4 장: 컨볼루션 신경망

4.1 CNN 구조

컨볼루션 신경망(Convolutional Neural Networks, CNN)은 영상처리를 위해 고안된 특수한 연결구조를 가진 다층신경망이다(LeCun & Bengio, 1995). 일반적인 다층신경망에서는 인접한 두 개의 층간의 뉴런들이 모두 연결된 완전 연결 구조를 갖는다. 이러한 완전 연결 구조를 이용하면 이론적으로 입력 영상에 있는 위상학적 구조를 찾을 수는 있으나 현실적으로는 입력 영상이 다차원으로 이루어져 있기 때문에 문제 복잡도가 매우 크다는 문제가 있다. CNN 의 핵심 아이디어는 영상의 위상학적 구조를 학습할 수 있는 신경망의 구조를 새롭게 설계한 것이다. 이러한 설계는 픽셀 영상으로부터 최소한의 전처리를 통해서 다양한 영상 인식에 사용할 수 있다.

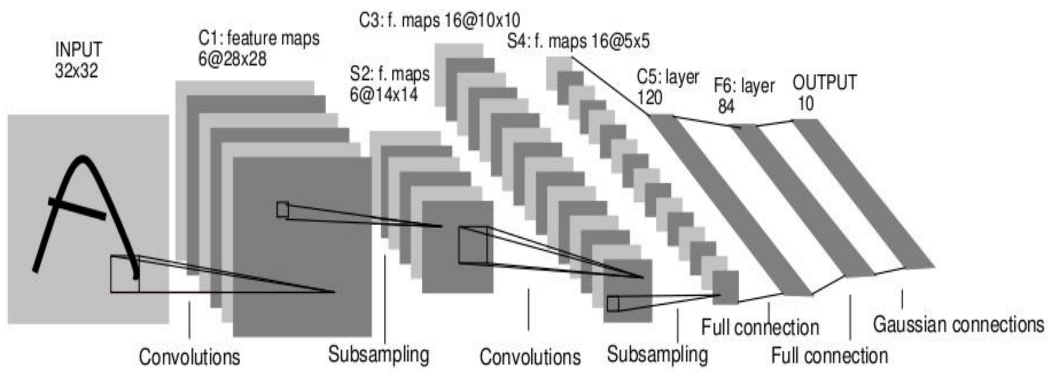


그림 4.1 컨볼루션 신경망(CNN)의 구조

CNN 은 기존의 패턴 인식 방식에서 특징 추출과 분류의 두 단계를 하나의 단계로 합친 통합한 모델로 볼 수 있다. CNN 은 기본적으로 다수의 컨볼루션 층(convolution layer)과 서브샘플링 층(subsampling layer)으로 구성되어 있다. 그림 4.1 은 “A”자의 문자 이미지를 인식하는 CNN 의 구조를 보여준다.

그림 4.2 는 컨볼루션과 서브샘플링의 핵심 아이디어를 기술하고 있다. 컨볼루션층은 입력에 대해서 다양한 컨볼루션 커널을 적용하여 특징맵(feature map)을 생성한다. 컨볼루션은 고차원의 입력 영상에 대해서 특징을 추출하는 일종의 템플릿 역할을 한다. 하나의 컨볼루션은 부분 입력을 대상으로 하며 위치를 바꾸어 가며 여러번 반복적으로 적용되어 전체 이미지를 스캔한다. 즉 하나의 컨볼루션은 적은 수의 연결선을 가지며 위치가 바뀌어도 그 연결 가중치가 공유된다(weight sharing). 이는 컨볼루션이 위치에 무관하게 특징을 추출할 수 있도록 해준다. 하나의 컨볼루션에 대해서는 위치가 바뀌어도 가중치를 공유하는 것은 학습할 전체 파라미터의 수를 줄여줌으로써 문제의 복잡도를 줄여주는 핵심 아이디어 중 하나이다. 서브샘플링(subsampling) 층은 만들어진 특징맵에 대해서 공간적 해상도를 줄인 뉴런층이다. 서브샘플링은 차원을 축소하는 과정이며 이를 통해서 역시 문제의 복잡도를 줄일 수 있다. 서브샘플링 연산자로는 대상 뉴런들에 대해서 최대치를 취하는 맥스풀링(max pooling) 연산자나 평균치를 취하는 평균풀링(average pooling) 연산자를 주로 사용한다. 풀링을 통해서 차원을 축소할 뿐만 아니라 특징맵이 이동(shift)과 왜곡(distortion)에 대해 강인하도록 하는 효과가 있다.

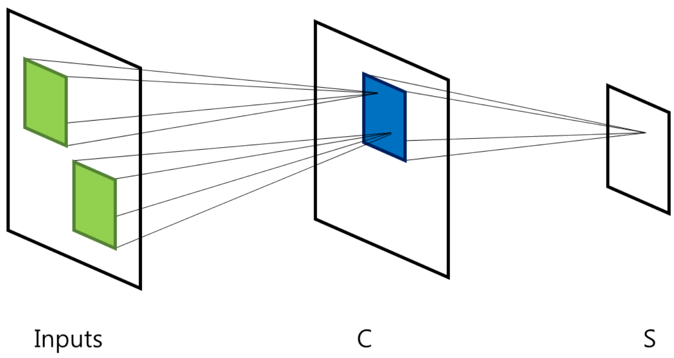


Figure 4.2 컨볼루션(C)과 서브샘플링(S)

그림 4.3 은 컨볼루션의 아이디어를 설명하고 있다. 완전연결층, 지역연결층과 비교하여 컨볼루션층이 어떻게 다른지를 설명한다. 완전연결층은 이웃한 층의 뉴런간에 연결선이 존재하는데, 하나의 층에 있는 모든 뉴런과 이웃한 층에 있는 모든 뉴런이 연결된다. 즉, 두 층의 뉴런의 수가 각각 5 개라고 하면 두 층간에는 총 5 x 5 = 25 개의 연결선이 존재하고 25 개의 가중치가 존재한다. 이에 반해서 지역연결층을 사용하면 상위층의 뉴런이 하부층에 있는 일부 뉴런에만 연결된다(가운데 그림). 가운데 그림을 보면, 상위층에 있는 5 개의 뉴런은 아래층에 있는 2~3 개의 뉴런과만 연결이 되어 있고, 각각 2 + 3+ 3 + 2 = 13 개의 연결선을 가진다. 각 뉴런의 연결선들은 서로 다른 가중치를 갖는다. 컨볼루션층은(오른쪽 그림) 부분연결층과 같이 부분적인 연결을 가지는 것은 같으나, 상위층 뉴런들이 서로 가중치를 공유한다. 즉, 각 상위 층에 있는 뉴런이 아래층에 있는 왼쪽, 중앙, 오른쪽 세개의 뉴런과 연결된다고 하면, 상위층에 있는 5 개의 뉴런은 이 세 개의 가중치를 공유한다. 따라서 연결선은 13 개임에도 불구하고 가중치는 3 개이다. 이렇게 함으로써 학습해야 할 파라미터의 수를 줄인다. 이러한 연결 구조를 이용하면 위치는 다르지만 비슷한 특징을 가지는 부분을 추출할 수 있기 때문에, 입력 이미지에서 물체가 좌우로 이동하는 것에 무관하게 특징을 추출할 수 있다.

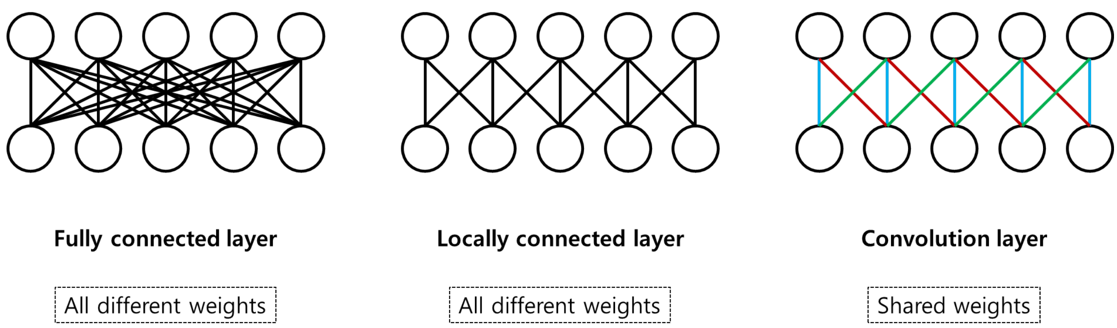


Figure 4.3 완전연결층, 지역연결층, 컨볼루션층의 비교

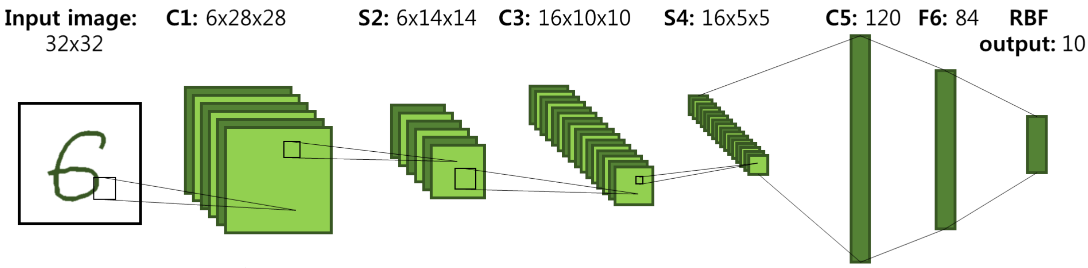


Figure 4.4 LeNet5 의 구조

구체적인 예로서 LeNet5 의 구조를 살펴보자(그림 4.4). 이 모델은 32 x 32 픽셀 이미지를 입력으로 받아서 세번의 컨볼류션과 두번의 서브샘플링을 거친 후 최종적으로 완전연결된 다층망을 통해서 출력을 결정한다. 컨볼루션은 5 x 5 매트릭스를 사용하며 서브샘플링은 $\frac{1}{2}$로 하였다. 전체 구조는 다음과 같이 기술될 수 있는데 여기서 Cn 은 n 번째 컨볼류션 층을 나타내며 Sm 은 m 번째 서브샘플링 층을 나타낸다.

입력 영상: 32 x 32
C1: 6 x 28 x 28
S2: 6 x 14 x 14
C3: 16 x 10 x 10
S4: 16 x 5 x 5
C5: 120
F6: 84
출력 뉴런 수(RBF): 10

- 32 x 32 의 이미지로부터 5 x 5 의 컨볼루션 매트릭스를 사용해서 28 x 28 특징맵 6 개를 구성한다(C1).
- C1 에 있는 각각의 28 x 28 의 맵으로부터 $\frac{1}{2}$ 서브샘플링을 사용하여 14 x 14 서브샘플링 맵 6 개를 구성한다(S2).
- 다시 5 x 5 의 컨볼루션 매트릭스를 사용해서 10 x 10 특징맵 16 개를 구성한 후(C3)
- 다시$\frac{1}{2}$ 서브샘플링을 사용하여 5 x 5 서브샘플링 맵 16 개를 구성한다(S4).
- 그 결과로부터 120 개의 컨볼루션을 구성하여(C5)
- 완전연결망 층구조를 이용하여 84 개의 서브샘플링 후(F6) 최종으로 10 개의 방사기저함수(RBF) 뉴런을 통해 출력을 계산한다.

F6 층 전의 모든 뉴런들은 시그모이드 활성화 함수를 가지며 다음으로 정의된다.

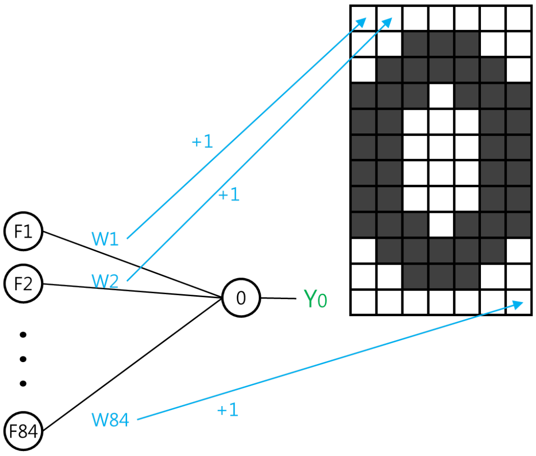
$$f_i = f(v_i) = A\tanh(Sv_i)$$

여기서 S와 A는 활성화함수의 모양을 결정하는 파라미터로 보통은 상수를 사용한다. 마지막 층에서 CNN 의 출력값은 다음과 같이 출력 특징지도와 영상지도가 정해진다.

$$y_j = \sum_{i=1}^{84}(f_i - w_{ij})^2,\ j = 0,...,9$$

여기서 f_i는 출력 영상 특징 값들이고 w_{ij}는 목표 영상 특징 값들이다.

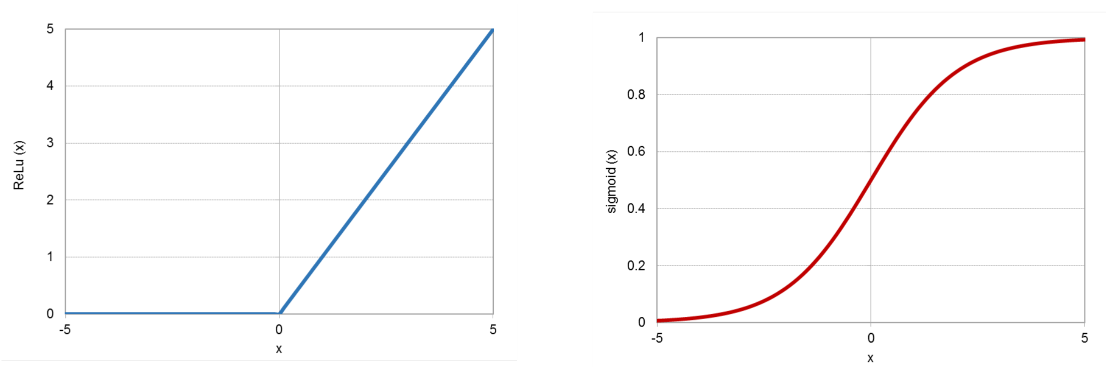
이렇게 설계할 때 전체 연결선 수는 187,000 개이다. 이는 완전연결된 (28 x 28) x 236 x 10 의 구조를 가진 다층신경망에서의 가중치 개수 (187,384)에 해당된다. 그러나 실제로 학습하는 가변 가중치의 수는 14,000 개로서, 이는 전체 연결선 수보다 훨씬 작다(10% 이하). LeNet CNN 의 구조는 완전연결 다층신경망(MLP)보다 크지는 않다. 그러나 실행시간에 컨볼루션을 수행하는 것이 시간을 67% 정도 차지하기 때문에 느리다. 위에서 기술한 CNN 의 경우 3 개의 컨볼루션 층을 사용하므로 실행 속도는 같은 크기의 MLP 보다 세배 정도 느리다.



$$Y_j = \sum_{y=1}^{84} (F_i - w_{ij})^2 , \quad j = 0, \dots, 9$$

Figure 4.5 완전연결층에서의 감독학습

LeNet 형태의 초기 CNN 모델은 전단의 컨볼루션층에서는 시그모이드 함수를 활성함수로 사용하고 후단의 완전연결층에서는 가우스함수 형태의 방사기저함수(radial basis function, RBF)을 사용하였다. 그러나 최근에는 활성함수로서 정류선형함수 ReLU (rectified linear unit) 뉴런을 많이 사용한다. ReLU 는 오차 신호 희석 문제(vanishing gradient problem)를 완화할 수 있는 것으로 알려져 있다.



ReLu **Sigmoid**

Figure 4.6 ReLU 함수와 시그모이드 함수의 비교

4.2 AlexNet

알렉스넷은 딥러닝 혁명을 이끈 초기 딥러닝 모델 중 하나로 컨볼루션 신경망 구조를 갖는다. 이 모델은 대규모 물체인식 콘테스트 (ImageNet LSVRC-2010)에서 2012 년에 우승하였다. LSVRC (Large-Scale Visual Recognition Competition) 대회는 120 만장의 영상을 1000 개의 클래스로 구분하는 대회이다. AlexNet 은 top-5 에러율(모델이 예측한 5 개의 클래스 중에 정답이 포함되지 않은 경우) 17%를 달성하여 기존의 기록을 갱신하였다. 이는 딥러닝으로 기존의 컴퓨터 비전 기법들을 사용한 알고리즘들의 성능을 월등히 능가하였다는 점에서 후속 딥러닝 연구를 촉발한 연구 결과이다. AlexNet 은 총 65 만개의 뉴런과 6 천만개의 파라미터로 이루어져 있고, 5 개의 컨볼루션층과 3 개의 완전연결층으로 구성되었으며 최상위층은 1000 개의 소프트맥스 뉴런으로 구성되었다. 학습을 빠르게하기

위해서 ReLU 뉴런을 사용하였으며 GPU 를 사용하여 컨볼루션 연산을 수행하였다. 과도학습을 방지하기 위해서 완전연결층에 드랍아웃 (dropout) 정규화 방법을 사용하였다. 이 모델을 사용하여 ILSVRC-2012 대회에도 참가하였으며 상위 5 개 테스트 에러율 15.3%로 최고 성능을 획득하였다. 이 성능은 26.2%를 획득한 2 등 보다 월등한 성과이다.
ILSVRC 대회와 기계학습 분야에서 많이 사용되는 ImageNet 데이터는 2 만 2 천개의 카테고리로 구분된 1 천 5 백만 여장의 고해상도 이미지로 이루어진 대용량 데이터집합이다. 이미지는 웹에서 수집하여 아마존 미케니컬 터크(AMT) 크라우스소싱 도구를 이용하여 사람들이 라벨링을 하였다. 2010 년부터 Pascal Visual Object Challenge 의 일부로 ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) 대회가 개최되었다. ILSVRC 는 1,000 개의 카테고리를 가지며, 120 만장의 훈련 이미지와 5 만장의 검증 이미지, 15 만 테스트 이미지로 구성되어 있다.

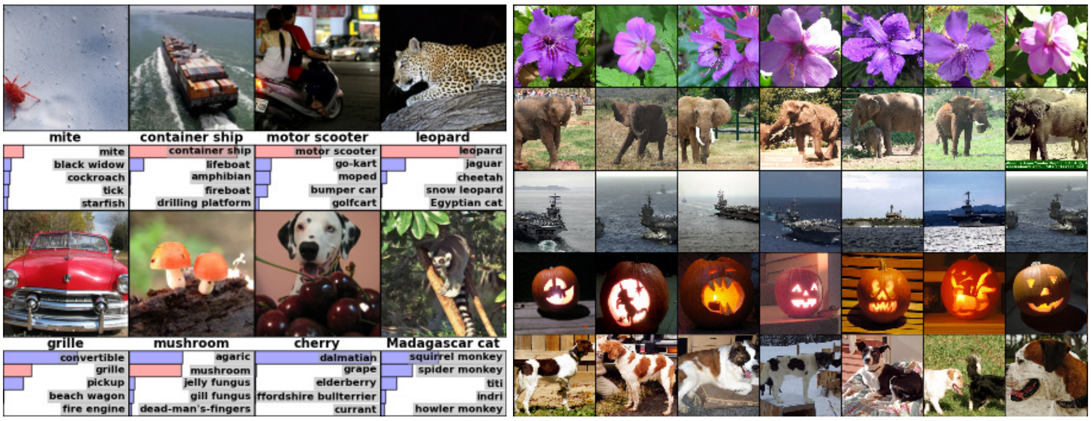


Figure 4.7 ImageNet 데이터

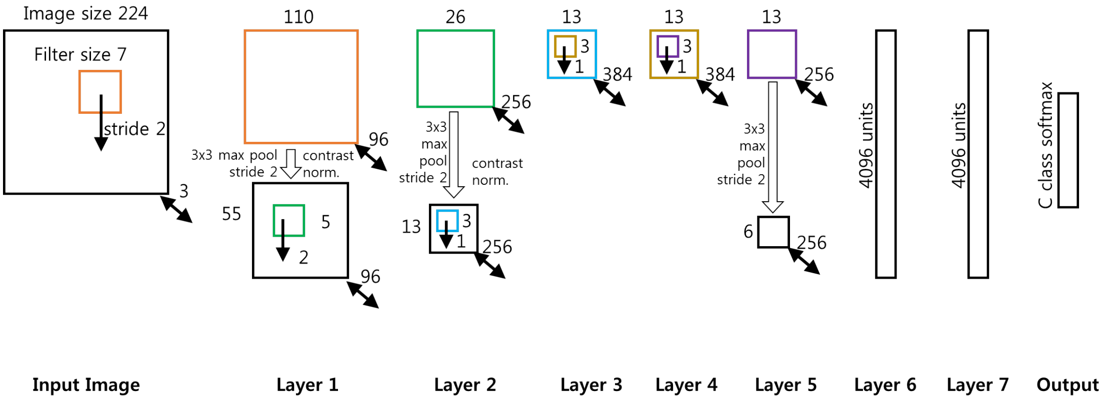


Figure 4.8 AlexNet 의 CNN 신경망 구조

AlexNet 의 구조를 그림 4.8 에 기술하였다. 그림에서 보인 바와 같이 AlexNet 은 8 개의 가중치 층으로 구성되어 있다. 다섯 개의 층은 컨볼루션 층이고 나머지 세 개의 층은 완전연결 층이다. 마지막 완전연결 층의 출력은 1000 개의 클래스 에 대한 확률 분포를 생성하는 소프트맥스에 입력된다. 소프트맥스 함수의 출력값을 이용하여 다항분포 로지스틱 회귀 함수를 (multinomial logistic regression function)를 최대화하도록 학습하였다.
제 1 컨볼루션 층은 224×224×3 의 입력 이미지를 스트라이드 4 (즉 커널맵에 있는 이웃 뉴런들의 수용영역 센터간의 픽셀 거리가 4)로 떨어진 96 개의 커널(각각 11×11×3 크기). 제 2 컨볼루션 층은 제 1 컨볼루션응의 출력을 입력으로 가지며 크기 5 × 5 × 48 인 256 개의 커널을 가지고 여과한다. 제 3, 4, 5 컨볼루션 층은 중간에 아무런 풀링이나 표준화 층 없이

상호 연결된다. 제 3 컨볼루션 층은 3 × 3 × 256 크기의 384 개의 커널을 가지며 제 2 컨볼루션 응에 연결되었다. 제 4 컨볼루션 층은 3 × 3 × 192 크기의 384 개의 커널, 제 5 컨볼루션 층은 3 × 3 × 192 크기의 256 개의 커널을 가진다. 완전연결층은 각각 4096 개의 뉴런을 갖는다.

AlexNet 의 특징 중 하나는 전체 망을 두 개의 GPU 로 분산시켰다는 점이다. 한 개의 GTX 580 GPU 를 사용하면 3GB 의 VRAM 을 사용할 수 있는데 120 만장의 훈련 이미지를 한 개의 GPU 로 학습하기에는 부족하여 두 대의 GPU 를 사용하였다. 두 대의 GPU 를 사용하기 위하여 전체 망을 두 개의 GPU 로 분산하였는데, 각 GPU 는 전체 커널 수의 반을 수용하며 두 개의 GPU 는 특정한 층에서만 서로 통신한다. 예를 들어, 세번째 층의 커널들은 두번째 층의 모든 커널들로부터 입력을 받지만, 네번째 층의 커널 입력은 같은 GPU 에 있는 세번째 층의 커널들로부터만 입력을 받는다. 망의 연결 패턴을 선택하는 것은 교차검증(cross-validation)에 어려움을 주지만 이것은 수용할 만한 계산 시간의 범위 내에서 통신양을 조절할 수 있게 해준다.

각 2 번째, 4 번째, 5 번째의 컨볼루션층의 커널은 같은 GPU 안에 있는 아래층의 커널맵에만 연결되어 있다. 3 층의 커널들은 2 층의 모든 커널맵에 연결되어 있다. 완전연결층에 있는 뉴런들은 바로 아래 층의 모든 뉴런들과 연결되어 있다. 1, 2 컨볼루션 층은 이어서 응답표준화 층을 거친다. 맥스풀링 층은 응답표준화 층과 5 번째 컨볼루션 층 다음에 사용되었다.

AlexNet 의 두 번째 특징은 노드의 활성화 함수로 ReLU 사용한 점이다. ReLU 는 포화가 되지 않는(non-saturating) 비선형 구조를 가지며 포화되는(saturating) 시그모이드 비선형 구조 보다 여러배 빠르게 수렴한다.

AlexNet 의 또 다른 혁신 중의 하나는 과도학습을 방지하기 위한 정규화 방법으로 드랍아웃(dropout)을 도입한 것이다. 서로 다른 모델들의 예측을 결합함으로써 테스트 에러를 줄일 수 있는 것은 잘 알려진 사실이다. 그러나 딥러닝에서는 한개의 큰 신경망만 훈련하는 데도 이미 훈련시간이 몇일씩 걸리는데 이를 여러개 결합하는 것은 현실적으로 쉽지 않다. 그러나 최대 2 배의 시간이 걸리면서 모델을 아주 효율적으로 결합하는 방법이 있다. “드랍아웃”(dropout)이라고 불리는 이 방법은 각 은닉뉴런의 출력을 0.5 의 확률로 0 으로 만든다. 이렇게 “제거된” 뉴런은 전방 패스에 기여하지 않고 또한 역전파에도 참여하지 않는다. 입력이 제시될 때마다 신경망은 다른 구조를 샘플링하며 이 모든 구조들은 가중치를 공유한다. 이 방법은 한 개의 뉴런이 특별한 다른 뉴런에 의존하지 않도록 만들기 때문에 뉴런들 간의 복잡한 공적응(co-adaptation)을 줄여준다. 결과적으로 뉴런들이 다른 뉴런들의 랜덤 부분집합들과 함께 보다 안정적인 유용한 특징을 학습할 수 있게 해 준다. 테스트 시에는 모든 뉴런들을 사용하나 그 출력을 0.5 만큼 곱한다. 이 값은 지수적으로 많은 수의 드랍아웃망에 의해 생성된 예측 분포의 기하 평균을 취하는 것에 해당한다. AlexNet 에서는 드랍아웃이 완전연결 신경망에만 이용되었다. 드랍아웃 없이는 과다학습 현상을 보였다. 그러나 드랍아웃을 사용함으로써 수렴하는데 걸린 반복 횟수는 두 배로 늘어났다.

4.3 DeepFace

최근 엄청난 수의 사진 데이터가 검색 엔진과 소셜 네트워크를 통해서 수집되고 있다. 이러한 데이터는 물체, 얼굴, 장면들을 담고 있는데, 실제 사진 데이터로부터 얼굴과 물체를 인식하기 위해서는 물체의 변형, clutter, 가림, illumination 등 다양한 문제들을 풀어야 한다. 이러한 문제에 대해서 SVM (support vector machines), PCA (principal component analysis), LDA (linear discriminant analysis) 등 기존의 기계학습 알고리즘으로 해결하려는 시도가 많이 있었으나, 학습 데이터의 수가 많을 때에는 학습이 잘 되지 않는다는 단점이 있다.

한편 기계학습이 아닌, 컴퓨터 비전 분야의 연구들은 얼굴이나 물체를 인식하는 문제를 해결하기 위하여 얼굴이나 물체에 대한 다양한 descriptor (기술자)를 찾는 연구를 해 왔다.

이러한 descriptor 는 사람이 미리 고안해 두어야 하고, 영상의 모든 위치에 같은 연산자를 사용한다는 특징이 있다. 하지만 최근에는 학습을 통해 descriptor를 자동으로 추출하는 방법이 많이 이용되고 있다. 이러한 방식은 주어진 문제에 특화된 특징들을 학습을 통해 자동으로 추출할 수 있기 때문에 얼굴이나 물체 인식 문제에서 더 좋은 성능을 보이고 있다. 딥페이스 시스템은 대규모 딥망을 통해서 얼굴 영상의 일반적인 표상을 학습하는 방법이다.

페이스북은 9층짜리 심층신경망을 사용하여 인간수준의 얼굴 인식 성능을 데모하였다. 이 심층망은 일반 CNN과는 달리 노드들이 가중치를 공유하지 않는 다는 점이 특징이다. 이 신경망은 지역적으로 연결된 1억 2천만 개의 노드와 파라미터로 구성되었다.

이 모델의 특징은 심층망으로 학습한 특징벡터와 정확한 모델기반 정렬 알고리즘을 결합하여, 최종 분류 단계에서 간단한 분류기를 사용하였음에도 불구하고 일반 환경에서 뛰어난 얼굴 인식 성능을 보였다는 점이다. 심층망의 학습을 위하여 4천명의 사람에 대한 4백만장의 얼굴 영상이 학습 데이터로 사용되었고 야외 환경에서 모은 얼굴 데이터셋인 LFW에 대해서 97.35%의 정확도를 보였다. 이는 최신 얼굴 인식 성능을 27% 향상한 것이고 인간이 보이는 성능과 거의 유사하다.

4.3.1 딥페이스의 구조와 학습.

그림 2는 딥페이스 시스템의 전체 구조를 보여준다. 3차원으로 정렬된 152 x 152 픽셀의 3채널(RGB) 얼굴 영상이11x11x3 크기의 32개 필터를 가진 컨볼루션 층 C1에 입력된다. 이를 32x11x11x3@152x152로 표기한다. 이로부터 나오는 32개의 특징맵은 2픽셀 간격으로(stride 2) 3x3의 공간 이웃에 대해서 최대치를 취하는 맥스풀링층 M2에 입력된다. 이는 다시 9x9x16 크기의 16개 필터를 갖는 컨볼루션 층 C3에 입력된다. 이 세 개의 층의 목적은 모서리(edge)와 질감(texture)과 같은 저수준의 특징들을 추출하는 것이다. 이 세 개의 층은 앞단의 적응적인 전처리 단계로 본다. 이 층들은 대부분의 계산에 관여하지만 파라미터의 수는 적으며, 입력을 단순한 지역적 특징 집합으로 단지 확장한다.

여기에서 사용된 맥스 풀링층은 CNN 의 출력이 지역적인 변환에 덜 민감하도록 만드는 역할을 한다. 얼굴 영상에 대해서는 특히 이는 작은 등록 에러에 대해서 더 견고하게 만든다. 그러나 많은 층의 풀링을 사용하면 구체적인 얼굴 구조와 정확한 위치에 대한 정보를 잃어버린다.

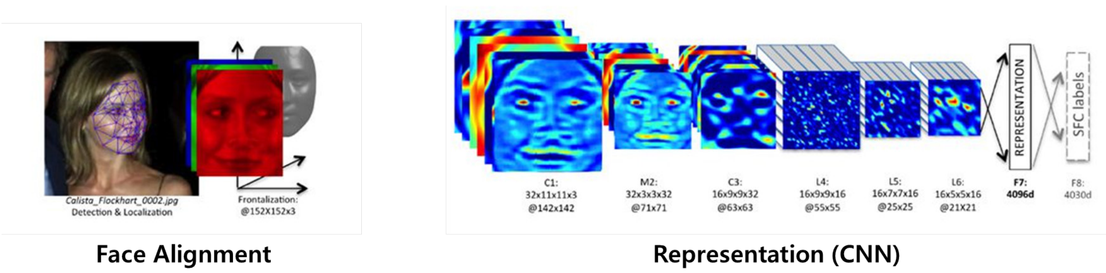


그림 4.7. 딥페이스 시스템의 구조. 정류 입력에 대한 컨볼루션-풀링-컨볼루션 필터링으로 구성된 전단과 이어서 3개의 지역적 연결층과 2개의 완전 연결층으로 구성되어 있다. 딥망은 1억 2천만개 이상의 파라미터를 가지며 이중 95%는 지역적 연결층과 완전 연결층에 속한다.

이어지는 세 개의 층(L4, L5, L6)은 지역적으로 연결되어 있다. 컨볼루션 층처럼 이들은 필터뱅크를 적용한다. 그러나 특징맵의 모든 위치들이 서로 다른 필터 집합을 학습한다. 정합된 영상의 다른 지역이 다른 지역적 통계값을 가지기 때문에 컨볼루션이 공간적으로 불변이라는 가정은 성립되지 않는다. 예를 들어, 눈과 눈썹 사이의 영역은 매우 다른 모습을 보이며 코와 입 사이의 영역과 비교하여 큰 변별 능력을 갖는다. 딥페이스에서는 입력 영상들이 정합되어 있다는 사실을 이용하여 심층 신경망의 구조를 설계하였다. 지역층을

사용하는 것은 특징 추출의 계산에는 부담을 주지 않지만 학습에 참여하는 파라미터의 수에는 영향을 준다. 표지된 학습 데이터셋의 크기가 아주 크기 때문에 세 개의 지역 연결층을 가질 수 있다. 가중치 공유없이 지역 연결층을 사용해도 되는 이유는 지역 연결층의 출력 유닛들 각각이 입력 영상의 아주 큰 팻치에 의해 영향을 받는 다는 가정하에 정당화될 수 있다. 예를 들어, L6의 출력은 입력의 74 x 74 x 3 팻치에 의해서 영향을 받으며 정합된 얼굴에 있는 큰 팻치들 간에는 통계적인 특성의 공유가 거의 없다.

마지막으로 제일 위의 두 층은(F7, F8)은 완전 연결되어 있다. 세 개의 층은 눈과 입의 위치와 모양과 같이 얼굴 영상의 멀리 떨어진 부분에서 추출된 특징들간의 상관관계를 잡아낸다. 첫번째의 완전 연결층(F7)의 출력은 얼굴표상 특징벡터로 사용된다. 이것은 아주 지역적인 디스크립터를 풀링하는 기존의 LBP기반 표상과 대조적이다. 마지막 완전연결층의 출력은 K개 클래스에 대한 소프트맥스에 입력되어 클래스 표지에 대한 분포를 출력한다. ok를 망의 k번째 출력이라고 하면 k번째 클래스에 대한 소프트맥스 함수의 출력은 다음과 같다.

$$\mathrm{p_k} = \frac{\exp(o_k)}{\sum_h \exp(o_h)}$$

학습의 목표는 올바른 클래스(얼굴 표지)의 확률을 최대화하는 것이다. 이는 학습 샘플 각각에 대한 교차 엔트로피 손실을 최소화함으로써 얻을 수 있다. K가 진짜 표지에 대한 인덱스 일때 손실은 L = − log p_k가 된다. 이 손실은 L의 파라미터에 대한 경사도를 계산하고 확률적 경사도 강하(SGD) 알고리즘을 사용하여 파라미터를 변경함으로써 최소화할 수 있다.

경사도는 표준적인 오류역전파 알고리즘에 의해 계산된다. 이 망에 의해서 생성되는 특징들의 특성 중 하나는 그들이 아주 sparse하다는 것이다. 평균적으로 최상위층에 있는 특징 요소들의 75%가 0에 해당한다. 그 이유는 RuLU 활성화 함수 max(0, x)를 사용하기 때문이다. 이 소프트역치 비선형함수는 매 컨볼루션 후에(국부연결층 및 완전연결층) 적용되며 긴 체인들이 매우 비선형적이고 희소한(sparse) 특징을 생성하도록 한다. 희소성은 드랍아웃(dropout)이라 불리는 정규화 방법을 사용하여 보장된다. 딥페이스 시스템에서는 첫 번째의 완전연결층에만 드랍아웃 기법이 사용되었다.

영상 I가 주어지면 표상 G(I)는 딥신경망에 의해 계산된다. L개의 층을 가진 어떤 순방향망도 함수들 g_ϕ의 결합으로 볼 수 있다. 딥페이스망의 경우 표상은

$$\mathrm{G(I)} = g^{\{F_7\}}(g^{\{L_6\}}(\ldots\left(g^{\{C_1\}}\left(T(I,\theta_T)\right)\right)\ldots)$$

이고 여기서 파라미터는 $\phi = \{C_1, \ldots, F_7\}$와 $\Theta_T = \{x_{2d}, P, r\}$ 이다.

ReLU 활성화 함수를 취하기 때문에 이 시스템은 영상 intensity의 recaling에 불변하지 않는다. DNN에 편향이 있기 때문에 표준화를 통해서 완전한 동치성은 손실하게 된다. 그러나 표준화는 보통 분류 성능을 향상하는데 효과적으로 알려져 있다.

4.3.2 데이터셋.

학습은 페이스북의 얼굴 사진 데이터셋인 *SFC* (Social Face Classification)로 학습하였다. 딥러닝에 의해서 학습된 표상을 *LFW* (Labeled Faces in the Wild database)와 *YTF* (YouTube Faces)에 적용하여 성능을 평가하였다.

LFW는 제약이 없는 실제 환경에서 모아진 얼굴 이미지 데이터 셋이며, YTF는 LFW와 유사하나 유튜브 비디오로부터 추출된 얼굴 사진 집합이다. SFC는 4,030명으로부터 각각 800 내지 1200개의 얼굴 사진을 찍은 총 4백 4십만 장의 표지된 얼굴을 포함한다. 이 중에서 5%는 테스트를 위해서 사용하였다.

LFW 데이터셋은 5,749명의 연예인의 웹 사진 13,323개로 구성되어 있다. 10개의 군으로 나누어 각각 6,000개의 얼굴의 쌍으로 구성되어 있다. 성능은 인식 정확도를 측정하였으며 세 가지 다른 방법의 프로토콜을 사용하였다. 첫번째로, 제한된 프로토콜에서는 훈련 데이터가 단지 같은 얼굴인지 다른 얼굴인지에 대한 표지만 포함하고 있다. 두번째의 비제한 프로토콜에서는 훈련시 추가의 훈련쌍을 사용하였다. 세번째의 무감독 세팅에서는 LFW 이미지에 대해서 아무런 훈련을 사용하지 않았다. YTF 데이터셋은1,595명의 연예인(LFW의 부분집합)의 3,425 개의 유튜브 동영상의 집합이다. 5,000개의 비디오 쌍으로 구성된 10개의 그룹으로 구성되어 있다. SFC의 표지는 사람이 하였으며 3%의 오류가 있다. 페이스북 얼굴 사진은 보통 스마트폰 보다는 전문 사진 작가에 의해 촬영되었으며 영상의 질, 조명 등에 많은 변화가 있다.

4.3.2 학습 방법 및 결과

SFC 데이터를 학습한 방법은 다음과 같다. 다중클래스 분류 문제로 보고 오류역전차 알고리즘을 사용하여 확률구배법(stochastic gradient descent, SGD)으로 전방향 딥신경망을 학습시켰다. GPU를 사용하였으며 모멘텀 상수는 0.9를 사용하였다. 미니배치 크기는 128을 사용하였으며 모든 층에 대해서 학습률은 0.01로 모두 같은 것을 적용하였고 검증 에러가 줄 때마다 수동적으로 학습률을 0.0001 까지 줄였다. 각 층의 연결 가중치를 평균값 0, 표준편차 0.01의 가우스 분포로 초기화 했으며 편향값은 0.5로 하였다. 전체 데이터집합을 약 15번 반복 학습했으며 이는 3일이 걸렸다. 학습된 딥러닝 구조의 완전연결층 F7을 추출하여 얼굴 표상으로 사용하였다.

다양한 구조의 심층신경망에 대해 평가하기 위하여 SFC 테스트 데이터에 대해서 평가하였다. 먼저, SFC의 부분 집합을 사용하여 훈련/테스트 데이터집합의 크기를 다양하게 변경하였다. 1.5K, 3K, 4K명의 사람(얼굴 사진)을 사용하여 *DF-1.5K*, *DF-3.3K*, *DF-4.4K*의 세 가지 딥구조를 실험하였을 때 오분류율이 1.5K 데이터에서 7.0%로부터 3K 데이터에서 7.2%로 다소 커지는 것이 확인되었다. 이는 이 딥구조의 메모리 능력이 3K 데이터를 수용할 수 있음을 시사한다. 오분류율은 4K명의 4.4K 이미지에 대해서 8.7%로 커졌으며 이는 더 많은 수의 사람을 포함해도 에러는 크게 증가되지 않는 것으로 해석될 수 있다.

이후 사람 수는 고정 시킨 후에 샘플의 수만 10%, 20%, 50%로 증가시킨 실험 결과에서는 작은 훈련셋에 대해서 과도학습됨에 따라서 테스트 에러가 20.7% 증가하는 것을 관찰할 수 있았다. 4백만개의 영상에 대해서도 성능이 포화되지 않았으며 이 딥구조는 더 큰 데이터셋을 사용함으로써 더욱 더 성능을 향상할 수 있음을 암시한다.

망의 깊이를 변경하여서도 실험하였다. 각각의 C3층, L4층, L5층 또는 이 모든 세개의 층을 제거하여 깊이를 줄여보았다. 이러한 더 얕은 신경망을 4.4M의 데이터셋에 학습했을 때 오분류율이 심층신경망 보다 더 이상 줄지 않는 것을 관측할 수 있었다. 이는 아주 큰 얼굴 인식 데이터에 대해서는 깊은 신경망 구조가 필요함을 의미한다.

4.5 GooLeNet

4.5.1 GooLeNet 구조

구글넷은 22 층짜리 컨볼루션 신경망으로 인셉션 모듈이라는 빌딩 블록을 반복적으로 사용한다. 이는 ILSVRC 2014 의 물체 분류와 물체 검출 과제에서 신기록을 갱신하였다.
일반적으로 딥러닝은 층의 수와 각 층에 포함되는 노드의 수를 증가시킬수록 성능이 향상된다. 하지만 컴퓨팅 파워와 기억 용량의 한계 때문에 무조건 큰 신경망을 구성할 수 없다. 예를 들어, 시각처리를 위한 깊은 컨볼루션 신경망의 경우 두 개의 층이 컨볼루션 연산으로 연결될 때 필터의 수를 균일하게 증가할 경우 계산량은 제곱으로 증가한다. 또한 큰 크기의 망은 더 많은 수의 파라미터를 요구하기 때문에 과다학습 될 가능성이 커진다.

구글넷 구조의 핵심은 제한된 계산 자원을 최대한 이용할 수 있는 신경망을 만드는 것이다. 구조를 잘 설계함으로써 제한된 자원 내에서 망의 깊이와 폭을 증가시킬 수 있도록 하였다. 이를 위한 인셉션 모듈의 핵심 아이디어는 컨볼루션 신경망의 최적인 지역적 희소 구조를 찾아내고 이를 사용 가능한 자원으로 근사해 내는 것이다.
translation invariance 를 컨볼루션 모듈을 조합하여 근사하는 방법을 예를 들어 생각해 보자. Arora et al. [2]는 아래층을 분석하여 상관관계가 높은 노드들을 군집화 하는 방식으로 층을 쌓는 방법을 제안하였다. 아래층의 각 노드들은 입력 영상의 특정한 영역에 대응되며 상관관계가 높은 노드들은 상위 층에서 군집을 형성하여 필터뱅크가 된다. 즉 아래쪽 층에서 상관관계가 큰 노드들이 상위 층에서 비슷한 지역이 위치하게 되고 이들은 다음 층에서 1×1 컨볼루션 층에 사용될 수 있다. 이 때 더 큰 패치에 대해서 컨볼루션 연산을 하게 되면 더 적은 수의 클러스터로도 더 넓은 지역을 커버할 수도 있다. 그러나 가장 적합한 패치의 크기를 미리 아는 것이 어려운 문제이기 때문에, 현재 인셉션 모듈은 필터의 크기를 1×1, 3×3, 5×5 의 세 가지로 제한한다. 이 세가지 크기의 출력 필터 뱅크들을 모두 연결하여 하나의 출력 벡터를 구성하고 이는 다음 층의 입력을 형성한다.
한편, 풀링 연산이 현재 CNN 의 성공에 중요한 것으로 여겨지기 때문에 각 단계에서 다른 병렬 풀링 경로를 더해주는 것이 유용하다고 알려져 있다(그림 4.8).

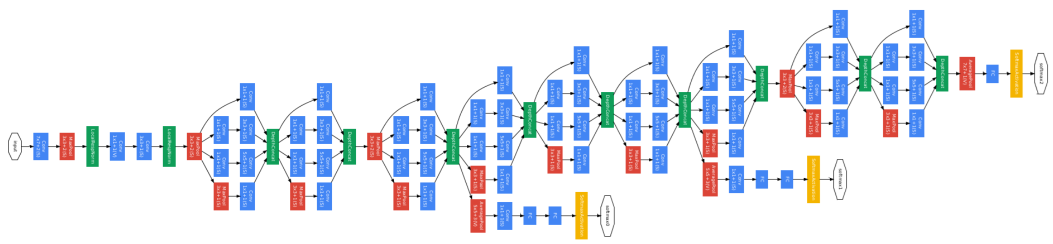


그림 4.8. GoogLeNet 의 구조. 인셉션 모듈 중심의 다층 구조로 구성되었다.

인셉션 모듈은 다른 모듈의 위에 쌓이게 되는데, 이때 출력 상관 통계가 변하게 된다. 일반적으로 상위층으로 갈수록 고차의 추상적인 특징들이 추출되어 그 공간적인 밀집도가 줄어들기 때문에 상위층으로 갈수록 3×3, 5×5 컨볼루션의 비율은 적어져야 할 것이다. 한가지 큰 문제는 작은 수인 5×5 컨볼루션 조차도 아주 많은 수의 필터에 대해서 컨볼루션 층을 구성할 때는 너무 경비가 많이 든다는 것이다. 이 문제는 풀링 노드들이 여기에 추가되면 더욱 심각해지는데 이는 출력 필터의 갯수가 아래 층의 필터의 수와 같기 때문이다. 풀링층 출력을 컨볼루션층의 출력과 결합하면 단계별로 출력의 수가 증가할 수밖에 없다. 이 구조는 최적의 희소 구조를 구성할 수 있지만, 몇 개의 층만 되어도 계산 시간이 폭증하게 되어서 비효율적이다.

이로 인해 인셉션 구조의 두 번째 아이디어가 필요한데, 계산량이 크게 증가하게 되면 차원을 줄이는 것이다. 이는 임베딩(embedding)의 이점을 활용하는 것인데, 실제로 고차원의 데이터가 저차원의 공간으로 잘 임베딩이 된다면 차원이 작더라도 상대적으로 많은 정보를 포함할 수 있다. 그러나 고밀도 압축 형태로 정보를 표상하고 압축된 정보를 처리하는 것이 어려운 작업이기 때문에 꼭 결합될 필요가 있을 때에만 신호들을 압축해야 한다. 따라서 구글넷에서는 비용이 더 큰 3×3, 5×5 컨볼루션을 사용하기 전에 1×1 컨볼루션을 사용하여 차원 축소 계산을 수행한다. 이 때 1×1 컨볼루션은 정류선형활성화 함수를 사용하는 ReLU 뉴런을 사용한다.

구글넷은 위의 모듈들이 적층된 망으로, 입력 해상도를 반으로 줄이기 위해서 간격이 2인 맥스풀링을 하는 층을 곳곳에 추가한다. 또한 아래층에서는 전통적인 컨볼루션을 사용하고 위층에서만 인셉션 모듈들을 사용하는 것이 더 효과적이라고 보고되었다. 결과적으로 구글넷 구조의 핵심은 상위층에서의 계산 복잡도를 줄이면서 각 층의 노드 수를 증가시키는 것인데, 큰 크기의 패치를 사용하여 계산 량이 많은 컨볼루션을 하기 전에 차원 축소를 사용함으로써

이를 구현한다. 뿐만 아니라, 이러한 설계는 다양한 스케일에서 처리된 시각정보들의 특징들을 동시에 결합하는 것이 좋다는 실용적인 직관을 따르는 것이다. 이러한 측면에서 구글넷의 구조는 성능에 약간의 손해를 보면서 계산 비용을 줄이는 방법으로도 볼 수 있다. 구조를 잘 설계한다면 3-10배의 속도를 향상하면서도 유사한 성능을 도달할 수 있다는 것이 알려져 있다.

4.5.2 GooLeNet 학습

구글넷의 깊이는 파라미터 층만 고려하면 22층이고 풀링까지 고려한다면 27층이다. 구글넷을 구성하는 전체 층의 수는 약 100층이지만 정확한 층의 수는 머신러닝 시스템에서 층을 어떻게 정의하느냐에 따라 다르다.

많은 층으로 구성된 깊은 신경망에서 기울기 정보(변화율 벡터, gradient)을 모든 층을 통해서 전파하는 능력이 항상 문제가 되었다. 실제로 신경망의 중간층에서 생성된 특징들이 매우 변별력이 있다는 연구결과가 많이 보고되었다. 이것은 중간층에 보조 분류기를 추가함으로써 분류기의 아랫단에서의 패턴 분류가 가능함을 시사한다. 구글넷에서는 인셉션 모듈의 출력 위에 작은 컨볼루션망을 추가하여 보조 분류기 아이디어를 적용하였다.

훈련시에 각각의 보조 분류기에서 계산되는 손실값이 구글넷 전체 손실값에 (0.3의 할인 가중치를 곱해서) 더해진다. 추론시에는 이 보조망은 사용되지 않는다.

보조 분류기를 포함해서 옆에 붙은 추가망의 구체적인 구조는 다음과 같다.

- 필터크기 와 간격의 평균 풀링 층을 사용하여 4×4×512의 (4a)층 출력과 4×4×528의 (4d) 염탐층(stag)
- 1,024개의 수정선형(ReLU) 뉴런으로 구성된 완전연결층
- 70% 비율로 소거된 출력을 갖는 드랍아웃층
- 소프트맥스 손실치를 갖는 선형층의 분류기 (주분류기와 같은 1000개의 분류를 예측하나 추론시에는 사용되지 않음)

실제로 구글넷을 학습할 때에는 모델과 데이터 병렬성을 이용하는 DistBelief 분산 학습 시스템을 이용하였다. 학습에는 비동기 확률적 기울기하강 알고리즘을 사용하였으며 0.9의 모멘텀 요소와 고정된 학습률 (매 8 번 학습 주기마다 4%씩 학습률을 줄임)을 사용하였다.

4.5.3 데이터 및 결과

구글넷은 ILSVRC 2014 데이터셋에 대해 당시 최고 성능을 보였던 알렉스넷(AlexNet)에 비해 12배 적은 수의 파라미터를 사용하면서도 더 높은 정확도를 획득하였다. 이 대회에서 구글넷은 독립적으로 7가지의 버전을 훈련시킨 후 이를 앙상블하여 결과를 예측하였다. 각 모델은 단지 샘플링 방법과 입력 이미지의 순서만 달랐으며 가중치 초기화나 학습률 정책 등은 모두 동일하였다. 최종 결과는 상위 5위 오류가 검증 데이터와 테스트 데이터에 대해서 6.67%였으며 참가팀 중에서 1등을 하였다. 이는 2012년의 SuperVision 방법에 비해서 성능을 상대적으로 56.5% 향상한 것이며, 직전해의 우승팀인 Clarifai에 비해서 40% 상대적인 성능 향상을 가져온 것이다.

참고 문헌

. [10] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backprop- agation applied to handwritten zip code recognition. *Neural Comput.*, 1(4):541–551, Dec. 1989.

. Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recog- nition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

. A. Krizhevsky, I. Sutskever, and G. Hinton. Im- agenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25*, pages 1106–1114, 2012.

. G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature de- tectors. *CoRR*, abs/1207.0580, 2012.