로스 손실 함수라 명명함)를 만들어 기존 손실 함수에 추가한다. 퓨샷 학습 모델의 기존 손실 함수에 아이겐로스 손실 함수를 추가하면 모델이 훈련 중에 입력받는 서포트 이미지에 대하여 추가적인 색상 및 위치 특징을 학습할 수 있다.

[0010] 아이겐이미지는 주로 컴퓨터 비전 문제에서 사람 얼굴의 이미지 벡터의 고유 벡터를 지칭하는 용어다. "Low-dimensional procedure for the characterization of human faces."라는 논문에서 처음으로 매트릭스로 표현된 이미지에서 주성분분석(PCA)을 적용하여 이미지의 기저 특징을 얻을 수 있음을 보였다. 잘 정돈된 사람의 얼굴 사진들에서 만들어진 이미지 행렬의 공분산행렬의 고유벡터를 아이겐이미지(eigenimage)라고 부른다.

[0011] 레이블이 지정된 트레이닝 세트에 M개의 이미지가 있는 경우 주성분 분석을 통해 N개의 아이겐이미지로 구성된 기저 집합이 형성된다($N \leq M$). 사용하는 아이겐이미지가 많을수록 아이겐이미지로 이미지 재구성을 할 때 에러가 줄어든다. "Face recognition using eigenfaces" 논문은 아이겐이미지를 사용하여 자동화된 얼굴인식 시스템을 만들었다. 현재까지 아이겐이미지는 주로 정규화된 사진들만을 가지고 만들어지고 사용되었다. 예를 들어, 얼굴 인식에 사용될 때 모든 얼굴 이미지는 정사각형 이미지로 되어있고 정중앙에 코끝이 오게 되어있었다. 그럼에도 불구하고, 본 발명에서는 이러한 제한된 조건과 작업에서만 사용되었던 아이겐이미지를 일반적인 이미지(미니이미지넷 및 CUB) 분류에 사용하고자 한다.

선행기술문헌

특허문헌

- [0012] (특허문헌 0001) 대한민국 공개특허 제2021-0029110호 (2021.03.15. 공개)
(특허문헌 0002) 대한민국 공개특허 제2023-0047866호 (2023.04.10. 공개)
(특허문헌 0003) 대한민국 공개특허 제2021-0157128호 (2021.12.28. 공개)
(특허문헌 0004) 대한민국 공개특허 제2022-0084632호 (2022.06.21. 공개)

발명의 내용

해결하려는 과제

[0013] 퓨샷 학습 모델을 훈련시킬 때 분류 정확도가 만족스럽지 않은 것은 대부분 적은 수의 학습 데이터로 인한 과적합 때문이다. 본 발명에서는 과적합으로 인해 낮은 성능을 내는 퓨샷러닝 모델들의 성능을 향상시킬 수 있는 장

치 및 방법을 제안하고자 한다.

과제의 해결 수단

- [0014] 상술한 문제점들을 해결하기 위한, 아이겐이미지를 이용한 퓨샷러닝 이미지 분류 모델 생성 장치 및 방법이 개시된다.
- [0015] 일 실시예에 따른 이미지 분류 모델 생성 방법은 적어도 프로세서(processor)를 포함하는 컴퓨팅 장치에 의해 수행되는, 퓨샷러닝(Few-shot learning)을 이용한 이미지 분류 모델 생성 방법으로써, 상기 이미지 분류 모델 생성 방법은 복수의 에피소드들로 구성된 학습을 수행하는 에피소드 학습 방법을 이용하고, 각 에피소드는, 학습 이미지를 포함하는 학습 데이터 세트와 쿼리 이미지를 포함하는 쿼리 데이터 세트를 획득하는 단계, 학습 이미지를 열벡터로 변환하는 단계, 아이겐이미지(eigenimage)를 생성하는 단계, 아이겐 이미지를 이용하여 상기 쿼리 이미지를 재구성하여 재구성된 쿼리 이미지를 생성하는 단계, 보조 손실 함수를 산출하는 단계, 및 미리 정해진 손실 함수와 상기 보조 손실 함수의 가중합을 손실 함수로 하여 이미지 분류 모델의 파라미터를 업데이트하는 단계를 포함한다.

발명의 효과

- [0016] 본 발명에 의한 경우, 아이겐로스 손실 함수라는 간단한 보조 손실 함수를 기존 손실 함수에 추가함으로써 퓨샷러닝 모델들의 오버피팅을 감소시킬 수 있다. 아이겐이미지에서 파생된 손실 함수 항을 기존 손실 함수에 더해 줌으로써 모델이 추가 매개변수(weight parameter)를 학습하지 않고도 서포트 데이터의 추가적인 특징을 학습할 수 있게 한다.
- [0017] 아이겐로스의 높은 실용성 덕분에 기존에 있는 최신 퓨샷러닝 모델들에 활용함으로써 오버피팅을 줄여 더욱 높은 분류 성능을 낼 수 있다.

도면의 간단한 설명

- [0018] 본 발명의 상세한 설명에서 인용되는 도면을 보다 충분히 이해하기 위하여 각 도면의 상세한 설명이 제공된다.
- 도 1은 5개의 서포트 데이터로 만든 5개의 아이겐이미지로 재구성한 목적 세 이미지를 도시한다.
- 도 2는 본 발명에서 제안하는 손실 함수를 도출하는 방법을 설명하기 위한 흐름도이다.
- 도 3과 도 4는 각각 5-way 5-shot과 5-way 1-shot 분류 태스크의 예시를 보여준다.
- 도 5는 본 발명의 일 실시예에 따른 퓨샷러닝 이미지 분류 모델 생성 방법을 설명하기 위한 흐름도이다.

발명을 실시하기 위한 구체적인 내용

- [0019] 본 명세서에 개시되어 있는 본 발명의 개념에 따른 실시예들에 대해서 특정한 구조적 또는 기능적 설명들은 단지 본 발명의 개념에 따른 실시예들을 설명하기 위한 목적으로 예시된 것으로서, 본 발명의 개념에 따른 실시예들은 다양한 형태로 실시될 수 있으며 본 명세서에 설명된 실시예들에 한정되지 않는다.
- [0020] 본 발명의 개념에 따른 실시예들은 다양한 변경들을 가할 수 있고 여러 가지 형태들을 가질 수 있으므로 실시예들을 도면에 예시하고 본 명세서에서 상세하게 설명하고자 한다. 그러나, 이는 본 발명의 개념에 따른 실시예들을 특정한 개시 형태들에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물, 또는 대체물을 포함한다.
- [0021] 제1 또는 제2 등의 용어는 다양한 구성 요소들을 설명하는데 사용될 수 있지만, 상기 구성 요소들은 상기 용어들에 의해 한정되어서는 안 된다. 상기 용어들은 하나의 구성 요소를 다른 구성 요소로부터 구별하는 목적으로만, 예컨대 본 발명의 개념에 따른 권리 범위로부터 벗어나지 않은 채, 제1 구성 요소는 제2 구성 요소로 명명될 수 있고 유사하게 제2 구성 요소는 제1 구성 요소로도 명명될 수 있다.
- [0022] 어떤 구성 요소가 다른 구성 요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성 요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성 요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성 요소가 다른 구성 요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는 중간에 다른 구성 요소가 존재하지 않는 것으로 이해되어야 할 것이다. 구성 요소들 간의 관계를 설명하는 다른 표현들, 즉 "~사이에"와 "바로 ~사이에" 또는 "~에 이웃하는"과 "~에 직접 이웃하는" 등

도 마찬가지로 해석되어야 한다.

[0023] 본 명세서에서 사용한 용어는 단지 특정한 실시예를 설명하기 위해 사용된 것으로서, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 명세서에서, "포함하다" 또는 "가지다" 등의 용어는 본 명세서에 기재된 특징, 숫자, 단계, 동작, 구성 요소, 부분품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성 요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.

[0024] 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 가진다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥상 가지는 의미와 일치하는 의미를 갖는 것으로 해석되어야 하며, 본 명세서에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.

[0025] 이하, 본 명세서에 첨부된 도면들을 참조하여 본 발명의 실시예들을 상세히 설명한다. 그러나, 특허출원의 범위가 이러한 실시예들에 의해 제한되거나 한정되는 것은 아니다. 각 도면에 제시된 동일한 참조 부호는 동일한 부재를 나타낸다.

[0026] 1) 퓨샷러닝 셋팅

[0027] 퓨샷러닝 문제는 일반적으로 학습 데이터와 테스트 데이터 모두에 서포트 세트와 쿼리 세트가 포함되어 있다. 본 발명에서는 퓨샷러닝 모델을 학습할 때 가장 효율적인 방법으로 알려진 에피소드 학습 방법이 사용될 수 있다. 각 에피소드(또는 태스크)에 대해 k 개(k 는 임의의 자연수이나, 일반적으로 1 또는 5의 값을 가짐) 레이블이 지정된 샘플이 있는 훈련 세트의 C (C 는 2 이상의 자연수)개 클래스가 서포트 세트 $S = (x_i, y_i)_{i=1}^m$ 로 무작위로 선택된다. 여기서 $m=C \times k$ 이다. x 는 이미지 이고 y 는 이미지에 해당하는 클래스 라벨이다. 이럴 때의 태스크는 C -way k -shot 분류(C -웨이 k -샷 분류)라고 한다. C 클래스 안에서 서포트데이터로 쓰이지 않고 남아있는 데이터에서 q 개의 데이터가 쿼리셋 $Q = (x_j, y_j)_{j=1}^n$ 으로 정의된다. 여기서, $n=C \times q$ 이다. 한 개의 에피소드 안에서 총 $C \times (k+q)$ 개의 데이터가 사용되는데 서포트 셋에서 $m=C \times k$ 개의 데이터가 사용되고 쿼리셋에서 $n=C \times q$ 개의 데이터가 사용된다. 에피소드의 목적은 서포트셋의 데이터들을 보여준 이후 모델이 학습을 하여 쿼리셋의 데이터를 보여줬을 때 알맞게 클래스 레이블을 예측하는 것이다. 새로운 에피소드를 지속적으로 학습함으로써 모델 안에 있는 매개변수들이 클래스별로 적은 데이터만을 사용하고도 이전에 보지 못한 클래스의 이미지들을 분류할 수 있게 학습된다.

[0028] 2) 아이젠이미지 생성

[0029] 본 발명에서는, 각 이미지 매트릭스의 차원 수를 줄이기 위해 주성분분석(PCA)을 적용하고 주요 특징을 추출할 수 있다. k 가 1보다 클 때, 각각의 클래스 a ($a=1, 2, \dots, C$)에 대하여 각각의 k 이미지들을 열벡터 $\Gamma_i^a (i=1, 2, \dots, k)$ 로 변환할 수 있다. 그 후, 수학적 1과 같이 Γ_i^a 의 평균값을 모든 벡터에서 차감할 수 있다. 차감의 결과로 생성된 벡터는 정규화 벡터로 명명될 수 있다.

[0030] [수학적 1]

$$\Phi_i^a = \Gamma_i^a - \psi^a \left(\psi^a = \frac{1}{k} \sum_{i=1}^k \Gamma_i^a \right)$$

[0031]

[0032] 그런 다음, 공분산행렬 $A_{cov}^a = (A^a)(A^a)^T$ 를 계산한다. 여기서, $A^a = [\Phi_1^a, \Phi_2^a, \dots, \Phi_k^a]$ 이다. 하지만, 위의 과정에서 $N \times N$ 크기의 이미지가 주어지면 $(A^a)(A^a)^T v_i^a = \lambda v_i^a$ 를 만족하는 아이젠벡터 v_i^a 를 구하기 위한 공분산행렬 A_{cov}^a 의 크기가 $N^2 \times N^2$ 가 된다. 이렇게 되면 너무 많은 양의 계산량이 요구됨으로, 적은 수의 차원에서 아이젠벡터를 구할 수 있는 다른 방법을 사용할 수 있다. 구체적으로, $(A^a)^T (A^a) u_i^a = \lambda u_i^a$ 를 만족하는 $(A^a)^T$ 의 공분산행렬의 아이젠벡터 u_i^a 을 이용할 수 있다. 그리고, A^a 를 u_i^a 앞에 곱함으로 A_{cov}^a 의 아이젠벡터는 다음 수학적 2와 같이 구해진다.

[0033] [수학식 2]

[0034]
$$A^a u_i^a = v_i^a \ (i = 1, 2, \dots, k)$$

[0035] 위의 수식을 이용하면 일반적인 퓨샷러닝에서 k 는 N^2 보다 훨씬 작기 때문에 아이젠벡터를 구하기 위한 계산량의 차수(order)가 N^2 에서 k 로 현저히 줄어든다. 이렇게 구한 아이젠벡터 v_i^a 는 클래스 a 의 아이젠이미지로 간주된다.

[0036] k 가 1인 경우에는(C-way 1-shot 태스크), 수학식 1의 정규화벡터 Φ_1^a 가 영벡터가 된다. 본 발명에서 C-way 1-shot 태스크에서는 클래스 a 안에 있는 한 개의 서포트 데이터를 한 개의 아이젠이미지로 간주한다.

[0037] **3) 쿼리이미지 재구성**

[0038] 각각의 에피소드에서 쿼리데이터 x_j 가 주어졌을 때 각 클래스에서 만든 k 개의 아이젠이미지들을 선형결합하여 x_j 를 재구성한다. 각각의 아이젠이미지들을 더할 때 사용되는 가중치는 다음 수학식 3과 같이 정의된다.

[0039] [수학식 3]

[0040]
$$w_i^a = v_i^a \cdot (x_j - \psi^a)$$

[0041] 가중치는 쿼리 이미지를 재구성할 때 각 아이젠이미지의 기여도를 의미한다. 결과적으로 재구성된 이미지 $r(a)$ 는 다음 수학식 4와 같이 표현된다.

[0042] [수학식 4]

[0043]
$$r(a) = \psi^a + \sum_{i=1}^k w_i^a v_i^a$$

[0044] k 개의 아이젠이미지를 각각의 클래스에서 사용하여, 한 개의 클래스별로 한 개씩, 총 C 개의 재구성 이미지가 만들어진다.

[0045] **4) 아이젠로스 손실 함수(보조 손실 함수)**

[0046] 각 클래스별로 재구성이미지 $r(a)$ 를 생성한 후에, 수학식 5와 같이, 쿼리 이미지와 재구성된 쿼리 이미지의 거리를 입력값으로 받는 소프트맥스 함수를 사용하여 레이블을 모르는 쿼리 데이터 x_j 에 대해 클래스간 분포를 생성한다.

[0047] [수학식 5]

[0048]
$$p(y_j = a | x_j) = \frac{\exp(-d(x_j, r(a)))}{\sum_{a=1}^C \exp(-d(x_j, r(a)))}$$

[0049] 벡터화된 이미지들 사이의 거리를 계산하는 함수 d 는 유클리디언 거리를 사용할 수 있다. 벡터공간 $\mathbb{R}^n$ 에서 두 개의 이미지벡터 A 와 B 사이의 거리 d 는 다음 수학식 6과 같이 정의된다.

[0050] [수학식 6]

[0051]
$$d(A, B) = \sqrt{\sum_{i=1}^n (A_i - B_i)^2}$$

[0052] 위 수학식 6에서 i 는 이미지 벡터의 각 원소를 뜻한다. 도 3과 도 4는 각각 5-way 5-shot과 5-way 1-shot 분류 태스크의 예시를 보여준다. 쿼리 이미지에서 재구성 이미지까지의 거리는 쿼리 데이터와 동일한 클래스의 아이젠이미지를 사용하여 재구성 이미지를 생성했을 때 가장 작다.

[0053] 그 후 아이젠 로스는 쿼리 이미지의 실제 클래스 a 에 대한 로그확률에 -1 을 곱한 것으로 다음 수학식 7과 같이 정의된다.

- [0054] [수학식 7]
- [0055] $J_{eig}(x_j) = -\log p(y_j = a | x_j)$
- [0056] 쿼리 데이터와 재구성 이미지 $r(a)$ 와의 거리가 작으면, 수학식 7에서의 p 는 1에 가까워져서 작은 로스값이 나오게 된다. 수학식 8에서와 같이 기존의 모델들의 로스 함수에 아이겐 로스를 가중치를 주어 더해준 후 학습함으로써 아이겐 로스는 퓨샷러닝 분야에서 사용될 수 있다. 수학식 8에서 $J_{\phi}(x_j)$ 는 기존의 퓨샷러닝 모델들에서 사용된 로스 함수이다. 상수 λ 는 아이겐 로스가 기존 손실에 영향을 미치는 정도를 결정하는 데 사용된다.
- [0057] [수학식 8]
- [0058] $L = J_{\phi}(x_j) + \lambda \cdot J_{eig}(x_j)$
- [0059] [실시예]
- [0060] 기존에 발표된 논문들 ProtoNet(Snell, J., Swersky, K., Zemel, R. (2017). Prototypical networks for few-shot learning. Advances in neural information processing systems, 30.), SS AmdimNet(Chen, D., Chen Y., Li, Y., et al. (2021). Self-supervised learning for few-shot image classification. In ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 1745-1749). IEEE.), TIM(Boudiaf, M., Ziko, I., Rony, J., et al. (2020). Information maximization for few-shot learning. Advances in Neural Information Processing Systems, 33, 2445-2457.), CAN(Hou, R., Chang, H., MA, B., et al. (2019). Cross attention network for few-shot classification. Advances in Neural Information Processing Systems, 32.), 그리고 ICI(Wang, Y., Xu, C., Liu, C., et al. (2020). Instance Credibility Inference for Few-Shot Learning. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 12836-12845).) 논문에서 제안한 모델들에, 본 발명에서 제안하는 아이겐 로스를 추가하여 퓨샷러닝을 학습시켜 분류성능을 알아보는 실험을 수행하였다. 또한 CAN과 TIM에서 제안한 손실 함수와 아이겐 로스까지 더하여 모델들을 학습시키는 실험도 하였다.
- [0061] [사용한 데이터셋]
- [0062] 퓨샷러닝 모델들을 학습할 때 많이 사용되는 미니이미지넷(MiniImageNet, Vinyals, O., Blundell, C., Lillicrap., et al. (2016). Matching networks for one shot learning. Advances in neural information processing systems, 29.)과 CUB(Wah, C., Branson, S., Welinder, P., et al. (2011). The caltech-ucsd birds-200-2011 dataset.) 그리고 TieredImageNet(Ren, M., Triantafillou, E., Ravi, S., et al. (2018). Metalearning for semi-supervised few-shot classification. arXiv preprint arXiv:1803.00676.)을 사용하였다.
- [0063] 미니이미지넷은 64개 클래스의 학습셋, 16개 클래스의 검증셋, 그리고 20개 클래스의 테스트셋으로 이루어졌고, 각각 84×84 크기의 이미지가 100개씩 포함되어 있다. TieredImageNet은 351개의 학습셋, 97개의 검증셋, 그리고 160개의 테스트셋으로 이루어져 있다. CUB는 200개의 클래스의 84×84 크기의 새(bird) 이미지 데이터셋이다. 11788개의 데이터가 포함되어 있고 총 100개의 트레인셋, 50개의 검증셋, 그리고 50개의 테스트셋으로 나뉘어져 있다.
- [0064] [실험 세부사항]
- [0065] 5-way 5-shot 그리고 5-way 1-shot 세팅으로 학습 및 테스트를 진행하였다. ProtoNet, TIM, SS AmdimNet에서는 손실함수를 학습시키는데 ADAM 옵티마이저를 사용하였다. CAN과 ICI에서는 SGD 옵티마이저를 사용하였다.
- [0066] [상수 λ]
- [0067] 각각의 모델을 아이겐 로스 함수를 더하여 실험할 때 쓰이는 상수 λ 는 0에서 0.5 사이의 값중에 임의탐색방법(random search method)를 사용하여 검증 단계시 가장 높은 분류성능을 내는 λ 를 실험 단계에서도 사용하였다. 실험별로 사용된 λ 의 값은 표 1에 기재되어 있다.

[0068] [표 1]

Model	MiniImageNet		CUB		TieredImageNet	
	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
ProtoNet	0.01	0.0005	0.01	0.01	-	-
SS AmdimNet	0.3	0.35	0.0025	0.001	-	-
TIM	0.2	0.2	0.01	0.01	0.1	0.1
CAN	0.2	0.2	0.2	0.2	0.2	0.2
ICI	0.05	0.05	0.05	0.05	0.1	0.1

[0069]

[0070] [기존의 손실 함수에 아이겐 로스를 더하여 학습한 실험]

[0071] ProtoNet, SS AmdimNet, ICI 논문은 기존에 교차 엔트로피 손실 함수(CE)만을 사용하여 모델을 학습시켰다. CAN은 최근접 분류 손실 함수(NN)에 광역분류 손실 함수(Global)를 추가로 더하여 모델을 학습시켰다. TIM은 교차 엔트로피 손실 함수(CE)에 한계 엔트로피 손실 함수(ME)와 조건부 엔트로피 손실 함수(C) 두 개를 더하여 모델을 학습시켰다.

[0072] 실험을 진행한 모든 모델들에서 기존에 사용하는 손실함수만 사용하여 학습한 실험과 아이겐 로스를 더한 후 학습한 실험을 하였다. 실험 결과 모든 모델들에서 아이겐 로스 함수(Eigen)을 추가로 더하여 모델을 학습했을 때 더 높은 분류 성능을 내는 것을 확인하였다. 모델들별로 실험한 실험결과는 표 2 내지 표 6에 기재되어 있다.

[0073] [표 2]

Model	Loss	MiniImageNet		CUB	
		1-shot	5-shot	1-shot	5-shot
ProtoNet	CE	43.2	65.4	48.9	68.1
	CE+ME	45.1	66.2	48.5	68.0
	CE+ME+C	45.4	65.9	49.8	68.9
	CE+Global	45.7	68.8	50.8	69.1
	CE+Eigen	45.0	65.9	49.2	68.5
	CE+ME+C+Eigen	45.5	65.9	50.2	69.5
	CE+Global+Eigen	45.9	68.8	50.8	69.6

[0074]

[0075] 표 2는 상이한 손실 함수들을 사용하는 ProtoNet의 정확도를 나타낸다. 표 2에서 CE는 cross-entropy loss를, ME는 marginal entropy loss를, C는 conditional entropy loss를, NN은 nearest neighbor classification loss를, Eigen은 eigen loss를 나타낸다. 가장 높은 테스트 결과를 나타내는 손실 함수는 볼드체(bold)로 표시되어 있다. 이하의 표 3 내지 표 6에서도 표 2에서와 동일한 약어가 사용된다.

[0076] [표 3]

Model	Loss	MiniImageNet		CUB	
		1-shot	5-shot	1-shot	5-shot
SS AmdimNet	CE	63.84	81.21	64.64	78.51
	CE+ME	61.23	78.81	64.21	79.11
	CE+ME+C	60.81	79.12	64.44	78.81
	CE+Global	65.26	83.06	66.35	79.67
	CE+Eigen	64.29	81.29	65.01	79.09
	CE+ME+C+Eigen	60.79	81.06	65.07	78.51
	CE+Global+Eigen	65.53	83.17	66.68	80.45

[0077]

[0078] 표 3은 상이한 손실 함수들을 사용하는 AdminNet의 정확도를 나타낸다.

[0079] [표 4]

Model	Loss	MiniImageNet		CUB		TieredImageNet	
		1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
CAN	NN	60.31	74.83	65.62	79.44	65.54	77.26
	NN+ME	61.47	75.91	65.97	79.51	65.87	77.32
	NN+ME+C	62.65	75.69	67.02	80.76	67.25	77.79
	NN+Global	63.43	79.08	67.23	81.35	69.44	83.57
	NN+Eigen	60.95	74.92	65.89	79.61	66.06	77.45
	NN+ME+C+Eigen	62.94	76.11	67.17	80.85	67.30	78.42
	NN+Global+Eigen	64.04	79.19	67.50	81.52	69.68	83.98

[0080]

[0081] 표 4는 상이한 손실 함수들을 사용하는 CAN의 정확도를 나타낸다.

[0082] [표 5]

Model	Loss	MiniImageNet		CUB		TieredImageNet	
		1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
TIM	CE	58.77	79.05	68.01	85.25	66.89	84.26
	CE+ME	66.89	81.55	67.70	85.07	67.02	84.47
	CE+ME+C	73.45	84.56	79.23	88.40	80.38	88.40
	CE+Global	60.42	80.22	69.54	86.65	68.23	85.94
	CE+Eigen	59.83	79.56	68.81	85.93	67.11	84.85
	CE+ME+C+Eigen	73.88	84.71	81.29	89.74	81.71	88.74
	CE+Global+Eigen	61.27	80.63	70.78	86.69	68.56	86.22

[0083] 표 5는 상이한 손실 함수들을 사용하는 TIM의 정확도를 나타낸다.

[0085] [표 6]

Model	Loss	MiniImageNet		CUB		TieredImageNet	
		1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
ICI	CE	67.98	81.47	74.38	84.64	76.19	86.45
	CE+ME	68.78	82.43	74.89	85.46	76.34	87.30
	CE+ME+C	70.93	84.38	76.22	86.69	77.73	88.62
	CE+Global	70.29	83.58	76.18	86.20	77.65	88.56
	CE+Eigen	68.11	82.62	75.23	85.42	76.46	88.17
	CE+ME+C+Eigen	71.02	84.49	76.55	86.96	78.06	88.81
	CE+Global+Eigen	70.31	83.80	76.27	86.36	77.88	88.69

[0086] 표 6은 상이한 손실 함수들을 사용하는 ICI의 정확도를 나타낸다.

[0088] 도 5는 본 발명의 일 실시예에 따른 퓨샷러닝 이미지 분류 모델 생성 방법을 설명하기 위한 흐름도이다.

[0089] 도 5를 참조하면, 퓨샷러닝 이미지 분류 방법 등으로도 명명될 수 있는 이미지 분류 모델 생성 방법은 적어도 프로세서(processor) 및/또는 메모리(memory)를 포함하는 컴퓨팅 장치에 의해 수행될 수 있다. 즉, 이미지 분류 모델 생성 방법을 이루는 단계들 중 적어도 일부는 컴퓨팅 장치에 포함되는 프로세서의 동작으로 이해될 수도 있다. 컴퓨팅 장치는 PC(Personal Computer), 서버(server), 랩탑 컴퓨터, 태블릿 PC, 노트북, 스마트폰(smart phone), HMD(Head Mounted Display) 등을 포함할 수 있다. 또한, 이미지 분류 모델 생성 방법을 설명함에 있어 앞선 기재와 중복되는 내용에 관하여는 그 구체적인 기재를 생략하기로 한다.

[0090] 우선, 퓨샷러닝 이미지 분류 모델 생성 방법은 복수의 에피소드 학습 방법을 사용한다. 따라서, 이미지 분류 모델 생성 방법은 에피소드를 반복적으로 학습함으로써, 이미지 분류 모델을 생성할 수 있다. 이하에서는, 각 에피소드를 이루는 구체적인 과정에 대한 설명, 즉 각 에피소드에 대한 설명이다. 또한, 이미지 분류 모델은 C-way k-shot의 퓨샷러닝을 통해 학습된다.

[0091] 우선, 학습 데이터 세트와 쿼리 데이터 세트가 획득된다(S110). C-way k-shot이기 때문에, 학습 데이터 세트에는 C 개의 클래스 각각에 대한 k 개의 학습 이미지가 포함되고, 쿼리 데이터 세트에는 C 개의 클래스 각각에 대한 q 개의 학습 이미지가 포함된다.

[0092] 각 클래스별로, 학습 데이터 세트를 이루는 각 학습 이미지를 열벡터로 변환한다(S120).

[0093] 그런 다음, 각 클래스 별로, 학습 이미지에 대응하는 아이겐이미지를 생성한다(S130). 아이겐이미지를 생성하기 위해서는, 변환된 열벡터로부터 정규화 벡터를 생성한 후 아이겐이미지(또는 아이겐벡터)를 생성할 수 있다. 아이겐이미지의 생성 과정에 대한 상세한 설명은 생략하기로 한다.

[0094] 다음으로, 각 클래스 별로, 쿼리 이미지를 재구성한다(S140). 재구성된 쿼리 이미지는 수학적 4를 통해 생성될 수 있으며, 하나의 쿼리 이미지에 대한 C 개의 재구성된 쿼리 이미지가 생성된다.

[0095] 보조 손실 함수(아이겐 손실이나 아이겐 손실 함수로 명명될 수 있음)가 산출(또는 계산)된다(S150). 보조 손실 함수는 수학적 7에 의해 산출될 수 있다.

[0096] 마지막으로, 미리 정해진 손실 함수와 보조 손실 함수의 가중합을 손실 함수로 이용하여, 분류 모델의 파라미터 (예컨대, 가중치)가 업데이트된다. 미리 정해진 손실 함수는 이미지 분류 모델에 따라 사전에 정해진 손실 함수로써, 각 모델 마다 상이할 수 있다. 즉, 본 발명에서는 제한하고자 하는 핵심적인 구성은, 분류 모델의 분류 정확도를 높이기 위해 활용 가능한 보조 손실 함수에 관한 것이다. 따라서, 분류 모델의 오리지널 손실 함수는 모델별로 상이할 수 있으며, 이는 본 발명의 권리범위에 제한되지 않는다.

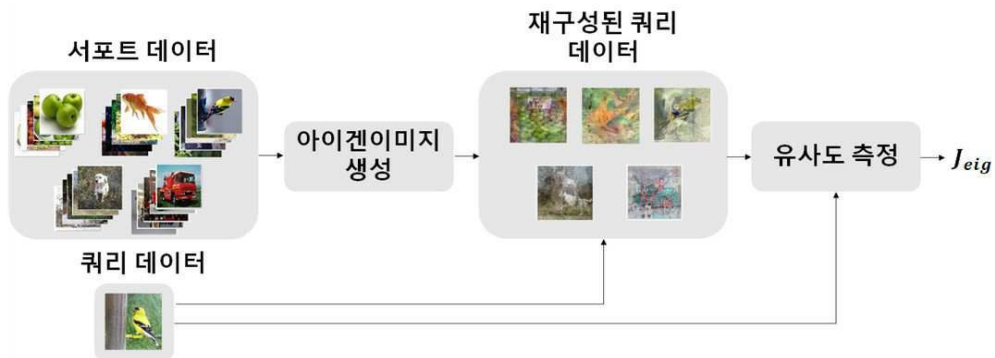
- [0097] 상술한 과정으로 구성된 에피소드를 반복적으로 수행함으로써, 보다 높은 분류 정확도를 갖는 이미지 분류 모델을 생성할 수 있다.
- [0098] 이상에서 설명된 장치는 하드웨어 구성 요소, 소프트웨어 구성 요소, 및/또는 하드웨어 구성 요소 및 소프트웨어 구성 요소의 집합으로 구현될 수 있다. 예를 들어, 실시예들에서 설명된 장치 및 구성 요소는, 예를 들어, 프로세서, 컨트롤러, ALU(Arithmetic Logic Unit), 디지털 신호 프로세서(Digital Signal Processor), 마이크로컴퓨터, FPA(Field Programmable array), PLU(Programmable Logic Unit), 마이크로프로세서, 또는 명령(instruction)을 실행하고 응답할 수 있는 다른 어떠한 장치와 같이, 하나 이상의 범용 컴퓨터 또는 특수 목적 컴퓨터를 이용하여 구현될 수 있다. 처리 장치는 운영 체제(Operation System, OS) 및 상기 운영 체제 상에서 수행되는 하나 이상의 소프트웨어 애플리케이션을 수행할 수 있다. 또한, 처리 장치는 소프트웨어의 실행에 응답하여, 데이터를 접근, 저장, 조작, 처리 및 생성할 수도 있다. 이해의 편의를 위하여, 처리 장치는 하나가 사용되는 것으로 설명된 경우도 있지만, 해당 기술 분야에서 통상의 지식을 가진 자는, 처리 장치가 복수 개의 처리 요소(Processing Element) 및/또는 복수 유형의 처리 요소를 포함할 수 있음을 알 수 있다. 예를 들어, 처리 장치는 복수 개의 프로세서 또는 하나의 프로세서 및 하나의 컨트롤러를 포함할 수 있다. 또한, 병렬 프로세서(Parallel Processor)와 같은, 다른 처리 구성(Processing Configuration)도 가능하다.
- [0099] 소프트웨어는 컴퓨터 프로그램(Computer Program), 코드(Code), 명령(Instruction), 또는 이들 중 하나 이상의 조합을 포함할 수 있으며, 원하는 대로 동작하도록 처리 장치를 구성하거나 독립적으로 또는 결합적으로(Collectively) 처리 장치를 명령할 수 있다. 소프트웨어 및/또는 데이터는, 처리 장치에 의하여 해석되거나 처리 장치에 명령 또는 데이터를 제공하기 위하여, 어떤 유형의 기계, 구성 요소(Component), 물리적 장치, 가상 장치(Virtual Equipment), 컴퓨터 저장 매체 또는 장치, 또는 전송되는 신호 파(Signal Wave)에 영구적으로, 또는 일시적으로 구체화(Embody)될 수 있다. 소프트웨어는 네트워크로 연결된 컴퓨터 시스템 상에 분산되어서, 분산된 방법으로 저장되거나 실행될 수도 있다. 소프트웨어 및 데이터는 하나 이상의 컴퓨터 판독 가능 기록 매체에 저장될 수 있다.
- [0100] 실시예에 따른 방법은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 매체에 기록되는 프로그램 명령은 실시예를 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(Magnetic Media), CD-ROM, DVD와 같은 광기록 매체(Optical Media), 플롭티컬 디스크(Floptical Disk)와 같은 자기-광 매체(Magneto-optical Media), 롬(ROM), 램(RAM), 플래시 메모리 등과 같은 프로그램 명령을 저장하고 수행하도록 특별히 구성된 하드웨어 장치가 포함된다. 프로그램 명령의 예에는 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드를 포함한다. 상기된 하드웨어 장치는 실시예의 동작을 수행하기 위해 하나 이상의 소프트웨어 모듈로서 작동하도록 구성될 수 있으며, 그 역도 마찬가지이다.
- [0101] 본 발명은 도면에 도시된 실시예를 참고로 설명되었으나 이는 예시적인 것에 불과하며, 본 기술 분야의 통상의 지식을 가진 자라면 이로부터 다양한 변형 및 균등한 타 실시예가 가능하다는 점을 이해할 것이다. 예를 들어, 설명된 기술들이 설명된 방법과 다른 순서로 수행되거나, 및/또는 설명된 시스템, 구조, 장치, 회로 등의 구성 요소들이 설명된 방법과 다른 형태로 결합 또는 조합되거나, 다른 구성 요소 또는 균등물에 의하여 대치되거나 치환되더라도 적절한 결과가 달성될 수 있다. 따라서, 본 발명의 진정한 기술적 보호 범위는 첨부된 등록청구범위의 기술적 사상에 의해 정해져야 할 것이다.

도면

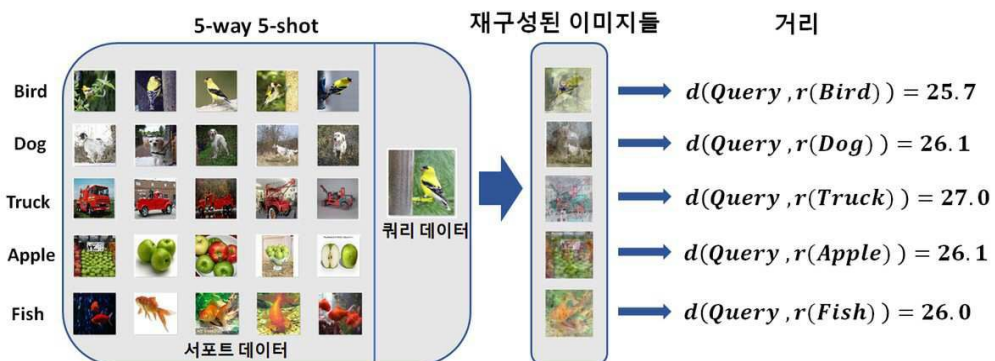
도면1



도면2



도면3



도면4



도면5

