01. Cross-Model Pseudo-Labeling for Semi-Supervised Action Recognition (2022, CVPR)

- supervised learning에서는 많은 양의 labeled 데이터가 필요하지만, 현실에선 거의 불가능함

- 하지만 unlabeled 데이터는 비교적 쉽게 구할 수 있음

- 이러한 unlabeled 데이터를 최대한 활용하는 것이 좋음

- 가장 대표적인 방법은 unlabeled 데이터에 임의의 가짜 레이블 (pseudo label)을 지정하여 이를 마치 labeled 데이터인 것처럼 훈련에 사용하는 것

- 대부분의 연구에서는 labeled 데이터로 학습된 분류기가 unlabeled 데이터의 label을 예측함

- 하지만, 적은 labeled 데이터로 학습된 분류기는 매우 부정확할 수 있고, 따라서 불확실한 pseudo-label을 생성할 수 있음

- 따라서, 본 연구는 정확한 pseudo-labeling을 위한 접근법을 제시하고 있음

- motivation으로, 연구진들은 서로 다른 크기의 모델은 주어진 액션 비디오의 서로 다른 측면에 집중하고 있음을 발견함: 작은 모델의 경우 temporal dynamics, 큰 모델의 경우 spatial semantics.

- 이는 만약 작은 모델과 큰 모델을 같이 pseudo-labeling 과정에 사용한다면 각자가 강점이 있는 액션 종류에 대한 상호 보완적인 label 지정이 가능할 것임

- 본 연구에서는 Cross-Model Pseudo-Labeling (CMPL) 방법을 제안함: primary backbone은 서로 구조가 다르고 더욱 가벼운 네트워크와 협엽함.

- 이러한 방식은 이전의 방식들보다 퀄리티 높은 pseudo-labeling 결과를 이끌어냄

- pseudo-labeling의 목적은 가능한 많은 high quality pseudo-labels를 만드는 것

- 하지만, 학습 가능한 labeled 데이터가 매우 적다면, labels를 생성하는 분류기의 정확도도 매우 낮아짐

- 따라서, 본 연구의 접근법은 서로 다른 구조의 네트워크를 학습시켜 서로가 서로에게 pseudo-labels를 생성하도록 함 (Primary backbone, Auxiliary network)

- 이는 두 네트워크가 서로의 상호보완적인 representations을 학습하도록 함

- 제한적인 labeled 데이터로 효과적으로 학습한 모델은 더욱 많은 high quality pseudo-labels를 생성하게 되고, 이는 결국 semi-supervised learning의 성능 향상으로 이어짐

- auxiliary network는 weakly augmented view에 대한 예측을 수행하고, 예측된 category-wise probabilities가 특정 threshold를 넘을 경우, 해당 category(class)를 strongly augmented view의 label(pseudo-label)로 사용하여 primary bakbone을 훈련

- 같은 방식으로 auxiliary network도 훈련

- inference는 오직 primary backbone만 사용

- baseline model인 FixMatch의 성능을 크게 개선

- 제안하는 방법은 추가적인 input 없이 raw RGB 비디오만 사용한다는 점에서 의의가 있음

- figure나 수식 등 매우 깔끔함

- ablation study, empirical analysis를 통해 제안하는 방법에 대해 자세하게 분석함

02. Task Discrepancy Maximization for Fine-Grained Few-Shot Classification (2022, CVPR)

- 딥러닝은 엄청나게 발전하였지만, 딥러닝이 기대만큼의 성능을 내기 위해서는 수많은 labeled 데이터셋이 필요함

- labeled 데이터가 적다면 성능은 급격히 낮아짐

- 대표적인 해결방법으로는 few-shot learning이 있음

- few-shot learning의 목적은 새로운 class에도 잘 적응하는 모델을 훈련시키는 것

- 학습된 모델이 새로운 class에 대한 적절한 표현을 인코딩하는 것이 few-shot learning의 핵심

- 이를 위해 주로 metric learning이 사용됨: 학습된 모델은 새로운 class들에 대한 representation으로 각 class의 cluster를 구성하고, 각 cluster의 center와 query의 representation 간의 거리를 기반으로 inference

- 이러한 방식의 효과를 강화하기 위해 task-dynamic feature alignment strategies이 사용됨

- 이 중 spatial alignment methods는 support set과 query set의 object location 차이로 인한 degradation을 해결하기 위해 제안됨

- 반면 channel alignment methods는 feature map 수준의 변환을 통해 새로운 class를 더욱 잘 표현하도록 함

- 이러한 방법들은 충분히 효과적이었지만, 이는 fine-grained image classification(FGIC)의 경우에는 부족함

- FGIC의 경우에는 각 클래스를 구분할 수 있는 distinctive한 attributes들이 필요함

- 따라서, 본 연구에서는 channel weight를 통해 object의 discriminatory details를 localize하는 few-shot 접근법을 제시함

- 즉, 본 연구에서는 각 채널마다 채널에 weight를 주어서 discriminative한 regions를 localize하는 모듈인 Task Discrepancy Maximization (TDM)을 제안함

- TDM은 discriminative한 regions를 표현하는 채널은 강조하고 나머지 불필요한 채널의 영향은 억제함

- TDM은 Support Attention Module (SAM)과 Query Attention Module (QAM)으로 구성됨

- SAM과 QAM은 각각 support set과 query set에 적용되어 각각의 채널 가중치를 도출함

- 두 sub-module로부터의 가중치는 task-specific weight로 통합되며, 이는 최종적으로 task-adaptive feature maps를 구성하기 위해 사용됨

- 먼저 feature extractor는 주어진 이미지로부터 feature map을 추출함

- 각 클래스의 k개 support image들의 feature map의 평균이 prototype feature map으로 사용됨

- prototype feature map의 채널 방향 평균이 mean spatial features: 해당 클래스의 salient object regions를 표현

- 한 클래스의 prototype feature map의 각 채널과 해당 클래스의 mean spatial features의 MAE를 계산하여 intra-class representatives score를 계산함: 각 채널이 해당 클래스에 대해 어느 정도의 대표성을 지니는지 계산

- 한 클래스의 prototype feature map의 각 채널과 다른 클래스의 mean spatial features의 MAE를 계산하여 inter-class representatives score를 계산함: 각 채널이 다른 클레스와 구분되는 특징을 얼마나 보유하고 있는지 계산

- SAM은 class prototype을 입력으로 받아 위의 두 스코어를 계산함

- 두 스코어는 FC 레이어를 거쳐 SAM이 사용하는 가중치로 변환됨

- 두 가중치는 weighted sum을 통해 support weight vector가됨

- support weight vector는 해당 클래스의 discriminative한 채널을 강조하고 클래스 간 공통적으로 공유하는 채널의 영향은 줄이는 역할을 함

- QAM의 경우도 비슷하게 처리하되, query의 경우 클래스 정보를 알 수 없으므로 채널 간의 관계만을 반영함

- query 이미지 feature map의 각 채널과 채널 방향 평균이 적용된 query 이미지의 mean spatial feature 사이의 MAE를 계산하여 채널 수준의 대표성을 계산함

- 계산된 score는 FC 레이어를 거쳐 query weight vector가 됨

- query weight vector는 모델이 query 이미지의 object-related 정보에 집중하도록 함

- support weight vector와 query weight vector는 합쳐져서 task-specific weight vector가 되며, support set의 이미지와 query 이미지의 feature map은 모두 task weight에 의해 task-specific한 feature map으로 변환됨

- refine된 support set 및 query image의 feature map을 통해 inference 수행

- TDM은 여러 few-shot system과 혼합되어 사용할 수 있다는 장점이 있음

- 이를 위해 여러 few-shot system과 혼합하여 실험 진행

- TDM을 같이 사용했을 때 거의 모든 datasets에서 성능 개선이 있었음

- ablation study도 진행함: sub-modules, pooling strategy, metric

- discussion 및 limitation 섹션을 통해 제안하는 방법의 contribution을 강조하거나, 다른 방법과의 차이점을 강조하고, 한계점을 언급하는 것도 좋아보임 (연구의 신뢰도가 높아짐)