



Researcher 배현성, 전자공학과 (bhs222@ajou.ac.kr)
김민규, 전자공학과 (minkyu9404@ajou.ac.kr)

Professor 선우명훈, 전자공학과

ABSTRACT

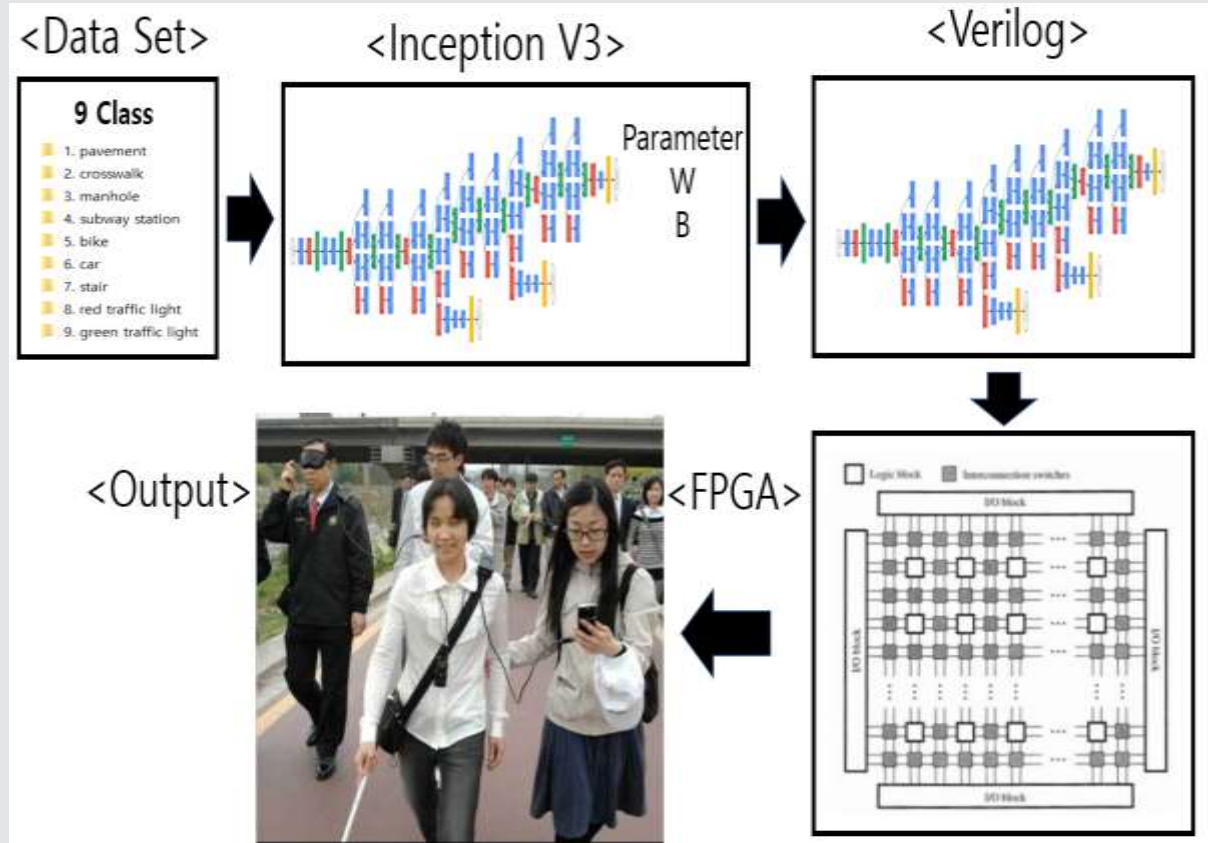
- 현재 맹안내견의 가격은 매우 비싼 편이다. 그래서 현재 시각장애인 중에 맹안내견을 사용하는 경우는 매우 드문 편이다. 이러한 시각장애인들의 애로사항을 해소해 주고자 딥러닝 기술을 이용하여 길을 안내해 줄 수 있는 시스템을 만들었다.
- 딥러닝 기술의 장점은 다른 기술보다 정확도가 높고 질이 좋은 데이터를 많이 모으면 모을수록 성능이 매우 증가한다는 점이다. 이러한 특징을 잘 살려서 제품을 만들게 된다면 신뢰도가 높은 제품을 만들 수가 있다.
- 제품의 경우 배터리 문제가 있을 수 있다. 전자기기 특성상 배터리 소모가 큰 편이므로 이를 해결하기 위해 보조배터리를 사용할 수 있도록 설계했다.

OBJECTIVES

- 시각장애인들은 맹인 안내견을 이용하기 보다는 지팡이나 지인들의 도움을 받아 이동하는 경우가 많으므로 이러한 분들의 도움을 주는 제품을 만드는 것이 목표이다.
- 두가지의 모듈을 3가지 방법으로 비교를 하고 질이 좋고 많은 Dataset을 통해 정확도가 거의 100%에 달하게 만들어 사람들의 신뢰를 많이 얻을 수 있는 제품을 만드는 것이 목표이다.

METHODOLOGY

1. System 구상도



- 1) Data Set - 총 9개의 Class로 각 2000장의 데이터를 수집
- 2) Inception V3, Vgg - 2가지 모듈을 각각 학습
- 3) Verilog - 모듈로부터 받은 Parameter를 통해 RTL 작성
- 4) FPGA - Verilog로 만든 RTL을 올려 실행

2. Training Model 비교

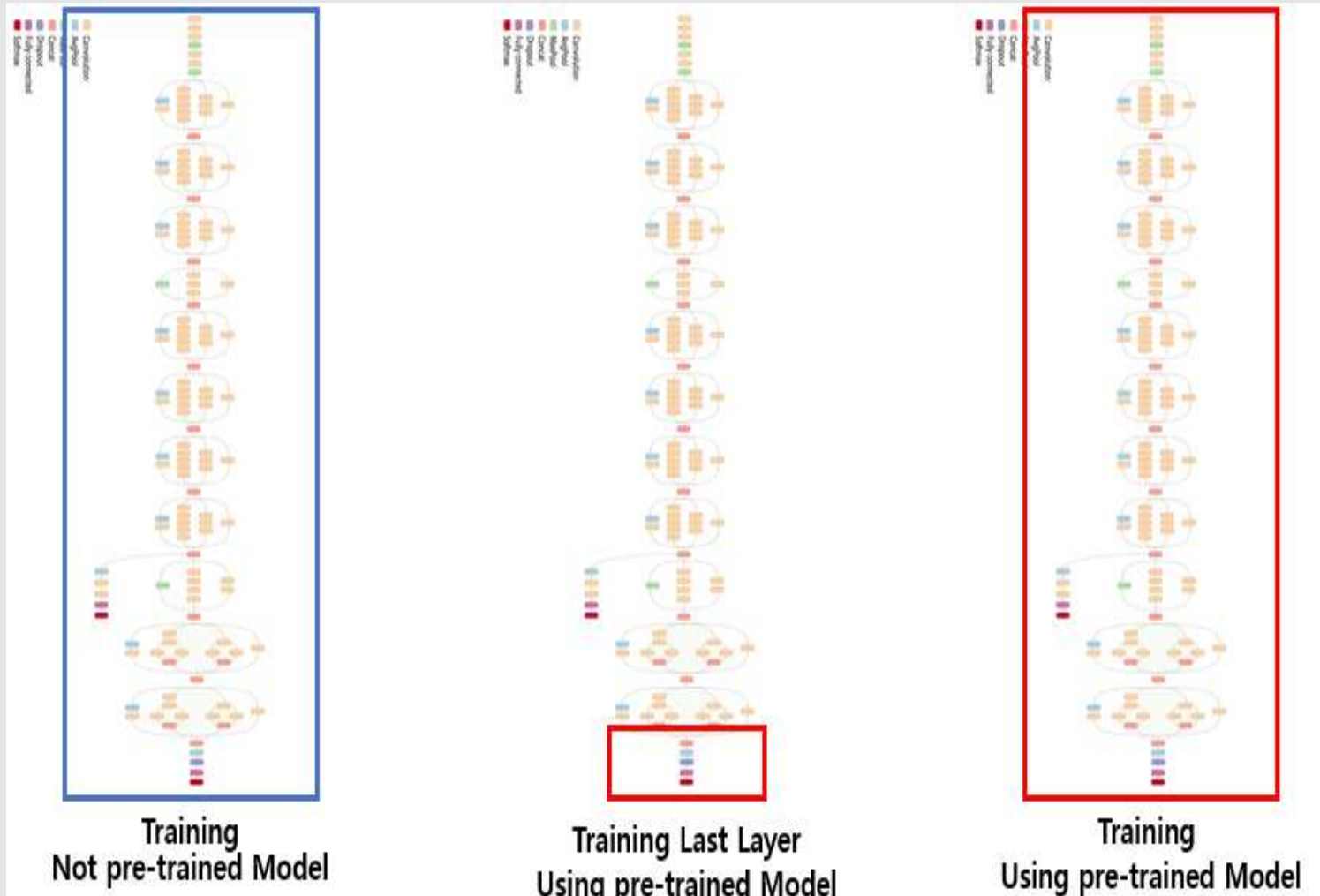


Fig 3. Three Learning Modes of Deep Learning

- (1) 모든 변수를 초기화 한 후 Train
 - 모든 변수를 초기화 한 만큼 Training을 하는데 더 오랜 시간이 걸리고 정확도도 많이 떨어진다.
- (2) 미리 Train된 변수를 사용하며 마지막 Layer 변수만 초기화 한 후 Train
 - Training 시간이 제일 짧고 (1)에 비해 정확도가 더 높게 나왔다.
- (3) 미리 Train된 변수를 사용하여 Train
 - Training 시간이 (1)보다는 짧지만 (2)보다는 크다. 하지만 정확도는 제일 높게 나왔다.

RESULTS

1. VGG Training



Fig 4. Three Learning modes of VGG

1. Not pre-trained mode - 21.7% 정확도
2. Training Last layer mode - 90.6% 정확도
3. Training pre trained mode - 92.3% 정확도

2. Googlenet Training



Fig 5. Three Learning modes of Googlenet

1. Not pre-trained mode - 85.6% 정확도
2. Training Last layer mode - 91.4% 정확도
3. Training pre trained mode - 97.6% 정확도

3. VGG 와 Googlenet의 Training 비교

- (1) Module의 구성 차이
 - VGG - 일반적인 CNN구조
 - Googlenet - Inception modul을 통한 더 Deep한 CNN구조
 - => 정확도 측면에서 Googlenet이 약 5% 높은 97.6% 기록
- (2) Parameter에 따른 연산량의 차이
 - VGG - 3X3 Conv만 사용하는 간단한 구조
 - Googlenet - VGG보다 복잡하지만 1X1 Conv를 통해 연산량 감소
 - => Googlenet이 복잡하지만 1X1 Conv Layer를 통해 연산량의 차이가 크지 않아 성능 개선

4. Googlenet - Test Chart

output input	pavement	crosswalk	manhole	subway station	bike	car	stair	red light	green light
pavement	98%	0.0003%	0.0003%	0.5%	0.0003%	0.0003%	0.7%	0.3%	0.2%
crosswalk	2%	96%	0.002%	0.0003%	1%	0.002%	0.0002%	0.0001%	0.0001%
manhole	0.0001%	0.0002%	99%	0.0002%	0.0003%	0.0002%	0.002%	0.0003%	0.0002%
subway station	0.0003%	0.0003%	0.0003%	99%	0.001%	0.0003%	0.002%	0.001%	0.0002%
bike	0.0002%	1%	0.004%	0.0005%	99%	0.0003%	0.0003%	0.0003%	0.0001%
car	0.001%	0.002%	0.0002%	0.0001%	0.0002%	99%	0.001%	0.004%	0.0003%
stair	0.004%	0.0003%	0.0002%	0.0005%	0.0002%	0.001%	99%	0.0002%	0.0003%
red light	0.0001%	0.0005%	0.0005%	0.0001%	0.0003%	0.0003%	0.0001%	97%	2%
green light	0.0001%	0.004%	0.0003%	0.0002%	0.0005%	0.0001%	0.0002%	2%	97%

Fig 6. Results of testing with Googlenet

CONCLUSIONS

- Convolution Neural Network 기술을 사용하여 현재 진행 방향에 위험이 있는지, 어떤 물체가 있는지 확인 할 수 있다.
- Googlnet, VGG를 둘 다 사용하여 정확도를 비교해본 결과 Googlenet이 VGG보다 대략 0.5% ~ 0.8% ej 높게 나왔다.
- Training 단계도 3단계로 나누어 비교를 해보았는데 미리 Training된 변수를 사용하는 것이 정확도가 제일 높았고 모든 변수를 초기화한 것이 정확도가 제일 낮았다.
- 향후 계획은 Training을 통해 얻은 변수들을 이용하여 Verilog를 통해 RTL Code를 작성한 후 FPGA에 옮겨 실제 SOC를 만드는 것을 목표 하고 있다.

- Class별로 각자 자신의 Data를 97~99%의 정확도가 나옴
- 자신의 Class가 아닌 Data는 약 0.003% ~ 2%정도의 오차가 있음
- 신호등의 경우 다른 Data에 비해 오차가 조금 더 큰 것을 확인