Check for updates

Journal of the Korean Institute of Industrial Engineers
Vol. 52, No. 5, pp. 394-405, October 2026.
ISSN 1225-0988 | EISSN 2234-6457
https://doi.org/10.7232/JKIIE.2026.52.5.394
© 2026 KIIE

# 제로샷 이미지 분류 성능 개선을 위한 핵심 텍스트 디스크립터 선별 기법

박소연 · 김지현 · 김성범†

고려대학교 산업경영공학과

# Zero-Shot Visual Classification Through Discriminative Text Descriptor Selection

Soyeon Park · Jihyun Kim · Seoung Bum Kim

Department of Industrial and Management Engineering, Korea University

Vision-language models such as CLIP have demonstrated remarkable zero-shot classification performance by aligning visual and textual representations. Recent studies have shown that enriching class descriptions with large language model-generated visual descriptors can further improve classification accuracy without additional training. However, existing descriptor-based methods use all pre-generated descriptors equally for every test image, regardless of which descriptors are actually useful for a given input. This is problematic in fine-grained classification, where visually similar classes share many common attributes, causing non-discriminative descriptors to reduce the influence of truly informative ones. The purpose of this study is to address this issue by proposing CoDES (confusion-aware discriminative descriptor selection), a method that dynamically selects discriminative descriptors at test time. For each test image, CoDES first identifies confusing classes and then selects descriptors that are effective in resolving the confusion. Experiments demonstrate that CoDES consistently outperforms existing methods across eight benchmark datasets, achieving competitive performance against descriptor selection methods requiring labeled data without using any labeled data. These results indicate that the proposed CoDES can effectively improve zero-shot classification performance by selecting descriptors that are helpful for distinguishing between confusing classes.

***Keywords:*** Vision-language Model, CLIP, Zero-shot Classification, Descriptor Selection, Training-free Adaptation

## 1. 서 론

최근 대규모 이미지-텍스트 쌍으로 학습된 비전-언어 모델(vision-language model, VLM)은 다양한 시각 인식 태스크에서 뛰어난 성능을 보이고 있다. 특히 CLIP(contrastive language-image pre-training)은 이미지와 텍스트 간 의미적 관계를 효과적으로 학습한 대표적인 파운데이션 모델로, 별도의 추가 학습 없이 제로샷 이미지 분류가 가능하다는 점에서 큰 주목을 받고 있다(Radford *et al.*, 2021).

그러나 CLIP의 표준 제로샷 분류 방식은 “A photo of a {class}”와 같이 클래스 이름만으로 구성된 단순한 텍스트 프롬프트를 사용하기 때문에, 클래스가 가진 풍부한 시각적 특성을 충분히 반영하지 못한다는 한계가 있다. 이러한 한계는 특히 세밀한 분류(fine-grained classification) 문제에서 두드러진다. Fine-grained classification은 동일한 상위 범주(예: 동물, 자동차, 꽃, 음식)에 속하는 하위 클래스들을 구분하는 분류 문제로, 일반적인 이미지 분류와 달리 클래스 간 시각적 차이가 미세한 특성을 가진다(Dubey *et al.*, 2018; Wei *et al.*, 2021).

† 연락저자 : 김성범 교수, 02841 서울특별시 성북구 안암로 145 고려대학교 산업경영공학과, Tel : 02-3290-3397, E-mail : sbkim1@korea.ac.kr
2026년 4월 24일 접수; 2026년 5월 16일, 2026년 5월 27일 수정본 접수; 2026년 5월 27일 게재 확정.

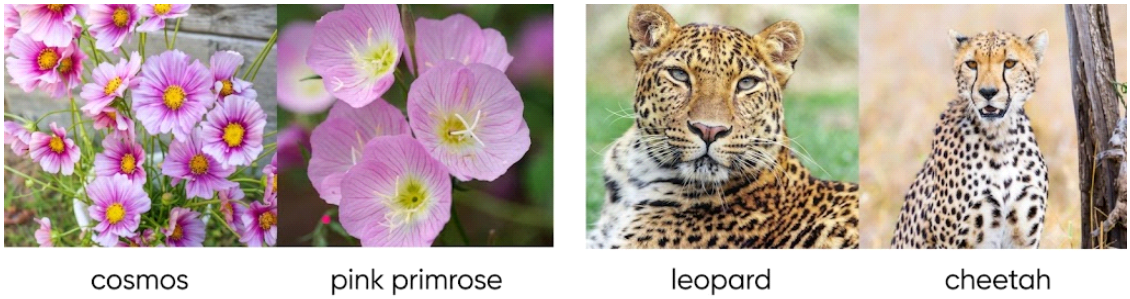



**Figure 1.** Examples of visually similar class pairs in fine-grained classification

<Figure 1>에서 보이는 바와 같이, 클래스를 구분하는 단서가 몸의 무늬, 꽃잎의 패턴과 같이 국소적이고 미세한 시각적 속성에 국한되기 때문에, fine-grained classification에서는 혼동 클래스(서로 유사한 클래스)간의 변별적 시각 단서를 정확히 포착하는 것이 핵심적이다.

CLIP의 표준적 제로샷 방식은 이러한 fine-grained classification 문제에서 높은 성능을 보이는 데 한계가 있다. 클래스 이름만으로 구성된 텍스트 임베딩은 상위 범주 수준의 의미 정보에 집중되기 때문에, 시각적으로 유사한 하위 클래스들 차이를 충분히 반영하지 못한다. 그 결과, 미세한 변별 단서가 텍스트 표현에 효과적으로 포함되지 못하며, 유사 클래스 간의 혼동이 발생하는 문제가 있다.

이러한 한계를 극복하기 위해, 최근에는 대규모 언어 모델(large language model, LLM)을 활용하여 각 클래스의 시각적 특징을 구체적으로 설명하는 디스크립터(descriptor)를 생성하고, 이를 클래스 표현에 통합함으로써 텍스트 임베딩을 풍부하게 만드는 연구들이 제안되었다. 대표적으로, Menon and Vondrick(2022)은 LLM에서 얻은 텍스트 디스크립터를 CLIP의 텍스트 프롬프트에 활용하는 방법(DCLIP)을 제안하고, 기존 CLIP 대비 분류 성능이 향상됨을 검증하였다. Pratt *et al.*(2023)은 LLM을 기반으로 클래스를 설명하는 완전한 문장 형태의 디스크립터들을 생성하고 이를 앙상블하여 활용하는 방법(CuPL)을 통해 다수의 제로샷 이미지 분류 벤치마크에서 성능 향상을 달성하였다. 이처럼 LLM이 생성한 다양한 형태의 텍스트 디스크립터를 활용하는 방법들은 클래스 이름만을 사용하는 기존 방식보다 모델에 풍부한 시각적 정보를 제공함으로써 추가적인 튜닝 없이 제로샷 분류 성능을 효과적으로 높인다.

그러나 이러한 디스크립터 활용 기반 방법론들도 다음과 같은 두 가지 근본적인 한계를 가진다. 첫 번째로, LLM은 실제 이미지가 아닌 클래스 이름만을 기반으로 디스크립터를 생성하기 때문에, 혼동 클래스들 사이에서 공통적으로 적용되는 변별력 없는 디스크립터가 포함될 수 있다. 예를 들어, 고양이 품종을 분류하는 경우 “has four legs”, “has fur”, “has a tail”과 같은 텍스트 디스크립터는 대부분의 고양이 품종에 공통적으로 나타나는 특징이다. 이러한 정보는 유사한 클래스 간 차이를 드러내지 못하기 때문에, 실제 분류 성능을 향상시키는 데에는 기여를 하지 못한다. 이처럼 변별력 없는 디스크립터들이 전체 평균에 포함되면, 실제로 클래스를 구별하는 데 유효한 디스크립터들의 영향력이 줄어든다. 두 번째로, 어떠한 디스크립터가 분류에 도움이 되는지는 입력 이미지마다 다르다. 이는 입력 이미지마다 모델이 혼동하는 클래스가 다르며, 각 혼동을 해소하는 데 필요한 핵심 특징 또한 달라지기 때문이다. 예를 들어, 동일한 고양이 이미지라도 어떤 이미지는 페르시안 고양이와 혼동될 수 있고, 다른 이미지는 랙돌(ragdoll) 고양이와 혼동될 수 있다. 이처럼 비교 대상이 되는 클래스에 따라 효과적인 디스크립터는 달라질 수 있다. 즉, 페르시안 고양이와의 구분에 유용한 표현이 랙돌 고양이와의 구분에도 반드시 동일하게 유효한 것은 아니다. 그럼에도 불구하고 기존 방법들은 모든 이미지에 사전에 생성된 동일한 디스크립터 집합을 적용함으로써, 각 이미지가 실제로 필요로 하는 구별 정보를 충분히 제공하지 못한다. 한편, Ma *et al.*(2024)이 제안한 DisCLIP은 클래스 간 변별력이 높은 디스크립터를 선별하는 방법을 제안하였으나, 디스크립터를 선별하기 위해 클래스별로 다수의 레이블 데이터를 필요로 한다는 점에서, 레이블 정보가 없는 실제 제로샷 환경에서의 적용하기에는 한계가 있다.

이러한 한계점을 해결하기 위해, 본 연구는 레이블 데이터 없이도 분류에 도움이 되는 디스크립터만을 선별하는 기법인 CoDES(confusion-aware discriminative descriptor selection)를 제안한다. CoDES는 테스트 시점에서 각 입력 이미지에 대해 혼동 가능한 클래스들로 구성된 혼동 집합(confusion set)을 동적으로 구성하고, 혼동 클래스 간 모호성을 해소하는 데 유효한 디스크립터만을 선별적으로 활용한다. 이를 통해 기존 방법의 한계를 극복하고 제로샷 분류 성능을 효과적으로 향상시킨다. 제안방법론의 효과를 검증하기 위해, 본 연구에서는 8개의 벤치마크 데이터셋과 3가지 크기의 CLIP 백본을 활용하여 실험을 수행하였으며, 그 결과 기존 방법론 대비 일관된 성능 향상을 달성하였다. 본 연구의 주요 기여는 다음과 같다.

1. 본 연구는 각 테스트 이미지에 대해 모델이 혼동하는 클래스들을 파악하고, 해당 혼동 클래스 간 구별에 유효한 디스크립터만을 선별하는 이미지 단위(instance-aware) 텍스트 디스크립터 선별 방법론을 제안한다. 기존 방법들이 입력 이미지와 무관하게 사전에 생성된 모든 디스크립터

들을 동일하게 적용하는 것과 달리, 제안 방법론은 각 이미지 분류에 실질적으로 유효한 디스크립터만을 활용함으로써 제로샷 분류 성능을 효과적으로 향상시킨다.

2. 본 연구는 디스크립터 선별 과정에서 레이블 데이터를 전혀 활용하지 않는 선별 메커니즘을 제안한다. 디스크립터 선별 시 클래스별 다수의 레이블 데이터를 필요로 하는 기존 선별 방식과 달리, 제안 방법론은 현재 입력된 테스트 이미지와의 유사도만을 활용한다. 구체적으로, 동일한 속성을 표현하는 디스크립터들 간의 의미적 경쟁을 통해 클래스를 미세하게 구별하는 핵심 단서를 추출함으로써, 별도의 레이블 정보 없이도 혼동 클래스 간 구별에 실질적으로 기여하는 디스크립터를 효과적으로 식별할 수 있다.

## 2. 관련 연구

### 2.1 VLM 기반 제로샷 분류

VLM은 대규모 이미지-텍스트 쌍을 활용하여 시각적 표현과 언어적 표현을 공통된 임베딩 공간에 정렬하는 방식으로 학습된다. 이 중 CLIP은 인터넷에서 수집된 수억 개의 이미지-텍스트 쌍을 대조 학습 방식으로 학습하여, 이미지 인코더와 텍스트 인코더를 동시에 학습시킨 대표적인 VLM이다(Radford *et al.*, 2021). CLIP의 제로샷 분류는 각 클래스 이름을 “A photo of a {class}”와 같은 텍스트 프롬프트로 변환하여 텍스트 임베딩을 생성하고, 입력 이미지의 임베딩과의 코사인 유사도를 계산하여 가장 유사한 클래스를 예측하는 방식으로 동작한다. 이러한 분류 방식은 별도의 추가 학습 없이도 새로운 데이터셋이나 클래스에 대한 분류 수행이 가능하다는 점에서 높은 실용성을 지닌다. 그러나 클래스 이름만으로 구성된 단순한 텍스트 프롬프트는 해당 클래스의 시각적 특성을 충분히 표현하지 못한다는 한계가 있다. 이를 보완하기 위해 CLIP 원논문에서는 80개의 수작업 텍스트 템플릿을 앙상블하여 활용하는 방법을 제안하였으나, 이는 데이터셋마다 별도의 수작업이 필요하다는 점에서 확장성에 제약이 있다(Radford *et al.*, 2021).

### 2.2 LLM 생성 디스크립터 활용 기법

최근에는 LLM에 내재된 풍부한 시각적 지식을 CLIP의 텍스트 프롬프트에 활용하기 위한 다양한 방법들이 제안되어 왔다. Menon and Vondrick(2022) 은 LLM에게 각 클래스의 시각적 특징을 묻는 방식으로 디스크립터를 생성하고, 이를 CLIP의 텍스트 프롬프트에 통합하여 분류를 수행하는 DCLIP을 제안하였다. Pratt *et al.*(2023)은 짧은 디스크립터 대신 클래스를 설명하는 완전한 문장형태의 디스크립터들을 LLM으로부터 다수 생성하여 프롬프트로 활용하는 기법인 CuPL을 제안하였으며, 두 방법론 모두 클래스 이름만을 사용하는 베이스라인 대비 다수의 제로샷 분류 벤치마크에서 성능 향상을 달성하였다.

한편, Roth *et al.*(2023)은 LLM이 생성한 디스크립터를 무작위 단어나 문자열로 대체하더라도 유사한 성능 향상이 관찰됨을 실험적으로 보이며, 이러한 성능 향상이 디스크립터들의 풍부한 의미론적 내용에서 기인하기보다는 단순한 앙상블 효과에 기인할 수 있음을 지적하였다. 이러한 한계를 극복하기 위해 두 가지 방향의 연구가 제안되었다. 첫째, Khattak *et al.*(2024)은 LLM이 생성한 텍스트 디스크립터만을 지도 신호로 활용하여 프롬프트 벡터를 학습하는 ProText를 제안하였다. 이를 통해 이미지 레이블 데이터 없이도 분류 성능 향상이 가능함을 보였으나, 추론 전 별도의 오프라인 학습 단계가 요구된다는 한계가 있다. 둘째, Ma *et al.*(2024) 이 제안한 DisCLIP은 변별력 있는 디스크립터를 선별하여 사용하는 기법이다. DisCLIP은 무작위 디스크립터 대신 의미론적으로 선별된 소수의 디스크립터를 사용할 때 분류 성능이 향상됨을 보임으로써, 디스크립터 자체의 의미론적 가치를 입증하였다. 다만, DisCLIP은 디스크립터의 변별력을 정량화하기 위해 모든 클래스에 대하여 레이블 데이터로부터 클래스별 유사도 행렬을 사전 구축해야 하며, 그 결과 선별 기준이 입력 이미지의 시각적 특성과 무관하게 고정되어 완전한 제로샷 환경에서의 적용에는 제약이 따른다.

이에 반해 본 연구에서 제안하는 CoDES는 레이블 데이터를 전혀 활용하지 않으면서도 이미지 별로 적응적인 디스크립터 선별을 수행한다. CoDES는 사전 구축된 유사도 행렬 대신, 현재 입력된 테스트 이미지와 디스크립터 간 유사도만을 직접 활용한다. 이때 단순히 이미지와의 유사도가 높은 디스크립터를 선별하는 것이 아니라, 동일한 속성을 다루는 디스크립터들 간의 비교를 통해 혼동 클래스를 미세하게 구별할 수 있는 핵심 단서가 되는 디스크립터를 선별한다. 이를 통해 CoDES는 레이블 데이터 없이도 현재 입력된 이미지의 시각적 특성을 반영한 정밀한 디스크립터 선별이 가능하다.

### 2.3 프롬프트 튜닝 적응 기법

프롬프트 튜닝(prompt tuning)은 VLM의 파라미터는 고정한 채로 입력 프롬프트에 학습 가능한 벡터를 추가하여 특정 태스크에 적응시키는 방법이다. CoOp, CoCoOp는 소수의 레이블 데이터를 활용하여 텍스트 프롬프트를 최적화함으로써 분류 성능을 향상시켰다(Zhou *et al.*, 2022b, 2022a). 그러나 이러한 방법들은 태스크별 레이블 데이터가 필요하다는 점에서 제로샷 환경에서의 적용에 어려움이 있다.

레이블 데이터 없이 테스트 시점에 프롬프트를 최적화하는 테스트 시점 프롬프트 튜닝(test-time prompt tuning) 방법들도 제안되었다. TPT는 단일 테스트 이미지에 대해 무작위로 생성한 증강 이미지들의 예측 분포 엔트로피를 최소화하는 방식으로 프롬프트를 최적화한다(Shu *et al.*, 2022). 이후 DiffTPT는 확산 모델(diffusion model) 기반의 증강 기법을 도입하여 TPT

의 증강 다양성을 개선하였으며(Feng *et al.*, 2023), C-TPT는 텍스트 임베딩 분포에 대한 보정 전략을 적용하여 최적화의 안정성을 높였다(Yoon *et al.*, 2024). 그러나 이러한 방법들은 테스트 시점에 반복적인 역전파(backpropagation) 연산이 필요하여 추가적인 계산 비용이 발생한다는 한계가 있다. 이에 반해, 본 연구에서 제안하는 CoDES는 어떠한 추가 학습이나 역전파 없이 텍스트 공간에서의 디스크립터 선별만으로 동작하는 완전한 학습 불필요(training-free) 방법이다.

## 3. 제안 방법론

### 3.1 문제 정의

본 연구는 CLIP을 활용한 제로샷 이미지 분류 문제를 다룬다. $N$개의 클래스 집합 $\{c_1, \ldots, c_N\}$과 각 클래스에 대해 LLM이 사전 생성한 디스크립터 풀 $\{D_1, \ldots, D_N\}$이 주어졌을 때, 테스트 이미지 $I$에 대해 올바른 클래스를 예측하는 것이 목표이다. 이때 추가적인 학습 데이터나 레이블 정보는 사용하지 않는 완전한 제로샷 설정을 가정한다.

CLIP의 표준 제로샷 분류는 이미지 인코더 $\phi_v$와 텍스트 인코더 $\phi_t$를 활용하여, 이미지 임베딩 $\phi_t(a\ photo\ of\ a\ \{c_n\})$과 각 클래스의 텍스트 프롬프트 임베딩 간의 코사인 유사도를 계산한 뒤, 유사도가 가장 높은 클래스를 예측 결과로 선택한다. LLM 기반 디스크립터를 활용하는 기존 방법(Menon and Vondrick, 2022; Pratt *et al.*, 2023)들은 각 클래스 별 디스크립터 풀 $D_c$ 내 모든 디스크립터 $d_c^{(j)}$에 대한 유사도를 평균하여 클래스 점수를 다음과 같이 계산한다:

$$s_c = \frac{1}{|D_c|} \sum_{j=1}^{D_c} \cos(\phi_v(I), \phi_t(d_c^{(j)})) \quad (1)$$

그러나 이러한 방식은 입력 이미지의 특성과 무관하게 사전에 생성된 모든 텍스트 디스크립터를 일괄적으로 활용한다는 한계를 지닌다. 그 결과, 분류에 실질적으로 기여하지 못하는 디스크립터가 포함될 경우, 유효한 디스크립터의 영향력이 희석되어 성능 저하로 이어질 수 있다. 본 연구에서는 이를 해결하기 위해 각 테스트 이미지에 특화된 혼동 집합을 구성하고, 해당 혼동을 해소하는 데 효과적인 디스크립터만을 선별하여 활용하는 CoDES를 제안한다. <Figure 2>은 CoDES의 전체 파이프라인을 나타낸다.

### 3.2 혼동 클래스 집합 구성

CoDES의 첫 번째 단계는 입력 이미지에 대해 모델이 혼동할 가능성이 높은 클래스들로 구성된 혼동 클래스집합 $C$를 정의하는 것이다. 전체 클래스를 대상으로 디스크립터를 선별

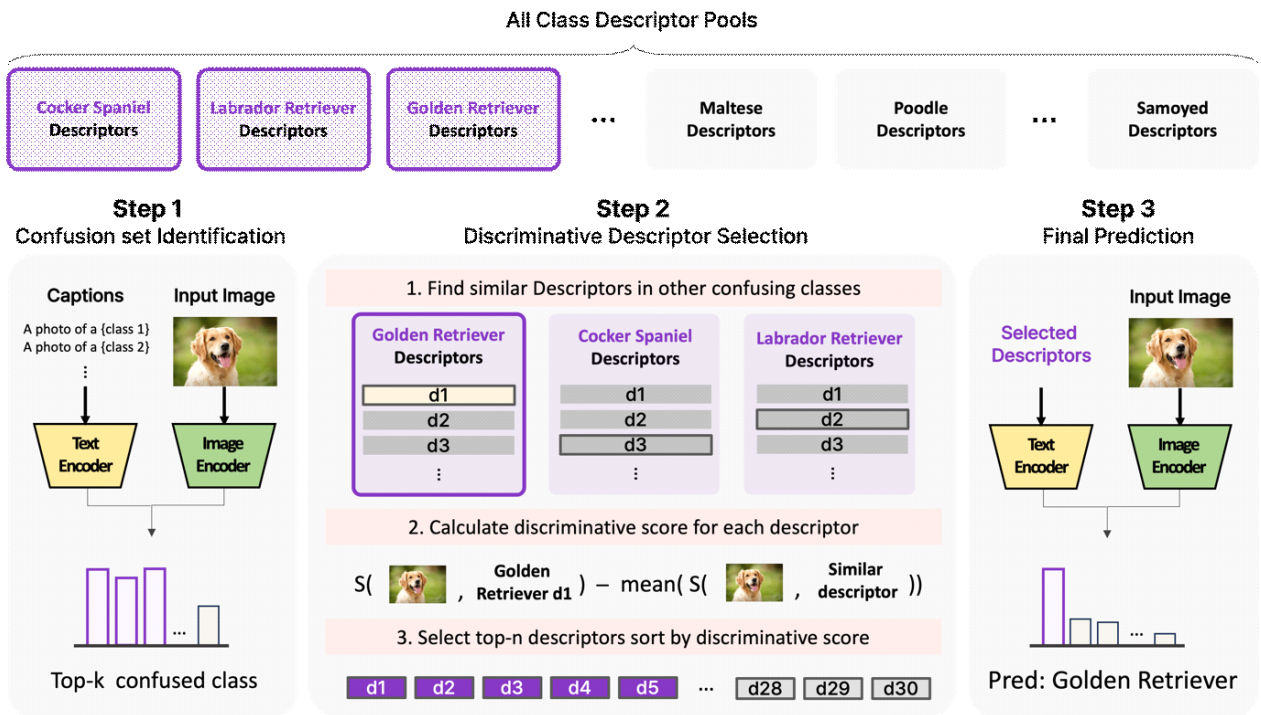



**Figure 2.** Overview of the proposed CoDES framework. Given a test image, we (1) identify confusing classes based on CLIP similarity, (2) select discriminative descriptors through semantic competition among confusing classes, and (3) make final prediction using only selected descriptors.

하는 것은 계산 비용이 클 뿐만 아니라, 실제 혼동과 무관한 클래스의 디스크립터들까지 고려하게 되어 오히려 분류 성능을 저하시킬 수 있다. 따라서 본 연구에서는 각 이미지에 대해 실제로 모델이 혼동하는 클래스들만을 선별하여 이후 단계의 탐색 범위를 좁히는 전략을 채택한다.

구체적으로, 입력 이미지 $I$의 임베딩을 $f=\phi_v(I)$, 각 클래스 $c_n$에 대한 프롬프트 임베딩을 $t_n=\phi_t(a\ photo\ of\ a\ \{c_n\})$라 할 때, 두 임베딩 간의 유사도는 다음과 같이 계산된다.

$$s_n = f \cdot t_n, \quad \forall n \in \{1,\ ...,\ N\} \tag{2}$$

이후 유사도 점수 $s_n$이 높은 상위 $k$개의 클래스를 선택하여 혼동 클래스 집합으로 정의한다.

$$C = Top-k\ indices\ of\ \ s_n \tag{3}$$

이때 선택된 상위 $k$개의 클래스는, CLIP이 해당 이미지를 분류할 때 모델이 실제로 혼동하는 후보 클래스들에 해당한다. 이처럼 비교 대상을 실제 혼동이 발생하는 소수의 클래스로 좁힘으로써, 이후 디스크립터 선별 과정에서 실제로 모델이 혼동하는 클래스들 사이에서 이를 구분할 수 있는 핵심 단서가 되는 디스크립터를 정밀하게 선별할 수 있게 된다.

### 3.3 변별적 텍스트 디스크립터 선별

두 번째 단계는 혼동 클래스 집합 $C$에 포함된 각 클래스에 대해, 혼동을 해소하는 데 실질적으로 유효한 텍스트 디스크립터만을 선별하는 과정이다. 본 단계의 핵심은 특정 클래스의 디스크립터가 대상 클래스와 혼동 클래스들 사이에서 얼마나 높은 변별력을 제공하는지를, 혼동 집합 내 다른 클래스들과의 비교를 통해 정량화하는 데 있다.

혼동 집합 내 각 클래스 $c \in C$를 대상 클래스(target class) 라 하고, $C$에서 $c$를 제외한 나머지 클래스 $c' \in C \setminus \{c\}$를 비교 클래스(rival class)라 한다. 대상 클래스 $c$의 디스크립터 풀 $D_c$에 포함된 각 디스크립터 $d_c^{(j)}$의 변별력 점수 $r^{(j)}$를 다음과 같이 계산한다.

먼저 대상 클래스의 각 디스크립터 $d_c^{(j)}$에 대해, 각 비교 클래스 $c'$의 디스크립터 풀 $D_{c'}$내에서 의미적으로 가장 유사한 디스크립터(이하 경쟁 디스크립터)의 인덱스를 찾는다. 이때 비교 클래스 $c'$의 $j'$번째 디스크립터를 $d_c^{(j')}$로 표기하며, 이 중 대상 디스크립터 $d_c^{(j)}$와 가장 유사한 디스크립터의 인덱스를 $j^*$로 정의한다. 이는 동일하거나 유사한 주제를 다루는 비교 클래스의 디스크립터를 찾아 비교 범위를 좁히는 과정이다:

$$j^* = argmax_{j'}(e_c^{(j)}) \cdot e_{c'}^{(j')}) \quad \forall c' \in C \setminus \{c\} \tag{4}$$

이후 이미지 임베딩 $f$와 대상 클래스 디스크립터 임베딩 $e_c^{(j)}$ 간의 유사도에서, 각 비교 클래스 에 대해 식 (4)로 선택된 경쟁 디스크립터들의 이미지 유사도 평균을 뺀 값을 변별력 점수로 다음과 같이 정의한다:

$$r^{(j)} = f \cdot e_c^{(j)} - mean_{c'}(f \cdot e_{c'}^{(j^*)}) \tag{5}$$

여기서 변별력 점수 $r^{(j)}$가 높다는 것은 해당 디스크립터가 유사한 주제를 다루는 혼동 클래스의 디스크립터와 비교했을 때, 이미지를 대상 클래스로 더 효과적으로 구별할 수 있음을 의미한다. 반대로 점수가 낮은 디스크립터는 대상 클래스와 혼동 클래스를 구별하는 데 기여하지 못하므로 제거 대상이 된다. 최종적으로 다음 식과 같이 변별력 점수 상위 $m$개의 디스크립터를 선택한다.

$$S_c = Top-m\quad indices\quad of\quad r^{(j)} \tag{6}$$

이처럼 혼동 클래스 집합 내 다른 클래스와의 비교를 통해 각 디스크립터의 변별력을 평가함으로써, 단순히 이미지와의 유사도가 높은 디스크립터가 아니라 실제로 혼동을 줄이는데 기여하는 디스크립터만을 선별할 수 있다.

### 3.4 선별된 디스크립터 기반 최종 예측

마지막 단계는 앞서 선별된 디스크립터만을 활용하여 최종 클래스를 예측하는 것이다. 혼동 클래스 집합 내 각 클래스 $c \in C$에 대해, 선별된 디스크립터 집합 $S_c$에 속하는 텍스트 디스크립터들과 이미지 임베딩 간 유사도 평균을 클래스 점수로 다음과 같이 계산한다:

$$\sigma_c = mean_{\ j \in S_c}(f \cdot e_c^{(j)}) \tag{7}$$

기존 방법들이 모든 디스크립터를 사용하는 것과 달리, 제안 방법은 혼동 해소에 유효한 디스크립터들의 사용함으로써 변별력 있는 시각적 특징에 집중할 수 있다. 최종 예측 클래스 $\hat{y}$는 클래스 점수가 가장 높은 클래스로 다음과 같이 결정된다:

$$\hat{y} = argmax_{\ c \in C}\ \sigma_c \tag{8}$$

제안 방법론의 전체 과정은 <Algorithm 1>에 요약되어 있다. CoDES는 이미지별로 혼동 클래스 집합을 동적으로 구성하고, 클래스 간 혼동 해소에 도움이 될 수 있는 디스크립터만을 선별하여 예측에 활용함으로써, 제로샷 분류 성능을 효과적으로 향상시킨다.

Algorithm 1: CoDES (Confusion-aware Discriminative dEscriptor Selection)

Require: Test set $T$
Pretrained CLIP encoders $\phi_v$, $\phi_t$
Class set $\{c_1, ..., c_N\}$
Per-class descriptor pool $\{D_1, ..., D_N\}$
$k$: number of confusion set classes
$m$: number of selected descriptors per class

for each test image $I \in$ T do

Step 1: Confusion set construction
$f \leftarrow \phi_v(I)$ // Encode image feature
$t_n \leftarrow \phi_t$ ("A photo of a $\{c_n\}$.) for all class // Encode class captions
$s_n \leftarrow f \cdot t_n$ for all class $n$ // Image-to-class similarities
$C \leftarrow$ Top-k indices of $s_n$ // Form confusion set

Step 2: Per-class discriminative descriptor selection
$e_c^{(j)} \leftarrow \phi_t(d_c^{(j)})$ for all // Encode all descriptors
for each $c \in C$ do
for each descriptor $d_c^{(j)} \in D_c$ do
for each $c' \in C\{c\}$ do
$j^* \leftarrow argmax_j (e_c^{(j)} \cdot e_{c'}^{(j)})$ // Find rival descriptors in other confusing classes
end for
$r^{(j)} \leftarrow f \cdot e_c^{(j)} - mean_{c'}(f \cdot e_{c'}^{(j^*)})$ // Compute discriminative score
end for
$S_c \leftarrow$ Top-$m$ indices of $r^{(j)}$ // Select descriptors
end for

Step 3: Prediction with selected descriptors
$\sigma_c \leftarrow mean_{j \in S_c}(f \cdot e_c^{(j)})$ for all $c \in C$
$\hat{y} \leftarrow argmax_{c \in C}\ \sigma_c$ // Final prediction
end for

Return: predicted labels $\hat{y}$ for all $I \in T$

## 4. 실험 및 결과

### 4.1 실험셋팅

본 연구는 8개의 이미지 분류 벤치마크 데이터셋을 활용하여 제안 방법의 성능을 평가하였다. 사용한 데이터셋은 Flowers102(Nilsback and Zisserman, 2008), Oxford Pets(Parkhi *et al.*, 2012.), DTD(Cimpoi *et al.*, 2013), Food101(Bossard *et al.*, 2014), Sun397(Xiao *et al.*, 2010.), UCF101(Soomro *et al.*, 2012), Caltech101(Fei-Fei *et al.*, 2004.), Stanford Cars(Krause *et al.*, 2013)로, 꽃, 동물, 질감, 음식, 장면, 행동, 사물, 자동차 등 다양한 도메인을 아우르는 표준 벤치마크 데이터셋이다.

본 연구에서는 CLIP ViT-B/16을 기본 백본 모델로 사용하였으며, 기존 연구의 평가 방식을 따라 평가 지표로는 top-1 분류 정확도를 적용하였다. 또한 혼동 집합 크기 k와 선별 디스크립터 수 m은 {5, 10, 15, 20} 범위에서 탐색하였으며, 각 데이터셋에서 가장 우수한 성능을 보인 값을 최종 설정하여 사용하였다. 데이터셋별 구체적인 하이퍼파라미터 설정값은 <Table 1>에 제시하였으며, 하이퍼파라미터 변화에 대한 성능 민감도 분석 결과는 4.4에서 상세히 기술한다.

**Table 1.** Hyperparameter Settings for Each Benchmark Datasets

| | flowers | pets | dtd | food | sun | ucf101 | caltech | cars |
|---|---|---|---|---|---|---|---|---|
| # Classes | 102 | 37 | 47 | 101 | 397 | 101 | 100 | 196 |
| $k$ | 20 | 20 | 20 | 5 | 20 | 20 | 5 | 10 |
| $m$ | 5 | 20 | 10 | 15 | 20 | 15 | 15 | 20 |

본 연구는 텍스트 디스크립터 생성 자체보다 주어진 디스크립터 풀에서 각 테스트 이미지별 유효한 디스크립터를 선별하는 데 초점을 두므로, CuPL에서 공개한 LLM 생성 디스크립터 풀을 추가 조작 없이 그대로 활용하였다. CuPL은 "Describe what a/the {class} looks like", "What are the identifying characteristics of a/the {class}?" 등의 프롬프트 템플릿을 GPT-3에 입력하여 각 클래스의 시각적 특징을 설명하는 디스크립터를 생성한다. 이렇게 구축된 CuPL의 디스크립터 풀은 클래스당 약 30개의 디스크립터로 구성되어 있다.

### 4.2 실험 결과

제안 방법론인 CoDES의 제로샷 이미지 분류 성능을 검증하기 위해, 본 연구에서는 표준 제로샷 분류 방법인 CLIP, 테스트 시점 프롬프트 튜닝 방법인 TPT, DiffTPT, C-TPT, 그리고 LLM 기반 디스크립터 생성 방법인 CuPL과의 성능 비교 실험을 수행하였다. <Table 2>는 이러한 비교 방법론들과의 실험 결과를 나타낸 것이다. 제안 방법론 CoDES는 8개의 벤치마크 데이터셋 평균 기준으로 모든 비교 방법론 대비 가장 높은 분류 정확도를 달성하였다. 또한, 8개 데이터셋의 top-1 분류 정확도를 대상으로 Friedman 검정을 수행한 결과, 분류 정확도 차이가 통계적으로 유의한 것을 확인하였다($p-value < 0.001$).

LLM 기반 디스크립터를 이용하는 CuPL, ProText, 그리고 CoDES는 전반적으로 테스트 시점 프롬프트 튜닝 방법들보다 우수한 성능을 보였다. 이 가운데 제안한 CoDES는 동일한 디스크립터 풀을 사용하는 CuPL에 디스크립터 선별 과정을 추가로 적용함으로써 평균 1.07%p의 성능 향상을 달성하였다. CoDES는 8개 데이터셋 중 7개에서 성능 향상을 보였으며, 특히 질감 분류(DTD)와 자동차 분류(Stanford Cars)에서 가장 뚜렷한 개선을 달성하였다. 이러한 결과는 테스트 시점 적응을 위해 반드시 역전파 기반 최적화가 필요한 것은 아니며, 주어진 디스크립터 풀 내에서 이미지별로 유효한 디스크립터를 선별하는 것만으로도 제로샷 분류 성능을 효과적으로 향상시킬 수 있음을 시사한다.

또한 CLIP 백본 규모에 따른 제안 방법의 성능 변화를 분석하기 위해, 가장 작은 모델인 ViT-B/32, 중간 규모인 ViT-B/16,

**Table 2.** Performance comparison of zero-shot image classification top-1 accuracy (%) on 8 benchmark datasets using ViT-B/16 backbone. The best performance among compared methods is highlighted in bold.

| Method | flowers | pets | dtd | food | sun | ucf101 | caltech | cars | Mean |
|---|---|---|---|---|---|---|---|---|---|
| CLIP | 67.07 | 88.14 | 44.27 | 83.65 | 61.78 | 64.92 | 92.94 | 63.45 | 70.78 |
| Test-time prompt tuning methods | | | | | | | | | |
| TPT | 68.98 | 87.79 | 47.75 | 84.67 | 65.50 | 68.04 | 94.16 | 66.87 | 72.97 |
| DiffTPT | 70.10 | 88.22 | 47.00 | 87.23 | 65.74 | 68.22 | 92.49 | **67.01** | 73.25 |
| C-TPT | 69.80 | 88.20 | 46.00 | 83.70 | 64.80 | 65.70 | 93.60 | 65.80 | 72.20 |
| Descriptor-based training-free methods | | | | | | | | | |
| CuPL | 71.98 | 91.93 | 47.66 | 89.22 | 65.69 | **71.35** | 93.18 | 64.42 | 74.43 |
| ProText | 72.35 | 91.01 | 47.93 | 86.66 | **67.34** | 69.60 | **94.81** | 66.00 | 74.46 |
| CoDES (Proposed) | **72.99** | **92.04** | **51.01** | **89.43** | 66.62 | 71.21 | 94.20 | 66.47 | **75.50** |

그리고 가장 큰 모델인 ViT-L/14를 사용하여 성능을 비교 하였다. 아울러 동일한 설정에서 CuPL과의 비교 실험도 함께 수행하였다. <Table 3>은 백본 크기를 달리했을 때 CoDES와 CuPL의 8개 데이터셋 평균 분류 정확도를 비교한 결과를 보여준다.

**Table 3.** Average top-1 accuracy (%) of CuPL and CoDES across 8 benchmark datasets for various CLIP backbone scales.

| Method | ViT-B/32 (small) | ViT-B/16 (medium) | ViT-L/14 (large) |
|---|---|---|---|
| CuPL | 70.89 | 74.43 | 80.04 |
| CoDES (Proposed) | 71.85 | 75.50 | 80.54 |

예상대로 CoDES와 CuPL 모두 백본 모델의 크기가 커질수록 성능이 향상되는 경향을 보였다. 아울러 CoDES는 세 가지 백본 모두에서 CuPL보다 높은 성능 향상을 기록하였다. 이는 제안한 혼동 클래스 기반 디스크립터 선별 전략이 특정 백본에 한정되지 않고, 다양한 규모의 CLIP 아키텍처 전반에서 일관되게 효과적임을 보여준다. 다만 백본 규모가 커질수록 개선 폭이 점차 감소하는 경향이 나타나는데, 이는 대규모 백본일수록 이미지 인코더의 표현력이 향상되어 기본적인 분류 성능 자체가 높아지기 때문에, 혼동 해소를 통해 추가로 얻을 수 있는 성능 향상 여지가 상대적으로 줄어들기 때문으로 해석할 수 있다.

추가로, 레이블된 이미지를 활용하여 디스크립터를 선별하는 선행연구인 DisCLIP과 비교함으로써, 어떠한 레이블 정보도 사용하지 않는 제안 방법인 CoDES의 성능이 어느 수준까지 도달할 수 있는지를 분석하였다. DisCLIP은 디스크립터 선별을 위해 클래스별로 다수의 레이블된 이미지와 각 디스크립터 간 유사도를 사전에 계산한 행렬을 활용한다. 반면 CoDES는 입력된 테스트 이미지 자체와의 유사도를 직접 이용하므로, 별도의 레이블 정보 없이 순수한 제로샷 설정에서 동작한다. <Table 4>는 8개 데이터 셋에 대한 CoDES와 DisCLIP의 성능 비교 결과를 보여주고 있다.

평균 성능 기준으로 CoDES는 75.50%로 DisCLIP(75.38%) 대비 0.12%p 높은 결과를 달성하였다. 일부 데이터셋에서는 DisCLIP이 우세하지만, 전반적으로는 두 방법론은 유사한 수준의 성능을 나타냈다 이러한 결과는 CoDES가 클래스별 다수의 레이블 데이터를 필요로 하는 DisCLIP과 달리, 어떠한 레이블 데이터도 사용하지 않으면서도 대등한 수준의 성능을 달성했음을 보여준다. 즉, 별도의 레이블 데이터 수집 없이도 레이블 기반 디스크립터 선별 방법에 준하는 수준의 효과적인 선별이 가능함을 시사한다.

제안방법론인 CoDES의 디스크립터 선별 효과를 직관적으로 확인하기 위해 정성적 분석을 수행하였다. <Figure 3>은 Electric Guitar 이미지에 대해 CoDES가 선별한 디스크립터를 보여주며, 하늘색으로 표시된 행이 해당 결과에 해당한다. 이 사례의 정답 클래스는 Electric Guitar이지만, CLIP과 CuPL은 모두 이를 Mandolin으로 오분류하였다. 두 악기는 줄이 있으며 손으로 연주하는 현악기라는 공통적인 외형적 특징을 지니고 있어, 육안으로도 구분이 쉽지 않은 혼동 사례에 해당한다. 반면 CoDES는 두 악기를 구별하는 핵심적인 시각적 차이점을 정확히 선별함으로써, 해당 이미지를 Electric Guitar로 올바르

**Table 4.** Performance comparison of zero-shot image classification top-1 accuracy (%) with DisCLIP, a descriptor selection method requiring labeled data, on 8 benchmark datasets using ViT-B/16 backbone. The best performance among compared methods is highlighted in bold.

| Method | flowers | pets | dtd | food | sun | ucf101 | caltech | cars | Mean |
|---|---|---|---|---|---|---|---|---|---|
| DisCLIP | 74.79 | 92.20 | 54.89 | 89.21 | 64.20 | 70.34 | 92.94 | 64.46 | 75.38 |
| CoDES (Proposed) | 72.99 | 92.04 | 51.01 | 89.43 | 66.62 | 71.21 | 94.20 | 66.47 | 75.50 |

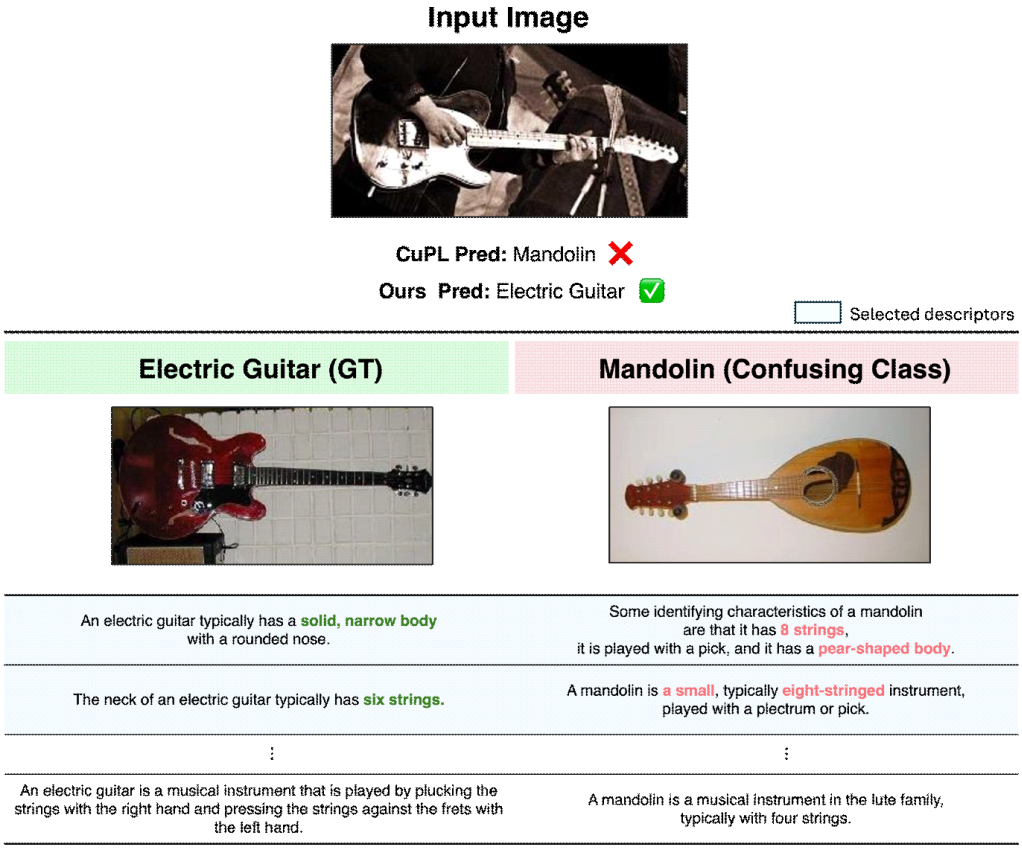



**Figure 3.** Qualitative analysis of descriptor selection by CoDES for an Electric Guitar image. Highlighted rows indicate selected descriptors. Although both CLIP and CuPL misclassify the image as Mandolin, CoDES correctly predicts Electric Guitar by selecting descriptors that are visually discriminative between the confusing classes.

게 분류하였다.

CoDES가 선별한 디스크립터를 살펴보면, 두 악기의 핵심적인 시각적 차이인 바디 형태와 현의 개수가 주요 단서로 활용되었음을 확인할 수 있다. Electric Guitar 측에서는 “solid, narrow body with a rounded nose”와 “six strings”와 같이 Mandolin과 시각적으로 명확히 구별되는 형태적 특징이 선별되었으며, Mandolin 측에서도 “pear-shaped body”와 “eight strings”와 같이 Electric Guitar와 명확히 구별되는 특징이 선택되었다. 이는 CoDES가 단순히 이미지와의 유사도가 높은 디스크립터를 선별하는 것이 아니라, 혼동 클래스의 디스크립터 중 의미적으로 가장 유사한 경쟁 디스크립터를 먼저 찾음으로써 비교 범위를 같은 주제로 좁히고, 그 안에서 변별력 차이가 가장 큰 디스크립터를 선별하기 때문이다. 예를 들어 “six strings”는 Mandolin의 “eight strings”와 동일한 ‘현의 개수’라는 속성에서 직접적으로 비교됨으로써, 두 클래스를 효과적으로 구별할 수 있는 핵심 단서로 작용한다. 반면 Electric Guitar 측의 “played by plucking the strings with the right hand”는 현을 손가락으로 뜯어 연주하는 동작을 설명하는 문장으로, 정적인 이미지에서는 시각적으로 확인할 수 없는 특징이기 때문에 이미지 임베딩과의 유사도가 낮아 선별에서 제외되었다. Mandolin 측의 “musical instrument in the lute family” 역시 Electric Guitar도 류트 계열 현악기에 해당하므로 두 클래스 모두에 공통적으로 적용되는 설명으로 변별력이 낮아 제외되었다. 이와 같이 CoDES는 혼동 클래스 간 구별에 실질적으로 기여하는 디스크립터만을 선별함으로써, 기존 방법들이 오분류한 사례를 올바르게 분류할 수 있음을 보여준다.

### 4.3 구성 요소별 기여도 분석

제안 방법론의 핵심 구성 요소인 혼동 클래스 후보 선정과 경쟁 디스크립터 매칭 기반 선별의 개별 기여도를 검증하기 위해 구성 요소별 기여도 분석을 수행하였다. <Table 5>는 각 구성 요소를 제거하였을 때의 8개 데이터셋 평균 정확도 변화를 나타낸 것이다.

혼동 클래스 후보 선정을 제거하고 전체 $N$개 클래스를 대상으로 경쟁 디스크립터 매칭을 수행할 경우 (w/o confusion set), CuPL 대비 +0.59%p의 개선이 관찰되지만 CoDES 대비 0.48%p 낮은 성능을 보였다. 이는 분류와 무관한 클래스의 디스크립터까지 경쟁 대상으로 포함될 경우 변별력 평가의 정밀도가 낮아짐을 의미하며, 탐색 범위를 실제 혼동 가능성이 높

은 클래스들로 제한하는 것이 경쟁 매칭의 효과를 극대화하는 데 중요함을 확인할 수 있었다.

**Table 5.** Ablation study evaluating the contribution of each component to the overall performance of CoDES (average top-1 accuracy, %).

| Method | acc | diff |
|---|---|---|
| CoDES(Proposed) | 75.50 | - |
| w/o confusion set | 75.02 | -0.48 |
| w/o descriptor matching | 74.91 | -0.59 |
| w/o selection(=CuPL) | 74.43 | -1.07 |

경쟁 디스크립터 매칭을 제거하고 이미지-디스크립터 유사도만으로 상위 m개 디스크립터를 선별할 경우 (w/o descriptor matching), CuPL 대비 +0.48%p 개선되어 혼동 클래스로 후보를 좁히는 것 자체의 효과를 확인할 수 있었다. 그러나 이 설정은 CoDES 대비 0.59%p 낮은 성능을 보였는데, 이는 단순히 이미지와의 유사도가 높은 디스크립터를 선별하는 것이 아니라 동일한 속성을 다루는 디스크립터들 간의 정교한 비교를 통해 혼동 클래스를 미세하게 구별하는 핵심 단서를 추출하는 과정이 CoDES 성능 향상의 핵심 요소임을 보여주었다.

두 구성 요소의 관계를 살펴보면, 혼동 클래스 후보 선정은 경쟁 디스크립터 매칭이 의미 있게 작동하기 위한 전제 조건이라는 점에서 그 필요성을 이해할 수 있다. 경쟁 디스크립터 매칭은 다른 클래스의 디스크립터 중 의미적으로 가장 유사한 경쟁 디스크립터를 찾아 변별력을 정량화하는 방식으로 작동하기 때문에, 비교 대상이 되는 클래스 집합의 구성이 변별력 평가의 의미를 좌우한다. 만약 비교 대상에 분류와 무관한 클래스까지 포함되면, 경쟁 매칭은 실제 혼동 해소와 무관한 클래스 쌍에 기반하여 변별력을 평가하게 되며, 그 결과 선별된 디스크립터가 실제로 분류에 필요한 핵심 단서를 반영하지 못하게 된다. 혼동 클래스 후보 선정은 이러한 문제를 해소함으로써, 경쟁 매칭이 실제로 분류 시 변별이 필요한 클래스 쌍에 집중하여 핵심 단서를 추출하도록 한다. 또한 비교 대상이 되는 클래스 수를 $N$에서 $k$로 축소함으로써 경쟁 매칭의 계산 비용을 줄이는 효율성 측면의 이점도 함께 확보된다.

### 4.4 하이퍼파라미터 민감도 분석

혼동 클래스 후보 개수 $k$와 선별 디스크립터 수 $m$의 변화가 분류 성능에 미치는 영향을 분석하기 위해, $k, m \in \{5, 10, 15, 20\}$의 총 16개 조합 각각에 대해 실험을 수행하였다. 각 조합별 성능은 8개 데이터셋의 평균 분류 정확도를 기준으로 평가하였으며, 그 결과를 <Table 6>에 제시하였다.

**Table 6.** Average top-1 accuracy (%) across 8 benchmark datasets for all (k, m) combinations.

| | m=5 | m=10 | m=15 | m=20 |
|---|---|---|---|---|
| $k$=5 | 74.09 | 74.48 | 74.7 | 74.64 |
| $k$=10 | 74.65 | 74.89 | 74.91 | 74.74 |
| $k$=15 | 74.80 | 75.10 | 75.18 | 75.05 |
| $k$=20 | 74.95 | 75.20 | 75.30 | 75.16 |

실험 결과, 16개 조합 전반에 걸쳐 평균 정확도는 74.09% ~ 75.30%로 1.21%p 이내에 분포하며, 표준편차는 0.3%에 불과하였다. 이는 제안 방법론이 하이퍼파라미터 $k$와 $m$의 변화에 대해 비교적 안정적인 성능을 보임을 의미한다. 또한 worst case 조합인 ($k$=5, $m$=5)를 제외한 15개 조합 모두에서 CuPL(74.43%) 대비 일관된 성능 향상이 관찰되었다.

추가로 ($k$, $m$)의 변화가 데이터셋별 성능에 미치는 영향을 시각적으로 확인하기 위해 4가지 데이터셋에 대하여 민감도 분석을 수행하였으며, 그 결과는 <Figure 4>에 제시하였다. 분석 결과, 4가지 데이터셋 모두에서 ($k$, $m$) 변화에 따라 약간의 성능 차이는 존재하나 그 변동이 유의미하게 크지 않음을

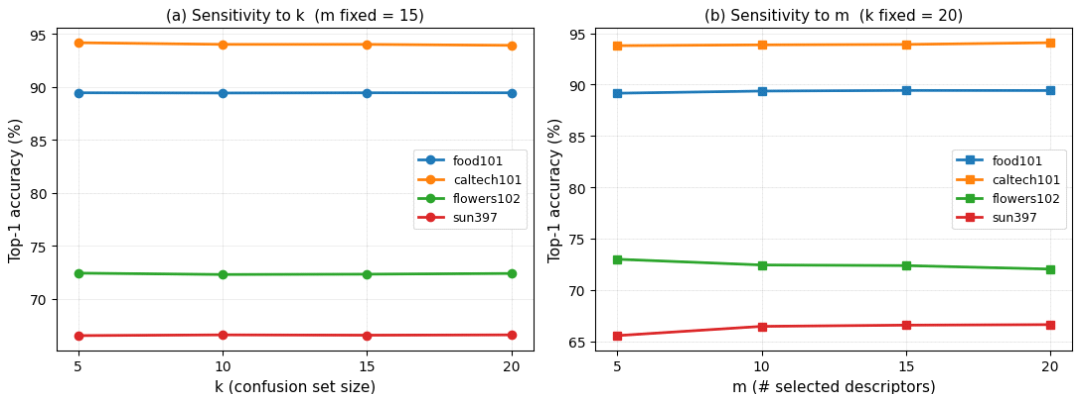



**Figure 4.** Hyperparameter sensitivity analysis of CoDES on four datasets (Food101, Caltech101, Flowers102, and SUN397). (a) Top-1 accuracy with varying confusion set size *k*, while fixing the number of selected descriptors to *m*=15. (b) Top-1 accuracy with varying number of selected descriptors m, while fixing the confusion set size to *k*=20.

확인하였다. 이러한 결과는 CoDES의 성능이 특정 하이퍼파라미터 설정에 과도하게 의존하지 않으며, 넓은 범위의 설정에서도 일관된 성능 향상을 제공함을 보여준다.

## 5. 결 론

본 연구는 제로샷 분류 문제에 있어 클래스 간 혼동을 효과적으로 해소하기 위해, 각 테스트 이미지에 특화된 혼동 집합을 동적으로 구성하고 해당 혼동을 해소하는 데 유효한 디스크립터만을 선별하는 CoDES를 제안하였다. 여덟 개의 벤치마크 데이터셋과 다양한 크기의 CLIP 백본에서 수행한 실험 결과, 동일한 디스크립터 풀을 사용하더라도 모든 디스크립터를 적용하는 것보다 이미지별로 유효한 디스크립터만을 선별하여 사용하는 것이 더 효과적임을 입증하였다. 특히 클래스별 다수의 레이블 데이터를 필요로 하는 DisCLIP에 필적하는 성능을 별도의 레이블 데이터 없이 달성함으로써, 학습 과정 없이도 효과적인 디스크립터 선별이 가능함을 보였다. 이는 디스크립터 풀을 활용하는 기존 방법론에 제안한 선별 방식을 플러그인 형태로 적용할 경우, 추가적인 성능 향상을 기대할 수 있음을 시사한다.

다만, 본 연구에서는 LLM이 각 클래스를 충분히 변별할 수 있는 수준의 디스크립터를 안정적으로 생성할 수 있는 상황을 전제로 한다. 그러나 의료 영상 진단, 법률 문서, 제조 공정 검사 등 전문 도메인에서는 클래스 간 차이가 도메인 특화 지식에 의존적이므로, 범용 LLM이 생성하는 디스크립터만으로는 충분한 변별력을 확보하기 어려울 수 있다. 이를 보완하기 위해, 향후 연구에서는 도메인 특화 지식을 학습한 전문 LLM이나 외부 지식 베이스를 활용하여 디스크립터 풀을 보강하는 방안을 고려하고자 한다. 또한 디스크립터 풀의 변별력 수준을 정량적으로 진단하여 부족한 영역을 자동으로 보완하는 기법을 함께 모색하고자 한다. 이러한 확장을 통해 CoDES는 범용 이미지 분류 벤치마크를 넘어 다양한 전문 도메인에서도 효과적으로 활용될 수 있는 실용적 프레임워크로 발전할 수 있을 것으로 기대된다.

## 참고문헌


Bossard, L.; Guillaumin, M.; and Van Gool, L. (2014), Food-101 – mining discriminative components with random forests, *In European Conference on Computer Vision (ECCV)*, 446-461.

Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., and Vedaldi, A. (2014), Describing Textures in the Wild, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3606-3613. https://doi.org/10.1109/CVPR.2014.461.

Dubey, A., Gupta, O., Guo, P., Raskar, R., Farrell, R., and Naik, N. (2018), Pairwise Confusion for Fine-Grained Visual Classification, Computer Vision – ECCV 2018, *Lecture Notes in Computer Science*, 11216, 71-88. https://doi.org/10.1007/978-3-030-01258-8_5

Fei-Fei, L., Fergus, R., and Perona, P. (2007), Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories, *Computer Vision and Image Understanding*, **106**(1), 59-70. https://doi.org/10.1016/j.cviu.2005.09.012.

Feng, C.-M., Yu, K., Liu, Y., Khan, S., and Zuo, W. (2023), Diverse Data Augmentation with Diffusions for Effective Test-time Prompt Tuning, *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2704-2714. https://doi.org/10.1109/ICCV51070.2023.00255

Khattak, M. U., Naeem, M. F., Naseer, M., Van Gool, L., and Tombari, F. (2025), Learning to Prompt with Text Only Supervision for Vision-Language Models, *Proceedings of the AAAI Conference on Artificial Intelligence*, *39*(4), 4230-4238. https://doi.org/10.1609/aaai.v39i4.32444

Krause, J., Stark, M., Deng, J., and Fei-Fei, L. (2013), 3D Object Representations for Fine-Grained Categorization, *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW)*, 554-561. https://doi.org/10.1109/ICCVW.2013.77

Ma, P., Rietdorf, L., Kotovenko, D., Hu, V. T., and Ommer, B. (2025), Does VLM Classification Benefit from LLM Description Semantics?, *Proceedings of the AAAI Conference on Artificial Intelligence*, **39**(6), 5973-5981. https://doi.org/10.1609/aaai.v39i6.32638

Menon, S. and Vondrick, C. (2023), Visual Classification via Description from Large Language Models, *Proceedings of the International Conference on Learning Representations (ICLR)*. https://arxiv.org/abs/2210.07183

Nilsback, M.-E. and Zisserman, A. (2008), Automated Flower Classification over a Large Number of Classes, *Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP)*, 722-729. https://doi.org/10.1109/ICVGIP.2008.47

Parkhi, O. M., Vedaldi, A., Zisserman, A., and Jawahar, C. V. (2012), Cats and Dogs, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3498-3505. https://doi.org/10.1109/CVPR.2012.6248092

Pratt, S., Covert, I., Liu, R., and Farhadi, A. (2023), What Does a Platypus Look Like? Generating Customized Prompts for Zero-Shot Image Classification, *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 15691-15701. https://doi.org/10.1109/ICCV51070.2023.01438

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. (2021), Learning Transferable Visual Models from Natural Language Supervision, *Proceedings of the 38th International Conference on Machine Learning (ICML)*, PMLR 139, 8748-8763. https://arxiv.org/abs/2103.00020

Roth, K., Kim, J. M., Koepke, A. S., Vinyals, O., Schmid, C., and Akata, Z. (2023), Waffling around for Performance: Visual Classification with Random Words and Broad Concepts, *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 15746-15757. https://doi.org/10.1109/ICCV51070.2023.01443

Shu, M., Nie, W., Huang, D.-A., Yu, Z., Goldstein, T., Anandkumar, A., and Xiao, C. (2022), Test-Time Prompt Tuning for Zero-Shot Generalization in Vision-Language Models, *Advances in Neural Information Processing Systems (NeurIPS)*, **35**, 14274-14289. https://arxiv.org/abs/2209.07511

Soomro, K., Zamir, A. R., and Shah, M. (2012), UCF101: A Dataset of 101 Human Actions Classes from Videos in the Wild, *Technical Report*

*CRCV-TR-12-01, Center for Research in Computer Vision*, University of Central Florida. https://arxiv.org/abs/1212.0402

Wei, X.-S., Song, Y.-Z., Mac Aodha, O., Wu, J., Peng, Y., Tang, J., Yang, J., and Belongie, S. (2022), Fine-Grained Image Analysis with Deep Learning: A Survey, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **44**(12), 8927-8948. https://doi.org/10.1109/TPAMI.2021.3126648

Xiao, J., Hays, J., Ehinger, K. A., Oliva, A., and Torralba, A. (2010), SUN Database: Large-Scale Scene Recognition from Abbey to Zoo, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3485-3492. https://doi.org/10.1109/CVPR.2010.5539970

Yoon, H. S., Yoon, E., Tee, J. T. J., Hasegawa-Johnson, M., Li, Y., and Yoo, C. D. (2024), C-TPT: Calibrated Test-Time Prompt Tuning for Vision-Language Models via Text Feature Dispersion, *Proceedings of the International Conference on Learning Representations (ICLR)*. https://arxiv.org/abs/2403.14119

Zhou, K., Yang, J., Loy, C. C., and Liu, Z. (2022a), Conditional Prompt Learning for Vision-Language Models, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 16816-16825. https://doi.org/10.1109/CVPR52688.2022.01631

Zhou, K., Yang, J., Loy, C. C., and Liu, Z. (2022b), Learning to Prompt for Vision-Language Models, *International Journal of Computer Vision*, **130**(9), 2337-2348. https://doi.org/10.1007/s11263-022-01653-1

# 부록 A. 추가 분석

## A.1 단일 고정 하이퍼파라미터 설정에서의 성능 비교

본 연구의 주 실험에서는 데이터셋별로 가장 우수한 성능을 보인 $(k, m)$ 조합을 사용하였다. 본 부록에서는 데이터셋별 최적값 대신 모든 데이터셋에 단일 고정값을 적용했을 때의 성능을 검증하기 위해, 16개 조합 중 평균 성능이 가장 높았던 ($k$=20, $m$=15)를 단일 설정으로 채택하여 추가 실험을 수행하였다. <Table A1>은 데이터셋별 최적값과 단일 고정값을 적용한 CoDES의 8개 데이터셋 분류 정확도를 비교한 결과를 보여준다.

**Table A1.** Comparison between dataset-specific best and single fixed (k=20, m=15) settings of CoDES on 8 benchmark datasets using ViT-B/16 backbone.

| | flowers | pets | dtd | food | sun | ucf | caltech | cars | avg |
|---|---|---|---|---|---|---|---|---|---|
| CuPL (baseline) | 71.98 | 91.93 | 47.66 | 89.22 | 65.69 | 71.35 | 93.18 | 64.42 | 74.43 |
| CoDES (fixed, k=20, m=15) | 72.37 | 91.77 | 50.69 | **89.43** | 66.57 | **71.21** | 93.91 | **66.47** | 75.30 |
| CoDES (dataset-specific best) | **72.99** | **92.04** | **51.01** | **89.43** | **66.62** | **71.21** | 94.20 | **66.47** | **75.50** |

고정 ($k$=20, $m$=15) 설정의 평균 정확도는 75.30%로, 데이터셋별로 최적화된 ($k$, $m$) 결과(75.50%) 대비 단 0.20%p의 차이만을 보였으며, CuPL 대비로는 +0.87%p의 일관된 향상을 유지하였다. 이는 제안 방법론의 성능이 데이터셋별 하이퍼파라미터 튜닝에 의존하지 않으며, 단일 고정값 설정에서도 충분히 효과적임을 보여준다.

## A.2 경쟁 디스크립터 유사도 집계 방식 비교

본 연구의 식 (5)에서는 경쟁 디스크립터들의 이미지 유사도를 평균(mean)으로 집계하여 변별력 점수를 산출한다. 본 부록에서는 해당 설계 선택의 타당성을 검증하기 위해, 가장 유사한 하나의 경쟁 디스크립터만을 활용하는 Max 방식과 가장 유사한 상위 3개의 경쟁 디스크립터의 평균을 활용하는 Top-3 방식을 비교 대상으로 실험을 수행하였다. <Table A2>는 세 가지 집계 방식의 8개 데이터셋 평균 정확도를 비교한 결과를 보여준다.

**Table A2.** Performance comparison of different aggregation methods for rival descriptor similarity in Eq. (5) on 8 benchmark datasets using ViT-B/16 backbone. The best performance is highlighted in bold.

| | flowers | pets | dtd | food | sun | ucf101 | caltech | cars | Avg |
|---|---|---|---|---|---|---|---|---|---|
| Max | 72.58 | 91.99 | 50.69 | 89.34 | 66.31 | 70.92 | 93.87 | 66.22 | 75.24 |
| Top-3 Mean | 72.74 | 91.93 | 50.48 | 89.42 | 66.56 | 71.16 | **94.20** | 65.96 | 75.31 |
| Mean (Ours) | **72.99** | **92.04** | **51.01** | **89.43** | **66.62** | **71.21** | **94.20** | **66.47** | **75.50** |

실험 결과, 제안된 mean 집계 방식이 8개 데이터셋 평균 정확도 기준 75.50%로 Max(75.24%) 및 Top-3 Mean(75.31%) 대비 가장 우수한 성능을 보였다. 이는 혼동 클래스 집합 $C$가 이미 CLIP 유사도 기준 상위 $k$개로 선별된 실제 혼동 후보군이라는 점에 기인한다. 즉 $C$ 내 모든 경쟁 클래스는 실제 분류 시 변별이 필요한 클래스들이므로, 일부 강하게 혼동되는 클래스에만 의존하기보다 전체 경쟁 클래스와의 변별을 종합적으로 평가하는 것이 더 안정적인 변별력 점수 산출로 이어진다.

## 저자소개

**박소연:** 숙명여자대학교 통계학과에서 2024년 학사학위를 취득하고 고려대학교 산업경영공학과에서 2026년 석사학위를 취득하였다. 연구분야는 test-time adaptation이다.

**김지현:** 홍익대학교 경영학과에서 2022년 학사학위를 취득하고 고려대학교 산업경영공학과에서 석박사통합과정에 재학 중이다. 연구분야는 domain adaptation, test-time adaptation이다.

**김성범:** 고려대학교 산업경영공학부 교수로 재직하고 있으며, 인공지능공학연구소 소장, 기업산학연협력센터 센터장, 한국데이터마이닝학회 회장을 역임했다. 미국 University of Texas at Arlington 산업공학과에서 교수를 역임하였으며, 한양대학교 산업공학과에서 학사학위를 미국 Georgia Institute of Technology에서 산업시스템공학 석사 및 박사학위를 취득하였다. 인공지능, 머신러닝, 최적화 방법론을 개발하고 이를 다양한 공학, 자연과학, 사회과학 분야에 응용하는 연구를 수행하고 있다.