|  |  |
| --- | --- |
|  |  |
|  |  |
|  | **딥러닝을 이용한 한국 자생 곤충**의 **분류** |
|  |  |
| **Abstraction**  **현재 미국과 중국에서는 큰 데이터 셋을 이용한 곤충의 분류 및 탐지에 대한 연구가 진행되고 있다. 그러나 곤충의 분포는 환경에 따라 판이하며, 현재 대한민국안에서 발견되는 곤충에 대한 데이터셋은 잘 정리된 곳이 없고, 있다 하더라도, 그 양이 부족하다. 그러므로, 이 프로젝트에서는 미국과 중국에서 사용된 데이터셋을 이용하여, 그에 비해 양이 부족한 한국 곤충에 대한 데이터셋의 학습을 보완하여, 한국 곤충을 분류하는 신경망을 학습시키는 것이 목적이다.** | | |

**1. Introduction**

현재 미국과 중국에서는 큰 데이터 셋을 이용한 곤충의 분류 및 탐지에 대한 연구가 진행되고 있다**[1, 2]**. 그러나 곤충의 분포는 환경에 따라 판이하며, 현재 대한민국에서 발견되는 곤충에 대한 데이터셋은 잘 정리된 곳이 없고, 그 양이 부족하여, 대한민국에서 발견되는 곤충을 분류하는 신경망을 학습시키기 위한 크고 다양한 데이터를 구하는 것은 굉장히 어려운 일이다. 그러므로, 한국이 아닌 미국과 중국 등 외국에서 딥러닝을 위하여 준비한 데이터셋을 이용, 이에 비해 수가 부족한 대한민국 곤충의 데이터셋을 보완하 여, 13종의 대한민국 곤충을 분류하는 신경망을 학습시키고

자 한다.

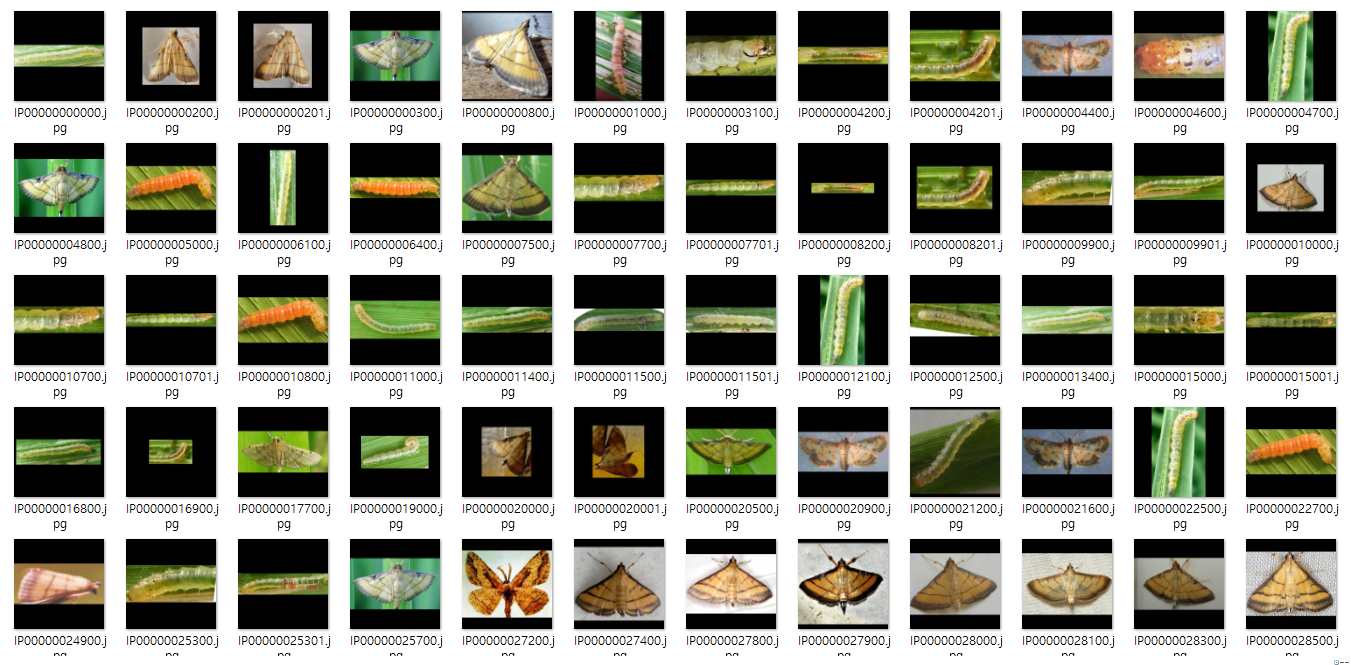
미국과 중국의 곤충으로 학습시킨 컨볼루션 신경망이 검출하는 이미지의 패턴이, 곤충이 가지는 패턴의 유사성에 기반하여 한국 곤충에 존재하는 패턴을 잘 검출할 수 있을 것이라 가정하고, 전이학습**[1, 2, 3]**의 방법을 통하여 한국의 수가 부족한 데이터셋을 외국의 데이터 셋을 이용하여 보완하고자 한다. 또한 추가적인 방법으로 앙상블 학습을 통하여 네트워크의 구성, 학습하여 성능을 향상시키

고자 한다.

**2. Methods**

**(1) 데이터 셋**

IP102 데이터 셋**[1]**은 102가지 class의 벌레로 구성된 18,975장의 데이터 셋이다. ip102 데이터 셋은 class간 불균형이 매우 심하므로, 최소 40장 이상의 사진이 있는 74종류의 class만을, 이미지넷으로 선훈련시킨 CNN을 전이학습하는데 사용하였다. 한 사진 내에, 여러 object가 있는 경우에는 분리하여 데이터의 양을 늘였고, 총 74가지 class에 대하여 19,820장의 데이터를 사용하였다. IP102 데이터 셋의 class는 그 곤충의 유충, 성충 단계의 구별없이 같은 class로 구성되어있다. ip102 데이터 셋으로 전이학습 시킨 신경망을 다시 한국 곤충 데이터를 이용하여 전이학습 할 예정이므로, 학습에 더 큰 비중을 두어 training data : validation data : test data의 비율을 8:1:1로 사용하였다.



한국 곤충 데이터 셋은 검색엔진을 통하여 한국 곤충명과 학명을 통하여 데이터를 검색하였고, 전문가가 분류한 사진과 그 분류 기준을 이용하여 전체 데이터의 class를 분류하였다. 검색엔진을 통하여 총 1983장의 데이터 셋을 확보하였다. 한국 곤충 데이터 셋의 크기가 작기 때문에, validation 및 test data set의 크기를 최대로 확보하기 위하여, training data : validation data : test data의 비율을 5:2.5:2.5로하여 사용하였다. 한국 곤충 데이터 셋은 총 13가지 class로 구성되어 있으며, 총 각 class의 label은 가루깍지벌레, 꽃노랑총채벌레, 네눈쑥가지나방 유충, 가루

이, 담배거세미나방유충, 담배나방유충, 배추좀나방 유충, 벼룩잎벌레, 복숭아순나방 유충, 사과응애, 아메리카잎굴

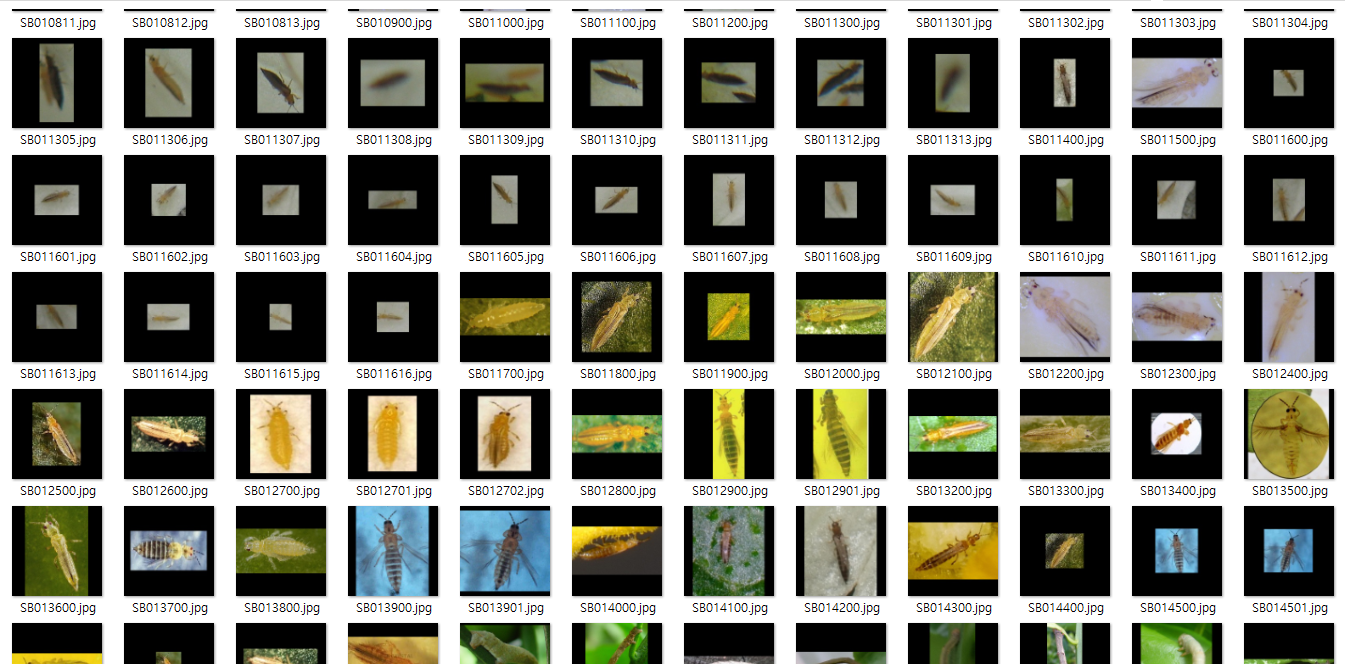
파리, 점박이응애, 파밤나방 유충이다. 아래의 사진은 각

class의 대표적인 이미지를 위의 나열한 순서와 일치하게 나열한 사진이다.



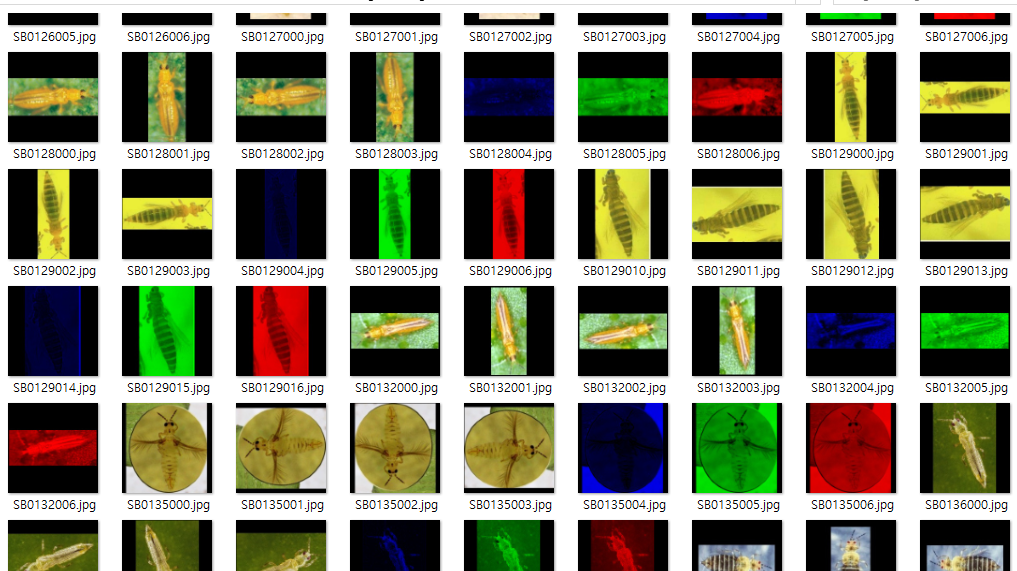


아래의 사진은 고정 크기의 이미지를 입력으로 받는 VGG16과 ResNet의 특성으로 인하여, resize후 여백을 zero padding한 한국 곤충 데이터 셋이다.



현재 한국 곤충데이터 셋의 양이 굉장히 부족하기 때문에, 5:2.5:2.5의 비율을 사용하는 경우, 학습에 사용되는 데이터셋의 양이 1000장도 되지 않는 문제가 발생한다. 이를 해결하기 위하여 data augmentation**[2, 3]**을 사용하

여 데이터를 7배로 늘렸다. 사용한 data augmentation은 회전과, 컬러채널을 하나씩 사용하고, 나머지 채널은 zero padding을 하는 방법을 사용하였다.



Data augmentation은 training 데이터, validation 데이터, 및 test 데이터를 분리한 후에, training 데이터에만 적용하

여 양을 하였다. 총 13개 class 960장 가량의 사진에 대하여

, data augmentation을 적용하여 총, 6783장의 데이터를 tr

aining에 사용하였다.

**(2) 사용 모델 및 학습 방법**

이번 논문에서는 새로 CNN을 만들어 학습하는 대신에 여

러 논문을 통하여 그 성능이 검증된 VGG16**[4]**, ResNet**[7**

**]** 두 가지 모델을 사용하였다.

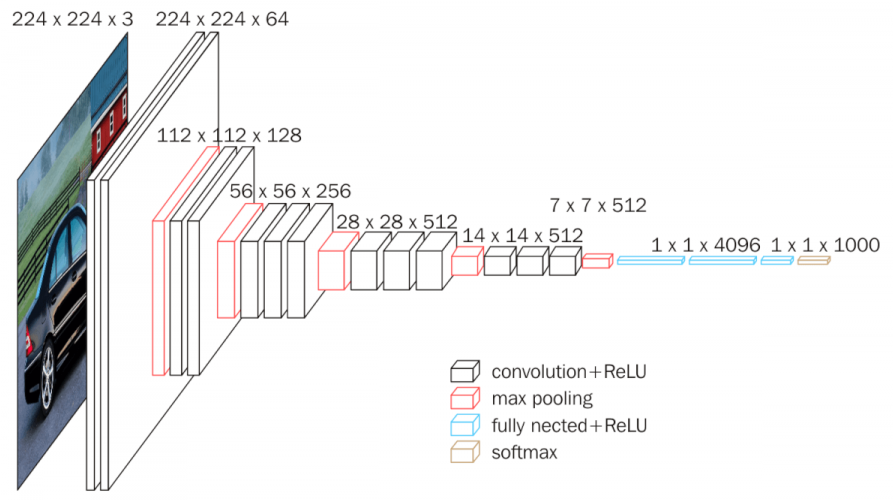
또한, Scratch부터 학습하는 것이 아니라, 다양하고 양이 많은 Image Net 전이학습과 앙상블 학습을 통하여, CNN의 성능을 향상시키고자 하였다.

**① VGG16**

VGG16 네트워크의 구조에서 soft-max층의 바로 직전의 층만 IP102와 한국 곤충 데이터셋의 class의 숫자에 맞추어 구조를 바꾸고, 그 이외에는 구조의 변경 없이 사용하였다.

VGG16 네트워크는 input으로 224\*224\*3의 고정된 크기 의 이미지를 입력으로 받기 때문에, 이미지의 크기를 조정하고, 원본이미지의 형태로 인하여 생기는 여백은 zero

padding하였고, 이는 Figure 1, 2, 3에서도 확인할 수 있다.





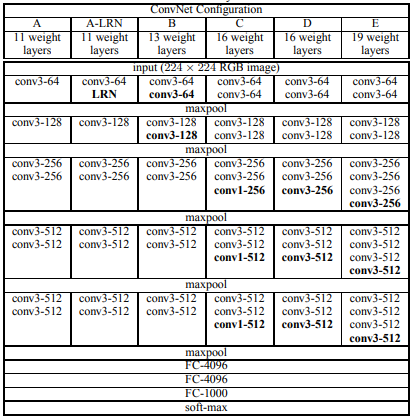
Loss function으로는 cross entropy 함수를 사용하였고, optimizer로는 Adam optimizer**[6]**를 사용하였다. IP102와 한국 곤충 데이터 셋 모두, 클래스간 불균형이 심하기 때문

에, 오버샘플링을 이용하여 클래스간 크기의 균형을 맞추어

학습하였다. Learning rate 0.0001, ip102 데이터를 학습할 때는 batch size를 128, 한국 곤충 데이터를 학습할 때는 batch size를 32로 하여 학습하였다. 완전연결층에는 drop

out layer를 추가하였고, dropout ratio로 0.5를 사용하였다. 오버샘플링의 비율은 모든 class가 가장 수가 많은 클래스의 data의 70%가 되도록 하였다.







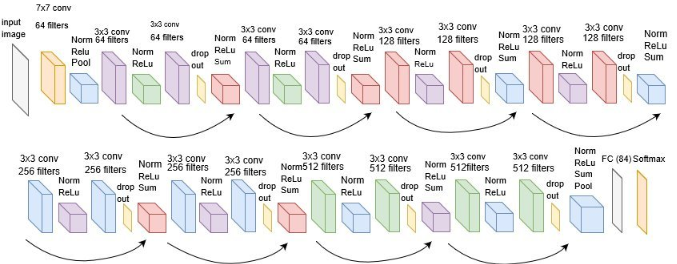
**② 18-layer ResNet**

ResNet은 building block을 16개 사용하는 총 18-layer ResNet을 사용하였다. building block은 residual shortcut

을 building block의 두 번째 활성화 함수를 통과하기 전에

summation되는 v1 building block을 사용하였다**[7]**.

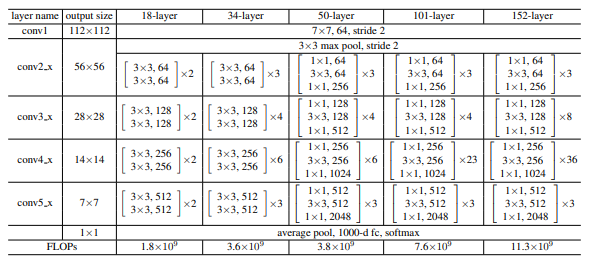
Batch normalization layer는 building block의 2개의 Relu 활성화 함수의 앞에 두어 네트워크를 구성하였다.



Loss function으로는 VGG16 network와 마찬가지로 cross entropy 함수를 사용하였고, optimizer로 Adam optimizer를 사용하였다. Learning rate, batch size 및 oversampling 비율과 같은 하이퍼 파라미터는 VGG16과 똑같은 수치를 사용하였다. 실험과정에서 18-layer ResNet

의 학습과정에서 갱신되어야 할 이동 평균과 이동 분산**[6]**

은 image net으로 학습시킨 resnet 18의 값을 고정하여 그대로 사용하였고, batch normalization layer에서는 beta와 gamma만 학습하였다**[6]**.







**③ 전이학습**

전이학습에는 컨볼루션 레이어의 가중치를 고정한 후에, 가장 끝단의 완전연결층의 가중치만을 초기화하고 학습하는 방법과 정상적으로 맨 아래 층부터 끝 층까지의 네트워크의 모든 층을 학습하는 두 가지 방법이 있다.

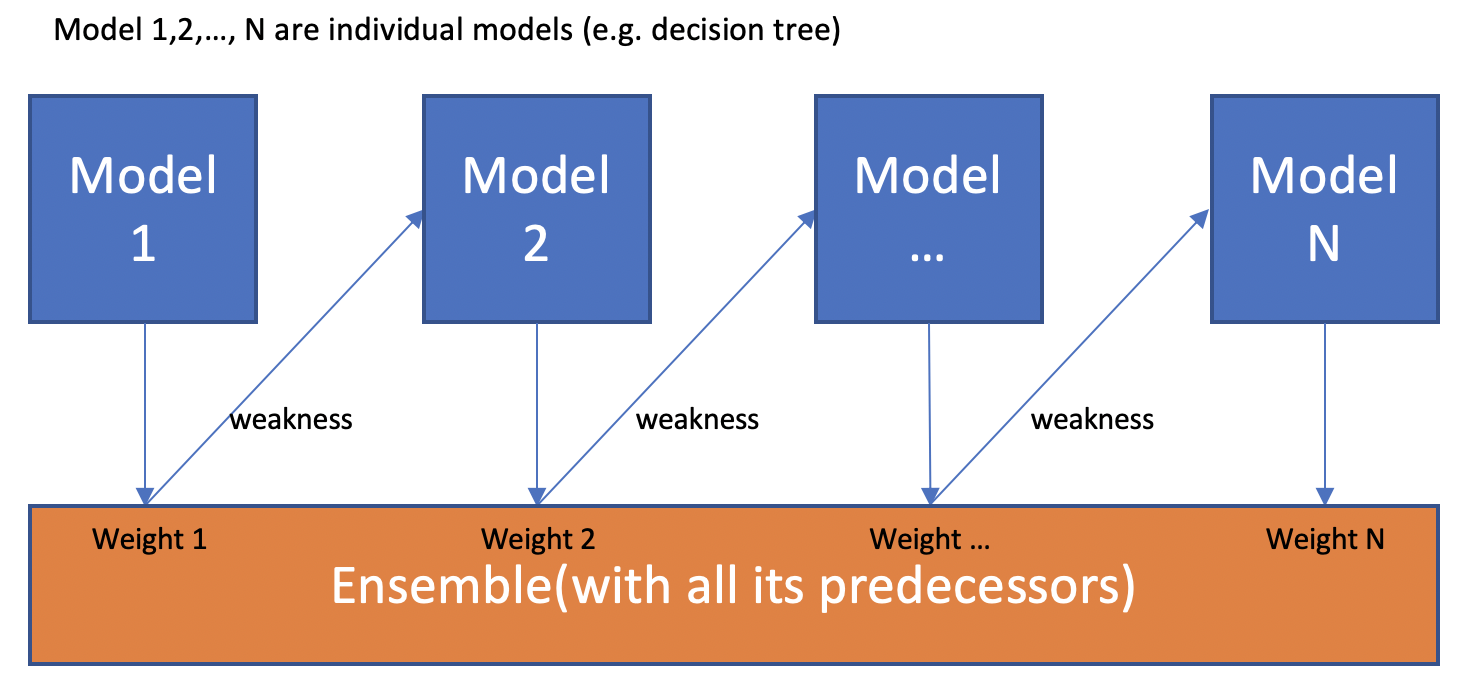
이번 논문에서는 IP102와 한국 곤충 데이터셋 모두 CNN의 가장 아래 층 부터 가장 위 층의 완전연결층을 모두 학습하

는 전이학습을 실험하였다.**[1, 2, 3]** 이는, IP102 데이터를 컨볼루션 레이어에 학습시켜, 벌레의 패턴을 찾는 데 특화된 CNN을 만드는 것을 의도하여 이렇게 실험을 진행하였다.

IP102의 경우, 컨볼루션 레이어와 batch normalization layer의 가중치들을 Image net으로 학습시킨 VGG16, 18-layer resnet의 가중치로 초기화한 후에 학습**[1, 2, 3]**하였으며, 한국 곤충데이터의 경우, 위의 IP102데이터로 학습시켜 얻은 가중치를 초기 가중치로 설정하여, 위와 같은 방식으로 학습을 진행하였다.

**④ 앙상블 학습**

앙상블학습은 전체 training data set에서 독립적으로 랜덤추출한 beg을 생성하여, 그 beg마다 독립적인 모델을 학습시킨 후, 각 모델의 inference를 투표 방식, 평균, 또는 합과 같은 방식으로 정리하여 전체 inference를 결정하는 begging방식과 모델을 순차적으로 학습하며, 그 모델의 약점을 보완하는 방식으로 beg을 구성하여 모델을 학습시킨 후, inference를 시행하는 boosting방법이 있다. 이 논문에서는 boosting 방식을 사용하여 ensemble model을 학습시켰다.



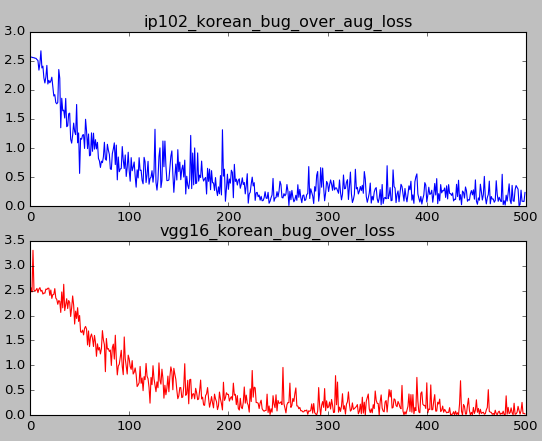
Boosting 기법을 2개의 model에 사용하여 앙상블학습을 진행하였다. 모델의 조합으로 (18-layer ResNet, 18-layer

ResNet), (VGG16, 18-layer ResNet), (18-layer ResNet, VGG16)의 3가지 모델 조합을 사용하였다. 이 때, model1은 ip102데이터로 전이학습한 CNN을 다시 한국 곤충 데이터로 전이학습한 모델을 사용하였으며, 학습은 model2에서만 진행하였다. model2는 model1이 답을 맞추지 못하거나, 답을 맞춘 경우에도, class score가 thresh hold이하인 경우에는 그 data의 경우에는 가중치를 두어 학습하였다. 가중치는 50배를 사용하였고, beg의 size는 전체 데이터의 60% 크기로 사용하였다. 추론 과정에서는 model1이 test data에 대하여 class score가 thresh hold미만일 경우에, 그 데이터는 second model로 inference하여 그 값을 최종 inference로 사용하였다.

**3. Data & Results**

**(1) VGG16**

ImageNet으로 학습한 네트워크에 한국 곤충 데이터 셋으로 학습하는 경우와, IP102 데이터 셋으로 학습한 네트워크를 한국 곤충 데이터로 학습하는 경우의 차이가 loss의 수렴 속도가 더 빠르다는 점 외에는 정확도에서 확인할 수 있듯이 큰 차이가 발생하지 않았다. 또한, 두 학습 설정의 경우 모두 loss의 감소속도가 굉장히 빠르다.

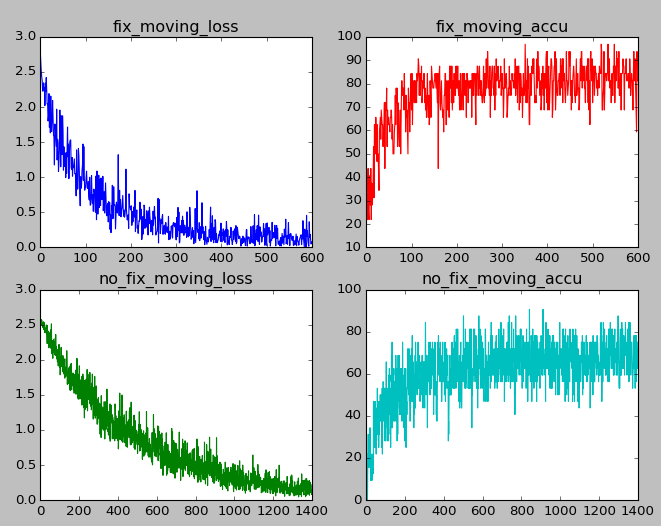


|  |  |
| --- | --- |
| 학습 상황 | 정확도 |
| IP102 선 훈련, 오버샘플링, augmentation | 80.24% |
| 이미지넷 선 훈련, 오버샘플링 | 80.46% |

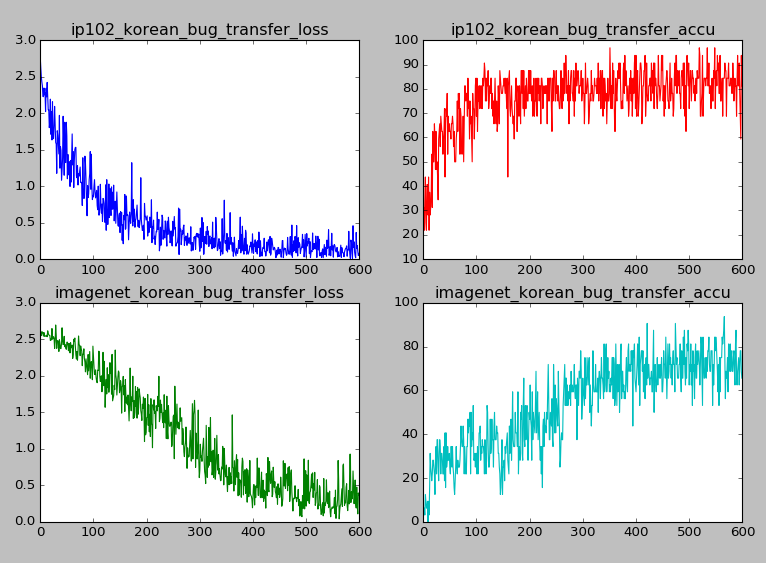


**(2) RES18**

Batch normalization layer에서 이동 평균과 이동 분산을 고정하는 경우와, 이미지넷에서 바로 한국 곤충 데이터를 훈련 시키는 경우와, ip102로 선 훈련 후, 한국 곤충 데이터를 훈련 시키는 3 가지 학습설정에서 loss의 수렴속도 및 정확도에서 큰 차이가 나타났다.







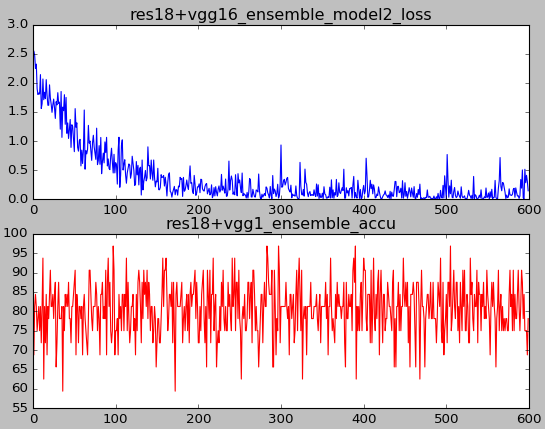


|  |  |
| --- | --- |
| 학습 상황 | 정확도 |
| IP102 선 훈련, 오버샘플링, augmentation, no fix | 69.14% |
| Imagenet 선 훈련, 오버샘플링, augmentation, fix | 73.33% |
| IP102 선 훈련, 오버샘플링, augmentation, fix | 79.31% |

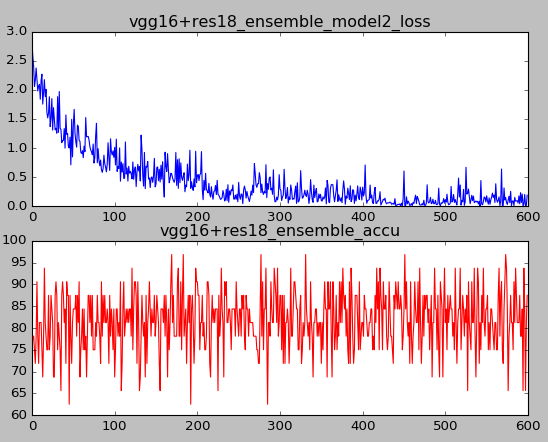


**(3)앙상블 학습**

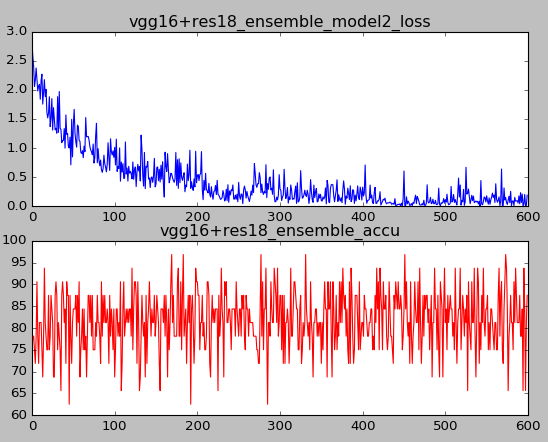
앙상블 학습의 경우, 모든 모델 조합에서 정확도가 약 2~3%씩 상승하였다. 상승 폭은 조합에 영향을 받지 않는 것으로 확인되었다. 두 번째 모델의 loss의 수렴속도는 단독 모델을 학습하는 것에 비하여, 육안으로 확인가능한 차이는 존재하지 않는 것으로 확인되었다.











|  |  |
| --- | --- |
| 학습 상황 | 정확도 |
| res18+vgg16에서 res18 단독 | 79.69% |
| res18+vgg16에서 앙상블 학습 후 | 82.03% |
| vgg16+res18에서 vgg16 단독 | 80.46% |
| vgg16+res18에서 앙상블 학습 후 | 83.01% |
| res18+res18에서 res18 단독 | 79.69% |
| res18+res18에서 앙상블 학습 후 | 82.03% |



**4. Conclusion & Discussion**

먼저 Instruction에서의 가정대로, 중국과 미국의 곤충 데이터 셋인 IP102로 선 훈련시킨 네트워크를 한국데이터로 전이학습하는 것이 성능 향상에 효과가 있음을 VGG16과 18-layer ResNet의 수렴속도의 차이와, 정확도의 차이에서 확인할 수 있다. 즉, 한국 곤충의 패턴은 외국의 곤충의 데이터로 학습한 CNN도 잘 포착할 수 있으며, 한국의 곤충 데이터가 부족한 경우에 외국의 데이터를 이용하여, 한국 곤충의 데이터의 부족을 보완이 가능함을 알 수 있다.

다만, VGG16 에서는 loss의 수렴속도에는 IP102로 선 훈련한 데이터에 한국 곤충 데이터를 학습시키는 것이 효과가 있으나, 정확도에는 큰 향상이 없다. 이는 ResNet에 비하여 복잡도가 큰 VGG16 네트워크가 한국 곤충 데이터만 사용하여도 이미지에서 곤충의 패턴을 잘 찾고 있는 것으로 추정된다. 다만 Table1에서 확인할 수 있듯이, parameter

의 수와 그로 인한 네트워크의 복잡도가 커, Figure 9와 같이 overfitting이 발생할 수 있는 loss까지 굉장히 빨리 도달함을 알 수 있다. 그렇기 때문에, 현재 데이터의 수가 부족한 한국 곤충의 특성상, 한국 곤충의 분류에 사용할 네트워크는 18-layer ResNet이 VGG16 네트워크 보다 더 적합하다고 생각된다.

18-layer ResNet에서는 IP102 데이터를 이용하여 선 훈련한 후에 한국 곤충 데이터를 학습시키는 경우가 ImageNet에서 바로 한국 곤충 데이터를 학습시키는 경우보다 5% 이상 상승하였는데, 이는 Instruction에서의 가정이 유의미함을 의미하며, 더 좋은 외국 곤충 데이터 셋을 이용하여 선 훈련할 경우에, 한국 곤충 데이터가 부족하더라도 충분히 이를 통하여 보완이 가능함을 시사한다.

또한, 18-layer ResNetmoving mean과 moving var를 고정하는 경우에, 그렇지 않은 경우보다 정확도의 향상이 두드러지는데, 이는 곤충 데이터에 비하여 다양하고 크기가 큰 ImageNet의 학습과정에서 구한, moving mean과 moving var가 이미지의 모평균과 모분산을 더 잘 대표하기 때문인 것으로 추정되며, 이를 크기가 작은 데이터셋으로 다시 재학습 하는 것은 성능에 큰 악영향을 주는 것이 확인되었다.

또한, 앙상블 학습을 통하여, 정확도를 상승이 확인되었지

만, 그 정확도의 차이가 모델의 조합에는 큰 영향을 받지 않음을 확인할 수 있다. 즉, 앙상블 학습에서 사용하는 모델보다는 Boosting 학습 단계에서 어떤 조건을 사용하여, 앞의 모델의 약점을 보완하는 beg을 구성할 지, 또한 그 하이퍼 파라미터를 더 잘 조절하는 것이 앙상블 학습으로 인한 정확도의 향상에 효과가 있음을 추정할 수 있고, 더 다양한 하이퍼 파라미터의 조합에 대한 실험이 필요함을 보여준다.

IP102 데이터 셋과 한국 곤충 데이터셋의 개선을 통하여, 네트워크의 성능을 향상시킬 수 있음이 예상된다.



먼저 IP102 데이터 셋의 경우에, 한국 곤충의 전이학습을

위하여 이미지넷으로 훈련하는 과정에서 Figure15처럼 학습률과 반복횟수 등의 하이퍼 파라미터를 수정하여도 ip102의 분류 정확도가 약 60%에서 증가하지 않는다. 이 정체가 데이터의 부족으로 인한 것으로 추정되나, 곤충의 분류라는 특성상 다른 해석의 가능성이 존재한다.

먼저, IP102 데이터 셋은 특정 곤충의 class가 유충, 성충 단계를 구분하지않고 구성되어있다. 그러나 곤충의 특성상 유충, 성충 단계에서 큰 외양의 차이가 존재하므로, class가 곤충의 종인 경우에 패턴을 학습하는 CNN의 특성상 같은 class가 완전히 다른 패턴으로 구성되어있는 경우, 더 많은 데이터가 필요하는 등의 학습의 어려움이 클 것으로 추정된다.

또한, 유충간에도 형태가 판이하게 다른 경우가 있기 때문에, ip102의 데이터에 대하여 특정 곤충의 각 단계에 대해서 다시 class를 분류하거나 또는 한국곤충의 분류를 위한 거쳐가는 과정이므로, IP102 데이터 셋을 곤충의 종류가 아니라, 형태적 유사함을 기반으로한 class의 재분

류후에 이를 기반으로 한 전이학습을 시도할 경우에 성능의 향상이 가능할 것으로 추정된다.

그리고 한국 곤충데이터의 양이 부족하다는 명확한 한계가 존재하므로, 먼저 데이터를 더 확충하는 것이 추후 연구를 위해서는 우선적으로 필요한 일로 예상된다. 또한 이번 논문에서는 검색엔진을 통하여 얻은 데이터를, 일부의 분류된 데이터의 기준으로 분류하여 실험을 진행하였기 때문에, 데이터의 분류에서 전문가의 감독 하에 분류를 진행하여 데이터셋을 준비하는 것이 더 정확한 네트워크를 만들기 위한 필수적인 과정으로 생각된다.**[1]**

이번 논문에서는 Figure 4에서 확인할 수 있듯이 굉장히 기본적인 data augmentation을 진행하였다. 이에 추가하여 Salt and Papper Noise나 더 다양한 회전 등**[3]**의 추가적인 data augmentation을 사용하면 네트워크의 성능을 더 향상시킬 수 있을 것으로 추정된다.

또한 공통적으로, 해충의 경우 사람이 육안으로 구분하기가 매우 어렵기 때문에, 벌레를 발견한 곳과 같은 부가적인 정보를 통하여 구분한다는 점에 착안하여, 발견 장소를 사진의 배경의 패턴으로 생각한다면, 노이즈를 제거하기 위하여**[3]** 좌측의 통제된 환경에서의 사진 데이터 셋보다 우측의 발견된 장소에서의 촬영한 사진 데이터가 학습에 더 도움이 될 수 도 있는 가능성이 있으므로, 이러한 데이터 셋에 대한 추가적인 고려도 추후의 곤충의 분류에 위한 네트워크를 학습할 때 고려대상이 될 수 있다.



**Reference**

[1] IP102: A Large-Scale Benchmark Dataset for Insect Pest Recognition, Xiaoping Wu, Chi Zhan, Yu-Kun Lai, Ming-Ming Cheng, Jufeng Yang, *CVPR*, 2019.

[2] Insect Detection and Classification Based, on an Improved Convolutional Neural Network, Denan Xia, Peng Chen, Bing Wang, Jun Zhang, and Chengjun Xience, *Sensors(Basel),* 2018.

[3] Using Deep Learning for Image-Based Plant Disease Detection

Sharada P.Mohanty, DavidP.Hughes, and MarcelSalath, *Front. Plant Sci,* 2016.

[4] Very Deep Convolutional Networks For Large-Scale Image Recognition, Karen Simonyan & Andrew Zisserman, *CVPR*, 2014.

[5] Adam: A method for stochastic optimizer, diederik. P. Kingma, Jimmy Lei Ba, *arXiv:1412,6980*, 2014.

[6] Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, Sergey Ioffe, Christian Szegedy, [*arXiv*](https://en.wikipedia.org/wiki/ArXiv)*:*[*1502.03167*](https://arxiv.org/abs/1502.03167), 2015.

[7] Deep Residual Learning for Image Recognition, [Kaiming He](https://arxiv.org/search/cs?searchtype=author&query=He%2C+K), [Xiangyu Zhang](https://arxiv.org/search/cs?searchtype=author&query=Zhang%2C+X), [Shaoqing Ren](https://arxiv.org/search/cs?searchtype=author&query=Ren%2C+S), [Jian Sun](https://arxiv.org/search/cs?searchtype=author&query=Sun%2C+J), *CVPR*, 2015.