컨볼루션으로 더 깊이 들어가기

크리스티안 세게디
Google Inc.

리우 웨이
노스캐롤라이나 대학교, 채플 힐

Yangqing Jia
Google Inc.

피에르 세르마네
구글 주식회사

스콧 리드
미시간 대학교

Dragomir Anguelov
Google Inc.

두미트루 에르한
구글 주식회사

빈센트 반후케
구글 주식회사

앤드류 라비노비치
구글 주식회사

추상적인

우리는 ILSVRC14(ImageNet Large-Scale Visual Recognition Challenge 2014)에서 분류 및 감지를 위한 새로운 최신 기술을 설정한 코드명 Inception이라는 심층 컨벌루션 신경망 아키텍처를 제안합니다. 이 아키텍처의 주요 특징은 네트워크 내부의 컴퓨팅 리소스 활용도가 향상되었다는 것입니다. 이것은 계산 예산을 일정하게 유지하면서 네트워크의 깊이와 폭을 증가시킬 수 있는 세심하게 만들어진 설계에 의해 달성되었습니다. 품질을 최적화하기 위해 아키텍처 결정은 Hebbian 원칙과 다중 규모 처리의 직관을 기반으로 했습니다. ILSVRC14에 대한 제출에 사용된 특정 화신은 GoogLeNet이라고 하는 22계층 심층 네트워크로 분류 및 탐지의 맥락에서 품질이 평가됩니다.

1. 소개

지난 3년 동안, 주로 딥 러닝, 보다 구체적으로 컨볼루션 네트워크 [10] 의 발전으로 인해 이미지 인식 및 객체 감지의 품질이 극적인 속도로 발전했습니다. 한 가지 고무적인 소식은 이러한 발전의 대부분이 더 강력한 하드웨어, 더 큰 데이터 세트 및 더 큰 모델의 결과일 뿐만 아니라 주로 새로운 아이디어, 알고리즘 및 개선된 네트워크 아키텍처의 결과라는 것입니다. 예를 들어 ILSVRC 2014 대회의 상위 항목은 탐지 목적으로 동일한 대회의 분류 데이터 세트 외에 새로운 데이터 소스를 사용하지 않았습니다. ILSVRC 2014에 대한 우리의 GoogLeNet 제출은 실제로 2년 전 Krizhevsky et al [9] 의 우승 아키텍처보다 12배 더 적은 매개변수를 사용하면서도 훨씬 더 정확합니다. 객체 감지에서 가장 큰 이득은 심층 네트워크 또는 더 큰 모델의 활용에서 비롯된 것이 아니라 Girshick et al [6]의 R-CNN 알고리즘과 같은 심층 아키텍처와 고전적인 컴퓨터 비전의 시너지 효과에서 비롯되었습니다 .

또 다른 주목할만한 요소는 모바일 및 임베디드 컴퓨팅의 지속적인 견인으로 알고리즘의 효율성, 특히 전력 및 메모리 사용이 중요해지고 있다는 것입니다. 이 백서에 제시된 심층 아키텍처 설계로 이어지는 고려 사항에는 정확도 수치에 대한 순전한 고정이 아니라 이 요소가 포함되어 있다는 점은 주목할 만합니다. 대부분의 실험에서 모델은 추론 시간에 15억 곱하기 덧셈의 계산 예산을 유지하도록 설계되었으므로 순수한 학문적 호기심으로 끝나지 않고 실제 사용에 사용할 수 있습니다. 합리적인 비용으로 대규모 데이터 세트에서.

이 백서에서는 코드명 Inception이라는 컴퓨터 비전을 위한 효율적인 심층 신경망 아키텍처에 중점을 둘 것입니다. Inception은 유명한 "우리는 더 깊이 들어가야 합니다"와 함께 Lin et al [12] 의 Network in network 논문에서 이름을 따왔습니다. 인터넷 밈 [1]. 우리의 경우 "딥"이라는 단어는 두 가지 다른 의미로 사용됩니다. 첫째, "인셉션 모듈"의 형태로 새로운 수준의 조직을 도입한다는 의미와 증가된 네트워크라는 보다 직접적인 의미에서 사용됩니다. 깊이. 일반적으로 Inception 모델은 Arora et al[2]의 이론적 작업에서 영감과 지침을 얻으면서 [12] 의 논리적 정점으로 볼 수 있습니다 . 아키텍처의 이점은 ILSVRC 2014 분류 및 감지 과제에서 실험적으로 검증되었으며 현재 최신 기술보다 훨씬 뛰어납니다.

2 관련 업무

LeNet-5 [10] 부터 CNN(컨볼루션 신경망)은 일반적으로 스택형 컨볼루션 레이어(선택적으로 콘트라스트 정규화 및 최대 풀링이 뒤따름) 다음에 하나 이상의 완전 연결 레이어가 오는 표준 구조를 가졌습니다. 이 기본 디자인의 변형은 이미지 분류 문헌에서 널리 퍼져 있으며 MNIST, CIFAR, 특히 ImageNet 분류 챌린지 [9, 21]에서 가장 뛰어난 결과를 얻었습니다. Imagenet과 같은 더 큰 데이터 세트의 경우 최근 추세는 레이어 수 [12] 와 레이어 크기 [21, 14] 를 늘리는 반면 드롭아웃 [7]을 사용하여 과적합 문제를 해결하는 것입니다.

최대 풀링 레이어로 인해 정확한 공간 정보가 손실된다는 우려에도 불구하고 [9] 와 동일한 컨볼루션 네트워크 아키텍처가 지역화 [9, 14], 물체 감지 [6, 14, 18, 5] 및 인간에게도 성공적으로 사용되었습니다. 포즈 추정 [19]. 영장류 시각 피질의 신경과학 모델에서 영감을 받은 Serre et al. [15] Inception 모델과 유사하게 여러 척도를 처리하기 위해 크기가 다른 일련의 고정 Gabor 필터를 사용합니다. 그러나 [15] 의 fixed 2-layer deep 모델과 달리 Inception 모델의 모든 필터가 학습됩니다. 게다가 인셉션 레이어가 여러 번 반복되어 GoogLeNet 모델의 경우 22 레이어 딥 모델로 이어집니다.

Network-in-Network는 Lin 등이 제안한 접근 방식입니다. [12] 신경망의 표현력을 높이기 위해. 컨볼루션 레이어에 적용할 때 이 방법은 일반적으로 정류된 선형 활성화가 뒤따르는 추가 1×1 컨볼루션 레이어로 볼 수 있습니다 [9]. 이를 통해 현재 CNN 파이프라인에 쉽게 통합할 수 있습니다. 우리는 아키텍처에서 이 접근 방식을 많이 사용합니다. 그러나 우리 설정에서 1 × 1 컨볼루션은 두 가지 목적을 가지고 있습니다. 가장 중요한 것은 주로 네트워크 크기를 제한하는 계산 병목 현상을 제거하기 위한 차원 축소 모듈로 사용된다는 것입니다. 이를 통해 상당한 성능 저하 없이 네트워크의 깊이뿐만 아니라 폭도 증가할 수 있습니다.

객체 감지를 위한 현재 선도적인 접근 방식은 Girshick 등이 제안한 R-CNN(Regions with Convolutional Neural Networks)입니다. [6]. R-CNN은 전체 감지 문제를 두 가지 하위 문제로 분해합니다. 먼저 범주에 구애받지 않는 방식으로 잠재적 개체 제안에 대한 색상 및 슈퍼픽셀 일관성과 같은 낮은 수준의 단서를 활용한 다음 CNN 분류기를 사용하여 해당 위치에서 개체 범주를 식별합니다. 이러한 2단계 접근 방식은 최신 CNN의 매우 강력한 분류 기능뿐만 아니라 낮은 수준의 단서가 있는 경계 상자 분할의 정확성을 활용합니다. 우리는 감지 제출에서 유사한 파이프라인을 채택했지만 더 높은 개체 경계 상자 리콜을 위한 다중 상자 [5] 예측 및 경계 상자 제안의 더 나은 분류를 위한 앙상블 접근 방식과 같은 두 단계 모두에서 개선 사항을 탐색했습니다.

3 동기 부여 및 높은 수준의 고려 사항

심층 신경망의 성능을 향상시키는 가장 간단한 방법은 크기를 늘리는 것입니다. 여기에는 네트워크의 깊이(레벨 수) 증가와 폭(각 레벨의 단위 수)이 모두 포함됩니다. 이것은 특히 많은 양의 레이블이 지정된 교육 데이터를 사용할 수 있는 경우 고품질 모델을 교육하는 쉽고 안전한 방법입니다. 그러나이 간단한 솔루션에는 두 가지 주요 단점이 있습니다.

일반적으로 크기가 클수록 매개변수의 수가 많아져 확장된 네트워크가 과적합되기 쉽습니다.

고품질 트레이닝 세트를 생성하는 것이 까다로울 수 있으므로 이는 주요 병목 현상이 될 수 있습니다.

(a) 시베리안 허스키

(b) 에스키모 개

그림 1: ILSVRC 2014 분류 챌린지의 1000개 클래스 중 두 개의 다른 클래스.

특히 그림 1에서 볼 수 있듯이 ImageNet(1000 클래스 ILSVRC 하위 집합에서도)과 같은 세분화된 시각적 범주를 구별하기 위해 전문 평가자가 필요한 경우 비용이 많이 듭니다.

균일하게 증가된 네트워크 크기의 또 다른 단점은 계산 리소스의 사용이 급격히 증가한다는 것입니다. 예를 들어, 딥 비전 네트워크에서 두 개의 컨벌루션 레이어가 연결된 경우 필터 수가 균일하게 증가하면 2차 계산이 증가합니다. 추가된 용량이 비효율적으로 사용되는 경우(예: 대부분의 가중치가 0에 가까워지는 경우) 많은 계산이 낭비됩니다. 실제로 계산 예산은 항상 한정되어 있기 때문에 주요 목표가 결과의 품질을 높이는 것일지라도 무차별적인 크기 증가보다 컴퓨팅 리소스의 효율적인 분배가 선호됩니다.

두 가지 문제를 해결하는 근본적인 방법은 궁극적으로 컨볼루션 내부에서도 완전히 연결된 아키텍처에서 드물게 연결된 아키텍처로 이동하는 것입니다. 생물학적 시스템을 모방하는 것 외에도 Arora et al. [2]. 그들의 주요 결과는 데이터 세트의 확률 분포가 크고 매우 희소한 심층 신경망으로 표현될 수 있다면 최적의 네트워크 토폴로지는 마지막 계층의 활성화에 대한 상관 통계를 분석하여 계층별로 구성할 수 있다는 것입니다. 높은 상관 출력을 가진 클러스터링 뉴런. 엄격한 수학 증명에는 매우 강력한 조건이 필요하지만, 이 진술이 잘 알려진 Hebbian 원리(함께 작동하는 뉴런, 함께 연결되는 뉴런)와 공명한다는 사실은 기본 아이디어가 실제로는 덜 엄격한 조건에서도 적용될 수 있음을 시사합니다.

단점은 오늘날의 컴퓨팅 인프라가 균일하지 않은 희소 데이터 구조에 대한 수치 계산과 관련하여 매우 비효율적이라는 것입니다. 산술 연산의 수가 100배 줄어들더라도 조회 및 캐시 미스의 오버헤드가 너무 지배적이어서 희소 행렬로 전환해도 효과가 없을 것입니다. 기본 CPU 또는 GPU 하드웨어 [16, 9] 의 미세한 세부 사항을 활용하여 매우 빠른 고밀도 매트릭스 곱셈을 허용하는 꾸준히 개선되고 고도로 조정된 수치 라이브러리를 사용하면 격차가 더욱 넓어집니다. 또한 비균일 희소 모델에는 보다 정교한 엔지니어링 및 컴퓨팅 인프라가 필요합니다. 현재의 대부분의 비전 지향 기계 학습 시스템은 컨볼루션을 사용하는 덕분에 공간 영역에서 희소성을 활용합니다. 그러나 컨볼루션은 이전 계층의 패치에 대한 조밀한 연결 모음으로 구현됩니다. ConvNets는 전통적으로 대칭성을 깨고 학습을 향상시키기 위해 [11] 부터 기능 차원에서 무작위 및 희소 연결 테이블을 사용했으며 , 추세는 병렬 컴퓨팅을 더 잘 최적화하기 위해 [9] 와 함께 전체 연결로 다시 변경되었습니다. 구조의 균일성과 많은 수의 필터 및 더 큰 배치 크기를 통해 효율적인 고밀도 계산을 활용할 수 있습니다.

이것은 다음 중간 단계에 대한 희망이 있는지에 대한 질문을 제기합니다. 이론에서 제안한 것처럼 필터 수준에서도 여분의 희소성을 사용하지만 우리의

조밀한 행렬에 대한 계산을 활용하여 현재 하드웨어. 희소 행렬 계산에 관한 방대한 문헌(예: [3])은 희소 행렬을 상대적으로 조밀한 부분 행렬로 클러스터링하는 것이 희소 행렬 곱셈에 대한 최첨단 실제 성능을 제공하는 경향이 있음을 시사합니다. 가까운 미래에 균일하지 않은 딥 러닝 아키텍처의 자동화된 구성에 유사한 방법이 활용될 것이라고 생각하는 것은 무리가 아닌 것 같습니다.

인셉션 아키텍처 는 비전 네트워크에 대해 [2] 에 의해 암시된 희소 구조를 근사화하려고 시도하는 정교한 네트워크 토폴로지 구성 알고리즘의 가상 출력을 평가하고 가정된 결과를 조밀하고 쉽게 사용 가능한 구성 요소. 매우 투기적인 작업임에도 불구하고 정확한 토폴로지 선택에 대한 두 번의 반복 후에야 [12] 에 기반한 참조 아키텍처에 비해 약간의 이득을 볼 수 있었습니다 . 학습 속도, 하이퍼파라미터 및 개선된 교육 방법론을 추가로 조정한 후 결과 Inception 아키텍처가 [6] 및 [5] 의 기본 네트워크로서 지역화 및 객체 감지의 맥락에서 특히 유용하다는 것을 확인했습니다. 흥미롭게도 대부분의 원래 아키텍처 선택에 대해 철저히 조사하고 테스트했지만 최소한 국부적으로는 최적인 것으로 판명되었습니다.

그러나 신중해야 합니다. 제안된 아키텍처가 컴퓨터 비전에서 성공을 거두었지만 그 품질이 구성으로 이어진 지침 원칙에 기인할 수 있는지 여부는 여전히 의문입니다. 이를 확인하려면 훨씬 더 철저한 분석과 검증이 필요합니다. 예를 들어, 아래에 설명된 원칙에 기반한 자동화 도구가 유사하지만 비전 네트워크에 대한 더 나은 토폴로지를 찾는 경우입니다. 가장 설득력 있는 증거는 자동화된 시스템이 네트워크 토폴로지를 생성하여 동일한 알고리즘을 사용하지만 매우 다르게 보이는 글로벌 아키텍처를 사용하는 다른 도메인에서 유사한 이점을 얻을 수 있는지 여부입니다. 적어도 Inception 아키텍처의 초기 성공은 이러한 방향으로 흥미로운 미래 작업에 대한 확고한 동기를 부여합니다.

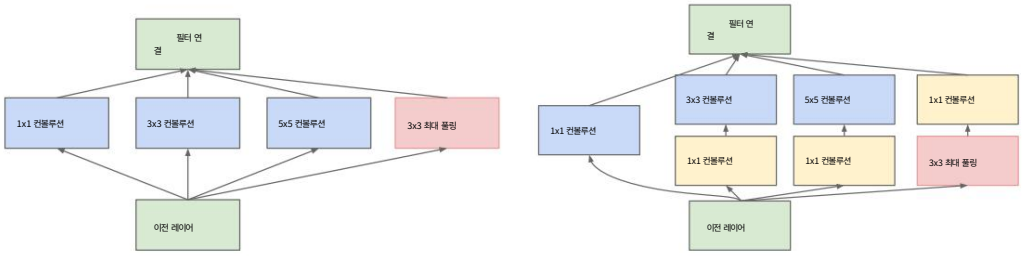
4 건축 세부 사항

Inception 아키텍처의 주요 아이디어는 컨볼루션 비전 네트워크에서 최적의 로컬 스파스 구조를 근사화하고 쉽게 사용할 수 있는 고밀도 구성 요소로 덮을 수 있는 방법을 찾는 데 기반합니다. 변환 불변성을 가정한다는 것은 네트워크가 컨볼루션 빌딩 블록으로 구축된다는 것을 의미합니다. 우리에게 필요한 것은 최적의 로컬 구성을 찾고 이를 공간적으로 반복하는 것입니다. Aroraet al. [2]는 마지막 레이어의 상관 통계를 분석하고 높은 상관 관계를 가진 단위 그룹으로 클러스터링해야 하는 레이어별 구성을 제안합니다.

이러한 클러스터는 다음 계층의 단위를 형성하고 이전 계층의 단위에 연결됩니다. 이전 레이어의 각 단위는 입력 이미지의 일부 영역에 해당하고 이러한 단위는 필터 뱅크로 그룹화된다고 가정합니다. 하위 계층(입력에 가까운 계층)에서는 상관 단위가 로컬 영역에 집중됩니다. 즉, [12] 에서 제안한 것처럼 단일 영역에 집중된 많은 클러스터로 끝나고 다음 레이어에서 1×1 컨볼루션 레이어로 덮을 수 있습니다. 그러나 더 큰 패치에 걸쳐 컨볼루션으로 덮을 수 있는 더 적은 수의 공간적으로 분산된 클러스터가 있고 더 크고 더 큰 영역에 걸쳐 패치 수가 감소할 것이라고 예상할 수도 있습니다. 패치 정렬 문제를 피하기 위해 Inception 아키텍처의 현재 화신은 필터 크기 1×1, 3×3 및 5×5로 제한되지만 이 결정은 필요성보다는 편의성에 더 기반합니다. 또한 제안된 아키텍처는 다음 단계의 입력을 형성하는 단일 출력 벡터로 연결되는 출력 필터 뱅크가 있는 모든 레이어의 조합임을 의미합니다. 또한, 풀링 작업은 최신 컨볼루션 네트워크의 성공에 필수적이기 때문에 이러한 각 단계에서 대체 병렬 풀링 경로를 추가하면 추가적인 유익한 효과도 있어야 합니다(그림 2(a) 참조).

이러한 "인셉션 모듈"이 서로 위에 쌓이면 출력 상관 통계가 달라질 수밖에 없습니다. 더 높은 추상화의 기능이 더 높은 계층에 의해 캡처됨에 따라 공간 집중도가 감소할 것으로 예상되어 3×3 및 상위 계층으로 이동함에 따라 5×5 컨볼루션이 증가해야 합니다.

적어도 이 순진한 형태에서 위 모듈의 한 가지 큰 문제는 많은 수의 필터가 있는 컨볼루션 레이어 위에 적당한 수의 5×5 컨볼루션이 엄청나게 비쌀 수 있다는 것입니다. 이 문제는 풀링 장치가 믹스에 추가되면 더욱 두드러집니다. 출력 필터의 수는 이전 단계의 필터 수와 같습니다. 풀링 레이어의 출력과 컨볼루션 레이어의 출력을 병합하면 필연적으로



(a) 인셉션 모듈, 순진한 버전 (b) 차원 축소가 있는 시작 모듈

그림 2: 인셉션 모듈

단계에서 단계로의 출력 수 증가. 이 아키텍처가 최적의 희소 구조를 커버할 수 있지만 매우 비효율적으로 수행하여 몇 단계 내에서 계산 폭증으로 이어집니다.

이것은 제안된 아키텍처의 두 번째 아이디어로 이어집니다. 계산 요구 사항이 너무 많이 증가할 때마다 차원 축소 및 예측을 신중하게 적용하는 것입니다.
이것은 임베딩의 성공을 기반으로 합니다. 낮은 차원의 임베딩이라도 상대적으로 큰 이미지 패치에 대한 많은 정보를 포함할 수 있습니다. 그러나 임베딩은 조밀하고 압축된 형태로 정보를 나타내며 압축된 정보는 모델링하기가 더 어렵습니다. 우리는 ([2]의 조건에서 요구하는 대로) 대부분의 장소에서 우리의 표현을 희소하게 유지하고 신호가 한꺼번에 집계되어야 할 때만 신호를 압축하려고 합니다. 즉, 값비싼 3×3 및 5×5 컨볼루션 이전에 감소를 계산하는 데 1×1 컨볼루션이 사용됩니다. 감소로 사용되는 것 외에도 정류된 선형 활성화의 사용을 포함하여 이중 목적으로 만듭니다. 최종 결과는 그림 2(b)에 나와 있습니다.

일반적으로 Inception 네트워크는 위 유형의 모듈이 서로 쌓이는 네트워크로, 간헐적으로 그리드 해상도를 절반으로 줄이기 위해 보폭이 2인 최대 풀링 레이어가 있습니다. 기술적인 이유(훈련 중 메모리 효율성)로 인해 하위 계층을 전통적인 컨볼루션 방식으로 유지하면서 상위 계층에서만 Inception 모듈을 사용하는 것이 유리해 보였습니다.

이것은 반드시 필요한 것은 아니며 단순히 현재 구현의 일부 인프라 비효율성을 반영합니다.

이 아키텍처의 주요 이점 중 하나는 제어되지 않은 계산 복잡성의 폭발 없이 각 단계에서 단위 수를 크게 늘릴 수 있다는 것입니다. 차원 축소의 유비쿼터스 사용을 통해 마지막 단계의 많은 수의 입력 필터를 다음 계층으로 차폐할 수 있으며, 큰 패치 크기로 필터를 컨볼루션하기 전에 먼저 차원을 줄입니다. 이 디자인의 또 다른 실질적으로 유용한 측면은 시각적 정보가 다양한 스케일에서 처리된 다음 다음 단계에서 동시에 다른 스케일의 기능을 추상화할 수 있도록 집계되어야 한다는 직관과 일치한다는 것입니다.

계산 리소스의 개선된 사용으로 인해 계산상의 어려움 없이 각 단계의 너비와 단계 수를 모두 늘릴 수 있습니다. 인셉션 아키텍처를 활용하는 또 다른 방법은 약간 열등하지만 계산상 더 저렴한 버전을 만드는 것입니다. 우리는 포함된 모든 손잡이와 레버가 비 Inception 아키텍처를 사용하는 유사한 성능의 네트워크보다 2 - 3배 더 빠른 네트워크를 생성할 수 있는 계산 리소스의 제어된 균형을 허용한다는 것을 발견했습니다. 그러나 이 시점에서 신중한 수동 설계가 필요합니다. .

5 구글넷

우리는 ILSVRC14 대회에서 팀 이름으로 GoogLeNet을 선택했습니다. 이 이름은 LeNet 5 네트워크를 개척한 Yann LeCuns에게 경의를 표합니다 [10]. 우리는 또한 GoogLeNet을 사용하여 대회 제출에 사용된 Inception 아키텍처의 특정 화신을 참조합니다. 우리는 또한 품질이 약간 열등한 더 깊고 더 넓은 Inception 네트워크를 사용했지만 앙상블에 추가하면 결과가 약간 개선되는 것처럼 보였습니다. 실험에서 정확한 아키텍처 매개변수의 영향이 상대적으로

유형	패치 크기/보폭	출력 크기	깊이 #1×1		#3×3 감소	#3×3	#5×5 감소	#5×5	수영장 프로젝트	매개변수	작전
회선	7×7/2	112×112×64	1							2.7K	34분
최대 풀	3×3/2	56×56×64	0								
회선	3×3/1	56×56×192	2		64	192				112K	360m
최대 풀	3×3/2	28×28×192	0								
시작(3a) 시작(3b)		28×28×256	2	64	96	128	16	32	32	159K	1억 2800만
		28×28×480	2	128	128	192	32	96	64	38만	304M
최대 풀	3×3/2	14×14×480	0								
시작 (4a)		14×14×512	2	192	96	208	16	48	64	364K	73M
시작 (4b)		14×14×512	2	160	112	224	24	64	64	437K	88M
시작 (4c)		14×14×512	2	128	128	256	24	64	64	463K	100M
시작(4d) 시작(4e)		14×14×528	2	112	144	288	32	64	64	580K	1억 1900만
		14×14×832	2	256	160	320	32	128	128	840K	170M
최대 풀	3×3/2	7×7×832	0								
시작 (5a)		7×7×832	2	256	160	320	32	128	128	1072K	54M
시작 (5b)		7×7×1024	2	384	192	384	48	128	128	1388K	71분
평균 풀	7×7/1	1×1×1024	0								
탈락(40%)		1×1×1024	0								
선의		1×1×1000	1							1000K	1M
소프트맥스		1×1×1000	0								

표 1: Inception 아키텍처의 GoogLeNet 화신

미성년자. 여기에서 가장 성공적인 특정 인스턴스(GoogLeNet이라고 함)는 데모용으로 표 1에 설명되어 있습니다. 앙상블의 7개 모델 중 6개 모델에 정확히 동일한 토폴로지(서로 다른 샘플링 방법으로 훈련됨)가 사용되었습니다.

Inception 모듈 내부를 포함하여 모든 컨볼루션은 수정된 선형 활성화를 사용합니다.
우리 네트워크에서 수용 필드의 크기는 224×224이며 평균 감산으로 RGB 색상 채널을 취합니다. "#3×3 reduce" 및 "#5×5 reduce"는 3×3 및 5×5 컨볼루션 이전에 사용된 감소 레이어의 1×1 필터 수를 나타냅니다. 풀 proj 열에 내장된 최대 풀링 후 투영 레이어에서 1×1 필터의 수를 볼 수 있습니다. 이러한 모든 감소/투영 레이어는 정류된 선형 활성화도 사용합니다.

이 네트워크는 계산 효율성과 실용성을 염두에 두고 설계되었으므로 계산 리소스가 제한된 장치, 특히 메모리 공간이 적은 장치를 포함하여 개별 장치에서 추론을 실행할 수 있습니다. 매개변수가 있는 계층만 계산할 때 네트워크의 깊이는 22개 계층입니다(또는 풀링도 포함하는 경우 27개 계층). 네트워크 구성에 사용되는 전체 레이어(독립적 빌딩 블록) 수는 약 100개입니다. 그러나 이 수는 사용되는 기계 학습 인프라 시스템에 따라 다릅니다. 분류기 전에 평균 풀링을 사용하는 것은 [12]를 기반으로 하지만 추가 선형 레이어를 사용한다는 점에서 구현이 다릅니다. 이를 통해 다른 레이블 세트에 대한 네트워크를 쉽게 조정하고 미세 조정할 수 있지만 대부분 편리하며 큰 영향을 미치지 않을 것으로 예상됩니다. 완전 연결 레이어에서 평균 풀링으로 이동하면 top-1 정확도가 약 0.6% 향상되었지만, 완전 연결 레이어를 제거한 후에도 드롭아웃 사용은 여전히 필수적이었습니다.

네트워크의 상대적으로 깊은 깊이를 감안할 때 효과적인 방식으로 모든 레이어를 통해 그래디언트를 다시 전파하는 기능이 문제였습니다. 한 가지 흥미로운 통찰은 이 작업에서 비교적 얕은 네트워크의 강력한 성능이 네트워크 중간에 있는 레이어에서 생성된 기능이 매우 차별적이어야 함을 시사한다는 것입니다. 이러한 중간 레이어에 연결된 보조 분류기를 추가함으로써 분류기의 하위 단계에서 차별을 장려하고 다시 전파되는 그래디언트 신호를 증가시키고 추가 정규화를 제공할 것으로 기대할 수 있습니다. 이러한 분류기는 Inception (4a) 및 (4d) 모듈의 출력 위에 놓인 더 작은 컨볼루션 네트워크의 형태를 취합니다. 학습하는 동안 그들의 손실은 네트워크의 총 손실에 할인 가중치를 적용하여 추가됩니다(보조 분류기의 손실에는 0.3의 가중치가 적용됨). 추론 시 이러한 보조 네트워크는 폐기됩니다.

보조 분류기를 포함한 측면의 추가 네트워크의 정확한 구조는 다음과 같습니다.

- 필터 크기가 5×5이고 보폭이 3인 평균 풀링 레이어는 (4a)에 대해 4×4×512 출력을, (4d) 단계에 대해 4×4×528을 생성합니다.

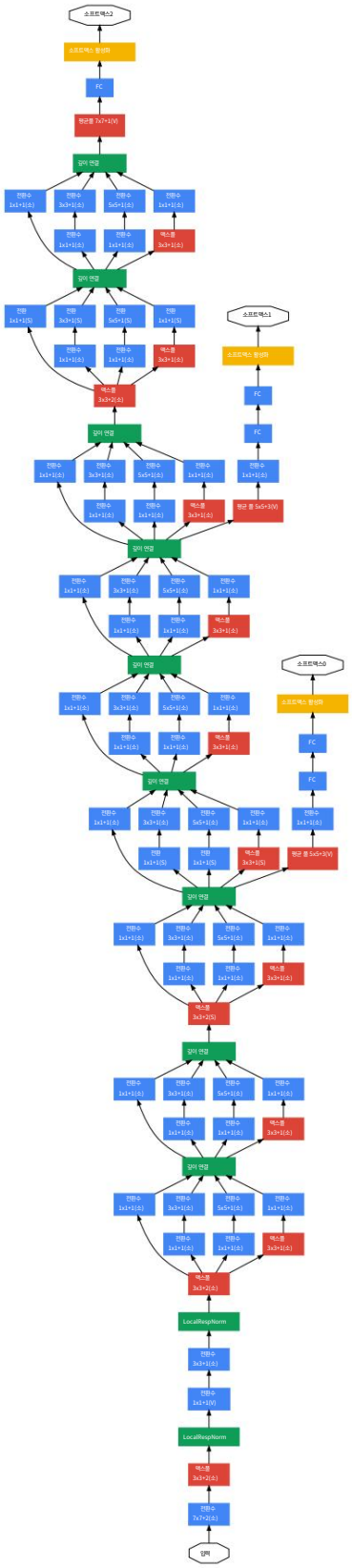


그림 3: 모든 부가 기능이 포함된 GoogLeNet 네트워크

• 차원 축소 및 수정된 선형 활성화를 위한 128개의 필터가 있는 1×1 컨볼루션.

• 1024개의 단위와 정류된 선형 활성화가 있는 완전히 연결된 계층. • 드롭아웃 출력 비율이 70%인

드롭아웃 레이어. • 분류기로 소프트맥스 손실이 있는 선형 계층(동일한

1000개 클래스를 예측
주 분류자이지만 추론 시 제거됨).

결과 네트워크의 개략도가 그림 3에 나와 있습니다.

6 교육 방법론

우리의 네트워크는 적당한 양의 모델과 데이터 병렬성을 사용하는 DistBelief [4] 분산 기계 학습 시스템을 사용하여 훈련되었습니다. 우리는 CPU 기반 구현만 사용했지만 대략적인 추정에 따르면 GoogLeNet 네트워크는 메모리 사용량이 주요 제한 사항인 몇 가지 고급 GPU를 사용하여 수렴에 대해 일주일 내에 훈련될 수 있습니다. 우리의 교육은 0.9 모멘텀 [17], 고정 학습 속도 일정(8 epoch마다 학습 속도를 4%씩 감소)으로 비동기 확률적 경사 하강법을 사용했습니다. Polyak 평균화 [13]는 추론 시간에 사용되는 최종 모델을 만드는 데 사용되었습니다.

우리의 이미지 샘플링 방법은 경쟁에 이르기까지 몇 달 동안 상당히 변경되었으며 이미 수렴된 모델은 다른 옵션으로 훈련되었으며 때로는 드롭아웃 및 학습률과 같은 변경된 하이퍼 매개변수와 함께 훈련되었습니다. 이러한 네트워크를 훈련시키는 가장 효과적인 단일 방법입니다. 문제를 더 복잡하게 만드는 것은 [8] 에서 영감을 받아 모델 중 일부는 주로 상대적으로 작은 작물에 대해 훈련되었고 다른 모델은 더 큰 작물에 대해 훈련되었습니다 .

그래도 경쟁 후 매우 잘 작동하는 것으로 확인된 한 가지 처방에는 크기가 이미지 영역의 8%에서 100% 사이에 고르게 분포되고 종횡비가 3/4에서 3/4 사이에서 임의로 선택되는 이미지의 다양한 크기 패치 샘플링이 포함됩니다. 4/3. 또한 우리는 Andrew Howard [8] 의 측광 왜곡이 과적합을 어느 정도 방지하는 데 유용하다는 것을 발견했습니다. 또한 상대적으로 늦게 크기를 조정하고 다른 하이퍼파라미터 변경과 함께 무작위 보간 방법(쌍선형, 영역, 가장 가까운 이웃 및 입방체, 동일한 확률 포함)을 사용하기 시작했기 때문에 최종 결과가 그들의

사용.

7 ILSVRC 2014 분류 챌린지 설정 및 결과

ILSVRC 2014 분류 챌린지는 이미지를 Imagenet 계층 구조에서 1000개의 리프 노드 범주 중 하나로 분류하는 작업을 포함합니다. 학습용 이미지는 약 120만 개, 검증용 이미지는 50,000개, 테스트용 이미지는 100,000개입니다. 각 이미지는 하나의 실측 범주와 연관되며 성능은 가장 높은 점수를 받은 분류기 예측을 기반으로 측정됩니다.

일반적으로 두 개의 숫자가 보고됩니다. 첫 번째 예측 클래스와 비교하는 top-1 정확도 비율과 처음 5개의 예측 클래스와 비교하는 top-5 오류율: 이미지가 올바르게 분류된 것으로 간주됩니다. 지상 진실이 순위에 관계없이 상위 5개 안에 속하는 경우. 이 챌린지는 순위를 매기기 위해 상위 5개 오류율을 사용합니다.

우리는 교육에 사용되는 외부 데이터 없이 챌린지에 참여했습니다. 이 백서에서 언급한 교육 기술 외에도 더 높은 성능을 얻기 위해 테스트 중에 일련의 기술을 채택했습니다. 자세한 내용은 아래에서 설명합니다.

1. 우리는 동일한 GoogLeNet 모델의 7개 버전(더 넓은 버전 포함)을 독립적으로 교육하고 앙상블 예측을 수행했습니다. 이러한 모델은 동일한 초기화(주로 감독으로 인해 동일한 초기 가중치를 사용하더라도) 및 학습률 정책으로 학습되었으며 샘플링 방법과 입력 이미지를 보는 무작위 순서만 다릅니다.

2. 테스트 중에 Krizhevsky et al보다 더 공격적인 자르기 접근 방식을 채택했습니다. [9]. 구체적으로, 우리는 이미지를 더 짧은 차원(높이 또는 너비)이 각각 256, 288, 320 및 352인 4가지 스케일로 크기를 조정하고 이러한 크기 조정된 이미지의 왼쪽, 중앙 및 오른쪽 정사각형을 취합니다(세로 이미지의 경우 상단, 중앙 및 하단 사각형).

   각 정사각형에 대해 4개의 모서리와 중앙 224×224 크롭 및

팀	연도 장소 오류(상위 5개) 외부 데이터 사용			
슈퍼비전 2012 1위			16.4%	아니요
슈퍼비전 2012 1위			15.3%	이미지넷 22k
클라리피스	2013년 1위		11.7%	아니요
클라리피스	2013년 1위		11.2%	이미지넷 22k
MSRA	2014년 3위		7.35%	아니요
VGG	2014년 2차 7.32%			아니요
구글넷 2014 1위			6.67%	아니요

표 2: 분류 성능

모델 수 기준 대비 작물 비용 상위 5개 오류 수				
1	1	1	10.07%	베이스
1	10	10	9.15%	-0.92%
1	144	144	7.89%	-2.18%
7	1	7	8.09%	-1.98%
7	10	70	7.62%	-2.45%
7	144	1008	6.67%	-3.45%

표 3: GoogLeNet 분류 성능 분석

정사각형은 224×224로 크기가 조정되고 미러링된 버전입니다. 그 결과 이미지당 4×3×6×2 = 144개의 크롭이 생성됩니다. Andrew Howard [8] 가 전년도 항목에서 비슷한 접근 방식을 사용했는데 제안된 체계보다 약간 더 나쁜 성능을 보이는 것으로 경험적으로 확인되었습니다.
합리적인 수의 작물이 존재하면 더 많은 작물의 이점이 미미해지기 때문에 실제 응용 프로그램에서는 이러한 공격적인 자르기가 필요하지 않을 수 있습니다(나중에 표시됨).

3. 소프트맥스 확률은 최종 예측을 얻기 위해 여러 작물과 모든 개별 분류자에 대해 평균화됩니다. 실험에서 우리는 작물에 대한 최대 풀링 및 분류기에 대한 평균화와 같은 검증 데이터에 대한 대체 접근 방식을 분석했지만 단순 평균화보다 성능이 떨어집니다.

이 백서의 나머지 부분에서는 최종 제출물의 전반적인 성능에 기여하는 여러 요소를 분석합니다.

챌린지의 최종 제출물은 유효성 검사 및 테스트 데이터 모두에서 6.67%의 상위 5개 오류를 획득하여 다른 참가자 중 1위를 차지했습니다. 이는 2012년 SuperVision 접근 방식에 비해 상대적으로 56.5% 감소한 것이며, 분류기 교육을 위해 외부 데이터를 사용한 전년도 최고의 접근 방식(Clarifai)에 비해 상대적으로 약 40% 감소한 것입니다. 다음 표는 최고 성능 접근 방식의 통계를 보여줍니다.

또한 다음 표에서 이미지를 예측할 때 사용되는 모델 수와 자르기 수를 변경하여 여러 테스트 선택의 성능을 분석하고 보고합니다. 하나의 모델을 사용할 때 검증 데이터에서 top-1 오류율이 가장 낮은 모델을 선택했습니다. 테스트 데이터 통계에 과적합되지 않도록 모든 수치가 유효성 검사 데이터 세트에 보고됩니다.

8 ILSVRC 2014 감지 챌린지 설정 및 결과

ILSVRC 감지 작업은 200개의 가능한 클래스 중에서 이미지의 개체 주위에 경계 상자를 생성하는 것입니다. 감지된 개체는 groundtruth의 클래스와 일치하고 해당 경계 상자가 50% 이상 겹치는 경우 올바른 것으로 간주됩니다(Jaccard 인덱스 사용). 관련 없는 탐지는 거짓 양성으로 간주되어 불이익을 받습니다. 분류 작업과 달리 각 이미지에는 다음이 포함될 수 있습니다.

팀	연도 장소 mAP 외부 데이터 앙상블 접근 방식 22.6% 없음				
UVA-유비전	2013년 1위			?	피셔 벡터
Deep Insight 40.5% ImageNet 1k, CUHK DeepID-Net 2014 2nd 40.7%				삼	CNN
ImageNet 1k 43.9% ImageNet 1k GoogLeNet				?	CNN
	2014년 1위			6	CNN

표 4: 감지 성능

팀	mAP 상황별 모델 경계 상자 회귀		
Trimps-Soushen	31.6%	아니요	?
버클리 비전	34.5%	아니요	예
UVA-유비전	35.4%	?	?
CUHK DeepID-Net2 37.7% GoogLeNet		아니요	?
Deep Insight	38.02%	아니요	아니요
	40.2%	예	예

표 5: 검출을 위한 단일 모델 성능

개체가 많거나 없거나 크기가 크거나 작을 수 있습니다. 결과는 평균 평균 정밀도(mAP)를 사용하여 보고됩니다.

탐지를 위해 GoogLeNet이 취하는 접근 방식은 [6]의 R-CNN과 유사 하지만 영역 분류기로 인셉션 모델로 보강됩니다. 또한 영역 제안 단계는 Selective Search [20] 접근 방식과 multi-box [5] 예측을 결합하여 더 높은 개체 경계 상자 리콜을 위해 개선되었습니다. 위양성 수를 줄이기 위해 슈퍼픽셀 크기를 2배로 늘렸습니다. 이렇게 하면 선택적 검색 알고리즘에서 오는 제안이 절반으로 줄어듭니다. 우리는 멀티 박스 [5] 에서 오는 200개의 지역 제안을 다시 추가하여 [6] 에서 사용된 제안의 총 약 60%를 얻었고 적용 범위를 92%에서 93%로 늘렸습니다. 적용 범위가 증가한 제안 수를 줄이는 전반적인 효과는 단일 모델 사례에 대한 평균 평균 정밀도의 1% 향상입니다. 마지막으로 각 영역을 분류할 때 6개의 ConvNet 앙상블을 사용하여 결과를 40%에서 43.9% 정확도로 향상시킵니다. R-CNN과 달리 시간이 부족하여 경계 상자 회귀를 사용하지 않았습니다.

먼저 상위 검색 결과를 보고하고 검색 작업의 초판 이후 진행 상황을 보여줍니다. 2013년 결과와 비교하여 정확도가 거의 두 배가 되었습니다. 최고의 성과를 내는 팀은 모두 Convolutional Networks를 사용합니다. 표 4의 공식 점수와 각 팀의 공통 전략(외부 데이터, 앙상블 모델 또는 상황별 모델 사용)을 보고합니다. 외부 데이터는 일반적으로 탐지 데이터에서 나중에 정제되는 모델을 사전 훈련하기 위한 ILSVRC12 분류 데이터입니다.

일부 팀은 현지화 데이터 사용에 대해서도 언급합니다. 지역화 작업 경계 상자의 상당 부분이 탐지 데이터 세트에 포함되어 있지 않기 때문에 분류가 사전 훈련에 사용되는 것과 같은 방식으로 이 데이터로 일반 경계 상자 회귀자를 사전 훈련할 수 있습니다. GoogLeNet 항목은 사전 훈련을 위해 현지화 데이터를 사용하지 않았습니다.

표 5에서는 단일 모델만을 사용하여 결과를 비교합니다. 최고의 성능을 발휘하는 모델은 Deep Insight에 의한 것으로 놀랍게도 3개 모델의 앙상블에서 0.3점만 향상되는 반면 GoogLeNet은 앙상블에서 훨씬 더 강력한 결과를 얻습니다.

9 결론

우리의 결과는 쉽게 사용할 수 있는 조밀한 빌딩 블록으로 예상되는 최적의 희소 구조를 근사화하는 것이 컴퓨터 비전을 위한 신경망을 개선하기 위한 실행 가능한 방법이라는 확실한 증거를 제공하는 것 같습니다. 이 방법의 주요 장점은 더 얕고 덜 넓은 네트워크에 비해 계산 요구 사항이 약간 증가하면서 상당한 품질 향상이 있다는 것입니다. 또한 우리의 탐지 작업은 컨텍스트를 활용하거나 바운딩 박스를 수행하지 않았음에도 불구하고 경쟁력이 있었습니다.

회귀와 이 사실은 Inception 아키텍처의 강점에 대한 추가 증거를 제공합니다. 유사한 깊이와 너비의 훨씬 더 비싼 네트워크로 유사한 품질의 결과를 얻을 수 있다고 예상되지만, 우리의 접근 방식은 일반적으로 희박한 아키텍처로 이동하는 것이 실현 가능하고 유용한 아이디어라는 확실한 증거를 제공합니다. 이것은 [2] 를 기반으로 자동화된 방식으로 더 희소하고 더 세련된 구조를 생성하기 위한 유망한 미래 작업을 제안합니다 .

10 감사의 말

[2] 에 대한 유익한 토론에 대해 Sanjeev Arora와 Aditya Bhaskara에게 감사드립니다 . 또한 우리는 특히 Rajat Monga, Jon Shlens, Alex Krizhevsky, Jeff Dean, Ilya Sutskever 및 Andrea Frome를 지원해 준 DistBelief [4] 팀에게 빚을 졌습니다. 또한 측광 왜곡에 도움을 주신 Tom Duerig와 Ning Ye에게도 감사드립니다. 또한 Chuck Rosenberg와 Hartwig Adam의 지원 없이는 작업이 불가능했을 것입니다.

참조

[1] 당신의 밈을 아십시오: 우리는 더 깊이 들어가야 합니다. http://knowyourmeme.com/memes/
더 깊이 들어가야 합니다. 접속일: 2014-09-15.

[2] Sanjeev Arora, Aditya Bhaskara, Rong Ge 및 Tengyu Ma. 학습에 대한 증명 가능한 범위
일부 깊은 표현. CoRR, abs/1310.6343, 2013.

[3] Umit V. C ¸ ataly urek, Cevdet Aykanat 및 Bora Uc ¸ ar. 2차원 희소 행렬 분할: 모델, 방법 및 레시피. SIAM J. Sci. Comput., 32(2):656–683, 2010년 2월.

[4] Jeffrey Dean, Greg Corrado, Rajat Monga, Kai Chen, Matthew Devin, Mark Mao, Andrew Senior, Paul Tucker, Ke Yang, Quoc V. Le 및 Andrew Y.
응. 대규모 분산 심층 네트워크. P. Bartlett, Fcn Pereira, Cjc Burges, L. Bottou 및 Kq Weinberger, 편집자, Advances in Neural Information Processing Systems 25, 페이지 1232–1240. 2012.

[5] Dumitru Erhan, Christian Szegedy, Alexander Toshev, Dragomir Anguelov. 심층 신경망을 사용한 확장 가능한 개체 감지. 컴퓨터 비전 및 패턴 인식, 2014년.
CVPR 2014. IEEE 컨퍼런스 온, 2014.

[6] Ross B. Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik. 정확한 객체 감지 및 시맨틱 분할을 위한 풍부한 기능 계층. 컴퓨터 비전 및 패턴 인식, 2014. CVPR 2014. IEEE 회의, 2014.

[7] Geoffrey E. Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, Ruslan Salakhut dinov. 특징 검출기의 공동 적응을 방지하여 신경망을 개선합니다. CoRR, abs/1207.0580, 2012.

[8] 앤드류 G. 하워드. 심층 컨벌루션 신경망 기반 이미지 분류에 대한 몇 가지 개선 사항. CoRR, abs/1312.5402, 2013.

[9] Alex Krizhevsky, Ilya Sutskever, Geoff Hinton. 심층 합성곱 신경망을 사용한 이미지넷 분류. 신경 정보 처리 시스템의 발전 25, 페이지 1106–1114, 2012.

[10] Y. LeCun, B. Boser, JS Denker, D. Henderson, RE Howard, W. Hubbard 및 LD Jackel.
손으로 쓴 우편번호 인식에 적용된 역전파. Neural Comput., 1(4):541–551, 1989년 12월.

[11] Yann LeCun, Leon Bottou, Yoshua Bengio 및 Patrick Haffner. 문서 인식에 적용된 기울기 기반 학습. IEEE 절차, 86(11):2278–2324, 1998.

[12] Min Lin, Qiang Chen 및 Shuicheng Yan. 네트워크의 네트워크. CoRR, abs/1312.4400, 2013.

[13] BT 폴리악과 AB 주디츠키. 평균화에 의한 확률적 근사의 가속화. SIAM J. Control Optim., 30(4):838–855, 1992년 7월.

[14] Pierre Sermanet, David Eigen, Xiang Zhang, Michael Mathieu, Rob Fergus, Yann Le-Cun. Overfeat: 컨볼루션 네트워크를 사용한 통합 인식, 지역화 및 탐지. CoRR, abs/1312.6229, 2013.

[15] Thomas Serre, Lior Wolf, Stanley M. Bileschi, Maximilian Riesenhuber, Tomaso Poggio.
피질과 유사한 메커니즘을 통한 강력한 객체 인식. IEEE 트랜스. 패턴 항문. 마하.
Intel., 29(3):411–426, 2007.

[16] 평광 송과 잭 동가라. CPU 코어가 1000개인 공유 메모리 매니코어 시스템에서 매트릭스 계산을 확장합니다. 슈퍼컴퓨팅에 관한 28차 ACM 국제 회의 진행, ICS '14, 333–342페이지, 미국 뉴욕주 뉴욕, 2014년. ACM.

[17] Ilya Sutskever, James Martens, George E. Dahl, Geoffrey E. Hinton. 딥러닝에서 초기화와 모멘텀의 중요성. 기계 학습에 관한 30차 국제 회의 절차, ICML 2013, 미국 조지아주 애틀랜타, 2013년 6월 16-21일, JMLR 절차 28권, 1139–1147페이지. JMLR.org, 2013.

[18] 크리스티안 세게디, 알렉산더 토셰프, 두미트루 에르한. 물체 감지를 위한 심층 신경망. Christopher JC Burges, Leon Bottou, Zoubin Ghahramani 및 Kilian Q.
Weinberger, 편집자, Advances in Neural Information Processing Systems 26: 2013년 신경 정보 처리 시스템에 관한 27차 연례 회의. 2013년 12월 5-8일, 미국 네바다주 레이크 타호에서 개최된 회의 절차, 2553–2561, 2013페이지.

[19] 알렉산더 토셰프와 크리스티안 세게디. Deeppose: 심층 신경망을 통한 인간 포즈 추정. CoRR, abs/1312.4659, 2013.

[20] Koen EA van de Sande, Jasper RR Uijlings, Theo Gevers 및 Arnold WM Smeulders.
객체 인식을 위한 선택적 검색으로 분할. 컴퓨터 비전에 관한 2011년 국제 회의 절차, ICCV '11, 페이지 1879–1886, 미국 워싱턴 DC, 2011. IEEE Computer Society.

[21] 매튜 D. 자일러와 롭 퍼거스. 컨볼루션 네트워크 시각화 및 이해 In David J. Fleet, Tomas Pajdla, Bernt Schiele 및 Tinne Tuytelaars, 편집자, Computer Vision - ECCV 2014 - 13차 유럽 회의, 스위스 취리히, 2014년 9월 6-12일, 강의 노트의 Pro ceedings, Part I, 8689권 컴퓨터 과학, 페이지 818–833. 스프링거, 2014.