Society for Neuroscience 10/21/2019

Visualizing and Understanding Artificial Neural Networks

Anh Nguyen

Assistant Professor

Auburn University

Neuroscience

to understand the brains of __________

humans ✓

cats ✓

mice ✓

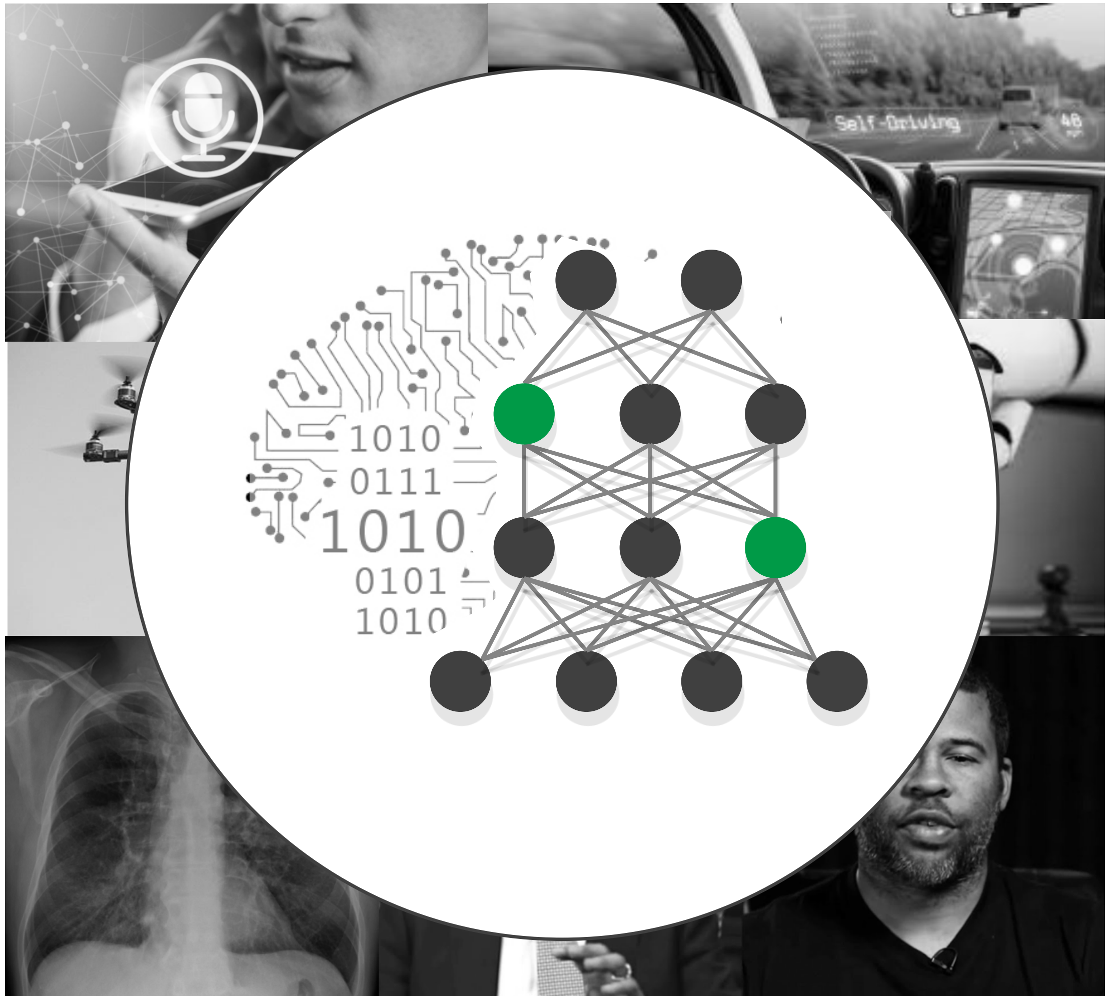
macaques ✓

...

AIs / machines ✓

Neuroscience

Using **AIs** to understand the brains of __________



humans ☑

cats ☑

mice ☑

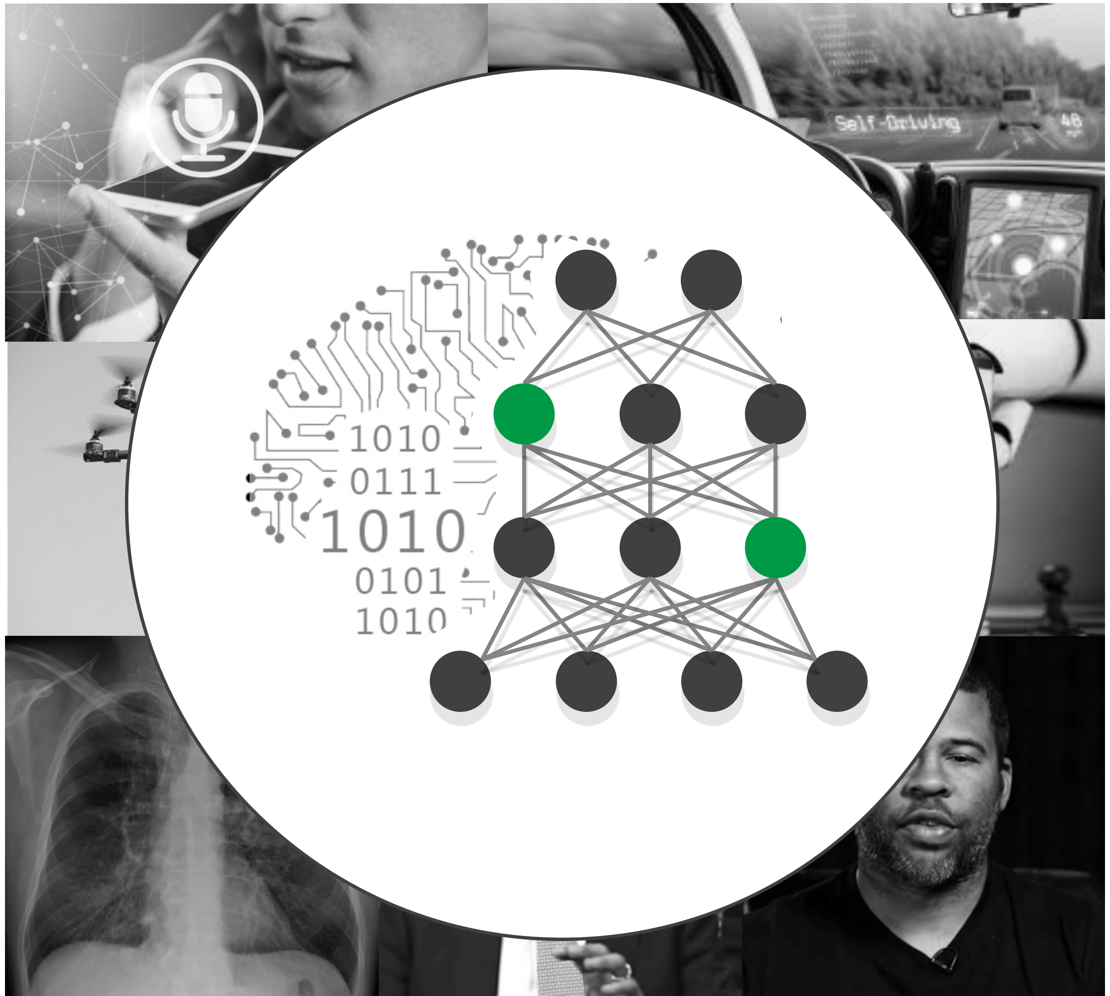
macaques ☑

…

AIs / machines ☑

Neuroscience

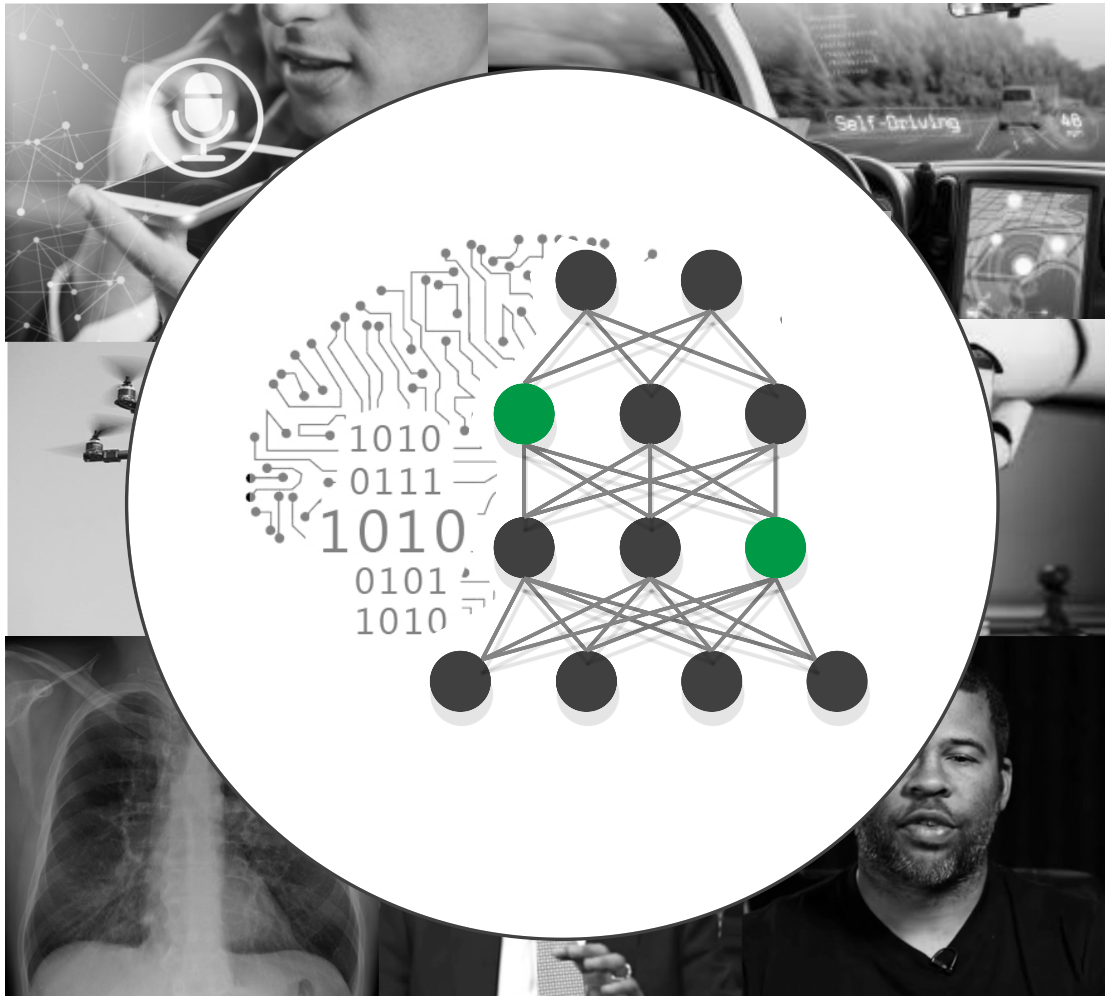
Using **AIs** to understand the brains of __________



humans ☑
cats ☑
mice ☑
macaques ☑
...
AIs / machines ☑

~~Neuroscience~~ "AI neuroscience"

Using AIs to understand the brains of __________



humans ☑

cats ☑

mice ☑

macaques ☑

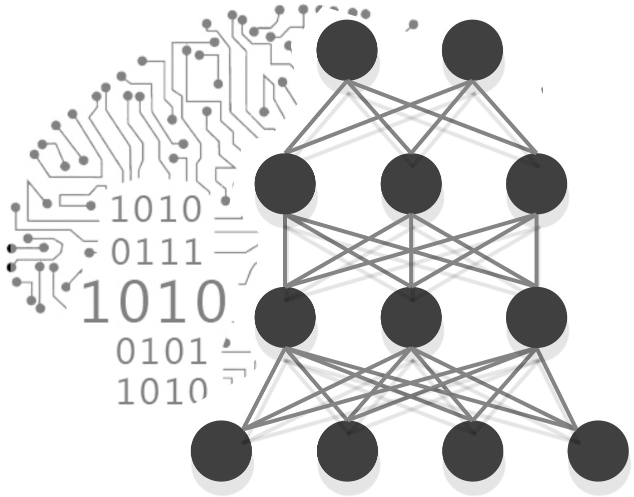
...

AIs / machines ☑

Subject: Image classifier

IMAGENET

cats
mice
macaques
…
school bus
daisy



AlexNet (Krizhevsky et al. 2012)

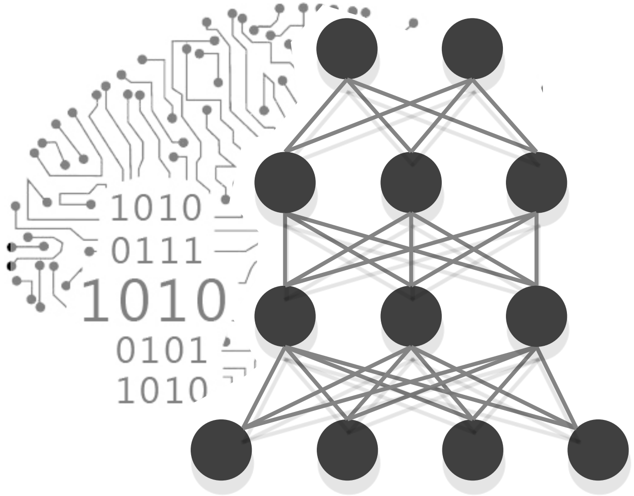
Subject: Image classifier

60M connections

500K neurons

5 conv + 5 dense layers

f :



AlexNet (Krizhevsky et al. 2012)

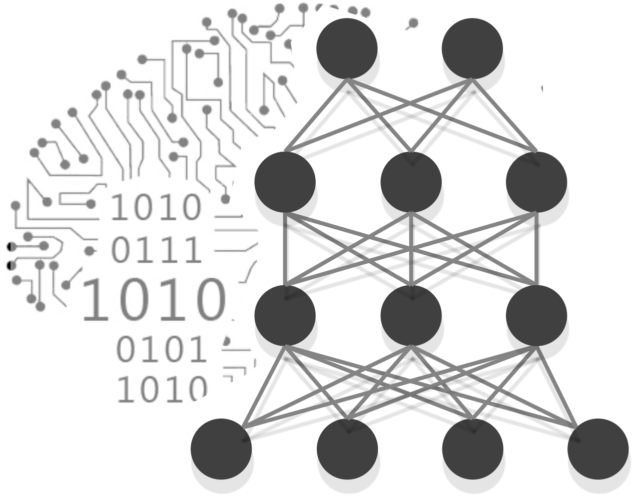
f

IMAGENET

cats	3%
mice	2%
macaques	0%
…	…
school bus	92%
daisy	1%

Subject: Image classifier

f:



AlexNet (Krizhevsky et al. 2012)

f

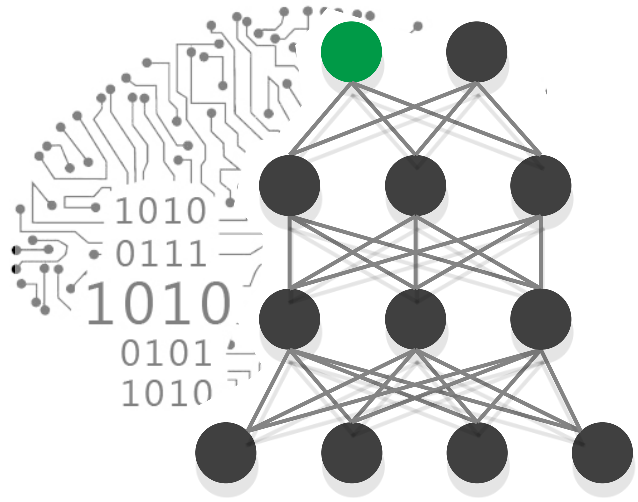
IMAGENET

cats	1%
mice	0%
macaques	2%
…	…
school bus	95%
daisy	0%

What does the school bus neuron want to see?

school bus

f:



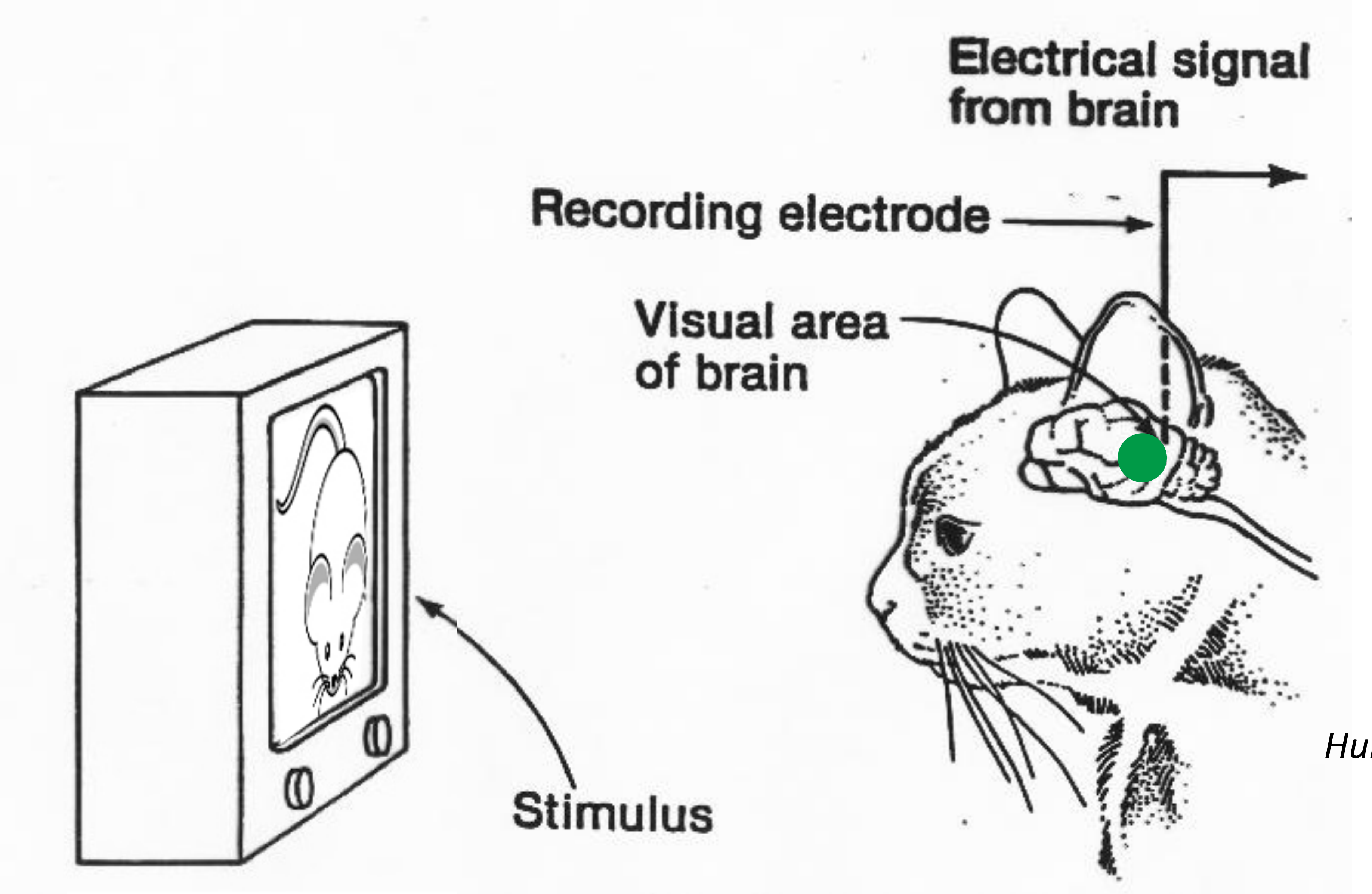
AlexNet (Krizhevsky et al. 2012)

f

IMAGENET

cats	1%
mice	0%
macaques	2%
…	…
school bus	95%
daisy	1%

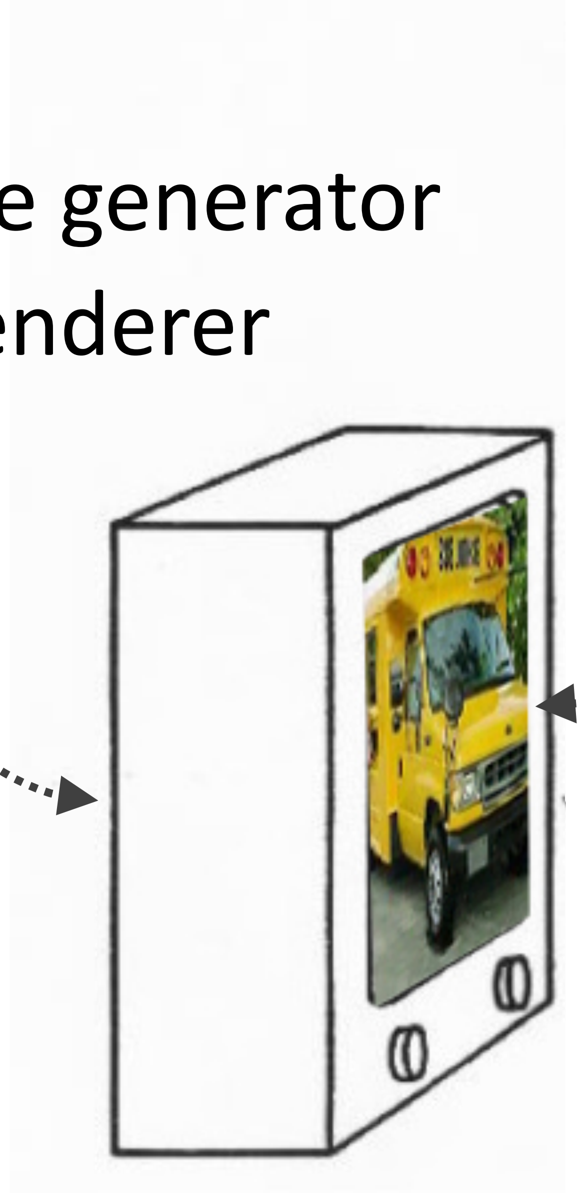
Finding what biological neurons want to see



Hubel & Wiesel
1954

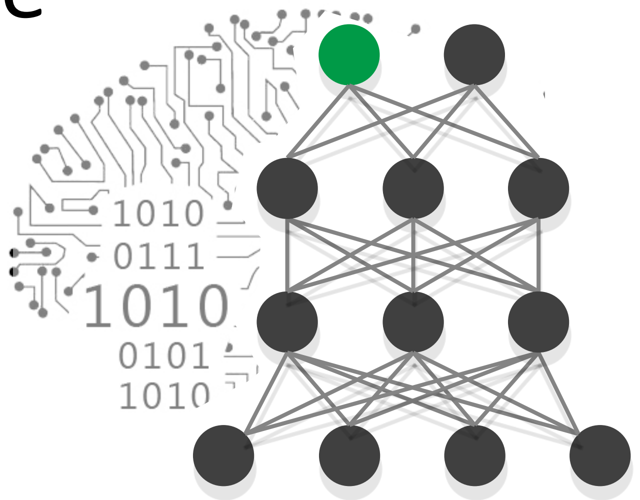
Finding what artificial neurons want to see

2. Image generator
3. 3D renderer



1. Pixel-wise

school bus



$$\boxed{x^* = \arg\max_x(\phi_{layer,idx}(x))}$$

"Activation maximization" Erhan et al. 2009

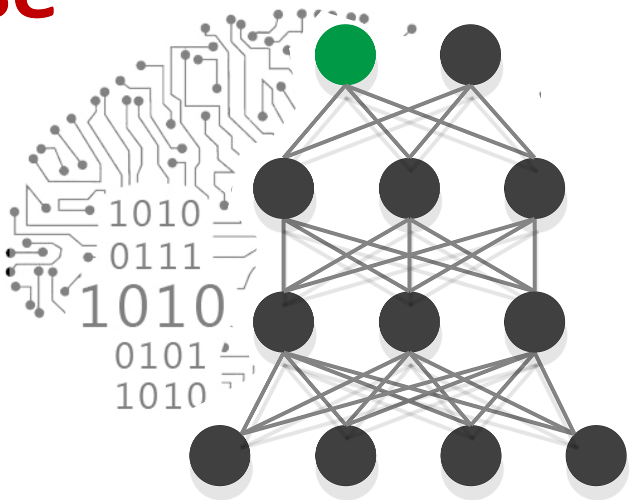
Finding what artificial neurons want to see

2. Image generator

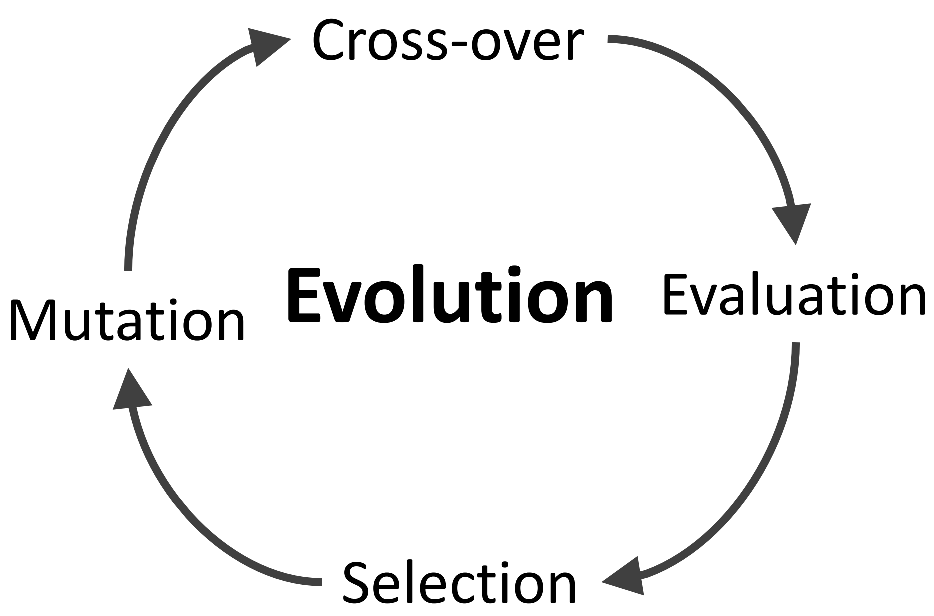
3. 3D renderer

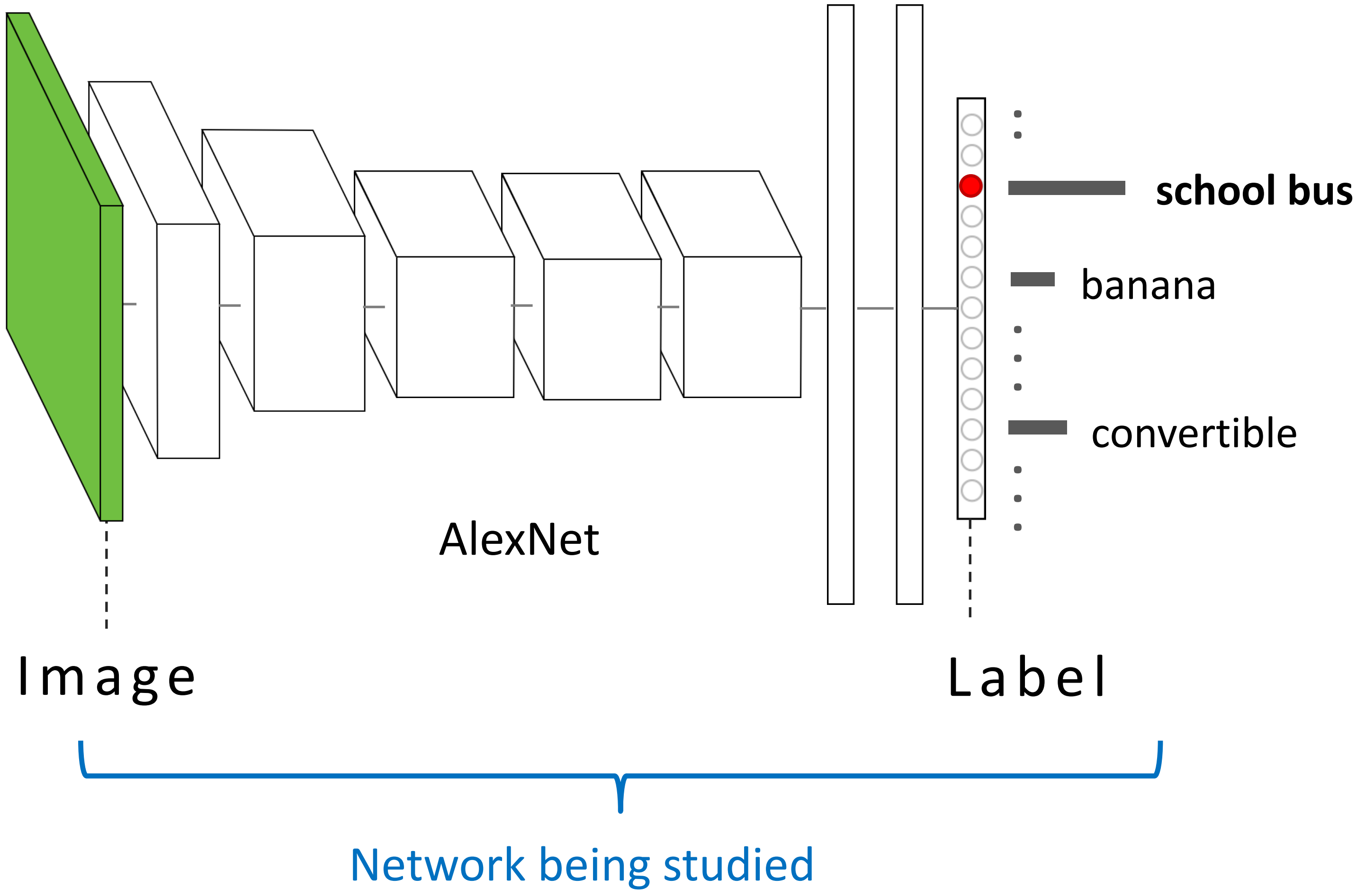
1. Pixel-wise

school bus

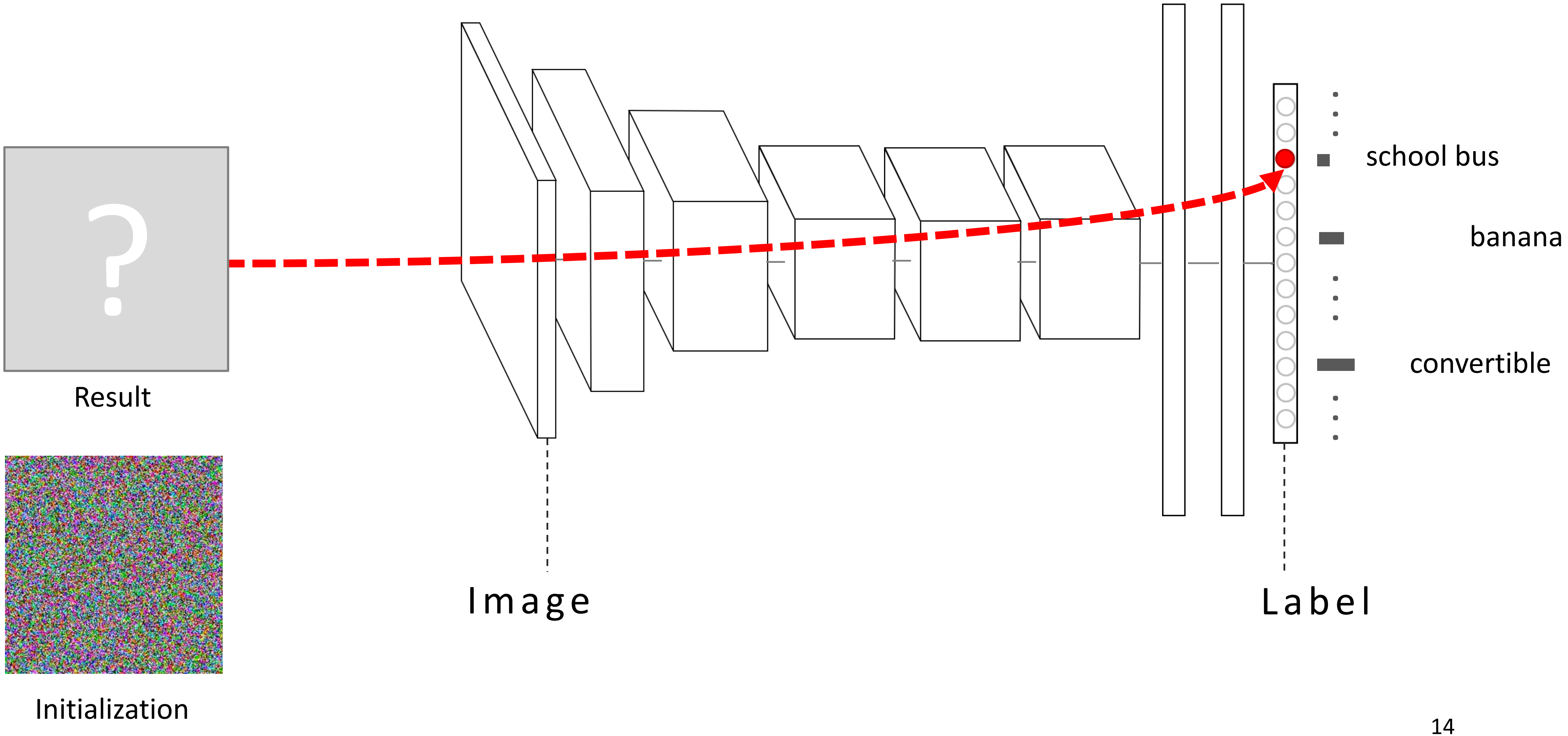


Pixel: Evolutionary algorithms

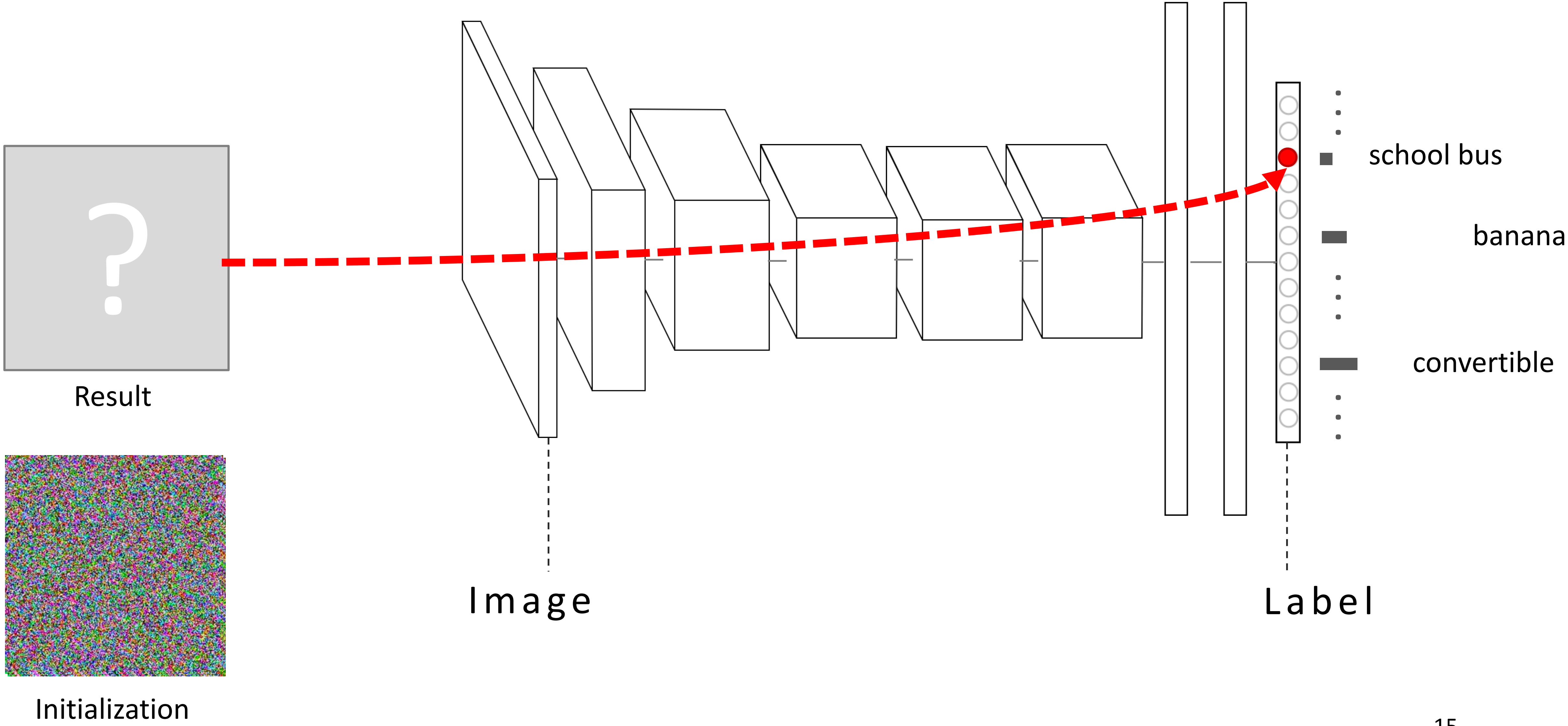




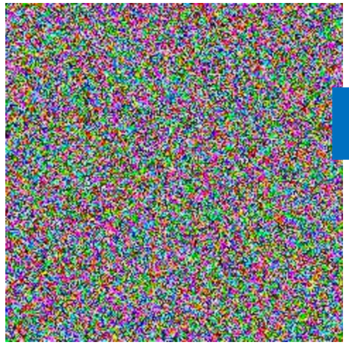
Pixel: Gradient descent



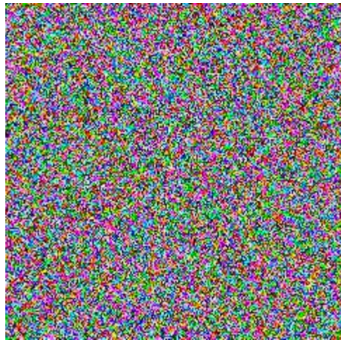
Pixel: Gradient descent



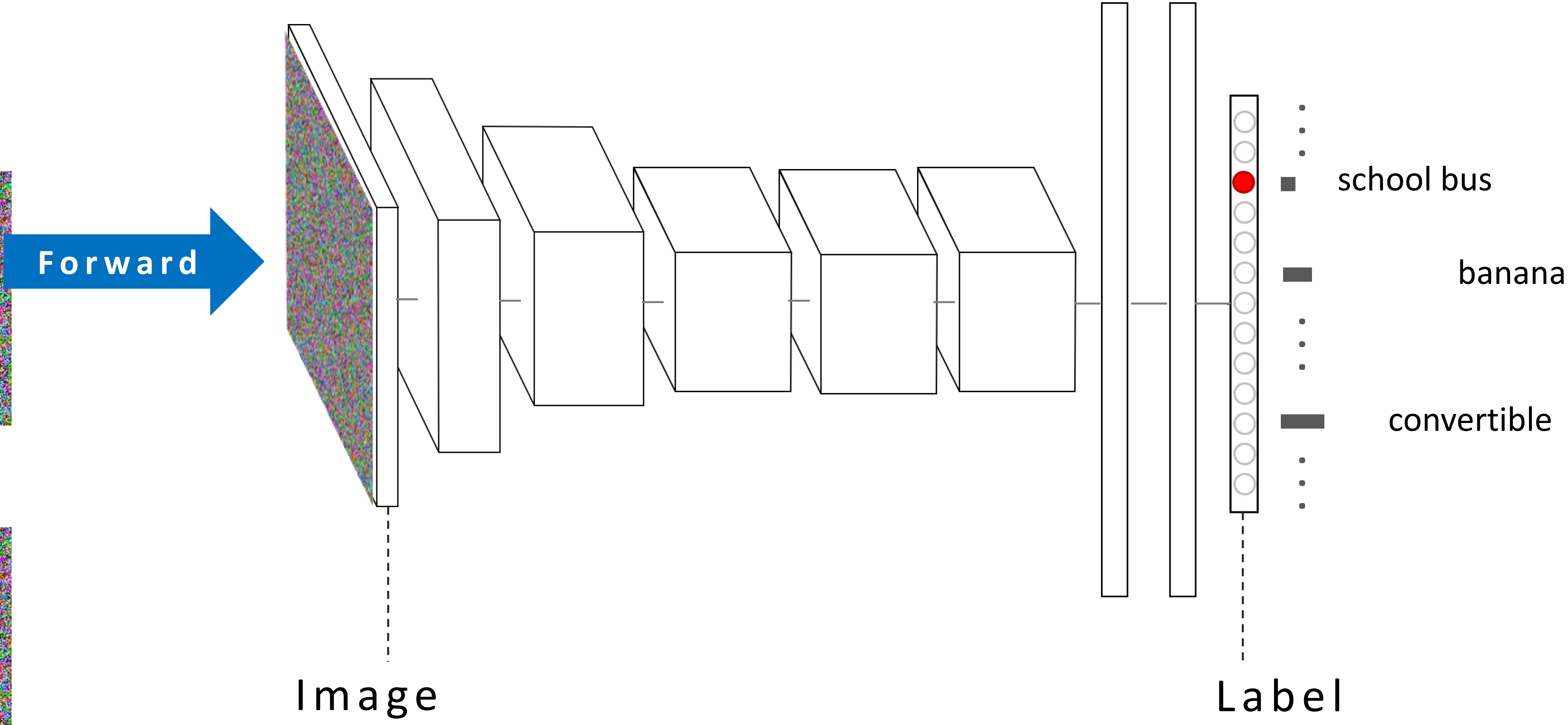
Pixel: Gradient descent



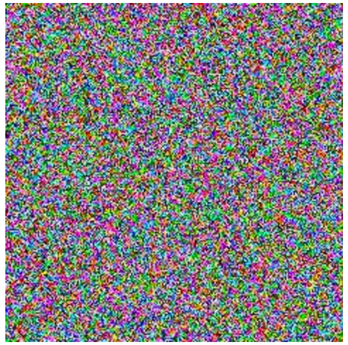
Result



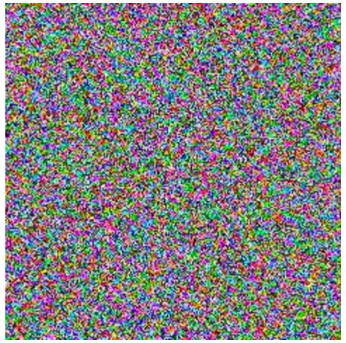
Initialization



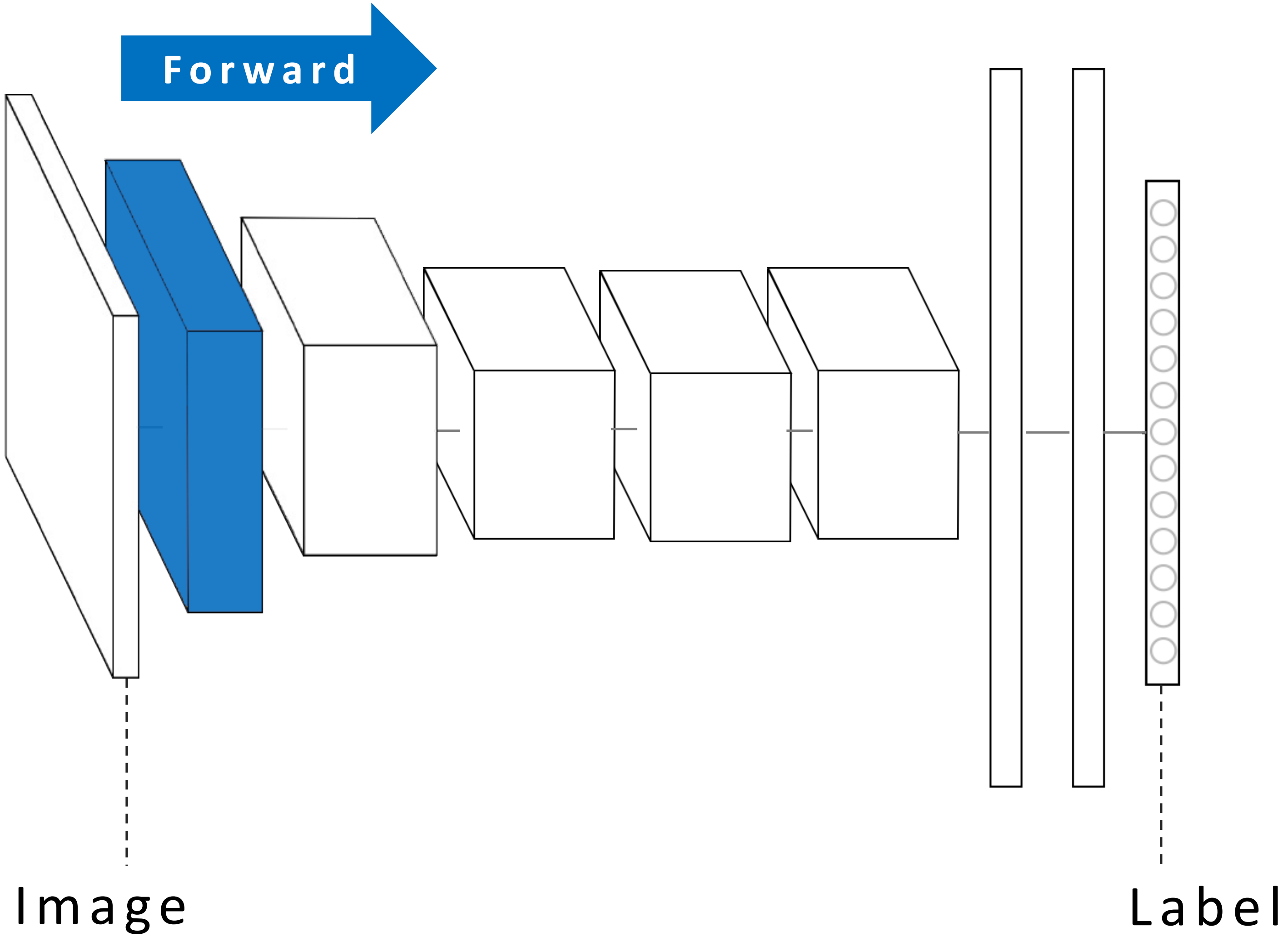
Pixel: Gradient descent



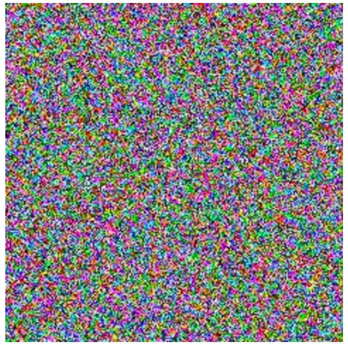
Result



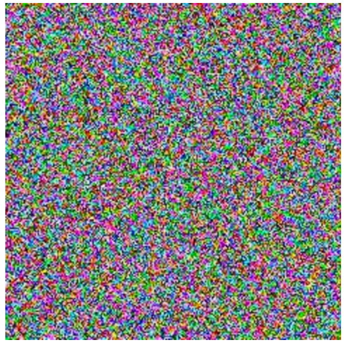
Initialization



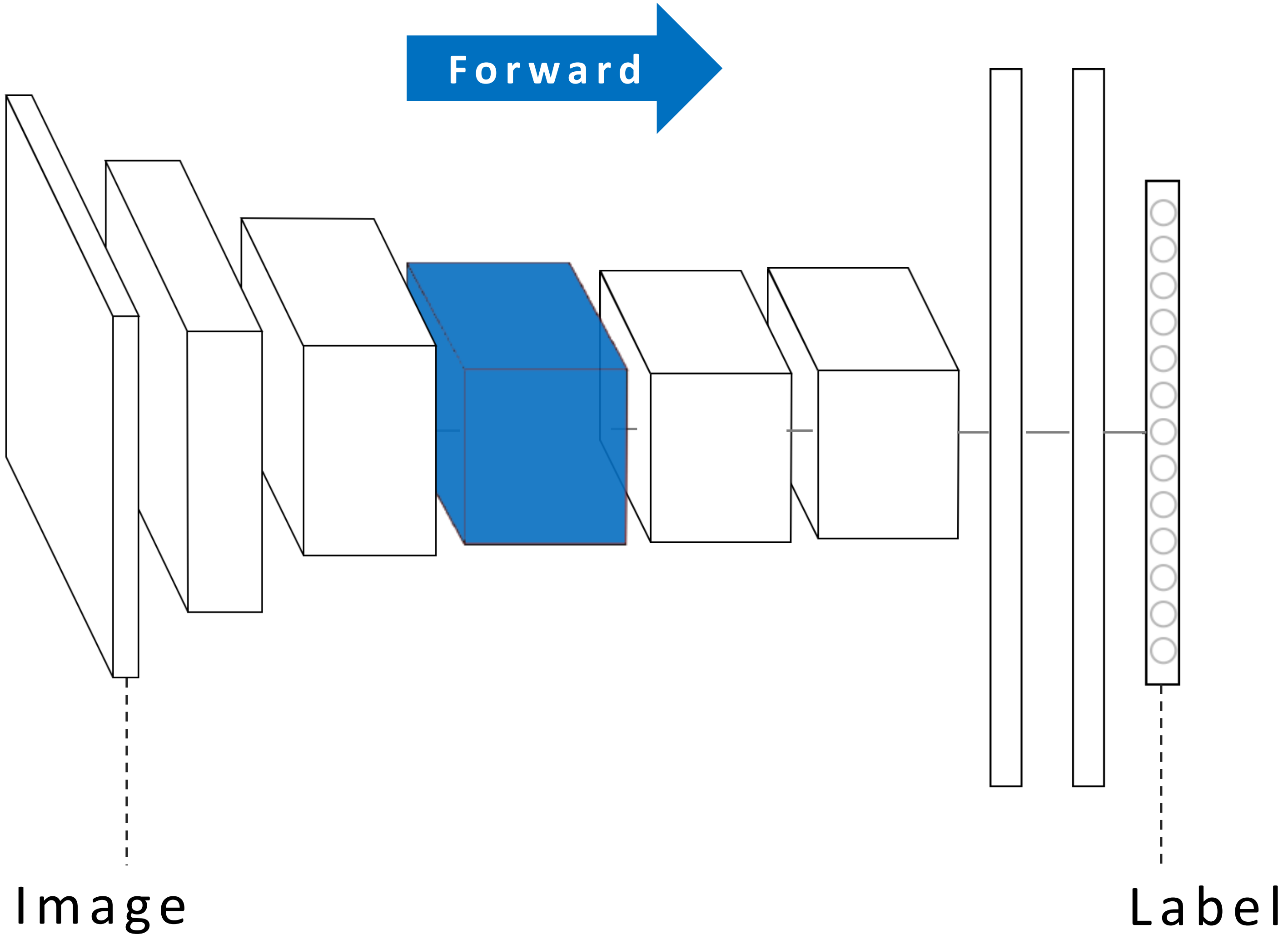
Pixel: Gradient descent



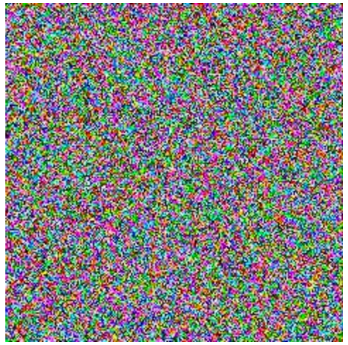
Result



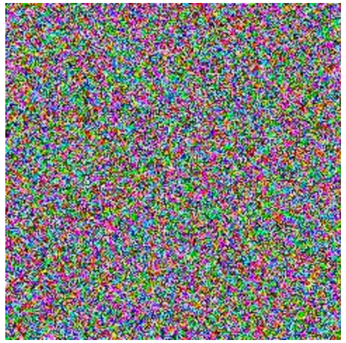
Initialization



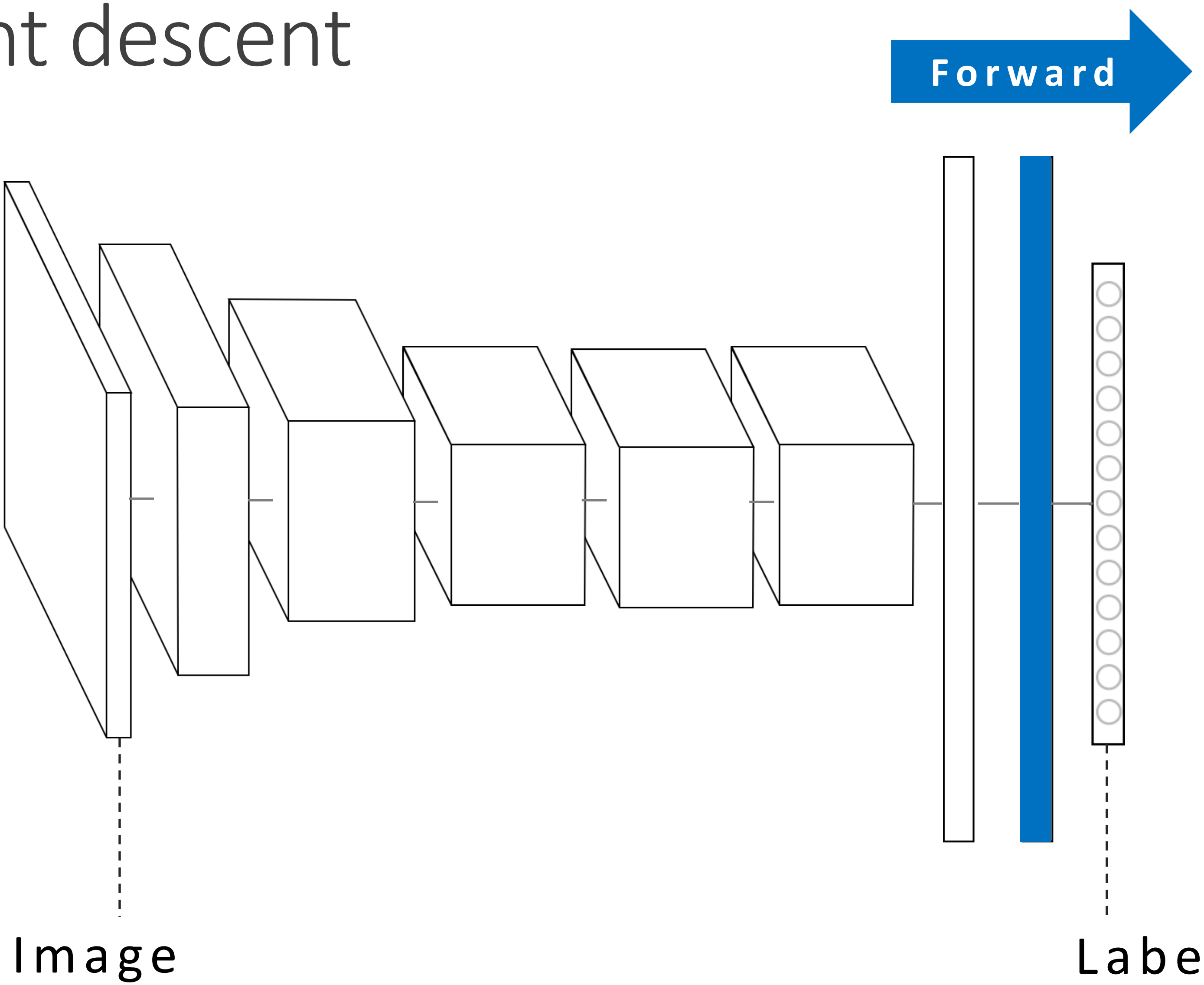
Pixel: Gradient descent



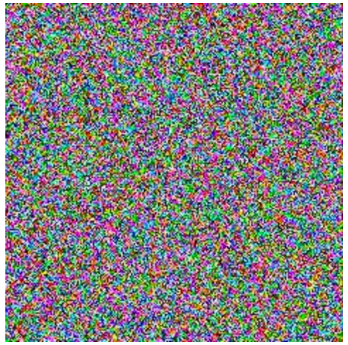
Result



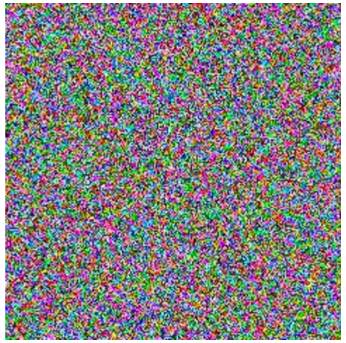
Initialization



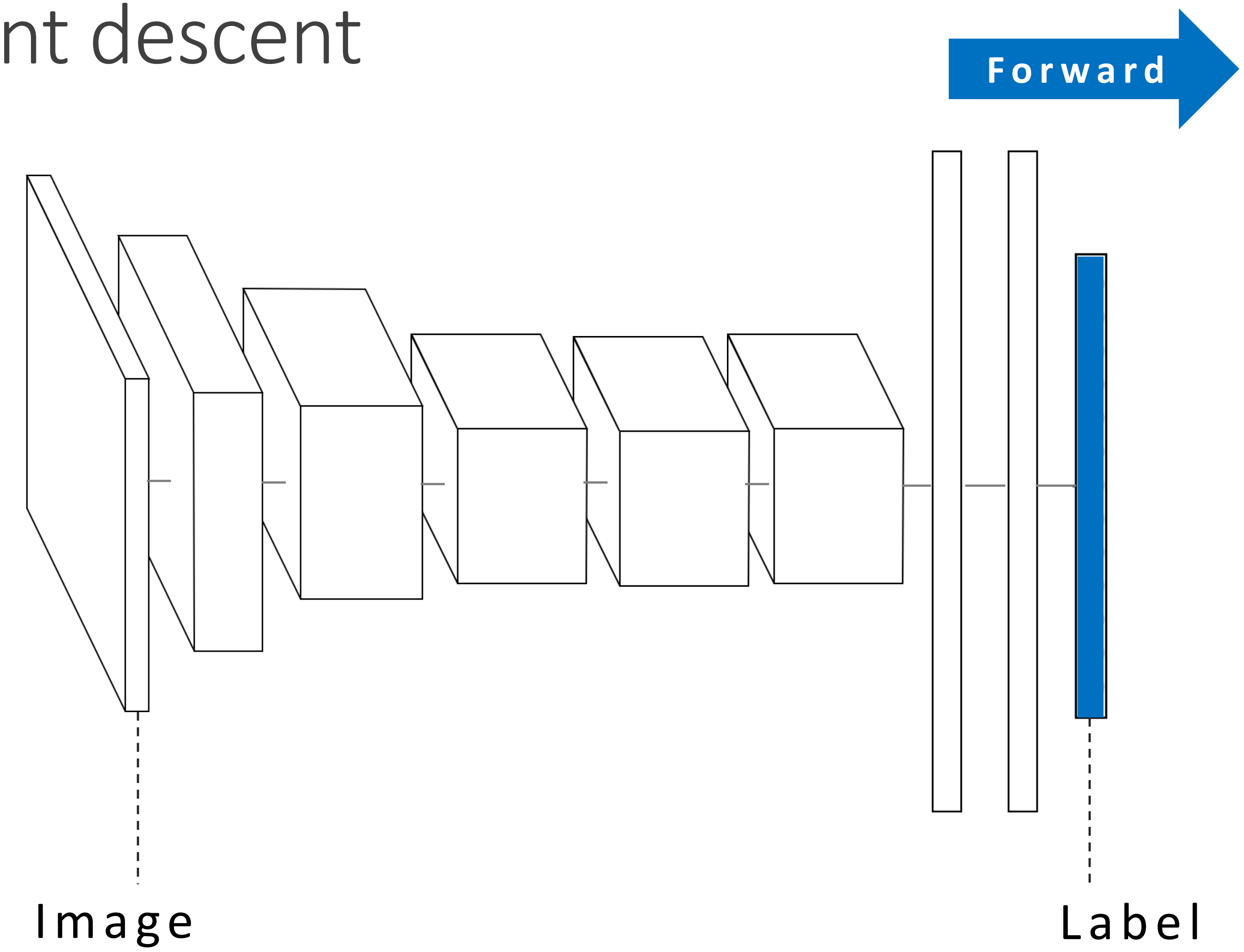
Pixel: Gradient descent



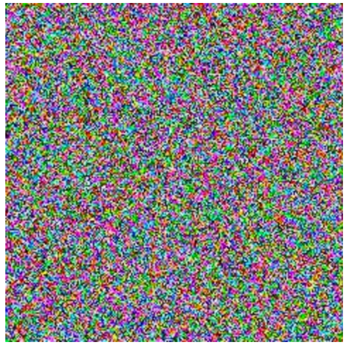
Result



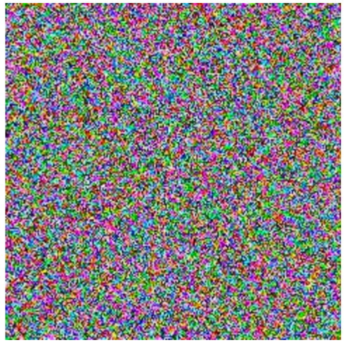
Initialization



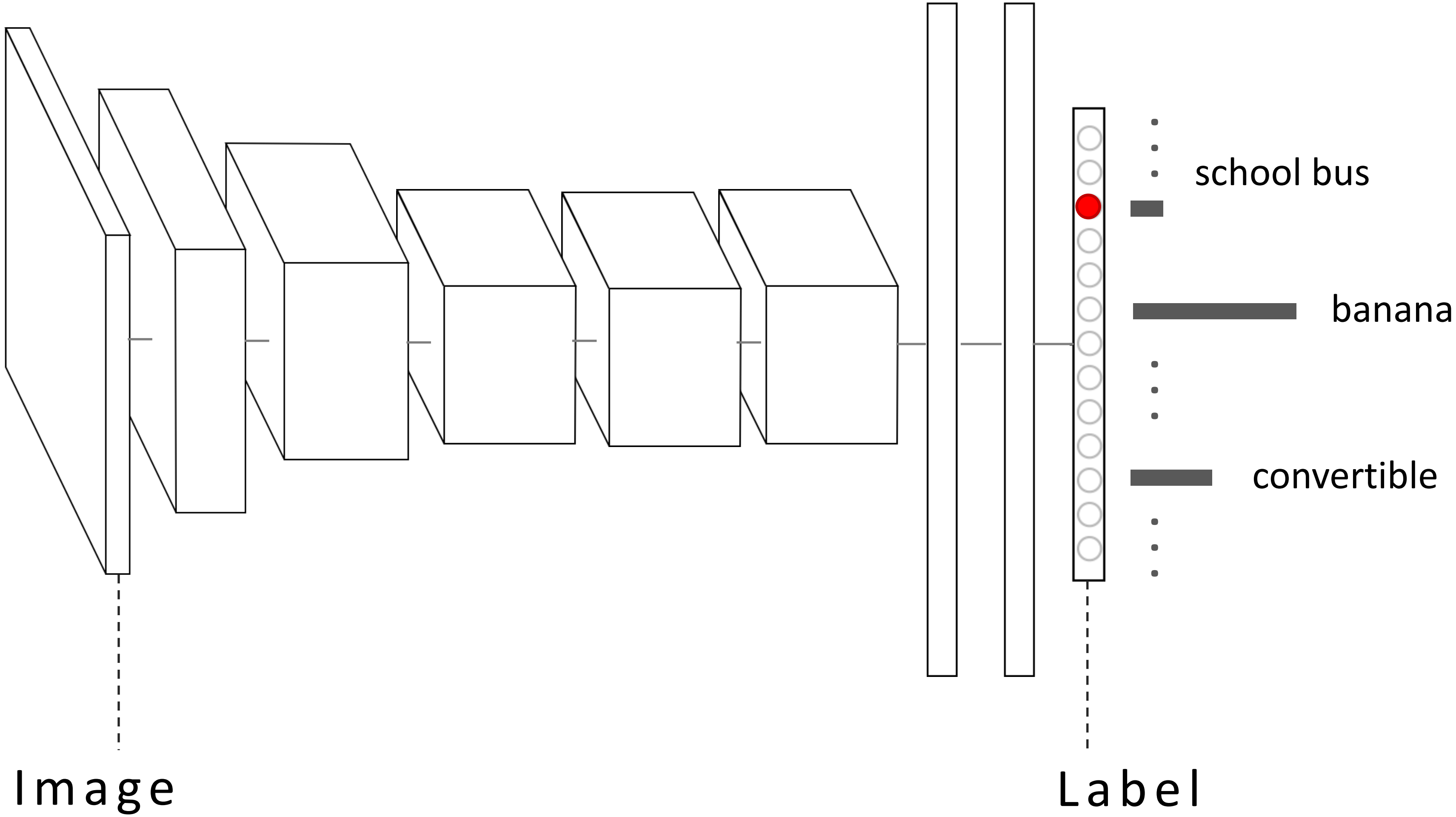
Pixel: Gradient descent



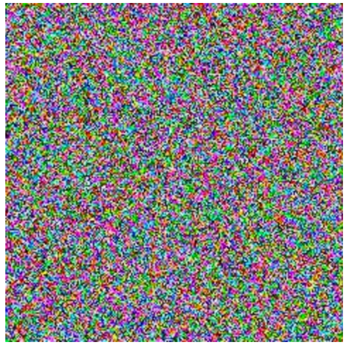
Result



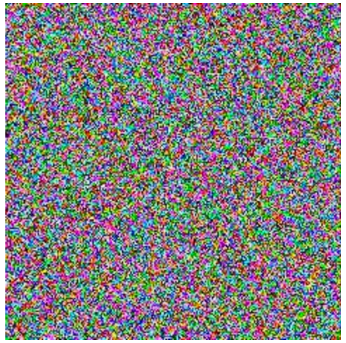
Initialization



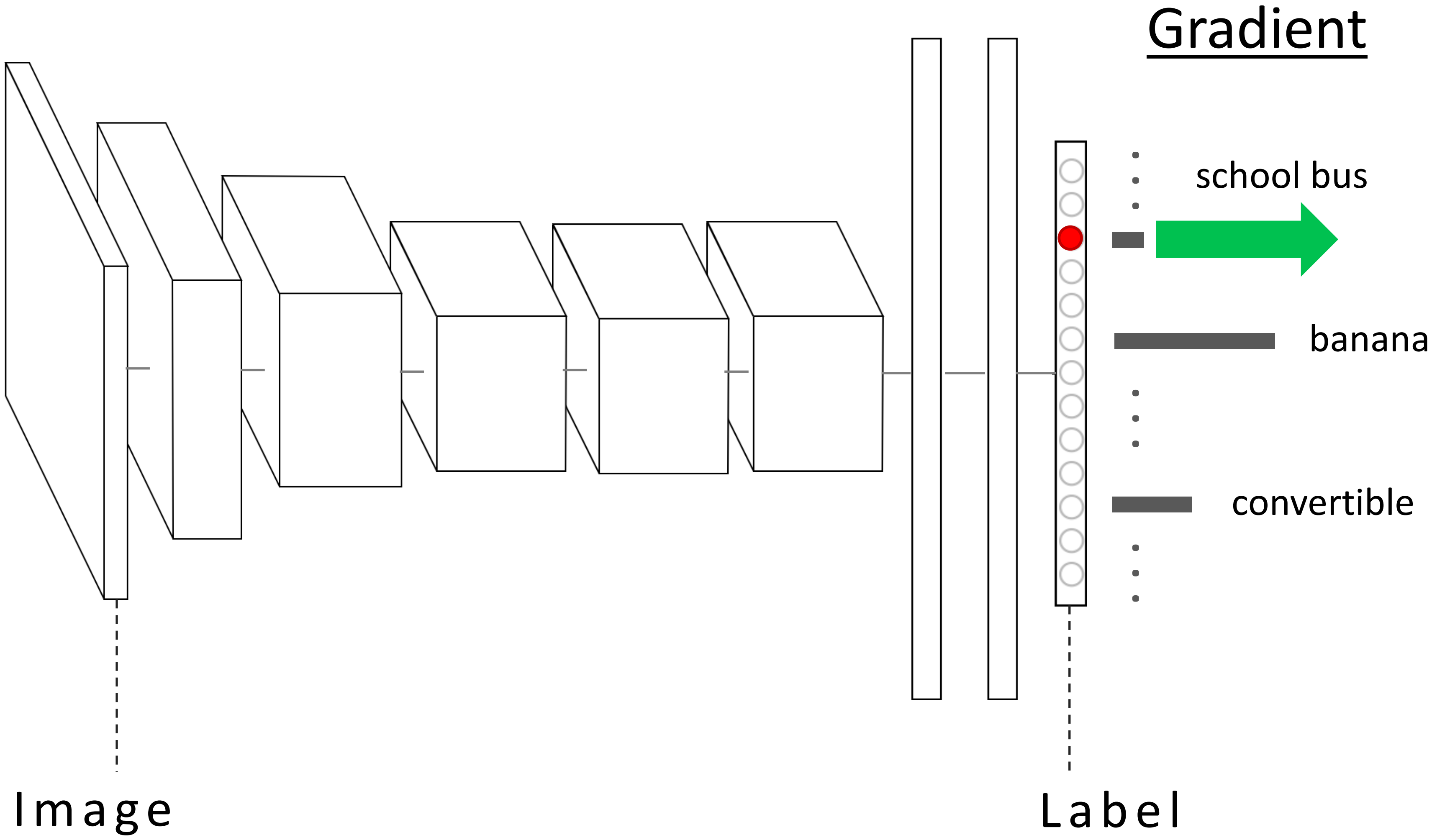
Pixel: Gradient descent



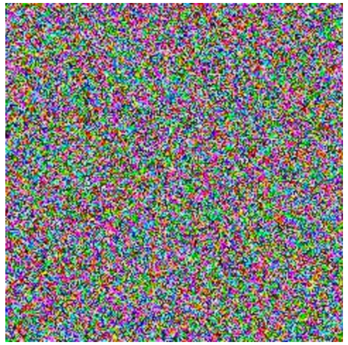
Result



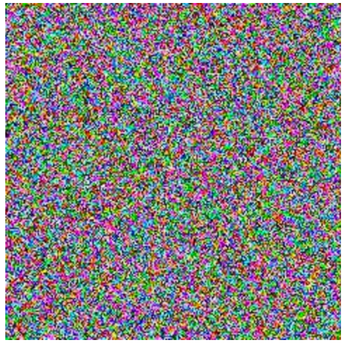
Initialization



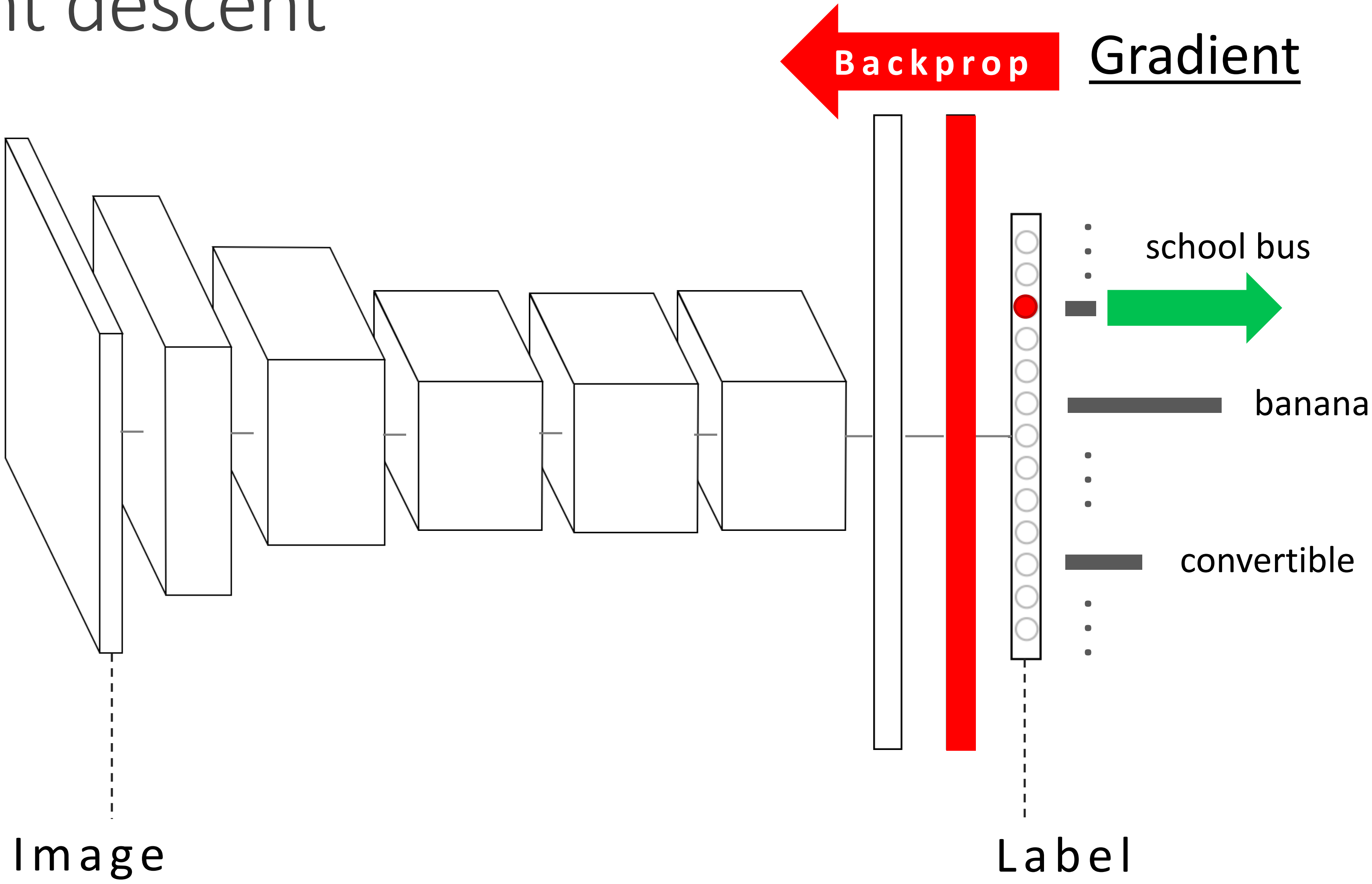
Pixel: Gradient descent



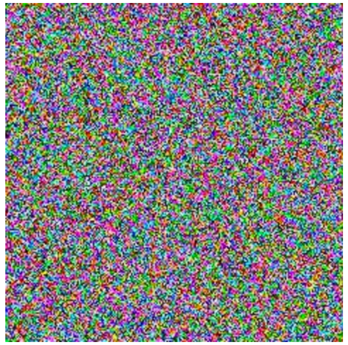
Result



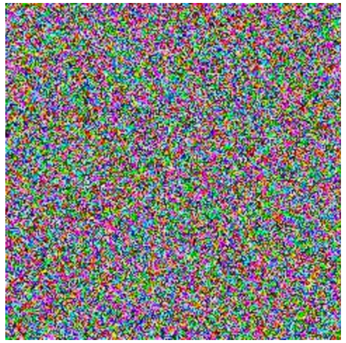
Initialization



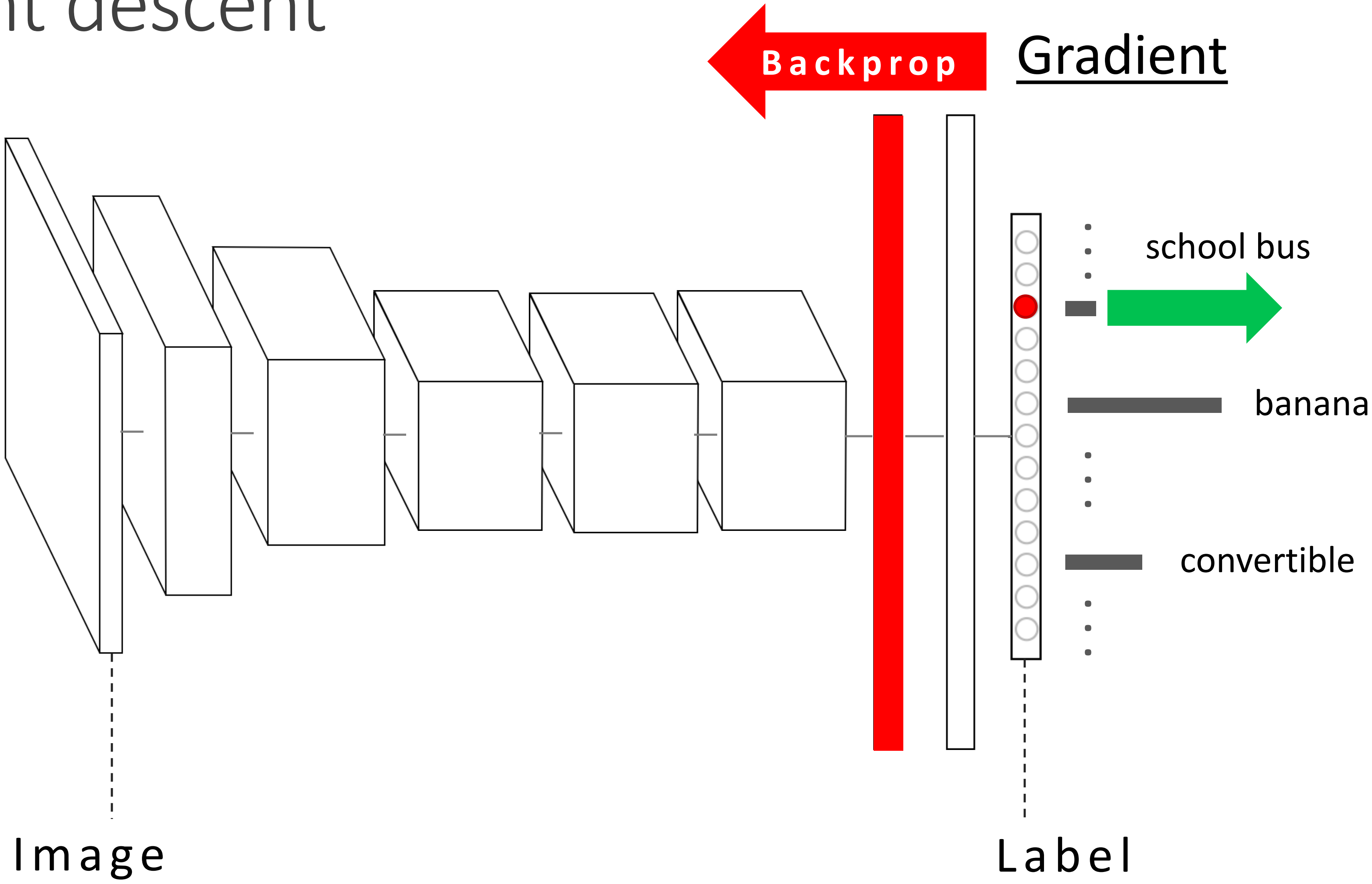
Pixel: Gradient descent



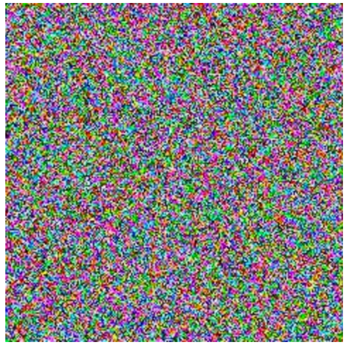
Result

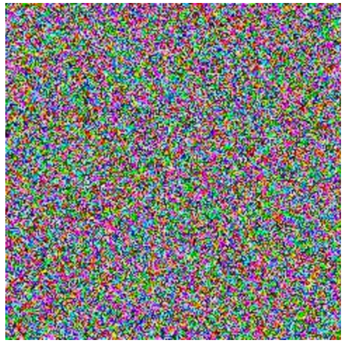


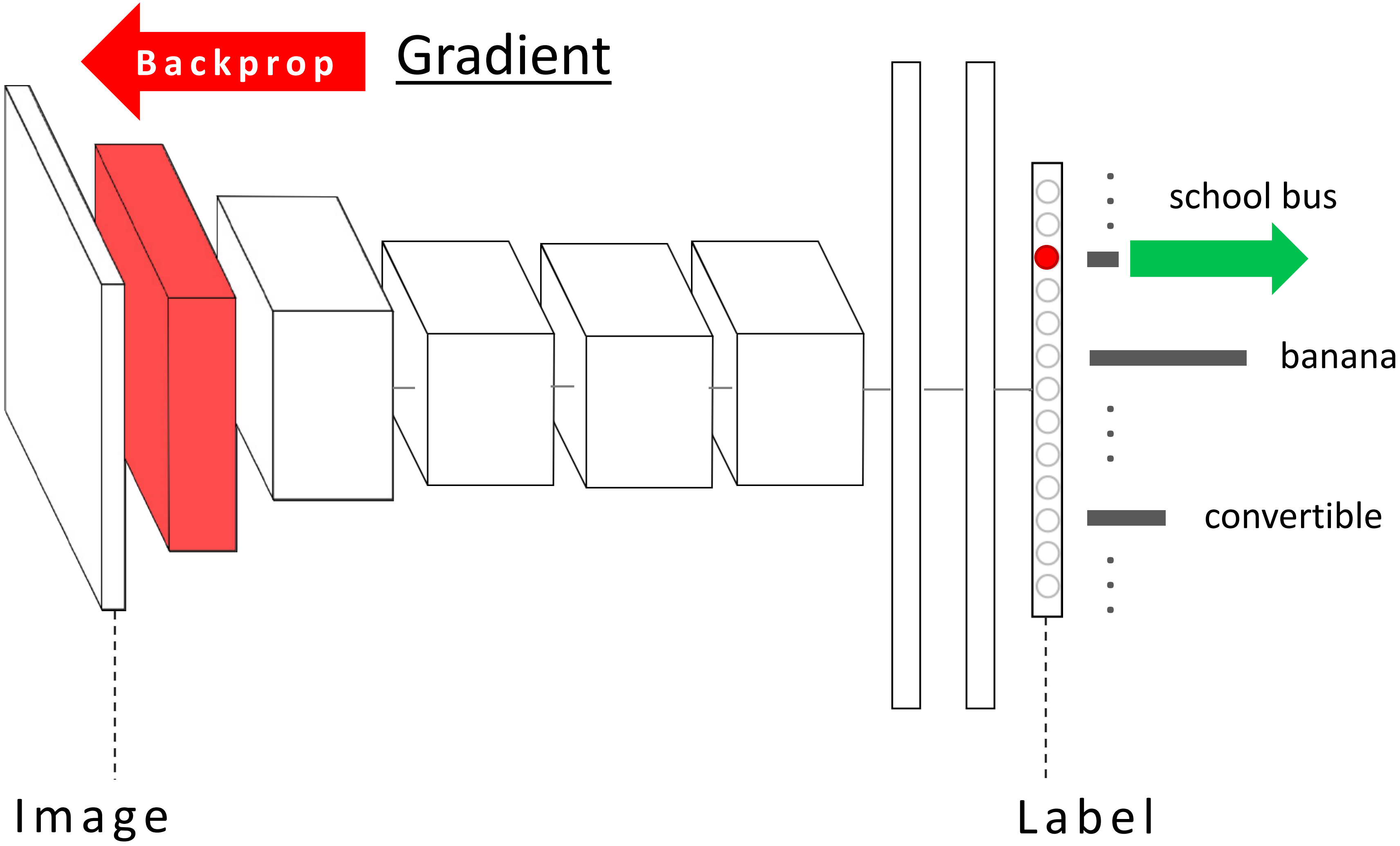
Initialization



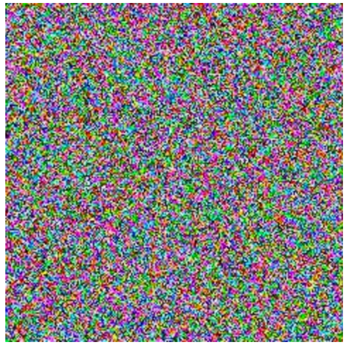
Pixel: Gradient descent

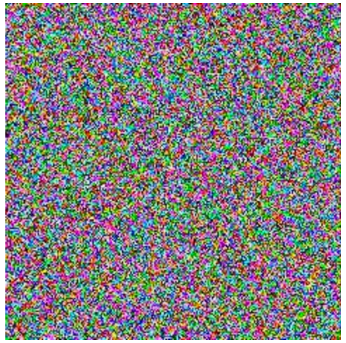

Result

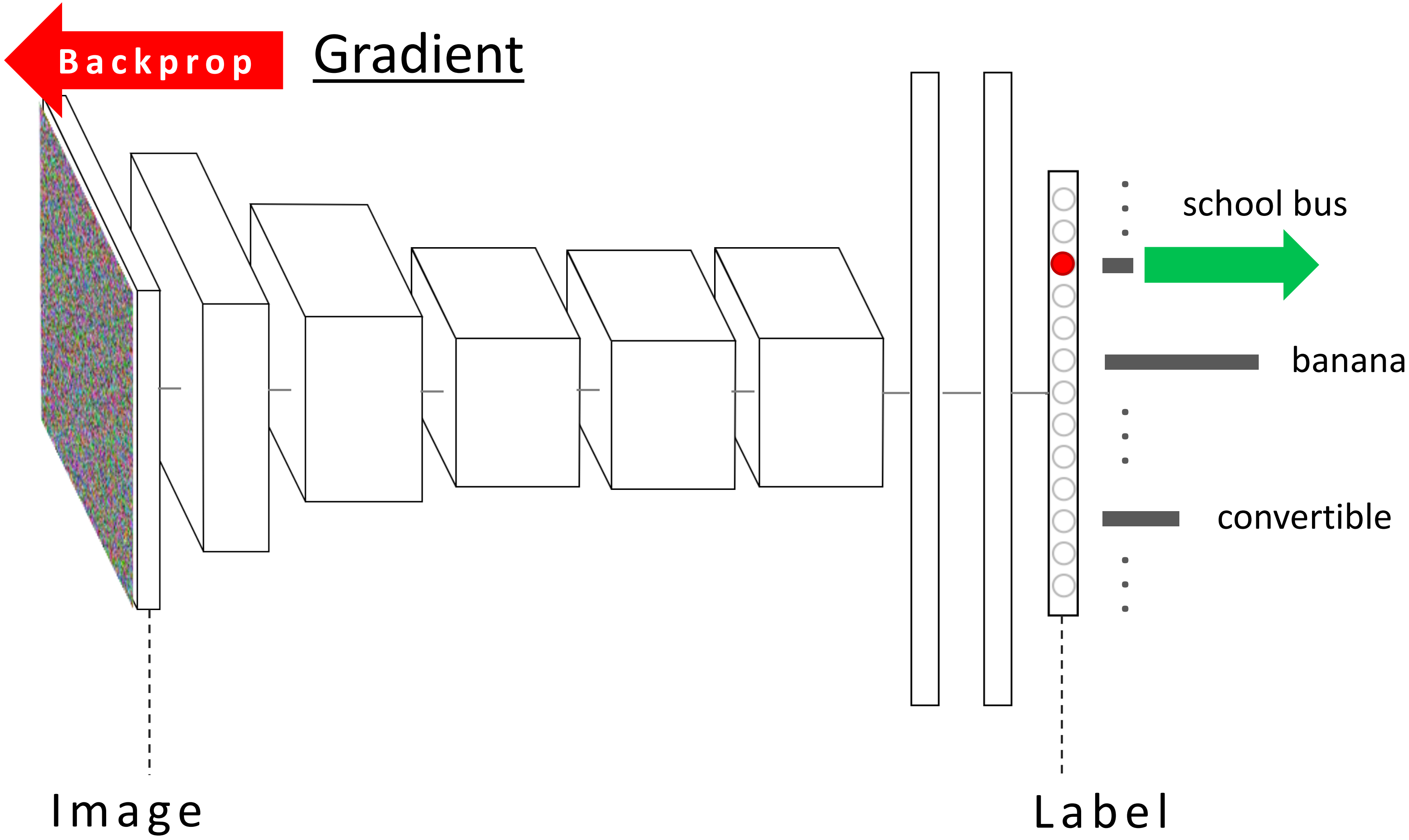

Initialization



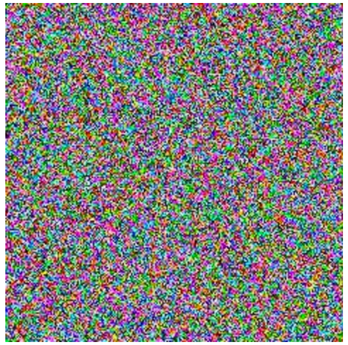
Pixel: Gradient descent


Result

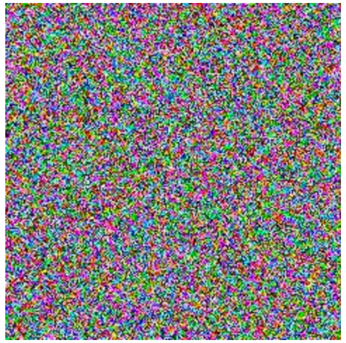

Initialization



Pixel: Gradient descent

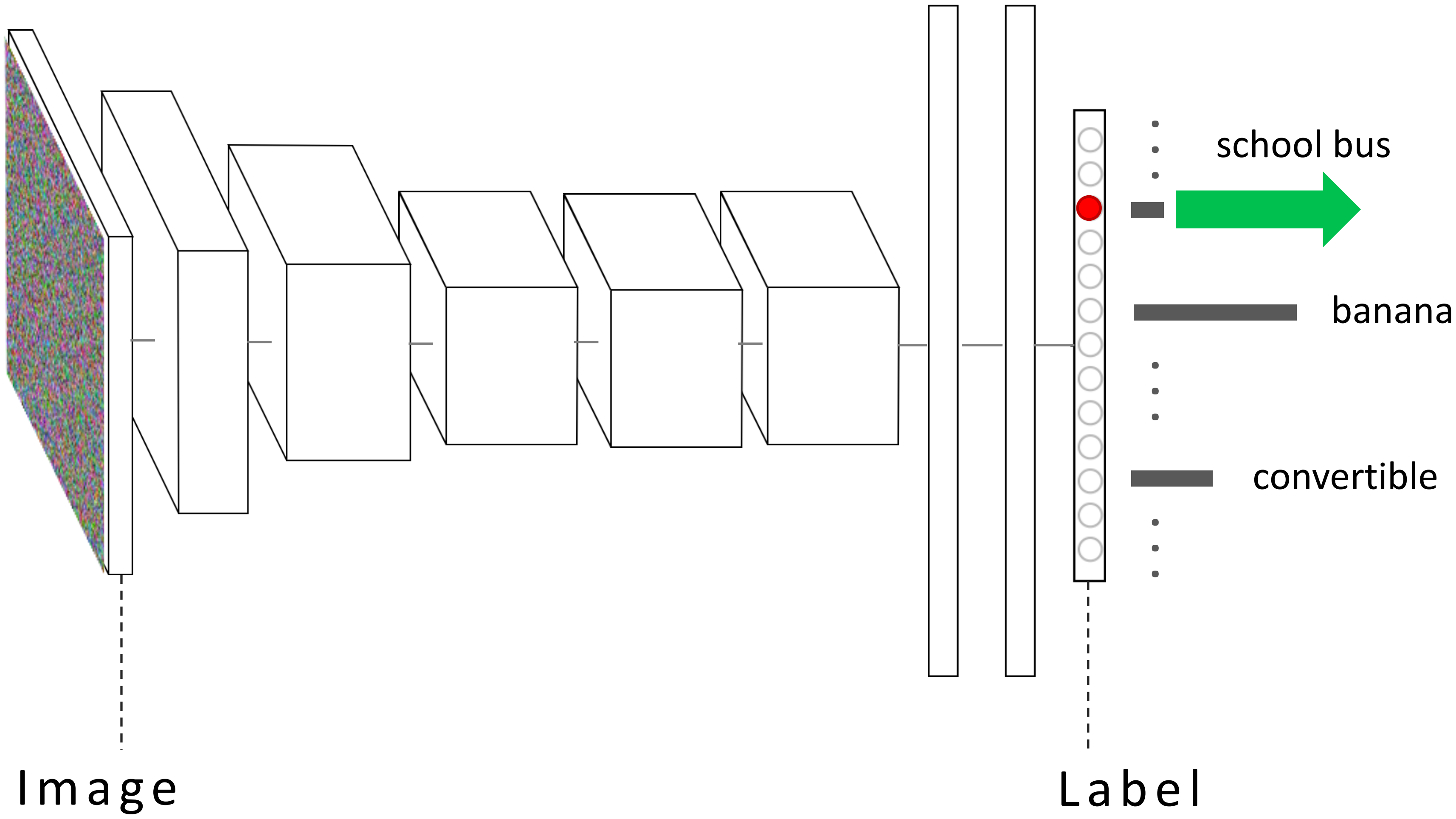


Result

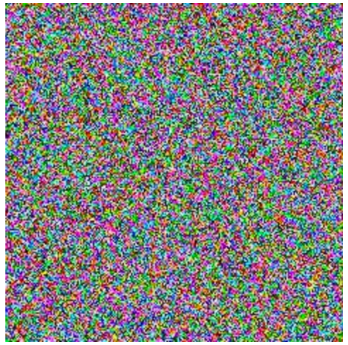


Initialization

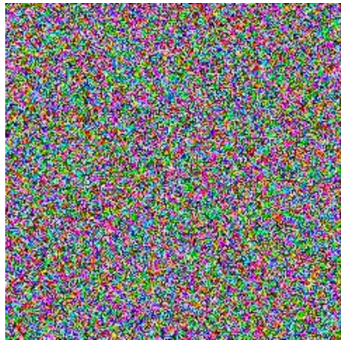
Update image



Pixel: Gradient descent

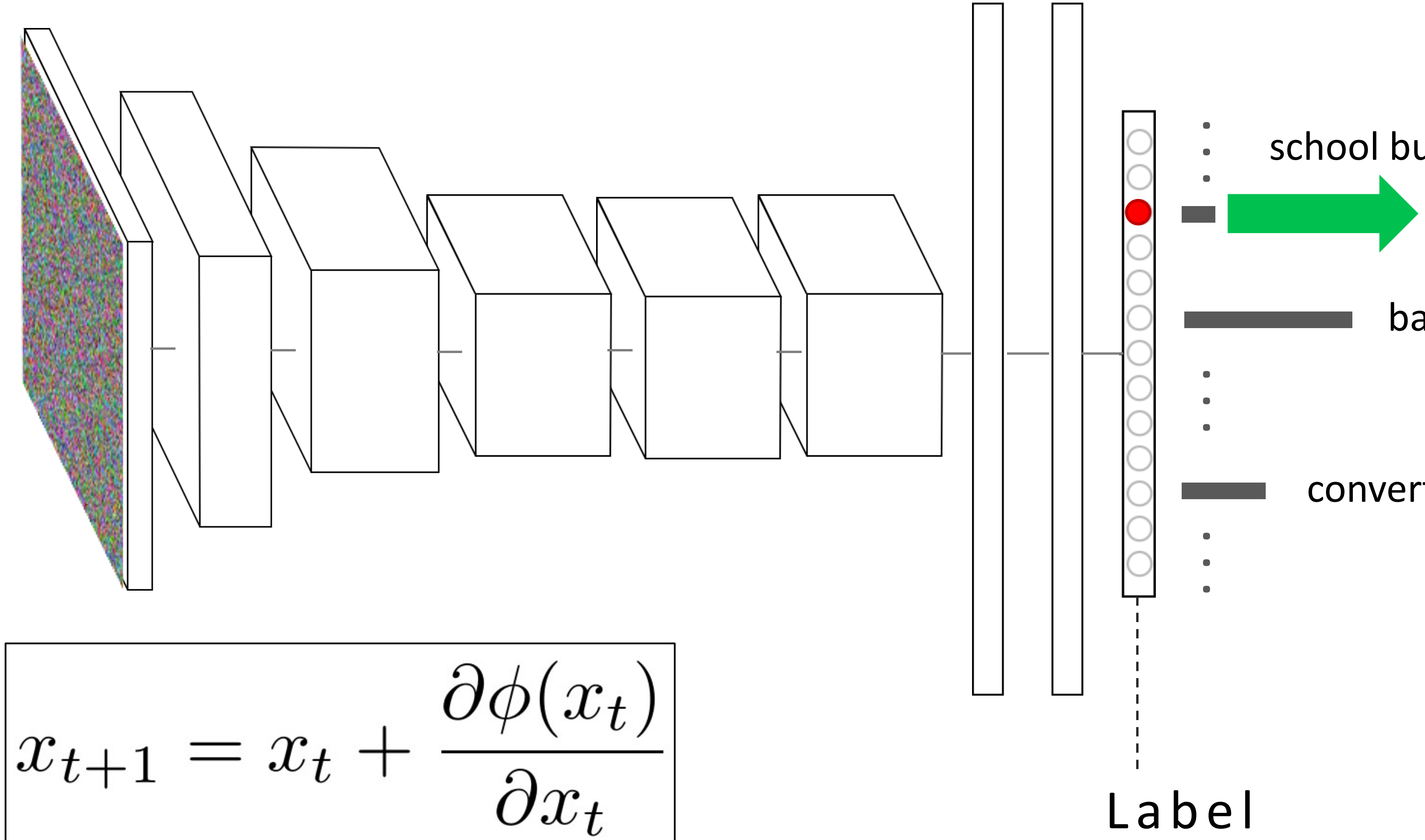


Result

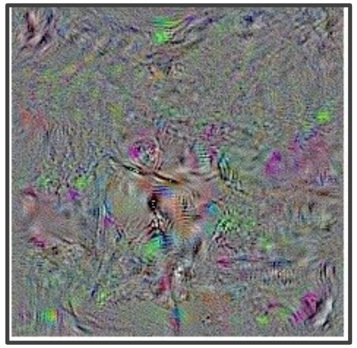


Initialization

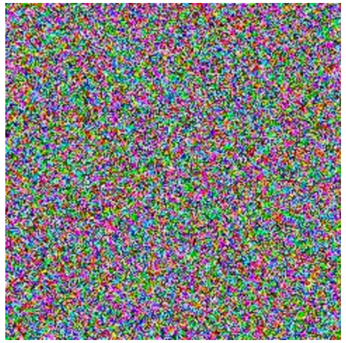
Update image



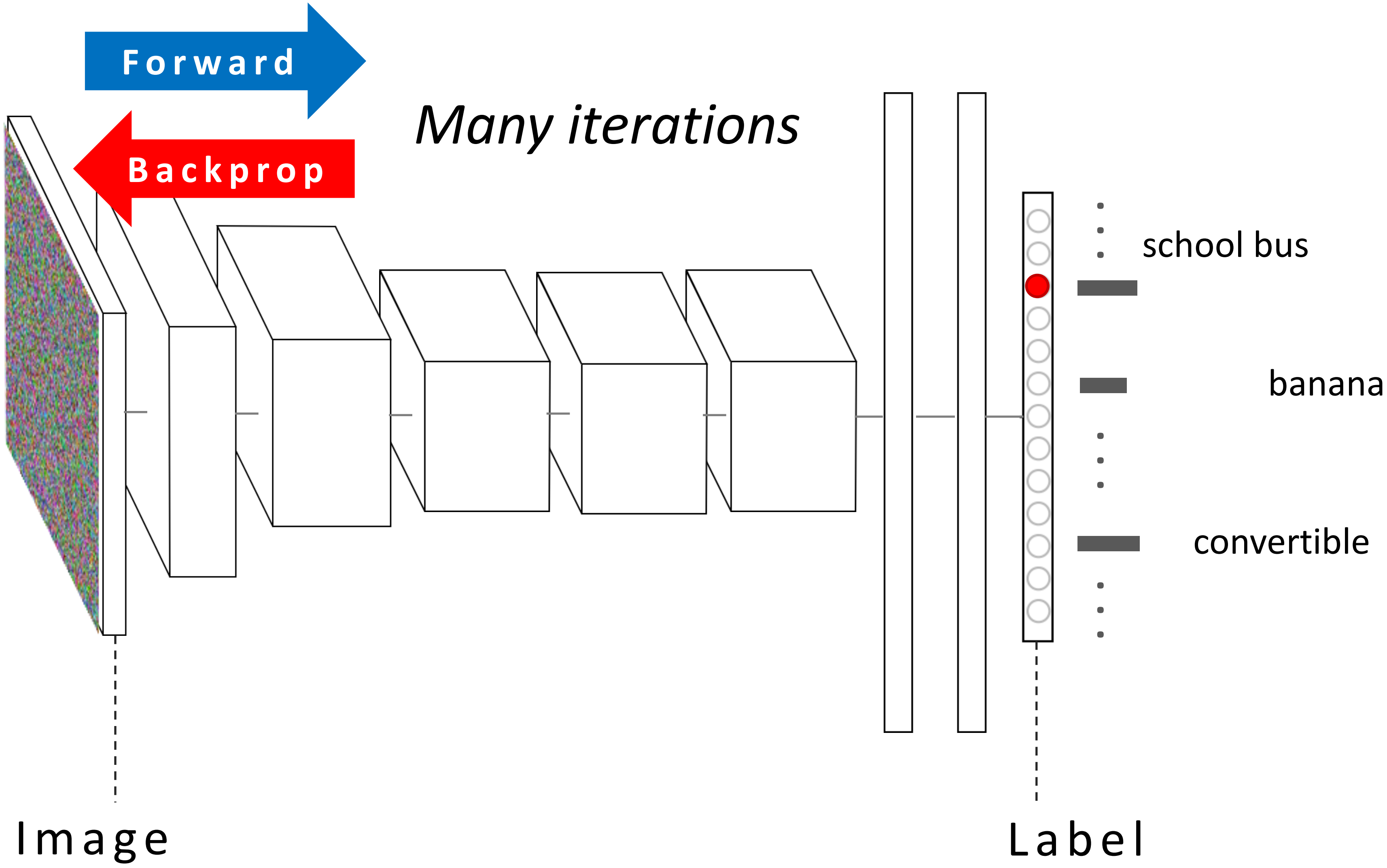
$$x_{t+1} = x_t + \frac{\partial \phi(x_t)}{\partial x_t}$$



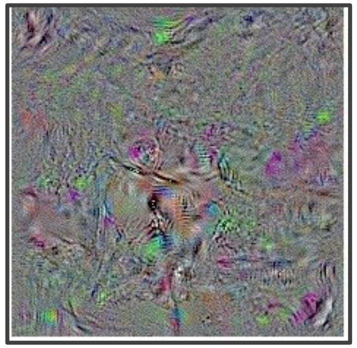
Result



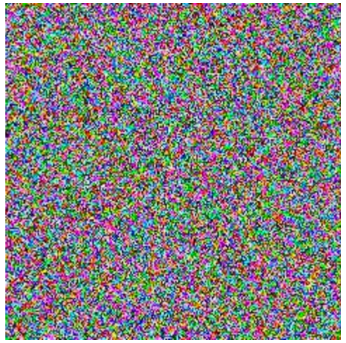
Initialization



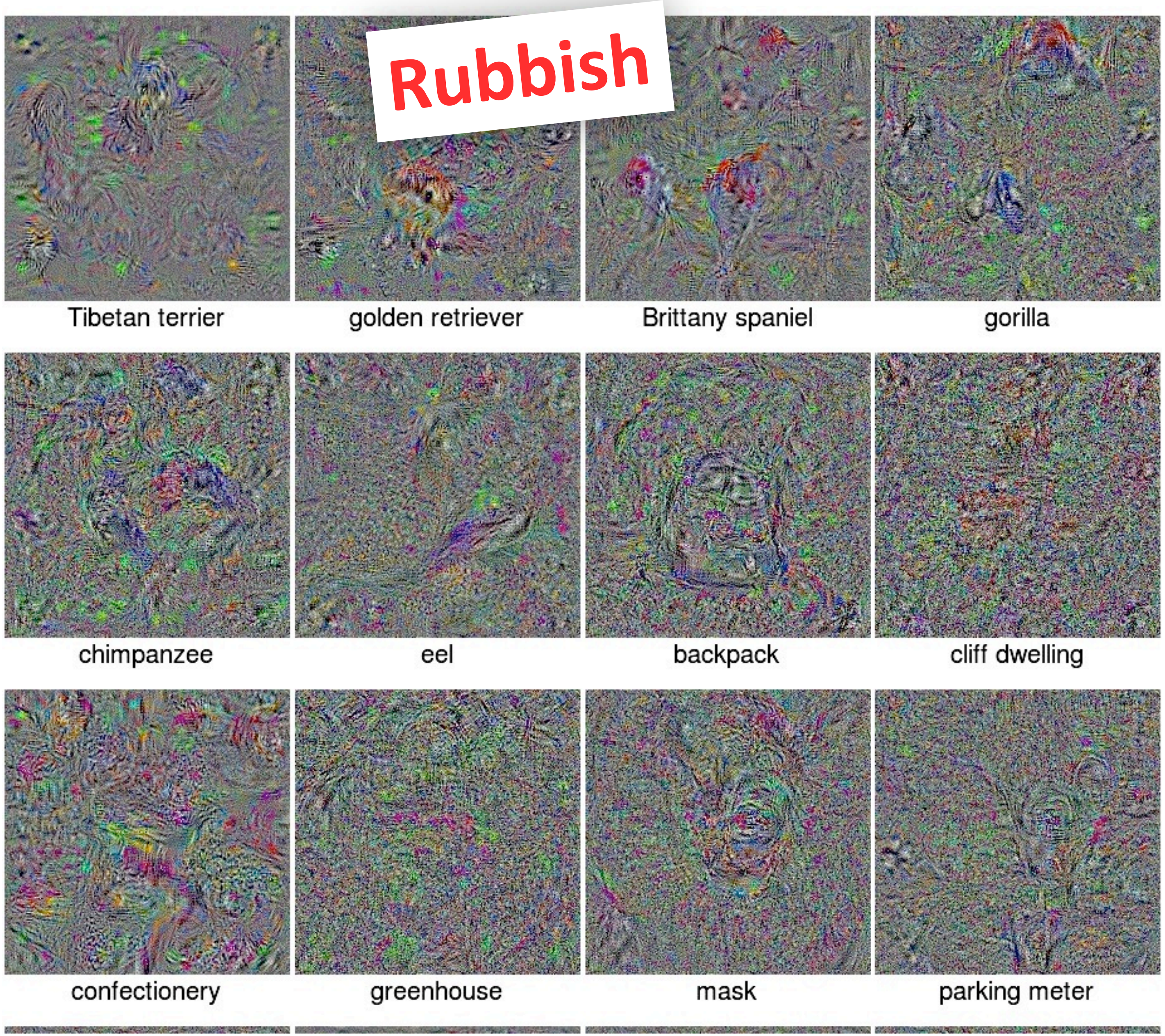
Nguyen et al. 2015 Deep Neural Networks are Easily Fooled



Result



Initialization



Yosinski

Clune

Probabilistic interpretation

image *class*

Classifier: $x \rightarrow y$

$$\underbrace{p(y|x)}_{\text{class}}$$

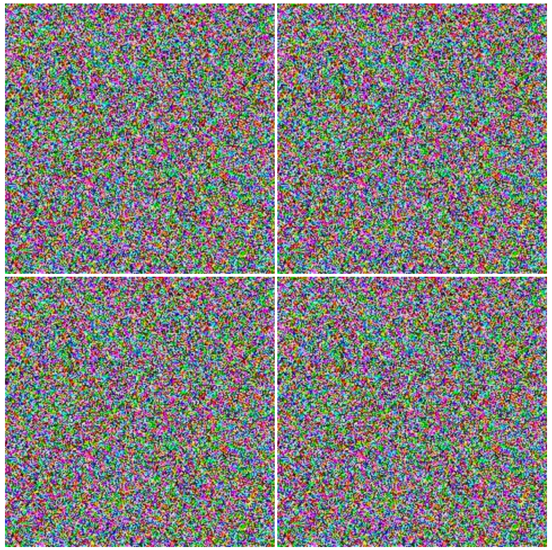
Problems:

1. Poor interpretability

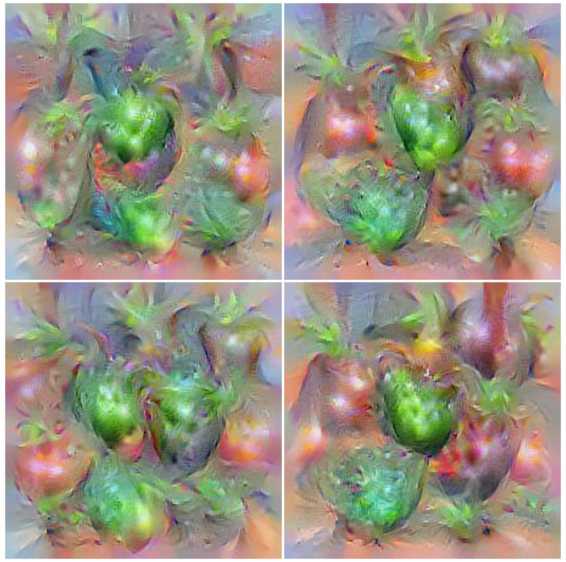
Classifier: $\overset{\text{image}}{x} \longrightarrow \overset{\text{class}}{y}$

$$p(x,y) = \underbrace{\cancel{p(x)}}_{\text{prior}}\;\underbrace{p(y|x)}_{\text{class}}$$

2. Mode collapse



Random initializations



Visualizations

Real images

Solutions:

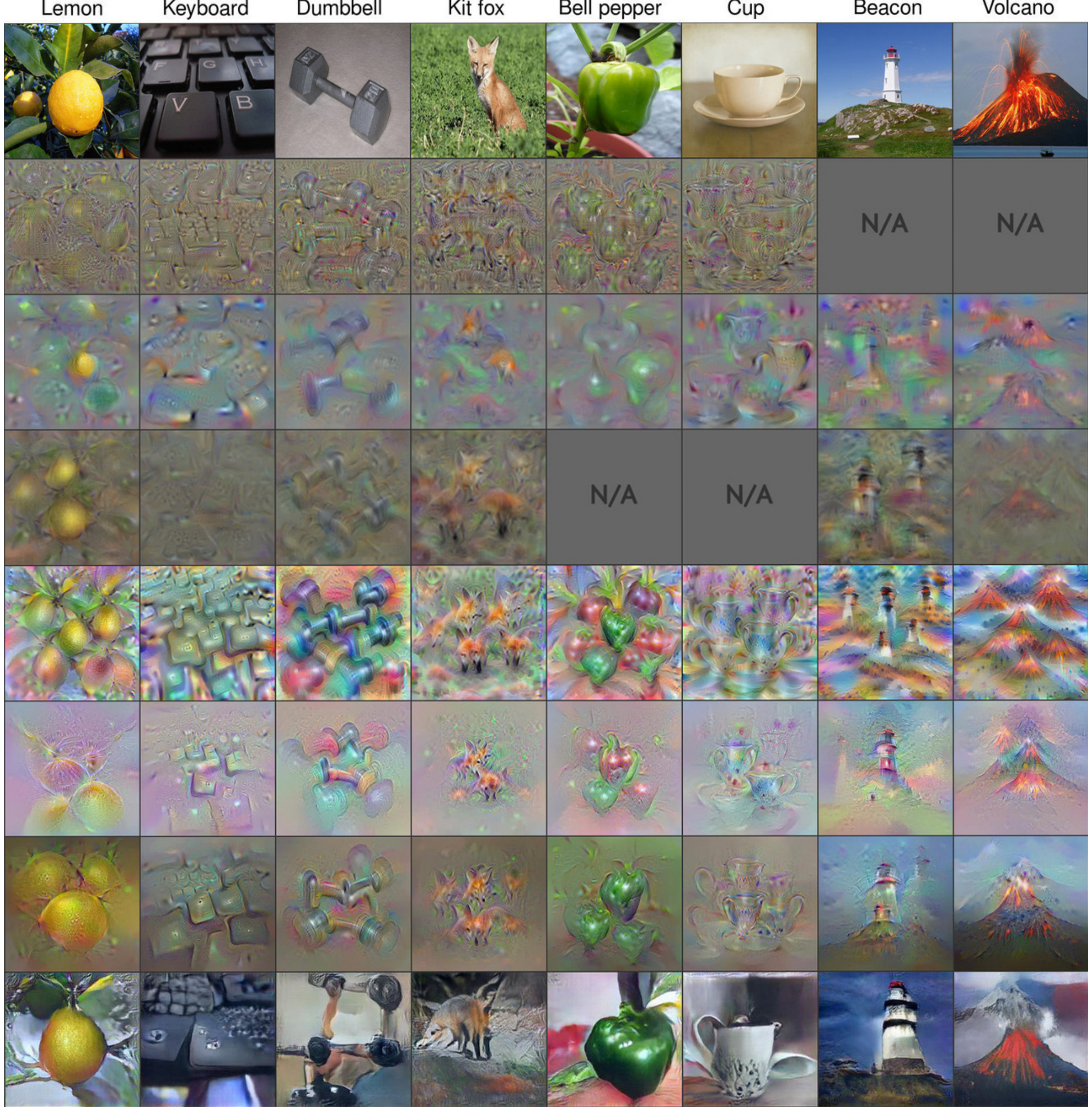
Classifier: $\underset{\textit{image}}{x} \longrightarrow \underset{\textit{class}}{y}$

- total variation (Mahendran & Vedaldi, 2015)
- Gaussian blur (Yosinski et al, 2015)
- α-norms (Simonyan et al, 2014)
- jitter (Mordvintsev et al, 2015)
- center bias (Nguyen et al, 2016)
- GMM (Mordvintsev et al, 2016)

...

$$p(x, y) = \underbrace{p(x)}_{\text{prior}}\underbrace{p(y|x)}_{\text{class}}$$

Update: $$x_{t+1} = x_t + \underbrace{\frac{\partial \phi(x_t)}{\partial x_t}}_{\text{higher activation}} + \underbrace{\frac{\partial R(x_t)}{\partial x_t}}_{\text{more realistic and diverse}}$$



2013

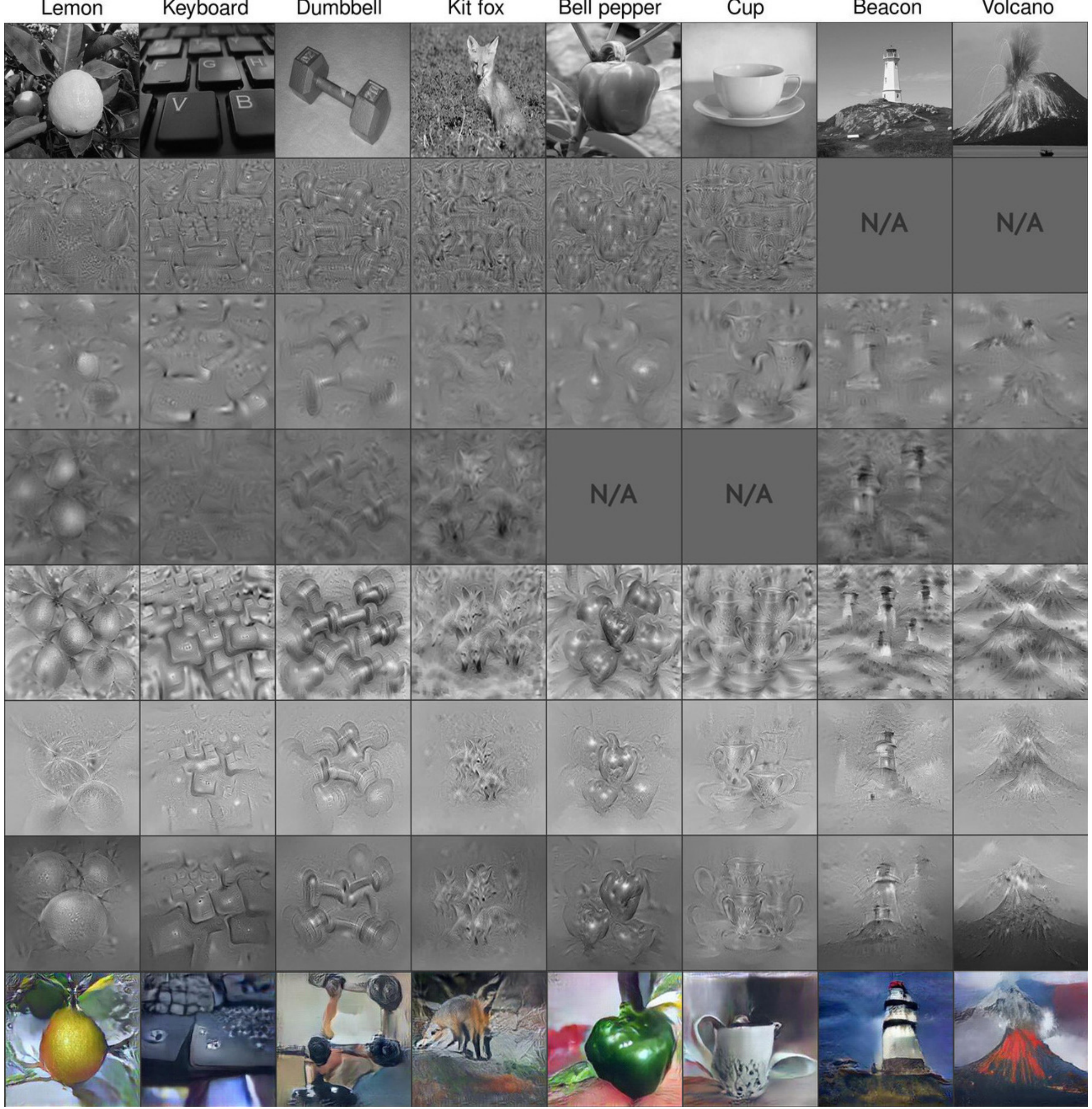
2017

Priors

L_2 norm	Simonyan et al 2014
Gaussian blur	Yosinski et al 2015
Patch statistics	Wei et al 2015
Total variation	Mahendran & Vedaldi 2015
Center bias	Nguyen et al 2016
Mean image	Nguyen et al 2016
Deep generator	Nguyen et al 2016 Nguyen et al 2017

2013

2017



Priors

L_2 norm	Simonyan et al 2014
Gaussian blur	Yosinski et al 2015
Patch statistics	Wei et al 2015
Total variation	Mahendran & Vedaldi 2015
Center bias	Nguyen et al 2016
Mean image	Nguyen et al 2016
Deep generator	Nguyen et al 2016 Nguyen et al 2017

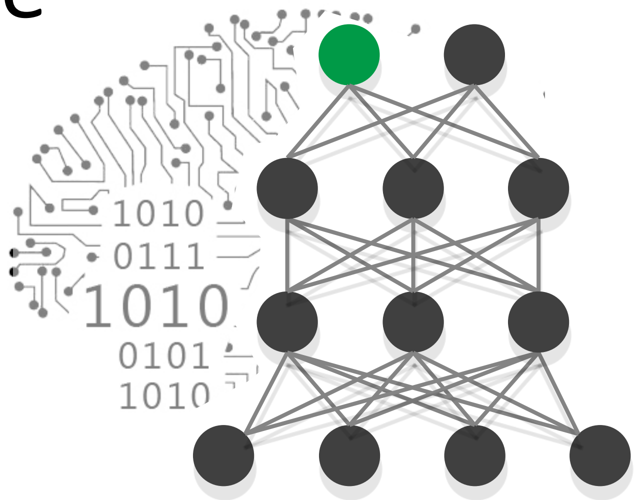
Finding what artificial neurons want to see

2. Image generator

3. 3D renderer

1. Pixel-wise

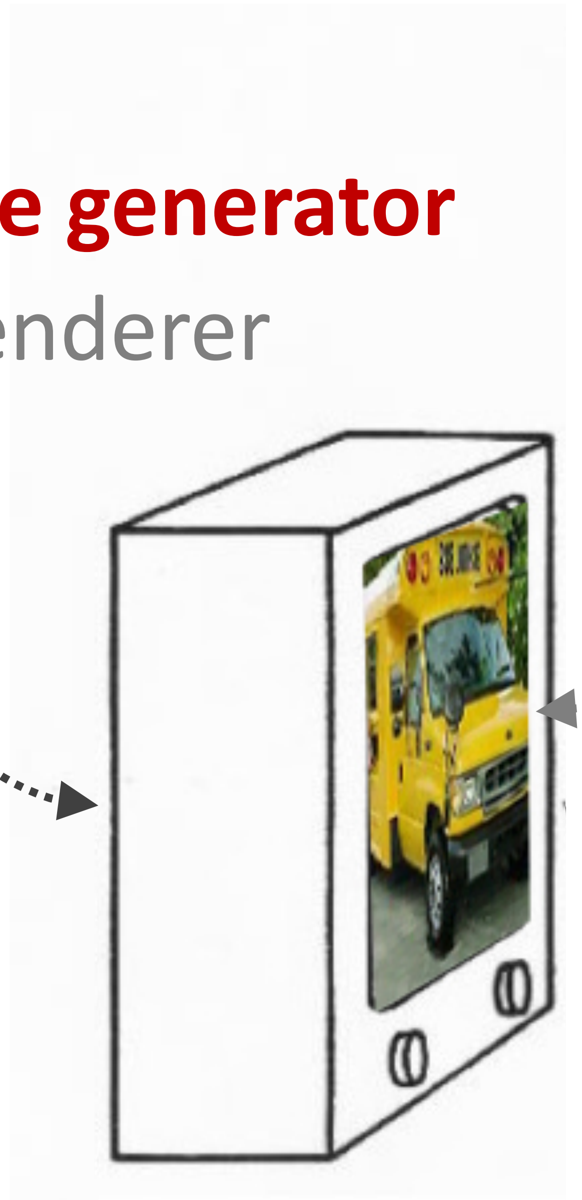
school bus



Finding what artificial neurons want to see

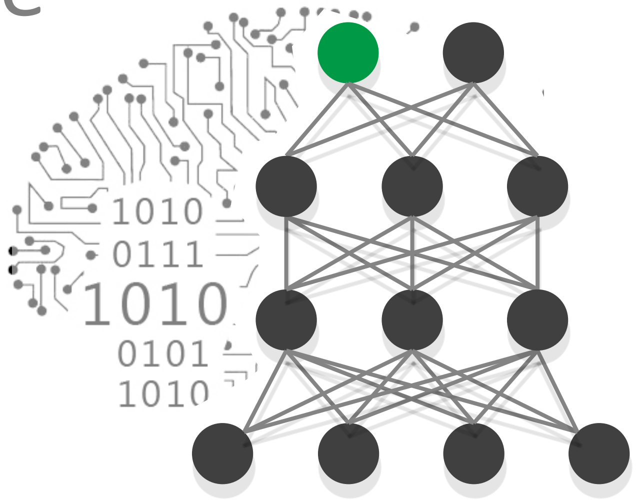
2. Image generator

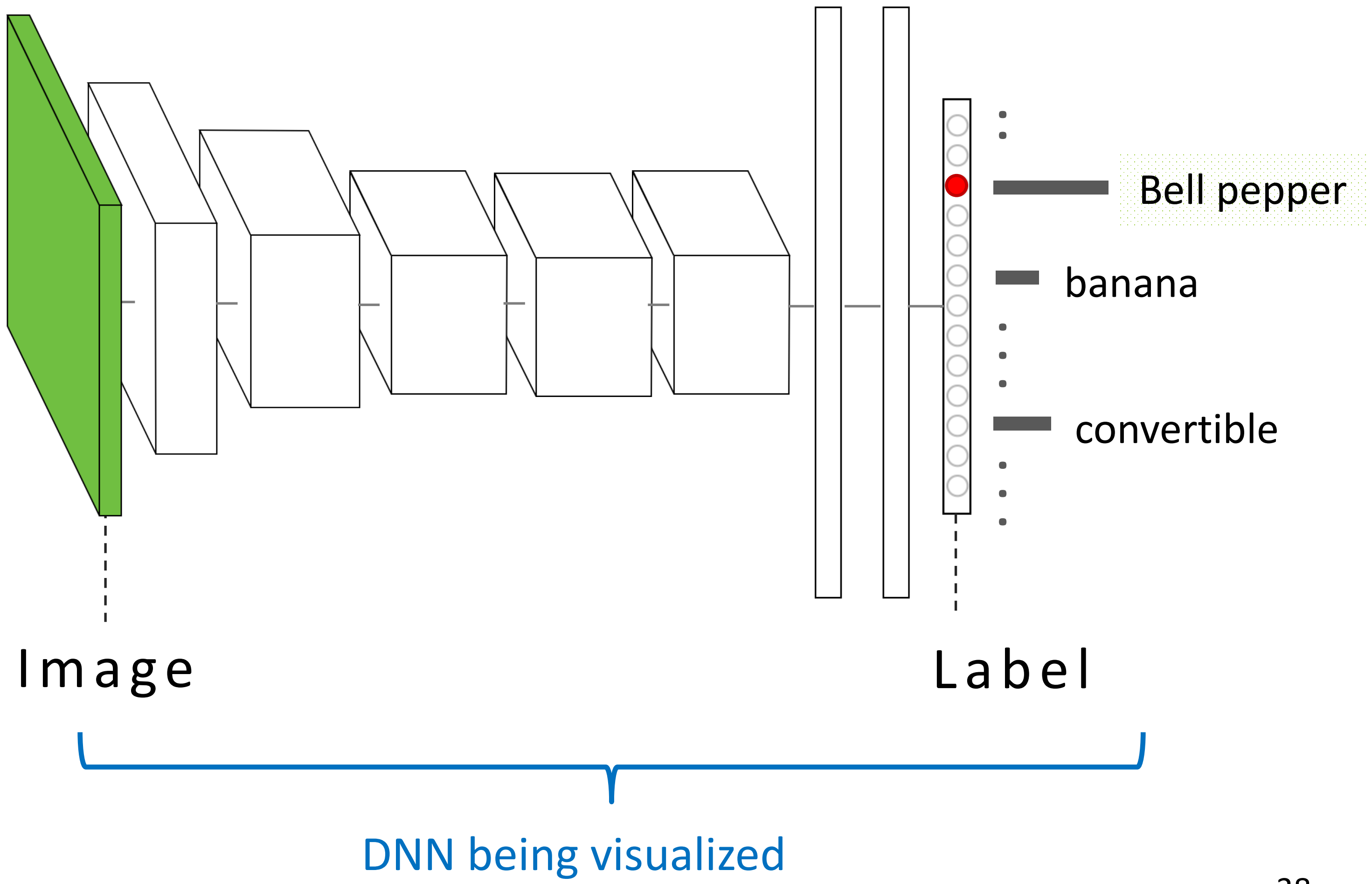
3. 3D renderer



1. Pixel-wise

school bus




Bell pepper
banana
convertible
Image
Label
DNN being visualized

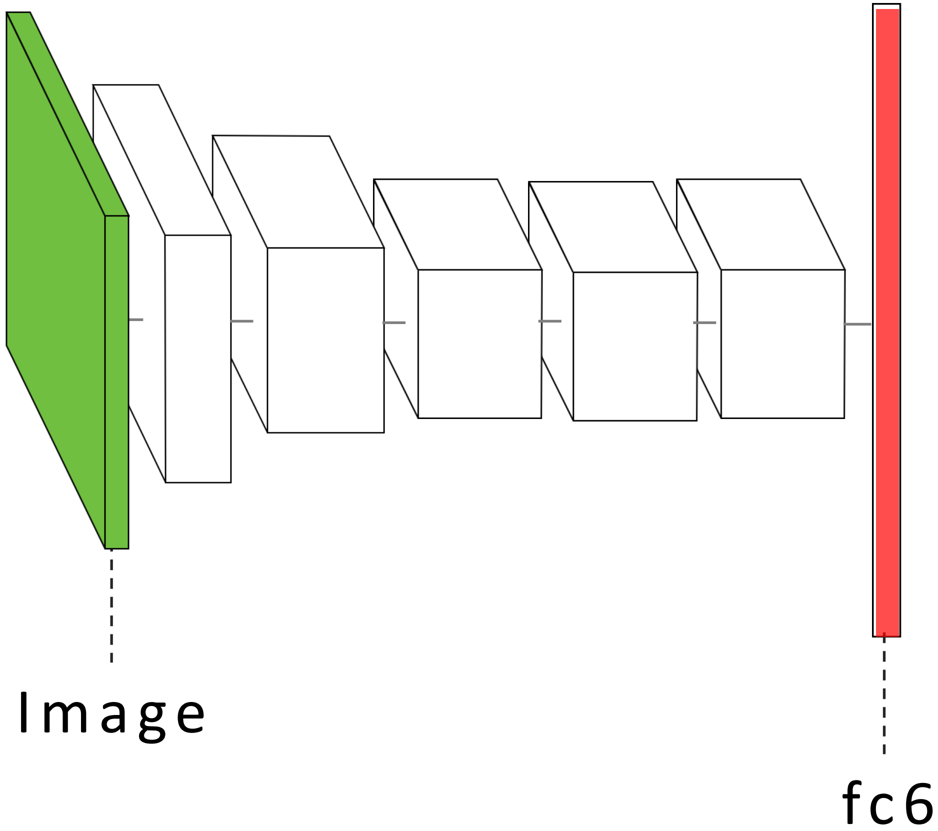
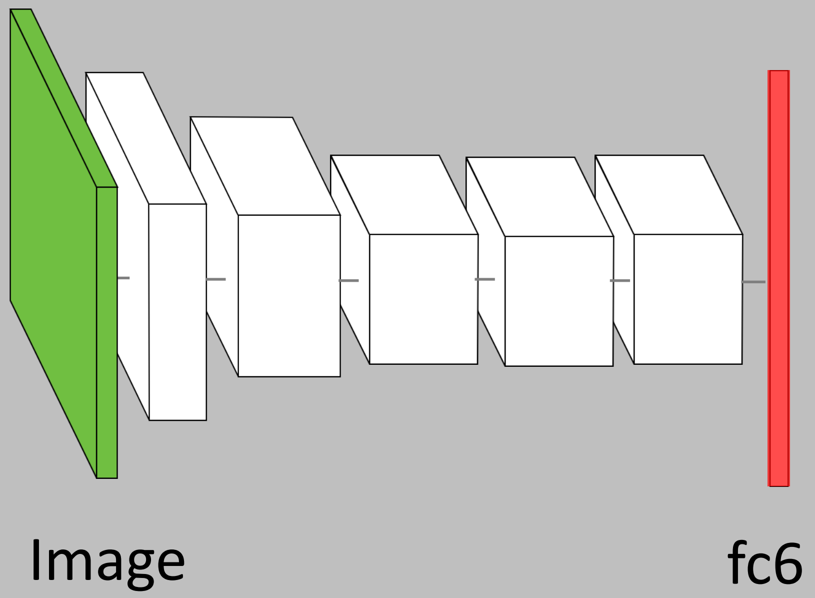
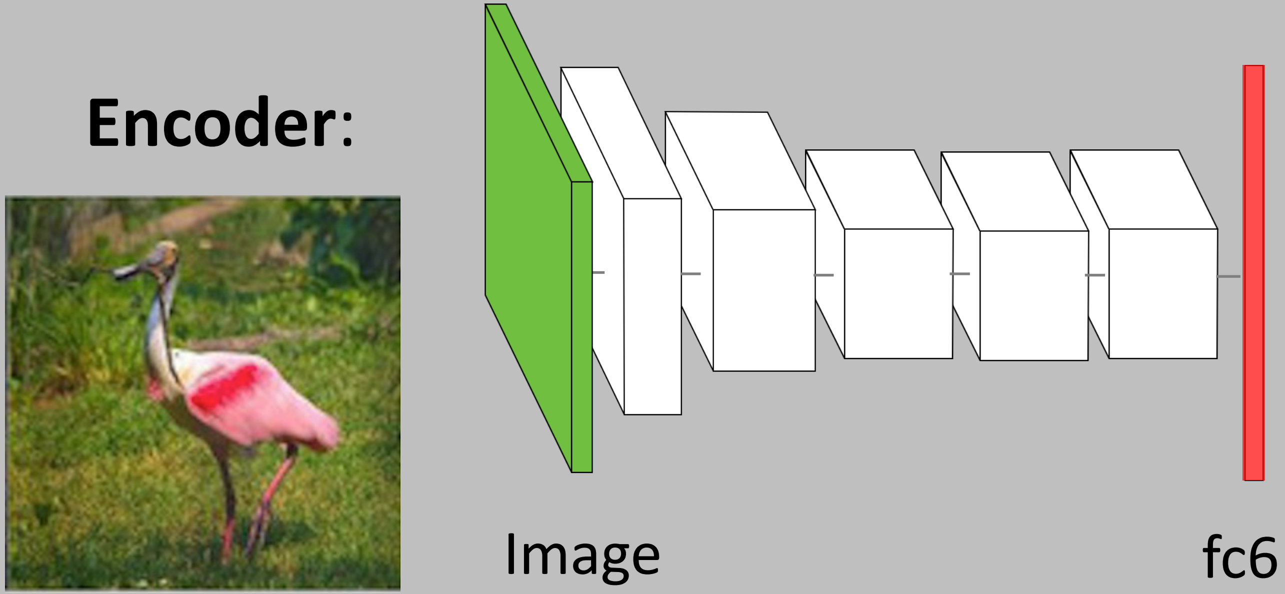


Image
fc6

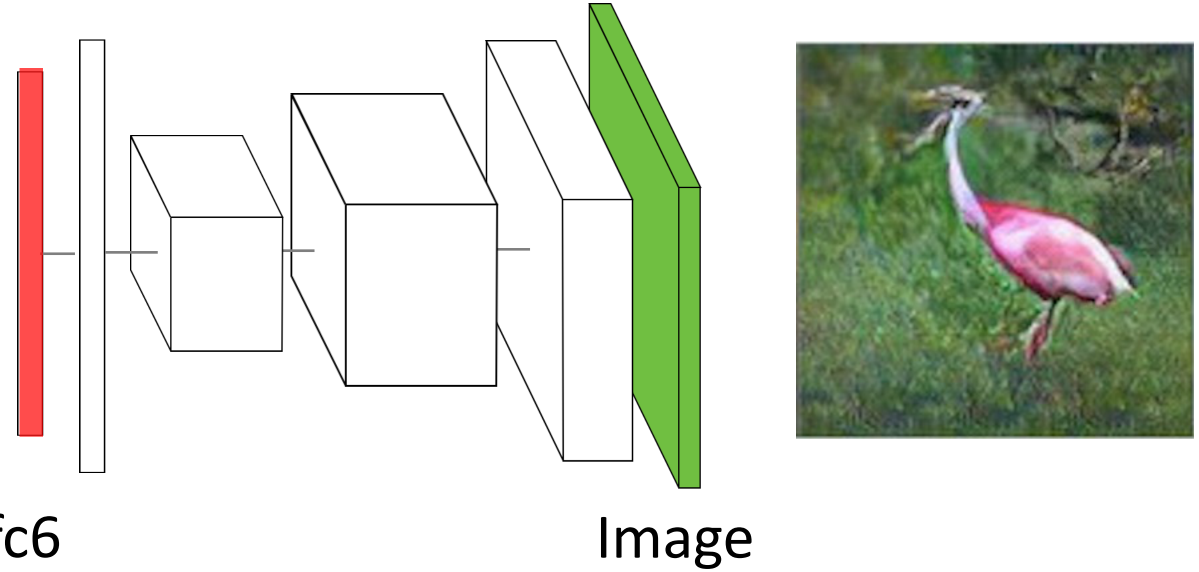
Encoder:



Encoder:

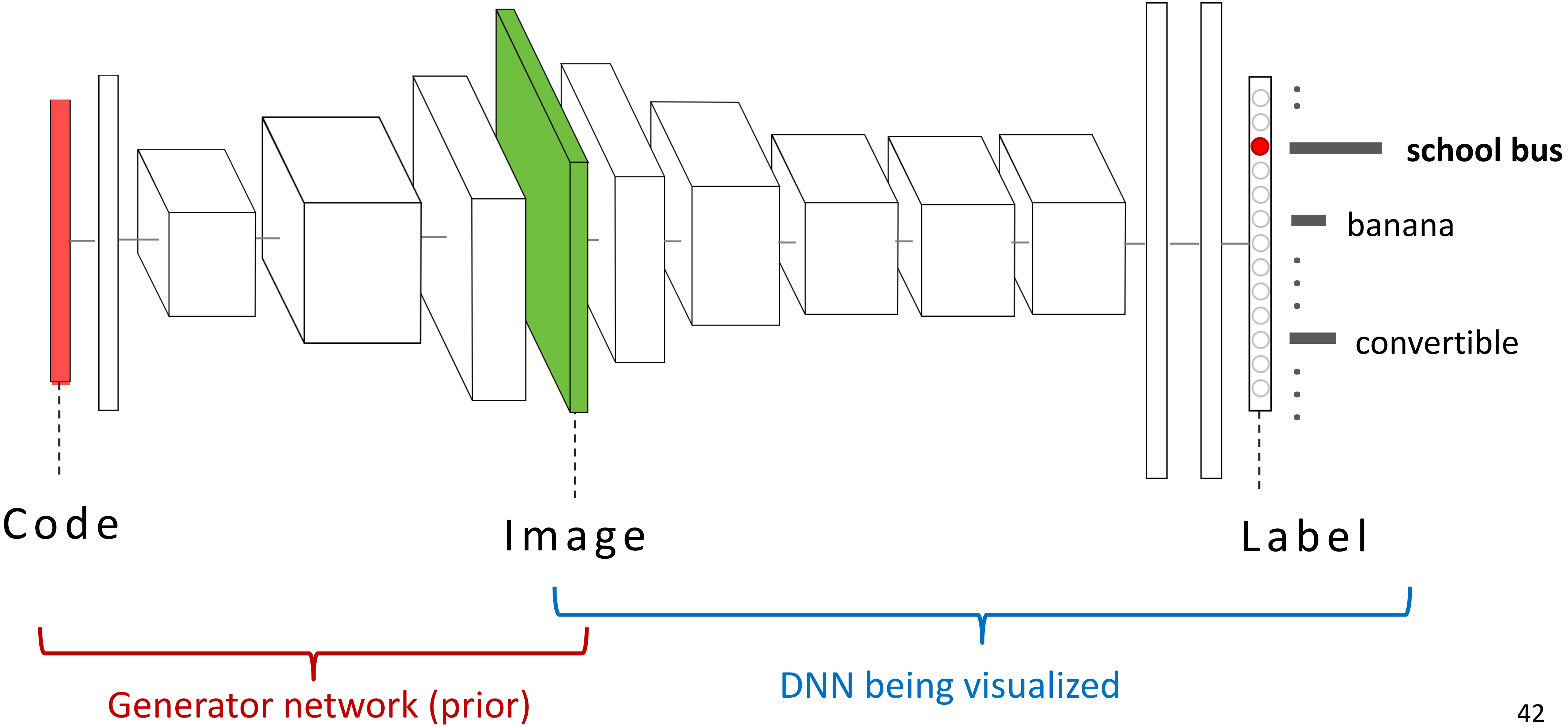


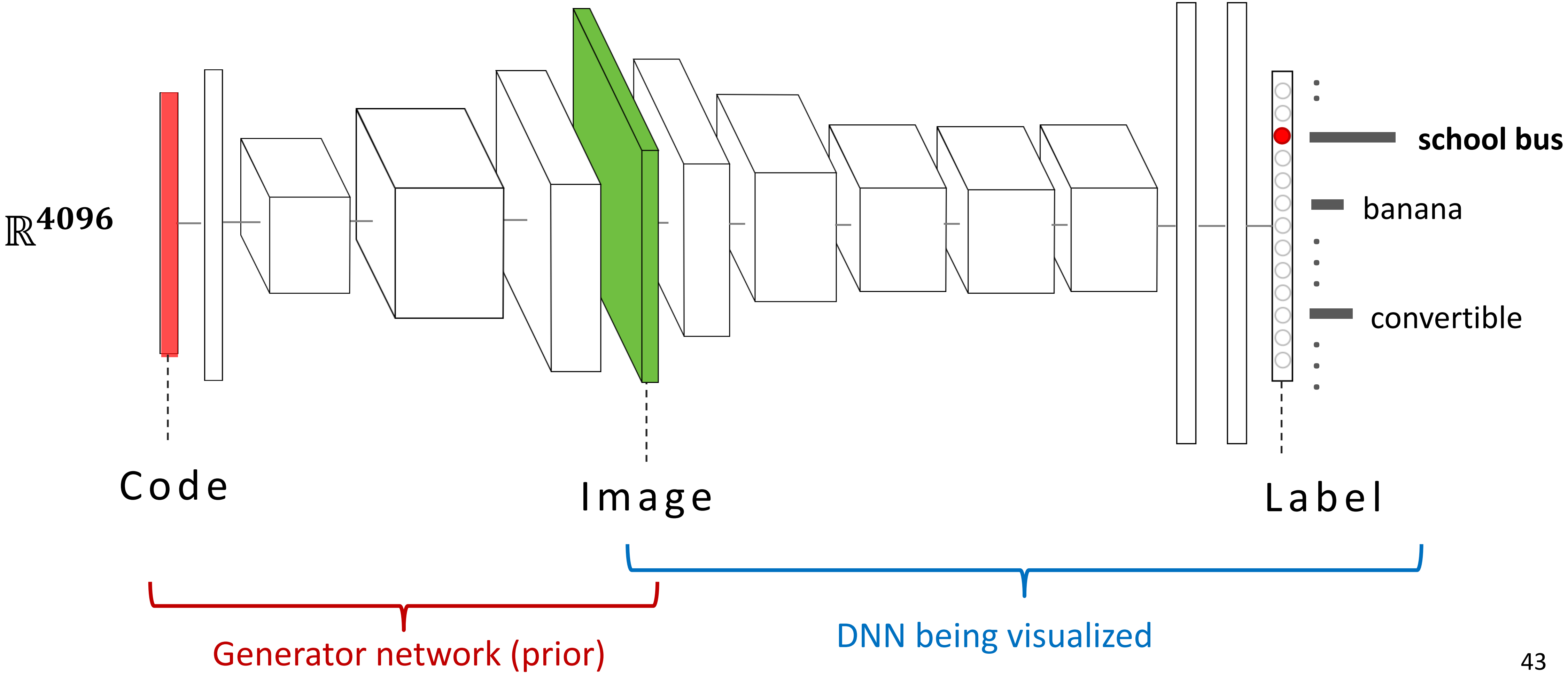
Generator / **Decoder**:



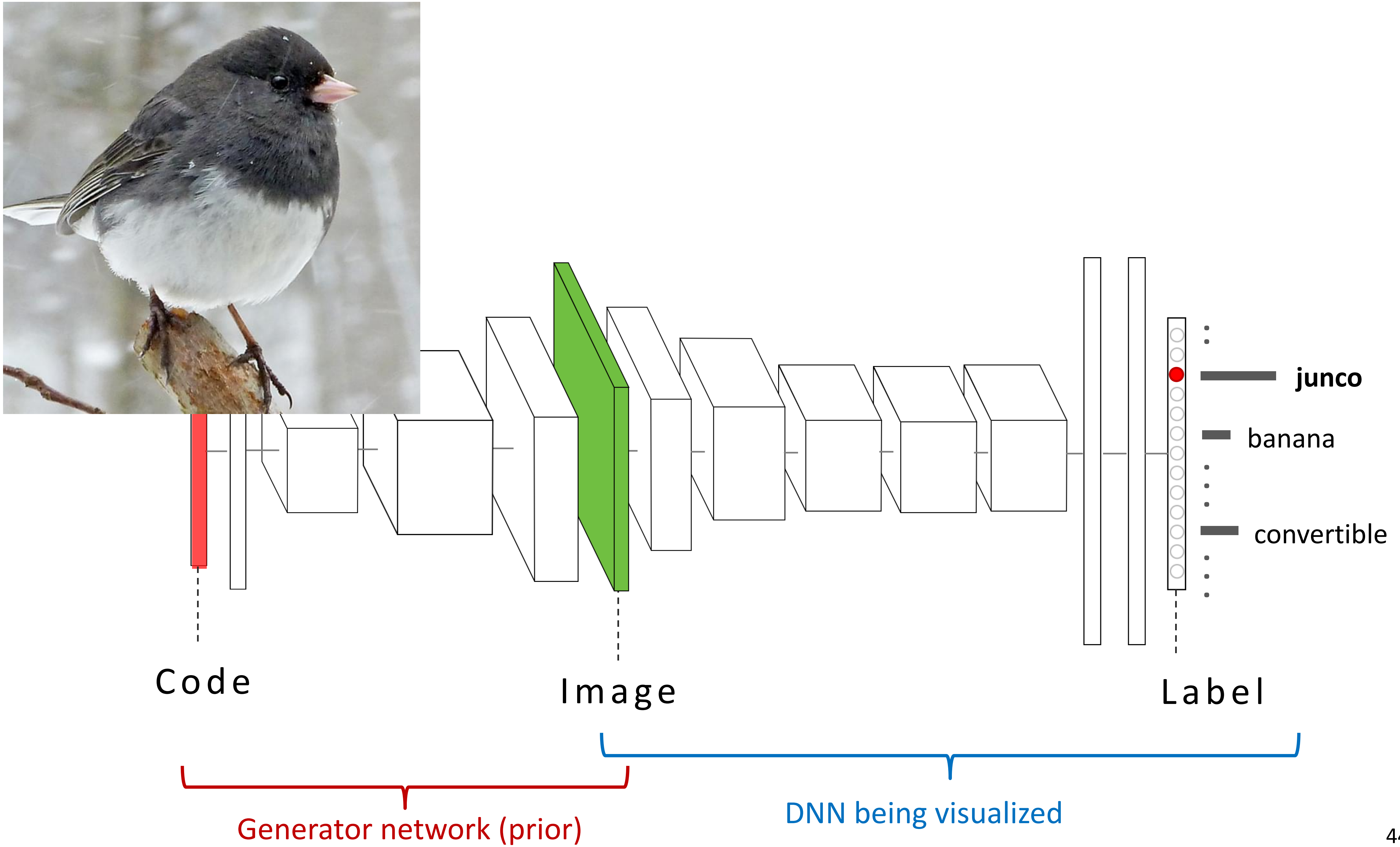
Training losses:

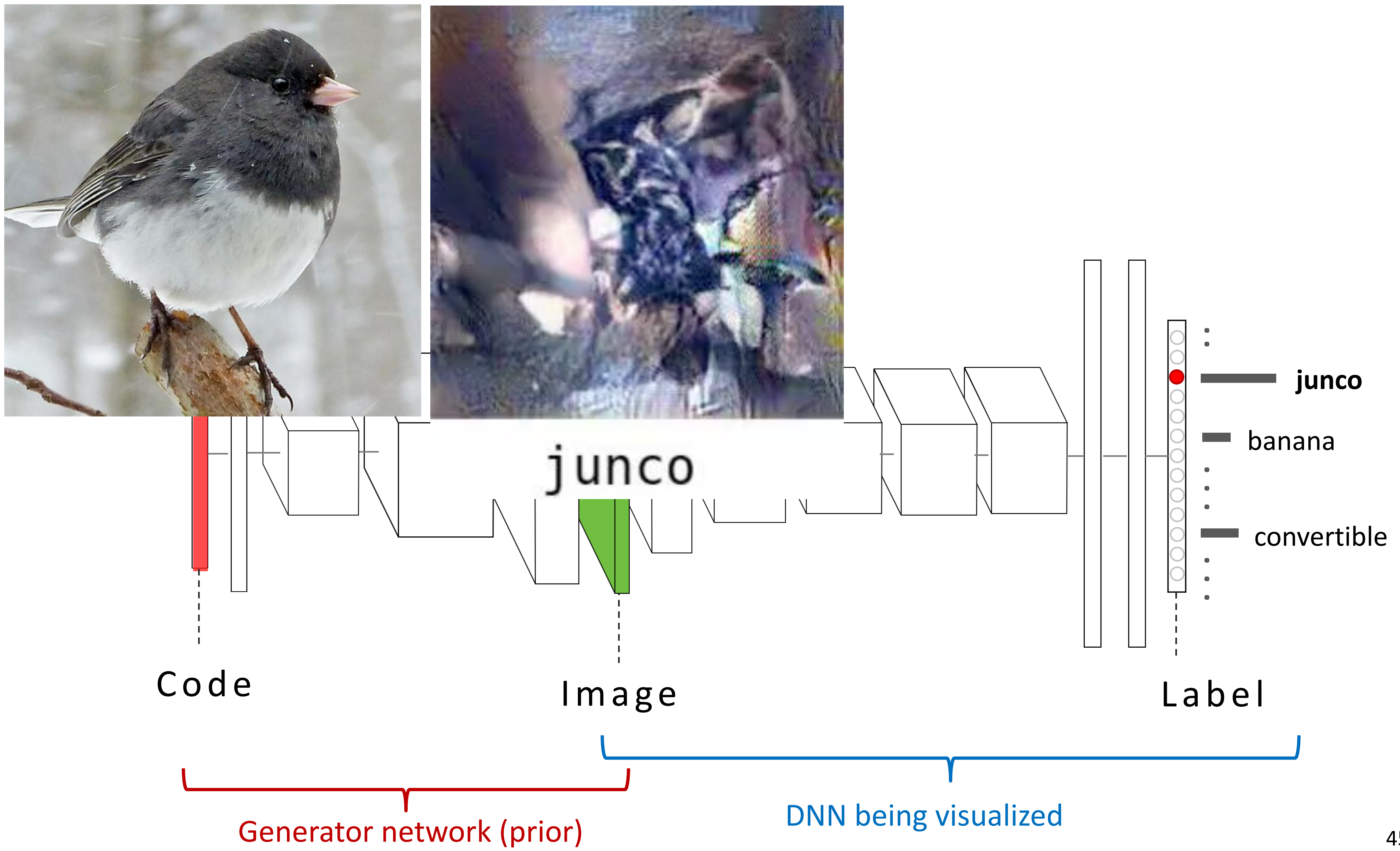
1. Reconstruction
2. Adversarial (GAN)
3. Feature matching


school bus
banana
convertible
Code
Image
Label
Generator network (prior)
DNN being visualized



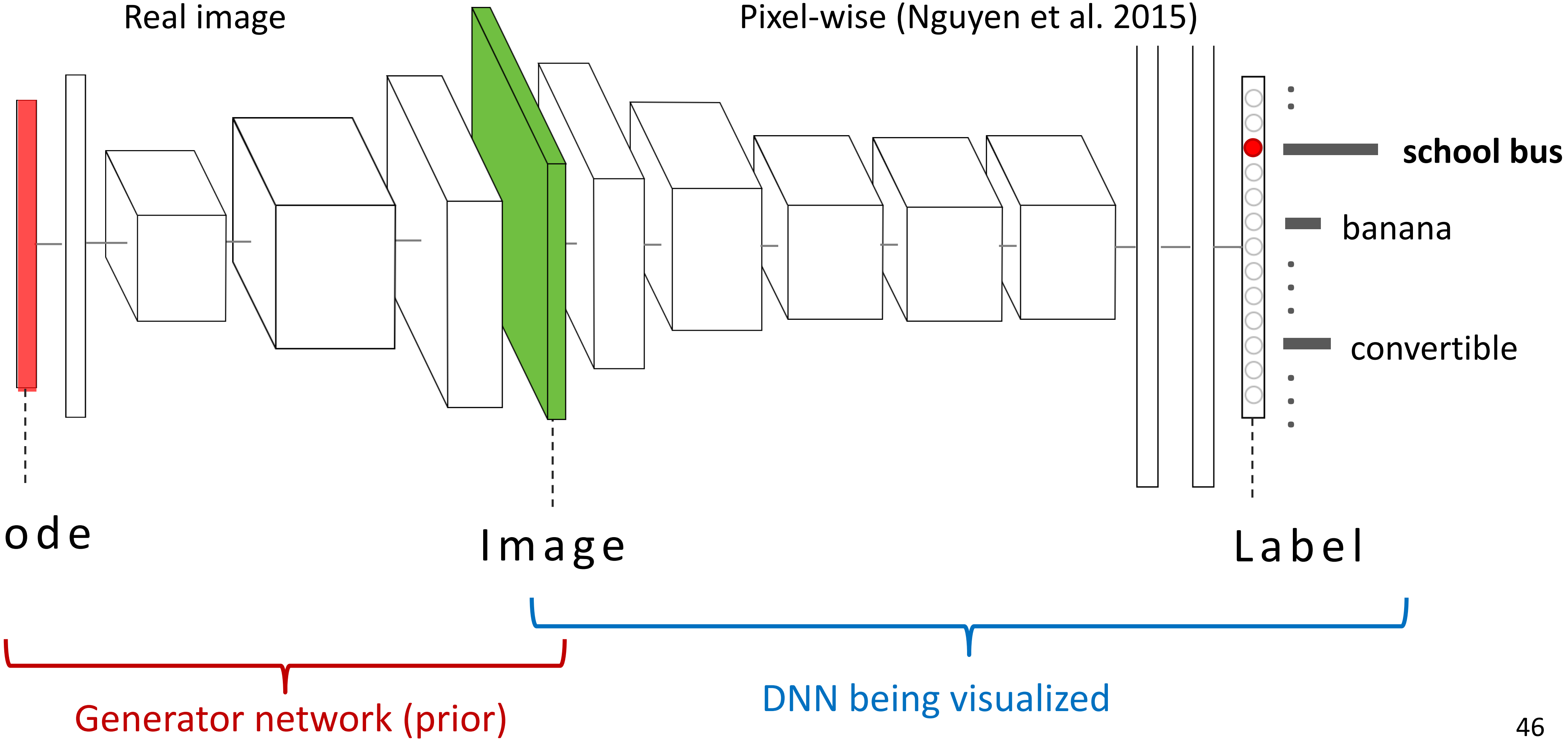
$\mathbb{R}^{4096}$
school bus
banana
convertible
Code
Image
Label
Generator network (prior)
DNN being visualized


junco
banana
convertible
Code
Image
Label
Generator network (prior)
DNN being visualized


junco
junco
banana
convertible
Code
Image
Label
Generator network (prior)
DNN being visualized

Real image

Pixel-wise (Nguyen et al. 2015)



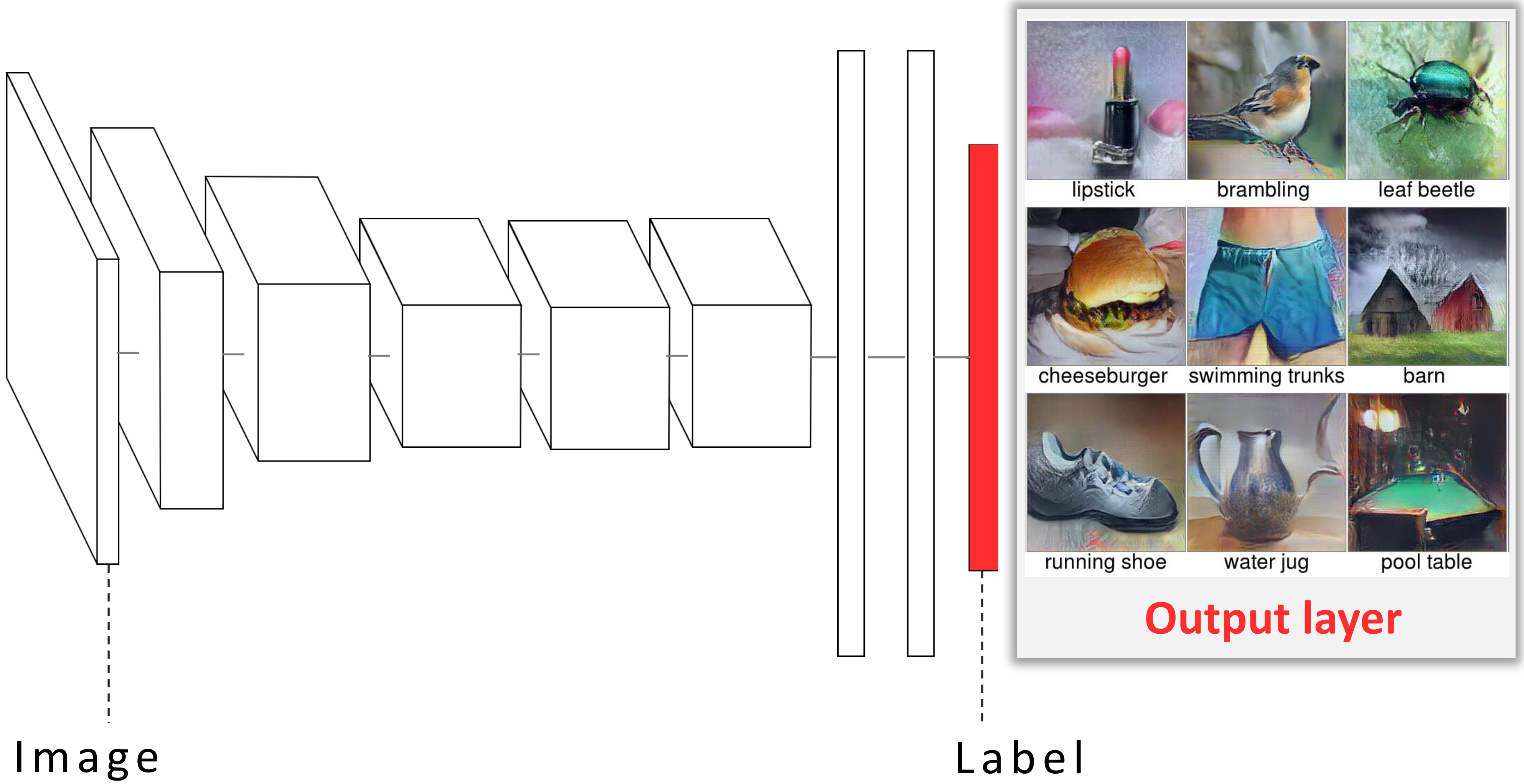
Nguyen et al. 2016 Synthesizing the preferred inputs for neurons in neural networks…


Dosovitskiy

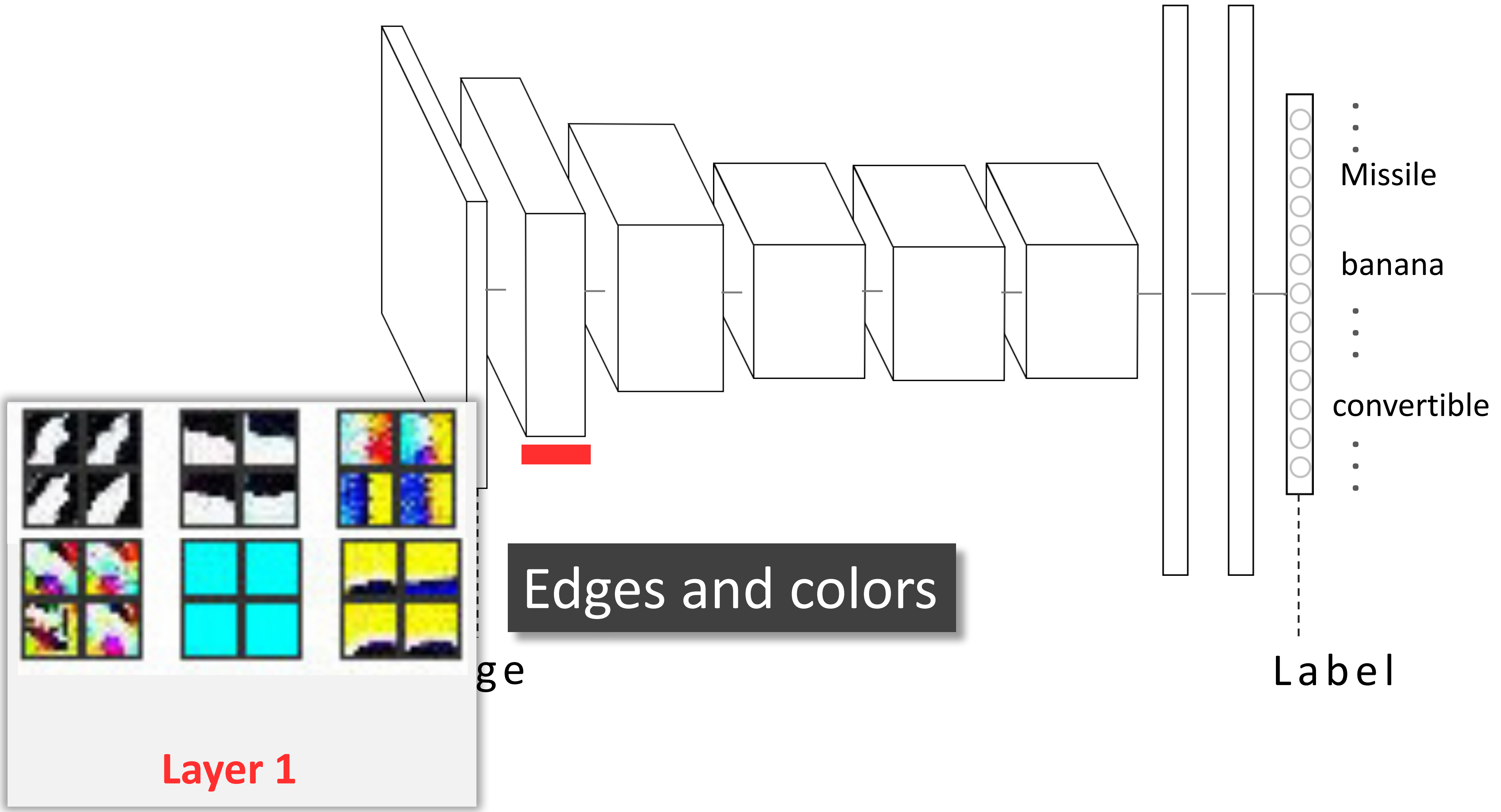

Yosinski


Brox

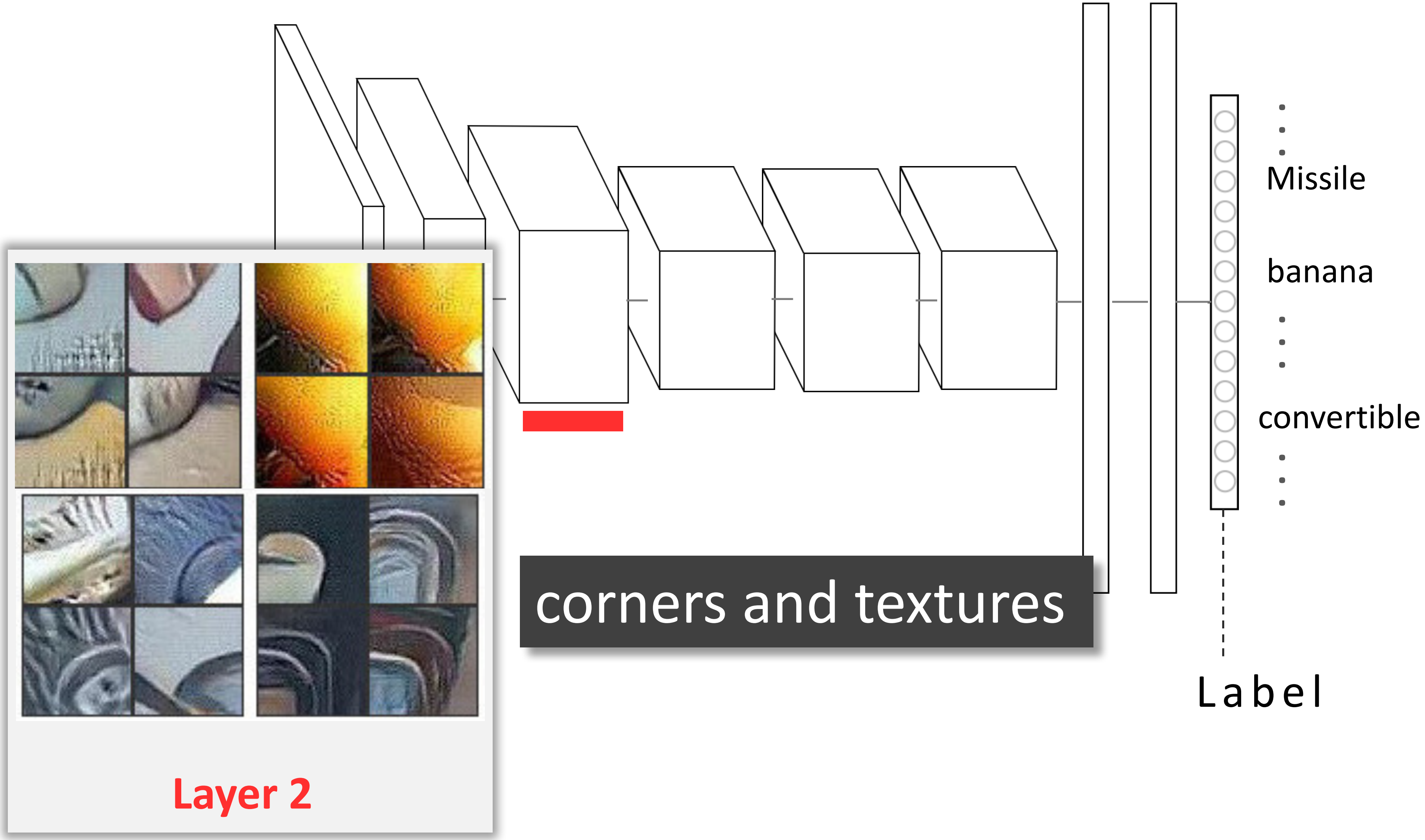

Clune


lipstick
brambling
leaf beetle
cheeseburger
swimming trunks
barn
running shoe
water jug
pool table
Output layer
Image
Label

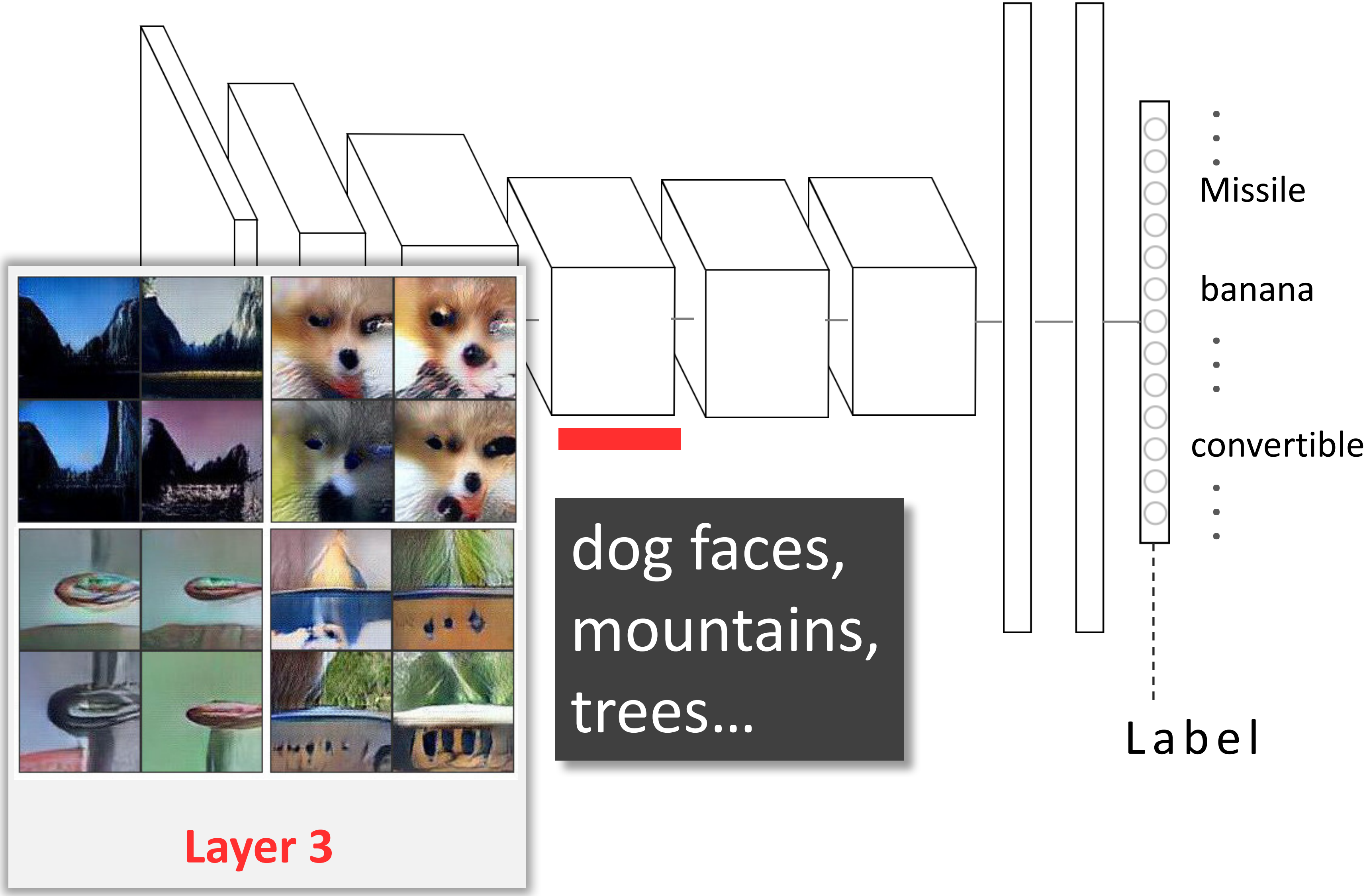
Hidden neurons



Hidden neurons

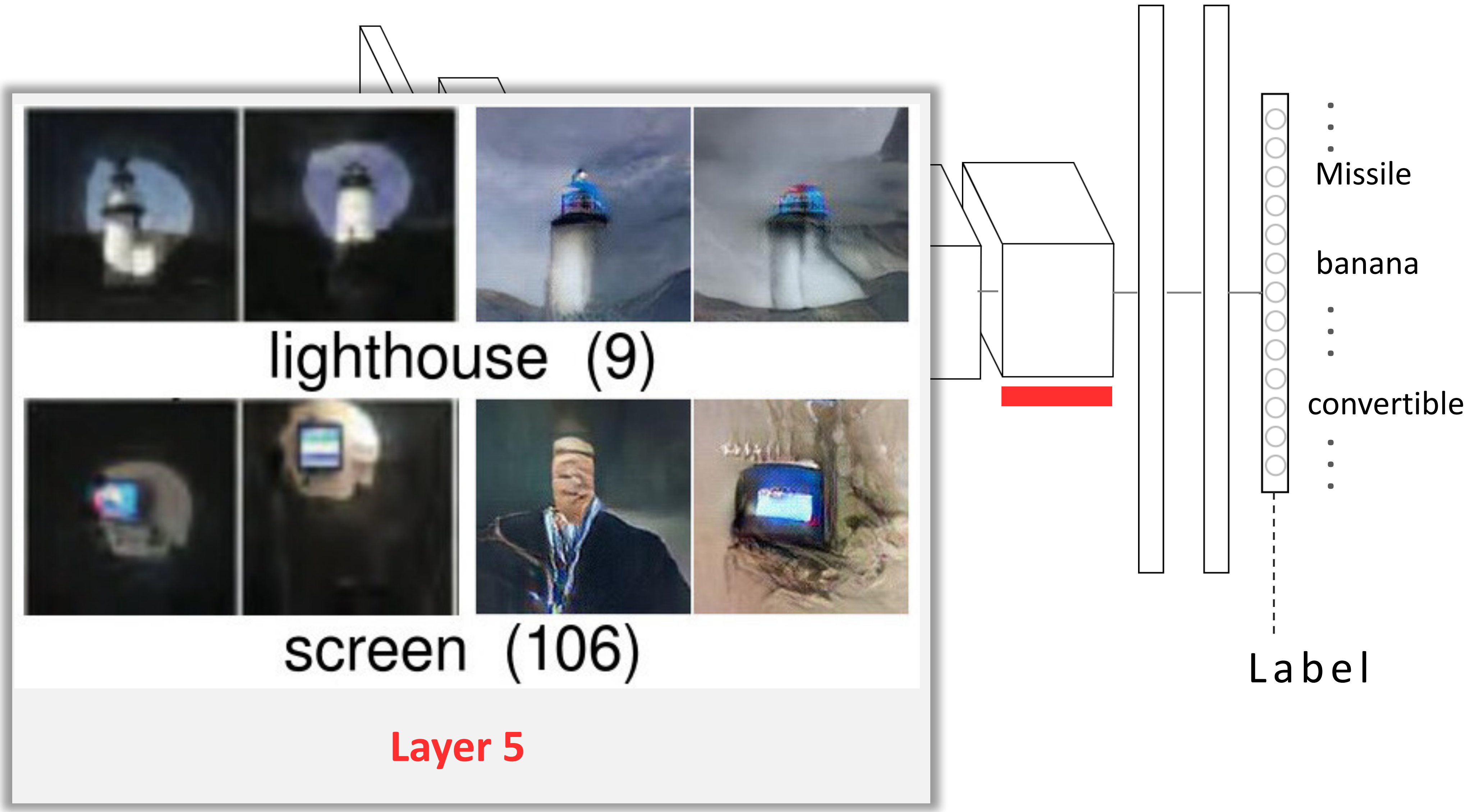


Hidden neurons

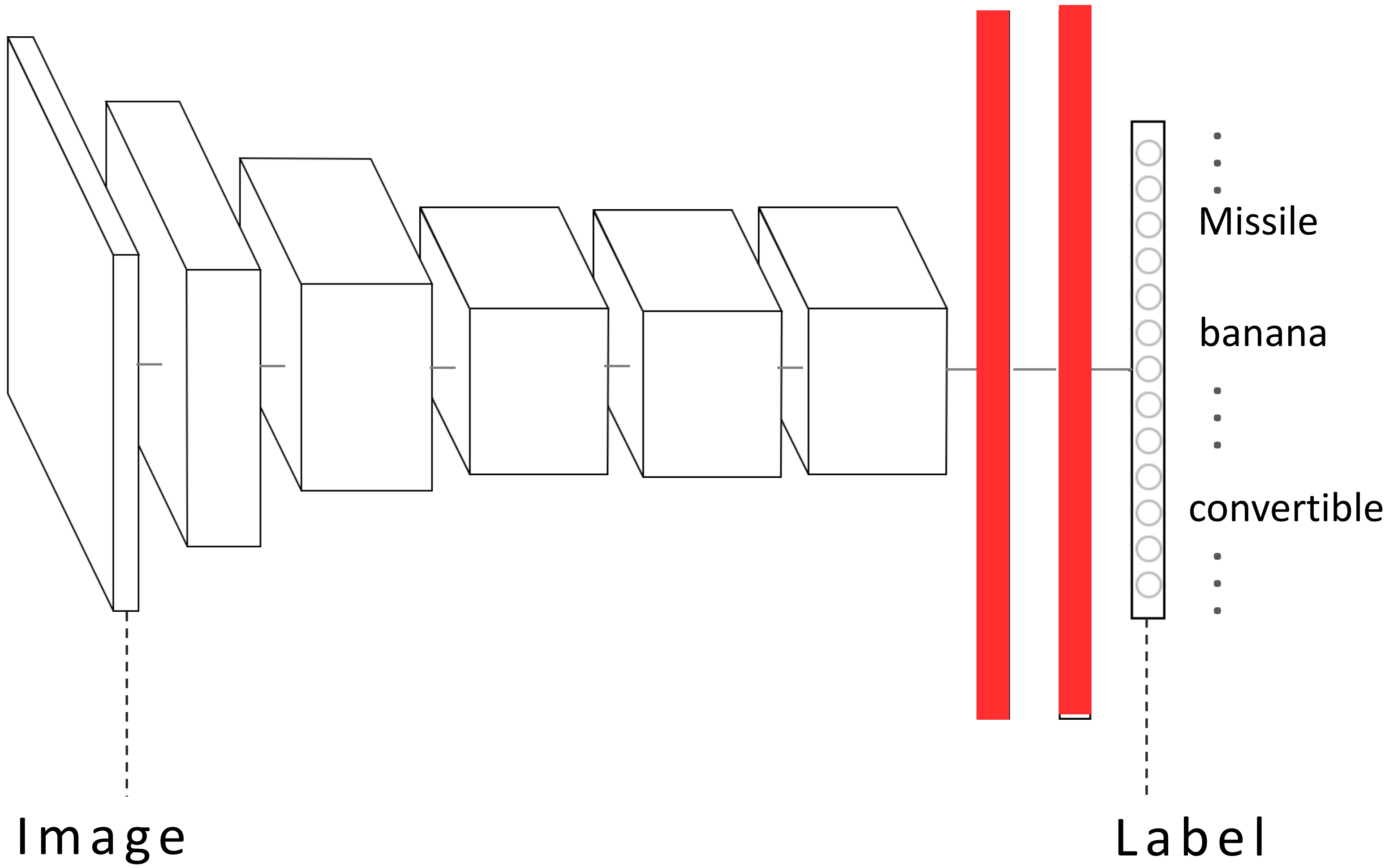


dog faces, mountains, trees...

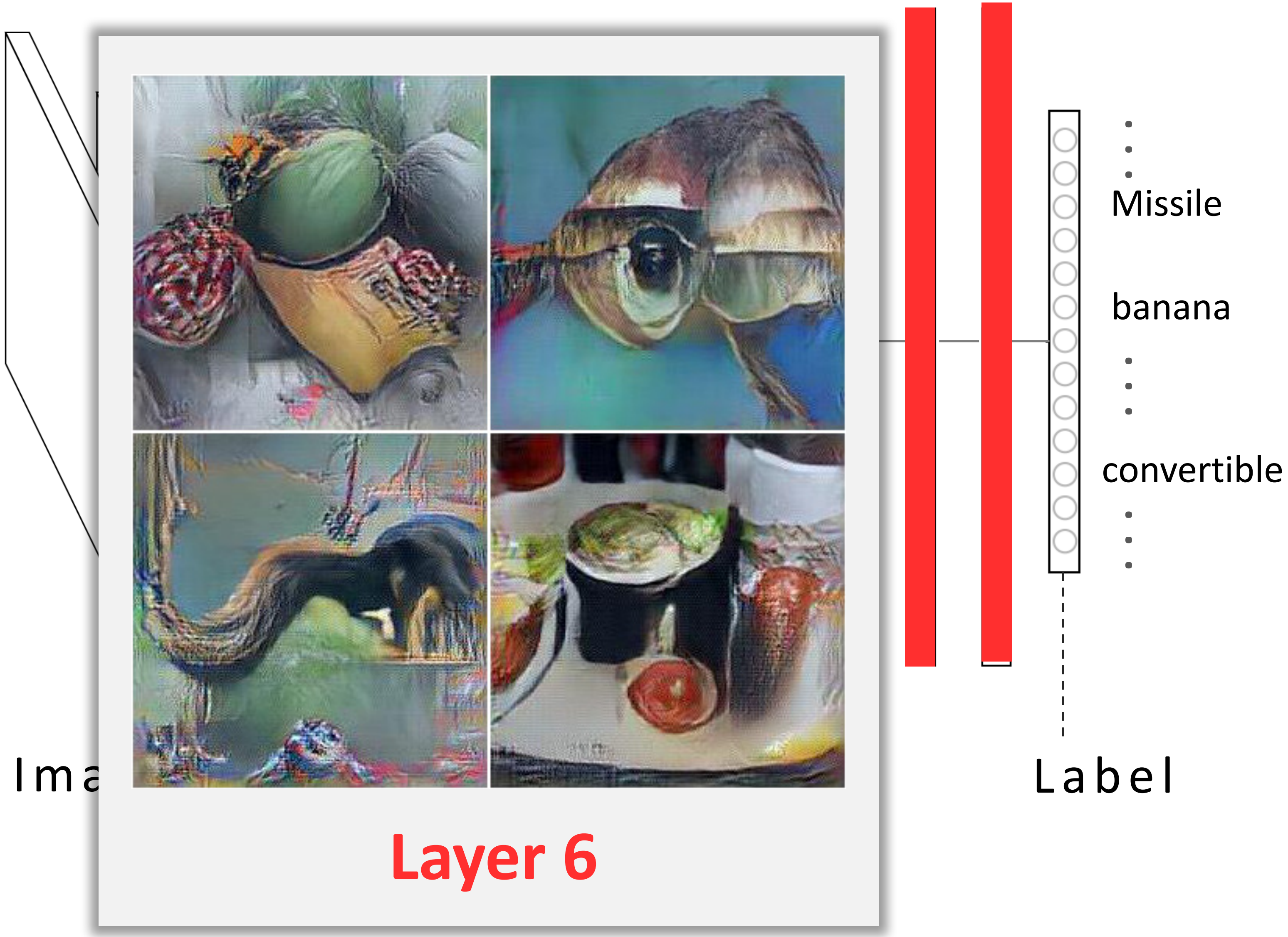
Hidden neurons



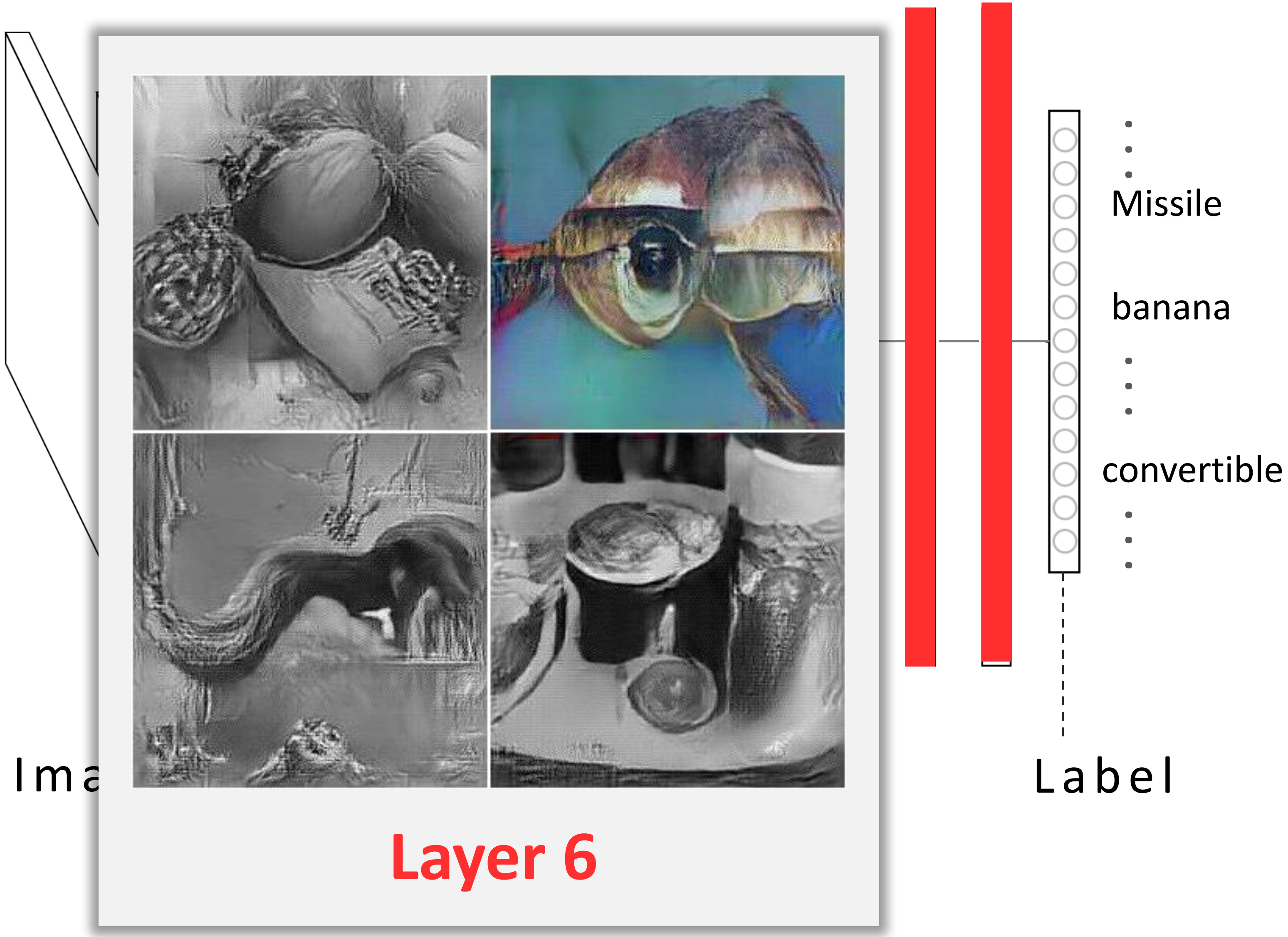
Fully-connected layer 6 and 7



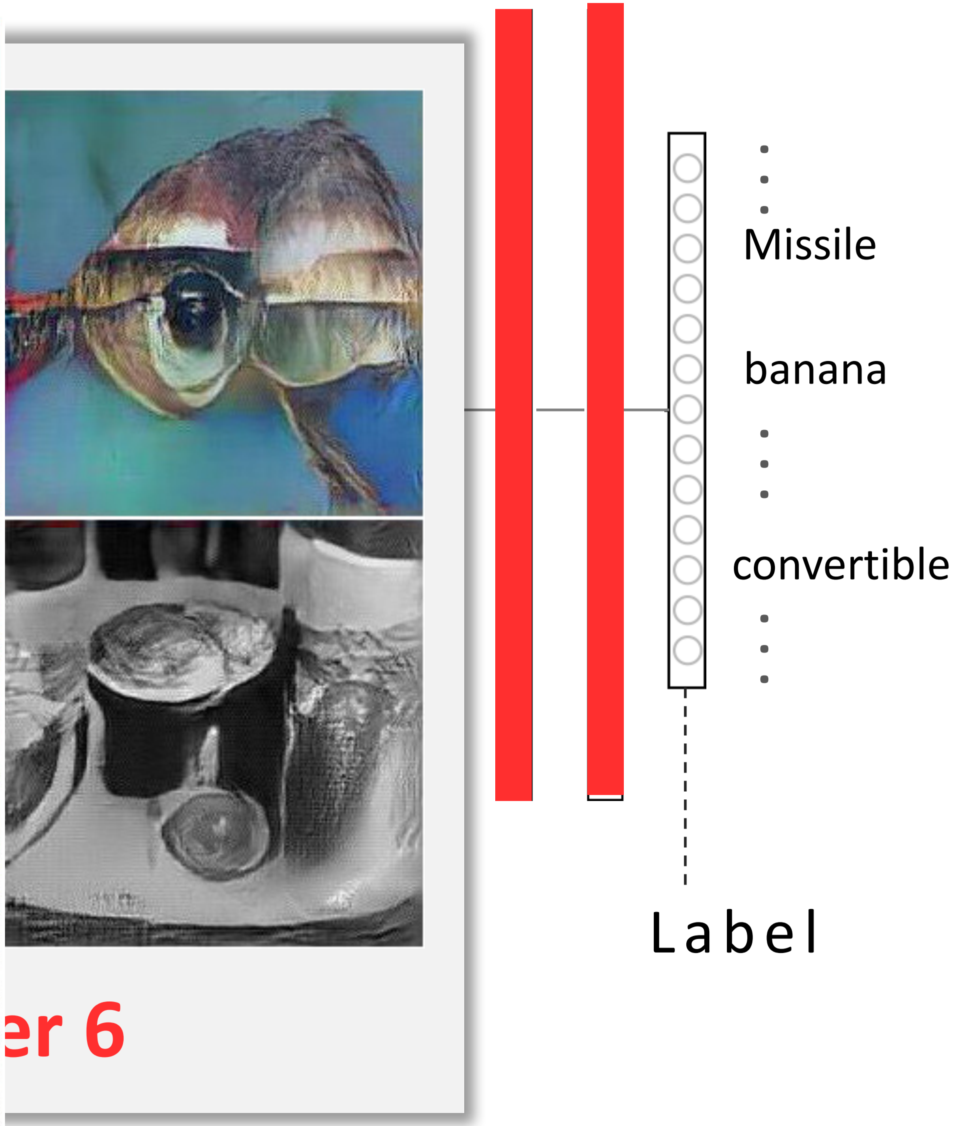
Uninterpretable stimuli – distributed coding?

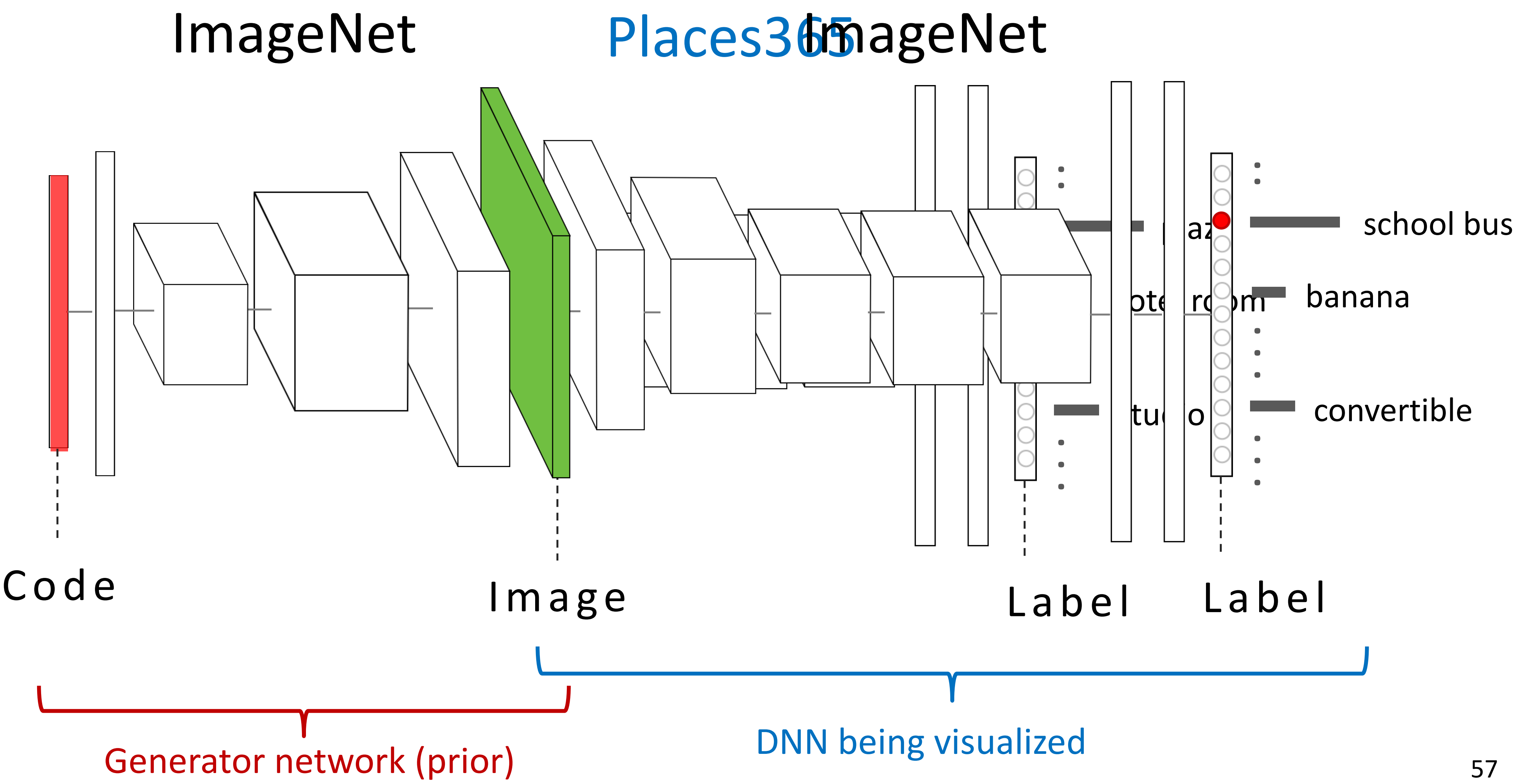


Uninterpretable stimuli – distributed coding?



Uninterpretable stimuli – distributed coding?




ImageNet
Places365
ImageNet
school bus
banana
convertible
Code
Image
Label
Label
Generator network (prior)
DNN being visualized

BigGAN-AM on Places365

Li et al. 2019


Qi Li


Long Mai

Real images


plaza


hotel room

Synthesized stimuli


plaza


hotel room

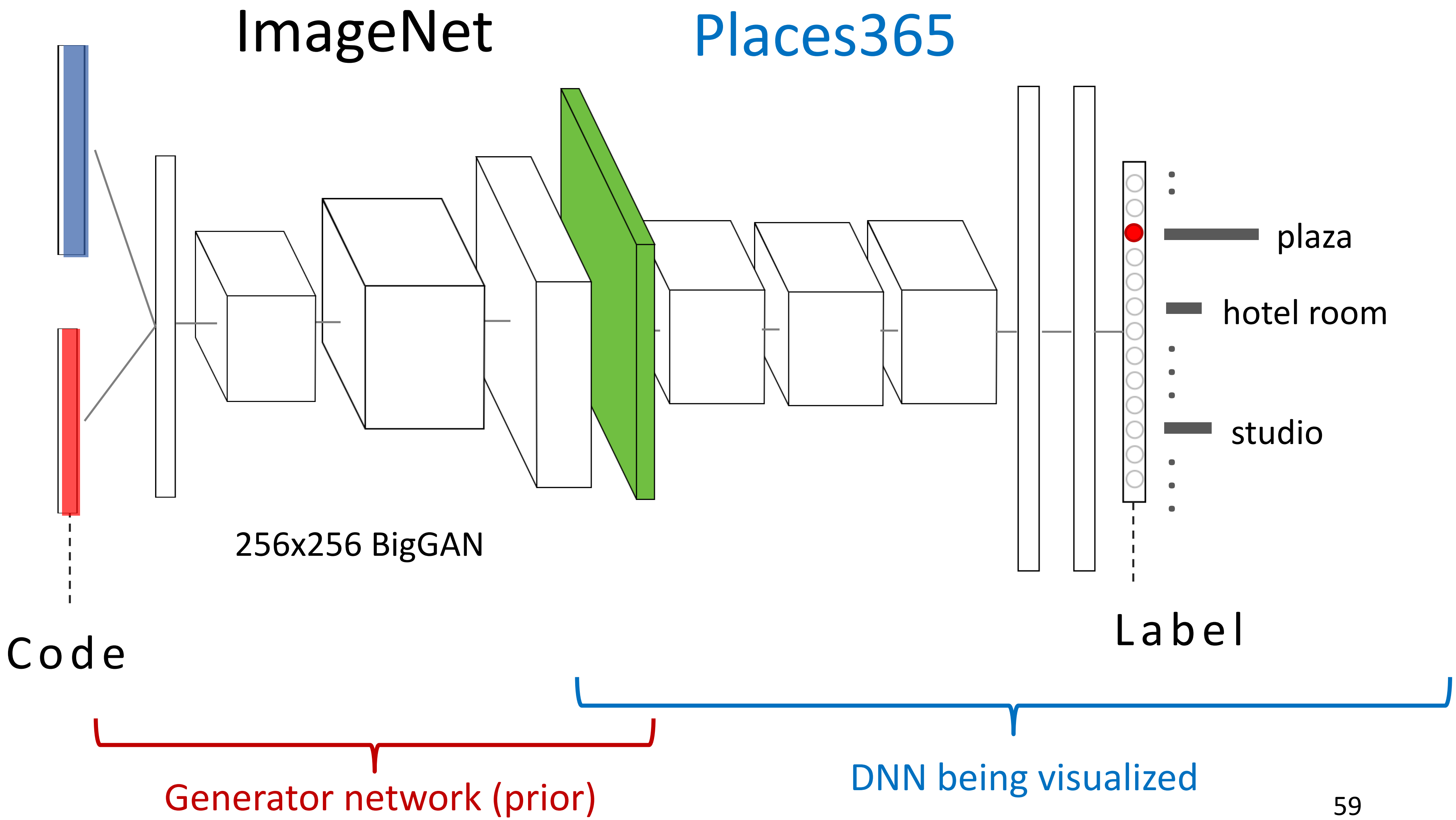
vs. Nguyen et al. 2016, 2017
- Higher image fidelity
- Synthesizing a batch instead of a single image

Label

Optimizing a set of stimuli simultaneously

Latent $\in \mathbb{R}^{140}$
- colors,
- poses,
- background

Semantic $\in \mathbb{R}^{128}$

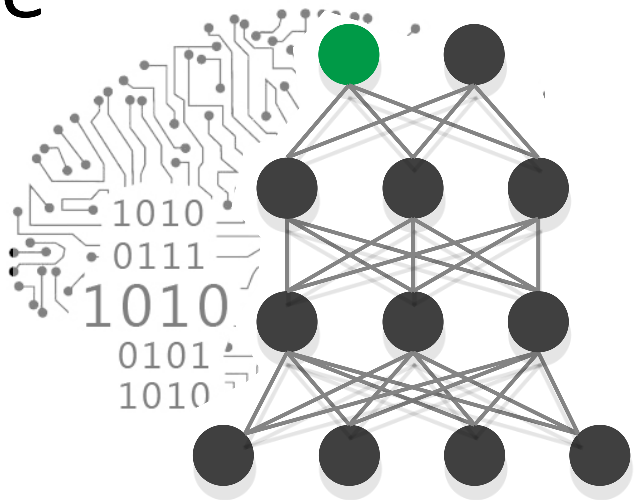


Finding what artificial neurons want to see

2. Image generator
3. 3D renderer

1. Pixel-wise

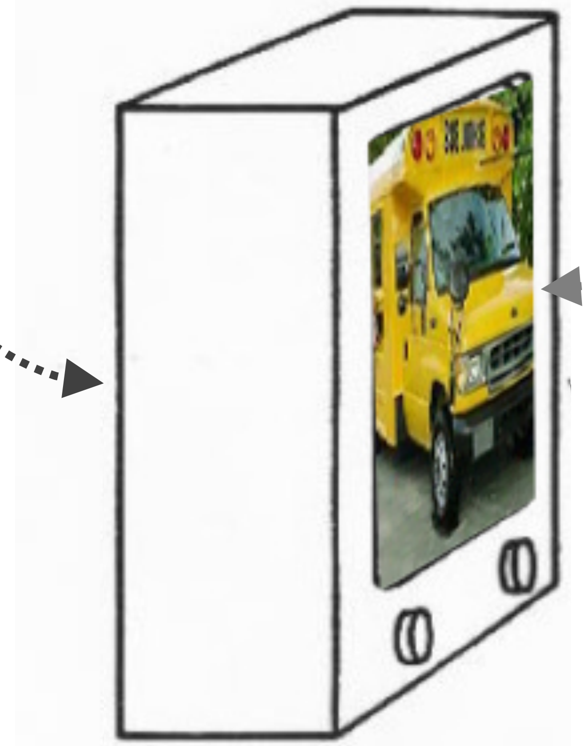
school bus



Finding what artificial neurons want to see

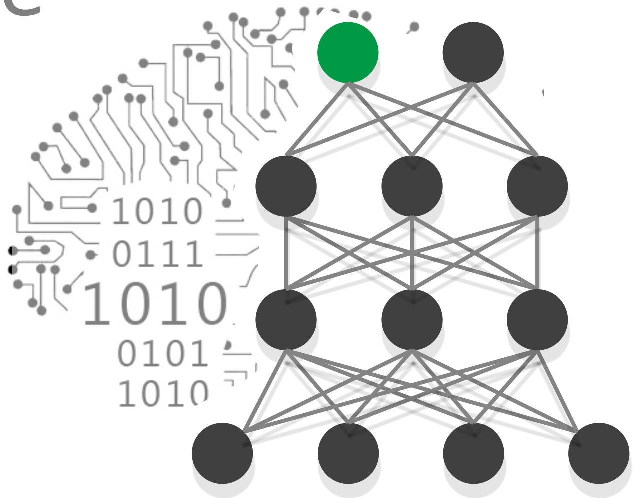
2. Image generator

3. 3D renderer

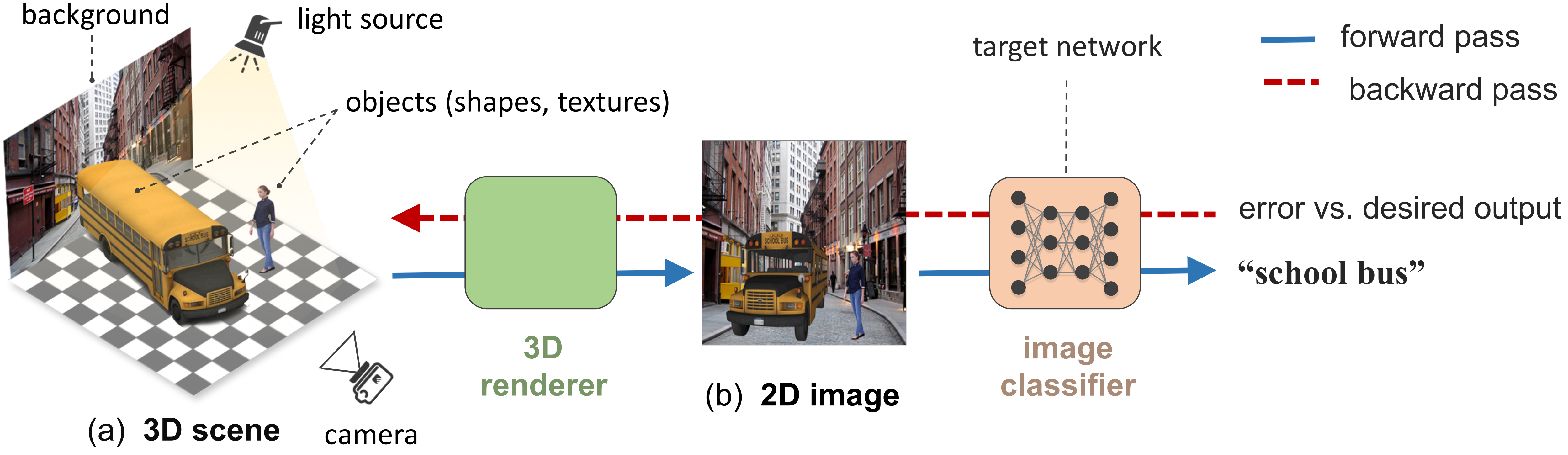


1. Pixel-wise

school bus



Fine-grained control over stimuli changes



Alcorn et al. 2019

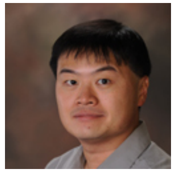
Alcorn

Qi Li

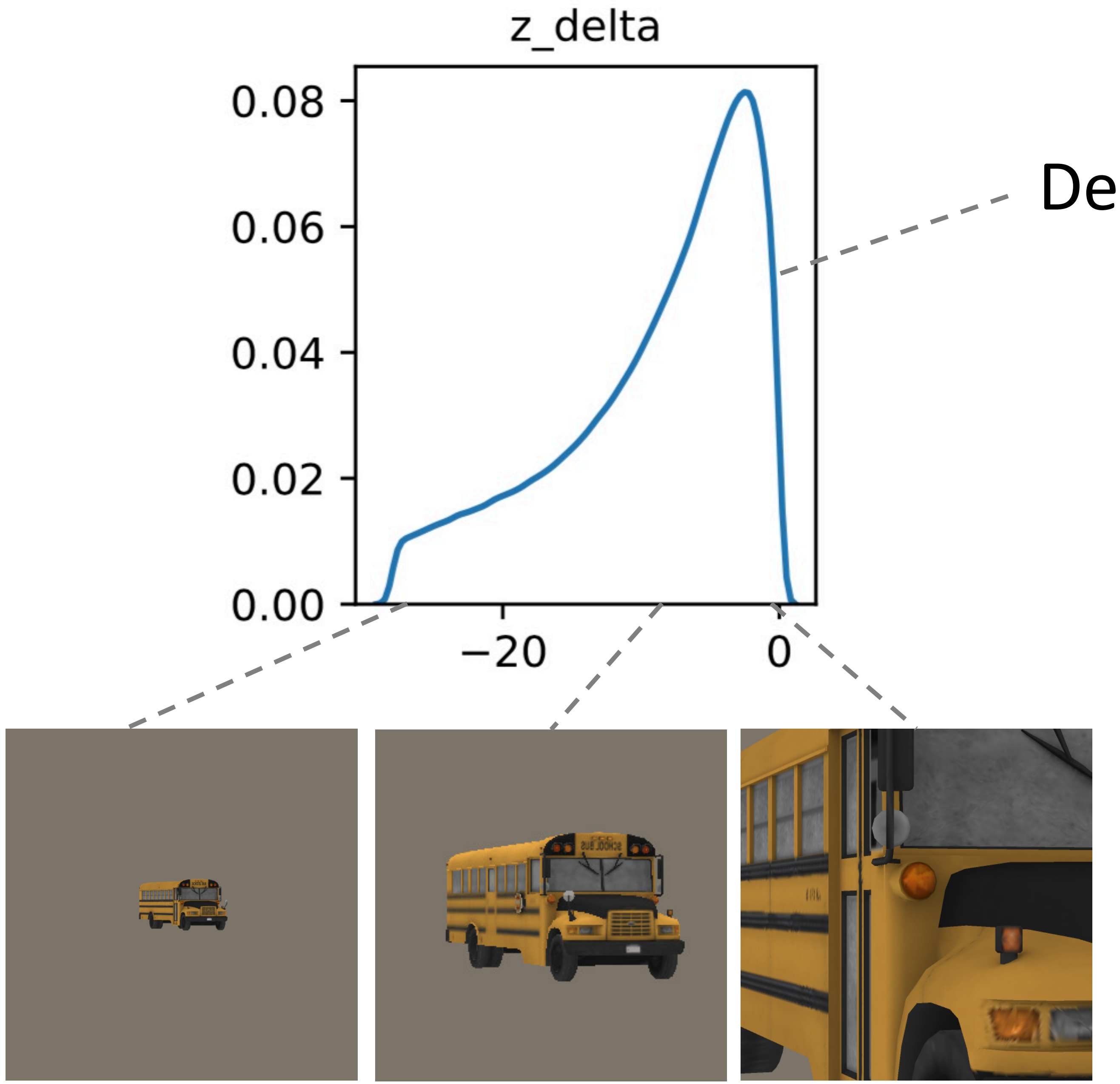
Gong

Wang

Long Mai



Jeff Ku



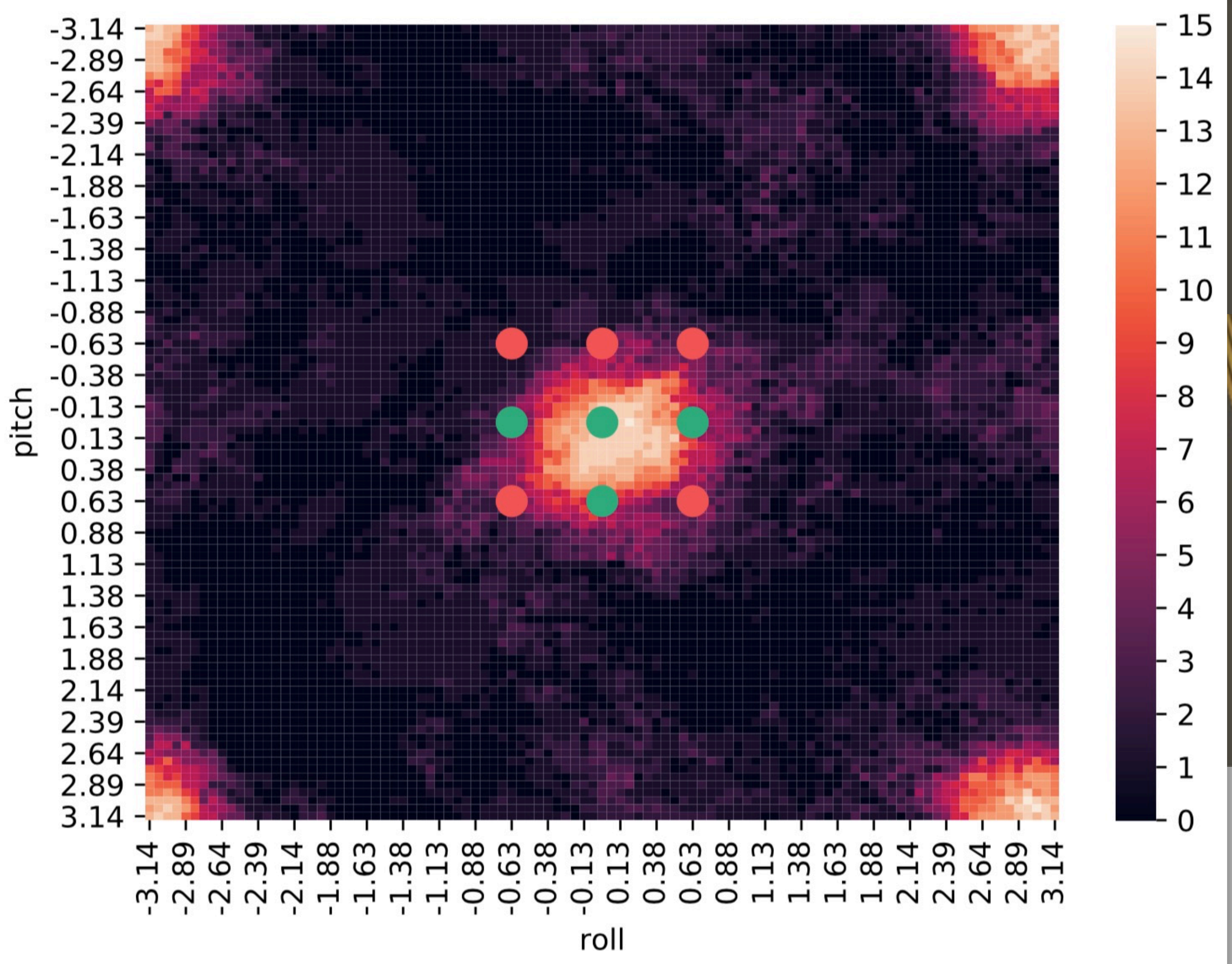
Density of **school bus** predictions

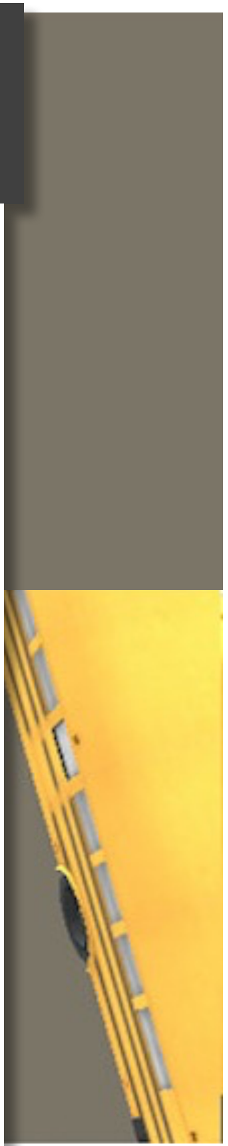
over

object-camera distances

Recognizes only 3% of the poses

99%





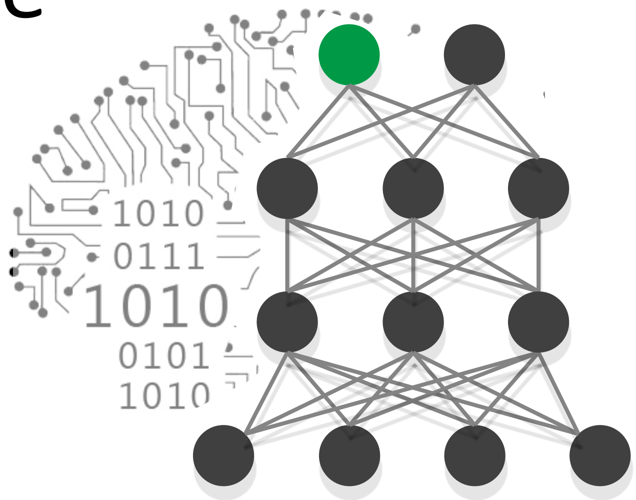
Alcorn et al. 2019

Finding what <u>artificial</u> neurons want to see

2. Image generator
3. 3D renderer

1. Pixel-wise

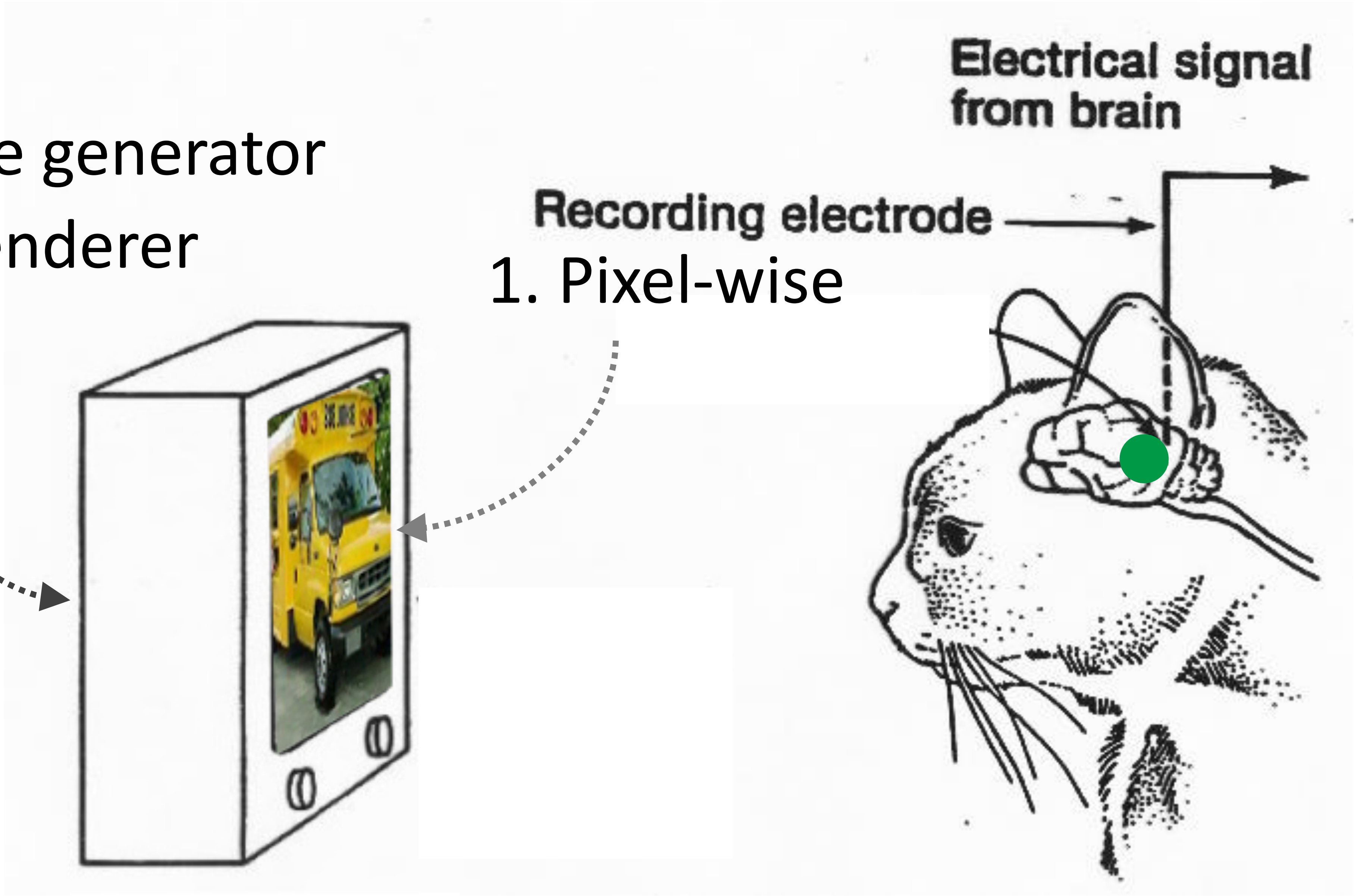
school bus



Finding what biological neurons want to see

2. Image generator
3. 3D renderer

1. Pixel-wise

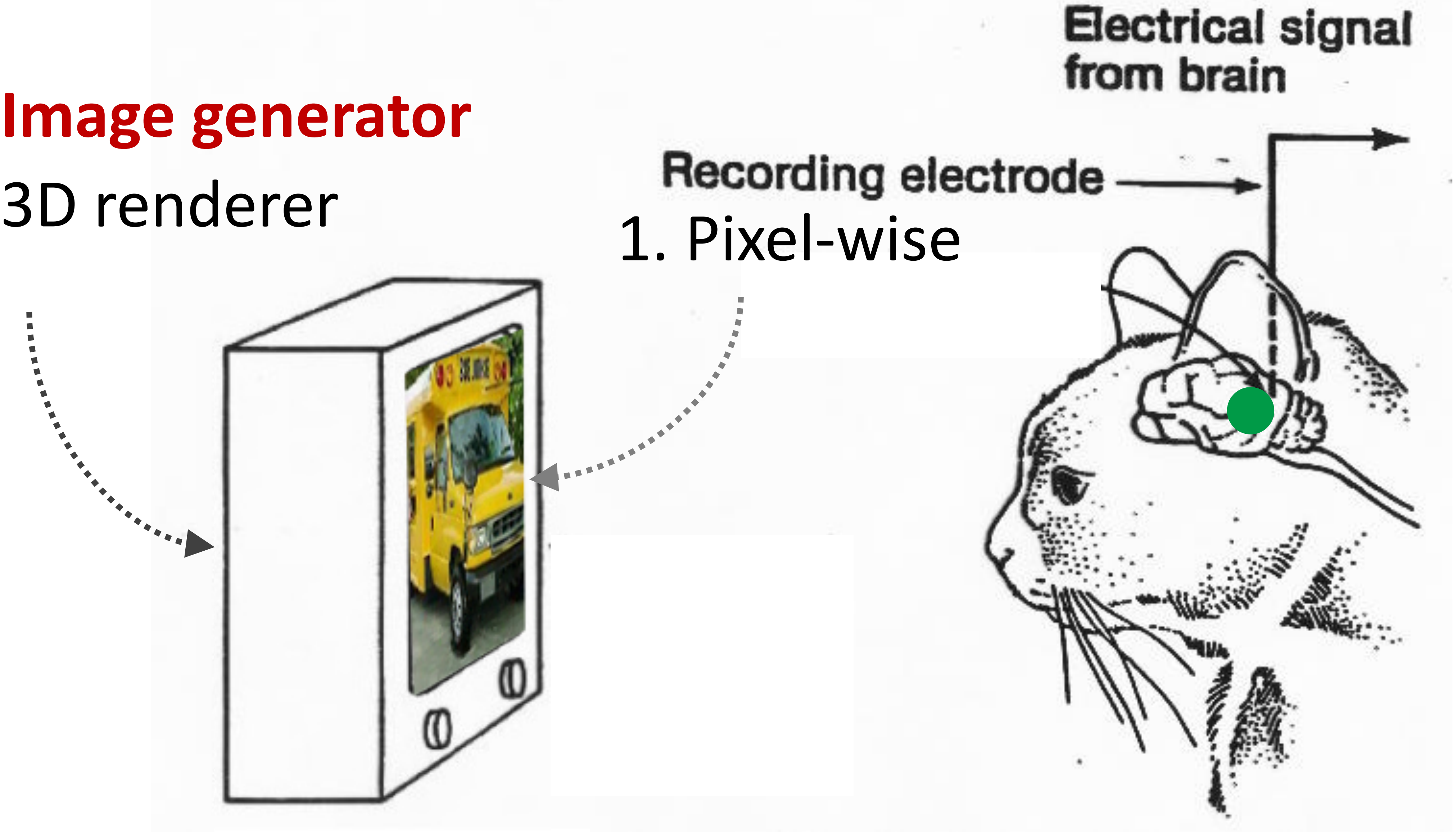


Finding what biological neurons want to see

2. Image generator

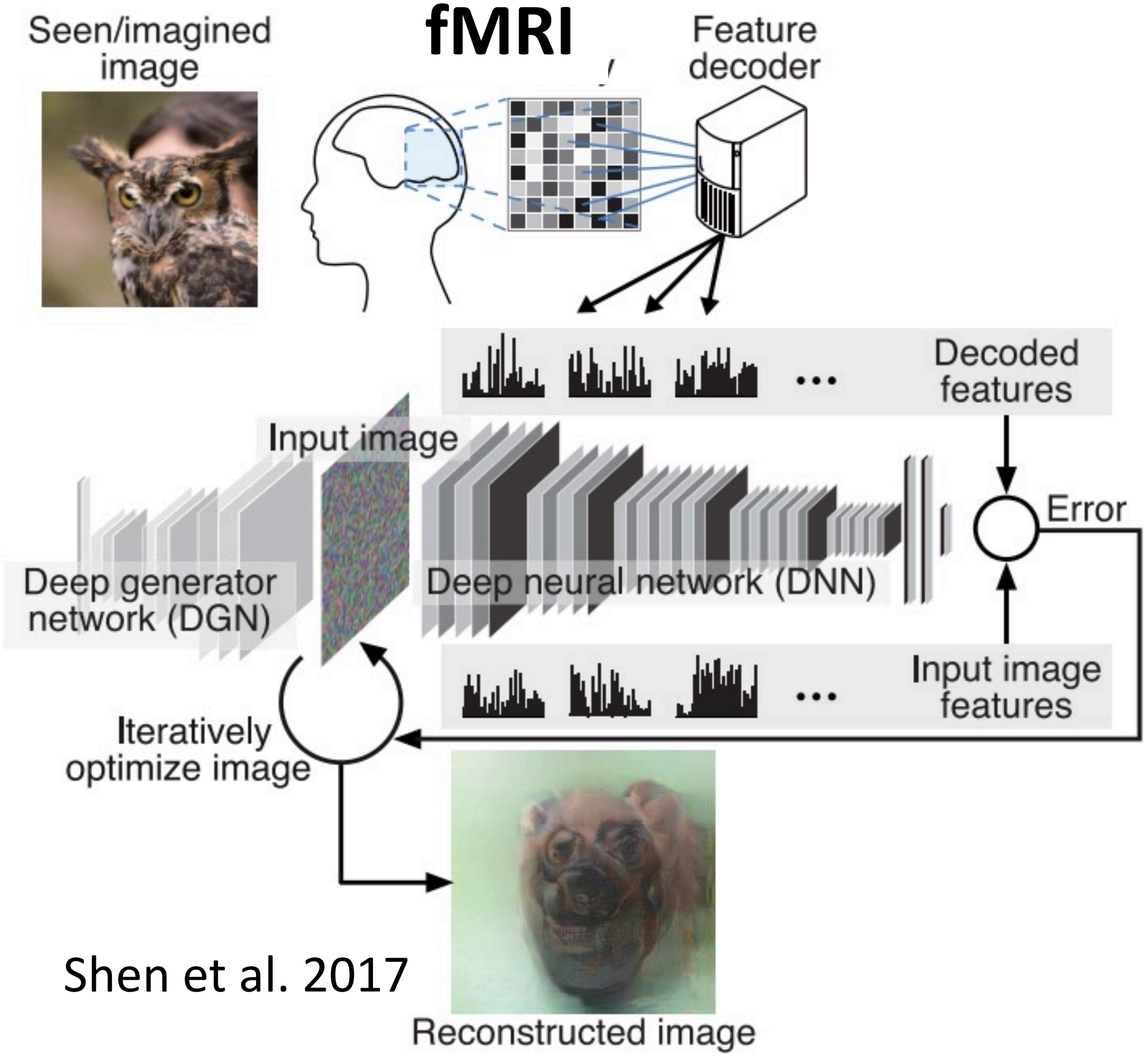
3. 3D renderer

1. Pixel-wise



Using image generators to decode **real brain signals**

fMRI



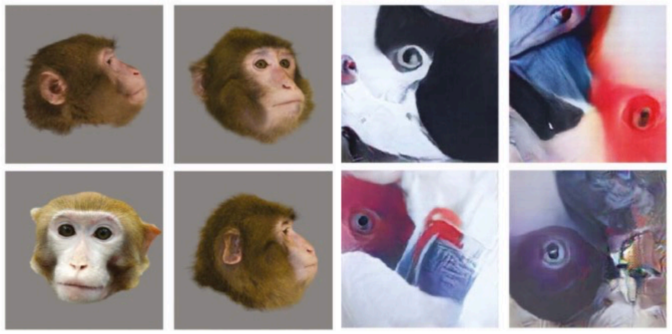
Shen et al. 2017

EEG signals

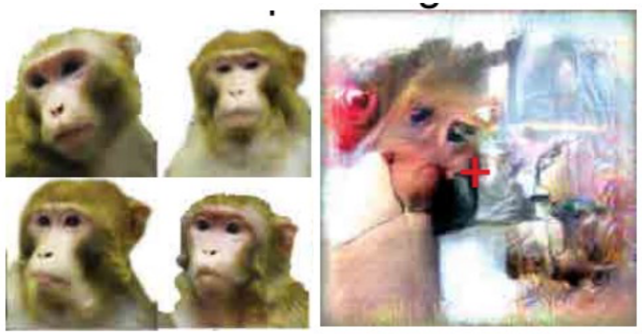
Palazzo et al. 2017

(b) Jack-o'-Lantern

Neural activations

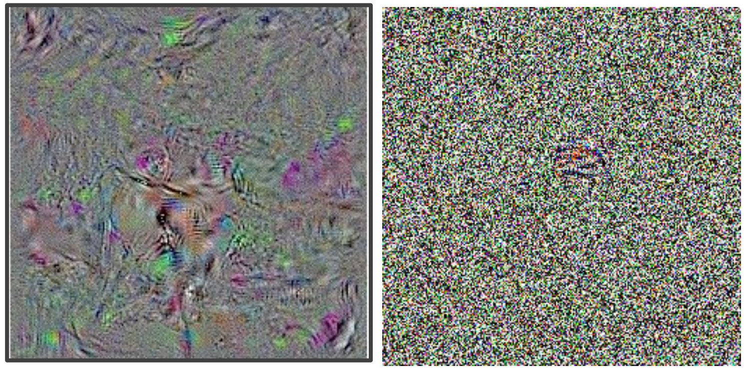


Malakhova 2018



Ponce, Xiao, Schade, et al. 2019

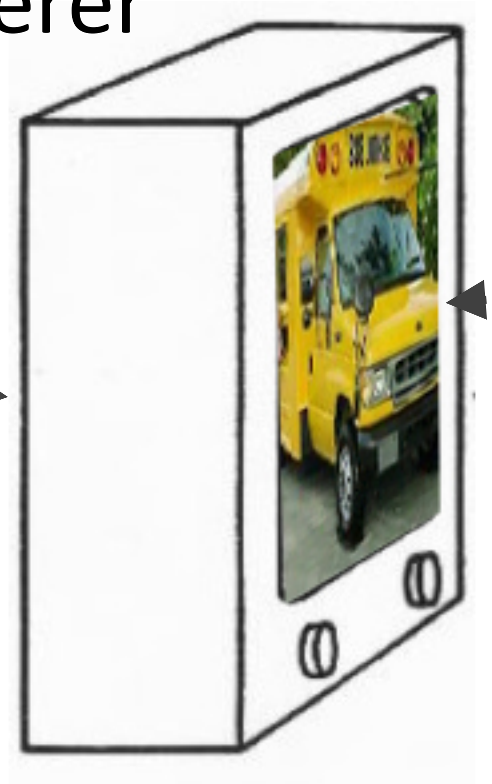
Thank you!



2. Image generator

3. 3D renderer

1. Pixel-wise



More info: http://AnhNguyen.me

AUBURN UNIVERSITY

References

- Nguyen, A., Yosinski, J., & Clune, J. (2015). Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 427-436).
- Nguyen, A., Dosovitskiy, A., Yosinski, J., Brox, T., & Clune, J. (2016). Synthesizing the preferred inputs for neurons in neural networks via deep generator networks. In *Advances in Neural Information Processing Systems* (pp. 3387-3395).
- Nguyen, A., Clune, J., Bengio, Y., Dosovitskiy, A., & Yosinski, J. (2017). Plug & play generative networks: Conditional iterative generation of images in latent space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 4467-4477).
- Alcorn, M. A., Li, Q., Gong, Z., Wang, C., Mai, L., Ku, W. S., & Nguyen, A. (2019). Strike (with) a pose: Neural networks are easily fooled by strange poses of familiar objects. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 4845-4854).
- Li, Q., Mai, L., & Nguyen, A. (2019). Improving sample diversity of a pre-trained, class-conditional GAN by changing its class embeddings. *arXiv preprint arXiv:1910.04760*.