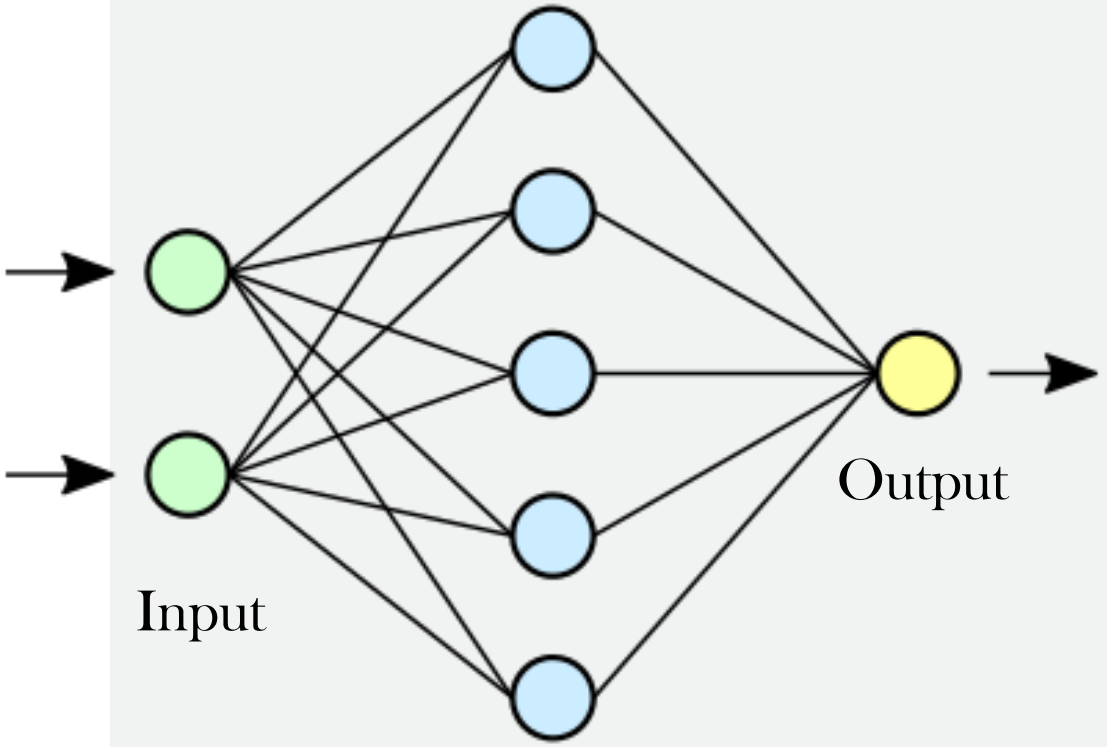
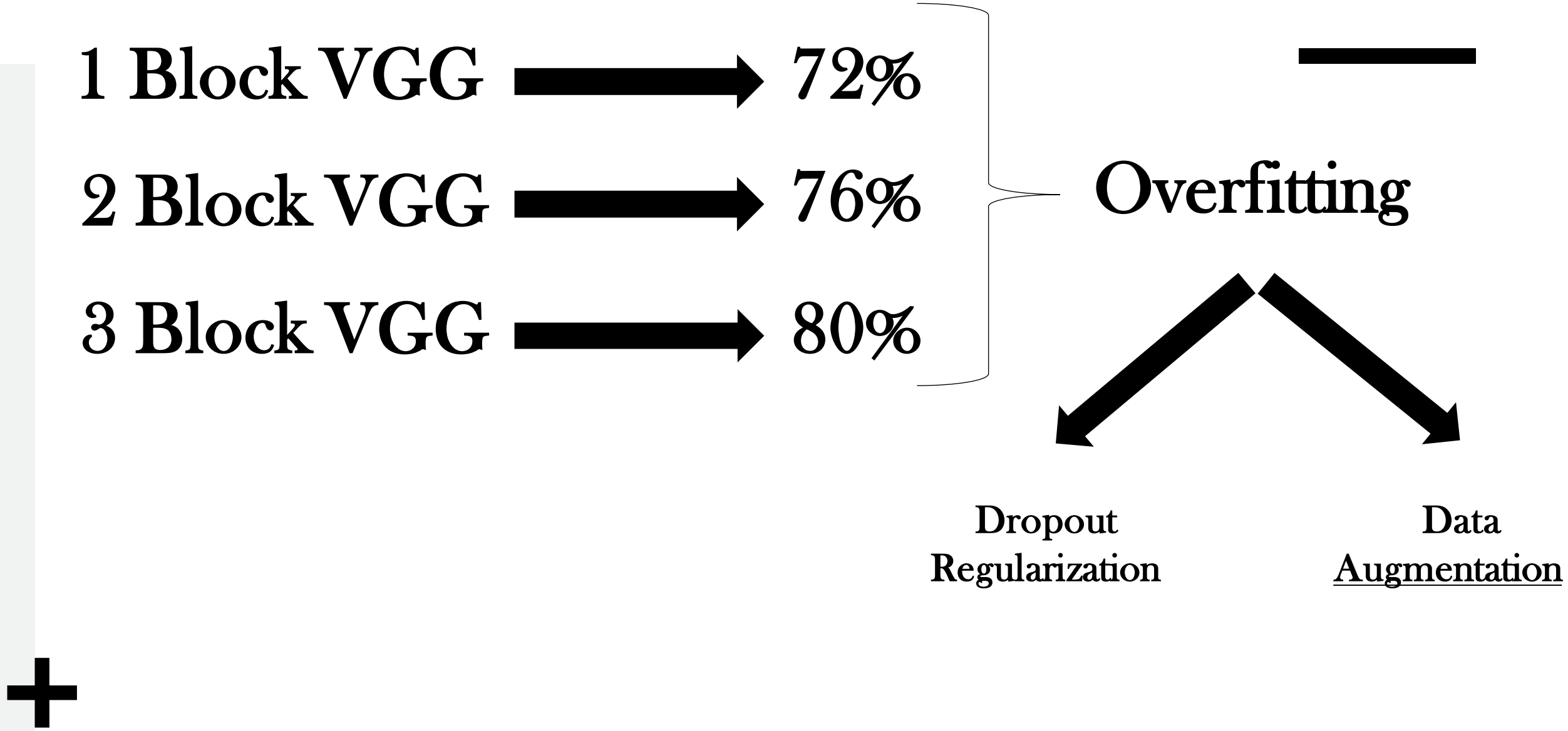
Neural Networks to Distinguish Images

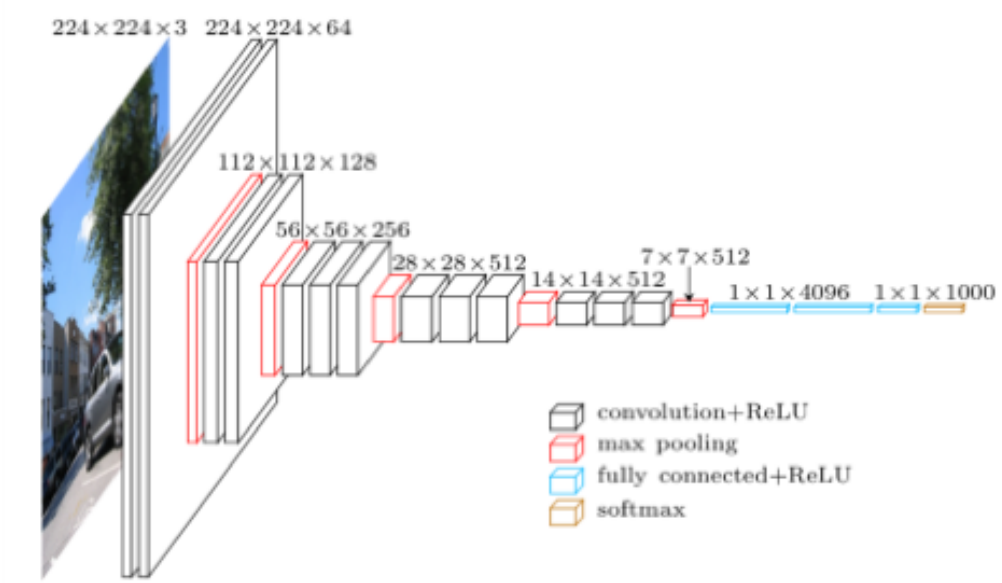
By Pedro Leite & Pedro Carvalho

Convolution Neural Networks



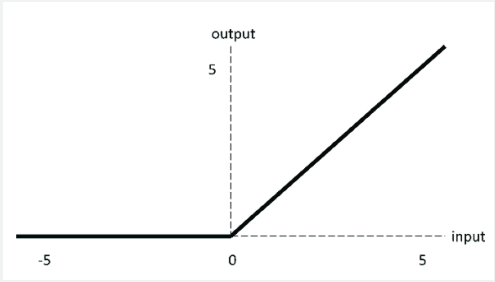

1 Block VGG
72%
2 Block VGG
76%
3 Block VGG
80%
Overfitting
Dropout Regularization
Data Augmentation

The VGG-16 Model

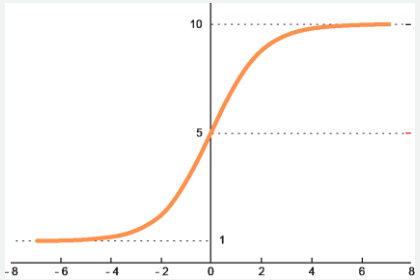



```
Epoch 1/5
293/293 [==============================] - 2748s 9s/step - loss: 0.2493 - accuracy: 0.9644 - val_loss: 0.0751 - val_accuracy: 0.9770
Epoch 2/5
293/293 [==============================] - 2747s 9s/step - loss: 0.0297 - accuracy: 0.9897 - val_loss: 0.0746 - val_accuracy: 0.9781
Epoch 3/5
293/293 [==============================] - 2750s 9s/step - loss: 0.0095 - accuracy: 0.9976 - val_loss: 0.0806 - val_accuracy: 0.9779
Epoch 4/5
293/293 [==============================] - 2749s 9s/step - loss: 0.0030 - accuracy: 0.9995 - val_loss: 0.0924 - val_accuracy: 0.9781
Epoch 5/5
293/293 [==============================] - 2721s 9s/step - loss: 0.0013 - accuracy: 0.9998 - val_loss: 0.1008 - val_accuracy: 0.9781
C:\Users\35191\Desktop\TAIA\tester.py:66: UserWarning: `Model.evaluate_generator` is deprecated and will be removed in a future version. Please use `Model.evaluate`, which supports generators.
  _, acc = model.evaluate_generator(test_it, steps=len(test_it), verbose=0)
> 97.811
```

- 13 convulation layers.
- 3 connected layers.
- 1 max pooling layer.



ReLu

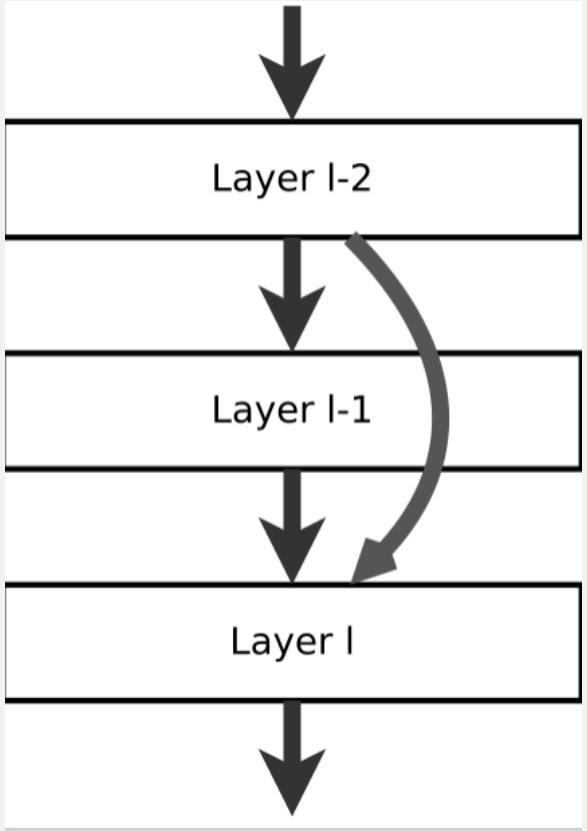


SoftMax

The Resnet Model

- 48 convulational layers.
- 1 max pooling layer.
- 1 average pooling layer.

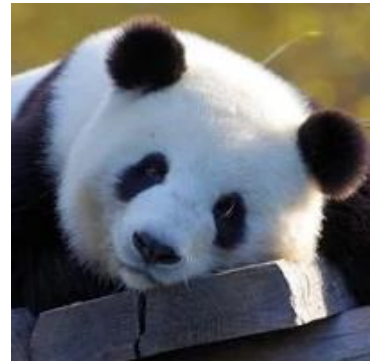
```
Epoch 1/5
625/625 [==============================] - 1057s 2s/step - loss: 0.0996 - accuracy: 0.9650 - val_loss: 0.0754 - val_accuracy: 0.9696
Epoch 2/5
625/625 [==============================] - 1057s 2s/step - loss: 0.0546 - accuracy: 0.9793 - val_loss: 0.0729 - val_accuracy: 0.9714
Epoch 3/5
625/625 [==============================] - 1056s 2s/step - loss: 0.0428 - accuracy: 0.9833 - val_loss: 0.0806 - val_accuracy: 0.9680
Epoch 4/5
625/625 [==============================] - 1058s 2s/step - loss: 0.0356 - accuracy: 0.9872 - val_loss: 0.0934 - val_accuracy: 0.9658
Epoch 5/5
625/625 [==============================] - 1056s 2s/step - loss: 0.0272 - accuracy: 0.9897 - val_loss: 0.0787 - val_accuracy: 0.9728
```



Adding Noisy Pictures

```
Epoch 1/5
625/625 [==============================] - 1057s 2s/step - loss: 0.0996 - accuracy: 0.9650 - val_loss: 0.0754 - val_accuracy: 0.9696
Epoch 2/5
625/625 [==============================] - 1057s 2s/step - loss: 0.0546 - accuracy: 0.9793 - val_loss: 0.0729 - val_accuracy: 0.9714
Epoch 3/5
625/625 [==============================] - 1056s 2s/step - loss: 0.0428 - accuracy: 0.9833 - val_loss: 0.0806 - val_accuracy: 0.9680
Epoch 4/5
625/625 [==============================] - 1058s 2s/step - loss: 0.0356 - accuracy: 0.9872 - val_loss: 0.0934 - val_accuracy: 0.9658
Epoch 5/5
625/625 [==============================] - 1056s 2s/step - loss: 0.0272 - accuracy: 0.9897 - val_loss: 0.0787 - val_accuracy: 0.9728
```

Adding Another Class



```
Epoch 1/5
293/293 [==============================] - 2641s 9s/step - loss: 0.0334 - accuracy: 0.9413 - val_loss: -1.6923 - val_accuracy: 0.9650
Epoch 2/5
293/293 [==============================] - 2626s 9s/step - loss: -26.3232 - accuracy: 0.9559 - val_loss: -587.1599 - val_accuracy: 0.9515
Epoch 3/5
293/293 [==============================] - 2627s 9s/step - loss: -6792.3960 - accuracy: 0.9465 - val_loss: -124492.6875 - val_accuracy: 0.9204
Epoch 4/5
293/293 [==============================] - 2630s 9s/step - loss: -1962060.7500 - accuracy: 0.9553 - val_loss: -35747860.0000 - val_accuracy: 0.9526
Epoch 5/5
293/293 [==============================] - 2628s 9s/step - loss: -467379904.0000 - accuracy: 0.9420 - val_loss: -10380189696.0000 - val_accuracy: 0.9305
C:\Users\35191\Desktop\TAIA\tester.py:66: UserWarning: `Model.evaluate_generator` is deprecated and will be removed in a future version. Please use `Model.evaluate`, which supports generators.
  _, acc = model.evaluate_generator(test_it, steps=len(test_it), verbose=0)
> 93.051
```

Using Image Description to Train the Model

	A	B	C	D	E	F	G	H	I	J	K	L
1	Id,Subject Focus,Eyes,Face,Near,Action,Accessory,Group,Collage,Human,Occlusion,Info,Blur,Pawpularity											
2	0007de18844b0dbbb5e1f607da0606e0,0,1,1,1,0,0,1,0,0,0,0,0,63											
3	0009c66b9439883ba2750fb825e1d7db,0,1,1,0,0,0,0,0,0,0,0,0,42											
4	0013fd999caf9a3efe1352ca1b0d937e,0,1,1,1,0,0,0,0,1,1,0,0,28											
5	0018df346ac9c1d8413cfcc888ca8246,0,1,1,1,0,0,0,0,0,0,0,0,15											
6	001dc955e10590d3ca4673f034feeef2,0,0,0,1,0,0,1,0,0,0,0,0,72											
7	001dd4f6fafb890610b1635f967ea081,0,0,1,0,0,0,0,0,0,0,0,1,74											
8	0023b8a3abc93c712edd6120867deb53,0,1,1,1,0,0,0,0,1,1,0,0,22											
9	0031d6a9ef7340f898c3e05f92c7bb04,0,1,1,0,0,0,1,1,0,0,1,0,35											
10	0042bc5bada6d1cf8951f8f9f0d399fa,0,1,1,1,0,0,0,0,0,0,0,0,53											
11	0049cb81313c94fa007286e9039af910,0,1,1,1,0,0,0,0,0,0,0,0,21											
12	005017716086b8d5e118dd9fe26459b1,0,1,1,1,0,0,0,0,0,0,0,0,28											
13	00524dbf2637a80cbc80f70d3ff59616,0,1,1,1,0,0,0,0,0,0,0,0,2											
14	00630b1262efe301cb15a3b2022ba744,0,1,1,1,0,0,0,0,0,0,0,0,18											
15	006483b96ca9c09b7afed3e3d3af539d,0,1,1,1,0,0,0,0,1,1,0,0,41											
16	00655425c10d4c082dd7eeb97fa4fb17,0,1,0,0,0,1,0,0,0,0,0,0,13											
17	0067aaaa500b530c76b9c91af34b4cb8,0,1,1,1,0,0,0,0,0,0,0,0,98											
18	006cda7fec46a527f9f627f4722a2304,0,0,0,1,0,0,0,0,0,0,0,1,34											
19	006fe962f5f7e2c5f527b2e27e28ed6d,0,1,1,0,0,0,0,1,1,1,0,0,69											
20	0075ec6503412f21cf65ac5f43d80440,0,0,1,1,0,0,0,0,0,0,0,0,53											
21	00768659c1c90409f81dcdecbd270513,0,1,1,0,0,0,0,0,0,0,0,0,100											
22	007c6ea0ccfca08d4736e5f459ed274a,0,1,1,1,0,0,0,1,0,0,0,0,29											
23	00838b02764e8d37edde39ed18693427,0,0,0,1,0,0,1,0,0,0,0,0,27											
24	0084a12f432ef6e47e8b9ff04d8a32b4,0,1,1,1,0,0,0,0,0,0,0,0,28											
25	0085bfb9a7ebfd776cb804d8b456bb05,0,1,1,1,0,0,0,0,0,0,0,0,50											
26	009367718f506d87cd898ed0310a2392,0,1,1,1,0,0,0,0,0,0,0,0,36											
27	0095f81bab3b68a4f70e99f0fcec7b06,0,0,1,1,0,0,0,0,1,0,0,1,15											
28	00a1ae8867e0bb89f061679e1cf29e80,0,1,1,0,0,0,1,0,0,0,0,0,52											
29	00a1e0c0f89ff89a8f32d42e9025f6b2,0,1,1,1,0,0,0,0,0,0,0,0,40											
30	00a6d367daed96f77be6c91db493dfed,0,1,1,0,0,0,0,0,0,0,0,0,29											
31	00b151a572c9aabedf8cfce0fa18be25,0,0,1,1,0,0,0,0,0,0,0,1,21											
32	00c7b596627da7425b624d6b16c88a25,0,1,1,1,0,0,0,0,0,0,0,0,34											
33	00d1cb2ec8b263ae076ff95cae513a88,0,1,1,1,0,0,0,0,0,0,0,0,31											

ID: 007de18844b0ddbbb5e1f607da0606e0
Subject Focus: 0
Eyes: 0
Face: 1
...
Popularity: 63%

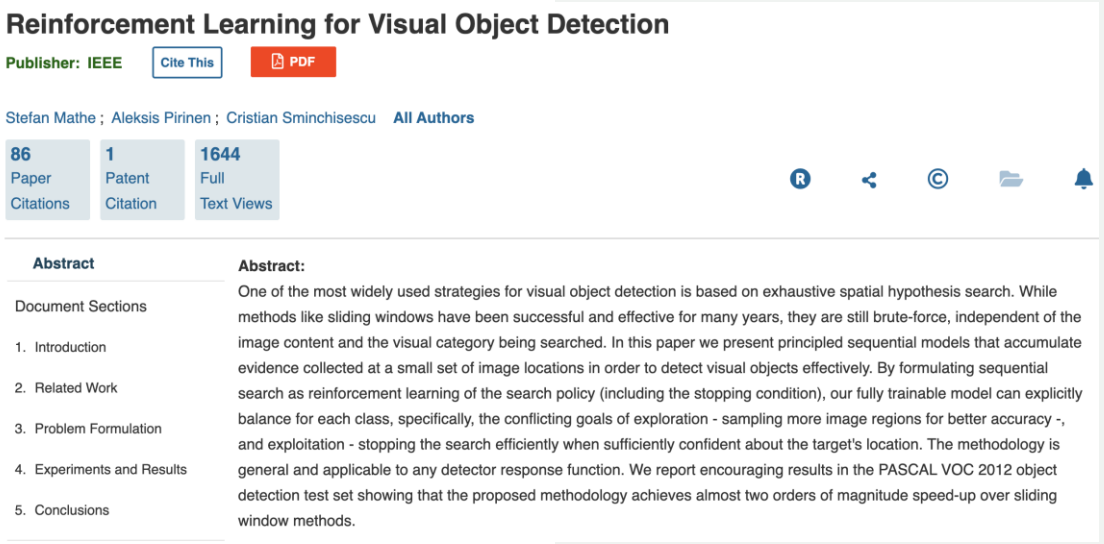
Reinforcement Learning

Image Classification by Reinforcement Learning with Two-State Q-Learning

Abdul Mueed Hafiz[1]

[1] Department of Electronics and Communication Engineering
Institute of Technology, University of Kashmir
Srinagar, J&K, India, 190006

ORC-ID: 0000-0002-2266-3708



Reinforcement Learning for Visual Object Detection

Publisher: IEEE | Cite This | PDF

Stefan Mathe ; Aleksis Pirinen ; Cristian Sminchisescu All Authors

86 Paper Citations | 1 Patent Citation | 1644 Full Text Views

Abstract

Document Sections

1. Introduction
2. Related Work
3. Problem Formulation
4. Experiments and Results
5. Conclusions

Abstract:
One of the most widely used strategies for visual object detection is based on exhaustive spatial hypothesis search. While methods like sliding windows have been successful and effective for many years, they are still brute-force, independent of the image content and the visual category being searched. In this paper we present principled sequential models that accumulate evidence collected at a small set of image locations in order to detect visual objects effectively. By formulating sequential search as reinforcement learning of the search policy (including the stopping condition), our fully trainable model can explicitly balance for each class, specifically, the conflicting goals of exploration - sampling more image regions for better accuracy -, and exploitation - stopping the search efficiently when sufficiently confident about the target's location. The methodology is general and applicable to any detector response function. We report encouraging results in the PASCAL VOC 2012 object detection test set showing that the proposed methodology achieves almost two orders of magnitude speed-up over sliding window methods.