Lampiran 1

Source Code

```
"cells": [
  "cell_type": "code",
  "execution_count": 1,
  "metadata": {},
  "outputs": [
  {
    "name": "stderr",
    "output_type": "stream",
    "text": "Using TensorFlow backend.\n"
  }
  ],
  "source": "from keras.models import Sequential\nfrom keras.
     layers import Conv2D, MaxPooling2D, Flatten, Dense, Dropout,
      Activation, BatchNormalization"
 },
 {
  "cell_type": "code",
  "execution_count": 2,
  "metadata": {},
  "outputs": [],
  "source": "import numpy as np\nimport pandas as pd\nimport
     matplotlib.pyplot as plt\nimport tensorflow\n\matplotlib
     inline"
 },
  "cell_type": "code",
  "execution_count": 3,
  "metadata": {},
  "outputs": [
    "name": "stdout",
    "output_type": "stream",
```

```
"text": "Model: \"sequential 1\"\
                                      Output Shape
      \nLayer (type)
      Param # \n
      nconv2d 1 (Conv2D)
                                     (None, 198, 198, 32)
      896
      \nconv2d 2 (Conv2D)
                                      (None, 196, 196, 16)
      4624
      \nmax_pooling2d_1 (MaxPooling2 (None, 65, 65, 16)
                                      (None, 67600)
      \nflatten 1 (Flatten)
               \
      \ndense_1 (Dense)
                                      (None, 12)
      811212
                                      (None, 12)
      \ndropout_1 (Dropout)
               /
                                      (None, 24)
      \ndense 2 (Dense)
      312
                \n
      nTotal params: 817,044\nTrainable params: 817,044\nNon-
      trainable params: 0\
      n''
 }
 ],
 "source": "nb classes = 24\nmodel = Sequential()\nmodel.add(
    Conv2D(32, kernel size = (3,3), activation = 'relu', input shape
    =(200,200,3)) \setminus nmodel.add(Conv2D(16, kernel size = (3,3),
    activation = 'relu', input_shape = (200,200,3)) \nmodel.add(
    MaxPooling2D(pool size = (3, 3)))\nmodel.add(Flatten())\nmodel
    .add(Dense(12, activation= 'relu'))\nmodel.add(Dropout(0.2))
    \nmodel.add(Dense(nb_classes, activation = 'softmax'))\n#
    model.layers[0].trainable = True\nmodel.summary()"
},
{
 "cell type": "code",
 "execution count": 8,
 "metadata": {},
```

```
"outputs": [],
 "source": "from keras.optimizers import RMSprop\noptimizers =
    RMSprop(lr = 0.00001) \setminus nmodel.compile( \setminus n
                      loss = 'categorical crossentropy',\n
    optimizers,\n
    metrics = ['accuracy']\n)"
},
 "cell type": "code",
 "execution count": 9,
 "metadata": {},
 "outputs": [
   "name": "stdout",
   "output type": "stream",
   "text": "Found 1712 images belonging to 24 classes.\nFound
      318 images belonging to 24 classes.\n"
  }
 ],
 "source": "from keras.preprocessing.image import
    ImageDataGenerator\nfrom keras.models import Sequential\
    nfrom keras.layers import Dropout, Flatten, Dense\nfrom
    keras import applications\ntrain_datagen =
    ImageDataGenerator (rescale = 1./255,\n
                                         horizontal flip=False)\
    ntest datagen = ImageDataGenerator(rescale = 1./255)\nclasses
    = ['A','A#','A#m','Am','B','Bm',\n
                                                    'C', 'C#', 'C#m
    ','Cm','D','D#',\n
                                   \"D#m\", 'Dm', 'E', 'Em', 'F', 'F
    #',\n
                     'F#m', 'Fm', 'G', 'G#', 'G#m', 'Gm']\
    ntrain generator = train datagen.flow from directory(\n
            'train/data train', \n
                                           target size = (200, 200)
    ,\n
                shuffle = True,\n
                                           batch_size=64,
             class_mode='categorical')\ntest_generator =
    test datagen.flow from directory(\n
                                                 'train/data test
    ',\n
                 target size = (200, 200),\n
                                                    shuffle = True
                batch size = 247,\n
                                          class mode='categorical
    , \ n
    ')"
},
 "cell type": "code",
 "execution count": 10,
 "metadata": {},
 "outputs": [],
 "source": "from keras.callbacks import ModelCheckpoint\
    nfilepath = \"weight.checkpoint5.cont.h5\"\ncheckpoint = [
    ModelCheckpoint(filepath , monitor='val_loss', verbose=1,
```

```
save best only=True, mode='min'),\n
    ModelCheckpoint(filepath , monitor='loss', verbose=1,
    save best only=True, mode='min')]"
},
{
"cell type": "code",
 "execution_count": 11,
 "metadata": {
 "scrolled": true
 },
 "outputs": [
  "name": "stdout",
  "output type": "stream",
  "text": "Epoch 1/1000 \ n26/26 [====
      - 60s 2s/step - loss: 3.0627 - accuracy: 0.1438 -
     val loss: 3.1323 - \text{val} \ \text{accuracy}: 0.0409 \ \text{n} \ \text{Depoch} \ 00001:
     val loss improved from inf to 3.13228, saving model to
     weight.checkpoint5.cont.h5\n\nEpoch 00001: loss improved
     from inf to 3.06291, saving model to weight.checkpoint5.
     cont.h5\nEpoch 2/1000\n26/26
     [=====] - 57s 2s/step - loss:
     3.0615 - accuracy: 0.1456 - val loss: 3.2120 -
     val accuracy: 0.0409\n\nEpoch 00002: val loss did not
     improve from 3.13228\n\nEpoch 00002: loss improved from
     3.06291 to 3.06162, saving model to weight.checkpoint5.
     cont.h5\nEpoch 3/1000\n26/26
     [======] - 56s 2s/step - loss:
     3.0638 - accuracy: 0.1468 - val loss: 3.2062 -
     val accuracy: 0.0409\n\nEpoch 00003: val loss did not
     improve from 3.13228\n\nEpoch 00003: loss did not improve
     from 3.06162\nEpoch 4/1000\n26/26
          3.0567 - accuracy: 0.1462 - val loss: 3.1539 -
     val accuracy: 0.0409\n\nEpoch 00004: val loss did not
     improve from 3.13228\n\nEpoch 00004: loss improved from
     3.06162 to 3.05760, saving model to weight.checkpoint5.
     cont.h5\nEpoch 5/1000\n26/26
     3.0700 - accuracy: 0.1390 - val loss: 3.1996 -
     val accuracy: 0.0409\n\nEpoch 00005: val loss did not
     improve from 3.13228\n\nEpoch 00005: loss did not improve
     from 3.05760\nEpoch 6/1000\n26/26
              3.0613 - accuracy: 0.1448 - val_loss: 3.1987 -
```

```
val accuracy: 0.0409\n\nEpoch 00006: val loss did not
improve from 3.13228\n\nEpoch 00006: loss did not improve
from 3.05760 \times poch 7/1000 \times n26/26
[======] - 76s 3s/step - loss:
3.0699 - accuracy: 0.1379 - val loss: 3.1736 -
val accuracy: 0.0409\n\nEpoch 00007: val loss did not
improve from 3.13228\n\nEpoch 00007: loss did not improve
from 3.05760 \nEpoch 8/1000 \n26/26
[=====] - 62s 2s/step - loss:
3.0619 - accuracy: 0.1490 - val loss: 3.1940 -
val accuracy: 0.0409\n\nEpoch 00008: val loss did not
improve from 3.13228\n\nEpoch 00008: loss did not improve
from 3.05760\nEpoch 9/1000\n26/26
[=====] - 56s 2s/step - loss:
3.0475 - accuracy: 0.1566 - val loss: 3.1843 -
val_accuracy: 0.0409\n\nEpoch 00009: val loss did not
improve from 3.13228\n\nEpoch 00009: loss improved from
3.05760 to 3.04834, saving model to weight.checkpoint5.
cont.h5\nEpoch 10/1000\n26/26
[=====] - 56s 2s/step - loss:
3.0682 - accuracy: 0.1403 - val_loss: 3.2236 -
val_accuracy: 0.0409\n\nEpoch 00010: val_loss did not
improve from 3.13228\n\nEpoch 00010: loss did not improve
from 3.04834\nEpoch 11/1000\n26/26
[======] - 56s 2s/step - loss:
3.0626 - accuracy: 0.1426 - val loss: 3.1664 -
val accuracy: 0.0409\n\nEpoch 00011: val loss did not
improve from 3.13228\n\nEpoch 00011: loss did not improve
from 3.04834\nEpoch 12/1000\n26/26
[=====] - 55s 2s/step - loss:
3.0682 - accuracy: 0.1424 - val loss: 3.1899 -
val accuracy: 0.0409\n\nEpoch 00012: val loss did not
improve from 3.13228\n\nEpoch 00012: loss did not improve
from 3.04834\nEpoch 13/1000\n26/26
[=====] - 61s 2s/step - loss:
3.0583 - accuracy: 0.1499 - val_loss: 3.2304 -
val accuracy: 0.0409\n\nEpoch 00013: val loss did not
improve from 3.13228\n\nEpoch 00013: loss did not improve
from 3.04834\nEpoch 14/1000\n26/26
[======] - 66s 3s/step - loss:
3.0549 - accuracy: 0.1477 - val loss: 3.2015 -
val accuracy: 0.0409\n\nEpoch 00014: val loss did not
improve from 3.13228\n\nEpoch 00014: loss did not improve
from 3.04834\nEpoch 15/1000\n26/26
[======] - 67s 3s/step - loss:
```

```
3.0755 - accuracy: 0.1370 - val loss: 3.1619 -
val_accuracy: 0.0409\n\nEpoch 00015: val_loss did not
improve from 3.13228\n\nEpoch 00015: loss did not improve
from 3.04834\nEpoch 16/1000\n26/26
[=====] - 66s 3s/step - loss:
3.0631 - accuracy: 0.1446 - val loss: 3.1893 -
val accuracy: 0.0409\n\nEpoch 00016: val loss did not
improve from 3.13228\n\nEpoch 00016: loss did not improve
from 3.04834\nEpoch 17/1000\n26/26
[======] - 67s 3s/step - loss:
3.0558 - accuracy: 0.1496 - val loss: 3.2430 -
val accuracy: 0.0409\n\nEpoch 00017: val loss did not
improve from 3.13228\n\nEpoch 00017: loss did not improve
from 3.04834\nEpoch 18/1000\n26/26
       3.0616 - accuracy: 0.1471 - val loss: 3.1461 -
val_accuracy: 0.0409\n\nEpoch 00018: val_loss did not
improve from 3.13228\n\nEpoch 00018: loss did not improve
from 3.04834\nEpoch 19/1000\n26/26
[=====] - 659s 25s/step - loss:
3.0645 - accuracy: 0.1420 - val_loss: 3.2351 -
val_accuracy: 0.0409\n\nEpoch 00019: val_loss did not
improve from 3.13228\n\nEpoch 00019: loss did not improve
from 3.04834\nEpoch 20/1000\n26/26
[======] - 54s 2s/step - loss:
3.0592 - accuracy: 0.1444 - val loss: 3.1370 -
val accuracy: 0.0409\n\nEpoch 00020: val loss did not
improve from 3.13228\n\nEpoch 00020: loss did not improve
from 3.04834\nEpoch 21/1000\n26/26
[=====] - 80s 3s/step - loss:
3.0646 - accuracy: 0.1472 - val loss: 3.1770 -
val accuracy: 0.0409\n\nEpoch 00021: val loss did not
improve from 3.13228\n\nEpoch 00021: loss did not improve
from 3.04834\nEpoch 22/1000\n26/26
[=====] - 64s 2s/step - loss:
3.0630 - accuracy: 0.1475 - val_loss: 3.2277 -
val accuracy: 0.0409\n\nEpoch 00022: val loss did not
improve from 3.13228\n\nEpoch 00022: loss did not improve
from 3.04834\nEpoch 23/1000\n26/26
[======] - 55s 2s/step - loss:
3.0635 - accuracy: 0.1408 - val loss: 3.1488 -
val accuracy: 0.0409\n\nEpoch 00023: val loss did not
improve from 3.13228\n\nEpoch 00023: loss did not improve
from 3.04834\nEpoch 24/1000\n26/26
```

```
3.0685 - accuracy: 0.1409 - val loss: 3.1949 -
val_accuracy: 0.0409\n\nEpoch 00024: val_loss did not
improve from 3.13228\n\nEpoch 00024: loss did not improve
from 3.04834\nEpoch 25/1000\n26/26
  ======] - 65s 3s/step - loss:
3.0628 - accuracy: 0.1472 - val loss: 3.1538 -
val accuracy: 0.0409\n\nEpoch 00025: val loss did not
improve from 3.13228\n\nEpoch 00025: loss did not improve
from 3.04834\nEpoch 26/1000\n26/26
[=====] - 66s 3s/step - loss:
3.0594 - accuracy: 0.1456 - val loss: 3.2175 -
val accuracy: 0.0409\n\nEpoch 00026: val loss did not
improve from 3.13228\n\nEpoch 00026: loss did not improve
from 3.04834\nEpoch 27/1000\n26/26
        3.0626 - accuracy: 0.1444 - val loss: 3.2237 -
val_accuracy: 0.0409\n\nEpoch 00027: val_loss did not
improve from 3.13228\n\nEpoch 00027: loss did not improve
from 3.04834\nEpoch 28/1000\n26/26
[=====] - 63s 2s/step - loss:
3.0645 - accuracy: 0.1432 - val_loss: 3.2014 -
val_accuracy: 0.0409\n\nEpoch 00028: val_loss did not
improve from 3.13228\n\nEpoch 00028: loss did not improve
from 3.04834\nEpoch 29/1000\n26/26
[======] - 59s 2s/step - loss:
3.0645 - accuracy: 0.1438 - val loss: 3.2313 -
val accuracy: 0.0409\n\nEpoch 00029: val loss did not
improve from 3.13228\n\nEpoch 00029: loss did not improve
from 3.04834\nEpoch 30/1000\n26/26
[=====] - 54s 2s/step - loss:
3.0619 - accuracy: 0.1468 - val_loss: 3.2027 -
val accuracy: 0.0409\n\nEpoch 00030: val loss did not
improve from 3.13228\n\nEpoch 00030: loss did not improve
from 3.04834\nEpoch 31/1000\n26/26
[=====] - 50s 2s/step - loss:
3.0614 - accuracy: 0.1438 - val_loss: 3.1087 -
val accuracy: 0.0409\n\nEpoch 00031: val loss improved
from 3.13228 to 3.10871, saving model to weight.
checkpoint5.cont.h5\n\nEpoch 00031: loss did not improve
from 3.04834\nEpoch 32/1000\n26/26
[======] - 51s 2s/step - loss:
3.0640 - accuracy: 0.1450 - val_loss: 3.1754 -
val accuracy: 0.0409\n\nEpoch 00032: val loss did not
improve from 3.10871\n\nEpoch 00032: loss did not improve
from 3.04834\nEpoch 33/1000\n26/26
```

```
[======] - 51s 2s/step - loss:
    3.0563 - accuracy: 0.1475 - val_loss: 3.2294 -
    val accuracy: 0.0409\n\nEpoch 00033: val loss did not
    improve from 3.10871\n\nEpoch 00033: loss did not improve
   from 3.04834\nEpoch 34/1000\n"
},
 "name": "stdout",
 "output type": "stream",
 "text": "26/26 [======] - 59s 2s/step
    - loss: 3.0678 - accuracy: 0.1408 - val loss: 3.1755 -
    val accuracy: 0.0409\n\nEpoch 00034: val loss did not
    improve from 3.10871\n\nEpoch 00034: loss did not improve
    from 3.04834\nEpoch 35/1000\n26/26
           3.0665 - accuracy: 0.1481 - val loss: 3.1536 -
    val_accuracy: 0.0409\n\nEpoch 00035: val_loss did not
   improve from 3.10871\n\nEpoch 00035: loss did not improve
   from 3.04834\nEpoch 36/1000\n26/26
   [=====] - 57s 2s/step - loss:
    3.0547 - accuracy: 0.1514 - val_loss: 3.1829 -
    val_accuracy: 0.0409\n\nEpoch 00036: val_loss did not
   improve from 3.10871\n\nEpoch 00036: loss did not improve
    from 3.04834\nEpoch 37/1000\n26/26
   [======] - 65s 3s/step - loss:
   3.0693 - accuracy: 0.1347 - val loss: 3.2082 -
    val accuracy: 0.0409\n\nEpoch 00037: val loss did not
   improve from 3.10871\n\nEpoch 00037: loss did not improve
    from 3.04834\nEpoch 38/1000\n26/26
   [======] - 65s 2s/step - loss:
    3.0557 - accuracy: 0.1477 - val_loss: 3.1935 -
    val accuracy: 0.0409\n\nEpoch 00038: val loss did not
   improve from 3.10871\n\nEpoch 00038: loss did not improve
   from 3.04834\nEpoch 39/1000\n26/26
   [=====] - 58s 2s/step - loss:
    3.0518 - accuracy: 0.1562 - val_loss: 3.1678 -
    val accuracy: 0.0409\n\nEpoch 00039: val loss did not
    improve from 3.10871\n\nEpoch 00039: loss did not improve
   from 3.04834\nEpoch 40/1000\n26/26
   [======] - 56s 2s/step - loss:
    3.0759 - accuracy: 0.1329 - val loss: 3.1427 -
    val accuracy: 0.0409\n\nEpoch 00040: val loss did not
   improve from 3.10871\n\nEpoch 00040: loss did not improve
   from 3.04834\nEpoch 41/1000\n26/26
   [======] - 59s 2s/step - loss:
```

```
3.0762 - accuracy: 0.1348 - val loss: 3.2024 -
    val_accuracy: 0.0409\n\nEpoch 00041: val_loss did not
    improve from 3.10871\n\nEpoch 00041: loss did not improve
    from 3.04834\nEpoch 42/1000\n26/26
            ======] - 54s 2s/step - loss:
    3.0533 - accuracy: 0.1541 - val loss: 3.2151 -
    val_accuracy: 0.0409\n\nEpoch 00042: val_loss did not
    improve from 3.10871\n\nEpoch 00042: loss did not improve
    from 3.04834\nEpoch 43/1000\n26/26
    [=====] - 53s 2s/step - loss:
    3.0579 - accuracy: 0.1460 - val loss: 3.1284 -
    val accuracy: 0.0409\n\nEpoch 00043: val loss did not
    improve from 3.10871\n\nEpoch 00043: loss did not improve
    from 3.04834\nEpoch 44/1000\n10/26
    - accuracy: 0.1458"
},
 "ename": "KeyboardInterrupt",
"evalue": "",
 "output_type": "error",
 "traceback": [
  "\u001b[0;31m]
     u001b[0m",
  "\u001b[0;31 mKeyboardInterrupt\u001b[0m
                             Traceback (most recent call last)
  "\u001b[0;32m<ipython-input-11-a42e6152a514>\u001b[0m in \]
     u001b \\ [0;36m < module > \\ \\ u001b \\ [0;34m() \\ \\ u001b \\ [0m\\ \\ n\\ \\ u001b \\ [1;32m]
          6\u001b[0m
                            \u001b[0mvalidation data\u001b[0m
     \u001b[0;34m=\u001b[0m\u001b[0mtest generator\u001b[0m\u001b]]]
     u001b[0;34m, u001b[0mu001b[0;34mu001b[0mu001b[0m]n]]]
     u001b[1;32m
                      7\u001b[0m
                                         \u001b[0
     mvalidation steps \u001b[0m\u001b[0;34m=\u001b[0m \u001b]])
     [0;36m2\u001b[0m\u001b[0;34m,\u001b[0m\u001b[0;34m\u001b[0]]]]]
     [0m\u001b[0m\n\u001b[0;32m]> 8\u001b[0;31m]
     callbacks=checkpoint)\n\u001b[0m",
  "\u001b[0;32m/Library/Frameworks/Python.framework/Versions"
     /2.7/lib/python2.7/site-packages/keras/legacy/interfaces.
     pyc\u001b[0m in \u001b[0;36mwrapper\u001b[0;34m(*args, **
     kwargs) \u001b[0m\n\u001b[1;32m]
                                       89\u001b[0m
                     warnings.warn('Update your '' +
     object name + '' call to the '+\n\001b[1;32m]
     u001b[0m
                                            'Keras 2 API: '+
```

```
signature, stacklevel=2)\n\sqrt{u001b}[0;32m\longrightarrow 91\u001b[0;31m]
                 \u001b[0;32mreturn\u001b[0m\u001b[0mfunc\
   u001b[0m\u001b[0;34m(\u001b[0m\u001b[0;34m*\u001b[0m\u001b[0;34m]
   u001b[0margs\u001b[0m\u001b[0;34m,\u001b[0m\u001b[0;34m]]]]
   **\u001b[0m\u001b[0mkwargs\u001b[0m\u001b[0;34m)\u001b[0mkwargs\u001b[0mkwargs]]]
   \u001b[0;34m\u001b[0m\u001b[0m\n\u001b[0m\u001b[1;32m]]]]
       92\u001b[0m
                             \u001b[0mwrapper\u001b[0m\u001b]
   [0;34m.\u001b[0m\u001b[0m\_original\_function\u001b[0m\u001b]]
   u001b[0;34m=\u001b[0m \u001b[0mfunc\u001b[0m\u001b[0;34m\
   u001b[0m\u001b[0m\n\u001b[1;32m]
                                           93\u001b[0m
   u001b[0;32 mreturn \u001b[0m \u001b[0mwrapper \u001b[0m \u001b]]]
   u001b[0;34m\u001b[0m\u001b[0m\n"]]
"\u001b[0;32m/Library/Frameworks/Python.framework/Versions"
   /2.7/lib/python2.7/site-packages/keras/engine/training.
   pyc\u001b[0m\ in\ \u001b[0;36mfit\_generator\u001b[0;34m(
   self, generator, steps per epoch, epochs, verbose,
   callbacks, validation data, validation steps,
   validation_freq, class_weight, max_queue_size, workers,
   use multiprocessing, shuffle, initial epoch)\u001b[0m\n\
   u001b[1;32m
                  1730\u001b[0m
                                               \u001b[0
   muse multiprocessing \u001b[0m\u001b[0;34m=\u001b[0m\u001b]]]
   u001b[0;34m\u001b[0m\u001b[0m\n\u001b[1;32m]
                                                      1731\u001b
   ΓOm
                     \u001b[0 \, mshuffle \u001b[0m\u001b[0;34m=\
   u001b[0m\u001b[0mshuffle\u001b[0m\u001b[0;34m,\u001b[0m\u001b]]]]
   u001b[0;34m\u001b[0m\u001b[0m\n\u001b[0;32m]> 1732\u001b]
                        initial epoch=initial epoch)\n\u001b[0
   [0;31m]
                    1733 \\ \verb|u001b| \\ 0m \\ \verb|u001b| \\ 0; \\ 34m \\ \verb|u001b| \\ 0m \\ \verb|u001b| \\
   m \ 001b[1;32m]
                         1734\u001b[0m
   [0m\n\u001b[1;32m]
                                             \u001b[0;34m@\u001b]
   [0m\u001b[0minterfaces\u001b[0m\u001b[0;34m.\u001b[0m\u001b]]]]
   u001b[0mlegacy_generator_methods_support\u001b[0m\u001b
   [0;34m\u001b[0m\u001b[0m\n"]]
"\u001b[0;32m/Library/Frameworks/Python.framework/Versions"
   /2.7/lib/python2.7/site-packages/keras/engine/
   training generator.pyc\u001b[0m in \u001b[0;36
   mfit_generator\u001b[0;34m(model, generator,
   steps per epoch, epochs, verbose, callbacks,
   validation data, validation steps, validation freq,
   class_weight, max_queue_size, workers,
   use multiprocessing, shuffle, initial epoch)\u001b[0m\n\
   u001b[1;32m
                   218\u001b[0m
                                                   \u001b[0
   msample \ weight \verb|\u001b| 0m\u001b| 0;34m = \verb|\u001b| 0m\u001b| 0
   msample\ weight \verb|\u001b|| 0m\verb|\u001b|| 0; 34m, \verb|\u001b|| 0m\verb|\u001b|| 0; 34m
   \u001b[0m\u001b[0m\n\u001b[1;32m]
                                          219\u001b[0m
```

```
\u001b[0
           mclass_weight\u001b[0m\u001b[0;34m=\u001b[0m\u001b[0]]]
           mclass\_weight \u001b[0m\u001b[0;34m,\u001b[0m\u001b[0;34m\wd0]]]
           u001b[0m\u001b[0m\n\u001b[0;32m]> 220\u001b[0;31m]
                                                                                                                                                                           reset metrics
           =False)\n\u001b[0m\u001b[1;32m
                                                                                                                                      221\u001b[0m\u001b
            [0;34m\u001b[0m\u001b[0m\n\u001b[1;32m]
                                                                                                                                                                    222\u001b[0m
                                                                     \u001b[0mouts\u001b[0m \u001b[0;34m=\
           u001b[0mouts\u001b[0m\u001b[0;34m)\u001b[0m\u001b[0;34m]
           u001b[0m\u001b[0m\n",
"\u001b[0;32m/Library/Frameworks/Python.framework/Versions"
           /2.7/lib/python2.7/site-packages/keras/engine/training.
           pyc \u001b[0m in \u001b[0;36mtrain on batch \u001b[0;34m(
           self, x, y, sample weight, class weight, reset metrics)\
           u001b[0m\n\u001b[1;32m]
                                                                                                     1512\u001b[0m
                                                                                                                                                                                                    \u001b
           m \u001b[0;34m+\u001b[0m \u001b[0m]\u001b[0m \u001b[0;34m]
           +\u001b[0m \u001b[0msample weights\u001b[0m\u001b[0;34m\u001b]]]
           u001b[0m\u001b[0m\n\u001b[1;32m]
                                                                                                                                   1513\u001b[0m
           u001b[0 \, mself \ u001b[0m \ u001b[0;34m. \ u001b[0m \ u001b[0]]]]
           m_make_train_function \u001b[0m\u001b[0;34m(\u001b[0m\u001b[0m\u001b[0]])])
           u001b[0;34m] u001b[0mu001b[0;34mu001b[0mu001b[0mn]]]
           u001b[0;32m \rightarrow 1514 u001b[0;31m]
                                                                                                                                                        \u001b[0moutputs\
           u001b[0m \u001b[0;34m=\u001b[0m \u001b[0mself\u001b[0m\w]]]
           u001b[0;34m.\u001b[0m\u001b[0mtrain function\u001b[0m\u001b]]]
           u001b[0;34m(\u001b[0m\u001b[0mins\u001b[0m\u001b[0;34m)\u001b]])]
           u001b [0m\u001b [0;34m\u001b [0m\u001b [0m\n\u001b [0m\u001b [0m\n\u001b [0m\u001b [0m\n\u001b [0m\u001b [0m\n\u001b [0m\u001b [0m\n\u001b [0m\n\u00
           [1;32m
                                            1515\u001b[0m\u001b[0;34m\u001b[0m\u001b[0m\n\]
           u001b[1;32m
                                                              1516\u001b[0m
                                                                                                                                             \u001b[0;32mif\u001b]
           \u001b[0;34m\u001b[0m\u001b[0m\n",
"\u001b[0;32m/Library/Frameworks/Python.framework/Versions"
           /2.7/lib/python2.7/site-packages/tensorflow core/python/
           keras/backend.pyc\u001b[0m in \u001b[0;36m call \u001b
           [0;34m(self, inputs)\u001b[0m\n\u001b[1;32m]]
                                                                                                                                                                                3725\u001b
           Γ0m
                                                       \u001b[0mvalue\u001b[0m \u001b[0;34m=\u001b[0]]]
          m \u001b[0mmath ops\u001b[0m\u001b[0;34m.\u001b[0m\u001b]]]
           [0\,mcast\u001b\,[0m\u001b\,[0;34m(\u001b\,[0m\u001b\,[0\,mvalue\u001b\,[0]])]]]
           u001b[0m\u001b[0;34m,\u001b[0m\u001b[0mtensor\u001b[0m\u001b]]]
           u001b[0;34m.\u001b[0m\u001b[0mdtype\u001b[0m\u001b[0;34m]
           \label{lem:u001b} $$ \u001b [0m\u001b [0m\u001b [0m\u001b [1;32m]]] $$
           3726\u001b[0m
                                                                                   \u001b[0mconverted inputs\u001b[0m\
           u001b [0;34m.\u001b [0m\u001b [0mappend\u001b [0m\u001b [0;34m.\u001b [0]]]] + u001b [0;34m.\u001b [0]] + u001b [0;34m.\u001b [
            (\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m
```

```
u001b[0;34m\u001b[0m\u001b[0m\n\u001b[0;32m]> 3727\u001b]
      [0;31m
                           \u001b[0moutputs\u001b[0m\u001b[0;34m=\u001b]]
      m \ graph \ fn\u001b[0m\u001b[0;34m(\u001b[0m\u001b[0;34m*\u001b[0;34m])])]
      u001b[0m\u001b[0mconverted inputs\u001b[0m\u001b[0;34m)\u001b[0]]
      u001b [0m\u001b [0;34m\u001b [0m\u001b [0m\n\u001b [0m\u001b]]] \\
                       3728\u001b[0m\u001b[0;34m\u001b[0m\u001b[0m\n\]
      [1:32m
                                 3729\u001b[0m
                                                                   \u001b[0;31m# EagerTensor
      u001b[1;32m
      .numpy() will often make a copy to ensure memory safety.
      u001b[0m\u001b[0;34m\u001b[0m\u001b[0;34m\u001b[0]]]]
      [0m\n",
"\u001b[0;32m/Library/Frameworks/Python.framework/Versions"
      /2.7/lib/python2.7/site-packages/tensorflow core/python/
      eager/function.pyc\u001b[0m in \u001b[0;36m call \u001b
      [0;34m(self, *args, **kwargs)\u001b[0m\n\u001b[1;32m]]
      1549\u001b[0m
                                            \u001b[0mTypeError\u001b[0m\u001b]
      [0;34m:\u001b[0m\u001b[0mFor\u001b[0m\u001b[0minvalid\u001b]]]]
      u001b[0m \u001b[0mpositional \u001b[0m \u001b[0;34m/\u001b]]]
      [0m\u001b[0mkeyword\u001b[0m\u001b[0margument\u001b[0m\w]]]
      u001b[0mcombinations \u001b[0m\u001b[0;34m.\u001b[0m\u001b]]]
      [0;34m\u001b[0m\u001b[0m\n\u001b[1;32m]
                                                                                     1550\u001b[0m
              """ n\u001b[0;32m > 1551\u001b[0;31m]
                                                                                                   \u001b
      [0;32 \, mreturn \u001b \cite{Mu001b} \cite
      .\u001b[0m\u001b[0m\call impl\u001b[0m\u001b[0;34m(\u001b[0])]])
      [0m\u001b[0margs\u001b[0m\u001b[0;34m,\u001b[0m\u001b[0]]]]]
      mkwargs \u001b [0m\u001b [0;34m) \u001b [0m\u001b [0;34m] \u001b]
      [0m\u001b[0m\n\u001b[0m\u001b[1;32m]
                                                                               1552\u001b[0m \
      u001b [0;34m\u001b [0m\u001b [0m\n\u001b [1;32m
                 \label{local_continuous} $$ \u001b[0;32mdef\u001b[0m \u001b[0m \call impl\u001b]] $$
      [0m \ u001b \ [0;34m \ (\ u001b \ [0m \ u001b \ [0 \ mself \ u001b \ [0m \ u001b \ [0 \ mself \ ]])]
      [0;34m,\u001b[0m\u001b[0margs\u001b[0m\u001b[0;34m,\u001b]]]]
      u001b[0m \u001b[0mkwargs\u001b[0m\u001b[0;34m,\u001b[0m \u001b[0m]]]])
      u001b[0mcancellation manager\u001b[0m\u001b[0;34m=\u001b]]
      [0m\u001b[0mNone\u001b[0m\u001b[0;34m)\u001b[0m\u001b]]]
      [0;34m:\u001b[0m\u001b[0;34m\u001b[0m\u001b[0m\n"]]]
"\u001b[0;32m/Library/Frameworks/Python.framework/Versions"
      /2.7/lib/python2.7/site-packages/tensorflow core/python/
      eager/function.pyc\u001b[0m in \u001b[0;36m call impl\
      u001b[0;34m(self, args, kwargs, cancellation_manager)\
      u001b[0m\n\u001b[1;32m]
                                                      1589\u001b[0m
                                                                                             raise
      TypeError (\"Keyword arguments {} unknown. Expected {}.\".
      format(\n\u001b[1;32m
                                                    1590\u001b[0m
      kwargs.keys()), list(self. arg keywords)))\n\u001b[0;32m
      -> 1591\u001b[0;31m
                                                    \u001b[0;32mreturn\u001b[0m \
```

 $u001b[0 mself \u001b[0m \u001b[0;34m. \u001b[0m \u001b[0]]]]$

```
u001b[0m\u001b[0;34m,\u001b[0m\u001b[0\,mself\u001b[0m\w001b]]])
      u001b[0;34m.\u001b[0m\u001b[0mcaptured inputs\u001b[0m\u001b]]]
      u001b[0;34m,\u001b[0m\u001b[0mcancellation manager\u001b
      [0m\u001b\[0;34m)\u001b\[0m\u001b\[0;34m\u001b\[0m\u001b\[0]]
      \u001b[0m\u001b[1;32m]
                                                    1592\u001b[0m\u001b[0;34m\u001b
      [0m\u001b[0m\n\u001b[1;32m]
                                                              1593\u001b[0m
                                                                                             \u001b[0;32
      mdef \\ \ u001b[0m \\ \ u001b[0m\_filtered\_call \\ \ u001b[0m\\ \ u001b[0;34] \\
     m(\u001b[0m\u001b[0mself\u001b[0m\u001b[0;34m,\u001b[0m\u]]]))
      u001b[0margs\u001b[0m\u001b[0;34m,\u001b[0m\u001b[0
      mkwargs \u001b[0m\u001b[0;34m) \u001b[0m\u001b[0;34m:\u001b]]
      [0m\u001b[0;34m\u001b[0m\u001b[0m\n",
"\u001b[0;32m/Library/Frameworks/Python.framework/Versions"
      /2.7/lib/python2.7/site-packages/tensorflow_core/python/
      eager/function.pyc\u001b[0m in \u001b[0;36m call flat\
      u001b[0;34m(self, args, captured inputs,
      \u001b[0;31m# No tape is watching; skip to
      running the function.\u001b[0m\u001b[0;34m\u001b[0m\u001b]]
      [0;34m\u001b[0m\u001b[0m\n\u001b[1;32m]
                                                                                     1691\u001b[0m
                  return self._build_call_outputs(self.
      _inference_function.call(\n\u001b[0;32m-> 1692\u001b[0;31
                             ctx, args, cancellation manager=
      cancellation manager))\n\u001b[0m\u001b[1;32m
                                                                                                   1693\
      u001b[0m
                               forward_backward = self.
      select forward and backward functions(\n\u001b[1;32m
      1694\u001b[0m
                                                \u001b[0margs\u001b[0m\u001b[0;34m]
      ,\u001b[0m\u001b[0;34m\u001b[0m\u001b[0m\n"]]
"\u001b[0;32m/Library/Frameworks/Python.framework/Versions"
      /2.7/lib/python2.7/site-packages/tensorflow core/python/
      eager/function.pyc\u001b[0m\ in\ \u001b[0;36mcall\\\u001b]
      [0;34m(self, ctx, args, cancellation manager)\u001b[0m\n\]
      u001b[1;32m
                                  543\u001b[0m
                                                                                       \u001b[0minputs
      \u001b[0m\u001b[0;34m=\u001b[0m\u001b[0margs\u001b[0m\u001b]]]]
      u001b[0;34m, u001b[0mu001b[0;34mu001b[0mu001b[0m]n]]]
      u001b[1;32m
                                   544\u001b[0m
                                                                                       \u001b[0mattrs]
      u001b[0m\u001b[0;34m=\u001b[0m\u001b[0;34m(\u001b[0m\u001b[0])])]]
      u001b[0;34m\ \ executor\ type\ \ \ u001b[0m\ \ u001b[0;34m,\ \ \ u001b]]
      [0m \u001b[0mexecutor\_type\u001b[0m\u001b[0;34m,\u001b[0m]]]]
        \u001b[0;34m\\config proto\\\c)u001b[0m\u001b[0;34m,\u001b]]
      [0m \u001b[0mconfig\u001b[0m\u001b[0;34m)\u001b[0m\u001b]]]
      [0;34m,\u001b[0m\u001b[0;34m\u001b[0m\u001b[0m\n\u001b]]]]
      [0;32m \rightarrow 545 \u001b[0;31m]
                                                                                   ctx=ctx)\n\u001b
      [0m\u001b[1;32m
                                          546\u001b[0m
                                                                                   \u001b[0;32 melse\
      u001b [0m\u001b [0;34m.\u001b [0m\u001b [0;34m\u001b [0m\u001b [0;34m\u001b [0]]]]] + u001b [0m\u001b [0;34m\u001b [0]]] + u001b [0m\u001b [0m\u001b [0m\u001b [0]]] + u001b [0m\u001b [0m\u001
```

```
[0m\n\u001b[1;32m]
                                                                      547\u001b[0m
                                                                                                                               outputs =
                    execute.execute_with_cancellation(\n",
           "\u001b[0;32m/Library/Frameworks/Python.framework/Versions"
                   /2.7/lib/python2.7/site-packages/tensorflow core/python/
                   eager/execute.pyc\u001b[0m in \u001b[0;36mquick execute\
                   u001b[0;34m(op name, num outputs, inputs, attrs, ctx,
                   name) \u001b[0m\n\u001b[1;32m]
                                                                                                    59\u001b[0m
                                                                                                                                            tensors
                   = pywrap_tensorflow.TFE_Py_Execute(ctx._handle,
                   device name,\n\u001b[1;32m
                                                                                               60\u001b[0m
                                                                                                                                       \u001b[0
                   mop name \u001b[0m\u001b[0;34m,\u001b[0m\u001b[0minputs]]] \\
                   u001b[0m\u001b[0;34m,\u001b[0m\u001b[0mattrs\u001b[0m\w001b]]])
                   u001b[0;34m, u001b[0mu001b[0;34mu001b[0mu001b[0m]n]]]
                   u001b[0;32m \rightarrow 61\u001b[0;31m]
                   num outputs)\n 001b[0m\u001b[1;32m]
                                                                                                                     62\u001b[0m
                   u001b[0;32mexcept\u001b[0m\u001b[0mcore\u001b[0m\u001b]]]
                    [0;34m.\u001b[0m\u001b[0m\ NotOkStatusException\u001b[0m\ \
                   u001b[0;32mas\u001b[0m\u001b[0m\u001b[0m\u001b[0;34m:\u001b]]]]
                   u001b [0m\u001b [0;34m\u001b [0m\u001b [0m\n\u001b [1;32m\u001b]]] + u001b [0,34m\u001b [0,34m\u001b]] + u001b [0,34m\u001b] + u00
                                                          \u001b[0;32mif\u001b[0m\u001b[0mname\
                   u001b[0m \u001b[0;32mis\u001b[0m \u001b[0;32mnot\u001b[0m \u001b[0;32mnot\u001b[0m \u001b[0;32mnot\u001b[0m \u001b[0;32mnot\u001b[0]]]]
                      \u001b[0mNone\u001b[0m\u001b[0;34m:\u001b[0m\u001b[0;34m:\u001b]]]]
                   \u001b[0m\u001b[0m\n"]
           "\u001b[0;31mKeyboardInterrupt\u001b[0m: "
        ]
    "source": "hist = model.fit generator(\n
                                                                                                                         train generator
                                       steps per epoch= len(train generator.filenames)
                                                 epochs = 1000, \ n
                                                                                                      shuffle = True,\n
            //64, n
                              validation_data=test_generator,\n
            validation steps= 2,\n
                                                                                     callbacks=checkpoint)"
    "cell_type": "code",
    "execution count": null,
    "metadata": {},
    "outputs": [],
    "source": ""
"metadata": {
  "kernelspec": {
    "display_name": "Python 2",
```

}],

},

}],

```
"language": "python",
   "name": "python2"
  },
  "language_info": {
   "codemirror_mode": {
   "name": "ipython",
   "version": 2
   "file_extension": ".py",
   "mimetype": "text/x-python",
   "name": "python",
   "nbconvert_exporter": "python",
   "pygments_lexer": "ipython2",
   "version": "2.7.16"
  }
 },
 "nbformat": 4,
 "nbformat_minor": 4
}
```