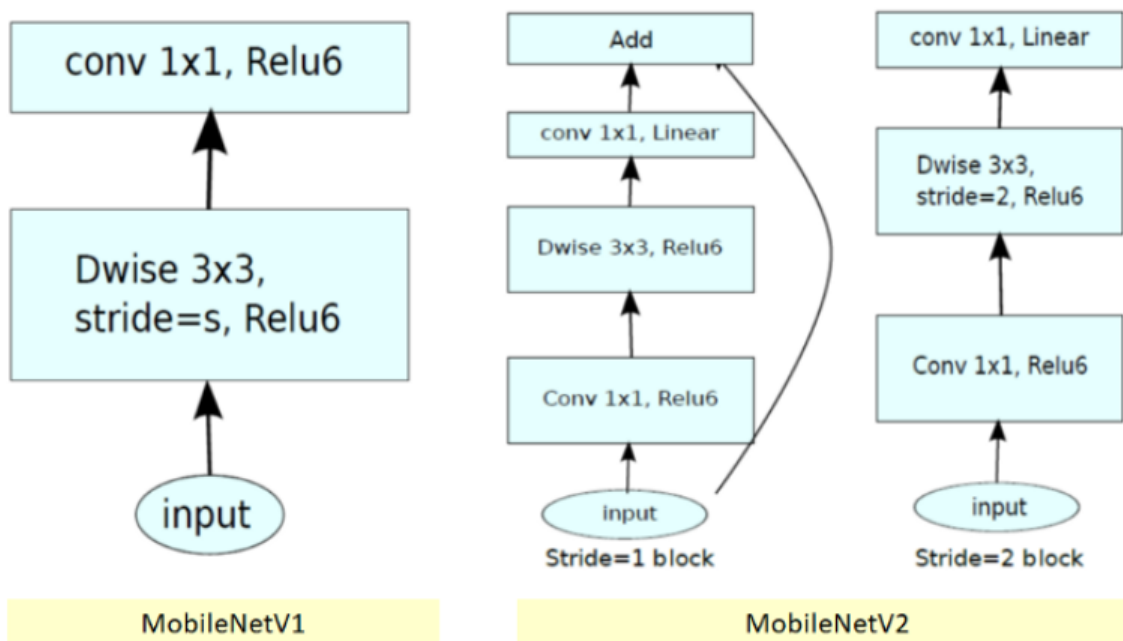


# 5

## MobileNet

태그

### MobileNet 정의



MobileNet은 2017년 Google의 연구원들이 개발한 CNN(Convolutional Neural Network) 아키텍처입니다. 모바일 기기나 엣지 컴퓨팅과 같이 리소스가 제한된 환경에서 사용할 수 있도록 효율적이고 가볍게 설계되었습니다.

MobileNet 아키텍처는 일련의 컨볼루션 계층과 완전히 연결된 계층으로 구성됩니다. 컨볼루션 레이어는 깊이별 컨볼루션 레이어와 점별 컨볼루션 레이어로 구성된 "깊이별 분리 가능 컨볼루션" 블록 스택으로 구성됩니다. 깊이별 컨볼루션 레이어는 공간 특징을 학습하고 포인트별 컨볼루션 레이어는 채널별 특징을 학습합니다.

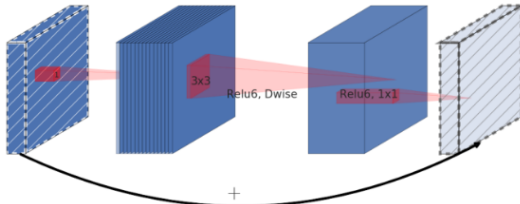
MobileNet은 네트워크가 공간적 특징과 채널별 특징을 개별적으로 학습할 수 있도록 하는 깊이별 분리 가능 컨볼루션 개념을 기반으로 합니다. 이것은 여전히 좋은 정확도를 유지하면서 네트워크의 매개변수 수와 계산 비용을 줄이는 데 도움이 됩니다.

MobileNet은 다양한 크기로 제공되며 각 크기는 정확도와 효율성 사이의 서로 다른 균형에 맞게 최적화되어 있습니다. 예를 들어 MobileNet-V1은 MobileNet의 원래 버전으로 상대적으로 작고 효율적이며 MobileNet-V3는 더 크고 정확한 버전입니다.

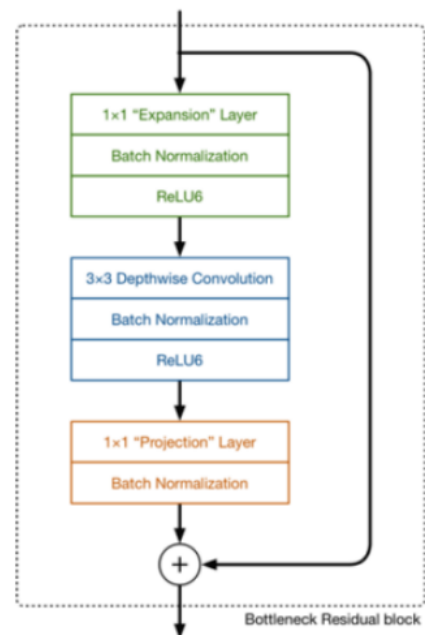
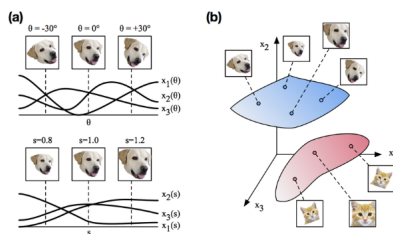
MobileNet은 네트워크가 기존 컨볼루션보다 더 적은 매개변수와 더 낮은 계산 비용을 사용하여 컨볼루션을 수행할 수 있게 해주는 깊이별 분리 가능 컨볼루션을 사용하는 것이 특징입니다. 또한 모델 크기와 계산 비용을 더욱 줄이기 위해 선형 병목 현상 및 배치 정규화와 같은 여러 다른 기술을 사용합니다.

MobileNet은 이미지 분류, 객체 감지 및 의미론적 분할을 포함한 광범위한 작업에서 성공했으며 산업 및 연구에서 널리 채택되었습니다. 또한 Xception 및 MnasNet과 같이 깊이 분리 가능한 컨볼루션을 사용하는 여러 다른 네트워크 아키텍처에 영감을 주었습니다.

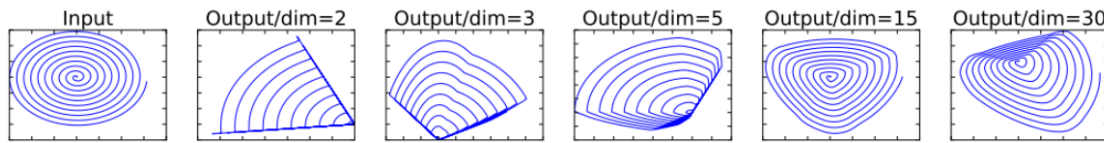
## (b) Inverted residual block



- Inverted Residual block은 일반적인 Residual block과는 정 반대인 narrow - wide - narrow한 형태를 하고있다.
- 이렇게 시도를 한 이유는 narrow에 해당하는 저차원의 layer에는 필요한 정보만 압축되어서 저장되어 있다라는 가정으로부터 나왔다. 따라서 필요한 정보는 narrow에 있기 때문에 skip connection으로 사용해두 필요한 정보를 더 깊은 layer까지 전달할 것이라는 기대를 할 수 있다



- 즉, 저차원으로 매핑하는 bottleneck architecture(projection convolution)를 만들 때, **linear transformation** 역할을 하는 **linear bottleneck layer(Do n't use ReLU)**를 만들어서 차원은 줄이되 manifold 상의 중요한 정보들은 그대로 유지해보자는 것이 컨셉이다.



- ReLU는 0이하의 값을 0으로 만들기때문에 그만큼의 정보손실이 있을수밖에 없다.
- 위 그림에서 볼 수 있듯이 저차원에서의 맵핑 정보손실이 더 크고 그 의미는 ReLU를 차원수가 충분히 큰 공간에서 사용하게 된다면 그만큼 정보 손실율일 낮아진다는 것을 의미한다.

Google의 연구원이 개발한 여러 버전의 MobileNet 아키텍처가 있습니다. MobileNet의 각 버전은 모델 크기, 매개변수 수 및 정확도의 특정 조합이 특징입니다.

다음은 MobileNet의 주요 버전에 대한 요약입니다.

**MobileNet-V1:** 2017년 Google에서 개발한 MobileNet의 원본 버전입니다. 깊이별 분리 가능 컨볼루션을 사용하는 것이 특징이며 다양한 크기로 제공되며 각 크기는 서로 다른 절충에 최적화되어 있습니다. 2017년 Google에서 개발한 MobileNet의 원본 버전입니다. 모델 크기가 약 400만 매개변수인 비교적 작고 효율적인 모델입니다. 계산 리소스가 제한된 응용 프로그램에 매우 적합합니다.

**MobileNet-V2:** 2018년 Google에서 개발한 MobileNet의 수정된 버전입니다. 여기에는 역 잔차 블록 및 선형 병목 현상을 사용하는 등 MobileNet-V1에 비해 여러 가지 개선 사항이 포함되어 있어 모델의 정확도와 성능 개선되었습니다. 2018년 Google에서 개발한 MobileNet의 약간 더 크고 더 정확한 버전입니다. 반전 잔차 블록과 선형 병목 현상을 사용하여 모델의 정확도와 학습 안정성을 향상시키는 것이 특징입니다. 모델 크기는 약 340만 매개변수입니다.

**MobileNet-V3:** 2019년 Google에서 개발한 MobileNet의 추가 수정 버전입니다. 더 효율적인 컨볼루션 블록 사용 및 hard-swish 활성화 기능 사용과 같은 MobileNet-V2에 대한 추가 개선 사항이 포함되어 있습니다. 모델의 정확성과 효율성을 더욱 향상시키는 데 도움이 됩니다. 2019년 Google에서 개발한 MobileNet의 더 크고 정확한 버전입니다. "MobileNet-V3 large"라는 MobileNet 아키텍처의 수정된 버전을 사용하는 것이 특징이며, 모델 크기가 약 540만 개의 매개변수. 또한 "MobileNet-V3 small"이라는 MobileNet-V3 아키텍처의 더 작은 버전이 포함되어 있으며 모델 크기는 약 250만 매개변수입니다.

이들은 개발된 다양한 MobileNet 버전의 몇 가지 예에 불과하다는 점에 주목할 가치가 있습니다. MobileNet-V4 및 MobileNet-V5와 같이 서로 다른 아키텍처를 갖고 있으며 서로 다른 작업에 최적화된 많은 MobileNet 변형이 제안되었습니다.