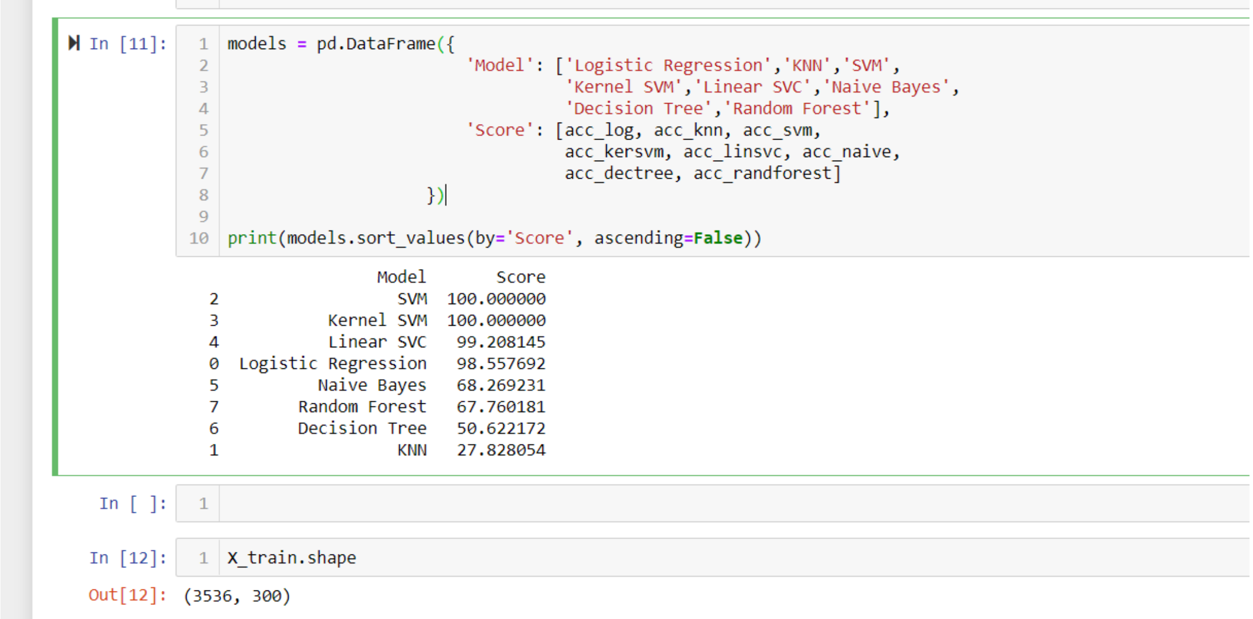
NewData

8 times augmented

Failed to predict - 51



In [11]:
```
models = pd.DataFrame({
                        'Model': ['Logistic Regression','KNN','SVM',
                                  'Kernel SVM','Linear SVC','Naive Bayes',
                                  'Decision Tree','Random Forest'],
                        'Score': [acc_log, acc_knn, acc_svm,
                                  acc_kersvm, acc_linsvc, acc_naive,
                                  acc_dectree, acc_randforest]
                    })

print(models.sort_values(by='Score', ascending=False))
```

```
                 Model       Score
2                  SVM  100.000000
3           Kernel SVM  100.000000
4           Linear SVC   99.208145
0  Logistic Regression   98.557692
5          Naive Bayes   68.269231
7        Random Forest   67.760181
6        Decision Tree   50.622172
1                  KNN   27.828054
```

In [ ]:

In [12]:
```
X_train.shape
```

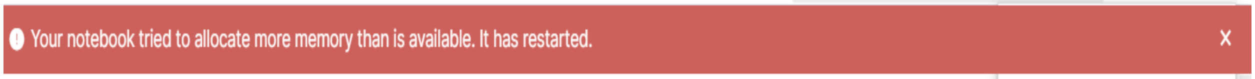
Out[12]:
```
(3536, 300)
```

NewData

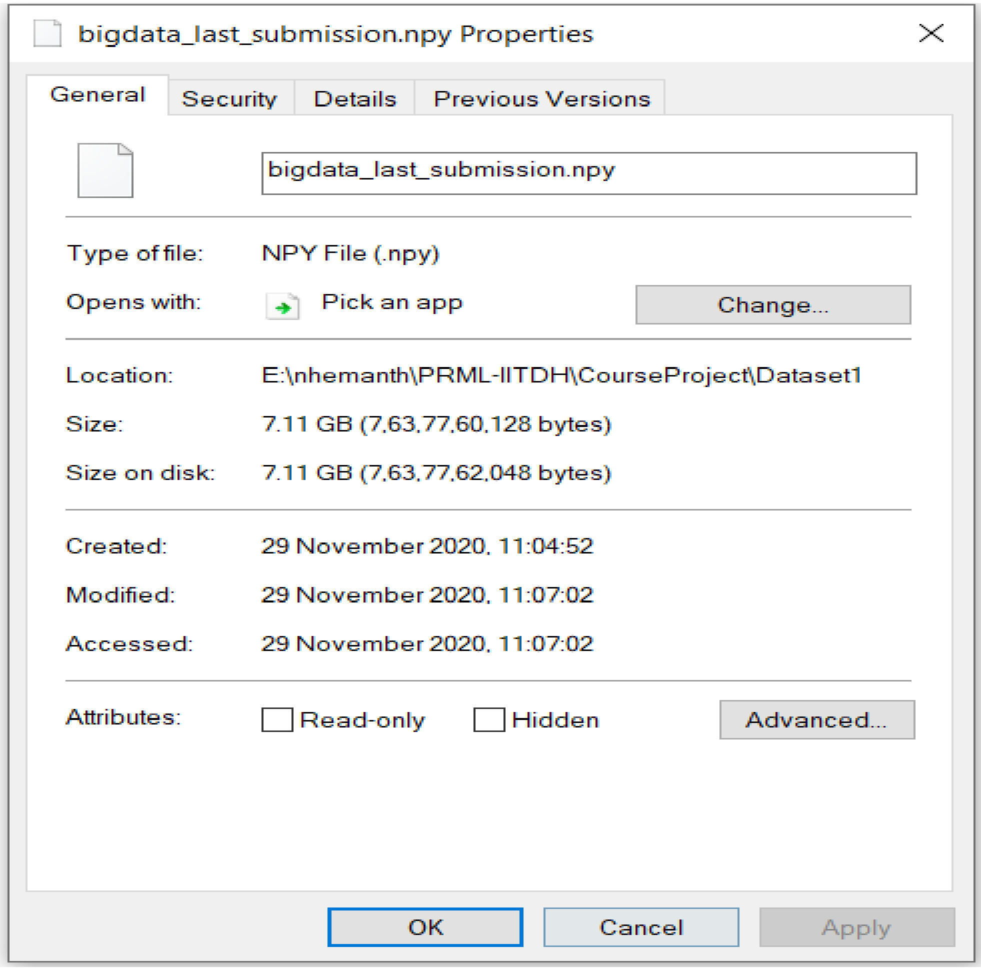
24 times augmented

10608 images - (300,300)

Have more data but could not train in my PC (results can be even more better with big dataset)



Your notebook tried to allocate more memory than is available. It has restarted.



bigdata_last_submission.npy Properties

bigdata_last_submission.npy

Type of file: NPY File (.npy)

Opens with: Pick an app

Location: E:\nhemanth\PRML-IITDH\CourseProject\Dataset1

Size: 7.11 GB (7,63,77,60,128 bytes)

Size on disk: 7.11 GB (7,63,77,62,048 bytes)

Created: 29 November 2020, 11:04:52

Modified: 29 November 2020, 11:07:02

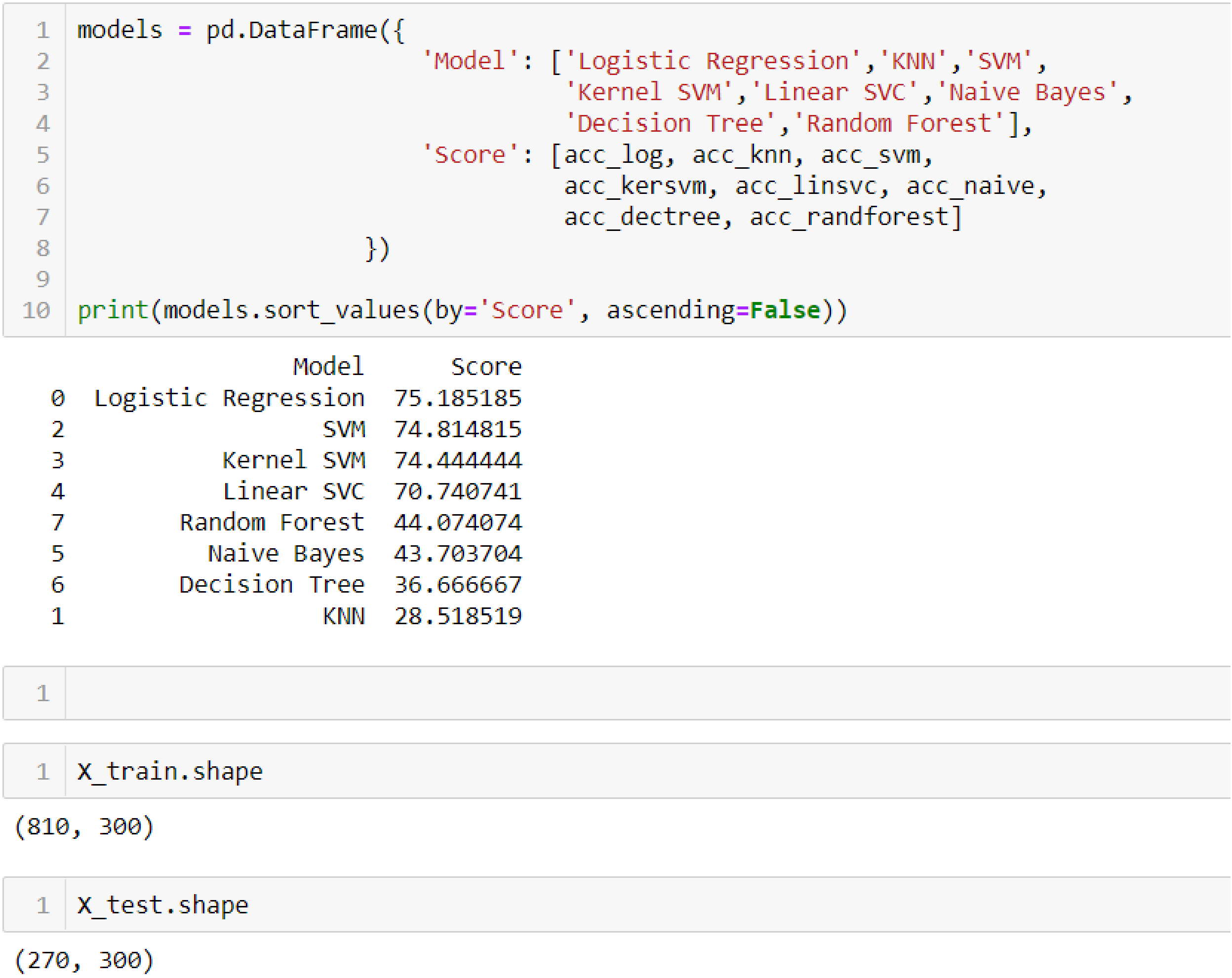
Accessed: 29 November 2020, 11:07:02

Attributes: □ Read-only □ Hidden

NewDataset

Same images

Total : 1080 images - (300,300)
- 25% split
- Train - 810, Test - 270



```python
models = pd.DataFrame({
                        'Model': ['Logistic Regression','KNN','SVM',
                                  'Kernel SVM','Linear SVC','Naive Bayes',
                                  'Decision Tree','Random Forest'],
                        'Score': [acc_log, acc_knn, acc_svm,
                                  acc_kersvm, acc_linsvc, acc_naive,
                                  acc_dectree, acc_randforest]
                    })

print(models.sort_values(by='Score', ascending=False))
```

```
                 Model      Score
0  Logistic Regression  75.185185
2                  SVM  74.814815
3           Kernel SVM  74.444444
4           Linear SVC  70.740741
7        Random Forest  44.074074
5          Naive Bayes  43.703704
6        Decision Tree  36.666667
1                  KNN  28.518519
```

```python
X_train.shape
```

```
(810, 300)
```

```python
X_test.shape
```

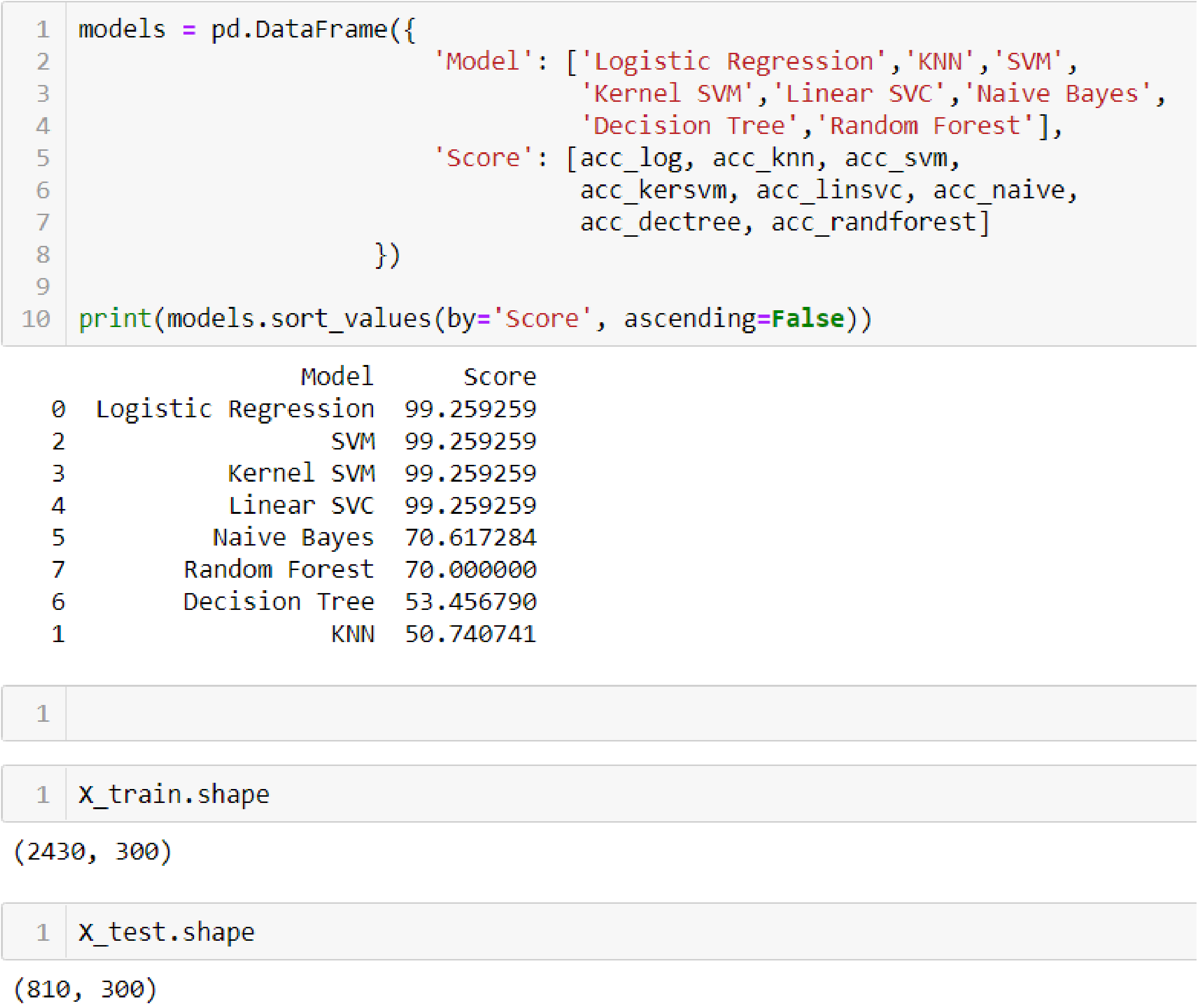
```
(270, 300)
```

NewData

3 times augmented(1080*3=3240)

Total : 3240 images - (300,300)

- 25% split
- Train - 2430, Test - 810



```python
models = pd.DataFrame({
                        'Model': ['Logistic Regression','KNN','SVM',
                                  'Kernel SVM','Linear SVC','Naive Bayes',
                                  'Decision Tree','Random Forest'],
                        'Score': [acc_log, acc_knn, acc_svm,
                                  acc_kersvm, acc_linsvc, acc_naive,
                                  acc_dectree, acc_randforest]
                    })

print(models.sort_values(by='Score', ascending=False))
```

```
                 Model      Score
0  Logistic Regression  99.259259
2                  SVM  99.259259
3           Kernel SVM  99.259259
4           Linear SVC  99.259259
5          Naive Bayes  70.617284
7        Random Forest  70.000000
6        Decision Tree  53.456790
1                  KNN  50.740741
```

```python
X_train.shape
```

```
(2430, 300)
```

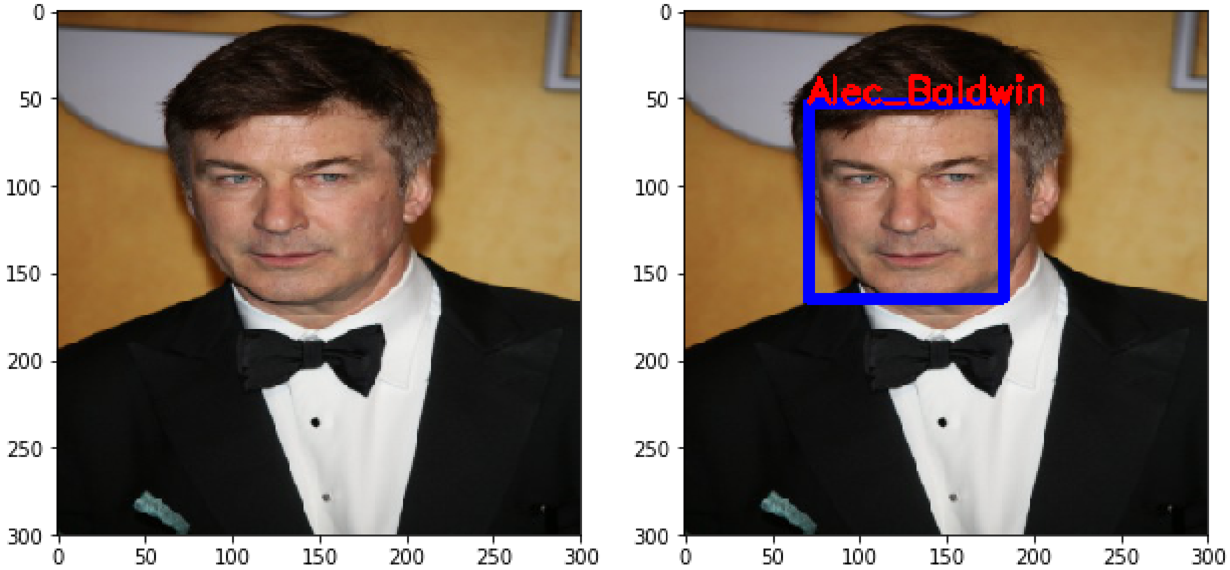
```python
X_test.shape
```

```
(810, 300)
```

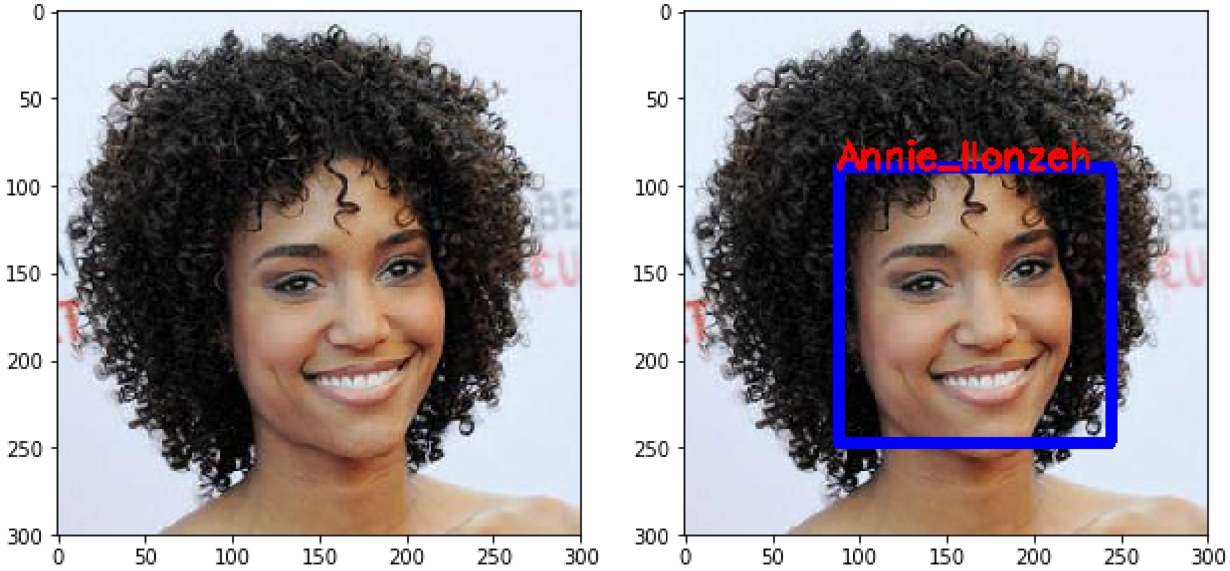
RESULTS

Single person:

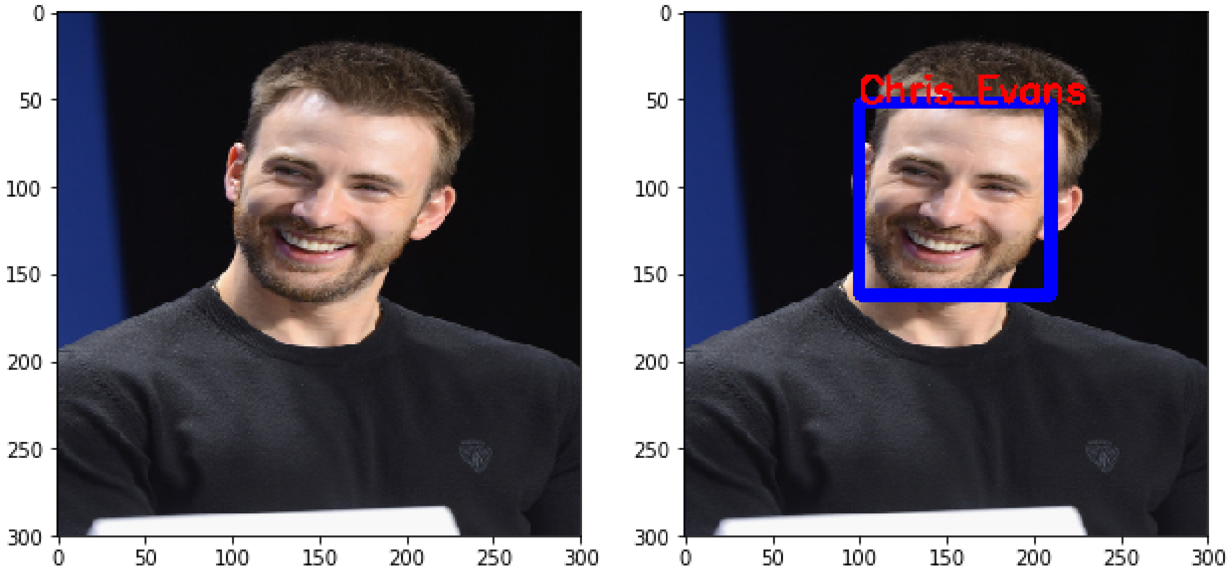
1.



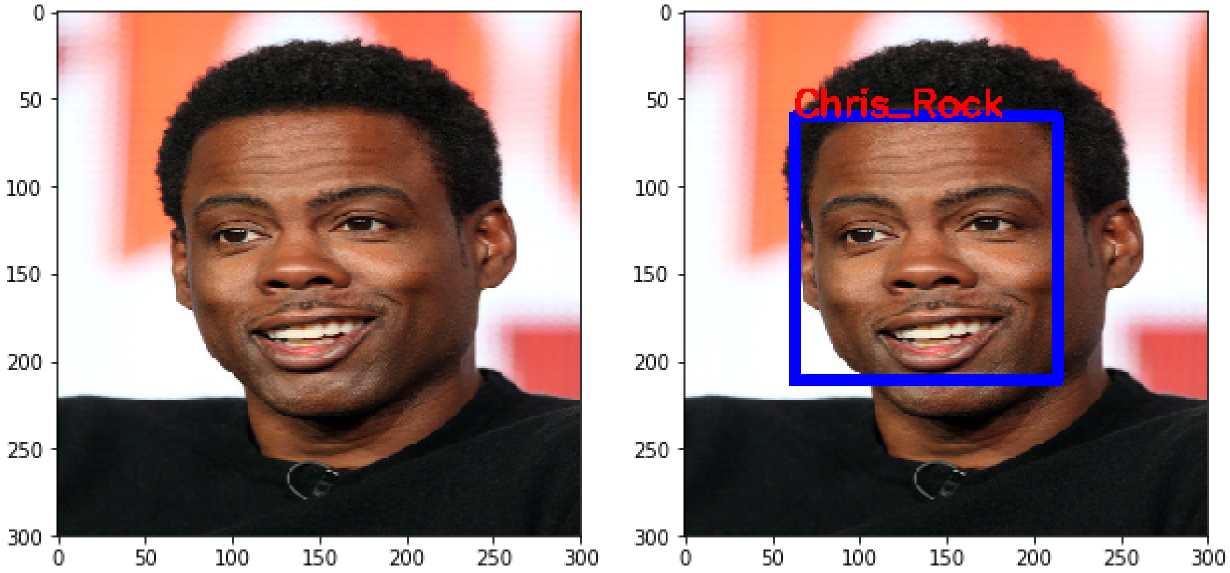
2.



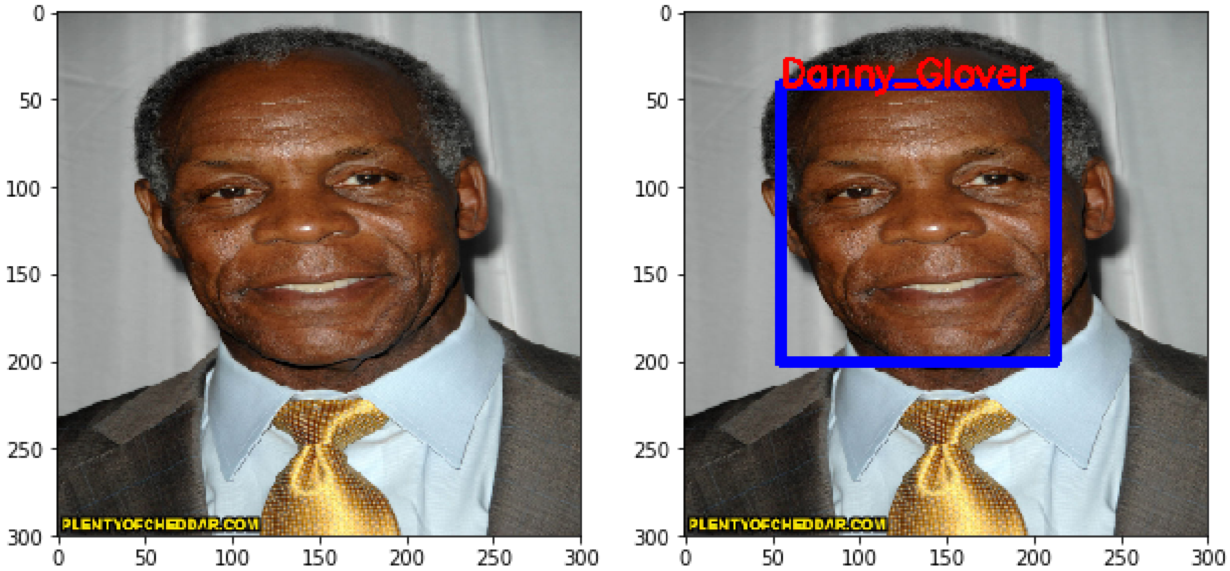
3.


0
50
100
150
200
250
300
0
50
100
150
200
250
300
Chris_Evans

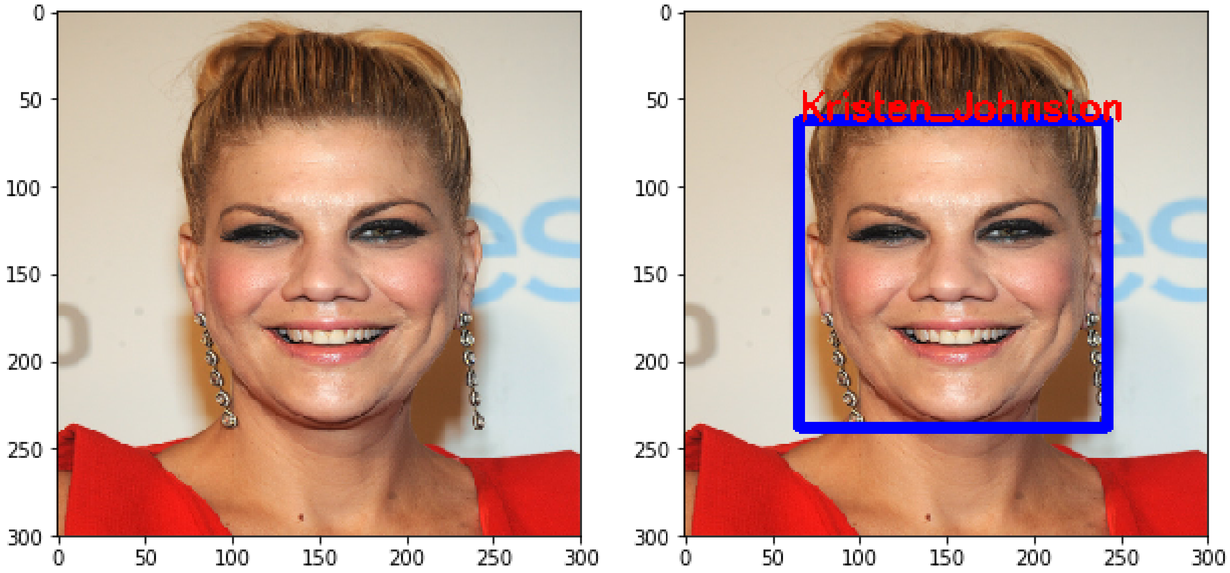
4.


0
50
100
150
200
250
300
0
50
100
150
200
250
300
Chris_Rock

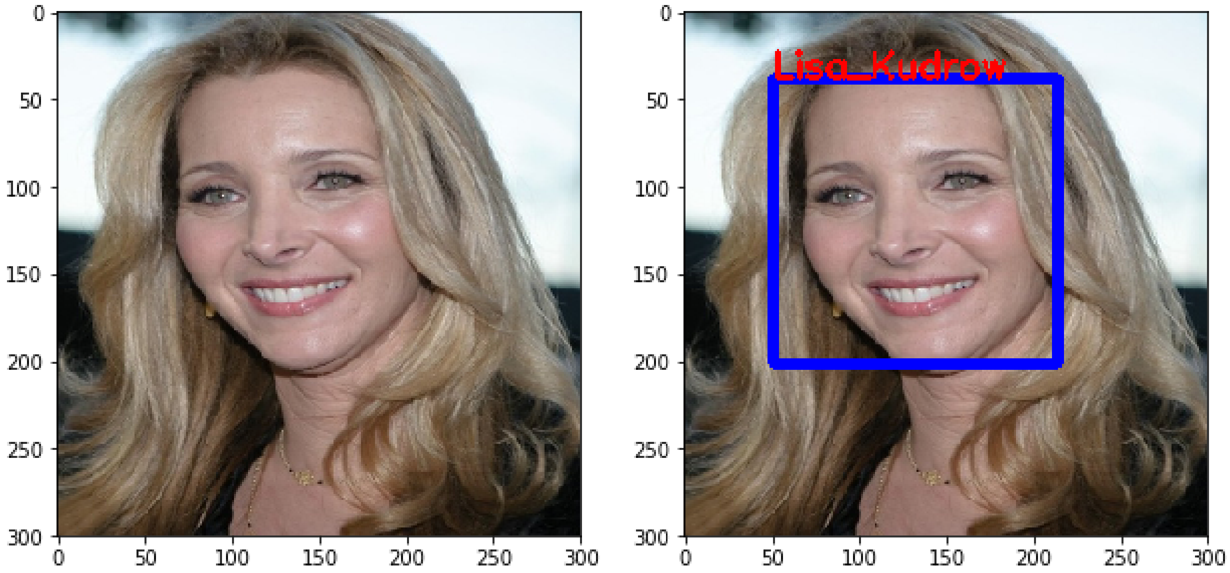
5.


PLENTYOFCHEDDAR.COM
Danny_Glover
PLENTYOFCHEDDAR.COM
0
50
100
150
200
250
300

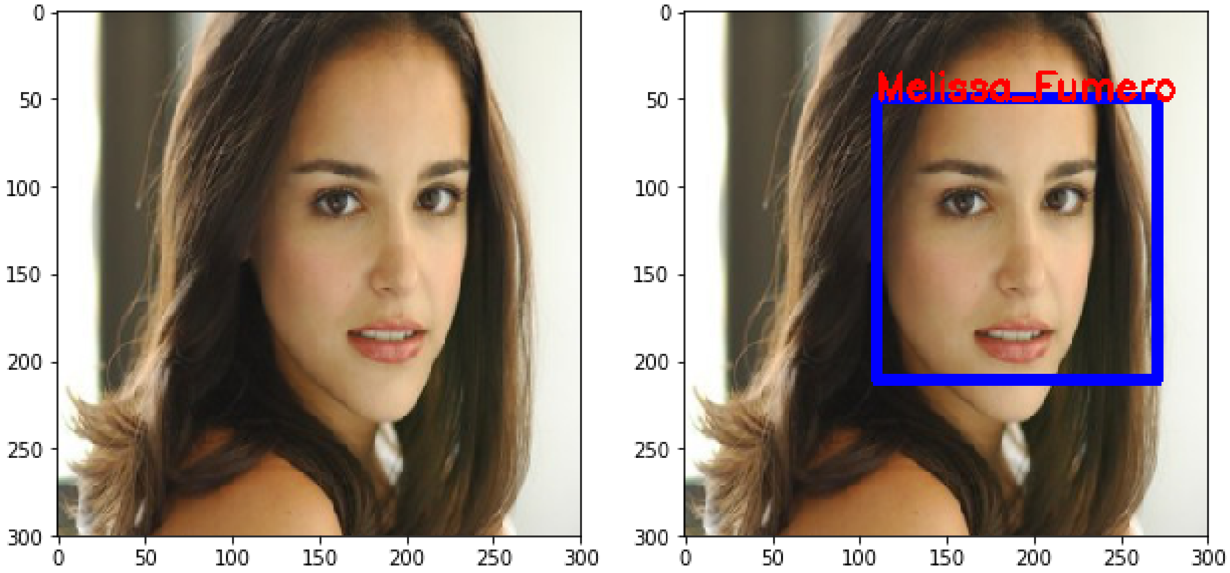
6.


Kristen_Johnston
0
50
100
150
200
250
300

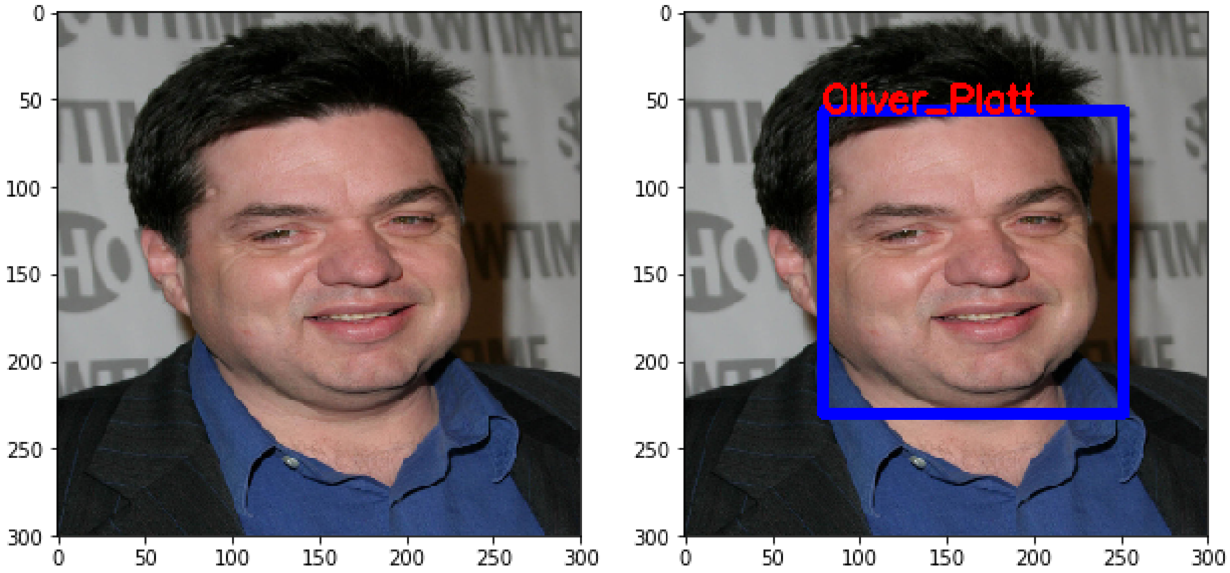
7.


0
50
100
150
200
250
300
0
50
100
150
200
250
300
Lisa_Kudrow

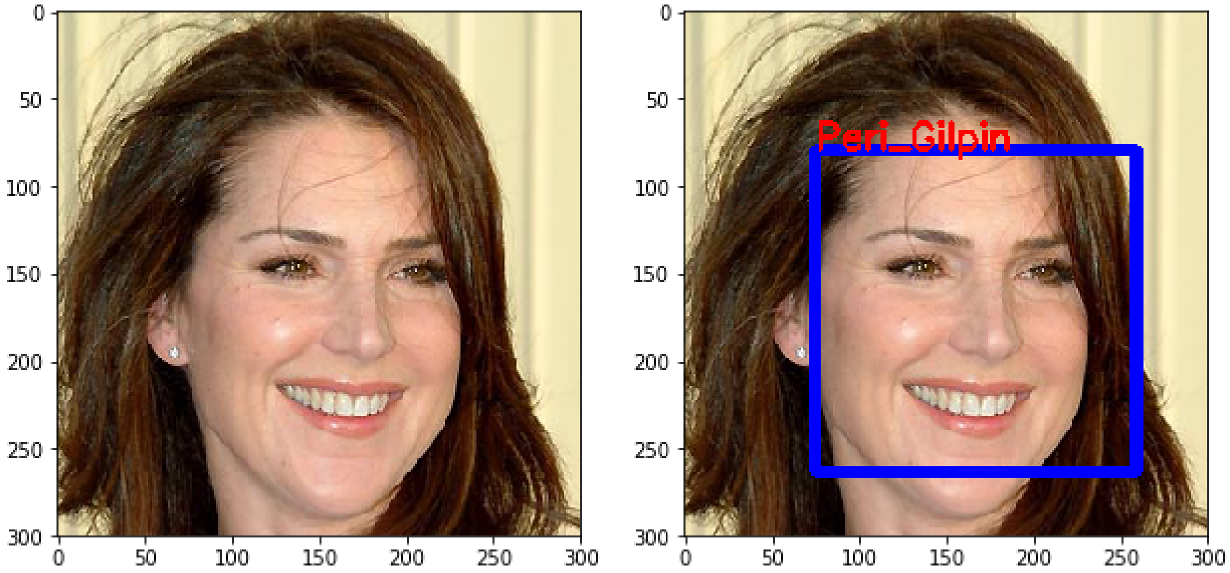
8.


0
50
100
150
200
250
300
0
50
100
150
200
250
300
Melissa_Fumero

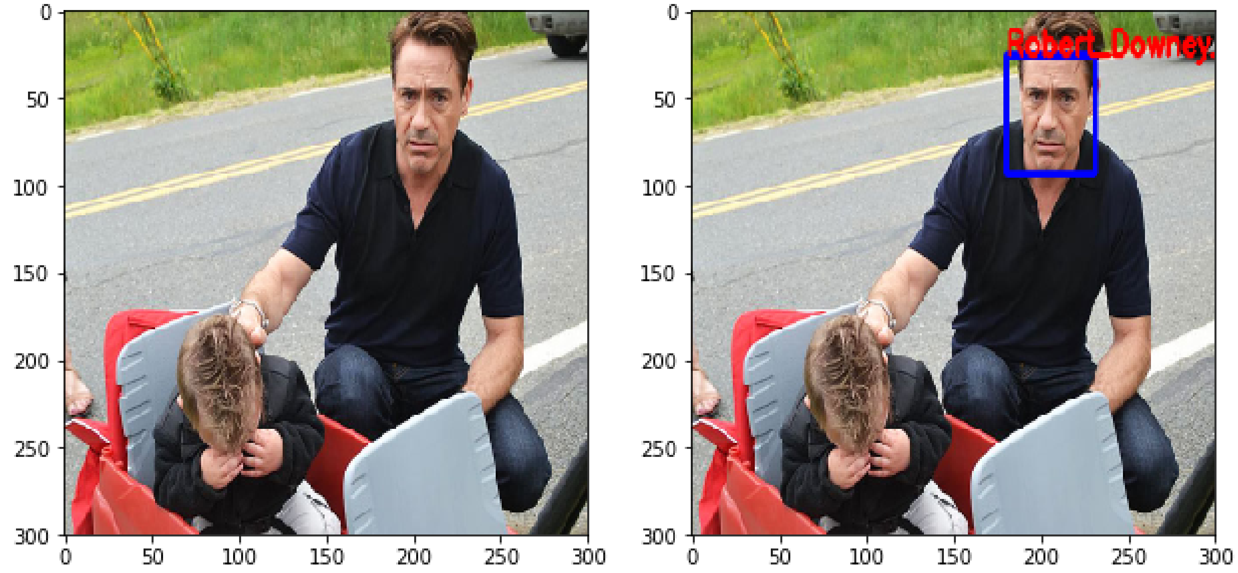
9.

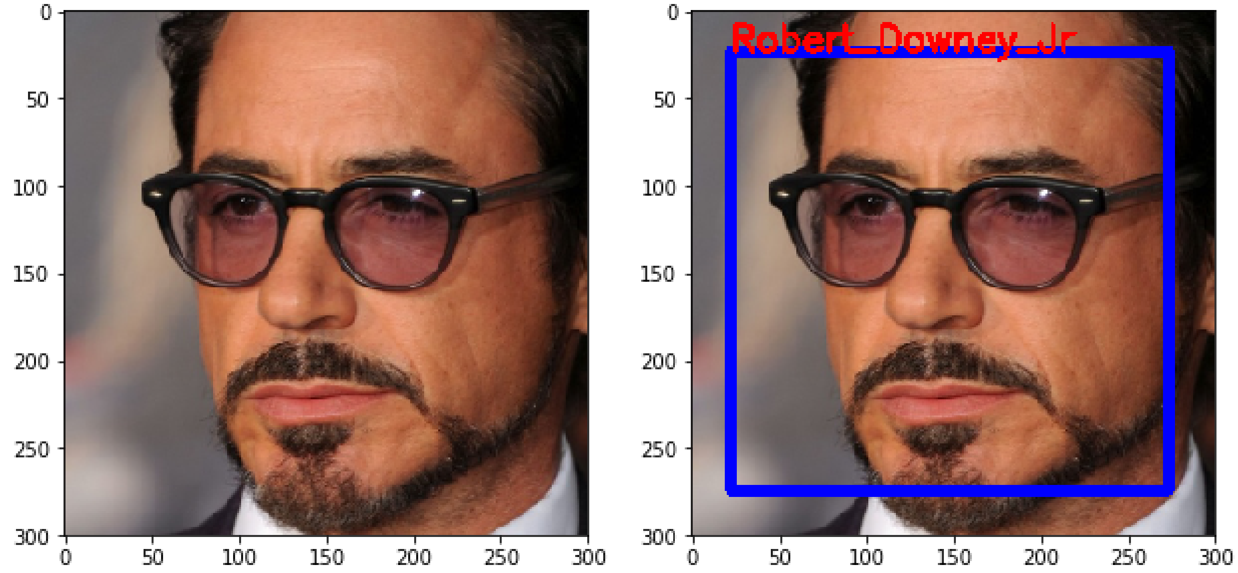

Oliver_Platt
0
50
100
150
200
250
300

10.

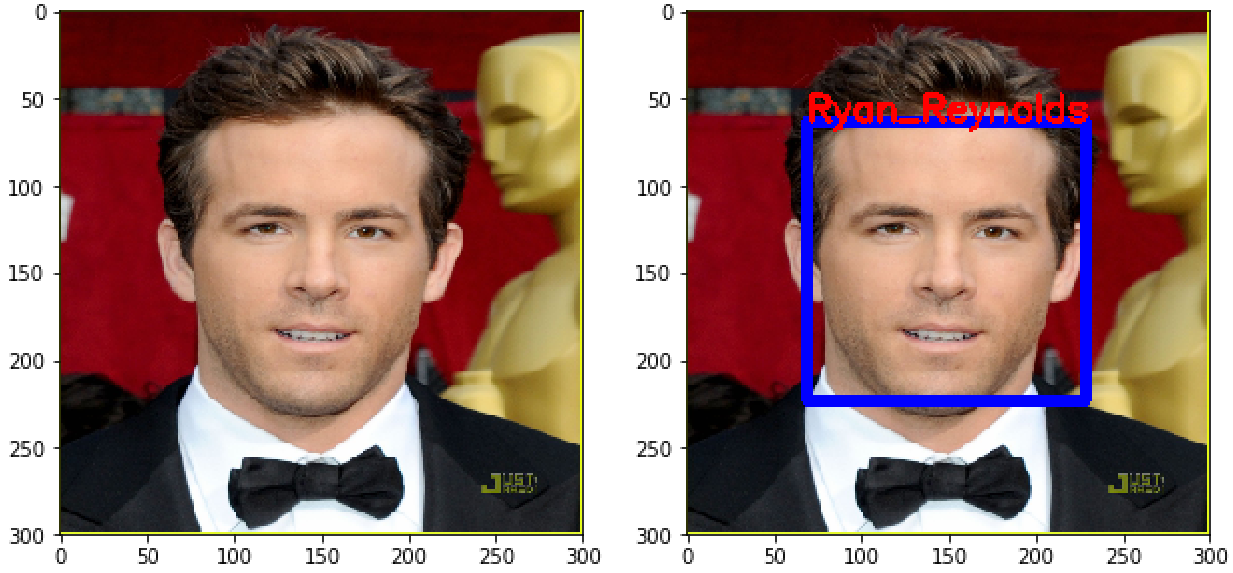

Peri_Gilpin
0
50
100
150
200
250
300

11.

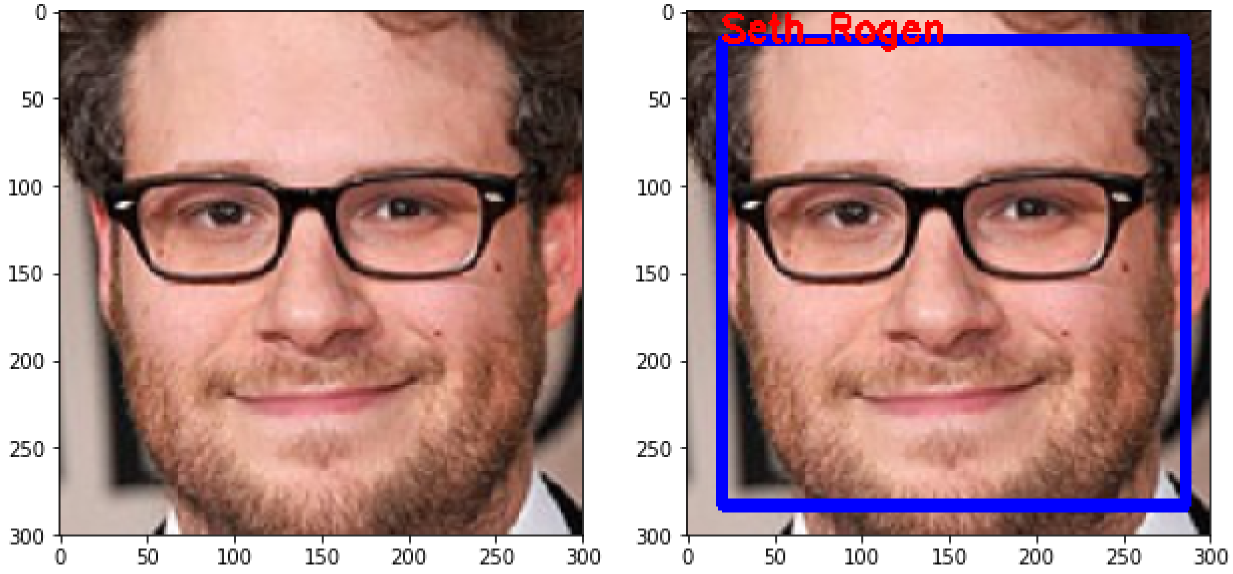

Robert_Downey.
0
50
100
150
200
250
300


Robert_Downey_Jr
0
50
100
150
200
250
300

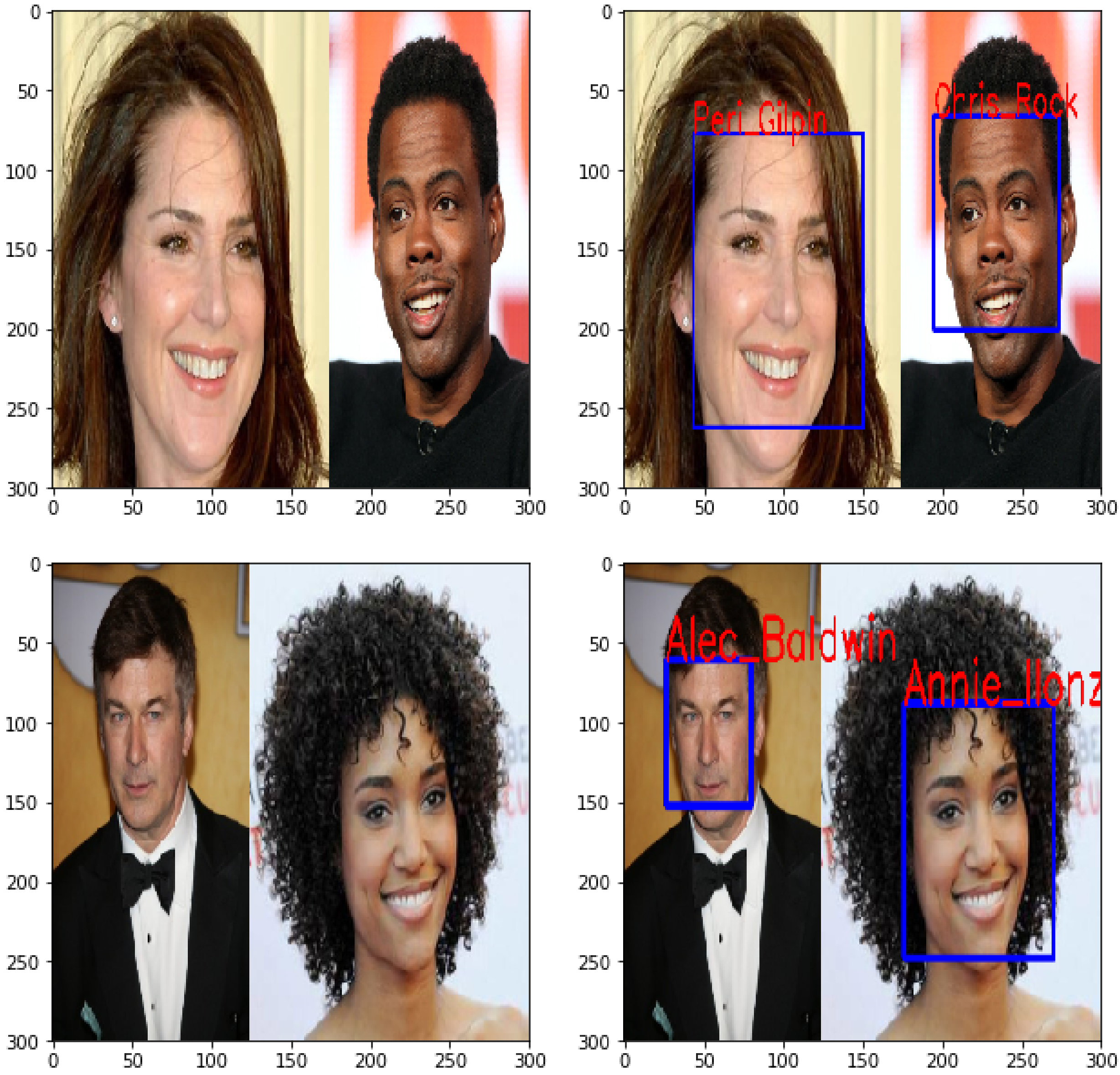
12.

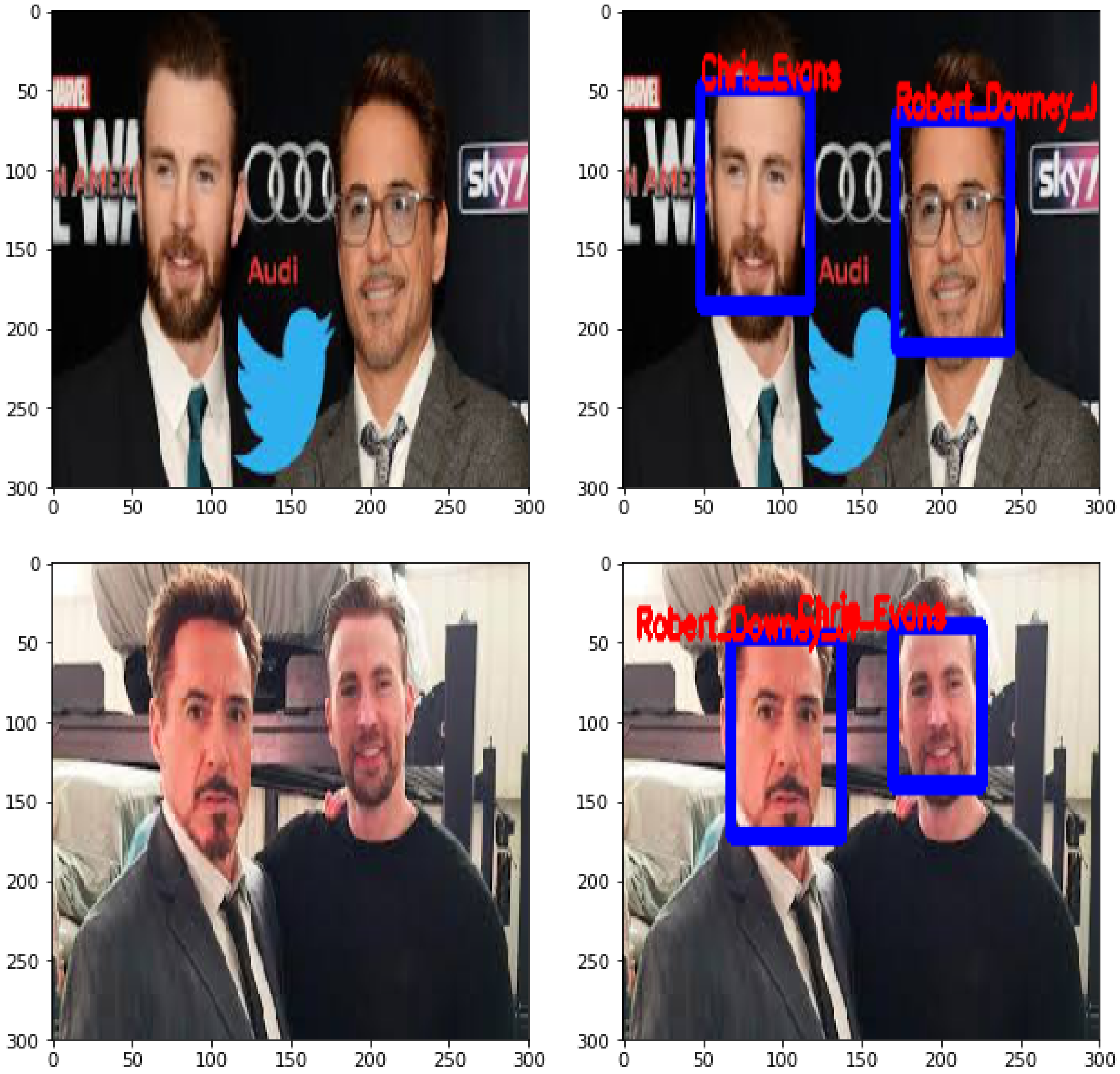


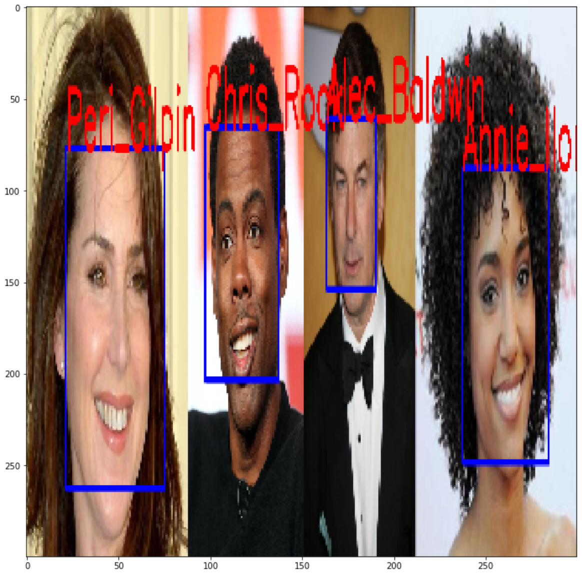
13.



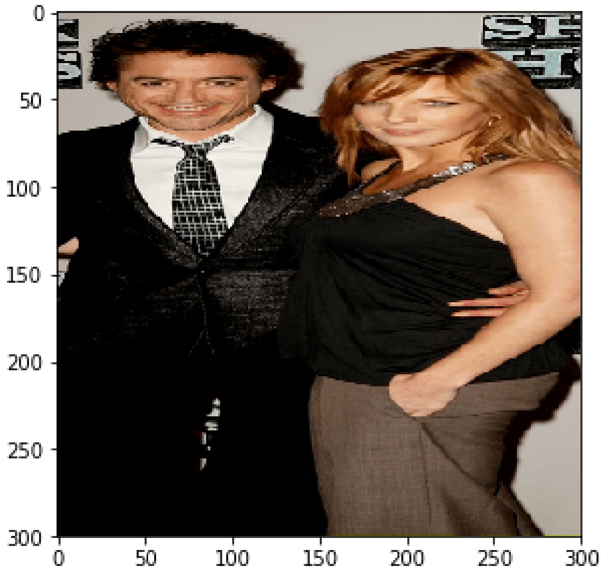
Multiple people :

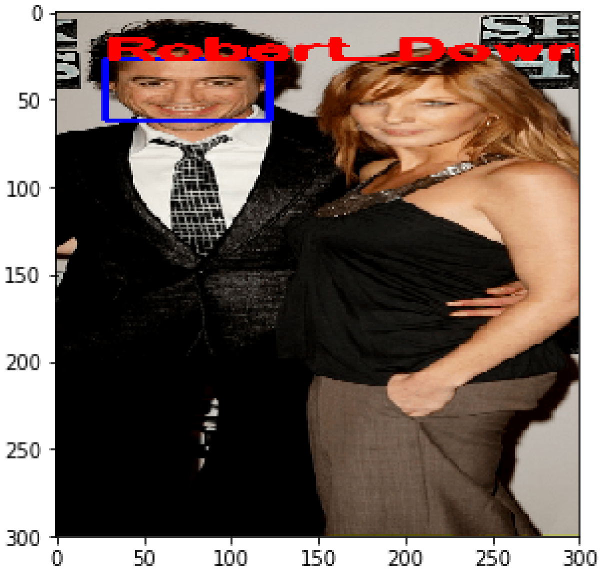

0
50
100
150
200
250
300
Peri_Gilpin
Chris_Rock
Alec_Baldwin
Annie_Ilonz

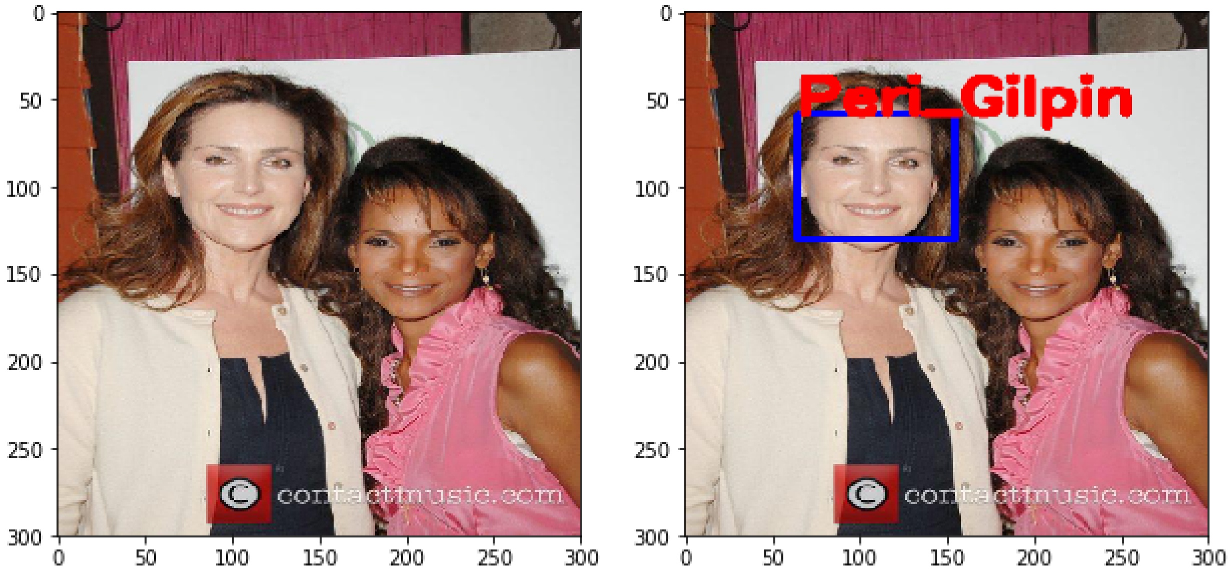

0
50
100
150
200
250
300
MARVEL
Audi
sky
Chris_Evans
Robert_Downey_J
Robert_Downey_J
Chris_Evans



Multiple people : predicted only trained people






0
50
100
150
200
250
300
Peri_Gilpin
contactmusic.com

0
50
100
150
200
250
300
Lisa_Kudr