Pneumonia detection on chest X-ray images

Meet our team

Alsu Vakhitova

Alsu.Vakhitova@skoltech.ru

Batyrkhan Gainitdinov

Batyrkhan.Gainitdinov@skoltech.ru

Mohammed Deifallah

Mohammed.Deifallah@skoltech.ru

1. Problem Description

2,560,000

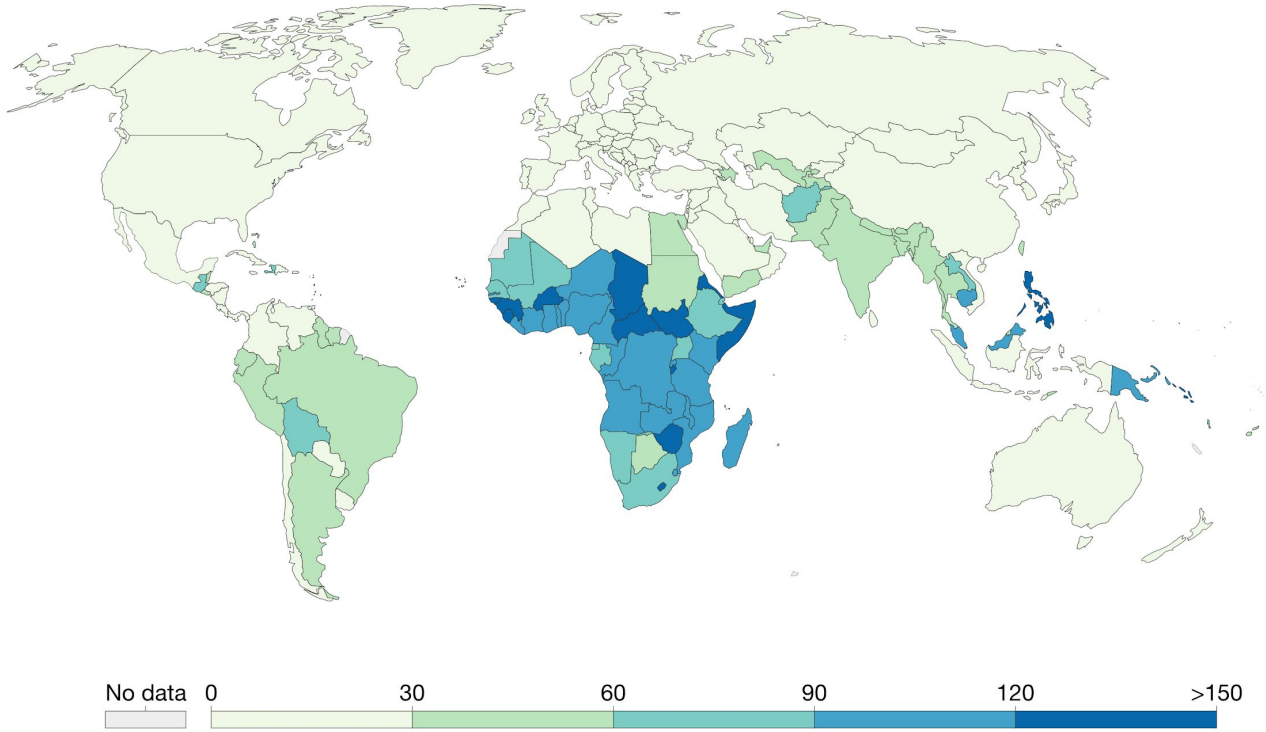
people died from pneumonia in 2017

https://ourworldindata.org/pneumonia

Death rate from pneumonia, 2017

The annual number of deaths from pneumonia per 100,000 people.

Our World in Data



Source: Global Burden of Disease Study, IHME (2018)

OurWorldInData.org/pneumonia • CC BY

Note: To allow comparisons between countries and over time this metric is age-standardized. Deaths from 'clinical pneumonia', which refers to a diagnosis based on disease symptoms such as coughing and difficulty breathing and may include other lower respiratory diseases.

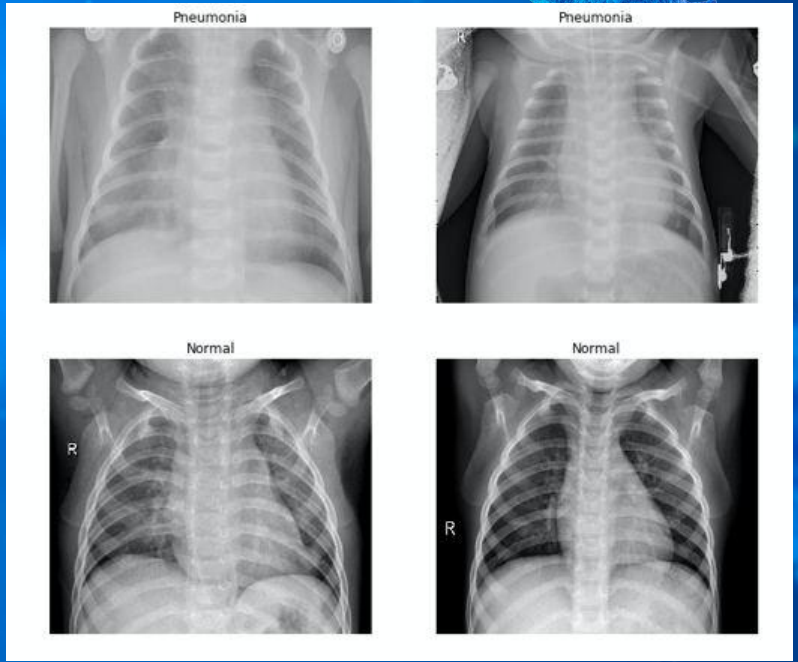

Pneumonia is one
of the most lethal
COVID-19
complications

Expected value

- Rapid screen of all patients
- Aid in early detection of the disease
- Protection of health care workers

Dataset description

- Patients of 1-5 years old
- 5,863 X-Ray images
- 2 categories: Pneumonia/Normal

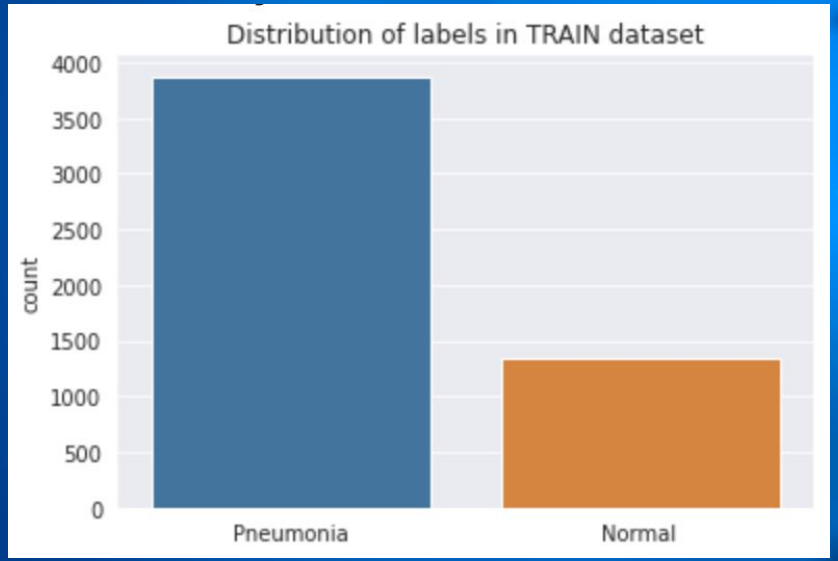


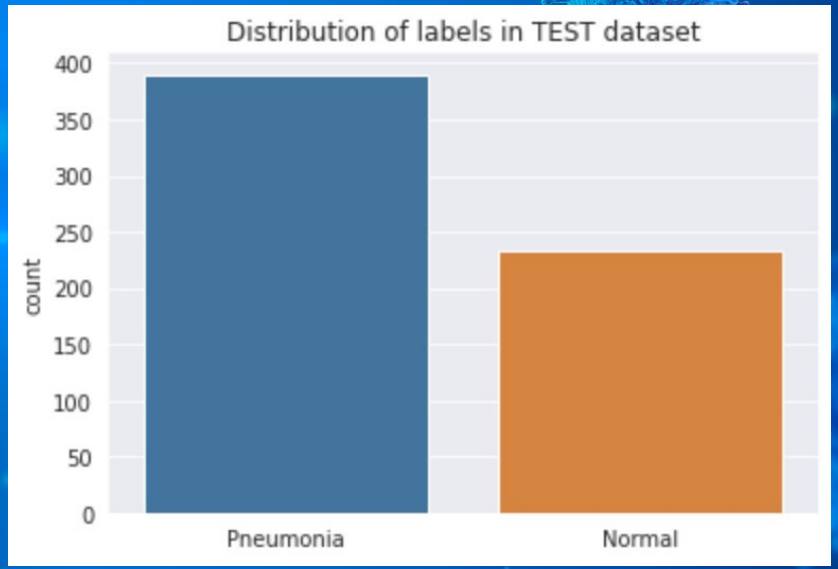
Additional dataset description

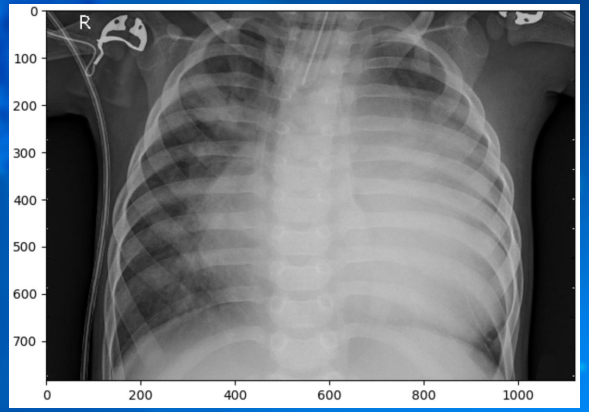
- From the personal archive by the doctor of Sechenov university, department of Pulmonology prof. Gainitdinova Viliya Vilevna
- 64 X-Ray images
- Only pneumonia

2. CV Algorithms

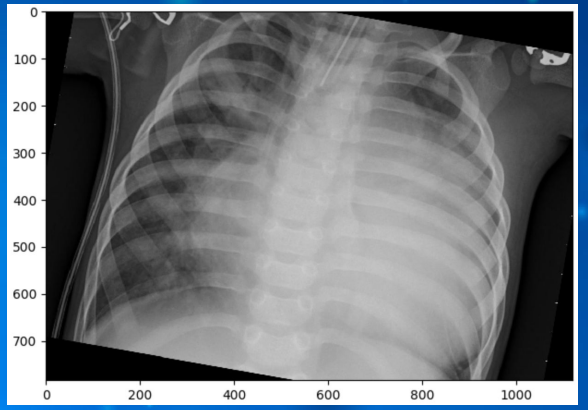
Data augmentation



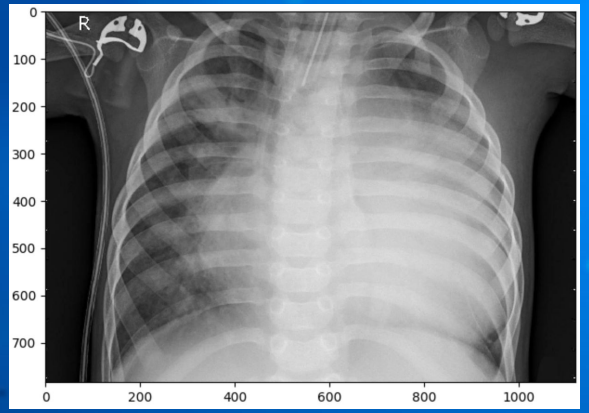




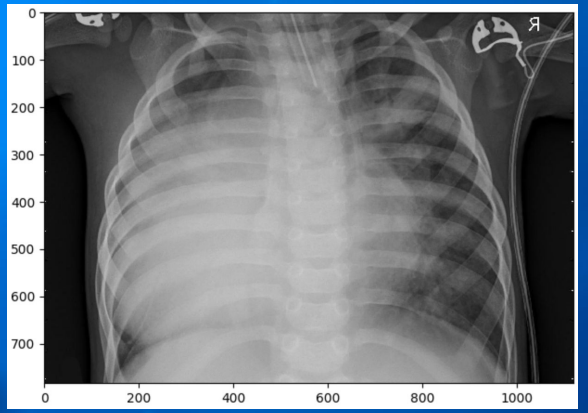
Original



Random Rotation

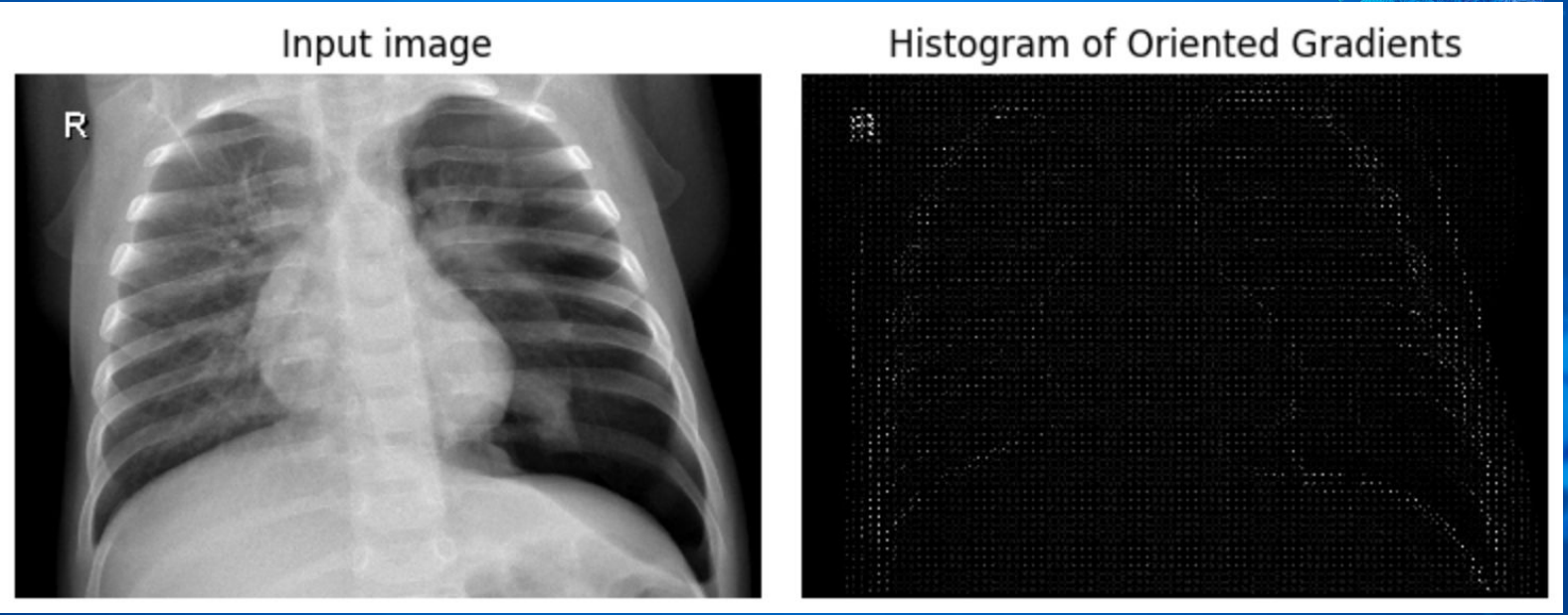


Random Brightness



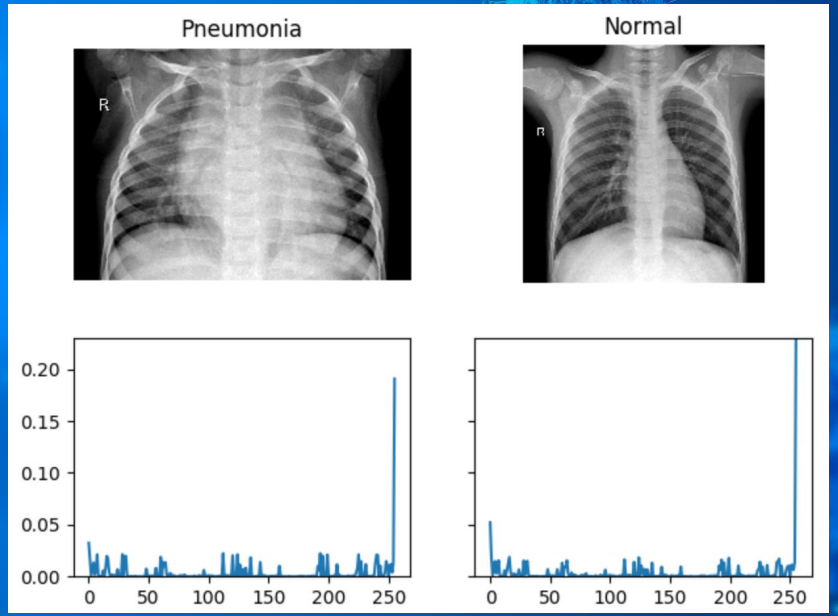
Horizontal Flip

Feature extraction: HOG



Feature extraction: LBP

- Local Binary Patterns
- Texture descriptor
- Labels the pixels of an image by thresholding the neighborhood of each pixel and considers the result as a binary number



More Information: https://www.pyimagesearch.com/2015/12/07/local-binary-patterns-with-python-opencv/

3. Results

HOG + SVC (with data augmentation)

1. GridSearch parameters (cv = 5, verbose = 10):

```
param_grid = {'C': [0.001, 0.01, 0.1, 1, 10, 100],
'gamma': [0.001, 0.01, 0.1, 1, 10, 100]}
```

```
grid_search.best_params_
```

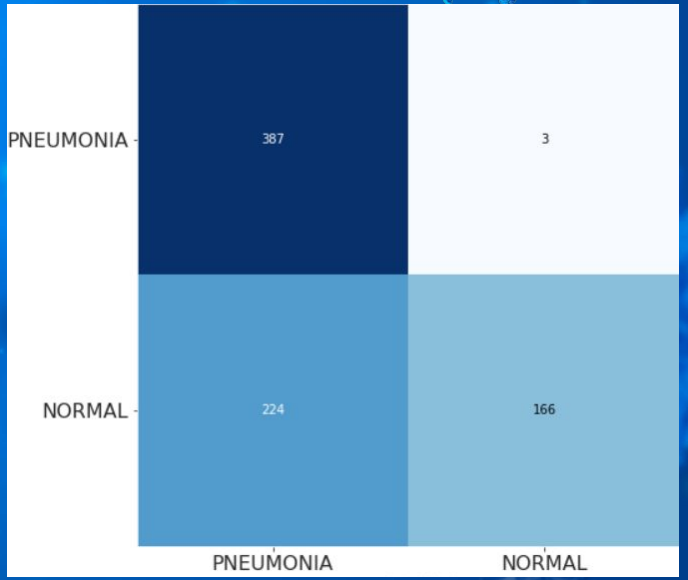
```
{'C': 10, 'gamma': 0.1}
```

2. With the accuracy:

```
print(accuracy_score(labels_test,labels_hog_aug_pred))
```

```
0.708974358974359
```

Results:



HOG + SVC (without data augmentation)

1. GridSearch parameters (cv = 5):

```
param_grid = {'C': [0.01, 0.1, 1, 10, 100],
              'gamma': [0.001, 0.01, 0.1, 1]}
```

```
grid_search.best_params_
```

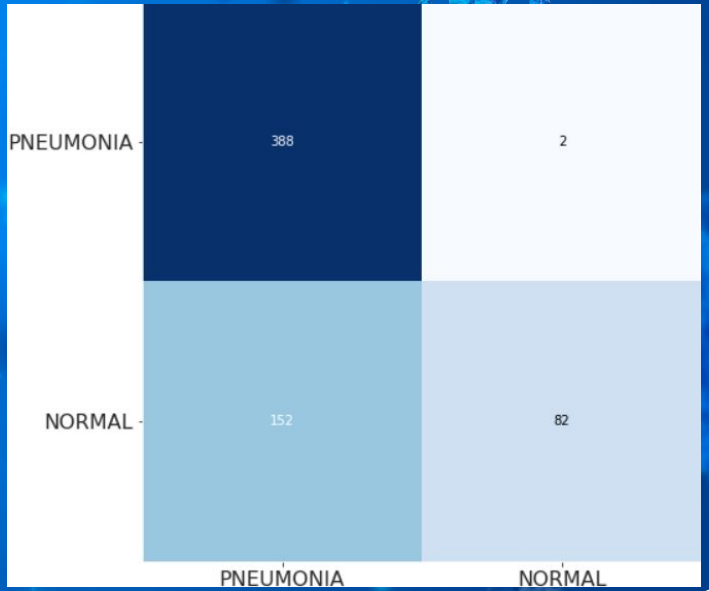
```
{'C': 0.01, 'gamma': 0.001}
```

2. With the accuracy:

```
print(accuracy_score(labels_test,labels_hog_pred))
```

```
0.7532051282051282
```

Results:



LBP + SVC (with data augmentation)

1. GridSearch parameters (cv = 5):

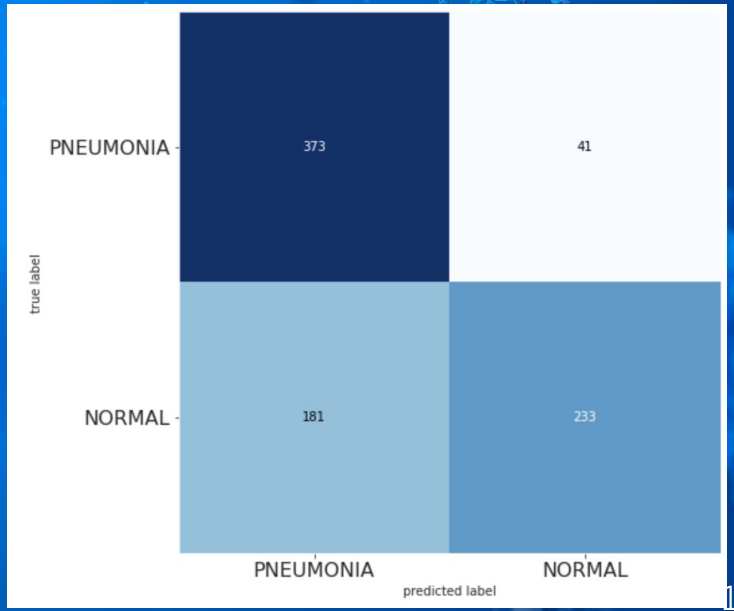
```
C_vals = np.logspace(start=-1, stop=4, num=10, endpoint=True, dtype=float)
params = {'C': C_vals,
          'gamma': np.logspace(start=-1, stop=3, num=10)}
```

```
[51] grid_search_lbp.best_params_

     {'C': 215.44346900318845, 'gamma': 1000.0}
```

2. With the accuracy:

```
print(accuracy_score(y_aug_test, y_aug_test_pred))
print(grid_search_lbp.best_estimator_.score(X=X_aug_test, y=y_aug_test))

0.7318840579710145
0.7318840579710145
```

Results:



LBP + SVC (without data augmentation)

1. GridSearch parameters (cv = 5):

```
C_vals = np.logspace(start=-2, stop=4, num=15, endpoint=True, dtype=float)
params = {'C': C_vals,
          'gamma': np.logspace(start=-5, stop=5, num=15)}
```

```
grid_search_lbp.best_params_
```

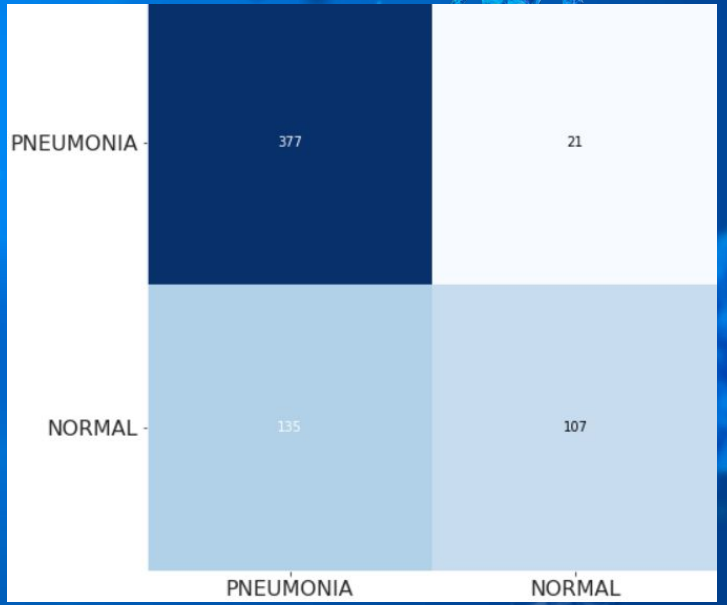
```
{'C': 517.9474679231213, 'gamma': 26.826957952797272}
```

2. With the accuracy:

```
print(accuracy_score(y_test, y_test_pred))
print(grid_search_lbp.best_estimator_.score(X=X_test, y=y_test))
```

```
0.75625
0.75625
```

Results:



4. Conclusion

Results analysis

- Augmentation+LBP+SVC is the best according to confusion matrix
- Data augmentation may not help if imbalance is too dramatic
- Simple classical algorithms can still give decent results
- Results can be improved by:
  - Stronger classification algorithms (for example, Logistic Regression)
  - Neural Networks

THANKS!

Any questions?

You can find us at:

- Alsu.Vakhitova@skoltech.ru
- Batyrkhan.Gainitdinov@skoltech.ru
- Mohammed.Deifallah@skoltech.ru