

Report HW7
Robotics Multimodal Deep Learning for Object
Recognition

Giorgio Giannone

August 1, 2018

Introduction

Our goal is to implement a Network to detect 15 class objects, fusing RGB and Depth information.

Architecture

We build a simple Convolutional Neural Network to solve this problem. In particular the structure is:

$$\begin{aligned} Conv(64, (5, 5)) \rightarrow MaxPool((3, 3), (2, 2)) \rightarrow Conv(64, (5, 5)) \rightarrow MaxPool((3, 3), (2, 2)) \rightarrow \\ Flatten() \rightarrow Dense(384) \rightarrow Dense(192) \rightarrow Dense(15) \end{aligned}$$

where $Conv(f, (k, k))$ is a convolutional layer with f filters and kernel size k . $MaxPool((r, r), (s, s))$ is a downsampling layer with downsampling rate r and stride s . $Dense(g)$ is a fully connected layer with g number of units.

Result

We report the results using only RGB (Fig.1), only Depth (Fig.2) and both modalities (Fig.3). We selected an image resolution of (48, 48) and we trained for 100 epochs to speed-up the training procedure.

Discussion

In a multimodal setting we expect to improve the metric using both modalities. In this case we improve over the RGB modality. But we obtain an unexpected result for the depth modality: this can be explained considering that the network is not able to fuse completely the two modalities.

```
Epoch 90 of 100 took 1.898329s
  train loss: 0.000004
  train acc: 1.000000
  val loss: 2.690682
  val acc: 0.764000
Epoch 90 of 100 took 3.059949s, loss 0.000004
Epoch 91 of 100 took 1.897033s, loss 0.000004
Epoch 92 of 100 took 1.899250s, loss 0.000004
Epoch 93 of 100 took 1.913825s, loss 0.000004
Epoch 94 of 100 took 1.922813s, loss 0.000004
Epoch 95 of 100 took 1.900828s, loss 0.000004
Epoch 96 of 100 took 1.894816s, loss 0.000003
Epoch 97 of 100 took 1.886841s, loss 0.000003
Epoch 98 of 100 took 1.889400s, loss 0.000003
Epoch 99 of 100 took 1.903993s, loss 0.000003
Epoch 100 of 100 took 1.892466s
  train loss: 0.000003
  train acc: 1.000000
  val loss: 2.729429
  val acc: 0.767667
Epoch 100 of 100 took 3.112134s, loss 0.000003
Total training time: 224.343680s
```

Figure 1: result using only RGB

```
Epoch 90 of 100 took 1.856027s
  train loss: 0.000019
  train acc: 1.000000
  val loss: 0.821696
  val acc: 0.882667
Epoch 90 of 100 took 2.924667s, loss 0.000020
Epoch 91 of 100 took 1.889924s, loss 0.000019
Epoch 92 of 100 took 1.818671s, loss 0.000018
Epoch 93 of 100 took 1.842571s, loss 0.000018
Epoch 94 of 100 took 1.862535s, loss 0.000016
Epoch 95 of 100 took 1.844655s, loss 0.000016
Epoch 96 of 100 took 1.896322s, loss 0.000015
Epoch 97 of 100 took 1.872103s, loss 0.000015
Epoch 98 of 100 took 1.846315s, loss 0.000014
Epoch 99 of 100 took 1.882488s, loss 0.000014
Epoch 100 of 100 took 1.877421s
  train loss: 0.000012
  train acc: 1.000000
  val loss: 0.815014
  val acc: 0.884333
Epoch 100 of 100 took 2.943821s, loss 0.000013
Total training time: 200.161965s
```

Figure 2: result using only Depth

```
Epoch 90 of 100 took 3.378668s
  train loss: 0.000000
  train acc: 1.000000
  val loss: 3.811879
  val acc: 0.839000
Epoch 90 of 100 took 5.396716s, loss 0.000000
Epoch 91 of 100 took 3.421786s, loss 0.000000
Epoch 92 of 100 took 3.404169s, loss 0.000000
Epoch 93 of 100 took 3.402133s, loss 0.000000
Epoch 94 of 100 took 3.404811s, loss 0.000000
Epoch 95 of 100 took 3.408915s, loss 0.000000
Epoch 96 of 100 took 3.456902s, loss 0.000000
Epoch 97 of 100 took 3.445029s, loss 0.000000
Epoch 98 of 100 took 3.414377s, loss 0.000000
Epoch 99 of 100 took 3.424299s, loss 0.000000
Epoch 100 of 100 took 3.429078s
  train loss: 0.000000
  train acc: 1.000000
  val loss: 3.819808
  val acc: 0.840000
Epoch 100 of 100 took 5.509321s, loss 0.000000
Total training time: 377.678986s
```

Figure 3: result using RGB and Depth