malaria

June 18, 2024

1 Libraries

```python
[12]: # file operations
      import os
      # to list files
      import glob

      # for numerical analysis
      import numpy as np
      # to store and process in a dataframe
      import pandas as pd

      # for ploting graphs
      import matplotlib.pyplot as plt
      # advancec ploting
      import seaborn as sns

      # image processing
      import matplotlib.image as mpimg

      # train test split
      from sklearn.model_selection import train_test_split
      # model performance metrics
      from sklearn.metrics import confusion_matrix, classification_report

      # utility functions
      from tensorflow.keras.utils import to_categorical, plot_model
      # process image
      from tensorflow.keras.preprocessing.image import ImageDataGenerator, load_img
      # sequential model
      from tensorflow.keras.models import Sequential
      # layers
      from tensorflow.keras.layers import Conv2D, MaxPooling2D, Dense, Flatten,␣
       ↪Dropout
      # callback functions
      from tensorflow.keras.callbacks import EarlyStopping, ModelCheckpoint,␣
       ↪LearningRateScheduler
```

2 Data

2.0.1 List files

```
[13]: # list of files in the dataset
      os.listdir('../input/cell-images-for-detecting-malaria/cell_images/cell_images')
```

```
[13]: ['Uninfected', 'Parasitized']
```

```
[14]: # list all the images in the directory Parasitized
      parasitized = glob.glob('../input/cell-images-for-detecting-malaria/cell_images/
       ↪cell_images/Parasitized/*.png')

      # no. of files in the directory Parasitized
      print('No. of files in the directory Parasitized', len(parasitized))

      # first few images
      parasitized[:5]
```

No. of files in the directory Parasitized 13779

```
[14]: ['../input/cell-images-for-detecting-malaria/cell_images/cell_images/Parasitized
      /C99P60ThinF_IMG_20150918_141001_cell_93.png',
       '../input/cell-images-for-detecting-malaria/cell_images/cell_images/Parasitized
      /C99P60ThinF_IMG_20150918_141001_cell_133.png',
       '../input/cell-images-for-detecting-malaria/cell_images/cell_images/Parasitized
      /C101P62ThinF_IMG_20150918_151942_cell_60.png',
       '../input/cell-images-for-detecting-malaria/cell_images/cell_images/Parasitized
      /C162P123ThinF_IMG_20151116_102655_cell_163.png',
       '../input/cell-images-for-detecting-malaria/cell_images/cell_images/Parasitized
      /C52P13thinF_IMG_20150725_124830_cell_174.png']
```

```
[15]: # list all the images in the directory Uninfected
      uninfected = glob.glob('../input/cell-images-for-detecting-malaria/cell_images/
       ↪cell_images/Uninfected/*.png')

      # no. of files in the directory Uninfected
      print('No. of files in the directory Uninfected', len(uninfected))

      # first few images
      uninfected[:5]
```

No. of files in the directory Uninfected 13779

```
[15]: ['../input/cell-images-for-detecting-malaria/cell_images/cell_images/Uninfected/
      C203ThinF_IMG_20151029_102123_cell_51.png',
       '../input/cell-images-for-detecting-malaria/cell_images/cell_images/Uninfected/
      C238NThinF_IMG_20151207_114038_cell_84.png',
```

```
  '../input/cell-images-for-detecting-malaria/cell_images/cell_images/Uninfected/
C119P80ThinF_IMG_20151002_124304_cell_112.png',
  '../input/cell-images-for-detecting-malaria/cell_images/cell_images/Uninfected/
C39P4thinF_original_IMG_20150622_111723_cell_33.png',
  '../input/cell-images-for-detecting-malaria/cell_images/cell_images/Uninfected/
C101P62ThinF_IMG_20150918_151149_cell_63.png']
```
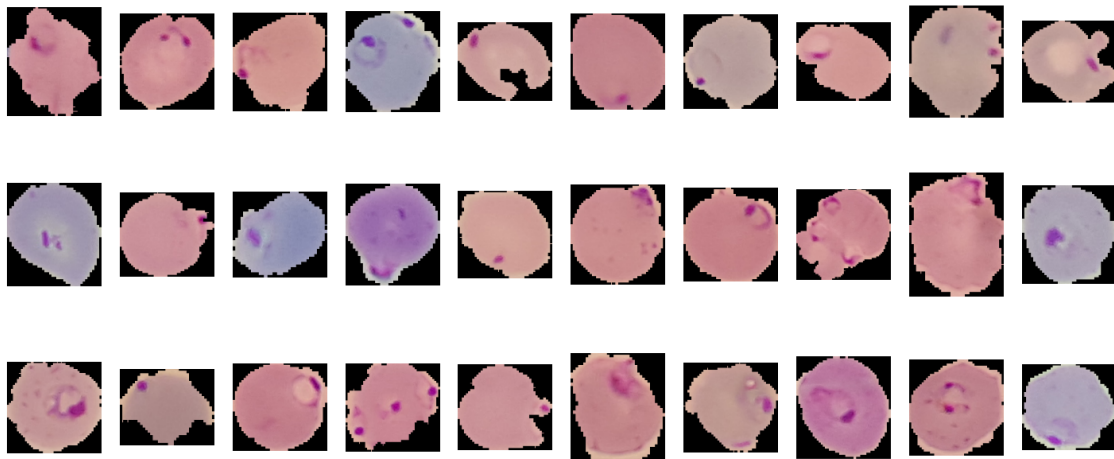
3 Images

```
[16]: fig, ax = plt.subplots(figsize=(18, 8))
      fig.suptitle('Parasitized cells', fontsize=24)

      for ind, img_src in enumerate(parasitized[:30]):
          plt.subplot(3, 10, ind+1)
          img = plt.imread(img_src)
          plt.axis('off')
          plt.imshow(img)
```

```
/tmp/ipykernel_35/3638847830.py:5: MatplotlibDeprecationWarning: Auto-removal of
overlapping axes is deprecated since 3.6 and will be removed two minor releases
later; explicitly call ax.remove() as needed.
  plt.subplot(3, 10, ind+1)
```


Parasitized cells

```
[17]: fig, ax = plt.subplots(figsize=(18, 8))
      fig.suptitle('Uninfected cells', fontsize=24)

      for ind, img_src in enumerate(uninfected[:30]):
          plt.subplot(3, 10, ind+1)
```
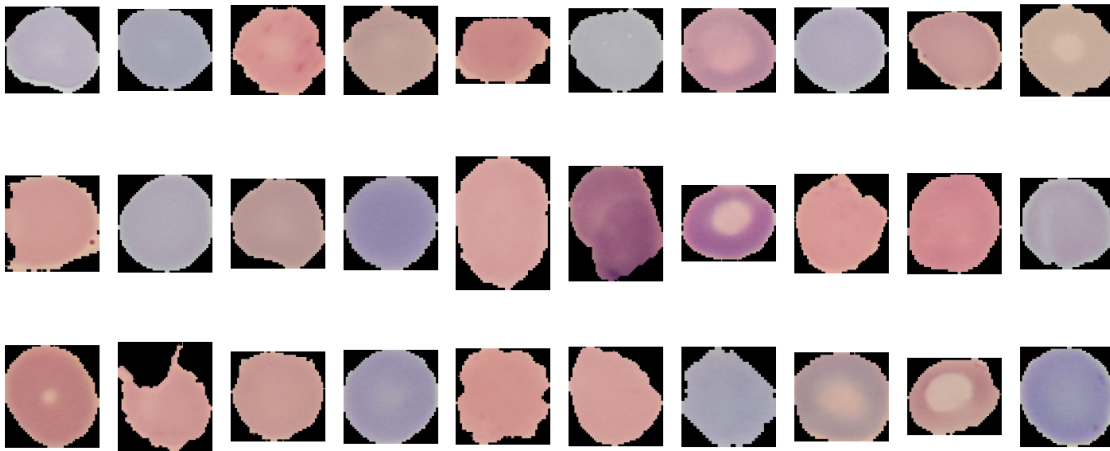
```
    img = plt.imread(img_src)
    plt.axis('off')
    plt.imshow(img)
```

/tmp/ipykernel_35/3420207871.py:5: MatplotlibDeprecationWarning: Auto-removal of
overlapping axes is deprecated since 3.6 and will be removed two minor releases
later; explicitly call ax.remove() as needed.
  plt.subplot(3, 10, ind+1)

Uninfected cells

4 Model

4.0.1 Model parameters

```
[18]: BATCH_SIZE = 100   # Number of training examples to process before updating our
      →models variables
      IMG_SHAPE  = 150   # Our training data consists of images with width of 150
      →pixels and height of 150 pixels
      TARGET_SIZE = 64
      EPOCHS = 10
```

4.0.2 Model initialization

```
[19]: model = Sequential()

      model.add(Conv2D(32, (3,3), activation='relu', input_shape=(IMG_SHAPE,
      →IMG_SHAPE, 3)))
      model.add(MaxPooling2D(2,2))
```

```python
model.add(Conv2D(64, (3,3), activation='relu'))
model.add(MaxPooling2D(2,2))

model.add(Conv2D(128, (3,3), activation='relu'))
model.add(MaxPooling2D(2,2))

model.add(Flatten())

model.add(Dropout(0.2))
model.add(Dense(128, activation='relu'))

model.add(Dropout(0.2))
model.add(Dense(1, activation='sigmoid'))

model.compile(optimizer='adam',
              loss='binary_crossentropy',
              metrics=['accuracy'])

model.summary()
```

/opt/conda/lib/python3.10/site-packages/keras/src/layers/convolutional/base_conv.py:107: UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When using Sequential models, prefer using an `Input(shape)` object as the first layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)

Model: "sequential_1"

Layer (type)	Output Shape	Param #
conv2d_3 (Conv2D)	(None, 148, 148, 32)	896
max_pooling2d_3 (MaxPooling2D)	(None, 74, 74, 32)	0
conv2d_4 (Conv2D)	(None, 72, 72, 64)	18,496
max_pooling2d_4 (MaxPooling2D)	(None, 36, 36, 64)	0
conv2d_5 (Conv2D)	(None, 34, 34, 128)	73,856
max_pooling2d_5 (MaxPooling2D)	(None, 17, 17, 128)	0
flatten_1 (Flatten)	(None, 36992)	0
dropout_2 (Dropout)	(None, 36992)	0

```
dense_2 (Dense)                  (None, 128)              4,735,104

dropout_3 (Dropout)              (None, 128)                      0

dense_3 (Dense)                  (None, 1)                      129
```
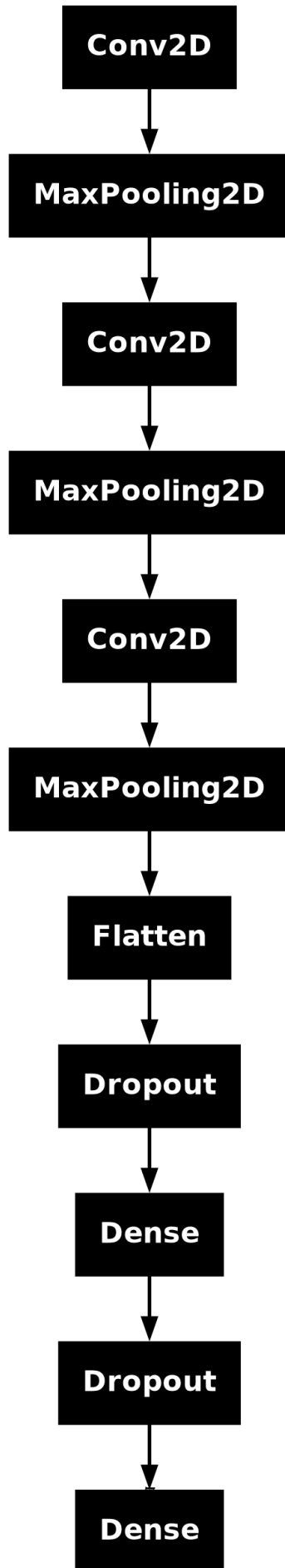
Total params: 4,828,481 (18.42 MB)

Trainable params: 4,828,481 (18.42 MB)

Non-trainable params: 0 (0.00 B)

```
[20]: plt.figure(figsize=(5, 10))
      plot_model(model, to_file="model.png")
```

[20]:

```
Conv2D
  |
  v
MaxPooling2D
  |
  v
Conv2D
  |
  v
MaxPooling2D
  |
  v
Conv2D
  |
  v
MaxPooling2D
  |
  v
Flatten
  |
  v
Dropout
  |
  v
Dense
  |
  v
Dropout
  |
  v
Dense
```

```
<Figure size 500x1000 with 0 Axes>
```

4.0.3 Data generator

```python
[21]: datagen = ImageDataGenerator(rescale=1./255,
                                   zoom_range=0.2,
                                   horizontal_flip=True,
                                   vertical_flip=True,
                                   width_shift_range=0.2,
                                   height_shift_range=0.2,
                                   validation_split=0.3)

      train_data = datagen.flow_from_directory('../input/
       ↪cell-images-for-detecting-malaria/cell_images/cell_images',
                                               target_size=(IMG_SHAPE,IMG_SHAPE),
                                               batch_size=BATCH_SIZE,
                                               shuffle=True,
                                               class_mode='binary',
                                               subset='training')

      validation_data = datagen.flow_from_directory('../input/
       ↪cell-images-for-detecting-malaria/cell_images/cell_images',
                                                    target_size=(IMG_SHAPE,IMG_SHAPE),
                                                    batch_size=BATCH_SIZE,
                                                    shuffle=True,
                                                    class_mode='binary',
                                                    subset='validation')
```

```
Found 19292 images belonging to 2 classes.
Found 8266 images belonging to 2 classes.
```

4.0.4 Callback functions

```python
[22]: # Instantiate an early stopping callback
      early_stopping = EarlyStopping(monitor='val_loss',
                                     min_delta = 0.01,
                                     patience=5)

      # Instantiate a model checkpoint callback
      model_save = ModelCheckpoint('best_model.keras',
                                   monitor='val_loss',
                                   mode='min',
                                   save_best_only=True)
```

4.0.5 Fit model

```
[23]: history = model.fit(train_data,
                          validation_data=validation_data,
                          epochs=EPOCHS,
                          verbose=1,
                          callbacks=[early_stopping, model_save])
```

Epoch 1/10

/opt/conda/lib/python3.10/site-
packages/keras/src/trainers/data_adapters/py_dataset_adapter.py:121:
UserWarning: Your `PyDataset` class should call `super().__init__(**kwargs)` in
its constructor. `**kwargs` can include `workers`, `use_multiprocessing`,
`max_queue_size`. Do not pass these arguments to `fit()`, as they will be
ignored.
  self._warn_if_super_not_called()
2024-06-18 10:27:29.405481: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 8:
6.5136, expected 5.75907
2024-06-18 10:27:29.405544: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 10:
5.95877, expected 5.20424
2024-06-18 10:27:29.405557: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 24:
6.38788, expected 5.63334
2024-06-18 10:27:29.405570: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 29:
4.83822, expected 4.08368
2024-06-18 10:27:29.405587: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 30:
6.36181, expected 5.60728
2024-06-18 10:27:29.405602: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 31:
6.32307, expected 5.56853
2024-06-18 10:27:29.405613: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 37:
6.15811, expected 5.40357
2024-06-18 10:27:29.405624: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 43:
6.51688, expected 5.76234
2024-06-18 10:27:29.405634: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 44:
6.30563, expected 5.5511
2024-06-18 10:27:29.405645: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 53:
5.30601, expected 4.55147
2024-06-18 10:27:29.436719: E

external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:705] Results mismatch between different convolution algorithms. This is likely a bug/unexpected loss of precision in cudnn.
(f32[100,32,148,148]{3,2,1,0}, u8[0]{0}) custom-call(f32[100,3,150,150]{3,2,1,0}, f32[32,3,3,3]{3,2,1,0}, f32[32]{0}), window={size=3x3}, dim_labels=bf01_oi01->bf01, custom_call_target="__cudnn$convBiasActivationForward", backend_config={"conv_result_scale":1,"activation_mode":"kRelu","side_input_scale":0,"leakyrelu_alpha":0} for eng20{k2=2,k4=1,k5=1,k6=0,k7=0} vs eng15{k5=1,k6=0,k7=1,k10=1}
2024-06-18 10:27:29.436779: E external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:270] Device: Tesla P100-PCIE-16GB
2024-06-18 10:27:29.436790: E external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:271] Platform: Compute Capability 6.0
2024-06-18 10:27:29.436802: E external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:272] Driver: 12020 (535.129.3)
2024-06-18 10:27:29.436816: E external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:273] Runtime: <undefined>
2024-06-18 10:27:29.436852: E external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:280] cudnn version: 8.9.0
2024-06-18 10:27:30.426964: E external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 8: 6.5136, expected 5.75907
2024-06-18 10:27:30.427024: E external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 10: 5.95877, expected 5.20424
2024-06-18 10:27:30.427037: E external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 24: 6.38788, expected 5.63334
2024-06-18 10:27:30.427050: E external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 29: 4.83822, expected 4.08368
2024-06-18 10:27:30.427067: E external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 30: 6.36181, expected 5.60728
2024-06-18 10:27:30.427078: E external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 31: 6.32307, expected 5.56853
2024-06-18 10:27:30.427088: E external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 37: 6.15811, expected 5.40357
2024-06-18 10:27:30.427099: E external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 43: 6.51688, expected 5.76234

2024-06-18 10:27:30.427109: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 44:
6.30563, expected 5.5511
2024-06-18 10:27:30.427120: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 53:
5.30601, expected 4.55147
2024-06-18 10:27:30.458001: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:705] Results
mismatch between different convolution algorithms. This is likely a
bug/unexpected loss of precision in cudnn.
(f32[100,32,148,148]{3,2,1,0}, u8[0]{0}) custom-
call(f32[100,3,150,150]{3,2,1,0}, f32[32,3,3,3]{3,2,1,0}, f32[32]{0}),
window={size=3x3}, dim_labels=bf01_oi01->bf01,
custom_call_target="__cudnn$convBiasActivationForward", backend_config={"conv_re
sult_scale":1,"activation_mode":"kRelu","side_input_scale":0,"leakyrelu_alpha":0
} for eng20{k2=2,k4=1,k5=1,k6=0,k7=0} vs eng15{k5=1,k6=0,k7=1,k10=1}
2024-06-18 10:27:30.458065: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:270] Device: Tesla
P100-PCIE-16GB
2024-06-18 10:27:30.458077: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:271] Platform:
Compute Capability 6.0
2024-06-18 10:27:30.458088: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:272] Driver: 12020
(535.129.3)
2024-06-18 10:27:30.458104: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:273] Runtime:
<undefined>
2024-06-18 10:27:30.458127: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:280] cudnn version:
8.9.0

  1/193   1:08:01 21s/step -
accuracy: 0.4800 - loss: 0.7004

WARNING: All log messages before absl::InitializeLog() is called are written to
STDERR
I0000 00:00:1718706457.770578   111 device_compiler.h:186] Compiled cluster
using XLA! This line is logged at most once for the lifetime of the process.

 14/193   2:40 896ms/step -
accuracy: 0.4997 - loss: 0.8354

2024-06-18 10:27:52.806735: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 0:
5.4949, expected 4.61249
2024-06-18 10:27:52.806787: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 1:
6.42841, expected 5.54601
2024-06-18 10:27:52.806802: E

external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 2:
6.27299, expected 5.39059
2024-06-18 10:27:52.806820: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 3:
6.7324, expected 5.85
2024-06-18 10:27:52.806847: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 4:
7.44829, expected 6.56589
2024-06-18 10:27:52.806858: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 6:
7.5376, expected 6.6552
2024-06-18 10:27:52.806868: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 7:
6.07025, expected 5.18785
2024-06-18 10:27:52.806878: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 8:
5.42679, expected 4.54439
2024-06-18 10:27:52.806888: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 9:
6.20133, expected 5.31892
2024-06-18 10:27:52.806900: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 10:
5.20637, expected 4.32397
2024-06-18 10:27:52.835506: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:705] Results
mismatch between different convolution algorithms. This is likely a
bug/unexpected loss of precision in cudnn.
(f32[92,32,148,148]{3,2,1,0}, u8[0]{0}) custom-call(f32[92,3,150,150]{3,2,1,0},
f32[32,3,3,3]{3,2,1,0}, f32[32]{0}), window={size=3x3},
dim_labels=bf01_oi01->bf01,
custom_call_target="__cudnn$convBiasActivationForward", backend_config={"conv_re
sult_scale":1,"activation_mode":"kRelu","side_input_scale":0,"leakyrelu_alpha":0
} for eng20{k2=2,k4=1,k5=1,k6=0,k7=0} vs eng15{k5=1,k6=0,k7=1,k10=1}
2024-06-18 10:27:52.835553: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:270] Device: Tesla
P100-PCIE-16GB
2024-06-18 10:27:52.835568: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:271] Platform:
Compute Capability 6.0
2024-06-18 10:27:52.835579: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:272] Driver: 12020
(535.129.3)
2024-06-18 10:27:52.835595: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:273] Runtime:
<undefined>
2024-06-18 10:27:52.835616: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:280] cudnn version:
8.9.0

2024-06-18 10:27:53.612760: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 0:
5.4949, expected 4.61249
2024-06-18 10:27:53.612820: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 1:
6.42841, expected 5.54601
2024-06-18 10:27:53.612841: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 2:
6.27299, expected 5.39059
2024-06-18 10:27:53.612850: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 3:
6.7324, expected 5.85
2024-06-18 10:27:53.612858: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 4:
7.44829, expected 6.56589
2024-06-18 10:27:53.612865: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 6:
7.5376, expected 6.6552
2024-06-18 10:27:53.612873: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 7:
6.07025, expected 5.18785
2024-06-18 10:27:53.612881: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 8:
5.42679, expected 4.54439
2024-06-18 10:27:53.612889: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 9:
6.20133, expected 5.31892
2024-06-18 10:27:53.612896: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at 10:
5.20637, expected 4.32397
2024-06-18 10:27:53.641293: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:705] Results
mismatch between different convolution algorithms. This is likely a
bug/unexpected loss of precision in cudnn.
(f32[92,32,148,148]{3,2,1,0}, u8[0]{0}) custom-call(f32[92,3,150,150]{3,2,1,0},
f32[32,3,3,3]{3,2,1,0}, f32[32]{0}), window={size=3x3},
dim_labels=bf01_oi01->bf01,
custom_call_target="__cudnn$convBiasActivationForward", backend_config={"conv_re
sult_scale":1,"activation_mode":"kRelu","side_input_scale":0,"leakyrelu_alpha":0
} for eng20{k2=2,k4=1,k5=1,k6=0,k7=0} vs eng15{k5=1,k6=0,k7=1,k10=1}
2024-06-18 10:27:53.641357: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:270] Device: Tesla
P100-PCIE-16GB
2024-06-18 10:27:53.641365: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:271] Platform:
Compute Capability 6.0
2024-06-18 10:27:53.641372: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:272] Driver: 12020

(535.129.3)
2024-06-18 10:27:53.641379: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:273] Runtime:
<undefined>
2024-06-18 10:27:53.641396: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:280] cudnn version:
8.9.0

192/193   **1s** 1s/step -
accuracy: 0.5240 - loss: 0.7139

2024-06-18 10:32:24.694481: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at
49459: 3.51436, expected 3.05234
2024-06-18 10:32:24.694601: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at
59394: 3.51436, expected 3.05234
2024-06-18 10:32:24.694752: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at
87616: 6.40813, expected 5.58182
2024-06-18 10:32:24.694768: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at
87617: 6.90036, expected 6.07405
2024-06-18 10:32:24.694787: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at
87618: 6.47252, expected 5.64622
2024-06-18 10:32:24.694800: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at
87619: 6.81659, expected 5.99029
2024-06-18 10:32:24.694811: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at
87620: 6.9288, expected 6.10249
2024-06-18 10:32:24.694821: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at
87624: 6.65719, expected 5.83088
2024-06-18 10:32:24.694833: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at
87625: 7.09887, expected 6.27257
2024-06-18 10:32:24.694857: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at
87626: 6.04046, expected 5.21415
2024-06-18 10:32:24.716646: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:705] Results
mismatch between different convolution algorithms. This is likely a
bug/unexpected loss of precision in cudnn.
(f32[66,32,148,148]{3,2,1,0}, u8[0]{0}) custom-call(f32[66,3,150,150]{3,2,1,0},
f32[32,3,3,3]{3,2,1,0}, f32[32]{0}), window={size=3x3},
dim_labels=bf01_oi01->bf01,
custom_call_target="__cudnn$convBiasActivationForward", backend_config={"conv_re

sult_scale":1,"activation_mode":"kRelu","side_input_scale":0,"leakyrelu_alpha":0
} for eng20{k2=2,k4=1,k5=1,k6=0,k7=0} vs eng15{k5=1,k6=0,k7=1,k10=1}
2024-06-18 10:32:24.716705: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:270] Device: Tesla
P100-PCIE-16GB
2024-06-18 10:32:24.716715: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:271] Platform:
Compute Capability 6.0
2024-06-18 10:32:24.716722: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:272] Driver: 12020
(535.129.3)
2024-06-18 10:32:24.716730: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:273] Runtime:
<undefined>
2024-06-18 10:32:24.716747: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:280] cudnn version:
8.9.0
2024-06-18 10:32:25.247994: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at
49459: 3.51436, expected 3.05234
2024-06-18 10:32:25.248096: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at
59394: 3.51436, expected 3.05234
2024-06-18 10:32:25.248233: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at
87616: 6.40813, expected 5.58182
2024-06-18 10:32:25.248252: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at
87617: 6.90036, expected 6.07405
2024-06-18 10:32:25.248267: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at
87618: 6.47252, expected 5.64622
2024-06-18 10:32:25.248278: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at
87619: 6.81659, expected 5.99029
2024-06-18 10:32:25.248288: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at
87620: 6.9288, expected 6.10249
2024-06-18 10:32:25.248299: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at
87624: 6.65719, expected 5.83088
2024-06-18 10:32:25.248309: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at
87625: 7.09887, expected 6.27257
2024-06-18 10:32:25.248326: E
external/local_xla/xla/service/gpu/buffer_comparator.cc:1137] Difference at
87626: 6.04046, expected 5.21415
2024-06-18 10:32:25.269320: E

external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:705] Results
mismatch between different convolution algorithms. This is likely a
bug/unexpected loss of precision in cudnn.
(f32[66,32,148,148]{3,2,1,0}, u8[0]{0}) custom-call(f32[66,3,150,150]{3,2,1,0},
f32[32,3,3,3]{3,2,1,0}, f32[32]{0}), window={size=3x3},
dim_labels=bf01_oi01->bf01,
custom_call_target="__cudnn$convBiasActivationForward", backend_config={"conv_re
sult_scale":1,"activation_mode":"kRelu","side_input_scale":0,"leakyrelu_alpha":0
} for eng20{k2=2,k4=1,k5=1,k6=0,k7=0} vs eng15{k5=1,k6=0,k7=1,k10=1}
2024-06-18 10:32:25.269377: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:270] Device: Tesla
P100-PCIE-16GB
2024-06-18 10:32:25.269386: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:271] Platform:
Compute Capability 6.0
2024-06-18 10:32:25.269393: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:272] Driver: 12020
(535.129.3)
2024-06-18 10:32:25.269400: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:273] Runtime:
<undefined>
2024-06-18 10:32:25.269417: E
external/local_xla/xla/service/gpu/conv_algorithm_picker.cc:280] cudnn version:
8.9.0

193/193   310s 2s/step -
accuracy: 0.5246 - loss: 0.7135 - val_accuracy: 0.7539 - val_loss: 1.4480
Epoch 2/10
193/193   184s 931ms/step -
accuracy: 0.8342 - loss: 0.4318 - val_accuracy: 0.9013 - val_loss: 0.7421
Epoch 3/10
193/193   180s 911ms/step -
accuracy: 0.9042 - loss: 0.2908 - val_accuracy: 0.9233 - val_loss: 0.4158
Epoch 4/10
193/193   179s 904ms/step -
accuracy: 0.9174 - loss: 0.2533 - val_accuracy: 0.9215 - val_loss: 0.3732
Epoch 5/10
193/193   179s 904ms/step -
accuracy: 0.9281 - loss: 0.2272 - val_accuracy: 0.9054 - val_loss: 0.6779
Epoch 6/10
193/193   180s 911ms/step -
accuracy: 0.9236 - loss: 0.2230 - val_accuracy: 0.9234 - val_loss: 1.0902
Epoch 7/10
193/193   179s 906ms/step -
accuracy: 0.9259 - loss: 0.2283 - val_accuracy: 0.9261 - val_loss: 1.0754
Epoch 8/10
193/193   179s 904ms/step -
accuracy: 0.9283 - loss: 0.2214 - val_accuracy: 0.9306 - val_loss: 1.6048

```
Epoch 9/10
193/193                  183s 926ms/step -
accuracy: 0.9335 - loss: 0.2036 - val_accuracy: 0.9205 - val_loss: 1.6935
```

4.0.6 Plot metrics

```python
[24]: plt.figure(figsize=(14, 5))

      plt.subplot(1, 2, 1)
      plt.plot(history.history['accuracy'], label='Training Accuracy')
      plt.plot(history.history['val_accuracy'], label='Validation Accuracy')
      plt.legend(loc='lower right')
      plt.title('Training and Validation Accuracy')

      plt.subplot(1, 2, 2)
      plt.plot(history.history['loss'], label='Training Loss')
      plt.plot(history.history['val_loss'], label='Validation Loss')
      plt.legend(loc='upper right')
      plt.title('Training and Validation Loss')

      plt.show()
```