

# 김응혁

이메일: coke1324@naver.com

연락처: +821051322182

하드웨어를 전공했지만, 현재는 소프트웨어로 직무를 전환하여 C++ 전문가로 성장하고 싶은 개발자입니다. 제조업에서 영상처리 및 MFC Application 업무를 진행한 경험이 있습니다. 대용량 이미지 병렬 처리, CCD 16Tap 조정 알고리즘 개선 등 영상처리 업무를 진행했으며, 모던 C++과 스크립트 언어를 활용한 리팩토링, MFC Framework에서 PyQt5 Framework 전환 프로젝트를 맡은 경험이 있습니다. 현재는 1인 프로젝트를 진행하여 C++ 및 REST 구조에 대해 학습하고 있습니다.

경력	뷰웍스	2019.1 - 2021.5
	광영상생산팀 / 사원	
		<ul style="list-style-type: none"><li>● <b>프로그래밍 언어 통합 프로젝트</b> 2021.1 - 2021.5<ul style="list-style-type: none"><li>- 소개: C++로 구현된 UI와 Lua로 구현된 Logic을 Python 언어로 통합하기 위한 프로젝트</li><li>- 개발 인원 및 기여도: 2명, 50%</li><li>- 주요 라이브러리: MFC, PyQt5, Boost.Python</li><li>- 목적:<ol style="list-style-type: none"><li>1. 두 가지 언어를 하나의 언어로 통합하여 유지보수 및 생산성 향상</li><li>2. Referece가 부족한 Lua로 인한 생산성 저하 해결</li></ol></li><li>- 핵심 내용:<ol style="list-style-type: none"><li>1. MFC Framework에서 PyQt5 Framework로 전환</li><li>2. Lua로 구현된 Logic을 Python으로 컨버팅</li></ol></li><li>- 본인이 수행한 역할:<ol style="list-style-type: none"><li>1. MFC Framework로 구현된 일부 GUI에 대해 PyQt5 Framework로 전환</li><li>2. Lua로 구현된 기존 Logic에 대해 Pyhton으로 컨버팅</li><li>3. Boost.Python 라이브러리를 도입하여 C++, Python 간 데이터 공유</li><li>4. 기존에 Lua와 바인딩 된 C++ 함수에 대해 Python에서 호출 가능하도록 추상화</li><li>5. pytest Framework를 통한 Python 테스트 환경 구축</li></ol></li></ul></li><li>● <b>대용량 이미지 병렬 처리</b> 2020.10 - 2020.12<ul style="list-style-type: none"><li>- 소개: 고해상도 이미지 처리 속도 향상을 위한 프로젝트</li><li>- 개발 인원 및 기여도: 2명, 50%</li><li>- 주요 라이브러리: TBB, PPL, CUDA</li><li>- 목적:<ol style="list-style-type: none"><li>1. 고해상도 이미지에 대한 영상처리 및 알고리즘 속도 개선</li></ol></li></ul></li></ul>

- 핵심 내용:

1. 병렬 프로그래밍 속도 비교 및 라이브러리 선정
2. 병렬 프로그래밍 적용 가능 Logic 분석 및 적용

- 본인이 수행한 역할:

1. TBB 및 PPL 라이브러리 동작 방식 파악 및 속도 측정
2. Histogram, 이미지 연산, Defect Cluster 알고리즘에 병렬 프로그래밍 적용

● Pixel Inspector Tool 개발 2020.7 - 2020.9

- 소개: 이미지에 대한 Pixel value 파악을 위한 Tool 개발
- 주요 라이브러리: MFC Grid control

- 목적:

1. 사람이 직접 계산하던 부분을 프로그램을 통해 계산하여 효과적인 카메라 불량 검출

- 핵심 내용:

1. 이미지에 대한 실시간 Block 단위 Pixel 최대, 최소, 평균, 편차 계산 기능
2. Block 단위 편차 filtering 기능

- 본인이 수행한 역할:

1. 이미지 포맷 별 계산 기능 구현 및 클래스 추상화
2. 이미지 포맷 별 Block 단위 편차 filtering 기능 구현
3. Pixel value 출력을 위한 이미지 포맷 별 MFC 클래스 추상화
4. 이미지 포맷에 대한 Strategy Pattern 적용

● CCD 16Tap 조정 알고리즘 개선 2020.3 - 2020.4

- 소개: CCD 16Tap 조정 알고리즘 개선 프로젝트
- 사용 언어: Lua

- 목적:

1. 비효율적인 기존 CCD 16Tap 조정 알고리즘 개선

- 핵심 내용:

1. 기존 알고리즘 동작 파악 및 개선

- 본인이 수행한 역할:

1. 기존 알고리즘 동작 파악
  - > 기존 Tap을 정해서 나머지 Tap 간의 mismatch를 반복문을 이용해서 계산
  - > Big-O( $N^2$ )의 시간복잡도
2. 개선된 알고리즘 구현
  - > 기존 Tap을 정해서 나머지 Tap 간의 mismatch를 선형 보간법을 이용해서 계산
  - > Big-O( $N$ )의 시간복잡도

● 카메라 조정 프로그램 리팩토링 2019.7 - 2019.12

- 소개: 노후화된 프로그램에 대한 리팩토링
- 개발 인원 및 기여도: 2명, 50%
- 주요 라이브러리: Luabind, Google Test

- 목적:

1. 메모리 누수 해결
2. 레거시 코드 개선
3. 수정이 잦은 C++ Logic으로 인한 빌드 시간 연장 개선
4. 비효율적인 알고리즘 개선
5. 카메라 설정 ini 파일 개선

- 핵심 내용:

1. 소유권이 없는 Raw Pointer 관리 미흡으로 인한 메모리 누수 해결
2. 이력이 없는 레거시 코드 동작 파악 및 개선
3. 런타임 중 실시간으로 C++로 구현된 Logic 을 수정할 수 있도록 개선
4. 부하가 큰 기존 알고리즘에 대한 동작 파악 및 개선
5. 유효성 확인이 어려운 카메라 설정 ini 파일 개선

- 본인이 수행한 역할:

1. 소유권이 없는 Raw Pointer를 관리하기 위한 C++11 스마트 포인터 도입
2. 레거시 코드 동작 파악 및 개선을 위한 Google Test Framework를 활용한 C++ 테스트 환경 구축
3. C++로 구현된 기존 Logic에 대해 Lua로 컨버팅
4. LuaBind 라이브러리를 도입하여 C++, Lua 간 데이터 공유
5. 속도가 중요한 Logic에 대해 C++로 구현 후 Lua에서 호출 가능하도록 바인딩
6. 데이터 특징에 맞는 컨테이너로 수정하여 알고리즘의 시간 복잡도 개선
7. 하나의 파일로 관리하던 카메라 설정 ini 파일을 180개의 JSON 파일로 분리
8. JSON schema 도입으로 데이터 유효성 검증 추가

---

학력

**숭실대학교**

2012.3 - 2019.2

전자공학과

- 주요 이수과목: 전자회로, 신호처리, 반도체소자, 센서공학, 디스플레이 원리, 디지털공학 등
- 졸업논문: 베타-산화 갈륨을 이용한 이중접합 애벌런치 포토 다이오드

---

수상 및 기타

**[삼성전자] 하계 대학생 S/W 알고리즘 역량 강화 특강**

2021.7

- 기간: 2021.07 ~ 2021.08
- 목적: 자료구조 및 알고리즘 개념 학습
- 핵심 내용: S/W 문제해결 역량 향상
- 주요 내용:
  1. Linked List, Tree, Heap, Trie, Hash 등
  2. 그리디, 완전 탐색, DP, 분할 정복, 그래프 탐색 등

**Windows API를 이용한 폴더 공유 시스템**

2021.6

- 기간: 2021.06 ~ 2021.08
- 소개: Windwos에서 클라이언트 간 폴더 공유 시스템

- 목적:

1. RESTful Client 및 Server 개념 학습
2. Modern C++ 기능 학습

- 핵심 내용:

Client:

1. 폴더의 하위 목록에 대한 메시지 모니터링
2. 클라이언트 오프라인 상태에서의 폴더의 변경사항 파악
3. 다른 클라이언트와 모니터링 중인 폴더 동기화
4. 폴더의 변경사항 데이터를 서버로 전달
5. Simple Storage Service 파일 업/다운로드 기능

Server:

1. RESTful API 지원
2. 클라이언트를 구분하여 Broadcast 전송

- 본인이 수행한 역할:

Client:

1. Windows API를 이용한 폴더의 변경 사항 모니터링 기능 구현
2. cpprestsdk 라이브러리를 활용한 RESTful API 및 WebSocket Client 구현
3. 로컬용 데이터베이스 구축 및 sqlite3 라이브러리를 활용한 DAO 클래스 구현
4. aws-cpp-sdk 라이브러리를 활용한 S3 Storage Object 업/다운로드 기능 구현
5. Google Test Framework를 활용한 테스트 환경 구축

Server:

1. Flask Framework를 활용한 RESTful API 및 WebSocket Server 구현
2. WebSocket 프로토콜을 활용한 Broadcast 기능 구현
3. 서버용 데이터베이스 구축 및 SQLAlchemy 라이브러리를 활용한 DAO 클래스 구현
4. Boto3 라이브러리를 활용한 S3 Storage Object garbage collection 기능 구현
5. pytest Framework를 활용한 테스트 환경 구축

- GitHub:

1. [https://github.com/EUING/monitor\\_client](https://github.com/EUING/monitor_client)
2. [https://github.com/EUING/monitor\\_server](https://github.com/EUING/monitor_server)

---

외국어

**영어**

일상회화

- TOEIC Speaking 2017.08.19  
140점/Level 6

---

링크

<https://github.com/EUING>