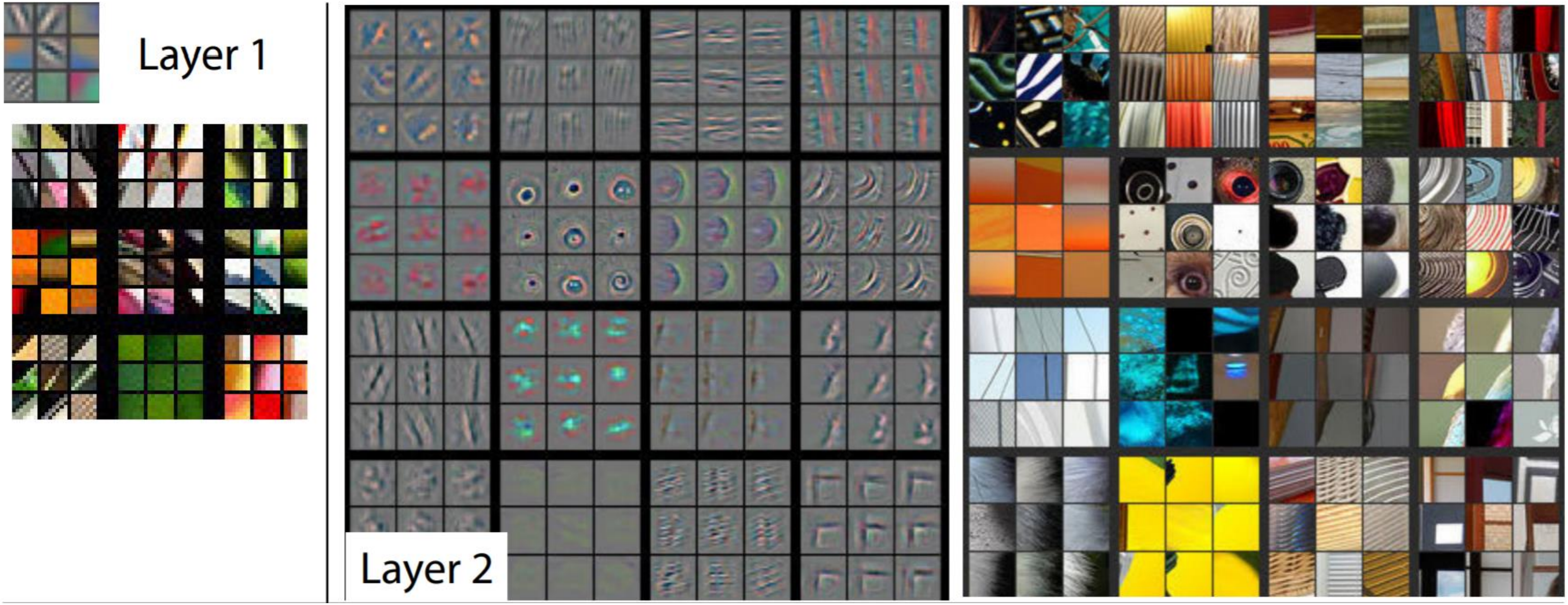
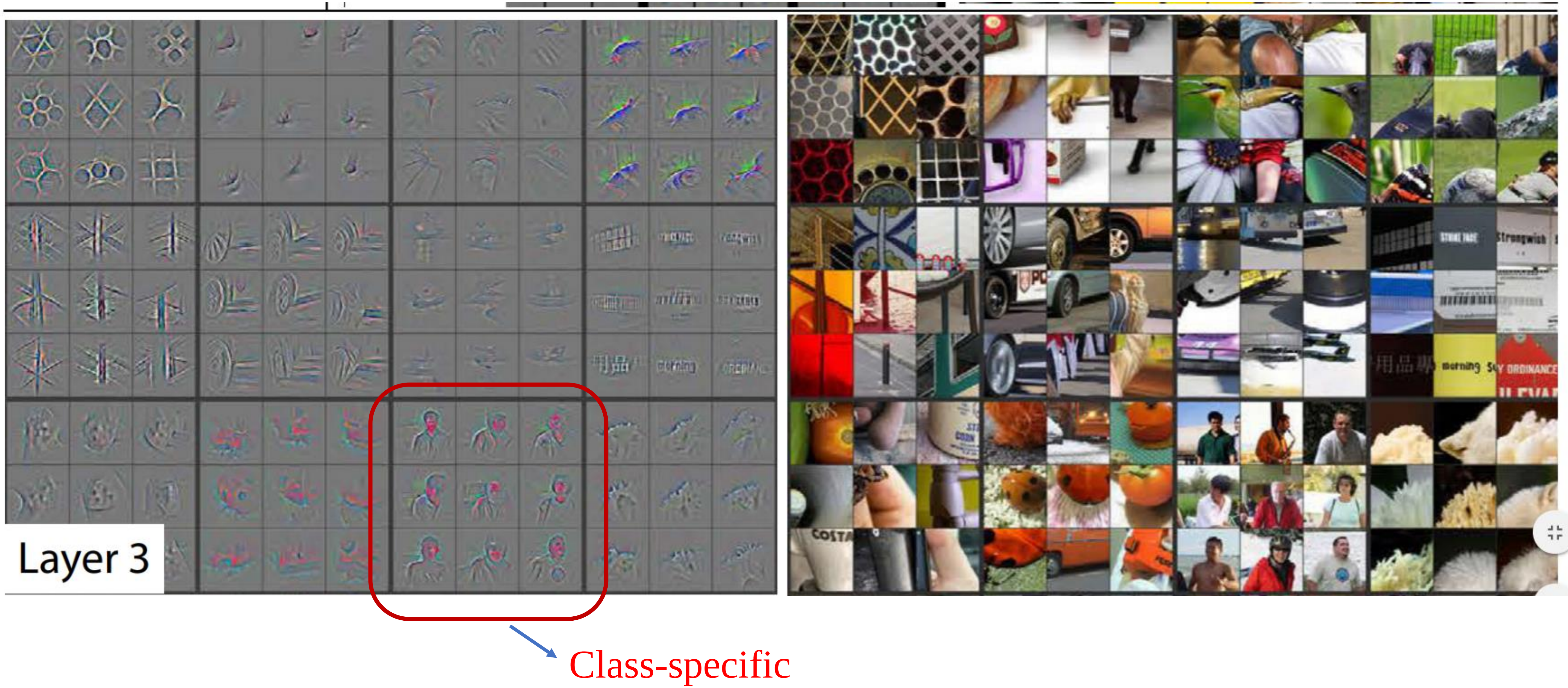
Understanding the Convolutional Network

➢ What kind of feature does every layer learn?



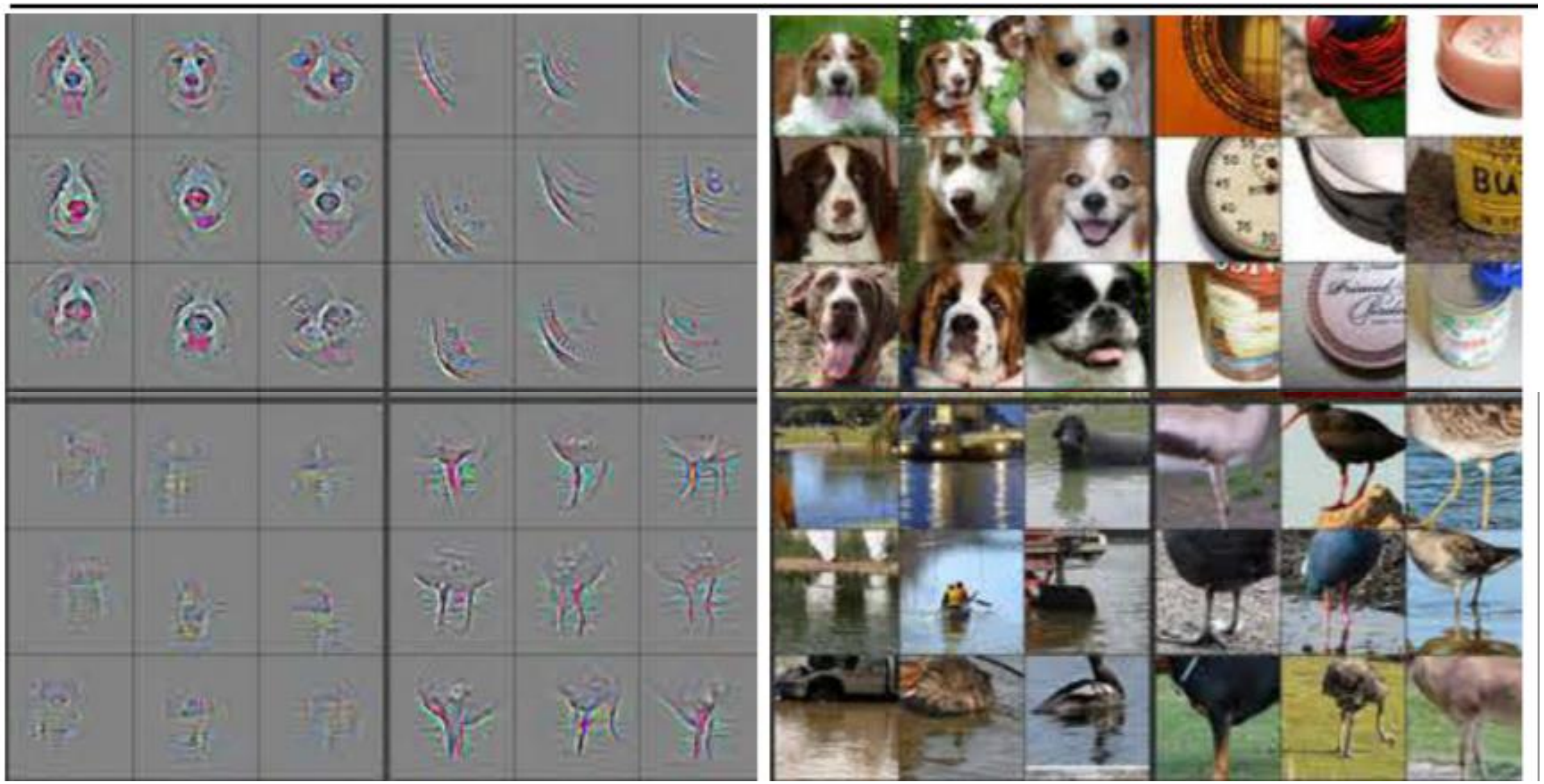
[1] Visualizing and Understanding Convolutional Networks

- What kind of feature does every layer learn?



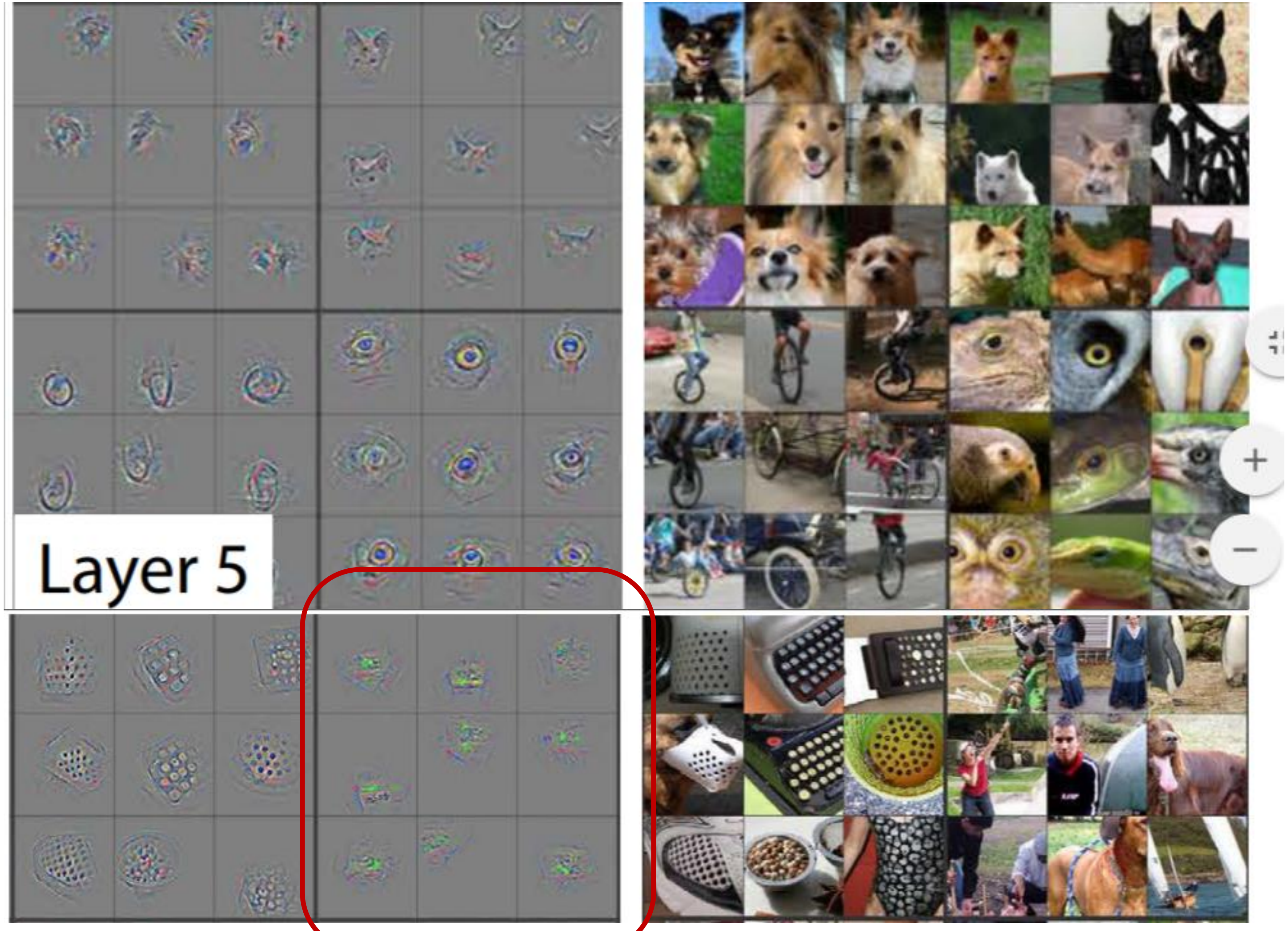
Understanding the Convolutional Network

- What kind of feature does every layer learn?



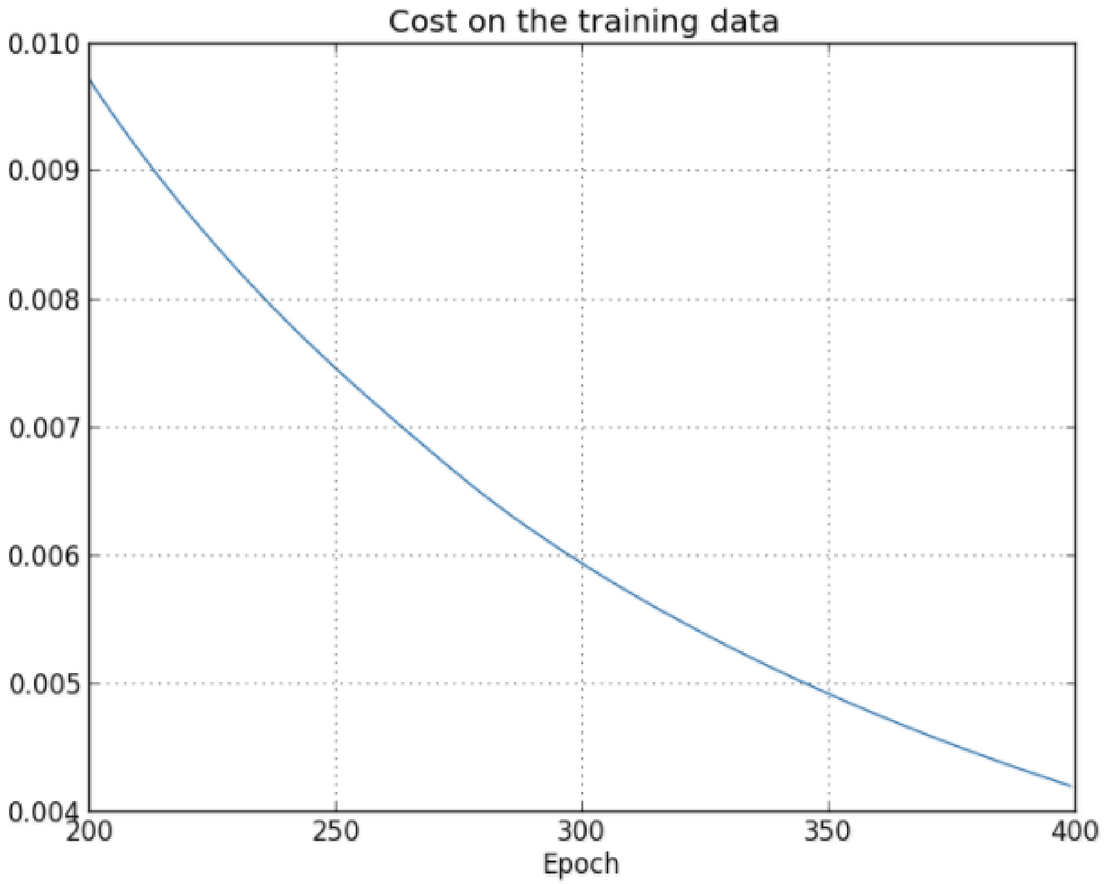
Become more and more class-specific

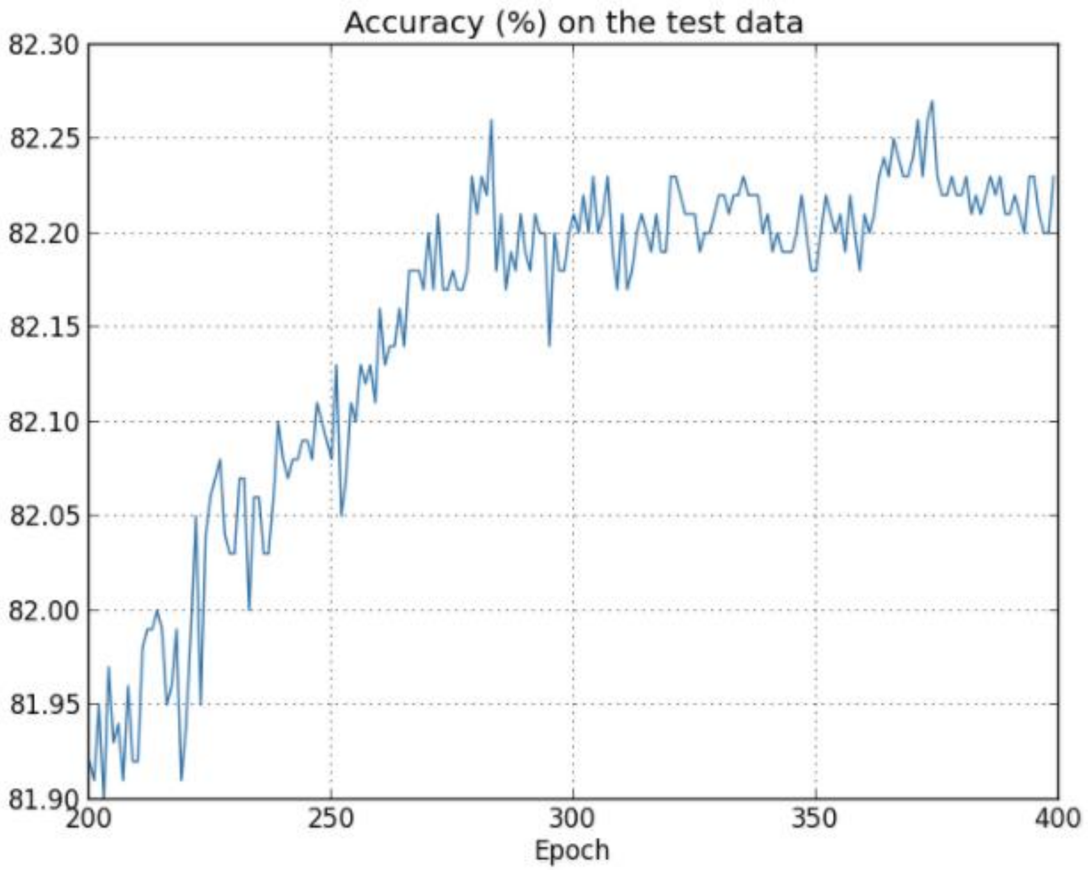
➢ What kind of feature does every layer learn?



Object of the same class with different pose variations

➢ How to decide that a network is overfitting?





The training loss keeps decreasing while the test accuracy stops improving or start to decrease

Figure : http://neuralnetworksanddeeplearning.com/chap3.html

➢ How to prevent overfitting

- Early stop choose the right time to stop
- Increasing the size of training set Sometimes large dataset is not available

Data Augmentation[1]
- noise, cropping, transformation (flipping, rotating)
- Generative model (VAE or GAN)

- Regularization L2 regularization

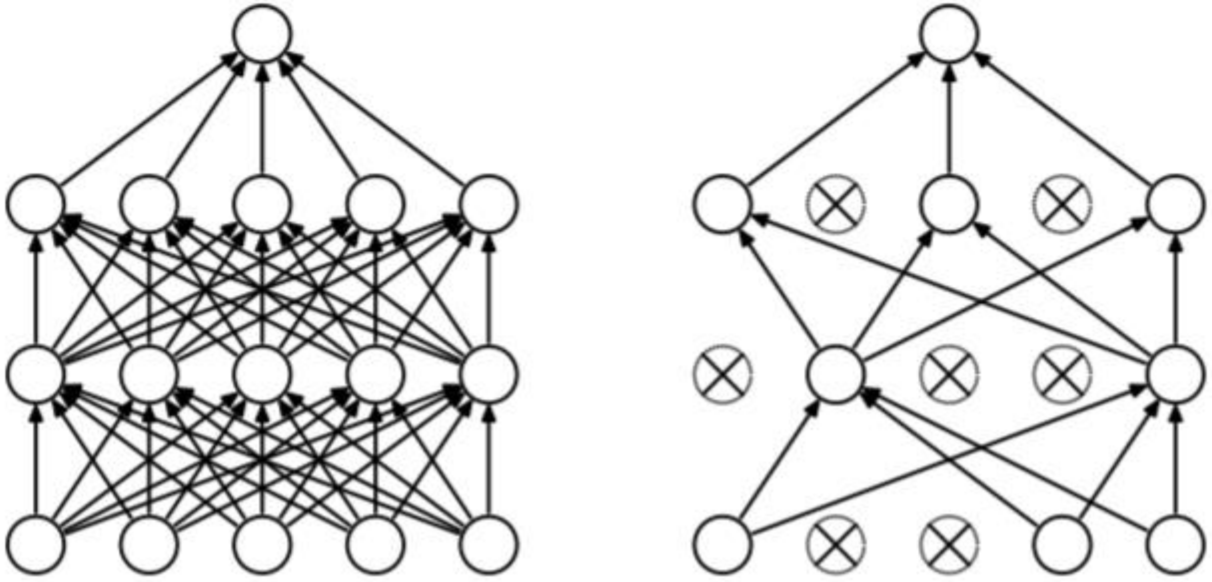
$$C = -\frac{1}{n}\sum_{xj}\left[y_j \ln a_j^L + (1-y_j)\ln(1-a_j^L)\right] + \frac{\lambda}{2n}\sum_w w^2.$$

- Dropout [2]

[1] The Effectiveness of Data Augmentation in Image Classification using Deep Learning
[2]Dropout: A Simple Way to Prevent Neural Networks from Overfitting

How to prevent overfitting

- Dropout

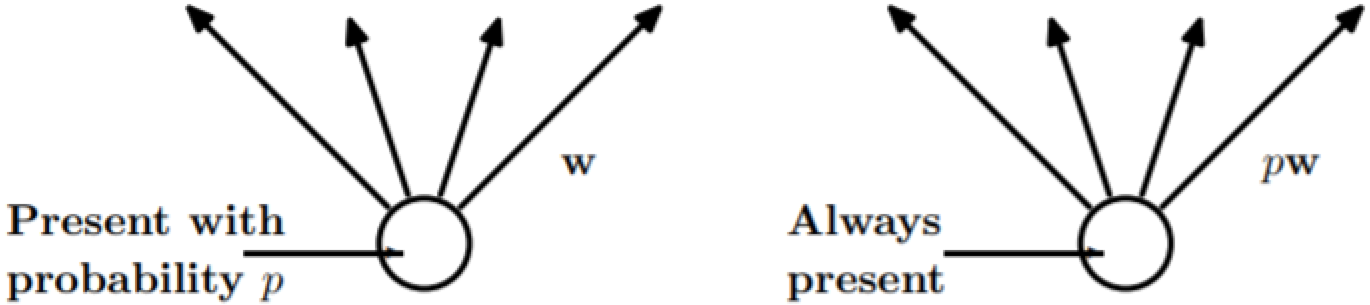


(a) Standard Neural Net (b) After applying dropout.

Inspiration: Combining smaller network together has better performance than one large network

A network with n neurons and p=0.5, 2^n possible thinned network

Generally p is set to 0.5, while for the input unit, the optimal p should be closer to 1.



(a) At training time (b) At test time

How to prevent overfitting

- Batch Normalization [3]

Input: Values of x over a mini-batch: $\mathcal{B} = \{x_{1\ldots m}\}$;
Parameters to be learned: γ, β

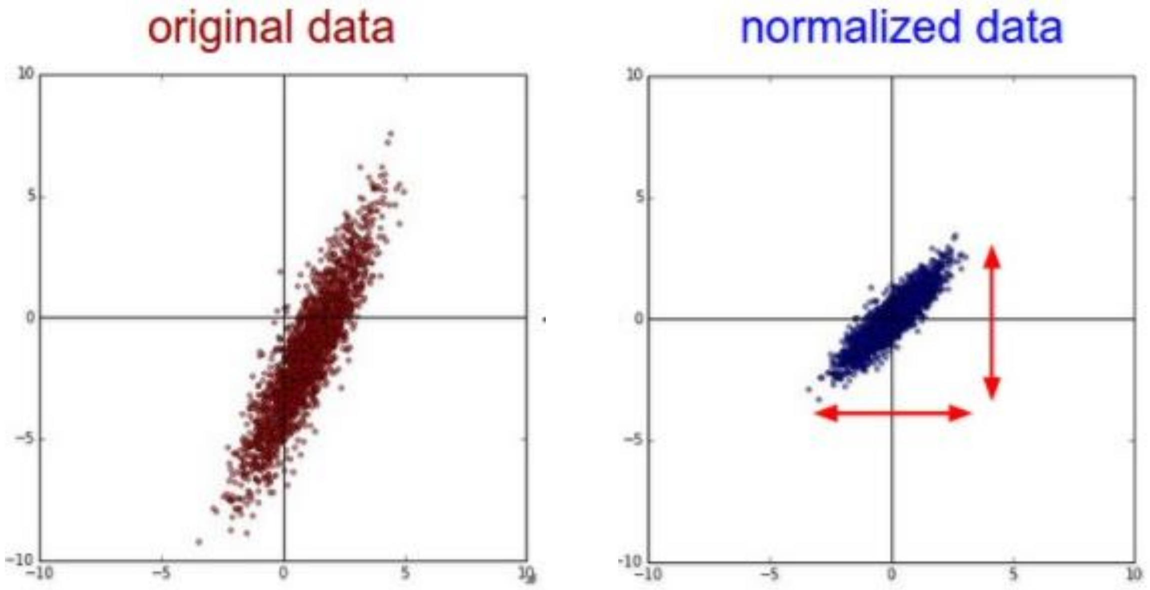
Output: $\{y_i = \mathrm{BN}_{\gamma,\beta}(x_i)\}$

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \qquad \text{// mini-batch mean}$$

$$\sigma_{\mathcal{B}}^2 \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_{\mathcal{B}})^2 \qquad \text{// mini-batch variance}$$

$$\widehat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} \qquad \text{// normalize}$$

$$y_i \leftarrow \gamma\widehat{x}_i + \beta \equiv \mathrm{BN}_{\gamma,\beta}(x_i) \qquad \text{// scale and shift}$$



Normalization process is different during training and inference, need to specify in BN Layer.

[3] Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift

Thank you !