**Bi-weekly Report**

| **월간**  **성과**  **계획** | **7월 목표** | **완료 예정일** | **7월 실적 (당월 末에 실적 작성)** | **완료일** |
| --- | --- | --- | --- | --- |
| **● Yolov6-face 모델 소이넷 포팅하기**  **● ByteTrack 트랙킹 모델 C++로 구현**   * **YOLOX 디텍션 모델과 결합하여 정합성, 성능 체크할 것** | 7/14  7/28 | **● Yolov6-face 모델 소이넷 포팅하기**  **● C++ class, STL, namespace 등등 공부 및 적용**  **● YOLOX + ByteTrack 모델 C++ 구현 완료 및 성능 측정 완료**  **● ByteTrack모델 C++로 구현 완료**  **● OC-Sort 거의 구현 완료** | **7/5**  **7/14**  **7/21**  **7/21**  **7/27** |
|  | **금주 실적 [7/15 ~ 7/31]** | **완료일** | **차주 계획  [8/1 ~ 8/15]** | **완료 예정일** |
| **완료**  **Task** | **● ByteTrack 모델 완성**   * **Pypy소스를 C++로 옮겨서 성능 최적화 - (2)** * **YOLOX 모델을 가져와서 C++에서 결합할 것** * **정합성, 성능 비교 완료**   **● OC-Sort 트랙킹 모델 C++ 구현 (1)**   * **코드 분석하기** * **YOLOX + pretrained weights랑 결합** * **OC-Sort 코드 작성: association.cpp, KalmanFilter.cpp, KalmanBoxTracker.cpp, OCSort.cpp, lapjv.cpp + headers** | 7/21  7/27 | **● OC-Sort 트랙킹 모델 C++ 구현 (2)**   * **정합성 비교 결과 코드 오류 찾기 ( 디버깅 )** * **Python으로 구현된 OC-Sort랑 성능 비교하기**   **● 지금까지 작업한 결과물 정리**   * **Yolov7** * **Yolov6 Face-S** * **ByteTrack** * **OC-Sort** | 8/3    8/4 |
| **지연**  **Task** |  |  |  |  |
| **기타**  **&**  **이슈** |  |  | 연차 2개 쓸 예정 |  |
| **지난주 교수님**  **Comments or 지시사항** |  | | | |

□ 본 보고서/교재는 2021년도 과학기술정보통신부 및 정보통신기획평가원에서 주관하여 진행하는 ‘SW중심대학사업’의 결과물입니다.

□ 본 보고서의 내용을 전재할 수 없으며, 인용할 때에는 반드시 과학기술정보통신부와 정보통신기획평가원의 ‘SW중심대학’의 결과물이라는 출처를 밝혀야 합니다.