tablenet final

April 20, 2021

1 TableNet: Deep Learning model for end-to-end Table detection and Tabular data extraction from Scanned Document Images

With the widespread use of mobile phones and scanners to photograph and upload documents, the need for extracting the information trapped in unstructured document images such as retail receipts, insurance claim forms and financial invoices is becoming more acute.

A major hurdle to this objective is that these images often contain information in the form of tables and extracting data from tabular sub-images presents a unique set of challenges. This includes accurate detection of the tabular region within an image, and subsequently detecting and extracting information from the rows and columns of the detected table.

While some progress has been made in table detection, extracting the table contents is still a challenge since this involves more fine grained table structure(rows & columns) recognition. Prior approaches have attempted to solve the table detection and structure recognition problems independently using two separate models.

TableNet: a novel end-to- end deep learning model for both table detection and structure recognition. The model exploits the interdependence between the twin tasks of table detection and table structure recognition to segment out the table and column regions.

2 problem statement

We need to detect tabular structure from a document image, then extract tabular information (row-col format).

3 data source

https://drive.google.com/drive/folders/1410iMmQCXbA9GJP5CqLEMfjjv-hOWlac?usp=sharing

4 ML/DL problem type

classification - we need to predict each pixel either white(background) or black(tabular region).

5 Performance Metric

1. precision
2. recall

3. f1 score

6 google drive mount

[1]:
```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

7 dependencies

[2]:
```
!sudo apt install tesseract-ocr
!pip install pytesseract

import warnings
warnings.filterwarnings('ignore')

import tensorflow as tf
import os
import matplotlib.pyplot as plt
import numpy as np
import cv2
import xml.etree.ElementTree as ET
from PIL import Image
import pandas as pd
import pytesseract
from sklearn.model_selection import train_test_split
import tensorflow as tf
from google.colab.patches import cv2_imshow
```

```
Reading package lists… Done
Building dependency tree
Reading state information… Done
The following additional packages will be installed:
  tesseract-ocr-eng tesseract-ocr-osd
The following NEW packages will be installed:
  tesseract-ocr tesseract-ocr-eng tesseract-ocr-osd
0 upgraded, 3 newly installed, 0 to remove and 31 not upgraded.
Need to get 4,795 kB of archives.
After this operation, 15.8 MB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu bionic/universe amd64 tesseract-ocr-eng
all 4.00~git24-0e00fe6-1.2 [1,588 kB]
Get:2 http://archive.ubuntu.com/ubuntu bionic/universe amd64 tesseract-ocr-osd
all 4.00~git24-0e00fe6-1.2 [2,989 kB]
Get:3 http://archive.ubuntu.com/ubuntu bionic/universe amd64 tesseract-ocr amd64
4.00~git2288-10f4998a-2 [218 kB]
```

```
Fetched 4,795 kB in 2s (2,860 kB/s)
debconf: unable to initialize frontend: Dialog
debconf: (No usable dialog-like program is installed, so the dialog based
frontend cannot be used. at /usr/share/perl5/Debconf/FrontEnd/Dialog.pm line 76,
<> line 3.)
debconf: falling back to frontend: Readline
debconf: unable to initialize frontend: Readline
debconf: (This frontend requires a controlling tty.)
debconf: falling back to frontend: Teletype
dpkg-preconfigure: unable to re-open stdin:
Selecting previously unselected package tesseract-ocr-eng.
(Reading database … 160983 files and directories currently installed.)
Preparing to unpack …/tesseract-ocr-eng_4.00~git24-0e00fe6-1.2_all.deb …
Unpacking tesseract-ocr-eng (4.00~git24-0e00fe6-1.2) …
Selecting previously unselected package tesseract-ocr-osd.
Preparing to unpack …/tesseract-ocr-osd_4.00~git24-0e00fe6-1.2_all.deb …
Unpacking tesseract-ocr-osd (4.00~git24-0e00fe6-1.2) …
Selecting previously unselected package tesseract-ocr.
Preparing to unpack …/tesseract-ocr_4.00~git2288-10f4998a-2_amd64.deb …
Unpacking tesseract-ocr (4.00~git2288-10f4998a-2) …
Setting up tesseract-ocr-osd (4.00~git24-0e00fe6-1.2) …
Setting up tesseract-ocr-eng (4.00~git24-0e00fe6-1.2) …
Setting up tesseract-ocr (4.00~git2288-10f4998a-2) …
Processing triggers for man-db (2.8.3-2ubuntu0.1) …
Collecting pytesseract
  Downloading https://files.pythonhosted.org/packages/a0/e6/a4e9fc8a93c1318540e8
de6d8d4beb5749b7960388a7c7f27799fc2dd016/pytesseract-0.3.7.tar.gz
Requirement already satisfied: Pillow in /usr/local/lib/python3.7/dist-packages
(from pytesseract) (7.1.2)
Building wheels for collected packages: pytesseract
  Building wheel for pytesseract (setup.py) … done
  Created wheel for pytesseract: filename=pytesseract-0.3.7-py2.py3-none-any.whl
size=13945
sha256=48bbfd21121b97c690b2ca92b9558dbc4ca5b70ac92e0092152525045099beca
  Stored in directory: /root/.cache/pip/wheels/81/20/7e/1dd0daad1575d5260916bb1e
9781246430647adaef4b3ca3b3
Successfully built pytesseract
Installing collected packages: pytesseract
Successfully installed pytesseract-0.3.7
```

8 paths

[3]:
```
originalImage = "/content/drive/MyDrive/case study - II/tablenet/data/
 ↪Marmot_data/10.1.1.1.2006_3.bmp"
imageMask = "/content/drive/MyDrive/case study - II/tablenet/data/Marmot_data/
 ↪10.1.1.1.2006_3.xml"
```

```
fileSavepath = "/content/drive/MyDrive/case study - II/tablenet/data/final data/
 ↪"
table_mask_path = "/content/drive/MyDrive/case study - II/tablenet/data/final␣
 ↪data/tablemask/"
col_mask_path = "/content/drive/MyDrive/case study - II/tablenet/data/final␣
 ↪data/colmask/"
org_image_path = "/content/drive/MyDrive/case study - II/tablenet/data/final␣
 ↪data/orgimage/"
dataPath = "/content/drive/MyDrive/case study - II/tablenet/data/Marmot_data/"
```

9 creating dataframe of image and xml files

[ ]:
```
"""
CREATE DATAFRAME OF PATHS.
dataframe
---------
image_path, xml_path

* go through every file in mamoth folder (dataPath).
* check a .bmp file, extract name, check if .xml file is present or not -->␣
 ↪store in row
"""

image_xml_dict = {"image_path":[], "xml_path":[]}

for file in os.listdir(dataPath):
  if ".bmp" in file:
    name = file.split(".bmp")[0]
    if os.path.exists(dataPath+name+".xml"):
      image_xml_dict['image_path'].append(name+".bmp")
      image_xml_dict['xml_path'].append(name+".xml")


image_xml_df = pd.DataFrame(image_xml_dict)

image_xml_df.head(2)
```

[ ]:

	image_path	xml_path
0	10.1.1.1.2044_7.bmp	10.1.1.1.2044_7.xml
1	10.1.1.34.330_9.bmp	10.1.1.34.330_9.xml

10 creating mask from .xml files

10.1 xml file for every image defining regions of every table in an image.

1. 'size' element denotes the height, width and depth of image.

2. ‘obejct’ elelment denotes every single col region in a image.

[ ]:
```
"""
<size>
                <width>793</width>
                <height>1123</height>
                <depth>3</depth>
</size>


<object>
                <name>column</name>
                <pose>Unspecified</pose>
                <truncated>0</truncated>
                <difficult>0</difficult>
                <bndbox>
                        <xmin>458</xmin>
                        <ymin>710</ymin>
                        <xmax>517</xmax>
                        <ymax>785</ymax>
                </bndbox>
        </object>


"""

# /content/drive/MyDrive/case study - II/tablenet/data/final data/

def euc_dist(point1, point2):
    dist = np.linalg.norm(point1 - point2)
    return dist

def show_image_plt(image_arr):
  plt.figure(figsize=(5,5))
  plt.imshow(image_arr)
  plt.show()

def save_image(name, image_arr):
  im = Image.fromarray(image_arr)
  im.save(name)


final_dataframe_dict = {"image":[], "table_mask":[], "col_mask":[]}

for index, row in image_xml_df.iterrows():
```

```
  # per row --> xml_path
  org_img_mask_xml = row['xml_path'] # .xml path
  image = dataPath + row['image_path'] # image .bmp path

  # file name
  name = org_img_mask_xml.split(".xml")[0]


  # reading xml file
  tree = ET.parse(dataPath + org_img_mask_xml)
  root = tree.getroot()


  size = root.find('size')
  width = int(size.find('width').text)
  height = int(size.find('height').text)
  depth = int(size.find('depth').text)

  # creating empty mask image
  col_mask_empty = np.zeros(shape=(height, width), dtype=np.uint8)
  table_mask_empty = np.zeros(shape=(height, width), dtype=np.uint8)

# finding objects
  objects = tree.findall('object')
  table_xmin = 0
  table_ymin = 0
  table_xmax = 0
  table_ymax = 0
  prev_dist = 0
  dist = 0
  forward_flag = False
  backward_flag = False
  newtable_flag = True

  # creating empty mask image
  col_mask_empty = np.zeros(shape=(height, width), dtype=np.uint8)
  table_mask_empty = np.zeros(shape=(height, width), dtype=np.uint8)

  plt.figure(figsize=(5, 5))

  objects = tree.findall('object')

  for index, object in enumerate(objects):

      bndbox = object.find('bndbox')
```

```
        xmin = int(bndbox.find('xmin').text)
        xmax = int(bndbox.find('xmax').text)
        ymin = int(bndbox.find('ymin').text)
        ymax = int(bndbox.find('ymax').text)

        col_mask_empty[ymin:ymax, xmin:xmax] = 255



        if index == 0:

            prev_xmin = int(bndbox.find('xmin').text)
            prev_ymin = int(bndbox.find('ymin').text)
            prev_xmax = int(bndbox.find('xmax').text)
            prev_ymax = int(bndbox.find('ymax').text)


        else:


            if xmin > prev_xmin and newtable_flag:


                table_xmin = prev_xmin
                table_ymin = prev_ymin
                newtable_flag = False
                forward_flag = True
                backward_flag = False

            if xmin < prev_xmin and newtable_flag:


                table_xmax = prev_xmax
                table_ymax = prev_ymax


                newtable_flag = False
                backward_flag = True
                forward_flag = False



            if forward_flag:
                dist = euc_dist(np.array([xmin, ymin]), np.array([prev_xmax,␣
↪prev_ymin]))

                if prev_dist == 0:
```

```
                    prev_dist = dist
                else:

                    if int(np.divide(dist, prev_dist)) > 5:
                        newtable_flag = True
                        table_mask_empty[table_ymin:prev_ymax, table_xmin:
↪prev_xmax] = 255

                        prev_dist = 0

                    if index==len(objects)-1:
                        newtable_flag = True
                        table_mask_empty[table_ymin:ymax, table_xmin:xmax] = 255

                        prev_dist = 0

            if backward_flag:
                dist = euc_dist(np.array([xmax, ymin]), np.array([prev_xmin,␣
↪prev_ymin]))

                if prev_dist == 0:
                    prev_dist = dist
                else:
                    if int(np.divide(dist, prev_dist)) > 5 or␣
↪index==len(objects)-1:
                        newtable_flag = True
                        table_mask_empty[ymin:table_ymax, xmin:table_xmax] = 255
                        prev_dist = 0

            prev_xmin = int(bndbox.find('xmin').text)
            prev_ymin = int(bndbox.find('ymin').text)
            prev_xmax = int(bndbox.find('xmax').text)
            prev_ymax = int(bndbox.find('ymax').text)
            prev_dist = dist


    save_image(table_mask_path+ name+".jpeg", table_mask_empty)
    save_image(col_mask_path + name+".jpeg", col_mask_empty)

    final_dataframe_dict['table_mask'].append(table_mask_path+ name+".jpeg")
    final_dataframe_dict['col_mask'].append(col_mask_path + name+".jpeg")
    final_dataframe_dict['image'].append(image)



# creating dataframe --> (oroginal_image, table_mask, col_mask)
final_dataframe = pd.DataFrame(final_dataframe_dict)
```

```
final_dataframe.head(2)
final_dataframe.to_csv("/content/drive/MyDrive/case study - II/tablenet/data/final_dataframe.csv", index=False)
```

```
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
```

```
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
```

```
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
```

```
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
```

```
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
```

```
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
```

```
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
```

```
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
```

```
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
```

```
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
```

```
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
```

```
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
```

```
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
```

```
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
```

```
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
```

```
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
<Figure size 360x360 with 0 Axes>
```

11 reading dataframe

[ ]:
```
final_dataframe = pd.read_csv("/content/drive/MyDrive/case study - II/tablenet/
 ↪data/final_dataframe.csv")
final_dataframe.head(2)
```

[ ]:
```
                                                  image  …
col_mask
0  /content/drive/MyDrive/case study - II/tablene…  …
/content/drive/MyDrive/case study - II/tablene…
1  /content/drive/MyDrive/case study - II/tablene…  …
/content/drive/MyDrive/case study - II/tablene…

[2 rows x 3 columns]
```

12 data generator

[ ]:
```
X_train, X_test = train_test_split(final_dataframe, test_size=0.2)
```

[ ]:
```
training_dataset = (
    tf.data.Dataset.from_tensor_slices(
        (
            tf.cast(X_train['image'].values, tf.string),
            tf.cast(X_train['table_mask'].values, tf.string),
            tf.cast(X_train['col_mask'].values, tf.string),
        )
    )
)



testing_dataset = (
    tf.data.Dataset.from_tensor_slices(
        (
            tf.cast(X_test['image'].values, tf.string),
            tf.cast(X_test['table_mask'].values, tf.string),
            tf.cast(X_test['col_mask'].values, tf.string),
        )
    )
)
```

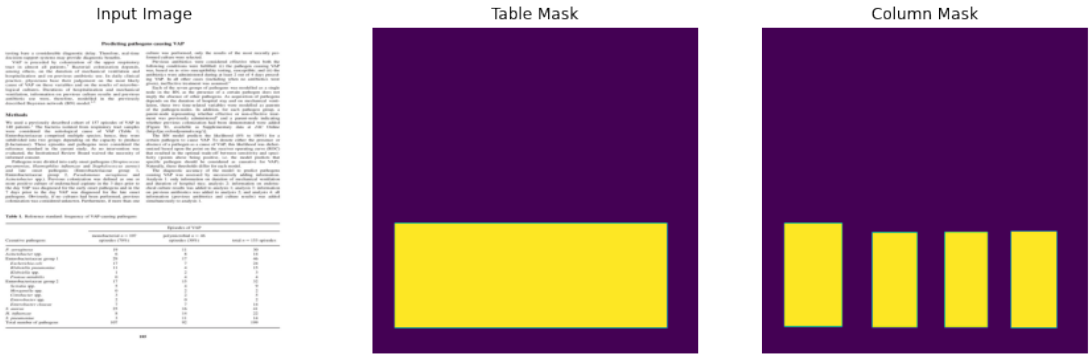
[ ]:
```
# https://www.tensorflow.org/tutorials/load_data/images

@tf.function
def load_image(image, table_mask, col_mask):

    image = tf.io.read_file(image)
    table_mask=tf.io.read_file(table_mask)
    col_mask=tf.io.read_file(col_mask)

    image=tf.io.decode_bmp(image, channels=3)
    image=tf.image.resize(image, [1024, 1024])
    image = tf.cast(image, tf.float32) / 255.0

    table_mask=tf.io.decode_jpeg(table_mask, channels=1)
    table_mask=tf.image.resize(table_mask, [1024, 1024])
    table_mask = table_mask / 255.0


    col_mask=tf.io.decode_jpeg(col_mask, channels=1)
    col_mask=tf.image.resize(col_mask, [1024, 1024])
    col_mask = col_mask / 255.0

    return image, {"table_mask":table_mask, "col_mask":col_mask}
```

```
# creating dataset object
train = training_dataset.map(load_image, num_parallel_calls=tf.data.AUTOTUNE)
test = testing_dataset.map(load_image)
```

[ ]:
```
BATCH_SIZE = 2
BUFFER_SIZE = 10
train_steps = len(X_train) // BATCH_SIZE

# for feeding to training
train_dataset = train.cache().shuffle(BUFFER_SIZE).batch(BATCH_SIZE).repeat()
train_dataset = train_dataset.prefetch(buffer_size=tf.data.experimental.
 ↪AUTOTUNE)
test_dataset = test.batch(BATCH_SIZE)
```

[ ]:
```
def display(display_list):
    plt.figure(figsize=(15, 15))
    title = ['Input Image', 'Table Mask', 'Column Mask', 'Masked image']
    for i in range(len(display_list)):
        plt.subplot(1, len(display_list), i+1)
        plt.title(title[i])

        image = display_list[i]

        plt.imshow(tf.keras.preprocessing.image.array_to_img(image))
        plt.axis('off')
    plt.show()



for image, mask in train.take(1):

    sample_image = image
    sample_table_mask = mask['table_mask']
    sample_col_mask = mask['col_mask']


    print(image.shape)
    print(mask['table_mask'].shape)
    print(mask['col_mask'].shape)
    display([image, mask['table_mask'], mask['col_mask']])
```

```
(1024, 1024, 3)
(1024, 1024, 1)
(1024, 1024, 1)
```



13 model

[4]:
```
import tensorflow as tf
from tensorflow.keras.applications import VGG19
from tensorflow.keras.layers import Input
from tensorflow.keras.models import Model
from tensorflow.keras.layers import Conv2D
from tensorflow.keras.layers import Dropout
from tensorflow.keras.layers import UpSampling2D
from tensorflow.keras.layers import Concatenate
from tensorflow.keras.layers import Layer
from tensorflow.keras.layers import Activation
from tensorflow.keras.layers import Conv2DTranspose
from tensorflow.keras.utils import plot_model
from tensorflow.keras import backend as K
```

[5]:
```
"""
Table decoder
-------------
x = conv7(1x1)
x = x(upscaled) + vgg19(pool4)
x = x(upscaled) + vgg19(pool3)
x = upscaled to match input dimention (1024)


column decoder
--------------
x = conv7(1x1, relu)
x = dropout(0.8)
x = conv8(1x1)
x = upscaled() + vgg19(pool4)
x = x(upscaled) + vgg19(pool3)
x = upscaled (1024)
```

```
"""


tf.keras.backend.clear_session()


class table_mask(Layer):

    def __init__(self):
        super().__init__()
        self.conv_7 = Conv2D(kernel_size=(1,1), filters=128,␣
↪kernel_regularizer=tf.keras.regularizers.l2(0.002))
        self.upsample_pool4 = UpSampling2D(size=(2, 2),␣
↪interpolation='bilinear')
        self.upsample_pool3 = UpSampling2D(size=(2, 2),␣
↪interpolation='bilinear')
        self.upsample_final = Conv2DTranspose(filters=2, kernel_size=3,␣
↪strides=2, padding='same', activation='softmax')

    def call(self, input, pool3, pool4):

        x = self.conv_7(input)
        x = self.upsample_pool4(x)
        x = Concatenate()([x, pool4])

        x = self.upsample_pool3(x)
        x = Concatenate()([x, pool3])

        x = UpSampling2D((2,2))(x)
        x = UpSampling2D((2,2))(x)

        x = self.upsample_final(x)



        return x

class col_mask(Layer):

    def __init__(self):
        super().__init__()
        self.conv_7 = Conv2D(kernel_size=(1,1), filters=128,␣
↪kernel_regularizer=tf.keras.regularizers.l2(0.004),␣
↪kernel_initializer='he_normal',)
        self.drop = Dropout(0.8)
```

```
        self.conv_8 = Conv2D(kernel_size=(1,1), filters=128,␣
↪kernel_regularizer=tf.keras.regularizers.l2(0.004),␣
↪kernel_initializer='he_normal',)
        self.upsample_pool4 = UpSampling2D(size=(2, 2),␣
↪interpolation='bilinear')
        self.upsample_pool3 = UpSampling2D(size=(2, 2),␣
↪interpolation='bilinear')
        self.upsample_final = Conv2DTranspose(filters=2, kernel_size=3,␣
↪strides=2, padding='same', activation='softmax')

    def call(self, input, pool3, pool4):

        x = self.conv_7(input)
        x = self.drop(x)
        x = self.conv_8(x)

        x = self.upsample_pool4(x)
        x = Concatenate()([x, pool4])

        x = self.upsample_pool3(x)
        x = Concatenate()([x, pool3])

        x = UpSampling2D((2,2))(x)
        x = UpSampling2D((2,2))(x)

        x = self.upsample_final(x)



        return x



input_shape = (1024, 1024, 3)
input_ = Input(shape=input_shape)

vgg19_ = VGG19(
    include_top=False,
    weights="imagenet",
    input_tensor=input_,
    input_shape=None,
    pooling=None,
    classes=1000,
    classifier_activation="softmax",
)

for layer in vgg19_.layers:
```

```
    layer.trainable = False

pool3 = vgg19_.get_layer('block3_pool').output
pool4 = vgg19_.get_layer('block4_pool').output

conv_1_1_1 = Conv2D(filters=128, kernel_size=(1, 1), activation='relu',␣
 ↪name="block6_conv1", kernel_regularizer=tf.keras.regularizers.l2(0.
 ↪004))(vgg19_.output)
conv_1_1_1_drop = Dropout(0.8)(conv_1_1_1)

conv_1_1_2 = Conv2D(filters=128, kernel_size=(1, 1), activation='relu',␣
 ↪name="block6_conv2", kernel_regularizer=tf.keras.regularizers.l2(0.
 ↪004))(conv_1_1_1_drop)
conv_1_1_2_drop = Dropout(0.8)(conv_1_1_2)

table_mask = table_mask()(conv_1_1_2_drop, pool3, pool4)
col_mask = col_mask()(conv_1_1_2_drop, pool3, pool4)

model = Model(input_, [table_mask, col_mask])

model.summary()
```

```
Downloading data from https://storage.googleapis.com/tensorflow/keras-
applications/vgg19/vgg19_weights_tf_dim_ordering_tf_kernels_notop.h5
80142336/80134624 [==============================] - 0s 0us/step
Model: "model"
_________________________________________________________________________________
__________________
Layer (type)                 Output Shape              Param #     Connected to
=================================================================================
==================
input_1 (InputLayer)         [(None, 1024, 1024,  0
_________________________________________________________________________________
__________________
block1_conv1 (Conv2D)        (None, 1024, 1024, 6 1792        input_1[0][0]
_________________________________________________________________________________
__________________
block1_conv2 (Conv2D)        (None, 1024, 1024, 6 36928
block1_conv1[0][0]
_________________________________________________________________________________
__________________
block1_pool (MaxPooling2D)   (None, 512, 512, 64) 0
block1_conv2[0][0]
_________________________________________________________________________________
__________________
block2_conv1 (Conv2D)        (None, 512, 512, 128 73856
block1_pool[0][0]
```

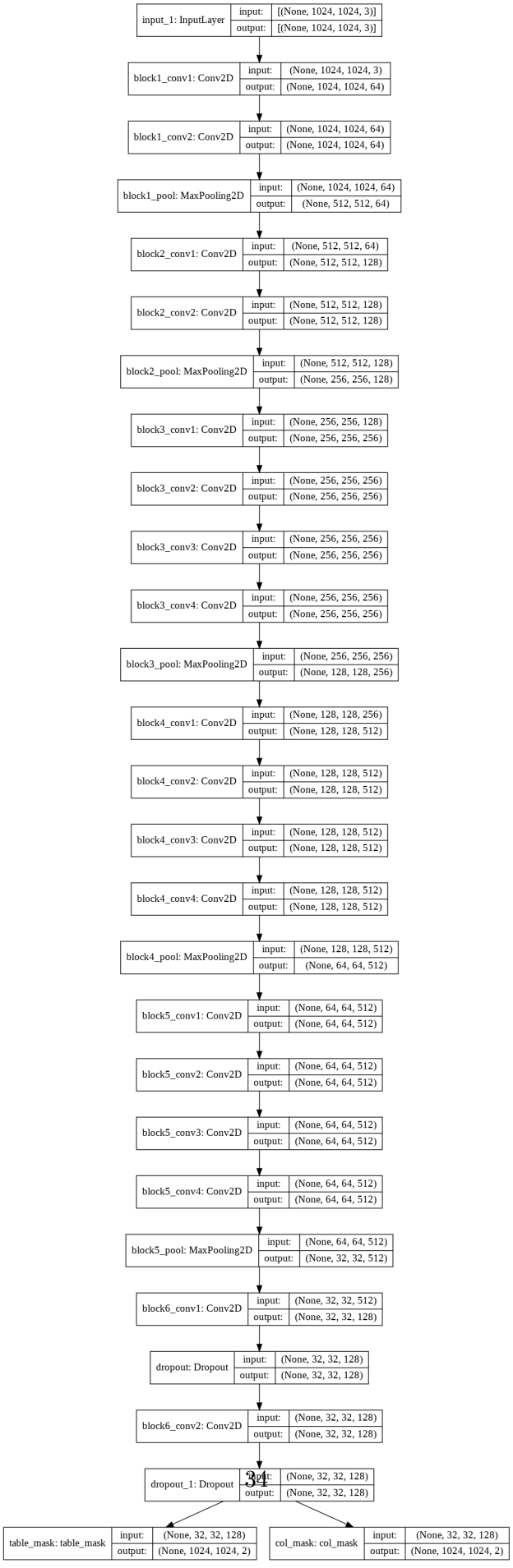
```
_____________________________________________________________________________
_________________
block2_conv2 (Conv2D)           (None, 512, 512, 128 147584
block2_conv1[0][0]
_____________________________________________________________________________
_________________
block2_pool (MaxPooling2D)      (None, 256, 256, 128 0
block2_conv2[0][0]
_____________________________________________________________________________
_________________
block3_conv1 (Conv2D)           (None, 256, 256, 256 295168
block2_pool[0][0]
_____________________________________________________________________________
_________________
block3_conv2 (Conv2D)           (None, 256, 256, 256 590080
block3_conv1[0][0]
_____________________________________________________________________________
_________________
block3_conv3 (Conv2D)           (None, 256, 256, 256 590080
block3_conv2[0][0]
_____________________________________________________________________________
_________________
block3_conv4 (Conv2D)           (None, 256, 256, 256 590080
block3_conv3[0][0]
_____________________________________________________________________________
_________________
block3_pool (MaxPooling2D)      (None, 128, 128, 256 0
block3_conv4[0][0]
_____________________________________________________________________________
_________________
block4_conv1 (Conv2D)           (None, 128, 128, 512 1180160
block3_pool[0][0]
_____________________________________________________________________________
_________________
block4_conv2 (Conv2D)           (None, 128, 128, 512 2359808
block4_conv1[0][0]
_____________________________________________________________________________
_________________
block4_conv3 (Conv2D)           (None, 128, 128, 512 2359808
block4_conv2[0][0]
_____________________________________________________________________________
_________________
block4_conv4 (Conv2D)           (None, 128, 128, 512 2359808
block4_conv3[0][0]
_____________________________________________________________________________
_________________
block4_pool (MaxPooling2D)      (None, 64, 64, 512)  0
block4_conv4[0][0]
```

```
__________________________________________________________________________________
__________________
block5_conv1 (Conv2D)           (None, 64, 64, 512)  2359808
block4_pool[0][0]
__________________________________________________________________________________
__________________
block5_conv2 (Conv2D)           (None, 64, 64, 512)  2359808
block5_conv1[0][0]
__________________________________________________________________________________
__________________
block5_conv3 (Conv2D)           (None, 64, 64, 512)  2359808
block5_conv2[0][0]
__________________________________________________________________________________
__________________
block5_conv4 (Conv2D)           (None, 64, 64, 512)  2359808
block5_conv3[0][0]
__________________________________________________________________________________
__________________
block5_pool (MaxPooling2D)      (None, 32, 32, 512)  0
block5_conv4[0][0]
__________________________________________________________________________________
__________________
block6_conv1 (Conv2D)           (None, 32, 32, 128)  65664
block5_pool[0][0]
__________________________________________________________________________________
__________________
dropout (Dropout)               (None, 32, 32, 128)  0
block6_conv1[0][0]
__________________________________________________________________________________
__________________
block6_conv2 (Conv2D)           (None, 32, 32, 128)  16512       dropout[0][0]
__________________________________________________________________________________
__________________
dropout_1 (Dropout)             (None, 32, 32, 128)  0
block6_conv2[0][0]
__________________________________________________________________________________
__________________
table_mask (table_mask)         (None, 1024, 1024, 2 32642       dropout_1[0][0]
block3_pool[0][0]
block4_pool[0][0]
__________________________________________________________________________________
__________________
col_mask (col_mask)             (None, 1024, 1024, 2 49154       dropout_1[0][0]
block3_pool[0][0]
block4_pool[0][0]
==================================================================================
==================
Total params: 20,188,356
```

```
Trainable params: 163,972
Non-trainable params: 20,024,384
_________________________________________________________________________________
__________________
```

[6]:
```
plot_model(model,show_shapes=True,show_layer_names=True)
```

[6]:


input_1: InputLayer | input: [(None, 1024, 1024, 3)] | output: [(None, 1024, 1024, 3)]
block1_conv1: Conv2D | input: (None, 1024, 1024, 3) | output: (None, 1024, 1024, 64)
block1_conv2: Conv2D | input: (None, 1024, 1024, 64) | output: (None, 1024, 1024, 64)
block1_pool: MaxPooling2D | input: (None, 1024, 1024, 64) | output: (None, 512, 512, 64)
block2_conv1: Conv2D | input: (None, 512, 512, 64) | output: (None, 512, 512, 128)
block2_conv2: Conv2D | input: (None, 512, 512, 128) | output: (None, 512, 512, 128)
block2_pool: MaxPooling2D | input: (None, 512, 512, 128) | output: (None, 256, 256, 128)
block3_conv1: Conv2D | input: (None, 256, 256, 128) | output: (None, 256, 256, 256)
block3_conv2: Conv2D | input: (None, 256, 256, 256) | output: (None, 256, 256, 256)
block3_conv3: Conv2D | input: (None, 256, 256, 256) | output: (None, 256, 256, 256)
block3_conv4: Conv2D | input: (None, 256, 256, 256) | output: (None, 256, 256, 256)
block3_pool: MaxPooling2D | input: (None, 256, 256, 256) | output: (None, 128, 128, 256)
block4_conv1: Conv2D | input: (None, 128, 128, 256) | output: (None, 128, 128, 512)
block4_conv2: Conv2D | input: (None, 128, 128, 512) | output: (None, 128, 128, 512)
block4_conv3: Conv2D | input: (None, 128, 128, 512) | output: (None, 128, 128, 512)
block4_conv4: Conv2D | input: (None, 128, 128, 512) | output: (None, 128, 128, 512)
block4_pool: MaxPooling2D | input: (None, 128, 128, 512) | output: (None, 64, 64, 512)
block5_conv1: Conv2D | input: (None, 64, 64, 512) | output: (None, 64, 64, 512)
block5_conv2: Conv2D | input: (None, 64, 64, 512) | output: (None, 64, 64, 512)
block5_conv3: Conv2D | input: (None, 64, 64, 512) | output: (None, 64, 64, 512)
block5_conv4: Conv2D | input: (None, 64, 64, 512) | output: (None, 64, 64, 512)
block5_pool: MaxPooling2D | input: (None, 64, 64, 512) | output: (None, 32, 32, 512)
block6_conv1: Conv2D | input: (None, 32, 32, 512) | output: (None, 32, 32, 128)
dropout: Dropout | input: (None, 32, 32, 128) | output: (None, 32, 32, 128)
block6_conv2: Conv2D | input: (None, 32, 32, 128) | output: (None, 32, 32, 128)
dropout_1: Dropout | input: (None, 32, 32, 128) | output: (None, 32, 32, 128)
table_mask: table_mask | input: (None, 32, 32, 128) | output: (None, 1024, 1024, 2)
col_mask: col_mask | input: (None, 32, 32, 128) | output: (None, 1024, 1024, 2)

14 model trainning

[ ]:

[ ]:
```
losses = {
    "table_mask": 'sparse_categorical_crossentropy',
    "col_mask": 'sparse_categorical_crossentropy',
}

# tf.keras.losses.SparseCategoricalCrossentropy(from_logits=False)



filepath = "/content/drive/MyDrive/case study - II/tablenet/model checkpoint/
 ↪table_net.h5"
model_checkpoint = tf.keras.callbacks.ModelCheckpoint(filepath, monitor =␣
 ↪"val_table_mask_loss", save_best_only=True, verbose = 0, mode="min")

es = tf.keras.callbacks.EarlyStopping(monitor='val_loss', mode='min',␣
 ↪patience=5,)


class F1_Score(tf.keras.metrics.Metric):

    def __init__(self, name='f1_score', **kwargs):
        super().__init__(name=name, **kwargs)
        self.f1 = self.add_weight(name='f1', initializer='zeros')
        self.precision_fn = tf.keras.metrics.Precision(thresholds=0.5)
        self.recall_fn = tf.keras.metrics.Recall(thresholds=0.5)

    def update_state(self, y_true, y_pred, sample_weight=None):
        p = self.precision_fn(y_true, tf.argmax(y_pred, axis=-1))
        r = self.recall_fn(y_true, tf.argmax(y_pred, axis=-1))
        # since f1 is a variable, we use assign
        self.f1.assign(2 * ((p * r) / (p + r + 1e-6)))

    def result(self):
        return self.f1

    def reset_states(self):
        # we also need to reset the state of the precision and recall objects
        self.precision_fn.reset_states()
        self.recall_fn.reset_states()
```

```
        self.f1.assign(0)


metrics = [F1_Score()]

global init_lr
init_lr = 0.0001

model.compile(optimizer=tf.keras.optimizers.Adam(learning_rate=init_lr,␣
↪epsilon=1e-8,),
              loss=losses,
              metrics=metrics, )
```

[ ]:
```
def show_predictions(dataset=None, num=1):

    if dataset:
        for image, mask in dataset.take(1):
            table_mask_pred, col_mask_pred = model.predict(image)

            table_mask_pred = tf.argmax(table_mask_pred, axis=-1)
            table_mask_pred = table_mask_pred[..., tf.newaxis][0]

            col_mask_pred = tf.argmax(col_mask_pred, axis=-1)
            col_mask_pred = col_mask_pred[..., tf.newaxis][0]

            im=tf.keras.preprocessing.image.array_to_img(image[0])
            im.save('image.png')

            im=tf.keras.preprocessing.image.array_to_img(table_mask_pred)
            im.save('table_mask_pred.png')

            im=tf.keras.preprocessing.image.array_to_img(col_mask_pred)
            im.save('col_mask_pred.png')



            img_org = Image.open('./image.png')
            table_mask = Image.open('./table_mask_pred.png')
            col_mask = Image.open('./col_mask_pred.png')


            # convert images
            img_mask = table_mask.convert('L')
            # img_mask = col_mask.convert('L')

            # grayscale
            # add alpha channel
```

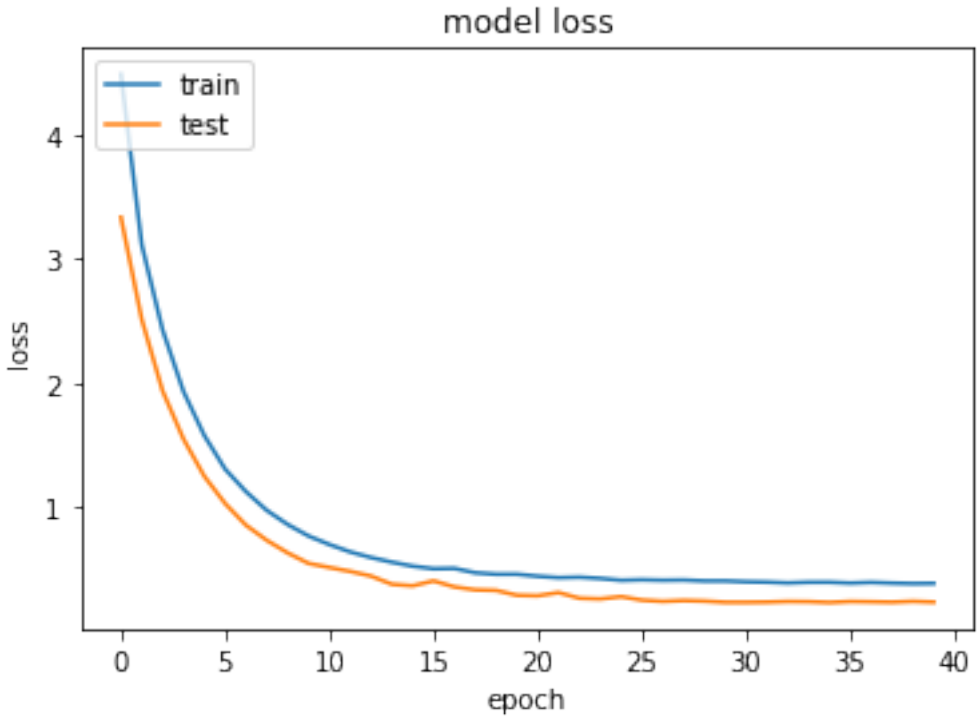
```
            img_org.putalpha(img_mask)

            # save as png which keeps alpha channel
            img_org.save('output.png')


            display([image[0], table_mask_pred, col_mask_pred, img_org])

            pytesseract.pytesseract.tesseract_cmd = r'/usr/bin/tesseract'
            text = pytesseract.image_to_string(Image.open('./output.png'),␣
↪lang='eng' ) # config='--psm 11'
            print(text)



class DisplayCallback(tf.keras.callbacks.Callback):

    def __init__(self):
        self.history = {'val_table_mask_loss':[]}
        self.init_lr = init_lr

    def on_epoch_end(self, epoch, logs=None):
        if epoch % 1 == 0:
            show_predictions(test_dataset, 1)


            self.history['val_table_mask_loss'].append(logs.
↪get('val_table_mask_loss'))
            if epoch > 2:
                cur_loss = self.history['val_table_mask_loss'][epoch]
                prev_loss = self.history['val_table_mask_loss'][epoch-1]

                if cur_loss > prev_loss:
                    self.init_lr = self.init_lr * 0.93
                    K.set_value(self.model.optimizer.learning_rate, self.
↪init_lr)
```

[ ]:
```
EPOCHS = 50
VAL_SUBSPLITS = 30
VALIDATION_STEPS = len(X_test)//BATCH_SIZE//VAL_SUBSPLITS

history = model.fit(train_dataset,
                              epochs=EPOCHS,
                              steps_per_epoch=train_steps,
                              validation_data=test_dataset,
                              validation_steps=VALIDATION_STEPS,
```

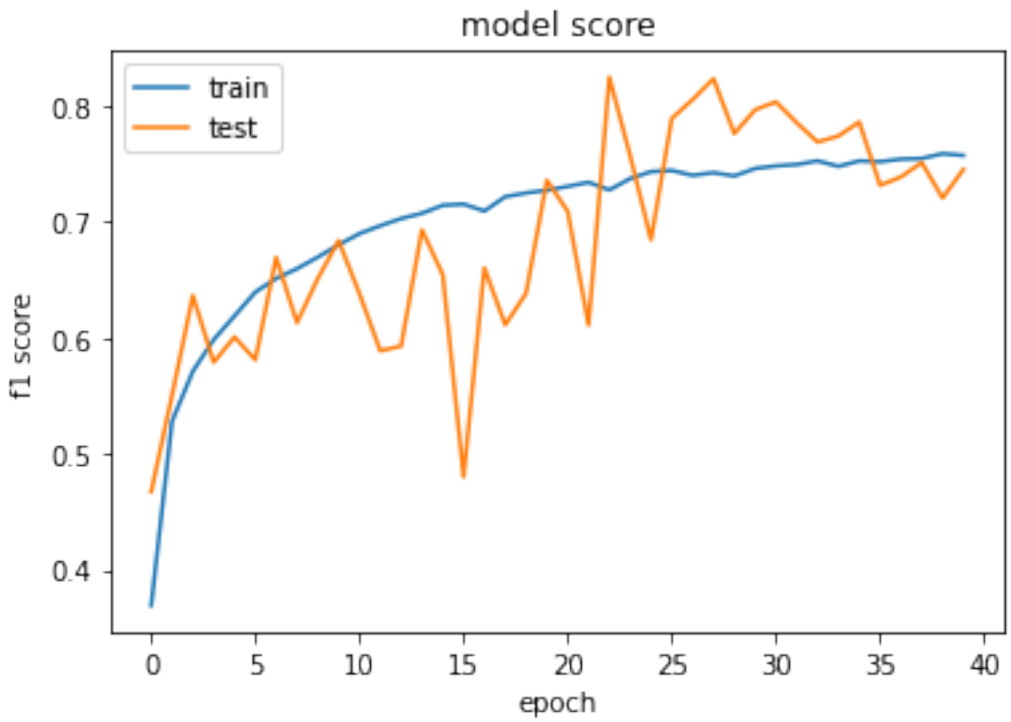
```
                                callbacks=[model_checkpoint, es,␣
↪DisplayCallback()])
```

Output hidden; open in https://colab.research.google.com to view.

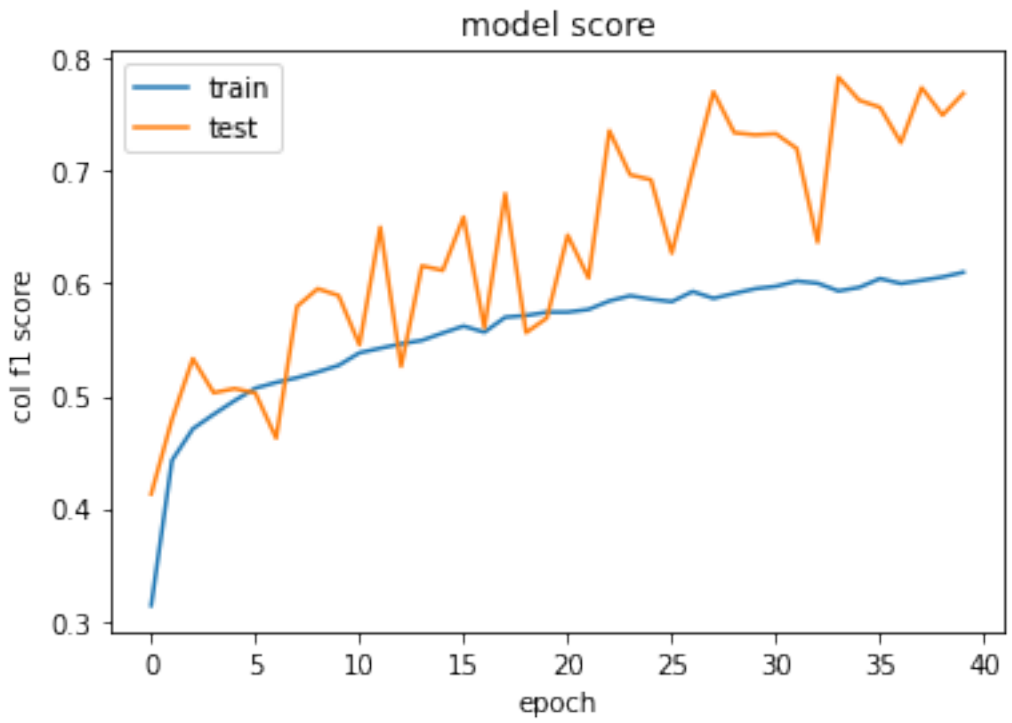
[ ]:
```
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('model loss')
plt.ylabel('loss')
plt.xlabel('epoch')
plt.legend(['train', 'test'], loc='upper left')
plt.show()
```



[ ]:
```
plt.plot(history.history['table_mask_f1_score'])
plt.plot(history.history['val_table_mask_f1_score'])
plt.title('model score')
plt.ylabel('table f1 score')
plt.xlabel('epoch')
plt.legend(['train', 'test'], loc='upper left')
plt.show()
```



[ ]:
```
plt.plot(history.history['col_mask_f1_score'])
plt.plot(history.history['val_col_mask_f1_score'])
plt.title('model score')
plt.ylabel('col f1 score')
plt.xlabel('epoch')
plt.legend(['train', 'test'], loc='upper left')
plt.show()
```

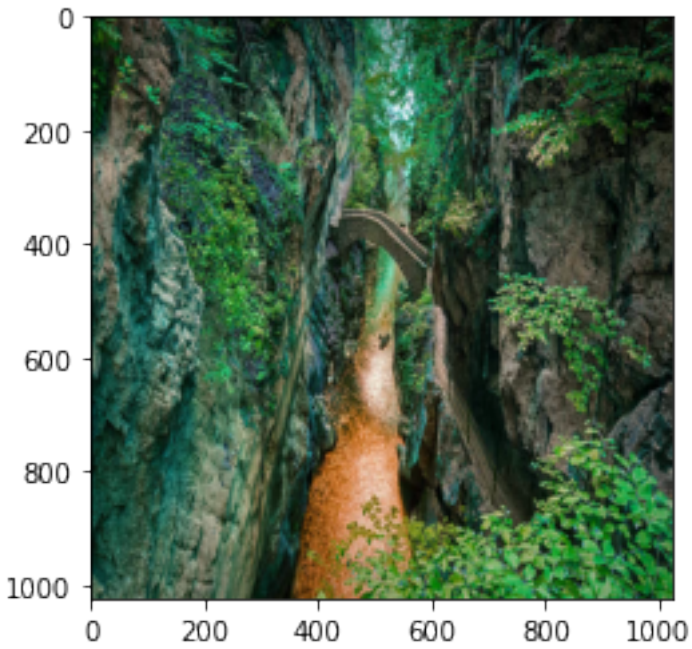


15 prediction

[7]:
```
model_path = "/content/drive/MyDrive/case study - II/tablenet/model checkpoint/
 ↪table_net.h5"
model.load_weights(model_path)
```

[16]:
```
image = cv2.imread('/content/9cdb1c60321e4636bc37363cdccc0e38.jpg', cv2.
 ↪IMREAD_UNCHANGED)
image = cv2.resize(image,(1024, 1024), cv2.INTER_NEAREST)
plt.imshow(image)
image = np.expand_dims(image, axis=0)
print(image.shape)
```

```
(1, 1024, 1024, 3)
```

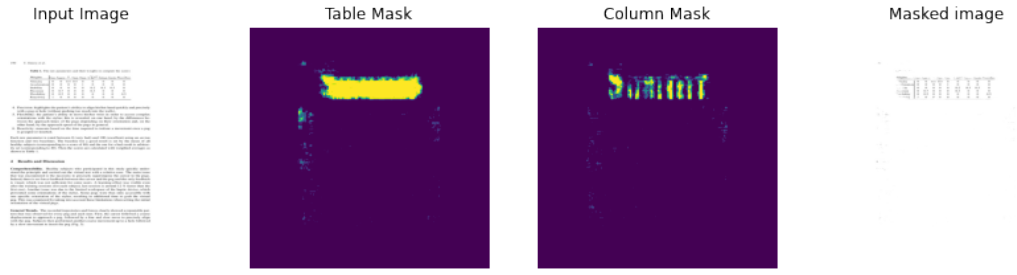


[17]:
```
table_mask_pred, col_mask_pred = model.predict(image)

table_mask_pred = tf.argmax(table_mask_pred, axis=-1)
table_mask_pred = table_mask_pred[..., tf.newaxis][0]

col_mask_pred = tf.argmax(col_mask_pred, axis=-1)
col_mask_pred = col_mask_pred[..., tf.newaxis][0]

im=tf.keras.preprocessing.image.array_to_img(image[0])
im.save('image.png')

im=tf.keras.preprocessing.image.array_to_img(table_mask_pred)
im.save('table_mask_pred.png')

im=tf.keras.preprocessing.image.array_to_img(col_mask_pred)
im.save('col_mask_pred.png')
```

[ ]:
```
count = 0

for image, mask in test_dataset.take(10):


    print(image.shape)

    table_mask_pred, col_mask_pred = model.predict(image)
```

```
    table_mask_pred = tf.argmax(table_mask_pred, axis=-1)
    table_mask_pred = table_mask_pred[..., tf.newaxis][0]

    col_mask_pred = tf.argmax(col_mask_pred, axis=-1)
    col_mask_pred = col_mask_pred[..., tf.newaxis][0]

    im=tf.keras.preprocessing.image.array_to_img(image[0])
    im.save('image.png')

    im=tf.keras.preprocessing.image.array_to_img(table_mask_pred)
    im.save('table_mask_pred.png')

    im=tf.keras.preprocessing.image.array_to_img(col_mask_pred)
    im.save('col_mask_pred.png')



    img_org = Image.open('./image.png')
    table_mask = Image.open('./table_mask_pred.png')
    col_mask = Image.open('./col_mask_pred.png')


    # convert images
    img_mask = table_mask.convert('L')
    # img_mask = col_mask.convert('L')

    # grayscale
    # add alpha channel
    img_org.putalpha(img_mask)

    # save as png which keeps alpha channel
    img_org.save('output.png')


    display([image[0], table_mask_pred, col_mask_pred, img_org])

    pytesseract.pytesseract.tesseract_cmd = r'/usr/bin/tesseract'
    text = pytesseract.image_to_string(Image.open('./output.png'), lang='eng' )␣
↪# config='--psm 11'
    print(text)
```

```
(2, 1024, 1024, 3)
```



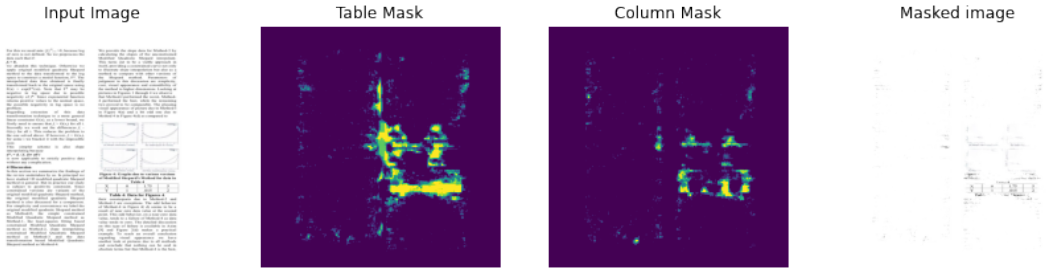
Seige eee Sapp Suerte Sar? Her

locity o" 0 os 05 0 0 0 © °
neration] 0 890 96 6 1 0 0 0 °

ay 0 0 0 0 0 02 08 os 0
xcossion | 0 0.7 0 0 0 038 0 0 o
camity | 0 05 0 0 0 © 0 © os

y |t @ @ © 0 © © 8 26

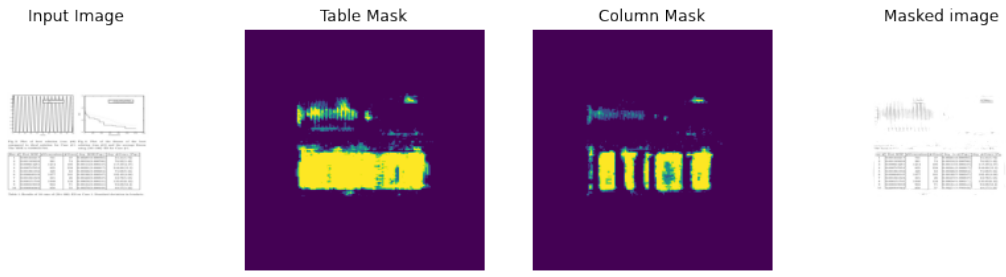
(2, 1024, 1024, 3)



slope

```
4
sagas 6
en Wied:

Tames 4
x 175
+ 015

Tah
* Nee Sar
```

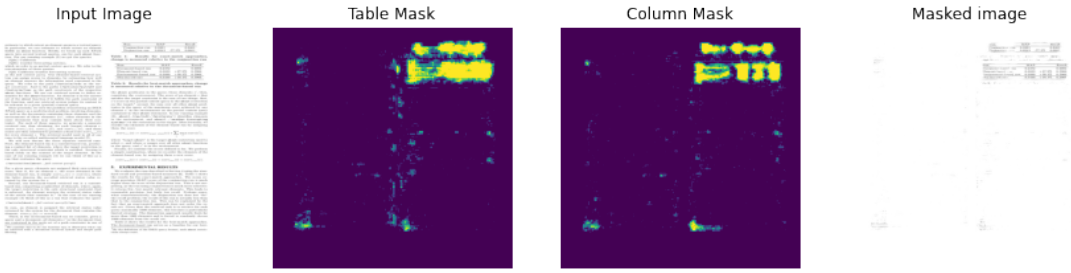
```
(2, 1024, 1024, 3)
```




```
aeeeeeenee
```

```
= =
croaysnreat nog
Phe Bass im HEE ah " i Sane
an W] Dent MSE [PGenarahon| Gene] Avs. WET Pop) [awa Benen POH
1 Jooortasai7] 73 "47 [o.00257(0.000765)] -45.31(5.72)
2 |ooorsse2s} x1 74 o.oos16(0.000780)]| -_76.92(3.42)
2 ooooersss:] 1214 105 Jo.oor14(0.000147)]-117.59(4.57)
4 Joooo7s7osa1 sas 234 fo.oozvico.q00s17)| --2.44.00(03.2)
5 Jooorseisse] 428 63 |o.00326(0.000684)
6 oooossior4s| 1077 101 |o-00186¢0.000347)|
7 |ooorsers23, 315 Jo.00440(0.000847
8 |oooorsizas} - 1040 124 Jo.00058°0.00018 }
° 933 Jo.00134,0.000343
re lnooesse1e2! 858 ln.oo27ine-ennata)
```

```
6 «

(2, 1024, 1024, 3)
```



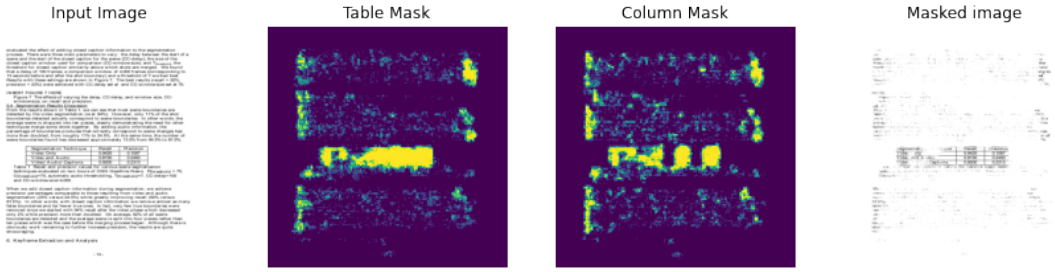
ie

Sonjunction smi 0.2467
Disiunets 7 0.0601,
wee a « ane
ia Recall
0.6004

Jocument Saeed am
Sllement-based re, 427.9% 0.5966

Taviroument-based rum 0.3090 430.3% 0.5966
66

(2, 1024, 1024, 3)



werag

techni ons
erase
nore

the

ef

i

al
wr
we
0 '
f
ii
pom
Recall Precision

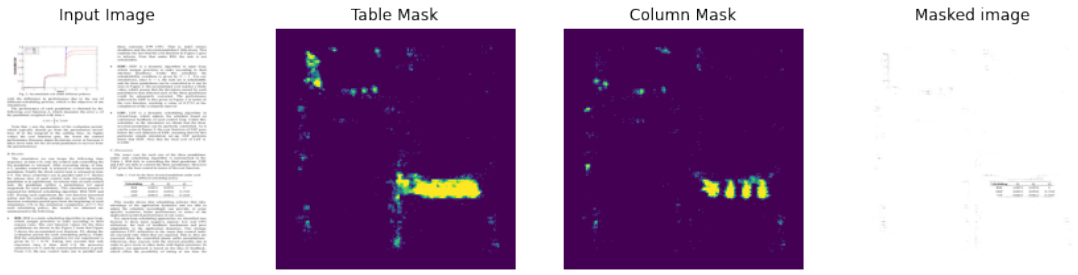
valk for variqus scene Sean

times ta

a

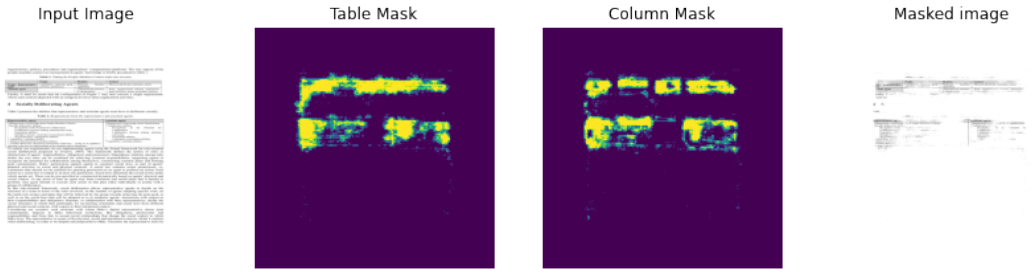
the

```
ing to
ost
= 2%

70,



we
het



eure. MAS
er of

we
ne
we SEES

(2, 1024, 1024, 3)
```




```
me 5 7 Le J:
RM 0.0033 0.0930 5
EDF 0.0033 0.0930 0.1769
LER 0.0033 0.0812 0.1645
```

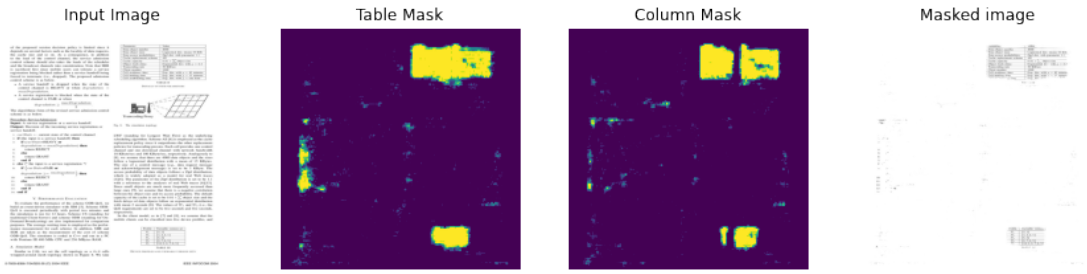
(2, 1024, 1024, 3)



here om

«
Tabe
+ ate mea te :
Aapreseniative musa: . ‘Assistant
Saree teens 2 SENSE TNamgenentott Ge eae
Moupieees : seme oe
‘cupatttiges care >. Bas Bae

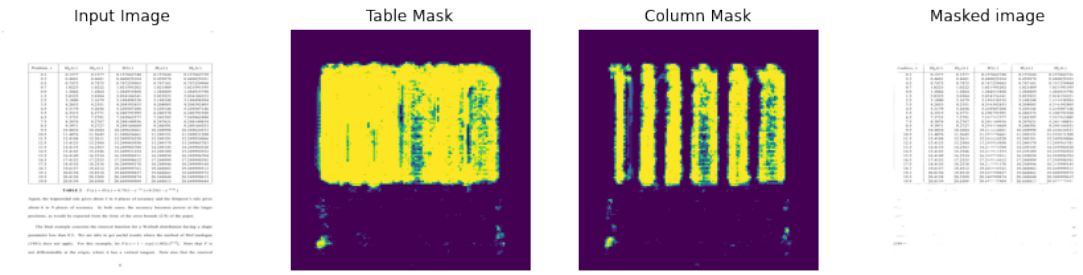
```
vrei damgnrete state

rome jew

(2, 1024, 1024, 3)
```




```
‘Data object sizes | _Lognowiia dit" (trea 1 Ke



AE
Cache capacity ‘0.01 = $" abject size
Object fetch delay Exposes dist wah p= 2.

"Transcoding rate [30 KBisce
Client numbe
Cell residence Gime

eit hoiding



ne
```

Frofile]_Viewabie vente i

(2, 1024, 1024, 3)



10.5

11s
: 12.5
: 13.5
14.5
15.5
16.5
17.5
18.7
19.1
19.5

(losis

3.1800
4.2653
5.3179
6.3515
7.3733
8.3876
9.3971
10.4034
11.4076
12.4104
13.4123

14.4135
15.4143
16.4149
17.4153
18.4155
19.6157
20.0158
20.4158
20.8159

3.1679
4.2351
5.2656
6.2771
7.2791
8.2767
9.2727

10.2684
11.2645
12.2611
13.2584
14.2563
15.2546
16.2534
17.2525
18.2518
19.4512
19.8510
20.2509
20.6508

0.460029104
0.747259965
1.021593202
1.284935808
2.024194341
3148498538

4.204392453

5.229507208

6.240791995

7.249 261577

§.241 140936

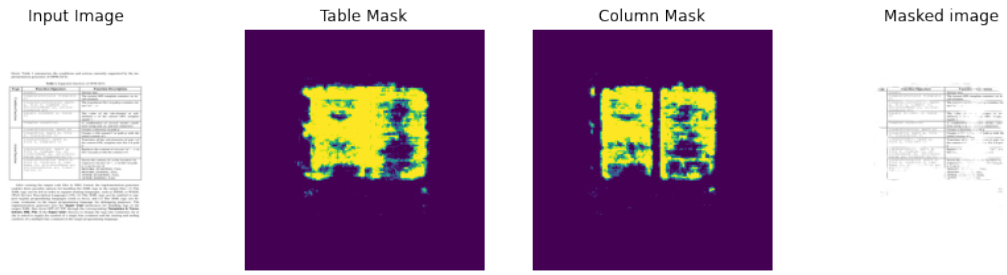
9.2491 4669
10.2428 24661
11.249624661
12.249924220
13.249965950
14.21924700
15.2493 1254
16.249885914
17.2499%8612
18.257990376
19.44909761
19.849999827
20.249999874

|
!
|
|
| 1.284809
| 2.023923
| 3.148248
| 4.204085
| 5.229146
| 6.240378
| 7.245395
8.247621
| 9.248591
| 10.248998
| 11249151

```
| 12.249191
| 13.249179
| 14.280145
| 15.249100
| 16.249050
| 17.248999
| 18.248946
| 19.448883
| 19848861
| 20.248840
| 20648813
```

```
. S
0.460029101
0.747259960
4021593195
1.284935798
2.024194323
3148498504
4.204392405
5.229507146
6.240791920
7249862488
8.248 140834
9.249164553
10.249624533
11.249831208
12.249924066
15.249968783
14.249984526
15.249992932
18.249996704
17.249998392
18.299999145
19.449999513
19.849999572
20.249999615
```

```
(2, 1024, 1024, 3)
```



Templatecontains (eemplace

sub-clement

attriputellame an,

breValue av)

Complex condition

Createbirectary,
GreateFile
translation ©)

Fanslsteda

SapiaceContais
Eaghame ny

Eributename any

value av,

attribece
Ezrunslation ©)

Tnsereatiocation (path pr

a

The alas
auriowte © nt

"The current OPL tamplaic contains tas i

The wenstatinn fie #" pSOh p comtalns <n

sat OPE wpe:

extra) alomie com

‘A combination =
tions using aid.

```
the cument 12



respect to
p.lean br
REPOR?
EPORF
AFTER 14
ARTE



"Translates aff

Taverte the brinicitt oi" 42 1h ocadon i
```

[ ]: