

촘촘하게 연결된 컨볼루션 네트워크

가오 황* 코넬 대

학교

gh349@cornell.edu

Zhuang Liu* 칭화대

학교

liuzhuang13@mails.tsinghua.edu.cn

로렌스 반 데어 마텐 페이스북

AI 리서치

lvdmaaten@fb.com

킬리안 Q. 와인버거 코넬 대

학교

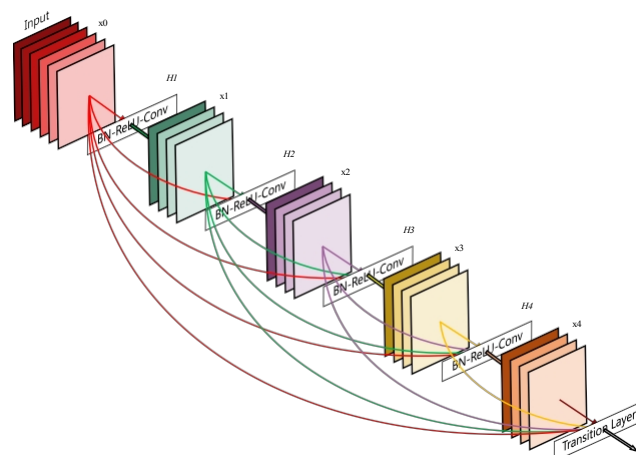
kqw4@cornell.edu

초록

최근 연구에 따르면 컨볼루션 네트워크는 입력에 가까운 레이어와 출력에 가까운 레이어 사이에 짧은 연결을 포함할 경우 훨씬 더 깊고 정확하며 효율적으로 훈련할 수 있습니다. 이 백서에서는 이러한 관찰을 수용하여 각 레이어를 피드 포워드 방식으로 다른 모든 레이어에 연결하는 고밀도 컨볼루션 네트워크(DenseNet)를 소개합니다. L 개의 레이어로 구성된 기존 컨볼루션 네트워크는 각 레이어와 그 다음 레이어 사이에 L 개의 연결이 있는 반면, 밀도 네트워크는 각 레이어와 그 다음 레이어 사이에

레이어에 $(L+1)$ 직접 연결이 가능합니다. For

각 레이어에서 모든 선행 레이어의 특징 맵이 입력으로 사용되고, 자체 특징 맵이 모든 후속 레이어의 입력으로 사용됩니다. 덴스 넷은 소실 그래데이션 문제를 완화하고, 특징 전파를 강화하며, 기능 재사용을 장려하고, 파라미터 수를 크게 줄이는 등 여러 가지 장점이 있습니다. 경쟁이 치열한 네 가지 물체 인식 벤치마크 과제(CIFAR-10, CIFAR-100, SVHN, ImageNet)에서 제안한 아키텍처를 평가합니다. DenseNet은 대부분의 작업에서 최신 기술에 비해 상당한 개선 효과를 얻으면서도 더 적은 연산으로 높은 성능을 달성합니다. 코드와 사전 훈련된 모델은 <https://github.com/liuzhuang13/DenseNet>



에서 확인할 수 있습니다.

1. 소개

컨볼루션 신경망(CNN)은 시각적 객체 인식을 위한 주요 머신 러닝 접근법으로 자리 잡았습니다. CNN은 20여 년 전에 처음 도입되었지만[18], 컴퓨터 하드웨어와 네트워크 구조가 개선되면서 최근에야 진정한 의미의 심층 CNN을 학습할 수 있게 되었습니다. 최초의 LeNet5[19]는 5개의 레이어로 구성되었고, VGG는 19개[29], 그리고 작년에는 Highway

*저자들이 동등하게 기여

그림 1: 성장률이 $k = 4$ 인 5계층 고밀도 블록. 각 레이어는 앞의 모든 피쳐 맵을 입력으로 받습니다.

네트워크[34]와 잔여 네트워크(ResNets)[11]가 100계층의 장벽을 넘어섰습니다.

CNN이 점점 더 깊어짐에 따라 입력 또는 그래디언트에 대한 정보가 여러 계층을 통과함에 따라 네트워크의 끝(또는 시작)에 도달할 때쯤에는 정보가 사라지거나 "씻겨 나갈" 수 있다는 새로운 연구 문제가 대두되었습니다. 최근 많은 논문에서 이 문제 또는 이와 관련된 문제를 다루고 있습니다. ResNets[11] 및 고속도로 네트워크[34]는 신원 연결을 통해 한 계층에서 다음 계층으로 신호를 우회합니다. 확률적 깊이[13]는 훈련 중에 레이어를 무작위로 떨어뜨려 더 나은 정보와 경사도 흐름을 허용함으로써 ResNet을 단축합니다. 프랙탈넷[17]은 네트워크에서 많은 짧은 경로를 유지하면서 큰 공칭 깊이를 얻기 위해 여러 개의 병렬 레이어 시퀀스를 서로 다른 수의 컨볼루션 블록과 반복적으로 결합합니다. 이러한 다양한 접근 방식은 네트워크 토폴로지와 훈련 절차가 다르지만, 모두 초기 레이어에서 이후 레이어로 가는 짧은 경로를 생성한다는 핵심적인 특징을 공유합니다.

이 백서에서는 이러한 인사이트를 간단한 연결 패턴으로 추출하는 아키텍처를 제안합니다. 네트워크 내 레이어 간의 정보 흐름을 최대화하기 위해 *모든 레이어*(피쳐 맵 크기가 일치하는)를 서로 직선으로 연결합니다. 피드 포워드 특성을 유지하기 위해 각 레이어는 앞의 모든 레이어에서 추가 입력을 얻고 자체 피쳐 맵을 모든 후속 레이어에 전달합니다. 그림 1은 이 레이아웃을 개략적으로 보여줍니다. 결정적으로, ResNet과 달리, 우리는 레이어에 전달되기 전에 합계를 통해 피쳐를 결합하지 않고, 대신 피쳐를 연결하여 ² 결합합니다. 따라서 ^m 레이어에는 앞선 모든 컨볼루션 블록의 특징 맵으로 구성된 입력이 있습니다. 자체 피쳐 맵은 모든 $L-1$ 후속 레이어에 전달됩니다. 이렇게 하면 기존 아키텍처에서처럼 L 이 아닌 L 계층 네트워크에 $L(L+1)$ 연결이 도입됩니다. 이러한 조밀한 연결성 패턴 때문에 이 접근 방식을 *조밀한 컨볼루션 네트워크(DenseNet)*라고 부릅니다.

이러한 고밀도 연결 패턴의 직관적이지 않은 효과는 중복된 특징 맵을 다시 학습할 필요가 없기 때문에 기존 컨볼루션 네트워크보다 더 적은 수의 매개변수가 필요하다는 점입니다. 기존의 피드 포워드 아키텍처는 레이어에서 레이어로 전달되는 상태를 가진 알고리즘으로 볼 수 있습니다. 각 레이어는 이전 레이어에서 상태를 읽고 후속 레이어에 씁니다. 이는 상태를 변경할 뿐만 아니라 보존해야 할 정보도 전달합니다. ResNets [11]은 추가적인 아이덴티티 변환을 통해 이러한 정보 보존을 명시적으로 구현합니다. 최근 ResNet의 변형[13]은 많은 레이어가 거의 기여하지 않으며 실제로 훈련 중에 무작위로 삭제될 수 있음을 보여줍니다. 이로 인해 ResNet의 상태는 (풀린) 순환 신경망[21]과 유사해졌지만, 각 레이어에 고유한 가중치가 있기 때문에 ResNet의 매개변수 수는 훨씬 더 많아졌습니다. 저희가 제안한 DenseNet 아키텍처는 네트워크에 추가되는 정보와 보존되는 정보를 명시적으로 구분합니다. DenseNet 레이어는 매우 좁아서(예: 레이어당 12개의 필터) 네트워크의 '집단 지식'에 소수의 특징 맵만 추가하고 나머지 특징 맵은 그대로 유지하며, 최종 분류기는 네트워크의 모든 특징 맵

을 기반으로 결정을 내립니다.

더 나은 파라미터 효율성 외에도, 밀집망의 큰 장점 중 하나는 네트워크 전체에 걸친 정보와 경사도의 흐름이 개선되어 훈련이 쉽다는 것입니다. 각 레이어는 손실 함수와 원래 입력 신호의 기울기에 직접 액세스할 수 있기 때문에 심층적인 감독이 가능합니다[20]. 이는 더 심층적인 네트워크 아키텍처를 훈련하는 데 도움이 됩니다. 또한, 밀도가 높은 연결은 규칙화 효과가 있어 훈련 세트 크기가 작은 작업에서 과도한 피팅을 줄여주는 것으로 나타났습니다.

경쟁이 치열한 네 가지 벤치마크 데이터 세트(CIFAR-10, CIFAR-100, SVHN, ImageNet)에서 DenseNet을 평가합니다. 우리 모델은 훨씬 적은 수의 파라미터를 필요로 하는 경향이 있습니다.

를 기존 알고리즘보다 더 정확하게 계산합니다. 또한, 대부분의 벤치마크 작업에서 현재의 최첨단 결과보다 훨씬 뛰어난 성능을 발휘합니다.

2. 관련 작업

네트워크 아키텍처에 대한 탐구는 신경망이 처음 발견되었을 때부터 신경망 연구의 일부였습니다. 최근 신경망에 대한 인기가 다시 높아지면서 이 연구 분야도 다시 활기를 띠고 있습니다. 최신 네트워크의 레이어 수가 증가함에 따라 아키텍처 간의 차이가 증폭되고 다양한 연결성 패턴을 탐색하고 오래된 연구 아이디어를 재검토하도록 동기를 부여합니다.

우리가 제안한 고밀도 네트워크 작업 레이아웃과 유사한 캐스케이드 구조는 이미 1980년대에 신경망 문헌에서 연구된 바 있습니다[3]. 이들의 선구적인 연구는 레이어 단위로 훈련된 완전히 연결된 다층 퍼셉트론에 초점을 맞췄습니다. 최근에는 일괄 경사 하강으로 훈련할 수 있는 완전 연결된 캐스케이드 네트워크가 제안되었습니다[40]. 이 접근 방식은 소규모 데이터 세트에는 효과적이지만, 수백 개의 파라미터를 가진 네트워크까지만 확장할 수 있습니다. [9, 23, 31, 41]에서는 스킵 연결을 통해 CNN의 다단계 특징을 활용하는 것이 다양한 비전 작업에 효과적인 것으로 밝혀졌습니다. 우리의 연구와 병행하여 [1]에서는 우리와 유사한 계층 간 연결이 있는 네트워크에 대한 순수 이론적 프레임워크를 도출했습니다.

고속도로 네트워크[34]는 100개 이상의 레이어가 있는 엔드투엔드 네트워크를 효과적으로 훈련할 수 있는 수단을 제공한 최초의 아키텍처 중 하나였습니다. 게이팅 유닛과 함께 바이패스 경로를 사용하면 수백 개의 레이어가 있는 고속도로 네트워크를 어려움 없이 최적화할 수 있습니다. 바이패스 경로는 이러한 매우 심층적인 네트워크의 트레이닝을 용이하게 하는 핵심 요소로 추정됩니다. 이 점은 순수한 아이덴티티 매핑을 우회 경로로 사용하는 ResNets [11]에 의해 더욱 뒷받침됩니다. ResNets는 ImageNet 및 COCO 객체 감지[11]와 같은 까다로운 이미지 인식, 로컬

라이제이션 및 감지 작업에서 놀랍고 기록적인 성능을 달성했습니다. 다시 말해, *확률적 심도*는 1202개의 레이어로 구성된 ResNet을 성공적으로 훈련하는 방법으로 제안되었습니다[13]. 확률적 심도는 훈련 중에 레이어를 무작위로 떨어뜨려 심층 잔류 네트워크의 훈련을 증명합니다. 이는 모든 레이어가 필요하지 않을 수 있음을 보여주며 심층(잔여) 네트워크에 상당한 양의 중복성이 있음을 강조합니다. 유니티의 페이퍼는 부분적으로 이러한 관찰에서 영감을 받았습니다. 또한 *사전 활성화 기능이 있는* 레스넷은 1000개 이상의 레이어를 가진 최첨단 네트워크의 트레이닝을 용이하게 합니다[12].

네트워크를 더 깊게 만들기 위한 직교적 접근 방식(*off-diagonal* 연결 건너뛰기 사용)은 네트워크 폭을 늘리는 것입니다. GoogLeNet [36, 37]은 다양한 크기의 필터로 생성된 특징 맵을 연결하는 "Inception 모듈"을 사용합니다. [38]에서는 넓은 일반화된 잔여 블록을 가진 ResNet의 변형이 제안되었습니다. 사실, 각 레이어에 있는 필터의 수를 늘리는 것만으로도

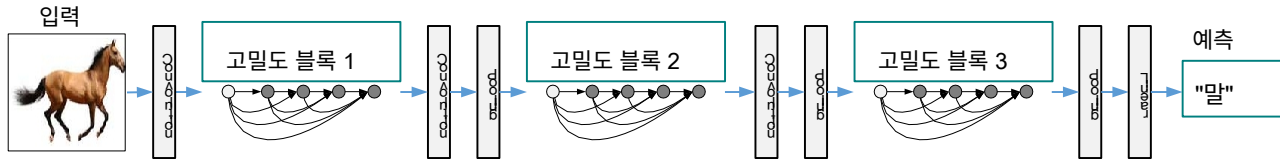


그림 2: 3개의 고밀도 블록이 있는 딥 덴스넷. 인접한 두 블록 사이의 레이어를 트랜지션 레이어라고 하며 컨볼루션과 풀링을 통해 피쳐 맵 크기를 변경합니다.

ResNets는 깊이가 충분하다면 성능을 향상시킬 수 있습니다 [42]. 프랙탈넷도 넓은 네트워크 구조를 사용하여 여러 데이터 세트에서 경쟁력 있는 결과를 얻을 수 있습니다 [17].

덴스넷은 매우 깊거나 넓은 아키텍처에서 표현력을 끌어오는 대신 특징재사용을 통해 네트워크의 잠재력을 활용하여 훈련하기 쉽고 매개변수 효율성이 높은 조밀화된 모델을 생성합니다. 서로 다른 레이어에서 학습한 특징 맵을 연결하면 후속 레이어 입력의 다양성이 증가하고 효율성이 향상됩니다. 이것이 바로 밀집망과 레즈넷의 주요 차이점입니다. 서로 다른 레이어의 특징을 연결하는 Inception 네트워크[36, 37]와 비교했을 때, DenseNet은 더 간단하고 효율적입니다.

경쟁력 있는 결과를 가져온 다른 주목할 만한 네트워크 아키텍처 혁신도 있습니다. 네트워크 내 네트워크 (NIN)[22] 구조는 컨볼루션 계층의 필터에 마이크로 다층 퍼셉트론을 포함시켜 더 복잡한 특징을 추출합니다. 심층 감독 네트워크(DSN)[20]에서는 내부 레이어가 보조 분류기에 의해 직접 감독되며, 이는 이전 레이어가 받은 그라데이션을 강화할 수 있습니다. 래더 네트워크[27, 25]는 자동 인코더에 측면 연결을 도입하여 반지도 학습 작업에서 인상적인 정확도를 제공합니다. 39]에서는 서로 다른 기본 네트워크의 중간 계층을 결합하여 정보 흐름을 증명하기 위해 심층 융합 네트워크(DFN)가 제안되었습니다. 재구성 손실을 최소화하는 경로로 네트워크를 보강하면 이미지 분류 모델을 개선하는 것으로 나타났습니다 [43].

3. DenseNets

컨볼루션 네트워크를 통과하는 단일 이미지 \mathbf{x}_0 를 생각해 봅시다. 이 네트워크는 L 개의 레이어로 구성되며, 각 레

이어는 비선형 변환 $H_A(-)$ 를 구현하고, 여기서 l 은 레이어를 인덱싱합니다. $H_A(-)$ 는 일괄 정규화(BN) [14], 정류된 선형 단위(ReLU) [6], 풀링[19] 또는 컨볼루션과 같은 연산들의 복합 함수일 수 있습니다.

레스넷의 장점은 그래디언트가 아이덴티티 함수를 통해 이후 레이어에서 이전 레이어로 직선으로 흐를 수 있다는 것입니다. 하지만 아이덴티티 함수와 H_A 의 출력은 합산으로 결합되기 때문에 네트워크의 정보 흐름을 방해할 수 있습니다.

축축한 연결성. 계층 간의 정보 흐름을 더욱 개선하기 위해 다른 연결 패턴을 제안합니다. 즉, 모든 계층에서 모든 후속 계층으로 직접 연결하는 방식을 도입합니다. 그림 1은 결과물인 DenseNet의 레이아웃을 개략적으로 보여줍니다. 결과적으로 l^{th} 레이어는 모든 선행 레이어의 피쳐 맵을 수신합니다,

$\mathbf{x}_0, \dots, \mathbf{x}_{l-1}$ 를 입력합니다:

$$\mathbf{x}_l = H_l([\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_{l-1}]), \quad (2)$$

여기서 $[\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_{l-1}]$ 는 레이어 $0, \dots, l-1$ 에서 생성된 피쳐 맵의 연결을 나타냅니다., $l-1$. 그 이유는 밀도 높은 연결성을 위해 이 네트워크 아키텍처를 *밀도 컨*

(Conv). l^{th} 레이어의 출력을 \mathbf{x}_l 로 표시합니다.

ResNets. 기존의 컨볼루션 피드 포워드 네트워크는 l^{th} 레이어의 출력을 입력으로 하여

$(l+1)^{th}$ 레이어 [16]를 생성하여 다음과 같은 레이어 전환이 발생합니다: $\mathbf{x}_l = H_l(\mathbf{x}_{l-1})$. ResNets [11]은 비선형 변환을 우회하는 스킵 연결을 추가합니다. 신원 함수를 사용합니다:

$$\mathbf{x}_l = H_l(\mathbf{x}_{l-1}) + \mathbf{x}_{l-1}. \quad (1)$$

컨볼루션 네트워크(DenseNet)라고 부릅니다. 구현을 쉽게 하기 위해 방정식 (2)에서 $H_A(-)$ 의 여러 입력을 단일 텐서로 연결합니다.

복합 함수. [12]에서 영감을 받아 $H_A(-)$ 를 일괄 정규화(BN) [14], 정류된 선형 귀 단위(ReLU) [6], 3×3 컨볼루션 (Conv) 등 세 가지 연산의 복합 함수로 정의합니다.

레이어 풀링. 식 (2)에 사용된 연결 연산은 특징 맵의 크기가 변경될 때 실행할 수 없습니다. 그러나 컨볼루션 네트워크의 필수적인 부분은 특징 맵의 크기를 변경하는 다운 샘플링 레이어입니다. 저희 아키텍처에서는 다운 샘플링을 용이하게 하기 위해 네트워크를 여러 개의 조밀하게 연결된 *고밀도 블록*으로 나눕니다(그림 2 참조). 블록 사이의 레이어를 *트랜지션 레이어*라고 하며, 이 레이어는 컨볼루션과 풀링을 수행합니다. 실험에 사용된 트랜지션 레이어는 일괄 정규화 레이어와 1×1 컨볼루션 레이어, 그리고 2×2 평균 풀링 레이어로 구성됩니다.

성장률. 각 함수 H_A K 기능을 생성합니다

맵의 경우, l^{th} 레이어는 $k_0 + k \times (l-1)$ 입력 피쳐 맵을 가지며, 여기서 k_0 는 인풋 레이어의 채널 수입니다. DenseNet과 기존 네트워크 아키텍처의 중요한 차이점은 DenseNet은 매우 좁은 레이어($k=12$)를 가질 수 있다는 것입니다. 우리는 하이퍼 파라미터 k 를 네트워크의 *성장률*이라고 부릅니다. 섹션 4에서는 상대적으로 작은 성장률로도 다음과 같은 효과를 얻을 수 있음을 보여줍니다.

레이어	출력 크기	DenseNet-121	DenseNet-169	DenseNet-201	DenseNet-264
컨볼루션	112×112	7×7 컨브, 보폭 2			
풀링	56×56	최대 3×3 풀, 보폭 2			
고밀도 블록 (1)	56×56	1×1 conv 3×3 conv $\times 6$	1×1 conv 3×3 conv $\times 6$	1×1 conv 3×3 conv $\times 6$	1×1 conv 3×3 conv $\times 6$
트랜지션 레이어 (1)	56×56 28×28	1×1 conv 2×2 평균 풀, 보폭 2			
고밀도 블록 (2)	28×28	1×1 conv 3×3 conv $\times 12$	1×1 conv 3×3 conv $\times 12$	1×1 conv 3×3 conv $\times 12$	1×1 conv 3×3 conv $\times 12$
트랜지션 레이어 (2)	28×28 14×14	1×1 conv 2×2 평균 풀, 보폭 2			
고밀도 블록 (3)	14×14	1×1 conv 3×3 conv $\times 24$	1×1 conv 3×3 conv $\times 32$	1×1 conv 3×3 conv $\times 48$	1×1 conv 3×3 conv $\times 64$
트랜지션 레이어 (3)	14×14 7×7	1×1 conv 2×2 평균 풀, 보폭 2			
고밀도 블록 (4)	7×7	1×1 conv 3×3 conv $\times 16$	1×1 conv 3×3 conv $\times 32$	1×1 conv 3×3 conv $\times 32$	1×1 conv 3×3 conv $\times 48$
분류 레이어	1×1	7×7 글로벌 평균 풀 1000D 완전 연결, 소프트맥스			

표 1: 이미지넷을 위한 DenseNet 아키텍처. 모든 네트워크의 성장률은 $k = 32$ 입니다. 표에 표시된 각 "conv" 레이어는 BN-ReLU-Conv 시퀀스에 해당합니다.

테스트한 데이터 세트에 대한 최신 결과를 얻을 수 있습니다. 이에 대한 한 가지 설명은 각 계층이 해당 블록의 모든 선행 피쳐 맵에 액세스할 수 있으므로 네트워크의 '집단 지식'에 액세스할 수 있다는 것입니다. 피쳐 맵을 네트워크의 글로벌 상태로 볼 수 있습니다. 각 레이어는 이 상태에 자체 피쳐맵을 추가합니다. 성장률은 각 계층이 글로벌 상태에 추가하는 새로운 정보의 양을 조절합니다. 글로벌 상태는 한 번 작성되면 네트워크 내 모든 곳에서 액세스할 수 있으며, 기존 네트워크 아키텍처와 달리 레이어 간에 복제할 필요가 없습니다.

병목 현상 레이어. 각 계층은 k 개의 출력 특징 맵만 생성하지만, 일반적으로 더 많은 입력이 있습니다. [37, 11]에서는 각 3×3 컨볼루션 전에 1×1 컨볼루션을 병목 계층으로 도입하여 입력 특징 맵의 수를 줄이고 계산 효율을 향상시킬 수 있다고 언급했습니다. 이러한 설계는 DenseNet에 특히 효과적이며, 이러한 병목 계층이 있는 네트워크, 즉 H_A 의 BN-ReLU-Conv(1×1)-BN-ReLU-Conv(3×3) 버전을 DenseNet-B라고 부릅니다. 실험에서는 각 1×1 컨볼루션이 $4k$ 피쳐 맵을 생성하도록 했습니다.

압축. 모델 압축을 더욱 개선하기 위해 트랜지션 레이어에서 피쳐 맵의 수를 줄일 수 있습니다. 밀도가 높은 블록에 피쳐 맵이 m 개 포함되어 있는 경우, 다음 트랜지션 레이어에서 $[0, m]$ 출력 피쳐 맵을 생성하도록 하며, 여기서 $0 < \theta \leq 1$ 을 압축 팩터라고 합니다. $\theta = 1$ 일 때, 전이 레이어에 걸친 특징 맵의 수는 변하지 않습니다. θ 가 1 미만인 DenseNet을 DenseNet-C라고 하며, 실험에서는 $\theta = 0.5$ 로 설정했습니다. θ 가 1 미만인 병목 및 전이 레이어를 모두 사용하는 경우 모델을 DenseNet-BC라고 합니다.

구현 세부 사항. ImageNet을 제외한 모든 데이터 세트에서 실험에 사용된 DenseNet은 각각 동일한 수의 레이어를 가진 세 개의 고밀도 블록을 가지고 있습니다. 첫 번째 고밀도 블록에 들어가기 전에 입력 이미지에 대해 16개(또는 DenseNet-BC의 경우 성장률의 두 배) 출력 채널의 컨볼루션이 수행됩니다. 커널 크기가 3×3 인 컨볼루션 레이어의 경우, 특징 맵 크기를 고정하기 위해 입력의 각 면에 1픽셀씩 제로 패딩을 적용합니다. 인접한 두 개의 고밀도 블록 사이의 전환 레이어로 1×1 컨볼루션과 2×2 평균 풀링을 사용합니다. 마지막 고밀도 블록의 끝에서 글로벌 평균 풀링이 수행된 다음 소프트맥스 분류기가 첨부됩니다. 세 개의 고밀도 블록의 특징 맵 크기는 각각 32×32 , 16×16 , 8×8 입니다. $\{L = 40, k = 12\}$, $\{L = 100, k = 12\}$, $\{L = 100, k = 24\}$ 로 구성된 기본 DenseNet 구조를 실험합니다. DenseNet-BC의 경우, $\{L = 100, k = 12\}$ 구성의 네트워크입니다, $\{L = 250, k = 24\}$ 및 $\{L = 190, k = 40\}$ 이 평가됩니다.

이미지넷에서의 실험에서는 224×224 입력 이미지에 4개의 고밀도 블록이 있는 DenseNet-BC 구조를 사용했습니다. 초기 컨볼루션 레이어는 보폭 2의 7×7 크기의 $2k$ 컨볼루션으로 구성되며, 다른 모든 레이어의 피쳐 맵 수 역시 k 설정에 따라 달라집니다. ImageNet에서 사용한 정확한 네트워크 구성은 표 1에 나와 있습니다.

4. 실험

여러 벤치마크 데이터 세트에 대한 DenseNet의 효과를 실증적으로 입증하고, 최첨단 아키텍처, 특히 ResNet 및 그 변종과 비교합니다.

방법	깊이	매개변수	C10	C10+	C100	C100+	SVHN
네트워크 내 네트워크 [22]	-	-	10.41	8.81	35.68	-	2.35
All-CNN [32]	-	-	9.08	7.25	-	33.71	-
심층 감시망 [20]	-	-	9.69	7.97	-	34.57	1.92
고속도로 네트워크 [34]	-	-	-	7.72	-	32.39	-
프랙탈넷 [17]	21	38.6M	10.18	5.22	35.34	23.30	2.01
드롭아웃/드롭 경로 포함	21	38.6M	7.33	4.60	28.20	23.73	1.87
ResNet [11]	110	1.7M	-	6.61	-	-	-
ResNet([13]에서 보고)	110	1.7M	13.63	6.41	44.74	27.22	2.01
확률적 깊이를 사용한 리셋 [13]	110	1.7M	11.66	5.23	37.80	24.58	1.75
	1202	10.2M	-	4.91	-	-	-
와이드 레스넷 [42]	16	11.0M	-	4.81	-	22.07	-
	28	36.5M	-	4.17	-	20.50	-
	16	2.7M	-	-	-	-	1.64
ResNet(사전 활성화) [12] [12]	164	1.7M	11.26*	5.46	35.58*	24.33	-
	1001	10.2M	10.56*	4.62	33.47*	22.71	-
DenseNet($k = 12$)	40	1.0M	7.00	5.24	27.55	24.42	1.79
DenseNet($k = 12$)	100	7.0M	5.77	4.10	23.79	20.20	1.67
DenseNet($k = 24$)	100	27.2M	5.83	3.74	23.42	19.25	1.59
DenseNet-BC($k = 12$)	100	0.8M	5.92	4.51	24.15	22.27	1.76
DenseNet-BC($k = 24$)	250	15.3M	5.19	3.62	19.64	17.60	1.74
DenseNet-BC($k = 40$)	190	25.6M	-	3.46	-	17.18	-

표 2: CIFAR 및 SVHN 데이터 세트의 오류율(%). k 는 네트워크의 성장률을 나타냅니다. 모든 경쟁 방법을 능가하는 결과는 **굵게** 표시되며, 전체적으로 가장 좋은 결과는 **파란색입니다**. "+"는 표준 데이터 증강(번역 및/또는 미러링)을 나타냅니다. *는 자체적으로 실행한 결과를 나타냅니다. 데이터 증강을 사용하지 않은 DenseNets의 모든 결과(C10, C100, SVHN)는 드롭아웃을 사용하여 얻은 결과입니다. 덴스넷이 달성하는 결과

ResNet보다 더 적은 수의 파라미터를 사용하면서 오류율을 낮춥니다. 데이터 증강이 없는 경우 DenseNet의 성능이 큰 폭으로 향상됩니다.

4.1. 데이터 세트

CIFAR. 두 개의 CIFAR 데이터 세트[15]는 32×32 픽셀의 컬러 자연 이미지로 구성되어 있습니다. CIFAR-10(C10)은 10개의 클래스에서 가져온 이미지로, CIFAR-100(C100)은 100개의 클래스에서 가져온 이미지로 구성됩니다. 훈련 및 테스트 세트에는 각각 50,000개와 10,000개의 이미지가 포함되며, 5,000개의 훈련 이미지를 검증 세트로 사용합니다. 이 두 데이터 세트에 널리 사용되는 표준 데이터 증강 방식(미러링/시프팅)을 채택합니다[11, 13, 17, 22, 28, 20, 32, 34]. 이 데이터 증강 방식은 데이터 세트 이름 끝에 "+" 표시로 표시합니다(예: C10+). 전처리를 위해 채널 평균과 표준 편차를 사용해 데이터를 정규화합니다. 최종 실행에서는 50,000개의 훈련 이미지를 모두 사용하고 훈련이 끝날 때 최종 테스트 오류를 보고합니다.

[24]에는 32×32 컬러 숫자 이미지가 포함되어 있습니다. 훈련 세트에는 73,257개의 이미지가 있고, 테스트 세트에는 26,032개의 이미지가 있으며, 추가 훈련을 위한 531,131개의 이미지가 있습니다. 일반적인 관행[7, 13, 20, 22, 30]에 따라 데이터 증강 없이 모든 훈련 데이터를 사용하고, 6,000개의 이미지로 구성된 검증 세트가 훈련 세트에서 분리됩니다. 훈련 중에 검증 오류가 가장 낮은 모델을 선택하고 테스트 오류를 보고합니다. [42]를 따르고 픽셀 값을 255로 나누어 $[0, 1]$ 범위가 되도록 합니다.

SVHN. 스트리트뷰 집 번호(SVHN) 데이터 세트

ImageNet. ILSVRC 2012 분류 데이터 세트 [2]는 1, 000 개의 클래스에서 120만 개의 훈련용 이미지와 5만 개의 평가용 이미지로 구성되어 있습니다. 훈련 이미지에는 [8, 11, 12]와 동일한 데이터 증강 방식을 채택하고 테스트 시 224×224 크기의 단일 크롭 또는 10 크롭을 적용합니다. [11, 12, 13]에 이어, 검증 세트에 대한 분류 오류를 보고합니다.

4.2. 교육

모든 네트워크는 확률적 그라데이션 디향(SGD)을 사용해 훈련됩니다. CIFAR과 SVHN에서는 각각 300개와 40개의 에포크에 대해 배치 크기 64를 사용해 훈련합니다. 초기 학습률은 0.1로 설정하고 총 훈련 에포크 수의 50%와 75%에서 10으로 나눕니다. 이미지넷에서는 배치 크기가 256인 90개의 에포크에 대해 모델을 훈련합니다. 학습 속도는 처음에 0.1로 설정되고 에포크 30과 60에서 10씩 낮아집니다. DenseNet을 순진하게 구현하면 메모리 비효율이 발생할 수 있습니다. GPU의 메모리 소비를 줄이면 메모리 효율적인 DenseNet 구현에 대한 기술 보고서를 참조하세요[26].

[8]에 따라 가중치 감쇠 10^{-4} 과 네스테로프 모멘텀[35]을 감쇠 없이 0.9로 사용합니다. W_e 는 [10]에서 도입한 가중치 초기화를 채택합니다. 데이터 증강이 없는 세 가지 데이터 세트, 즉 C10, C100의 경우

모델	top-1	상위 5위
DenseNet-121	25.02 / 23.61	7.71 / 6.66
DenseNet-169	23.80 / 22.08	6.85 / 5.92
DenseNet-201	22.58 / 21.46	6.34 / 5.54
DenseNet-264	22.15 / 20.80	6.12 / 5.29

표 3: 단일 크롭/10개 크롭 테스트에서 ImageNet 유효성 검사 세트의 상위 1~5위 오류율.

및 SVHN의 경우, 각 컨볼 루션 레이어(첫 번째 레이어 제외) 뒤에 드롭아웃 레이어[33]를 추가하고 드롭아웃 비율을 다음과 같이 설정합니다.

0.2. 테스트 오류는 각 작업 및 모델 설정에 대해 한 번만 평가되었습니다.

4.3. CIFAR 및 SVHN의 분류 결과

다양한 깊이(L)와 성장률(k)을 가진 DenseNet을 훈련합니다. CIFAR 및 SVHN에 대한 주요 결과는 표 2에 나와 있습니다. 일반적인 추세를 강조하기 위해 기존 최신 기술을 능가하는 모든 결과를 **굵은 글씨로 표시**하고 전체 최고 결과를 **파란색으로** 표시했습니다.

정확도. 가장 눈에 띄는 경향은 표 2의 맨 아래 줄에서 확인할 수 있는데, $L = 190$, $k = 40$ 인 DenseNet-BC가 모든 CIFAR 데이터 세트에서 기존 최신 기술을 일관되게 능가하는 것으로 나타났습니다. C10+에서 3.46%, C100+에서 17.18%의 오류율은 넓은 ResNet 아키텍처에서 달성한 오류율보다 훨씬 낮습니다[42]. C10과 C100(데이터 증강 없이)에서의 최고 결과는 더욱 고무적입니다. 두 경우 모두 드롭 경로 정규화를 사용한 프랙탈넷보다 30% 가까이 낮습니다[17]. 드롭아웃이 있는 SVHN에서, $L = 100$, $k = 24$ 의 DenseNet도 넓은 ResNet이 달성한 현재 최고 결과를 능가합니다. 그러나 250 계층의 DenseNet-BC는 더 짧은 계층에 비해 성능이 더 뛰어나지 않습니다. 이는 SVHN이 비교적 쉬운 작업이며, 매우 심층적인 모델이 훈련 세트에 과도하게 적합할 수 있기 때문일 수 있습니다.

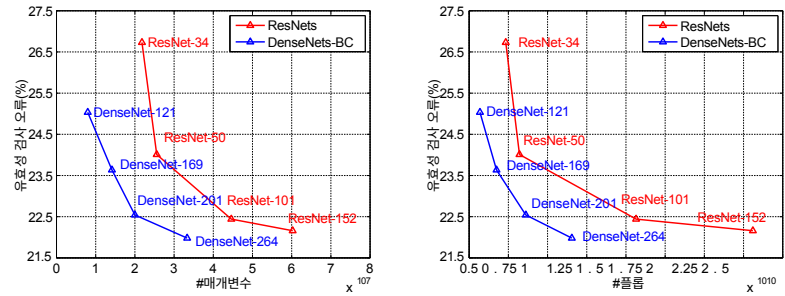


그림 3: 학습된 파라미터(왼쪽)와 테스트 시간(오른쪽)의 함수로 이미지넷 검증 데이터 세트에서 DenseNet과 ResNet의 상위 1% 오류율(단일 크롭 테스트)을 비교한 결과입니다.

용량. 압축이나 병목 현상 레이어가 없는 경우, 일반적으로 L 과 k 가 증가함에 따라 DenseNet의 성능이 향상되는 경향이 있습니다. 이는 주로 모델 용량의 상관 관계에 기인합니다. 이는 C10+와 C100+의 컬럼에서 가장 잘 드러납니다. C10+에서 오차는 5.24%에서 4.10%로 떨어지고, 매개변수 수가 1.0만 개에서 7.0만 개, 27.2만 개로 증가함에 따라 최종적으로 3.74%까지 떨어집니다. C100+에서도 비슷한 추세가 관찰됩니다. 이는 덴스넷이 더 크고 심층적인 모델의 향상된 표현력을 활용할 수 있음을 시사합니다. 또한 과적합이나 잔존 네트워크의 최적화 어려움도 겪지 않는다는 것을 나타냅니다[11].

파라미터 효율성. 표 2의 결과는 DenseNet이 대체 아키텍처(특히 ResNet)보다 파라미터를 더 효율적으로 활용한다는 것을 나타냅니다. 특히 병목 구조와 전이 계층에서의 차원 감소를 갖춘 DenseNet-BC는 파라미터 효율성이 뛰어납니다. 예를 들어, 250개 레이어 모델에는 1530만 개의 파라미터만 있지만 3,000만 개 이상의 파라미터를 가진 프랙탈넷이나 와이드 레스넷과 같은 다른 모델보다 지속적으로 뛰어난 성능을 발휘합니다. 또한 $L = 100$, $k = 12$ 인 DenseNet-BC는 90% 더 적은 수의 파라미터를 사용하여 1001개 레이어 사전 활성화 ResNet과 비슷한 성능($\mathcal{O}/C10+$ 에서 4.51% 대 4.62% 오류, $C100+$ 에서 22.27% 대 22.71% 오류)을 달성한다는 점을 강조합니다. 그림 4(오른쪽 패널)는 $C10+$ 에서 이 두 네트워크의 훈련 손실과 테스트 오류를 보여줍니다. 1001층 심층 ResNet은 더 낮은 훈련 손실 값으로 수렴하지만 테스트 오류는 비슷합니다. 이 효과는 아래에서 더 자세히 분석합니다.

과적합. 파라미터를 보다 효율적으로 사용하는 것의 긍정적인 측면 중 하나는 DenseNet이 과적합에 덜 취약한 경향이 있다는 것입니다. 데이터 증강이 없는 데이터 세트에서 이전 작업에 비해 DenseNet 아키텍처의 개선이 특히 두드러진 것을 관찰할 수 있습니다. $C10$ 에서 개선은 7.33%에서 5.19%로 29%의 상대적 오류 감소를 나타냅니다. $C100$ 에서는 28.20%에서 19.64%로 약 30% 감소했습니다. 실험에서 우리는 단일 설정에서 잠재적인 과적합을 관찰했습니다. $C10$ 에서 $k = 12$ 를 $k = 24$ 로 증가시킴으로써 생성된 파라미터의 4배 증가는 오차를 5.77%에서 5.83%로 소폭 증가시켰습니다. DenseNet-BC 병목 현상과 압축 레이어는 이러한 추세에 대응하는 효과적인 방법인 것으로 보입니다.

4.4. 이미지넷의 분류 결과

이미지넷 분류 작업에서 서로 다른 깊이와 성장률을 가진 DenseNet-BC를 평가하고, 이를 최첨단 ResNet 아키텍처와 비교합니다. 두 아키텍처 간의 공정한 비교를 위해,

공개적으로 사용 가능한 ResNet용 Torch 구현을 채택하여 데이터 전처리 및 최적화 설정의 차이와 같은 다른 모든 요소를 제거했습니다 [8].¹

¹<https://github.com/facebook/fb.resnet.torch>

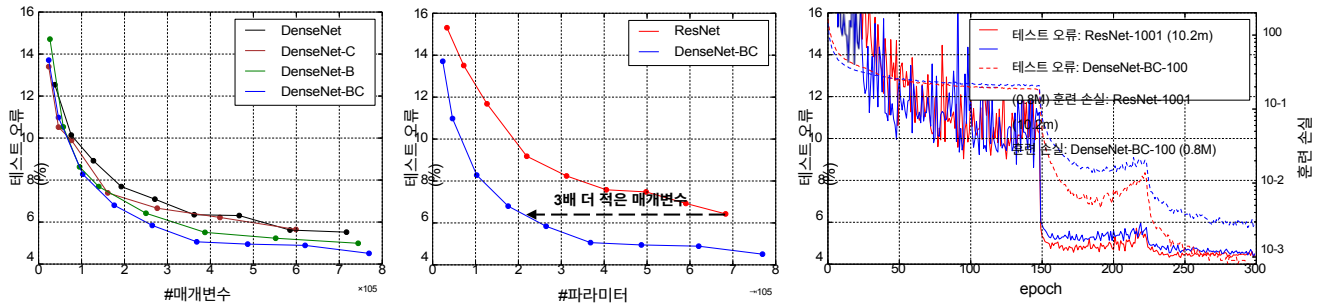


그림 4: 왼쪽: C10+의 매개변수 효율을 DenseNet 버전별로 비교. 가운데: DenseNet-BC와 (활성화 전) ResNet 간의 파라미터 효율성 비교. DenseNet-BC는 비슷한 정확도를 달성하기 위해 ResNet보다 약 1/3의 파라미터가 필요합니다. 오른쪽: 1,000만 개 이상의 파라미터가 있는 1001층 사전 활성화 ResNet[12]과 0.8만 개의 파라미터가 있는 100층 DenseNet의 훈련 및 테스트 곡선.

ResNet 모델을 DenseNet- BC 네트워크로 교체하고 모든 실험 설정을 ResNet에 사용된 것과 똑같이 유지하면 됩니다.

표 3은 ImageNet에서 DenseNets의 단일 크롭 및 10 크롭 유효성 검사 오류를 보여줍니다. 그림 3은 매개변수 수(왼쪽)와 플롭 수(오른쪽)의 함수로 DenseNets와 ResNets의 단일 크롭 상위 1단계 검증 오류를 보여줍니다. 그림에 표시된 결과를 보면 DenseNet은 최신 ResNet과 동등한 성능을 발휘하면서도 훨씬 적은 수의 파라미터와 컴퓨팅으로 비슷한 성능을 달성할 수 있음을 알 수 있습니다. 예를 들어, 2,000만 개의 파라미터가 있는 DenseNet-201은 4,000만 개 이상의 파라미터가 있는 101층 ResNet과 유사한 검증 오류를 생성합니다. 유효성 검사 오류를 FLOP 수에 따른 함수로 표시한 오른쪽 패널에서도 비슷한 경향을 볼 수 있습니다. ResNet-50만큼 많은 연산이 필요한 DenseNet은 두 배의 연산이 필요한 ResNet-101과 동등한 성능을 보입니다.

실험적 설정에서는 레스넷에는 최적화되었지만 덴스넷에는 최적화되지 않은 하이퍼파라미터 설정을 사용했다는 점에 유의할 필요가 있습니다. 보다 광범위한 하이퍼파라미터 검색을 통해 이미지넷에서 DenseNet의 성능을 더욱 향상시킬 수 있을 것으로 생각합니다.

5. 토론

식 (2)는 H_A (-)에 대한 입력이 합산되지 않고 연결되는 점에서 식 (1)과 다르지만, 표면적으로 밀집 네트워크는

ResNet과 매우 유사합니다. 그러나 이 사소한 변경으로 인해 두 네트워크 아키텍처의 동작이 크게 달라집니다.

모델 간결성. 인풋 연결의 직접적인 결과로, DenseNet 레이어 중 어느 레이어에서든 학습한 피쳐 맵을 모든 후속 레이어에서 액세스할 수 있습니다. 이는 네트워크 전체에서 피쳐 재사용을 장려하고 보다 간결한 모델로 이어집니다.

그림 4의 왼쪽 두 플롯은 모든 DenseNets 변형의 매개변수 효율성을 비교하기 위한 실험의 결과를 보여줍니다.

ResNet 아키텍처(가운데). 우리는 C10+에서 다양한 깊이를 가진 여러 개의 소규모 네트워크를 훈련하고 네트워크 매개변수의 함수로 테스트 성능을 플롯합니다. AlexNet [16] 또는 VGG-net [29]과 같이 널리 사용되는 다른 네트워크 아키텍처와 비교할 때, 사전 활성화 기능이 있는 ResNet은 더 적은 매개변수를 사용하면서도 일반적으로 더 나은 결과를 달성합니다 [12]. 따라서 이 아키텍처와 DenseNet($k = 12$)을 비교합니다. DenseNet의 훈련 설정은 이전 섹션에서와 동일하게 유지됩니다.

이 그래프는 DenseNet-BC가 지속적으로 가장 파라미터 효율성이 높은 DenseNet의 변형임을 보여줍니다. 또한 동일한 수준의 정확도를 달성하기 위해 DenseNet-BC는 ResNet의 매개변수 중 약 1/3만 다시 필요로 합니다(가운데 그래프). 이 결과는 그림 3에서 제시한 ImageNet의 결과와 일치합니다. 그림 4의 오른쪽 플롯은 학습 가능한 파라미터가 0.8백만 개에 불과한 DenseNet-BC가 1,020만 개의 파라미터를 가진 1001층(활성화 전) ResNet [12]과 비슷한 정확도를 달성할 수 있음을 보여줍니다.

암시적 심층 감독. 고밀도 컨볼루션 네트워크의 정확도가 입증된 이유 중 하나는 개별 레이어가 짧은 연결을 통해 손실 함수로부터 추가적인 감독을 받기 때문일 수 있습니다. 일종의 '심층 감독'을 수행한다고 해석할 수도 있습니다. 심층 감독의 이점은 이전에 모든 숨겨진 레이어에 분류기가 부착되어 중간 레이어가 차별적 특징을 학습하도록 하는 심층 감독 네트워크(DSN; [20])에서 입증된 바 있습니다.

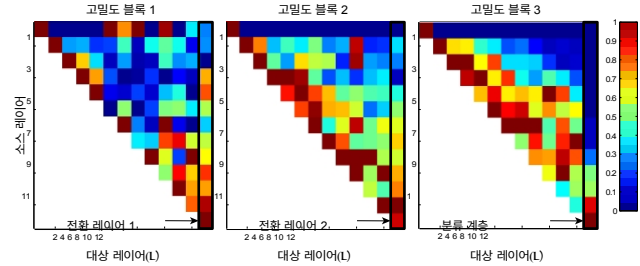
밀집 네트워크는 네트워크 상단에 있는 단일 분류기가 최대 두세 개의 전이 계층을 통해 모든 계층에 직접 감독을 제공하는 방식으로 유사한 심층 감독을 수행합니다. 하지만, 모든 레이어에서 동일한 손실 함수를 공유하기 때문에 DenseNets의 손실 함수와 그래디언트는 훨씬 덜 복잡합니다.

확률론적 대 결정론적 연결. 고밀도 컨볼루션 네트워크와 잔여 네트워크의 확률적 깊이 정규화 사이에는 흥미로운 연관성이 있습니다 [13]. 확률적 깊이에서는 잔여 네트워크의 레이어가 무작위로 삭제되어 직접 연결이 생성됩니다.

주변 레이어를 연결합니다. 풀링 레이어가 삭제되지 않기 때문에 네트워크는 DenseNet과 유사한 연결 패턴을 갖게 됩니다. 즉, 모든 중간 레이어가 무작위로 삭제될 경우 동일한 풀링 레이어 사이에 있는 두 레이어가 직접 연결될 확률은 거의 없습니다. 궁극적으로 방법은 상당히 다르지만, 확률적 깊이에 대한 DenseNet의 해석은 이 정규화의 성공에 대한 통찰력을 제공할 수 있습니다.

피쳐 재사용. 설계상, 덴스넷은 레이어가 모든 이전 레이어의 피쳐 맵에 액세스할 수 있도록 허용합니다(때로는 트랜지션 레이어를 통해). 훈련된 네트워크가 이러한 기회를 잘 활용하는지를 조사하기 위해 실험을 진행했습니다. 먼저 $L = 40, k = 12$ 인 C10+에서 DenseNet을 훈련합니다. 블록 내의 각 컨볼루션 레이어 l 에 대해 레이어 s 와의 연결에 할당된 평균(절대) 가중치를 계산합니다. 그림 5는 세 개의 고밀도 블록 모두에 대한 히트 맵을 보여줍니다. 평균 절대 가중치는 컨볼루션 레이어가 이전 레이어에 대한 종속성을 대신하는 역할을 합니다. 위치 (l, s) 의 빨간색 점은 레이어 l 이전에 생성된 피쳐 맵을 평균적으로 많이 사용한다는 것을 나타냅니다. 이 플롯에서 몇 가지 관찰을 할 수 있습니다:

1. 모든 레이어는 동일한 블록 내의 많은 입력에 가중치를 분산시킵니다. 이는 매우 초기 레이어에서 추출한 특징이 실제로 동일한 밀도 블록의 심층 레이어에서 직접 사용됨을 나타냅니다.
2. 트랜지션 레이어의 가중치도 이전 밀집 블록 내의 모든 레이어에 분산되어 있어, 몇 가지 간접 경로를 통해 밀집 네트워크의 첫 번째 레이어에서 마지막 레이어까지 정보가 흐르고 있음을 나타냅니다.
3. 두 번째와 세 번째 고밀도 블록 내의 레이어는 트랜지션 레이어(삼각형의 맨 윗줄)의 출력에 일관되게 가장 낮은 가중치를 할당하여 트랜지션 레이어가 평균적으로 낮은 가중치를 가진 많은 재연산 피쳐를 출력한다는 것을 나타냅니다. 이는 정확히 이러한 출력이 압축되는 DenseNet-BC의 강력한 결과와 일치합니다.



4. 맨 오른쪽에 표시된 최종 분류 계층도 전체 고밀도 블록에 걸쳐 가중치를 사용하지만, 최종 피쳐 맵에 집중되어 있는 것으로 보아 네트워크 후반에 생성된 더 높은 수준의 피쳐가 있을 수 있음을 시사합니다.

6. 결론

저희는 새로운 컨볼루션 네트워크 아키텍처를 제안했는데, 이를 밀도 컨볼루션 네트워크(DenseNet)라고 부릅니다. 이 네트워크는 피쳐 맵 크기가 같은 두 레이어 사이에 직접 연결을 도입합니다. 실험 결과, DenseNet은 수백 개의 레이어까지 자연스럽게 확장하면서도 최적화 문제를 일으키지 않는 것으로 나타났습니다. 실험 결과

그림 5: 훈련된 DenseNet에서 컨볼루션 레이어의 평균 절대 필터 가중치. 픽셀(s, l)의 색상은 고밀도 블록 내에서 컨볼루션 레이어 s 와 l 을 연결하는 가중치의 평균 LI 규범(입력 피쳐 맵의 수로 정규화)을 인코딩합니다. 검은색 직사각형으로 강조 표시된 세 개의 열은 두 개의 전이 레이어와 분류 레이어에 해당합니다. 첫 번째 행은 고밀도 블록의 입력 레이어에 연결된 가중치를 인코딩합니다.

밀집 네트워크는 성능 저하나 과적합의 징후 없이 매개변수 수가 증가함에 따라 정확도가 지속적으로 향상되는 경향이 있습니다. 다양한 설정에서 경쟁이 치열한 여러 데이터 세트에서 최첨단 결과를 달성했습니다. 또한, DenseNet은 최신 성능을 달성하기 위해 훨씬 적은 수의 파라미터와 더 적은 컴퓨팅이 필요합니다. 본 연구에서는 잔여 네트워크에 최적화된 하이퍼파라미터 설정을 채택했기 때문에, 하이퍼파라미터와 학습 속도 스케줄을 더욱 세밀하게 조정하면 DenseNet의 정확도를 더욱 향상시킬 수 있을 것으로 예상됩니다.

단순한 연결 규칙을 따르면서도 DenseNets는 ID 매핑, 심층 감독, 다양한 깊이의 속성을 자연스럽게 통합합니다. 네트워크 전체에서 피쳐를 재사용할 수 있으므로 결과적으로 더 간결한 모델을 학습할 수 있으며, 실험에 따르면 더 정확한 모델을 학습할 수 있습니다. 내부 표현이 간결하고 피쳐 중복성이 감소하기 때문에 DenseNets는 컨볼루션 피쳐를 기반으로 하는 다양한 컴퓨터 비전 작업([4, 5])에 적합한 피쳐 추출기가 될 수 있습니다. 앞으로의 작업에서 이러한 특징 전송에 대해 DenseNets를 사용해 연구할 계획입니다.

감사의 말씀. 저자들은 NSF III-1618134, III-1526012, IIS-1149882, Of.

해군 연구 보조금 N00014-17-1-2175 및 빌 앤 멜린다 게이트 재단의 지원을 받았습니다. GH는 중국 박사후위원회(20150015)의 국제 박사후 교류 펠로우십 프로그램의 지원을 받았습니다. ZL은 중국 국가 기초 연구 프로그램 보조금 2011CBA00300, 2011CBA00301, NSFC의 지

원을 받았습니다.

61361136003. 또한 많은 통찰력 있는 토론을 해주신 다니엘 세드라, 제프 플레이스, 유 선에게도 감사드립니다.

참조

- [1] C. 코르테스, X. 곤잘보, V. 쿠즈네초프, M. 모리, 및 S. 양 Adanet: 인공 신경망의 적응적 구조 학습. *arXiv preprint arXiv:1607.01097*, 2016. 2

- [2] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, 및 L. Fei-fei. Imagenet: 대규모 계층적 이미지 데이터베이스. In *CVPR*, 2009. 5
- [3] S. E. Fahlman and C. Lebiere. 캐스케이드 상관관계 학습ing 아키텍처. In *NIPS*, 1989. 2
- [4] J. R. Gardner, M. J. Kusner, Y. Li, P. Upchurch, K. Q. Weinberger 및 J. E. Hopcroft. 심층 다양체 탐색: 컨볼루션 특징으로 레이블 변경. *arXiv 사전 인쇄* *arXiv:1511.06421*, 2015. 8
- [5] L. Gatys, A. Ecker, and M. Bethge. 예술적 스타일의 신경 알고리즘. *Nature Communications*, 2015. 8
- [6] X. 글로트, A. 보르데스, Y. 벤지오. 딥 스파스 정류기 신경망. In *AISTATS*, 2011. 3
- [7] I. 굿펠로우, D. 워드-팔리, M. 미르자, A. 쿠르빌, 및 Y. 벤지오. 맥스아웃 네트워크. In *ICML*, 2013. 5
- [8] S. Gross and M. Wilber. 잔류 그물 훈련 및 조사, 2016. 5, 7
- [9] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. 객체 세분화 및 세분화된 현지화를 위한 하이퍼 컬럼. In *CVPR*, 2015. 2
- [10] K. He, X. Zhang, S. Ren, J. Sun. 정류기에 대해 깊이 파고 들기: imagenet 분류에서 인간 수준의 성능을 뛰어넘다. In *ICCV*, 2015. 5
- [11] K. He, X. Zhang, S. Ren, and J. Sun. 이미지 인식을 위한 심층 잔여 학습. In *CVPR*, 2016. 1, 2, 3, 4, 5, 6
- [12] K. He, X. Zhang, S. Ren, and J. Sun. 심층 잔여 네트워크의 신원 매핑. In *ECCV*, 2016. 2, 3, 5, 7
- [13] G. Huang, Y. Sun, Z. Liu, D. Sedra, 및 K. Q. 와인버거. 확률적 깊이를 가진 심층 네트워크. In *ECCV*, 2016. 1, 2, 5, 8
- [14] S. 이오페와 C. 세게디. 일괄 정규화: 내부 공변량 이동을 줄임으로써 심층 네트워크 훈련 가속화. In *ICML*, 2015. 3
- [15] A. 크리제프스키와 G. 힌튼. 작은 이미지에서 기능의 여러 레이어 학습하기. *기술 보고서*, 2009. 5
- [16] A. 크리제프스키, I. 수츠케버, 및 G. E. 힌튼. 심층 컨볼루션 신경망을 이용한 이미지넷 분류. In *NIPS*, 2012. 3, 7
- [17] G. 라르손, M. 마이어, G. 샤크나로비치. 프랙탈넷: 잔여물이 없는 초심층 신경망. *arXiv 사전 인쇄* *arXiv:1605.07648*, 2016. 1, 3, 5, 6
- [18] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, L. D. Jackel. 필기 우편번호 인식에 적용된 역전파. *Neural computation*, 1(4):541-551, 1989. 1
- [19] Y. 르쿤, L. 보투, Y. 벤지오, 및 P. 하프너. 문서 인식에 적용된 그라디언트 기반 학습. *Proceed- IEEE*, 86(11):2278-2324, 1998. 1, 3
- [20] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu. Deeply-감독된 그물. In *AISTATS*, 2015. 2, 3, 5, 7
- [21] Q. 리아오와 T. 포지오. 잔여 학습, 순환 신경망 및 시각 피질 사이의 격차 해소. *arXiv 사전 인쇄* *arXiv:1604.03640*, 2016. 2
- [22] M. Lin, Q. Chen, and S. Yan. 네트워크 내 네트워크. In *ICLR*, 2014. 3, 5
- [23] J. Long, E. Shelhamer, 및 T. Darrell. 의미론적 세분화를 위한 완전 컨볼루션 네트워크. In *CVPR*, 2015. 2

- [24] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng. 비지도 인공지능망 학습으로 자연 이미지에서 숫자 읽기, , 2011. In *NIPS Workshop*, 2011. 5
- [25] M. 페제스키, L. 팬, P. 브레이켈, A. 쿠르빌, Y. 벤지오. 래더 네트워크 아키텍처 해체. In *ICML*, 2016. 3
- [26] G. Pleiss, D. Chen, G. Huang, T. Li, L. van der Maaten, 및 K. Q. Weinberger. densenets의 메모리 효율적인 구현. *arXiv preprint arXiv:1707.06990*, 2017. 5
- [27] A. 라스무스, M. 베르글룬드, M. 혼칼라, H. 발폴라, 및 T. 라이코. 사다리 네트워크를 사용한 반지도 학습. *NIPS*, 2015. 3
- [28] A. 로메로, N. 발라스, S. E. 카호우, A. 차상, C. 가타, Y. 벤지오. 핏넷: 얇고 깊은 그물을 위한 힌트. In *ICLR*, 2015. 5
- [29] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein 외. Imagenet 대규모 시각 인식 챌린지. *IJCV*. 1, 7
- [30] P. Sermanet, S. Chintala, and Y. LeCun. 집 번호 자릿수 분류에 적용된 컨볼루션 신경망. In *ICPR*, 3288-3291 페이지. IEEE, 2012. 5
- [31] P. Sermanet, K. Kavukcuoglu, S. Chintala, 및 Y. LeCun. 비지도 다단계 기능을 사용한 보행자 감지 학습. In *CVPR*, 2013. 2
- [32] J. T. 스프링겐버그, A. 도소비츠키, T. 브록스, M. 리드밀러. 단순성을 위한 노력: *arXiv preprint arXiv:1412.6806*, 2014. 5
- [33] N. 스리바스타바, G. E. 힌튼, A. 크리제프스키, I. 수츠키버, 및 R. 살라쿠트디노프. 드롭아웃: 신경망 네트워크의 과적합을 방지하는 간단한 방법. *JMLR*, 2014. 6
- [34] R. K. 스리바스타바, K. 그레프, J. 슈미드huber. 교육 매우 깊은 네트워크. In *NIPS*, 2015. 1, 2, 5
- [35] I. Sutskever, J. Martens, G. Dahl, 및 G. Hinton. 딥러닝에서 초기화와 모멘텀의 중요성에 대해. In *ICML*, 2013. 5
- [36] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. 안젤로프, D. 에르한, V. 반호케, 및 A. 라비노비치. 컨볼루션으로 더 깊이 들어가기. *CVPR*, 2015. 2, 3
- [37] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna. 컴퓨터 비전을 위한 시작 아키텍처 재고. In *CVPR*, 2016. 2, 3, 4
- [38] S. 타그, D. 알메이다, 및 K. 라이먼. 리셋에서 리셋: 잔여 아키텍처 일반화. *arXiv 사전 인쇄본 arXiv:1603.08029*, 2016. 2
- [39] J. Wang, Z. Wei, T. Zhang, and W. Zeng. 깊게 융합된 그물. *arXiv 사전 인쇄본 arXiv:1605.07716*, 2016. 3
- [40] B. M. Wilamowski and H. Yu. 역전파 없는 신경망 학습. *IEEE 신경망 트랜잭션-works*, 21(11):1793-1803, 2010. 2
- [41] S. 양과 D. 라마난. dag- cnns를 사용한 다중 스케일 인식. In *ICCV*, 2015. 2
- [42] S. 자가루이코와 N. 코모다키스. 광범위한 잔여 네트워크. *arXiv 사전 인쇄본 arXiv:1605.07146*, 2016. 3, 5, 6
- [43] Y. Zhang, K. Lee, and H. Lee. 대규모 이미지 분류를 위한 비지도 목표로 지도 신경망 강화하기. In *ICML*, 2016. 3