Feature Evaluation for Detection of COVID-19 in Pulmonary CT Scan Images

Andrei Popa
University of Twente
P.O. Box 217, 7500AE Enschede
The Netherlands
a.popa@student.utwente.nl

ABSTRACT

At the beginning of 2020 a novel coronavirus, SARS-CoV-2, has started its spread, which has become a global issue. The current method of detecting the infection with the disease (COVID-19) caused by the virus is the transcription polymerase chain reaction (RT-PCR), which is limited by the low detection rate when the viral load is low and the high demand for it, which means that a new detection method is needed. Another method, that has the possibility to detect the signs of COVID-19, is through medical imaging, like computed tomography (CT) scan or X-rays scans of the chest.
The method that is presented by this paper uses the patches of the pulmonary 2D CT scans to extract feature vectors and a support vector machine (SVM) classifier to detect the affections caused by the disease. The aim is to see which characteristics of the affection, namely color, texture, position or the combination of all three of them, are better suited in detecting the signs of the disease.
In order to evaluate the results different values for regularization and class weights parameters are used and the precision, recall and Matthews correlation coefficient are computed. The dataset contains 100 CT scan images of patients that show signs of the disease.
The results show that the best single feature vector that can be used to detect the affections caused by the disease is the color, while the best results were achieved by the combination of all three features.

Keywords

COVID-19, CT scans, disease detection, image processing

1. INTRODUCTION

1.1 Background

The novel severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) is an infectious disease, which was first identified in December 2019, and since, its spread resulted in a pandemy. While there are several ways of detecting the virus, the most used method is the reverse transcription polymerase chain reaction, RT-PCR. However, it is shown to have several limitations, like low detection rate in case of low viral load or the supply rate not keeping up with the demand [1]. Because of these limitations, the detection of the virus through computed tomography (CT) scan images is strongly recommended [1].

28th Twente Student Conference on IT Febr. 2nd, 2018, Enschede, The Netherlands.

1.2 Literature Review

There are several possible ways of detecting different diseases or the infection with the virus through CT scans. The first one implies the usage of 3D image reconstruction. It was shown that a convolutional neural network can be used to detect COVID-19, by using a series of CT slices as input in order to generate a series of features that characterize the pulmonary lesions caused by it and to ease the detection performed by the network, the lung regions are extracted as the region of interest [2]. The result of using convolutional networks and 3D image reconstruction were favourable, a high accuracy was achieved in detecting the pulmonary affection caused by the disease and also differentiating it from pneumonia or other lung diseases [2]. However, 3D reconstruction has its limitations. The dataset used for one reconstruction needs to be big and the CT scan images need to have no unwanted artifacts, in order to properly 3D reconstruct the image of the lungs of the patient. While the artifacts can be removed by preprocessing the images, the need of a big dataset can actually endanger the patients, because CT scanning uses X-rays.
The second possibility is the use of 2D CT scan images. It was shown that detection of the infection with COVID-19 using 2D scan images has a good accuracy [4]. First a preprocessing of the images was done, by applying threshold filters and cropping the lung section. After this a deep learning model was trained, by extracting the features of the images from the training dataset [4]. Another promising method of detecting the COVID-19 using 2D CT scan images also implies the use of other data about the patients like clinical symptoms, exposure history and laboratory testing [6]. However, detection using 2D CT scans also has its own limitations. They usually contain a variety of image artifacts, like quantum noise or nonlinear partial volume effects[8]. In order to remove the image artifacts that can hinder the process of detecting COVID-19, different image enhancement techniques can be used, like automatic contrast enhancement, deblurring, or intensity or color modification [8].
A third possible method of detecting the COVID-19 affection through CT scan images is the use of both 2D and 3D images [3]. In order to detect the signs of the disease a deep learning algorithm was used, which also makes use of clinical data from each patient [3]. This method was also shown to be able to assess the disease by using 3D volume analysis [3].
Beside a deep learning algorithm, the detection methods also use image segmentation in order to extract the lung area from CT scans. There are several ways to extract

the lungs area from a CT scan, beside thresholding, which were proposed throughout time. One proposed method is a neural network based on shape analysis [9]. The network algorithm uses the gray levels and the information of neighbouring pixels in order to segment each image [9]. Another possible method of segmentation is the use of color-edge extraction and seeded region growing, where after finding the edges, the centroids of each area and of the adjacent areas are used as seeds for the seeded region growing [5]. While these methods can be proved to be more precise for segmenting the lung in the long run, it takes a fair amount of time to implement the proposed algorithm or training the neural network. The neural network option was shown to have achieved poor results, the main issue being that the organs that were needed to be extracted have substructures that have different textures and those textures are not well-defined [9]. Another problem that arises when a neural network is used are the requirement of a large dataset and the impossibility of improvement of the network due to its opacity [9]. While the method proposed in [5] might be considered a better option than a neural network, it also has its drawbacks, the most important being implementation time.

1.3 Objective and Goals

To avoid the major problems that can be caused by 3D image reconstruction, I propose a model that uses a classifier to detect the disease from 2D CT scan images. It was found that the pulmonary tissue affected by the virus has several main characteristics on CT scan images: ground-glass opacity, tiny nodules, fine mesh shadows and peripheral, asymmetric and posterior distribution [7]. Given these characteristics, the classifier will be trained to detect the signs of the disease using the following feature vectors: color, position, texture and the combination of all three feature vectors.
The characteristics of the affections caused by COVID-19 are important in training a classifier, but it can happen that some characteristics are better in detecting the infection than others, while other characteristics of it are similar to other pulmonary diseases. By comparing the precision of detection of the disease, using one of these characteristics or a combination of them, it can be determined which is more suitable for future research in detection of signs of COVID-19 in CT scan images.

1.4 Research Questions

Given the context of this paper the following research question is going to be investigated: *"How does the position, color or texture of pulmonary affections can be used to detect the infection with COVID-19 in CT scan images?"*. To answer the question, first it needs to be split into smaller questions, whose answer is the results of each experiment. The supporting questions are the following:

Q1: How does the position of pulmonary affections influence the detection of infection with COVID-19 in CT scan images?

Q2: How does the color of pulmonary affections influence the detection of infection with COVID-19 in CT scan images?

Q3: How does the texture of pulmonary affections influence the detection of infection with COVID-19 in CT scan images?

Q4: How does the combination of all characteristics, position, color and texture, influence the detection of COVID-19 in CT scan images?

Q5: How does image preprocessing using color mapping and thresholding help in detection of pulmonary affections in 2D CT scan images?

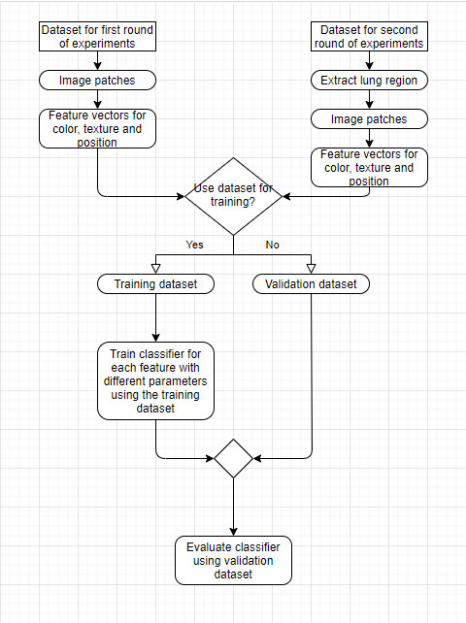

Figure 1. Flow chart of the methods used

2. METHODS

The method I propose includes: lung region extraction, using image preprocessing, and training of a classifier which can predict whether a CT scan image has signs of COVID-19 or not. For training the following steps are taken:

1. Preprocess images
2. Extract feature vectors from each patch, for each characteristic of the disease
3. Train classifier using the feature vectors

To test the classifier the following steps are taken:

1. Preprocess images
2. Extract feature vectors from each patch, for each characteristic of the disease
3. Use classifier to predict each patch

2.1 Lung Region Extraction

The areas outside the lung parts are not relevant when training the classifier, so extracting the lung regions from the image and using those to train the classifier can improve the accuracy of the predictions while also lowering the training time. To extract the lung regions two methods are used. The first one uses color mapping and thresholding. It can be seen in Figure 4 that by only applying a threshold to the CT scan image from Figure 2, where the effects of COVID-19 are quite severe, there is a lot of information lost, because the effects blend with the background, which can decrease the accuracy of classifying the image. However, by first applying a color map where brighter colors are changed to red nuances while darker colors are changed to blue nuances and then thresholding, as it can be seen in Figure 3, the lung regions and the pulmonary affections have a better delimitation.

The second method uses bounding boxes which are drawn around the lung regions of the image, which can be seen in Figure 5.

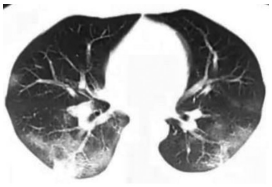

Figure 2. Original pulmonary CT scan

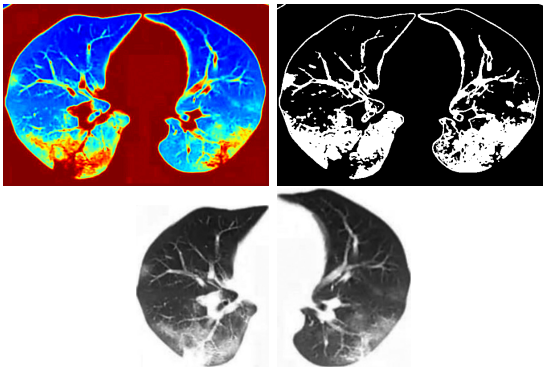

Figure 3. Pulmonary CT scan preprocessed using color mapping and thresholding and resulting lung sections

Figure 4. Pulmonary CT scan preprocessed using only thresholding

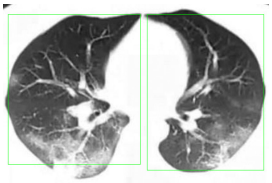

Figure 5. Pulmonary CT scan with lung bounding boxes

2.2 Feature vectors

The training and validation datasets consist of feature vectors which are extracted from patches of the CT scan. A patch is a section of the image with a set size, which can either contain signs of COVID-19 or not. They are used to extract the feature vector for each characteristic, namely color, texture and position, that are used to detect the signs of the disease.
The color characteristic for the signs of COVID-19 is gray, which is also called ground-glass opacity. The feature vector for color consists of the mean and standard deviation of each color channel.
The texture of the signs of COVID-19 can be described as having fine mesh shadow and tiny nodules [7]. The feature vector for texture consists of the histogram of the local binary pattern of the patch. Local binary pattern (lbp) is a visual descriptor which gives each pixel a value based on the value of its 8 neighbors. If a neighboring pixel has a smaller value than the current pixel then 0 is added to the value of it in the resulting patch, else 1. This results in a binary value for the current pixel, which is changed to decimal as a final step. The values of the lbp range from 0 to 255, and their distribution is represented using a histogram with 8 bins. Lbp is one of the most common ways to describe the texture features of an image, so it is used to describe the texture of the signs of COVID-19.
It was shown that the affections caused by COVID-19 have a posterior and lateral distribution [7], meaning that the signs can be seen in the bottom and side of lungs on a CT scan. The feature vector for the position consists of coordinates of the upper left corner, lower right corner and the center of the patch.

2.3 Classifier

To detect COVID-19 signs in 2D CT scan images a supporting vector machine (SVM) is used, which takes feature vectors as training dataset. It builds the model by creating a linear function where the value given is the feature vector and the result is the class the vector is labeled as. There are two types of data that are used for training a classifier. The first one is called linearly separable, where a linear function, or hyperplane, can be used to properly separate the data for each class into planes, while the second type is non linearly separable. In the case of an image there might be feature vectors that are similar, while their classes are different which means that the data can not be linearly separated. In this case the classifier will try to find a soft margin, by minimizing the following function, where w is the normal vector to the hyperplane, x is the feature vector, y is the class the feature vector is labeled as and λ determines the trade-off between increasing the margin size and ensuring that x lies on the correct side of the margin:

$$\frac{1}{2} * ||\vec{w}^2|| + \lambda * [\frac{1}{n} * \sum max(0, 1 - y(\vec{w} * \vec{x} - b))]$$

The reason for using SVM, instead of other classification method, like a neural network or bayesian network, is the fact that the prediction that needs to be done is binary, the patches of the image either contain signs of COVID-19 or not and it does not use probabilities like a bayesian network. Moreover, compared to a neural network the overall training time is faster, because the neural network needs to extract the feature vectors automatically.

3. EXPERIMENTS

3.1 Measurement Tools and Environment

In order to achieve a result for each research question presented, several python scripts have been created. The scripts use different libraries specialized for image processing and machine learning, like numpy, opencv and Scikit-learn. OpenCV and Numpy are used to extract different features from the images, while also taking care of several preprocessing tasks, like resizing, grayscale, threshold, colormap the image or extracting the feature vectors for the color. For training the classifier, separating the image into patches and extracting the texture feature vector, Scikit-learn library was used. The resulting data from each experiment is used to create different plots in order to evaluate each classifier and feature vectors used.

Table 1. Experimental setup for the first round of experiments

Name	Covid sample size	Non-covid sample size
Experiment 1	753177	753177
Experiment 2	753177	1506354
Experiment 3	753177	2259531
Experiment 4	753177	3012708

3.2 Evaluation

The evaluation of each feature determines which parameters chosen for the classifier and which feature vector is better for detecting the signs of the disease in CT scan images. The evaluation was done in two rounds of experiments. The dataset contains 100 images of pulmonary CT scans which show signs of the disease, which are split on average into 100000 16 by 16 patches. For training a total of 95 images are used and for testing the rest of 5 images are used. On each image from the dataset bounding boxes were drawn around the signs of COVID-19, so during feature extraction it can be checked whether a patch contains signs of COVID-19 or not. In Table 1 the setup for the first round of experiments can be seen, where the differences between each experiment reside in the difference between the sample sizes of the covid and non-covid patches. Because the total number of non-covid patches is bigger than the number of covid patches, the non-covid patches used for training are picked randomly. For each feature vector 9 SVM classifiers with different parameters are used. The first parameter used is regularization parameter (C), which adds information to the classifier in order to prevent overfitting. The values used for C are 1 or default value, 0.1, 0.01, 0.001 and 0.0001. The smaller the value of C is the chance of overfitting the data in training decreases, but the chance of underfitting increases. The second parameter used is class weights, which are either default (none), meaning that no class weights are applied, or balanced, meaning that in case of an unbalance in the sizes of the samples used the classifier will try to balance them.

The second round of experiments, which can be seen in Table 2, is used to check which lung extraction method is better for detecting the signs of the disease. In this round the features used are only color and the combination of all features and the parameters used are only class weights, which are either balanced or none, and C, which is either 1 or 0.1.

3.3 Metrics

The metrics used to evaluate the performance of each classifier are precision, recall and Matthews correlation coefficient.

3.3.1 Precision

The precision, also known as positive predictive value, calculates the fraction of relevant instances, or true positives, among the retrieved instances, so a bigger precision means that more patches of the same class were predicted correctly and fewer of different classes were predicted correctly.

3.3.2 Recall

The recall, also known as sensitivity, is the fraction of the total amount of relevant instances that were actually retrieved, which means that a bigger recall means that a bigger number of patches from the same class were predicted correctly or there is a bias towards that class.

3.3.3 Matthews correlation coefficient

Matthews correlation coefficient is used to measure the quality of a binary classification, in our case the one done by the SVM, which takes into account the true and false positives and negatives. If the value of the coefficient is bigger than 0 it means that the classifier is more balanced, if it is less than 0 it is very unbalanced and if it is 0 then it is not better than guessing the answer.

4. RESULTS

While training a classifier there are metrics that are used to assess the model, which are precision, recall and the Matthews correlation coefficient.

4.1 Precision

As seen in Table 3 in case of non-covid patches the maximum precision is achieved when the feature vectors are either for color or the combination of all three features. For color feature vectors, this is achieved when the weights are balanced, regardless of the experiment, or when the weights are none but only in Experiment 1. The highest precision value for the combination of all feature vectors is achieved when the weights are balanced, regardless of the experiment, or when the class weights are none and C is 0.1, 0.01 or 0.001 but only in Experiment 1. In the case of covid patches the biggest precision is achieved by the combination of all feature vectors when C is 0.1 and class weights are none in Experiment 4. The complete set of results from each experiment can be seen in Appendix A.

4.2 Recall

It can be seen in Table 4 that for the non-covid patches the biggest recall is achieved by every feature vector and the combination of all feature vectors in Experiment 2, 3 and 4 when the class weights are none, regardless of C. The biggest value of recall in the case of covid patches is achieved for the color feature vector when class weights are balanced and C is 0.0001 and when class weights are none and C is 0.0001. The complete set of results from each experiment can be seen in Appendix B.

4.3 Matthews Correlation Coefficient

As seen in Table 9 the biggest value for the Matthews correlation coefficient is achieved by the combination of all feature vectors when the class weights are not balanced and C is 0.1 in Experiment 3. In the second round of experiments the biggest value of the coefficient is achieved by the combination of all feature vectors as it can be seen in Table 6 in Experiment 5, when C is 0.1. The complete set of results from each experiment can be seen in Appendix C.

4.4 Lung Segmentation

In the second round of experiments the color feature vector achieves the highest coefficient value in Experiment 6, regardless of the parameters used, while for the combination of all feature vectors the biggest value is achieved when C is 0.1 in Experiment 5, regardless of the class weights value. Out of these two feature vectors, the biggest value was achieved by the combination of all feature vectors, as can be seen in Table 6.

4.5 Discussion

A big difference between this paper and other related work in detecting the signs of infection with the coronavirus is

Table 2. Experimental setup for the second round of experiments

Name	Segmentation method	Testing set
Experiment 5	Bounding boxes	Lung sections from bounding boxes
Experiment 6	Bounding boxes	Complete images
Experiment 7	Color mapping and thresholding	Lung sections from color mapping and thresholding
Experiment 8	Color mapping and thresholding	Complete images

Table 3. Maximum precision values

Feature	Non-covid value	Covid value
Color	1	0.09
Texture	0.99	0.09
Position	0.98	0.05
All features	1	0.33

Table 4. Maximum recall values

Feature	Non-covid value	Covid value
Color	1	1
Texture	1	0.8
Position	1	0.72
All features	1	1

Table 5. Matthews correlation coefficient values in the first round of experiments

Feature	Maximum value	Minimum value
Color	0.194	$\sim$ -0.007
Texture	0.14	$\sim$ -0.002
Position	0.07	0
All features	0.39	0

Table 6. Matthews correlation coefficient values in the second round of experiments

Feature	Maximum value	Minimum value
Color	0.19	0.13
All features	0.41	0.14

the classifier used. The most used classifier for detecting the signs of the virus are convolutional networks, which showed promising results. In the detection of COVID-19, when 3D images are used to extract the features of the affection caused by the disease, a convolutional network is used which had a recall rate of 0.96 for covid images, while detecting community acquired pneumonia the recall rate was 0.95 [2]. Another work, which combines 2D and 3D image analysis and the use of deep learning algorithms showed a recall rate 0.98 [3]. When using only 2D CT scan images it was found that by using two neural networks, one for extracting the features of the image and another for classifying the images, the internal recall rate, when using the validation set, for detecting the signs of the disease is 0.88, while the external recall, when using it is 0.83 [4]. Another method that showed good results, recall rate of 0.68, uses a combination of 2D CT scan images and different data about the patients, from whom the images came, like clinical symptoms, exposure history and laboratory testing [6]. It can be seen that all related work is image based, while this paper presents a method which uses patches of 2D CT scan images to detect the signs of the disease.

5. CONCLUSION

5.1 Research Questions Result

From the tables presented above, it can be seen that the position characteristic of the affection caused by COVID-19 had the lowest metric values, meaning that the feature vectors for position do not present sufficient data for the proper classification of the image. The feature vector for color showed the best results when it comes to using only one feature vector, which means that it can be used for future possibilities to detect the signs of the disease. The feature vector for texture is the second that showed the best results when it comes to using only one feature vector to detect the affection caused by COVID-19 on a CT scan image, which means that it can also be used for future possibilities. The combination of all three feature vectors showed the best results, which means that for future work this feature vector is best suited for detecting the signs of the disease. Even though the use of color mapping and thresholding to preprocess the CT scan images looked like it could easily extract the lung regions, in the end it was not better than manually drawing bounding boxes around the lung region.

5.2 Outlook

As it stands right now, beside the combination of all three features, the rest of the classifiers that use only one feature to detect the disease can be considered almost unbalanced, from the point of the correlation coefficient. The main solution that can improve the Matthews correlation coefficient of the classifiers is improving the feature vectors that are used for the classifier. By improving them it can be tested whether an improvement will either help the classifier or it will show that the feature is not completely relevant for detecting the affections of the disease. Another possible future update to the classifier is to instead of returning only a yes or no prediction, it should also return a score which shows the actual probability that a patch of the image shows COVID-19 signs or not. After a definitive feature vector has been found to define the patches that have signs of the disease, the classifier can move to the next step. Based on the amount of patches that are detected as covid and based on whether they form clusters, experiments can be done in order to detect whether the lungs in a CT scan image truly have signs of the disease or not.

6. REFERENCES

[1] W. cai Dai et al. Ct imaging and differential diagnosis of covid-19 - wei-cai dai, han-wen zhang, juan yu, hua-jian xu, huan chen, si-ping luo, hong zhang, li-hong liang, xiao-liu wu, yi lei, fan lin. March 2020.

[2] L. L. et al. Artificial intelligence distinguishes covid-19 from community acquired pneumonia on chest ct. March 2020.

[3] S. et al. Ai ct scan analysis for covid-19 detection and patient monitoring. March 2020.

[4] S. W. et al. A deep learning algorithm using ct images to screen for corona virus disease (covid-19). January 2020.

[5] A. K. E. Jianping Fan, D. K. Y. Yau and W. G. Aref. Automatic image segmentation by integrating color-edge extraction and seeded region growing. *IEEE Transactions on Image Processing*, 10(10):1454–1466, October 2001.

[6] D. K. e. a. Mei X., Lee H. Artificial intelligence–enabled rapid diagnosis of patients with covid-19. 2020.

[7] S. B. Sana Salehi, Aidin Abedi and A. Gholamrezanezhad. Coronavirus disease 2019 (covid-19): A systematic review of imaging findings in 919 patients. *American Journal of Roentgenology*, pages 1–7, 2019.

[8] O. G. Seletchi E. D. Duliu. Image processing and data analysis in computed tomography. *Rom. Journ. Phys.*, 52(5-7):667–675, 2007.

[9] L. R. Varshney. Abdominal organ segmentation in ct scan images : A survey: Semantic scholar. January 1970.

APPENDIX

A. PRECISION

Table 7. Precision values

Feature vector	Class weights	C	Exp.1 non-covid	Exp.2 non-covid	Exp.3 non-covid	Exp.4 non-covid	Exp.1 covid	Exp.2 covid	Exp.3 covid	Exp.4 covid
color	balanced	1	1	1	1	1	0.07	0.07	0.07	0.07
color	none	0.1	1	0.99	0.97	0.97	0.07	0.09	0.04	0
color	none	0.01	1	0.97	0.97	0.97	0.07	0.08	0	0
color	none	0.001	1	0.97	0.97	0.97	0.06	0	0	0
color	none	0.0001	1	0.97	0.97	0.97	0.05	0	0	0
color	balanced	0.1	1	1	1	1	0.07	0.07	0.07	0.07
color	balanced	0.01	1	1	1	1	0.07	0.07	0.07	0.07
color	balanced	0.001	1	1	1	1	0.06	0.06	0.06	0.06
color	balanced	0.0001	1	1	1	1	0.05	0.05	0.06	0.06
texture	balanced	1	0.99	0.99	0.99	0.99	0.06	0.06	0.06	0.06
texture	none	0.1	0.99	0.97	0.97	0.97	0.06	0.08	0.06	0.06
texture	none	0.01	0.99	0.97	0.97	0.97	0.06	0.09	0.06	0.05
texture	none	0.001	0.99	0.97	0.97	0.97	0.06	0.08	0.03	0
texture	none	0.0001	0.98	0.97	0.97	0.97	0.05	0.05	0	0
texture	balanced	0.1	0.99	0.99	0.99	0.99	0.06	0.06	0.06	0.06
texture	balanced	0.01	0.99	0.99	0.99	0.99	0.06	0.06	0.06	0.06
texture	balanced	0.001	0.99	0.99	0.99	0.99	0.06	0.06	0.06	0.06
texture	balanced	0.0001	0.98	0.99	0.99	0.99	0.05	0.05	0.05	0.06
position	balanced	1	0.98	0.98	0.98	0.98	0.05	0.05	0.05	0.05
position	none	0.1	0.98	0.97	0.97	0.97	0.04	0	0	0
position	none	0.01	0.98	0.97	0.97	0.97	0.04	0	0	0
position	none	0.001	0.98	0.97	0.97	0.97	0.04	0	0	0
position	none	0.0001	0.97	0.97	0.97	0.97	0.04	0	0	0
position	balanced	0.1	0.98	0.98	0.98	0.98	0.04	0.04	0.04	0.04
position	balanced	0.01	0.98	0.98	0.98	0.98	0.04	0.04	0.04	0.04
position	balanced	0.001	0.98	0.98	0.98	0.98	0.04	0.04	0.04	0.04
position	balanced	0.0001	0.97	0.97	0.97	0.98	0.04	0.04	0.04	0.04
all in one	balanced	1	1	1	1	1	0.12	0.12	0.12	0.12
all in one	none	0.1	1	0.99	0.99	0.98	0.12	0.22	0.29	0.33
all in one	none	0.01	1	0.99	0.98	0.97	0.11	0.22	0.29	0.26
all in one	none	0.001	1	0.97	0.97	0.97	0.08	0.13	0.01	0
all in one	none	0.0001	0.99	0.97	0.97	0.97	0.05	0	0	0
all in one	balanced	0.1	1	1	1	1	0.12	0.12	0.12	0.12
all in one	balanced	0.01	1	1	1	1	0.11	0.11	0.11	0.12
all in one	balanced	0.001	1	1	1	1	0.08	0.08	0.09	0.09
all in one	balanced	0.0001	0.99	1	1	1	0.05	0.06	0.06	0.06

B. RECALL

Table 8. Recall values

Feature vector	Class weights	C	Exp.1 non-covid	Exp.2 non-covid	Exp.3 non-covid	Exp.4 non-covid	Exp.1 covid	Exp.2 covid	Exp.3 covid	Exp.4 covid
color	balanced	1	0.62	0.62	0.62	0.62	0.92	0.92	0.92	0.92
color	none	0.1	0.6	0.77	0.99	1	0.93	0.67	0	0
color	none	0.01	0.56	0.92	1	1	0.97	0.2	0	0
color	none	0.001	0.49	1	1	1	0.98	0	0	0
color	none	0.0001	0.4	1	1	1	1	0	0	0
color	balanced	0.1	0.6	0.6	0.6	0.6	0.93	0.93	0.93	0.92
color	balanced	0.01	0.56	0.57	0.58	0.58	0.97	0.97	0.96	0.96
color	balanced	0.001	0.49	0.51	0.52	0.53	0.98	0.98	0.98	0.98
color	balanced	0.0001	0.4	0.42	0.43	0.44	1	1	1	1
texture	balanced	1	0.6	0.6	0.6	0.6	0.8	0.8	0.8	0.8
texture	none	0.1	0.6	0.95	1	1	0.8	0.15	0	0
texture	none	0.01	0.6	0.95	1	1	0.8	0.15	0	0
texture	none	0.001	0.6	0.97	1	1	0.8	0.09	0	0
texture	none	0.0001	0.6	0.99	1	1	0.77	0.01	0	0
texture	balanced	0.1	0.6	0.6	0.6	0.6	0.8	0.8	0.8	0.8
texture	balanced	0.01	0.6	0.6	0.6	0.6	0.8	0.8	0.8	0.8
texture	balanced	0.001	0.6	0.6	0.6	0.6	0.8	0.8	0.8	0.8
texture	balanced	0.0001	0.5	0.52	0.54	0.55	0.77	0.78	0.79	0.8
position	balanced	1	0.52	0.52	0.52	0.52	0.68	0.68	0.68	0.68
position	none	0.1	0.47	1	1	1	0.7	0	0	0
position	none	0.01	0.47	1	1	1	0.72	0	0	0
position	none	0.001	0.47	1	1	1	0.71	0	0	0
position	none	0.0001	0.53	1	1	1	0.54	0	0	0
position	balanced	0.1	0.47	0.48	0.49	0.49	0.7	0.7	0.7	0.7
position	balanced	0.01	0.44	0.44	0.45	0.45	0.72	0.72	0.72	0.71
position	balanced	0.001	0.44	0.44	0.43	0.43	0.71	0.71	0.72	0.72
position	balanced	0.0001	0.53	0.53	0.5	0.49	0.54	0.58	0.61	0.63
all in one	balanced	1	0.77	0.77	0.77	0.77	0.95	0.94	0.94	0.94
all in one	none	0.1	0.76	0.9	0.95	0.98	0.95	0.81	0.6	0.33
all in one	none	0.01	0.77	0.91	0.95	0.98	0.95	0.76	0.4	0.15
all in one	none	0.001	0.6	0.98	1	1	0.95	0.15	0	0
all in one	none	0.0001	0.46	1	1	1	0.93	0	0	0
all in one	balanced	0.1	0.76	0.77	0.77	0.77	0.97	0.96	0.97	0.96
all in one	balanced	0.01	0.73	0.74	0.74	0.75	0.98	0.98	0.98	0.97
all in one	balanced	0.001	0.6	0.63	0.65	0.67	0.98	0.99	0.99	0.98
all in one	balanced	0.0001	0.46	0.48	0.49	0.51	0.93	0.94	0.94	0.95

C. MATTHEWS CORRELATION COEFFICIENT

Table 9. Matthews correlation coefficient values

Feature vector	Class weights	C	Exp.1	Exp.2	Exp.3	Exp.4	Exp.5	Exp.6	Exp.7	Exp.8
color	balanced	1	0.19	0.19	0.19	0.19	0.16	0.19	0.13	0.18
color	none	0.1	0.19	0.17	0.06	-0.007	0.16	0.19	0.13	0.18
color	none	0.01	0.19	0.08	0	0	-	-	-	-
color	none	0.001	0.16	0	0	0	-	-	-	-
color	none	0.0001	0.14	0	0	0	-	-	-	-
color	balanced	0.1	0.194	0.193	0.193	0.193	0.16	0.19	0.13	0.18
color	balanced	0.01	0.19	0.192	0.193	0.194	-	-	-	-
color	balanced	0.001	0.167	0.172	0.174	0.18	-	-	-	-
color	balanced	0.0001	0.14	0.15	0.152	0.153	-	-	-	-
texture	balanced	1	0.138	0.138	0.14	0.14	-	-	-	-
texture	none	0.1	0.14	0.07	0.01	0.001	-	-	-	-
texture	none	0.01	0.14	0.07	0.009	0.001	-	-	-	-
texture	none	0.001	0.14	0.05	-0.001	-0.001	-	-	-	-
texture	none	0.0001	0.09	0.007	-0.002	0	-	-	-	-
texture	balanced	0.1	0.14	0.14	0.14	0.14	-	-	-	-
texture	balanced	0.01	0.14	0.14	0.14	0.14	-	-	-	-
texture	balanced	0.001	0.14	0.14	0.14	0.14	-	-	-	-
texture	balanced	0.0001	0.09	0.1	0.11	0.12	-	-	-	-
position	balanced	1	0.07	0.07	0.07	0.07	-	-	-	-
position	none	0.1	0.06	0	0	0	-	-	-	-
position	none	0.01	0.05	0	0	0	-	-	-	-
position	none	0.001	0.05	0	0	0	-	-	-	-
position	none	0.0001	0.02	0	0	0	-	-	-	-
position	balanced	0.1	0.062	0.064	0.065	0.068	-	-	-	-
position	balanced	0.01	0.055	0.056	0.057	0.058	-	-	-	-
position	balanced	0.001	0.055	0.054	0.053	0.053	-	-	-	-
position	balanced	0.0001	0.024	0.033	0.04	0.043	-	-	-	-
all in one	balanced	1	0.29	0.288	0.286	0.285	0.36	0.29	0.14	0.16
all in one	none	0.1	0.29	0.38	0.39	0.30	0.41	0.32	0.18	0.24
all in one	none	0.01	0.27	0.38	0.32	0.15	-	-	-	-
all in one	none	0.001	0.2	0.08	-0.001	0	-	-	-	-
all in one	none	0.0001	0.14	0	0	0	-	-	-	-
all in one	balanced	0.1	0.295	0.297	0.297	0.297	0.41	0.32	0.18	0.24
all in one	balanced	0.01	0.27	0.281	0.284	0.287	-	-	-	-
all in one	balanced	0.001	0.2	0.22	0.23	0.24	-	-	-	-
all in one	balanced	0.0001	0.14	0.147	0.154	0.16	-	-	-	-