

Portfolio

JINSEO CHOI

최진서

ABOUT : <https://js-choi513.github.io/JS.html>

Profile

■ Education

- 2016. 03. ~ 2018. 02. 마산대학교 로봇메카트로닉스과 전문학사 (졸업)
- 2018. 03. ~ 2022. 02. 창원대학교 컴퓨터공학과 학사 (졸업)
- 2022. 03. ~ 2022. 12. 창원대학교 컴퓨터공학과 석사 (수료)
- 2023. 03. ~ 가천대학교 컴퓨터공학과 석사 (졸업예정)

■ Experience

- 2018. 12. ~ 2022. 02. Noslab 학부과정 연구원
- 2021. 12. ~ 2022. 01. NH농협은행(창원 상남지점) 전산관리 현장실습
- 2022. 02. ~ Noslab 석사과정 연구원

Tech skills



Language

C/C++
Python



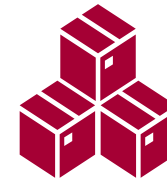
System

Linux kernel
Ubuntu
Bash
CUDA
ftrace
fio



Monitoring

Grafana
TensorBoard
nvprof
sysstat



Virtualization

Cgroups
Docker
Anaconda3



AI/ML

PyTorch
TensorFlow
Scikit-learn
Pandas

Projects

◆ TensorFlow 성능 분석

- I/O 간섭에 의한 텐서플로우 성능 분석(2021. 06.)

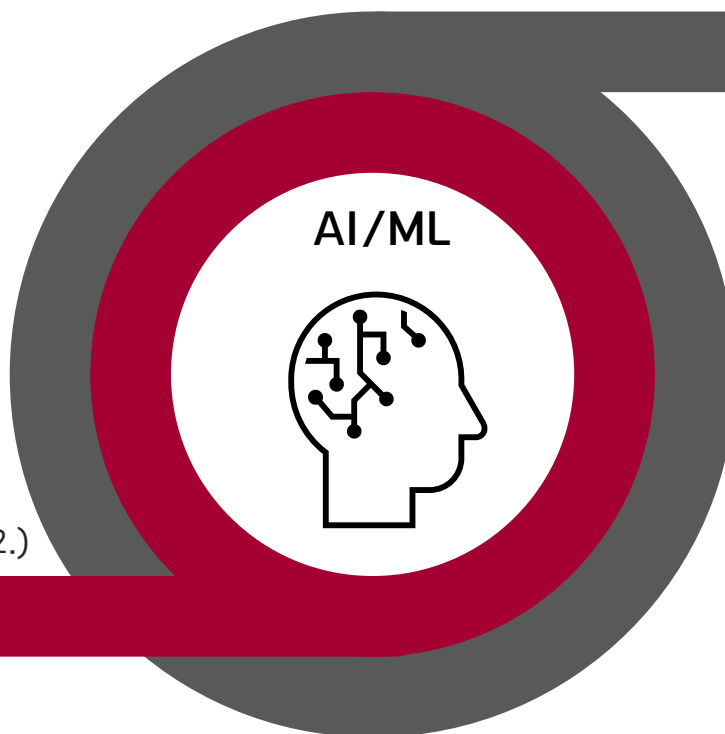
◆ 딥 러닝 파이프라인 최적화 연구

- Interleaved Data Processing Scheme for Optimizing Tensorflow Framework(2021. 11.)
- 딥 러닝 훈련 시간 개선을 위한 쓰레드 기반 셔플링 기법(2022. 06.)
- Overapped Data Processing Scheme for Accerlation Training and Validation in Machine Learning(2022.07)

◆ 딥 러닝 연산분석 연구

- 딥 러닝 연산을 위한 GPU/CPU 성능 분석(2022.12.)

Optimization



Application

◆ 딥 러닝 기반 주택가격지수 예측

- End-to-end 애플리케이션 개발
- 창원시 지역구별 6개월 주택가격지수 예측
- Tensorflow, Flask 사용

◆ 딥 러닝 기반 문서 내 테이블 검출 기법

- 이미지 형식 문서에서 테이블 검출 모델 정확도 개선

◆ 딥 러닝 기술을 이용한 DX 연구

- 이미지 형식 설계도면 전산화
- 텍스트 위치기반 심볼 탐지

Publication / Patent

- 최진서, 강동현, **"I/O 간섭에 의한 텐서플로우 성능분석"**, 한국컴퓨터종합학술대회 논문집(KCC), 2021. 06 **Conference**
 - 학부생 부문 우수논문상
- J. Choi, M. Cho, D. Kang, **"Interleaved data processing scheme for optimizing tensorflow framework"**, *In Proceedings of the IEEE 11th International Conference on Consumer Electronics*, 2021. 11 **Conference**
- 강동현, 최진서, **"병렬 처리 기반 훈련 및 검증 성능 향상 장치 및 방법"** **Patent**
 - 출원번호 10-2022-0000138, 2022.01.03
- 최진서, 강동현, **"딥 러닝 훈련 시간 개선을 위한 스레드 기반 셔플링 기법"**, 정보과학회 컴퓨팅의 실제 논문지(KTCP), 제 28권, 6호, 2022. 06 **Journal**
- 최진서, 강동현, **"훈련 및 검증 성능 개선을 위한 텐서플로우 병렬 처리 기법"**, 정보과학회논문지(JOK), 제 46권, 6호, 2022. 05 **Journal**
- 최진서, 강동현, **"딥 러닝 기반 테이블 검출 기법"**, 한국컴퓨터종합학술대회 논문집(KCC), 2022. 06 **Conference**
- J.Choi, D. Kang, **"Overlapped data processing scheme for Accelerating Training and validation in machine learning"**, *IEEE Access*, Vol. 10, pp. 72015-72023, July 2022. 07 **Journal** **SCI**
- 강동현, 최진서, **"딥 러닝 기술을 이용한 객체 인식 방법 및 시스템"** **Patent**
 - 출원번호 10-2022-0122336, 2022.09.27
- 최진서, 강동현, **"딥 러닝 연산을 위한 GPU/CPU 성능분석"**, 한국 소프트웨어 종합 학술대회 논문집(KSC), 2022. 12 **Conference**

◆ TensorFlow 성능 분석

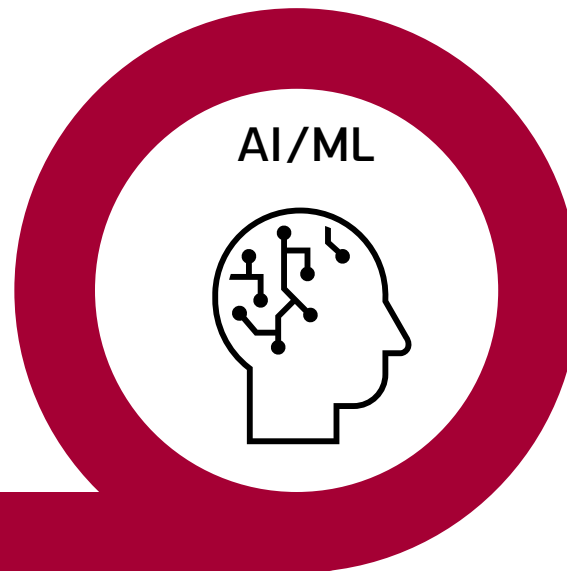
- I/O 간섭에 의한 텐서플로우 성능 분석(2021. 06.)

◆ 딥 러닝 파이프라인 최적화 연구

- 딥 러닝 훈련 시간 개선을 위한 쓰레드 기반 셔플링 기법(2021. 06.)
- Overapped Data Processing Scheme for Accerlation Training and Validation in Machine Learning(2022.07)

◆ 딥 러닝 연산분석 연구

- 딥 러닝 연산을 위한 GPU/CPU 성능 분석(2022.12.)

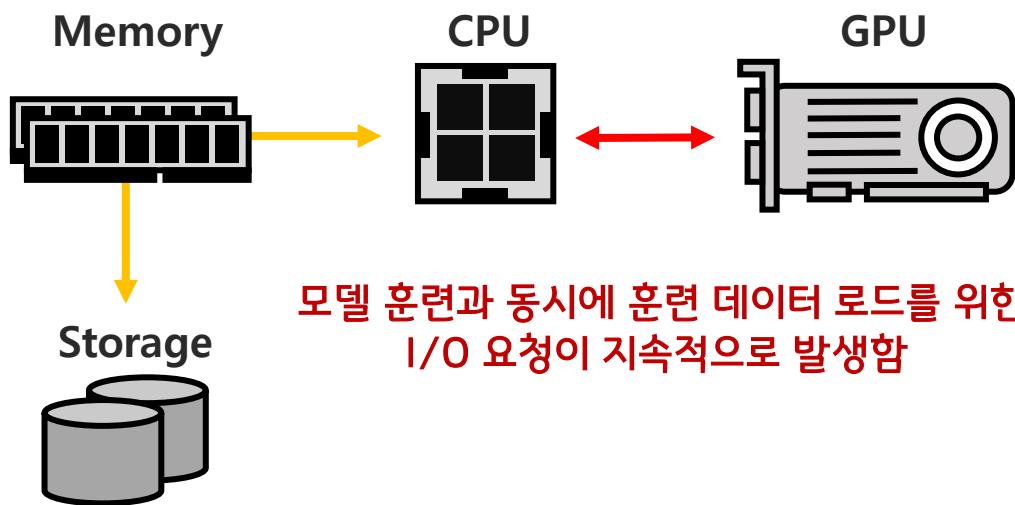
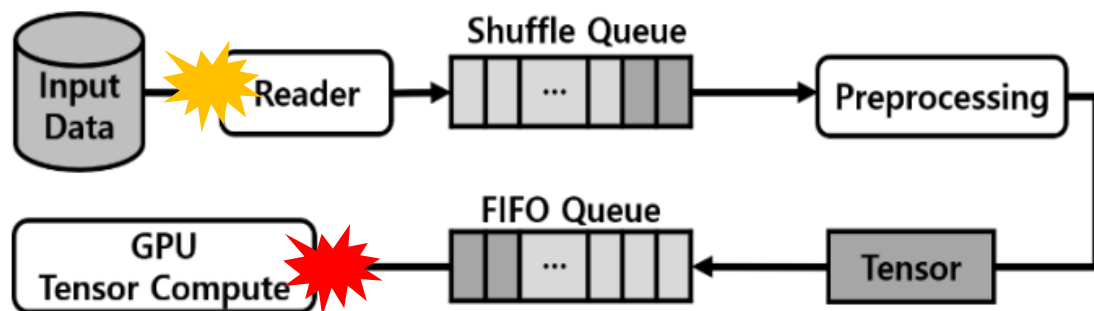


Optimization

Optimization Projects

DETAILS

TensorFlow 성능 분석



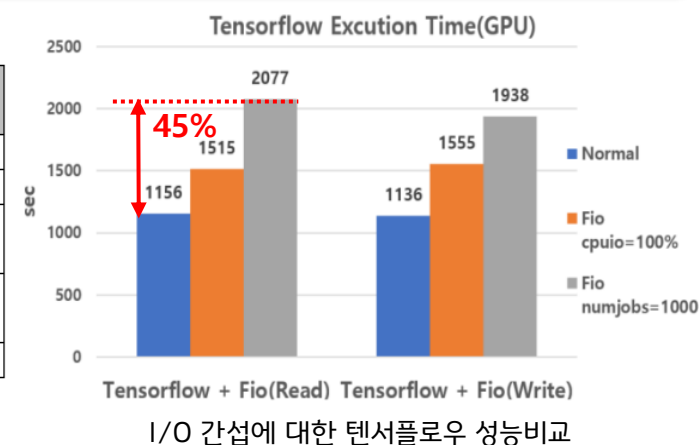
✓ I/O 간섭에 의한 텐서플로우 성능분석

- 훈련 데이터의 규모가 클 경우, In-memory 학습이 어려움
- 일반적으로 파이프라인을 구성하여 데이터 로드와 학습을 동시에 수행
- Fio를 사용하여 I/O 간섭 시 딥 러닝 모델의 훈련시간 변화 측정
- 높은 I/O 간섭 발생 시 수행시간이 최대 45% 증가하는 것을 확인

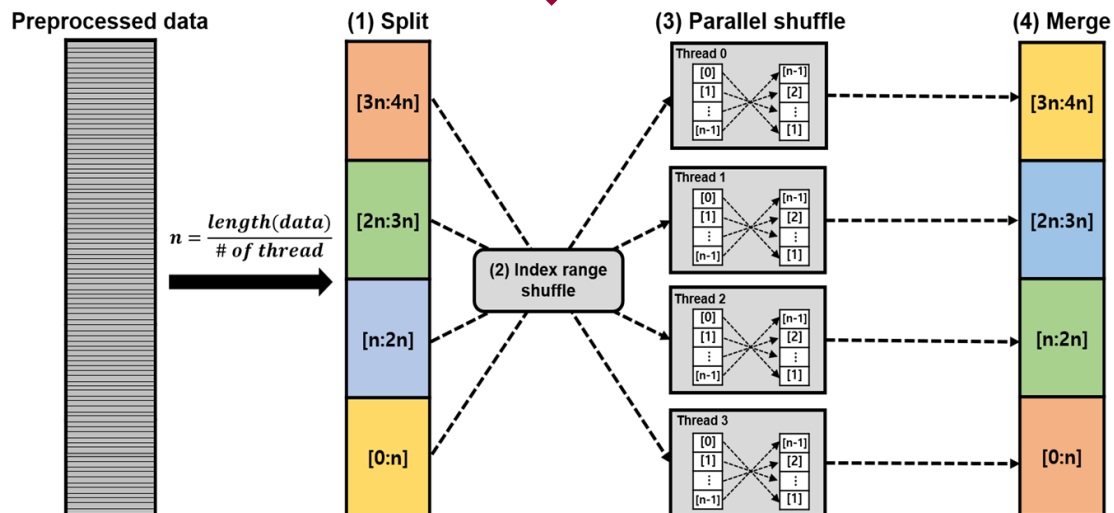
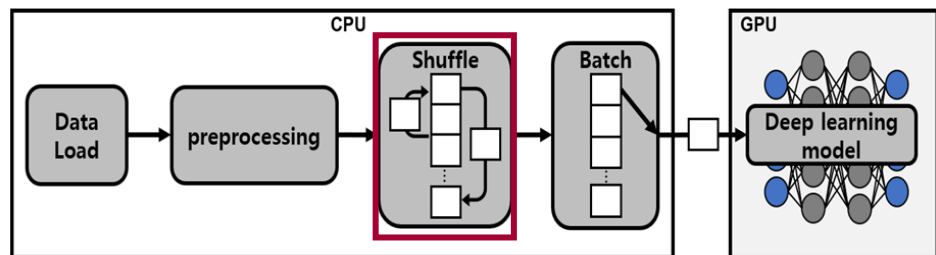
✓ 한국 컴퓨터종합학술대회

- 학부생 부문 우수논문상 🏆
- 한국정보과학회 학술발표논문집, pp. 2145-2147, 2021. 06

	Tensorflow	Tensorflow+Fio (numjobs=1000)
Avg Step Time	1296.4(ms)	15409.5(ms)
Input Time	1193.8(ms)	15257.9(ms)
Device Compute Time	100.2(ms)	112.3(ms)
Kernel Launch Time	0.9(ms)	10.9(ms)
Others	1.5(ms)	14.5(ms)



딥 러닝 파이프라인 최적화(1)

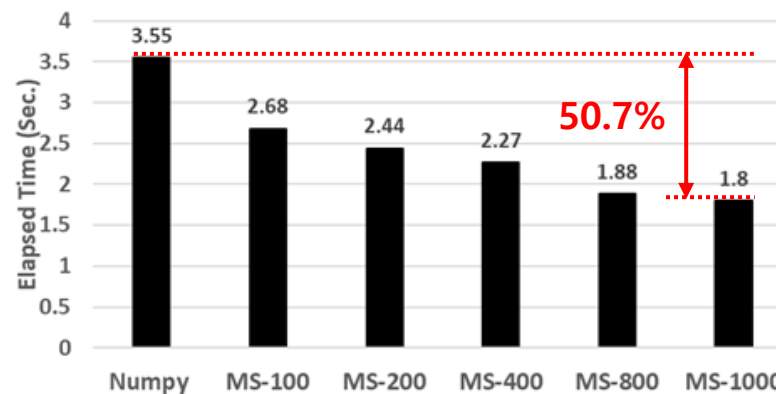


✓ 딥 러닝 훈련시간 개선을 위한 스레드 기반 셔플링 기법

- 모델의 과적합을 방지하기 위해 훈련데이터 셔플 수행
- 기존 데이터 셔플은 단일 스레드를 사용하여 수행해 왔음
- 멀티 스레드를 사용하여 데이터 셔플을 병렬 처리하여 처리시간 감소
- 기존 Numpy 라이브러리 대비 데이터셔플 시간을 최대 50.7% 감소

✓ 정보과학회 컴퓨팅의 실제 논문지 (KCI 등재)

- 정보과학회 컴퓨팅의 실제 논문지, vol. 28(2), pp. 75-80. 2022. 02



MNIST 데이터셋 셔플 수행시간 비교

*MS: Multi-thread shuffling

딥 러닝 파이프라인 최적화(2)

✓ 훈련 및 검증 성능 개선을 위한 텐서플로우 병렬처리 기법

- 기존 딥 러닝 파이프라인 처리과정 세부분석
- GPU/CPU의 IDLE구간이 존재
- 데이터로드, 전처리 시, CPU busy, GPU idle
- 학습모델 훈련, 검증 시, CPU idle, GPU busy
- 학습모델 검증 데이터 전처리 단계, 추론 단계 병렬처리
- 현재 Epoch 검증단계 추론, 다음 Epoch 훈련데이터 전처리 과정 중첩

✓ 장점

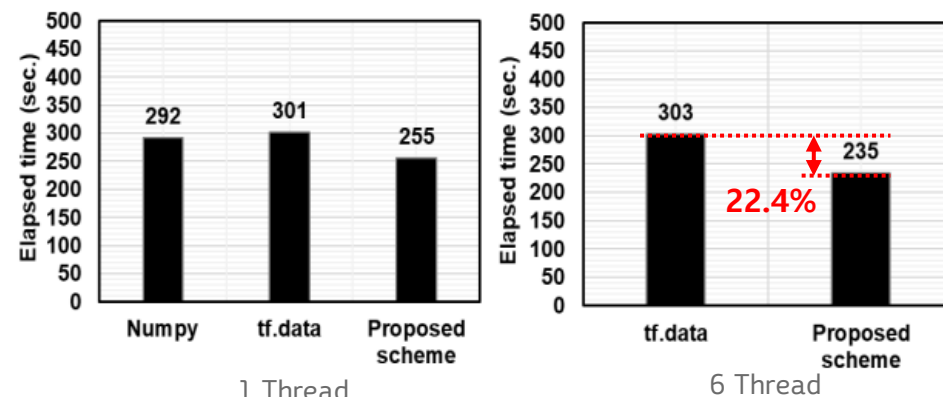
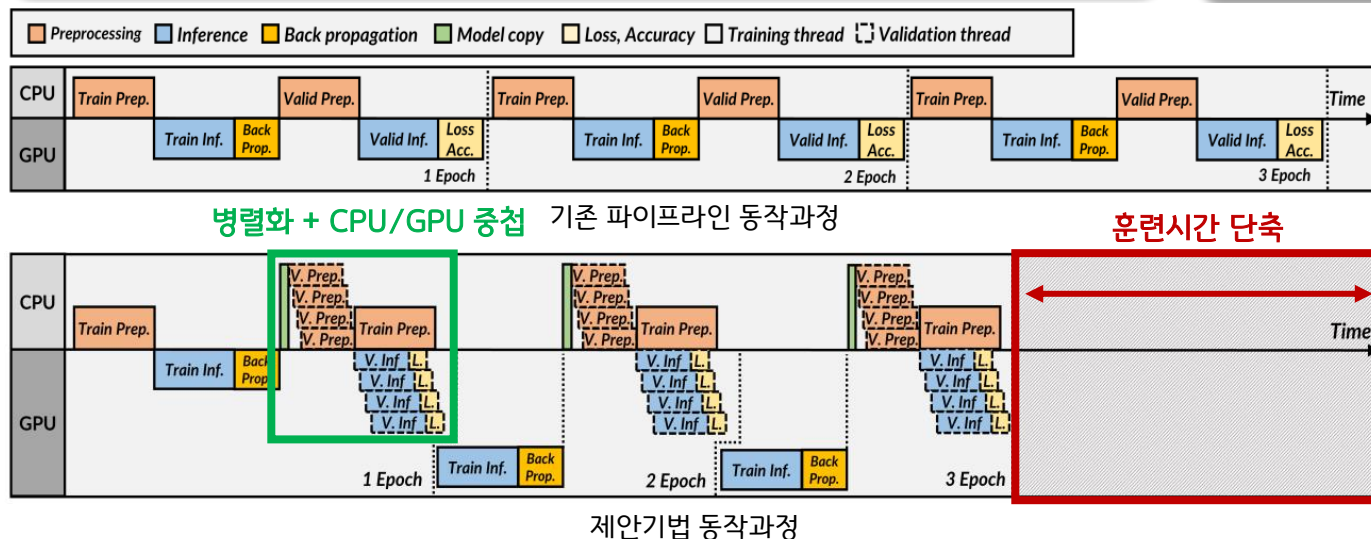
- 중첩+병렬처리로 인해 기존방식(tf.data) 대비 최대 22.4% 훈련시간 감소
- GPU, CPU idle 구간 감소, 효율증가

✓ 단점

- 데이터셋 규모가 작을 경우 훈련시간 감소량이 작음 (쓰레드 생성 오버헤드 존재)

✓ 한국정보과학회논문지 (KCI 우수 등재)

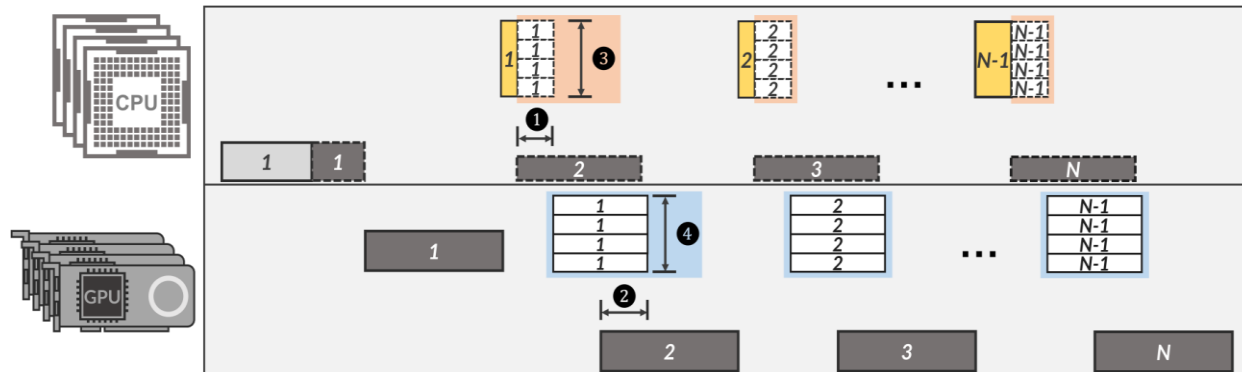
- 데이터셋 규모가 작을 경우 훈련시간 감소량이 작음



1GB MNIST 훈련시간 비교

딥 러닝 파이프라인 최적화(3)

Data load
 Model copy
 Train preprocessing
 Training
 Validation preprocessing
 Validation



✓ ol.data

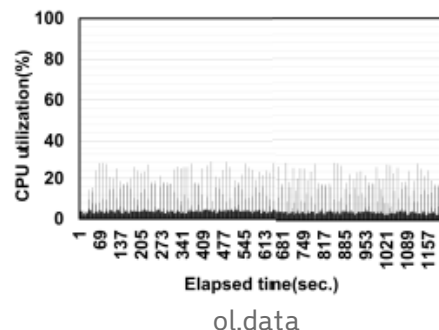
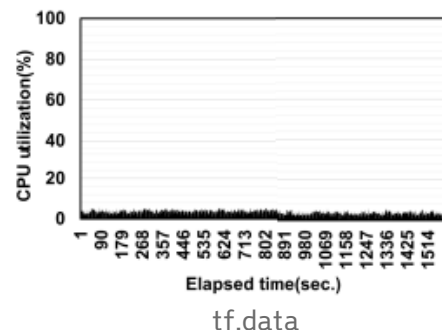
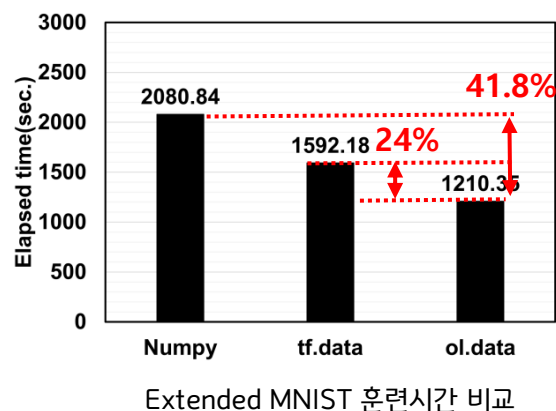
- 훈련과정 병렬처리를 위한 파이프라인 프레임워크
- 중첩구간 추가(4개 동작 중첩, 2개 동작 병렬처리)
- CPU: 현재 Epoch 검증데이터 전처리 + 다음 Epoch 훈련데이터 전처리(①)
- GPU: 현재 Epoch 검증추론 + 다음 Epoch 훈련(②)
- 최대 41.8% 훈련시간 감소, CPU 사용률 75.7%, GPU 사용률 38.7% 증가

✓ 특허출원

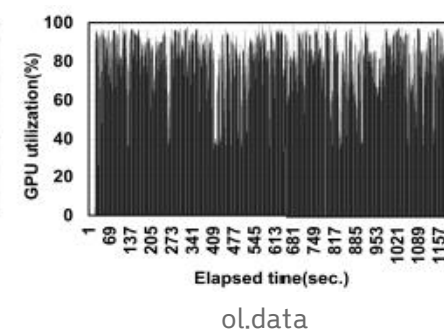
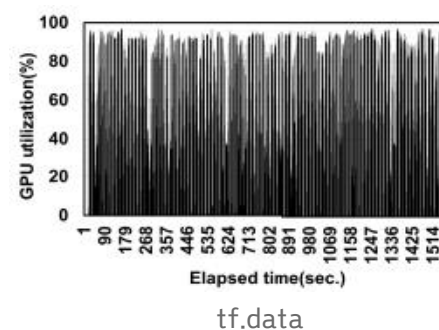
- 출원번호 10-2022-0000138, 2022.01.03

✓ IEEE Access (SCI 등재)

- IEEE Access, vol. 10, pp. 72015-72023, 2022. 07

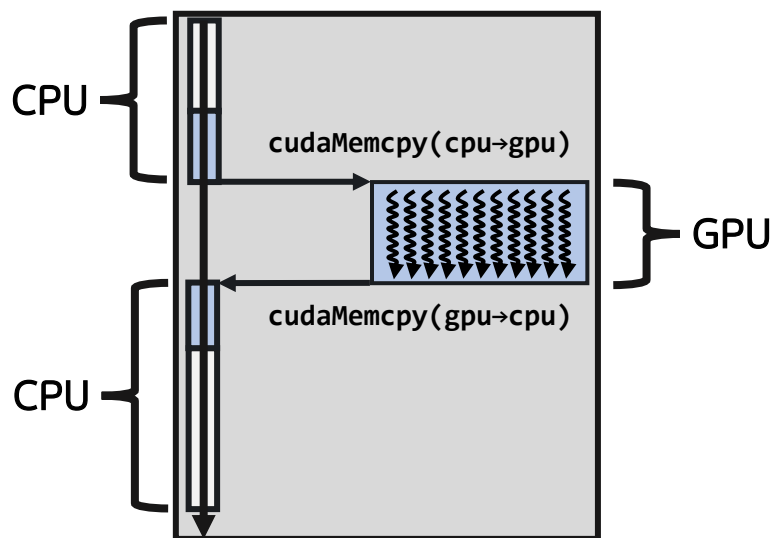
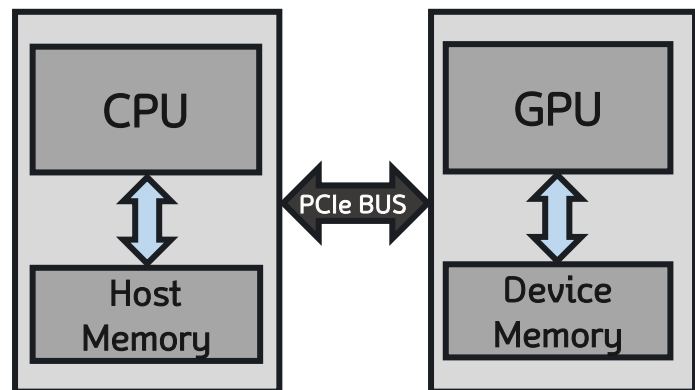


CPU 사용률 비교



GPU 사용률 비교

딥 러닝 연산분석 연구

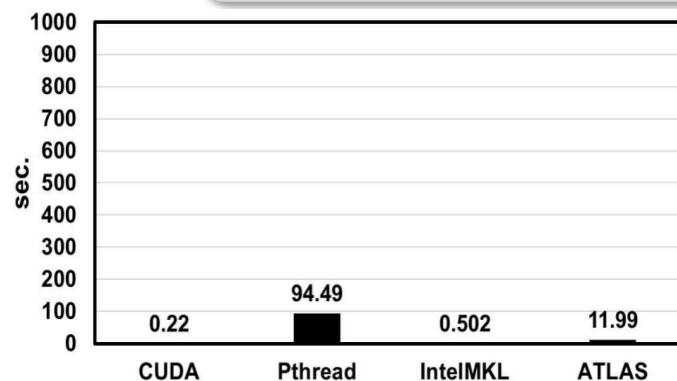


✓ 딥 러닝 연산을 위한 GPU/CPU 성능분석

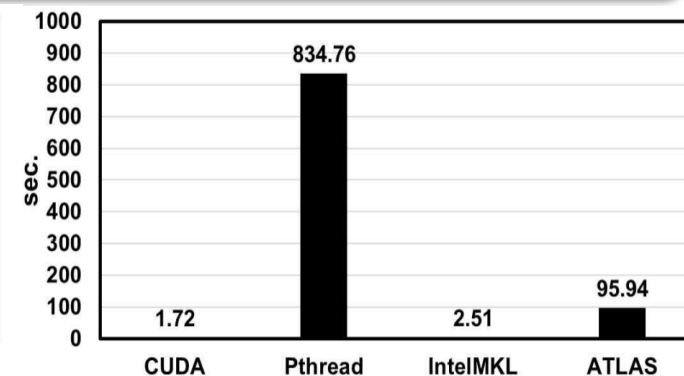
- 최근에는 CPU 기반 연산 라이브러리가 높은 성능을 보임
- 이기종 시스템인 GPU는 CUDA를 통해 제어 (CPU-GPU간 오버헤드 존재)
- GPU와 CPU기반 라이브러리의 행렬 곱셈 처리성능 비교
- CPU-GPU 간 데이터 이동 오버헤드를 포함해도 CPU 대비 최대 485배 높은 성능을 보임

✓ 한국소프트웨어종합학술대회

- 한국정보과학회 학술발표논문집, 1145-1147, 2022. 12



4096 x 4096



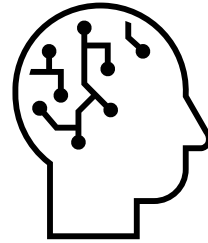
8192 x 8192

행렬 곱 연산 처리시간 비교

Application Projects

DETAILS

AI/ML



Application

◆ 딥 러닝 기반 주택가격지수 예측

- End-to-end 애플리케이션 개발
- 창원시 지역구별 6개월 주택가격지수 예측
- Tensorflow, Flask 사용

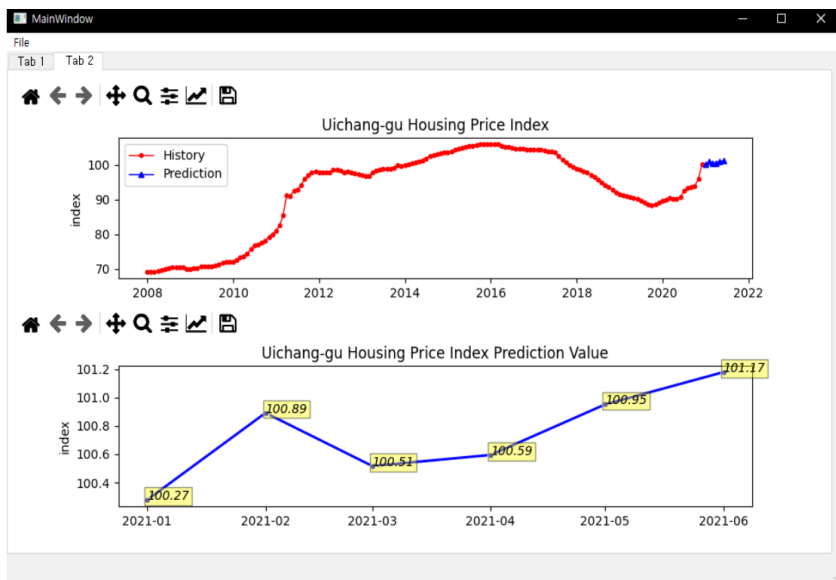
◆ 문서 내 테이블 검출 기법

- 이미지 형식 문서에서 테이블 검출 모델 정확도 개선

◆ 딥 러닝 기술을 이용한 DX 연구

- 이미지 형식 설계도면 전산화
- 텍스트 위치기반 심볼 탐지

딥 러닝 기반 주택가격지수 예측



앱 실행 후 창원시 의창구 주택가격지수 예측 실행 결과

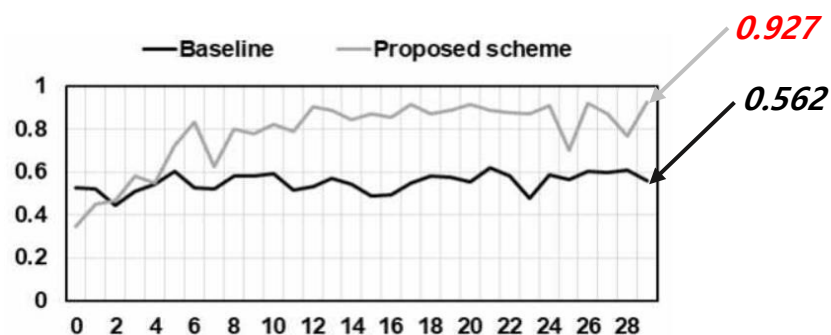
✓ 창원시 지역구별 주택가격지수 예측 서비스

- 주택가격과 관련된 5가지 경제지표 데이터를 학습
- LSTM기반 시계열 데이터 예측 모델 구현
- 전국단위 데이터셋을 학습한 모델을 사용하여 전이학습 및 미세조정
- 입력 데이터에 대해 모델의 예측 값을 반환하는 웹 API 서버 구축
- <https://github.com/JS-Choi513/Predicting-the-Housing-Price-Index-in-Changwon-City-Using-the-LSTM-Model>

✓ Technical skills

- App GUI 구현 (python, PyQt5, matplotlib)
- 웹서버 구축(FLASK, AWS)
- 딥러닝 모델 설계(TensorFlow)

문서 내 테이블 검출기법



F-1 score 비교

4.3.2 Algorithm parameters
The only parameters of the AHP algorithm are the parameters of the VGG19 model, which are listed in Table 3. The default values are listed in the column 'Default' of Table 3.

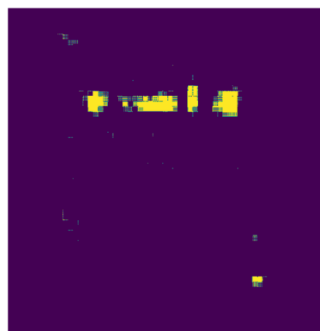
Table 3. AHP parameters

Parameter	Default	Variable Range
Learning rate	0.001	[0.0001, 0.01]
Batch size	32	[16, 64]
Number of epochs	100	[50, 200]

The results of the experiments are shown in Figure 2, 3, 4, and 5. As described in [10], we used a grid search to find the best parameters for the proposed scheme. The results of the experiments are shown in Figure 2, 3, 4, and 5. As described in [10], we used a grid search to find the best parameters for the proposed scheme. The results of the experiments are shown in Figure 2, 3, 4, and 5.

4.3.3 Model evaluation
The results of the experiments are shown in Figure 2, 3, 4, and 5. As described in [10], we used a grid search to find the best parameters for the proposed scheme. The results of the experiments are shown in Figure 2, 3, 4, and 5.

문서 이미지



기존기법 테이블 탐지결과



제안기법 테이블 탐지결과

문서 내 테이블 탐지결과 비교

✓ 딥러닝 기반 문서 내 테이블 검출 기법

- 기존 모델은 FCN 구조를 기반으로 1개의 인코더와 2개의 디코더를 사용
- 인코더는 이미지분류모델 VGG19를 사용하며, 각 디코더는 인코더로부터 문서 내 테이블 영역과 컬럼 영역을 추론
- VGG19의 기울기 소실로 인한 정확도 감소를 해결하기 위해 인코더로 ResNet추가
- VGG19의 출력 값을 ResNet출력 값으로 보정
- 기존기법 대비 F1-score가 64.9%증가

✓ 한국 컴퓨터종합학술대회

- 한국정보과학회 학술발표논문집, pp. 903-932, 2022. 06

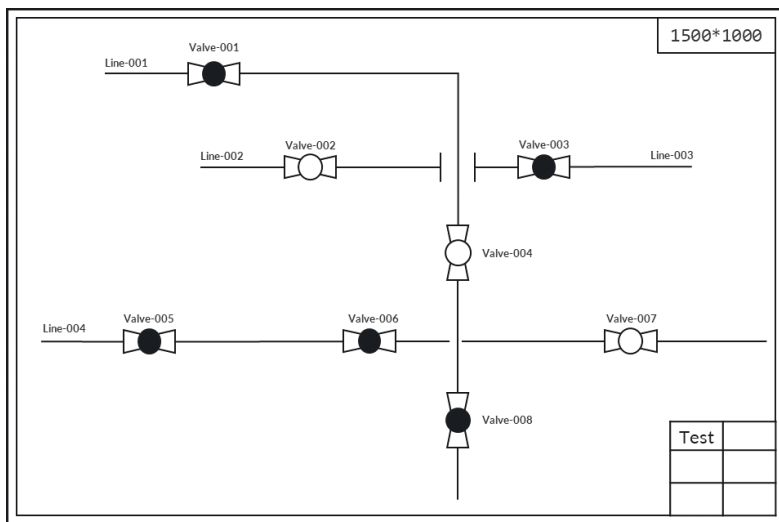
✓ Technical skills

- 논문 내 모델구현 및 제안기법 설계(Tensorflow)
- 데이터셋 크롤링(python)
- 예측결과 시각화(matplot)

딥 러닝 기술을 사용한 DX연구

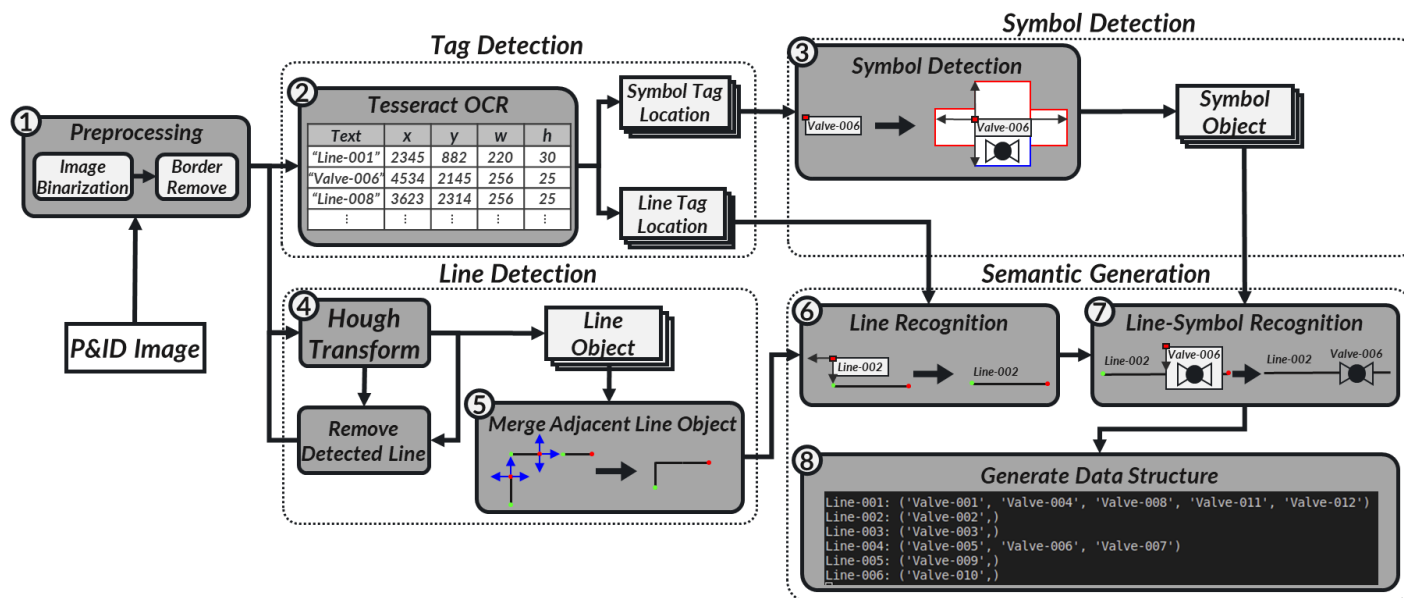
✓ 대우조선해양(DSME) 스마트야드 산학 연구과제 수행

- 디지털 기반 설계정보 공유 시스템 구축
- 선박, 해양플랜트 공정배관계장도(P&ID) 전산화
- 이미지 형식 P&ID 설계도 자동 전산화 기법 개발



✓ 이미지 형식 P&ID에서 텍스트 인식모델 기반 심볼 탐지기법

- 딥 러닝 기반 텍스트 인식모델을 사용하여 도면 내 텍스트 위치 탐지
- 탐지된 텍스트 위치를 기반으로 심볼객체, 라인객체 탐색
- 탐지된 객체들 간 픽셀 좌표 기반 인접성을 기준으로 연결관계 파악
- 밸브(심볼)와 파이프라인(라인)의 연결관계 정의



제안기법 동작과정

딥 러닝 기술을 사용한 DX연구

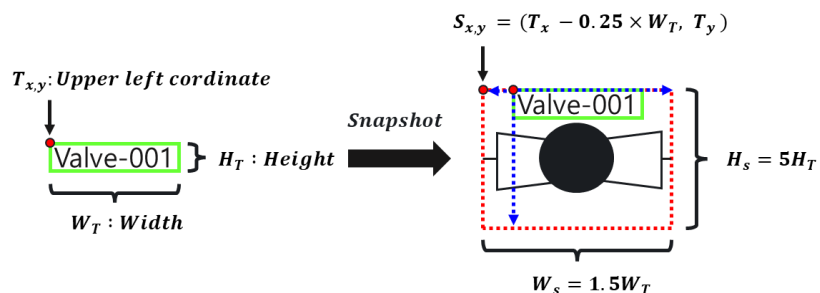
✓ 특징

- 기존 딥 러닝 기반 P&ID인식 연구는 심볼 인식을 위한 훈련데이터가 필요
→ 제안기법의 경우 정확도가 높고 일반화된 텍스트 탐지모델 사용, 훈련 데이터셋 구축 불필요
- 학습모델 훈련시간, 설계도면 처리시간이 오래 걸림
→ 훈련 불필요, 도면 당 평균 처리시간: 평균 7.04초

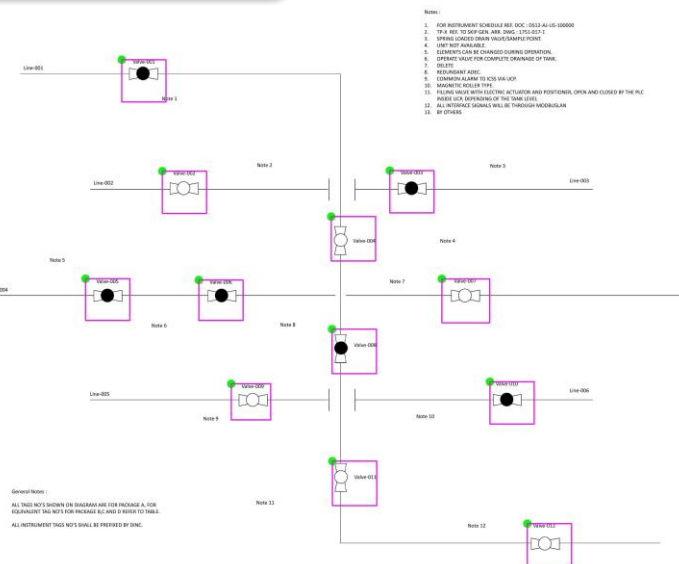
✓ 이미지형식 P&ID에서 텍스트 인식모델 기반 심볼 탐지기법

✓ 특허출원

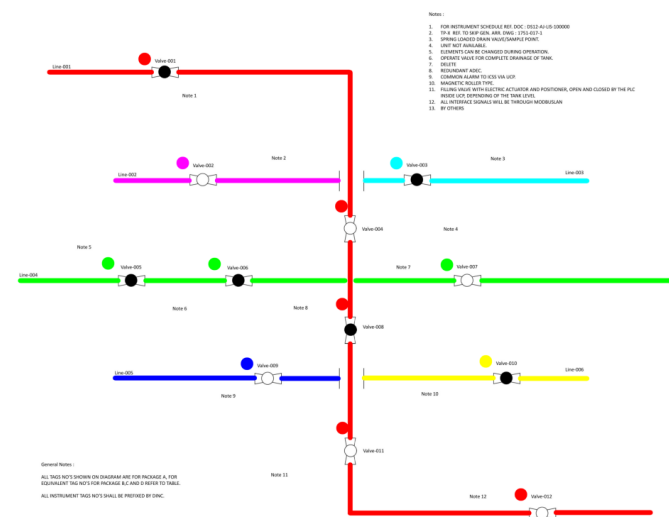
- 출원번호 10-2022-0122336, 2022.09.27



텍스트 위치기반 심볼 탐색 과정



심볼 탐지결과



심볼, 라인 탐지 및 관계성 분석결과



Thank you

Contact

Email : tjwjs513@gmail.com

Github : <https://github.com/JS-Choi513>

Blog : <https://js-choi513.github.io>

2023. 5. 18.

