Сверточные нейронные сети (convolutional neural networks, CNN) — это весьма широкий класс архитектур, основная идея которых состоит в том, чтобы переиспользовать одни и те же части нейронной сети для работы с разными маленькими, локальными участками входов.

Как и многие другие нейронные архитектуры, сверточные сети известны довольно давно, и в наши дни у них уже нашлось много самых разнообразных применений, но основным приложением, ради которого люди когда-то придумали сверточные сети, остается *обработка изображений*.

Аффинные преобразования. В каждом слое полносвязной сети повторяется одна и та же операция: на вход подается вектор, который умножается на матрицу весов, а к результату добавляется вектор свободных членов; только после этого к результату применяется некая нелинейная функция активации.

Однако многие типы данных имеют свою собственную внутреннюю структуру, которая отлично известна нам заранее. Главный пример такой структуры в этой главе — изображение, которое обычно представляют как массив векторов чисел: если изображение черно-белое, то это просто массив интенсивностей, а если цветное, то массив векторов из трех чисел, обозначающих интенсивности трех основных цветов (красного, зеленого и черного в стандартном RGB, синего, зеленого и красного в трех типах колбочек в человеческом глазе и т. д.). Если же обобщить такую внутреннюю структуру до максимальной все еще полезной нам общности, описание получится такое:

1) исходные данные представляют собой многомерный массив («тензор»);

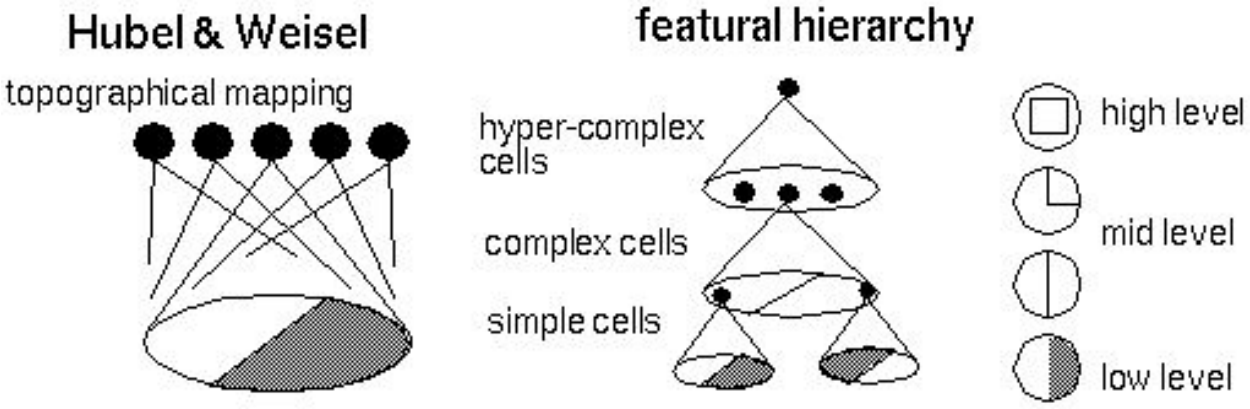
2) среди размерностей этого массива есть одна или более осей, порядок вдоль которых играет важную роль; например, это может быть расположение пикселов в изображении, временная шкала для музыкального произведения, порядок слов или символов в тексте;

3) другие оси обозначают «каналы», описывающие свойства каждого элемента по предыдущему подмножеству осей; например, три компонента для изображений, два компонента (правый и левый) для стереозвука и т. д.

Когда мы обучаем полносвязные нейронные сети, это дополнительное знание о структуре задачи никак не используется. Вспомним пример сети для распознавания рукописных цифр, которую мы строили в разделе 3.6: там мы просто превращали изображение размера 28 _ 28 пикселов в вектор длины 784 и подавали его на вход. Получалось, что наши аффинные преобразования никак не учитывали структуру картинки, топологию данных!

Основная идея сверточной сети состоит в том, что обработка участка изображения очень часто должна происходить независимо от конкретного расположения этого участка.

Мозг обрабатывает визуальную информацию иерархически: сначала находят границы, углы, а на более глубоких слоях – сложные объекты.



Теперь осталось только формально определить, что же такое свертка и как устроены слои сверточной сети. Свертка — это всего лишь линейное преобразование входных данных особого вида. Если x^l – карта признаков в слое под номером l, то результат двумерной свертки с ядром размера $2d+1$ и матрицей весов W размера $(2d+1)\times(2d+1)$ на следующем слое будет таким:

$$y_{i,j}^l = \sum_{-d\le a,b\le d} W_{a,b} x_{i+a,j+b}^l,$$

где $y_{i,j}^l$ – результат свертки на уровне l, а $x_{i,j}^l$ – ее вход, то есть выход всего предыдущего слоя.

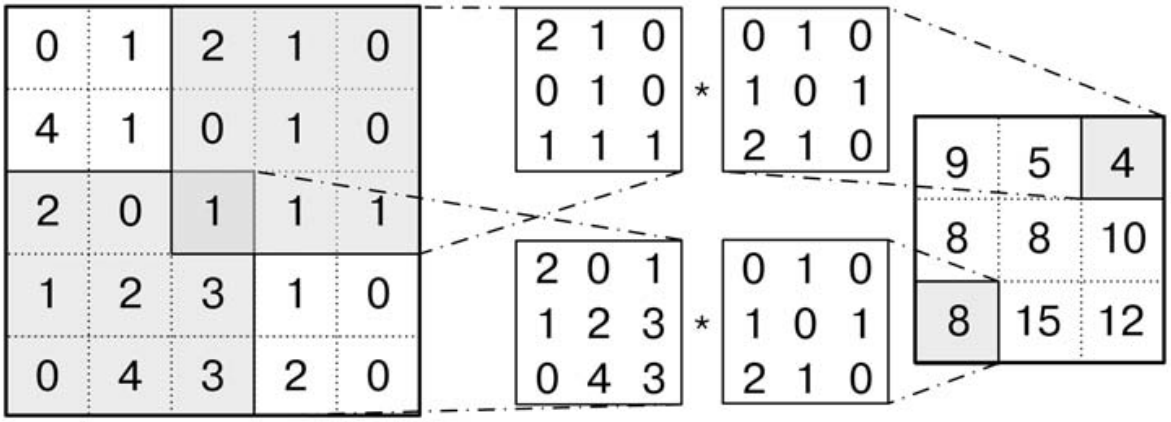


Рисунок 1. Пример подсчета результата свертки: два примера подматрицы и общий результат

$$\begin{pmatrix} 0 & 1 & 2 & 1 & 0 \\ 4 & 1 & 0 & 1 & 0 \\ 2 & 0 & 1 & 1 & 1 \\ 1 & 2 & 3 & 1 & 0 \\ 0 & 4 & 3 & 2 & 0 \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 2 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 9 & 5 & 4 \\ 8 & 8 & 10 \\ 8 & 15 & 12 \end{pmatrix}$$

Обратите внимание, что умножение подматрицы исходной матрицы X, соответствующей окну, и матрицы весов W — это не умножение матриц, а просто скалярное произведение соответствующих векторов. А всего окно умещается в матрице X девять раз.

Это преобразование обладает как раз теми свойствами, о которых мы говорили выше:

• свертка сохраняет структуру входа (порядок в одномерном случае, взаимное расположение пикселов в двумерном и т. д.), так как применяется к каждому участку входных данных в отдельности;

• операция свертки обладает свойством разреженности, так как значение каждого нейрона очередного слоя зависит только от небольшой доли входных нейронов (а, например, в полносвязной нейронной сети каждый нейрон зависел бы от всех нейронов предыдущего слоя);

• свертка многократно переиспользует одни и те же веса, так как они повторно применяются к различным участкам входа.

После свертки в нейронной сети используется нелинейность:

$$z_{i,j}^{l} = h(y_{i,j}^{l})$$

В качестве функции h часто используют ReLU.

В классическом сверточном слое, кроме линейной свертки и следующей за ней нелинейности, есть и еще одна операция: *субдискретизация* (pooling; по-русски ее иногда называют еще операцией «подвыборки», от альтернативного английского термина subsampling). Смысл субдискретизации прост: в сверточных сетях обычно исходят из предположения, что наличие или отсутствие того или иного признака гораздо важнее, чем его точные координаты. Например, при распознавании лиц сверточной сетью нам гораздо важнее понять, есть ли на фотографии лицо и чье, чем узнать, с какого конкретно пиксела оно начинается и в каком заканчивается. Поэтому можно позволить себе «обобщить» выделяемые признаки, потеряв часть информации

об их местоположении, но зато сократив размерность. формально станем определять субдискретизацию (в тех же обозначениях, что выше) так:

$$x_{i,j}^{l+1} = \max_{-d \le a \le d, -d \le b \le d} z_{i+a,j+b}^{l}$$

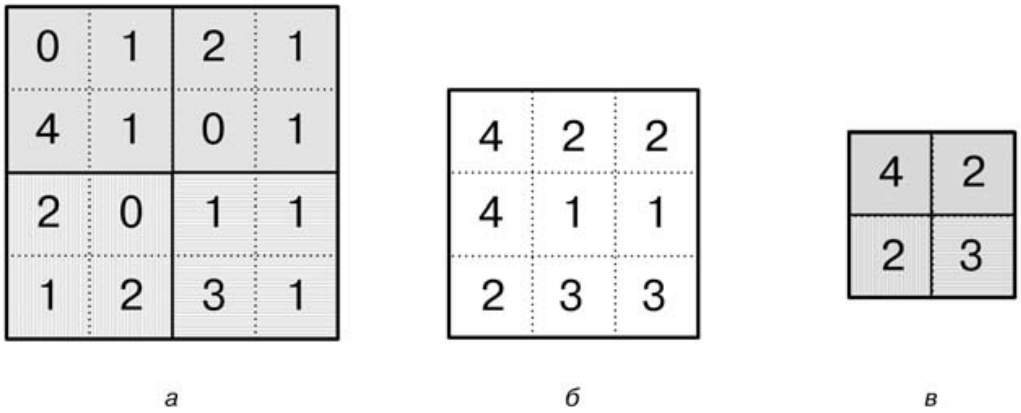


Рисунок 2. Пример субдискретизации с окном размера 2 × 2; *а* — исходная матрица; *б* — матрица после субдискретизации с шагом 1; *в* — матрица после субдискретизации с шагом 2. Штриховка в исходной матрице *а* — соответствует окнам, по которым берется максимум с шагом 2; в части *в* — результат показан соответствующей штриховкой

Стандартный слой сверточной сети состоит из трех компонентов:

• свертка в виде линейного отображения, выделяющая локальные признаки;

• нелинейная функция, примененная покомпонентно к результатам свертки;

• субдискретизация, которая обычно сокращает геометрический размер получающихся тензоров.

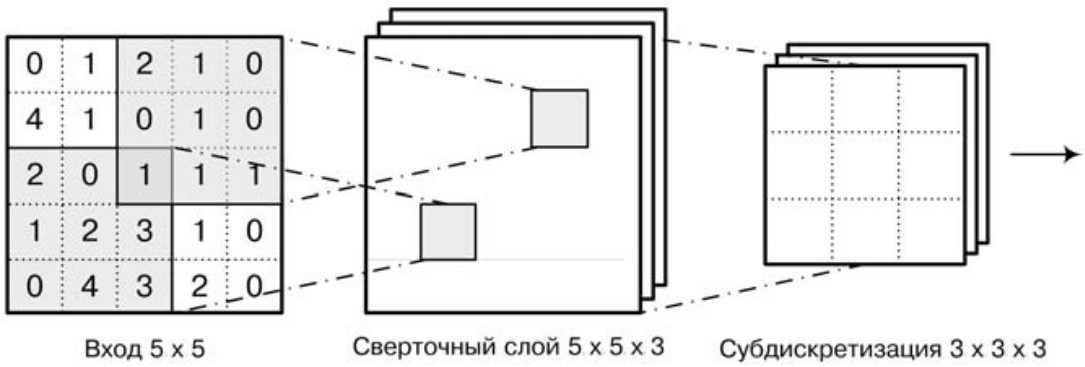


Рисунок 3. Схема одного слоя сверточной сети: свертка, за которой следует субдискретизация

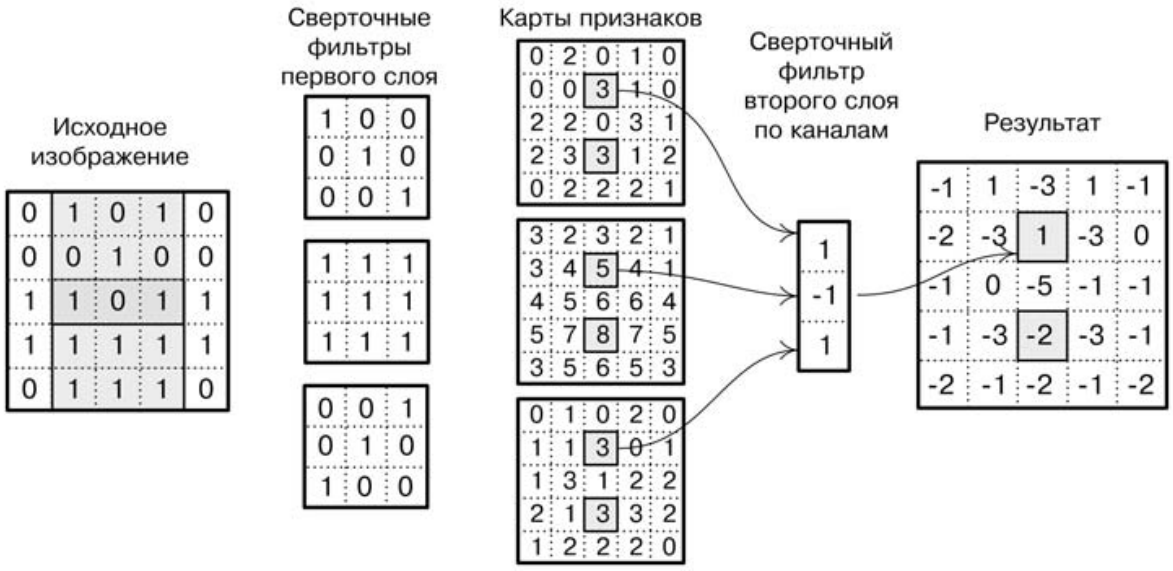
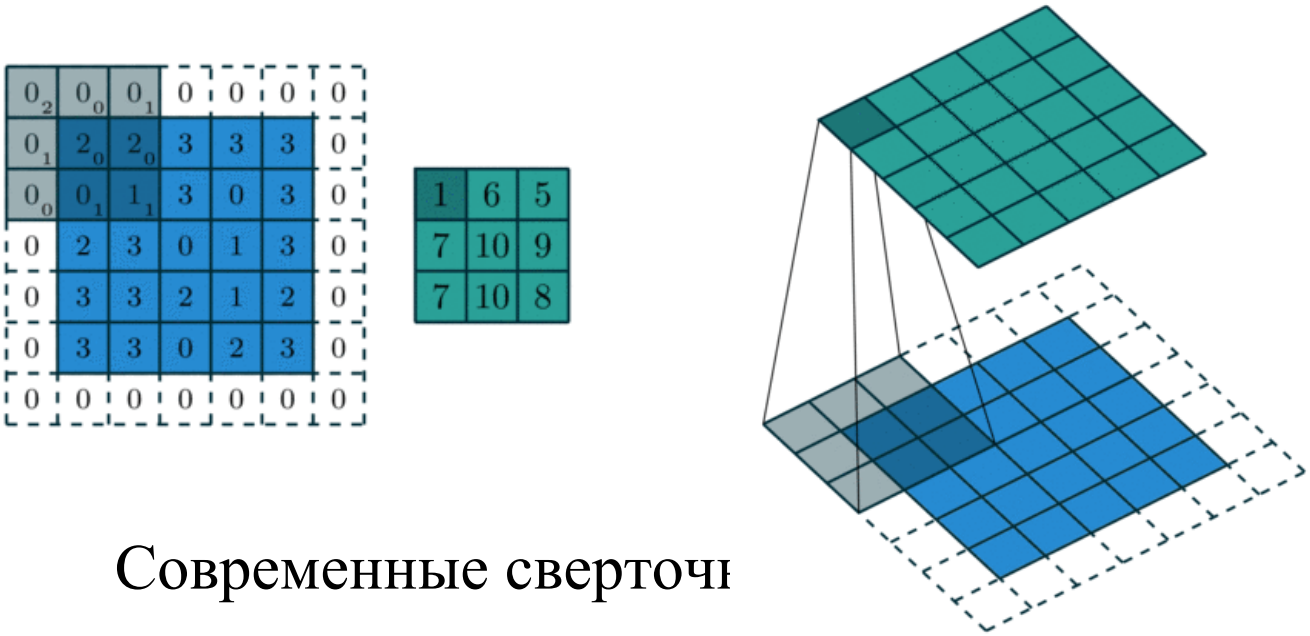


Рисунок 4. Пример выделения признаков на двух сверточных слоях



Современные сверточ

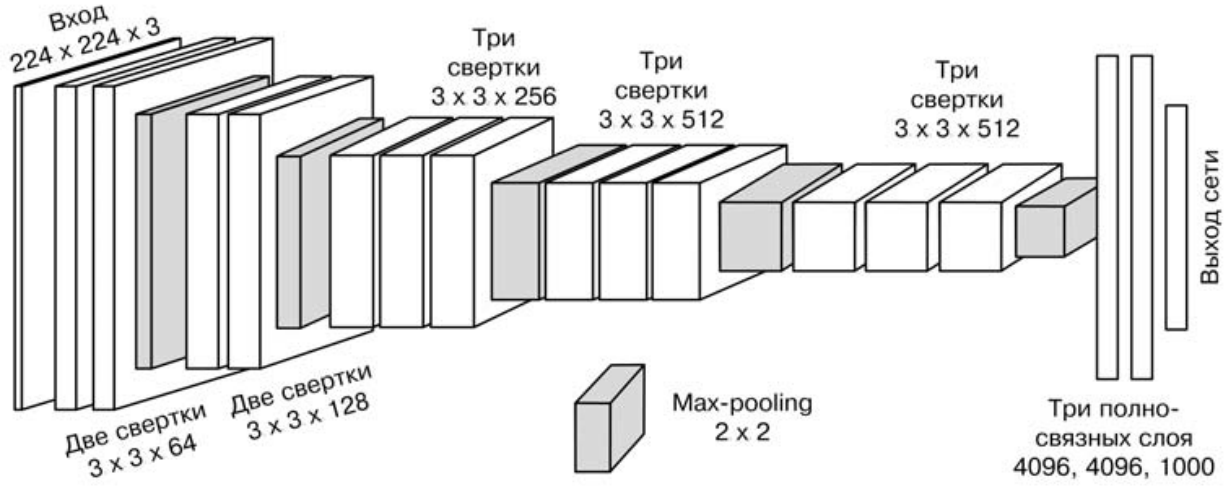


Рисунок 5. Схема сети VGG-16

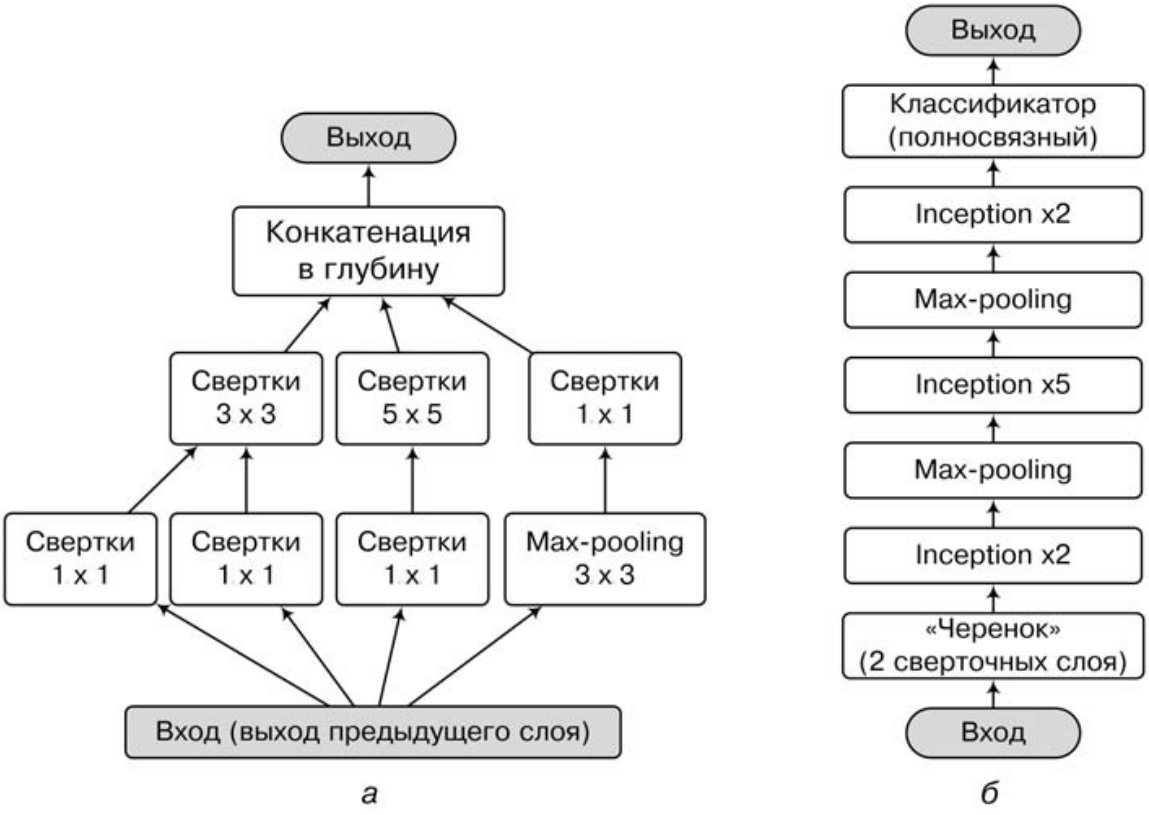


Рисунок 6. Inception: *а* — схема одного Inception-модуля; *б* — общая схема сети GoogLeNet

CNN для распознавания звуков и текстов

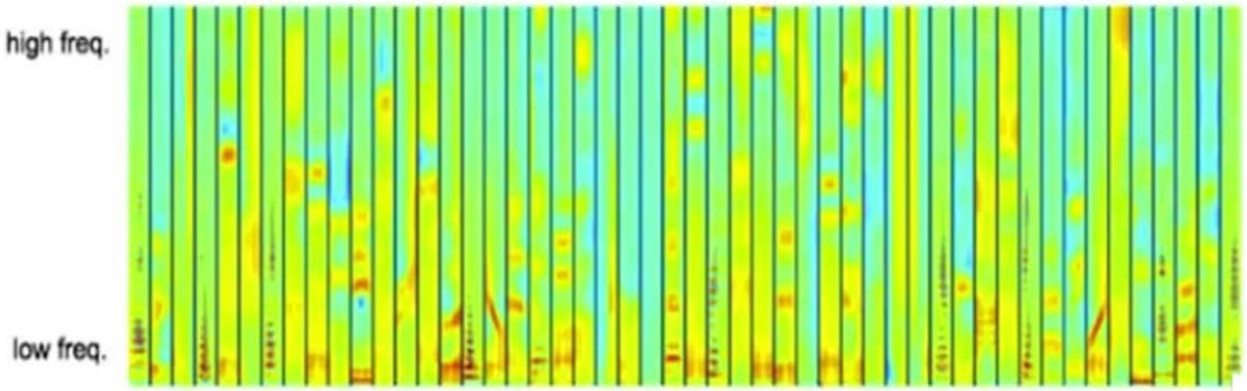


Рис.: Спектрограмма голосового сигнала

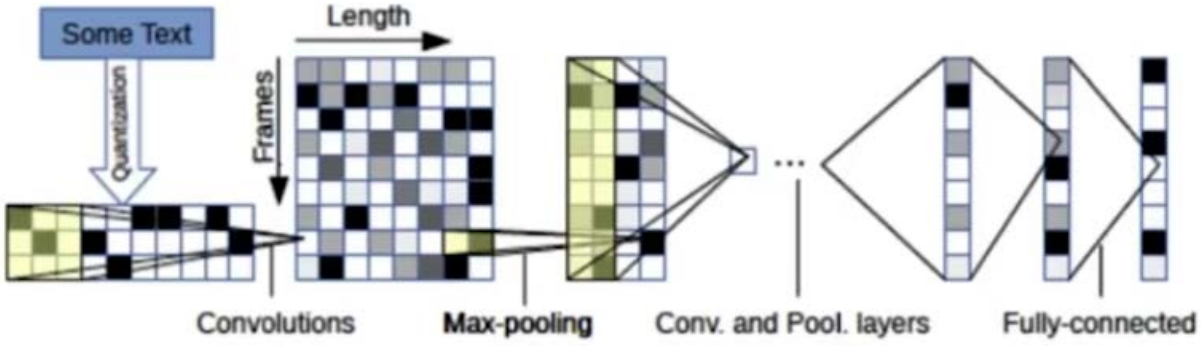


Рис.: Обработка изображения представляющего текст