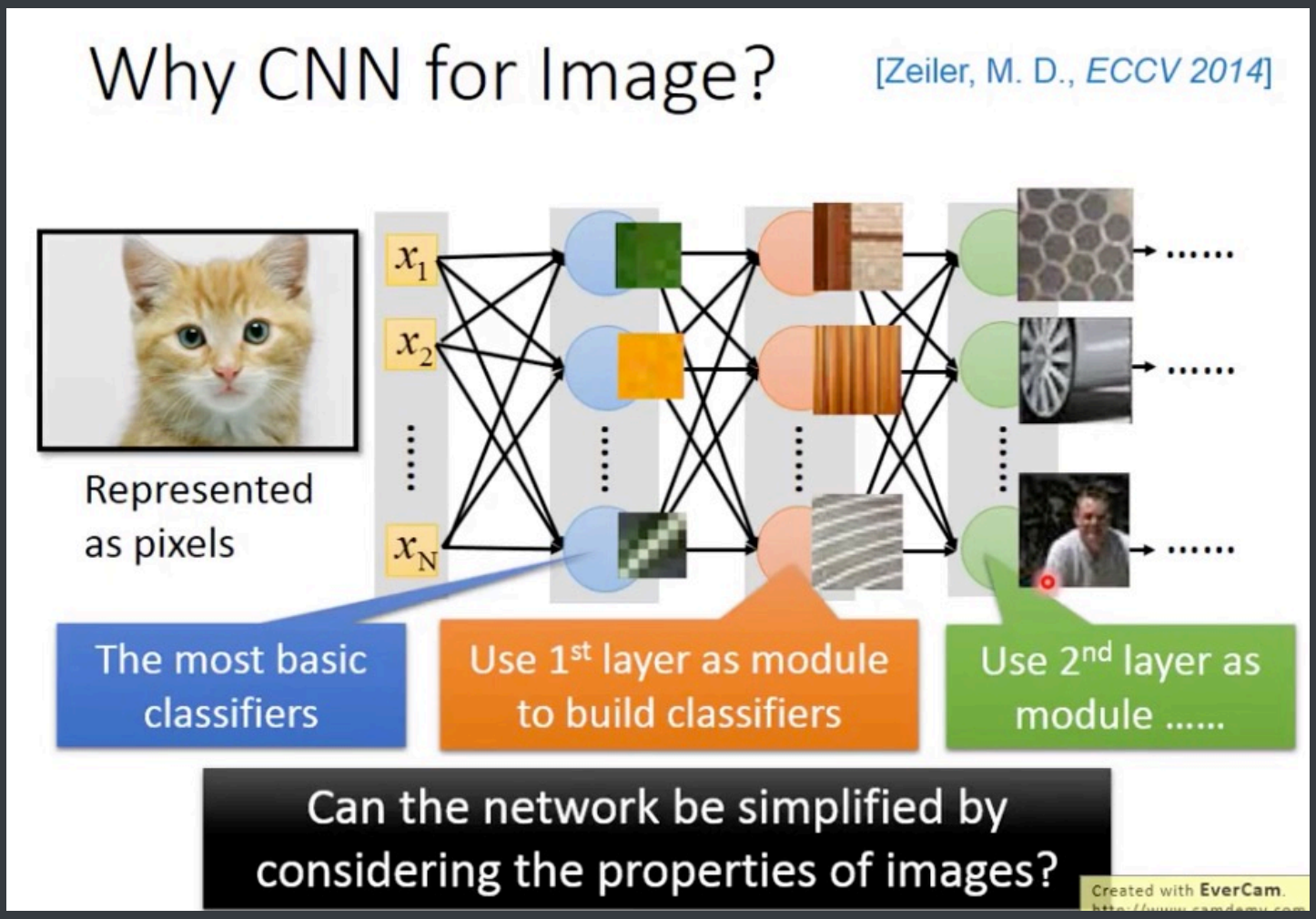
CNN



What CNN did is simplifying the structure of the Neural Network: Using prior knowledge to remove some parametes in the structure.

Why CNN for Image?

- Some paterns are much smaller than the whole image. A neuron doesn't have to see the whole image to discover the pattern.

(e.g. detect a bird in a picture, a layer detect bird beak, a layer detect bird wings etc. A layer only needs to see the part of the image that potentially contians a bird beak - no need to see the whole bird picture)

- The same patterns appear in different regions.

  (e.g. beak can be at the upper-left of the picture, and can be middle of the picture, the 'upper-left beak detector' and 'middle beak detector' do almost the same thing. They can use the same set of parameters)

- Subsampling the pixels will not change the object.

(use subsampling to make image smaller >>> less parameter for the network to process the image)

The whole CNN

Image >>> Convolution >>> Max Pooling >>> Convlution >>> Max Pooling ....Flatten >>> Fully Connected Feedforward network >>> cat/dog

The 'Convonlution >>> Max Pooling' can repeat many times

3 Properities of CNN:

1) Some patterns are much smaller than the whole image - Convolution layer

2) The same patterns appear in different regions - Convolution layer

3) Subsampling the pixels will not change the object - Max Pooling

Example 1: Black and White image (one layer: 0 – no ink, 1 – ink)

CNN – Convolution

Those are the network parameters to be learned.

1	0	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	1	0
0	0	1	0	1	0

6 x 6 image

1	-1	-1
-1	1	-1
-1	-1	1

Filter 1

Matrix

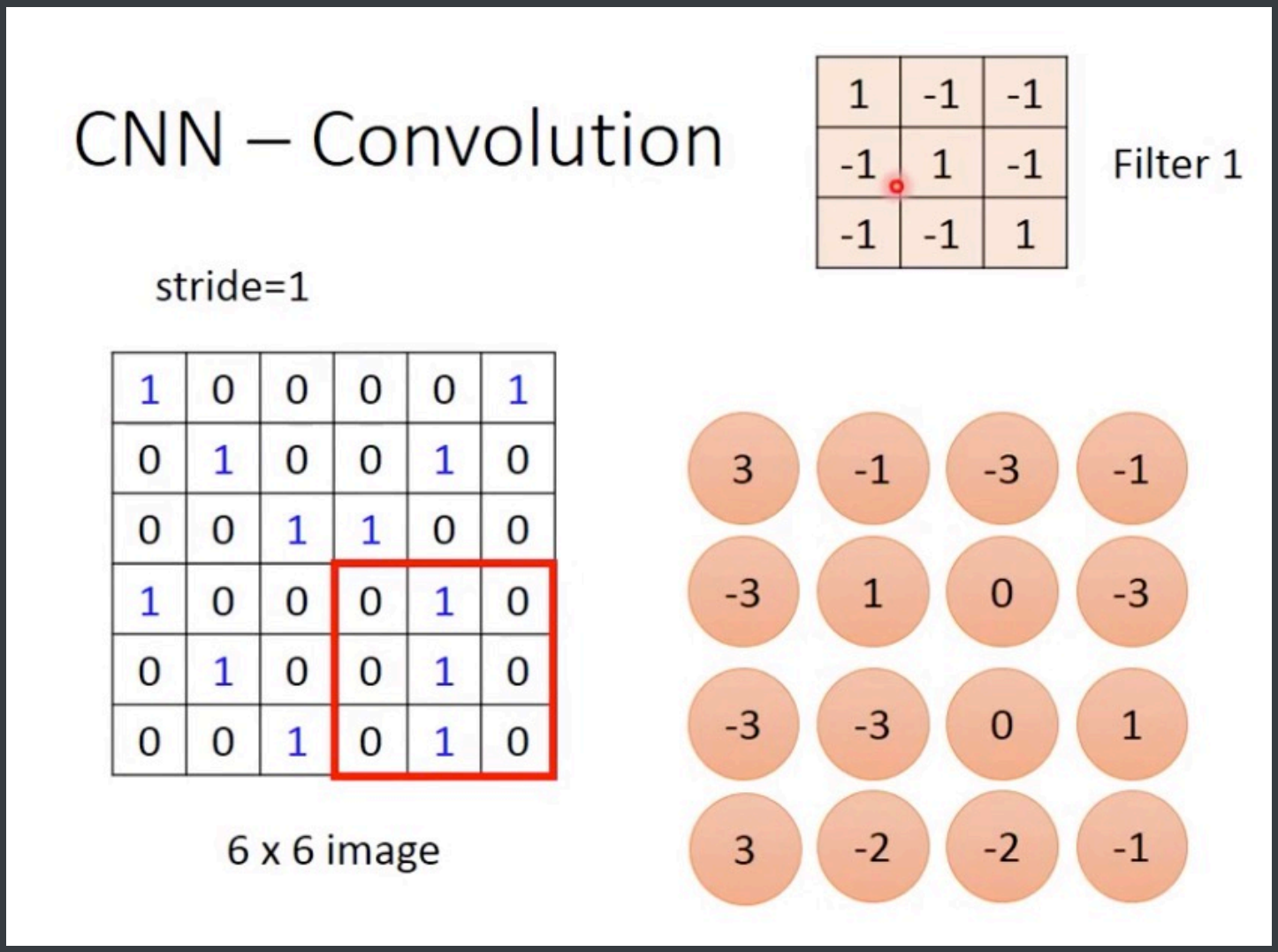
-1	1	-1
-1	1	-1
-1	1	-1

Filter 2

Matrix

⋮

Property 1 Each filter detects a small pattern (3 x 3).

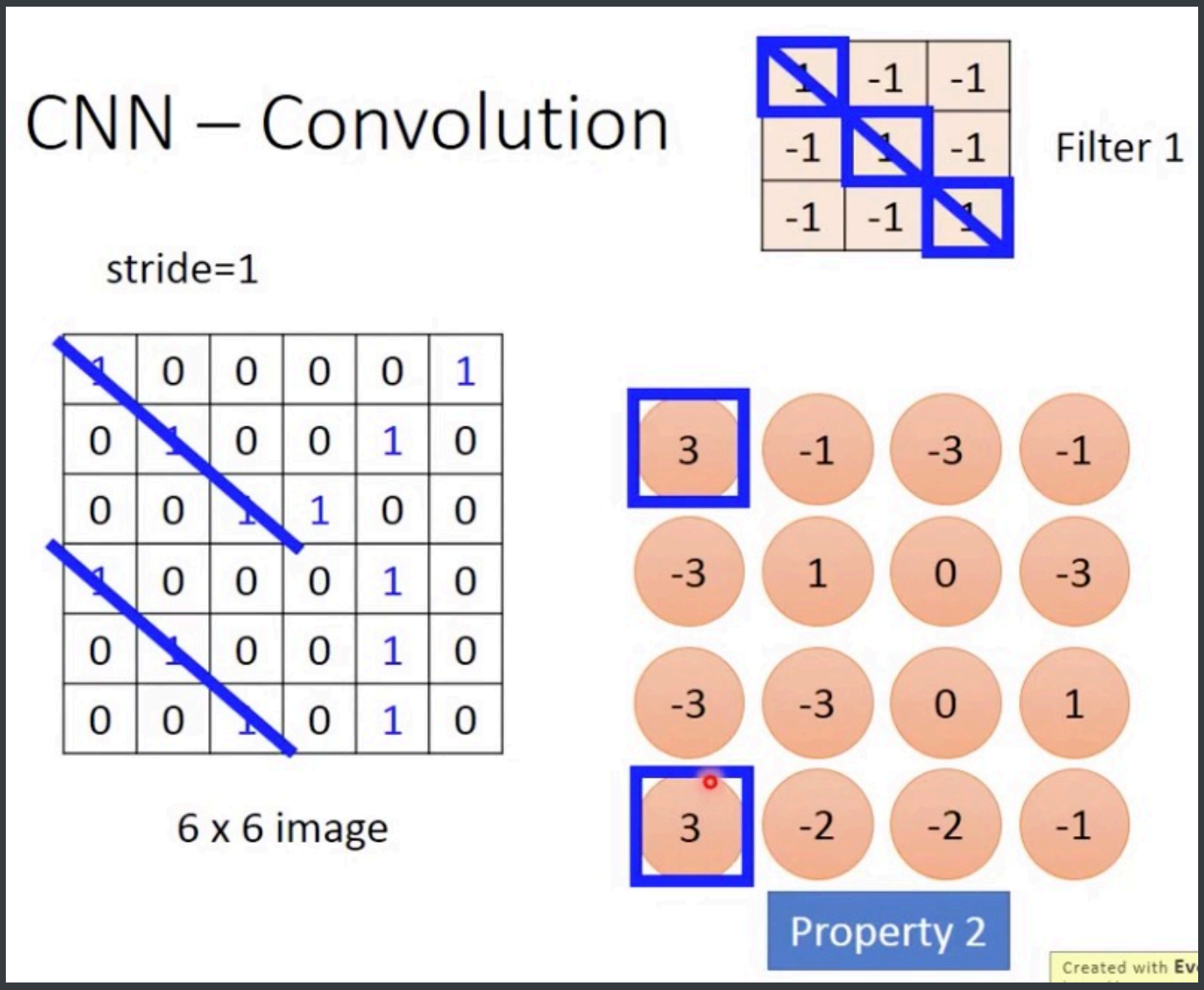


Set up the stride, and use the 'Filter' to move on the image.

Calculate the inner product of the matrix and have the number. e.g.

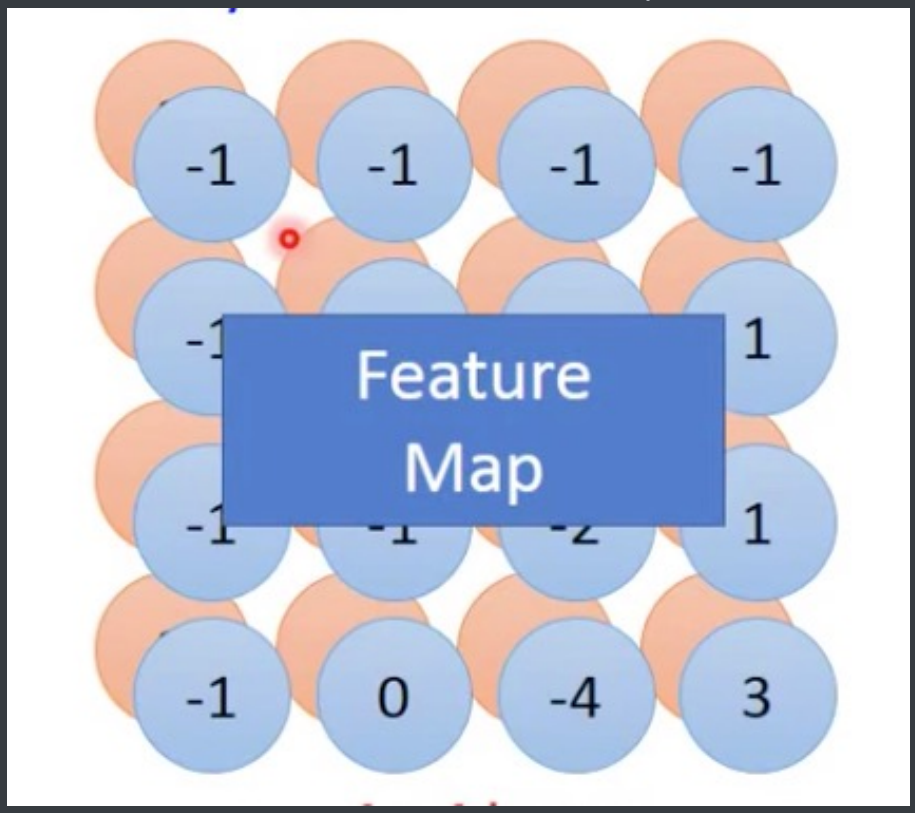
$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & -1 & -1 \\ -1 & 1 & -1 \\ -1 & -1 & 1 \end{bmatrix} = 0*1+1*-1+0*-1+1*-1+1*1+0*-1+0*-1+1*-1+0*1=-1$$

Property1:



We can detect the maximum value for filter 1 is at the left top and left bottom. Filter 1 is to detect the (1, 1, 1) diagonal feature.

Do the same for each filter. Put them on top of each other. Have a Feature Map.



Example 2: Colorful Image