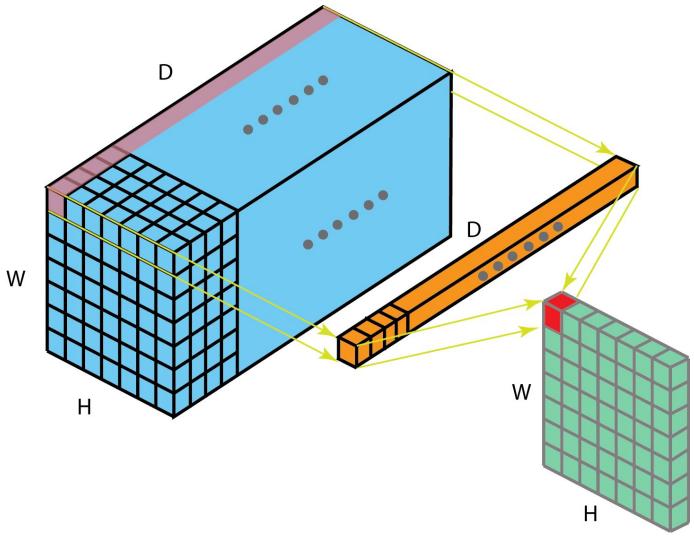
Стркуктура (4 слайд)
Mobilenet состоит из одного обычного сверточного слоя с ядром 3x3 в начале и 13 блоков, изображенных справа, с постепенно увеличивающимся числом фильтров и понижающейся пространственной размерностью тензора.

Особенностью данной архитектуры является отсутствие max pooling-слоёв. Вместо них для снижения пространственной размерности используется свёртка с параметром **stride**, равным 2.

Strides: Целое число или кортеж/перечень одного целого, задающего длину шага свертки, количество сегментов, по которым одновременно проходится фильтр. Более широкие шаги обычно помогают уменьшать объём вычислений, обобщать результаты изучения признаков и т. д.
Тут можно посмотреть gif, показывающую принцип работы:
https://distill.pub/2016/deconv-checkerboard/

Пакетная нормализация (англ. batch-normalization) — метод, который позволяет повысить производительность и стабилизировать работу искусственных нейронных сетей. Суть данного метода заключается в том, что некоторым слоям нейронной сети на вход подаются данные, предварительно обработанные и имеющие нулевое математическое ожидание и единичную дисперсию.

1x1 свертка:
На следующем рисунке показано, как свертка 1 x 1 работает для входного слоя с размером H x W x D. После свертки 1 x 1 с размером фильтра 1 x 1 x D выходной канал имеет размерность H x W x 1.



Deepwise separable convolution (5 слайд)
Обычный сверточный слой одновременно обрабатывает как пространственную информацию (корреляцию соседних точек внутри одного канала), так и межканальную информацию, так как свёртка применяется ко всем каналам сразу.

Deepwise convolution базируется на предположении о том, что эти два вида информации можно обрабатывать последовательно без потери качества работы сети, и раскладывает обычную свёртку на pointwise convolution (которая обрабатывает только межканальную корреляцию) и spatial convolution (которая обрабатывает только пространственную корреляцию в рамках отдельного канала).

Обычный сверточный слой применяет ядро свертки (или “фильтр") ко всем каналам входного изображения. Он скользит этим ядром по изображению и на каждом шаге выполняет взвешенную сумму входных пикселей, покрытых ядром по всем входным каналам.

Важно то, что операция свертки объединяет значения всех входных каналов. Если изображение имеет 3 входных канала, то запуск одного ядра свертки через это изображение приводит к выходному изображению только с 1 каналом на пиксель.

Таким образом, для каждого входного пикселя, независимо от того, сколько каналов он имеет, свертка записывает новый выходной пиксель только с одним каналом. (На практике мы запускаем множество ядер свертки по входному изображению. Каждое ядро получает свой собственный канал на выходе.)

В отличие от обычной свертки Deepwise не объединяет входные каналы, а выполняет свертку на каждом канале отдельно. Для изображения с 3 каналами Deepwise свертка создает выходное изображение, которое также имеет 3 канала. Каждый канал получает свой собственный набор весов.

Обычная свертка выполняет как фильтрацию, так и объединение за один раз, но при разделяемой по глубине свертке эти две операции выполняются как отдельные шаги.

Для ядер 3×3 этот новый подход примерно в 9 раз быстрее и все еще эффективен. Поэтому неудивительно, что MobileNets использует до 13 deepwise separate сверток подряд.

Обычная свертка представляет из себя фильтр $Dk * Dk * Cin$, где Dk — это размер ядра свёртки, а Cin — количество каналов на входе. Общая вычислительная сложность сверточного слоя составляет $Dk * Dk * Cin * Df * Df * Cout$, где Df — это высота и ширина слоя (мы считаем, что пространственные размеры входного и выходного тензоров совпадают), а Cout — число каналов на выходе.

Идея depthwise separable convolution состоит в том, чтобы разложить подобный слой на depthwise-свертку, которая представляет из себя поканальный фильтр, и 1x1-свёртку (также называемую pointwise convolution). Суммарное количество операций для применения такого слоя равно (Dk ∗ Dk+Cout) ∗ Cin ∗ Df ∗ Df.

Гиперпараметры
Двумя гиперпараметрами архитектуры MobileNet являются α (множитель ширины) и ρ (множитель глубины или множитель разрешения).

Множитель ширины отвечает за количество каналов в каждом слое. Например, α=1 даёт нам архитектуру, описанную в статье, а α=0.25 — архитектуру с уменьшенным в четыре раза числом каналов на выходе каждого блока.

Множитель разрешения отвечает за пространственные размеры входных тензоров. Например, ρ=0.5 означает, что высота и ширина feature map, подаваемой на вход каждому слою будет уменьшена вдвое.

Оба параметра позволяют варьировать размеры сети: уменьшая α и ρ, мы снижаем точность распознавания, но в то же время увеличиваем скорость работы и уменьшаем потребляемую память.

Ссылки из презентации:
- https://arxiv.org/abs/1704.04861
- https://machinethink.net/blog/googles-mobile-net-architecture-on-iphone/
- https://habr.com/ru/post/352804/
- https://habr.com/ru/post/347564/