toy_problem1

November 17, 2020

[1]:
```
from __future__ import division, print_function
%matplotlib inline
import matplotlib.pyplot as plt
import matplotlib
import numpy as np
plt.rcParams['image.cmap'] = 'gist_earth'
np.random.seed(98765)
```

[2]:
```
from tf_unet import image_gen
from tf_unet import unet
from tf_unet import util
```

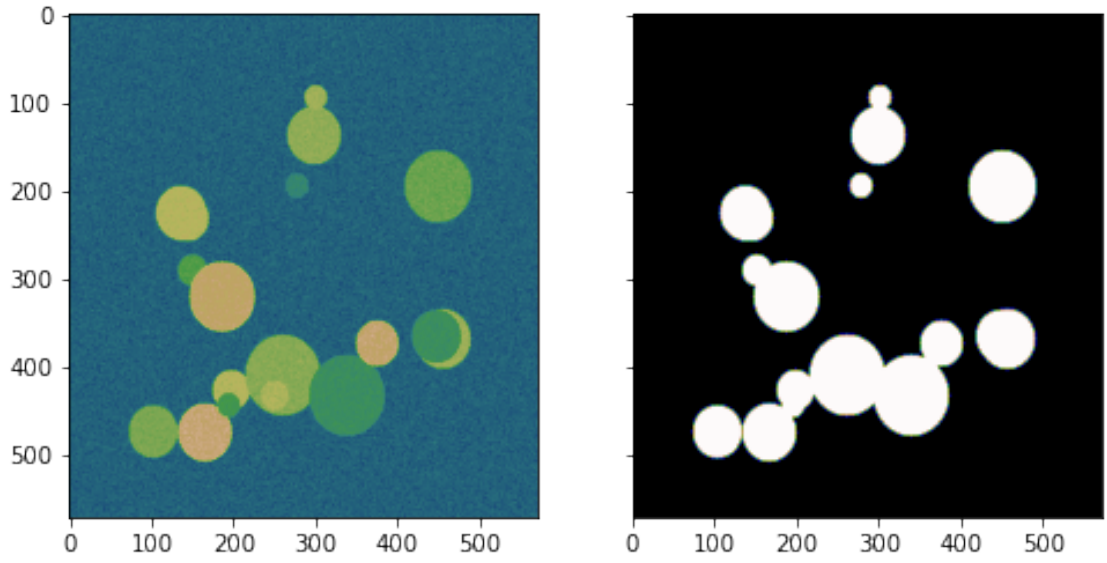
[3]:
```
nx = 572
ny = 572
```

[4]:
```
generator = image_gen.GrayScaleDataProvider(nx, ny, cnt=20)
```

[5]:
```
x_test, y_test = generator(1)
```

[6]:
```
fig, ax = plt.subplots(1,2, sharey=True, figsize=(8,4))
ax[0].imshow(x_test[0,...,0], aspect="auto")
ax[1].imshow(y_test[0,...,1], aspect="auto")
```

[6]:
```
<matplotlib.image.AxesImage at 0x7ffad9d4f130>
```



[7]:
```
import tensorflow.compat.v1 as tf
tf.disable_v2_behavior()
```

```
WARNING:tensorflow:From /home/royal/anaconda3/lib/python3.8/site-
packages/tensorflow/python/compat/v2_compat.py:96: disable_resource_variables
(from tensorflow.python.ops.variable_scope) is deprecated and will be removed in
a future version.
Instructions for updating:
non-resource variables are not supported in the long term

2020-11-17 08:49:19,595 From /home/royal/anaconda3/lib/python3.8/site-
packages/tensorflow/python/compat/v2_compat.py:96: disable_resource_variables
(from tensorflow.python.ops.variable_scope) is deprecated and will be removed in
a future version.
Instructions for updating:
non-resource variables are not supported in the long term
```

[8]:
```
net = unet.Unet(channels=generator.channels, n_class=generator.n_class,␣
↪layers=3, features_root=16)
```

```
2020-11-17 08:49:19,610 Layers 3, features 16, filter size 3x3, pool size: 2x2
```

[9]:
```
trainer = unet.Trainer(net, optimizer="momentum", opt_kwargs=dict(momentum=0.2))
```

[10]:
```
path = trainer.train(generator, "./unet_trained", training_iters=32, epochs=10,␣
↪display_step=2)
```

```
2020-11-17 08:49:21,398 Removing '/home/royal/Desktop/UNET/demo/prediction'
2020-11-17 08:49:21,399 Removing '/home/royal/Desktop/UNET/demo/unet_trained'
```

```
2020-11-17 08:49:21,399 Allocating '/home/royal/Desktop/UNET/demo/prediction'
2020-11-17 08:49:21,400 Allocating '/home/royal/Desktop/UNET/demo/unet_trained'
2020-11-17 08:49:24,816 Verification error= 16.4%, loss= nan
/home/royal/.local/lib/python3.8/site-
packages/tf_unet-0.1.2-py3.8.egg/tf_unet/util.py:77: RuntimeWarning: invalid
value encountered in true_divide
  img /= np.amax(img)
2020-11-17 08:49:25,416 Start optimization


      ␣
↪---------------------------------------------------------------------------

       TypeError                                 Traceback (most recent call␣
↪last)

       <ipython-input-10-20d861088f45> in <module>
   ----> 1 path = trainer.train(generator, "./unet_trained", training_iters=32,␣
↪epochs=10, display_step=2)


       ~/.local/lib/python3.8/site-packages/tf_unet-0.1.2-py3.8.egg/tf_unet/
↪unet.py in train(self, data_provider, output_path, training_iters, epochs,␣
↪dropout, display_step, restore, write_graph, prediction_path)
       447
       448                     if step % display_step == 0:
   --> 449                         self.output_minibatch_stats(sess,␣
↪summary_writer, step, batch_x,
       450                                                     util.
↪crop_to_shape(batch_y, pred_shape))
       451


       ~/.local/lib/python3.8/site-packages/tf_unet-0.1.2-py3.8.egg/tf_unet/
↪unet.py in output_minibatch_stats(self, sess, summary_writer, step, batch_x,␣
↪batch_y)
       486     def output_minibatch_stats(self, sess, summary_writer, step,␣
↪batch_x, batch_y):
       487         # Calculate batch loss and accuracy
   --> 488         summary_str, loss, acc, predictions = sess.run([self.
↪summary_op,
       489                                                         self.net.
↪cost,
       490                                                         self.net.
↪accuracy,
```

```
      ~/anaconda3/lib/python3.8/site-packages/tensorflow/python/client/session.
↪py in run(self, fetches, feed_dict, options, run_metadata)
      955
      956     try:
  --> 957       result = self._run(None, fetches, feed_dict, options_ptr,
      958                          run_metadata_ptr)
      959       if run_metadata:


      ~/anaconda3/lib/python3.8/site-packages/tensorflow/python/client/session.
↪py in _run(self, handle, fetches, feed_dict, options, run_metadata)
     1163
     1164     # Create a fetch handler to take care of the structure of␣
↪fetches.
  -> 1165     fetch_handler = _FetchHandler(
     1166         self._graph, fetches, feed_dict_tensor,␣
↪feed_handles=feed_handles)
     1167


      ~/anaconda3/lib/python3.8/site-packages/tensorflow/python/client/session.
↪py in __init__(self, graph, fetches, feeds, feed_handles)
      475     """
      476     with graph.as_default():
  --> 477       self._fetch_mapper = _FetchMapper.for_fetch(fetches)
      478     self._fetches = []
      479     self._targets = []


      ~/anaconda3/lib/python3.8/site-packages/tensorflow/python/client/session.
↪py in for_fetch(fetch)
      264     elif isinstance(fetch, (list, tuple)):
      265       # NOTE(touts): This is also the code path for namedtuples.
  --> 266       return _ListFetchMapper(fetch)
      267     elif isinstance(fetch, collections_abc.Mapping):
      268       return _DictFetchMapper(fetch)


      ~/anaconda3/lib/python3.8/site-packages/tensorflow/python/client/session.
↪py in __init__(self, fetches)
      376     else:
      377       self._fetch_type = type(fetches)
  --> 378     self._mappers = [_FetchMapper.for_fetch(fetch) for fetch in␣
↪fetches]
      379     self._unique_fetches, self._value_indices =␣
↪_uniquify_fetches(self._mappers)
      380
```

```
       ~/anaconda3/lib/python3.8/site-packages/tensorflow/python/client/session.
↪py in <listcomp>(.0)
       376     else:
       377       self._fetch_type = type(fetches)
   --> 378     self._mappers = [_FetchMapper.for_fetch(fetch) for fetch in␣
↪fetches]
       379     self._unique_fetches, self._value_indices =␣
↪_uniquify_fetches(self._mappers)
       380

       ~/anaconda3/lib/python3.8/site-packages/tensorflow/python/client/session.
↪py in for_fetch(fetch)
       260     """
       261     if fetch is None:
   --> 262       raise TypeError('Fetch argument %r has invalid type %r' %
       263                       (fetch, type(fetch)))
       264     elif isinstance(fetch, (list, tuple)):

       TypeError: Fetch argument None has invalid type <class 'NoneType'>
```

[ ]:

[ ]: