**JARVIS MODEL 산출 보고서**

1. **서론**
   1. **학습 환경**
      1. 라이브러리  
         - Pytorch 2.4.1  
         - CUDA 11.2  
         - CUDNN 8.2.1  
         - Python 3.10.13  
         - RTX 4060Ti (16GB)
      2. 하이퍼 파라미터  
         - batch\_size : 16  
         - lr : 0.01  
         - Optimizer : SGD
   2. **모델 정보**
      1. Summary  
         : 269 layers, 9.44M, 27.7 GFLOPs
      2. 텍스트, 스크린샷, 폰트이(가) 표시된 사진

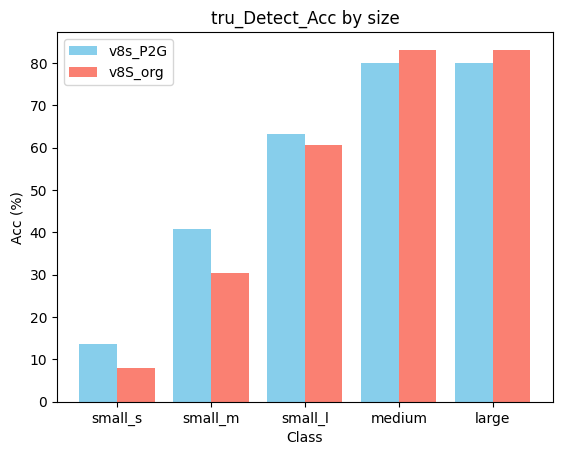
         자동 생성된 설명Yaml
   3. **성능 평가**
      1. - GmAP50 : 0.650

- NmAP50 : 0.467

* + 1. 보존율 : 72.0 %  
       (NmAP50 / GmAP50) \* 100

1. **본론**
   1. **모델링 배경(flow)**
      1. 연구 진행 과정 :  
         - Dataset과 모델 성능 상관관계 분석  
         - P2 feature map 기반 소형 객체 탐지 성능 향상  
         - P2를 적용한 S 모델 경량화 / N 모델 중량화  
         - 연산 양자화 손실 분석  
         - 모델 양자화 손실 분석  
         - 최종 모델 선정 : v8s\_ghost\_c3ghost
   2. **활용 기법**
      1. 텍스트, 스크린샷, 도표, 번호이(가) 표시된 사진

         자동 생성된 설명모델 아키텍처
      2. P2 : 소형 객체 탐지 성능을 향상을 위해 적용  
         - S 모델 분석 보고서\_p2 3, 4page 참조
      3. GhostConv & C3Ghost : 기본 S 모델에서 경량화 위해 적용, 양자화 에러 감소  
         - S 모델 분석 보고서\_GhostConvolution 3, 4page 참조  
         - 연산 양자화 분석 보고서 2, 14page 참조

1. **결론**
   1. **분석결과**
      1. P2 추가 : 소형 객체 탐지 성능 향상  
           
         small\_s, small\_m에서의 예측 정확도가 크게 향상된 것을 확인  
         객체의 크기 분류 방법은 1280\*720 이미지를 기준으로, BBox 내부의 총 픽셀 개수이며 아래와 같음  
         - small\_s : [0 ~ 460]  
         - small\_m : [460 ~ 870]  
         - small\_l : [870 ~ 1600]  
         - medium : [1600 ~ 6300]  
         - large : [6300 ~ ]
      2. 경량화 :   
         P2를 적용한 기존 yolov8s 모델의 FPS : 50.21 / Params : 11.3M  
         이후 경량화를 시도한 모델의 FPS : 54.05 / Params : 9.4M  
         - S 모델 분석 보고서\_p2 5page 참조
   2. **문제점**

\*모델 목록  
- v8s\_org : Yolo 기본 s모델  
- v8s\_P2 : 기본 모델에 P2 기법을 적용한 모델  
- v8s\_P2G : 기본 모델에 P2 기법을 적용하고, Neck 부분의 Convolution과 C3를 GhostConvolution과 C3Ghost로 변경한 모델(output 채널수 변경됨)

* + 1. 모델의 파라미터수 증가와 양자화 손실 상관관계   
       - 기존 가설 : 모델의 파라미터수가 증가하면, 양자화 손실이 증가해 보존율이 감소한다  
       (S 모델 분석 보고서\_Ghostconvolution 2page 참조)  
       - 실험 결과 : v8s\_org과 v8s\_P2를 비교했을 때, v8s\_P2가 파라미터수가 증가했지만, 보존율이 오히려 감소하는 것을 발견했다. 따라서 기존 가설에 대한 검증이 필요하다.
    2. 특정 layer의 양자화 손실간의 상관관계  
       - 기존 가설 : 연산 과정에서 여러 개의 채널을 concat 할 수록(C2f와 C3Ghost의 구조적 차이점) 양자화 손실이 증가해 보존율이 감소한다.  
       - 실험 결과 : v8s\_P2와 v8s\_P2G를 비교했을 때, 가설대로 C2f를 사용한 v8s\_P2의 보존율이 더 낮은 것을 확인했다. 하지만 기존 실험들 중 C2f와 C3를 비교한 실험은 존재하는데, C2f와 C3Ghost를 비교한 실험은 없기 때문에 추가 실험을 통해 검증이 필요하다.  
       (연산 양자화 분석 보고서 2, 14page 참조)
    3. 변인통제 오류  
       : v8s\_P2를 v8s\_P2G로 변경하는 과정에서 GhostConvolution 레이어의 output 채널 수를 변경했기 때문에 정확한 변인통제가 이루어지지 않았다. 따라서 v8s\_P2와 채널수가 동일한 v8s\_P2G 모델에 대한 추가 실험이 필요하다.
  1. **향후 계획**
     1. Conv & GhostConv & C3Ghost & C3의 연산 양자화 손실률 추가 비교
     2. v8s\_P2와 동일한 채널 수의 v8s\_P2G의 추가 실험 및 채널수 변경에 대한 영향 분석
     3. v5 버전 기준 제출할 모델 기법 적용해서 모델 특징에 맞는 양자화 메소드 적용.  
        - 양자화 분석 보고서 8page 참조