



دانشگاه علم و صنعت
دانشکده مهندسی کامپیوتر

شبکه عصبی پیچشی

Convolutional Neural Network (CNN)

مدرس: آرش عبدی هجراندوست

arash_abdi@aut.ac.ir

نیم سال اول ۱۴۰۱-۱۴۰۰

شبکه عصبی پیچشی

❖ کاربردها:

❖ پردازش سیگنال

❖ پردازش تصویر

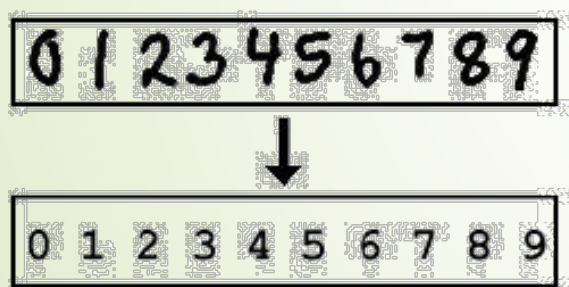
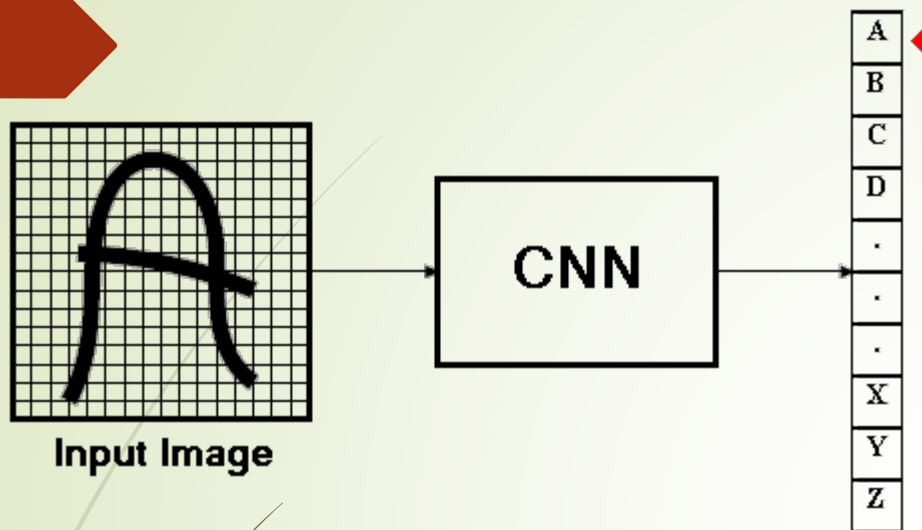
❖ بهبودی بر شبکه عصبی پرسپترون چندلایه

❖ افزایش کارایی

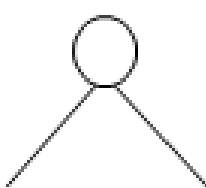
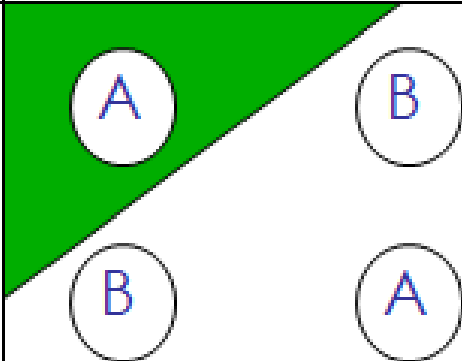
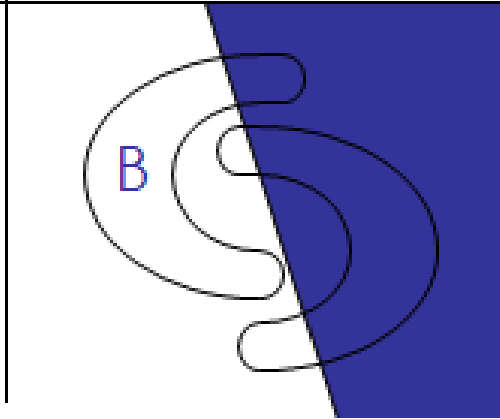
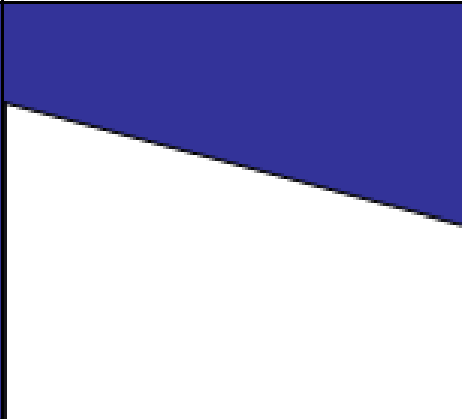
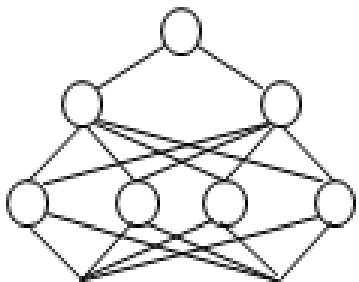
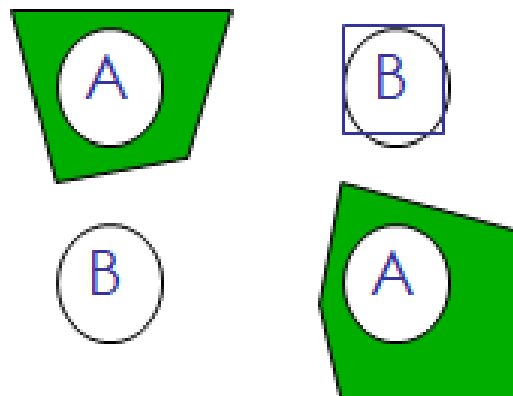
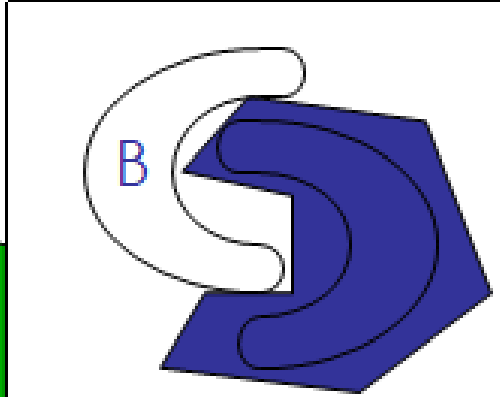
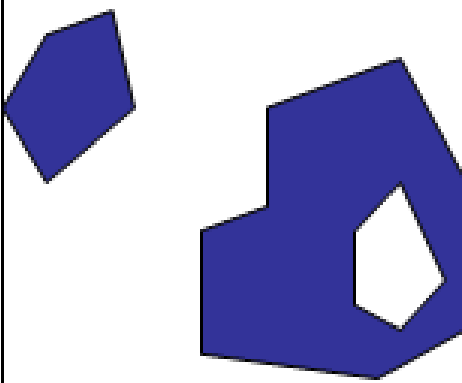
❖ سرعت اجرا

❖ دقت

❖ تا حدودی مقاومت در برابر به هم ریختگی در تصویر ورودی



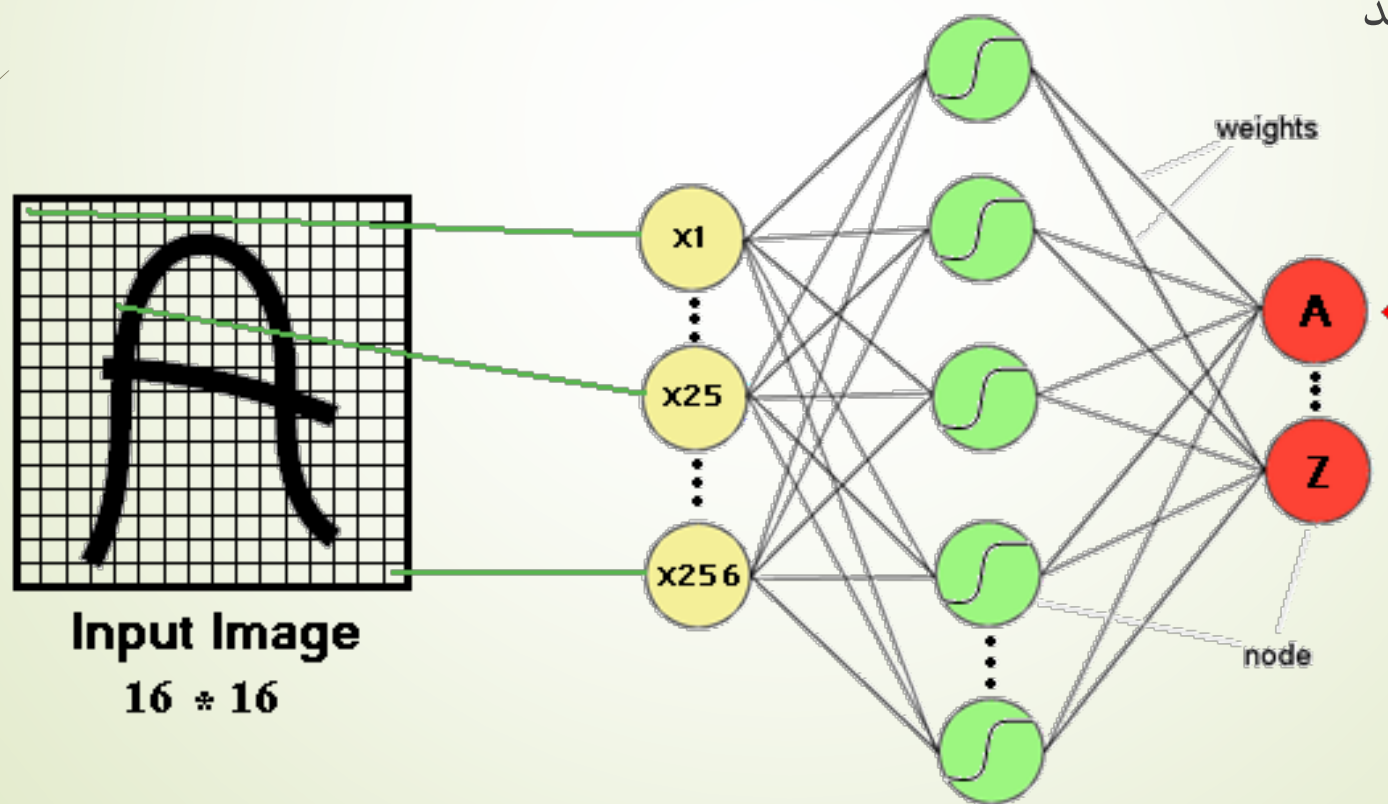
توان بازنمایی شبکه عصبی

Structure	Types of Decision Regions	Exclusive-OR Problem	Classes with Meshed regions	Most General Region Shapes
Single-Layer 	Half Plane Bounded By Hyper plane			
Multi-Layer 	Arbitrary (Complexity Limited by No. of Nodes)			

کاربرد شبکه عصبی چندلایه در پردازش تصویر

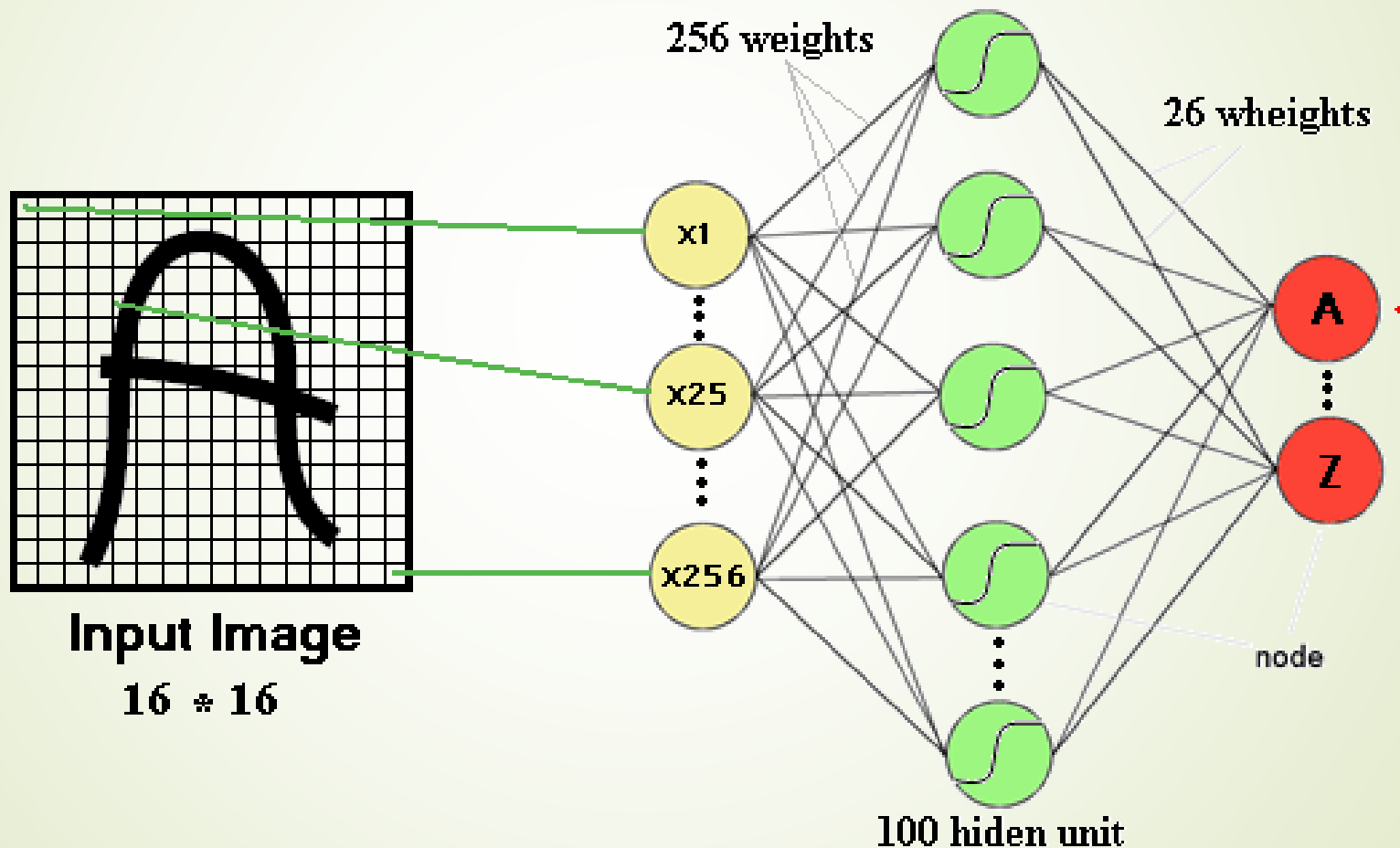
❖ یک یا چند لایه پنهان

❖ تابع فعالسازی سیگموئید



مشکلات شبکه عصبی: مشکل تعداد پارامترها

❖ تعداد پارامترهای قابل یادگیری خیلی زیاد است.

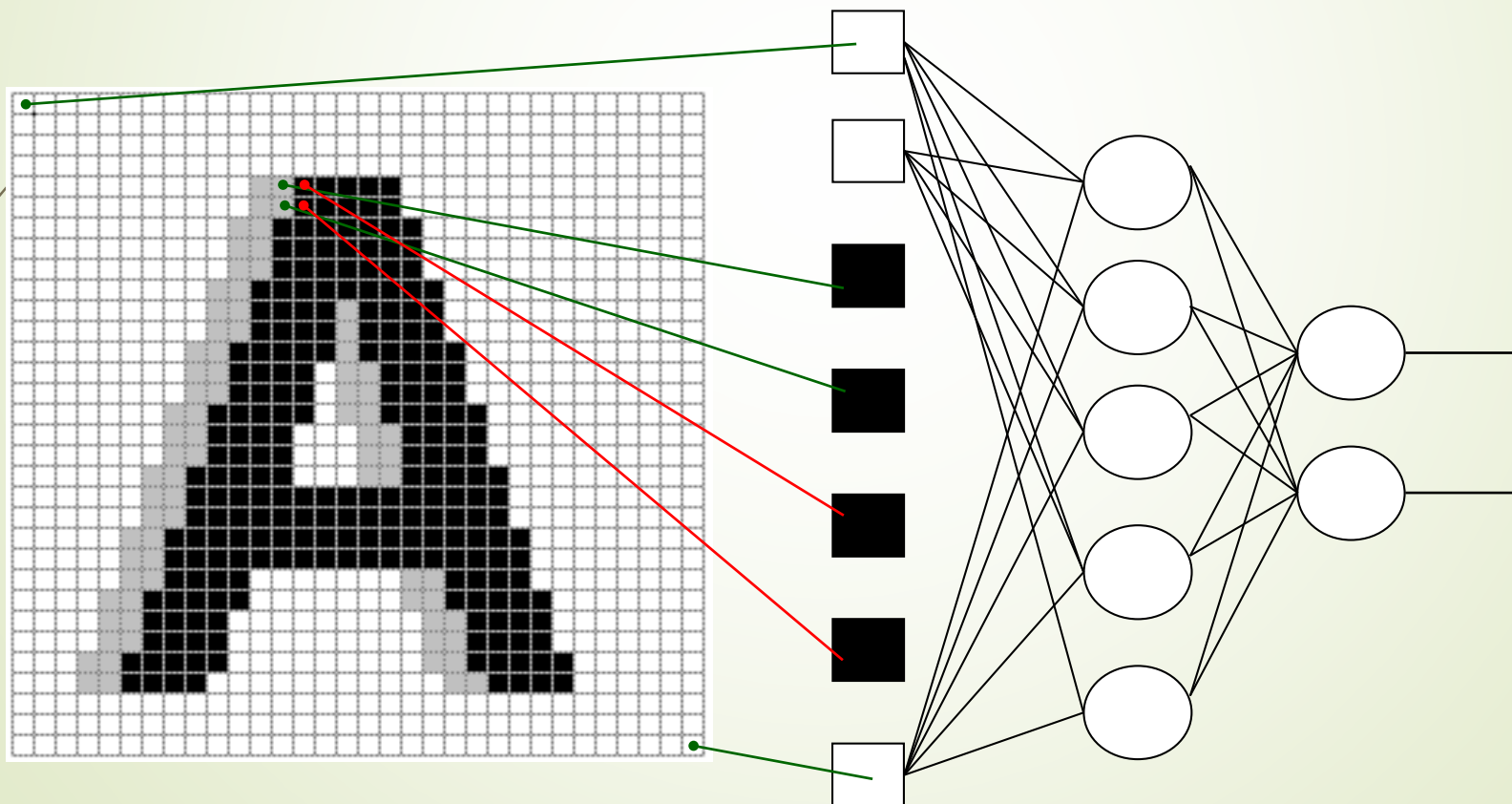


مشکلات شبکه عصبی: حساسیت به تغییرات ورودی

❖ جابجایی

❖ تغییر اندازه

❖ تغییر شکل

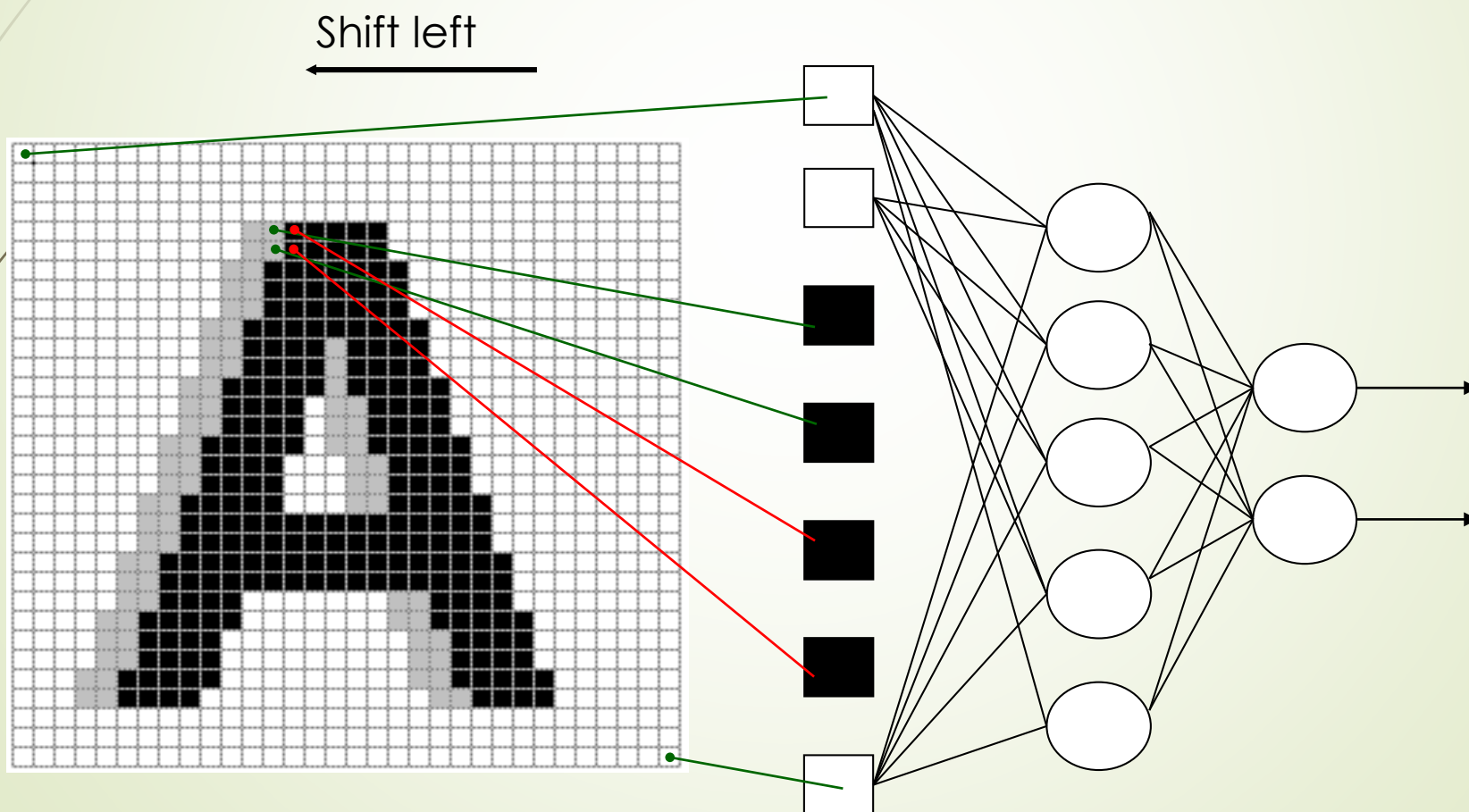


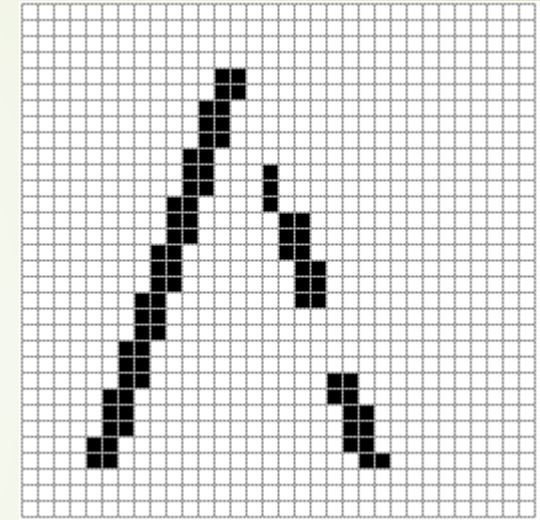
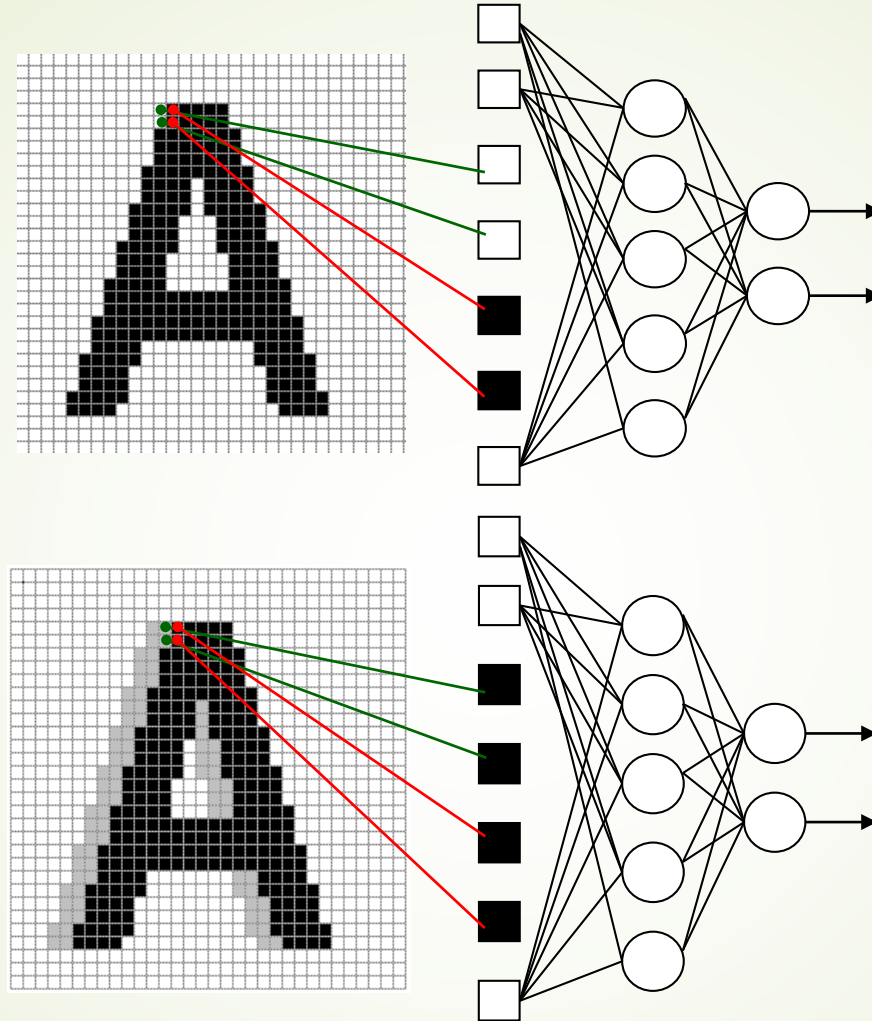
مشکلات شبکه عصبی: حساسیت به تغییرات ورودی

❖ جابجایی

❖ تغییر اندازه

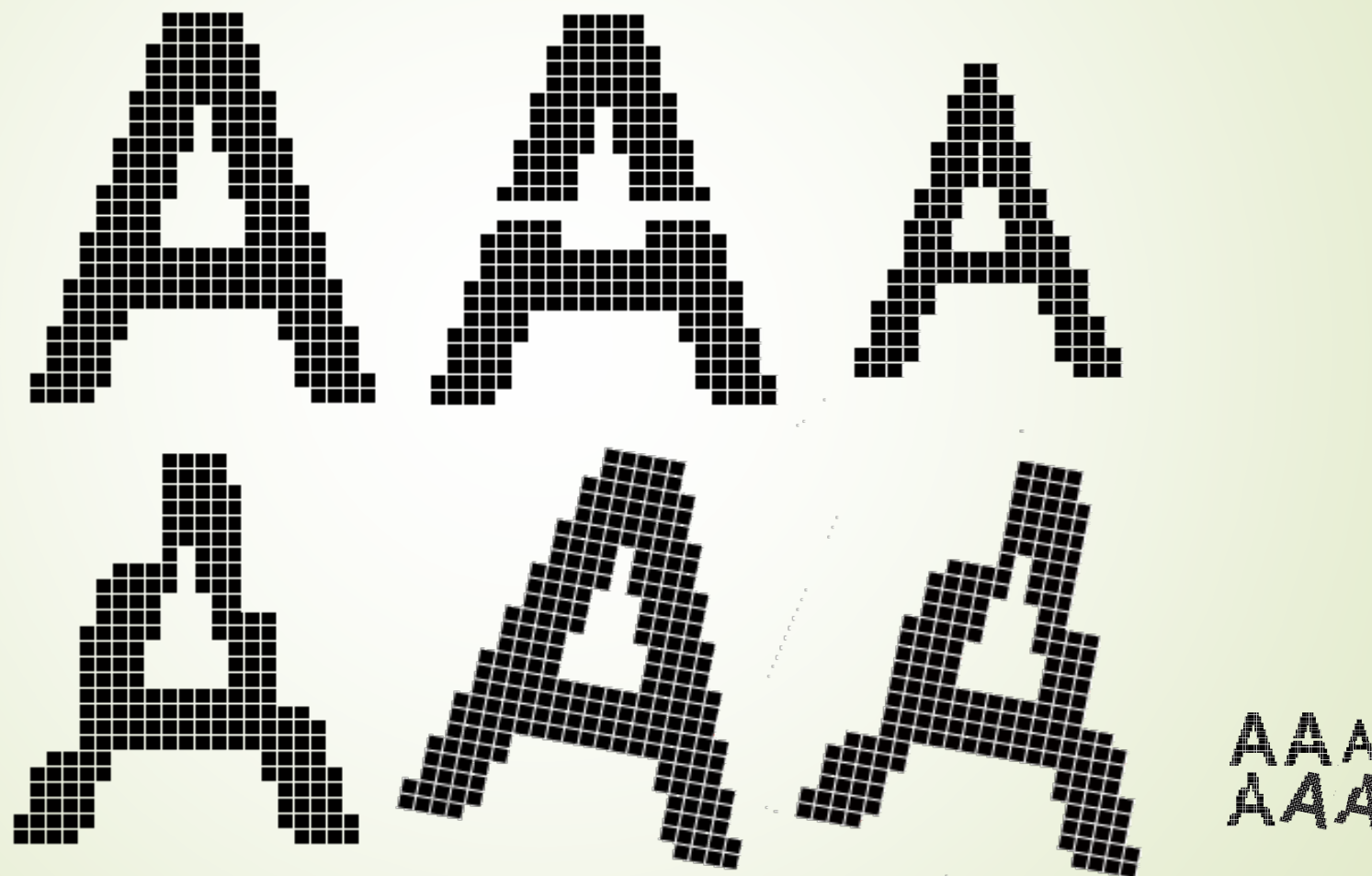
❖ تغییر شکل





154 input change
from 2 shift left
77 : black to white
77 : white to black

مشکلات شبکه عصبی: تغییر اندازه و تغییرات دیگر در ورودی

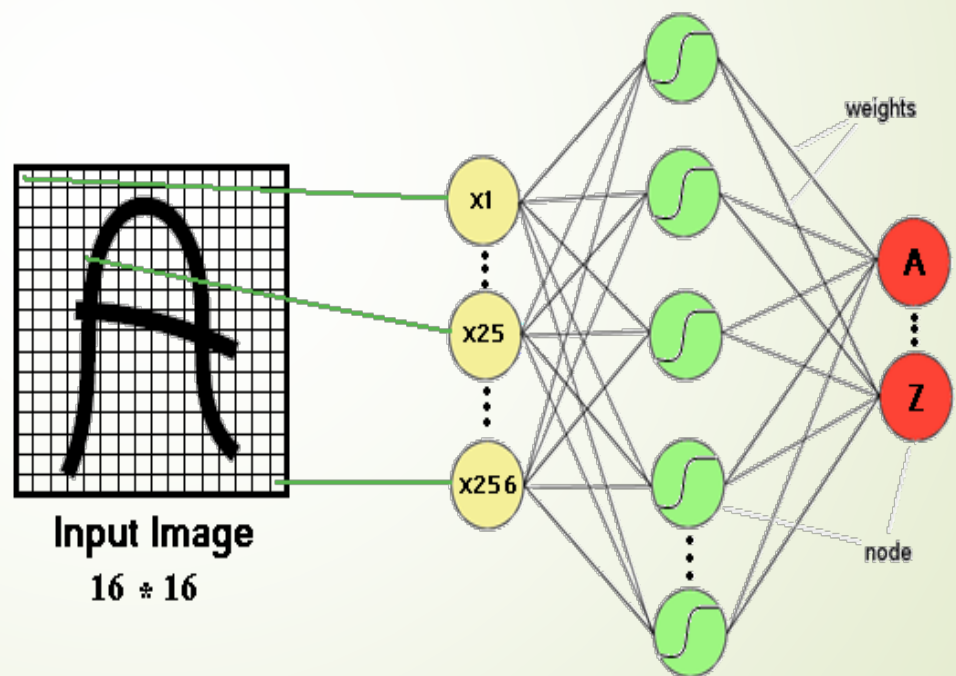
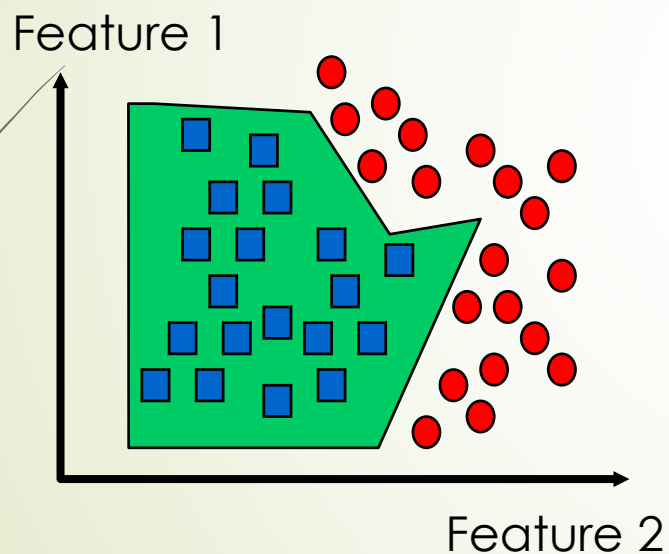


مشکلات شبکه عصبی: داده خام

10

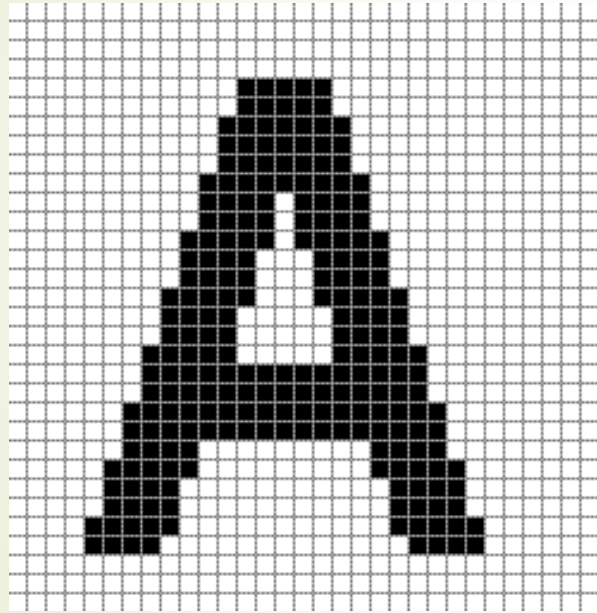
❖ ساختار تصویر (داده) ورودی در نظر گرفته نمی‌شود

❖ داده‌ها به شکل خام استفاده می‌شوند (اندازه بزرگ)



مشکلات شبکه عصبی: بزرگی داده خام

11



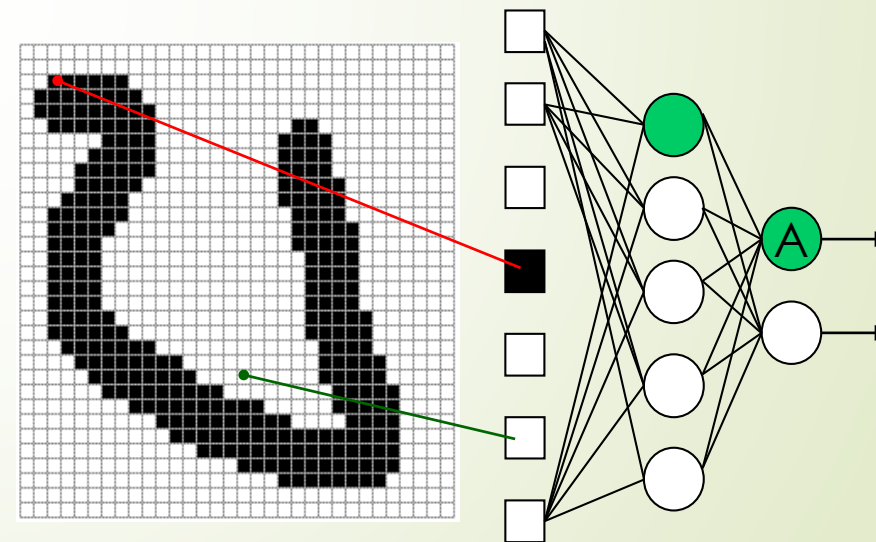
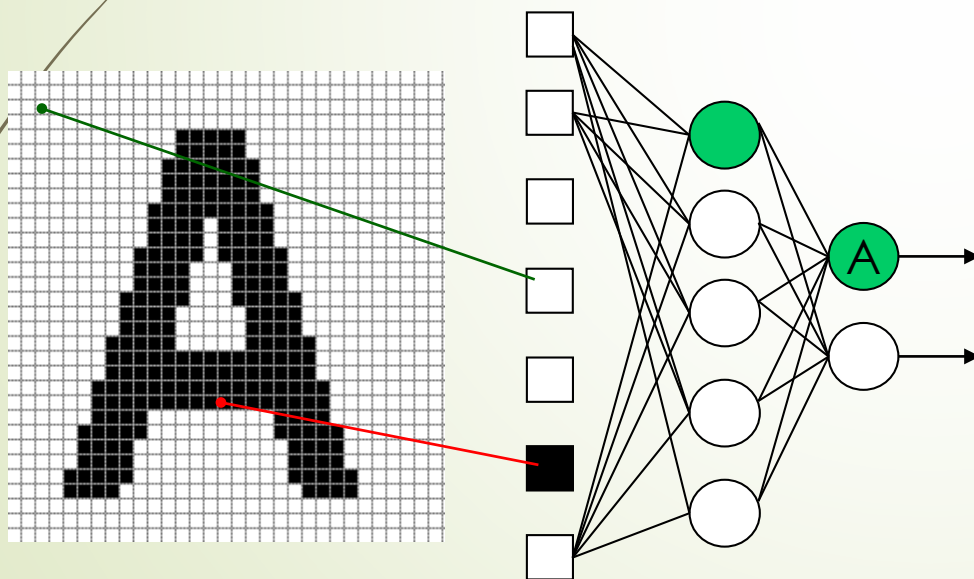
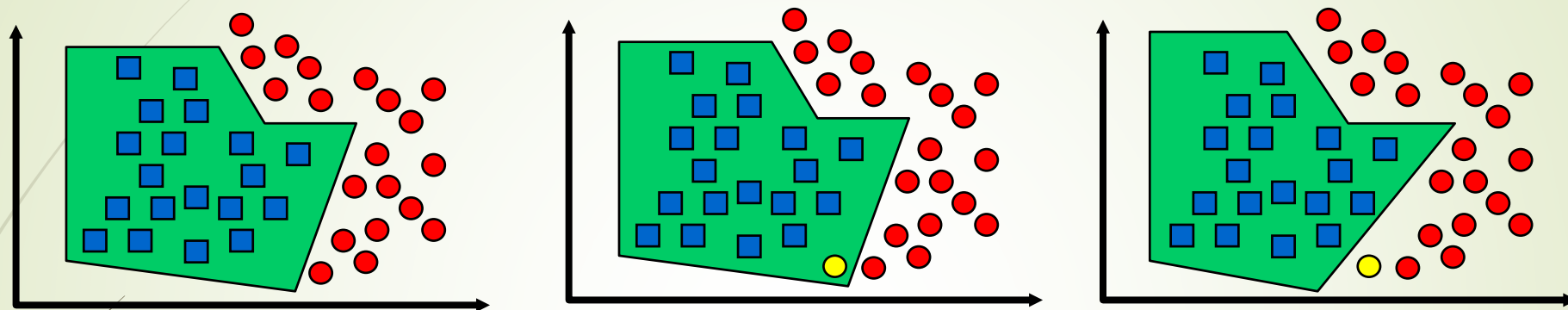
32 * 32 input image

❖ تصویر سیاه و سفید: $2^{32 \times 32} = 2^{1024}$

❖ تصویر خاکستری: $256^{32 \times 32} = 256^{1024}$

مشکلات شبکه عصبی: مدیریت نویز؟

12



حل مشکلات فوق با ...

❖ یک شبکه به اندازه کافی بزرگ، میتواند مشکلات فوق را حل کند!

❖ زمان اجرا؟

❖ اندازه شبکه؟

❖ پارامترهای آزاد؟

❖ تعداد داده‌های آموزشی؟



Convolutional neural network (CNN)

تاریخچه



Yann LeCun, Professor of Computer Science
The Courant Institute of Mathematical Sciences
New York University
Room 1220, 715 Broadway, New York, NY 10003, USA.
(212)998-3283 yann@cs.nyu.edu

- ☯ In 1995, **Yann LeCun** and **Yoshua Bengio** introduced the concept of convolutional neural networks.

درباره CNN

❖ الهام گرفته از ساختار نوروهای عصبی در لایه مربوط به بینایی مغز

❖ ساختار شبکه‌ای که به صورت غیر مستقیم، ویژگی‌هایی از تصویر استخراج می‌کند.

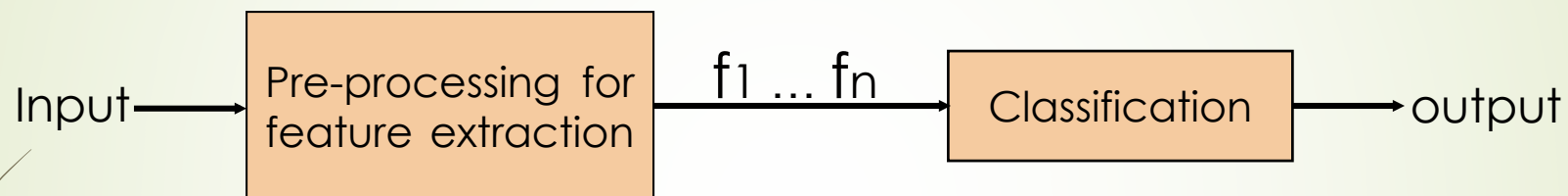
❖ نوعی از شبکه‌های عصبی چند لایه

درباره CNN

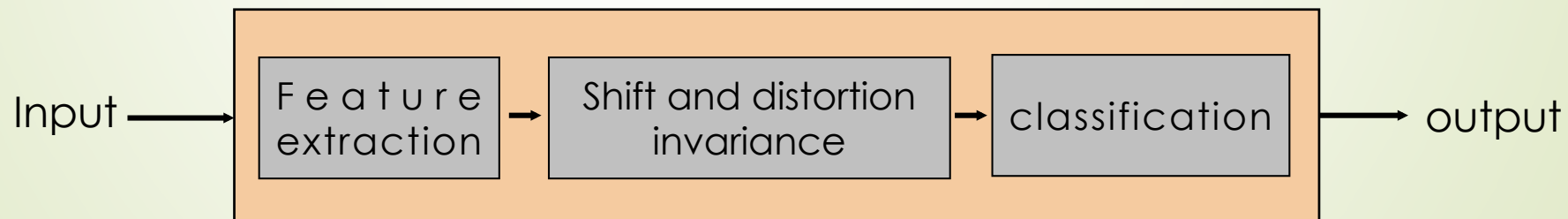
- ❖ CNN شبکه ای feed-forward است
- ❖ ویژگی‌های ساختاری از تصویر استخراج می‌کند.
- ❖ یادگیری با Back-Propagation
- ❖ کار اصلی این شبکه، تشخیص الگوهای بصری از تصویر خام (پیکسل‌ها) با کمترین پیش پردازش
- ❖ قدرت تشخیص الگوی دارای تغییرات زیاد (مثل دست خط)

دسته بندی

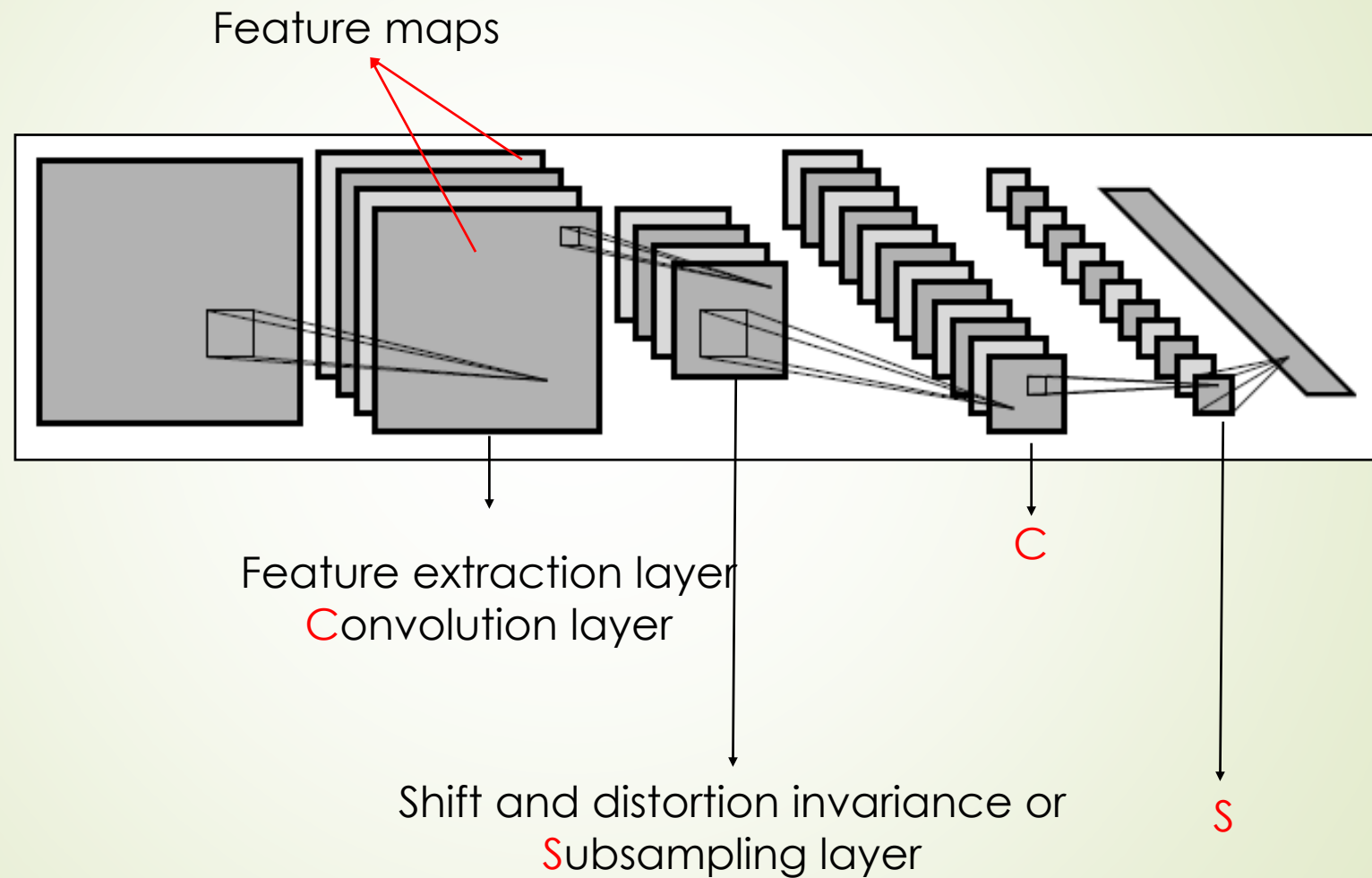
❖ سنتی:



❖ CNN:

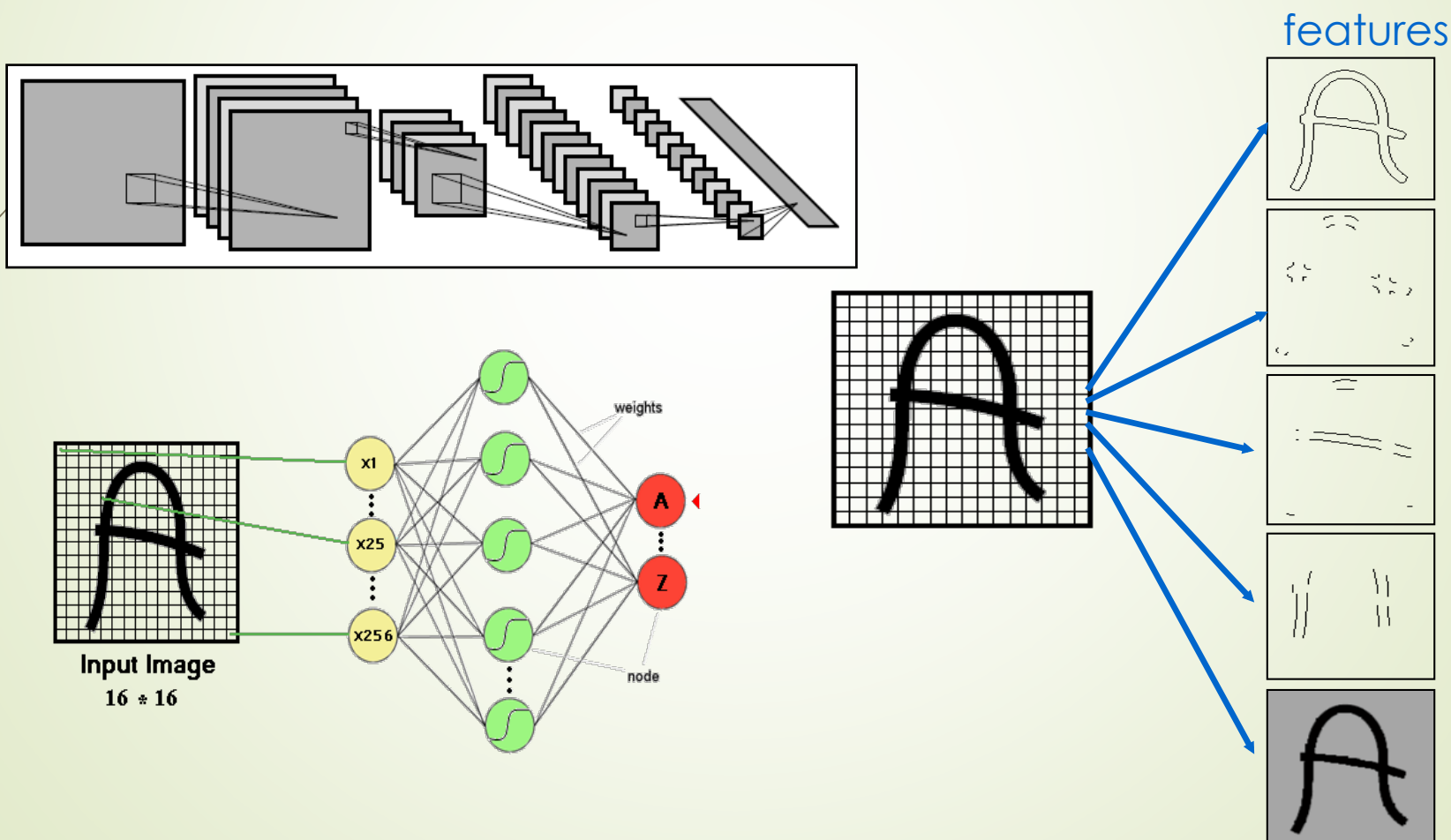


ساختار CNN

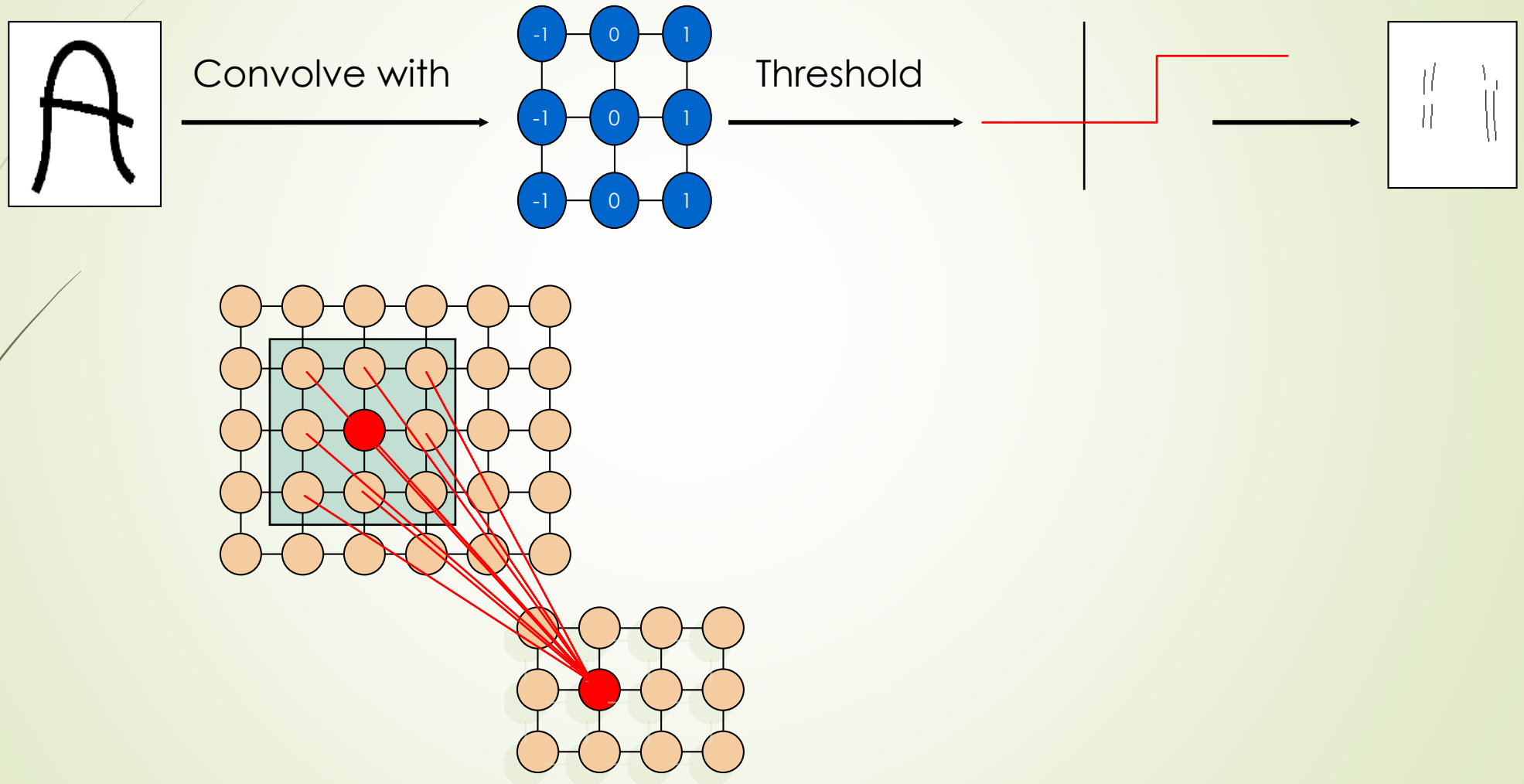


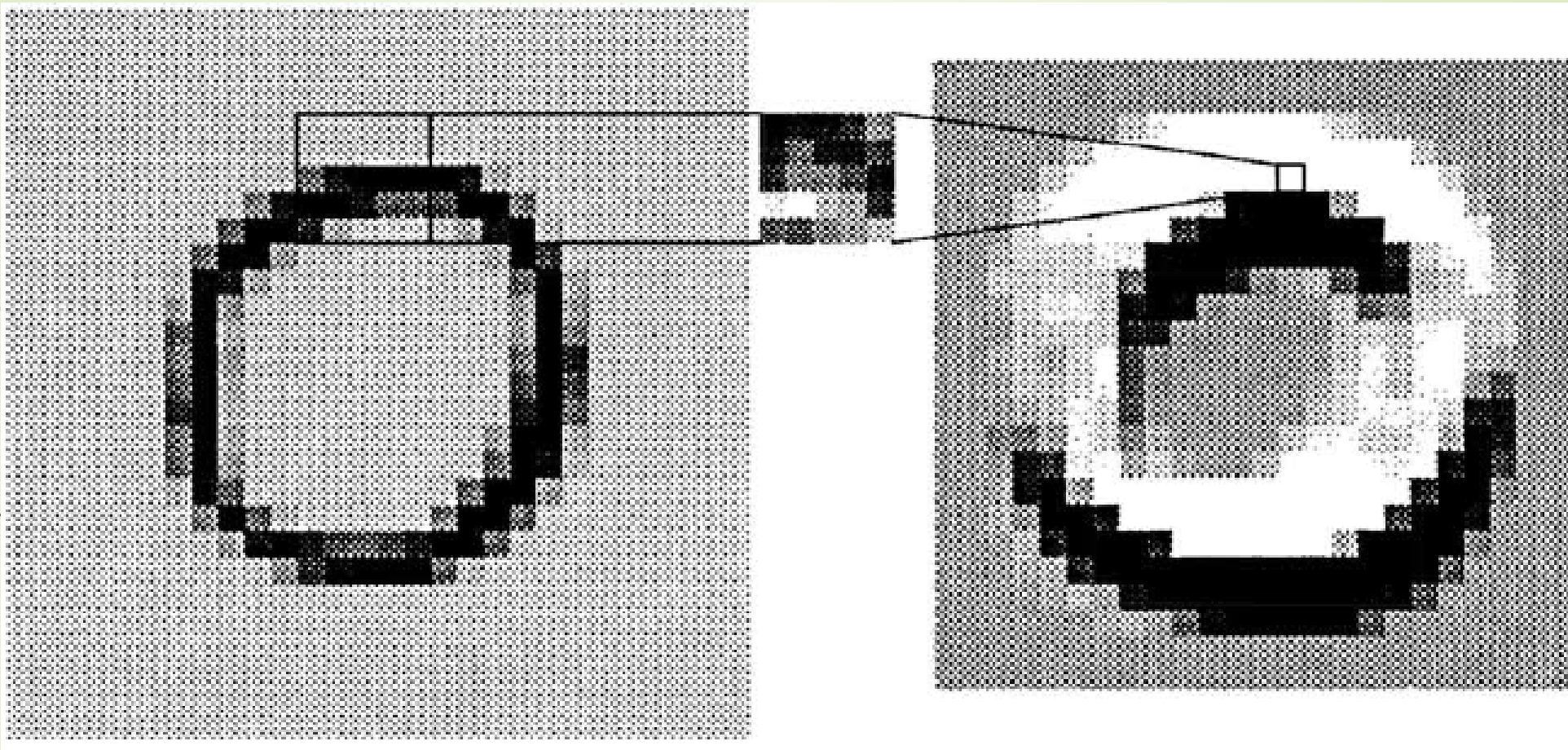
لایه استخراج ویژگی – لایه پیکشی

❖ تشخیص یک (چند) ویژگی خاص از بخش‌های مختلف تصویر ورودی



استخراج ویژگی





❖ هر نورون در لایه نگاشت ویژگی به یک پنجره از تصویر ورودی با یک مجموعه وصل است و نتیجتاً خروجی تمام این پنجره ها در لایه بعدی ذخیره میشود.



10 OUTPUT

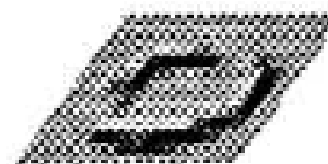
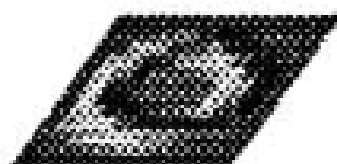
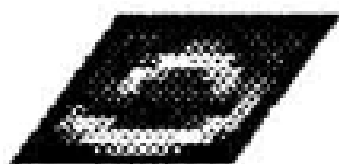
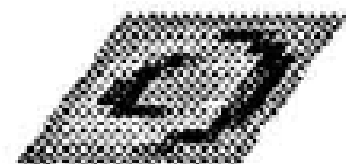
23



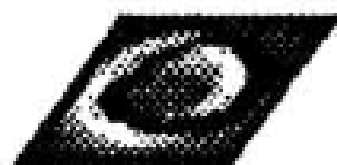
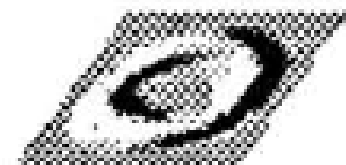
12@4x4



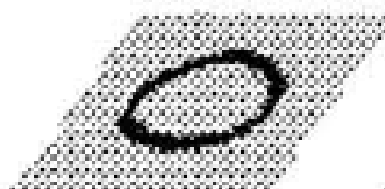
12@8x8



4@12x12



4@24x24



28x28 INPUT

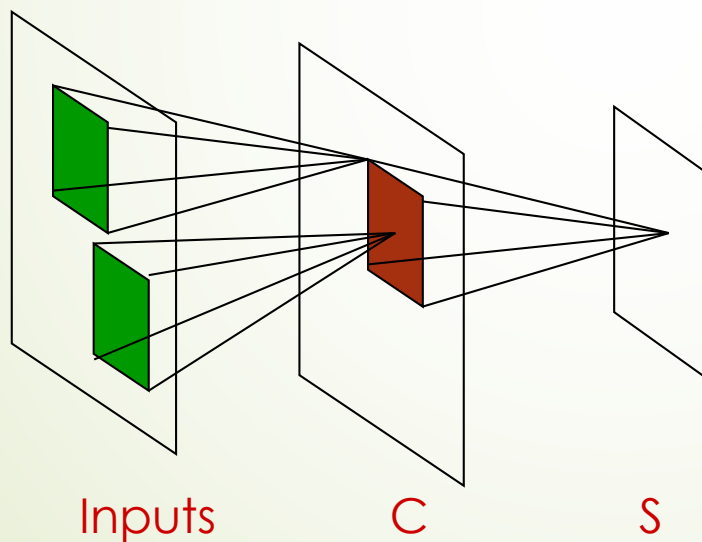
استخراج ویژگی

❖ وزن مشترک:

❖ تمام نورون‌های مربوط به یک ویژگی خاص، وزن مشترک دارند (و نه لزوماً بایاس مشترک)

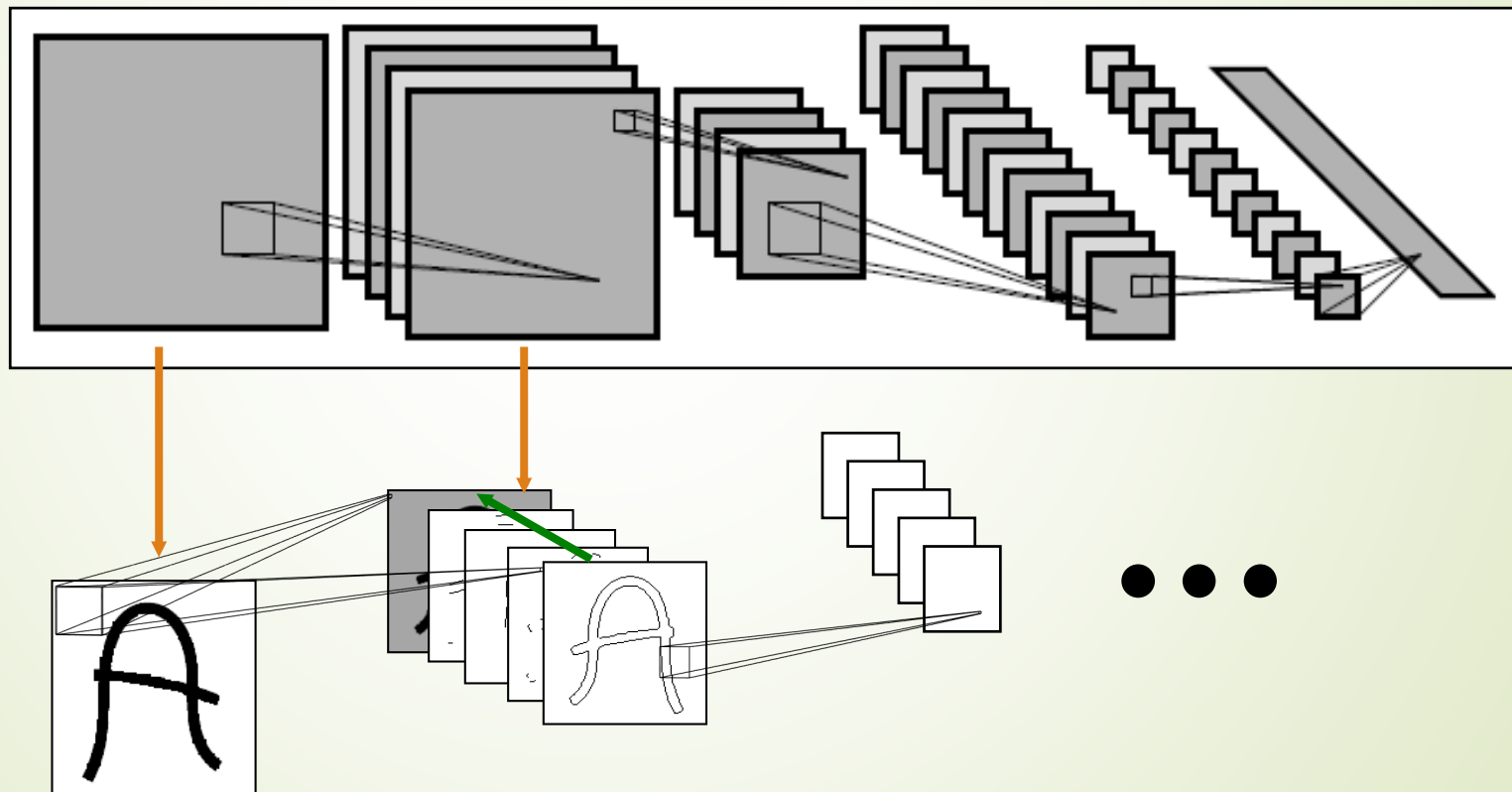
❖ تمام نورون‌های این چینی، ویژگی یکسانی را در نقاط مختلفی از تصویر استخراج می‌کنند.

❖ تعداد پارامترهای آزاد مساله کاهش می‌یابد.



استخراج ویژگی

❖ اگر نورونی در لایه نگاشت ویژگی (Feature Map) فعال شود، یعنی یک الگوی خاص در آن نقطه از تصویر تشخیص داده شده است.



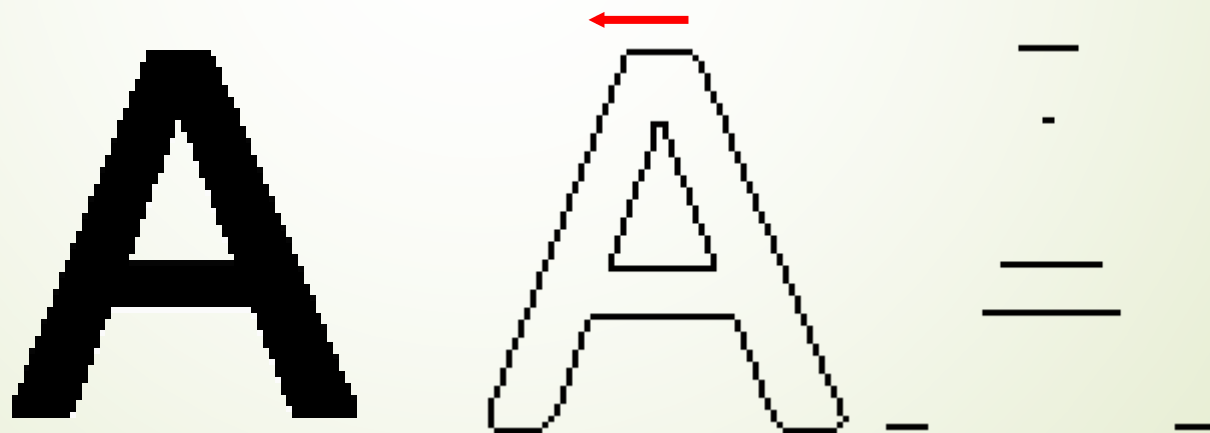
لایه نمونه برداری

❖ کاهش رزولوشن هر نگاشت ویژگی

❖ کاهش رزولوشن:

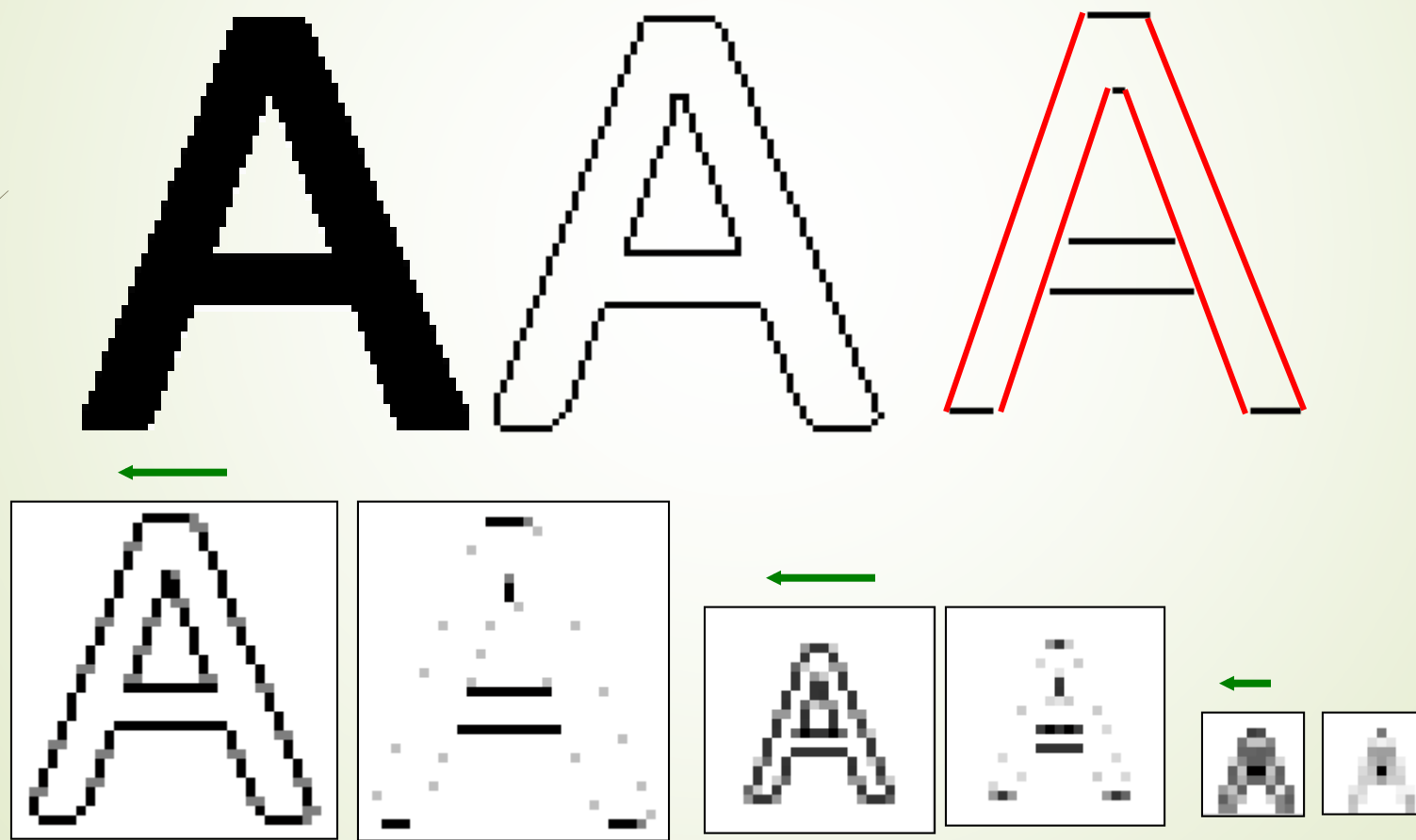
❖ مقاومت در برابر قدری جابجایی در ورودی

❖ مقاومت در برابر قدری دگرگونی در ورودی

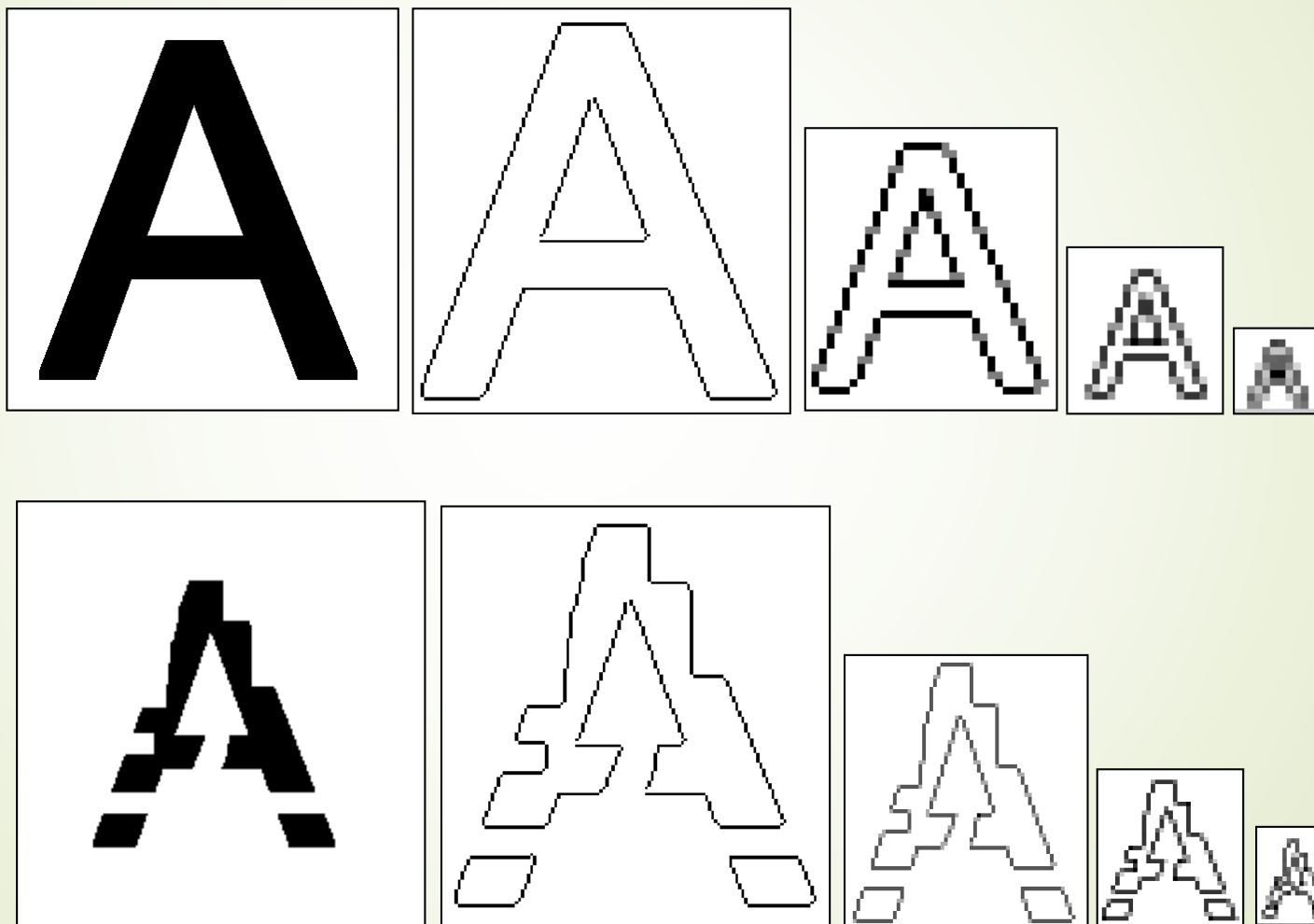


لایه نمونه برداری

❖ کاهش رزولوشن نگاشت ویژگی

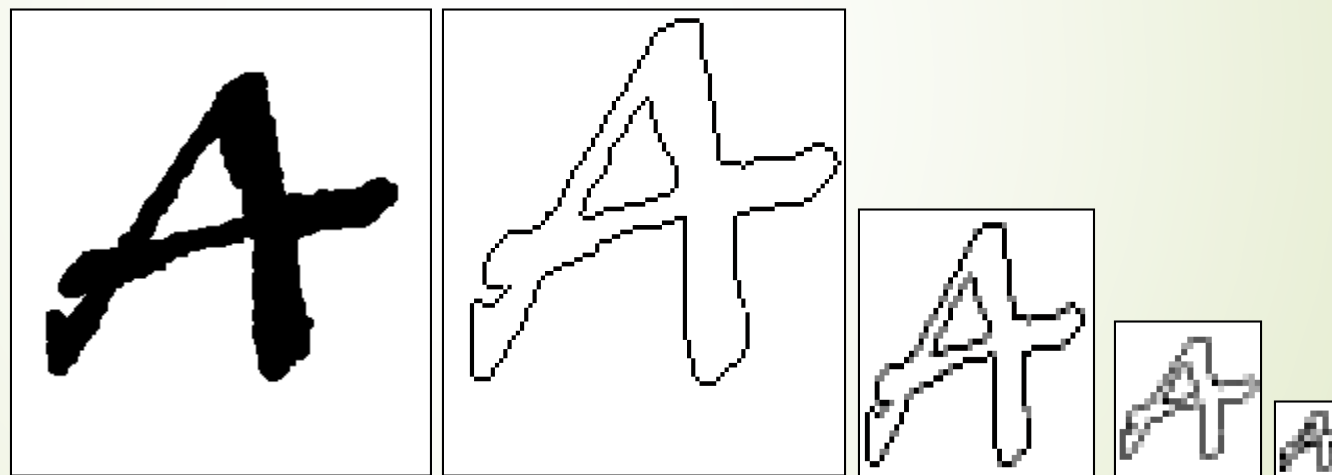


لایه نمونه برداری



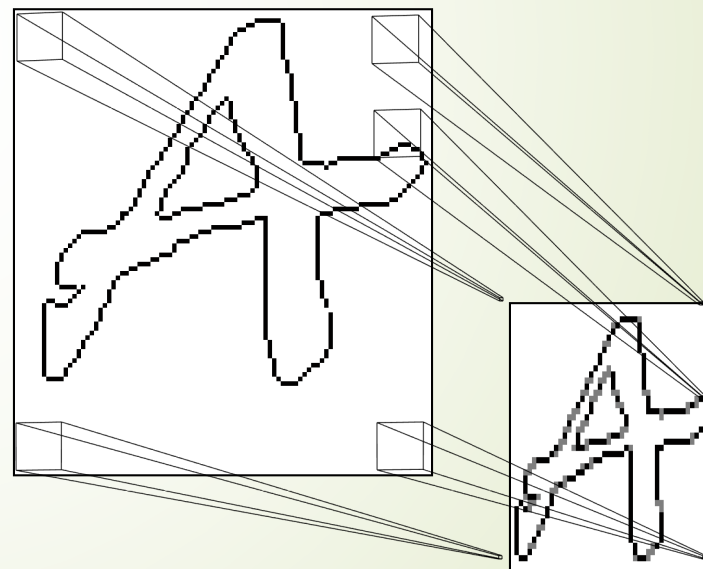
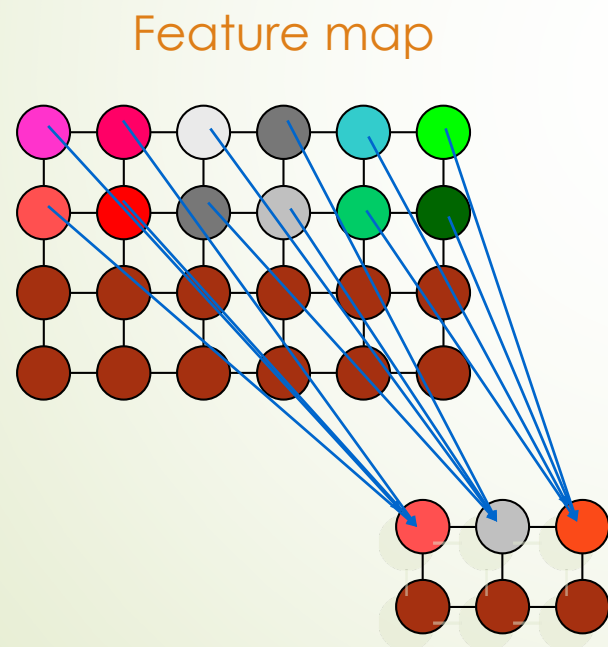
لایه نمونه برداری

❖ اشتراک وزن در لایه نمونه برداری نیز اعمال می شود



لایه نمونه برداری

❖ کاهش تاثیر نویز، جابجایی و دگرگونی در تصویر ورودی



لایه اتصال کامل

❖ بعد از تعدادی لایه پیچشی و لایه نمونه برداری، در لایه آخر میتوان از اتصال کامل استفاده کرد

❖ مانند MLP

❖ چرا همه لایه های دارای اتصال کامل نباشند؟

❖ امکان بیشتر بیش برآزش در شبکه های MLP

❖ به عنوان راه حل، در MPL می توان ترم جریمه برای وزن ها در نظر گرفت

❖ مثلاً تابع خطا شامل مجموع اندازه وزن ها هم باشد تا وزن ها به سمت صفر سوق داده شوند.

❖ راه حل CNN:

❖ استفاده از ساختار سلسله مراتبی الگوهای تصویری

❖ استفاده از ساختارهای ساده برای رسیدن به ساختارهای پیچیده

❖ بنابراین نیازی به داشتن اتصال کامل (و در نتیجه پیچیدگی) در لایه های مختلف شبکه نیست.

❖ با اتصالات محدود و یکسان (و در نتیجه ساختارهای ساده) میتوان به مرور ساختارهای پیچیده تر را نیز ایجاد کرد.

حل مشکل تعداد پارامتر زیاد در CNN



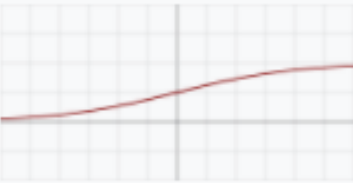

- ❖ در MLP، اگر اندازه تصویر ورودی بزرگ شود، تعداد وزن های ورودی به لایه اول بسیار بالا میرود.
- ❖ تصویر کوچکی به اندازه ۱۰۰ در ۱۰۰ = ۱۰۰۰۰ وزن لایه اول برای هر نورون!
- ❖ برای لایه های بعدی نیز همین میزان وزن/پارامتر آزاد باید بهینه شود.
- ❖ تعداد لایه ها از یک حدی بیشتر نمیتواند باشد.
- ❖ لایه های پیچشی در CNN با کم کردن تعداد پارامترهای آزاد، امکان عمیق تر شدن شبکه فراهم میشود.

ناپدید شدن یا انفجار گرادیان

- ❖ در روش انتشار رو به عقب خطا، امکان دو پدیده وجود دارد
 - ❖ ناپدید شدن یا کم اثر شدن گرادیان
 - ❖ تغییرات شدید و ناگهانی (انفجاری) گرادیان
- ❖ در تقسیم سهم خطا بین لایه های قبلی (در روش انتشار رو به عقب خطا)، مشتق تابع فعالساز به عنوان ضریب نقش دارد
- ❖ مشتق تابع Sigmoid برای مقادیر ورودی کوچک یا بزرگ، خیلی کوچک است.
- ❖ نتیجتاً اگر تعداد لایه های شبکه زیاد باشد، ممکن است خطا بین لایه ها به مرور ناپدید شود.
- ❖ همچنین اگر مشتق بزرگ باشد (یا وزن ها بزرگ باشند) ممکن است در لایه های مختلف این مقادیر بزرگ هم افزایی ایجاد کنند و سبب خیلی بزرگ (NaN) شدن خطای منتشر شده شوند.
- ❖ راه حل ها: تغییر تابع فعالساز، دقت در مقدار دهی اولیه وزنها (تصادفی؟ صفر؟ بزرگ؟ کوچک؟)، تغییر معماری شبکه و ...

تابع فعال ساز



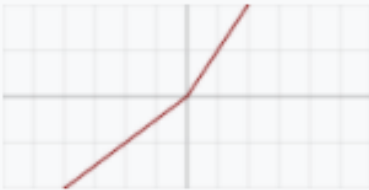
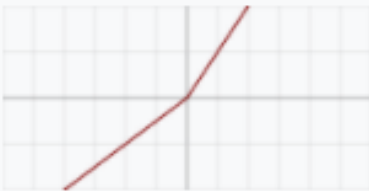
34

Name	Plot	Equation
Identity		$f(x) = x$
Binary step		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Logistic (a.k.a. Sigmoid or Soft step)		$f(x) = \sigma(x) = \frac{1}{1 + e^{-x}}$ [1]
TanH		$f(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$

مساله ناپديد شدن / انفجار گراديان در تابع Logistic يا Tanh

استفاده در لایه‌های
پیش‌بینی CNN



Name	Plot	Equation
Rectified linear unit (ReLU) ^[11]		$f(x) = \begin{cases} 0 & \text{for } x \leq 0 \\ x & \text{for } x > 0 \end{cases} = \max\{0, x\} = x \mathbf{1}_{x>0}$
Gaussian Error Linear Unit (GELU) ^[6]		$f(x) = x\Phi(x) = \frac{1}{2}x \left(1 + \operatorname{erf} \left(\frac{x}{\sqrt{2}} \right) \right)$
Leaky rectified linear unit (Leaky ReLU) ^[15]		$f(x) = \begin{cases} 0.01x & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$
Parameteric rectified linear unit (PReLU) ^[16]		$f(\alpha, x) = \begin{cases} \alpha x & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$

Pooling

- ❖ Convolutional networks may include local or global pooling layers to streamline the underlying computation.
- ❖ Pooling layers reduce the dimensions of the data by combining the outputs of neuron clusters at one layer into a single neuron in the next layer.
- ❖ Local pooling combines small clusters, typically 2×2 .
- ❖ Global pooling acts on all the neurons of the convolutional layer.
- ❖ In addition, pooling may compute a max or an average.
 - ❖ *Max pooling* uses the maximum value from each of a cluster of neurons at the prior layer.
 - ❖ *Average pooling* uses the average value from each of a cluster of neurons at the prior layer.

AlexNet (2012)

- ❖ Small datasets like CIFAR and Caltech
 - ❖ Sufficient for machine learning models to learn basic recognition tasks.
 - ❖ Real life is never simple and has many more variables than are captured in these small datasets.
- ❖ Large datasets like ImageNet, which consist of hundreds of thousands to millions of labeled images
- ❖ The need for an extremely capable deep learning model.
 - ❖ AlexNet

ImageNet

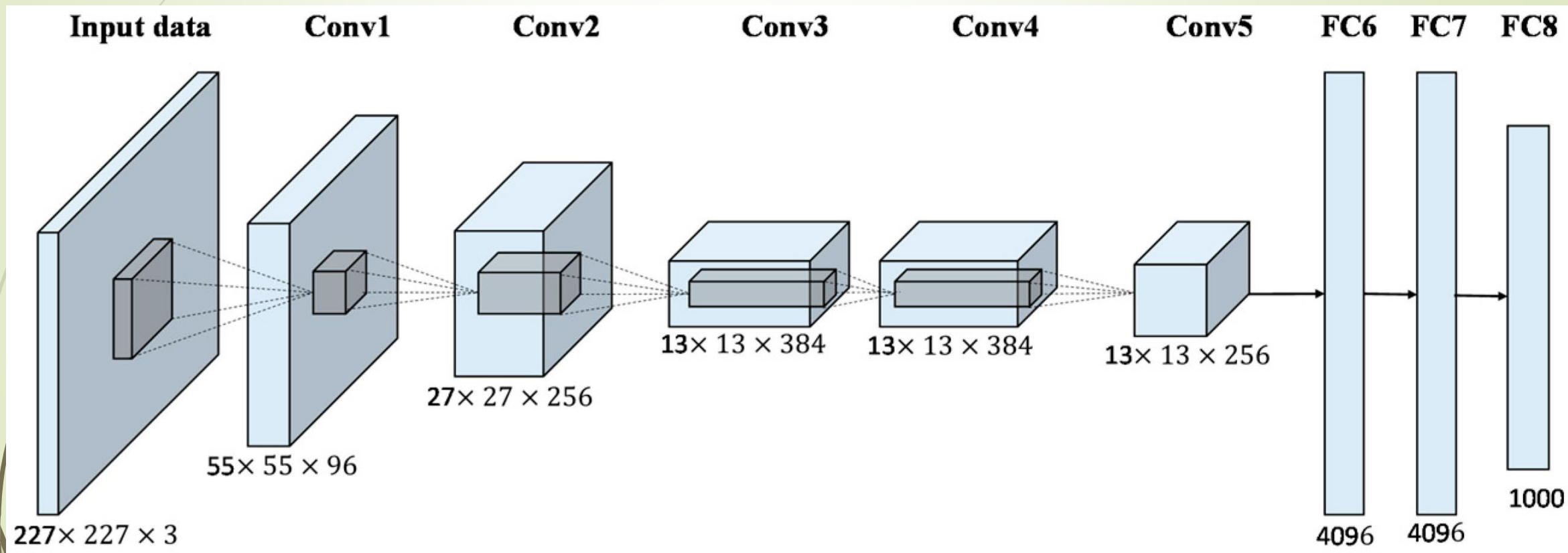
- ❖ More than 15 million high-resolution images
- ❖ 22 thousand classes
- ❖ ImageNet even has its own competition:
 - ❖ the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC). T
 - ❖ Uses a subset of ImageNet's images and challenges researchers to achieve the lowest top-1 and top-5 error rates
 - ❖ Data is not a problem; there are about 1.2 million training images, 50 thousand validation images, and 150 thousand testing images.
 - ❖ The authors enforced a fixed resolution of 256x256 pixels for their images by cropping out the center 256x256 patch of each image.

AlexNet

39

❖ The architecture consists of eight layers:

- ❖ five convolutional layers
- ❖ three fully-connected layers.



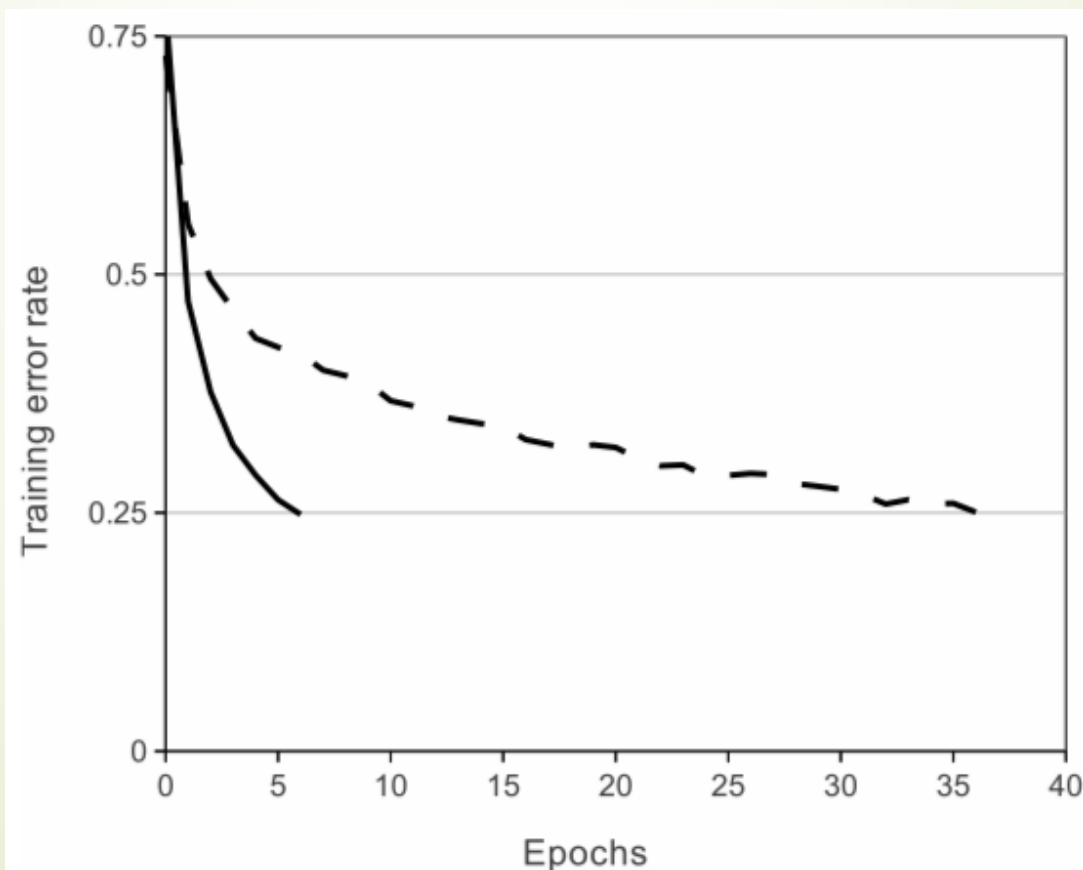
❖ What makes AlexNet special?

What makes AlexNet special?

40

❖ ReLU Nonlinearity:

- ❖ AlexNet uses Rectified Linear Units (ReLU) instead of the tanh function, which was standard at the time.
- ❖ ReLU's advantage is in training time; a CNN using ReLU was able to reach a 25% error on the CIFAR-10 dataset six times faster than a CNN using tanh.



What makes AlexNet special?

- ❖ **Multiple GPUs.**

- ❖ What is GPU?

- ❖ Back in the day, GPUs were still rolling around with 3 gigabytes of memory
- ❖ This was especially bad because the training set had 1.2 million images.
- ❖ AlexNet allows for multi-GPU training by putting half of the model's neurons on one GPU and the other half on another GPU.
 - ❖ Bigger model can be trained
 - ❖ Cuts down on the training time

What makes AlexNet special?

42

❖ **Overlapping Pooling**

- ❖ CNNs traditionally “pool” outputs of neighboring groups of neurons with no overlapping.
- ❖ Introducing the overlap, results in reduction in error by about 0.5%
 - ❖ models with overlapping pooling: harder to overfit

What makes AlexNet special?

43

- ❖ **The Overfitting Problem**
- ❖ AlexNet had 60 million parameters
- ❖ A major issue in terms of overfitting.
- ❖ Two methods were employed to reduce overfitting:
 - ❖ **Data Augmentation**
 - ❖ **Dropout**