Kidney Stone Detection from Abdominal Ultrasound Images

Csv file:

	image_id	path	diag	target	Class
0	Tumor-(1044)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1044).jpg	Tumor	3	Tumor
1	Tumor-(83)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (83).jpg	Tumor	3	Tumor
2	Tumor-(580)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (580).jpg	Tumor	3	Tumor
3	Tumor-(1701)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1701).jpg	Tumor	3	Tumor
4	Tumor-(1220)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1220).jpg	Tumor	3	Tumor
5	Tumor-(249)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (249).jpg	Tumor	3	Tumor
6	Tumor-(356)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (356).jpg	Tumor	3	Tumor
7	Tumor-(52)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (52).jpg	Tumor	3	Tumor
8	Tumor-(501)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (501).jpg	Tumor	3	Tumor
9	Tumor-(948)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (948).jpg	Tumor	3	Tumor
10	Tumor-(1724)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1724).jpg	Tumor	3	Tumor
11	Tumor-(2077)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (2077).jpg	Tumor	3	Tumor
12	Tumor-(2240)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (2240).jpg	Tumor	3	Tumor
13	Tumor-(389)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (389).jpg	Tumor	3	Tumor
14	Tumor-(1616)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1616).jpg	Tumor	3	Tumor
15	Tumor-(911)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (911).jpg	Tumor	3	Tumor
16	Tumor-(1178)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1178).jpg	Tumor	3	Tumor
17	Tumor-(186)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (186).jpg	Tumor	3	Tumor
18	Tumor-(1599)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1599).jpg	Tumor	3	Tumor

19	Tumor-(496)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (496).jpg	Tumor	3	Tumor
20	Tumor-(193)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (193).jpg	Tumor	3	Tumor
21	Tumor-(1305)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1305).jpg	Tumor	3	Tumor
22	Tumor-(1189)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1189).jpg	Tumor	3	Tumor
23	Tumor-(2151)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (2151).jpg	Tumor	3	Tumor
24	Tumor-(637)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (637).jpg	Tumor	3	Tumor
25	Tumor-(1232)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1232).jpg	Tumor	3	Tumor
26	Tumor-(22)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (22).jpg	Tumor	3	Tumor
27	Tumor-(1266)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1266).jpg	Tumor	3	Tumor
28	Tumor-(602)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (602).jpg	Tumor	3	Tumor
29	Tumor-(1440)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1440).jpg	Tumor	3	Tumor
30	Tumor-(2214)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (2214).jpg	Tumor	3	Tumor
31	Tumor-(1418)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1418).jpg	Tumor	3	Tumor
32	Tumor-(2130)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (2130).jpg	Tumor	3	Tumor
33	Tumor-(237)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (237).jpg	Tumor	3	Tumor
34	Tumor-(358)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (358).jpg	Tumor	3	Tumor
35	Tumor-(1668)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1668).jpg	Tumor	3	Tumor
36	Tumor-(235)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (235).jpg	Tumor	3	Tumor
37	Tumor-(633)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (633).jpg	Tumor	3	Tumor
38	Tumor-(550)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (550).jpg	Tumor	3	Tumor
39	Tumor-(960)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (960).jpg	Tumor	3	Tumor
40	Tumor-(698)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (698).jpg	Tumor	3	Tumor
41	Tumor-(176)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (176).jpg	Tumor	3	Tumor
42	Tumor-(359)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (359).jpg	Tumor	3	Tumor
43	Tumor-(882)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (882).jpg	Tumor	3	Tumor

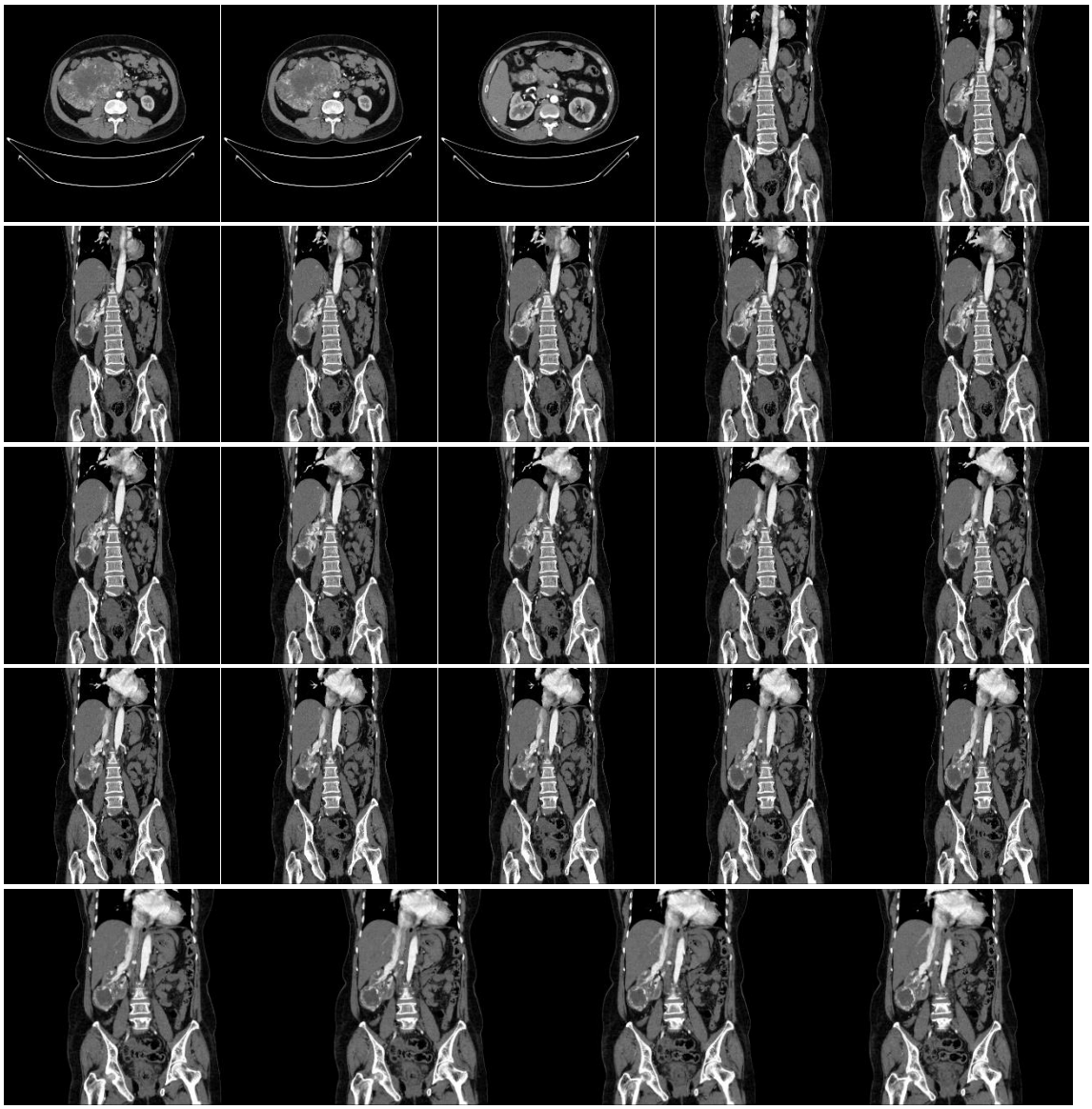
44	Tumor-(982)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (982).jpg	Tumor	3	Tumor
45	Tumor-(1588)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1588).jpg	Tumor	3	Tumor
46	Tumor-(735)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (735).jpg	Tumor	3	Tumor
47	Tumor-(1970)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1970).jpg	Tumor	3	Tumor
48	Tumor-(1787)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1787).jpg	Tumor	3	Tumor
49	Tumor-(2159)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (2159).jpg	Tumor	3	Tumor
50	Tumor-(1469)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1469).jpg	Tumor	3	Tumor
51	Tumor-(273)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (273).jpg	Tumor	3	Tumor
52	Tumor-(96)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (96).jpg	Tumor	3	Tumor
53	Tumor-(1344)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1344).jpg	Tumor	3	Tumor
54	Tumor-(497)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (497).jpg	Tumor	3	Tumor
55	Tumor-(996)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (996).jpg	Tumor	3	Tumor
56	Tumor-(1913)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1913).jpg	Tumor	3	Tumor
57	Tumor-(1912)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1912).jpg	Tumor	3	Tumor
58	Tumor-(1496)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1496).jpg	Tumor	3	Tumor
59	Tumor-(1069)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1069).jpg	Tumor	3	Tumor
60	Tumor-(1631)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1631).jpg	Tumor	3	Tumor
61	Tumor-(940)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (940).jpg	Tumor	3	Tumor
62	Tumor-(1793)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1793).jpg	Tumor	3	Tumor
63	Tumor-(84)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (84).jpg	Tumor	3	Tumor
64	Tumor-(1914)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1914).jpg	Tumor	3	Tumor
65	Tumor-(2119)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (2119).jpg	Tumor	3	Tumor
66	Tumor-(139)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (139).jpg	Tumor	3	Tumor
67	Tumor-(455)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (455).jpg	Tumor	3	Tumor
68	Tumor-(1071)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1071).jpg	Tumor	3	Tumor

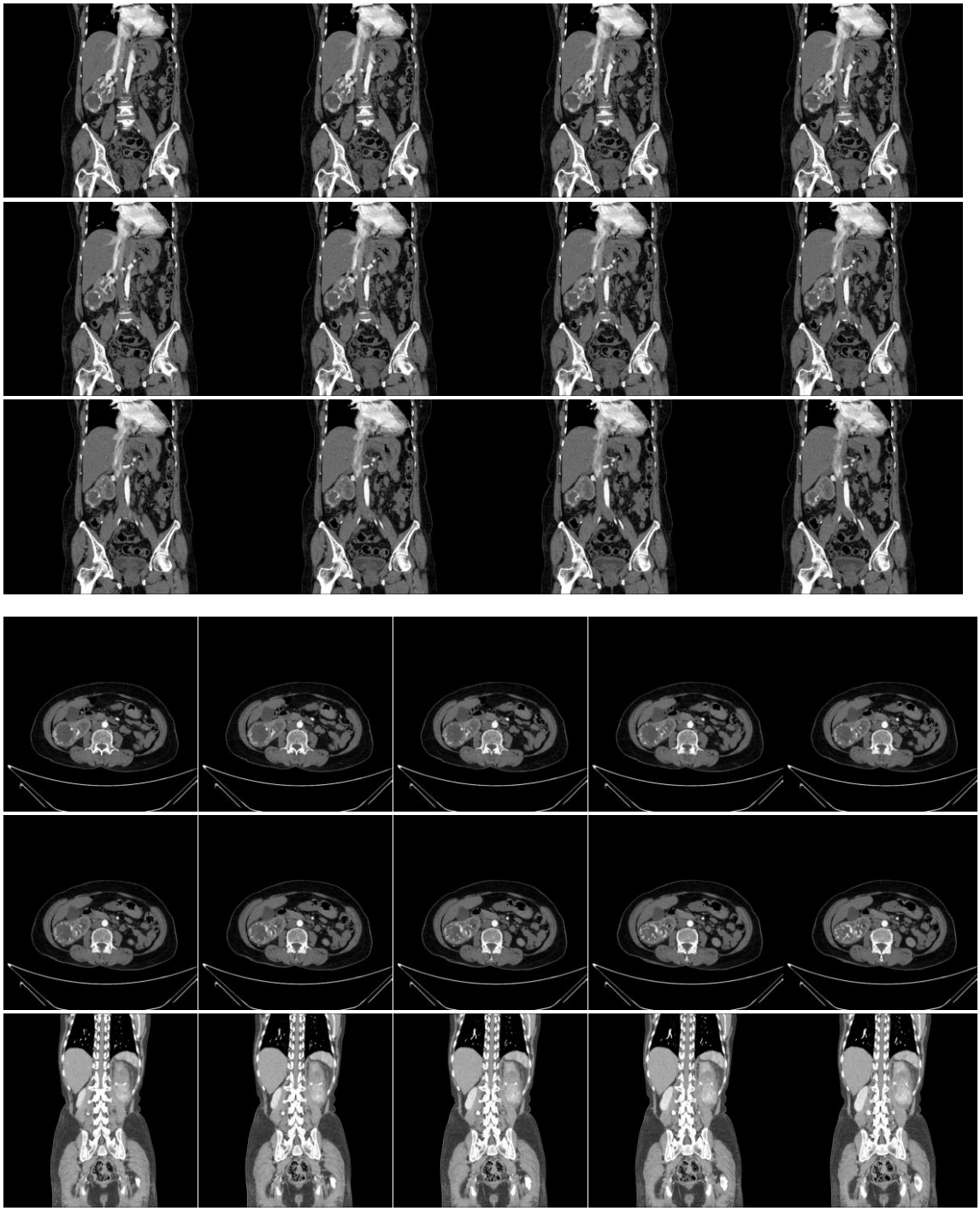
69	Tumor-(428)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (428).jpg	Tumor	3	Tumor
70	Tumor-(708)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (708).jpg	Tumor	3	Tumor
71	Tumor-(2018)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (2018).jpg	Tumor	3	Tumor
72	Tumor-(924)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (924).jpg	Tumor	3	Tumor
73	Tumor-(121)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (121).jpg	Tumor	3	Tumor
74	Tumor-(1375)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1375).jpg	Tumor	3	Tumor
75	Tumor-(2210)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (2210).jpg	Tumor	3	Tumor
76	Tumor-(1213)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1213).jpg	Tumor	3	Tumor
77	Tumor-(656)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (656).jpg	Tumor	3	Tumor
78	Tumor-(1843)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1843).jpg	Tumor	3	Tumor
79	Tumor-(774)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (774).jpg	Tumor	3	Tumor
80	Tumor-(1477)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1477).jpg	Tumor	3	Tumor
81	Tumor-(1868)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1868).jpg	Tumor	3	Tumor
82	Tumor-(1405)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1405).jpg	Tumor	3	Tumor
83	Tumor-(481)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (481).jpg	Tumor	3	Tumor
84	Tumor-(1875)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1875).jpg	Tumor	3	Tumor
85	Tumor-(1311)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1311).jpg	Tumor	3	Tumor
86	Tumor-(1137)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1137).jpg	Tumor	3	Tumor
87	Tumor-(109)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (109).jpg	Tumor	3	Tumor
88	Tumor-(978)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (978).jpg	Tumor	3	Tumor
89	Tumor-(1934)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1934).jpg	Tumor	3	Tumor
90	Tumor-(294)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (294).jpg	Tumor	3	Tumor
91	Tumor-(1584)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1584).jpg	Tumor	3	Tumor
92	Tumor-(405)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (405).jpg	Tumor	3	Tumor
93	Tumor-(1878)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1878).jpg	Tumor	3	Tumor

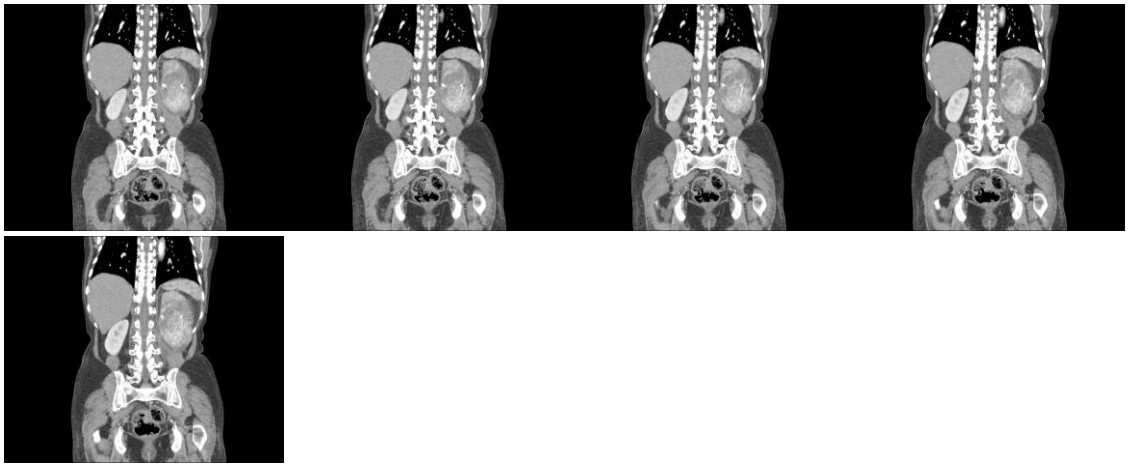
94	Tumor-(1249)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1249).jpg	Tumor	3	Tumor
95	Tumor-(38)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (38).jpg	Tumor	3	Tumor
96	Tumor-(74)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (74).jpg	Tumor	3	Tumor
97	Tumor-(931)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (931).jpg	Tumor	3	Tumor
98	Tumor-(1463)	/content/data/CT KIDNEY DATASET Normal, CYST, TUMOR and STONE/TUMOR/Tumor- (1463).jpg	Tumor	3	Tumor

Image datasets:

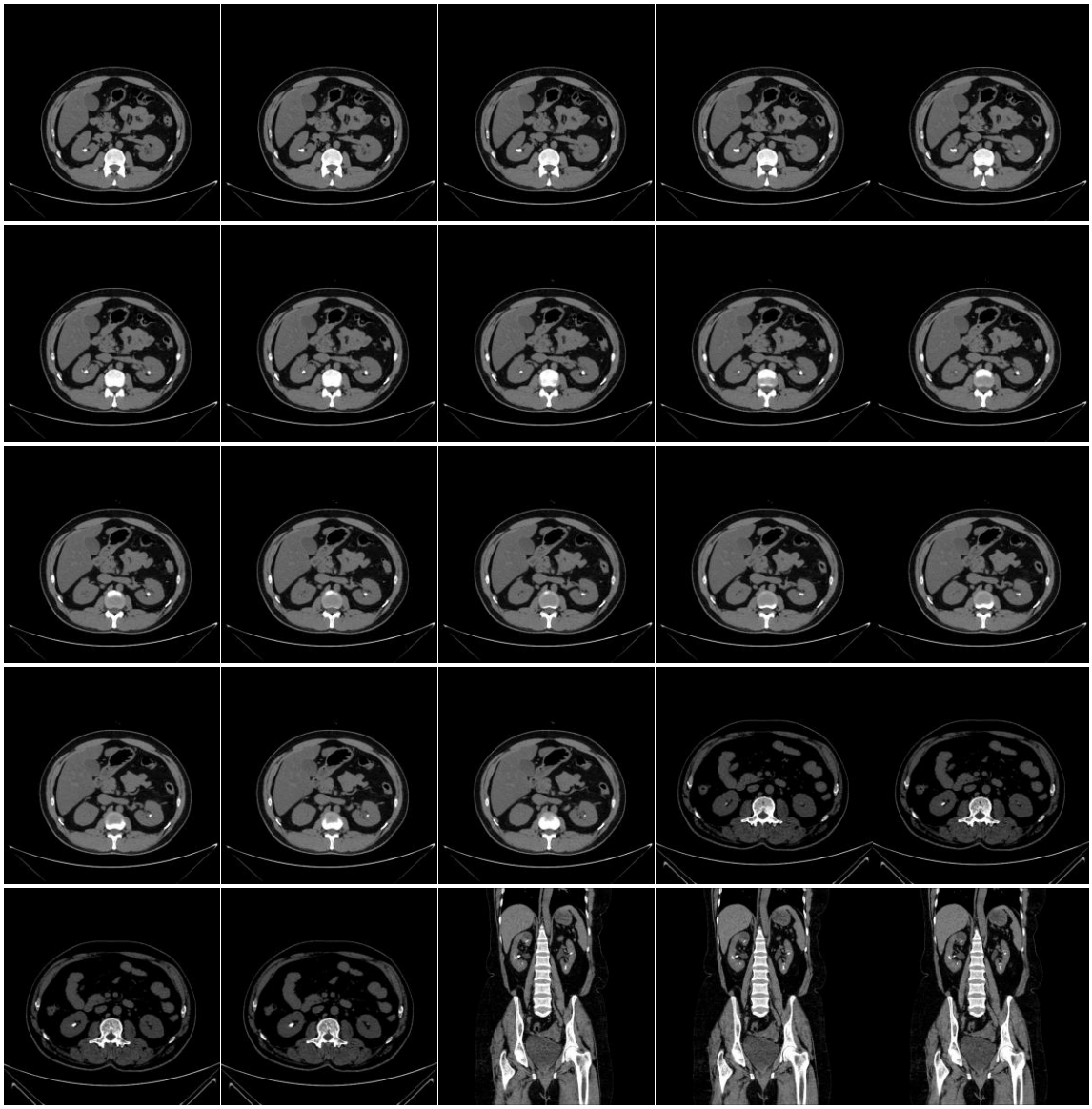
Tumor:

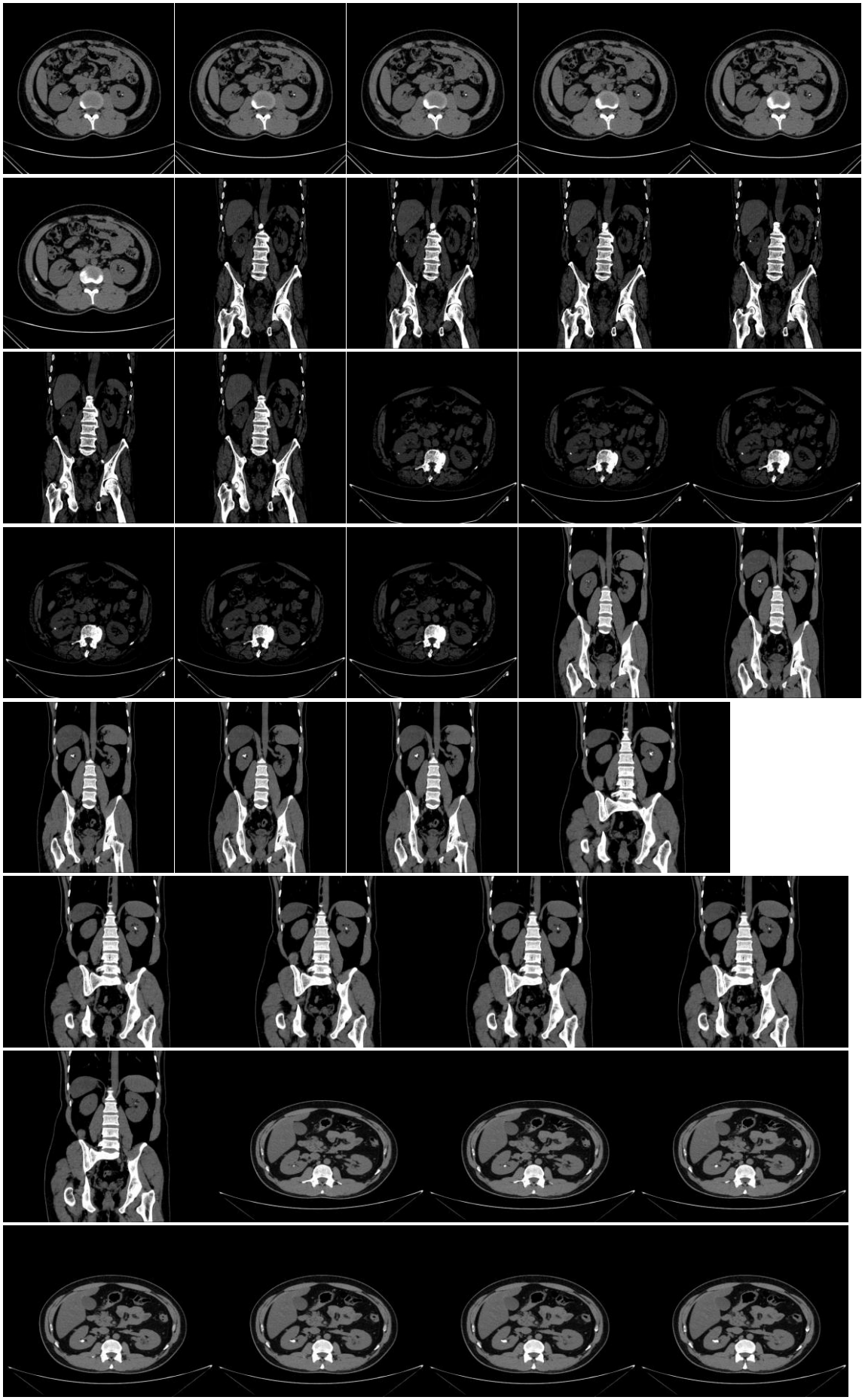




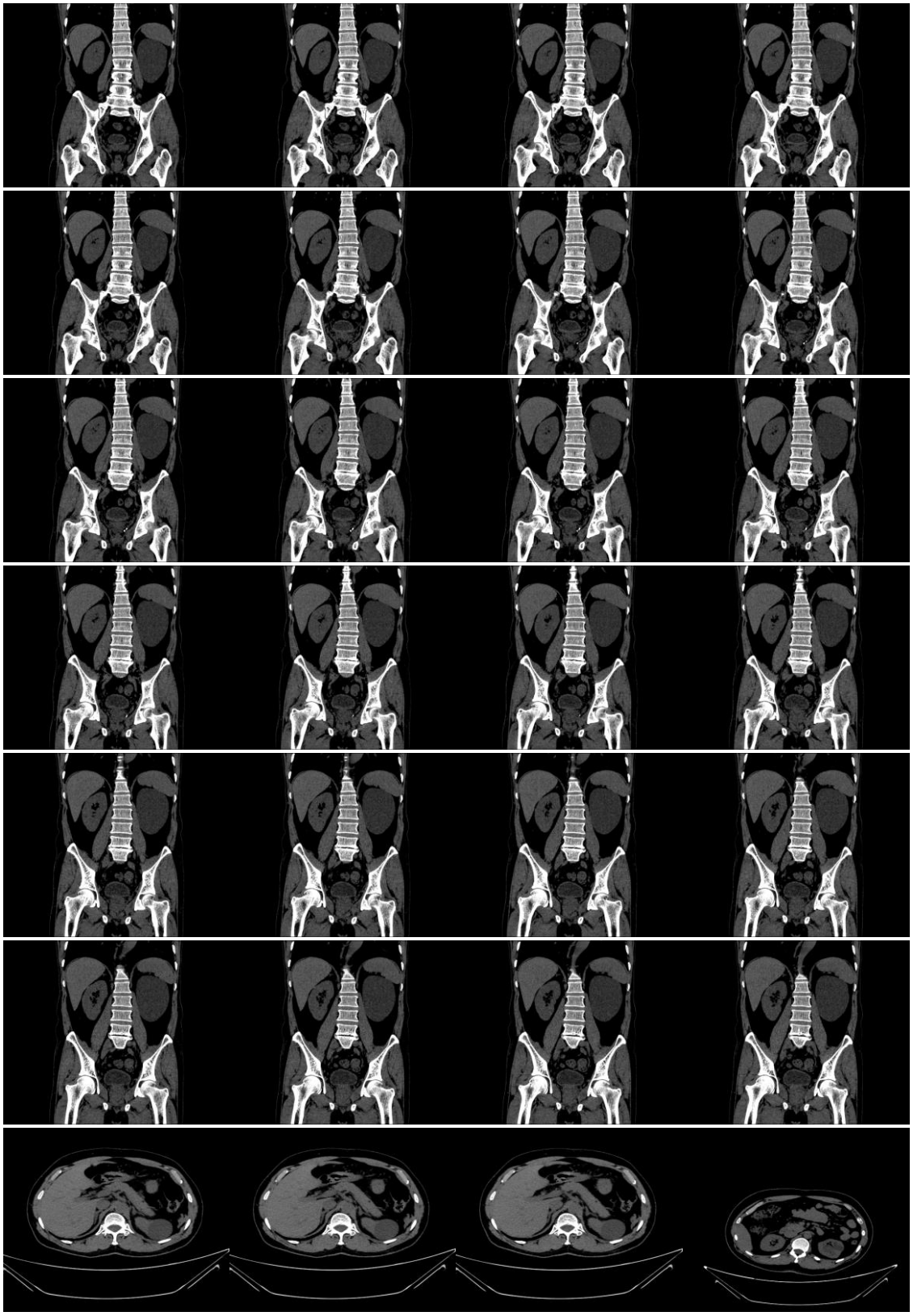


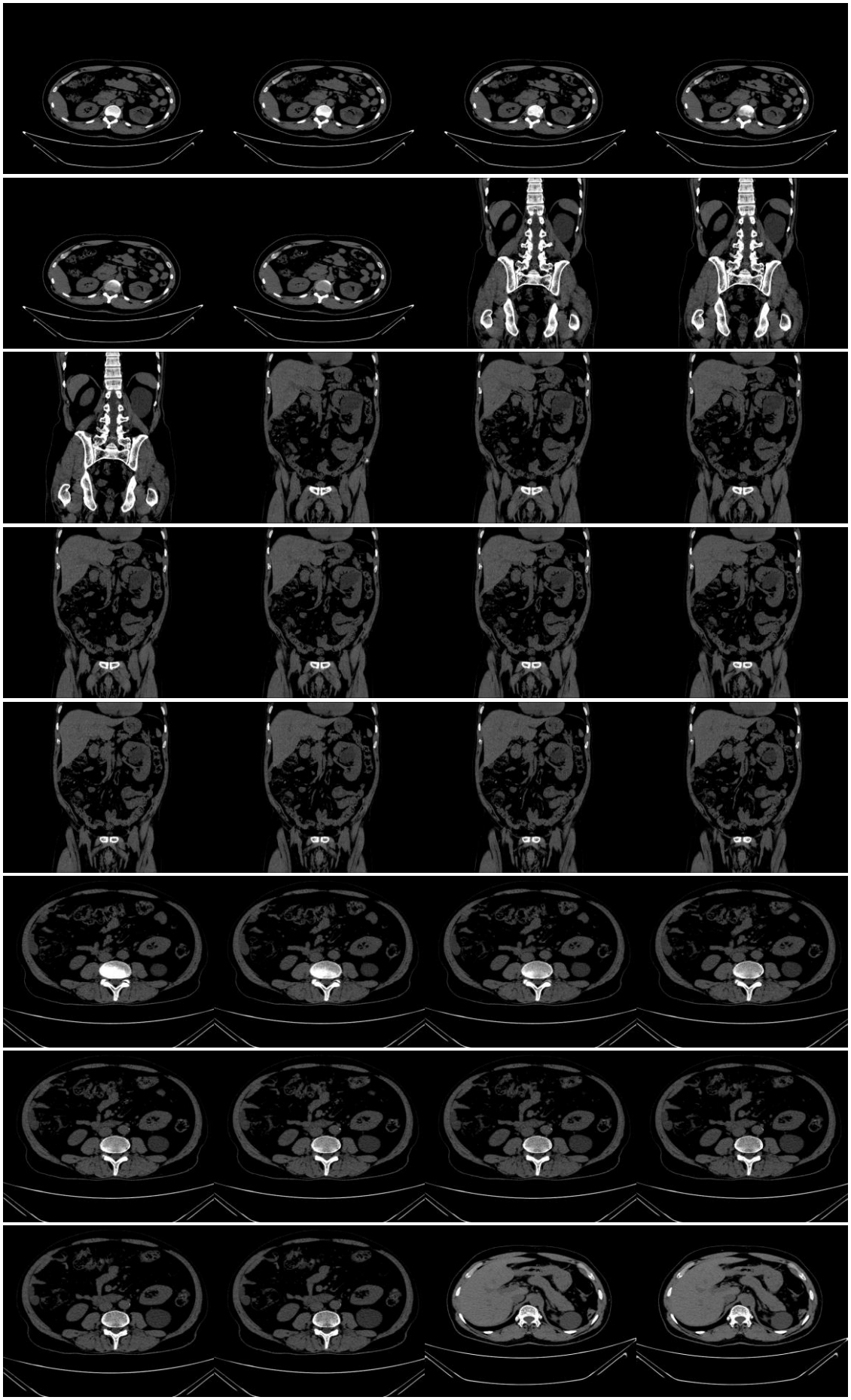
Stone:

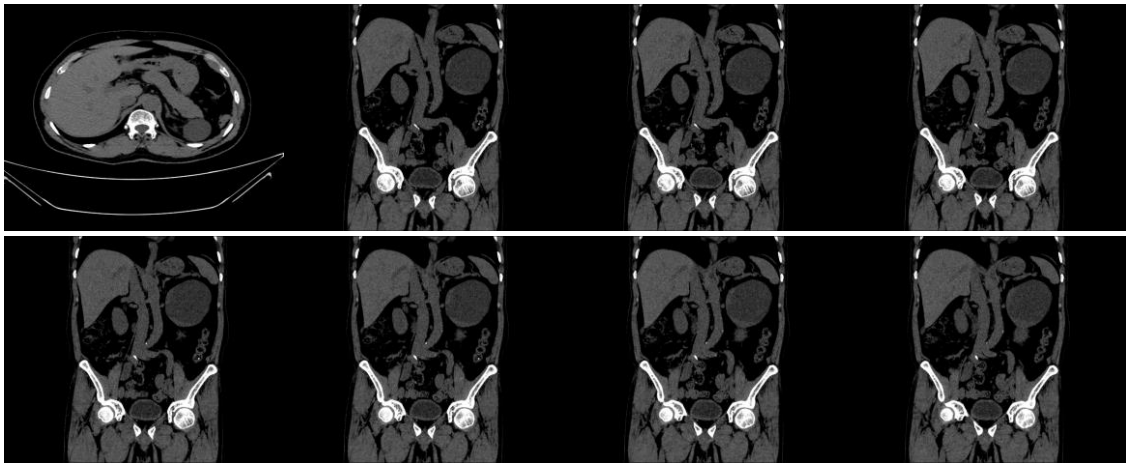




Cyst:







PYTHON CODE IMPLEMENTATION:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plot
data=pd.read_csv('/content/kidney-stone-dataset.csv')
data.shape
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 90 entries, 0 to 89
Data columns (total 8 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   Unnamed: 0    90 non-null     int64  
 1   gravity       90 non-null     float64
 2   ph            90 non-null     float64
 3   osmo          90 non-null     int64  
 4   cond          90 non-null     float64
 5   urea          90 non-null     int64  
 6   calc          90 non-null     float64
 7   target        90 non-null     int64  
dtypes: float64(4), int64(4)
memory usage: 5.8 KB
```

```
data.columns
```

```
Index(['Unnamed: 0', 'gravity', 'ph', 'osmo', 'cond', 'urea', 'calc',
       'target'],
      dtype='object')
```

```
data.isnull()
```

	Unnamed: 0	gravity	ph	osmo	cond	urea	calc	target
0	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...
85	False	False	False	False	False	False	False	False
86	False	False	False	False	False	False	False	False
87	False	False	False	False	False	False	False	False
88	False	False	False	False	False	False	False	False
89	False	False	False	False	False	False	False	False

90 rows × 8 columns

```
data.isnull().sum()
```

	0
Unnamed: 0	0
gravity	0
ph	0
osmo	0
cond	0
urea	0
calc	0
target	0

dtype: int64

```
data.dropna()
```

	Unnamed: 0	gravity	ph	osmo	cond	urea	calc	target
0	0	1.021000	4.910000	725	14.000000	443	2.450000	0
1	1	1.017000	5.740000	577	20.000000	296	4.490000	0
2	2	1.008000	7.200000	321	14.900000	101	2.360000	0
3	3	1.011000	5.510000	408	12.600000	224	2.150000	0
4	4	1.005000	6.520000	187	7.500000	91	1.160000	0
...	...	...	...	...	...	...	...	...
85	85	1.021452	5.556081	756	24.241481	367	7.669120	1
86	86	1.016501	6.900257	549	20.549790	204	5.775256	1
87	87	1.032754	5.443491	1085	23.188653	576	8.664169	1
88	88	1.023870	5.106433	325	12.124689	50	0.781620	1
89	89	1.013723	6.308943	472	16.907792	174	2.556405	1

90 rows × 8 columns

```
data.dropna(how='any')
```

	Unnamed: 0	gravity	ph	osmo	cond	urea	calc	target
0	0	1.021000	4.910000	725	14.000000	443	2.450000	0
1	1	1.017000	5.740000	577	20.000000	296	4.490000	0
2	2	1.008000	7.200000	321	14.900000	101	2.360000	0
3	3	1.011000	5.510000	408	12.600000	224	2.150000	0
4	4	1.005000	6.520000	187	7.500000	91	1.160000	0
...	...	...	...	...	...	...	...	...
85	85	1.021452	5.556081	756	24.241481	367	7.669120	1
86	86	1.016501	6.900257	549	20.549790	204	5.775256	1
87	87	1.032754	5.443491	1085	23.188653	576	8.664169	1
88	88	1.023870	5.106433	325	12.124689	50	0.781620	1
89	89	1.013723	6.308943	472	16.907792	174	2.556405	1

90 rows × 8 columns

```
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.metrics import confusion_matrix
from matplotlib.colors import ListedColormap
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
data = data.dropna()
x = data.iloc[:, [2, 3]].values
y = data.iloc[:, 4].values
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y,
test_size=0.25, random_state=0)
print(f"Training Set (x_train):\n{x_train}")
print(f"Test Set (x_test):\n{x_test}")
```

```
Training Set (x_train):
[[   5.53        775.        ]
 [   5.56394009  377.        ]
 [   6.13        364.        ]
 [   4.81        410.        ]
 [   5.93249307  538.        ]
 [   6.74178795  427.        ]
 [   5.51        408.        ]
 [   5.98        487.        ]
 [   6.81        594.        ]
 [   6.61        225.        ]
 [   6.88        395.        ]
 [   5.86        531.        ]
 [   6.27        371.        ]
 [   5.4         803.        ]
 [   6.19        956.        ]
 [   5.73        874.        ]
 [   6.52        187.        ]
 [   4.9         684.        ]
 [   6.81        947.        ]
 [   5.33        815.        ]
 [   6.37        325.        ]
 [   5.27        668.        ]
 [   7.00457192  443.        ]
 [   4.76        312.        ]
 [   4.91        725.        ]
 [   5.85        970.        ]
 [   5.87        241.        ]
 [   5.53        907.        ]
 [   5.66        702.        ]
```

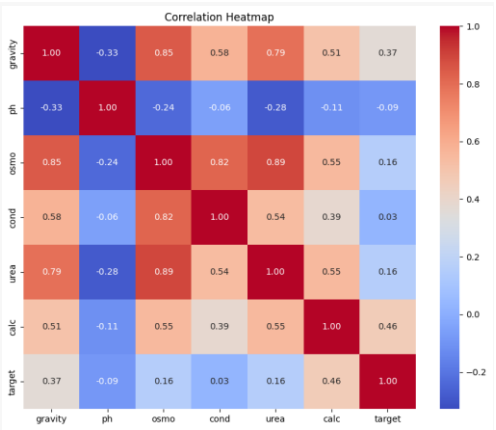
```
Test Set (x_test):
[[   7.2         321.        ]
 [   5.35        283.        ]
 [   6.21        442.        ]
 [   5.64        386.        ]
 [   6.9         945.        ]
 [   7.92        680.        ]
 [   5.94        774.        ]
 [   5.97        343.        ]
 [   5.4         831.        ]
 [   6.30894329  472.        ]
 [   5.67       1107.        ]
 [   7.61        527.        ]
 [   5.94        256.        ]
 [   5.58        516.        ]
 [   5.41        543.        ]
 [   6.63        253.        ]
 [   5.68        749.        ]
 [   7.94        567.        ]
 [   5.68419966  283.        ]
 [   6.79        541.        ]
 [   7.38        577.        ]
 [   5.62        461.        ]
 [   5.58       1032.        ]]
```

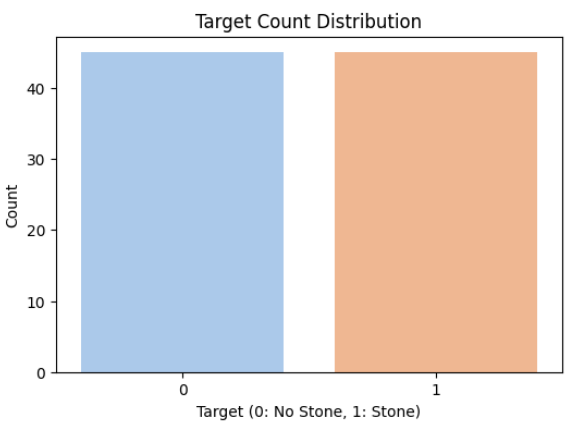
```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
data=pd.read_csv('/content/kidney-stone-dataset.csv')
print(data.info())
print(data.columns)
data = data.dropna()
numerical_cols = data.select_dtypes(include=np.number).columns
print(f"Numerical Columns: {numerical_cols}")
x = data.loc[:,numerical_cols].values
y = data.iloc[:, 4].values
x_train, x_test, y_train, y_test = train_test_split(x, y,
test_size=0.25, random_state=0)
print(f"Training Set (x_train):\n{x_train}")
print(f"Test Set (x_test):\n{x_test}")
st_x = StandardScaler()
x_train = st_x.fit_transform(x_train)
x_test = st_x.transform(x_test)
```

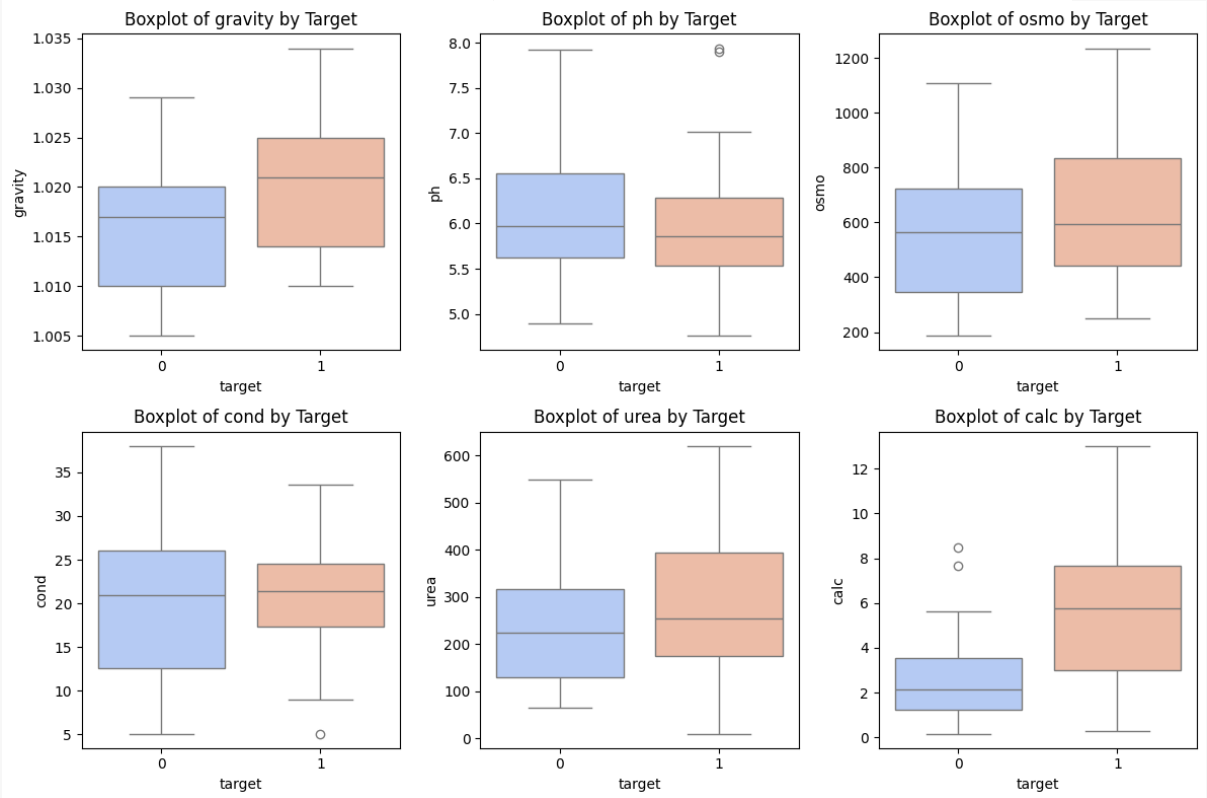
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 90 entries, 0 to 89
Data columns (total 8 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   Unnamed: 0  90 non-null     int64
 1   gravity     90 non-null     float64
 2   ph          90 non-null     float64
 3   osmo        90 non-null     int64
 4   cond        90 non-null     float64
 5   urea        90 non-null     int64
 6   calc        90 non-null     float64
 7   target      90 non-null     int64
dtypes: float64(4), int64(4)
memory usage: 5.8 KB
None
Index(['Unnamed: 0', 'gravity', 'ph', 'osmo', 'cond', 'urea', 'calc',
       'target'],
      dtype='object')
Numerical Columns: Index(['Unnamed: 0', 'gravity', 'ph', 'osmo', 'cond', 'urea', 'calc',
       'target'],
      dtype='object')
Training Set (x_train):
[[4.80000000e+01 1.02100000e+00 5.53000000e+00 7.75000000e+02
  3.12000000e+01 3.02000000e+02 6.19000000e+00 1.00000000e+00]
 [8.00000000e+01 1.01977069e+00 5.56394009e+00 3.77000000e+02
  1.26532095e+01 1.16000000e+02 6.04317608e+00 1.00000000e+00]
 [5.40000000e+01 1.01100000e+00 6.13000000e+00 3.64000000e+02
  1.09000000e+01 1.59000000e+02 3.10000000e+00 1.00000000e+00]
 [7.30000000e+01 1.01700000e+00 4.81000000e+00 4.10000000e+02
  1.33000000e+01 1.95000000e+02 5.80000000e-01 1.00000000e+00]
 [7.90000000e+01 1.01385974e+00 5.93249307e+00 5.38000000e+02
```

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
```

```
from sklearn.svm import SVC
from sklearn.metrics import confusion_matrix
from sklearn.linear_model import LinearRegression
data_set = pd.read_csv('/content/kidney-stone-dataset.csv')
data_set = data_set.dropna()
x = data_set.iloc[:, [2, 3]].values
y = data_set.iloc[:, 4].values
x_train, x_test, y_train, y_test = train_test_split(x, y,
test_size=0.25, random_state=0)
print(f"Training Set (x_train):\n{x_train}")
print(f"Test Set (x_test):\n{x_test}")
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
file_path = "//content/kidney-stone-dataset.csv"
data = pd.read_csv(file_path)
data = data.drop(columns=['Unnamed: 0'])
numeric_cols = ['gravity', 'ph', 'osmo', 'cond', 'urea', 'calc',
'target']
data[numeric_cols] = data[numeric_cols].apply(pd.to_numeric,
errors='coerce')
data = data.dropna(subset=numeric_cols)
plt.figure(figsize=(10, 8))
sns.heatmap(data.corr(), annot=True, cmap='coolwarm', fmt='.2f')
plt.title("Correlation Heatmap")
plt.show()
plt.figure(figsize=(6, 4))
sns.countplot(x='target', data=data, palette='pastel')
plt.title("Target Count Distribution")
plt.xlabel("Target (0: No Stone, 1: Stone)")
plt.ylabel("Count")
plt.show()
feature_cols = ['gravity', 'ph', 'osmo', 'cond', 'urea', 'calc']
plt.figure(figsize=(12, 8))
for i, col in enumerate(feature_cols):
    plt.subplot(2, 3, i+1)
    sns.boxplot(x='target', y=col, data=data, palette='coolwarm')
    plt.title(f"Boxplot of {col} by Target")
plt.tight_layout()
plt.show()
```

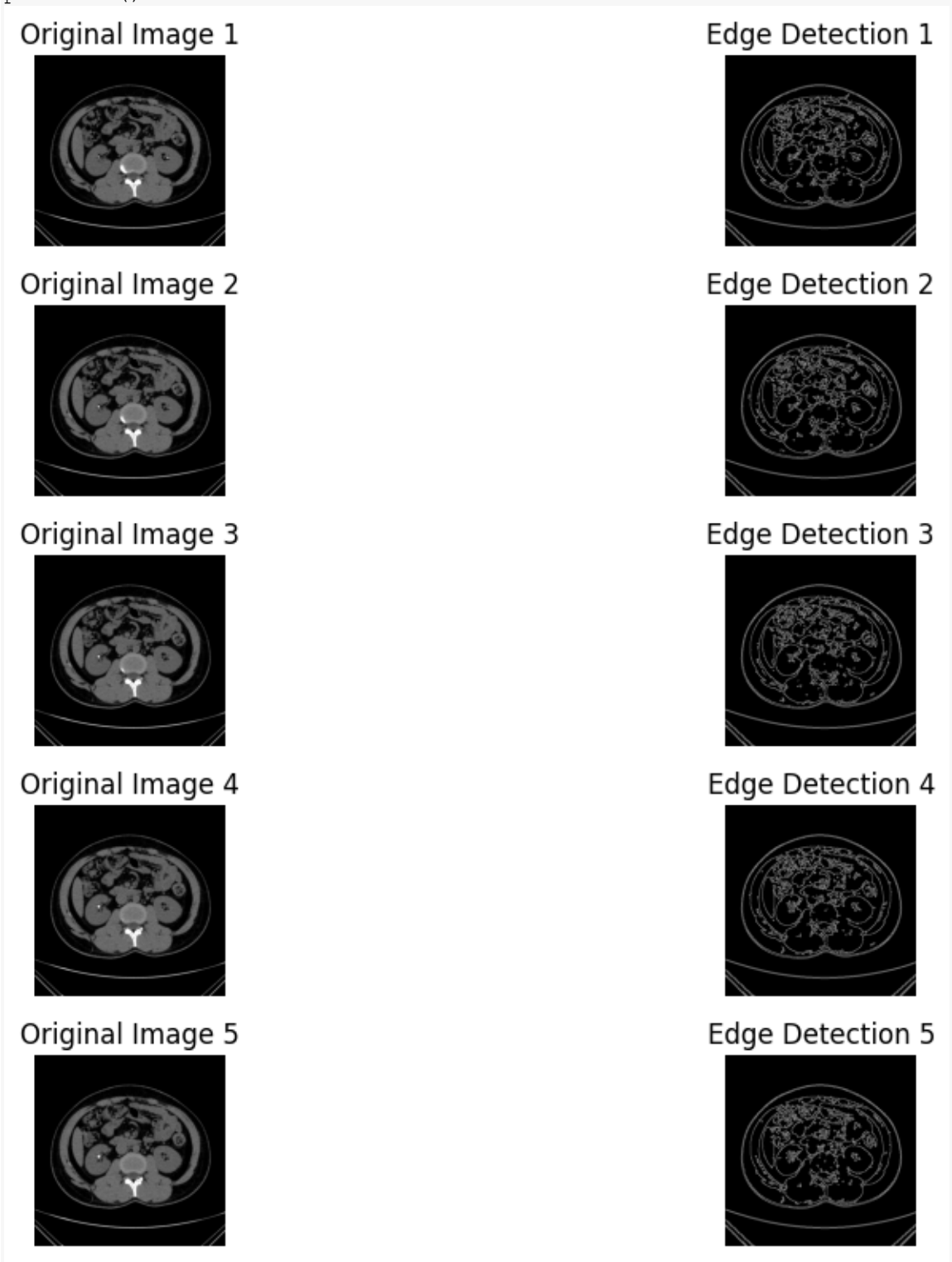







```
import cv2
import matplotlib.pyplot as plt
file_paths = [
    "/content/Stone- (994).jpg",
    "/content/Stone- (995).jpg",
    "/content/Stone- (996).jpg",
    "/content/Stone- (997).jpg",
    "/content/Stone- (998).jpg"
]
plt.figure(figsize=(12, 8))
for i, path in enumerate(file_paths):
    img = cv2.imread(path, cv2.IMREAD_GRAYSCALE)
    edges = cv2.Canny(img, 100, 200)
    plt.subplot(5, 2, 2*i+1)
    plt.imshow(img, cmap='gray')
    plt.title(f"Original Image {i+1}")
```

```
    plt.axis('off')
    plt.subplot(5, 2, 2*i+2)
    plt.imshow(edges, cmap='gray')
    plt.title(f"Edge Detection {i+1}")
    plt.axis('off')
plt.tight_layout()
plt.show()
```



```
import pandas as pd
import numpy as np
from sklearn.metrics import multilabel_confusion_matrix
csv_file_path = '//content/kidney-stone-dataset.csv'
df = pd.read_csv(csv_file_path)
df['target'] = df['target'].apply(lambda x: eval(x) if isinstance(x,
str) else x)
y_true = np.array(df["target"].tolist())
np.random.seed(42)
y_pred = np.random.randint(0, 2, size=y_true.shape)
conf_matrices = multilabel_confusion_matrix(y_true, y_pred)
label_names = df['target'].apply(pd.Series).columns
conf_matrices_dict = {label: conf_matrices[i] for i, label in
enumerate(label_names)}
for label, matrix in conf_matrices_dict.items():
    print(f"Confusion Matrix for {label}:")
    print(matrix)
    print()
```

```
Confusion Matrix for 0:
[[24 21]
 [23 22]]
```

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
data = pd.read_csv('/content/kidney-stone-dataset.csv')
X = data[['ph', 'osmo']]
y = data['target']
X = pd.get_dummies(X, drop_first=True)
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.3, random_state=42)
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
print(f'Accuracy: {accuracy}')
```

```
Accuracy: 0.5555555555555556
```

```
from sklearn.metrics import precision_score
y_pred = model.predict(X_test)
precision = precision_score(y_test, y_pred, average='weighted')
print(f'Precision: {precision}')
```

```
Precision: 0.5925925925925926
```

```
from sklearn.metrics import recall_score
y_pred = model.predict(X_test)
recall = recall_score(y_test, y_pred, average='weighted')
```

```
print(f'Recall: {recall}')
```

```
Recall: 0.5555555555555556
```

```
from sklearn.metrics import f1_score
y_pred = model.predict(X_test)
f1 = f1_score(y_test, y_pred, average='weighted')
print(f'F1 Score: {f1}')
```

```
F1 Score: 0.5444444444444444
```