이미지 인식을 위한 딥 잔여 학습

카이밍 허 샹위 장 샤오칭 렌 지안 선

Microsoft 연구

{kahe, v-xiangz, v-shren, jiansun}@microsoft.com

Abstract

심층 신경망은 훈련하기가 더 어렵습니다. 우리는 이전에 사용된 것보다 훨씬 더 심층적인 네트워크를 쉽게 훈련할 수 있는 잔여 학습 프레임워크를 제시합니다. 우리는 레이어 입력을 참조하여 레이어를 학습 잔차 함수로 명시적으로 재구성합니다.

참조되지 않은 함수를 학습하는 대신에. 이러한 잔여 네트워크가 최적화하기 쉽고 깊이가 상당히 증가하면 정확도를 높일 수 있음을 보여주는 종합적인 경험적 증거를 제공합니다. 이미지넷 데이터 세트에서 최대 152층-8× 의 깊이를 가진 잔여 네트워크를 평가한 결과, VGG 네트워크[41]보다 더 깊지만 복잡도는 여전히 낮았습니다. 이러한 잔여 네트워크의 앙상블은 ImageNet 테스트 세트에서 3.57%의 오류를 달성했습니다. 이 결과는 ILSVRC 2015 분류 과제에서 1위를 차지했습니다. 또한 100개와 1000개의 레이어를 가진 CIFAR-10에 대한 분석 결과도 제시합니다.

표현의 깊이는 많은 시각 인식 작업에서 매우 중요합니다. 전적으로 매우 깊은 표현을 사용했기 때문에 COCO 물체 감지 데이터 세트에서 28%의 상대적 개선 효과를 얻었습니다. 딥 잔류망은 ILSVRC & COCO 2015 대회에 제출한 이미지넷 검출, 이미지넷 로컬라이제이션, COCO 검출 및 COCO 분할 과제에서 1위를 차지한 기반이 되었습니다.

1. 소개

심층 컨볼루션 신경망[22, 21]은 이미지 분류에 일련의 혁신을 가져왔습니다[21, 50, 40]. 딥 네트워크는 저/중/고 수준의 특징[50]과 분류기를 엔드투엔드 멀티레이어 방식으로 자연스럽게 통합하며, 특징의 '수준'은 쌓인 레이어의 수(깊이)에 따라 강화될 수 있습니다. 최근의 증거[41, 44]에 따르면 네트워크 깊이가 매우 중요하며, 까다로운 이미지넷 데이터 세트[36]에 대한 주요 결과[41, 44, 13, 16]는 모두 16[41]~30[16]의 깊이를 가진 "매우 깊은"[41] 모델을 활용하고 있습니다. 다른 많은 비사소한 시각 인식 작업[8, 12, 7, 32, 27]에서도 다음과 같은 결과를 얻었습니다.

[1] http://image-net.org/challenges/LSVRC/2015/ 과 http://mscoco.org/dataset/#detections-challenge2015.

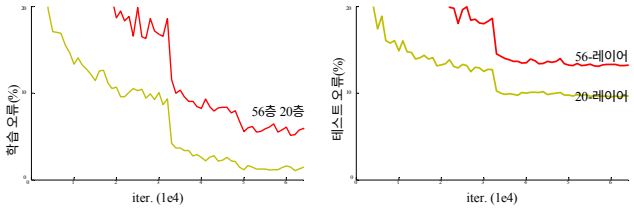

그림 1. 20층 및 56층 "일반" 네트워크를 사용한 CIFAR-10의 훈련 오차(왼쪽) 및 테스트 오차(오른쪽). 네트워크가 깊을수록 훈련 오차가 커지고 따라서 테스트 오차도 커집니다. 이미지넷에서도 비슷한 현상이 그림 4에 나와 있습니다.

는 매우 심층적인 모델의 이점을 크게 누리고 있습니다.

깊이의 중요성에 따라 한 가지 의문이 생깁니다: *더 나은 네트워크를 학습하는 것이 더 많은 레이어를 쌓는 것만큼 쉬운 일인가?* 이 질문에 답하는 데 걸림돌이 되는 것은 처음부터 수렴을 방해하는 소실/폭발 그라디언트[1, 9]라는 악명 높은 문제였습니다. 그러나 이 문제는 정규화된 초기화[23, 9, 37, 13]와 중간 정규화 레이어[16]로 대부분 해결되었으며, 이를 통해 수십 개의 레이어를 가진 네트워크가 역전파를 통해 확률적 기울기 하강(SGD)을 위한 수렴을 시작할 수 있게 되었습니다[22].

더 깊은 네트워크가 수렴을 시작할 수 있게 되면 네트워크 깊이가 증가함에 따라 정확도가 포화 상태가 되고(이는 놀라운 일이 아닐 수 있습니다) 급격히 저하되는 *성능* 저하 문제가 노출되었습니다. 의외로 이러한 성능 저하는 *과적합으로 인한* 것이 아니며, [11, 42]에서 보고되고 실험을 통해 철저히 검증된 바와 같이 적절히 깊은 모델에 더 많은 레이어를 추가하면 *훈련 오류가 더 커집니다*. 그림 1은 대표적인 예시입니다.

(훈련 정확도의) 저하는 모든 시스템이 비슷하게 최적화하기 쉬운 것은 아니라는 것을 나타냅니다. 더 얕은 아키텍처와 그 위에 더 많은 레이어를 추가하는 더 깊은 아키텍처를 고려해 보겠습니다. 추가된 계층은 *ID* 매핑이고 다른 계층은 학습된 얕은 모델에서 복사한 것입니다. 심층 모델에 대한 *구성에 의한* 솔루션이 존재합니다. 이렇게 구성된 솔루션이 존재한다는 것은 더 깊은 모델이 더 얕은 모델보다 더 높은 훈련 오류를 생성하지 않아야 한다는 것을 의미합니다. 그러나 실험에 따르면 현재 사용 중인 솔버는 다음과 같은 해를 찾지 못합니다.

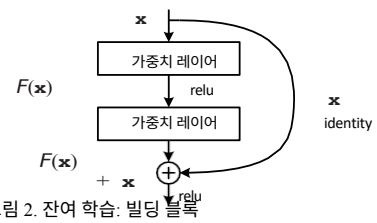

그림 2. 잔여 학습: 빌딩 블록

가 구축된 솔루션보다 비슷하거나 더 나은 경우(또는 실현 가능한 시간 내에 그렇게 할 수 없는 경우).

이 백서에서는 *심층 잔여 학습* 프레임워크를 도입하여 성능 저하 문제를 해결합니다. 스택된 몇 개의 레이어가 원하는 기본 매핑에 직접적으로 부합하기를 바라는 대신, 이러한 레이어가 잔여 매핑에 부합하도록 명시적으로 허용합니다. 공식적으로 원하는 기저 매핑을 $H(\mathbf{x})$로 표시하면, 스택된 비선형 레이어가 $F(\mathbf{x}) := H(\mathbf{x}) - \mathbf{x}$의 다른 매핑에 맞도록 하고, 원래의 기저 매핑은 $F(\mathbf{x})+\mathbf{x}$로 다시 캐스팅됩니다. 우리는 참조되지 않은 원본 매핑을 최적화하는 것보다 잔여 매핑을 최적화하는 것이 더 쉽다는 가설을 세웁니다. 극단적으로 아이덴티티 매핑이 최적이라면 비선형 레이어 스택으로 아이덴티티 매핑을 맞추는 것보다 잔차를 0으로 밀어내는 것이 더 쉬울 것입니다.

$F(\mathbf{x})+\mathbf{x}$ 공식은 '바로 가기 연결'이 있는 피드포워드 신경망으로 구현할 수 있습니다(그림 2). 바로가기 연결[2, 34, 49]은 하나 이상의 레이어를 건너뛰는 연결입니다. 우리의 경우, 바로 가기 연결은 단순히 *ID* 매핑을 수행하고 그 출력이 스택된 레이어의 출력에 추가됩니다(그림 2). ID 바로 가기 연결은 추가 매개변수나 컴퓨팅 복잡성을 추가하지 않습니다. 전체 네트워크는 여전히 SGD를 통해 역전파를 통해 엔드 투 엔드로 훈련할 수 있으며, 솔버를 수정하지 않고도 일반적인 라이브러리(*예*: Caffe [19])를 사용하여 쉽게 구현할 수 있습니다.

우리는 ImageNet에 대한 포괄적인 실험을 제시합니다.
[36]에 대한 포괄적인 실험을 제시하여 성능 저하 문제를 보여주고 우리의 방법을 평가합니다. 우리는 그것을 보여줍니다: 1) 매우 깊은 잔류 네트워크는 최적화하기 쉽지만, 이에 대응하는 "일반" 네트워크(단순히 레이어를 쌓는)는 깊이가 증가할 때 더 높은 훈련 오류를 보입니다. 2) 깊은 잔류 네트워크는 깊이가 크게 증가해도 정확도 향상을 쉽게 누릴 수 있어 이전 네트워크보다 훨씬 나은 결과를 생성합니다.

CIFAR-10 세트[20]에서도 비슷한 현상이 나타나, 최적화 어려움과 우리 방법의 효과가 특정 데이터 세트에만 국한된 것이 아님을 시사합니다. 100개 이상의 레이어가 있는 이 데이터 세트에서 성공적으로 훈련된 모델을 제시하고, 1000개 이상의 레이어가 있는 모델을 살펴봅니다.

ImageNet 분류 데이터 세트[36]에서는 매우 깊은 잔여 네트워크를 통해 우수한 결과를 얻었습니다. 152개의 레이어로 구성된 잔여 네트워크는 ImageNet에서 지금까지 제시된 네트워크 중 가장 깊은 네트워크이면서도 VGG 네트워크[41]보다 복잡도가 낮습니다. 우리 앙상블의 상위 5위 오차는 **3.57%입니다**.

ImageNet *테스트* 세트에서 *1위를 차지했으며*, *ILSVRC 2015 분류 대회에서 1위를 차지했습니다*. 매우 심층적인 대표 표현은 다른 인식 작업에서도 우수한 일반화 성능을 보이며 추가적으로 *1위를 차지할 수 있었습니다: 이미지넷 검출, 이미지넷 로컬라이제이션, COCO 검출, COCO 세분화에서* ILSVRC & COCO 2015 대회에서 1위를 차지했습니다. 이러한 강력한 증거는 잔존 학습 원리가 일반적이라는 것을 보여주며, 다른 시각 및 비시각 문제에도 적용될 수 있을 것으로 기대합니다.

2. 관련 작업

잔여 표현. 이미지 인식에서 VLAD
[18]은 사전을 기준으로 잔여 벡터로 부호화하는 표현이며, 피셔 벡터(Fisher Vector) [30]는 VLAD의 확률론적 버전[18]으로 공식화할 수 있습니다. 두 가지 모두 이미지 재검색 및 분류를 위한 강력한 얕은 표현입니다[4, 48]. 벡터 양자화의 경우, 잔여 벡터[17]를 인코딩하는 것이 원본 벡터를 인코딩하는 것보다 더 효과적인 것으로 나타났습니다.

저수준 비전 및 컴퓨터 그래픽에서 편미분 방정식(PDE)을 풀 때 널리 사용되는 멀티그리드 방법[3]은 시스템을 여러 스케일에서 하위 문제로 재구성하며, 각 하위 문제는 더 큰 스케일과 더 작은 스케일 사이의 잔류 해에 대해 응답할 책임이 있습니다. 멀티그리드의 대안으로 계층적 기반 사전 조건화[45, 46]가 있는데, 이는 두 스케일 사이의 잔여 벡터를 재전송하는 변수에 의존합니다. 이러한 솔버는 해의 잔류 특성을 인식하지 못하는 표준 솔버보다 훨씬 빠르게 수렴하는 것으로 나타났습니다[3, 45, 46]. 이러한 방법은 좋은 재구성 또는 전제 조건이 최적화를 단순화할 수 있음을 시사합니다.

바로 가기 연결. 바로 가기 연결[2, 34, 49]로 이어지는 관행과 이론은 오랫동안 연구되어 왔습니다. 다층 퍼셉트론(MLP) 훈련의 초기 사례는 네트워크 입력에서 출력으로 연결된 선형 계층을 추가하는 것입니다[34, 49]. 44, 24]에서는 소실/폭발 그라데이션을 처리하기 위해 몇 개의 중간 계층이 보조 분류기에 직접 연결됩니다. 39, 38, 31, 47]의 논문에서는 단축 연결로 구현된 레이어 재연결, 그라데이션 및 전파된 오류의 중심을 잡는 방법을 제안합니다. 44]에서 "시작" 레이어는 지름길 가지와 몇 개의 더 깊은 가지로 구성됩니다.

우리의 작업과 동시에 "고속도로 네트워크"[42, 43]는 게이팅 기능[15]을 갖춘 바로가기 연결을 제시합니다. 이러한 게이트는 데이터에 따라 달라지며 매개변수가 있는 반면, 우리의 아이덴티티 바로가기는 매개변수가 없습니다. 게이트 지름길이 "닫힌"(0에 가까워짐) 경우, 고속도로 네트워크의 레이어는 *잔류* 함수가 없는 함수를 나타냅니다. 반대로, 우리의 공식은 항상 잔여 함수를 학습하며, 아이덴티티 지름길은 닫히지 않고 모든 정보가 항상 통과하며 추가적으로 학습해야 할 잔여 함수가 있습니다. 또한, 하이

방식 네트워크는 깊이가 극도로 증가해도(*예*: 100개 이상의 레이어) 정확도 향상을 입증하지 못했습니다.

3. 심층 잔차 학습

3.1. 잔차 학습

$H(\mathbf{x})$를 몇 개의 적층된 레이어(반드시 전체 망은 아님)에 의해 맞춰지는 기본 매핑으로 간주하고, $\mathbf{x}$는 이러한 레이어 중 첫 번째 레이어에 대한 입력을 나타냅니다. 여러 개의 비선형 계층이 복잡한 함수를 점근적으로 *근사화할* 수 있다고 가정하면[2].잔여 함수를 점근적으로 근사화할 수 있다고 가정하는 것과 동일하게 $H(\mathbf{x})-\mathbf{x}$(입력과 출력의 차원이 같다고 가정)가 성립합니다(입력과 출력의 차원이 같다고 가정하면). 따라서 스택된 레이어가 $H(\mathbf{x})$를 근사화할 것으로 기대하는 대신, 이러한 레이어가 명시적으로 잔차 함수 $F(\mathbf{x}) := H(\mathbf{x})-\mathbf{x}$를 근사화하도록 하면 원래 함수는 $F(\mathbf{x})+\mathbf{x}$가 됩니다. 두 형태 모두 (가설대로) 원하는 함수를 점근적으로 근사화할 수 있어야 하지만, 학습의 용이성은 다를 수 있습니다.

이러한 재구성은 성능 저하 문제에 대한 반직관적인 현상에서 동기를 얻었습니다(그림 1, 왼쪽). 서론에서 설명했듯이, 추가된 레이어를 아이덴티티 매핑으로 구성할 수 있다면 더 깊은 모델은 더 얕은 모델보다 학습 오차가 크지 않아야 합니다. 성능 저하 문제는 솔버가 여러 비선형 레이어로 아이덴티티 매핑을 근사화하는 데 어려움을 겪을 수 있음을 시사합니다. 잔여 학습 재구성을 사용하면 아이덴티티 매핑이 최적이라면 솔버는 여러 비선형 레이어의 가중치를 0으로 하여 아이덴티티 매핑에 근접할 수 있습니다.

실제 사례에서 아이덴티티 매핑이 최적일 가능성은 낮지만, 우리의 재구성은 문제를 전제 조건으로 삼는 데 도움이 될 수 있습니다. 최적 함수가 제로 매핑보다 아이덴티티 매핑에 더 가깝다면 솔버가 새로운 함수를 학습하는 것보다 아이덴티티 매핑을 참조하여 섭동을 찾는 것이 더 쉬워질 것입니다. 실험(그림 7)을 통해 학습된 잔차 함수는 일반적으로 작은 응답을 가지며, 이는 아이덴티티 매핑이 합리적인 전제 조건을 제공한다는 것을 보여줍니다.

3.2. 단축키를 통한 아이덴티티 매핑

우리는 몇 개의 스택된 레이어마다 잔여 학습을 채택합니다. 빌딩 블록은 그림 2에 나와 있습니다. 공식적으로 이 백서에서는 빌딩 블록을 다음과 같이 정의합니다:

$$\mathbf{y}= F(\mathbf{x}, \{W_i\})+ \mathbf{x}. \quad (1)$$

여기서 x와 $\mathbf{y}$는 고려되는 레이어의 입력 및 출력 벡터입니다. 함수 $F(\mathbf{x}, \{W_{(i)}\})$는 학습할 잔여 매핑을 나타냅니다. 그림 2의 예시에서 두 개의 레이어가 있는 경우 $F= W_{(2)}\sigma(W_{(1)}\mathrm{x})$ 여기서 σ는 다음을 나타냅니다.

[2] 그러나 이 가설은 아직 미해결 문제입니다. 28]을 참고하세요.

ReLU [29]와 편향은 노테이션을 단순화하기 위해 생략했습니다. 연산 $F+\mathbf{x}$는 바로 가기 연결과 요소별 덧셈으로 수행됩니다. 우리는 덧셈 후 초차 비선형성을 채택합니다(*즉*, $\sigma(\mathbf{y})$, 그림 2 참조).

식 (1)의 지름길 연결은 추가 매개변수나 계산 복잡성을 유발하지 않습니다. 이는 실제로 매력적일 뿐만 아니라 일반 네트워크와 잔여 네트워크를 비교할 때에도 중요합니다. 매개변수의 수, 깊이, 폭, 계산 비용이 동일한 일반/잔여 네트워크를 동시에 비교할 수 있기 때문입니다(무시할 수 있는 요소별 추가를 제외하면). 식 (1)에서 x와 F의 치수는 같아야 합니다. 그렇지 않은 경우(*예*: 입력/출력 변경 시 채널을 변경하는 경우 등), 바로 가기 연결을 통해 선형 투영 $W_{(s)}$를수행하여 차원을 일치시킬 수 있습니다:

$$\mathbf{y}= F(\mathbf{x}, \{W_i\})+ W_s\mathbf{x}. \quad (2)$$

식 (1)의 정사각형 행렬 $W_{(s)}$를사용할 수도 있습니다. 하지만 실험을 통해 아이덴티티 매핑이 성능 저하 문제를 해결하기에 충분하고 경제적이므로 차원을 일치시킬 때만 W_s를 사용한다는 것을 보여줄 것입니다.

잔차 함수 F의 형태는 유연합니다. 이 백서의 실험에서는 두 개 또는 세 개의 층을 가진 함수 F를 사용했지만(그림 5), 더 많은 층을 사용할 수도 있습니다. 그러나 F에 단 하나의 층만있는 경우, 식.(1)은 선형 층과 유사합니다: $\mathbf{y}= W_{(1)}\mathbf{x}+\mathbf{x}$, 이 경우 이점을 관찰하지 못했습니다. 또한 위의 표기는 단순화를 위해 완전히 연결된 레이어에 관한 것이지만, 컨볼루션 레이어에도 적용 가능합니다. 함수 $F(\mathbf{x}, \{W_{(i)}\})$는 여러 컨볼루션 레이어를 다시 전송할 수 있습니다. 요소별 추가는 두 개의 피처 맵에서 채널별로 수행됩니다.

3.3. 네트워크 아키텍처

다양한 일반/잔여 네트워크를 테스트한 결과, 일관된 현상을 확인했습니다. 토론을 위한 사례를 제공하기 위해 이미지넷의 두 가지 모델을 다음과 같이 설명합니다.

플레인 네트워크. 플레인 기준선(그림 3, 가운데)은 주로 VGG 네트워크[41]의 철학에서 영감을 얻었습니다(그림 3, 왼쪽). 컨볼루션 레이어는 대부분 3개의× 3 필터로 구성되며, 두 가지 간단한 설계 규칙을 따릅니다. (i) 출력 피처 맵 크기가 동일한 경우 레이어의 필터 수는 동일하고, (ii) 피처 맵 크기가 절반이면 필터 수는 두 배가 되어 레이어당 시간 복잡도를 보존합니다. 네트워크는 글로벌 평균 풀링 레이어와 소프트맥스를 사용한 1000방향 완전 연결 레이어로 마무리되며, 보폭이 2인 컨볼루션 레이어에서 직접 다운샘플링을 수행합니다. 그림 3(가운데)에서 가중치 레이어의 총 개수는 34개입니다.

우리 모델은 VGG 네트워크[41]보다 필터 *수가* 적고 복잡성이 *낮다는* 점에 주목할 필요가 있습니다(그림 3, 왼쪽). 34층 베이스라인의 경우 36억 FLOP(곱하기 덧셈)으로, VGG-19(196억 FLOP)의 18%에 불과합니다.

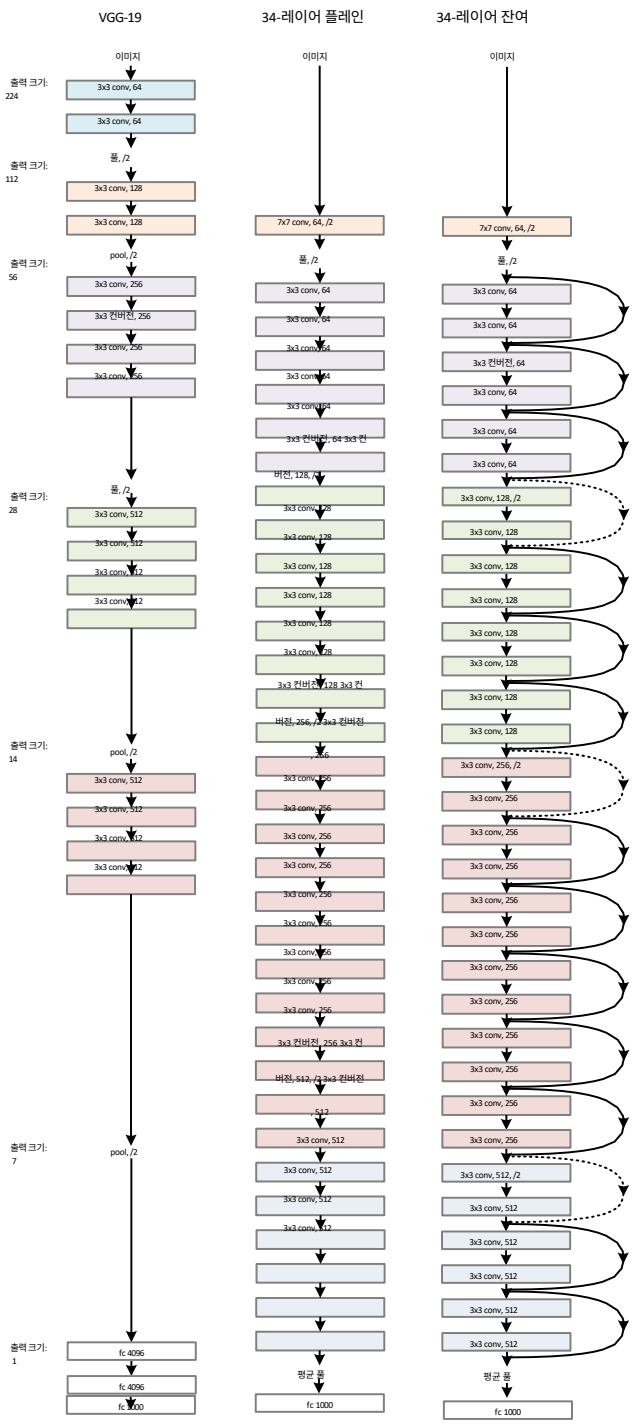

그림 3. ImageNet의 네트워크 아키텍처 예시. **왼쪽**: 참조용 VGG-19 모델[41](196억 FLOPs). **가운데**: 34개의 파라미터 레이어가 있는 일반 네트워크(36억 FLOPs). **오른쪽**: 34개의 파라미터 레이어가 있는 잔여 네트워크(36억 FLOPs). 점선으로 표시된 지름길은 차원을 증가시킵니다. **표 1은** 자세한 내용과 기타 변형을 보여줍니다.

잔여 네트워크. 위의 플레인 네트워크를 기준으로 바로 가기 연결(그림 3, 오른쪽)을 삽입하여 네트워크를 해당 잔여 버전으로 전환합니다. 입력과 출력이 동일한 차원(그림 3의3의 실선 지름길)일 때는 동일성 지름길(식.(1))을 직접 사용할 수 있습니다(그림 실선 지름길). 차원이 증가하면(그림 3의 점선 바로 가기) 두 가지 옵션을 고려합니다. (A) 바로 가기는 여전히 아이덴티티 매핑을 수행하지만 차원 증가에 따라 추가 0 항목이 추가됩니다. 이 옵션은 추가 매개변수가 필요하지 않습니다. (B) 식(2)의 투영 바로 가기를 사용하여 차원을 일치시킵니다(1× 1 컨볼루션으로 수행). 두 옵션 모두 두 가지 크기의 피처 맵을 가로지르는 경우 바로 가기가 2의 보폭으로 수행됩니다.

3.4. 구현

이미지넷에 대한 우리의 구현은 [21, 41]의 사례를 따릅니다. 이미지의 크기는 스케일 확대를 위해 [256, 480]에서 샘플링된 짧은 측면의 랜덤으로 조정됩니다 [41]. 224× 224 크롭은 이미지 또는 이미지의 수평 뒤집기에서 무작위로 샘플링되며, 픽셀당 평균을 뺍니다 [21]. 21]의 표준 색상 확대가 사용됩니다. 각 컨볼루션 직후와 활성화 전에 [16]에 따라 일괄 정규화(BN)[16]를 적용합니다. 13]에서와 같이 가중치를 초기화하고 모든 일반/잔여 네트워크를 처음부터 훈련합니다. 미니 배치 크기가 256인 SGD를 사용합니다. 학습 속도는 0.1부터 시작하여 오차가 정체되면 10으로 나누고, 모델은 최대 60×10^4반복으로 훈련됩니다. 0.0001의 가중치 감쇠와 0.9의 모멘텀을 사용합니다. 드롭아웃 [14]은 [16]의 관행에 따라 사용하지 않습니다.

테스트에서는 비교 연구를 위해 표준 10-작물 테스트 [21]를 채택합니다. 최상의 결과를 위해 [41, 13]에서와 같이 완전 컨볼루션 형태를 채택하고 여러 스케일에서 점수를 평균합니다(이미지의 크기가 {224, 256, 384, 480, *640}에* 속하도록 조정됨).

4. 실험

4.1. 이미지넷 분류

1000개의 클래스로 구성된 ImageNet 2012 분류 데이터 세트[36]에서 우리의 방법을 평가합니다. 128만 개의 훈련 이미지로 모델을 훈련하고 5만 개의 검증 이미지로 평가합니다. 또한 테스트 서버에서 보고한 10만 개의 테스트 이미지에 대한 최종 결과를 얻습니다. 오류율 상위 1%와 상위 5%를 모두 평가합니다.

플레인 네트워크. 먼저 18-레이어 및 34-레이어 플레인 네트워크를 평가합니다. 34 계층 플레인 네트워크는 그림 3(가운데)에 있습니다. 그리고 18층 플레인 넷도 비슷한 형태입니다. 디테일 아키텍처는 표 1을 참조하세요.

표 2의 결과는 더 깊은 34층 플레인 넷이 더 얕은 18층 플레인 넷보다 검증 오차가 더 크다는 것을 보여줍니다. 그 이유를 밝히기 위해 그림 4(왼쪽)에서 훈련 과정 중 훈련/검증 오류를 비교했습니다. 성능 저하 문제를 관찰했습니다.

레이어 이름	출력 크기	18-레이어	34 레이어	50 레이어	101 레이어	152-layer
conv1	112× 112	7× 7, 64, 보폭 2				
conv2 x_	56× 56	3× 3 최대 풀, 스트라이드 2				
		3× 3, 64 3× 3, 64 ×2	3× 3, 64 3× 3, 64 ×3	1× 1, 64 3× 3, 64 3× 1× 1, 256	1× 1, 64 3× 3, 64 3× 1× 1, 256	1× 1, 64 3× 3, 64 3× 1× 1, 256
conv3 x	28× 28	3× 3, 128 3× 3, 128 ×2	3× 3, 128 3× 3, 128 ×4	1× 1, 1 2 8 3× 3, 128 4× 1× 1, 512	1× 1, 1 2 8 3× 3, 128 4× 1× 1, 512	1× 1, 1 2 8 3× 3, 128 8× 1× 1, 512
conv4 x	14× 14	3× 3, 256 3× 3, 256 ×2	3× 3, 256 3× 3, 256 ×6	1× 1, 2 5 6 3× 3, 256 6× 1× 1, 1024	1× 1, 2 5 6 3× 3, 256× 23 1× 1, 1024	1× 1, 2 5 6 3× 3, 256× 36 1× 1, 1024
conv5 x	7 7×	3× 3, 512 3× 3, 512 ×2	3× 3, 512 3× 3, 512 ×3	1× 1, 5 1 2 3× 3, 512 3× 1× 1, 2048	1× 1, 5 1 2 3× 3, 512 3× 1× 1, 2048	1× 1, 5 1 2 3× 3, 512 3× 1× 1, 2048
	1 1×	평균 풀, 1000-D FC, 소프트맥스				
FLOPs		1.8× 10⁹	3.6× 10⁹	3.8× 10⁹	7.6× 10⁹	11.3× 10⁹

표 1. ImageNet의 아키텍처. 빌딩 블록은 괄호 안에 표시되어 있으며(그림 5 참조), 블록의 개수는 쌓인 개수입니다. 다운 샘플링은 conv3 1, conv4 1, conv5 1의 보폭 2로 수행됩니다.

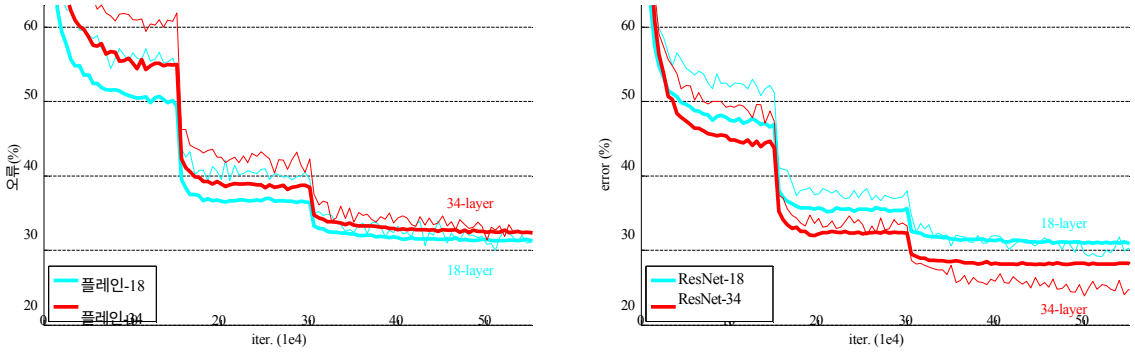

그림 4. **ImageNet에서의** 훈련. 가는 곡선은 훈련 오류를 나타내고 굵은 곡선은 중앙 크롭의 검증 오류를 나타냅니다. 왼쪽: 18개 및 34개의 레이어로 구성된 일반 네트워크. 오른쪽: 18층과 34층의 레즈넷. 이 그래프에서 잔류 네트워크는 일반 네트워크에 비해 추가 파라미터가 없습니다.

	plain	ResNet
18 레이어	27.94	27.88
34 레이어	28.54	**25.03**

표 2. 이미지넷 유효성 검사에서 상위 1% 오류(%, 10-크롭 테스트). 여기서 ResNet은 일반 이미지넷에 비해 추가 매개변수가 없습니다. 그림 4는 훈련 절차를 보여줍니다.

18층 일반 네트워크의 솔루션 공간이 34층 일반 네트워크의 하위 공간임에도 불구하고 전체 훈련 절차에서 34층 일반 네트워크가 더 높은 *훈련* 오류를 보였습니다.

이러한 최적화 어려움은 소실 경사 때문에 발생하는 것 같지는 *않습니다*. 이러한 일반 네트워크는 순방향으로 전파되는 신호가 0이 아닌 편차를 갖도록 보장하는 BN [16]으로 훈련됩니다. 또한 역방향으로 전파된 그라디언트도 BN을 통해 건강한 규범을 나타내는지 확인합니다. 따라서 순방향 신호나 역방향 신호 모두 사라지지 않습니다. 실제로 34층 플레인 넷은 여전히 경쟁사 수준의 정확도를 달성할 수 있으며(표 3), 이는 솔버가 어느 정도 작동한다는 것을 시사합니다. 딥 플레인 넷은 기하급수적으로 낮은 수렴율을 가질 수 있으며, 이로 인해 훈련 오류의 감소[3]. 이러한 최적화 어려움에 대한 이유는 향후 연구될 예정입니다.

잔여 네트워크. 다음으로 18층 및 34층 잔류망(*ResNet*)을 평가합니다. 기본 아키텍처는 위의 플레인 넷과 동일하며, 그림 3(오른쪽)과 같이 3개의× 3 필터 쌍에 각각 바로가기 연결이 추가된다고 가정합니다. 첫 번째 비교(표 2 및 그림 4 오른쪽)에서는 모든 바로가기에 ID 매핑을 사용하고 차원을 늘리기 위해 제로 패딩을 사용합니다(옵션 A). 따라서 일반 옵션에 비해 *추가 매개변수가 없습니다*.

표 2와 그림 4에서 크게 세 가지를 관찰할 수 있습니다. 첫째, 잔존 학습에서는 상황이 역전되어 34층 ResNet이 18층 ResNet보다 2.8% 더 우수합니다. 더 중요한 것은 34층 ResNet이 학습 오류가 훨씬 더 낮고 검증 데이터에 일반화할 수 있다는 점입니다. 이는 이 설정에서 성능 저하 문제가 잘 해결되고 심도 증가로 인한 정확도 향상을 얻을 수 있음을 나타냅니다.

둘째, 34 계층은 일반 계층과 비교했을 때

[3] 더 많은 훈련 반복(3×)을 실험했지만 여전히 성능 저하 문제가 발생했으며, 이는 단순히 더 많은 반복을 사용하는 것만으로는 이 문제를 해결할 수 없음을 시사합니다.

model	TOP-1 ERR.	TOP-5 ERR.
VGG-16 [41]	28.07	9.33
GoogLeNet [44]	-	9.15
PReLU-net [13]	24.27	7.38
plain-34	28.54	10.02
ResNet-34 A	25.03	7.76
ResNet-34 B	24.52	7.46
ResNet-34 C	24.19	7.40
ResNet-50	22.85	6.71
ResNet-101	21.75	6.05
ResNet-152	**21.43**	**5.71**

표 3. ImageNet 검증 시 오류율(%, **10-크롭** 테스트). VGG-16은 당사 테스트 기준입니다. ResNet-50/101/152는 치수를 늘리는 데만 투영을 사용하는 옵션 B입니다.

메서드	top-1 err.	TOP-5 ERR.
VGG [41] (ILSVRC'14)	-	8.43†
GoogLeNet [44] (ILSVRC'14)	-	7.89
VGG [41] (v5)	24.4	7.1
PReLU-net [13]	21.59	5.71
BN-inception [16]	21.99	5.81
ResNet-34 B	21.84	5.71
ResNet-34 C	21.53	5.60
ResNet-50	20.74	5.25
ResNet-101	19.87	4.60
ResNet-152	**19.38**	**4.49**

표 4. ImageNet 검증 세트에서 **단일 모델** 결과의 오류율(%)(테스트 세트에서 보고된†제외).

방법	TOP-5 ERR. (**테스트**)
VGG [41] (ILVRC'14)	7.32
GoogLeNet [44] (ILSVRC'14)	6.66
VGG [41] (v5)	6.8
PReLU-net [13]	4.94
BN-인셉션 [16]	4.82
ResNet(ILSVRC'15)	**3.57**

표 5. **앙상블의** 오류율(%). 상위 5위 오류는 이미지넷의 테스트 세트에 있으며 테스트 서버에서 보고한 오류입니다.

ResNet은 학습 오류를 성공적으로 줄인 결과 상위 1%의 오류를 3.5% 감소시켰습니다(표 2). 이 비교를 통해 매우 심층적인 시스템에서 잔여 학습의 효과를 확인할 수 있습니다.

마지막으로, 18층 일반/잔류망의 정확도는 비슷하지만(표 2), 18층 ResNet이 더 빠르게 수렴한다는 사실도 확인할 수 있습니다(그림 4 오른쪽과 왼쪽 비교). 그물이 "지나치게 깊지 않은" 경우(여기서는 18층), 현재 SGD 솔버는 여전히 일반 그물에 대한 좋은 해를 찾을 수 있습니다. 이 경우 ResNet은 초기 단계에서 더 빠른 수렴을 제공함으로써 최적화를 용이하게 합니다.

아이덴티티 *대*투영 지름길. 우리는 다음을 보여주었습니다.

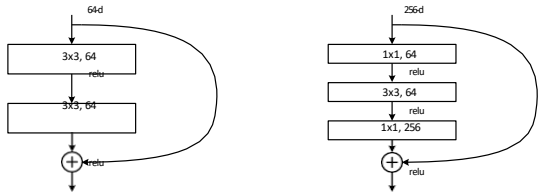

그림 5. ImageNet의 심층 잔차 함수 *F*. ×왼쪽: 그림 3에서와 3의 같이 빌딩 블록(그림 ResNet-.

34. 오른쪽: ResNet-50/101/152의 "병목" 빌딩 블록.

매개변수가 없는 아이덴티티 단축키는 훈련에 도움이 됩니다. 다음으로 투영 지름길(식.(2))을 살펴봅니다. 표 3에서는 (A) 차원을 증가시키는 데 제로 패딩 지름길을 사용하고 모든 지름길은 파라미터가 없는 경우(표 2 및 그림 4 오른쪽과 동일), (B) 차원을 증가시키는 데 프로젝션 지름길을 사용하고 다른 지름길은 동일성, (C) 모든 지름길은 투영의 세 가지 옵션을 비교합니다. 표 3을 보면 세 가지 옵션이 모두 일반 옵션보다 훨씬 더 나은 것으로 나타났습니다. B가 A보다 약간 더 나은데, 이는 A의 제로 패딩 차원에 실제로 잔여 학습이 없기 때문이라고 주장합니다. C는 B보다 약간 더 나은데, 이는 많은(13개) 투영 지름길에 의해 도입된 추가 매개변수 때문이라고 생각합니다. 그러나 A/B/C의 차이가 크지 않다는 것은 투영 단축키가 성능 저하 문제를 해결하는 데 필수적이지 않다는 것을 나타냅니다. 따라서 이 백서의 나머지 부분에서는 메모리/시간 복잡성과 모델 크기를 줄이기 위해 옵션 C를 사용하지 않습니다. 아이덴티티 지름길은 특히 다음 과 같은 복잡성을 증가시키지 않는 데 중요합니다. 아래에서 소개하는 병목 현상 아키텍처를 살펴보세요.

더 심층적인 병목 아키텍처. 다음으로 ImageNet에 대한 심층적인 그물에 대해 설명합니다. 우리가 감당할 수 있는 훈련 시간에 대한 우려 때문에 빌딩 블록을 *병목* 설계로 수정합니다[4]. 각 잔여 함수 *F에* 대해 2개가 아닌 3개의 레이어로 구성된 스택을 사용합니다(그림 5). 3개의 레이어는 1개의× 1, 3개의× 3, 1개의× 1 컨볼루션으로, 1개의× 1 레이어는 차원을 줄였다가 다시 늘리는(복원하는) 역할을 담당하고 3개의× 3 레이어는 입력/출력 차원이 작은 병목 현상을 남깁니다. 그림 5는 두 설계의 시간 복잡성이 비슷한 예시를 보여줍니다.

매개변수가 없는 아이덴티티 지름길은 병목 현상이 발생하는 아키텍처에서 특히 중요합니다. 그림 5(오른쪽)의 아이덴티티 지름길을 투영으로 대체하면 지름길이 두 개의 고차원 끝에 연결되므로 시간 복잡도와 모델 크기가 두 배가 되는 것을 볼 수 있습니다. 따라서 아이덴티티 지름길은 병목 설계에 더 효율적인 모델로 이어집니다.

50층 ResNet: 각 2계층 블록을

[4] 더 깊은 *비보틀넥* ResNet(*예:* 그림 5 왼쪽)도 깊이가 증가하면 정확도가 향상되지만(CIFAR-10에 표시된 것처럼), 병목 현상 ResNet만큼 경제적이지는 않습니다. 따라서 병목 설계를 사용하는 것은 주로 실용적인 고려 사항 때문입니다. 또한 병목 설계에서도 플레인 넷의 성능 저하 문제가 목격됩니다.

34 레이어 넷에 이 3 레이어 병목 블록을 추가하면 50 레이어 ResNet이 됩니다(표 1). 차원을 늘리기 위해 옵션 B를 사용합니다. 이 모델에는 38억 개의 FLOP이 있습니다.

101 레이어 및 152 레이어 ResNet: 3 레이어 블록을 더 많이 사용하여 101 레이어 및 152 레이어 ResNet을 구성합니다(표 1). 놀랍게도, 깊이가 크게 증가했음에도 불구하고 152층 ResNet(113억 FLOPs)은 여전히 VGG-16/19 넷(153억/19.6억 FLOPs)보다 *복잡도가 낮습니다*.

50/101/152 계층 ResNet은 34 계층보다 상당한 차이로 더 정확합니다(표 3과 4). 성능 저하 문제가 관찰되지 않았으며, 따라서 상당히 증가한 깊이로 인해 상당한 정확도 향상을 누릴 수 있습니다. 모든 평가 지표에서 심도의 이점을 확인할 수 있습니다(표 3 및 4).

최신 방법과의 비교. 표 4는 이전의 최고 단일 모델 결과와 비교한 것입니다. 기준이 되는 34계층 레스넷은 매우 경쟁력 있는 정확도를 달성했습니다. 152개 레이어 ResNet의 단일 모델 상위 5개 검증 오류는 4.49%입니다. 이 단일 모델 결과는 이전의 모든 앙상블 결과를 능가합니다(표 5). 깊이가 다른 6개의 모델을 결합하여 앙상블을 구성합니다(제출 당시에는 152층 모델 2개만 사용). 그 결과 테스트 세트에서 **3.57%의** 상위 5위 오차가 발생했습니다(표 5). *이 출품작은 ILSVRC 2015에서 1위를 차지했습니다*.

4.2. CIFAR-10 및 분석

10개 클래스의 5만 개의 훈련 이미지와 1만 개의 테스트 이미지로 구성된 CIFAR-10 데이터 세트[20]에 대한 추가 연구를 수행했습니다. 여기서는 훈련 세트에서 학습하고 테스트 세트에서 평가한 실험을 소개합니다. 우리는 극도로 심층적인 네트워크의 동작에 초점을 맞추고 있지만, 최첨단 결과를 제시하는 것이 아니기 때문에 의도적으로 다음과 같은 간단한 아키텍처를 사용합니다.

일반/잔여 아키텍처는 그림 3(가운데/오른쪽)의 형식을 따릅니다. 네트워크 입력은 32개의× 32개의 이미지이며, 픽셀당 평균을 뺀 값입니다. 첫 번째 레이어는 3× 3개의 컨볼루션입니다. 그런 다음 {32, 16, *8}* 크기의 피처 맵에 각각 3개의× 3 컨볼루션이 있는 *6n* 레이어 스택을 사용하고 각 피처 맵 크기에 대해 2n 레이어를 사용합니다. 필터의 수는 각각 {16, 32, *64} 개입니다*. 서브샘플링은 보폭이 2인 컨볼루션으로 수행됩니다. 네트워크는 글로벌 평균 풀링, 10방향 완전 연결 레이어, 소프트 맥스로 끝납니다. 총 6n+2개의 스택 가중치 레이어가 있습니다. 다음 표는 아키텍처를 요약한 것입니다:

출력 맵 크기	32× 32	16× 16	8 8×
# 레이어	*1+2n*	*2n*	*2n*
# 필터	16	32	64

바로 가기 연 결 이 사용되는 경우, 3개의× 3 레이어 쌍(총 3n 개의 바로 가기)으로 연결됩니다. 이 데이터 세트에서는 모든 경우(*즉*, 옵션 A)에 ID 바로 가기를 사용합니다,

method			error (%)
Maxout [10]			9.38
NIN [25]			8.81
DSN [24]			8.22
	# 레이어	# params	
FitNet [35]	19	2.5M	8.39
고속도로 [42, 43]	19	2.3M	7.54 (7.72± 0.16)
고속도로 [42, 43]	32	1.25M	8.80
ResNet	20	0.27M	8.75
ResNet	32	0.46M	7.51
ResNet	44	0.66M	7.17
ResNet	56	0.85M	6.97
ResNet	110	1.7M	**6.43** (6.61± 0.16)
ResNet	1202	19.4M	7.93

표 6. **CIFAR-10** 테스트 세트의 분류 오류. 모든 방법은 데이터 증강을 사용했습니다. ResNet-110의 경우, 5회 실행하여 [43]과 같이 "최고(평균± std)"를 표시했습니다.

따라서 잔여 모델은 일반 모델과 깊이, 폭, 매개변수 수가 정확히 동일합니다.

0.0001의 가중치 감쇠와 0.9의 모멘텀을 사용하고, [13]과 BN [16]의 가중치 초기화를 채택하지만 드롭아웃은 없습니다. 이 모델은 두 개의 GPU에서 128개의 미니 배치 크기로 훈련됩니다. 0.1의 학습 속도로 시작하여 32k 및 48k 반복에서 10으로 나누고, 45k/5k 훈련/값 분할로 결정되는 64k 반복에서 훈련을 종료합니다. 훈련에는 [24]의 간단한 데이터 증강을 따릅니다. 각 면에 4픽셀을 패딩하고 패딩된 이미지에서 32× 32 크롭을 무작위로 샘플링하거나 수평으로 뒤집습니다. 테스트에서는 원본 32× 32 이미지의 단일 보기만 평가합니다.

n= {3, 5, 7, *9}를* 비교하여 20, 32, 44, 56 계층 네트워크로 이어집니다. 그림 6(왼쪽)은 플레인 넷의 동작을 보여줍니다. 깊이가 깊은 플레인 넷은 깊이가 깊어질수록 훈련 오차가 커집니다. 이러한 현상은 이미지넷(그림 4, 왼쪽)과 MNIST([42] 참조)에서도 유사한데, 이는 이러한 최적화 난이도가 근본적인 문제임을 시사합니다.

그림 6(가운데)은 ResNets의 동작을 보여줍니다. 이미지넷의 경우(그림 4, 오른쪽)와 마찬가지로, 우리의 ResNet도 최적화 어려움을 극복하고 깊이가 증가할 때 정확도가 향상되는 것을 보여줍니다.

110층 ResNet으로 이어지는 *n*= 18을 추가로 살펴봅니다. 이 경우 초기 학습률 0.1이 수렴을 시작하기에는 약간 너무 크다는 것을 알 수 있습니다[5]. 그래서 우리는
0.01로 설정하여 학습 오류가 80% 미만(약 400회 반복)이 될 때까지 학습을 워밍업한 다음 0.1로 돌아가 학습을 계속합니다. 나머지 학습 일정은 이전과 동일합니다. 이 110층 네트워크는 잘 수렴합니다(그림 6, 가운데). 다른 딥 및 씬 네트워크보다 파라미터 *수가 적습니다*.

[5] 초기 학습률이 0.1이면 몇 번의 에포크 후에 수렴하기 시작하지만(< 90% 오류) 여전히 비슷한 정확도에 도달합니다.

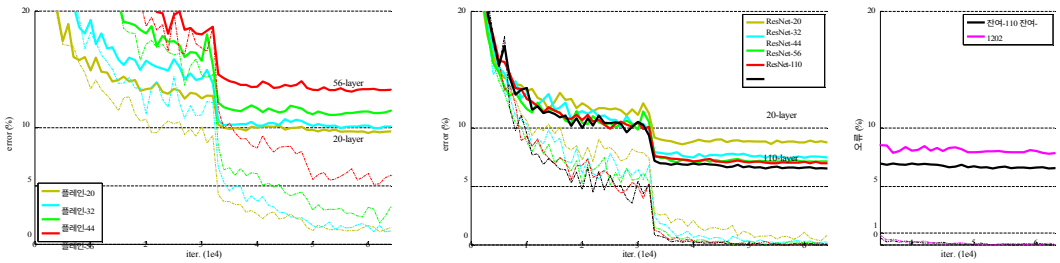

그림 6. **CIFAR-10에** 대한 훈련. 점선은 훈련 오류를, 굵은 선은 테스트 오류를 나타냅니다. **왼쪽**: 일반 네트워크. plain-110의 오차는 60%보다 높아 표시되지 않습니다. **가운데**: ResNets. **오른쪽**: 110 및 1202 레이어가 있는 ResNets.

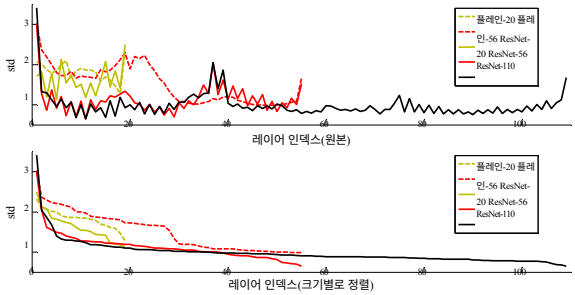

그림 7. CIFAR-에 대한 레이어 응답의 표준편차(std)
10. 응답은 각 3개의× 3 레이어의 출력으로, BN 이후와 비선형성 이전입니다. **위**: 레이어가 원래 순서대로 표시됩니다. **아래**: 응답이 내림차순으로 순위가 매겨집니다.

FitNet [35] 및 Highway [42]와 같은 네트워크(표 6)에 비해 낮은 수준이지만 최신 결과 중 하나입니다(6.43%, 표 6).

레이어 응답 분석. 그림 7은 레이어 응답의 표준편차(std)를 보여줍니다. 응답은 각 3개의× 3 계층의 출력으로, BN 이후와 다른 비선형성(ReLU/추가)이 발생하기 전입니다. ResNets의 경우, 이 분석은 잔여 함수의 응답 강도를 보여줍니다. 그림 7은 ResNet이 일반적으로 일반 함수에 비해 응답이 작다는 것을 보여줍니다. 이러한 결과는 잔차 함수가 일반적으로 비잔차 함수보다 0에 가까울 수 있다는 우리의 기본 동기(섹션 31).를 뒷받침합니다. 또한 그림 7의 ResNet-20, 56, 110을 비교한 결과에서 알 수 있듯이, 더 깊은 ResNet일수록 응답의 크기가 더 작다는 것을 알 수 있습니다. 레이어가 많을수록 ResNet의 개별 레이어가 신호를 덜 수정하는 경향이 있습니다.

1,000개 이상의 레이어 탐색. 1000개 이상의 레이어로 구성된 공격적으로 심층적인 모델을 탐색합니다. n= 200을 설정하여 위에서 설명한 대로 훈련된 1202층 네트워크로 연결합니다. 우리의 방법은 *최적화 난이도가 높지* 않으며, 이 10^3층 네트워크는 *학습 오차*
<0.1%(그림 6, 오른쪽). 테스트 오류는 여전히 상당히 양호합니다.
(7.93%, 표 6).

하지만 이렇게 공격적으로 심층적인 모델에는 여전히 미해결 문제가 남아 있습니다. 이 1202 계층 네트워크의 테스트 결과는 110 계층 네트워크의 테스트 결과보다 더 나쁩니다.

훈련 데이터	07+12	07++12
테스트 데이터	VOC 07 테스트	VOC 12 테스트
VGG-16	73.2	70.4
ResNet-101	**76.4**	**73.8**

표 7. **베이스라인** Faster R-CNN을 사용한 PASCAL VOC 2007/2012 테스트 세트의 객체 탐지 맵(%). 더 나은 결과는 표 10과 11을 참조하세요.

메트릭	mAP@.5	mAP@[.5, .95]
VGG-16	41.5	21.2
ResNet-101	**48.4**	**27.2**

표 8. **기준선** Faster R-CNN을 사용한 COCO 검증 세트의 객체 감지 맵(%). 더 나은 결과는 표 9를 참조하십시오.

비슷한 훈련 오류를 보입니다. 우리는 이것이 과적합 때문이라고 주장합니다. 1202층 네트워크는 이 작은 데이터 세트에 비해 불필요하게 클 수 있습니다(19.4M). 이 데이터 세트에서 최상의 결과([10, 25, 24, 35])를 얻기 위해 최대 아웃 [10] 또는 드롭아웃 [14]과 같은 강력한 정규화를 적용합니다. 이 논문에서는 최적화의 어려움에 초점을 맞추지 않기 위해 맥아웃/드롭아웃을 사용하지 않고 설계상 깊고 얇은 아키텍처를 통해 단순히 정규화를 적용했습니다. 하지만 더 강력한 정규화와 결합하면 결과가 개선될 수 있으며, 이에 대해서는 향후 연구할 예정입니다.

4.3. PASCAL 및 MS COCO에서의 물체 감지

우리의 방법은 다른 인식 작업에서도 우수한 일반화 성능을 보입니다. 표 7과 표 8은 PASCAL VOC 2007과 2012의 객체 디텍션 기준 결과를 보여줍니다.
[5] 및 COCO [26]. 저희는 *더 빠른 R-CNN* [32]을 디텍션 방법으로 채택했습니다. 여기서는 VGG-16 [41]을 ResNet-101로 대체했을 때의 개선점에 관심이 있습니다. 두 모델을 사용하는 탐지 구현(부록 참조)은 동일하므로 더 나은 네트워크에서만 이득을 얻을 수 있습니다. 가장 주목할 만한 점은 까다로운 COCO 데이터 세트에서 COCO의 표준 메트릭(mAP@[.5,
.95]), 이는 28%의 상대적 개선입니다. 이러한 이득은 전적으로 학습된 표현에 기인합니다.

딥 잔류망을 기반으로 ILSVRC & COCO 2015 대회에서 여러 트랙에서 1위를 차지했습니다: Im-ageNet 검출, ImageNet 로컬라이제이션, COCO 검출, COCO 분할에서 1위를 차지했습니다. 자세한 내용은 부록에 나와 있습니다.

참고 자료

[1] Y. 벤지오, P. 시마드, P. 프라스코니. 경사 하강으로 장기 의존성을 학습하는 것은 어렵습니다. *IEEE 신경망 트랜잭션 네트워크*, 5(2):157-166, 1994.

[2] C. M. Bishop. *패턴 인식을 위한 신경망*. Oxford 대학 출판부, 1995.

[3] W. L. 브릭스, S. F. 맥코믹 외. *멀티그리드 튜토리얼*. Siam, 2000.

[4] K. 챗필드, V. 렘피츠키, A. 베달디, 그리고 A. 지서만. 악마는 디테일에 있다: 최근 피처 인코딩 방법의 평가. In *BMVC*, 2011.

[5] M. 에버링햄, L. 반 굴, C. K. 윌리엄스, J. 윈, A. 지스-서먼. 파스칼 시각 객체 클래스 (VOC) 챌린지. *IJCV*, 303-338 페이지, 2010.

[6] S. 기다리스와 N. 코모다키스. 다중 지역 및 시맨틱 세그먼테이션 인식 CNN 모델을 통한 객체 감지. In *ICCV*, 2015.

[7] R. Girshick. 빠른 R-CNN. In *ICCV*, 2015.

[8] R. Girshick, J. Donahue, T. Darrell, and J. Malik. 정확한 객체 감지 및 의미론적 세분화를 위한 풍부한 기능 계층 구조. In *CVPR*, 2014.

[9] X. 글로롯과 Y. 벤지오. 심층 피드포워드 신경망 훈련의 어려움에 대한 이해. In *AISTATS*, 2010.

[10] I. J. 굿펠로우, D. 워데-팔리, M. 미르자, A. 쿠르빌, 및 Y. 벤지오. 최대 아웃 네트워크. *arXiv:1302.4389*, 2013.

[11] K. 그와 J. Sun. 제한된 시간에서의 컨볼루션 신경망 비용. In *CVPR*, 2015.

[12] K. He, X. Zhang, S. Ren, and J. Sun. 시각 인식을 위한 심층 컨볼루션 네트워크의 공간 피라미드 풀링. In *ECCV*, 2014.

[13] K. He, X. Zhang, S. Ren, and J. Sun. 정류기에 대해 깊이 파헤치기: 이미지넷 분류에서 인간 수준의 성능을 뛰어넘기. In ICCV, 2015.

[14] G. E. 힌튼, N. 스리바스타바, A. 크리제프스키, I. 수츠케버, 및 R. R. 살라쿠트디노프. 특징 검출기의 공동 적응을 방지하여 신경망 개선. *arXiv:1207.0580*, 2012.

[15] S. 호크라이터와 J. 슈미드후버. 장단기 기억. *신경 계산*, 9(8):1735-1780, 1997.

[16] S. Ioffe and C. Szegedy. 배치 정규화: 내부 공변량 이동을 줄임으로써 심층 네트워크 훈련 가속화. In *ICML*, 2015.

[17] H. Jegou, M. Douze, 및 C. Schmid. 가장 가까운 이웃 검색을 위한 제품 양자화. *TPAMI*, 33, 2011.

[18] H. Jegou, F. Perronnin, M. Douze, J. Sanchez, P. Perez, and C. Schmid. 로컬 이미지 디스크립터를 압축 코드로 통합하기. *TPAMI*, 2012.

[19] Y. 지아, E. 셸하머, J. 도나휴, S. 카라예프, J. 롱, R. 기르식, S. 과다라마, 그리고 T. 대럴. Caffe: 빠른 기능 임베딩을 위한 컨볼루션 아키텍처. *arXiv:1408.5093*, 2014.

[20] A. 크리제프스키. 작은 이미지에서 여러 계층의 특징 학습하기- . *기술 보고서*, 2009.

[21] A. Krizhevsky, I. Sutskever, and G. Hinton. 심층 컨볼루션 신경망을 사용한 이미지넷 분류 . In *NIPS*, 2012.

[22] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. 손으로 쓴 우편 번호 인식에 적용된 역전파- . *신경 계산*, 1989.

[23] Y. LeCun, L. Bottou, G. B. Orr, and K.-R. Mu¨ller. 효율적인 배경. *신경망에서: 무역의 트*릭, 9-50 페이지. Springer, 1998.

[24] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu. Deeply- 감독망. *arXiv:1409.5185*, 2014.

[25] M. 린, Q. 첸, 그리고 S. 얀. 네트워크 속의 네트워크. *arXiv:1312.4400*, 2013.

[26] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dolla´r, and C. L. Zitnick. Microsoft COCO: 컨텍스트의 공통 개체. In *ECCV*. 2014.

[27] J. Long, E. Shelhamer, and T. Darrell. 의미론적 세분화를 위한 완전 컨볼루션 네트워크. In *CVPR*, 2015.

[28] G. 몬투파, R. 파스카누, K. 조, Y. 벤지오. 심층 신경망의 선형 영역의 수에 대해. In *NIPS*, 2014.

[29] V. Nair와 G. E. Hinton. 정류된 선형 단위는 제한된 볼츠만 머신을 개선합니다. In *ICML*, 2010.

[30] F. 페로닌과 C. 댄스. 이미지 분류를 위한 시각 어휘에 대한 피셔 커널. In *CVPR*, 2007.

[31] T. 라이코, H. 발폴라, 및 Y. 르쿤. 퍼셉트론의 선형 변환으로 더 쉬워진 딥러닝. In *AISTATS*, 2012.

[32] S. Ren, K. He, R. Girshick, and J. Sun. 더 빠른 R-CNN: 영역 제안 네트워크를 통한 실시간 물체 감지를 향하여. In *NIPS*, 2015.

[33] S. Ren, K. He, R. Girshick, X. Zhang, and J. Sun. 컨볼루션 특징 맵에서 객체 감지 네트워크. *arXiv:1504.06066*, 2015.

[34] B. D. 리플리. *패턴 인식 및 신경망*. 캠브리지 대학 출판부, 1996.

[35] A. 로메로, N. 발라스, S. E. 카호우, A. 차상, C. 가타, 그리고 Y. Bengio. Fitnets: 얇고 깊은 그물을 위한 힌트. In *ICLR*, 2015.

[36] O. 러사코브스키, J. 덩, H. 수, J. 크라우스, S. 사테쉬, S. 마, Z. 황, A. 카파시, A. 코슬라, M. 번스타인 외. Imagenet 대규모 시각 인식 도전. *arXiv:1409.0575*, 2014.

[37] A. M. 색스, J. L. 맥클렐랜드, 그리고 S. 강굴리. 심층 선형 신경망에서 학습의 비선형 동역학에 대한 정확한 솔루션. *arXiv:1312.6120*, 2013.

[38] N. N. Schraudolph. 인자 중심 분해에 의한 가속 경사 하강. 기술 보고서, 1998.

[39] N. N. 슈뢰돌프. 신경망 그라데이션 인자 센터링. 신 경 망 에 서 *: 무역의 트*릭, 207-226 페이지. 스프링거, 1998.

[40] P. Sermanet, D. 아이겐, X. 장, M. 마티유, R. 퍼거스, Y. 르쿤. 오버피트: 컨볼루션 네트워크를 이용한 통합 인식, 로컬라이제이션 및 탐지 . In *ICLR*, 2014.

[41] K. Simonyan and A. Zisserman. 대규모 이미지 인식을 위한 매우 심층적인 컨볼루션 네트워크 . In *ICLR*, 2015.

[42] R. K. 스리바스타바, K. 그레프, 및 J. 슈미드후버. 고속도로 네트워크. *arXiv:1505.00387*, 2015.

[43] R. K. 스리바스타바, K. 그레프, 그리고 J. 슈미드후버. 매우 깊은 네트워크 훈련. *1507.06228*, 2015.

[44] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Er-han, V. Vanhoucke, A. Rabinovich. 컨볼 루션으로 더 깊이 들어가기 . In *CVPR*, 2015.

[45] R. Szeliski. 계층적 기초 함수를 사용한 빠른 표면 보간 . *TPAMI*, 1990.

[46] R. Szeliski. 국부적으로 적응된 계층적 기저 사전 조건화. In *시그그래프*, 2006.

[47] T. Vatanen, T. Raiko, H. Valpola, and Y. LeCun. 비선형성의 변환을 이용한 역전파 학습을 위한 2차 방법으로 스토캐스틱 그라데이션 추진. *신경 정보 처리*, 2013.

[48] A. 베달디와 B. 풀커슨. VLFeat: 컴퓨터 비전 알고리즘의 개방형 휴대용 라이브러리 , 2008.

[49] W. Venables and B. Ripley. S-plus를 사용한 현대 응용 통계. 1999.

[50] M. D. Zeiler와 R. Fergus. 컨볼 루션 신경망 시각화 및 이해. *ECCV*, 2014.

A. 객체 감지 기준선

이 섹션에서는 기준선인 Faster R-CNN [32] 시스템에 기반한 감지 방법을 소개합니다. 이 모델은 이미지넷 분류 모델에 의해 초기화된 다음 객체 감지 데이터에 따라 미세 조정됩니다. 우리는 ILSVRC & COCO 2015 검출 대회 당시 ResNet-50/101로 실험을 진행했습니다.

32]에서 사용된 VGG-16과 달리, 저희 ResNet에는 숨겨진 FC 레이어가 없습니다. 이 문제를 해결하기 위해 "네트워크 온 컨브 피처 맵"(네트워크 온 컨브 피처 맵, NoC)[33]이라는 개념을 채택했습니다. 이미지의 보폭이 16픽셀 이하인 레이어를 사용하여 전체 이미지 공유 conv 특징 맵을 계산합니다(*즉*, conv1, conv2 x, conv3 x, conv4 x, ResNet-101의 총 91개 conv 레이어; 표 1). 이러한 레이어는 VGG-16의 13개 conv 레이어와 유사하며, 이렇게 함으로써 ResNet과 VGG-16 모두 동일한 총 보폭(16픽셀)의 conv 특징 맵을 갖게 됩니다. 이러한 레이어는 지역 제안 네트워크(RPN, 300개의 제안 생성)에 의해 공유됩니다.

[32] 및 고속 R-CNN 탐지 네트워크[7]를 사용합니다. RoI 풀링[7]은 conv5 1 이전에 수행됩니다. 이 RoI 풀링 기능에서는 각 영역에 대해 conv5 x 이상의 모든 레이어가 채택되어 VGG-16의 $\bar{f}$c 레이어의 역할을 수행합니다. 최종 분류 레이어는 두 개의 형제 레이어(분류 및 상자 회귀 [7])로 대체됩니다.

BN 레이어를 사용하기 위해 사전 훈련 후, ImageNet 훈련 세트에서 각 레이어에 대한 BN 통계(평균과 분산)를 계산합니다. 그런 다음 물체 감지를 위한 미세 조정 중에 BN 레이어를 고정합니다. 따라서 BN 레이어는 일정한 오프셋과 스케일을 가진 선형 활성화가 되며, 미세 조정을 통해 BN 통계가 업데이트되지 않습니다. 우리는 주로 Faster R-CNN 훈련에서 메모리 소비를 줄이기 위해 BN 레이어를 수정합니다.

파스칼 VOC

7, 32]에 따라, PASCAL VOC 2007 *테스트* 세트의 경우, 훈련("07+12")을 위해 VOC 2007의 5k *trainval* 이미지와 VOC 2012의 16k *train-val* 이미지를 사용합니다. PASCAL VOC 2012 *테스트* 세트의 경우, 훈련("07++12")을 위해 VOC 2007의 10k *trainval+ 테스트* 이미지와 VOC 2012의 16k *trainval* 이미지를 사용합니다. Faster R-CNN을 훈련하기 위한 하이퍼 파라미터는 [32]와 동일합니다. 표 7은 결과를 보여줍니다. ResNet-101은 VGG-16에 비해 mAP를> 3% 향상시킵니다. 이러한 이득은 전적으로 ResNet이 학습한 향상된 기능 덕분입니다.

MS COCO

MS COCO 데이터 세트[26]에는 80개의 객체 카테고리가 포함되어 있습니다. PASCAL VOC 메트릭(mAP @ IoU= 0.5)과 표준 COCO 메트릭(mAP @ IoU =
.5:.05:.95). 훈련에는 트레인 세트의 8만 개 이미지를 사용하고 평가에는 밸 세트의 4만 개 이미지를 사용합니다. COCO에 대한 탐지 시스템은 PASCAL VOC에 대한 탐지 시스템과 유사합니다. 8-GPU 구현으로 COCO 모델을 훈련하므로 RPN 단계의 미니 배치 크기는 다음과 같습니다.

8개 이미지(*즉*, GPU당 1개), Fast R-CNN 단계의 미니 배치 크기는 16개 이미지입니다. RPN 단계와 Fast R-CNN 단계는 모두 0.001의 학습률로 240,000회 반복 학습한 다음 0.0001의 학습률로 80,000회 반복 학습합니다.

표 8은 MS COCO 검증 세트의 결과를 보여줍니다. ResNet-101은 VGG-16에 비해 mAP@[.5, .95]가 6% 증가했는데, 이는 28%의 상대적 개선이며, 이는 전적으로 더 나은 네트워크에서 학습한 특징에 기인합니다. 다시 한 번 주목할 만한 점은 mAP@[.5, .95]의 절대적인 증가율(6.0%)이 mAP@.5의 증가율(6.9%)과 거의 비슷하다는 점입니다. 이는 네트워크가 더 깊어질수록 인식과 로컬라이제이션이 모두 향상될 수 있음을 시사합니다.

B. 물체 감지 개선

완성도를 높이기 위해 대회에서 개선된 사항을 보고합니다. 이러한 개선 사항은 딥 피처를 기반으로 하므로 잔여 학습의 이점을 누릴 수 있습니다.

MS COCO

박스 개선. 박스 개선은 부분적으로 [6]의 반복적 로컬라이제이션을 따릅니다. Faster R-CNN에서 최종 출력은 제안 상자와는 다른 회귀된 상자입니다. 따라서 추론을 위해 회귀 상자에서 새로운 특징을 풀링하고 새로운 분류 점수와 새로운 회귀 상자를 얻습니다. 이 300개의 새로운 예측을 원래의 300개의 예측과 결합합니다. 0.3[8]의 IoU 임계값을 사용하여 예측된 박스의 연합 집합에 비최대 억제(NMS)를 적용한 다음 박스 투표를 실시합니다[6]. 박스 재세분화는 mAP를 약 2점 향상시킵니다(표 9).

글로벌 컨텍스트. 고속 R-CNN 단계에서 글로벌 컨텍스트를 결합합니다. 전체 이미지 변환 피처 맵이 주어지면, 전체 이미지의 경계 상자를 RoI로 사용하여 "RoI" 풀링으로 구현할 수 있는 글로벌 공간 피라미드 풀링[12]("단일 레벨" 피라미드 포함)을 통해 피처를 풀링합니다. 이 풀링된 피처는 포스트 RoI 레이어에 공급되어 글로벌 컨텍스트 피처를 얻습니다. 이 글로벌 피처는 원래의 영역별 피처와 연결되고, 그 다음에는 형제 분류 및 박스 회귀 레이어가 이어집니다. 이 새로운 구조는 엔드투엔드로 학습됩니다. 글로벌 컨텍스트는 mAP@.5를 약 1점 향상시킵니다(표 9).

멀티 스케일 테스트. 위의 모든 결과는 [32]에서와 같이 단일 규모 훈련/테스트로 얻은 것으로, 이미지의 짧은 면은= 600픽셀입니다. 멀티스케일 훈련/테스트는 [12, 7]에서 특징 피라미드에서 스케일을 선택하고 [33]에서 맥아웃 레이어를 사용하여 개발되었습니다. 현재 구현에서는 [33]에 따라 멀티 스케일 *테스트를* 수행했으며, 시간 제약으로 인해 멀티 스케일 훈련은 수행하지 않았습니다. 또한, 고속 R-CNN 단계에 대해서만 수행별 멀티스케일 테스트를 수행했습니다(아직 RPN 단계는 수행하지 않았습니다). 훈련된 모델을 사용하여 이미지의 짧은 변이 $s \in \{200, 400, 600, 800, 1000\}$*인* 이미지 피라미드에서 컨볼루션 피처 맵을 계산합니다.

훈련 데이터	COCO train		COCO trainval	
테스트 데이터	COCO val		COCO test-dev	
mAP	@.5	@[.5, .95]	@.5	@[.5, .95]
기준선 더 빠른 R-CNN(VGG-16)	41.5	21.2		
기준선 더 빠른 R-CNN(ResNet-101)	48.4	27.2		
+박스 개선	49.9	29.9		
+컨텍스트	51.1	30.0	53.3	32.2
+멀티 스케일 테스트	53.8	32.5	**55.7**	**34.9**
앙상블			**59.0**	**37.4**

표 9. Faster R-CNN과 ResNet-101을 사용한 MS COCO의 물체 감지 성능 향상.

시스템	net	데이터	mAP	areo	자전거	새	보트	병	버스	car	고양이	의자	소	테이블	개	말	엠바이크	사람	식물	양	소파	기차	TV
baseline	VGG-16	07+12	73.2	76.5	79.0	70.9	65.5	52.1	83.1	84.7	86.4	52.0	81.9	65.7	84.8	84.6	77.5	76.7	38.8	73.6	73.9	83.0	72.6
기준선	ResNet-101	07+12	76.4	79.8	80.7	76.2	68.3	55.9	85.1	85.3	**89.8**	56.7	87.8	69.4	88.3	88.9	80.9	78.4	41.7	78.6	79.8	85.3	72.0
기준선+++	ResNet-101	COCO+07+12	**85.6**	**90.0**	**89.6**	**87.8**	**80.8**	**76.1**	**89.9**	**89.9**	89.6	**75.5**	**90.0**	**80.7**	**89.6**	**90.3**	**89.1**	**88.7**	**65.4**	**88.1**	**85.6**	**89.0**	**86.8**

표 10. PASCAL VOC 2007 테스트 세트의 탐지 결과. 기준은 Faster R-CNN 시스템입니다. 표 9의 시스템 "베이스라인+++"에는 박스 세분화, 컨텍스트 및 멀티스케일 테스트가 포함됩니다.

시스템	net	데이터	mAP	areo	자전거	새	보트	병	버스	car	고양이	의자	소	테이블	개	말	엠바이크	사람	식물	양	소파	기차	TV
baseline	VGG-16	07++12	70.4	84.9	79.8	74.3	53.9	49.8	77.5	75.9	88.5	45.6	77.1	55.3	86.9	81.7	80.9	79.6	40.1	72.6	60.9	81.2	61.5
기준	ResNet-101	07++12	73.8	86.5	81.6	77.2	58.0	51.0	78.6	76.6	93.2	48.6	80.4	59.0	92.1	85.3	84.8	80.7	48.1	77.3	66.5	84.7	65.6
기준선+++	ResNet-101	COCO+07++12	**83.8**	**92.1**	**88.4**	**84.8**	**75.9**	**71.4**	**86.3**	**87.8**	**94.2**	**66.8**	**89.4**	**69.2**	**93.9**	**91.9**	**90.9**	**89.6**	**67.9**	**88.2**	**76.8**	**90.3**	**80.0**

표 11. PASCAL VOC 2012 테스트 세트의 탐지 결과 (`http://host.robots.ox.ac.uk:8080/leaderboard/displaylb.php?challengeid=11&compid=4`). 기준은 Faster R-CNN 시스템입니다. "기준선+++" 시스템에는 표 9의 박스 세분화, 컨텍스트 및 멀티스케일 테스트가 포함됩니다.

피라미드에서 [33]에 따라 인접한 두 개의 스케일을 선택합니다. 이 두 스케일의 피처 맵에서 RoI 풀링과 후속 레이어가 수행되며 [3333]에서와 같이 최대 아웃으로 병합됩니다[]. 멀티스케일 테스트는 mAP를 2점 이상 향상시킵니다(표 9).

검증 데이터 사용. 다음으로 훈련에는 80,000+40,000 trainval 세트를 사용하고 평가에는 20,000 test-dev 세트를 사용합니다. 테스트 개발 세트에는 공개적으로 사용 가능한 기준 데이터가 없으며 평가 서버에서 결과를 보고합니다. 이 설정에서 결과는 mAP@.5 55.7%, mAP@[.5, .95]는 34.9%입니다(표 9). 이것은 단일 모델 결과입니다.

앙상블. Faster R-CNN에서는 시스템이 영역 제안과 객체 분류기를 학습하도록 설계되었기 때문에 앙상블을 사용하여 두 가지 작업을 모두 향상시킬 수 있습니다. 우리는 영역 제안에 앙상블을 사용하고, 제안의 연합 집합은 영역별 분류기의 앙상블에 의해 처리됩니다. 표 9는 3개 네트워크의 앙상블을 기반으로 한 결과를 보여줍니다. mAP는 59.0%, 테스트-개발 세트에서는 37.4%입니다. *이 결과는 COCO 2015에서 탐지 과제에서 1위를 차지했습니다.*

파스칼 VOC

위의 모델을 기반으로 PASCAL VOC 데이터 세트를 다시 살펴봅니다. COCO 데이터 세트의 단일 모델(표 9의 55.7% mAP@.5)을 사용하여 PASCAL VOC 세트에서 이 모델을 미세 조정합니다. 박스 세분화, 콘텍스트 및 멀티스케일 테스트의 개선 사항도 채택되었습니다. 이렇게 함으로써

	val2	test
GoogLeNet [44] (ILSVRC'14)	-	43.9
단일 모델 (ILSVRC'15)	60.5	58.8
앙상블 (ILSVRC'15)	**63.6**	**62.1**

표 12. ImageNet 탐지 데이터 세트에 대한 결과(mAP, %). 우리의 탐지 시스템은 표 9의 개선 사항을 적용한 더 빠른 R-CNN[32]으로, ResNet-101을 사용합니다.

PASCAL VOC 2007(표 10)에서는 85.6%의 mAP를, PASCAL VOC 2012(표 11)에서는 83.8%를 달성했습니다[6]. PASCAL VOC 2012의 결과는 이전 최신 결과보다 10% 포인트 더 높습니다[6].

이미지넷 탐지

이미지넷 감지(DET) 작업에는 200개의 객체 범주가 포함됩니다. 정확도는 mAP@.5로 평가됩니다. ImageNet DET의 객체 감지 알고리즘은 표 9의 MS COCO의 알고리즘과 동일합니다. 네트워크는 1000클래스 ImageNet 분류 세트에 대해 사전 학습되고 DET 데이터에 대해 미세 조정됩니다. 검증 세트는 [8]에 따라 두 부분(val1/val2)으로 나눕니다. DET 훈련 세트와 val1 세트를 사용하여 탐지 모델을 미세 조정합니다. val2 세트는 유효성 검사에 사용됩니다. 다른 ILSVRC 2015 데이터는 사용하지 않습니다. ResNet-101을 사용한 단일 모델은 다음과 같습니다.

[6] http://host.robots.ox.ac.uk:8080/anonymous/3OJ4OJ.html, 2015-11-26에 제출되었습니다.

LOC method	LOC network	testing	GT CLS의 LOC 오류	분류 네트워크	예측된 CLS의 상위 5위 LOC 오차
VGG [41]	VGG-16	1-crop	33.1 [41]		
RPN	ResNet-101	1-crop	13.3		
RPN	ResNet-101	밀도	11.7		
RPN	ResNet-101	dense		ResNet-101	14.4
RPN+RCNN	ResNet-101	dense		ResNet-101	**10.6**
RPN+RCNN	앙상블	dense		앙상블	**8.9**

표 13. 이미지넷 검증의 현지화 오류(%). "GT 클래스의 LOC 오류"([41]) 열에서는 기준 실측 클래스가 사용됩니다. "테스트" 열에서 "1-crop"은 224× 224 픽셀의 중심 크롭에 대한 테스트를 나타내고, "dense"는 고밀도(완전 컨볼루션) 및 다중 스케일 테스트를 나타냅니다.

58.8% mAP, 3개의 모델로 구성된 앙상블은 DET 테스트 세트에서 62.1% mAP를 기록했습니다(표 12). *이 결과는* 2위를 **8.5점**(절대 점수) 차이로 제치고 *ILSVRC 2015의 ImageNet 탐지 과제에서 1위를 차지했습니다.*

C. 이미지넷 로컬라이제이션

이미지넷 로컬라이제이션(LOC) 작업[36]은 객체를 분류하고 로컬라이즈해야 합니다. 40, 41]에 따라 이미지 레벨 분류기가 이미지의 클래스 레이블을 예측하는 데 먼저 채택되고, 로컬라이제이션 알고리즘은 예측된 클래스를 기반으로 경계 상자를 예측하는 데만 사용된다고 가정합니다. 각 클래스에 대해 바운딩 박스 회귀를 학습하는 "클래스별 회귀"(PCR) 전략[40, 41]을 채택합니다. Im-ageNet 분류를 위해 네트워크를 사전 훈련한 다음 현지화를 위해 미세 조정합니다. 제공된 1000 클래스 Ima-geNet 훈련 세트에서 네트워크를 훈련합니다.

현지화 알고리즘은 [32]의 RPN 프레임워크에 기반하지만 몇 가지 수정 사항이 있습니다. 기존의 방식과 달리
[32]와 같이 카테고리에 구애받지 않는 현지화를 위한 RPN은 *클래스별* 형태로 설계되었습니다. 이 RPN은 [32]에서와 같이 이진 분류(*cls*)와 박스 회귀(*reg*)를 위한 두 개의 형제자매 1× 1 컨볼루션 레이어로 끝납니다. cls와 *reg* 레이어는 모두 [32]와 달리 *클래스 단위로* 구성됩니다. 구체적으로, *cls* 계층은 1000-d 출력을 가지며 각 차원은 객체 클래스의 존재 여부를 예측하기 위한 *이진 로지스틱 회귀이고*, *reg* 계층은 1000개의 클래스에 대한 박스 회귀로 구성된 1000× 4-d 출력을 가집니다. 32]에서와 마찬가지로, 바운딩 박스 회귀는 각 위치에서 여러 개의 번역 불변 "앵커" 박스를 참조합니다.

이미지넷 분류 훈련(3.4절)에서와 마찬가지로, 데이터 증강을 위해 224개의× 224개의 크롭을 무작위로 샘플링합니다. 미세 조정을 위해 256개 이미지의 미니 배치 크기를 사용합니다. 부정적인 샘플이 우세해지는 것을 방지하기 위해 각 이미지에 대해 8개의 앵커를 런덤 샘플링하며, 샘플링된 양성 및 음성 앵커의 비율은 1:1입니다[32]. 테스트를 위해 네트워크는 이미지에 완전 컨볼루션 방식으로 적용됩니다.

표 13은 로컬라이제이션 결과를 비교한 것입니다. 먼저 [41]에 따라 기준 진실 클래스를 분류 예측으로 사용하여 "오라클" 테스트를 수행합니다. VGG의 논문 [41]을 다시 참조합니다.

메서드	TOP-5 현지화 오류	
	val	test
오버피트 [40] (ILSVRC'13)	30.0	29.9
GoogLeNet [44] (ILSVRC'14)	-	26.7
VGG [41] (ILSVRC'14)	26.9	25.3
우리 (ILSVRC'15)	**8.9**	**9.0**

표 14. ImageNet 데이터 세트의 로컬라이제이션 오차(%)와 최신 방법의 비교.

는 기준값 클래스를 사용하여 33.1%의 중심 자르기 오류를 포팅합니다(표 13). 동일한 설정에서 ResNet-101 넷을 사용하는 RPN 방법은 센터 크롭 오류를 13.3%로 크게 줄였습니다. 이 비교는 프레임워크의 뛰어난 성능을 보여줍니다. 고밀도(완전 컨볼루션) 및 멀티스케일 테스트에서 기준값 클래스를 사용하는 ResNet-101의 오류는 11.7%입니다. 클래스를 예측하는 데 ResNet-101을 사용하면(상위 5위 분류 오류 4.6%, 표 4), 상위 5위 지역화 오류는 14.4%입니다.

위의 결과는 Faster R-CNN [32]의 *제안 네트워크*(RPN)만을 기준으로 한 것입니다. 결과를 개선하기 위해 Faster R-CNN의 *감지 네트워크*(Fast R-CNN [7])를 사용할 수도 있습니다. 그러나 이 데이터 세트에서 하나의 이미지에는 일반적으로 하나의 지배 객체가 포함되어 있고 제안 영역이 서로 매우 겹쳐서 매우 유사한 RoI 풀링 특징을 가지고 있음을 알 수 있습니다. 그 결과, Fast R-CNN[7]의 이미지 중심 훈련은 확률론적 훈련에 적합하지 않을 수 있는 작은 변화의 샘플을 생성합니다. 이 점에 착안하여 현재 실험에서는 Fast R-CNN 대신 RoI 중심인 오리지널 R-CNN[8]을 사용합니다.

R-CNN 구현은 다음과 같습니다. 위와 같이 학습된 클래스별 RPN을 학습 이미지에 적용하여 기준값 클래스에 대한 바운딩 박스를 예측합니다. 이렇게 예측된 박스는 클래스 종속 제안의 역할을 합니다. 각 훈련 이미지에 대해 가장 높은 점수를 받은 200개의 제안을 훈련 샘플로 추출하여 R-CNN 분류자를 훈련합니다. 이미지 영역은 제안에서 잘라내어 224× 224픽셀로 워핑한 다음 R-CNN [8]에서와 같이 분류 네트워크에 공급합니다. 이 네트워크의 출력은 클래스별 형태로 cls와 *reg에* 대한 두 개의 형제 fc 레이어로 구성됩니다. 이 R-CNN 네트워크는 RoI 중심 방식으로 256개의 미니 배치 크기를 사용하여 훈련 세트에서 미세 조정됩니다. 테스트를 위해 RPN은 예측된 각 클래스에 대해 가장 높은 점수를 받은 200개의 제안을 생성하고, R-CNN 네트워크는 이러한 제안의 점수와 박스 위치를 업데이트하는 데 사용됩니다.

이 방법을 사용하면 상위 5개의 로컬라이제이션 오류가 10.6%로 감소합니다(표 13). 이것은 검증 세트에 대한 단일 모델 결과입니다. 클래스화 및 로컬라이제이션 모두에 네트워크 앙상블을 사용하면 테스트 세트에서 9.0%의 상위 5위 로컬라이제이션 오류를 달성합니다. 이 수치는 ILSVRC 14 결과(표 14)를 크게 능가하는 것으로, 64%의 상대적 오류 감소를 보여줍니다. *이 결과는 ILSVRC 2015의 ImageNet 로컬라이제이션 과제에서 1위를 차지했습니다.*