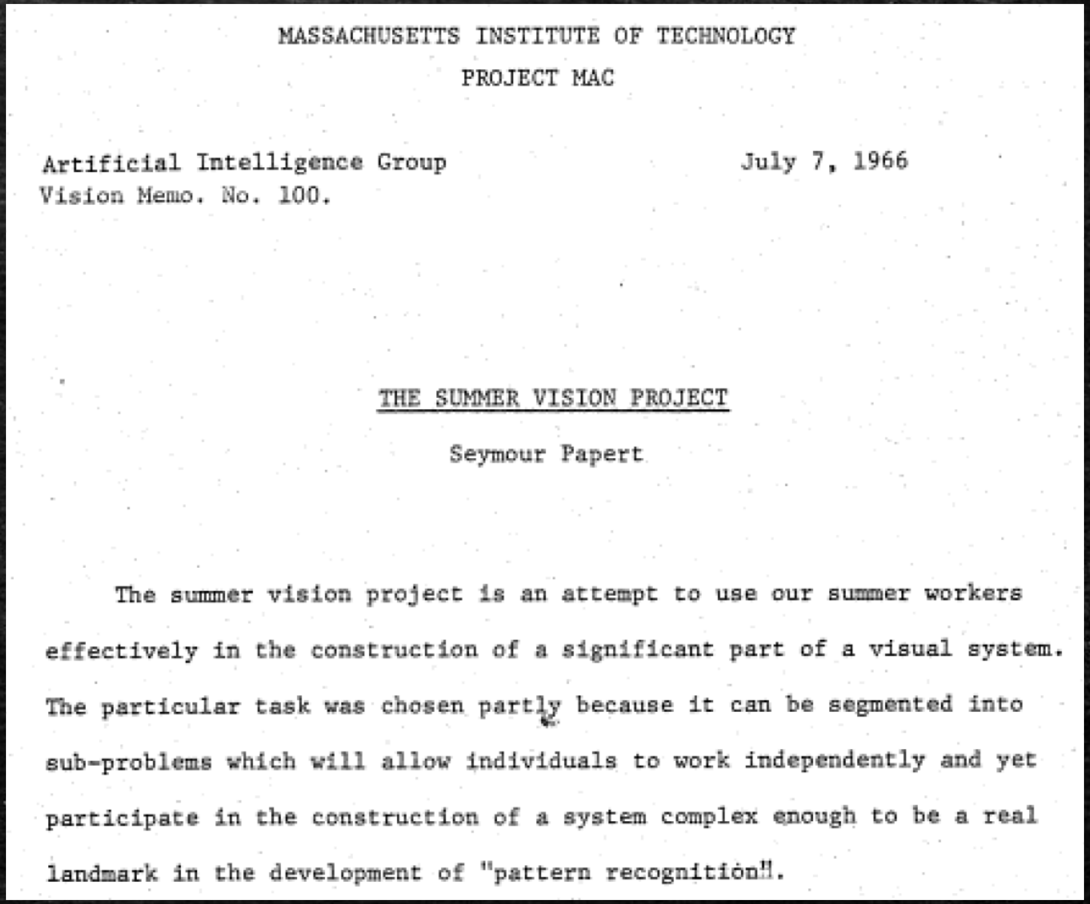
Convolutional Neural Network

Joe Yeh, M.D.

A Brief History on Computer Vision

MIT Summer Vision Project

…in the construction of a system complex enough to be a real landmark in the development of “pattern recognition”.



MASSACHUSETTS INSTITUTE OF TECHNOLOGY

PROJECT MAC

Artificial Intelligence Group July 7, 1966

Vision Memo. No. 100.

THE SUMMER VISION PROJECT

Seymour Papert

The summer vision project is an attempt to use our summer workers effectively in the construction of a significant part of a visual system. The particular task was chosen partly because it can be segmented into sub-problems which will allow individuals to work independently and yet participate in the construction of a system complex enough to be a real landmark in the development of "pattern recognition".

General Goals of MIT Summer Vision Project

Goals - General

The primary goal of the project is to construct a system of programs which will divide a vidisector picture into regions such as

likely objects

likely background areas

chaos.

We shall call this part of its operation FIGURE-GROUND analysis.

It will be impossible to do this without considerable analysis of shape and surface properties, so FIGURE-GROUND analysis is really insepar-able in practice from the second goal which is REGION DESCRIPTION.

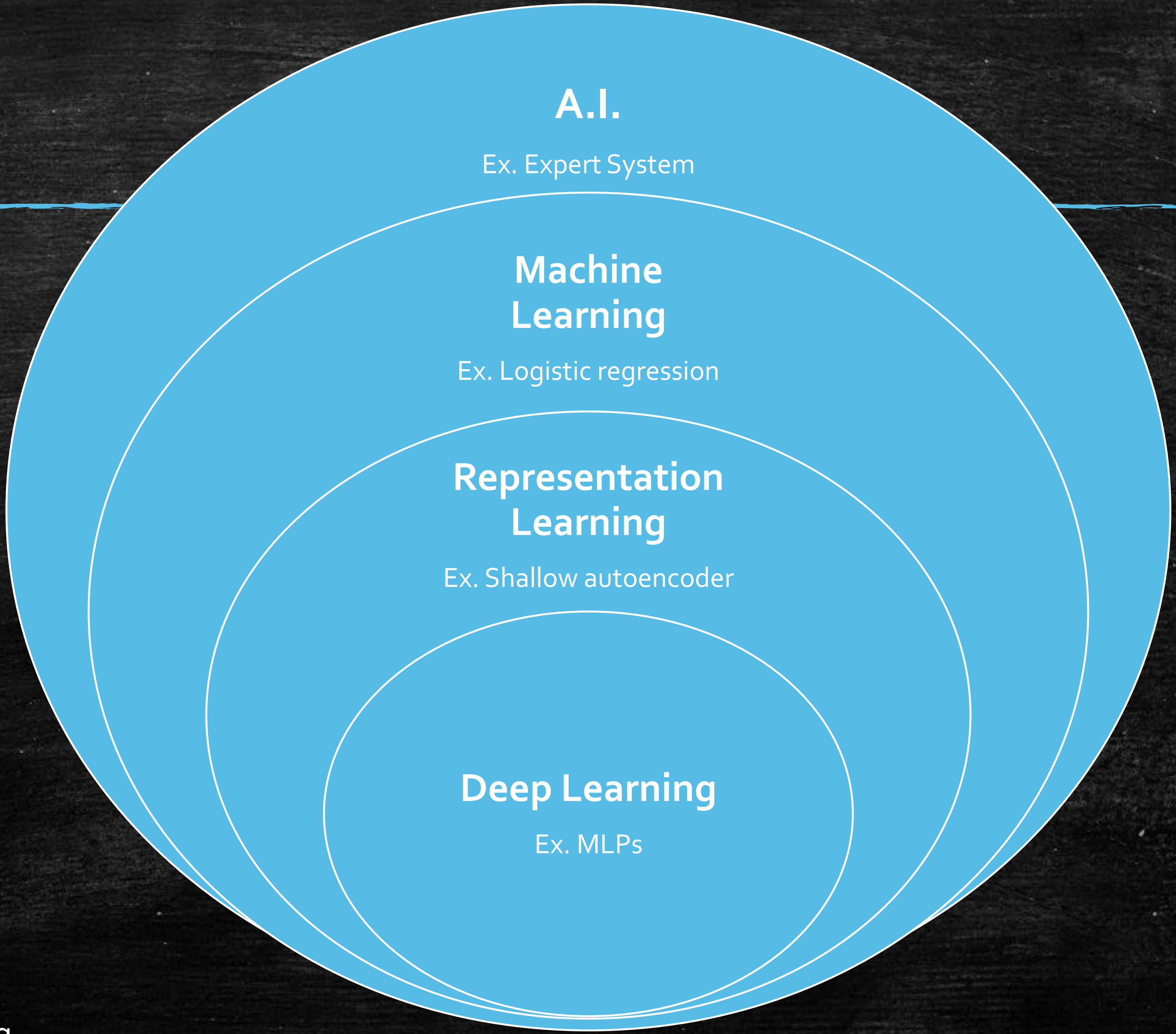
The final goal is OBJECT IDENTIFICATION which will actually name objects by matching them with a vocabulary of known objects.

Artificial Intelligence : The beginning

- Dartmouth Summer Research Project on Artificial Intelligence (1959)
  - Proposed by John McCarthy, Marvin Minsky, Nathaniel Rochester and Claude Shannon
  - to proceed on the basis of the conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it.

AI@50 From left to right: Trenchard More, John McCarthy, Marvin Minsky, Oliver Selfriege, Ray Solomonoff

A.I and Deep Learning

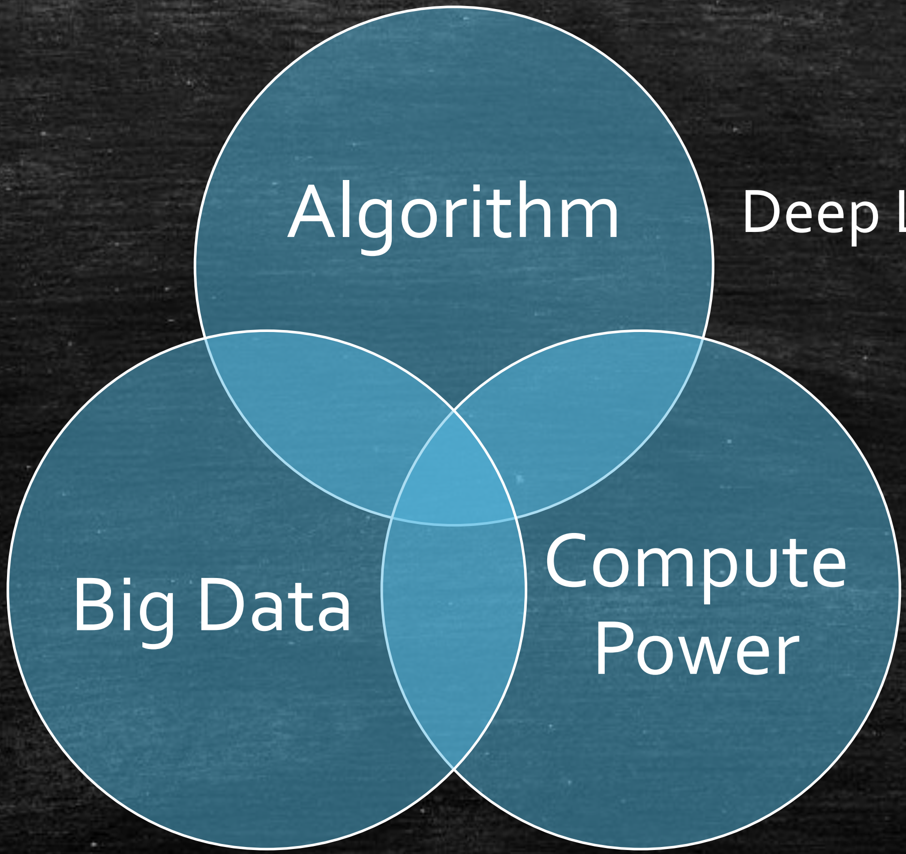


Ian Goodfellow et al. Deep Learning

Three A.I. winters

- Machine translation : 1950-1960s
  - Georgetown Experiment showing Russian to English translation 1954
  - Automated Language Processing Advisory Committee says progress is slow 1966
- Making AI in a controlled environment: 1970s
  - Teaching AI to perform task in micro world
  - Chatbot for talk therapy
  - 1974 UK Lighthill report : …utter failure of AI to achieve its grandiose objectives
- Expert systems : 1980s
  - Symbolic Lisp machines, IBM's Integrated Reasoning Shell
  - Collapse of Symbolics

What's different this time ?



Deep Learning !

Internet, Digitization

Compute accelerators GPU, TPU, NPU

What is Convolutional Neural Network ?

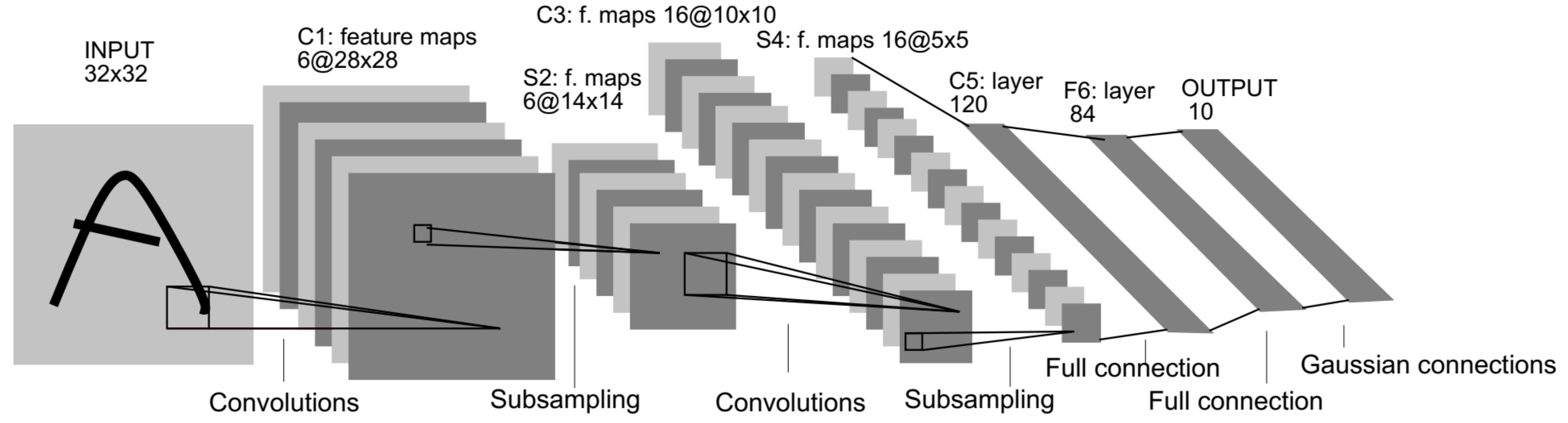


Fig. 2. Architecture of LeNet-5, a Convolutional Neural Network, here for digits recognition. Each plane is a feature map, i.e. a set of units whose weights are constrained to be identical.

Landmark CNN Architectures

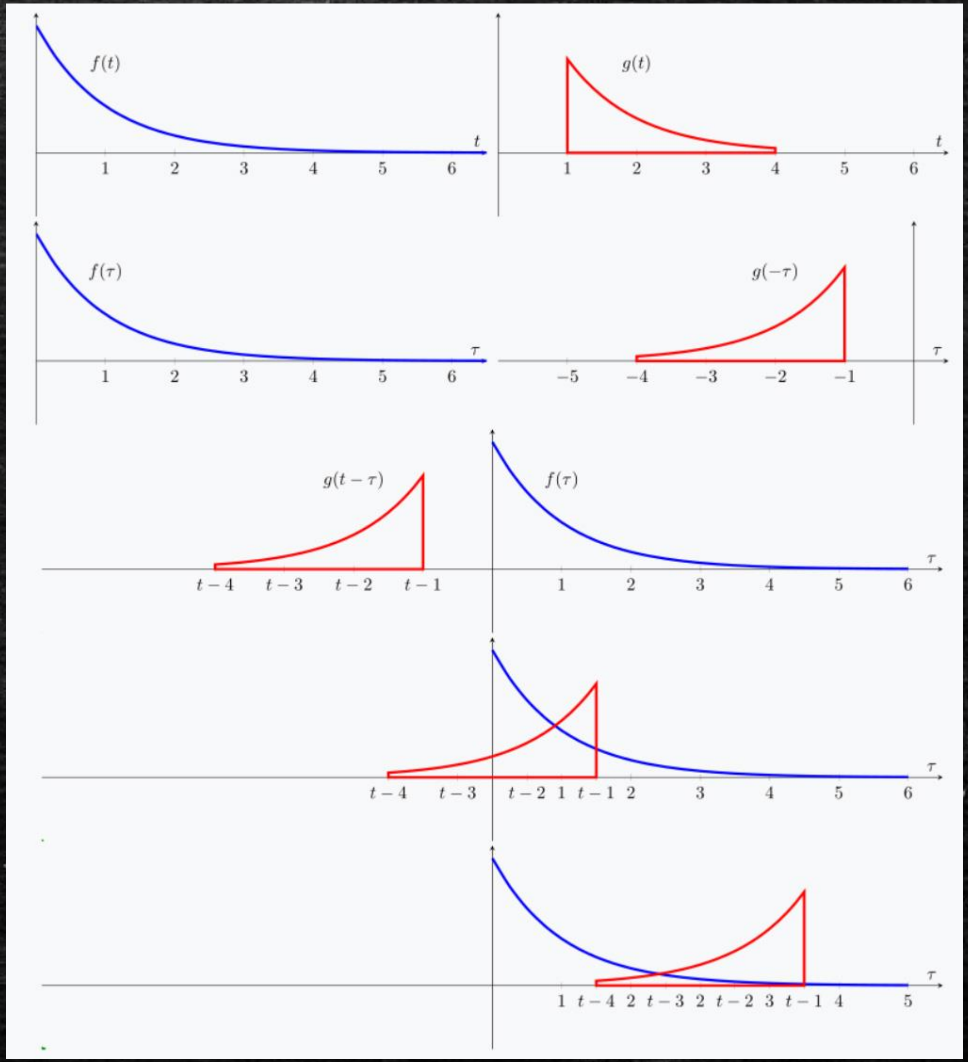
- LeNet (1998, U. Montreal, LeCun, Bengio)
- AlexNet (2012, U. Toronto, Krizhevsky, Hinton)
- VGG (2014, Oxford)
- Inception (2014, Google)
- ResNet (2015, Microsoft)
- DenseNet(2017, Facebook)

Computer Vision Tasks

- Image classification (categorical output)
  - Example : chest x-ray → diagnosis of pneumonia
- Image regression (continuous real number output)
  - Example : CT image→ bone age
- Object detection
  - Example : Lung CT image → 3-D bounding box enclosing tumor
- Image segmentation
  - Example : Brain MR image → contour of tumor

What is Convolution?

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau) g(t-\tau) d\tau$$



1D Convolution : step by step

```
Result                   0
Padded original signal   0 0 0 0 1 0 0 0 0
Kernel                   1 2 3
```

```
0 0 3 2 1 0
0 0 0 0 1 0 0 0 0
          1 2 3
```

```
0 0
0 0 0 0 1 0 0 0 0
  1 2 3
```

```
0 0 3 2 1 0 0
0 0 0 0 1 0 0 0 0
            1 2 3
```

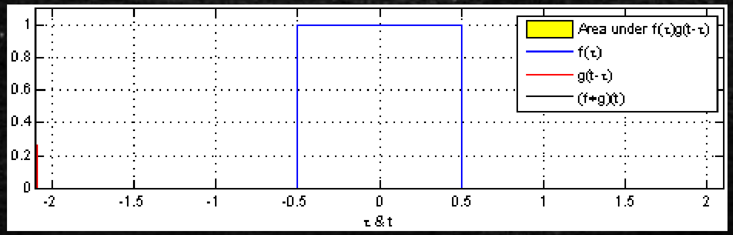
```
0 0 3
0 0 0 0 1 0 0 0 0
    1 2 3
```

'Same' padding result:
0 3 2 1 0

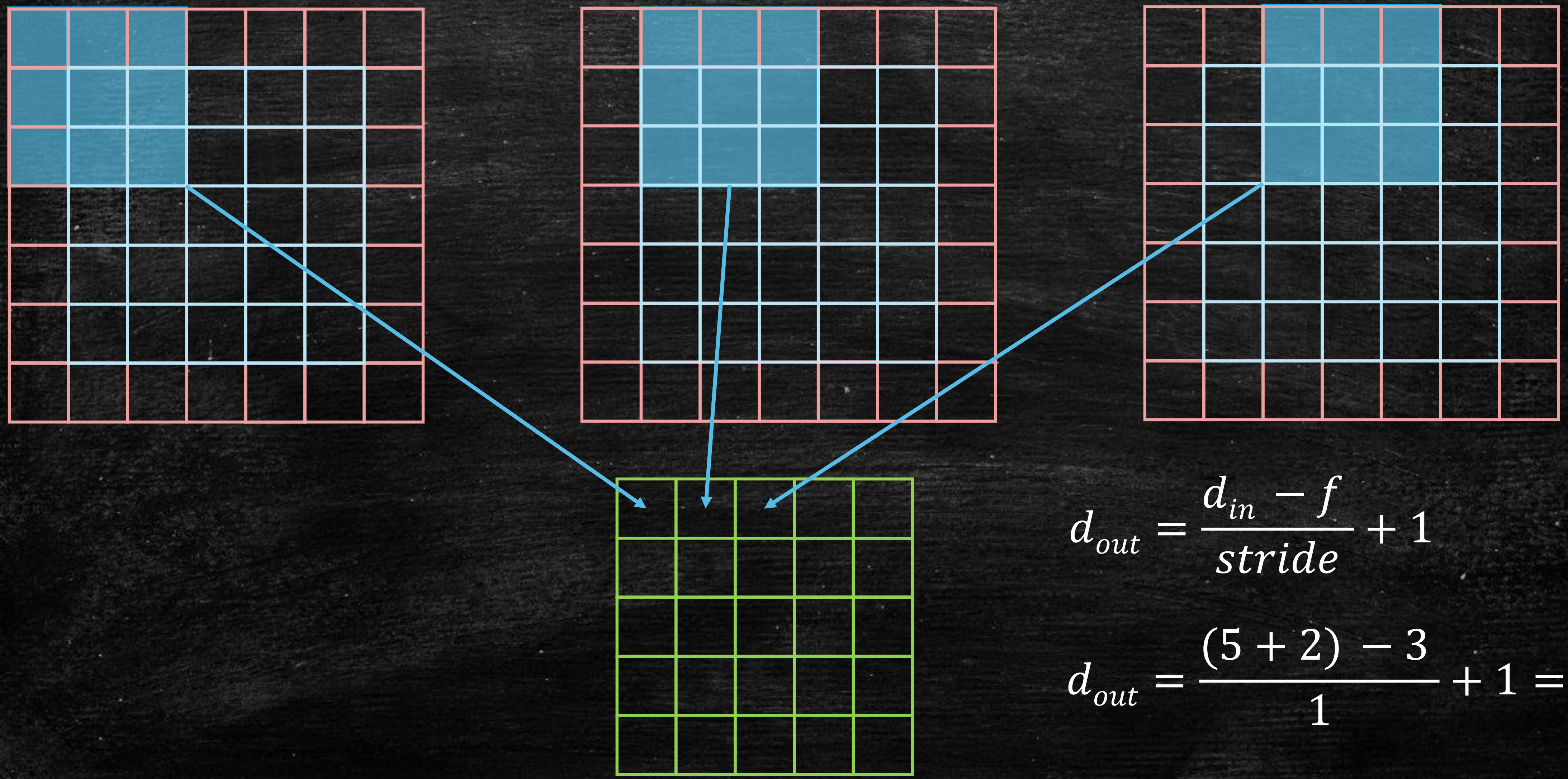
```
0 0 3 2
0 0 0 0 1 0 0 0 0
      1 2 3
```

'Valid' padding result:
3 2 1

```
0 0 3 2 1
0 0 0 0 1 0 0 0 0
        1 2 3
```



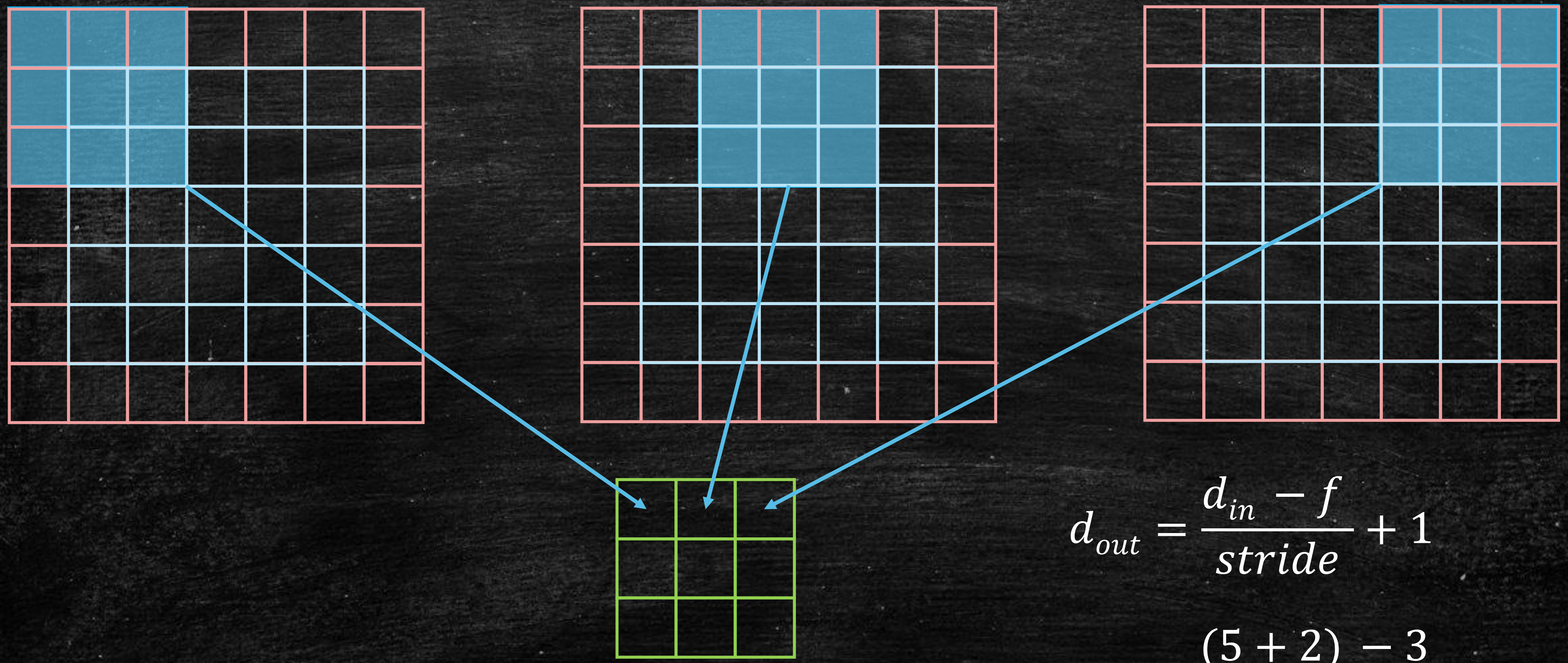
2D Convolution : Same padding, stride = 1



$$d_{out} = \frac{d_{in} - f}{stride} + 1$$

$$d_{out} = \frac{(5+2) - 3}{1} + 1 = 5$$

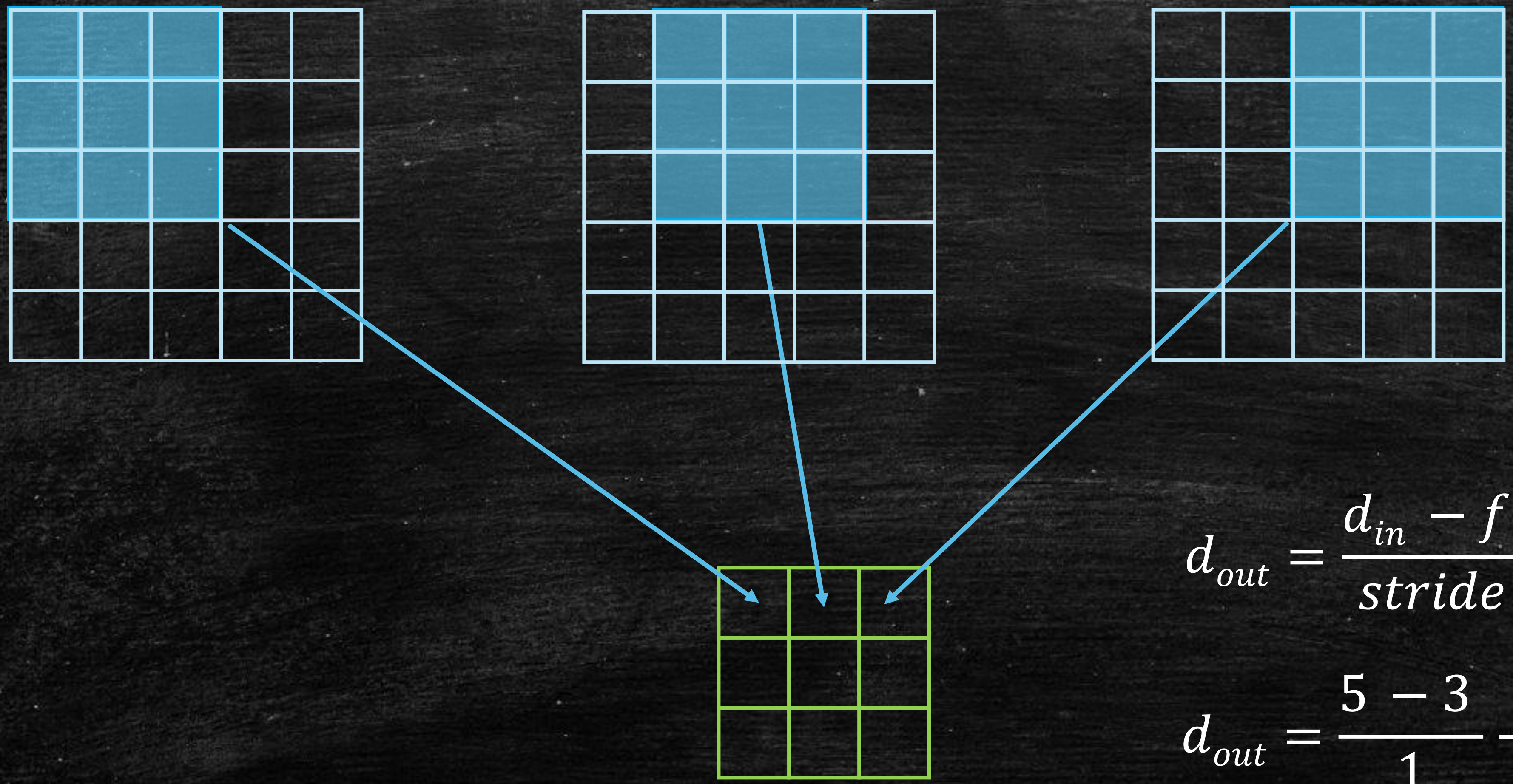
2D Convolution : Same padding, stride = 2



$$d_{out} = \frac{d_{in} - f}{stride} + 1$$

$$d_{out} = \frac{(5+2) - 3}{2} + 1 = 3$$

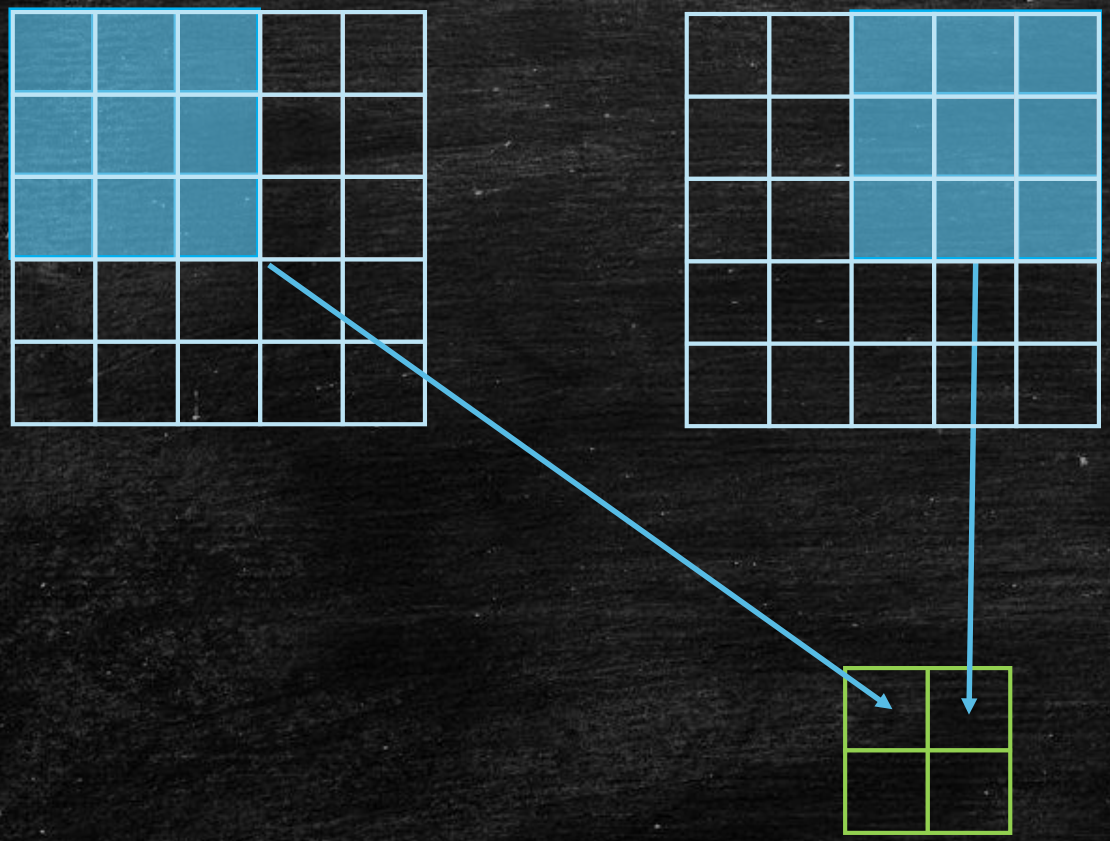
2D Convolution : Valid padding, stride =1



$$d_{out} = \frac{d_{in} - f}{stride} + 1$$

$$d_{out} = \frac{5 - 3}{1} + 1 = 3$$

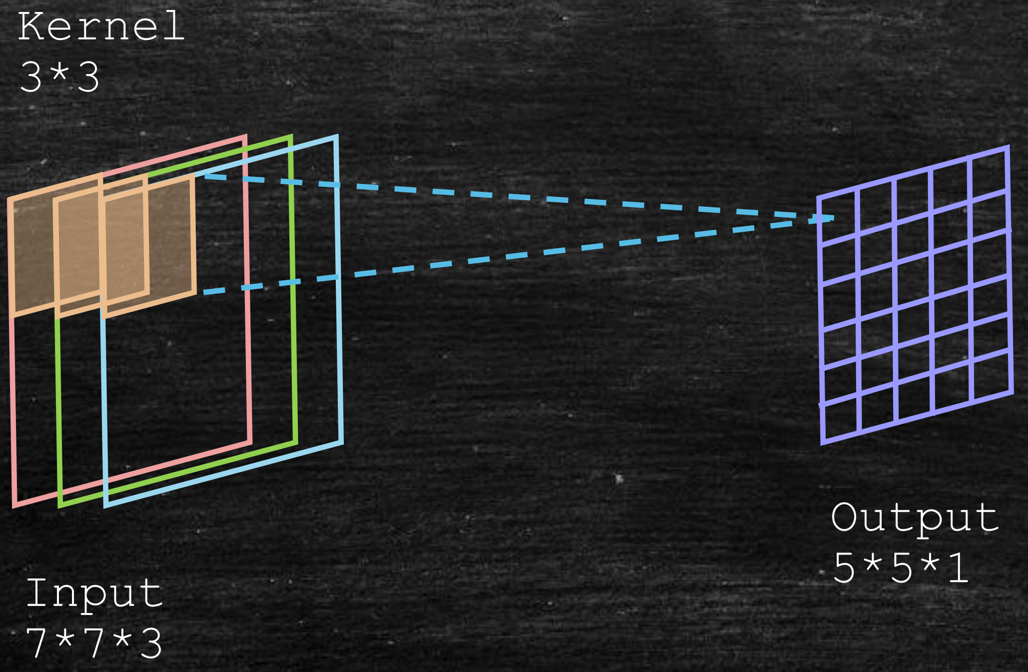
2D Convolution : Valid padding, stride = 2



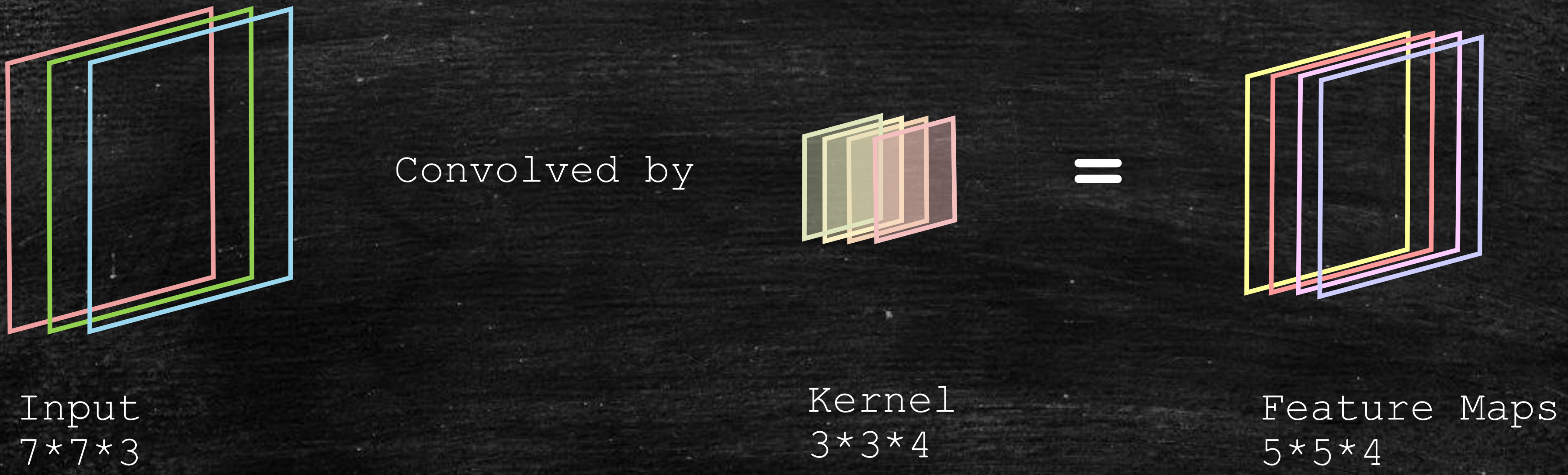
$$d_{out} = \frac{d_{in} - f}{stride} + 1$$

$$d_{out} = \frac{5 - 3}{2} + 1 = 2$$

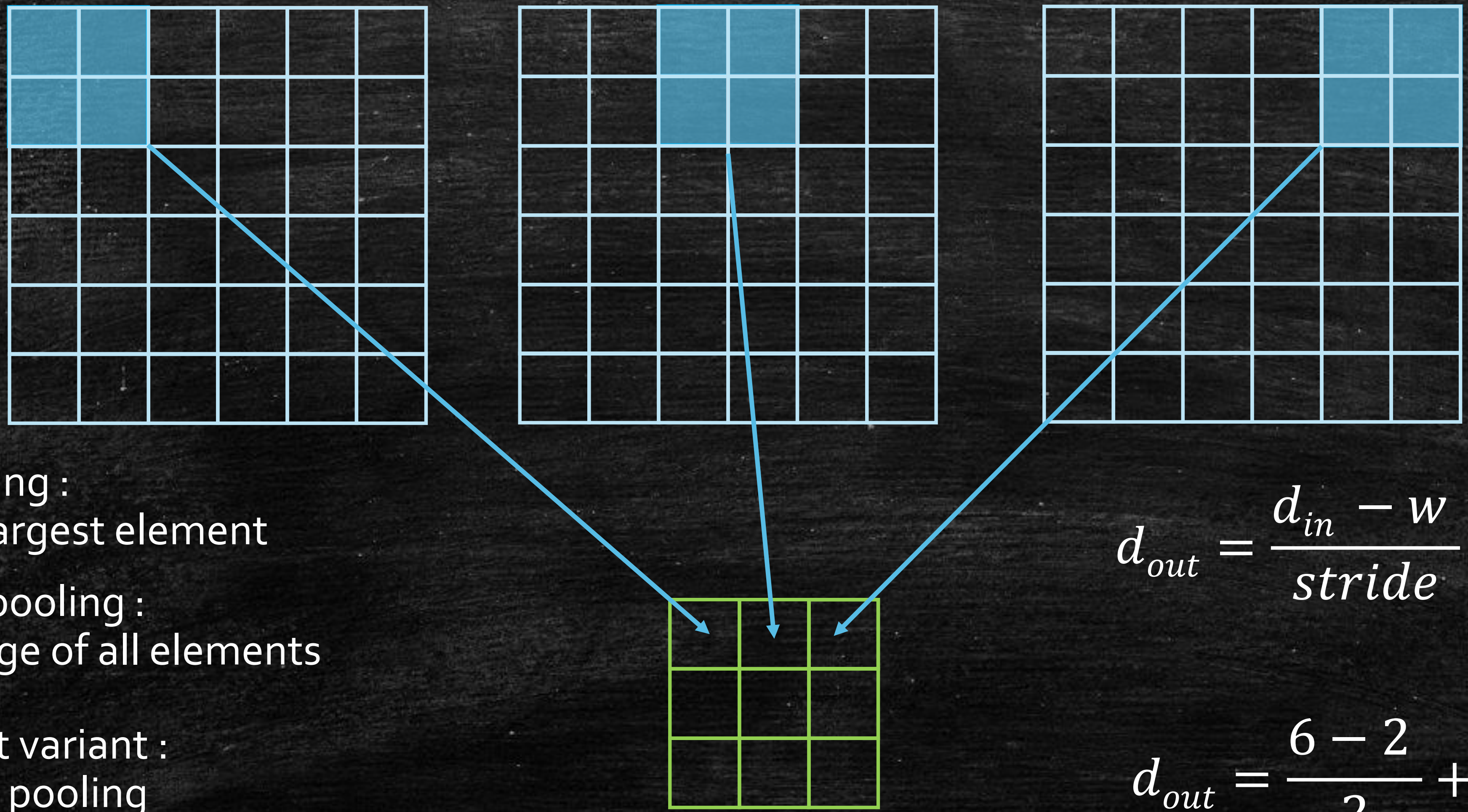
2D Convolution on a 3D Volume



2D Convolution on a 3D Volume



Pooling (subsampling)



Max pooling :
pick the largest element

Average pooling :
get average of all elements

Important variant :
fractional pooling

$$d_{out} = \frac{d_{in} - w}{stride} + 1$$

$$d_{out} = \frac{6 - 2}{2} + 1 = 3$$

Anatomy of a typical Convolutional Neural Network : Using LeNet5 as an example

Gradient-Based Learning Applied to Document Recognition

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner

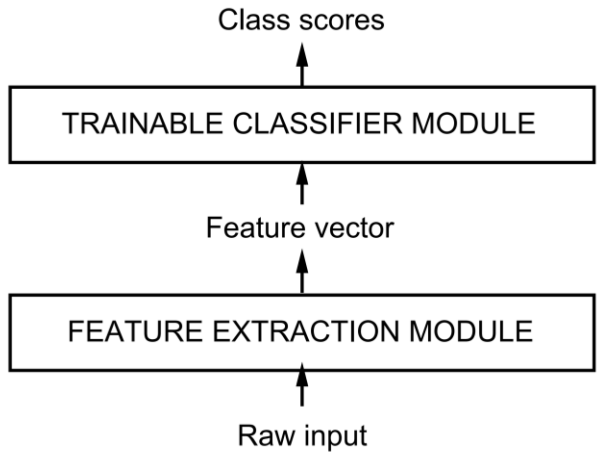


Fig. 1. Traditional pattern recognition is performed with two modules: a fixed feature extractor, and a trainable classifier.

http://yann.lecun.com/exdb/publis/pdf/lecun-98.pdf

Gradient-based learning method

- The usefulness of gradient-based learning method was realized after three events:
  - Success of Bolzmann machine despite concerns about local minima
  - The work of Rumelhart, Hinton and Williams : *Learning representation by back-propagating errors, Nature 323, 533-536, 1986*
  - Demonstration that back-propagation applied to multi-layer neural network with sigmoid units can solve complex learning tasks
- Basic idea of back-propagation
  - Gradients can be computed efficiently by propagation from the output to the input

Advantage of CNN in Image Recognition

CNN performs far better than fully connected neural net because of :

- Weight sharing (using the same kernel for convolution)
  - Reduced number of weight to train and store
  - Rotation, scale and shift invariance
- Preservation of information topology
  - Preservation of spatial relationship between input
  - Hierarchical representation of image object

Anatomy of LeNet-5

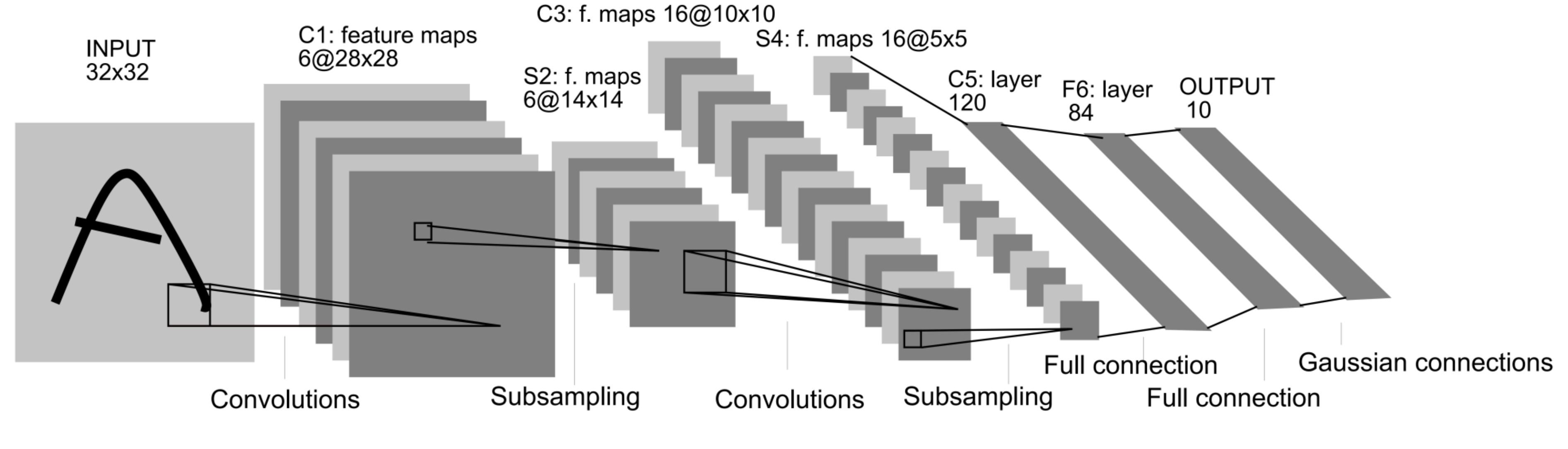


Fig. 2. Architecture of LeNet-5, a Convolutional Neural Network, here for digits recognition. Each plane is a feature map, i.e. a set of units whose weights are constrained to be identical.

$$d_{out} = \frac{32-5}{1} + 1 = 28 \qquad d_{out} = \frac{14-5}{1} + 1 = 10$$

$$d_{out} = \frac{28-2}{2} + 1 = 14 \qquad d_{out} = \frac{10-2}{2} + 1 = 5$$

Squashing function : hyperbolic tangent

Output unit : Euclidean Radial Basis Function (RBF) unit

Anatomy of LeNet-5

- Convolutional layers uses hyperbolic tangent as squashing (activation) function
- Subsampling (pooling) operation is the dot product of trainable weights and input : this essentially means per channel convolution
- Output unit is Euclidean Radial Basis Function (RBF) unit.
- Loss function : Mean Squared Error

Output coding for LeNet-5

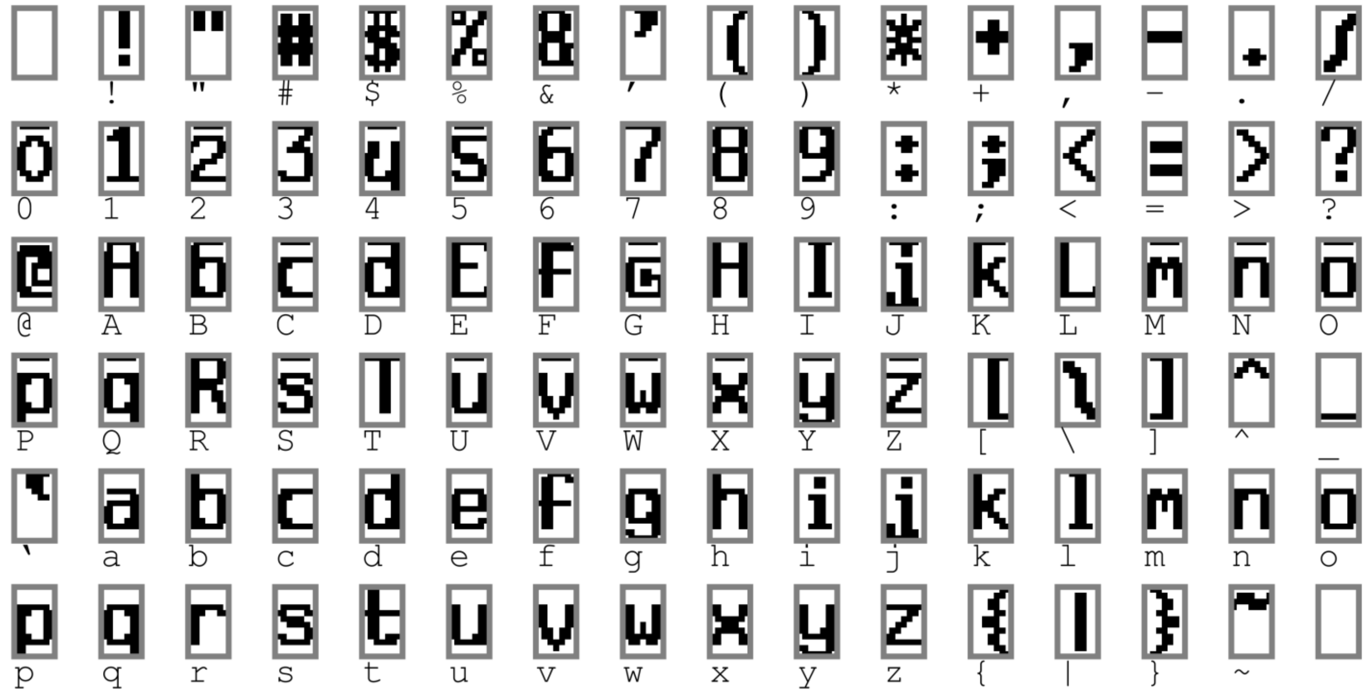


Fig. 3. Initial parameters of the output RBFs for recognizing the full ASCII set.

Rationale for choosing distributed coding for LeNet-5

- Similar characters have similar coding pattern
- Non-distributed codes (for example, one-hot code) behave badly when output class is more than a few dozens
- For non-distributed codes, all but one output unit must remain off all the time, which is difficult to achieve with sigmoid