**What is CNN?**

Convolutional neural networks (CNNs) are a type of artificial neural network that use convolution to recognize and classify images and objects.

**CONVOLUTIONAL VARIANTS:**

1. TARANSPOSED CONVOLUTIONAL
2. DILATED CONVOLUTIONAL
3. STRIDED CONVOLUTIONAL
4. **Transposed Convolutional Layer**

* Transposed convolutional layers are used to increase the size of the feature map, making them larger than the input.
* They are often called "deconvolutional layers" but don't truly reverse a convolution layer. Instead, they upsample while preserving certain features.
* Unlike true deconvolution, the output from a transposed convolution doesn’t perfectly recreate the original input—it approximates it while matching the desired size.
* Transposed convolutions are used in tasks like:

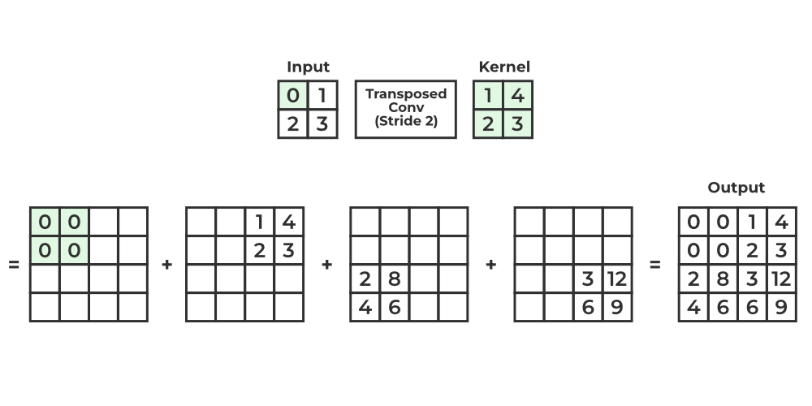
**1)Image generation** (e.g., creating images from noise in GANs).

**2)Super-resolution** (e.g., making low-resolution images high-resolution).

**3)Image segmentation** (e.g., identifying objects in images pixel by pixel).

* The size of the output is determined by the stride and padding parameters, allowing the layer to upsample the input to the desired dimensions.

**Example:**



 **Multiply the Input by the Kernel**:

* For each input value, multiply it by the kernel (filter).

 **Place Results with Stride**:

* Place the resulting values in the output grid, shifted based on the **stride (2)**.

 **Add Overlapping Values**:

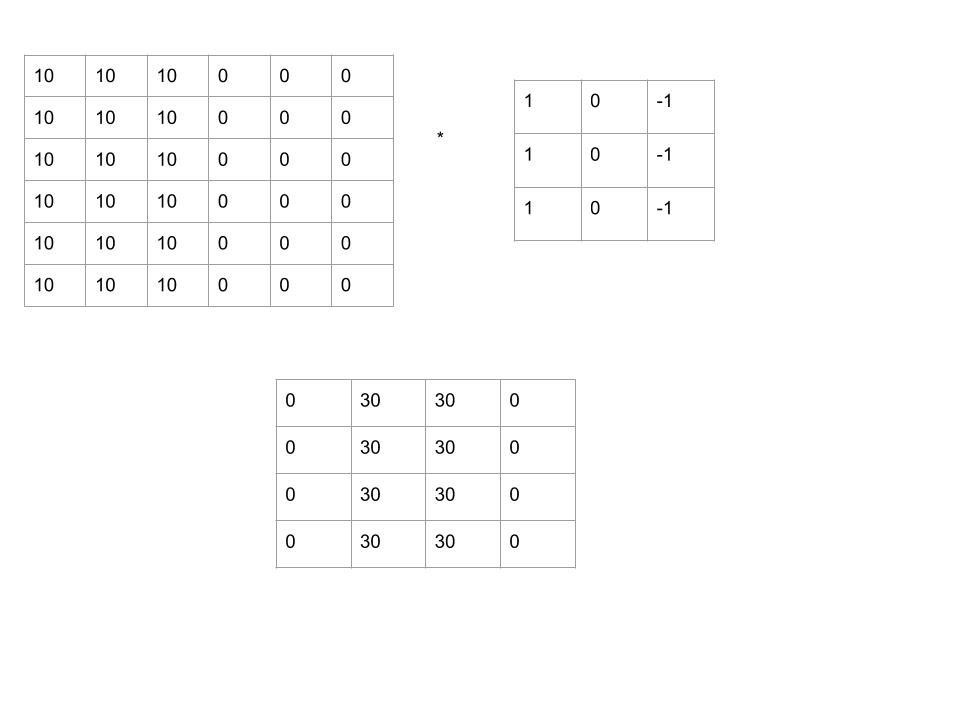
* Sum any overlapping values in the output grid to get the final output.

# Strided Convolutions

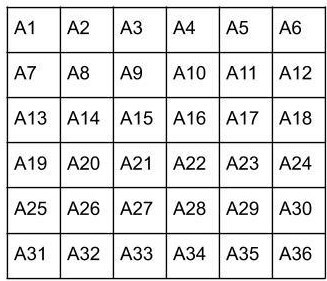
When performing a convolution operation, the stride determines how many units the filter shifts at each step.

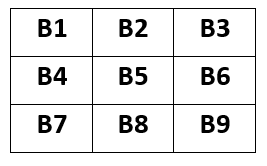
**Example:** Consider a 6 x 6 image as shown in figure below. It is to be convoluted with a 3 x 3 filter.

**Figure 1: Image obtained after convolution of 6×6 image with a 3×3 filter and s=0**



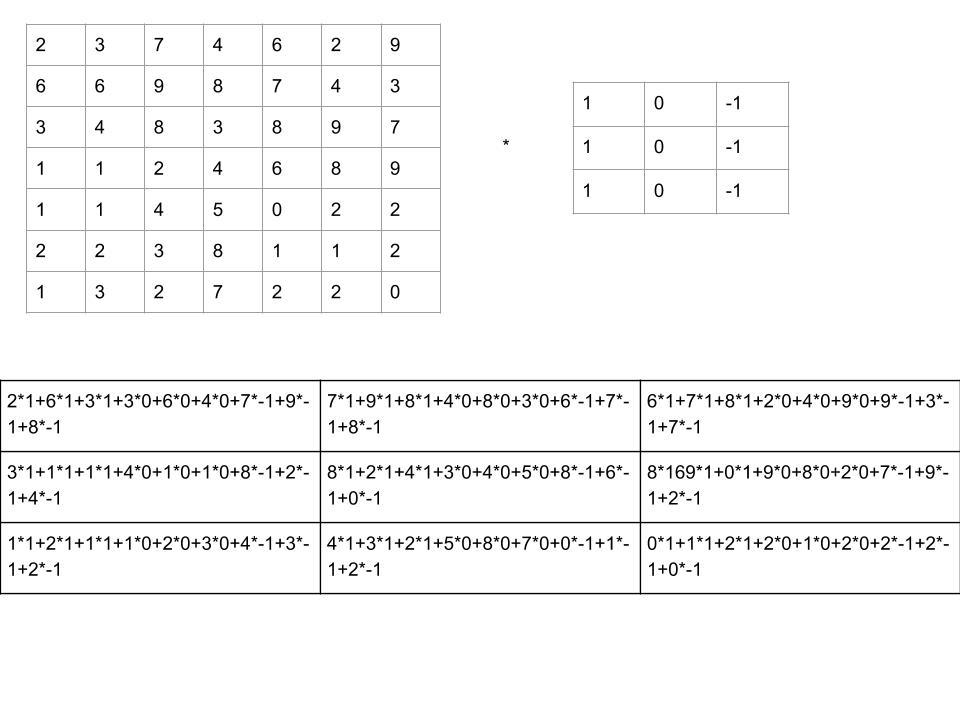
**Figure 2: 6 x 6 filter**



**Figure 3: 3 x 3 filter**

**Figure 4: Element wise multiplication**

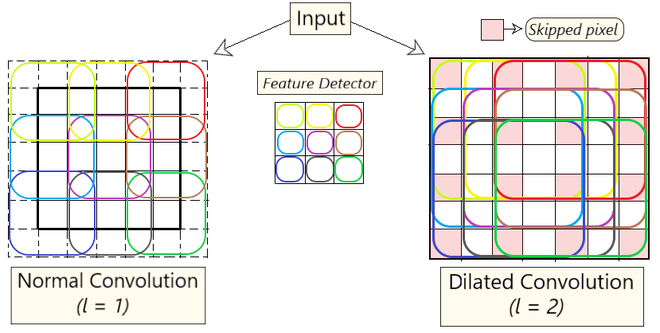
**Figure 5: Image obtained after convolution of 6×6 image with a 3×3 filter and a stride of 2**



# Dilated Convolution

* It's like regular convolution, but with **gaps (holes)** inserted between the filter values to cover a larger area of the input without increasing the filter size.
* The **dilation rate** controls the size of the gaps between filter values.When the dilation rate is 1, it behaves like a regular convolution (no gaps).
* It **increases the receptive field**, allowing the network to "see" a larger portion of the input, which helps in capturing more context without adding extra parameters.

**Example:**



**Intuition:**

* It’s a convolution technique that skips pixels (based on the **dilation factor lll**) to cover a larger area of the input without pooling or increasing filter size.
* The **dilation factor lll** determines the gap between filter elements. For example:
  1. l=2l = 2l=2: Skip 1 pixel between filter elements.
  2. This wider spacing lets the filter capture more context in each step.
* It expands the **field of view** without increasing the number of filter parameters or computational cost, making it ideal for tasks like image segmentation or object recognition.

**Advantages of Dilated Convolution:**

* Larger receptive field (i.e. no loss of coverage)
* Computationally efficient (as it provides a larger coverage on the same computation cost)
* Lesser Memory consumption (as it skips the pooling step) implementation
* No loss of resolution of the output image (as we dilate instead of performing pooling)
* Structure of this convolution helps in maintaining the order of the data.